Topological mapping for visualization of high-dimensional historical linguistic data

Hermann Moisl

Newcastle University, UK

Abstract

This paper addresses an issue in visualization of high-dimensional data abstracted from historical corpora whose importance in quantitative and corpus linguistics has thus far not been sufficiently appreciated: the possibility that the data is nonlinear. Most applications of data visualization in these fields use linear proximity measures which ignore nonlinearity, and, if the data is significantly nonlinear, can give misleading results. Topological mapping is a nonlinear visualization method, and its application via a particular topological mapping method, the Self-Organizing Map, is here exemplified with reference to a small historical text corpus.

Keywords: Nonlinearity, topological mapping, self-organizing map, data visualization, high-dimensional data, historical linguistics

Introduction

Discovery of the chronological or geographical distribution of collections of historical text can be more reliable when based on multivariate rather than on univariate data because, assuming that the variables describe different aspects of the texts in question, multivariate data necessarily provides a more complete description. Where the multivariate data is high-dimensional, however, its complexity can defy analyis using traditional philological methods. Increasingly, the first step in interpreting such complexity is data visualization because it gives insight into latent structure, thereby facilitating hypotheses which can then be tested using a range of other mathematical and statistical methods (Moisl 2015).

The present discussion addresses an issue in data visualization whose importance in quantitative and corpus linguistics has thus far not been sufficiently appreciated: the possibility that the data is nonlinear. Most visualization applications in these fields use linear proximity measures which ignore nonlinearity, and, if the data is significantly nonlinear, can give misleading results.

The discussion is in three main parts: the first part outlines the nature of nonlinearity in data generally and in linguistic data specifically, the second shows why nonlinearity is a problem for linear visualization methods, and the third shows how topological mapping can be used to visualize high-dimensional data in a way that takes nonlinearity into account.

1. Nonlinearity

1.1 Nonlinearity in natural processes

In natural processes there is a fundamental distinction between linear and nonlinear behavior. Linear processes have a constant proportionality between cause and effect. If a ball is kicked *x* hard and it goes *y* distance, then a 2*x* kick will appear to make it go 2*y*, a 3*x* kick 3*y*, and so on. Nonlinearity is the breakdown of such proportionality. In the case of our ball, the linear relationship increasingly breaks down as it is kicked harder and harder. Air and rolling resistance become significant factors, so that for, say, 5*x* it only goes 4.9 *y*, for 6*x* 5.7*y*, and again so on until eventually it bursts and goes hardly any distance at all. Such nonlinear effects pervade the natural world and gives rise to a wide variety of complex and often unexpected --including chaotic—behaviours (Strogatz 2000, Bertuglia & Vaio 2005).

1.2 Nonlinearity in data

Data is a description of objects from a domain of interest in terms of a set of variables such that each variable is assigned a value for each of the objects. Given *m* objects described by *n* variables, a standard representation of data for computational analysis is a matrix M in which each of the *m* rows represents a different object, each of the *n* columns represents a different variable, and the value at $M_{i,j}$ describes object *i* in terms of variable *j*, for $i = 1..m$, $j = 1..n$. The matrix thereby makes the link between the researcher's conceptualization of the domain in terms of the semantics of the variables s/he has chosen and the actual state of the world, and allows the resulting data to be taken as a representation of the domain based on empirical observation.

M is linear when the functional relationships between all its variables, that is, the values in its columns, conform to the mathematical definition of linearity. A linear function *f* is one that satisfies the following properties, where *x* and *y* are variables and *a* is a constant (Lay 2010):

- Additivity: $f(x+y) = f(x) + f(y)$ -- adding the results of f applied to x and y separately is equivalent to adding x and y and then applying f to the sum.
- Homogeneity: $f(ax) = af(x)$ -- multiplying the result of applying f to x by a constant is equivalent to multiplying x by the constant and then applying f to the result.

A function which does not satisfy these two properties is nonlinear, and so is a data matrix in which the functional relationships between two or more of its columns are nonlinear.

Matrices have a geometrical interpretation. For each row vector of M:

- The dimensionality of the vector, that is, the number of its components n, defines an n-dimensional Euclidean space.
- The sequence of n numbers comprising the vector specifies the coordinates of the vector in the space.
- The vector itself is a point at the specified coordinates

The set of row vectors in M defines a configuration of points in the n-dimensional space called the data manifold. Linear manifolds are shapes consisting of straight lines and flat planes and represent linear data, whereas nonlinear manifolds consist of curved lines and surfaces and represent nonlinear data; examples are given in Figure 1.

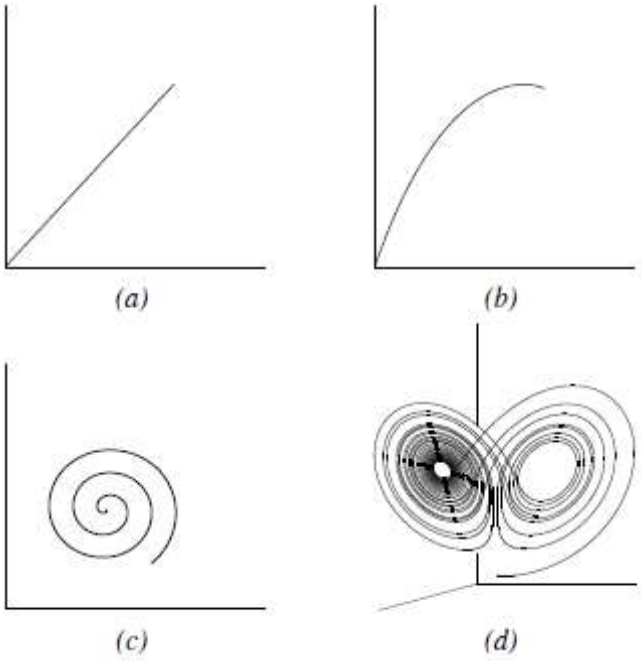
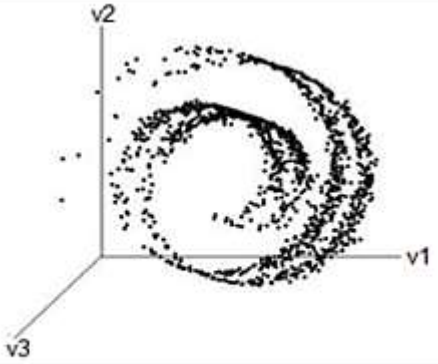


Figure 1. Linear and nonlinear manifolds in two and three dimensional space.

An unbounded range of nonlinear manifolds is possible in any dimensionality. Figure 2 gives another example of a nonlinear manifold in three-dimensional space.

	V1	V2	V3
1	5.214	7.087	6.729
2	0.421	8.426	7.896
3	6.134	5.686	6.088
...	...	...	...
1600	0.945	11.501	10.867

a: 1600 3-dimensional vectors



b: The vectors in (a) plotted in 3-dimensional space

Figure 2. Nonlinear manifold in three dimensional space.

1.3 **Nonlinearity in linguistic data**

Data abstracted from a natural process known to be linear is itself guanteed to be linear. Data abstracted from a known nonlinear process is not necessarily nonlinear, but may be. The human brain - the generator of language - is a nonlinear dynamical system that exhibits highly complex physical behaviour in which nonlinearity arises on account of latency and saturation effects in individual neuron and neuron assemblies. One must, therefore, always reckon with the possibility that data abstracted from speech or text will be nonlinear.

2. **The problem**

The problem that nonlinearity poses for cluster analysis of high-dimensional multivariate data is easily seen. A metric space M(V,*d*) is a vector space V on which a metric *d* is defined in terms of which the distance between any two points in the space can be measured. Numerous distance metrics exist (Deza & Deza 2009: chs. 17, 19). For present purposes these are divided into two types:

1. Linear metrics, where the distance between two points in a manifold is taken to be the length of the straight line joining the points, or some approximation to it, without reference to the shape of the manifold.

2. Nonlinear metrics, where the distance between the two points is the length of the shortest line joining them along the surface of the manifold and where this line can but need not be straight.

This categorization is motivated by the earlier observation that manifolds can have shapes which range from perfectly flat to various degrees of curvature. Where the manifold is flat, as in Figure 3a, linear and nonlinear measures are identical. Where it is curved, however, linear and nonlinear measurements can differ to varying degrees depending on the nature of the curvature, as shown in Figures 3b and 3c.

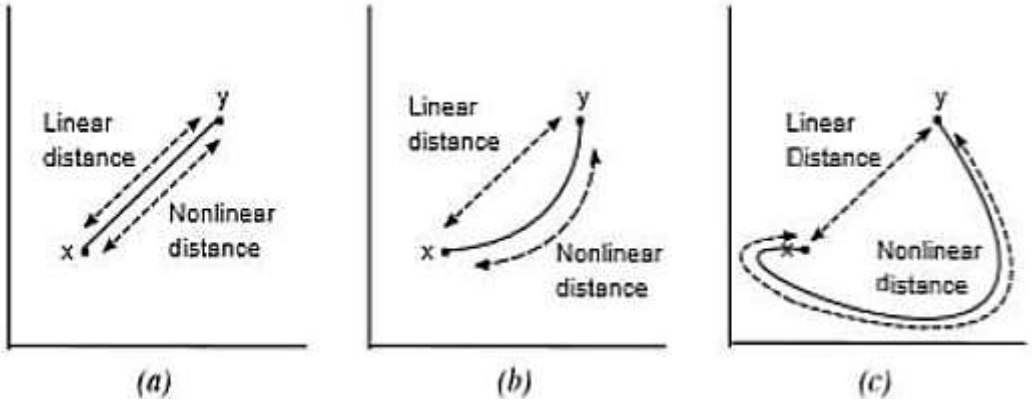


Figure 3. Linear and nonlinear distances.

For Figure 2b, the linear distance is shown in Figure 4 between two points; the nonlinear distance follows the surface of the curve, and is obviously much greater.

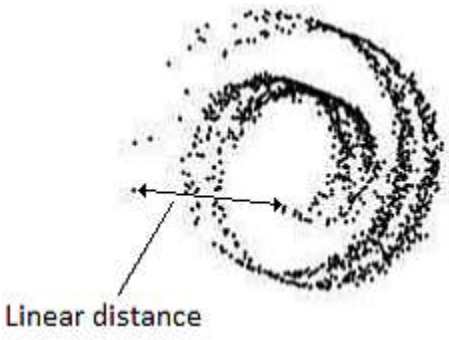


Figure 4. Linear distance between points on a nonlinear manifold.

Commonly-used visualization methods such as principal component analysis for projection into two- or three-dimensional space, or hierarchical cluster analysis using proximity measures like the Euclidean, are linear: they take no account of any curvature in the manifold, and can thereby introduce distortions into visualization results in some proportion to the degree of nonlinearity in the manifold. This is shown in Figure 4 for three-

dimensional data, but the situation extends to any dimensionality.

One way of precluding nonlinearity errors is to use a nonlinear distance measure (Moisl 2015). The alternative is to use a topological method, outlined in what follows.

3. **Topological mapping**

3.1 **Topology**

Topology (Munkres 2000, Reid & Szendroi 2005, Sutherland 2009) is an aspect of contemporary mathematics that grew out of metric space geometry. Its objects of study are manifolds, but these are studied as spaces in their own right, topological spaces, without reference to any embedding metric space and associated coordinate system. Topology would, for example, describe a manifold embedded in the metric space of Figure 5a independently both of the metric defined on the space and of the coordinates relative to which the distances among points are calculated, as in Figure 5b. Topology replaces the concept of metric and associated coordinate system with relative nearness of points to one another in the manifold as the mathematical structure defined on the underlying set; relative nearness of points is determined by a function which, for any given point *p* in the manifold, returns the set of all points within some specified proximity to *p*. But how, in the absence of a metric and a coordinate system, is the proximity characterized?

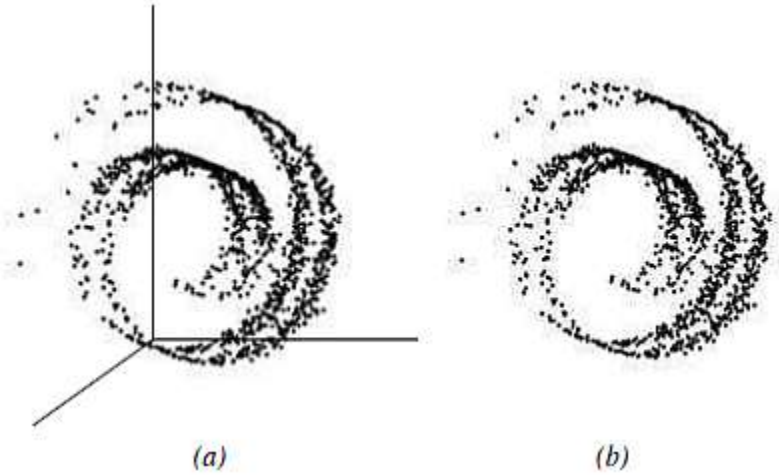


Figure 5. A manifold embedded in a three-dimensional coordinate system and as a topological object.

The answer is that topological spaces are derived from metric ones and inherit from the latter the concept of neighbourhoods. In a metric space, a subset of points which from a topological point of view constitutes a manifold can itself be partitioned into subsets of a fixed size called neighbourhoods, where the neighbourhood of a point *p* in the manifold can be defined either as the set of all points within some fixed radius *ε* from *p* or as the *k* nearest neighbours of *p* using the existing metric and coordinates; in Figure 6 a small region of the manifold from Figure 5 is magnified to exemplify these two types of neighbourhood.

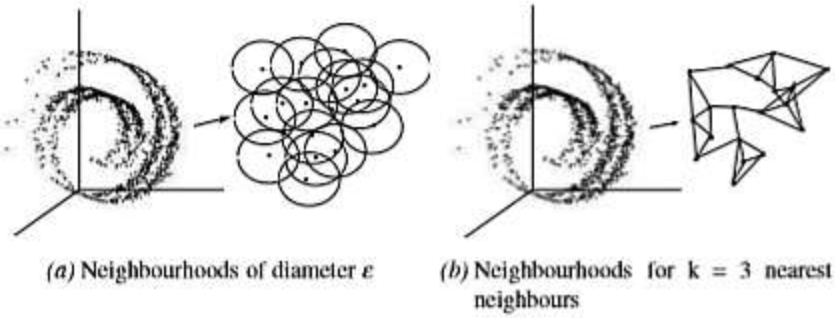


Figure 6. Neighbouthoods in a magnified fragment of a geometric object in metric space.

In Figure 6a the neighbourhood of every point is the other points within a radius of *ε*, shown as circles within the magnification rectangle; in 6b a neighbourhood of any point is the *k* nearest points irrespective of distance, shown for *k* = 3 as lines connecting each point to the three nearest to itself. Once a manifold has been partitioned into neighbourhoods and thereby transformed into a topological space, the frame of reference is discarded and only the neighbourhoods defined in terms of the metric are retained. In this way, manifolds of arbitrary shape can be conceptualized as being composed of metric subspaces; if the original

metric is Euclidean, for example, the manifold in Figure 5b can be understood as a patchwork of locally-Euclidean subspaces. Intuitively, this corresponds to regarding the curved surface of the Earth as a patchwork of flat neighbourhoods, which is how most people see it.

3.2 **Projection of topological structure into low-dimensional space**

High-dimensional manifolds can be visualized as low-dimensional ones by means of projection in which the topology of the high-dimensional manifold, that is, the neighbourhood structure, is preserved in the low-dimensional one, so that points close to one another in high dimensions are close to one another in the low-dimensional projection. This can be conceptualized as in Figure 7, where a three-dimensional manifold is projected onto a two-dimensional surface.

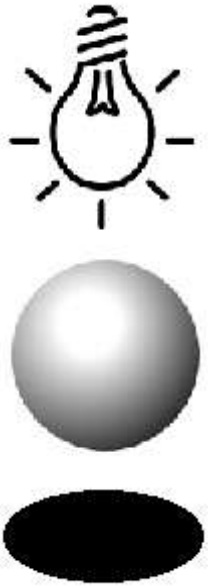


Figure 7. Projection from three to two dimensions.

3.3 **Preservation of nonlinearity**

The set of neighbourhoods which constitutes the topology of a manifold by definition follows the surface of the manifold, whatever its shape. Because a projection preserves the topology, that shape is preserved - in other words, nonlinearity is preserved in the projection.

3.4 **Example**

The aim of this section is to show how topological mapping can be used to discover structure in high-dimensional multivariate data abstracted from a multi-document corpus. It does this by using a particular topological mapping method, the self-organizing map (SOM), to infer the relative chronology of a collection of Old English, Middle English, and Early Modern English texts from spelling data abstracted from them.

3.4.1 **The text collection**

The list of texts comprising the example corpus is given in Table 1. Because the aim is methodological - to exemplify the application of a visualization method to historical linguistic data rather than to generate novel results about the history of English - this discussion assumes that the textual accuracy offered by current critical editions is unnecessary, and that readily available online texts of diverse provenances suffice. To this end, the Old English texts were downloaded from the *Sacred Text Archive* (https://www.sacred-texts.com/), the Middle English ones from the *Corpus of Middle English Prose and Verse* (https://quod.lib.umich.edu/c/cme/), and the Early Modern English ones from *Corpora of Historical English* (http://davies-linguistics.byu.edu/personal/histengcorp.htm); in a few cases more conveniently formatted texts were downloaded from other sites.

Table 1. The example corpus C.

Old English	Middle English	Early Modern English
Exodus	Sawles Warde	King James Bible
Phoenix	Henryson, Testament of Cressid	Campion, Poesie
Juliana	The Owl and the Nightingale	Milton, Paradise Lost
Elene	Malory, Morte Darthur	Bacon, Atlantis
Andreas	Gawain and the Green Knight	More, RichardIII
Genesis A	Morte Arthure	Shakespeare, Hamlet
Beowulf	KingHorn	Jonson, Alchemist
	Alliterative Morte Arthure	
	Bevis Of Hampton	
	Chaucer, Troilus	
	Langland, Piers Plowman	
	York Plays	
	Cursor Mundi	

3.4.2 **Spelling data**

Spelling is used as the basis for inference of the relative chronology of the above texts on the grounds that it reflects the phonetic, phonological, and morphological development of English over time. The variables used to represent spelling in the texts are letter pairs: for "the cat sat", the first letter pair is (t,h), the second (h,e), the third (e,<space>), and so on. All distinct pairs across the entire text collection were identified, and the number of times each occurs in each text was counted. The fragment of the resulting data matrix M in Table 2 exemplifies this.

Table 2. Fragment of the frequency matrix M abstracted from C.

	1. hw	2. we	3. fe	...	841. jm
Exodus	35	149	125	...	0
Sawles Warde	52	147	45	...	0
...	...	...	...	...	...
King James	0	42	36	...	0

M was normalized to compensate for variation in document length and truncated to the most important 100 letter pairs, yielding a new matrix M' used in the analysis that follows. Details of normalization and truncation are available in (Moisl 2015: ch. 3).

3.4.3 **The Self-Organizing Map**

The Self-Organizing Map (SOM) is a topological mapping method. It is an artificial neural network that was originally invented to model a particular kind of biological brain organization, but can also be used without reference to neurobiology as a way of visualizing high-dimensional data manifolds by projecting and displaying them in low-dimensional space. It has been extensively and successfully used for this purpose across a wide range of disciplines. Figure 8 shows the architecture of the SOM.

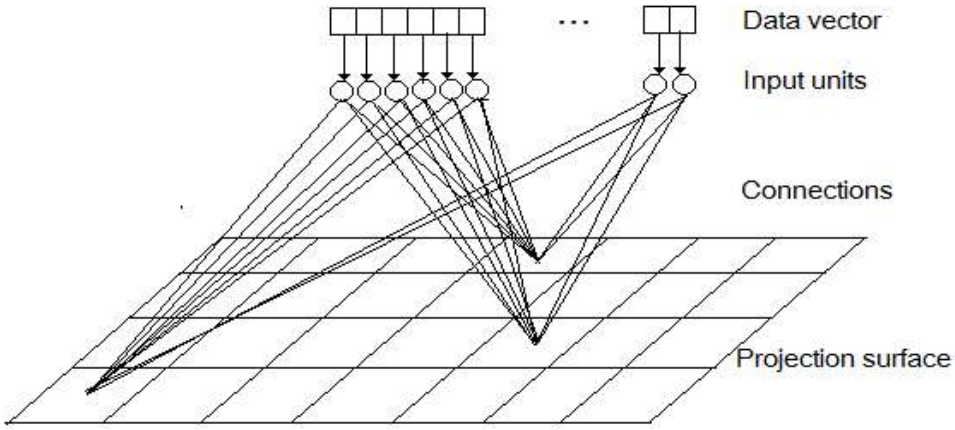


Figure 8. SOM architecture.

An *n*-dimensional data vector is loaded into the input units. These values are propagated along the connections in such a way that the vector is assigned to one of the cells on the two-dimensional projection surface; only a selection of connections is shown, and in reality every input unit is connected to every projection surface cell. Once every input vector has been projected onto the surface, the topology of the *n*-dimensional data manifold has been mapped onto the two-dimensional projection space, where it is available for visual inspection. The relative strengths of the connections linking the input units to the projection surface are fundamental to the SOM's mapping, and, as with other artificial neural network architectures, these are iteratively learned from the set of input vectors in any given application. The learning procedure is quite complex and so is not rehearsed here; details are given in the following citations. The standard work on the SOM is Kohonen (2001). Shorter accounts are Haykin (1999: ch. 9), Van Hulle (2000), Lee & Verleysen (2007: ch. 5), Izenman (2008: ch. 12.5), Xu & Wunsch (2009: ch. 5.3.3), Moisl (2015: ch.4); collections of work on the SOM are in Oja and Kaski (1999) and Allinson et al. (2001). For overviews of applications of the SOM to cluster analysis and data analysis more generally see Kohonen (2001: ch. 7) and Vesanto & Alhoniemi (2000).

An intuition for how the SOM works can be gained by looking at the biological brain structure it was originally intended to model: sensory input systems (Van Hulle 2000, Kohonen 2001: chs. 2, 4). The receptors in biological sensory systems generate very high dimensional signals which are carried by numerous nerve pathways to the brain. The retina of the human eye, for example, contains on the order of 10^8 photoreceptor neurons each of which can generate a signal in selective response to light frequency, and the number of pathways connecting the retina to the brain is on the order of 10^6 (Hubel & Wiesel 2005). At any time *t*, a specific visual stimulus *s*(*t*) to the eye is transmitted via the nerve pathways to the visual cortex, and this generates a pattern of retinal activation *a*(*t*) which is in turn transmitted to the rest of the brain for further processing. It is the response of the visual cortex to retinal stimulation which is of primary interest here. The visual cortex is essentially a two-dimensional region of neurons whose response to stimulation is spatially selective: any given retinal activation *s*(*t*) sent to it activates not the whole two-dimensional cortical surface but only a relatively small region of it, and subsequent stimuli *s*(*t*+1), *s*(*t*+2)...*s*(*t*+*n*) activate other regions whose distances from *a*(*t*) and from one another on the cortical surface are proportional to the relative similarities of the *s*(*t*)...*s*(*t*+*n*). If, say, seven activations *s*(1)...*s*(7) are input in temporal succession, and if the members of each of three sets {*a*(1), *a*(3), *a*(7)}, {*a*(2), *a*(5)},and {*a*(4), *a*(6)} are similar to one another but different from members of the other two sets, then the "cortex" is sequentially activated, and these successive activations, when superimposed as in Figure 9, show a cluster structure. This is the basis for the SOM's use as a projection method.

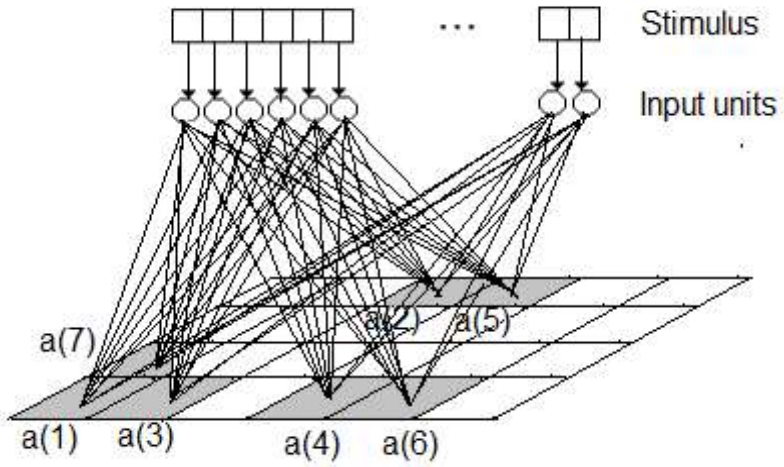


Figure 9. Selective activation of the projection surface in response to stimuli.

The mathematical model corresponding to the above physical one has three components together with operations defined on them:

- An n-dimensional input vector R, for some arbitrary n, which represents the retina.
- A $p \times q$ output matrix M which represents the sensory cortex, henceforth referred to as the lattice.
- A $p \times q \times n$ matrix C which represents the connections, where C(i,j,k) is the connection between the neuron at M(i,j) (for $i = 1... p$, $j = 1...q$), and the one at R(k) (for $k = 1...n$).

For data projection a SOM works as follows, assuming an $m \times n$ data matrix D is given. For each row vector D(i) (for $i = 1...m$) repeat the following two steps: (1) Present D(i) as input to R, and (2) Propagate the input along the connections C to selectively activate the cells of the lattice M; in mathematical terms this corresponds to the inner product of R with each of the connection vectors at C(i,j). The result of the inner product R.C(i,j) is stored at M(i,j): M(i,j) = R.C(i,j). Once all the data vectors have been processed there is a pattern of activations on the lattice M, and this pattern is the projection of the data matrix D.

There is a strong intuitive temptation to interpret the lattice activation pattern spatially, that is, to interpret any groups of adjacent, highly activated units as clusters, and the distance between and among clusters as proportional to the relative distances among data items in the high-dimensional input manifold. That temptation needs to be resisted. The SOM differs from the distance-based clustering methods in that the latter try to preserve relative distance relations among objects on the data manifold, whereas the SOM tries to preserve the manifold topology – cf. Kaski (1997), Verleysen (2003), Lee & Verleysen (2007: ch. 5). To see the implications of this, it is necessary to understand that, when it is said that a SOM preserves the topology of the input manifold on the output lattice, what is meant is that it preserves its neighbourhood structure: all the vectors in a given neighbourhood are mapped to the same lattice cell, and the vectors in the adjoining neighbourhoods are mapped to nearby cells. The result is that the vectors which are close to one another in the input manifold in the sense that they are in the same or nearby neighbourhoods will be close on the SOM output lattice. The problem, though, is this: just because active cells are close together on the SOM lattice does not necessarily mean that the vectors which map to them are topologically close in the input manifold. This apparently-paradoxical situation arises for two reasons – see discussion in, for example, Ritter et al. (1992: ch. 4) and Moisl (2015: ch.4).

1. The topology of the output manifold on the lattice to which the SOM maps the input one must be fixed in advance. In the vast majority of applications the SOM output topology is a two-dimensional plane, that is, a linear manifold, with rectangular or hexagonal neighbourhoods which are uniform across the lattice except at the edges, where they are necessarily truncated. There is no guarantee that the intrinsic dimensionality of the input manifold is as low as 2, and therefore no guarantee that the output topology will be able to represent the input manifold well. In theory, the SOM is not limited to two-dimensional linear topology, and various developments of it propose other ones, but where the standard one is used some degree of distortion in the lattice's representation must be expected – cf. Verleysen (2003), Lee & Verleysen (2007: ch. 5); the projection is optimal when the dimensionality of the lattice is equal to the intrinsic dimensionality of the data.
2. The dynamics of SOM training do not at any stage use global distance measures. The mapping from input to output space depends entirely on local neighbourhood adjacency. As such, the SOM cannot

be expected consistently to preserve proportionalities of distance between individual vectors and vector neighbourhoods. As a result, the SOM may squeeze its representation of the input topology into the lattice in such a way that units associated with neighbourhoods which are far apart on the input manifold may nevertheless be spatially close to one another on the lattice.

In view of (1) and (2), how can a SOM lattice be interpreted so as to differentiate cells which are spatially close because they are topologically adjacent in the input manifold, and cells which are spatially close on account of the above distorting factors but topologically more or less distant? The answer is that it cannot be done reliably by visual inspection alone; interpretation of a SOM lattice by visual inspection is doubly unreliable – a subjective interpretation of an ambiguous data representation. This is a well known problem with SOMs (Kohonen 2001: 165), and a variety of ways of achieving the required differentiation exist. A frequently-used one is the U-matrix (Ultsch & Siemon 1990, Ultsch 2003), which graphically shows the boundaries between areas of the lattice which are genuinely close topologically by means of peaks and troughs, or by colour coding, or by a combination of the two. The U-matrix is used here, and exemplified below.

3.4.4 **Result**

The result of the SOM projection of M' is shown in Figure 10.

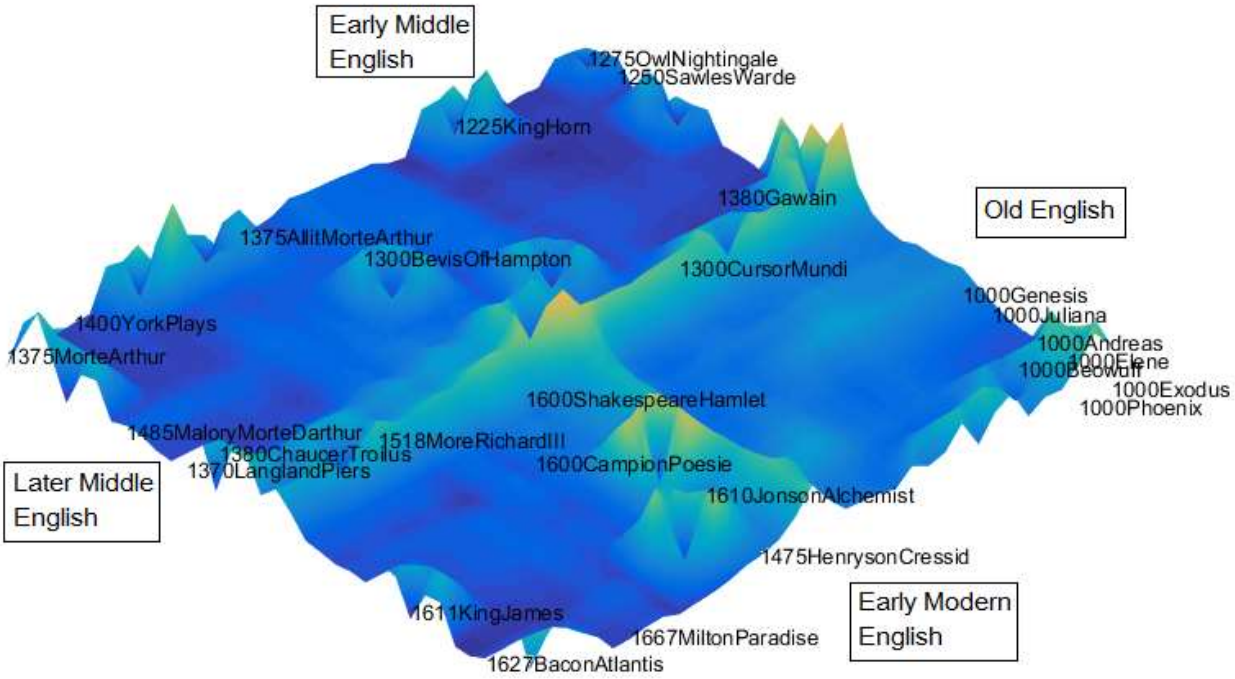


Figure 10. U-matrix representation of the two-dimensional SOM projection of the 100-dimensional frequency matrix M' abstracted from the example corpus C.

The lattice shows four main topological areas separated by peaks, with sub-regions within each. The main regions correspond to the four chronological stages of English; text-labels are anchored on the left, that is, '1600ShakespeareHamlet' is, for example, located at the initial '1' on the map. As can be seen, the projection from the 100-dimensional data matrix onto a two-dimensional surface has clustered the texts in accordance with what is independently known of their dates.

Conclusion

Topological mapping is widely applicable to data abstracted from multi-text historical linguistic corpora:

- Where the characteristics of the corpus language are well known, as for the well-studied European languages, topological mapping can be used to bestow the fundamental scientific characteristics of objectivity and replicability on them.
- Where they are less well known, as for corpora in non-European languages, it can be used to identify objective, replicable geographical and relative chronological distributions.

References

Allinson, Nigel, Hujun Yin, Lesley Allinson, Jon Slack (eds.). 2001. *Advances in Self-Organising Maps.* Berlin: Springer.

Bertuglia, Cristoforo & Franco Vaio. 2005. *Nonlinearity, Chaos, and Complexity: The Dynamics of Natural and Social Systems.* Oxford: Oxford University Press.

Deza, Michel & Elena Deza. 2009. *Encyclopedia of Distances.* Berlin: Springer.

Haykin, Simon. 1999. *Neural Networks. A Comprehensive Foundation.* Upper Saddle Hall NJ: Prentice Hall International.

Hubel, David & Torsten Wiesel. 2005. *Brain and visual perception: The story of a 25-year collaboration.* Oxford: Oxford University Press.

Izenman, Alan. 2008. *Modern Multivariate Statistical Techniques. Regression, Classification, and Manifold Learning.* Berlin: Springer.

Kaski, Samuel. 1997. *Data exploration using self-organizing maps.* Helsinki: Acta Polytechnica Scandinavica. Mathematics, Computing, and Management in Engineering Series 82.

Kohonen, Teuvo. 2001. *Self Organizing Maps.* 3rd ed. Berlin: Springer.

Lay, David. 2010. *Linear Algebra and its Applications.* 4th ed. London: Pearson Education International.

Lee, John. 2010. *Introduction to Topological Manifolds*, 2nd ed. Berlin: Springer.

Lee, John & Michel Verleysen. 2007. *Nonlinear Dimensionality Reduction.* Berlin: Springer.

Moisl, Hermann. 2015. *Cluster Analysis for Corpus Linguistics.* Berlin*:* de Gruyter.

Munkres, James. 2000. *Topology.* 2nd ed. London: Pearson Education International.

Oja, Erkki & Samuel Kaski. 1999. *Kohonen Maps. Amsterdam:* Elsevier.

Reid, Miles & Balasz Szendroi. 2005. *Geometry and Toplogy.* Cambridge: Cambridge University Press.

Ritter, Helge, Thomas Martinetz, Klaus Schulten. 1992. *Neural Computation and Self -Organizing Maps.* Boston: Addison-Wesley.

Strogatz, Steven. 2000 *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry and Engineering.* New York: Perseus Books.

Sutherland, Wilson. 2009. *Introduction to Metric and Topological Spaces.* 2nd ed. Oxford: Oxford University Press.

Ultsch, Alfred. 2003. *U∗-Matrix: a tool to visualize cluster in high-dimensional data.* Technical report 36, Department of Computer Science, University of Marburg.

Ultsch, Alfred & Peter Siemon. 1990. Kohonen's self-organizing feature maps for exploratory data analysis. *Proceedings of the International Neural Network Conference, INNC '90.* 305–8.

Van Hulle, Marc. 2000. *Faithful Representations and Topographic Maps.* Hoboken NJ: John Wiley and Sons.

Verleysen, Michel. 2003. Learning high-dimensional data. In Sergey Ablameyko et al. (eds.) *Limitations and future trends in neural computation,* 141–162*.* Amsterdam: IOS Press.

Vesanto, Juha & Esa Alhoniemi. 2000. Clustering of the self-organizing map. *IEEE Transactions on Neural Networks* 11. 586–600.

Xu, Rui & Don Wunsch. 2005. *Clustering.* Hoboken NJ: Wiley.