University of Limerick
Department of Sociology Working Paper Series

Working Paper WP2013-05
June 2013

Brendan Halpin

Department of Sociology, University of Limerick

Three Narratives of Sequence Analysis

Available at `http://www.ul.ie/sociology/pubs/wp2013-05.pdf`

Three narratives of sequence analysis

Brendan Halpin

2013-06-11

Contents

1 Three narratives of sequence analysis

While the Optimal Matching algorithm (OM) has been very successful in sequence analysis in sociology, it has limitations, particularly when applied to lifecourse data. This paper addresses some of these problems, and considers a number of alternative ways of defining similarity between lifecourse trajectories. Similarity measures can be compared in terms of three related dimensions: one, their algorithmic structure, two, their practical results, and three, the extent to which they can yield sociologically interpretable

distances between sequences, their "narrative" of similarity. OM's narrative depends on defining similarity as edit-distance, and this can be attractive for sequences that are naturally discrete in time, but it does not map without problem onto sociologically meaningful intuitions of similarity, particularly when time is continuous.

The alternatives considered in this paper include two relatively minor modifications of OM, a group of methods that focus on enumerating common subsequences, and a time-warping method. Each of the alternative algorithms implicitly defines similarity in a different way, providing a different narrative and orienting us to different intuitions of similarity. The two modifications of OM attempt to address problems arising from OM's focus on context-free comparisons between tokens, but for technical reasons they fail to produce valid distance data. The subsequence-oriented approach defines similarity in terms of the extent to which two sequences go through the same states in the same order, which is radically different from the principles underlying OM, and empirically provides very different distances. Finally, the time-warping method provides distances defined in terms of local expansion and compression of the time axis, which yields a very different way of looking at similarity of longitudinal data. The implementation of the time-warping algorithm is structurally very similar to OM, though it produces distances that are in many cases intermediate between OM and the subsequence approach.

Thus, in terms of tools for comparing sequences, we have three distinct narratives: similarity in terms of edit distance (potentially with modifications), similarity in terms of going through the same states in the same order, and similarity in terms of warping the time axis. At one level measures are algorithms, with deterministic behavior set by their rules and parameterisation, but we can think of "narrative" as referring to our understanding, perhaps heuristic, of the relationship between the algorithms and interpretable, meaningful comparisons made between sequences.

1.1 Method and meaning

For sequence analysis to be useful to sociology, it has to make sociological sense. We can reduce this requirement in the current context to a demand that inter-sequence distances can be made to correspond systematically to sociological ideas of similarity, leaving aside the important but less distinctive issue of what use (clustering, comparison with ideal types, etc.) is made of the similarity data. This topic has not received a great deal of systematic treatment, though proponents of sequence-analytic measures will implicitly or explicitly assert that their measure is meaningful.

Sociological meaningfulness was one of the multiple foci of criticism of sequence analysis posed by Wu (2000) and Levine (2000) (in response to Abbott & Tsay, 2000), who still represent the most sustained intellectual attack on the approach even now, some 14 years later.[1] Other criticisms tend to focus on the advantages of one distance measure compared to others, while taking the utility of sequence analysis as such for granted (e.g., Lesnard, 2010; Hollister, 2009; Halpin, 2010).

Wu and Levine raised many concerns about the black-box nature of the process, and the inability of its proponents to satisfactorily elucidate the link between the operation of the OM algorithm and theoretically relevant differences between trajectories. They saw its utility in molecular biology as arising from a close relationship between its elementary operations and processes of change in DNA recombination, with this mapping being entirely invalid for social processes. However, OM's algorithm does not intentionally (or even particularly closely) model molecular biology processes; rather it is driven by computational tractability, and can be applied to token strings in a variety of contexts, including DNA and social applications. This view of the algorithm as modelling the data generation process lead them to compare it unfavourably to the alternative narrative commonly applied to longitudinal social data,

[1] A relatively rare example of an application that draws negative conclusions about the utility of OM is McVicar and Anyadike-Danes (2010).

hazard rate modelling or "event history analysis". EHA presents a very attractive mental (as well as statistical) model of lifecourse processes, with causality operating in continuous time, taking account of current state, history and duration dependence, and can indeed model the data generating process. Moreover, as a stastical model it can estimate the effects of other measured variables and test precise hypotheses about them. However, sequence analysis (which typically has other goals) is not a model of data generation, but rather (at the level of the distance measure) a way of defining pairwise similarity for (typically) descriptive and exploratory purposes. Wu in particular read the algorithm as competing with models like EHA, which lead him to unnecessarily strong objections to the logic of the elementary operations. In particular he misread the substitution operation as a kind of transition, and objected to "impossible" transitions (such as from married to never-married). While he may have been mistaken, parameterisation of OM worries lots of people – rightly, because parameterisation has a stark effect on the resulting distances. However, the problems are not insurmountable, particularly if substitution costs are thought of as simply describing differences between the categories of the state variable. Sequence analysis therefore consists of a mapping of the state-space distances onto the sequence domain, yielding a set of sequence-space distances. Viewed this way substitution costs are less intimidating, and in many simple ways the state-space patterns will influence the sequence-space patterns, though given the extra complexity inherent in sequences, this is not a deterministic relationship.

As we have seen, the competing narrative of event history analyis is very attractive, though it does a very different task to SA. Other statistical models that provide competing narratives include latent class models (e.g., Barban & Billari, 2012), which focus on the search for classes rather than modelling the data generation process, or latent growth curve models (e.g., Lovaglio & Mezzanzanica, 2013) which focus on the multiple-individual time-series nature of the data.

While one may discount some of Wu and Levine's objections, and note that sequence analysis can serve different functions from EHA, LCA, LGCM and other statistical models, it is well to note that their fundamental injunction holds true: for sequence analysis to be worth the candle we need to be able to justify the narrative it gives us about similarities between sequences through sociological state spaces.

1.2 The narrative of sequence analysis: mapping state-space onto sequence space

Sequence analysis is concerned with identifying similarity between sequences. We do this by reference to information about the state space through which the sequences move, at the minimum in terms of identity and difference of states (as in subsequence approaches), but usually with more detailed ideas of distances within the state space (as when we use the "substitution cost matrix" in OM). In a broad sense the narrative of sequence analysis is the mapping of information about the state space onto the sequence space. How the mapping occurs is obviously critical but varies, sometimes dramatically, from one algorithm to another.

The simplest mapping is the Hamming distance, where the inter-sequence distance is the sum of the state distances at each timepoint. While the correspondence between the state space and sequence space is thus utterly clear, Hamming is blind to similarity that is displaced in time. The other available algorithms allow time-dislocation in different ways, at the expense of a more complicated relationship between the state space and the sequence space. OM uses deletion and insertion of elements in the sequence to allow time-dislocation by "alignment", but otherwise replicates the Hamming distance's mapping between state and sequence (though in the context of OM this is described as "substitution"). Time-warping methods achieve the dislocation by warping the time axis, and then doing a Hamming-style comparison. Subsequence-based methods depart entirely from the Hamming-style comparison, and achieve a radical time-dislocation by focusing on the order of states rather than when they occur. Thus the available

algorithms will have different consequences and give different ways to think about the similarity.

OM's edit-distance narrative leads to differences after alignment being understood as substitutions (an edit operation). However, viewing the substitution cost matrix as a state-space distance matrix has advantages, stressing the commonality with other contexts where substitution is a less attractive metaphor, such as Hamming distance and time-warping. Viewing substitution costs as state-space distances also demystifies the derivation of substitution costs: state-space distances are just statements about differences between the categories of the state-space variable. These differences can be notional, based on external data or even derived from the sequences on the basis of transition rates. It might also avoid common misunderstandings about substitution, sometimes seen as directly modelling the empirical sequence-generating process, or leading to worries about the misapplication of molecular-biology models to sociology, and about "impossible transitions" etc.

The paper proceeds by looking at this set of measures, benchmarked against the simplest of them all, the Hamming distance, and compares how they perform with real lifecourse history data. All methods but Hamming allow some sort of time dislocation, be it by insertion and deletion, compression and expansion, or counting of subsequences wherever they appear. The measures' performance in a typical sequence-analysis workflow is compared: cluster analysis to generate a descriptive data-driven typology. All methods except the subsequence methods give results that are not very different from Hamming, suggesting that time-dislocation is important only a small part of the time; the subsequence method produces quite different results. The next section considers the shape of the sequence space in so far as multidimensional scaling can reveal it; this again reveals that the subsequence method is different, but also that for OM and the timewarping method there is little evidence of natural clustering among the trajectories. The following section bypasses the relative instability of cluster analysis by analysing correlations between measures: this shows again that the subsequence method is very different, and that the structure of distances in the state space is more important, much of the time, than the nature of the measure. In the final section, the time-warping measure is explored in more depth, showing that as its parameters are varied, it constitutes a bridge between OM-style measures and order-based measures like the subsequence algorithm.

2 Lifecourse data and token sequences

Optimal matching, or the Needleman-Wunsch algorithm (Needleman, Wunsch et al., 1970), and related approaches to defining distance or similarity between sequences, work with sequences that pass through a finite, discrete state space, such as strings of bytes representing characters (in the case of Levenshtein, 1966, bits or other tokens representing "words"). This maps well onto domains like fuzzy text search in computer science, or molecular biology where sequences are linear macro-molecules with repeating elements draw from a small set (e.g., DNA and the four CAGT bases). It also works for other naturally discrete state spaces such as a coding of utterances in conversations, steps in a dance, voting behaviour at successive elections, and so on. OM's elementary operations of insertion, deletion and substitution implicitly require sequences to consist of discrete, successive tokens, which can be operated on as independent atomic units. In contrast, life courses operate in continuous time (though in practice always measured with rounding, for example, in whole months). Two approaches are commonly used to represent continuous time/discrete space trajectories as token sequences: either whole spells in a given state are represented as single tokens, ignoring duration, or time units are represented as tokens so that spells are represented by runs of consecutive tokens, representing duration as repetition. The latter approach is used widely and, on the whole, it works well. However, if the distance measure implicitly treats tokens as strictly distinct from their neighbours, this is sub-optimal. Where the average spell length is longer than

one token, successive tokens are highly likely to have the same value, simply because no transition has occurred. In a true token string, while successive tokens will be correlated and often identical, each token is an independent realisation. For instance, if our sequence is vote at successive elections, party loyalty and patterns of flow between parties will mean that successive observations are related, but each vote is a distinct event. If however, the sequence is an annually-collected report of the most recent vote, it can only change when there has been an election: in this sequence for most $t/t+1$ transitions there is a very different sort of dependence. A more common example would be a trajectory through an occupational group state space: here most months will succeed the previous month without change of state, but only because no change of job has occurred, not that a new job in the same group has been found. This sort of dependence is not well picked up by OM and similar techniques, because we are dealing with spell data and they do not take account of that sort of structure.

That is, we often have reason to feel that the token-oriented view is inadequate for lifecourse data. Alternatives exist, but they have not been considered systematically together. In this paper I consider together a number of approaches that

- attempt to look at the context of tokens (the values of their neighbours, the length of the spells in which they are embedded)
- focus on spells, weighted by duration, rather than repeating tokens, or
- focus on the time dimension, defining similarity in terms of warping time.

I consider both the technical characteristics of the approaches, and the sort of sociological story we can use them to tell about similarity between life course sequences.

3 Modifying the algorithm: "localised" and "duration-adjusted" OM

One key problem with OM is that by defining distance in terms of token editing, it is blind to context. Only the values of the pair of tokens, each in one sequence, are taken into account, and their environment is invisible. Intuitively it is attractive that edits could mean different things according to context. In the lifecourse context, where transitions are relatively rare, and spells are typically rather longer than one unit, a deletion that alters a spell length is less important than one which removes a short spell entirely. Under OM, `AAAB` is as distant from `AACB` as from `AABB` (given $d(A,B) = d(A,C)$). More generally, it is reasonable that the importance of an edit can be affected by context, such that substituting `B` for `A` in `WAX` may not mean the same thing as in `YAZ`, while for OM the substitution cost is a function of A and B and nothing else.

This general problem motivated two independent approaches to modifying OM. While different in detail, both the modified algorithms proposed by Hollister (2009) and Halpin (2010) take context into account. Unfortunately, neither approach preserves the metric character of OM distances, which limits their utility considerably. I consider them here insofar as they throw light on the relationship between the algorithm and the meaning of the resulting distance, rather than as measures to be recommended for general use.

3.1 Hollister's LOM algorithm

Hollister's motivation is specifically the issue of context, that, in not taking account of a token's neighbours, OM gives edits that should be substantively different the same weight. If one inserts an element in

a string, that insertion should be less costly, the more the inserted element is similar to the two adjacent tokens. This will have the consequence that insertions which lengthen a spell (i.e., the two neighbours are identical to the inserted element) will be very cheap, but it is more general in that insertions between tokens similar but not identical to the inserted token also cost less. Thus, to insert element *k* between elements *i* and *j* the indel cost is:

$$\iota_{ijk} = \alpha\frac{d_{ik} + d_{jk}}{2} + \beta$$

where α and β are chosen by the analyst and d_{ij} is the distance between i and j in the state space (i.e., the ij substitution cost). To insert at an end of a sequence, $\iota = \alpha d_{ik} + \beta$. The β is a fixed cost for insertion, and α weights the distance, d, between the inserted element and each of its neighbours. (Note that while the OM algorithm is usually described in terms of insertions, deletions, and substitutions, we can think of deletions as insertions in the other sequence, and of substitutions as insertions (or deletions) in both sequences.)

3.2 Duration-adjusted OM

OMv, the duration-adjusted version of OM described in Halpin (2010), has a motivation that is superficially different from Hollister's LOM, but with strong underlying similarities. The key intuition is that operations on longer spells should cost less than operations on shorter spells. It is easier to see how this works if it is noted that insertions, deletions and substitutions can all be cast as deletions: an insertion of a missing element in s_1 is equivalent to deleting the extra element in s_2; a substitution is equivalent to a paired deletion of the corresponding elements in the two sequences. Thus all operations can be considered deletions, and deletions shorten the spell in which they occur, obliterating it if it has only one element. OMv operates by weighting such operations less, the longer the spell in which they operate. A weight of $1/\sqrt{l}$ is used as the default, where l is the spell length. Thus deleting an entire 1-unit spell would cost more than shortening, for example, a 5-unit spell by 1 unit. However, deleting a whole spell still costs more the longer it is. LOM will have a similar effect of favouring the cheapening operations within spells, though only by considering the adjacent elements, not the whole spell. OMv is more general than LOM in looking to the entire spell rather than the two adjacent elements, but LOM is more general than OMv in looking at the distance to the adjacent tokens, rather than just whether they are part of the same spell. While LOM has a narrative of attention to local context, OMv has a narrative of paying attention to lifecourses as sequences of spells.

Both measures are implemented as relatively simple modifications of the Needleman–Wunsch algorithm, but it can be shown that, unlike OM, the dissimilarities they produce do not have the metric property. In the next section I discuss what this means and why it matters, and then go on to demonstrate that the measures are not metric.

3.3 Why metricity matters

It is important that dissimilarities be metric, if we are to use them with conventional cluster-analytic or multi-dimensional scaling tools, which rely on the dissimilarities having a coherent global relationship, such that they can be used to construct a (perhaps latent) space within which the observations can be arrayed. Even more, it is very hard to think of a non-metric dissimilarity as a distance. The metric property distills critical characteristics of distance as we experience it in everyday Euclidean 3-dimensional space, and allows us to think of certain non-Euclidean measures as distances. A dissimilarity $\Delta(x, y)$ can be considered as metric if it has the following characteristics:

1. $\Delta(x, y) = 0 \Leftrightarrow x$ and y are the same

2. $\Delta(x, y) >= 0$
3. $\Delta(x, y) = \Delta(y, x)$
4. $\Delta(x, y) <= \Delta(x, z) + \Delta(z, y)$: the triangle inequality

In other words: distance is zero if and only if the entities are identical (or are in the same location); negative distances are impossible; distances are symmetric; and the distance from x to y cannot be greater than the distance from x to any other point plus the distance from that point to y – there are no short cuts! These conditions are naturally fulfilled by distances in Euclidean space, but they are also satisfied for a large body of measures in non-Euclidean space. At an intuitive level it is easy to see that dissimilarities that satisfy these conditions are like distances, and that those that do not may cause difficulties. For many dissimilarities the first three are easy to satisfy (though if the measures are travel times in a city with one-way streets, requirement 3 is not satisfied). The triangle inequality often causes difficulties.

It is inherent in the OM algorithm that the triangle inequality is satisfied, since it calculates the dissimilarity as the cheapest additive concatenation or summation of single-edit distances. First, note that as long as the state-space distances are metric, all one-operation distances in OM are metric. That is, the distance from ABA to ACA depends on the substitution cost or state-space distance, $d(B, C)$, and for there to exist a state D such that $\delta(ABA, ACA) > \delta(ABA, ADA) + \delta(ACA, ADA)$ would imply that the distance in the original state space from B to C is greater than the sum of $B \rightarrow D$ and $D \rightarrow C$; i.e., substitution gives metric distances between sequences unless the space described by the state-space distance matrix is non-metric. Since insertions and deletions have a single fixed cost, distances like $\delta(ABC, AC)$ will also be metric.

That OM's minimising concatenations will generate metric distances can be demonstrated by considering a route, $\delta'_1 2$, from s_1 to s_2, made by a sequence of elementary OM operations without minimising the cost. If we find a sequence s_3 such that there exist routes such that $\delta_{13} + \delta_{23} < \delta'_{12}$ we have shown that δ'_{12} is not the cheapest route, since the $s_1 \rightarrow s_3 \rightarrow s_2$ route is also valid and is cheaper. Thus OM necessarily produces metric routes, and if we find the *cheapest* route from s_1 to s_2, there can be no third sequence such that the triangle inequality is not satisfied.

In a general sense, we can consider non-metric dissimilarities to arise where the measures are not coherent enough to imply a space, a structured relationship independent of the observations, within which the observations can be located. Some measures are very good at identifying close matches, but where the dissimilarity is great, differences between the values of the measure are uninformative. Some measures may compare x and z using one set of shared characteristics, and z and y using another set, such that x and z are similar, and z and y are similar, but if x and y do not share characteristics they may be judged as very dissimilar. Elzinga (2006) has a valuable discussion on measures and metric properties.

3.3.1 Localised and duration-adjusted OM are not metric

It is easy to demonstrate that LOM is not metric. If we consider the following three sequences:

- s_1 = `BBBBAB`
- s_2 = `CCCACC`
- s_3 = `BBBACC`

Given the following cost structure:

- $\iota = 0.5\frac{d_{ik}+d_{jk}}{2} + 0.5$
- $d_{ij} = 1, i \neq j$

- $d_{ij} = 0, i = j$

the following dissimilarities are generated:

	LOM	OM	
Pair	$d = 1, \alpha = \beta = 0.5$	$\iota = 1.0$	$\iota = 0.75$
δ_{12}	6	6	5.5
δ_{13}	2.5	3	2.5
δ_{23}	3	3	3

The dissimilarity between s_1 and s_2 is greater than the sum of the dissimilarity between s_1 and s_3, and s_3 and s_2: d(1,3) + d(3,2) = 2.5 + 3 < d(1,2) = 6; the triangle inequality does not hold. The sequence s_3 becomes s_1 with a discounted insertion of a `B` between two `B`s plus two other operations, which reduces the dissimilarity between the level OM gives with the corresponding *indel* cost of 1.0. If we reduce OM's *indel* cost to reduce δ_{13}^{OM} to match, it also reduces δ_{12}^{OM}, preserving the triangle inequality. Thus under LOM, because there is a δ_{13}^{LOM} discounted edit that is not triggered in the δ_{12}^{LOM} or δ_{23}^{LOM} comparisons, the measure is not metric.

Like LOM, OMv also produces non-metric dissimilarities. Effectively, sequences with long spells will be judged more similar to all other sequences, because they attract discounted operations. Two sequences with many short spells may be quite dissimilar from each other but each may be judged quite similar to the long-spell sequence, violating the triangle inequality. For instance, the OMv distance between `BBBBAB` and `CCAAAC` is 3. However, going through `BBBBBB` the distance is $0.41 + 2.45 = 2.86$. The fact that `BBBBBB` consists of a single spell means that its distance even from a spell with no shared elements such as `CCAAAC` is reduced.

To summarise, from a narrative point of view, OM gives a definition of distance in terms of edits in token strings, which has great virtues of simplicity and efficiency, and takes into account the spatial structure of the initial state space in a clean way. However, by working with a model of time as naturally discrete, and of elementary operations as blind to their neighbourhood, the distances do not map unproblematically on to lifecourse history and other sociological data. The two attempts to adapt the algorithm to improve its narrative are sociologically attractive, but run into technical difficulties. [2]

4 Combinatorial measures: similarity by counting subsequences

Token-editing approaches define distance as the cost of the edits necessary to turn one sequence into another, but another tradition in sequence analysis has a radically different approach – and, as we will see below, distinctly different results.

In the 1990s Dijkstra and Taris (1995) (see also Dijkstra, 1994) proposed a way of looking at similarity of longitudinal data that focused on order. Leaving aside for the moment concerns with duration, two sequences are defined as more similar, the more they go through the same states in the same order. They proposed a way of calculating this similarity; by ignoring non-common elements and repeated elements their algorithm was tractable. However, as Abbott (1995) pointed out, ignoring non-common and repeated elements discards a lot of meaningful information, and for token sequences OM offers a more flexible and general measure. Nonetheless, there is something very attractive about this order-focused principle of similarity.

[2]The LOM and OMv measures are available as part of the SADI package of sequence analysis tools for Stata, at http://teaching.sociology.ul.ie/sadi/.

Using more complex algorithms, Elzinga proposed a more general approach which draws on this general concept of similarity (Elzinga, 2003, 2005, 2006). His method efficiently counts subsequences (order-preserving subsets of a sequence, not necessarily consecutive) and he proposed a variety of measures based on enumerating matching subsequences. If two sequences share a subsequence, they go through those states in the same order; Elzinga's method efficiently enumerates all such matching subsequences, and thus achieves in a more general way Dijkstra and Taris's goal.

All three of OM, the DT measure and Elzinga's measures are oriented to token-strings – all the algorithms treat the individual elements as atomic and capable of being considered out of context. Where duration is important, we can represent time by repeating tokens proportionally to spell length; as discussed above this has drawbacks, but further with Elzinga's method, long sequences are costly to process (since the number of subsequences is 2^l where l is the sequence length). However, he describes a number of variants of his measure which treat spells as tokens whose subsequence matches can be weighted by a function of the spell durations. This is a big improvement, in efficiency terms, over treating person–time-units as tokens: a five-year sequence will contain 60 person–months but perhaps only three or four spells. However, it must be noted that spells are no more atomic than person–months are, in the sense that their duration is stochastic.

Thus Elzinga's version of Dijkstra and Taris's principle of similarity provides another compelling narrative, which can be applied to lifecourses considered as sequences of spells with durations. In so far as questions remain about its operation, they have to do with the fact that existing implementations count matches as binary, and cannot deal with partial similarity (as other methods do via state-space distance matrices), and whether the algorithms of enumeration (which involve different types of double counting, of subsequences of subsequences), and duration-weighting of spells as tokens, make for interpretable distances in lifecourse contexts. With reference to the issue of partial matches, it is worth noting that Elzinga and colleagues are preparing methods that deal with "soft matching" (**elzinga_wang_studer_forthcoming**; Elzinga & Wang, 2012).

4.1 Implementation

Below, I present results using a duration-weighted, spell-oriented version of Elzinga's "number of matching subsequences" (NMS) similarity measure, which I refer to as the X/t measure. My implementation in Stata differs from the algorithm described in Elzinga (2006), in two main ways. First, rather than use the efficient algorithm he describes for enumerating common subsequences of a pair of sequences (which is called $\frac{N(N-1)}{2}$ times), I enumerate by brute force the subsequences of each sequence (called only N times) and then efficiently count the matches ($\frac{N(N-1)}{2}$ times). Second, Elzinga proposes a number of ways of taking duration into account, the most intuitively attractive of which is to weight by the sum of the spell-wise product of the durations of the subsequences. Thus, for a pair of subsequences, [A/5,B/4] and [A/1,B/2], Elzinga suggests the matching AB be weighted by $5 \times 1 + 4 \times 2 = 13$. However, as a consequence of the internal data structure, I cannot exactly replicate the duration weighting and instead weight by the product of the cumulated subsequence duration, $(5 + 4) \times (1 + 2) = 27$.

Table 1 shows three example sequences with their subsequences. All three have the same number of elements, and thus the same number of subsequences. However, since S_3 has a repeated element, it has a smaller number of *distinct* subsequences (the subsequence `B` appears twice, with a total duration of 15). The SXX measure calculated in the final row is the sum of the square of the cumulated duration in each distinct subsequence (so the `BB` subsequence in S_3 yields 15×15 rather than $9 \times 9 + 6 \times 6$).

The distance measure is defined as:

$$\delta^{X/t} = \sqrt{SXX + SYY - 2 \times SXY}$$

Table 1: Three example sequences, with their duration-weighted subsequences

S_1 = (A/10, B/4, C/6)	S_2 = (A/10, B/7, C/3)	S_3 = (B/9, A/5, B/6)
ABC / 20	ABC / 20	BAB / 20
AB / 14	AB / 17	BA / 14
AC / 16	AC / 13	BB / 15
BC / 10	BC / 10	AB / 11
A / 10	A / 10	B / 9 (+ 6)
B / 4	B / 7	A / 5
C / 6	C / 3	(B / 6)
SXX: $\sum t_{1i}^2 = 1104$	$\sum t_{2i}^2 = 1116$	$\sum t_{2i}^2 = 1192$

Table 2: Enumerating common tuples

S_1	S_3	Product of duration	
ABC / 20	ABC / 20	20 × 20	400
AB / 14	AB / 17	14 × 17	238
AC / 16	AC / 13	16 × 13	208
BC / 10	BC / 10	10 × 10	100
A / 10	A / 10	10 × 10	100
B / 4	B / 7	4 × 7	28
C / 6	C / 3	6 × 3	18
SXY_{12}	$= \sum t_{1i}t_{2i}$		1092

S_1	S_2	Product of duration	
ABC / 20	—	—	0
—	BAB / 20	—	0
AB / 14	AB / 11	11 × 14	154
AC / 16	—	—	0
—	BA / 14	—	0
BC / 10	—	—	0
—	BB / 15	—	0
A / 10	A / 5	10 × 5	50
B / 4	B / 9	4 × 9	36
B / 4	B / 6	4 × 6	24
C / 6	—	—	0
SXY_{13}	$= \sum t_{1i}t_{3i}$		264

Table 3: Calculating the distances from the sums of products of duration

	SXY				Distance		
	S_1	S_2	S_3		S_1	S_2	S_3
S_1	1104	1092	264	S_1	0	6.0	42.0
S_2	1092	1116	342	S_2	6.0	0	40.3
S_3	264	342	1192	S_3	42.0	40.3	0

where SXY is the sum of the product of the cumulated duration of each subsequence shared between sequences X and Y. SXX and SYY represent the same measure for X compared with X and Y with Y, respectively, that is, the sum of the square of the cumulated duration of each subsequence. An alternative would be to weight shared subsequences according to the sum of the product of the time in each state – this will yield greater differences between sequences with similar spell order but different durations. For instance, in the example above, S1 and S3 share an `ABC` subsequence, and this is weighted at $20^2 = 400$, the same as the subsequences' contributions to SXX and SYY, rather than $10 \times 10 + 4 \times 7 + 6 \times 3 = 146$ (compared to 152 to SXX and 158 to SYY). However, since the differences in the state-specific durations will feature in the shorter subsequences (`AB`, `AC`, `BC`, `A`, `B` and `C`) this does not compromise the measures' ability to distinguish between similar sequences. Indeed, it is possible to argue that the other approach is deficient in multiply counting the differences. The primary reason for using the subsequence cumulated duration is computational convenience: it requires storing a single datum per subsequence, rather than a vector as long as the number of elements. Where sequences are long, the number of subsequences can be extremely large.

Table 2 shows the calculations of the SXY quantity and Table 3 the final distances. The sequences s_1 and s_2 share the `ABC` strucure so match quite closely, while s_3 is judged quite different, though it does match the other two with the `AB` structure.

Elzinga has implemented many of his proposed measures in his own software, CHESA. A number are also implemented in the R package for sequence analysis, TraMineR (Gabadinho, Ritschard, Müller & Studer, 2011), and this X/t implementation is available in SADI.

5 Time-warping

The concept of time-warping provides a third type of narrative: trajectories may have different or varying speeds such that we can locally compress and expand time to maximise their similarity. The amount of distortion plus the amount of residual difference can be viewed as a measure of dissimilarity (see Figure 1 for an illustration using a sequence that has one real-valued dimension). While this gives a mechanism to "align" parts of sequences, to respond to full or partial matches at the same or different times, it does so by mechanisms that have a different surface logic from OM, one which is much more appealing for cases where we think of time as essentially continuous (such as lifecourse data). While the surface logic is, and resulting dissimilarities can be, quite different from OM and related algorithms, it is interesting to note that it is implemented in a manner very similar to OM's Needleman-Wunsch algorithm.

Time warping has been around as a term for quite a while: for instance, Abbott and Hrycak (1990) use the term to suggest using non-linear time scales (for instance the log of sequence time) to cope with domains where the rate of transition varies with time (e.g., labour market volatility in youth giving way to stability), and in a sense OM warps time by matching patterns at different time-points. However, in processing longitudinal data the term has always had a more specific meaning, which features in the seminal work on sequence analysis, *Time Warps, String Edits and Macromolecules* (Sankoff & Kruskal, 1983).

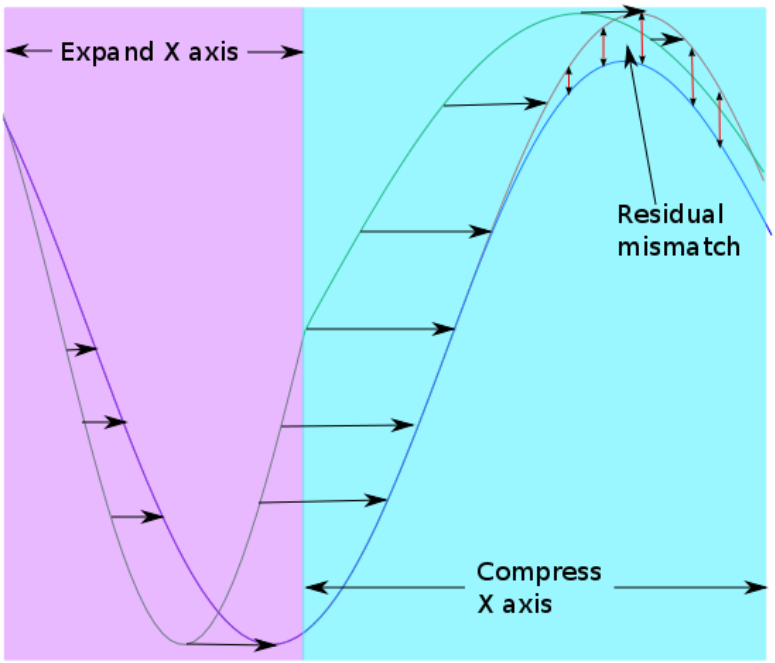

Figure 1: Two one-dimensional sequences compressed and stretched to maximise match

Time-warping has been used widely in computer-science contexts. It was favoured for tasks like speech recognition, signature verification, and other machine learning tasks. It was typically used to match a high-dimensional time-series to a "dictionary" of standard elements, a strategy that does not depend on its being a metric. That is, where two sequences resemble each other well, their TW dissimilarity will reliably be low, but differences between high dissimilarity scores may not be informative. Conceptually it is a continuous time approach, but in practice any time-series data processing must discretise, either by sampling the data at regular intervals (e.g., sound recording samples air pressure 41,000 times per second, unemployment figures are given on a weekly basis) or through periodic summaries (such as income per month, daily cumulated rainfall).

While not all time-warping dissimilarities are metric, a modified time-warping distance measure, the time-warp edit distance (TWED), has been proposed by Marteau (2007, 2008), who has shown it to be metric. I consider the measure here as a competitor to OM for lifecourse data. It is quite similar to OM in its operation, as it uses a state-space distance matrix (functionally equivalent to OM's substitution matrix), and has operations analogous to substitution, insertion and deletion (though the latter two are better thought of as compression and expansion, or even better as compress-A and compress-B). It has a stiffness parameter ν and a gap-penalty, λ.

Formally, time warping is a family of algorithms that can be said to do "continuous time-series to time-series correction" while OM *et al* do "string to string correction" (Marteau, 2007). That is, conceptually time-warping uses continuous time, but it can be shown to work well in discrete time (Kruskal & Liberman, 1983). Marteau shows that there is a low bound to the discrepancy caused by such discretisation for this measure. While TWED can accommodate any sort of state space, and is usually described in terms of $\mathbb{R}^n$, a space composed of possibly many real dimensions where distances between points can be calculated in Euclidean or other terms, there is no difficulty in mapping to a discrete state space where distances between states can be given in a lookup table, the state-space distance matrix. TWED is designed to accommodate irregular time-sampling, but is a little simpler to program when we have fixed time steps, as is the case considered here, and as is typically the case with lifecourse data.

In its internal operation, it differs strongly from OM in that the operations (i) consider consecutive

pairs of tokens in all three operations (ii) has a stiffness parameter that cumulates each time a comparison is made where time is realigned, and (iii) does not edit the content or order of the sequence (insert or delete) but aligns by altering the time dimension (though in reality there is some elision of elements).

The compress operations are costed at $d(s_{i-1}, s_i) + \nu + \lambda$. That is, compressing at point i depends on the similarity of s_i to s_{i-1}, plus the stiffness parameter (ν) and the gap penalty λ.[3] This operation is considered as time-warping, stretching or compressing the time-axis, depending on which sequence is being considered. Matching is TWED's equivalent of substitution: when we stop deleting or warping time, we consider the difference between the now-aligned tokens as a matching cost (with exactly the same effect as substitution). However, the comparison is between consecutive pairs of tokens, and has a stiffness penalty of $2\nu(|i - j|)$, i.e., twice the stiffness parameter times the time dislocation between the two sequences.

TWED offers an alternative to OM that is very similar in terms of implementation, but quite different in its motivation. By virtue of its stretching and compressing operations, and its attention to successive pairs of tokens, it is likely to respect the spell structure of the trajectory better than OM. In this respect, and since it generates metric distances, it may well achieve what LOM and OMv attempted.

5.1 The TWED algorithm

In practice, the implementation and application of TWED is very similar to OM. It takes as parameters a matrix of distances between the states (identical to the substitution cost matrix, except that the operation is not thought of as substitution), and two parameters for stiffness and gap penalty that are broadly analogous to *indels* in that they facilitate or deter compression/expansion. Thus we are looking for full and partial similarity at the same or near location, where "partial" and "near" are affected by our parameterisation. The same description applies to Optimal Matching.

Also similar to OM is the internal detail of the implementation. As is well known, OM can be described as a recursive algorithm such that the distance between sequence A (up to element p) and sequence B (up to element q) is given by:

$$\delta^{OM}(A^p, B^q) = min \begin{cases} \delta^{OM}(A^{p-1}, B^q) & + \iota \\ \delta^{OM}(A^{p-1}, B^{q-1}) + d(a_p, b_q) & \\ \delta^{OM}(A^p, B^{q-1}) & + \iota \end{cases}$$

where ι is the *indel* cost and $d(a_p, b_q)$ the substitution cost between element p of sequence A and element q of sequence B. This can be programmed efficiently in $p \times q$ operations.

TWED can be expressed in an identical structure:

$$\delta^{TW}(A^p, B^q) = min \begin{cases} \delta^{TW}(A^{p-1}, B^q) & + d(a_p, a_{p-1}) & & + \nu d(t_{a_p}, t_{a_{p-1}}) + \lambda \\ \delta^{TW}(A^{p-1}, B^{q-1}) + d(a_{p-1}, b_{q-1}) + d(a_p, b_q) & & & + 2\nu d(t_{a_p}, t_{b_q}) \\ \delta^{TW}(A^p, B^{q-1}) & + d(b_q, b_{q-1}) & & + \nu d(t_{b_q}, t_{b_{q-1}}) + \lambda \end{cases}$$

On the simplifying assumption that observations are taken at fixed 1-unit intervals (i.e., $t_{i+1} - t_i = 1$), this simplifies to:

$$min \begin{cases} \delta^{TW}(A^{p-1}, B^q) & + d(a_p, a_{p-1}) & + \nu & + \lambda \\ \delta^{TW}(A^{p-1}, B^{q-1}) + d(a_p, b_q) + d(a_{p-1}, b_{q-1}) & & + 2\nu|p - q| & \\ \delta^{TW}(A^p, B^{q-1}) & + d(b_q, b_{q-1}) & + \nu & + \lambda \end{cases}$$

[3] More generally, ν should be multiplied by the time difference between $t_{[i-1]}$ and t_i, but that is always 1 in the sort of data we are using.

The first row represents compressing sequence A, the third compressing sequence B (equivalently expanding sequence A), and the middle represents the residual-mismatch cost. While the structural analogy to OM is strong, the manner in which the costings work is different: the equivalent of *indel* operations take context into account: if we compress A the cost depends on the pair of values, $d(a_p, a_{p-1})$ as well as the parameters ν and λ (stiffness and gap penalty). Thus compression is cheaper within a spell in the same state, like OMv, and is cheaper where the previous value is similar, like LOM. The cost of the residual mismatch is also an incomplete analogy to substitution: we look at the mismatch at both t_i and t_{i-1} [4], and incur an extra penalty depending on how far apart the locations are in the unwarped sequences ($2\nu|p - q|$). Effectively we are paying a cumulating penalty for alignment, both in the compression/expansion operation and in the subsequent comparison.

Thus, though the structure is very similar, we can expect the resulting distances to be different from OM, since on the one hand more context is taken into account in the elementary operations, and the penalty for alignment bears not only in the act of alignment (compression/expansion or insertion/deletion) but also in the comparison of segments warped or aligned out of their original locations.

Table 4: The "linear" and "flat" state-space distance matrices

Linear matrix				
	FT	PT	UE	Non
Full-time employed	0	1	2	3
Part-time employed	1	0	1	2
Unemployed	2	1	0	1
Non-employed	3	2	1	0

Flat matrix				
	FT	PT	UE	Non
Full-time employed	0	1	1	1
Part-time employed	1	0	1	1
Unemployed	1	1	0	1
Non-employed	1	1	1	0

6 Some results

In what follows I compare OM, TWED and X/t using an example data set which consists of 6 years of monthly labour market data for women who have a birth at the end of the second year (derived from British Household Panel Study data). The state space differentiates between full- and part-time employment, unemployment and non-employment.

Additionally to OM, TWED and X/t, I also present results for Hamming distance, which can be considered as a special case of OM where alignment through *indels* is suppressed. The Hamming distance makes for a very simple "narrative" where the mapping between state-space distances and sequence distances is a simple summing of the distance at each time point: full or partial similarity at the same time. This is an important comparison because where sequences tend to have long spells and differ largely in when exactly transitions occur (as is often the case with lifecourse data), the Hamming distance will be fairly low, and the amount it can further drop if alignment were allowed will be relatively small. It may be that with OM and other measures we gain only a little information (through better distances) at the expense of a much more complicated story about similarity.

[4] DRAFT NOTE: Marteau's description of TWED is not consistent, and this double counting is not always present; I need to test what happens when only the t_i pair is compared.

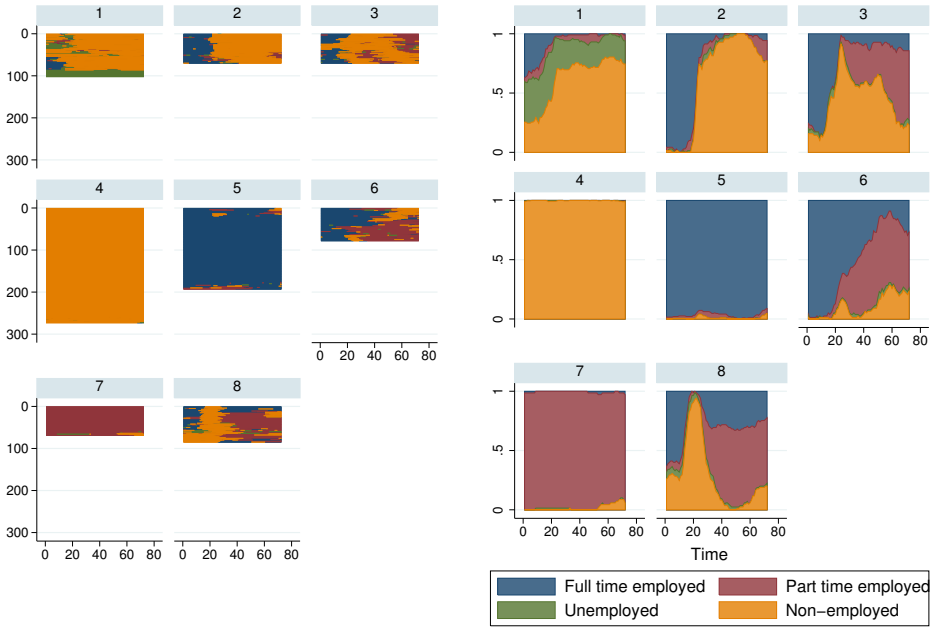

Figure 2: Indexplot (l) and chronogram (r) for Hamming distance, linear state space

In the next section I compare Hamming, OM and TWED using a very simple state-space distance structure, which places the four states on a single dimension (hence "linear") with equal intervals between them (see Table 4, upper panel). Since X/t does not use such information on state-space distances, it cannot be fairly compared in this context, so I also compare X/t with Hamming, OM, and TWED using the another simple state-space cost structure, where all states are equally dissimilar (hence "flat": see Table 4, lower panel):

6.1 Patterns

For each measure, distances are calculated and grouped into clusters using Ward's method. Eight clusters are chosen for convenience and ease of exposition. Results for the "linear" cost structure are presented in Figures 2, 3 and 4, which show indexplots (sorted within clusters by the nested subcluster structure[5]) and state-distributions (or chronograms). Cluster solutions have been re-ordered to maximise agreement across measures, in so far as possible. While there are clearly differences in the assignment of sequences to clusters, there is a remarkable level of similarity (especially considering that cluster analysis can be relatively unstable). In terms of differences, the Hamming measure, for instance, clusters the handful of 100% unemployed cases with cases that mix employment with unemployment, while OM and TWED put them in a distinct cluster. OM takes some cases from Hamming's cluster 8 and puts them in cluster 4 (which is really closer to Hamming's cluster 6: it is not entirely possible to reconcile the cluster solutions) – presumably in the latter case the differences have to do with Hamming seeing a block of similarity at a certain time (in this case, non-employment immediately around the birth) which matches well with other cases in cluster 8, whereas OM can recognise similarity in the full-time/non-employed/full-time pattern which features in cluster 4, though not all with the same timing. TWED picks up rather more of a spike in non-employment around the birth in cluster 4 than OM does, but less in cluster 8. However, notwithstanding these differences, for all three measures, the overall character of clusters 3, 4 (6 for

[5]As will become apparent below, clustering does not recover stable natural clusters for OM and TWED. Thus cluster solutions are somewhat unstable and arbitrary. Nevertheless, the full hierarchical structure of the cluster analysis captures a lot of the information embedded in the distance matrix. Sorting index plots within cluster by the nested subcluster structure (i.e. presenting the dendrogram order), thus makes a lot more useful information available to the eye.

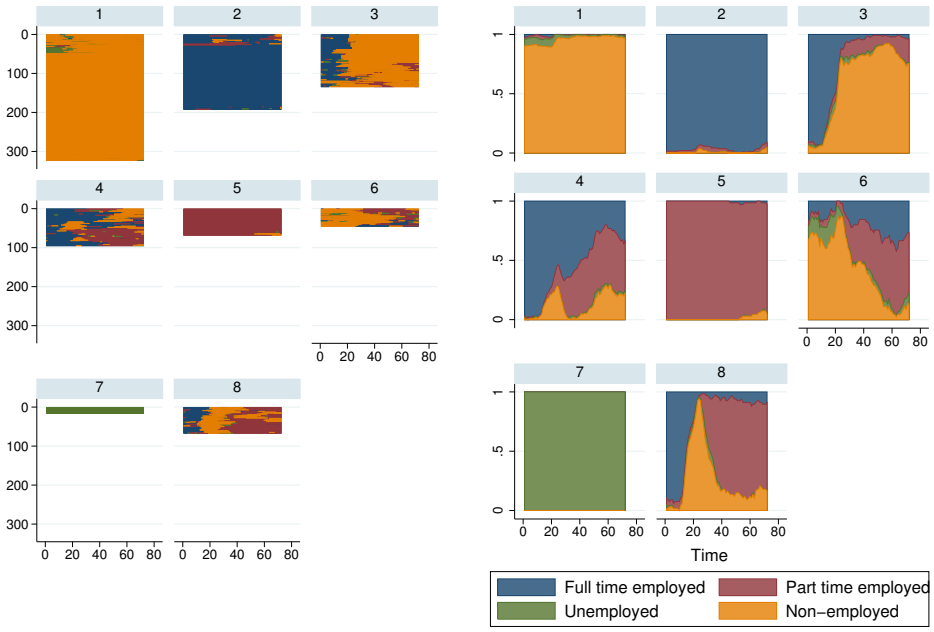

Figure 3: Indexplot (l) and chronogram (r) for OM distance, linear state space

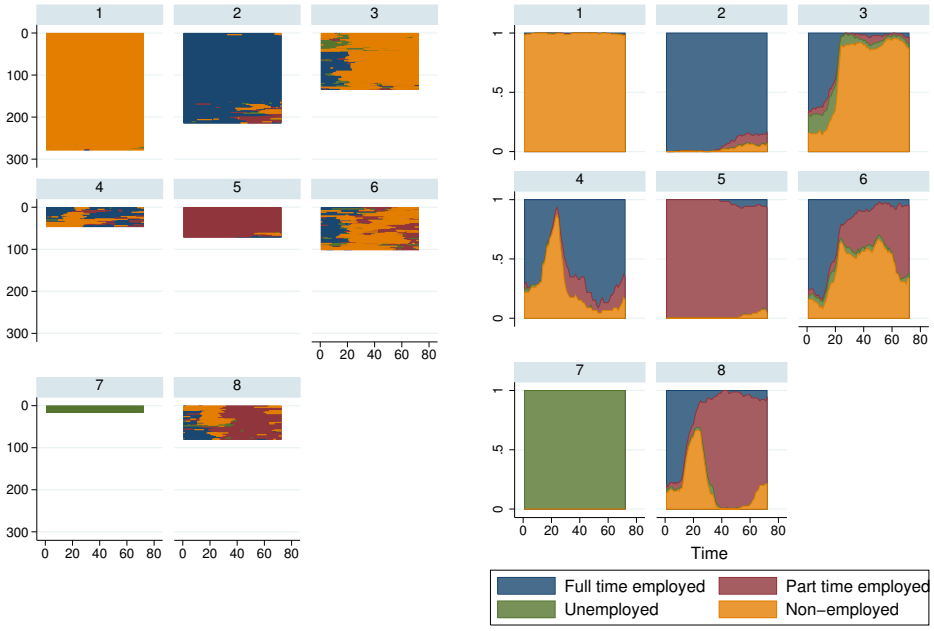

Figure 4: Indexplot (l) and chronogram (r) for TWED distance, linear state space, $\nu = \lambda = 0.5$

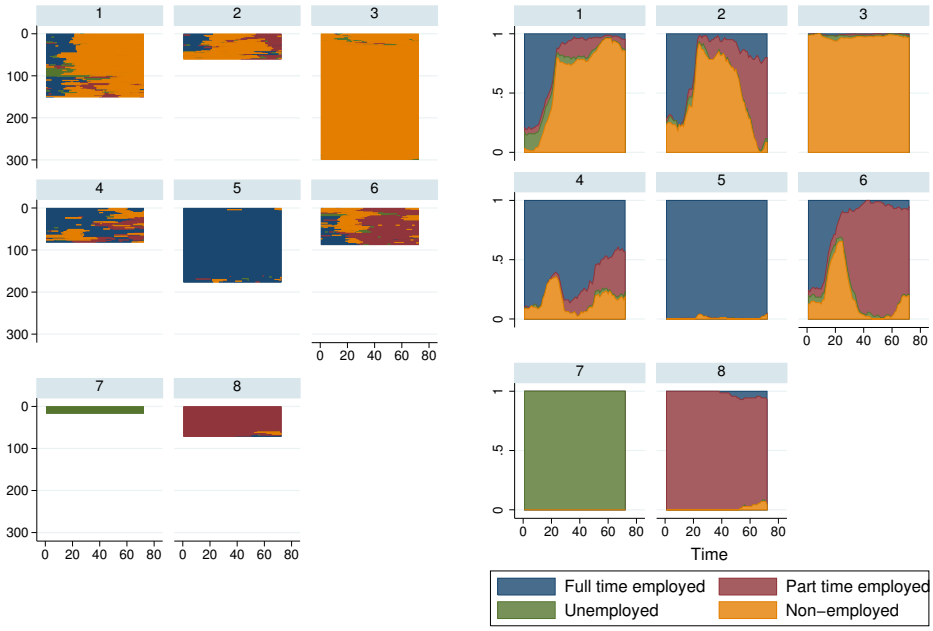

Figure 5: Indexplot (l) and chronogram (r) for Hamming distance, flat state space

Hamming) and 8 is quite similar (broadly, from full-time to predominantly non-employment, though with some part-time, particularly for Hamming; full-time with a greater or lesser period of non-employment around the birth with a return to full-time or part-time work, with the part-time increasing over time; from full-time through non-employment around the birth, to part-time). That clusters 1 (4 for Hamming), 2 (7) and 5 are similar is not much of an achievement as they are very simple clusters, dominated by single-spell sequences.

Looking at the second set of graphics, with the "flat" state-space distance matrix (implicit in the case of X/t), we see the same sort of similarity across the first three measures (Figures 5, 6 and 7). If anything the similarity is stronger with the flat matrix than with the linear, possibly because the absence of the strong good or bad matches possible with the linear matrix means that there is less opportunity to benefit from time-displacement, causing OM and TWED to converge on Hamming. However, it is worth emphasising that the difference between measures is much less than the difference within measures across state-space matrices: switching from the linear to the flat matrix makes a bigger difference than switching between algorithms. It should not be surprising that the state-space distance structure should matter: it creates quite different landscapes, one in which full-time work and non-employment are very distinct, while unemployment and part-time are more similar, and the other in which everything is equally distinct so nothing stands out.

Figure 8 displays the X/t results. While all the foregoing results show a good deal of similarity (even, to a lesser extent, across the two state-space distance matrices), Elzinga's measure produces radically different results. Single-spell sequences are naturally maximally distinct (or identical) so are pulled out in pure simple clusters, without the admixture of sequences that are only close to 100% in that state. There are three substantial clusters with moderate numbers of transitions, and two of these (2 and 6) correspond relatively well with patterns found by the other measures, though cluster 1 is very heterogeneous (and across the eight clusters, only 71% of sequences fall in the corresponding OM cluster, 42% if we exclude the three clusters of simple sequences). Finally there are two tiny clusters containing sequences with high numbers of transitions (and therefore complex sets of subsequences) which are very distinct from all other sequences. It is evident that the number of transitions is very important

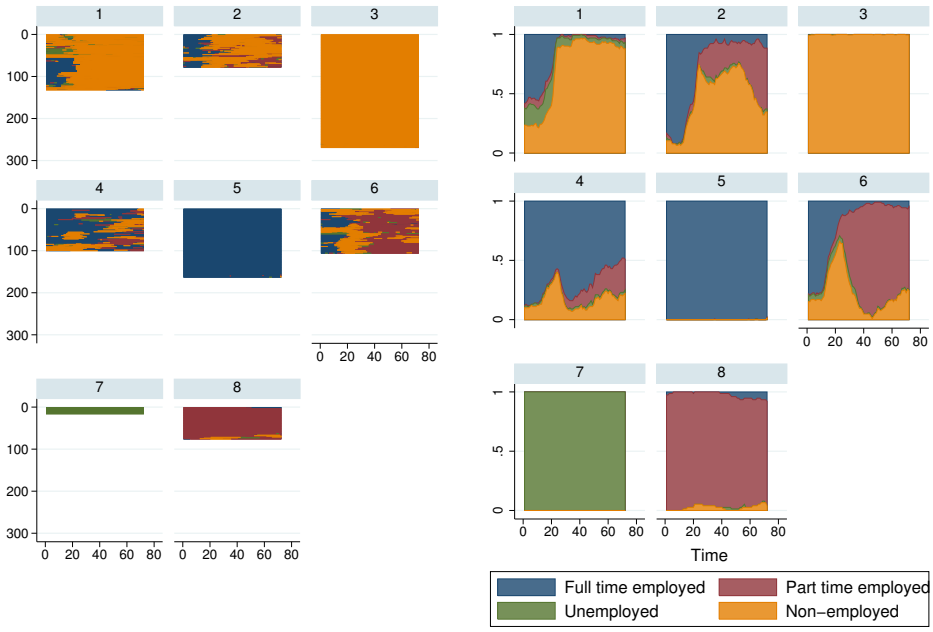

Figure 6: Indexplot (l) and chronogram (r) for OM distance, flat state space

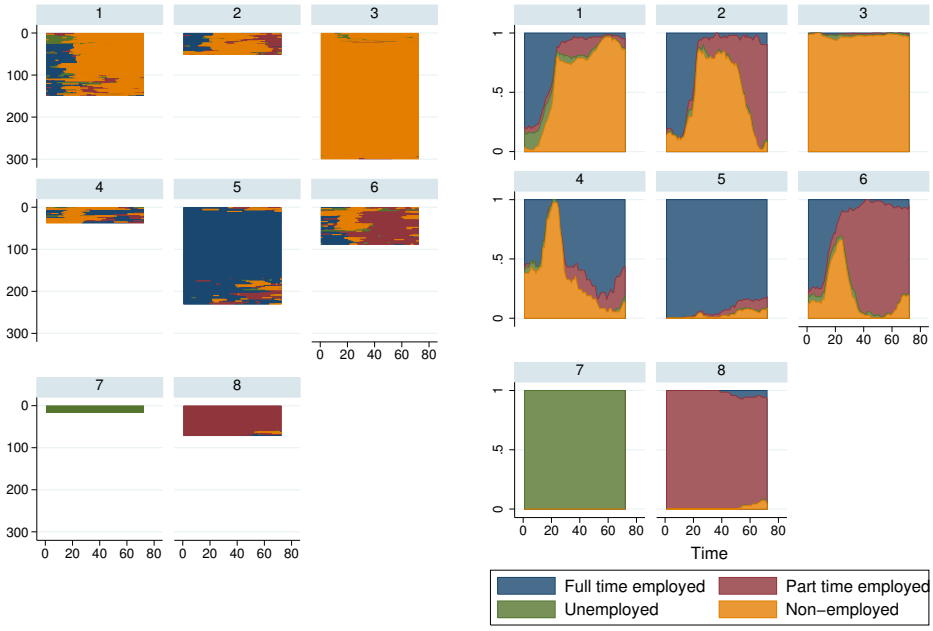

Figure 7: Indexplot (l) and chronogram (r) for TWED distance, flat state space

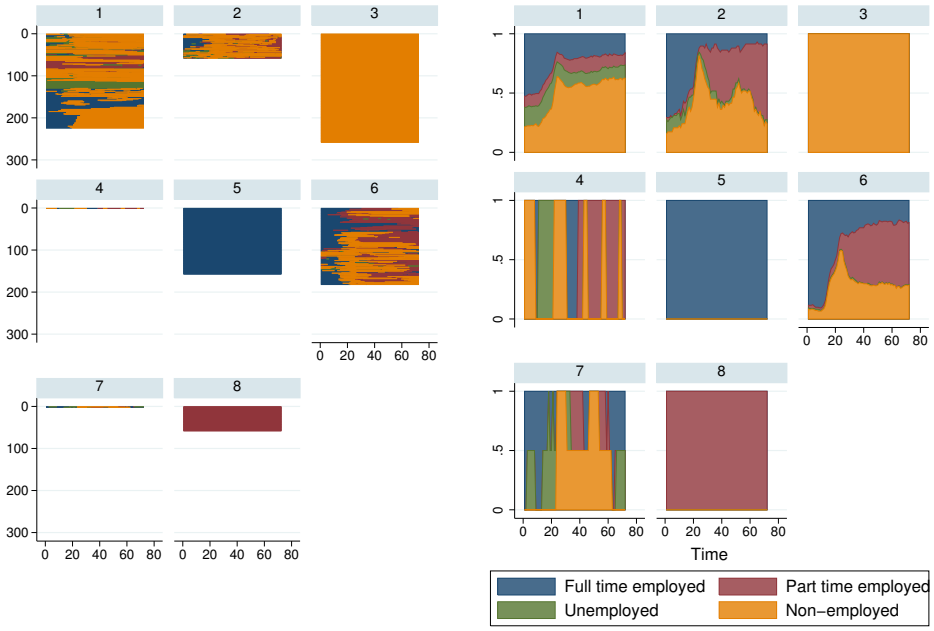

Figure 8: Indexplot (l) and chronogram (r) for X/t distance

(clusters 1, 2 and 6 have respectively 2.9, 6.5 and 3.8 spells on average, and the two micro-clusters 11 and 12), and that ever being in the same state is important. Moreover, while duration is explicitly brought into consideration, this does not tie time down in the same way as the other measure: i.e., since according to all common-subsequence measures, `xxABC`, `AwBwC` and `ABCzz` will all be equidistant because the measures focuses on order (if weighted by time) and not time, while OM and TWED penalise for temporal displacement. Where the substantive concern is with order, X/t has big advantages, but with life-course sequences such as these, where there is an explicit anchor time point (here the birth at the end of year two, but more typically the start of the trajectory) and a developmental time-scale of some sort (here, changing constraints on labour market activity as the child ages) measures that pick up "full or partial similarity at the same or similar time" produce more useful patterns.

6.2 Multidimensional scaling of inter-sequence distances

The foregoing cluster analysis reflects a typical work-flow with sequence analysis, where distances are converted into a data-driven classification of sequences. We can understand a little more about how the clustering generates the classification by looking at the space implied by the distance. Do the sequences have a lumpy distribution in this space (and hence generate robust clusters)? Can we understand something about the overall pattern of distances: which sequences are distant from each other, which more close? Does this structure emerge in a similar way across measures or are there systematic differences. In this section we look at multi-dimensional scaling of the OM, TWED and X/t distances.

Figure 9 graphs the first two dimensions for the OM distances, distinguishing the 8-cluster solution, using the linear state space. The sequences are located in an approximately oval cloud, with clear poles marked by sequences that are 100% full-time employed or 100% non-employed, and less distinct poles dominated by the other two states (more easily detected with more dimensions). Some sequences appear to form strings: inspection of the data shows these tend to contain one transition, where as the transition occurs earlier they move progressively closer to the pole concerning the second state. However, apart from these structures, the distribution through space is relatively even, such that there is no natural clustering. As is clear, the cluster solution that Ward's algorithm generates does not identify well defined

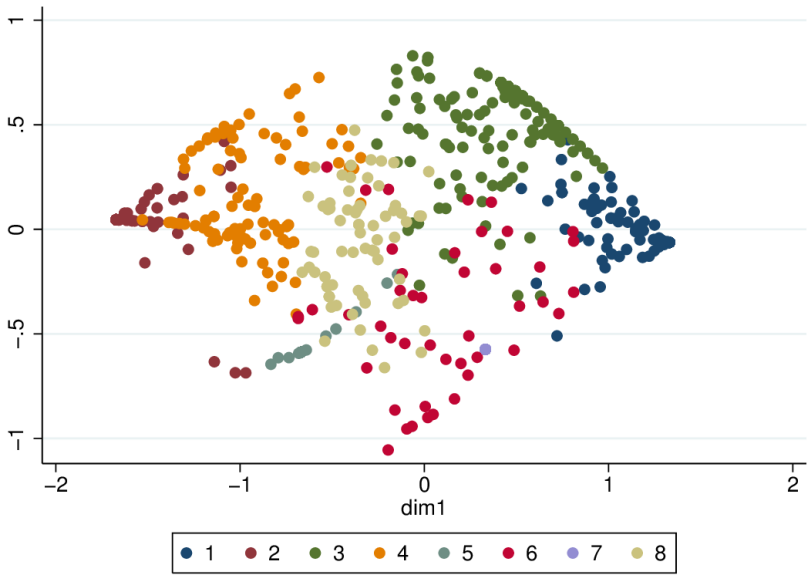

Figure 9: Multidimensional scaling of OM distances, by cluster

groups. However, it does partition the space into distinct areas, albeit with rather arbitrary boundaries. In that much, we can consider it a useful data reduction strategy but not a successful discovery of natural clusters.

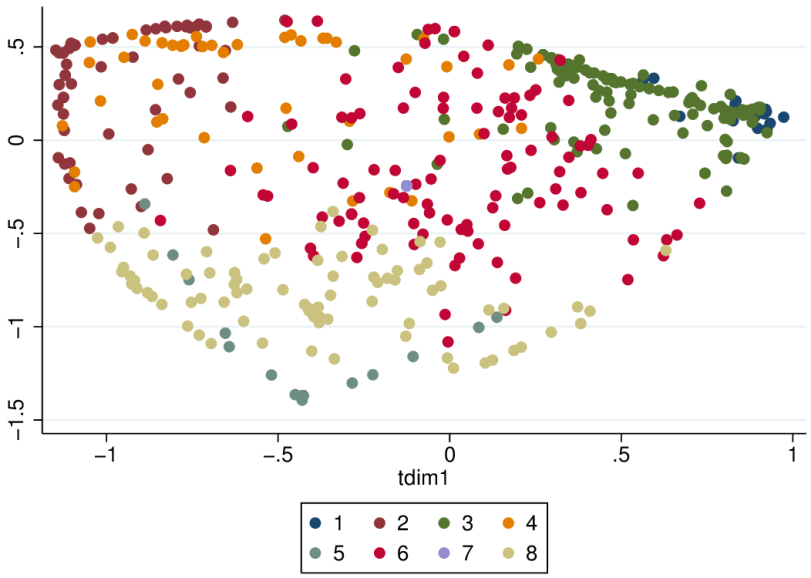

Figure 10: Multidimensional scaling of TWED distances, by cluster, linear state space, $\lambda = \nu = 0.5$

Figure 10 shows the corresponding graph for TWED. The shape of the cloud is different, but the qualitative statements relating OM also apply here. The poles and strings are apparent, as is the absence of natural clustering.

The structure apparent in the MDS of the X/t distances is very different. Figure 11 shows the whole data set and Figure 12 zooms in on the core, where the vast majority of the sequences lie. The main graph shows a single dense cluster and a number of very remote points. The most remote points consist of sequences with very high numbers of spells and, as we move towards the cluster, sequences get simpler.

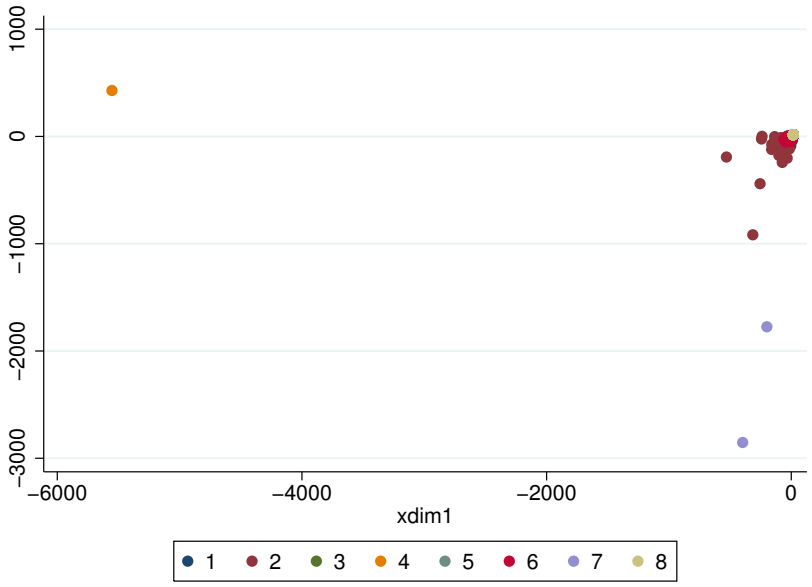

Figure 11: Multidimensional scaling of X/t distances, by cluster

Figure 13 replicates Figure 12 displaying number of spells rather than cluster membership, and this demonstrates the pattern very clearly: the very simplest sequences are in the top right, and as we move left the number of spells increase steadily.

In this measure, sequences with many spells are further from all other spells, particularly those with fewer spells, but also from other high-spell sequences with different patterns. Low-spell sequences, and *a fortiori* single-spell sequences have lower distances, but are still distinct from each other (when different). Because simple sequences are common, they will tend to form clear clusters despite their relatively low distances to other sequences – they form dense natural clusters in the space, clearly separated from each other albeit by small distances. Correspondingly, while high-spell sequences are distant from others, they are sparsely distributed in the space and therefore tend to make quite loose clusters, which can be affected by number of spells at least as much as substantive pattern.

The main conclusions we can draw from this analysis is that OM and TWED are relatively similar and do not show a natural cluster structure, while X/t gives a very different result, with strong natural distinctions between clusters of simple sequences, but lower power to distinguish complex sequences.

6.3 Correlation analysis

As we have seen, cluster analysis is a somewhat unstable technique unless there are strong natural clusters in the data, giving results that are sensitive to small changes in parameterisation or sampling. However, the underlying distances are not the cause of this instability, so another approach is to compare the distances directly. That is, comparing cluster solutions across measures can be messy and risks being subjective, but we can get a alternative overview of the various measures by looking at correlations between the distance matrices. Focusing on a single number makes for a less rich but more tractable comparison, and it is clearly the case that the difference between a pair of measures has many more than one degree of freedom, and differences two highly correlated measures may well be important. However, the extent to which measures agree about sequence pairs is important, and looking at their correlation is clearly informative.

Tractability, moreover, allows us to compare larger numbers of measures for context: here as well as OM, TWED (with three parameterisations) and X/t, I include LOM (with two parameterisations) and

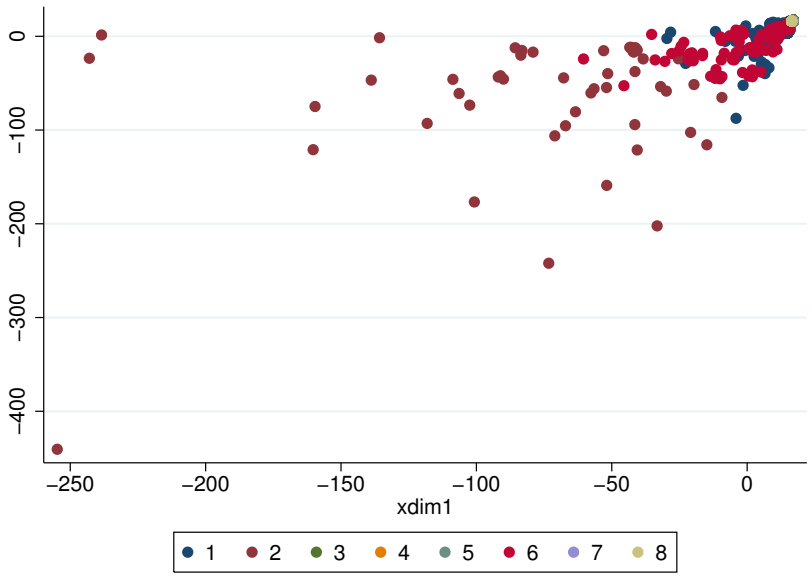

Figure 12: Multidimensional scaling of X/t distances, by cluster, zooming in

OMv. I also include Lesnard's dynamic Hamming measure (Lesnard, 2010), which is a version of the Hamming distance that calculates the state-space matrix dynamically from the time-dependent transition rates. Where relevant, I apply both the linear and flat state-space matrices.

The resulting correlation matrix is presented in Figure 14 as a heatmap. Rows and columns have been sorted manually to maximise coherence. To get a sense of the scale, white cells correspond to coefficients of 0.98 and higher, yellow about 0.85, and the darkest in the range from about 0.1 to 0.3.

The first thing to emerge clearly is that Hamming, OM, LOM and OMv (with the same state-space matrix) produce distances with very high correlations, between 0.984 and 0.999. This is true for the linear and flat matrices, but not across matrices: the state-space distance structure matters quite a bit more than the measure, within this set of measures. Effectively, for most pairs of sequences, particularly simple ones, the difference between Hamming and the more complicated distance measures is negligible or zero. This is not to say that, for the pairs where the measures do differ, the time-dislocating measures do not offer significant value added.

The second thing to emerge is that for the flat matrix, TWED is also part of this group, highly correlated with Hamming and OM. Related to this is the fact that for flat *and* linear state-space matrices, TWED is close to the OM/Hamming results for the flat matrix, though the correlation is slightly lower across matrices (e.g., for TWED with the first parameterisation and the linear matrix, the correlation with OM with the flat matrix is 0.937, versus 0.983 when both TWED and OM use the flat matrix). Thus we see evidence that TWED is less affected by the state space matrix for the parameterisations used. Dynamic Hamming also falls in the flat-matrix group, presumably because the dynamic state-space values calculated from the transition rates are not distinctly different from the flat matrix (i.e., there is no pair of states with distinctly higher transition rates; this can clearly differ with other data sets).

The final piece of information to be gleaned is the complete distinctness of the X/t measure. Its correlations range from 0.078 with OM (linear matrix) to 0.297 with TWED (linear matrix, first parameterisation). It is only with TWED that the correlation exceeds 0.13, suggesting that TWED respects order in a manner like, but rather weaker than, X/t, and more than the other measures.

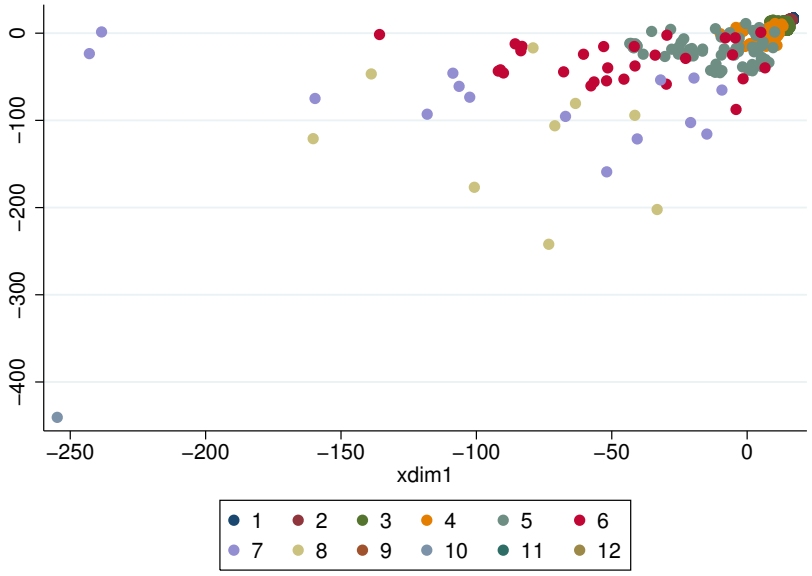

Figure 13: Multidimensional scaling of X/t distances by number of spells, zooming in

6.4 Parameterising TWED and similarity to other measures

The correlation analysis suggests that while there is a big gulf between X/t and the other measures, TWED may serve as an intermediate measure, though closer to OM *et al* than X/t. However, TWED is not well understood, and in particular the effects of its parameterisation is not clear. The ν "stiffness" parameter bears on all three operations, making compression and expansion more expensive, and penalising comparisons proportionately with their displacement, and the λ gap penalty adds further to the cost of expansion and compression. However, it is not clear what the consequences of the parameterisation is – while high values will reduce TWED to Hamming distance, it is not clear what lower values do. We have seen so far that changing ν between 0.0 and 0.25, holding λ constant at 0.5, has relatively little effect on the correlation of the distances with other measures (values of the parameters not much higher than these constrain the measure to Hamming; this is analogous to raising *indel* with OM, but takes effect at lower values).

To explore this issue, I present an analysis of the correlation structure, comparing OM and X/t, with TWED with multiple parameterisations: each parameter takes values between 0 and 1 (0.0, 0.002, 0.005, 0.01, 0.02, 0.05, 0.1, 0.2, 0.5 and 1.0). Figure 15 plots the correlation of TWED with OM (using the flat matrix) and X/t. For high levels of either parameter (where high means approximately 0.2 to 1), TWED is very close to OM and is almost indistinguishable as either value approaches 1 (in fact, each is near indistinguishable from Hamming), but for lower values the correlation with OM declines slowly at first but then more quickly, and the correlation with X/t rises initially quickly but finally slowly, peaking at about 0.6 as the parameter values reach zero. We thus see that varying the parameters of TWED can yield distances that differ strongly from OM, in a way that responds to dimensions of sequence similarity that X/t detects (the implication is order) but without abandoning the fundamental sensitivity to timing, albeit with an elastic interpretation.

In this data set, the two parameters seem to be almost interchangeable, such that their sum seems to be driving most of the pattern. This suggests that expansion and compression are doing a lot of the work, particularly at lower values (i.e., further from OM, closer to X/t), since both parameters bear on those operations, while the stiffness parameter, ν, alone bears on the residual match operation. With other data sets, the overall pattern is the same, with a little more evidence of independent effects of the two

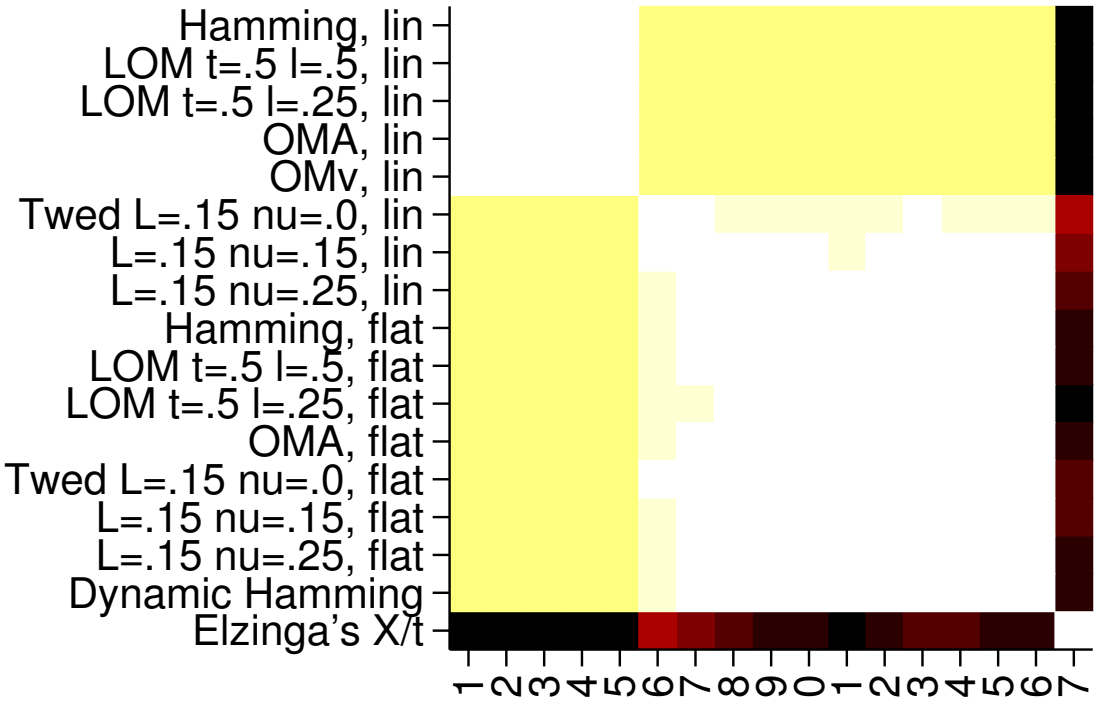

Figure 14: Correlations between a variety of measures, BS data

parameters.

6.5 What the analysis tells us

The comparison between measures using the mothers' labour market data throws a good deal of light on the difference between measures. One striking finding is that there is little difference between many of the measures, and that the state-space distance matrix is often far more important than the algorithm. The difference between the time-dislocating measures and Hamming can be very small, viewed through the lens of the correlation of the distance matrices, though cluster analysis can amplify small differences (not always pathologically, either, one hopes; such small differences are likely to have to do with sociologically interesting, higher-transition sequences).

The TWED measure has strong commonalities with OM at an algorithmic level, though its narrative is different. We see that at high parameter values it is quite similar to the OM/Hamming complex, though its sensitivity to the state-space matrix seems lower. The combinatorial duration-weighted X/t measure is very distinct: while it picks up some of the same broad patterns at the cluster analysis level (and there, largely by virtue of isolating the single spell sequences, as do the other measures with a little less efficiency) it gives a very different overall account of sequence similarity, and correlates poorly with all the other measures. TWED, however, provides a bridge, approaching X/t at very low levels of its gap and stiffness parameters, suggesting that at high levels of the parameters it is picking up on the time-structure of the sequences in a manner very similar to OM *et al* and at low levels, picking up on order information in a manner quite similar to X/t.

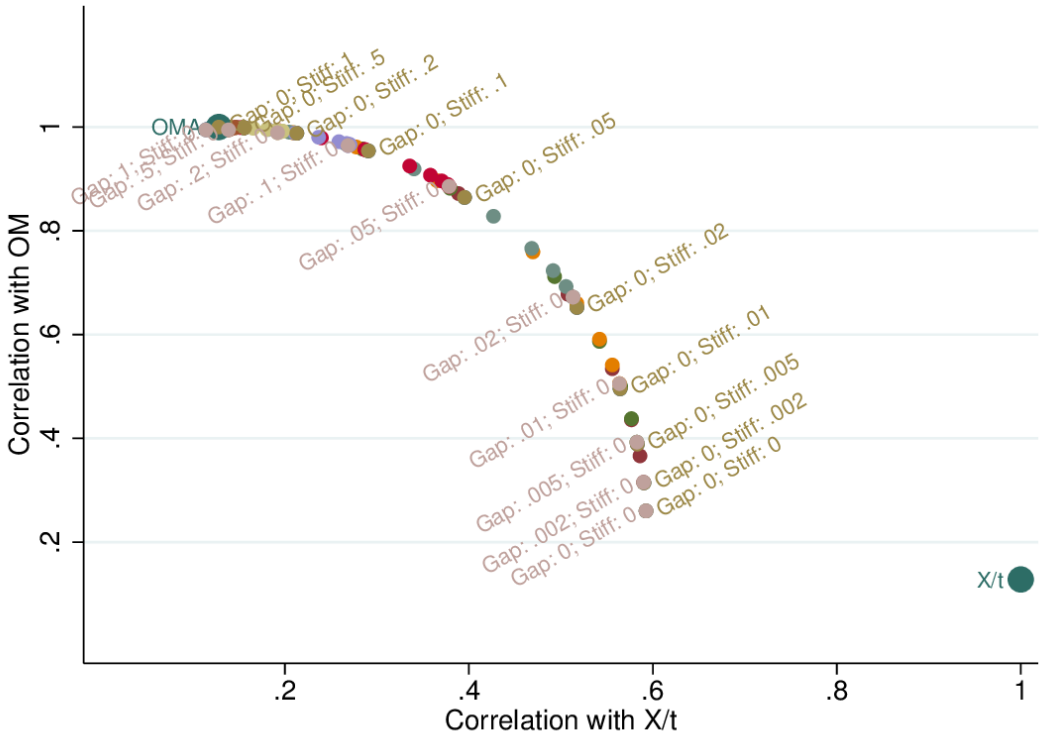

Figure 15: Correlations between TWED, OM and X/t for varying gap and stiffness parameters. Selected points labelled with their λ (gap penalty) and ν (stiffness) parameter values.

7 Conclusion

Inter-sequence distance measures are deterministic algorithms which map state space onto sequence space according to a set of fixed rules. We can understand the mapping in terms of the algorithmic rules, though it is hard to predict what the consequences will be for a given empirical domain or data set. We can also understand the mapping in more heuristic terms, e.g. by considering the algorithm as a model of the data generating process, or as indirectly capturing elements of inter-sequence relationships in a way we can describe more generally.

There may be situations where a distance measure's operation corresponds closely enough to the data generating process that it can be seen as modelling it, but these are rare. Some commentators see the OM algorithm as directly modelling the processes of divergence of DNA across generations. This is not correct: though there are some similarities, OM's operations do not accurately model DNA transcription and mutation, but are driven by algorithmic tractability. What is important for OM's utility in the context is (i) that DNA is indeed a token string and (ii) that DNA does have a pattern of relatedness (closeness due to inheritance) that will show up as potentially displaced matching patterns of tokens. OM is not like, say, the Fisher–Wright model, which does explicitly attempt to model the effect of inheritance on population genetics; rather it is a computationally-efficient heuristic that captures significant features of the phenomenon.

As a heuristic, it will also work for other domains, more or less well depending on the extent to which the token-string representation, with operations on context-less tokens, captures the nature of similarity. Even where it does not work perfectly, it will still capture similarity.

What this paper has been concerned with is both algorithm and the story or narrative with which we can represent the heuristic, and the applicability of distance measures to life course data. In a general

sense the narrative is one of mapping between state space and sequence space, starting logically with the simplest mapping, the Hamming distance (full or partial similarity at the same time), and then moving on to measures that allow time-dislocation in similarity. It has become evident that part of OM's success in the sociological sequence analysis (despite worries about its non-applicability) has to do with its achievement of a difficult job, to wit, efficient production of metric distances between sequences. We have seen alternatives that attempt to stay in the same paradigm as OM fail for technical reasons. LOM and OMv attempt to make OM more relevant to lifecourse data by taking account of context, but lose the metric property. The combinatorial approach of X/t has a radically different narrative (the same states in the same order), and while this narrative is sociologically very attractive, in practice the focus on time as order means a weaker connection with time as calendar or developmental scale; calendar and development are often important in life course perspectives. And while X/t also tends to make some very strong distinctions, it also fails to separate complex sequences where it would be sociologically attractive to do so.

In terms of narrative and results, TWED offers a real alternative. While the mechanics of the algorithm are extremely similar to OM, with direct mapping between the three elementary operations, it has a different genealogy: while it works with sequences of discrete tokens, it is consistent with and derived from a continuous-time perspective. Because of this genealogy and because the implementation explicitly takes context into account, it offers us an escape from the contextless-token problem while providing a metric distance. As we have seen, empirically it can range between results that are very close to Hamming and OM, when its gap and stiffness parameters are high, to results that move strongly towards the order-sensitive results of the duration-weighted combinatorial measure, X/t. Thus it gives us a way of moving between calendar- and order-oriented time.

In some ways the result of this investigation is negative: in so far as we started with a worry about the validity of OM for lifecourse data, a part of the finding is that many (but not all) different measures produce remarkably similar results (and that the structure of the state space can matter much more than the algorithm). That is, OM may be technically inadequate but its problems have little consequence; its narrative produces results consistent with quite dissimilar narratives. However, it has also been shown that measure does matter, that differences do exist, particularly between time as scale and time as order, and that with other algorithms we have alternatives to OM's discrete-token string-editing paradigm.

References

Abbott, A. (1995). A comment on "Measuring the agreement between sequences". *Sociological Methods and Research*, *24*(2), 232–243.

Abbott, A. & Hrycak, A. (1990). Measuring resemblance in sequence data: an optimal matching analysis of musicians' careers. *American Journal of Sociology*, *96*(1), 144–85.

Abbott, A. & Tsay, A. (2000). Sequence analysis and optimal matching methods in sociology. *Sociological Methods and Research*, *29*(1), 3–33.

Barban, N. & Billari, F. (2012). Classifying life course trajectories: a comparison of latent class and sequence analysis. *Journal of the Royal Statistical Society Series C*, *61*(5), 765–784.

Dijkstra, W. (1994). Sequence – a program for analysing sequential data. *Bulletin de Méthodologie Sociologique*, *43*, 134–142.

Dijkstra, W. & Taris, T. (1995). Measuring the agreement between sequences. *Sociological Methods and Research*, *24*(2), 214–231.

Elzinga, C. H. (2003). Sequence similarity: a non-aligning technique. *Sociological Methods and Research*, *32*(1), 3–29.

Elzinga, C. H. (2005). Combinatorial representations of token sequences. *Journal of Classification*, *22*(1), 87–118.

Elzinga, C. H. (2006). *Sequence analysis: metric representations of categorical time series*. Free University of Amsterdam.

Elzinga, C. H. & Wang, H. (2012). *Versatile string kernels*. Paper presented at LaCOSA conference, Lausanne, June 6-8 2012.

Gabadinho, A., Ritschard, G., Müller, N. S. & Studer, M. (2011). Analyzing and visualizing state sequences in R with TraMineR. *Journal of Statistical Software*, *40*(4), 1–37. Retrieved from http://www.jstatsoft.org/v40/i04

Halpin, B. (2010). Optimal matching analysis and life course data: the importance of duration. *Sociological Methods and Research*, *38*(3), 365–388.

Hollister, M. (2009). Is optimal matching suboptimal? *Sociological Methods and Research*, *38*(2), 235–264.

Kruskal, J. B. & Liberman, M. (1983). The symmetric time-warping problem. In D. Sankoff & J. B. Kruskal (Eds.), *Time warps, string edits and macromolecules* (pp. 125–161). Reading, MA: Addison-Wesley.

Lesnard, L. (2010). Setting cost in optimal matching to uncover contemporaneous socio-temporal patterns. *Sociological Methods and Research*, *38*(3), 389–419.

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics-Doklady*, *10*(8), 707–710.

Levine, J. H. (2000). But what have you done for us lately? Commentary on Abbott and Tsay. *Sociological Methods and Research*, *29*(1), 34–40.

Lovaglio, P. G. & Mezzanzanica, M. (2013). Classification of longitudinal career paths. *Quality Quantity*, *47*(2), 989–1008. doi:10.1007/s11135-011-9578-y

Marteau, P.-F. (2007). Time Warp Edit Distance with Stiffness Adjustment for Time Series Matching. *ArXiv Computer Science e-prints*. eprint: cs/0703033

Marteau, P.-F. (2008). Time Warp Edit Distance. *ArXiv e-prints*, *802*. eprint: 0802.3522

McVicar, D. & Anyadike-Danes, M. (2010). Does optimal matching really give us anything extra for the analysis of careers: an application to British crime careers. *(under review)*.

Needleman, S. et al., Wunsch, C. et al. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, *48*(3), 443–453.

Sankoff, D. & Kruskal, J. B. (Eds.). (1983). *Time warps, string edits and macromolecules*. Reading, MA: Addison-Wesley.

Wu, L. L. (2000). Some comments on "Sequence analysis and optimal matching methods in sociology: review and prospect". *Sociological Methods and Research*, *29*(1), 41–64.