Chapter 2

Robustness

2.1 Robust statistics

All statistical methods rely on a number of assumptions. Typically two types of assumptions occur. Distributional assumptions assume that the observations are generated from a specific distribution, often the Gaussian distribution. Classical linear least squares regression for instance assumes that the errors are normally distributed with constant variance. Secondly, even when such an explicit distribution is not required, classical statistical methods rely on the assumption that all observations are generated from the same underlying process. However, in many real life applications it is observed that data sets contain several atypical observations called outliers. These are observations that deviate from the main pattern in the data.

Classical statistical methodology often turns out to be very sensitive to outliers. A few deviating observations can have a huge impact on the results. Robust statistics constructs estimators that control the effects of outliers. Typically robust methods give similar results as classical estimates when there are no outliers, but keep giving appropriate results when outliers are present.

Three R packages with robust methods that will be frequently used, are MASS [48], robustbase [41] and rrcov [47].

2.2 A univariate example

Consider the data set containing the logarithm of the annual incomes of 10 persons. If the data are located at the directory `C:\annualincome.txt`, the following lines import the data into R and provide some basic graphical displays (histogram, boxplot and normal qq-plot).

```
> data <- read.table("c:\\annualincome.txt",col.names="logIncome")
> anninc <- data$logIncome
> par(mfrow=c(1,3))
> boxplot(anninc)
> hist(anninc)
> qqnorm(anninc)
```

From this graphical analysis, it is clear that the log income of person 10 is quite large compared to the other log incomes. Now let us analyze the sample mean and the sample median.

```
> mean(anninc)
> median(anninc)
```

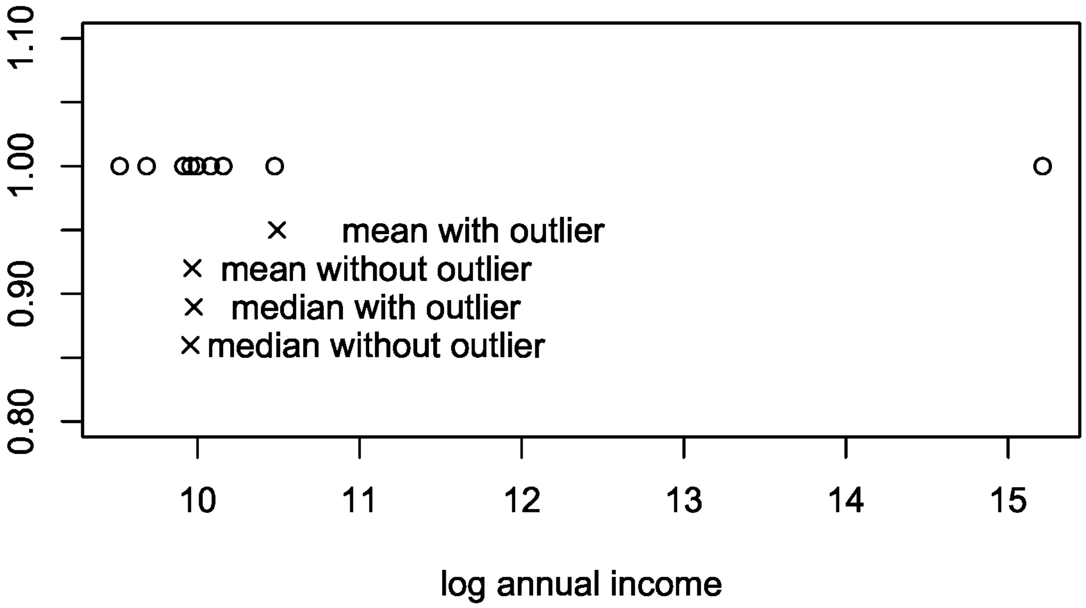
Observe that there is a large difference between both estimates. Now consider the data set without observation 10, the outlier.

```
> anninc2 <- anninc[-10]
> mean(anninc2)
> median(anninc2)
```

A simple visualization can be obtained by the following commands:

```
> plot(anninc,rep(1,10),ylab="",xlab="log annual income")
> points(mean(anninc),0.95,pch=4)
> points(mean(anninc2),0.92,pch=4)
> points(median(anninc),0.89,pch=4)
> points(median(anninc2),0.86,pch=4)
> text(11.5,0.95,"mean with outlier",cex=0.8)
> text(10.9,0.92,"mean without outlier",cex=0.8)
> text(10.9,0.89,"median with outlier",cex=0.8)
> text(10.9,0.86,"median without outlier",cex=0.8)
```

which results in the following figure:



Note that both the mean and the median of the sample without the outlier give approximately the same result as the median of the sample containing the outlier, but a very different result from the mean of the sample containing the outlier. This illustrates that the sample mean is not a robust estimator: a single observation can have an arbitrary large effect on the result. For the median the influence of a single observation is bounded. The sample median is an example of a robust estimator of location.

For univariate scale estimators the situation is similar. Consider the sample standard deviation s:

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

Again an outlier can be arbitrary influential: notice again the difference between using the data set with or without the outlier:

```
> sqrt(var(anninc))
> sqrt(var(anninc2))
```

For a robust scale estimator, e.g. the Median Absolute Deviation (MAD),

$$\text{MADN} = 1.4826\ \text{med}_i(|x_i - \text{med}_j(x_j)|),$$

the effect of individual observations is again much smaller:

```
> mad(anninc)
> mad(anninc2)
```

Exercise: retake the previous analysis replacing observation 10 by arbitrary chosen values. Compute sample mean, sample median, sample standard deviation and MADN. Convince yourself that the median and the mad are indeed much more robust.

At this point one might think that robust estimation could also be obtained by detecting outliers and deleting them. However, it is important to realize that detection of outliers is far from trivial. In the univariate case, one might suggest to flag points as outliers when the distance to the mean is larger than 3 standard deviations. This is based on the fact that for $X \sim N(0,1)$, $P(|X| > 3) = 0.003$. Let us apply this rule to the income data.

```
> (anninc[10]-mean(anninc))/sqrt(var(anninc))
```

Observe that the result is actually smaller than 3! This is because the sample mean is shifted and the sample variance is blown up due to the outlier. This is called the *masking effect*: outliers can change non-robust estimates so dramatically, that outlier detection using these estimates fails. However, when using the robust estimates in the outlier detection rule, we do find that observation 10 is very far away from the other observations.

```
> (anninc[10]-median(anninc))/mad(anninc)
```

This illustrates the importance of using a robust method in order to detect outliers.

2.3 A multivariate example

Given n data vectors $\mathbf{x}_i \in \mathbb{R}^p$. The classical estimators of location and scatter are the sample mean and the sample covariance matrix.

$$\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i, \qquad \mathrm{C} = \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})'.$$

Consider for example the **phosphor** data set in the robustbase package. This bivariate chemometrical data set contains values for inorganic and organic soil phosphorus at 18 samples.

```
> data(phosphor)
> phosphor.x <- phosphor[, 1:2]
> colMeans(phosphor.x)
> cov(phosphor.x)
```

The Minimum Covariance Determinant (MCD) estimator is a robust alternative

```
> cmcd <- covMcd(phosphor.x,alpha=0.5)
> cmcd
```

Large differences are observed. To visualize these quantities it is interesting to plot a so-called tolerance ellipse. This is based on the Mahalanobis distances

$$\mathrm{MD}(\mathbf{x}) = \sqrt{(\mathbf{x}-\bar{\mathbf{x}})'C^{-1}(\mathbf{x}-\bar{\mathbf{x}})}.$$

The points $\mathbf{x}$ for which the Mahalanobis distance $\mathrm{MD}(\mathbf{x})$ is constant are lying on an ellipsoid. If the data are multivariate normally distributed, then the squared Mahalanobis distances are asymptotically χ^2_p distributed. A $(1-\alpha)$ tolerance ellipse contains the points for which $\mathrm{MD}(\mathbf{x}) = \sqrt{\chi^2_{p,1-\alpha}}$ with $\chi^2_{p,1-\alpha}$ the $(1-\alpha)$ quantile of a chi-squared distribution with p degrees of freedom. Let us plot such a tolerance ellipse using both the classical estimates and the robust estimates.

```
> plot(cmcd,which ="tolEllipse",classic=TRUE)
```

It is now clear that two outliers inflate the classical covariance matrix. Also note that these outliers would not be detected by using a classical tolerance ellipse. Only using the robust MCD method they fall outside the tolerance ellipse.

2.4 Linear Regression

Consider the linear model with intercept

$$y_i = \beta_1 x_{1i} + \ldots + \beta_p x_{pi} + \epsilon_i \quad i = 1, \ldots, n$$

The classical approach to estimate the parameter vector $\boldsymbol{\beta}$ is the Least Squares method minimizing the sum of the squared residuals.

$$\min_{\boldsymbol{\beta}\in\mathbb{R}^p} \sum_{i=1}^{n} (y_i - \mathbf{x}_i'\boldsymbol{\beta})^2.$$

Consider the **telef** data with the number of phone calls in Belgium for each year between 1950 and 1973. If we want to model the number of phone calls as a function of the year, we can fit a simple linear regression line.

$$y_i = \beta_1 + \beta_2 x_{2i} + \epsilon_i \quad i = 1, \ldots, n$$

The least squares method provides estimates for the intercept β_1 and the slope β_2. A plot of the data together with the LS line is obtained as follows:

```
> data(telef)
> tel.lm <- lm(Calls~Year,data=telef)
> plot(telef$Year,telef$Calls,xlab="Year",ylab="Calls")
> abline(tel.lm)
```

Looking at the data one observes that something strange is going on in the years 1964-1969 when it appears as if an unusual large amount of phone calls was made. These 6 outlying years have a very big influence on the resulting LS fit. This can be seen by taking out these 6 outliers and retaking the analysis on this reduced data set.

```
> telreduced.lm <- lm(Calls[-(15:20)]~Year[-(15:20)],data=telef)
> abline(telreduced.lm)
```

Robust regression methods yield fits that are not so much influenced by a minority of observations, e.g. the Least Trimmed Squares (LTS) method (see Chapter 5 for details):

```
> tel.lts <- ltsReg(Calls~Year,data=telef)
> abline(tel.lts,lty=2)
```

Observe that LTS automatically manages to fit the majority of the data, even when a minority of outliers ruins the pattern.

From this simple regression example in two dimensions it might still seem easy to spot outliers without specific methodology. But in higher dimensions this can become very tricky.

Consider the **Toxicity** data set. It contains 38 samples of carboxylic acids for which the aquatic toxicity was measured together with 9 molecular descriptors. The goal is to predict the logarithm of toxicity.

```
> tox.lm <- lm(toxicity~.,data=toxicity)
> summary(tox.lm)
> plot(tox.lm, id.n=10)
```

Analyzing the results of a LS regression fit we see that several residuals are deviating on the normal qq-plot (cases 28, 34, 38, 13, 37, 36, 18, 32), but only observation 28 has a standardized residual larger than 2.5. The model fits relatively well ($R^2 = 0.84$). The qq-plot is not that bad. Now compare this to a robust fit.

```
> tox.lts <- ltsReg(toxicity~.,data=toxicity)
> plot(tox.lts)
```

Only by performing robust regression, it becomes clear that a minority of observations strongly deviates from the majority of data points.

2.5 Principal Component Analysis

Principal Component Analysis (PCA) is a technique for dimensionality reduction. The first principal component corresponds to the projection of the data onto the direction for which the variance of the projections is maximized. To find the second principal component, the same principle is applied in the $p - 1$ dimensional subspace orthogonal to the first principal direction. For the third principal component, the variance is maximized in the $p-2$ dimensional subspace orthogonal to the space spanned by the first two principal directions, etc.

Consider for instance the **Animals2** data set containing average brain and body weights for 65 species of animals. We analyze the log transformed data. Suppose that we want to reduce the dimension from 2 to 1. Then classical PCA searches the direction maximizing the variance of the projections.

```
> plot(log(Animals2),xlab="logBodyWt",ylab="logBrainWt")
> ani.pca <- PcaClassic(log(Animals2), k = 1)
> c <- getCenter(ani.pca)
> L <- getLoadings(ani.pca)
> slope <- L[2]/L[1]
> int <- c[2]-slope*c[1]
> abline(c(int,slope))
```

Again we see that there are three outliers in the data: three dinosaurs with a large body weight but a relatively small brain weight. The first principal direction is influenced quite a lot by these three dinosaurs. Let us apply a robust method as well.

```
> ani.robpca <- PcaHubert(log(Animals2), k = 1)
> cr <- getCenter(ani.robpca)
> Lr <- getLoadings(ani.robpca)
> sloper <- Lr[2]/Lr[1]
> intr <- cr[2]-sloper*cr[1]
> abline(c(intr,sloper),lty=2)
```

Clearly the robust PCA method is less sensitive to the outliers.

In high dimensions outliers can be even more dangerous, hiding themselves when one tries to detect them using classical PCA. Consider for instance the **Octane** data.

This data set contains NIR absorbance spectra for 39 gasoline samples. Each spectrum is measured at 226 wavelengths. The first variable contains the octane number. Such a high dimensionality is often difficult to work with. Applying a dimension reduction by PCA can provide a way out. Assume that we want to reduce the dimension to 3 by projecting onto the space spanned by the first 3 principal directions.

```
> octane <- read.table("c:\\octane.txt",sep=",")
> oct.pca <- PcaClassic(octane[,-1],k=3)
```

A simple idea to detect outliers would be to check the orthogonal distances between points and their projections (like one checks residuals in regression).

```
> plot(oct.pca)
```

The orthogonal distances are visible on the Y-axis (we ignore the X-axis for now). Observations 25 and 26 seem a bit outlying, but apart from them everything looks ok. Now let us compare with a robust PCA method.

```
> oct.robpca <- PcaHubert(octane[,-1],k=3)
> plot(oct.robpca)
```

One observes a huge difference with respect to the classical fit. In fact there are 6 extreme outliers in the data. Actually, in this case it turned out that these 6 samples were contaminated with alcohol making them indeed very different from the rest.