Visual Clustering of High-dimensional Data by Navigating Low-dimensional Spaces

Oldford, R. Wayne
University of Waterloo, Department of Statistics & Actuarial Science
200 University Avenue West
Waterloo, N2L 3G1, Canada
E-mail: rwoldford@uwaterloo.ca

Waddell, Adrian
University of Waterloo, Department of Statistics & Actuarial Science
200 University Avenue West
Waterloo, N2L 3G1, Canada
E-mail: arwaddell@uwaterloo.ca

Introduction

The purpose of cluster analysis is to conjecture plausible differences in kind amongst a given collection of instances. This is also what our human visual system excels at; it has evolved to facilitate quick and considered detection of the visually like and unlike through a wide variety of cues – e.g. location and relative proximity, movement, shape, colour, texture, and matching against predetermined patterns. Consequently, visualization is a natural and powerful resource for cluster analysis; it is especially valuable in identifying unanticipated structure.

Unfortunately, the same evolutionary path has meant our visual system is poorly equipped to be of much help in identifying high dimensional structure. And most data these days are of high, and ever increasing, dimensionality. Consequently, automated methods of pattern recognition and cluster analysis have seen increasing recent use and development; even so, intuition as to what constitutes a "cluster" in high dimensions remains largely, though by no means exclusively, based on our experience with our own visual perception – e.g. near neighbours, k-means, local density modes, etc.

Automated and purely visual methods for cluster detection are largely complementary in the circumstances in which they have most value. Automated methods may be routinely applied to data of many more dimensions than three, where our visual experience and ability necessarily end. Unfortunately, to do so, automated methods rely (at least implicitly) on determining pre-defined patterns in data configurations and so different methods can produce different clusterings. Moreover, in two or three dimensions, data configurations are routinely constructed on which a favourite automatic method will fail, but for which the human visual system excels.

The point of visual clustering is to use interactive data visualization tools in concert with automated methods so as to take best advantage of both. Following Hurley and Oldford (2011), we do this by introducing a graph structure, called a *navigational graph*, or *navGraph*, whose vertices represent a unique pair of variates. At each vertex, then, sits a two dimensional space defined by the variates there. When we add only edges between vertices which share a variate, the edge itself represents a three dimensional space formed by variates of the union of the variate pairs at each node. Such a *navGraph* is called a *3d-transition graph* in Hurley and Oldford (2011). Visualization methods will be restricted to these 2- and 3-d spaces; whereas automated methods could be applied to any number of variates, and hence dimensions, from the data itself.

For example, consider the graph shown at the centre of Figure 1. Clockwise from the large

A* navGraph *for selected pairs of variates from the Olive data.

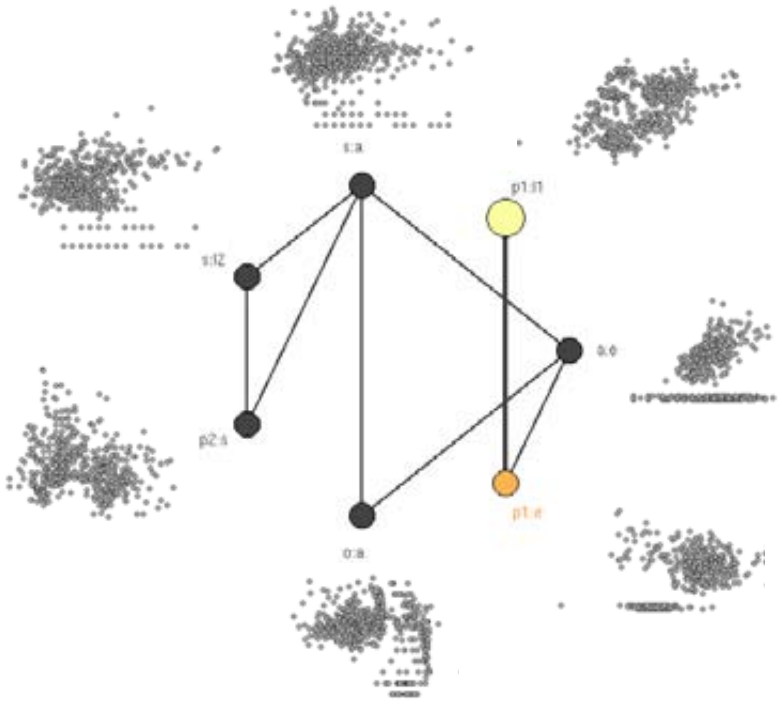


Figure 1: A 3d-transition graph for selected pairs of variates from the Olive data. The "bullet" is the large (yellow) vertex at `p1:l1`. Vertices connected to the bullet vertex are coloured differently (orange) from those which are not (black). Edges connect vertices which share a variate – hence the transition/edge is "3d". Outside each node, the point cloud of the 572 data points is shown for that variate pair.

(yellow) vertex or "bullet", the vertices are labelled by pairs of variates: `p1:l1`, `a:e`, `p1:e`, `o:a`, `p2:s`, `s:l2`, and `s:a`. The colon ":" separates the variate names, in this case "shortnames" for the longer variate names given in Table 1. Each vertex represents a pair of variates and each edge connects

Table of Variate names for the Olive data.

Short name	`a`	`e`	`l1`	`l2`	`o`	`p1`	`p2`	`s`
Fatty Acid	Arachidic	Eicosenoic	Linoleic	Linolenic	Oleic	Palmitic	Palmitoleic	Stearic

Table 1: Variates are percentage of fatty acid in each of 572 olive oils from various regions of Italy. See Forina (1983) et al for details.

vertices whose variate pair share a variate. This means that the vertices represent two dimensional data spaces, and the edges three dimensional spaces – ideal number of dimensions for visualizing the data. A walk on this graph should correspond to 2d data visualizations at each node (e.g. the 2d-point clouds which appear at each vertex of the above graph) and a 3d data visualization on the edges joining vertices.

In what follows, we use this data set to illustrate a new interactive scatterplot tool for visual clustering which, as proposed in Hurley and Oldford (2011), uses graph structures like that shown

in Figure 1 to navigate high dimensional data through low dimensional subspaces – any walk on this navgraph is a low dimensional trajectory through the higher dimensional space. The software illustrated is from our `R` package called `RnavGraph` (Oldford and Waddell, 2011) and is freely available from the Comprehensive R Archive Network (http://cran.r-project.org/).

The next section shows features of `RnavGraph` which are useful to visually cluster the Olive data. This is followed by a discussion of the size of the graph may be usefully reduced and an example of navGraph on image data.

Visually clustering the Olive data using `RnavGraph`

As can be seen from the point clouds shown in Figure 1, there is considerable structure in this data set and we might expect to separate the data points into several groups. From the cloud at the bullet, we see that there seem to be several, perhaps five to seven groups that we might distinguish based simply on ellipsoidal regions of high density. Proceeding clockwise, the next two plots suggest that the variate `e` (Eicosenoic acid) seems to dramatically separate the data into two major regions. The bottommost plot of Figure 1 also indicates at least two groups, a large cloud at the left and a sparser cloud at the right. Closer inspection, however, suggests the right most cloud might be split into one or more horizontal lines and a strong vertical grouping at the right. Continuing clockwise we see again the possibility of at least two large groups. The last two point clouds reinforce the idea of some points separating into fairly regularly spaced locations and others into a more contiguously dense region. It is hard to imagine an automated method which would successfully identify all of these various groupings which so readily present themselves to the eye.

It is difficult to tell whether these groupings agree or disagree without some means of visually connecting one set of point clouds to another. Here is where the edges provide value. In `RnavGraph`, we can select the bullet on the navigational graph and drag it towards any other vertex, provided it has an edge connected to the bullet's vertex. As Figure 2 shows, dragging the bullet along an edge

A 3d transition.

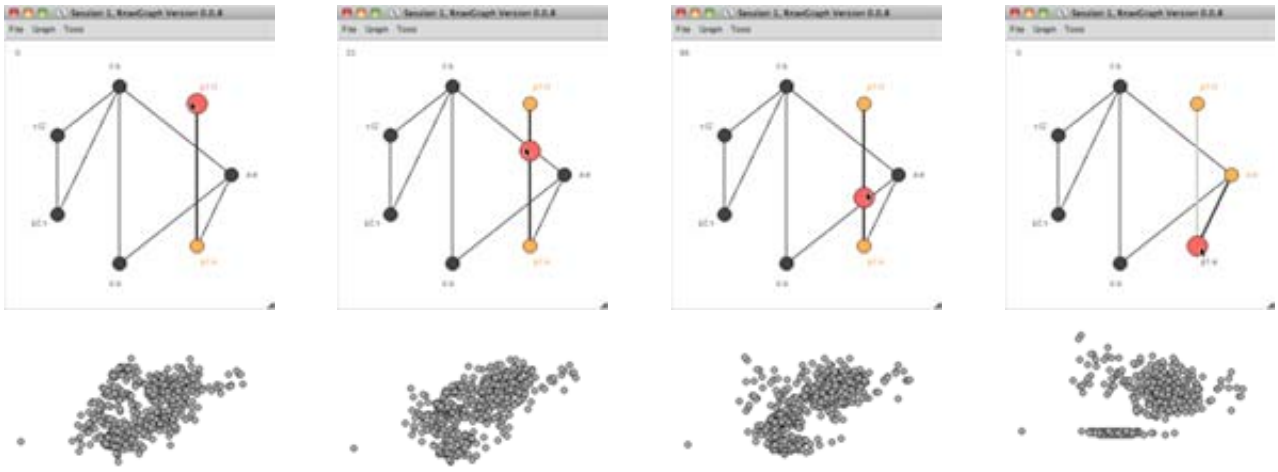
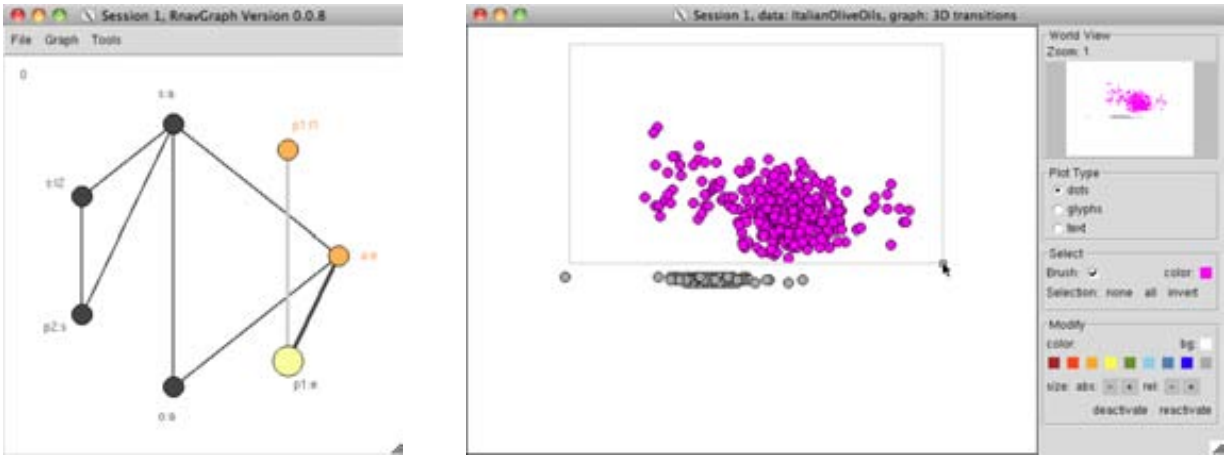


Figure 2: The same navigation graph is shown with four different bullet positions. As the bullet is dragged from its starting position at `p1:l1` on the left down to its final position at `p1:e` on the right, the corresponding point cloud is rotated in 3d to rotate the `l1` axis into the `e` axis. Note how the colours of the vertices and the bullet change as the bullet is selected and moves. Note also that the visited edge also changes colour.

effects a visualization of the data in the three dimensions corresponding to that edge – one scatterplot is dynamically rotated into the other, through this 3d space, as the bullet moves. The points are visually connected through motion and the bullet can be stopped at any point, or moved back and forth – whatever the analyst chooses to best effect the pattern recognition.

The `RnavGraph` interface actually has two major pieces – the navigation graph, or navGraph, and an interactive scatterplot. The two displays are shown side by side in Figure 3 as they might

RnavGraph session windows.



(a) The navGraph window. (b) The interactive scatterplot window.

Figure 3: On right, the brush has been used to highlight the top group in the point cloud.

appear on a data analyst's screen. The positions of the points in the scatterplot display are determined by the position of the bullet in the navGraph display. Here we show that the analyst has selected a brushing operation and highlighted all points in the top group by sweeping out a rectangular area.

Once selected, any operation in the "Modify" section of the scatterplot display may be applied to the selected points. In particular, the selected points may be "deactivated", causing them to disappear from all views, so as to allow the analyst to focus on the remaining data. Selecting "reactivate" from the "Modify" section will return all deactivated points to view.

Having deactivated the points shown selected in Figure 3, in Figure 4 we see the result of a number of steps taken by the analyst on the points that remain (i.e. on the non-highlighted grey points of Figure 3).

As the bullet location on the navGraph of Figure 4 indicates, the visualization has returned to the `p1:l1` two dimensional space to view the remaining active points. With fewer data points, the analyst has also rescaled the data by zooming in (effected through scrolling while the mouse is overtop the point cloud). In the "World view" (top right corner of the scatterplot window) we see the relative size of the scaling numerically (next to "Zoom") and, more immediately, visually as the smaller white rectangle. This rectangle also shows where in the entire (2d) data cloud, the displayed data is located; grabbing this tiny rectangle and moving it around allows the analyst to dynamically pan over any region of the data. Finally, the analyst has selected and coloured three different subsets of the data in this view. Two small groups – one green, one blue – are distinguished at the top of the point cloud and a singleton point at the left of the point cloud has been coloured brown. The other points remain coloured grey. These colourings indicate at least a tentative grouping within the active group

Focus on the remaining active points.

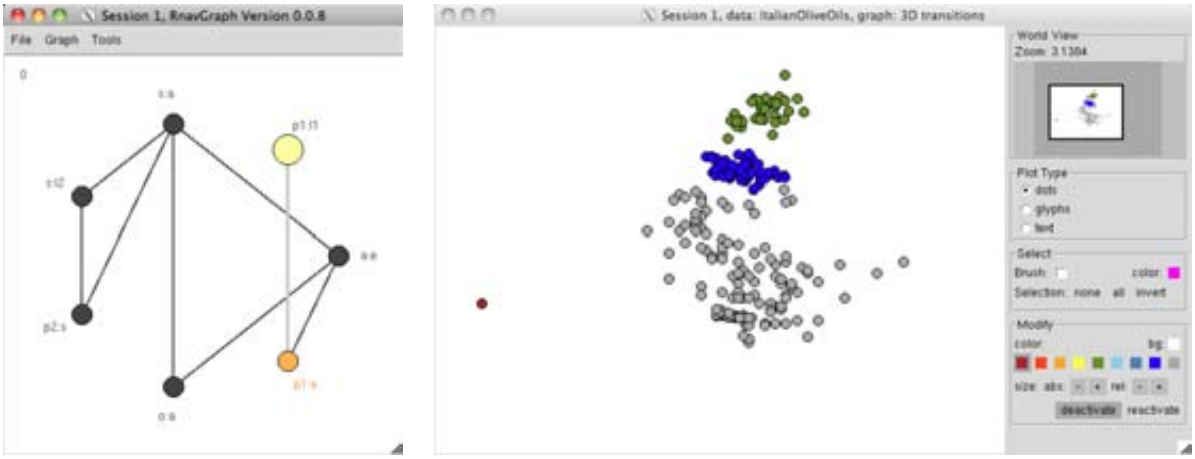


Figure 4: On right, deactivated points are removed, the data zoomed, and points coloured.

Figure 5 shows the analysis a little further on. As indicated by the lighter grey edges, the

Old groups reinforced, new groups appear.

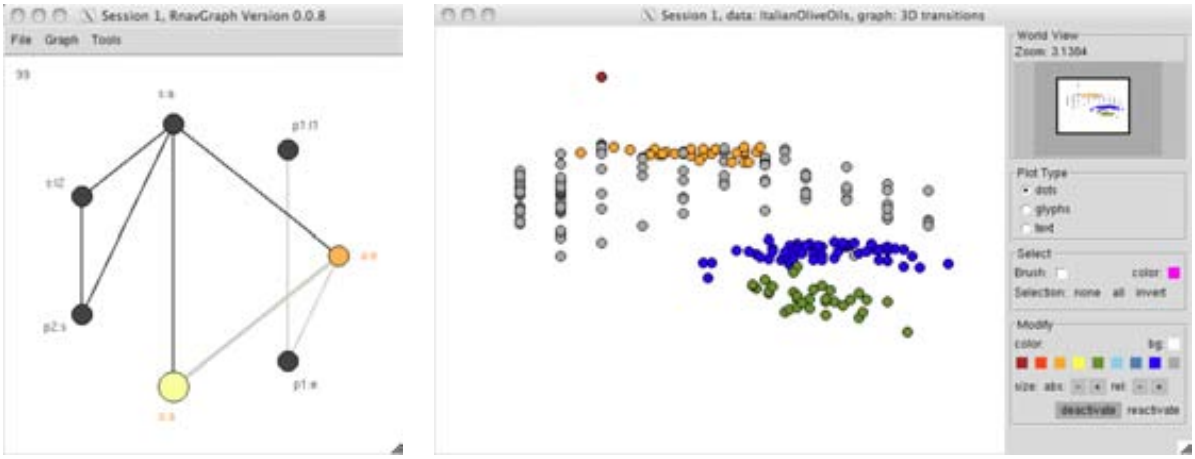


Figure 5: Same subset as before but now viewed in the `o:a` space.

navGraph has now been walked from `p1:l1` to `p1:e`, then to `a:e`, and from there finally to `o:a` resulting in the point cloud display of Figure 5. Each step of this walk is along a 3d-transition, the resulting display being a rigid rotation of the data through that 3d-space; the walk becomes a sequence of visually natural rigid rotations from one 2d space to another. All the while the smooth rotations allow the analyst to follow any group of points by their movement.

In the point cloud of Figure 5, we see that the groups identified earlier also appear to separate in this display, thus reinforcing the earlier choice. Moreover, at least two more groups have appeared – the horizontal group at the top of this point cloud (now coloured orange) and the remaining grey points which seem to have a coarser granularity of measurement. Because of this regular spacing, Occam's razor might suggest, at this point, to treat this latter bunch of points as a single group, rather than as possibly ten separate vertical groups. Of course, as the analysis progressed yet further, a different choice could yet be made.

The top (orange coloured) group identified in Figure 5 was selected largely on the basis of being

horizontal; some points were not selected to be orange because they aligned with the regular spacing of the remaining grey points. This could be investigated more closely by zooming in further on this region of the plot. Figure 6 shows a smaller region of the `o:a` space and uses star glyphs as described

Zooming in more and replacing dots with star glyphs.

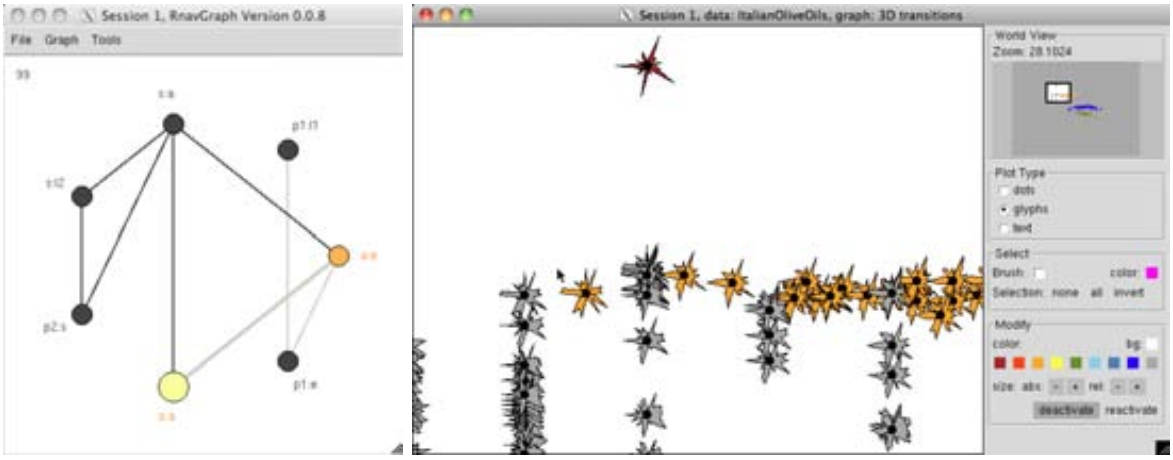


Figure 6: Closer examination of the `o:a` space, with dots replaced by glyphs.

in Hurley and Oldford (2010). Colour is preserved so that the previously proposed groupings are not lost. For each point, these star glyphs show variate values on separate radial axes, with axes repeated to ensure that every pair of variates occur next to one another somewhere in the glyph (in fact, those shown here are the Hamiltonian decomposition arrangements of Hurley and Oldford, 2010). This means that we not only have the spatial positions, shown in Figure 6, with which to group points, but we can also compare nearby points on all other variate values simultaneously, simply by comparing the shapes of the glyphs.

For example, the point previously identified as a possible outlier (and coloured brown) in Figure 4, is less certainly an outlier in Figure 5. In the latter, it lines up along one of the grey columns of Figure 5 and might reasonably belong to the same group. However, when one looks at its glyph in Figure 6, it is clear that this point is very different from its potential cluster mates and probably merits separate investigation.

More subtle comparisons can be made based on the shape of glyphs of at least near neighbours. An example is the grey glyph to the left of the cursor arrow shown in Figure 6. By spatial location in this plot, it aligns with the granularity of the gray group. However, closer examination of this grey glyph in comparison with those below it, and with the orange glyph to the right of the arrow cursor, suggests that it belongs more to the orange horizontal group than to the grey vertical group. Similarly, the grey group itself shows considerable variability in glyph shape, some having rather large relatively convex segments on their right.

`RnavGraph` also allows the glyphs to be selected and temporarily placed anywhere on the display to facilitate comparison of, and grouping by, glyph shape. One might, for example, move apart points which have occluded another. The top of the grey column of points, to the right of the arrow cursor in Figure 6 have been moved apart in Figure 7 (a). The top two grey glyphs and the left most grey glyph more clearly belong with their orange neighbour at the far left; the remaining grey glyphs are not as strongly similar in shape to these. The "true" grouping of these olive oils by geographic region is shown in Figure 7 (b).

Moving points to avoid occlusion.

(a) Glyphs moved apart (b) True geographic regions/areas

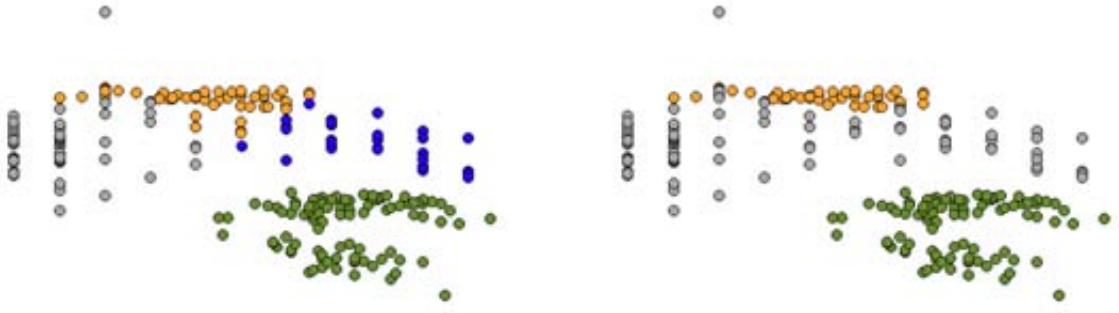
Figure 7: Closer examination of the `o:a` space, with dots replaced by glyphs.

If the geographic regions are the "true" cluster structure, we seem to be able to recover much of this structure in `RnavGraph` simply through spatial structure of the data in low dimensions, occasionally supplemented by glyphs that summarize many dimensions by 2d-shape. The points selected in Figure 3 (and later deactivated) are all from the "South" region of Italy; the remaining points are from either the "North" or "Sardina". In Figure 4, two more groups were identified; these were both from the "Sardinia" region – one from the "Coastal Sardinia" area (green), the other from the "Inland Sardinia" (blue). The horizontal group (orange) group of Figure 5 is from the "North" region, "Umbria" area; the remaining grey points are from the "North" and mainly from either the "East-" or "West-Liguria".

From this limited example, it is clear that purely visual clustering can be very powerful and allows the analyst to decide on the spot which features of the data configuration stand out as determining groups – spatial density in different dimensions, measurement granularity, point cloud shape, simultaneous comparison of all variates on a small number of points (glyphs), and so on. Though these features might have been unanticipated by the analyst, they are readily seen in these displays and so can be immediately used to distinguish groups.

One might also choose an automated method and use colour to distinguish the groups selected by that method. Figure 8 shows results for two different, and popular, automated clustering methods;

Start with groups given from some automated method.



(a) k-means: `kmeans(data,k=9)` (b) Model based clustering: `Mclust(data,1:20)`

Figure 8: Closer examination of the `o:a` space, with dots replaced by glyphs.

for comparison only the same points as in Figure 5 are shown. In Figure 8 (a), k-means is used with

$k = 9$; there are known to be nine different geographic regions, so this choice is a bit of a cheat. In Figure 8 (b), model-based clustering (Fraley and Raftery, 1999) is used and up to twenty different clusters are considered. Surprisingly, neither method separates the "Inland" from the "Coastal" areas of Sardinia (green groups, bottom right of each plot). Both attempt to separate the top horizontal group (orange, "Umbria"), with model based doing a better job than k-means; k-means, however, attempts to separate "West-Liguria" (grey, left vertical) from "East-Liguria" (right, blue) which is ignored by model-based clustering. Neither method points out the outlying point, which had been identified in the purely visual analysis.

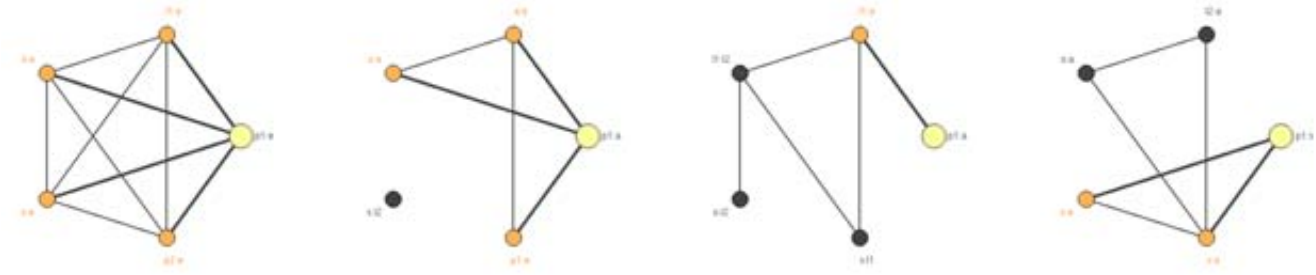
Different automated methods are predisposed to find different structures. As such, they are perhaps best used as starting points for a cluster analysis. Visualization tools, as demonstrated above, could then be used as a quality assessment of the proposed clusters, with the freedom to change proposed clusters and to introduce new clusters as the analyst sees fit. Better still would be to combine the results of a variety of automated methods from the start. Oldford and Zhou (2011) provide a formal framework and automated mechanism for such "ensemble" methods.

Graph construction

A serious challenge is to determine the low-dimensional spaces worth visiting. For p variables, there are $\binom{p}{2}$ possible nodes in a navGraph. Hurley and Oldford (2011) describe a variety of methods for construction of graphs with a small number of nodes and edges. All such methods are available in the `RnavGraph` package and any 3d- or 4d-transition graph can be viewed through the `navGraph(···)` function, with an unlimited variety of visualizations beyond point clouds (see Waddell and Oldford, 2011).

Experience to date suggest that scagnostic measures (Wilkinson et al, 2005, and available as the `scagnostics` package in `R`) are particularly valuable in identifying subspaces with interesting data structures. `RnavGraph`'s `scagNav(···)` function produces navGraphs whose vertices have variates scoring highest on any combination of these scatterplot diagnostics. Figure 9 shows several such

Scagnostic based navGraphs



(a) Clumpy (b) Not Convex (c) Striated (d) Sparse

Figure 9: Different scagnostic measures are tailored to different two dimensional data configurations. Here the 3d-transition navGraphs are shown for the top 15% of scatterplots in the Olive data for each of the named scagnostic measures. A total of 9×2 such measures (and their absence), and any combination thereof, are available from `scagNav`.

graphs for the Olive data. Each of these graphs could be walked, looking for the specified structure (or absence of structure). Note that for this data set, there are $\binom{8}{2} = 28$ possible 2d-scatterplots/vertices to consider. Scagnostics allow this to be reduced to the 7 considered in Figure 1.

While scagnostics, and other measures, may be calculated over all pairs of variates to determine the vertices of an interesting navGraph, the number of pairs to examine grows as p^2 for p variates. For very high dimensions, when the context (see Hurley and Oldford, 2011) does not naturally produce a graph with small numbers of vertices and/or edges, as with other methods some dimensionality reduction should be pursued before building the navGraph.

Figure 10 is an example of a data set of images taken from a movie Brendan Frey recorded of

Image data

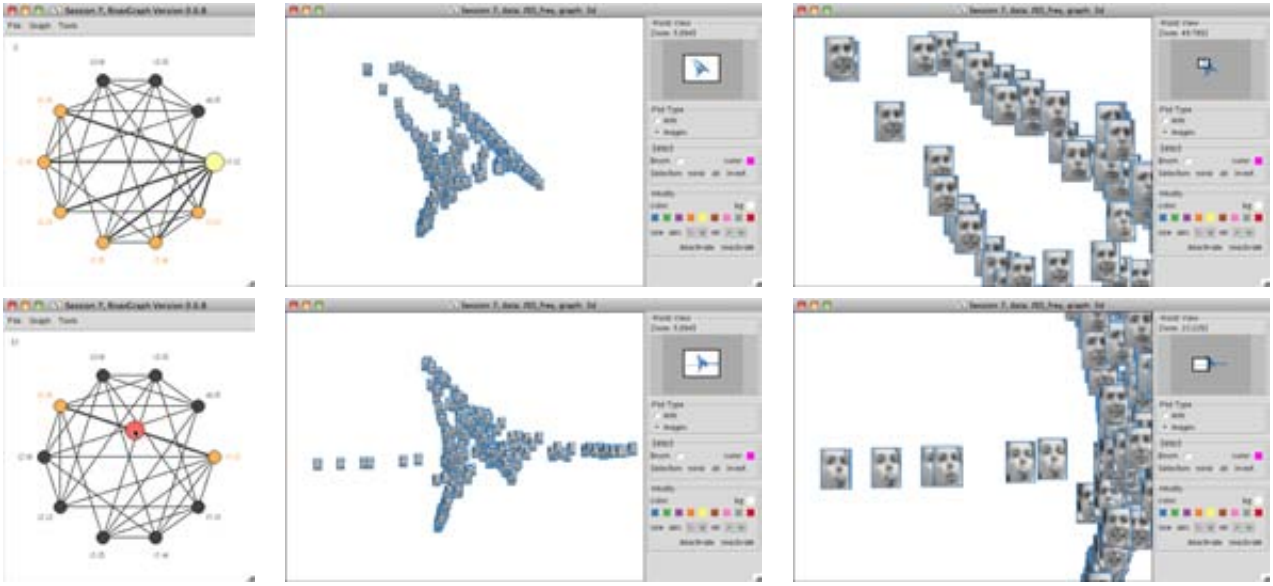


Figure 10: 560-dimensional images nonlinearly embedded in a five dimensional space using local linear embedding (Roweis and Saul, 2000, $knn = 12$). Each row shows the navGraph of all 3d-transitions, the corresponding scatterplot of images, and a zoom in on a particular region of the 2d-subspace (shown in the "world view" of the plot). As indicated by the bullet position, in the corresponding navGraph, the first row shows the configuration in the first two LLE dimensions (`i1:i2`), the second row partway along a 3d-transition from `i1:i2` to (`i2:i5`).

himself making faces. Each image is an array of 28×20 greyscale pixels; column concatenated, each image is a point in $p = 560$ dimensions. Local linear embedding (Roweis and Saul, 2000) was used to reduce the dimensions from 560 to only 5 and the navGraph for all $\binom{5}{2} = 10$ variate pairs constructed.

Clearly, there is considerable structure in this data and it is not restricted to the first two dimensions. By using a navGraph to explore the reduced dimension set of variates, the target number of dimensions can be considerably larger than usual, e.g. 10 or 20. Indeed, multiple dimension reduction methods (e.g. ISOMAP, Tenenbaum et al, 2000) might be used in concert. All methods used for the Olive data could be used on the reduced dimension data set, including scagnostics to further reduce the complexity of the navGraph. The only concern, of course, with non-linear methods is whether the structure revealed in the reduced space is a characteristic of the data and not an artefact of the method.

REFERENCES (RÉFERENCES)

Forina, M., Armanino, C., Lanteri, S. and E. Tiscornia. (1983). "Classification of olive oils from their fatty acid composition", in *Food Research and Data Analysis* (eds. H. Martens and H. Russwurm Jr.), Applied Science

Publishers, London, 189-214.
Fraley, C. and Raftery, A. E. (1999), MCLUST: Software for model-based cluster analysis, *Journal of Classification*, 16, 297306.
Hurley, C. B., and R. W. Oldford. (2010). Pairwise Display of High-Dimensional Information via Eulerian Tours and Hamiltonian Decompositions, *Journal of Computational and Graphical Statistics* 19, no. 4, 861-886.
Hurley, C. B., and R. W. Oldford. (2011). Graphs as navigational Infrastructure for high dimensional data spaces, Computational Statistics (Online First February 2011).
Oldford, R. W. and A.R. Waddell (2011). "The `RnavGraph` package", CRAN repository, `http://cran.r-project.org`.
Oldford, R. W. and W. Zhou (2011). "Tree reduced ensemble clustering via a graph algebraic framework", (submitted for publication, March 30, 2011), 17 pages.
Roweis, S. T., and L. K. Saul (2000). "Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science* 290, no. 5500: 2323-2326.
Tenenbaum, J. B., V. de Silva, and J. C. Langford (2000). "A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science 290*, no. 5500: 2319-2323.
Waddell, A.R. and R.W. Oldford (2011). "RnavGraph: A visualization tool for navigating through high-dimensional data", *58th Congress of the International Statistical Institute*, *Invited Paper Session 117*, Dublin, Ireland.
Wilkinson, L., A. Anand, and R. Grossman. (2005). Graph-theoretic scagnostics, *Proceedings - IEEE Symposium on Information Visualization*: 157-164.

RÉSUMÉ (ABSTRACT)

The structure of a set of high dimensional data objects (e.g. images, documents, molecules, genetic expressions, etc.) is notoriously difficult to visualize. In contrast, lower dimensional structure (esp. 3 or fewer dimensions) is natural to us and easy to visualize. A not unreasonable approach, then, is to explore one low dimensional visualization after another in the hope that, together, these will shed light on the higher dimensional structure. A familiar example is the parallel coordinate plot, where a sequence of one-dimensional projections are connected to provide insight into the structure of a high dimensional data set.

In this paper, we describe the graph theoretic structure, recently proposed in Hurley and Oldford (2011, Comp. Stat.), that represents low-dimensional spaces as graph nodes and transitions between spaces as edges. Of interest, are walks along these graphs that reveal meaningful structure. If the nodes are one-dimensional, a walk corresponds to a parallel coordinate plot; if they are two dimensional and edges exist, say, only between 2d spaces which share a variate, then the walk could be represented dynamically as a series of scatterplots, one transitioning into the next via a 3d rigid transformation.

We show how these graphs can be used to dynamically explore high dimensional data to visually reveal cluster structure. ***RnavGraph*** *is the tool we have developed for that purpose. We show how this visualization tool can be used for visual cluster analysis – in place of, or in concert with, automated methods. This demonstrated by a quick analysis of the "Olive data" containing eight measurements of fatty acid concentrations on 572 Italian olive oils.*

These graphs, unfortunately, grow in size with the square of the number of dimensions. Fortunately, there are numerous means for constructing only the more interesting regions of each graph. Some restrictions are imposed by the statistical context, others by empirical measures on the data itself. Of the latter, scatterplot diagnostics (scagnostics) are especially valuable. When the objective is to visually cluster the data, nearest neighbour based dimension reduction methods are particularly effective in conjunction with appropriate scagnostics. The "Frey image" data is used to demonstrate how images can be analysed using ***RnavGraph****.*