Chapter 6

Principal Component Analysis

6.1 Introduction

Principal component analysis (PCA) is used to explain the dispersion structure with a few linear combinations of the original variables, called principal components. These linear combinations are uncorrelated if $\boldsymbol{S}$ or $\boldsymbol{R}$ is used as the dispersion matrix. The analysis is used for data reduction and interpretation. The notation $\boldsymbol{e}_j$ will be used for orthonormal eigenvectors: $\boldsymbol{e}_j^T \boldsymbol{e}_j = 1$ and $\boldsymbol{e}_j^T \boldsymbol{e}_k = 0$ for $j \neq k$. The eigenvalue eigenvector pairs of a matrix $\boldsymbol{\Sigma}$ will be $(\lambda_1, \boldsymbol{e}_1), ..., (\lambda_p, \boldsymbol{e}_p)$ where $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p$. The eigenvalue eigenvector pairs of a matrix $\hat{\boldsymbol{\Sigma}}$ will be $(\hat{\lambda}_1, \hat{\boldsymbol{e}}_1), ..., (\hat{\lambda}_p, \hat{\boldsymbol{e}}_p)$ where $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \cdots \geq \hat{\lambda}_p$. The generalized correlation matrix defined below is the correlation matrix when second moments exist if $\boldsymbol{\Sigma} = c \operatorname{Cov}(\boldsymbol{x})$ for some constant $c > 0$.

Definition 6.1. Let $\boldsymbol{\Sigma} = ((\sigma_{ij}))$ be a positive definite symmetric $p \times p$ dispersion matrix. A *generalized correlation matrix* $\boldsymbol{\rho} = \big((\rho_{ij})\big)$ where

$$\rho_{ij} = \frac{\sigma_{ij}}{\sqrt{\sigma_{ii}\sigma_{jj}}}.$$

The following theorem holds since the eigenvalues and generalized correlation matrix are continuous functions of $\boldsymbol{\Sigma}$. Also see Theorem 3.29. When the distribution of the $\boldsymbol{x}_i$ is unknown, then a good dispersion estimator estimates $c\boldsymbol{\Sigma}$ on a large class of distributions where $c > 0$ depends on the unknown distribution of $\boldsymbol{x}_i$. For example, if the $\boldsymbol{x}_i \sim EC_p(\boldsymbol{\mu}, \boldsymbol{\Sigma}, g)$, then the sample covariance matrix $\boldsymbol{S}$ estimates $\operatorname{Cov}(\boldsymbol{x}) = c_X \boldsymbol{\Sigma}$.

Theorem 6.1. Suppose the dispersion matrix $\boldsymbol{\Sigma}$ has eigenvalue eigenvector pairs $(\lambda_1, \boldsymbol{e}_1), ..., (\lambda_p, \boldsymbol{e}_p)$ where $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p$. Suppose $\hat{\boldsymbol{\Sigma}} \xrightarrow{P} c\boldsymbol{\Sigma}$ for some constant $c > 0$. Let the eigenvalue eigenvector pairs of $\hat{\boldsymbol{\Sigma}}$ be $(\hat{\lambda}_1, \hat{\boldsymbol{e}}_1), ..., (\hat{\lambda}_p, \hat{\boldsymbol{e}}_p)$ where $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \cdots \geq \hat{\lambda}_p$. Then $\hat{\lambda}_j(\hat{\boldsymbol{\Sigma}}) \xrightarrow{P} c\lambda_j(\boldsymbol{\Sigma}) = c\lambda_j$, $\hat{\boldsymbol{\rho}} \xrightarrow{P} \boldsymbol{\rho}$ and $\hat{\lambda}_j\left(\hat{\boldsymbol{\rho}}\right) \xrightarrow{P} \lambda_j\left(\boldsymbol{\rho}\right)$ where $\lambda_j(\boldsymbol{A})$ is the jth eigenvalue of $\boldsymbol{A}$ for $j = 1, ..., p$.

Eigenvectors $\boldsymbol{e}_j$ are not continuous functions of $\boldsymbol{\Sigma}$, and if $\boldsymbol{e}_j$ is an eigenvector of $\boldsymbol{\Sigma}$ then so is $-\boldsymbol{e}_j$. The software produces $\hat{\boldsymbol{e}}_j$ which sometimes approximates $\boldsymbol{e}_j$ and sometimes approximates $-\boldsymbol{e}_j$ if the eigenvalue λ_j is unique, since then the set of eigenvectors corresponding to λ_j has the form $a\boldsymbol{e}_j$ for any nonzero constant a. The situation becomes worse if some of the eigenvalues are equal, since the possible eigenvectors then span a space of dimension equal to the multiplicity of the eigenvalue. Hence if the multiplicity is two and both $\boldsymbol{e}_j$ and $\boldsymbol{e}_k$ are eigenvectors corresponding to the eigenvalue λ_i, then $\boldsymbol{e}_i = \boldsymbol{x}_i/\|\boldsymbol{x}_i\|$ is also an eigenvector corresponding to λ_i where $\boldsymbol{x}_i = a_j\boldsymbol{e}_j + a_k\boldsymbol{e}_k$ for constants a_j and a_k which are not both equal to 0. The software produces $\hat{\boldsymbol{e}}_j$ and $\hat{\boldsymbol{e}}_k$ that are approximately in the span of $\boldsymbol{e}_j$ and $\boldsymbol{e}_k$ for large n by the following theorem, which also shows that $\hat{\boldsymbol{e}}_i$ is asymptotically an eigenvector of $\boldsymbol{\Sigma}$ in that $(\boldsymbol{\Sigma} - \lambda_i)\hat{\boldsymbol{e}}_i \xrightarrow{P} \boldsymbol{0}$. It is possible that $\hat{\boldsymbol{e}}_{i,n}$ is arbitrarily close to $\boldsymbol{e}_i$ for some values of n and arbitrarily close to $-\boldsymbol{e}_i$ for other values of n so that $\hat{\boldsymbol{e}}_i \equiv \hat{\boldsymbol{e}}_{i,n}$ oscillates and does not converge in probability to either $\boldsymbol{e}_i$ or $-\boldsymbol{e}_i$.

Theorem 6.2. Assume the $p \times p$ symmetric dispersion matrix $\boldsymbol{\Sigma}$ is positive definite.

a) If $\hat{\boldsymbol{\Sigma}} \xrightarrow{P} \boldsymbol{\Sigma}$, then $\hat{\boldsymbol{\Sigma}}\boldsymbol{e}_i - \hat{\lambda}_i\boldsymbol{e}_i \xrightarrow{P} \boldsymbol{0}$.

b) If $\hat{\boldsymbol{\Sigma}} \xrightarrow{P} \boldsymbol{\Sigma}$, then $\boldsymbol{\Sigma}\hat{\boldsymbol{e}}_i - \lambda_i\hat{\boldsymbol{e}}_i \xrightarrow{P} \boldsymbol{0}$.

If $\hat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma} = O_P(n^{-\delta})$ where $0 < \delta \leq 0.5$, then

c) $\lambda_i\boldsymbol{e}_i - \hat{\boldsymbol{\Sigma}}\boldsymbol{e}_i = O_P(n^{-\delta})$, and

d) $\hat{\lambda}_i\hat{\boldsymbol{e}}_i - \boldsymbol{\Sigma}\hat{\boldsymbol{e}}_i = O_P(n^{-\delta})$.

e) If $\hat{\boldsymbol{\Sigma}} \xrightarrow{P} c\boldsymbol{\Sigma}$ for some constant $c > 0$, and if the eigenvalues $\lambda_1 > \cdots > \lambda_p > 0$ of $\boldsymbol{\Sigma}$ are unique, then the absolute value of the correlation of $\hat{\boldsymbol{e}}_j$ with $\boldsymbol{e}_j$ converges to 1 in probability: $|\text{corr}(\hat{\boldsymbol{e}}_j, \boldsymbol{e}_j)| \xrightarrow{P} 1$.

Proof. a) $\hat{\boldsymbol{\Sigma}}\boldsymbol{e}_i - \hat{\lambda}_i\boldsymbol{e}_i \xrightarrow{P} \boldsymbol{\Sigma}\boldsymbol{e}_i - \lambda_i\boldsymbol{e}_i = \boldsymbol{0}$.

b) Note that $(\boldsymbol{\Sigma} - \lambda_i\boldsymbol{I})\hat{\boldsymbol{e}}_i = [(\boldsymbol{\Sigma} - \lambda_i\boldsymbol{I}) - (\hat{\boldsymbol{\Sigma}} - \hat{\lambda}_i\boldsymbol{I})]\hat{\boldsymbol{e}}_i = o_P(1)O_P(1) \xrightarrow{P} \boldsymbol{0}$.

c) $\lambda_i \boldsymbol{e}_i - \hat{\boldsymbol{\Sigma}} \boldsymbol{e}_i = \boldsymbol{\Sigma} \boldsymbol{e}_i - \hat{\boldsymbol{\Sigma}} \boldsymbol{e}_i = O_P(n^{-\delta})$.
d) $\hat{\lambda}_i \hat{\boldsymbol{e}}_i - \boldsymbol{\Sigma} \hat{\boldsymbol{e}}_i = \hat{\boldsymbol{\Sigma}} \hat{\boldsymbol{e}}_i - \boldsymbol{\Sigma} \hat{\boldsymbol{e}}_i = O_P(n^{-\delta})$.

e) Note that a) and b) hold if $\hat{\boldsymbol{\Sigma}} \xrightarrow{P} \boldsymbol{\Sigma}$ is replaced by $\hat{\boldsymbol{\Sigma}} \xrightarrow{P} c\boldsymbol{\Sigma}$. Hence for large n, $\hat{\boldsymbol{e}}_i \equiv \hat{\boldsymbol{e}}_{i,n}$ is arbitrarily close to either $\boldsymbol{e}_i$ or $-\boldsymbol{e}_i$, and the result follows.

Rule of thumb 6.1. To use PCA, assume the DD plot and subplots of the scatterplot matrix are linear. Want $n > 10p$ for classical PCA and $n > 20p$ for robust PCA that uses FCH, RFCH or RMVN. For classical PCA, use the correlation matrix $\boldsymbol{R}$ instead of the covariance matrix $\boldsymbol{S}$ if $\max_{i=1,...,p} S_i^2 / \min_{i=1,...,p} S_i^2 > 2$. If $\boldsymbol{S}$ is used, also do a PCA using $\boldsymbol{R}$.

The trace of a matrix $\boldsymbol{A}$ is the sum of the diagonal elements of $\boldsymbol{A}$ and the sum of the eigenvalues of $\boldsymbol{A}$. If $\boldsymbol{A}$ is a $p \times p$ matrix, then trace($\boldsymbol{A}$) = $tr(\boldsymbol{A}) = \sum_{i=1}^p \boldsymbol{A}_{ii} = \sum_{i=1}^p \lambda_i$. Note that $tr(\text{Cov}(\boldsymbol{x})) = \sigma_1^2 + \cdots + \sigma_p^2$ and $tr(\hat{\boldsymbol{\rho}}) = p$.

Definition 6.2. Let dispersion estimator $\hat{\boldsymbol{\Sigma}}$ have eigenvalue eigenvector pairs $(\hat{\lambda}_1, \hat{\boldsymbol{e}}_1), ..., (\hat{\lambda}_p, \hat{\boldsymbol{e}}_p)$ where $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \cdots \geq \hat{\lambda}_p$. Then the *p principal components* corresponding to the jth case $\boldsymbol{x}_j$ are $Z_{j1} = \hat{\boldsymbol{e}}_1^T \boldsymbol{x}_j, ..., Z_{jp} = \hat{\boldsymbol{e}}_p^T \boldsymbol{x}_j$. Let the vector $\boldsymbol{z}_j = (Z_{j1}, ..., Z_{jp})^T$. The *proportion of the trace explained* by the first kth principal components is $\sum_{i=1}^k \hat{\lambda}_i / \sum_{j=1}^p \hat{\lambda}_j = \sum_{i=1}^k \hat{\lambda}_i / tr(\hat{\boldsymbol{\Sigma}})$. When a correlation or covariance matrix is being estimated, "trace" is replaced by "variance." The population analogs use the dispersion matrix $\boldsymbol{\Sigma}$ with eigenvalue eigenvector pairs $(\lambda_i, \boldsymbol{e}_i)$ for $i = 1, ..., p$. The population principal components corresponding to the j case are $Y_{ji} = \boldsymbol{e}_i^T \boldsymbol{x}_j$, and $Z_{ji} = \hat{Y}_{ji}$ for $i = 1, ..., p$.

Note that the principal components can be collected into an $n \times p$ data matrix

$$\boldsymbol{Z} = \begin{bmatrix} Z_{1,1} & Z_{1,2} & \ldots & Z_{1,p} \\ Z_{2,1} & Z_{2,2} & \ldots & Z_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ Z_{n,1} & Z_{n,2} & \ldots & Z_{n,p} \end{bmatrix} = \begin{bmatrix} \boldsymbol{u}_1 & \boldsymbol{u}_2 & \ldots & \boldsymbol{u}_p \end{bmatrix} = \begin{bmatrix} \boldsymbol{z}_1^T \\ \vdots \\ \boldsymbol{z}_n^T \end{bmatrix}.$$

Then $\boldsymbol{u}_i$ corresponds to the ith principal component. A plot of the second principal component versus the first principal component can be useful.

The data matrix $\boldsymbol{W}$ corresponds to the usual axes where $\boldsymbol{e}_i$ is a vector of zeroes except for a one in the ith position. Hence the ith axis corresponds to the ith variable X_i. The data matrix $\boldsymbol{Z}$ corresponds to axes that are parallel to the axes of the hyperellipsoid corresponding to the dispersion matrix $\hat{\boldsymbol{\Sigma}}$. These axes are a rotation of the usual axes about the origin.

If $\hat{\boldsymbol{\Sigma}} = \boldsymbol{S}$, then the definition of the estimated proportion of the total population variance may make little sense if the variables are measured on different scales. Assume the population covariance matrix is I_2. Then $\lambda_j/(\lambda_1 + \lambda_2) = 0.5$, but if x_j is multiplied by 3 then $V(x_j) = 9 = \lambda_j$, and $\lambda_j/(\lambda_1 + \lambda_2) = 0.9$. Then x_j seems much more important than the other variable just by scaling. This is why rule of thumb 6.1 says $\boldsymbol{R}$ should be used instead of $\boldsymbol{S}$ if $\max_{i=1,...,p} S_i^2 / \min_{i=1,...,p} S_i^2 > 2$.

Examine Theorems 2.4, 2.5 and Figure 2.1. The hyperellipsoid $\{\boldsymbol{x}|D^2_{\boldsymbol{x}} \leq h^2\} = \{\boldsymbol{x} : (\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu}) \leq h^2\}$, where $h^2 = u_{1-\alpha}$ and $P(U \leq u_{1-\alpha}) = 1 - \alpha$, is the highest density region covering $1 - \alpha$ of the mass for an elliptically contoured distribution. The hyperellipsoid is centered at $\boldsymbol{\mu}$. If $\boldsymbol{\mu} = \boldsymbol{0}$, then points at squared distance $\boldsymbol{w}^T \boldsymbol{S}^{-1} \boldsymbol{w} = h^2$ from the origin lie on the hyperellipsoid centered at the origin whose axes are given by the eigenvectors $\boldsymbol{e}_i$ where the half length in the direction of $\boldsymbol{e}_i$ is $h\sqrt{\lambda_i}$.

The projection vector of a vector $\boldsymbol{x}$ onto a vector $\boldsymbol{e}$ is

$$\frac{\boldsymbol{e}\boldsymbol{e}^T\boldsymbol{x}}{\boldsymbol{e}^T\boldsymbol{e}}.$$

Hence if $\boldsymbol{e}^T\boldsymbol{e} = 1$, the projection vector is $\boldsymbol{v} = [\boldsymbol{e}^T\boldsymbol{x}]\boldsymbol{e}$ and $\|\boldsymbol{v}\| = |\boldsymbol{e}^T\boldsymbol{x}|$. So $\boldsymbol{e}^T\boldsymbol{x}$ is the signed length of the projection vector of $\boldsymbol{x}$ onto $\boldsymbol{e}$, and $\boldsymbol{e}^T\boldsymbol{x}$ is called the (scalar) projection of $\boldsymbol{x}$ onto $\boldsymbol{e}$.

The $\boldsymbol{e}_i$ are the directions of the axes through the origin that are parallel to the axes of the hyperellipsoid. Suppose $\boldsymbol{\mu} = \boldsymbol{0}$. Then the ith principle component is the linear combination of the predictors that is the projection on the ith axis of the hyperellipsoid. That is, get the projection vectors of the $\boldsymbol{x}_i$ onto $\boldsymbol{e}_i$ and find their signed lengths $\boldsymbol{e}_i^T\boldsymbol{x}_i$ from the origin. Then these scalars form the ith principal components corresponding to the n data cases $\boldsymbol{x}_1, ..., \boldsymbol{x}_n$. So the first principal component is the projection on the major axis, the second principal component is the projection on the next longest axis, ..., the pth principal component is the projection on the minor axis. The axes are orthogonal, so the directions $\boldsymbol{e}_i$ are orthogonal.

When $\boldsymbol{\mu} \neq \boldsymbol{0}$ the projections on $\boldsymbol{e}_i$ are projections on the axes through the origin that are parallel to the axes of the hyperellipsoid. Figure 2.1 shows two ellipsoids where $p = 2$.

The first k principal components can be regarded as a good k dimensional approximation to the p dimensional data. Suppose the data cloud approximates the hyperellipsoid $\{\boldsymbol{x} | D^2_{\boldsymbol{x}} \leq h^2\}$ where $h^2 = D^2_{(n)}$, the largest squared distance, so the hyperellipsoid contains all of the data. Then a good one dimensional approximation is the projection on the major axis since this captures the dimension with the greatest variability or dispersion as measured by $\boldsymbol{\Sigma}$. A good two dimensional approximation uses the projection on the major axis and the projection on the next largest axis since these are the two orthogonal directions where the two projections have the greatest variability. Following Mardia, Kent and Bibby (1979, p. 220), if $\boldsymbol{S}$ (with centered data) or $\boldsymbol{R}$ is used as the dispersion matrix, then the vector space spanned by the first k principal components has smaller mean square deviation from the p variables than any other $k-$dimensional subspace.

Since $\boldsymbol{Z}$ represents a new coordinate system, the ith case $\boldsymbol{x}_i = (\boldsymbol{x}_i^T \hat{\boldsymbol{e}}_i)\hat{\boldsymbol{e}}_1 + \cdots + (\boldsymbol{x}_i^T \hat{\boldsymbol{e}}_p)\hat{\boldsymbol{e}}_p = Z_{i,1}\hat{\boldsymbol{e}}_1 + \cdots + Z_{i,p}\hat{\boldsymbol{e}}_p$. Also $\boldsymbol{x}_i = \tilde{\boldsymbol{x}}_i(k) + \boldsymbol{r}_i(k)$ where $\tilde{\boldsymbol{x}}_i(k) = \sum_{j=1}^k Z_{i,j}\hat{\boldsymbol{e}}_j$ and the residual vector $\boldsymbol{r}_i(k) = \sum_{j=k+1}^p Z_{i,j}\hat{\boldsymbol{e}}_j$. The squared length of the residual vector is $\|\boldsymbol{r}_i(k)\|^2 = \boldsymbol{r}_i(k)^T \boldsymbol{r}_i(k) = Z^2_{i,k+1} + \cdots + Z^2_{i,p}$.

Suppose $\boldsymbol{S}$ or $\boldsymbol{R}$ is used as the as the dispersion matrix and that $T = \boldsymbol{0}$ so the hyperellisoid is centered at the origin. Following Kendall (1980, p. 17), the eigenvector corresponding to the largest eigenvalue determines the major axis of the hyperellipsoid. This axis forms the line through the origin such that the sum of squared distances from the n data points $\boldsymbol{x}_i$ to this line is a minimum. If the data points are projected onto a hyperplane perpendicular to the major axis line, then the eigenvector corresponding to the next largest eigenvalue determines the second longest axis of the hyperellipsoid, and this axis is the line through the origin in the hyperplane that minimizes the sum of squared distances, and so on.

When the covariance matrix is used, that the first principal component $\boldsymbol{e}_1^T\boldsymbol{x}$ is the linear combination $\boldsymbol{g}_1^T\boldsymbol{x}$ that maximizes $\text{Var}(\boldsymbol{g}_1^T\boldsymbol{x})$ subject to $\boldsymbol{g}_1^T\boldsymbol{g}_1 = 1$, while the jth principal component is the linear combination $\boldsymbol{g}_j^T\boldsymbol{x}$ that maximizes $\text{Var}(\boldsymbol{g}_j^T\boldsymbol{x})$ subject to $\boldsymbol{g}_j^T\boldsymbol{g}_j = 1$ and $\text{Cov}(\boldsymbol{g}_j^T\boldsymbol{x}, \boldsymbol{g}_k^T\boldsymbol{x}) = 0$ for $k < j$. This result can be proved using Theorem 1.1.

Definition 6.3. A *scree plot* is a plot of component number versus eigenvalue.

Dimension reduction involves using the first k principal components to approximate the data matrix without losing much important information. Want the proportion of the trace explained by the first k principal components to be higher than 0.8 or 0.9.

Rule of thumb 6.2. The value of k should be such that

$$\frac{\sum_{i=1}^{k} \hat{\lambda}_i}{\sum_{i=1}^{p} \hat{\lambda}_i} \geq 0.9.$$

The scree plot is also useful for choosing k since often there is a sharp bend in the scree plot when the components are no longer important. See Cattell (1966).

Following Johnson and Wichern (1988, p. 343, 347), let $\boldsymbol{x} = (X_1, ..., X_p)$ be a random vector such that the $\boldsymbol{x}_i$ and $\boldsymbol{x}$ have the same distribution. Let $Y_i = \boldsymbol{e}_i^T \boldsymbol{x}$ be the population principal components based on the covariance matrix $\text{Cov}(\boldsymbol{x}) = \boldsymbol{\Sigma}_{\boldsymbol{x}}$. Let $\boldsymbol{e}_i = (e_{1i}, ..., e_{pi})^T$. Then e_{ki} is proportional to the correlation between Y_i and X_k, in fact,

$$\text{corr}(Y_i, X_k) = \frac{e_{ki}\sqrt{\lambda_i}}{\sqrt{\sigma_{kk}}}$$

for $i, k = 1, ..., p$. If the correlation matrix $\boldsymbol{\rho}$ is used instead of $\boldsymbol{\Sigma}_{\boldsymbol{x}}$, then $\text{corr}(Y_i, X_k) = e_{ki}\sqrt{\lambda_i}$.

Following Johnson and Wichern (1988, p. 252-253), some software that uses $\boldsymbol{S}$ or $\boldsymbol{R}$ centers the data by using $\boldsymbol{x}_i - \overline{\boldsymbol{x}}$. Centering does not change $\boldsymbol{S}$ or $\boldsymbol{R}$ but makes the ith principal component equal to $\hat{\boldsymbol{e}}_i^T(\boldsymbol{x} - \overline{\boldsymbol{x}})$ for observation $\boldsymbol{x}$.

Warning: If $\hat{\lambda}_p \approx 0$, then $\hat{\boldsymbol{\Sigma}}$ is nearly singular, and there could be an unnoticed linear dependency in the data set, eg $X_p \approx \sum_{i=1}^{p-1} c_i X_i$. Then one or more of the variables is redundant and should be deleted. Following Johnson and Wichern (1988, p. 360), suppose $p = 4$ and X_1, X_2 and X_3 are midterm exam scores while X_4 is the total of the midterm scores so that $X_4 = X_1 + X_2 + X_3$. Due to rounding, $\hat{\lambda}_4$ could be nonzero, but very close to zero.

6.2 Robust Principal Component Analysis

A robust "plug in" method uses an analysis based on the $(\hat{\lambda}_i, \hat{\boldsymbol{e}}_i)$ computed from a robust dispersion estimator $\boldsymbol{C}$. The RPCA method performs the classical principal component analysis on the RMVN subset, using either the sample covariance matrix $\boldsymbol{C}_U = \boldsymbol{S}_U$ or the sample correlation matrix $\boldsymbol{R}_U$. Under assumption (E1) from Chapter 4, $\boldsymbol{C}_U$ and $\boldsymbol{R}_U$ are $\sqrt{n}$ consistent highly outlier resistant estimators of $c\boldsymbol{\Sigma} = d\text{Cov}(\boldsymbol{x})$ and the population correlation matrix $\boldsymbol{D}\text{Cov}\big(\boldsymbol{x}\big)\boldsymbol{D} = \boldsymbol{\rho}$, respectively, where $\boldsymbol{D} = \text{diag}(1/\sqrt{\sigma}_{11}, ..., 1/\sqrt{\sigma}_{\text{pp}})$ and the σ_{ii} are the diagonal entries of $\text{Cov}(\boldsymbol{x}) = \boldsymbol{\Sigma}_{\boldsymbol{x}} = c_X\boldsymbol{\Sigma}$. Let $\lambda_i(\boldsymbol{A})$ be the eigenvalues of $\boldsymbol{A}$ where $\lambda_1(\boldsymbol{A}) \geq \lambda_2(\boldsymbol{A}) \geq \cdots \geq \lambda_p(\boldsymbol{A})$. Let $\hat{\lambda}_i(\hat{\boldsymbol{A}})$ be the eigenvalues of $\hat{\boldsymbol{A}}$ where $\hat{\lambda}_1(\hat{\boldsymbol{A}}) \geq \hat{\lambda}_2(\hat{\boldsymbol{A}}) \geq \cdots \geq \hat{\lambda}(\hat{\boldsymbol{A}})$.

Theorem 6.3. Under (E1), the correlation of the eigenvalues computed from the classical PCA and RPCA converges to 1 in probability.

Proof: The eigenvalues are continuous functions of the dispersion estimator, hence consistent estimators of dispersion give consistent estimators of the population eigenvalues. See Eaton and Tyler (1991) and Bhatia, Elsner and Krause (1990). Let $\lambda_i(\boldsymbol{\Sigma}) = \lambda_i$ be the eigenvalues of $\boldsymbol{\Sigma}$ so $c_X\lambda_i$ are the eigenvalues of $\text{Cov}(\boldsymbol{x}) = \boldsymbol{\Sigma}_{\boldsymbol{x}}$. Under (E1), $\lambda_i(\boldsymbol{S}) \overset{P}{\to} c_X\lambda_i$ and $\lambda_i(\boldsymbol{C}_U) \overset{P}{\to} c\lambda_i = \dfrac{c}{c_X}c_X\lambda_i = d\ c_X\ \lambda_i$. Hence the population eigenvalues of $\boldsymbol{\Sigma}_{\boldsymbol{x}}$ and $d\ \boldsymbol{\Sigma}_{\boldsymbol{x}}$ differ by the positive multiple d, and the population correlation of the two sets of eigenvalues is equal to one.

Now let $\lambda_i\big(\boldsymbol{\rho}\big) = \lambda_i$. Under (E1), both $\boldsymbol{R}$ and $\boldsymbol{R}_U$ converge to $\boldsymbol{\rho}$ in probability, so $\hat{\lambda}_i(\boldsymbol{R}) \overset{P}{\to} \lambda_i$ and $\hat{\lambda}_i(\boldsymbol{R}_U) \overset{P}{\to} \lambda_i$ for $i = 1, ..., p$. Hence the two population sets of eigenvalues are the same and thus have population correlation equal to one. $\square$

Note that if $\boldsymbol{\Sigma}_{\boldsymbol{x}}\ \boldsymbol{e} = \lambda\boldsymbol{e}$, then

$$d\ \boldsymbol{\Sigma}_{\boldsymbol{x}}\ \boldsymbol{e} = d\lambda\boldsymbol{e}.$$

Thus $\hat{\lambda}_i(\boldsymbol{S}) \overset{P}{\to} \lambda_i(\boldsymbol{\Sigma}_{\boldsymbol{x}})$ and $\hat{\lambda}_i(\boldsymbol{C}_U) \overset{P}{\to} d\lambda_i(\boldsymbol{\Sigma}_{\boldsymbol{x}})$ for $i = 1, ..., p$. Since plotting software fills space, two scree plots of two sets of eigenvalues that differ by a constant positive multiple will look nearly the same, except for the labels of the vertical axis, and the "trace explained" by the largest k eigenvalues will be the same for the two sets of eigenvalues. Theorem 6.2 implies that for a large class of elliptically contoured distributions and for large n, the classical and robust scree plots should be similar visually, and the "trace explained" by the classical PCA and the robust PCA should also be similar.

The eigenvectors are not continuous functions of the dispersion estimator, and the sample size may need to be massive before the robust and classical eigenvectors or principal components have high absolute correlation. In the software, sign changes in the eigenvectors are common, since $\boldsymbol{\Sigma}_{\boldsymbol{x}}\ \boldsymbol{e} = \lambda \boldsymbol{e}$ implies that $\boldsymbol{\Sigma}_{\boldsymbol{x}}\ (-\boldsymbol{e}) = \lambda(-\boldsymbol{e})$.

Table 6.1: Estimation of $\boldsymbol{\Sigma}$ with $\gamma = 0.4$, $n = 35p$

p	type	n	pm	Q
5	1	135	16	0.153
5	2	135	6	0.213
10	1	350	21	0.326
10	2	350	6	0.326
15	1	525	26	0.856
15	2	525	7	0.675
20	1	700	33	0.798
20	2	700	8	0.792
25	1	875	39	1.014
25	2	875	10	1.867

A simulation was done to check that RMVN estimates $\boldsymbol{\Sigma}$ if the clean data is MVN and γ is the percentage of outliers. The clean cases were MVN: $\boldsymbol{x} \sim N_p(\boldsymbol{0}, diag(1, 2, ..., p))$. Outlier types were $\boldsymbol{x} \sim N_p((0, ..., 0, pm)^T, 0.0001\boldsymbol{I}_p)$, a near point mass at the major axis, and the mean shift $\boldsymbol{x} \sim N_p(pm\boldsymbol{1}, diag\ (1, 2, ..., p))$ where $\boldsymbol{1} = (1, ..., 1)^T$. On clean MVN data, $n \geq 20p$ gave good results for $2 \leq p \leq 100$. For the contaminated MVN data, the first $n\gamma$ cases were outliers, and the classical estimator $\boldsymbol{S}_c$ was computed on the clean cases. The diagonal elements of $\boldsymbol{S}_c$ and $\hat{\boldsymbol{\Sigma}}_{RMVN}$ should both be estimating $(1, 2, ..., p)^T$. The average diagonal elements of both matrices were computed for 20 runs, and the criterion Q was the sum of the absolute differences of the p diagonal elements from the two averaged matrices. Since $\gamma = 0.4$ and the initial subsets for the RMVN estimator are half sets, the simulations used $n = 35p$. The values of Q shown in Table 6.1 correspond to good estimation of the diagonal elements. Values of pm slightly smaller than the tabled values led to poor estimation of the diagonal elements.

Example 6.1. Buxton (1920) gives various measurements on 87 men including *height*, *head length*, *nasal height*, *bigonal breadth* and *cephalic index*. Five *heights* were recorded to be about 19mm with the true heights

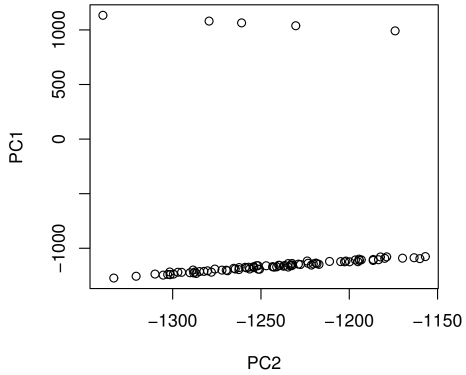

Figure 6.1: First Two Principal Components for Buxton data

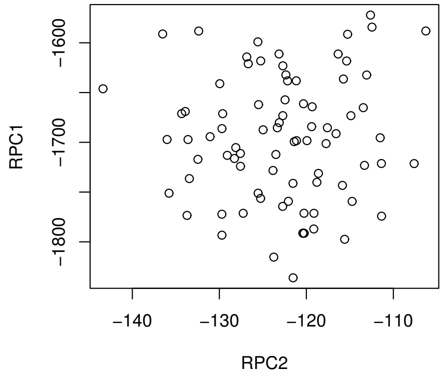

Figure 6.2: First Two Robust Principal Components with Outliers Omitted

recorded under head length. Performing a classical principal components analysis on these five variables using the covariance matrix resulted in a first principal component corresponding to a major axis that passed through the outliers. See Figure 6.1 where the second principal component is plotted versus the first. The robust PCA, or the classical PCA performed after the outliers are removed, resulted in a first principal component that was approximately $-$ *height* with $\hat{\boldsymbol{e}}_1 \approx (-1.000, 0.002, -0.023, -0.002, -0.009)^T$ while the second robust principal component was based on the eigenvector $\hat{\boldsymbol{e}}_2 \approx (-0.005, 0.848, -0.054, -0.048, 0.525)^T$. The plot of the first two robust principal components, with the outliers deleted, is shown in Figure 6.2. These two components explain about 86% of the variance.

The *R* function prcomp can be used to compute output. Suppose the data matrix is z. The commands

```
zz <- prcomp(z)
zz
```

will create and display output. The term *zz$sd* gives the square roots of the eigenvalues while the term *zz$rot* displays the eigenvectors using the covariance matrix. Hence Figure 6.1 can be made with the following commands.

```
z <- cbind(buxy,buxx)
zz <- prcomp(z)
PC1 <- z%*%zz$rot[,1]
PC2 <- z%*%zz$rot[,2]
plot(PC2,PC1)
```

It usually makes more sense to use the correlation matrix. the *mpack* function rprcomp does robust principal components. The two functions use "scale=T" or "cor=T" to use a correlation matrix.

```
zzcor <- prcomp(z,scale=T)
zrcor <- rprcomp(z,cor=T)
```

Then

```
zrcor$out$sd^2
```

gives the eigenvalues and *zrcor$out$rot* gives the eigenvectors. Scree plots can be made with the following commands, and Figure 6.3 shows the robust scree plot which suggests that the last principal component can be deleted.

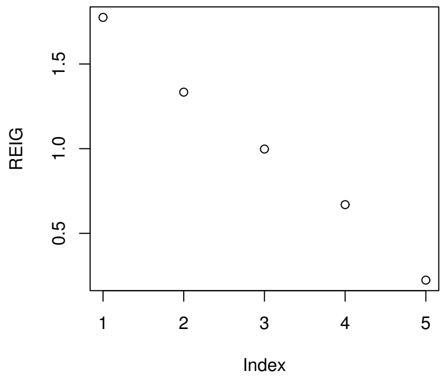

Figure 6.3: Robust Scree Plot

```
EIG <- zzcor$sd^2
plot(EIG)
#robust scree plot
REIG <- zrcor$out$sd^2
plot(REIG)
```

The outliers are known from the DD plot so the robust principal component analysis can be done with and without the outliers. The data matrix *zw* is the clean data without the outliers.

```
zw <-z[-c(61,62,63,64,65),]
zzcorc <- prcomp(zw,scale=T)
# clean data with corr matrix
> zzcorc
Standard deviations:
[1] 1.3184358 1.1723991 1.0155266 0.7867349 0.4867867
Rotation:
          PC1      PC2     PC3      PC4      PC5
buxy      0.01551  0.71466 0.02247 -0.68890 -0.11806
len       0.70308 -0.06778 0.07744 -0.16901  0.68302
nasal     0.15038  0.68868 0.02042  0.70385  0.08539
bigonal   0.11646 -0.04882 0.96504  0.02261 -0.22855
cephalic -0.68502  0.08950 0.24854 -0.03071  0.67825
```

```
zrcor <- rprcomp(z,cor=T)
> zrcor
$out
Standard deviations:
[1] 1.3323400 1.1548879 0.9988643 0.8182741 0.4730769
Rotation:
              PC1       PC2       PC3       PC4       PC5
buxy     -0.10724 -0.69431 -0.11325  0.69184 -0.12238
len       0.69909 -0.06324  0.02560  0.17129  0.69085
nasal     0.04094 -0.70310 -0.08718 -0.70093  0.07123
bigonal   0.02638 -0.13994  0.98660  0.01120 -0.07884
cephalic -0.70527 -0.00317  0.07443  0.02432  0.70460

> zrcorc <- rprcomp(zw,cor=T)
> zrcorc
$out
Standard deviations:
[1] 1.3369152 1.1466891 1.0016463 0.8123854 0.4842482
Rotation:
              PC1       PC2       PC3       PC4       PC5
buxy     -0.21306  0.67557 -0.01727 -0.68852 -0.15446
len       0.67272  0.21639  0.05560 -0.15178  0.68884
nasal    -0.22213  0.66958  0.05174  0.68978  0.15441
bigonal  -0.01374 -0.02995  0.99668 -0.03546 -0.06543
cephalic -0.67270 -0.21807  0.02363 -0.16076  0.68813
```

Note that the square roots of the eigenvalues, given by "Standard deviations," do not change much for the following three estimators: the classical estimator applied to the clean data, and the robust estimator applied to the full data or the clean data. The first eigenvector is roughly proportional to *length* $-$ *cephalic* while the second eigenvector is roughly proportional to *buxy* + *nasal*. The third principal component is highly correlated with *bigonal*, the fourth principal component is proportional to *buxy* $-$ *nasal*, and the fifth principal component to *length* + *cephalic*.

In simulations for principal component analysis, FCH, RMVN, OGK and Fake-MCD seem to estimate $c\boldsymbol{\Sigma}_{\boldsymbol{x}}$ if $\boldsymbol{x} = \boldsymbol{A}\boldsymbol{z} + \boldsymbol{\mu}$ where $\boldsymbol{z} = (z_1, ..., z_p)^T$ and the z_i are iid from a continuous distribution with variance σ^2. Here

$\boldsymbol{\Sigma}_{\boldsymbol{x}} = \text{Cov}(\boldsymbol{x}) = \sigma^2 \boldsymbol{A}\boldsymbol{A}^T$. The bias for the MB estimator seemed to be small. It is known that affine equivariant estimators give unbiased estimators of $c\boldsymbol{\Sigma}_{\boldsymbol{x}}$ if the distribution of z_i is also symmetric. DGK and Fake-MCD (with fixed random number seed) are affine equivariant. FCH and RMVN are asymptotically equivalent to a scaled DGK estimator. But in the simulations the results also held for skewed distributions.

The simulations used 1000 runs where $\boldsymbol{x} = \boldsymbol{A}\boldsymbol{z}$ and $\boldsymbol{z} \sim N_p(\boldsymbol{0}, \boldsymbol{I}_p)$, $\boldsymbol{z} \sim LN(\boldsymbol{0}, \boldsymbol{I}_p)$ where the marginals are iid lognormal(0,1), or $\boldsymbol{z} \sim MVT_p(1)$, a multivariate t distribution with 1 degree of freedom so the marginals are iid Cauchy(0,1). The choice $\boldsymbol{A} = diag(\sqrt{1}, ..., \sqrt{p})$ results in $\boldsymbol{\Sigma} = diag(1, ..., p)$. Note that the population eigenvalues will be proportional to $(p, p-1, ..., 1)^T$ and the population "variance explained" by the ith principal component is $\lambda_i / \sum_{j=1}^p \lambda_j = 2(p+1-i)/[p(p+1)]$. For $p = 4$, these numbers are 0.4, 0.3 and 0.2 for the first three principal components. If the "correlation" option is used, then the population "correlation matrix" is the identity matrix $\boldsymbol{I}_p$, the ith population eigenvalue is proportional to $1/p$ and the population "variance explained" by the ith principal component is $1/p$.

Table 6.2 shows the mean "variance explained" along with the standard deviations for the first three principal components. Also a_i and p_i are the average absolute value of the correlation between the ith eigenvectors or the ith principal components of the classical and robust methods. Two rows were used for each "n–data type" combination. The a_i are shown in the top row while the p_i are in the lower row. The values of a_i and p_i were similar. The standard deviations were slightly smaller for the classical PCA for normal data. The classical method failed to estimate (0.4,0.3,0.2) for the Cauchy data. For the lognormal data, RPCA gave better estimates, and the p_i were not high except for $n = 10000$.

To compare affine equivariant and non-equivariant estimators, Maronna and Zamar (2002) suggest using $\boldsymbol{A}_{i,i} = 1$ and $\boldsymbol{A}_{i,j} = \rho$ for $i \neq j$ and $\rho = 0, 0.5, 0.7, 0.9,$ and 0.99. Then $\boldsymbol{\Sigma} = \boldsymbol{A}^2$. If ρ is high, or if p is high and $\rho \geq 0.5$, then the data are concentrated about the line with direction $\boldsymbol{1} = (1, ..., 1)^T$. For $p = 50$ and $\rho = 0.99$, the population variance explained by the first principal component is 0.999998. If the "correlation" option is used, then there is still one extremely dominant principal component unless both p and ρ are small.

Table 6.3 shows the mean "variance explained" along with the standard deviations multiplied by 10^7 for the first principal component. The a_1 value is given but p_1 was always 1.0 to many decimal places even with Cauchy data.

Table 6.2: Variance Explained by PCA and RPCA, $p = 4$

n	type	M/S	vexpl	rvexpl	a_1/p_1	a_2/p_2	a_3/p_3
40	N	M	0.445,0.289,0.178	0.472,0.286,0.166	0.895	0.821	0.825
		S	0.050,0.037,0.032	0.062,0.043,0.037	0.912	0.813	0.804
100	N	M	0.419,0.295,0.191	0.425,0.293,0.189	0.952	0.926	0.963
		S	0.033,0.030,0.024	0.040,0.032,0.027	0.956	0.923	0.953
400	N	M	0.404,0.298,0.198	0.406,0.298,0.198	0.994	0.991	0.996
		S	0.019,0.017,0.014	0.021,0.019,0.015	0.995	0.990	0.994
40	C	M	0.765,0.159,0.056	0.514,0.275,0.147	0.563	0.519	0.511
		S	0.165,0.112,0.051	0.078,0.055,0.040	0.776	0.383	0.239
100	C	M	0.762,0.156,0.060	0.455,0.286,0.173	0.585	0.527	0.528
		S	0.173,0.112,0.055	0.054,0.041,0.034	0.797	0.377	0.269
400	C	M	0.756,0.162,0.060	0.413,0.296,0.194	0.608	0.562	0.575
		S	0.172,0.113,0.054	0.030,0.025,0.022	0.796	0.397	0.308
40	L	M	0.539,0.256,0.139	0.521,0.268,0.146	0.610	0.509	0.530
		S	0.127,0.075,0.054	0.099,0.061,0.047	0.643	0.439	0.398
100	L	M	0.482,0.270,0.165	0.459,0.279,0.172	0.647	0.555	0.566
		S	0.180,0.063,0.052	0.077,0.047,0.041	0.654	0.492	0.474
400	L	M	0.437,0.282,0.185	0.416,0.290,0.194	0.748	0.639	0.739
		S	0.080,0.048,0.044	0.049,0.035,0.033	0.727	0.594	0.690
10000	L	M	0.400,0.301,0.200	0.402,0.300,0.199	0.982	0.967	0.991
		S	0.027,0.023,0.018	0.013,0.011,0.009	0.976	0.967	0.989

Table 6.3: Variance Explained by PCA and RPCA, SSD = 10^7 SD, $p = 50$

n	type	vexpl	SSD	rvexpl	SSD	a_1
200	N	0.999998	1.958	0.999998	2.867	0.687
1000	N	0.999998	0.917	0.999998	0.971	0.944
1000	C	0.999996	161.3	0.999998	1.482	0.112
1000	L	0.999998	0.919	0.999998	1.508	0.175

Hence the eigenvectors from the robust and classical methods could have low absolute correlation, but the data was so tightly clustered that the first principal components from the robust and classical methods had absolute correlation near 1.

6.3 Summary

1) Let $\boldsymbol{\Sigma} = ((\sigma_{ij}))$ be a positive definite symmetric $p \times p$ dispersion matrix. A *generalized correlation matrix* $\boldsymbol{\rho} = \big((\rho_{ij})\big)$ where

$$\rho_{ij} = \frac{\sigma_{ij}}{\sqrt{\sigma_{ii}\sigma_{jj}}}.$$

The generalized correlation matrix is the correlation matrix when second moments exist if $\boldsymbol{\Sigma} = c \text{ Cov}(\boldsymbol{x})$ for some constant $c > 0$.

2) Classical principal component analysis (PCA) gets the eigenvalues and eigenvectors $(\hat{\lambda}_i, \hat{\boldsymbol{e}}_i)$ of the sample covariance matrix $\boldsymbol{S}$ or of the sample correlation matrix $\boldsymbol{R}$.

3) Let U be the subset of at least half of the cases from which the robust estimator is computed. Let $\boldsymbol{S}_U$ and $\boldsymbol{R}_U$ denote the sample covariance matrix and sample correlation matrix computed from the cases in $\boldsymbol{U}$. Then the robust estimator $\boldsymbol{C} = d\boldsymbol{S}_U$ for some constant $d > 0$ and $\boldsymbol{R}_U$ is the generalized correlation matrix corresponding to $\boldsymbol{C}$. The robust PCA uses U corresponding to the RMVN estimator.

4) Want $n > 10p$ for the classical PCA and $n > 20p$ for the robust PCA.

5) Both *R* and *SAS* output give the eigenvectors as shown in symbols for the following table.

PC1	PC2	$\cdots$	PCp
$\hat{\boldsymbol{e}}_1$	$\hat{\boldsymbol{e}}_2$	$\cdots$	$\hat{\boldsymbol{e}}_p$

R output shows the square roots of the eigenvalues

$$\sqrt{\hat{\lambda}_1}, \sqrt{\hat{\lambda}_2}, ..., \sqrt{\hat{\lambda}_p}$$

while *SAS* output gives the eigenvalues $\hat{\lambda}_i$.

6) Given the eigenvalues or square roots of the eigenvalues, be able to sketch a
scree plot of i versus $\hat{\lambda}_i$.

7) The *trace explained* or *variance explained* by the first k principal components is $\frac{\sum_{i=1}^k \hat{\lambda}_i}{\sum_{i=1}^p \hat{\lambda}_i}$ where the denominator is equal to p if the correlation option $\boldsymbol{R}$ or $\boldsymbol{R}_U$ is used, as recommended in point 10).

8) Use k principal components if the trace explained is bigger than some percentage like 90%, 80% or 70%. There is often a sharp bend in the scree plot when the components are no longer useful.

9) When $\boldsymbol{R}$ or $\boldsymbol{R}_U$ is used, the correlation of the ith variable with the jth principal component is proportional to the ith entry of the jth eigenvector $\hat{\boldsymbol{e}}_j$. To try to explain the jth principal component, look at entries in $\hat{\boldsymbol{e}}_j$ that are large in magnitude and ignore entries close to zero. Sometimes only one entry is large. Sometimes all of the large entries have approximately the same size and sign, then the principal component is interpreted as an average of these entrees. If exactly two entries are of similar large magnitude but of different sign, the principal component is interpreted as a difference of the two entrees. If there are $j \geq 2$ large entrees that differ in magnitude, then the principal component is interpreted as a linear combination of the corresponding variables.

10) PCA based on $\boldsymbol{R}$ or $\boldsymbol{R}_U$ is easier to interpret than PCA based on $\boldsymbol{S}$ or $\boldsymbol{S}_U$.

i) If $\boldsymbol{S}$ is used, the variance explained by the first principal component could be large because one variable has much larger variance than the other variables.

ii) If $\boldsymbol{S}$ is used, the correlation of the ith variable with the jth principal component is proportional to the ith entry of the jth eigenvector $\hat{\boldsymbol{e}}_j$ divided by the standard deviation of ith variable: $e_{ij}/\sqrt{S_{ii}}$.

Hence PCA based on $\boldsymbol{S}$ is harder to interpret if p random variables do not have similar sample variances. The variances could differ if different units are used or if some variables are transformed while others are not. Hence PCA based on $\boldsymbol{R}$ or $\boldsymbol{R}_U$ is recommended.

```
11) Typical Routput is shown. Standard deviations:
[1] 1.3369152 1.1466891 1.0016463 0.8123854 0.4842482
Rotation: PC1         PC2         PC3         PC4         PC5
len        0.67271620 -0.21639022  0.05559575  0.15178244 -0.68883916
nasal     -0.22213361 -0.66957907  0.05173705 -0.68978370 -0.15440936
bigonal   -0.01373814  0.02995162  0.99668240  0.03545927  0.06542933
```

```
cephalic -0.67269993  0.21806615  0.02362841  0.16076405 -0.68812686
buxy     -0.21306252 -0.67556583 -0.01727087  0.68851877  0.15446292
```

12) Let $\hat{\boldsymbol{\Sigma}}$ be a consistent estimator of $\boldsymbol{\Sigma}$. The following theorems show that asymptotically, the eigenvalues and eigenvectors of $\hat{\boldsymbol{\Sigma}}$ act as those of $\boldsymbol{\Sigma}$ and vice verca. This result is useful since eigenvectors are not continuous functions of the dispersion matrix. The following theorem holds because eigenvalues and the generalized correlation matrix are continuous functions of the dispersion matrix.

i) **Theorem 6.1.** Suppose the dispersion matrix $\boldsymbol{\Sigma}$ has eigenvalue eigenvector pairs $(\lambda_1, \boldsymbol{e}_1), ..., (\lambda_p, \boldsymbol{e}_p)$ where $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p$. Suppose $\hat{\boldsymbol{\Sigma}} \xrightarrow{P} c\boldsymbol{\Sigma}$ for some constant $c > 0$. Let the eigenvalue eigenvector pairs of $\hat{\boldsymbol{\Sigma}}$ be $(\hat{\lambda}_1, \hat{\boldsymbol{e}}_1), ..., (\hat{\lambda}_p, \hat{\boldsymbol{e}}_p)$ where $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \cdots \geq \hat{\lambda}_p$. Then $\hat{\lambda}_j(\hat{\boldsymbol{\Sigma}}) \xrightarrow{P} c\lambda_j(\boldsymbol{\Sigma}) = c\lambda_j$, $\hat{\boldsymbol{\rho}} \xrightarrow{P} \boldsymbol{\rho}$ and $\hat{\lambda}_j \left(\hat{\boldsymbol{\rho}}\right) \xrightarrow{P} \lambda_j \left(\boldsymbol{\rho}\right)$ where $\lambda_j(\boldsymbol{A})$ is the jth eigenvalue of $\boldsymbol{A}$ for $j = 1, ..., p$.

ii) **Theorem 6.2.** Assume the $p \times p$ symmetric dispersion matrix $\boldsymbol{\Sigma}$ is positive definite. a) If $\hat{\boldsymbol{\Sigma}} \xrightarrow{P} \boldsymbol{\Sigma}$, then $\hat{\boldsymbol{\Sigma}}\boldsymbol{e}_i - \hat{\lambda}_i\boldsymbol{e}_i \xrightarrow{P} \boldsymbol{0}$.

b) If $\hat{\boldsymbol{\Sigma}} \xrightarrow{P} \boldsymbol{\Sigma}$, then $\boldsymbol{\Sigma}\hat{\boldsymbol{e}}_i - \lambda_i\hat{\boldsymbol{e}}_i \xrightarrow{P} \boldsymbol{0}$.

If $\hat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma} = O_P(n^{-\delta})$ where $0 < \delta \leq 0.5$, then

c) $\lambda_i\boldsymbol{e}_i - \hat{\boldsymbol{\Sigma}}\boldsymbol{e}_i = O_P(n^{-\delta})$, and

d) $\hat{\lambda}_i\hat{\boldsymbol{e}}_i - \boldsymbol{\Sigma}\hat{\boldsymbol{e}}_i = O_P(n^{-\delta})$.

e) If $\hat{\boldsymbol{\Sigma}} \xrightarrow{P} c\boldsymbol{\Sigma}$ for some constant $c > 0$, and if the eigenvalues $\lambda_1 > \cdots > \lambda_p > 0$ of $\boldsymbol{\Sigma}$ are unique, then the absolute value of the correlation of $\hat{\boldsymbol{e}}_j$ with $\boldsymbol{e}_j$ converges to 1 in probability: $|\text{corr}(\hat{\boldsymbol{e}}_j, \boldsymbol{e}_j)| \xrightarrow{P} 1$.

iii) **Theorem 6.3.** Under (E1), the correlation of the eigenvalues computed from the classical PCA and robust PCA converges to 1 in probability.

13) Centering uses $\boldsymbol{w}_i = \boldsymbol{x}_i - T$ where T is the sample mean or the sample mean of the standardized data for the full data set or for the set U used to compute the robust estimator. Centering does not change $\boldsymbol{S}, \boldsymbol{S}_U, \boldsymbol{R}$ or $\boldsymbol{R}_U$, but the jth principal component is $\hat{\boldsymbol{e}}_j^T\boldsymbol{w}_i = \hat{\boldsymbol{e}}_j^T(\boldsymbol{x}_i - T)$.

14) For PCA, the `summary(out)` statement shows

Importance of components:	PC1	PC2	$\cdots$	PCk	$\cdots$	PCp
Standard deviation	$\sqrt{\hat{\lambda}_1}$	$\sqrt{\hat{\lambda}_2}$	$\cdots$	$\sqrt{\hat{\lambda}_k}$	$\cdots$	$\sqrt{\hat{\lambda}_p}$
Proportion of variance	$\frac{\hat{\lambda}_1}{\sum_{i=1}^p \hat{\lambda}_i}$	$\frac{\hat{\lambda}_2}{\sum_{i=1}^p \hat{\lambda}_i}$	$\cdots$	$\frac{\hat{\lambda}_k}{\sum_{i=1}^p \hat{\lambda}_i}$	$\cdots$	$\frac{\hat{\lambda}_p}{\sum_{i=1}^p \hat{\lambda}_i}$
Cumulative Proportion	$\frac{\hat{\lambda}_1}{\sum_{i=1}^p \hat{\lambda}_i}$	$\frac{\sum_{j=1}^2 \hat{\lambda}_j}{\sum_{i=1}^p \hat{\lambda}_i}$	$\cdots$	$\frac{\sum_{j=1}^k \hat{\lambda}_j}{\sum_{i=1}^p \hat{\lambda}_i}$	$\cdots$	1

Recall that if $\boldsymbol{R}$ or $\boldsymbol{R}_U$ is used, then $\sum_{i=1}^p \hat{\lambda}_i = p$. Typically want to keep the first m principal components where $\dfrac{\sum_{j=1}^m \hat{\lambda}_j}{\sum_{i=1}^p \hat{\lambda}_i} > a$ where the threshold a is a number like 0.9, 0.8 or 0.7.

15) For PCA, a *biplot* is a plot of the first principal component versus the second principal component. The plotted points are $\hat{\boldsymbol{e}}_j^T \boldsymbol{x}_i$ for $j = 1, 2$ where the classical biplot uses $i = 1, ..., n$ and the robust plot uses cases in the RMVN set U. Let $\hat{\boldsymbol{e}}_j = (\hat{e}_{1j}, \hat{e}_{2j}, ..., \hat{e}_{pj})^T$. Then $\hat{e}_{kj}$ is called the *loading* of the kth variable on the jth principal component. An arrow with the kth variable name is the vector from the origin $(0, 0)^T$ to the loadings $(\hat{e}_{k1}, \hat{e}_{k2})^T$. So if the arrow is in the first quadrant, both loadings are positive, etc. If the arrow is long to the right but short down, then the loading with the first principal component is large and positive while the loading with the second principal component is small and negative. Be able to interpret the classical and robust biplots.

6.4 Complements

Suppose $\boldsymbol{Z}$ is the standardized $n \times p$ data matrix and $\boldsymbol{Y} = \boldsymbol{Z}/\sqrt{n-1}$. If $n < p$, then the correlation matrix $\boldsymbol{R} = \boldsymbol{Y}^T\boldsymbol{Y} = \boldsymbol{Z}^T\boldsymbol{Z}/(n-1)$ does not have full rank. By singular value decomposition (SVD) theory, the SVD of $\boldsymbol{Y}$ is $\boldsymbol{Y} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{V}^T$ where the positive singular values are square roots of the positive eigenvalues of both $\boldsymbol{Y}^T\boldsymbol{Y}$ and of $\boldsymbol{Y}\boldsymbol{Y}^T$. Also $\boldsymbol{V} = (\hat{\boldsymbol{e}}_1 \ \hat{\boldsymbol{e}}_2 \ \cdots \ \hat{\boldsymbol{e}}_p)$, and $\boldsymbol{Y}^T\boldsymbol{Y}\hat{\boldsymbol{e}}_i = \sigma_i^2 \hat{\boldsymbol{e}}_i$. Hence classical principal component analysis on the standardized data can be done using $\hat{\boldsymbol{e}}_i$ and $\hat{\lambda}_i = \sigma_i^2$. The SVD of $\boldsymbol{Y}^T$ is

$\boldsymbol{V}\boldsymbol{\Lambda}^T\boldsymbol{U}^T$, and

$$\boldsymbol{Y}\boldsymbol{Y}^T = \frac{1}{n-1}\begin{bmatrix} \boldsymbol{z}_1^T\boldsymbol{z}_1 & \boldsymbol{z}_1^T\boldsymbol{z}_2 & \dots & \boldsymbol{z}_1^T\boldsymbol{z}_n \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{z}_n^T\boldsymbol{z}_1 & \boldsymbol{z}_n^T\boldsymbol{z}_2 & \dots & \boldsymbol{z}_n^T\boldsymbol{z}_n \end{bmatrix}$$

which is the matrix of scalar products divided by $(n-1)$. For more information about the SVD, see Datta (1995, p. 552-556).

It may be possible to do robust PCA when $n < p$ by standardizing the data with the MED(X_i) and MAD(X_i). Then plot the Euclidean distaces of the standardized data from the coordinatewise median MED$(\boldsymbol{Z})$ and delete outliers, leaving m cases in an $m \times p$ matrix $\boldsymbol{Y}$. Then use the SVD of $\boldsymbol{Y}$ to perform a "robust" PCA.

Jolliffe (2010) is an authoritative text on PCA. Cattell (1966) and Bentler and Yuan (1998) are good references for scree plots. Mϕller, von Frese and Bro (2005) discuss PCA, principal component regression and drawbacks of M estimators. Waternaux (1976) and Tyler (1983) give some large sample theory for PCA. In particular, if the $\boldsymbol{x}_i$ are iid from a multivariate distribution with fourth moments and a covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{x}}$ such that the eigenvalues are distinct and positive, then $\sqrt{n}(\hat{\lambda}_i - \lambda_i) \xrightarrow{D} N(0, \kappa_i + 2\lambda_i^2)$ where κ_i is the kurtosis of the marginal distribution of x_i, for $i = 1, ..., p$.

The literature for robust PCA is large, but the "high breakdown" methods are impractical or not backed by theory. Some of these methods may be useful as outlier diagnostics. The theory of Boente (1987) for mildly outlier resistant principal components is not based on DGK estimators since the weighting function on the D_i is continuous. Spherical principal components is a mildly outlier resistant bounded influence approach suggested by Locantore, Marron, Simpson, Tripoli, Zhang and Cohen (1999). Boente and Fraiman (1999) claim that basis of the eigenvectors is consistently estimated by spherical principal components for elliptically contoured distributions. Also see Maronna, Martin and Yohai (2006, p. 212-213) and Taskinen, Koch and Oja (2012).

Bali, Boente, Tyler and Wang (2011) gave possibly impressive theory for infinite complexity impractical robust projection estimators, but should have given theory for the practical Fake-projection estimator actually used. This "bait and switch hoax" occurs far too often in multivariate "robust statistics" papers.

To estimate the first principal direction for principal component analysis, the Fake-projection (CR) estimator uses n projections $\boldsymbol{z}_i = \boldsymbol{w}_i/\|\boldsymbol{w}_i\|$ where $\boldsymbol{w}_i = \boldsymbol{y}_i - \hat{\boldsymbol{\mu}}_n$. Note that for $p = 2$ one can select 360 projections through the origin and a point on the unit circle that are one degree apart. Then there is a projection that is highly correlated with any projection on the unit circle. If $p = 3$, then 360 projections are not nearly enough to adequately approximate all projections through the unit sphere. Since the surface area of a unit hypersphere is proportional to n^{p-1}, approximations rapidly get worse as p increases.

Theory for the Fake-projection (CR) estimator may be simple. Suppose the data is multivariate normal $N_p(\boldsymbol{0}, diag(p, 1, ..., 1))$. Then $\boldsymbol{\beta} = (1, 0, ..., 0)^T$ (or $-\boldsymbol{\beta}$) is the population first direction. Heuristically, assume $\hat{\boldsymbol{\mu}}_n = \boldsymbol{0}$, although in general $\hat{\boldsymbol{\mu}}_n$ should be a good $\sqrt{n}$ consistent estimator of $\boldsymbol{\mu}$ such as the coordinatewise median. Let $\boldsymbol{b}_o$ be the "best" estimated projection $\boldsymbol{z}_j$ that minimizes $\|\boldsymbol{z}_i - \boldsymbol{\beta}\|$ for $i = 1, ..., n$. "Good" projections will have a $\boldsymbol{y}_i$ that lies in one of two "hypercones" with a vertex at the origin and centered about a line through the origin and $\pm\boldsymbol{\beta}$ with radius r at $\pm\boldsymbol{\beta}$. So for $p = 2$ the two "cones" are determined by the two lines through the origin with slopes $\pm\ r$. The probability that a randomly selected $\boldsymbol{y}_i$ falls in one of the two "hypercones" is proportional to r^{p-1}, and for $\boldsymbol{b}_o$ to be consistent for $\boldsymbol{\beta}$ need $r \rightarrow 0$, P(at least one $\boldsymbol{y}_i$ falls in "hypercone") $\rightarrow 1$ and $n \rightarrow \infty$. If these heuristics are correct, need $r \propto n^{\frac{-1}{p-1}}$ for $\|\boldsymbol{b}_o - \boldsymbol{\beta}\| = O_P(n^{\frac{1}{p-1}})$. Note that $\boldsymbol{b}_o$ is not an estimator since $\boldsymbol{\beta}$ is not known, but the rate of the "best" projection $\boldsymbol{b}_o$ gives an upper bound on the rate of the Fake-projection estimator $\boldsymbol{v}_1$ since $\|\boldsymbol{v}_1 - \boldsymbol{\beta}\| \geq \|\boldsymbol{b}_o - \boldsymbol{\beta}\|$. If the scale estimator is $\sqrt{n}$ consistent, then for a large class of elliptically contoured distributions, a conjecture is that $\|\boldsymbol{v}_1 - \boldsymbol{\beta}\| = O_P(n^{\frac{1}{2(p-1)}})$ for $p > 1$.

Simulations were done in *R*. The `MASS` library was used to compute FMCD and the `robustbase` library was used to compute OGK. The *mpack* function `covrmvn` computes the FCH, RMVN and MB estimators while `covfch` computes the FCH, RFCH and MB estimators. The following functions were used in the three simulations and have more outlier configurations than the two described in the text. Function `covesim` was used to produce Table 6.1 and `pcasim` for Tables 6.2 and 6.3. See Zhang (2011) for more extensive simulations.

For a nonsingular matrix, the inverse of the matrix, the determinant of the matrix and the eigenvalues of the matrix are continuous functions of the matrix. Hence if $\hat{\boldsymbol{\Sigma}}$ is a consistent estimator of $\boldsymbol{\Sigma}$, then the inverse, determinant and eigenvalues of $\hat{\boldsymbol{\Sigma}}$ are consistent estimators of the inverse, determinant and eigenvalues of $\boldsymbol{\Sigma}$. See, for example, Bhatia, Elsner and Krause (1990), Stewart (1969) and Severini (2005, p. 348-349).

6.5 Problems

PROBLEMS WITH AN ASTERISK * ARE ESPECIALLY USEFUL.

6.1*. Assume the $p \times p$ dispersion matrix $\boldsymbol{\Sigma}$ is positive definite. If $\hat{\boldsymbol{\Sigma}} \xrightarrow{P} c\boldsymbol{\Sigma}$ for some constant $c > 0$, prove that $\boldsymbol{\Sigma}\hat{\boldsymbol{e}}_i - \lambda_i \hat{\boldsymbol{e}}_i \xrightarrow{P} \boldsymbol{0}$.

6.2. Shown below is PCA output using the correlation matrix for the Buxton data where 5 outliers were deleted. The variables were *length, nasal height, bigonal breadth, cephalic* and *buxy* = *height*/20. The "standard deviations" line corresponds to the square roots of the eigenvalues. The Rotation matrix gives the 5 principal components.

a) For the robust `rprcomp` output make a scree plot. What proportion of the trace is explained by the first 4 principal components?

b) Which principal component corresponds to i) bigonal, ii) nasal + buxy, iii) length + cephalic, iv) length − cephalic and v) nasal − buxy?

```
rprcomp(z)
$out
Standard deviations:
[1] 1.3369152 1.1466891 1.0016463 0.8123854 0.4842482

Rotation:
                PC1         PC2         PC3         PC4         PC5
len       0.67271620 -0.21639022  0.05559575  0.15178244 -0.68883916
nasal    -0.22213361 -0.66957907  0.05173705 -0.68978370 -0.15440936
bigonal  -0.01373814  0.02995162  0.99668240  0.03545927  0.06542933
cephalic -0.67269993  0.21806615  0.02362841  0.16076405 -0.68812686
buxy     -0.21306252 -0.67556583 -0.01727087  0.68851877  0.15446292

prcomp(z,scale=T)
Standard deviations:
```

```
[1] 1.3184358 1.1723991 1.0155266 0.7867349 0.4867867

Rotation:
                 PC1         PC2        PC3         PC4        PC5
len      -0.70308364 -0.06777853 0.07743938  0.16900791  0.6830219
nasal    -0.15038248  0.68867720 0.02042098 -0.70384733  0.0853859
bigonal  -0.11646120 -0.04882199 0.96504341 -0.02261327 -0.2285455
cephalic  0.68502160  0.08950469 0.24854103  0.03070660  0.6782468
buxy     -0.01551443  0.71465734 0.02246533  0.68889840 -0.1180614
```

6.3. Let $Y_j = \boldsymbol{e}_j^T \boldsymbol{x}$ be the first population principal component where $\text{Cov}(\boldsymbol{x}) = \boldsymbol{\Sigma}_{\boldsymbol{x}}$.

a) Using $\text{Cov}(\boldsymbol{Ax}, \boldsymbol{Bx}) = \boldsymbol{A\Sigma_x B}^T$, show $\text{Cov}(\boldsymbol{x}, Y_j) = \boldsymbol{\Sigma_x e}_j = \lambda_j \boldsymbol{e}_j$.

b) Now $V(Y_j) = \text{Cov}(\boldsymbol{e}_j^T \boldsymbol{x}, \boldsymbol{e}_j^T \boldsymbol{x})$. Show that $V(Y_j) = \lambda_j$.

c) Let $\boldsymbol{x} = (X_1, ..., X_p)^T$ where X_i is the ith random variable with $V(X_i) = \sigma_{ii}$ and by a) $\text{Cov}(X_i, Y_j) = \lambda_j e_{ij}$ where $\boldsymbol{e}_j = (e_{1j}, ..., e_{ij}, ..., e_{pj})^T$. Find $\text{corr}(X_i, Y_j)$.

6.4. The classical PCA output below is for the Buxton data described in Problem 6.2 where 5 cases have massive outliers in the height and length variables. Interpret PC1 and PC2.

```
prcomp(z,scale=T)
[1] 1.431 1.074 0.964 0.926 0.106
        PC1    PC2    PC3    PC4    PC5
len   0.685  0.037  0.004 -0.189 -0.702
nas  -0.199  0.568  0.153 -0.783  0.047
big  -0.049 -0.569  0.783 -0.247 -0.007
ceph -0.100 -0.594 -0.603 -0.523  0.008
ht   -0.692 -0.000 -0.008  0.131 -0.710
```

6.5. SAS output for PCA using the correlation matrix is shown below. The Khattree and Naik (1999, p. 11) cork data gives the weights of cork borings in four directions for 28 trees in a block of plantations.

a) What is the variance explained by the first two principal components?

b) Interpret the first principal component.

```
                 Eigenvalues of the Covariance Matrix
            Eigenvalue    Difference    Proportion    Cumulative
       1      3.5967        3.3431        0.8992        0.8992
       2      0.2536        0.1735        0.0634        0.9626
       3      0.0801        0.0107        0.0200        0.9826
       4      0.0694                      0.0174        1.0000
                           Eigenvectors
             Prin1      Prin2      Prin3        Prin4
       north -0.5108992  0.1267234  0.803287920  0.2786606
       east  -0.4829921  0.7604818 -0.328918253 -0.2831940
       south -0.5082783 -0.3006659 -0.496526386  0.6361719
       west  -0.4973468 -0.5614345  0.001687729 -0.6613884
```

```
Rotation:  PC1        PC2        PC3
length 0.5771831 -0.5884323 -0.5662218
width  0.5811769 -0.1910978  0.7910215
height 0.5736663  0.7856393 -0.2316848

> summary(out$out)
Importance of components:PC1     PC2     PC3
Standard deviation     1.7065 0.25601 0.14961
Proportion of Variance 0.9707 0.02185 0.00746
Cumulative Proportion  0.9707 0.99254 1.00000
```

6.6. The Johnson and Wichern (1988, p. 262) turtle data has $X_1 =$ *length*, $X_2 =$ *width* and $X_3 =$ *height* for painted turtle shells with 48 cases. Principal component analysis output is shown above based on the (robust) correlation matrix.

a) How many principal components are needed?

b) Interpret the first principal component.

6.7. The output below describes lawyers' ratings of state judges in the US Superior Court with 43 observations on 12 numeric variables: CONT Number of contacts of lawyer with judge, INTG Judicial integrity, DMNR Demeanor, DILG Diligence, CFMG Case flow managing, DECI Prompt decisions, PREP Preparation for trial, FAMI Familiarity with law, ORAL Sound oral rulings, WRIT Sound written rulings, PHYS Physical ability, RTEN Worthy of retention.

```
> rprcomp(USJudgeRatings)
Standard deviations:
 [1] 3.22195231 1.03832823 0.51049711 0.41049221 0.22797980 0.16242562
 [7] 0.11155709 0.09407153 0.07441343 0.05595849 0.04492358 0.03805913

Rotation:
             PC1         PC2
CONT  0.09651014  0.90089601
INTG -0.29727192 -0.19029004
DMNR -0.28269055 -0.21697647
DILG -0.30634676  0.01963176
CFMG -0.29804314  0.19297945
DECI -0.30227359  0.18417871
PREP -0.30428044  0.10879296
FAMI -0.30144067  0.11286037
ORAL -0.30874784  0.05751148
WRIT -0.30769444  0.06085970
PHYS -0.28368257 -0.03718180
RTEN -0.30728474 -0.02411832
```

a) Interpret the first principal component.

b) Interpret the second principal component.

6.8. From the SAS output shown below, what is the variance explained by the second principal component?

```
                 Eigenvalues of the Covariance Matrix
         Eigenvalue    Difference    Proportion    Cumulative
    1    154.310607    145.147647        0.9439        0.9439
    2      9.162960                      0.0561        1.0000
                            Eigenvectors
                                 Prin1         Prin2
                July          0.343532      0.939141
                January       0.939141      -.343532
```

R/Splus Problems

Warning: Use the command *source(“G:/mpack.txt”)* **to download the programs. See Preface or Section 15.2.** Typing the name of the `mpack` function, eg *ddplot*, will display the code for the function. Use the `args` command, eg *args(pcasim)*, to display the needed arguments for the function.

6.9. a) Type the *R* command `pcasim()` and paste the output into *Word*.

This command computes the first 3 eigenvalues and eigenvectors for the classical and robust PCA using the $\boldsymbol{R}$ and $\boldsymbol{R}_U$. The multivariate normal data is such that the cases cluster tightly about the eigenvector $c(1, 1, ..., 1)^T$ corresponding to the largest eigenvalue. The term mncor gives the mean correlation between the classical and robust eigenvalues while the terms vexpl and rvexpl give the average variance explained by the largest 3 eigenvalues. The terms abscoreigvi give the absolute correlation between the i classical and robust eigenvector for $i = 1, ..., 3$ while the term abscorpc gives the absolute correlations of the first 3 principal components.

b) Are the robust and classical eigenvalues highly correlated? Is the absolute correlation for first classical principal component and the robust principal component high?

6.10. The Venables and Ripley (2003) CPU data has variables syct = cycle time,
mmin = minimum main memory,
chmin = minimum number of channels,
chmax = maximum number of channels,
perf = published performance, and
estperf = estimated performance.

a) There are nonlinear relationships among the variables and 1 is added to each variable to make them positive. Read more about the data set and make a scatterplot matrix with the *R commands* for this part. You can make the help window small by clicking the box with the − in the upper right corner. Include the scatterplot matrix in *Word*.

b) The log rule suggests using the log transformation on all of the variables. Make the log transformations, scatterplot matrix and DD plot with the *R commands* for this part. Right click "Stop" to go from the DD plot to the *R* prompt. Wait until part d) until you put plots in *Word*.

c) You might be able to get a better scatterplot matrix and DD plot by doing alternative transformations on the last two variables. The commands for this part give the log transformation for the first 4 variables and possible transformations for the last variables. Clearly state which transformations you use for the 5th and 6th variable. For example if you decide logs are ok, write down the following transformations.

```
zz[,5] <- log(z[,5])
zz[,6] <- log(z[,6])
```

d) For your data set zz of transformed variables, make the scatterplot matrix and DD plot and put the two plots in *Word*.

e) Put the classical PCA output using the correlation matrix into *Word* with the command for this problem.

f) Put the robust PCA output using the correlation matrix into *Word* with the command for this problem.

g) Comment on the similarities or differences of the classical and robust PCA.

6.11. The *R* data set USArrests contains statistics, in arrests per 100,000 residents, for assault, murder, and rape in each of the 50 US states in 1973. The fourth variable, UrbanPop, is the percent urban population in each state. For PCA, the *R* `summary` command can be used to get proportion of variance explained and cumulative proportion of variance explained, similar to *SAS* output.

a) Use the *R commands* for this part to get the classical and robust PCA summaries where $\boldsymbol{S}$ or $\boldsymbol{S}_U$ is used. Paste the summaries into *Word*.

i) Are the summaries similar?

ii) Using the 0.9 threshold, how many principal components are needed?

a) Use the *R commands* for this part to get the classical and robust PCA summaries where $\boldsymbol{R}$ or $\boldsymbol{R}_U$ is used. Paste the summaries into *Word*.

i) Are the summaries similar?

ii) using the 0.9 threshold, how many principal components are needed?

6.12. For PCA, a *biplot* is a plot of the first principal component versus the second principal component. The plotted points are $\hat{\boldsymbol{e}}_j^T \boldsymbol{x}_i$ for $j = 1, 2$ where the classical biplot uses $i = 1, ..., n$ and the robust plot uses cases in the RMVN set U. Let $\hat{\boldsymbol{e}}_j = (\hat{e}_{1j}, \hat{e}_{2j}, ..., \hat{e}_{pj})^T$. Then $\hat{e}_{kj}$ is called the *loading* of the kth variable on the jth principal component. An arrow with the kth variable name is the vector from the origin $(0, 0)^T$ to the loadings $(\hat{e}_{k1}, \hat{e}_{k2})^T$. So if the arrow is in the first quadrant, both loadings are positive, etc. If the arrow is long to the right but short down, then the loading with the first principal component is large and positive while the loading with the second principal component is small and negative.

The Buxton (1920) data has a cluster of 5 massive outliers. The first classical principal component tends to go right through a cluster of large outliers.

a) These *R* commands make the classical scree plot and biplot. Paste the plots into *Word*.

b) These *R* commands make the robust scree plot and biplot. Paste the plots into *Word*.

c) From the classical scree plot, how many principal components are needed? From the robust scree plot, how many principal components are needed?

d) The four variables used were *len, nasal, bigonal*, and *cephalic* . From the classical biplot, which variable had the 5 massive outliers.

e) From the robust biplot, which two variables loaded highest with the first principal component?