STATISTICS IN MEDICINE
*Statist. Med.* 2003; **22**:1383–1397 (DOI: 10.1002/sim.1502)

# Visual data mining

Edward J. Wegman*,†

*Center for Computational Statistics, George Mason University, Fairfax, VA 22030-4444, U.S.A.*

SUMMARY

Data mining strategies are usually applied to opportunistically collected data and frequently focus on the discovery of structure such as clusters, bumps, trends, periodicities, associations and correlations, quantization and granularity, and other structures for which a visual data analysis is very appropriate and quite likely to yield insight. However, data mining strategies are often applied to massive data sets where visualization may not be very successful because of the limits of both screen resolution, human visual system resolution as well as the limits of available computational resources. In this paper I suggest some strategies for overcoming such limitations and illustrate visual data mining with some examples of successful attacks on high-dimensional and large data sets. Copyright © 2003 John Wiley & Sons, Ltd.

KEY WORDS: parallel co-ordinates; grand tour; saturation brushing; knowledge discovery; EDA

## 1. INTRODUCTION

Statisticians have, for more than 30 years, recognized exploratory data analysis (EDA) as a important precursor to a confirmatory data analysis. In the original EDA framework, both visual and analytical tools were used to explore data and confirm that the data conformed to the assumptions underlying the confirmatory statistical methods employed to analyse the data. With the advent of high performance personal computing, more aggressive exploration of data has come into vogue. No longer was EDA simply used to verify underlying assumptions, it also was used to search for unanticipated structure. Within the last decade, computer scientists interested in databases have also come to the conclusion that a more powerful form of data analysis could be used to exploit data residing in databases. Their work has generally been formulated under the rubric of *knowledge discovery in databases* and *data mining*. The confluence of these perspectives has given rise to a sometimes symbiotic and sometimes competitive view of data mining. One important issue within the data-mining framework is

* Correspondence to: Edward J. Wegman, Center for Computational Statistics, George Mason University, Fairfax, VA 22030-4444, U.S.A.
† E-mail: ewegman@gmu.edu

Contract/grant sponsor: National Science Foundation; contract/grant number: DMS-9631351.
Contract/grant sponsor: Army Research Office; contract/grant number: DAAG55-98-1-0404, DAAD 19-99-1-0314.
Contract/grant sponsor: Defense Advanced Research Projects Agency; contract/grant number: 8905-48174.

the issue of data set size and scalability of algorithms. These issues are far from a final resolution.

Data mining from a computer science point of view is often defined in terms of approaches that can deal with large to massive data set sizes. An important implication of this definition is that analysis almost by definition has to be automated so that interactive approaches and approaches that exploit very complex algorithms are prohibited in a data mining framework. Generally, I prefer to define data mining as follows. *Data mining is an extension of exploratory data analysis and has basically the same goals, the discovery of unknown and unanticipated structure in the data. The chief distinction between the two topics resides in the size and dimensionality of the data sets involved. Data mining in general deals with much more massive data sets for which highly interactive analysis is not fully feasible.* In some sense, then there is continuity between exploratory data analysis and data mining and, in a real sense, a contemporary view is that exploratory data analysis tools are being incorporated into the data mining toolkit. Indeed, examples preferred by computer scientists in the data mining framework often fall in the data set sizes statisticians might characterize as small or medium. In a real sense, data mining is emerging as an interdisciplinary area involving computer scientists, statisticians, data analysts and disciplinary scientists.

While the very largest data sets, that is, data sets larger than $10^8$ bytes, are not easily accessible via graphical visualization methods, data visualization is often a powerful tool for the exploration of data. In particular, I have found several visualization tools, which when used in conjunction with each other, form an extremely useful approach to visual data mining. In this paper, I am not intending to survey all data mining methods, or even all visual data mining methods. Rather I am interested in explaining the basic motivation of my methods and illustrating how these methods can be used to accomplish visually a number of traditional statistical tasks. I illustrate these tasks with a number of data sets.

Section 2 of this paper describes data set sizes and focuses on the implications of data set size on the scalability of visual data mining. Section 3 describes several of the tools I have found to be most useful for visual data mining. Section 4 describes the use of these methods for density estimation and data editing. Section 5 describes application of these methods to inverse regression and tree-structured decision rules. Section 6 illustrates variable selection and dimension reduction procedures while Section 7 illustrates application of visual data mining to clustering, classification and discrimination. Finally, Section 8 discusses outliers and unique events.

## 2. THE HUMAN VISUAL SYSTEM AND VISUALIZATION LIMITS

Wegman [1] discusses issues of computational complexity and data transfer as well as visual complexity. There seem to be several critical way-points where distinct modes of analysis take over. At somewhere between $10^6$ to $10^7$ bytes, analysis with complex algorithms on conventional desktop computing begins to fail. By complex algorithms I mean algorithms with computational complexity of $O(n^2)$ or higher. Similarly, data transfer over standard Ethernet begins to take an unfeasibly long amount of time. At this data set size, concerns about statistical optimality give way to concerns about computational optimality.

A second critical way-point occurs at the data set size of about $10^{12}$, a size Wegman [1] characterized as massive. Notice, for small enough data set sizes data can be stored in memory or primary storage. For larger data sets, data must be stored on hard disks or secondary storage and accessed piecemeal. For data set sizes I characterize as massive, data must be stored in robotic magnetic tape silos or tertiary storage. Access becomes orders of magnitude slower and focus shifts dramatically from statistical considerations to computational considerations.

For purposes of visual data mining, a critical question is 'What is the maximum number of data points I can hope to process visually?' After all, exploratory data analysis is traditionally graphically oriented. Such a question must be answered in terms of the human visual system, which clearly begins with a finite number of cells in the eye, most importantly the high-resolution foveal cones. I pose the following thought experiment. Suppose that I could encode each observation into a single pixel. The question then becomes 'What is the maximum number of pixels I could hope to resolve?' Alternatively, how close can two pixels be in angular separation and still be resolvable by the human visual system?

This question was also addressed in Wegman [1]. Based on both psychometric experiments and anatomical considerations, somewhere around $10^6$ to $10^7$ observations appears to be the practical limit for visualization of data while massive data sets easily venture into the range of $10^{12}$ bytes. The human eye has approximately $10^7$cones implying that visualizing one observation per cone would optimistically put the upper limit of visual resolution at about $10^7$ observations. Thus, data mining as such cannot successfully exploit visualization for truly massive data sets without some modification of the raw data. In Wegman [2] I have suggested several approaches including binning and thinning to reduce the size of data sets making visual analysis more feasible. Given that something on the order of $10^6$ to $10^7$ observations is a practical limit for visual data mining, this still leaves an enormous range of data set sizes that are amenable to some form of visual exploration.

## 3. THREE VISUAL DATA MINING TOOLS

With this caveat made, I would like to illustrate in this paper how several more or less standard statistical tasks can be carried out visually. My basic approach involves a combination of three tools: parallel co-ordinates multi-dimensional displays; the *d*-dimensional grand tour; and saturation brushing. In combination these three tools are available in downloadable software for Silicon Graphics systems called ExplorN (available at ftp://www.galaxy.gmu.edu/pub/software/ExplorN_v1.tar) and also in a self-extracting demonstration version of a commercial PC software called CrystalVision (available at ftp://www.galaxy.gmu.edu/pub/software/CrystalVision.exe).

### *3.1. Parallel co-ordinates*

Parallel co-ordinates is a multi-dimensional visualization tool discussed by Inselberg [3] and employed for data visualization by Wegman [4]. In order to represent a *d*-dimensional point, the basic idea is to draw *d* parallel axes labelling them according to the data variables. A point is then represented by locating the value of each variable (component) along its respective

axis and then joining the resulting points by a broken line segment. Many such diagrams are found in this paper. A full discussion of the statistical and data analytic interpretations of parallel co-ordinate displays is given in Wegman [4]. Some of the mathematics underlying parallel co-ordinate displays are discussed in Wegman and Solka [5].

Obviously, one great advantage of the parallel co-ordinate display is that it represents $d$-dimensional points in a two-dimensional planar diagram. In principle, there is no upper limit to the number of dimensions that can be represented although there are practical limits related to the extent of 'real estate' available on the paper or computer screen and, of course, limits imposed by the human visual system. The big idea of the parallel co-ordinates representation, however, proceeds from its interpretation in terms of projective geometry. Both the Cartesian co-ordinate display and the parallel co-ordinate display can be regarded as projective planes. The mappings from the Cartesian co-ordinate system to the parallel co-ordinate system can be shown to preserve certain mathematical properties through the projective geometry notion of duality of points and lines. Focusing for the moment in two dimensions, notice that a point in Cartesian co-ordinates maps into a line in parallel co-ordinates. The dual of this is also true, that is, a line in Cartesian co-ordinates maps into a point in parallel co-ordinates. The point-line duality is illustrated in Plate 4 where the near coincidence of lines in the region between the NUB and CRACK axes represents a line in Cartesian co-ordinate space.

This duality holds for a wide range of mathematical structures. One extremely useful duality is that conic sections in Cartesian co-ordinates map into conic sections in parallel co-ordinates. In particular, ellipses in Cartesian co-ordinates map into hyperbolas in parallel co-ordinates. This mapping is covered in a number of projective geometry texts, including Semple and Kneebone [6], and was first applied to parallel co-ordinate displays by Dimsdale [7] and was subsequently extended by Inselberg [3]. Thus, point clouds from high-dimensional ellipsoidal distributions can readily be recognized in parallel co-ordinates by structures with hyperbolic boundaries. (Plates 1 and 2 using the pollen data illustrate this effect.) Other dualities of interest include the fact that rotations in Cartesian co-ordinates get mapped into translations in parallel co-ordinates and vice versa, and inflection points in Cartesian co-ordinates are represented by cusps in parallel co-ordinates and vice versa.

Perhaps the other single most useful feature of parallel co-ordinate displays is the ability to distinguish clusters. In the diagrams accompanying this paper, I draw the parallel axes horizontally although others have drawn the parallel axes as vertical axes. In principle, there is no difference as to what can be seen although I have always thought that because of standard aspect ratios, the horizontal layout provided slightly easier visualizations. Assuming the horizontal layout, then any gap in any horizontal slice of the diagram separates two clusters. This ability to separate clusters is an extremely important feature of parallel co-ordinates.

Other interpretations include ability to detect linear structures and multi-dimensional modes. The appearance of these is illustrated in Wegman [4]. One final feature worth mentioning is the ability to compare observations on a common scale. Of all of the multi-dimensional data representations such as star plots, Chernoff faces, glyphs, and so on, parallel co-ordinates are unique in their ability to represent common scales on parallel axes. Cleveland and McGill [8] point out that this is the easiest form of measurement comparison for humans to make. Finally, I note that a parallel co-ordinate display is a generalization of a two-dimensional scatter plot. I shall refer to 'data clouds' even when strictly speaking the parallel co-ordinate display involves line segments rather than point clouds.

### 3.2. The *d*-dimensional grand tour

The *d*-dimensional grand tour is a generalization of the two-dimensional grand tour introduced by Asimov [9] and Buja and Asimov [10]. The grand tour is an animation of the data. The basic idea of a grand tour is to look at a data cloud from all possible points of view. There are two key elements of a grand tour algorithm: *space filling* and *continuous*. The notion of space filling is key because it allows for the data analyst to look at the data from all points of view. The notion of continuous is equally important because it allows the human visual system to follow the data cloud in the sense that, as the tour progresses, individual points make only small incremental changes at each step of the tour. As implemented, the *d*-dimensional tour is a continuous geometric transformation of a *d*-dimensional co-ordinate system such that all possible orientations of the co-ordinate axes are eventually achieved. This animation allows for much more structure to be revealed than would be from simply gazing at static plots.

The application of the Asimov–Buja algorithm for this purpose was originally described in Wegman [11] and a fuller discussion of several different approaches to finding continuous, space filling grand tours is described in Wegman and Solka [5]. The key idea is to find a space-filling path through the set of all rotation matrices as a function of a time parameter. This can be done by one of several algorithms including the Asimov–Buja winding algorithm, a random path algorithm, or the Solka–Wegman fractal algorithm. Once a rotation matrix is determined, the canonical basis vectors for the co-ordinate system are rotated by the matrix and then the data cloud is projected into the rotated co-ordinate system. Finally, the projected data are displayed in a parallel co-ordinate display (or alternatively in a scatter plot matrix). Coupled with the parallel co-ordinate display, these two techniques allow for an in-depth study of high dimensional data. Partial grand tours can be accomplished by holding one or more variables fixed. A grand tour is in some sense a generalization of a two-dimensional rotation, although it is not a rotation in the conventional sense.

### 3.3. Saturation brushing

Saturation brushing is a generalization of ordinary brushing. Ordinary brushing is accomplished by brushing a data cloud with a colour for the purpose of visually isolating segments of the data. Normally this is implemented by drawing a rectangular box. Any data point or line segment which intersects the rectangular box is coloured with the chosen colour. Cutting is closely related to brushing. The same basic rectangular box is drawn. However, with cutting, any point or line within or touching the box is eliminated. Cropping eliminates any point or line *not* within or touching the box. Together, cutting and cropping allow one to prune the data set to focus on data points of interest.

In data sets where there is considerable overplotting, ordinary brushing is potentially confusing and misleading, especially where there is an animation such as rotation or grand tour. This is the case because in many computer graphics algorithms, colours are drawn according to the *z*-depth; lowest *z*-depth points are drawn last. This can lead to apparently arbitrary changes of colour and certainly gives no clue as to the amount of overplotting. In saturation brushing, each point is assigned a highly desaturated colour (nearly black) and when points are overplotted, their colour saturations are added via the so-called $\alpha$-channel.

The $\alpha$-channel is a computer hardware device for blending two images. Generally it is used to provide transparency in advanced computer graphics applications. Because it is a hardware rather than software implementation, it is usually extraordinarily fast. I use the

$\alpha$-channel recursively to add saturation levels of pixels. Thus, heavily overplotted pixels have fully saturated colours, whereas pixels with little overplotting remain nearly black. Saturation brushing is described by Wegman and Luo [12] and is an effective method for dealing with large data sets. Coupled with parallel co-ordinates and the grand tour, these methods allow for an extremely effective visual approach to large high-dimensional data.

I have discussed the so-called brush tour strategy, which is a strategy for data analysis using parallel co-ordinates and the grand tour, see Wilhelm *et al.* [13]. The essence of the idea is to isolate clusters recursively. Begin with a parallel co-ordinate view of the data cloud and use colour brushing to isolate clusters that are visible in the initial view. After those are brushed with colour individually, initiate the grand tour. As new subclusters emerge, halt the tour and brush the new clusters with distinct colours. Repeat the process until no new clusters emerge. The grand tour has the ability to give new perspectives so that clusters that may not have been apparent in the original view will turn up after a rotation. Because the tour is constructed as space filling, eventually all clusters will be seen. Of course, in theory this could take forever. In practice I have found that a few minutes of the tour seems sufficient to identify useful subclusters. The brush tour is illustrated in Wilhelm *et al.* [13] with a data set known as the Oronsay sand particle size data set and gives another illustration of using a combination of these data visualization tools for data mining.

### *3.4. Colour design*

Colour design is an important aspect of visual data mining. Thus, a word on colour strategy is also useful. Because I use the $\alpha$-channel for blending to build up saturation levels, I can also use the $\alpha$-channel to blend different colours. The computer screen is an additive colour system. The primary colours are red, green and blue. Red blended with green makes yellow, red blended with blue makes magenta, and green blended with blue makes cyan. All three together make white. Because yellow is the sum of red and green, yellow plus blue make white. Similarly, cyan plus red makes white, and magenta plus green also makes white. The choice of colour for bushing can greatly aid in the visual analysis. A strategy I have followed when comparing or contrasting two classes is to choose complementary colours such as cyan and red. Where these classes overlap, the result is white. Thus one can somewhat create a visual hypothesis test simply by seeing the strength of the white (or grey). Similarly, if I compare three classes, a useful strategy is to choose red, green and blue so that their overlap leads to distinct colours such as cyan, magenta, yellow and white. For more classes, of course, the colour choices become more complex. However, some additional choices one can make are 100–50 per cent mixtures. For example 100 per cent red and 50 per cent green yields an orange colour that is also highly distinguishable. If this is blended with 100 per cent blue, a pink colour results. Thus some experimentation yields a set of highly distinguishable colours, which may be used with a larger number of clusters.

One other comment on colour design is appropriate. In this paper, almost all illustrations are made against a white background. This is done in order to conserve ink and make the printed version somewhat more legible. On a computer screen I often choose to have a black background. There are two advantages to a black background. First, saturated colours tend to be prominent against a black background, whereas yellows, light cyans, and white tend to disappear against a white background. Thus some of the colour strategy is lost when using a white background. Second, when using saturation brushing, pixels representing observations

that are not overplotted are nearly black and hence fade into the background. Thus they are visually unimportant, which reflects their statistical significance. On the other hand, if one desires to see the infrequent observations, one only has to make the background white. Thus, depending on the goal and the sample size, one can choose between a black or a white background colour.

While I have found the combination of parallel co-ordinate displays, $d$-dimensional grand tours and saturation brushing to be particularly effective tools for data mining of large, high dimensional data sets when used in concert, there are more well-known auxiliary tools I also find very helpful. These include scatter plot matrices, three-dimensional stereoscopic scatter plots, density plots (all of which can be animated using the grand tour), linked views, and pruning and cropping tools for visual data editing. All of these tools are available in the CrystalVision software. The illustrations in this paper are also made with the CrystalVision software.

## 4. DENSITY ESTIMATION AND RAPID DATA EDITING

Saturation brushing can be used with either parallel co-ordinate displays or with ordinary scatter plots. In effect, by using saturation brushing with a single colour (say white against a black background or grey against a white background) the brushed display becomes a density estimate. Traditionally density estimation is done with local smoothing, for example by a kernel smoother. With saturation brushing, local smoothing is not required. The data are, in effect, binned by the resolution of the screen so that the screen image becomes in effect a histogram with many bins with bin count mapped into colour intensity. For ordinary high resolution displays this means there are approximately $1.3 \times 10^6$ bins or pixels. With the $\alpha$-channel, this density estimation is accomplished with the same speed as the rendering of an ordinary two-dimensional image. If a high-density region is isolated, then pruning and cropping tools can be used to rapidly focus on the region of interest.

A nearly perfect first illustration of these visual data mining techniques is the so-called pollen data set. This data set was the 1986 JSM Exposition's data set and was assembled by David Coleman of RCA Labs. Artificially generated, the data are comprised of 3848 observations on five variables. It is available at http://lib.stat.cmu.edu/data-expo/1986.html. The first view of this data is a scatter plot matrix view that one might conventionally see, see Plate 1. In this scatter plot matrix with fully saturated pixels, we see a series of pairs plots with elliptical cross-section. An immediate conclusion might be that this is multivariate normal data in five dimensions with no additional structure suspected.

Plate 2 shows the parallel co-ordinate view of the same data. Notice that in the parallel co-ordinate view, one has a series of scalloped edges, which are essentially hyperbolas. As mentioned earlier, hyperbolas in parallel co-ordinates are the duals of ellipses in Cartesian co-ordinates. Thus, the parallel co-ordinate view is showing the same information as the scatter plot matrix view. However, the strategy is now to desaturate the image. I brush with a very dark grey, nearly black colour. Where there is heavy overplotting the small white components add up to a much lighter grey as in Plate 3.

When the desaturated parallel co-ordinate plot is examined in Plate 3($a$), at least two interesting features can be observed. First, a bright feature in the middle of the diagram suggests that there is a hidden structure. In addition, the larger X-features in the display

suggest that the overall elliptical clouds have a five-dimensional hole in the middle. The five-dimensional hole was deliberately put in this data set as a feature hopefully to be discovered. The X-feature suggests that points that have a large component on one axis have a small component on the next. There is a deficit of medium components on one axis joining to medium size components on the next axis. This corresponds to having a hole. Of course, a hole would never be recognized by simply plotting a fully saturated lower dimensional projection because the projections would cover the hole. The five-dimensional hole can be confirmed by initiating a grand tour and observing whether the X-feature persists. If it does, then the hole is real; if it disappears, then the X-feature was an artefact of one static view. In this case, it persists.

The central structure can rapidly be isolated using a combination of the cutting and the cropping tools. This is best done in the parallel co-ordinate display. The cropping tool is used to remove all observations except those belonging to the central feature found in Plate 3($a$). As can be seen in Plate 3($a$), the central feature is a high-density region of points buried in the middle of the much larger and more variable collection of points. Plate 3($b$) illustrates an intermediate state of pruning. The results are shown in Plates 4 and 5. Plate 4 is the parallel co-ordinate view that shows several key features. First, there are five gaps in horizontal slices at many places. This corresponds to six clusters that have been coloured in Plate 4 with six distinct colours. Also visible in this image are a number of places where the line segments in the parallel co-ordinate display cross at a single point. Recall that points in the parallel co-ordinate display are duals of lines in Cartesian co-ordinates. The six subclusters contain linear features, which turn out to be the vertical strokes in the letters spelling EUREKA. This central structure shown in Plate 4 is actually 99 data points that spell the word EUREKA as illustrated in Plate 5.

In summary, I have used the combination of parallel co-ordinates and saturation brushing as a form of visualizing data density to isolate two major features. The X-structure is indicative of a five-dimensional hole in the ellipsoidal data cloud, but within the data cloud was also buried a high density feature that I isolated with the data editing tools of cropping and cutting. This central feature was only 99 data points of the original 3848 and contained six clusters corresponding to the word EUREKA. I also observed that there were linear subfeatures within each of the six clusters. I like to use the pollen data set because it is very useful not only as a illustration of the analysis process, but also in developing intuition about parallel co-ordinates. I use the grand tour in order to verify the persistence of the five-dimensional hole and also to verify that the clusters are real and stay coherent through a generalized rotation.

## 5. INVERSE REGRESSION AND TREE-STRUCTURED DECISION RULES

Perhaps one of the most popular data mining tools is the tool known in the statistical literature as CART (classification and regression trees) known more generically simply as decision trees. Classification trees focus on a categorical response variable and a number of predictor variables. At the most rudimentary level, the classification tree begins with a binary split on one of the predictor variables with the idea of separating the population into as nearly pure response variable as possible. Having made such a split, another predictor variable is chosen and a second binary split is made. At each split further branches on the tree are created. The splits are made according to some optimality criteria, typically some ‘purity criterion’ so as at

each split to make the result leaves on the tree as pure as possible. This is a greedy algorithm. Obviously, beginning with a different predictor variable will yield a different tree so that there is no uniqueness to the algorithm. Continuing the binary splits will eventually lead to a very large tree in which each leaf is an individual observation. Clearly, some pruning is typically required. Usually, the process begins with training data and a cross-validation process aids with the pruning.

Regression trees are similar to classification trees except that the response variable is continuous rather than categorical. The classic reference for CART is Breiman *et al.* [14]. This methodology is extremely successful and has been developed by the original authors with many improvements. Further information on CART and its follow-ons can be found at http://www.salford-systems.com/. Incidentally, the classic application of this methodology is medical diagnostics, that is, separating populations into disease categories.

Much the same thing can be done visually, but with certain added desirable features. I titled this section 'inverse regression and tree-structured decision rules' because I intend to show how both aspects can be accomplished. In order to illustrate the process, I begin with some bank risk data. The data at hand are 132 147 records of eight-dimensional data. The variables refer to individual bank customers and are PFT (profit), AGE, YRS (years as customer), OCC (occupation code, a categorical variable), MOS (months in present residence), RST (residence status, for example, homeowner, renter etc., also a categorical variable), MST marital status, also categorical), and SEX (a binary variable). Essentially, the variables other than PFT are demographic variables. I am searching for combinations of demographic variables that will identify customers who cause the bank to lose money or to make profits. Hence, this becomes a risk analysis.

The general strategy, after a preliminary data pruning to eliminate cases with age unknown, occupation unknown etc., is to use brushing to colour cases with negative profit red and positive profit green. I remove the PFT variable from the tour, and alternate pruning and touring operations. I allow the grand tour to run until a strongly red or strongly green region appears among the touring variables. The customers represented by those regions have demographics leading, respectively, to losses or to profits for the bank. It is easy to see how this tour-prune strategy will lead to a decision tree similar to CART, but with two added features. Because I am doing a grand tour, I am no longer considering the original variables, but orthogonal linear combinations of the original variables. This leads to a richer decision space, but of course loses the simple interpretability of splitting on the original variables. (In the bank risk application this is desirable since decisions are not being made solely on the basis of age, occupation or sex.) In addition, the visual data mining using saturation brushing allows one to get a sense of how significant a pruning step is and how many cases a particular pruning step involves. I illustrate several steps in Plates 6 to 11.

The sequence of tour-prune steps leads to a tree-structured decision rule in a manner similar to the CART algorithm. By recording the combination of variables at the end of each tour step, I can reconstruct the exact combination of variables being pruned and construct an accurate decision tree. Just as with the CART strategy, training data are required and cross-validation is used to verify the level of pruning of the resulting tree that is required. The visual cues are quite helpful in determining a sensible stopping point.

I also labelled this section as an inverse regression process. When using regression, one normally views the response variable or dependent variable as a function of the explanatory or independent variables. The process I have been using is to fix the response variable (PFT) and

allow the tour on the independent variables. In a sense, what I was doing is seeing how the independent variable behaved as a function of the response variable. Hence, this is an example of inverse regression done visually rather than analytically. There is an interesting aside to this process. Because each of the new variables resulting from the grand tour is an orthogonal linear combination of the original independent variables, one can compute the $R^2$ associated with the new variables as predictors of the response variable and simply keep a record of those new variables with large predictive capability. In other words, the partial grand tour I have described amounts to simultaneous orthogonal linear predictors of the response variable. Simply tracking the $R^2$ as a function of the tour variable $t$ would allow us to essentially visually do a near optimal regression fit by recording where the local maxima of $R^2$ occurs. In my experience, the tour will come close to the optimum relatively rapidly.

## 6. VARIABLE SELECTION AND DIMENSIONALITY REDUCTION

One significant problem associated with using multivariate data for discriminant analysis is the problem of dimension reduction, that is, variable selection. Typically, one wishes to do dimension reduction for two purposes. First, many of the variables may not be good discriminators and may in fact be essentially noise in the process and thus reduce discrimination capabilities. Second, even if all the variables are reasonable discriminators, the computations required with a large number of discriminators may sufficiently increase so as to render real-time computation impossible.

Such is the case with the so-called SALAD data. SALAD is an acronym for Shipboard Automatic Liquid Agent Detector. The SALAD system is a sensor system for detecting airborne chemical-biological warfare agents. The system works by allowing aerosols to settle on a chemically treated paper strip and then examining the resulting colour changes in the paper. The SALAD data I am using for this example consists of 9375 points in 14 dimensions. One dimension is used for a class variable. There are three classes of chemical agents in this particular data set. The remaining 13 variables are intensity of colour in 13 spectral bands. Colours are less bright on the left end of the axis, brighter to the right end of the axis. Parallel co-ordinate displays coupled with saturation brushing can allow us to select variables visually for the purpose of discriminant analysis. The data are shown in Plate 12. The 13 variables are intensity of spectral response in 13 colour bands with increasing wavelength.

The variable on the bottom axis is the classification variable and is brushed with one of three colours red, blue or green according to the class of chemicals. The idea is to look for one or more of the spectral bands that adequately discriminates the three classes of chemicals. Because I have three classes, using the three colours red, blue and green is the colour combination of choice. The additive colour feature can again be exploited: red + blue = magenta; red + green = yellow; blue + green = cyan.

Notice that variable B10 separates blue and red, and, in fact, shows two distinct red clusters. Recall from the pollen example that any gap in a horizontal axis is interpreted as a separation of clusters. Unfortunately, B10 does not discriminate red from green. However, in parallel co-ordinate displays, the slope of the line segments matters considerably. The slope of the green line segments between B9 and B10 is substantially different from the slope of the red line segments between the same axes. Thus, the variable B10 – B9, a surrogate for slope, will distinguish red from green. Thus, only two variables, B9 and B10, are adequate to discriminate

all three classes and in fact also discriminate the two subclasses of the red. Notice also that many of the variables offer poor discrimination capabilities. For example, B4 to B8 have large cyan regions, indicating that these do not discriminate between the blue and the green classes of agents. Similarly B1, B2 and B13 have white regions indicating that they would be unable to discriminate among all three classes of agents. Thus, the judicious choice of variables not only allows for noise reduction in the discrimination process, but also dramatically speeds up the computation. This dimensionality reduction allows for real-time discrimination of chemical warfare agents that obviously is of crucial importance for this situation. I observe by the way that in this application, I used only parallel co-ordinates and saturation brushing. The grand tour was not needed.

## 7. CLUSTERING, CLASSIFICATION AND DISCRIMINATION

The data in this section arise from a DARPA-sponsored project known among those of us working on the project as the artificial dog nose project. Canines are used because of their extremely sensitive olfactory capabilities to detect mines in minefields. The project was aimed at examining the possibility of creating an artificial nose that would have the capability of sniffing the organic decomposition products of high explosives and thus provide a capability for locating and disarming anti-personnel mines. Alas, while the visualization techniques reported here provide excellent discrimination ability and point towards automated analytical methods for discrimination, the artificial nose itself is about two orders of magnitude less sensitive than the real canine olfactory system and hence is not yet able to replace the real thing.

Because of the desirability of distinguishing clusters in highly multivariate data and because distance-based clustering is normally an $O(n^2)$ algorithm, clustering and classification are ideal applications that can exploit visual data mining techniques. One of the most interesting clustering/classification/discrimination applications I have run across is the application to time series data.

The set-up is as follows. The ends of fibre optic strands are doped with one of 19 different chemical dopants. The dopants are reactive to organic molecules and exhibit transient reactions. The reactions change the light reflectivity of the fibre optic strand. The change in reflectivity of the fibre optic is also sensitive to the exciting frequency of the light. In the experiments, some 300 different combinations of organic molecules in various concentrations and various mixtures were used to excite the artificial nose. Two distinct frequencies of light were also used. The measurements were time series of length 60 for each combination of organic stimulant and frequency of light.

Some preliminary plots are given in Plates 13 to 15. These are, of course, ordinary scatter plots of time series taken from sniffs of TCE across the 19 fibre optic sensors in the two frequency bands. One band is coded in red the other is coded in cyan. This colour choice is made because I am attempting to discriminate two classes. Red and cyan are complementary colours in an additive colour system. They will add up to white. Plate 13 shows that in fibre 1, the waveforms are similar but with differing amplitudes. However, Plate 14 plots fibre 17 against fibre 19 and shows that a phase loop exists. Finally, Plate 15 shows that the recording for fibre 7 is defective in the frequency represented by red. Because fibre 7 is a defective component, I want to eliminate it from both the display and any grand tour I might initiate. By also eliminating the time variable from the grand tour, one interesting capability

I have is to form a grand tour of the time series. In particular, I can form orthogonal linear combinations of the time series. This idea was first described in two-dimensions as an image grand tour in Wegman *et al.* [15]. The idea is that one can extract subtle features from combinations of images (or time series) that do not show up in any single image or time series. This idea was applied by Vanderluis [16] to twelve-lead electrocardiograms to create synthetic leads for viewing the posterior side of the heart. In my application, however, I am interested in classification of and discrimination among chemical species.

Plate 16 is the parallel co-ordinate display of this data after a grand tour. I have eliminated fibre 7 from the display and from the grand tour. I also hold the time variable out of the grand tour. The goal of my exercise is to find a tour rotation that allows us to separate the two frequency components in some maximal sense. I would like to believe that the two frequencies provide independent information. The partial grand tour was stopped in Plate 16 because of the excellent separation properties. Notice that the rotated variables (indicated by an asterisk) X19*, X18*, X16*, X15*, X9*, X3* and X2* all provide strong separation between frequencies coded with cyan and with red. The data are essentially 18-dimensional time series (leaving out X7) and I have found a seven-dimensional hyperplane that separates these two time series. Note also that I have, in essence, discarded the time order information from this particular exercise.

A word on the colour choices in Plate 16 is in order. Normally, when doing exploratory analysis on the computer screen, I prefer to use a black background. Colour perception is usually stronger against a black background. As mentioned earlier, when I am working with two classes I prefer to use complementary colours. Cyan and red are complements and when overplotted add up to white. White against a black background is very easily distinguished. For purposes of reproduction in a printed journal, it is more efficient to have a white background. In the case of most of the diagrams I use in this paper, I have simply converted the black background to white. However, for Plate 16 this would have resulted in the loss of ability to distinguish the overplotted pixels from the background. Plate 16 is actually the negative of the black background computer screen. I do this so that the overplotted pixels will show up as black.

Plate 17 represents another set of four time series traces each with kerosene in different concentrations. The third trace also has a TCE contaminant. The goal again is to find a separating hyperplane in which the TCE containing mix will be distinguishable from the time series that contain kerosene only. In this data set, the same frequency is being used so I do not have the luxury of a look in two separate frequency bands. As shown above, the two frequency bands provide a substantial amount of independent information. Because the waveforms generally have the same shape, the discrimination problem is fairly difficult.

Plate 18, like Plate 16, represents the data after a partial grand tour rotation. As in Plate 16, I have used a negative image so that the overplotted pixels show up as black against a white background. Distinguishing these four chemical species in a single frequency regime is a considerably more difficult problem, yet, with reasonable separation rotated variables X1*, X3*, X5* and X11* provide a four-dimensional separating hyperplane for this data. Notice that variable X7 is omitted from the display and from the grand tour as in my earlier example because of its defective measurement. Of course, as before the time variable is also omitted from the grand tour. In Plate 18 the second time series that corresponded to the highest concentration of kerosene is also deleted. It is actually easy to detect because of its high concentration levels. Because it produces large values, it unnecessarily compresses the scale

of the other chemical mixes making it very difficult to see the separations. As in Plate 16, the time information is essentially suppressed.

## 8. OUTLIERS AND UNIQUE EVENTS

Perhaps one of the hardest tasks associated with data mining in general is the detection of outliers and unique events. Most statistical techniques as well as computer science-based data mining techniques are oriented towards finding usual behaviour as opposed to unusual behaviour, that is, behaviour that is not exhibited on a large scale. Multivariate outlier detection is a particularly compute intensive task. Techniques for multivariate outlier detection available in the statistical literature, such as minimum volume ellipsoids (MVE) or minimum covariance determinant (MCD), are exponentially complex and exact algorithms for these methods fail with sample sizes as small as 100 or 200 observations.

Because of the exponential complexity of multivariate outlier detection algorithms, it is usually not feasible to use analytical methods for multivariate outlier detection. It is also well known that a multivariate outlier may not show up in any of the low-dimensional projections. The combination of grand tour and parallel co-ordinate displays provides an ideal tool for finding multivariate outliers in a way similar to finding clusters. Outliers are, in effect, clusters of very small size. While an outlier may not show up in a low-dimensional projection, the grand tour in effect searches all possible multi-dimensional co-ordinate systems and will eventually (and usually quite quickly) find co-ordinate systems in which the outliers are clearly exhibited. I note that inherently both the grand tour and the parallel co-ordinates plot are algorithms whose computational complexity is $O(n)$. The limitations of sample size are not inherently due to the computational complexity of the algorithms, but rather are limited by the abilities of the human visual system and resolution of the computer screen.

Plate 19 is based on a project I carried out at the Bureau of Labor Statistics in which point-of-sales (POS) data for breakfast cereals was being examined as a test bed for possible inclusion in the consumer price index. The entire data set was some 5.5 gigabytes of basic information about the amount and price of cereals sold in a major metropolitan area over a period from 1995 until the present. One of the questions of interest is 'What is the effect of aggressive sales promotions in creating outliers?' Outliers can have a substantial effect on distorting the consumer price index. Plate 19 examines the outliers in the variables quant1, expend1, quant2 and expend2. These variables represent, respectively, the quantities and expenditures on breakfast cereals in year 1 which was 1999 and year 2 which was the year 2000. In Plate 19 all observations were initially brushed red and then outliers in year 1 blue and year 2 green. The rather unexpected result was that the outliers in both years were associated with the same chain of grocery stores. A more extensive discussion of the data analysis of the BLS POS cereal data can be found in Wegman and Dorfman [17].

## 9. CONCLUSIONS

Our thesis in this paper is that, while data visualization techniques have some inherent limitations for really massive data sets, visual data mining methods have powerful capabilities for interactive data analysis. I have focused on a combination of three methods – parallel

co-ordinates, $d$-dimensional grand tours, and saturation brushing – that I believe have substantial capabilities for use in the analysis of large, high-dimensional data sets. I have illustrated their use in performing a number of tasks including rapid data editing, density estimation, inverse regression, the formulation of tree-structured decision rules and their application to risk assessment, dimension reduction and variable selection, classification, clustering, discriminant analysis, multivariate outlier detection and unique event detection. I have also described the brush-tour and the tour-prune strategies. All of the visualization methods described here have $O(n)$ computational complexity whereas analytical alternatives may have $O(n^2)$ complexity or even exponential complexity.

ACKNOWLEDGEMENTS

This paper is based on an invited talk of the same title presented at the 2001 CDC/ATSDR Symposium on Statistical Methods. The kind invitation to speak is gratefully acknowledged. The pollen data were originally prepared by David Coleman and are available through the StatLib website. The bank data were provided by the AMS Corporation through my friend and frequent co-author Qiang Luo. These data are proprietary and are not publicly available. The SALAD data were provided by the Naval Surface Warfare Center courtesy of another friend and frequent co-author Jeffrey Solka. These data are also not publicly available. The artificial dog nose data were generated as part of a DARPA project and were provided by my friend and frequent co-author Carey Priebe. Finally, the BLS cereal data were provided as part of a Senior Faculty Fellowship project I carried out at the Bureau of Labor Statistics and are also proprietary. The analysis of the data was carried out using the CrystalVision data mining software. This research was supported by the National Science Foundation grant DMS-9631351, by the Army Research Office under grants DAAG55-98-1-0404 and DAAD19-99-1-0314, and by the Defense Advanced Research Projects Agency under Agreement 8905-48174 with The Johns-Hopkins University. A portion of the analysis presented here was completed while Dr Wegman was an ASA/NSF/BLS Senior Research Fellow. The opinions expressed in this paper are personal opinions of the author and do not represent official policy of the Bureau of Labor Statistics.

REFERENCES

1. Wegman E. Huge data sets and the frontiers of computational feasibility. *Journal of Computational and Graphical Statistics* 1995; **4**(4):281–295.
2. Wegman E. Visions: the evolution of statistics. *Research in Official Statistics* 1999; **2**(1):7–19.
3. Inselberg A. The plane with parallel co-ordinates. *Visual Computer* 1985; **1**:69–91.
4. Wegman E. Hyperdimensional data analysis using parallel co-ordinates. *Journal of the American Statistical Association* 1990; **85**:664–675.
5. Wegman E, Solka J. On some mathematics for visualizing high dimensional data. *Sankhya* (A) 2002; **64**:429–452.
6. Semple JG, Kneebone GT. *Algebraic Projective Geometry*. Oxford Science Publication: Oxford, 1952.
7. Dimsdale B. Conic transformations and projectivities. *Technical Report 6320-2753*, IBM Los Angeles Scientific Center, Santa Monica, CA, 1984.
8. Cleveland W, McGill R. Graphical perception: theory, experimentation, and application to the development of graphical methods. *Journal of the American Statistical Association* 1984; **79**:531–554.
9. Asimov D. The grand tour: a tool for viewing multidimensional data. *SIAM Journal on Scientific and Statistical Computing* 1985; **6**:128–143.
10. Buja A, Asimov D. Grand tour methods: an outline. In *Computer Science and Statistics: Proceedings of the Seventeenth Symposium on the Interface*, Allen D (ed.). North Holland: Amsterdam, 1985; 63–67.
11. Wegman E. The grand tour in k-dimensions. In *Computing Science and Statistics: Proceedings of the 22nd Symposium on the Interface* 1991; 127–136.
12. Wegman E, Luo Q. High dimensional clustering using parallel co-ordinates and the grand tour. *Computing Science and Statistics* 1997; **28**:352–360 (republished in *Classification and Knowledge Organization*, Klar R, Opitz O (eds). Springer-Verlag: Berlin, 1997; 93–101).
13. Wilhelm A, Symanzik J, Wegman E. Visual clustering and classification: the Oronsay particle size data set revisited. *Computational Statistics* 1999; **14**(1):109–146.

14. Breiman L, Friedman J, Olshen R, Stone C. *Classification and Regression Trees*. Wadsworth: Pacific Grove, 1984.
15. Wegman E, Poston W, Solka J. Image grand tour. *Automatic Target Recognition VIII – Proceedings of SPIE* 1998; **3371**:286–294 (republished, vol. 6, *Automatic Target Recognition. The CD-ROM*, Firooz Sadjadi (ed.). SPIE: Bellingham, WA, 1999).
16. Vanderluis JP. *Enhanced visualization of cardiac electrophysiology using virtual leads*. PhD dissertation, School of Computational Sciences, George Mason University, 2000.
17. Wegman E, Dorfman A. Visualizing cereal world. *Computational Statistics and Data Analysis* 2002 (to appear).
18. Wegman E, Carr D. Statistical graphics and visualization. In *Handbook of Statistics 9*: *Computational Statistics*, Rao CR (ed.). North Holland: Amsterdam, 1993; 857–958.

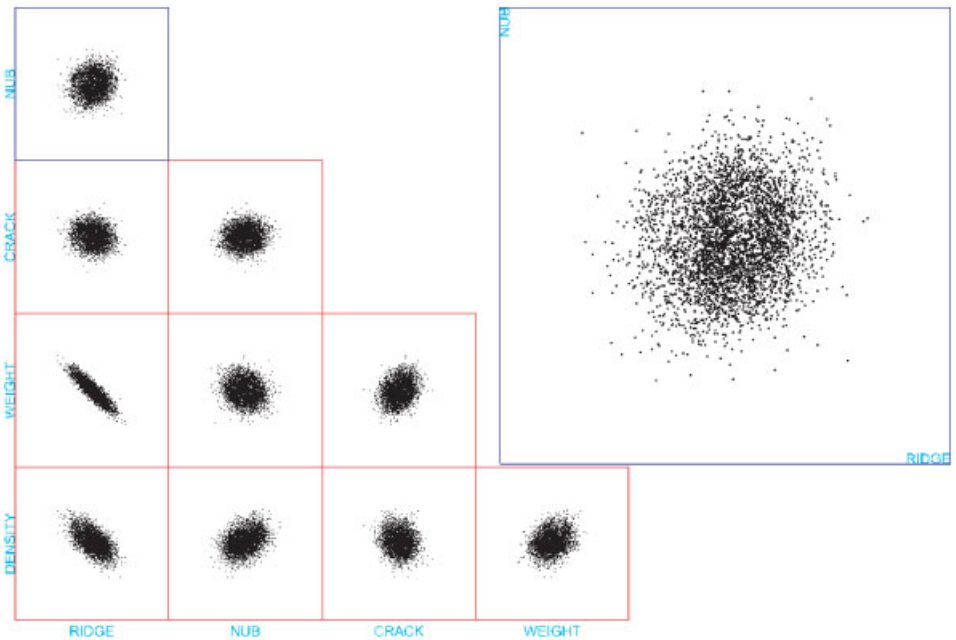



Plate 1. Scatter plot matrix of the pollen data.

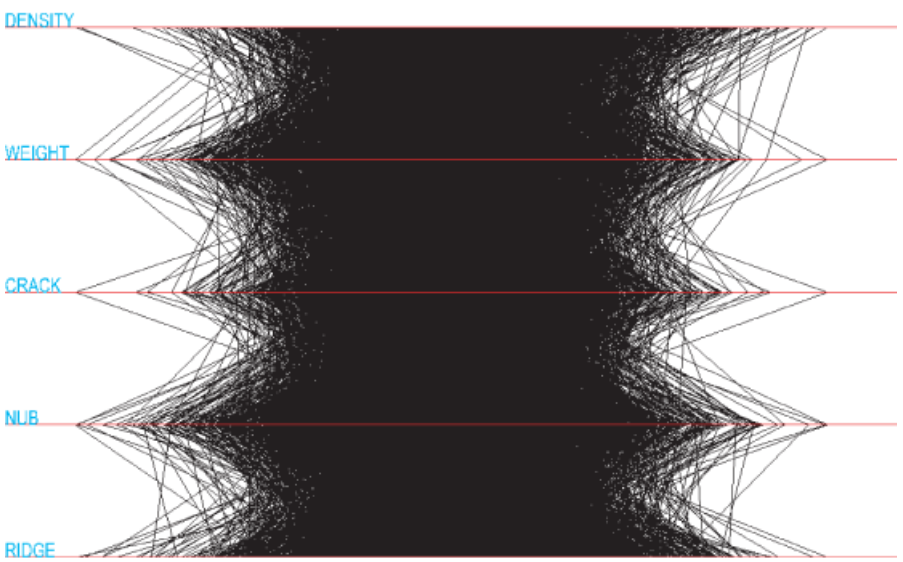
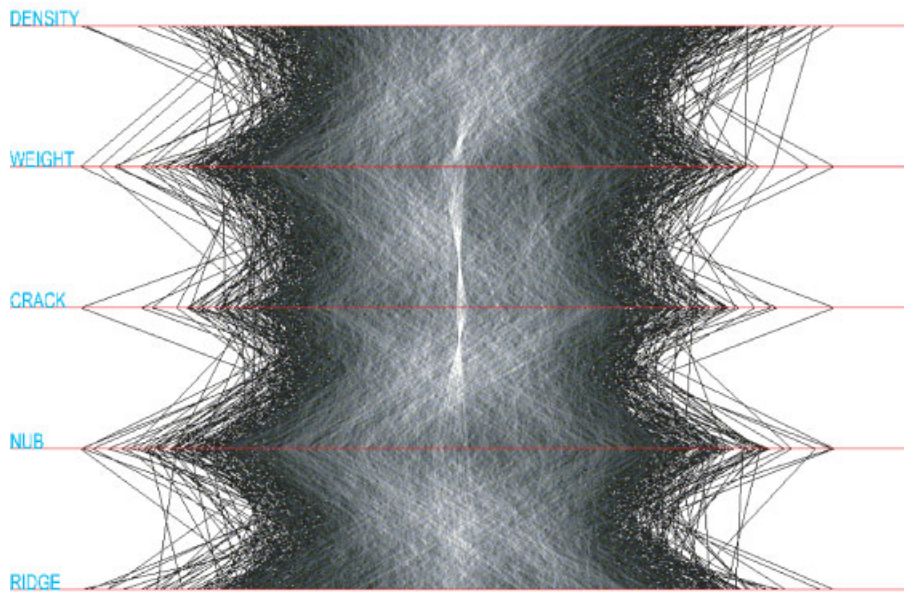
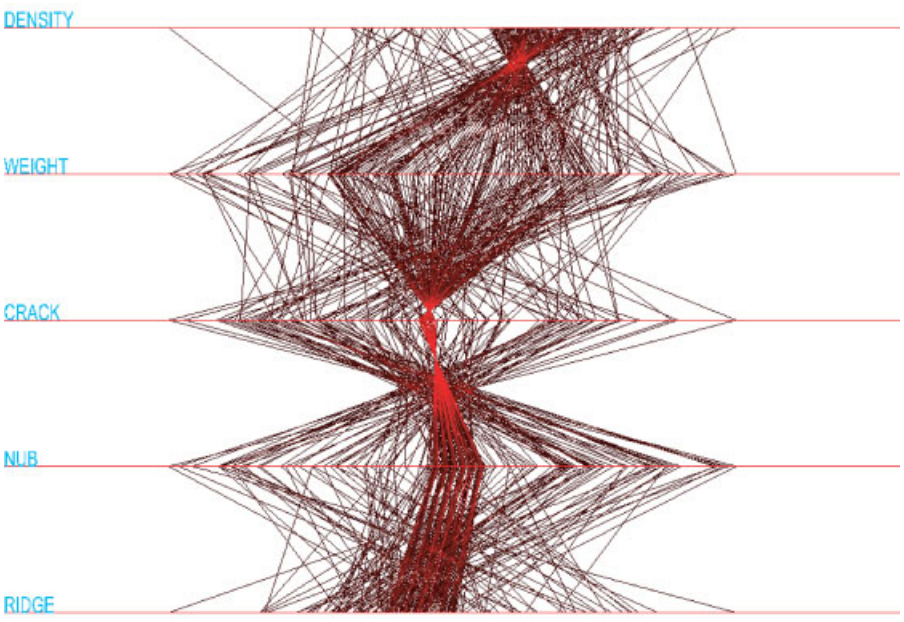



Plate 2. Parallel co-ordinate display of the pollen data. This is the standard view with every occupied pixel drawn in full black.




(a)




(b)

Plate 3. (a) Desaturated parallel co-ordinate plot showing a central feature not visible in the standard plots. (b) Parallel co-ordinate plot with intermediate level of pruning showing the central feature brushed with desaturated red.

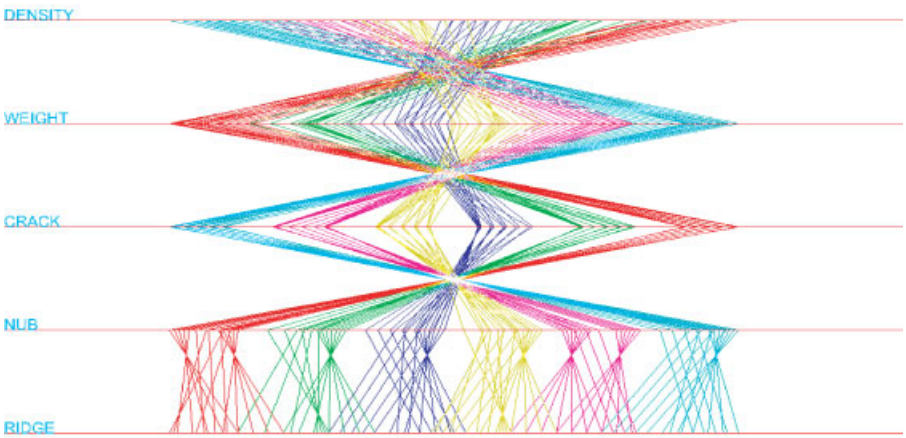



Plate 4. The central feature isolated by using the cropping tool. Six clusters are recognized which are coloured respectively with red, green, blue, yellow, magenta and cyan.

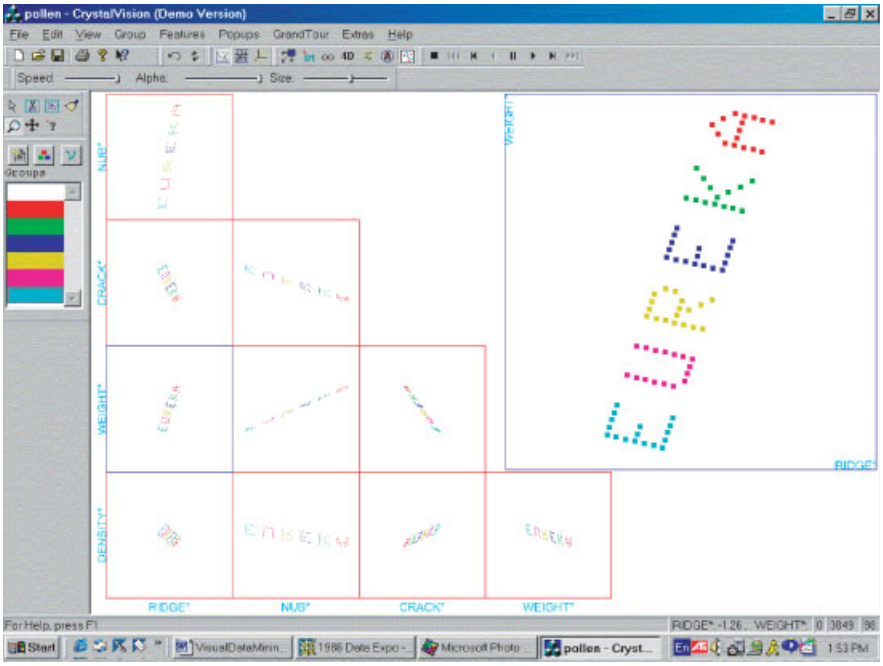



Plate 5. The central feature shown in scatter plot matrix form. This is a screen shot from the CrystalVision software.

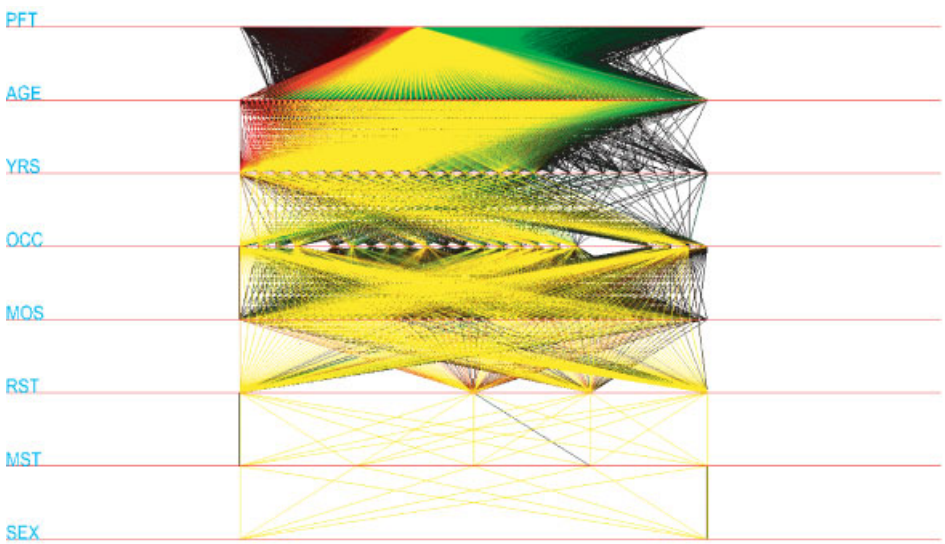



Plate 6. Original parallel co-ordinate display of bank risk data. PFT variable is brushed with green for positive profit, red with negative profit. Note that yellow is the sum of red and green and indicates neutrality. Note in general young customers tend to lead to negative profits while older customers tend to lead to positive profits.

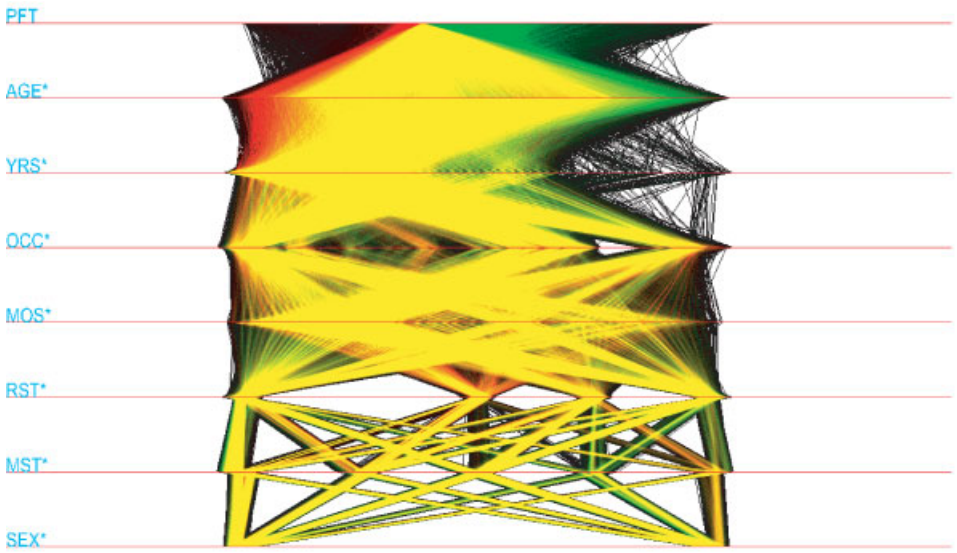



Plate 7. A small distance along the grand tour has the effect of dithering the data. The toured variables are now marked with an asterisk and are no longer solely the original variables, although at this stage the original variables dominate. It is clear that there is at least one risky occupation, a fairly risky subset of the RST variable and a fairly risky subset of the MST variable.

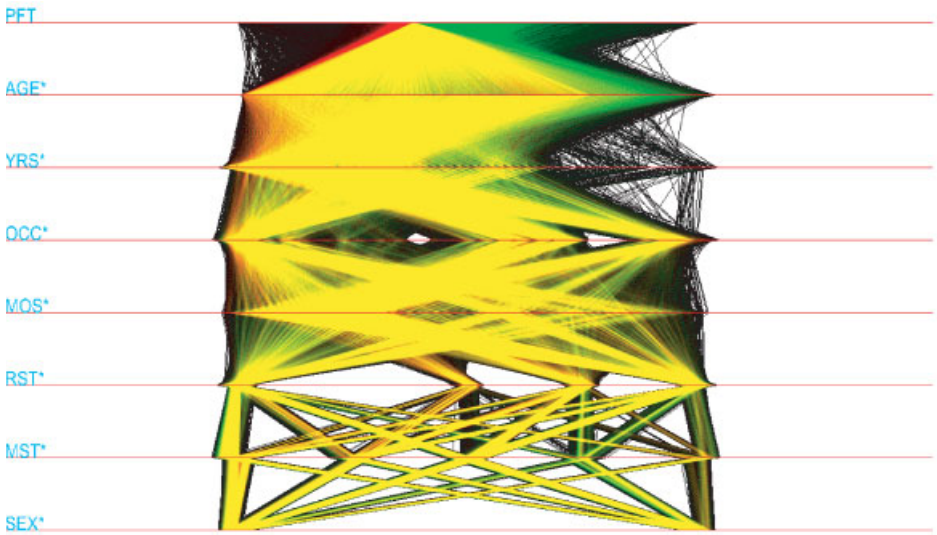



Plate 8. This is the same image as Plate 7, but with those three higher risk categories pruned. The software allows us to record the new linear combinations so that I know precisely which new variables have been pruned.

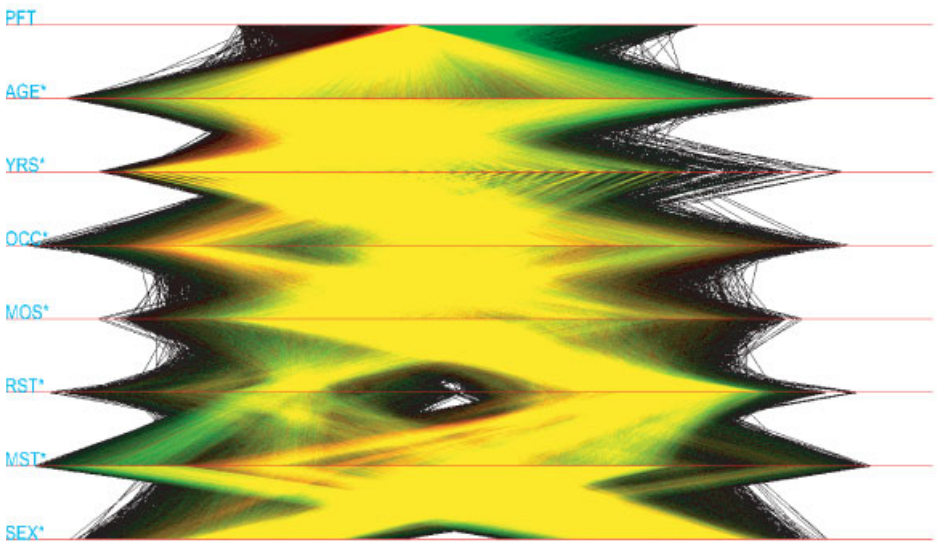



Plate 9. This image is much further along the way in the grand tour after several pruning steps have already been accomplished.

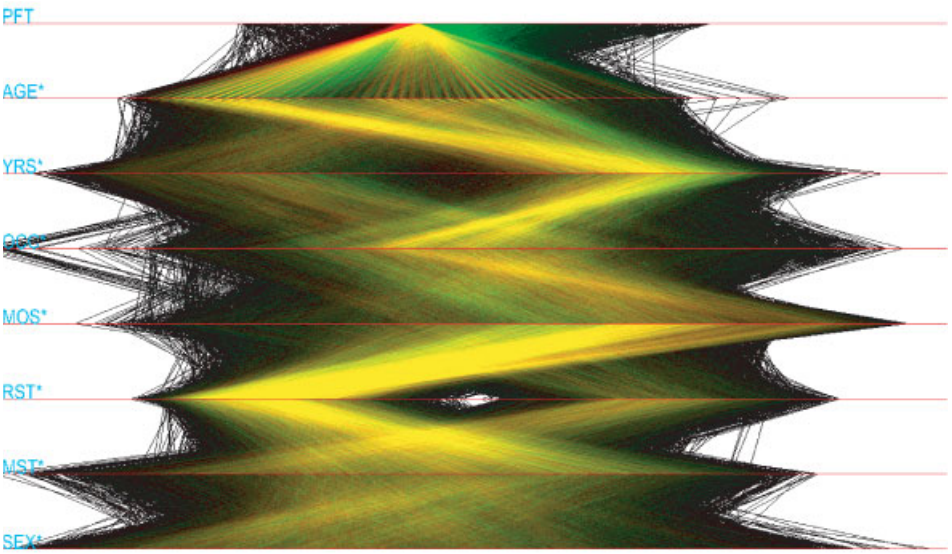



Plate 10. After several additional tour-prune steps. The increasingly dark images indicates that the sample size is decreasing and that too much further pruning may not be useful.

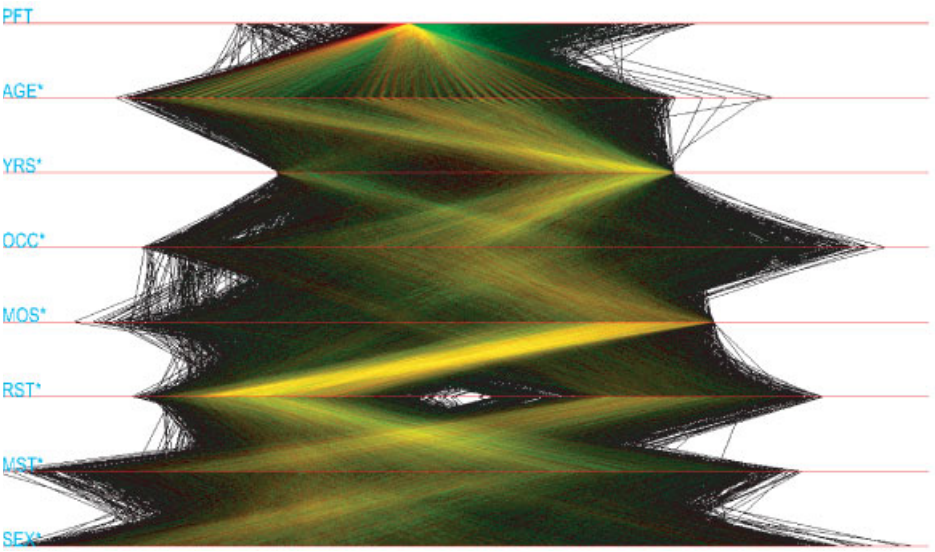



Plate 11. This is the same image as Plate 10 with one last pruning step accomplished. I had actually carried analysis through 11 tour-prune steps and reduced the original data set of 132,147 observations to 5713 observations. The present image is after eight tour-prune steps.

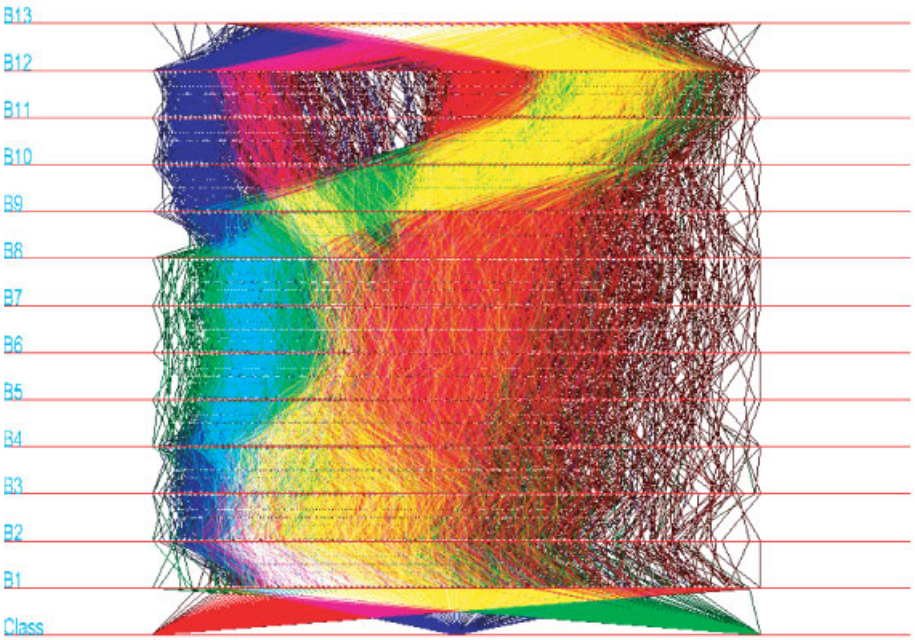



Plate 12. Parallel co-ordinate plot of SALAD data for chemical-biological agent detection. Variables B10 and B9 together allow for rapid discrimination of each class of chemical agents and even provide information on two subclasses of the 'red' agent.

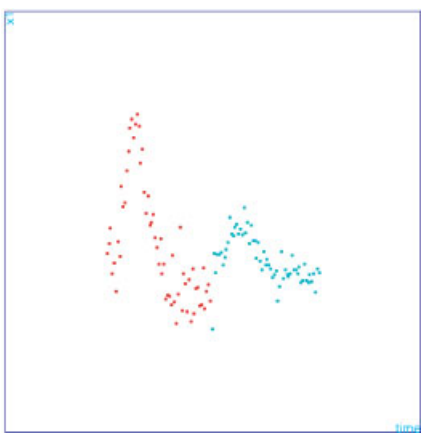


Plate 13. Time series plots of two sniffs (one coloured red, the other coloured cyan) of trichloroethylene (TCE) at different wavelengths for one doped strand of the fibre optic artificial nose. Although these waveforms appear similar except for amplitude, other fibres give substantially different appearance.

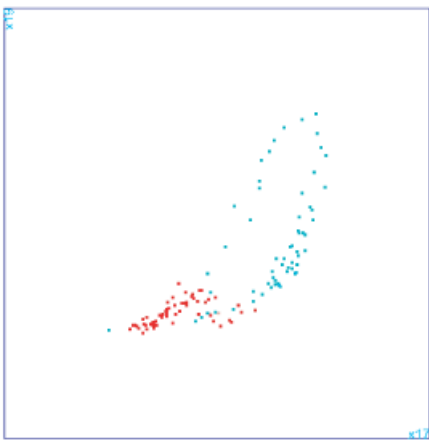


Plate 14. Fibres 17 and 19 plotted in a scatter plot. The frequency coded with cyan shows a phase loop, while the frequency with red does not.

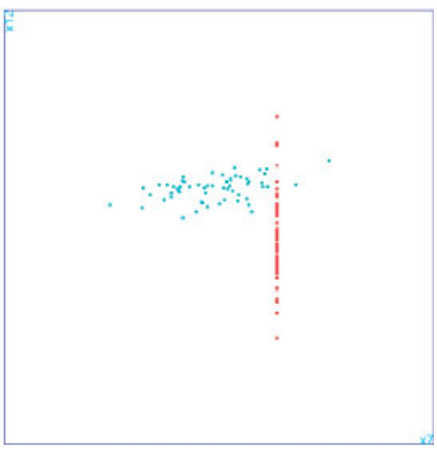


Plate 15. Fibre 12 plotted against fibre 7 in scatter plot. Note that fibre 7 is degenerate in the frequency band coded with red.

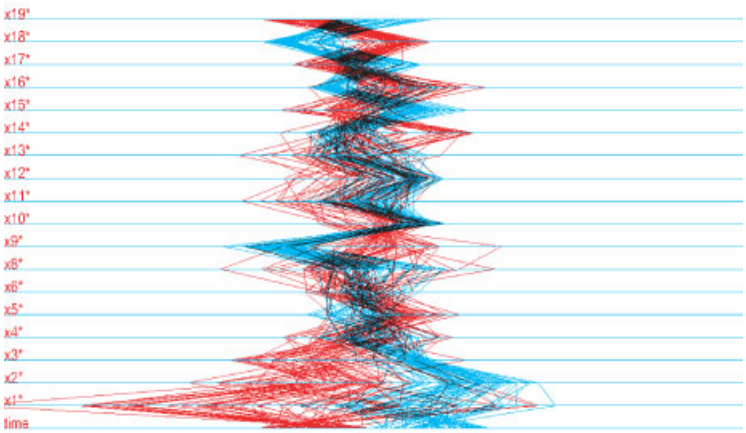



Plate 16. Parallel co-ordinate display of the 19 fibres after a grand tour. This image is actually a negative so that the overlap of cyan and red that would normally show up as white shows as black in this image.

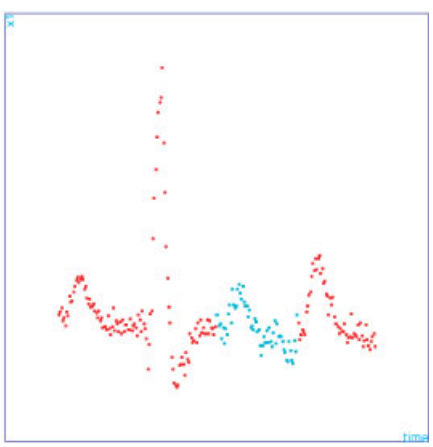


Plate 17. Four time series traces concatenated. The target species is coloured with cyan. The remaining three traces are coded in red. The goal again is to find a maximally separating hyperplane. Notice that the general shape of the waveform is the same for each trace.

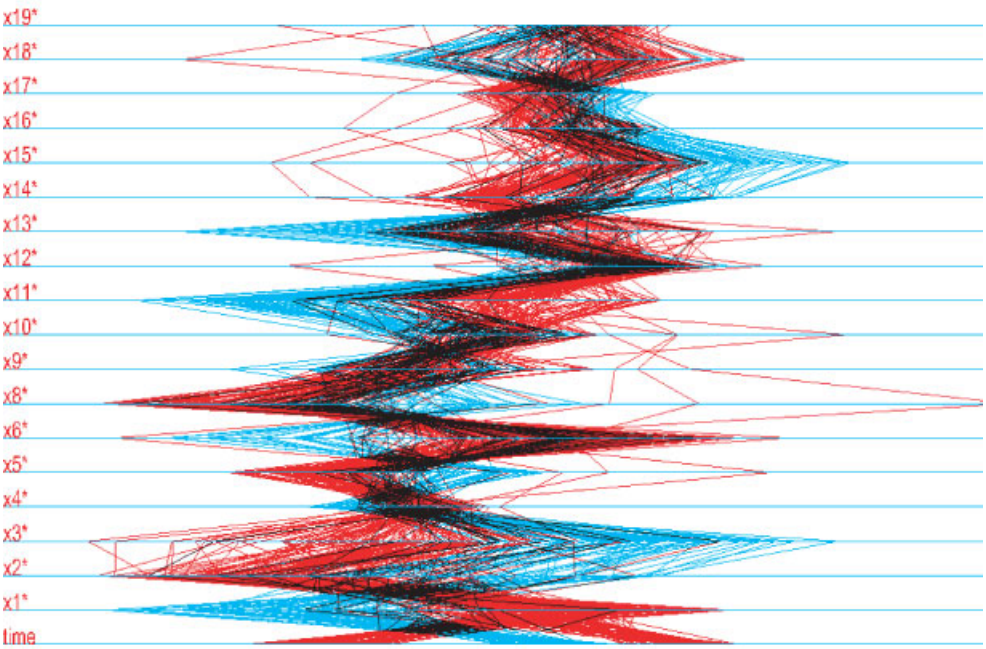



Plate 18. Parallel co-ordinate plot of the four species time series showing strong separation in variables $x1^*$, $x3^*$, $x5^*$ and $x11^*$.

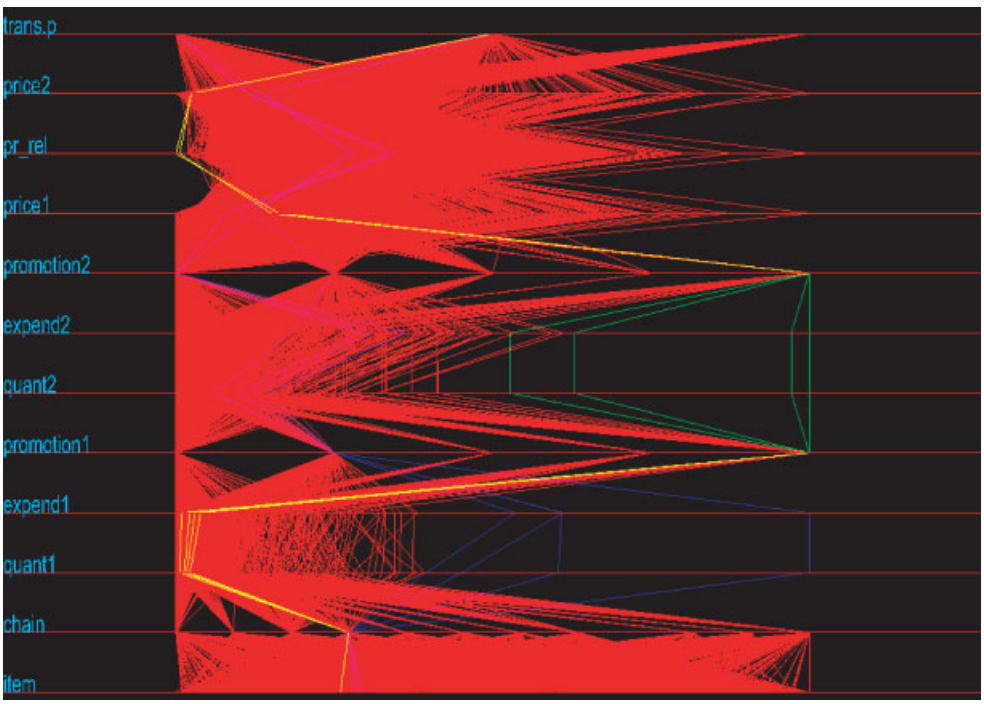



Plate 19. Outliers in terms of high sales volumes for certain breakfast cereals in two separate years, 1999 and 2000.