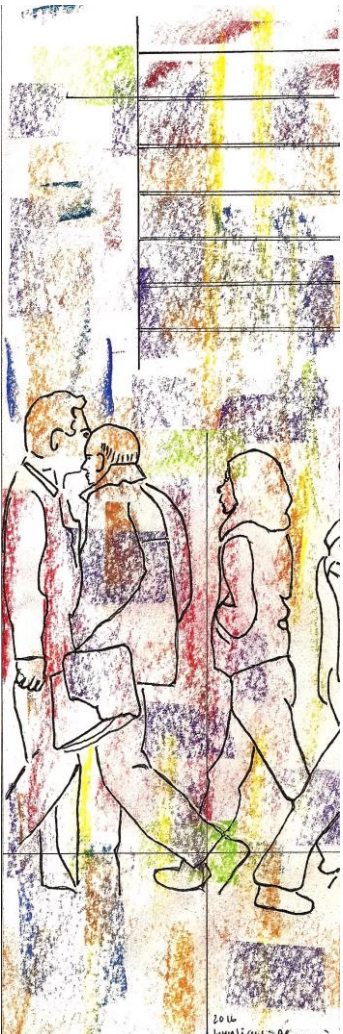
INCASI
Working Paper Series

2018, No. 4

INCASI *International Network for Comparative Analysis of Social Inequalities*



La operativización del concepto de trayectoria con TraMineR. Una introducción al análisis de secuencias y al Optimal Matching

Lídia Yepes-Cayuela

European Commission | Horizon 2020 European Union funding for Research & Innovation

Marie Skłodowska-Curie Actions (MSCA)
Research and Innovation Staff Exchange (RISE)
H2020-MSCA-RISE-2015
GA-691004

La operativización del concepto de trayectoria con TraMineR. Una introducción al análisis de secuencias y al Optimal Matching

Lídia Yepes-Cayuela [1]

[1] Centre d'Estudis Sociològics sobre la Vida Quotidiana i el Treball
Institut d'Estudis del Treball
Universidad Autónoma de Barcelona, España
lidia.yepes@uab.cat

European Commission
Horizon 2020
European Union funding
for Research & Innovation

Marie Skłodowska-Curie Actions (MSCA)
Research and Innovation Staff Exchange (RISE)
H2020-MSCA-RISE-2015
GA-691004

This working paper was elaborated in the context of INCASI Network, a European project that has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie GA, No. 691004, and coordinated by Dr. Pedro López-Roldán. This article reflects only the author's view and the Agency is not responsible for any use that may be made of the information it contains.

Digital edition: https://ddd.uab.cat/record/199390

Dipòsit Digital de Documents
Bellaterra, Cerdantola del Vallès (Barcelona)
Universitat Autònoma de Barcelona

INCASI Working Paper Series
2018, No. 4
Published online: https://ddd.uab.cat/record/199390

INCASI *International Network for Comparative Analysis of Social Inequalities*

La operativización del concepto de trayectoria con TraMineR. Una introducción al análisis de secuencias y al Optimal Matching

Lídia Yepes-Cayuela

Resumen

En este *paper* se presenta el procedimiento técnico para operativizar trayectorias mediante el programa TraMineR, un paquete estadístico de R. Se expondrá brevemente que es el análisis de secuencias y el Optimal Matching Analysis (Análisis de Alineación Óptima) así como la sintaxis básica. Aunque existen otras metodologías, estas técnicas han sido probadas como una de las más convenientes para el análisis de datos longitudinales.

Palabras Clave

Análisis de secuencias, Optimal Matching Analysis, metodología, trayectorias, TraMineR

Índice

1. Introducción

La metodología longitudinal, a pesar de seguir siendo poco frecuente en España en comparación con otros contextos (como el norteamericano, alemán o inglés para poner algunos ejemplos) ha acaparado un mayor interés en los últimos años, tanto desde el punto de vista teórico, enfatizando las cuestiones dinámicas de la vida social, como metodológico. En este segundo punto, han jugado un papel importante los avances tecnológicos e informáticos, cada vez más sofisticados, que permiten operativizar conceptos anteriormente desarrollados teóricamente, como el concepto que aquí nos ocupa, el de "trayectoria".

Este *working paper* surge con el objetivo de esclarecer cual ha sido el procedimiento para operativizar cuantitativamente el concepto de "trayectoria", laboral en este caso, pero la misma metodología se puede aplicar para cualquier otro tipo de dato como por ejemplo referido a trayectorias educativas, migratorias o residenciales para poner algunos ejemplos. Aunque no es la única manera, aquí se ha elegido el análisis de secuencias.

En este *working paper* explicaremos qué es el análisis de secuencias y expondremos un caso práctico que nos sirve como excusa para ejemplificar las técnicas. Los datos utilizados son resultado de un trabajo de campo realizado bajo el marco del proyecto 'Las redes sociales en sus diferentes modalidades, como recursos y mecanismos de búsqueda e inserción laboral en el empleo y de apoyo social en los jóvenes' (REDEMAS) a 250 jóvenes de entre 20 y 34

años del Área Metropolitana de Barcelona (de la ciudad de Barcelona, Sant Feliu de Llobregat, L'Hospitalet de Llobregat y Santa Coloma de Gramenet) que trabajan o buscan trabajo (con experiencia laboral previa). Este proyecto incluía como parte de sus objetivos analizar cómo eran las trayectorias laborales de las personas jóvenes incluidas en la muestra.

Los análisis se han realizado con el programa R y específicamente con un paquete asociado al mismo llamado TraMineR desarrollado por investigadores de la Universidad de Ginebra (Gabadinho, Ritschard, Müller, y Studer, 2011). Este programa resulta especialmente adecuado para tratar datos secuenciales en las ciencias Sociales.

Expondremos la sintaxis para poder ejecutar algunas de las funciones más útiles que se incluyen en este programa como son la visualización gráfica de las trayectorias, el cálculo del tiempo medio en cada estado y el análisis de las transiciones.

Finalmente, explicaremos con más detalle el proceso de creación de la tipología de trayectorias mediante la técnica del Optimal Matching Analysis (OMA) reconocida como uno de los métodos más adecuados para el análisis de datos secuenciales.

2. El análisis de secuencias

El término *sequence analysis* se aplica a multitud de objetos de estudio ya que los datos secuenciales están presentes en muchas disciplinas distintas, algunas muy alejadas de la sociología como la física o la biología. Una secuencia es una lista ordenada de cosas (pueden ser números, eventos, estados o cualquier otra cosa) dentro de un alfabeto (Abbott y Forrest, 1986; Ritschard, 2012). Aunque para poder hablar de datos secuenciales no es obligatorio el orden cronológico (por ejemplo, en biología se utilizan las cadenas de proteínas o las secuencias de ADN) las secuencias que nos interesan aquí sí que tienen en cuenta el orden temporal. Por lo tanto, el análisis de secuencias que trataremos en este *paper* se refiere al estudio de datos longitudinales (no transversales), ordenados cronológicamente.

La metodología longitudinal es básica e imprescindible para captar distintos fenómenos sociales. Por ejemplo, para la sociología de la juventud desde la perspectiva de itinerarios y trayectorias que considera la juventud como un proceso de transición (Casal, Merino y García, 2010). El concepto de trayectoria (o *pathway* en inglés, que invoca a la idea de camino) sirve para definir una serie de eventos (laborales, familiares, o del tipo que sean, en función del objeto de estudio) que se van sucediendo a lo largo de la vida. De esta manera no se trata de una serie de sucesos aislados uno detrás del otro, sino que la noción de trayectoria remite a la idea de interconexión en que los acontecimientos pasados influyen en los presentes y los futuros, lo que es denominado como *path dependency* (Billari, 2001; Mayer, 2001; Verd y López-Andreu, 2011).

Los datos longitudinales se diferencian de los transversales en la medida en que recogen información de la unidad de análisis en más de un momento en el tiempo (Belvis y Benach, 2013). Cabe subrayar que los datos transversales repetidos (*repeated cross sectional* o *trend data* en inglés), es decir comparar datos transversales recogidos en momentos distintos (por ejemplo, comparar la tasa de paro en 2014 y en 2015) no se consideran datos longitudinales ya que las unidades de análisis son distintas en cada momento (Belvis y Benach, 2013). Así, los datos transversales capturan la foto fija en un momento determinado, pero no permiten analizar la evolución de esa unidad de análisis a lo largo del tiempo.

En los estudios incluidos en la perspectiva del curso de vida, las secuencias se refieren a los diferentes estados (eventos, actividades) por los que pasan los individuos a lo largo de su biografía. Los estados analizados tendrán que ver con el objeto de estudio y pueden ser tan variados como los relacionados con la vida familiar y matrimonial (por ejemplo, estar soltero, casado, viudo, con hijos, etc.), con trayectorias migratorias o residenciales (vivir en el país de origen, vivir en casa de los padres,

constituir hogar propio, etc.) o, como en este caso, eventos relacionados con el ámbito laboral (trabajar, estar desempleado, estudiar, etc.).

La particularidad de usar datos secuenciales en el estudio de trayectorias es que permite tratar las trayectorias como una sola unidad de análisis (Abbott, 1990; Abbott y Tsay, 2000; Robette, 2010; Gabadinho et al. 2011). La concepción holística de la trayectoria es un punto muy importante ya que marca la diferencia con otras metodologías. Una de sus principales ventajas es la posibilidad de crear tipologías como veremos en el apartado 3 de este *paper*.

Las secuencias manejadas en este *paper* se incluyen en el grupo de secuencias "recurrentes" ya que los estados se pueden repetir, en oposición a las secuencias no-recurrentes (*permutation*) dónde los estados sólo aparecen una vez (Abbott y Tsay, 2000).

Una de las principales dificultades es tener los datos en el formato adecuado para ser susceptibles de ser analizados como datos secuenciales. La manera en que codificamos los datos va a condicionar de manera irreversible los resultados por lo que es un punto importante a tener en cuenta (Abbott y Tsay, 2000; Solís y Billari, 2003). Abordaremos con más detalle esta cuestión en el siguiente apartado.

Además de la codificación de los datos en los diferentes estados, es importante la temporalidad. La mayoría de las investigaciones utilizan intervalos de tiempo regulares, normalmente de un año, por lo que se recogen los estados anualmente. Por ejemplo, si la investigación trata de los cambios en los hogares, habrá un estado (hogar paterno, hogar propio, etc.) para cada año. Por la complejidad de la realidad juvenil en el contexto español y catalán (Castelló et al., 2013; Serracant, 2014), creímos pertinente utilizar intervalos de tiempo mensuales en lugar de anuales, como suele ser habitual, por lo que nuestros datos aportan una riqueza mucho mayor a la de otras investigaciones que tratan temas similares.

2.1 Programa estadístico utilizado

Para realizar las explotaciones que se detallarán a continuación se ha utilizado el programa estadístico R y concretamente un paquete del mismo que se llama TraMineR (Gabadinho et al., 2011). El paquete TraMineR fue desarrollado en la Universidad de Ginebra por un grupo de investigadores interesados en el análisis de secuencias y especialmente concebido para las ciencias sociales lo que lo hace muy pertinente para nuestro análisis. Aunque es muy popular y utilizado en muchos países, en España sigue siendo bastante desconocido. A pesar de que las investigaciones con interés para añadir la dimensión temporal y hacer análisis longitudinales crecen en nuestro contexto, una de las mayores dificultades es encontrar bases de datos pertinentes y suficientemente completas. Ese fue uno de los motivos por los que decidimos hacer un trabajo de campo propio para recoger nuestros propios datos.

El paquete TraMineR, al igual que R en su conjunto, está en constante evolución ya que, al ser un software libre, es desarrollado por una comunidad en la red muy activa. Aunque hay otros programas estadísticos que permiten realizar análisis de secuencias (como Stata) nos decantamos por utilizar R para manejar nuestros datos y realizar los distintos análisis cuantitativos por diversas razones:

1) La primera y más importante es la existencia del paquete TraMineR que tiene como objetivo el análisis de datos secuenciales.
2) Además, TraMineR fue especialmente concebido para temáticas relacionadas con las ciencias sociales (Sociología, Demografía, Economía, etc.) hecho que facilita su uso (Gabadinho et al., 2011).
3) De hecho, los investigadores mencionados se inscriben dentro del paradigma de la *life course perspective* y son abundantes las investigaciones que utilizan el programa que comparten base teórica con nuestro proyecto de

investigación, por lo que es especialmente apropiado para nuestros análisis[1].

4) El objetivo de este programa es simplificar y categorizar la información secuencial para poder trabajarla de manera cómoda. Así, una de sus principales ventajas es la capacidad de manejar un gran volumen de datos secuenciales de manera relativamente sencilla. Mediante distintas funciones, el programa permite ordenar, agrupar y comparar las secuencias (en nuestro caso trayectorias laborales) para luego ser susceptibles de ser utilizadas posteriormente con métodos inferenciales clásicos como por ejemplo regresiones.
5) Permite distintos formatos de datos. Aunque tiene formatos predeterminados, acepta otro tipo distinto de datos después de un trabajo previo de preparación.
6) Ofrece múltiples opciones de visualización.
7) Permite analizar características longitudinales de las secuencias tanto de manera individual como transversal o agrupada. Si bien podemos analizar las secuencias de forma individual, existe toda una corriente que analiza las secuencias desde un punto de vista agregado (Widmer y Ritschard 2009, Robette, 2010). Desde esta perspectiva, por ejemplo, se pueden observar de manera agregada para la muestra o un subgrupo seleccionado que estados son los más comunes a una edad concreta o comparar la evolución de estos estados en función de otras variables como el sexo, la cohorte, el nivel de estudios, etc.
8) Incluye distintos métodos para calcular la disimilitud y distancia entre las secuencias y crear tipologías. De esta manera se pueden realizar los cálculos para generar la matriz de distancias mediante la técnica del Optimal Matching que explicaremos a continuación.
10) Es un programa de acceso libre y gratuito.
11) Está en constante desarrollo por una comunidad de internautas muy activa en la red que, además de proveer ayuda, van ampliando las funciones y las herramientas disponibles en el programa.

2.2. Formato de los datos

Para poder utilizar las funciones incluidas en el paquete TraMineR es necesario disponer de los datos en el formato correcto. Esta es una de las partes más complejas y que lleva más tiempo, pero una vez se dispone de los datos en este formato las funciones se ejecutan de manera bastante rápida y sencilla.

Hay distintos formatos aceptados por TraMineR, pero el formato por defecto y más apropiado es el State-Sequence (STS). En la tabla 1, podemos observar los distintos formatos soportados por el programa:

Tabla 1. Formatos de TraMineR

Code	Conversion	Example
STS	from/to	*Id* 18 19 20 21 22 23 24 25 26 27 101 S S S M M MC MC MC MC D 102 S S S MC MC MC MC MC MC MC
SPS	from/to	*Id* 1 2 3 4 101 (S,3) (M,2) (MC,4) (D,1) 102 (S,3) (MC,7)
DSS	to	*Id* 1 2 3 4 101 S M MC D 102 S MC
SPELL	from	*Id* *Index* *From* *To* *State* 101 1 18 20 S (single) 101 2 21 22 M (married) 101 3 23 26 MC (married with children) 101 4 27 27 D (divorced) 102 1 18 20 S (single) 102 2 21 27 MC (married with children)

Fuente: Gabadinho, Ritschard, Müller y Studer (2011: 9).

En el formato por defecto, el STS, cada fila corresponde a un caso, mientras que cada columna es la unidad temporal. La información de cada casilla corresponde a cada posible estado. Por el contrario, la forma original de nuestros datos estaba en formato SPELL por lo que cada fila era un evento. En el ejemplo del cuadro, cada caso (ID) se repite tantas veces como eventos tenga su trayectoria. La variable "Index" nos indica el orden de los eventos, mientras que tanto la variable "from" como la "to" nos muestran la unidad temporal.

[1] Para consultar dichas investigaciones se puede visitar la página web del proyecto LIVES dónde están publicadas en abierto: https://www.lives-nccr.ch/fr/view/biblio/year

Aunque es deseable tener los datos en formato inicial STS, existen funciones que permiten convertir los distintos formatos al STS.

Estos datos pueden estar en una matriz de spss (o en otro programa) y ser exportados a R como explicaremos inmediatamente.

2.3. Sintaxis básica

Empezar

En este apartado presentaremos brevemente la sintaxis básica para empezar a trabajar el análisis de secuencias con el paquete TraMineR de R. Se asume que el lector tiene ciertos conocimientos básicos de R. Para intentar facilitar la lectura de la sintaxis, las partes que no hay que modificar están marcadas en azul, mientras que el resto se refiere a nombres o números que el autor ha de escribir acorde a sus datos.

En primer lugar hay que descargar el paquete TraMineR y abrirlo, para ello utilizaremos la siguiente sintaxis:

```
Install.packages ("TraMineR")
Library ("TraMineR")
```

En segundo lugar hay que exportar la matriz de datos y abrirlos en R. Si la matriz proviene de otros programas hay que descargarse también el paquete "foreign" con la misma sintaxis que acabamos de ver:

```
Install.packages ("Foreign")
Library ("Foreign")
```

Para abrir una matriz de datos podemos utilizar el siguiente comando:

```
Nombre_objeto de R <- read.spss ("nombre de la matriz en spss.sav", to.data.frame = TRUE)
```

Donde "read.spss" es el nombre de la función, "nombre_objeto de R" es el nombre que nosotros le damos a la matriz que se abrirá en R y "nombre de la matriz en spss.sav" el nombre previo que tenía el archivo en formato spss. El "nombre_objeto de R" será abreviado como "OR" a partir de ahora en sintaxis posteriores.

Una vez tengamos la matriz cargada ya podemos empezar.

El objeto de secuencia

El primer paso antes de ejecutar las distintas funciones incluidas en el paquete TraMineR es crear un objeto de secuencia (*State Secuence Object*). Esta es la parte principal y más importante para realizar los cálculos y análisis con ya que el "objeto de secuencias" (OSEQ a partir de ahora) es la base para toda la sintaxis posterior.

Mediante este objeto el programa entiende que cada una de las filas es una secuencia -en nuestro caso, una trayectoria laboral- y cada columna es la unidad temporal. De esta manera cada trayectoria es tratada como una sola unidad de análisis.

Para crear el OSEQ hay que tener en cuenta los siguientes elementos:

– **Alfabeto:** mediante la creación del alfabeto definimos dentro del OSEQ cuáles son los estados incluidos en el análisis, es decir, el tipo de evento incluido en la trayectoria. Por ejemplo, en nuestro caso los estados son: estudiar, estar empleado con un contrato temporal, estar empleado con un contrato estable, desempleado, etc.

  Estos estados deben haber sido definidos previamente en la matriz (como variables de tipo factor) que acabamos de cargar y tienen que estar escritos exactamente igual, es decir que si en la matriz el estado es "contrato indefinido" en el alfabeto escribiremos "contrato indefinido". Para evitar posibles problemas, se recomienda que se simplifiquen los nombres y que no se incluyan caracteres especiales tales como tildes o la ñ. Para saber cuáles son se puede utilizar esta función:

  ```
  seqstatl (OR [, 1:N])
  ```

  Donde OR es el nombre de la matriz que le damos en R que acabamos de abrir, 1 es el número de columna dónde se encuentra la

variable "estado" y N el número de filas (u observaciones).

En segundo lugar, hay que crear el alfabeto y las etiquetas. La sintaxis que genera el alfabeto es la siguiente:

```
OR.alphabet <- c("nombre del estado 1", "nombre del estado 2", "nombre del estado 3")
```

Aunque el programa soporta tantos estados como se desee, no es recomendable más de 9 o 10 ya que las interpretaciones posteriores de los resultados pueden ser muy complicadas.

- **Etiquetas**: mediante las etiquetas se pueden modificar los nombres de los estados que acabamos de definir, ya que no hace falta que sean los mismos que estaban en la matriz de datos original. Estas etiquetas son los nombres que aparecerán en la leyenda.

```
OR.labels <- c("etiqueta del estado 1", " etiqueta del estado 2", " etiqueta del estado 3")
```

- **Etiquetas cortas**: las etiquetas cortas aparecerán en los gráficos y pueden ser también distintas tanto a los estados incluidos en el alfabeto como a las etiquetas que acabamos de definir. La sintaxis para generar las etiquetas cortas es la siguiente:

```
OR.shortlab <- c("etiqueta corta del estado 1", " etiqueta corta del estado 2", " etiqueta corta del estado 3")
```

- **Paleta de colores**: a cada estado se le asigna un color predeterminado que se puede modificar con la siguiente sintaxis:

```
cpal_OR= c("nombre del color para el estado 1", "nombre del color para el estado 2", "nombre del color para el estado 3")
```

Los nombres de los colores pueden ser consultados a Biecek (2014: 71).

- Cuando todos estos elementos hayan sido creados ya se puede **crear el OSEQ** mediante la siguiente sintaxis:

```
Nombre_OSEQ  <- seqdef(OR,var=2:250, cpal=OR.cpal, alphabet=OR.alphabet, labels=OR.labels)
```

Donde:

. Nombre_OSEQ: es el nombre que nosotros asignamos al OSEQ que acabamos de crear.
. Seqdef: nombre de la función de TraMineR mediante la cual se genera el OSEQ.
. OR: Objeto de R.
. Var=2:250: número de fila y columna dónde empiezan los datos referentes a las trayectorias. Es probable que nuestra matriz de datos incluya tanto variables referentes a la trayectoria como otras que no, como por ejemplo las variables sociodemográficas típicas como podrían ser la edad, sexo, nivel de estudios, etc. Mediante este comando se acota dónde están los datos para crear el OSEQ.
. Cpal=OR.cepal: queda definida la paleta de colores.
. Alphabet=OR.alphabet: queda definido el alfabeto.
. Labels=OR.labels: quedan definidas las etiquetas.

Una vez realizada esta sintaxis ya tendremos listo el OSEQ para poder ejecutar las funciones incluidas en el programa TraMineR.

Principales funciones de TraMineR

En el programa TraMineR hay infinidad de funciones para el análisis de trayectorias (Ritschard, 2018). Sin embargo, por motivos de extensión y practicidad en este apartado nos centraremos en las tres siguientes:

- Gráficos de secuencias.
- Tiempo medio en cada estado.
- Número de transiciones.

Gráficos de secuencias

TraMineR incluye funciones para representar gráficamente las trayectorias de manera interesante y novedosa. Los gráficos tanto pueden ser individuales, dónde cada fila es una observación (individuos en nuestro ejemplo)

como transversales, que explicaremos a continuación.

Para ejecutar un gráfico individual hay que seguir esta sencilla sintaxis:

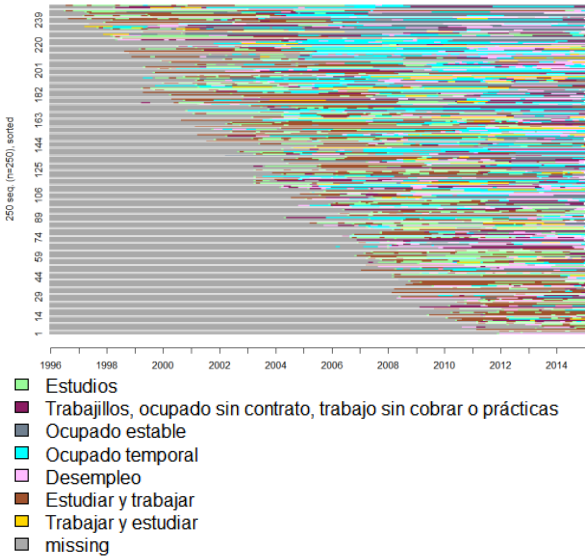
```
Seqiplot  (nombre_OSEQ)
```

Donde "seqiplot" es el nombre de la función.

En el gráfico 1 podéis ver un ejemplo con los datos de nuestro proyecto. En este caso se ha ordenado los casos por la edad añadiendo en la sintaxis:

```
(sortv=OR$variable_orden)
```

Gráfico 1. Secuencias individuales de las trayectorias laborales ordenadas por edad. N 250.



Fuente: elaboración propia a partir de datos de REDEMAS.

Además de los gráficos individuales también se pueden realizar gráficos transversales. Se trata de un gráfico de distribución transversal en que se ve que tipo de estados o actividades tienen más peso dentro de la muestra en un momento determinado. El gráfico 2 es un ejemplo de ello. En este caso, en lugar de ordenar las trayectorias cronológicamente, empiezan todas cuando la totalidad de los jóvenes tenía 16 años, independientemente de su edad actual.

Para elaborar este grafico la sintaxis que hay que ejecutar es la siguiente:

```
Seqdplot (NO_OSEQ)
```

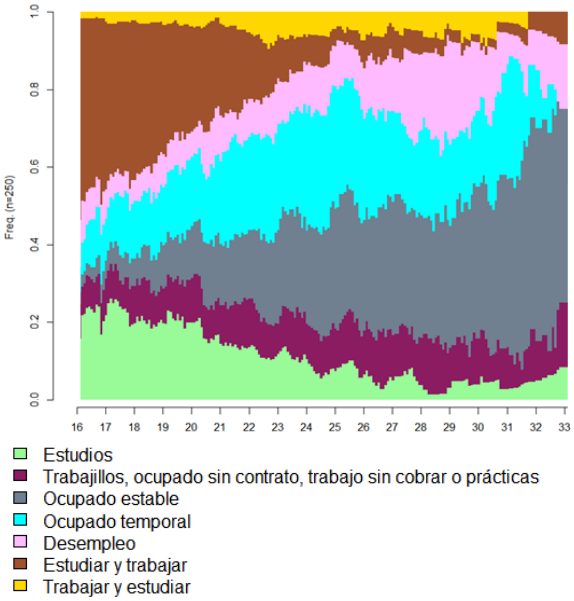
Tiempo medio en cada estado

TraMineR incluye funciones para calcular el tiempo dedicado a cada tipo de evento. Esta función es especialmente útil para caracterizar cómo son las trayectorias y en qué tipo de actividades se pasa más tiempo en función de las variables sociodemográficas incluidas en el análisis.

La sintaxis es la siguiente:

```
Seqistatd (nombre_OSEQ)
```

Gráfico 2. Secuencias transversales de las trayectorias laborales. N 250.



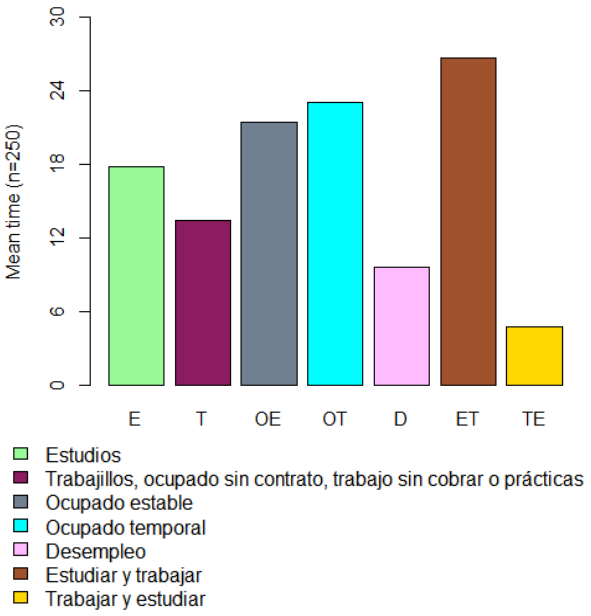
Fuente: elaboración propia a partir de datos de REDEMAS.

Al ejecutar esta sintaxis se calcula para cada individuo el tiempo total para cada estado. Esta información puede ser exportada como variable para poder realizar cálculos y análisis posteriores.

También se puede representar gráficamente esta información como vemos en el gráfico 3:

```
Seqmtplot (nombre_OSEQ)
```

Gráfico 3. Tiempo medio en cada estado para el conjunto de la muestra. N 250.



Fuente: elaboración propia a partir de datos de REDEMAS.

Finalmente, también se puede calcular la media del tiempo dedicada a cada tipo de evento para el conjunto de la muestra con la siguiente función:

```
Seqmeant (nombre_OSEQ)
```

O en función de variables que tengamos en la matriz con el siguiente comando:

```
By (nombre_OSEQ,
OR$variable_segmentación, seqmeant)
```

Este es el resultado para toda la muestra utilizando nuestros datos como ejemplo (en este caso el tiempo esta expresado en meses):

```
   Mean
E  17.8
T  13.4
OE 21.4
OT 23.1
D   9.6
ET 26.7
TE  4.8
```

Número de transiciones

El programa TraMineR contiene también algunas funciones para calcular índices que tienen que ver con la variabilidad de estados dentro de una secuencia, es decir cuanta complejidad interna hay en cada trayectoria. Una manera de aproximarse a esta dimensión es mediante la observación del número de transiciones, es decir, cuantas veces se pasa de un estado a otro. Esta operación es muy útil para analizar situaciones de estancamiento o de estabilidad o inestabilidad en las trayectorias. Es la base para generar la matriz de transiciones que sirve para calcular la probabilidad de transitar de un estado a otro.

Para calcular el número de transiciones para toda la muestra se ejecuta la siguiente sintaxis:

```
Seqtransn (nombre_OSEQ)
```

Igual que con el tiempo medio en cada estado, esta información puede ser añadida como una variable en la matriz.

También se puede calcular segmentado la muestra:

```
By (nombre_OSEQ,
OR$variable_segmentación, seqtransn)
```

Finalmente, también se puede calcular el número medio de transiciones, ya sea para toda la muestra o segmentado. Primero hay que crear un objeto de R:

```
Nombre_Transiciones <- seqtransn
(nombre_OSEQ)
```

Y luego ejecutar la siguiente sintaxis:

```
Objeto_Transiciones <- aggregate
(Nombre_Transiciones, by=list
(OR$variable_segmentación), FUN= mean)
```

Para visualizar las medias de las transiciones se ejecuta el objeto de transiciones que acabamos de crear:

```
Objeto_Transiciones
```

Este es un ejemplo de la media de transiciones para cada grupo de edad de nuestra muestra:

```
Group.1   Trans.
1   20-24 4.508772
2   25-29 7.635294
3   30-35 9.490741
```

3. La técnica del Optimal Matching Analysis (OMA) para la creación de tipologías.

El análisis de alineación óptima o Optimal Matching Analysis, en su nombre original, fue desarrollado para analizar secuencias de ADN y cadenas de proteínas durante los años 70 y 80, en disciplinas muy alejadas de las ciencias sociales como la biología y más adelante relacionado con la informática (Abbott y Tsay, 2000).

Hay distintos métodos de análisis de datos secuenciales y, aunque no son los únicos, y surgirán nuevos en el futuro, distinguimos dos por su relevancia actual. Por una parte, *Event Structure Anlysis* desarrollado por Heise (1991) y por el otro, el que nos ocupa ahora: el *Optimal Matching Analysis (OMA)* introducido en las ciencias sociales por Andrew Abbott y John Forrest en 1986 (Abbott y Forrest, 1986).

El éxito del análisis de secuencias en las ciencias sociales es ampliamente atribuido a Abbott y Forest quienes en 1986 fueron las primeras personas a utilizar este tipo de metodología aplicada al estudio de las ciencias sociales (concretamente analizaron como han cambiado las formas de baile rituales, aunque en este caso el tema sirve de simple excusa para poder aplicar la metodología de manera clara y entendedora). Abbott y Forrest (1986: 2) se dan cuenta del vacío que hay en las ciencias sociales para analizar cuantitativamente patrones similares: "The only practical approach to these tasks has been to generate a common pattern using an 'ideal type' or comparative analysis, and then to consider the variations from it on an individual basis. There have been no effective quantitative methods for analyzing such 'sequence data'". Consecuentemente, el objetivo de los autores era utilizar metodología cuantitativa para analizar patrones, cosa que en ciencias sociales solo se había estudiado de manera cualitativa.

La particularidad del OMA es que compara secuencias enteras y no compara los eventos entre sí (Abbott y Forrest, 1986). Para que sea más fácil de explicar, lo trasladamos a nuestro objeto de estudio: las trayectorias laborales de los jóvenes. Se considera cada trayectoria como una secuencia que se analiza como una sola unidad de análisis y no se analizan los distintos eventos (empleo en X empresa, empleo en Y empresa, desempleo, etc.) por separado.

La técnica del Optimal Matching compara las secuencias de una muestra y calcula la distancia entre ellas, es decir *cuánto* se asemejan y *cuánto* se diferencian. En palabras de Gilbert Ritschard, (2012: 4): "OM analysis consist in computing pairwise dissimilarities between sequences by means of an edit distance and then running a clustering analysis from the obtained dissimilarities".

Para realizar este cálculo la lógica que se utiliza es cuantas modificaciones hay que hacerle a una secuencia (A) para que sea exactamente igual a otra (B).

Las modificaciones pueden ser de dos tipos: eliminación y creación de elementos (conocido como operaciones *indel*) o substitución de elementos (Abbott y Forrest, 1986; Abbott y Tsay, 2000; Robette, 2010). Como hay muchas maneras de conseguir que una secuencia se transforme en otra, el criterio escogido es el que suponga el mínimo de cambios. Cada cambio tiene un "coste" y se trata de minimizar ese coste. La última cosa a tener en cuenta es que no todas las operaciones tienen asociado el mismo "coste".

Hay muchas formas de establecer el sistema de costes para las operaciones *indel* y de substitución, pero hay algunas más extendidas. Algunos investigadores usan criterios teóricos basados en la matriz de probabilidades de transitar (Robette, 2010), ya que a veces la distancia entre los estados no es la misma (ver más en Solís y Billari, 2003), pero los costes fijos, criterio que vamos a utilizar, también es muy habitual (Abbott y Tsay, 2000; Studer et al., 2010). Esta segunda opción consiste en calcular la disimilitud teniendo en cuenta que todas las operaciones *indel* cuestan 1 y que las operaciones de substitución cuestan 2. Las operaciones de tipo *indel* tienen que tener costes menores tal y como apuntó Gauvreau (1994, en Abbott y Tsay, 2000) ya que, en caso contrario, en secuencias de igual longitud nunca saldría a cuenta utilizar este tipo de operaciones.

Vemos ahora un ejemplo. Siguiendo a Studer et al. (2010: 11), si disponemos de estas dos secuencias:

1	SC	SC	SC	EM	EM	EM	JL
2	SC	SC	SC	EM	EM	JL	JL

Hay dos maneras de transformar las secuencias hasta que sean iguales. La primera es añadiendo un nuevo acontecimiento a la secuencia 2 y eliminar otro. Si utilizamos un cálculo de costes constante, el coste total de la operación es 2, 1 por añadir un evento y otro por eliminarlo.

1	SC	SC	SC	EM	EM	EM	JL
2	SC	SC	SC	EM	EM	EM	JL	~~JL~~

La segunda es substituir el acontecimiento "JL" por el de "EM". Dado que la operación de substitución cuesta 2, cualquiera de las dos maneras nos lleva al mismo resultado. En este caso pues, la "distancia" entre ambas secuencias es 2. Si por el contrario una manera tuviera un coste menor que otra, siempre se escogería la que lleva asociado el coste menor.

1	SC	SC	SC	EM	EM	EM	JL
2	SC	SC	SC	EM	EM	EM	JL

Una vez está hecho este cálculo, el resultado es una matriz de costes dónde cada secuencia lleva asociado un coste para transformarse en otra. A partir de estos costes mínimos se genera una matriz de distancias, resultado del grado de similitud entre las secuencias.

De esta manera, se pueden generar grupos de secuencias similares. Aquellas que compartan más elementos en común, y por lo tanto, sea menos costoso transformarse en otras, se agruparan. Las agrupaciones se hacen siguiendo los métodos típicos de clúster en el que el resultado final minimiza la distancia intra-grupal al tiempo que maxifica la distancia enter-grupal.

3.1. Sintaxis para la creación de tipologías con Optimal Matching

Para crear la tipología mediante la técnica del Optimal Matching lo primero que hay que hacer es generar la matriz de distancias entre las secuencias que será utilizada para agrupar aquellas secuencias que sean más parecidas para crear los clústeres.

Esta es la sintaxis para crear la matriz de distancias:

```
Nombre_objeto_matriz de distancias <-
seqdist (nombre_OSEQ, method = "OM", indel
= 1, sm = "CONSTANT », with.missing = TRUE)
```

Como veis el método es OM (Optimal Matching) y en este caso el cálculo de los costes está hecha mediante el criterio de costes fijos: sm = "CONSTANT". Si se quiere utilizar la matriz de la probabilidad de transitar entre un estado se tiene que especificar de la siguiente manera: sm = "TRATE".

Para visualizar la matriz de distancias se ejecuta el objeto que acabamos de crear:

```
Nombre_objeto_matriz de distancias
```

```
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
 [1,]    0   77  105  208   67  216   82  197  135   165
 [2,]   77    0  152  167   44  145   75  162   82   124
 [3,]  105  152    0  187  136  211  175  286  222   146
 [4,]  208  167  187    0  157  252  206  257  217   155
 [5,]   67   44  136  157    0  171   63  166  104   112
 [6,]  216  145  211  252  171    0  216  203  191   173
 [7,]   82   75  175  206   63  216    0  181   63   163
 [8,]  197  162  286  257  166  203  181    0  162   210
 [9,]  135   82  222  217  104  191   63  162    0   190
[10,]  165  124  146  155  112  173  163  210  190     0
```

El siguiente paso es generar la tipología a partir de esta matriz de distancias. Utilizaremos el método de clústeres jerárquicos. En primer lugar hay que descargarse un paquete (igual que hemos hecho con el TraMineR) para poder ejecutar las siguientes funciones:

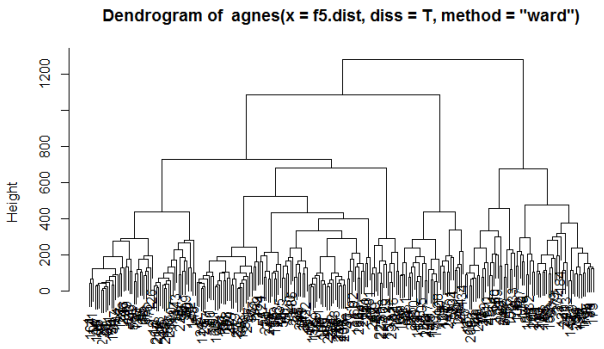
```
Install.packages ("Cluster")
Library ("Cluster")
```

A continuación, generamos los clústeres utilizando la matriz de distancias recientemente creada:

```
Nombre_clusterward <- agnes
(Nombre_objeto_matriz de distancias, diss
= T, method = "ward")
```

Podemos visualizar el dendrograma siguiendo esta sintaxis:

```
Plot (Nombre_clusterward, ask = F,
which.plots = 2)
```



El último paso es definir el número de clústeres mediante la siguiente sintaxis:

```
Nombre_cluster <- cutree
(nombre_clusterward, k = 4)
```

En este caso se han creado 4 clústeres. Se pueden guardar los clústeres como un objeto de R y se puede utilizar como si fuera una variable más. Por ejemplo para segmentar los gráficos o los distintos cálculos que hemos presentado como el número medio de transiciones por ejemplo.

Primero ponemos nombre a cada clúster, tantos nombres como clústeres hayamos definido:

```
cluster.labels <- c("nombre clúster 1",
"nombre clúster 2","nombre clúster
3","nombre clúster 4")
```

Luego guardamos el clúster como un objeto de R:

```
Objeto _cluster <- factor(Nombre_cluster,
levels = 1:4, labels = cluster.labels)
```

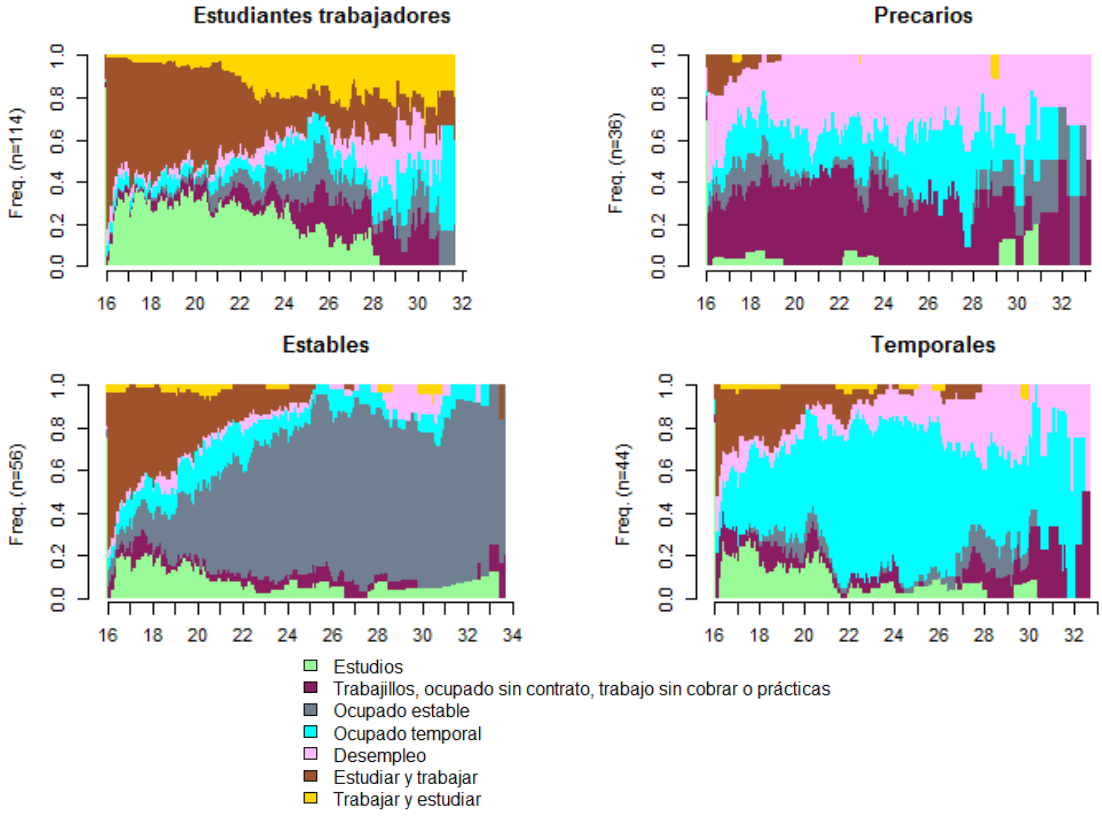
Donde **levels =** 1:4 se refiere al número de clústeres. Si por ejemplo hubiesen dos grupos en lugar de cuatro aquí deberíamos poner **levels =** 1:2.

Ahora ya tenemos creado el clúster como un objeto y podemos utilizarlo como una variable más. Por ejemplo, ejecutando la siguiente sintaxis visualizaremos los gráficos de secuencias para cada clúster:

```
Seqdplot (nombre_OSEQ, group =
objeto_cluster)
```

El gráfico 4 es un ejemplo de la visualización de los clústeres que acabamos de crear con gráficos de distribución transversal.

Gráfico 4. Secuencias transversales para cada clúster. N 250.



Fuente: elaboración propia a partir de datos de REDEMAS.

4. Consideraciones finales

Como hemos visto el paquete TraMineR para R ofrece una serie de funciones muy útiles para la operativización cuantitativa de trayectorias. Así, además de poder representar gráficamente las secuencias, tanto de manera individual como transversal, se pueden realizar cálculos muy útiles como el tiempo dedicado a cada estado o el número de veces que se transita de un estado a otro por poner algunos ejemplos. Estas funciones son sencillas de ejecutar siempre y cuando se disponga de los datos en el formato correcto.

Otra de las principales ventajas de usar el análisis de secuencias y TraMineR es la posibilidad de elaborar tipologías basadas en el cálculo de distancias mediante la técnica del Optimal Matching (alineación óptima). Mediante esta técnica se agrupan las secuencias según su similitud considerando cada trayectoria como una sola unidad de análisis. Asimismo, como hemos visto la sintaxis es relativamente sencilla de ejecutar.

Aunque este *paper* tiene un carácter introductorio, creemos que puede ser útil para aquellos investigadores que estén pensando en adentrarse en el análisis cuantitativo con datos longitudinales.

5. Bibliografía

Abbott, A. y Forrest, J. (1986) "Optimal Matching Methods for Historical Sequences" The Journal of Interdisciplinary History, Vol. 16, No. 3 (Winter, 1986), pp. 471-494.

Abbott, A. (1990) "Conceptions of time and events in social science methods", *Historical Methods*; Vol. 23 Issue 4, p140, 11p, 1 Chart.

Abbott, A. y Tsay, A. (2000) "Secuence analysis and Optimal Matching Methods in Sociology", *Sociological Methods and Research*, vol 29, nº1. Pp3-33.

Belvis, F. X. y Benach, J. (2013) *Guia introductòria a l'anàlisi longitudinal de dades de panel. Exemples pràctics a partir del Panel de Desigualtats Socials a* Catalunya-PaD. Fundació Jaume Bofill.

Biecek, P. (2014) "Przewodnik po pakiecie R", Oficyna Wydawnicza "GIS". http://www.biecek.pl/R/R.pdf

Billari, F. (2001) "Sequence analysis in demographic research". *Canadian studies in population*, 28(2), 439-458.

Casal, J., Merino, R. y García, M. (2010). "Pasado y futuro del estudio sobre la transición de los jóvenes". *Papers: revista de sociologia*, 96/4, 1139-1162.

Castelló, L., Bolíbar, M., Barranco, O. y Verd, J. M. (2013). "Treball: Condicions en el mercat de treball i trajectòries laborals de la joventut catalana". En: Serracant, Pau (coord.). *Enquesta de la Joventut de Catalunya 2012. Volum 1. Transicions juvenils i condicions materials d'existència.* Barcelona: Generalitat de Catalunya. Direcció General de Joventut, 117-224.

Gabadinho, A., Ritschard, G., Müller, N. S., y Studer, M. (2011). Analyzing and visualizing state sequences in R with TraMineR. *Journal of Statistical Software*, *40*(4), 1-37.

Heise D. (1991). "Event structure analysis". *Using Computers in Qualitative Research*, ed. Fielding, R. Lee, pp. 136-63. Newbury Park CA: Sage.

Mayer, K.U. (2001). "The paradox of global social change and national path dependencies: life course patterns in advanced societies", en A.E. Woodwa rd y M. Kohli (eds.): *Inclusion and exclusion in European societies*, Londres: Routledge, 89-110.

Ritschard, G. (2012) "Exploring secuencial data" en Jean-Gabriel Ganascia Philippe Lenca Jean-Marc Petit (Eds.) *Discovery Science 15th International Conference*. Springer.

Ritschard, G. (2018) *Package 'TraMineR'.* https://cran.r-project.org/web/packages/TraMineR/TraMineR.pdf

Robette, N. (2010) "The diversity of pathways to adulthood in France: Evidence from a holistic approach" *Advances in Life Course Research,* 15. Pp 89–96.

Serracant, P. (2014) *Canvis i continuïtats en les trajectòries de transició de la joventut catalana.* Tesis doctoral. http://ddd.uab.cat/pub/tesis/2014/hdl_10803_284954/psm1de1.pdf

Solís, P. y Billiari F. (2003) "Vidas laborales entre la continuidad y el cambio social: trayectorias ocupacionales masculinas en Monterrey, México". *Estudios Demográficos y Urbanos* 3-18, pp. 559-595.

Studer, M., Gabadinho, A., Ritschard, G., Müller, N., (2010) "Sequence analysis for social scientists Part IV - Analyzing sequences using dissimilarities" Summer School on Advanced Methods for the Analysis of Complex Event History Data, Bristol, 28-29 June 2010.

Verd, J. M., López-Andreu, M., (2011). "The Rewards of a Qualitative Approach to Life-Course Research. The Example of the Effects of Social Protection Policies on Career Paths", *Forum Qualitative Sozialforschung /Forum: Qualitative Social Research*, *12*(3), Art. 15.

Widmer, E. y Ritschard, G. (2009) "The de-standardization of the life course: Are men and women equal?" *Advances in Life Course Research*, 14, 28-39.