Entropy in Multivariate Analysis: Projection Pursuit

G P Nason.
School of Mathematical Sciences,
University of Bath,
BATH BA2 7AY,
UK.

Abstract

Projection pursuit is an exploratory data-analytic method in multivariate (MV) analysis. It is similar to the well-known *principal components analysis* (PCA) in that it can be used to find interesting structure within a MV data set. However, unlike PCA, which finds linear projections of maximum *variance*, projection pursuit finds linear projections of maximum *non-normality*, which sometimes is better at revealing structure within a MV data set.

We describe how the negative Shannon entropy can be used for measuring non-normality. As a result we can view projection pursuit with the Shannon index as a method which finds the projection with the maximum entropy. We outline the constrained optimising projection pursuit algorithm and mention briefly the role of sphering a MV data set.

Finally we illustrate the method as applied to the famous Lubischew beetle data and mention how it can be applied to multispectral images.

1 Introduction

Multivariate data sets are becoming increasingly common, especially with the advent of the computer, where rapid simultaneous collection of data is recorded on many variables, often giving rise to an initially incomprehensible set. These sets are usually written as a $K \times N$ data matrix X, where K is the number of variables and N the number of observations recorded on each variable.

Sometimes the experimenter has some questions already in mind, which hopefully the collected data can answer. Alternatively, we may just be interested in exploring the data set, formulating theories as we go. In both cases *exploratory data analysis* (EDA) and common sense should be applied in copious amounts to thoroughly investigate the general structure of the data.

1.1 Exploratory Data Analysis

Most statistical packages allow us to freely explore multivariate data sets. We should consider simple, easy to understand, methods before progressing onto more complicated techniques. The simplest method is probably to draw a pairwise plot of all the variables.

There exist many exciting statistical packages with excellent graphical facilities, which are useful for exploring multidimensional data sets. A notable example of such packages is S[1] which allows both the high quality plots that we are already accustomed to (e.g. scatter-plots, histograms, normal probability plots) and also more advanced facilities like 3D spinning plots which allow the display of 3-dimensions of a data set at one time. By spinning the point cloud, we can identify groups and interesting structure. A thorough examination of a reasonably-sized data set (say up to 10 variables and 500 observations) is perfectly feasible using such graphical facilities.

1.2 Further Methods

After an EDA session, or if one really has too many variables, a *dimension-reducing* technique may be appropriate. A common choice for this is *principal components analysis* (see Mardia *et. al* [7] for example) which we will abbreviate to PCA.

PCA involves performing an eigen-analysis on the variance (or correlation) matrix of the multivariate data set. For a K variable data set we will have K eigenvalues and K eigenvectors each of length K. The eigenvectors are usually called the *principal components* of the data set and we use them to form new variables out of the old by linear combination. The eigenvalues represent the variation of the data set with respect to each principal component. We usually plot the eigenvalues to aid in the selection of the target dimension. Usually we look for a large drop in eigenvalues to indicate the desired cutoff point. Sometimes we may have prior information or certain physical restrictions on what the target dimensionality should be.

We can also look at plots of the data with respect to the new set of variables. If the target dimension was 1 then we can view the new univariate data set as, say, a histogram, or some other form of density estimate. For a 2D set then a scatter-plot is indispensable. Dynamic graphics may again aid us in the 3D case.

Although PCA is a method based on analysing the variance matrix, we note that the data is eventually viewed by scatter-plots (or histograms). This is not to say that the eigenvectors and eigenvalues are not useful, but at the end of the day we look to the plots for identifying structure.

2 Structure In Multivariate Data

We should begin by asking what we mean by structure in multivariate data set? We include the following in our features of interest:

- outliers
- clusters
- linear structure

Many classical standard statistical procedures are badly derailed by outliers. For instance an outlier can severely affect a variance matrix, which is the basis for many statistical techniques. Much current statistical research is dedicated to *robustifying* statistical methods against the affect of outliers. EDA can go some way to identifying outliers. Once we have identified an outlier we can perform various actions, to be discussed below (Section 4.5), that mitigate its effects.

Of course, one could almost view an outlier to be a cluster of just one point, but we would usually require a cluster to be a close collection of many points, with some convenient definitions for "many" and "close". Clusters are very interesting since they provide the basis of possible discrimination between groups of individuals, and can lead to theories as to why groups of individuals should be so discriminated.

Outliers and clusters can be thought of as examples of non-linear structure. Linear structure is measured by correlation, and the correlation matrix for a data set is a summary of the linear structure within a data set. Correlation, with accompanying plots, can be essential in understanding how variables relate to each other.

Of course there are other elements of structure that we haven't mentioned. Correlation matrices and PCA deal with the linear structure admirably. With plots, PCA may even go some way to identifying clusters. Although before we apply these methods we should identify the destructive outliers and deal with them in some way. The remaining parts of this article will consider attempts at finding non-linear structure.

3 Structure And Non-uniformity

It doesn't take long to convince oneself that structure can be viewed as a non-uniform distribution of data points. Somehow then, we wish to find views of the multivariate data set that are non-uniform, and therefore contain structure. So given a particular view of a multivariate data set we would wish to be able to measure how non-uniform it was, and use this measure to search for views that were very non-uniform.

3.1 Measuring Non-uniformity

We will look at the discrete case first of all. Suppose that you have a set of N elements $S = \{1, \ldots, N\}$. We can let the random variable U have the *uniform* distribution on S defined by the *probability mass function* (pmf):

$$p_U(n) = \begin{cases} \frac{1}{N} & n \in S \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Given another random variable X, on the same sample space S, with probability mass p_X we may wish to measure how non-uniform it is. One way we can do this is to use the (negative) order-1 Shannon entropy:

$$\mathcal{I}(p_X) = \sum_{n=1}^{n=N} p_X(n) \log p_X(n) \tag{2}$$

This is a measure of how non-uniform p_X is: we'll show later for a simple case that this sum is least when p_X is the uniform distribution. Hence we can obtain a measure of how non-uniform p_X is compared to p_U by examining:

$$\mathcal{I}(p_X) - \mathcal{I}(p_U) \tag{3}$$

the larger this difference, the more non-uniform X.

As an interesting historical aside, Rényi[9] derived a more general characterisation of a measure of entropy called the *entropy of order-α* namely

$$\frac{1}{1-\alpha} \log \left(\sum_{n=1}^{n=N} p_X^{\alpha}(n) \right) \tag{4}$$

which tends to the Shannon entropy as $\alpha \to 1$.

3.2 A Proof

The mathematics in this section is not difficult but pass this section on a first reading if you're not interested in the details of why the uniform distribution is the minimiser of the entropy (2).

We only deal with the case $N = 2$ anyway! In this case the uniform distribution is just the one usually used for modelling the toss of a fair coin (as long as it doesn't land on its side!) namely $p_U(n) = \frac{1}{2}, n = 1, 2$. For the density of X we'll use the arbitrary distribution:

$$p_X(n) = \begin{cases} p_1 & n = 1 \\ p_2 & n = 2 \end{cases} \tag{5}$$

and remember that, since this is a distribution, we must have

$$p_1 + p_2 = 1. \tag{6}$$

We must show that the uniform is the minimiser of (2). Therefore let's work out the difference in entropies between the arbitrary and uniform:

$$\begin{aligned} \mathcal{I}(p_X) - \mathcal{I}(p_U) &= \sum_{n=1}^{n=2} p_X(n) \log p_X(n) - \sum_{n=1}^{n=2} p_U(n) \log p_U(n) \\ &= p_1 \log p_1 + p_2 \log p_2 - \tfrac{1}{2} \log \tfrac{1}{2} - \tfrac{1}{2} \log \tfrac{1}{2} \end{aligned} \tag{7}$$

Now by (6) we can replace p_2 by $1 - p_1$ to get:

$$\mathcal{I}(p_X) - \mathcal{I}(p_U) = p_1 \log p_1 + (1 - p_1) \log(1 - p_1) + \log 2 \tag{8}$$

Essentially the arbitrariness of the above expression is concentrated wholly on p_1. To find the required minimum we may just consider the behaviour of the function

$$f(x) = x \log x + (1 - x) \log(1 - x) + \log 2 \tag{9}$$

when x is in the interval $[0, 1]$. This function is plotted in Figure 1. One can see that $f(x)$ has a unique minimum at $x = \frac{1}{2}$. So the difference in entropies in (8) is minimimised when and only when $p_1 = \frac{1}{2}$, which of course means that $p_2 = \frac{1}{2}$ as well, and thus X would have to be uniformly distributed. If it is felt that this method of examining a plot lacks rigour then it is very easy to use calculus methods on (9) to establish the result. The result does extend to the more general case of N sample points.

3.3 Measuring Non-uniformity With Continuous Data

For dealing with continuous data the first thing we have to do is remember we're dealing with a *probability density function* (pdf) instead of a pmf, and the random variables X, U etc. will be measured on a continuous scale.

If we know that our data are bounded, say on the interval $[a, b]$, then we can redefine our entropy in (2) by

$$\mathcal{I}(f) = \int_a^b f(x) \log f(x) dx \tag{10}$$

and then by Section 3.2 it's believable that the continuous uniform density on $[a, b]$ is the minimising distribution. However we can obtain data that are recorded on an infinite domain. In this case our entropy becomes

$$\mathcal{I}(f) = \int_{-\infty}^{\infty} f(x) \log f(x) dx. \tag{11}$$

It can be shown that the minimising distribution for (11) amongst all distributions of mean zero and variance one is the standard normal distribution. This can be shown by an information theoretic or calculus of variations methods.

3.4 Structure And Non-normality

By the previous section we identify structure with non-normality and we can use the entropy defined by (11) to measure the non-normality of any density f. (Although we may have to translate and scale f to have mean zero and unit variance.)

4 Projections And Density Estimates

How does the previous section, dealing only with univariate quantities, relate to *multivariate* statistics.

4.1 Projections

We can obtain a univariate view of the data by taking suitable linear combinations of the original variables, or by forming a *projection* of the original data matrix. For a $K \times N$ data matrix X we can form a linear projection Y by using a $K \times 1$ projection vector α in the following way:

$$Y = \alpha^T X \tag{12}$$

where the superscript T denotes transpose. Note that Y is a N-dimensional vector, and thus is a univariate shadow of what is going on in the higher dimensional space.

4.2 Density Estimates

We can obtain some idea as to the spread of the observations of the univariate data set Y by computing a density estimate. One could use histograms to do this, but it's more convenient to use a technique called *kernel density estimation* (see Silverman[10]).

Given density points $Y_1, \ldots, Y_N$ we can form the kernel density estimate (KDE) for the density of Y by

$$\hat{f}(x) = \frac{1}{Nh} \sum_{n=1}^{n=N} K(\frac{x - Y_n}{h}). \tag{13}$$

Although this formula looks quite complicated, it is in fact performing a very simple task. Everywhere where there is a data point Y_n the KDE places a bump (kernel). The summation then adds up the effects of all the bumps to produce a final estimate of density. Where the data points are more dense the procedure will put more kernels close to each other, causing a greater value for the final density after the summation.

We usually choose a smooth kernel function, and here we use the normal density as our kernel. This gives the final density estimate many nice properties,

one of the most important being the fact that we can differentiate the final density as many times as we like.

We also have control over the value h which is called the *bandwidth* (or window-width). This controls the width of the bump placed over each data-point, and as a consequence controls how peaked the final density estimate is. The choice of the value h is one of great research interest, and several methods have been proposed as to how to choose the bandwidth automatically.

4.3 How Non-normal Is My Projection?

We are now at the stage where we can merge the idea of measuring non-normality using the entropy (11) and that of the projection of a multivariate data set to a univariate one.

In Figure 2 we have constructed an artificial bivariate data set. Notice that the set splits clearly into two groups. In Figure 3 we have added a projection vector (solid line) at approximately 45° to the axes. We have then projected all the data points onto this vector. The projected data points are a univariate data set and we can form a kernel density estimate of them as in Figure 4.

In Figure 5 we have added a projection vector at $-45°$ to the axes, and projected the data onto it. The corresponding kernel density estimate of the projected data is shown in Figure 6.

The view that should be most interesting to us is that given in Figure 5. The density estimate certainly shows the interesting structure. Notice that the density estimate is a function of the projection vector, so we can write $\hat{f}_\alpha(x)$ to reflect this. Remember we can measure departures from normality by using the entropy index, and in fact this becomes a function of the projection direction as well

$$\mathcal{I}(\alpha) = \mathcal{I}(\hat{f}_\alpha) = \int_{-\infty}^{\infty} \hat{f}_\alpha(x) \log \hat{f}_\alpha(x) dx \tag{14}$$

We call $\mathcal{I}(\alpha)$ the *projection index*.

4.4 Projection Pursuit

So we have a direct measure of how interesting a particular direction α is. It doesn't take much to believe that $\mathcal{I}$ is in fact a differentiable function of the components of α, so once you are at a particular direction, you have an indication of how you should change direction to improve the interest of the projection. This is in fact exactly what we do. We start at an initial position, compute the index and its derivatives, work out which way to change the projection direction to improve matters and move towards that better projection. In fact the exact optimisation problem can be mathematically stated as

$$\max \mathcal{I}(\alpha) \quad \text{subject to } \alpha^T \alpha = 1 \tag{15}$$

The constraint on α at the end of this enforces the length of α to be 1, since it is purely the direction of α that is interesting, not its length.

Details of how much to move in the chosen direction, and how we know when we've got to the "best" projection, are left to a numerical optimisation procedure. We have used steepest ascent, conjugate gradient and variable metric optimisation methods, the details of which can be found in many texts on numerical analysis (Press *et al.* [8] for example).

Figure 7 outlines the projection pursuit algorithm. The initial projection direction can be chosen at random, or maybe from previous principal components directions. The optimality condition is tested by examining for convergence of the projection index and checking the size of its derivatives.

4.5 Other Considerations

We have deliberately omitted discussion of a technique called *sphering* (Tukey and Tukey[11]). This is a preprocessing operation which transforms the data to have zero mean and identity variance matrix. Due to the length constraint on α the projected data will inherit this property. This incidentally makes the computation of the projection index easier.

We can also project into 2 or 3 dimensions. We could project into more, but this probably isn't useful. When we have many data points, the entropy index is quite expensive to compute (especially when projecting into more than 1 dimension), so approximations have been developed to it. One is called the moment index and based upon higher moments of the data set[5].

For details of these approximations, projection into more than one dimension, and how sphering fits in, the reader is referred to Jones and Sibson[5]. Also useful on projection pursuit are the papers by Huber[4] and Friedman[2], the latter author (with Tukey) coining the phrase *projection pursuit* in an early paper[3].

We now turn our attention to the problem of outliers. The moment index is sometimes attracted to projections with outliers in. Sometimes, however, we are more interested in true clusters. We could remove the outliers and then perform projection pursuit. We could also reduce the distance of the outliers by a certain amount, and thus destroy their ability to unduly influence the method. Tukey in [5] suggested two ways of doing this.

5 Some Examples

We detail two real data sets to which we have applied the methods described above.

5.1 Lubischew Beetle Data

We review the work performed in Jones and Sibson's [5] paper on a set of data collected on 74 beetles by Lubischew[6]. According to Lubischew, some species may only be identified by examination of the male copulative organs only; the females remaining indistinguishable. Also some species have radical economic consequences, making identification essential. Lubischew goes on to describe the use of discriminant functions in identifying species. The hope being that a rule can be found to discriminate between beetles, based solely on external characteristics such as lengths and angles of various body-parts. We use the data in the paper to experiment with PCA and projection pursuit, and see if these techniques can tell us anything. Each of the 74 male beetles has 6 measurements taken on them leading to a 6×74 data matrix.

First we should view the data with the original axes, as in Figure 8. We plot the species labels as groups 1, 2 and 3. In this case, were we to obtain female measurements (and assuming that the females were of approximately the same dimensions as their male counterparts), we could plot them as well, and try to identify which species they were. From Figure 8 we may conclude that although certain variable combinations are able to discriminate between say group 3 and not group 3, it would be difficult to discriminate between all three groups at once (although the plot of Variable 4 against 5 comes close).

We could perform a PCA, and plot the data with respect to the first two principal components. This gives the picture in Figure 9. You can see the three groups very nicely, although there may still be some confusion over groups 1 and 2.

We exhibit three projection pursuit solutions in Figures 10-12. The projected data sets are on a different scale to Figure 9, due to the sphering transformation mentioned in Section 4.5. The projection in Figure 10 does less well than PCA, with groups 1 and 2 very confused. Here the numerical optimisation routine has found a local optima of the projection index. Sometimes local optima can be interesting, so we're glad to find them.

Figure 11 is a more non-normal projection (having an negative entropy index of 1.49 as opposed to 1.44 for Figure 10). The solution is perhaps, subjectively, as good as the PCA solution. Figure 12 is very interesting in that the three groups are very well separated. The projection was obtained not by the negative entropy index, but an approximation called the moment index and mentioned in Section 4.5 (see also Jones and Sibson[5]).

So clearly, for this data set at least, projection pursuit comes up with a very interesting projection.

5.2 Multispectral Data

Often multivariate data sets relate to experiments where we record data for N observations on K spectra. Projection pursuit can then be used in a PCA role in

reducing the dimensionality of the data set. The multi-spectral data sets occur in many areas of science, for example NIR spectroscopy and remote-sensing just to name two. The author has, under the supervision of Professor R. Sibson, been experimenting with the application of projection pursuit to multispectral images obtained using equipment similar to the Thematic Mapper aboard the LANDSAT series of satellites with interesting results.

6 Conclusions

We hope that this article has explained the principles behind projection pursuit and how it makes use of the Shannon entropy to reach its goal. We also hope that we have put across the value of projection pursuit for exploratory data analysis and that we will, by means of this article, promote the use of the method, not as a replacement for PCA or any other statistical method, but as a complementary method.

A library of FORTRAN subroutines to perform projection pursuit is available from the author.

Finally the author would like to acknowledge Professor R. Sibson for stimulating his interest in projection pursuit and making many helpful suggestions, and Mr J. Stander for many helpful comments concerning this article.

References

[1] R.A. Becker, J. M. Chambers, and A. R. Wilks. *The New S Language*. Wadsworth & Brooks/Cole Advanced Books & Software, Pacific Grove, California, 1988.

[2] J. H. Friedman. Exploratory projection pursuit. *J. Am. Statist. Ass.*, 82(397):249–266, March 1987.

[3] J. H. Friedman and J. W. Tukey. A projection pursuit algorithm for exploratory data analysis. *IEEE Trans. Comput.*, C23(9):881–890, 1974.

[4] P. J. Huber. Projection pursuit (with discussion). *Ann. Statist.*, 13:435–525, 1985.

[5] M. C. Jones and R. Sibson. What is projection pursuit? (with discussion). *J. R. Statist. Soc. A*, 150:1–36, 1987.

[6] A. A. Lubischew. On the use of discriminant functions in taxonomy. *Biometrics*, 18:455–477, 1962.

[7] K. V. Mardia, J. T. Kent, and J. M. Bibby. *Multivariate Analysis*. Probability and Mathematical Statistics. Academic Press, London, 1979.

[8] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes.* Cambridge University Press, Cambridge, 1989. FORTRAN Version.

[9] A Rényi. On measures of entropy and information. In J. Neyman, editor, *Proceedings of 4th Berkeley Symposium on Math. Statist. and Probab.*, pages 547–561. Berkeley, 1961.

[10] B. W. Silverman. *Density Estimation for Statistics and Data Analysis.* Chapman and Hall, London, 1986.

[11] P. A. Tukey and J. W. Tukey. Preparation; prechosen sequences of views. In V. Barnett, editor, *Interpreting Multivariate Data.*, pages 189–213. Wiley, Chichester, 1981.