Stat 432: Assignment 3

Q3: Non-Parametric Bootstrapping

After each census a group of researchers from the University of Otago assign a socio-economic deprivation score to each Statistical Area 2 (SA2) across the country (SA2s can be thought of as suburbs, e.g. Hataitai, Berhampore, Ponsonby, etc…). The higher the deprivation score assigned to an SA2, the higher the level of socio-economic deprivation endured by the people living in therein.

In order construct the deprivation scores the researchers determine the **proportion** of people living in each SA2:

- Who are unemployed or claiming Job Seeker Support (JSS_PC).
- Who are single parents or claiming Sole Parent Support (SPS_PC)
- Who are claiming some form of Means Tested Benefit (MTB_PC)
- Who have no qualifications (No_Quals_PC)
- Who live in homes with significant levels of damp and mould (Damp_Mould_PC)
- Who do not have access to the internet at home (No_Int_PC).
- Who have household income less than $50,000 (Inc_Leq_50k_PC).
- Who live in accommodation deemed overcrowded (Crowded_PC)

These 9 variables are then condensed into an overall deprivation score by performing principal component analysis on the data and projecting onto the first principal component. The reader is directed to the Appendix below, Tutorial 3 and Section 7.4.2 of the course notes for an overview of this technique.

```
# Section 1
#------------------------------------------------------------------------
# Reading in the data and obtaining the eigenvalue & eigenvector corresponding to the 1st principal component

# 1.1a - Read in the 'NZDEP18_SA2_Data.csv' data set
> Dep1a <- read.csv("F:\\Data Sets\\...\\NZDEP18_SA2_Data.csv")


# 1.1b - Refine to required variables and specifying
      # - the number of rows (i.e. # of SA2s # appearing in the study)
      # - the number of variables (measures of socio-economic deprivation)
> Dep1b = subset(Dep1a, select = -c(SA2_2018_Code, SA2_2018_Name))
> rows <- nrow(Dep1b)
> cols <- ncol(Dep1b)


# 1.1c - Obtaining the covariance matrix of the normalised data (i.e. obtaining the correlation matrix of the non-normalised data)
> Dep1c <- cor(Dep1b)


# 1.1d - Obtaining the eigenvalues and eigenvectors of the correlation matrix
> Dep1d <- eigen(Dep1c)
      # Obtain eigenvalues
      > Dep1d1 <- Dep1d$values
      # Obtain proportion of variance in the data explained by 1st principal component
      > Dep1d2 <- Dep1d1[[1]] / cols
      # Obtain eigenvectors.
      > Dep1d3 <- Dep1d$vectors
      # Obtain the first principal component.
      # Note that eigenvectors are unique up to a factor of +-1. Since we know from the previous output that the elements
      # (loadings) of the 1st principal component all have the same (negative) sign, we are perfectly within our rights simply to
      # take the absolute value of each element
      > Dep1d4 <- abs(Dep1d3[,1])
```

The data set entitled 'NZDEP18_SA2_Data.csv' shows the value of each of the aforementioned variables for each SA2 across the country. The data set is read into R and analysed using the code below:

a) Run the code given above and determine:
   - The proportion of the variance in the data explained by the first principal component
   - The loadings of the first principal component

b) There is interest in constructing a confidence interval for the proportion of variance explained by the first principal component and confidence intervals for the loadings associated to it. Construct R code to create 5,000 bootstrap samples of the data set Dep1b from the above code. For each bootstrap sample calculate:
   - The proportion of the variance explained by the first principal component.
   - The loadings associated to the first principal component. It can be assumed that the loadings all have the same sign.

   Use set.seed(12345) to ensure that you get the same bootstrap samples each time the code is run.

c) Construct a histogram of the proportion of the variance explained by the first principal component within each of the bootstrapped samples.

d) Calculate the standard error of this parameter and use it to construct the standard 95% bootstrap confidence interval.

e) Construct histograms of the loadings associated to the first principal component within each of the bootstrapped samples.

f) Calculate the standard error of each of the loadings and use it to construct standard 95% bootstrap confidence intervals for the loadings of each of the 9 variables.

g) Explain why bootstrapped confidence intervals where appropriate in these scenarios.

Appendix – Principal Component Analysis

Principal Component Analysis (PCA) is a commonly used dimensionality reduction technique which, **under certain circumstances,** can be used to reduce n-dimensional data to a one-dimensional score without significant loss of information.

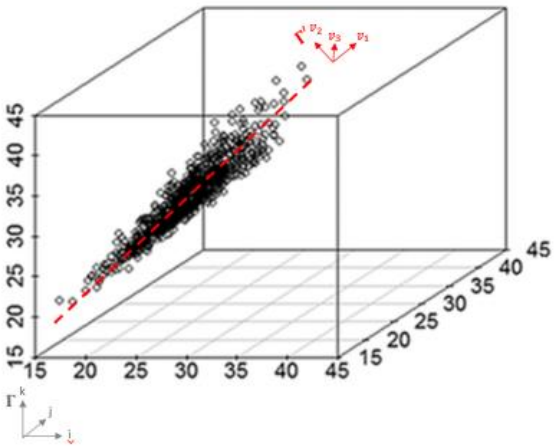
Typically, the PCA process begins by normalising the data such that each variable has a mean of zero and a standard deviation equal to one. This is done to account for the fact that the variables may have different scales. The covariance matrix of the normalised data is then constructed[1]. Since the covariance matrix is symmetric, its eigenvectors $\underline{\boldsymbol{v_1}}, \underline{\boldsymbol{v_2}}, \ldots, \underline{\boldsymbol{v_n}}$ form an orthonormal decomposition of $\mathbb{R}^n$, i.e. these eigenvectors form a set of *n* orthonormal axes. Arranged in descending order of their associated eigenvalues, the first eigenvector $\underline{\boldsymbol{v_1}}$ (referred to as the first principal component) explains the largest amount of the variance in the data, the second eigenvector $\underline{\boldsymbol{v_2}}$ explains the largest amount of the *remaining* variance and so on. Herein, the relative magnitude of the associated eigenvalue $\frac{|\lambda_i|}{n}$ describes the proportion of the variance explained by the corresponding eigenvector.

More succinctly, if Γ describes the axes of the standard coordinate system…

$$\Gamma = \left\{ \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix} \begin{matrix} \cdots \\ \cdots \\ \ddots \\ \cdots \end{matrix} \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix} \right\}$$

…then the eigenvectors of the covariance matrix $\underline{\boldsymbol{v_1}}, \underline{\boldsymbol{v_2}}, \ldots, \underline{\boldsymbol{v_n}}$ form a set of orthonormal axes for an alternative coordinate system Γ'. This new co-ordinate system Γ' can be thought of as the coordinate system that 'best describes the data'.

Figure 1 – Illustration of Γ and Γ' for a given three-dimensional data set



In order to convert a point $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ from the standard coordinate system Γ to the new coordinate system Γ', one simply multiplies $\boldsymbol{x}$ on the right by the matrix of eigenvectors $\boldsymbol{M}$:

$$\boldsymbol{x}\,\boldsymbol{M} = \boldsymbol{x}'$$

$$(x_1, x_2, \ldots, x_n) \left[\begin{pmatrix} v_{11} \\ v_{12} \\ \vdots \\ v_{1n} \end{pmatrix} \begin{pmatrix} v_{21} \\ v_{22} \\ \vdots \\ v_{2n} \end{pmatrix} \begin{matrix} \cdots \\ \cdots \\ \ddots \\ \cdots \end{matrix} \begin{pmatrix} v_{n1} \\ v_{n2} \\ \vdots \\ v_{nn} \end{pmatrix} \right] = (x_1', x_2', \ldots, x_n')$$

Setting $\underline{\boldsymbol{v_j}} = \underline{\boldsymbol{0}}$ for all *j = 2, … n* has the effect of projecting the data onto the first axis of the new coordinate system Γ' (this process is known as projecting onto the first principal component).

$$(x_1, x_2, \ldots, x_n) \left[\begin{pmatrix} v_{11} \\ v_{12} \\ \vdots \\ v_{1n} \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \begin{matrix} \cdots \\ \cdots \\ \ddots \\ \cdots \end{matrix} \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \right] = (x_1', 0, \ldots, 0)$$

[1] Note that the covariance matrix of the normalised data is equal to the correlation matrix of the non-normalised data.

In essence, this reduces the data from n-dimensions to one dimension. The elements of the vector $\underline{\boldsymbol{v_1}}$ are known as '*loadings*'. As long as the loadings are all non-negative, they can be interpreted as weights converting each of the original data points into an overall score since

$$(x_1\ ,\ x_2,\ \ldots,\ x_n\ )\ (v_{11}\ ,\ v_{12},\ \ldots,\ v_{1n}\ )^T\ =\ x_1$$

The hope is that the first principal component explains a sufficiently high proportion of the variance in the data (i.e. $\frac{|\lambda_1|}{n}$ is sufficiently close to 1) such that that this is achieved without too much loss of information.

Prosaically, one may think of the process of projecting n-dimensional data onto the first principal component as plotting the data in n-dimensional space, inserting a line of best fit into the data and projecting each point onto the line of best fit. This is illustrated in the diagram below.

Figure 2 – Illustration of projecting three-dimensional data onto the first principal component