High-Dimensional Visualizations

Georges Grinstein[1, 2], Marjan Trutschl[1], Urška Cvek[1]

[1]Institute for Visualization and Perception Research and [2]AnVil Informatics, Inc.
University of Massachusetts Lowell

Abstract

In this paper we provide a brief background to data visualization and point to key references. We differentiate between high-dimensional data visualization and high-dimensional data visualizations and review the various high-dimensional visualization techniques. Our goal is to define metrics that identify how visualizations deal with *n* dimensions when displayed on the screen. We define *intrinsic dimensionality* metrics that assess these techniques and closely analyze selected high-dimensional visualizations' display of data.

Keywords: visualization techniques overview, evaluation, high-dimensional data visualization, metrics

1 INTRODUCTION

A visualization is a visual representation of data. Data is mapped to some numerical form and translated into some graphical representation. The term "high-dimensional data visualization" and "high-dimensional visualization" are often used interchangeably. However, a visualization of high-dimensional data is different than a high-dimensional visualization. In the first the term "high" refers to data whereas in the second it refers to visualization. This paper defines some simple metrics for high-dimensional visualization.

We assume the data is *n*-dimensional where *n* is an integer. In this paper we focus on high-dimensional data visualizations and more specifically visualizations that can present a large number of dimensions or parameters of the data. We attempt to identify what constitutes a high-dimensional visualization.

All visualizations basically still end up on a display surface (soft or hardcopy). There are a few 3D-displays and much of what follows still apply to these. One interpretation therefore is that all visualizations project the *n*-dimensional data down to 2 dimensions. Although this is correct we wish to differentiate between the dimensionality of the physical medium (2 dimensions) and the logical representation of the data that may be higher. An example can be given by considering a 3D scatterplot.

1 Department of Computer Science, University of Massachusetts Lowell, Lowell, MA, 01854, USA. Email: {grinstein | mtrutsch | ucvek} @cs.uml.edu

2 AnVil Informatics, Inc., 600 Suffolk Street, Lowell, MA 01854, USA.

Here the data is *n*-dimensional, 3 axes are selected and laid out on the plane (the physical medium). The *n*-dimensional points are projected on the 2D surface. Hence this is a 3-dimensional visualization on the 2D surface. Note that we could also consider the dimensionality of the data represented. By using color and shape we could argue that a 3D scatterplot is a 5-dimensional representation of *n*-dimensional data on a 2D surface. In such a display there are perceptual ambiguities resulting from the occlusion of points. These can be resolved by providing various tools, including interactive ones. For example the user can rotate such a display to see hidden points.

We can thus classify visualizations based on the intrinsic dimensionality of the logical representation as well as its potential dimensionality by adding in additional data attributes. Since the additional data attributes can often be applied to most visualizations**,** we will only consider the intrinsic dimensionality.

2 VISUALIZATION BACKGROUND

Visualization is used increasingly in the data exploration process but still not to the extent possible. In its early years it was mostly, if not only, used to convey the results of statistical computation or data mining algorithms [7], [49], [10]. Over the last decade it has been used in the data massaging and cleansing process, and somewhat in the data management process. It is still not being used in the computational steering processes within the data exploration pipeline except for some research systems.

2.1 Visualization Taxonomies

There are numerous visualizations and a good number of valuable taxonomies [45].

Historically static displays, most of which have been extended to support probing and even more dynamic interactions, include histograms, scatterplots, and numerous of their extensions. These can be seen in most commercial graphics and statistical packages.

We focus on tables of numerical data (rows and columns) although many of the techniques apply to categorical data. Looking at the taxonomies the following stand out as high-dimensional visualizations:

- 2D and 3D scatterplots
- Matrix of scatterplots
- Heat maps
- Height maps
- Table lens
- Survey plots
- Iconographic displays

- Dimensional stacking (general logic diagrams)
- Parallel coordinates
- Line graph, multiple line graph
- Pixel techniques, circle segments
- Multi-dimensional scaling and Sammon plots
- Polar charts
- RadViz
- PolyViz
- Principal component and principal curve analysis
- Grand Tours
- Projection pursuit
- Kohonen self-organizing maps

Several of these are quite similar and related. We give a brief description and visualization for each, along with key references (see [23], [19], [13]). We use the Fisher Iris flower data set [15] or the car data set from UC Irvine Machine Learning Repository, whenever possible. The Iris flower data set contains 50 specimens from each of the three species of Iris flowers: *Iris setosa*, *I. Versicolor*, and *I. Virginica*. The dimensions of the data set are sepal length, sepal width, petal length and petal width, measured in millimeters.

3 HIGH-DIMENSIONAL DATA VISUALIZATIONS

3.1 2D and 3D Scatterplots

A scatterplot is a point projection (usually affine) of the data into a 2D or 3D dimensional space represented on the screen in classic (X, Y) or (X, Y, Z) format. This is the most commonly utilized data visualization method.

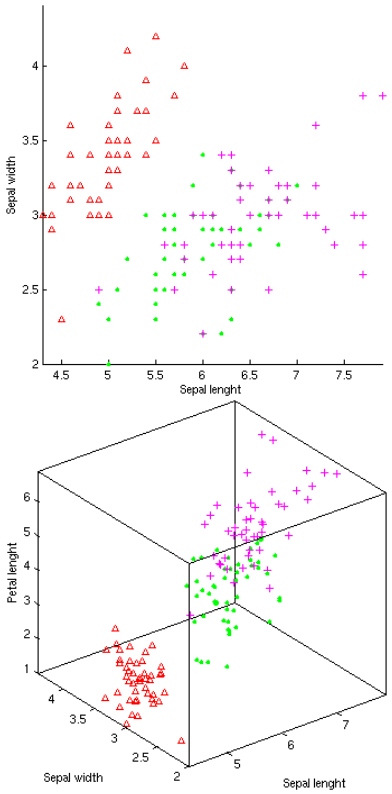

Figure 3.1: 2D and 3D scatterplots of the Iris data set

Numerous mappings or transformations can be applied to it. The displayed points can have numerous attributes such as color, size, shape, texture, motion and even sound (when interacted with). To interpret the 3D projection interaction, it is necessary to resolve ambiguities, although other techniques have been used (animation). In its most general form this method is related to iconographic and pixel displays. Figure 3.1 displays the Iris Flower data set as 2D and 3D scatterplots.

3.2 Matrix of Scatterplots

A matrix of scatterplots is an array of scatterplots displaying all possible pairwise combinations of dimensions or coordinates. For n-dimensional data this yields $n(n-1)/2$ scatterplots with shared scales, although most often n^2 scatterplots are displayed. The scatterplots can also be positioned in a non-array format (circular, hexagonal, etc.). One can visually link features of one scatterplot with features on another, which greatly increases its power.

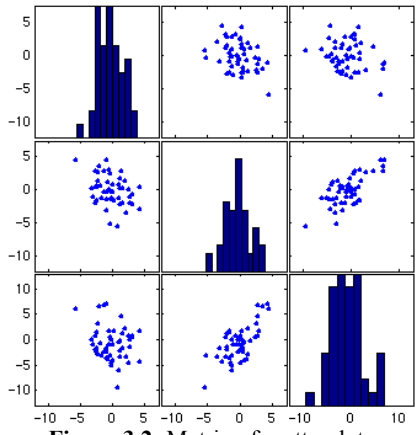

Figure 3.2: Matrix of scatterplots

This technique has been in use long before its publication [3], [7]. Several variations on the theme of a matrix of scatterplots have since been developed: the hyperslice [50], *N*-vision [14], prosection [47], hyperbox [2], just to name a few. The hyperslice is a matrix of panels where "slices" of multivariate function are shown at a certain focal point of interest. The method is similar to *N*-vision, where the matrix panel accommodates for interactive exploration of a multivariate function. Prosection is a method more suitable for data mining, since it does not project all points onto the scatterplot matrix, but rather projects only points within a certain range of each dimension, similar to brushing and dynamic queries [1]. The hyperbox uses the same pairwise projections of the data, but projects onto panels of an n-dimensional box. Each of the panels has a different orientation and the dimensions can be cut in order to show histograms on the panels, according to ranges of the dimensions being cut.

3.3 Heat Maps

This is an array of cells where each cell is colored based on some data value or function on the data. The method is a generalization of a scatterplot where the points are grid cells and each cell is colored. There are many named variants (clustered image map, heatmaps, patchgrid).

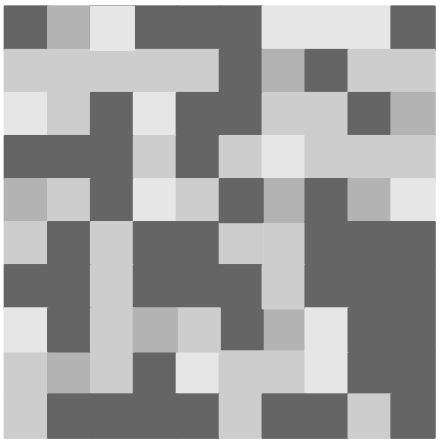

Figure 3.3: Heat map of a random data set

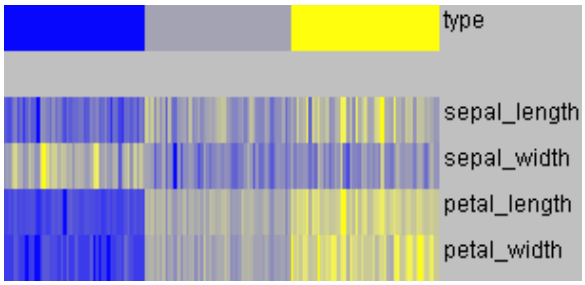

Figure 3.4: Heat map of the Iris data set

3.4 Height Maps

A height map is a further extension of a heat map with the grid represented as a height field instead of by color. Making the cell size small can generate an almost continuous map. An example is ThemeView™ [54], where the topics or themes within a set of documents are shown as a relief map of natural terrain.

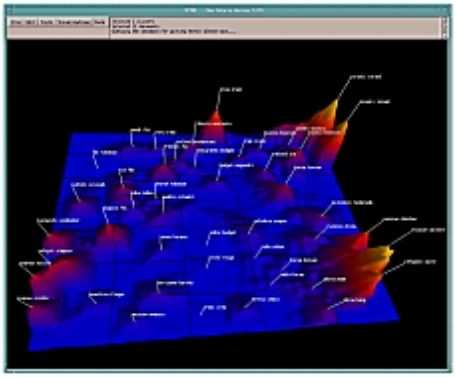

Figure 3.5: Height map of document themes
Source: Pacific Northwest National Laboratory

In Figure 3.5 the mountains indicate themes within the documents with the peak heights as the relative strengths of the topics. The layout of the themes depends on a similarity metric. This visualization is similar to self-organizing maps (SOMs), described below.

3.5 Table Lens

The table lens takes a spreadsheet and allows each cell to be displayed optionally using a line whose length depends on the numerical value of the cell and whose color can represent some other attribute of the data [42]. This provides for both a symbolic and graphical representation of data within a single table. This can be viewed as an intermediate view of data between a pure spreadsheet and a heat map where each item is represented as a number.

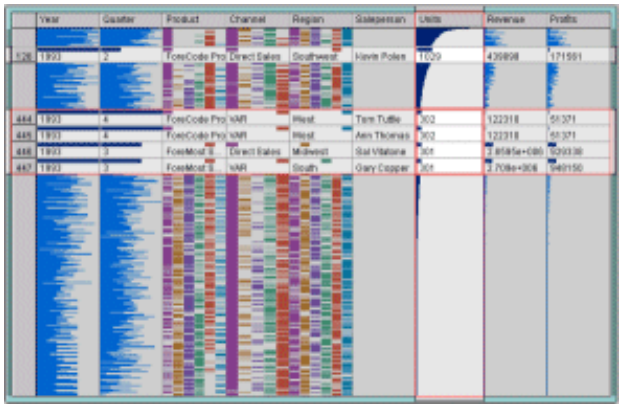

Figure 3.6: Table lens with selected rows of a sales data set
Source: Inxight Software, *http://inxight.com*

3.6 Survey Plots

A survey plot is a 2D or 3D point projection of the data [36] and generally consists of *n* rectangular areas, each representing one dimension in a data matrix. A point in a line graph (like a bar graph) is extended down to an axis. A line (or a rectangle, depending on the number of records and the size of the output area) is used to represent the data for each dimension, with its length proportional to the dimensional value it represents. The method gives insight to correlation between any two variables (especially when the data is sorted by a dimension) and can find exact rules in a machine learning dataset.

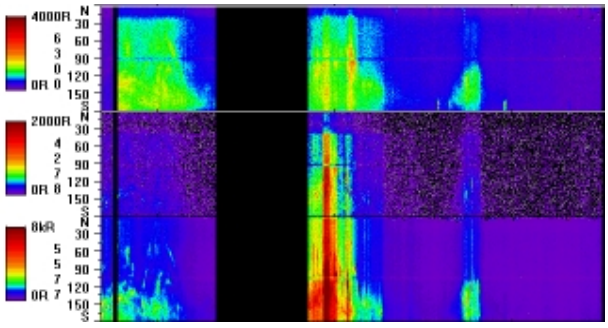

Figure 3.7: Survey plot of atmospheric data
Source: Geophysical Institute, University of Alaska Fairbanks

3.7 Iconographic Displays

An iconographic display is a graphical representation visualizing high-dimensional data by letting each coordinate dimension of a record drive some parameter or attribute of an entity (pixel, icon

or glyph) and displaying a number of these entities (records) at once on the screen. These displays integrate several dimensions at once and thus can represent high-dimensional data sets [3], [8], [41], [35].

There are two types of glyph and icon visualizations; the *first* are displays where certain dimensions of the *n*-dimensional data set are mapped to certain features of the glyph or icon. These include: Chernoff faces [8], where data dimensions are mapped to facial features; star glyphs (plots) [7], where the dimensions are represented as equal angualr spokes radiating from the center of a circle. The *second* type of glyph and icon visualizations have glyphs or icons packed together in a dense display, with textures representing features of the dataset [41]. Some other icon visualizations are shape coding [5], color icons [35], [27], [12] and tilebars [21].

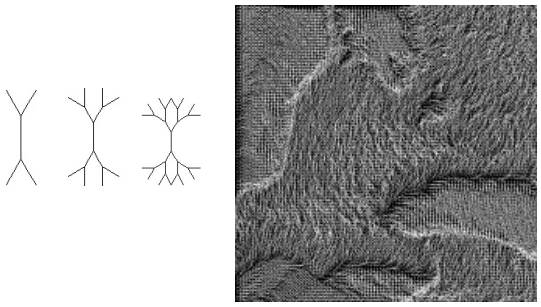

Figure 3.8: An icon and an integrated iconographic display of 5 satellite images of the Great Lakes region

3.8 Dimensional Stacking (General Logic Diagrams)

Dimensional stacking is a 2D or 3D point projection of the data where dimensions are embedded within other dimensions. It was initially used only to visualize binary data [37]. The method was later extended to discrete categorical values and binned ordinal values, and used for general data exploration [52]. The stacking divides a 2D grid into sets of embedded rectangles, representing categorical dimensions or attributes of the data. Two outer dimensions are placed along the X and Y axes, and each additional pair of dimensions is embedded into the outer level rectangles, until all dimensions are incorporated.

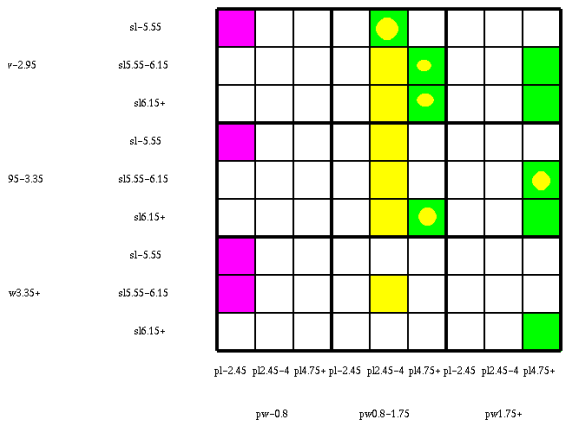

Figure 3.9: Dimensional Stacking of the Iris data set

3.9 Parallel Coordinates

Parallel coordinates use parallel axes instead of perpendicular to represent dimensions of a multidimensional data set [25], [26]. A vertical line is used for the projection of each dimension or attribute, with the maximum and minimum values of each dimension usually scaled to the upper and lower boundaries on those vertical lines. A polyline made up of *n*-1 lines at the appropriate dimensional values connects the axes to represent an *n*-dimensional point.

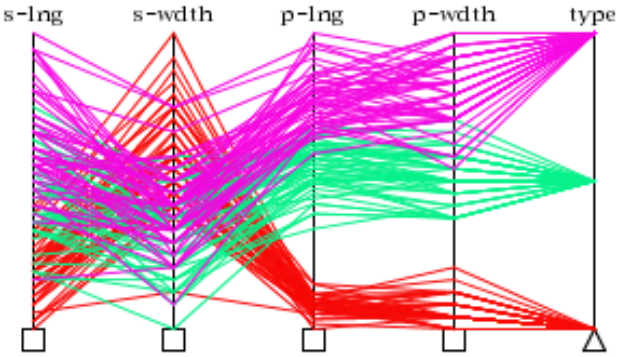

Figure 3.10: Parallel coordinate display of the Iris data set

3.10 Line Graph, Multiple Line Graph

Line graphs display single-valued or piecewise continuous functions of one dimension. To accommodate multi-dimensional data sets, multiple line graphs are displayed in a multi-line graph. Often, the ordering of the data is correlated to one of the dimensions of the data, such as time. The dimensions are distinguished using different colored lines, and/or types of continuous lines (dashed, dotted).

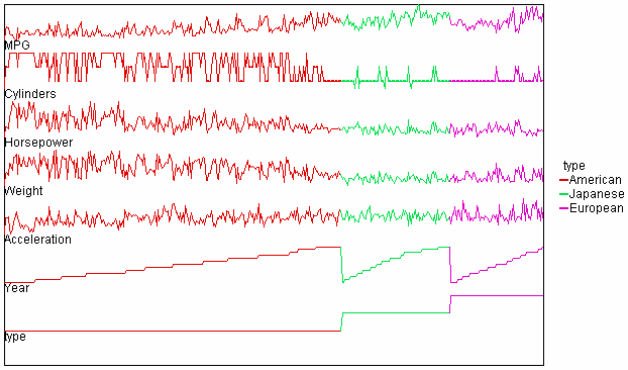

Figure 3.11: Multiple line graph of the car data set

3.11 Pixel Techniques, Circle Segments

Pixel techniques represent a generalization of heat maps, extending them to very large multi-dimensional data sets. These visualizations arrange the data into an area, starting from some origin, according to the size and number of dimensions, using various techniques including recursive, spiral, and circle segments. The interpretation of the (X, Y) position of the cell depends on the mapping. In VisDB [27] the goal is to show

similarities between attributes of the data. Various similarity functions may be used and their values represented as colors.

For circle segments each arc on the circle represents a data value of one dimension. Originally, the arc would represent many data values, one for each pixel in the arc, but variations now use straight lines.

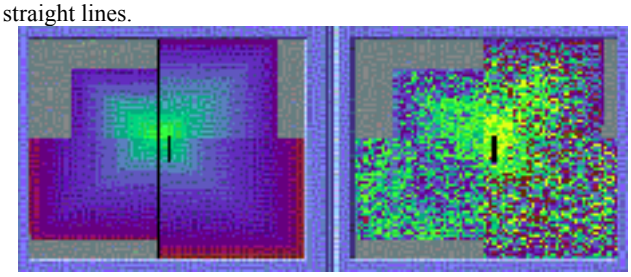

Figure 3.12: Pixel display of an eight-dimensional data set of 1,000 records using 2D arrangement
Source: VisDB, [27]

3.12 Multi-Dimensional Scaling and Sammon Plots

An analytic or graphical representation that maps a data set into a space of lower dimensionality is considered a projection method. In most cases some invariants are preserved or closely preserved (such as distance). This is a classic technique, well over 50 years old [57], [48], [33], [11], [55], [56].

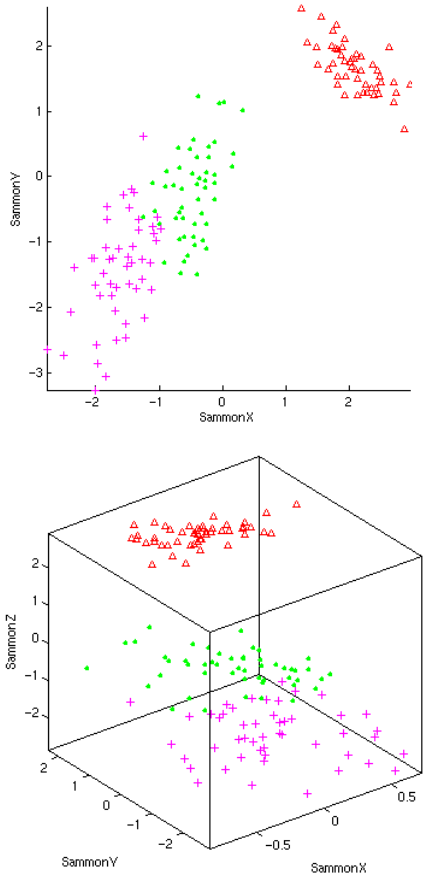

Figure 3.13: 2D and 3D Sammon Plots of the Iris data set

The goal of Multi-Dimensional Scaling (MDS) techniques is to identify meaningful underlying dimensions that could explain similarities or dissimilarities in the data. MDS typically preserves the distance metric. Most often the projection space is 2-dimensional. Other techniques attempt to preserve some degree of structure. The result is a 2D or 3D display in which points close to each other are close in the original *n*-dimensional space.

There are numerous variations and in all cases a dissimilarity matrix is built (based on the selected metric) with various cost functions and other parameters. Bentley and Ward presented extensions to MDS to enhance visualizations of high-dimensional data, such as animation, stochastic perturbation and flow visualization techniques [6]. The most frequently used variation of MDS is Sammon plot, a non-linear MDS mapping [44].

3.13 Polar Charts

A polar chart is a circular graph for plotting polar coordinates. Polar coordinates map data onto a 2D surface using the angle and radius, creating a "wrap-around" version of a line graph. Polar charts bridge the limitation of line graphs, which are used only for displaying single valued or piecewise continuous functions of one dimension. These can be considered circular representations of parallel coordinates and thus can reduce the limiting effect of a large number of dimensions. However, the size of the data point representations depends on the closeness to the center.

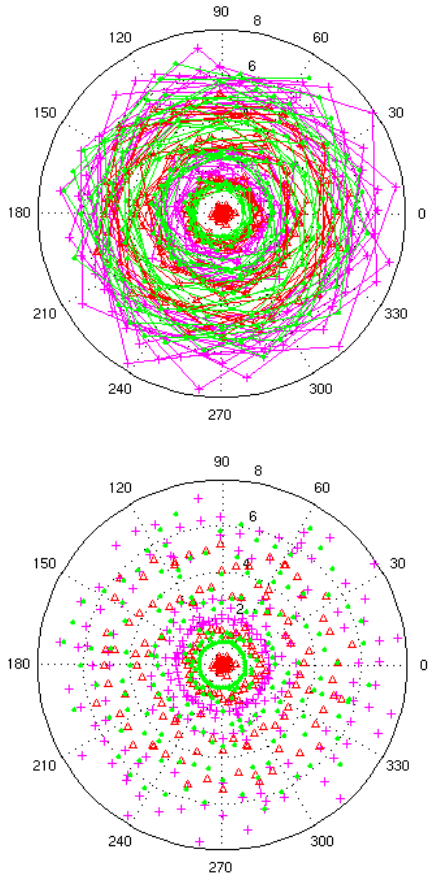

Figure 3.14: Polar line and polar glyph plot of the Iris data set

3.14 RadViz

RadViz is a display technique that places dimensional anchors (dimensions) around the perimeter of a circle [22]. Spring constants are utilized to represent relational values among points - one end of a spring is attached to a dimensional anchor, the other is attached to a data point. The values of each dimension are usually normalized to 0 to1 range. Each data point is displayed at the point where the sum of all spring forces equals zero. The position of a data point depends largely on the arrangement of dimensions around the circle.

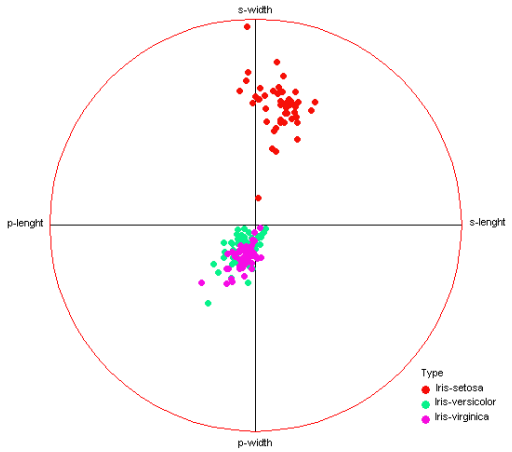

Figure 3.15: RadViz visualization of the Iris data set

3.15 PolyViz

The PolyViz visualization extends the RadViz method with each of the dimensions anchored as a line not just a point. Spring constants are utilized along the dimensional anchor (the line) that corresponds to all the values the dimension has. Each data point is positioned as in RadViz. The position of the point in the display depends as in RadViz on the arrangement of the dimensions. PolyViz provides more information than RadViz by giving insight into the distribution of the data for each dimension.

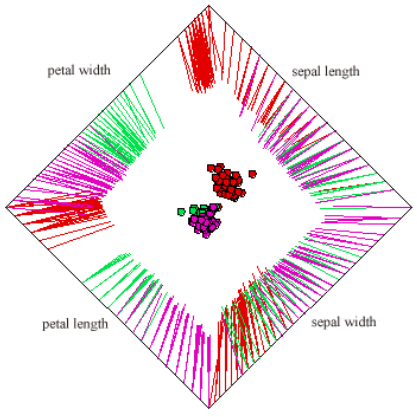

Figure 3.16: PolyViz visualization of the Iris data set

3.16 Principal Component and Principal Curves Analysis

Principal component analysis (PCA) is an analytic technique often coupled with a visual representation that identifies a lower dimensional space preserving variance (spread) in the data [24]. Numerous implementations exist, including neural networks [40], [9]. Self-organizing Maps (described below) can produce a PCA. PCA does not handle non-linearity well since it identifies linear subspaces. If the data set is non-linear then extensions must be used.

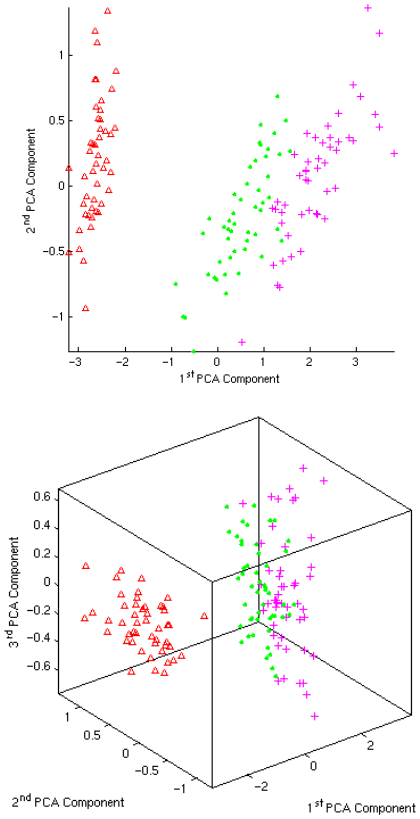

Figure 3.17: 2D and 3D principal component analysis of the Iris data set

Principal curves analysis [20] identifies smooth curves which represent the mean of all projected data points [39], [43], generalizing linear principal component analysis.

3.17 Grand Tour and Projection Pursuit

Projections of the data using a scatterplot matrix (or any other static representation of data) do not necessarily guarantee the best insight into the data. The most insight might be gained by some projection that allows a linear discrimination of two or more classes of data. In the grand tour method [4], sequences of 2D or 3D projections are displayed. The grand tour is most often applied to a single 2D or 3D scatterplot with the coordinate axes, moving through a sequence of projections that cover almost all of the *n*-dimensional space. In the classic grand tour a step and space-filling curve are defined. A plane is moved along this curve and the data projected.

The grand tour can be interpreted as an unguided exploratory projection pursuit. After a particular goal is identified, a guided projection pursuit is utilized. This produces projections of the data where a particular goal drives the projections, such as discrimination of two data classes. Linear projections are selected which attempt to identify and bring out the data deviating from normal distribution as much as possible. Projection pursuit can handle some non-linearity but it too is not general enough [16], [17]. Depending on the utilized display techniques and when a useful projection is found, it is not always clear how to extract useful information from the linear combinations of dimensions.

3.18 Kohonen Self-Organizing Maps (SOM)

The Self-Organizing Map (SOM) combines an analytic and graphical technique to group data in order to reduce its size. It is a summarization technique that attempts to reduce the complexity of the data set by displaying clusters of the data in a grid.

The self-organizing map (SOM) [29], [30], [31], [32] is a neural network algorithm that has been used to cluster in an unsupervised fashion and generate a visual representations of the clusters. SOMs both cluster and reduce the dimensionality of the data by projecting the clusters typically onto a 2-dimensional space. The Kohonen SOM is similar to a *k*-means clustering algorithm, extending it by providing a topological structure and placing similar objects in neighboring clusters. Numerous SOM algorithms and extensions have been developed in a multitude of fields which include engineering applications and neural networks (see [32], [38], [28], [51] and [53]).

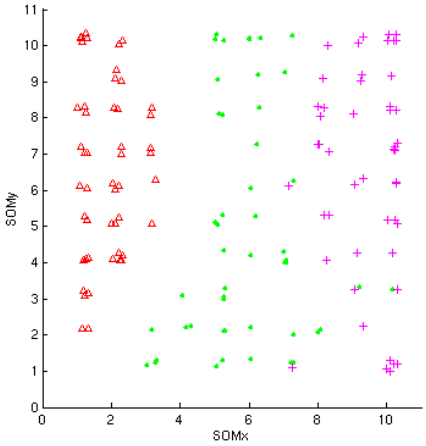

Figure 3.18: Self-organizing map of the Iris data set

3.19 Remarks

There are many systems incorporating a number of the techniques described above. Along with traditional static sorts of displays such as histograms, scatterplots, and parallel coordinates, most software packages provide interactive and dynamic querying of data. Currently, most PC or workstation-based tools are used to view multivariate data. These tools display 3D graphics on a traditional computer monitor. However, extensions using virtual reality devices offer the capability to display graphics in stereoscopic 3D, allowing the user to better perceive depth information. To date, there have been little commercial virtual reality data exploration environments.

A number of interactive techniques can also be provided to alter each of these visualizations. For example display transformations such as hyperbolic mappings and other distortion mappings can be applied to the resulting images to provide non-linear expansions of the data ([46], [18], [2], [34], [42]).

4 INTRINSIC DIMENSIONALITY

We now define precisely intrinsic dimensionality. The goal is to define metrics that identify how visualizations deal with n dimensions when displayed on the screen. The main problems are that points may overlap and that coordinate data may be lost in the projection. With probing one can get all the coordinate values of a single point.

We will consider two extreme cases: the set of n-dimensional unit vectors in $\Re^n$, where one coordinate (dimension) is 1 and all others 0, and a set of n-dimensional binary vectors, where each coordinate is 1 or 0.

4.1 Intrinsic Dimension

Given an n-dimensional space, the ***intrinsic dimension*** **(ID)** of a visualization is defined to be the largest k, $k \le n$, for which a set of k unit vectors in that n-dimensional space can be uniquely identified (perceived) in the visualization.

The intrinsic dimension of a 2D scatterplot is 2: the n unit vectors project to 3 points, (0, 0) and either (0, 1) or (1, 0), only two of which obviously come from unique points.

4.2 Intrinsic Record Ratio

Given an n-dimensional space, the ***intrinsic record ratio*** **(IRR)** of a visualization is defined to be k/n, where k is the largest value for which the set of 2^n binary vectors with all 0's and 1's in that n-dimensional space can be uniquely identified (perceived) in the visualization. It represents the percentage of records that can be distinguished, if one had reasonably distributed records. We can more precisely define this ratio using Monte Carlo techniques.

We have 2^n points (binary vectors) that represent values $[0,\ldots,(2^n\text{-}1)]$. If all are discernible then the intrinsic record ratio is 1. The 2^n binary vectors project to 4 points, (0, 0), (0, 1), (1,0) and (1, 1), and the intrinsic record ratio is $4/2^n$. As n gets large, the intrinsic record ratio decreases and approaches 0.

4.3 Intrinsic Coordinate Dimension

Given a n-dimensional space, the ***intrinsic coordinate dimension*** **(ICD)** of a visualization is defined to be the largest k, $k \le n$ for which k-coordinates of **any** vector in that n-dimensional space can be uniquely identified in the visualization.

The intrinsic dimension of a 2D scatterplot is 2 and its intrinsic coordinate dimension is 2. The intrinsic dimension for the 3D scatterplot is 3 whereas its intrinsic coordinate dimension is 2 (the projected point may come from several ones projecting to a line in 3D). Note that in many cases the intrinsic coordinate dimension is smaller than the intrinsic dimension since we know the vectors being examined in the first case whereas in the second we look to identify coordinates of any vector. Using rigid transformations such as rotate, pan and zoom, one can often increase the number of coordinates determined.

5 ANALYSIS

There are a number of factors that can affect the result. Color, size and shape of the points will make a difference. Perception is dependent on the viewer and the environment. Screen resolution and size have a significant bearing on the evaluation of intrinsic dimensions since the metric involves perception of unique points or values.

For more than a certain number of records or dimensions the screen/dot ratio becomes the limiting factor. In all visualizations it is either the linear dimensions of the screen (e.g., the axes in a scatterplot) or the surface dimensions (e.g., the points in a scatterplot) that limit the perception of data.

In order to avoid all these perceptual problems and issues we look to the first two definitions as theoretical. That is, what is the best that one could do having an arbitrarily large screen with infinite resolution. We look to the last definition to handle perceptual issues by permitting interaction to resolve size problems. The intrinsic dimension and coordinate dimension require being able to pull out coordinate interpretations whereas the intrinsic record ratio pulls out records. In all cases we assume that the selected color and shape of the points or tick marks is reasonable.

We now look at some examples of the visualizations described above and their intrinsic dimensions. In order to get a sense of the high dimensionality of the various visualizations we analyze the different visualizations into three classes as follows:

1. 10 - 100 intrinsic dimensions
2. 100 - 1000 intrinsic dimensions
3. 1000 or more intrinsic dimensions.

In this paper, we show several sample visualizations and their intrinsic properties for both 10- and 100-dimensional spaces (in some cases) and discuss the 1000 ones.

5.1 2D and 3D Scatterplot

We project 10 and 100 unit vectors in 2D and 3D to produce scatterplots of 10- and 100-dimensional space (Figure 5.1). The scatterplots for the 100-dimensional unit and binary vectors are identical to the scatterplots of the 10-dimensional unit and binary vectors, respectively.

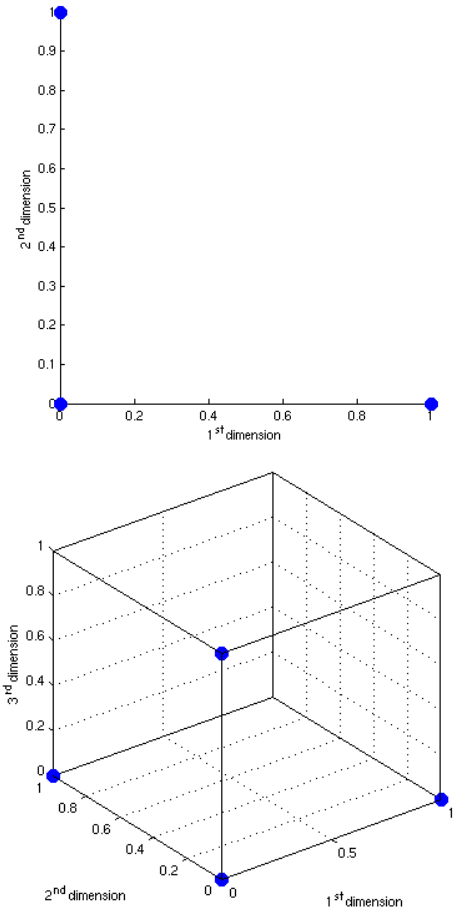

Figure 5.1: 2D and 3D scatterplots of the 10-dimensional and 100-dimensional unit vectors

Only two and three data records, respectively, are uniquely identifiable. Thus the intrinsic dimension is 2 for a 2D scatterplot and 3 for the 3D scatterplot.

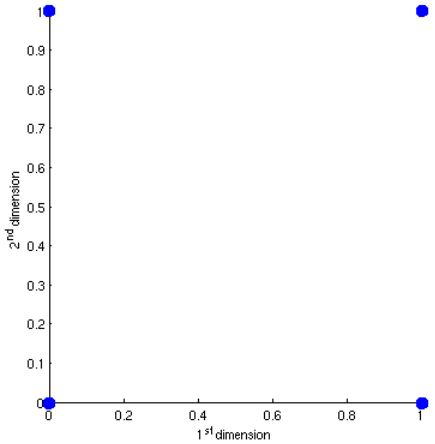

Figure 5.2: 2D scatterplot of the 10-dimensional binary vectors

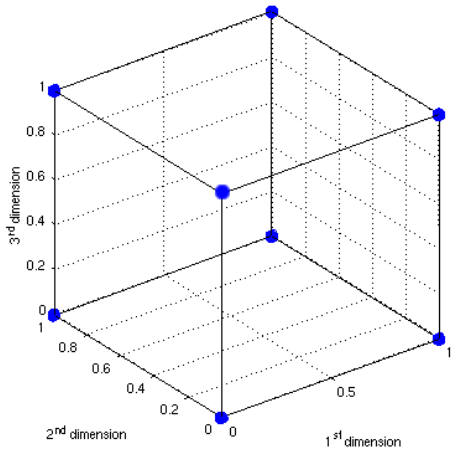

Figure 5.3: 3D scatterplot of the 10-dimensional binary vectors

The intrinsic record ratio for the 10-dimensional binary dataset is therefore 4/1024 = 1/256 as a 2D scatterplot and 8/1024 = 1/128 as a 3D scatterplot.

The intrinsic coordinate dimension for the 10-dimensional and 100-dimensional data sets is 2.

5.2 2D and 3D Sammon Plot

The Sammon plot representation of the 10- and 100-dimensional unit vectors is displayed in Figure 5.4 through Figure 5.6.

While the points are all visible, it is impossible to identify the values associated with them. Therefore, the intrinsic dimensions for both datasets are 0.

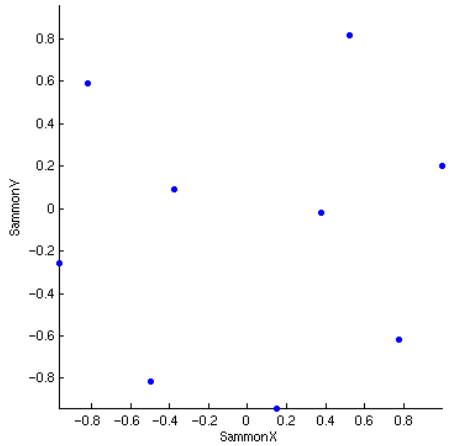

Figure 5.4: 2D Sammon plot of the 10-dimensional unit vector data set

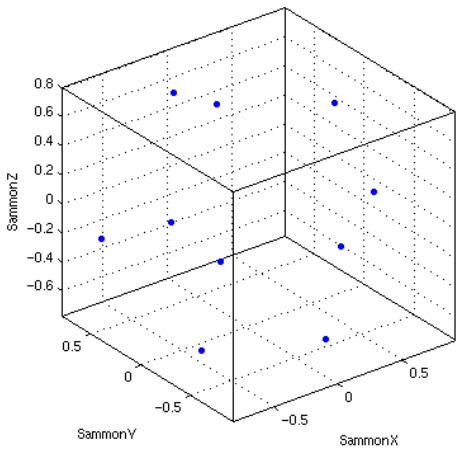

Figure 5.5: 3D Sammon plot of the 10-dimensional unit vector data set

In Figure 5.6 we display the 100 points (unit vectors). These are well distributed in the rendering space. The intrinsic dimension is 0 as points are not distinguishable. Here, too, the intrinsic record ratio is approximately 1, depending on the Sammon plot output and the number of points.

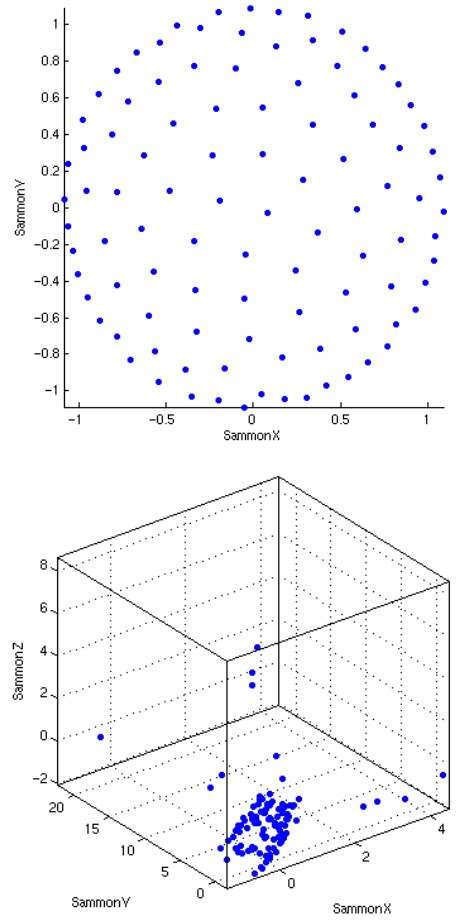

Figure 5.6: 2D and 3D Sammon plot of the 100-dimensional unit vector data set

Thus we find that the ID and IRR are dimension independent.

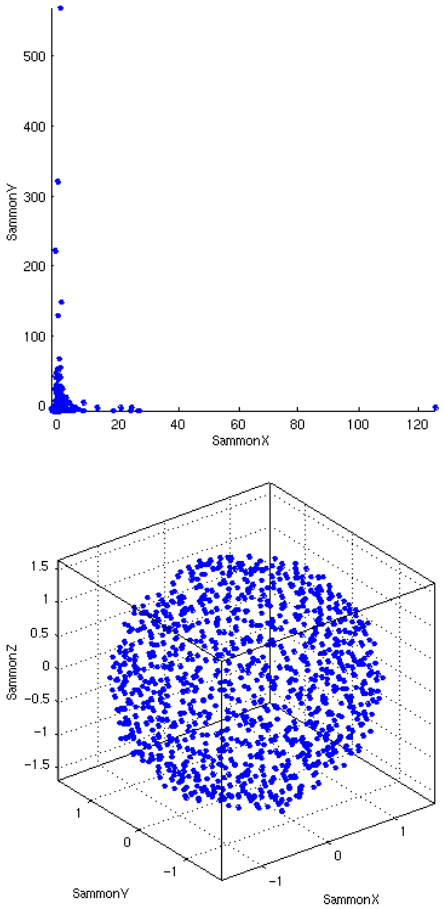

Figure 5.7: 2D and 3D Sammon plot of the 10-dimensional binary vectors

When we display the 1024 10-dimensional points (binary vectors) using the Sammon plot (Figure 5.7), we cannot easily determine the number of points in the display and so the intrinsic record ratio cannot be precisely determined visually. Estimate yields an intrinsic record ratio of ≈ 0.2 (2D) and ≈ 0.9 (3D). Note that repeated application of the Sammon plot algorithm may yield different intrinsic record ratios.

5.3 Parallel Coordinates

Parallel coordinates representing the 10- and the 100-dimensional unit vector datasets are displayed in Figure 5.8. The specific unit vector (polyline) is identifiable by the coordinate with value equal to 1. The intrinsic dimensions thus are respectively 10 and 100.

Since we assume that we are dealing with a perfect display of unlimited resolution, these limitations do not affect the intrinsic dimension but will effect its perception.

The intrinsic record ratio under perfect conditions (unlimited resolution) is 0, as it is not possible to identify a single unique point in the display and thus we cannot determine the number of points. The intrinsic coordinate dimension is equal to the number of dimensions in the data set, as we can uniquely identify each of the coordinate values.

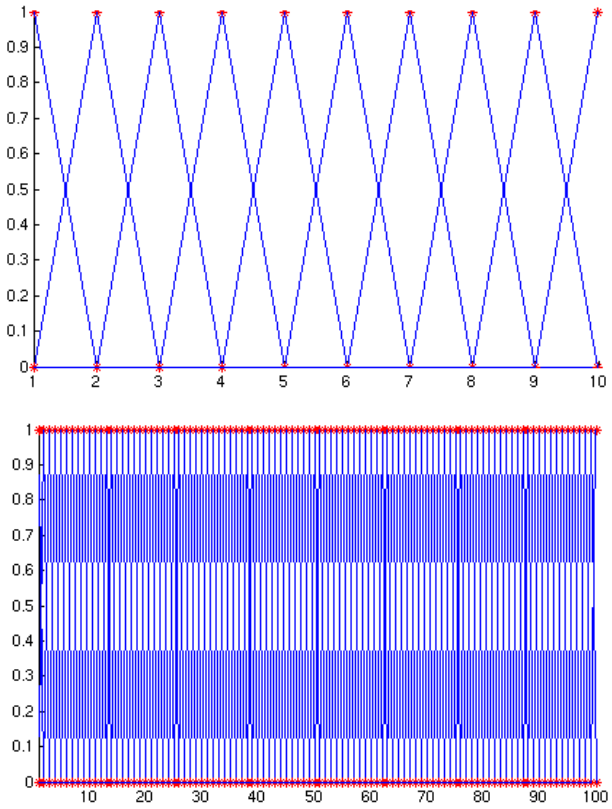

Figure 5.8: Parallel coordinates of the 10- and 100-dimensional unit vectors

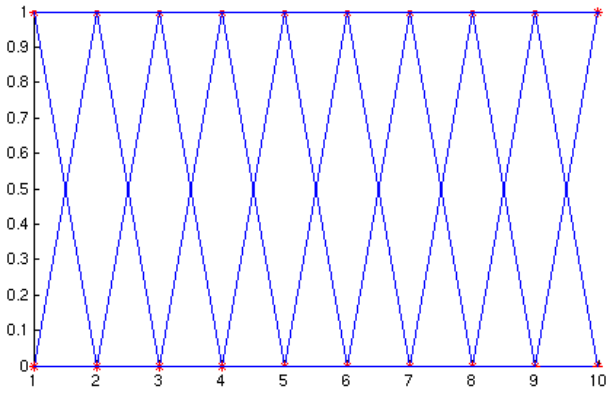

Figure 5.9: Parallel coordinates of the 10-dimensional binary vectors

5.4 Pixel Display

A pixel display of a 10-dimensional unit vector data set is shown in Figure 5.10.

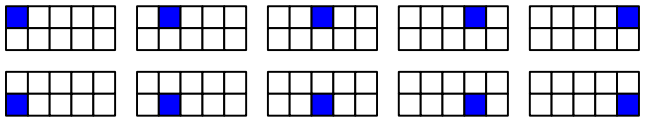

Figure 5.10: Pixel display of the 10-dimensional unit vector data

The intrinsic dimension is 10 as one can identify each coordinate directly from the multiple grids. For the 10-dimensional binary

vectors data set, the pixel display would consist of 10 rectangles, each containing 2^n cells. Each record is uniquely identifiable and the intrinsic record ratio is 1.0. The intrinsic coordinate dimension is not precisely determinable as the coordinate value is represented by a color, which depends on the color map as well as the viewer's perceptual capabilities.

5.5 RadViz

RadViz displays of the 10- and 100-dimensional unit vector data sets are shown in Figure 5.11, followed by a display of the 10-dimensional binary vectors data set (Figure 5.12).

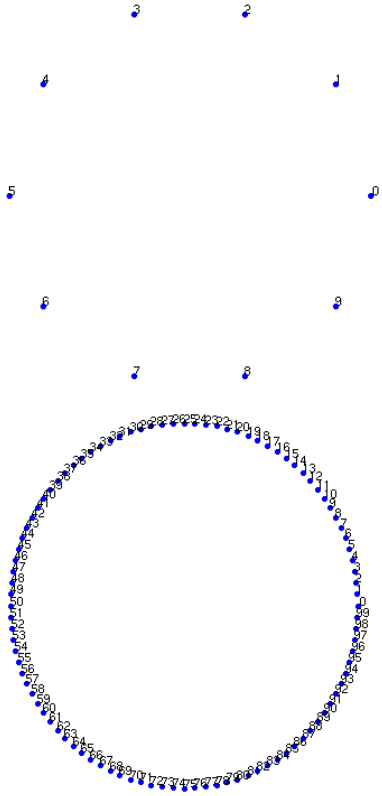

Figure 5.11: 10- and 100-dimensional unit vector data sets rendered using RadViz algorithm

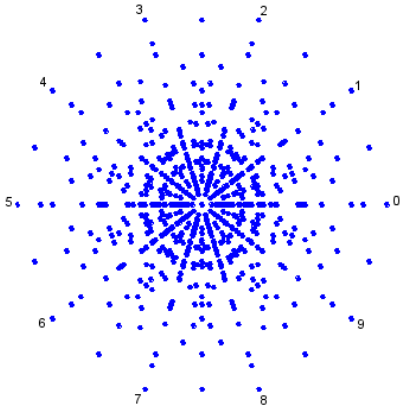

Figure 5.12: RadViz display of the 10-dimensional binary vectors data set

The intrinsic dimension is 10 and 100 respectively. The intrinsic record ratio is 1 and intrinsic coordinate dimension is not determinable in general if the point is not on the boundary of the circle.

5.6 PolyViz

PolyViz display of the 10-dimensional unit vector data sets is shown in Figure 5.13.

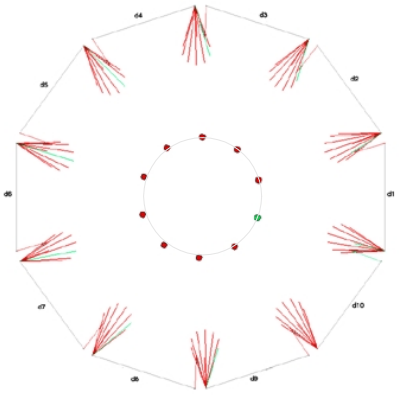

Figure 5.13: PolyViz display of the 10-dimensional unit vectors data set colored by the first dimension

The intrinsic dimension for this data set is 10 and for the 100 unit vectors it would be 100. The intrinsic record ratio is 1 and intrinsic coordinate dimension is *d* as each coordinate for a single record can be discerned.

5.7 Kohonen Self-Organizing Map (SOM)

Figure 5.14 and Figure 5.15 display a SOM of an arbitrary size for the 10- and 100-dimensional unit vector data sets.

The intrinsic dimension is 0.

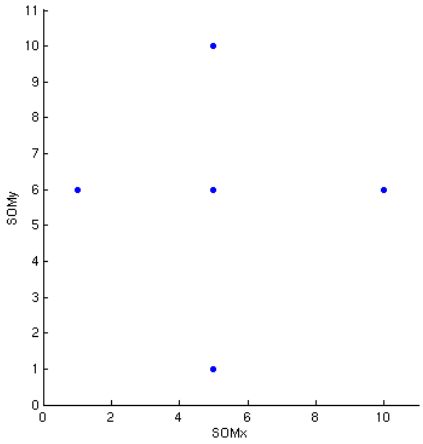

Figure 5.14: 10x10 SOM of the 10-dimensional unit vectors

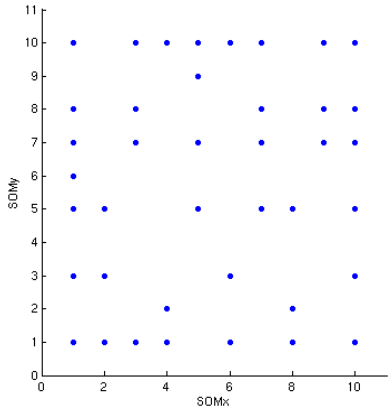

Figure 5.15: 10x10 SOM of the 100-dimensional unit vectors

Looking at Figure 5.16 we find that the intrinsic record ratio is 1.0 if the number of grids is large enough and that the intrinsic coordinate dimension is 0.

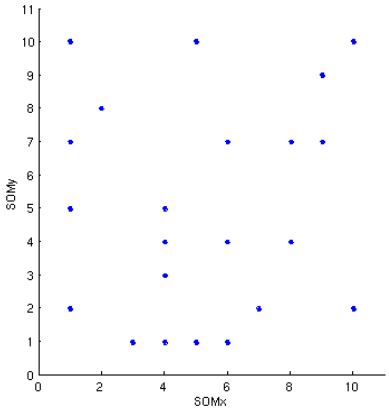

Figure 5.16: 10x10 SOM of the 10-dimensional binary vectors

6 SUMMARY

These visualizations are just a few of many possible examples. Table 1 provides a summary of intrinsic properties for visualizations discussed above. Both 10- and a 100-dimensional unit vector datasets were used for this task. Since an ideal display (of unlimited size and resolution) is used, there is no difference between the 10- and the 100-dimensional dataset.

Visualization	Intrinsic Dim.	Intrinsic Record Ratio	Intrinsic Coord. Dim.
2D Scatterplot	2	$4/2^d$	2
3D Scatterplot	3	$8/2^d$	2
2D Sammon Plot	0	≈ 0.2	0
3D Sammon Plot	0	≈ 0.9	0
Parallel Coord.	*d*	0.0	*d*
Pixel Display	*d*	1.0	*Indeterminate*
RadViz	*d*	1.0	*Indeterminate*
PolyViz	*d*	1.0	*d*
SOM	0	1.0	0

Table 1: A summary of intrinsic properties for selected visualizations
d = dimensionality of the data set

It is clear that some of the computations for the IRR require a precise determination of the number of distinguishable points, since this applies to both Sammon plots (and other visualization techniques not listed). Perceived separation determination with automatic computation with Monte Carlo techniques is necessary.

These definitions were used to begin to try to identify intrinsic metrics for high-dimensional visualizations. We see that several visualizations deal with high dimensions quite well. These include Pixel Displays, RadViz and PolyViz. Realistically, the limitations of screen resolution and color perception do have a bearing. These problems can be resolved through multiple linked visualizations or with interactions and tools that increase the intrinsic coordinate dimensions.

Acknowledgements: We thank Dr. Patrick Hoffman for his detailed critique and help in generating some of the images.

References

[1] C. Ahlberg, C. Williamson, and B. Shneiderman, "Dynamic Queries for Information Exploration: an Implementation and Evaluation," presented at ACM CHI, 1992.

[2] B. Alpern, "Hyperbox," presented at IEEE Visualization '91, San Diego, CA, 1991.

[3] D. F. Andrews, "Plots of High-Dimensional Data," *Biometrics*, vol. 29, pp. 125-136, 1972.

[4] D. Asimov, "The Grand Tour: A tool for Viewing Multidimensional Data," *DIAM Journal on Scientific and Statistical Computing*, vol. 61, pp. 128-143, 1985.

[5] J. Beddow, "Shape Coding of Multidimensional Data on a Microcomputer Display," presented at IEEE Visualization '90, San Francisco, CA, 1990.

[6] C. L. Bentley and M. O. Ward, "Animating Multidimensional Scaling to Visualize N-Dimensional Data Sets," presented at IEEE Information Visualization '96, San Francisco, CA, 1996.

[7] J. M. Chambers, W. S. Cleveland, B. Kleiner, and P. A. Tukey, *Graphical Methods for Data Analysis*. New York: Chapman and Hall, 1976.

[8] H. Chernoff, "The Use of Faces to Represent Points in k-Dimensional Space Graphically," *Journal of the American Statistical Association*, vol. 68, pp. 361-368, 1973.

[9] A. Cichocki and R. Unbehauen, *Neural Networks for Optimization and Signal Processing*. Chichester, England: John Wiley, 1993.

[10] W. S. Cleveland and M. E. McGill, *Dynamic Graphics for Statistics*. Belmont, CA: Wadsworth Advanced Books and Software, 1988.

[11] J. de Leeuw and W. Heiser, "Theory of Multidimensional Scaling," in *Handbook of Statistics*, vol. 2, P. R. Krishnaiah and L. N. Kanal, Eds. Amsterdam: North-Holland Publishing, 1982, pp. 285-316.

[12] R. F. Erbacher, D. Gonthier, and H. Levkowitz, "The Color Icon: A New Design and a Parallel Implementation," presented at SPIE '95 Conference on Visual Data Exploration and Analysis II, San Jose, CA, 1995.

[13] U. Fayyad, G. Grinstein, and A. Wierse, *Information Visualization in Data Mining and Knowledge Discovery*, 1st ed: Morgan-Kaufmann Publishers, 2001.

[14] S. Feiner and C. Beshers, "Worlds Within Worlds: Metaphors for Exploring N-Dimensional Virtual Worlds," presented at UIST '90 (ACM Symp. on User Interface Software and Technology), Snowbird, UT, 1990.

[15] R. A. Fisher, "The Use of Multiple Measurements in Taxonomic Problems," *Annals of Eugenics*, vol. 7, pp. 179-188, 1936.

[16] J. H. Friedman, "Exploratory Projection Pursuit," *Journal of the American Statistical Association*, vol. 82, pp. 249-266, 1987.

[17] J. H. Friedman and J. W. Tukey, "A Projection Pursuit Algorithm for Exploratory Data Analysis," *IEEE Transactions on Computers*, vol. C, pp. 881--889, 1974.

[18] G. Furnas, "Generalized Fisheye Views," presented at Human factors in Computing Systems ACM CHI '86, Boston, MA, 1986.

[19] G. Grinstein, P. E. Hoffman, S. Laskowski, and R. Pickett, "Benchmark Development for the Evaluation of Visualization for Data Mining," in *Information Visualization in Data Mining and Knowledge Discovery*, *The Morgan Kaufmann Series in Data Managament Systems*, U. Fayyad, G. Grinstein, and A. Wierse, Eds., 1st ed: Morgan-Kaufmann Publishers, 2001.

[20] T. Hastie and W. Stuetzle, "Principal Curves," *Journal of the American Statistical Association*, vol. 84, pp. 502-516, 1989.

[21] M. A. Hearst, "Tilebars: Visualization of Term Distribution Information in Full Text Information Access," presented at ACM CHI '95 Human Factors in Computing Systems, Denver, CO, 1995.

[22] P. Hoffman and G. Grinstein, "Dimensional Anchors: A Graphic Primitive for Multidimensional Multivariate Information Visualizations," presented at NPIV '99 (Workshop on New Paradigmsn in Information Visualization and Manipulation), 1999.

[23] P. E. Hoffman and G. Grinstein, "Multidimensional Information Visualizations for Data Mining with Applications for Machine Learning Classifiers," in *Information Visualization in Data Mining and Knowledge Discovery*, *The Morgan Kaufmann Series in Data Managament Systems*, U. Fayyad, G. Grinstein, and A. Wierse, Eds., 1st ed: Morgan-Kaufmann Publishers, 2001.

[24] H. Hotelling, "Analysis of a Complex of Statistical Variables into Principal Components," *Journal of Educational Psychology*, vol. 24, pp. 417-441, 498-520, 1933.

[25] A. Inselberg, "The Plane with Parallel Coordinates," *Special Issue on Computational Geometry: The Visual Computer*, vol. 1, pp. 69-91, 1985.

[26] A. Inselberg and B. Dimsdale, "Parallel Coordinates for Visualizing Multidimensional Geometry," presented at Computer Graphics International '87, Tokyo, 1987.

[27] D. A. Keim and H.-P. Kriegel, "VisDB: Database Exploration Using Multidimensional Visualization," *IEEE Computer Graphics and Applications*, vol. 14, pp. 40-49, 1994.

[28] S. Klinke and J. Grassmann, *Visualization and Implementation of Feedforward Neural Networks via Multidimensional Scaling in XploRe*, 1996.

[29] T. Kohonen, "Self-Organized Formation of Topologically Correct Feature Maps," *Biological Cybernetics*, vol. 43, pp. 59-69, 1982.

[30] T. Kohonen, "The Self-Organizing Map," presented at IEEE, 1990.

[31] T. Kohonen, *Self-Organizing Maps*. Berlin: Springer, 1995.

[32] T. Kohonen, E. Oja, O. Simula, A. Visa, and J. Kangas, "Engineering Applications of the Self-Organizing Map," presented at IEEE, 1996.

[33] J. B. Kruskal and M. Wish, *Multidimensional Scaling*: Sage Publications, 1978.

[34] J. Lamping and R. Rao, "Laying out and Visualizing Large Trees Using a Hyperbolic Space," presented at UIST '94, 1994.

[35] H. Levkowitz, "Color Icons: Merging Color and Texture Perception for Integrated Visualization of Multiple Parameters," presented at IEEE Visualization '91, 1991.

[36] H. Lohninger, "INSPECT, a Program System to Visualize and Interpret Chemical Data," *Chemometrics and Intelligent Laboratory Systems*, vol. 22, pp. 147-153, 1994.

[37] R. S. Michalski, "A Planar Geometric Model for Representing Multidimensional Discrete Spaces and Multiple-Valued Logic Functions," University of Illinois at Urbana-Champaign, Technical Report UIUCDCS-R-78-897, 1978.

[38] N. J. S. Mørch, U. Kjems, L. K. Hansen, C. Svarer, I. Law, B. Lautrup, S. Strother, and K. Rehm, "Visualization of Neural Networks Using Saliency Maps," presented at ICCN '95, 1995.

[39] F. Mulier and V. Chrkassky, "Self-organization as an Iterative Kernel Smoothing Process," *Neural Computation*, vol. 7, pp. 1165-1177, 1995.

[40] E. Oja, *Subspace Methods of Pattern Recognition*. Letchworth, England: Research Studies Press, 1983.

[41] R. M. Pickett and G. G. Grinstein, "Iconographic Displays for Visualizing Multidimensional Data," presented at IEEE Conference on Systems, Man and Cybernetics, Beijing and Shenyang, People's Republic of China, 1988.

[42] R. Rao and S. K. Card, "The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus+Context Visualization for Tabular Information," presented at ACM CHI '94, Boston, MA, 1994.

[43] H. Ritter, T. Martinetz, and K. Schulten, *Neural Computation and Self-Organizing Maps: An Introduction*. Reading, MA: Addison-Wesley, 1992.

[44] J. W. J. Sammon, "A Nonlinear Mapping for Data Structure Analysis," *IEEE Transactions on Computers*, vol. 18, pp. 401-409, 1969.

[45] B. Shneiderman, "The Eyes Have It: A Task by Data Type Taxonomy of Information Visualization," presented at IEEE Symposium on Visual Languages '96, Boulder, CO, 1996.

[46] R. Spence, "Data Base Navigation: An Office Environment for the Professional," *Behaviour and Information Technology*, vol. 1, pp. 43-54, 1982.

[47] R. Spence, L. Tweedie, H. Dawkes, and H. Su, "Visualization for Functional Design," presented at IEEE Information Visualization Symposium '95, 1995.

[48] W. S. Torgerson, "Multidimensional Scaling: Theory and Method," *Psychometrika*, vol. 17, pp. 401-419, 1952.

[49] J. W. Tukey, *Exploratory Data Analysis*. Reading, MA: Addison-Wesley, MA, 1977.
[50] J. J. van Wijk and R. van Liere, "HyperSlice," presented at IEEE Visualization, San Jose, CA, 1993.
[51] L. G. Vuurpijl and T. Schouten, "Convis, a distributed environment for control and visualization of neural networks," presented at International Conference on Artificial Neural Networks, Amsterdam, 1993.
[52] M. O. Ward, J. LeBlanc, and R. Tipnis, "N-Land: A Graphical Tool for Exploring N-Dimensional Data," presented at Computer Graphics International Conference, Melbourne, 1994.
[53] P. Wilke, "Visualization of Neural Networks using NeuroGraph," presented at IFIP WG 3.2 Working Conference on Visualization in Scientific Computing: Uses in University Education, Irvine, CA, 1993.
[54] J. A. Wise, J. J. Thomas, et al, "Visualizing the Non-Visual: Spatial Analysis and Interaction with Information from Text Docments," presented at IEEE Information Visualization'95, Atlanta, GA, 1995.
[55] M. Wish and J. D. Carroll, "Multidimensional Scaling and its Applications," in *Handbook of Statistics*, vol. 2, P. R. Krishnaiah and L. N. Kanal, Eds. Amsterdam: North-Holland Publishing, 1982, pp. 317-345.
[56] F. W. Young, "Multidimensional Scaling," in *Encyclopedia of Statistical Sciences*, vol. 5, S. Kotz and N. L. Johnson, Eds. New York: Wiley, 1985, pp. 649-659.
[57] G. Young and A. S. Householder, "Discussion of a Set of Points in Terms of Their Mutual Distances," *Psychometrika*, vol. 3, pp. 19-22, 1938.