

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 730 951**

51 Int. Cl.:

C12Q 1/6883 (2008.01)

C12Q 1/6874 (2008.01)

C07K 16/00 (2006.01)

C12N 15/10 (2006.01)

C07K 16/06 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **11.10.2011 PCT/US2011/055801**

87 Fecha y número de publicación internacional: **12.04.2012 WO12048340**

96 Fecha de presentación y número de la solicitud europea: **11.10.2011 E 11831760 (1)**

97 Fecha y número de publicación de la concesión europea: **13.03.2019 EP 2625295**

54 Título: **Secuenciación inmune de alto rendimiento**

30 Prioridad:

08.10.2010 US 391337 P

45 Fecha de publicación y mención en BOPI de la traducción de la patente:

13.11.2019

73 Titular/es:

**PRESIDENT AND FELLOWS OF HARVARD
COLLEGE (100.0%)
17 QUINCY STREET
CAMBRIDGE, US**

72 Inventor/es:

**CHURCH, GEORGE M.;
VIGNEAULT, FRANCOIS;
LASERSON, URI y
BACHELET, IDO**

74 Agente/Representante:

SÁEZ MAESO, Ana

ES 2 730 951 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

DESCRIPCIÓN

Secuenciación inmune de alto rendimiento

Datos de solicitud relacionados

5 Esta solicitud reivindica prioridad a la Solicitud de Patente Provisional de los Estados Unidos No. 61/391,337, presentada el 8 de octubre de 2010.

Campo

La presente invención se refiere a métodos y composiciones para diagnósticos y terapéuticos.

Antecedentes

10 La función del sistema inmunitario adaptativo está mediada en gran medida por linfocitos (células B y T) que expresan un repertorio diverso de receptores inmunitarios contra prácticamente cualquier sustancia extraña (Abbas et al., Cellular and Molecular Immunology, Ed. 6th. (Saunders Elsevier, Filadelfia; 2007); Murphy et al., Janeway's Immunobiology, Ed. 7 (Garland Science, Nueva York; 2008); Paul, Fundamental Immunology, Ed. 6 (Wolters Kluwer/Lippincott Williams & Wilkins, Filadelfia; 2008)). Con el fin de generar el repertorio de anticuerpos necesarios para el reconocimiento de antígenos, cada linfocito construye independientemente un receptor único a través del
 15 proceso de recombinación VDJ. Id. Cada célula selecciona aleatoriamente un solo segmento de gen V, D y J mediante recombinación genética, introduciendo nucleótidos adicionales no codificados en la línea germinal en las uniones (Figura 1). Este proceso crea diversidad de anticuerpos, la mayoría de los cuales está codificada en la región 3 determinante de la complementariedad de la cadena pesada (CDR3) (Rock et al. (1994) J. Exp. Med. 179: 323; Xu and Davis (2000) Immunity 13: 37). La complejidad y la dinámica de un solo repertorio inmune humano aún no se han
 20 investigado profundamente, y la capacidad de rastrear el repertorio dinámicamente no se ha demostrado en ningún organismo.

Resumen

25 La caracterización de la diversidad y la dinámica del repertorio inmune o "VDJ-oma" tiene implicaciones significativas en la comprensión del sistema inmune, particularmente en el contexto de diagnósticos personalizados y descubrimiento terapéutico. Por consiguiente, la presente invención está dirigida en parte a métodos y composiciones que permiten diagnósticos clínicos y terapéuticos personalizados utilizando la información única de secuencia de ácidos nucleicos contenida en un sistema inmune individual. Con este objetivo, se proporcionan métodos y composiciones de alto rendimiento y alta resolución para perfilar el repertorio de anticuerpos de un solo individuo. Por ejemplo, las realizaciones de la presente invención incluyen el uso de tecnología de secuenciación de alto rendimiento
 30 para rastrear dinámicamente la naturaleza y el alcance de los anticuerpos producidos por un individuo como resultado de una reacción inmune, para clonar anticuerpos completos de cadenas livianas y pesadas emparejadas de millones de células individuales en paralelo, y para seleccionar anticuerpos de alta afinidad contra múltiples antígenos en una sola reacción.

35 En ciertas realizaciones de ejemplo, se proporcionan métodos para determinar un estado inmune de un individuo. Los métodos incluyen los pasos para obtener secuencias de ácidos nucleicos que codifican receptores de linfocitos de la muestra biológica y realizar una secuenciación de alto rendimiento de las secuencias de ácidos nucleicos amplificadas para determinar una pluralidad de secuencias que representan el estado inmune de la muestra biológica. En ciertos aspectos, la muestra biológica se selecciona del grupo que consiste en sangre, saliva, líquido sinovial, células cultivadas, una sección de tejido y una biopsia. En ciertos aspectos, la muestra biológica se obtiene de un individuo.
 40 En otros aspectos, se determina un estado inmune del individuo. En aún otros aspectos, los métodos incluyen además la etapa de amplificar las secuencias de ácidos nucleicos que codifican receptores de linfocitos antes de la etapa de secuenciación de alto rendimiento, usando opcionalmente cebadores específicos para secuencias de ácidos nucleicos de cadena pesada o cadena liviana. En otros aspectos, los métodos incluyen la etapa de obtención de linfocitos (por ejemplo, células T, células B o una combinación de las mismas) de la muestra biológica antes de la etapa de purificación. De acuerdo con un aspecto, el uso del receptor de células T de un individuo se correlaciona con el tipo MHC del individuo para diagnosticar ciertas enfermedades o afecciones. En ciertos aspectos, las secuencias de ácidos nucleicos codifican una o más de las regiones V (cadena pesada o cadena liviana), regiones D (cadena pesada o cadena liviana), regiones J (cadena pesada o cadena liviana) y combinaciones de las mismas. En otros aspectos, las secuencias de ácidos nucleicos son ADN (por ejemplo, ADN genómico) o ARN (por ejemplo, ARNm). En otros
 45 aspectos, los métodos incluyen la etapa de comparar el estado inmune del individuo con una muestra de control para diagnosticar una enfermedad o un trastorno tal como, por ejemplo, una infección, un trastorno autoinmune, un trastorno proliferativo celular y cualquier combinación de los mismos. En otros aspectos, la pluralidad de secuencias incluye al menos una secuencia de ácidos nucleicos común.

55 En ciertas realizaciones de ejemplo, se proporcionan métodos para controlar el efecto de un agente en un estado inmune de una muestra biológica. Los métodos incluyen los pasos de aislar linfocitos de la primera muestra biológica, purificar un primer conjunto de secuencias de ácidos nucleicos que codifican receptores de linfocitos, realizar una secuenciación de alto rendimiento del primer conjunto purificado de secuencias de ácidos nucleicos para proporcionar

una primera biblioteca de referencia, en contacto con un segundo muestra biológica con un agente, aislando linfocitos de la segunda muestra biológica, purificando un segundo conjunto de secuencias de ácidos nucleicos que codifican receptores de linfocitos, realizando una secuenciación de alto rendimiento del segundo conjunto purificado de secuencias de ácidos nucleicos para proporcionar una segunda biblioteca de referencia y comparando secuencias de la primera biblioteca de referencia y la segunda biblioteca de referencia para monitorizar el efecto de un agente en un estado inmune de la muestra biológica. En ciertos aspectos, el agente se selecciona del grupo que consiste en un antígeno, una vacuna, un fármaco, una molécula pequeña, un agente biológico y cualquier combinación de los mismos. En otros aspectos, la muestra biológica se selecciona del grupo que consiste en sangre, saliva, líquido sinovial, células cultivadas, una sección de tejido y una biopsia. En otros aspectos, la muestra biológica se obtiene de un individuo. En otros aspectos, se determina un estado inmune del individuo. En otros aspectos, el individuo padece una enfermedad o trastorno seleccionado del grupo que consiste en una infección, un trastorno autoinmune, un trastorno proliferativo celular y cualquier combinación de los mismos. En ciertos aspectos, se determina la eficacia del agente para tratar una enfermedad o un trastorno. En otros aspectos, se determina un pronóstico para tratar una enfermedad o un trastorno. En otros aspectos, los pasos de secuenciación se realizan simultáneamente o en momentos separados. En ciertos aspectos, se obtienen múltiples muestras a lo largo de un curso de tiempo y, opcionalmente, se realizan los pasos de aislamiento, purificación y ejecución de secuenciación de alto rendimiento para cada una de las múltiples muestras obtenidas a lo largo del curso de tiempo.

En ciertas realizaciones de ejemplo, se proporcionan métodos para controlar una respuesta inmune de una muestra biológica. Los métodos incluyen los pasos de obtener una primera muestra biológica en un primer momento, aislar linfocitos de la primera muestra biológica, purificar un primer conjunto de secuencias de ácidos nucleicos que codifican receptores de linfocitos, realizar una secuenciación de alto rendimiento del primer conjunto purificado de secuencias de ácidos nucleicos para proporcionar una primera biblioteca de referencia, obteniendo una segunda muestra biológica en un segundo punto de tiempo, aislando linfocitos de la segunda muestra biológica, purificando un segundo conjunto de secuencias de ácidos nucleicos que codifican receptores de linfocitos, realizando una secuenciación de alto rendimiento del segundo conjunto purificado de secuencias de ácidos nucleicos para proporcionar una segunda biblioteca de referencia, y comparar las secuencias de la primera biblioteca de referencia y la segunda biblioteca de referencia para rastrear una respuesta inmune en una muestra biológica. En ciertos aspectos, la respuesta inmune es contra un agente seleccionado del grupo que consiste en un antígeno, una vacuna, un fármaco, una molécula pequeña, un agente biológico y cualquier combinación de los mismos. En otros aspectos, los pasos de secuenciación se realizan simultáneamente o en momentos separados. En otros aspectos, se obtienen múltiples muestras a lo largo de un curso de tiempo y, opcionalmente, se realizan los pasos de aislamiento, purificación y ejecución de secuenciación de alto rendimiento para cada una de las múltiples muestras obtenidas a lo largo del curso de tiempo. En otros aspectos, se determina un diagnóstico de enfermedad o trastorno. En otros aspectos, la muestra biológica se selecciona del grupo que consiste en sangre, saliva, líquido sinovial, células cultivadas, una sección de tejido y una biopsia. En otros aspectos, la muestra biológica se obtiene de un individuo. En otros aspectos, se determina el estado inmune del individuo.

Otras características y ventajas de ciertas realizaciones de la presente invención se harán más evidentes en la siguiente descripción de las realizaciones y dibujos de las mismas, y a partir de las reivindicaciones.

Breve descripción de los dibujos

El archivo de patente o solicitud contiene al menos un dibujo ejecutado en color. La Oficina proporcionará copias de esta publicación de patente o solicitud de patente con dibujos en color, previa solicitud y pago de la tasa necesaria. Las anteriores y otras características y ventajas de la presente invención se entenderán más completamente a partir de la siguiente descripción detallada de las realizaciones ilustrativas tomadas junto con los dibujos adjuntos, en los que:

La Figura 1 representa una visión general de la recombinación VDJ y el diseño experimental. Un anticuerpo contiene una cadena pesada y una cadena liviana, donde en un segmento único V, D y J se unen a partir de un conjunto de genes codificados en la línea germinal.

Las Figuras 2A-2E representan la reproducibilidad técnica y la diversidad inmune de ciertos métodos descritos en este documento. (A) El número de clones únicos (arriba) y las lecturas correspondientes (abajo) compartidas entre dos ejecuciones de repetición de secuencia (SR1 y SR2) y una ejecución de replicación técnica independiente (TR1). Un pequeño número de clones únicos compartidos representaron la mayoría de las lecturas. (B) El gráfico de dispersión del día 1 de SR1 frente a TR1 de las frecuencias de los clones demostró una fuerte correlación y reproducibilidad de la metodología de muestreo descrita aquí. (C) Para los tamaños de muestra múltiples, se tomaron muestras de las distribuciones iniciadas mediante el cálculo de la correlación de Pearson entre 50 pares de muestras extraídas independientemente. Esto demostró que 10^5 lecturas eran suficientes para lograr altas correlaciones entre muestras independientes. (D) Las distribuciones de abundancia de frecuencias de clones de replicado de diversidad 1 (DR1) y todos los puntos de tiempo de SR1+SR2+TR1. Las frecuencias de clonación abarcaban múltiples órdenes de magnitud y se distribuían mediante ley de potencia con una escala de cuadrados aproximadamente inversa. (E) Se observaron secuencias únicas y se realizaron estimaciones de la diversidad de la cadena pesada humana expresada en muestras de sangre de 1 mL utilizando los estimadores no paramétricos Chao1 y ACE. Los números se calcularon utilizando recuentos de clones (posterior al agrupamiento, corrección de errores), así como secuencias de unión CDR3 únicas

(sin procesar, sin agrupamiento) (arriba). Estimaciones de la diversidad total de la cadena pesada humana expresada en sangre utilizando múltiples estimadores no paramétricos y el método de Arstila (Arstila et al. (1999) Science 286: 958) (abajo).

5 Las figuras 3A-3C representan el análisis del repertorio estático. (A) La distribución de todos los segmentos funcionales V y J observados en ejecuciones repetidas en todos los puntos de tiempo muestra un uso altamente no uniforme de los segmentos genéticos. Los tamaños de los círculos son proporcionales a los recuentos de lectura. (B) La distribución de la longitud de la unión CDR3 de todas las lecturas en SR1+SR2+TR1 (luz) y todas las uniones de cadena pesada de Ig en IMGT/LIGM (oscuro) mostraron una gran variabilidad en el tamaño del bucle de CDR3. El patrón de diente de sierra reflejaba uniones en el marco. (C) Ejemplos representativos de tipos de pseudoespectros de CDR3 para 10 varias combinaciones V-J en múltiples puntos de tiempo. El pico de 16 aminoácidos se anotó en todos los tipos de espectros. Los números anotados mostraron expansiones de clones identificados en la Figura 4B, lo que confirmó la utilidad del método de tipos de espectros para identificar candidatos de interés.

15 Las figuras 4A-4D representan un análisis dinámico del repertorio. (A) La serie de tiempos de todos los clones en SR1+SR2+TR1 reveló fluctuaciones significativas del repertorio a través del tiempo. Un puñado de clones mostró una expresión significativamente mayor, mientras que la magnitud de las fluctuaciones se muestra en una escala logarítmica (recuadro). La aparición de muchos clones en aumento gradual desde el día -14 hasta el día 0 se debió a la falta de resolución de muestreo entre estos puntos temporales. (B) Agrupación de mapas de organización automática (SOM) de series de tiempo de clonación, donde cada agrupación se ubicó en una posición de mapa constante, y los colores representan niveles de expresión de clonación. Se seleccionaron grupos de grupos con 20 comportamiento dinámico similar y los clones correspondientes se representaron como series de tiempo. Las series temporales se colorearon según el isotipo más común observado en el momento de la expresión máxima. Las anotaciones se refieren a clones específicos de interés correspondientes a picos de tipos de espectros específicos en la Figura 3C. (C) La dinámica de la distribución global de isotipos indicó un cambio de clase potencial. Las líneas gruesas representan la media de 6 ejecuciones repetidas (mostradas como líneas finas) con una desviación estándar. 25 (D) Distribuciones de isotipos específicos para clones anotados seleccionados; los gráficos de barras apilados representan la distribución de isotipos en un punto de tiempo determinado, mientras que la línea negra representa la serie de tiempos de la frecuencia de clonación. Números referidos a la frecuencia máxima de clonación, con el número correspondiente de lecturas. El esquema de color del isotipo es consistente entre los paneles B, C y D.

30 La Figura 5 muestra esquemáticamente la secuenciación inmune de alto rendimiento y el análisis del repertorio inmune y de anticuerpos.

Las Figuras 6A-6C representan gráficamente el análisis de clasificación FACS de las secuencias seleccionadas de la Figura 4.

Las Figuras 7A-7C representan gráficamente el análisis por microbalanza de cristal de cuarzo (QCM).

35 La Figura 8 muestra un gráfico que muestra curvas ROC para el algoritmo de alineación VDJ descrito aquí utilizando los resultados de IMGT/V-QUEST como referencia. Como era de esperar, las alineaciones de la región V fueron las más confiables, seguidas de J, seguidas de D.

40 La Figura 9 muestra diagramas de dispersión que muestran la correlación de los niveles de expresión entre las réplicas SRI, SR2, TR1 y OR3. Los puntos en rojo tienen valor cero. El código de barras 1 (-14 d) se perdió significativamente en la preparación de la réplica TR1. Tenga en cuenta que OR3 mostró consistentemente peor correlación. Como se produjo con un conjunto de cebadores recién mezclados, sin pretender vincularse a la teoría científica, esto indicó que el protocolo descrito en el presente documento era altamente sensible a las concentraciones de cebadores.

45 La Figura 10 muestra gráficamente los coeficientes de variación (CV) calculados para diversas comparaciones repetidas. Las frecuencias de clones más altas mostraron un CV más pequeño, consistente con la mayor reproducibilidad de los clones de alta expresión. Los valores de CV de 0.5 se alcanzaron en frecuencias de 10^{-4} a 10^{-3} , lo que indica límites a la capacidad de cuantificar con precisión la frecuencia de clonación antes de la aparición del ruido de Poisson.

La Figura 11 muestra gráficamente la acumulación de rangos de frecuencia de clones para la replicación de DR1 y los diversos puntos de tiempo de SR1+SR2+TR1.

50 Las Figuras 12A-12D representan curvas de rarefacción utilizando muestreo sin reemplazo. Los valores finales (calculados sobre los conjuntos de datos completos) se resumen aquí. (A) Chao 1, eje lineal y logarítmico. (B) ACE, eje lineal y logarítmico. (C) Número de secuencias únicas descubiertas, ejes lineales y logarítmicos. (D) Valor del estimador frente al número de lecturas. La falta de correlación positiva indica que los valores del estimador son independientes de los tamaños de la muestra y, por lo tanto, se muestrean lo suficiente.

55 Las Figuras 13A-13B representan pseudotipos de espectros de CDR3 para varias combinaciones de V-J en múltiples puntos de tiempo para todos los clones seleccionados de la Figura 4B.

La Figura 14 muestra diagramas replicados de series de tiempo de clones seleccionados al azar. Cada panel contiene 10 clones seleccionados al azar, trazados utilizando datos de SRI, SR2 y TR1, por separado. Los clones únicos se trazan en el mismo color. Los clones se rastrearon cualitativamente entre réplicas separadas, mostrando una alta reproducibilidad de series de tiempo entre réplicas.

- 5 Las Figuras 15A-15B representan medias k de agrupamiento de series de tiempo de clonación. (A) Agrupación de todos los clones en 100 agrupamientos utilizando la métrica de distancia euclidiana. (B) La agrupación similar de todos los clones observados en al menos dos puntos de tiempo. Las curvas representan medias de grupo; su grosor es proporcional al tamaño del agrupamiento.

- 10 Las Figuras 16A-16C representan distribuciones de isotipos de los clones más altamente expresados en los días 1 (A), 7 (B) y 21 (C). Los gráficos de barras apiladas representan la distribución de isotipos en un punto de tiempo determinado, mientras que la línea negra representa la serie temporal de frecuencia de clonación. Los números se refieren a la frecuencia máxima de clonación, con el número correspondiente de lecturas. IgA; verde, IgG; rojo, IgM; azul.

- 15 Las Figuras 17A-17B muestran la representación Collier de Perles (Ruiz and Lefranc (2002) Immunogenetics 53:857) de la entrada IMGTLIGM anti-digoxina (AR161172; arriba) y de la lectura de la secuencia correspondiente encontrada en un sujeto (FXQ8H8O01CY52L; inferior).

La Figura 18 muestra gráficamente la distribución de longitud de lectura en bruto de alta calidad de una única ejecución de secuenciación 454 GS FLX. El pico más significativo ocurrió en el tamaño esperado de 420 pb.

- 20 Las Figuras 19A-19B representan (A) la distribución de distancias de cofenéticas para agrupamiento jerárquico de enlaces simples y completos en secuencias de una combinación V-J con un pequeño número de lecturas; y (B) el número de agrupaciones obtenidas en función de la distancia a la que se cortó el árbol de enlace. Ambas gráficas muestran descensos rápidos a 4.5 ediciones, seguidas de hombros. Esto indica 4.5 ediciones como la distancia adecuada para recortar el árbol de enlaces para definir los clones.

- 25 Las Figuras 20A-20B representan (A) la distribución de log (cambio de múltiplo) para cada transición en series de tiempo que fueron positivas en los 8 puntos de tiempo; y (B) correlación entre la desviación estándar del log (cambio de múltiplo) y la duración de la transición.

- 30 La Figura 21 muestra un primer ejemplo de alineación para la hepatitis. Consulta: FXQ8H8O01DXEUI rango=0514859 x=1493.0 y=2520.0 longitud=408; Objetivo: 155621 | antígeno de superficie anti-virus de hepatitis B (VHB) (HBsAg) (humano); Modelo: afin:local:dna2dna; Puntaje bruto: 1740; Rango de consulta: 8 -> 398; Rango objetivo: 27 -> 423; aln_resumen: FXQ8H8O01DXEUI 408 8 398 + 155621 423 27 423 + 1740 390 372 95.38.

- La figura 22 representa un segundo ejemplo de alineación para la hepatitis. Consulta: FX4D8HT02FMIIW rango=0345098 x=2189.0 y=2486.0 longitud=408; Objetivo: 155621 | antígeno de superficie anti-virus de hepatitis B (VHB) (HBsAg) (humano); Modelo: afin:local:dna2dna; Puntaje bruto: 1743; Rango de consulta: 5 -> 404; Rango objetivo: 24 -> 423; aln_resumen: FX4D8HT02FMIIW 408 5 404 + 155621 423 24 423 + 1743 399 371 92.

- 35 La Figura 23 representa una alineación de solo unión. Consulta: FXQ8H8O01CY52L [traducir(1)]; Objetivo: AR161172 | anti-digoxina; Modelo: protein2dna: local; Puntaje bruto: 58; Rango de consulta: 0 -> 9; Rango objetivo: 237 -> 264; aln_resumen: FXQ8H8O01CY52L 9 0 9. AR161172 282 237 264 + 58 9 9 100.00.

- 40 La Figura 24 muestra una alineación de lectura completa. Consulta: FXQ8H8O01CY52L; Objetivo: AR161172 | anti-digoxina; Modelo: coding2coding; Puntaje bruto: 512; Rango de consulta: 73 -> 355; Rango objetivo: 0 -> 282; aln_resumen: FXQ8H8O01CY52L 390 73 355 + AR161172 282 0 282 + 512 94 93 98.94.

- 45 La Figura 25 representa una vista general de un diseño experimental de acuerdo con ciertos aspectos de la invención. El ARN total de linfocitos se extrajo de muestras de sangre recolectadas en múltiples puntos de tiempo después de la vacunación de un individuo humano. Se utilizó un conjunto multiplex de cebadores específicos de genes para amplificar la cadena pesada de inmunoglobulina del ARN de células B y se procesó en bibliotecas de secuenciación compatibles con 454. Las secuencias fueron alineadas y agrupadas para el análisis del repertorio.

La Figura 26 muestra una tabla que representa la distribución y el análisis de la secuencia de lectura.

La Figura 27 representa esquemáticamente métodos de ejemplo de acoplamiento de cadena.

La Figura 28 muestra esquemáticamente métodos de ejemplo de aislamiento de células.

La figura 29 muestra esquemáticamente métodos de ejemplo para acoplar cadenas pesadas y livianas.

- 50 La Figura 30 muestra esquemáticamente métodos de ejemplo para selecciones de afinidad multiplex.

La Figura 31 representa una célula inmune que presenta un anticuerpo que contiene una cadena pesada y una cadena liviana.

- La Figura 32 muestra esquemáticamente un método para optimizar la PCR de entrecruzamiento.
- La Figura 33 muestra esquemáticamente un método para optimizar el aislamiento de la celda.
- La Figura 34 muestra esquemáticamente un método para caracterizar la contaminación cruzada.
- 5 La Figura 35 muestra esquemáticamente un método para caracterizar la contaminación cruzada utilizando un ensayo TaqMan.
- La Figura 36 representa esquemáticamente la generación de una tabla de búsqueda de antígeno-anticuerpo para el virus de la influenza.
- La Figura 37 representa esquemáticamente anticuerpos acoplados a antígenos.
- 10 La Figura 38 representa esquemáticamente una comparación de bibliotecas con bibliotecas para la selección combinatoria de pares de antígeno-anticuerpo replicantes (Bowley et al. (2009) Proc. Natl. Acad. Sci. USA 106(5): 1380).
- La Figura 39 representa esquemáticamente métodos de captura de emulsión de perlas a modo de ejemplo.
- La Figura 40 representa esquemáticamente la transfección de oligonucleótidos pequeños de células individuales.
- La Figura 41 muestra esquemáticamente SOE-PCR.
- 15 La Figura 42 muestra esquemáticamente la ligación de SOE-PCR USER.
- La Figura 43 representa esquemáticamente la ligación de restricción de la SOE-PCR.
- La Figura 44 muestra esquemáticamente 3x SOE-PCR.
- La Figura 45 representa gráficamente las distribuciones de cadenas pesadas y livianas.
- La figura 46 muestra histogramas de longitudes de lectura de cadenas livianas.
- 20 La figura 47 muestra la distribución de uniones.
- La figura 48 representa una serie de tiempos de clonación.
- La Figura 49 muestra una serie de tiempos para cadenas pesadas y livianas que usan uniones únicas.
- La Figura 50 muestra una nueva serie temporal de uniones de cadenas pesadas y livianas altamente expresadas.
- La Figura 51 muestra un gel que muestra la ligación del adaptador Y.
- 25 La Figura 52 representa un gel de diagnóstico que muestra la eficiencia de la PCR.
- La Figura 53 muestra un gel que muestra una preparación de prueba de Pippin con la muestra de control número 8.
- La Figura 54 muestra muestras ejecutadas en una preparación Pippin.
- 30 La Figura 55 representa datos de imágenes que muestran la capacidad de clonación de las perlas. Los tres ensayos demostraron la capacidad de clonación, por lo que la emulsión de una biblioteca genómica humana funcionó bien, incluso cuando se utiliza un hidrociclador (que es una gran solución de ahorro de tiempo y técnicamente fácil). Fila superior: luz blanca 20x, zoom 200%. Fila central: 20x, Cy5, Cy3, Alineación de colores FITC. Fila inferior: 20x, zoom 200%, Cy5, Cy3, alineación de colores FITC.
- La Figura 56 muestra los datos de imágenes de una única lisis celular y PCR con perlas de captura VDJ.
- La Figura 57 representa datos de imágenes de PCR con perlas de captura VDJ.
- 35 La Figura 58 representa un gel que muestra las pruebas de cebador individual. Carriles de izquierda a derecha: escalera de 1.100 pb; 2, lleno; 3, lleno-2; 4, J1; 5, J2; 6, J3; 7, V30; 8, V29; 9, V28; 10, V27.
- La Figura 59 representa geles que muestran pruebas de cebador individual.
- La Figura 60 muestra un gel que muestra la prueba combinada de solapamiento de cebador Illumina.
- La figura 61 representa un gel que muestra un conjunto.
- 40 La Figura 62 representa un gel que muestra ADNc y ADN de Coriel.
- La figura 63 representa un gel de SDS PAGE. 200 voltios, corrido durante 45 minutos.

La figura 64 representa un gel de SDS PAGE. 200 voltios, corrido durante 30 minutos.

La figura 65 muestra un gel de SDS PAGE. Corrido durante 35 minutos, tinción azul.

La figura 66 muestra ensayos de neutralización.

La figura 67 representa gráficamente los resultados de neutralización.

5 La Figura 68 muestra un gel de diagnóstico utilizado para confirmar la eficacia de la PCR.

La Figura 69 muestra un gel de diagnóstico utilizado para confirmar la eficacia de las perlas de bloqueo y AMPure.

La Figura 70 muestra un gel de diagnóstico utilizado para confirmar la eficiencia de AMPure.

La Figura 71 representa un gel de diagnóstico utilizado para confirmar la eficiencia de reacción combinada.

10 La Figura 72 muestra un gel de diagnóstico utilizado para confirmar la eficiencia de la PCR utilizando cebadores que amplifican el producto final.

La Figura 73 muestra los geles de extracción pre-Pippin (panel izquierdo) y la extracción post-Pippin (panel derecho).

La Figura 74 muestra un gel de diagnóstico utilizado para confirmar la eficacia de la PCR.

La Figura 75 muestra un gel de diagnóstico utilizado para confirmar la eficiencia de la ligación.

La figura 76 muestra geles de prueba de Pippin.

15 Descripción detallada

El alcance de la invención se define en las reivindicaciones adjuntas. En ciertas realizaciones de ejemplo, se proporciona un conjunto de tecnologías y herramientas analíticas para inspeccionar de manera eficiente la información registrada en el sistema inmune, permitiendo así diagnósticos clínicos y/o terapéuticos personalizados. Ciertos aspectos de la invención utilizan la secuenciación de la próxima generación para rastrear grandes cantidades de secuencias codificadoras de anticuerpos de millones de células individuales en una única reacción, en contraste con una célula por ensayos de pozos que se realizan típicamente en la técnica. Ciertos aspectos de la invención se dirigen a la secuenciación e identificación de porciones de secuencias codificadoras de anticuerpos. Otros aspectos de la invención se dirigen a métodos para expresar bibliotecas de anticuerpos complejos, por ejemplo, para uso con pantallas funcionales y/o análisis de evolución contra una biblioteca de antígenos o proteínas. En ciertos aspectos, los métodos y composiciones descritos en el presente documento se pueden usar para la identificación de biomarcadores para el desarrollo de ensayos de diagnóstico o detección frente a enfermedades infecciosas, trastornos celulares proliferativos tales como cáncer, alergias y/o trastornos autoinmunes. En otros aspectos, los métodos y composiciones descritos en el presente documento pueden usarse para el descubrimiento y desarrollo de agentes terapéuticos, utilizando, por ejemplo, monoclonales, monoclonales de múltiples agrupaciones, policlonales, fragmentos, secuencias de anticuerpos completas y/o parciales.

Ciertas realizaciones de ejemplo adicionales utilizan la caracterización de la diversidad y dinámica del repertorio inmune de acuerdo con los métodos descritos en este documento para identificar dianas de vacuna y anticuerpos neutralizantes. Dicha caracterización a nivel individual es útil en métodos de diagnóstico y/o pronóstico de autoinmunidad y/o enfermedades alérgicas. Los métodos de caracterización descritos en el presente documento son además útiles para crear bases de datos de afinidades antígeno/anticuerpo que son útiles para predecir la especificidad del anticuerpo a partir de la secuencia y para el diagnóstico de enfermedades o afecciones caracterizadas por ciertas afinidades antígeno/anticuerpo. Los métodos de caracterización descritos en el presente documento son aún más útiles en los métodos para determinar el rechazo de trasplantes y la histoincompatibilidad basados en el análisis de los componentes del repertorio inmune de un individuo en un solo punto en el tiempo o en varios puntos en el tiempo. Los anticuerpos caracterizados de acuerdo con los métodos descritos en el presente documento son útiles en métodos para extraer e identificar ciertos efectores inmunes y antígenos de una muestra o sujeto.

En ciertas realizaciones de ejemplo, se proporcionan métodos para determinar el estado genético de una pluralidad de células inmunes individuales (por ejemplo, linfocitos) en un momento dado (por ejemplo, una "instantánea" de una colección de células inmunes individuales). En ciertos aspectos, la identidad de las cadenas pesadas y/o livianas (es decir, una secuencia de ácidos nucleicos del receptor inmune) para cada uno de una pluralidad de receptores se determinará en uno o más puntos dados en el tiempo. Una colección de identidades de células inmunitarias también se denomina aquí como un "repertorio inmune" o un "VDJ-oma".

Las realizaciones de los métodos de caracterización descritos en este documento permiten la identificación, creación y/o uso de anticuerpos de respuesta rápida en métodos de diagnóstico, terapéuticos o de investigación. Dichos anticuerpos de respuesta rápida se expresan rápidamente, normalmente en cuestión de horas, como resultado de un desafío inmunitario. De acuerdo con los aspectos de la presente invención, los anticuerpos de respuesta rápida se

pueden caracterizar por su amplitud de afinidad contra objetivos o antígenos y su capacidad para estimular otras funciones inmunitarias del huésped requeridas. Los anticuerpos de respuesta rápida identificados por los métodos descritos en este documento se pueden usar como un principio agente terapéutico para ayudar rápidamente al sistema inmunológico a responder a una enfermedad u otra condición mientras el sistema inmunitario desarrolla anticuerpos adicionales.

Las realizaciones de la presente invención incluyen además el uso de secuencias derivadas de IGHV3-23 humanas o secuencias derivadas de IGHV1-69 humanas identificadas usando secuenciación de alto rendimiento en anticuerpos funcionalmente usados. Para los fines de la presente invención, las secuencias derivadas incluyen la secuencia misma, así como cualquier fragmento o mutante de la misma.

Los factores principales en el fracaso del desarrollo de anticuerpos monoclonales han sido la falta de transición del modelo animal al humano, así como la capacidad limitada de seleccionar una cantidad suficiente de candidatos de anticuerpos para permitir el descubrimiento eficaz de productos terapéuticos. Los métodos y composiciones descritos en el presente documento resuelven estos problemas al permitir la selección del repertorio inmune o la respuesta inmune directamente de uno o más humanos, por lo que producen candidatos de anticuerpos que son completamente humanos (es decir, no son anticuerpos humanizados), evitando así los riesgos de inmunogenicidad. En ciertos aspectos, los métodos y composiciones descritos en el presente documento no están limitados en su rendimiento, lo que permite la detección simultánea de una cantidad ilimitada de secuencias codificadoras de anticuerpos, lo que no sería posible utilizando métodos inmunológicos clásicos.

Como los métodos y las composiciones descritas en el presente documento permiten la selección de una gran cantidad de secuencias por muestra, se puede realizar una investigación eficiente de la reacción inmune en un individuo. Por ejemplo, la presencia de una o más reacciones inmunes naturales o secuencias de anticuerpos específicas creadas por el paciente se pueden usar como una pantalla de diagnóstico en un sujeto, por ejemplo, un paciente con una infección actual o previa. Los métodos y composiciones descritos en el presente documento permiten la determinación instantánea de secuencias de anticuerpos en un individuo en cualquier momento (por ejemplo, porque el ARN y/o el ADN que codifican la proteína del anticuerpo están secuenciados), en lugar de tener que conducir complejos y no complejos y determinación de secuencias de proteínas no multiplexables. En la Figura 5 se muestra una descripción general de ciertos métodos descritos en el presente documento. En la Figura 25 se muestra una aplicación de los métodos de acuerdo con ciertos aspectos de la divulgación.

De acuerdo con ciertos aspectos, los métodos de caracterización de componentes de una respuesta inmune en un individuo que usa métodos de secuenciación de alto rendimiento descritos en el presente documento permiten la selección de secuencias de anticuerpos específicas, incluidos los anticuerpos de isotipo IgA, útiles para diagnosticar una enfermedad o afección, o tratar terapéuticamente una enfermedad o condición. Los métodos descritos en el presente documento permiten métodos para comparar componentes de la reacción inmunitaria en un individuo en diferentes puntos en el tiempo (información de series de tiempo) que es más útil en la asignación de funciones a los anticuerpos identificados. Dichos métodos incluyen la identificación de anticuerpos útiles en métodos diagnósticos o terapéuticos, e incluyen la identificación de anticuerpos de dominio único (nanocuerpos) derivados de humanos que son útiles como reactivos de afinidad, por ejemplo, como biomarcadores de diagnóstico o terapéuticos. Los métodos descritos en el presente documento también permiten la identificación de secuencias derivadas de seres humanos utilizando una secuenciación de alto rendimiento que luego puede formatearse en diferentes moléculas de afinidad, como diacuerpos, nanocuerpos, dobles scFv y otros componentes definidos a continuación como un anticuerpo.

De acuerdo con aspectos alternativos, una vez que los métodos descritos en este documento se usan para identificar los componentes en un individuo que resulta de una respuesta inmune (referida aquí como un repertorio inmune), los componentes pueden clonarse y expresarse para crear bibliotecas de anticuerpos usando cualquiera sistema de expresión sin células o clonación en vectores de expresión para la expresión en la célula. Los anticuerpos pueden luego probarse o cribarse para determinar su capacidad para unirse a los antígenos, su afinidad y/o capacidad de neutralización. Además, una vez que los métodos descritos en el presente documento se usan para identificar los componentes en un individuo que resulta de una respuesta inmune, los componentes pueden clonarse seguidos de la síntesis de genes de anticuerpos seleccionados en función de la información funcional y luego expresarse para crear bibliotecas de anticuerpos usando cualquiera sistema de expresión libre o clonación en vectores de expresión para la expresión en la célula. Los anticuerpos pueden luego probarse o cribarse para determinar su capacidad para unirse a los antígenos, su afinidad y/o capacidad de neutralización. De acuerdo con ciertos aspectos, los anticuerpos resultantes de la respuesta inmune que se identifican mediante los métodos descritos en el presente documento se pueden sintetizar utilizando un preparación de síntesis paralelo a partir de oligonucleótidos o la química de la síntesis de oligonucleótidos estándar derivada de fragmentos de genes, a partir de la síntesis de impresión de chips, o cualquier otro método conocido por los expertos en la materia.

De acuerdo con un aspecto adicional, los anticuerpos identificados por los métodos de alto rendimiento descritos en el presente documento se clonan directamente en tecnologías de visualización de superficie que incluyen levadura, fago, bacteriano, mamífero, ribosomal, presentación de ARNm y similares y luego se analizan para determinar la especificidad o afinidad funcional o capacidad de neutralización. Un aspecto adicional de esta realización incluye los anticuerpos que se clonan directamente en tecnologías de visualización de superficie y luego evolucionan mediante enfoques de evolución dirigida conocidos por los expertos en la técnica y, opcionalmente, caracterizan los enfoques

de evolución dirigida en varios puntos en el tiempo, en cualquier paso o cualquier ciclo que utilice secuenciación de alto rendimiento para identificar el estado de enriquecimiento, nivel de diversidad, etc. Tales métodos de evolución dirigida pueden realizarse con un antígeno único, antígenos múltiples o una gran biblioteca de antígenos en una sola reacción.

5 Como se usa en el presente documento, los términos "sujeto", "individuo" y "huésped" pretenden incluir organismos vivos tales como mamíferos. Ejemplos de sujetos y hospedadores incluyen, pero no se limitan a, caballos, vacas, camellos, ovejas, cerdos, cabras, perros, gatos, conejos, cobayas, ratas, ratones (por ejemplo, ratones humanizados), jerbos, primates no humanos (por ejemplo, macacos), humanos y similares, no mamíferos, incluidos, por ejemplo, vertebrados no mamíferos, como aves (por ejemplo, pollos o patos) peces (por ejemplo, tiburones) o ranas (por ejemplo, Xenopus), e invertebrados no mamíferos, así como las especies transgénicas de los mismos.

10 En ciertos aspectos, un sujeto se refiere a un solo organismo (por ejemplo, humano). En ciertos aspectos, o un grupo de individuos que componen una pequeña cohorte que tiene un factor inmune común para estudiar y/o enfermedad, y/o una cohorte de individuos sin la enfermedad (por ejemplo, control negativo/normal) se proporcionan. Un sujeto del que se obtienen muestras puede ser infligido con una enfermedad y/o trastorno (por ejemplo, una o más alergias, infecciones, cánceres o trastornos autoinmunes o similares) o ser inyectado con un agente (por ejemplo, uno o más de vacuna, un antígeno de interés, un fármaco, una molécula pequeña, un agente biológico o similar para estimular un desafío inmunológico, y compararse con un sujeto de control negativo que no está afectado por la enfermedad y/o no está en contacto con el agente.

15 En ciertos aspectos, una o más muestras biológicas se aíslan de uno o más sujetos. Como se usa en este documento, una "muestra biológica" puede ser una célula única o muchas células. Una muestra biológica puede comprender un solo tipo de célula o una combinación de dos o más tipos de células. Una muestra biológica incluye además una colección de células que realizan una función similar, como las que se encuentran, por ejemplo, en un tejido. Como se usa en este documento, un tejido incluye, pero no se limita a, tejido epitelial (por ejemplo, piel, el revestimiento de glándulas, intestino, piel y órganos como el hígado, pulmón, riñón), endotelio (por ejemplo, el revestimiento de sangre y vasos linfáticos), mesotelio (por ejemplo, el revestimiento de los espacios pleural, peritoneal y pericárdico), mesénquima (por ejemplo, células que llenan los espacios entre los órganos, incluidas las células de grasa, músculo, hueso, cartílago y tendón), células sanguíneas (por ejemplo, células rojas) y glóbulos blancos), neuronas, células germinales (por ejemplo, espermatozoides, ovocitos), células de líquido amniótico, placenta, células madre y similares. Una muestra de tejido incluye muestras microscópicas, así como muestras macroscópicas. En ciertos aspectos, una muestra biológica es sangre periférica. En otros aspectos, una muestra biológica es un fluido como la saliva, el fluido sinovial o similares. En otros aspectos, una muestra biológica es de uno o más cultivos celulares, secciones de tejido y/o biopsias.

20 En ciertas realizaciones de ejemplo, una fracción de célula inmunitaria (por ejemplo, un linfocito) se aísla de una muestra biológica utilizando cualquier técnica conocida por un experto en la técnica. En ciertos aspectos, una fracción de linfocitos se aísla mediante centrifugación Ficoll. En otros aspectos, una fracción de linfocitos se inmoviliza sobre un sustrato o un soporte (por ejemplo, un sustrato que se une a una o más de las células B y/o células T). El soporte puede ser simples rejillas cuadradas, cuadrículas de ajedrez (por ejemplo, compensadas), matrices hexagonales y similares. Los soportes adecuados incluyen, entre otros, diapositivas, perlas, astillas, partículas, hebras, geles, láminas, tubos, esferas, recipientes, capilares, almohadillas, rodajas, películas, placas de cultivo, placas (por ejemplo, 96 pozos, 48 pozos, 24 pozos, 12 pozos, ocho pozos, seis pozos, cuatro pozos, pozo individual y similares), superficies celulares (por ejemplo, células de *S. aureus*) y similares. En diversas realizaciones, un soporte sólido puede ser biológico, no biológico, orgánico, inorgánico o cualquier combinación de los mismos.

25 En ciertos ejemplos de realización, se proporcionan perlas y matrices basadas en perlas. Como se usa en el presente documento, el término "perla" se refiere a una partícula discreta que puede ser esférica (por ejemplo, microesferas) o que tiene una forma irregular. Las perlas pueden ser tan pequeñas como de aproximadamente 0.1 μm de diámetro o tan grandes como de aproximadamente varios milímetros de diámetro. Las perlas pueden comprender una variedad de materiales que incluyen, entre otros, materiales paramagnéticos, cerámica, plástico, vidrio, poliestireno, metilostireno, polímeros acrílicos, titanio, látex, sefarosa, celulosa, nylon y similares.

30 De acuerdo con ciertos ejemplos, un soporte (por ejemplo, una perla) puede tener grupos funcionales unidos a su superficie que pueden usarse para unir uno o más reactivos descritos en el presente documento a la perla. Se pueden unir uno o más reactivos a un soporte (por ejemplo, una perla) por hibridación, unión covalente, unión magnética, unión por afinidad y similares. Por ejemplo, un soporte (por ejemplo, una perla) se puede recubrir con un anticuerpo secundario para usar con un anticuerpo primario. En otro ejemplo, un soporte (por ejemplo, una perla) puede recubrirse con grupos reactivos de glicidil éter (epoxi) y/o grupos reactivos de p-toluensulfonilo (tosilo) para usar con un anticuerpo primario. Las perlas recubiertas con una variedad de sustratos están disponibles comercialmente (Dynabeads, Invitrogen). Los soportes (por ejemplo, perlas) también pueden funcionalizarse usando, por ejemplo, químicas en fase sólida conocidas en la técnica (véase, por ejemplo, Patente de los Estados Unidos No. 5,919,523).

35 Como se usa en el presente documento, el término "unir" se refiere tanto a interacciones covalentes como a interacciones no covalentes. Una interacción covalente es un enlace químico entre dos átomos o radicales formados por el intercambio de un par de electrones (es decir, un enlace simple), dos pares de electrones (es decir, un doble

enlace) o tres pares de electrones (es decir, un triple enlace). Las interacciones covalentes también son conocidas en la técnica como interacciones de pares de electrones o enlaces de pares de electrones. Las interacciones no covalentes incluyen, pero no se limitan a, interacciones de van der Waals, enlaces de hidrógeno, enlaces químicos débiles (es decir, a través de fuerzas no covalentes de corto alcance), interacciones hidrófobas, enlaces iónicos y similares. Una revisión de las interacciones no covalentes se puede encontrar en Alberts et al., en *Molecular Biology of the Cell*, 3d edition, Garland Publishing, 1994.

En ciertas realizaciones de ejemplo, las secuencias de ácidos nucleicos (por ejemplo, ADN y/o ARN) se extraen de células inmunes (por ejemplo, linfocitos, por ejemplo, células B y/o células T). En ciertos aspectos, las secuencias de ácidos nucleicos (por ejemplo, ADN y/o ARN) se extraen de una o más células inmunes de fracciones enriquecidas (por ejemplo, linfocitos, por ejemplo, células B y/o células T). En otros aspectos, las secuencias de ácidos nucleicos (por ejemplo, ADN y/o ARN) se extraen de muestras que no han sido enriquecidas para células inmunes. En ciertos aspectos, los subconjuntos específicos de poblaciones de células se enriquecen preliminarmente, por ejemplo, utilizando FACS o perlas paramagnéticas, lo que permite el enriquecimiento de una célula de memoria B frente a la fracción de células plasmáticas, que puede ser útil para aumentar la calidad de los datos para su uso posterior, por ejemplo, para uso en pronóstico, diagnóstico y similares. En ciertos aspectos, se extraen al menos 25, 50, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1,000, 2,000, 3,000, 4,000, 5,000, 6,000, 7,000, 8,000, 9,000, 10,000, 15,000, 20,000, 25,000, 30,000, 35,000, 40,000, 45,000, 50,000, 55,000, 60,000, 65,000, 70,000, 75,000, 80,000, 85,000, 90,000, 95,000, 100,000, 150,000, 200,000, 250,000, 300,000, 450,000, 500,000, 550,000, 600,000, 650,000, 700,000, 750,000, 800,000, 850,000, 900,000, 950,000, 1,000,000, 1,500,000, 2,000,000, 2,500,000, 3,000,000, 3,500,000, 4,000,000, 4,500,000, 5,000,000, 5,500,000, 6,000,000, 6,500,000, 7,000,000, 7,500,000, 8,000,000, 8,500,000, 9,000,000, 9,500,000, 10,000,000, 20,000,000, 30,000,000, 40,000,000, 50,000,000, 60,000,000, 70,000,000, 80,000,000, 90,000,000, 100,000,000, 200,000,000, 300,000,000, 400,000,000, 500,000,000, 600,000,000, 700,000,000, 800,000,000, 900,000,000, 1,000,000,000, 2,000,000,000, 3,000,000,000, 4,000,000,000, 5,000,000,000, 6,000,000,000, 7,000,000,000, 8,000,000,000, 9,000,000,000, 10,000,000,000, 15,000,000,000, 20,000,000,000, 30,000,000,000, 40,000,000,000, 50,000,000,000, 60,000,000,000, 70,000,000,000, 80,000,000,000, 90,000,000,000, 100,000,000,000, 200,000,000,000, 300,000,000,000, 400,000,000,000, 500,000,000,000, 600,000,000,000, 700,000,000,000, 800,000,000,000, 900,000,000,000, 1,000,000,000,000, 2,000,000,000,000, 3,000,000,000,000, 4,000,000,000,000, 5,000,000,000,000, 6,000,000,000,000, 7,000,000,000,000, 8,000,000,000,000, 9,000,000,000,000, 10,000,000,000,000 o más secuencias de ácidos nucleicos diferentes (por ejemplo, ADN (por ejemplo, genómico) y/o ARN (por ejemplo, ARNm)) o cualquier valor entre y/o rango(s) definido por los números anteriores o intervalos entre o superposición de las células inmunes.

En ciertas realizaciones de ejemplo, se utilizan cebadores no específicos, cebadores degenerados o cebadores específicos (por ejemplo, cebadores que se hibridan con las regiones J y/o C para las cadenas pesadas y/o livianas de las células B o T) para amplificar secuencias de ácido nucleico (por ejemplo, ADN y/o ARN). En ciertos aspectos, el ADNc (si se usa ARNm) o el ADN (si se usa ADN genómico) se amplifica mediante PCR utilizando conjuntos de cebadores que hibridan con el gen codificador de anticuerpos de las cadenas pesada y liviana de las células B y/o células T. En ciertos aspectos, un conjunto de cebadores se hibrida con las regiones V y otro conjunto de cebadores se hibrida con las regiones C. En otros aspectos, ciertos conjuntos de cebadores se hibridan con otras ubicaciones en la región V o en sentido ascendente en la región líder (lo que es favorable para la muestra de ARN, lo que permite la amplificación del ADNc convertido en ARN y no del ADN), mientras que otros conjuntos de cebadores se hibridan en sentido descendente de la región C, y/o a la región J.

En ciertos aspectos, un segmento que abarca la región CDR3 se amplifica utilizando cebadores que se hibridan a cada lado de la región CDR3. Cuanto más separados estén los cebadores en los segmentos VDJ, más secuencia de anticuerpos se recuperará y secuenciará, agregando así información adicional sobre el estado inmune (por ejemplo, información sobre las regiones CDR1, CDR2 y CDR3 y/o hipermutación a través de los segmentos de codificación del anticuerpo). Debido a que hay muchas regiones V, D, J y C, se pueden diseñar cebadores degenerados que hibridan con muchos segmentos, o se puede usarse una sola secuencia de cebador para cada uno de estos segmentos con el fin de amplificar representativamente el repertorio del sujeto (por ejemplo, vía multiplex PCR). En otros aspectos, la amplificación de un conjunto único muy específico de combinación de VDJ se realiza utilizando un conjunto de cebadores. En ciertos aspectos, los ensayos de extracción pueden utilizarse para enriquecer las secuencias de ácidos nucleicos de las células inmunitarias (por ejemplo, ARN y/o ADN). En ciertos aspectos, se realiza un ensayo desplegable que usa una sonda de ARN o de ADN complementaria a alguna parte de los segmentos de interés de VDJ. En ciertos aspectos, los marcadores se usan para enriquecer las secuencias de ácidos nucleicos de células inmunitarias (por ejemplo, ARN y/o ADN) (por ejemplo, fluoróforos para la clasificación de FACS, biotina con estreptavidina, perlas recubiertas de forma covalente y similares). También se pueden utilizar estrategias de enriquecimiento dirigidas, como el uso de sondas de inversión molecular o el uso de hibridación en matriz.

Como se usa en el presente documento, el término "hibridación" se refiere al proceso en donde dos polinucleótidos de cadena simple se unen de manera no covalente para formar un polinucleótido de cadena doble estable. El término "hibridación" también puede referirse a la hibridación de cadena triple. El polinucleótido de doble cadena resultante (generalmente) es un "híbrido" o "dúplex". Las "condiciones de hibridación" normalmente incluirán concentraciones de sal de menos de aproximadamente 1 M, más generalmente de menos de aproximadamente 500 mM e incluso más generalmente de menos de aproximadamente 200 mM. Las temperaturas de hibridación pueden ser tan bajas como 5°C, pero típicamente son mayores que 22°C, más típicamente mayores que aproximadamente 30°C, y con frecuencia

en exceso de aproximadamente 37°C. Las hibridaciones se realizan generalmente en condiciones de restricción, es decir, condiciones en las que una sonda hibridará con su subsecuencia objetivo. Las condiciones de restricción dependen de la secuencia y son diferentes en diferentes circunstancias. Los fragmentos más largos pueden requerir temperaturas de hibridación más altas para una hibridación específica. Como otros factores pueden afectar la rigurosidad de la hibridación, incluida la composición de la base y la longitud de las cadenas complementarias, la presencia de disolventes orgánicos y el grado de emparejamiento incorrecto de la base, la combinación de parámetros es más importante que la medida absoluta de uno solo. En general, las condiciones de restricción se seleccionan para que sean aproximadamente 5°C más bajas que la T_m para la secuencia específica a una fuerza iónica y pH definidos. Las condiciones de restricción de ejemplo incluyen una concentración de sal de al menos 0.01 M a no más de 1 M de concentración de ion Na (u otras sales) a un pH de 7.0 a 8.3 y una temperatura de al menos 25°C. Por ejemplo, las condiciones de 5XSSPE (NaCl 750 mM, fosfato de Na 50 mM, EDTA 5 mM, pH 7.4) y una temperatura de 25-30°C son adecuadas para hibridaciones de sondas específicas de alelos. Para condiciones de restricción, véase, por ejemplo, Sambrook, Fritsche and Maniatis, *Molecular Cloning A Laboratory Manual*, 2nd Ed. Cold Spring Harbor Press (1989) y Anderson *Nucleic Acid Hybridization*, 1st edition, BIOS Scientific Publishers Limited (1999). "Hibridación específica a" o "específicamente hibridación a" o expresiones similares se refieren a la unión, dúplex o hibridación de una molécula sustancialmente a o solo a una secuencia de nucleótidos particular o secuencias en condiciones de restricción cuando esa secuencia está presente en una mezcla compleja (por ejemplo, total celular) ADN o ARN.

En ciertas realizaciones de ejemplo, se proporcionan métodos para amplificar secuencias de ácidos nucleicos. Los métodos de ejemplo para amplificar ácidos nucleicos incluyen la reacción en cadena de la polimerasa (PCR) (véase, por ejemplo, Mullis et al. (1986) *Cold Spring Harb. Symp. Quant. Biol.* 51 Pt 1: 263 and Cleary et al. (2004) *Nature Methods* 1:241 y Patentes de los Estados Unidos Nos. 4,683,195 y 4,683,202), PCR de anclaje, RACE PCR, reacción en cadena de la ligación (LCR) (véase, por ejemplo, Landegran et al. (1988) *Science* 241: 1077-1080; y Nakazawa et al. (1994) *Proc. Natl. Acad. Sci. U.S.A.* 91: 360-364), replicación de secuencia autosostenida (Guatelli et al. (1990) *Proc. Natl. Acad. Sci. U.S.A.* 87: 1874), sistema de amplificación transcripcional (Kwoh et al. (1989) *Proc. Natl. Acad. Sci. U.S.A.* 86: 1173), Q-Beta Replicasa (Lizardi et al. (1988) *BioTechnology* 6: 1197), PCR recursiva (Jaffe et al. (2000) *J. Biol. Chem.* 275: 2619 and Williams et al. (2002) *J. Biol. Chem.* 277: 7790), los métodos de amplificación descritos en las Patentes de los Estados Unidos Nos. 6,391,544, 6,365,375, 6,294,323, 6,261,797, 6,124,090 y 5,612,199, amplificación isotérmica (por ejemplo, amplificador de círculo rodante (RCA), amplificación de círculo rodante hiperramificado (HRCA), amplificación por desplazamiento de cadena (SDA), amplificación dependiente de helicasa (HDA), PWGA) o cualquier otro método de amplificación de ácido nucleico utilizando técnicas bien conocidas por los expertos en la técnica.

"Reacción en cadena de la polimerasa" o "PCR", se refiere a una reacción para la amplificación in vitro de secuencias de ADN específicas mediante la extensión simultánea del cebador de cadenas complementarias de ADN. En otras palabras, la PCR es una reacción para realizar múltiples copias o réplicas de un ácido nucleico objetivo flanqueado por sitios de unión del cebador, dicha reacción comprende una o más repeticiones de los siguientes pasos: (i) desnaturar el ácido nucleico objetivo, (ii) fusión de cebadores a los sitios de unión del cebador, y (iii) extensión de los cebadores por una polimerasa de ácido nucleico en presencia de nucleósidos trifosfatos. Normalmente, la reacción se realiza en ciclos a diferentes temperaturas optimizadas para cada paso en un instrumento termociclador. Las temperaturas particulares, las duraciones en cada paso y las tasas de cambio entre los pasos dependen de muchos factores bien conocidos por los expertos en la técnica, por ejemplo, ejemplificados por las referencias: McPherson et al., Editores, *PCR: A Practical Approach and PCR2: A Practical Approach* (IRL Press, Oxford, 1991 y 1995, respectivamente). Por ejemplo, en una PCR convencional que utiliza Taq ADN polimerasa, un ácido nucleico diana bicatenario se puede desnaturar a una temperatura superior a 90°C, los cebadores se fusionan a una temperatura en el rango de 50-75°C y los cebadores se extienden a una temperatura en el rango de 72-78°C.

El término "PCR" abarca formas derivadas de la reacción, que incluyen, pero no se limitan a, RT-PCR, PCR en tiempo real, PCR anidada, PCR cuantitativa, PCR multiplexada, PCR de preparación y similares. Los volúmenes de reacción varían desde unos pocos cientos de nanolitros, por ejemplo, 200 nL, hasta unos pocos cientos de microlitros, por ejemplo, 200 microlitros. "PCR de transcripción reversa" o "RT-PCR" significa una PCR que está precedida por una reacción de transcripción reversa que convierte un ARN objetivo en un ADN monocatenario complementario, que luego se amplifica, por ejemplo, Tecott et al., Patente de los Estados Unidos No. 5,168,038. "PCR en tiempo real" significa una PCR para la cual se monitoriza la cantidad de producto de reacción, es decir, un amplicón, a medida que avanza la reacción. Hay muchas formas de PCR en tiempo real que difieren principalmente en las químicas de detección utilizadas para monitorizar el producto de reacción, por ejemplo, Gelfand et al., Patente de los Estados Unidos No. 5,210,015 ("Taqman"); Wittwer et al., Patentes de los Estados Unidos Nos. 6,174,670 y 6,569,627 (colorantes intercalantes); Tyagi et al., Patente de los Estados Unidos No. 5,925,517 (balizas moleculares). Las químicas de detección para la PCR en tiempo real se revisan en Mackay et al., *Nucleic Acids Research*, 30: 1292-1305 (2002). "PCR anidada" significa una PCR de dos etapas en donde el amplicón de una primera PCR se convierte en la muestra para una segunda PCR utilizando un nuevo conjunto de cebadores, al menos uno de los cuales se une a una ubicación interior del primer amplicón. Como se usa en este documento, "cebadores iniciales" en referencia a una reacción de amplificación anidada significa los cebadores utilizados para generar un primer amplicón, y "cebadores secundarios" significa el uno o más cebadores utilizados para generar un segundo amplicón, o anidado. "PCR multiplexada" significa una PCR en la que múltiples secuencias diana (o una única secuencia diana y una o más secuencias de referencia) se llevan a cabo simultáneamente en la misma mezcla de reacción, por ejemplo, Bernard et al. (1999) *Anal. Biochem.*,

273: 221-228 (PCR en tiempo real de dos colores). Normalmente, se emplean distintos conjuntos de cebadores para cada secuencia que se amplifica. "PCR cuantitativa" significa una PCR diseñada para medir la abundancia de una o más secuencias diana específicas en una muestra o muestra. Las técnicas para la PCR cuantitativa son bien conocidas por los expertos en la técnica, como se ejemplifica en las siguientes referencias: Freeman et al., *Biotechniques*, 26: 112-126 (1999); Becker-Andre et al., *Nucleic Acids Research*, 17: 9437-9447 (1989); Zimmerman et al., *Biotechniques*, 21: 268-279 (1996); Diviacco et al., *Gene*, 122: 3013-3020 (1992); Becker-Andre et al., *Nucleic Acids Research*, 17: 9437-9446 (1989); y similares.

En ciertas realizaciones de ejemplo, se proporcionan métodos para determinar las identidades de secuencia de secuencias de ácidos nucleicos. La determinación de la secuencia de una secuencia de ácidos nucleicos de interés (por ejemplo, secuencias de ácidos nucleicos de células inmunitarias) se puede realizar utilizando una variedad de métodos de secuenciación conocidos en la técnica que incluyen, pero no se limitan a, secuenciación por hibridación (SBH), secuenciación por ligación (SBL), secuenciación cuantitativa de adición de nucleótidos fluorescentes incrementales (QIFNAS), ligación y escisión escalonadas, transferencia de energía de resonancia de fluorescencia (FRET), balizas moleculares, digestión de la sonda reportera TaqMan, pirosecuenciación, secuenciación in situ fluorescente (FISSEQ), perlas FISSEQ (Patente de los Estados Unidos No. 7,425,431), secuenciación por bamboleo (PCT/US05/27695), secuenciación multiplex (número de serie de los Estados Unidos 12/027,039, presentada el 6 de febrero de 2008; Porreca et al (2007) *Nat. Methods* 4: 931), colonia polimerizada (POLONY) secuenciación (Patentes de los Estados Unidos Nos. 6,432,360, 6,485,944 y 6,511,803, y PCT/US05/06425); secuenciación de círculo rodante de nanogrid (ROLONY) (número de serie 12/120,541 de los Estados Unidos, presentada el 14 de mayo de 2008), ensayos de ligamiento de oligo específicos de alelo (por ejemplo, ensayo de ligamiento de oligo (OLA), OLA de molécula de plantilla única que utiliza una sonda lineal ligada y una lectura de amplificación de círculo rodante (RCA), sondas de candado ligadas y/o molécula de plantilla única OLA utilizando una sonda de candado circular ligada y una lectura de amplificación de círculo rodante (RCA) y similares. También se pueden utilizar métodos de secuenciación de alto rendimiento, por ejemplo, en secuenciación de matriz cíclica que utiliza plataformas como las plataformas Roche 454, Illumina Solexa, ABI-SOLiD, ION Torrents, Genómica completa, Pacific Bioscience, Helicos, Polonator (Sitio web mundial: Polonator.org), y similares. Los métodos de secuenciación de alto rendimiento se describen en el número de serie de los Estados Unidos 61/162,913, presentada el 24 de marzo de 2009. En la técnica se conocen diversas tecnologías de secuenciación basadas en la luz (Landegren et al. (1998) *Genome Res.* 8: 769-76; Kwok (2000) *Pharmacogenomics* 1:95-100; y Shi (2001) *Clin. Chem.* 47: 164-172).

En ciertas realizaciones de ejemplo, los segmentos VDJ enriquecidos se transforman para que puedan ser compatibles con las tecnologías de secuenciación de alto rendimiento de próxima generación. En ciertos aspectos, los adaptadores de secuenciación compatibles se ligan o se agregan mediante PCR y cebadores con saliente en los segmentos VDJ. En otros aspectos, el adaptador tiene un código de barras para que muchas muestras se puedan agrupar y secuenciar en un solo instrumento. Según un aspecto, las muestras que pertenecen al mismo estudio de investigación se procesan lo más posible de la misma manera al mismo tiempo para lograr un alto grado de correlación y maximizar la calidad de los datos obtenidos. Idealmente, la longitud de lectura debería ser suficiente para permitir la identificación de cada segmento V, D y J, originado de cada molécula única. El uso del código de barras también se puede utilizar para lograr la reasignación de segmentos cortos a su molécula inicial original (PMID: 20081835).

En ciertas realizaciones de ejemplo, los datos de secuenciación se alinean contra los segmentos V, D y J conocidos o esperados utilizando una base de datos personal, bases de datos NCBI, bases de datos IMGT o similares para ayudar en la identificación de los segmentos V, D y J. De esta manera, los segmentos V y J pueden identificarse, lo que permite la extracción de la región de unión central de la V con D y los segmentos D con J, que componen la región CDR3. Todas las lecturas de secuenciación (por ejemplo, millones), se pueden agrupar de acuerdo con su uso de V y J, y agruparse para su CDR3 único utilizando varios algoritmos de agrupación descritos en este documento. Esto permitirá agrupar segmentos VDJ altamente similares en clones VDJ, que probablemente codifican para el mismo anticuerpo o un anticuerpo muy similar. Luego, cada lectura (o segmentos VDJ únicos secuenciados) y/o cada clon VDJ (o segmentos VDJ altamente similares de un mismo grupo de grupos) pueden analizarse para determinar la relevancia biológica (por ejemplo, para una enfermedad, trastorno y/o respuesta inmune).

La frecuencia de los clones se puede representar para examinar la reproducibilidad entre experimentos y/o para identificar clones de VDJ únicos o relevantes de diferentes expresiones en diferentes muestras o cohortes. En el caso de un estudio de curso de tiempo, la frecuencia de clones puede representarse en una gráfica de curso de tiempo para identificar la dinámica de expresión del anticuerpo expresado en el sujeto (Figura 4A). Esto permite la zonificación de posibles secuencias de anticuerpos que forman parte de la respuesta inmune contra el desafío inmune que representa posibles biomarcadores terapéuticos o relevantes contra el desafío inmune (Figura 4B). En el caso de la investigación de análisis estático para una enfermedad (por ejemplo, varias muestras de una cohorte que tiene una enfermedad contra una cohorte que no tiene la enfermedad), el agrupamiento comparativo de qué secuencia se realizó y con qué frecuencia en una cohorte contra otra puede revelar terapias o biomarcadores candidatos. El estudio adicional de la divergencia de una secuencia de anticuerpos (como la acumulación de hipermutación somática) de lecturas de VDJ o clones también se puede usarse para revelar posibles candidatos de interés, así como otra información biológica que se puede derivar de los datos, como la distribución de isotipos de los anticuerpos (como IgA, IgD, IgE, IgG e IgM), que se pueden usar para centrarse en un tipo específico de anticuerpo o pseudoespectrotipificación, por ejemplo.

En ciertas realizaciones de ejemplo, se proporcionan métodos para determinar y/o analizar un estado inmune de un individuo que incluyen la etapa de obtención de células inmunes de una muestra biológica. Tal como se usa en el presente documento, el término "célula inmune" pretende incluir, pero no se limita a, una célula que es de origen hematopoyético y que desempeña un papel en la respuesta inmune. Las células inmunes incluyen, pero no se limitan a, linfocitos, tales como células B y células T; células asesinas naturales; y células mieloides, tales como monocitos, macrófagos, eosinófilos, mastocitos, basófilos y granulocitos.

El término "célula presentadora de antígeno" incluye células presentadoras de antígenos profesionales (por ejemplo, linfocitos B, monocitos, células dendríticas y células de Langerhans) así como otras células presentadoras de antígeno (por ejemplo, queratinocitos, células endoteliales, astrocitos, fibroblastos y oligodendrocitos).

Como se usa en el presente documento, el término "respuesta inmune" incluye, pero no se limita a, respuestas inmunes mediadas por células T y/o células B que están influenciadas por la modulación de la estimulación de células T. Las respuestas inmunes de ejemplo incluyen respuestas de células B (por ejemplo, producción de anticuerpos), respuestas de células T (por ejemplo, producción de citoquinas y citotoxicidad celular) y activación de células sensibles a citoquinas, por ejemplo, macrófagos. Como se usa en el presente documento, el término "modulación descendente" con referencia a la respuesta inmune incluye una disminución en una o más respuestas inmunes, mientras que el término "modulación ascendente" con referencia a la respuesta inmune incluye un aumento en una o más respuestas inmunes. Se entenderá que la modulación ascendente de un tipo de respuesta inmunitaria puede conducir a una modulación descendente correspondiente en otro tipo de respuesta inmunitaria. Por ejemplo, la modulación ascendente de la producción de ciertas citoquinas (por ejemplo, IL-10) puede conducir a una modulación descendente de las respuestas inmunes celulares.

Como se usa en el presente documento, el término "anticuerpo" se refiere a moléculas de inmunoglobulina y porciones inmunológicamente activas de moléculas de inmunoglobulina, es decir, moléculas que contienen un sitio de unión a antígeno que se une específicamente (inmunorreacciona con) un antígeno. Se entiende que el término "anticuerpo" en su término general completo incluye componentes en la cadena inmune que incluyen un anticuerpo, inmunoglobulina, receptor de células B, cadena pesada de anticuerpo, cadena liviana de anticuerpo, receptor de células T (TCR), cadena alfa de TCR, cadena beta TCR, cadena gamma TCR, cadena delta TCR o cualquier variación o modificación de las cadenas de anticuerpos y además incluye scFV, Fab, Fab2, Fab3, Bis-scFv, minicuerpo, triacuerpo, diacuerpo, tetracuerpo, nanocuerpo y cualquiera y todas las variantes de isotipo de anticuerpos. Incluyendo IgG, IgA, IgD, IgM e IgE. Ejemplos de porciones inmunológicamente activas de moléculas de inmunoglobulina incluyen fragmentos F(ab) y F(ab')₂ que pueden generarse tratando el anticuerpo con una enzima tal como pepsina. En ciertas realizaciones de ejemplo, se proporcionan anticuerpos policlonales y monoclonales que se unen a uno o más antígenos de células inmunitarias. Los términos "anticuerpo monoclonal" y "composición de anticuerpo monoclonal", como se usan en este documento, se refieren a una población de moléculas de anticuerpo que contiene solo una especie de un sitio de unión a antígeno capaz de inmunorreaccionar con un epítipo particular de un antígeno de células inmunitarias. Por lo tanto, una composición de anticuerpo monoclonal típicamente muestra una afinidad de unión única para un antígeno de célula inmune particular con el que inmunorreacciona.

En ciertas realizaciones de ejemplo, se proporcionan ensayos de selección para identificar moduladores del sistema inmune, es decir, compuestos o agentes candidatos o de prueba (por ejemplo, anticuerpos, péptidos, péptidos cíclicos, peptidomiméticos, moléculas pequeñas, moléculas orgánicas pequeñas u otros fármacos). Los compuestos de ensayo descritos en el presente documento pueden obtenerse utilizando cualquiera de los numerosos enfoques en métodos de biblioteca combinatoria conocidos en la técnica, que incluyen: bibliotecas biológicas; bibliotecas de fase sólida o de fase en solución paralelas espacialmente direccionables; métodos de biblioteca sintética que requieren la desconvolución; el método de la biblioteca "una perla, un compuesto"; y métodos de biblioteca sintética que utilizan la selección por cromatografía de afinidad. El enfoque de la biblioteca biológica se limita a las bibliotecas de péptidos, mientras que los otros cuatro enfoques son aplicables a las bibliotecas de compuestos de péptidos, oligómeros no peptídicos o moléculas pequeñas (Lam, K. S. (1997) *Anticancer Drug Des.* 12:145).

En ciertas realizaciones de ejemplo, uno o más agentes o sales farmacéuticamente aceptables de los mismos descritos en el presente documento se proporcionan en un vehículo farmacéuticamente aceptable. Tal como se usa en el presente documento, el lenguaje "vehículo farmacéuticamente aceptable" pretende incluir todos y cada uno de los disolventes, medios de dispersión, recubrimientos, agentes antibacterianos y antifúngicos, agentes isotónicos y de retardo de la absorción, y similares, compatibles con la administración farmacéutica. El uso de tales medios y agentes para sustancias farmacéuticamente activas es bien conocido en la técnica. Excepto en la medida en que cualquier medio o agente convencional sea incompatible con el compuesto activo, se contempla su uso en las composiciones. Los compuestos activos suplementarios también pueden incorporarse en las composiciones. Los vehículos farmacéuticamente aceptables y sus formulaciones son conocidos por los expertos en la técnica y se describen, por ejemplo, en Remington's Pharmaceutical Sciences, (19^a edición), ed. A. Gennaro, 1995, Mack Publishing Company, Easton, PA.

En ciertas realizaciones de ejemplo, las formulaciones farmacéuticas de uno o más agentes descritos en el presente documento o sus sales farmacéuticamente aceptables, se administran mediante inyección intravenosa, inyección intraperitoneal, administración oral o por otras vías parenterales (por ejemplo, administración intradérmica, subcutánea, oral (por ejemplo, inhalación), transdérmica (tópica), transmucosa y rectal), o mediante inyecciones

intratecales e intraventriculares en el SNC, en una mezcla con un vehículo farmacéuticamente aceptable adaptado para la vía de administración.

5 Las soluciones o suspensiones utilizadas para la aplicación parenteral, intradérmica, subcutánea o del sistema nervioso central pueden incluir los siguientes componentes: un diluyente estéril como agua para inyección, solución salina, aceites fijos, polietilenglicoles, glicerina, propilenglicol u otros disolventes sintéticos; agentes antibacterianos tales como alcohol bencílico o metilparabenos; antioxidantes tales como ácido ascórbico o bisulfito de sodio; agentes quelantes tales como ácido etilendiaminotetraacético; reguladores tales como acetatos, citratos o fosfatos y agentes para el ajuste de la tonicidad como el cloruro de sodio o la dextrosa. El pH se puede ajustar con ácidos o bases, como ácido clorhídrico o hidróxido de sodio. La preparación parenteral se puede incluir en ampollas, jeringas desechables o viales de dosis múltiples de vidrio o plástico.

10 Métodos bien conocidos en la técnica para hacer formulaciones se encuentran, por ejemplo, en Remington's Pharmaceutical Sciences (19ª edición), ed. A. Gennaro, 1995, Mack Publishing Company, Easton, PA. Las composiciones destinadas al uso oral pueden prepararse en forma sólida o líquida de acuerdo con cualquier método conocido en la técnica para la fabricación de composiciones farmacéuticas. Las composiciones pueden contener opcionalmente agentes edulcorantes, aromatizantes, colorantes, perfumantes y/o conservantes con el fin de proporcionar una preparación más sabrosa. Las formas de dosificación sólidas para administración oral incluyen cápsulas, tabletas, píldoras, polvos y gránulos. En tales formas sólidas, el compuesto activo se mezcla con al menos un vehículo o excipiente farmacéuticamente aceptable. Estos pueden incluir, por ejemplo, diluyentes inertes, tales como carbonato de calcio, carbonato de sodio, lactosa, sacarosa, almidón, fosfato de calcio, fosfato de sodio o caolín. También se pueden usar agentes aglutinantes, agentes reguladores y/o agentes lubricantes (por ejemplo, estearato de magnesio). Los comprimidos y las píldoras también se pueden preparar con recubrimientos entéricos.

15 Las composiciones farmacéuticas adecuadas para uso inyectable incluyen soluciones acuosas estériles (cuando son solubles en agua) o dispersiones y polvos estériles para la preparación extemporánea de soluciones o dispersiones estériles inyectables. Para la administración intravenosa, los vehículos adecuados incluyen solución salina fisiológica, agua bacteriostática, CREMOPHOR EL™ (BASF, Parsippany, NJ) o solución salina regulada con fosfato (PBS). En todos los casos, la composición debe ser estéril y debe ser fluida en la medida en que exista una fácil aplicación con jeringa. Debe ser estable en las condiciones de fabricación y almacenamiento y debe conservarse contra la acción contaminante de microorganismos como bacterias y hongos. El vehículo puede ser un disolvente o medio de dispersión que contenga, por ejemplo, agua, etanol, poliol (por ejemplo, glicerol, propilenglicol y polietilenglicol líquido, y similares), y mezclas adecuadas de los mismos. La fluidez adecuada se puede mantener, por ejemplo, mediante el uso de un recubrimiento como la lecitina, mediante el mantenimiento del tamaño de partícula requerido en el caso de la dispersión y mediante el uso de surfactantes. La prevención de la acción de los microorganismos se puede lograr mediante diversos agentes antibacterianos y antifúngicos, por ejemplo, parabenos, clorobutanol, fenol, ácido ascórbico, timerosal y similares. En ciertas realizaciones de ejemplo, agentes isotónicos, por ejemplo, azúcares, polialcoholes tales como manitol, sorbitol y/o cloruro de sodio, se incluirán en la composición. La absorción prolongada de las composiciones inyectables se puede lograr incluyendo en la composición un agente que retrasa la absorción, por ejemplo, monoestearato de aluminio y gelatina.

20 Las soluciones inyectables estériles pueden prepararse incorporando agentes descritos en el presente documento o sus sales farmacéuticamente aceptables en la cantidad requerida en un disolvente apropiado con uno o una combinación de ingredientes enumerados anteriormente, según se requiera, seguido de esterilización por filtración. En general, las dispersiones se preparan incorporando el compuesto activo en un vehículo estéril que contiene un medio de dispersión básico y los otros ingredientes requeridos de los enumerados anteriormente. En el caso de polvos estériles para la preparación de soluciones inyectables estériles, los métodos de preparación a modo de ejemplo son el secado al vacío y el secado por congelación, que produce un polvo del ingrediente activo más cualquier ingrediente deseado adicional de una solución del mismo filtrada previamente estéril.

25 Las composiciones orales generalmente incluyen un diluyente inerte o un vehículo comestible. Pueden incluirse en cápsulas de gelatina o comprimirse en tabletas. Para el propósito de la administración terapéutica oral, el compuesto activo puede incorporarse con excipientes y usarse en forma de comprimidos, trociscos o cápsulas. Las composiciones orales también pueden prepararse usando un vehículo fluido para usar como un enjuague bucal, en donde el compuesto en el vehículo fluido se aplica oralmente y se agita y expectora o traga. Agentes de unión farmacéuticamente compatibles y/o materiales adyuvantes pueden incluirse como parte de la composición. Los comprimidos, píldoras, cápsulas, trociscos y similares pueden contener cualquiera de los siguientes ingredientes o compuestos de naturaleza similar: un aglutinante como celulosa microcristalina, goma de tragacanto o gelatina; un excipiente tal como almidón o lactosa, un agente desintegrante tal como alginato, ácido, primogel o almidón de maíz; un lubricante tal como estearato de magnesio o esteroides; un deslizante: tal como dióxido de silicio coloidal; un agente edulcorante tal como sacarosa o sacarina; o un agente saborizante tal como menta, salicilato de metilo o saborizante de naranja.

30 En una realización, uno o más agentes descritos en el presente documento o sus sales farmacéuticamente aceptables se preparan con vehículos que protegerán el compuesto contra la eliminación rápida del cuerpo, tal como una formulación de liberación controlada, que incluye implantes y sistemas de administración microencapsulados. Se pueden usar polímeros biodegradables y biocompatibles, como etilenvinilacetato, polianhídridos, ácido poliglicólico,

colágeno, poliolefinos y ácido poliláctico. Los métodos para la preparación de tales formulaciones serán evidentes para los expertos en la técnica. Los materiales también pueden obtenerse comercialmente de Alza Corporation y Nova Pharmaceuticals, Inc. Las suspensiones liposomales (que incluyen liposomas dirigidos a células infectadas con anticuerpos monoclonales contra antígenos virales) también pueden usarse como vehículos farmacéuticamente aceptables. Estos pueden prepararse de acuerdo con métodos conocidos por los expertos en la técnica, por ejemplo, como se describe en la Patente de los Estados Unidos No. 4,522,811.

Las composiciones nasales generalmente incluyen aerosoles nasales e inhalantes. Los aerosoles e inhalantes nasales pueden contener uno o más componentes activos y excipientes tales como conservantes, modificadores de la viscosidad, emulsionantes, agentes reguladores y similares. Los aerosoles nasales se pueden aplicar a la cavidad nasal para uso local y/o sistémico. Los aerosoles nasales se pueden dispensar en un dispensador no presurizado adecuado para el suministro de una dosis medida del componente activo. Los inhalantes nasales están diseñados para ser administrados a los pulmones por inhalación oral para uso local y/o sistémico. Los inhalantes nasales se pueden dispensar mediante un sistema de envase cerrado para administrar una dosis medida de uno o más componentes activos.

En una realización, los inhalantes nasales se usan con un aerosol. Esto se logra preparando un aerosol acuoso, una preparación liposomal o partículas sólidas que contienen el compuesto. Se podría usar una suspensión no acuosa (por ejemplo, propelente de fluorocarbono). Se pueden usar nebulizadores sónicos para minimizar la exposición del agente al corte, lo que puede resultar en la degradación del compuesto.

En general, se hace un aerosol acuoso formulando una solución o suspensión acuosa del agente junto con vehículos y estabilizantes farmacéuticamente aceptables convencionales. Los vehículos y estabilizadores varían con los requisitos del compuesto en particular, pero típicamente incluyen surfactantes no iónicos (Tweens, Pluronic o polietilenglicol), proteínas inocuas como la albúmina sérica, ésteres de sorbitán, ácido oleico, lecitina, aminoácidos tales como glicina, reguladores, sales, azúcares o alcoholes de azúcar. Los aerosoles generalmente se preparan a partir de soluciones isotónicas.

La administración sistémica también puede ser por medios transmucosas o transdérmicos. Para la administración transmucosa o transdérmica, en la formulación se usan penetrantes apropiados para la barrera a permear. Dichos penetrantes son generalmente conocidos en la técnica e incluyen, por ejemplo, para administración transmucosa, detergentes, sales biliares y derivados de ácido fusídico. La administración transmucosa se puede lograr mediante el uso de aerosoles nasales o supositorios. Para la administración transdérmica, los compuestos activos se formulan en ungüentos, pomadas, geles o cremas como se conoce generalmente en la técnica.

Uno o más agentes descritos en el presente documento o sus sales farmacéuticamente aceptables también pueden prepararse en forma de supositorios (por ejemplo, con bases de supositorio convencionales tales como manteca de cacao y otros glicéridos) o enemas de retención para administración rectal.

En una realización, uno o más agentes descritos en el presente documento o sus sales farmacéuticamente aceptables se preparan con vehículos que los protegerán contra la eliminación rápida del cuerpo, tal como una formulación de liberación controlada, que incluye implantes y sistemas de administración microencapsulados. Se pueden usar polímeros biodegradables y biocompatibles, como etilvinilacetato, polianhídridos, ácido poliglicólico, colágeno, poliolefinos y ácido poliláctico. Los métodos para la preparación de tales formulaciones serán evidentes para los expertos en la técnica. Los materiales también pueden obtenerse comercialmente de Alza Corporation y Nova Pharmaceuticals, Inc. Las suspensiones liposomales (que incluyen liposomas dirigidos a células infectadas con anticuerpos monoclonales contra antígenos virales) también pueden usarse como vehículos farmacéuticamente aceptables. Estos pueden prepararse de acuerdo con los métodos conocidos por los expertos en la técnica, por ejemplo, como se describe en la Patente de los Estados Unidos No. 4,522,811.

Es especialmente ventajoso formular composiciones orales, parenterales o de administración directa en el SNC en forma de unidades de dosificación para facilitar la administración y la uniformidad de la dosificación. La forma de unidad de dosificación como se usa en este documento se refiere a unidades físicamente discretas adecuadas como dosificaciones unitarias para el sujeto a tratar; cada unidad contiene una cantidad predeterminada de compuesto activo calculada para producir el efecto terapéutico deseado en asociación con el vehículo farmacéutico requerido. La especificación para las formas unitarias de dosificación de la invención está dictada por y depende directamente de las características únicas del compuesto activo y el efecto terapéutico particular que se va a lograr, y las limitaciones inherentes a la técnica de composición de dicho compuesto activo para el tratamiento de individuos.

La toxicidad y la eficacia terapéutica de uno o más agentes descritos en el presente documento o sus sales farmacéuticamente aceptables pueden determinarse mediante procedimientos farmacéuticos estándar en cultivos celulares, animales experimentales o en un individuo, por ejemplo, en un ser humano. La relación de dosis entre los efectos tóxicos y terapéuticos es el índice terapéutico y puede expresarse como la relación LD50/ED50. Se prefieren los compuestos que exhiben grandes índices terapéuticos. Si bien se pueden usar compuestos que exhiben efectos secundarios tóxicos, se debe tener cuidado al diseñar un sistema de administración que dirija dichos compuestos al sitio del tejido afectado para minimizar el daño potencial a las células no infectadas y, por lo tanto, reducir los efectos secundarios.

Los datos obtenidos de la monitorización del estado inmune de un individuo pueden usarse para formular un rango de dosis para uso en el individuo (por ejemplo, medicina personalizada). La dosis generalmente estará dentro de un rango de concentraciones circulantes que incluyen el ED50 con poca o ninguna toxicidad. La dosis puede variar dentro de este rango dependiendo de la forma de dosificación empleada y de la vía de administración utilizada. Para cualquier compuesto usado en el método de la invención, la dosis terapéuticamente efectiva puede estimarse inicialmente a partir de ensayos de cultivos celulares. Se puede formular una dosis en modelos animales para lograr un rango de concentración circulante en plasma que incluya la IC50 (es decir, la concentración del compuesto de prueba que logra una inhibición de los síntomas a mitad del máximo) determinada en el cultivo celular. Dicha información se puede utilizar para determinar con mayor precisión las dosis útiles en los seres humanos. Los niveles en plasma se pueden medir, por ejemplo, mediante cromatografía líquida de alto rendimiento.

La monitorización de la influencia de una composición farmacéutica sobre la respuesta inmune de un individuo se puede aplicar no solo en la selección de fármacos básicos, sino también en ensayos clínicos. Por ejemplo, la efectividad de una composición farmacéutica determinada por un ensayo de detección tal como se describe en el presente documento para alterar la respuesta inmune de un sujeto puede controlarse en ensayos clínicos.

En ciertas realizaciones de ejemplo, se proporciona un método para controlar la eficacia del tratamiento de un sujeto con un agente descrito en el presente documento que incluye las etapas de (i) obtener una muestra previa a la administración de un sujeto antes de la administración del agente; (ii) detectar el nivel de expresión de una o más secuencias de ácidos nucleicos del sistema inmunitario en la muestra previa a la administración; (iii) obtener una o más muestras posteriores a la administración del sujeto; (iv) detectar el nivel de una o más secuencias de ácidos nucleicos del sistema inmunitario en las muestras posteriores a la administración; (v) comparar el nivel de expresión de una o más secuencias de ácidos nucleicos del sistema inmunitario en la muestra previa a la administración con el nivel de expresión de una o más secuencias de ácidos nucleicos del sistema inmunitario en la muestra o muestras posteriores a la administración; y (vi) la alterar de la administración del agente al sujeto en consecuencia. De acuerdo con una realización de este tipo, los niveles de expresión alterados de una o más secuencias de ácidos nucleicos del sistema inmune pueden usarse como un indicador de la eficacia de un agente, incluso en ausencia de una respuesta fenotípica observable.

En ciertas realizaciones de ejemplo, se proporciona un método para diagnosticar, pronosticar, monitorizar, tratar, mejorar y/o prevenir en un sujeto una enfermedad, trastorno, síntoma y/o afección. Los sujetos en riesgo de una enfermedad y/o trastorno descritos en el presente documento pueden identificarse, por ejemplo, mediante cualquiera o una combinación de ensayos de diagnóstico o pronóstico como se describe en el presente documento. La administración de un agente profiláctico puede ocurrir antes de la manifestación de los síntomas característicos de una enfermedad o trastorno, de tal manera que se prevenga la enfermedad o trastorno o, alternativamente, se retrase su progresión. Dependiendo del tipo de enfermedad o trastorno, uno o más agentes o sales farmacéuticamente aceptables de los mismos pueden usarse para tratar al sujeto. El agente apropiado puede determinarse basándose en los ensayos de selección descritos en el presente documento.

En ciertas realizaciones de ejemplo, se proporcionan métodos de pronóstico, diagnóstico y/o control de uno o más trastornos o enfermedades asociadas con un agente infeccioso. Los agentes infecciosos incluyen, pero no se limitan a, virus, bacterias, hongos, parásitos, proteínas infecciosas y similares.

Los virus incluyen, pero no se limitan a, virus animales de ADN o ARN. Como se usa en el presente documento, los virus de ARN incluyen, pero no se limitan a, familias de virus como Picornaviridae (por ejemplo, poliovirus), Reoviridae (por ejemplo, rotavirus), Togaviridae (por ejemplo, virus de la encefalitis, virus de la fiebre amarilla, virus de la rubéola), Orthomyxoviridae (por ejemplo, virus de la influenza), Paramyxoviridae (por ejemplo, virus sincicial respiratorio, virus del sarampión, virus de las paperas, virus de la parainfluenza), Rhabdoviridae (por ejemplo, el virus de la rabia), Coronaviridae, Bunyaviridae, Flaviviridae, Filoviridae, Arenaviridae, Bunyaviridae y Retroviridae (por ejemplo, virus linfotrópicos de células T humanas (HTLV), virus de inmunodeficiencia humana (VIH)). Como se usa en este documento, los virus de ADN incluyen, pero no se limitan a, familias de virus tales como Papovaviridae (por ejemplo, virus de papiloma), Adenoviridae (por ejemplo, adenovirus), Herpesviridae (por ejemplo, virus de herpes simple) y Poxviridae (por ejemplo, virus de viruela).

Las bacterias incluyen, pero no se limitan a, bacterias gram positivas, bacterias gram negativas, bacterias ácidas y similares.

Como se usa en este documento, las bacterias gram positivas incluyen, pero no se limitan a, Actinomyces israelii, Bacillus anthracis, Bacillus cereus, Clostridium botulinum, Clostridium difficile, Clostridium perfringens, Clostridium tetani, Corynebacterium, Enterococcus faecalis, Listeria monocytogenes, Nocardia, Propionibacterium acnes, Staphylococcus aureus, Staphylococcus epiderm, Streptococcus mutans, Streptococcus pneumoniae y similares. Como se usa en este documento, las bacterias gramnegativas incluyen, pero no se limitan a, Afipia felis, Bacteriodes, Bartonella bacilliformis, Bortadella pertussis, Borrelia burgdorferi, Borrelia recurrentis, Brucella, Calymmatobacterium granulomatis, Campylobacter, Escherichia coli, Francisella tularensis, Gardnerella vaginalis, Haemophilus aegyptius, Haemophilus ducreyi, Haemophilus influenzae, Helicobacter pylori, Legionella pneumophila, Leptospira interrogans, Neisseria meningitidis, Porphyromonas gingivalis, Providencia stuartii, Pseudomonas aeruginosa, Salmonella enteridis, Salmonella typhi, Serratia marcescens, Shigella boydii,

Streptobacillus moniliformis, *Streptococcus pyogenes*, *Treponema pallidum*, *Vibrio cholerae*, *Yersinia enterocolitica*, *Yersinia pestis* y similares.

Como se usa en el presente documento, las bacterias ácido-rápidas incluyen, pero no se limitan a, *Myobacterium avium*, *Myobacterium leprae*, *Myobacterium tuberculosis* y similares.

5 Como se usa en este documento, otras bacterias que no entran en las otras tres categorías incluyen, pero no se limitan a, *Bartonella henselae*, *Chlamydia psittaci*, *Chlamydia trachomatis*, *Coxiella burnetii*, *Mycoplasma pneumoniae*, *Rickettsia akari*, *Rickettsia prowazekii*, *Rickettsia rickettsii*, *Rickettsia tsutsugamushi*, *Rickettsia typhi*, *Ureaplasma urealyticum*, *Diplococcus pneumoniae*, *Ehrlichia chaffeensis*, *Enterococcus faecium*, *Meningococci* y similares.

10 Como se usa en este documento, los hongos incluyen, pero no se limitan a, *Aspergilli*, *Candidae*, *Candida albicans*, *Coccidioides immitis*, *Cryptococci*, y combinaciones de los mismos.

Como se usa en este documento, los microbios parásitos incluyen, pero no se limitan a, *Balantidium coli*, *Cryptosporidium parvum*, *Cyclospora cayatanensis*, *Encephalitozoa*, *Entamoeba histolytica*, *Enterocytozoon bieneusi*, *Giardia lamblia*, *Leishmaniae*, *Plasmodii*, *Toxoplasma gondii*, *Trypanosomae*, *trapezoidal amoeba* y similares.

15 Como se usa en el presente documento, los parásitos incluyen gusanos (por ejemplo, helmintos), particularmente gusanos parasitarios que incluyen, pero no se limitan a, *Nematoda* (gusanos redondos, por ejemplo, gusanos de látigo, gusanos del gancho, lombrices, ascárides, filáridos y similares), *Cestoda* (por ejemplo, tenias).

20 Como se usa en el presente documento, las proteínas infecciosas incluyen priones. Los trastornos causados por priones incluyen, entre otros, trastornos humanos como la enfermedad de Creutzfeldt-Jakob (ECJ) (que incluye, por ejemplo, la enfermedad iatrogénica de Creutzfeldt-Jakob (ICJD), la variante de la enfermedad de Creutzfeldt-Jakob (vCJD), la enfermedad de Creutzfeldt-Jakob familiar (fCJD), y enfermedad de Creutzfeldt-Jakob esporádica (sCJD), síndrome de Gerstmann-Sträussler-Scheinker (GSS), insomnio familiar fatal (FFI), insomnio fatal esporádico (SFI), kuru y similares, así como trastornos en animales como la tembladera (ovejas y cabras), encefalopatía espongiiforme bovina (EEB) (ganado bovino), encefalopatía de visón transmisible (TME), enfermedad de desgaste crónico (CWD) (alce, ciervo mula), encefalopatía espongiiforme felina (gatos), encefalopatía de ungulados exótica (EUE) (*nyala*, *oryx*, *kudu mayor*), encefalopatía espongiiforme del avestruz y similares.

30 En ciertas realizaciones de ejemplo, se proporcionan métodos de pronóstico, diagnóstico y/o control de uno o más trastornos de proliferación celular. Los trastornos de proliferación celular pretenden incluir trastornos asociados con la rápida proliferación. Como se usa en el presente documento, el término "trastorno proliferativo celular" incluye trastornos caracterizados por una proliferación indeseable o inapropiada de uno o más subconjuntos de células en un organismo multicelular. El término "cáncer" se refiere a varios tipos de neoplasias malignas, la mayoría de las cuales pueden invadir los tejidos circundantes, y pueden metastatizar a diferentes sitios (véase, por ejemplo, *PDR Medical Dictionary 1st edition (1995)*, incorporado aquí como referencia en su totalidad para todos los fines). Los términos "neoplasia" y "tumor" se refieren a un tejido anormal que crece por proliferación celular más rápidamente de lo normal. Id. Dicho tejido anormal muestra una falta parcial o completa de organización estructural y coordinación funcional con el tejido normal que puede ser benigno (es decir, tumor benigno) o maligno (es decir, tumor maligno).

40 La expresión "tratamiento de trastornos proliferativos celulares" pretende incluir la prevención de la inducción, el inicio, el establecimiento o el crecimiento de neoplasmas en un sujeto o una reducción en el crecimiento de neoplasmas preexistentes en un sujeto. El lenguaje también puede describir la inhibición de la invasión de células neoplásicas en tejidos vecinos o la metástasis de una neoplasia de un sitio a otro. Ejemplos de los tipos de neoplasmas que pretenden abarcar la presente invención incluyen, entre otros, los neoplasmas asociados con cánceres de mama, piel, huesos, próstata, ovarios, útero, cuello uterino, hígado, pulmón, cerebro, laringe, vesícula biliar, páncreas, recto, paratiroides, tiroides, glándulas suprarrenales, sistema inmunológico, tejido neural, cabeza y cuello, colon, estómago, bronquios y/o riñones.

45 Los trastornos proliferativos celulares pueden incluir además trastornos asociados con la hiperproliferación de células del músculo liso vascular tales como trastornos cardiovasculares proliferativos, por ejemplo, aterosclerosis y restenosis. Los trastornos de la proliferación celular también pueden incluir trastornos tales como trastornos proliferativos de la piel, por ejemplo, ictiosis ligada al X, psoriasis, dermatitis atópica, dermatitis alérgica de contacto, hiperqueratosis epidermolítica y dermatitis seborreica. Los trastornos de proliferación celular pueden incluir además trastornos tales como enfermedad poliquística renal autosómica dominante (ADPKD), mastocitosis y trastornos de proliferación celular causados por agentes infecciosos tales como virus.

55 En ciertas realizaciones de ejemplo, se proporcionan métodos de pronóstico, diagnóstico y/o control de uno o más trastornos autoinmunes. Como se usa en el presente documento, el término "trastorno autoinmune" es una enfermedad o trastorno causado por un sujeto que produce una respuesta inmune inadecuada contra sus propios tejidos. Como se usa en este documento, un trastorno autoinmune incluye, pero no se limita a, trastornos como la enfermedad de Addison, alopecia areata, espondilitis anquilosante, síndrome de antifosfolípido, anemia hemolítica autoinmune, hepatitis autoinmune, enfermedad autoinmune del oído interno (EAED), síndrome linfoproliferativo autoinmune (ALPS), púrpura trombocitopénica autoinmune (ATP), enfermedad de Balo, enfermedad de Bechet, cardiomiopatía penfigoide bullosa, enfermedad celíaca, dermatitis herpetiforme, síndrome de disfunción inmune por

fatiga crónica (CFIDS), polineuropatía desmielinizante inflamatoria crónica, penfigoide cicatricial, aglutinina fría, síndrome CREST, enfermedad de Crohn, enfermedad de Degos, dermatomiositis, dermatomiositis juvenil, lupus discoide, crioglobulinemia mixta esencial, fibromialgia-fibromiositis, enfermedad de Graves, Guillain-Barrera, tiroiditis de Hashimoto, fibrosis pulmonar idiopática, trombocitopenia púrpura idiopática (PTI), nefropatía por IgA, diabetes insulino-dependiente, artritis juvenil, liquen plano, lupus, enfermedad de Ménière, enfermedad mixta del tejido conectivo, esclerosis múltiple, miastemia grave, pénfigo vulgaris, anemia perniciosa, poliarteritis nodosa, policondritis, síndromes poliglandulares, polimialgia reumática, polimiositis y dermatomiositis, agammaglobulinemia primaria, cirrosis biliar primaria, psoriasis, fenómeno de Raynaud, síndrome de Reiter, fiebre reumática, artritis reumatoide, sarcoidosis, esclerodermia, síndrome de Sjögren, síndrome de la persona rígida, artritis de Takayasu, arteritis temporal/arteritis de células gigantes, colitis ulcerosa, vasculitis, vitiligo, granulomatosis de Wegener y similares (Consulte el sitio web de la American Autoimmune Related Diseases Association, Inc.: aarda.org).

En ciertas realizaciones de ejemplo, la naturaleza y las características de los síntomas, afecciones, enfermedades y/o trastornos se reducen por los métodos de la presente invención en comparación con la naturaleza y características de los síntomas, afecciones, enfermedades y/o trastornos observados en un paciente o muestra (por ejemplo, una muestra de prueba o una muestra tomada de un sujeto antes, durante o después del tratamiento). En ciertos aspectos, la naturaleza y las características de los síntomas, afecciones y enfermedades y/o trastornos de los fenotipos se reducen en aproximadamente un 5%, 10%, 15%, 20%, 25%, 30%, 35%, 40%, 45%, 50%, 55%, 60%, 65%, 70%, 75%, 80%, 85%, 90%, 91%, 92%, 93%, 94%, 95%, 96%, 97%, 98%, 99%, 99.9% o más, o cualquier rango(s) entre ellos.

En ciertas realizaciones de ejemplo, un método para modular, mejorar, prevenir y/o tratar enfermedades, síntomas y/o trastornos como se describe en el presente documento incluye la etapa de administrar una cantidad terapéuticamente eficaz de un agente a un sujeto. Como se define aquí, una cantidad terapéuticamente eficaz de agente (es decir, una dosis efectiva) varía de aproximadamente 0.0001 a 30 mg/kg de peso corporal, de aproximadamente 0.001 a 25 mg/kg de peso corporal, de aproximadamente 0.01 a 20 mg/kg de peso corporal, de aproximadamente 0.1 a 15 mg/kg de peso corporal, o de aproximadamente 1 a 10 mg/kg, de 2 a 9 mg/kg, de 3 a 8 mg/kg, de 4 a 7 mg/kg, o de 5 a 6 mg/kg peso corporal. El experto en la materia apreciará que ciertos factores pueden influir en la dosis requerida para tratar eficazmente a un sujeto, incluidos, entre otros, la gravedad de la enfermedad o trastorno, los tratamientos anteriores, la salud general y/o la edad del sujeto y otras enfermedades presentes. Además, el tratamiento de un sujeto con una cantidad terapéuticamente eficaz de uno o agentes o sales farmacéuticamente aceptables del mismo puede incluir un solo tratamiento o, en ciertas realizaciones de ejemplo, puede incluir una serie de tratamientos. También se apreciará que la dosis efectiva del agente utilizado para el tratamiento puede aumentar o disminuir en el transcurso de un tratamiento particular. Los cambios en la dosis pueden resultar de los resultados de los ensayos de diagnóstico como se describe en el presente documento. Las composiciones farmacéuticas pueden incluirse en un envase, envase o dispensador junto con instrucciones para la administración.

Las realizaciones de la presente invención se dirigen adicionalmente a métodos de emparejamiento de cadenas pesadas y livianas haciendo coincidir sus frecuencias de clon relativa o haciendo coincidir sus series de tiempo similares. Otras realizaciones incluyen el emparejamiento de cadenas pesadas y livianas en el nivel de una sola célula simultáneamente en muchas células en paralelo en un solo medio de reacción. Dichos métodos incluyen la secuenciación de alto rendimiento de los componentes resultantes de una reacción inmune que utiliza métodos para aislar células, incluida la bioquímica en células formadas con formaldehído entrecruzado/permeabilizado, células encapsuladas en emulsión, células encapsuladas en perlas de agarosa, transfección de liposomas de células individuales, pequeñas transfección con oligonucleótidos de células individuales, incluidos ácidos nucleicos bloqueados o sondas fluorescentes, y similares. Tales métodos reducen la contaminación cruzada y los emparejamientos incorrectos. Los métodos de emparejamiento de cadenas pesadas y livianas descritos en este documento utilizan métodos enzimáticos que incluyen SOE-PCR que puede dar lugar a la incorporación de secuencias de forro de scFv, ligación directa que puede dar como resultado la incorporación de secuencia de engarce de scFv y también puede incluir ligaciones bimoleculares o trimoleculares, clonación de USER, captura de perlas tal como por amplificación en perlas o hibridación en perlas o captura por oligonucleótidos pre-fusionados con secuencias complementarias específicas de la cadena inmune.

Las realizaciones de la presente invención están aún más dirigidas a métodos para obtener moléculas de cadena inmune completa mediante hibridación usando información de CDR3. Dicha realización incluye el uso de tecnologías de secuenciación de lectura corta para seleccionar secuencias de CDR3 y usar estas secuencias de CDR3 para hibridar con una molécula de cadena inmunitaria completa usando oligonucleótidos complementarios.

Las realizaciones de la presente invención incluyen además métodos para reducir la contaminación cruzada o el emparejamiento incorrecto de cadenas pesadas y livianas mediante exonucleasas unidas a perlas y métodos para controlar la contaminación cruzada o el emparejamiento incorrecto de cadenas pesadas y livianas mediante PCR en tiempo real. Los mismos métodos o similares para el emparejamiento de cadenas pesadas y livianas descritos en el presente documento también pueden extenderse a la selección de bibliotecas de anticuerpos frente a bibliotecas de antígenos. De acuerdo con un aspecto, las bibliotecas grandes y complejas de reactivos de afinidad, tales como anticuerpos, nanocuerpos y similares, se analizan frente a bibliotecas grandes y complejas de dianas antigénicas en una única reacción, seguido de un acoplamiento de ADN que codifica la secuencia del anticuerpo al ADN que codifica la secuencia antigénica.

Las realizaciones de la invención incluyen el uso de software informático para automatizar el diseño y/o análisis de secuencias de ácidos nucleicos. Dicho software se puede utilizar junto con individuos que realizan la síntesis de polinucleótidos manual o de forma semiautomática o combinados con un sistema de síntesis automatizado. En al menos algunas realizaciones, el software de diseño/análisis de genes/oligonucleótidos se implementa en un programa escrito en el lenguaje de programación JAVA. El programa se puede compilar en un ejecutable que luego puede ejecutarse desde un indicador de comandos en el sistema operativo WINDOWS XP. El código de software ejemplar se establece como Apéndice D. A menos que se establezca específicamente en las reivindicaciones, la invención no se limita a implementaciones que utilizan los mismos algoritmos, estructura organizativa u otras características específicas del código de software ejemplar. La invención tampoco se limita a la implementación utilizando un lenguaje de programación específico, un entorno de sistema operativo o una plataforma de hardware.

Debe entenderse que las realizaciones de la presente invención que se han descrito son meramente ilustrativas de algunas de las aplicaciones de los principios de la presente invención.

Los siguientes ejemplos se exponen como representativos de la presente invención. Estos ejemplos no deben interpretarse como limitantes del alcance de la invención, ya que estas y otras realizaciones equivalentes serán evidentes a la vista de la presente divulgación, tablas, figuras, apéndices y reivindicaciones adjuntas.

Ejemplo I

Seguimiento de la dinámica de la respuesta inmune humana a través de la secuenciación VDJ

Se extrajeron muestras de sangre periférica de un sujeto a -14 días antes de la vacunación, aproximadamente una hora después de la vacunación (día 0), y 1, 3, 7, 14, 21 y 28 días después de la vacunación. Para cada muestra, se extrajo el ARN total de los leucocitos de sangre periférica, a partir de los cuales se generaron las bibliotecas de secuenciación de ADNc y código de barras 454 utilizando cebadores específicos para la región V_H completa (Figura 1B). El ARNm de células B se analizó para evitar el cebado cruzado del ADN de la línea germinal, para evitar reordenamientos de receptores no funcionales, para beneficiarse de la posible expansión clonal de células específicas de antígeno, y para aprovechar los mayores números de copias de ARNm en células que expresan anticuerpos. Se utilizó la plataforma de secuenciación Roche 454, ya que proporcionaba longitudes de lectura lo suficientemente largas para cubrir la región V_H completa.

A lo largo del curso de 7 ejecuciones de la secuenciación de 454 GS FLX, se obtuvieron aproximadamente 5.5 millones de lecturas de tamaño filtrado que posteriormente se alinearon con la base de datos de referencia de IMGt (Sitio web mundial: imgt.cines.fr/) (Lefranc et al. (2009) *Nucleic Acids Res.* 37: D1006). Para ello, se identificaron las regiones V, luego J y finalmente D, ya que su longitud decreciente permitió la identificación más confiable. El nuevo algoritmo descrito aquí seleccionó las pocas mejores coincidencias al comparar las frecuencias de palabras y las puntuó utilizando la alineación de programación dinámica (Durbin, R., Eddy, S.R., Krogh, A. and Mitchison, G. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids* (Cambridge Univ. Press, Cambridge; 1998)). Para definir clones, el algoritmo luego sometió a partición las lecturas de acuerdo con su uso de V-J y agrupó sus uniones CDR3 usando agrupamiento jerárquico aglomerado con distancia de edición como la métrica (Gan, G., Ma, C., and Wu, J. *Data Agrupación: Theory, Algorithms, and Applications*. (SIAM American Statistical Association, Filadelfia, PA, Alexandria, VA; 2007)). Si bien las coincidencias de las regiones V y J generalmente no fueron ambiguas, las identificaciones de la región D no se enfocaron ya que los algoritmos a menudo produjeron alineaciones ambiguas debido a la corta longitud de las regiones D junto con la naturaleza altamente mutada de la CDR3 (Figura 8). Un total de 4.027.253 lecturas pasaron exitosamente este proceso (incluida la identificación con código de barras), a partir del cual se identificó el isotipo de inmunoglobulina buscando las secuencias de cebadores correspondientes en el extremo de la región constante de la lectura (Figura 26).

Dado que se estima que la diversidad potencial del repertorio humano es bastante significativa (hasta 10^{14} según algunas estimaciones) (Davis et al. (1988) *Nature* 334:395), la factibilidad técnica y la reproducibilidad de los métodos descritos en este documento se establecen de la siguiente manera. Se secuenció una biblioteca de todos los puntos de tiempo dos veces (secuenciación de réplicas SR1 y SR2), y una tercera biblioteca completamente independiente preparada a partir de las mismas muestras de ARN se secuenció una vez (réplica técnica TR1). Entre estas tres ejecuciones de secuenciación, se identificaron 477.118 clones únicos donde solo el 3% se compartió entre las tres corridas y el 14% se observó en al menos dos corridas (Figura 2A). Sin embargo, esos clones compartidos representaron el 59% y el 71% de todas las lecturas, respectivamente, lo que indica que los clones altamente expresados en realidad se muestrean significativamente entre ejecuciones duplicadas. Esto se validó adicionalmente mediante una fuerte correlación entre muestras técnicas replicadas, lo que confirma la reproducibilidad técnica del enfoque descrito aquí (Figuras 2B y 9). Además, el remuestreo de los datos mostró que 10^5 lecturas eran suficientes para obtener altas correlaciones entre réplicas (Figura 2C). Al calcular los coeficientes de variación (CV), se determinó que se podía obtener una cuantificación confiable de frecuencias de clones tan bajas como 10^{-4} , aunque este número dependía de la profundidad de secuenciación de una muestra dada (Figura 10).

Existen métodos desarrollados para evaluar estructuras de poblaciones ecológicas. (Magurran, A.E. *Measuring Biological Diversity*. (Blackwell Pub., Malden, Ma.; 2004)). Las gráficas de distribución de tamaño de clonación y abundancia de rangos mostraron que la estructura clonal sigue una distribución de ley de potencia con un exponente

de 1.97 en >2 décadas (Figuras 2D y 11). Sin pretender estar limitados por la teoría científica, esto podría reflejar las presiones de selección evolutiva experimentadas por las células durante los procesos de diferenciación y maduración de la afinidad (Caldarelli et al. (2002) Phys. Rev. Lett. 89: 258702; Bianconi and Barabasi (2001) Europhysics Letters 54: 436). Para estimar la diversidad del repertorio V_H total en una muestra de sangre típica de 1 mL, se utilizaron los estimadores de riqueza de especies no paramétricas de Chao1 y ACE (Magurran, Supra). Una ejecución de secuenciación completa también se comprometió a un solo punto de tiempo (ejecución DR1; día -14) para obtener una estimación más confiable. Se observaron puntos de inflexión y convexidad en casi todas las curvas de estimador re-muestreadas, lo que indica que las muestras fueron suficientes para realizar estimaciones confiables (Figuras 12A-12D). Se determinó que una muestra de sangre de 1 mL contenía aproximadamente 100.000-500.000 clones de cadena pesada únicos, dependiendo de la muestra de sangre y el estimador utilizado (Figura 2E). Finalmente, dado que solo se consideraron las cadenas pesadas de inmunoglobulina, y sin la intención de limitarse a la teoría científica, se espera que la diversidad real sea al menos un orden de magnitud mayor cuando se empareja con cadenas livianas. Estos datos demuestran que se puede lograr un muestreo adecuado de la diversidad del repertorio humano en una sola extracción de sangre de 1 mL, lo que indica la utilidad potencial de este enfoque como una herramienta de diagnóstico clínico personalizada para identificar la exposición actual o previa a patógenos (Lerner et al. (1991) Proc. Natl. Acad. Sci. USA 88: 9705). Se generará una gran base de datos de secuencias de anticuerpos anotadas de especificidad.

A continuación, se evaluaron las características del repertorio inmune estático. En general, el uso de V y J fue altamente no uniforme (Figura 3A). Los segmentos V observados con mayor frecuencia fueron IGHV3-23 (11.4% de todas las lecturas), IGHV4-59 (8.5%), IGHV3-30-3 (7.3%) mientras que los segmentos J más frecuentes fueron IGHJ4 (35.1%) e IGHJ6 (18.9%), de acuerdo con estudios anteriores (Boyd, Supra; Brezinschek et al. (1995) J. Immunol. 155: 190; Glanville et al. (2009) Proc. Natl. Acad. Sci. USA (106: 20216). Luego se analizó la distribución de las longitudes de unión de CDR3 (Figura 3B). El percentil 95 de dos lados de las longitudes de CDR3 observadas fue de 33 nucleótidos (nt) a 75 nt, con una longitud media de 54 nt (con la unión más larga observada a 146 nt), que muestra una similitud con la distribución del tamaño del receptor CDR3 de las células T (Freeman et al., Supra) y los datos de IMGT/LIGM (Lefranc, Supra). En un intento de comparación con los métodos tradicionales de baja resolución para analizar los repertorios inmunes, seudo también se generaron gráficos de tipos de espectros en los que se utiliza la diversidad de longitud del receptor como un sustituto de la diversidad general (Figura 3C y Figuras 13A-13B) (Pannetier et al. (1993) Proc. Natl. Acad. Sci. USA 90: 4319). Finalmente, se analizó la distribución de isotipos de inmunoglobulina y se observó que los anticuerpos IgM eran los más abundantes (37% de todas las lecturas), seguidos de IgA (36%), IgG (23%), IgD (4%) e IgE (0.01%). Aunque se midieron los niveles de ARNm expresados en las células B circulantes, estos números aún eran consistentes con la edad del sujeto y los títulos de proteína plasmática esperados (Lefranc, M.-P. & Lefranc, G.E. The Immunoglobulin Factsbook. (Academic Press, San Diego; 2001). En conjunto, estos resultados muestran el primer perfil en profundidad del repertorio de la cadena pesada de inmunoglobulina expresada en un humano individual.

Si bien los análisis anteriores se centraron en establecer la reproducibilidad y la creación de perfiles de la VDJ-oma estática, fue de gran interés examinar las fluctuaciones dinámicas del repertorio inmunológico. Con la esperanza de capturar al menos un evento inmunológico, se indujeron respuestas a múltiples problemas inmunes al vacunar al sujeto contra tres patógenos para los cuales su historial médico sugiere que debería tener memoria inmune: influenza (3 cepas), hepatitis A y hepatitis B. Las frecuencias de clones se rastrearon a través del tiempo para producir casi 500.000 series de tiempo en los ocho puntos de tiempo, revelando la alta complejidad de la dinámica del repertorio (Figura 4A). Mediante la selección aleatoria de clones a través de ejecuciones técnicas repetidas, se observaron correlaciones sólidas cualitativamente a diferentes niveles de expresión, lo que confirmó aún más la reproducibilidad del enfoque descrito aquí (Figura 14). El agrupamiento inicial de la serie de tiempos utilizando un algoritmo de k-medias mostró que los grupos más poblados eran de clones efímeros de varios niveles de expresión observados en cada punto de tiempo individual. Sin embargo, después de eliminar estos clones de punto de tiempo único, los grupos resultantes exhibieron una amplia gama de comportamientos dinámicos (Figuras 15A-15B).

Tradicionalmente, los intentos de caracterizar las respuestas a los desafíos inmunes se realizan mediante la clasificación de las células con ciertos marcadores de superficie, lo que permite la generación de anticuerpos específicos de antígeno mediante el análisis de las cadenas pesadas y livianas de las células individuales (Paul, Supra; Wrarmert et al. (2008) Nature 453: 667). En contraste con este enfoque, se realizó un intento de identificar linfocitos específicos de antígeno sin conocimiento previo de antígenos específicos o estados de activación celular y sin utilizar ensayos funcionales. En un intento por identificar clones específicos para los desafíos inmunes, se utilizó un método de agrupamiento de mapas autoorganizados (SOM) (Kohonen, T. Self-Organizing Maps, Ed. 2 (Springer, Berlín; Nueva York; 1997); Eichler et al. (2003) Bioinformatics 19: 2321). Esto reveló grupos de clones en los días 7 y 21 que exhibieron los patrones de activación típicamente esperados de células secretoras de anticuerpos (ASC) y células de memoria B (Figura 4B) (Abbas et al., Supra; Murphy et al., Supra; Wrarmert et al. al., supra). Estos clones también se correspondían con los picos principales observados en los pseudotipos de espectros tipográficos (identificados por los números asociados con cada pico en la Figura 3C). Sorprendentemente, los clones más dinámicos y altamente expresados (en órdenes de magnitud) se observaron el día 0 (aproximadamente 1 hora después de la vacunación) y el día 1, con una caída significativa en el día 3 (Figura 4A). Si bien no se puede excluir la posibilidad de que el sujeto experimentara inadvertidamente una respuesta inmune no relacionada con las vacunas, se cree que se observó una respuesta de memoria inducida por la vacunación. Si bien se sabe que las respuestas de memoria generan títulos

altos de anticuerpos rápidamente, se espera que alcancen un máximo de 3-5 días después de la exposición (Abbas et al., Supra; Murphy et al., Supra; Paul, Supra). Mientras que los títulos séricos más altos inducidos por la exposición pueden ocurrir en los días 3-5, los niveles más altos de expresión de ARNm de las células correspondientes se observaron el día 1, lo que sugiere que los estudios funcionales basados en la clonación de muestras de ADNc recolectados en los días 3-5 probablemente perderían a los candidatos de interés. Sin embargo, es importante tener en cuenta que se tomaron muestras de los ARNm de las células B circulantes, que preceden a la acumulación de proteínas de anticuerpos reales. En contraste con las respuestas de los días 7 y 21, el inicio rápido y la mayor magnitud de la respuesta de memoria del día 1 indican que estos clones pueden haberse pasado por alto en estudios funcionales de descubrimiento de anticuerpos monoclonales terapéuticos (Wrammert et al., Supra; Lanzavecchia et al. (2006) Immunol. Rev. 211: 303; Jin et al. (2009) Nat. Med. 15: 1088).

A continuación, se observaron fluctuaciones globales en la distribución de isotipos en el transcurso del experimento. Curiosamente, se observó un pequeño aumento de IgG en el día 7, mientras que en el día 21, se observó un aumento simultáneo de IgA y una disminución de IgM, lo que indica un posible cambio de clase en los clones inducidos por la vacuna (Figura 4C) (Cerutti (2008) Nat. Rev. Immunol. 8: 421). Para obtener una vista más detallada de la dinámica de isotipos, se analizaron las distribuciones de isotipos de los clones seleccionados por SOM (Figura 4D y Figura 16). Los clones del día 7 estaban compuestos principalmente del isotipo IgM e IgG, mientras que los clones del día 21 estaban compuestos principalmente de IgM e IgA. Lo más interesante es que los clones del día 1 estuvieron dominados por IgA, lo que refleja un papel importante para este isotipo en la defensa antiviral (Cerutti, Supra; Qiao et al. (2006) Nat. Immunol. 7: 302). Dicha respuesta de memoria IgA es característica de la reexposición a antígenos de la gripe (Cox et al. (2006) Vaccine 24: 6577; Wright et al. (1983) Infect. Immun. 40: 1092). Si bien los anticuerpos IgA son en gran parte responsables de la defensa en las secreciones de la mucosa (Cerutti, Supra; Stavnezer (1996) Adv. Immunol. 61:79), se informa que la vacunación contra la influenza (tanto nasal como intramuscular) provoca una respuesta sistémica rápida, fuerte y de corta duración de ASC IgA específica de influenza en sangre periférica (Cox et al., Supra). Sin embargo, la magnitud de la respuesta observada en las horas posteriores a la vacunación indica un nuevo papel efector para la IgA plasmática en la memoria antiviral, y destaca la IgA como una posible inmunoterapia no tradicional en enfermedades virales.

La Figura 6 representa el análisis de clasificación por FACS de algunas de las secuencias seleccionadas de la Figura 4, donde el antígeno de hemaglutinina (Brisbane 10 o Brisbane 59) se recubrió en perlas y los candidatos de anticuerpos se marcaron de manera fluorescente de manera que la interacción correcta entre el antígeno y el anticuerpo daría lugar a un cambio de los picos durante el análisis FACS. El análisis adicional de microbalanza de cristal de cuarzo (QCM) en la Figura 7 demuestra además la afinidad de unión (con el cálculo de kD) del anticuerpo recién descubierto, lo que demuestra el potencial de los métodos y composiciones descritos en el presente documento para identificar de manera eficiente los candidatos de anticuerpos.

De acuerdo con el aspecto de que la invención es una herramienta de diagnóstico personalizada, las lecturas se alinearon con el subconjunto de secuencias de anticuerpos en la base de datos IMGT/LIGM para las cuales existen especificidades de antígeno conocidas (Lefranc et al., Supra). Los CDR se intentaron hacer coincidir perfectamente. Como el conjunto de datos IMGT/LIGM anotado en la especificidad era bastante pequeño (menos de 9.000 secuencias), no se esperaba que se encontraran coincidencias casi perfectas. Sorprendentemente, sin embargo, un solo clon de secuencias logró igualar un anticuerpo anti-digoxina con una única mutación de aminoácido en la periferia de CDR1 (Figura 17). Como el sujeto experimentó dos episodios cardíacos en 1989 y 1994, no se pudo descartar que se le administró digoxina y desarrolló sensibilidad a ella. En cuanto a los candidatos seleccionados de SOM de los días 1, 7 y 21, se encontraron múltiples coincidencias con varios niveles de identidad de secuencia, algunos de los cuales coincidían con anticuerpos específicos de influenza o de hepatitis.

Se describe aquí el análisis de secuenciación de alto rendimiento del repertorio de cadenas pesadas de anticuerpos de células B humanas en respuesta a una vacuna en un experimento de curso temporal. Se ha demostrado aquí que, sin conocimiento previo de antígenos o estados de activación celular, muestras de sangre de un mililitro fueron suficientes para rastrear de manera reproducible la dinámica del sistema inmunológico, a pesar de la gran diversidad y las rápidas fluctuaciones de su repertorio. Debido a esta impresionante diversidad y la influencia del historial de exposición, la respuesta obtenida durante la vacunación humana es un componente importante de la medicina personalizada y la caracterización requerida a nivel individual. Por lo tanto, en el contexto de la medicina personalizada, el enfoque descrito aquí se ha demostrado con éxito para el diagnóstico clínico y el descubrimiento terapéutico.

Referencias

Freeman et al. (2009) Genome Res. 19:1817

Weinstein et al. (2009) Science 324:807

Boyd et al. (2009) Sci. Transl. Med. 1:12ra23

Allison (2008) Nat. Biotechnol. 26:509

Wrammert et al. (2008) Nature 453:667

Draenert et al. (2006) *J. Exp. Med.* 203:529

Pereyra et al. (2008) *J. Infect. Dis.* 197:563

Weinstein et al. (2009) *Science* 324:807

Freeman et al. (2009) *Genome Res.* 19:1817

5 Glanville et al. (2009) *Proc. Natl. Acad. Sci. USA* 106:20216

Kenneth et al. (2008) *Janeway's Immunobiology*. Garland Science, New York, 7th ed.

Lefranc and Lefranc (2001) *The Immunoglobulin Factsbook*. Academic Press, San Diego

Moore and Clayton (2003) *Nature* 426 :725

Lerner et al. (1001) *Proc. Natl. Acad. Sci. USA* 88:9705

10 Brezinschek et al. (1998) *J. Immunol.* 160:4762

Brezinschek et al. (1995) *J. Immunol.* 155:190

Pannetier et al. (1995) *Immunol. Today* 16:176

Pannetier et al. (1993) *Proc. Natl. Acad. Sci. USA* (1993) 90:4319

Marasco and Sui (2007) *Nat. Biotechnol.* 25:1421

15 Ejemplo II

Repertorio de clonación de cadenas pareadas pesadas y livianas

Para confirmar rigurosamente que ciertos clones están implicados en las respuestas inmunes o para descubrir nuevos anticuerpos contra antígenos de interés, es deseable capturar con éxito las cadenas pesadas y livianas de los anticuerpos individuales. Debido a la falta de métodos para capturar cadenas VH y VL pareadas en alto rendimiento, los protocolos conocidos en la técnica implican la clasificación de células individuales en pozos individuales y la realización de PCR para las cadenas pesadas y livianas en serie (por ejemplo, Meijer et al. (2006) *J. Mol. Biol.* 358: 764). Sin embargo, incluso con robots de manejo de líquidos automatizados, los rendimientos típicos están prácticamente limitados a 10^6 (a un gran coste). Otra solución popular es capturar repertorios de cadenas pesadas y livianas por separado y asociarlos aleatoriamente entre sí en vectores de expresión (Wright et al. (2009) *Proc. Natl. Acad. Sci. USA* 106: 19078). Sin embargo, el emparejamiento de cadenas pesadas y livianas probablemente está lejos de ser distribuido al azar. En consecuencia, estos métodos no proporcionan representaciones precisas de los repertorios subyacentes.

Este ejemplo describe un método general para capturar cadenas pesadas y livianas emparejadas en millones de células individuales en un formato de reacción única. Se proporcionan múltiples métodos que permiten la manipulación simultánea de millones de células en paralelo, mientras que las mantienen aisladas entre sí para mantener el emparejamiento de la cadena natural.

Métodos para el acoplamiento unicelular de cadenas.

Se proporcionan múltiples métodos para acoplar las cadenas pesadas y livianas de celdas individuales. Los métodos generales se dividen en dos partes: la bioquímica de enlace de cadena y el método de aislamiento celular. Las soluciones para cada parte se eligen principalmente de forma independiente y se exploran múltiples combinaciones.

Algunos de los métodos bioquímicos se basan en la PCR. La elección entre la amplificación de las cadenas diana del ADN genómico o la realización de RT-PCR se realiza en función de la naturaleza de los ARNm expresados. El primero no requiere un paso de transcripción reversa, pero tiene el riesgo de amplificar los receptores no funcionales, mientras que el último puede beneficiarse de un mayor número de copias y solo debería capturar receptores expresos y funcionales. El entrecruzamiento físico se produce opcionalmente a través de múltiples mecanismos. El primero es la PCR estándar de empalme por solapamiento (SOE-PCR) (o PCR de fusión o PCR cruzada), donde dos de los cebadores de PCR tienen secuencias complementarias para que los dos amplicones funcionen como cebadores y se fusionen entre sí (Heckman and Pease (2007) *Nat. Protocol.* 2:924). Una ventaja de este método es que la secuencia de superposición puede diseñarse de manera que la construcción fusionada se encuentre inmediatamente en un formato scFv utilizable. El siguiente mecanismo es similar al SOE-PCR en donde las etiquetas se incorporan en los cebadores de PCR. En este caso, las etiquetas contienen sitios loxP, de modo que la fusión se producirá en la recombinación mediada por Cre (Albert et al. (1995) *Plant J.* 7:649; Chapal et al. (1997) *Biotechniques* 23:518). Finalmente, en el caso de los métodos de emulsión, otra opción bioquímica es amplificar tanto las cadenas pesadas como las livianas en perlas (Diehl et al. (2006) *Nat. Methods* 3: 551; Shendure et al. (2005) *Science* 309:1728). Una ventaja es que las perlas se pueden procesar inmediatamente para la secuenciación en sistemas de secuenciación

de próxima generación basados en perlas. Sin embargo, esto también puede ser una desventaja, ya que puede limitar el rango de opciones después del acoplamiento de la cadena. Un método alternativo basado en perlas es amplificar ambas cadenas en perlas y luego acoplar las cadenas en las perlas. Esto aumenta la especificidad de todo el proceso, aunque a mayor complejidad del protocolo. Cada uno de estos métodos se resumen en la Figura 27.

5 Los métodos de aislamiento celular se dividen en dos categorías principales: métodos in-cell y métodos de emulsión. Para los métodos dentro de la célula (Embleton et al. (1992) *Nucleic Acids Res.* 20:3831), las células se fijan en formalina y se permeabilizan para permitir la difusión de reactivos bioquímicos en la célula. La membrana celular funciona como la barrera que evita la contaminación cruzada de cadenas pesadas y livianas entre las células. Las ventajas de este enfoque general son la relativa simplicidad de fijar las células y también la capacidad de aplicar en serie conjuntos de reactivos a todas las células en paralelo. Sin embargo, sin la intención de limitarse a la teoría científica, la permeabilización puede aumentar potencialmente la posibilidad de que las cadenas de Ig se escapen de las células y provoquen una contaminación cruzada.

10 Para los métodos basados en emulsión, las células individuales se colocan en compartimentos individuales de una emulsión de agua en aceite (Clausell-Tormos et al. (2008) *Chem. Biol.* 15:427; Leamon et al. (2006) *Nat. Métodos* 3:541). Una ventaja de este tipo de enfoque es que la separación de los compartimentos a base de aceite puede proporcionar un aislamiento casi absoluto de la contaminación cruzada de la cadena. Pero mientras que los compartimentos separados por aceite pueden detener cualquier intercambio de material entre compartimentos, un problema común de las emulsiones de ciclos térmicos es que los compartimentos se fusionan entre sí, lo que lleva a la no capacidad de clonación. Además, es considerablemente más difícil manipular las emulsiones. Las emulsiones generalmente se forman utilizando métodos físicos (por ejemplo, vórtice) que dependen de las estadísticas de Poisson para lograr la capacidad de clonación (Nakano et al. (2005) *J. Biosci. Bioeng.* 99:293; Williams et al. (2006) *Nat. Methods* 3:545). Sin embargo, esto tiende a conducir a una pequeña fracción de compartimentos no clonales, y también conduce a un gran número de compartimentos desocupados. Para combatir estos problemas, las emulsiones se generan utilizando tecnología de microfluidos (Clausell-Tormos et al., *Supra*). Una desventaja de usar métodos de emulsión es que una vez que se forma una emulsión, es difícil intercambiar material adicional con los compartimentos de una manera controlada. Sin embargo, se utiliza una tecnología para fusionar gotitas de emulsión de una manera controlada (Tewhey et al. (2009) *Nat. Biotechnol.* 27:1025; Meyers and Gelfand (1991) *Biochemistry* 30:7661) para abordar esta desventaja. Finalmente, la PCR en emulsión se realiza opcionalmente usando condiciones que están lejos de los protocolos estandarizados. Los métodos de aislamiento celular se resumen en la Figura 28.

20 Con estas consideraciones generales, se han identificado al menos seis estrategias para lograr un acoplamiento robusto de cadenas pesadas y livianas. Estas estrategias se representan en la Figura 29 e incluyen, pero no se limitan a: 1) PCR en emulsión a partir de ADN_g; 2) RT en célula y SOE-PCR o acoplamiento Cre-Lox; 3) RT-PCR en emulsión mediada por Tth; 4) RT-PCR en la célula encapsulada con acrilamida; 5) PCR en emulsión sobre perlas; y 6) emulsión RT-PCR con fusión de gotitas.

35 PCR en emulsión a partir de ADN_g

Este es el enfoque más simple basado en emulsión. Las células se colocan en la emulsión junto con los reactivos para una reacción de PCR tradicional. SOE-PCR luego se realiza utilizando el ADN_g como plantilla.

RT en la celda y SOE-PCR o acoplamiento Cre-Lox

40 Esto implica fijar células en formalina y permeabilizarlas usando uno de varios métodos (por ejemplo, proteinasa K) (Chapal et al., *Supra*; Embleton et al., *Supra*). Debido a que todas las células están en solución, se realiza una reacción de RT-PCR tradicional aplicando las enzimas relevantes en serie.

RT-PCR en emulsión mediada por Tth

45 Es deseable capturar las cadenas de Ig de la secuencia del ARNm, ya que esto evita cualquier reorganización del receptor no funcional y también se beneficia de los números de copia potencialmente más altos de las células expresadas. Sin embargo, la PCR en emulsión solo permite la adición de reactivos bioquímicos una vez. Por lo tanto, se usa la Tth polimerasa, que es capaz de realizar tanto RT como PCR (Myers and Gelfand, *supra*). El rendimiento de la polimerasa se caracteriza y se utiliza en el contexto de las emulsiones.

RT-PCR en la célula encapsulada con acrilamida

50 Esta técnica es similar a la RT-PCR en la célula, pero implica el paso adicional de encapsular células individuales en geles de poliacrilamida (Yokoyama et al. (1990) *Jinrui Idengaku Zasshi* 35:131), que agrega una capa adicional de Protección contra la contaminación cruzada.

PCR en emulsión en perlas

55 Como alternativa a SOE-PCR para entrecruzar las cadenas pesadas y livianas, las dos cadenas se capturan mediante PCR convencional en perlas (Diehl et al., *Supra*). Cada compartimento de emulsión pretende tener una sola célula y una única cuenta. Las perlas están recubiertas con dos cebadores diferentes: uno para la cadena pesada y otro para

la cadena liviana. Después de romper la emulsión, las perlas se manipulan opcionalmente de varias maneras. Un método, que puede aumentar la especificidad de la técnica, es entrecruzar las dos cadenas en las perlas utilizando un sistema Cre-Lox modificado.

Emulsión RT-PCR con fusión de gotitas

- 5 Como alternativa al uso de la polimerasa Tth, las gotitas de emulsión se fusionan de manera controlada. De esta manera, las "bolsas" de enzimas se pueden fusionar en serie con los compartimentos de emulsión para realizar pasos bioquímicos separados, como RT seguido de PCR. Este enfoque permite el uso de enzimas robustas y bien caracterizadas para realizar por separado RT y PCR.

Diseño de cebadores y vectores para acoplamiento múltiple de cadenas

- 10 Los cebadores están diseñados para los diversos protocolos de RT y PCR descritos en este documento. Los cebadores incluyen, pero no se limitan a: cebadores RT; multiplex PCR cebadores para los segmentos V y J para ADNc (los cebadores V son específicos para ADNc ya que abarcarán el límite del exón L1-L2 para la secuencia líder); cebadores de PCR multiplex para los segmentos V y J para ADNg (estos se utilizan durante la PCR en emulsión);
 15 cebadores de PCR que son inmovilizados en perlas; cebadores de PCR que contienen etiquetas complementarias para la PCR de solapamiento (es importante que la superposición sea adecuada como secuencia de enlace para una cadena scFv y coloque las cadenas pesada y liviana en el marco); cebadores de PCR que contienen sitios LoxP para entrecruzar cadenas pesadas y livianas (Chapal et al., supra); y cebadores que incorporan secuencias comunes en cebadores para adaptar fácilmente las secuencias a la secuenciación de la próxima generación y para clonar fácilmente secuencias en varios vectores de expresión (es decir, para varias tecnologías de visualización de proteínas). Además, los vectores están diseñados para ser adecuados para una variedad de tecnologías de visualización, utilizando vectores estándar mediante el uso de protocolos de clonación de Gateway.

Optimización de las condiciones de PCR para el entrecruzamiento de PCR

- 25 Las condiciones de reacción de PCR se optimizan en varias etapas. El primer ADNg o ARNm preparado a partir de poblaciones clonales de células se utiliza para asegurar que las cadenas pesada y liviana estén reticuladas en un entorno idealizado. Una vez que se logran condiciones de PCR sólidas, las cadenas de Ig (de células enteras) se entrecruzan en soluciones utilizando SOE-PCR o recombinación de Cre-Lox. La tecnología de microfluidos se utiliza para colocar células individuales en compartimentos de emulsión de 100 µm de diámetro (aproximadamente 0.5 nL) (Clausell-Tormos et al., Supra). En ciertas realizaciones, el tamaño del compartimento se reduce para asegurar que la emulsión no se rompa durante el ciclo térmico. Se puede usarse una concentración celular equivalente a
 30 compartimientos de emulsión de 20 µm, que corresponderían a aproximadamente 1.5×10^6 células en una reacción típica de 50 µL.

Optimización del protocolo de aislamiento celular

- 35 Las cadenas pesadas y livianas están reticuladas en celdas individuales. Usando una única línea celular monoclonal, se realizan múltiples métodos para aislar células individuales y realizar PCR. Después de la fijación celular con formalina, se realiza la permeabilización de la célula (los detergentes como Nonidet P40 (Embleton et al., Supra) pueden ser demasiado agresivos para tal uso). Las digestiones de proteinasa K se pueden usar para permeabilizar las células al masticar proteínas de membrana y poros (Bagasra (2007) Nat. Protocol. 2:2782). Las etiquetas superpuestas se utilizan para entrecruzar los dos amplicones durante la PCR. Un método alternativo implica la incorporación de sitios LoxP en los dos cebadores internos y el entrecruzamiento de las dos cadenas mediante
 40 recombinación de Cre (Chapal et al., Supra).

Caracterización de la contaminación cruzada en el entrecruzamiento de cadenas.

- Tras el exitoso entrecruzamiento de cadenas pesadas y livianas en células individuales, se mide el nivel de contaminación cruzada. El estudio de Embleton et al. encontró que al mezclar dos líneas celulares de hibridoma conocidas en proporciones 1:1 y 9:1, no observaron amplicones mal pareados (Embleton et al., supra). Sin embargo,
 45 estas relaciones son demasiado indulgentes en comparación con las relaciones reales efectivas de mezclas complejas. En consecuencia, se caracteriza el nivel de emparejamiento erróneo utilizando dos líneas celulares en proporciones de hasta $10^6:1$. Además, el método utilizado en Embleton et al. se basó en la captura de clones de la línea celular "rara" utilizando métodos de detección tradicionales. Estos métodos no son prácticos cuando se intentan en proporciones de células más estrictas, ya que el tipo de célula rara nunca se puede observar. Para abordar esto,
 50 se proporcionan métodos basados en PCR que son mucho más sensibles que el método utilizado en Embleton et al.

Ensayo de contaminación cruzada basado en PCR

- Se usa un ensayo TaqMan para cuantificar el nivel de contaminación cruzada. Sin pretender estar limitados por la teoría científica, después de realizar la PCR de entrecruzamiento unicelular, se esperan cuatro especies en la mezcla combinada: A-a, A-b, B-a, B-b (donde A y B son los dos clones y A contra a es cadena pesada versus liviana). Como
 55 la secuencia del enlazador scFv que conecta las cadenas pesada y liviana es la misma para todas las especies, se utiliza una sonda TaqMan con fluoróforo X para hibridar con esta porción común. Por separado, se utilizan cuatro

5 construcciones de la misma longitud que se amplifican solo por una de las cuatro combinaciones de cebadores, que contienen una secuencia única común que se hibrida con una segunda sonda TaqMan con fluoróforo Y. Las construcciones sintéticas se ajustan a concentraciones conocidas. Se utiliza una reacción de PCR en tiempo real en la que la mezcla experimental de cuatro construcciones se divide en cuatro tubos separados. Cada tubo se amplifica luego con una de las cuatro combinaciones de cebadores junto con la plantilla sintética correspondiente a una concentración conocida. Cada tubo también se amplifica en presencia de ambas sondas TaqMan. La fluorescencia se mide en ambas longitudes de onda en cada ciclo. En última instancia, las mediciones de fluorescencia de Y se utilizan como estándar para comparar las concentraciones relativas de las diversas especies en la mezcla de partida.

Quantificación de la contaminación cruzada.

10 Después de un emparejamiento incorrecto basado en PCR, se miden los niveles de contaminación cruzada en diferentes proporciones de las dos líneas celulares. Para los métodos de PCR en la célula, una fuente de contaminación cruzada es el ARNm/ADNc que se filtró en la solución sobrenadante. Para combatir este problema, se analizan diferentes parámetros de permeabilización, por ejemplo, como la modificación de la digestión con proteinasa K.

15 Para los métodos basados en emulsión, una fuente de contaminación cruzada es la existencia de compartimentos no clonales. Esto puede ocurrir durante la formación de la emulsión cuando se colocan múltiples células en un solo compartimento, así como durante el ciclo térmico de la emulsión, donde diferentes compartimentos de emulsión se pueden fusionar. Para evitar la no capacidad de clonación en la etapa de formación de la emulsión, se proporcionan plataformas de microfluidos para determinar los métodos de formación de la emulsión que son más fáciles de controlar.
20 Se proporciona una variedad de diferentes aceites de emulsión y/o surfactantes y/o compartimentos de emulsión de tamaño reducido para evitar la fusión de gotitas durante el ciclo térmico.

El nivel de contaminación cruzada se obtiene utilizando mezclas de dos líneas celulares conocidas (A y B) en relaciones A:B que varían de 1:1 a 10^6 :1. A medida que aumenta la concentración de A en relación con B, aumenta la probabilidad de que una cadena B se empareje mal con una cadena A, lo que lleva a una mayor cantidad relativa de cadenas B mal apareadas. Dada una compleja mezcla de linfocitos obtenidos de la sangre, se determina el nivel de frecuencia más bajo para el cual se determina el emparejamiento de cadenas pesadas y livianas.
25

Creación de bibliotecas scFv a partir de muestras complejas.

Las muestras de linfocitos derivadas de sangre complejas se utilizan para generar nuevas bibliotecas de secuenciación 454. Algunas de las muestras de vacunación recogidas se secuencian para confirmar la reproducibilidad.

30 Análisis de las propiedades del emparejamiento de cadenas, incluidas las estimaciones de diversidad total

Los datos scFv completos recién generados se ejecutan a través de una canalización informática para determinar el uso de VDJ y dividir los datos en clones únicos. Estos datos proporcionan las estimaciones más precisas hasta la fecha de las diversidades de anticuerpos expresadas. Además, muchos estudios suponen que el emparejamiento de cadenas pesadas y livianas es independiente y aleatorio. Sin embargo, sin pretender estar limitado por la teoría científica, se cree que la generación de cadenas pesadas y livianas emparejadas reales mostrará que la distribución está lejos de ser aleatoria y proporciona métodos novedosos para caracterizarla.
35

Caracterización de múltiples soluciones a la infección por influenza H1N1

Se proporciona un nuevo protocolo para recolectar muestras de sangre de voluntarios que serán inmunizados contra la cepa de influenza H1N1 ("gripe porcina"). El protocolo selecciona voluntarios que no han estado expuestos a la gripe porcina y también voluntarios que contrajeron la enfermedad y la combatieron con éxito. Las respuestas inmunes de las personas que han estado expuestas al virus pueden compararse con las personas inmunes a aquellas que no han estado expuestas al virus (aunque probablemente hayan contraído alguna otra cepa de influenza en algún momento de su historia).
40

Además, cada muestra obtenida de un sujeto que combatió con éxito la infección por H1N1 contiene una solución inmunológica única para combatir el virus. Se obtienen preparaciones de proteínas disponibles comercialmente de hemaglutinina H1N1 glicosilada y neuraminidasa, y las bibliotecas de scFv expresadas de cada individuo se clonan utilizando un sistema de visualización in vitro (por ejemplo, presentación de fagos). Los anticuerpos contra la gripe se enriquecen funcionalmente utilizando técnicas de exploración estándar, y los grupos de scFv enriquecidos resultantes se secuencian utilizando una tecnología de secuenciación de próxima generación. Esto permite la observación directa de cómo los diferentes individuos combaten el H1N1, y también confirma los resultados obtenidos al secuenciar el repertorio sin tener que desplazarse contra el antígeno.
45
50

Además, las muestras se guardan para expresar los supuestos anticuerpos de respuesta y caracterizar su interacción contra bibliotecas de antígenos más grandes.

Ejemplo III

Caracterización de alto rendimiento y selección de repertorios de anticuerpos contra bibliotecas de patógenos

Se utiliza un enfoque similar al descrito anteriormente para caracterizar las interacciones antígeno-anticuerpo en el alto rendimiento. Las tecnologías actuales requieren la selección de nuevos anticuerpos contra un solo antígeno a la vez (Carter (2006) *Nat. Rev. Immunol.* 6:343). Un experimento típico consiste en purificar e inmovilizar un antígeno de interés y exponerlo a algún tipo de tecnología de visualización de proteínas que codifica una biblioteca de secuencias de anticuerpos candidatos (normalmente en formato scFv). Después de varias rondas de selección, lavado y amplificación, se secuencian una pequeña cantidad de clones y se lleva a cabo para su posterior análisis. Alternativamente, un animal se inmuniza con el antígeno de interés para generar una respuesta policlonal. Luego se recolectan linfocitos específicos de antígeno (para obtener la respuesta policlonal) y se analizan para obtener anticuerpos monoclonales de alta afinidad. Id. Se proporcionan métodos que permiten la selección de nuevos anticuerpos contra múltiples antígenos en paralelo.

Dos cadenas de ácidos nucleicos se acoplan entre sí. Los anticuerpos pueden codificarse en un formato de presentación de proteínas (por ejemplo, Fagos (Clackson et al. (1991) *Nature* 352:624; McCafferty et al. (1990) *Nature* 348:552), levadura (Boder and Wittrup (1997) *Nat. Biotechnol.* 15:553; Boder et al. (2000) *Proc. Natl. Acad. Sci. USA* 97:10701; Chao et al. (2006) *Nat. Protoc.* 1:755), o presentación de ribosomas (Zahnd et al. (2007) *Nat. Methods* 4:269)) y los antígenos se empaquetan con su información de codificación (por ejemplo, Bibliotecas de péptidos en formato de visualización o partículas de virus completas). Se permite que las dos bibliotecas interactúen y se seleccionan complejos interactivos (Bowley et al. (2009) *Proc. Natl. Acad. Sci. USA* 106:1380). Las cadenas de codificación del antígeno y el anticuerpo que interactúan se acoplan físicamente utilizando métodos similares a los descritos en este documento. Esto permitirá la caracterización de anticuerpos específicos de antígeno para múltiples antígenos en paralelo. En última instancia, este tipo de enfoque se ampliará para analizar bibliotecas de antígenos muy grandes contra bibliotecas de antígenos muy grandes (por ejemplo, proteoma humano completo, todas las proteínas virales conocidas y similares).

Un aspecto único de los métodos descritos en este documento es que los protocolos para capturar anticuerpos completos ya habrán formateado los repertorios capturados en un formato scFv, permitiendo una expresión fácil usando una de las tecnologías de visualización de proteínas. En comparación con muchos estudios previos que generan bibliotecas de anticuerpos al azar, cada anticuerpo en las bibliotecas descritas en este documento se genera por un sistema inmune natural que funciona. De esta manera, se pueden usar millones de años de evolución para proporcionar bibliotecas de anticuerpos eficientes que permitirán a un experto en la técnica la capacidad de descubrir rápidamente nuevos anticuerpos funcionales.

Métodos para selecciones de afinidad multiplex

Un enfoque para capturar las interacciones antígeno-anticuerpo (Ag-Ab) implica colocar complejos de Ag-Ab individuales en compartimientos de emulsión individuales. Con el fin de mostrar correctamente los anticuerpos y antígenos, varias opciones son adecuadas, entre ellas, la presentación de fagos, levaduras o ribosomas, partículas virales completas (en el caso de antígenos virales). La visualización de ribosomas ofrece las mayores bibliotecas potenciales y está completamente *in vitro*. La visualización de fagos es el método más antiguo y proporciona bibliotecas de gran tamaño. La visualización de levadura ofrece los tamaños de biblioteca más pequeños, pero es particularmente adecuada para bibliotecas de antígenos, ya que puede transportar cargas útiles más grandes y tiene maquinaria de glicosilación (Hoogenboom (2005) *Nat. Biotechnol.* 23:1105). El uso de partículas de virus completas proporciona acceso a los antígenos más realistas, pero tiene un genoma potencialmente grande y otras preocupaciones de seguridad adicionales para el trabajo de laboratorio. Es importante tener en cuenta que debido a que los repertorios expresados naturalmente se clonan, se espera que el tamaño máximo de biblioteca de todas estas tecnologías sea suficiente para capturar la diversidad de secuencias en cualquier muestra de sangre práctica o conjuntos de antígenos. Phage-Ab contra levadura-Ag es un enfoque general prometedor para las selecciones de biblioteca contra biblioteca (Bowley et al., *Supra*). La levadura es adecuada para fragmentos de proteínas más grandes, los dos sistemas pueden replicarse de manera independiente. Además, el ribosoma-Ab contra el ribosoma-Ag también se utiliza, ya que este sistema es totalmente *in vitro*. Esto permitirá métodos de acoplamiento que no dependen de las emulsiones. Finalmente, estos métodos también se usan contra partículas virales enteras.

Para los sistemas basados en fagos, se utiliza un método basado en emulsión para capturar anticuerpos y antígenos que interactúan. Después de varias o más rondas de selección de afinidad, los dúplex de fago-levadura-Ag se purifican dos veces usando clasificación magnética o FACS para eliminar las partículas que no interactúan. Id. Estos dúplex se colocan luego en compartimientos de emulsión individuales donde una reacción de PCR de entrecruzamiento asociará físicamente la secuencia codificadora de Ag y Ab. Estas especies reticuladas se preparan para la secuenciación de la próxima generación y las interacciones se detectan mediante el análisis de los datos de secuenciación. Cuando el Ag utilizado es partículas virales completas, la secuencia del genoma puede ser relativamente grande. Sin embargo, como se analiza un número relativamente pequeño de virus únicos, el método es suficiente para encontrar un código de barras único para identificar la cepa específica del virus.

Para el sistema de visualización de ribosomas completamente *in vitro*, además de intentar el protocolo basado en emulsión, se utilizan métodos de no emulsión. Se utiliza un ensayo de ligamiento de proximidad (Soderberg et al.

(2006) Nat. Methods 3:995) para capturar etiquetas únicas en las bibliotecas de Ag y Ab. Estas etiquetas proporcionarán suficiente información para obtener las secuencias correspondientes completas.

Ensayo de un control positivo para la interacción Ab-Ag

5 Un solo par de Ag-Ab que se sabe que interactúa con alta afinidad y especificidad está reticulado. Un fragmento scFv se crea en un formato de presentación de fagos a partir de una secuencia que se sabe que se une a la neuraminidasa de la gripe, junto con el fragmento de antígeno correspondiente en un formato de presentación de levadura. Se permite que las preparaciones de las dos especies interactúen en solución, y los métodos descritos aquí se usan para entrecruzar las dos cadenas de codificación.

10 En un primer método, solo el anticuerpo de la gripe y el antígeno de la gripe pueden interactuar en la solución, y la PCR se utiliza para entrecruzar las dos cadenas codificadoras. Los cebadores se ensayan para el entrecruzamiento mediante PCR solapada y/o recombinación Cre/lox. Del mismo modo, los mismos pares de control positivo Ab y Ag se colocan en un sistema de visualización de ribosomas comercial en donde se permite que el par interactúe. La transcripción reversa y la PCR de entrecruzamiento se realizan utilizando los métodos descritos en este documento para analizar la fusión exitosa de las cadenas codificadoras de Ab y Ag.

15 Cuantificación del enriquecimiento de doble purificación para la interacción de pares Ab-Ag

Además del control positivo par Ab-Ag, se elige un segundo par ortogonal Ab-Ag, junto con un fago-Ab que no es específico para ningún antígeno usado y un levadura-Ag que no es específico para ninguno de los anticuerpos. Después de permitir que la mezcla interactúe, los complejos de Ab-Ag afines se purifican mediante citometría de flujo de células de levadura utilizando un anticuerpo anti-fago fluorescente, como se describe en (Bowley et al., Supra). A partir de las células clasificadas, se realizan el entrecruzamiento y la PCR y se mide la cantidad de ADN contaminante del fago y la levadura que se espera que no muestre afinidad de unión. Con el fin de reducir la cantidad de fondo, se realiza una purificación doble seguida de la amplificación de las bibliotecas de fagos y levaduras por separado. Id. Para el sistema de visualización de ribosomas, dos etiquetas de afinidad diferentes se codifican en la biblioteca Ag y la biblioteca Ab (por ejemplo, las etiquetas His y flag). Luego se realiza la purificación de la etiqueta de His seguido de la purificación de la etiqueta de bandera para obtener partículas que interactúan.

25 Cuantificación del aislamiento de complejos Ab-Ag en emulsiones.

Después de realizar la selección para partículas que interactúan, los complejos Ab-Ag se colocan en emulsiones usando los métodos descritos en este documento. En paralelo, se ensayan progresivamente las relaciones más extremas de los dos pares de control positivo ortogonal para determinar la cantidad de contaminación cruzada durante la fusión de la cadena de codificación. En la Figura 30 se presenta un resumen.

30 Controles iniciales para la ligación de proximidad del ribosoma

Una ventaja de los métodos de presentación de ribosomas descritos en este documento es que todo el sistema está in vitro y las secuencias de codificación son expuestas a una solución. Usando solo uno de los pares Ab-Ag de control positivo, se ensaya el sistema de ligamiento de proximidad basado en círculo rodante (Soderberg et al., Supra). Las dos bibliotecas contienen secuencias únicas que funcionan como códigos_barras. Se añaden sondas adicionales a la reacción que forman círculos cerrados cuando las dos cadenas de los ARNm de Ab y Ag están muy cerca entre sí.

35 El mismo ensayo se realiza luego con dos controles positivos después de la doble purificación para complejos que interactúan. Esto aclarará si el ensayo de ligamiento de proximidad es específico solo para los pares Ab-Ag que realmente interactúan, o si los círculos pueden cerrarse de forma promiscua.

40 Creación de una tabla de búsqueda Ab-Ag utilizando bibliotecas de visualización virales

Después de vincular con éxito las cadenas de codificación de pares Ab-Ag interactivos de un pequeño número de controles, el método se aplica a bibliotecas complejas. Se crean bibliotecas de presentación de fagos y visualización de ribosomas de scFv expresados naturalmente que se clonaron como se describe en este documento. Para las muestras de vacunación, se genera una pequeña biblioteca de proteínas virales que corresponden a las mismas cepas a las que los sujetos fueron expuestos. Estas bibliotecas se utilizan para caracterizar qué anticuerpos se unen a qué proteínas. Los resultados obtenidos se correlacionan con los clones de anticuerpos que se identificaron para reaccionar a las vacunas administradas en los pacientes.

45 Además de generar bibliotecas de proteínas virales artificialmente, se utiliza un método similar con partículas virales completas. Los virus son un sistema conveniente para esta estrategia, ya que funcionan como sus propias partículas de exhibición de proteínas. Además, dado que solo se necesita un pequeño número de cepas virales, es suficiente para identificar qué cepa ha sido capturada mediante el entrecruzamiento con una o más secuencias genómicas de códigos_barras únicas.

Ejemplo IV

Materiales y métodos

Muestras biológicas e inmunización

Se extrajeron muestras de sangre periférica en tubos de 9 mL de K3-EDTA de un solo individuo 14 días antes y 0, 1, 3, 7, 14, 21 y 28 días después de la vacunación. Las vacunas administradas fueron las siguientes: Hepatitis A+B (Twinrix) y vacuna contra la influenza de temporada, incluidas las cepas A/Brisbane/59/2007 (similar a H1N1), A/Brisbane/10/2007 (similar a H3N2) y B/Florida/4/2006. Extracción de ARN

Las muestras de sangre se procesaron inmediatamente a través de una unidad de filtrado para aislar la fracción de leucocitos (LeukoLOCK™, Ambion, Austin, TX) y luego se almacenaron a -80°C hasta que se obtuvieron todas las muestras. El ARN total se extrajo posteriormente de acuerdo con el protocolo del fabricante. La integridad y la concentración de ARN se evaluaron utilizando un espectrofotómetro Nanodrop (Thermo Scientific, Wilmington, DE) y un Bionanalizador 2100 (Agilent, Foster City, CA).

Síntesis de ADNc y PCR

Para cada muestra, 2 x 1 µg de ARN total se transcribieron de manera inversa en 2 reacciones independientes de 20 µL utilizando la transcriptasa inversa SuperScript™ III (Invitrogen, Carlsbad, CA) a 55°C durante 60 minutos en presencia de 5 pmoles del gen de la región C mezcla de cebadores específicos (IGHC-RT) (Tabla 1), seguida de inactivación enzimática y digestión con RNasa H (Epicenter Biotechnologies, Madison, WI) de acuerdo con los protocolos de Invitrogen. Cada reacción de ADNc de 20 µL se dividió luego en 4 reacciones de PCR de 50 µL cada una, para un total de 8 reacciones de PCR independientes por muestra originaria. La PCR se realizó en presencia de dNTP 200 nM (Enzymatics, Beverly, MA), 25 pmoles de la mezcla de cebadores específicos del gen de la región V (IGHV-PCR), 25 pmoles de la mezcla de cebadores específicos del gen de la región C (IGHC-PCR), y 1 U de ADN de polimerasa de inicio de alta fidelidad Phusion™ (New England Biolabs, Ipswich, MA) y termociclado de la siguiente manera: 98°C durante 1 min, 16 ciclos de 98°C durante 10 segundos, 62°C durante 20 segundos, 72°C durante 20 segundos, seguido de 72°C durante 5 minutos y luego se enfrió a 4°C. Después de la PCR, todas las reacciones que se originaron en la misma muestra se combinaron en una mezcla de 400 µL a la que se agregaron 160 U de exonucleasa I (Epicenter Biotechnologies, Madison, WI) y se incubaron a 37°C durante 45 min. Las muestras se purificaron luego utilizando el kit de purificación de PCR QIAquick (Qiagen, Valencia, CA) y se eluyeron en 36 µl de regulador EB de acuerdo con el protocolo del fabricante.

Tabla 1

Oligonucleótidos de transcripción reversa

Mezcla de cebadores específicos de gen de la región C-(IGHC-RT)

VDJ-20080707-IGHG	AGGGYGCCAGGGGGAAGA (SEQ ID NO:133)
VDJ-20080707-IGHM	GGAGACGAGGGGGAAAAGG (SEQ ID NO:134)
VDJ-20080707-IGHA	CAGCGGGAAGACCTTGGG (SEQ ID NO:135)
VDJ-20080707-IGHD	CACATCCGGAGCCTTGGT (SEQ ID NO:136)
VDJ-20080707-IGHE	TCAAGGGGAAGACGGATGG (SEQ ID NO:137)

Oligonucleótidos de PCR

Cebadores específicos del gen de la región C (IGHC-PCR)

VDJ-20080924-IGHG-1	CCGATGGGCCCTTGGTGG (SEQ ID NO:138)
VDJ-20080924-IGHG-2	CGGATGGGCCCTTGGTGG (SEQ ID NO:139)
VDJ-20080924-IGHM	GGGTTGGGGCGGATGCAC (SEQ ID NO:140)
VDJ-20080924-IGHA	CCTTGGGGCTGGTCGGGG (SEQ ID NO:141)
VDJ-20080924-IGHD	CATCCGGAGCCTTGGTGG (SEQ ID NO:142)

ES 2 730 951 T3

Oligonucleótidos de transcripción reversa

Mezcla de cebadores específicos de gen de la región C-(IGHC-RT)

VDJ-20080924-IGHE CGGATGGGCTCTGTGTGG (SEQ ID NO:143)

Cebadores específicos del gen de la región V-(IGHV-PCR)

VDJ-20080924-IGHV1-1 GAGCAGCGACAGGTGCCC (SEQ ID NO:144)

VDJ-20080924-IGHV1-2 CAGCAGCCACAGGTGCCC (SEQ ID NO:145)

VDJ-20080924-IGHV1-3 GCAGCAGCTACAGGTGTCC (SEQ ID NO:146)

VDJ-20080924-IGHV1-4 CTGTAGCACCAGGTGCCC (SEQ ID NO:147)

VDJ-20080924-IGHV1-5 GCTGTAGCTCCAGGTGCTC (SEQ ID NO:148)

VDJ-20080924-IGHV1-6 CAGCACCAACAGGTGCCC (SEQ ID NO:149)

VDJ-20080924-IGHV1-7 CAGCAGCCACAGNTGCCT (SEQ ID NO:150)

VDJ-20080924-IGHV1-8 CAGCAGCTACAAGTGCCC (SEQ ID NO:151)

VDJ-20080924-IGHV1-9 CAGCAGCCACAGGAGCCC (SEQ ID NO:152)

VDJ-20080924-IGHV1-10 CAGCAGCCACAGGTGTCC (SEQ ID NO:153)

VDJ-20080924-IGHV1-11 CAGCAGCTACAGGCACCC (SEQ ID NO:154)

VDJ-20080924-IGHV1-12 CCTGTTTTTGGTGCCC (SEQ ID NO:155)

VDJ-20080924-IGHV1-13 TGGCAGCACCAGGCGCCC (SEQ ID NO:156)

VDJ-20080924-IGHV1-14 TCATAGCTGCAGGTGCCC (SEQ ID NO:157)

VDJ-20080924-IGHV2-1 CTGTCCCGTCCTGGGTCT (SEQ ID NO:158)

VDJ-20080924-IGHV2-2 ACCATCCCTTCATGGGTCT (SEQ ID NO:159)

VDJ-20080924-IGHV2-3 CCACCCCTTCCTGGGTCT (SEQ ID NO:160)

VDJ-20080924-IGHV3-1 TTCTGTGCTATATTAAGGTGTCC (SEQ ID NO:161)

VDJ-20080924-IGHV3-2 TTGTTGCTATTTTAAAAGGTGTCC (SEQ ID NO:162)

VDJ-20080924-IGHV3-3 CGTTGCTCTTTTAAAGGTGTCC (SEQ ID NO:163)

VDJ-20080924-IGHV3-4 TTGTTGCTATTTTAAAGGTGTCC (SEQ ID NO:164)

VDJ-20080924-IGHV3-5 TTGTTGCTATATTAGAAGGTGTCC (SEQ ID NO:165)

VDJ-20080924-IGHV3-6 GCTATTTTAAAAGGTGTCC (SEQ ID NO:166)

ES 2 730 951 T3

VDJ-20080924-IGHV3-7	TTGTTGCTATTTTAGAAGGTGTCC (SEQ ID NO:167)
VDJ-20080924-IGHV3-8	TTGTGGCTATTTTAAAAGGTGTCC (SEQ ID NO:168)
VDJ-20080924-IGHV3-9	TTGTTGTTATTTTACAAGGTGTCC (SEQ ID NO:169)
VDJ-20080924-IGHV3-10	TTCCTGCTATTTTAAAAGGTGTCC (SEQ ID NO:170)
VDJ-20080924-IGHV3-11	TTGCTGCTATTTTAAAAGGTGTCC (SEQ ID NO:171)
VDJ-20080924-IGHV3-12	TTTTGGCTATTTTAAAAGGTGTCC (SEQ ID NO:172)
VDJ-20080924-IGHV3-13	TTGTGGCTAAAATAAAAAGGTGTCC (SEQ ID NO:173)
VDJ-20080924-IGHV3-14	TTGTTGCTATAATAAAAAGGTGTCC (SEQ ID NO:174)
VDJ-20080924-IGHV3-15	TTGCTGGTATTTTAAAAGGTGTCC (SEQ ID NO:175)
VDJ-20080924-IGHV3-16	TTGTTGGTATTTTAAAAGGTGTCC (SEQ ID NO:176)
VDJ-20080924-IGHV4-1	CAGCTCCCAGATGGGTCC (SEQ ID NO:177)
VDJ-20080924-IGHV4-2	CGGCTCCCAGATGGGTCC (SEQ ID NO:178)
VDJ-20080924-IGHV4-3	GCTCCCAGATGTGGGTCC (SEQ ID NO:179)
VDJ-20080924-IGHV5-1	GGCTGTTCTCCAAGGAGTCT (SEQ ID NO:180)
VDJ-20080924-IGHV5-2	CCTCCACAGTGAGTGAGTCT (SEQ ID NO:181)
VDJ-20080924-IGHV5-3	CCTAGCTATTCTCCAAGGAGTCT (SEQ ID NO:182)
VDJ-20080924-IGHV6-1	GCCTCCCATGGGGTGTCC (SEQ ID NO:183)
VDJ-20080924-IGHV6-2	GGGCCTCCATGGGTGTCC (SEQ ID NO:184)
VDJ-20080924-IGHV7-1	CAGCAGCAACAGGTGCCC (SEQ ID NO:185)
VDJ-20080924-IGHV7-2	GCAGCAGCAACAGGTACCT (SEQ ID NO:186)

454 bibliotecas de construcción de oligonucleótidos

Adaptador A de oligonucleótidos

Adaptador 454 A1_MID1	C*C*A*T*CTCATCCCTGCGTGTCTCCGACTCAGACGAGT*G*C*G* T (SEQ ID NO:187)
Adaptador 454 A2_MID2	C*C*A*T*CTCATCCCTGCGTGTCTCCGACTCAGACGCTC*G*A*C*A (SEQ ID NO:188)
Adaptador 454 A3_MID3	C*C*A*T*CTCATCCCTGCGTGTCTCCGACTCAGAGACGC*A*C*T*C (SEQ ID NO:189)

ES 2 730 951 T3

Adaptador 454 A4_MID4	C*C*A*T*CTCATCCCTGCGTGTCTCCGACTCAGAGCACT*G*T*A*G (SEQ ID NO:190)
Adaptador 454 A5_MID5	C*C*A*T*CTCATCCCTGCGTGTCTCCGACTCAGATCAGA*C*A*C*G (SEQ ID NO:191)
Adaptador 454 A6_MID6	C*C*A*T*CTCATCCCTGCGTGTCTCCGACTCAGATATCG*C*G*A*G (SEQ ID NO:192)
Adaptador 454 A8_MID8	C*C*A*T*CTCATCCCTGCGTGTCTCCGACTCAGCTCGCG*T*G*T*C (SEQ ID NO:193)
Adaptador 454 A9_MID9	C*C*A*T*CTCATCCCTGCGTGTCTCCGACTCAGTAGTAT*C*A*G*C (SEQ ID NO:194)
Adaptador 454 A10_MID10	C*C*A*T*CTCATCCCTGCGTGTCTCCGACTCAGTCTCTA*T*G*C*G (SEQ ID NO:195)
Adaptador Aprime1_MID1	454 A*C*G*C*ACTCGTCTGAGTCG*G*A*G*A (SEQ ID NO:196)
Adaptador Aprime2_MID2	454 T*G*T*C*GAGCGTCTGAGTCG*G*A*G*A (SEQ ID NO:197)
Adaptador Aprime3_MID3	454 G*A*G*T*GCGTCTCTGAGTCG*G*A*G*A (SEQ ID NO:198)
Adaptador Aprime4_MID4	454 C*T*A*C*AGTGCTCTGAGTCG*G*A*G*A (SEQ ID NO:199)
Adaptador Aprime5_MID5	454 C*G*T*G*TCTGATCTGAGTCG*G*A*G*A (SEQ ID NO:200)
Adaptador Aprime6_MID6	454 C*T*C*G*CGATATCTGAGTCG*G*A*G*A (SEQ ID NO:201)
Adaptador Aprime8_MID8	454 G*A*C*A*CGCGAGCTGAGTCG*G*A*G*A (SEQ ID NO:202)
Adaptador Aprime9_MID9	454 G*C*T*G*ATACTACTGAGTCG*G*A*G*A (SEQ ID NO:203)
Adaptador 454 Aprime10_MID10	C*G*C*A*TAGAGACTGAGTCG*G*A*G*A (SEQ ID NO:204)

Adaptador B de oligonucleótidos

Adaptador B 454	/5BioTEG/C*C*T*A*TCCCCTGTGTGCCTTGGCAGTC*T*C*A*G (SEQ ID NO:205)
Adaptador B 454 prime	C*T*G*A*GACT*G*C*C*A (SEQ ID NO:206)

Construcción y secuenciación de biblioteca 454 GS FLX

Las muestras de PCR de inmunoglobulina VH se procesaron siguiendo el protocolo de Preparación de la Biblioteca General Titanium 454 GS FLX. En resumen, las muestras se bloquearon de forma romaa y se fosforilaron en 5' utilizando el kit End-It™ DNA End-Repair (Epicenter Biotechnologies, Madison, WI) en reacciones de 50 µL de acuerdo con las instrucciones del fabricante, seguidas de QIAquick PCR Purification (Qiagen, Valencia, CA) en la presencia de

5 μ L de NaOAc 3M, pH 5.5. Los adaptadores de ADN compatibles con Roche 454 que contenían un código de barras de identificador molecular único (MID) para cada muestra se ligaron en un volumen de reacción de 100 μ L en presencia de 30 pmoles de ambos, adaptador A (incluido MID 1 a 8) y adaptador B (idéntico para todas las muestras) y 1200 U de ADN ligasa T4 rápida (Enzymatics, Beverly, MA) a 22°C durante 30 min. Para eliminar los adaptadores no ligados, las muestras se purificaron utilizando una relación 1:1 de perlas AMPure XP (Agencourt Bioscience Corporation, Beverly MA) y se eluyeron en 50 μ L de Tris-HCL 10 mM. Los extremos de los fragmentos ligados de los adaptadores de biblioteca se rellenaron en reacciones de 100 μ L utilizando 32 U de ADN polimerasa Bst (NEB, Ipswich, MA) en presencia de dNTP 625 μ M y regulador de reacción ThermoPol 1X. Las reacciones se purificaron utilizando el kit de purificación por PCR MinElute (Qiagen, Valencia, CA) y se eluyeron en 15 μ L de regulador EB. Las muestras se separaron en Novex® TBE Gels al 6% (Invitrogen, Carlsbad, CA) a 200 voltios durante 26 minutos, de los cuales se extrajo la banda adecuada (aproximadamente 450 pb) a 50°C durante 1 hora y se resuspendieron en 20 μ L de 10 Tris-HCL mM como se detalla anteriormente (Vigneault et al. (2008) Nat. Methods 5:777). Se encontró que todas las etapas relacionadas con el aislamiento de la biblioteca de ADN monocatenario (ADNsst) de la plantilla no eran necesarias para la posterior PCR en emulsión y la secuenciación 454. Las concentraciones de la biblioteca se determinaron utilizando un Bionanalyzer 2100 (Agilent, Foster City, CA), luego se diluyeron a concentraciones equivalentes. Las 8 muestras de sangre, cada una representando un punto temporal único y caracterizadas por un código de barras MID único, se agruparon a una concentración equimolar. La PCR en emulsión, el enriquecimiento de las perlas y la secuenciación de 454 GS FLX se realizaron en las instalaciones de 454 Life Sciences de acuerdo con el protocolo estándar del fabricante. Este proceso completo (incluida la síntesis de ADNc y la PCR) se repitió de forma independiente varias veces, o específicamente en una sola muestra, con el fin de generar repeticiones técnicas completas para los diversos análisis del estudio.

Flujo informático

Los datos de secuenciación sin procesar se analizaron con la tubería de 454 amplicones para producir lecturas de calidad recortada. Las lecturas se filtraron por tamaño para el tamaño de amplicón apropiado (350-500 pb) y se convirtieron en cadenas de codificación comparando las composiciones de k-mero de las lecturas de consulta con el IMG/GENE-DB (Giudicelli et al. 2005 Nucleic Acids Res. 33:D256) base de datos y su complemento inverso. Luego, las lecturas se alinearon con la base de datos de referencia IMG/GENE-DB para determinar el uso de V, D y J, y extraer la región de unión CDR3. En resumen, las composiciones k-mero de secuencias de consulta se compararon con los segmentos de referencia para encontrar las mejores coincidencias, y la mejor coincidencia se eligió utilizando la alineación de programación dinámica tradicional. Luego, las lecturas se dividieron mediante el uso de V-J, y sus uniones CDR3 se agruparon utilizando un agrupamiento jerárquico aglomerado de enlace simple o de enlace completo con la distancia de edición de Levenshtein como métrica. Los clones se definieron cortando el árbol de enlace en 4.5 ediciones. Al dividir las lecturas en los puntos de tiempo separados (identificando su código de barras MID), se definió una serie de tiempos para cada clon. Las series de tiempo se agruparon utilizando la agrupación k-medias con k=100 y la métrica de distancia euclidiana. Para eliminar los clones triviales que aparecieron en puntos de tiempo únicos, se filtraron los clones que mostraban recuentos positivos en al menos dos puntos de tiempo. Las series de tiempo también se agruparon usando mapas autoorganizados (SOM) implementados en el paquete de software GEDI. El software de análisis descrito en este documento está disponible para su descarga como un módulo de Python (Sitio web mundial: arep.med.harvard.edu/vdj).

40 Diseño de cebador

Todos los oligonucleótidos se adquirieron de Integrated DNA Technologies (IDT, Coralville, IA). Para el diseño de los oligonucleótidos de la región variable corriente arriba (IGHV-PCR), las secuencias L-PART1 y L-PART2 se extrajeron de todos los segmentos de referencia IMG/GENE-DB³ anotados como "funcionales" u "ORF". Estos dos segmentos se unen in vivo para formar la secuencia líder. La secuencia del cebador se colocó para cruzar el límite exón-exón para asegurar la amplificación de ADNc en lugar de ADNg. Los cebadores se dispusieron para tener 6 nucleótidos 3' del límite del exón, y variaron de 18-24 nucleótidos en longitud para apuntar sus temperaturas de fusión predichas a 60°C. Todas las secuencias duplicadas fueron eliminadas. Para el diseño de los oligonucleótidos de la región constante en sentido descendente (IGHC-RT e IGHc-PCR), los primeros 100 nucleótidos del exón CH1 se extrajeron de la IMG/GENE-DB³. Los oligonucleótidos se seleccionaron lo más cerca posible del extremo 5' de la región C (cerca de la región variable del receptor), comenzando con 3 nt desde el extremo. Estos se seleccionaron a mano tanto para IGHc-RT como para IGHc-PCR para aprovechar la conservación de la secuencia entre diferentes variantes y para asegurar que los isotipos fueran distinguibles. Todas las secuencias de oligonucleótidos se muestran en la Tabla 1.

Biblioteca de control

55 Con el fin de evaluar el sesgo de PCR y la eficiencia de amplificación, se extrajo el ARN total de múltiples individuos, se agruparon y se transcribieron de forma inversa como se describe en el Resumen de métodos. Se llevaron a cabo reacciones de PCR independientes para cada cebador específico de gen de la región C (IGHc-PCR) durante 25 ciclos, de los cuales se extrajo en gel la banda apropiada de interés como se describió anteriormente (Vigneault et al., Supra). Para monitorizar la eficiencia de amplificación de los cebadores de la región constante, los amplicones de PCR individuales se cuantificaron utilizando el Bionanalyzer 2100 (Agilent, Foster City, CA), seguido de una PCR en tiempo real realizada por triplicado utilizando el kit KAPA SYBR® FAST qPCR (Kapa Biosystems, Woburn, MA) en un

instrumento de PCR en tiempo real Bio-Rad CFX96 (Bio-Rad, Hercules, CA). Tanto la PCR de regresión lineal (LinRegPCR) (Ramakers et al. (2003) Neurosci. Lett. 339: 62) como el análisis de la curva de dilución estándar se utilizaron para determinar y comparar las eficiencias de amplificación (Tabla 2).

Tabla 2.

Cebador individual/reacción	Eficacia media de LinRegPCR en concentración de plantilla uniforme (triplicado de 1/125)	Eficacia media de LinRegPCR en series de curvas de ETS (series de dilución 1/5)	Eficacia media de la curva STD (series de dilución 1/10)	Eficiencia media
VDJ-20080924-IGHG-1	1.774	1.650	1.803	1.742
VDJ-20080924-IGHG-2	1.784	1.740	1.824	1.783
VDJ-20080924-IGHM	1.729	1.757	1.766	1.751
VDJ-20080924-IGHA	1.777	1.731	1.782	1.763
VDJ-20080924-IGHD	1.779	1.726	1.755	1.753
VDJ-20080924-IGHE	1.607	1.541	1.831	1.660

5

Identificación de la hebra positiva de 454 lecturas de secuenciación

Después de filtración por tamaño de las lecturas para el tamaño V_H esperado (Figura 18), se identificaron cadenas codificadoras, ya que era importante para la tubería de alineación posterior. Para ello, los k-meros se calcularon previamente utilizando el peine Positive Strand (Tabla 3) para toda la base de datos del segmento de referencia y por separado para su complemento inverso. Luego se calcularon los mismos k-meros para la lectura de la consulta, y la cadena de codificación se determinó calculando el número de k-meros compartidos de la versión positiva frente a la versión negativa de la base de datos.

10

Tabla 3.

Peine	Patrón
Hebra positiva	111111111111
Alineación A	111011001011010111
Alineación B	1111000100010011010111
Alineación C	111111111111
Alineación D	110100001100010101111
Alineación E	1110111010001111
Mini-Alineación A	111011

Peine	Patrón
Mini-Alineación B	110111

Resumen de alineación VDJ

La alineación de VDJ permite la identificación de qué segmentos V, D y J se usan en una lectura de secuencia dada. Para cada segmento, se empleó un método heurístico mediante el cálculo de los recuentos de k-mero utilizando múltiples semillas (Tabla 3). Los segmentos candidatos se clasificaron según la cantidad de superposición en sus distribuciones k-mero. Luego se realizó una alineación de programación dinámica completa en los candidatos con mayor puntuación para determinar la mejor coincidencia.

Proceso de alineación VDJ

Para maximizar el número de nucleótidos distintivos, la alineación se realizó en orden decreciente de la longitud del segmento (V, luego J y luego D). Para minimizar la cantidad de secuencia fuera del objetivo (especialmente para los recuentos de k-mero), la región V o J alineada previamente se eliminó antes de intentar la alineación del siguiente segmento. Para alinear los segmentos V y J, se calcularon las semillas 11-mero o 12-mero utilizando los peines listados en la Tabla 3. Los conteos k-mero también se calcularon previamente en todos los segmentos de referencia. Los 5 segmentos de referencia V principales y los 2 segmentos de referencia J superiores fueron elegidos para la alineación y puntuación de la programación dinámica. Como se sabía que los segmentos V y J deben residir en los extremos de las lecturas, se utilizó un método similar al algoritmo de Needleman-Wunsch (Needleman and Wunsch (1970) J. Mol. Biol. 48:443). En contraste con el algoritmo canónico, se utilizaron cero condiciones iniciales para permitir que el inicio de la alineación se produjera en cualquier lugar sin penalización. Luego, la alineación se reconstruyó y se anotó comenzando con el valor máximo de la matriz de puntuación a lo largo de la última fila o última columna, y retrocediendo. Finalmente, los segmentos V o J identificados se eliminaron antes de proceder a la alineación J o D, respectivamente.

Alineación de la región D

Las alineaciones D se realizaron de manera similar. Sin embargo, las dos semillas Mini-Alineación 5-mero enumeradas en la Tabla 3 se utilizaron para seleccionar los 10 segmentos D de referencia principales para llevar a cabo la alineación de la programación dinámica. En este caso, se realizó el algoritmo de alineamiento local canónico de Smith-Waterman (Smith and Waterman (1981) J. Mol. Biol. 147:195), ya que no se disponía de información previa sobre dónde debía residir el segmento D.

Extracción de CDR3

Los segmentos V y J se podaron de acuerdo con las anotaciones de IMGT para la región de unión CDR3: del segundo residuo de cisteína conservado en la región V a través del triptófano o fenilalanina conservado en la región J. Este segmento se almacenó como la región de unión de la región variable correspondiente.

Rendimiento de alineación VDJ

Como actualmente no había una gran base de datos de secuencias de inmunoglobulina de cadena pesada humana para validar el rendimiento del algoritmo, el alineador se calibró utilizando el algoritmo V-QUEST proporcionado como parte de la base de datos IMGT. 66.497 lecturas de tres ejecuciones de secuenciación diferentes se alinearon utilizando el software de alineación V-QUEST. Para cada lectura, las puntuaciones de alineación de programación dinámica para las regiones V, D y J se calcularon por separado para cada segmento de referencia posible, además del segmento correcto producido con el alineador V-QUEST. Las curvas ROC mostraron que la alineación de la región V era la más confiable, seguida de J, seguida de D, como se esperaba (Figura 8). La región D mostró la alineación más ambigua, que se esperaba debido a su corta longitud y alta naturaleza mutada. Por lo tanto, las alineaciones de la región D se ignoraron en los análisis posteriores, y las alineaciones V y J se centraron, junto con la correspondiente región de unión CDR3. Las curvas ROC se utilizaron para elegir los umbrales de puntuación para mantener el número de alineaciones falsamente emparejadas a aproximadamente el 10% o menos.

Agrupación de secuencias

La agrupación de secuencias se realizó para agrupar las secuencias (lecturas) en clones únicos. Este proceso se usó principalmente para asociar secuencias que se originaron a partir de la misma célula/clon, mientras que permite variaciones menores atribuibles a errores de secuenciación. El agrupamiento jerárquico aglomerado de enlace simple o completo se usó con la distancia de edición de Levenshtein como la métrica. Para hacer que el proceso de agrupación sea más manejable, las lecturas se dividieron en función de la identidad V-J. Dentro de cada partición, la agrupación de secuencias se realizó utilizando solo la secuencia de unión CDR3. Además, todas las uniones idénticas se colapsaron antes de la agrupación para mejorar el rendimiento. Debido a que las distancias del clúster solo dependían de los miembros únicos del clúster, esto no afectó los resultados finales del clúster (aunque esto no es cierto para los métodos de clúster de vinculación promedio). Como se describe aquí, los tamaños de los grupos

variaron en varios órdenes de magnitud. A continuación, se muestran varias alineaciones (utilizando CLUSTALW) de varias agrupaciones más pequeños como ilustraciones. La mayoría de las diferencias entre las secuencias estuvieron cerca de los homopolímeros, que son el tipo de mutación más común para la secuenciación 454.

Ejemplo de agrupación 1:

```

1 TGTGCGAGAGAGGGCTACGGTGACTACCGTTACTACTACGGTATGGACGTCTGG 54
  (SEQ ID NO:1)
2 TGTGCGAGAGAGGGCTACGGTGACTACCGTTACTACTACGGTATGGACGTCTGG 54
  (SEQ ID NO:2)
5 TGTGCGAGAGAGGGCTACGGTGACTACCGTTACTACTACGGTATGGACGTCTGG 54
  (SEQ ID NO:3)
6 TGTGCGAGAGAGGGCTACGGTGACTACCGTTACTACTACGGTATGGACGTCTGG 54
  (SEQ ID NO:4)
3 TGTGCGAGAGAGGACTACGGTGACTAC-GT--CTACTACGGTATGGACGTCTGG 51
  (SEQ ID NO:5)
4 TGTGCGAGAGGGTACTACGGTGACTACGG---CCACTACGGTATGGACGTCTGG 51
  (SEQ ID NO:6)
5      ***** * ***** * * *****

```

Ejemplo de agrupación 2:

ES 2 730 951 T3

1 TGTGCGGCAGTTCCCCCCCCT-NAGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 57
(SEQ ID NO:7)
3 TGTGCGGCAGTTCCCCCCCCT-NAGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 58
(SEQ ID NO:8)
9 TGTGCGGCAGTTCCCCCCCCT-CAGGGAACGACATTTTGGGGTGCTTTT-GAGATCTGG 59
(SEQ ID NO:9)
22 TGTGCGGCAGTTCCCCCCCCT-CAGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 58
(SEQ ID NO:10)
5 TGTGCGGCAGTTCCCCCCCCTCAGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 58
(SEQ ID NO:11)
24 TGTGCGGCAGTTCCCCCCCCTC---AGGGAACGACATTT---GGGTGCTTTT-GAGATCTGG 53
(SEQ ID NO:12)
6 TGTGCGGCAGTTCCCCCCCCTC--AGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 56
(SEQ ID NO:13)
10 TGTGCGGCAGTTCCCCCCCCTC--AGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 56
(SEQ ID NO:14)
11 TGTGCGGCAGTTCCCCCCCCTC--AGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 56
(SEQ ID NO:15)
13 TGTGCGGCAGTTCCCCCCCCTC--AGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 56
(SEQ ID NO:16)
19 TGTGCGGCAGTTCCCCCCCCTC--AGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 56
(SEQ ID NO:17)
21 TGTGCGGCAGTTCCCCCCCCTC--AGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 56
(SEQ ID NO:18)
4 TGTGCGGCAGTTCCCCCCCCT-CAGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 57
(SEQ ID NO:19)
17 TGTGCGGCAGTTCCCCCCCCT-CAGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 57
(SEQ ID NO:20)
8 TGTGCGGCAGTTCCCCCCCCT-CAGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 57
(SEQ ID NO:21)
18 TGTGCGGCAGTTCCCCC--CT-CAGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 55
(SEQ ID NO:22)
23 TGTGCGGCAGTTCCCCCCCCT-CAGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 57
(SEQ ID NO:23)
12 TGTGCGGCAGTTCCCCCCCCT-CAGGGAACGACATTTTGGGGTGCTTTT-GAGATCTGG 59
(SEQ ID NO:24)
14 TGTGCGGCAGTTCCCCCCCCT-CAGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 58
(SEQ ID NO:25)
16 TGTGCGGCAGTTCCCCCCCCT-CAGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 57
(SEQ ID NO:26)
0 TGTGCGGCAGTTCCCCCCCCT-CAGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 57
(SEQ ID NO:27)
7 TGTGCGGCAGTTCCCCCCCCT-CAGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 58
(SEQ ID NO:28)
2 TGTGCGGCAGTTCCCCCCCCT-CAGGGAACGACATTTT-GGGGTGCTTTT-GAGATCTGG 58
(SEQ ID NO:29)
20 TGTGCGGCAGTTCCCCCCTCA---GGGAACGACATTT---GGGGTGCTTTT---GAGATCTGG 53
(SEQ ID NO:7)
15 TGTGCGGCAGTTCCCCCCCCTCCAGGGAACGACATTT---GGGGTGCTTTT-GAGATCTGG 57
(SEQ ID NO:30)

Ejemplo de agrupación 3:

ES 2 730 951 T3

4 TGTGCGACGG-TGGGA-GTTCCCC-ACCGGTTTT-GATATCTGG 40
(SEQ ID NO:31)
79 TGTGCGACGG-TGGGACGTTCCCCTACCGGTTTT-GATATCTGG 42
(SEQ ID NO:32)

ES 2 730 951 T3

1 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:33)
16 TGTGCGACGG-TGGGA-GTTCCC--AC-GGTTTT-GATATCTGG 38
(SEQ ID NO:34)
3 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:35)
6 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:36)
8 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:37)
11 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:38)
20 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:39)
27 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:40)
32 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:41)
38 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:42)
43 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:43)
49 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:44)
55 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:45)
59 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:46)
64 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:47)
69 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:48)
74 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:49)
12 TGTGCGACGG-TGGGA-GTTCTC--ACCGGTTTTTGATATCTGG 40
(SEQ ID NO:50)
67 TGTGCGACGG-TGGGA-GTTC-C--ACCGGTTTT-GATATCTGG 38
(SEQ ID NO:51)
70 TGTGCGACGG-TGGGA-ATTC-C--ACCGGTTTT-GATATCTGG 38
(SEQ ID NO:52)
10 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTTTGATATCTGG 40
(SEQ ID NO:53)
24 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTTTGATATCTGG 40
(SEQ ID NO:54)
34 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTTTGATATCTGG 40

ES 2 730 951 T3

(SEQ ID NO:55)
44 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTTTGATATCTGG 40
(SEQ ID NO:56)
47 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTTTGATATCTGG 40
(SEQ ID NO:57)
85 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTTTGATATCTGG 40
(SEQ ID NO:58)
87 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTTTGATATCTGG 40
(SEQ ID NO:59)
13 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:60)
15 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:61)
22 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:62)
28 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:63)
33 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:64)
39 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:65)
45 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:66)
50 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:67)
56 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:68)
61 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:59)
65 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:70)
71 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:71)
78 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:72)
83 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:73)
89 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:74)
92 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:75)
95 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:76)
98 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:77)

ES 2 730 951 T3

101 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:78)
23 TGTGCGACGG-TGG-A-GTTCCC--ACCGGTTTT-GATATCTGG 38
(SEQ ID NO:79)
51 TGTGCGACGG-TGG-A-GTTCCC--ACCGGTTTT-GATATCTGG 38
(SEQ ID NO:80)
80 TGTGCGACGG-TGG-A-GTTCCC--ACCGGTTTT-GATATCTGG 38
(SEQ ID NO:81)
17 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:82)
25 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:83)
30 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:84)
35 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:85)
40 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:86)
46 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:87)
52 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:88)
57 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:89)
62 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:90)
66 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:91)
72 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:92)
81 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:93)
84 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:94)
90 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:95)
93 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:96)
96 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:97)
99 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:98)
100 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:99)
97 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39

ES 2 730 951 T3

(SEQ ID NO:100)
94 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:101)
91 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:102)
86 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:103)
82 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:104)
77 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:105)
75 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:106)
73 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:107)
68 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:108)
63 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:109)
58 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:110)
53 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:111)
48 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:112)
41 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:113)
37 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:114)
31 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:115)
26 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:116)
19 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:117)
9 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:118)
7 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:119)
5 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:120)
2 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTTT-GATATCTGG 39
(SEQ ID NO:121)
14 TGTGCGACGG-TGGGA-GTTCCC-TACCGGTTTT-GATATCTGG 40
(SEQ ID NO:122)

21 TGTGCGACGG-TGGGA-GTTCCC-TACCGGTTTT-GATATCTGG 40
 (SEQ ID NO:123)
 42 TGTGCGACGG-TGGGA-GTTCCC-TACCGGTTTT-GATATCTGG 40
 (SEQ ID NO:124)
 102 TGTGCGACGG-TGGGA-GTTCCC-TACCGGTTTT-GATATCTGG 40
 (SEQ ID NO:125)
 18 TGTGCGACGG-TGGGA-GTTCCC--ACCGGTTT--GATATCTGG 38
 (SEQ ID NO:126)
 54 TGTGCGACGG-TGGGA-GTTCCCCTACCGGTTTT-GATATCTGG 41
 (SEQ ID NO:127)
 60 TGTGCGACGG-TGGGA-GTTCCCCTACCGGTTTT-GATATCTGG 41
 (SEQ ID NO:128)
 76 TGTGCGACGGGTGGGACGTTCCCCTACCGGTTTT-GATATCTGG 43
 (SEQ ID NO:129)
 88 TGTGCGACGG-TGGGA-CTTCCC--ACCGTTTT-GATTTCTGG 39
 (SEQ ID NO:130)
 29 TGTGCGACGG-TGGGA-GTTCCC--ACCGTTTTGATTATCGTG 40
 (SEQ ID NO:131)
 36 TGTGCGACGG-TGGGACGTTCCCCTACCGTTTTGAT-ATCGTG 42
 (SEQ ID NO:132)

***** ** * ** * ** * ** *

5 Luego se examinó la distribución de las distancias de cofenética observadas en el árbol de enlace para determinar la distancia óptima para cortar el árbol. Se esperaba que se observara un gran número de eventos de vinculación que se producen a una baja distancia de copenética que representan errores de secuencia. De hecho, se observó una caída rápida en la distribución hasta una distancia de 4-5 ediciones, después de lo cual la distribución mostró un hombro y un descenso más lento (Figura 19A). Esto fue consistente con la relación observada entre el corte de distancia y el número de agrupamientos obtenidos (Figura 19B), ya que también hubo una caída rápida a corta distancia seguida de un hombro y una marcada desaceleración.

Agrupación de series de tiempo: k-medias

10 Aunque la agrupación de secuencias descrita anteriormente se realizó en datos combinados para los ocho puntos de tiempo, cada identificador de clon único se dividió en los 8 puntos de tiempo identificando los 454 códigos_barras MID. Por lo tanto, cada clon definió una serie de tiempos donde el número de lecturas era un proxy para el nivel de expresión. Como análisis inicial, la agrupación de k-medias se realizó en todas las series temporales utilizando datos combinados de las ejecuciones SR1, SR2 y TR1. El algoritmo de k-medias se realizó en frecuencias de clones (recuentos de lectura
 15 en un punto de tiempo determinado normalizado por el número de lecturas) utilizando la métrica de distancia euclidiana con 100 agrupamientos (elegidos arbitrariamente). Los grupos más grandes contenían clones que aparecieron por un solo punto de tiempo y desaparecieron. Esto se observó en una variedad de niveles de expresión y en todos los puntos de tiempo (Figura 15A). Cuando estos agrupamientos se filtraron y se realizó nuevamente el agrupamiento de k-medias, se observó una gran variedad de comportamiento dinámico. Sin embargo, algunos de los grupos más grandes
 20 aún eran clones que aumentaron y disminuyeron rápidamente en frecuencia (Figura 15B).

Agrupación de series de tiempo: mapas autoorganizables (SOM)

Debido a la selección arbitraria en el número de clones para el agrupamiento de k-medias y la gran diversidad de comportamiento dinámico, las series de tiempo se agruparon utilizando un método SOM implementado en el paquete de software GEDI (Eichler et al. (2003) Bioinformatics 19:2321). Cada mosaico dentro de un mosaico representa un minigrupo de clones que tienen patrones de expresión muy similares en todos los puntos de tiempo analizados. Al definir una topología en los mosaicos, se agrupan grupos similares, lo que permite una determinación simple del número adecuado de grupos de grupos. Los mismos clones están obligados a la misma posición de mosaico para todos los mapas SOM, lo que permite la comparación directa de los clones en función del patrón de mosaico general. El color de las cuadrículas indica el valor centroide del nivel de expresión génica para cada minigrupo. El agrupamiento de SOM se realizó en una cuadrícula rectangular de 50x49 usando una inicialización aleatoria. Se definió una distancia euclidiana en las series de tiempo y se realizaron dos fases de iteración de entrenamiento (20 y 80). Después de la agrupación, los grupos de interés se identificaron manualmente para su posterior análisis.

Reproducibilidad, requisitos de muestreo y límites de cuantificación

Se realizaron múltiples análisis para determinar si el VDJ-oma se tomaba una muestra lo suficientemente profunda y a qué nivel la dinámica podía cuantificarse de manera confiable (es decir, el límite de cuantificación (Currie (1968) Analytical Chemistry 40:586)). La serie de tiempos replicada de clones elegidos al azar se trazó (de las ejecuciones SR1, SR2 y TR1) para obtener una vista cualitativa de la replicación (Figura 15). Los diagramas de dispersión de las ejecuciones duplicadas mostraron altos niveles de correlación (Figura 9, se muestran las correlaciones de Pearson). Para determinar las características de muestreo de las muestras, se dibujaron pares de submuestras de los datos de diferentes tamaños y se calcularon los coeficientes de correlación (Figura 2C). Los coeficientes de correlación se aproximaron a la unidad a un nivel de muestreo de 10^5 . Para determinar los límites de cuantificación, se generaron diagramas de dispersión para cada punto de tiempo de una réplica dada contra otra. Los datos se agruparon en un eje y la media y la desviación estándar se calcularon en el otro eje. Usando estos valores, se calculó el coeficiente de variación (CV) como una función del nivel de expresión (Figura 10). Se esperaba que los niveles de expresión más altos tuvieran variaciones relativas más bajas y, por lo tanto, deberían haber resultado en CV más pequeños. El ruido en el medio de las curvas fue un artefacto del esquema de muestreo ponderado utilizado, y también se observó ruido en niveles de expresión altos debido a un muestreo deficiente. Un límite inferior de CV de aproximadamente 0.5 ocurrió en frecuencias de clon de $<10^{-4}$ - 10^{-3} , dependiendo de las muestras. El muestreo de clones de baja frecuencia se acercó rápidamente al régimen de ruido de Poisson (CV de 1). Diferentes números de lecturas en diferentes puntos de tiempo/réplicas causaron la aparición del ruido de Poisson en diferentes frecuencias.

Estimaciones de la diversidad

Se utilizaron múltiples métodos para estimar la diversidad de anticuerpos. Para estimar las diversidades típicas de pequeñas muestras de sangre de 1 mL, se utilizaron los estimadores basados en la abundancia de Chao1 y ACE (Magurran, A.E. Measuring Biological Diversity (Blackwell Pub., Malden, MA; 2004)). En resumen, Chao1 y ACE se derivaron como estimadores no paramétricos basados en la abundancia para la diversidad. Siguiendo a Magurran, si S_{obs} era el número de especies observadas en la muestra, F_1 era el número de especies observadas una vez (singletons) y F_2 era el número de especies observadas dos veces (dobletes), entonces el estimador Chao1 se definió como

$$S_{Chao1} = S_{obs} + \frac{F_1^2}{2F_2}$$

Si S_{raro} fue el número de especies raras (especies con menos o igual a 10 observaciones), $S_{abundante}$ fue el número de especies abundantes (especies con más de 10 observaciones), N_{raro} fue el número de individuos en especies raras, F_i fue el número de especies con i observaciones, $C_{ACE} = 1 - F_1/N_{raro}$, y

$$\gamma_{ACE}^2 = \max \left\{ \frac{S_{raro}}{C_{ACE}} \frac{\sum_{i=1}^{10} i(i-1)F_i}{N_{raro} (N_{raro} - 1)} - 1, 0 \right\}$$

entonces

$$S_{ACE} = S_{abundante} + \frac{S_{raro}}{C_{ACE}} + \frac{F_1}{C_{ACE}} \gamma_{ACE}^2$$

Sin embargo, las muestras utilizadas no fueron homogéneas, ya que se extrajeron en diferentes puntos de tiempo después de un ataque inmunitario, lo que puede afectar la distribución de anticuerpos, en comparación con las muestras homogéneas. Se esperaba que tales efectos fueran pequeños, por lo que los estimadores aún se aplicaban. Más significativamente, el nivel de cobertura acaba de pasar un punto de inflexión (lo que indica el inicio de la convergencia hacia una asíntota). En tales casos, las diversidades reales tendieron a ser más altas que los valores estimados, y los estimadores funcionaron como límites inferiores de la diversidad. Las versiones remuestreadas de

estos estimadores se muestran en las Figuras 12A-12D. Se observaron puntos de inflexión en casi todas las curvas, lo que indica un acercamiento hacia una asíntota.

De manera similar, los estimadores análogos basados en la incidencia Chao2, ICE, Jack1, Jack2 (Magurran, Supra) se utilizaron para estimar la diversidad sanguínea total, utilizando las múltiples muestras de sangre de manera análoga a los cuadrantes múltiples en ecología. Las mismas consideraciones estadísticas aplicadas a estos conjuntos de estimadores. Los resultados se muestran en la Figura 2E. El método de estimación de diversidad de Arstila et al. (Arstila et al. (1999) Science 286:958) también se simuló utilizando todas las combinaciones posibles de V-J y todos los picos posibles de pseudotipos de espectros. Este método no tuvo en cuenta las consideraciones estadísticas, se esperaba una gran variedad de posibles estimaciones de diversidad. De hecho, se obtuvo una distribución muy amplia de valores, aunque el valor medio todavía estaba cerca del rango de 10^6 (Figura 2E).

Distribución de las fluctuaciones

Para caracterizar la dinámica de los clones de anticuerpos, se analizó la distribución de sus fluctuaciones típicas. Para las series de tiempo que fueron positivas en todos los puntos de tiempo, el cambio de veces logarítmico de un punto de tiempo se calculó sobre el punto de tiempo anterior. Las distribuciones se muestran en las Figuras 20A-20B. Parecían estar log-distribuidos normalmente y centrados en cero. La distribución del punto temporal de -14 d fue descentrada debido a un artefacto de muestreo. También se esperaba que la varianza de la distribución se correlacionara con la duración del intervalo de tiempo (es decir, las duraciones más largas conducen a mayores fluctuaciones). De hecho, las Figuras 20A-20B muestran una fuerte correlación, con un coeficiente de correlación de Pearson de 0.91.

Generación de tipos de pseudoespectros

Para cada punto de tiempo, los datos de recuento se dividieron por el uso de V-J y luego por la longitud de la unión CDR3. Para cada longitud de trama posible (múltiplo de 3), se calculó el número de lecturas en trama frente a las lecturas fuera de trama que fueron una base demasiado larga o demasiado corta, y se ajustaron estos datos a una distribución Gaussiana. Para cada combinación V-J, todos los gaussianos se superponen y la curva se renormaliza al número total de lecturas. En la Figura 13 se muestran los tipos de pseudoespectros completos. El uso de los gaussianos fue solo para fines de visualización.

Alineación a la base de datos IMGT/LIGM anotada en especificidad

La base de datos IMGT/LIGM (Giudicelli et al. (2006) Nucleic Acids Res. 34:D781) contiene aproximadamente 9000 secuencias con especificidades anotadas, 14 de las cuales son específicas para la hemaglutinina de la influenza, 6 para la hepatitis A y 85 para la hepatitis B. Aunque esto representa una muestra muy pequeña, las lecturas de secuenciación a este conjunto de secuencias anotadas de especificidad se alinearon para buscar secuencias altamente idénticas. Para realizar las alineaciones, se utilizó el paquete de software exonerado (Slater and Birney (2005) BMC Bioinformatics 6:31), que permite la alineación utilizando varios modelos de alineación. Las alineaciones se realizaron utilizando la lectura completa o solo la unión CDR3, y utilizando versiones traducidas y/o versiones de nucleótidos de las secuencias. Debido a que se esperaban muchas coincidencias de fondo (en el caso de alineación de lectura completa), y debido a la sensibilidad de la afinidad de unión a la secuencia exacta de aminoácidos, las alineaciones se clasificaron por identidad de secuenciación, y solo se analizaron las coincidencias que eran casi completamente idénticas mano. Se realizaron ejemplos de coincidencias con anticuerpos específicos de influenza o de hepatitis (Figuras 21 y 22) utilizando la alineación completa de secuencias de nucleótidos. Se observó una alineación deficiente en la región de unión CDR3, que es la forma más común de desalineación. La mejor alineación coincidente se encontró alineando solo las regiones de unión traducidas. Un anticuerpo anti-digoxina fue emparejado con éxito. Se muestran la alineación solo de unión (Figura 23) y la alineación de lectura completa (Figura 24).

Herramientas de software y disponibilidad de datos

El paquete de software de análisis utilizado para los experimentos descritos en este documento se puede descargar en el sitio web de Worldwide: arep.med.harvard.edu/vdj, o la versión más reciente se puede extraer como un repositorio git de GitHub en el Worldwide Web Site: github.com/laserson/vdj. El software se implementó principalmente en python con algún código C utilizado para aumentar el rendimiento. Los paquetes NumPy (el Worldwide Web Site: numpy.scipy.org/) y SciPy (el Worldwide Web Site: scipy.org/) también se confiaron en gran medida en los cálculos numéricos, y el matplotlib (el Worldwide Web Site: matplotlib.sourceforge.net/) se confió en el paquete para la visualización de datos. Los datos de secuenciación están disponibles para descargar en formato FASTA en el Worldwide Web Site: arep.med.harvard.edu/vdj junto con las versiones procesadas en formato XML. Los datos de secuenciación se enviarán al Archivo de Lectura Corta del NCBI y a la base de datos internacional ImMunoGeneTics (IMGT) para su incorporación al Laboratoire d'ImmunoGénétique Moléculaire (LIGM).

Ejemplo V

Captura de cadena pesada y liviana de una célula única

Se proporcionan métodos para capturar tanto las cadenas pesadas como las livianas de un anticuerpo que se origina en una sola, permitiendo la captura simultánea de millones de células a la vez (en lugar de las técnicas anteriores de clasificación de una célula por pozo en una placa).

Pequeña oligotransfección de células individuales

5 Dos oligos, cada uno de los cuales presenta una secuencia complementaria a la cadena pesada y liviana respectivamente, están fusionados parcialmente entre sí a través de una secuencia universal. Este enlace de fusionado parcial se mantiene en la etapa posterior y el uso de ácido nucleico bloqueado (LNA) puede ser útil para este propósito. Los oligos también pueden albergar fosforotioatos para proteger al oligo de las nucleasas. Además, se puede incluir una biotina interna o distal en uno o ambos de los oligos, que se usarán en la recuperación o en las moléculas corriente abajo. Este oligo enlazador luego se transfecta en muestras de linfocitos utilizando tecnologías de transfección comunes, como el reactivo de transfección TransIT®-Oligo (Mirus Bio LLC), por ejemplo. El componente de transfección también puede incluir un inhibidor de la ARNasa como precaución. Luego, las células se incuban a una temperatura más alta, de modo que se realiza el fusionado del ARN celular que se captura (también se puede hacer con ADN). Idealmente, la temperatura se eleva lo suficiente para favorecer la fusión de la estructura secundaria del ARN objetivo y para lograr un fusionado específico al enlazador de oligo haciendo coincidir la temperatura de fusión lo más cerca posible (generalmente cerca de 65°C), mientras que el fusionado parcial de los dos oligos se mantiene debido al diseño de secuencia inherente (particularmente con el uso de cebadores de LNA, donde la unión de fusionado puede diseñarse para alcanzar una temperatura superior a 90°C). Tras el fusionado de incubación de los ARN dirigidos, las células pueden lisarse químicamente y la molécula de ARN diana de enlace oligo puede recuperarse a través de su biotina utilizando una perla magnética recubierta con estreptavidina (como Dynabead M280 de Invitrogen, u otro equivalente). También se podría usar cualquier otro sustrato sólido recubierto con estreptavidina. El extremo 3' del enlazador oligo se puede usar directamente para la transcripción reversa (utilizando el superíndice III, Invitrogen u otra transcriptasa inversa equivalente) de las cadenas pesada y liviana, seguida de la síntesis de ADNc de la segunda cadena. La molécula final se amplifica de las perlas magnéticas utilizando un cebador distal con los resultados en la recuperación de un ScFv completo compuesto por las cadenas pesadas y livianas que se originan en una sola célula, y esto se logró en millones de células a la vez en la misma muestra de reacción.

Captura en perlas

En ciertas realizaciones de ejemplo, una perla magnética se acopla con una mezcla de oligos complementarios a las cadenas pesadas y livianas, pero también aloja una secuencia universal corriente arriba (una perla no magnética funcionará, sin embargo, las perlas magnéticas facilitan los lavados y recuperaciones corriente abajo). En ciertos aspectos, la química de acoplamiento covalente o los enlaces estreptavidina-biotina (por ejemplo, si se agrega una biotina a los oligos durante la fabricación) se utilizan para producir oligoperlas. La oligoperla luego se transfecta en células individuales utilizando tecnologías de transfección comunes, o los enfoques de transfección basados en liposomas (siempre que la perla sea lo suficientemente pequeña para ser transfectada), o la perla (o algunas perlas), se encapsula dentro de una emulsión en presencia de un solo linfocito. La célula se somete a lisis luego mediante desnaturalización térmica o ciclos de congelación y descongelación. En ciertos aspectos, los inhibidores de la ARNasa se agregan durante la etapa de emulsificación. Las emulsiones luego se amplifican por PCR (asumiendo que la PCR y el cebador también se introdujeron durante la emulsificación), o se incubaron a una temperatura óptima de hibridación de oligo y/o concentración de sal, de modo que tanto las cadenas pesadas como las livianas se unirán a su secuencia complementaria en la perla. Las emulsiones se rompen y las perlas se recuperan. Los componentes celulares no unidos y el ADN o ARN no específico se eliminan por lavado y el ARN de perla resultante se somete a transcripción reversa (para ARN) o extensión de cebador (para ADN), de modo que la perla ahora alberga la secuencia completa de cadenas pesada y liviana de una celda individual. Estas perlas se pueden someter luego a cualquiera de las técnicas descritas para unir ambas cadenas entre sí antes de la secuenciación, como la PCR superpuesta (SOE-PCR), o mediante el uso de la recombinación CRE-LOW de ambas cadenas en emulsión o en una reacción superdiluida. La unión Cre-lox de las cadenas livianas y pesadas de una sola célula se ha demostrado previamente en los linfocitos reticulados con formaldehído, pero con una eficiencia muy baja (Chapal et al. (1992) *Biotechniques*, PMID: 9298226), mientras que los métodos proporcionados en este documento tienen acoplamiento cruzado al proceso de secuenciación de alto rendimiento, lo que permite a un experto en la técnica la capacidad de examinar un gran repertorio inmune.

Variación SOE-PCR

SOE-PCR nunca se ha acoplado a los métodos de secuenciación y análisis de alto rendimiento de un gran repertorio inmune, sino que, en cambio, perpetuo se ha limitado al estudio de un número limitado de células (como las que realiza Symphogen en su tecnología Symplex (Worldwide Web: symphogen.com/web/guest/symplex) descrita por Meijer (2006) *J. Mol. Biol.*, PMID: 16563430, por ejemplo). En consecuencia, se proporcionan múltiples métodos de SOE-PCR que mejoran la eficiencia. Estos métodos se utilizan en SOE-PCR de células individuales ya sea capturando células individuales en emulsión o reticulando células en formaldehído como se describe más adelante en este documento. Esto permite la generación de ScFv de una sola célula de millones de células individuales en un solo volumen de reacción, de modo que se pueda acoplar con la secuenciación de alto rendimiento y el análisis de repertorios inmunes.

Ejemplo VI

La expresión y el ensayo funcional del anticuerpo y nanocuerpo seleccionaron candidatos del análisis de secuenciación de alto rendimiento de un repertorio inmune humano

Expresión in vitro del anticuerpo de dominio único.

- 5 1. Una colonia aleatoria de cada clon se levantó con un palillo y se inoculó en 5 mL de LB/Amp, y las bacterias se incubaron durante la noche a 37°C con agitación a 225 rpm.
2. Las bacterias se centrifugaron a 4.700 g durante 10 min, se desechó el sobrenadante y los gránulos se usaron para miniprep (utilizando un kit de miniprep Qiagen).
- 10 3. Las reacciones de transcripción se establecieron: agua libre de nucleasas 8 µL; Regulador de transcripción 5X 4 µL; mezcla NTP 4 µL; plantilla de ADN (~0.5 µg/mL) 2 µL; T7 ARN polimerasa 2 µL; TOTAL 20 µL.
4. La reacción se incubó durante 120 minutos a 32°C. Las soluciones se volvieron turbias después de este tiempo, lo que indica una transcripción eficiente (precipitados de calcio).
- 15 5. Las reacciones de traducción se establecieron directamente: Lisado para la expresión de proteínas 12.5 µL; Proteínas accesorias 2.5 µL; Agua libre de nucleasas 3.75 µL; Solución salina A 1 µL; Aminoácidos -Met 0.5 µL; Aminoácidos -Leu 0.5 µL; Inhibidor de la ARNasa 1 µL; Mezcla de energía 1.25 µL; Plantilla de ARN de la reacción de transcripción 2 µL; TOTAL 25 µL.
6. Las reacciones de traducción se incubaron a 32°C durante 2 horas, luego se agregaron 2 µL de la reacción de transcripción.
- 20 7. La reacción se incubó durante 2 horas más a 32°C, luego se enriqueció nuevamente con 1 µL de la reacción de transcripción y se incubó durante 2 horas más.

Las proteínas se purificaron de la siguiente manera

1. La reacción se diluyó con 28 µl de regulador de unión a His 2X (fosfato 40 mM, NaCl 1 M, imidazol 40 mM, pH 7.4).
2. Las muestras se limpiaron en columnas Vivaspin de 0.2 µm para eliminar todos los residuos y complejos grandes.
- 25 3. La resina de sefarosa Ni-NTA (Amersham) se agitó suavemente durante 5 minutos hasta que toda la suspensión fue homogénea, luego se transfirió una cantidad total de (50 µL x (# muestras + 1)) a un tubo de 1.5 mL.
4. La resina se lavó 3 veces con 1X de regulador de unión a His (fosfato 20 mM, NaCl 0.5 M, imidazol 20 mM, pH 7.4) para eliminar el etanol y equilibrar la resina.
5. Se agregaron 50 µl de la resina a cada muestra.
6. Las muestras se voltearon durante la noche a 4°C.
- 30 7. Las muestras se lavaron 3 veces con 150 µL de regulador de unión a His.
8. Las proteínas se eluyeron de la resina en 100 µL de regulador de elución de His (fosfato 20 mM, NaCl 0.5 M, imidazol 0.5 M, pH 7.4) con un volteo a temperatura ambiente durante 1 hora.
- 35 9. La resina se retiró y las muestras se desalaron utilizando Vivaspin 5K de la siguiente manera: a. Las columnas Vivaspin 5K se equilibraron con 0.4 mL de PBS. b. Se agregaron 100 µL de muestra a la columna. c. Las columnas se colocaron en una centrifuga con el filtro mirando hacia afuera, luego se centrifugaron (15.000 g, 15 min). d. El flujo se descartó y la muestra se resuspendió en 0.4 mL de PBS y se centrifugó nuevamente. e. El paso d se repitió dos veces más. f. Los retenidos (~50 µL promedio) se transfirieron a tubos nuevos, se añadió solución de azida de sodio a una concentración final de 0.02% y las muestras se almacenaron a 4°C. La concentración de proteínas y/o la pureza se evaluaron utilizando Nanodrop.
- 40 Síntesis de genes y protocolo de clonación.

La PCR amplifica la síntesis del gen IDT utilizando el cebador directo e inverso M13

Preparar de una reacción de la siguiente manera (1 tubo/13 tubos): ~50 ng/µl Síntesis de genes (1 µl/-); dH₂O (34.6 µl/449.8 µl); 5x Regulador HF (10 µl/130 µl); 15 µM de cebador 1* (1.7 µl/22.1 µl); 15 µM de cebador 2* (1.7 µl/22.1 µl); DNTP 25 mM (0.5 µl/6.5 µl); Phusion Hotstart (0.5 µl/6.5 µl); (49 µl/tubo). Ciclo térmico como sigue: 9 a 12 ciclos.

1- 98°C durante 30 segundo

ES 2 730 951 T3

- 2- 98°C durante 10 segundos
- 3- 50°C durante 20 segundos
- 4- 72°C durante 20 segundos para el paso 2, 25x
- 5- 72°C durante 5 minutos
- 6- 4°C pausa

Purificar con Qiagen, resuspender en 30 µl.

Doble digestión con EcoRI-HF y NotI-HF

Preparar la reacción como sigue

	1 tubo	13 tubos
• H2O	10.5 (a 50 µl)	136.5
• ADN	30 µl	-
• 10xNEB regulador 4	5 µl	65
• 100x BSA	0.5 µl	6.5
• NotI-HF	2 µl	26
• EcoRI-HF	2 µl	26

Incubar a 37°C durante 15 min (20 µl/tubos). Inactivar por calor a 65°C durante 20 min.

- 5 Purificación por Qiagen, elución en 42 (cargaré 20 un gel); Nanodrop unos pocos ~50 ng/µl. Extraer utilizando geles selectivos por tamaño: purificar por Qiagen, eluir en 30 µl; Nanodrop: ~9 ng/µl.

Ligación en vector de termoexpresión

- 10 Se usaron insertos de relación 3:1 al vector. Reacción preparada de la siguiente manera (1 tubo/14 tubos): H₂O (4 µl/56 µl); regulador 2x Ligasa (10 µl/140 µl); Vector 50 ng/µl (2 µl (0.04 pmol)/28 µl); Inserciones de ~350 pb (9 ng/µl) (3µl (0.12 pmol=30 ng/-µl); Ligasa rápida (1 µl/14 µl); (17 µl/tubos). Incubar a 22°C durante 30 min. Se realizó la clonación con células X10. En placas de 200 µl usando perlas, esta placa de la mañana tuvo resultados completos y satisfactorios. Reclonada en NEB Shuffle T7 compiten E. coli (en C3026H). Se utilizó miniprep Ido, se diluyó 3 veces, se usó 1 µl (~80 ng) en 25 µl de E. coli. Se generaron lotes medios de clones seleccionados.

Se escogió una colonia en 50 ml de LB + 1x ampicilina, creció durante la noche a 30°C

clon ID	nombre del clon	ID de síntesis	unión	lote grande
Muestrador de placa IDT (fila de etiquetas amarillas A) de izquierda a derecha.				
1	GMC_1688_Día 1_2	1-IDT		
2	GMC_1337_Día 1_3	2-IDT		
-	GMC_299_Día 1_4	3-IDT	no hecho todavía	
3	GMC_162_Día 1_5	4-IDT	10, 59	X

ES 2 730 951 T3

clon ID	nombre del clon	ID de síntesis	unión	lote grande	
Muestrador de placa IDT (fila de etiquetas amarillas A) de izquierda a derecha.					
4	GMC_961_Día 7_2	5-IDT			
5	GMC_1458_Día 7_3	6-IDT			
6	GMC_1494_Día 7_4	7-IDT	10, 59		X
7	GMC_769_Día 7_5	8-IDT	59		X
8	GMC_967_Día 21_2	9-IDT	59		X
9	GMC_750_Día 21_3	10-IDT			
10	GMC_2629_Día 21_4	11-IDT	10, 59		X
11	GMC_668_Día 21_5	12-IDT			
Placa de ADN2.0 (etiqueta roja de cualquier pozo de 1 a 5) muestrador de arriba a abajo.					
12	IDO_1_NEGCONT_BCELLTUMOR_optHEK1	IDO1			
13	GMC_1687_Día 1_1	IDO2			
14	GMC_932_Día 7_1	IDO3	59		X
15	GMC_393_Día 7_6	IDO4	10, 59		X
16	GMC_376_Día 21_1	IDO5			
17	IDO_6_POSCONT_GFPMINIMIZER_optHEK1	IDO6			X

Reclón

5 Se reclonó en T7 aleatoria E. coli: 3, 6, 7, 8, 10, 12. Utiliza 2 µl de 13 dilución en 20 µl de E. coli T7 Shuttle (NEB C3026H). Se sembraron 200 µl en una placa supergrande (exceso de clon, intentar con 100 µl la siguiente vez). Se produjo una colonia en 5 ml de LB a 30°C durante la noche. Utilizado este un cultivo iniciador 500 µl para 50 ml de LB + 1000x ampi (50 mg/ml de uso en IX). Creció durante 3-4 horas, se verificó OD. Nanodrop. Cubeta: 0.3. Pedestal: 0.08 (Nanodrop dice que debería ser 10 veces más bajo que el OD600 normal). Se mantuvo un poco de lisado de 10 células para la prueba directa de SPR.

10 Q8: ¿Cómo debo expresar mi proteína de interés en Shuffle? A8: Para las condiciones iniciales, recomendamos el uso de medios enriquecidos a 30°C. De lo contrario, durante la noche a 16°C es posible. A 30°C o 16°C, inocule 1% de cultivo durante la noche y haga crecer las células a 30°C durante 3 horas hasta OD600 ~0.8 y luego induzca la expresión de proteína durante al menos 5 horas a 30°C o durante la noche a 16°C. Si usa 37°C, inocule 1% de cultivo durante la noche y haga crecer las células durante 2 horas a 37°C hasta OD600 ~0.8 y luego induzca la expresión de proteína durante al menos 6 horas a 37°C.

15 Protocolo

Plásmido de expresión transformado en la cepa T7 Express. Se colocan en placas de selección de antibióticos y se incuban durante la noche a 37°C. Se resuspendió una sola colonia en 10 ml de cultivo líquido con antibiótico. Se

incubó a 37°C hasta OD600 alcanzó 0.4-0.6. Inducido con 40 µl de un patrón de 100 mM de IPTG (concentración final de 0.4 mM) y fue inducido durante 2 horas a 37°C. Se verificó la expresión mediante gel de proteína teñido con Coomassie, inmunoprecipitación Western o ensayo de actividad. Expresión controlada tanto en el extracto celular total (soluble + insoluble) como en la fracción soluble sola. Para gran escala, se inocularon 1 L de medio líquido (con antibiótico) con una colonia recién cultivada o 10 ml de cultivo recién cultivado. Se incubó a 37°C hasta que OD600 alcanzó 0.4-0.6. Se agregó IPTG a 0.4 mM. Se indujo por 2 horas a 37°C o 15°C durante la noche.

Inducción de IPTG

Los cultivos se cultivaron durante un total de 4 horas. IPTG estaba a 800 mM, por lo que se añadieron 25 µl a cada matraz de cultivo de 50 ml. Se incubó durante 6 horas a 30°C (por lo tanto, no se detiene antes de las 8h30). Las células bacterianas fueron convertidas en gránulos por centrifugación a 5.000 × g durante 10 minutos, (ultracentrífuga). Se eliminó el sobrenadante y se almacenó a -80°C.

Procedimiento para extraer proteínas de las bacterias (protocolo B-PER de Thermo)

Opcional: Agregar 2 µl de lisozima y 2 µl de DNasa I por 1 ml de Reactivo B-PER. Se añaden inhibidores de proteasa sin EDTA. Se añaden 4 ml de reactivo B-PER por gramo de sedimento celular. Se obtuvieron 0.5 g, se añadieron 2 ml. Se pipetea la suspensión hacia arriba y hacia abajo hasta que quede homogénea. Se incubó 10-15 minutos a temperatura ambiente. Lisado centrifugado a 15.000 × g durante 5 minutos para separar las proteínas solubles de las proteínas insolubles. Nota: Si un gran porcentaje de proteína sobreexpresada permanece en el sedimento, la proteína de interés podría expresarse en cuerpos de inclusión. Se usa el Reactivo de solubilización de cuerpos de inclusión (Producto No. 78115) o alteran las condiciones de expresión para minimizar la formación de cuerpos de inclusión.

Purificación de His-tag utilizando resina Ni-NTA.

Para condiciones nativas, se prepararon los siguientes reguladores: Regulador de equilibrio: fosfato de sodio 20 mM, cloruro de sodio (PBS) 300 mM con imidazol 10 mM; pH 7.4; Regulador de lavado: PBS con imidazol 25 mM; pH 7.4; Regulador de elución: PBS con imidazol 250 mM; pH 7.4.

Procedimiento para la purificación de proteínas marcadas con His mediante un método por lotes

La resina HisPur Ni-NTA permite la personalización de la estrategia de purificación. Las condiciones de purificación se pueden escalar según sea necesario. El procedimiento puede realizarse a temperatura ambiente o a 4°C. Se agregó una cantidad apropiada de resina de Ni-NTA a un tubo. El tubo se centrifuga durante 2 minutos a 700 × g y se retira y desecha con cuidado el sobrenadante. Se agregan dos volúmenes de lecho de resina de regulador de equilibrio y se mezcla hasta que la resina esté completamente suspendida. El tubo se centrifuga durante 2 minutos a 700 × g y se retira con cuidado y se desecha el regulador. Se preparó la muestra mezclando extracto de proteína con el regulador de equilibrio, de modo que el volumen total sea igual a dos volúmenes de lecho de resina (en su lugar, se agrega todo el componente del regulador de equilibrio a la muestra de manera que la muestra esté a 1x, luego se agrega 1 volumen de lecho a la resina, luego se agregan 650 µl de suspensión de resina a la muestra).

Se añadió el extracto de proteína preparado al tubo y se mezcló en un rotador de extremo a extremo durante 60 minutos. Se centrifugó el tubo durante 2 minutos a 700 × g. Si se desea, se guarda el sobrenadante para el análisis posterior. Se lavó la resina con dos volúmenes de lecho de resina de regulador de lavado. Se centrifugó el tubo durante 2 minutos a 700 × g. Si se desea, se guarda el sobrenadante para el análisis posterior. Repetir el paso de lavado y controlar el sobrenadante midiendo su absorbancia a 280 nm hasta alcanzar la línea de base. Proteínas unidas a His unidas eluidas usando un volumen de lecho de resina de regulador de elución. El tubo se centrifuga durante 2 minutos a 700 × g. Retirar con cuidado y guardar el sobrenadante. Se repite este paso dos veces, guardando cada fracción de sobrenadante en un tubo separado. Se monitorizó la elución de la proteína midiendo la absorbancia de las fracciones a 280 nm o mediante el reactivo de ensayo Coomassie Plus (Bradford) (producto No. 23238) o el ensayo proteico Pierce® a 660 nm (producto No. 22660). La proteína eluida se pudo analizar directamente mediante SDS-PAGE. Nota: Para eliminar el imidazol para aplicaciones posteriores, se usó filtración por gel (por ejemplo, columnas de desalinización Thermo Scientific Zeba Spin) o diálisis (por ejemplo, casetes de diálisis Thermo Scientific Slide-A-Lyzer). Las muestras que contienen guanidina 6 M•HCl deben dializarse frente a un regulador que contiene urea 8 M antes del análisis SDS-PAGE. El kit de preparación de muestras SDS-PAGE de Thermo Scientific Pierce (producto No. 89888) también se puede usar para eliminar la guanidina.

Nanodrop fase A280

Muestra ID	Proteína Conc. mg/ml
enlazar lisado	34.649
Lavar 1	5.319

ES 2 730 951 T3

Muestra ID	Proteína Conc. mg/ml
Lavar 2	1.19
Lavar 3	0.343
Eluir 1	0.875
Eluir 2	0.308

5 Limpieza con Vivaspín y concentrar. Los tubos se centrifugaron (15.000 g, 5 min) y el sobrenadante se desaló corriendo en columnas Vivaspín 5K (15.000 g, 15 min), después de que se descartó cada flujo de lavado y se reconstituyó el retenido a 0.5 mL con PBS, se repitió 4 veces. El volumen promedio de retenido después de cada lavado fue de ~100 mL. Las proteínas fueron examinadas en Nanodrop. A280

Nanodrop fase A280

Muestra ID	Conc. Proteína mg/ml
3	9.304
6	9.115
7	12.924
8	7.937
10	9.823
12	5.792

SDS-PAGE

	Muestra ID	µg/µl	diluido 1/10 o 1/5	para 5 µg	a 6.5 h2o
1	Escalera de proteínas NEB				
2	enlazar lisado	34.65	3.46	1.44	5.06
3	Lavar 1	5.32	-	0.94	5.56
4	Lavar 2	1.19	-	4.20	2.30
5	Lavar 3	0.34	-	14.58	-8.08
6	3	9.30	1.86	2.69	3.81
7	6	9.12	1.82	2.74	3.76
8	7	12.92	2.58	1.93	4.57

ES 2 730 951 T3

	Muestra ID	µg/µl	diluido 1/10 o 1/5	para 5 µg	a 6.5 h2o
9	8	7.94	1.59	3.15	3.35
10	10	9.82	1.96	2.55	3.95
11	12	5.79	1.16	4.32	2.18
12	Inclusión				
13	Pro inhib				

MOPS, 200 voltios 45 minutos (Figuras 63, 64)

SDS toma 2

- 5 Se ejecutó el primer lote de proteínas, también se cargó el clon de la proteína de muestra 3 después de la segunda purificación de la etiqueta de miltenyi his. MOPS, 200 voltios, 30 minutos (debería haber hecho 45 minutos) (Figura 95). La proteína estaba en BSA al 0.5% y glicerol al 10%. Los materiales grandes eran BSA. 1, escalera NEB; 2, clon 6 Ni-NTA; 3, clon 2 post myltenyi; 4-11, Mi-NTA (+BSA)

Se vuelve a limpiar y se concentra el ADN del clon 393 (el neutralizador potencial)

Se extrajeron 3 pozos, se purificaron con AMPure, se resuspendieron en 15 µl de H₂O. Nanodrop: 500 ng/ul

- 10 Traducción in vitro (kit de expresión Thermo Human in vitro)

Transcripción

Se harán 600 µl de kit de proteína final (20x 25 µkit), por lo tanto, se necesitarán 40 µl de reacción de transcripción (s0 2 valor de reacción).

	1 tubo	8 tubos
Agua sin Nucleasa	8	16
Regulador de transcripción 5X	4	8
Mezcla NTP	4	8
ADN clonado (500 ng/µl)	2	4
ARN polimerasa de T7	2	4
Total	20	40

- 15 Mezclar suavemente los tubos e incubar durante 60-75 minutos a 32°C.

Traducción

Se agregaron los reactivos en el orden indicado en un tubo sin ARNasa/ADNasa de 1.5 o 0.5 ml. Para obtener los mejores resultados, se incubó el lisado con las proteínas accesorias durante 5 minutos antes de agregar los componentes posteriores.

ES 2 730 951 T3

	1 tubo	20 tubos (en un 1.5 ml)
Lisado para la expresión de proteínas	12.5	250
Proteínas accesorias	2.5	50
Agua libre de nucleasas	3.75	75
Solución salina A	1	20
Aminoácidos menos Met	0.5	10
Aminoácidos menos Leu	0.5	10
Inhibidor de ARNasa (opcional)	1	20
Mezcla de energía	1.25	25
Mezcla de transcripción	2	40
Total	25	500

D

Dividir en 5 tubos de 100 µl. Se incubó a 30°C en PCR durante 4 horas, luego a 4°C durante la noche

Prueba de purificación

División en 5 tubos de 95 µl: 1, lisado; 2, Mylteniy (se usan 50 µl de perlas en 500 µl de regulador de lisis); 3, Invitrogen; 4, Cobalt Thermo; 5, Ni (Tabla 17).

5

#	descripción		ug/ml o ng/ul	A280	260/280	ug/ul	para 10 ug	factor de dilución		volumen para 10ug	agua a 6.5
1		in vitro lysate	46297	46.297	1.6	46.30	0.22	10	4.63	2.16	4.3
2	mylteniy	postenlace	186235	186.235	0.56	186.24	0.05	50	3.72	2.68	3.82
3		elución AB/Ag	882	0.882	1.11	0.88	11.34	1	0.88	11.34	0.00
4		elución Invitrogen	27	0.027	2.65	0.03	370.37	1	0.03	370.37	0.00
5		elución SDS mylteniy	3390	3.39	0.71	3.39	2.95	1	3.39	2.95	3.55
6	Invitrogen	postenlace	34384	34.384	1.65	34.38	0.29	10	3.44	2.91	3.59
7		elución Invitrogen	837	0.837	1.79	0.84	11.95	1	0.84	11.95	0.00
8	Cobalt thermo	postenlace	22703	22.703	1.73	22.70	0.44	10	2.27	4.40	2.10
9		elución 1	43	0.043	2.2	0.04	232.56	1	0.04	232.56	0.00
10		elución 2	0	0	82.49	0.00	#DIV/0!	1	0.00	#DIV/0!	0.00
11	Ni-Nta nugen	postenlace	14513	14.513	1.66	14.51	0.69	5	2.90	3.45	3.05
12		elución 1	833	0.833	1.4	0.83	12.00	1	0.83	12.00	0.00
13		elución 2	417	0.417	1.64	0.42	23.98	1	0.42	23.98	0.00
14	escalera										

Tabla 17.

SDS PAGE. Regulador MES, se ejecuta durante 35 minutos, simplemente tinción azul (Figura 65). La clonación puede haber puesto la etiqueta His fuera del marco, pero la proteína está en el marco.

Protocolos de ensayo funcionales

Preparación de ensayo FACS

- 5 1. La mitad de las muestras de proteínas se etiquetaron con FITC de la siguiente manera: a. Se transfirieron ~20 μ L de la muestra de proteína a un tubo de 0.2 mL y el volumen se reconstituyó a 100 μ L con PBS (la concentración final de proteína se estimó en 0.01 mg/mL); b. Se disolvieron ~1 mg de FITC (Pierce) en 1.5 mL de DMF; c. Se añadió 1 μ L de la solución de FITC/DMF a la muestra y los tubos se incubaron en la oscuridad a temperatura ambiente durante 30 minutos; d. Las proteínas se limpiaron del exceso de FITC mediante columnas Vivaspin 5K y se lavaron 2 veces con 0.4 mL de PBS; e. Se añadió azida de sodio a una concentración final de 0.02% y las muestras (~20 μ L) se almacenaron a 4°C hasta su uso.
- 10 2. Las proteínas HA (Bris59, Bris10 y Flor04, adquiridas de Sino Biologicals, China) se biotinilaron utilizando EZ-Link-NHS-Biotina (Pierce) mediante incubación en hielo durante 1 hora y luego se limpiaron en columnas Vivaspin 10K, con 2 lavados con 0.4 mL de PBS.
3. Las proteínas HA biotiniladas se usaron para recubrir las perlas magnéticas recubiertas con estreptavidina de Dynal mediante incubación en PBS (10 μ L de perlas, 5 μ L de proteína biotinilada, en 100 μ L de volumen total de PBS), con volteo, a temperatura ambiente durante 1 hora.
- 15 4. Las perlas se lavaron 3 veces con 100 μ L de regulador de ensayo de perlas (BSA al 0.1%, glicerol al 10%, azida de sodio al 0.02% en PBS, pH 7.4) usando un imán.
5. Las perlas terminadas se almacenaron en 100 μ L de regulador de ensayo a 4°C hasta su uso.

Ensayo FACS

- 20 1. Se mezclaron 1 μ L de suspensión de perlas (agitada en vórtice poco antes del uso) con 5 μ L de solución de anticuerpo y 19 μ L de regulador FX (BSA al 0.1%, azida de sodio al 0.02% en HBSS, pH 7.4) 30 min en hielo (final concentración de anticuerpos ~10 μ g/mL).
2. Las muestras se reconstituyeron con 25 μ L de regulador FX frío y se analizaron con un citómetro de flujo Accuri C6.

Mediciones de QCM de unión de anticuerpo/antígeno

Los cristales se recubrieron con anticuerpo un día antes del experimento de la siguiente manera:

- 25 1. Los cristales de oro QCM (adquiridos en Qsense) se recubrieron con HS-PEG-NH₂ (de Nanocs) incubando con 100 μ L de una solución de 5 mg/mL de HS-PEG-NH₂ en agua ultrapura en una cámara húmeda, 1 hora a temperatura ambiente.
2. Los cristales se lavaron suavemente con 0.5 mL de PBS.
- 30 3. Se mezclaron 30 μ L de solución de anticuerpo (~1.5 μ g de proteína total) con 70 μ L de PBS que contenía 200 pmol de EDC y 200 pmol de sulfo-NHS (ambos de Pierce, sulfo-NHS no pesa, mezclado con 22 μ L de agua ultrapura antes de mezclar), se incubó a temperatura ambiente durante 1 minuto, luego se aplica sobre la superficie del cristal y se incubó a temperatura ambiente durante 1 hora en una cámara húmeda.
4. Se añadió 1 μ L de una solución de BSA de 10 mg/mL y la fase líquida se mezcló suavemente.
5. La cámara húmeda se movió a 4°C para su almacenamiento durante la noche sin eliminar el anticuerpo.
- 35 Las mediciones de QCM se realizaron en un instrumento Qsense E4 de la siguiente manera
1. PBS se hizo fluir a través de las celdas a 0.1 mL/min hasta que los cristales se equilibraron (~30 minutos).
2. Se hicieron fluir 2.2 mL de soluciones de proteína HA (1 y 10 μ g/mL) a través de las células cambiando de nuevo a PBS una vez que se terminó la muestra.
3. Los cristales se lavaron con NaOH 0.1 M entre diferentes concentraciones.

40 Purificación de nanocuerpos expresados en E. coli

1. Los sedimentos celulares (el sobrenadante se eliminó completamente, los sedimentos congelados a -20°C) se resuspendieron en 1 mL de regulador de lisis (Tris-HCl 10 mM, pH 7.5, EDTA 1 mM, Triton X-100 al 0.05%, 5 μ g/mL DNasa I, cóctel inhibidor de proteasa 1X de Pierce).
- 45 2. Las suspensiones se transfirieron a tubos cónicos de 15 mL y se sometieron a sonicación con un sonicador de sonda estándar, mientras se mantenían en hielo. Parámetros de sonicación: [10 seg encendido + 5 seg apagado] x 4 pulsos, 100% de salida.
3. Los ultrasonidos se centrifugaron (4.800 g, 10 min, 4°C) y los sobrenadantes se transfirieron a tubos de 1.5 mL.

ES 2 730 951 T3

4. 4 mL (8 muestras x 0.5 mL/muestra) de resina de sefarosa Ni-NTA (Amersham) se lavaron en un tubo de 15 mL para eliminar el regulador de almacenamiento (contiene 20% de etanol), mediante 3 centrifugaciones repetidas (500 g, 1 min) y lavados con 10 mL de regulador de unión a His (fosfato 20 mM, NaCl 0.5 M, imidazol 20 mM, pH 7.4), después del último lavado, el sedimento de resina se reconstituyó con 4 mL de regulador de unión a His 2X.
- 5 5. Se agregaron 0.5 mL de resina a cada muestra y los tubos se rotaron a 4°C durante la noche.
6. La resina se lavó dos veces con 1.5 mL de regulador de unión a His (30 min cada lavado, centrifugación 1000 g, 1 min).
7. Después del lavado final, el lecho de resina se dejó húmedo (~0.6 mL) y se almacenó durante la noche a 4°C.
- 10 8. Se añadieron 0.5 mL de regulador de elución de His (fosfato 20 mM, NaCl 0.5 M, imidazol 0.5 M, pH 7.4) y los tubos se rotaron durante la noche a 4°C.
9. Los tubos se centrifugaron (15.000 g, 5 min) y el sobrenadante se desalinizó corriendo en columnas Vivaspin 5K (15.000 g, 15 min, después de que se descartó el flujo de cada lavado y se reconstituyó el retenido a 0.5 mL con PBS, repetido 4 veces. El volumen promedio de retenido después de cada lavado fue de ~100 µL).
10. Las proteínas fueron examinadas en Nanodrop.
- 15 11. Se tomaron muestras de 50 µL para el etiquetado.
Etiquetado de nanocuerpos
1. 50 µL (~35 µg según Nanodrop) de cada muestra se mezclaron con 450 µL de regulador de borato 50 mM pH 8.5.
2. Las muestras se limpiaron en columnas Vivaspin 5K (15.000 g, 15 min, retenido ~20 µL). La concentración de proteína calculada (promedio en las muestras) después de este paso fue de ~1.5 mg/mL.
- 20 3. Se disolvió DyLight-649 (1 mg) en 100 µL de DMF y se agregó 1 µL a cada muestra.
4. Los tubos se incubaron en la oscuridad durante 1 hora a temperatura ambiente.
5. Las proteínas marcadas se limpiaron del exceso de colorante mediante columnas Vivaspin 5K, con 3 lavados con 470 µL de PBS y un lavado en 470 µL de regulador Tris 10 mM, pH 7.5, para inactivar el colorante residual.
6. La concentración se calculó en el retenido (~20-25 µL cada uno): 1.3-1.5 mg/mL.
- 25 Ensayo de neutralización
Cepas de gripe - crecimiento en células A549 y titulación inmune
1. Las células utilizadas fueron células epiteliales de pulmón humano A549 (carcinoma). Las células se cultivaron en placas de 48 pozos en 0.4 mL de medio (medio ATCC # F-12K + FBS al 10% + Pen/Strep). Las células se transfirieron a la placa después de colocar en placa A549 desde un matraz de cultivo T75 [confluencia > medio aspirado > 1.3 mL de TrypLE o tripsina-EDTA > 3-5 min a 37°C > completado a 6 mL con medio > se toman 50 µL por pozo en una placa de 48 pozos que contiene 0.35-0.4 mL de medio].
- 30 2. Al 80% de confluencia, las células infectadas con el virus, al agregar 10 µL de virus a la dilución deseada, mezclar bien con una pipeta.
3. Se incubó a 37°C durante 24-48 h.
- 35 4. Medio aspirado, aplicar 100 µL de TrypLE, incubar 5 min a 37°C.
5. Mezclado bien con una pipeta, tome 200 µL de células en una placa de 96 pozos nueva en forma de u.
6. Fijado con formaldehído, concentración final del 2.5%. En hielo durante al menos 10 min.
7. Lavado (centrifugado 3000 rpm, 3 min, 4°C), resuspendido en 100 µL de metanol al 100% congelado. Se mezcla bien.
- 40 8. Se incubó en hielo durante 10 min.
9. Se agregaron 100 µL de regulador FX frío, se giró nuevamente como en 7. Se resuspende en 100 µL de regulador FX frío.
10. Se agregó 1 µL de nucleoproteína A anti-Influenza A conjugada con FITC (de Millipore) por pozo, se mezcló bien y se incubó en hielo durante 60 min.

11. Se centrifugó como en 7, se resuspendió en 200 µL de FX frío y se centrifugó nuevamente.

12. Se resuspendió en 200 µL de FX frío, analizado por FACS.

5 13. Título determinado de la siguiente manera: Título (UIF/µL) = D50 × C/(2 × V). D50: la concentración de virus en la que se infectan el 50% de las células (FITC+) C, número inicial de células en el pozo (consulte el Worldwide Web Site: [invitrogen.com/etc/medialib/en/filelibrary/pdf.Par\).4786.File.dat/Useful_Numbers_Y1447_2_Useful_Nmbrs.pdf](http://invitrogen.com/etc/medialib/en/filelibrary/pdf.Par).4786.File.dat/Useful_Numbers_Y1447_2_Useful_Nmbrs.pdf) para obtener detalles útiles); V, Volumen de virus (en µL) usado para infectar el pozo.

Ensayo de neutralización

1. Se incubaron células A549 en confluencia al 75% a 0.4 ml de medio en una placa de 48 pozos. Se agregaron nanocuerpos a diversas concentraciones (50, 5, 0.5, 0.05 µg/mL, diluidos en medio A549).

10 2. Inmediatamente después, las células se infectaron con la cepa Bris10 a una MOI de ~33% (0.33) a un volumen de 10 µL.

3. Las células se incubaron durante 24 h.

4. Se aspiró el medio (y se blanqueó). Las células se lavaron una vez con 0.5 mL de HBSS.

5. Se agregaron 100 µL de TrypLE y la placa se incubó a 37 grados durante 5 min.

15 6. Las células se aspiraron en una placa de 96 pozos y se fijaron inmediatamente con formaldehído como en el paso 6 en la sección anterior.

7. El protocolo continuó desde el paso 6 en la sección anterior.

Resultados de neutralización, Figuras 66 y 67.

Ejemplo VII

20 Correlación de cadena pesada y liviana

Análisis inicial de la cadena liviana

25 Los datos de la cadena liviana se obtuvieron de bibliotecas creadas a partir de unos pocos puntos de tiempo de un experimento. Los histogramas de las longitudes de lectura se exponen en la Figura 46. Hubo un pico en 380 (esperado), y una agregación más pequeña alrededor de 325. Se alinearon aleatoriamente 10 lecturas de cada pico. 7/10 del pico 380 eran cadenas livianas (todos kappa). Ninguno de los 325 picos fueron cadenas livianas o cadenas inmunes. Eran proteínas aleatorias que debían de haberse amplificado. Los valores de corte para la selección del tamaño fueron 365-405.

Alineación VDJ/clasificación y agrupamiento de datos de la cadena liviana

30 Los datos de la cadena liviana se tomaron a través de una tubería inicial. El contenido del archivo README se detalla en la Tabla 4 a continuación. El trabajo se realizó en orquesta. Los enlaces simbólicos se colocaron en el directorio de datos estables con los resultados.

Tabla 4

```

Al ejecutar el canal completo en la orquesta: estos guiones se ejecutaron cuando el repositorio vdj está en la etiqueta
v1.3 Los datos intermedios están en el directorio correspondiente en orquesta: /home/u12/VDJ-
oma/análisis/20100810_canal_cadenas_livianas 1. fasta2vdjxml.py 2 . tamaño_select.py Primero convierto el
archivo fasta a vdjxml, y, Tamaño selecciona las lecturas basadas en el historial de lectura, 365-405, (de
/home/ul2/VDJ-oma/estable-data/bruto_cadenas_livianas): python ~/código/vdj/bin/ fasta2vdjxml.py
cadenas_livianas.20100802.fasta|python ~/código/vdj/bin/tamaño_select.py --min 365 --max 405 >
cadenas_livianas.20100802.tamaño365-405.vdjxml Hay 496605 cadenas del tamaño seleccionado en el archivo. 3.
vdjxml2parts.py Divide vdjxml en trozos pequeños y colóquelo en el directorio de trabajo python
~/código/vdj/bin/vdjxml2parts.py --tamaño del paquete 10000 - nombre base ~/VDJ-
oma/analysis/canal_cadenas_livianas/data/cadenas_livianas.20100802.tamaño365-405.vdjxml
cadenas_livianas.20100802.tamaño365-405.vdjxml Cambie el directorio a todas las partes: cd~/ VDJ-
oma/análisis/canal_cadenas_livianas/data 4. código de barras_id.py 5. hebra_codificación.py Identifique los
códigos_barras para cada lectura, y Determine si tenemos la cadena correcta o no para el ARCHIVO en
cadenas_livianas.20100802.tamaño365-405.vdjxml. *; hacer NOMBRE = $ {ARCHIVO%.tamaño *}".
prealineación.vdjxml." ${FILE #*.vdjxml.*} bsub - qcompartido_2h -o pre-alineación.log "python
~/código/vdj/bin/código de barras_id.py --códigos_barras ~/VDJ-oma/estable-
data/códigos_barras/454MID.códigos_barras.fasta $FILE | python ~/código/vdj/bin/hebra_codificación.py --locus
    
```

```

IGK --locus IGL> $NAME "hecho # python ~/código/vdj/bin/código de barras_id.py --códigos_barras ~/VDJ-
oma/estable-data/códigos_barras/454MID.códigos_barras.fasta # python ~/código/vdj/bin/hebra_codificación.py --
locus IGK --locus IGL Algunas ESTADÍSTICAS: # Número de cadenas cat *prealineación* | grep "<CadenaInmune>"
| wc -1 496605 # Num con códigos_barras cat *prealineación* | grep "<código de barras>" | wc-1 486258 # Números
que se complementaron de forma inversa cat *prealineación* | grep "revcomp" | wc-1 240928 # Desglose del código
de barras para NUM; en 1 2 3 4 5 6 8 9; do cat *prealineación* | grep "<código de barras> 0$NUM" | wc -1 ejecutado
01 71 0.015% 02 0 03 174550 35.9% 04 7 0.001% 05 142036 29.2% 06 0 08 169594 34.9% 09 0 6. align_vdj.py
para FILE en cadenas_livianas.20100802.prealineación.vdjxml.*; HACER
NOMBRE=${ARCHIVO%.prealineación*}.vdjxml."${FILE#*. vdjxml.*} bsub -qcompartido_12h -o alineación.log
python ~/código/vdj/bin/align_vdj.py --locus IGK --locus IGL $FILE $NAME ejecutado 7. cat_vdjxml.py python
~/código/vdj/bin/cat_vdjxml.py cadenas_livianas.20100802.vdjxml.* > cadenas_livianas.20100802.alineado.vdjxml
Mover datos procesados hacia abajo en un directorio: mv cadenas_livianas .20100802.alineado.vdjxml .. cd .. 8.
filtro_VJ.py python ~/código/vdj/bin/filtro_VJ.py cadenas_livianas.20100802.alineado.vdjxml
cadenas_livianas.20100802.VJ_filtered.vdjxml grep "<CadenaInmune>"
cadenas_livianas.20100802.VJ_filtered.vdjxml | wc-1 356813 para NUM en 3 5 8; do grep "<código de barras>
0$NUM" cadenas_livianas.20100802.VJ_filtered.vdjxml | wc-1 ejecutado 03 131837 05 99618 08 117703 9.
partition_VJ.py mkdir particiones python ~/código/vdj/bin/ partition_VJ.py --particiones nombre
base/cadenas_livianas.20100802 cadenas_livianas.20100802.VJ_filtered.vdjxml cd particiones # Cuántas cadenas
en cada partición? para ARCHIVO en cadenas_livianas.20100802.*.vdjxml; do grep "<CadenaInmune>" $FILE | wc
-1 ejecutado | selección-n resultados parciales: ... 10021 11239 11285 12592 13451 13999 17935 10. agrupación
cdr3.py para INFILE en cadenas_livianas.20100802.*.vdjxml; do VJID=${INFILE#cadenas_livianas.20100802.}
VJID=${VJID%.vdjxml} OUTFILE=${INFILE%.vdjxml}agrupado.vdjxml bsub -qcompartido_1limitado -o
agrupación.log python ~/código/vdj/bin/cluster_cdr3.py --corte 4.5 --tag $VJID --enlace sencillo$EN ARCHIVO
$FUERA DE ARCHIVO ejecutado # ¿Cuánto tiempo tomó? grep "CPU" agrupación.log | Resultados parciales de
sort-n-k4: tiempo de CPU: 25.38 seg. Tiempo de CPU: 29.70 seg. Tiempo de CPU: 34.28 seg. Tiempo de CPU:
42.63 seg. Tiempo de CPU: 44.29 seg. Tiempo de CPU: 46.46 seg. Tiempo de CPU: 48.57 seg. Tiempo de CPU:
57.70 seg. Tiempo de CPU: 59.88 seg. 11. cat_vdjxml.py python ~/código/vdj/bin/cat_vdjxml.py
cadenas_livianas.20100802.*. Agrupado.vdjxml> ../cadenas_livianas.20100802.agrupado.vdjxml cd .. # ¿Cuántas
cadenas? grep "<Cadena Inmune>" cadenas_livianas.20100802.agrupado.vdjxml | wc -1 356813 #; ¿Cuántos
clones únicos en total? grep "<clon>" cadenas_livianas.20100802.agrupado.vdjxml | selección| uniq | wc -1 5009 #
¿Cuántas uniones únicas en total? grep "<unión>" cadenas_livianas.20100802.agrupado.vdjxml | selección | uniq |
wc-1 53880

```

Series temporales y distribución clónica de datos de cadenas livianas

5 Las series de tiempo y los histogramas de frecuencia de clonación se generaron a partir de los datos de la cadena liviana. Solo había unos 5000 clones. Sin pretender estar limitados por la teoría científica, es posible que el valor de corte para definir clones deba recalibrarse para las cadenas livianas. Sin pretender estar limitados por la teoría científica, se puede esperar una menor diversidad ya que no existe una región D. Ver Figuras 47 y 48.

Series de tiempo de cadenas livianas que utilizan uniones únicas en lugar de clones

10 Debido a la preocupación de que las definiciones de agrupamiento para cadenas pesadas no se traduzcan en cadenas livianas, se generaron cifras de series de tiempo para cadenas pesadas y livianas utilizando solo uniones únicas. En la Figura 49 se muestra una gráfica de los dos (NOTA: cadenas pesadas en la parte superior, cadenas livianas en la parte inferior).

Las secuencias de unión superiores se establecen en la Tabla 5 a continuación.


```

uniones de cadena pesada: 1d
TGTGCGCAAACCGATAGCAGCAACTTAGACTACTACTACCACCGAATGGGCGTCTGG
TGTGCGAAAGGGGAGATTGTGGTGGTGGCTAGTTGCCCCATTTAGACTACTACTACGGTATGGACGTCTGG
TGCGGAGAGGGACGGGGGACCCCTACGGTGACTATTATGGTGGTGGCTTTTGGATGTCTGG
TGTGCGCAAACCGATAGCAGCAACTTAGACTTCTACTTCTACGGTTTGGACGACTGG
TGTGCGGAGATCGGGGATAGTTGGAGTCTCCACAATTTGACTTCTGG
TGTGCGCAAACCGATAGCAGCCACATAGATTCTACTACTATGGTATGGACGACTGG
TGTGCGGTTCAAGATTGTAGTACTACCACCTGCTATCCTGCGAGTTCCTACTACTACTATAACATGGGCGTCTGG
TGTGCGGTTCAAGATTGTAGTACTACCACCTGTTATCCTGCGAGTTCCTACTACTACTACAACATGGGCGTCTGG
TGTGCGCAAACCGATAGCAGCCACATAGACTTCTACTACTACGGTATGGACGACTGG
TGTGCGGAGATCGGGGATAGTTGGAGCCCTCCACAATTTGACTTCTGG 7d
TGTGCGCAAACCGATAGCAGCAACTTAGACTACTACTACCACCGAATGGGCGTCTGG
TGTGCGGAGATGATCCATATTACGGCAGTATTGGTTATCGTATTGACTCCTGG
TGTGCGGAGAAATCCCGCTAAGATCGAGTGGGACGCTACTACTACTACGGTATGGACGTCTGG
TGTGCGGAGAGACCTTACCTGGAGATACTTTGACTCCTGG
TGTGCGGAGAGTGTCTCTAGTAGTGGTTATTACTACTACTTTGATTACTGG
TGTGCAAGATCCCTCATTCTATATAGTACTACATTGCCTACTGG
TGTGCGAGGACGTACGGTAAACCGAGGATTCCGGTACTCTTACTCTACTATTACTTCATGGACGTCTGG
TGCGGAGAACTTTGTATTCTCTGGTAAAGTATAGTACTGGCTGGTACTACTTTGACTACTGG
TGTGCGGAGAACTTTTTCATATCTGG
TGTGTAAGAGTTAAGGGTGGCATAGCAGCAGCTGGTACCCTGCGGGTACTTCGATCTCTGG 21d
TGTGCGGAGAGTACGGGGATATTGTAATGGTGGTAGCTGCTACTTTGACTACTGG
TGTGCGGAGAGAGAGGGCATTAGTGGGAGGTAGTACGACTCTCGGATACTGG
TGTGCGGAGAGGCGGGCTTCAACCTTTAAAGTCTACTATCACTACATGGACGTCTGG
TGTGCGGAGACATATGCGGGTGGGAGCCCTAGTCAAAGTCTTTTGGATGTCTGG
TGTGCGGAGACATATGCGGGTGGGAGTCCTAGTCAAAGTCTTTTGGATGTCTGG
TGTGCGGAGACATATGCGGGTGGGAGTCCTAGTCAAAGTCTTTTGGATGTCTGG
TGTGTAAGAGCGGTTTCCGGGTCGAACATACATCTTTGACTACTGG
TGTGCGGAGAGGCGGAGTGGGAGCTAAAGAGCCGACCGTTTACTACTTTGACCACCTGG
TGTGCGGAAATTTTGGAGCGAATTTGGAAACTACTGGTATGGTTTGGACGTCTGG
TGTGCGGAGACATCAGTATAATGTTGGTAACTCCTGGGCTTTTGGATCTGG light chain junctions: 1d
TGTCAACAGGCTAACAGTTTCCCGCTCACTTTC TGTCAATCACCAGACAGCAGTGGTACTAATGAAGTCTTC
TGTCAATCACCAGACAGCAGTGGTACTTATGAAGTCTTC
TGCGGAACATGGGATACCAGCCTGGGTAATAATTGGGTGTTTC
TGCGGAACATGGGATACCAGCCTGAGAATTAATTGGGTGTTTC TGCAACAGTTTACTAATTTCCCGCTCACTTTC
TGTCAACAGTATGGTAGTTTATGGCGCACTTTT TGTGAGGCGTGGGACAGCAGCAGTGGTATTC
TGCGGAACATGGGATACCAGCCTGAGGATTAATTGGGTGTTTC
TGCTCAGCATGGGACAGCAGCCTCAGTGTCTGGGTGTTTC 7d TGTGAGGCGTGGGACAGCAGCAGTGGTATTC
TGCTGCTCATATGCAGGCAACTCTTATGTCTTC TGTCAATCACCAGACAGCAGTGGTACTAATGAAGTCTTC
TGTGAGCAGTATGGTAGCTCACCTTGGAGCTTC TGTGAGCAATATGGTAGCTCACCGACGTTTC
TGTAACCTCCCGGACAGCAGTGGTAACCATGTGGTATTC TGTGAGCAGGCTAGCACCTGGCCTGGGACTTTC
TGTGAGGTGTGGGATAGTAGTAGTATCATGTGGTATTC

TGTCAATCAGCAGACAGCAGTGGTACTTATGTGGTATTC
TGCGGAGCGTGGGATAGCAGCCTGAGTGTCTGTGGTCTTC 21d TGTGAGGCGTGGGACAGCAGCAGTGGTATTC
TGTCAACAAATATGGTAACTCACCGGTCACTTTC TGCTTCTCTACTTTGATGGTCTTATAGTTTTC
TGTGAGCATTATCATACTCCACCGTACACTTTT TGCCAAACAGTATAATACCTGGTGGACATTC
TGTTTACTCTATTATAATGGTGTGAGGGTGTTC TGTAACCTCCCGGACAGCAGTGGTAAACCATGTGGTATTC
TGCTGCTCATATGCAGGTAGTAGCACTTTTGTCTTC TGTCAATCAGCAGACAGCAGTGGTACTTATGTGGTATTC
TGCTCAGCATGGGACAGCAGCCTCAGTGTCTGGGTGTTTC

```

Tabla 5

5 Las secuencias correspondientes a las uniones expresadas más altas de la cadena liviana y las series de tiempo de la cadena pesada se exponen en la Tabla 6 a continuación. La secuencia de comandos fue ecxtraer_secuencias.py y la salida se colocó manualmente en secuencias_uniones_altamente_expresadas.fasta.

```

>pesada_1d|FWMETYS01DQGF4|2
TTGTGGCTATTTTAAAAGGTGTCCAGTGTGACGTGCAGCTGGTGGAGTCTGGGGGAGGCTTGGTACAGCCTGGGAAGT
CCCTGAGACTCTCCTGTGCAGCCTCTGGATTACCTTTAGCAACTATGCCATGTCCTGGGTCCGCCAGGCTCCAGGGA
AGGGCCTGGAGTGGGTCTCAGGTATTAGTGGTAGTGGTGCTGCCACATACTACGCAGACTCCGTGAAGGGCCGGTTCA
CCATCTCCAGAGTCAATTCAGGAACACGCTCCTTCTGCAAATGAACAGCCTGAGAGCCGAGGACACGGCCCTATATT
ACTGTGCGCAAACCGATAGCAGCAACTTAGACTACTACTACCACGGAATGGGCGTCTGGGGCCAAGGGACCACGGTCA
CCGTCTCCTCAGCATCCCCGACCAGCCCCAAGG >pesada_1d|FWMETYS01A182M|2
TTGTGGCTATTTTAAAAGGTGTCCAGTGTGAGGTGCAGCTCTTGGAGTCTGGGGGAGGCTTGGCACAGCCTGGGGGT
CCCTGAGACTCTCCTGTGGAGCCTCTGGATTACCTTTAGTAACATATGGCATGACCTGGGTCCGCCAGGCTCCAGGGA
AGGGGCTGGAGTGGGTCTCAGGTATTAGTGGTAGTGGTGCTAGTACATTCTACGCAGACTCCGTGAAGGGCCGGTTCA
CCATCTCCAGAGACAATCCAAGAACACGCTGTATCTGCAAATGGACAGCCTGAGAGCCGAGGACACGGCCGTCTATT
ACTGTGCGAAAGGGGAGATTGTGGTGGTGCTAGTTGCCCCATTTAGACTACTACTACTACGGTATGGACGTCTGGG
GCCAAGGGACCACGGTCACCGTCTCCTCAGCATCCCCGACCAGCCCCAAGG >pesada_1d|FWMETYS01ESRCQ|2
CGGCTCCAGATGGGTCTGTCCAGCTGCAGTTGCAGGAGTCGGGCCAGGAGTGGTGAAGCCTTCGGAGACCCTGT
CCCTCACCTGCACTGTCTCTGGTGGCTCCATCAGCACTAGCAGTTATTACTGGGGCTGGATCCGCCAGCCCCAGGGA
AGGGGCTGGAGTGGATTGGGACTATCTATCTAGTGGGGCTCCTACTACAACCCGTCCCTCAAGAGTCAAGTCGCCA
TATCCGTTGACATGTCCAAGAATCAATTCCTCCTGAAGGTGAACCTATAATCGCCGAGACACGGCTGTGTATTACT
GCGCGAGAGGGACGGGGGGACCTACGGTGACTATTATGGTGGTGCTTTTGATGTCTGGGGCCAAGGGAAAAGGGTCC
CCGTCTCTCAGCATCCCCGACCAGCCCCAAGG >pesada_1d|FWMETYS01ALGIE|2
TTGTGGCTATTTTAAAAGGTGTCCAGTGTGAGGTGCGGCTGTTGGAGTCTGGGGGAGGCTTAGTGCAGCCTGGGGGT
CCCTGAGACTCTCCTGTGCAGCCTCTGGATTACCTTTAGCAACTATGCCATGACCTGGGTCCGCCAGGCTCCAGGGC
AGGGGCTGGAGTGGGTCTCAGGAATTAGTGGTAGTGGTGCAAGCACATACTACGAAGACTCCGTGAAGGGCCGGTTCA
CCATCTCCAGACAGAATCCAAGAACACGCTGTATCTGCAAATGAATAGCCTGAGAGCCGAGGACACGGCCGTATACT
ACTGTGCGCAAACCGATAGCACCAACTTAGACTTCTACTTCTACGGTTTGGACGACTGGGGCCAAGGGACCACGGTCA
CCGTCTCCTCAGCATCCCCGACCAGCCCCAAGG >pesada_1d|FWMETYS01EMQ99|2
CGGCTCCAGATGGGTCTGTCCAGGTGCAGCTGCAGGAGTCGGGCCAGGACTGGTGAAGCCTTCGGAGACCCTGT
CCCTCACCTGCAAAGTTTCTATTGGCTCCGTGAGGAATTATTACTGGAGCTGGATCCGGCAGTCCCCGGGAAGGGAC
TGGAGTGGATTGCATATATCTTTCCCAATGGGAGGACCAGCCGCAATCCCTCCTCAGAGTCGAGTCACCATATCAA

```

TTGACACACCCAAAAATCAGTTCTCCATGTTGCTGAGCTCTGCGACCGCCGAGACACGGCCGCTATTACTGTGCGA
 GATCGGGGATAGTTGGAGTCCCTCCACAATTTGACTTCTGGGGCCAGGGAATCCTAGTCACCGTCTCCTCAGCATCCC
 CGACCAGCCCCAAGG >pesada_1d|FWMETYS01A36YN|2
 TTGTGGCTATTTTAAAAGGTGTCCAGTGTGAGGTGCAGCTGTTGGAGTCTGGGGGAGGCTTGGTGCAGCCTGGGGGT
 CCCTGAGACTCTCCTGTGCAGCCTCTGGATTACGTTTCAGCAACTATGCCATGAGCTGGGTCCGCCAGGCTCCAGGGA
 AGGGGCTGGAGTGGGTCTCAGGAATGAGTGGTAGTGGTGTAGCACATACTACGAAGACTCCGTGAAGGGCCGGTTCA
 CCATCTCCAGAGACAATCCAAGAACACGCTGTATCTGCAATGAATAGCCTGAGAGCCGAGGACACGGCCGTATATT
 ACTGTGCGCAAACCGATAGCAGCCACATAGATTTCTACTACTATGGTATGGACGACTGGGGCCAAGGGACCACGGTCA
 CCGTCTCCTCAGCATCCCCGACCAGCCCCAAGG >pesada_1d|FWMETYS01A36AT|0
 GCAGCAGCTACAGGTGTCAGTCCCAGGTGCAGTGGTGCAGTCTGGGGCTGAGGTGAAGAGGCCCTGGGTCTCGGTG
 AAGGTCTCCTGCAAGGCTTCTGGAGGCACCTTCAGCAACTATGCAATCAGCTGGGTGCGACAGGCCCTGGACAAGGG
 CTTGAGTGGATGGGAGGGATCATCCCTACTTTGGGACACCGACGTACGCACAGAAGTTCAGGCCAGAGTCACGATT
 ACCGCGGACGAATCTACGAGCACAGCCTACATGGAGCTGAGCAGCCTGAGATCTGAGGACACGGCCGTCTATTGGTGT
 GCGGTTCAAGATTGTAGTACTACCACCTGCTATCCTGCGAGTTCCTACTACTATAACATGGGCGTCTGGGGCAA
 GGGACCAGGTCACCGTCTCCTCAGCATCCCCGACCAGCCCCAAGG >pesada_1d|FWMETYS01BAHLT|2
 CAGCAGCTACAGGTGCCAGTCCCAGGTGCAGTGGTGCAGTCTGGGGCTGAGGTGAAGAGGCCCTGGGTCTCGGTGA
 AGGTCTCCTGCAAGGCTTCTGGAGGCACCTTCAGCAACTATGCTATCAGCTGGGTGCGACAGGCCCTGGACAAGGGC
 TTGAGTGGATGGGAGGGATCATCCCTATGTTTGGTACACCAAAGTACGCACAGCAGTTCCTGGACAGAGTCACGATAA
 CCGCGGACGAATCCACGAGTACAGCCTACATGGAGCTGAGCAGCCTGAGATCTGCGGACATGGCCGTTTATTGGTGTG
 CGGTTCAAGATTGTAGTACTACCACCTGTTATCCTGCGAGTTCCTACTACTACTACAACATGGGCGTCTGGGGCAAAG
 GGACCACGGTCACCGTCTCCTCAGCATCCCCGACCAGCCCCAAGG >pesada_1d|FWMETYS01COGNO|0
 GTGGCTATTTTAAAAGGTGTCCAGTGTGAGGTGCAGCTGTTGGAGTCTGGGGGAGGCTTGGTGCAGCCTGGGGGTCC
 CTGAGACTCTCCTGTGTAGCCTCTGGATTACCTTTAGCAACTATGCCATGAGCTGGGTCCGCCAGGCTCCAGGGAAG
 GGGCTGGAGTGGGTCTCAGGAATTAGTGGTAGTGGTGTAGCACATACTACGAAGACTCCGTGAAGGGCCGGTTCA
 ATCTCCAGAGACAATCCAAGAACACGCTGTATCTGGACATGAATAGCCTGAGAGCCGAGGACACGGCCGTATATTAC
 TGTGCGCAAACCGATAGCAGCCACATAGACTTCTACTACTACGGTATGGACGACTGGGGCCAAGGGACCACGGTCACC
 GTCTCCTCAGCATCCCCGACCAGCCCCAAGG >pesada_1d|FWMETYS01B1GQ7|2
 CGGCTCCAGATGGGTCTGTCCCAGGTGCAGTGCAGGAGTGGGGCCAGGACTGGTGAAGCCTTCGGAGACCCTGT
 CCTCACTTGCAGTGTCTTAGTGGCTCCATCAGGAATTACTACTGGAGCTGGATCCGGCAGACCCCAGGGAAGGGAC
 TGGAGTGGATTGGATAATCTTTTCCAATGGGAGGATCAAGTACAATTCCTCCCTCCAAGGTGCACTACCATGTAC
 TAAACACGCCCAGAAATCAGTTCCTCCTGTGGCTGAGCTCTGTGACCGCCGAGACACGGCCGTCTATTACTGTGCGA
 GATCGGGGATAGTTGGAGCCCTCCACAATTTGACTTCTGGGGCCAGGGAATCCTGGTACCGTCTCCTCAGCATCCC
 CGACCAGCCCCAAGG >pesada_21d|FWMETYS01ENDE7|2
 TTGTGGCTATTTTAAAAGGTGTCCAGTGTGAGGTGCAGTGTGTTGGAGTCTGGGGGAGGCTTGGTACAGCCTGGGGGT
 CCCTGAGACTCTCCTGTGCAGCCTCTGGATTACCTTTAACAACTATGCCACAAGCTGGGTCCGCCAGGCTCCAGGGA
 AGGGGCTGGAGTGGGTCTCAACTATTAGTGGTGTGGTGTATACCACTACTACGCAAATTCGTGAAGGGCCGGTTCA
 CCATCTCCAGAGACACTTCCAAGAACACCCCTGTATCTGCAATGAACAGCCTGAGAGCCGAGGACACGGCCGAATATT
 ACTGTGCGAGAGTACGGGGATATTGTAATGGTGGTAGCTGCTACTTTGACTACTGGGGCCAGGGAACCCCGGTACCCG
 TCTCCTCAGGGAGTGCATCCGCCCC >pesada_21d|FWMETYS01BI6DS|2
 CAGCAGCCACAGGAGCCCACTCCCAGGTGCAACTGGTGCAGTCTGGGGCTGAGGTGAAGAGGCCCTGGGGCCTCAGTGA
 AGGTCTCCTGCAAGGCCTCTGGATACACCTTACCAGGCTACTATGTACTACTGGGTGCGACAGGCCCTGGACAAGGTC
 TTGAGTGGATGGGATGGATCAACCTAACAGTGGTGTACAACTACGCACAGAACTTCAGGACAGGGTCACCATGA
 CCAGGGACACGTCATCAGCACAGCCTACATGGAGCTGACCAAGCTGAGATCTGACGACACGGCCCTATATTACTGTG
 CGAGAGAGAGGGCATTAGTGGGAGGTAGTACGACTCTCGGATACTGGGGCCAGGGAACCCCTGGTACCGTCTCCTCAG
 GGAGTGCATCCGCCCCAACCC >pesada_21d|FWMETYS01ASJFG|2
 CGGCTCCAGATGGGTCTGTCCCAGGTGCAGTTACAGCAGTGGGGCCAGGACTGTTGAAGCCTTCGGAGACCCTGT

CCCTCACCTGCGTTGTCTATGGTGGGTCTTTTCAGTCTTATTA TACTGGAGCTGGATCCGCCAGACCCAGGGAAGGGGC
 TGGAGTGGATTGGGGAAATCAATCATAGTGGAAGCACCAACTACAACCCGTCCCTCAAGAGTCGAGTCTCCATATCAC
 TAGACACGTCCAAGAATGAGTTCTCCCTGAGGCTGAACTCTCTGACCGCCGCGGACACGGCTGTGTATTACTGTGCGA
 GAGGCAGGGCTTCAACCTTTAAAGTCTACTATCACTACATGGACGCTCTGGGGCAATGGGACCACGGTCACCGTCTCTCT
 CAGCATCCCCGACCAGCCCCAAGG >pesada_21d|FWMETYS01EEVTL|2
 CGGCTCCAGATGGGTCTGTCCCAGGTGCAGCTGCAGGAGTCGGGCCAGGACTGGTGAAGCCTGCGGAGACCCTGT
 CCCTCACCTGCAATGTCTCTGGTGGCTCCATGAGTAGTTACTACTGGAGCTGGATCCGGCAGCCCCAGGGAAGGGAC
 TGGAGTGGATCGGGTACATCCACTACAGGGGGACCACCAAATACAATCCCTCCCTCAAGAGTCGCGTCACCATATCAA
 TAGACCTGTCCAAGAACCAGTTCTCCCTGAAACTGAGCTCTATGACCGCCGAGATACGGCCAGATATTACTGTGCGA
 GACATATGCGGGGTGGGAGCCCTAGTCAAACCTGCTTTTGATGTCTGGGGCCAAGGGACAATGGTCACCGTCTCTTCAG
 CATCCCCGACCAGCCCCAAGG : >pesada_21d|FWMETYS01BGHT7|2
 CGGCTCCAGATGGGTCTGTCCCAGGTGCAGCTGCAGGAGTCGGGCCAGGACTGGTGAAGCCTTCGGAGACCCTGT
 CCCTCACCTGCAATGTCTCTGGTGGCTCCATCAGTAGTTACTACTGGAGCTGGATCCGGCAGCCCCAGGGAAGGCAC
 TGGAGTGGATCGGGTATATCCACTACAGGGGGACTACCAAATACAATCCCTCCCTCAAGAGTCGCGTCACCATATCAG
 TAGACATGTCCAAGAACCAGTTCTCCCTGAAGCTGAGCTCTATGACCGCCGAGATACGGCCATCTATTACTGTGCGA
 GACATATGCGGGGTGGGAGTCTAGTCAAACCTGCTTTTGATGTCTGGGGCCAAGGGACAATGGTCACCGTCTCTTCAG
 CATCCCCGACCAGCCCCAAGG >pesada_21d|FWMETYS01DUYF3|2
 CGGCTCCAGATGGGTCTGTCCCAGGTGCAGCTGCAGGAGTCGGGCCAGGACTGGTGAAGCCTTCGGGGACCCTGT
 CCCTCACCTGCAATGTCTCTGGTGGCCCCATCAGTAGTTACTACTGGAGCTGGATCCGGCAGCCCCAGGGAAGGGAC
 TGGAGTGGATCGGGTATATCTATTACAGGGGGACTACCAAATACAATCCCTCCCTCAAGAGTCGCGTCACCATATCAG
 TAGACATGTCCAAGAACCAGTTCTCCCTGAACCTGAGCTCTATGACCGCCGAGATACGGCCATGTACTACTGTGCGA
 GACATATGCGGGGTGGGAGTCTAGTCAAACCTGCTTTTGATGTCTGGGGCCAAGGGACAATGGTCACCGTCTCTTCAG
 CATCCCCGACCAGCCCCAAGG >pesada_21d|FWMETYS01DI6U3|2
 TTGTGGCTATTTTAAAGGTGTCCAGTGTGAGGTGCAGCTGTTGGAGTCTGGGGGAGGCTTGATACAGCCTGGGGGGT
 CCCTGAGACTCTCCTGTGCAGCCTCTGGATTACCTTTAGCAAGTATGCCATGACCTGGGTCCGCCAGGCTCCAGGGA
 AGGGCCTGGAGTGGGTCTCGACTATTAGTGGTAGTGTACTGCCACATACTACGCAGACTCCGTGAAGGGCCGCTTCA
 CCATCTCCAGAGACAATTCCAAGAACACGCTGTATCTGCAAATGAACAGCCTGAGAGCCGCGGACACGGCCGTTTATT
 ACTGTGTGAAGCGGTTTCGGGGTGAACACTACATCTTTGACTACTGGGGCCAGGGAACCCAGGTACCGTCTCCTCAG
 CATCCCCGACCAGCCCCAAGG >pesada_21d|FWMETYS01D3L49|2
 TTGTGGCTATTTTAAAGGTGTCCAATGTGAGGTGCAACTGTTAGAATATGGGGGAGGCTTGTTACAGCCGGGGGGT
 CCCTGAGACTCTCCTGTGAAGCCTCTGGAATCCCTTTAACAACATATGCCATGAGCTGGGTCCGCCAGGCTCCAGGGA
 AGGGGCTGGAGTGGGTCTCAAGTATCAGTGGTAGTGGTAGTGGCACATATTACGGAGACTCCGTGAAGGGCCGTTCA
 CCATCTCCAGAGACAATTCCAAGAACACACTGTATCTGCAAATGAATAGCCTGAGAGCCGAGGACACGGCCGTATATT
 ATTGTGCGAGAGGCCGAGTGGGAGCTAAAGAGCCGACCGTTTACTACTTTGACCCTGGGGCCAGGGAACCCCTGGTCA
 CCGTCTCCTCAGCATCCCCGACCAGCCCCAAGG >pesada_21d|FWMETYS01EOI8H|0
 GTGGCTATTTTAAAGGTGTCCAGTGTGAGGTGCAGCTGTTGGAGTCTGGGGGAGGCTTGTTACAGCCTGGGGGGTCC
 TTGAGACTCTCCTGTGCAGCCTCTGGATTACCTTCAGCAGCTATGCCATGAACCTGGGTCCGCTGCCTCCAGGGATG
 GGGATGGAGAGCATCTCATCCATTAGTCGTAGTGGTATAGAACATACTACGCAGACTCCGTGAAGGGCCGTTTACC
 ATCTCCAGAGACAATTCCAAGAACACGATGTATCTGAAATGAACAGCCTGAGAGCCGAAGACACGGCCGTATATTAC
 TGTGCGAAAATTTTGGAGCGAATTTGGAAACTACTGGTATGGTTTGGACGCTCTGGGGCCAAGGGACCACGGTCACC
 GTCTCCTCAGGGAGTGATCCGCCCAACCC >pesada_21d|FWMETYS01BIG98|2
 CGGCTCCAGATGGGTCTGTCCCAGGTGCAGCTGCAGGAGTCGGGCCAGGACTGGTGAAGCCTTCGGAGACCCTGT
 CCCTCACCTGCGCTGTCTCTGGTGCCTCCATTAGAAGTTACTATTGGAGCTGGATCCGGCTGCCCCAGGGAAGGGAC
 TGGAGTGGATTGGGCATGTGTATCACAGTGGGAGCACAGTTACAATCCCTCCCTCAAGAGTCGAGTCACCATATCAG
 TGGACACGTCCAAGATGCAGATCTCCCTGAGGCTGAACTCTGCGACTGCTGCGGACACGGCCGTGTATTACTGTGCGA
 GACATCAGTATAATGTTGGTAACCTCTGGGCTTTTGATATCTGGGGCCAAGGGACAGTGGTCACCGTCTCTTCAGCAT

ES 2 730 951 T3

CCCCGACCAGCCCCAAGG >pesada_7d|FWMETYS01DQGF4|2
TTGTGGCTATTTAAAAGGTGTCCAGTGTGACGTGCAGCTGGTGGAGTCTGGGGGAGGCTTGGTACAGCCTGGGAAGT
CCCTGAGACTCTCCTGTGCAGCCTCTGGATTACCTTTAGCAACTATGCCATGTCTGGGTCCGCCAGGCTCCAGGGA
AGGGCCTGGAGTGGGTCTCAGGTATTAGTGGTGTGGTGTGCCACATACTACGCAGACTCCGTGAAGGGCCGGTTCA
CCATCTCCAGAGTCAATTCCAGGAACACGCCTCCTTCTGCAAATGAACAGCCTGAGAGCCGAGGACACGGCCCTATATT
ACTGTGCGCAAACCGATAGCAGCAACTTAGACTACTACTACCACGGAATGGGCGTCTGGGGCCAAAGGGACCACGGTCA
CCGTCTCCTCAGCATCCCCGACCAGCCCCAAGG >pesada_7d|FWMETYS01E08GA|2
CAGCAGCAACAGGTGCCCACTCCCAGGTTCAACTGGTGAATCTGGAGCTGAGGTGAAGAGCCTGGGGCCTCAGTAA
AGGTCTCCTGCACGGCTTCTGGTTACACGTTTGACACTTATGGAGTCAGCTGGTTGCGACAGGCCCTGGACAAGGGC
TTGAGTGGATGGGCTGGATCAGCGGTGACAGTAGTCATACCAGATATGCAATGAGACTCCAGGGCAGAGTCACCATGA
CCACAGACTCATCCACGAGCACAGCCTACATGGAACCTGAGGAGCCTGAGATCTGACGACACGGCCGTCTATTACTGTG
CGAGAGATGATCCATATTACGGCAGTATTGGTTATCGTATTGACTCCTGGGGCCAGGGAACCTGGTACACCGTCTCCT
CAGCATCCCCGACCAGCCCCAAGG >pesada_7d|FWMETYS02GPFII|2
CGGCTCCCAGATGGGTCTGTCCCAGGTGCAACTGCAGGAGTCGGGCCAGGACTGGTGAAGCCTTCGGAGACCCTGT
CCCTCACCTGCACTGTCTCTGGTGGCTCCATCAGTAGTTACTACTTGAGCTGGATCCGGCAGCCCCCAGGGAAGGGGC
TGGAGTGGATTGGGTATATCTATTACAGTGGGAGCACCAACTACAACCCCTCCCTCAAGAGTCGAGTCACCATATCAG
TAGACACGTCCAAGAACCAGTTCTCCCTGAGGCTGAGCTCTGTGACCGCTGCGGACACGGCCGTGTATTACTGTGCGA
GAATCCCCGTAAGATCGAGTGGGACGCCTACTACTACTACGGTATGGACGTCTGGGGCCAAAGGGACCACGGTCAACC
TCTCCTCAGGAGTGCATCCGCCCC >pesada_7d|FWMETYS01D3CAE|2
TTGTTGCTATTTTAAAGGTGTCCAGTGTGAGGTGCAGTTGGTGGAGTCTGGGGGAGGCTTGGTACAGCCTGGGGGT
CCCTGAGACTCTCCTGTGCAGCCTCTGGATTCACTTTCACTGGCTATAGCATGGACTGGGTCCGCCAGGCTCCGGGGA
AGGGGCTGGAGTGGGTTTCATTTCATAAGTGATAATACTGGCAGTCACATATACTACGCAGACTCTGTGAAGGGCCGAT
TCACCATCTCTAGAGACAATGCCGAGAACTCACTGTATCTACAAATGAACAGCCTGAGAGACGAGGACACGGCTGTGT
ATTACTGTGCGAGAGACCTTACCTGGAGATACTTTGACTCCTGGGGCCATGGAGTCTGGTACACCGTCTCCTCACGGA
GTGCATCCGCCCC >pesada_7d|FWMETYS01ASETU|2
TTGTTGCTATTTTAAAAGGTGTCCAGTGTGAGGTGCAGCTGGTGGAGTCTGGAGGAGGCCTGATCCAGCCTGGGGGT
CCCTGAGACTCTCCTGTGCAGCCTCTGGGTTACCCTCAGTAGCAGCTACATGACCTGGGTCCGCCAGGCTCCAGGGA
AGGGGCTGGAGTGGGTCTCAGTTATTTATAGTGGTGGTACCACATACTACGCAGACTCCGTGAAGGGCCGATTACCA
TCTCCAGAGACAATTCCAAGAACACGCCTGTATCTTCAAATGAACAGCCTGAGAGCCGAGGACACGGCCGTGTATTACT
GTGCGAGAGTGTCTCTAGTAGTGGTTATTACTACTACTTTGATTACTGGGGCCAGGGAACCTGGTACACCGTCTCCT
CAGGGAGTGCATCCGCCCC >pesada_7d|FWMETYS01EL10A|2
CGGCTCCCAGATGGGTCTGTCCCAGGTGCAGCTGCAGGAGTCGGGCCAGGACTGGTGAAGCCTTCGGAGACCCTGT
CCCTCACCTGCACTGTCTCTGGTGGCTCCATTAGTAGTTACTCCTGGAGCTGGATCCGGCAGTCCCCAGGGAAGGGAC
TGGAGTGGATTGGGTCTACCTATTACAGTGGGAGCACCAACTACCAACTACAACCCCTCCCTCAAGAGTCGAGTCA
CCACATCAGTAGACACGTCCAAGAACCAGTTGTCCCTGGGGCTGAACTCTGTGACCGCAGCGACACGGCCATTTATT
ACTGTGCAAGATCCCTCATTCTATATAGTACTACATTGCCTACTGGGGCCAGGGAACCTGGTACACCGTCTCCTCAG
CATCCCCGACCAGCCCCAAGG >pesada_7d|FWMETYS01CU1FY|2
CAGCAGCAACAGGTGCCCGCTCCCAGGTTCAACTGATGCAGTCTGGAGCTGAAGTGAAGGAAGCCTGGGGCCTCAGTGA
CGGTCTCCTGCAAGACTTCTGGTTACACCTTTACCTACTATGGTATCAGTTGGGTGCGACAGGCCCTGGACAAGGCC
TTGAGTGGATGGGATGGTTACAGCCTTACAATGGTAAAGACAAAATATGCACAGAATCTCCAGGACAGAGTCACCATGA
CAATTGACACATCCACGAGGACAGCCTACATGGAGCTGAGGAGCCTGAGATCTGACGACACGGCCGTATATTACTGTG
CGAGGCAGTCAGGTAACCGAGGATTCCGGTACTCTTACTCCTACTATTACTTCATGGACGTCTGGGGCAAAGGGACCA
CGGTACACCGTCGCCTCAGCATCCCCGACCAGCCCCAAGG >pesada_7d|FWMETYS01DFT1H|2
CGGCTCCCAGATGGGTCTGTCCCAGGTGCAGCTACAGCAGTGGGGCCAGGACTGTTGAAGCCTTCGGAGACCCTGT
CCCTCACCTGCGCTGTTTATGGTGAAGTCTTCCGTGGTTACTCCTGGAGCTGGATCCGGCAGCCCCCAGGGAAGGGGC
TGGAGTGGATTGGTGAATCAATCTTACTGGAAGCACCAACTACAACCCGTCCCTCAAGAGTCGGATCACCGTATCAA

ES 2 730 951 T3

TTGACCCGTCACCAAGACTCAGTTCTCCCTGAAGCTGACCTCTGTGACCGCCGCGGACACGGCTGTATATTACTGCGCGA
GAACTTTGTATTCTCTGGTAAAGTATAGTACTGGCTGGTACTACTTTGACTACTGGGGCCAGGGAATCCTGGTCACCG
TCTCCTCAGCCTCCACCAAGGGCCCATCCG >pesada_7d|FWMETYS01DE08R|2
CGGCTCCCAGATGGGTCTGTCCCAGGTGCAGCTGCAGGAGTCCGGGCCAGGACTGGTGAAGCCTTCGGGAGACTCTGT
CCCTCACCTGCAGTGTCTCTGGTACTCCATTAGTAGTACTATTGGAGCTGGATCCGGCAGTCCCAGGGAAGGGAC
TGGAGTGGATTGGGTTTATTTCAGTATACTGGGAGATCCCACTCCAACCCCTCCCTCCAGAGTCCAGTACCATATCAC
TAGACAGTCCAAGAACAACTTCTCCCTGAGGCTGACCTCTGTGACCGCTGCGGACACGGCCGTGTATTATTGTGCGA
GAACAAATGCTTTTCATATCTGGGGCCAAGGGACAATGGTCACCGTCTCTTCAGGGAGTGCATCCGCCCC
>pesada_7d|FWMETYS01BLJ3E|0
GTTGCTATATTAGAAGGTGTCAGTGTGAGGTGCAGCTGGTGAATCTGGGGGAGGCTTAGTACAACCTGGGGGTTCCG
CTGAGACTCTCTGTGTCAGCCTCTGGATTCACTTTTCAGTGATTACGACATGCACTGGGTCCGCCAAGTACAGGAAAA
GGTCTGGAGTGGGTCTCAGCTATAGGAACCTTGATGACACATACTATCCAGGCTCCGTGGAGGGCCGATTACCCTC
TCCAGAGACAATGCCAGGGATTCCCTTGTATCTTCAAATGAAGAGCCTCAGAGTCCGGACACGGCTGTATATTACTGT
GTAAGAGTTAAGGGTGGCATTAGCAGCAGCTGGTACCCTGCGGGTACTTCGATCTCTGGGGCCGTGGCACCCCTGGTC
ACTGTCTCCTCAGGGAGTGCATCCGCCCAACCC >ligera_1d|012171_0240_0412|2
TCTGGTCCCAGGTTCCAGATGCGACATCCAGATGACCCAGTCTCCGTCTTCCGTGTCTGCATCTGTGGGGACAGAG
TCACCATCACTTCCCGGGCAGTCCAGAGTCTTAGCGGCTTTTTCAGCCTGGTATCAGCAGAAACCAGGAAAGCCCTA
AGTTCTGATCGATACTACCTCCATTTTGCAAAGTGGGGTCCCATCTAGATTCACTGGCAGTGGATCTGGGACATTTT
TCACTCTCACCATCAGCAGCCTCCAGCCTGAAGATTTTGCAACTTACTATTGTCAACAGGCTAACAGTTTCCCGCTCA
CTTTCGGCGGAGGGACCAAGGTGGAGAGGAAACGAACCTGTGGCTGCACCATCTG
>ligera_1d|016824_0202_0719|2
CTCACTGCACAGGCTCTGAGGCTCCTATGAGCTGACACAGCCACCCTCGGTGTCACTGTCCCCAGGACAGACGGCCA
GGATCACCTGCTCTGGAGATGCATTGCCAAAGCACTTTGCTTATTGGTACCAGCAGAAGCCAGGCCAGGCCCTGTAC
TGGTGATATATAAAGACACTGAGAGGCCCTCAGGGATCCCTGAGCGATTCTCTGGCTCCAGCTCAGGGACAACAGTCA
CGTTGACCATCAGTGGAGTCCAGGCAGAAGACGAGGCTGACTATTACTGTCAATCACCAGACAGCAGTGGTACTAATG
AAGTCTTCGGAACCTGGGACCAAGGTACCGTCTTAAGTCAGCCCAAGGCCAACCCACGGT
>ligera_1d|025990_0147_0876|2
CTCACTGCACAGGCTCTGAGGCTCCTATGAGCTGACACAGCCACCCTCGGTGTCACTGTCCCCAGGACAGACGGCCA
GGATCACCTGCTCTGGAGATGCATTGCCAAAGCACTATGCTTATTGGTACCAGCAGAAGCCAGGCCAGGCCCTGTGT
TGGTGATATATAAAGACAGTGAAGGCCCTCAGGGATCCCTGAGCGATTCTCTGGCTCCAGCTCAGGGACAACAGTCA
CGTTGACCATCAGTGGAGTCCAGGCAGAAGACGAGGCTGACTATTACTGTCAATCACCAGACAGCAGTGGTACTTATG
AAGTCTTCGGAACCTGGGACCAAGGTACCGTCTTAGGTGAGCCCAAGGCCAACCCCACTGT
>ligera_1d|001721_0226_0129|2
CTCACTGCACAGGTCCTGGGCCAGTCTGTGTTGACGCAGCCGCCCTCAGTGTCTGCGGCCCCAGGACAGAAGGTCA
CCATCTCCTGCTCTGGAAGCAGCTCCAACATTGGGAGTAATCGTGTATCCTGGTACCAGCAGCTCCCAGGAACAGCCC
CCAAACTCCTCATCTATGAAAATGATGAGCGACCCTCAGGGATTCCCTGACCGATTCTCTGCCTCCAAGTCTGGCAGT
CAGCCACCCTGGGCATCACCAGGACTCCAGACTGGGGACGAGGCCACTTATTTCTGCGGAACATGGGATAACCAGCCTGG
GTACTAATTGGGTGTTTCGGCGGAGGGACCAAGCTGACCGTCTTAGGTGAGCCCAAGGCTGCCCCCTCGGT
>ligera_1d|035774_0389_0427|2
CTCACTGCACAGGTCCTGGGCCAGTCTGTGTTGACGCAGCCGCCCTCAGTGTCTGCGGCCCCAGGACAGAAGGTCA
CCATCTCCTGCTCTGGAAGCAGCTCCAACATTGGGAGTAATCGTGTATCCTGGTACCAGCAGCTCCCAGGAACAGCCC
CCAAACTCCTCATCTATGAAAATAATGAGCGACCCTCAGGGATTCCCTGACCGATTCTCTGCCTCCAAGTCTGGCAGT
CAGCCACCCTGGTCATCACCAGGACTCCAGACTGGGGACGAGGCCGATTATTATTGCGGAACATGGGATAACCAGCCTGA
GAATTAATTGGGIGTTTCGGCGGAGGGACCAAGCTGACCGTCTTAGGTGAGCCCAAGGCTGCCCCCTCGGT
>ligera_1d|007823_0201_0454|2
TCTGGCTCCCAGGTGCCAGATGTGACATCCTGTTGACCCAGTCTCCATCCTTCTGTCTGCAGCTGTAGGAAACAGAA

ES 2 730 951 T3

TCACCATTACTTGGCCGGGCCAGTCAGGGCATTAGTAGTTATTTAGCCTGGTTTCAGGAAAAACCAGGGAAAGCCCTA
AACTCCTGATTTATGGTGCATCCATTTTGC AAAGTGGGGTCCCATCAAGGTT CAGCGGCAGTGGATCTGGGACAGAGT
TCACTCTCACAATCAGGAGCCTGCAGCCTGAAGATTTTGCAACTTATTACTGTCAACAGTTTACTAATTTCCCGCTCA
CTTTCCGGCGGAGGGACCAAGGTGGAGATCAAACGAACTGTGGCTGCACCATCTG

>ligera_1d|008086_0088_0336|1

ACTCTGGCTCACAGATAACCACCGGAGAAATTTGTGTTGACGCAGTCTCCAGGCACCCTGTCTTTGTCTCCAGGGGAAAG
AGCCACCCTCTCCTGCAGGGCCAGTCAGAGAGTTAGCAGCAACTACTTAGCCTGGTACCAGCAGAAATCAGGCCAGGC
TCCCAGGCTCCTCATCTATAGTGCATCCCGCAGGGCCACTGGCATCCCAGACAGGTTTCAGTGGCAGTGGGTCTGGGAC
AGACTTCACTCTCACCATCAGCAGACTGGAGCCTGAAGATTTTGCCGTGTACTGTCAACAGTATGGTAGTTTCATG
GCGCACTTTTGGCCAGGGACCAAGGTGGAGATCAGACGAACTGTGGCTGCACCATCTG

>ligera_1d|004730_0220_0168|2

CTTACTGCACAGGATCCGTGGCCTCCTATGAGCTGACTCAGCCACCCTCAGTGTCCGTGTCCCCAGGACAGACAGCCA
GCATCACCTGCTCTGGAGATAAAATTTGGGGATAAAATATGCTTGCTGGTATCAGCAGAAGCCAGGCCAGTCCCCCTGTGC
TGGTCACTATCAAGATAGCAAGCGGCCCTCAGGGATCCCTGAGCGATTCTCTGGCTCCAACCTCTGGGAAACACAGCCA
CTCTGACCATCAGCGGGACCCAGGCTATGGATGAGGCTGACTATTACTGT CAGGCGTGGGACAGCAGCACTGTGGTAT
TCGGCGGAGGGACCAAGCTGACCGTCTTAGGTCAGCCCAAGGCTGCCCCCTCGGT

>ligera_1d|018024_0239_0556|2

CTCACTGCACAGGGTCTGGGCCAGTCTGTGTTGACGCAGCCGCCCTCAGTGTCTGCGGCCCTCAGGACAGGAGGTCA
CCATCTCCTGCTCTGGAAGCAGCTCCAACATTTGGGAGTTATCGTGTATCCTGGTATCAGCACCTCCCAGGAACAGCCC
CCAAACTCCTCATCTATGAAAATGATCAGCGACCCTCAGGGATTCTCTGACCGATTCTCTGGCTCCAAGTCTGGCAGCT
CAGCCACCCTGGTCATCACCAGACTCCTGACTGCGGACGAGGCCGATTATTACTGCGGAAACATGGGATACCAGCCTGA
GGATTAATTGGGTGTTTCGGCGGAGGGACCAAGCTGACCGTCTTAGGTCAGCCCAAGGCTGCCCCCTCGGT

>ligera_1d|003754_0225_0116|1

CACTCACTCTGCAGTGTGAGTGGTCCAGGCAGGACTGACTCAGCCACCCTCGGTGTCCAAGGACTTGAGACAGACCCGC
CACACTCACCTGCAGTGGGAACAGCAACATGTTGGCAACCTAGGAGCAGCTTGGCTGCAGCAGCACCAGGGCCACCC
TCCCACACTCCTATCCTACAGGGATAACAACCGGCCCTCAGGGATCTCAGAGAGATTCTCTGCATCCAGGT CAGGAAA
TACAGCTCCTGTCCATTACTGGACTCCAGCCTGACGACGAGGCTGACTATTACTGTCTCAGCATGGGACAGCAGCCT
CAGTGTCTGGGTGTTTCGGCGGAGGGACCAAGCTGACCGTCTTAGGTCAGCCCAAGGCTGCCCCCTCGGT

>ligera_21d|004730_0220_0168|2

CTTACTGCACAGGATCCGTGGCCTCCTATGAGCTGACTCAGCCACCCTCAGTGTCCGTGTCCCCAGGACAGACAGCCA
GCATCACCTGCTCTGGAGATAAAATTTGGGGATAAAATATGCTTGCTGGTATCAGCAGAAGCCAGGCCAGTCCCCCTGTGC
TGGTCACTATCAAGATAGCAAGCGGCCCTCAGGGATCCCTGAGCGATTCTCTGGCTCCAACCTCTGGGAAACACAGCCA
CTCTGACCATCAGCGGGACCCAGGCTATGGATGAGGCTGACTATTACTGT CAGGCGTGGGACAGCAGCACTGTGGTAT
TCGGCGGAGGGACCAAGCTGACCGTCTTAGGTCAGCCCAAGGCTGCCCCCTCGGT

>ligera_21d|010947_0053_0325|1

ACTCTGGCTCACAGATAACCACCGGAGAGATTTGTGTTGACGCAGTCTCCAAGCACCCCTGTCTTTCTCTCCAGGAGAAAG
AGCCACCCTCTCCTGCAGGGCCAGTCAGATTATTACCAGCAGCTACTTAGCCTGGTACCAGCAGAAACCTGGCCAGGC
TCCCAGGCTCCTCATCTATGGTGGATCCAGCAGGGCCACTGGCATCCCAAACAGGTTTAGTGGCAGTGGGGCTGGGAC
AGACTTCAAGCTCACCATCAGCAGACTGGAGCCTGAAGATTTTGAGTGTATTACTGTCAACAACATGGTAACTCACC
GGTCACTTTTCGGCGGAGGGACCAAGGTGGAGATCAAAGGAACTGTGGCTGCACCATCTG

>ligera_21d|016345_0101_0575|2

CTTGCTGCCAGGGTCCAATTCACAGGCTGTGGTACTCAGGAGCCCTCACTGACTGTGTCCCCAGGAGGGACAGTCA
CTCTCACCTGTGGCTCCAGCACTGGAGCTGTCACCAGTGGTCAATATCCCTACTGGTTCCAGCAGAAGCCTGGCCAAG
CCCCAGGACACTGATTTCTGATACAAGCAACAAATGTTCTTGGACCCTGGCCGGTTCTCAGGCTCCTCCTTGGGG
GCAAAGCTGCCCTGACCCTTTCCGGGTGCGCAGCCTGAGGATGAGGCTGACTATTATTGCTTCTCTACTTTGATGGTC
CTATAGTTTTTCGGCGGAGGGACCAAGCTGACCGTCTTAAGTCAGCCCAAGGCTGCCCCCTCGGT

ES 2 730 951 T3

>ligera_21d|016683_0138_0575|0
CTCTGGATCTCTGGTGCCTACGGGGACATCGTGATGACCCAGTCTCCAGATTCCCTGGCTGTGTCTCTGGGCGAGAGG
GCCACCATCAACTGCAAGTCCAGCCAGAGTGTTTTGTGGGGCCCCAACAAATAAGAACTACTTAGCTTGGTACCAGCAG
AAACCAGGACAGTCTCCTAAGTTGCTCATTACTGGGGATCTACCCGGAAATCCGGGGTCCCTGACCGATTTCAGTGGC
AGCGGGTCTGGGACAGATTTCACTCTCACCATCAGCAGCCTGCAGGCTGAAGATGTGGCAGTTTATTACTGTCCAGCAT
TATCATACTCCACCGTACACTTTTGGCCAGGGGACCAAGCTGGAGATCAAACGAACTGTGGCTGCACCATCTG
>ligera_21d|014243_0165_0699|2
TCTGGGTCCCAGGTGCCAAATGTGTCTGTCAGATGACCCAGTCTCCTTCCACCCTGTCTGCATCTGTAGGAGACAGAG
TCACCATCACTTGCCGGGCCAGTGAGACTGTTGGAACGTGGTTGGCTGGTATCGGCAGAAACCAGGGAAAGCCCCCTA
ACCTCCTGATCTATGAGGCCTCTATTTTAGAAAGTGGGGTCCCATCGAGGTTCCAGCGGCAGTGGATCTGGGACAGAGT
TCACTCTCACCATCAGCAGCCTGCAGCCTGATGATTTTGAACCTTATTAAGTCCAAACAGTATAAATACCTGGTGGACAT
TCGGCCAAGGGACCAAGGTGAAATCAAGCGAACTGTGGCTGCACCATCTG
>ligera_21d|007238_0089_0294|2
CTTGCTGCCAGGGTCCAATTCCCAGGCTGTGGTGACTCAGGAGCCCTCACTGACTGTGTCCCAGGAGGGACAGTCA
CTCTCACCTGTGGCTCCAGCACTGGAGCTGTCCAGTGGTCAATATCCCTACTGGTTTCAGCAGAAGGCTGGCCAAG
CCCCCAGGACACTGATGTATGATAAAGCATCAAAGTGTCTGGACCCCTGCCCGGTTCTCAGGCGCCCTCCTTGGGG
GCAAAGCTGCCCTGACCCCTTTCGGGTGCGCAGCCTGAGGATGAGGCTGAATATTATTGTTTACTCTATTATAATGGTG
TCAGGGTGTTCGGCGGAGGGACCAACTGACAGTCTTAGGTTCAGCCCAAGGCTGCCCCCTCGGT
>ligera_21d|015152_0165_0715|1
CCTCACTCTTTGCATAGGTTCTGTGGTTCTTCTGAGCTGACTCAGGACCCTGTGTGTCTGTGGCCTTGGGACAGAC
AGTCAGGATCACATGCCAAGGAGACAGCCTCAGAAGCTATATGCAAGCTGGTACCAGCAGAAGCCAGGACAGGCCCC
TGTACTTGTCTATGTTAGTAAAAACAACCGGCCTCAGGGATCCCAGACCGATTCTCTGGCTCCAGCTCAGGAAACAC
AGCTTCCCTTGACCATCACTGGGGCTCAGGCGGAAGATGAGGCTGACTATTACTGTAAGTCCCGGACAGCAGTGGTAA
CCATGTGTATTTCGGCGGAGGGACCAAGCTGACCGTCTTAGGTTCAGCCCAAGGCTGCCCCCTCGGT
>ligera_21d|003565_0121_0193|2
CTCGGGACACAGGGTCTTGGGCCAGTCTGCCCTGACTCAGCCTGCCTCCGTGTCTGGGTCTCTGGACAGTCCGATCA
CCATCTCCTGCACTGGAAACCAGCAGTGTGTTGGGAGATATAACCTTGTCTCCTGGTACCAACAACACCCAGGCAAAG
CCCCCAAATTTCTGATTTATGAGGTCAGTAAGGGGCCCTCAGGGGTTCTAATCGCTTCTCTGGCTCCAAGTCTGGCA
ACACGGCCTCCCTGACAACTCTCTGGGCTCCAGGCTGAGGACGAGGCAGATTATTACTGCTGCTCATATGCAGGTAGTA
GCACTTTTGTCTTCGGAAGTGGGACCACGGTCCCGTCTTAGGTTCAGCCCAAGGCAACCCACCGT
>ligera_21d|010775_0076_0480|2
CTCACTGCACAGGCTCTGAGGCCTCCTATGAGCTGACACAGCCACCCTCGGTGTCTAGTGTCCCAGGACAGACGGCCA
GGATCACCTGTCTGGAGATGCATTGCCAAAGCAATATGCTTATTGGTACCAGCAGAAGCCAGGCCAGGCCCTGTGC
TGGTGATATATAAAGACAGTGTAGAGGCCCTCAGGGATCCCTGAGCGATTCTCTGGCTCCAGCTCAGGGACAACAGTCA
CGTTGACCATCAGTGGAGTCCAGGCAGAAGACGAGGCTGACTATTACTGTCAATCAGCAGACAGCAGTGGTACTTATG
TGGTATTTCGGCGGAGGGACCAAGCTGACCGTCTTAGGTTCAGCCCAAGGCTGCCCCCTCGGT
>ligera_21d|003754_0225_0116|1
CACTCACTCTGCAGTGTCTAGTGGTCCAGGCAGGACTGACTCAGCCACCCTCGGTGTCCAAGGACTTGTAGACAGACCGC
CACACTCACCTGCCTGGGAACAGCAACAATGTTGGCAACCTAGGAGCAGCTTGGCTGCAGCAGCACACAGGCCACCC
TCCCACACTCCTATCCTACAGGGATAACAACCGGCCCTCAGGGATCTCAGAGAGATTCTCTGCATCCAGGTTCAGGAAA
TACAGCCTCCCTGTCCATTAAGTGGACTCCAGCCTGACGACGAGGCTGACTATTACTGCTCAGCATGGGACAGCAGCCT
CAGTGTCTTGGGTGTTTCGGCGGAGGGACCAAGCTGACCGTCTTAGGTTCAGCCCAAGGCTGCCCCCTCGGT
>ligera_7d|004730_0220_0168|2
CTTACTGCACAGGATCCGTGGCCTCCTATGAGCTGACTCAGCCACCCTCAGTGTCCGTGTCCCAGGACAGACAGCCA
GCATCACCTGTCTGGAGATAAATTTGGGGATAAATATGCTTGGTATCAGCAGAAGCCAGGCCAGTCCCTGTGC
TGGTCACTATCAAGATAGCAAGCGGCCCTCAGGGATCCCTGAGCGATTCTCTGGCTCCAACCTTGGGAACACAGCCA

ES 2 730 951 T3

CTCTGACCATCAGCGGGACCCAGGCTATGGATGAGGCTGACTATTACTGTCCAGGCGTGGGACAGCAGCACTGTGGTAT
TCGGCGGAGGGACCAAGCTGACCGTCTTAGGTCAGCCCAAGGCTGCCCCCTCGGT

>ligera_7d|016513_0160_0702|2

CTCAGGGCACAGGGTCTGGGCTCAGTCTGCCTGACTCAGCCTCGCTCAGTGTCCGGGTCTCCTGGACAGTCAGTCA
CCATCTCCTGCACTGGAACCAGCAGTGTGTTGGTAGTTATAACTATGTCTCCTGGTACCAACAGCAGCCAGGCAAAG
CCCCCACTCATGATTTATGATGTCACTAAGCGGCCCTCAGGGTCCCTGATCGCTTCTCTGGTTCCAAGTCTGGCA
ACACGGCCTCCCTGACCATCTCTGGGCTCCAGGCTGAGGATGAGGCTGATTATTACTGTCTCATATGCAGGCAACT
CTTATGTCTTCGGAACCTGGGACCAAGGTCACCGTCTTAGGTCAGCCCAAGGCCAACCCACGGT

>ligera_7d|016824_0202_0719|2

CTCACTGCACAGGCTCTGAGGCTCCTATGAGCTGACACAGCCACCCTCGGTGTCTCAGTGTCCCCAGGACAGACGGCCA
GGATCACCTGTCTGGAGATGCATTGCCAAAGCACTTGTCTTATTGGTACCAGCAGAAGCCAGGCCAGGCCCTGTAC
TGGTGATATATAAAGACACTGAGAGGCCCTCAGGGATCCCTGAGCGATTCTCTGGCTCCAGCTCAGGGACAACAGTCA
CGTTGACCATCAGTGGAGTCCAGGCAGAAGACAGGCTGACTATTACTGTCAATCACCAGACAGCAGTGGTACTAATG
AAGTCTTCGGAACCTGGGACCAAGGTCACCGTCTTAAGTCAGCCCAAGGCCAACCCACGGT

>ligera_7d|076796_0530_0700|1

ACTCTGGCTCACAGATACCACCGGAGAAATGTGTTGACGCAGTCTCCAGGCACCCTGTCTTTGTCTCCAGGGGAAAG
AGCCACCCTCTCCTGCAGGGCCAGTCAGAGTGTAGCAGCAGCTACTTAGCCTGGTACCAGCAGAAACCTGGCCAGGC
TCCCAGGCTCCTCATCTATGGTGCATCCAGCAGGGCCACTGGCATCCCAGACAGGTTCAAGTGGCAGTGGGTCTGGGAC
AGACTTCACTCTCACCATCAGCAGACTGGAGCCTGAAGATTTGTCAGTGTATTACTGTCTCAGCAGTATGGTAGCTCACC
TTGGACGTTTCGGCCAAAGGACCAAGGTGGAAATCAAACGAACTGTGGCTGCACCATCTG

>ligera_7d|006092_0191_0398|1

ACTCTGGCTCACAGATACCACCGGAGAAATGTGTTGACGCAGTCTCCAGGCACCCTGTCTTTGTCTCCAGGGGAAAG
AGCCTTCTCTCCTGCAGGGCCAGTCAGACTGTTCCAGCAGCTACTTAGCCTGGTACCAGCAGAGACCTGGCCAAGT
TCCCAGGCTCCTCATCTATGATGCATCCAGCAGGGCCACTGGCATCTCAGACAGGTTTAGTGGCAGTGGGTCTGGGAC
AGACTTCACTCTCACCATCAACACACTGGAGCCTGAAGATTTCTGCTGTATTACTGTCTCAGCAATATGGTAGCTCACC
GACGTTTCGGCCAAAGGACCAAGGTGGAAATCAAACGAACTGTGGCTGCACCATCTG

>ligera_7d|015152_0165_0715|1

CCTCACTCTTTGCATAGGTTCTGTGGTTTCTTCTGAGCTGACTCAGGACCCTGTGTGTCTGTGGCCTTGGGACAGAC
AGTCAGGATCACATGCCAAGGAGACAGCCTCAGAAGCTATTATGCAAGCTGGTACCAGCAGAAGCCAGGACAGGCCCC
TGTACTTGTCTATGTTGTAACAAACCGGCCCTCAGGGATCCCAGACCGATTCTCTGGCTCCAGCTCAGGAAACAC
AGCTTCTTGCATCACTGGGGCTCAGGCGGAAGATGAGGCTGACTATTACTGTAACTCCCGGGACAGCAGTGGTAA
CCATGTGGTATTTCGGCGGAGGACCAAGCTGACCGTCTTAGGTCAGCCCAAGGCTGCCCCCTCGGT

>ligera_7d|037667_0422_0484|1

ACTCTGGCTCACAGATACCACCGGAGAAATGTGTTGACACAGTCTCCAGCCACCCTGTCTTTGTCTCCAGGGGAAAG
AGCCACCCTCTCCTGCAGGGCCAGTCAGAGTGTAGCACCTACTTAGCCTGGTACCAACAGAACTGGCCAGCCTCC
CAGACTCCTCATGATGCATCCAACAGGGCCACTGGCATCCCAGCCAGGTTCAAGTGGCAGTGGGTCTGGGACAGA
CTTCACTCTCACCATCAGCAGCCTAGAGCCTGAAGATTTCTGAGTTTTATTACTGTCTCAGCAGCTAGCACCTGGCCTGC
GACTTTCGGCCCTGGGACCAAGTGGATATCAAACGAACTGTGGCTGCACCATCTG

>ligera_7d|059523_0412_0968|2

CTCACTGCACAGGCTCTGTGACCTCCTATGTGCTGACTCAGCCACCCTCGGTGTCTCAGTGGCCCCAGGAAAGACGGCCA
GGATTACCTGTGGGGGAAACAACATTGGAAGTAAAAGTGTGCACTGGTACCAGCAGAAGCCAGGCCAGGCCCTGTGC
TGGTCTGTATGATGATAGCGACCGGCCCTCAGGGATCCCTGAGCGATTCTCTGGCTCCAACCTCTGGGAAACAGGCCA
CCCTGACCATCAGCAGGCTCGAAGCCGGGGATGAGGCCGACTATTACTGTCTCAGGTTGGGATAGTAGTAGTATCATG
TGGTATTTCGGCGGAGGACCAAGCTGACCGTCTTAGGTCAGCCCAAGGCTGCCCCCTCGGT

>ligera_7d|010775_0076_0480|2

CTCACTGCACAGGCTCTGAGGCTCCTATGAGCTGACACAGCCACCCTCGGTGTCTCAGTGTCCCCAGGACAGACGGCCA

```

GGATCACCTGCTCTGGAGATGCATTGCCAAAGCAATATGCTTATTGGTACCAGCAGAAGCCAGGCCAGGCCCTGTGC
TGGTGATATATAAAGACAGTGAGAGGCCCTCAGGGATCCCTGAGCGATTCTCTGGCTCCAGCTCAGGGACAACAGTCA
CGTTGACCATCAGTGGAGTCCAGGCAGAAGACGAGGCTGACTATTACTGTCAATCAGCAGACAGCAGTGGTACTTATG
TGGTATTCGGCGGAGGGACCAAGCTGACCGTCTTAGGTCAGCCCAAGGCTGCCCCCTCGGT
>ligera_7d|030140_0479_0232|2
CTCACTGCACAGGGTCTGGGCCAGTCTGTATTGACGCAGCCGCCCTCAGTGTCTGCAGCCCCAGGACAGAAGGTCA
CCATCTCCTGCTCTGGAAGCACCTCCAACATAGTTTCATAATTTTGTATCGTGGTTCCAGCATCTCCAGGAACAGCCC
CCAAACTTCTCATATATGACAATAAGAGGCGGCCCTCAGGGATTCTGACCGATTCTCTGGCTCCAAGTCTGGCGCGT
CAGCCACCCTGGACATCACTGGACTCCAGACTGGGGACGAGGCCGATTATTACTGCGGAGCGTGGGATAGCAGCCTGA
GTGCTGTGGTCTTCGGCGGAGGGACCAAGCTGACCGTCTGGGTGAGCCCAAGGCTGCCCCCTCGGT
    
```

Tabla 6

Los clones más altamente expresados se combinaron manualmente con los clones originales. Dos de ellos no coinciden perfectamente. Los nombres de las secuencias incluyen el marco de lectura (último campo; 0, 1 o 2)

- 5 Para hacer coincidir las cadenas pesadas sintetizadas originales identificadas utilizando clones altamente expresados con las nuevas cadenas pesadas identificadas mediante uniones altamente expresadas, ambos conjuntos de secuencias y se analizaron con alineación de secuencias múltiples (CLUSTAL) para determinar cómo se emparejaron. En su mayor parte, tenían pares de secuencias idénticos, lo que fue una verificación consistente de que los dos métodos produjeron las mismas cadenas pesadas. En la Figura 50 se presenta una nueva trama de series de tiempo altamente expresadas para uniones de cadenas pesadas y livianas.

Ejemplo VIII

Secuenciación inmune de cadena pesada

Las muestras se concentraron usando el kit RNeasy de Qiagen, se eluyó en 2x 30 µl y se extrajo con Nanodrop (consulte la Tabla 7 a continuación)

15 Tabla 7

VDJ-ID	MID	Muestra de vacunación	Fecha	Reserva de concentración de ARN (ng/µl) antes de la digestión con DNasa	Volumen para 8 µg Digestión con DNasa	Agua hasta 150 µl
1	GMC 1	-8 día	Dic 7 2009	91.71	87.23	62.77
2	GMC 2	-2 día	Dic 13 2009	75.6	105.82	44.18
3	GMC 3	-1 hora	Dic 15 2009	89.51	89.38	60.62
4	GMC 4	+1 hora	Dic 15 2009	96.87	82.58	67.42
5	GMC 5	+1 día	Dic 16 2009	117.3	68.20	81.80
6	GMC 6	+3 día	Dic 18 2009	90.22	88.67	61.33
7	GMC 7	+7 día	Dic 22 2009	87.45	91.48	58.52

ES 2 730 951 T3

VDJ-ID	MID	Muestra de vacunación	Fecha	Reserva de concentración de ARN (ng/μl) antes de la digestión con DNasa	Volumen para 8 μg Digestión con DNasa	Agua hasta 150 μl
8	GMC 8	+14 día	Díc 29 2009	131.12	61.01	88.99
9	GMC 9	+21 día	Ene 5 2010	127.17	62.91	87.09
10	GMC 10	+28 día	Ene 12 2010	139.24	57.45	92.55
11	IDO 1	-8 día	Dic 7 2009	98.84	80.94	69.06
12	IDO 2	-2 día	Dic 13 2009	84.45	94.73	55.27
13	IDO 3	-1 hora	Dic 15 2009	72.56	110.25	39.75
14	IDO 4	+1 hora	Dic 15 2009	78.14	102.38	47.62
15	IDO 5	+1 día	Dic 16 2009	51	156.86	-6.86
16	IDO 6	+3 día	Dic 18 2009	45.22	176.91	-26.91
17	IDO 7	+7 día	Dic 22 2009	101.42	78.88	71.12
18	IDO 8	+14 día	Dic 29 2009	109.09	73.33	76.67
19	IDO 9	+21 día	Ene 5 2010	127.49	62.75	87.25
20	IDO 10	+28 día	Ene 12 2010	89.01	89.88	60.12
21	FV 1	-8 día	Dic 7 2009	96.87	82.58	67.42
22	FV 2	-2 día	Dic 13 2009	127.13	62.93	87.07
23	FV 3	-1 hora	Dic 15 2009	115.25	69.41	80.59
24	FV 4	+1 hora	Dic 15 2009	136.05	58.80	91.20

ES 2 730 951 T3

VDJ-ID	MID	Muestra de vacunación	Fecha	Reserva de concentración de ARN (ng/μl) antes de la digestión con DNasa	Volumen para 8 μg Digestión con DNasa	Agua hasta 150 μl
25	FV 5	+1 día	Dic 16 2009	142.15	56.28	93.72
26	FV 6	+3 día	Dic 18 2009	113.49	70.49	79.51
27	FV 7	+7 día	Dic 22 2009	91.81	87.14	62.86
28	FV 8	+14 día	Dic 29 2009	84.92	94.21	55.79
29	FV 9	+21 día	Ene 5 2010	131.06	61.04	88.96
30	FV 10	+28 día	Ene 12 2010	90.98	87.93	62.07

Vacunación y extracción de sangre

Información de la vacuna: Gripe estacional 2010. Se extrajeron muestras de 9 mL de sangre de un solo individuo en diversos puntos de tiempo y se procesaron con el kit Leucolock (protocolo alternativo). El flujo de sangre se centrifugó y el plasma se guardó y almacenó a -80°C. La fracción de glóbulos rojos se guardó y se entrecruzó en formaldehído y se almacenó a 4°C.

5

Extracción de ARN

Se extrajo el ARN utilizando el protocolo alternativo de Leucolock (trizol) con las siguientes modificaciones: Se llevó a cabo el protocolo de extracción de ARN total que incluía ARN pequeños. Solo se aspira el primer paso de unión, para cada lavado, centrifugado a un máximo de 10.000 x G. Se eluye con 250 μl de dH₂O a 85°C, centrifugado a la velocidad máxima. Se transfiere el ARN en solución sin la pella blanca (probablemente parte del filtro). El ARN se sometió a Nanodrop, y se almacenó a -80°C.

10

Promedio	ng/μl	260/280	260/230	A260	A280
	100.95	2.09	1.99	2.52	1.20

Patrón de ARN protegido (llevar a Tris 10 mM, EDTA 0.1 mM). A los 250 μl de ARN, se agregó lo siguiente:

	1 tubo	50 tubos
• 1M Tris-HCl	2.5 μl	125 μl
• 50 mM EDTA	0.5 μl	25 μl

(3 μl/tubos)

15 Digerir 8 μg con DNasa y concentrado

Digestión con DNasa ejecutada utilizando Ambion TURBO libre de ADN (AM1907) con la siguiente modificación. Se añadió 0.1 mL (15 μl) de regulador de ADNasa TURBO 10X y 1 μL de ADNasa TURBO al ARN, y se mezcló suavemente. Se incubó a 37°C durante 20-30 min. Se agregó reactivo de desactivación de la ADNasa resuspendido

ES 2 730 951 T3

(típicamente 0.1 volumen) (15 µl) y se mezcló bien. Se incubó por 5 min a temperatura ambiente, mezclando ocasionalmente. Se centrifugó a 10.000 x g durante 5 min y se transfirió el ARN a un tubo nuevo. Realización de fenol ácido y extracción con CHCl₃, seguido de precipitación con etanol (EtOH) con NaOAc y Glicoblue. Se eluyó en 50 µl de Tris 10 mM EDTA 0.1 mM, QC y se almacenó a -80°C. Esto se repitió con 5 µg para las muestras 5 y 22 porque el ARN era demasiado bajo (resuspender en 25 µl).

ARN tratado con QC DNasa. Cuantificación con Nanodrop (véase tabla 8 a continuación)

Tabla 8

ID	Muestra ID	ng/ul	260/280	260/230	para 500 ng	Agua a 7 µl
1	GMC 1	136.32	2.09	1.95	3.67	3.33
2	GMC 2	132.63	2.12	1.98	3.77	3.23
3	GMC 3	95.13	2.07	1.85	5.26	1.74
4	GMC 4	96.39	2.08	1.89	5.19	1.81
5	GMC 5	91.17	2	1.61	5.48	1.52
6	GMC 6	131.56	2.03	1.89	3.80	3.20
7	GMC 7	138.75	2.1	1.99	3.60	3.40
8	GMC 8	120.75	2.09	1.92	4.14	2.86
9	GMC 9	82.29	2.05	1.92	6.08	0.92
10	GMC 10	87.76	2.08	1.85	5.70	1.30
11	IDO 1	111.26	2.12	1.9	4.49	2.51
12	IDO 2	119.51	2.11	1.9	4.18	2.82
13	IDO 3	203.4	2.12	2.06	2.46	4.54
14	IDO 4	190.66	2.11	2.04	2.62	4.38
15	IDO 5	308.04	2.12	2.09	1.62	5.38
16	IDO 6	311.67	2.13	2.12	1.60	5.40
17	IDO 7	137.57	2.11	1.97	3.63	3.37
18	IDO 8	155.92	2.1	1.99	3.21	3.79
19	IDO 9	121.76	2.12	1.99	4.11	2.89
20	IDO 10	159.76	2.11	1.95	3.13	3.87
21	FV 1	112.87	2.09	1.91	4.43	2.57

ES 2 730 951 T3

ID	Muestra ID	ng/ul	260/280	260/230	para 500 ng	Agua a 7 µl
22	FV 2	83.53	2.03	1.67	5.99	1.01
23	FV 3	95.56	2.08	1.87	5.23	1.77
24	FV 4	84.47	2.1	1.85	5.92	1.08
25	FV 5	95.18	2.12	1.9	5.25	1.75
26	FV 6	100.07	2.11	1.9	5.00	2.00
27	FV 7	113.7	2.1	1.89	4.40	2.60
28	FV 8	182.87	2.12	2.05	2.73	4.27
29	FV 9	86.99	2.09	1.83	5.75	1.25
30	FV 10	128.91	2.11	1.91	3.88	3.12
promedio	133.88	2.09	1.92			

Mezcla de cebador (véase tabla 9 a continuación).

Tabla 9.

Tipo de cebador	Fecha de la siembra	Patrón de siembra	Ubicación del pozo	µl para cada cebador	µl H ₂ O	Mezcla final
Cebador 5 x J IGH RT-PCR	20080707					2 µM
Cebador 6 x J IGH PCR	20080924	200 µM	Plato 1 A1-D7	20 µl cada uno (100 µl total)	880 µl	4 µM (1000 µl)
Cebador 43x V IGH PCR	20080924	200 µM	Plato 3 A1-A6	20 µl cada uno (860 µl total)	140 µl	4 µM (1000 µl)

5 Transcripción reversa

Se preparó una reacción de RT de la siguiente manera:

	1 tubo	35 tubos
• H ₂ O (DEPC)	0 µl(a 12 µl)	-
• 2 µM de cebador específico de gen (mezcla IGHC antigua)	5 µl (5 pmol)	
• Total ARN	7 µl (750 ng)	-

Se calentó a 95°C durante 1 min. seguido de 65°C durante 5 min e hielo durante 1 min.

ES 2 730 951 T3

Se centrifugó y añadió lo siguiente:

	1 tubo	35 tubos
• 5x regulador de primera hebra	4 µl	140
• mezcla dNTP 10 mM	1 µl	35
• 0.1 M DTT	1 µl	35
• RNAsalnh (Enzy)	1 µl	35
• Superscript III	1 µl	35
Se incubó a 55°C durante 60 min		(8 µl/tubos)

Enzima inactivada por calentamiento a 70°C durante 15 min. Se eliminó el híbrido ARN/ADN agregando 1 µl de E. coli RNasaH. Se incubó a 37°C durante 20 min, luego hielo.

- 5 Se hizo una mezcla maestra global, luego se dividió en todas las muestras (3x 10 veces el punto de 200 µl), luego se dividió cada muestra en 4 x. Se utilizó la mitad para la PCR y mantuvo la otra mitad como respaldo. Se preparó una reacción de prueba de PCR de la siguiente manera:

	1t	2x35t
• dH ₂ O	21.6 µl (a 50 µl)	756
• cADN	5 µl	-
• 5xRegulador HF	10 µl	350
• dNTP(25 mM)	0.4 µl	14
cebador alto (IGHV nuevo-corto) (4 µM)	6.25 µl (25 pmol)	218.75
• cebador bajo (IGHC nuevo-corto) (4 µM)	6.25 µl (25 pmol)	218.75
• Phusion	0.5 µl	17.5

- 10 Se agregó 90 mezcla de PCR a 10 µl de ADNc, se dividieron las reacciones de PCR en 2 tubos de 50, para cada muestra.

Realización de ciclos térmicos de la siguiente manera:

- 1- 98°C durante 1 min
- 2- 98°C durante 10 segundos
- 3- 62°C durante 20 segundos
- 4- 72°C durante 20 segundos para ir a la etapa 2, 20x

ES 2 730 951 T3

5- 72°C durante 5 min

6- 4°C pausa

Se reagruparon cada 2 reacciones en 1. Se agregaron 2 µl de exonucleasa I (20 U/µl) a cada tubo y se incubaron a 37°C durante 20 minutos. AMPure XP relación purificada de 1.8:1. Resuspender en 40 µl.

Se llevó a cabo la purificación SPRI de la siguiente manera:

- 5
- Agregar la cantidad requerida de perlas AMPure XP (relación 1.8:1) a la muestra de ADN en el regulador EB.
 - Vórtice para mezclar
 - Incubar durante 5 minutos a temperatura ambiente.
 - Imán (MPC) durante 5 minutos. Dejar el tubo de perlas en el MPC durante todos los pasos de lavado
- 10
- Se retira el sobrenadante (se mantiene en caso de falla) y se lavan las perlas dos veces con 500 µl de etanol al 70%, incubando durante 30 segundos cada vez.
 - Se retira todo el sobrenadante, se centrifuga rápidamente, se retira la última gota y se deja que las perlas AMPure se sequen al aire por completo (2 min).
 - Se retira el tubo del MPC, se agregan 40 µl de Tris-HCl 10 mM, EDTA 0.1 mM, pH 8.0 (o Qiagen Regulador EB), y se agita durante 30 segundos para volver a suspender las perlas, se deja reposar durante 3 minutos el imán durante 2 minutos y se transfiere el sobrenadante a un nuevo tubo.
- 15

Se corrió un gel de diagnóstico para verificar la eficacia de la PCR. Se corrieron 2 µl en un 2% de Egel-X durante 12 min.

Finalizar biblioteca

- 20
- El siguiente paso fue escrito para finalizar toda la biblioteca restante. La placa se ordenó como se establece en la Tabla 10 a continuación.

Tabla 10

Tipo de biblioteca (todas las cadenas pesadas)	Colocación de fila e información de BC 1=13, etc.
Triple VDJ-HC GMC	A1-A10
Triple VDJ-HC IDO	B1-B10
Triple VDJ-HC FV	C1-C10
RA	D1-D8
VIH ½	E1-E8 (5 progresores + 5 negativo) BC; 1-2-3-4-5/6-7-8-9-10
VIH 2/2	F1-F7 (5 elite) BC; 1-2-3-4-5

Se realiza la reacción de bloquear utilizando el kit de reparación Enzymatics End de la siguiente manera:

	1t	<u>60</u>
• H ₂ O	0 µl(a 25 µl)	0
• Post PCR Purificado de ADN	19 µl	-

ES 2 730 951 T3

	1t	<u>60</u>
Regulador de reparación End 10 ×	2.5 ul	150
• Mezcla dNTP 1mM	2.5 µl (0.1 mM)	150
Mezcla de enzimas (HC).	1 µl	60

Se incubó a una temperatura de 25°C durante 30 min. (19/t). Se inactivó con calor a 75°C durante 20 min.

Relave en A agregando lo siguiente directamente a la mezcla:

	1t	60t
• H ₂ O	20 µl (a 50µl)	1200
• Bloquear muestra de ADN	25 µl	-
• regulador Klenow 10X	2.5 µl	150
• dATP 10 mM	0.5 µl	30
• Klenow exo HC(3' a 5' exo menos)	2 µl	120

La reacción total fue ahora de 50 ul (25/t)

- 5 Se incubó a 37°C durante 30 min. Se inactivó con calor a 75°C durante 20 min. Se prepararon 454 adaptadores en Y. 10 µl de cada cebador A y B 100 µM + 30 µl de (Tris 10 mM, EDTA 0.1 mM, NaCl 50 mM)) (cada adaptador está a 20 µM final). Se incubó a 95°C durante 3 minutos, descender a 15 perpetua a 0.1°C/seg.

Para la ligación del adaptador Y, se añadió directamente a la reacción lo siguiente:

	1t	2x30t
• H ₂ O	0 (agregar µl a 100 µl)	-
• ADN con relave en A	50 ul	-
• regulador de ligasa rápida 2x	46 µl	1380
• Adaptador 20 µM	2 µl	-
• T4 ADN ligasa (rápida)	2 µl	60

Incubar la PCR a 22°C durante 15 min. (48/t). La reacción total fue entonces de 100 µl.

- 10 Se usó AMPure y el sustrato se eluyó en 40 µl (usando el liquidador). El QC de la biblioteca para la combinación se probó en una muestra de VIH. Solo se analizaron las 5 primeras muestras (progresivas) (Figura 51).

Ejemplo IX

VIH VDJ

Las muestras se exponen en la Tabla 11 a continuación.

ES 2 730 951 T3

Paciente	Cohorte	Fecha	Tipo HLA	Carga viral	Sexo	Fecha de nacimiento	Etnicidad
CR0744R	Progresor	4/12/2010	A-0201 A-0301 B-3701 B-5201 C-0602 C-1202	4-12-10-31100	M	1970	Afroamericano
CR0795Q	Progresor	4/19/2010	A-0101 A-0201 B-2705 B-5101 C-0102 C-1502	4-19-10 - 61700	M	1964	Blanco
CR0216R	Progresor	4/29/2010	A-0301 A-6801 B-1401 B-3801 C-0802 C-1203	4-29-10 - 86600	F	1963	Blanco
CR0778Z	Progresor	5/3/2010	A-3303 A-6801 B-0702 B-1503 C-0702 C1800	5-3-10-45700	M	1964	Afroamericano
CR0757R	Progresor	5/6/2010	A-0201 A-2902 B-1501 B-4404 C-0102 C-1601	5-6-10-26300	M	1964	Blanco
CTR0147	Controlador Elite	4/15/2010	A-0103 A-3201 B-3508 B-7301 C-0401 C-1505	4-18-10 - <48	M	1953	Blanco
CTR24	Controlador Elite	4/15/2010	A-0201 A-3001 B-1302 B-5701 C-0602 C-0602	4-15-10 - <48	F	1954	Birracial
523507	Controlador Elite	5/14/2010	A-3002 A-2402 B-1801 B-5301 C-0401 C-1203	5-14-10 - 56	M	1962	Blanco
CTR174	Controlador Elite	6/28/2010	A-0201 A-2301 B-5201 B-5801 C-0302 C-1601	2-18-10 - <48	M	1961	Afroamericano
FHCR0013E	Controlador Elite	7/1/2010	A-0301 A-1101 B-0702 B-3501 C-0401 C-0702	2-17-10 - 133	M	1947	Blanco
924426	VIH Negativo	4/7/2010	A-0201 A-0205 B-5701 B-5701 C-0602 C-0602	0			
714739	VIH Negativo	4/7/2010	A-0101 A-0201 B-1801 B-5101 C-0602 C-1402	0			
564006	VIH Negativo	4/7/2010	A-0201 A-0201 B-1801 B-3901 C-0501 C-1203	0			

ES 2 730 951 T3

Paciente	Cohorte	Fecha	Tipo HLA	Carga viral	Sexo	Fecha de nacimiento	Etnicidad
084928	VIH Negativo	4/6/2010	A-0201 A-0301 B-1401 B-1801 C-0501 C-0802	0			
039701	VIH Negativo	4/6/2010	A-0201 A-3303 B-1501 B-5801 C-0302 C-0801	0			

Adquisición de muestras

Se obtuvo una muestra nueva. Se prepararon 2 tubos de EDTA para cada agente de avance, controlador de élite y muestra VIH negativa. Las muestras fueron procesadas inmediatamente después de la recepción (Ficoll, recuento de células). El aislamiento de células B se realizó de la siguiente manera:

- 5 1. Suspensión celular centrifugada a 300 x G durante 10 minutos, sobrenadante aspirado.
2. Pellas de células resuspendidas en 80 µL de regulador frío por 10⁷ células.
3. El regulador era una dilución 1:20 de la solución madre de MACS BSA con la solución de enjuague autoMACS. Desgasificado antes de su uso.
4. Se agregaron 20 µL de microperlas CD19 por 10⁷ células.
- 10 5. Se incubó durante 15 minutos en el refrigerador (2-8°C).
6. Lavar las células mediante la adición de 1-2 mL de regulador frío por 10⁷ células y centrifugar a 300 x G durante 10 minutos de sobrenadante de aspirado.
7. Resuspender hasta 10⁸ células en 500 µL de regulador frío
8. Colocar la columna MS en un campo magnético y enjuague la columna con 500 µL de regulador desgasificado.
- 15 9. Suspensión celular añadida.
10. Recolectar células sin marcar que pasaron y se lavaron tres veces con 500 µL de regulador.
11. Se retiró la columna y se colocó en un nuevo tubo de recogida.
12. Se agregó 1 mL de regulador y se sumergieron las células marcadas magnéticamente con el émbolo.
13. Contar células.
- 20 14. En una centrífuga pequeña, centrifugar las células CD19+ durante 10 minutos a 9.000 x G.
15. Se aspiró el sobrenadante y añadió 300 µL de regulador de lisis. Se congeló y se almacenó a -80°C.

Extracción de ARN

- 25 Se usó Mirvana según el protocolo del fabricante, se eluyó en 150 (doble fenol, la fase fue menor en la segunda ejecución). Nanodrop QC, demasiado bajo, por lo que EtOH precipitó con Gglicoblue, se eluye en 20 µl, Nanodrop. Diluir el número 4 1:2. Véase la Tabla 12 a continuación.

Tabla 12. Se usó mezcla extra para hacer 5x FV1 (180 ng/ul ... 1.67 ul de ARN + 5.33 ul de H₂O) para cadenas livianas y 5 para cadenas pesadas.

ID	ng/ul ARN	260/280	260/230	Total ARN	Volumen hasta 150 ng	Agua hasta 7
1	65.9	1.98	1.7	1252.1	2.276	4.724
2	32.4	1.92	1.33	615.6	4.630	2.370
3	26.1	2	1.48	495.9	5.747	1.253

ES 2 730 951 T3

ID	ng/ul ARN	260/280	260/230	Total ARN	Volumen hasta 150 ng	Agua hasta 7
4	216.1/120.6	2.03/1.97	2.05/1.58	4106/4583	1.244	5.756
5	103.4	2.02	1.93	1964.6	1.451	5.549
6	54.3	1.98	1.66	1031.7	2.762	4.238
7	89.9	2.02	1.89	1708.1	1.669	5.331
8	51.4	1.66	0.88	976.6	2.918	4.082
9	35.8	2.05	1.66	680.2	4.190	2.810
10	79.8	1.85	1.25	1516.2	1.880	5.120
11	104.4	1.96	1.62	1983.6	1.437	5.563
12	82.2	1.71	0.91	1561.8	1.825	5.175
13	154.8	1.97	1.72	2941.2	0.969	6.031
14	56.2	1.87	1.33	1067.8	2.669	4.331
15	84.4	1.8	1.06	1603.6	1.777	5.223

5 La muestra 8 se concentró mediante precipitación con etanol, se volvió a suspender en 20 µl, Nanodrop = 134.0. Estos eran solo ARN de células B, por lo que no había necesidad de la misma cantidad de ARN inicial para la investigación de VDJ. Teóricamente, los linfocitos representan el 30% de las PBMC, para las cuales las células B representan el 10%, por lo tanto, 15 ng deberían ser suficientes. Se usaron 150 ng.

Una reacción de transcripción reversa (RT) se montó de la siguiente manera:

	<u>2x 1 tubo</u>
• H ₂ O(DEPC)	0 µl(a 12 µl)
• Cebador específico del gen (pesado o ligero) 2 µM	5 µl (5 pmol)
• Total ARN	7 µl (150 ng)

Se calienta a 95°C durante 1 minuto seguido de 65°C durante 5 minutos, luego hielo durante 1 minuto.

Se centrifugó y se añadió lo siguiente: 1 tubo 30 tubos

• 5x regulador de primera hebra	4 µl	120
• Mezcla dNTP 10 mM	1µl	30
• DTT 0.1M	1µl	30

ES 2 730 951 T3

Se centrifugó y se añadió lo siguiente: 1 tubo 30 tubos

• Inhibidor Rnasa Enzymatics	1µl	30
• Superíndice III	1µl	30

Se incubó a 55°C durante 60 min (8 µl/tubos)

Enzima inactivada por calentamiento a 70°C durante 15 min. Se eliminó el híbrido ARN/ADN agregando 1 µl de E. coli RNasaH (Enzy). Se incubó a 37°C durante 20 min., luego hielo. Véase la Tabla 13 a continuación.

Tabla 13.

Tipo cebador	Fecha de la siembra	Patrón de siembra	Ubicación del pozo	µl para cada cebador	µl H2O	Mezcla final
5 x J IGH RT-PCR cebador	Orden de tubo 5933901			20 ul*5 tubos	+400 ul	4 µM* Diluir 1-2 antes de usar
6 x J IGH PCR cebador	20080924	200 µM	Placa 3 A1-A6	20 µl cada (100 µl total)	880 µl	4 µM (1000µl)
43x V IGH PCR cebador	20080924	200 µM	Placa 1 A1-D7	20 µl cada (860 µl total)	140 µl	4 µM (1000µl)

5

Se prepara una reacción de prueba de PCR de la siguiente manera:

1x x 2 por muestra x 15 muestras (45 t) (cadena liviana de 12 t)

dH ₂ O	21µl (a 50 µl)	945 (1084.46)	252 (259.2)
ADNc	5 µl	-	-
5x Regulador HF	10 µl	450	120
dNTP(10 mM) (25 mM)	1 µl (0.4)	45 (18)	12 (4.8)
Cebador alto (IGHV nuevo-corto) (4µM)	6.25 µl (25 pmol)	281.25 (225)	75
Cebador bajo (IGHC nuevo-corto)(4µM)	6.25 µl (25 pmol)	281.25 (224)	75
Fusión	0.5 µl	22.5	6

Se agregó mezcla de 90 PCR a 10 µl de ADNc, se dividió la reacción de PCR en 2 tubos de 50, para cada muestra. Era cebados bajo, por lo que se usaron valores reforzados

10 Ciclo térmico de la siguiente manera:

1- 98°C durante 1 min

2- 98°C durante 10 segundos

ES 2 730 951 T3

- 3- 62°C durante 20 segundos
- 4- 72°C durante 20 segundos, va al paso 2, 23x
- 5- 72°C durante 5 min
- 6- 4°C pausa

Se reagruparon cada 2 reacciones en 1. Se agregaron 2 µl de exonucleasa I (20 U/µl) a cada tubo y se incubaron a 37°C durante 20 minutos. Relación de purificación AMPure XP de 1.8:1. Resuspender en 40 µl.

- 5 Se ejecutó purificación por SPRI: Se agregó la cantidad requerida de perlas AMPure XP (relación de 1.8:1) a la muestra de ADN en el regulador EB. Se sometió a vórtice para mezclar. Se incubó durante 5 minutos a temperatura ambiente. Imán (MPC) durante 5 minutos. Se dejó el tubo de perlas en el MPC durante todos los pasos de lavado. Se eliminó el sobrenadante (se mantuvo en caso de falla) y se lavaron las perlas dos veces con 500 µl de etanol al 70%, incubando durante 30 segundos cada vez. Se eliminó todo el sobrenadante, se centrifugó rápidamente, se eliminó la última gota y se dejaron secar al aire las perlas de AMPure completamente (2 min). Se extrajo el tubo del MPC, se agregaron 40 µl de Tris-HCl 10 mM, EDTA 0.1 mM, pH 8.0 (o Regulador EB de Qiagen), y se agitó en vórtice durante 30 segundos para volver a suspender las perlas, dejar reposar durante 3 min., imán durante 2 min. y se transfirió el sobrenadante a un nuevo tubo.

10 Se corrió un gel de diagnóstico para verificar la eficacia de la PCR (Figura 52). Se corrió 2 µl en un 2% de Egel-X durante 12 min.

- 15 Finalizar biblioteca

El siguiente método fue desarrollado para finalizar la biblioteca restante.

La placa se ordenó de la siguiente manera en la Tabla 14 a continuación.

Tabla 14.

Tipo de biblioteca (todas las cadenas pesadas)	Colocación de fila e información de BC 1=13, etc.
Triple VDJ-HC GMC	A1-A10
Triple VDJ-HC IDO	B1-B10
Triple VDJ-HC FV	C1-C10
RA	D1-D8
VIH 1/2	E1-E8 (5 progresores + 5 negativos) BC; 1-2-3-4-5/6-7-8-9-10
VIH 2/2	F1-F7 (5 elite) BC; 1-2-3-4-5

- 20 Se realizó la reacción de bloqueo utilizando el kit de reparación Enzymatics End de la siguiente manera:

	1t	<u>60</u>
• H ₂ O	0 µl(a 25 µl)	0
• Post PCR Purificado de ADN	19 µl	-
• 10× Regulador de reparación End	2.5 µl	150

ES 2 730 951 T3

	1t	<u>60</u>
• Mezcla dNTP 1mM	2.5 µl (0.1 mM)	150
• Mezcla de enzimas (HC)	1 µl	60
Se incubó a una temperatura de 25°C durante 30 min.		(19/t)

Se inactivó con calor a 75°C durante 20 min.

Relave en A agregando lo siguiente directamente a la mezcla:

	1t	60t
• H ₂ O	20 µl(to 50 µl)	1200
Muestra de ADN bloqueado	25 µl	-
• 10X regulador Klenow	2.5 µl	150
• dATP 10 mM	0.5 µl	30
• Klenow exo HC(3' a 5' exo menos)	2 µl	120
Total rx es ahora 50 ul		(25/t)

- 5 Se incubó a 37°C durante 30 min. Se inactivó con calor a 75°C durante 20 min.

Se prepararon 454 adaptadores Y: 10 µl de cada cebador A y B 100 µM + 30 µl de (Tris 0.1 10 mM, EDTA mM a NaCl 50 mM)) (cada adaptador está a 20 µM final). Se incubó a 95°C durante 3 minutos, descendió hasta 15°C a 0.1°C/seg.

La ligación del adaptador Y, se añade directamente a la reacción de la siguiente manera;

	1t	2x30t
• H ₂ O	0 (añadir µl a 100 µl)	
• ADN relave en A	50 µl	
• 2x regulador de ligasa rápida	46 µl	1380
• Adaptador 20 µM	2 µl	-
• T4 ADN ligasa (rápida)	2 µl	60

- 10 Incubado en PCR a 22°C durante 15 min (48/t). La reacción total era ahora de 100 µl.

AMPure, se eluye en 40 ul (se utiliza el liquidador)

Biblioteca de QC para el conjunto

Prueba en muestra de VIH, prueba solo las primeras 5 muestras (progresor). Muestra en progreso Nanodrop.

Diluir a 5 ng/ul. Nanodrop:

ES 2 730 951 T3

1	36.8
2	31.7
3	24.8
4	33.5
5	31.3

Diluido a 1 ng/μl utilizando tris 10 mM, Tween 20 al 0.05%. Diluido a 0.01 ng/μl (en 200 μl) en tris 10 mM, Tween 20 al 0.05%.

5 Se preparó una mezcla rápida de SYBR de preaviso suficiente para una ejecución por triplicado. Se utilizaron 18 μl de mezcla y se le agregaron 2 μl de plantilla.

	1t	20
• H ₂ O	7.2 ul (a 20 ul)	144
• 2x SYBR rápido	10 ul	200
• 10 μm Cebador alto	0.4 ul	8
• 10 μm Cebador abajo	0.4 ul	8
• 0.01 ng/ul ADN plantilla	2 ul	-

Ciclo:

95	1 min
95	5 seg
60	20 seg
72	15 seg repetir 40 veces

Se preparó la mezcla rápida de SYBR lo suficiente para una ejecución por triplicado

Se hicieron 2 corridas, una serie de la muestra de 0.01 ng y una de la muestra inicial de 5 ng/ul

10 Se usaron 15 ul de mezcla, y se agregaron 5 ul de plantilla. Para muestra de 0.01 ng, se usa de la siguiente manera:

0.01 ng volumen buscado	Tris
1.76	3.24
2.50	2.50
3.52	1.48
2.17	2.83

ES 2 730 951 T3

0.01 ng volumen buscado	Tris
5.00	0.00

Para la muestra de ~5 ng, la siguiente dilución se realizó hasta un nivel real de 1 ng/μl y se diluyó 1/100 hasta un valor real de 0.01 ng/μl. Utilizado 5 μl.

Muestra	veces de dilución
1	13.2
2	16.4
3	15.3
4	30.1
5	10.9

	1t	20
◦ H ₂ O	4.2 ul (a 20 ul)	84
◦ 2x SYBR rápido	10 ul	200
◦ 10 μm cebador alto	0.4 ul	8
◦ 10 μm cebador bajo	0.4 ul	8
◦ 0.01 o 5 ng/ul plantilla de ADN	5 ul	

5 Lector de placa

La muestra fue diluida como sigue. Volumen de H₂O agregado a un volumen de 25 ul de la muestra para lograr la misma concentración de 250 nM.

GMC	26.95	11.56	8.99	9.23	15.31	10.02	10.67	0.22	14.69	14.23
IDO	11.40	12.94	8.58	9.95	11.15	11.05	26.07	8.04	11.02	10.27
FV	15.61	21.09	17.18	16.65	6.26	12.26	20.76	14.12	13.01	15.07
RA	7.58	6.95	8.50	7.51	8.94	5.92	9.59	6.97		
VIH	45.28	23.65	21.59	29.23	23.86	27.46	29.26	29.74	34.51	28.37
VIH elite	37.04	29.70	27.82	34.51	37.93					

10 GMC1, se mezclaron 25 μl de cada muestra para un total de 30*10=300 μl. IDO1, se mezclaron 30 μl. FV1, se mezclaron 30 μl. RA, se mezclaron los primeros 4 a 35 μl y los últimos 4 a 35 μl por separado, se precipitó y se cargó en un gel para decidir cómo mezclarlos. VIH, proporción igual de agrupamiento de cada uno (40 μl), pero no se

ES 2 730 951 T3

mezclaron muestras de élites con el combo neg+prog, se ejecutarán con una junta (por lo tanto, se leerá el total para la élite).

La preparación de Qiagen +5 µl de NaOAc (lavado de 2 minutos), se eluyó en 30 µl de Nanodrop.

#	Muestra ID	Ácido nucleico conc.	260/280	260/230	total ng	µM
1	GMC	16	2.01	1.98	480	1.39
2	IDO	25.7	1.93	1.87	771	2.22
3	FV	31.4	1.87	1.99	942	2.72
4	RA1	8.4	1.95	2.02	252	0.73
5	RA2	6.4	2.36	1.87	192	0.55
6	VIH1	75.7	1.85	2.17	2271	6.55
7	VIH2	37.9	1.87	2.18	1137	3.28
8	VIH control	22.4	1.95	1.99	672	1.94

- 5 Preparación probada de Pippin con muestra de control número 8; 475-525, 475-550, 500-575 (Figura 53) Se corrieron todas las muestras en Pippin preparativa. División VIH1 en 2 corridas, Extracto 475-550. 1, GMC; 2, IDO; 3, FV; 4, VIH1/2 (Figura 54).

Biblioteca Qubit HS ADNds

Se reunieron ambas muestras de RA juntas se aplicó AMPure porque RA2 no tiene nada.

no ID/Qubit HS ADNds/diluido a 5 ng/µl, tomar X µl/agregar Y H₂O/volumen total

1	GMC	8.44	3	2.064	5.064
2	IDO	7.99	3	1.794	4.794
3	FV	6.59	3	0.954	3.954
4	RA1	6.59	3	0.954	3.954
5	RA2			reunir con RA1	
6	VIH1	49.8	3	26.88	29.88
7	VIH2	25.2	3	12.12	15.12

- 10 Diluido 1/100 a 0.05 ng/ul. Cargar 1 ul en e GEL de cada muestra.

Corrida en tiempo real

Se preparó una mezcla rápida de SYBR lo suficiente para una corrida por triplicado. Usar 18 ul de mezcla y agregar 2 ul de plantilla.

ES 2 730 951 T3

	1t	20
• H ₂ O	7.2 µl (a 20 µl)	144
• 2x SYBR rápido	10 µl	200
• 10 µm cebador alto	0.4 µl	8
• 10 µm Cebador bajo	0.4 µl	8
• 0.05 ng/ul plantilla de ADNA	2 µl	-

Ciclo

95 1 min
 95 5 seg
 60 20 seg
 72 15 seg repetir 40 veces

Diluido todo a 5 ng/µl usando 13.5 µl de muestra. Se agregó H₂O a 13.5 µl de muestra.

1	9.45
2	8.073
3	4.293
4	4.293
5	120.96
6	54.54

Enviados 15 µl a 454.

5 Ejemplo X

PCR en emulsión de célula única y perla

Oligonucleótido 5'-3' (forma ordenada HPLC IDT purificada cuando sea aplicable)	
PR1-F4-2BIO (1 mM)	/52-Bio/CCA CTA CGC CTC CGC TTT CCT CTC TAT GG
PR1-R_Corto (2 mM)	CTG CCC CGG GTT CCT CAT T

Tubo	Perlas	Proceso
1	Dynabeads 1 µm myone C1 10mg/ml	todo lavado con imán

ES 2 730 951 T3

Tubo	Perlas	Proceso
2	Dynabeads 1 µm myone C1 10 mg/ml	todo lavado por centrifugación (1 min velocidad máxima)
3	IBS 1 µm no magnético 10 mg/ml	todo lavado por centrifugación (1 min velocidad máxima)

Carga de perla con cebador de anclaje en emulsión

5 Perlas magnetizadas o centrifugadas durante todos los pasos de acuerdo con la tabla de configuración del ensayo. Las perlas se agitaron en vórtice para resuspender y se transfirieron 100 µl a 1.5 ml de tubo de silicona Ambion. Se lavó dos veces con 200 µl de 2x de regulador de unión y lavado (2xB&W). Perlas resuspendidas en 100 µl de 2xB&W, agregar cebador de anclaje premezclado; 1 mM PR1-F4-2Bio (2 µl); H₂O (98 µl). Se incubó en rotador durante 20 min. Se lava dos veces con 200 µl de 1xB&W. Se lava una vez con 200 µl de TE. Resuspender las perlas en 100 µl de TE y almacenarlas en el rotador en un recinto frío hasta que se usen.

PCR en emulsión

Mezcla acuosa preparada como sigue: 1 tubo

• dH ₂ O	316.4 µl (a 800 µl)
• 10x regulador de PCR (Enzymatics)	96 µl
• MgCl ₂ 50 mM	242 µl
• Mezcla dNTP 25 mM	135 µl
• PR1R-S2 mM	6 µl
• 30% (p/v) BSA (Sigma)	1.6 µl
• Plantilla de biblioteca*	3 µl

10

Se somete a vórtice bien para mezclar. *para la plantilla de biblioteca, se utilizó una dilución 1:20 de una biblioteca genómica humana de 280 pb1 nM.

Mezcla de aceite preparado: 1 tubo 4 tubos (prep. todo en un Falcon de 509 ml)

• Tegosoft DEC	4.4 ml	17.6 ml
• aceite mineral 1.2 ml	4.8 ml	
• ABIL WE09	425 µl	1.7 ml

15

Se agitó en vórtice, se dejó desgasificar. Se dividieron en alícuotas 5.5 ml en tubos de Falcon de aluminio recubiertos de teflón de 50 ml.

Crear emulsión. A la mezcla de PCR de 800 µl, se agregaron 100 µl de Enzymatics Taq (5 U/µl), se agitaron rápidamente y se centrifugaron. Se añadieron inmediatamente 60 µl de perlas de PR1F4, se agitaron rápidamente y se centrifugaron. El total de 960 µl se transfirió inmediatamente a un tubo de aceite y se agitó en vórtice durante 2.25 min a 2200 rpm. El tubo se colocó en la rejilla del hidrociclador y se inició el programa de PCR para el ciclo nocturno.

20

Ciclo como sigue:

1- 94°C durante 5 min

ES 2 730 951 T3

2- 94°C durante 15 segundos

3- 58°C durante 30 segundos (1 grado por encima para compensar el ajuste del hidrociclador)

4- 70°C durante 75 segundos

5- Se llevó 2 para 119x

5 6- 72°C durante 2 min

7- Tanque delantero frío (~10°C perpetua)

Ruptura de emulsión

10 Se añadieron 22 ml de isopropanol a cada tubo de Falcon de aluminio, y los tubos se agitaron durante 50 segundos a 3000 rpm. Los contenidos se transfirieron a un nuevo tubo de Falcon de polipropileno de 50 ml y se centrifugaron (toque a 4000 rpm y se detuvieron). Los sobrenadantes se vertieron (el sedimento se pegó bien al fondo). Se agregaron 30 ml de isopropanol y los tubos se agitaron durante 40 segundos a 3000 rpm, se centrifugaron (toque 4000 rpm y se detuvo), y el sobrenadante se vertió (la pella se pegó bien al fondo) .

15 Se añadieron 30 ml de regulador NXS (tris-HCl 10 mM, pH 8.0, NaCl 100 mM, Triton X-100 al 1% (v/v), SDS al 1% (v/v)) y los tubos se agitaron con vórtice durante 1.5 min. a 3000 rpm (agitar más el vórtice si la pella todavía estaba atascada en la parte inferior), se centrifuga (se mantiene a 4000 rpm durante 15 segundos y se detiene), y se pipeteó el sobrenadante (cuidadosamente ya que la pella no se pegó muy bien al fondo esta vez). Se añadieron 100 µl de TE y el sedimento se resuspendió mediante pipeteo. La solución se transfirió a un tubo Eppendorf siliconado 1.5, se lavó 1x con 200 µl de NXS, se lavó 2x con 200 µl de TE, se incubó durante 5 minutos en 100 µl de NaOH 100 mM a temperatura ambiente, se lavó 1x con 120 µl de NaOH 100 mM, se lavó 2x con 150 µl de TE, se resuspendió en 30 µl de TE y se transfirió a un nuevo tubo siliconado para su almacenamiento.

20 Secuenciación por diagnóstico de ligación de perlas

La secuenciación por ligación de la posición menos uno se realizó usando métodos conocidos en la técnica.

Captura de perlas en dispositivo de acrilamida para SBL

Se preparó un portaobjetos de silano unido a acrilamida de la siguiente manera:

Perlas en TE	7 µl
• 40% Acrílico 19:1	1.25 µl
• 5% TEMED	0.5 µl
• 0.5% APS	0.75 µl

25

Esto se colocó en un portaobjetos por medio de la absorción bajo el cubreobjetos y deslizando un cubreobjetos en la parte superior y el portaobjetos se colocó boca abajo. El portaobjetos polimerizó (~30 min, máx. 2 h) y se formó un círculo seco (preparar el lavado 1 durante este tiempo). Se quitó el cubreobjetos y se puso el portaobjetos en un tubo cónico con TE (agitar durante 10 minutos)

30 Lavado preparado 1:

• 1M Tri-HCl pH 7.5	10 ml
• 2M KCl	25 ml
• 0.5M EDTA	4 ml
• 10% Triton X	1 ml

Mezcla de cebador preparada:

ES 2 730 951 T3

- 6X SSPE con 0.01% Triton X-100 100 µl
- cebador de anclaje 1 mM 1µl

5 Se retiró el portaobjetos del tubo cónico, se secó la superficie de teflón blanca con Kimwipe, se agregaron 100 µl de la mezcla de cebador de anclaje, se incubó a 56°C durante 1 minuto, se colocó en el tubo cónico con el Lavado 1, se cambió inmediatamente a un nuevo Lavado 1, se incubó a temperatura ambiente durante 5 min en el lavado 1 con agitación suave.

Mezcla de ligación preparada como sigue;

- H₂O 41 µl
- regulador de T4 ADN ligasa 10x 5 µl
- Mezcla fluorescente de nonúmero 4 µl
- T4 ADN ligasa (2K U/µl) 1 µl

10 Se retiró el portaobjetos del tubo cónico, se secó la superficie de teflón blanca con Kimwipe, se agregaron 50 µl de mezcla de ligación, se incubaron a 35°C durante 5 minutos, se colocaron en el tubo cónico con el Lavado 1, se cambiaron inmediatamente a un nuevo Lavado 1, se incubaron a temperatura ambiente durante 5 minutos en el Lavado 1 con agitación suave, se cambió el Lavado 1 para TE, y se incubó con agitación suave durante 10 minutos a 2 horas. Se tomaron imágenes al microscopio y se analizaron los datos de las imágenes (consulte las Figuras 55-57).

Ejemplo XI

Ilumina VDJ

15 Objetivo

Para desarrollar una muestra, Illumina emparejó el protocolo de secuenciación final para el ADN de inmunoglobulina.

Oligonucleótidos

20 Se ordenaron cebadores que tenían una placa de cocción, desalinización estándar, una segunda sección compatible con el ADN plano del primer lote, con la extensión primaria adecuada del adaptador Illumina (no agrupamiento) (Tabla 15). Las placas se almacenaron a 200 µM.

Table 15.

Ref #	Nombre	Pos		TM
50494259	VDJ_IGHV_gDNA_01	A1	ACA GAA GTT CCA GGG CAG	54.62 °C
50494263	VDJ_IGHV_gDNA_02	A2	AGA CTC CAG GAA GGG CAG	56.48 °C
50494264	VDJ_IGHV_gDNA_03	A3	AGA CTC CAT GAA GGG CCA	55.89 °C
50494265	VDJ_IGHV_gDNA_04	A4	AGA CTC TGT GAA AGG CCG	54.53 °C
50494266	VDJ_IGHV_gDNA_05	A5	AGC ACA TCT CTG AAG ACC AG	54.19 °C
50494267	VDJ_IGHV_gDNA_06	A6	AGC ACA TCT CTG AAG AGC AG	54.46 °C

ES 2 730 951 T3

Ref #	Nombre	Pos		TM
50494268	VDJ_IGHV_gDNA_07	A7	AGC ACG TCT CTG AAG AAC AG	54.53 °C
50494269	VDJ_IGHV_gDNA_08	A8	ATT ATG CAG TAT CTG TGA AAA GTC G	53.36 °C
50494270	VDJ_IGHV_gDNA_09	A9	CAG AAG CTC CAG GGC AG	55.68 °C
50494260	VDJ_IGHV_gDNA_10	A10	CAG ACT CTG TGA AGG GCA G	55.21 °C
50494261	VDJ_IGHV_gDNA_11	All	CAG AGA AGT TCC AGG GCA G	55.28 °C
50494262	VDJ_IGHV_gDNA_12	A12	CAG GGC TTC ACA GGA CG	55.85 °C
50494271	VDJ_IGHV_gDNA_13	B1	CCC CTC CCT CAA GAG TCG	57.07 °C
50494275	VDJ_IGHV_gDNA_14	B2	CCC GTC CCT CAA GAG TCT	56.07 °C
50494276	VDJ_IGHV_gDNA_15	B3	CCG TCC CTC AAG AGT CG	54.86 °C
50494277	VDJ_IGHV_gDNA_16	B4	CCG TCC TTC CAA GGC CA	57.77 °C
50494278	VDJ_IGHV_gDNA_17	B5	CGC ACA GAA ATT CCA GGA CAG	56.24 °C
50494279	VDJ_IGHV_gDNA_18	B6	CGC ACA GAA GTT CCA GGA AAG	56.18 °C
50494280	VDJ_IGHV_gDNA_19	B7	CGC GTC TGT GAA AGG CAG	56.79 °C
50494281	VDJ_IGHV_gDNA_20	B8	GAC TCC GTG AAG GGC CG	58.62 °C
50494282	VDJ_IGHV_gDNA_21	B9	GAC TCA GTG AAG GGC CG	55.55 °C
50494272	VDJ_IGHV_gDNA_22	B10	GAC TCC GTG AAG GGC AG	55.55 °C
50494273	VDJ_IGHV_gDNA_23	B11	GAC TCT GTG AAG GGC CG	55.55 °C
50494274	VDJ_IGHV_gDNA_24	B12	GCA AAC TCT GTG AAG GGC AG	56.54 °C
50494283	VDJ_IGHV_gDNA_25	C1	GCA CAG AAG TTT CAG GGC AG	56.54 °C
50494284	VDJ_IGHV_gDNA_26	C2	GCA CCC GTG AAA GGC AG	57.09 °C
50494285	VDJ_IGHV_gDNA_27	C3	GCC CAT CTC TGA AGA GCA G	55.37 °C
50494286	VDJ_IGHV_gDNA_28	C4	GCG TCG GTG AAA GGC AG	56.97 °C
50494287	VDJ_IGHV_gDNA_29	C5	CTC CGT GAA GCG CCG	56.96 °C
50494288	VDJ_IGHV_gDNA_30	C6	TGC GTC TGT GAA AGG CAG	55.64 °C

ES 2 730 951 T3

Ref #	Nombre	Pos		TM
50494292	VDJ_IGHJ_gDNA_01	D1	CTT ACC TGA GGA GAC GGT GAC C	58.73 °C
50494293	VDJ_IGHJ_gDNA_02	D2	CTC ACC TGA GGA GAC AGT GAC C	58.72 °C
50494294	VDJ_IGHJ_gDNA_03	D3	CTT ACC TGA AGA GAC GGT GAC C	56.83 °C
50494295	VDJ_IGHJ_gDNA_01lluoverPE	D7	CTC GGC ATT CCT GCT GAA CCG CTC TTC CGA TCT CTT ACC TGA GGA GAC GGT GAC C	71.88 °C
50494296	VDJ_IGHJ_gDNA_02lluoverPE	D8	CTC GGC ATT CCT GCT GAA CCG CTC TTC CGA TCT CTC ACC TGA GGA GAC AGT GAC C	71.90 °C
50494297	VDJ_IGHJ_gDNA_03lluoverPE	D9	CTC GGC ATT CCT GCT GAA CCG CTC TTC CGA TCT CTT ACC TGA AGA GAC GGT GAC C	71.28 °C
50494298	VDJ_IGHV_gDNA_01lluover	E1	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT ACA GAA GTT CCA GGG CAG	69.95 °C
50494302	VDJ_IGHV_gDNA_02lluover	E2	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AGA CTC CAG GAA GGG CAG	70.46 °C
50494303	VDJ_IGHV_gDNA_03lluover	E3	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AGA CTC CAT GAA GGG CCA	70.54 °C
50494304	VDJ_IGHV_gDNA_04lluover	E4	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AGA CTC TGT GAA AGG CCG	69.81 °C
50494305	VDJ_IGHV_gDNA_05lluover	E5	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AGC ACA TCT CTG AAG ACC AG	69.24 °C
50494306	VDJ_IGHV_gDNA_06lluover	E6	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AGC ACA TCT CTG AAG AGC AG	69.30 °C
50494307	VDJ_IGHV_gDNA_07lluover	E7	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AGC ACG TCT CTG AAG AAC AG	69.25 °C

ES 2 730 951 T3

Ref #	Nombre	Pos		TM
50494308	VDJ_IGHV_gDNA_08lluover	E8	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT ATT ATG CAG TAT CTG TGA AAA GTC G	67.88 °C
50494309	VDJ_IGHV_gDNA_09lluover	E9	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CAG AAG CTC CAG GGC AG	71.06 °C
50494299	VDJ_IGHV_gDNA_10lluover	E10) ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CAG ACT CTG TGA AGG GCA G	70.35 °C
50494300	VDJ_IGHV_gDNA_11lluover	E11	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CAG AGA AGT TCC AGG GCA G	70.42 °C
50494301	VDJ_IGHV_gDNA_12lluover	E12	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CAG GGC TTC ACA GGA CG	71.03 °C
50494310	VDJ_IGHV_gDNA_13lluover	F1	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CCC CTC CCT CAA GAG TCG	71.14 °C
50494314	VDJ_IGHV_gDNA_14lluover	F2	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CCC GTC CCT CAA GAG TCT	71.05 °C
50494315	VDJ_IGHV_gDNA_15lluover	F3	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CCG TCC CTC AAG AGT CG	70.66 °C
50494316	VDJ_IGHV_gDNA_16lluover	F4	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CCG TCC TTC CAA GGC CA	71.86 °C
50494317	VDJ_IGHV_gDNA_17lluover	F5	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CGC ACA GAA ATT CCA GGA CAG	70.19 °C
50494318	VDJ_IGHV_gDNA_18lluover	F6	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CGC ACA GAA GTT CCA GGA AAG	70.16 °C
50494319	VDJ_IGHV_gDNA_19lluover	F7	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CGC GTC TGT GAA AGG CAG	70.93 °C
50494320	VDJ_IGHV_gDNA_20lluover	F8	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GAC TCC GTG AAG GGC CG	71.82 °C

ES 2 730 951 T3

Ref #	Nombre	Pos		TM
50494321	VDJ_IGHV_gDNA_21lluover	F9	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GAC TCA GTG AAG GGC CG	71.03 °C
50494311	VDJ_IGHV_gDNA_22lluover	F10	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GAC TCC GTG AAG GGC AG	71.03 °C
50494312	VDJ_IGHV_gDNA_23lluover	F11	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GAC TCT GTG AAG GGC CG	71.03 °C
50494313	VDJ_IGHV_gDNA_24lluover	F12	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GCA AAC TCT GTG AAG GGC AG	70.65 °C
50494322	VDJ_IGHV_gDNA_25lluover	G1	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GCA CAG AAG TTT CAG GGC AG	70.65 °C
50494323	VDJ_IGHV_gDNA_26lluover	G2	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GCA CCC GTG AAA GGC AG	71.50 °C
50494324	VDJ_IGHV_gDNA_27lluover	G3	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GCC CAT CTC TGA AGA GCA G	70.51 °C
50494325	VDJ_IGHV_gDNA_28lluover	G4	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GCG TCG GTG AAA GGC AG	71.35 °C
50494326	VDJ_IGHV_gDNA_29lluover	G5	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CTC CGT GAA GCG CCG	71.68 °C
50494327	VDJ_IGHV_gDNA_30lluover	G6	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT TGC GTC TGT GAA AGG CAG	70.40 °C

Extraer ADN genómico

5 Sangre extraída. Extracción de PBMC (ficoll), se congeló una parte en DMOS para un posible arrastre en vivo. Se utilizaron reservas de células congeladas (3 viales, una para el día 0, día 7 y día 21 de células totales, 5.5 millones, 70% de viables). ADN extraído (kit Agencourt; mezcla de lisis usada 10x + 50 µl de cóctel de RNasa; unión de perlas usadas al 50% en volumen, es decir, 300 µl de perlas para 600 µl de solución). Nanodrop: 62.2 ng/µl (260/180 1.90, 280/230 2.0) total ~18 µg.

Configuración de cebador

10 Se hizo una placa diluida a 5 µM cada una; 2 µl de cada + 78 µl de H₂O, pero también a tubos agrupados a 5 µM una vez agrupados. V; 3 µl de cada (=90 µl en total) + 30 µl de H₂O = 120 µl a 5 µM. J; 2 µl de cada (=6 µl total) + 74 µl de H₂O = 80 µl a 5 µM

Prueba de cebador individual

ES 2 730 951 T3

Se probaron todos los cebadores uno por uno (solapamiento no ilumina solamente): cada V contra un conjunto de los 3 J, (30 reacción de PCR); cada J contra un conjunto de todos los V (3 reacción de PCR).

Se preparó una reacción de la siguiente manera:

	1 tubo		5t 35t	
dH ₂ O	9.8 µl (a 20 µl)		49	343
62.2 ng/µl ADN	1.60 µl	100 ng	8	56
5x Regulador HF	4 µl	1X	20	140
dNTP 10 mM	0.4 µl	0.2 µM	2	14
cebador V <u>5 µM</u>	2 µl	0.5 µM	10	-
cebador J <u>5 µM</u>	2 µl	0.5 µM	-	70
Phusion Hot Start	0.2 µl	0.02 U/µl	1	7
(18 µl /tubos)				

5 Ciclo térmico de la siguiente manera:

- 1- 98°C durante 1 min
- 2- 98°C durante 10 segundos
- 3- 55°C durante 30 segundos
- 4- 72°C durante 30 segundos, se pasa a la etapa 2, 30x
- 5- 72°C durante 5 min
- 6- 4°C pausa

Se agregaron 5 µl de 5X regulador Novex y se cargaron 6 µl en geles de agarosa (no se pudo ver nada). Se corrieron 15 µl en 6% de TBE, se tiñe con SYBR oro 12 min (Figura 58)

Configuración de cebador parte 2

10 La placa (VDJ_ADNg_Apri102_10) se diluyó a 400 µM. Las placas diluidas se hicieron a 5 µM cada una: 2 µl para cada V y J + 158 µl de H₂O = 160 µl, pero también a tubos agrupados a 5 µM una vez agrupados. V; 2 µl de cada (=90 µl en total) + 70 µl de H₂O = 160 µl a 5 µM (pozo F1). J; 2 µl de cada (=6 µl en total) + 154 µl de H₂O = 160 µl a 5 µM (pozo F2).

Prueba de cebador individual

15 Se probó cada cebador uno por uno (solapamiento no ilumina solamente): cada V contra un conjunto de los 3 J, (45 reacción de PCR); cada J contra un conjunto de todos los V (3 reacción de PCR); y todos los V y J en una reacción (1 reacción de PCR) (Figura 59).

Se preparó una reacción de la siguiente manera:

	1 tubo	5t (J y todo) 50t	
• dH ₂ O	9.8 µl (a 20 µl)	49	490

ES 2 730 951 T3

Se preparó una reacción de la siguiente manera: 1 tubo 5t (J y todo) 50t

• 62.2 ng/μl ADN	1.60 μl	100 ng	8	80
• 5x Regulador HF	4 μl	1X	20	140
• dNTP 10 mM	0.4 μl	0.2 μM	2	20
• <u>cebador V 5 μM</u>	2 μl	0.5 μM	10	-
• <u>5 μM</u> cebador J	2 μl	0.5 μM	-(10 todos)	100
Phusion Hot Start	0.2 μl	0.02 U/μl	1	10
(18 μl/tubos)				

El ciclo térmico fue el siguiente:

- 1- 98°C durante 1 min
- 2- 98°C durante 10 segundos
- 3- 58°C durante 20 segundos
- 4- 72°C durante 30 segundos, se pasa a la etapa 2, 30x
- 5- 72°C durante 5 min
- 6- 4°C pausa

Con saliente Illumina (Tabla 16).

5

Tabla 16.

Referencia	Pos	Nombre de secuencia	Secuencia	Tm (NaCl 50mM) C
51403096	A1	II_VDJ_IGHV_gDN_trunc135_01	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CCT GGA GTG GAT GGG GAT CA	71.5
51403097	A2	II_VDJ_IGHV_ADNg_trunc135_10	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GGC TGG AGT GGG TCT CAG TT	71.7
51403098	A3	II_VDJ_IGHV_ADNg_trunc135_11	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AGG GCT TGA GTG GAT GGG AC	70.9

ES 2 730 951 T3

Referencia	Pos	Nombre de secuencia	Secuencia	Tm (NaCl 50mM) C
51403099	A4	II_VDJ_IGHV_ADNg_trunc135_12	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AGA GCT TGG GTG GAT GGG AC	70.9
51403100	A5	II_VDJ_IGHV_ADNg_trunc135_02	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AAG GGG CTG GAG TGG GTT T	71.4
51403101	A6	II_VDJ_IGHV_ADNg_trunc135_03	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CCT GGA GTG GAT TGG GTA CAT CT	70.7
51403102	A7	II_VDJ_IGHV_ADNg_trunc135_04	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GGA AGG GGC TGG AGT GGA TT	71.8
51403103	A8	II_VDJ_IGHV_ADNg_trunc135_05	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AGG GTC TGG AGT GGG TCT CA	71.1
51403104	A9	II_VDJ_IGHV_ADNg_trunc135_06	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CTG GAG TGG CTT GCA CAC A	71.5
51403105	A10	II_VDJ_IGHV_ADNg_trunc135_07	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CTG GAG TGG ATG GGG AGG ATT	71.2
51403106	A11	II_VDJ_IGHV_ADNg_trunc135_08	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AAA GGC CCT GGA GTG GCT T	71.4
51403107	A12	II_VDJ_IGHV_ADNg_trunc135_09	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CCT GGA GTG GCT TGC ACT CA	71.8
51403108	B1	II_VDJ_IGHV_gDN_trunc135_13	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AGG GCT TGA GTG GAT GGG AG	70.8
51403109	B2	II_VDJ_IGHV_ADNg_trunc135_22	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CAG GGA AGG GAC TGG AAT ATG TTT C	69.8

ES 2 730 951 T3

Referencia	Pos	Nombre de secuencia	Secuencia	Tm (NaCl 50mM) C
51403110	B3	II_VDJ_IGHV_ADNg_trunc135_23	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GGA AAG GGC TGG AGT GGG TT	72.0
51403111	B4	II_VDJ_IGHV_ADNg_trunc135_24	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GAG TGG GTC TCT CTT ATT AGT TGG GA	69.8
51403112	B5	II_VDJ_IGHV_ADNg_trunc135_14	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GCT GGA GTG GGT CTC ATC CA	71.6
51403113	B6	II_VDJ_IGHV_ADNg_trunc135_15	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GGG CTG GAG TGG GTA GGT TT	71.8
51403114	B7	II_VDJ_IGHV_ADNg_trunc135_16	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GTT GGC CGT ACT AGA AAC AAA GCT	70.2
51403115	B8	II_VDJ_IGHV_gDN_trunc135_17	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CTG GAG TGG GTA TCG GGT GT	71.4
51403116	B9	II_VDJ_IGHV_ADNg_trunc135_18	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GAG TGG ATG GGA TTG GTG TGC	71.0
51403117	B10	II_VDJ_IGHV_ADNg_trunc135_19	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CTT GAG TGG ATG GGA GGT TTT GAT C	69.9
51403118	B11	II_VDJ_IGHY_ADNg_trunc135_20	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AGG CTT GAG TGG ATG GGA TGG	70.7
51403119	B12	II_VDJ_IGHV_ADNg_trunc135_21	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CCT GGA GTG GCT TGC TCA CA	71.8
51403120	C1	II_VDJ_IGHV_ADNg_trunc135_25	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AGT GGG TGG CAG TTA TAT GGT ATG A	69.5

ES 2 730 951 T3

Referencia	Pos	Nombre de secuencia	Secuencia	Tm (NaCl 50mM) C
51403121	C2	II_VDJ_IGHV_ADNg_trunc 135_34	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT TGA GTG GAT AGG ATG GAT CGT CG	69.9
51403122	C3	II_VDJ_IGHV_ADNg_trunc135_35	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GGG TTG GCC GTA TTA AAA GCA AAA C	70.0
51403123	C4	II_VDJ_IGHV_ADNg_trunc135_36	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT ACT GGA GTG GAT TGG GTA CAT CIA TT	69.3
51403124	C5	II_VDJ_IGHV_ADNg_trunc135_26	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AGG GCT TGA GTG GAT GGG AA	70.9
51403125	C6	II_VDJ_IGHV_ADNg_trunc135_27	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CTG GAG TGG ATT GGG TAC ATC TAT CA	69.9
51403126	C7	II_VDJ_IGHV_ADNg_trunc135_28	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CGC TTG AGT GGA TGG GAT GG	71.1
51403127	C8	II_VDJ_IGHV_ADNg_trunc135_29	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GGG TGG CCA ACA TAA AGC AAG A	71.0
51403128	C9	II_VDJ_IGHV_ADNg_trunc135_30	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GGG ACT GGA GTG GAT TGG GT	71.7
51403129	C10	II_VDJ_IGHV_ADNg_trunc135_31	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GCT AGA GTG GGT GGC AGT TAT ATC A	70.2
51403130	C11	II_VDJ_IGHV_ADNg_trunc135_32	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CTT GAG TGG ATG GGA TGG ATG AAC	70.1

ES 2 730 951 T3

Referencia	Pos	Nombre de secuencia	Secuencia	Tm (NaCl 50mM) C
51403131	C12	II_VDJ_IGHV_ADNg_trunc135_33	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AGT GGG TCT CAG CTA TTA GTG GTA G	69.2
51403132	D1	II_VDJ_IGHV_ADNg_trunc135_37	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GCT GGA GTG GGT CTC TGG TA	71.4
51403133	D2	II_VDJ_IGHV_ADNg_trunc135_38	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GCT GGT GTG GGT CTC ACG TA	71.5
51403134	D3	II_VDJ_IGHV_ADNg_trunc135_39	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GAG TGG GTC TCA GGT ATT AGT TGG A	70.1
51403135	D4	II_VDJ_IGHV_ADNg_trunc135_40	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT CAG GAA AAG GTC TGG AGT GGG T	71.1
51403136	D5	II_VDJ_IGHV_ADNg_trunc135_41	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT TGG AGT GGG TGG CAG TTA TAT CA	70.2
51403137	D6	II_VDJ_IGHV_ADNg_trunc135_42	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT GGG ACT GGA GTG GGT TTC ATA CA	70.9
51403138	D7	II_VDJ_IGHV_ADNg_trunc135_43	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AGG GCT TGA GTG GAT GGG ATG	70.7
51403139	D8	II_VDJ_IGHV_ADNg_trunc135_44	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT AGG GAA AGG GCT AGA GTT GGT AG	69.8
51403140	D9	II_VDJ_IGHV_ADNg_trunc135_45	ACA CTC TTT CCC TAC ACG ACG CTC TTC CGA TCT TCC CCA TCG AGA GGC CTT GA	71.6

ES 2 730 951 T3

Referencia	Pos	Nombre de secuencia	Secuencia	Tm (NaCl 50mM) C
51403141	E1	II_VDJ_IGHJ_ADNg_01	CTC GGC ATT CCT GCT GAA CCG CTC TTC CGA TCT CTT ACC TGA GGA GAC GGT GAC C	71.9
51403142	E2	II_VDJ_IGHJ_ADNg_02	CTC GGC ATT CCT GCT GAA CCG CTC TTC CGA TCT CTC ACC TGA GGA GAC AGT GAC C	71.9
51403143	E3	II_VDJ_IGHJ_ADNg_03	CTC GGC ATT CCT GCT GAA CCG CTC TTC CGA TCT CTT ACC TGA AGA GAC GGT GAC C	71.3

Configuración de cebador parte 2

5 La placa (VDJ_ADNg_April23_10_illumina) se diluyó a 400 µM. Las placas se suministraron a 10 nM, resuspender todos los oligos en 25 µl cada uno. No se probaron uno por uno, en su lugar se probó la intensidad de la banda en general, y también se combinaron los tubos a 5 µM una vez agrupados. V; 2 µl de cada (=90 µl en total) + 70 µl de H₂O = 160 µl a 5 µM (pozo F1). J; 2 µl de cada (=6 µl en total) + 154 µl H₂O = 160 µl a 5 µM (pozo F2) pruebas combinadas de superposición de cebadores de Illumina

Se prepara una reacción de la siguiente manera:

	1 tubo		4 tubos
• dH ₂ O	14.42 µl (a 25 µl)		57.68
• 180 ng/µl ADN	0.83 µl	150 ng	3.33
• 5x Regulador HF	5 µl	1X	20
• dNTP 10 mM	0.5 µl	0.2 µM	2
• cebador V <u>5 uM</u>	2 µl	0.5 µM	8
• cebador J <u>5 uM</u>	2 µl	0.5 µM	8
Phusion Hot Start	0.25 µl	0.02 U/µl	1
	(25 µl/tubos)		

10 El ciclo térmico se realizó de la siguiente manera:

- 1- 98°C durante 1 min
- 2- 98°C durante 10 segundos
- 3- 58°C durante 20 segundos
- 4- 72°C durante 20 segundos vaya a la etapa 2, 30x

ES 2 730 951 T3

5- 72°C durante 5 min

3- 4°C pausa

Qiagen purificado, eluido en 30 µl. Nanodrop

1 70 ng/µl sin saliente

2 128 ng/µl Saliente de Illumina

3 90 ng/µl Saliente Illumina, Veraseq

5 A 6 µl, se agregaron 6 µl de Blue juice 2X y se cargaron 6 µl. Se corrió con 2x 6 µl en 6% de TBE (20 min), se tiñó la mitad con bromuro de etidio (EtBR) y SYBR oro 12 min. Carga inversa (de modo que el número 3 es Phusion con cebador sin saliente) (Figura 60).

Se armó una reacción de la siguiente manera:

	1 tubo		5t
• dH ₂ O	10.85 µl (a 20 µl)		54.25
• 180 ng/µl ADN	0.55 µl	100 ng	-
• 5x Regulador HF	4 µl	1X	20
• dNTP 10 mM	0.4 µl	0.2 µM	2
• <u>cebador V 5 µM</u>	2 µl	0.5 µM	-
• <u>5 µM</u> cebador J	2 µl	0.5 µM	-
Phusion Hot Start	0.2 µl	0.02 U/µl	1
	(15.45 µl/tubos)		

El ciclo térmico se realiza de la siguiente manera (Figura 61):

1- 98°C durante 1 min

2- 98°C durante 10 segundos

3- 55°C durante 30 segundos

4- 55°C durante 30 segundos

5- 72°C durante 30 segundos vaya a la etapa 2, 30x

6- 4°C pausa

ES 2 730 951 T3

Repetido con ADNc de 62 ng/ μ l y ADN Coriell(diluido 1:5). Se probó a partir de ADNc de FV1 (RT-PCR realizado como en la secuenciación de la cadena pesada 454).

Se prepara una reacción como sigue:

		1 tubo	
• dH ₂ O	6.4 μ l	(a 20 μ l)	
• ADNc	5 μ l	100	ng
• 5x Regulador HF	4 μ l	1X	
• dNTP 10 mM	0.4 μ l	0.2	μ M
• cebador V <u>5 μM</u>	2 μ l	0.5	μ M
• cebador J <u>5 μM</u>	2 μ l	0.5	μ M
Phusion Hot Start	0.2 μ l	0.02	U/ μ l
(18 μ l/tubos)			

5 Ciclo térmico se realizó de la siguiente manera:

- 1- 98°C durante 1 min
- 2- 98°C durante 10 segundos
- 3- 58°C durante 20 segundos
- 4- 72°C durante 20 segundos vaya a la etapa 2, 25x
- 5- 72°C durante 5 min
- 6- 4°C pausa

Se cargaron 5 μ l en gel de TBE al 6%, se corrieron 25 minutos, se tiñeron con 10 μ l de EtBr 15 minutos (Figura 62).

Ejemplo XII

Secuenciación inmune de la artritis reumatoide

10 La información de la muestra se expone en la Tabla 17.

CCP+ RA															
ID de Espécimen	Tipo de Esp.	Des de Esp.	Diagnóstico	Género	Edad	RF	Datos de RF	RF más alto	Datos de RF más alto	CCP	Datos de CCP	CCP más alto	Datos de CCP más alto	CRP	Datos de CRP
DL-SF-0306	S F	Rodilla	RA	F	6 3	48	6/2/2011	59	8/4/2008	> 1 0 0	6/2/2011	>10 0	6/2/2011	11.6	6/2/2011
DL-SF-0331	S F	Rodilla	RA	F	5 9	17 3	9/14/2009	17 3	9/14/2009	> 1 0 0	9/14/2009	>10 0	9/14/2009		
DL-SF-0263	S F	Rodilla	Psoriático	M	2 6	48	4/3/2009	48	3/12/2009	> 1 0 0	3/12/2009	>10 0	3/12/2009		
DL-SF-0375	S F	Rodilla	RA + Sec OA	M	6 0	15 4	10/3/2007	23 3	5/2/2004	> 1 0 0	10/3/2007	>10 0	10/3/2007	82.6	10/3/2007
Sin RA															
ID de Espécimen	Tipo de Esp.	Des de Esp.	Diagnóstico	Género	Edad	RF	Datos de RF	RF más alto	Datos de RF más alto	CCP	Datos de CCP	CCP más alto	Datos de CCP más alto	CRP	Datos de CRP
DL-SF-0291	S F	Rodilla	OA	M	0										
DL-SF-0237	S F	Rodilla	OA	M	0										
DL-SF-0313	S F	Rodilla	Gota	F	7 3										
DL-SF-0264	S F	Rodilla	OA	F	7 7										

Tabla 17

Extracción de ARN

Se utilizó mirvana según el protocolo del fabricante. Nanodrop QC. Diluido todo a 250 ng/μl (Tabla 18).

Tabla 18

ID	ng/ul ARN	260/280	260/230	dilución 1/3	Volumen para 500 ul	Agua a 7
1	768.6	2.06	1.4	260.6	1.92	5.08
2	644.7	2.09	2.05	213.1	2.35	4.65
3	349.3	2.07	2.05	112.0	4.46	2.54
4	181.7	2.04	1.7	-	2.75	4.25
5	434.6	2.1	1.11	141.6	3.53	3.47
6	1071.3	2.09	1.59	360.4	1.39	5.61
7	597	2.04	1.82	199.2	2.51	4.49
8	18.7/134.0	1.88	1.13	-	3.73	3.27

ES 2 730 951 T3

La muestra concentrada 8 por precipitación con EtOH, se volvió a suspender en 20 ml, Nanodrop = 134.0. Transcripción reversa. Se procedió de la siguiente manera para RT-PCR:

	<u>2x 1 tubo</u>
• H ₂ O (DEPC)	0 µl (a 12 µl)
• 2 µM de cebador específico del gen (IGHC K y L)	5 (5 pmol)
• Total ARN	7 µl (500 ng)

5 Se calienta a 95°C durante 1 min seguido de 65°C durante 5 min, y luego en hielo durante 1 min. Se centrifugó y añadió lo siguiente:

	<u>1 tubo</u>	<u>10 tubos</u>
• 5x regulador de primera hebra	4 µl	40
• Mezcla dNTP 10 mM	1 µl	10
• 0.1 MDTT	1 µl	10
• Inhibidor de la RNasa-Enzy.	1 µl	10
• Superíndice III	1 µl	10

Incubación a 55°C durante 60 min. (8 µl/tubos). Enzima inactivada por calentamiento a 70°C durante 15 min. Se eliminó el híbrido ARN/ADN agregando 1 µl de RNasaH de E. coli. Se incubó a 37°C durante 20 min., luego hielo. La reacción de PCR se prepara de la siguiente manera (se utiliza la mitad para la PCR, se mantiene la otra mitad como respaldo)

	<u>1x x 2 por muestra</u>	<u>x 12 muestras (16t)</u>
• dH ₂ O	21µl (a 50 µl)	420
• cADN	5 µl	-
• 5x Regulador HF	10 µl	240
• dNTP(10 mM)	1 µl	24
• cebador alto (IGHV nuevo-corto) (4 µM)	6.25 µl (25 pmol)	150
• cebador bajo (IGHC nuevo-corto)(4 µM)	6.25 µl (25 pmol)	150
• Phusion	0.5 µl	12

10

Se agregaron 45 µl a 5 µl dos veces para cada muestra de ADNc. Ciclo térmico de la siguiente manera:

- 1- 98°C durante 1 min.
- 2- 98°C durante 10 segundos

ES 2 730 951 T3

- 3- 62°C durante 20 segundos
- 4- 72°C durante 20 segundos se pasa a la etapa 2, 23x
- 5- 72°C durante 5 min.
- 6- 4°C pausa

Se reunieron los tubos (total 100 µl). Se agregaron 2 µl de exonucleasa I (20 U/µl) a cada tubo de 100 µl y se incubaron a 37°C durante 20 min. Se purificó con AMPure XP, relación 1.8:1. Se resuspendió en 40 µl, se transfirió a la tira de PCR. Se realizó la purificación SPRI de la siguiente manera:

- 5 • Se agregó la cantidad requerida de perlas AMPure XP (relación 1.8:1) a la muestra de ADN en el regulador EB.
- Se somete a vórtice para mezclar.
- Incubar durante 5 minutos a temperatura ambiente.
- Imán (MPC) durante 5 minutos. Se dejó el tubo de perlas en el MPC durante todos los pasos de lavado.
- 10 • Se eliminó el sobrenadante (se mantuvo en caso de falla) y se lavaron las perlas dos veces con 500 µl de etanol al 70%, se incubaron durante 30 segundos cada vez.
- Se eliminó todo el sobrenadante, se centrifugó rápidamente, se eliminó la última gota y se dejó que las perlas de AMPure se secaran al aire por completo (2 min.).
- 15 • Se retiró el tubo del MPC, se agregaron 40 µl de Tris-HCl 10 mM, EDTA 0.1 mM, pH 8.0 (o Regulador EB de Qiagen) y se agitó en vórtice durante 30 segundos. Para resuspender las perlas, dejar reposar durante 3 min. Aplicar imán por 2 min y transferir el sobrenadante a un nuevo tubo.

Se usó un gel de diagnóstico para verificar la eficacia de la PCR. Se realizaron 2 µl en un 2% de Egel-X durante 12 min. (Figura 68). Las reacciones de bloqueo se realizaron utilizando el kit de reparación Enzymatics End de la siguiente manera:

	1t	12t
• H ₂ O	0 µl(a 50 µl)	0
• ADN purificado	38 µl	-
• 10 × Regulador de reparación End	5 µl	60
• mezcla dNTP 1 mM	5 µl	60
• Mezcla de enzimas.	5 µl	60

- 20 Se incubó a una temperatura de 25°C durante 30 min. Se inactivó con calor a 75°C durante 20 min. Se purificó con AMPure XP, relación 1.8:1. Se resuspendió en 37 µl (se mantuvo como volumen de respaldo sin vinculación). Se guardaron 2 µl de cada muestra para diagnóstico de gel. Se corrió un gel de diagnóstico para verificar la eficiencia del bloqueo y las perlas AMPure. Se corrieron 2 µl de muestra en un 2% de Egel-X durante 12 min. (Figura 69). Se purificó con AMPure XP para eliminar todo el dNTP, se eluyó en 44 µl. E Gel diagnóstico 2 µl. Se probaron los siguientes
- 25 pasos solo en una muestra: (muestra 10 FV1, la que tiene la banda adicional a 275 pb).

Relave en A - solo muestra FV1

Relave Klenow en exo- "A" y "T". Se preparó la siguiente mezcla de reacción:

- H₂O
- 42 µl (a 50 µl)

ES 2 730 951 T3

- Muestra de ADN 41 µl
- 10X regulador Klenow 5 µl
- dATP 10 mM 1 µl
- Klenow exo (3' a 5' exo menos) 3 µl

5 Incubado durante 30 minutos a 37°C. Purificado con AMPure, eluido en 26 µl. Corrido en gel de diagnóstico (Figura 70). Ln1 post A-tail, ln2, segunda ronda de AMPure de sobrenadante postenlace no finaliza. Todo se veía bien. Sin pretender estar limitados por la teoría científica, las bandas de gel anteriores de FV1 (muestras 9 y 10) probablemente se disiparon debido a una mala carga del gel.

Ligación con adaptador Y

Autofusión de oligo 454 en adaptadores A y B juntos. Se siguió el protocolo 454 (10 µl de cada cebador 100 µM + 30 µl de (Tris 10 mM, EDTA 0.1 mM a NaCl 50 mM)) (cada adaptador estaba a 20 µM final), 95°C durante 3 minutos, rampa a 15°C perpetua a 0.1°C/seg.

10 Ligación del adaptador rápido de 454 Y de la siguiente manera (se mantuvieron 2 µl de ADN en el lado para la comparación del gel):

- | | |
|---------------------------------|----------------|
| | 1 tubo |
| • H ₂ O | 0 µl (a 50 µl) |
| • 2x regulador de ligasa rápida | 25 µl |
| • VDJ ADN | 23 µl |
| • Adaptador de 20 µM | 1 µl |
| • T4 ADN ligasa (rápida) | 2 µl |

Incubado en PCR a 22°C durante 15 min. Se purificó con AMPure XP, relación 1.8:1. Resuspendido en 25 µl. Se corrió la muestra en un gel de diagnóstico.

15 Se probaron todas las reacciones combinadas, como en el protocolo rápido 454. Se usó la muestra 9 (FV1 para esto), todavía no se había ejecutado Endit (por causa de la necesidad de un regulador en la mezcla). Se realizó una reacción de bloqueo con el kit de reparación Enzymatics End de la siguiente manera (Endit y relave en A simultáneos):

- | | |
|------------------------------------|---------------------|
| | 1 tubo |
| • H ₂ O | 0 µl(a 50 µl) |
| • ADN purificado | 40 µl |
| • 10 × regulador de reparación End | 5 µl |
| • Mezcla dNTP 1 mM | 5 µl |
| • Mezcla de enzimas. | 2 µl (T4 pol + PNK) |
| • TAQ polimerasa | 2 µl |

ES 2 730 951 T3

Se incubó a una temperatura de 25°C durante 20 minutos, luego a 72°C durante 20 minutos, luego se mantiene a 4°C. La ligación del adaptador en Y, se agrega directamente a la reacción de la siguiente manera:

• ADN relavado en A Endit	50 µl
• 2x regulador de ligasa rápida	50 µl
• Adpatador de 20 µM	1 µl
• T4 ADN ligasa (rápida)	2 µl

5 Incubado en PCR a 22°C durante 15 min. (el volumen total fue de 107 µl), por lo que 1.8 AMPure = 193 µl. Se purificó con AMPure XP, relación 1.8:1. Resuspendido en 25 µl. Se realizó un diagnóstico de gel en la muestra (Figura 71), 2% de E-gel, corrió 16 min. Carril 1, marcador; carril 2, control de preligación; carril 3, relave Klenow A; carril 4, relave Taq A.

10 Las bandas adicionales en el carril 2 fueron las bandas no específicas de la ARNasa H que se ligaron. Ahora la pregunta era saber si la ligación mostraría un adaptador Y frente a 2 adaptadores Y. Cebador A 41 nt, cebador B 43 nt = total 84. Sin pretender estar vinculado por la teoría científica, esto significaría que la primera banda sería una ligación adecuada y la otra anterior sería algo diferente porque el producto final es de 100 pb o más, a menos que la FAM esté causando este patrón de migración. Pero, idealmente, un conjunto de cebadores fo-fo, seguido en la PCR por un ciclo de TAQ, probablemente sería el camino a seguir ... pero la fusión competiría por un relave de taq. Pruebe la ligación del adaptador en Y en la escalera eterna y pruebe algunos tamaños. (No hay una preocupación de relave aquí). Luego, prueba Taq A de relave secuencial vs. mezclado con el final bloqueado.

Prueba de PCR

Se utilizaron cebadores (Tabla 19) que amplificaban el producto final, por lo que solo la ligación de doble adaptador debería amplificarse exponencialmente. Comenzó con una dilución 1/10, realizó una PCR de 15 ciclos y se vio qué sucedía en el gel.

20

Tabla 19.

Referencia	Purificación	Nombre de la secuencia	Secuencia	Tm
53164334	Desalado estándar	454 PCR superior rápida	CCA TCT CAT CCC TGC GTG TCT CC	61.4
53164335	Desalado estándar	454 PCR baja rápida	CCT ATC CCC TGT GTG CCT TGA GAG	60.9

Se prepara una reacción de prueba de PCR de la siguiente manera:

	1x	5x
• dH ₂ O	15.75 µl (a 25 µl)	78.75
• Dilución 1/10 de ADN.	1 µl	-
• 5x Regulador HF	5 µl	25
• dNTP(10 mM)	0.5 µl	2.5
• cebador alto 10 µM	1.25	6.25
• cebador bajo 10 µM	1.25	6.25

ES 2 730 951 T3

	1x	5x
• Phusion	0.25 µl	1.25

Ciclo térmico de la siguiente manera:

- 1- 98°C durante 1 min.
- 2- 98°C durante 10 segundos
- 3- 65°C durante 20 segundos
- 4- 72°C durante 20 segundos se pasa a la etapa 2, 15x
- 5- 72°C durante 5 min.
- 6- 4°C pausa

5 Se cargaron muestras de 2 µl y 20 µl en un 2% de gel ex y se corrieron durante 16 minutos. (Figura 72). Muestra 1, control de preligación; muestra 2, relave Klenow A; muestra 3, relave Taq A; muestra 4, sin plantilla de control de PCR. Sin la intención de limitarse por la teoría científica, parecía que la banda superior era la correcta, la ausencia de FAM allí la haría migrar ligeramente más abajo. Una extracción lo confirmaría de una vez por todas, pero el enriquecimiento resultante se veía muy bien. (Se reordenó el cebador con un 5' FAM en el cebador superior y ambos purificados por HPLC).

10 Se realizó una prueba de extracción de preparación de Pippin. El patrón de extracción de la banda fue el siguiente:

1	475	450-500
2	500	475-525 muy bien hecho
3	538	500-575
4	500	460-540 Escalera de 100 pb (7 µl Pippin) extracto de 500 pb

Se cargaron 20 µl de la cámara de elución resultante (60-80 µl) en un 2% de Exgel (Figura 73. Panel anterior izquierdo, panel derecho, posterior del canal. Carril 1, control FV-µl de VDJ, HC purificado. Extracción por PCR validada. 1. control negativo FV-µl (usado 1 µl). 2. 475-525 (hipo: solo un adaptador Y). 3. 500-550 (hipo: adaptador 2 Y).

15 Se preparó una reacción de prueba de PCR como sigue

	1x	4
• dH ₂ O	11.7 5µl (a 25 µl)	47
• Dilución de la elución en Pippin del ADN	5 µl	-
• 5x Regulador HF	5 µl	20
• dNTP(10 mM)	0.5 µl	2
• cebador alto 10 µM	1.25	5

ES 2 730 951 T3

	1x	4
• cabdor bajo 10 µM	1.25	5
• Phusion	0.25 µl	1

Ciclo térmico de la siguiente manera (20 µl/tubo):

- 1- 98°C durante 1 min.
- 2- 98°C durante 10 segundos
- 3- 65°C durante 20 segundos
- 4- 72°C durante 20 segundos se pasa a la etapa 2, 15x
- 5- 72°C durante 5 min.
- 6- 4°C pausa

Se cargaron muestras de 2 µl y 20 µl en un Ex-gel, se corrió 16 min. (Figura 74).

5 Optimizar un relave en A

Se intentó agregar Klenow o Taq justo después de la finalización (sin limpieza), con un exceso de dATP y regulador nuevo. Luego se combina con o sin limpieza de ligación previa. Muestra de postligación usada para pruebas iniciales. Se obtuvieron dos bandas de buen tamaño para evaluar la eficiencia: 1, klenow-lig; 2, klenow- limpieza + lig; 3, Taq lig; 4, Taq limpieza + lig; 5, bst largefrag lig; 6, bst largefrag limpieza + lig.

10 Se utilizó un FV1 de cadena pesada, posterior a la PCR, posterior a la limpieza. Reacción de división en 6 para la prueba. Se realiza la reacción de bloqueo utilizando el kit de reparación Enzymatics End de la siguiente manera (¿puede usarse solo la mitad del producto de PCR?):

	1t	<u>8</u>
• H ₂ O	0 µl(a 25 µl)	104
• ADN purificado	19 µl (6 DNA + 13 agua)	-
• 10 × Regulador de reparación End	2.5 µl	20
• Mezcla dNTP 1 mM	2.5 µl (0.1 mM)	20
• Mezcla de enzimas (HC)	1 µl	8

15 Se incubó a una temperatura de 25°C durante 30 min. (19/t). Se inactivó con calor a 75°C durante 20 min. Relave en A agregando lo siguiente directamente a la mezcla (también se hizo con TAQ y BST con los reguladors adecuados):

	1t	2t
• H ₂ O	20 µl (agregar 25 a 50 µl)	40
• Muestra de ADN bloqueado	25 µl	50

ES 2 730 951 T3

	1t	2t
• 10X regulador Klenow	2.5 µl	5
• dATP 10 mM	0.5 µl	1
• Klenow exo HC (3' a 5' exo menos)	2 µl	4

5 La reacción total fue ahora de 50 µl. Klenow, se incubó a 37°C durante 30 min, inactivado por calor a 75°C durante 20 min. Taq, se incubó a 72°C durante 30 min. BST, se incubó a 65°C durante 30 min, inactivado por calor a 80°C durante 10 min. Se divide cada reacción en 2 tubos de 50. Se purificó por AMPure la mitad de cada reacción, se eluyó en 50 µl. La otra mitad se corrió directamente en la ligación.

Ligación del adaptador Y. Los siguientes se añadieron directamente a la reacción:

	1t	7t
• H ₂ O	0 (agregar µl a 100 µl)	
ADN con relave en A	50 µl	-
• 2x regulador de ligasa rápida	46 µl	322
• Adaptador 20 µM	2 µl	14
• T4 ADN ligasa (rápido)	2 µl	14

Incubado en PCR a 22°C durante 15 min. La reacción total fue entonces de 100 µl. AMPure, relación 1.8. Eluido en 25 µl (Figura 75).

10 Prueba de preparación Pippin 2 (cantidad de carga)

15 Se agruparon todas las muestras. Nanodrop: la relación de 25 ng/µl era muy alta (¿FAM?). Carga en la preparación Pippin en diferentes cantidades. Se conservó una parte para correr como control negativo. Extracto del rango 475 a 550. Cargado en Pippin Nanodrop post-Pippin. Muestra 1, 2.5 µl; muestra 2, 5; muestra 3, 10; muestra 4, 20; muestra 5, ref. AMPure XP, eluido en 20 µl. Nanodrop a todos 0. Se cargaron 5 µl en 2% de E-gel Ex (Figura 76). No parecía que se había extraído el tamaño correcto. Sin pretender estar sujetos a la teoría científica, esto puede deberse al hecho de que se usó 475-550 en lugar de 475-525.

Finalizar biblioteca

El siguiente protocolo fue diseñado para finalizar la biblioteca restante. La placa se configuró de la siguiente manera (Tabla 20):

20

Tabla 20

Tipo de biblioteca (todas las cadenas pesadas)	Colocación de fila e información de BC 1=13, etc.
Triple VDJ-HC GMC	A1-A10
Triple VDJ-HC IDO	B1-B10
Triple VDJ-HC FV	C1-C10
RA	D1-D8

ES 2 730 951 T3

Tipo de biblioteca (todas las cadenas pesadas)	Colocación de fila e información de BC 1=13, etc.
VIH 1/2	E1-E8 (5 progresores +5 negativo) BC; 1-2-3-4-5/6-7-8-9-10
VIH 2/2	F1-F7 (5 elite) BC; 1-2-3-4-5

Se realiza la reacción de bloqueo utilizando el kit de reparación Enzymatics End de la siguiente manera (usar solo la mitad del producto de PCR):

	1t	<u>60</u>
• H ₂ O	0 µl (a 25 µl)	0
• ADN purificado post-PCR	19 µl	-
• 10 × Regulador de reparación End	2.5 µl	150
• Mezcla dNTP 1mM	2.5 µl (0.1 mM)	150
• Mezcla enzimática (HC)	1 µl	60

- 5 Se incubó a una temperatura de 25°C durante 30 min. (19/t). Se inactivó con calor a 75°C durante 20 min. Prelave en A agregando lo siguiente directamente a la mezcla:

	1t	60t
• H ₂ O	20 µl(a 50 µl)	1200
• Muestra de ADN bloqueado	25 µl	-
• 10X regulador Klenow	2.5 µl	150
• dATP 10 mM	0.5 µl	30
• Klenow exo HC (3' a 5' exo menos)	2 µl	120

La reacción total fue entonces de 50 µl (25/t). Se incubó a 37°C durante 30 min. Se inactivó con calor a 75°C durante 20 min.

- 10 Preparados 454 adaptadores en Y. 10 µl de de cada cebador A y B 100 µM + 30 µl de (Tris 10 mM, EDTA 0.1 mM a NaCl 50 mM) (cada adaptador estaba en 20 µM final). Se incubó a 95°C durante 3 minutos, descienda hasta 15°C perpetua a 0.1°C/seg.

Ligación del adaptador Y. Se añadió lo siguiente directamente a la reacción:

	1t	2x30t
• H ₂ O	0 (agregar µl a 100 µl)	
ADN prelave en A	50 µl	-
2x regulador de ligasa rápida	46 µl	1380

	1t	2x30t
• Adaptador de 20 µM	2 µl	-
• T4 ADN ligasa (rápida)	2 µl	60

Incubado en PCR a 22°C durante 15 min. (48/t). La reacción total fue entonces de 100 µl. Se aplicó AMPure, se eluyó en 40 µl (utilizado el liquidador). Biblioteca de QC para la agrupación. Probado en muestras de VIH, probó solo las primeras 5 muestras (progresor).

- 5 El apéndice A presenta cebadores ScFv. El apéndice B establece los métodos de generación de SOE-PCR y ScFV a partir de células individuales. El apéndice C establece los métodos de análisis de VDJome.

Cebadores ScFv

Del cuaderno de notas

Contenidos
▪ (para estrategia de PCR anidada)
▪ 1.1 Info
▪ 1.2 Placas
▪ 1.3 Mezclando
_attB (clonación de Gateway)
▪ 2.1 Info
▪ 2.2 Secuencias
Información de pedido
Información de mezcla
_scFv_USUARIO y USER
▪ 3.1 Info
▪ 3.2 Secuencias
• 3.3 Información de pedido
▪ 3.4 Información de mezcla
_SOE_PEDS_11_405_LLA4
▪ 4.1 Info
▪ 4.2 Secuencias

Contenidos
▪ 4.3 Información de placa
▪ 4.4 Información de mezcla
▪ 5.1 Info
▪ 5.2 Secuencias
▪ 5.3 Información de placa
▪ 5.4 Información de mezcla
_SOE_PEDS_11_405_LLA4
▪ 6.1 Info
▪ 6.2 Secuencias
▪ 6.3 Información de placa
▪ 6.4 Información de mezcla
) y _SOE_PNAS_85_5879_G4S3

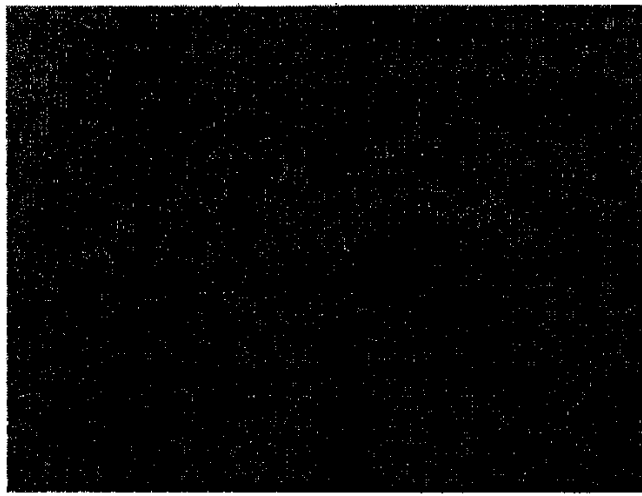
(para estrategia de PCR anidada)

Información▪ Estos son los mismos cebadores utilizados para el proyecto VDJ-oma original

▪ Ver VDJ-oma_Cebadores para más detalles sobre su diseño

Placas

5 Estos cebadores se ordenan en dos palcas, dispuestas como:





- Los archivos excel para IDT son:

▪ Medios	_izquierda.xls	
▪: Medios	I_derecha_	_RT.xls

- NOTA: Esta placa solo tiene 16 cebadores, por lo que IDT no me permite ordenarla. Lo ordené en tubos separados en su lugar.

5 Mezcla

- Las placas se ordenan a 200 µM.
- Los tubos individuales se ordenan a LabReady a 100 µM
- Para las placas: Para hacer 2.5 µM de cada cebador, agregar 1 µL de patrón a 80 µL de volumen total de cada cebador

10 ▪ Para los tubos: Para hacer 2.5 µM de cada cebador, agregar 2 µL de patrón a 80 µL de volumen total de cada cebador

- Nota, IGHV, IGK/LV. IGHC. IHK/LC son cuatro mezclas de cebadores diferentes. Bly attB (clonación de la puerta de enlace)

Información

- El diseño de cebador está activado.

15 ▪ Estos son los cebadores "externos" de las etiquetas attB agregadas para la clonación de la puerta. Secuencias

- Las secuencias del cebador _attB son:

```

>20100505_IGHV_left_attB1_1
ggggacaagtttgtacaaaaaagcaggcttcGAGGTGCAGCTGGTGGAGTCCG
>20100505_IGHV_left_attB1_2
ggggacaagtttgtacaaaaaagcaggcttcGAGGTGCAGCTGGTGGAGTCTGG
>20100505_IGHV_left_attB1_3
ggggacaagtttgtacaaaaaagcaggcttcCAGGTTCACCTTGAGGGAGTCTGGTCC
>20100505_IGHV_left_attB1_4
ggggacaagtttgtacaaaaaagcaggcttcCAGGTTCAGCTGTTGCAGCTGG
>20100505_IGHV_left_attB1_5
ggggacaagtttgtacaaaaaagcaggcttcCAGGTGCAGCTACAGCAGTGGGG
>20100505_IGHV_left_attB1_6
ggggacaagtttgtacaaaaaagcaggcttcCAGGTGCAGCTGGTGCATCTGG
>20100505_IGHV_left_attB1_7
ggggacaagtttgtacaaaaaagcaggcttcCAGGTTCACCTTGAAGGAGTCTGGTCC
>20100505_IGHV_left_attB1_8
ggggacaagtttgtacaaaaaagcaggcttcCAGGTTCAGCTGGTACAGTCTGGG
>20100505_IGHV_left_attB1_9
ggggacaagtttgtacaaaaaagcaggcttcCAGGACCAGTGGTGCAGTCTGGG
>20100505_IGHV_left_attB1_10
ggggacaagtttgtacaaaaaagcaggcttcCAGGTGCAGCTGGTGGAGTCTGG
>20100505_IGHV_left_attB1_11
ggggacaagtttgtacaaaaaagcaggcttcCAGATGCAGCTGGTGCAGTCTGG
>20100505_IGHV_left_attB1_12
ggggacaagtttgtacaaaaaagcaggcttcCAATGCAGCTGGTGCAGTCTGGG
>20100505_IGHV_left_attB1_13
ggggacaagtttgtacaaaaaagcaggcttcGAAGTGCAGCTGGTGGAGTCTGGG
>20100505_IGHV_left_attB1_14
ggggacaagtttgtacaaaaaagcaggcttcCAGGTGCAGCTGGTGCAGTCTG
>20100505_IGHV_left_attB1_15
ggggacaagtttgtacaaaaaagcaggcttcGAGGATCAGCTGGTGGAGTCTGGG
>20100505_IGHV_left_attB1_16
ggggacaagtttgtacaaaaaagcaggcttcCAGGTTCAGCTGGTACAGTCTGGG
>20100505_IGHV_left_attB1_17
ggggacaagtttgtacaaaaaagcaggcttcCAGGTGCAGCTGCAGGAGTCC
>20100505_IGHV_left_attB1_18
ggggacaagtttgtacaaaaaagcaggcttcGAGGTGCAGCTGGTGGAGTCTGG
>20100505_IGHV_left_attB1_19
ggggacaagtttgtacaaaaaagcaggcttcCAGGTGCAGCTGGTGCAGTCTG
>20100505_IGHV_left_attB1_20
ggggacaagtttgtacaaaaaagcaggcttcCAGGTACAGCTGCAGCAGTCAAGT
>20100505_IGHV_left_attB1_21
ggggacaagtttgtacaaaaaagcaggcttcGAGATGCAGCTGGTGGAGTCTGGG
>20100505_IGHV_left_attB1_22
ggggacaagtttgtacaaaaaagcaggcttcGAGGTGCATCTGGTGGAGTCTGGG
>20100505_IGHV_left_attB1_23
ggggacaagtttgtacaaaaaagcaggcttcGAGGTGCAGCTGGTGGAGTCTGG
>20100505_IGHV_left_attB1_24
ggggacaagtttgtacaaaaaagcaggcttcGAGGTGCAGCTGGTGGAGTCTGG
>20100505_IGHV_left_attB1_25
ggggacaagtttgtacaaaaaagcaggcttcCAGGTCCAAGTGGTGTAGTCTGGAGC
>20100505_IGHV_left_attB1_26
ggggacaagtttgtacaaaaaagcaggcttcGAGGTGCAGCTGGTGCAGTCTG
>20100505_IGHV_left_attB1_27
ggggacaagtttgtacaaaaaagcaggcttcGAGGTGCAGCTGGTGGAGTCTCG
>20100505_IGHV_left_attB1_28
ggggacaagtttgtacaaaaaagcaggcttcGAGGTTCAGCTGGTGCAGTCTGGG
>20100505_IGHV_left_attB1_29
ggggacaagtttgtacaaaaaagcaggcttcCAGGTGCAGCTGGTGCAGTCTG
>20100505_IGHV_left_attB1_30
ggggacaagtttgtacaaaaaagcaggcttcGAAGTGCAGCTGGTGCAGTCTGG
>20100505_IGHV_left_attB1_31
ggggacaagtttgtacaaaaaagcaggcttcGAGGTGGAGCTGATAGAGTCCATAGA
>20100505_IGHV_left_attB1_32
ggggacaagtttgtacaaaaaagcaggcttcCAGTGCAGCTGGTGGAGTCTGG
>20100505_IGHV_left_attB1_33
ggggacaagtttgtacaaaaaagcaggcttcGAGGTGCAGCTGGAGGAGTCTGG
>20100505_IGHV_left_attB1_34
ggggacaagtttgtacaaaaaagcaggcttcGAGGTACAGCTGGTGGAGTCTGAAGA
>20100505_IGHV_left_attB1_35
ggggacaagtttgtacaaaaaagcaggcttcCAGGTGCAGCTGCAGGAGTCC
>20100505_IGHV_left_attB1_36
ggggacaagtttgtacaaaaaagcaggcttcGAGGTGCAGCTGGTGGAGTCTGGG
>20100505_IGHV_left_attB1_37
ggggacaagtttgtacaaaaaagcaggcttcCAGGTGCAGCTGGGGCAGTC
>20100505_IGHV_left_attB1_38
ggggacaagtttgtacaaaaaagcaggcttcCAGCTGCAGCTGCAGGAGTCC
>20100505_IGHV_left_attB1_39
ggggacaagtttgtacaaaaaagcaggcttcCAGGTTCAGCTGGTGCAGTCTGGA
>20100505_IGHV_left_attB1_40

```



```

|ggggacaagtttgtacaaaaaagcaggcttcGAGGTGCAGCTGGTAGAGTCTGGG
|>20100505_IGHV_left_attB1_41
|ggggacaagtttgtacaaaaaagcaggcttcCAGGTCCAGCTTGTGCAGTCTGGG
|>20100505_IGHV_left_attB1_42
|ggggacaagtttgtacaaaaaagcaggcttcGAGGTGCAGCTGTTGCAGTCTGC
|>20100505_IGHV_left_attB1_43
|ggggacaagtttgtacaaaaaagcaggcttcGAGGTACAACCTGGTGGAGTCTGGGGG
|>20100505_IGHV_left_attB1_44
|ggggacaagtttgtacaaaaaagcaggcttcCAGATCACCTTGAAGGAGTCTGGTCC
|>20100505_IGHV_left_attB1_45
|ggggacaagtttgtacaaaaaagcaggcttcCAGGTACAGCTGATGCAGTCTGGGG
|
|>20100505_IGKJ_right_attB2_1
|ggggaccactttgtacaagaaagctgggtcTTTGATCTCCACCTTGGTCCCTCCGC
|>20100505_IGKJ_right_attB2_2
|ggggaccactttgtacaagaaagctgggtcTTTGATCTCCAGCTTGGTCCCTCGG
|>20100505_IGKJ_right_attB2_3
|ggggaccactttgtacaagaaagctgggtcTTTGATATCCACTTGGTCCCAGGGC
|>20100505_IGKJ_right_attB2_4
|ggggaccactttgtacaagaaagctgggtcTTTGATTCCACCTTGGTCCCTTGGC
|>20100505_IGKJ_right_attB2_5
|ggggaccactttgtacaagaaagctgggtcTTTAATCTCCAGTCGTGCCCTTGGC
|>20100505_IGLJ_right_attB2_1
|ggggaccactttgtacaagaaagctgggtcGAGGACGGTCACCTTGGTGCCA
|>20100505_IGLJ_right_attB2_2
|ggggaccactttgtacaagaaagctgggtcTAGGACGGTCAGCTGGTCCCTCC
|>20100505_IGLJ_right_attB2_3
|ggggaccactttgtacaagaaagctgggtcGAGGACGGTCAGCTGGGTGCC
|>20100505_IGLJ_right_attB2_4
|ggggaccactttgtacaagaaagctgggtcTAAAATGATCAGCTGGGTCCCTCCAC
|>20100505_IGLJ_right_attB2_5
|ggggaccactttgtacaagaaagctgggtcTAGGACGGTGACCTTGGTCCCAGT
|>20100505_IGLJ_right_attB2_6
|ggggaccactttgtacaagaaagctgggtcTAGGACGGTCAGCTCGGTCCCC
|
|-----

```

- Los cebadores de enlace solamente (obsérvese el truncamiento de uno de ellos para que coincida con Tm):

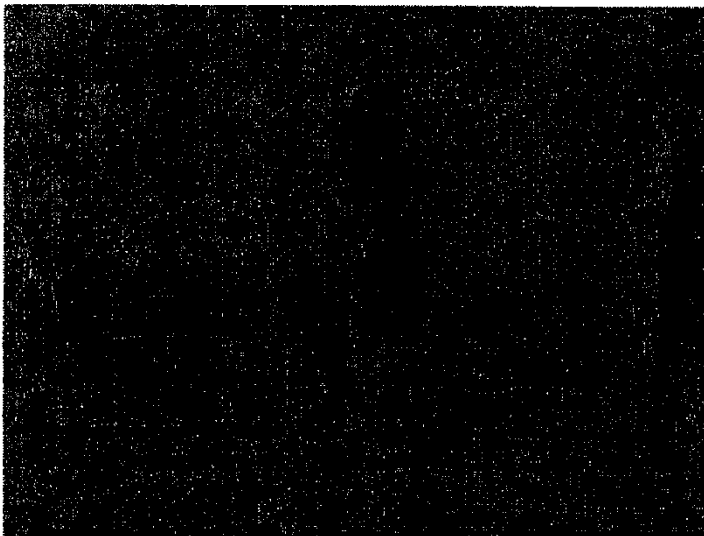
```

|-----
|>attB1_left (Tm: 61.7)
|GGGGACAAGTTTGTACAAAAAAGCAGGCTTC
|
|>attB2_right (Tm: 62.8)
|GGGGACCACCTTTGTACAAGAAAGCTGGG
|-----

```

Información de pedido

- 5 ▪ La placa está dispuesta de la siguiente manera:



- El archivo de Excel para ordenar está aquí:

Medios	attB.xls
--------	----------

Información de mezcla

- Todos los cebadores se ordenan a 200 µM. Hacemos mezclas a 2.5 µM para cada cebador.
- Por cada 80 µL de mezcla, agregar µL de patrón de cada cebador (diluído en Tris-HCl)

_scFv_USUARIO Y _USUARIO

5 Información

- Diseño de cebador.
- Estos cebadores incorporan uracilos para la posterior escisión por el USUARIO. Secuencias
- _scPv_USUARIO

```

>20100826_scFv_USER_VH1/2for
ggttaggUGAGGAGACRGTGACCCAGGGTG
>20100826_scFv_USER_VH4/5for
ggttaggUGAGGAGACGGTGACCCAGGGTT
>20100826_scFv_USER_VH3for
ggttaggUGAAGAGACGGTGACCCATTGT
>20100826_scFv_USER_VH6for
ggttaggUGAGGAGACGGTGACCCGTGGTCC

>20100826_scFv_USER_VL1back
ggggatecuCAGTCTGTSBTGACGCAGCCGCC
>20100826_scFv_USER_VL3back
ggggatecuTCCTATGWGCTGACWCAGCCAC
>20100826_scFv_USER_VL38back
ggggatecuTCCTATGAGCTGAYRCAGCVACC
>20100826_scFv_USER_VL4back
ggggatecuCAGCCTGTGCTGACTCARYC
>20100826_scFv_USER_VL7/Bback
ggggatecuCAGDCTGTGGTGACYCAGGAGCC
>20100826_scFv_USER_VL9back
ggggatecuCAGCCWKGCTGACTCAGCCMCC
>20100826_scFv_USER_VL11back
ggggatecuTCCTCTGAGCTGASTCAGGASCC
>20100826_scFv_USER_VL13back
ggggatecuCAGTCTGYICTGAYTCAGCCT
>20100826_scFv_USER_VL15back
ggggatecuAATTTTATGCTGACTCAGCCCC

>20100826_scFv_USER_VK1back
ggggatecuGACATCCRGDTGACCCAGTCTCC
>20100826_scFv_USER_VK2backts
ggggatecuGAAATTGTRWTGACRCAGTCTCC
>20100826_scFv_USER_VK9back
ggggatecuGATATTGTGHTGACBCAGWCTCC
>20100826_scFv_USER_VK12back
ggggatecuGAAACGACACTCACGCAGTCTC

>20100826_scFv_linker_upper_PEDS_11_405_LLA4_term_T
AGCGGAGCCACAGTGGTTCTGCTCCGAATAC?AGTTCTGCTCCGGGATCT

>20100826_scFv_linker_lower_PEDS_11_405_LLA4_term_T
GGACGAGACTAGTATTCCGAGCAGAACCCTGTGGCTCGCGCTGTTAGGt
    
```

- 10 ▪ _USUARIO

```

>20100827_USER_VH1/2for
agtctagUGAGGAGACRGTGACCAGGGTG
>20100827_USER_VH4/5for
agtctagUGAGGAGACGGTGACCAGGGTT
>20100827_USER_VH3for
agtctagUGAAGAGACGGTGACCATTGT
>20100827_USER_VH6for
agtctagUGAGGAGACGGTGACCGTGGTCC

>20100827_USER_VL1back
actagacuCAGTCTGTSBTGACGCAGCCGCC
>20100827_USER_VL3back
actagacuTCCTATGWGCTGACWCAGCCAC
>20100827_USER_VL38back
actagacuTCCTATGAGCTGAYRCAGCYACC
>20100827_USER_VL4back
actagacuCAGCCTGTGCTGACTCARYC
>20100827_USER_VL7/8back
actagacuCAGDCTGTGGTGACYCAGGAGCC
>20100827_USER_VL9back
actagacuCAGCCWGXGCTGACTCAGCCMCC
>20100827_USER_VL11back
actagacuTCCTTGAGCTGASTCAGGASCC
>20100827_USER_VL13back
actagacuCAGTCTGYCTGAYTCAGCCT
>20100827_USER_VL15back
actagacuAATTTTATGCTGACTCAGCCCC

>20100827_USER_VK1back
actagacuGACATCCRBDTGACCCAGTCTCC
>20100827_USER_VK2backts
actagacuGAAATTGTRWTGACRCAGTCTCC
>20100827_USER_VK9back
actagacuGATATTGTGMTGACBCAGWCTCC
>20100827_USER_VK12back
actagacuGAAACGACACTCAGCAGTCTC

```

Información de pedido

- Las porciones del enlazador scFv se ordenan en tubos individuales,
- 5 ▪ El resto de los cebadores se ordenan en una placa como se indica a continuación:



▪ NOTA: para ordenar desoxiuridina, el código IDT es: /ideoxyU/

```

20100826_scFv_USER_VH1/2for gttagg/ideoxyU/GAGGAGACRGTGACCAGGGTG
20100826_scFv_USER_VH4/5for gttagg/ideoxyU/GAGGAGACGGTGACCAGGGTT
20100826_scFv_USER_VH3for gttagg/ideoxyU/GAAGAGACGGTGACCATTGT
20100826_scFv_USER_VH6for gttagg/ideoxyU/GAGGAGACGGTGACCGTGGTCC
20100826_scFv_USER_VL1back gggatc/ideoxyU/CAGTCTGTSBTGACGCAGCCGCC
20100826_scFv_USER_VL3back gggatc/ideoxyU/TCCTATGWGCTGACWCAGCCAC
20100826_scFv_USER_VL38back gggatc/ideoxyU/TCCTATGAGCTGAYRCAGCYACC
20100826_scFv_USER_VL4back gggatc/ideoxyU/CAGCCTGTGCTGACTCARYC
20100826_scFv_USER_VL7/8back gggatc/ideoxyU/CAGDCTGTGGTGACYCAGGAGCC
20100826_scFv_USER_VL9back gggatc/ideoxyU/CAGCCWKGCTGACTCAGCCMCC
20100826_scFv_USER_VL11back gggatc/ideoxyU/TCCTCTGAGCTGASTCAGGASCC
20100826_scFv_USER_VL13back gggatc/ideoxyU/CAGTCTGYCTGAYTCAGCCT
20100826_scFv_USER_VL15back gggatc/ideoxyU/AATTTTATGCTGACTCAGCCCC
20100826_scFv_USER_VK1back gggatc/ideoxyU/GACATCCRGDTGACCCAGTCTCC
20100826_scFv_USER_VK2backts gggatc/ideoxyU/GAAATTGTRWTGACRCAGTCTCC
20100826_scFv_USER_VK9back gggatc/ideoxyU/GATATTGTGMTGACBCAGWCTCC
20100826_scFv_USER_VK12back gggatc/ideoxyU/GAAACGACACTCAGCAGTCTC

20100827_USER_VH1/2for agtctag/ideoxyU/GAGGAGACRGTGACCAGGGTG
20100827_USER_VH4/5for agtctag/ideoxyU/GAGGAGACGGTGACCAGGGTT
20100827_USER_VH3for agtctag/ideoxyU/GAAGAGACGGTGACCATTGT
20100827_USER_VH6for agtctag/ideoxyU/GAGGAGACGGTGACCGTGGTCC
20100827_USER_VL1back actagac/ideoxyU/CAGTCTGTSBTGACGCAGCCGCC
20100827_USER_VL3back actagac/ideoxyU/TCCTATGWGCTGACWCAGCCAC
20100827_USER_VL38back actagac/ideoxyU/TCCTATGAGCTGAYRCAGCYACC
20100827_USER_VL4back actagac/ideoxyU/CAGCCTGTGCTGACTCARYC
20100827_USER_VL7/8back actagac/ideoxyU/CAGDCTGTGGTGACYCAGGAGCC
20100827_USER_VL9back actagac/ideoxyU/CAGCCWKGCTGACTCAGCCMCC
20100827_USER_VL11back actagac/ideoxyU/TCCTCTGAGCTGASTCAGGASCC
20100827_USER_VL13back actagac/ideoxyU/CAGTCTGYCTGAYTCAGCCT
20100827_USER_VL15back actagac/ideoxyU/AATTTTATGCTGACTCAGCCCC
20100827_USER_VK1back actagac/ideoxyU/GACATCCRGDTGACCCAGTCTCC
20100827_USER_VK2backts actagac/ideoxyU/GAAATTGTRWTGACRCAGTCTCC
20100827_USER_VK9back actagac/ideoxyU/GATATTGTGMTGACBCAGWCTCC
20100827_USER_VK12back actagac/ideoxyU/GAAACGACACTCAGCAGTCTC
    
```

• La placa está aquí:

Medios:	placa.xls
---------	-----------

Información de mezcla

- 5
 - Las secuencias del enlazador scFv se ordenan a LabReady como 100 µM en pH 8 TE
 - Los cebadores en la placa están a 200 µM, por lo que deben mezclarse y diluirse.
 - Hacer una mezcla de 2.5 µM en cada cebador. Dado que son 200 µM, debemos dividir por 80.
 - Por 80 µL en total, poner 1 µL de cada cebador.
 - Poner los cebadores K y L juntos.
- 10
 - Hacer primero tubos con Tris-HCl:

Tubo	Tris-HCl inicial por 80 µL
A	76
B	67
E	76
F	67

_SOE_PEDS_11_405_LLA4

Información

- Cebadores basados en diseño de
- El enlazador es:

```
>PDS_11_405_LLA4
CCTAACAGCGCGAGCCACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCC
```

▪ La superposición es:

```
VL                               ACAGTGGTTC TGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATTGTGTTGACGCAGTCT
VH CTGGTGCCAGTGGCAGAGGAGTGGATTGTCGCGCTCGGTGTCACCAAGACGAGGCTTATGATC
```

Diana del cebador	Número de cebadores
IGHV (VH/izquierda pesada)	45
IGHJ (VH/derecha pesada)	8
IGK/LV (VL/izquierda liviana)	68
IGK/LJ (VL/derecha liviana)	11

Secuencias

```

>20100505_IGHV_left_1
GAGGTGCAGCTGGTGGAGTCCG
>20100505_IGHV_left_2
GAGGTGCAGCTGGTGGAGTCTGG
>20100505_IGHV_left_3
CAGGTCACCTTGAGGGACTCTGGTCC
>20100505_IGHV_left_4
CAGGTCAGCTGTTGCAGCCTGG
>20100505_IGHV_left_5
CAGGTGCAGTACAGCAGTGGGG
>20100505_IGHV_left_6
CAGGTGCAGCTGGTGCATCTGG
>20100505_IGHV_left_7
CAGGTCACCTTGAAGGAGTCTGGTCC
>20100505_IGHV_left_8
CAGGTCCAGCTGGTACAGTCTGGG
>20100505_IGHV_left_9
CAGGACCAGTTGGTGCAGTCTGGG
>20100505_IGHV_left_10
CAGGTGCAGCTGGTGGAGTCTGG
>20100505_IGHV_left_11
CAGATGCAGCTGGTGCAGTCTGG
>20100505_IGHV_left_12
CAATGCAGCTGGTGCAGTCTGGG
>20100505_IGHV_left_13
GAAGTGCAGCTGGTGGAGTCTGGG
>20100505_IGHV_left_14
CAGGTGCAGCTGGTGCAGTCTG
>20100505_IGHV_left_15
GAGGATCAGCTGGTGGAGTCTGGG
>20100505_IGHV_left_16
GAGGTCCAGCTGGTACAGTCTGGG
>20100505_IGHV_left_17
CAGCTGCAGCTGCAGGAGTCC
>20100505_IGHV_left_18
GAGGTGCAGCTGCTGCAGTCTGG
>20100505_IGHV_left_19
CAGGTGCAGCTGGTGCAGTCTG
>20100505_IGHV_left_20
CAGGTACAGCTGCAGCAGTCAAGT
>20100505_IGHV_left_21
CAGATGCAGCTGGTGCAGTCTGGG
>20100505_IGHV_left_22
GAGGTGCATCTGGTGGAGTCTGGG
>20100505_IGHV_left_23
GAGGTGCAGCTGGTGGAGTCTGG
>20100505_IGHV_left_24
GAGGTGCAGCTGGTGGAGTCTGG
>20100505_IGHV_left_25
CAGGTCCAACCTGGTGTAGTCTGGAGC
>20100505_IGHV_left_26
GAGGTGCAGCTGGTGCAGTCTG
>20100505_IGHV_left_27
GAGGTGCAGCTGGTGGAGTCTGG
>20100505_IGHV_left_28
GAGGTTCAGCTGGTGCAGTCTGGG
>20100505_IGHV_left_29
CAGGTGCAGCTGGTGCAGTCTG
>20100505_IGHV_left_30
GAAGTGCAGCTGGTGCAGTCTGG
>20100505_IGHV_left_31
GAGGTGGAGCTGATAGAGTCCATAGA
>20100505_IGHV_left_32
ACAGTGCAGCTGGTGGAGTCTGG
>20100505_IGHV_left_33
GAGGTGCAGCTGGAGGAGTCTGG
>20100505_IGHV_left_34
GAGGTACAGCTGGTGCAGTCTGAAGA
>20100505_IGHV_left_35
CAGGTGCAGCTGCAGGAGTCC
>20100505_IGHV_left_36
GAGGTGCAGCTGTTGCAGTCTGGG
>20100505_IGHV_left_37
CAGGTGCAGCTGGGGCAGTC
>20100505_IGHV_left_38
CAGCTGCAGCTGCAGGAGTCC
>20100505_IGHV_left_39
CAGGTTCAGCTGGTGCAGTCTGGA

```

>20100505_IGHV_left_40
 GAGGTGCAGCTGGTAGAGTCTGGG
 >20100505_IGHV_left_41
 CAGGTCCAGCTTGTGCAGTCTGGG
 >20100505_IGHV_left_42
 GAGGTGCAGCTGTGCAGTCTGC
 >20100505_IGHV_left_43
 GAGGTACAACCTGGTGGAGTCTGGGG
 >20100505_IGHV_left_44
 CAGATCACCTTGAAGGAGTCTGGTCC
 >20100505_IGHV_left_45
 CAGGTACAGCTGATGCAGTCTGGGG
 >20100505_IGHJ_right_1_SOE_PEDS_11_405_LLA4
 CTAGTATTCCGGAGCAGAACCCTGTGGCTCGCGCTGTTAGGTGAGGAGACGGTGACCGTGGTC
 >20100505_IGHJ_right_2_SOE_PEDS_11_405_LLA4
 CTAGTATTCCGGAGCAGAACCCTGTGGCTCGCGCTGTTAGGTGAGGAGACAGTGACCAGGGTGCC
 >20100505_IGHJ_right_3_SOE_PEDS_11_405_LLA4
 CTAGTATTCCGGAGCAGAACCCTGTGGCTCGCGCTGTTAGGTGATGTGGCTGCGGTCTCAGGG
 >20100505_IGHJ_right_4_SOE_PEDS_11_405_LLA4
 CTAGTATTCCGGAGCAGAACCCTGTGGCTCGCGCTGTTAGGTGAGGAGACGGTGACCAGGGTTCC
 >20100505_IGHJ_right_5_SOE_PEDS_11_405_LLA4
 CTAGTATTCCGGAGCAGAACCCTGTGGCTCGCGCTGTTAGGGACCCCTCAGAAGCCAGACCACC
 >20100505_IGHJ_right_6_SOE_PEDS_11_405_LLA4
 CTAGTATTCCGGAGCAGAACCCTGTGGCTCGCGCTGTTAGGTGAGGAGACGGTGACCAGGGTGC
 >20100505_IGHJ_right_7_SOE_PEDS_11_405_LLA4
 CTAGTATTCCGGAGCAGAACCCTGTGGCTCGCGCTGTTAGGGACGGTCCCAGGGAGACGGTGT
 >20100505_IGHJ_right_8_SOE_PEDS_11_405_LLA4
 CTAGTATTCCGGAGCAGAACCCTGTGGCTCGCGCTGTTAGGTGAAGAGACGGTGACCATTGTCCTCC
 >20100505_IGKV_left_1_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_2_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_3_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_4_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAACGACACTCACGCAGTCTCCAGC
 >20100505_IGKV_left_5_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAACGACACTCACGCAGTCTCCAGC
 >20100505_IGKV_left_6_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_7_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_8_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_9_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_10_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_11_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_12_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_13_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_14_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_15_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_16_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_17_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_18_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_19_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_20_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_21_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_22_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_23_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_24_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_25_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_26_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_27_SOE_PEDS_11_405_LLA4
 ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATGTGTGACGCAGTCTCCAGC
 >20100505_IGKV_left_28_SOE_PEDS_11_405_LLA4

ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGACATCCAGTTGACCCAGTCTCCATC
>20100505_IGKV_left_29_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGATATTTGTGATGACCCAGACTCCACT
>20100505_IGKV_left_30_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGTCATCTGGATGACCCAGCTCCATC
>20100505_IGKV_left_31_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGACATCCAGATGACCCAGCTCCATC
>20100505_IGKV_left_32_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATTTGTGTTGACACAGTCTCCAGG
>20100505_IGKJ_right_1
TTTGATCTCCACCTTGGTCCCTCCCG
>20100505_IGKJ_right_2
TTTGATCTCCAGCTTGGTCCCTGG
>20100505_IGKJ_right_3
TTTGATATCCACTTGGTCCAGGGC
>20100505_IGKJ_right_4
TTTGATTCCACCTTGGTCCCTGGC
>20100505_IGKJ_right_5
TTTAATCTCCAGTCGTGCCCTGGC
>20100505_IGLV_left_1_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCAGTCTGCCCTGATTCAGCTCCCC
>20100505_IGLV_left_2_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCTGCCTGTGCTGACTCAGCCCC
>20100505_IGLV_left_3_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCAGCCTGTGCTGACTCAGCCAC
>20100505_IGLV_left_4_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCTCTCTGAGCTGACTCAGGACCTCCG
>20100505_IGLV_left_5_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAATCTGCCCTGACTCAGCTCCCT
>20100505_IGLV_left_6_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCTCTATGAGCTGACACAGCCACCC
>20100505_IGLV_left_7_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCGGCCGTGCTGACTCAGC
>20100505_IGLV_left_8_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCAGCCTGTGCTGACTCAATCATCTCT
>20100505_IGLV_left_9_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCTCTGAGCTGAGTCAGGAGCCT
>20100505_IGLV_left_10_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAAGCTGCCCTGACTCAGCTCC
>20100505_IGLV_left_11_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAAGCTGTGTTGACGCAGCCGC
>20100505_IGLV_left_12_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCAGCCAGCCCTGACTCAGCCA
>20100505_IGLV_left_13_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAAGCTGTGTTGACGCAGCCGC
>20100505_IGLV_left_14_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAAGCTGTGCTGACTCAGCCACCC
>20100505_IGLV_left_15_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCTCTATGAGCTGACTCAGCCACTCTC
>20100505_IGLV_left_16_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAAGCTGTGCTGACTCAGCCAAC
>20100505_IGLV_left_17_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCTCTATGAGCTGACACAGCCACCC
>20100505_IGLV_left_18_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCTCTATGAGCTGACACAGCCATCTCTC
>20100505_IGLV_left_19_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAAGCTGTGCTGACTCAGCCAAC
>20100505_IGLV_left_20_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCTCTATGAGCTGACACAGCTACCCCTC
>20100505_IGLV_left_21_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAAGCTGTGCTGACTCAGCCATCT
>20100505_IGLV_left_22_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCTCTGCGGCCAACTCAGGTGC
>20100505_IGLV_left_23_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCTCTATGAGCTGACTCAGCCACCC
>20100505_IGLV_left_24_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAAGCTGTGTTGACCCAGGAGCC
>20100505_IGLV_left_25_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAAGCTGTGCTGACTCAGCCCTGC
>20100505_IGLV_left_26_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAATTTTATGCTGACTCAGCCCCACTC
>20100505_IGLV_left_27_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCTCTATGAGCTGATGCAGCCACCC
>20100505_IGLV_left_28_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAAGCTGTGTTGACTCAGGAGCC
>20100505_IGLV_left_29_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAAGCTGTGCTGACTCAATCGCCC
>20100505_IGLV_left_30_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAAGCTGTGTTGACTCAGGAGCCC
>20100505_IGLV_left_31_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCTCTATGAGCTGACTCAGCCACACTC
>20100505_IGLV_left_32_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAAGCTGTGCTGACTCAGCCG

ES 2 730 951 T3

```

>20100505_IGLV_left_33_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCTCCTATGTGCTGACTCAGCCACCC
>20100505_IGLV_left_34_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCAGTCTGCCCTGACTCAGCCTCG
>20100505_IGLV_left_35_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCAGTCTGTTCTGACTCAGCCTCGCT
>20100505_IGLV_left_36_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCAGTCTGTGCTGACGCAGCCG
>20100505_IGLJ_right_1
GAGGACGGTCACCTGGTGCCA
>20100505_IGLJ_right_2
TAGGACGGTCAGCTGGTCCCTCC
>20100505_IGLJ_right_3
GAGGACGGTCAGCTGGGTGCC
>20100505_IGLJ_right_4
TAAAATGATCAGCTGGGTCCCTCCAC
>20100505_IGLJ_right_5
TAGGACGGTGACCTGGTCCAGT
>20100505_IGLJ_right_6
TAGGACGGTCAGCTGGTCCCC

```

Información de la placa

- Ordenada a IDT:

Medios	I_SOE_PEDS_11_405_LLA4.xls
--------	----------------------------

Información de mezcla

- 5
- Hacer una mezcla de 2.5 μM en cada cebador. Como son 200 μM , debemos dividir por 80.
 - Por el total de 80 μL , poner 1 μL de cada cebador.
 - Hacer primero tubos con Tris-HCl:

Tubo	Regulador inicial por 80 μL
A	18
B	17
C	72
D	4
E	4
F	4
G	69
H	

- Luego, utilizando un canal múltiple, pipetee 1 μL por cada 80 columnas en los tubos.
- 10
- Luego combine los tubos A+B, C, D+E+F, G

20100505

Información

- Ver diseño

ES 2 730 951 T3

- La T_m fue diseñada para 62°C.

Diana de los cebadores	Número de los cebadores
IGHV (VH/izquierda pesada)	45
IGHJ (VH/derecha pesada)	8
IGK/LV (VL/izquierda liviana)	68
IGK/LJ (VL/derecha liviana)	11

Secuencias

```

>20100505_IGHV_left_1
GAGGTGCAGCTGGTGGAGTCCG
>20100505_IGHV_left_2
GAGGTGCAGCTGGTGGAGTCTGG
>20100505_IGHV_left_3
CAGGTCCACCTTGAGGGAGTCTGGTCC
>20100505_IGHV_left_4
CAGGTTACAGCTGTTGCAGCCTGG
>20100505_IGHV_left_5
CAGGTGCAGCTACAGCAGTGGGG
>20100505_IGHV_left_6
CAGGTGCAGCTGGTGAATCTGG
>20100505_IGHV_left_7
CAGGTCCACCTTGAGGGAGTCTGGTCC
>20100505_IGHV_left_8
CAGGTCCAGCTGGTACAGTCTGGG
>20100505_IGHV_left_9
CAGGACCAGTTGGTGCAGTCTGGG
>20100505_IGHV_left_10
CAGGTGCAGCTGGTGGAGTCTGG
>20100505_IGHV_left_11
CAGATGCAGCTGGTGCAGTCTGG
>20100505_IGHV_left_12
CAAATGCAGCTGGTGCAGTCTGGG
>20100505_IGHV_left_13
GAAGTGCAGCTGGTGGAGTCTGGG
>20100505_IGHV_left_14
CAGGTGCAGCTGGTGCAGTCTG
>20100505_IGHV_left_15
GAGGATCAGCTGGTGGAGTCTGGG
>20100505_IGHV_left_16
GAGGTCCAGCTGGTACAGTCTGGG
>20100505_IGHV_left_17
CAGCTGCAGCTGCAGGACTCC
>20100505_IGHV_left_18
GAGGTGCAGCTGGTGGAGTCTGG
>20100505_IGHV_left_19
CAGGTGCAGCTGGTGCAGTCTG
>20100505_IGHV_left_20
CAGGTACAGCTGCAGCAGTCAAGT
>20100505_IGHV_left_21
GAGATGCAGCTGGTGGAGTCTGGG
>20100505_IGHV_left_22
GAGGTGCATCTGGTGGAGTCTGGG
>20100505_IGHV_left_23
GAGGTGCAGCTGGTGGAGTCTGG
>20100505_IGHV_left_24
GAGGTGCAGCTGGTGGAGTCTGG

```

```

>20100505_IGHV_left_25
CAGGTCCAACCTGGTGTAGTCTGGAGC
>20100505_IGHV_left_26
GAGGTGCAGCTGGTGCAGTCTG
>20100505_IGHV_left_27
GAGGTGCAGCTGGTGGAGTCTCG
>20100505_IGHV_left_28
GAGGTTACAGCTGGTGCAGTCTGGG
>20100505_IGHV_left_29
CAGGTGCAGCTGGTGCAGTCTG
>20100505_IGHV_left_30
GAAGTGCAGCTGGTGCAGTCTGG
>20100505_IGHV_left_31
GAGGTGGAGCTGATAGAGTCCATAGA
>20100505_IGHV_left_32
ACAGTGCAGCTGGTGCAGTCTGG
>20100505_IGHV_left_33
GAGGTGCAGCTGGAGGAGTCTGG
>20100505_IGHV_left_34
GAGGTACAGCTGGTGGAGTCTGAAGA
>20100505_IGHV_left_35
CAGGTGCAGCTGCAGGAGTCCG
>20100505_IGHV_left_36
GAGGTGCAGCTGTTGGAGTCTGGG
>20100505_IGHV_left_37
CAGGTGCAGCTGGGGCAGTC
>20100505_IGHV_left_38
CAGCTGCAGCTGCAGGAGTCCG
>20100505_IGHV_left_39
CAGGTTACAGCTGGTGCAGTCTGGA
>20100505_IGHV_left_40
GAGGTGCAGCTGGTAGAGTCTGGG
>20100505_IGHV_left_41
CAGGTCCAGCTTGTGCAGTCTGGG
>20100505_IGHV_left_42
GAGGTGCAGCTCTTGCAGTCTGC
>20100505_IGHV_left_43
GAGGTACAACCTGGTGGAGTCTGGGGG
>20100505_IGHV_left_44
CAGATCACCTTGAAGGAGTCTGGTCC
>20100505_IGHV_left_45
CAGGTACAGCTGATGCAGTCTGGGG
.
.
>20100505_IGHJ_right_1
TGAGGAGACGGTGACCGTGCTC
>20100505_IGHJ_right_2
TGAGGAGACAGTGACCCAGGGTGCC
>20100505_IGHJ_right_3
TGATGTGGCTGGGCTCTCAGGG
>20100505_IGHJ_right_4
TGAGGAGACGGTGACCCAGGGTCC
>20100505_IGHJ_right_5
GACCCCTCAGAAGCCAGACCACC
>20100505_IGHJ_right_6
TGAGGAGACGGTGACCCAGGGTGC
>20100505_IGHJ_right_7
CACGTTCCAGGGAGACGGCTCT
>20100505_IGHJ_right_8
TGAAGAGACGGTGACCATGTCCCTT
.
.
>20100505_IGKV_left_1
GAAATGTGTTGACGCAGTCTCCAGC
>20100505_IGKV_left_2
GAAATGTGTTGACGCAGTCTCCAGC
>20100505_IGKV_left_3
SCCATCCAGTTGACCCAGTCTCCA
>20100505_IGKV_left_4
GACATCCAGATGACTCAGCCTCCATC
>20100505_IGKV_left_5
GAAACGACACTCAGGCAGTCTCCAGC
>20100505_IGKV_left_6
GACATCCAGATGACCCAGTCTCCATC
>20100505_IGKV_left_7
GACATCCAGTTGACCCAGTCTCCATC
>20100505_IGKV_left_8
GACATGTGATGACCCAGACTCCACT
>20100505_IGKV_left_9
GACATCGTGTGACCCAGTCTCCAGA
>20100505_IGKV_left_10
SCCATCCGGATGACCCAGTCTCC
>20100505_IGKV_left_11
CATGTTGTGATCACACAGTCTCCAGC
>20100505_IGKV_left_12

```

```

GATATTGTGATGACCCAGCATCTGCT
>20100505_IGKV_left_13
GAAATAGTGATGACCGAGTCTCCAGC
>20100505_IGKV_left_14
GATATTGTGATGACCCAGACTCCACC
>20100505_IGKV_left_15
GACATCCAGATGATCCAGTCTCCATC
>20100505_IGKV_left_16
GAAATTGTGTTGACACAGTCTCCAGC
>20100505_IGKV_left_17
GATATTGTGATGACTCAGTCTCCACT
>20100505_IGKV_left_18
GCCATCCGGATGACCCAGTCTCC
>20100505_IGKV_left_19
AACATCCAGATGACCCAGTCTCCATC
>20100505_IGKV_left_20
GAAATTGTAATGACACAGTCTCCAGC
>20100505_IGKV_left_21
GAAATTGTAATGACACAGTCTCCACC
>20100505_IGKV_left_22
GATGTTGTGATGACTCAGTCTCCACT
>20100505_IGKV_left_23
GACATTGTGCTGACCCAGTCTCCAGC
>20100505_IGKV_left_24
GACATCCAGATGACCCAGTCTCCCTC
>20100505_IGKV_left_25
GAAATTGTGCTGACTCAGTCTCCAGA
>20100505_IGKV_left_26
GAGATTGTGATGACCCAGACTCCACT
>20100505_IGKV_left_27
GCCATCCAGATGACCCAGTCTCCATC
>20100505_IGKV_left_28
GACATCCAGTTGACCCAGTCTCCATC
>20100505_IGKV_left_29
GATATTGTGATGACCCAGACTCCACT
>20100505_IGKV_left_30
GTCATCTGGATGACCCAGTCTCCATC
>20100505_IGKV_left_31
GACATCCAGATGACCCAGCTCCATC
>20100505_IGKV_left_32
GAAATTGTGTTGACACAGTCTCCAGC
.
>20100505_IGLV_left_1
CAGTCTGCCCTGATTCAGCCTCCC
>20100505_IGLV_left_2
CTGCCCTGTGCTGACTCAGCCCC
>20100505_IGLV_left_3
CAGCCTGTGCTGACTCAGCCAC
>20100505_IGLV_left_4
TCTTCTGAGCTGACTCAGGACCTGTC
>20100505_IGLV_left_5
CAATCTGCCCTGACTCAGCCTCCT
>20100505_IGLV_left_6
TCCTATGAGCTGACACAGCCACCC
>20100505_IGLV_left_7
CGGCCCGTGCTGACTCAGC
>20100505_IGLV_left_8
CAGCCTGTGCTGACTCAATCATCCTC
>20100505_IGLV_left_9
TCCTCTGAGCTGAGTCAGGAGCCT
>20100505_IGLV_left_10
CAGTCTGCCCTGACTCAGCCTCC
>20100505_IGLV_left_11
CAGTCTGTGTTGACGCAGCCGC
>20100505_IGLV_left_12
CAGGCAGGGCTGACTCAGCCA
>20100505_IGLV_left_13
CAGTCTGTGTTGACGCAGCCGC
>20100505_IGLV_left_14
CAGTCTGTGCTGACTCAGCCACCC
>20100505_IGLV_left_15
TCCTATGAGCTGACTCAGCCACTCTC
>20100505_IGLV_left_16
CAGCCTGTGCTGACTCAGCCAAC
>20100505_IGLV_left_17
TCCTATGAGCTGACACAGCCACCC
>20100505_IGLV_left_18
TCCTATGAGCTGACACAGCCATCCTC
>20100505_IGLV_left_19
CAGCCTGTGCTGACTCAGCCAAC
>20100505_IGLV_left_20
TCCTATGAGCTGACACAGTACCCTC
>20100505_IGLV_left_21

```

```

CAGCCTGTGCTGACTCAGCCATCT
>20100505_IGLV_left_22
TCCTCTGGGCAACTCAGGTGC
>20100505_IGLV_left_23
TCCTATGAGCTGACTCAGCCACCCT
>20100505_IGLV_left_24
CAGACTGTGGTGACCCAGGAGCC
>20100505_IGLV_left_25
CAGTCTGCCCTGACTCAGCCTGC
>20100505_IGLV_left_26
AATTTTATGCTGACTCAGCCCCACTC
>20100505_IGLV_left_27
TCCTATGAGCTGATGCAGCCACCC
>20100505_IGLV_left_28
CAGGCTGTGGTGACTCAGGAGCC
>20100505_IGLV_left_29
CAGCTTGTGCTGACTCAATCGCCC
>20100505_IGLV_left_30
CAGACTGTGGTGACTCAGGAGCCC
>20100505_IGLV_left_31
TCCTATGAGCTGACTCAGCCACACTC
>20100505_IGLV_left_32
CAGGCTGTGCTGACTCAGCCG
>20100505_IGLV_left_33
TCCTATGTGCTGACTCAGCCACCC
>20100505_IGLV_left_34
CAGTCTGCCCTGACTCAGCCTCG
>20100505_IGLV_left_35
CAGTCTGTTCTGACTCAGCCTCGCT
>20100505_IGLV_left_36
CAGTCTGTCTGACCGAGCCG

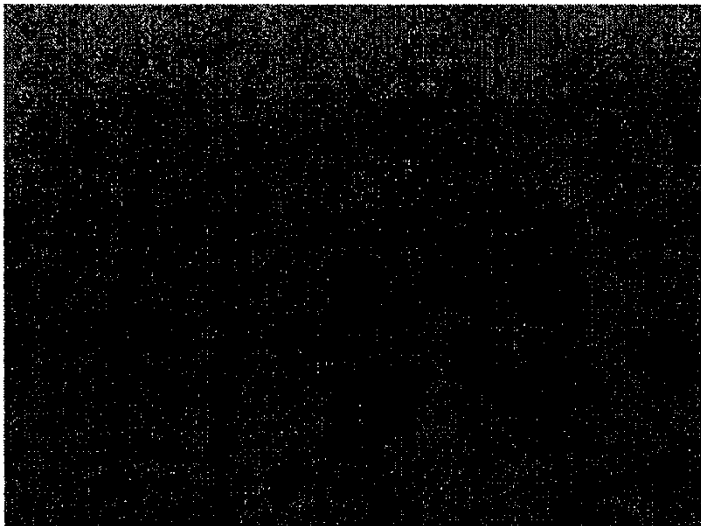
>20100505_IGKJ_right_1
TTTGATCTCCACCTTGGTCCCTCCGC
>20100505_IGKJ_right_2
TTTGATCTCCAGCTTGGTCCCTGG
>20100505_IGKJ_right_3
TTTGATATCCACTTGGTCCCGGGC
>20100505_IGKJ_right_4
TTTGATTCCACCTTGGTCCCTGGC
>20100505_IGKJ_right_5
TTTAATCTCCAGTCGTGCCCTGGC

>20100505_IGLJ_right_1
GAGGACGGTCACCTTGGTGCCA
>20100505_IGLJ_right_2
TAGGACGGTCAGCTTGGTCCCTCC
>20100505_IGLJ_right_3
GAGGACGGTCAGCTGGGTGCC
>20100505_IGLJ_right_4
TAAATGATCAGCTGGGTCCCTCCAC
>20100505_IGLJ_right_5
TAGGACGGTGACCTTGGTCCAGT
>20100505_IGLJ_right_6
TAGGACGGTCAGTCGGTCCCC

```

Información de la placa

- Ordenado a IDT:



Secuencias

```

-----
>20100329_VH4back|1
CAGGTGCAGCTGCAGGAGTCCG
>20100329_VH4back|2
CAGGTGCAGCTGCAGGAGTCCG
>20100329_VH5back|1
CAGGTACAGCTGCAGCAGTCA
>20100329_VH6back|1
CAGGTGCAGCTACAGCAGTGGG
>20100329_VH10back|1
GAGGTGCAGCTGGTGGAGACC
>20100329_VH10back|2
GAGGTGCAGCTGGTGGAGACT
>20100329_VH10back|3
GAGGTGCAGCTGGTGGAGTCC
>20100329_VH10back|4
GAGGTGCAGCTGGTGGAGTCT
>20100329_VH10back|5
GAGGTGCAGCTGTTGGAGACC
>20100329_VH10back|6
GAGGTGCAGCTGTTGGAGACT
>20100329_VH10back|7
GAGGTGCAGCTGTTGGAGTCC
>20100329_VH10back|8
GAGGTGCAGCTGTTGGAGTCT
>20100329_VH12back|1
CAGGTCCAGCTGGTACAGTCTGG
>20100329_VH12back|2
CAGGTCCAGCTGGTGCAGTCTGG
>20100329_VH12back|3
CAGGTCCAGCTGTGTACAGTCTGG
>20100329_VH12back|4
CAGGTCCAGCTGTGTACAGTCTGG
>20100329_VH14back|1
CAGATCACCTTGAAGGAGTCTG
>20100329_VH14back|2
CAGGTACACCTTGAAGGAGTCTG
>20100329_VH22back|1
CAGGTGCAGCTGGTGCAATCTGG
>20100329_VH22back|2
CAGGTGCAGCTGGTGCAGTCTGG
>20100329_VH22back|3
CAGGTGCAGCTGGTGGAAATCTGG
>20100329_VH22back|4
CAGGTGCAGCTGGTGGAGTCTGG
>20100329_VH1/2for|1_SOE_PEDS_11_405_LLA4
CTAGTATTCGGAGCAGAACCCTGTGGCTCGCGCTGTTAGGTGAGGAGACAGTACCAGGGTG
>20100329_VH1/2for|2_SOE_PEDS_11_405_LLA4
CTAGTATTCGGAGCAGAACCCTGTGGCTCGCGCTGTTAGGTGAGGAGACAGTACCAGGGTG
>20100329_VH4/5for|1_SOE_PEDS_11_405_LLA4
CTAGTATTCGGAGCAGAACCCTGTGGCTCGCGCTGTTAGGTGAGGAGACAGTACCAGGGTG
>20100329_VH3for|1_SOE_PEDS_11_405_LLA4
CTAGTATTCGGAGCAGAACCCTGTGGCTCGCGCTGTTAGGTGAGGAGACAGTACCATTGT
>20100329_VH6for|1_SOE_PEDS_11_405_LLA4
CTAGTATTCGGAGCAGAACCCTGTGGCTCGCGCTGTTAGGTGAGGAGACAGTACCCTGGTCC
>20100329_VL1back|1_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAGTCTGTCTGACGCAGCCGCC
>20100329_VL1back|2_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAGTCTGTCTGACGCAGCCGCC
>20100329_VL1back|3_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAGTCTGTCTGACGCAGCCGCC
>20100329_VL1back|4_SOE_PEDS_11_405_LLA4

```



```

>20100329_VL13back|6_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAGTCTGTCTGATTCAGCCT
>20100329_VL13back|7_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAGTCTGTCTGATTCAGCCT
>20100329_VL13back|8_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAGTCTGTCTGATTCAGCCT
>20100329_VL15back|1_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCCAATTTTATGCTGATTCAGCCCC
>20100329_VK1back|1_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGACATCCAGATGACCCAGTCTCC
>20100329_VK1back|2_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGACATCCAGGTGACCCAGTCTCC
>20100329_VK1back|3_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGACATCCAGTTGACCCAGTCTCC
>20100329_VK1back|4_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGACATCCGGATGACCCAGTCTCC
>20100329_VK1back|5_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGACATCCGGGTGACCCAGTCTCC
>20100329_VK1back|6_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGACATCCGGTTGACCCAGTCTCC
>20100329_VK2backts|1_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATTTGATGACACAGTCTCC
>20100329_VK2backts|2_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATTTGATGACACAGTCTCC
>20100329_VK2backts|3_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATTTGATGACACAGTCTCC
>20100329_VK2backts|4_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATTTGATGACACAGTCTCC
>20100329_VK2backts|5_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATTTGATGACACAGTCTCC
>20100329_VK2backts|6_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATTTGATGACACAGTCTCC
>20100329_VK2backts|7_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATTTGATGACACAGTCTCC
>20100329_VK2backts|8_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAATTTGATGACACAGTCTCC
>20100329_VK9back|1_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGATATTGTGATGACCCAGTCTCC
>20100329_VK9back|2_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGATATTGTGATGACCCAGTCTCC
>20100329_VK9back|3_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGATATTGTGATGACCCAGTCTCC
>20100329_VK9back|4_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGATATTGTGATGACCCAGTCTCC
>20100329_VK9back|5_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGATATTGTGATGACCCAGTCTCC
>20100329_VK9back|6_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGATATTGTGATGACCCAGTCTCC
>20100329_VK9back|7_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGATATTGTGATGACCCAGTCTCC
>20100329_VK9back|8_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGATATTGTGATGACCCAGTCTCC
>20100329_VK9back|9_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGATATTGTGATGACCCAGTCTCC
>20100329_VK9back|10_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGATATTGTGATGACCCAGTCTCC
>20100329_VK9back|11_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGATATTGTGATGACCCAGTCTCC
>20100329_VK9back|12_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGATATTGTGATGACCCAGTCTCC
>20100329_VK12back|1_SOE_PEDS_11_405_LLA4
ACAGTGGTTCTGCTCCGAATACTAGTTCTGCTCCGGGATCCGAAACGACACTCACGCAGTCTC
>20100329_VL1/2for|1
TAGGACGGTCACCTTGGTCC
>20100329_VL1/2for|2
TAGGACGGTCAGCTTGGTCC
>20100329_VL1/2for|3
TAGGACGGTGACCTTGGTCC
>20100329_VL1/2for|4
TAGGACGGTGAGCTTGGTCC
>20100329_VL7for|1
GAGGACGGTCAGCTGGGTGC
>20100329_VK1for|1
TTTGATTCACCTTGGTCC
>20100329_VK2/4for|1
TTTGATTCACCTTGGTCC
>20100329_VK2/4for|2
TTTGATTCACCTTGGTCC
>20100329_VK3for|1
TTTGATTCACCTTGGTCC
>20100329_VK5for|1
TTTGATTCACCTTGGTCC

```

Información de la placa

- Los cebadores se ordenaron de IDT en IDTE pH 8 a una concentración de 200 µM.

ES 2 730 951 T3

Medios	_SOE_PEDS_11_405_LLA4.xls
--------	---------------------------

Información de mezcla

- Hacer una mezcla de 2.5 μ M en cada cebador. Como son 200 μ M, debemos dividir por 80.
- Por el total de 80 μ L, poner 1 μ L de cada cebador.
- Hacer primero tubos con Tris-HCl:

Tubo	Regulador inicial por 80 μ L
A	58
B	75
C	1
D	1
E	1
F	1
G	35
H	35

5

- Luego, utilizando un canal múltiple, pipetee 1 μ L por cada 80 columnas en los tubos.
- Luego combine los tubos A,B, C+D+E+F, G+H

_SOE_PNAS_85_5879_G4S3

•

10

- secuencia del enlazador scFv:

```

>(G_4 S)_3 scFv linker from Huston et al, PNAS 85: 5879
GGTGGCGGTGGCTCGGGCGGTGGTGGGTGGGTGGCGCGGATCT
    
```

- Los conteos son:

Diana del cebador	Número de los cebadores
IGHV (VH/izquierda pesada)	7
IGHJ (VH/derecha pesada)	4
IGK/LV (VL/izquierda liviana)	13
IGK/LJ (VL/derecha liviana)	6

- Los cebadores están debajo, "back" son los cebadores 5' (a la izquierda) y "for" son los cebadores 3' (a la derecha).

```

>20100329_VH4back
CAGGTGCAGCTGCAGGAGTCSG
>20100329_VH5back
CAGGTACAGCTGCAGCAGTCA
>20100329_VH6back
CAGGTGCAGCTACAGCAGTGGG
>20100329_VH10back
GAGGTGCAGCTGRTGGAGWCY
>20100329_VH12back
CAGGTCCAGCTKGTFCAGTCTGG
>20100329_VH14back
CAGRTCACCTGAAGGAGTCTG
>20100329_VH22back
CAGGTGCAGCTGGTGSARTCTGG

>20100329_VH1/2for_SOE
CCCGACCCACCACCGCCGAGCCACCGCCACCTGAGGAGACRGTGACCAGGGTG
>20100329_VH4/5for_SOE
CCCGACCCACCACCGCCGAGCCACCGCCACCTGAGGAGACGGTGACCAGGGTT
>20100329_VH3for_SOE
CCCGACCCACCACCGCCGAGCCACCGCCACCTGAAGAGACGGTGACCATTGT
>20100329_VH6for_SOE
CCCGACCCACCACCGCCGAGCCACCGCCACCTGAGGAGACGGTGACCCTGGTCC

>20100329_VL1back_SOE
CGGGCGGTGGTGGGTCGGTGGCGGGGATCTCAGTCTGTSBTGACGCAGCCGCC
>20100329_VL3back_SOE
CGGGCGGTGGTGGGTCGGTGGCGGGGATCTTCCTATGWCTGACWCAGCCAC
>20100329_VL38back_SOE
CGGGCGGTGGTGGGTCGGTGGCGGGGATCTTCCTATGAGCTGAYRCAGCYACC
>20100329_VL4back_SOE
CGGGCGGTGGTGGGTCGGTGGCGGGGATCTCAGCCTGTGCTGACTCARYC
>20100329_VL7/8back_SOE
CGGGCGGTGGTGGGTCGGTGGCGGGGATCTCAGDCTGTGGTGACYCAGGAGCC
>20100329_VL9back_SOE
CGGGCGGTGGTGGGTCGGTGGCGGGGATCTCAGCCWKGCTGACTCAGCCMCC
>20100329_VL11back_SOE
CGGGCGGTGGTGGGTCGGTGGCGGGGATCTTCCTGTGAGCTGASTCAGGASCC
>20100329_VL13back_SOE
CGGGCGGTGGTGGGTCGGTGGCGGGGATCTCAGTCTGYCTGAYTCAGCCT
>20100329_VL15back_SOE
CGGGCGGTGGTGGGTCGGTGGCGGGGATCTAATTTTATGCTGACTCAGCCCC

>20100329_VL1/2for
TAGGACGGTSASCTTGGTCC
>20100329_VL7for
GAGGACGGTCAGCTGGGTGC

>20100329_VK1back_SOE
CGGGCGGTGGTGGGTCGGTGGCGGGGATCTGACATCCREDTGACCCAGTCTCC
>20100329_VK2backts_SOE
CGGGCGGTGGTGGGTCGGTGGCGGGGATCTGAAATGTRWTGACRCAGTCTCC
>20100329_VK9back_SOE
CGGGCGGTGGTGGGTCGGTGGCGGGGATCTGATATTGTMGTGACBCAGWCTCC
>20100329_VK12back_SOE
CGGGCGGTGGTGGGTCGGTGGCGGGGATCTGAAACGACACTCACCAGTCTC

>20100329_VK1for
TTTGATTTCCACCTTGGTCC
>20100329_VK2/4for
TTTGATCTCCASCTTGGTCC
>20100329_VK3for
TTTGATATCCACTTTGGTCC
>20100329_VK5for
TTTAAATCTCCAGTCTGGTCC

```

- La placa IDT está dispuesta de la siguiente manera:

Posición del pozo	Nombre	Secuencia
A1	20100329_VH4back	CAGGTGCAGCTGCAGGAGTCSG
A2	20100329_VH5back	CAGGTACAGCTGCAGCAGTCA
A3	20100329_VH6back	CAGGTGCAGCTACAGCAGTGGG
A4	2010329_VH10back	GAGGTGCAGCTGKTGGAGWCY
A5	20100329_VH12back	CAGGTCCAGCTKGRRCAGTCTGG
A6	20100329_VH14back	CAGRTCACCTTGAAGGAGTCTG
A7	20100329_VH22back	CAGGTGCAGCTGGTGSARTCTGG
B1	20100329_VH1/2for_SOE	CCCGACCCACCACCCGCCGAGCCACCACCACCACCTGAGGAGACRGTGACCAGGGTG
B2	20100329_VH4/Sfor_SOE	CCCGACCCACCACCCGCCGAGCCACCACCACCACCTGAGGAGACRGTGACCAGGGTT
B3	20100329_VH3for_SOE	CCCGACCCACCACCCGCCGAGCCACCACCACCACCTGAAGAGACGGTGACCATTGT
B4	20100329_VH6for_SOE	CCCGACCCACCACCCGCCGAGCCACCACCACCACCTGAGGAGACRGTGACCAGGGTCC
C1	20100329_VL1back_SOE	CGGGCGGTGGTGGGTCGGGTGGCGGGGATCTCAGTCTGTSBTGACCGAGCCGCC
C2	20100329_VL3back_SOE	CGGGCGGTGGTGGGTCGGGTGGCGGGGATCTTCTATGWGCTGACWCAGCCAC
C3	20100329_VL38back_SOE	CGGGCGGTGGTGGGTCGGGTGGCGGGGATCTTCTATGAGCTGAYRCAGCYACC
C4	20100329_VL4back_SOE	CGGGCGGTGGTGGGTCGGGTGGCGGGGATCTCAGCCTGTGCTGACTCARYC
C5	20100329_VL7/8back_SOE	CGGGCGGTGGTGGGTCGGGTGGCGGGGATCTCAGDCTGGTGACYCAGGAGCC
C6	20100329_VL9back_SOE	CGGGCGGTGGTGGGTCGGGTGGCGGGGATCTCAGCCWKGCTGACTCAGCCMCC
C7	20100329_VL11back_SOE	CGGGCGGTGGTGGGTCGGGTGGCGGGGATCTTCTCTGAGCTGASTCAGGASCC
C8	20100329_VL13back_SOE	CGGGCGGTGGTGGGTCGGGTGGCGGGGATCTCAGTCTGYCTGAYTCAGCCT
C9	20100329_VL15back_SOE	CGGGCGGTGGTGGGTCGGGTGGCGGGGATCTAAATTTATGCTGACTCAGCCCC
D1	20100329_VL1/2for	TAGGACGGTSASCTTGGTCC
D2	20100329_VL7for	GAGGACGGTCAGCTGGGTGC
E1	20100329_VK1back_SOE	CGGGCGGTGGTGGGTCGGGTGGCGGGGATCTGACATCCRBDTGACCCAGTCTCC
E2	20100329_VK2backts_SOE	CGGGCGGTGGTGGGTCGGGTGGCGGGGATCTGAAATGTRWTGACRCAGTCTCC
E3	2010329_VK9back_SOE	CGGGCGGTGGTGGGTCGGGTGGCGGGGATCTGATATTGTMGTACBCAGWCCTCC
E4	20100329_VK_12back_SOE	CGGGCGGTGGTGGGTCGGGTGGCGGGGATCTGAAACGACACTCAGCCAGTCTC

Posición del pozo	Nombre	Secuencia
F1	20100329_VK1for	TTTGATTTCCACCTTGGTCC
F2	2010329_VK2/4for	TTTGATCTCCASCCTTGGTCC
F3	20100329_VK3for	TTTGATATCCACTTTGGTCC
F4	20100329_VK5for	TTTAATCTCCAGTCGTGTC

ES 2 730 951 T3

- También ordené "cebadores de enlace" que son solo complementarios al enlace.

```

5'                                     CGGGCGGTGGTGGGTGGGTGGGCGGATCTCAGTCTGTSBTGACGCAGCCGCC
3' TTGGGACCAGTGGCAGAGGAGTCCACCGCCACCGAGCCCGCCACCCAGCCCGCC

>VHfor_SOE_linker
CCCGACCCACCACCGCCCG
>VLback_SOE_linker
CGGGCGGTGGTGGGTGGG

>20100329_VH1/2for
TGAGGAGACRGTGACCAGGGTG
>20100329_VH4/5for
TGAGGAGACGGTGACCAGGGTT
>20100329_VH3for
TGAAGAGACGGTGACCATTGT
>20100329_VH6for
TGAGGAGACGGTGACCGTGGTCC

>20100329_VL1back
CAGTCTGTSBTGACGCAGCCGCC
>20100329_VL3back
TCCTATGWGCTGACWCAGCCAC
>20100329_VL38back
TCCTATGAGCTGAYRCAGCYACC
>20100329_VL4back
CAGCCTGTGCTGACTCARYC
>20100329_VL7/8back
CAGDCTGTGGTGACYCAGGAGCC
>20100329_VL9back
CAGCCWKGCTGACTCAGCCMCC
>20100329_VL11back
TCCTCTGAGCTGASTCAGGASCC
>20100329_VL13back
CAGTCTGYCTGAYTCAGCCT
>20100329_VL15back
AATTTATGCTGACTCAGCCCC

>20100329_VK1back
GACATCCRGDTGACCCAGTCTCC
>20100329_VK2backts
GAAATTGTRWTGACRCAGTCTCC
>20100329_VK9back
GATATTGTGMTGACBCAGWCTCC
>20100329_VK12back
GAAACGACACTCAGCAGTCTC
    
```

- Estos se disponen en la placa de la siguiente manera:

Posición del pozo	Nombre	Secuencia
A1	20100329_VH1/2for	TGAGGAGACRGTGACCAGGGTG
A2	20100329_VH4/5for	TGAGGAGACGGTGACCAGGGTT
A3	20100329_VH3for	TGAAGAGACGGTGACCATTGT
A4	20100329_VH6for	TGAGGAGACGGTGACCGTGGTCC
C1	20100329_VL1back	CAGTCTGTSBTGACGCAGCCGCC
C2	20100329_VL3back	TCCTATGWGCTGACWCAGCCAC
C3	20100329_VL38back	TCCTATGAGCTGAYRCAGCYACC
C4	20100329_VL4back	CAGCCTGTGCTGACTCARYC
C5	20100329_VL7/8back	CAGDCTGTGGTGACYCAGGAGCC

ES 2 730 951 T3

Posición del pozo	Nombre	Secuencia
C6	20100329_VL9back	CAGCCWKGCTGACTCAGCCMCC
C7	20100329_VL11back	TCCTCTGAGCTGASTCAGGASCC
C8	20100329_VL13back	CAGTCTGYYCTGAYTCAGCCT
C9	20100329_VL15back	AATTTTATGCTGACTCAGCCCC
E1	20100329_VK1back	GACATCCRGDTGACCCAGTCTCC
E2	20100329_VK2backts	GAAATTGTRWTGACRCAGTCTCC
E3	20100329_VK9back	GATATTGTGMTGACBCAGWCTCC
E4	20100329_VK12back	GAAACGACACTCACGCAGTCTC
H6	residuo 1	ACGTATGCATGCTAG
H7	residuo 2	ACGTATGCAGTCTAG
H8	residuo 3	ACCCGTGCATGCTAG
H9	residuo 4	GGTTATGCATGCTAG
H10	residuo 5	ACGTATGCAGACTAG
H11	residuo 6	ACGTATGCATTTTAG
H12	residuo 7	ACGTATGGGTGCTAG

Apéndice B

Generación SOE_PCR y ScFV en celdas individuales.

SOE VDJ (emparejamiento de cadenas pesadas y livianas)

- 5 Use una mezcla extra para hacer 5x FV1 (180 ng/ul ... 1.67ulRNA+5.33ulH2O) para cadenas livianas y 5 para cadenas pesadas

1. Transcripción reversa; Proceda de la siguiente manera para RT-PCR

Se hicieron 5 tubos para cadena pesada y 5 tubos para cadena liviana (la mezcla que se muestra aquí fue porque esto se hizo al mismo tiempo que el VIH VDJ)

- | | | |
|-----|--|------------------|
| 1.1 | Preparar una reacción de RT como sigue; | 2x 1 <u>tubo</u> |
| | • H2O(DEPC) | 0 µl(a 12 µl) |
| | • cebador específico del gen 2µM (pesado o ligero) | 5 ul (5 pmol) |
| | • Total ARN | 7 ul (150 ng) |

- 10 Calentar a 95°C durante 1 minuto seguido de 65°C durante 5 min luego en hielo durante 1 min

ES 2 730 951 T3

Centrifugar y agregar lo siguiente:	<u>1 tubo</u>	<u>30 tubos</u>
• 5x regulador de primera hebra	4 µl	120
• Mezcla dNTP 10 mM	1 µl	30
• DTT 0.1M	1 µl	30
• Inhibidor de la RNasa-enzimática	1 µl	30
• Superíndice III	1 µl	30
Incubar a 55°C durante 60 min		(8 µl/tubos)

1.2 Inactivar la enzima calentando a 70°C durante 15 min

1.3 Eliminar el híbrido ARN/ADN agregando 1 µl de RNasaH de E.coli (Enzy)

Incubar a 37°C durante 20 min, luego hielo.

5 2. Preparar la reacción de PCR de la siguiente manera:

Usar la mitad para PCR mantener la otra mitad como respaldo ... cantidad adicional realizada porque se realiza en paralelo con el proyecto VIH_VDJ

2.1 Preparar una reacción de prueba de PCR como sigue

	1 muestra x2 por x 15 muestras (45t) (cadena liviana de 12t)		
• dH ₂ O	21 µl (a 50 µl)	945(1084.46)	252(259.2)
• ADNc	5 µl	-	-
• 5x Regulador HF	10 µl	450	120
• dNTP(10 mM) (25 mM)	1 µl (0.4)	45 (18)	12(4.8)
• cebador alto (IGHV nuevo-corto) (4 µM)	6.25 µl (25 pmol)	281.25(225)	75
• cebador bajo (IGHC nuevo-corto)(4 µM)	6.25 µl (25 pmol)	281.25(224)	75
• Phusion	0.5 µl	22.5	6

10 Agregar 90 pcr de mezcla a 10 ul de cADN, divida la PCR rx en 2 tubos de 50, para cada muestra. Era bajo en cebador, así que dio números rojos

2.2 Ciclo térmico de la siguiente manera:

1- 98°C durante 1 min

2- 98°C durante 10 segundos

15 3- 62°C durante 20 segundos

4- 72°C durante 20 segundos, pasar a etapa 2, 23x

5- 72°C durante 5 min

6- 4°C pausa

ES 2 730 951 T3

2.3 Reunir cada reacción 2 en 1

2.4 Agregar 2 µl de Exonucleasa (20 U/µl) a cada tubo e incubar a 37°C durante 20 min

2.5 AMPure XP 1.8:1. Resuspendido en 40 µl (menos si es posible)

Se realiza la purificación SPRI de la siguiente manera:

- 5 • Agregar la cantidad requerida de perlas AMPure XP (relación 1.8:1) a la muestra de ADN en el regulador EB.
- Vórtice para mezclar
- Incubar durante 5 minutos a temperatura ambiente.
- Imán (MPC) durante 5 minutos. Dejar el tubo de perlas en el MPC durante todos los pasos de lavado
- 10 • Se retira el sobrenadante (manténgalo en caso de fallo) y lavar las perlas dos veces con 500 µl de etanol al 70%, incubando durante 30 segundos cada vez.
- Se retira todo el sobrenadante, centrifugar rápidamente, se retira la última gota y dejar que las perlas AMPure se sequen al aire por completo (2 min).
- 15 • Se retira el tubo del MPC, agregar 40 µl de Tris-HCl 10 mM, EDTA 0.1 mM, pH 8.0 (o Regulador EB de Qiagen), y agitar durante 30 segundos para volver a suspender las perlas, dejar reposar durante 3 minutos el imán durante 2 minutos y transferir el sobrenadante a un nuevo tubo.

3. Gel de diagnóstico (comprobar la eficacia de la PCR)

Correr 2 µl en un 2% de Egel-X durante 12 minutos

VH = 410 pb

VL = 360 pb

- 20 Agrupar los 5 tubos. Y Se vuelve a concertar con AMPure hasta 60 µl para que pueda cargarse en 2 carriles en la preparación de Pippin.

3.1.1 PCR de Phusion de los cebadores SOE LLA4 a partir de ADNc

Plan

Reacciones

- 25 • Hacer una mezcla maestra de los siguientes componentes.

		Mezcla principal 4.1x
dH ₂ O	11	45.1
5x Regulador HF (det. libre)	4	16.4
dNTP 10 mM	0.4	1.64
ADNc	4	16.4
Phusion Hot Start	0.2	0.82
Total	19.6	

• Alícuotas de 19.6 µL en 4 tubos de PCR. Agregar 0.2 µL de cada cebador apropiado, como sigue (8 mezclas de cebadores diferentes en total):

1. Cebadores VH

ES 2 730 951 T3

2. Cebadores LLA4 SOE VH
3. Cebadores VL
4. Cebadores LLA4 SOE VL

Ciclos

1. 98°C, 30 s
2. 98°C, 10 s
- 5 3. 65°C, 30 s
4. 72°C, 30 s
5. Ir a 2, 29x
6. 72°C, 5 min
7. 4°C, perpetuo

10 Resultados

- El gel está abajo. 2% de E-gel EX ejecutado durante 10 min.

Línea	Muestra	Cantidad
M	Escalera de 50 pb NEB	1 µL en 20
1		
2	VH PCR	10 µL en 20
3	VH LLA4 SOE PCR	10 µL en 20
4		
5	VL PCR	10 µL en 20
6	VL LLA4 SOE PCR	10 µL en 20
7		
8	Muestra de Billy	
9		
10		

3.1.2 Gradiente PCR de LLA4 cebadores con Phusion a partir de ADNc

Reacciones

- 15 ▪ Mezclar lo siguiente. Hay dos mezclas maestras.

ES 2 730 951 T3

		Patrón VH (4.1x)	Patrón VL (4.1x)
dH ₂ O	11	45.1	45.1
5x regulador HF (det. libre)	4	16.4	16.4
dNTP 10 mM	0.4	1.64	1.64
Cebador 1	0.2	0.82	0.82
Cebador 2	0.2	0.82	0.82
ADNc	4	16.4	16.4
Phusion Hot Start	0.2	0.82	0.82
Total	20		

- Alícuotas 20 µL de cada mezcla en 4 tubos de una tira de PCR, utilizando los tubos 1, 3, 5, 7.
- Las tiras se colocarán en las posiciones 3 a 10 del bloque de gradiente.
- Las temperaturas de gradiente son:

1. 64.9°C
2. 66.9°C
3. 70.0°C
4. 72.5°C

5 Ciclo

1. 98°C, 30 s
2. 98°C, 10 s
3. 3. Graduación 64°C-74°C (ajuste de la máquina), 30 s. 72°C, 30 s
5. Ir a 2, 24x
- 10 6. 4°C, perpetuo

Resultados

- El gel está debajo: 2% de E-gel EX ejecutado durante 10 minutos

Línea	Muestra	Cantidad
M	Escalera de 50 pb NEB	1 µL en 20
1		
2	VH (gradiente) 64.9°C	10 µL en 20

ES 2 730 951 T3

Línea	Muestra	Cantidad
3	VH (gradiente) 66.9°C	10 µL en 20
4	VH (gradiente) 70.0°C	10 µL en 20
5	VH (gradiente) 72.5°C	10 µL en 20
6		
7	VL (gradiente) 64.9°C	10 µL en 20
8	VL (gradiente) 66.9°C	10 µL en 20
9	VL (gradiente) 70.0°C	10 µL en 20
10	VL (gradiente) 72.5°C	10 µL en 20

3.1.3 Kapa 2G Robust PCR con LLA4 a partir de ADNc

Plan

- Probaré la enzima Kapa 2G Robust para ver si funciona mejor usando el mismo conjunto de cebadores que Phusion
- 5
- También creo que Kapa no tiene actividad exo, y creo que la banda de dímero de cebador puede requerir exo para formarse de manera sólida, por lo que quizás esto ayude a evitarlo.

Reacciones

- Mezclar lo siguiente en tubos de PCR separados

	Control negativo VH	VH	Control negativo VL	VL
dH ₂ O	19	15	19	15
Kapa 2G regulador A	5	5	5	5
dNTP 10 mM	0.5	0.5	0.5	0.5
Cebador 1	0.2	0.2	0.2	0.2
Cebador 2	0.2	0.2	0.2	0.2
ADNc	0	4	0	4
Kapa 2G Robust	0.1	0.1	0.1	0.1
Total	25	25	25	25

10 Ciclo

1. 95°C, 1 min

2. 95°C, 20 s

ES 2 730 951 T3

- 3. 65°C, 20 s
- 4. 72°C, 30 s
- 5. Ir a 2, 24x
- 6. 4°C, perpetuo

5 Resultados

- El gel está debajo (e-gel ex 2%)

Línea	Muestra	Cantidad
M	Escalera de 50 pb NEB	1 µL en 20
1		
2	VH control negativo	10 µL en 20
3	VH	10 µL en 20
4		
5	VL control negativo	10 µL en 20
6	VL	10 µL en 20

3.1.4 Correspondencia de clones de cadena pesada/liviana

- Combinamos manualmente algunas posibles cadenas pesadas y livianas para nuestra síntesis. La información está en la hoja de cálculo.

3.1.5 Fusión por PCR utilizando diferentes números de ciclos.

Plan

- Ver si reducir la cantidad de ciclos más ayudará a los PCR

Reacciones

		Patrón VH (5.2x)	Patrón VL (5.2x)
dH ₂ O	11	57.2	57.2
regulador HF (regulador det. libre)	4	20.8	20.8
dNTP 10 mM	0.4	2.08	2.08
Cebador 1	0.2	1.04	1.04
Cebador 2	0.2	1.04	1.04
ADNc	4	20.8	20.8
Phusion Hot Start	0.2	1.04	1.04

ES 2 730 951 T3

		Patrón VH (5.2x)	Patrón VL (5.2x)
<ul style="list-style-type: none"> ▪ NOTA: Me quedé sin una de las muestras de ADNc, así que tuve que usar dos muestras de ADNc diferentes: ▪ VH ▪ VL ▪ Distribuir 20 µL en 5 tubos de PCR para cada uno de VH y VL ▪ Los tubos de PCR se cortan y se cargan en un termociclador 			

Ciclo

- Prueba de diferentes cantidades de ciclos: 9, 13, 17, 21, 25 ciclos.

5 ▪ Después de 9 ciclos, sacamos uno de los tubos para VH y VL, y luego sacamos un tubo cada 4 ciclos hasta que el último tubo haga 25 ciclos.

El programa es:

1. 98°C, 30 s
2. 98°C, 10 s
3. 72°C, 30 s
- 10 4. Ir a 2, 8x
5. 20°C, perpetuo (retirar tubo)
6. 98°C, 10 s
7. 72°C, 30 s
8. Ir a 6, 3x
- 15 9. Ir a 5, 3x
- 10.4°C, perpetuo

Resultados

- El gel está debajo (e-gel ex 2% 10 min)

Línea	Muestra	Cantidad
M	Escalera de 50 pb NEB	1 µL en 20
1	VH 9 ciclos	10 µL en 20
2	VH 13 ciclos	10 µL en 20
3	VH 17 ciclos	10 µL en 20
4	VH 21 ciclos	10 µL en 20
5	VH 25 ciclos	10 µL en 20
6	VL 9 ciclos	10 µL en 20

ES 2 730 951 T3

Línea	Muestra	Cantidad
7	VL 13 ciclos	10 µL en 20
8	VL 17 ciclos	10 µL en 20
9	VL 21 ciclos	10 µL en 20
10	VL 25 ciclos	10 µL en 20

3.1.6 Síntesis de ADNc utilizando SuperScript III

▪ Usé el ARN de PBMC etiquetado con ARN que me dio François. Terminé todo el tubo (por lo que usé un poco menos de ARN que en la síntesis de ADNc anterior)

Reacciones

- 5 ▪ Mezclar lo siguiente:

		(correcto) 16.5x mezcla maestra	Mezcla real
total ARN (PBMC) (70 ng/µL)	5	82.5	66
hexámeros aleatorios	1	16.5	16.5
dNTP 10 mM	1	16.5	16.5
DEPC H ₂ O	3	49.5	72

▪ NOTA: La mezcla real está demasiado diluida porque en lugar de agregar 16.5 µL de agua como debería haberlo hecho, en realidad se agregaron 22.5 por alguna extraña razón.

▪ Tomé alícuotas de 10 µL de esta mezcla en 16 tubos de PCR

- 10 ▪ Incubar a 65°C durante 5 min.
 ▪ Incubar a 4°C durante >1 min.
 ▪ Mientras tanto, haga la siguiente mezcla:

		16.5x patrón
10x RT regulador	2	33
MgCl ₂ 25 mM	4	66
DTT 0.1 M	2	33
Rnasa OUT (40 U/µL)	1	16.5
SuperScript III (200 U/µL)	1	16.5

▪ Alícuotas 10 µL en cada tubo

- 15 ▪ Incubar a 25°C, 10 min.

ES 2 730 951 T3

- Incubar a 50°C, 50 min.
- Incubar a 85°C, 5 min.
- Enfriar en hielo
- Combiné las 16 reacciones en un solo Eppy.

5 ▪ Añadí 16 µL de RNasa H e incubé a 37°C durante 20 minutos

3.1.7 Gradiente de concentración del cebador para la PCR de Phusion a partir de ADNc

Plan

- Una de las optimizaciones finales que quiero realizar antes de pasar al entrecruzamiento es verificar las concentraciones óptimas del cebador

10 ▪ Haré un gradiente de concentración de cebadores para verificar si hay productos secundarios y eficiencia.

Reacciones

- Hacer la siguiente mezcla maestra. (Puedo hacer VH y VL juntos porque los cebadores se agregarán por separado).

		16.5x maestra
dH ₂ O	0.2	3.3
Regulador HF (det. libre)	4	66
dNTP 10 mM	0.4	6.6
ADNc	4	66
Phusion Hot Start	0.2	3.3

- Alícuotas de 8.8 µL en cada uno de los 16 tubos de PCR (8 VH, 8 VL)

15 ▪ Los cebadores se agregarán de la siguiente manera:

Tubo	Cebador conc. (nM; cada cebador)	Volumen de mezcla cebador	dH ₂ O volumen
1	12.5	0.1	5.5
2	25	0.2	5.4
3	40	0.32	5.28
4	70	0.56	5.04
5	125	1	4.6
6	250	2	3.6
7	500	4	1.6
8	700	5.6	0

ES 2 730 951 T3

- Tenga en cuenta que las cantidades agregadas son para cada mezcla de cebador, de modo que cada tubo individual requiere 4 pasos de pipeteo (cebador derecha + agua, cebador izquierda + agua).

Ciclo

1. 98°C, 30 s
- 5 2. 98°C, 10 s
3. 72°C, 30 s
4. Ir a 2, 24x
5. 4°C, perpetuo

Resultados

- 10 ▪ Los geles están por debajo (E-gel EX 2%):

Línea	Muestra	Cantidad
M	Escalera de 50 pb NEB	1 µL en 20
1		
2	VH 12.5 nM	10 µL en 20
3	VH 25 nM	10 µL en 20
4	VH 40 nM	10 µL en 20
5	VH 70 nM	10 µL en 20
6	VH 125 nM	10 µL en 20
7	VH 250 nM	10 µL en 20
8	VH 500 nM	10 µL en 20
9	VH 700 nM	10 µL en 20
10		

Análisis

- 15 ▪ El VH PCR muestra un claro máximo de producto producido. Es un poco difícil ver productos secundarios (probablemente debido a la cantidad ligeramente reducida de ciclos), pero parece que la cantidad óptima es probablemente 40-70 nM de cada cebador (0.32-0.56 µL de mezcla de cebadores a 2.5 µM de cada cebador).

3.1.8 PCR para la purificación en gel de cadenas VH y VL

Plan

- Se realiza la PCR utilizando cebadores SOE y cebadores USER para extraer en gel el material

Reacciones

- 20 ▪ Preparar las siguientes reacciones:

ES 2 730 951 T3

Cebadores LLA4 (SOE)	VH	VL
dH ₂ O	11	11
Regulador HF (det. libre)	4	4
dNTP 10 mM	0.4	0.4
Cebador 1	0.2	0.2
Cebador 2	0.2	0.2
ADNc	4	4
Fusión	0.2	0.2

Cebadores de usuarios	VH	VL
dH ₂ O	14.4	14.4
Regulador A (5x)	5	5
dNTP 10 mM	0.5	0.5
Cebador 1	0.5	0.5
Cebador 2	0.5	0.5
Reacción de ligación	4	4
Kapa 2G Robust	0.1	0.1

Ciclo

▪ LLA4 SOE

1. 98°C, 30 s
2. 98°C, 10 s
3. 67°C, 30 s
4. 72°C, 30 s
5. Vaya a 2, 24x
6. 4°C, perpetuo

5 ▪ Usuario

ES 2 730 951 T3

1. 95°C, 1 min
2. 95°C, 10 s
3. 55°C, 30 s
4. 72°C, 15 s
5. Ir a 2, 4x
6. 95°C, 10 s
7. 60°C, 30 s
8. 72°C, 15 s
9. Ir a 6, 19x
10. 4°C, perpetuo

Extracción

- Las bandas de ~450 pb se cortan y purifican utilizando el kit Qiagen MinElute.
- El ADN se eluye utilizando 10 µL de regulador EB, y luego agrego 9 µL de Tris-HCl a cada tubo.

5 Resultados

- E-gel EX 2% 10 min

Línea	Muestra	Cantidad
M	Escalera de 50 pb NEB	1.25 µL en 25
1		
2	SOE VH	20 µL en 25
3		
4	SOE VL	20 µL en 25
5		
6	USUARIO VH	20 µL en 25
7		
8	USUARIO VL	20 µL en 25
9		
10		

ES 2 730 951 T3

3.1.9 Preparación de VH/VL enlazadas a partir de cadenas inmunes seleccionadas por tamaño a través de SOE-PCR o USUARIO/T4 ligasa

3.1.9.1 Montaje SOE-PCR

5 Reacciones • Mezclar lo siguiente.

- NOTA: NO AGREGAR LOS CEBADORES. Se añadirán después de los primeros 5 ciclos de PCR.
- NOTA: Los amplicones VH aquí son los amplicones cebados con SOE de LLA4

dH ₂ O	13
Regulador HF (det. libre)	4
dNTP 10 mM	0.4
Cebador 1	(0.2)
Cebador 2	(0.2)
VH amplicón	1
VL amplicón	1
Fusión	0.2

Ciclo

- 10
1. 98°C, 30 s
 2. 98°C, 10 s
 3. 67°C, 30 s
 4. 72°C, 30 s
 5. Ir a 2, 4x
- 15
6. 16°C, perpetuo (AGREGAR el cebador aquí)
 7. 98°C, 10 s
 8. 67°C, 30 s
 9. 72°C, 30 s
 10. Ir a 7, 24x
- 20
11. 4°C, perpetuo

3.1.9.2 USUARIO/T4 ligasa

Protocolo

- NOTA: Los amplicones aquí son los amplicones cebados 0827 para la digestión por el USUARIO (sin scFv)

Digestión USUARIO

- 25
- Combina lo siguiente:
 - Amplicón VH de 3 µL

- Amplicón VL de 3 µL
- 4 µL de dH₂O
- 0.5 µL de UDG (Enzymatics)
- 0.5 µL Endo VIII (Enzymatics)

5 ▪ Incubar a 37°C, 30 min.

Reacción de ligación

- A la reacción de digestión, agregar lo siguiente:

- 7 µL de dH₂O
- 1 µL de regulador de ligasa T4 10x

10 ▪ 1 µL de ligasa T4 (NEB)

- Incubar a 16°C, 30 min.
- Incubar a 65°C, 10 min.

Amplificación por PCR del producto enlazado.

- Hacer la siguiente mezcla de PCR:

dH ₂ O	13.4
Regulador A (5x)	5
dNTP 10 mM cada uno	0.5
Cebador 1	0.5
Cebador 2	0.5
Reacción de ligación	5
Kapa 2G Robust	0.1

15

- Ciclo de la siguiente manera:

1. 95°C, 1 min
2. 95°C, 10 s
3. 55°C, 30 s
4. 72°C, 15 s
5. Ir a 2, 4x
6. 95°C, 10 s
7. 60°C, 30 s
8. 72°C, 15 s

ES 2 730 951 T3

- 9. Ir a 6, 19x
- 10. 4°C, perpetuo

3.1.9.3 Resultados

- E-gel EX 2% 10 min

Línea	Muestra	Cantidad
M	Escalera de 50 pb NEB	1 µL en 20
1		
2	SOE-PCR	10 µL en 20
3		
4	USUARIO/ligación T4	10 µL en 20

5 3.1.10 PCR multipaso de VH y VL

- Intentar obtener scFvs usando reacciones de PCR en serie con diferentes conjuntos de cebadores

3.1.10.1 Control PCR con cebadores originales

- Mezcle lo siguiente (nota, cebadores separados para las cadenas Kappa y Lambda)

	VH	VL
dH ₂ O	11.15	10.9
Regulador HF (det. libre; 5x)	4	4
dNTP (10 mM ea)	0.4	0.4
Cebador 1 (4 µM ea)	0.125	0.125 x2 (cadenas K y L)
Cebador 2 (4 µM ea)	0.125	0.125 x2 (cadenas K y L)
ADNc	4	4
Phusion	0.2	0.2

- Ciclo:

- 1. 98°C, 1 min
- 2. 98°C, 10 s
- 3. 62°C, 20 s

ES 2 730 951 T3

4. 72°C, 20 s
5. 2, 24x
6. 4°C, perpetuo

3.1.10.2 PCR multipaso

PCR A

- Primera PCR usando los cebadores externos de PCR originales
- 5
- Mezclar lo siguiente. Las cadenas pesada y liviana se amplifican juntas.

	VH+VK+VL
dH ₂ O	10.65
Regulador HF (det. libre; 5x)	4
dNTP (10 mM ea)	0.4
Cebador (4 μM ea)	0.125 x6 (todo loci)
ADNc	4
Phusion	0.2

▪ Ciclo:

1. 98°C, 1 min
2. 98°C, 10 s
3. 62°C, 20 s
4. 72°C, 20 s
5. Ir a 2, 9x
6. 4°C, perpetuo

▪ Agregar 1 μL de Exo I e incube para:

- 10
- 37°C, 30 min.
 - 80°C, 20 min.

PCR B & C

▪ PCR anidada interna utilizando los cebadores originales. Intentando con las versiones no SOE y SOE solo para probar las reacciones de PCR.

- 15
- Mezclar lo siguiente

ES 2 730 951 T3

	PCR B No SOE	SOE (LLA4) PCR C
dH ₂ O	13.6	13.6
Regulador HF (det. libre; 5x)	4	4
dNTP (10 mM ea)	0.4	0.4
Cebador (2.5 μM ea)	0.2 x4 (VH+VL)	0.2 x4 (VH+VL)
PCR A	1	1
Phusion	0.2	0.2

▪ Ciclo:

1. 98°C, 30 s
2. 98°C, 10 s
3. 67°C, 30 s
4. 72°C, 30 s
5. Ir a 2, 9x
6. 4°C, perpetuo

▪ Agregar 1 μL de Exo I e incube para:

5. 37°C, 30 min
- 80°C, 20 min

PCR D

▪ Ahora estamos agregando las etiquetas SOE LLA4 a PCR B

▪ Mezclar lo siguiente

	0505 SOE (LLA4) PCR D
dH ₂ O	13.6
Regulador HF (det. libre; 5x)	4
dNTP (10 mM ea)	0.4
Cebador (2.5 μM ea)	0.2 x4 (VH+VL)
PCR B	1

ES 2 730 951 T3

	0505 SOE (LLA4) PCR D
Phusion	0.2

▪ Ciclo:

1. 98°C, 30 s
2. 98°C, 10 s
3. 67°C, 30 s
4. 72°C, 30 s
5. Ir a 2, 9x
6. 4°C, perpetuo

▪ Agregar 1 µL de Exo I e incube para:

- 5
- 37°C, 30 min.
 - 80°C, 20 min.

▪ NOTA: El resto se realizó otro día.

Ciclo de extensión de superposición ▪

Este es un ciclo sin cebador para obtener fragmentos scFv de SOE.

- 10
- Para las reacciones de PCR C y D, agregar lo siguiente:

dNTP (10 mM ea)	0.2
Phusion	0.4

▪ Ciclo:

1. 98°C, 30 s
2. 98°C, 10 s
3. 67°C, 30 s
4. 72°C, 30 s
5. Ir a 2, 9x
6. 4°C, perpetuo

PCR externa final

- 15
- Este es el PCR final para amplificar el scFv. Aquí hacemos 25 ciclos.

ES 2 730 951 T3

- A las reacciones cíclicas anteriores, ahora agregar lo siguiente. Nota, solo estamos agregando los "cebadores externos":

VH cebador izquierdo 2.5 μ M	0.2
VL cebador derecho 2.5 μ M	0.2

- Ciclo:

1. 98°C, 30 s
2. 98°C, 10 s
3. 67°C, 30 s
4. 72°C, 30 s
5. Ir a 2, 24x
6. 4°C, perpetuo

5

- Los resultados están en el gel a continuación (2% de EX-E-gel, 10 min)

Línea	Muestra	Cantidad
M	Escalera de 100 pb NEB	1 μ L en 20
1		
2	control de VH PCR	10 μ L en 20
3	control de VL PCR	10 μ L en 20
4		
5	PCR A	10 μ L en 20
6	PCR B	10 μ L en 20
7	PCR C (final)	10 μ L en 20
8	PCR D (final)	10 μ L en 20

4.

10 5. Canalización completa de ADNc-> scFv con ciclos reducidos

6. PCR a partir de ADNc con cebadores.

PCR A

ES 2 730 951 T3

▪ Mezclar lo siguiente. Los cebadores son los cebadores:

	VH	VL	VH+VL
dH ₂ O	11.15	10.9	10.65
Regulador HF (det. libre; 5x)	4	4	4
dNTP (10 mM ea)	0.4	0.4	0.4
Cebador (4 μM ea)	0.125 x2	0.125 x4 (K+L)	0.125 x6 (todo loci)
ADNc	4	4	4
Phusion	0.2	0.2	0.2

▪ Ciclo:

1. 98°C, 30 s
2. 98°C, 10 s
3. 60°C, 30 s
4. 72°C, 30 s
5. Ir a 2, 14x
6. 10°C, perpetuo

5 ▪ Agregar 1 μL de Exo I e incube para:

- 37°C, 30 min
- 80°C, 20 min

7. PCR de PCR A utilizando cebadores SOE attB PCR B

▪ Mezclar lo siguiente. Los cebadores no son los cebadores SOE-attB.

	VH	VL	VH+VL
dH ₂ O	14.5	14.5	14.1
Regulador HF (det. libre; 5x)	4	4	4
dNTP (10 mM ea)	0.4	0.4	0.4
Cebador (2.5 μM ea)	0.2 x2	0.2 x2	0.2 x4
PCR A	0.5	0.5	0.5

ES 2 730 951 T3

	VH	VL	VH+VL
Phusion	0.2	0.2	0.2

▪ Ciclo:

1. 98°C, 30 s
2. 98°C, 10 s
3. 65°C, 30 s
4. 72°C, 30 s
5. Ir a 2, 14x
6. 10°C, perpetuo

▪ Agregar 1 µL de Exo I e incube para:

- 5
- 37°C, 30 min.
 - 80°C, 20 min.

NOTA: El resto del experimento se terminará mañana.

8. PCR de PCR B utilizando cebadores universales attB

PCRC

- 10
- NOTA: Las reacciones individuales VH y VL ahora deben combinarse para obtener scFvs. Esta reacción combinada se etiquetará como "VH/VL sep".

▪ Mezclar lo siguiente:

	VH/VL sep.	VH+VL
dH ₂ O	12.4	12.9
Regulador HF (det. libre; 5x)	4	4
dNTP (10 mM ea)	0.4	0.4
PCR B	0.5 x2 (VH y VL)	0.5
Phusion	0.2	0.2
Total	18	18

▪ Agregar 1 µL de cada 10 µM de cebadores attB universales en el punto de pausa

- 15
- Ciclo:

1. 98°C, 30 s

ES 2 730 951 T3

2. 98°C, 10 s
3. 67°C, 30 s
4. 72°C, 30 s
5. Ir a 2, 9x
6. 10°C, perpetuo

Añadir cebador aquí

7. 98°C, 30 s
8. 98°C, 10 s
9. 63°C, 30 s
10. 72°C, 30 s
11. Ir a 8, 21x
12. 10°C, perpetuo

▪ Correr en E-gel EX 2% durante 10 min

Línea	Muestra	Cantidad
M	Escalera de 100 pb NEB	1 µL en 20
1		
2	VH/VL sep.	10 µL en 20
3		
4	VH+VL	10 µL en 20

Apéndice C

5 VDJome/Immune-seq/solo cadena liviana

Realizar la preparación de la biblioteca de cadenas livianas solo en las muestras de los días 1, 7 y 21.

1. ID de muestra

La muestra se extrajo de 9 mL de sangre de un solo individuo en varios puntos de tiempo y se procesó con el kit de extracción Leukolock de Ambion:

10 Luego, las muestras se concentraron utilizando el kit RNeasy de Qiagen, se eluyó en 2x 30 µl y se utilizó Nanodrop.

3. Cadena liviana, el trato real.

ES 2 730 951 T3

Muestra original usada			O segundo lote		
ID	para 500 ng	Agua a 7	para 500 ng	Agua a 7	
3	90.57 ng/ul 5.52	1.48	2.0	5	
5	84.74 ng/ul 5.9	1.1	2.16	4.84	
7	71.89 ng/ul 6.96	0	2.7	4.3	
FV1	180 ng/ul 2.78	4.22			

ID	MID	muestra de vacunación	de concentración (ng/μl)	260/280	Vol para 500 ng	H ₂ O para 9.5 μl
1	1	-n	84.28	2.13	8.93	0.57
2	2	0	63.15	2.12	7.92	1.58
3	3	24 hora	90.57	2.13	5.52	3.98
4	4	3 días	94.84	2.15	5.27	4.23
5	5	7 días	84.74	2.14	5.90	3.60
6	6	14 días	72.97	2.13	6.85	2.65
7	8	21 días	71.89	2.14	6.96	2.54
8	9	28 días	79.11	2.17	6.32	3.18
9	10	FV muestra de prueba	133.09	2.16	3.75	5.75

Luego, las muestras se concentraron utilizando el kit RNeasy de Qiagen, se eluyó en 2x 30 μl y Nanodrop

VDJ-ID	MID	muestra de vacunación	de concentración (ng/μl)	Bioanalizador-nano	RNA	RIN	Vol para 1000 ng	H ₂ O para 9.5 μl
1	1	-n	128.32	128		7.20	7.79	1.71
2	2	0	167.18	163		6.10	5.98	3.52
3	3	24 hora	251.42	287		7.20	3.98	5.52
4	4	3 días	263.68	200		6.60	3.79	5.71
5	5	7 días	231.84	235		8.90	4.31	5.19
6	6	14 días	193.86	165		7.70	5.16	4.34

ES 2 730 951 T3

VDJ-ID	MID	muestra de vacunación	de concentración (ng/μl)	Bioanalizador-nano	RNA	RIN	Vol para 1000 ng	H ₂ O para 9.5 μl
7	8	21 días	185.23	186		7.60	5.40	4.1
8	9	28 días	193.29	156		6.30	5.17	4.33

Tubo	ID	notas
1	viejo-3	
2	viejo-5	
3	viejo-7	Apenas nada, 1 ul tal vez
4	FV1	
5	rneasy-3	
6	rneasy-5	
7	rneasy-7	

3. Transcripción reversa; Proceda de la siguiente manera para RT-PCR

3.1	Preparar una reacción de RT como sigue;	2x <u>1 tubo</u>
	• H ₂ O(DEPC)	0 μl(a 12μl)
	• cebador específico del gen 2μM (IGHC K y L)	5 (5 pmol)
	• Total ARN	7 μl (500 ng)

5 Calentar a 95°C durante 1 minuto seguido de 65°C durante 5 minutos luego en hielo durante 1 minuto

Centrifugar y agregar lo siguiente:	<u>1 tubo</u>	<u>9 tubos</u>
• 5x regulador de primera hebra	4μl	36
• mezcla dNTP 10 mM	1 μl	9
• DTT 0.1M	1 μl	9
• Inhibidor de la ARNasa-enzima	1 μl	9
• Superíndice III	1 μl	9
Incubar a 55 ° C durante 60 min		(8 μl/tubos)

3.2 Inactivar la enzima calentando a 70°C durante 15 min

ES 2 730 951 T3

3.3 Elimine el híbrido de ARN/ADN agregando 1 µl de RNasaH de E.coli (Enzy)

Incubar a 37°C durante 20 min, luego hielo.

4. Preparar la reacción de PCR de la siguiente manera:

Use la mitad para PCR, mantenga la otra mitad como respaldo

5 4.1 Preparar una reacción de prueba de PCR como sigue

	1x	x2 por muestra x 7 muestras (16t)
• dH ₂ O	21 µl (a 50 µl)	336
• ADNc	5 µl	-
• 5x Regulador HF	10 µl	160
• dNTP(10 mM)	1 µl	16
• cebador alto (IGHV nuevo-corto) (4 µM)	6.25 µl (25 pmol)	100 (50 de cada h K y L)
• cebador bajo (IGHC nuevo-corto)(4µM)	6.25 µl (25 p mol)	100
• Phusion	0.5 µl	8

Agregar 90 pcr de mezcla a 10 ul de ADNc, luego divídalo en 2 tubos de 50 para cada muestra

4.2 Ciclo térmico de la siguiente manera:

- 1- 98°C durante 1 min
- 2- 98°C durante 10 segundos
- 3- 62°C durante 20 segundos
- 4- 72°C durante 20 segundos ir al paso 2, 23x
- 5- 72°C durante 5 min
- 6- 4°C pausa

Muestra de Feeze

10 5. Gel de diagnóstico (comprobar la eficacia de la PCR).

Correr 2 ul en un 2% de Egel-X durante 12 minutos (hizo 11 minutos)

5.1 Agregar 2 µl de Exonucleasal (20 U/µl) a cada tubo e incube a 37°C durante 45 minutos (hizo 20 minutos)

5.2 AMPure XP relación de purificación de 1.8:1. Resuspender en 40 ul (mantenga el volumen no vinculante como respaldo). Se guardan 2 ul de cada muestra para el diagnóstico de gel

15 Se realiza la purificación SPRI de la siguiente manera:

- Agregar la cantidad requerida de perlas AMPure XP (relación 1.8:1) a la muestra de ADN en el regulador EB.
- Vórtice para mezclar
- Incubar durante 5 minutos a temperatura ambiente.

- Imán (MPC) durante 5 minutos. Dejar el tubo de perlas en el MPC durante todos los pasos de lavado
- Se retira el sobrenadante (manténgalo en caso de fallo) y lavar las perlas dos veces con 500 µl de etanol al 70%, incubando durante 30 segundos cada vez.
- Se retira todo el sobrenadante, centrifugar rápidamente, se retira la última gota y dejar que las perlas AMPure se sequen al aire por completo (2 min).
- Se retira el tubo del MPC, agregar 40 µl de Tris-HCl 10 mM, EDTA 0.1 mM, pH 8.0 (o Regulador EB de Qiagen), y agitar durante 30 segundos para volver a suspender las perlas, dejar reposar durante 3 minutos el imán durante 2 minutos y transferir el sobrenadante a un nuevo tubo.

5.3 Gel de diagnóstico (solo para comprobar la eficacia de perlas de AMPure)

10 Correr 2 ul en un 2% de Egel-X durante 12 minutos

Primeros 4 carriles

Se ve bien

6. Extremos bloqueados

6.1 Se realiza la reacción de bloqueo utilizando el kit de reparación Enzymatics End de la siguiente manera:

	1t
• H2O	0 µl(a 50µl)
• ADN purificado	38 µl
• 10 × End regulador de reparación	5 ul
• mezcla dNTP 1 mM	5 µl
• Mezcla de enzimas	5 µl

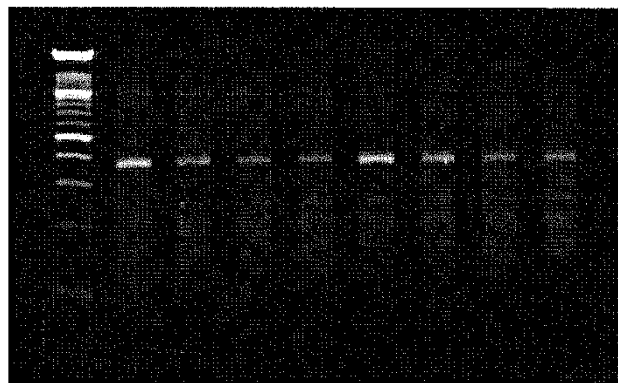
15

Incubar a una temperatura de 25°C durante 30 min.

El calor se inactiva a 75°C durante 20 min.

6.2 AMPure XP relación de purificación de 1.8:1. Resuspender en 37 ul (mantener el volumen sin enlace como copia de seguridad)

20 Ahorre 2 ul de cada muestra para diagnóstico de gel.



6.3 Gel de diagnóstico (verifique la eficiencia del bloqueo y perlas de AMPure)

Correr 2 ul en un 2% de Egel-X durante 12 minutos

ES 2 730 951 T3

Últimos 4 carriles

Se ve bien

7. Ligación del adaptador (25x de exceso)

7.1 Autofusión oligo 454 en el adaptador A y B juntos

- 5 Siga el protocolo 454 (10 µl de cada cebador 100 µM + 30 µl de (Tris 10 mM EDTA 0.1 mM a NaCl 50 mM) (cada adaptador está en 20 µM final), 95 durante 3 min, rampa a 15 perpetua a 0.1°C/seg.

Preparar el adaptador A MID 3-5-8 y el adaptador B solamente.

7.2 Ligación del adaptador 454-A y 454-B (454 usa una proporción de 15:1 de adaptador ADN)

	1t
• H2O	10 µl (to 100 µl)
• 2x regulador de ligasa rápida	50 µl
• VDJ DNA	35 µl (~1.25 pmol a 400 pb)
• 20 µM 454-A	1.5 µl (~30 pmol)
• 20 µM 454-B	1.5 µl (~30 pmol)
• T4 ADN ligasa (rápido)	2 µl

- 10 Incubar en PCR a 22°C durante 30 min.

7.3 AMPure XP relación de purificación de 1.8:1. Resuspender en 37 ul

7.4 Gel de diagnóstico (prueba)

Correr 2 ul en un 2% de Egel-X durante 12 minutos

Escalera de 100 pb

- 15 1- FV1 pre ligación

2-5 postligación, postpurificación

8. Reacción de relleno

*Si usa BST, tenga en cuenta la presencia de un relave en A en los extremos 3'

8.1 En un tubo de microcentrífuga, agregar los siguientes reactivos, en el orden indicado, y mezcle:

	1t	5t
• H2O	5.5 (a 50µl)	27.5
• ADN	35	-
• 10x regulador Thermopol	2.5µl	12.5
• 10x regulador Phi29	2.5	12.5
• 100x BSA (10 mg/ml)	0.5	2.5

ES 2 730 951 T3

	1t	5t
• dNTP 10mM Mezcla	1 µl (0.2mM)	5
• Bst Polimerasa (8 U/µl)	2 µl	10
• Phi29	1ul	5

Incubar a 30°C durante 10 min, luego a 60°C durante 15 min (termociclador) Inactivar por calor a 80°C durante 15 min

8.2 AMPure XP relación de purificación de 1.8:1. Resuspender en 30 ul

9. Extracción en gel de biblioteca ligada de cadena pesada

9.1 Capacidad de carga de prueba con muestra FV1 solo, carga 10 ul 5 ul y 2.5 ul (espacio entre cada carril)

5 Correr en un e-gel ex 2% , correr 15 min

9.2 Cargar otra muestra a través de 3 carriles (10 ul/carriles)

9.3 Usar el kit de extracción de gel Qiagen regulador (fundir banda a temperatura ambiente en termomezclador)Eluir en 17 ul (usar columna Mini-elute)

10. QC y agrupación

10 10.1 Nanodrop: el número es extraño (230 es muy alto)

10.2 Agregar 34 ul de TE, AMPure XP purifique la relación 1.8:1. Resuspender en 20 ul (remagnetización manual en tubo de PCR)

10.3	Nanodrop	Qubit ADNds HS	densidad de área
ID	ng/ul		
FV1	6.4	9.84	
24 hora	7.6	16.6	18.0=15.6
7 días	7.9	13.8	15.9=13.8
21 días	9.1	16.0	18.7=16.23

10.4 Gel de diagnóstico (2% e gel ex 11 min, 2 ul cada muestra)

15 (escalera Sage, escalera de 100 bp, escalera de 25 bp, muestras VDJ)

Diluir todas las muestras a la misma concentración más baja en ng/ul;

Muestra	Volumen	Agua Qbit	gel en agua
24 hora	12 ul	2.43	1.56
7 días	12 ul	0	0
21 días	12 ul	1.91	2.11

10.5 Mezclar 12 ul de cada muestra juntos. Almacenar a -20

Adjunto D

```
importar tipos
importar xml.eárbol.cElementoÁrbol como ElementoÁrbol
```

```
importar número como np
importar
```

Bio.SeqIO

```
importar seqherramientas
```

```
# importar refseg
# importar segutils
# importar alineamiento
# importar
```

```
agrupación
# importar LSF
# importar params
# importar análisis
```

```
# =====
# = DATOS
```

```
ESTRUCTURAS =
# =====
```

```
clase cadenainmune(objeto):
    """Estructura de datos a
```

```
representa una cadena inmune."""
```

```
    def __init__(auto,**kw):
        """Inicializar
```

Cadenainmune

```
        seq es 5' -> 3'
        """
```

```
        def kw_init(atrib):
```

```
            si kw.tiene_llave(atrib):
                auto._establecerattr_(atrib,kw[atrib])
```

```
            kw_init('seq')
            kw_init('descri')
            kw_init('locus')
            kw_init('v')
```

```
            kw_init('d')
            kw_init('j')
            kw_init('c')
            kw_init('unión')
```

```

    si kw.tiene_llave('etiquetas')
        etiquetas = kw['etiquetas']
        si

esinstancia(etiquetas,tipos.TiposdeCuerdas): etiquetas = [etiquetas]
    auto.etiquetas = establecer(etiquetas)

de otro modo:
    auto.etiquetas = establecer([])

    def obtener_cdr3(auto):
        regreso

len(auto.unión)
    def establecer_cdr3(auto, valor):
        pasar
    cdr3=

propiedad(fobtener=obtener_cdr3,festablecer=establecer_cdr3)

    def añadir_etiquetas(auto,estableceretiquetas):
        si

esinstancia(estableceretiquetas,tipos.TiposdeCuerdas): estableceretiquetas = [estableceretiquetas]
    auto.etiquetas = actualizar(estableceretiquetas)

    def añadir_etiqueta(auto,etiqueta):
        auto.añadir_etiquetas(etiqueta)

    def

retirar_etiquetas(auto,estableceretiqueta):
    si esinstancia(estableceretiquetas,tipos.TiposdeCuerdas): estableceretiquetas = [estableceretiquetas]
        para etiqueta en establecer etiquetas: auto.etiquetas.retirar(etiqueta)

    def

retirar_etiqueta(auto,etiqueta):
    auto.retirar_etiquetas(etiqueta)

    def tiene_etiqueta(auto,etiqueta):

si etiqueta en auto.etiquetas:
    regreso Verdadero
de otro modo:
    regreso Falso

def __len__(auto)
    regreso len(auto.seq)
def __str__(auto):
    regreso
auto.__repr__(auto):
    regreso auto.obtener_XML()

```

```

def
obtener_XML(auto):
    formato_xml = atrib lambda,valor:

    "\t<%(atrib)s>%(valor)s</%(atrib)s>\n" % {'atrib':atrib,'valor':valor}
    xmlcuerda

= '<Cadenalnmune>\n'
    para (atrib,valor) en auto.__dict__.iterarticulos():
        si

atrib == 'etiquetas':
    para etiqueta en auto.etiquetas:
        xmlcuerda +=

formato_xml('etiqueta', etiqueta)
    de otro modo:
        xmlcuerda +=

formato_xml(atrib,valor)
    xmlcuerda += '<Cadenalnmune>\n'
    regreso

xmlcuerda

# =====
# = ENTRADA/SALIDA =
# =====

clase
AnalizadorVDJXML(objeto):
    """" Analizador para VDJXML """"
    def __init__(auto)
        auto.cadena

= Ninguno

    def inicio_manejador(auto,elem):
        si elem.etiqueta == 'Cadenalnmune':

            auto.cadena = Cadenalnmune()

        def fin_manejador(auto,elem):
            si elem.etiqueta ==

'etiqueta':

                auto.cadena.afñadir_etiquetas(elem.texto)
                elif elem.etiqueta == 'v_fin_idx' o

elem.etiqueta. == 'j_inicio_idx':
                auto.cadena.__estableceratrib__(elem.etiqueta,int(elem.texto))

    de otro modo:
        auto.cadena.__estableceratrib__(elem.etiqueta,elem.texto)

def

```

```

análisis(auto,ingresararchivo):
    para evento, elem en

ElementoÁrbol.iteranálisis(ingresararchivo,eventos=('inicio','fin')):
    si evento == 'inicio':

        auto.inicio_manejador(elem)
    elif evento == 'fin':
        si

elem.etiqueta == 'Cadenalnmune':
    rendimiento auto.cadena
    de otro modo:

        auto.fin_manejador(elem)

clase AnalizadordePredicadoVDJXML (AnalizadorVDJXML):
    """VDJXML Analizador que toma una función de predicado"""
    def __init__(auto,predicado):

AnalizadorVDJXML.__init__(auto)
    auto.predicado = predicado

    def

análisis(auto,ingresararchivo):
    para evento, elemento en

ElementoÁrbol.iteranálisis(ingresararchivo,eventos=('inicio','fin')):
    si evento == 'inicio':

        auto.inicio_manejador(elem)
    elif evento = 'fin':
        si

elem.etiqueta == 'Cadenalnmune':
    si auto.predicado(auto.cadena) == Verdadero:

        rendimiento auto.cadena
    de otro modo

auto.fin_manejador(elem)

def análisis_VDJXML(ingresararchivo):
    vjxmlanalizador = AnalizadorVDJXML()

regreso vjxmlanalizador.análisis(ingresararchivo)

def filtro_análisis_VDJXML(ingresararchivo,predicado):

```



```

vdjxmlanalizador = AnalizadordePredicadoVDJXML(predicado)
regreso vdjxmlanalizador.análisis(ingresararchivo)

#
=====
# = Contando =
# =====

def recuentos_VJ(ingresararchivo)
    si
esinstancia(ingresararchivo,tipos.TiposdeCuerdas):
    ip = abrir(ingresararchivo,'r')
    elif
esinstancia(ingresararchivo,archivo):
    ip = ingresararchivo

    recuentos = np.ceros(
(len(refseq.IGHV_lista),len(refseq.IGHJ_lista)) )
    para cadena en análisisVDJXML(ip):
recuentos[refseq.IGHV_idx[cadena.v],refseq.IGHJ_idx[cadena.j]] += 1
    si
esinstancia(ingresararchivo,tipos.TiposdeCuerdas):
    ip.cerrar()

    regreso a recuentos

def
recuentos_VDJ(rep)_
    cn = np.ceros(
(len(refseq.IGHV_lista), len(refseq.IGHD_lista),len(refseq.IGHJ_lista)) )
    para cadena en
rep.cadenas:
cn[refseq.IGHV_idx[cadena.v],refseq.IGHD_idx[cadena.d],refseq.IGHJ_idx[cadena.j]] += 1
regreso cn

def remodelar_recuentos_VDJ_2D(recuentos):
    regreso
recuentos.remodelar(len(refseq.IGHV_lista), len(refseq.IGHD_lista),#len(refseq.IGHJ_lista))

def
recuento_dict_clon_idx(clon_idx,referencia_clones=Ninguno):

```

```

"""Toma un diccionario
de nombres de clúster mapeados para una secuencia de índices en una lista de CadenalInmune.

Regresa una
matriz np de la misma longitud que los referencia_clones con los recuentos de cada
clúster en los
referencia_clones.

La necesidad de referencia_clones se debe al hecho de que dividir un
repertorio dado
puede resultar en que algunas partes no observe ninguno de un clon dado, por lo que
debe haber una forma común
de comparar dos conjuntos de clones.

Si los clones de referencia se
omiten, se utilizará el establecer clones presentes en clon_idx.
"""
si
referencia_clones == Ninguno
    referencia_clones = clon_idx.llaves()
    recuentos =
np.zeros(len(referencia_clones))
    para (i,nombre) en enumerar(referencia_clones):
recuentos[i] = len(clon_idx.obtener(nombre,[ ]))
    regresar recuentos

def
recuento_dict_clon_recuentos(clon_recuentos,referencia_clones=Ninguno):
    si referencia_clones ==
Ninguno:
        referencia_clones = clon_recuentos.llaves.()
        recuentos =
np.zeros(len(referencia_clones))
        para (i,nombre) en enumerar(referencia_clones):
recuentos[i] = clon_recuentos.obtener(nombre,0)
        regresar recuentos

#
=====
# = Recuperando etiquetas y filtrando =
#
=====

```

```

def obtener_etiqueta_con_prefijo(cadena,prefijo):
    para etiqueta en
        cadena.etiquetas:
            si etiquea.iniciacon(prefijo):
                regresar etiqueta
            aumento del ErrorValor,
                "Etiqueta que inicia con" + prefijo + "no encontrada."

def obtener_clon(cadena):
    regresar
        obtener_etiqueta_con_prefijo(cadena,'clon')

def obtener_códigos de barras(cadena)
    tratar:
        regresar
            obtener_etiqueta_con_prefijo(cadena,'códigos de barras'?
            excepto ErrorValor:
                regresar ''

def
    filtro_etiquetas_y(etiquetas,enmanejo,fuerademanejo):
        si esinstancia(etiquetas,tipos.TiposdeCuerda): etiquetas =
            [etiquetas]
            etiquetas = establecer(etiquetas)
            para cadena en análisis_VDJXML(enmanejo):
                si etiquetas <=
                    cadena.todas_etiquetas: # prueba que todo en las etiquetas es en todas_etiquetas
                        imprimir
                            >>fuerademanejo, cadena

def filtro_etiquetas_o(etiquetas, enmanejo,fuerademanejo):
    si
        esinstancia(etiquetas,tipos.TiposdeCuerda): etiquetas = [etiquetas]
            etiquetas = establecer(etiquetas)
            vacío_establecer =
                establecer()
                para cadena en análisis_VDJXML(enmanejo):
                    si etiquetas & cadena.todas_etiquetas !=
                        vacío_establecer:# prueba que todas las etiquetas y todas_etiquetas comparten algo
                            imprimir >>fuerademanejo,
                                cadena

def es_completo_VJ(cadena):
    si (cadena.v en refseq.IGHV_seqs.llaves()) y (cadena.j en
        refseq.IGHJ_seqs.llaves()):

```

```

        regreso Verdadero
de otro modo:
        regreso Falso

def

obtener_clon_idxes(enmanejo):
    clústeres = { }
    i = 0
    para cadena en

análisis_VDJXML(enmanejo):
        tratar: clústeres[obtener_clon(cadena)] += [i]
    excepto

ErrordeLlave: clústeres[obtener_clon(cadena)] = [i]
    | += 1
    regresar clústeres

def

obtener_clon_recuentos(enmanejo):
    clústeres = { }
    para cadena en análisis_VDJXML(enmanejo):

        tratar: clústeres[obtener_clon(cadena)] += 1
        excepto ErrordeLlave: clústeres[obtener_clon(cadena)] =

1

    regresar clústeres

# =====
# = Funciones en canal =

=====

def vdjxml2fasta(enmanejo,fuerademanejo):
    para cadena en

análisis_VDJXML(enmanejo):
    imprimir >>fuerademanejo, '>' +cadena.descr
    imprimir

>>fuerademanejo, cadena.seq

# para generar identificadores a partir de combos VJ
def

vj_id(v_seg,j_seg):
    regresar seqherramientas.limpiar_id(v_seg)+'_'+seqherramientas.limpiar_id(j_seg)

def dividir_vdjxml_dentro_VJ_partes(enmanejo,fueradenombre):
    partes = [ ]
    vj_ids = [ ]

```

```

fuerademanijos = { }

    # abrir los archivos de salida para todos los combos VJ
    i = 0
    para v_seg en

refseq.IGHV_seqs.llaves():
    para j_seg en refseq.IGHJ_seqs.llaves():

curr_fueradenombre = fueradenombre + '.' + str(i)
    curr_vj_id = vj_id(v_seg,j_seg)

    partes.append(curr_fueradenombre)
    vj_ids.append(curr_vj_id)

fuerademanijos[curr_vj_id] = abrir(curr_fueradenombre, 'w')
    i +=1

    para cadena en

análisis_VDXML(enmanejo):
    curr_vj_id = vj_id(cadena.v,cadena.j)

>>fuerademanijos[curr_vj_id], cadena

    para manejo en desordenados.valoresinter()_
manejo.cerrar()

    regresar (partes,vj_ids)

def análisis_VDXML_partes(partes)
    para parte

en partes:
    para cadena en análisis_VDXML(parte):
        rendimiento de cadena

def

esperar_para_subproceso(proceso_lista,intervalo=30):
    terminado = Falso
    mientras no

terminado:
    terminado = Verdadero
    tiempo.dormir(intervalo)
    para p en

proceso_lista:
    si p.poll() == Ninguno:
        terminado = Falso

romper
adeventencias de importación

```

```

importar copia

importar número como np

importar seqherramientas

importar vdj
importar

refseq
importar sequtils
importar núcelodealineación

advertencias.filtrosimple('siempre')

clase

vdj_alineador(objeto):

    def __init__(auto,**kw):

auto.numCrudoVCandidatos = 5
    auto.numCrudoVCandidatos = 10

auto.numCrudoVCandidatos = 2

    # Definir patrones de semilla

patrónA= '111011001011010111'
    patrónB='1111000100010011010111'

patrónC='111111111111'
    patrónD='11010000110001010111'

patrónE='1110111010001111'
    auto.patronesdesemilla =

[patrónA, patrónB, patrónC, patrónD, patrónE]
    auto.patronesdesemillamini =

['111011','110111']
    auto.patránPos = '111111111111'

    auto.puntajeminV

= 100 # derivado de datosde calibración 20090710
    auto.puntajeminD = 4

auto.puntajeminJ = 13

    # establecer secuencias de referencia (locus) y generar picadillos

de datos de ref
    auto.locus = kw['locus']

    auto.refv_seqs =

```

ES 2 730 951 T3

```
refseq.__obteneratributo__(auto.locus+'V_seqs')
    auto.Vseqlistallaves =

vdj_alineador.seqdict2kmers( auto.refV_seqs, auto.patronesdesemilla )

auto.refJ_seqs = refseq.__obteneratributo__(auto.locus+'J_seqs')
    auto.Jseqlistallaves =

vdj_alineador.seqdict2kmers( auto.refJ_seqs, auto.patronesdesemilla )

    tratar:    #

este locus puede no tener segmentos D
    auto.refD_seqs =

refseq.__obteneratributo__(auto.locus+'D_seqs')
    auto.Dseqlistallavesmini =

vdj_alineador.seqdict2kmers( auto.refD_seqs, auto.patronesdesemillamini )
    excepto

ErrordeAtributo:
    pasar

    auto.refV_compensar =

refseq.__obteneratributo__(auto.locus+'V_compensar')
    auto.refJ_compensar =

refseq.__obteneratributo__(auto.locus+'J_compensar')

    # Generar datos de referencia

para secuencia ID positiva
    posVseqlistallaves = vdj_alineador.seqdict2kmers (
auto.refV_seqs, [auto.patronPos] )
    posJseqlistallaves = vdj_alineador.seqdict2kmers (
auto.refJ_seqs, [auto.patronPos] )
    negVseqlistallaves = vdj_alineador.seqdict2kmers (
vdj_alineador.seqdict2revcompseqdict(auto.refV_seqs), [auto.patronPos] )
    negVseqlistallaves = vdj_alineador.seqdict2kmers(
vdj_alineador.seqdict2revcompseqdict(auto.refJ_seqs), [auto.patronPos] )

    #

recoger llaves posibles
    posestablecer = establecer ([])
    para llave en posVseqlistallaves.llaves():

        posestablecer.actualizar(posVseqlistallaves[llave][auto.patronPos])
    para llave en

posJseqlistallaves.llaves():
```

```

        posestablecer.actualizar(posJseqlistallaves[llave][auto.patronPos])

    negestablecer = establecer ([ ])
    para llave en negVseqlistallaves.llaves():

negestablecer.actualizar(negVseqlistallaves[llave][auto.patronPos])
    para llave en

negJseqlistallaves.llaves():
    negestablecer.actualizar(negJseqlistallaves[llave][auto.patronPos])

    # obtener únicas llaves para versiones positivas o negativas del establecer referencia

posestablecernuevo = posestablecer – negestablecer
negestablecernuevo = negestablecer – posestablecer

    auto.posestablecer

= posestablecernuevo
    auto.negestablecer = negestablecernuevo

def

Valign_cadenas(auto, cadena,verboso=Falso):
    pregunta = cadena.seq

    # calcular

picadillos de seq pregunta
    preguntallaves = vdj_alineador.seqdict2kmers(pregunta,auto.patronesdesemilla)

    # para cada segmento de referencia V y cada patrón, ¿cuántos k-mers compartidos hay
    ahí?
    puntajesV_picadillo = vdj_alineador.puntajepicadillo(auto.Vseqlistallaves,preguntallaves)

    # obtener puntajes más altos de numCrudoVCandidatos en puntajesJ y almacenar sus nombres en
    orden

descendente
    buenoVseglista = ordenado(auto.refV_seqs.llaves(),llave=lambda k:

puntajesV_picadillo[k],inverso=Verdadero) [0:auto. numCrudoVCandidatos]
    buenoVsegdict =

dict([(seg,auto.refV_seqs[seg]) para seg en buenoVseglista )

    #

Needleman-Wunsch de segmento V
    (mejorVseg,mejorpuntajeV,mejorcuadrodemandoV,mejorVtrazamat) =

```



```

vdj_alineador.mejoralignNW(buenVsegdict,pregunta,auto.puntajeVmin)

    # si exitoso

alineamiento
    si mejorVseg no es Ninguno:
        cadena.V = mejorVseg

    # encontrar el límite de CDR3
    # alineamiento del constructo

Valnref,Valnrefcoords,Valnpregunta,Valnpreguntacoords = vdj_alineador.constructo_alineamiento (
auto.refV_seqs[mejorVseg], pregunta, mejorcuadrodemandov,mejorVtrazamat )

    #

encontrar el límite de CDR3
    cadena.v_fin_idx = vdj_alineador.reducirVregión( Valnref,
Valnref, Valnpregunta,Valnrefcoords, auto.refV_compensar[mejorVseg] )

regresar mejorVpuntaje

    def Jalign_cadena(auto,cadena,verboso=False):
        # tratar

poda región V para alineamiento J
    tratar:
        pregunta =

cadena.seq[cadena.v_fin_idx:]
    excepto ErrordeAtributo:
        pregunta = cadena.seq

    # cálculo de picadillo de seq pregunta
    preguntallaves =

vdj_alineador.seq2kmers(pregunta,auto.patronesdesemilla)

    # para cada referencia J

segmento y cada patrón, ¿cuántos k-mers compartidos hay ahí?
    puntajeJ_picadillo =

vdj_alineador.puntajepicadillo(auto.Jseqlistallaves,preguntallaves)

    # obtener

puntajes más altos de numCrudoJCandidatos en puntajesJ y almacenar sus nombres en
orden descendente

    buenoJseglista = ordenado(auto.refJ_seqs.llaves(),llave=lambda k:

```

```

puntajesJ_picadillo[k],inverso=Verdadero) [0:auto. numCrudoJCandidatos]
    buenoJsegdict =

dict([(seg,auto.refV_seqs[seg]) para seg en buenoJseglista] )

    #

Needleman-Wunsch de segmento J
    (mejorJseg,mejorpuntajeJ,mejorcuadrodemandoJ,mejorJtrazamat) =

vdj_alineador.mejoralignNW(buenoJsegdict,pregunta,auto.puntajeJmin)

    # si exitoso

alineamiento
    si mejorJseg no es Ninguno:
        cadena.J = mejorJseg

    # encontrar el límite de CDR3
    # alineamiento del constructo

Jalnref,Jalnrefcoords,Jalnpregunta,Jalnpreguntacoords = vdj_alineador.constructo_alineamiento (
    auto.refJ_seqs[mejorJseg], pregunta, mejorcuadrodemandoJ,mejorJtrazamat )

    #
encontrar el límite de CDR3
    j_inicio_compensar = vdj_alineador.reducirJregión( Jalnref,

Jalnref, Jalnpregunta, Jalnpreguntacoords,Jalnrefcoords, auto.refJ_compensar[mejorJseg] )
    # tratar

    cadena.j_inicio_idx = cadena.v_fin_idx + j_inicio_compensar
    excepto

ErrordeAtributo:
    cadena.j_inicio_idx = j_inicio_compensar

    regresar

mejorpuntajeJ

def Dalign_cadena(auto,cadena,verboso=Falso) :
    # podar V

y regiones J para alineamiento de D
    # no deberíamos estar intentando el alineamiento de D a menos que

tenemos
    # un CD3 bien definido
    pregunta = cadena.unión

    # cálculo

picadillos de seq pregunta
    preguntallaves =

```

```

vdj_alineador.seq2kmers(pregunta,auto.patronesdesemillamini)

    # para cada referencia D
segmento y cada patrón, ¿cuántos k-mers compartidos hay ahí?
    puntajeD_picadillo =

vdj_alineador.puntajepicadillo(auto.Dseqlistallavesmini,preguntallaves)

    # obtener

puntajes más altos de numCrudoJCandidatos en puntajesJ y almacenar sus nombres en
orden descendente

    buenoDseglista = ordenado(auto.refD_seqs.llaves(),llave=lambda k:
puntajesD_picadillo[k],inverso=Verdadero) [0:auto. numCrudoDCandidatos]
    buenoDsegdict =
dict([(seg,auto.refD_seqs[seg]) para seg en buenoDseglista] )

    #

Needleman-Wunsch de segmento J
    (mejorDseg,mejorpuntajeD,mejorcuadrodemandoD,mejorDtrazamat) =

vdj_alineador.mejoralignNW(buenDsegdict,pregunta,auto.puntajeDmin)

    # si exitoso

alineamiento
    si mejorDseg no es Ninguno:
        cadena.d = mejorDseg

regresar mejorpuntajeD

    def align_cadena(auto,cadena,verboso=Falso) :

si no cadena.tiene_etiqueta('positiva') y no cadena.tiene_etiqueta('codificación'):

advertencias.advertir('%s de la cadena pueden no ser la cadena correcta de % de cadena.descr)

puntajes = { }

    puntajes['v'] = auto.Valign_cadena(cadena,verboso)

puntajes['j'] = auto.Jalign_cadena(cadena,verboso)

    # solo proceso de unión si V

y J exitosamente
    si tieneattr(cadena,'v') y tieneattr(cadena,'j'):

```

```

cadena.unión = cadena.seq[cadena.v.fin_idx:cadena.j_inicio_idx]

#
solo align D si yo estoy en un locus que tiene cadenas D
si auto.locus en
['IGH','TRB','TRD']:
    puntajes['d'] = auto.Dalign_cadena(cadena.verboso)

regresar puntajes

def align_seq(auto,seq,verboso=False):
    cadena =
vdj.CadenaInmune(descr='secuencia?',seq=seq)
    auto.align_cadena(cadena,verboso)

regresar cadena

def codificación_cadena(auto,cadena,verboso=False):
    hebra=

auto.seq2codificación(cadena.seq)
si hebra == -1:
    cadena.seq =

seqherramientas.inverso_complemento(cadena.seq)
    cadena.añadir_etiqueta('revcomp')

cadena.añadir_etiqueta('codificación')

def seq2codificación(auto,seq) :
    seqllaves =

vdj_alineador.seq2kmers(seq, [auto.patónPos])
    seqpalabras = seqllaves[auto.patónPos]

hebraid = 1
    si len(auto.negestablecer & seqpalabras) > len(auto.posestablecer & seqpalabras):

    hebraid = -1
    regresar hebraid

@métodoestático
def
seq2kmers(seq,patrones):
    """ Secuencia dada y patrones, para cada patrón, calcular

todos los k-mers que corresponden de la secuencia.

El resultado es

```

```

sequnno[patrón] [llave] = [pos1,pos2,...,posN] en seq
                        seqllaves[patrón] =

establecer([kmers])

    """"
    seqllaves = { }
    patlente = []
    para

patrón en patrones:
    patlente.append(len(patrón))
    seqllaves[patrón]=

establecer()

    máxpatlente = máx(patlente)

    para i en rangox(len(seq)):

        palabra = seq[i:i+ máxpatlen]
        para patrón en patrones:

patlen = len(patrón)
    si len(palabra) >= patlen:
        llave= ' '

        para j en rangox(patlen):
            si patrón[j] == '1':

                llave += palabra[j]
                seqllaves[patrón].adicionar(llave)

    regresar seqllaves

@métodoestático
def seqdict2kmers(seqdict,patrones):

    seqlistallaves = { }
    para seq en seqdict.iterartículos():

seqlistallaves[seq[0]] = vdj_alineador.seq2kmers(seq[1],patrones)
    regresar seqlistallaves

@métodoestático
def puntajepicadillo(refllaves,preguntallaves):
    """"Calcular el número de

llaves comunes para cada secuencia de referencia.

    preguntallaves es dict de establecers, donde dict

llaves son patrones
    llaves de referencia es dict de ref seqs, donde cada elt es un

```

dict de patrones con establecers como valores. los patrones deben ser los mismos

```

puntuajes = { }
para seg en regllaves.iterllaves():
    puntuaje = 0
    para
        patrón en preguntallaves.iterllaves():
            puntuaje += len( rellaves[seg] [patrón] &
preguntallaves[patrón] )
            puntuajes[seg] ) puntuaje
regresar puntuajes

@métodoestático
def mejoralignNW(candidatodict,pregunta,puntajemin) :
    mejorseg = Ninguno

    mejorpuntuaje = puntajemin
    mejorcuadrodemando = Ninguno
    mejortrazamat = Ninguno

    seq2 = pregunta
    para (seg,seq1) en candidatodict.iterartículos():
        # C

implementación:
    # extraer la memoria
    # tenga en cuenta que estamos usando cero

condiciones iniciales, entonces las matrices son también inicializadas
# notación es como Durbin

p.29
    puntuajes = np.ceros( [len(seq1)+1, len(seq2)+1] )
    lx = np.ceros(
[ len(seq1)+1, len(seq2)+1 ] )
    ly = np.ceros( [ len(seq1)+1, len(seq2)+1 ] )

    traza = np.ceros( [ len(seq1)+1, len(seq2)+1 ] ), dtipo=np.int)
alineamientonúcleo.alignNW( puntuajes, lx, ly, traza, seq1, seq2 )

    currpuntuaje =

vdj_alineador.puntuajeVJalign(puntuajes)
    si currpuntuaje > mejorpuntuaje:

```

```

mejorpuntaje = currpuntaje
                mejorseg = seg
                mejorcuadrodemando = puntajes

                mejortrazamat = traza

            regresar

(mejorseg,mejorpuntaje,mejorcuadrodemando,mejortrazamat)

    @métodoestático
    def

mejoralignSW(candidatodict,pregunta,puntajemin) :
    mejorseg = Ninguno
    mejorpuntaje =

puntajemin
    mejorcuadrodemando = Ninguno
    mejortrazamat = Ninguno

    seq2 =

pregunta
    para (seg,seq1) en candidatodict.iterarticulos():
        # C implementación:

        # extraer la memoria
        # tenga en cuenta que estamos usando cero condiciones iniciales,

entonces las matrices son también inicializadas
        # notación es como Durbin p.29

puntajes = np.ceros( [len(seq1)+1, len(seq2)+1] )
        traza = np.ceros( [len(seq1)+1,

len(seq2)+1] dtype=np.int)
        alineamientonúcleo.alignSW( puntajes, traza, seq1, seq2 )

        currpuntaje = vj_alineador.puntajeDJalign(puntajes)

        si currpuntaje

> mejorpuntaje:
        mejorpuntaje = currpuntaje
        mejorseg = seg

        mejorcuadrodemando = puntajes
        mejortrazamat = traza

    regresar

(mejorseg,mejorpuntaje,mejorcuadrodemando,mejortrazamat)

    @métodoestático
    def

```

ES 2 730 951 T3

```
reducirVregión( alnref, alnrefcoords, alnpregunta, alnpreguntacoors, compensar ):
    """ Reducir V
```

región fuera de secuencia pregunta basada en alineamiento.

Ref dada y pregunta

alineamientos de región V, refID, y de la secuencia
pregunta original, regresar a secuencia

con la región V cortada, dejando
el 2nd-CYS. También necesita las coords de alineamiento de pregunta.

```
    """
    # FR3fin = auto.refV_compensar[refID] - alnrefcoords[0]      #
```

posición del primer candidato
FR3fin = compensar - alnrefcoords[0] # posición

primer candidato
refdiferencias = alnref[:FR3fin].recuento('-') # contar diferencias hasta

CYS putativo pos
diferenciasvistas += refdiferencias
FR3fin += refdiferencias #

ajustar si para diferencias en alineamiento ref
refdiferencias = alnref[:FR3fin].recuento('-') -

diferenciasvistas # ¿cualquier diferencia adicional'!

```
    preguntadiferencias = alnpregunta[:FR3fin].recuento('-')
```

```
    # v_fin_idx = idx de inicio de aln de pregunta + distancia dentro aln - # de diferencias
```

```
v_fin_idx = alnpreguntacoors[0] + FR3fin - preguntadiferencias
```

```
    regresar v_fin_idx
```

```
@métodoestático
```

```
def reducirJregión( alnref, alnrefcoords, alnpregunta, alnpreguntacoors,
```

```
compensar ):
    """ Reducir región J fuera de secuencia pregunta basada en alineamiento.
```

Ref dada y alineamientos de pregunta de región J, refID, y de la secuencia
pregunta

ES 2 730 951 T3

original, regresar a secuencia con la región J cortada, dejando el J-TRP. También

necesita las coords de alineamiento de pregunta.

```

"""
# FR4inicio =
auto.refJ_compensar[refID] - alnrefcoords[0] # posición del primer candidato de J-TRP inicio
FR4inicio = compensar - alnrefcoords[0] # posición del primer candidato de J-TRP inicio
refdiferencias = alnref[:FR4inicio].recuento('-'). # contar diferencias hasta TRP putativo pos
diferenciasvistas
= 0
    mientras refdiferencias > 0: # iterativamente encuentra todas las diferencias hasta el TRP
diferenciasvistalas += refdiferencias
    F4inicio += refdiferencias # ajustar para diferencias en alineamiento ref
    refdiferencias = alnref[:FR4inicio].recuento('-') - diferenciasvistas # ¿cualquier diferencia adicional?

preguntadiferencias = alnpregunta[:FR4inicio].recuento('-')
# j_inicio_compensar = idx de
inicio de aln de pregunta + distancia dentro aln - # de diferencias
# nota: j_inicio_compensar es de
la seq pregunta posada
j_inicio_idx = alnpreguntacoords[0] + F4inicio - preguntadiferencias

regresar j_inicio_compensar

@métodoestático
def
constructo_alineamiento(seq1,seq2,cuadro de mando,trazamat):
    """ Alineamiento del constructo de ref
segmento para pregunta del puntaje y matrices de traza. """
    nfilas,ncols = cuadro de mando.forma

    # hacer alguna comprobación de errores
    si len(seq1)+1 != nfilas o len(seq2)+1 !=
ncols:
        aumento Excepción, "nfilas y ncols deben ser igual a len(seq1)+1 y
len(seq2)+1"
```

```

# traducir trazas enteras a coords
deltas = {
0 : (1,1)
  1 : (1,0),
  2 : (0,1),
  3 : (0,0)
}

# calcular col donde debe comenzar alineamiento
si nfilas <= ncols:

col = np.argmax( cuadro de mando[nfilas-1, :])
fila = nfilas - 1
de otro modo:

col = ncols - 1
fila = np.argmax( cuadro de mando[ncols-1, :])

# si
fila es coord en matriz, fila - 1 es coord en seq (b/c de condiciones init)

aln1
= seq1[fila-1]
aln2 = seq2[col-1]

aln1fin = fila
aln2fin =

col

mientras (fila-1 > 0) y (col-1 > 0) :
# calcular la dirección de
movimientos
filacambio,colcambio = deltas [trazamat[fila,col] ]

# emitir simbolos apropiados
si filacambio == 1:
fila -= 1

aln1 = seq1[fila-1] + aln1
elif filacambio == 0:
aln1 = '-'

+aln1
de otro modo:
aumento Excepción, "Salto contenido en matriz de traza de
más de una fila/col."
si colcambio == 1:
col

```

-= 1

```
    aln2 = seq2[col-1] + aln2
    elif colcambio == 0:
```

```
    aln2 = '.' + aln2
    de otro modo:
        aumento Excepción, "Salto contenido
```

en matriz de traza de más de una fila/col."

```
    aln1inicio = fila-1
```

```
aln2inicio = col - 1
```

```
    # las coords se refieren a coords en la secuencia (pitónico)
```

```
    regresar aln1, (aln1inicio,aln1fin), aln2, (aln1inicio,aln2fin)
```

@métodoestático

```
def puntajeVJalign(puntajematrix):
    "Puntaje de cómputo de alineamiento de V
```

dada la matriz de puntaje de Needleman-Wunsch

```
    ASUME núm. de filas < núm. de cols, es decir,
```

```
reqseq V seg es en el eje vertical
```

```
"""
```

```
    nfilas,ncols =
```

```
puntajematrix.forma
```

```
    si nfilas <= ncols:
        regresar np.máx(
```

```
puntajematrix[nfilas-1, :] )
```

```
    de otro modo:
```

```
        regresar np.máx( puntajematrix[:,ncols-1, :] )
```

@métodoestático

```
def puntajeDalign(puntajematrix):
    "Puntaje de cómputo de alineamiento
```

de D dada la matriz de puntaje de Smith-Waterman

```
"""
```

```
    regresar np.máx(
```

```
puntajematrix )
```

@métodoestático

```

def seqdict2revcompseqdict(seqdict) :
    revcompdict = { }
    para artículo en seqdict.iterartículos () :
        revcompdict[artículo[0] ] =
            sequtils.reversa_complemento(artículo[1])
    regresa revcompdict

clase
    vdj_alineador_combinado(objeto):
        """vdj alineador para cadenas 'livianas'

        esta clase
        realizará alineamiento para ambos loci, por ejemplo, IGK e IGL
        y elige uno con el mejor puntaje
        V
        """
        def __init__(auto,**kw):
            auto.loci = kw['loci']

        auto.alineadores = [vdj_alineador(locus=locus) para locus en auto.loci]

        auto.patronesPos = '111111111111'
        auto.posestablecer = establecer()
        auto.negestablecer = establecer()

        para alinador en auto.alineadores:
            auto.posestablecer.actualizar(alineador.posestablecer)

        auto.negestablecer.actualizar(alineador.negestablecer)

        def align_cadena(auto,cadena,verboso=False) :

            alineamientos = [ ]
            para alineador en auto.alineadores:
                curr_cadena =
                    copia.copiaprofunda(cadena)
                curr_puntaje = alineador.align_cadena(curr_cadena)

            alineamientos.append( (curr_cadena,curr_puntaje) )
            alineamientos = ordenado(filtro(lambda a:
                tieneattr(a[0], 'v')alineamientos),llave=lambda a: a[1] ['v'],reversa=Verdadero)

            len(alineamientos) > 0:
                mejorcadena = alineamientos [0] [0]
                si

```

```

tieneattr(mejorcadena,'v')
    cadena.v = mejorcadena.v

cadena.v_fin_idx = mejorcadena.v_fin_idx
    si tieneattr(mejorcadena,'j'):

cadena.j = mejorcadena.j
    cadena.j_inicio_idx = mejorcadena.j_inicio_idx

si tieneattr(mejorcadena,'unión'):
    cadena.unión= mejorcadena.unión

    si tieneattr(mejorcadena,'d'):
        cadena.d= mejorcadena.d
    regresar

alineamientos[0] [1]. # NOTA: solo regreso los puntajes al éxito aln

def

align_seq(auto,seq,verboso=False):
    cadena =

vdj.CadenaInmune(descr='secuencia',seq=seq)
    auto.align_cadena(cadena,verboso)

regresar cadena

def codificando_cadena(auto,cadena) :
    hebra =

auto.seq2codificando(auto,cadena) :
    si hebra == -1:
        cadena.seq =

seqherramientas.reversa_complemento(cadena.seq)
    cadena.añadir_etiqueta('revcomp')

cadena.añadir_etiqueta('codificando')

def seq2codificando(auto,seq) :
    seqllaves =

vdj_alineador.seq2kmers(seq, [auto.patónPos] )
    seqpalabras = seqllaves[auto.patónPos]

hebraid = 1
    si len(auto.negestablecer & seqpalabras) > len(auto.posestablecer & seqpalabras) :

    hebraid = -1
    regresar hebraid

def igh_alineador():
    regresar

```

```

vdj_alineador('locus=IGH')

def igk_alineador():
    regresar vdj_alineador(locus='IGK')

def
    igl_alineador():
        regresar vdj_alineador(locus='IGL')

def igkl_alineador():
    regresar

vdj_alineador_combinado(loci=['IGK','IGL'])

def trb_alineador():
    regresar

vdj_alineador(locus='TRB')

def tra_alineador():
    regresar vdj_alineador(locus='TRA')

def
    trd_alineador():
        regresar vdj_alineador(locus='TRS')

def trg_alineador():
    regresar

vdj_alineador(locus='TRG')
importar número como np
de número importar ma
importar scipy como sp
importar

scipy.estadísticas
importar scipy.especial
importar matplotlib como mpl
importar matplotlib.pyplot como plt

importar vdj

def códigos de barras_clon_recuentos(enmanejo) :
    """Regresar recuento dict de archivo vdjxml

con recuentos[códigos de barras] [clon]"""
    recuentos = dict()
    para cadena en

vdj.análisis_VDJXML(enmanejo):
    tratar: # cadena puede no tener códigos de barras

recuentos_códigos de barras[cadena.clon] = recuentos_códigos de barras.obtener(cadena.clon,0) + 1

```

```

regresar recuentos

def códigos de barras_unión_recuentos(enmanejo) :
    """Regresar recuento dict de
    archivo vdjxml con recuentos[códigos de barras] [unión]"""
    recuentos = dict()
    para cadena en
        vdj.análisis_VDJXML(enmanejo):
            tratar: # cadena puede no tener códigos de barras

recuentos_códigos de barras= recuentos_establecerpredeterminado(cadena.códigos de barras, dict() )
    excepto ErrordeAtributo:

        continuar
        recuentos_códigos de barras[cadena.unión] =
recuentos_códigos de barras.obtener(cadena.unión,0) + 1

def
códigos de barras_clon_recuentos2matriz(recuentos,códigos de barras=Ninguno,clones=Ninguno) :
    """Genera matriz de
    recuento dict"""
    si códigos de barras == Ninguno:
        códigos de barras = recuentos.llaves()
    si clones ==
Ninguno:
        clones = lista( reducir( lambda x,y: x|y, [establecer(c.llaves()) para c en
recuentos.valoresiter()] ) )
        matriz = np.ceros ( (len(clones), len(códigos de barras) ) :
        para
            (col,códigos de barras) en enumerar(clones):
                matriz[fila,col] = recuentos.obtener(códigos de barras,dict() ).obtener(clon,0)
                regresar
        (clones, códigos de barras, matriz)

códigos de barras_unión_recuentos2matriz = códigos de barras_clon_recuentos2matriz

#
=====
# = VIEJO VIEJO VIEJO VIEJO VIEJO
VIEJO VIEJO VIEJO VIEJO VIEJO VIEJO VIEJO VIEJO VIEJO VIEJO VIEJO =
#
=====

```

```

# =====
# =

Series de tiempo =
# =====

def clon_tiemposeries(enmanejo, códigos de barras,
referencia_clones=Ninguno) :
    # generar datos de recuento
    para cadena en
vdj.análisis_VDXML(enmanejo):
    recuentos

def
clon_tiemposeries(enmanejo, tiempo_etiquetas, referencia_clones=Ninguno) :
    # obtener datos de recuento

tiempo_etiquetas_establecer = establecer(tiempo_etiquetas)
clon_recuentos = { }
para etiqueta en tiempo_etiquetas:

clon_recuentos [etiqueta] = { }
para cadena en vdj.análisis_VDXML(enmanejo):
    tratar:

curr_tiempo_etiqueta = (cadena.etiquetas&tiempo_etiquetas_establecer).pop()
excepto ErrordeLlave:

continuar

    tratar:

clon_recuentos[curr_tiempo_etiqueta] [vdj.obtener_clon(cadena)] += 1
excepto ErrordeLlave:

clon_recuentos[curr_tiempo_etiqueta] [vdj.obtener_clon(cadena)] = 1

    # establecer clones de referencia

si referencia_clones == Ninguno:
    referencia_clones = establecer ()
    para recuentos en

clon_recuentos.valoresiter():
    referencia_clones.actualizar(recuentos.llaves())

referencia_clones = lista(referencia_clones)

    # construir matriz seriesdetiempo
    núm_clones

```



```

= len(referencia_clones)
    núm_tiempos = len(tiempo_etiquetas)
    datosderecuento =

np.ceros( (núm_clones,núm_tiempos) )
    para (i,etiqueta) en enumerar(tiempo_etiquetas) :

datosderecuento[:, i] = vdj.recuento_dict_clon_recuentos(clon_recuentos[etiqueta],referencia_clones)

regresar a datosderecuento, referencia_clones

def
seriesdetiempo2dimensiones(datosderecuento,freq=Verdadero,log=Verdadero,pseudorrecuento=1e-1) :
    núm_tiempo_series,

núm_tiempos = datosderecuento.forma
    núm_transiciones = núm_tiempos - 1
    si pseudorrecuento != 0:

        dimensiones = np.ceros( (núm_tiempo_series,núm_transiciones) )
        datosderecuento_pseudo =

datosderecuento + np.flotar_(pseudorrecuento)
    si freq = Verdadero
        datosderecuento_pseudo =

datosderecuento_pseudo / datosderecuento_pseudo.sum(eje=0)
    para i en rango(núm_transiciones) :

        dimensiones[:, i] = datosderecuento_pseudo[:, i+1] / datosderecuento_pseudo[:, i]
    de otro modo: #

solo mirar a las series de tiempo que no son cero en todo el camino
    idxs =

np.sum(recuentodedatos>0,eje=1)==núm_tiempos
    dimensiones =

np.ceros( (np.sum(idxs),núm_transiciones) )
    si freq == Verdadero

datosderecuento_modificado = np.flotar_(datosderecuento) / datosderecuento.sum(eje=0)
    de otro modo:

datosderecuento_modificado = np.flotar_(datosderecuento)
    para i en rango(núm_transiciones):

        dimensiones[:, i] = datosderecuento_modificado[idxs, i+1] / datosderecuento_modificado[idxs, i]
    si

log==Verdadero
    regresar a np.log10(dimensiones)

```

de otro modo:
regresar a dimensiones

def

seriesdetiempo2autocorrelación(seriesdetiempo):
ac = [1.] +

[sp.estadísticas.pearsonr(seriesdetiempo[:-i],seriesdetiempo[i:]) [0] para i en
rango(1,len(seriesdetiempo)-2)]

regresar a ac

=====
= Tipos de espectro =
=====

def

cdr3s2tipodeespectro(cdr3s):
mín_crudo_cdr3 = np.mín(cdr3s)
máx_crudo_cdr3 = np.máx(cdr3s)

mín_cdr3 = np.int(np.hacertecho(mín_crudo_cdr3 / 3.) *3) # será un múlt de 3 diferente a cero

máx_cdr3 = np.int(np.piso(máx_crudo_cdr3 / 3.) *3) # será un múlt de 3

contener las

longitudes de CDR3. El primer elt es rep cero len(y debe ser cero)
y el último bin

siempre representa uno mayor que el múlt más grande de 3
cdr3scontenidos=

np.histograma(cdr3s,contenedores=np.unrango(0,máx_cdr3+2)) [0] # el 2+ se debe a la pecul.
de

np.hist.

gaussianos = []
para cdr3len en np.unrango(mín_cdr3,máx_crudo_cdr3,3) :

totalcdr3s = np.sum(cdr3scontenidos[cdr3len-1:cdr3len+2])
buencdr3s =

cdr3scontenidos[cdr3len]
si totalcdr3s == 0 :
continuar
mu = cdr3len

x = cdr3len-0.5
cola = (1- (np.flotar(buencdr3s)/totalcdr3s)) / 2.

sigma = (x-mu) / (np.sqrt(2.)*sp.especial.erfinv(2*cola-1))
rv =

```

sp.estadísticas.norm(loc=mu,escala=sigma)
    gaussianos.append( (totalcdr3s,rv) )

    t =

np.linspace(0,máx_cdr3+1,1000)
    y = np.zeros(len(t) )
    para (s,rv) en gaussianos:

y += s*rv.pdf(t)
    regresar a (t,y)

def tipodeespectro_curvas(enmanejo):
    # NOTA: requiere

cadenas con V y J alns

    # estructura de datos init
    cdr3s = { }
    para v_seg en

vdj.refseq.IGHV.seqs.llaves( ):
    para j_seg en vdj.refseq.IGHJ.seqs.llaves( ):

cdr3s[vdj.vk_id(v_seg,j_seg) ] = [ ]

    # datos de carga
    para cadena en

análisis_VDJXML(enmanejo):
    si cadena.v == '' o cadena.j == '' o cadena.unión ==
    '':
        continuar
        cdr3s[vdj.vj_id(cadena.v,cadena.j)] append(cadena.cdr3)

espectros = { }
    para v_seg en vdj.refseq.IGHV.seqs.llaves( ):
    para j_seg en

vdj.refseq.IGHJ.seqs.llaves( ):
    si len(cdr3s[vdj.vj_id(v_seg,j_seg) ]) == 0:

        # combo VJ vacío:
        espectros[vdj.vj_id(v_seg,j_seg) ] =

(np.arreglo([0,150],np.arreglo([0,0]) )
    de otro modo:

espectros[vdj.vj_id(v_seg,j_seg) ] = cdr3s2tipodeespectro(cdr3s[vdj.vj_id(v_seg,j_seg) ] ):

regresar a espectros

```

```

# =====
# = Estimación de diversidad =
#
# =====

def estimador_chao1(recuentos):
    " " " Sesgo corregido. Ver

Estimaciones doc(Colwell) " " "
    s = len(recuentos)
    F1=

np.flotar_(np.sum(np.int_(recuentos)==1) )
    F2 = np.flotar_(np.sum(np.int_(recuentos)==2) )

chao1 = Sobs + F1*(F1-1)/(2*F2+1) )
    regresar a chao1

def

estimador_chao1_varianza(recuentos):
    F1 = np.flotar_(np.sum(np.int_(recuentos)==1) )
    F2 =

np.flotar_(np.sum(np.int_(recuentos)==2) )
    si F1 > 0 y F2 > 0:
        chao1_var =

(F1*(F1-1)/(2*(F2+1))) + (F1*(2*F1-1)*(2*F1-1)/(4*(F2+1)*(F2+1))) +
(F1*F1*F2*(F1-1)*(F1-1)/(4*(F2+1)*(F2+1)*(F2+1)*(F2+1)))
    elif F1 > 0 y F2 == 0:

Schao1 = estimador_chao1(recuentos)
    chao1_var = (F1*(F1-1)/2)+

(F1*(2*F1-1)*(2*f1-1)/4) - (F1*F1*F1*F1/(4*Scaho1) )
    elif F1 == 0:
        N =

np.flotar_(np.sum(recuentos) )
    Sobs = np.flotar_(len(recuentos) )
    chao1_var =

Sobs*np.exp(-1*N* Sobs) * (1-np.exp(-1*N* Sobs) )
    regresar a chao1_vat

def

estimador_ace(recuentos,raro_cortar=10) :
    Sobs = np.flotar_(len(recuentos) )
    Sraro =

np.flotar_(np.sum(np.int_(recuentos)<=raro_cortar) )
    Sabund = Sobs - Sraro

```

```

F1 =
np.flotar_(np.sum(np.int_(recuentos)==1) )
  F =lambda i :
np.flotar_(np.sum(np.int_(recuentos)==i) )
  Nraro = np.flotar_(np.sum([ i*F(i) para i en
rango(1,raro_cortar+1] ] ) )
  #Nraro =
np.flotar_(np.sum(recuentos[np.int_(recuentos)<=raro_cortar] ) )
  si Nraro == F1: # de acuerdo

con Estimaciones
  regresar a estimador_chao1(recuentos)
  Cace = 1 - (F1/Nraro)

gama_alcuadrado = Sraro*np.sum( [ i*(i-1)*F(i) para i en
rango(1,raro_cortar+1] ] ) / (Cace*Nraro*(Nraro-1) )
  si gama_alcuadrado < 0:

gama_alcuadrado = 0
  Sace = Sabund + (Sraro/Cace) + (F1/Cace)*gama_alcuadrado
  regresar a

Sace

def acumulación_curva(muestra,muestreo_niveles) :
  pasar

#
=====
# = Utilidades estáticas      =
# =====

def
recuentos2muestra(recuentos):
  " " " Calcula una muestra consistente de un vector de recuentos.

Toma un vector de recuentos y devuelve un vector de índices x
  tal que len(x) = sum(c)

y cada elt de x es el índice de
  un elt correspondiente en c
  " " "
  x =

np.unos(np.sum(recuentos),dtipo=np.int_)

  inicio_idx = 0
  fin_idx = 0
  para i en

```

```

rangox(lent(recuentos) ) :
    inicio_idx = fin_idx
    fin_idx = fin_idx + recuentos[i]

    x[inicio_idx:fin_idx] = x[inicio_idx:fin_idx] * i
    regresar a x

def

muestra2recuentos(muestra) :
    """ Vector de recuento de regreso de las lista de muestras.

    El

    etc ordenamiento es ignorado; solo la unicidad
    de los objetos es considerada.
    """

    núm_categorías = len(establecer(muestra) )
    recuento_dict = { }
    para elt en muestra:
        tratar:

    recuento_dict[elt] += 1
    excepto ErrordeLlave: recuento_dict[elt] = 1
    regresar

    recuento_dict.valores( )

# def muestra2recuentos(muestra, categorías=0) :
#     """ Vector de recuento

    de regreso de las lista de muestras.
#
#     Tomar el vector de muestras y regresar al vector de

    recuentos. Los etls
#     se refieren a índices en algo que en última instancia se asignaría a la
#
#     categoría de origen (como de un multinomial). Por lo tanto, si
#     hay,

    por ejemplo, 8 categorías, los valores válidos en la muestra deben ser 0-7.
#     Si no se dan

    categorías, entonces lo calculo a partir del valor más alto
#     presente en la muestra (+1).
#     """
#
#     recuentos = np.contenerrecuento(muestra)
#     si(categorías > 0) y (categorías > len(recuentos) ) :
#
#         recuentos = np.append( recuentos, np.ceros(categorías-len(recuentos)) )
#     regresar

    recuentos

```

```

def puntajeenpercentil(valores,rango) :
    regresar

sp.estadísticas.puntajeenpercentil(valores,rango) :

def percentiildelpuntaje(valores,rango) :

valores.clasificar()
    regresar a valores.búsquedaclasificada(puntaje) / np.flotar_(len(valores) )

def

arranque(x, nbota, theta, *args) :
    '''regresar a replicaciones de arranque n de teta desde
x'''
    N = len(x)
    th_estrella = np.ceros(nbota)

    para i en rangox(nbota) :

th_estrella [i] = theta ( x[ np.aleatorio.randint(0,N,N) ], *args ) # repl de arranque desde
x

    regresar a th_estrella

def submuestra( x, núm_muestras, muestra_tamaño, theta, *args) :
    """ regresar

evaluaciones de núm_muestras de theta estadística
en submuestras de muestra_tamaño de tamaño """

N = len(x)
th_estrella = np.ceros(núm_muestras)

para i en rangox(núm_muestras) :

th_estrella [i] = theta ( x[ np.aleatorio.randint(0,N,muestra_tamaño) ], *args ) # submuestra
desde x

    regresar a th_estrella

def randint_sin_reemplazo(bajo,alto=Ninguno,tamaño=Ninguno) :

si alto == Ninguno :
    alto = bajo
    bajo = 0
si tamaño == Ninguno:

```

```

    tamaño = 1

urn = rango(bajo,alto)
N = len(urn)
dar vuelta = Falso
si tamaño > N/2:
    dar vuelta =

Verdadero
    tamaño = N - tamaño
    muestra = []
    para i en rangox(tamaño) :
        dibujar =

np.aleatorio.radint(0,N-1)
    muestra.append(urn.pop(dibujar) )
si no se voltea:

regresar a np.asarreglo(muestra)
de otro modo:
    regresar np.asarreglo(urn)

def

submuestra_sin_reemplazo(x, núm_muestras, muestra_tamaño, theta, *args) :
    " " " regresar

evaluaciones de núm_muestras de theta estadística
en submuestras de muestra_tamaño de tamaño" " "

N = len(x)
th_estrella = np.ceros(núm_muestras)

para i en rangox(núm_muestras) :

    th_estrella [i] = theta ( x[ radint_sin_reemplazo(0,N,muestra_tamaño) ], *args )    #

submuestra desde desde x

    regresar a th_estrella

# =====
# = Visualización =
#
=====

clase ConstAnchoRectángulo(mpl.parches.Parche) :

    def __init__(auto,
x, y1, y2, w, **kwargs) :
    auto.x = x
    auto.y1 = y1
    auto.y2 = y2

```



```

    auto.w = w
        mpl.parches.Parche.__init__(auto, **kwargs)

    def

obtener_camino(auto):
    regresar a mpl.camino.Camino.unidad_rectángulo()

    def

obtener_transformar(auto) :
    caja = np.arreglo( [ [ auto.x,auto.y1],

[ auto.x,auto.y2] ] )
    caja = auto.ejes.Datosdetrans.transformar(caja)

    w =

auto.w * auto.ejes.cajab.ancho / 2.0

    caja[0,0] -= w
    caja[1,0] += w

    regresar a mpl.transforma.TransformarCajaBA(mpl.transforma.CajaB(caja) )

clase
ConstAnchoLinea(mpl.líneas.Línea2D) :

    def __init__(auto, x, y, w, **kwargs) :
        auto.x

= x
        auto.y = y
        auto.y = y

mpl.líneas.Línea2D.__init__(auto, [0,1], [0,0], **kwargs) # init a la línea de unidad

    def

obtener_transformar(auto) :
    # definir transformar que toma la línea del horiz de la unidad seg
    # y

colóquela en la posición correcta utilizando las coords de
    # visualización

    caja =

np.arreglo( [ [ auto.x,auto.y],
                [ auto.x,auto.y+1] ] )
    caja =

auto.ejes.Datosdetrans.transformar(caja)

    w = auto.w * auto.ejes.cajab.ancho / 2.0

```

```

        caja[0,0] -= w
        caja[1,0] += w

        #xdisp,ydisp =

auto.ejes.DatosdeTrans.transformar_punto([auto.x,auto.y])
        #xdisp -= w
        #izquierda =

xdisp - w
        #derecha = xdisp + w

        regresar

mpl.transforma.TransformarCajaBA(mpl.transforma.CajaB(caja) )
        # regresar

mpl.transforma.Afin2D().escala(w,1).traducir(xdisp,ydisp)

def

dibujar(auto,renderizador) :
        # el ÚNICO propósito de redefinir esta función es forzar al
objeto
        # de Línea2D a ejecutar recaché(). De lo contrario, ciertos cambios en la escala

        # no invalidan el objeto de Línea2D, y la transarreglo no se
        # volverá a calcular

(y, por lo tanto, las coords de los Ejes que se calcularon antes serán obsoletos)
        auto.recaché()

regresar a mpl.líneas.Línea2D.dibujar(auto,renderizador)

clase

ConstAlturaRectángulo(mpl.parches.Parche) :

        def __init__(auto, x1, x2, y, h,

**kwargs) :
        auto.x1 = x1
        auto.x2 = x2
        auto.y = y
        auto.h =

h

        mpl.parches.Parche.__init__(auto, **kwargs)

        def obtener_camino(auto):

regresar a mpl.camino.Camino.unidad_rectángulo()

        def obtener_transformar(auto) :
        caja =

```

```

np.arreglo( [ [auto.x,auto.y],
              [auto.x2,auto.y] ] )
    caja =

auto.ejes.DatosdeTrans.transformar(caja)

    h = auto.h * auto.ejes.cajab.altura /

2.0

    caja[0,1] -= h
    caja[1,1] += h

    regresar

mpl.transforma.TransformarCajaBA(mpl.transforma.CajaB(caja) )

clase
ConstAlturaLinea(mpl.lineas.Linea2D):
    def __init__(auto,x,y,h,**kwargs):

auto.x = x
    auto.y = y
    auto.h = h

mpl.lineas.Linea2D.__init__(auto, [0,0], [0,1], **kwargs) # init a la línea de unidad
    #

auto.x = x
    # auto.y = y
    # auto.w = w
    #

mpl.lineas.Linea2D.__init__(auto, [0,1], [0,0], **kwargs) # init a la línea de unidad

    def

obtener_transformar(auto) :
    # definir transformar que toma la línea del horiz de la unidad seg
    # y

colóquela en la posición correcta utilizando las coords de
    # visualización

    caja =

np.arreglo( [ [ auto.x,auto.y],
              [ auto.x+1,auto.y] ] )
    caja =

auto.ejes.DatosdeTrans.transformar(caja)

    h = auto.h * auto.ejes.cajab.altura /

2.0

    caja[0,0] -= h
    caja[1,1] += h

```

```

        #xdisp,ydisp =
auto.ejes.Datosdetrans.transformar_punto([auto.x,auto.y])
        #xdisp -= w
        #xizquierda =

xdisp - w
        #xderecha = xdisp + w

        regresar

mpl.transforma.TransformarCajaBA(mpl.transforma.CajaB(caja) )
        # regresar

mpl.transforma.Afin2D().escala(w,1).traducir(xdisp,ydisp)

def

dibujar(auto,renderizador) :
        # el ÚNICO propósito de redefinir esta función es forzar al
objeto
        # de Línea2D a ejecutar recaché(). De lo contrario, ciertos cambios en la escala

        # no invalidan el objeto de Línea2D, y la transarreglo no se
        # volverá a calcular

(y, por lo tanto, las coords de los Ejes que se calcularon antes serán obsoletos)
        auto.recaché()

regresar a mpl.líneas.Línea2D.dibujar(auto,renderizador)

def cajagráfica(ax, x, posiciones=Ninguna,
anchos=Ninguna, vert=1) :
        # adaptado desde Matplotlib

        # convertir x a la lista de

vectores
        si tieneattr(x, 'forma'):
            si len(x.forma) == 1:
                si
tieneattr(x[0], 'forma')
                x = lista(x)
            de otro modo:
                x =

[x,]
        elif len(x.forma) == 2:
            nr, nc = x.forma
            si nr == 1:

                x = [x]
            elif nc == 1:
                x = [x.desvariar () ]

```

de otro modo:

`x = x[:,i]` para `i` en `rango(nc)`

de otro modo:
levantar

ErrorValor, "x de entrada no puede tener más de tan 2 dimensiones"
si no tieneattr(x[0],

'__len__'):

`x = [x]`
`col = len(x)`

obtener alguna info de gráfica
si posiciones

es Ninguna:

`posiciones = rango(1, col+1)`
si anchos es Ninguno:
`anchos =`

`mín(0.3/len(posiciones),0.05)`
si esinstancia(anchos,flotar) p esinstancia(anchos, int) :

`anchos = np.unos((col,), flotar) * anchos`

dar vuelta a columnas, sumando

a gráfica

para `i`, `pos` en `enumerar(posiciones)` :
`d = np.complicación(x[i])`
`fila =`

`len(d)`

si `fila==0`:
sin datos, omitir esta posición
continuar

obtener info de distrib
`q1, med, q3 = mpl.mlab.prctile(d,[25,50,75])`
`dmáx =`

`np.máx(d)`
`dmín = np.mín(d)`

`línea_color = '#074687'`

`cara_color = '#96B7EC'`
si `vert == 1`:
`medlínea =`

`ConstAnchoLinea(pos,med,anchos[i],color=línea_color,zorden=3)`
`caja =`

`ConstAnchoRectángulo(pos,q1,q3,anchos[i],colordecara=cara_color,colordelborde=línea_color,zo`
`rden=2)`

`vertlínea=`
`mpl.líneas.Linea2D([pos,pos], [dmín,dmáx], color=línea_color,zorden=1)`

```

de otro modo:
    medlínea = ConstAlturaLínea(med,pos,anchos[i],color=línea_color,zorden=3)

    caja =

ConstAlturaRectángulo(q1,q3,pos,anchos[i],colorde cara=cara_color,colordelborde=línea_color,z
orden=2)
    vertlínea=
mpl.líneas.Línea2D([dmín,dmáx], [pos,pos], color=línea_color,zorden=1)

    ax.adicionar_línea(vertlínea)
    ax.adicionar_parche(caja)

ax.adicionar_línea(medlínea)

#####
#####
#####

def
rep2tipodeespectro(rep) :
    " " " Calcular curvas de tipodeespectro para el objeto de Repertorio. " " "

cdr3s = np.fomación([c.cdr3 para c en rep si c-uni3n != ' '])
    mín_crudo_cdr3 =

np.mín(cdr3s)
    máx_crudo_cdr3 = np.máx(cdr3s)
    mín_cdr3 = np.int(np.hacertecho( mín_crudo_cdr3 /

3.) *3) # será un múlt de 3 diferente a cero
    máx_cdr3 = np.int(np.piso(máx_crudo_cdr3 / 3.) *

3) # será un múlt de 3

    # contener las longitudes de CDR3. El primer elt es rep cero len

(y debe ser cero)
    # y el último bin siempre representa uno mayor que el

múlt más grande de 3
    cdr3scontenidos= np.histograma(cdr3s,contenedores=np.unrango(0,máx_cdr3+2) ) [0] # el 2+

se debe a la pecul. de np.hist.

    gaussianos = [ ]
    para cdr3len en

np.unrango(mín_cdr3,máx_crudo_cdr3,3) :

```

```

totalcdr3s =
np.sum(cdr3scontenidos[cdr3len-1:cdr3len+2] )
    buencdr3s = cdr3scontenidos[cdr3len]

si totalcdr3s == 0 :
    continuar
    mu = cdr3len
    x = cdr3len-0.5

cola = (1- (np.flotar(buencdr3s)/totalcdr3s) ) / 2.
sigma = (x-mu) /

(np.sqrt(2.)*sp.especial.erfinv(2*cola-1) )
rv = sp.estadísticas.norm(loc=mu,escala=sigma)

gaussianos.append( (totalcdr3s,rv) )

t = np.linspace(0,máx_cdr3+1,1000)
y =

np.ceros(len(t) )
para (s,rv) en gaussianos:
    y += s*rv.pdf(t)
regresar

(t,y)

def
dispersión_repertorios_ontología(reps,info='VJCDR2',datosbuenos=Falso,medición='proporci
ones'):

    " " "Crear una cuadrícula de gráficas de dispersión que muestre correlaciones entre todos los pares de
repertorios.

    reps -- lista de objetos de Repertorios

    " " "
    númreps = len(reps)

listadedatos = [ ]
para rep en reps:
    listadedatos.append(

vdj.recuentos_ontología_1D(rep,info,datosbuenos) )

    si medición == 'proporciones':

para i en rangox(len(listadedatos) ) :
    listadedatos[i] = np.flotar_(listadedatos[i]) /

```

```

np.sum(listadedatos[i])

    mín_nocero = np.mín([np.mín(datos[datos>0] para datos en
listadedatos])
    máx_nocero = np.máx([np.máx(datos[datos>0] para datos en listadedatos])
    ejelo
= 10**np.piso( np.frexp(mín_nocero) [1] * np.log10(2) )
    ejehi = 10**np.hacertecho (
np.frexp(máx_nocero) [1] * np.log10(2) )

    fig = plt.figura ()

    hist_axs = [ ]

    para fila en rangox(númreps) :
        col = fila
        gráficánúm = númreps*fila + col + 1

        ax = fig.adicionar_subgráfica(númreps,númreps,gráficánúm)

ax.hist(listadedatos[filas], contenedores=100,log=Verdadero,colordecara='k')
    hist_axs.append(ax)

dispersión_axs = [ ]
    para fila en rangox(númreps-1) :
        para col en
rangox(fila+1,númreps)
            gráficánúm = númreps*fila + col + 1
            ax =
fig.adicionar_subgráfica(númreps,númreps,gráficánúm)
ax.dispersión(listadedatos[filas],listadedatos[cols],c='k',marcador='o',s=2,coloresdelborde=Niguno)
ax.establecer_escalax('log')
    ax.establecer_escalay('log')

ax.eje([ejelo,ejehi,ejelo,ejehi])
    dispersión_axs.append(ax)

    regresar

fig

def dispersión_repertorios_clústeres(reps,refclústeres,medición='proporciones'):
    " " "Crear una cuadrícula de gráficas de dispersión que muestre correlaciones entre todos los pares de
repertorios.

```



```

reps -- lista de objetos de Repertorios

"""
númreps = len(reps)

listadedatos = []
para rep en reps:
    clústeres = vdj.obtenerClústeres(rep)

listadedatos.append( vdj.recuentosClústeres(clústeres,refclústeres) )

si medición ==

'proporciones':
    para i en rangox(len(listadedatos) ) :
        listadedatos[i] =

np.flotar_(listadedatos[i]) / np.sum(listadedatos[i])

mín_nocero =

np.mín([np.mín(datos[datos>0] para datos en listadedatos])
máx_nocero =

np.máx([np.máx(datos[datos>0] para datos en listadedatos])
ejelo = 10**np.piso(

np.frexp(mín_nocero) [1] * np.log10(2) )
ejehi = 10**np.hacertecho ( np.frexp(máx_nocero) [1]

* np.log10(2) )

fig = plt.figura ()

hist_axs = []
para fila en

rangox(númreps) :
    col = fila
    gráficánúm = númreps*fila + col + 1
    ax =

fig.adicionar_subgráfica(númreps,númreps,gráficánúm)

ax.hist(listadedatos[fila], contenedores=100,log=Verdadero,colordecara='k')
hist_axs.append(ax)

dispersión_axs = []
para fila en rangox(númreps-1) :
    para col en

rangox(fila+1,númreps)
    gráficánúm = númreps*fila + col + 1
    ax =

fig.adicionar_subgráfica(númreps,númreps,gráficánúm)

```

```
ax.dispersión(listadedatos[fil],listadedatos[col],c='k',marcador='o',s=0.5,coloresdelborde=Ninguno)
```

```
ax.establecer_escalax('log')
    ax.establecer_escalay('log')
```

```
ax.eje([ejelo,ejehi,ejelo,ejehi])
    dispersión_axs.append(ax)
```

regresar

fig

```
def reps2seriesdetiempo(reps,refclústeres):
    """Regresar a matriz de series de tiempo de lista u
```

objs de Repertorio en orden cron.

reps es lista de objetos de Repetorio
refclústeres

es la lista maestra de clústeres de referencia

```
"""
    númreps = len(reps)
    númclústeres =
```

```
len(refclústeres)
```

```
    datosderecuento = np.ceros( (númclústeres,númreps) )
    para (i,rep) en
```

```
enumerar(reps) :
    clústeres = vdj.objetenerClústeres(rep)
    datosderecuento[:, i] =
```

```
vdj.recuentosClústeres(clústeres,refclústeres)
```

regresar a datosderecuento

def

```
seriesdetiempo_repertorios(tiempos,reps,refclústeres,idxsbool=Ninguno,todopositivo=Falso) :
```

""" Crear unas series de tiempo de clústeres diferente en refclústeres.

Si todopositivo es

Verdad, entonces se limitará el mismo a dibujar seriesdetiempo
solo para aquellos clústeres que son

no cero en todos los puntosdetiempo.

```
"""
    ax = plt.gca()
```

```
    númreps = len(reps)
```

```
    númclústeres = len(refclústeres)
```

```

datosderecuento = np.ceros( (númclústeres,númreps) )
para
(i,rep) en enumerar(reps) :
    clústeres = vdj.objetenerClústeres(rep)
    datosderecuento[:, i] =
vdj.recuentosClústeres(clústeres,refclústeres)

sumas = datosderecuento.suma(0)
proporciones =

np.flotar_(datosderecuento) / sumas

si idxsbool == Ninguno:
    si todospositivo == Verdadero:

        idxsbool = np.suma(proporciones,eje=1) > 0
    de otro modo:
        idxsbool =

np.arreglo([Verdadero]*proporciones.forma[0] )

ax.gráfica(tiempos,datosderecuento[idxsbool, :].transponer(),'k-',anchodelínea=0.2)

plt.dibujar_si_interactivo()
regresar a ax

def rep2tipodeespectro(rep) :
    " " " Calcular
curvas de tipodeespectro para el objeto de Repertorio. " " "

    cdr3s = np.fomación([c.cdr3 para c en rep

si c-unión != ' '])
    mín_crudo_cdr3 = np.mín(cdr3s)
    máx_crudo_cdr3 = np.máx(cdr3s)

mín_cdr3 = np.int(np.hacertecho( mín_crudo_cdr3 / 3.) *3) # será un múlt de 3 diferente a cero
máx_cdr3 = np.int(np.piso(máx_crudo_cdr3 / 3.) * 3) # será un múlt de 3

# contener las
longitudes de CDR3. El primer elt es rep cero len (y debe ser cero)
# y el último bin

siempre representa uno mayor que el múlt más grande de 3
cdr3scontenidos=

np.histograma(cdr3s,contenedores=np.unrango(0,máx_cdr3+2) ) [0] # el 2+ se debe a la pecul.
de

np.hist.

```

```

gaussianos = [ ]
para cdr3len en np.unrango(mín_cdr3,máx_crudo_cdr3,3) :

totalcdr3s = np.sum(cdr3scontenidos[ cdr3len-1: cdr3len+2] )
buencdr3s =

cdr3scontenidos[ cdr3len]
si totalcdr3s == 0 :
    continuar
mu = cdr3len

x = cdr3len-0.5
cola = (1- (np.flotar(buencdr3s)/totalcdr3s) ) / 2.

sigma = (x-mu) / (np.sqrt(2.)*sp.especial.erfinv(2*cola-1) )
rv =

sp.estadísticas.norm(loc=mu,escala=sigma)
gaussianos.append( (totalcdr3s,rv) )

t =

np.linespacio(0,máx_cdr3+1,1000)
y = np.ceros(len(t) )
para (s,rv) en gaussianos:

y += s*rv.pdf(t)
regresar a (t,y)

def

mapacircularVJ( ax,recuentos,etiquetasdefil=Ninguno,escala='lineal') :
    númV =

recuentos.forma[0]
númJ = recuentos.forma[1]
X, Y = np.rejillademalla(rango(númJ),rango(númV))

# enmascarar las posiciones cero
X,Y = ma.arreglo(X), ma.arreglo(Y)
C = ma.arreglo(recuentos)

máscaracero = (recuentos == 0)
X.máscara = máscaracero
Y.máscara = máscaracero
C.máscara =

máscaracero

# etls nocero complicación (eliminar posiciones cero)
x = ma.comprimido(X)

y = ma.comprimido(Y)
c = ma.comprimido(C)

```

ES 2 730 951 T3

```
# log de recuentos normalizados si es solicitado

si escala = 'log':
    c = ma.log10(c)

    si escala == 'lineal' o escala ==

'log':
    # recuentos normalizados para rango de tamaño deseado
    máx_recuentos = ma.máx(c)

    mín_recuentos = ma.mín(c)
    recuentos_rango = máx_recuentos - mín_recuentos

máx_tamaño = 100
mín_tamaño = 5
tamaño_rango = máx_tamaño - mín_tamaño

tamaños = (np.flotar(tamaño_rango) / recuentos_rango) * (c - mín_recuentos) + mín_tamaño

si

escala == 'personalizar' :
    trans_recuentos = 1000
    lineal_posiciones = c >= trans_recuentos

    log_posiciones = c < trans_recuentos

    mín_tamaño = 3
    trans_tamaño =

40 # 30
    máx_tamaño = 200 # 150
    log_tamaño_rango = trans_tamaño - mín_tamaño

lineal_tamaño_rango = máx_tamaño - trans_tamaño

    lineal_máx_recuentos =

ma.máx(c[lineal_posiciones])
    lineal_mín_recuentos = ma.mín(c[lineal_posiciones])

lineal_recuentos_rango = lineal_máx_recuentos - lineal_mín_recuentos
    log_máx_recuentos =

ma.máx(c[log_posiciones])
    log_mín_recuentos = ma.mín(c[log_posiciones])

log_recuentos_rango = np.log10(log_máx_recuentos) - np.log10(log_mín_recuentos)

tamaños = np.ceros(len(c))
```

ES 2 730 951 T3

```
tamaños[lineal_posiciones] = (np.flotar(lineal_tamaño_rango) /
lineal_recuentos_rango) * (c[lineal_posiciones] - lineal_mín_recuentos) + trans_tamaño
tamaños[log_posiciones] = (np.flotar(log_tamaño_rango) / log_recuentos_rango) *
(ma.log10[log_posiciones] - ma.log10(log_mín_recuentos)) + mín_tamaño
colección =
mpl.colecciones.CírculoColección(
    tamaños,
    compensaciones = zip(x,y),
transCompensar = ax.transDatos, # es posible que necesite establecer explícitamente la inarreglo xlim y ylim
para que
funcione correctamente
    coloresdecara = '#1873C1',
    anchosdelínea = 0.25,
acortar_en = Falso)
    ax.adicionar_colección(colección)
    ax.establecer_aspecto('igual')
ax.autoescala_ver()
ax.ejex.establecer_mayor_localizador(mpl.ticker.LocalizadorFijo(rango(recuentos.forma[1])))
ax.ejey.establecer_mayor_localizador(mpl.ticker.LocalizadorFijo(rango(recuentos.forma[0])))
    si
etiquetasdefila != Ninguna:
ax.ejex.establecer_mayor_formateador(mpl.ticker.FormateadoeFijo(etiquetasdecol))
    si etiquetasdecol !=
Ninguno:
    ax.ejey.establecer_mayor_formateador(mpl.ticker.FormateadoeFijo(etiquetasdefila))
para etiquetatick en ax.ejex.obtener_etiquetastick():
etiquetatick.establecet_alineamientohorizontal('izquierda')
etiquetatick.establecer_rotación(-45)
```

```

etiquetatick.establecer_tamaño(8)

    para etiquetatick en ax.ejey.obtener_etiquetastick():
etiquetatick.establecer_tamaño(8)

    si escala == 'lineal' o escala == 'log':
        regresar

(mín_recuentos,máx_recuentos) , (mín_tamaño,máx_tamaño)
de otro modo:
    regresar

(lineal_mín_recuentos,trans_recuentos,log_máx_recuentos) , (mín_tamaño,trans_tamaño,máx_tamaño)

# definir

mapadecolor para -1 a 1 (verde-negro-rojo) como expresión de gen
rojoverdedict = { 'rojo' : [(0.0,
0.0, 0.0),
                        (0.5, 0.0, 0.0),
                        (1.0,
1.0, 0.0)],
                'verde': [(0.0, 0.0, 1.0),
                        (0.5, 0.0, 0.0),
                        (1.0, 0.0, 0.0)],
                'azul': [(0.0, 0.0, 0.0),
                        (0.5, 0.0, 0.0),
                        (1.0, 0.0, 0.0)]}

rojoverde =

mpl.colores.mapadeColorSegmentadoLineal('rojoverde', rojoverdedict,256)
rojoverde.establecer_malo(color='w')
importar número como np
importar scipy como sp
importar scipy.clúster

importar vdj
importar nucleodeagrupación

def pdist(X,métrico):
    m = len(X)
    dm = np.cero((m
* (m - 1) / 2,), dtype=np.doble)
    k = 0
    para i en rangox(0, m - 1):
        para j en

```

```

rangox(i+1, m)
    dm[k] = métrico(X[i], X[j])
    k += 1
    regresar a dm

def clústeres_seqs(seqs,cortar=4.5,enlace='único):
    # revisar casos triviales
    si
        len(seqs) == 0
            regresar (np.arreglo[ ], { })
            # levantar Excepción, " cadenas no tienen
                nada"

            # colapsar seqs idénticas entre si
            único_seqs =

        lista(establecer(seqs) )
        seq_idx = dict ( [(j,i) para (i,j) en enumerar(único_seqs)] )

        # revisar caso trivial
        si len(único_seqs) == 1:
            T = np.arreglo([1]*len(seqs) )

            regresar (T,seq_idx)

            # calcular la matriz de distancia
            Y = pdist ( único_seqs,
                nucleodeagrupación.levenshtein )

            # calcular el enlace
            Z =

        sp.clúster.jerarquía.enlace(Y,método=enlace)

        # determinar los clústeres al nivel de
            corte
            T = sp.clúster.jerarquía.fclúster(Z,cortar,criterio='distancia')

            regresar
                (T,seq_idx)
            importar tipos

            importar vdj

            # def análisis_VDXML(archivodeentrada):
            #     " " "Generador para

            regresar a objetos de la Cadenalnmune de un archivo vdxml.
            #
            #     Utiliza bibliotecas XML python.
            #
            #     " " "
            #     # NOTA: esto probablemente se puede hacer más elegante si se implementa

```



```

#      # como una

clase de
#      # variables de estado global.
#      cadena = Ninguno
#      daros_regulador = Ninguno
#

#      def inicio_manejador(nombre,atributos) :
#          si nombre = 'Cadenalnmune':
#

#          cadena = Cadenalnmune()
#          de otro modo:
#              datos_regulador = ''
#
#      def

fin_manejador(nombre):
#          si nombre == 'Cadenalnmune' :
#              cadena de rendimiento
#

nombre de elif == 'etiqueta':
#          cadena.adicionar_etiquetas(datos_regulador)
#          de otro modo:
#

cadena.__establecerattr__(nombre,datos_regulador)
#
#      def datos_manejador(datos).
#

datos_amortiguador += datos
#
#          analizadordexml = xml.analizadores.expat.CrearunAnalizador()
#

analizadordexml.ManejadordeElementodeInicio = inicio_manejador
#      analizadordexml.ManejadordeDatosdeCarácter = datos_manejador
#
#      si no

tieneatr(archivodenentrada,'leer'):
#          archivodeentrada = abrir(archivodeentrada, 'r')
#
#

Analizadordexml.ArchivodeAnálisis(archivodeentrada)

#

=====
# = INICIO DE DESAPROBADO
=====
#

```

=====

def

análisis_VDJXML_antiguo(archivodeentrada):

 " " " Cargar unos datos del archivo VDJXML como un Repertorio o lista

de CadenasInmunes

 NOTA: este fn NO utiliza las bibliotecas XML; este implementa un

analizador de manual

 que toma la línea de entrada por línea.

 ESTO ASUME QUE CADA ELEMENTO XML

TOMA UNA Y SOLO UNA LINEA

 " " "

 si

esinstancia(archivodeentrada,tipos.TiposdeCuerda) :

 ip = abrir(archivodeentrada,'r')

 elif

esinstancia(archivodeentrada,archivo) :

 ip = archivodeentrada

 númCadenas = 0

posible_elementos = [

 'descr',

 'seq',

 'v',

 'd'

 'j',

 'ighc',

 'cdr3',

 'unión',

 'func',

 'etiqueta'

]

para línea en ip:

 línea = línea.cuerda()

 elementosfinalespos = línea.encontrar('>') + 1

 elementoxml = línea[0:elementosfinalespos]

 elemento = elementoxml[1:-1]

```

si elementoxml == '<CadenalnMune>':
    cadena = vdj.CadenalnMune()
    elif

elementoxml == '<CadenalnMune>':
    nmCadenas += 1
    cadena de rendimiento

elemento elif en posible_ elementos:
    si elemento == 'etiqueta':
        datosdeetiqueta

= lnea[elementosfinalespos:-1*( elementosfinalespos+1)].divisin('l')
    si datosdeetiqueta[0] ==

'experimento':
    cadena.__establecerattr__('experimento', 'l'.unin(datosdeetiqueta[1:]))

    elif datosdeetiqueta[0] == 'clon':

cadena.__establecerattr__('clon', 'l'.unin(datosdeetiqueta[1:]))
    elif datosdeetiqueta[0] ==

'cdigos de barras':
    cadena.__establecerattr__('cdigos de barras', 'l'.unin(datosdeetiqueta[1:]))

    elif datosdeetiqueta[0] == 'v_fin_idx':

cadena.__establecerattr__('v_fin_idx', 'l'.unin(datosdeetiqueta[1:]))
    elif datosdeetiqueta[0] ==

'j_inicio_idx':
    cadena.__establecerattr__('j_inicio_idx', 'l'.unin(datosdeetiqueta[1:]))

de otro modo:

cadena.adicionar_etiquetas(lnea[elementosfinalespos:-1*( elementosfinalespos+1)])
    elemento elif ==

'ighc':

    datoselt = lnea[elementosfinalespos:-1*( elementosfinalespos+1)]

si datoselt != '':
    cadena.c = datoselt
    de otro modo:

    pasar
    elementos elif == 'func':
        pasar
    de otro modo:

    datoselt = lnea[elementosfinalespos:-1*( elementosfinalespos+1)]
    si datoselt !=

```

```

    cadena.__establecerattr__(elemento,datoselt)
de otro modo:

    pasar

si esinstancia(archivodeentrada,tipos.TiposdeCuerda) :
    ip = cerrar()

#
=====
# = FIN DE DESAPROBADO
=====
#
=====

# def

filtrar_análisis_VDJXML(archivodeentrada,predicado):
#     " " " Cargar unos datos del archivo VDJXML como un

Repertorio o lista de CadenasInmunes
#
#     predicar es una función que toma una cadena

y la regresa Verdadera o Falsa. Cosas
#     que regresan falso son omitidas.
#
#     NOTA:

este fn NO utiliza las bibliotecas XML; este implementa un analizador de manual
#     que toma la

línea de entrada por línea.
#
#     ESTO ASUME QUE CADA ELEMENTO XML TOMA UNA Y SOLO UNA

LÍNEA
#
#     " " "
#
#     si esinstancia(archivodeentrada,tipos.TiposdeCuerda) :
        ip

= abrir(archivodeentrada,'r')
#     elif esinstancia(archivodeentrada,archivo) :
#         ip = archivodeentrada
#
#     númCadenas = 0
#
#     posible_elementos =[
#         'descr',
#
#         'seq',
#         'v',

```

```

#         'd'
#         'j',
#
#         'ighc',
#         'cdr3',
#         'unión',
#
'func',
#         'etiqueta'
#         ]
#     para línea en ip:
#
línea = línea.cuerda()
#         elementosfinalespos = línea.encontrar('>') + 1
#         elementoxml =
línea[0:elementosfinalespos]
#         elemento = elementoxml[1:-1]
#
#         si
elementoxml == '<Cadenalnmune>':
#         cadena = Cadenalnmune()
#         elif elementoxml
== '<Cadenalnmune>':
#         númCadenas += 1
#         si predicado(cadena) ==
Verdadero:
#         cadena de rendimiento
#         de otro modo:
#         pasar
#
elemento elif en posible_elementos:
#         si elemento == 'cdr3':
#
cadena.cdr3 = eval(línea[elementosfinalespos:-1*(elementosfinalespos+1)]
#         elemento elif ==
'etiqueta':
#         cadena.adicionar_etiquetas(línea[elementosfinalespos:-1*(elementosfinalespos+1)])
#
de otro modo:
#
cadena.__establecerattr__(elemento,línea[elementosfinalespos:-1*(elementosfinalespos+1)])
#
#         si
siesinstancia(archivodeentrada.tipos.TiposdeCuerda) :
#         ip = cerrar()
""" params.py
Definir

```

nombres de directorio y archivo que deben ser modificados manualmente al punto a ciertos recursos

PIRATEAR. Averiguar mejor manera para referirse a este directorio

vdj_dir =

'/Usuarios/laserson/buscar/iglesia/código/lib/vdj'

dir de datos empaquetados

datos_dir = 'datos'

IGHV_fasta = 'IGHV.fasta'

IGHD_fasta = 'IGHD.fasta'

IGHJ_fasta = 'IGHJ.fasta'

IGKV_fasta =

'IGKV.fasta'

IGKJ_fasta = 'IGKJ.fasta'

IGLV_fasta = 'IGLV.fasta'

IGLJ_fasta = 'IGLJ.fasta'

TRBV_fasta = 'TRBV.fasta'

TRBD_fasta = 'TRBD.fasta'

TRBJ_fasta = 'TRBJ.fasta'

TRAV_fasta =

'TRAV.fasta'

TRAJ_fasta = 'TRAJ.fasta'

TRDV_fasta = 'TRDV.fasta'

TRDD_fasta = 'TRDD.fasta'

TRDJ_fasta = 'TRDJ.fasta'

TRGV_fasta = 'TRGV.fasta'

TRGJ_fasta = 'TRGJ.fasta'

EI

siguiente directorio y archivos no se empaquetarán con vdj

pero se calcularán la primera

vez refseq es importado. Después

de eso, no se volverá a calcular a menos que sea un directorio

#

de datos persistentes forzados

Pickle_dir = 'Pickle'

Registros de LIGM relevantes en formato de Pickle

a menos que sea forzado

IGHV_Pickle =

'IGHV.Pickle'

IGHD_Pickle = 'IGHD.Pickle'

IGHJ_Pickle = 'IGHJ.Pickle'

IGKV_Pickle =

'IGKV.Pickle'

IGKJ_Pickle = 'IGKJ.Pickle'

IGLV_Pickle = 'IGLV.Pickle'

```

IGLJ _Pickle =

'IGLJ.Pickle'
TRBV _Pickle = 'TRBV.Pickle'
TRBD _Pickle = 'TRBD.Pickle'
TRBJ _Pickle =

'TRBJ.Pickle'
TRAV _Pickle = 'TRAV.Pickle'
TRAJ _Pickle = 'TRAJ.Pickle'
TRDV _Pickle =

'TRDV.Pickle'
TRDD _Pickle = 'TRDD.Pickle'
TRDJ _Pickle = 'TRDJ.Pickle'
TRGV _Pickle =

'TRGV.Pickle'
TRGJ _Pickle = 'TRGJ.Pickle'

## directorio de la base de datos de archivosplanos de IMGT completo
# imgt_dir =

'/Usuarios/laserson/buscar/iglesia/vdj-ome/ref-datos/IMGT'
# ligm_nombredelarchivo = 'imgt.dat'

#

refdatosdir = '/Usuarios/laserson/buscar/iglesia/vdj-ome/ref-datos/IMGT'
# imgtdat =

'imgt.dat'
# imgtfasta = 'imgt.fasta'
# imgtvseq = 'vdj_ref.fasta'
# imgtrefseqfasta =

'imgtspec.vdxml'
# imgtrefseqFR3fincoords = 'imgtrefseqFR3fincoords.dat'
#

imgtrefseqJTRPiniciocoords = 'imgtrefseqJTRPiniciocoords.dat'
# funciones para canal

operaciones

importar vdj
importar vdj.agrupación
importar seqherramientas

def

iterador2partes(iterador,nombrebase,tamañodelpaquete,prefijo=' ',sufijo=' '):
    " " " Datos de división de

iterador dentro archivos múltiples" " "
    partes = []
    núm_procesado = 0
    archivo_núm = 1

```

```

curr_nombredesalida = nombrebase+'.'+str(archivo_núm)

para obj en iterador:
    si

núm_procesado == 0:
    op = abrir(curr_nombredesalida,'w')
    imprimir >>op, prefijo

    partes.append(curr_nombredesalida)

    imprimir >>op, cadena
    núm_procesado +=1

    si núm_procesado == tamañodepaquete:
        imprimir >>op, sufijo

op.cerrar()
núm_procesado =
archivo_núm +=1
curr_nombredesalida

= nombrebase+'.'+str(archivo_núm)
si no op.cerrado:
    imprimir >>op, sufijo

op.cerrar()

regresar a partes

def carga_códigos de barras(códigos de barras_archivo) :
    bcip =

abrir(códigos de barras_archivo,'r')
códigos de barras = { }
para (descr,seq) en

seqherramientas.FastaInterador(bcip) :
    códigos de barras[seq.superior()] = descr
    bcip.cerrar()

# verificar que los códigos de barras cumplan con los criterios necesarios
códigos de barras_len = len(códigos de barras.llaves()[0])

para bc en códigos de barras.llaves ():
    si len(bc) != códigos de barras_len:
        levantar Excepción

" ERROR: Todas las longitudes de los códigos de barras deben ser iguales."

regresar a códigos de barras

```



```

def
id_códigos de barras(cadena,códigos de barras) :
    # se supone que los códigos de barras son de una sola longitud
    códigos de barras_len =

len(códigos de barras.llaves ())[0]
    tratar:
        curr_códigos de barras =

códigos de barras[cadena.seq[:códigos de barras_len].superior()]
    excepto ErrordeLlave: # códigos de barras no encontrados; cadena

sin cambio
    regresar # cadena mantiene sin cambio
    cadena.seq =

cadena.seq[códigos de barras_len:] # reducir código de barras de seq
    cadena.códigos de barras = curr_códigos de barras

def
cargar_isotipos(isotipo_archivo):
    ighcip = abrir(isotipo_archivo):
    isotipos = { }
    para

(descr,seq) en seqherramientas.Fastalterador(ighcip):
    Isotipos[seq.superior()] = descr

ighcip.cerrar()

    regresar a isotipos

def id_idotipo(cadena,isotipos) :
    si no

cadena.tiene_etiqueta('positivo') y no cadena.tiene_etiqueta('codificación'):
    advertencias.advertir('cadenas' %s

pueden no ser la hebra correcta' % cadena.descr)

    para iso en isotipos.iterarticulos():
        si iso[0] en cadena.seq[-50:]: # corte arbitrario del final 3'

iso[1]

def gato_vdjxml(archivos,fuerademanejo):
    imprimir >>fuerademanejo, "<raíz>"
    para f en

archivos:
    enmanejo = abrir(f,'r')
    para cadena en vdj.análisis_VDXML(enmanejo):

        imprimir >>fuerademanejo,cadena
        imprimir >>fuerademanejo, "</raíz>"

```

```

def
división_VJ(enmanejo,nombrebase):
    # ignora los números de alelos
    def
vj_id_no_alelo(cadena):
    regresar seqherramientas.limpiar_id(cadena.v.división('*')[0]) + '_' +
seqherramientas.limpiar_id(cadena.j.división('*')[0]) + '_' +
    def nombredesalida(nombrebase,vj_id):
regresar a "%s.%s.vdjxml" % (nombrebase,vj_id)
    fuerademanejo = { }
    para cadena en
vdj.análisis_VDJXML(enmanejo):
    curr_vj_id = vj_id_no_alelo(cadena)
    tratar:
    imprimir >>fuerademanijos[curr_vj_id], cadena
    excepto ErrordeLlave:
fuerademanijos[curr_vj_id] = abrir( nombredesalida(nombrebase,curr_vj_id), 'w' )
    imprimir
>>fuerademanijos[curr_vj_id], "<raíz>"
    imprimir >>fuerademanijos[curr_vj_id], cadena
para fuerademanejo en fuerademanijos.itervalores():
    imprimir >>fuerademanijos, "</raíz>"
regresar [nombredesalida(nombrebase,vj_id) para vj_id en fuerademanejo.iterllaves()]
# =====
# =
Matplotlib =
# =====
# algunas herramientas para automatizar las manipulaciones MPL que realizoimportan
os
importar sis
importar tipos
importar cPickle como Pickle
importar params
importar refsequtils
#
=====
# = Inicializaciones por primera vez =

```

```

#
=====

# ¿existe el directorio de Pickle?
si no os.camino.existe(

os.camino.unir(params.vdj_dir,params.Pickle_dir ) :
    od.mkdir (

os.camino.unir(params.vdj_dir,params.encertido_dir ) )

# prueba para cada archivo de Pickle tipo gen
si

no
os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.IGHV_Pickle) ):

refsequtils.proceso_IMGT_referencias(refsequtils.EntradadeReferenciaV,.os.camino.unir(params.
vdj_dir,

params.datos_dir,params.IGHV_fasta),os.camino.unir(params.vdj_dir,params.Pickle_dir,par
ams.IGHV

_Pickle), verboso=Verdadero)
si no

os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.IGHD_Pickle) ):

refsequtils.proceso_IMGT_referencias(

refsequtils.EntradadeReferencia,os.camino.unir(params.vdj_dir,params.datos_dir,params.IGHD_f
asta),os

.camino.unir(params.vdj_dir,params.Pickle_dir,params.IGHD_Pickle), verboso=Verdadero)
si no

os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.IGHJ_Pickle) ):

refsequtils.proceso_IMGT_referencias(refsequtils.EntradadeReferenciaJ,os.camino.unir(params.
vdj_dir,

params.datos_dir,params.IGHJ_fasta),os.camino.unir(params.vdj_dir,params.Pickle_dir,par
ams.IGHJ

_Pickle), verboso=Verdadero)
si no

os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.IGKV_Pickle) ):

refsequtils.proceso_IMGT_referencias(refsequtils.EntradadeReferenciaV,os.camino.unir(params.
vdj_dir,

params.datos_dir,params.IGKV_fasta),os.camino.unir(params.vdj_dir,params.Pickle_dir,par
ams.IGKV

_Pickle), verboso=Verdadero)
si no

os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.IGKJ_Pickle) ):

refsequtils.proceso_IMGT_referencias(refsequtils.EntradadeReferenciaJ,os.camino.unir(params.
vdj_dir,

```

```

params.datos_dir,params.IGKJ_fasta),os.camino.unir(params.vdj_dir,params.Pickle_dir,par
ams.IGKJ

_Pickle), verboso=Verdadero)
si no

os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.IGLV_Pickle) ):

refsequutils.proceso_IMGT_referencias(refsequutils.EntradadeReferenciaV,os.camino.unir(params.
vdj_dir,

params.datos_dir,params.IGLV_fasta),os.camino.unir(params.vdj_dir,params.Pickle_dir,par
ams.IGLV

_Pickle), verboso=Verdadero)
si no

os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.IGLJ_Pickle) ):

refsequutils.proceso_IMGT_referencias(refsequutils.EntradadeReferenciaJ,os.camino.unir(params.
vdj_dir,

params.datos_dir,params.IGLJ_fasta),os.camino.unir(params.vdj_dir,params.Pickle_dir,par
ams.IGLJ

_Pickle), verboso=Verdadero)
si no

os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRBV_Pickle) ):

refsequutils.proceso_IMGT_referencias(refsequutils.EntradadeReferenciaV,os.camino.unir(params.
vdj_dir,

params.datos_dir,params.TRBV_fasta),os.camino.unir(params.vdj_dir,params.Pickle_dir,par
ams.TRBV

_Pickle), verboso=Verdadero)
si no

os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRBD_Pickle) ):

refsequutils.proceso_IMGT_referencias(

refsequutils.EntradadeReferencia,os.camino.unir(params.vdj_dir,params.datos_dir,params.TRBD_f
asta),os

.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRBD_Pickle), verboso=Verdadero)
si no

os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRBJ_Pickle) ):

refsequutils.proceso_IMGT_referencias(refsequutils.EntradadeReferenciaJ,os.camino.unir(params.
vdj_dir,

params.datos_dir,params.TRBJ_fasta),os.camino.unir(params.vdj_dir,params.Pickle_dir,par
ams.TRBJ

_Pickle), verboso=Verdadero)
si no

os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRAV_Pickle) ):

```

ES 2 730 951 T3

```
refsequutils.proceso_IMGT_referencias(refsequutils.EntradaReferenciaV,os.camino.unir(params.vdj_dir,
params.datos_dir,params.TRAV_fasta),os.camino.unir(params.vdj_dir,params.Pickle_dir,par
ams.TRAV
_Pickle), verboso=Verdadero)
si no
os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRAJ_Pickle) ):
refsequutils.proceso_IMGT_referencias(refsequutils.EntradaReferenciaJ,os.camino.unir(params.vdj_dir,
params.datos_dir,params.TRAJ_fasta),os.camino.unir(params.vdj_dir,params.Pickle_dir,par
ams.TRAJ
_Pickle), verboso=Verdadero)
si no
os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRDV_Pickle) ):
refsequutils.proceso_IMGT_referencias(refsequutils.EntradaReferenciaV,os.camino.unir(params.vdj_dir,
params.datos_dir,params.TRDV_fasta),os.camino.unir(params.vdj_dir,params.Pickle_dir,par
ams.TRDV
_Pickle), verboso=Verdadero)
si no
os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRDD_Pickle) ):
refsequutils.proceso_IMGT_referencias(
refsequutils.EntradaReferencia,os.camino.unir(params.vdj_dir,params.datos_dir,params.TRDD_f
asta),os
.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRDD_Pickle), verboso=Verdadero)
si no
os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRDJ_Pickle) ):
refsequutils.proceso_IMGT_referencias(refsequutils.EntradaReferenciaJ,os.camino.unir(params.vdj_dir,
params.datos_dir,params.TRDJ_fasta),os.camino.unir(params.vdj_dir,params.Pickle_dir,par
ams.TRDJ
_Pickle), verboso=Verdadero)
si no
os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRGV_Pickle) ):
refsequutils.proceso_IMGT_referencias(refsequutils.EntradaReferenciaV,os.camino.unir(params.vdj_dir,
params.datos_dir,params.TRGV_fasta),os.camino.unir(params.vdj_dir,params.Pickle_dir,par
ams.TRGV
_Pickle), verboso=Verdadero)
si no
```

ES 2 730 951 T3

```
os.camino.existe(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRGJ_Pickle) ):
refsequtils.proceso_IMGT_referencias(refsequtils.EntradaReferenciaJ,os.camino.unir(params.
vdj_dir,
params.datos_dir,params.TRGJ_fasta),os.camino.unir(params.vdj_dir,params.Pickle_dir,par
ams.TRGJ
_Pickle), verboso=Verdadero)
# en este punto, debe haber archivos de Pickle con procesado completamente
genes de referencia
# (incluyendo las partes dependientes de LIGM)

# =====
# = Cargar

datos de referencia =
# =====

IGHV =
Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.IGHV_Pickle) ))
IGHD =
Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.IGHD_Pickle) ))
IGHJ =
Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.IGHJ_Pickle) ))
IGKV =
Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.IGKV_Pickle) ))
IGKJ =
Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.IGKJ_Pickle) ))
IGLV =
Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.IGLV_Pickle) ))
IGLJ =
Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.IGLJ_Pickle) ))
# TRBV
=
Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRBV_Pickle) ))
#
TRBD =
Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRBD_Pickle) ))
# TRBJ
=
Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRBJ_Pickle) ))
#
TRAV =
Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRAV_Pickle) ))
# TRAJ
```

ES 2 730 951 T3

```
=
Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRAJ_Pickle) ))
#

TRDV =

Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRDD_Pickle) ))
# TRDD

=
Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRDD_Pickle) ))
#

TRDJ =

Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRDJ_Pickle) ))
# TRGV

=
Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRGV_Pickle) ))
#

TRGJ =

Pickle.cargar(abrir(os.camino.unir(params.vdj_dir,params.Pickle_dir,params.TRGJ_Pickle) ))
#

=====
# = Definir estructuras datos para

compatibilidad con alineador =
# =====

#

IGHn_lista -- lista de los identificadores de refseq (por ejemplo, IGHJ1, IGHV6-1, etc.)
# IGHn_seqs -- dict

donde las llaves son IDs de refseq y los valores son las secuencias actuales
#           de

refseq.
# IGHn_idx -- dict donde las llaves son IDs de refseq y valores son los indices dentro

la
#
           lista de IGHn_lista.
# IGHV_compensar

def legado_datos(ref_datos):

" " " Generar estructura de datos usados por herramientas aln etc. " " "
locus_lista = [elt.alelo para elt en

ref_datos]
locus_seqs = dict( [ (elt.alelo,elt.seq) para elt en ref_datos] )
locus_idx =

dict( [ (g,i) para i,g en enumerar(locus_lista) ] )
tratar:
```

ES 2 730 951 T3

```
locus_compensar =
dict( [ (elt.alelo,elt.cdr3_límite-elt.adhesión_coords[0]) para elt en ref_datos ] )

regresar (locus_lista,locus_seqs,locus_idx,locus_compensar)
    excepto ErrordeAtributo:

regresar (locus_lista,locus_seqs,locus_idx)

(IGHV_lista, IGHV_seqs, IGHV_idx, IGHV_compensar) =

legado_datos (IGHV)
(IGHD_lista, IGHV_seqs, IGHV_idx) = legado_datos (IGHD)
(IGHJ_lista, IGHJ_seqs, IGHJ_idx, IGHJ_compensar) = legado_datos (IGHJ)
(IGKV_lista, IGKV_seqs, IGKV_idx, IGKV_compensar) = legado_datos (IGKV)
(IGKJ_lista, IGKJ_seqs, IGKJ_idx, IGKJ_compensar) = legado_datos (IGKJ)
(IGLV_lista, IGLV_seqs, IGLV_idx, IGLV_compensar) = legado_datos (IGLV)
(IGLJ_lista, IGLJ_seqs, IGLJ_idx, IGLJ_compensar) = legado_datos (IGLJ)
#

(TRBV_lista, TRBV_seqs, TRBV_idx, TRBV_compensar) = legado_datos (TRBV)
#

(TRBD_lista, TRBD_seqs, TRBD_idx) = legado_datos (TRBD)
#

(TRBJ_lista, TRBJ_seqs, TRBJ_idx, TRBJ_compensar) = legado_datos (TRBJ)
#

(TRAV_lista, TRAV_seqs, TRAV_idx, TRAV_compensar) = legado_datos (TRAV)
#

(TRAJ_lista, TRAJ_seqs, TRAJ_idx, TRAJ_compensar) = legado_datos (TRAJ)
#

(TRDV_lista, TRDV_seqs, TRDV_idx, TRDV_compensar) = legado_datos (TRDV)
#

(TRDD_lista, TRDD_seqs, TRDD_idx) = legado_datos (TRDD)
#

(TRDJ_lista, TRDJ_seqs, TRDJ_idx, TRDJ_compensar) = legado_datos (TRDJ)
#

(TRGV_lista, TRGV_seqs, TRGV_idx, TRGV_compensar) = legado_datos (TRGV)
#

(TRGJ_lista, TRGJ_seqs, TRGJ_idx, TRGJ_compensar) = legado_datos (TRGJ)

## refseq.py
#
#
# " " " Contiene datos y funciones para tratar con datos de referencia IMGT.
#
# refdatosdir debe establecerse
```


ES 2 730 951 T3

```
en el directorio donde la liberación del archivo plano IMGT
# está presente. Esto debe incluir

imgt.dat, imgt.fasta, e imgtrefseq.fasta.
#
# IGHn_lista -- lista de los identificadores de refseq

(por ejemplo, IGHJ1, IGHV6-1, etc.)
# IGHn_acc -- dict donde las llaves son IDs de refseq y los valores son los
números de adhesión de IMGT
# IGHn_coords -- dict donde las llaves son IDs de refseq y los valores son pares
de coords que son
#          extraídos de la base de datos de refseq. Estos números son
coords de seq
#          no modificadas del elemento de referencia dentro de la entrada LIGM
#

IGHn_seqs -- dict donde las llaves son IDs de refseq y los valores son las secuencias
actuales
#
#          de refseq.
# IGHn_idx -- dict donde las llaves son IDs de refseq y los valores
son los índices dentro de la
#          lista de IGHn_lista.
# IGHV_FR3_IMGT_fin_coord -- dict
donde las llaves son IDs de refseq y los valores son la coord del
#

fin de FR3-IMGT (incl el 2do-CYS) no modificado
# IGHJ_J_TRP_inicio_coord -- dict donde las
llaves son IDs de refseq y los valores son la
#          coord del
inicio del sitio J-TRP no modificado
#
# " " "
#
# importar os
# importar cPickle
#
# de Bio

importar SeqIO
#
# importar vdj
# importar params
# importar sequtils
#
# #

=====
## FNs UTILIDAD para análisis de base de datos

de referencia =
## =====
#
```

```

# def

obtener_refseq_elementos(locus,alelos,func,especies,obtenercoords,refdatosdir,imgtrefseqfasta)
:
#

" " " Extraer todos los identificadores que cumplen ciertos criterios de IMGT/GEN-DB (refseq).
#
#

locus -- identificador como 'IGH', 'TBV', etc
# alelos -- pueden ser cualquiera de los dos '01' o

'todos'
# func -- lista de IDs de func, por ejemplo, ['F', 'ORF', 'P']. Si vacío, incluye

todas
# especies -- especies de animal (por ejemplo, 'Homo+sapiens')
# obtenercoords --

determina si procesar y regresar coords (boolean)
#
# refdatosdir --

directorio donde reside IMGT
# imgtrefseqfasta -- nombredelarchivo del archivo imgtrefseqfasta
#

# " " "
#
# reflDs = [ ]
# refacc = { }
# refseqs = { }
# si

obtenercoords:
# refcoords = { }
# todasrefseqs = sequitils.cargar_fasta(

os.camino.unir(refdatosdir,imgtrefseqfasta) )
#
# si func == [ ]:
# func =

['F', 'ORF', 'P']
#
# para seq en todasrefseqs:
# # obtener info en seq curr
#

currdescr = seq.descripcion
# currlocus = currdescr.división(' ')[0].división('* ')[0]
#

curralelo = currdescr.división(' ')[0].división('* ')[1]
# currspecies =

currdescr.división(' ')[1]
# currfun = currdescr.división(' ')[2].división(' ')[-1].banda('

(' ')
# currID = currdescr.división(' ')[0].lbanda(' > ')

```

```

#         curracc =

currdescr.división(' , ') [3]
#         si obtenercoords:
#         tratar:
#

currcoords = ( eval (currdescr.división(' , ')[4].división(' . ')[0] ),

eval (currdescr.división(' , ')[4].división(' . ')[2] )
#         excepto:
#         #imprimir

"Coordenadas para", currlocus, curracc, "no son interpretables"
#         pasar
#

#         # realizar pruebas para excluir seq actuales
#         si locus no en currlocus:
#

#         continuar

#         si alelos != 'todos' :
#         si curralelo !=

alelos:
#         continuar
#
#         si currepecies != especies:
#

#         continuar

#         si currfunc no en func:
#         continuar
#

#         # verificar para "parcial"
#         si len(currdescr.división('.') ) > 5 y

'parcial' en '.unir(currdescr.división('.') [5:] ) :
#         continuar
#

refIDs.append( currID )
#         refacc [currID] = curracc
#         refseqs [currID] =

seq.seq.encadenar().superior()
#         si obtenercoords:
#         refcoords [currID] =

currcoords
#
#         refIDs.append( ' ' )
#         refIDs.clasificar ( )
#
#         si no

```

```

obtenercoords:
#         regresar a refIDs,refacc,refseqs
#         regresar a

refIDs,refacc,refseqs,refcords
#
#
#
## =====
## =

Definición de datos de referencia =
## =====
#
# LOCI =

['IGH', 'IGK', 'IGL', 'TRA', 'TRB', 'TRD', 'TRG']
#
# IGHV_lista,IGHV_acc,IGHV_seqs,IGHV_coords =

obtener_refseq_elementos(locus='IGHV',alelos='01',func=['F', 'ORF'],especies='Homo+sapiens',obtenercoo

rds=Verdadero,refdatosdir=params.refdatosdir,imgtrefseqfasta=params.imgtrefseqfasta)
#

IGHD_lista,IGHD_acc,IGHD_seqs =

obtener_refseq_elementos(locus='IGHD',alelos='01',func=['F', 'ORF'],especies='Homo+sapiens',obtenercoo

rds=False,refdatosdir=params.refdatosdir,imgtrefseqfasta=params.imgtrefseqfasta)
#

IGHJ_lista,IGHJ_acc,IGHJ_seqs,IGHJ_coords =

obtener_refseq_elementos(locus='IGHJ',alelos='01',func=['F', 'ORF'],especies='Homo+sapiens',obtenercoo

rds=Verdadero,refdatosdir=params.refdatosdir,imgtrefseqfasta=params.imgtrefseqfasta)
#
# IGHV_idx

= dict( [ (g,i) para i,g en enumerar(IGHV_lista) ] )
# IGHV_idx = dict( [ (g,i) para i,g en

enumerar(IGHD_lista) ] )
# IGHJ_idx = dict( [ (g,i) para i,g en enumerar(IGHJ_lista) ] )
#
#

IGHC_lista = [
#
#         'IGHA1',
#         'IGHA2',
#

'IGHD',
#         'IGHE',
#         'IGHG1',
#         'IGHG2',
#

```

```

'IGHG3',
#       'IGHG4',
#       'IGHM',
#       ]
# IGHC_idx =

dict( [ (g,i) para i,g en enumerar(IGHC_lista) ] )
#
# TODOS_IDs = lista(establecer(IGHV_lista + IGHD_lista
+ IGHJ_lista + IGHC_lista )
# TODOS_IDs.clasificar ()
#
#
# # =====
# # =

Datos de especificidad =
# # =====
#
# def

obtener_LIGM_con_especificidades(refdatosdir,imgtdat,imgtfasta,fastadesalida,vdjxmldesalida) :
#
...
#       Tomar los archivos planos de IMGT LIGM y fasta, y regresar un fasta con solo esas
seqs
#
#       que tienen una especificidad asociada con ellos en un archivo fasta. El encabezado
contiene
#
#
#       la accesión y la especificidad.
#
#
#       vdj.refseq.obtener_LIGM_con_especificidades("imgt.dat","imgt.fasta","imgt.especificidades.f
asta","i
mgt.especificidades.vdjxml")
#       '...'
#       especificidades = { }      # lista de 2- tuplas
(ACCESIÓN,especificidad)
#       LIGMplano = abrir(os.camino.unir(refdatosdir,imgtdat), 'r')
#
LIGMfasta = abrir(os.camino.unir(refdatosdir,imgtdat), 'r')
#       opEspecificidadfasta =

abrir(os.camino.unir(refdatosdir,fastadesalida), 'w')
#
#       númRegistro = 0
#
#
#       númArchivoconSpec =
#
#       ID = ' '
#       DE = ' '
#       para línea en LIGMplano:

```

```

#
#
#   líneadivisión = línea.división ( )
#
#   si líneadivisión[0] == 'ID':
#
enRegistro = Verdadero
#           ID = líneadivisión[1]
#           númRegistros += 1
#   elif
líneadivisión[0] == 'DE':
#           enRegistro:
#           DE += '
'.unir(líneadivisión[1:]) + ''
#   elif líneadivisión[0] == 'XX' :
#   si DE == ' ':
#
# si no ha almacenado la descripción todavía
#           continúe
#           de otro modo:
# registro finalizado
#           si 'especificidad' en DE :
#           specidx =
#
DE.rencontrar('especificidad')
#           spec =
DE[specidx+len('especificidad') : ] .banda().rbanda (' . ')
#           si
spec.iniciacon('anti') :
#           especificidadesID] = spec
#
#           númRegistroconSpec +=1
#           ID = ''
#           DE = ''
#
#           enRegistro = Falso
#   imprimir "Número de lecturas del registro de LIGM:" +
str(númRegistrosconSpec)
#
#   númFatsa = 0
#
#   para seq en
SeqIO.análisis(LIGMfasta, 'fasta') :
#           spec = especificidades.obtener(seq.id, ' ')
#           si
spec == ' ':
#           continuar
#           de otro modo:

```

```

#           imprimir >>opEspecificidadfasta,

">" + seq.id + " | " + spec
#           imprimir >>opEspecificidadfasta,

seq.seq.abanda().superior()
#           númFasta += 1
#
#   imprimir "Número de registros

Fasta con especificidades encontradas e impresas: " + str(númFasta)
#
#

LIGMplano.cerrar()
#   LIGMfasta.cerrar()
#   opEspecificidadfasta.cerrar()
#
#   #

VDJXML y alineamiento
#   rep =

vdj.inicial_importar( [os.camino.unir(refdatosdir,fastadesalida) ], os.camino.unir(refdatosdir,vd
jxmldesali

da).etiquetasmeta=[especificidad_referencia : '+vdj.tiempostamp() ] )
#   rep =

vdj.positivo_banda(rep)
#   rep = vdj.align_rep(rep)
#
#   #refiltrada =

rep.obtener_cadenas_VJcompleta()
#   repfiltrada = rep
#   para cadena en repfiltrada:
#

spec = especificidades.obtener(cadena.descr, '' )
#   si spec == '' :
#   imprimir

"Cadena de referencia tiene especificidad vacía: " + cadena.descr
#   continuar
#

de otro modo:
#   cadena.adicionar_etiquetas( 'especificidad|' + spec )
#
#

vdj.escribirVDJ(repfiltrada,os.camino.unir(refdatosdir,vdjxmldesalida) )
#
#   regresar
#
#
#

si no os.camino.existe(os.camino.unir(params.refdatosdir,params.imgtspecfasta ) ) o no
os.camino.existe(os.camino.unir(params.refdatosdir,params.imgtspecvdjxml ) )
#

```

```

obtener_LIGM_con_especificidades(params.refdatosdir,params.imgtdat,params.imgtfasta,params
.imgtspe

cfasta,paramsvdjxml) :
#
#
# si

os.camino.existe(os.camino.unir(params.refdatosdir,params.imgtspecfasta ) ) y

os.camino.existe(os.camino.unir(params.refdatosdir,params.imgtspecvdjxml ) ):
#     ipspectfasta =

abrir(os.camino.unir(params.refdatosdir,params.imgtspecfasta), 'r' )
#
#     SPEC_lista =

establecer( )
#     para línea en ipsepcfasta :
#         si línea[0] == '>':
#             currspec =

' l '.unir(línea.división('l')[1:].banda()
#             SPECT_lista.adicionar(currspec)
#

ipspectfasta.cerar( )
#     SPECT_lista.adicionar(' ')
#     SPEC_lista = lista(SPECT_lista)
#

SPEC_lista.clasificar( )
importar sis
importar banda
de cBandaO importar BandaO
importar cPickle como

Pickle
importar urllib2

importar advertencias # esto es porque IMGT tiene muchos errores

importar

FormulariodeCliente
de Bio importar SeqIO

importar seqherramientas
importar params

identidad =

banda.hacetrans(",")

# =====
# = Estructuras de datos =
#
# =====

clase de EntradadeReferencia(objeto):
    " " Estructura de datos para mantener una secuencia de

```


de referencia de IMGT/GEN-DB o
 IMGT/V-BÚSQUEDA.

Algunos de los atributos son

computados, y algunos son tomados de IMGT.

accesión_coordss utiliza numeración de python

convertida

diferenciada_seq incluye las diferencias de IMGT que proporcionan
 marcos de usos de coords python:

0 significa ya en el marco.

1 significa omitir un nt

2 significa omitir dos nts
 parciales tomará valores de "parcial en 5' ", "parcial

en 3' ", o "parcial en 5' y 3' "
 dependiendo donde esté la del.
 ""

def

```
__init__(auto,**kw):
    def kw_init(attrib):
        si kw.tiene_llave(attrib):

    auto.__establecerattr__(attrib,kw[attrib])

    kw_init('accesión')
```

```
kw_init('seq')
    kw_init('diferenciada_seq')
    kw_init('descripción')
```

```
kw_init('locus')
    kw_init('gen')
    kw_init('alelo')
```

```
kw_init('especies')
    kw_init('funcional')
    kw_init('imgt_etiqueta')
```

```
kw_init('accesión_coords')
    kw_init('longitud')
    kw_init('marco')
```

```
kw_init('parcial')
```

```
def init_de_imgt(auto,fasta_encabezado,seq):
    "" "Inicializar
```

objeto del encabezado fasta IMGT y seq" " "

```

datos =

fasta_encabezado.lbanda('>').rbanda().división('1')
    auto.acesión = datos [0]

auto.diferenciada_seq = seq
    # auto.seq = seq.traducir(Ninguno, '.').superior()    # eliminar períodos

(diferencias) # python >2.5
    auto.seq = seq.traducir(identidad, '.').superior()

auto.descripcion = datos[1]
    auto.locus = auto.descripcion[0:4]
    auto.gen =

auto.descripcion.división("**")[0]
    auto.alelo = auto.descripcion
    auto.especied =

datos[2]
    auto.funcional = datos[3]
    auto.imgt_etiqueta = datos [4]

crudo_coords_inicio = int(datos[5].división('.')[0])
    crudo_coords_fin =

int(datos[5].división('.')[1])
    coords = (crudo_coords_inicio - 1, crudo_coords_fin) # notar

la conversión al sistema de coord python
    auto.acesión_coords = coords

auto.longitud = len(auto.seq)
    si auto.longitud != int(datos[5].división(' ')[0]):

levantar ErrorValor, "Longitudes son inconsistentes: %s" % fasta_encabezado
    auto.marco =

int(datos[7]) - 1 # notar cambio a numeración python (0-basado)
    auto.parcial =

datos[13]

def PULL_LIGM_registro(auto) :
    " " "Obtener objeto de SeqRegistro para LIGM
registro del servidor IMGT" " "

    # NOTA: esto puede potencialmente
simplificarse significativamente al acceder al URL
    # interface a LIGM, a través de:
    #

http://imgt.cines.fr/cgi-bin/IMGTelect.jv?query=5+numacc
    # donde numacc es el

número de accesión

```

```

solicitud =
urllib2.Solicitud('http://imgt.cines.fr/cgi-bin/IMGTlect.jv?livret=0')
# página LIGM

respuesta = urllib2.urlabrir(solicitud)
formas = FormulariodeCliente.RespuestadeAnálisis(respuesta,

forma_analizado_clase=FormulariodeCliente.XHTMLAnlizadordeFormaCompatible,

    hacia atrás_compat=Falso)
    forma = formas[1]
    forma['I01p01c02'] =

auto.acesión
    solicitud2 = forma.hacer clic( )
    # página del formato de datos
    respuesta2

= urllib2.urlabrir(solicitud2)
    formas2 = FormulariodeCliente.RespuestadeAnálisis(respuesta2,

        forma_analizado_clase=FormulariodeCliente.XHTMLAnlizadordeFormaCompatible,

            hacia atrás_compat=Falso)
    forma = formas2[0]

afirmar( forma2.controla[8].attrs['valor'] == 'archivo plano de 2 IMGT' )

forma2.controla[8].id = 'archivoplano'
solicitud3 = forma2.hacer clic(id='archivoplano')
#

resultados del registro de LIGM
    respuesta3 = urllib2.urlabrir(solicitud3)

    # gueto

análisis de los resultados. el texto del registro de LIGM está en <pre>...</pre> etiquetas

datos1crudos = respuesta3.lectura()
datos2crudos =

datos1crudos.división('<pre>') [1].división('</pre>') [0].lbanda( )
datos3crudos =

BandaO(datos2crudos)
    auto.registro = SeqIO.lectura(datos3crudos, 'imgt')

clase

EntradadeRefereciaV(Entradadereferencia):

```

```

def __init__(auto,**kw):
EntradadeReferencia.__init__(auto,**kw)

def establecer_CDR3_límite(auto):      # Fin de FR3
    """ "Obtener coord del fin de FR3 de la base de datos de IMGT LIGM." """
    # algunos registros pueden tener

referencias múltiples en ellos
    objetivo_alelo = auto.alelo
    característica_iter =

auto.registro.características.__iter__( )
    v_gen =

seqherramientas.avanzar_a_características(característica_iter, ['V-REGIÓN', 'V-GEN'])
    mientras

v_gen.calificadores.obtener( 'alelo', [ ' ' ] [0] != objetivo_alelo: # avanzar al gen
objetivo

    v_gen = seqherramientas.avanzar_a_características(característica_iter, ['V-REGIÓN', 'V-GEN'])

conservado_cys = seqherramientas.avanzar_a_características(característica_iter, '2do-CYS')

    # nota:

características biopython ya utilizan indexación de python
    auto.cdr3_límite =

conservado_cys.localización.inicio.posición

clase EntradadeReferenciaJ(Entradadereferencia):
    def

    __init__(auto,**kw):
        EntradadeReferencia.__init__(auto,**kw)

    def

establecer_CDR3_límite(auto):      # Inicio de FR4
    """ "Obtener coord del inicio de FR4 de la base de datos

de IMGT LIGM." """
    # algunos registros pueden tener referencias múltiples en ellos

objetivo_alelo = auto.alelo
    característica_iter = auto.registro.características.__iter__( )

j_gen = seqherramientas.avanzar_a_características(característica_iter, ['J-REGIÓN', 'J-GEN'])
    mientras

j_gen.calificadores.obtener( 'alelo', [ ' ' ] [0] != objetivo_alelo: # avanzar al gen
objetivo

```

ES 2 730 951 T3

```
j_gen = seqherramientas.avanzar_a_características(característica_iter, ['J-REGIÓN', 'J-GEN'])
#
nota, pude haber un TRP o PHE conservado
conservado_trp =

seqherramientas.avanzar_a_características(característica_iter,['J-TRP', 'J-PHE'])

# nota:

características biopython ya utilizan indexación de python
auto.cdr3_límite =

conservado_trp.localización.inicio.posición

# =====
# = Analizando IMGT =
#
=====

# importar pdb

def

proceso_IMGT_referencias(ref_entrada_cls,fasta_ennombredearchivo,Pickle_fueradenombredearchivo,verboso=False):

" " " Referencias de carga de las refs del archivo fasta de IMGT/V-BÚSQUEDA

por ejemplo, IGHV.fasta, perente

en el directorio de datos
ref_entrada_cls es el objeto de clase para el tipo de referencia,

por ejemplo, ReferenciadeEntradaV,ReferenciadeEntradaJ, etc.
" " "

# pdb.establecer_traza()

referencias = []
ip = abrir(fasta_ennombredearchivo, 'r')
para registro en

SeqIO.análisis(ip, 'fasta')
curr_refeencia = ref_entrada_cls( )
curr_encabezado =

registro.descripcion
curr_seq = registro.seq.abanda( )

# Problemas

potenciales con encabezados FASTA
tratar:

curr_referencia.init_de_imgt(curr_encabezado,curr_seq)
```

```

    excepto ErrorValor:

advertencias.advertir('Encabezado inválido: %s' % curr_encabezado)
    continuar

    # No

quiero lidiar con las seqs parciales en este momento
    si 'parcial' en

curr_referencia.parcial:
    continuar

curr_referencia.PULL_LIGM_registro( )
    # Problemas potenciales para encontrar el límite de CDR3

anotado
    tratar:
        curr_referencia-establecer_CDR3_límite( )
    excepto

ErrordeAtributo:
    # EntradadeReferencia no tiene establecer_CDR3_límite. Usado para pasar

    segmentos D
    excepto ErrorValor, e:
        advertencias.advertir("no se pudo encontrar el

límite de CDR3 en %s. Omitiendo... " % curr_referencia.alelo)
    continuar

referencias.append(curr_referencia)
    si verboso: imprimir "Procesamiento acabado %s" %

curr_referencia.alelo
    sis.stdafuera.color( )
    ip.cerrar( )

    op =

abrir(Pickle_fueradenombredearchivo, 'w')
    Pickle.basura(referencias,op,protocolo=2)
    op.cerrar( )

    regresar a referencias
# configuración.py
de configuración de importación de número.distutils.núcleo, Extensión
de

número.distutils.misc_util importar obtener_número_incluir_dirs

núcleodealineamientoext = Extensión(
    "núcleodealineamiento",

```

```

        ["núcleodealineamiento.c"],
        incluir_dirs = obtener_número_incluye_dirs( )
    )
núcleodeagrupamientoext = Extensión(
    "núcleodeagrupamiento",
    ["núcleodeagrupamiento.c"],
        incluir_dirs = obtener_número_incluye_dirs( )
    )
configuración( nombre = "vdj"
                versión = "0.1",
                ext_módulos =
[núcleodealineamientoext, núcleodeagrupamientoext]
)

```

Un montón de guiones para realizar operaciones vdj.

fasta2vdjxml.py Convertir archivo fasta a vdjxml. Toma primero espacios en blanco
delim

campo para descr
tamaño_seleccionar.py Lecturas de selección de tamaño
vdjxml2partes.py

División del archivo vdjxml en partes
código de barras_id.py Anotar código de barras en secuencias
codificación_hebra.py Convertir cadenas a secuencia de codificación
isotipo_id.py

Anotar isotipos
Align_vdj.py Realizar clasificación de vdj
cat_vdjxml.py

operación cat en archivos vdjxml (maneja el elemento de raíz)
filtro_VJ.py

Seleccionar solo cadenas con alineamientos V y J
clúster_cdr3.py Realiza jerarquía

agrupamiento de CadenasInmunes utilizando sus uniones
partición_VJ.py Divisiones

vdjxml en archivos por combo VJ
actualizar_vdjxml.py Actualizar vdjxml de la versión anterior

a una versión más reciente

Generación anterior:

clúster_división_VJ.py
clúster_división_VJ_LSF.py
filtro_etiquetas_y.py
filtro_etiquetas_no.py
filtro_etiquetas_o.py
división_en_etiquetas.py
etiqueta_cadenas.py

```

vdj_completo_canal_LSF.py
vdjxml2clon_recuentos.py
vdjxml2fasta.py
#!/usr/bin/env python

importar sis
importar optanálisis

importar vdj
importar vdj.alineamiento

analizador =

optanálisis.AnalizadordeOpción( )
analizador.adicionar_opción('-L', '--locus', acción='append',dest='loci')
(opciones, args) = analizador.análisis_args( )

si len(args) == 2:
    enmanejo = abrir(args[0], 'r')

    fuerademanejo = abrir(args[1], 'w')
elif len(args) == 1:
    enmanejo = abrir(args[0], 'r')

fuerademanejo = sis.stdafuera
elif len(args) == 0:
    enmanejo = sis.stdadentro
    fuerademanejo =

sis.stafuera
de otro modo:
    levantar Excepción, "Número equivocado de argumentos."

alineador =

vdj.alineamiento.vdj_alineador_combinado(loci=opciones.loci)
imprimir >>fuerademanejo, "<raíz>"
para cadena

en vdj.análisis_VDJXML(enmanejo):
    alineador.align_cadena(cadena)
    imprimir >>fuerademanejo, cadena
imprimir >>fuerademanejo, "<raíz>"
#!/usr/bin/env Python

importar sis
importar optanálisis

importar

seqherramientas

importar vdj
importar vdj.canal

analizador = optanálisis.AnalizadordeOpción( )
analizador.adicionar_opción('-b', '--códigos de barras',dest='códigos de barras_fasta')
(opciones, args) =

analizador.análisis_args( )

```



```

si len(args) == 2:
    enmanejo = abrir(args[0], 'r')
    fuerademanejo =

abrir(args[1], 'w')
elif len(args) == 1:
    enmanejo = abrir(args[0], 'r')
    fuerademanejo =

sis.stdafuera
elif len(args) == 0:
    enmanejo = sis.stdadentro
    fuerademanejo =sis.stdafuera

# NOTA

todos los códigos de barras deben ser la misma longitud

códigos de barras =

vdj.canal.cargar_códigos de barras(opciones.códigos de barras_fasta)

# iterar a través de cadenas
imprimir

>>fuerademanejo, "<raíz>"
para cadena en vdj.análisis_VDJXML(enmanejo):

vdj.canal.id_códigos de barras(cadena,códigos de barras)
imprimir >>fuerademanejo, cadena
imprimir >>fuerademanejo,

"<raíz>"
#!/usr/bin/env python

importar sis
importar optanálisis
importar glob

importar

vdj.canal

analizador = optanálisis.AnalizadordeOpción( )
(opciones, args) = analizador.análisis_args ()

archivos

= [ ]
para arg en args:
    archivos.extend(glob.glob(arg) )

vdj.canal.cat_vdxml(archivos,sis.stdafuera)
#!/usr/bin/env python

importar sis
importar

optanálisis

importar vdj
importar vdj.agruamiento

```

```

analizador = opanalizador.AnalizadordeOpción ()
analizador.adicionar_opción('-c', '--cortar, tipo= 'flotar', defecto=4.5)
analizador.adicionar_opción('-t', '--etiqueta', defecto= ' ')
analizador.adicionar_opción('-l', '--enlace, tipo= 'elección', elecciones=['único', 'completo'] defe
cto=único')

(opciones, args) = analizador.análisis_args( )

si len(args) == 2:
    enmanejo =

abrir(args[0], 'r')
    fuerademanejo = abrir(args[1], 'w')
elif len(args) == 1:
    enmanejo =

abrir(args[0], 'r')
    fuerademanejo = sis.stdafuera
elif len(args) == 0:
    enmanejo = sis.stdadentro

    fuerademanejo =sis.stafuera

# NOTA: este guion requiere que haya una secuencia de unión
#
# bien definida. Se levanta una excepción si no es así. Por lo tanto, seqs
# deben ser
#
pre-filtradas para tener uniones legítimas
# NOTA: esta guion debe contener todas las cadenas en la memoria
#
para poder
# realizar el agrupamiento y asignar los nombres de clúster

# datos de carga
cadenas = []
uniones = []
para cadena en vdj.análisis_VDJXML(en manejo):
    # comprobar la presencia

de V, J y unión no trivial
    si no tieneattr(cadena,'v') o no tieneattr(cadena,'j') o

no tieneattr(cadena, 'unión') :
    levantar ErrorValor, "% de cadena no tiene unión de V-J

aln." % de cadehan.descr
    cadenas.append(cadena)
    uniones.append(cadena.unión)

# realizar

el agrupamiento de secuencia
(T,seq_idx) =

vdj.agrapamiento.clúster_seqs(uniones,opciones.cortar,opciones.enlace)

# cadenas de etiqueta con
IDs de clúster únicos
si opciones.etiqueta == ' ':

```

```

    etiqueta = ''
de otro modo:
    etiqueta = opciones.etiqueta+ 'l'

imprimir

>>fuera demanejo, "<raiz>"
para (i,cadena), en enumerar(cadenas):
    clonID = '%s%s' %

(etiqueta,T[seq_idx[cadena.unión] ] )
    cadena.clon = clonID
imprimir >>fuera demanejo, "<raiz>"
#!/usr/bin/env python

importar sis
importar optanálisis
importar

subproceso
importar tiempo

importar vdj

analizador = optanálisis.AnalizadordeOpción( )
analizador.adicionar_opción('-c', '--cortar, defecto=4.5, tipo= 'flotar')
(opciones, args) =

analizador.análisis_args ()

si len(args) == 2:
    enmanejo = abrir(args[0], 'r')
    fueradenombre =

args[0]
    fuera demanejo = abrir(args[1], 'w')
elif len(args) == 1:
    enmanejo =

abrir(args[0], 'r')
    fueradenombre = args[0]
    fuera demanejo = sis.stdafuera
elif len(args) == 0:

enmanejo = sis.stdadentro
    fueradenombre = 'VJ_partes.vdxml'
    fuera demanejo =sis.stafuera

imprimir

>>sis.stderr, "NOTA: las cadenas deben ser filtradas para un VJ aln y uniones válidas ANTES del
agrupamiento."

(VJ_partes,VJ_IDs) = vdj.división_vdxml_dentro_VJ_partes(enmanejo,fueradenombre)

VJ_partes_agrupadas = [ ]
procesos = [ ]
para (vj_archivo,vj_id) en zip(VJ_partes,VJ_IDs):

vj_archivo_agrupado = vj_archivo + '.agrupado'
    VJ_partes_agrupadas.append(vj_archivo_agrupado)

```

```

    params = { 'cortar':opciones.cortar,
              'etiqueta': vj_id

'archivodeentrada': vj_archivo
              'fueraearchivo': vj_archivo_agrupado}
    clúster_cdm = r'python

clúster_cdr3 --cortar %(cortar)f --etiqueta %(etiqueta)s %(archivodeentrada)s %(fueraearchivo)s' % params
    p =

subproceso.Pabrir(clúster_cdm,Shell =Verdadero
    procesos.append(p)
    # realizar en serie:

# vj.clúster_cadenas(opciones.cortar, vj_id, vj_archivo, vj_archivo_agrupado)

vdj.esperar_para_subprocesos(procesos)

para cadena en

vdj.análisis_VDJXML_partes(VJ_partes_agrupado):
    imprimir >>fueraedemanejo, cadena #! /usr/bin/env

python

importar sis
importar optanálisis
importar subprocesso
importar os
importar archivodetiempo

importar

vdj

analizador = optanálisis.AnalizadordeOpción( )
analizador.adicionar_opción('-c', '--cortar, defecto=4.5, tipo= 'flotar')
analizador.adicionar_opción('-l', '--enlace, tipo= 'elección', elecciones=['único', 'completo'] defe
cto=úni

co')
analizador.adicionar_opción('-q', '--cola)
analizador.adicionar_opción('-o', '--LSFsalida')
(opciones, args)

= analizador.análisis_args ()

si len(args) == 2:
    enmanejo = abrir(args[0], 'r')
    fueradenombre =

args[0]
    tempdirnombre = args[1]
    fuerademanejo = abrir(args[1], 'w')
elif len(args) == 1:

```

```

enmanejo = abrir(args[0], 'r')
    fueradenombre = args[0]
    tempdirnombre = fueradenombre
    fuerademanejo =

sis.stdafuera
elif len(args) == 0:
    enmanejo = sis.stdadentro
    fueradenombre = 'VJ_partes.vdxml'

tempdirnombre = fueradenombre
    fuerademanejo =sis.stdafuera

imprimir >>sis.stderr, "NOTA: las cadenas deben ser
filtradas para un VJ aln y uniones válidas ANTES del agrupamiento."

# directorio temporal para deshacerse de

archivos intermedios
tempdir =

temparchivo.mkdtemp(prefijo=tempdirnombre+ '.intermedio.', dir = '.')

(VJ_partes,VJ_IDs) =

vdj.división_vdxml_dentro_VJ_partes(enmanejo,os.camino.unir(tempdir,fueradenombre)

VJ_partes_agrupadas =

[]
trabajos = []
para (vj_archivo,vj_id) en zip(VJ_partes,VJ_IDs):
    vj_archivo_agrupado =

vj_archivo + '.agrupado' # NOTA: vj_archivo ya tiene prefijo de directorio

VJ_partes_agrupadas.append(vj_archivo_agrupado)
    params = { 'cortar':opciones.cortar,

        'enlace':opciones.enlace,
            'etiqueta' : vj_id

'archivodeentrada' : vj_archivo
    'fueraearchivo': vj_archivo_agrupado}
    clúster_cdm =

r'clúster_cdr3.py --cortar %(cortar)f --etiqueta %(etiqueta)s --enlace %(enlace)s %(archivodeentrada)s

%(fueraearchivo)s' % params
    si os.stat(vj_archivo).st_tamaño < 8e6:
        trabajoID =

vdj.LSF.enviar_a_LSF(opciones.cola,opciones.LSFsalida,clúster_cdm)
    de otro modo:
        trabajoID =

```

```

vdj.LSF.enviar_a_LSF(opciones.cola,opciones.LSFsalida,clúster_cdm,mem_uso=4096)

trabajos.append(trabajoID)

vdj.esperar_para_LSF_trabajos(trabajos)

para cadena en
vdj.análisis_VDJXML_partes(VJ_partes_agrupado):
    imprimir >>fuera demanejo, cadena
#! /usr/bin/env

python

importar sis
importar optanálisis

importar vdj
importar vdj.alineamiento

analizador =

optanálisis.AnalizadordeOpción( )
analizador.adicionar_opción('-L', '--locus',acción= 'append',dest= 'loci')
(opciones, args) = analizador.análisis_args ()

si len(args) == 2:
    enmanejo = abrir(args[0], 'r')

    fuera demanejo = abrir(args[1], 'w')
elif len(args) == 1:
    enmanejo = abrir(args[0], 'r')

fuera demanejo = sis.stdafuera
elif len(args) == 0:
    enmanejo = sis.stdadentro
    fuera demanejo =

sis.stafuera

alineador = vdj.alineamiento.vdj_alineador_combinado(loci=opciones.loci)
imprimir

>>fuera demanejo, "<raíz>"
para cadena en vdj.análisis_VDJXML(enmanejo) :

alineador.codificando_cadena(cadena)
    imprimir >>fuera demanejo, cadena
imprimir >>fuera demanejo, "<raíz>"
#!

/usr/bin/env python

importar sis
importar optanálisis

importar seqherramientas

```

```

importar vdj

analizador =

optanálisis.AnalizadordeOpción( )
(opciones, args) = analizador.análisis_args ( )

si len(args) == 2:

enmanejo = abrir(args[0], 'r')
    fuerademanejo = abrir(args[1], 'w')
elif len(args) == 1:

enmanejo = abrir(args[0], 'r')
    fuerademanejo = sis.stdafuera
elif len(args) == 0:
    enmanejo =

sis.stdadentro
    fuerademanejo =sis.stafuera

vdj.canal.fasta2vdjxml(enmanejo,fuerademanejo):
#!

/usr/bin/env python

importar sis
importar optanálisis

importar vdj

analizador =

optanálisis.AnalizadordeOpción( )
(opciones, args) = analizador.análisis_args ( )

si len(args) == 2:

enmanejo = abrir(args[0], 'r')
    fuerademanejo = abrir(args[1], 'w')
elif len(args) == 1:

enmanejo = abrir(args[0], 'r')
    fuerademanejo = sis.stdafuera
elif len(args) == 0:
    enmanejo =

sis.stdadentro
    fuerademanejo =sis.stafuera

imprimir >>fuerademanejo, '<raíz>'
para cadena en

vdj.análisis_VDJXML(enmanejo):
    si tieneattr(cadena, 'V') y tieneattr(cadena, 'j'):
        imprimir

>>fuerademanejo, cadena
imprimir >>fuerademanejo, '</raíz>'

```

```

#!/usr/bin/env python

importar sis
importar

optanálisis

importar vdj

analizador = optanálisis.AnalizadordeOpción( )
analizador.adicionar_opción('-t', '--etiqueta', acción= 'append', dest= 'etiquetas')
(opciones, args) =

analizador.análisis_args ()

si len(args) == 2:
    enmanejo = abrir(args[0], 'r')
    fuerademanejo =

abrir(args[1], 'w')
elif len(args) == 1:
    enmanejo = abrir(args[0], 'r')
    fuerademanejo =

sis.stdafuera
elif len(args) == 0:
    enmanejo = sis.stdadentro
    fuerademanejo = sis.stafuera

pregunta_etiquetas = establecer(opciones.etiquetas)
para cadena en vdj.análisis_VDJXML(enmanejo) :
    si pregunta_etiquetas <=

cadena.todas_etiquetas:
    imprimir >>fuerademanejo, cadena
#!/usr/bin/env python

importar sis
importar
optanálisis

importar vdj

analizador = optanálisis.AnalizadordeOpción( )
analizador.adicionar_opción('-t', '--etiqueta', acción= 'append', dest= 'etiquetas')
(opciones, args) =

analizador.análisis_args ()

si len(args) == 2:
    enmanejo = abrir(args[0], 'r')
    fuerademanejo =

abrir(args[1], 'w')
elif len(args) == 1:
    enmanejo = abrir(args[0], 'r')
    fuerademanejo =

sis.stdafuera
elif len(args) == 0:
    enmanejo = sis.stdadentro
    fuerademanejo = sis.stafuera

```



```

vacío_establecer = establecer ( )
pregunta_etiquetas = establecer(opciones.etiquetas)
para cadena en vdj.análisis_VDJXML(enmanejo) :

    si pregunta_etiquetas & cadena.todas_etiquetas == vacío_establecer:
        imprimir >>fuera demanejo, cadena
#!

/usr/bin/env python

importar sis
importar

optanálisis

importar vdj

analizador =

optanálisis.AnalizadordeOpción( )
analizador.adicionar_opción('-t', '--etiqueta', acción= 'append', dest= 'etiquetas')
(opciones, args) = analizador.análisis_args ( )

si len(args) == 2:
    enmanejo = abrir(args[0], 'r')

    fuera demanejo = abrir(args[1], 'w')
elif len(args) == 1:
    enmanejo = abrir(args[0], 'r')

fuera demanejo = sis.stdafuera
elif len(args) == 0:
    enmanejo = sis.stdadentro
    fuera demanejo =

sis.stafuera

vacío_establecer = establecer ( )
pregunta_etiquetas = establecer(opciones.etiquetas)
para cadena en

    vdj.análisis_VDJXML(enmanejo) :
        si pregunta_etiquetas & cadena.todas_etiquetas != vacío_establecer:
            imprimir

>>fuera demanejo, cadena
#! /usr/bin/env python

importar sis
importar advertencias
importar optanálisis

importar seqherramientas
importar vdj
importar vdj.canal

analizador = optanálisis.AnalizadordeOpción( )
analizador.adicionar_opción('-i', '--IGHC', dest= 'ighc,fasta')
(opciones, args) = analizador.análisis_args ( )

si

```

```

len(args) == 2:
    enmanejo = abrir(args[0], 'r')
    fuerademanejo = abrir(args[1], 'w')
elif

len(args) == 1:
    enmanejo = abrir(args[0], 'r')
    fuerademanejo = sis.stdafuera
elif len(args)

== 0:
    enmanejo = sis.stdadentro
    fuerademanejo = sis.stafuera

# isotipos de carga
isotipos =

vdj.canal.carga_isotipos(opciones.ighc_fasta)

para cadena en vdj.análisis_VDJXML(enmanejo) :

vdj.canal.id_isotipo(cadena,isotipos)
imprimir >>fuerademanejo, cadena
#! /usr/bin/env

python

importar sis
importar optanálisis

importar seqherramientas
importar vdj

analizador =

optanálisis.AnalizadordeOpción( )
analizador.adicionar_opción('-b', '--nombrebase')
(opciones, args) =

analizador.análisis_args ()

si len(args) == 1:
    enmanejo = abrir(args[0], 'r')
elif len(args) ==

0:
    enmanejo = sis.stdadentro

# NOTA: este guion ignora los números de alelos

vdj.canal.partición_VJ(enmanejo,opciones.nombrebase)
# nombre de base para todos los archivos de datos
nombrebase

# archivos de entrada (camino completos)
entrada_fasta # los datos fasta iniciales
códigos de barras_fasta

# identificadores de códigos de barras
isotipo_fasta # identificadores de isotipo

```

```

# directorios de trabajo
análisis_dir      # camino completo; directorio base para todo

# parámetros de análisis
mín_tamaño
máx_tamaño
paquete_tamaño
loci

#!/usr/bin/env python

importar sis
importar optanálisis

importar

vdj

analizador = optanálisis.AnalizadordeOpción( )
analizador.adicionar_opción('-m', '--mín', 'tipo= 'int', defecto=0)
analizador.adicionar_opción('-M', '--máx', 'tipo= 'int', defecto=flotar('inf'))
(opciones, args) =

analizador.análisis_args ( )

si len(args) == 2:
    enmanejo = abrir(args[0], 'r')
    fuerademanejo =

abrir(args[1], 'w')
elif len(args) == 1:
    enmanejo = abrir(args[0], 'r')
    fuerademanejo =

sis.stdafuera
elif len(args) == 0:
    enmanejo = sis.stdadentro
    fuerademanejo = sis.stafuera

imprimir

>>fuerademanejo, "<raíz>"

para cadena en vdj.análisis_VDJXML(enmanejo) :
    si len(cadena) >=

opciones.mín y len(cadena) <= opciones.máx:
    imprimir >>fuerademanejo, cadena

imprimir

>>fuerademanejo, "<raíz>"
#!/usr/bin/env python

importar sis
importar optanálisis
importar os

importar

```

```

vdj

analizador = optanálisis.AnalizadordeOpción( )
analizador.adicionar_opción('-t', '--etiqueta', acción= 'append', dest= 'etiquetas')
(opciones, args) =

analizador.análisis_args ( )

si len(args) == 1:
    ennombre == args[0]
    enmanejo =

abrir(args[0], 'r')
de otro modo:
    levantar Excepción, "Necesita un archivo de entrada único."

(nombre de base, ext)=

os.camino.divisiónnext(ennombre)
nombre de base = os.camino.nombre de base(nombre de base)

fuera de manejos = { }
para etiqueta en

opciones.etiquetas:
    fueradenombre = nombre de base+ '.' +vdj.sequentials.limpiar_id(etiqueta)+ext

fuera de manejos[etiqueta] = abrir(fueradenombre, 'w')

pregunta_etiquetas = establecer(opciones.etiquetas)
para cadena en

vdj.análisis_VDJXML(enmanejo):
    tratar:
        etiqueta = (pregunta_etiquetas & cadena.todas.etiquetas).pop( )

    imprimir >>fuera de manejos[etiqueta], cadena
    excepto Error de Llave:
        continuar

para manejo en

fuera de manejos.valoresiter():
    manejo.cerrar()
#! /usr/bin/env python

importar sis
importar

optanálisis

importar vdj

analizador = optanálisis.AnalizadordeOpción( )
analizador.adicionar_opción('-t', '--etiqueta', acción= 'append', dest= 'etiquetas')
(opciones, args) =

analizador.análisis_args ( )

si len(args) == 2:
    enmanejo = abrir(args[0], 'r')

```

```

    fuerademanejo =

abrir(args[1], 'w')
elif len(args) == 1:
    enmanejo = abrir(args[0], 'r')
    fuerademanejo =

sis.stdafuera
elif len(args) == 0:
    enmanejo = sis.stdadentro
    fuerademanejo = sis.stafuera

para
cadena en vdj.análisis_VDJXML(enmanejo):
    cadena.adicionar_etiquetas(opciones.etiquetas)
    imprimir >>fuerademanejo,

cadena
#!/usr/bin/env python

importar sis

importar vdj
importar vdj.legado

si len(sis.argv) ==

3:
    enmanejo = abrir(sis.argv[1], 'r')
    fuerademanejo = abrir(sis.argv[2], 'w')
elif

len(sis.argv) == 2:
    enmanejo = abrir(sis.args[1], 'r')
    fuerademanejo = sis.stdafuera
elif

len(sis.argv) == 1:
    enmanejo = sis.stdadentro
    fuerademanejo = sis.stafuera

para cadena en
vdj.legado.análisis_VDJXML_antiguo(enmanejo):
    imprimir >>fuerademanejo, cadena
#!/usr/bin/env

python

importar lsf
importar seqherramientas
importar vdj
importar vdj.canal

unir =

os.camino.unir

# DEFINICIÓN DE PARÁMETRO

```

ES 2 730 951 T3

```
# procesar el archivodetrabajo para parámetros de entrada
# define las

siguientes variables:
    # nombre de base          # identificador de base única para datos
    #

entrada_fasta    # los datos fasta iniciales
    # códigos de barras_fasta # identificador de códigos de barras
    #

isotipo_fasta    # identificadores de isotipo
    # análisis_dir          # camino completo; directorio base para

todo
    # mín_tamaño            # selección de tamaño mín
    # máx_tamaño            # tamaño máx

selección
    # paquete_tamaño        # tamaño de paquete para trabajos de alineamiento
    # loci                   # el

loci para usar para VDJ aln

# directorios de trabajo
partes_dir = 'partes'
log_dir = 'logs'
partición_dir = 'particiones'

# archivos de salida
crudo_vdjxml = nombrebase+ '.alineado.vdjxml'
vj_filtrado_archivo =
nombrebase+ '.VJ_filtrado.vdjxml'
tamaño_seleccionado_archivo = nombrebase + '.tamaño%i-%i' %
(mín_tamaño,máx_tamaño) + '.vdjxml'

locus_opciones = ' '.unir([' --locus %s' locus para locus
en loci.división()]])

# CANAL COMIENZA AQUÍ

# 0. CONVERSIÓN A VDXML
# 1. TAMAÑO

SELECCIÓN
enmanejo = abrir(entrada_fasta, 'r')
fueraenmanejo =

abrir(unir(análisis_dir,tamaño_seleccionado_archivo), 'w')
# iterar a través de entradas fasta
para
```

ES 2 730 951 T3

```
(descri,seq) en seqherramientas.Fastalterador(enmanejo, lambda d: d.división() [0]) :
# convertir a

CadenaInmune
cadena = vdj.CadenaInmune(descr=descr,seq=seq)

# selección de tamaño
si

len(cadena) < opciones.mín o len(cadena) > opciones.máx:
continuar

imprimir

>>fueraemanejo, cadena
enmanejo.cerrar()
fueraemanejo.cerrar()

# 2. DIVISIÓN EN PARTES
enmanejo =

abrir(unir(análisis_dir,tamaño_seleccionado_archivo), 'r')
partes = vdj.canal.iterador2partes(

vdj.análisis_VDJXML(enmanejo),

unir(análisis_dir,tamaño_seleccionado_archivo),

tamañodepaquete,

                                prefijo= '<raíz>',

                                sufijo = '</raíz>')

cdm = 'códigos de barras_id --códigos de barras %s ' % códigos de barras_archivo # 3.

CÓDIGOS DE BARRAS, IDENTIFICACIÓN # 4. CODIFICACIÓN
cdm += ' | codificación_hebra' + locus_opciones

HEBRA # 5. ID DE ISOTIPO
si 'IGH' en loci:

(solo cadena pesada)
cdm += ' | isotipo_id --IGHC %s ' % isotipo_archivo
cdm += ' | align_vdj +

locus_opciones # 6. CLASIFICACIÓN DE VDJ

# enviar cdm a LSF para cada parte
trabajoIDs = [ ]
logarchivos = [ ]
fueraenombres = [ ]
para parte en partes
parteID =parte.división('.') [-1]

partefueraenombre = nombrebase+ '.prealign.vdjxml.' +parteID
fueraenombres.append(partefueraenombre)
```



```
#####  
#####
```

Este es un análisis

de 454 datos generados a partir del segundo experimento de vacunación del segundo individuo. Creo

que estos datos vienen de Ido.

Los datos crudos están localizados en

vdj-ome/crudo-seq-datos/454/crudo_20100927_flu2/
con los archivos relevantes

RD_análisis/*GAC*.fna

Estoy concatenando estos archivos en un solo archivo FASTA ubicado aquí

```
vdj-ome/estable-datos/crudo_flu2_20100927/  
usando este comando:  
cat *GAC*.fna >
```

```
~/investigación/iglesia/vdj-ome/estable-datos/crudo_flu2_20100927/pesada_cadenas.flu2.20100927.f  
asta  
del directorio  
vdj-ome/crudo-seq-datos/454/crudo_20100927_flu2/RD_análisis  
Este archivo es
```

```
llamado:  
pesada_cadenas.flu2.20100927.fasta
```

Los histogramas de longitud de lectura inicial son

generados en este guión:
ejecuciónestadísticas.py

La salida es:
Número de lecturas: 912388

Lectura más corta: 40 pb
Lectura más larga: 600 pb
junto con dos figuras que muestran un

pico muy apretado en ~455 pb.

La figura sugiere cortes de tamaño de 395 pb a 520 pb.

1.

```
fasta2vdjxml.py  
2. tamaño_seleccionar.py
```

Primero convierto el archivo fasta en vdjxml, y,
tamaño seleccionar las lecturas basadas en el hist de lectural, 395-520

ES 2 730 951 T3

```
(de
/inicio/ul2/vdj-ome/estable-datos/crudo_flu2_20100927):
python ~/código/vdj/bin/fasta2vdjxml.py

pesada_cadenas.flu2.20100927.fasta | python ~/código/vdj/bin/tamaño_seleccionar.py --mín 395
--máx 520

> pesada_cadenas.flu2.20100927.tamaño395-520.vdjxml

Hay 902805 cadenas del tamaño
seleccionado en el archivo.

3. vdjxml2partes.py

Dividir vdjxml en trozos pequeños y colocarlos en
el directorio de trabajo

mkdir ~/vdj-ome/análisis/flu2_seq_canal/datos
~/código/vdj/bin/vdjxml2partes.py --tamañodepaquete 10000 --nombrebase
~/vdj-ome/análisis/flu2_seq_canal/datos/pesada_cadenas.flu2.20100927.tamaño395-520.vdj
xml
pesada_cadenas.flu2.20100927.tamaño395-520.vdjxml

Cambio de directorio a todas las partes:
cd

~/vdj-ome/análisis/flu2_seq_canal/datos

4. códigos de barras_id.py
5. codificación_hebra.py

Identificar códigos de barras para cada lectura, y

Determinar donde tenemos la hebra correcta o
no

Asegúrese que el archivo fasta de código de barras esté correctamente referenciado.

para ARCHIVO en

pesada_cadenas.flu2.20100927.tamaño395-520.vdjxml.*; hacer
NOMBRE=${ARCHIVO%.tamaño*}*.prealign.vdjxml."${ARCHIVO}*.vdjxml.*}
# eco $NOMBRE
bsub

-qcompartido_2h -o -pre-alineamiento.log "python ~/código/vdj/bin/código de barras_id.py --código de barras
~/vdj-ome/estable-datos/código de barras/IDT.454.rápido.MIDs.fasta $ARCHIVO | python
~/código/vdj/bin/codificación_hebra.py --locus IGH > $NOMBRE"
```

ES 2 730 951 T3

```
hecho
# python

~/código/vdj/bin/código de barras_id.py --código de barras

~/vdj-ome/estable-datos/código de barras/IDT.454.rápido.MIDs.fasta
# python

~/código/vdj/bin/codificación_hebra.py --locus IGH

Algunas ESTADÍSTICAS:
# Núm de cadenas
cat

*prealign* | grep "<Cadenalnmune>" | wc -1
902805

# Núm con código de barras
cat

*prealign* | grep "<código de barras>" | wc -1
882978

# Núm que fueron

complementados en reversa
cat *prealign* | grep "<revcomp>" | wc -1
434034

#

Desglose de código de barras
para NÚM en 13 14 15 16 17 18 19 20 21 22; hacer
cat *prealign* |

grep "<código de barras>RL0$NÚM" | wc -1
hecho

      RL013  18255  2.1%
      RL014  96568
10.9%
      RL015  88732  10.0%
      RL016  80088  9.1%
      RL017  79637
9.0%
      RL018  97962  11.1%
      RL019  223744  25.3%
      RL020  60479
6.8%
      RL021  73613  8.3%
      RL022  63900  7.2%

6. align_vdj.py
para

ARCHIVO en pesada_cadenas.flu2.20100927.prealign.vdxml.*; hacer
```

REIVINDICACIONES

1. Un método para identificar un anticuerpo útil en métodos diagnósticos o terapéuticos, que comprende:
- 5 (a) acoplar las secuencias de polinucleótidos de la región variable de la cadena pesada (V_H) y de la región variable de la cadena liviana (V_L) de una sola célula de una pluralidad de células de una muestra biológica de un sujeto humano en un primer punto en el tiempo para formar secuencias acopladas, en donde el sujeto humano tiene una enfermedad o trastorno o se le ha administrado un agente para estimular un desafío inmunitario;
- (b) realizar una secuenciación de alto rendimiento de polinucleótidos amplificados a partir de las secuencias acopladas;
- 10 (c) reagrupar las lecturas de secuenciación de acuerdo con al menos su segmento V y usar del segmento J para formar clones VDJ y determinar la frecuencia de clonación de los clones VDJ;
- (d) comparar la frecuencia de clones de un clon de VDJ con (i) la frecuencia de clones de clones de VDJ de una muestra biológica del sujeto humano en un segundo punto en el tiempo, o (ii) la frecuencia de clones de clones de VDJ de una muestra biológica de un negativo sujeto de control humano sin la enfermedad o trastorno o un sujeto humano al que no se ha administrado el agente para estimular un desafío inmunitario; y
- 15 (e) identificar un anticuerpo útil en métodos de diagnóstico o terapéuticos con base en (d).
2. El método de la reivindicación 1, que comprende además seleccionar el anticuerpo para determinar la especificidad funcional, la afinidad y/o el anticuerpo en cuanto a la capacidad de neutralización, que además comprende preferiblemente la identificación de un antígeno que interactúa con el anticuerpo como un biomarcador para una enfermedad.
- 20 3. El método de la reivindicación 1 o 2, que comprende además clonar el anticuerpo directamente en tecnología de visualización de superficie.
4. El método de una cualquiera de las reivindicaciones 1-3, en donde el anticuerpo es un anticuerpo neutralizante o un anticuerpo de respuesta rápida.
- 25 5. El método de una cualquiera de las reivindicaciones 1-4, que comprende además determinar un isotipo del anticuerpo.
6. El método de las reivindicaciones 1-5, que comprende amplificar polinucleótidos mediante PCR con cebadores no específicos, cebadores degenerados o cebadores específicos.
- 30 7. El método de la reivindicación 6, en donde los polinucleótidos amplificados se amplifican por PCR con conjuntos de cebadores específicos que se hibridan con las cadenas pesadas y livianas de las células B, células T o células B y células T.
8. El método de la reivindicación 6 o 7, en donde los polinucleótidos amplificados se amplifican por PCR con dos conjuntos de cebadores, en donde el primer conjunto de cebadores se hibrida a las regiones V y el segundo conjunto de cebadores se hibrida a las regiones C.
- 35 9. El método de una cualquiera de las reivindicaciones 1-8, que además comprende generar una base de datos de secuencias V_H y V_L pareadas, preferiblemente en donde la identificación comprende alinear los datos de secuencia de la base de datos contra los segmentos V, D y J conocidos o esperados en una base de datos personal, una base de datos NCBI, o una base de datos IMGT.
10. El método de una cualquiera de las reivindicaciones 6-9, en donde los polinucleótidos amplificados comprenden adaptadores de secuencia compatibles con la secuenciación de alto rendimiento de próxima generación.
- 40 11. El método de una cualquiera de las reivindicaciones 1-10, en donde las secuencias de polinucleótidos enlazadas que codifican un V_H y un V_L están codificadas con barras, en donde se puede usar la codificación de barras para lograr la reasignación de los polinucleótidos amplificados a sus moléculas de partida originales.
12. El método de una cualquiera de las reivindicaciones 1-11, en donde se produce un coeficiente de variación para comparaciones replicadas de aproximadamente 0.5 en frecuencias de clonación VDJ de $<10^{-4}$ - 10^{-3} .
- 45 13. El método de las reivindicaciones 1-12, en donde (c) comprende alinear las lecturas de secuencia con los segmentos V, D y J conocidos o esperados.
14. El método de cualquiera de las reivindicaciones 1-13, en donde (d) comprende alinear las lecturas de secuenciación de (c) a (i) lecturas de secuencia obtenidas de la muestra biológica del sujeto humano en un segundo punto en el tiempo, o (ii) lecturas de secuencia obtenidas de la muestra biológica de un sujeto humano sin la

enfermedad o trastorno o lecturas de secuencia obtenidas de la muestra biológica de un sujeto humano al que no se le ha administrado el agente para estimular un desafío inmune.

15. El método de una cualquiera de las reivindicaciones 1-14, en donde (c) comprende lecturas de secuencia de agrupamiento para su CDR3 único para formar los clones VDJ.

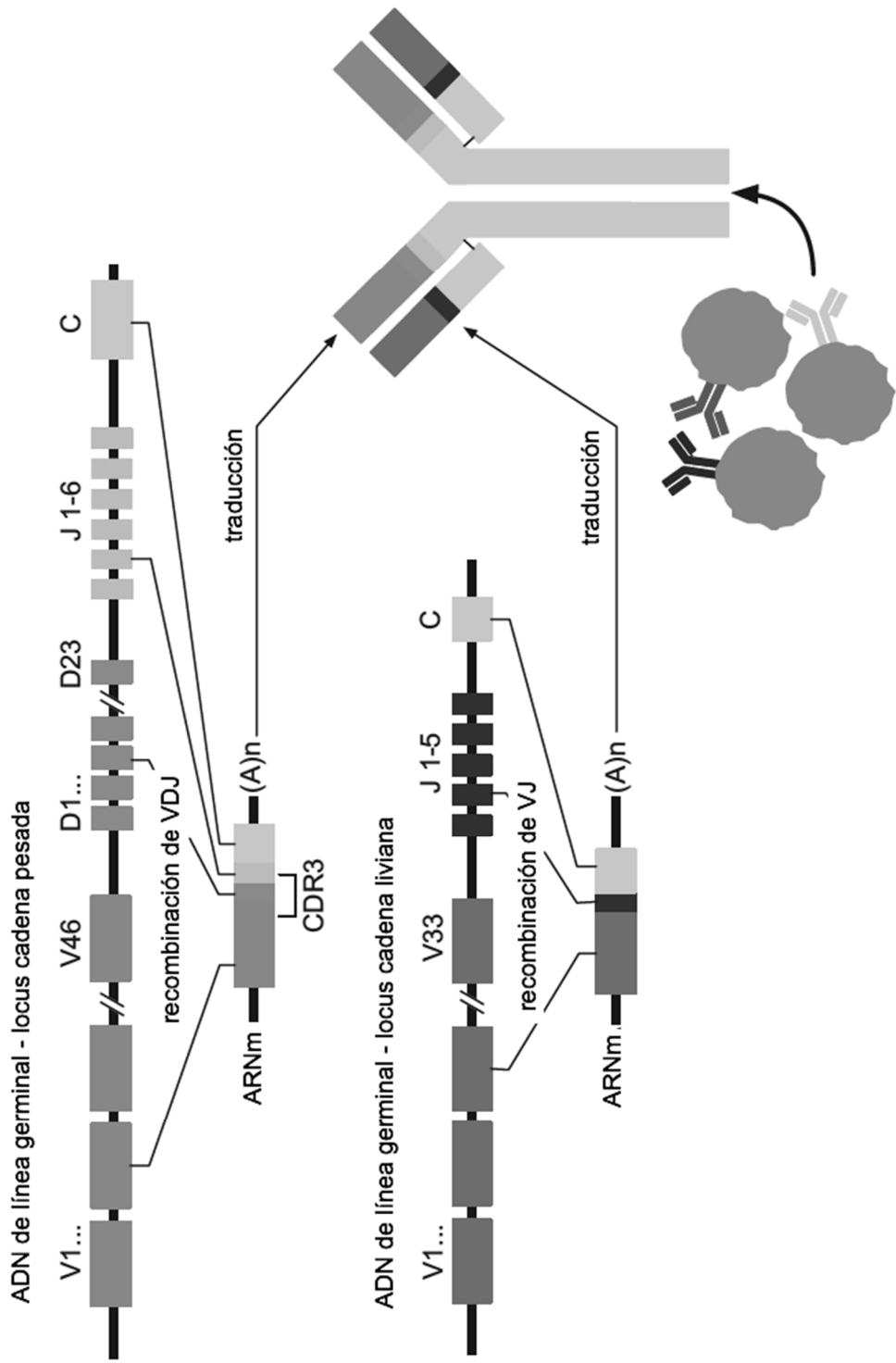


FIG. 1

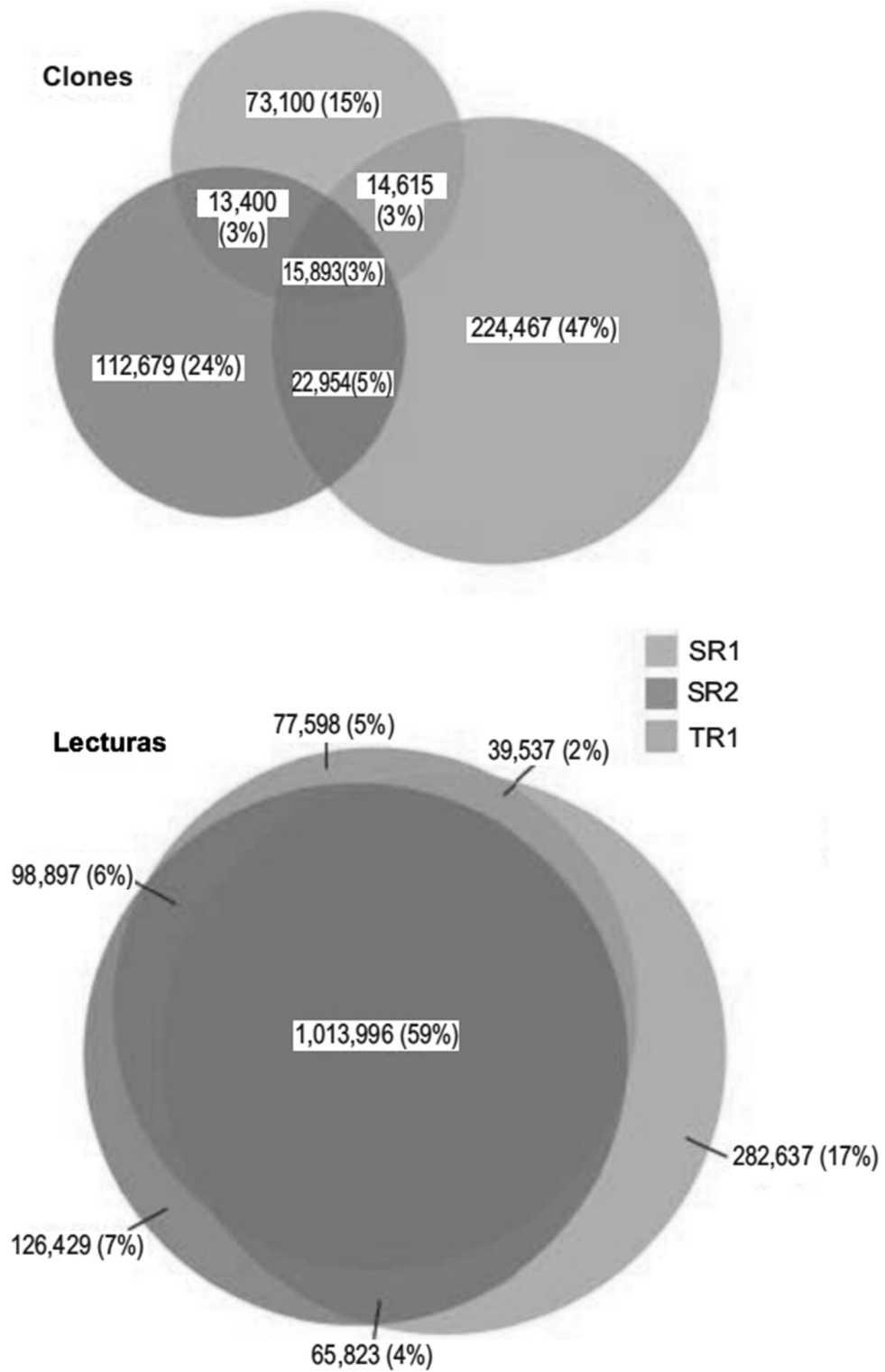


FIG. 2A

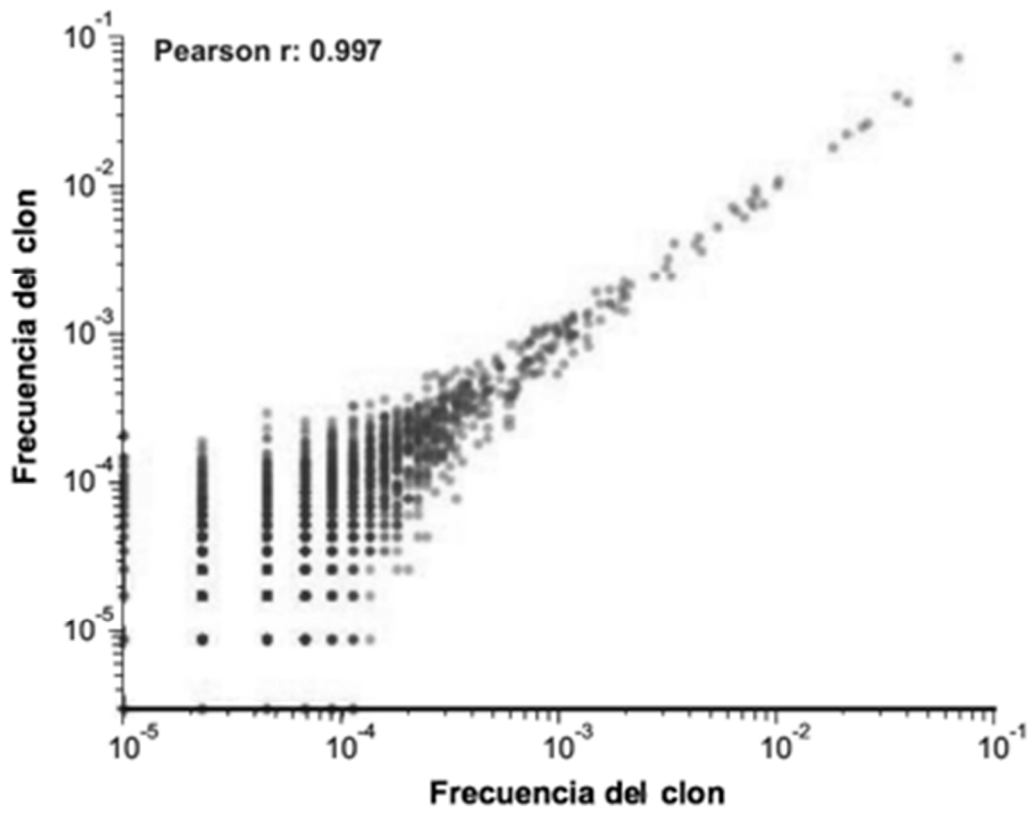


FIG. 2B

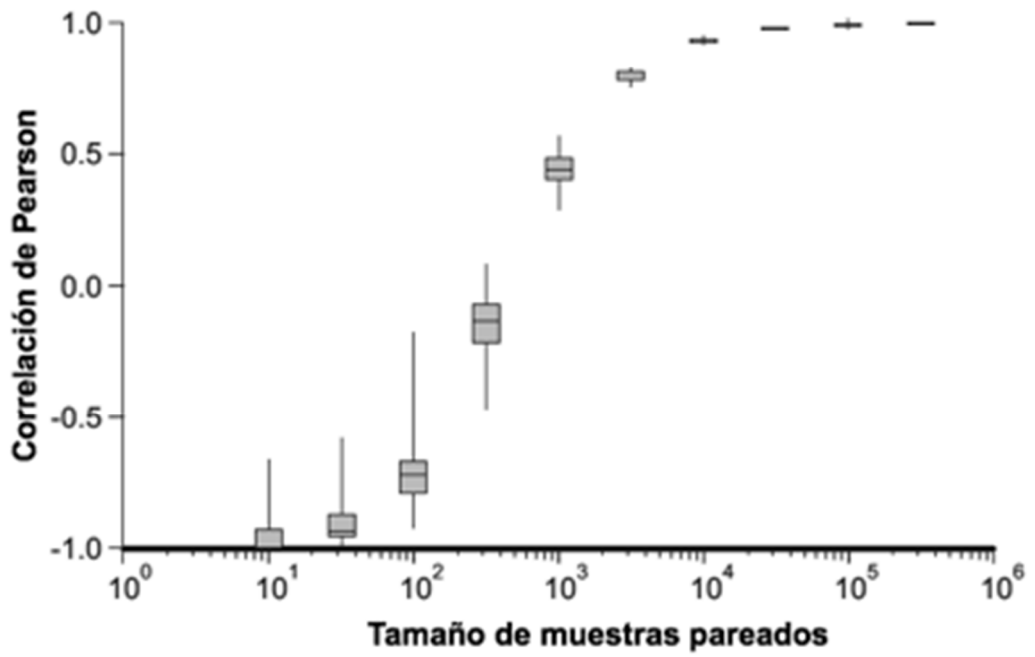


FIG. 2C

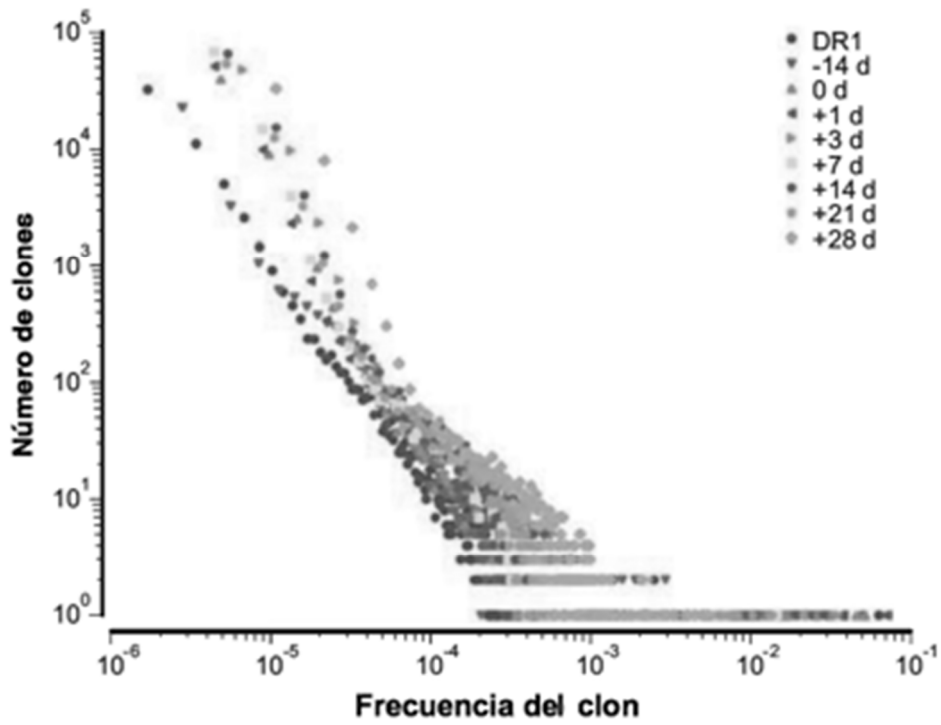


FIG. 2D

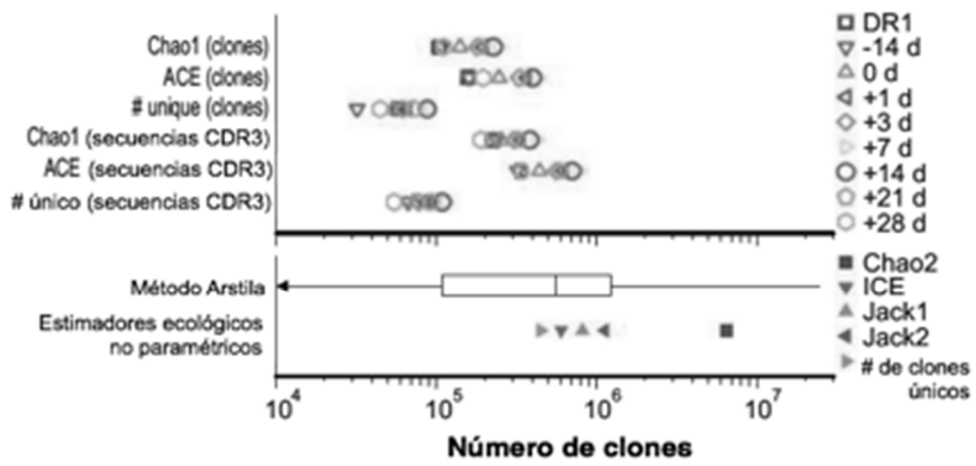


FIG. 2E

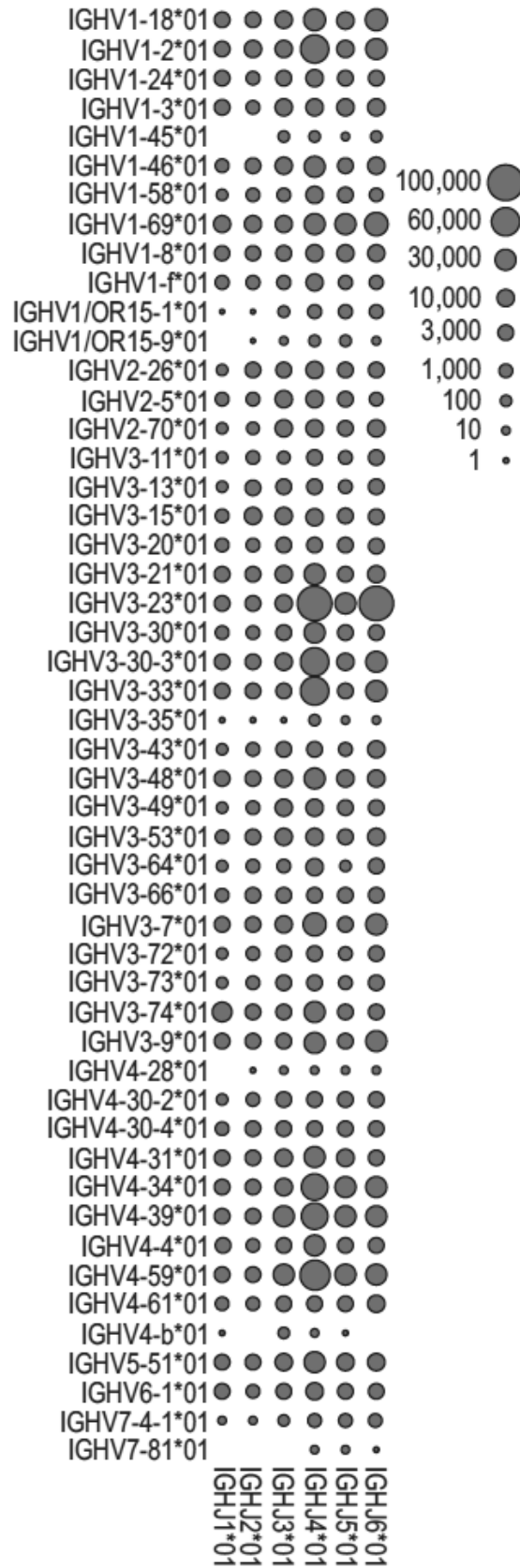


FIG. 3A

FIG. 3B

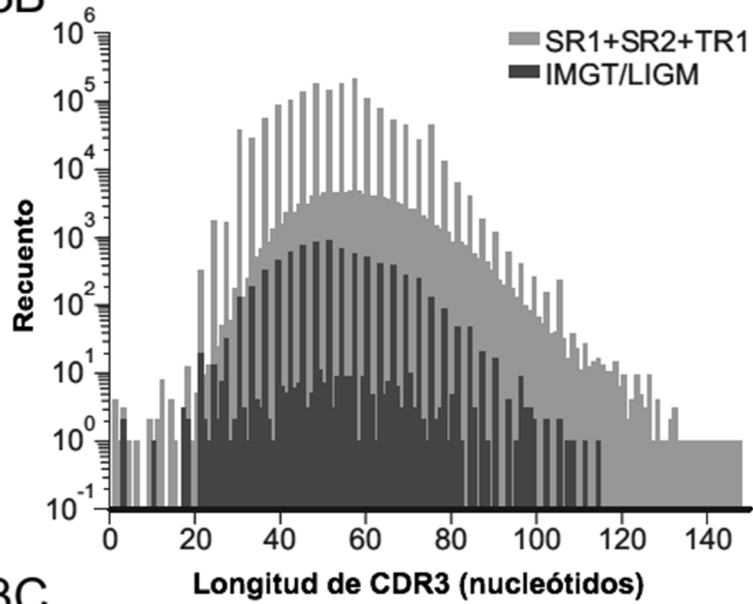
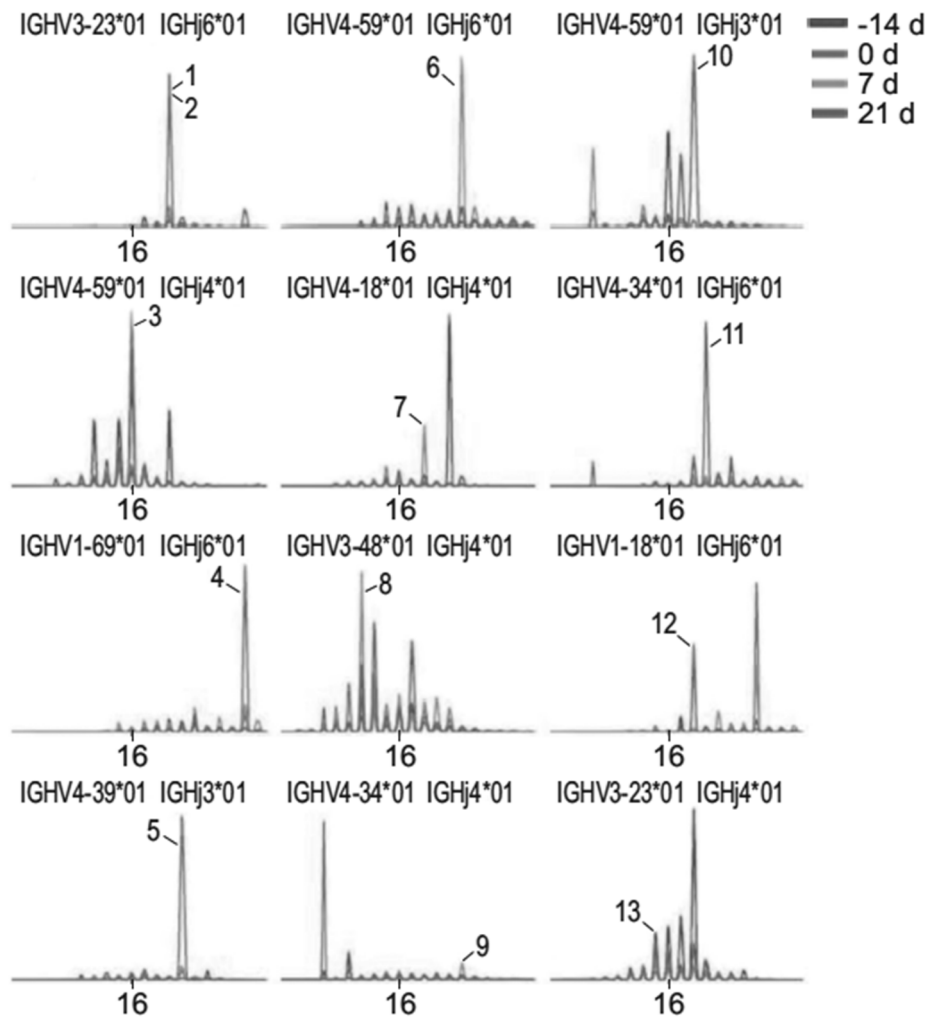


FIG. 3C



Vigneault F. et al. Figura 4

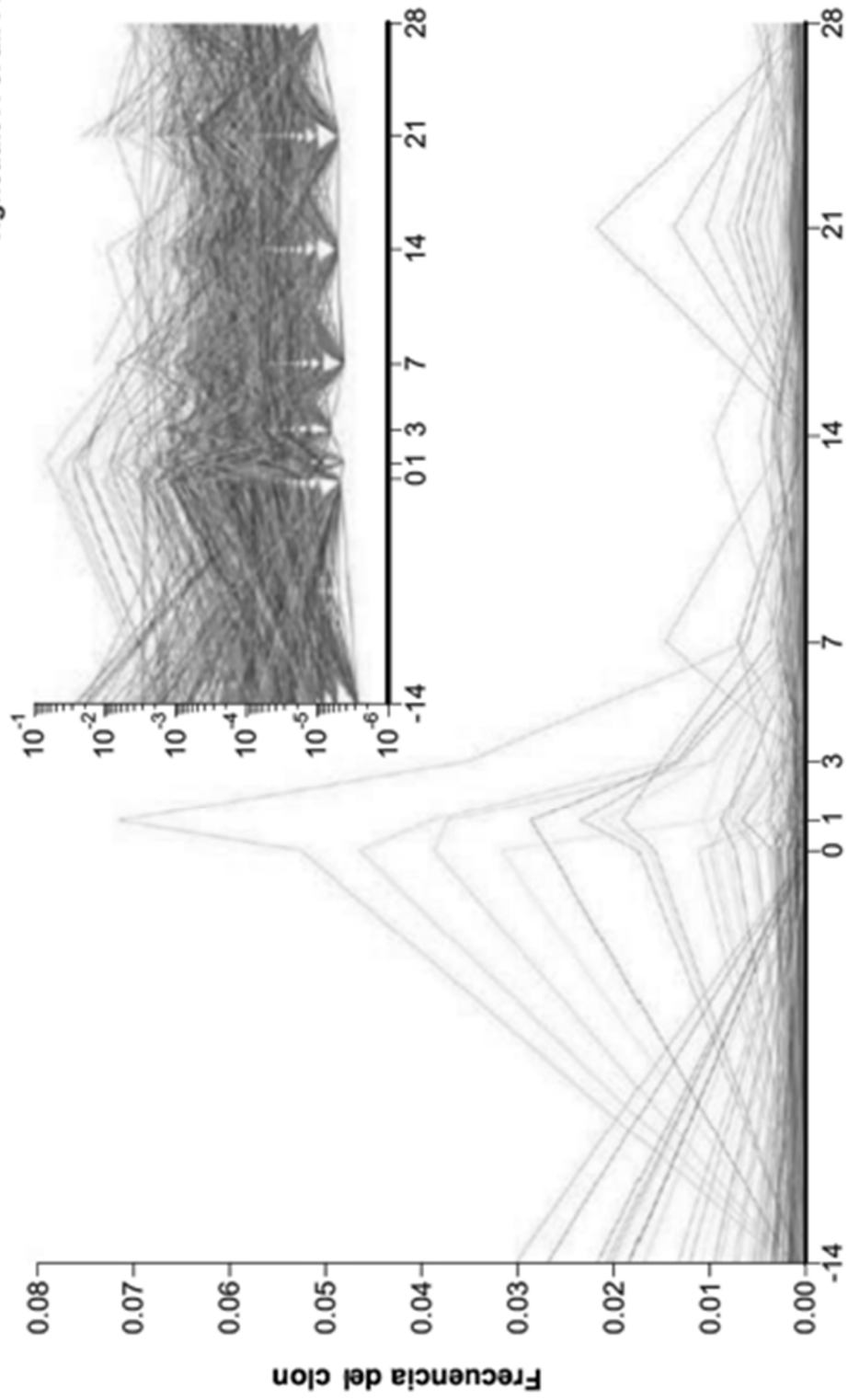


FIG. 4A

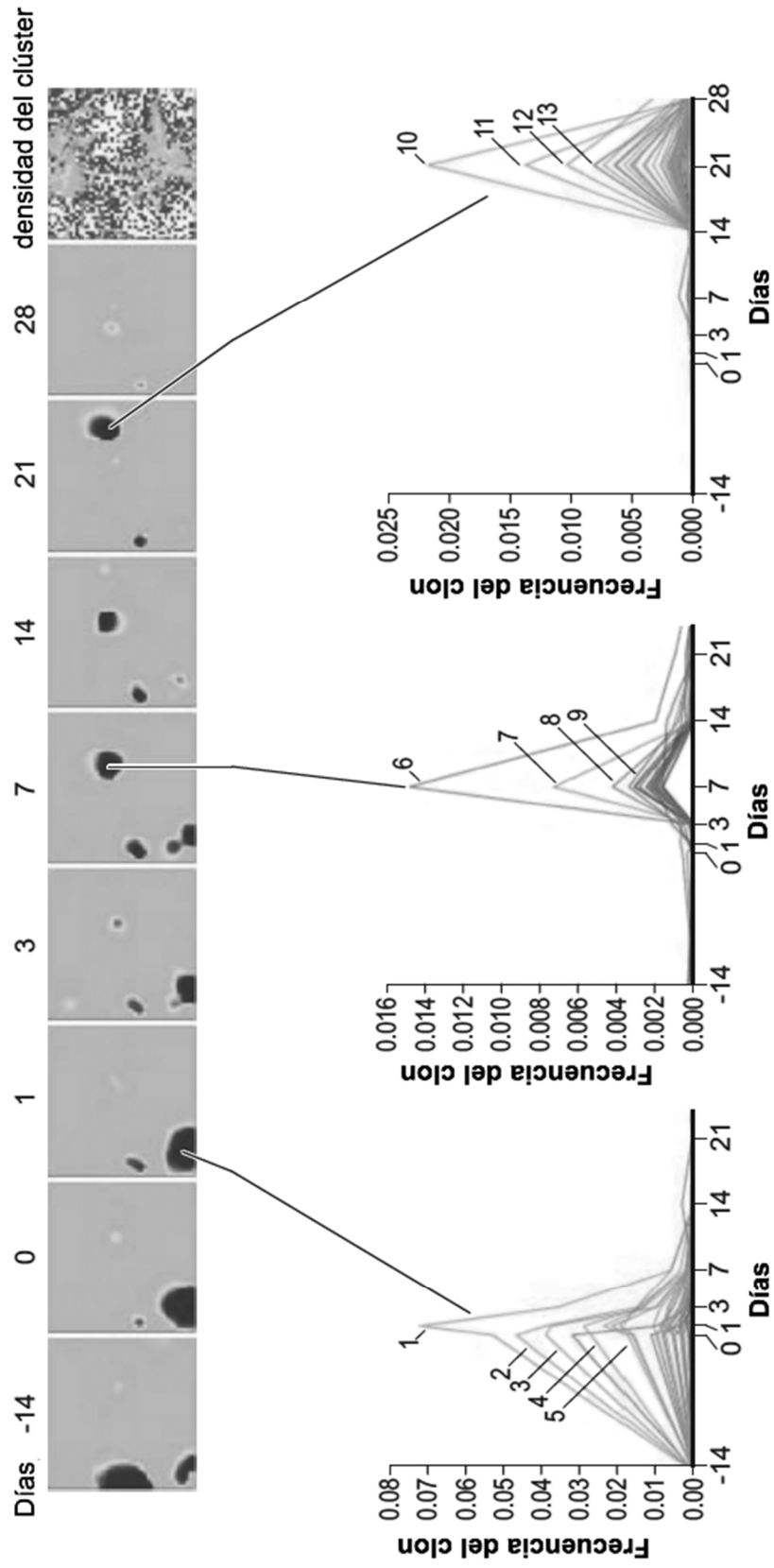


FIG. 4B

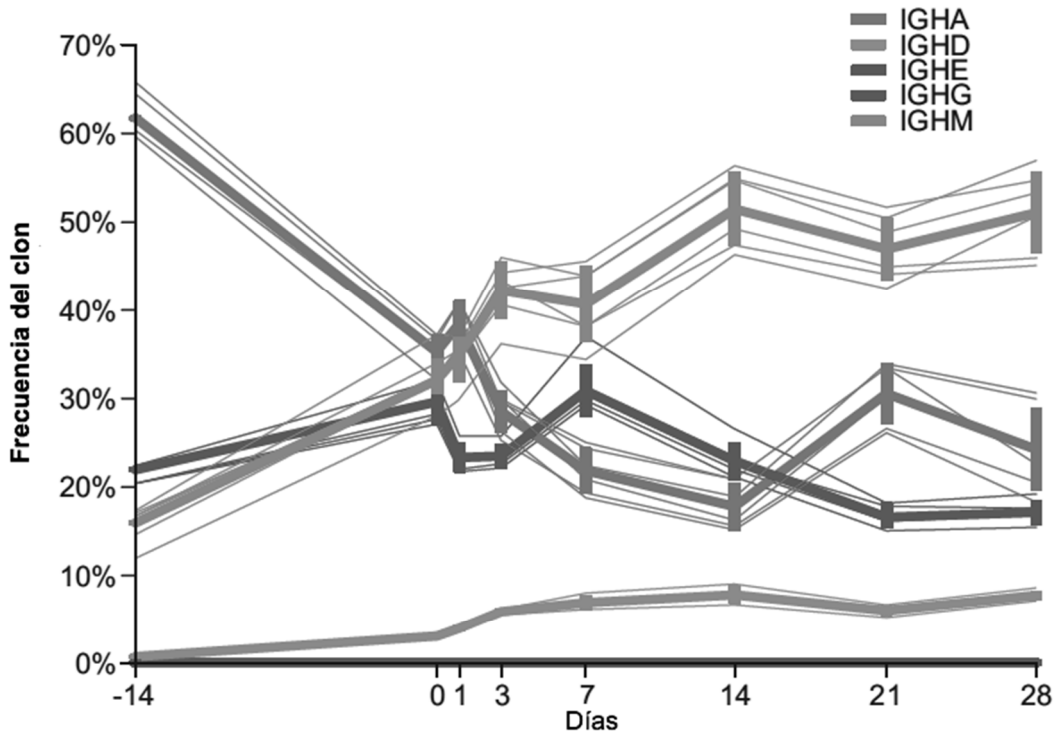


FIG. 4C

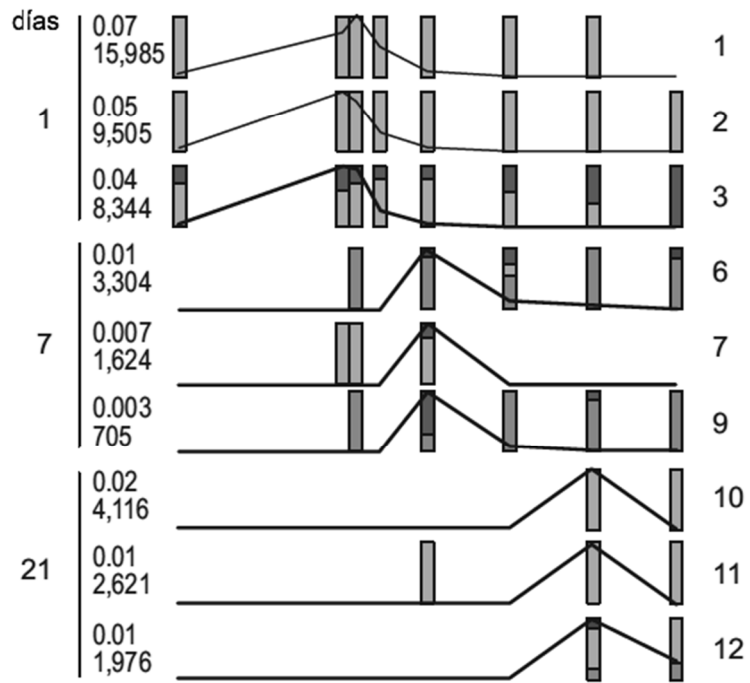


FIG. 4D

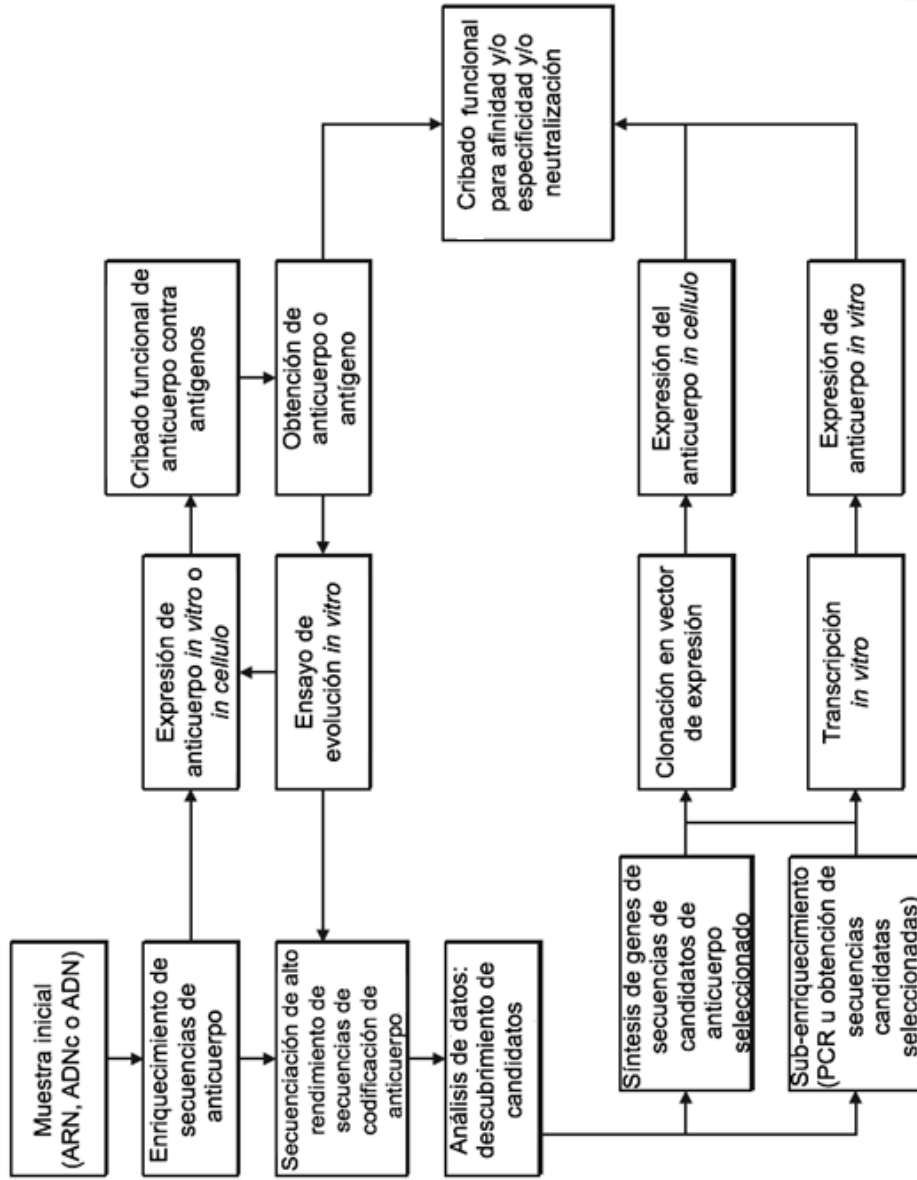


FIG. 5

FIG. 6A

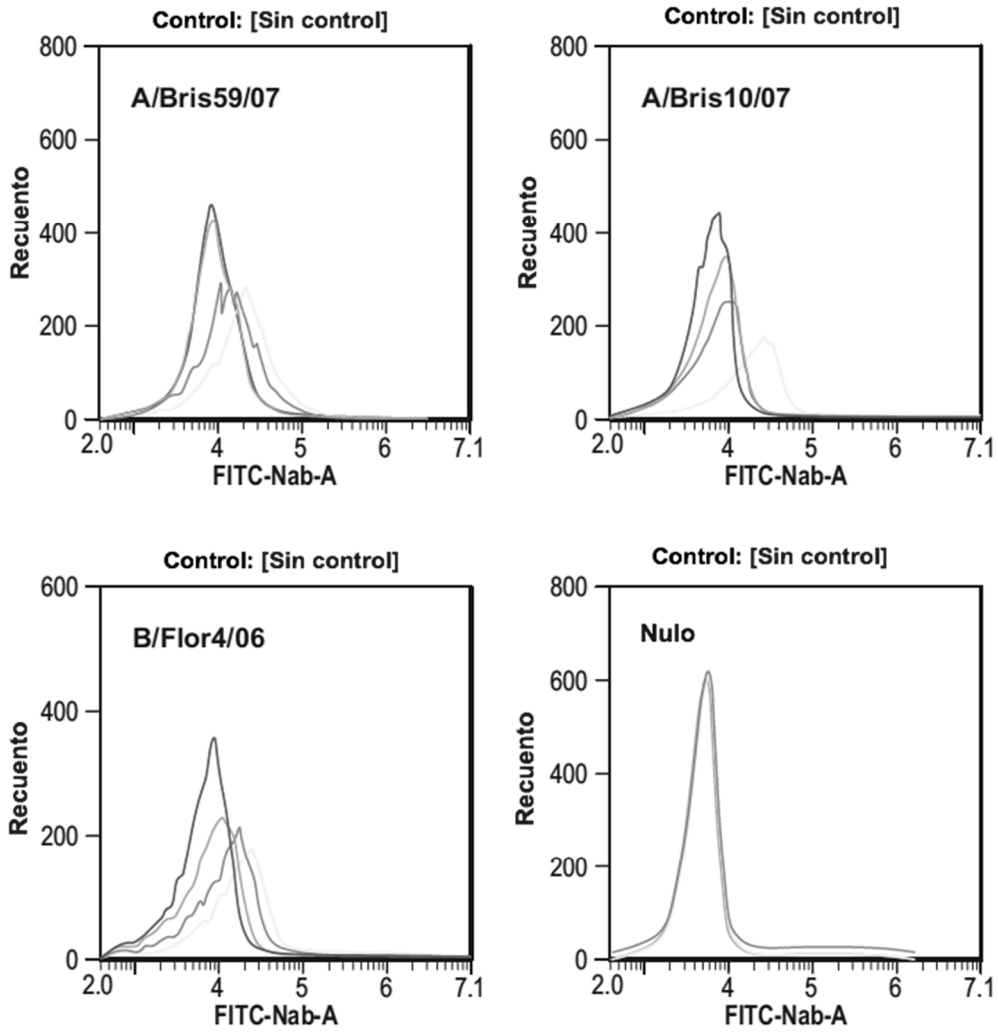


FIG. 6B

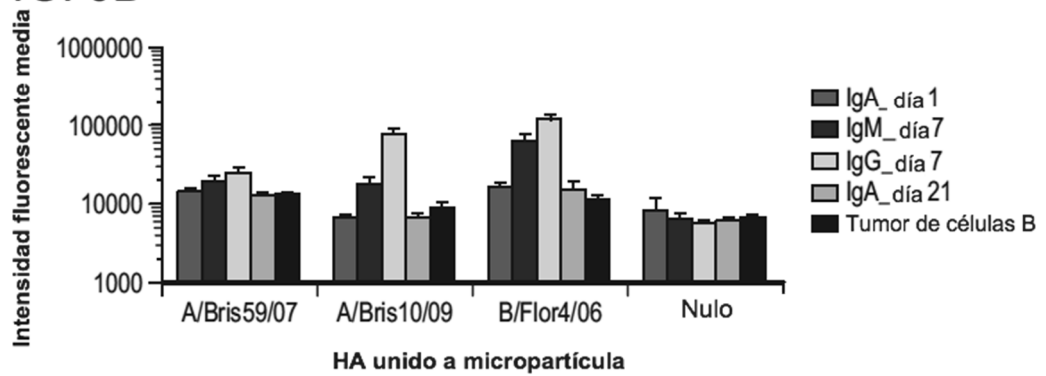
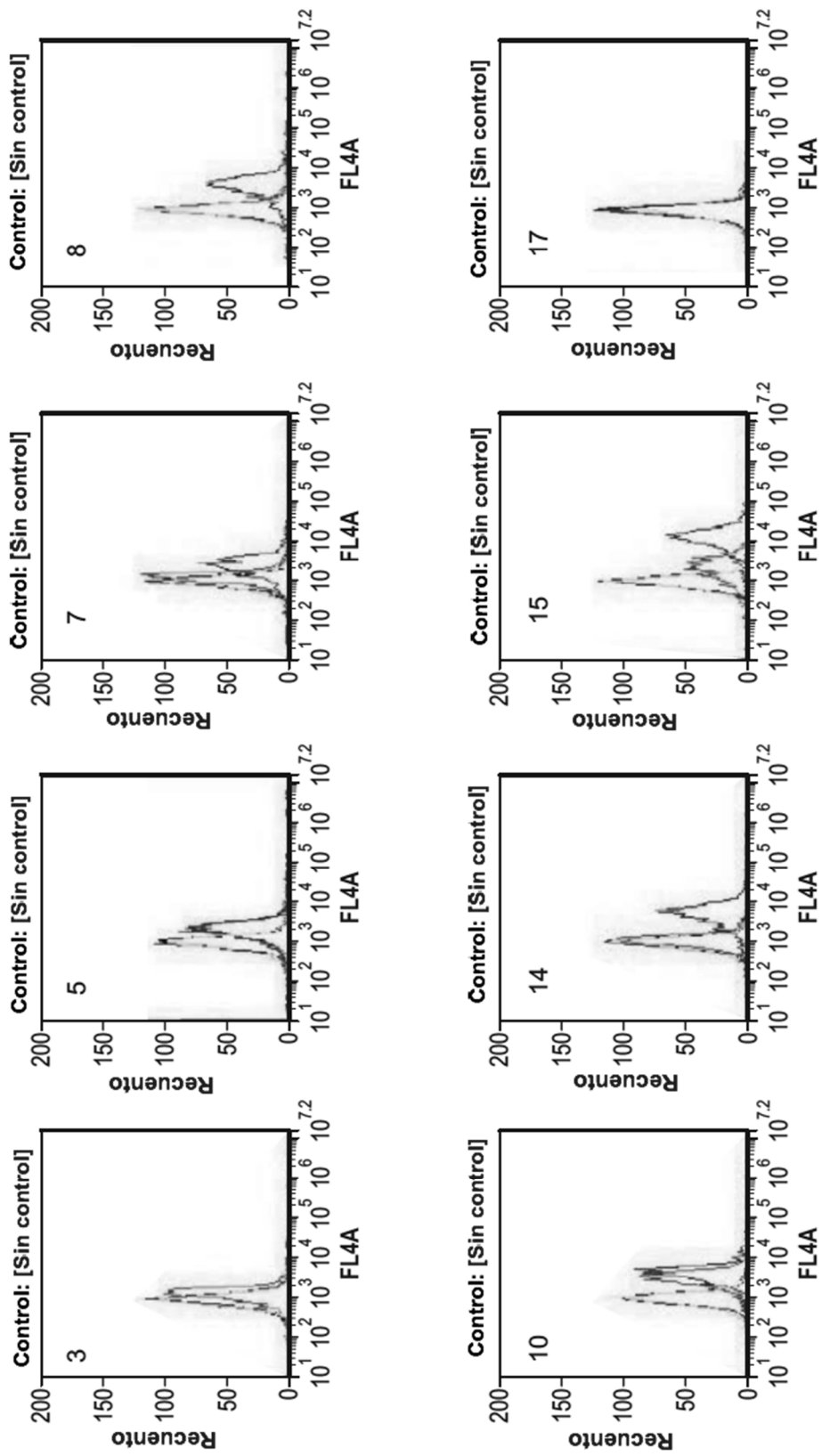


FIG. 6C



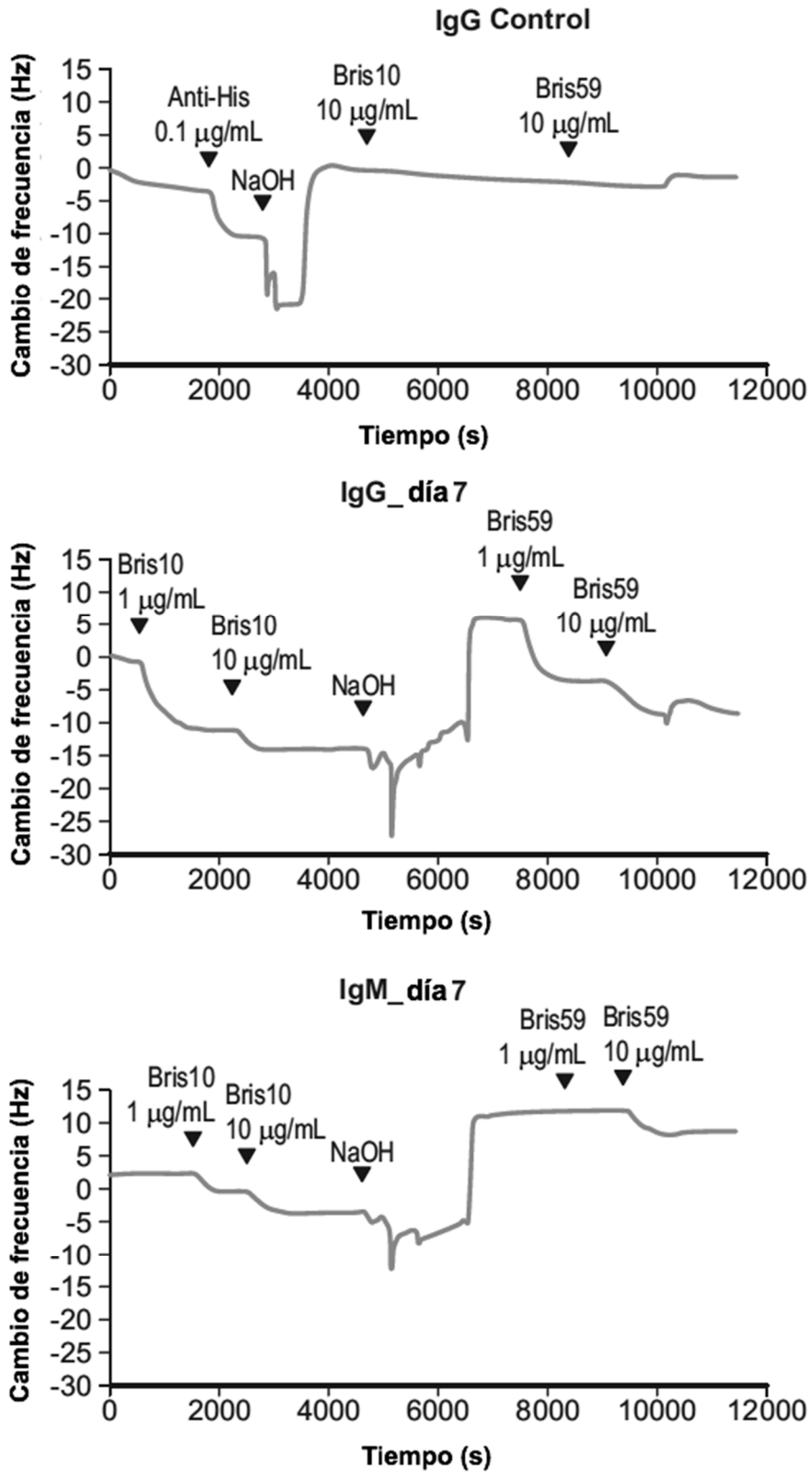


FIG. 7A

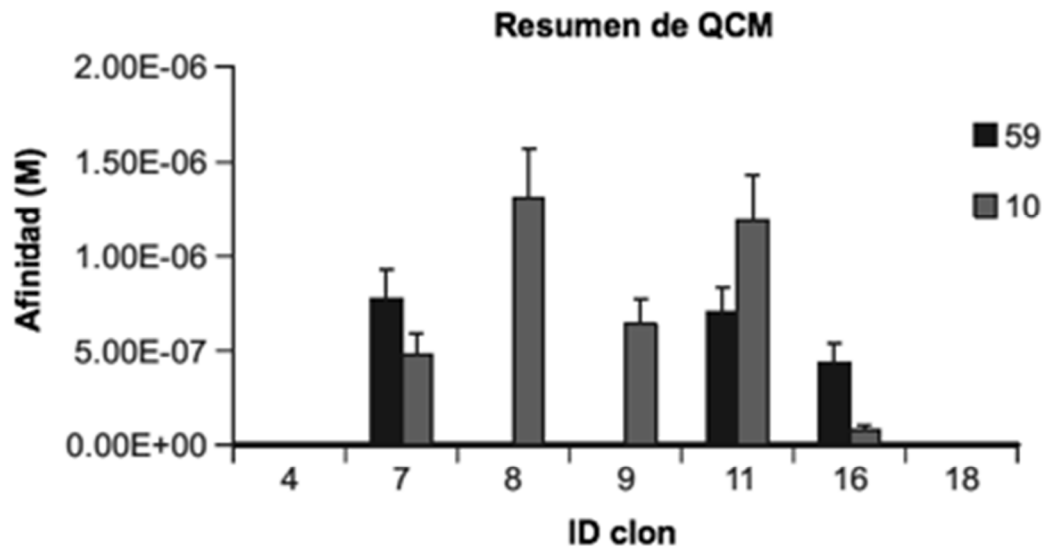


FIG. 7B

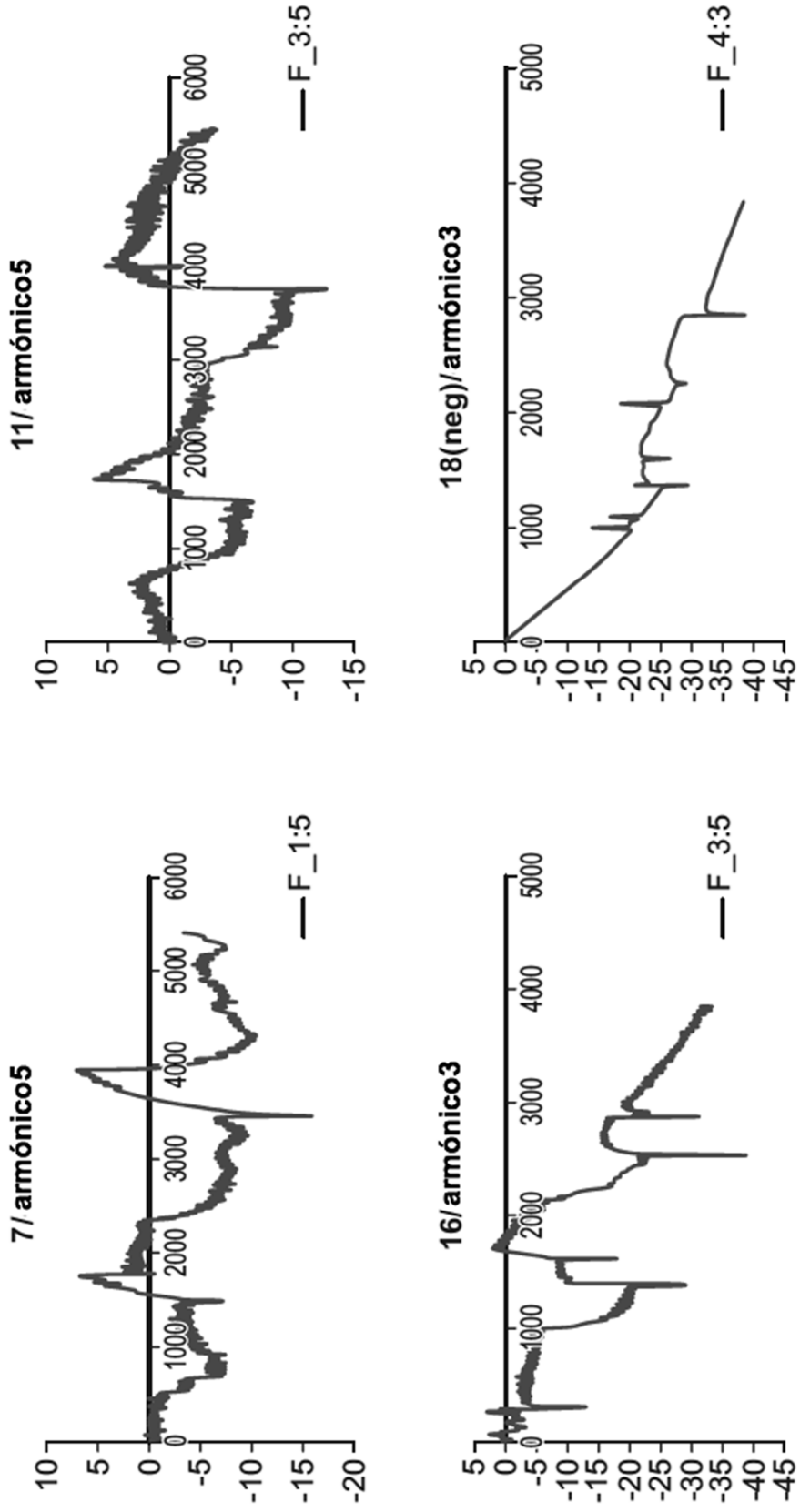


FIG. 7C

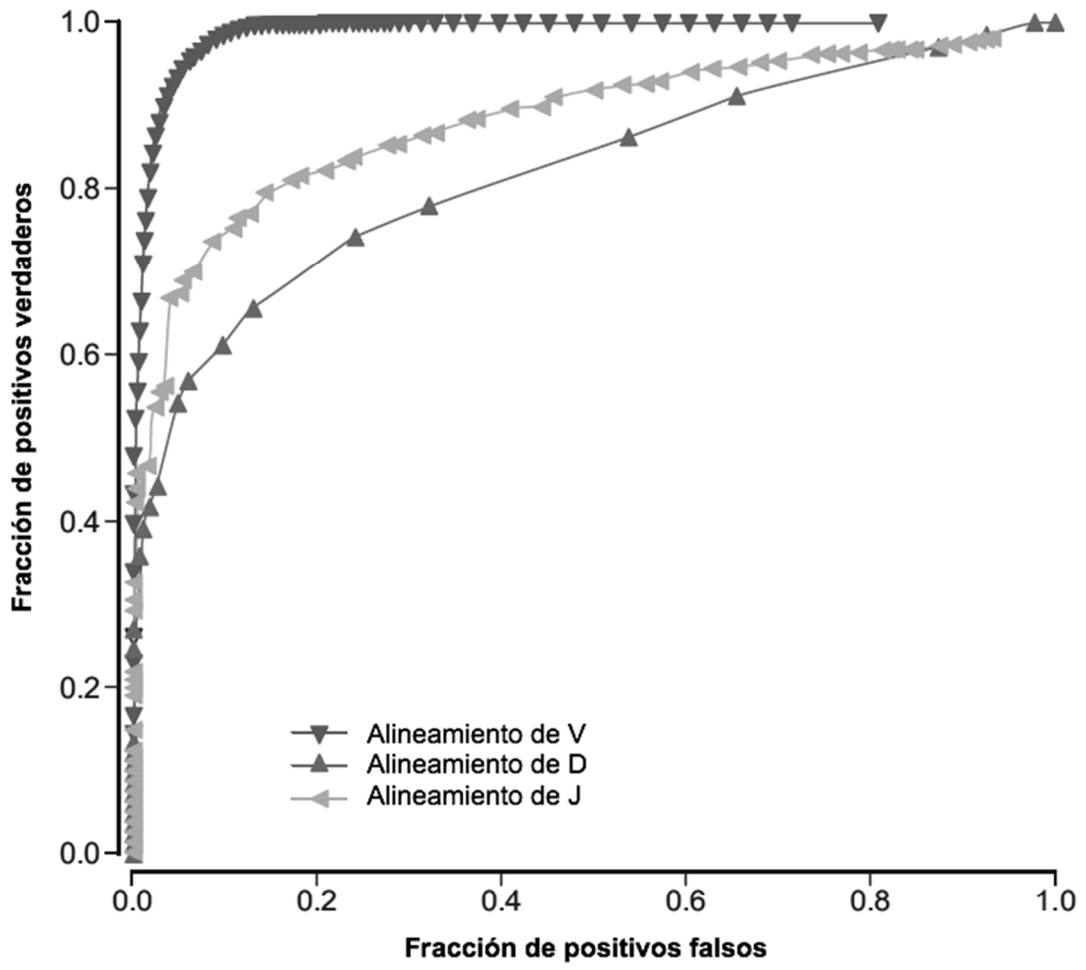


FIG. 8

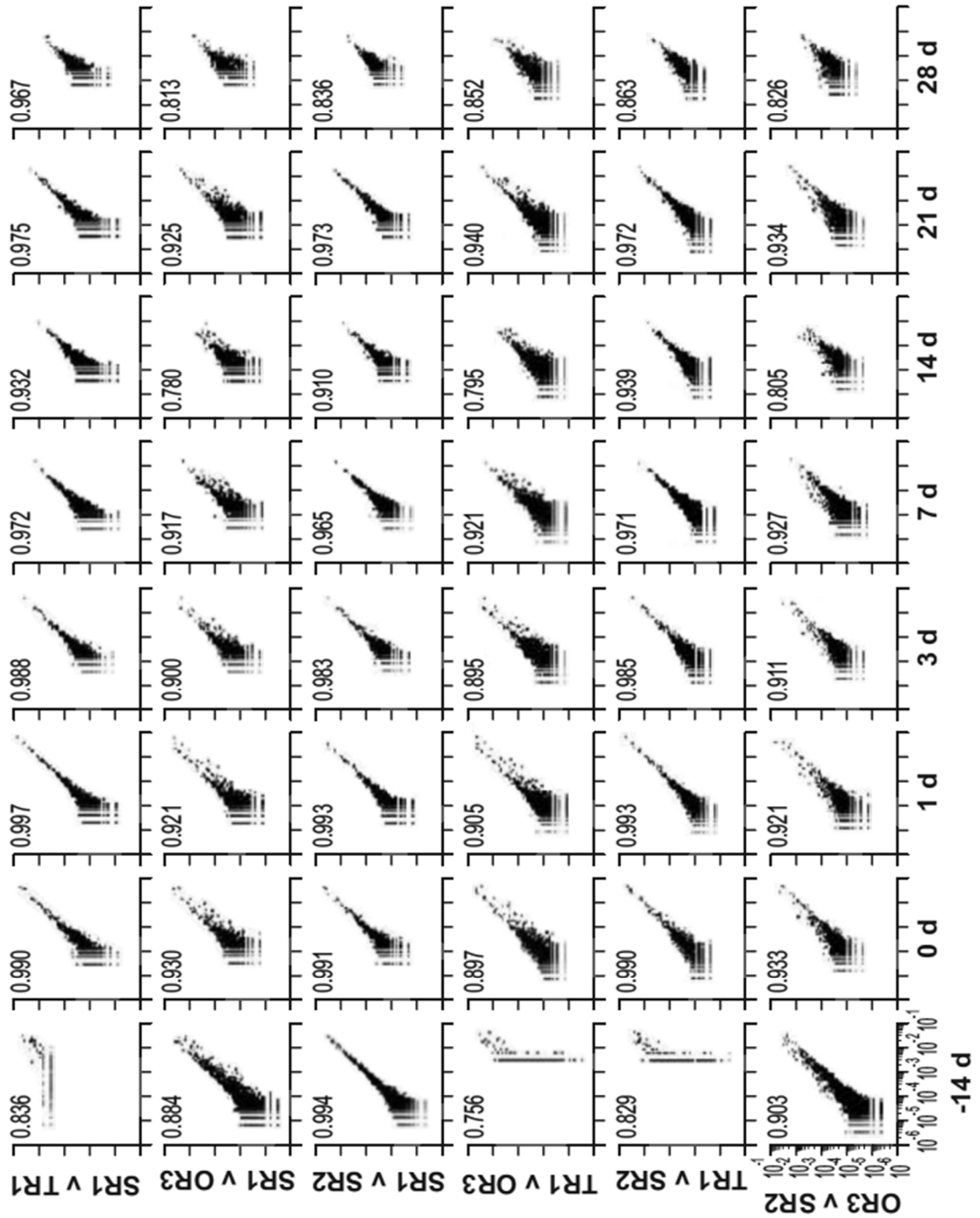


FIG. 9

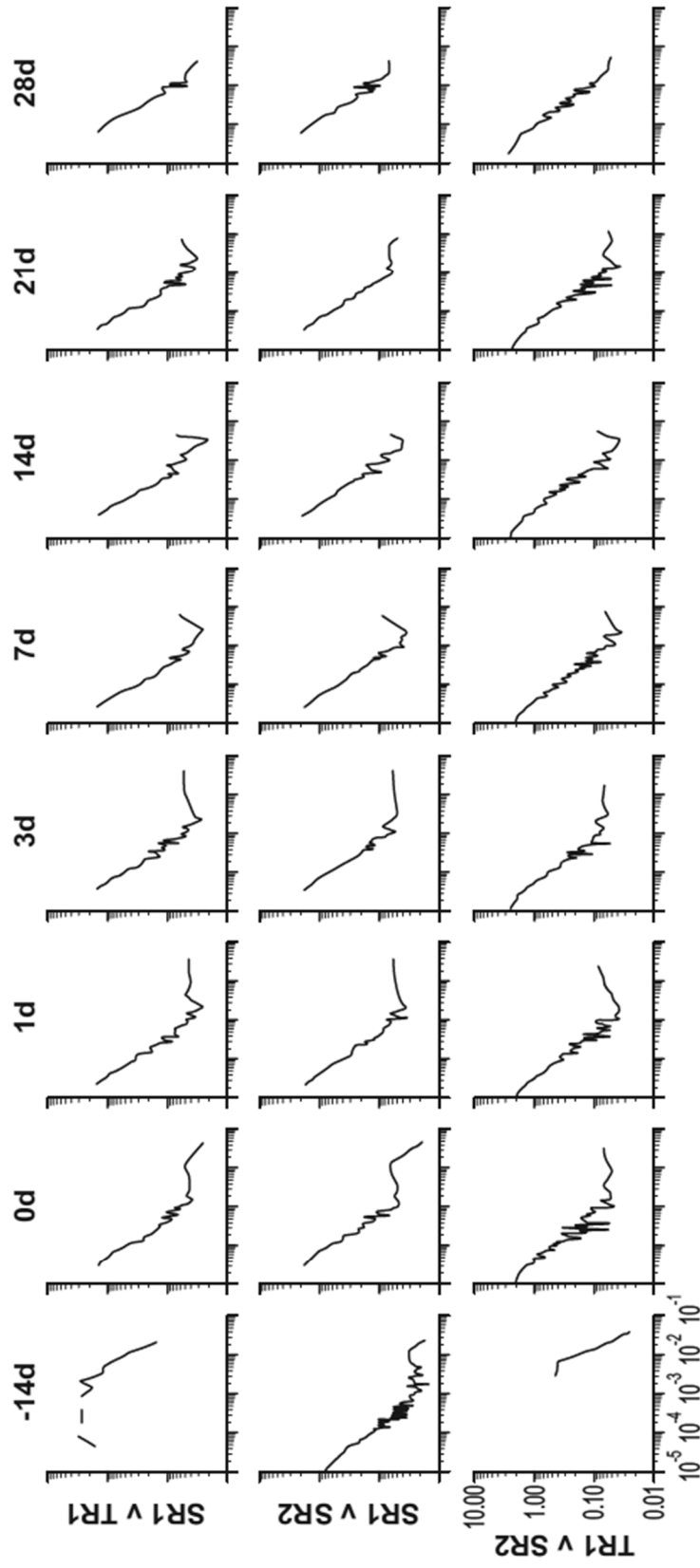


FIG. 10

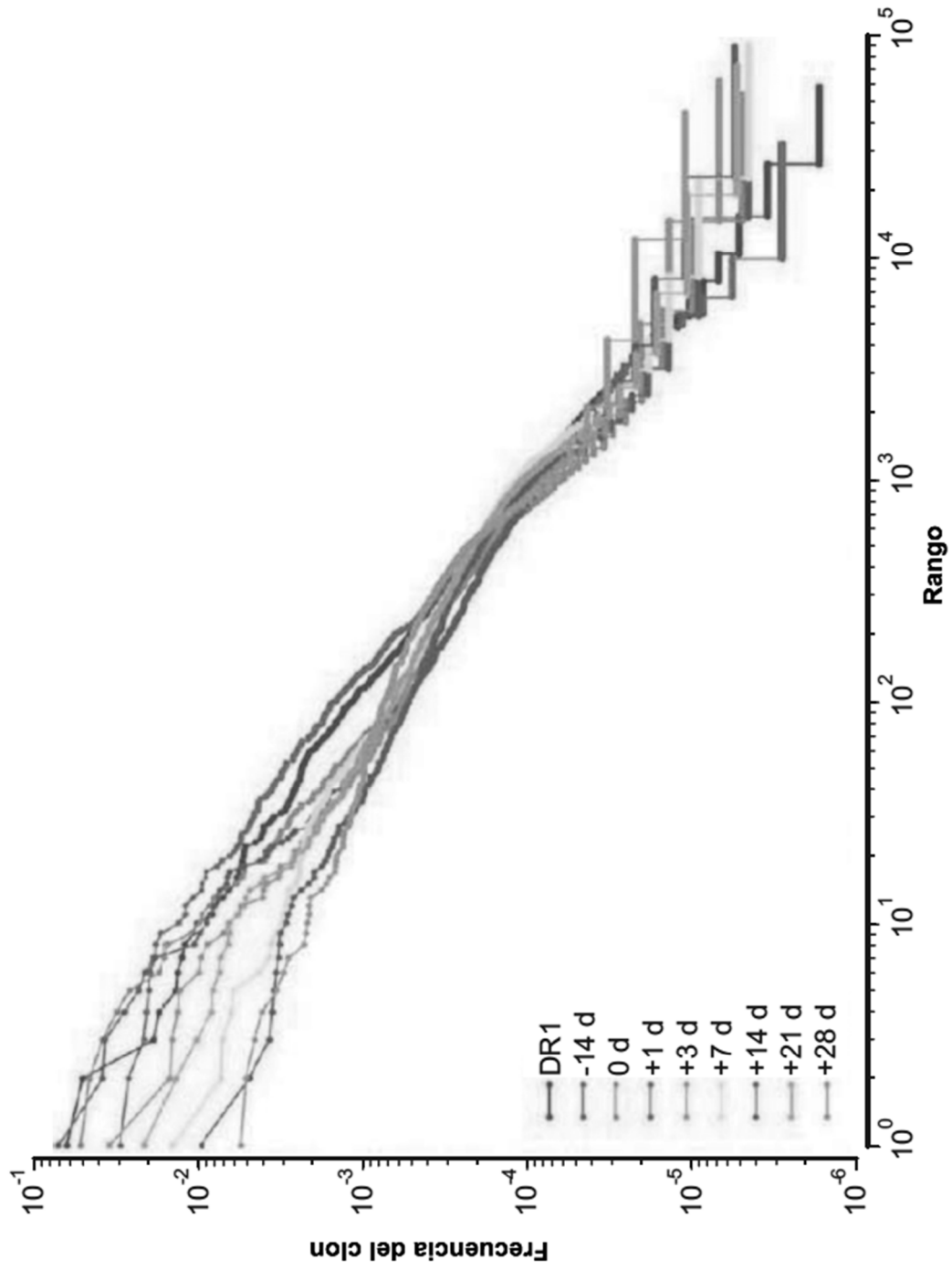


FIG. 11

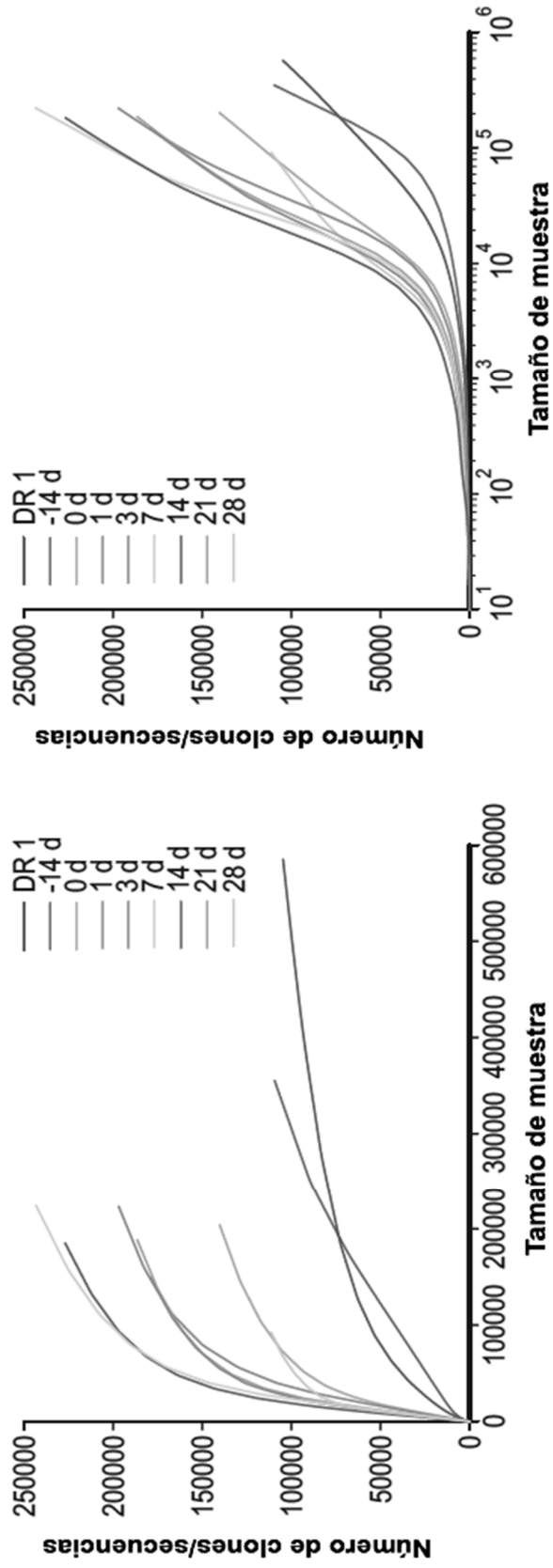


FIG. 12A

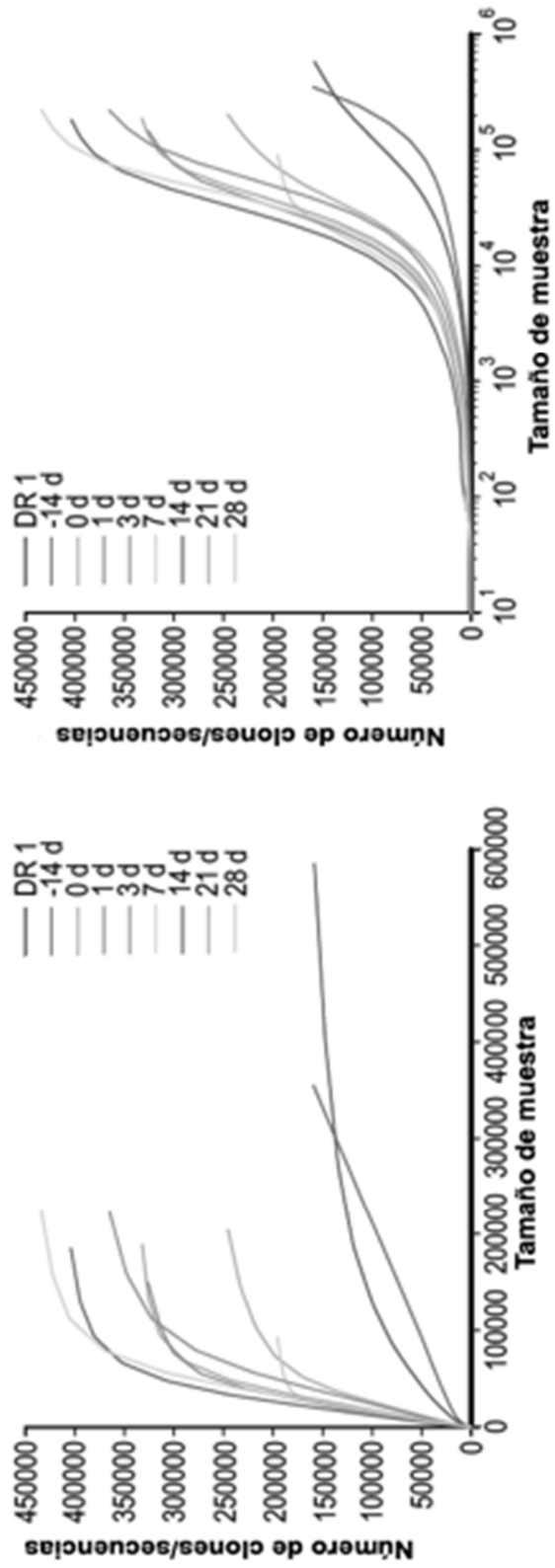


FIG. 12B

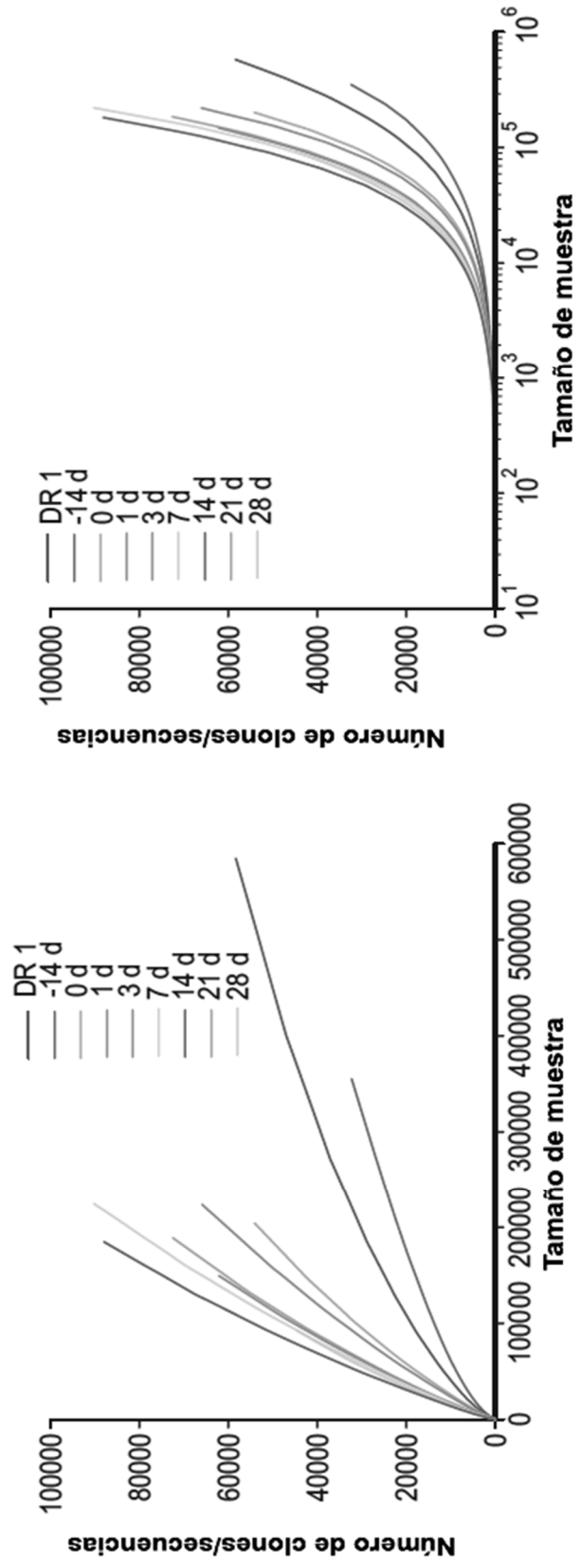


FIG. 12C

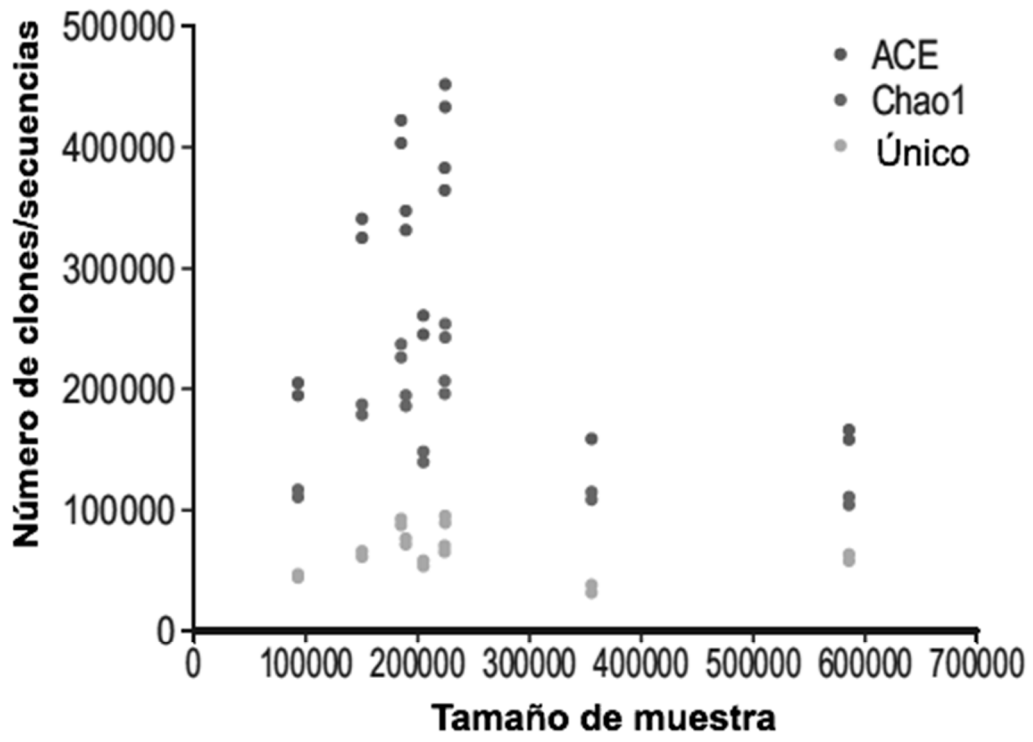


FIG. 12D

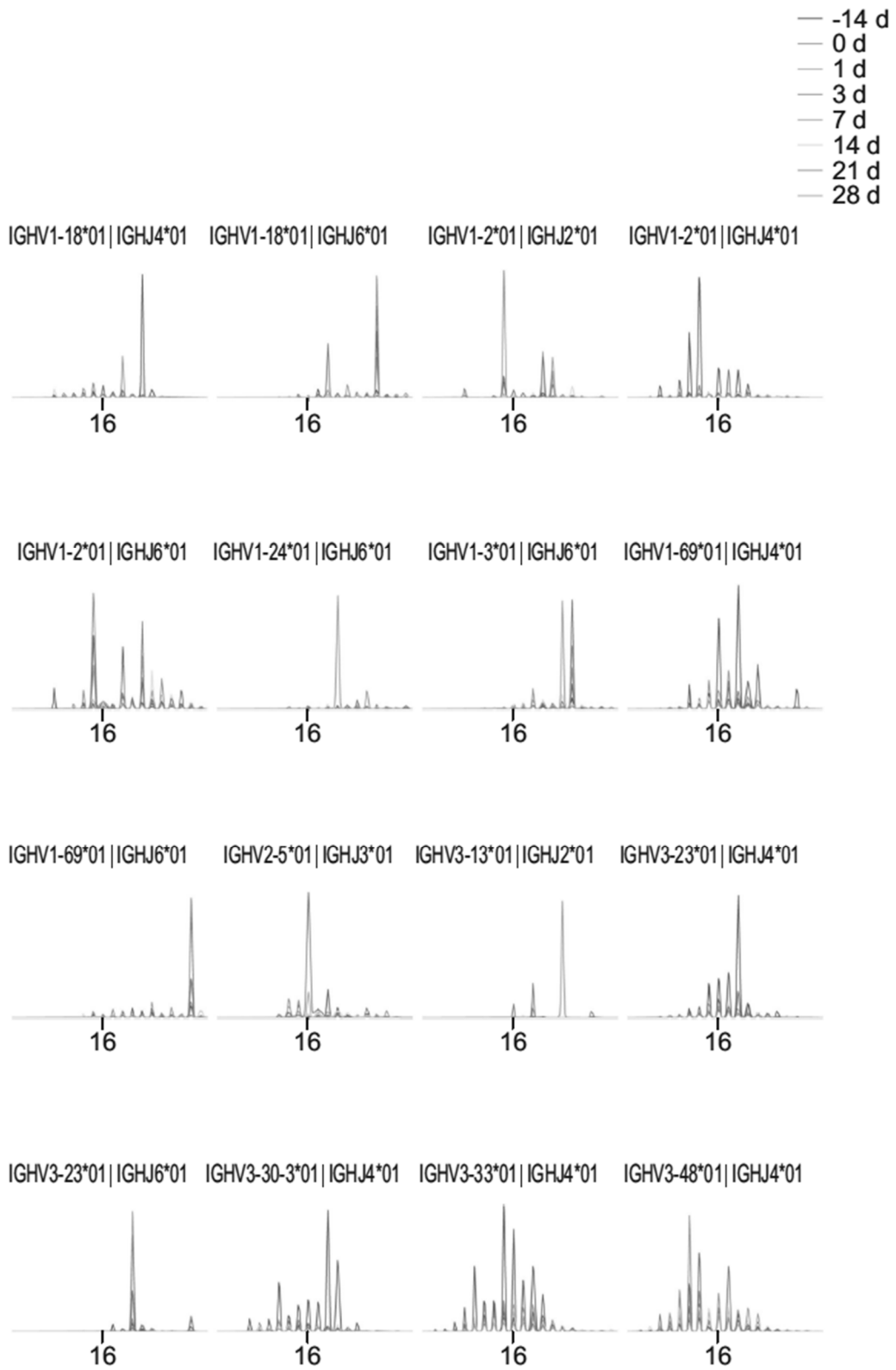


FIG. 13A

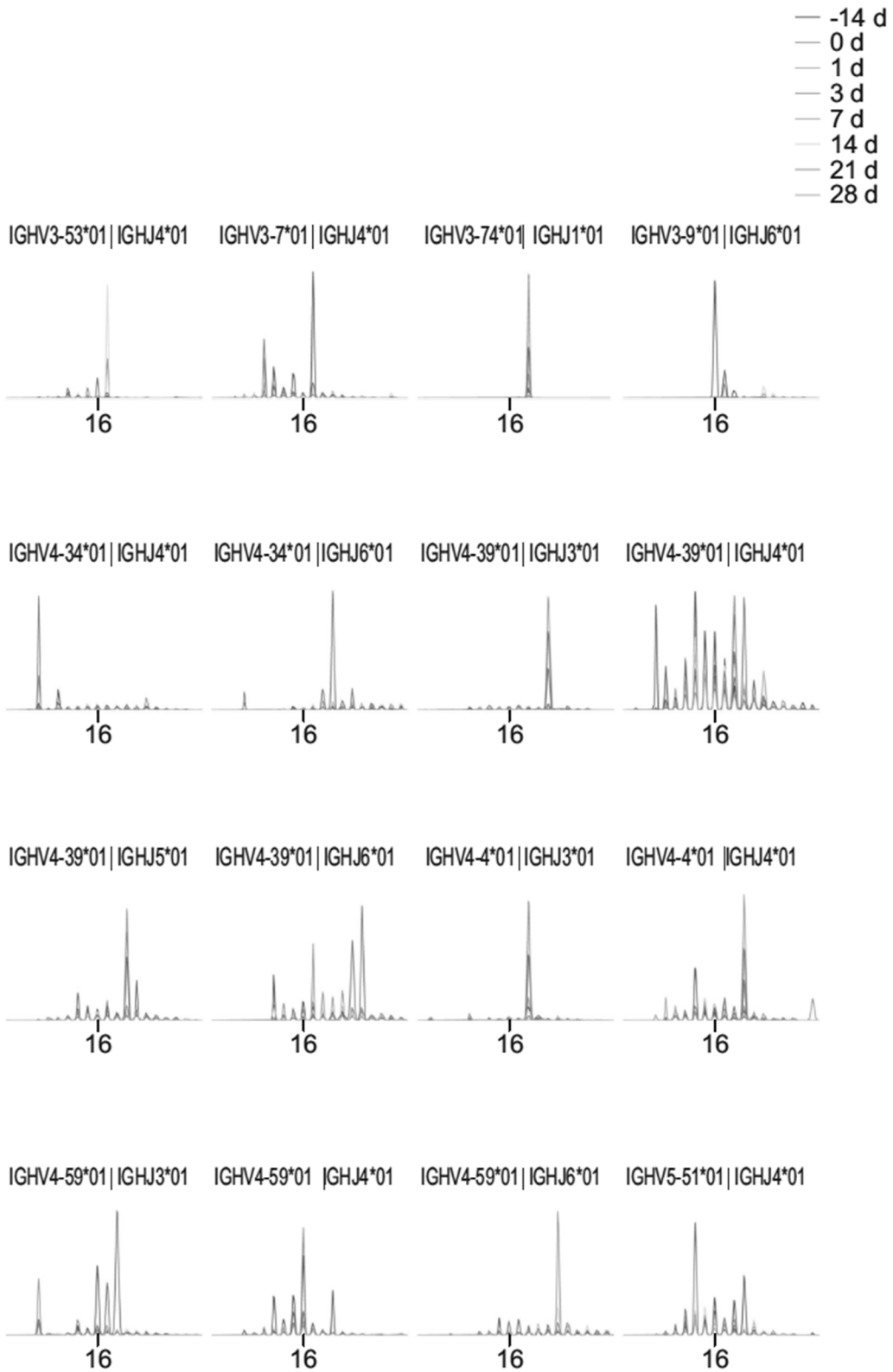


FIG. 13B

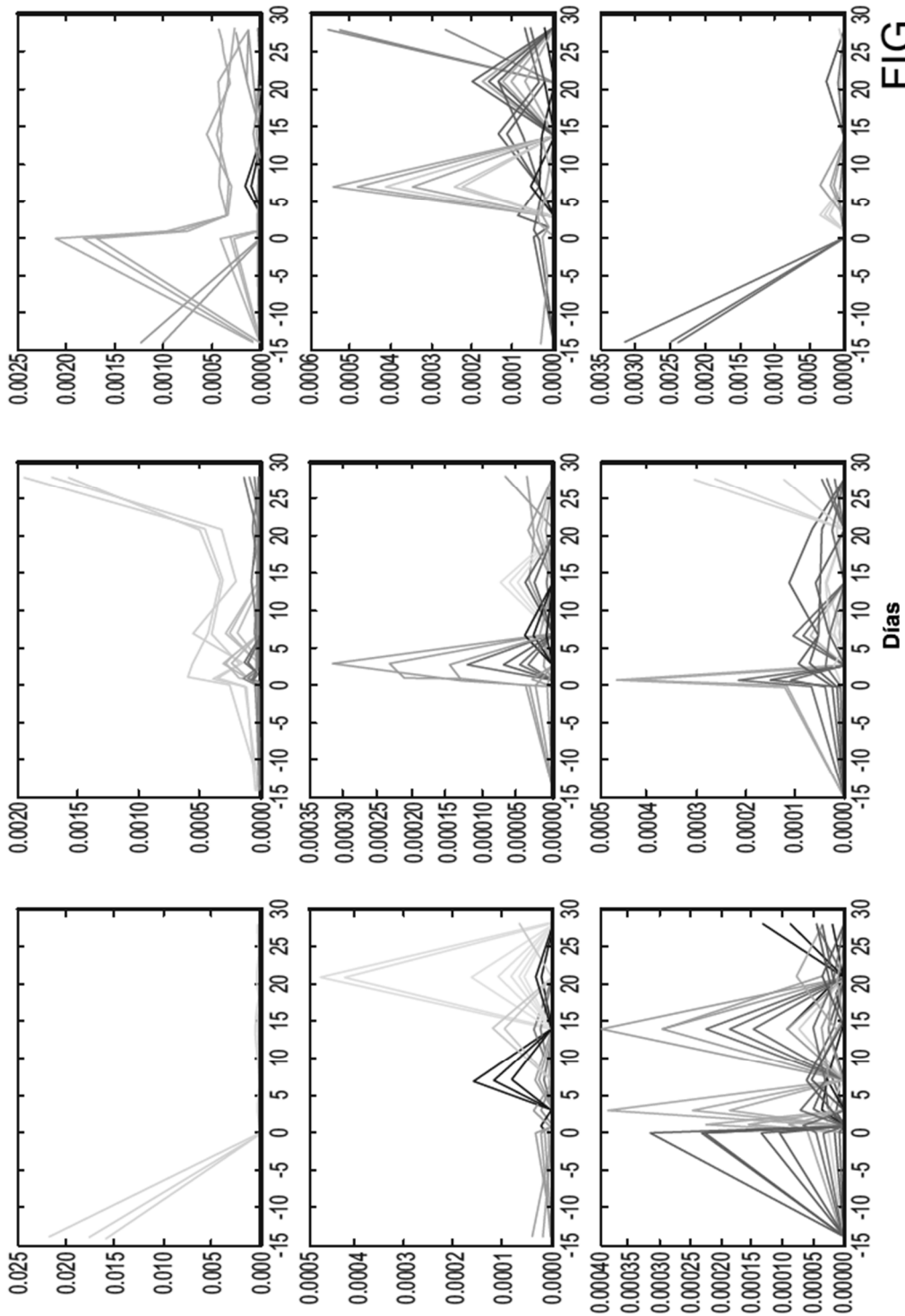


FIG. 14

FIG. 15A

medias de series K de tiempo (k=100);
 dist. euclidiana; Ejecución 4, 5, 7;
 clones vistos ≥ 2 veces; valores son
 frecuencias

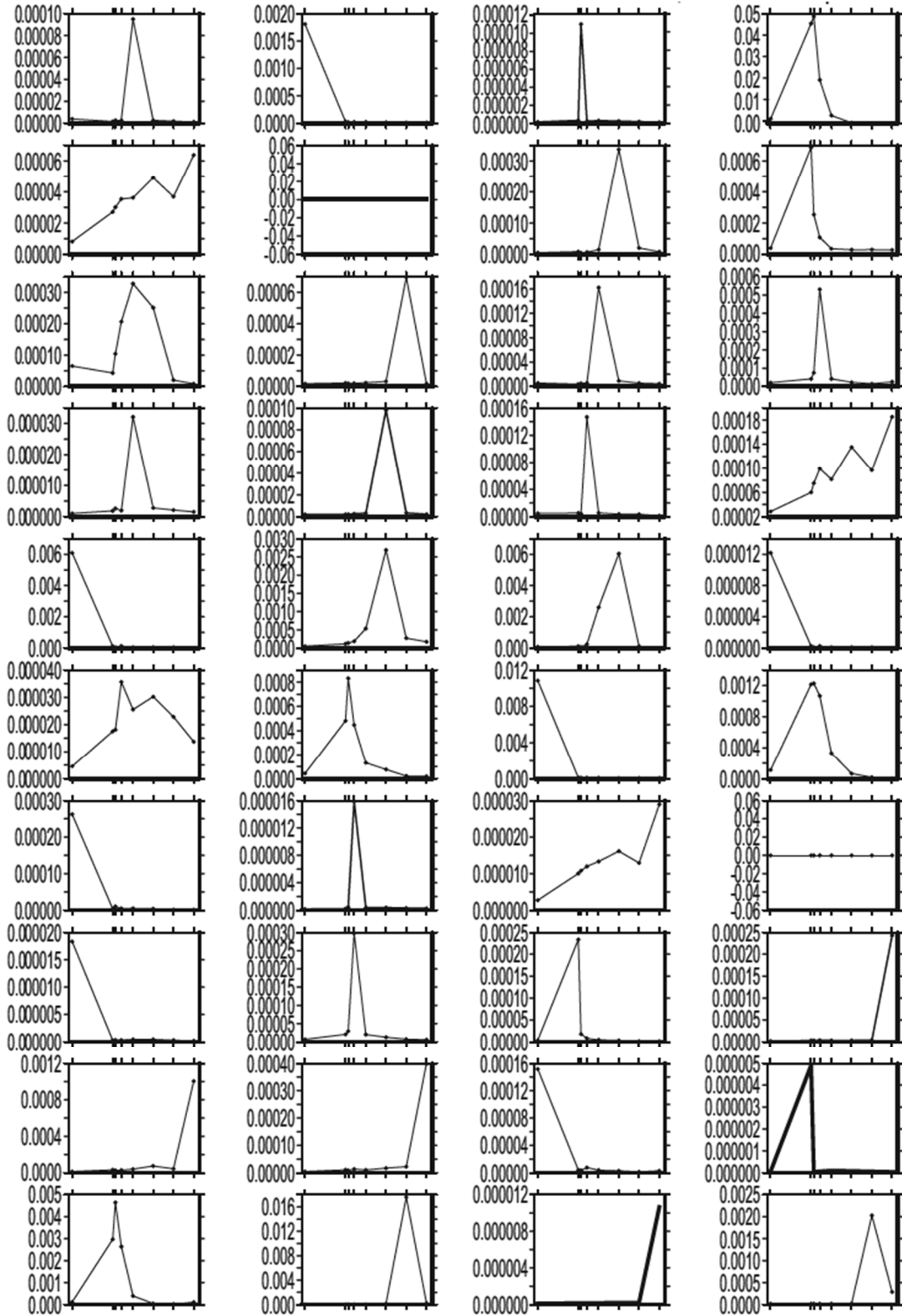


FIG. 15A Cont.

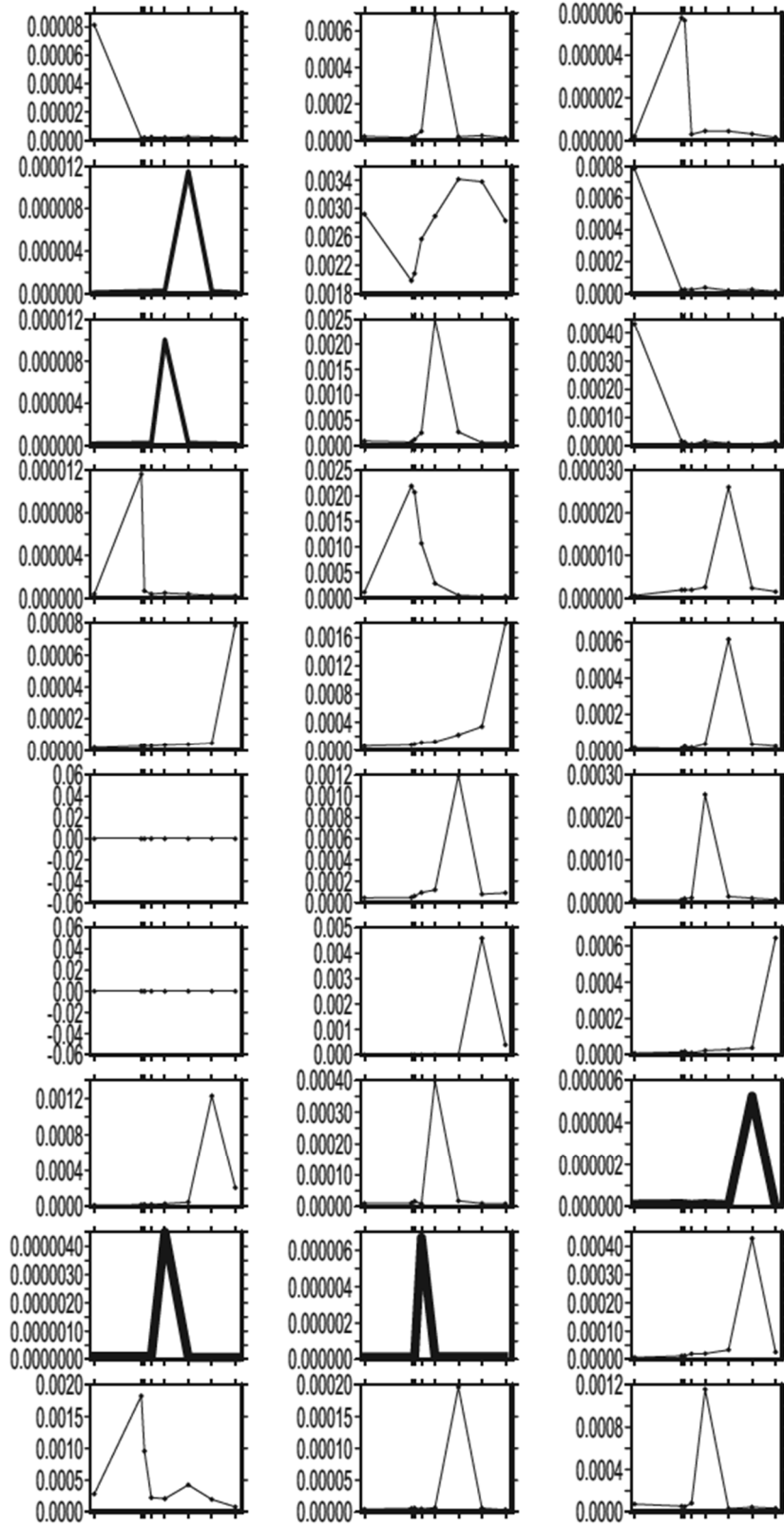


FIG. 15A Cont.

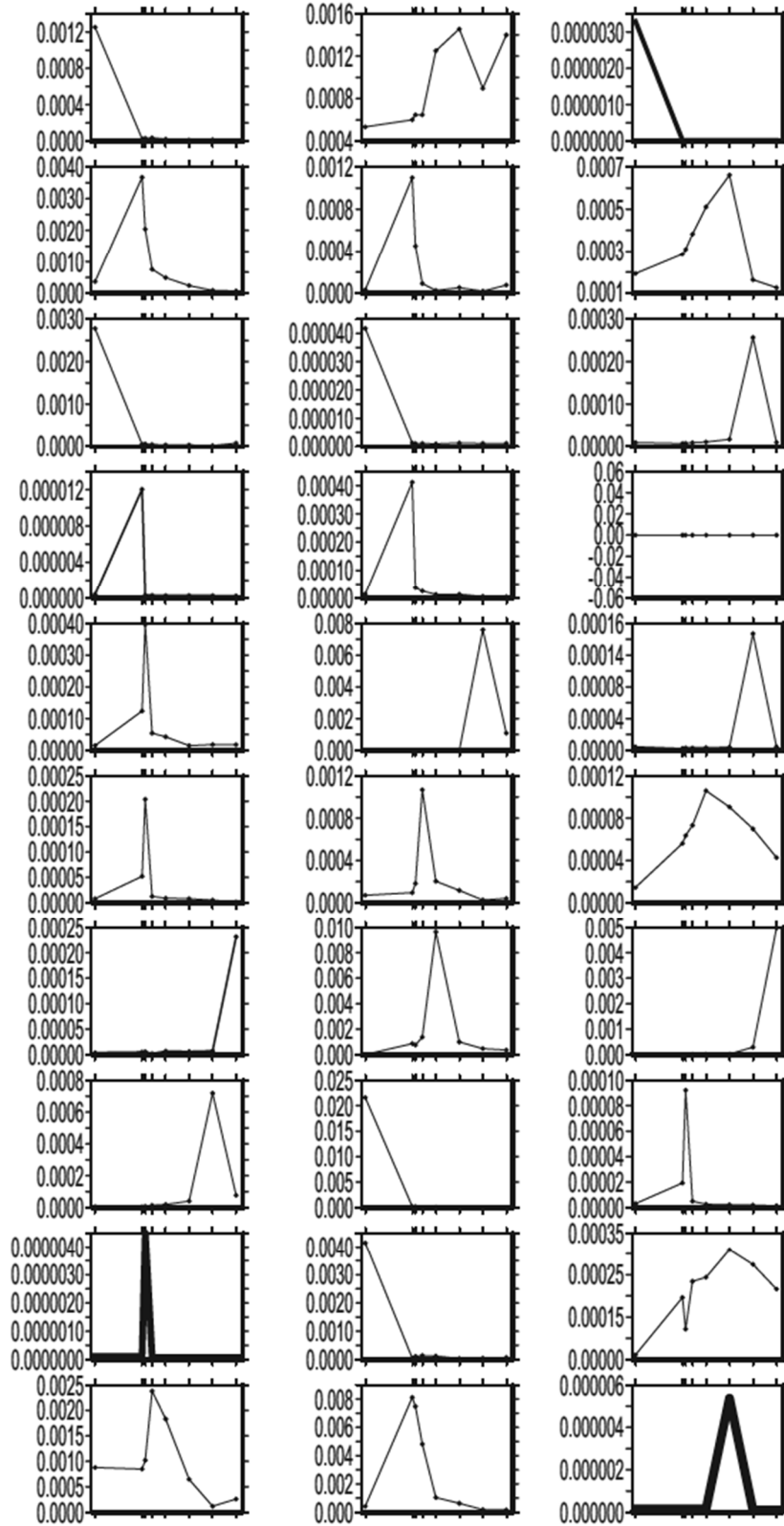


FIG. 15B

medias de series K de tiempo (k=100);
 dist. euclidiana; Ejecución 4, 5, 7;
 clones vistos ≥ 2 veces; valores son
 frecuencias

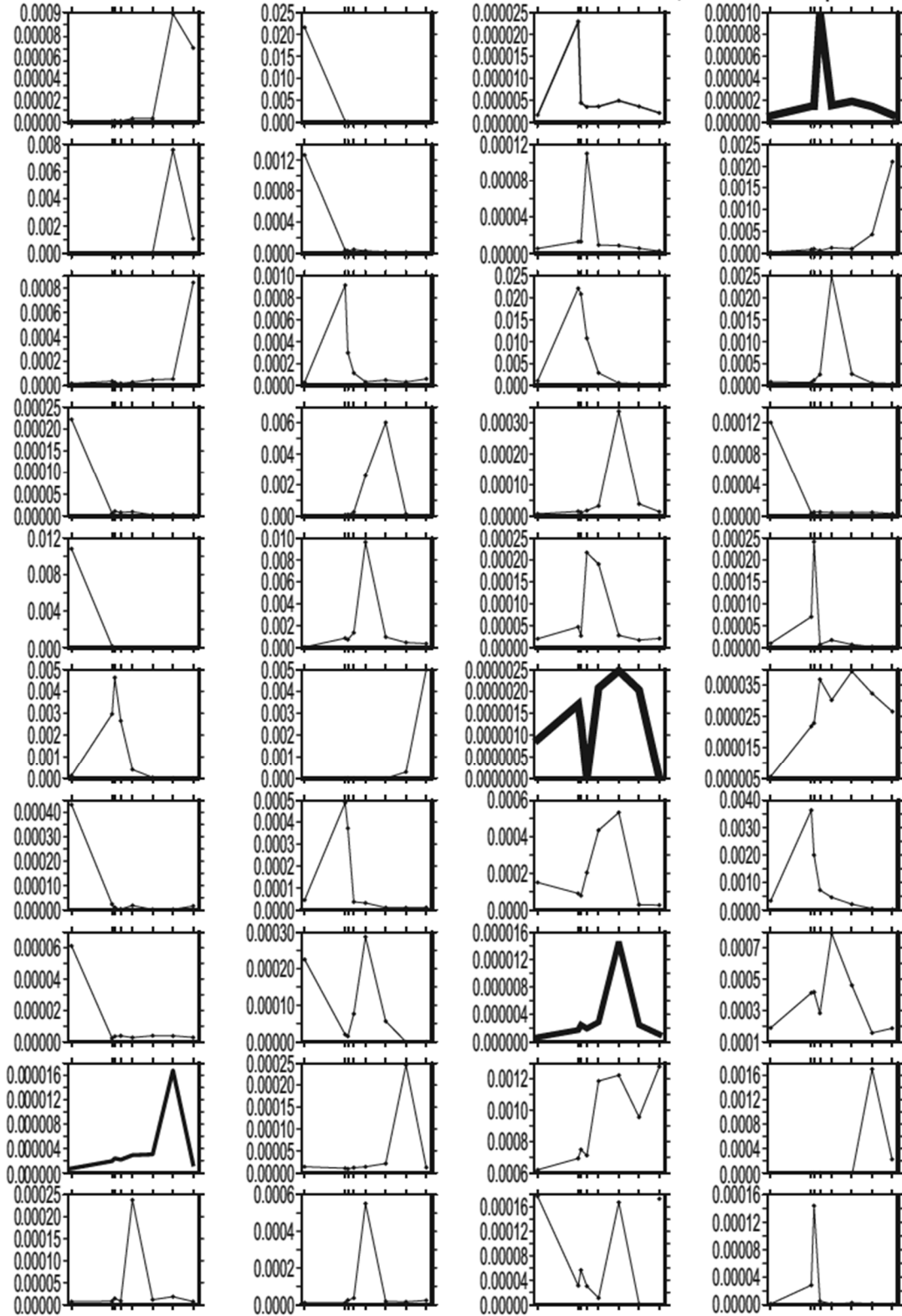


FIG. 15B Cont.

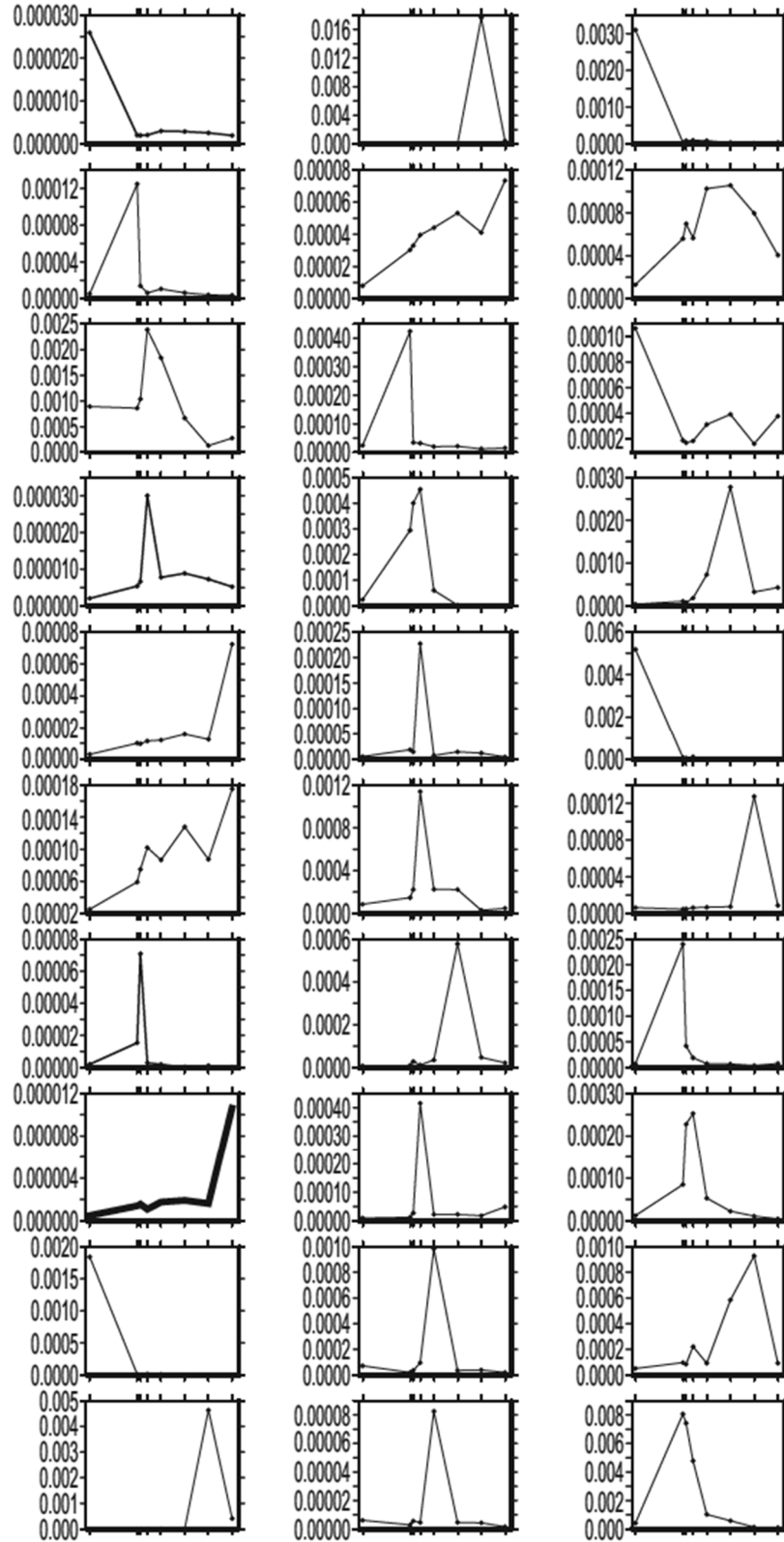


FIG. 15B Cont.

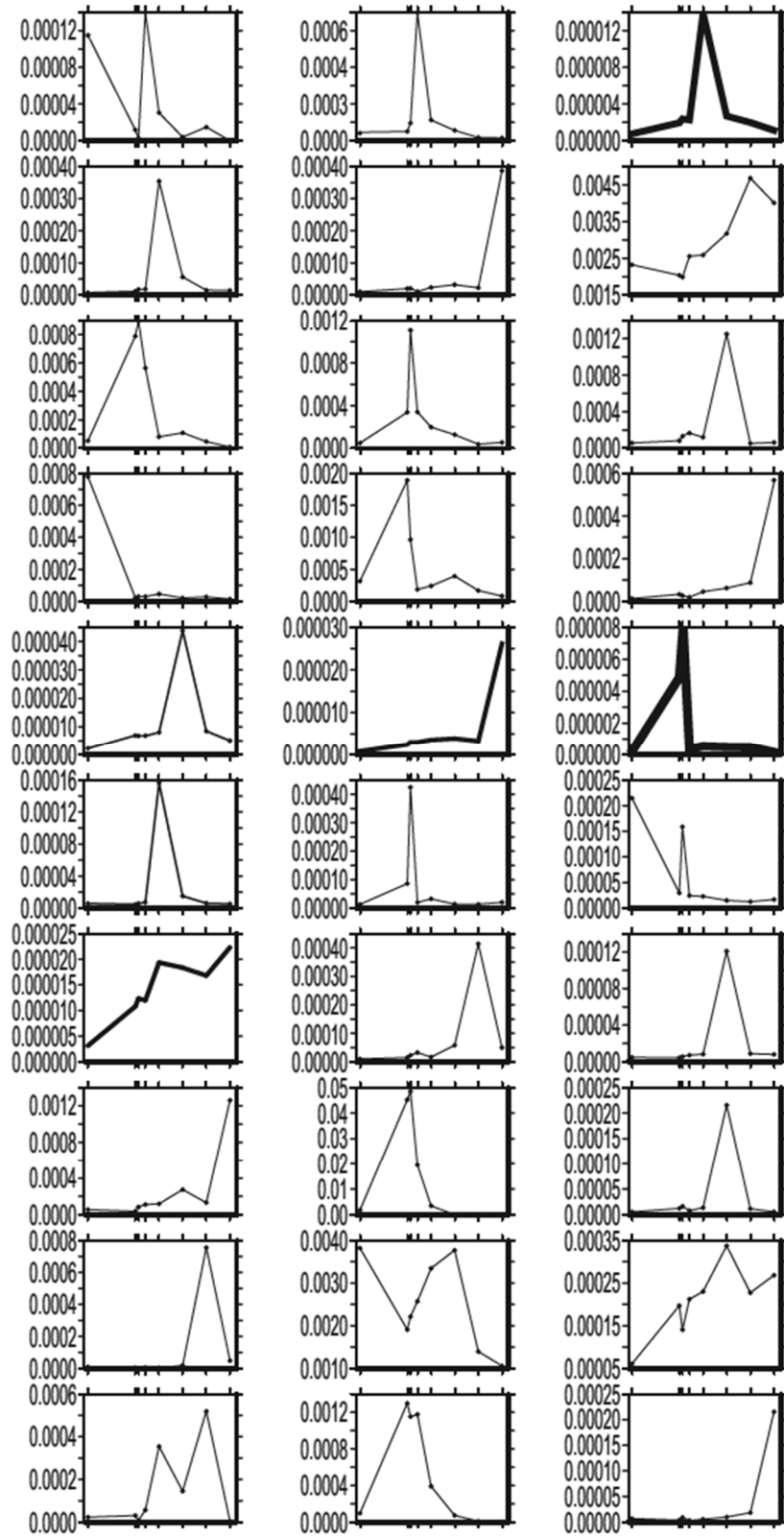


FIG. 16A

Día 1

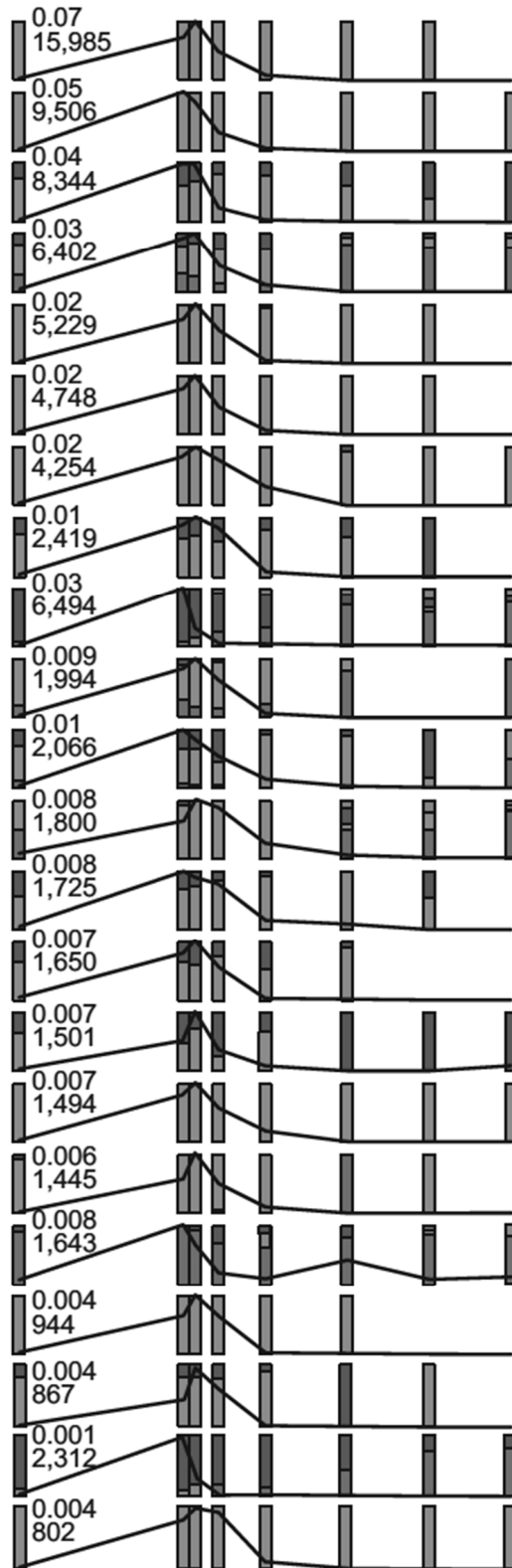


FIG. 16B

Día 7

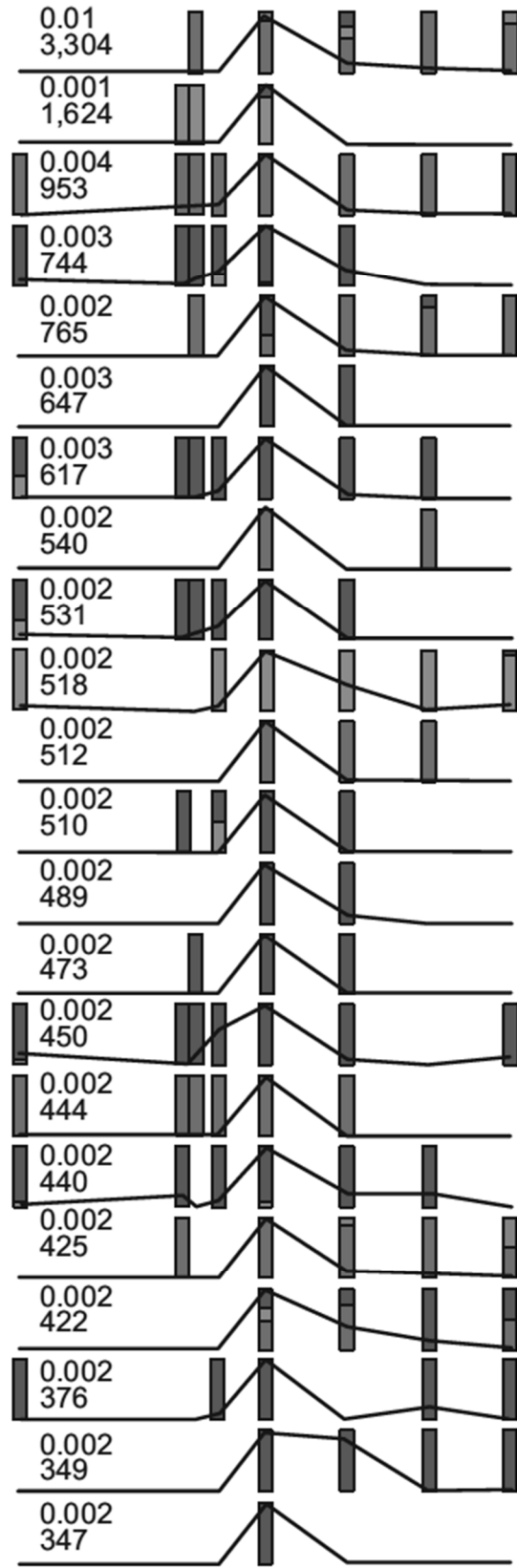
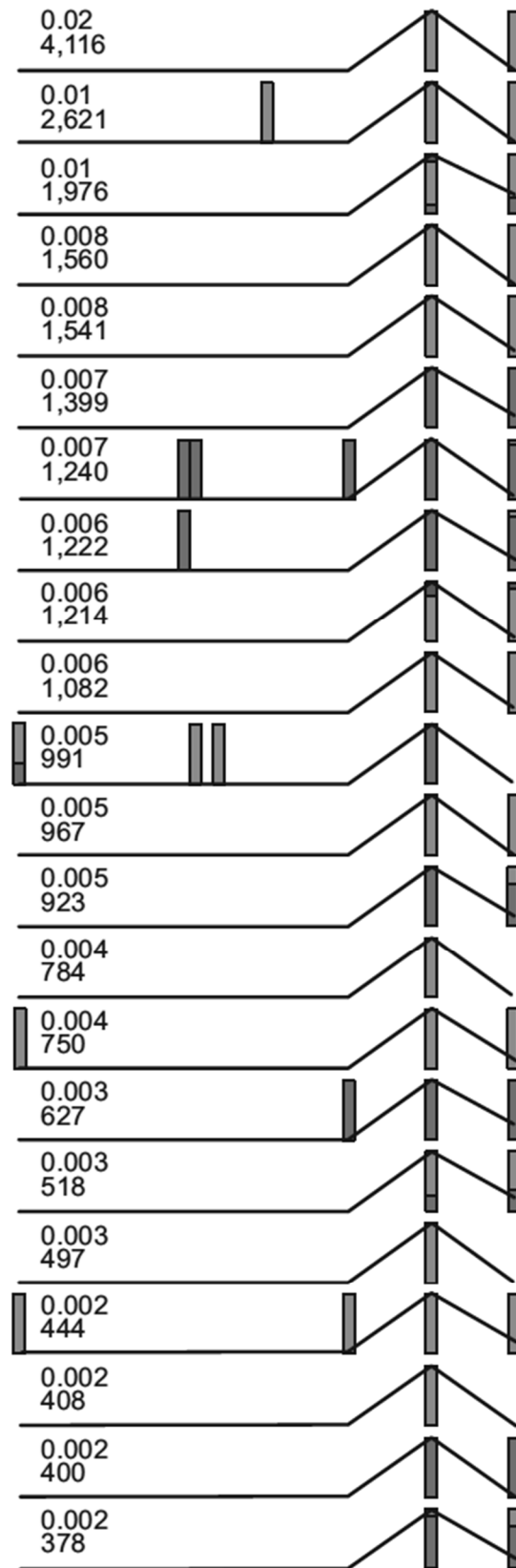


FIG. 16C

Día 21



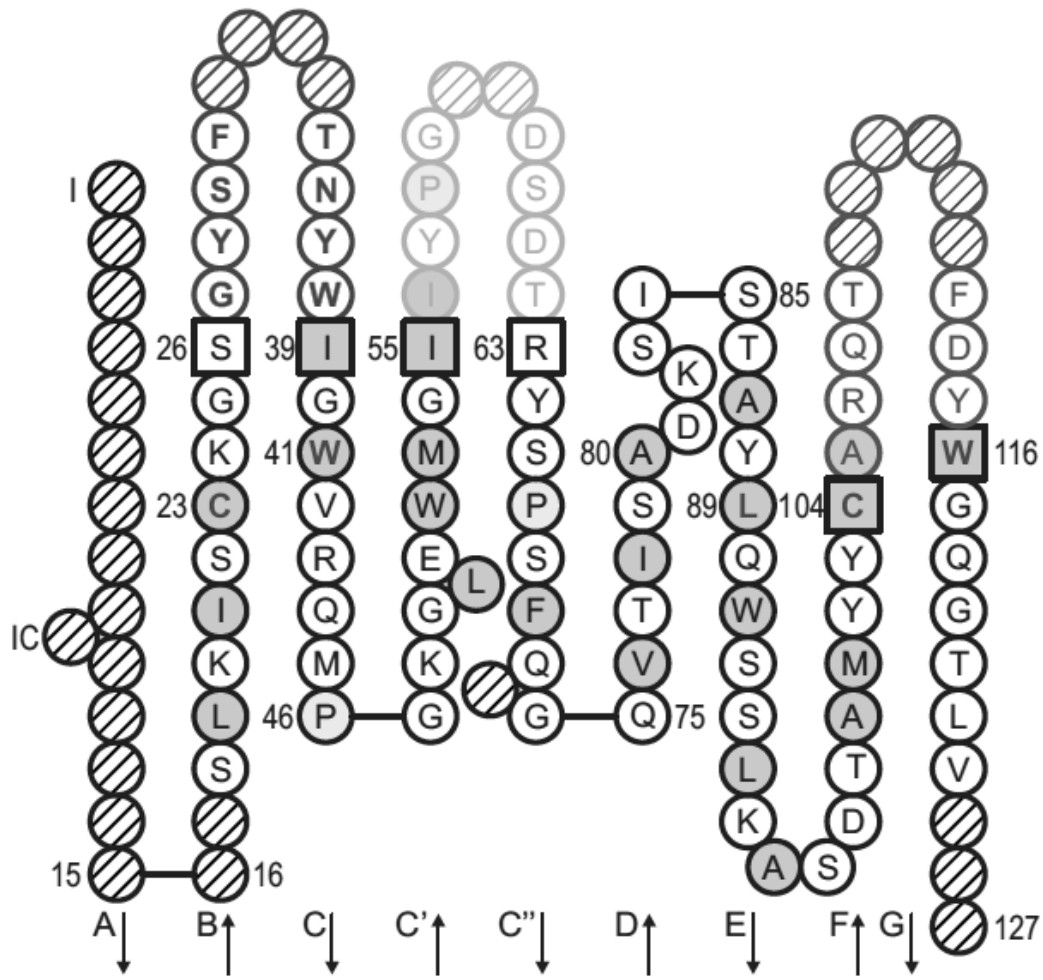


FIG. 17A

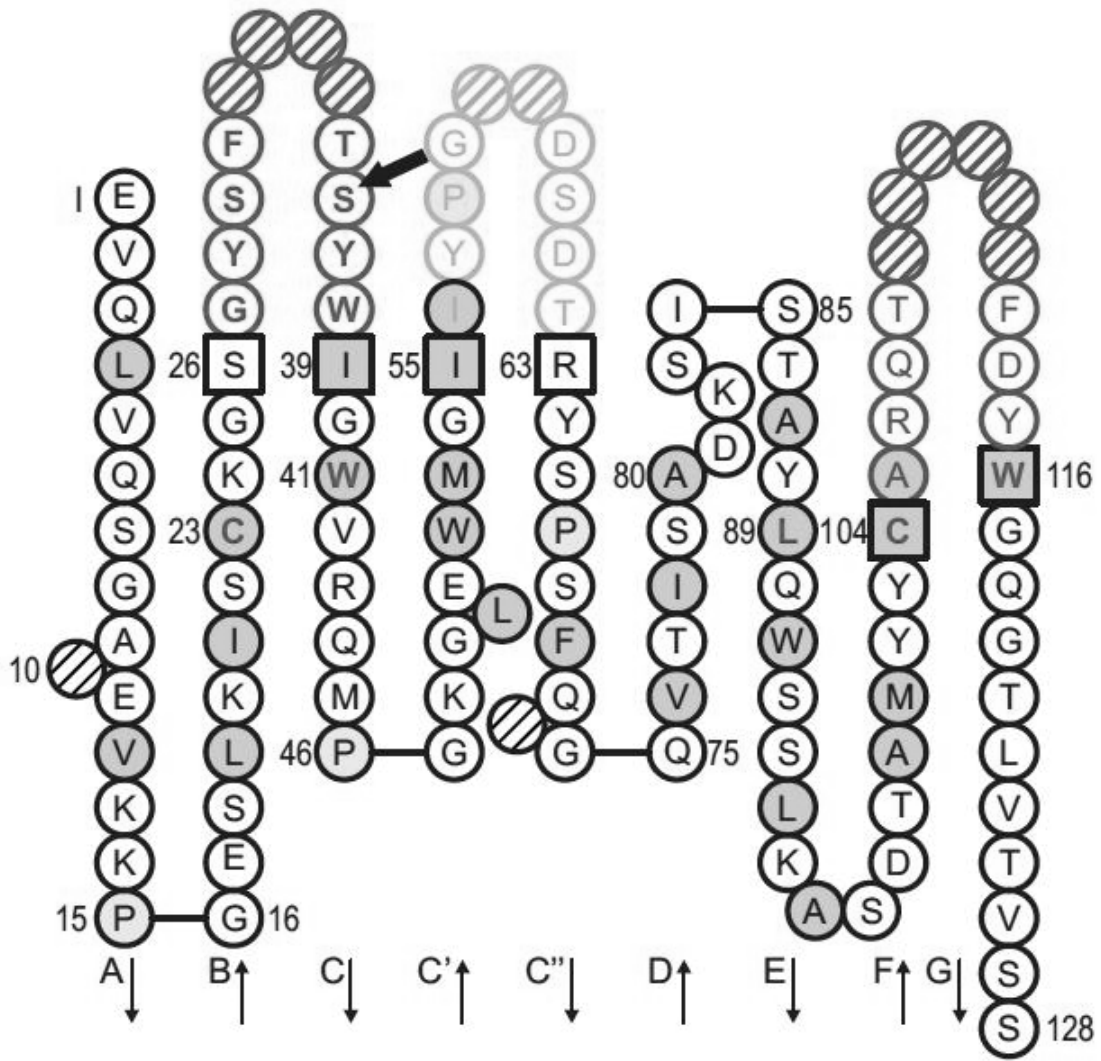


FIG. 17B

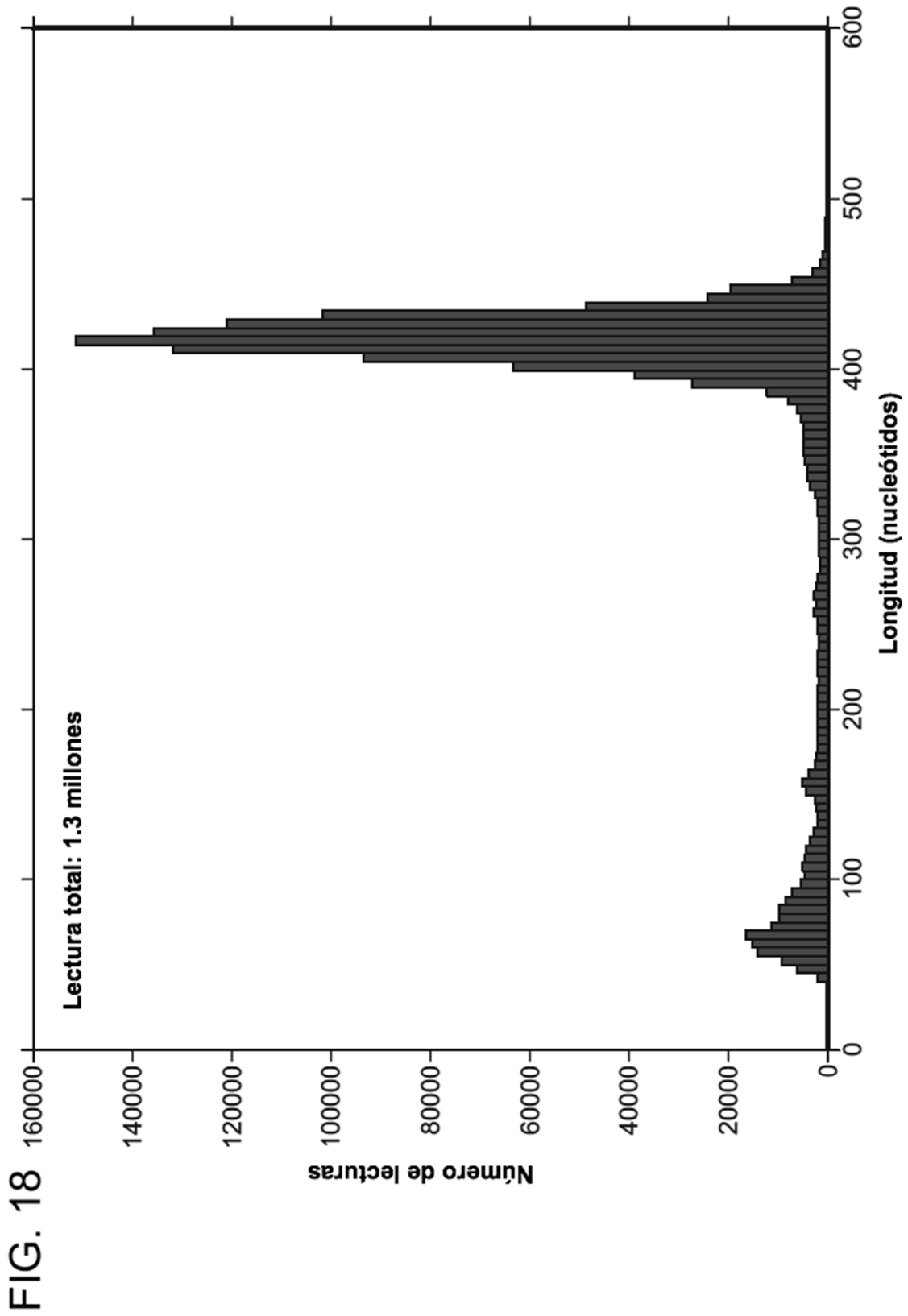


FIG. 19A

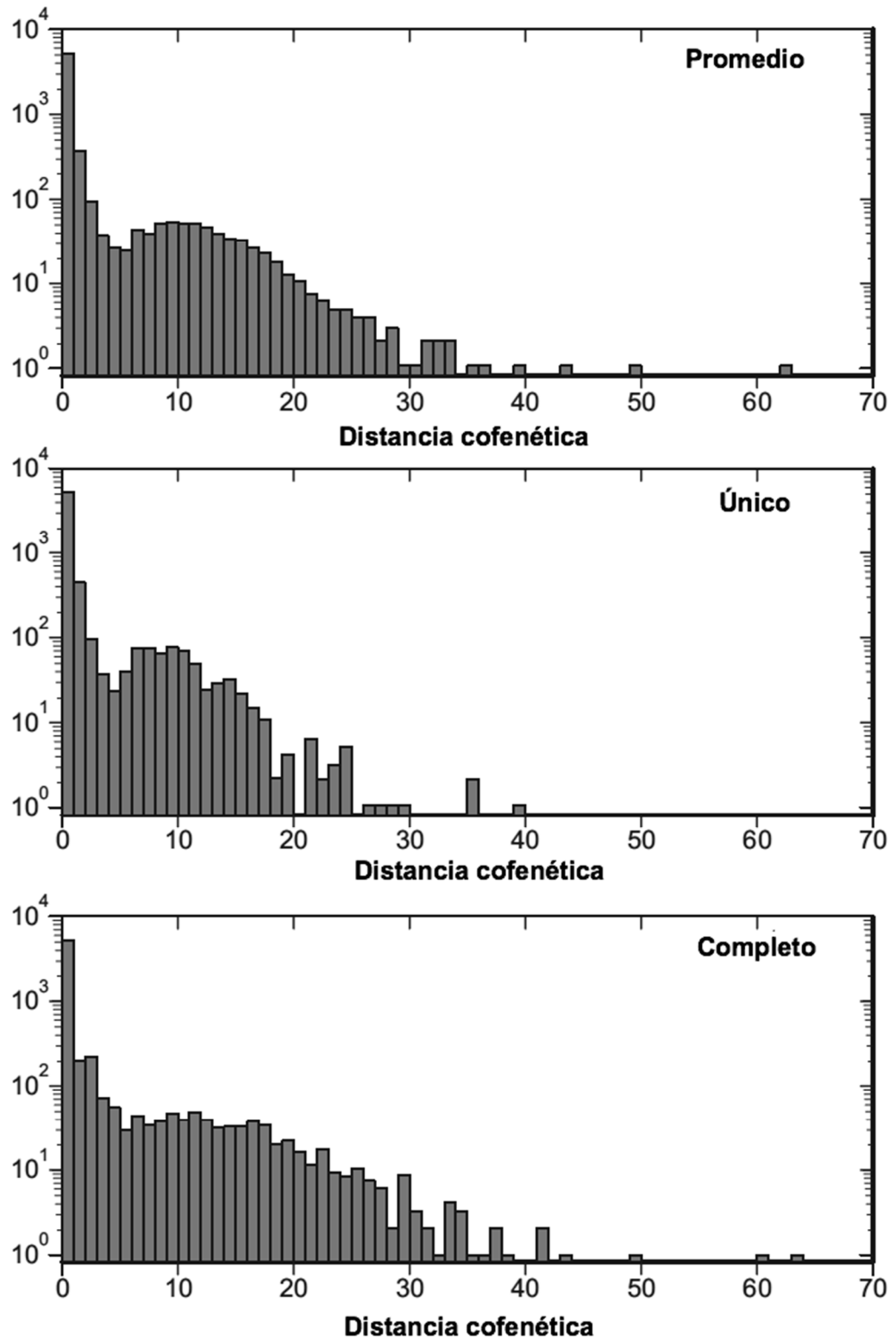


FIG. 19B

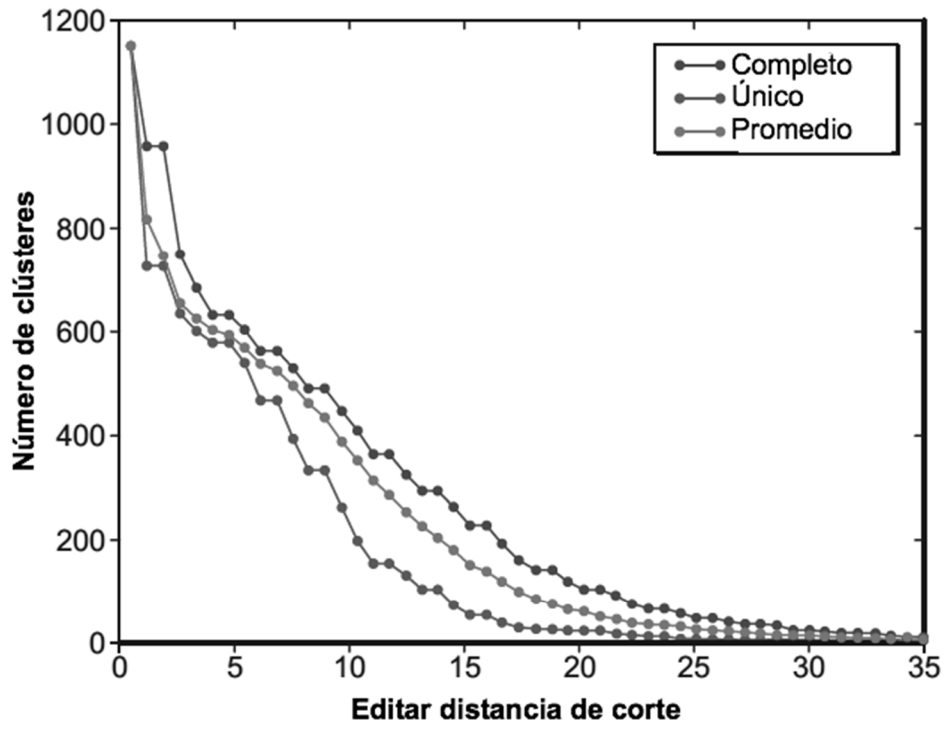
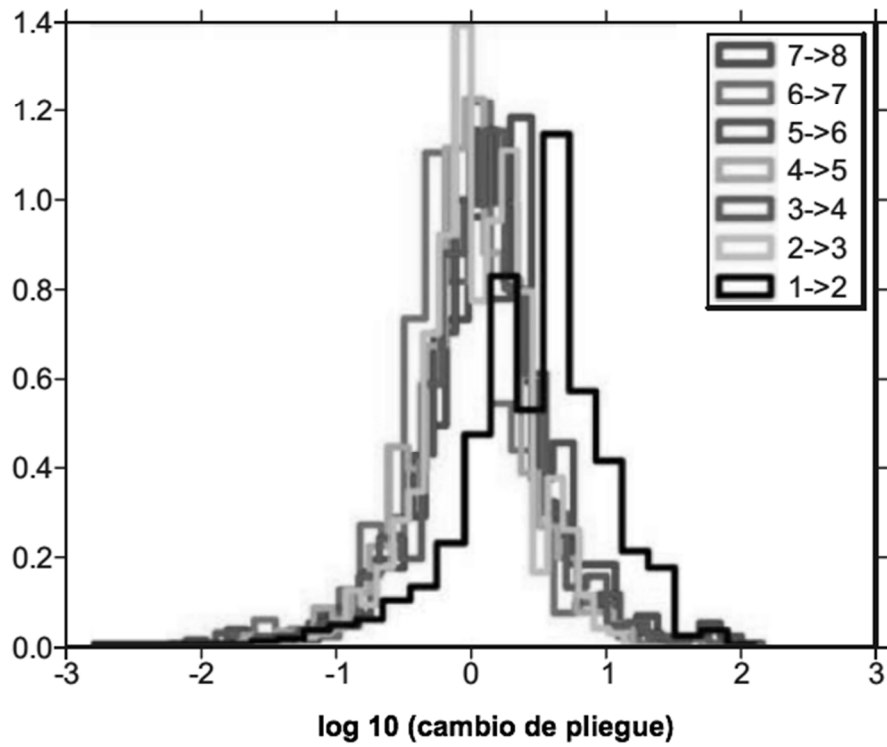


FIG. 20A



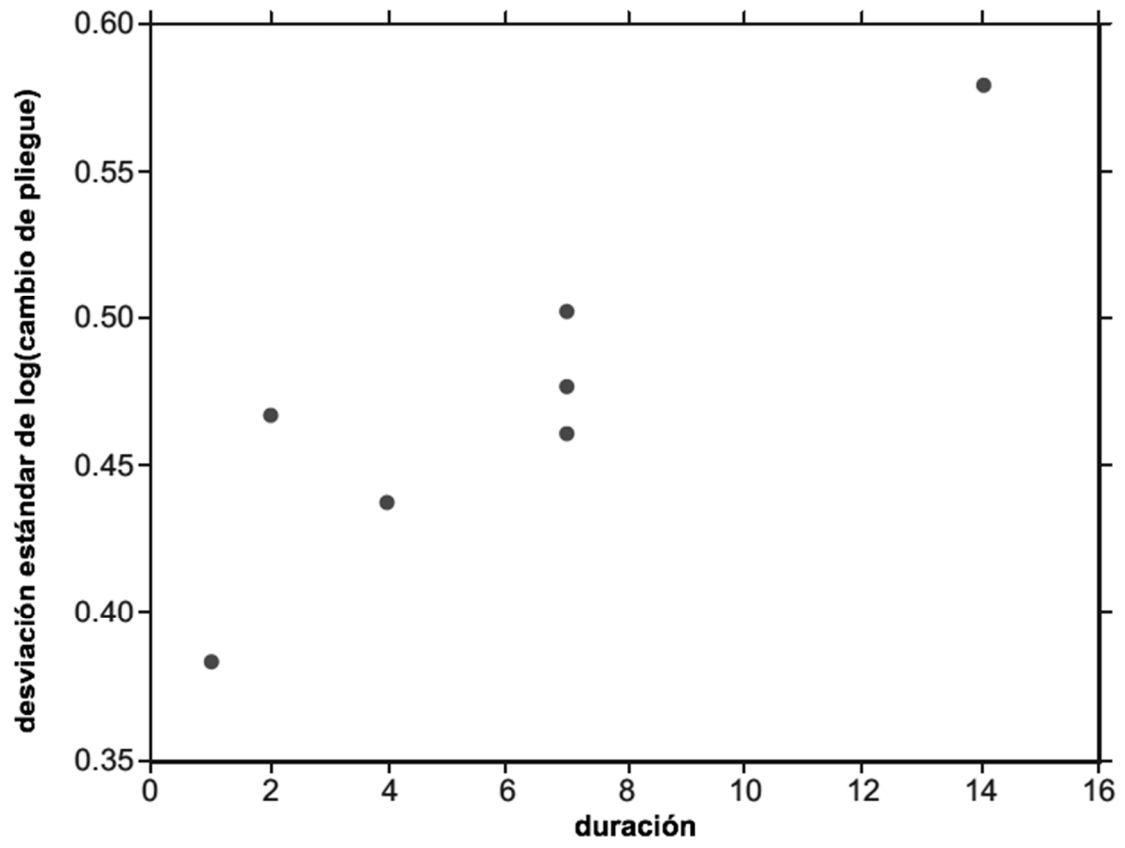


FIG. 20B

FIG. 23

```
1 : CysAlaArgGlnThrPheAspTyrTrp : 9 (SEQ ID NO:137)
    |||
    CysAlaArgGlnThrPheAspTyrTrp
238 : TGTGCGAGACAAACTTTGACTACTGG : 264 (SEQ ID NO:138)
```

FIG. 24

74 : TCTCTGAAGATCTCCTGTAAGGGTTCCTGGATACAGCTTTACCAGCTACTGGATCGGCTGGGTGC : 137
 SerLeuLysIleSerCysLysGlySerGlyTyrSerPheThrSerTyrTrpIleGlyTrpVala
 |||
 SerLeuLysIleSerCysLysGlySerGlyTyrSerPheThrAsnTyrTrpIleGlyTrpVala
 1 : TCTCTGAAGATCTCCTGTAAGGGTTCCTGGATACAGCTTTACCAGCTACTGGATCGGCTGGGTGC : 64

 138 : GCCAGATGCCCGGAAAGCCCTGGAGTGGATGGGGATCATCTATCCTGGTACTCTGATACCCAG : 200
 rgGlnMetProGlyLysGlyLeuGluTrpMetGlyIleIleTyrProGlyAspSerAspThrAr
 |||
 rgGlnMetProGlyLysGlyLeuGluTrpMetGlyIleIleTyrProGlyAspSerAspThrAr
 65 : GCCAGATGCCCGGAAAGCCCTGGAGTGGATGGGGATCATCTATCCTGGTACTCTGATACCCAG : 127

 201 : ATACAGCCCGTCCCTCCAAGGCCAGGTCACCATCTCAGCCGACAAAGTCCATCAGCACCCGCTAC : 263
 gTyrSerProSerPheGlnGlyGlnValThrIleSerAlaAspLysSerIleSerThrAlaTyr
 |||
 gTyrSerProSerPheGlnGlyGlnValThrIleSerAlaAspLysSerIleSerThrAlaTyr
 128 : ATACAGCCCGTCCCTCCAAGGCCAGGTCACCATCTCAGCCGACAAAGTCCATCAGCACCCGCTAC : 190

 264 : CTGCAGTGGAGCAGCCCTGAAGGCCCTCGGACACCCGCCATGTATTACTGTGCGAGACAGACTTTTG : 329
 LeuGlnTrpSerSerLeuLysAlaSerAspThrAlaMetTyrTyrCysAlaArgGlnThrPhea
 |||
 LeuGlnTrpSerSerLeuLysAlaSerAspThrAlaMetTyrTyrCysAlaArgGlnThrPhea
 191 : CTGCAGTGGAGCAGCCCTGAAGGCCCTCGGACACCCGCCATGTATTACTGTGCGAGACAAACTTTTG : 256

 330 : ACTACTGGGGCCAGGGAACCCCTGGTC : 355 (SEQ ID NO:139)
 spTyrTrpGlyGlnGlyThrLeuVal
 |||
 spTyrTrpGlyGlnGlyThrLeuVal
 257 : ACTACTGGGGCCAGGGAACCCCTGGTC : 282 (SEQ ID NO:140)

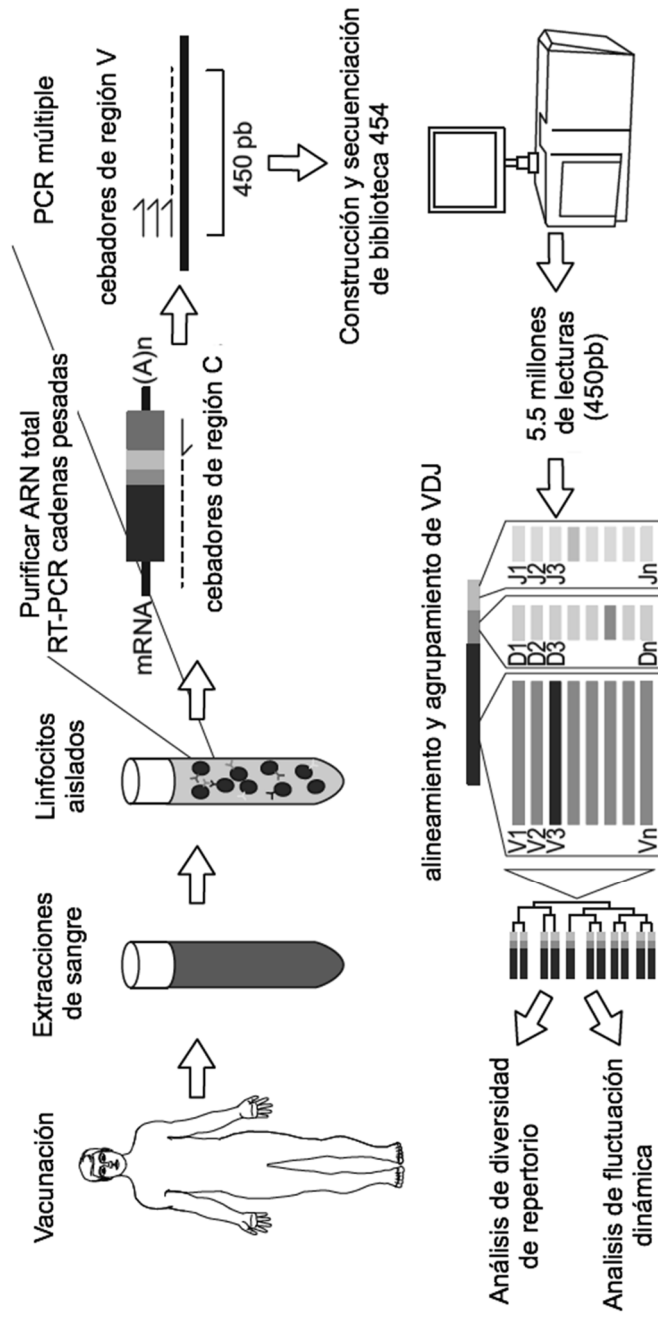


FIG. 25

FIG. 26

	DR1	%	OR1	%	OR2	%	SR1	%	TR1	%	OR3	%	SR2	%	Total
Lecturas totales	1,183,769	100	455,187	100	1,147,232	100	622,356	100	1,473,502	100	1,268,232	100	1,004,209	100	7,154,487
Lecturas filtradas por tamaño	793,976	67	396,339	87	994,138	87	491,747	79	1,087,398	74	1,021,570	81	709,138	71	5,494,306
Núm. con código de barras	783,514	66	388,467	85	978,201	85	470,122	76	1,038,683	70	947,307	75	676,660	67	5,282,954
Núm. con BC e isotipo	549,751	46	264,802	58	675,828	59	271,529	44	517,539	35	618,670	49	434,549	43	3,332,668
Núm. con BC, isotipo, y VJ ain	411,370	35	214,338	4	546,118	48	218,622	35	423,888	29	486,541	38	350,936	35	2,651,813
Núm. con BC, VJ ain	578,114	49	310,792	68	785,941	69	354,136	57	743,527	50	727,521	57	527,222	53	4,027,253
código de barras01	n/a		76,904	19	193,546	19	203,132	41	492	0	424,293	42	286,285	41	1,186,652
código de barras02	n/a		62,404	16	153,280	15	37,082	8	177,188	16	73,609	7	53,070	7	556,633
código de barras03	n/a		47,901	12	117,265	12	55,773	11	154,035	14	91,313	9	80,222	11	546,509
código de barras04	n/a		46,742	12	115,577	12	35,350	7	108,827	10	73,035	7	51,734	7	431,265
código de barras05	n/a		31,452	8	79,290	8	48,500	10	192,850	18	89,292	9	72,972	10	514,356
código de barras06	n/a		41,098	10	109,170	11	33,395	7	164,853	15	74,007	7	47,885	7	470,408
código de barras08	n/a		36,920	9	89,610	9	37,904	8	163,523	15	80,067	8	54,807	8	462,831
código de barras09	n/a		45,046	11	120,463	12	18,986	4	76,915	7	41,691	4	27,685	4	330,786
sin código de barras	n/a		7,872	2	15,937	2	21,625	4	48,715	4	74,263	7	32,478	5	200,890
IGHA	239,447	31	99,500	26	252,049	26	113,414	24	129,354	12	293,256	31	197,176	29	1,324,196
IGHD	1,950	0	12,498	3	31,868	3	8,634	2	28,929	3	20,042	2	13,434	2	117,355
IGHE	20	0	18	0	55	0	23	0	71	0	58	0	36	0	281
IGHG	127,371	16	59,533	15	160,522	16	62,561	13	124,327	12	151,614	16	94,596	16	780,524
IGHM	180,963	23	93,253	24	231,334	24	86,897	18	234,858	23	153,700	16	129,307	19	1,110,312
sin isotipo	233,763	30	123,665	32%	302,373	31	198,593	42	521,144	50	328,637	35	242,111	36	1,950,286

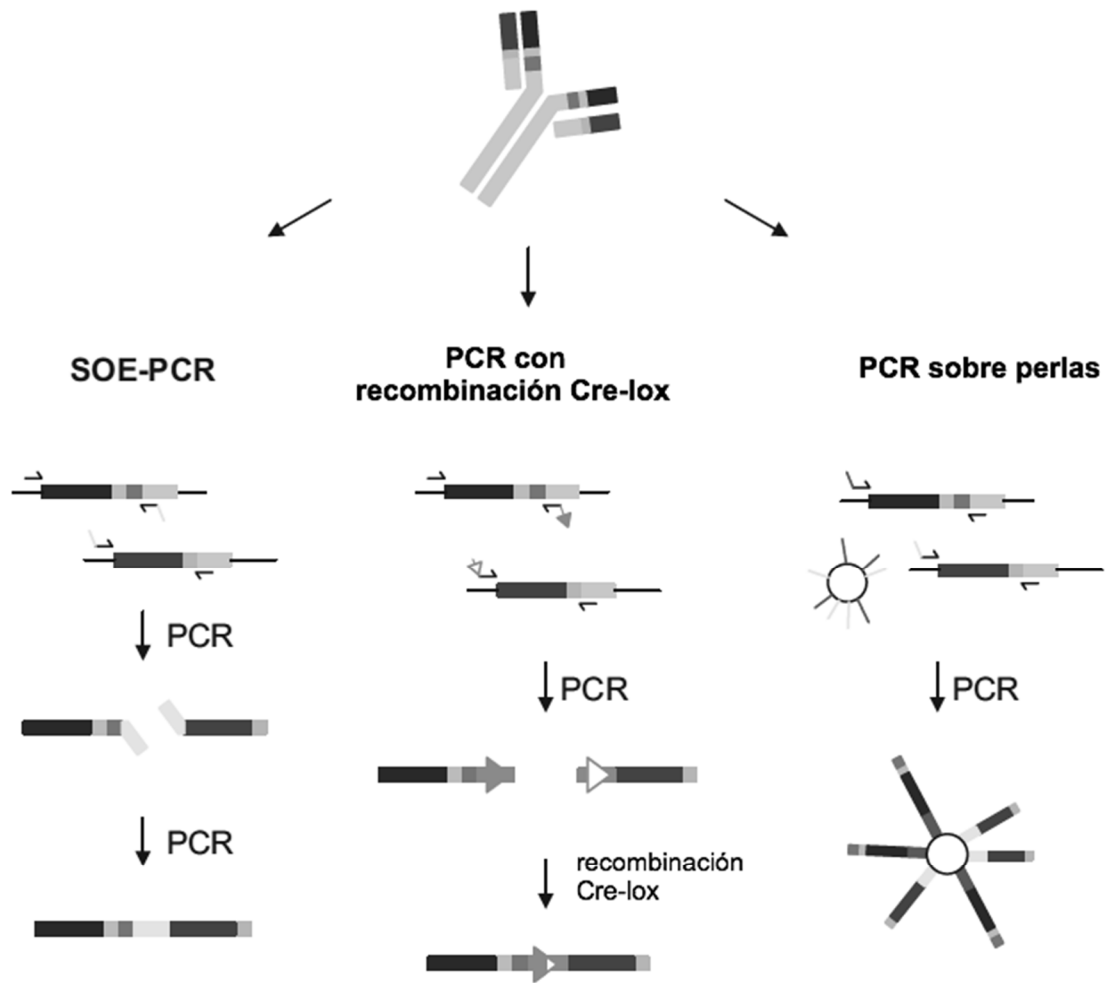


FIG. 27

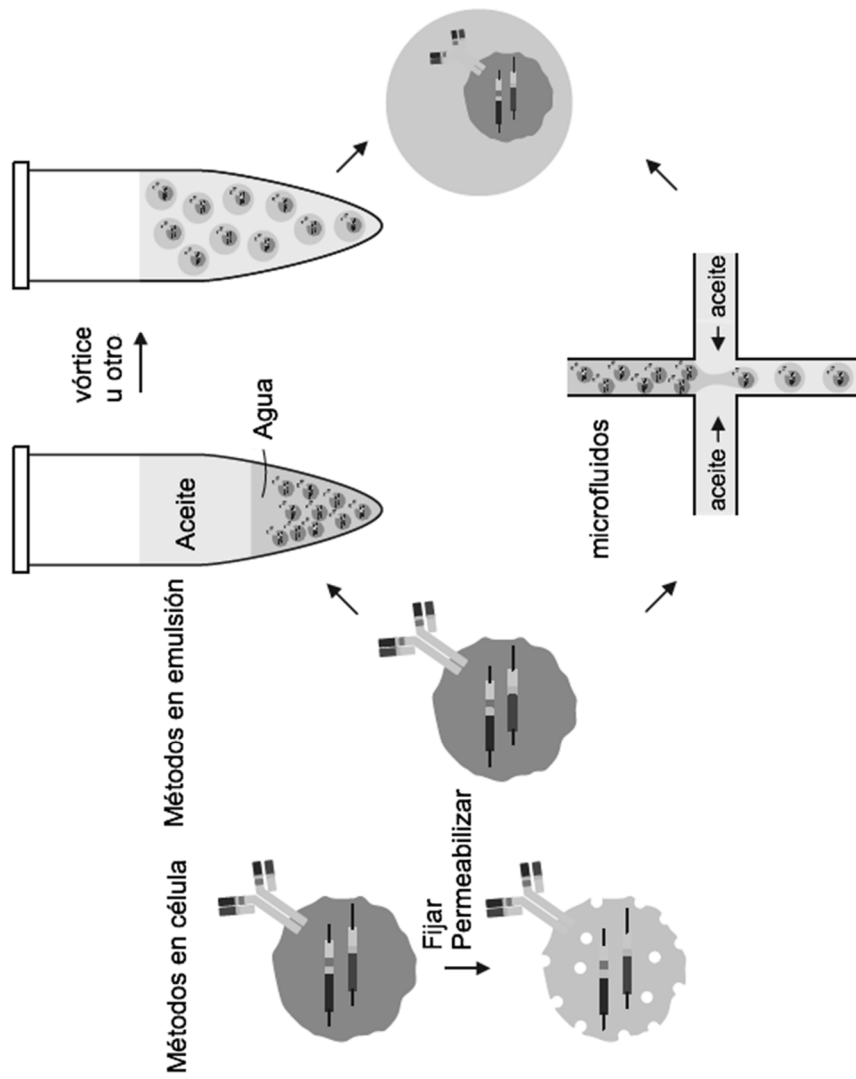


FIG. 28

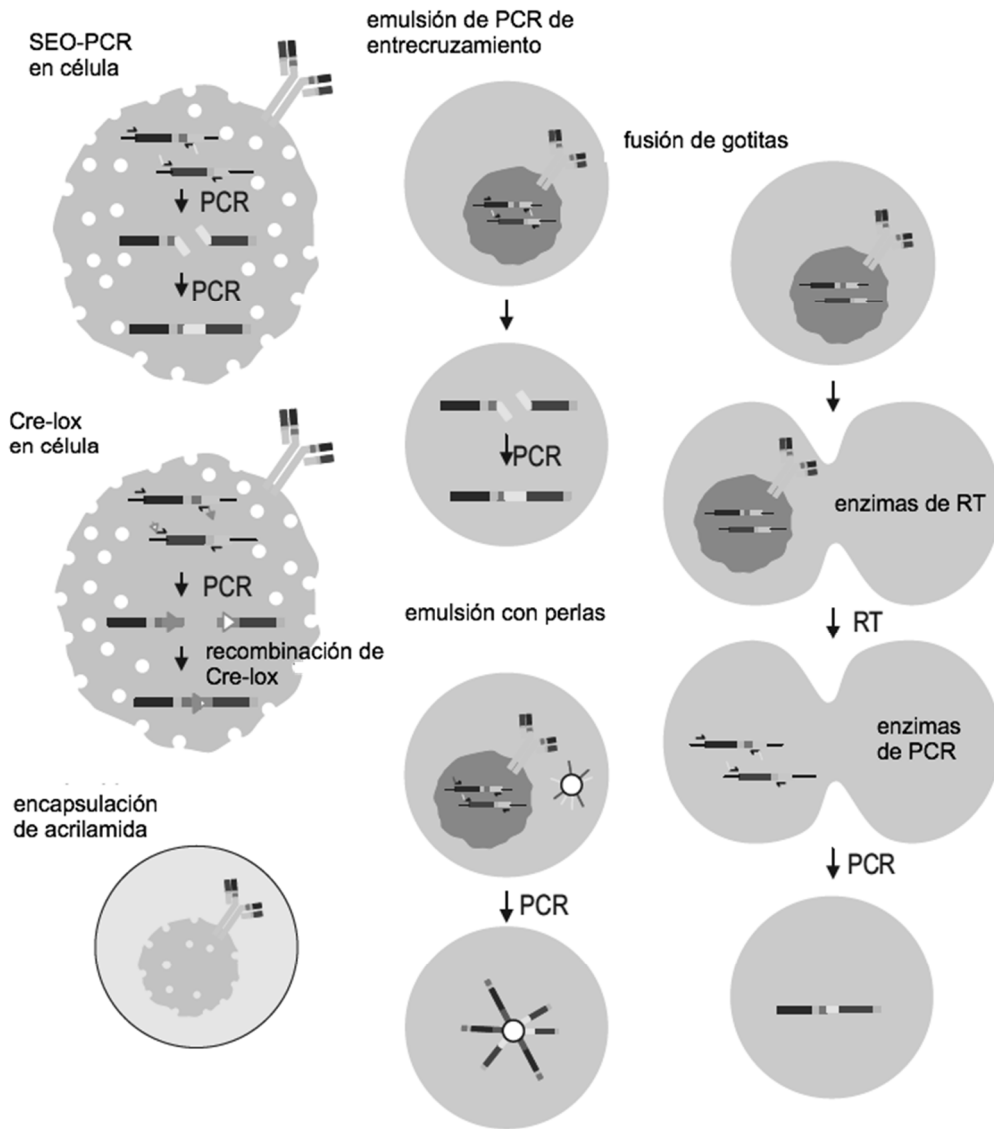


FIG. 29

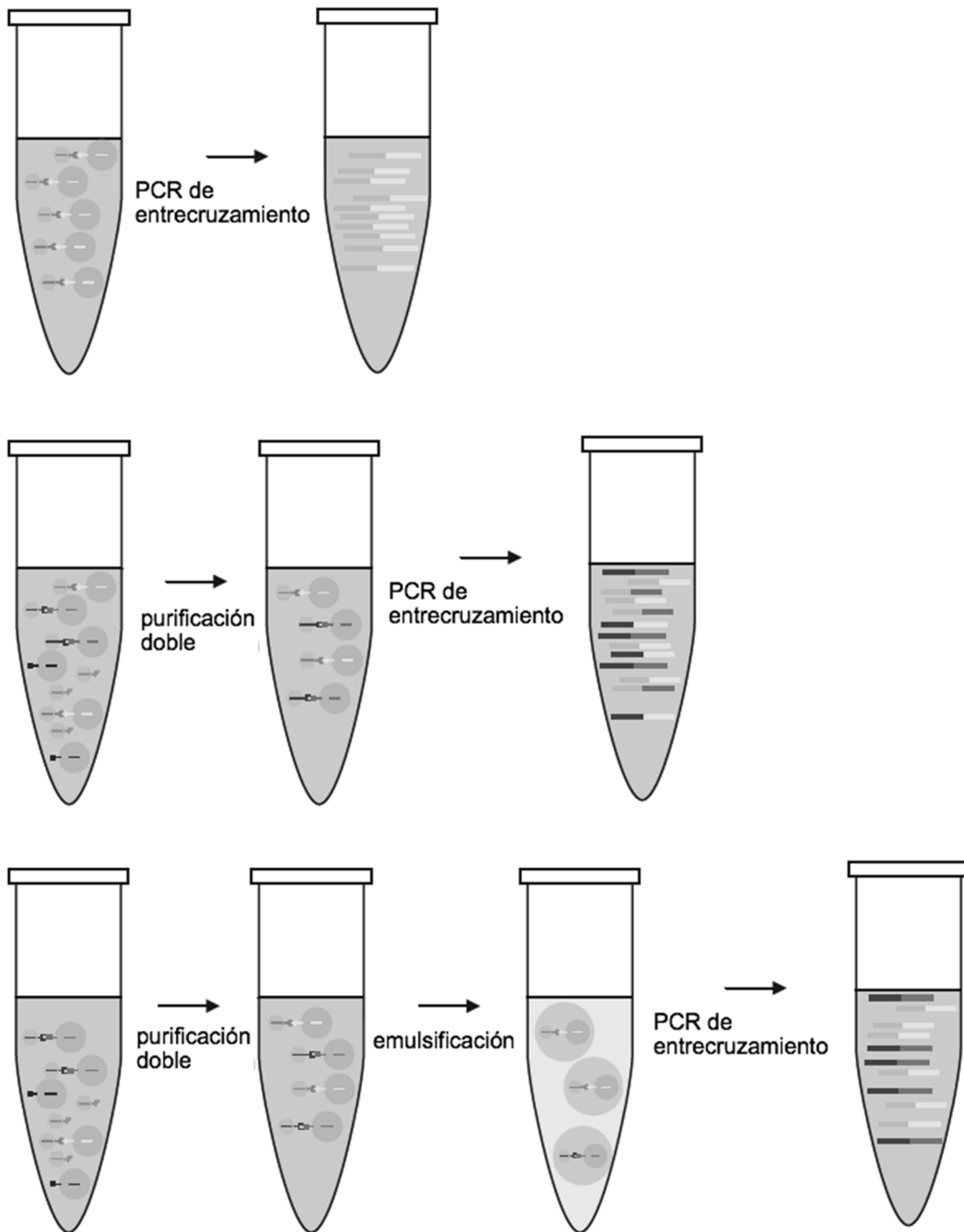


FIG. 30

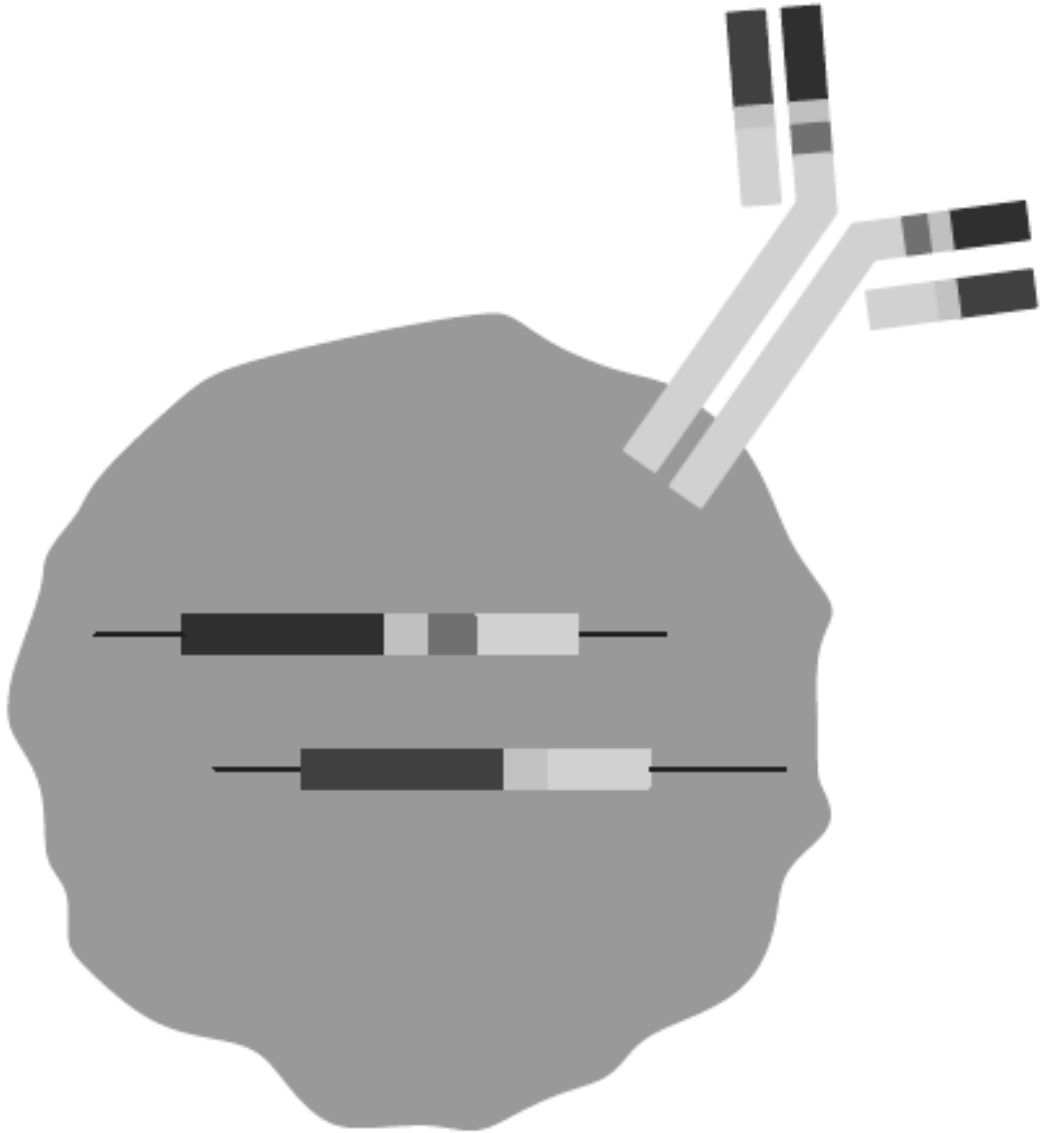


FIG. 31

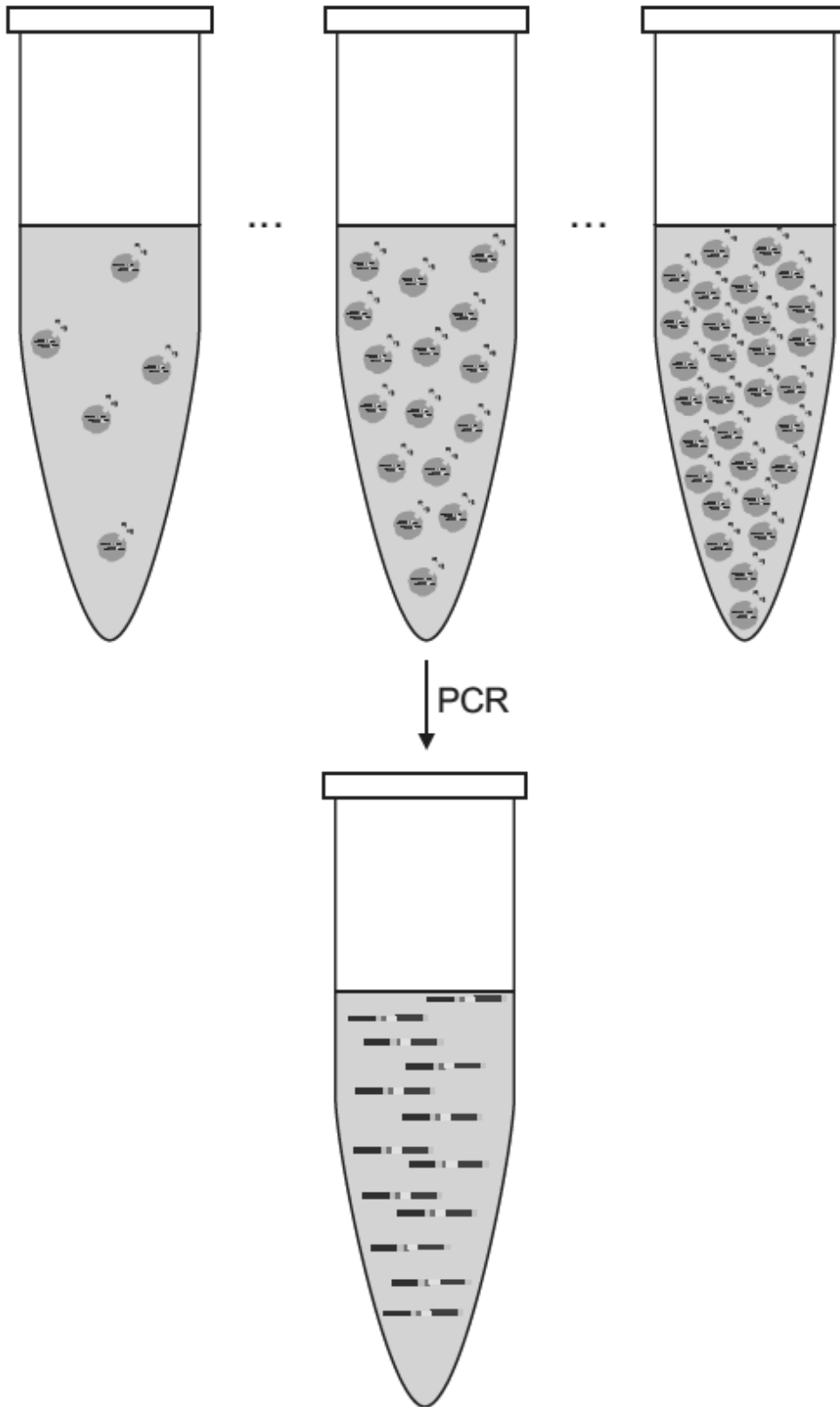


FIG. 32

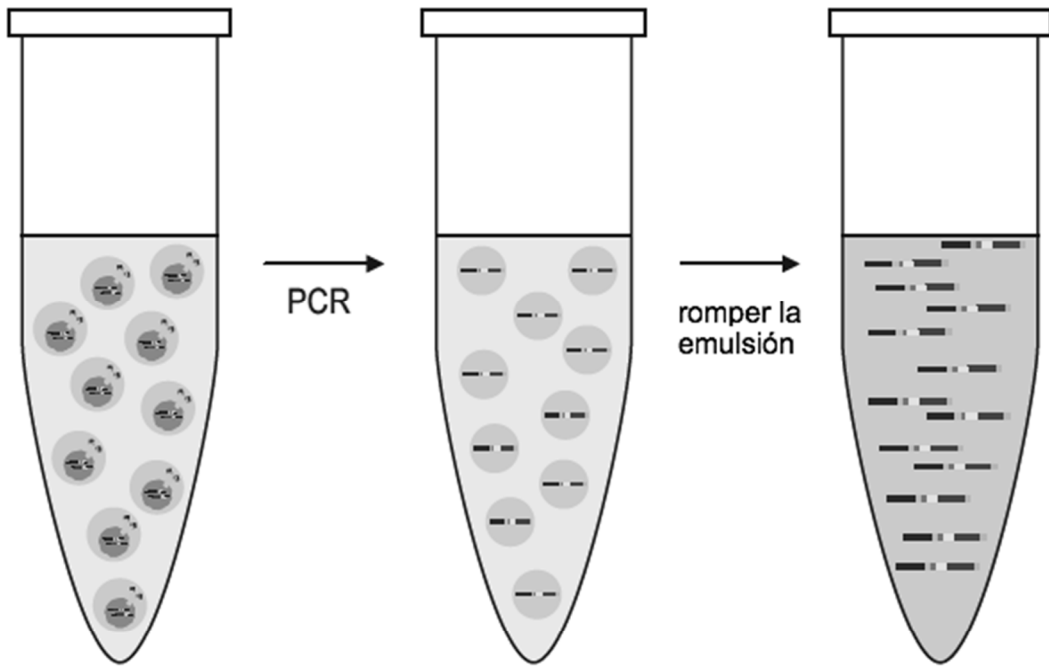


FIG. 33

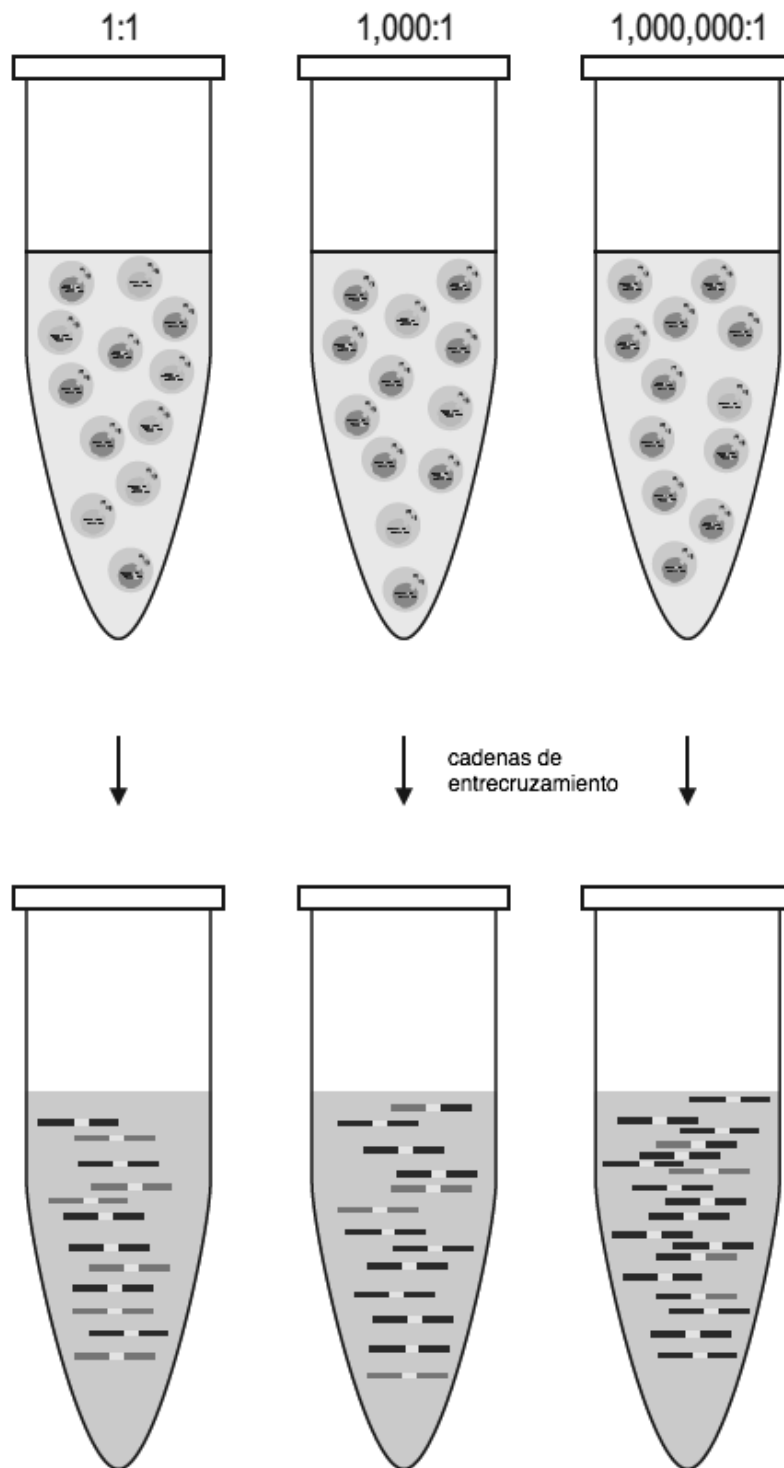


FIG. 34

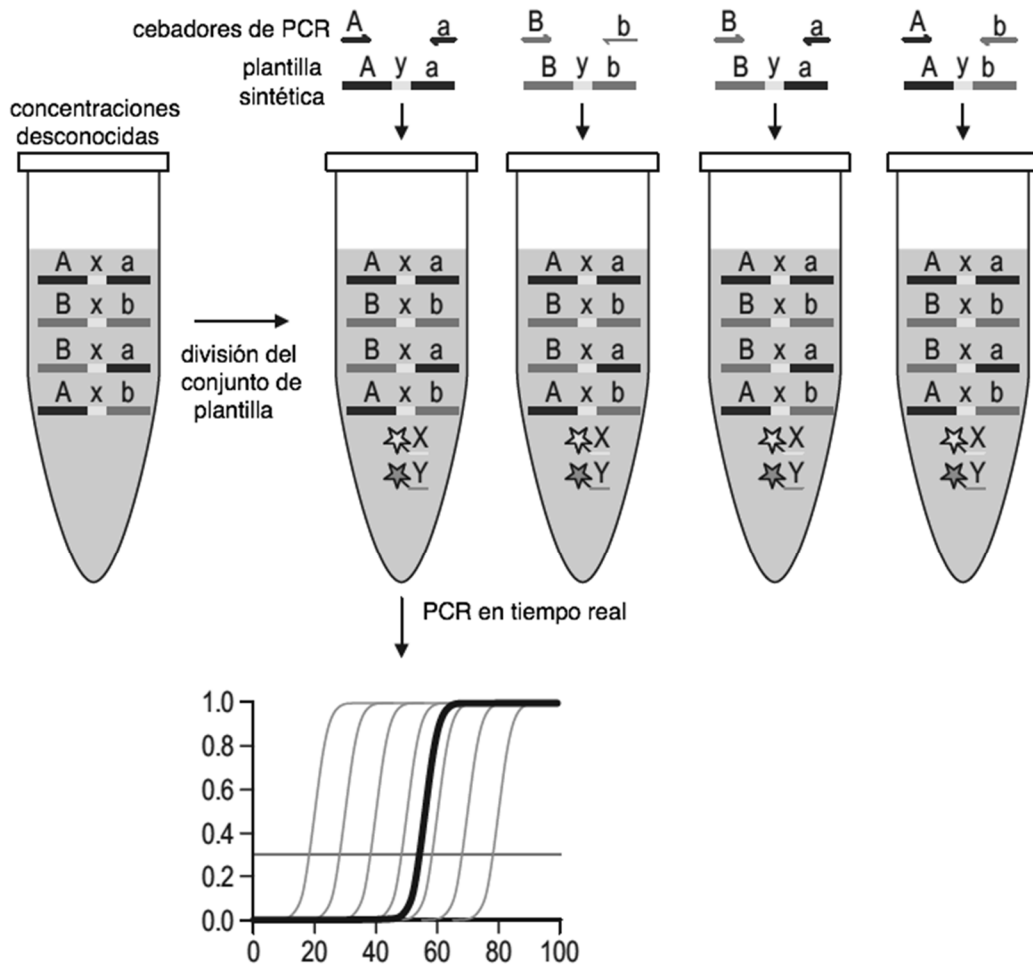


FIG. 35

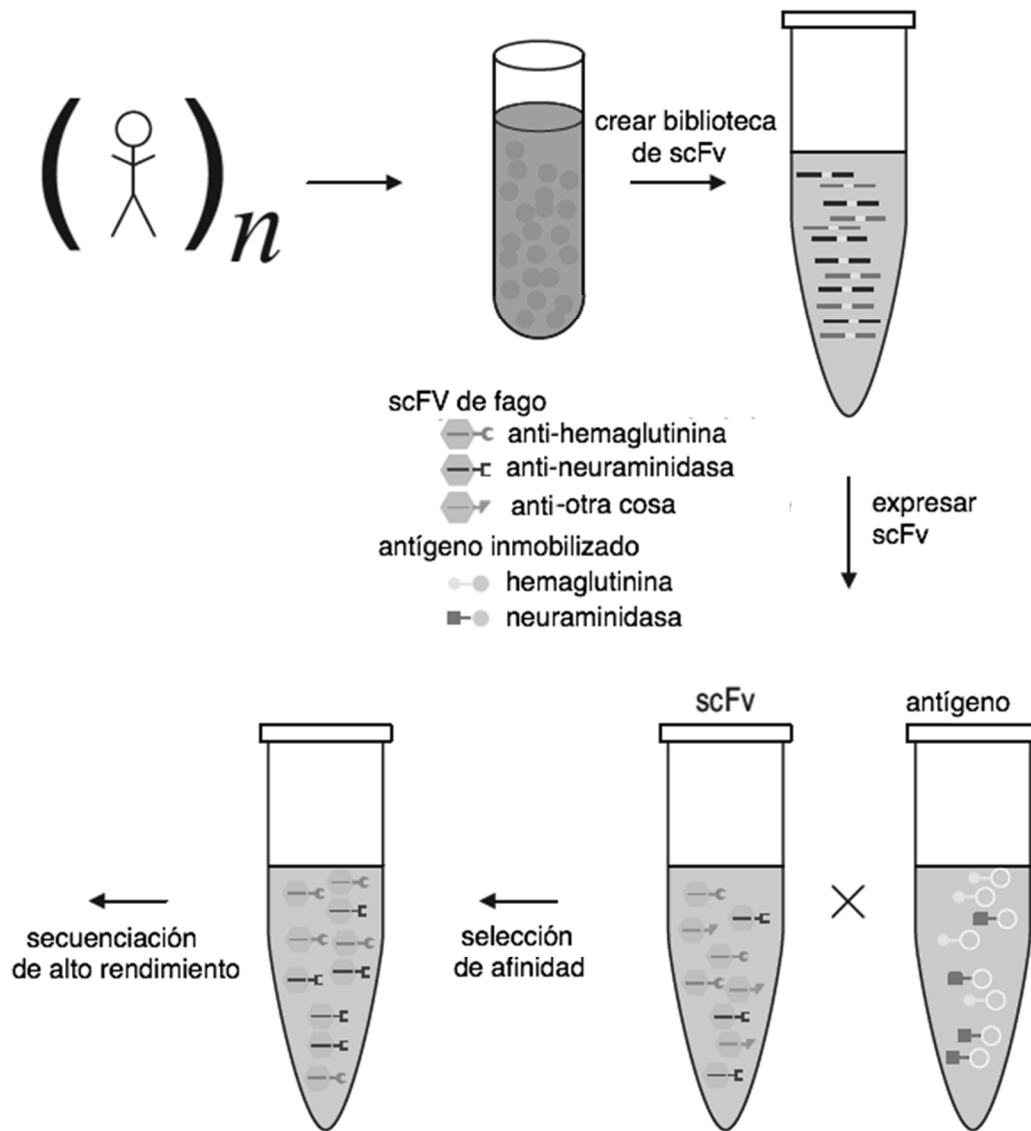


FIG. 36

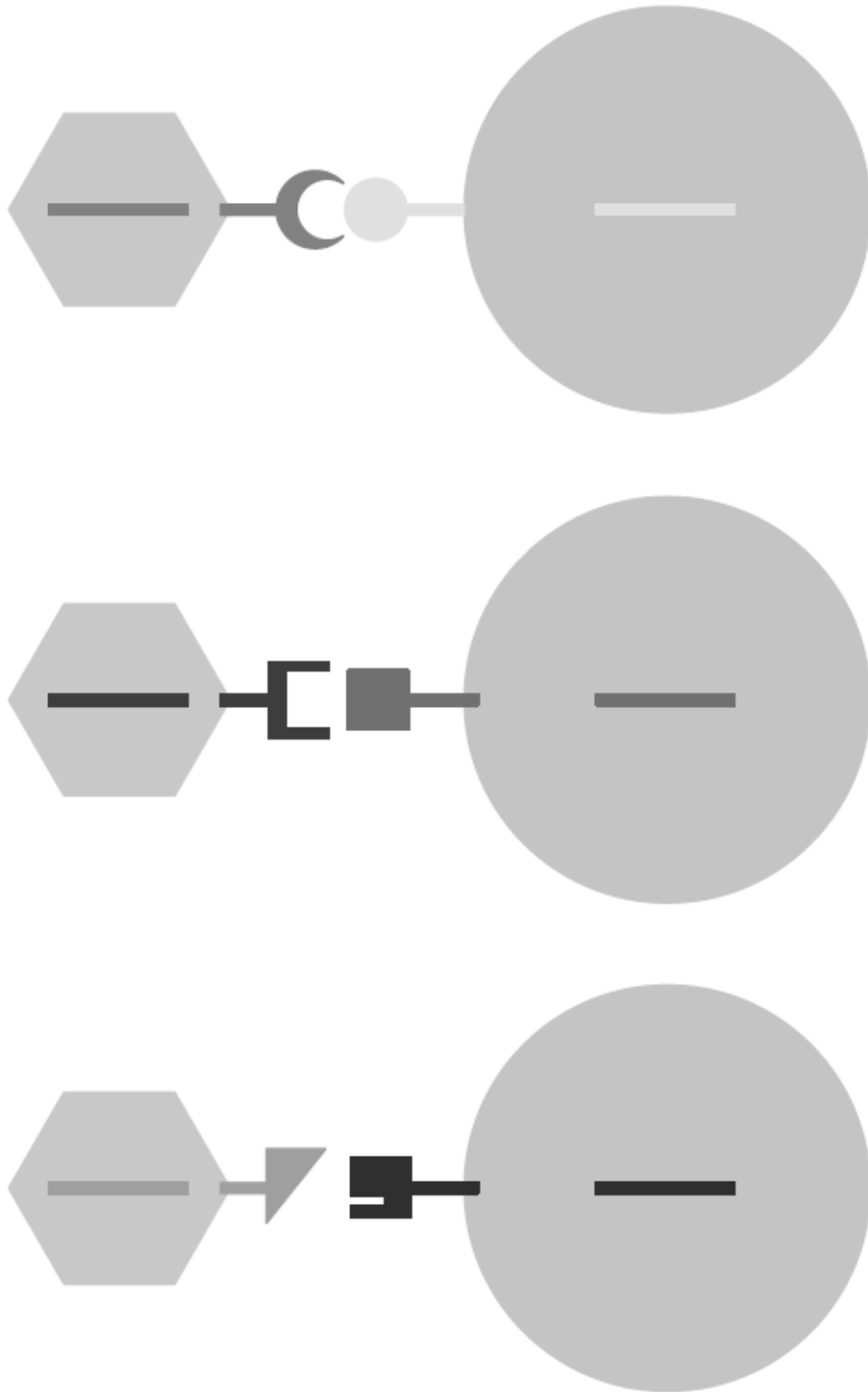


FIG. 37

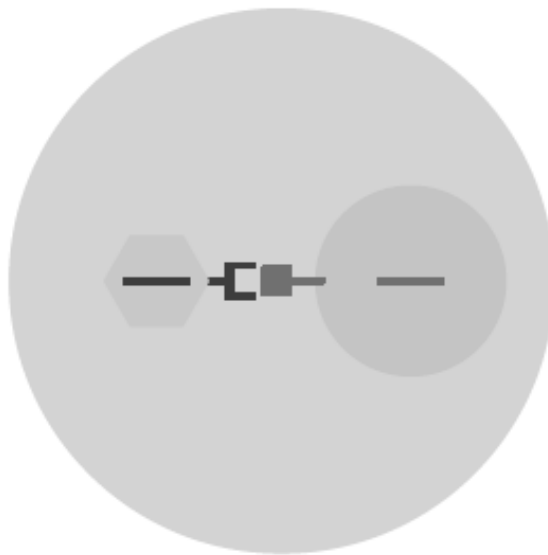
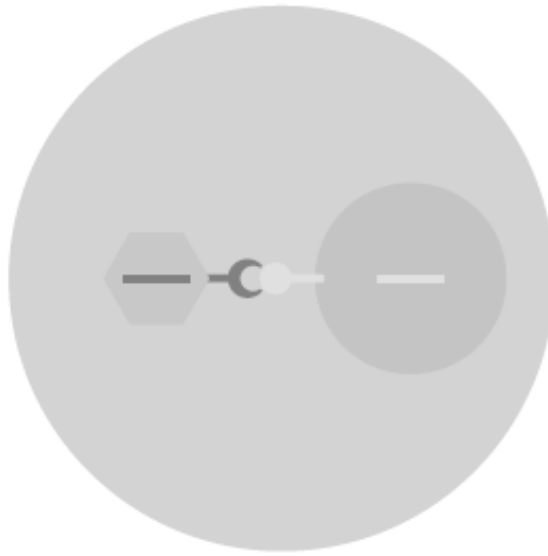


FIG. 38

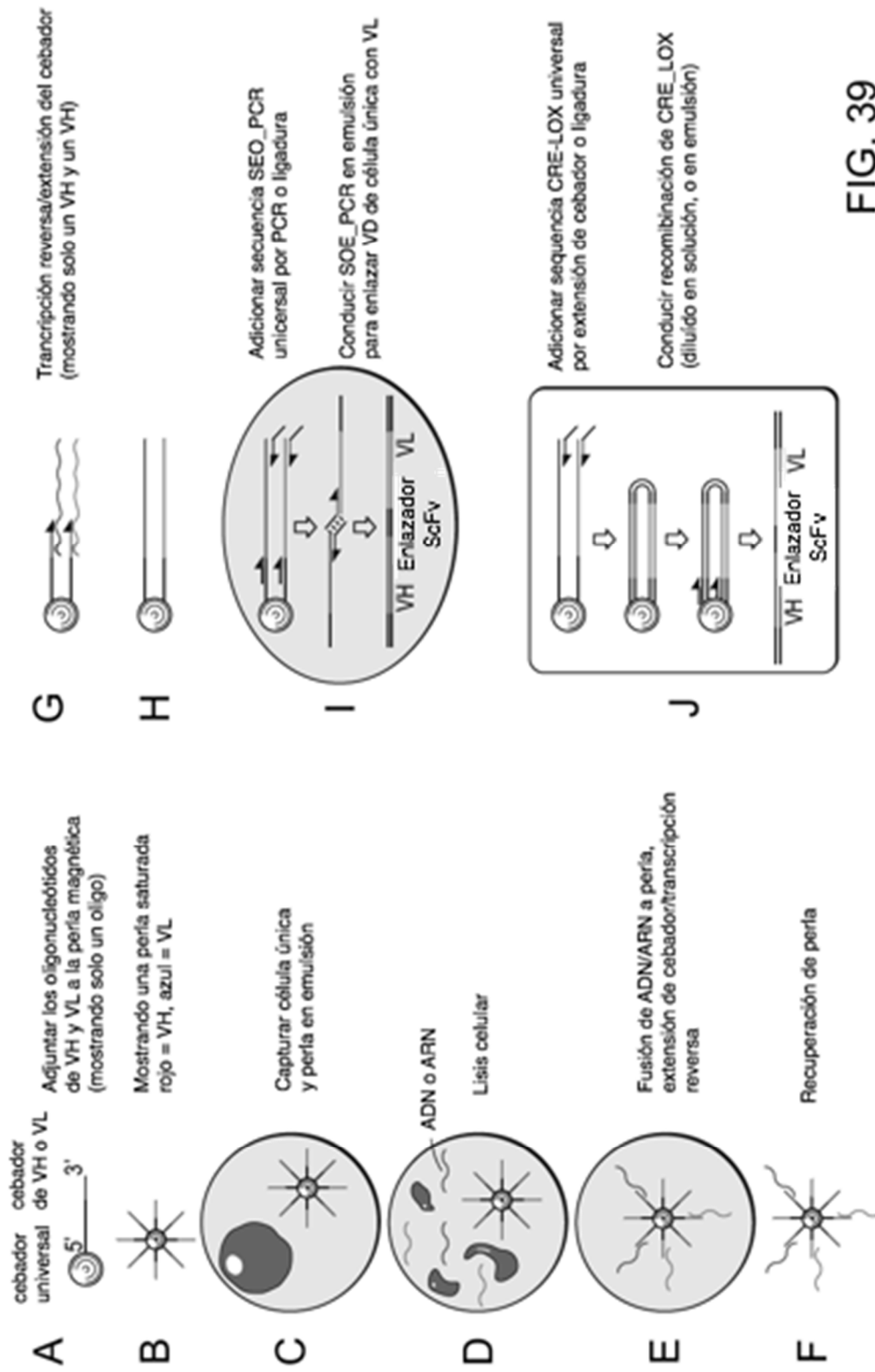


FIG. 39

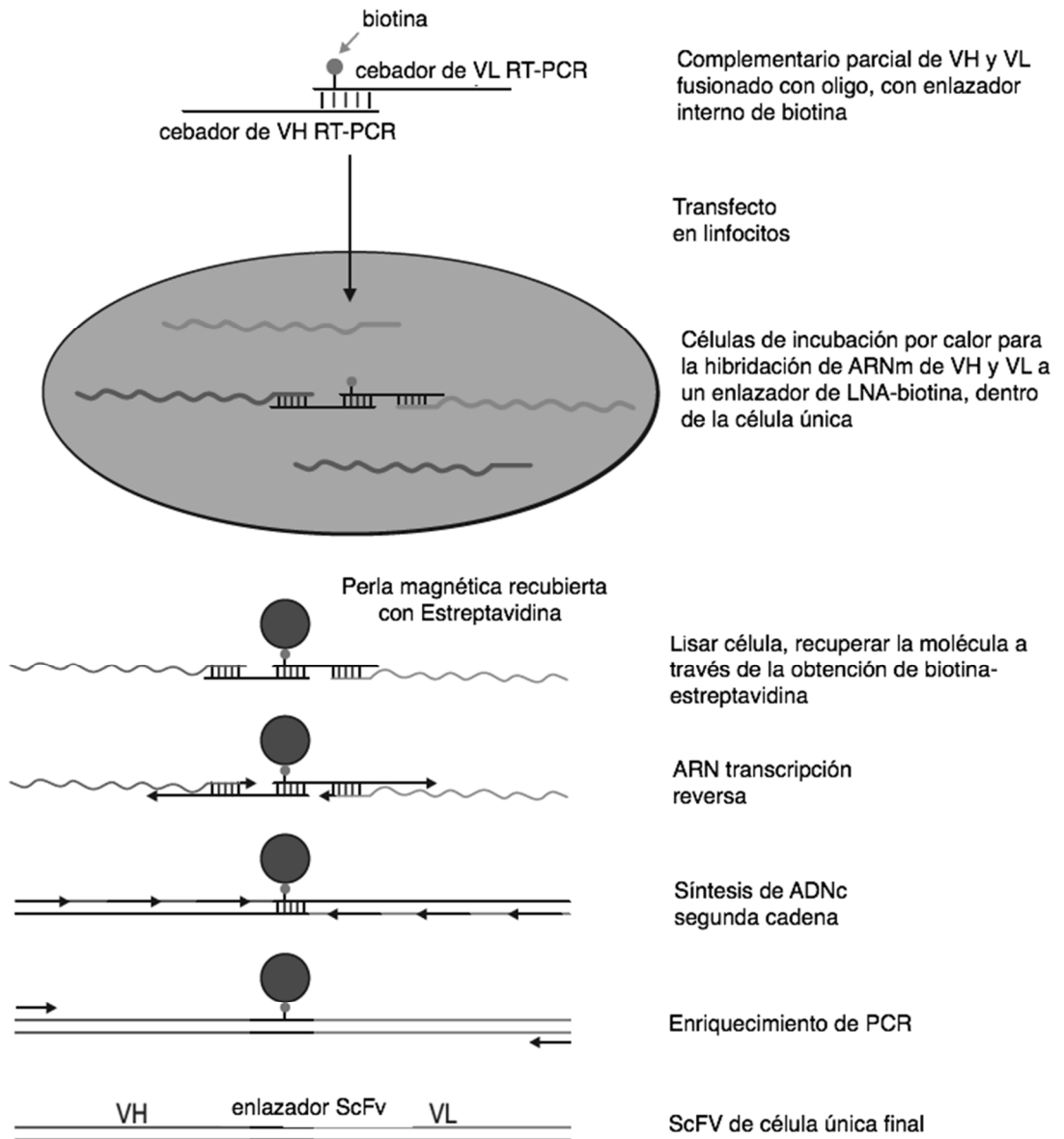


FIG. 40

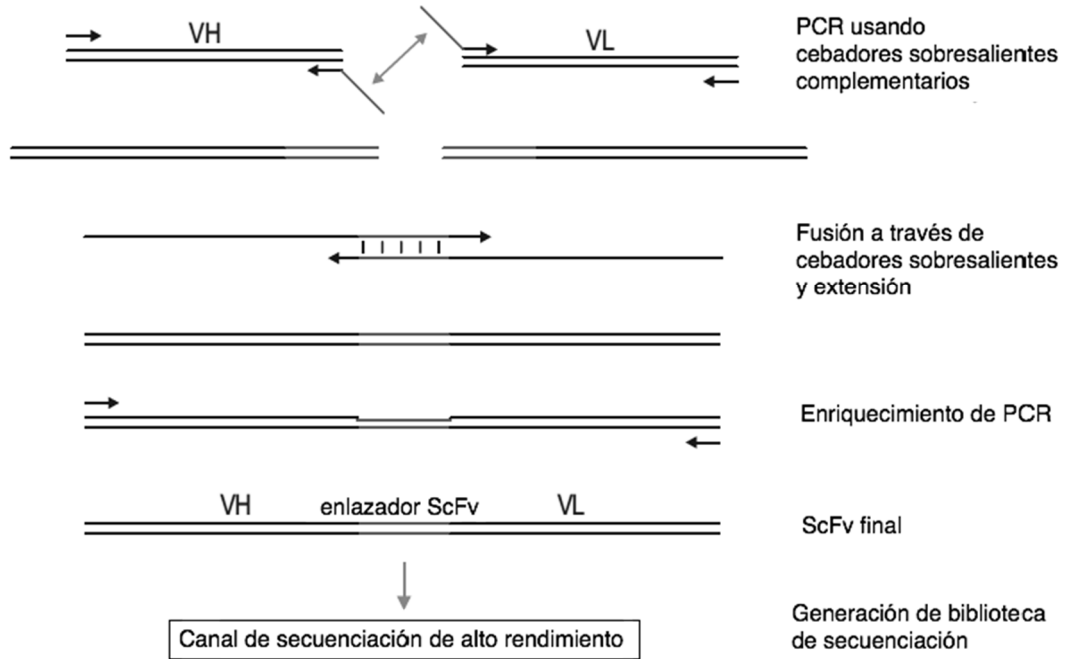


FIG. 41

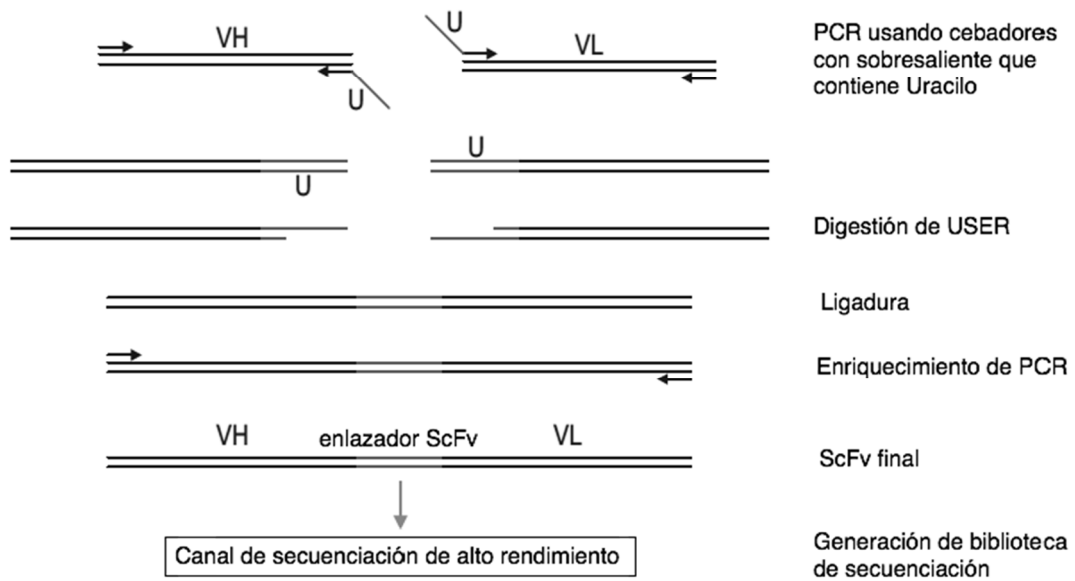


FIG. 42

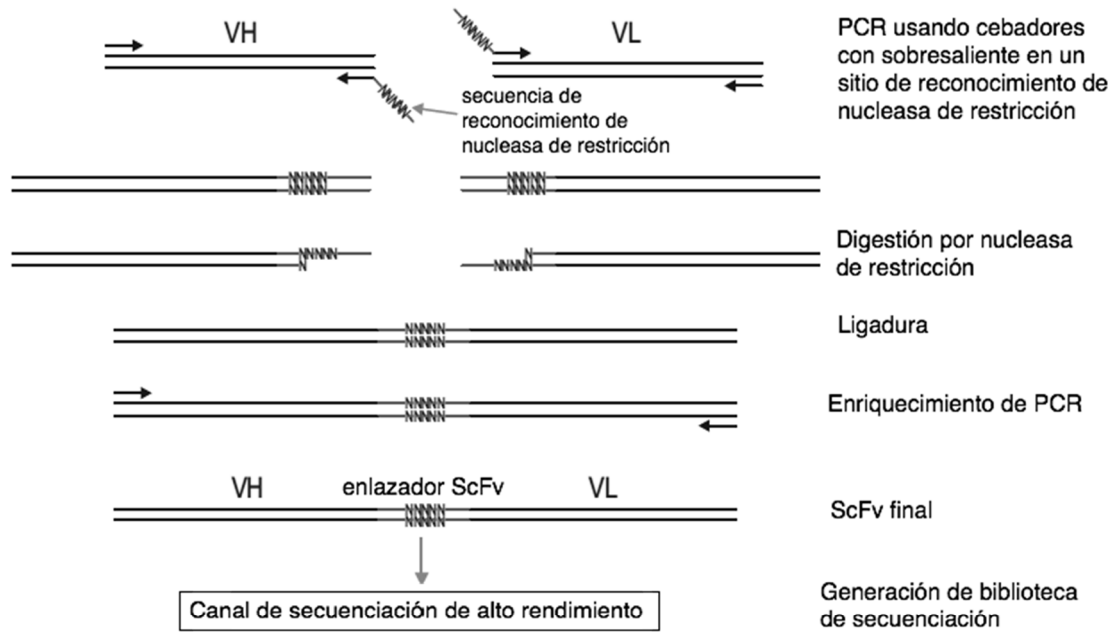


FIG. 43

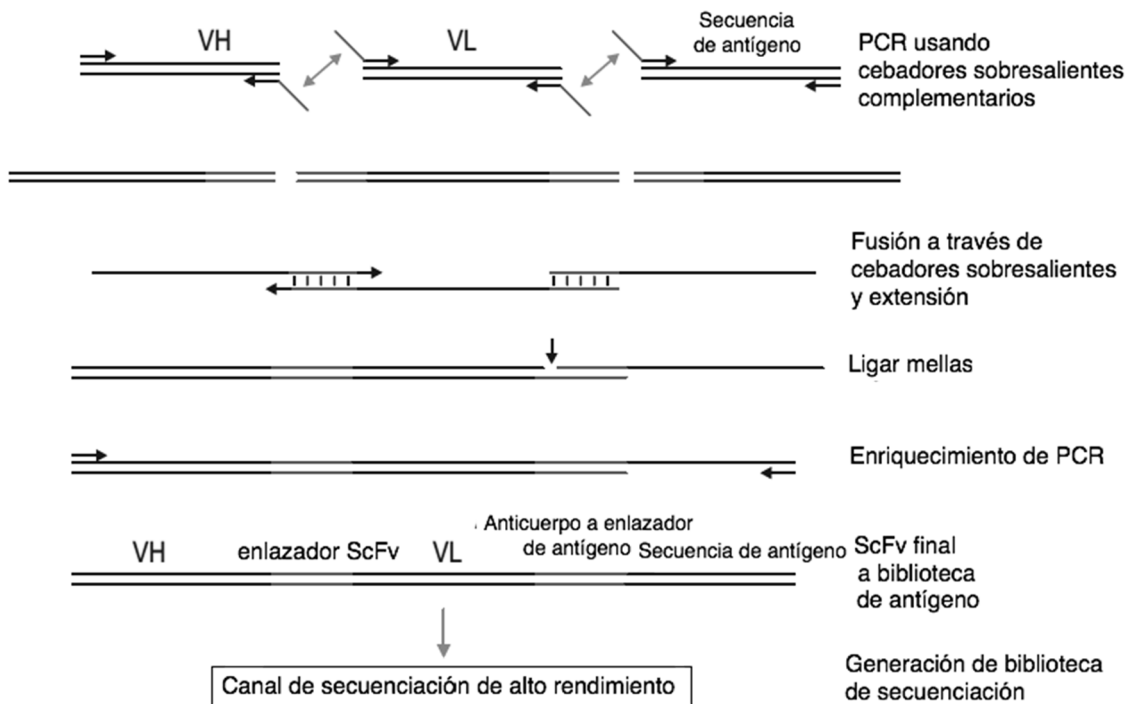


FIG. 44

FIG. 45

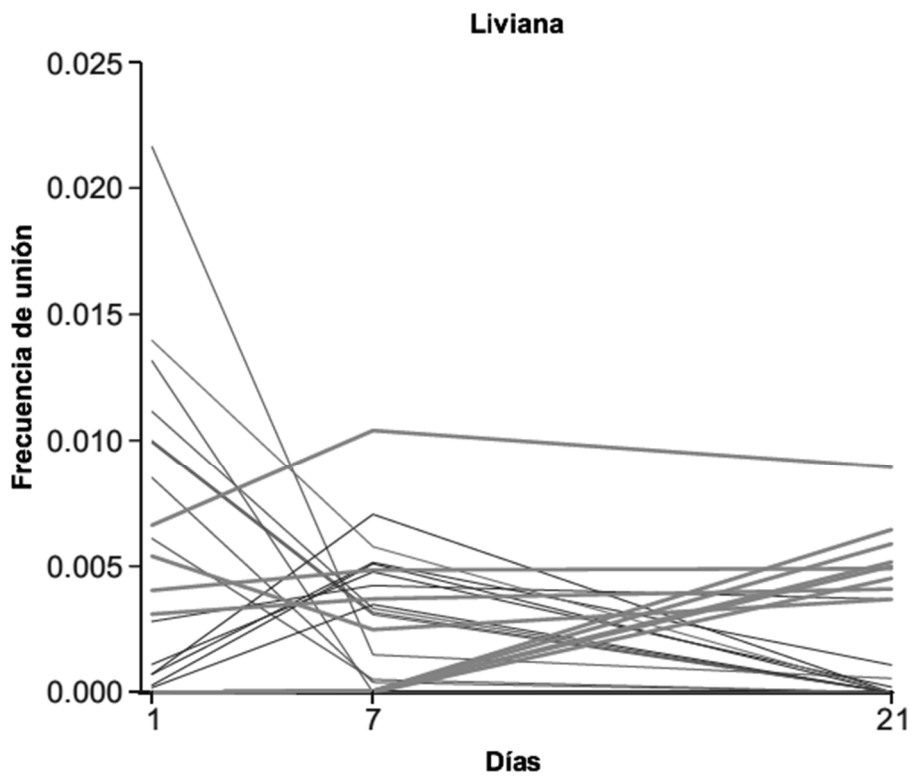
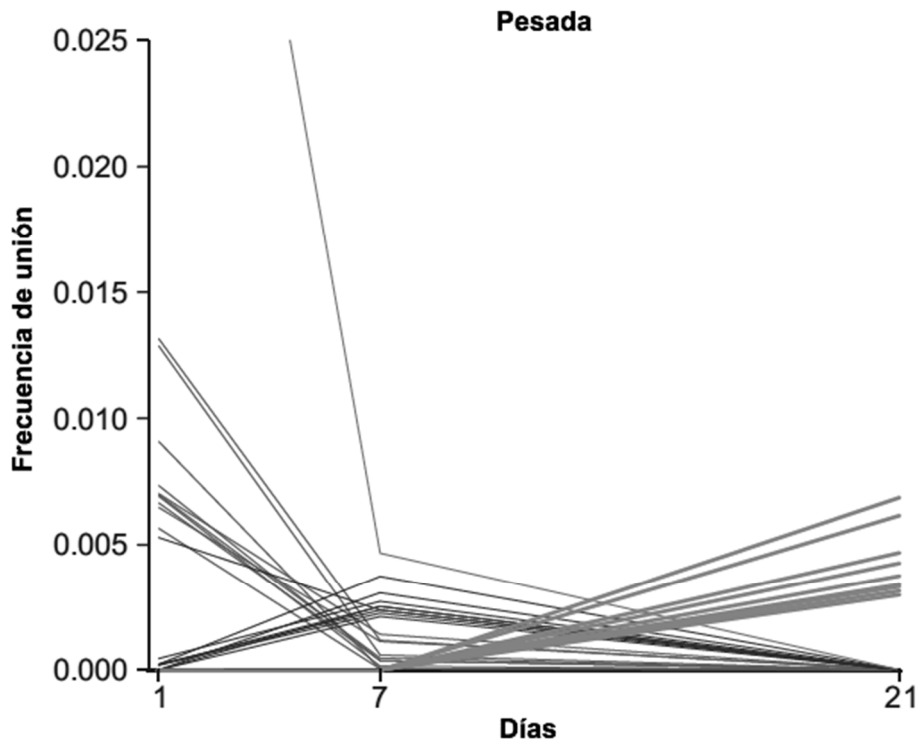


FIG. 46

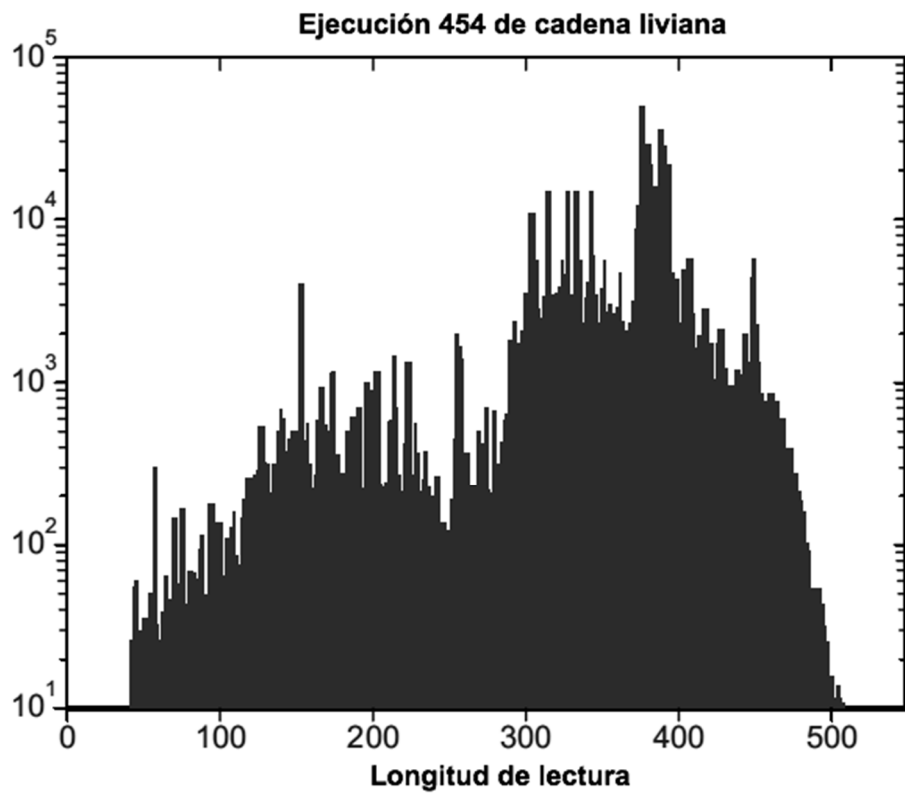
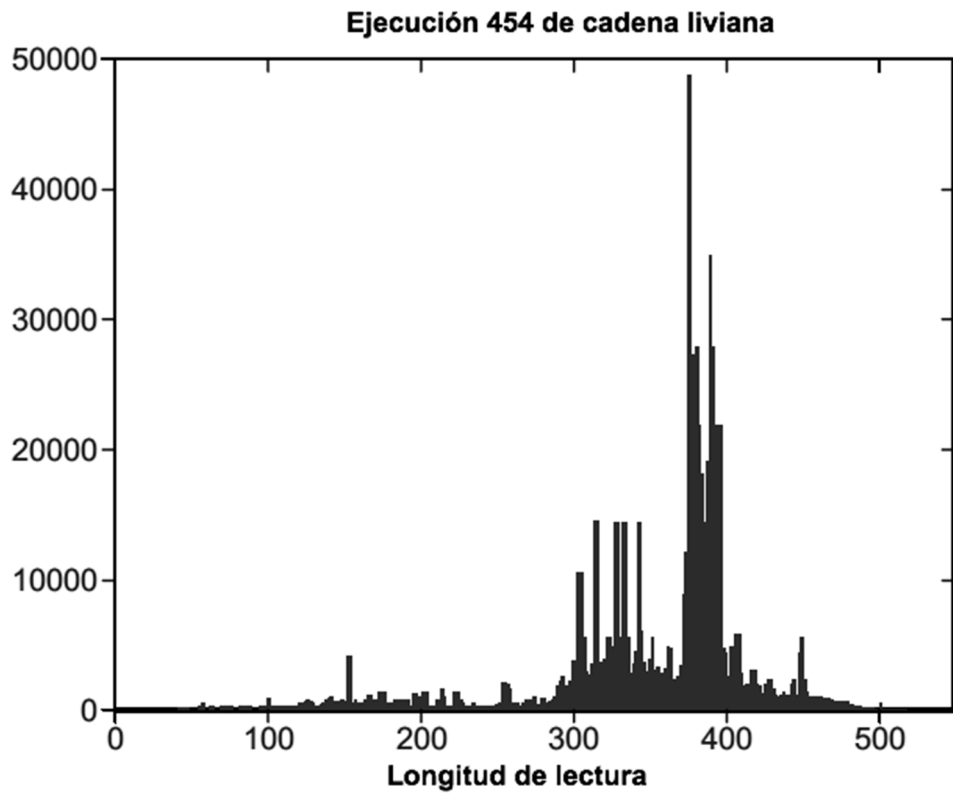
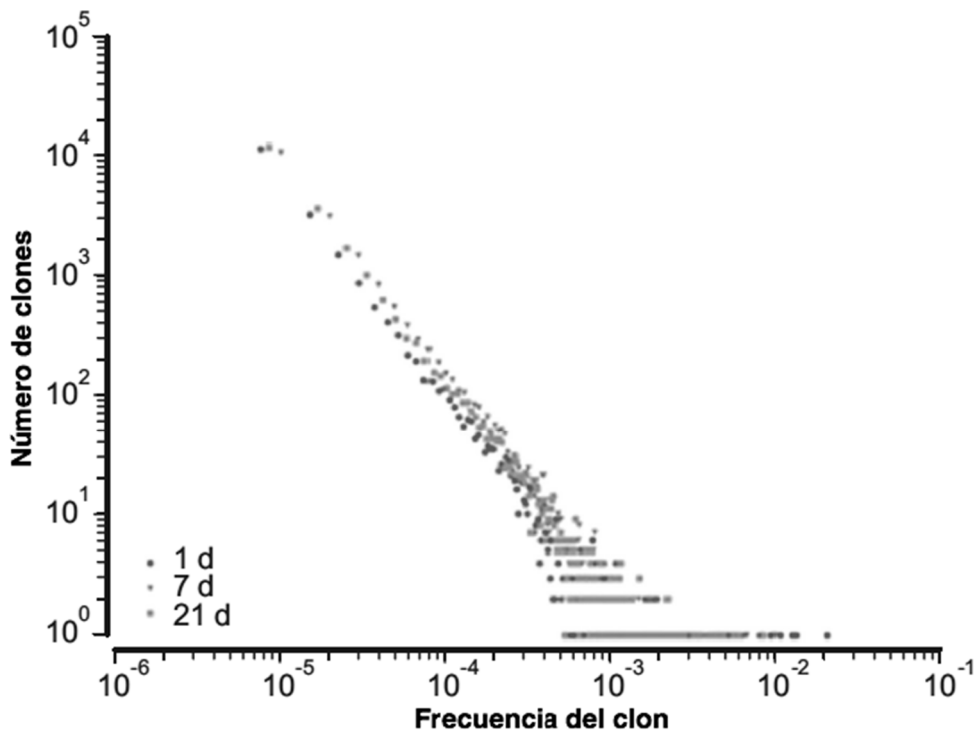
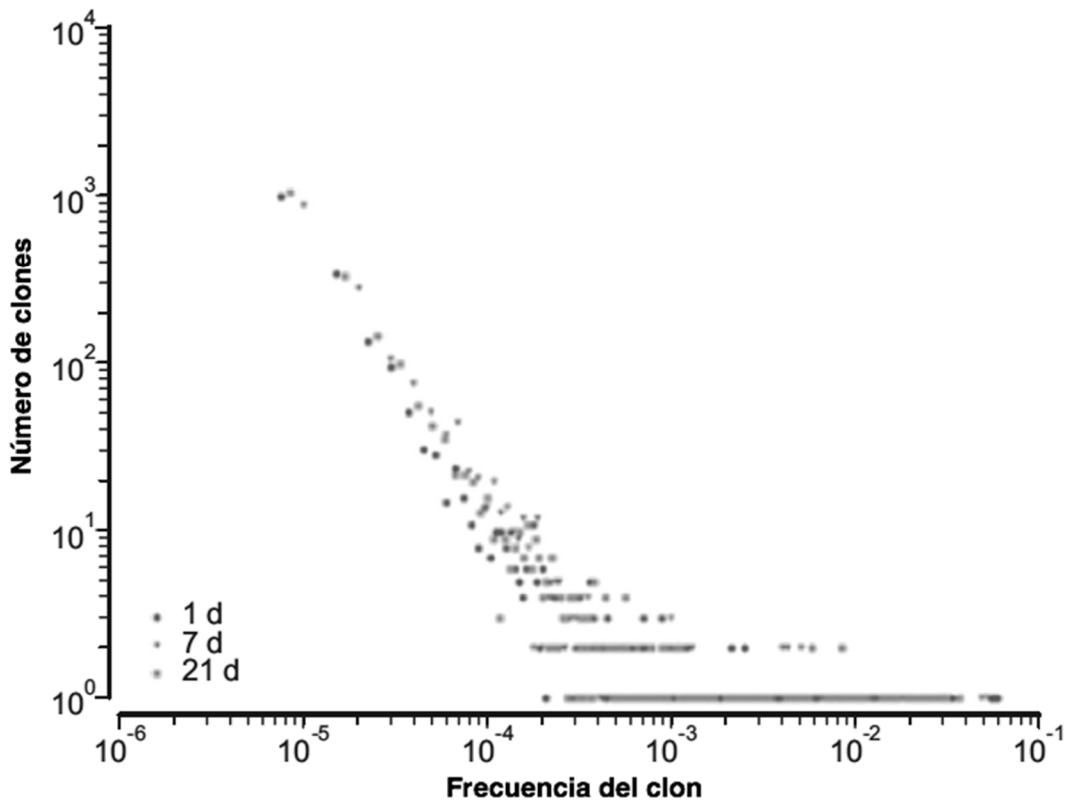


FIG. 47



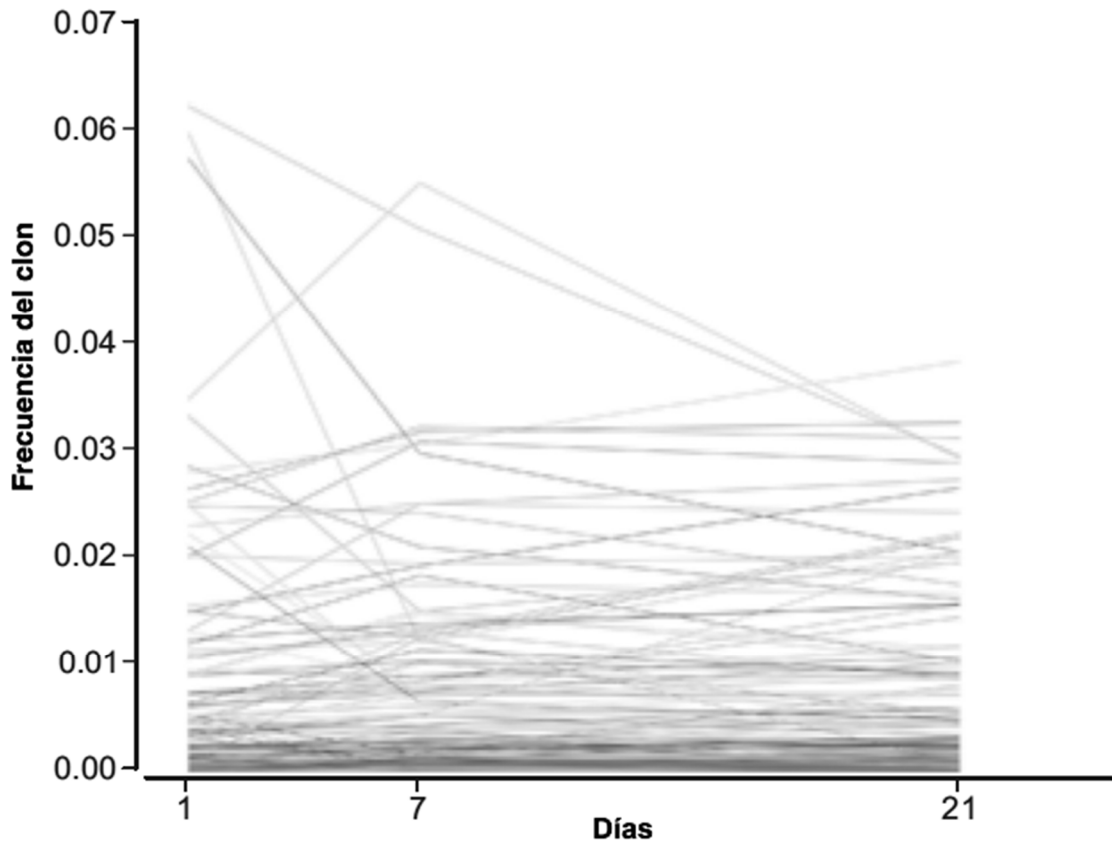


FIG. 48

FIG. 49

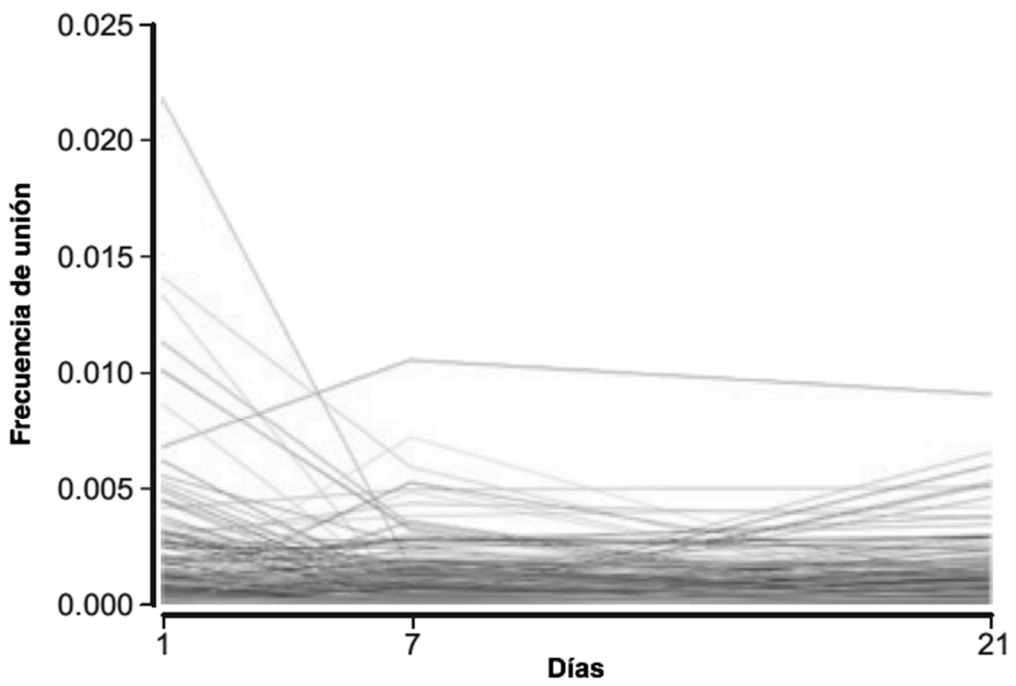
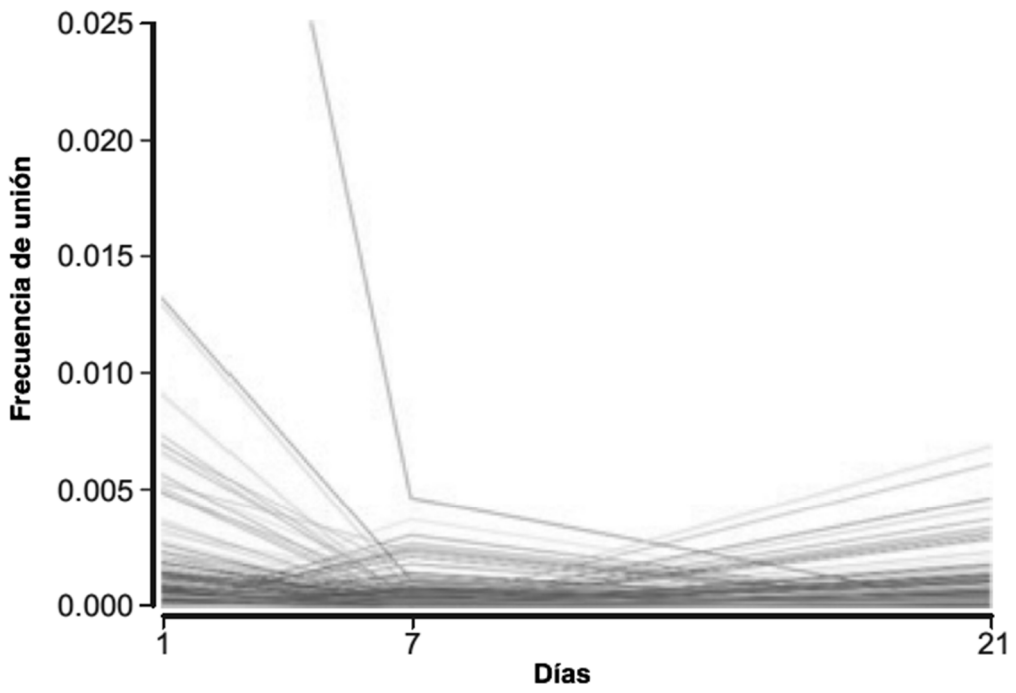
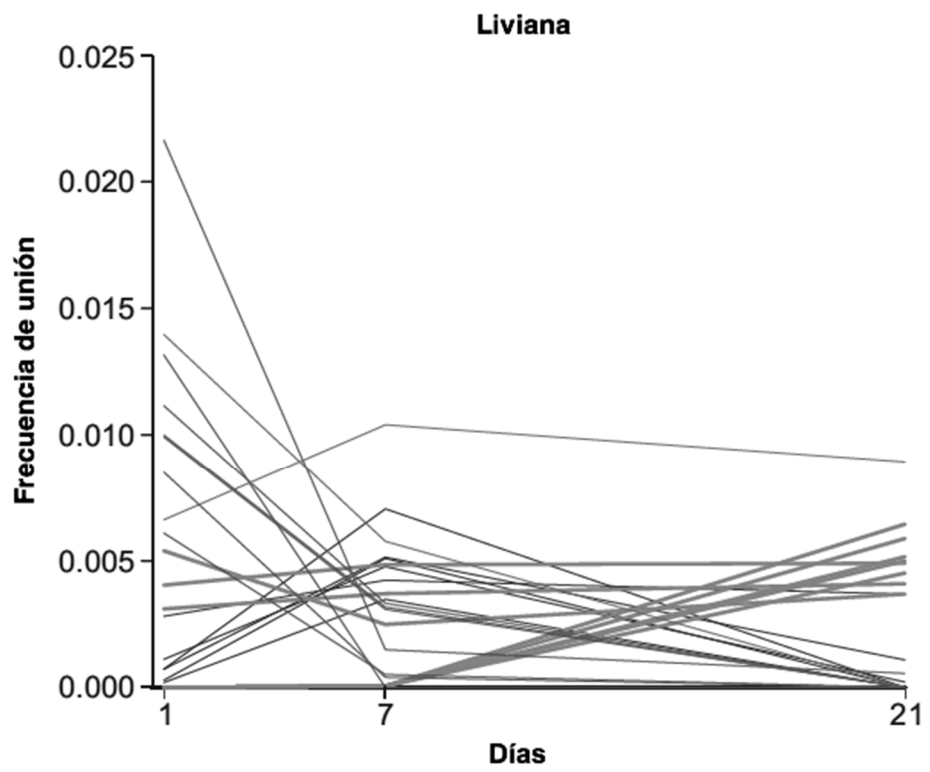
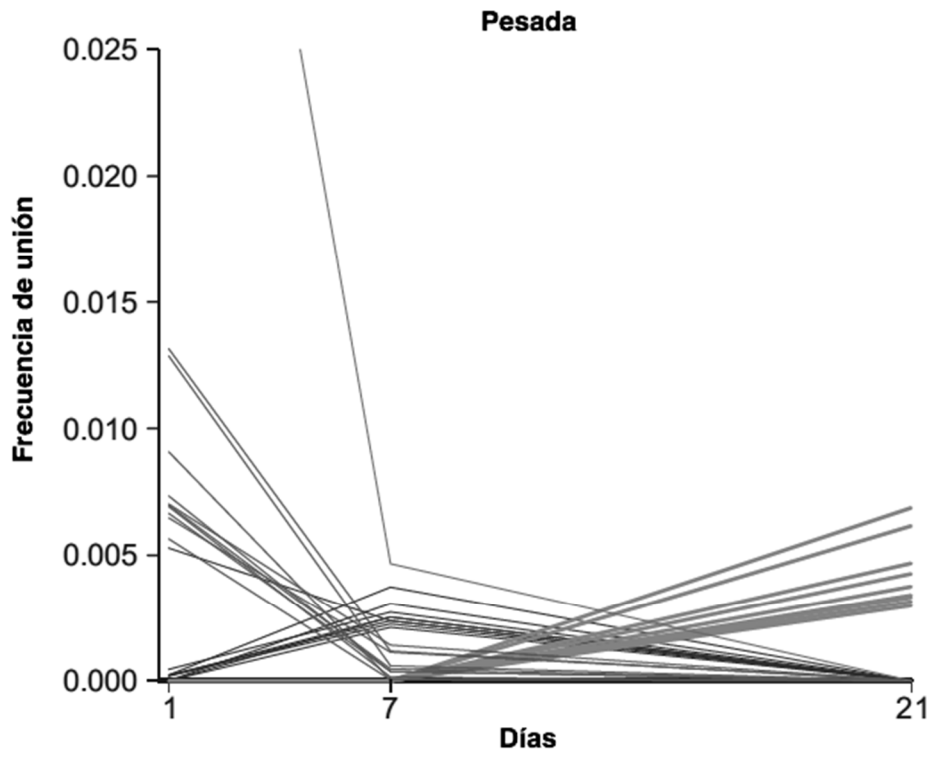


FIG. 50



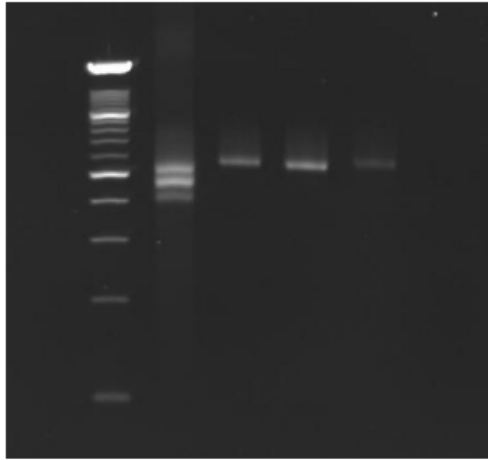


FIG. 51

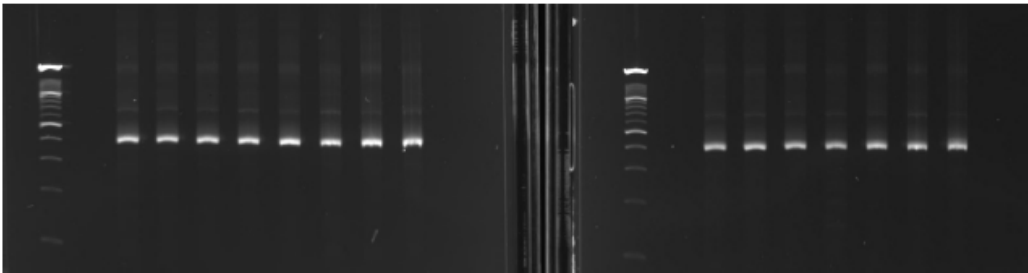


FIG. 52

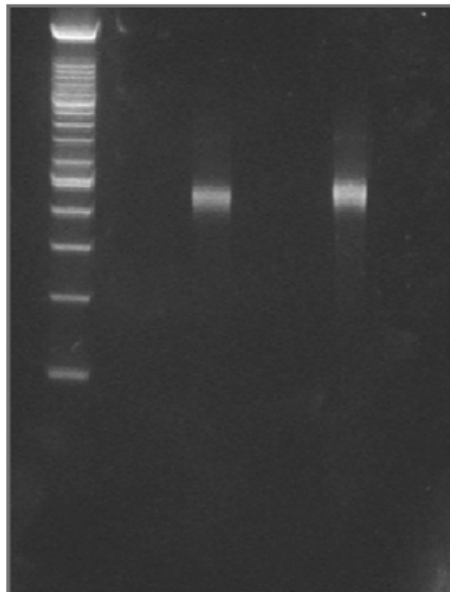


FIG. 53

FIG. 54

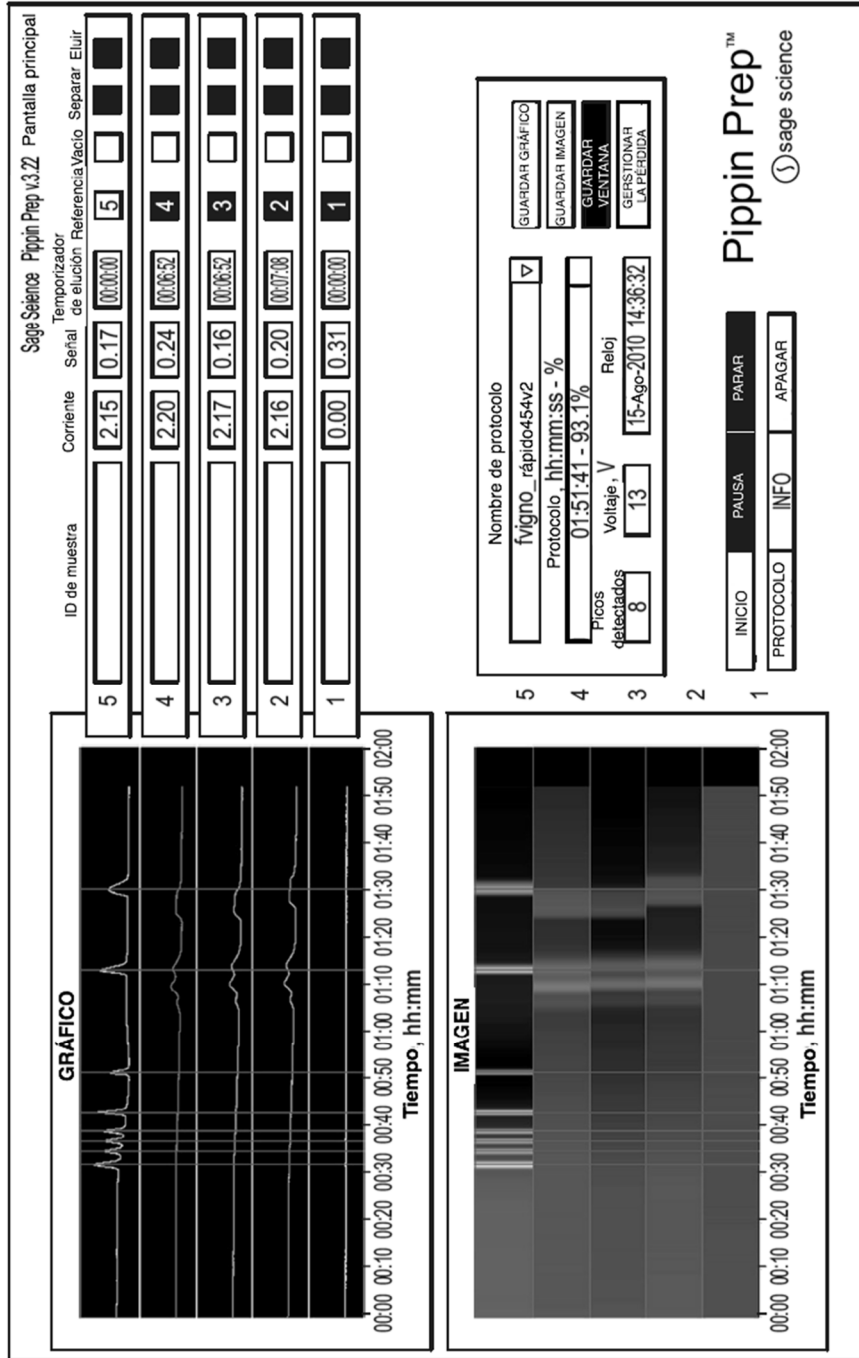
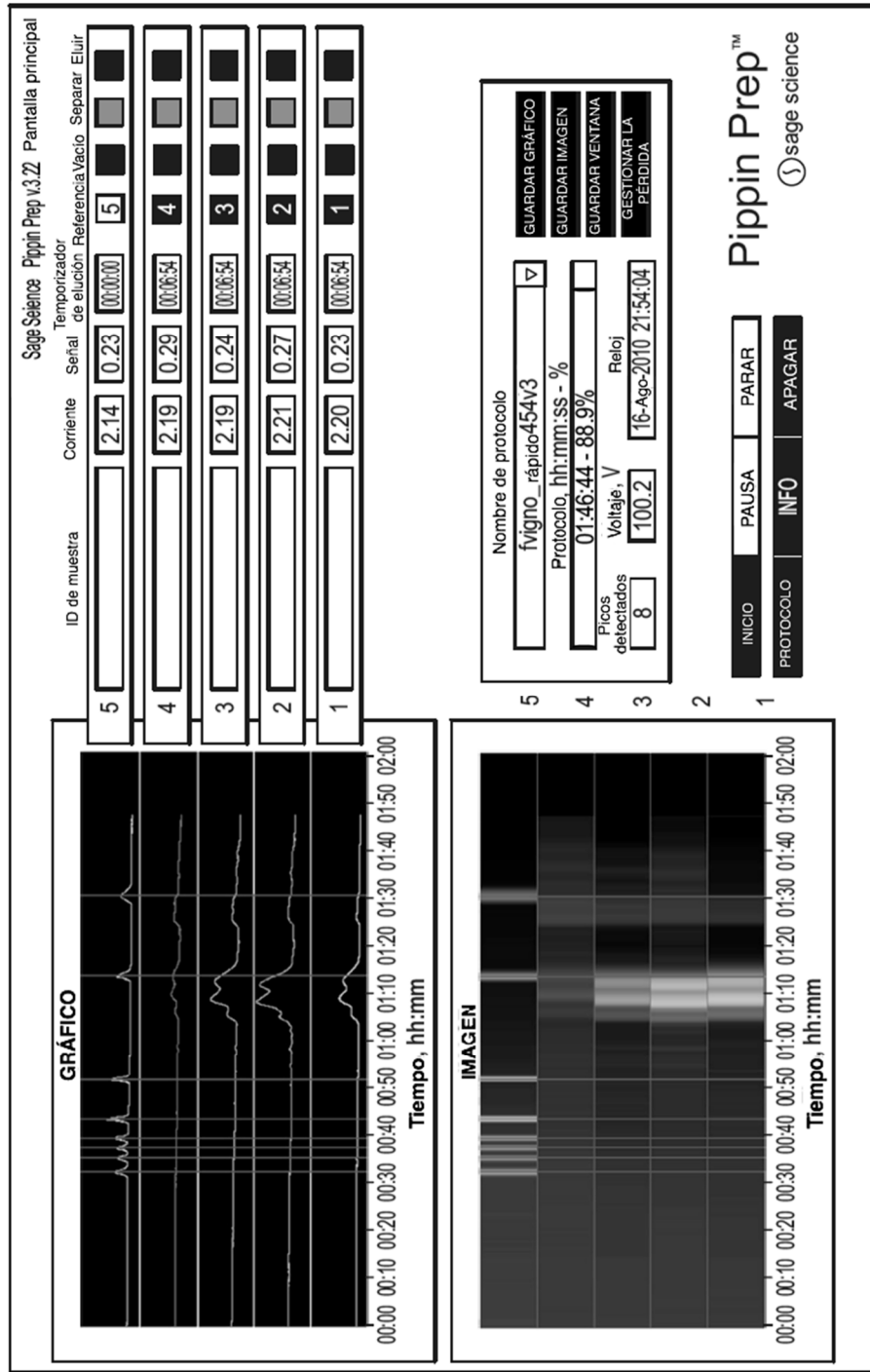


FIG. 54 Cont.



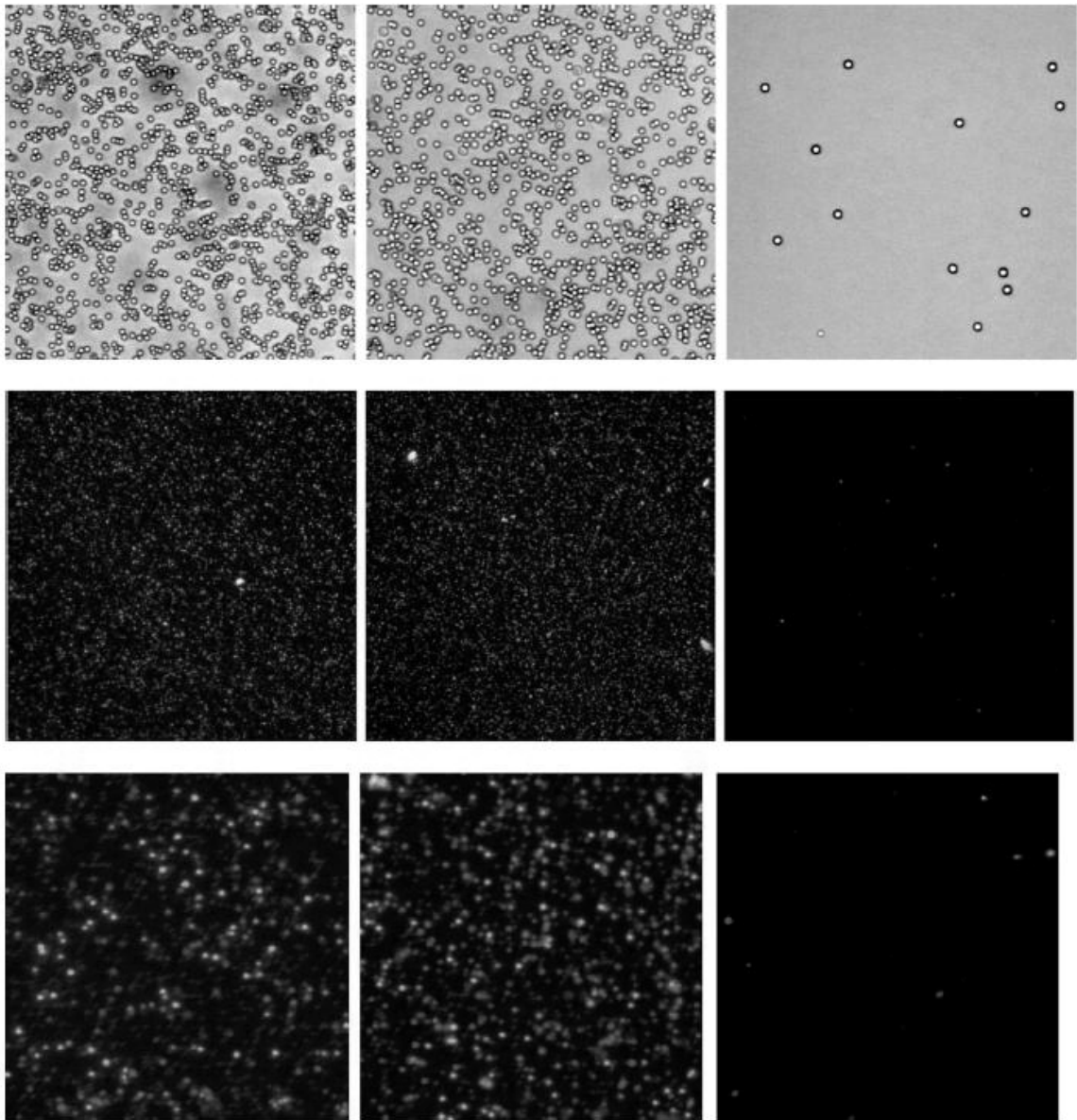


FIG. 55

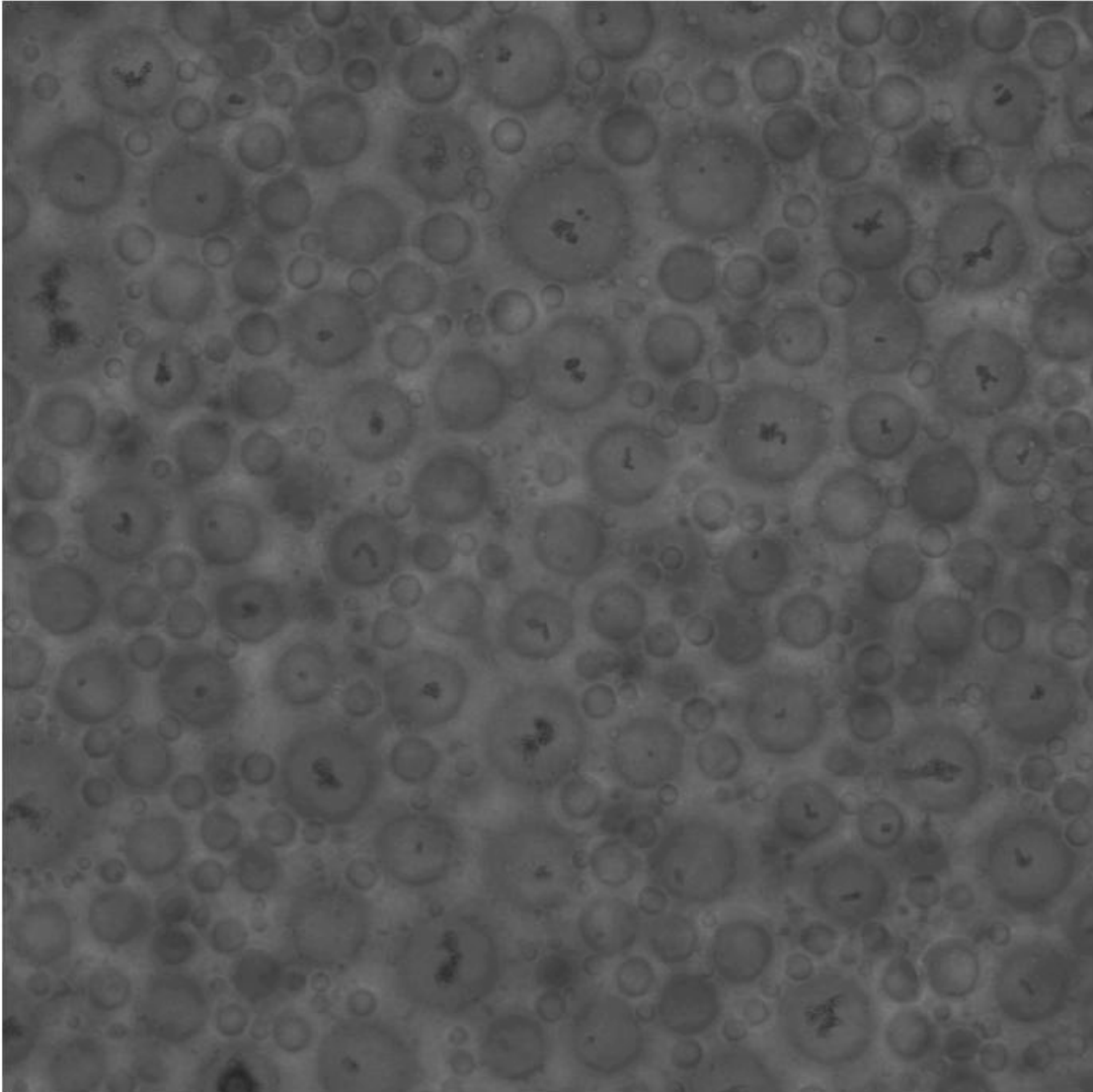


FIG. 56

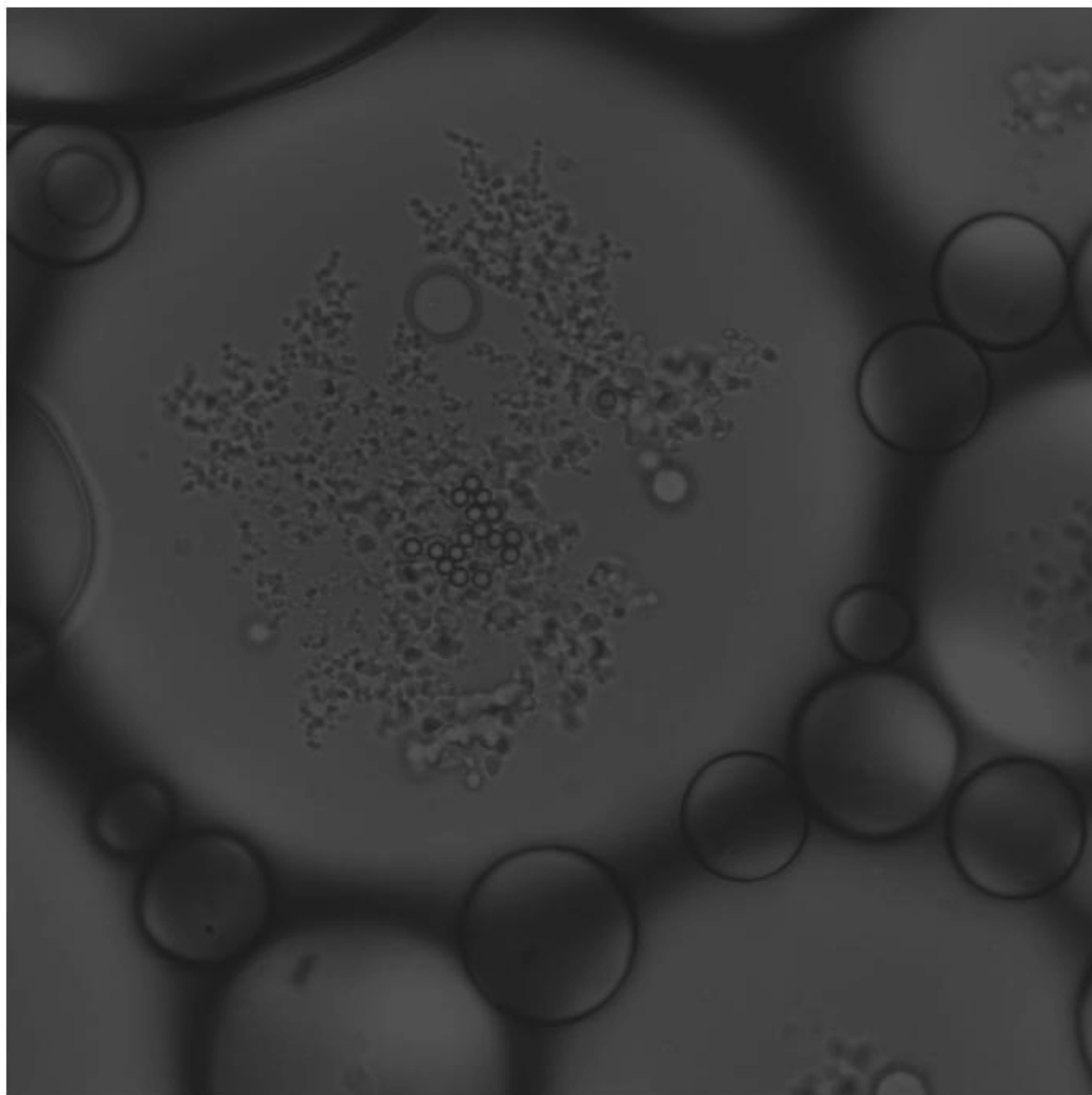


FIG. 57

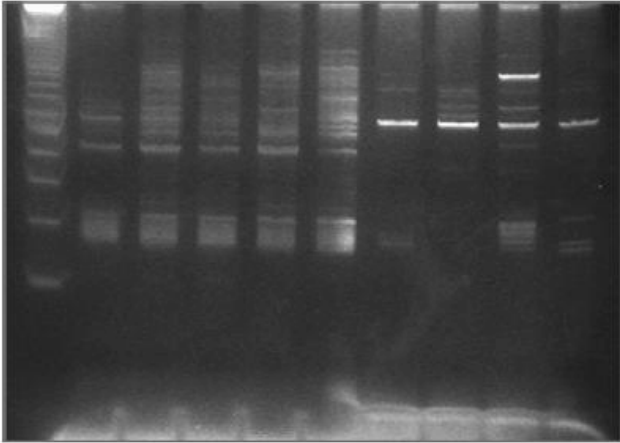


FIG. 58

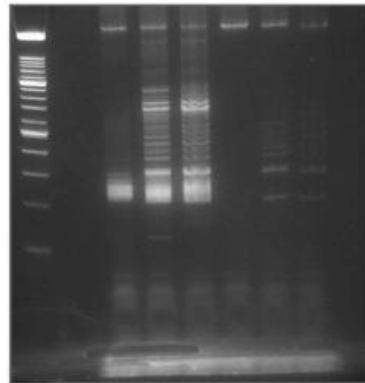
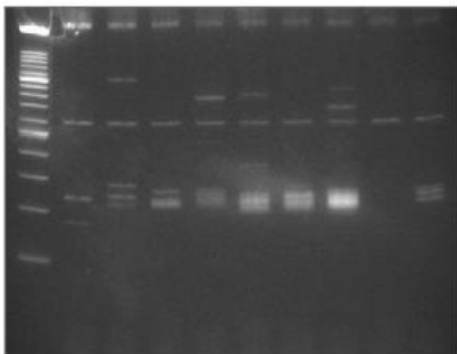
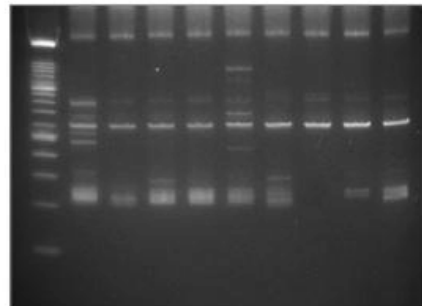
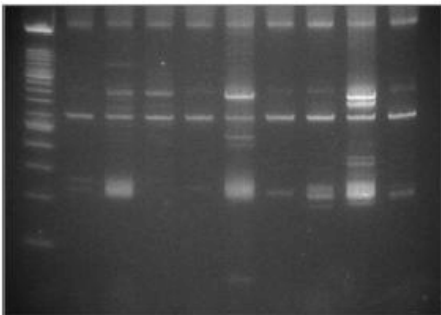
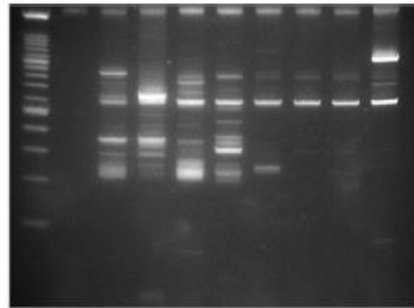
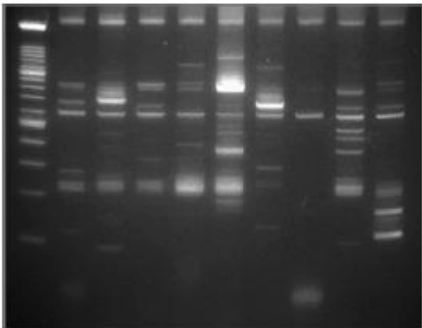


FIG. 59

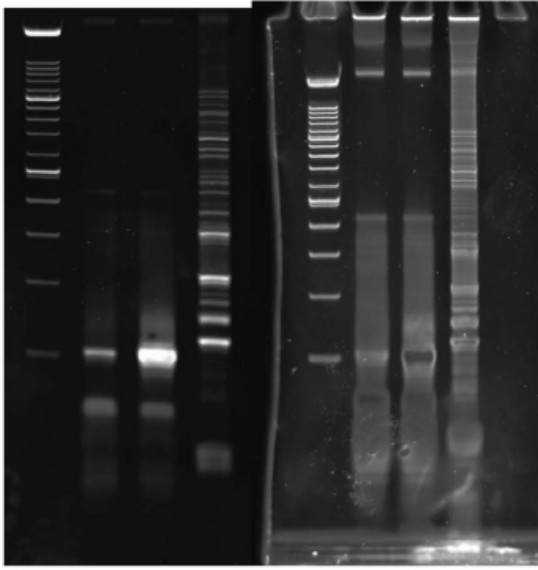


FIG. 60

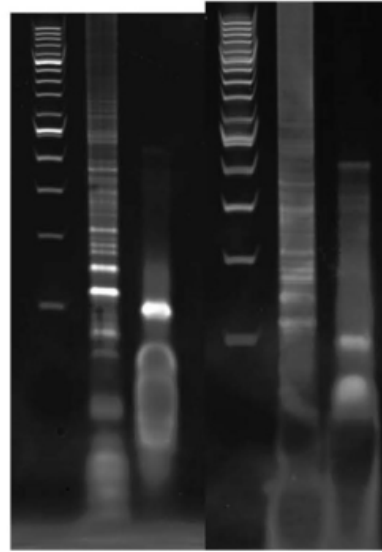


FIG. 61

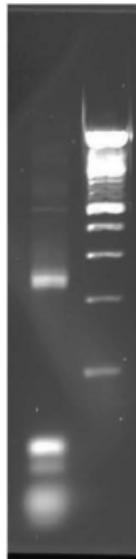


FIG. 62

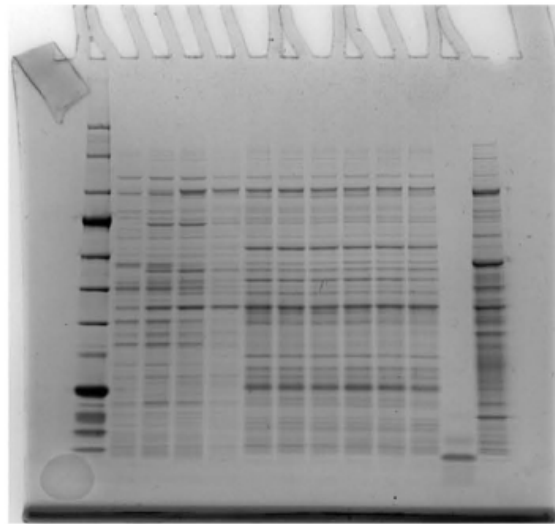


FIG. 63

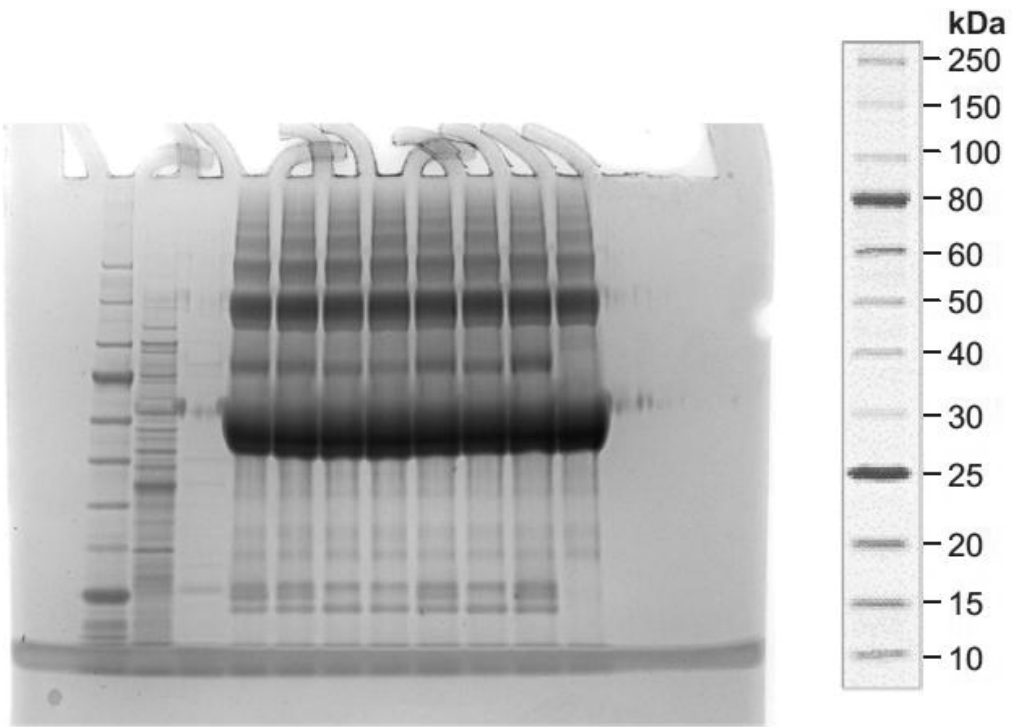


FIG. 64

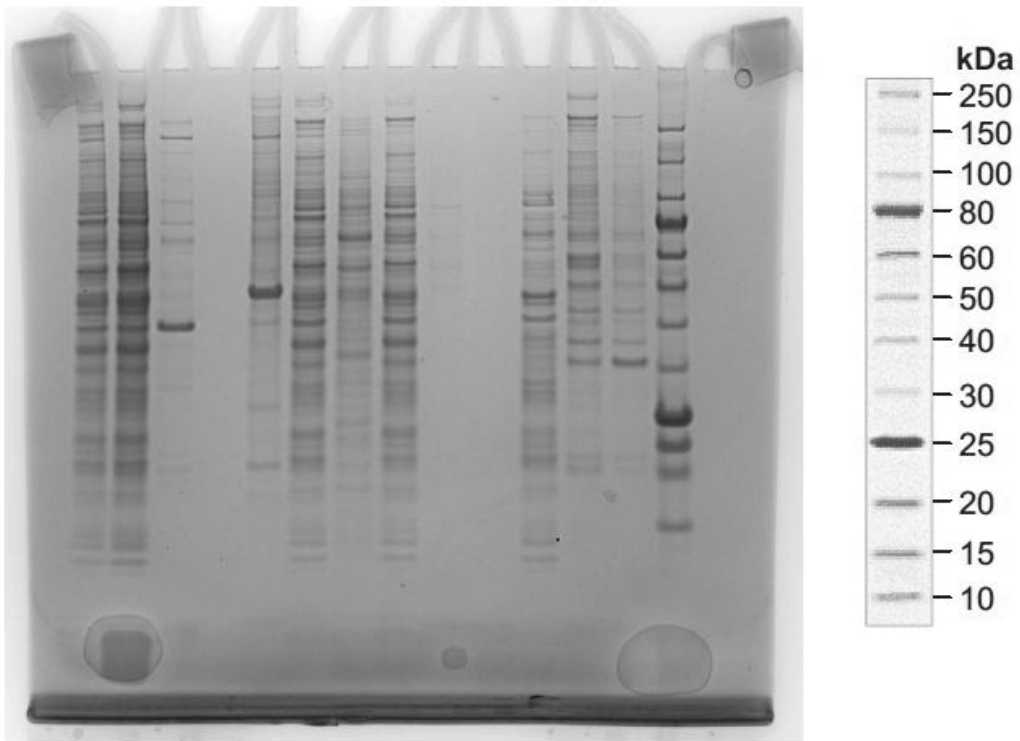
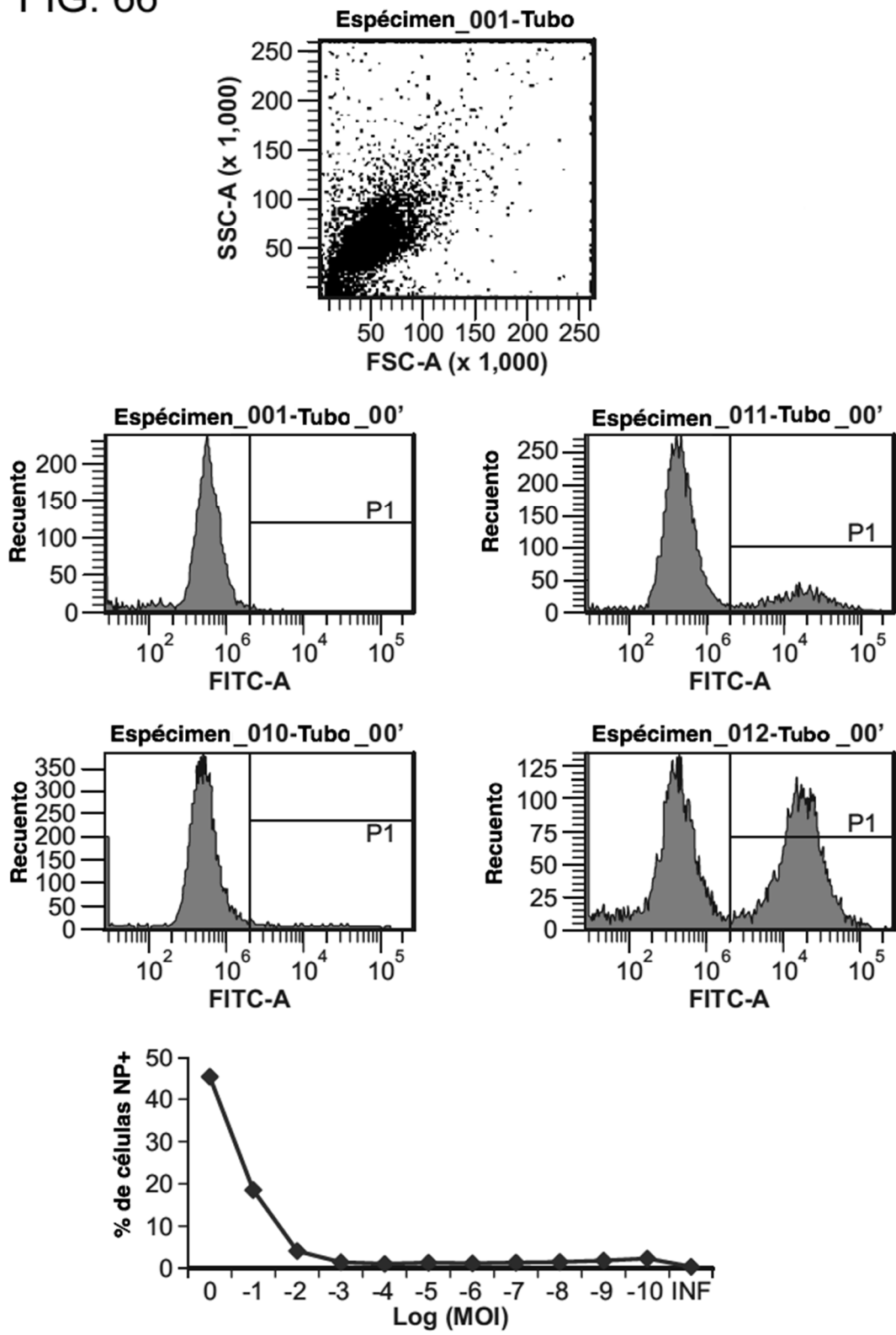


FIG. 65

FIG. 66



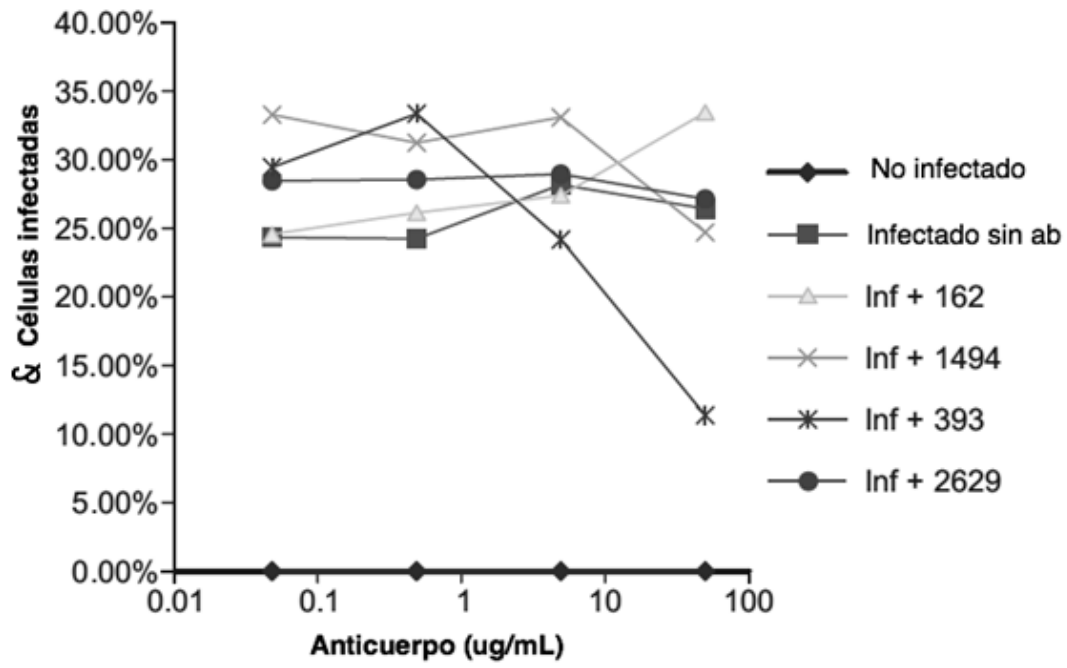


FIG. 67

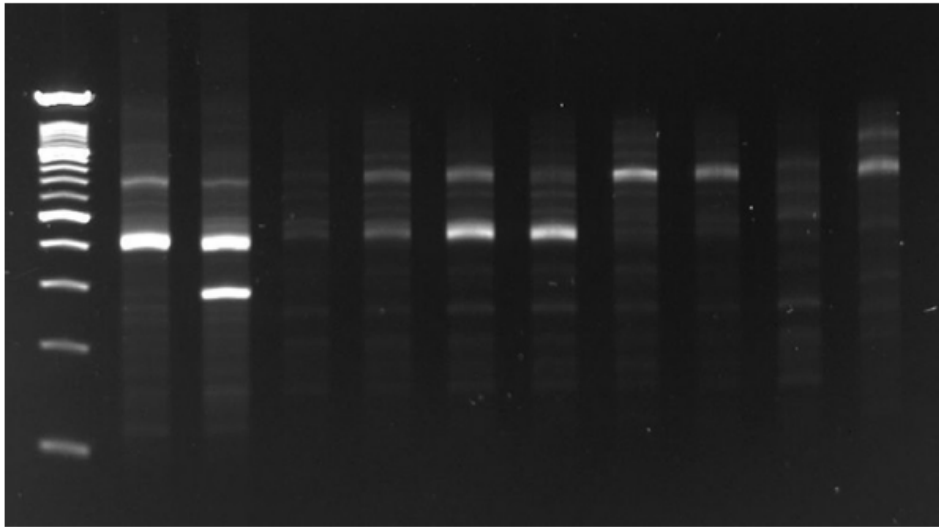


FIG. 68

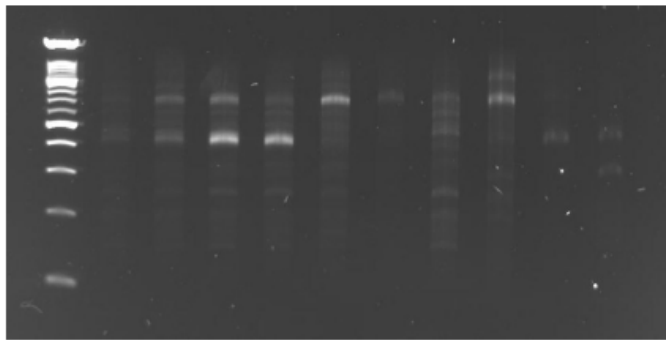


FIG. 69

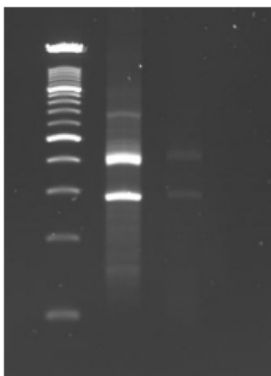


FIG. 70

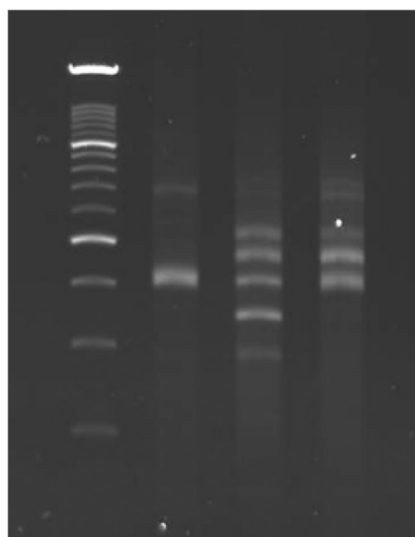


FIG. 71

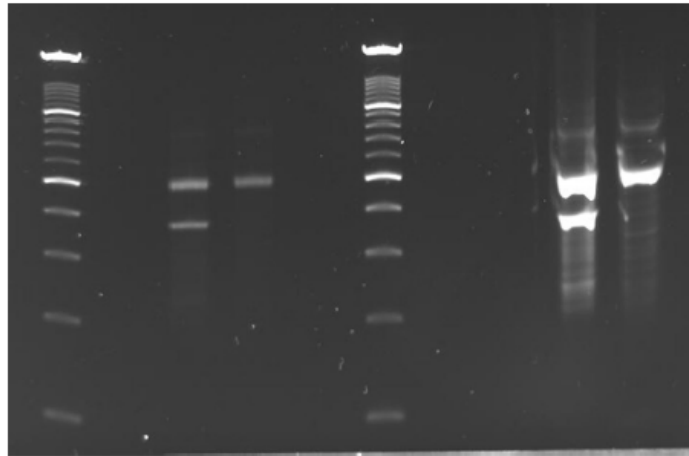


FIG. 72

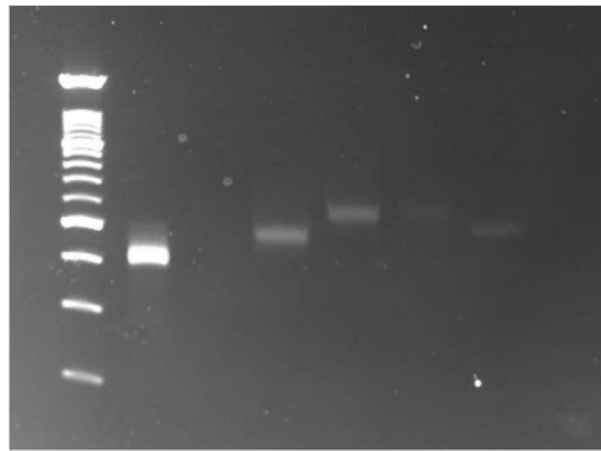
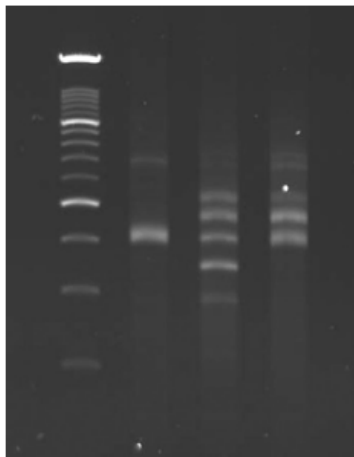


FIG. 73

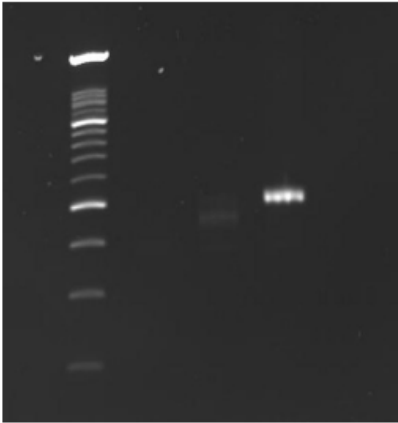


FIG. 74

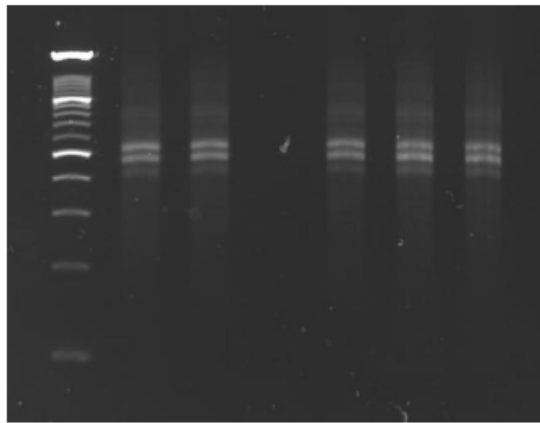


FIG. 75

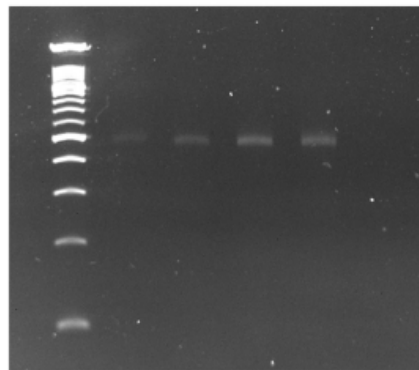
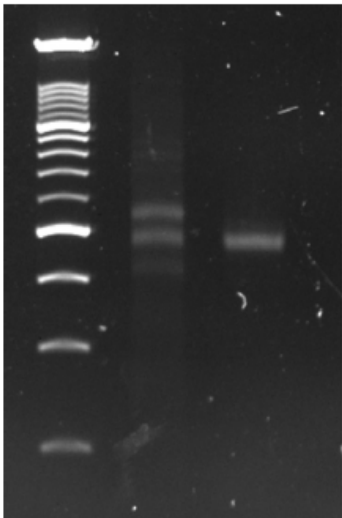


FIG. 76