Sequence Analysis

Liam Wright

25 June, 2019

Introduction

- The DWP wants to move people from no-pay, low-pay or part-time work into high paid, full time work.
- It also wants this change to be permanent.
- As analysts, how do we find the factors which enable this?
- Typically, we look for the factors which predict an event - here, 'exit' into high paid work - at a single time point, either in a discrete time (probability) or continuous time (hazard rate) set up.

Introduction

- But, this type of analysis throws away a lot of information.
- Importantly, it lacks **memory** - it ignores long term processes made up of multiple transitions.
- Another approach might be to use the full trajectory and categorise these into types using logical rules.
- The Resolution Foundation (2014, 2017) do this by following individuals in low paid work in a baseline year over the next decade and categorise them into groups such as 'escapers' (high paid in final three years) and those that remain 'stuck'.

Introduction

- But, the categories are somewhat arbitrary (why not high paid in final four years?) and choosing rules is difficult when there are several time points and categories.
    - With 3 categories and 10 time points, there are $3^{10} = 59,049$ possible sequences.
- Sequence Analysis is a data-driven approach to categorises trajectories ("sequences") based on their **similarity** to one another.
- It uses information on the **duration**, **timing**, and **order** of spells within a sequence, which the other approaches do not do or do as well.

What is a Sequence?

- In sequence analysis, a sequence is a series of mutually exclusive states ordered in discrete time.
- For instance, a set of labour market states in the 12 months after leaving education.
- Let's imagine there are three possible states: low-pay (L), no-pay (N), high-pay (H).

```
LLLLLLLLLLLL    Low Paid months 1-12
HHHHHHHHHHHH    High Paid months 1-12
LLLLNNLLLLLL    Low Paid except No Pay months 5-6
```

Sequences as Processes

- Sequence Analysis is interested in **processes**, so the sequence should have a common meaning across individuals (Studer, personal communication).
- The start of the process should have meaning across individuals
  - Labour market trajectories after leaving full-time education.
  - Experience following redundancy.
  - Common age.
- The whole process should be captured
  - Identical duration (e.g. 24 months)
  - Attainment of a particular outcome (e.g. acquiring full-time work)

Similarity between Sequences

- We would like to say the 1st and 3rd sequences below are more similar than the 2nd is to the 1st or 3rd.
- But, how can we express this numerically?

```
LLLLLLLLLLLL    Low Paid months 1-12
HHHHHHHHHHHH    High Paid months 1-12
LLLLNNLLLLLL    Low Paid except No Pay months 5-6
```

Similarity between Sequences

- Luckily, there are many algorithms available to do this.
- These prioritise the timing, duration and order of spells within sequences to varying extents (see Studer and Ritschard (2016) for a review).
- Here I will talk about the most frequently used algorithm, **Optimal Matching** (OM).
  - Optimal Matching is sometimes used interchangeably with Sequence Analysis.
  - The standard OM algorithm requires sequences of the same length. This is not true for other algorithms (see Studer and Ritschard, 2016).

Optimal Matching

- In Optimal Matching, similarity is measured as the minimum number of **operations** required to turn one sequence into another.
- Sequences requiring fewer operations are more similar to one another.
- The operations are:
  - Insertions
  - Deletions
  - Substitutions
- Insertions and Deletions are sometimes collectively termed **indels**.

```
LLLLLLLLLLLL to HHHHHHHHHHHH requires 12 substitutions.
LLLLLLLLLLLL to LLLLNNLLLLLL requires 2 substitutions.
```

Optimal Matching

- We need to go further than this though as some operations may not be of equal value.

`LLLLNNLLLLLL to LLLLLLLLLLLL requires 2 substitutions.`
`LLLLNNLLLLLL to LLLLHHLLLLLL requires 2 substitutions.`
`LLLLHHLLLLLL to LLLLLLLLLLLL requires 2 substitutions.`

- But, we wouldn't want to say these operations (`N -> L`, `N -> H`, `H -> L`) are identical.

Optimal Matching

- To get around this, we assign **costs** to the operations.
- For instance, the costs for substituting `N -> H` could be set higher than `L -> H` which could be set higher than `N -> L`.
- The costs could also depend on the point in the sequence the operation takes place at.
    - Some transitions may be more likely at the beginning of a sequence than later (e.g. exit from unemployment early in one's career).
- Similarity is then the number of operations multiplied by their cost.

Choosing Costs

- There are three ways of choosing costs (Studer and Ritschard, 2016):
- Theory
  - Defined by analyst. For example, set higher cost for move between inactive to unemployment than unemployed to high paid.
- State Attributes
  - Distance between characteristics of state/those in state. For example, use average income, age, qualifications of people in each state.
- Data-Driven
  - Use information in the sequences themselves. For instance, base costs on the likelihood of a particular transition between state a and b.
  - Alternatively, use the χ^2 difference in the probability distribution of possible states following state a or b, t periods in the future.

Choosing Costs

- Costs based on the likelihood of transition are most often used, to my knowledge.
- However, transitions between certain states may be impossible/unlikely though the states themselves are qualitatively similar (e.g. single and divorced).
- Note, the indel and substitution costs need to be compatible.
  - If the substitution cost is >2x indel cost, a deletion and insertion is *cheaper* than a substitution.
- Triangular Inequality also needs to hold.
  - Two sequences must be at least as directly similar as they are going *via* a third sequence.
  - In other words, shortest distance between two sequences should be a straight line.

Creating Typologies

- The Optimal Matching algorithm produces an `n x n` matrix containing the dissimilarity between each combination of the `n` sequences in the dataset.
- To create typologies from these, we find clusters of sequences from this matrix using **cluster analysis**. There are many types of clustering algorithm available.
- The overall object is to minimise the distance between points within a cluster.
- The number of clusters, `c`, can be between 1 and `n`, with choice of `c` based on set of "quality measures" (see Studer (2013) for detail on these).

Creating Typologies

- Increasing the number of clusters reduces the average sample size in each cluster, but it is possible to group qualitatively similar clusters together.
- The clusters (or larger groups) can then be used as a categorical variable in analysis (i.e. as an `x` or `y` variable).

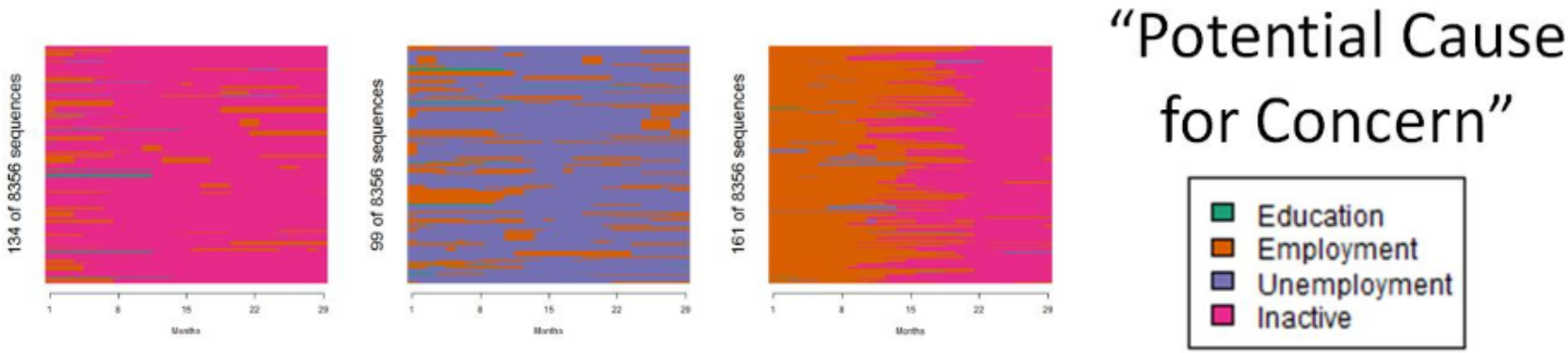

Figure 1: Anders and Dorsett (2017, p.87)

Example

- Anders and Dorsett (2017) are interested in how the transition to adulthood has changed across time.
- They study labour market sequences from age 16-19 in four cohort studies:
  - NCDS (1958 Birth Cohort Study)
  - BCS70 (1970 Birth Cohort Study)
  - YCS8 (Youth Cohort Study 8, born 1979/80)
  - LSYPE (born 1989/90)
- They use optimal matching in each cohort separately using time-dependent transition probabilities as substitution costs.
- The number of clusters in each cohort was chosen by qualitative assessment subject to average silhouette distance being above 0.7.

Example

- After visualising the clusters, they grouped similar clusters into three overarching groups:
  - Entering the Labour Market
  - Accumulating Human Capital
  - Potentially Difficult Transition
- They then carried out a series of analyses:
  - Checking change in proportions in each group across cohorts.
  - Testing whether associations between individual characteristics and group membership had changed across cohorts.
  - Decomposing change in proportions into changes in background characteristics and differences in associations.
  - Assessing extent to which age 16-19 group is related to age 20-24 trajectories.

Example

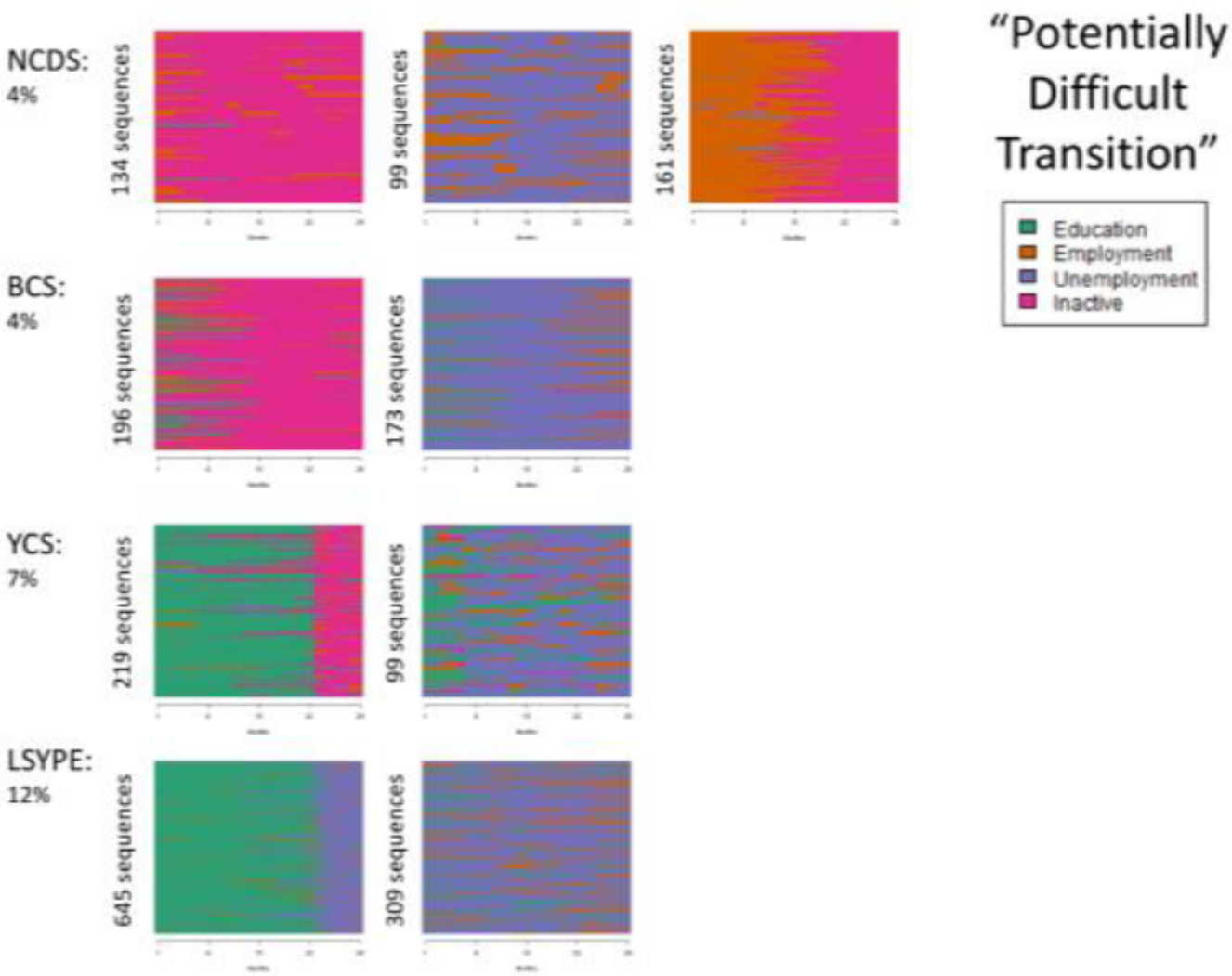

Figure 2: Anders and Dorsett (2017, p.87)

Extensions

- The OM algorithm relies on complete sequences of the same length. Halpin (2012, 2013) introduces an algorithm for imputing missing sequence data.
- As the clustering algorithm depends on an `n x n` matrix of similarities, it can get slow and memory intensive where `n` is large.
  - The `WeightedCluster R` package (Studer, 2013) can be used to aggregate identical sequences together and add weights into the clustering algorithm.

Extensions

- In the form I've described, Sequence Analysis does not allow study of how time-varying factors can change how a sequences *unfolds*.
  - Studer et al. (2018) introduce Sequence Analysis Multi-State Modelling (SAMM), a method which combines sequence analysis with event history analysis.
  - SAMM takes subsequences from the overall sequence and using clustered subsequences as the events to predict in a typical discrete- or continuous-time event model.
- Another extension is to assess sequences across multiple domains simultaneously (e.g. family, housing and labour market trajectories.) Pollock (2007) introduces "multi-channel" sequence analysis for this.

Measuring Sequence "Quality"

- So far, we've discussed how to put similar clusters together. Alternatively, we might be interested in getting a measure of sequence "quality" directly.
- Several measures of sequence quality exist (see Ritschard et al., 2018, for a review).
- Gabadinho et al. (2015) create a complexity index which captures the volatility of a sequence through terms for number of transitions and entropy (time spent in each state).
- Ritschard et al. (2018) build on this by adding costs to the states and transitions. Some transitions are positive (e.g. `L -> H`), others are negative (e.g. `H -> L`).
- Their precarity index is highly associated (in significance and effect size) with future labour market outcomes in a sample of Northern Irish school leavers.

Software

- In `R`, the `TraMineR` package is available (Gabadinho et al., 2015).
  - The package has a great accompanying website and set of guides, face-to-face training sessions are often run on it (link to one **here**), and the creators respond to queries.
  - `TraMineR` also has wonderful data viz capabilities.
- In `Stata`, the `SADI` (Halpin, 2017) and `SQ` (Brzinsky-Fay et al., 2006) packages are available.
- I do not know of any package in `SAS`.

DWP Applications

- What are the long term effects of precarious employment histories in the UK?
- Is there no-pay/low-pay cycling in the UK?
- If so, what are the characteristics and histories of the low-pay/no-pay cyclers?
- What predicts exit from low-pay/no-pay cycle?
- What are the characteristics of those who make sustained moves out of Universal Credit?
- To what extent and why do individuals move between UC streams?
- Are there any groups who are unlikely to make the move from part-time to full-time work?

References

- Anders, J., & Dorsett, R. (2017). What young English people do once they reach school-leaving age: A cross-cohort comparison for the last 30 years. Longitudinal and Life Course Studies, 8(1), 75-103. https://doi.org/10.14301/llcs.v8i1.399
- Brzinsky-Fay, C., Kohler, U., & Luniak, M. (2006). Sequence Analysis with Stata. Stata Journal, 6(4), 435-460. https://doi.org/10.1177/1536867X0600600401
- Gabadinho, A., Ritschard, G., M?ller, N. S., & Studer, M. (2015). Analyzing and Visualizing State Sequences in R with TraMineR . Journal of Statistical Software, 40(4). https://doi.org/10.18637/jss.v040.i04

References

- Halpin, B. (2012). Multiple Imputation for Life-Course Sequence Data. Retrieved from http://teaching.sociology.ul.ie/seqanal/shortptex.pdf
- Halpin, B. (2013). Imputing Sequence Data: Extensions to initial and terminal gaps, Stata's mi Imputing Sequence Data: Extensions to initial and terminal gaps, Stata's mi *. Retrieved from http://www.ul.ie/sociology/pubs/wp2013-01.pdf
- Halpin, B. (2017). SADI: Sequence analysis tools for stata. Stata Journal, 17(3), 546-572. https://doi.org/10.1177/1536867X1701700302
- Pollock, G. (2007). Holistic trajectories: A study of combined employment, housing and family careers by using multiple-sequence analysis. Journal of the Royal Statistical Society. Series A: Statistics in Society, 170(1), 167-183. https://doi.org/10.1111/j.1467-985X.2006.00450.x

References

- Resolution Foundation. (2017). The Great Escape? Low pay and progression in the UK's labour market. Retrieved from https://www.resolutionfoundation.org/app/uploads/2017/10/Great-Escape-final-report.pdf
- Resolution Foundation. (2014). Escape Plan: Understanding who progresses from low pay and who gets stuck. Retrieved from https://www.resolutionfoundation.org/app/uploads/2014/11/Escape-Plan.pdf
- Ritschard, G., Bussi, M., & O'Reilly, J. (2018). An Index of Precarity for Measuring Early Employment Insecurity, 279-295. https://doi.org/10.1007/978-3-319-95420-2_16

References

- Studer, M. (2013). WeightedCluster Library Manual: A practical guide to creating typologies of trajectories in the social sciences with R. LIVES Working Papers, 1-34. https://doi.org/10.12682/lives.2296-1658.2013.24
- Studer, M., & Ritschard, G. (2016). What matters in differences between life trajectories: A comparative review of sequence dissimilarity measures. Journal of the Royal Statistical Society. Series A: Statistics in Society, 179(2), 481-511. https://doi.org/10.1111/rssa.12125
- Studer, M., Struffolino, E., & Fasang, A. E. (2018). Estimating the Relationship between Time-varying Covariates and Trajectories: The Sequence Analysis Multistate Model Procedure. Sociological Methodology, 48(1), 103-135. https://doi.org/10.1177/0081175017747122