

Análisis del archivo out222.txt – extractos del trayecto FRAY 1

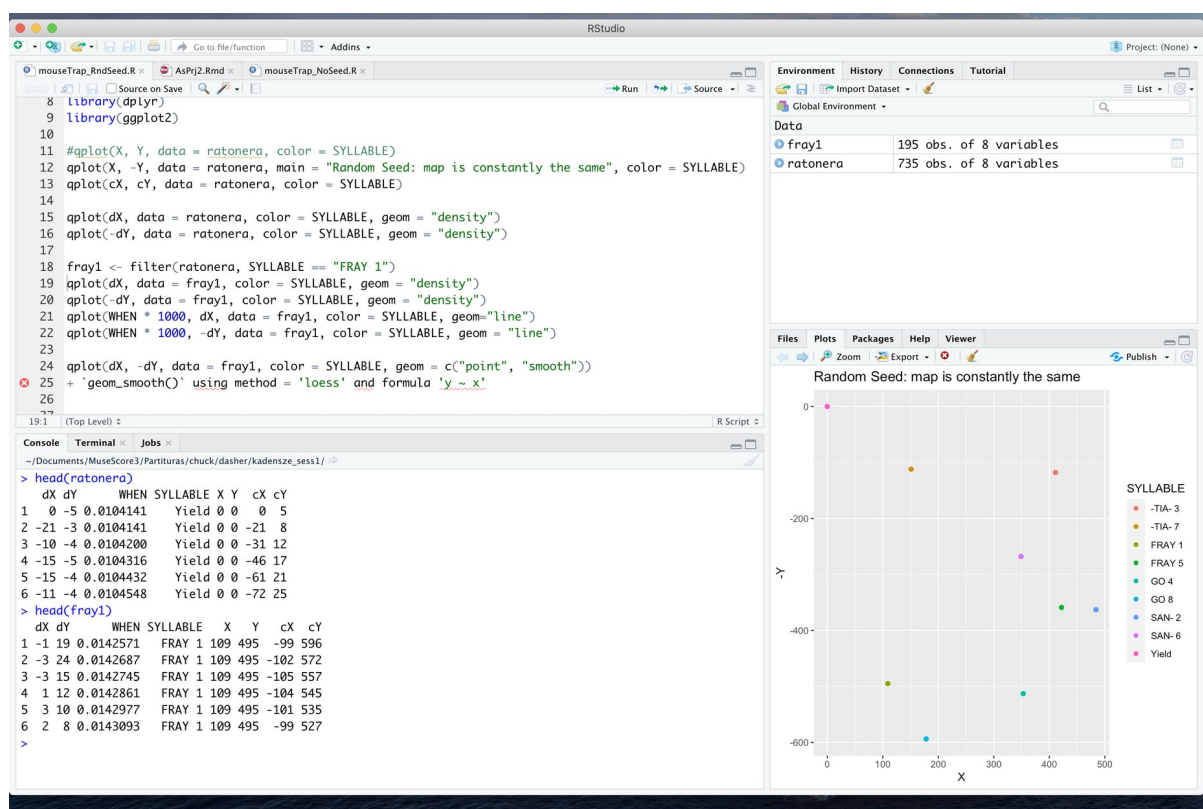


Figura 1: Mapa de botones. Encabezados de los dataframes "ratonera" y "fray1"

Columnas / campos de los “dataframes” *ratonera* y *fray1* en Rstudio:

- **dX** y **dY** son las *deltas* de los movimientos del ratón en sentido horizontal y vertical respectivamente.
- **WHEN** es el *tiempo* relativo de la máquina en el cual se dan dichos movimientos.
- **SYLLABLE** es la leyenda sobre el *botón / prompt* al que hay que dirigir el ratón y dar un click que despliegue el siguiente botón que propone un nuevo trayecto.
- **X** y **Y** son las coordenadas de cada *prompt* en la matriz de 500 x 600. En el mapa, las **Ys** han de ser graficadas con el signo cambiado para que repliquen la apariencia del ejercicio.
- **cX** y **cY** (cumulative X y cumulative Y, respectivamente) son las columnas que van sumando las dXs en cX; y, restando las dYs en cY a cada pulso de reloj capturado por los movimientos del ratón. El conjunto de estas coordenadas serán las responsables de replicar los trayectos que seguimos con el ratón para dar el siguiente click en el *prompt* actualmente desplegado en la ventana del ejercicio.

A continuación haremos un análisis prospectivo sobre las características del trayecto **FRAY 1**. En la siguiente lámina se muestra un mapa con las trayectorias seguidas. Los valores cY muestran la acumulación de las magnitudes dY con el signo cambiado. Es decir, que mientras el ejercicio en el ambiente gráfico del programa Chuck (miniAudicle) genera valores positivos y para desplazarse en

la ventana de arriba hacia abajo y viceversa (-dY), el rStudio grafica dichos valores de acuerdo a la manera como se grafica en la matemática tradicional. Ya hecho el cambio de signo, para graficar el siguiente “mapa con trayectorias”, no es necesario como en la lámina anterior, desplegar los valores del eje de las y (cY) con signo negativo; como en las y de la Figura 1: -Y.

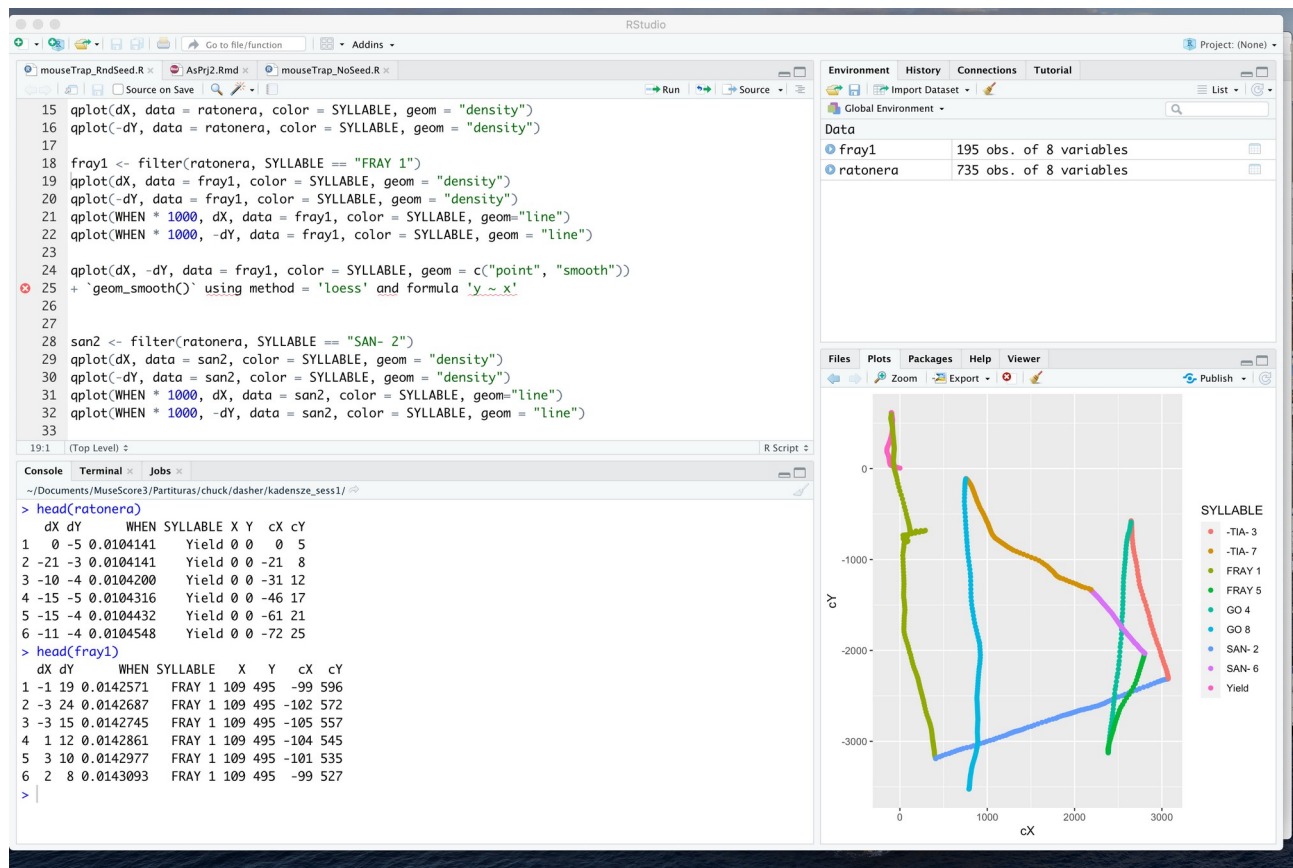


Figura 2: Mapa de trayectorias.

En la Figura 2 se puede ver en color fucsia el trayecto correspondiente al botón con la leyenda *yield* (esperar) que aporta al ejercicio las coordenadas de su origen relativo: X=0, Y=0; a partir del cual se acumularán en cX y cY las magnitudes delta de los movimientos del ratón de todo el ejercicio.

De modo que la Figura 2 muestra, después del comando `head(ratonera)`, los primeros movimientos positivos (ascendentes) reflejados en las coordenadas cY; y, el sesgo hacia la izquierda que se ve reflejado en los valores de coordenada negativos en cX. Es decir, el arranque de *yield* en fucsia.

Para poder observar, como se muestra en la Figura 3, a continuación, los sutiles movimientos hacia la izquierda (-1, -3, -3) del encabezado del extracto *fray1*, bajo el comando `head(fray1)`, en la columna dX de la Figura 2, es preciso hacer un zoom sobre el inicio del trayecto verde oliva de FRAY 1 (Figura 3), donde termina el fucsia del trayecto *yield*. Los correspondientes valores del eje vertical en dY (19, 24, 15) los acumulamos negativamente, es decir, los restamos en cY de el último valor del trayecto de *yield*, 615; y los primeros dos de FRAY 1: 596 y 572, respectivamente.



Figura 3: Inicio del trayecto FRAY 1, en verde oliva

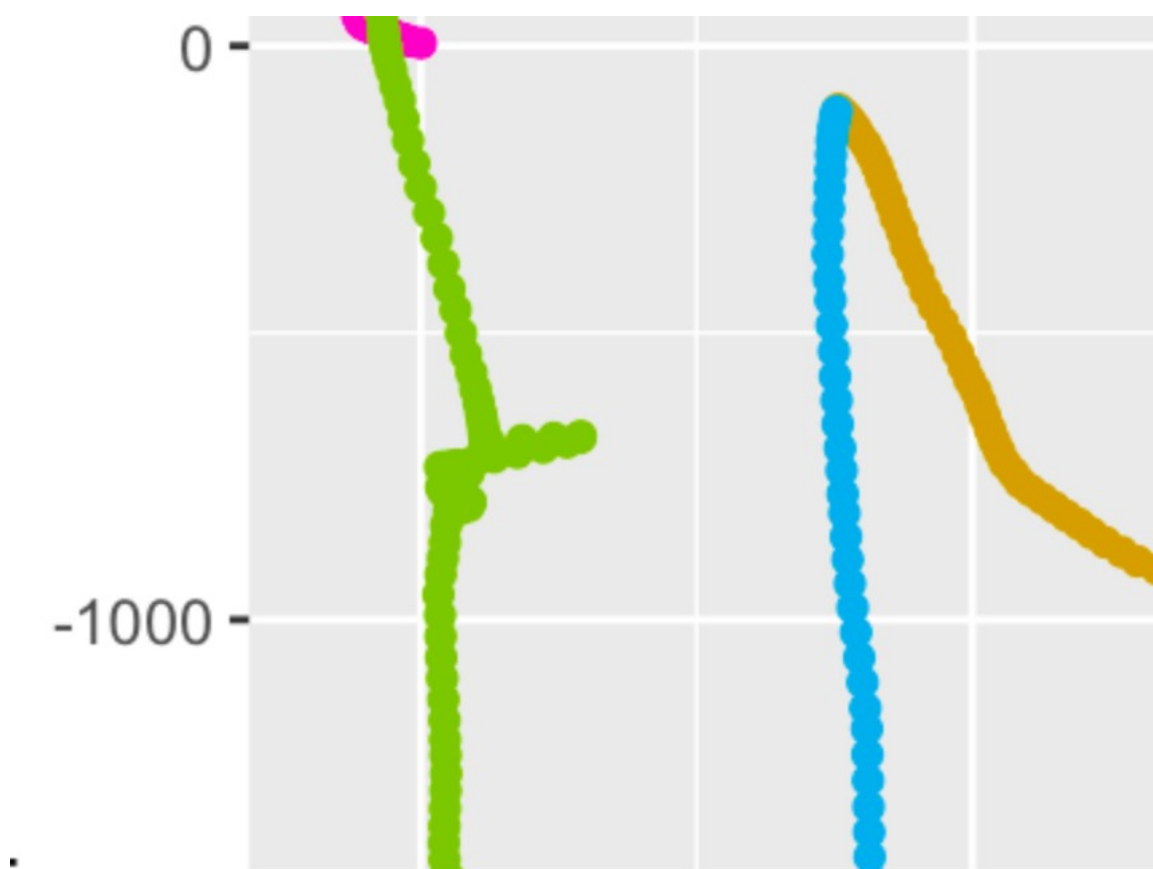


Figura 4: Amplificación del espasmo simulado en FRAY 1

El apoyo visual de la Figura 4, donde se muestra la gráfica del “espasmo” que se simuló en el trayecto FRAY 1 del ejercicio, permite aproximarnos al lugar (las coordenadas en cX y cY) donde se produjo el movimiento inesperado, y al dibujo grueso del trazo; pero no permite ver el detalle con claridad ya que se vuelve casi sobre los mismos pasos y se solapan los puntos en la gráfica. Para ver de manera nítida los detalles de dicho “espasmo”, ir a la Figura 9.

No se ven como solapamientos, en el caso de este archivo out222.txt , las reiteraciones de las coordenadas X y Y a lo largo de todos los valores de las observaciones que pertenecen a un único trayecto. X y Y en *yield* siempre serán 0; mientras que en FRAY 1, X será igual a 109 y Y será igual a -495 en el mapa, y 495, en la tabla de la Figura 1. En apariencia, un solo punto por cada trayecto.

A continuación, tres láminas con los datos del espasmo simulado propiamente dicho.

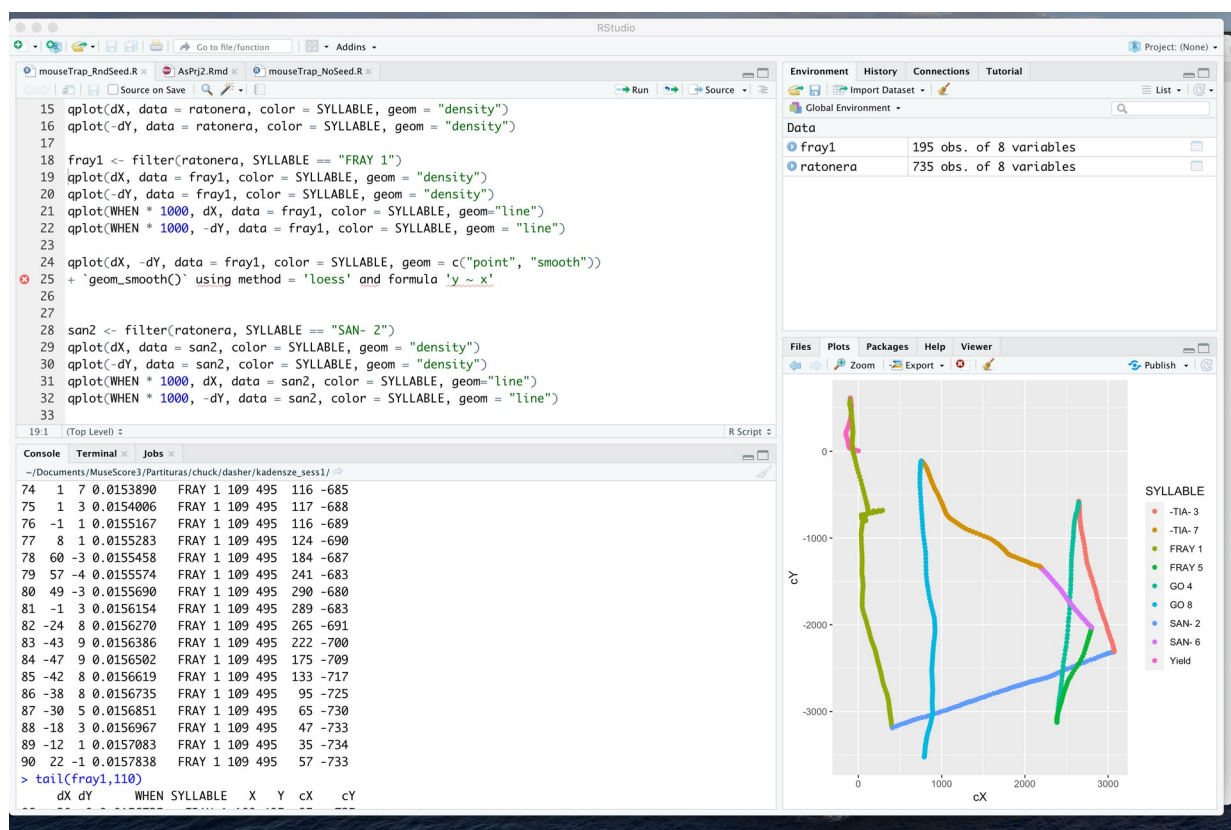


Figura 5: Espasmo (1/3)

En estos tres gráficos, estaremos haciendo énfasis en las magnitudes de las primeras dos columnas, dX y dY, correspondientes a los movimientos del ratón en el sentido de ambos ejes x e y; así como las de las últimas dos, cX y cY, correspondientes a las coordenadas de los puntos que integran las líneas de los trayectos en la gráfica.

En la Figura 5, el evento tipo espasmo muestra el inicio en la tabla a partir del renglón 75. De los renglones 76 al 80 hay un movimiento de izquierda a derecha y de abajo hacia arriba con excepción de la dYs en los renglones 76 y 77 que son igual a 1, hacia abajo; que son replicados en sentido contrario, de derecha a izquierda y de arriba hacia abajo, de los renglones 81 al 89 (Figura 9).

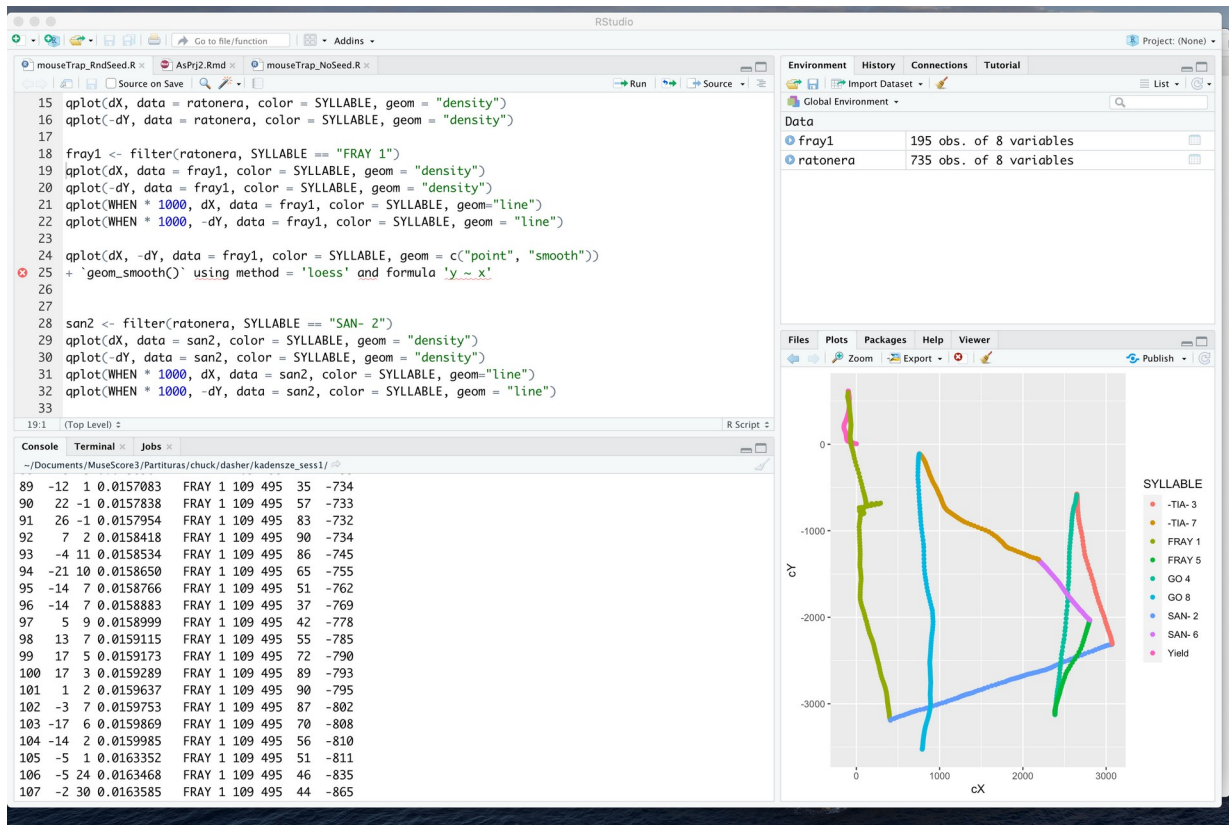


Figura 6: Espasmo (2/3)

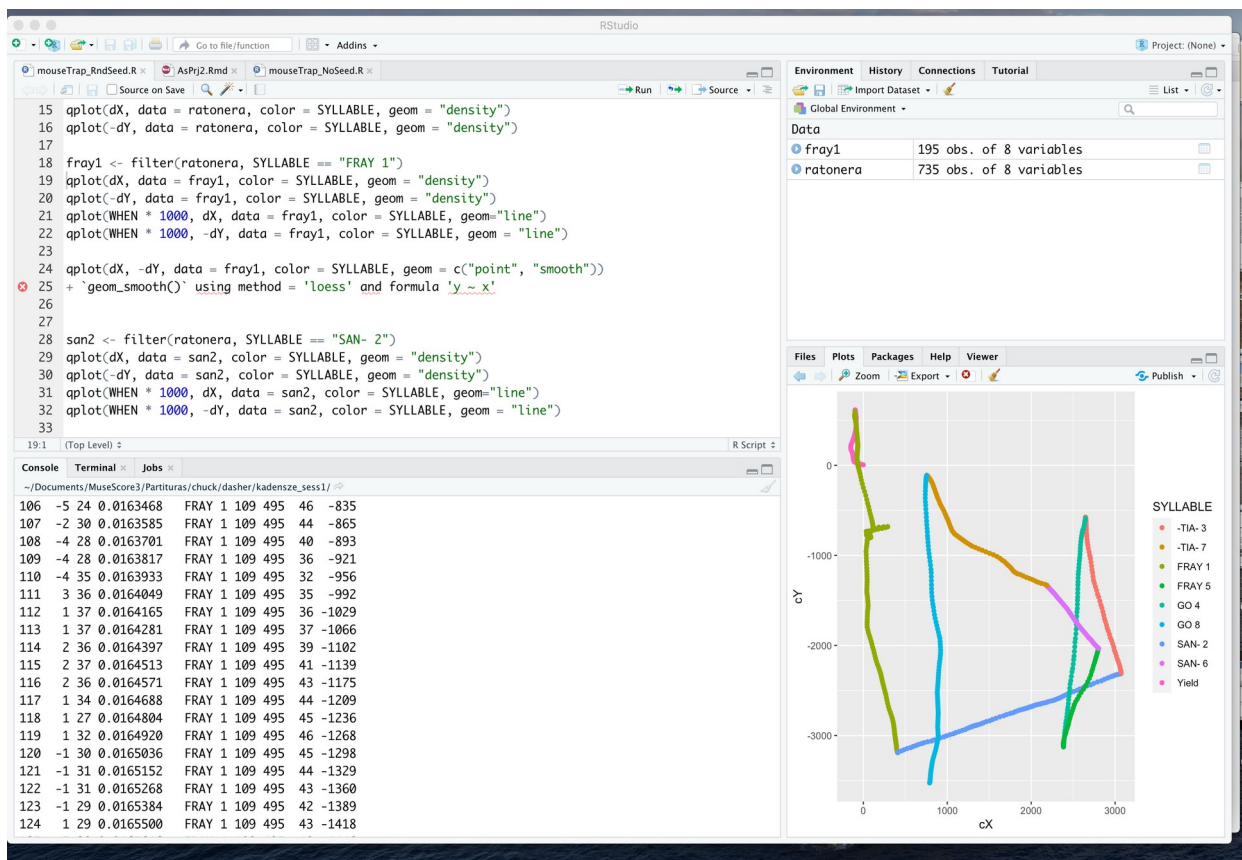


Figura 7: Espasmo (3/3)

En las siguientes láminas (Figuras 6 y 7) se ven registrados, de los renglones 89 al 105, hasta cuatro quiebros (derecha, izquierda, derecha, izquierda; Figura 9), que en una ausencia de espasmos, no formarían parte de un movimiento “más suave y directo” hacia las coordenadas del botón FRAY 1 (109,495) en la Figura 1; y las coordenadas 409,-3190 del fin del trayecto FRAY 1 tal como se muestra en la Figura 8 bajo el comando *tail(fray1)*.

A partir del renglón 111, la fluctuación en dX (Figura 7) del intervalo que va de 3 positivo a 1 negativo, denota el fragmento casi totalmente vertical que arranca alrededor de la cY -992, muy cerca del cuadrante etiquetado -1000, y que en la tabla de datos llega hasta la coordenada cY -1418.

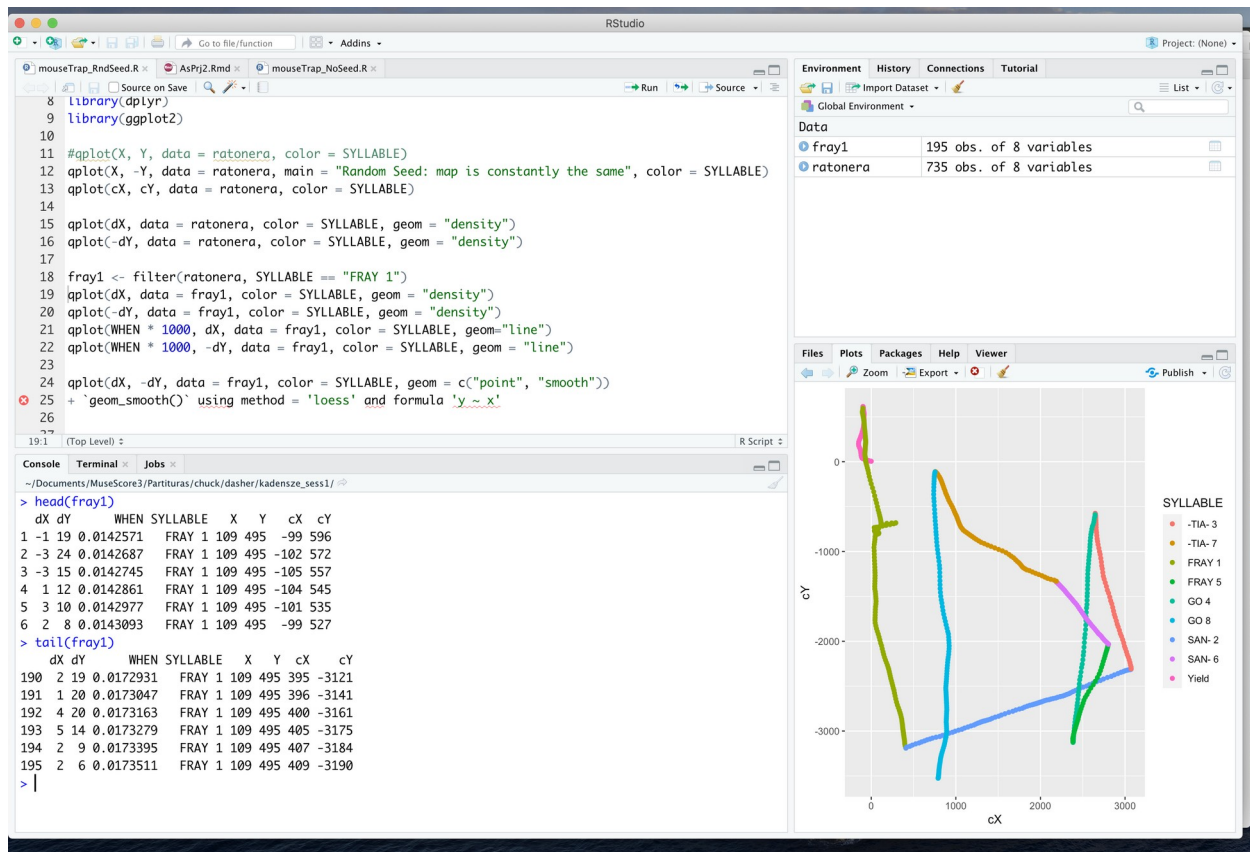


Figura 8: Inicio y fin del trayecto FRAY 1. De (-99, 596) a (409, -3190)

En resumen, este tipo de análisis en un número suficiente de capturas; alternando mapas idénticos de distribución de botones (como en este ejercicio) con otros de distribución random (al azar), deberá permitirnos identificar aquellos movimientos cuya estereotipicidad denote los diferentes patrones de espasmos en un usuario dado y, finalmente, filtrar dichos patrones y “corregir” el funcionamiento del Dasher (programa de introducción de texto sin uso de teclado) para que al usuario que incide en movimientos involuntarios estereotípicos le sea posible un uso más amigable y eficiente.

En el código fuente del Dasher, hasta donde hemos podido ver, candidatas a filtrar dichos patrones podrán ser aquellas secciones cuyo código controle y/o modifique la velocidad de presentación de los inicios de las palabras que el mismo programa predice, entre otros. ¿Se podrá entrenar el Dasher con estos patrones de espasmo de manera semejante a como los textos en diferentes idiomas y el uso del usuario mismo, entrenan el programa?

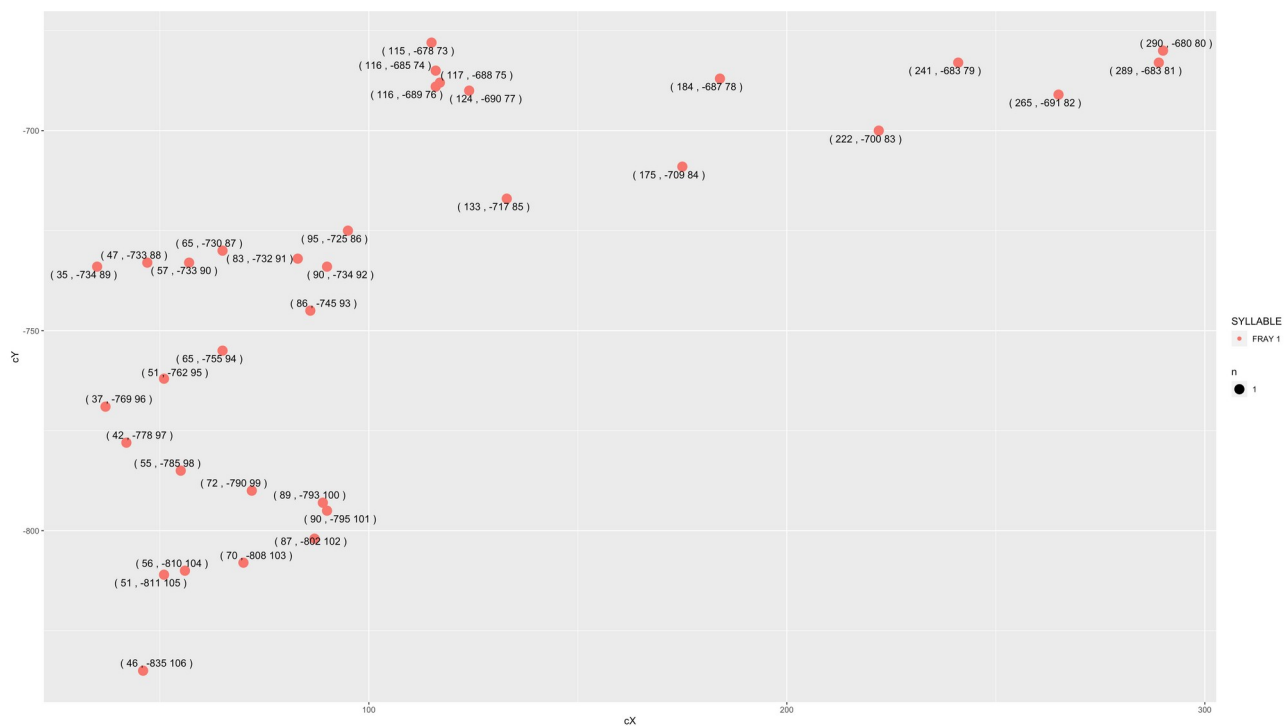


Figura 9: El espasmo simulado. Coordenadas cX, cY + número de renglón en la tabla.