TMA4267 - Linear Statistical Models
Solutions to Exercise 7 - V2014

May 17, 2014

Last changes: Problem 3a, problem text (E3) was changed to $\lambda = 40$ instead of $\lambda = 0.4$ to give the anticipated result.

Problem 1: Gasoline data

a) Studying the scatter plots we may expect that there can be a linear relationship between all of the predictors and the response. However, we see that all the predictores are highly correlated - which points to the problem of multicollinearity. Multicollinearity will give large values off the diagonal in $(\boldsymbol{X}^T\boldsymbol{X})$, which again will lead to large variances of the estimated regression coefficients.

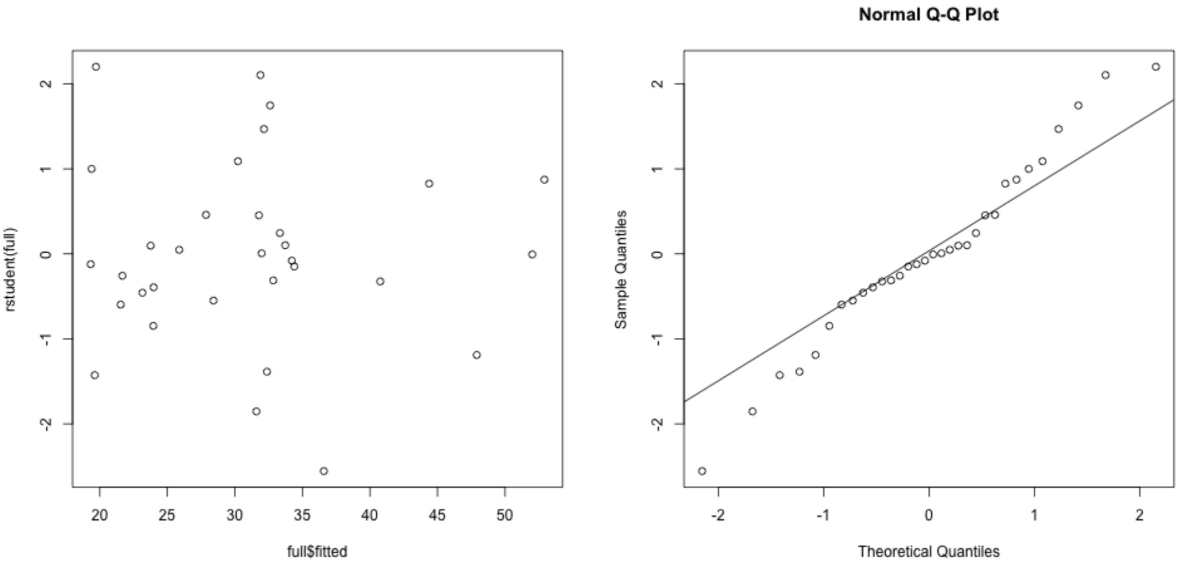
```
> cov(x)
          TankTemp   GasTemp  TankPres   GasPres
TankTemp 1.0000000 0.7742909 0.9554116 0.9337690
GasTemp  0.7742909 1.0000000 0.7815286 0.8374639
TankPres 0.9554116 0.7815286 1.0000000 0.9850748
GasPres  0.9337690 0.8374639 0.9850748 1.0000000
```

Remark: observe that the covariance matrix of the scaled variables equals the correlation matrix.

b) The full MLR in the original data gives a model where GasTemp and GasPres are the only two significant covariates. This model explains 92.6% of the variability in the data. Confidence intervals are very wide, expecially for GasPres. Residual plots look ok.

```
> summary(full)
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  31.1250     0.4826  64.494  < 2e-16 ***
xTankTemp    -0.5582     1.7677  -0.316  0.75461
xGasTemp      3.3953     1.0654   3.187  0.00362 **
xTankPres    -6.2737     4.1403  -1.515  0.14132
xGasPres     12.4904     3.8587   3.237  0.00319 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

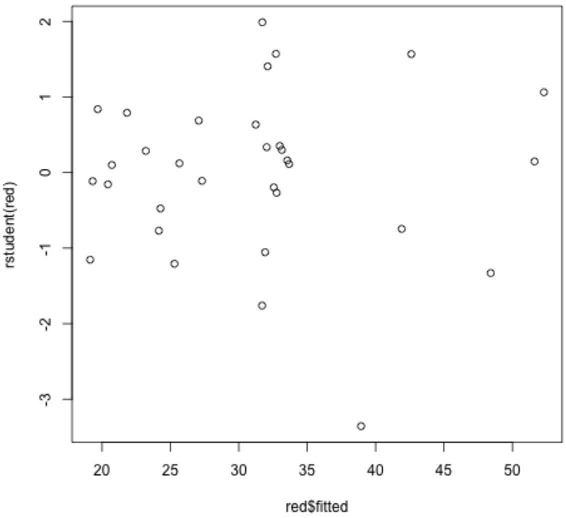
```
Residual standard error: 2.73 on 27 degrees of freedom
Multiple R-squared:  0.9261,Adjusted R-squared:  0.9151
F-statistic: 84.54 on 4 and 27 DF,  p-value: 7.249e-15

> confint(full)
                 2.5 %    97.5 %
(Intercept)  30.134788 32.115212
xTankTemp    -4.185204  3.068844
xGasTemp      1.209363  5.581255
xTankPres   -14.768913  2.221418
xGasPres      4.573047 20.407838
```

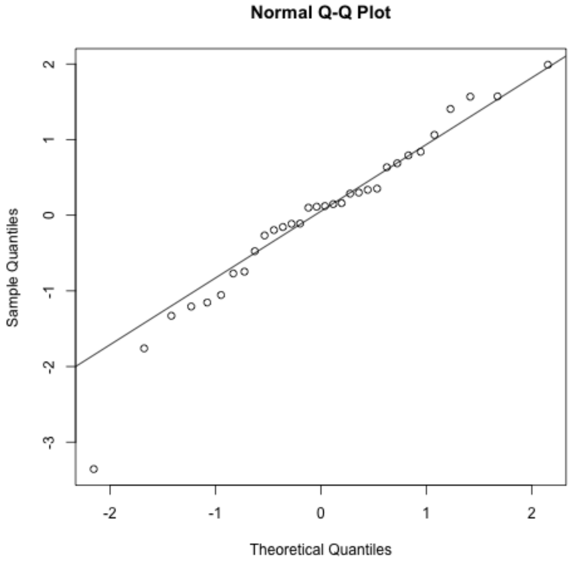


c) A possible reduced model could contain GasTemp and GasPres. Confidence intervals are now a bit narrower. An F-test with $H_0 : \beta_1 = \beta_3 = 0$ vs. at least one different from zero, fails to reject H_0 with a p-value of 0.1. Thus, this seems to be a sensible model.

```
> summary(red)
Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)         31.1250     0.5068  61.410  < 2e-16 ***
x[, c(2, 4)]GasTemp   4.3222     0.9423   4.587 7.98e-05 ***
x[, c(2, 4)]GasPres   5.0129     0.9423   5.320 1.04e-05 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

Residual standard error: 2.867 on 29 degrees of freedom
Multiple R-squared:  0.9124,Adjusted R-squared:  0.9064
F-statistic:   151 on 2 and 29 DF,  p-value: 4.633e-16

> confint(red)
                        2.5 %    97.5 %
(Intercept)         30.088406 32.161594
```

```
x[, c(2, 4)]GasTemp  2.394998  6.249326
x[, c(2, 4)]GasPres  3.085751  6.940079

> anova(full,red)
Analysis of Variance Table

Model 1: y ~ x
Model 2: y ~ x[, c(2, 4)]
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1     27 201.23
2     29 238.39 -2   -37.159 2.4929 0.1015
```





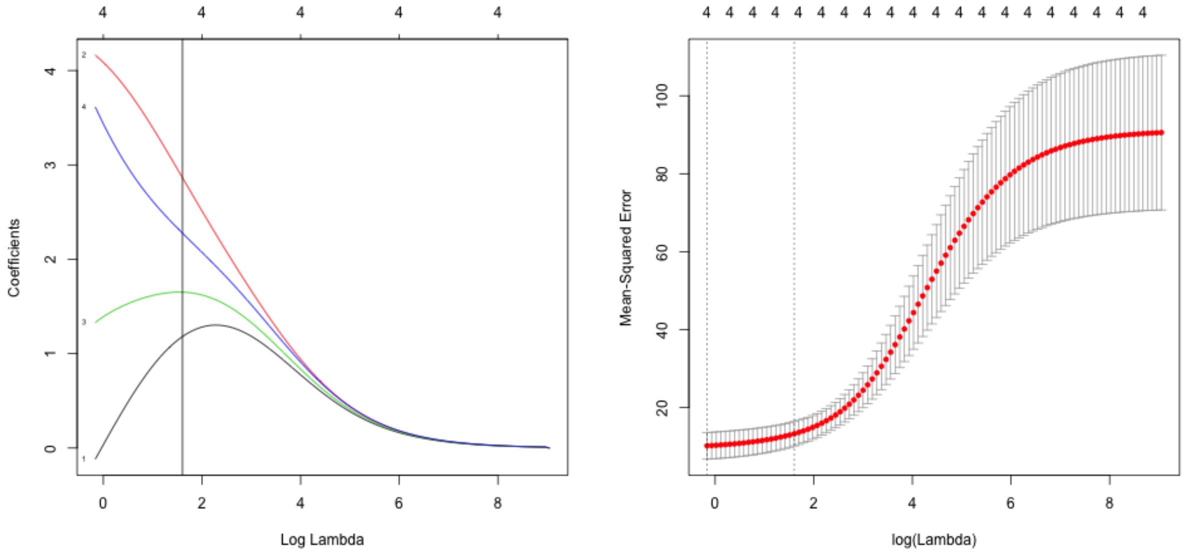
c) Best subset selection using Mallows C_p and R^2_{adj} both leads to a model with 3 predictors: GasTemp, TankPres and GasPres.

```
> bests <- regsubsets(x,y)
> sumbests <- summary(bests)
Subset selection object
4 Variables  (and intercept)
Selection Algorithm: exhaustive
         TankTemp GasTemp TankPres GasPres
1  ( 1 ) " "      " "     " "      "*"
2  ( 1 ) " "      "*"     " "      "*"
3  ( 1 ) " "      "*"     "*"      "*"
4  ( 1 ) "*"      "*"     "*"      "*"
> which.max(sumbests$adjr2)
[1] 3
> which.min(sumbests$cp)
[1] 3
```

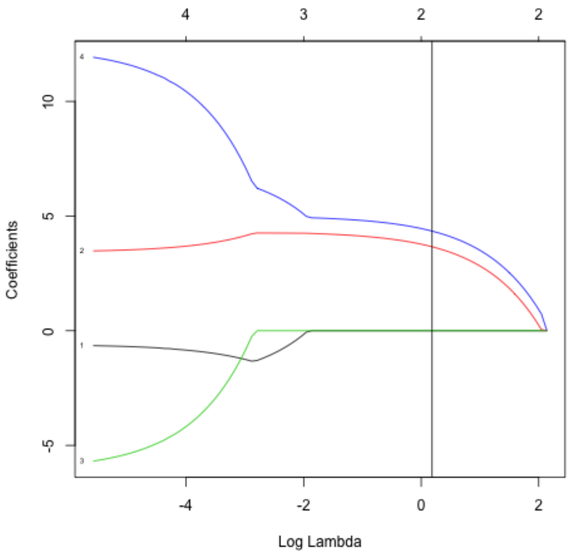
d) Cross-validation selects a small value for the penalty parameter for the ridge regression. The resulting ridge model has a substantially smaller estimated coefficient for the GasPres covariate, than the full and reduced MLR in b) and c).

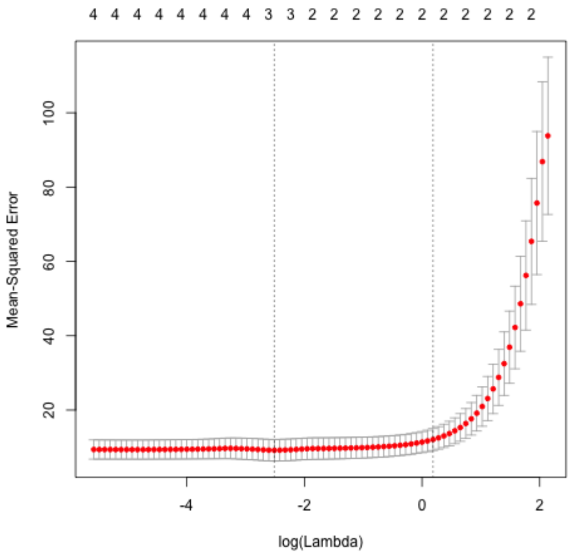
```
> fit.ridge=glmnet(x,y,alpha=0)
> cv.ridge=cv.glmnet(x,y,alpha=0)
> cv.ridge$lambda.min
[1] 0.8496615
> cv.ridge$lambda.1se
[1] 4.976485
> coef(cv.ridge)
5 x 1 sparse Matrix of class "dgCMatrix"
                    1
(Intercept) 31.125000
TankTemp     1.181294
GasTemp      2.863238
TankPres     1.651347
GasPres      2.276497
> full$coeff
(Intercept)   xTankTemp    xGasTemp   xTankPres    xGasPres
 31.1250000  -0.5581796   3.3953090  -6.2737478  12.4904423
```

For the lasso, cross-validation leads to a penalty parameter giving two non-zero coefficients in the regression model: GasTemp and GasPres. The GasTemp coefficient is a bit larger than for the full model, but the GasPres is much smaller (4.2 lasso as compared to 12.5 full).



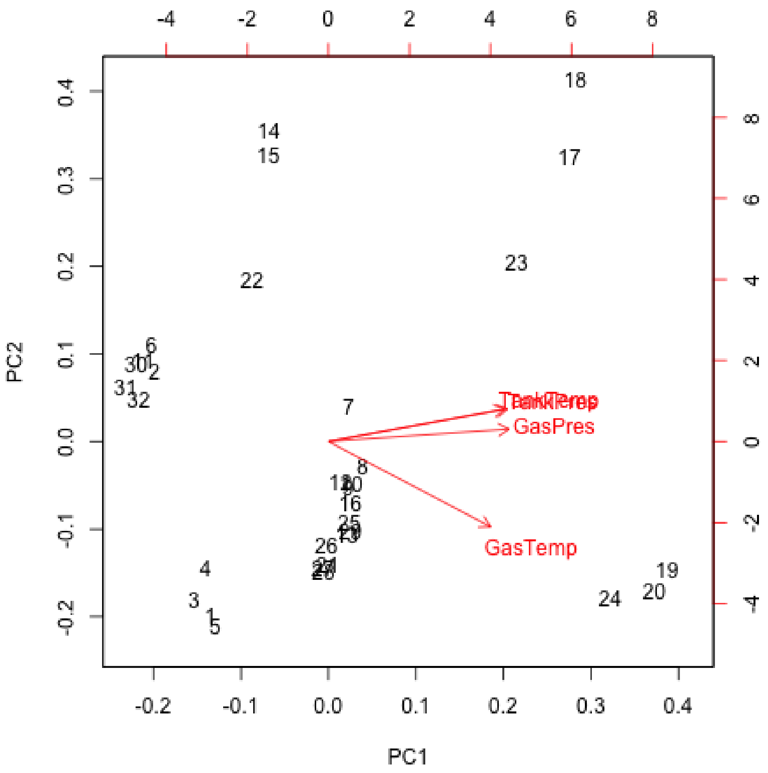
```
> cv.lasso=cv.glmnet(x,y)
> which.min(cv.lasso$cvm)
[1] 50
> coef(cv.lasso)
5 x 1 sparse Matrix of class "dgCMatrix"
                    1
(Intercept) 31.125000
TankTemp     .
GasTemp      3.524047
TankPres     .
GasPres      4.207423
```





e) The first principal component explains 91% of the total variance in the data. The scree plot suggests 2 PCs, and the λ-rule (variance larger than 1) suggests 1 PC. The first PC is almost an average of the four variables. The second PC is a contrast between GasTemp and the three others. We go with 2 PCs.

```
> res <- prcomp(x,scale=FALSE) #already scaled
> biplot(res)
> plot(res)# screeplot
> summary(res)
Importance of components:
                         PC1     PC2     PC3     PC4
Standard deviation     1.908 0.53285 0.26216 0.08684
Proportion of Variance 0.910 0.07098 0.01718 0.00189
Cumulative Proportion  0.910 0.98093 0.99811 1.00000
> res$rotation
               PC1        PC2         PC3        PC4
TankTemp 0.5045382  0.3335261 -0.77857671  0.1673913
GasTemp  0.4638348 -0.8725141 -0.09851185 -0.1177787
TankPres 0.5128498  0.3341220  0.31999566 -0.7231531
GasPres  0.5169948  0.1258646  0.53076973  0.6596650
```



f) Ridge and PCR both include all variables in the model. The coefficients for Ridge and PCR is more similar than from the full model.

```
# full, ridge and PCR
> cbind(full$coeff[-1],coef(cv.ridge)@x[-1],betas)
                [,1]     [,2]     [,3]
xTankTemp -0.5581796 1.181294 1.062248
xGasTemp   3.3953090 2.863238 5.307172
xTankPres -6.2737478 1.651347 1.097737
xGasPres  12.4904423 2.276497 1.881398
```

Lasso and Best subset regression only includes some variables, and the coefficient for GasPres is very different - because lasso has shruken it substantially.

```
                      [,1]     [,2]
                  0.000000 0.000000
x[, 2:4]GasTemp   3.289707 3.660973
x[, 2:4]TankPres -7.098875 0.000000
x[, 2:4]GasPres  12.870481 4.342993
```

Problem 2: Best subsets and stepwise methods

Answer exercise 1 in chapter 6.8 (page 259) of An Introduction to Statistical Learning with Applications in R.

a) Best subset

b) Either may have the smallest test set SSE.
c) i: TRUE, ii:TRUE, iii: FALSE, iv: FALSE, v: FALSE.

Problem 3: Ridge regression

This problem is taken, with permission from Wessel van Wieringen, from a course in High-dimensional data analysis at Vrije University, Amsterdam, The Netherlands.

a) First calculate $\mathbf{X}^T\mathbf{X}$ and $\mathbf{X}^T\mathbf{Y}$. These are given by:

$$\mathbf{X}^T\mathbf{X} = \begin{pmatrix} 8 & 0 \\ 0 & 16 \end{pmatrix} \qquad \mathbf{X}^T\mathbf{Y} = \begin{pmatrix} 320 \\ 35 \end{pmatrix}.$$

To penalize only the slope parameter add:

$$\mathbf{\Lambda} = \begin{pmatrix} 0 & 0 \\ 0 & \lambda \end{pmatrix}$$

to $\mathbf{X}^T\mathbf{X}$ in the normal equations. This leads to following ridge estimate:

$$\begin{aligned} \hat{\beta}_{ridge} &= (\mathbf{X}^T\mathbf{X} + \mathbf{\Lambda})^{-1}\mathbf{X}^T\mathbf{Y} \\ &= \begin{pmatrix} 8 & 0 \\ 0 & 16+\lambda \end{pmatrix}^{-1} \begin{pmatrix} 320 \\ 35 \end{pmatrix} \\ &= \begin{pmatrix} 1/8 & 0 \\ 0 & 1/(16+\lambda) \end{pmatrix} \begin{pmatrix} 320 \\ 35 \end{pmatrix} \\ &= \begin{pmatrix} 40 \\ 35/(16+\lambda) \end{pmatrix}. \end{aligned}$$

Choosing $\lambda = 40$ yields the reported estimates.

b) A projection matrix $\mathbf{Q}$ would satisfy $\mathbf{Q} = \mathbf{Q}^2$. Verify:

$$\begin{aligned} \mathbf{Q}^2 &= \mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^{\mathrm{T}} \\ &= \mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}(\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I}_{pp})(\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^{\mathrm{T}} \\ &\quad -\lambda\mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}(\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^{\mathrm{T}} \\ &= \mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^{\mathrm{T}} - \lambda\mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I}_{pp})^{-2}\mathbf{X}^{\mathrm{T}} \\ &= \mathbf{Q} - \lambda\mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I}_{pp})^{-2}\mathbf{X}^{\mathrm{T}} \\ &\neq \mathbf{Q}. \end{aligned}$$

Hence, $\mathbf{Q}$ is not a projection matrix.

c) The ridge fit is given by $\hat{\mathbf{Y}}(\lambda) = \mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{Y} = \mathbf{Q}\mathbf{Y}$ and the associated residuals by: $\hat{\varepsilon}(\lambda) = \mathbf{Y} - \mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I}_{pp})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{Y} = [\mathbf{I}_{pp} - \mathbf{Q}]\mathbf{Y}$. Would the residual and the fit be orthogonal, their inner product becomes zero: $\langle \hat{\mathbf{Y}}(\lambda), \hat{\varepsilon}(\lambda) \rangle = 0$. Verify:

$$\begin{aligned} \langle \hat{\mathbf{Y}}(\lambda), \hat{\varepsilon}(\lambda) \rangle &= [\hat{\mathbf{Y}}(\lambda)]^{\mathrm{T}}\hat{\varepsilon}(\lambda) \\ &= [\mathbf{Q}\mathbf{Y}]^{\mathrm{T}}[\mathbf{I}_{pp} - \mathbf{Q}]\mathbf{Y} \\ &= \mathbf{Y}^{\mathrm{T}}\mathbf{Q}^{\mathrm{T}}(\mathbf{I}_{pp} - \mathbf{Q})\mathbf{Y} \\ &= \mathbf{Y}^{\mathrm{T}}(\mathbf{Q}^{\mathrm{T}} - \mathbf{Q}^{\mathrm{T}}\mathbf{Q})\mathbf{Y} \\ &= \mathbf{Y}^{\mathrm{T}}(\mathbf{Q} - \mathbf{Q}^2)\mathbf{Y}, \end{aligned}$$

where we have used the symmetry of $\mathbf{Q}$. Invoke the result of b) to conclude.

Problem 4: The cork data and PCA

a) Here the measurement scale for the variables are the same, so using the original or the standardized version of the data will not do much change. We go with the standardized version.

b) The scree plot suggests to use 2 PCs. 2 PCs explain 96% of the total variance of the data. One PC explain nearly 90% of the data. The first PC gives nearly equal weight to all variables, and can be seen as a mean effect. The second PC contrasts the North+East with the South+West, with the highest weight on the East and West.

c) If the cork data can be seen as a sample from a four-dimensional normal distribution the first eigenvector is the principal axis of the equal probability ellipsoid.

```
# TMA4267, Exercise 7, Problem 4
# the cork data is taken from Multivariate Analysis by Mardia, Kent and Bibby.

# a
corkds <- as.matrix(read.table("http://www.math.ntnu.no/~mettela/TMA4267/Data/corkMKB.txt"))
dimnames(corkds)[[2]] <- c("N","E","S","W")

xbar <- apply(corkds,2,mean)
xbar
smat <- var(corkds)
smat

# the variances are similar, only small differences, and the measurement scale is the same fo

# plotting data, one colour for each variable
thisylim <- range(corkds)
plot(1:28,corkds[,1],ylim=thisylim,type="p",pch=20,col=2)
points(1:28,corkds[,2],pch=20,col=3)
points(1:28,corkds[,3],pch=20,col=4)
points(1:28,corkds[,4],pch=20,col=5)

pairs(corkds)


# PCA
# go for scaled version

x <- apply(corkds,2,scale)

#  without the use of built-in functions:
s <- cov(x)
ee <- eigen(s)

# proportion of total variance for each PC
print(ee$values/sum(ee$values))
```

```
# cumulative proportions
print(cumsum(ee$values)/sum(ee$values))

# loadings
rownames(ee$vectors) <- dimnames(corkds)[[2]]
colnames(ee$vectors) <- c(paste("PC",1:4))

print(ee$vectors)

# using built-in functions for PCA

cork.pca <- prcomp(x,scale=FALSE)
summary(cork.pca)

screeplot(cork.pca)
screeplot(cork.pca,type="lines")

loadings(cork.pca)

pairs(cork.pca$x)
# looks rather uncorrelated
```