Problem Set 2

Luca Frattegiani (1013326)

17/5/2022

Exercise 1: Pulp Paper Data

Data Description

The dataset "pulp_paper" contains measurements of properties of pulp fibers and the paper made from them. In total, there're $n = 62$ observations measured on 8 variables that we can divide in two main groups according on the features they look for:

Paper Properties:

- **Breaking Length (BL):** Breaking Length is generally used in paper trading to characterize the intrinsic strength of paper. It offers an excellent basis for comparing the strength of cards made from different furniture and having different base weight.
- **Elastic Modulus (EM):** Elastic Modulus is essentially the rigidity of a material. In other words, it's the ease with which it's bent or stretched.
- **Stress at Failure (SF):** Stress at Failure is a measure of how long the sample is stretched at failure.
- **Burst Strenght (BS):** Burst Strength says how much pressure paper he can tolerate before the breakage.

Fiber Features:

- **Arithmetic Fiber Length (AFL):** Is the sum of all lengths of the individual fibers divided by the total number of fibers measured.
- **Long Fiber Fraction (LFF):** Fraction of long fibers in the observation.
- **Fine Fiber Fraction (FFF):** Fraction of fine fibers in the observation.
- **Zero Span Tensile (ZST):** It provides an idea of fiber tensile strength and not fiber bonding strength. It is measured with a strip of paper, but mainly a property of pulp.

Data importation

We display the first rows of the dataset:

```
##       BL    EM    SF    BS    AFL    LFF    FFF   ZST
## 1 21.312 7.039 5.326 0.932 -0.030 35.239 36.991 1.057
## 2 21.206 6.979 5.237 0.871  0.015 35.713 36.851 1.064
## 3 20.709 6.779 5.060 0.742  0.025 39.220 30.586 1.053
## 4 19.542 6.601 4.479 0.513  0.030 39.756 21.072 1.050
## 5 20.449 6.795 4.912 0.577 -0.070 32.991 36.570 1.049
## 6 20.841 6.919 5.108 0.784 -0.050 31.140 38.115 1.052
```

1.1) Obtain the maximum likelihood solution for m = 2 and m = 3 common factors on the standardize observations and compute the proportion of total sample variance due to each factor. List the estimated communalities, specific variances, and the residual matrix $S = \widehat{L}\widehat{L}^T + \widehat{\Psi}$. Compare the result. Which choice of m do you prefer? Why?

To start, we can have a look at the correlation structure of Data in order to see if there're interesting patterns of correlations:

```
##             BL         EM         SF         BS        AFL        LFF
## BL   1.0000000  0.9138256  0.9838790  0.9875554  0.6477987  0.7350138
## EM   0.9138256  1.0000000  0.9422199  0.8746665  0.5370190  0.6085413
## SF   0.9838790  0.9422199  1.0000000  0.9745114  0.6807025  0.7644251
## BS   0.9875554  0.8746665  0.9745114  1.0000000  0.7063811  0.7962528
## AFL  0.6477987  0.5370190  0.6807025  0.7063811  1.0000000  0.9055912
## LFF  0.7350138  0.6085413  0.7644251  0.7962528  0.9055912  1.0000000
## FFF -0.5418813 -0.5559586 -0.5745904 -0.5636592 -0.7334321 -0.7109855
## ZST  0.8217782  0.8495981  0.8651424  0.8132479  0.7842212  0.7927309
##             FFF        ZST
## BL  -0.5418813  0.8217782
## EM  -0.5559586  0.8495981
## SF  -0.5745904  0.8651424
## BS  -0.5636592  0.8132479
## AFL -0.7334321  0.7842212
## LFF -0.7109855  0.7927309
## FFF  1.0000000 -0.7845570
## ZST -0.7845570  1.0000000
```

We see that there're really strong correlations between the predictors and so we can analyze better their values by ordering them and see the details for each single variable:

```
##    Variable.s.names Correlations
## 1            BS : BL    0.9875554
## 2            SF : BL    0.9838790
## 3            BS : SF    0.9745114
## 4            SF : EM    0.9422199
## 5            EM : BL    0.9138256
## 6          LFF : AFL    0.9055912
## 7            BS : EM    0.8746665
## 8           ZST : SF    0.8651424
## 9           ZST : EM    0.8495981
## 10          ZST : BL    0.8217782
```

```
## 11         ZST : BS    0.8132479
## 12         LFF : BS    0.7962528
## 13        ZST : LFF    0.7927309
## 14        ZST : AFL    0.7842212
## 15         LFF : SF    0.7644251
## 16         LFF : BL    0.7350138
## 17         AFL : BS    0.7063811
## 18         AFL : SF    0.6807025
## 19         AFL : BL    0.6477987
## 20         LFF : EM    0.6085413
## 21         AFL : EM    0.5370190
## 22         FFF : BL   -0.5418813
## 23         FFF : EM   -0.5559586
## 24         FFF : BS   -0.5636592
## 25         FFF : SF   -0.5745904
## 26        FFF : LFF   -0.7109855
## 27        FFF : AFL   -0.7334321
## 28        ZST : FFF   -0.7845570
```

Looking at the grid above, we detect that some variables have a strong level of their correlation coefficient, indeed the smallest absolute values for are around *0.53* while the highest are even close to *1.* To have a clearer view of some possible groups of highly correlated variables we can see in detail the levels of correlations for every single variable:

```
##     Ordered Correlations:  BL
## BS                  0.9875554
## SF                  0.9838790
## EM                  0.9138256
## ZST                 0.8217782
## LFF                 0.7350138
## AFL                 0.6477987
## FFF                -0.5418813
```

```
##     Ordered Correlations:  EM
## SF                  0.9422199
## BL                  0.9138256
## BS                  0.8746665
## ZST                 0.8495981
## LFF                 0.6085413
## AFL                 0.5370190
## FFF                -0.5559586
```

```
##     Ordered Correlations:  SF
## BL                  0.9838790
## BS                  0.9745114
## EM                  0.9422199
## ZST                 0.8651424
## LFF                 0.7644251
## AFL                 0.6807025
## FFF                -0.5745904
```

```
##     Ordered Correlations:  BS
## BL                  0.9875554
## SF                  0.9745114
```

```
## EM                  0.8746665
## ZST                 0.8132479
## LFF                 0.7962528
## AFL                 0.7063811
## FFF                -0.5636592
```

```
##     Ordered Correlations:  AFL
## LFF                  0.9055912
## ZST                  0.7842212
## BS                   0.7063811
## SF                   0.6807025
## BL                   0.6477987
## EM                   0.5370190
## FFF                 -0.7334321
```

```
##     Ordered Correlations:  LFF
## AFL                  0.9055912
## BS                   0.7962528
## ZST                  0.7927309
## SF                   0.7644251
## BL                   0.7350138
## EM                   0.6085413
## FFF                 -0.7109855
```

```
##     Ordered Correlations:  FFF
## BL                  -0.5418813
## EM                  -0.5559586
## BS                  -0.5636592
## SF                  -0.5745904
## LFF                 -0.7109855
## AFL                 -0.7334321
## ZST                 -0.7845570
```

```
##     Ordered Correlations:  ZST
## SF                   0.8651424
## EM                   0.8495981
## BL                   0.8217782
## BS                   0.8132479
## LFF                  0.7927309
## AFL                  0.7842212
## FFF                 -0.7845570
```

From this *"Correlation Report"* above we can confirm the presence of groups of highly correlated variables. In particular, we can detect the following possible grouping:

- **Variables "BL", "EM", "SF", "BS":** The first 4 variables shows really high correlations between them (in particular, each of them has a correlation coefficient with all the other that is always between 0.87 and 0.98). They're also quite good correlated with the rest of the variables in the dataset but with lower absolute values (between 0.53 and around 0.8). So, since these are also the measurements related to the paper properties, it seems natural to consider them as a part of a single group ("Group 1").
- **Variables "AFL" and "LFF":** Also these two measurements have a strong correlation coefficient (around 0.9) while their coefficients in absolute value with the other are smaller (from 0.53 to 0.79), even if they still have remarkable values. So we can consider these variables as a part of another group ("Group 2").
- **Variable "FFF":** The variable related to the fraction of the fine fibers is the only one which is

negatively correlated with all the other. This particular behavior may be related to the feature that it measures (as we'll see later in the attempt to interpret the factors). Anyway, looking at the absolute values of the coefficients, we can see that it's lowly correlated with the predictors we put in the "Group 1" (all of the coefficients are between -0.53 and -0.57) while it has stronger correlations with the variables of the "Group 2" and with "ZST".

- **Variable "ZST":** This predictor seems to have quite strong levels of correlation with all the other variable (all the coefficients in absolute value are between 0.78 and 0.86) and so, until now, it's difficult to assign it to a group instead of another.

Finally, we can have a look at the eigenvalues of the Correlation Matrix *"R"* to see if some of them are close to "0" (which may can indicate that the intrinsic dimensionality of the data is smaller than $p = 8$):

```
## Eigenvalues of R:
```

```
## 6.393944 0.9176631 0.4096977 0.1394933 0.07811816 0.04735759 0.008658692 0.005067234
```

We can see that we have really low values in particular for the last two eigenvalues and this confirm that this dataset has a correlation structure that is adequate to perform a tecnique of dimensionality reduction such as the *"Factor Analysis"* (the only issue can be related to the fact that the variable are strongly correlated and not only the ones that form a group).

Factor Analysis
We perform Factor Analysis in order to identify *"m<p"* unobserved latent sources (called "Factors") that we use to represent our Data. Indeed, these Factors are Random Variables $F = (F_1, ..., F_m)$ that can express the original $X = (X_1, ..., X_p)$ by a linear combination:

$$X - \mu = LF + \epsilon$$

where: 1) $\mu = (\mu_1, ..., \mu_p)$: Is the Population Mean Vector of X, 2) L: Is the $m \times p$ Matrix of Factor Loadings, which contains the coefficients of the linear combination and is the object estimated in the model, 3) $\epsilon = (\epsilon_1, ..., \epsilon_p)$: Vector of the Errors.

We estimate the Matrix of Loadings using the maximum Likelyhood approach and we choose the "Varimax" rotation in order to obtain a set of coefficients that allow us to interpret easily the underlying Factors. Data will be standardized, so we'll deal with Loadings scaled by the standard deviations "σ_j" of the variable X_j they're referred to (so in that case, the Sample Covariance Matrix S is equal to the Sample Correlation Matrix R).

Choice of the number of Factors
To choose the best number of Factors to retain in the Model, we wish both to reach a satisfactory dimensionality reduction and to explain well the original Data. To evaluate those aspects, we'll perform Factor Analysis on *m=2* and *m=3* Factors and we'll look at:

- **Proportion of Total Sample Variance:** It's the amount of the total variation explained by the m factors (sum of the squared columns of L) considered with respect to the overall variance of data (equal to $trace(R)$, a good percentage is considered to be around the 80
- **Approximation of R:** Since the Factor Analysis Model aims to give a good approximation of the Sample Correlation Matrix $R = \hat{L}\hat{L}^T + \hat{\Psi}$ (where $\hat{\Psi}$ is the diagonal matrix that contains the "Specific Variances", so the (j, j)-th element of this matrix is the portion of variance of the variable X_j not explained by the Factors). We'll compute the squared Frobenius Norm of the "Residual Matrix" ($\|R - (\hat{L}\hat{L}^T + \hat{\Psi})\|_F^2 = trace((R - (\hat{L}\hat{L}^T + \hat{\Psi}))(R - (\hat{L}\hat{L}^T + \hat{\Psi}))^T)$) which is equal to the sum of the squares of all the elements of the residual matrix and allows us to understand how better R is approximated by the model.

- **Communalities and Specific Variances:** We'll see which are the variables that are explained better by the Factors chosen and which of them may require an additional Factor (since the communalities are defined as the portion of variance of the variable X_j explained by all the factors considered, and they're computed as the sum of the squares of the j-th row of $\widehat{L}$).
- **Interpretability of the Factors:** Considering all the previous element, we'll evaluate the number of Factors also considering how much they're able to give a clear interpretation of the groups of variables. To do so, we'll look at the Loading Matrix and notice if there're variables that express a sufficiently high coefficient with respect to a certain factor (an acceptable threshold for the absolute value of the coefficient is around 0.6 or higher).

Model with m = 2
We show the results obtained setting the number of Factors *m=2*, starting from the Matrix of Loadings $\widehat{L}$:

```
##        Factor1    Factor2
## BL   0.9213739  0.3827685
## EM   0.8746995  0.2875079
## SF   0.8882650  0.4330007
## BS   0.8770311  0.4632941
## AFL  0.3267184  0.9050746
## LFF  0.4506678  0.8408353
## FFF -0.2955999 -0.7044288
## ZST  0.6320881  0.6279802
```

Looking at the values of the Loadings for the different variables, we can see that there's a clear distinction of two main groups of variables related to different Factors. Indeed, the first four variables loads higly on the first Factor and lowly on the second one, vice-versa for the variables "AFL","LFF", "FFF". This is coherent with the possible grouping that we identify before looking at the correlation structure of Data, and the variable "FFF" (the one which was negatively correlated with all the other measurements) here is assigned at the "Group 2" of variables. The only difficult interpretation is for the variable "ZST" (the one which was quite equally correlated with all the others) because the loadings doesn't clearly assign this measurement to one of the groups detected since its loadings are actually equal for F_1 and F_2.
Now we look at the proportion of variance explained by F_1 and F_2:

```
##           Proportion of Variance Cumulative Proportion of Variance
## Factor  1              0.4961235                          0.4961235
## Factor  2              0.3810058                          0.8771294
```

We can see that both the factors explain a good percentage of the total variation cause for F_1 we've $\sum_{j=1}^{p} h_{1,j}^2 \simeq 0.5$ and for F_2 we've $\sum_{j=1}^{p} h_{2,j}^2 \simeq 0.38$ (confirming that almost two factors are necessary to explain well the total variation related to the two main groups of variables). The cumulative proportion is around 0.88 and so we can conclude that it's a satisfactory percentage.
To be more precise, we can see the communalities and specific variances for each of the X_j's, so to understand which of them has been well-explained and which may require an additional Factor (we recall that since we've standardized our data, $Var[X_j] = 1$, $j = 1, ..., p$):

```
##                                BL      EM         SF         BS        AFL
## Estimated Communality 0.995441595 0.84776 0.97650442 0.98382494 0.92590495
## Specific Variance     0.004558405 0.15224 0.02349558 0.01617506 0.07409505
##                               LFF       FFF       ZST
## Estimated Communality 0.91010535 0.5835993 0.7938944
## Specific Variance     0.08989465 0.4164007 0.2061056
```

We can see that with two factors we reach a really good proportion of variance for each of the first 6 measurements (indeed, the communalities for them are all between 0.91 and 0.99), so they don't require an

additional factor to be well approximated. Also variable "ZST" reach an acceptable percentage of variance explained by the factors (0.79) while the only issue is for variable "FFF" that has a low communality if compared to the other (indeed, $\psi_{FFF} \simeq 0.41$). So we can confirm once again that *m=2* is a correct number of factors to explain the variables for which we identify to be part of two possible groups ("BL", "EM", "SF", "BS" for Group 1 and "AFL", "LFF" for Group 2), while variable "FFF" may require an additional one.
Now we reproduce the decomposition of the Sample Correlation Matrix ($R = \widehat{L}\widehat{L}^T + \widehat{\Psi}$) and we compute the squared Frobenius Norm of the Residual Matrix ($R - (\widehat{L}\widehat{L}^T + \widehat{\Psi})$):

1) $\widehat{L}\widehat{L}^T$**:**

```
##             BL         EM         SF         BS        AFL        LFF
## BL   0.9954416  0.9159742  0.9841633  0.9854079  0.6474639  0.7370788
## EM   0.9159742  0.8477600  0.9014561  0.9003393  0.5459965  0.6359456
## SF   0.9841633  0.9014561  0.9765044  0.9796427  0.6821105  0.7643947
## BS   0.9854079  0.9003393  0.9796427  0.9838249  0.7058580  0.7848037
## AFL  0.6474639  0.5459965  0.6821105  0.7058580  0.9259049  0.9082601
## LFF  0.7370788  0.6359456  0.7643947  0.7848037  0.9082601  0.9101054
## FFF -0.5419913 -0.4610899 -0.5675893 -0.5856081 -0.7341386 -0.7255260
## ZST  0.8227605  0.7334364  0.8333776  0.8453004  0.7748837  0.8128896
##            FFF        ZST
## BL  -0.5419913  0.8227605
## EM  -0.4610899  0.7334364
## SF  -0.5675893  0.8333776
## BS  -0.5856081  0.8453004
## AFL -0.7341386  0.7748837
## LFF -0.7255260  0.8128896
## FFF  0.5835993 -0.6292125
## ZST -0.6292125  0.7938944
```

2) $\widehat{\Psi}$**:**

```
##             [,1]    [,2]       [,3]       [,4]       [,5]       [,6]       [,7]
## [1,] 0.004558405 0.00000 0.00000000 0.00000000 0.00000000 0.00000000 0.0000000
## [2,] 0.000000000 0.15224 0.00000000 0.00000000 0.00000000 0.00000000 0.0000000
## [3,] 0.000000000 0.00000 0.02349558 0.00000000 0.00000000 0.00000000 0.0000000
## [4,] 0.000000000 0.00000 0.00000000 0.01617506 0.00000000 0.00000000 0.0000000
## [5,] 0.000000000 0.00000 0.00000000 0.00000000 0.07409505 0.00000000 0.0000000
## [6,] 0.000000000 0.00000 0.00000000 0.00000000 0.00000000 0.08989465 0.0000000
## [7,] 0.000000000 0.00000 0.00000000 0.00000000 0.00000000 0.00000000 0.4164007
## [8,] 0.000000000 0.00000 0.00000000 0.00000000 0.00000000 0.00000000 0.0000000
##           [,8]
## [1,] 0.0000000
## [2,] 0.0000000
## [3,] 0.0000000
## [4,] 0.0000000
## [5,] 0.0000000
## [6,] 0.0000000
## [7,] 0.0000000
## [8,] 0.2061056
```

3) **Residual Matrix:**

```
##                BL           EM            SF            BS           AFL
## BL   0.0000000000 -0.002148674 -2.843073e-04  0.0021474876  0.0003348155
## EM  -0.0021486745  0.000000000  4.076378e-02 -0.0256728242 -0.0089774836
## SF  -0.0002843073  0.040763781  0.000000e+00 -0.0051313097 -0.0014080323
## BS   0.0021474876 -0.025672824 -5.131310e-03  0.0000000000  0.0005231226
## AFL  0.0003348155 -0.008977484 -1.408032e-03  0.0005231226  0.0000000000
## LFF -0.0020650075 -0.027404289  3.043716e-05  0.0114491873 -0.0026688674
## FFF  0.0001099772 -0.094868698 -7.001138e-03  0.0219489105  0.0007064916
## ZST -0.0009823212  0.116161709  3.176476e-02 -0.0320524814  0.0093374294
##               LFF           FFF           ZST
## BL  -2.065008e-03  0.0001099772 -0.0009823212
## EM  -2.740429e-02 -0.0948686984  0.1161617093
## SF   3.043716e-05 -0.0070011383  0.0317647586
## BS   1.144919e-02  0.0219489105 -0.0320524814
## AFL -2.668867e-03  0.0007064916  0.0093374294
## LFF  0.000000e+00  0.0145404313 -0.0201586404
## FFF  1.454043e-02  0.0000000000 -0.1553444791
## ZST -2.015864e-02 -0.1553444791  0.0000000000
```

4) **Squared Frobenius Norm:**

```
## 0.1064639
```

We can conclude that the model with *m=2* performs a good approximation of the Sample Correlation Matrix R since the value of the squared Frobenius Norm is sufficiently low. Here again, the main issues in the approximation are related to the last two variables, indeed if we look at the Residual Matrix, we can notice that the correlations of these two variables with the other measurements (which are the element off of the main diagonal in the columns "FFF" and "ZST") are in general the worst approximated (even if they still obtain a good replication). For instance, the worst approximation errors are $\simeq -0.15$ for the correlation between "ZST" and "FFF" and $\simeq 0.12$ for the correlation between "ZST" and "EM". Again, this result suggest that maybe an additional factor can improve the explanation of the last two variables, but anyway we can be satisfied of the results obtained for *m=2*.

Model with m = 3

Now, we consider the Model with *m=3* Factors, again starting by the Loading Matrix:

```
##        Factor1    Factor2     Factor3
## BL   0.9172865  0.3756688  0.10286850
## EM   0.8488135  0.2421478  0.44480594
## SF   0.8802950  0.4133204  0.20891003
## BS   0.8841863  0.4612491  0.02726793
## AFL  0.3310154  0.8917119  0.09293561
## LFF  0.4583223  0.8424389  0.03338921
## FFF -0.2577291 -0.6987779 -0.37133050
## ZST  0.5893794  0.6101345  0.46029325
```

The loading's structure for the case of *m=3* is pretty much the same that we saw in the previous model with *m=2*: There're still two clear groups of variables related to the two different factors (the same detected before), but adding a factor, the absolute values of the highest loading for each variable is lower than before, meaning that the third factor makes the pattern of loadings scarcely less interpretable than before. Variable "ZST" has still loadings that are pretty much the same for each factors so we can't clearly assign it to a specific group instead than another (again, this is coherent with the fact that this measurement has values

of the correlations with the other that are very similar and maybe this can depend on the nature of this variable). So, considering what we've said above and taking into account the fact that F_3 does not even have a sufficiently high coefficient for none of the X_j's, we can conclude that adding a new factor doesn't improve the interpretability and less clear patterns than the case of *m=2*.
Let's go through the proportion of variance explained:

```
##           Proportion of Variance Cumulative Proportion of Variance
## Factor  1             0.48150329                          0.4815033
## Factor  2             0.36859439                          0.8500977
## Factor  3             0.07654116                          0.9266388
```

As could expect, increasing the number of factors leads to explain an higher proportion of variance than before. Anyway, the variation from the previous case is not very huge (from $\simeq 0.87$ to $\simeq 0.93$), because the third factor accounts for a little proportion of variance (around the 7.7%) so its contribution is not extremely significant. To sum up, we obtain a better overall explanation but most of the total variance can be explained even by considering only *m=2* factors.
An useful advantage of choosing 3 factors can be seen by looking at the communalities:

```
##                                BL         EM         SF          BS         AFL
## Estimated Communality 0.993123526 0.97697218 0.98939634 0.995279762 0.91335826
## Specific Variance     0.006876474 0.02302782 0.01060366 0.004720238 0.08664174
##                               LFF        FFF        ZST
## Estimated Communality 0.92087752 0.6926012 0.93150203
## Specific Variance     0.07912248 0.3073988 0.06849797
```

From the table above, we can notice that the third factor is important to explain the variability of the last two measurements. Indeed, the communalities for the first 6 X_j's remain the same seen in the previous model (actually, they are a bit lower than before, but the variation is very small) while we see a remarkable improvement for "FFF" and "ZST" (their communalities change from 0.58 to 0.69 and from 0.79 to 0.93). This means that the contribution provided by the third factor in explaining the total variance, is mainly resulting from the better approximation that this model gives for these variables. So, a good reason to choose *m=3* factors instead of *m=2* may be related to the better reproduction of "FFF" and "ZST".
We end the comparison showing the Residual Matrix and its Frobenius Norm:

1) $\widehat{L}\widehat{L}^T$**:**

```
##             BL         EM         SF         BS         AFL         LFF
## BL   0.9931235  0.9153290  0.9842445  0.9871341  0.6481845  0.7403256
## EM   0.9153290  0.9769722  0.9402153  0.8743287  0.5382347  0.6078766
## SF   0.9842445  0.9402153  0.9893963  0.9746850  0.6793691  0.7586313
## BS   0.9871341  0.8743287  0.9746850  0.9952798  0.7065148  0.7947270
## AFL  0.6481845  0.5382347  0.6793691  0.7065148  0.9133583  0.9060276
## LFF  0.7403256  0.6078766  0.7586313  0.7947270  0.9060276  0.9208775
## FFF -0.5371187 -0.5531415 -0.5932714 -0.5603166 -0.7429307 -0.7191991
## ZST  0.8171880  0.8527571  0.8671686  0.8150965  0.7819354  0.7994956
##             FFF        ZST
## BL  -0.5371187  0.8171880
## EM  -0.5531415  0.8527571
## SF  -0.5932714  0.8671686
## BS  -0.5603166  0.8150965
## AFL -0.7429307  0.7819354
## LFF -0.7191991  0.7994956
## FFF  0.6926012 -0.7491696
## ZST -0.7491696  0.9315020
```

2) $\widehat{\Psi}$:

```
##             [,1]       [,2]       [,3]        [,4]       [,5]       [,6]
## [1,] 0.006876474 0.00000000 0.00000000 0.000000000 0.00000000 0.00000000
## [2,] 0.000000000 0.02302782 0.00000000 0.000000000 0.00000000 0.00000000
## [3,] 0.000000000 0.00000000 0.01060366 0.000000000 0.00000000 0.00000000
## [4,] 0.000000000 0.00000000 0.00000000 0.004720238 0.00000000 0.00000000
## [5,] 0.000000000 0.00000000 0.00000000 0.000000000 0.08664174 0.00000000
## [6,] 0.000000000 0.00000000 0.00000000 0.000000000 0.00000000 0.07912248
## [7,] 0.000000000 0.00000000 0.00000000 0.000000000 0.00000000 0.00000000
## [8,] 0.000000000 0.00000000 0.00000000 0.000000000 0.00000000 0.00000000
##           [,7]       [,8]
## [1,] 0.0000000 0.00000000
## [2,] 0.0000000 0.00000000
## [3,] 0.0000000 0.00000000
## [4,] 0.0000000 0.00000000
## [5,] 0.0000000 0.00000000
## [6,] 0.0000000 0.00000000
## [7,] 0.3073988 0.00000000
## [8,] 0.0000000 0.06849797
```

3) **Residual Matrix:**

```
##                BL             EM             SF             BS            AFL
## BL   0.0000000000 -0.0015034646 -0.0003655615  0.0004213123 -0.0003857217
## EM  -0.0015034646  0.0000000000  0.0020046308  0.0003378556 -0.0012157073
## SF  -0.0003655615  0.0020046308  0.0000000000 -0.0001735520  0.0013334403
## BS   0.0004213123  0.0003378556 -0.0001735520  0.0000000000 -0.0001337087
## AFL -0.0003857217 -0.0012157073  0.0013334403 -0.0001337087  0.0000000000
## LFF -0.0053117948  0.0006647280  0.0057938289  0.0015258655 -0.0004363556
## FFF -0.0047625985 -0.0028171708  0.0186809841 -0.0033425081  0.0094985638
## ZST  0.0045902161 -0.0031590242 -0.0020262415 -0.0018485477  0.0022857283
##               LFF          FFF          ZST
## BL  -0.0053117948 -0.004762599  0.004590216
## EM   0.0006647280 -0.002817171 -0.003159024
## SF   0.0057938289  0.018680984 -0.002026242
## BS   0.0015258655 -0.003342508 -0.001848548
## AFL -0.0004363556  0.009498564  0.002285728
## LFF  0.0000000000  0.008213598 -0.006764623
## FFF  0.0082135979  0.000000000 -0.035387410
## ZST -0.0067646234 -0.035387410  0.000000000
```

4) **Squared Frobenius Norm:**

```
## 0.003930366
```

As predictably, by adding a third factor we can see that the approximation of R is better than before. We can notice this first of all by looking at the Frobenius Norm of the Residual Matrix which now is $\simeq 0.0039$ instead of $\simeq 0.11$, so the improvement in remarkable. Again, if we watch the Residual Matrix, this improvement may be related to the better explanation of the last two variables, indeed we saw before that there were in particular to correlations that achieved a certain error in the replication ($\simeq -0.15$ for the correlation between "ZST" and "FFF" and $\simeq 0.12$ for the correlation between "ZST" and "EM"), instead now these two errors in the approximation are lower.

Conclusions

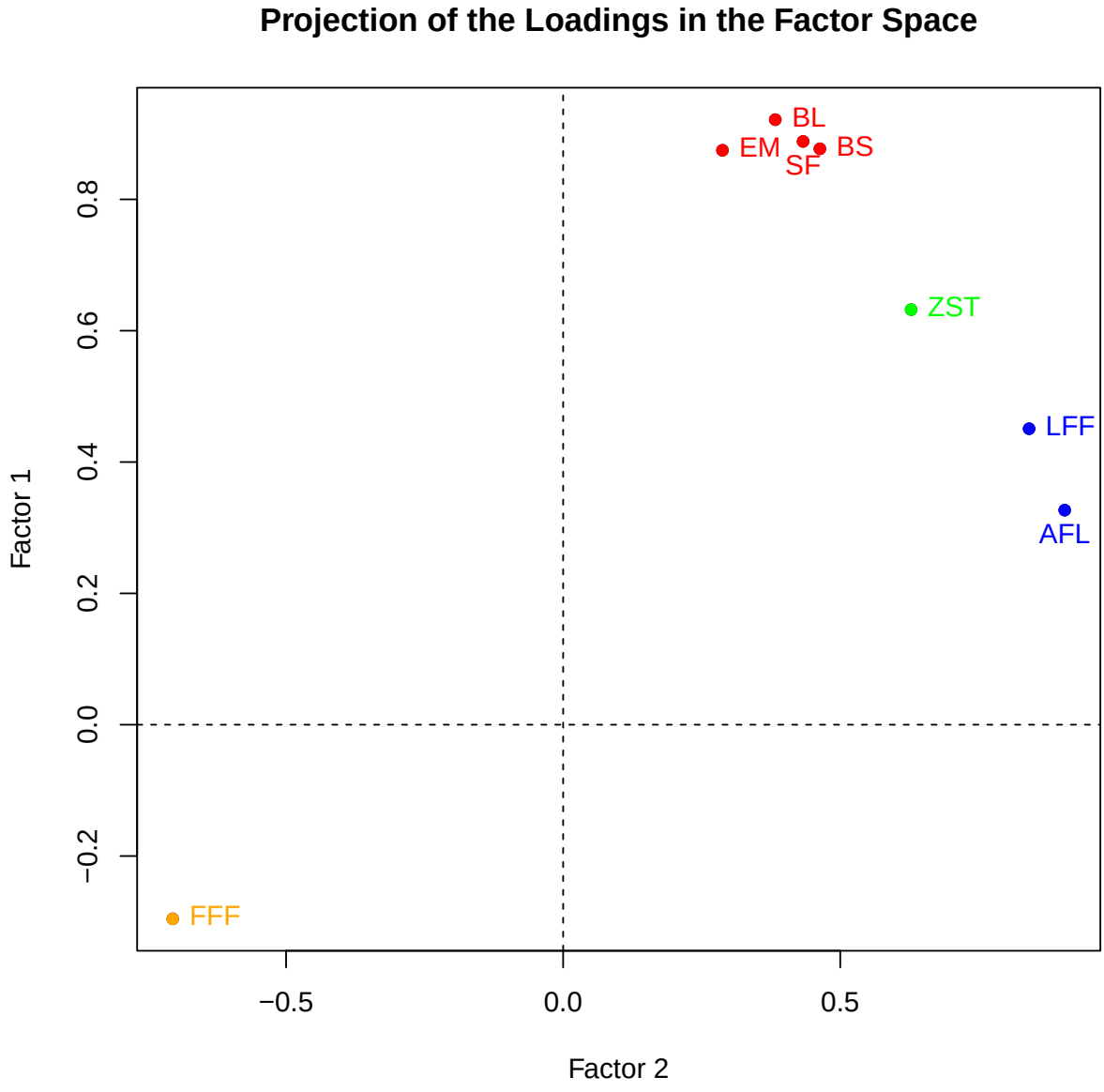
All considered, there isn't a clear preference about which could be the best number of factors between *m=2* and *m=3* since in the first case we obtain a clearer pattern of loadings with respect to the factors (except for the last variable) and an higher reduction in dimensionality, while the inclusion of the third factor produces a better explanation of the variance and correlation structure of the last two measurements. Anyway, since the proportions of variance explained are sufficiently high in both the models, we can say that, for interpretation's aims, we prefer to retain *m=2* due to the fact both that we can distinguish two evident groups of variables and that we can plot in just two dimensions the loadings in the Factor space.

1.2) Give an interpretation to the common factors in the m = 2 solution

To assign a particular interpretation to the 2 Factors estimated by the model, we'll take into account:

- **Loadings:** Similarly to what we saw at section 1.1) we'll consider the values and signs of the loadings and their plot in the Factor Space in order to distinguish the groups of variables.
- **Definitions of the variables:** In order to assign a possible name and a meaning to the Factors, we look at the definitions of the variables which should be coherent with the values and signs of the loadings.

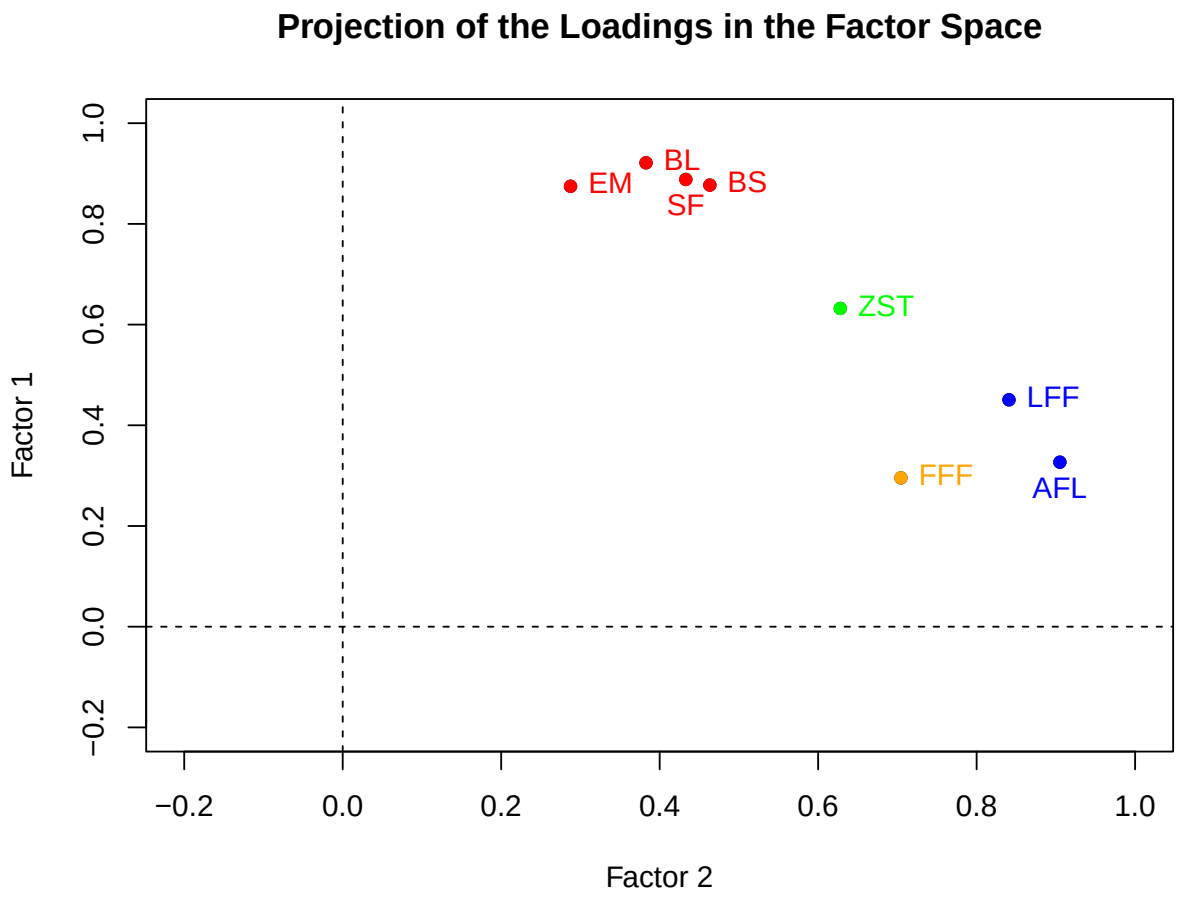
Plot of the loadings:



As we can see from the above plot, there're two clear groups of variables with respect to the position of the loadings in the factor space:

1) **Group 1:**
The first cluster of loadings is referred to the first 4 variables of the dataset ("BL", "EM", "SF", "BS") which we found before to have high values of the loadings for the first factor and low values for the second one. These variables were all highly correlated. These variables all belong to the set of measurements that express the "paper properties".

2) **Group 2:**
The second cluster of loadings is made of variables "LFF" and "AFL", which had a strong value of their correlation coefficient and are measurements related to the characteristics of the fibers that compose the pulp paper. We have to notice that also variable "FFF" could be considered as part of this cluster, indeed as we said before, its higher correlation coefficients (in absolute values) were the ones related to variables "LFF" and "ZST" and it also shows a remarkable loading with respect to the second factor while the loading related to the first one is not significant. In addiction, also "FFF" is a measurement about the fiber features, so it seems logical to consider it as part of this cluster. Graphically, it appears distant from the loadings of "ZST" and "LFF" due to its negative value (which is related to the neagtive correlations that "FFF" has with all th other that may can be a consequence of the characteristic underlined by it), but if we consider its absolute value, we can see that it's very close to "ZST" and "LFF":



2) **Variable "ZST":**

Looking at the position of the loadings of this variable in the Factor Space, we can see that it doesn't belong clearly to a specific group since it's exactly in the middle point between the clusters we identified as "Group 1" and "Group 2" (indeed, it has values of the loadings that are over the threshold of 0.6 for both the factors). Recalling that this variable has also quite equal values of the correlation coefficients with all the other measurement, we can say that it influences in the same manner both the variables that load highly on Factor 1 and the ones that load higly on Factor 2.

Now, we try to give an interpretation to the two factors, looking also at the definitions of the measurements:

- **Factor 1: Pulp Structural Factor (Strength and Elasticity)**
  F_1 is mainly related to variables that measure some structural properties of the paper pulp, such as its strength ("BL" and "BS") and its elasticity ("EM", "SF"). Also "ZST", which express a certain influence on F_1 is about the tensile strength of the paper pulp (even if it's measured on the fibers). So, it can be coherent to interpret this Factor as an overall measure related to the structural properties of the paper pulp, in particular it's about the strength and elasticity features.
- **Factor 2: Fiber Length Factor**
  F_2 loads highly on variables which are about characteristics related to to the dimension and composition of the fibers ("AFL", "LFF", "FFF"), indeed the values and signs of the correlations between them is a logical consequence of their definitions (since "AFL" is the arithmetic length while "LFF" and "FFF" measure respectively the fraction of long and fine fibers). Looking again the relationship between these quantities and "ZST" (which expresses a paper property but is a variable measured on the fibers), it seems logical to have a certain pattern of correlations cause an higher portion of long fibers (which produce an high value also in their arithmetic length) may leads to a paper which has more tensile strength (and vice-versa for "FFF" which has a negative correlation with "ZST"). All considered, we can suppose that this Factor is mainly concerned on the length of the fibers.

1.3) Make a scatterplot of the factor scores for m = 2 obtained by the regression method. Is their correlation equal to zero? Should we expect so? Comment

"Factor Scores" $f_i = (f_{i1}, ..., f_{im})$, $i = 1, ..., n$ represent estimates of the values hired by the factors estimated in the model. They're unobserved quantities estimated by the so called "Regression Method" through the formula:

$$f_i = \widehat{L}^T S^{-1}(x_i - \bar{x})$$

And in our case they can computed and plotted for diagnostic pourposes about the estimated Factors. In particular we can check if the assumptions made on the model are sufficiently satisfied by the factors extracted:

- **Uncorrelation between Factors**
- **Gaussianity of the Factors**
- **Mean zero and unitary Variance**
- **Gaussianity of the Factors**

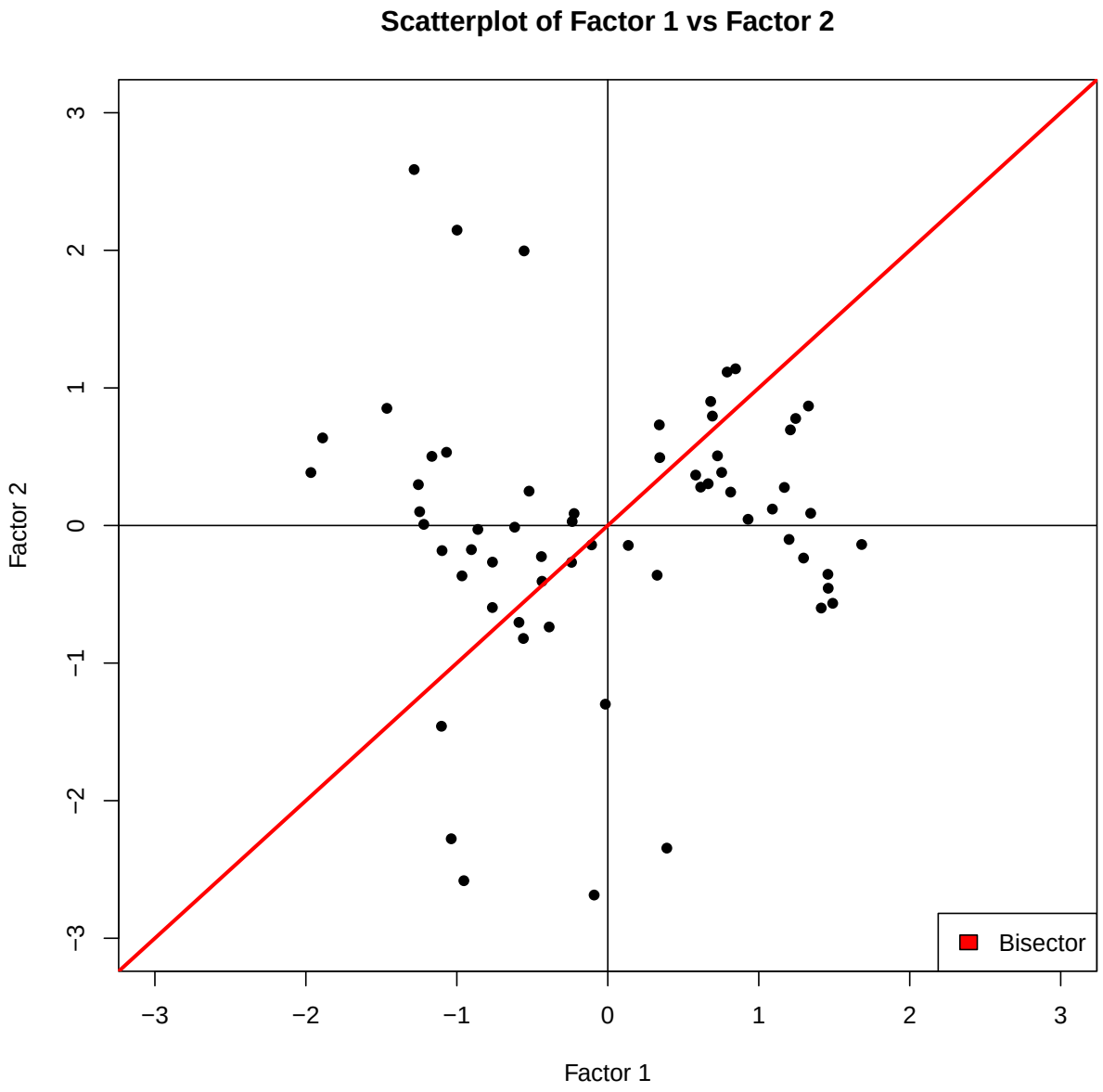
We start by computing the Factor Scores and displaying the first values:

```
##         Factor1     Factor2
## [1,] -0.1076162 -0.14090067
## [2,] -0.2360760  0.02895543
## [3,] -0.5215161  0.24958219
## [4,] -1.0685003  0.53191032
## [5,] -0.4397863 -0.22623463
## [6,] -0.2402071 -0.26762832
```

Then, we can rapidly check if the basic assumptions on Factors are satisfied:

```
##          Expectations Variances Correlation
## Factor 1 2.624469e-17 0.9815886  0.03223625
## Factor 2 1.024046e-17 0.9347222  0.03223625
```

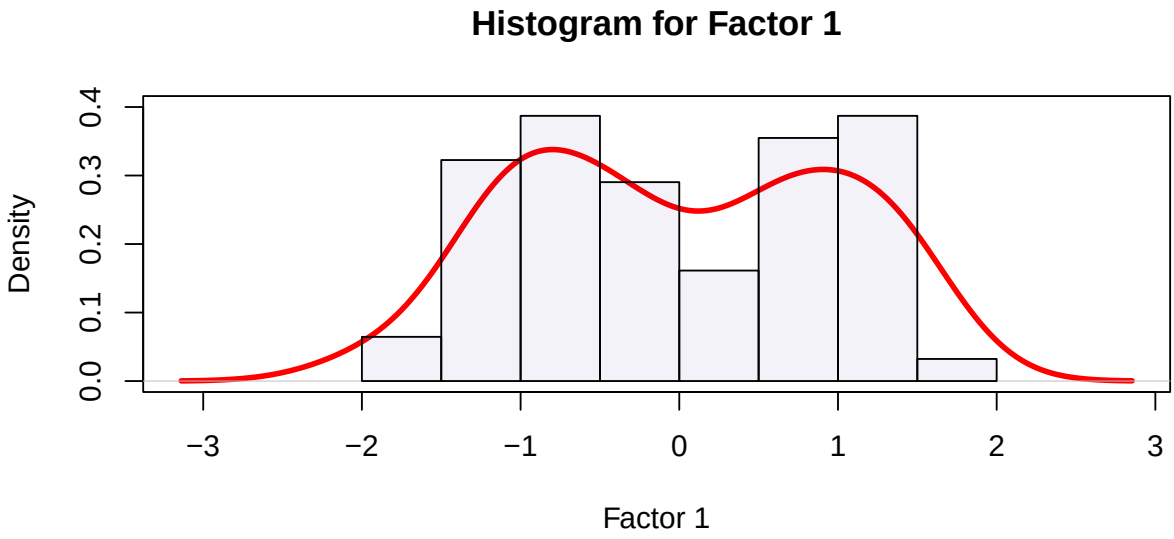
The assumptions are almost perfectly satisfied by the factors since the expectations are equal to "0" and their variances are very close to "1". About the correlation coefficient, it's nearly close to "0" ($\simeq 0.03$) and so we can conclude that the two factors are uncorrelated. We could expect such a result since uncorrelation between factors is one of the assumptions made in the model (as we said before). We can detect this aspect also by looking at the scatterplot of the factors scores:

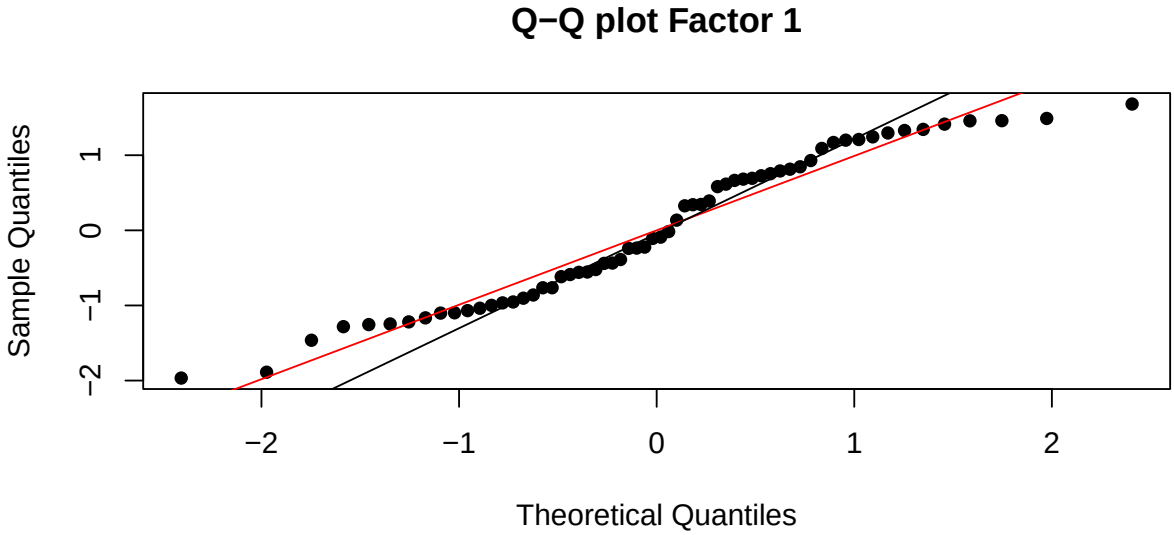


This plot graphically confirms what we saw before by computing the correlation coefficient, since the points seems to be scattered around the plane without a specific relationship, indeed they don't follow the bisector of the 1^{st} and 3^{rd} quadrant at all.

Then, we can check if normality assumption is satisfied plotting histograms and qq-plots of the factors:

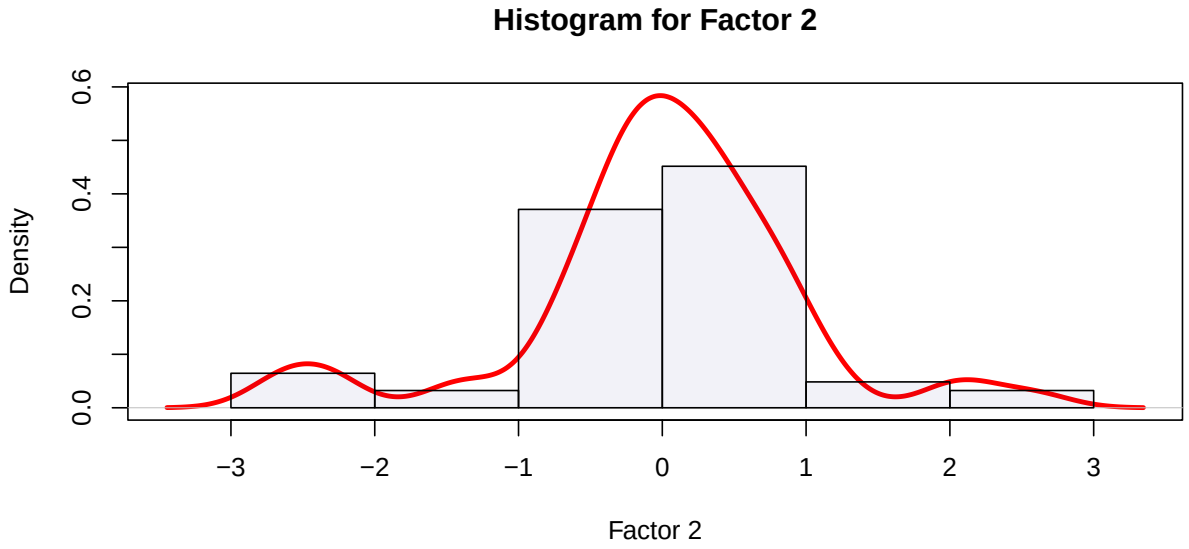
Factor 1:

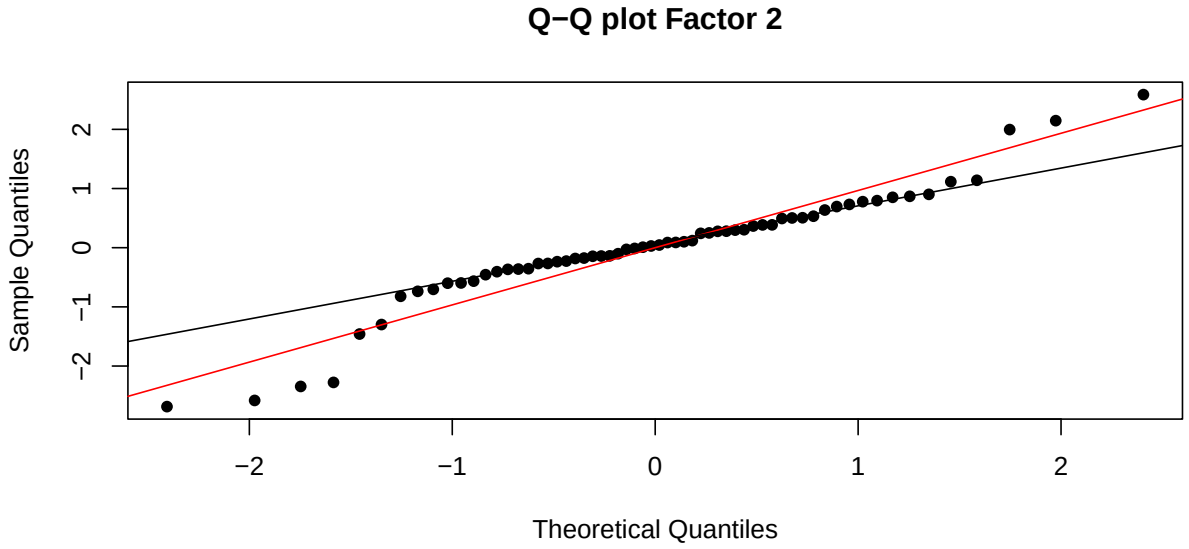




Both the histogram and the qq-plot clearly confirms that we can't consider F_1 to be normally distributed since it its density function shows 2 different peaks and its quantiles deviates from the theoretical ones it should have under gaussian assumption.

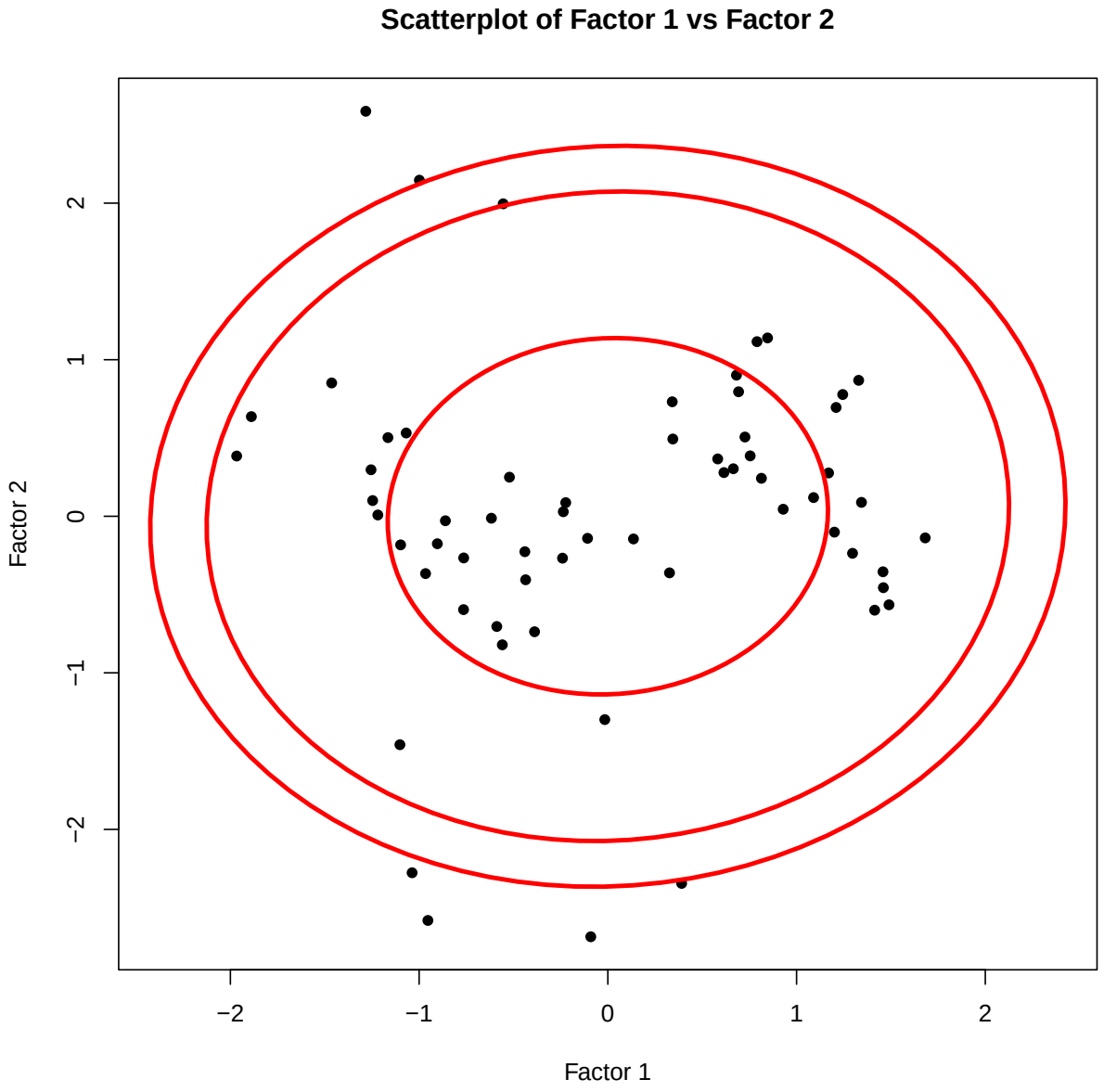
Factor 2:





The situation is a bit better if we consider F_2, indeed its histogram looks more like a Normal Distribution, except for two local peaks of density we detect for the lowest and highest values of the variable. Indeed, also the qq-plot shows that the quantiles of F_2 deviates from the theoretical quantiles of a Normal Distribution just for the initial and the final quantiles.

We can finally make use of the scatterplots of the factors to see if the cloud of points shows an elliptical shape (which means that $F = (F_1, F_2) \sim \mathcal{N}_2(\begin{bmatrix} 0, & 0 \end{bmatrix}, I)$):

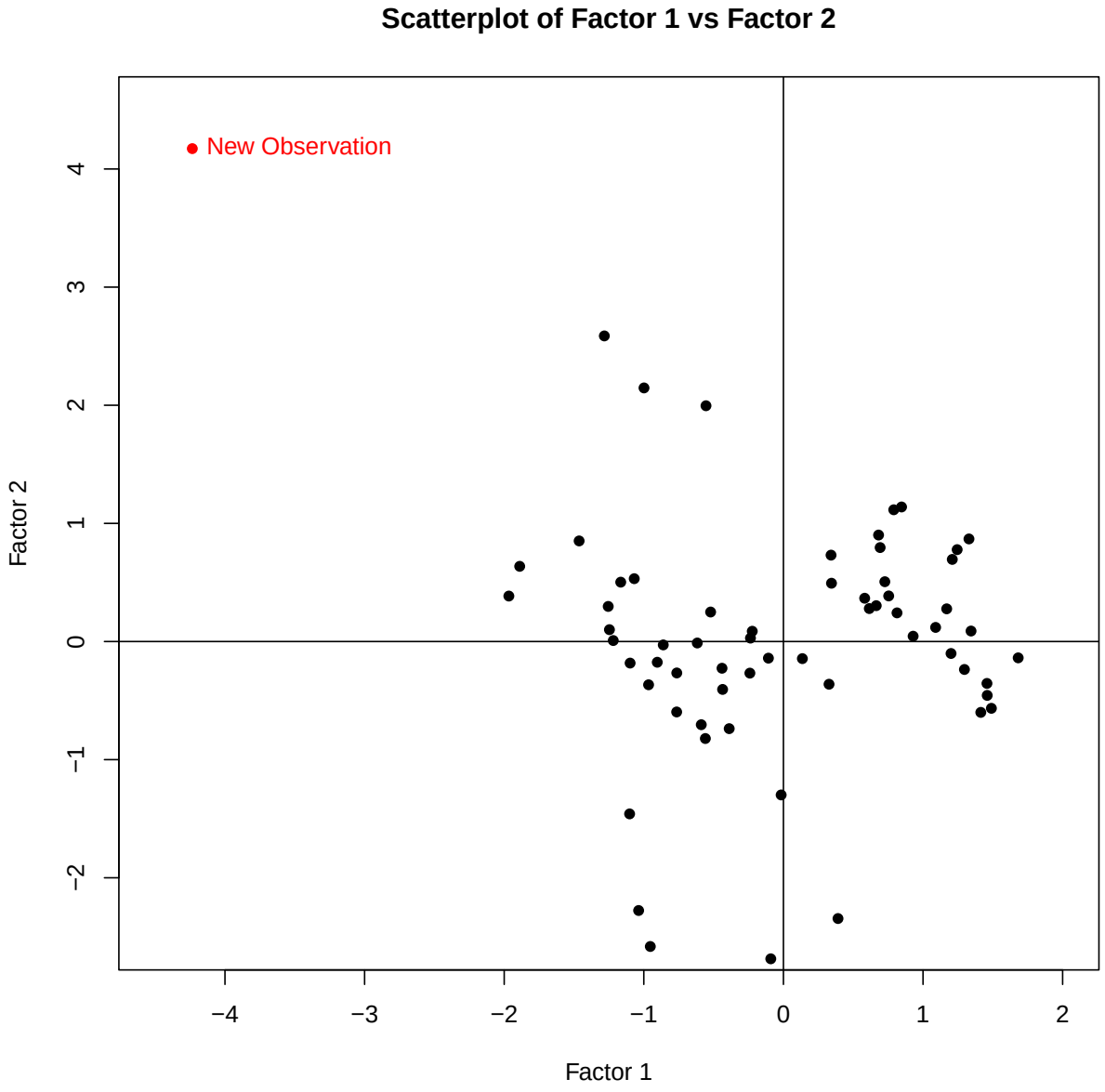


The bivariate plot of factor scores shows that the cloud of points seems to have an elliptical shape just for the low/middle quantiles, while for high quantiles we can't confirm that gaussianity assumption is satisfied. These considerations are confirmed by the numerical criterion which counts the number of observations above a certain quantile and compare it with the thoeretical ones we should found under normality of the factors:

```
##                Number of elements Observed Percentage Expected Percentage
## Quantile 0.25:                 48              77.42%                 75%
## Quantile 0.5:                  31                 50%                 50%
## Quantile 0.75:                 12              19.35%                 25%
## Quantile 0.9:                   7              11.29%                 10%
## Quantile 0.95:                  6               9.68%                  5%
```

1.4) Suppose we have a new observation (15.5, 5.5, 2, -0.55, 0.6, 65, -5, 1.2). Calculate the corresponding m = 2 factor scores and add this bivariate point to the plot in 1.3). How is it placed compared to the rest of the n = 62 points? Could you tell without computing the factor scores? Comment

We start by computing the Factor Scores for the new observation and add it in the scatterplot of the *m=2* factors retained in our model, in order to comment about the position of the scores related to the new observation:

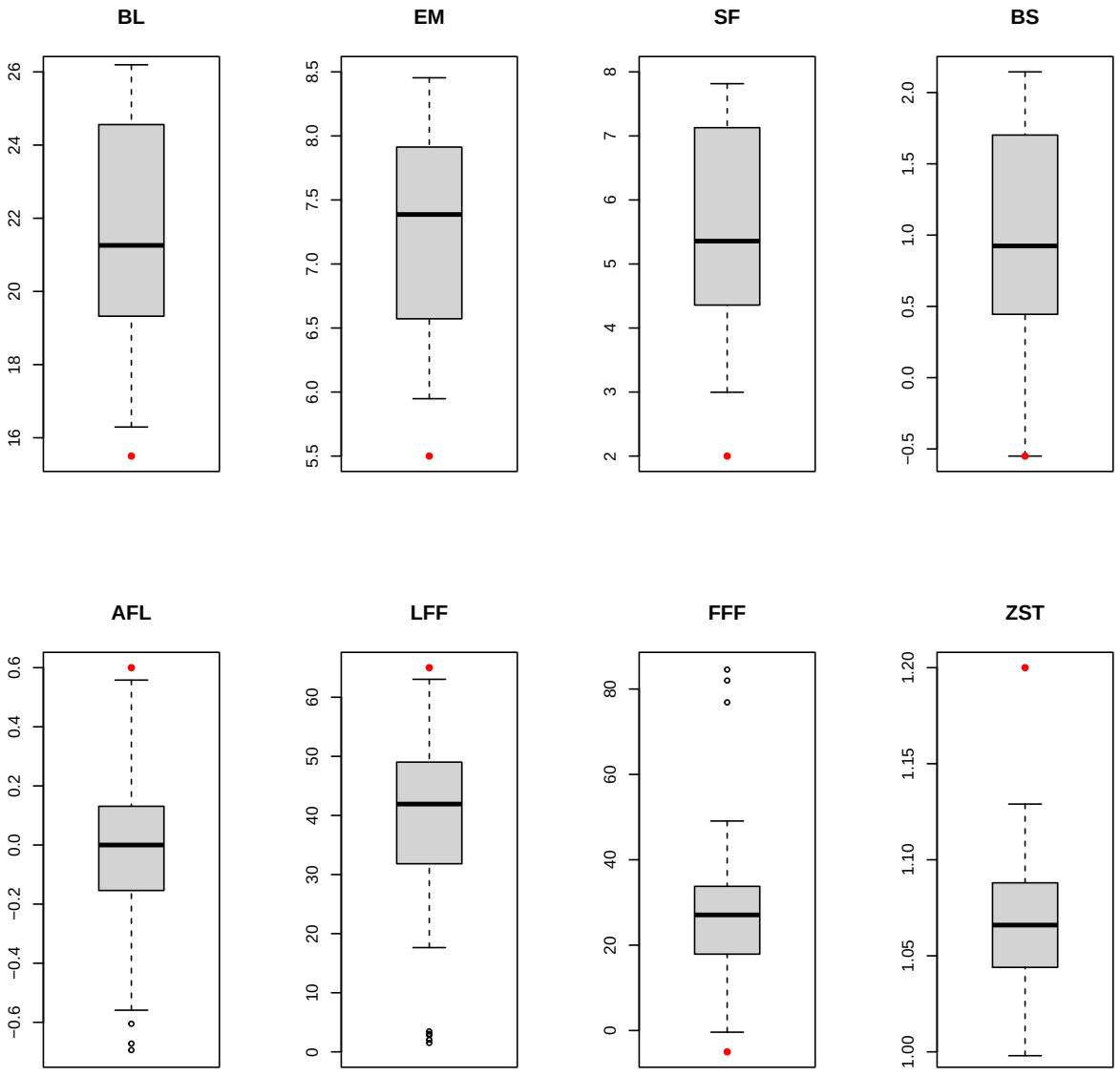


As we can clearly see from the above bivariate scatterplot, the new observation is in a strange position since it's really distant from rest of the points. In particular, we can see that it hires an high value for F_2 and a particular low value for F_1. According to the interpretation of the factors we gave before, this extreme location means that the last observation added in the dataset maybe is associated to really low values for the first 4 variables (the ones that loads highly on the first factor) and high values for the measurements that loads higly on the second factor (i. e. "AFL", "LFF", "FFF").

To notice that the factor scores observation 63 could probably have occupied a strange position in the scatterplot without computing any factor score, we can look at:

1) **Position of this last observation in the distribution of each original variable**
2) **Changes in the Correlation Structure produced by the new observation**

Univariate Boxplots
We show the boxplots of the original variables, including the last new observation to see which will be its position in the distributions of each measurement (we color it in red).



As expected, the boxplots confirm that the new observation has low values for the first 4 measurements (indeed, it's always lowe than the minimum value hired before by each of the variables), while it has all large values for the variables which form the "Group 2" and that load highly on F_2, indeed it's over the maximum for "LFF", "AFL" and "ZST". The value hired for variable "FFF" represent the new minimum for

this measurement, and this is coherent with the correlation structure seen before in the dataset, indeed, since "FFF" is negatively correlated with all the other variables related to the second factor ("AFL" and "LFF" but also "ZST"), when "AFL", "ZST" and "LFF" hire an high value, we should expect to see a low value for "FFF" (we should expect the same behavior also for the first 4 variable, but since observation 63 doesn't follow this pattern, maybe the new observation will produce some changes in the correlation structure).

Variations in Correlations
We consider again the correlations for the variables of the second group and of "ZST" (the ones related to high values for the new observation) in order to see if there're some changes.

```
## $AFL
##     Original Correlations New Correlations Variation (absolute values)
## LFF             0.9055912        0.9082403                     0.293  %
## ZST             0.7842212        0.7986777                     1.843  %
## BS              0.7063811        0.5638575                   -20.177  %
## SF              0.6807025        0.5279767                   -22.437  %
## BL              0.6477987        0.5157878                   -20.378  %
## EM              0.5370190        0.3984434                   -25.805  %
## FFF            -0.7334321       -0.7489737                     2.119  %
##
## $FFF
##     Original Correlations New Correlations Variation (absolute values)
## BL             -0.5418813       -0.4503005                   -16.901  %
## EM             -0.5559586       -0.4504287                   -18.982  %
## BS             -0.5636592       -0.4663157                    -17.27  %
## SF             -0.5745904       -0.4668162                   -18.757  %
## LFF            -0.7109855       -0.7249475                     1.964  %
## AFL            -0.7334321       -0.7489737                     2.119  %
## ZST            -0.7845570       -0.7744735                    -1.285  %
##
## $LFF
##     Original Correlations New Correlations Variation (absolute values)
## AFL             0.9055912        0.9082403                     0.293  %
## BS              0.7962528        0.6872388                   -13.307  %
## ZST             0.7927309        0.7790708                    -2.158  %
## SF              0.7644251        0.6464097                   -15.438  %
## BL              0.7350138        0.6347152                   -13.646  %
## EM              0.6085413        0.5022446                   -17.467  %
## FFF            -0.7109855       -0.7249475                     1.964  %
##
## $ZST
##     Original Correlations New Correlations Variation (absolute values)
## SF              0.8651424        0.5657407                   -31.157  %
## EM              0.8495981        0.5546071                   -30.038  %
## BL              0.8217782        0.5558002                   -31.657  %
## BS              0.8132479        0.5404548                   -31.084  %
## LFF             0.7927309        0.7790708                    -8.301  %
## AFL             0.7842212        0.7986777                    -7.683  %
## FFF            -0.7845570       -0.7744735                    -1.285  %
```

We notice that the new observation has produced important changes in the correlation structure of Data. In particular, each of the variable related to the second factor ("LFF", "AFL" and "FFF") shows a decrease in the absolute values of their correlation coefficients with the measurements related to F_1 ("BL", "BS", "EM" and "SF") while the correlation coefficients between the variable of the same group remains practically the

same. There's an important impact also in the correlations of "ZST": the coefficients with the variables of the first group suffers a remarkable decreasing in their absolute values ($\simeq -30$ for all of them) while the correlations with the variables of the second group are stable and doesn't change a lot if compared with the ones computed before the inclusion of the new observation. The behavior of the correlations for variable "ZST" after the new observation, leads us to believe that now this variable can be part of the "Group 2". This confirms that the new observation represent a very extreme point in the predictor space.

Conclusions:
To sum up, we could expect an extreme position for the new observation in the (F_1, F_2) bivariate scatterplot also without computing the factor scores. First of all, because we saw that the new observation produces important changes in the Correlation structure (so for sure we could notice that it's an extreme one), and secondly because the it shows particular low values for the first 4 variables (the ones that loads highly on F_1), determining a very little score of F_1 and high values for the last 4 (the ones that loads highly on F_2) determining a huge score of F_2. So, it's normal to expect such a position in the (F_1, F_2) scatterplot.

Exercise 2: Glass Data

Data Description

The Glass dataset is made of *n=214* observations (glass fragments) on which we've 9 measurements that we use as predictors for the target variable "type".The Variables are the following:

- **Refractive Index (RI):**
  It's an index that quantifies the overall refraction achieved by a certain material (it expresses how much light changes its direction and speed when it passes through the material).
- **Composition of the glass fragments (Na, Mg, Al, Si, K, Ca, Fe):**
  Percentages of elements that compose the fragments (where the sum of the percentages is less than 100, th remaining portion of the fragment is made of sand).
- **Type of glass (type):**
  It's the categorical target variable which has 6 classes in total (window float glass (WinF), window non float glass (WinNF), vehicle window glass (Veh), containers (Con), tableware (Tabl) and vehicle headlamps (Head)).

Data importation

We display the first rows of the dataset:

```
##        RI    Na   Mg   Al    Si    K   Ca Ba   Fe type
## 1 1.52101 13.64 4.49 1.10 71.78 0.06 8.75  0 0.00 WinF
## 2 1.51761 13.89 3.60 1.36 72.73 0.48 7.83  0 0.00 WinF
## 3 1.51618 13.53 3.55 1.54 72.99 0.39 7.78  0 0.00 WinF
## 4 1.51766 13.21 3.69 1.29 72.61 0.57 8.22  0 0.00 WinF
## 5 1.51742 13.27 3.62 1.24 73.08 0.55 8.07  0 0.00 WinF
## 6 1.51596 12.79 3.61 1.62 72.97 0.64 8.07  0 0.26 WinF
```

2.1) Use linear discriminant analysis to predict the glass type. Look at the first two discriminant directions: what are the most important variables in separating the classes? Comment

Linear Discriminant Analysis (LDA) is a classification model which can be used to determine the class k that belongs to the discrete set $\zeta = 1, ..., K$ of a categorical target variable G of an observation, basing the evaluation on the values hired by some predictor variables for that observation. The model estimates for the i-th observation, the posterior probability of beign "class k" given the vector of realization of the predictor variables $x_i = (x_{i1}, ..., x_{ip})$, i. e.:

$$P(G = k|X = x_i)$$

And then it assign the observation "i" to the class "k" (which is the predicted class $\widehat{G}(x_i)$) related to the highest posterior probability, so:

$$\widehat{G}(x_i) = \underset{k \in \zeta}{\text{argmax}}[P(G = k|X = x_i)]$$

In the contest of LDA, posterior probabilities are estimated making two important assumptions about the distributions of data:

- **Multivariate Gaussian:** We assume that each of the classes come from a Multivariate Gaussian Model $\mathcal{N}_p(\mu_k, \Sigma)$.
- **Equal Covariance Matrix:** We assume that each class has the same Covariance Matrix, i. e. $\Sigma_k = \Sigma$, for $k = 1, ..., K$.

So we fit the model making these assumptions and we use the following estimates for the quantities of interest:

1) **Prior Probabilites:** $\pi_k = \frac{n_k}{n}$, where n_k is the numerosity of the k-th class. In our data they are the followings:

```
##       WinF      WinNF        Veh        Con       Tabl       Head
## 0.32710280 0.35514019 0.07943925 0.06074766 0.04205607 0.13551402
```

2) **Centroid of class "k":** $\widehat{\mu}_k = \frac{1}{n_k} \sum_{x_i \in k} x_i$. The classes of "type" show these centroids:

```
##             RI       Na        Mg       Al       Si         K        Ca
## WinF  1.518718 13.24229 3.5524286 1.163857 72.61914 0.4474286  8.797286
## WinNF 1.518619 13.11171 3.0021053 1.408158 72.59803 0.5210526  9.073684
## Veh   1.517964 13.43706 3.5435294 1.201176 72.40471 0.4064706  8.782941
## Con   1.518928 12.82769 0.7738462 2.033846 72.36615 1.4700000 10.123846
## Tabl  1.517456 14.64667 1.3055556 1.366667 73.20667 0.0000000  9.356667
## Head  1.517116 14.44207 0.5382759 2.122759 72.96586 0.3251724  8.491379
##                Ba         Fe
## WinF  0.012714286 0.05700000
## WinNF 0.050263158 0.07973684
## Veh   0.008823529 0.05705882
## Con   0.187692308 0.06076923
## Tabl  0.000000000 0.00000000
## Head  1.040000000 0.01344828
```

3) **Pooled Sample Covariance Matrix:** $\widehat{\Sigma} = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{g_i=k} (x_i - \widehat{\mu}_k)(x_i - \widehat{\mu}_k)^T$, which is the common estimate of the Covariance Matrix. The Pooled Sample covariance Matrix of our dataset is the following:

```
##               RI            Na           Mg            Al           Si
## RI  9.092566e-06 -0.0001899992 -0.001013662 -0.0005093867 -0.001218341
## Na -1.899992e-04  0.4049642860  0.064610576 -0.0257854278 -0.137649222
## Mg -1.013662e-03  0.0646105756  0.827210011  0.0166571189 -0.063730138
## Al -5.093867e-04 -0.0257854278  0.016657119  0.1373253084 -0.028209210
## Si -1.218341e-03 -0.1376492216 -0.063730138 -0.0282092099  0.575764361
## K  -6.708453e-04 -0.0583649626  0.058363169  0.0812624925 -0.065082660
## Ca  3.482782e-03 -0.2356325239 -0.863624813 -0.2034827540 -0.206969172
## Ba  1.680743e-04 -0.0012312334 -0.038077969  0.0187271293 -0.081324610
## Fe  3.127207e-05 -0.0077332532 -0.006671334  0.0003842195 -0.003489577
##                K           Ca            Ba           Fe
## RI -0.0006708453  0.003482782  0.0001680743  3.127207e-05
## Na -0.0583649626 -0.235632524 -0.0012312334 -7.733253e-03
## Mg  0.0583631688 -0.863624813 -0.0380779690 -6.671334e-03
## Al  0.0812624925 -0.203482754  0.0187271293  3.842195e-04
## Si -0.0650826603 -0.206969172 -0.0813246103 -3.489577e-03
## K   0.3598959870 -0.380593816 -0.0008057840 -3.200061e-03
## Ca -0.3805938160  1.935782491 -0.0292816527  1.466093e-02
## Ba -0.0008057840 -0.029281653  0.1307095493  2.942978e-03
## Fe -0.0032000609  0.014660934  0.0029429784  9.127717e-03
```

Model Assumptions:

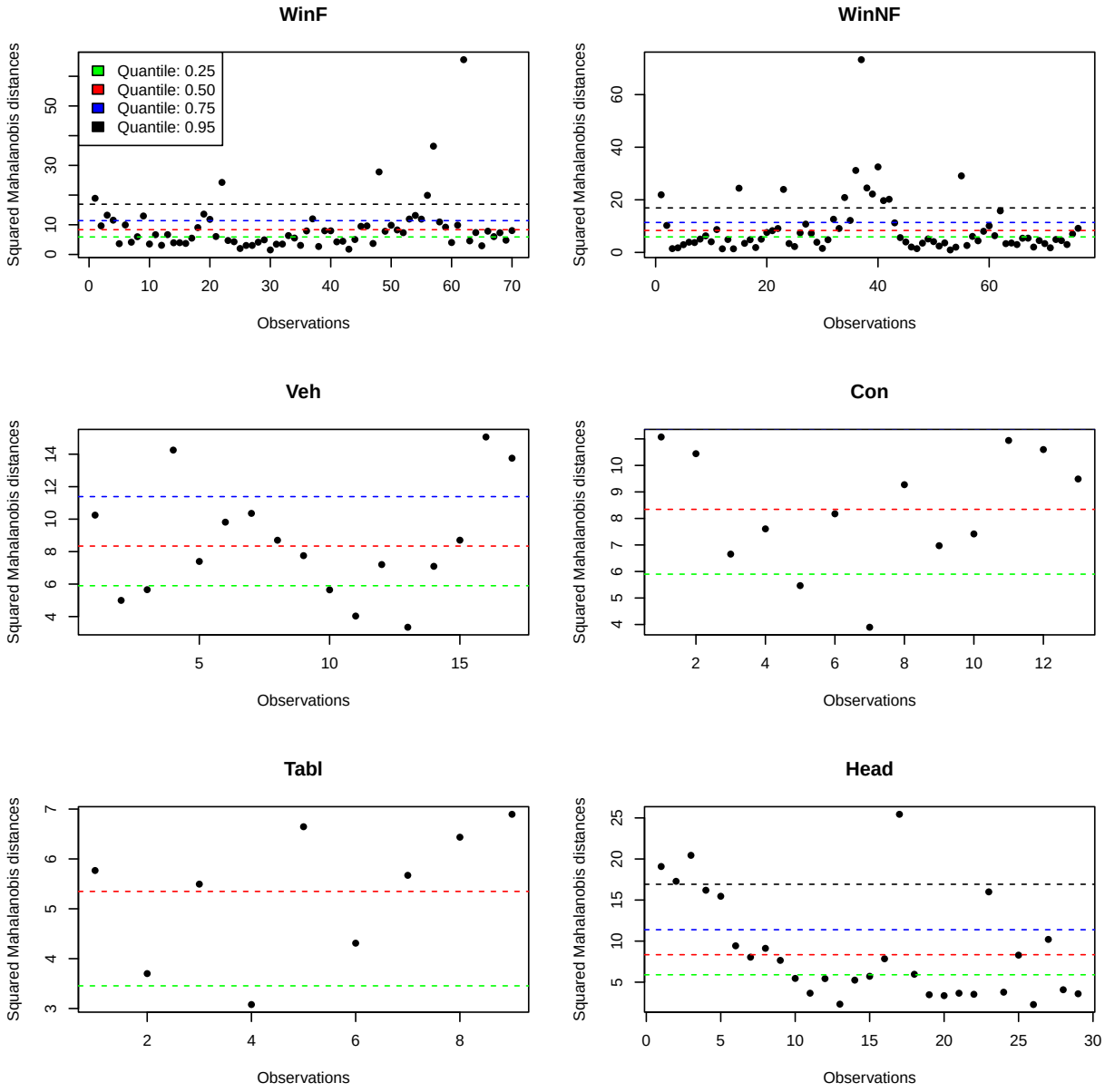
We can rapidly check if the assumptions of LDA are satisfied by our dataset.

1) *Equal Covariance Matrix:*
We compute the squared Frobenius Norms of the approximation between $\widehat{\Sigma}$ and the Sample covariance Matrices of each class Σ_k in order to see if we can consider the different classes of data to have the same Covariance structure:

```
##      WinF     WinNF       Veh       Con      Tabl      Head
##  5.483954  7.892325  6.307316 77.358133 10.339918  3.788116
```

The results show that probably we can't consider the different classes to have the same Covariance Matrix since they return quite different approximations of $\widehat{\Sigma}$ (in particular we can see that class "Tabl" has zero variance for variables "K", "Ba" and "Fe" which are elements that we don't find in that type of glass).

2) *Multivariate Normality of the K classes*
In order to evaluate if each class glass is $\mathcal{N}_p(\widehat{u}_k, \widehat{\Sigma})$ we evaluate the distribution of the observed Squared Mahalanobis Distance in order to see if they follow the χ^2_p (which means that they're Multivariate Gaussian). Here we plot the distances and we compare them with the quantiles of the χ^2_p:



We also add the results of the numerical criterion applied on the quantiles of the Squared Mahalanobis Distances:

```
##                Expected   WinF  WinNF    Veh    Con   Tabl   Head
## Quantile:  0.25     75 % 57.14% 42.11% 70.59% 84.62% 88.89% 51.72%
## Quantile:  0.5      50 % 34.29% 30.26% 47.06% 46.15% 66.67% 34.48%
## Quantile:  0.75     25 % 21.43% 19.74% 17.65%     0%     0% 24.14%
## Quantile:  0.95      5 %  8.57% 15.79%     0%     0%     0% 13.79%
```

We can conclude that (especially for some classes) we've an unsufficient number of observation to evaluate Multivariate Normality of the classes. Anyway, looking at the sample distribution of the observed Squared Mahalanobis Distances, none of the classes seems to be Multivariate Gaussian.

We're interested in finding the so called Discriminant Directions, which are vectors of coefficients $a_l = (a_{l1}, ..., a_{lp})$ for $l = 1, ..., min(K-1, p)$ to form the Discriminant Variables, which are linear combinations

($Z_l = a_l^T X$) of the original data derived estimated in order to spread as much as possible (on the 1-D projection that is the Discriminant Direction) the class centroids $\widehat{\mu}_k$, according to the within-class variance of data. Each successive discriminant variable produce a lower spread of the centroids with respect to the previous one.
So the model returns these vectors of coefficients which represent the discriminant directions and we can use their components to determine which variables contribute the more in separating classes of observations (since the discriminant directions are found as $a_l = \widehat{\Sigma}^{-1/2} e_l$, where e_l is the l-th orthonormal eigenvector of $\widehat{\Sigma}$ associated to the l-th ordered eigenvalue $\lambda_1 \geq ... \geq \lambda_l \geq ... \geq \lambda_p$).

Extraction of the Discriminant Directions

We fit the LDA model using the built-in command in R which provides the discriminant variables with spherical within-class Sample Covariance Matrix, and we display the components of the coefficients in order to make the interpretation:

```
##            LD1        LD2         LD3          LD4          LD5
## RI 311.6912516 29.3910394 356.0188308 246.85720802 -804.6553938
## Na   2.3812158  3.1650800   0.4596785   6.92435141    2.3987509
## Mg   0.7403818  2.9858720   1.5728838   6.84983896    2.8002951
## Al   3.3377416  1.7247396   2.2024668   6.41923638    0.9371345
## Si   2.4516520  3.0063507   1.7026191   7.54220302    0.9562989
## K    1.5714954  1.8620159   1.2861127   8.07611300    2.8209927
## Ca   1.0063101  2.3729126   0.6475200   6.69663574    3.7110859
## Ba   2.3140953  3.4431987   2.5964981   6.43849270    4.4077058
## Fe  -0.5114573  0.2166388   1.2026071  -0.04474935   -1.3029207
```

We notice that the coefficients have really different values, and this may be related to strong differences in the scales used for measure the predictors, so we scale the coefficients of the first two discriminant directions in order to obtain a better interpretation of their values by mitigating their differences in scale. We use the following formula: $a_l^* = diag(\widehat{\Sigma})^{1/2} a_l$.

```
##     Scaled LD1 Scaled LD2
## RI  0.93987011 0.08862539
## Na  1.51532963 2.01415576
## Mg  0.67338533 2.71568299
## Al  1.23688125 0.63914417
## Si  1.86029241 2.28119300
## K   0.94276105 1.11704814
## Ca  1.40010345 3.30149026
## Ba  0.83663281 1.24484634
## Fe -0.04886416 0.02069747
```

- **First Discriminant Direction:** The first direction (which is the one that spreads the more the class centroids) shows all positive values for the coefficients except for the one related to "Fe" (that anyway has a really low absolute value, which means that it contribution is very little). The highest value is for the coefficient of "Si" ($\simeq 1.86$), and then we find that "Na", "Al" and "Ca" have all similar values (between $\simeq 1.23$ and $\simeq 1.51$) so they've a similar and quite high importance in separating the classes. Finally, "K", Ba", "RI" and "Mg" have lower and less significant values of their coefficient with respect to the first discriminant direction.
- **Second Discriminant Direction:** The second direction present a really high value of the coefficient related to "Ca" ($\simeq 3.30$) but also "Mg", "Si" and "Na" can be consider to have a relevant weight on LD2 (their coefficients are respectively $\simeq 2.72$, $\simeq 2.28$ and $\simeq 2.02$). The other predictors produce a lower influence on this discriminant direction, especially for "RI" and "Fe" (which are again the less important variables in discriminating classes, since they've coefficients equal to $\simeq 0.088$ and $\simeq 0.021$).

To conclude, the analysis of the coefficients leads us to think that the most important variables to discriminate the classes are “Si” (which appears as relevant in both the directions), “Na”, “Ca” (since “Na” has quite remarkable values in LD1 and LD2 while “Ca” is the main variable for the second direction).
For sure we can deduce that “RI” and “Fe” are the less important predictors (since they occupy the lasts positions in both the vectors a_1 and a_2, in particular “Fe” has really little coefficients). Instead, it’s more complex to understand the relevance of “Mg” (that is non significant in LD1 but it hires importance in LD2) and also of “K”, “Al” and “Ba” (that are associated to intermediate values in both the directions).

Training Error Rate

This section is related to the first part of the point "2.2) Compute the training error. Are there any groups less homogeneous than the others?".
We'll extract the prediction for the glass "type" performed by the LDA model fitted before, then we can compute the *Training Error Rate* (defined as the fraction between the number of missclassifications in the prediction over the total number of observations).

First of all, we look at some of the posterior probabilities computed by the model for the training observations (we choose 5 observations randomly chosen):

```
##      WinF WinNF   Veh Con Tabl Head
## 49  0.754 0.145 0.101   0    0    0
## 64  0.509 0.198 0.294   0    0    0
## 32  0.659 0.295 0.046   0    0    0
## 194 0.000 0.000 0.000   0    0    1
## 92  0.150 0.731 0.120   0    0    0
```

And then we look at the classes predicted by the model:

```
## [1] WinF  WinF  WinF  Head  WinNF
## Levels: WinF WinNF Veh Con Tabl Head
```

We can rapidly confirm that the model has assigned to each observation the class which scores the highest posterior probability (since class "WinF" scores 0.754, 0.509, 0.659 for observations 64, 49 and 32 while the 194^{th} as posterior probability equal to 1 for "Head" and 92 has 0.731 for "WinNF"). Here we compute the percentage of missclassifications achieved by our model:

```
## 0.3271028
```

The model predicts correctly around the *68%* of the classes on which it has been trained. We can consider this result as quite good because it's close the *70%* of correct predictions (so it isn't totally wrong) but surely it can be improved because it missclassifies 1/3 of the observations and so there're clearly some problems that need to be fixed in order to reach a better error rate. The issues can probably derive by the fact that data don't respect perfectly the assumptions required by LDA (in particular, we couldn't conclude for sure about Multivariate Normality of data from different classes due to the low number of observations available but we showed before that maybe the assumption of equal Covariance Matrix maybe is not satisfied), but in order to have a more detailed comprehension of the reasons of this not perfect Training Error Rate, we can look at the Confusion Matrix of the Model (which is a table that compares the predicted classes with the real ones and allows us to understand better the nature of missclassifications):

```
##           true
## predicted WinF WinNF Veh Con Tabl Head
##     WinF    52    17  11   0    1    1
##     WinNF   15    54   6   5    2    2
##     Veh      3     0   0   0    0    0
##     Con      0     3   0   7    0    1
##     Tabl     0     2   0   0    6    0
##     Head     0     0   0   1    0   25
```

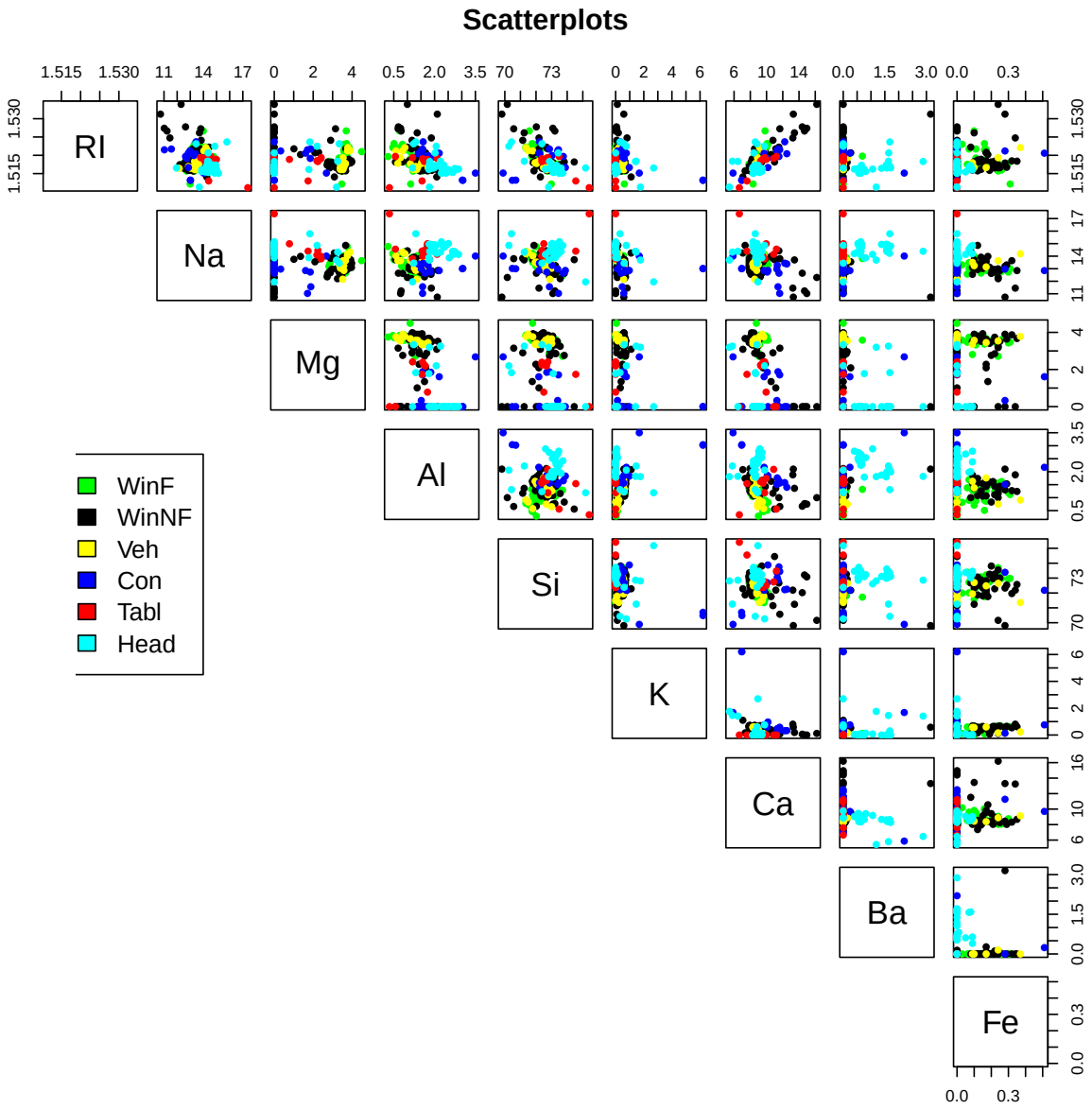
The Confusion Matrix shows that the model has a sufficient accuracy in the prediction of classes "WinF" and "WinNF" (even if for some observations the model makes confusion of these two classes) and a really good result in the prediction of the class "Head". The main issues are found in the prediction of classes "Con", "Tabl" and especially "Veh" (which achieves all the observations missclassified). The reason of this behavior may be related to the prior probabilities of the classes (which are high for "WinF" and "WinNF" and very low for "Tabl", "Veh" and "Con") or to the "nature" of these kinds of glass and we'll analyze these aspects better in the next section.

Homogeneity of the classes

In this section, we'll attempt to comment about the "homogeneity" that characterize the different groups of data labelled by the classes of the target variable. To do so, we consider both the questions required at point "2.2) Compute the training error. Are there any groups less homogeneous than the others?" and "2.4) Use the first two discriminant variables for a two-dimensional representation of the data together with centroids by using color-coding for the 6 classes of the class variable type", since they both provide useful information to try to obtain some conclusions about "homogeneity".

Personal interpretation of Homogeneity
In general, we can consider a class to be homogeneous if the observations that belong to the class are likely to hire similar values of the predictors, which means that they usually show common features that identify data which come from that class. Instead, a class can be considered as heterogeneous if data coming from it tends to present different values of the predictors, which means that they don't have common shared characteristics. Once we've given an interpretation to the concept of homogeneity, we list the elements that we'll take into account in our evaluation:

- **Preliminary 2x2 scatterplots in the predictor space:** We'll plot the position of the observations (coloring them by the class they belong) in all the possible pairs of 2x2 scatterplots with axes represented by the predictors. This will show if some classes tend to form an evident cluster clearly discernible from the others.
- **Differences in the training errors for the classes:** We'll compute the training errors performed by each of the classes in the estimated LDA model, our idea is that if a class is sufficiently homogeneous, it will be characterized by a certain uniformity of its features and so it'll be more easily predicted by the model, so it'll be related to a lower training error rate. We'll also take into account the Confusion Matrix to see more in detail the pattern of missclassifications.
- **Plot of data in the (LD1, LD2) space:** We'll compute the projection of data in the space spanned by the first two discriminant variables (the ones that spread the most the centroids with respect to the within-class variance), so that we can deal with a synthetic 2-D representation of data (but still based on the values hired by the predictors $(X_1, ..., X_p)$ for the observations) in order to see which region of the space is occupied by the classes centroids and how much the data for each class are concerned around the centroid. We can suppose that if a class show a low dispersion over the (LD1, LD2) and has observations usually projected in points which are really close to the class centroid, maybe the class is more homogeneous cause it present recurrent features with respect to the Discriminant variables (which are linear combinations of the original predictors). We'll produce an opposite conclusion if the points seems to be scattered along the plane.



The above scatterplots don't show a clear separation for the different groups of data for none of the pairs of variable, so from it is difficult to conclude if some classes present a strong homogeneity. The only particular feature we can notice is the position of the observations related to class "Head" in all the scatterplots of the variable "Ba" because they occupy a middle position (so they tends to have intermediate values of this element in their composition) which is almost not shared by non of the other glass types. This can probably suggest that observations from this class are quite homogeneous (indeed, data from "Head" are the only one that have no "K", "Ba" and "Fe" in their composition and this is a common feature shared by all the fragments of this class that distinguish them by the others also in this preliminary plot).

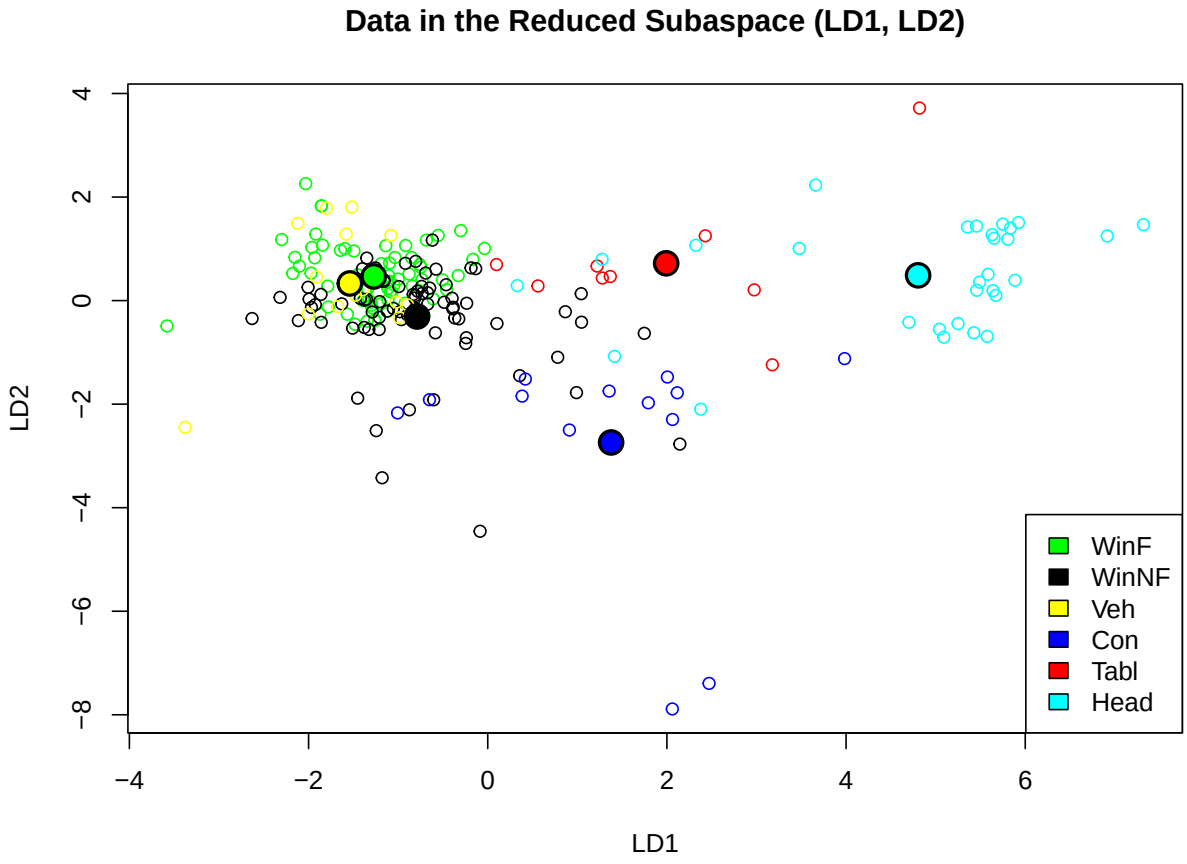
Training Errors labelled by classes

```
##       Training Error Rates
## WinF             0.2571429
## WinNF            0.2894737
## Veh              1.0000000
## Con              0.4615385
## Tabl             0.3333333
## Head             0.1379310
```

The computation of the specific error rates confirms that "Head" is probably an homogeneous variable since it's easily detected by the model (indeed its training error rate is the best one) so it's clearly distinguishable from the others. Also classes "WinF" and "WinNF" achieves a good percentage of correct predictions ($\simeq 0.26$ and $\simeq 0.29$) and this can lead us to the conclusion we did for "Head", but we need to take into account that these variables are the ones with the highest prior probabilities (respectively $\simeq 0.33$ and $\simeq 0.36$) and this can may influence the model's performance (indeed, looking at the Confusion Matrix, we detect that most of the missclassifications for the other classes are assigned to "WinF" or "WinNF"). "Tabl" has a quite limited percentage of missclassifications which means that it could have a certain level of homogeneity that helps the model to distinguish it quite well. Instead, "Con" and in especially "Veh" have a very high rate of missclassifications, this can come from the fact that they're the most heterogeneous classes of the target variable (so the model makes a big effort in recognize them) but we need to recall that they've very low values of prior probabilities ($\simeq 0.061$ and $\simeq 0.079$) and so maybe we need to evaluate other aspects.

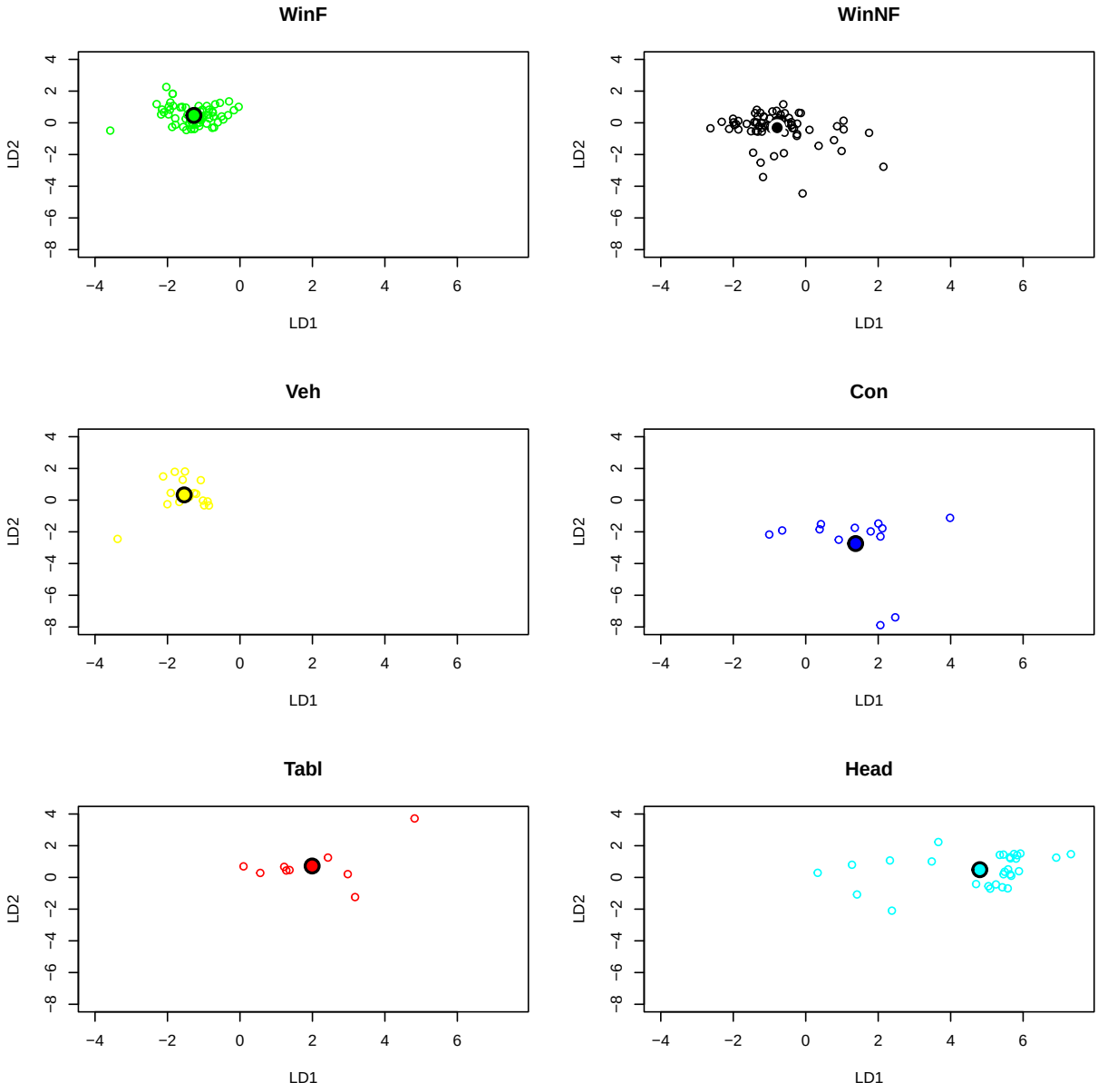
Projection in the Discriminant Variable's Space

We start by plotting together all the class centroids and the observations in the (LD1, LD2) space:



The above plot shows that some centroids are set in a region of the (LD1, LD2) space which is not shared by the other, and this means that the have a clear and distinguishable position that allows the model to produce a good prediction of these classes, while there're some centroids which are closer to each other and present some overlap for their observations (indeed, these are the classes that have the highest missclassifications between each other in the Confusion Matrix).

To understand better this aspect, we plot the same graph but with only one class every time:



From here we can see that the (LD1, LD2) projections of classes "WinF", "WinNF" and "Veh" practically share the same region of the space and this can be the reason of the high rate of missclassifications detected before for class "Veh" (indeed all its observations were assigned either to "WinF" or "WinNF"). So maybe (due too its little prior probability especially if compared to the ones of "Winf" and "WinNF") class "Veh" totally missclassified by the model not due to its heterogeneity (indeed, if we look at the cloud of points around the centroid of "Veh", we see that it's sufficiently concerned, so there aren't huge differences in the observations of this class) but because it overlaps a lot with two classes of glass that are very likely to be found and this leads

the LDA to “not see” this other category of “type”. Also “WinF” seems to have data considerably concerned around the centroid, confirming the homogeneity hypothesis for this kind of glass. Instead, “WinNF” seems to be more scattered along the plane (in particular for the LD1 variable) so we can’t say that it’s heterogeneous but maybe it achieves a good percentage of error thanks to its high prior probability rather than the uniformity of its data. Variable “Con” and “Tabl” show a remarkable dispersion of their observations especially with respect to LD1 (instead for LD2 they have some deviations from the main cloud just for 1 or 2 fragments) so they’re not really homogeneous, indeed “Con” has also a bad error rate because it shares a similar region of “WinNF” while “Table” presents a good missclassification’s score (even if it has the lowest prior probability) because its overall position in the (LD1, LD2) is clearly discernible from other centroids. Finally, we confirm once again that “Head” can be considered as an homogeneous class because the main part of its observations has an high value for the LD1 dimension which is a common and recognizable feature for this class.

Conclusions
All considered, we can attempt to summarize these considerations about homogeneity:

- **Most homogeneous classes:** We can say that "Head" and "WinF" are the most homogeneous because it arises as a distinguishable class in all the three evaluations performed above. Indeed they have the better scores for the error rates and a good concentration of points in the discriminant space.
- **Quite homogeneous classes:** Variables "Veh" and "WinNF" can be considered as sufficiently homogeneous because they both have a quite clear position in the (LD1, LD2) plot (a part for some observations) and an acceptable concentration around the centroids. Additionally, "WinNF" has also a good error rate (but due to its high prior) while "Veh" has the worst one because its observations tends to present common features, but are too much similar to "WinF" and "WinNF" to be correctly predicted. Also "Tabl" can be considered "quite homogeneous" because even if it has a remarkable dispersion along the LD1 variable, it achieves an acceptable error rate (despite its prior) because it occupies an overall region of the Discriminant Space which is quite unique and this makes its observations detectable.
- **Low homogeneous classes:** Finally, class "Con" is maybe the less homogeneous class because it obtains bad results in all the evaluations made, since it has the second worst error rate, it’s scattered along LD1 and it’s not discernible from other classes.

2.3) Implement a 10-fold cross validation using the partition of the observations provided by the variable groupCV to estimate the error rate

We implement a 10-fold Cross Validation procedure in order to obtain an estimate of the *cross validation error* in the predictions produced by the model over 10 different folds of data taken from the original dataset. So, the aim is to see how the model performs on unknown data (since we distinguish the training sets from the validation sets) and to compare it with the training errors. Finally, we'll try to understand the general performance of the model. We're going to compute the overall *cross validation error* as the mean of the 10 errors obtained for each different validation fold.

Here we report the results obtained for the 10-fold cross validation procedure both for training and validation sets:

```
##         Validation Error Rates Training Error Rates
## Fold 1               0.2666667            0.3316583
## Fold 2               0.4285714            0.3523316
## Fold 3               0.4285714            0.3172043
## Fold 4               0.4117647            0.3350254
## Fold 5               0.5882353            0.3299492
## Fold 6               0.3846154            0.3457447
## Fold 7               0.4090909            0.3229167
## Fold 8               0.1250000            0.3526316
## Fold 9               0.4230769            0.3244681
## Fold 10              0.3888889            0.3163265
## Average              0.3854482            0.3328256
## Worst                0.5882353            0.3526316
## Best                 0.1250000            0.3163265
```

Looking at the above table which notice that:

1) *Overall Error Rates:* Considering the means of the error rates obtained in the 10 different folds, we can see that the rate of missclassifications in the training set is $\simeq 0.333$ which approximately means that the model is able to discern correctly 2/3 of the observations on which it has been trained, while the average cross validation error computed is $\simeq 0.386$ that is a little big higher than the training one. Considering the error rate on the unseen observations (validation's folds), we can conclude that the model doesn't reach a terrific performance and maybe it can be improved by changing the number of Discriminant directions used to perform the classification or shifting to another model which can more adapt with to the features of the data's distributions.

2) *General behavior of the error rates:* Looking at the values of the error rates over the 10 different folds, we can see that in general the validation error rate is always higher than the training error rate (the only exception are found in the 1^{st} and in the 2^{nd} folds) but this is logical since the model has been constructed using the data structure of the training set while the validation sets are "unseen" observations on which we test the real prediction ability of the model. So, we naturally expect to see a better missclassification rate on data used for the training since in the validation set we can find some "strange observations" particularly different from the training ones that are not easy to predict correctly. Anyway, we notice that in all the folds the difference between the training rates and the validation rate is not huge (it's never higher than the 10%) and this is a positive characteristic since it means that the model is sufficiently stable and don't fit too much on data used for training.

3) *Particular Folds:* The only folds in which the model has a stranger behavior are "Fold 5" and "Fold 8", since in that validation sets we find two unusual validation error rates, respectively $\simeq 0.588$ and 0.125. So we need to notice that, even if the model seems to be quite stable since it doesn't suffers huge

variations in the values of the error rates, these two folds deviates consistently from the general behavior, leading to an optimal score for "Fold 8" and to a terrible score for "Fold 5". This is probably related to a particular structure of these two subsets of the original dataset that in one case enhance the performance of the model (Fold 8) while in the other case underline all the issues it has, so maybe these weird evidences suggest that probably the model requires to be trained on a bigger dataset in order to "learn" better how to predict some particular fragments.

2.5) Compute the training error and the 10-fold cross validation error for each reduced-rank LDA classifier. Plot both error curves against the number of discriminant directions, add full-rank LDA errors found in points 2.2) and 2.3). What classifier do you prefer?

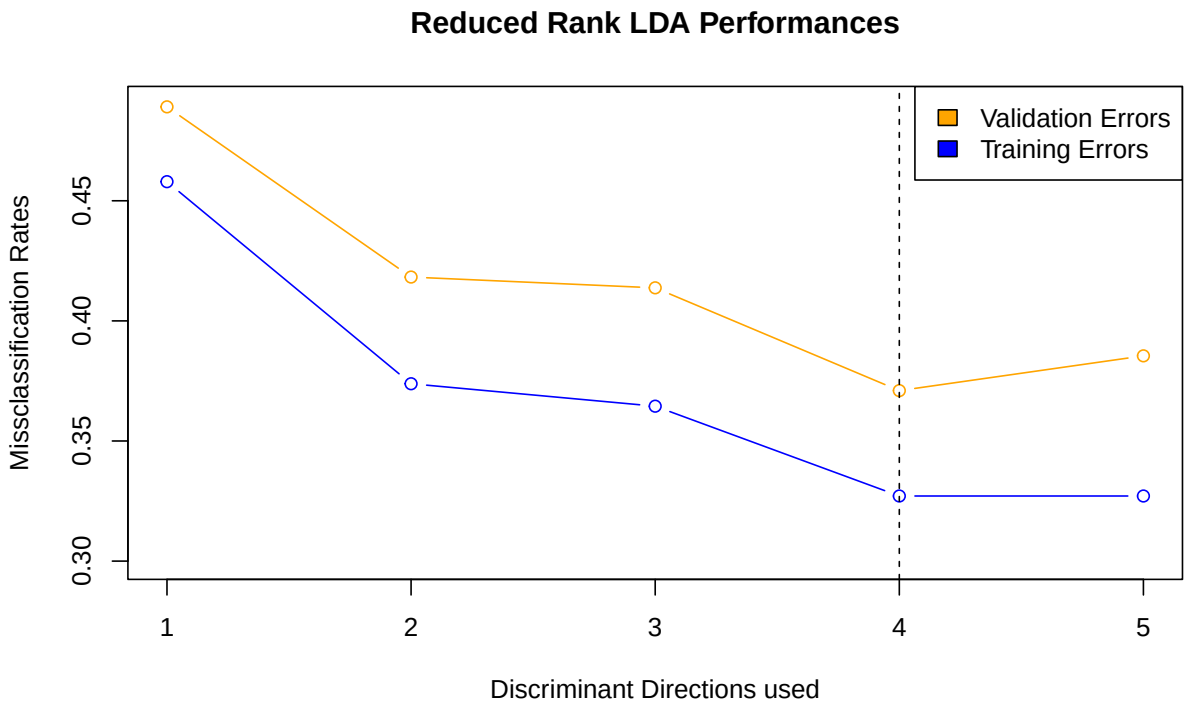
We'll fit $min(K-1, p)$ (so *K-1=5*) LDA models that will predict the classes of the observations basing on a different and progressively higher number of discriminant directions in order to see both how the model performs on a Reduced Subspace and which is the best number of dimension to choose to obtain the best error rates in training and validation sets (so for each different number of Discriminant Directions used, we'll implement a 10-fold cross validation procedure using the same folds of point 2.3)). We'll analyze our results taking into account:

- **General Performances on Training and Validation sets**
- **Error rates labelled by classes**
- **Coefficients of the Discriminant Directions**

Here we show the results of the error rates for different Reduced Rank LDA's performed:

```
##              Training Errors Validation Errors
## 1 direction        0.4579439         0.4890922
## 2 directions       0.3738318         0.4182441
## 3 directions       0.3644860         0.4137569
## 4 directions       0.3271028         0.3709695
## 5 directions       0.3271028         0.3854482
```

To visualize better the scores, we plot them:



The above plot clearly shows that the best number of directions to use to make the classification is "4", since when we predict in that subspace we minimize both the Training Error Rate (which is the same obtained at

point 2.3) in the full-Rank LDA) that decreases progressively passing from "1" to "4" dimensions and the Validation errors (that is higher if we use the full-Rank LDA). So we've found the optimal value for the parameter of the dimension that has to been used to perform the prediction.

Then we can look at the changes in the classes error rates obtained through the different Reduced Rank LDA's (these rates have been measured as the means of the 10-folds cross validations for each Reduced Rank LDA):

```
##       1 direction 2 directions 3 directions 4 directions 5 directions
## WinF        0.464        0.323        0.296        0.325        0.335
## WinNF       0.333        0.397        0.413        0.316        0.349
## Veh         1.000        1.000        1.000        1.000        1.000
## Con         0.778        0.556        0.556        0.556        0.556
## Tabl        1.000        0.750        0.583        0.500        0.500
## Head        0.233        0.233        0.211        0.211        0.211
```

First of all, we confirm that the 5^{th} direction is useless for all the classes since it gets worse the rate for the first two classes and doesn't change anything in the other. Then we notice that unfortunately class "Veh" has always all the observations missclassified, so we conclude that it doesn't have a clear distinction in none of the discriminant directions and so the model is not able to detect this kind of glass at all (this is coherent with what we pointed out at when we discuss homogeneity about this class and confirms that this category is totally "covered" by "WinF" and "WinNF"). Classes "Head" obtains always good scores and it doesn't decrease its error rate no more after the 2^{nd} discriminant direction, so this confirms that its main important distinguished position with respect to the other classes is in LD1 (as we saw in the projection of data in (LD1, LD2)). The main reasons why "4" is preferable to "3" dimensions is that the 4^{th} is able to improve consistently the error rates of "WinNF" (maybe is the dimension on which there's a better distintion between this class and "WinF") and "Tabl".

As last consideration, we can look at the scaled coefficients of the Discriminant Direction to try to deduce the reasons why the 5^{th} is not important at all:

```
##     Scaled LD1 Scaled LD2 Scaled LD3   Scaled LD4 Scaled LD5
## RI  0.93987011 0.08862539  1.0735350  0.744370299 -2.4263483
## Na  1.51532963 2.01415576  0.2925247  4.406435953  1.5264884
## Mg  0.67338533 2.71568299  1.4305549  6.230002951  2.5468988
## Al  1.23688125 0.63914417  0.8161776  2.378803999  0.3472779
## Si  1.86029241 2.28119300  1.2919327  5.722958651  0.7256313
## K   0.94276105 1.11704814  0.7715562  4.844967731  1.6923511
## Ca  1.40010345 3.30149026  0.9009102  9.317190062  5.1633229
## Ba  0.83663281 1.24484634  0.9387321  2.327758204  1.5935521
## Fe -0.04886416 0.02069747  0.1148960 -0.004275312 -0.1244799
```

We notice that the two main variables that discriminate data in LD5 are "Ca" and "Mg", which are variables that load highly practically in all the other Discriminant Directions (except for "Mg" in LD1 which is a low coefficient). This means that maybe the "discriminant information" provided by these variables has already been used by the other directions and this can be the reason why LD5 is useless in the classification.