

Targeted Projection Pursuit for Interactive Exploration of High-Dimensional Data Sets

Joe Faith

Northumbria University, Newcastle, UK

joe.faith@unn.ac.uk

Abstract

High-dimensional data is, by its nature, difficult to visualise. Many current techniques involve reducing the dimensionality of the data, which results in a loss of information. Targeted Projection Pursuit is a novel method for visualising high-dimensional datasets which allows the user to interactively explore the space of possible views to find those that meet their requirements. A prototype tool that utilises this method is introduced, and is shown to allow users to explore data through an interface that is transparent and efficient. The tool and underlying technique are general purpose – applicable to any high-dimensional numeric data, and supporting a wide range of exploratory data analysis activities – but are evaluated on three particular tasks using gene expression data: identifying discriminatory genes, visualising diagnostic classes, and detecting misdiagnosed samples. It is found to perform well in comparison with standard techniques.

Keywords—Information Visualisation; Dimensionality Reduction; Multi-Dimensional Scaling; Projection Pursuit

1 Introduction

The problem of visualising high-dimensional data is primarily one of *dimensionality reduction* (DR); *i.e.* representing the data in a low- (typically two- or three-) dimensional space such that it can be presented visually to the user. Many DR techniques are available, and most have been applied to the problem of visualisation, including Multi-Dimensional Scaling (MDS), Sammon Mapping, Self-Organising Maps (SOM), iconographic and glyph-based approaches, *etc* [1].

Among these are techniques based on linear projections of the data, such as Principal Components Analysis (PCA), Singular Value Decomposition (SVD), and Projection Pursuit. Friedman and Tukey introduced the term *projection pursuit* to describe the process of finding interesting linear projections by optimizing some function (the *projection pursuit index*)[15]. The definition of what makes a projection ‘interesting’ depends on the projection pursuit index and on the application or purpose. For example, Lee

et al [18] discuss a projection pursuit index that measures how well each projection shows the separation of classes in the data.

The advantage of techniques based on linear projections is that they not only may show an informative view of the data, but the weights of the projection itself may include useful information. For example, if one particular projection is found to show a clear separation between classes in the data, then the most significant weights in the underlying projection will indicate which variables in the original data were the best discriminators for those classes.

However, unless the data contains a great deal of redundancy, any dimension reduction process involves the loss of information. The resulting view may emphasise one aspect of the original data; but other aspects are inevitably lost as the space of possibilities is reduced. Therefore one possibility is to present a range of different plots, each involving a different pair of coordinates. The scatter-plot matrix, for example, shows the data pictured against all possible pairs of coordinates, resulting in $\frac{1}{2}n(n-1)$ plots, where n is the dimensionality of the original data. Another approach is to allow the user to choose which coordinate system to use, but to guide their choice by providing descriptive statistics of the resulting views [2]. One dynamic alternative to such static views is Asimov’s Grand Tour [9] – described as an attempt to look at the data ‘from all possible angles’. A Grand Tour is a video sequence in which each frame shows the result of a single projection of the data, with the sequence as a whole including all possible projection planes. However, the Grand Tour replaces the quality of projection pursuit with quantity: a grand tour in high dimensional space may be long and mostly uninformative.

Ideally we would have some way of allowing user to guide the tour, to use their perception of the data to find projections of interest. Cook and Buja [10][11] proposed and implemented an interface that allows the user to not only pause and rewind a given Grand Tour, but also to amend the resulting view by controlling the input from each dimension independently. The problem is that projec-

tion component manipulation is an *opaque* interface in the sense that it is rarely possible for the user to anticipate the effect of their actions. Where the user has strong intuitions about the nature of the structure of interest in data, and its relationship with the underlying coordinate system, then it may be possible for them to determine how best to use component-based controls to reveal structure in the data more clearly. In other words, once the user knows what they are looking for then such an interface will help them find it, but it is unsuited to true exploration of the data. The user has n controls to manipulate (one for each dimension of the original data set), the effect of each will be unknown and which will have unpredictable effects in combination. The user can do little more than random search – which has its place, but is of little use when faced with a truly large dimensionality set.

We present an alternative technique for interactively exploring the space of possible linear projections of a data set that we call Targeted Projection Pursuit (TPP). The basis of TPP is that the user manipulates their view of the data directly, rather than manipulate the projection that produces that view. A linear projection is then found that produces a view of the data that best matches the target view produced by the user.

In TPP, the user manipulates a two-dimensional view of the data using mouse actions to ‘drag and drop’ the data points; the tool then finds projections of the data that best matches the users requirements. A prototype implementation of this tool has been built, and we have evaluated it on a number of data analysis tasks using publicly available data sets including class separation visualisation, misclassification detection and feature detection. The technique is found to outperform other standard techniques at class visualisation, and to be efficient at discriminatory feature identification and misclassification detection.

The next section discusses how the new technique works and considers some of the benefits of this approach; Section 3 discusses the algorithm required to implement the technique; and Section 4 discusses the prototype tool. Section 5 discusses the results of our initial evaluation on a range of tasks using reference data sets comprising gene expression data.

The prototype tool, along with example data sets, is publicly available from the associated web-site¹.

2 Targeted Projection Pursuit Tool

We can motivate the principle behind the TPP technique with the following example. Suppose that an experimenter is conducting an initial inspection of a set of results using a scatter plot matrix or PCA. Suppose further that they find that the data seems to fall into distinct clusters, albeit with

some outliers, or that there seems to be some suggestion of a curvilinear relationship in the data. The natural question would be whether an alternative view of the data could be found in which the perceived regularity were more clearly defined, or whether it were just the result of noise.

Faced with this situation, a natural impulse would be to try to ‘grab’ those points that fail to fit the perceived pattern and try to move them into place. TPP allows the user to do just this. If a projection can be found in which the changes requested by the user can be found, then the resulting view is displayed. If not, then the points will fail to move. The overall process is thus one of hypothesis-formation and testing: by attempting to move some points, the experimenter is suggesting a hypothesis about the structure of the data; if the data fits the hypothesis, then the result is shown.

This process is illustrated in Figure 1. The user is initially presented with an arbitrary two-dimensional view of the data, such as that produced using PCA. Suppose the user can discern some kind of pattern, for example if there appears to be some clustering in the data (Figure 1a). If this is the case the user would hypothesise that the clustering is due to a genuine regularity in the data and that any outliers are simply a product of the particular projection – for example, due to the inclusion of a component of the data comprised mostly of noise. In this case the user would select an outlier and attempt to drag it into the nearest cluster (Figure 1b). The tool then attempts to find a projection of the data that best matches this revised view, and redisplay the data. If such a view can be found then it will be displayed and the clusters will ‘fall into place’ (Figure 1c), for example by removing the contribution of the noisy component. Otherwise the partial clusters will be revealed to have been solely been an artefact of the initial projection (Figure 1d).

As well as manipulating the projections to find clusters, the user can also try dragging and dropping points into curvilinear relationships, or linearly separable regions. And, rather than single points, the user can select a region including a cluster of points, to fix or move. Alternatively, if the data is already classified into known classes then the tool may be used to find low-dimensional views in which those classes are most clearly shown, and to identify outliers.

The principal advantage of such an interface is that it is transparent, in the sense that the response of the system is intuitive and predictable. If the user spots a partial pattern then they manipulate the elements of that pattern directly, rather than controls whose effects on the pattern are unpredictable. In addition, the interface takes advantage of three key strengths of the human visual system:

¹<http://computing.unn.ac.uk/staff/CGJF1/tpp/tool.html>

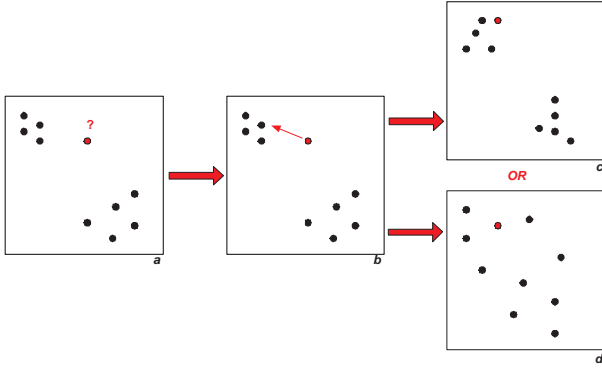


Figure 1: The use of targeted projection pursuit for interactive data exploration. *a* An initial view of the data with two partial clusters and an outlier. *b* The user hypothesises that the outlier is part of the upper cluster and drags it into place. *c* If the data supports such a clustering then the tool finds a view of the data that matches the hypothesis. *d* The data does not support the hypothesis and moving the point disrupts the partial clusters.

- We are efficient at spotting patterns in data when presented as two-dimensional images. We are especially good at spotting partial or obscured patterns, ignoring noise, and disregarding outliers: tasks that pattern recognition algorithms struggle with. In TPP it is the human user, rather than an algorithm, that spots partial patterns in the data; while the computer acts as a ‘hypothesis tester’ to see how well a partial pattern may be completed.
- We are efficient at recognising structure in correlated movements, as illustrated in ‘point light experiments’ [20][21], even where that movement is obscured by distracters. This is exploited in TPP since those data points that are mostly closely correlated will tend to move together compared to the behaviour of outliers or misclassified samples.
- We are efficient at detecting the effect of our actions on patterned stimuli. The effect of this in TPP is that, when causing a group of points to move we can detect which other points start to move simultaneously, and hence will be correlated with those points we originally selected.

Thus TPP uses the full power of the human visual system to do what it is best at – *i.e.* spotting patterns – while the computer based tool is left to do ‘dumb’ linear algebra.

3 Targeted Projection Pursuit Algorithm

Conventional projection pursuit proceeds by searching the space of all possible projections to find that which maximises an index that measures the quality of each resulting view. Targeted projection pursuit, on the other hand, proceeds by allowing the user to define an ideal target view

of the data, and then finding a projection that best approximates that target.

Suppose X is an $n \times p$ matrix that describes the value of p variables in n samples and T is a $n \times 2$ matrix that describes a two-dimensional target view of those samples. We require the $p \times 2$ projection matrix, P , that minimises the size of the difference between the view resulting from this projection of the data and our target:

$$\min \| T - XP \| \quad (1)$$

where $\| \cdot \|$ denotes the Euclidean norm.

A solution to Equation (1) may be found by training a single layer perceptron with p input units and two linear output units (see Figure 2). Each of the n data rows in X are presented in turn, and standard back-propagation is used to train the network to produce the corresponding row of T in response with the total least-squares error calculated as usual. (In other words, the entire training set is used as the testing set.) Once converged, the network can be used to transform data from the original gene-space to a two-dimensional view, with the weight of the connection from the i^{th} input neuron to the j^{th} output neuron corresponding to the value of the projection matrix P_{ij} .

In the implementation discussed below, the network is considered to have converged if one of the following two conditions is met:

1. The mean error for the entire data set is less than a defined constant parameter.
2. The rate of change of mean error between training cycles is less than a defined constant.

This definition of convergence is used to allow for the situation in which the user attempts to ‘pursue’ a target that

is inconsistent with the data: in this case the network may converge whilst the error is still large.

The initial training of a network to an arbitrary target T may be time-consuming. However, once trained, the resulting network may then be incrementally trained to ‘pursue’ subsequent perturbations in the target, rather than having to re-train a new network *ab initio*. This technique is used below (Section 4) to provide an efficient and responsive interface.

(An alternative solution to Equation (1) based on Procrustes methods is discussed in [13].)

4 Tool Implementation

A prototype TPP tool was implemented using Java, incorporating data-handling functionality from Weka [19] and neural net functionality from Gurel [16]. A snapshot of the interface in use is shown in Figure 5. Although it is currently at the prototype stage, the tool supports the following functionality.

First, the user loads a classified data sets from a standard .arff format file [19], and an initial view of the data is presented using the first two principal components (though the particular choice of initial view is unimportant). Each class in the data set is represented by points of a particular colour. Subsets of data points can be selected using a rectangular ‘rubber band’ and then dragged in the X and Y directions; the closest possible projection is then found dynamically and the resulting data positions redisplayed. Mouse drags can also be used to contract the area of the selected points in an attempt to ‘bunch’ a putative cluster; or to expand an area in an attempt to differentiate points that are not clearly distinguished.

The resulting projection is also displayed in the form of a table, with the values of the X and Y components shown for each variable in the original data, along with the significance of each variable (*i.e.* the sum of squares of each component). This table can be used in two ways. First it can be used to find which variables are most significant in producing the current view of the data – this is made easier by including the ability to re-order the projection table by significance, by variable name, or by each component. The table is used in this way in the gene identification task discussed in Section 5.2.

Alternatively, the user may input values for the X and Y components directly into the table, with the view of the data being updated dynamically to reflect the new projection. This may be used to produce scatter plots of the data against particular pairs of variables (by setting the components corresponding to those variables to 1, and all other components to 0), or by emphasising the effect of one particular variable (by setting the corresponding components

to a suitably high number).

On a standard desktop PC², the tool is able to calculate projections of 100 data points of 200 dimensions and display them at approximately 5 frames per second.

5 Experimental Evaluation

The tool is designed to be a general purpose method for exploring and visualising any high dimensional data. However, in order to better test the utility of the tool in a concrete application we have concentrated on evaluating its use on gene expression data:

cDNA and oligonucleotide microarray technology and genome sequencing have made it possible to measure gene expression levels on a genomic scale [4]. Microarrays currently measure expression levels across thousands of genes, and this number will rise. Parallel array technologies, such as protein arrays, tissue arrays, and combinatorial chemistry arrays will generate similarly high-dimensional data. Thus analysis of this data requires mathematical tools that are adaptable to both the large quantity and high dimensionality of data, while reducing its complexity to make it comprehensible [5]. Therefore such data seems a highly suitable candidate for analysis using TPP.

Thus far the TPP tool has been evaluated on three tasks: finding views of classified samples, identifying discriminatory genes associated with particular diagnostic classes, and detecting misclassified samples. (The data sets for these experiments can be found along with the prototype tool on the associated web site.)

5.1 Sample Classification Visualisation

In this task the user was presented with views of gene expression data sets in which each sample is of a known diagnostic class. They were then invited to use the tool to find views that best show the separation between classes. The resulting two-dimensional view of the data was then tested using standard statistical measures of class separation, and compared with standard DR techniques, including Sammon Mapping [3]. One advantage of this task in evaluating the tool is that it allows a direct, objective, and quantifiable comparison between techniques; and in practice the TPP tool was able to find views of gene expression data that showed a much clearer separation between classes than standard methods (for experimental procedures and detailed results see [14]).

For example, Figures 3 and 4 show two views of the same data set, one produced using Sammon Mapping and one produced using the TPP tool. This dataset comprises cDNA microarray analysis of small, round blue cell childhood tumors (SRBCT), including neuroblastoma (NB), rhabdomyosarcoma (RMS), Burkitt Lymphoma (BL; a subset of non-Hodgkin lymphoma) and members of Ew-

²1.8GHz Pentium 4 CPU, 800MB RAM, Windows XP OS.

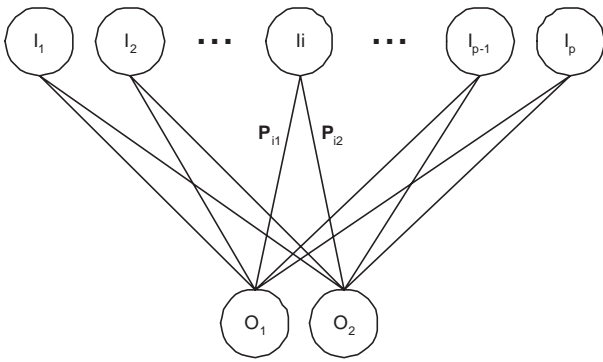


Figure 2: Schematic of a Single Layer Perceptron for projecting p -dimensional data presented to the top input layer (I_i), to a 2-dimensional view output at the bottom layer (O_1, O_2). The connection weight (P_{ij}) describes the weight given to the i^{th} variable in the projection onto the j^{th} ordinate in the view.

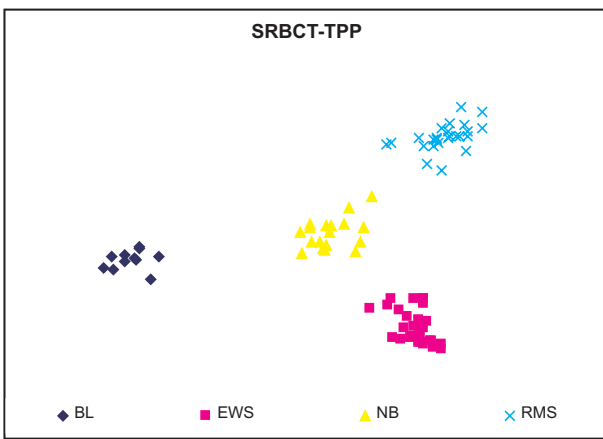


Figure 3: Two-dimensional view of SRBCT data set produced by TPP tool, showing clear separation between all classes.

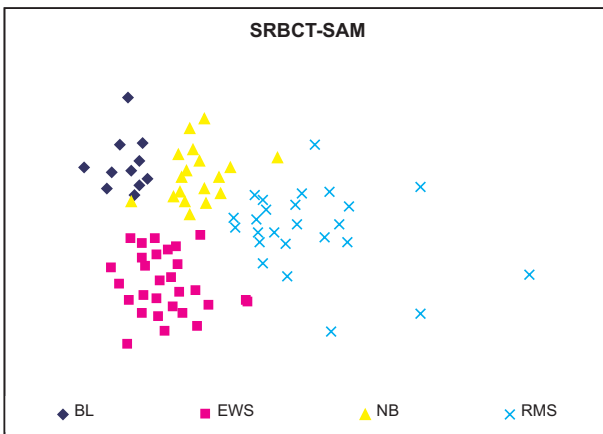


Figure 4: Two-dimensional view of SRBCT data set produced using Sammon Mapping showing one aspect of the 'curse of dimensionality': the small variance between points in high dimensional space produces a reduced view with little 'bunching'.

ings family of tumors (EWS). Expression levels from 6567 genes for 83 samples were taken [17].

The view produced by the TPP method shows a clear separation between all classes. However the view produced using Sammon Mapping shows one aspect of the ‘curse of dimensionality’: the small variance between points in high dimensional space means that a mapping in lower dimensional space that reproduces these inter- point differences will result in a view with very little difference between intra-class and inter-class point distances. In other words there is little ‘bunching’ or clear separation between classes.

It is interesting to note that the view of the data produced by human-driven search was better (in the sense of separating sample classes) than that produced by a conventional computer-driven projection pursuit algorithm. In other words, a human was more effective at searching the extremely large space of all possible projections than an automatic algorithm (in this case, gradient descent combined with simulated annealing); a result which reinforces the value of the ‘division of labour’ between human user and machine discussed in Section 2

5.2 Gene Identification

As discussed in Section 1, a fundamental advantage of using linear projections for visualisation compared to, for example, MDS, is that they define a transform that can be applied to any point in data- space. In particular, the projection contains information about the respective significance of each variable in the data, and how they can be best combined to perform functions such as classification and feature selection. The potential of the TPP technique in this regard was tested by using the tool to identify particular genes associated with particular diagnostic classes. This is achieved by finding a view that separates the class in question from all other samples, and then inspecting the resulting projection to see which genes are most significant.

For example, using the SRBCT data set discussed above the user can separate the samples of Burkitt’s lymphoma from the other classes. Those genes that are most associated with this classification of the data are then found to have the highest weighting in the resulting projection – CD83 in this case: a result which is consistent with the biological literature [7].

Another example of gene identification can be seen using gene expression data from the 60 cell lines from the National Cancer Institute’s anticancer drug screen [22]. It consists of 8 different tissue types where cancer was found, including nine samples of ovarian cancer. 9703 cDNA sequences were used. A Desmoplakin gene known to be associated with ovarian cancer can be identified by selecting the ovarian cancer samples [6]. A screenshot of the tool

used in this way is given in Figure 5.

5.3 Misclassification Identification

The use of TPP to detect possible cases of misdiagnosis – *i.e.* the misclassification of samples – was tested in the following task. The user is presented with a data set in which a small proportion of the samples have had their classes changed at random. The user is then asked to determine which of the samples are misclassified by observing the movements of which data points are uncorrelated with the others of their class – or are correlated with another class. The dataset used in this task was the result of a study of gene expression in two types of acute leukemia: acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML) [8]. The samples consist of 38 cases of B-cell ALL, 9 cases of T-cell ALL, and 25 cases of AML with the expression levels of 7219 genes measured.

Ten replica data sets were prepared, each containing three random misclassifications. The user was presented with a random initial view of each set in order to prevent them using cues from the initial position of points to spot misclassifications. Despite this the user accurately identified 27 of the 30 misclassifications (90% correct).

5.4 Data Preparation

For the purposes of these evaluations all data sets were reduced to 50 dimensions by selecting the 50 most discriminatory genes on the basis of Between-Group to Within-Group Sum of Squares [12]. The purpose of this was to reduce the degree of independence in the data by eliminating ‘noisy’ variables, as well as to make the TPP algorithm computationally viable.

One of the problems with high dimensional data sets is that where $p \gg n$, *i.e.* the number of samples is much less than the number of variables, then the degree of independence means that, by carefully selecting which variables are to be used, the data can be made to fit any hypothesis. This is a particular problem for any visualisation process based on projection pursuit.

Nonetheless, the tool is capable of working with data with a greater number of dimensions than 50. The misclassification task described above, for example, has been conducted with data from 500 genes and, although performance is sluggish, the misclassified samples can still be effectively identified.

6 Conclusions

The visualisation of high-dimensional data is limited by the fact that the human visual system only works in two or three dimensions. Visualisation research has produced a number of responses to this. One is to carefully select and preserve some aspect of the information contained in the higher data space and present it in the restricted visualisation space, though inevitably this results in the loss of other, potentially useful, information. Another approach

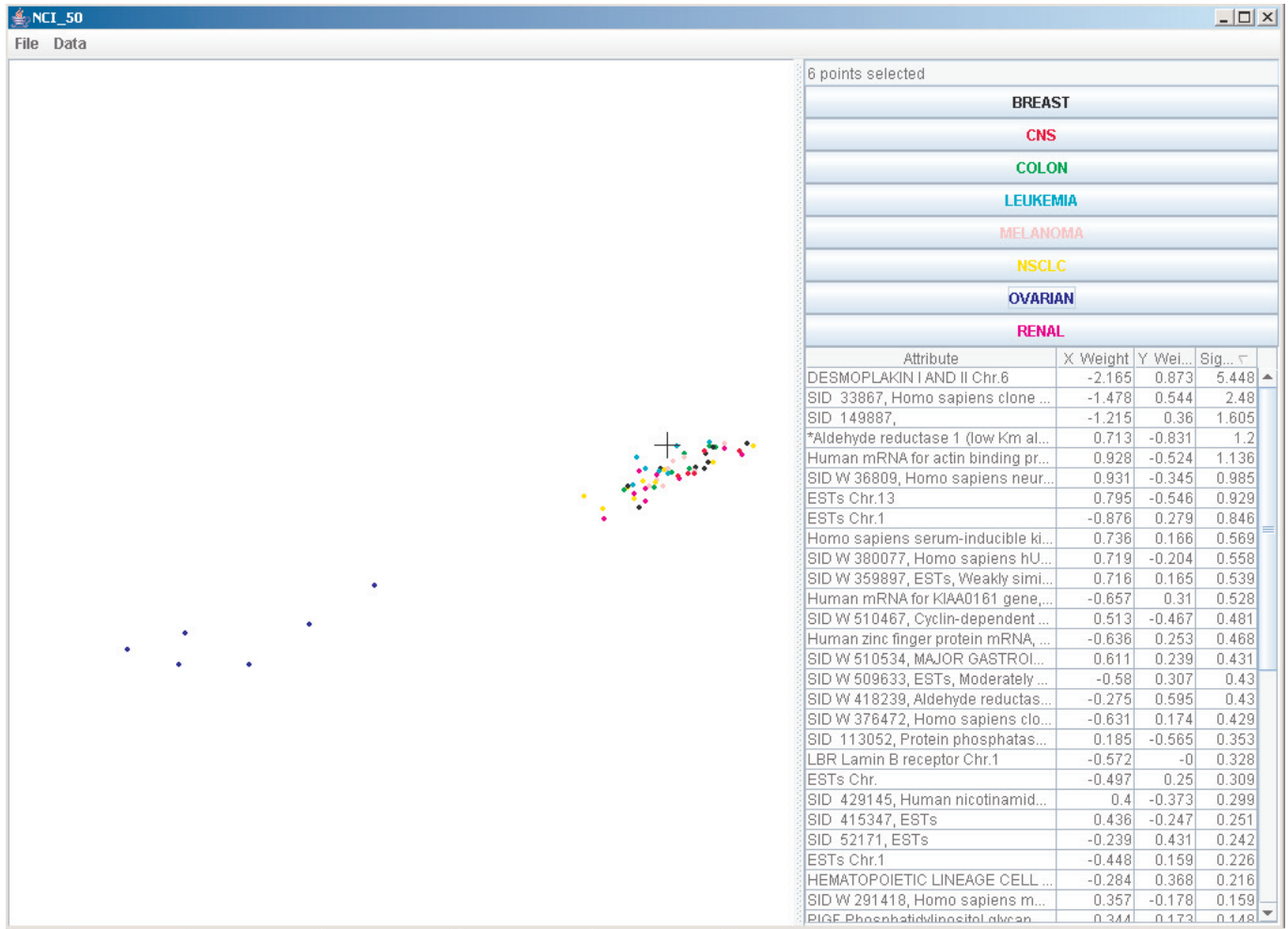


Figure 5: Screen shot of the prototype tool in use, showing its application to gene identification. The samples of ovarian cancer are shown to the left of the origin in the main panel, with all other samples bunched to the right. The projection table (bottom right) has been ordered to show the relative significance of each gene in generating this projection, with the Desmoplakin found to be most significant.

is to attempt to squeeze more than two dimensions of information into a two dimensional visualisation space either using glyphs or icons to embed extra dimensions of information into a graph, or by presenting the user with many different plots. Attempts have also been made to allow the user to explore the space of possible views, but the resulting interfaces have typically been opaque and unintuitive.

TPP seems to present a transparent interface that supports and exploits the natural human tendency to spot partial patterns and form hypotheses. These hypotheses are then tested through a process of continual feedback to the user.

Thus far the authors have concentrated on testing the potential of the approach on specific tasks and particular types of data, however it seems that TPP could be applied much more widely and used as a general purpose tool in the early stages of data analysis in a range of domains.

References

- [1] Grinstein G, Trutschl M, Cvek U. (2001) High-dimensional visualizations, *VII Data Mining Conference KDD Workshop 2001* (San Francisco-CA, USA, 2001), ACM Press: New York; 7-19.
- [2] Seo,J. and Shneiderman,B. (2005), A rank-by-feature framework for interactive exploration of multidimensional data,*Information Visualisation*, 4, 96-113.
- [3] Sammon,J.W. (1969), A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers*, C-18(5):401-409, May 1969.
- [4] Schena M, *et al* (1995), Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science*, 1995 Oct 20;270(5235):467-70.
- [5] Zhang,MQ. (1999), Large-scale gene expression data analysis: a new challenge to computational biologists. *Genome Res*, 9:681– 688.
- [6] Adams *et al* (2006) Meningothelial meningioma in a mature cystic teratoma of the ovary, *Pathologie* Mar 23
- [7] Dudziak *et al* (2003), Latent Membrane Protein 1 of Epstein-Barr Virus Induces CD83 by the NF- κ B Signaling Pathway. *J Virol*, 77(15): 8290–8298.
- [8] Golub,T.R.*et al* (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*,286(5439):531-7.
- [9] Asimov, D. (1985). The Grand Tour: A Tool for Viewing Multidimensional Data. *SIAM Journal of Scientific and Statistical Computing* 6(1), pp128– 11.
- [10] Cook, D., Buja, A., Cabrera, J., and Hurley, H. (1995), Grand tour and projection pursuit, *Journal of Computational and Graphical Statistics*, 2(3), pp.225– 250.
- [11] Cook,D. and Buja,A. (1997), Manual Controls for High-Dimensional Data Projections, *Journal of Computational and Graphical Statistics*, 6(4), pp. 464-480.
- [12] Dudoit, S., Fridlyand, J., and Speed, T. P. (2002) Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association*, Vol. 97, No. 457, p.77-87.
- [13] Faith,J and Mintram,R. and Angelova,M. (2006) Targeted Projection Pursuit for Gene Expression Data Classification and Visualisation, *Bioinformatics*, 22(21):2667.
- [14] Faith,J. and Brockway,M. (2006), Targeted Projection Pursuit Tool for Gene Expression Visualisation. *Journal of Integrative Bioinformatics*, 3(2):43.
- [15] Friedman, J. H., and Tukey, J. W., (1974),A Projection Pursuit Algorithm for Exploratory Data Analysis,*IEEE Transactions on Computers*, C- 23, 881-890.
- [16] Gurel,A. (2003) *Feed Forward Neural Networks - A Java Implementation V2.0*, <http://aydingurel.brinkster.net/neural>
- [17] Khan,J.*et al* (2001) Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature Medicine*, 7(6): 673–679.
- [18] Lee,E.K, Cook,D., Klinke,S. and Lumley,T. (2005), Projection Pursuit for Exploratory Supervised Classification, *Journal of Computational and Graphical Statistics*, 14(4), 831-846
- [19] Witten,I.H. and Frank,E. (2005) *Data Mining: Practical machine learning tools and techniques*, 2nd Edition, Morgan Kaufmann, San Francisco, 2005.
- [20] Johansson,G. (1973), Visual perception of biological motion and a model for its analysis. *Perception and Psychophysics*, 14, 201-211.
- [21] Blake,R., *Perception of Biological Motion*, <http://www.psy.vanderbilt.edu/faculty/blake/BM/BioMot.html>
- [22] Scherf,U.*et al* (2000) A Gene Expression Database for the Molecular Pharmacology of Cancer, *Nature Genetics*, 24(3), 236-244.