GR5223 - Take home Final

Mathieu Sauterey - UNI: mjs2364

May 1, 2018

Problem 1

A firm is attempting to evaluate the quality of its sales staff and is trying to find an examination or series of tests that may reveal the potential for good performance in sales. The firm has selected a random sample of 50 sales people and has evaluated each on three measures of performance:

y_1 = growth of sales y_2 = profitability of sales y_3 = new-account sales

x_1 = creativity score x_2 = mechanical reasoning score x_3 = abstract reasoning score x_4 = mathematics score

The n = 50 observations on p = 7 variables are are in the data file Salespeople.csv on Courseworks. Assume an orthogonal factor model holds for the standardized variables

```
# Reads "Salespeople.csv" and creates its sample correlation matrix
sales <- read.csv("Salespeople.csv")
R_sales <- cor(sales)
```

(a) Obtain the principal component solution for k = 3 common factors.

i. List the estimated specific variances, that is, the diagonal entries of Psi_hat.

```
# First, we perform a spectral decomposition on R_sales to get its evalues and evectors
Gamma <- eigen(R_sales)$vectors
Lambda <- diag(eigen(R_sales)$values)

# Then we calculate Q_hat for k = 3 (one common factor)
Q_hat <- (Gamma %*% sqrt(Lambda) )[,1:3]
Q_hat
```

```
##            [,1]         [,2]         [,3]
## [1,] -0.6603168  0.645814259  0.318588471
## [2,] -0.7832895  0.284971246  0.004178606
## [3,] -0.6488301 -0.620656841  0.426047603
## [4,] -0.9141297 -0.193592498 -0.306273206
## [5,] -0.9730692 -0.107975602 -0.053267303
## [6,] -0.9428714  0.028297179 -0.312228320
## [7,] -0.9447504  0.008890823  0.144082982
```

```
# We calculate Psi_hat (uniquenesses)
Psi_hat <- diag(diag(R_sales - Q_hat %*% t(Q_hat)))
diag(Psi_hat)
```

```
## [1] 0.04540700 0.30523146 0.01228801 0.03308555 0.03864028 0.01270628
## [7] 0.08660774
```

The uniquenesses are $diag(\hat{\Psi}_{PC})$ = (0.0454, 0.305, 0.0123, 0.0331, 0.0386, 0.0127, 0.0866)’

ii. Obtain the (varimax) rotated loadings, and interpret the resulting factor solution.

```
# Calculates the varimax rotated PC loadings
Q_star <- varimax(Q_hat)$loadings
Q_star
```

```
##
## Loadings:
##      [,1]   [,2]   [,3]  
## [1,] -0.213  0.952       
## [2,] -0.552  0.607  0.145
## [3,] -0.287         0.950
## [4,] -0.909  0.181  0.328
## [5,] -0.779  0.387  0.452
## [6,] -0.908  0.356  0.189
## [7,] -0.616  0.548  0.483
##
##                 [,1]  [,2]  [,3]
## SS loadings    3.071 1.889 1.506
## Proportion Var 0.439 0.270 0.215
## Cumulative Var 0.439 0.709 0.924
```

The varimax rotated principal component loadings $\hat{Q}^*{}_{PC}$ are printed above.

The first rotated PC loading is roughly a straight average.

The second PC loading is a weighted average with largest weight on y_1 (growth of sales) and zero weight given to y_3 (new-account sales).

The third PC is somewhat opposite to the second PC: it is a weighted average with largest weighted assigned to y_3 (new-account sales) and zero weight assinged to y_1 (growth of sales).

(b) Obtain the maximum likelihood solution for k = 3 common factors.

i. List the estimated specific variances, that is, the diagonal entries of Psi_hat.

```
# The function below performs factor extraction using the MLE method for k=3
MLE_sales <- factanal(~x1+x2+x3+x4+y1+y2+y3, data=sales, factors=3, rotation="none")
MLE_sales$uniquenesses
```

```
##         x1         x2         x3         x4         y1         y2 
## 0.00500000 0.44662048 0.00500000 0.03750980 0.03857165 0.03448071 
##         y3 
## 0.08812176
```

The uniquenesses are $diag(\hat{\Psi}_{MLE}) = (0.005, 0.447, 0.005, 0.0375, 0.0386, 0.0345, 0.0881)$'

ii. Obtain the (varimax) rotated loadings, and interpret the resulting factor solution.

```
# Calculates the varimax rotated MLE loadings
Q_star_MLE <- varimax(MLE_sales$loadings)$loadings
Q_star_MLE
```

```
##
```

```
## Loadings:
##    Factor1 Factor2 Factor3
## x1  0.964   0.255         
## x2  0.465   0.542  -0.207 
## x3          0.299  -0.950 
## x4  0.180   0.917  -0.298 
## y1  0.374   0.793  -0.438 
## y2  0.317   0.911  -0.185 
## y3  0.544   0.651  -0.438 
##
##                 Factor1 Factor2 Factor3
## SS loadings       1.718   3.175   1.453
## Proportion Var    0.245   0.454   0.208
## Cumulative Var    0.245   0.699   0.906
```

The varimax rotated MLE loadings $\hat{Q}^*{}_{MLE}$ are printed above. Overall the MLE loadings are similar in interpretations and values to the PC loadings:

The first MLE loading is a weighted average with largest weighted assigned to y_1 (growth of sales) and zero weight assigned to y_3 (new-account sales).

The second rotated MLE loading is roughly a straight average.

The third MLE loading is somewhat opposite to the second PC: it is a weighted average with largest weight on y_3 (new-account sales) and zero weight given to y_1 (growth of sales).

(c) Suppose a new salesperson, selected at random, has achieved the performance measures (y1,y2,y3)=(110,98,105) and obtains test scores of (x1,x2,x3,x4) = (15,18,12,35). Using the rotated principal component solution for k = 3, calculate this salesperson's factor scores by the regression method.

```
# Loads and standardizes the salesperson's test scores
x0 <- c(15,18,12,35)
xbar <- apply(sales[,1:4], 2, mean)
s_x <- apply(sales[,1:4], 2, sd)
z0 <- (x0 - xbar)/s_x

# Loads and standardizes the salesperson's performance measures
y0 <- c(110,98,105)
ybar <- apply(sales[,5:7], 2, mean)
s_y <- apply(sales[,5:7], 2, sd)
w0 <- (y0 - ybar)/s_y

# calculates the salesperson's factor scores by the regression method
f.hat.reg <- t(Q_star) %*% solve(R_sales, c(z0,w0))
f.hat.reg
```

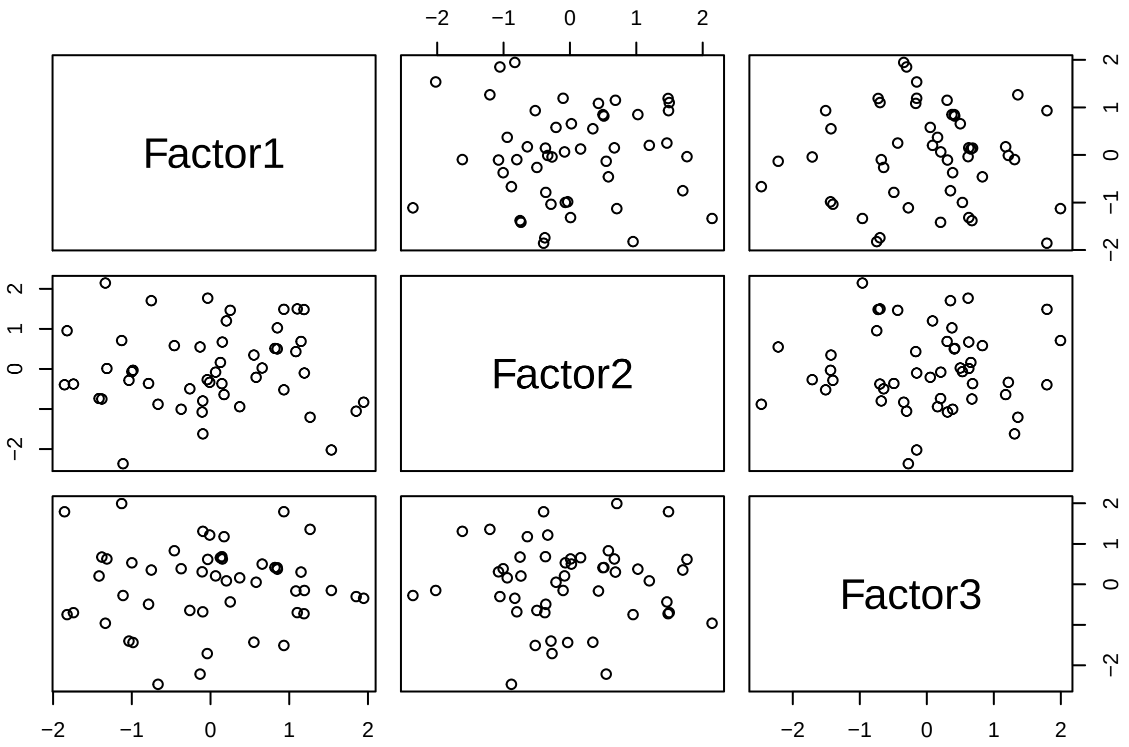
```
##           [,1]
## [1,] 0.4099016
## [2,] 1.1710833
## [3,] 0.9850043
```

The predicted factor scores by the regression method are $\hat{f}_0 = (0.409, 1.17, 0.985)$'

(d) Using the rotated maximum likelihood solution for k = 3, make a scatterplot matrix of the factor scores, calculated by the regression method, for the n = 50 observed cases.

```
# MLE factor scores (regression method) for the 50 cases
all_scores <- factanal(~x1+x2+x3+x4+y1+y2+y3, data=sales, factors=3,
                       rotation="varimax", scores = "regression")$scores

# scatterplot matrix of the 50 tri-dimensional MLE factor scores
pairs(all_scores)
```



Problem 2

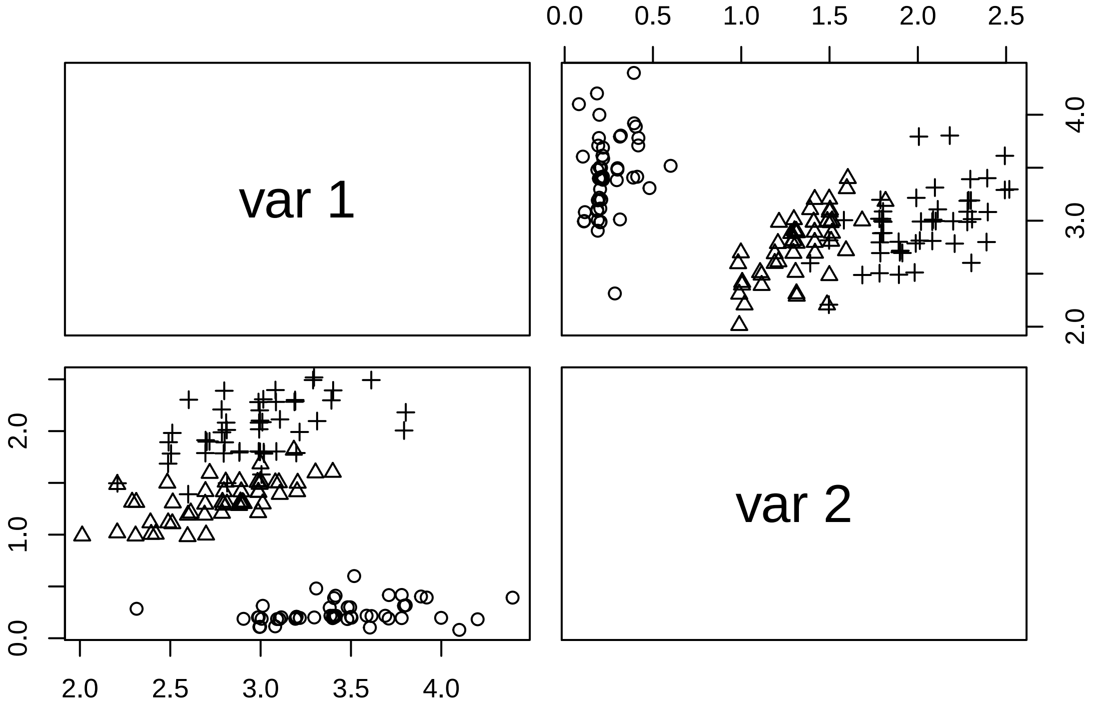
The data file Irises.csv contains observations of x1 = sepal width and x2 = petal width for samples from three species of iris: Π_1 is Iris setosa, Π_2 is Iris versicolor, and Π_3 is Iris virginica. There are n1 = n2 = n3 = 50 observations in each sample.

```
# Reads the data file and stores the dataset into a dataframe
irises <- read.csv("Irises.csv")
```

(a) Draw the scatterplot of (x1, x2) using different plotting symbols to indicate species. You may wish to jitter the values so identical observations are not obscured. Does the assumption of bivariate normal populations seem reasonable?.

```r
# jitters the values (slightly modifies them)
jit.iris <- t(rbind(jitter(irises[,1]),jitter(irises[,2])))

# Scatterplot of (x1, x2) using different plotting symbols to indicate species
pairs(jit.iris, pch = irises[,3])
```



In the plot above, the sample of Iris setosa is represented by circles, Iris versicolor by triangles and Iris virginica by crosses.

(b) (b) Assuming the samples are from bivariate normal populations with a common covariance matrix, test the hypotheses

$H_0 : \mu_2 = \mu_3$ versus $H_1 : \mu_2 \neq \mu_3$

Find the p-value and clearly state your conclusion. Is the assumption of a common covariance matrix reasonable in this problem?

```r
# Calculates basic parameters necessary to perform a test
n2 <- 50
n3 <- 50
p <- 2
```

```r
xbar2 <- colMeans(irises[51:100,-3])
xbar3 <- colMeans(irises[101:150,-3])

S2 <- cov(irises[51:100,-3])
S3 <- cov(irises[101:150,-3])

# Calculates the pooled variance
S <- 1/(n2 + n3 - 2) * ((n2-1)*S2 + (n3-1)*S3)

# Calculates the T^2 test statistic (follows a F distribution)
T2 <- 1/(1/n2 + 1/n3) * sum((xbar2 - xbar3) * solve(S, xbar2 - xbar3))
T2
```

```
## [1] 256.892
```

```r
# p-value of the test
1 - pf((n2+n3-p-1)/((n2+n3-2)*p)*T2, df1=p, df2=n2+n3-p-1)
```

```
## [1] 0
```

The p-value is zero, we conclude that the mean vectors of the populations of Iris versicolor and Iris virginica are not equal.

(c) Assuming the populations are bivariate normal with a common covariance matrix, predict the species of a new observation x = (x1,x2) = (3.50, 1.75), based on the linear classifier with pi__1 = pi__2 = pi__3. Estimate the posterior probabilities P(pi__j|x) for j = 1, 2, 3.

```r
irises$Species <- as.factor(irises$Species)

# We must create a dataframe with new data that we want to classify
x.new <- c(3.50, 1.75)
x.new <- data.frame(x1 <- x.new[1], x2 <- x.new[2])

# Prior probabilities of the LDA classifier
prior <- c(1/3, 1/3, 1/3)

library(MASS)

# Uses LDA method to classify the new data
lda.iris <- lda(Species ~ x1+x2, prior=prior, data=irises)
predict(lda.iris, prior=prior, newdata=x.new)$class
```

```
## [1] 2
## Levels: 1 2 3
```

```r
predict(lda.iris, prior=prior, newdata=x.new)$posterior
```

```
##              1         2         3
## 1 3.209389e-14 0.7187594 0.2812406
```

Here we chose a LDA classifier because we assume that the populations are bivariate normal with a common covariance matrix. Based on the LDA classifier we predict that this new observation belongs to population 2 (Iris versicolor). The posterior probabilities printed above support this conclusion as the population (Iris versicolor) is given the largest posterior probability.

(d) Evaluate the performance of the linear classifier with equal prior probabilities by computing the cross-validation estimate of the actual error rate.

```
# Preallocates space for the n=150 Holdout predictions
Holdout.lda <- irises$Species

# Holdout cross validation
for(ij in 1:length(Holdout.lda)){

 hold <- lda(Species ~ x1+x2, data=irises[-ij,], prior=prior)
 Holdout.lda[ij] <- predict(hold, prior=prior, newdata=irises[ij,])$class
}

# Confusion matrix
confusion <- xtabs(~ irises$Species + Holdout.lda)
confusion
```

```
##                Holdout.lda
## irises$Species  1  2  3
##               1 50  0  0
##               2  0 48  2
##               3  0  4 46
```

```
# Cross-validation estimate of the actual error rate
error.cv <- 1 - sum(diag(confusion)) / sum(confusion)
error.cv
```

```
## [1] 0.04
```

The Holdout cross-validation estimate of the actual error rate is 4%.

Problem 3

Annual financial data were collected for bankrupt firms approximately two years prior to their bankruptcy, and for financially sound firms at about the same time. The data on four variables

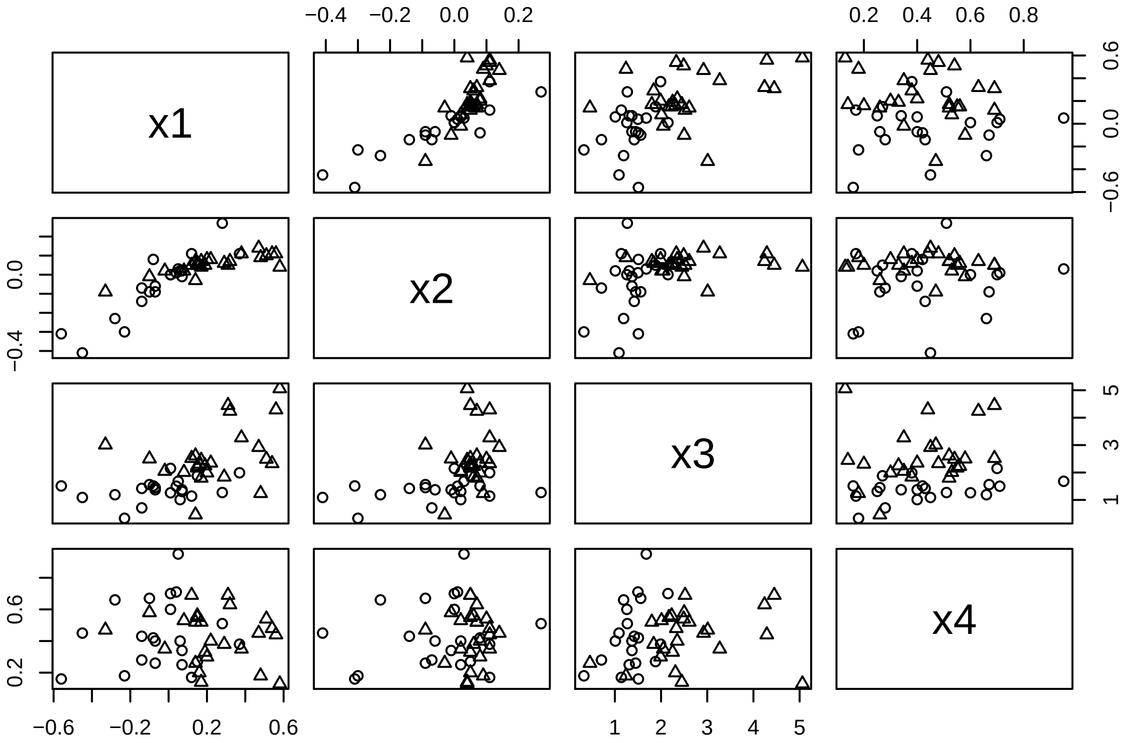
x_1 = ratio of cash flow to total debt x_2 = ratio of net income to total assets x_3 = ratio of current assets to current liabilities x_4 = ratio of current assets to net sales

are given in the file Bankruptcy.csv on Courseworks. The last column indicates group membership by Population=1 for Π_1 : bankrupt firms, and Population=2 for Π_2 : non-bankrupt firms. Using the $n1 = 21$ observations for bankrupt firms and the $n2 = 25$ observations for financially sound firms:

```
# Reads the data file and stores the dataset into a dataframe
bank <- read.csv("Bankruptcy.csv")
```

(a) Construct a scatterplot matrix of (x1, x2, x3, x4), using a different plotting symbol for bankrupt versus non-bankrupt. Which pair of variables seems to give the best separation between the two groups?

```
# Scatterplot matrix of (x1, x2, x3, x4)
pairs(bank[,-5], pch = bank$Population)
```



On the scatterplot above, bankrupt firms are represented by circles and non-bankrupt firms are represented by triangles.

(b) Assuming both random samples are from multivariate normal populations, estimate the posterior probability of bankruptcy for a firm with (x1, x2, x3, x4) = (-0.05,-0.05, 1.25, 0.3). Use the prior probabilities pi_1 = 0.05 pi_2 = 0.95.

```
bank$Population <- as.factor(bank$Population)

# We must create a dataframe with new data that we want to classify
x.new.bank <- c(-0.05,-0.05, 1.25, 0.3)
x.new.bank <- data.frame(x1 <- x.new.bank[1], x2 <- x.new.bank[2],
                         x3 <- x.new.bank[3], x4 <- x.new.bank[4])

# Prior probabilities of the QDA classifier
prior.b <- c(0.05, 0.95)

# Uses QDA method to classify the new data
library(MASS)
qda.bank <- qda(Population ~ x1+x2+x3+x4, prior=prior.b, data=bank)
predict(qda.bank, prior=prior.b, newdata=x.new.bank)$class
```

```
## [1] 1
## Levels: 1 2
```

```
predict(qda.bank, prior=prior.b, newdata=x.new.bank)$posterior
```

```
##           1         2
## 1 0.5089642 0.4910358
```

Here we chose a QDA classifier because we assume that the populations are multivariate normal but the scatterplot shows that they have different covariance matrices. Based on the QDA classifier we predict that this new observation belongs to population 1 (bankrupt firms). The posterior probabilities printed above support this conclusion as the population 1 (bankrupt firms) is given the largest posterior probability.

(c) Evaluate the classification rule used in part (b) by computing the apparent error rate, as well as the cross-validation estimate of the actual error rate.

```
## Apparent error Rate

confusion.b <- xtabs(~ bank$Population + predict(qda.bank, prior=prior.b)$class)
confusion.b
```

```
##                 predict(qda.bank, prior = prior.b)$class
## bank$Population  1  2
##               1 12  9
##               2  0 25
```

```
error.b <- 1 - sum(diag(confusion.b)) / sum(confusion.b)
error.b
```

```
## [1] 0.1956522
```

```
## Holdout CV Error estimate

Holdout.qda.b <- bank$Population

for(ij in 1:length(Holdout.qda.b)){

 hold <- qda(Population ~ x1+x2+x3+x4, data=bank[-ij,], prior=prior.b)
 Holdout.qda.b[ij] <- predict(hold, prior=prior.b, newdata=bank[ij,])$class
}

confusion.b.cv <- xtabs(~ bank$Population + Holdout.qda.b)
confusion.b.cv
```

```
##                 Holdout.qda.b
## bank$Population  1  2
##               1 10 11
##               2  1 24
```

```
error.b.cv <- 1 - sum(diag(confusion.b.cv)) / sum(confusion.b.cv)
error.b.cv
```

```
## [1] 0.2608696
```

The apparent error rate is 19.6% which is typically over-optimistic. This is confirmed by the the fact that the Holdout cross-validation error estimate is computed to be 26.1%.

Problem 4

Protein consumption in n = 25 European countries for p = 9 food groups is given in the data file Protein.csv on Courseworks.

```
# Reads the data file and stores the dataset into a dataframe
protein <- read.csv("Protein.csv")
```

(a) Obtain the sample correlation matrix and determine its eigenvalues and eigenvectors. Interpret the first two normalized principal components.

```
# Correlation matrix of the protein consumption per European coutries
R_prot <- cor(protein[,-1])

# Eigenvalues of the correlation matrix
L.R <- eigen(R_prot)$values
L.R
```

```
## [1] 3.9892112 1.6236516 1.1300087 0.9656216 0.4760342 0.3260847 0.2713040
## [8] 0.1160319 0.1020522
```

```
# Eigenvectors of the correlation matrix
G.R <- eigen(R_prot)$vectors[,1:2]
G.R
```

```
##             [,1]        [,2]
##  [1,] -0.3032262  0.05763629
##  [2,] -0.3118826  0.23464255
##  [3,] -0.4273740  0.03487312
##  [4,] -0.3792955  0.18282306
##  [5,] -0.1357873 -0.65004842
##  [6,]  0.4380061  0.23677045
##  [7,] -0.2891092 -0.35376960
##  [8,]  0.4214673 -0.14045554
##  [9,]  0.1129741 -0.53242525
```

The first NPC is a contrast between Fish, Nuts, Fruit/Vegetables and all the others.

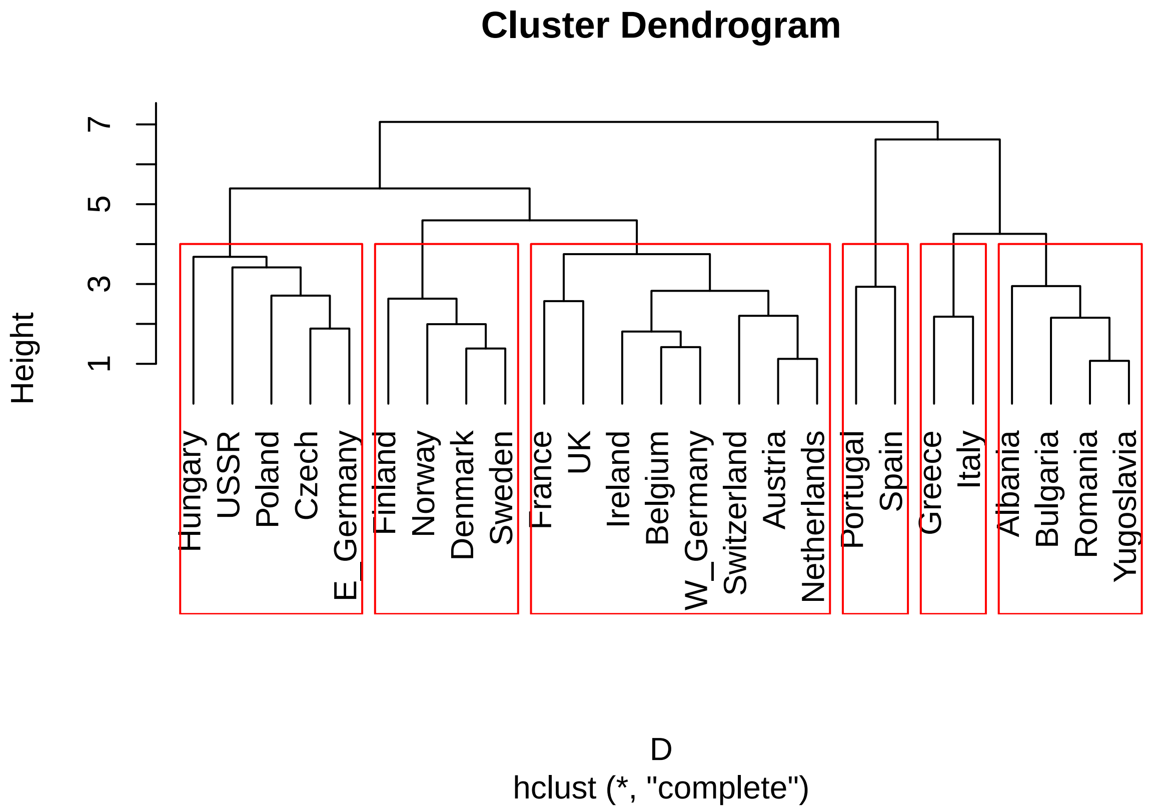
The second NPC is a contrast between Milk, Starch, Nuts, Fruit/Vegetables and all the others.

(b) Compute the distance matrix (Euclidean, or squared L2-norm) using the standardized variables. Produce a dendogram for the complete-linkage, agglomerative hierarchical clustering algorithm, indicating the solution for k = 6 clusters. Does this clustering make sense, given what you know about the countries?

```
# Calculates the Euclidean distance matrix
D <- dist(scale(protein[,-1]), method="euclidean")

# Dendogram with complete-Linkage
hc.complete <- hclust(D, method="complete")

plot(hc.complete, hang=-1, labels=protein$Country)
rect.hclust(hc.complete, k=6, border="red")
```



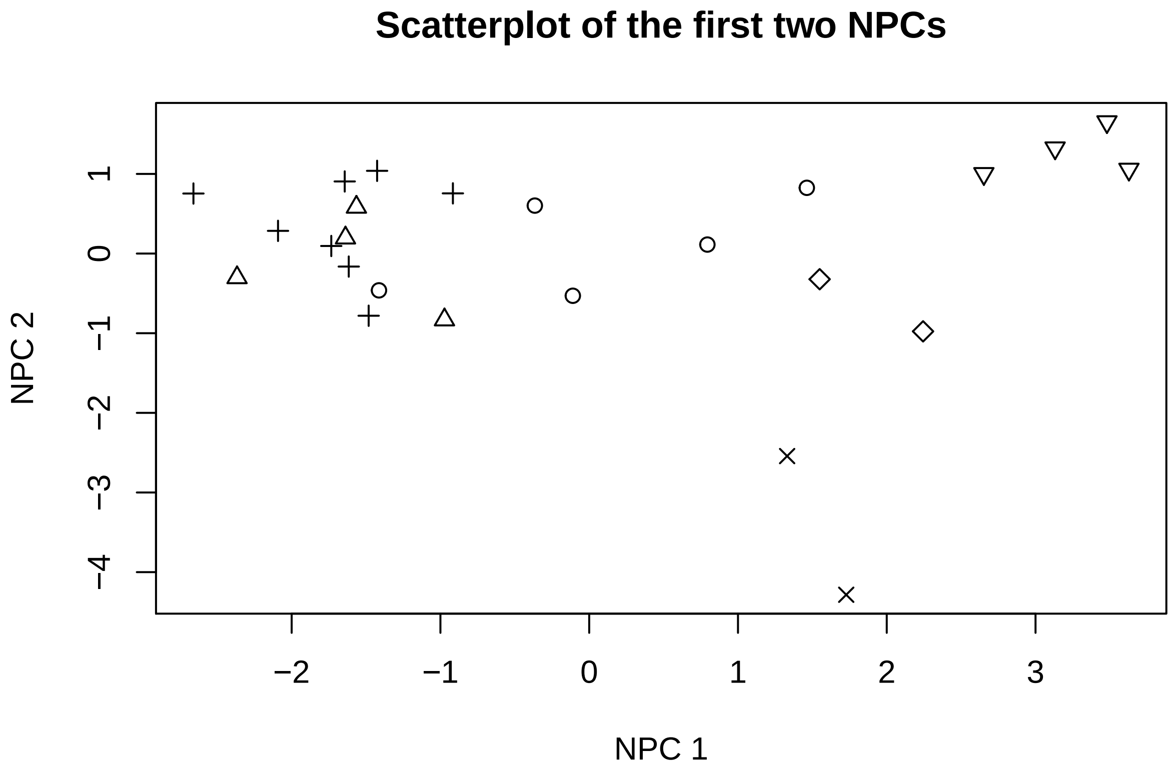
This dendogram makes sense as each cluster corresponds to a group of countries that are close geographically and similar culturally.

(c) Make a scatterplot of the first two principal components, using different plotting symbols to indicate cluster membership. Briefly explain what this analysis tells us about the differences in diet across different regions of Europe.

```
# Obtains the score of the first two NPCs
Z <- scale(protein[,-1]) %*% G.R

# Label each countries with a cluster
Cluster <- c(6,3,3,6,1,2,1,2,3,5,1,3,5,3,2,1,4,6,4,2,3,3,1,3,6)

# Scatterplot of first 2 NPCs with labeled clusters
plot(Z[,1], Z[,2], xlab = "NPC 1", ylab ="NPC 2",
     main = "Scatterplot of the first two NPCs", pch = Cluster)
```



This analysis tells us that consumption of Fish, Nuts, Fruit/Vegetables, Milk and Starch, are the main predictors that determine the different eating habits among European countries.