RM3 Cookbook

Abraham Nofal

2021-12-13

Contents

Preface

This is my Advanced Research Methods cookbook written in **Markdown**.

There will be eight chapters, each covering a different multivariate data analysis technique.

Enjoy!

Chapter 1

Principal Component Analysis

1.1 Method: PCA

Principal component analysis (PCA) is used to analyze one table of quantitative data. PCA mixes the input variables to give new variables, called principal components. The first principal component is the line of best fit. It is the line that maximizes the inertia (similar to variance) of the cloud of data points. Subsequent components are defined as orthogonal to previous components, and maximize the remaining inertia.

PCA gives one map for the rows (called factor scores), and one map for the columns (called loadings). These 2 maps are related, because they both are described by the same components. However, these 2 maps project different kinds of information onto the components, and so they are *interpreted differently.* Factor scores are the coordinates of the row observations. They are interpreted by the distances between them, and their distance from the origin. Loadings describe the column variables. Loadings are interpreted by the angle between them, and their distance from the origin.

The distance from the origin is important in both maps, because squared distance from the mean is inertia (variance, information; see sum of squares as in ANOVA/regression). Because of the Pythagorean Theorem, the total information contributed by a data point (its squared distance to the origin) is also equal to the sum of its squared factor scores.

1.2 The Data Set

Eight sensory panelists evaluated six wheat beers using 42 sensory attributes Sensory attributes were evaluated using a 10-points intensity scale (imported from excel).

```
df_beers <- import("Wheat beer no alcohol.xlsx")
dm_beers <- data.matrix(df_beers[1:48,3:44])
head(dm_beers) %>% kable()
```

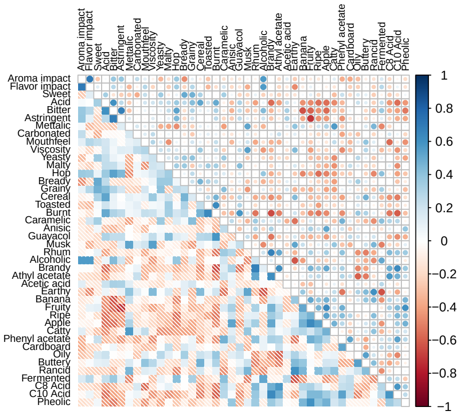
Aroma impact	Flavor impact	Sweet	Acid	Bitter	Astringent	Mettalic	Carbonated	Mou
6.8	7.0	0.2	0.4	5.3	4.8	0.2	1.2	
1.0	1.0	2.0	2.1	0.9	1.0	1.0	0.5	
6.0	9.0	7.0	8.0	8.0	6.0	1.0	1.0	
7.1	8.1	4.0	3.0	5.7	5.0	7.5	6.5	
4.5	5.0	4.1	4.2	5.6	5.4	5.7	5.5	
7.0	8.4	3.0	5.0	7.5	6.0	1.5	7.0	

```
dm_beers <- t(dm_beers)
dm_beers <- scale(dm_beers, center = TRUE, scale = FALSE)
#The data was centered based on columns instead of the rows.
dm_beers <- t(dm_beers)
```

1.3 The correlation plot

Here is the correlation plot for the data. Most sensory attributes are negatively correlated with each other, but some are strongly correlated, such as flavor impact and aroma impact, bitter and astringent, sweet and bready/grainy, alcoholic and aroma impact/flavor impact, the acidic attributes with each other and fruity, etc. These positive correlations make sense and show that we've most likely performed the correlation analysis correctly.

```
cor.res <- cor(dm_beers)
corrplot.mixed(cor.res, lower = 'shade', tl.pos = 'lt', tl.cex = 0.7, tl.col = "black"
```



1.4 Analysis

Set the seed so that your analysis is reproducible.

```
set.seed(42)
```

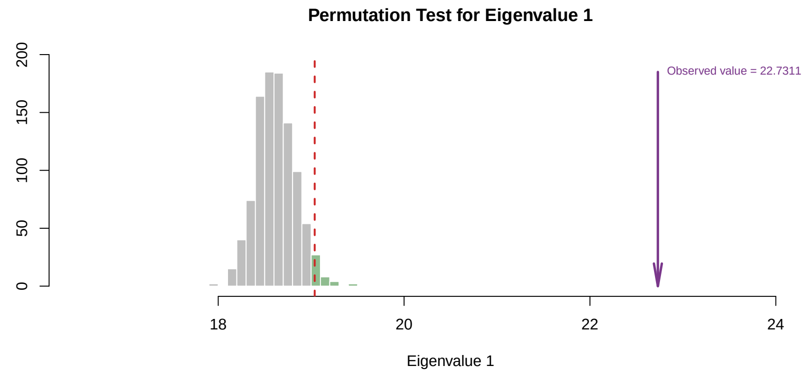
Run PCA with inference.

```
res_pcaInf <- epPCA.inference.battery(dm_beers, center = FALSE, scale = "SS1", graphs = FALSE,
                                      test.iters = 999)
```

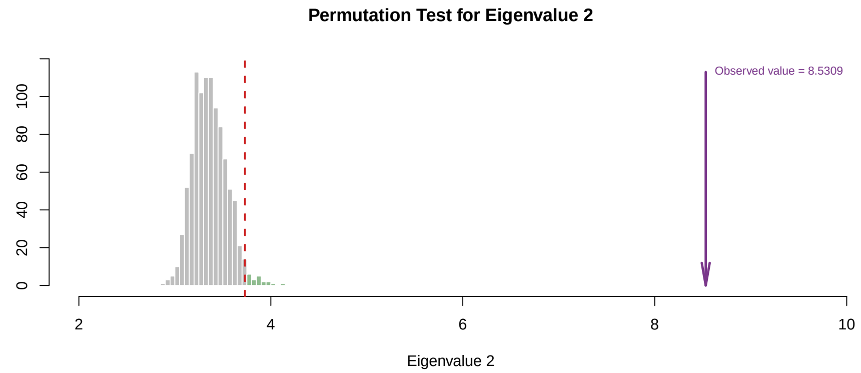
1.4.1 Testing the eigenvalues

Here we test the eigenvalues from the permutation test to see if they fall above the 97.5% cutoff. The results from the permutation test are shown in the histogram. We tested the eigenvalues for dimension one and dimension two and both passed the test and surpassed the 97.5% cutoff.

```
zeDim = 1
pH1 <- prettyHist(
  distribution = res_pcaInf$Inference.Data$components$eigs.perm[,zeDim],
           observed = res_pcaInf$Fixed.Data$ExPosition.Data$eigs[zeDim],
           xlim = c(16.5, 24.5), # needs to be set by hand
           breaks = 20,
           border = "white",
           main = paste0("Permutation Test for Eigenvalue ",zeDim),
           xlab = paste0("Eigenvalue ",zeDim),
           ylab = "",
            counts = FALSE,
            cutoffs = c( 0.975))
```



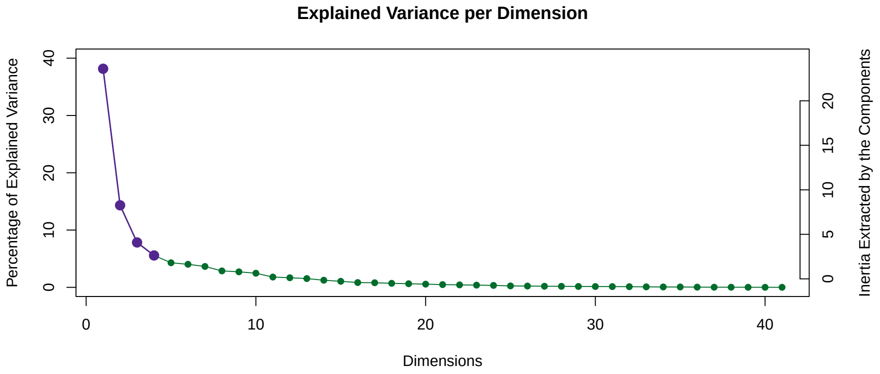
```
zeDim = 2
pH2 <- prettyHist(
  distribution = res_pcaInf$Inference.Data$components$eigs.perm[,zeDim],
            observed = res_pcaInf$Fixed.Data$ExPosition.Data$eigs[zeDim],
            xlim = c(2, 9.75), # needs to be set by hand
            breaks = 20,
            border = "white",
            main = paste0("Permutation Test for Eigenvalue ",zeDim),
            xlab = paste0("Eigenvalue ",zeDim),
            ylab = "",
            counts = FALSE,
            cutoffs = c(0.975))
```



1.4.2 Scree Plot

The results from the permutation with Scree plot (significant components colored) are shown here. We see that the first four dimensions have significant p values. Dimension one explains nearly 40% of the variance in our data, and dimension two nearly 15%. We will focus on analyzing the first two dimensions in this analysis.

```
my.scree.pca <- PlotScree(ev = res_pcaInf$Fixed.Data$ExPosition.Data$eigs,
                          p.ev = res_pcaInf$Inference.Data$components$p.vals)
```

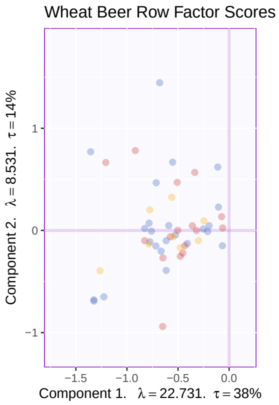


1.4.3 Factor scores

The factor scores on components one and two are plotted here. We see that the row factor scores do not exhibit any obvious clusters based on beer type since we centered the rows of the data to eliminate the effects of the rows (otherwise we would have two effects, one from the panelists and one from the beers). We will focus on the column factor scores for this PCA. Centering the columns can also be done in a different PCA in order to examine the effects of the rows, however, the variances will likely conflict from our two independent variables. The factor scores are colored according to their beer type. The blue beers are alcoholic, the red nonalcoholic, and the yellow the regular Mx product.

```
col4Beers <- c(res_pcaInf$Fixed.Data$Plotting.Data$fi.col[1:16],recode(res_pcaInf$Fixed.Data$Plot
my.fi.plot <- createFactorMap(res_pcaInf$Fixed.Data$ExPosition.Data$fi, # data
                              title = "Wheat Beer Row Factor Scores", # title of the plot
                              axis1 = 1, axis2 = 2, # which component for x and y axes
                              pch = 19, # the shape of the dots (google `pch`)
                              cex = 2, # the size of the dots
                              text.cex = 2.5, # the size of the text
```

```
                               alpha.points = 0.3,
                               col.points = col4Beers, # color of the dots
                               col.labels = col4Beers, display.labels = FALSE # color for
                               )

fi.labels <- createxyLabels.gen(1,2,
                                lambda = res_pcaInf$Fixed.Data$ExPosition.Data$eigs,
                                tau = round(res_pcaInf$Fixed.Data$ExPosition.Data$t),
                                axisName = "Component "
                                )
fi.plot <- my.fi.plot$zeMap + fi.labels # you need this line to be able to save them i
fi.plot
```



Here we get the color for each group:

```
# get index for the first row of each group
beer_type <- df_beers[,2]
grp.ind <- order(beer_type)[!duplicated(sort(beer_type))]
grp.col <- col4Beers[grp.ind] # get the color
grp.name <- beer_type[grp.ind] # get the corresponding groups
names(grp.col) <- grp.name
```

1.4.3.1 With group means

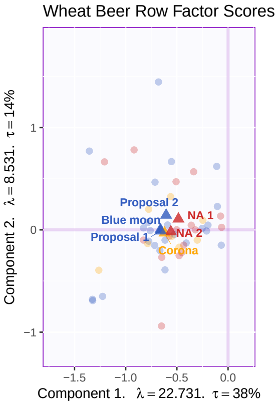
Get the group means from the PCA.

```
group.mean <- aggregate(res_pcaInf$Fixed.Data$ExPosition.Data$fi,
                       by = list(beer_type), # must be a list
                       mean)
group.mean
```

```
# need to format the results from `aggregate` correctly
rownames(group.mean) <- group.mean[,1] # Use the first column as row names
fi.mean <- group.mean[,-1] # Exclude the first column
fi.mean
```

Plot them!

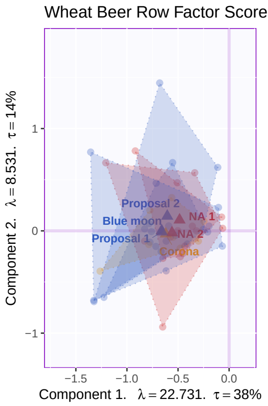
```
fi.mean.plot <- createFactorMap(fi.mean,
                                alpha.points = 0.8,
                                col.points = grp.col[rownames(fi.mean)],
                                col.labels = grp.col[rownames(fi.mean)],
                                pch = 17,
                                cex = 3,
                                text.cex = 3)
fi.WithMean <- my.fi.plot$zeMap_background + my.fi.plot$zeMap_dots +
  fi.mean.plot$zeMap_dots + fi.mean.plot$zeMap_text + fi.labels
fi.WithMean
```



1.4.4 Tolerance interval

We can plot the tolerance intervals for the beers, we see they are mostly overlapping due to the centering of the rows of the data.

```
TIplot <- MakeToleranceIntervals(res_pcaInf$Fixed.Data$ExPosition.Data$fi,
                            design = as.factor(beer_type),
                            col = grp.col[rownames(fi.mean)],
                            line.size = .50,
                            line.type = 3,
                            alpha.ellipse = .2,
                                   alpha.line    = .4,
                                   p.level       = .95)
# If you get some errors with this function, check the names.of.factors argument in th

fi.WithMeanTI <- my.fi.plot$zeMap_background + my.fi.plot$zeMap_dots +
  fi.mean.plot$zeMap_dots + fi.mean.plot$zeMap_text + TIplot + fi.labels

fi.WithMeanTI
```



1.4.5 Bootstrap interval

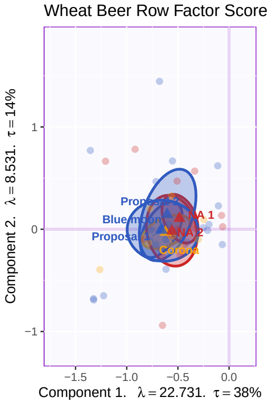
We can also add the bootstrap interval for the group means to see if these group means are significantly different. The bootstrap intervals are mostly overlapping in this case due to the centering of the data based on rows.

First step: bootstrap the group means

```
# Depend on the size of your data, this might take a while
fi.boot <- Boot4Mean(res_pcaInf$Fixed.Data$ExPosition.Data$fi,
                     design = beer_type,
                     niter = 1000)
# Check what you have
fi.boot

# What is the cube? Check the first 4 tables. You don't need to include this in
# your output, because it's a lot of junk text.
fi.boot$BootCube[,,1:4]
```

Second step: plot it!

```
# Check other parameters you can change for this function
bootCI4mean <- MakeCIEllipses(fi.boot$BootCube[,c(1:2),], # get the first two components
                              col = grp.col[rownames(fi.mean)])

fi.WithMeanCI <- my.fi.plot$zeMap_background + bootCI4mean +
  my.fi.plot$zeMap_dots + fi.mean.plot$zeMap_dots +
  fi.mean.plot$zeMap_text + fi.labels

fi.WithMeanCI
```

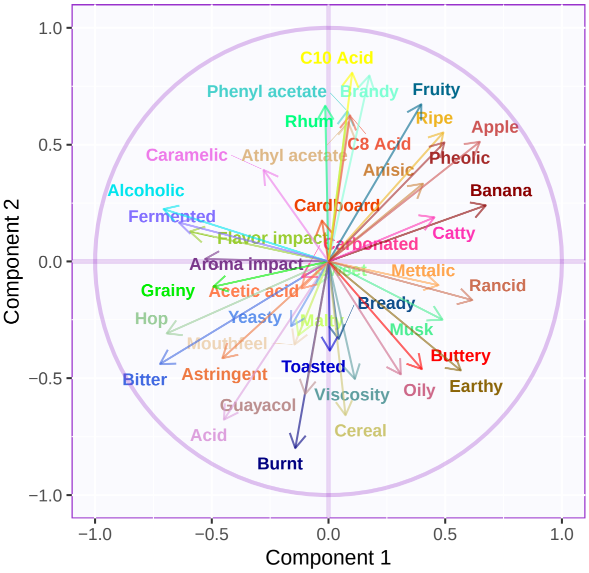


1.4.6 Loadings

Here we show the column factor scores, which is where most of our interpretation of the PCA will take place. Alcoholic, fermented, flavor impact, and aroma impact make strong negative contributions on component one, while Apple/Banana, Buttery/Oily, and Earthy/Musk make strong positive contributions on componenet one. These interpretations are repeated below the graph.

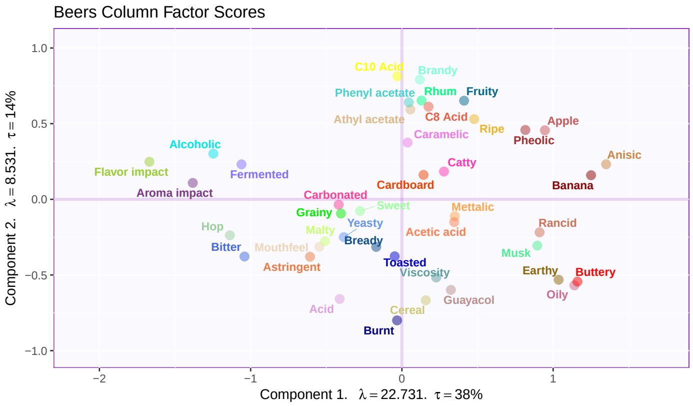
```
cor.loading <- cor(dm_beers, res_pcaInf$Fixed.Data$ExPosition.Data$fi)
rownames(cor.loading) <- rownames(cor.loading)
col4points <- prettyGraphsColorSelection(n.colors = 42)
loading.plot <- createFactorMap(cor.loading,
                                constraints = list(minx = -1.0, miny = -1.0,
                                                   maxx = 1.0, maxy = 1.0),
                                col.points = col4points, alpha.points = 0,
                                col.labels = col4points, text.cex = 3)
LoadingMapWithCircles <- loading.plot$zeMap +
  addArrows(cor.loading, color = col4points, size = 0.5, alpha = 0.6) +
  addCircleOfCor() + xlab("Component 1") + ylab("Component 2")
```

```
LoadingMapWithCircles
```



You can also include the variance of each component and plot the factor scores for the columns (i.e., the variables):

```
my.fj.plot <- createFactorMap(res_pcaInf$Fixed.Data$ExPosition.Data$fj, # data
                            title = "Beers Column Factor Scores", # title of the plot
                            axis1 = 1, axis2 = 2, # which component for x and y axes
                            pch = 19, # the shape of the dots (google `pch`)
                            cex = 3, # the size of the dots
                            text.cex = 3, # the size of the text
                            col.points = col4points, # color of the dots
                            col.labels = col4points, # color for labels of dots
                            )

fj.plot <- my.fj.plot$zeMap + fi.labels # you need this line to be able to save them i
fj.plot
```



- Component 1: Flavor/Aroma impact, Alcoholic, Fermented Vs. Apple/Banana, Buttery/Oily, Earthy/Musk
- Component 2: Burnt,Cereal,Acid.Toasted Vs. C10 acid, Brandy/Rhum, Athyl Acetate

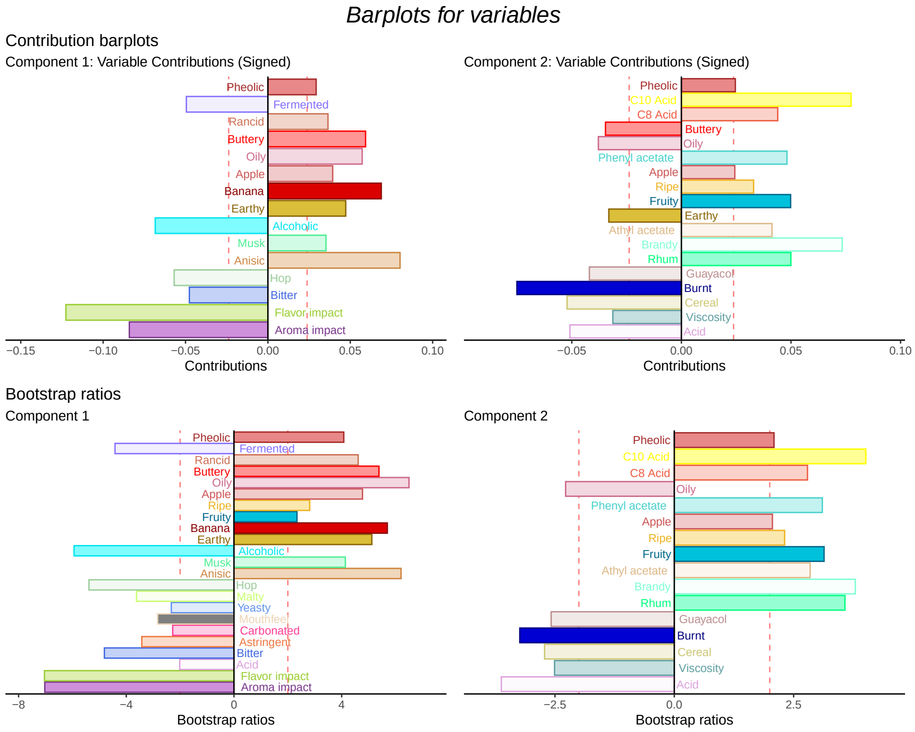
1.4.6.1 Bootstrap Ratio of columns

1.4.6.1.1 Component 1 Here the boot strap ratio is applied to the contributions for both component 1 and 2. Our interpretations from the column factor scores are confirmed here by the significant bootstrap ratios for the sensory attributes mentioned. Findings are summarized in the next section of this chapter.

```
signed.ctrJ <- res_pcaInf$Fixed.Data$ExPosition.Data$cj * sign(res_pcaInf$Fixed.Data$ExPosition.D

# plot contributions for component 1
ctrJ.1 <- PrettyBarPlot2(signed.ctrJ[,1],
                         threshold = 1 / NROW(signed.ctrJ),
                         font.size = 3,
                         color4bar = gplots::col2hex(col4points), # we need hex code
                         ylab = 'Contributions',
                         ylim = c(1.2*min(signed.ctrJ[,1]), 1.2*max(signed.ctrJ[,1])),
                         horizontal = FALSE, signifOnly = TRUE
) + ggtitle("Contribution barplots", subtitle = 'Component 1: Variable Contributions (Signed)')

# plot contributions for component 2
ctrJ.2 <- PrettyBarPlot2(signed.ctrJ[,2],
                         threshold = 1 / NROW(signed.ctrJ),
                         font.size = 3,
                         color4bar = gplots::col2hex(col4points), # we need hex code
                         ylab = 'Contributions',
                         ylim = c(1.2*min(signed.ctrJ[,2]), 1.2*max(signed.ctrJ[,2])),
                         horizontal = FALSE, signifOnly = TRUE
) + ggtitle("",subtitle = 'Component 2: Variable Contributions (Signed)')


BR <- res_pcaInf$Inference.Data$fj.boots$tests$boot.ratios
laDim = 1

# Plot the bootstrap ratios for Dimension 1
ba001.BR1 <- PrettyBarPlot2(BR[,laDim],
                        threshold = 2,
                        font.size = 3,
                   color4bar = gplots::col2hex(col4points), # we need hex code
                  ylab = 'Bootstrap ratios',
                  horizontal = FALSE, signifOnly = TRUE
                  #ylim = c(1.2*min(BR[,laDim]), 1.2*max(BR[,laDim]))
) + ggtitle("Bootstrap ratios", subtitle = paste0('Component ', laDim))

# Plot the bootstrap ratios for Dimension 2
laDim = 2
ba002.BR2 <- PrettyBarPlot2(BR[,laDim],
                        threshold = 2,
                        font.size = 3,
                   color4bar = gplots::col2hex(col4points), # we need hex code
                  ylab = 'Bootstrap ratios',
                  horizontal = FALSE, signifOnly = TRUE
                  #ylim = c(1.2*min(BR[,laDim]), 1.2*max(BR[,laDim]))
) + ggtitle("",subtitle = paste0('Component ', laDim))
```

We then use the next line of code to put two figures side to side:

```
  grid.arrange(
    as.grob(ctrJ.1),
    as.grob(ctrJ.2),
    as.grob(ba001.BR1),
    as.grob(ba002.BR2),
    ncol = 2,nrow = 2,
    top = textGrob("Barplots for variables", gp = gpar(fontsize = 18, font = 3))
  )
```



1.5 Summary

When we interpret the factor scores and loadings together, the PCA revealed:

- Component 1: Alcoholic Beers were astringent, bitter, high flavor and aroma impact and Non-Alcoholic Beers were fruity, buttery, oily, earthy, and had high musk
- Component 2: Wheat Beers were fruity, pheolic, high in c10 acid, Rhum, and Brandy content and Non-Wheat Beers had burnt, cereal, high viscosity and high acid attributes.
- Both: Non-alcoholic wheat beers are particularly fruity.

Chapter 2

Correspondence Analysis

2.1 Introduction

Correspondence analysis (CA) turns a data table into two new variables called factor scores (one factor score is a linear combination of the rows, and the other the columns). The factor scores give the best representation of the similarity in structure of the rows and columns. Correspondence analysis maximizes the variance of the factor scores.

source: Abdi, H., & Béra, M. (2014). Correspondence Analysis.

2.2 The data and pattern

Six vowels (i,y,e,a,o, and u) were shown to a set of participants and asked to associate a color (Yellow, Green, Orange, Blue, Red, and Violet) to each of them.

```
# Get the data ----
X <- import("FrenchVowelsAndColors.csv")

# The active data set
X <- as.matrix(X)
row.names(X) <- c('i','y','e','a','o','u')
X
```

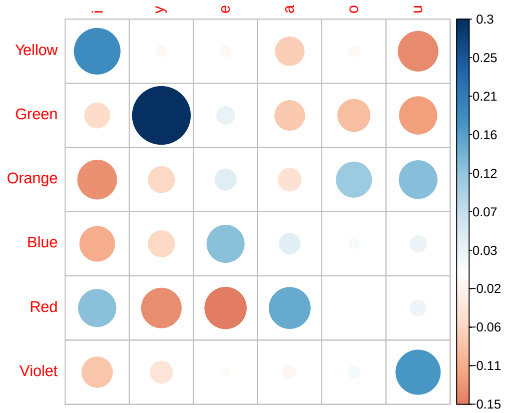
```
##   Yellow Green Orange Blue Red Violet
## i     46    17      2   11  42      5
## y      18     48      6    12   6       6
## e      17     20     13    29   4       8
## a       8      7      5    17  30       6
## o      18      9     19    19  21      10
## u       1      2     15    14  16      16
```

Computing the chi-square of the data matrix:

```
# get Chi2 -- we can use the available package to get the Chi2
chi2    <-  chisq.test(X)
```

Chi-square is in counts, but CA analyzed probabilities (i.e., the profiles). So, we need to divide the chi-square statistics by the total sum of the data. Also, the chi-square statistic adds the chi-squares in all cells and give one number. In CA, however, we keep the pattern of chi-squares instead of adding all of them up.

```
# Components of chi2: the chi-squares for each cell before we add them up to compute th
Inertia.cells <- chi2$residuals / sqrt(sum(X))
# To be Plotted

# You can always compute it directly from the data
Z <- X / sum(X) # observed
r <- as.matrix(rowSums(Z)) # expected for each row
c <- as.matrix(colSums(Z)) # expected for each column
# Inertia.cells
test.Inertia.cells <- diag( as.vector(r^(-1/2)) ) %*%
                          (Z - r%*%t(c) ) %*% diag(as.vector(c^(-1/2)))
```



Plotting the residual:

2.3 Analysis

Computing Symmetric and Asymmetric CA analysis. epCA function is used with property ‘symmetric =’ true or false respectively.

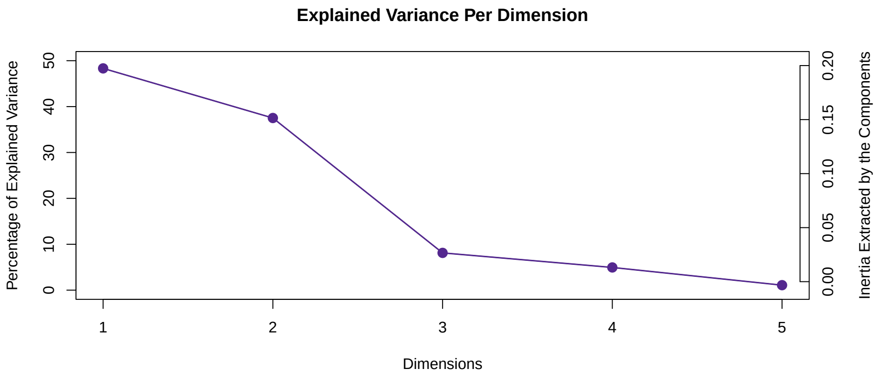
```
# run CA
resCA.sym  <- epCA(X, symmetric = TRUE, graphs = FALSE)
resCAinf.sym4bootJ  <- epCA.inference.battery(X, symmetric = TRUE, graphs = FALSE, test.iters = 2
resCAinf.sym4bootI  <- epCA.inference.battery(t(X), symmetric = TRUE, graphs = FALSE, test.iters

# to run a plain CA but asymmetric
# this is using the columns as the simplex (you can also use rows by running epCA with t(X))
resCA.asym <- epCA(X, symmetric = FALSE, graphs = FALSE)
```

Here the inference battery is produced using function fastperm4CA, with 1000 iterations for the permutations:

```
res_fast_perm <- data4PCCAR::fastPerm4CA(X, nIter = 1000, compact = FALSE)
res_fast_boot <- data4PCCAR::fastBoot4CA(X)
```

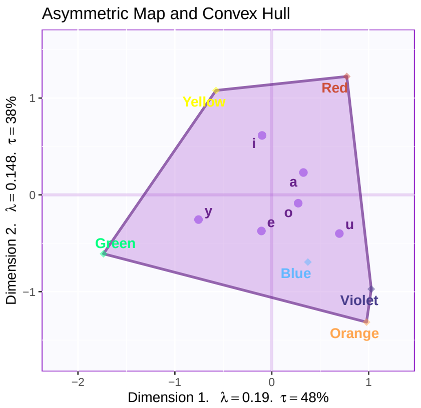
2.3.1 Scree Plot

The results from permutation with Scree plot are shown here. The estimated p-values were added to the `PlotScree` function.



2.3.2 Plot the asymmetric factor scores

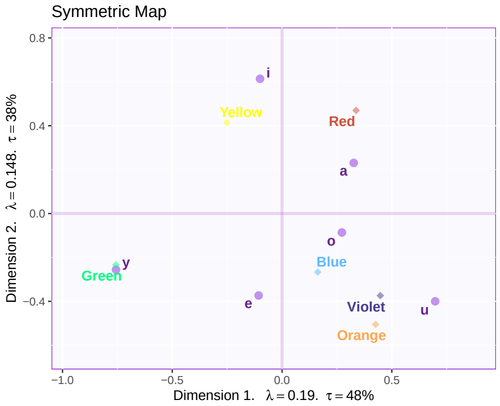
Here the asymmetric plot is shown inside the convex hull for the values of the columns. Further interpretation of these results are discussed in the conclusion section of this chapter, however, we are able to interpret which letters are most attributed to which colors from this map.



2.3.3 Plot the symmetric plot

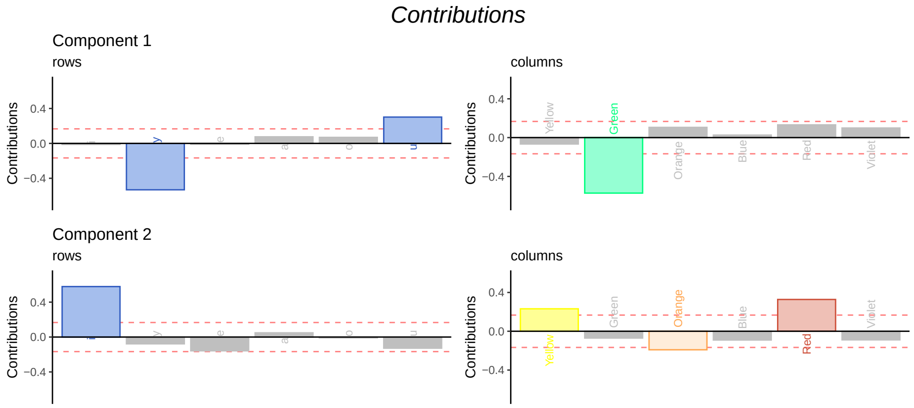
Creating the symmetric plot with all labels printed. In this case, the symmetric map and asymmetric map are quite similar, and our interpretations of the relationship between colors and letters remains the same.

2.3.3.1 Biplot:



2.3.3.2 Contributions and bootstrap ratios barplots

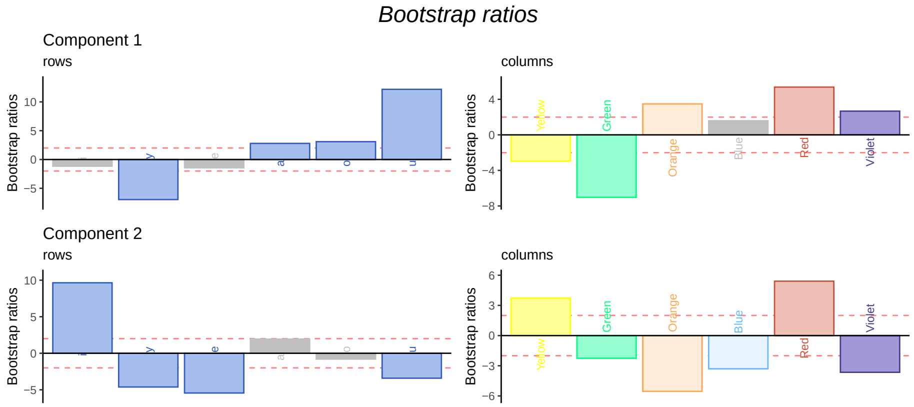
2.3.3.2.1 Contribution barplots Contributions for rows and columns are shown in this section. For component one row contributions, y has an important negative contribution and u has an important positive contribution. For the columns, green has an important negative contribution. For component two row contributions, i has an important negative contribution. For the columns, yellow and red have important positive contributions and green has an important negative contribution.



2.3.3.2.2 Bootstrap ratios

The bootstrap ratio plot showed many more significant contributions as shown. We can now infer that the y contribution is related to the green contribution on component one, and a, o, and u related to orange, red, and violet. For component two, i is related to yellow and red while y,e, and u are related to green, orange, blue, and violet. The contributions along with the scores plot (either symmetric or asymmetric) allow us to make some inferences of which letters were most attributed to which colors.

We then use the next line of code to put two figures side to side:



2.4 Summary

When we interpret the factor scores and loadings together, the CA revealed:

- Do you prefer symmetric or asymmetric plot for your data? I prefer the symmetric plot since it better shows the relationship between the variables.
- Component 1: Strong negative loadings: row: y column: Green Strong positive loadings: row: u column: violet, orange, red
- Component 2: Strong negative loadings: row: e column: blue, green, violet, orange Strong positive loadings: row: i column: yellow, red
- Both: The colors orange, blue, and violet mostly described the letters a,o,u, and e. The colors green, yellow, and red mostly described the letters i and y. —

Chapter 3

Multiple Correspondence Analysis

3.1 Introduction

Multiple correspondence analysis (MCA) extends correspondence analysis (CA). It allows for the analysis of the pattern of relationships of more than one categorical dependent variable. It can also be seen as a generalization of principal component analysis when when the variables analyzed are categorical instead of quantitative. MCA is obtained by applying correspondence analysis to an indicator matrix.

source: Abdi, H., & Valentin, D. (2007). Multiple correspondence analysis. Encyclopedia of measurement and statistics, 2(4), 651-657.

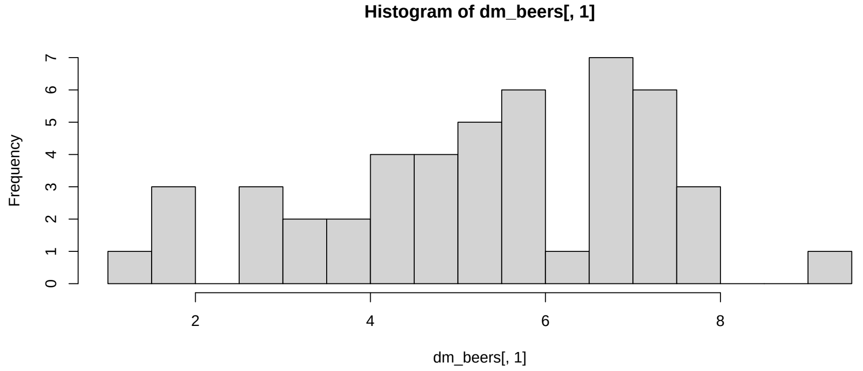
MCA is applied in this chapter to a set of quantitative data, transformed into categorical by recoding each variable into bins (a range of scores goes into different bins; low, medium, or high scores).

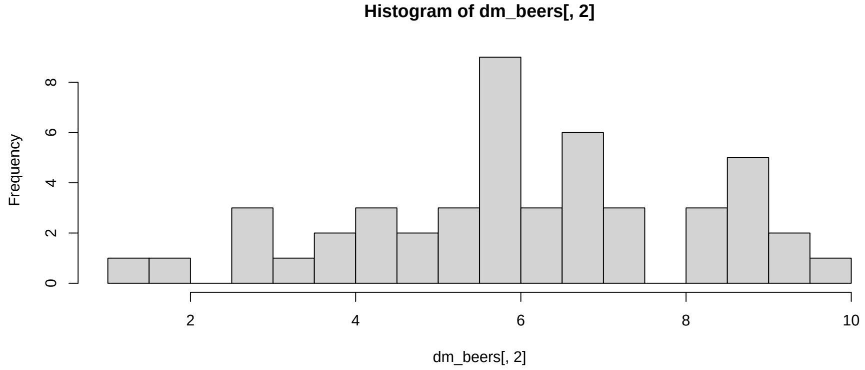
3.2 The Data Set

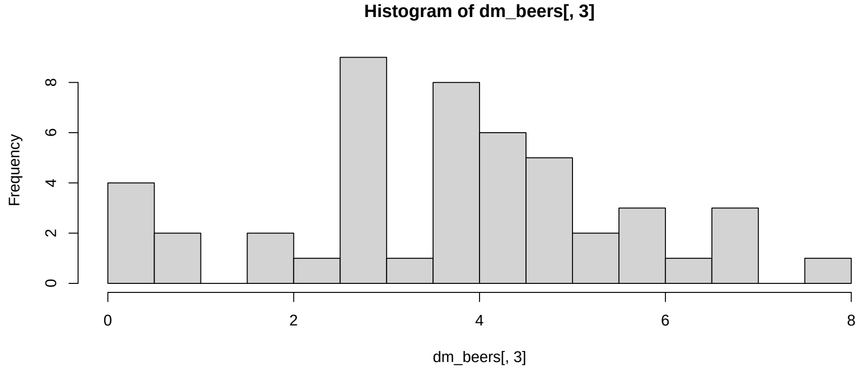
Data set to be analyzed by MCA is the same as in the PCA chapter. The first 6 rows represent panelists 1 - 6 grading of beer "Proposal 1." There are 8 panelists grading 6 different beers according to the sensory attributes below. 2 of the beers were Non-alcoholic and 4 alcoholic. 5 of the beers were wheat based and one alcoholic beer was non-wheat based.

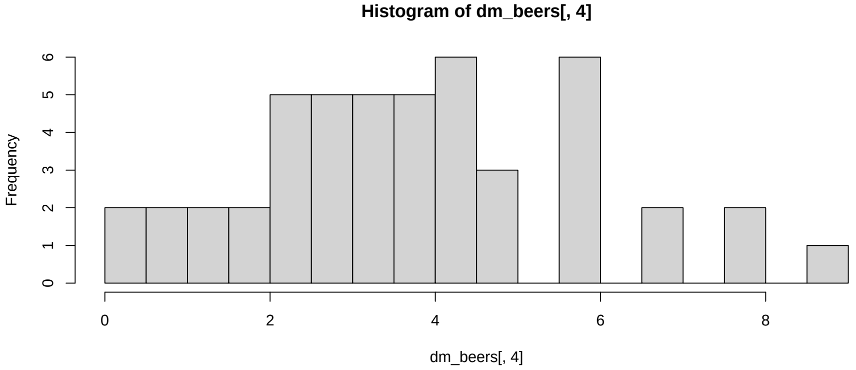
3.3 Binning the Data

Here are the histograms of the first 4 sensory attributes. Based on their roughly normal distributions and the number of observations (48 for each sensory attribute), I decided to cut the data into 3 bins.







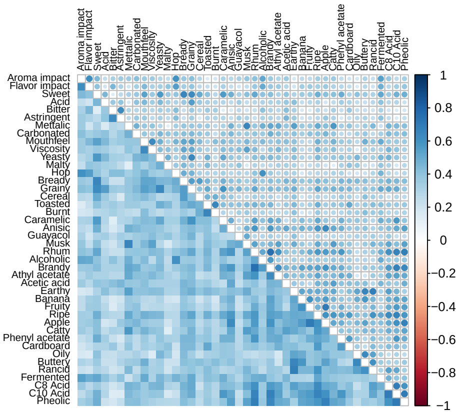


A for loop was used to iterate through each of 42 quantitative variables and bin them accordingly:

```
mesBeerRecoded <- data.frame(
                  row.names = rownames(dm_beers))
irec = c(1:42)
for(val in irec)
{
  mesBeerRecoded[,colnames(dm_beers)[irec]] <- BinQuant(
               dm_beers[,irec], nClass = 3, stem = '')
}
```

3.4 Phi Correlation

The heat map for the phi correlation is shown here. This was achieved by taking the square root of the phi squared. We see very similar correlations as compared to the PCA run in chapter 2.



3.5 MCA Code

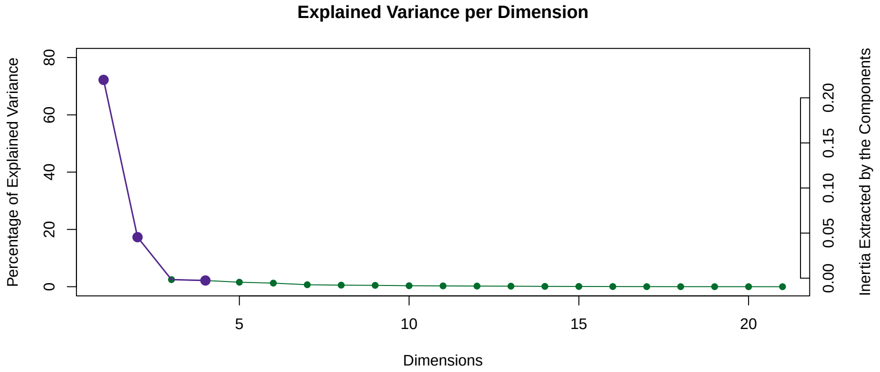
Multiple Correspondence Analysis is run here using the function epMCA in package ExPosition version 2.8.23. The design was manually recoded later in this chapter.

```
dm_beers_cat <- as.factor(df_beers[,2])
resMCA <- epMCA(DATA = mesBeerRecoded,
                graphs = FALSE )
#Inference battery is also run here:
resMCA.inf <- InPosition::epMCA.inference.battery(
                    DATA = mesBeerRecoded,
                    graphs =  FALSE)
```

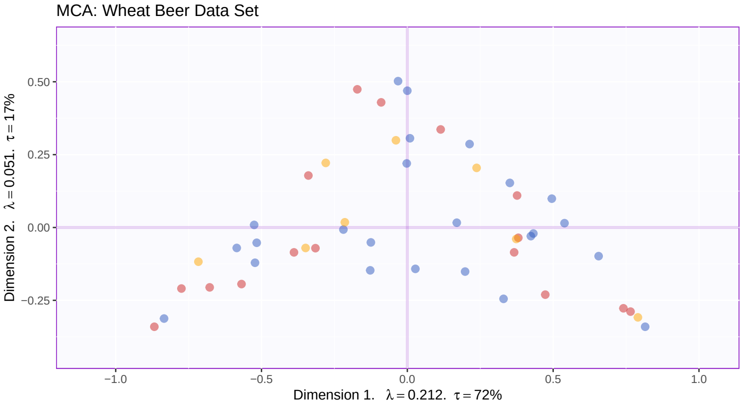
```
## [1] "It is estimated that your iterations will take 0.08 minutes."
## [1] "R is not in interactive() mode. Resample-based tests will be conducted. Please
## ================================================================================
```

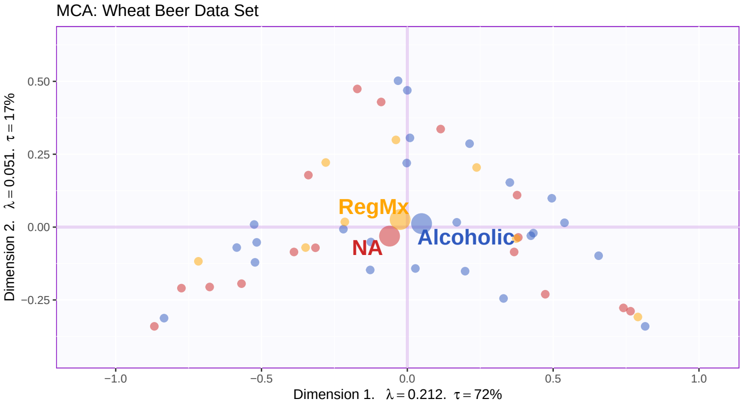
3.6 Plots

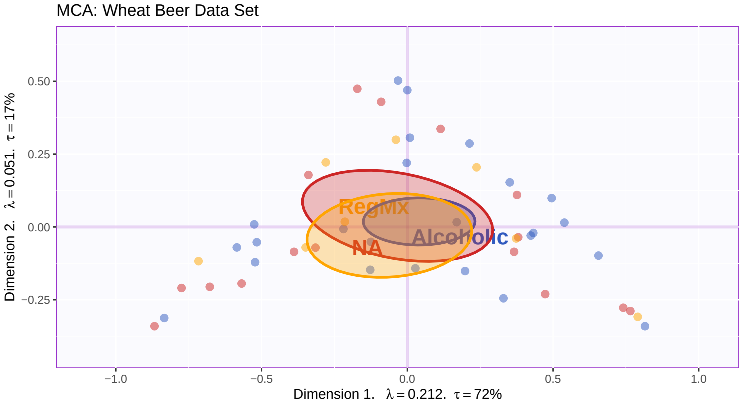
The following is the scree plot for the eigenvalues of the MCA. We see that component one explains roughly 72% of the variance in the data followed by roughly 17% and 2.5% for components 2 and 3 respectively.

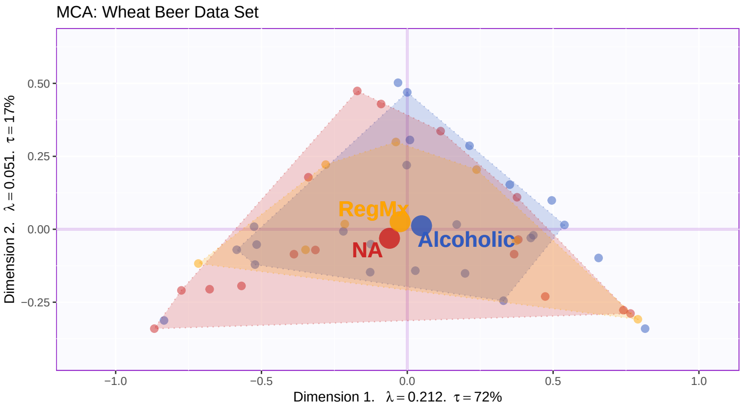


The factor scores for the rows of the data are created here. Not much can be interpreted by the eye from these factor scores since no obvious clusters are exhibited. The yellow dots represent the regular Mx non-wheat beer product, the blue dots represent alcoholic beers, and red non-alcoholic beers.

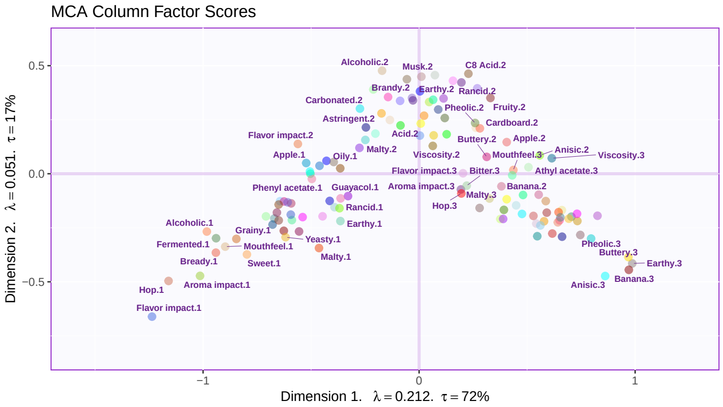

MCA: Wheat Beer Data Set
Dimension 2. λ = 0.051. τ = 17%
0.50
0.25
0.00
−0.25
−1.0
−0.5
0.0
0.5
1.0
Dimension 1. λ = 0.212. τ = 72%


MCA: Wheat Beer Data Set
RegMx
NA
Alcoholic
Dimension 2. λ = 0.051. τ = 17%
0.50
0.25
0.00
−0.25
−1.0
−0.5
0.0
0.5
1.0
Dimension 1. λ = 0.212. τ = 72%

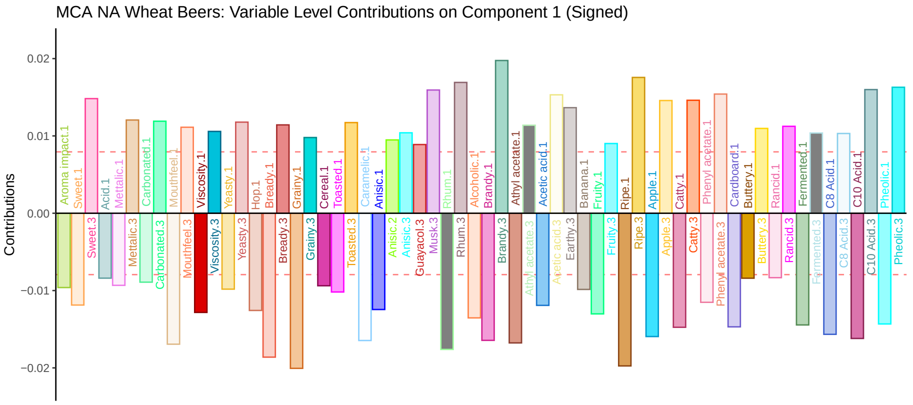

MCA: Wheat Beer Data Set
RegMx
NA
Alcoholic
Dimension 2. λ = 0.051. τ = 17%
0.50
0.25
0.00
−0.25
−1.0
−0.5
0.0
0.5
1.0
Dimension 1. λ = 0.212. τ = 72%



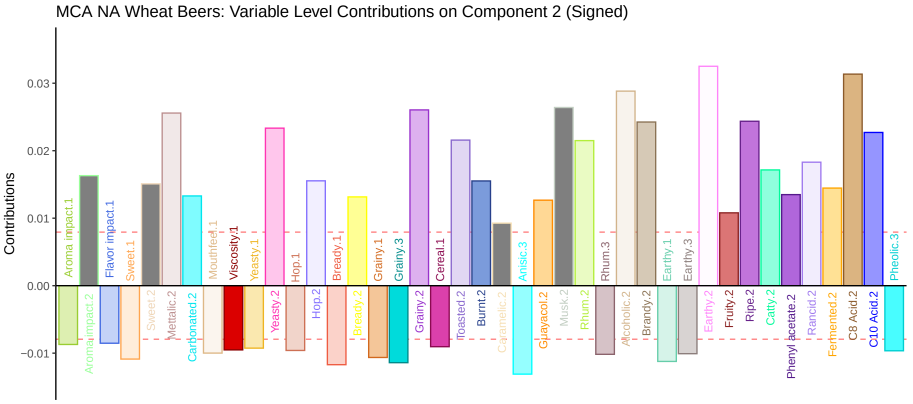
The factor scores for the columns of the data (sensory attributes) are created here. We see large contributions for alcoholic related sensory attributes on component 1 and wheat-beer related attributes on component 2.



The signed contribution plots are created here. The first contribution barplot is for component 1 and the second for component 2. Significant positive contributions on component 1 were shown in red, and negative shown in yellow.



MCA NA Wheat Beers: Variable Level Contributions on Component 1 (Signed)



MCA NA Wheat Beers: Variable Level Contributions on Component 2 (Signed)

3.7 Conclusion

Conclusion: The variance of the data explained by component 1 is due to the differences between non alcoholic and alcoholic beers. Alcoholic sensory attributes showed strong negative contributions on component 1, and non-alcoholic sensory attributes (such as sweet,fruity, bready, and grainy) The variance of the data explained by component 2 is due to the differences between wheat beers and non-wheat beers, with wheat beers showing positive contributions on component 2 and non-wheat beers negative.

Chapter 4

Barycentric Discriminant Analysis

4.1 Introduction

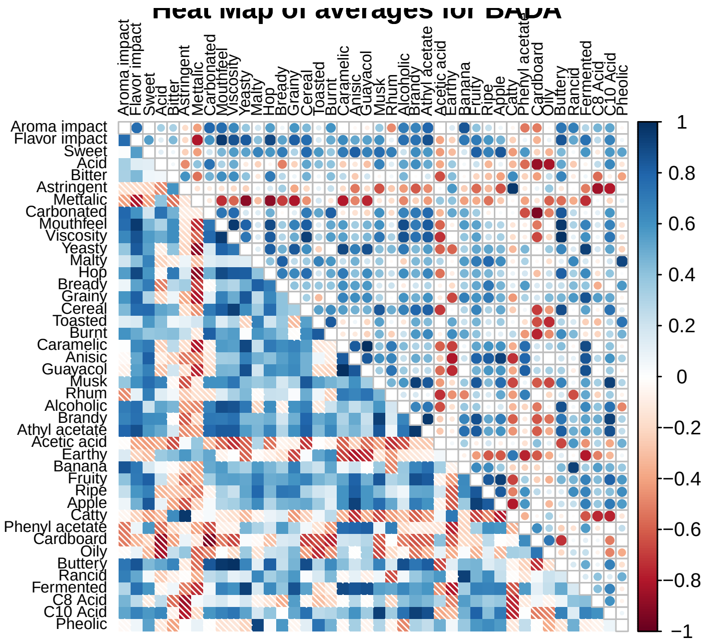
Barycentric Discriminant Analysis (BADA) generalizes discriminant analysis (DA). As in DA, BADA combines measurements of observations and assigns them to categories. BADA can be used when DA cannot, like when there are more variables than observations or when the measurements are categorical. BADA relies on the principle that each each category is represented by the barycenter (weighted average) of its observations.

References: Abdi, H., & Williams, L. J. (2010). Barycentric discriminant analysis (BADIA). Encyclopedia of research design, 64-75.

The data set to be analyzed is the same as in previous chapters: The alcoholic vs non-alcoholic wheat beer sensory profile.

4.2 Heatmap

Here is the heatmap of averages for the BADA. We see mostly positive correlations accross the board, however, we see acetic acid and mettalic are mostly negatively correlated with the other sensory attributes.

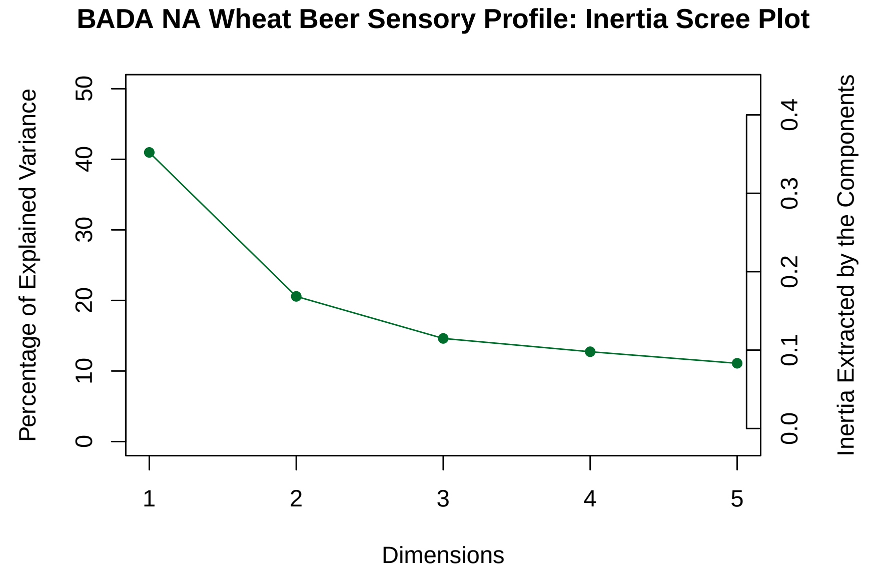


4.3 BADA Code and Scree

Here is the code used to run the BADA, as well as the scree plot. We see that dimension one here explains roughly 40% of the variance in our data and dimension two explains roughly 20%. We will focus on the first two dimensions in this analysis.

```
# Computations ----
# Run BADA  ----
resBADA <- tepBADA(XYmat, DESIGN = df_beers$Product,
                   graphs = FALSE)
# Inferences ----
#set.seed(70301) # we had a problem
# with the inference part
# it is addressed iin the Fix from Luke's github
nIter = 1000
resBADA.inf <- tepBADA.inference.battery(XYmat,
                  DESIGN = df_beers$Product,
                  test.iters = nIter,
                  graphs = FALSE)
```

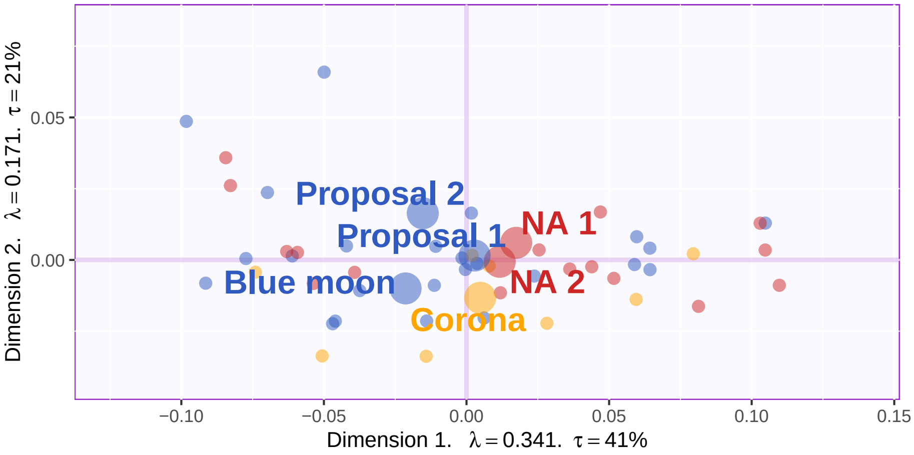
```
## [1] "It is estimated that your iterations will take 0.56 minutes."
## [1] "R is not in interactive() mode. Resample-based tests will be conducted. Please
## ============================================================================
```

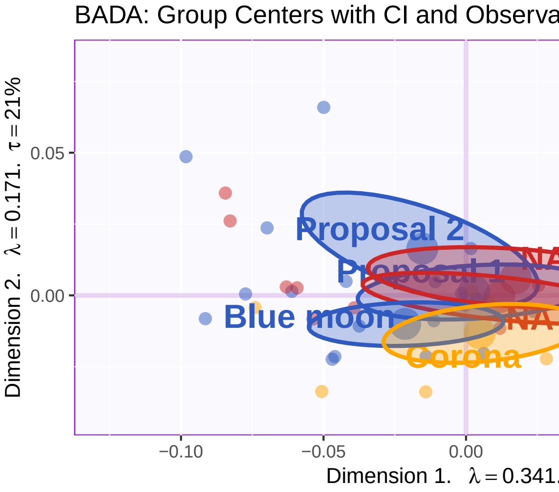


4.4 Plots

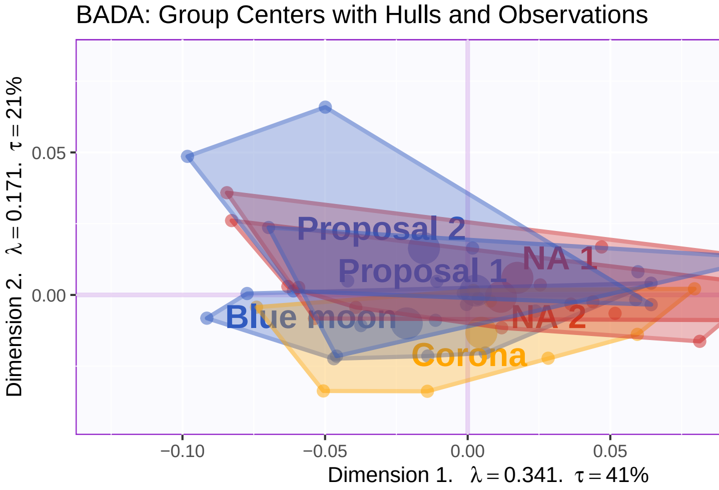
4.4.1 Row Factor Scores

Here are the row factor scores with observations and group means. We see blue moon and proposal two beer means (both of which are alcoholic) have negative contributions, while the rest of the beer means are positive. However, there is not a clear separation of beer type in this case.





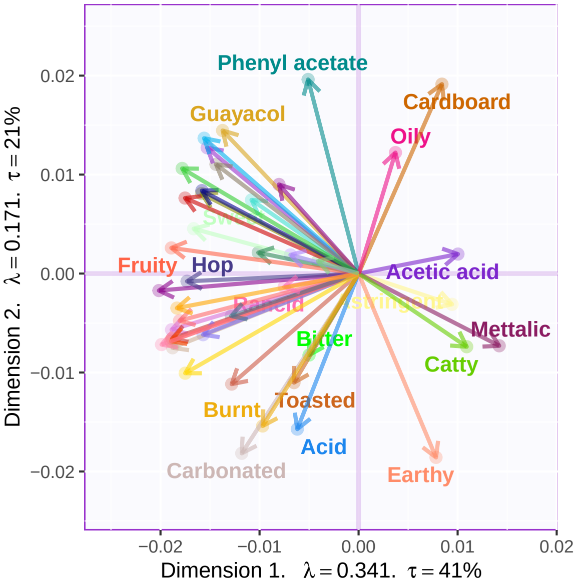
Here are the row factor scores with confidence intervals:



Here are the row factor scores with tolerance intervals:

4.4.2 Col Factor Scores

Here are the column factor scores. For component one, we see a distinction between Fruity/Hop vs. Acetic Acid/Mettalic/Astringent. For component two, we see the differences between carbonated, acid, and earthy vs. phenyl ac-

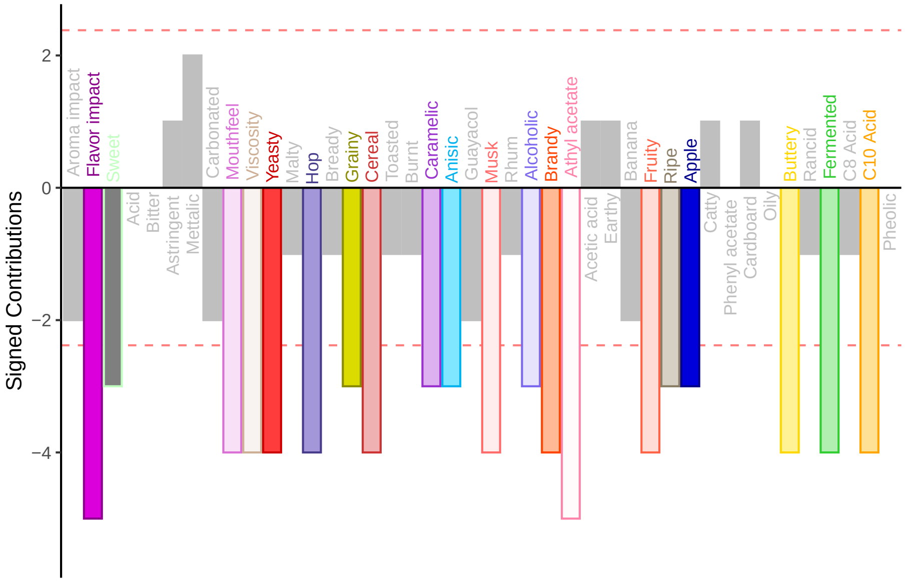


etate/oily/cardboard.

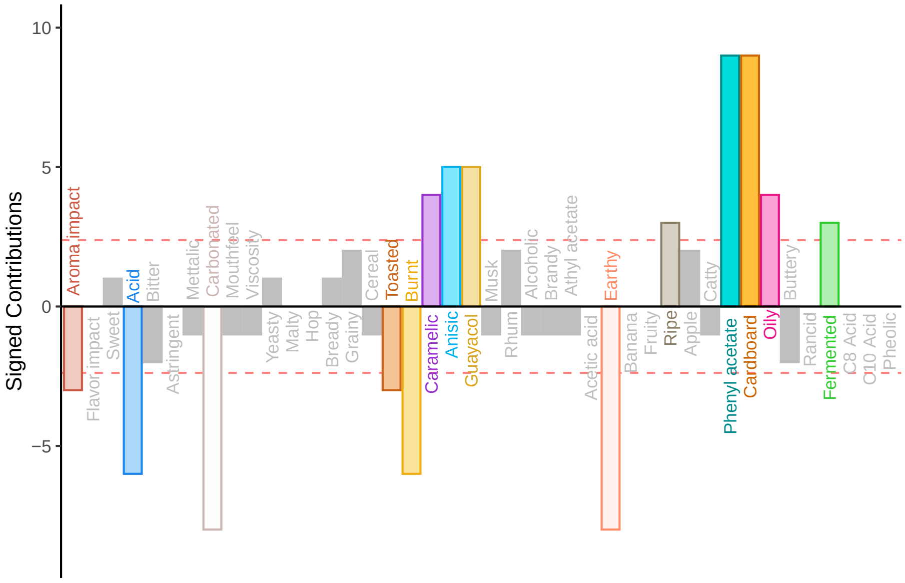
4.4.3 Contribution Plots

Here are the contribution plots for the BADA. alcoholic sensory attributes were important on component 1 and component 2 was some wheat related sensory attributes, meaning the panelists may have interpreted these attributes differently.

Important Contributions Variables. Dim 1.

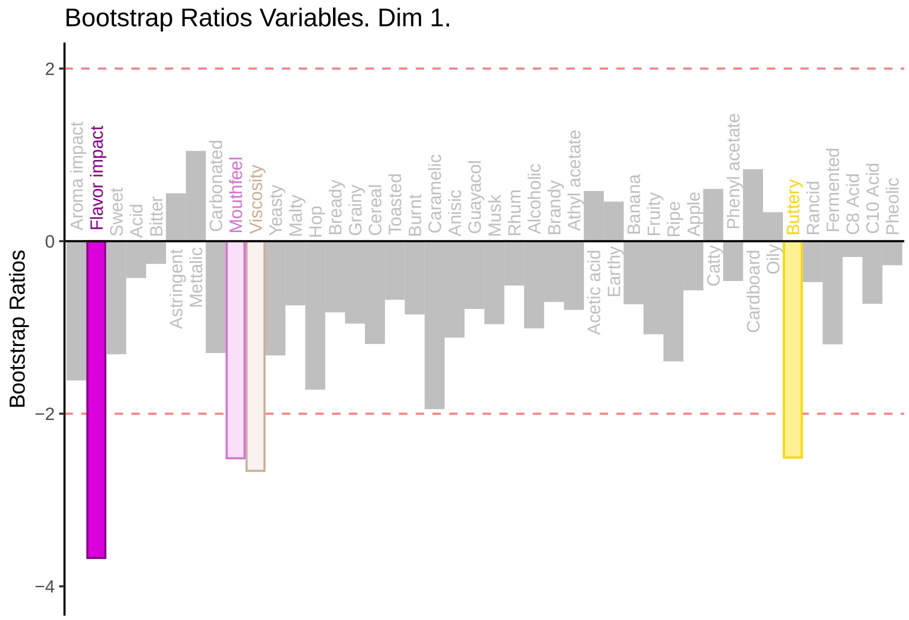


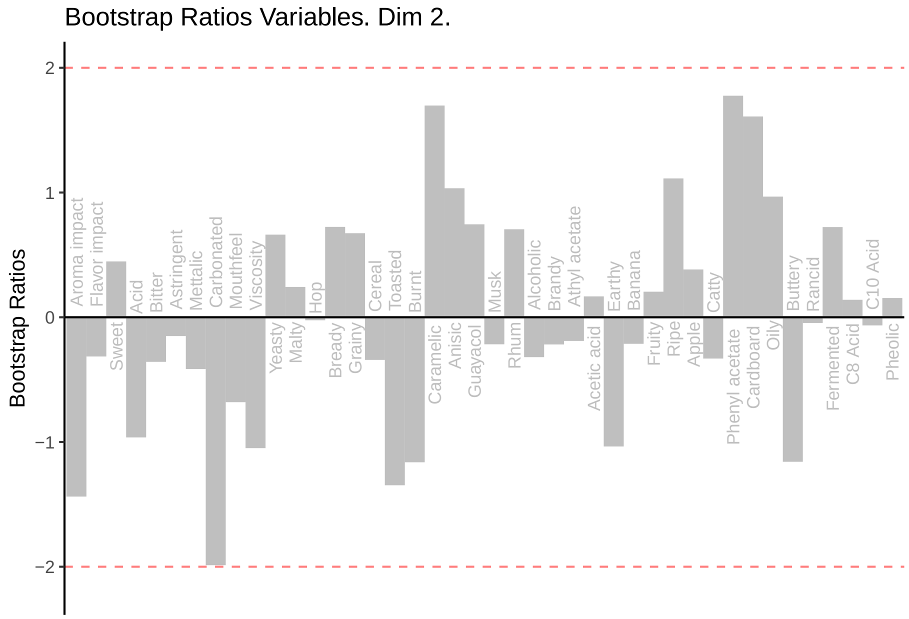
Important Contributions Variables. Dim 2.



4.4.4 Bootstrap plots

Here are the bootstrap ratio barplots for the variables. Although the contribution plots showed many important contributions, The bootstrap ratios show that most of these contributions did not reach threshold significance levels.





4.5 Confusion Matrices

Here are the fixed and random effect confusion matrices, along with the prediction accuracy. There was a large drop in the fixed accuracy vs. the random accuracy, however, the fixed accuracy is still quite low at 42% meaning BADA did a poor job of classifying our observations.

```
#fixed confusion matrix
fixedCM <- resBADA.inf$Inference.Data$loo.data$fixed.confuse
head(fixedCM) %>% kable()
```

	.Proposal 1	.Proposal 2	.NA 1	.NA 2	.Blue moon	.Corona
.Proposal 1	4	3	0	1	3	1
.Proposal 2	1	4	2	3	0	1
.NA 1	1	0	2	1	1	1
.NA 2	1	0	2	2	0	0
.Blue moon	0	0	1	0	3	0
.Corona	1	1	1	1	1	5

```
#fixed accuracy
resBADA.inf$Inference.Data$loo.data$fixed.acc
```

```
## [1] 0.4166667
```

```
#random confusion matrix
randomCM <- resBADA.inf$Inference.Data$loo.data$loo.confuse
head(randomCM) %>% kable()
```

	.Proposal 1.actual	.Proposal 2.actual	.NA 1.actual	.NA 2.actual	.Blue moon.actua
.Proposal 1.predicted	0	4	1	3	3
.Proposal 2.predicted	1	1	2	2	2
.NA 1.predicted	1	0	0	1	1
.NA 2.predicted	4	0	3	1	0
.Blue moon.predicted	0	2	1	0	0
.Corona.predicted	2	1	1	1	2

```
#random accuracy
resBADA.inf$Inference.Data$loo.data$loo.acc
```

```
## [1] 0.04166667
```

4.6 Conclusion

Conclusion: There was some separation of row factor scores between beer type, as a separation of non alcoholic beers was witnessed from alcoholic beers on component 1, and alcohol related contributions were important on component 1. Component 2 is most likely related to the effects of the panelists grading the beers (for the rows). The contribution plots also tell the same story, as alcoholic sensory attributes were important on component 1 and component 2 was some wheat related sensory attributes, meaning the panelists may have interpreted these attributes differently. The bootstrap ratios, however, show that these contributions did not reach threshold significance levels.

Chapter 5

Discriminant Correspondence Analysis

5.1 Introduction

Introduction: Discriminant Correspondence Analysis (DiCA) is an extension of Discriminant Analysis (DA) and Correspondence Analysis (CA). In DiCA, observations are categorized into groups (as in DA). With DiCA, each group is represented by the sum of it's observations and a CA is performed on the data matrix. Afterward the original observations are assigned to the closest group after projecting them as supplementary elements. This chapter will go through and example where DiCA is applied to the "Wheat Beer Sensory Profile" data set, the same data frame used in the previous chapters.

References: Abdi, H. (2007). Discriminant correspondence analysis. Encyclopedia of measurement and statistics, 2007, 1-10.

5.2 Data and Binning

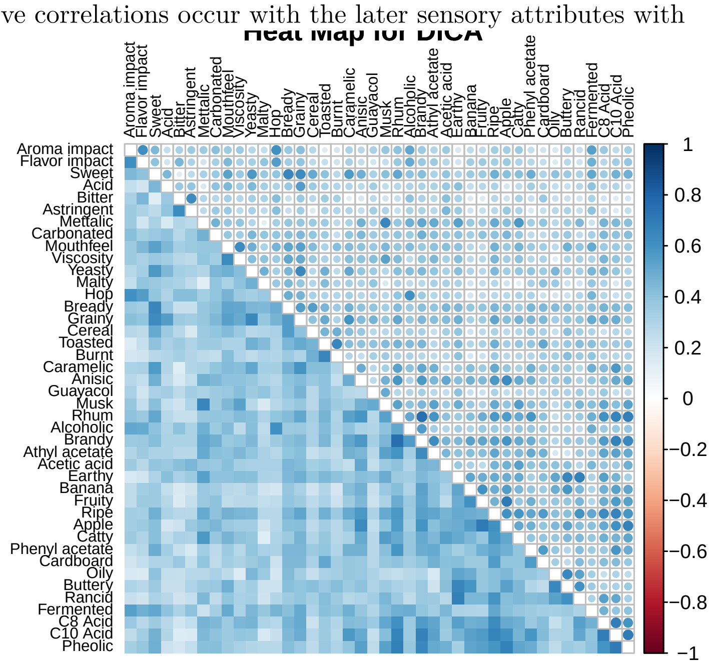
Here is the importing and binning of the data. We are using the same data set: the alcoholic vs. non-alcoholic sensory profile.

```
df_beers <- import("Wheat beer no alcohol.xlsx")
dm_beers <- data.matrix(df_beers[1:48,3:44])
head(dm_beers)
XYmat <- data.frame(
               row.names = rownames(dm_beers))
irec = c(1:42)
```

```r
for(val in irec)
{
  XYmat[,colnames(dm_beers)[irec]] <- BinQuant(
               dm_beers[,irec], nClass = 3, stem = '')
}
XYmat <- as.matrix(XYmat)
XYmat <- apply(XYmat,2,as.numeric)
```

5.3 Heat Map

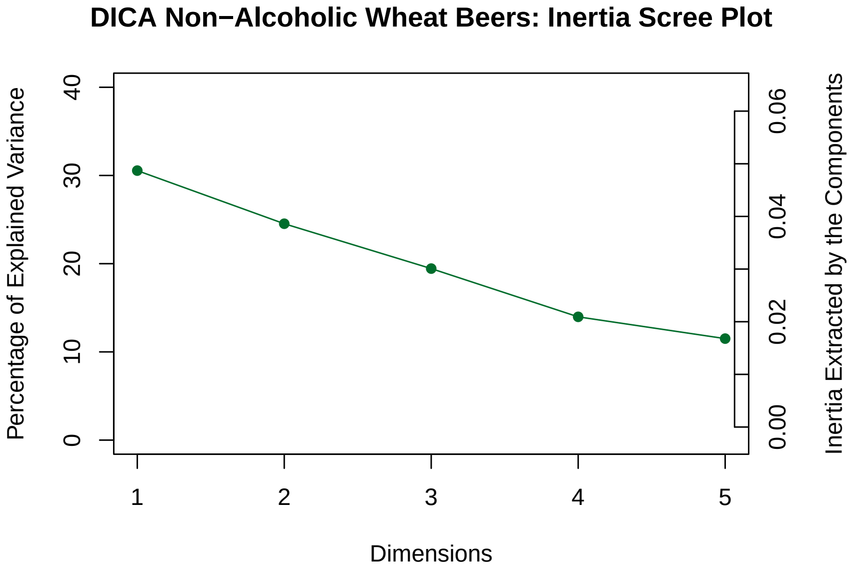
Here is the heatmap for the discriminant correspondence analysis. We see that most strongly positive correlations occur with the later sensory attributes with each other.



5.4 DiCA Code and Scree

Here I've applied the DiCA and inference battery functions to the binned data. The Scree plot is also shown here. We see the first dimension explains roughly 30% of the data, while the second dimension explains roughly 25% of the data. We will focus on the first two dimensions in this analysis.

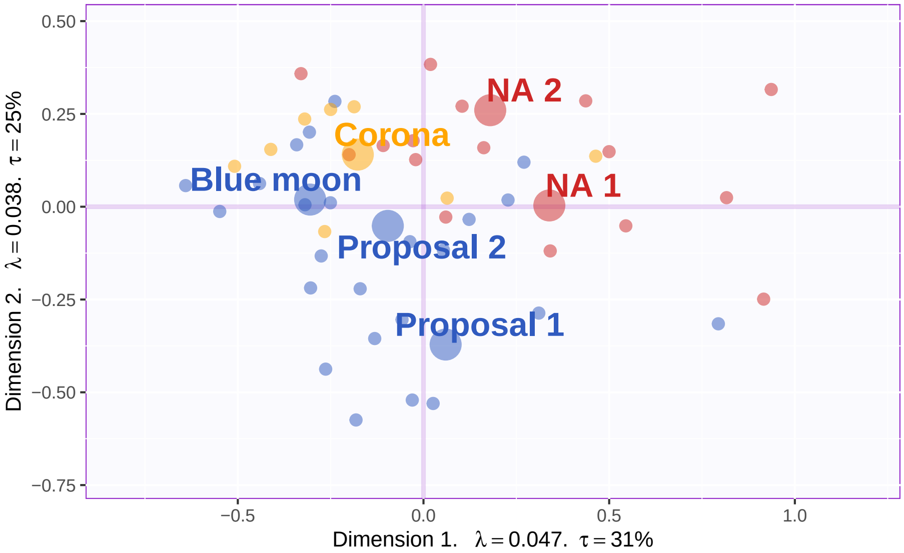
```
## [1] "It is estimated that your iterations will take 0.15 minutes."
## [1] "R is not in interactive() mode. Resample-based tests will be conducted. Please take note
## ================================================================================
```



5.5 Plots

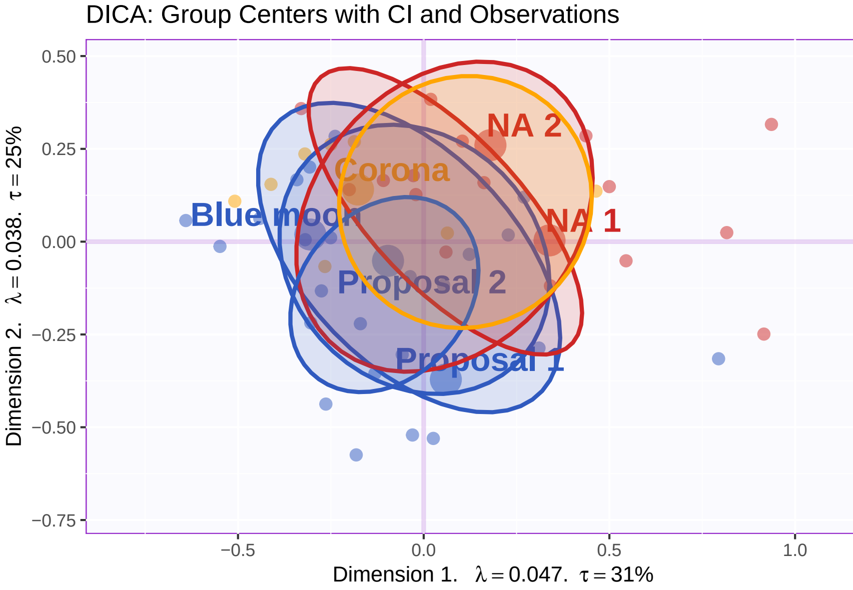
5.5.1 Row factor scores

Here is the code to plot the row factor scores with group means. The separation of row factor scores between beer type was quite apparent, as a separation of non alcoholic beers was witnessed from alcoholic beers on component 1. Component 2 is most likely related to the effects of the panelists grading the beers (for the

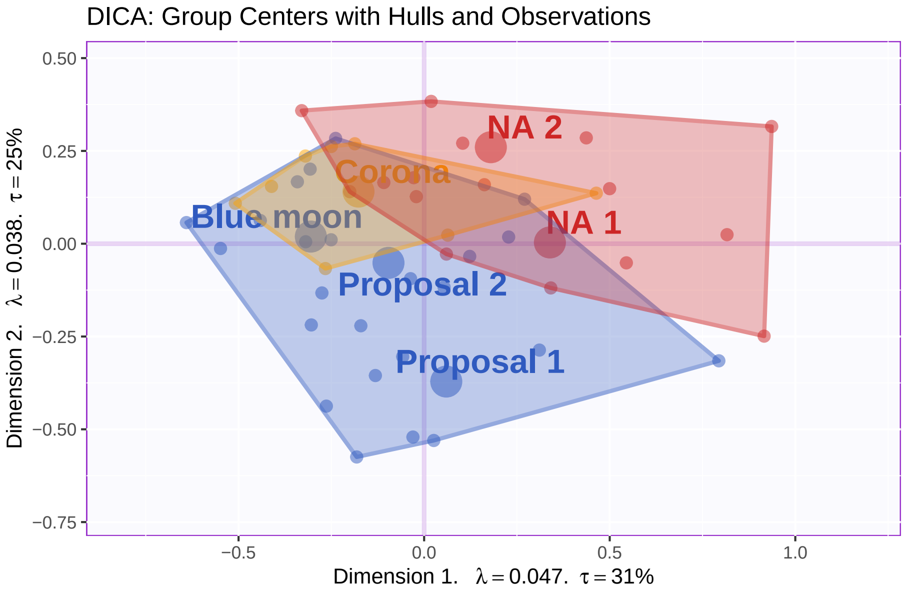


rows).

Here I've ploted the row factor scores with confidence intervals, we see some overlap of the confidence intervals but there is still a clear seperation between alcoholic and non-alcoholic beers.

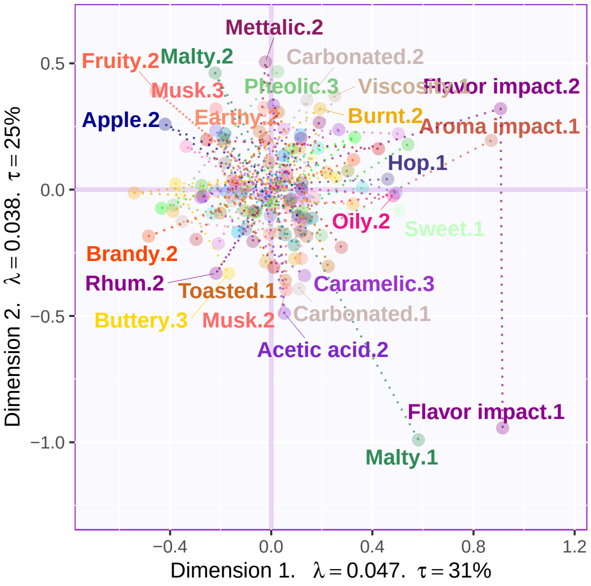

DICA: Group Centers with CI and Observations

Here is I've plotted the same row factor scores with tolerance interval hulls:



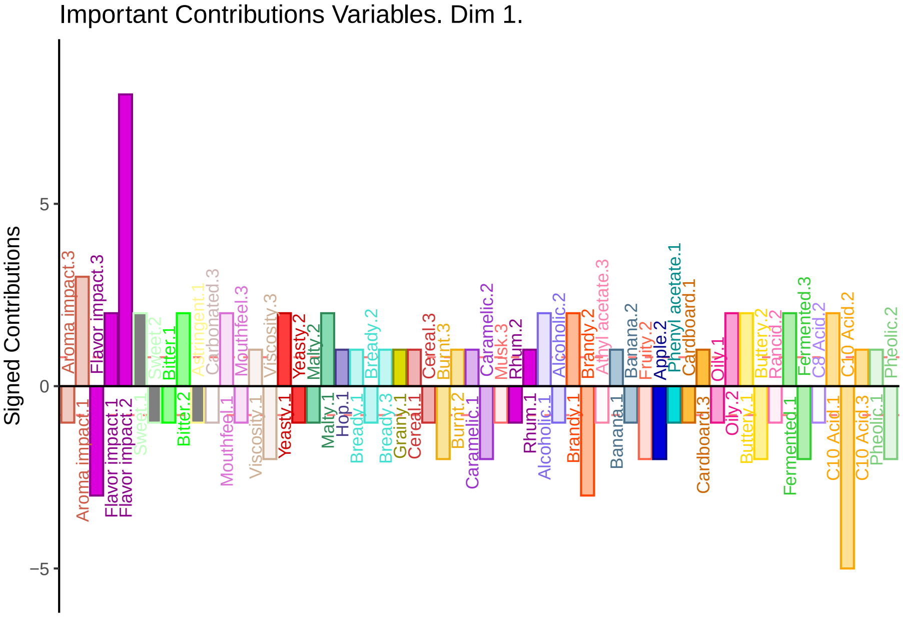
5.5.2 Col factor scores

Here is the code producing the column factor scores. We see that component one shows the difference brandy/rhum/fruity vs. flavor impact/aroma impact/oily sensory attributes. Component two shows the differences between mettalic/pheolic vs. buttery/malty/acetic acid sensory attributes.

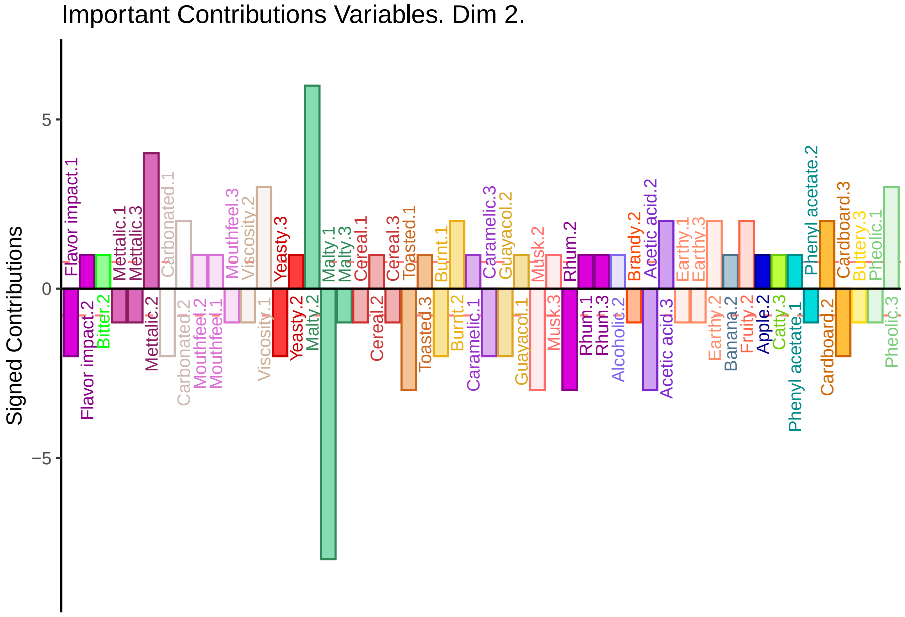


5.5.3 Contribution Plots

Here is the code producing the Contribution Plots. Alcoholic sensory attributes were important on component 1 and component 2 was some wheat related sensory attributes, meaning the panelists may have interpreted these sensory at-

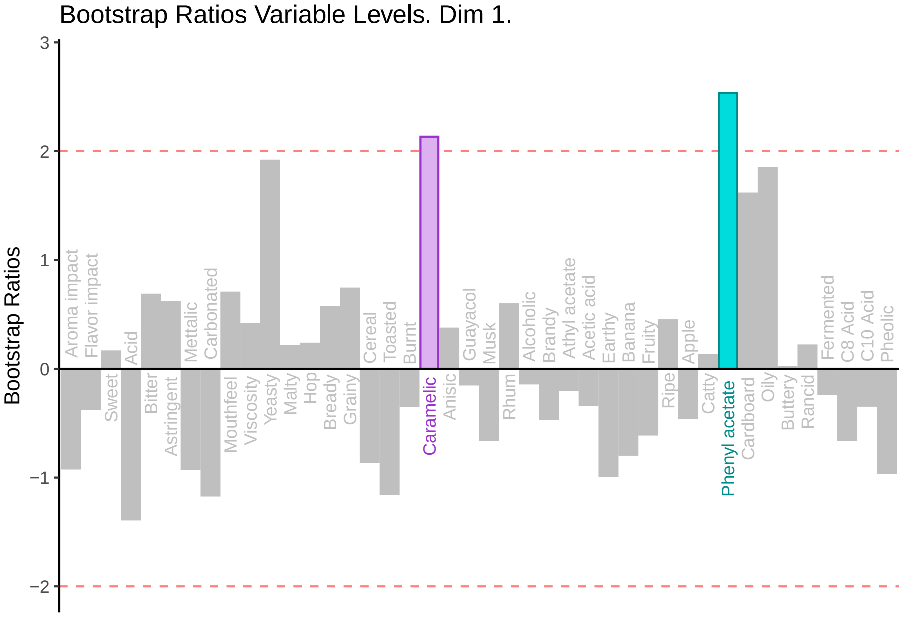


tributes differently.

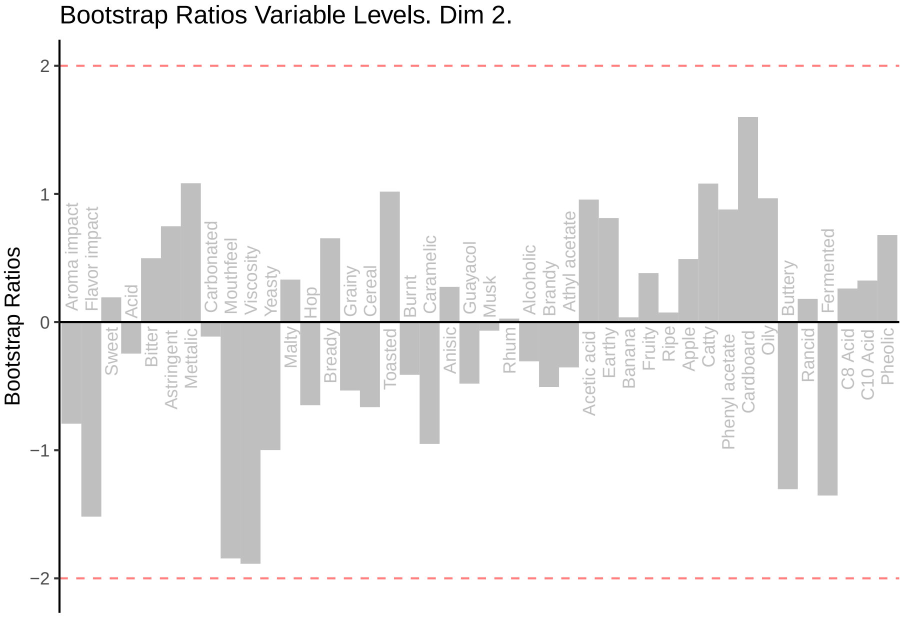


5.5.4 Bootstrap plots

The following generates the bootstrap ratio barplots. Although the contribution plots showed many important contributions, The bootstrap ratios show that most of these contributions did not reach threshold significance levels.



Bootstrap Ratios Variable Levels. Dim 1.



Bootstrap Ratios Variable Levels. Dim 2.

5.6 Confusion Matrices

Here are the fixed and random effect confusion matrices, along with the prediction accuracies. Similar to BADA, we see a steep drop in prediction accuracy between fixed and random data, however, the fixed accuracy in this case is larger than BADA, therefore DiCA was more accurate in predicting the classifications of our observations than.

	.Proposal 1	.Proposal 2	.NA 1	.NA 2	.Blue moon	.Corona
.Proposal 1	5	1	0	0	0	1
.Proposal 2	0	5	0	1	0	0
.NA 1	1	0	6	1	1	0
.NA 2	1	1	1	6	1	1
.Blue moon	1	1	0	0	6	2
.Corona	0	0	1	0	0	4

```
## [1] 0.6666667
```

	.Proposal 1.actual	.Proposal 2.actual	.NA 1.actual	.NA 2.actual	.Blue moon.actua
.Proposal 1.predicted	0	3	2	2	1
.Proposal 2.predicted	3	0	1	1	1
.NA 1.predicted	1	1	1	2	2
.NA 2.predicted	1	1	2	1	1
.Blue moon.predicted	2	3	0	0	1
.Corona.predicted	1	0	2	2	2

```
## [1] 0.08333333
```

5.7 Conclusion

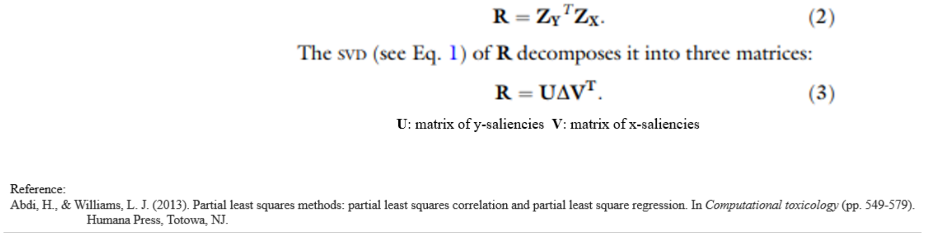
The separation of row factor scores between beer type was quite apparent, as a separation of non alcoholic beers was witnessed from alcoholic beers on component 1. Component 2 is most likely related to the effects of the panelists grading the beers (for the rows). The contribution plots also tell the same story, as alcoholic sensory attributes were important on component 1 and component 2 was some wheat related sensory attributes, meaning the panelists may have interpreted these sensory attributes differently.

Chapter 6

Partial Least Squares Correlation

6.1 Introduction

Partial Least Squares Correlation strives to relate the measurements of two data tables taken on the same set of observations and find the shared information in the tables. To do this, maximal covariance is found between two new variables (one for each table) called latent variables, which are derived as linear combinations of the original observations, which are found by projecting the original matrices to their saliencies.



$$\mathbf{R} = \mathbf{Z_Y}^T\mathbf{Z_X}. \qquad (2)$$

The SVD (see Eq. 1) of **R** decomposes it into three matrices:

$$\mathbf{R} = \mathbf{U}\Delta\mathbf{V}^\mathbf{T}. \qquad (3)$$

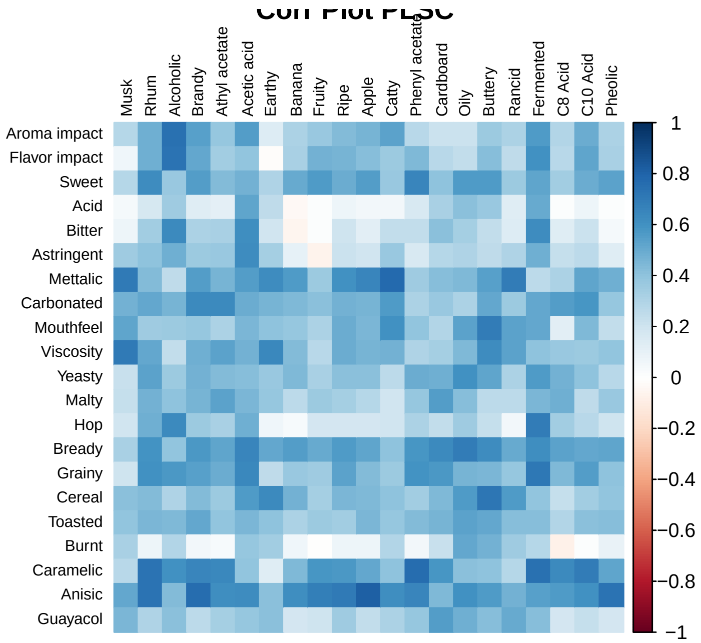
U: matrix of y-saliencies **V**: matrix of x-saliencies

Reference:
Abdi, H., & Williams, L. J. (2013). Partial least squares methods: partial least squares correlation and partial least square regression. In *Computational toxicology* (pp. 549-579). Humana Press, Totowa, NJ.

6.2 Data and Correlation matrix

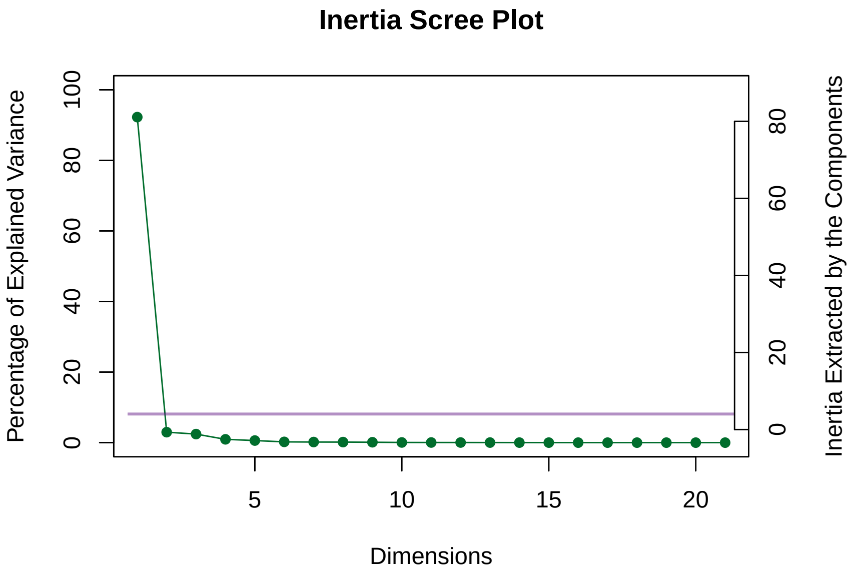
In this analysis, Xmat consists of the first half of sensory attributes and Ymat consists of the last half of sensory attributes. Here is shown the correlation matrix and data import. The correlation plots shows mostly positive or zero correlation values between the sensory attributes of Xmat and Ymat.

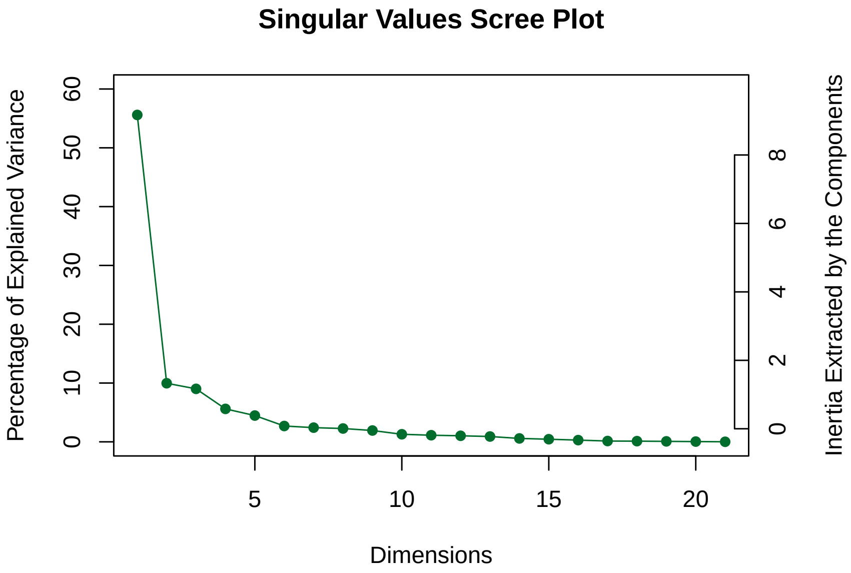
```
#data set:
df_beers <- import("Wheat beer no alcohol.xlsx")
dm_beers <- data.matrix(df_beers[1:48,3:44])
Xmat <- dm_beers[,1:21]
Ymat <- dm_beers[,22:42]
```



6.3 Scree Plot

The PLSC function call and scree plot (after 1000 iterations permutation testing) are shown. The Scree plot shows how much variance is explained by each dimension. The singular values scree plot is computed by taking the square root of the eigenvalues, and therefore is not additive. The inertia scree plot shows that dimension one explained roughly 95% of the variance, and is the only dimension that falls above the Kaiser Line.


Inertia Scree Plot
Percentage of Explained Variance
0
20
40
60
80
100
Inertia Extracted by the Components
0
20
40
60
80
5
10
15
20
Dimensions

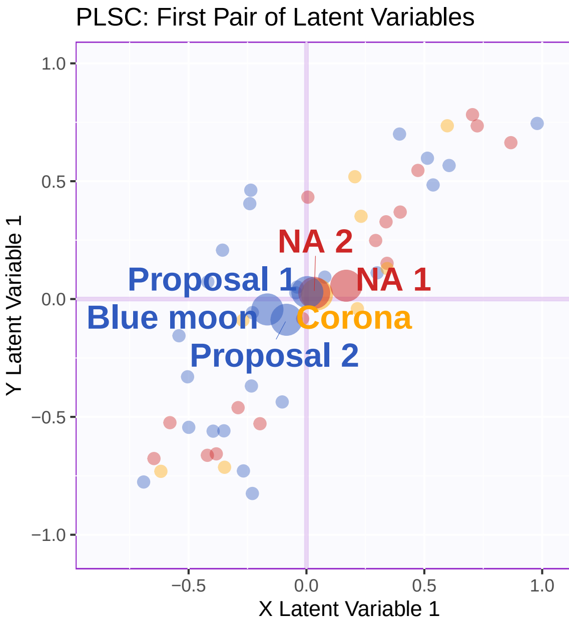

Singular Values Scree Plot
Percentage of Explained Variance
0
10
20
30
40
50
60
Inertia Extracted by the Components
0
2
4
6
8
5
10
15
20
Dimensions

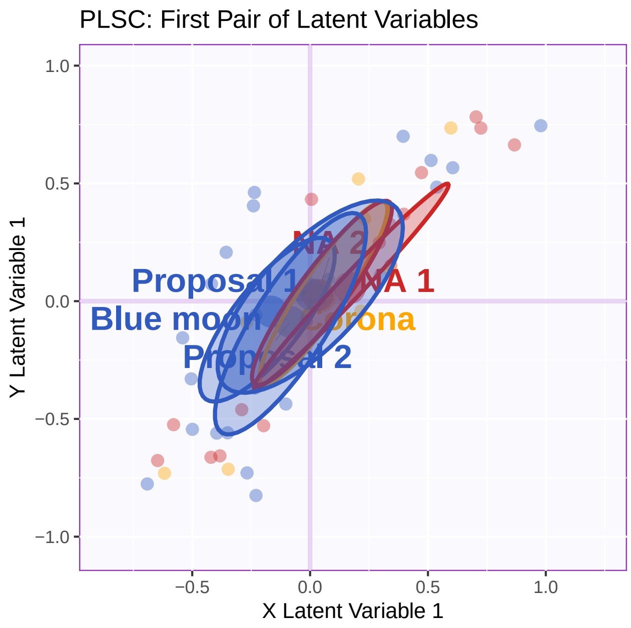
6.4 Dimension One

The following analyzes dimension one of the PLSC. The latent variable scores are plotted for dimension one and dimension two, and the confidence intervals for each group (Alcoholic, Non-Alcoholic (NA), and Corona) are plotted on top.

6.4.1 Latent Variables

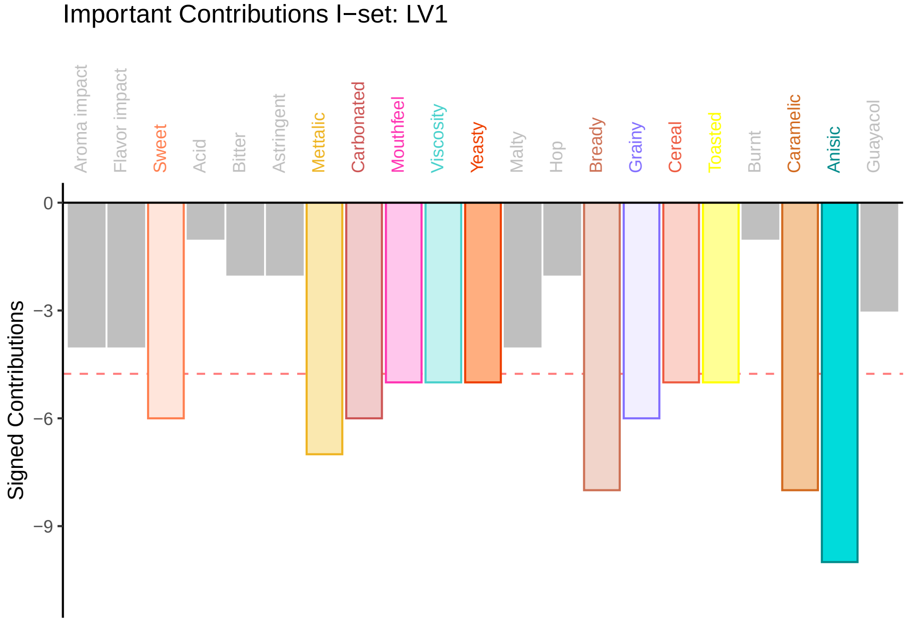
The latent variable plot for dimension one is shown here. We see a very slight seperation of alcoholic beers from nonalcoholic beers, with alcoholic beers displaying negative scores on component one.

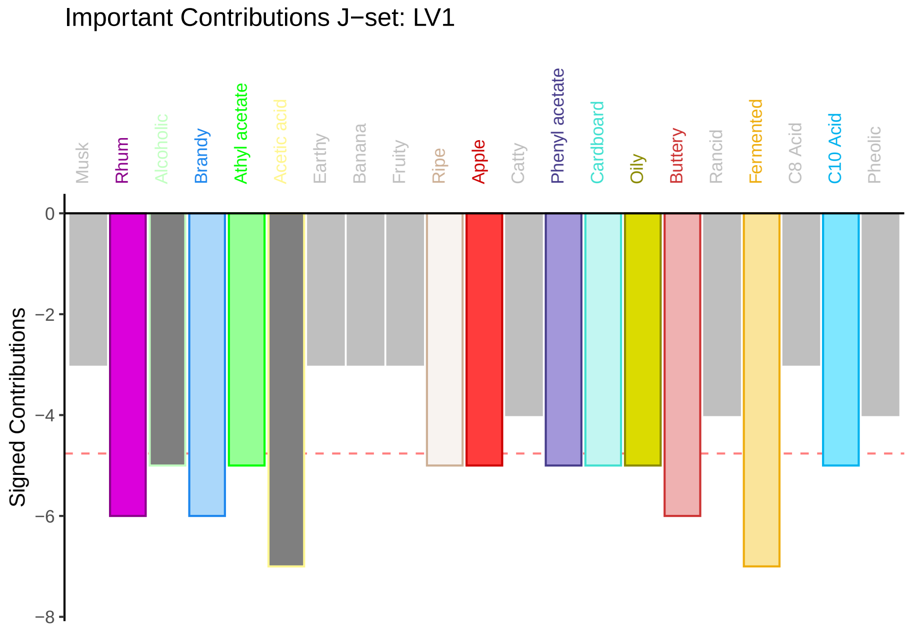




6.4.2 Contribution Plots

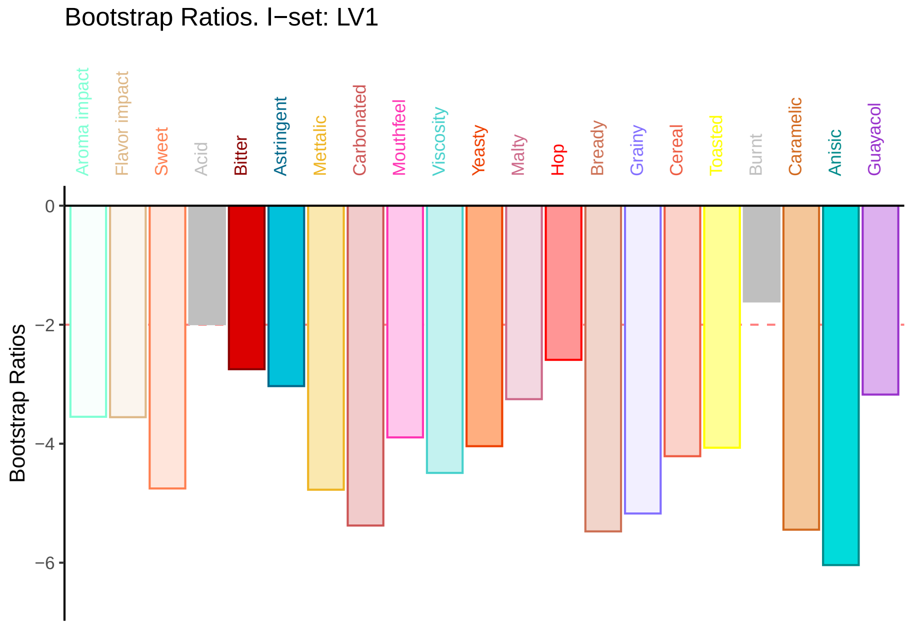
For dimension one, all contributions from the sensory attributes were negative, meaning no important differences between the attributes were seen on this dimension.

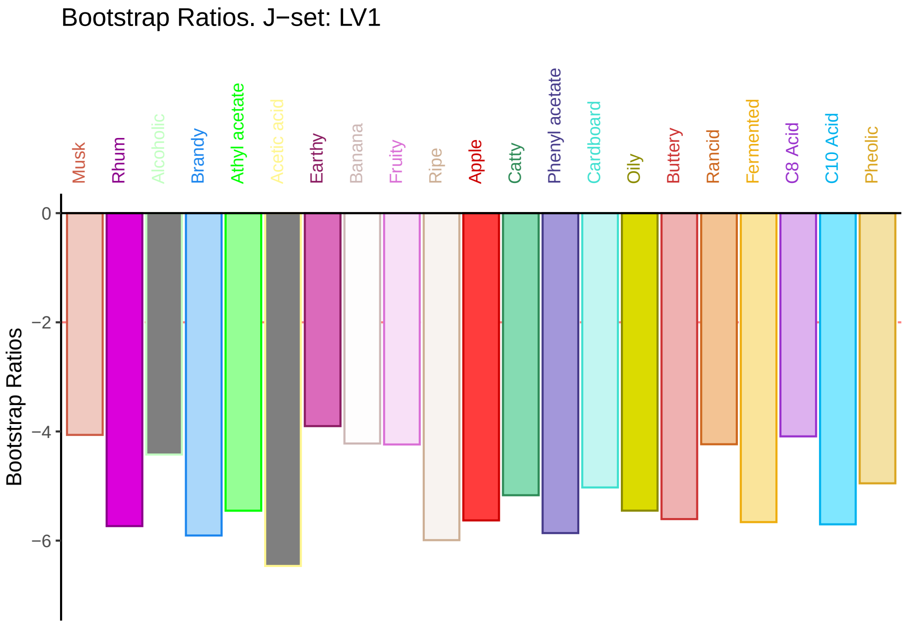




6.4.3 Bootstrap Plots

The bootstrap ratio significance level was applied to the contributions, and we see the same findings. Many of the contributions were found significant.



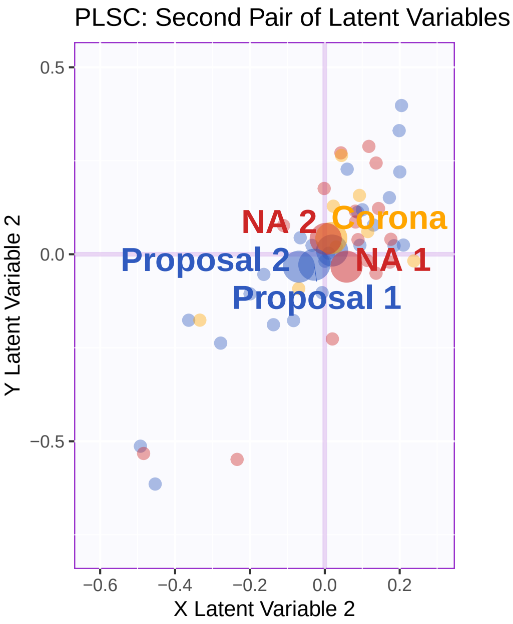


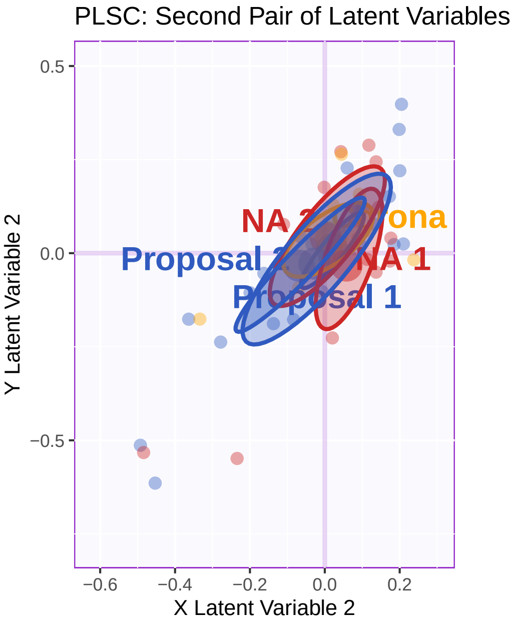
6.5 Dimension Two

The following analyzes dimension two of the PLSC.

6.5.1 Latent Variables 2

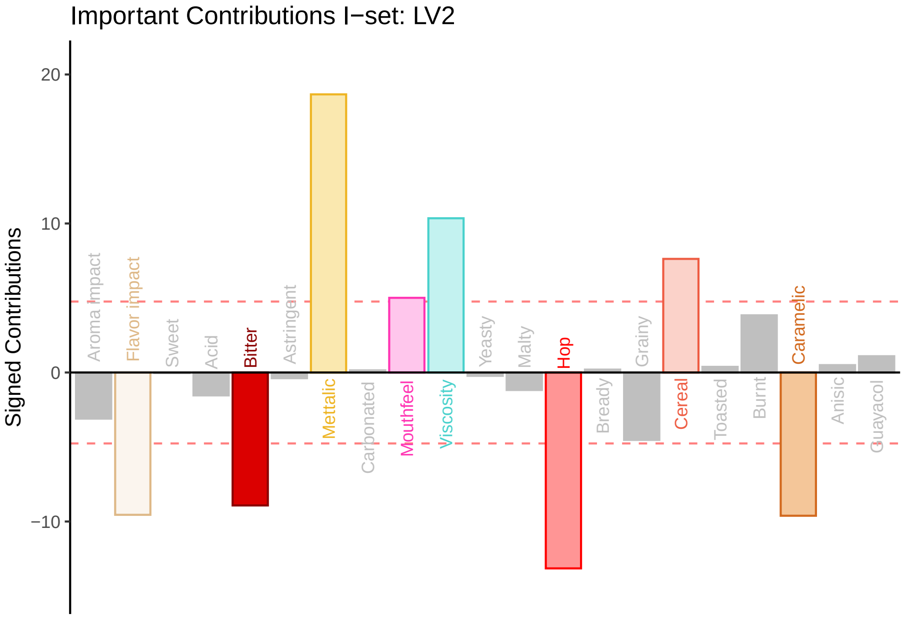
For the latent variables scores on the second dimension, we don't see any differences of the row means according to beer type, therefore no inferences can be made.

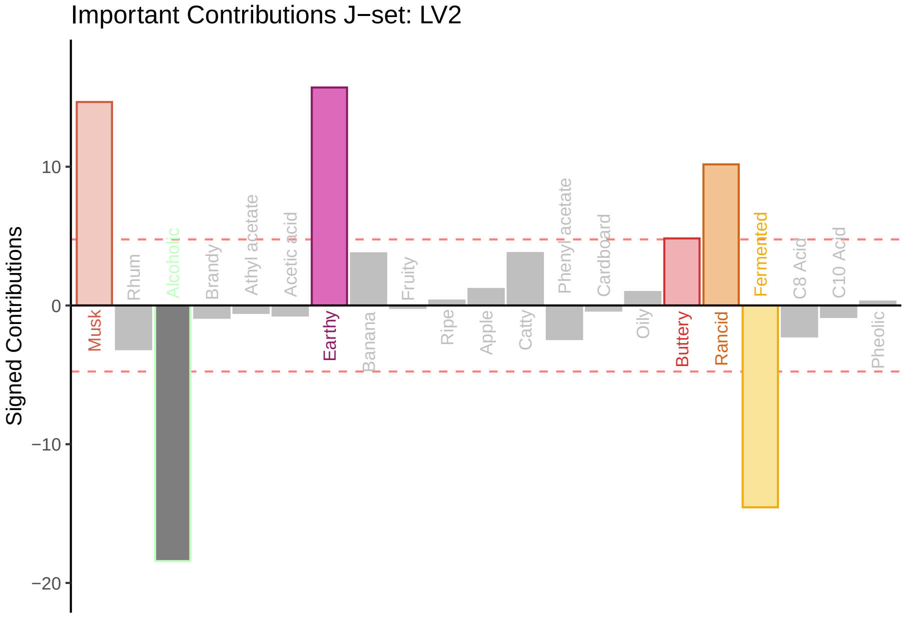




6.5.2 Contribution Plots 2

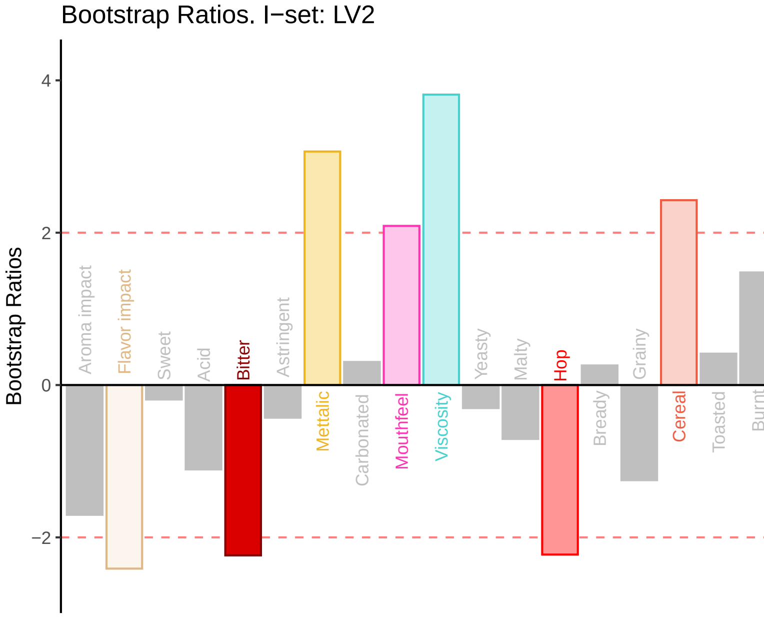
The contribution plot I set shows the differences between mettalic/viscosity/cereal vs. flavor impact/bitter/hop/carmelic sensory attributes. For the J set we see the differences between musky/earthy/buttery/rancid vs. fermented/alcoholic. The J set is most likely attributed to the difference between alcoholic and nonalcoholic beers.

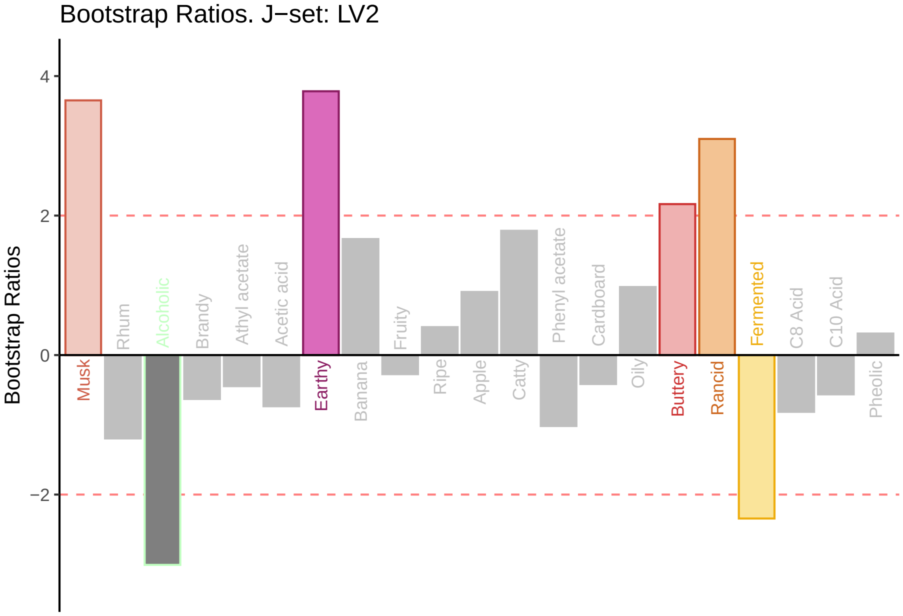




6.5.3 Bootstrap Plots 2

The bootstrapped significance is applied to the contributions, and we see similar results as with the contribution plots.





6.6 Conclusion

Dimension 1:
Latent Variables: No reliable differences between beer types.
I-plots: All attributes except acid and burnt significant.
J-plots: All attributes significant.
Dimension 2:
Latent Variables: No obvious differences between beer types.
I-plots: Flavor-impact, bitter, hop vs. Metallic, mouthfeel, viscosity, cereal.
J-plots: Fermented vs. Musk, Earthy, Buttery, Rancid

Chapter 7

DiSTATIS

7.1 Introduction

DiSTATIS is a generalization of multidimensional scaling (MDS); it is used to analyze a set of distance matrices instead of a single distance matrix. Distance matrices are compated by creating a common structure called a compromise, which is a combination of the matrices. The original distance matrices are then projected on the compromise.

Reference: Abdi, H., Valentin, D., Chollet, S., & Chrea, C. (2007). Analyzing assessors and products in sorting tasks: DISTATIS, theory and applications. Food quality and preference, 18(4), 627-640.

7.2 Data Set & Function Call

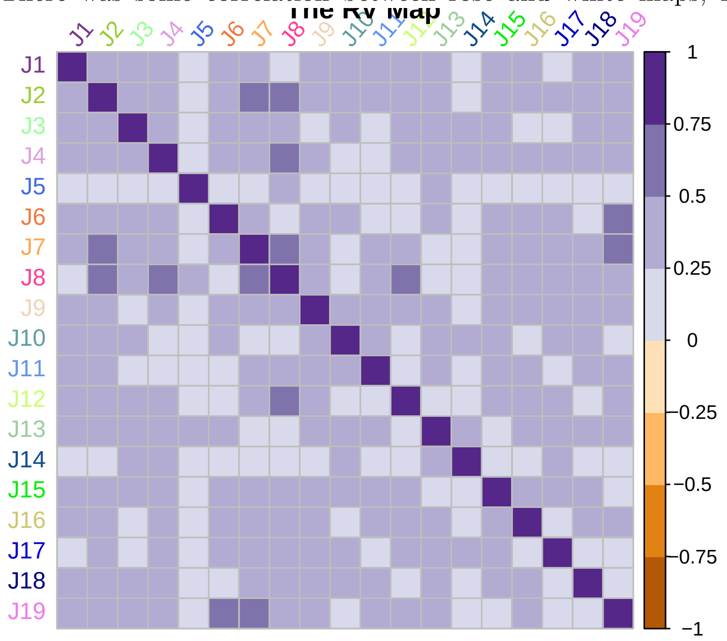
Here we import the data set and compute the necessary variables for the DiSTATIS analysis. The data is the groupings of 19 judges on 18 (6 red, 6 white, and 6 rose wines) different wines. Groupings scaled from 1 to 9 in this data set. The purpose of the lines of code are commented for your reference.

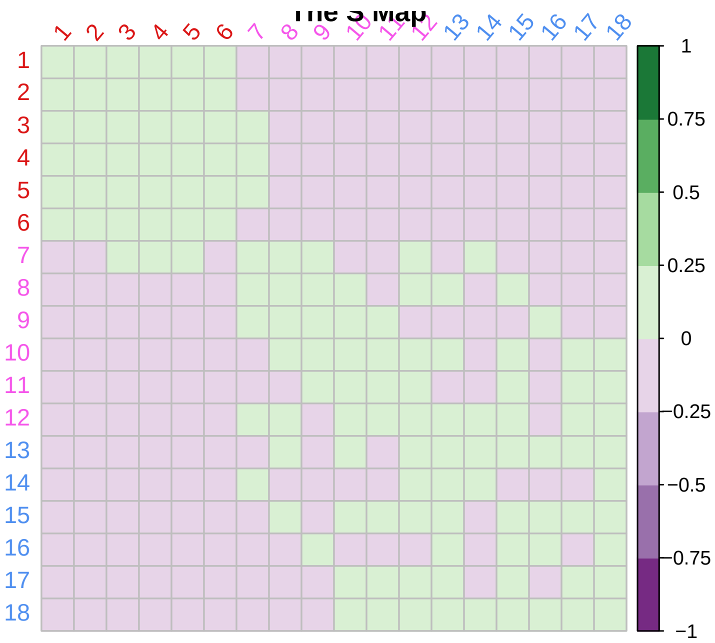
```r
#import the data frame and pull quantitative data to data matrix
df_wines <- import("WinesAndColors.xlsx")
dm_wines <- data.matrix(df_wines[,4:22])
multiSort <- dm_wines
#Compute the distance matrix
DistanceCube <- DistanceFromSort(multiSort)
# **** Computations ----
## runDistatis---------------------------------
```

```r
resDistatis <- distatis(DistanceCube,
                        nfact2keep = 10)
n.active <- dim(DistanceCube)[3]
```

7.3 Heat Maps

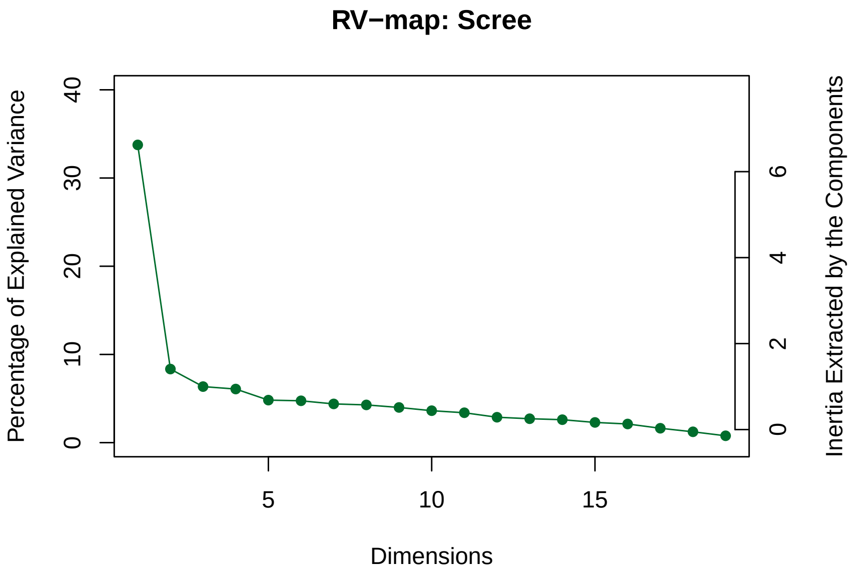
The Rv and S heat maps from the DiSTATIS are shown here. We use these heat maps to get an overall idea of how the variables in our data relate. For the Rv map, we see only positive correlations with a mix of slightly stronger positive correlations. Judge 5 across the board had low correlation values as compared to the other judges. For the S map, the wines of the same type correlated with each other, but not with other wines. There was some correlation between rose and white maps, however.

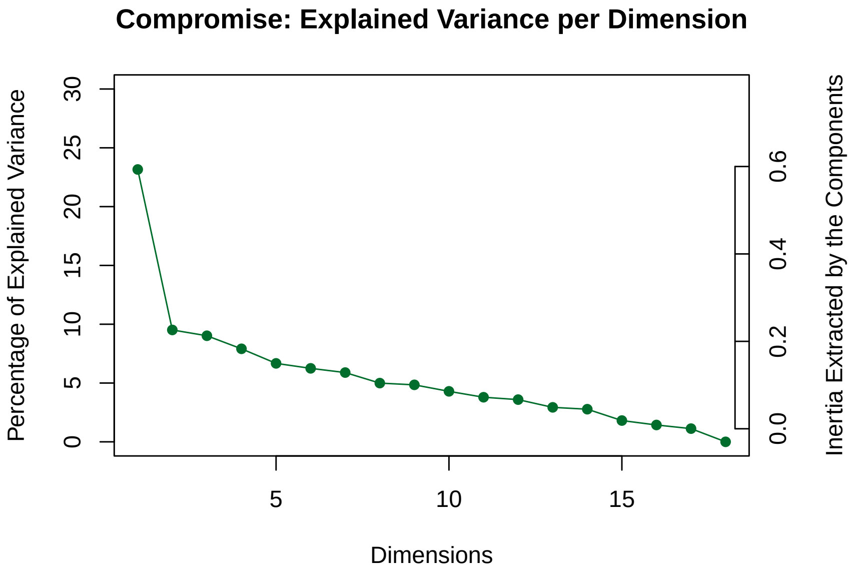




7.4 Scree Plots

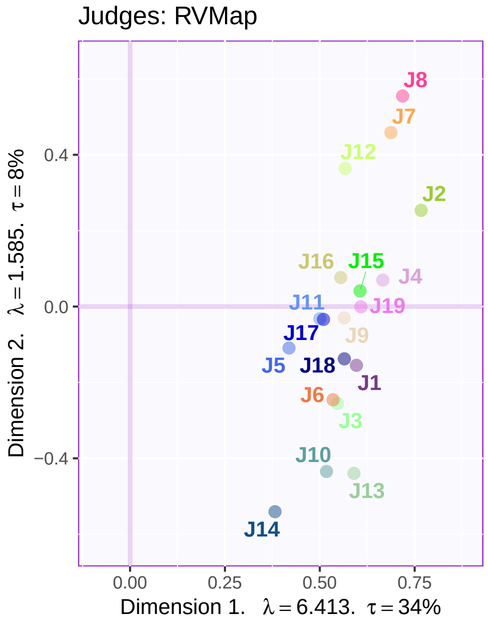
The Scree plots of the Rv and S maps are shown below. For the Rv map, we see that dimension one explains 34% of the variance in the data while dimension two explains 8%. For the S map, dimension one explains 24% of the the variance while dimension two explains 9%.


RV−map: Scree
Percentage of Explained Variance
0
10
20
30
40
Inertia Extracted by the Components
0
2
4
6
5
10
15
Dimensions

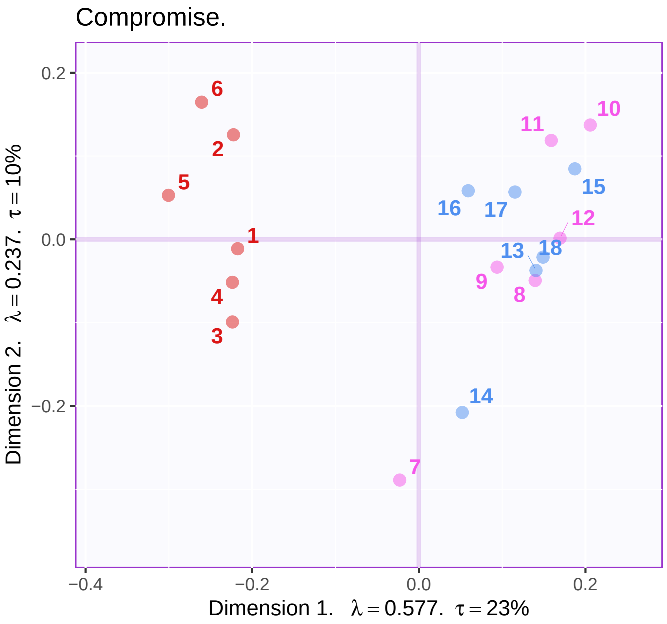

Compromise: Explained Variance per Dimension
Percentage of Explained Variance
0
5
10
15
20
25
30
Inertia Extracted by the Components
0.0
0.2
0.4
0.6
5
10
15
Dimensions

7.5 Factor Scores

The RV and S factor score maps are shown below. We see a strong positive contribution on dimension two on the Rv map from Judges 7, 8, and 12, and strong negative contribution from judges 14, 13, and 10. The judges mostly agreed along dimension one. For the S map, dimension one explains the variance between red wines vs rose and white wines, while dimension two there was no clear seperation of wines based on type and the variance here is likely due to the differences in judges grouping the wines.



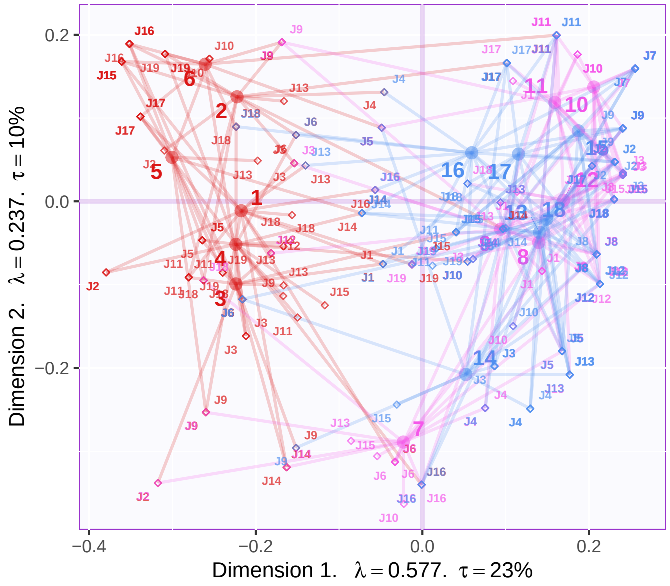
```
## [1] Bootstrap On Factor Scores. Iterations #:
## [2] 1000
```



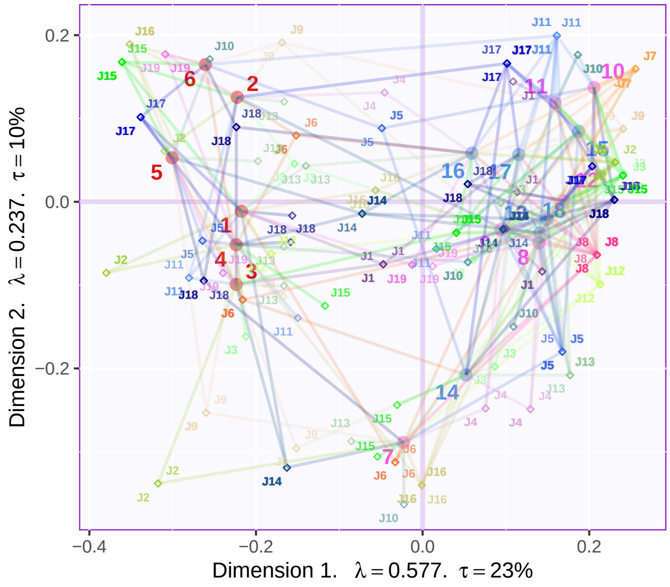
7.6 Partial Factor Scores

The Global factor scores along with the partial factor scores are computed below. We see that the judges seem to be explaining an equal amount of variance, and the judges are dispersed relatively evenly across the map.

Compromise.



Compromise.



7.7 Conclusion

The variance dimension one explains is attributed to the difference between red vs. white and rose wines. The judges mostly agreed along dimension one. The second dimension is most likely attributed to the differences in judges 13, 14, and 10 vs. 8, 7, and 12, as there was not any observable separation between white and rose wines along dimension two.

Chapter 8

Multiple Factor Analysis

8.1 Introduction

Multiple Factor analysis (MFA) is used to analyze more than one sets of variables by seeking the common structure in these sets. MFA utilizes two steps. First a PCA is performed on each set, and each observation of the sets are divided by the square root of the first eigenvalue obtained in the PCA. Second, these new "normalized" data sets are merged to form a a matrix and a global PCA is performed on this matrix. The original data sets are then projected onto the global PCA for the analysis of the commanalities.

Reference: Abdi, H., & Valentin, D. (2007). Multiple factor analysis (MFA). Encyclopedia of measurement and statistics, 1-14.

8.2 Data Set

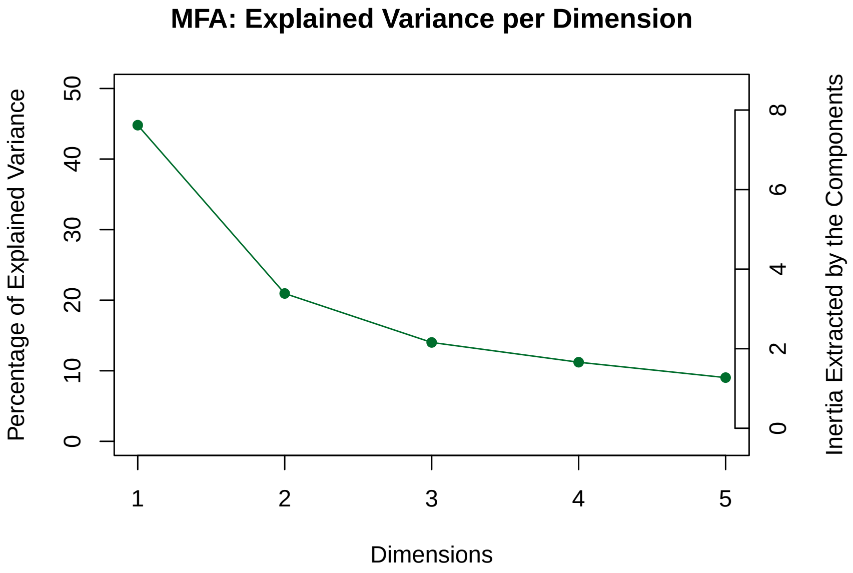
Here we import the data set and compute the necessary parameters for the MFA. The data is the ratings of 19 judges on 18 (6 red, 6 white, and 6 rose wines) different wines, rating scaled from 1 to 9 in this data set (with a presumable maximum rating of 10). The first observations from judge one are shown.

```
## New names:
## * ‘‘ -> ...1
```

Aroma impact.1	Flavor impact.1	Sweet.1	Acid.1	Bitter.1	Astringent.1	Mettalic.1	Carbo
6.8	7.0	0.2	0.4	5.3	4.8	0.2	
4.9	5.7	0.2	0.6	5.6	5.0	0.1	
4.7	4.3	0.5	1.1	2.7	3.4	5.2	
7.6	7.0	0.3	1.0	5.3	6.2	3.3	
4.0	6.7	3.0	2.0	1.8	1.2	2.0	
3.0	4.1	2.1	2.2	1.9	1.1	0.9	

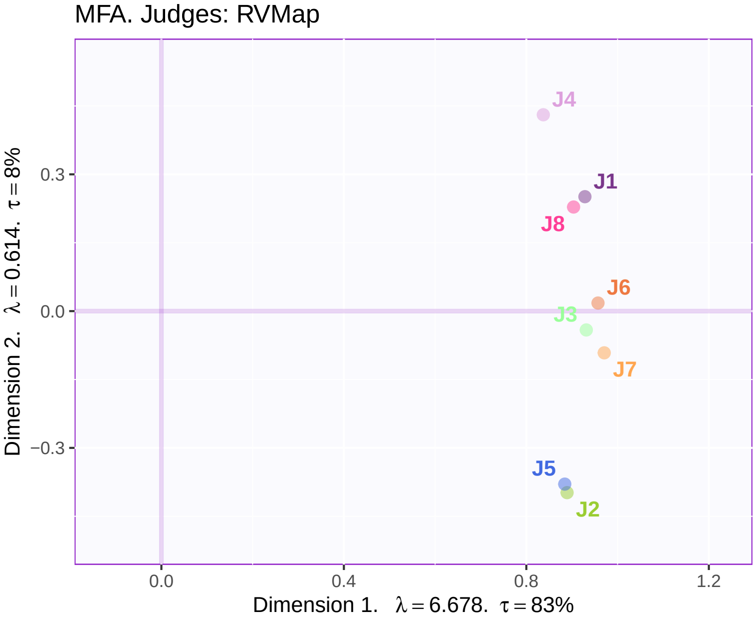
8.3 Scree Plot

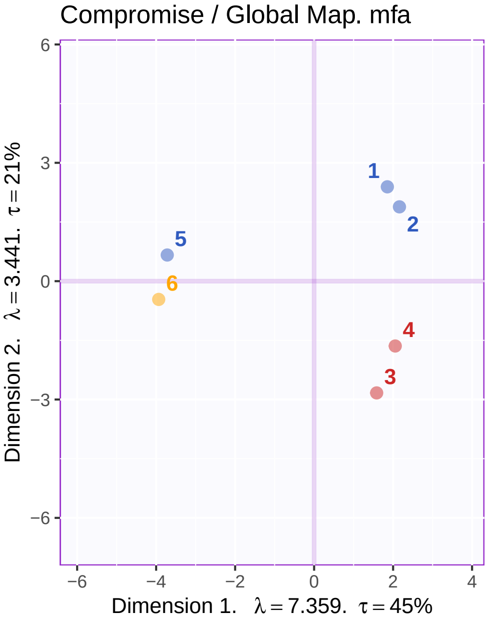
The Scree plot for the MFA is shown below. We see that dimension one explains roughly 45% of the variance in the data while dimension two explains roughly 20%. The eigenvalues from the MFA were used to determine the component with highest variance in the scree plot.



8.4 Row Factor Scores

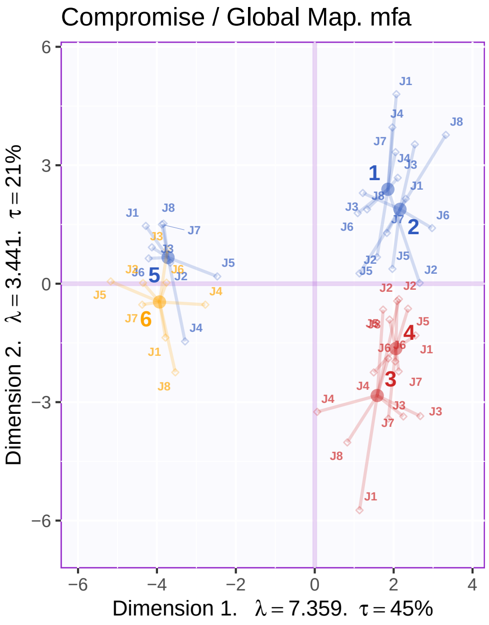
The row (beers) factor scores are shown here. We see a clear separation of scores along dimension one for traditional beers (blue moon and corona) vs. proposal beers and non alcoholic beers. This means we can infer that the variance explained by dimension one is due to this discrepancy between beers. Dimension two shows the separation between alcoholic and non-alcoholic beers. The factor scores map for the judges is also shown, and we see agreeable scores for the judges along dimension one, while along dimension two we have a separation of judges with judges 5 and 2 making large negative contributions and judges 4,1, and 8 making large positive contributions.

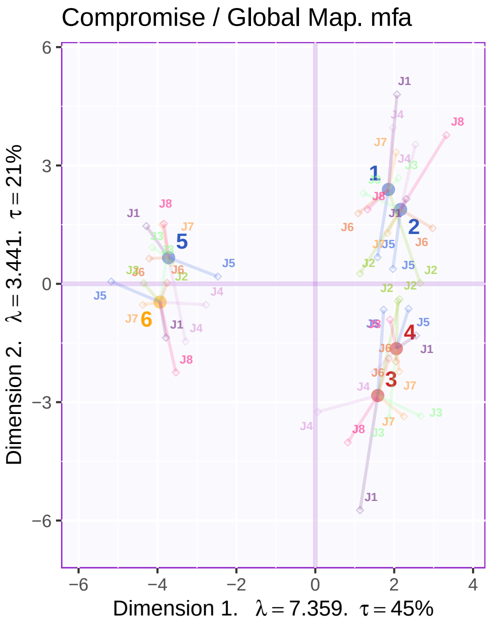




8.5 Partial Factor Scores

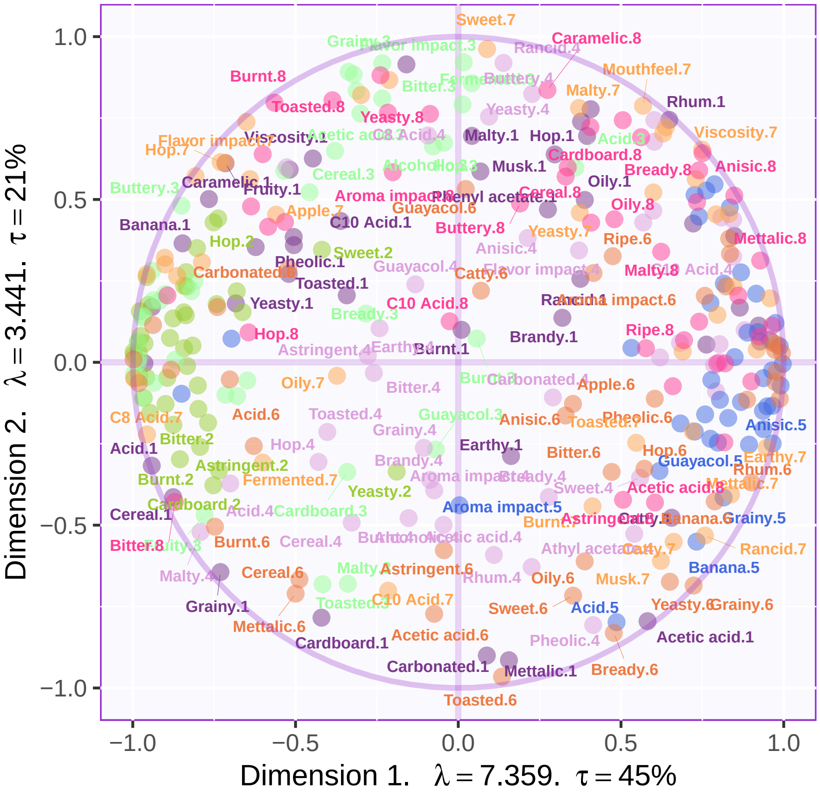
Here the partial factor scores are shown for the MFA. The lines were made slightly transparent in order to improve readability. We see judge 1 contributes to large positive and negative contributions to alcoholic and non-alcoholic beers respectively, accounting for a large variance. We see the same effect by judge 8 but to a lesser extent.

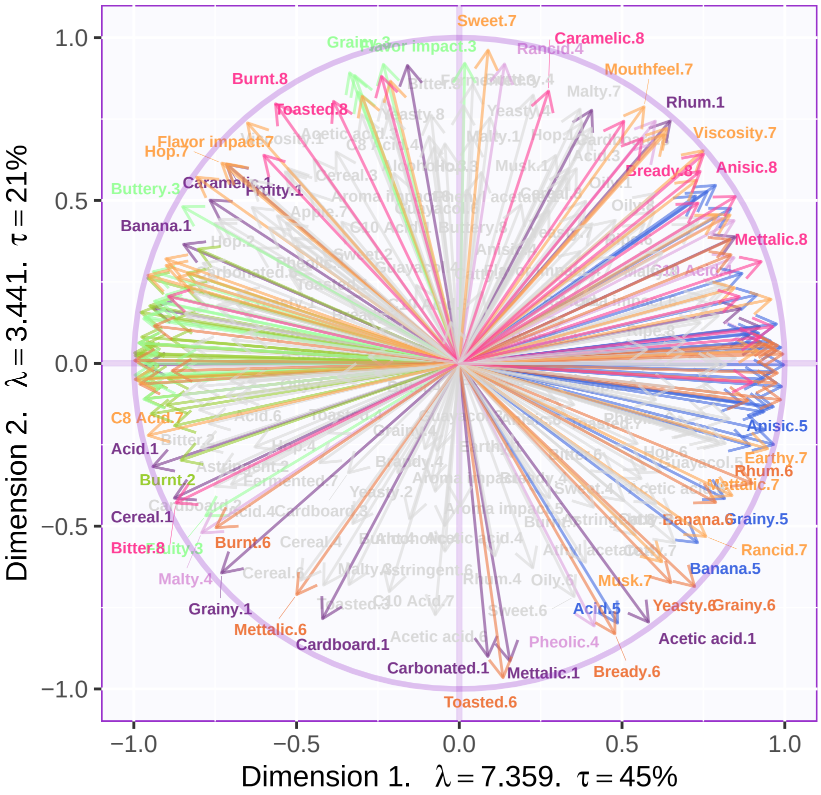




8.6 Circle of Correlation

Here the correlation circle from the MFA is displayed. Mostly contributions from judge 1 and 6 showed correlations that surpassed the threshold correlation level of 0.75, agreeing with the distributions seen in the factor scores map.


Dimension 1. λ = 7.359. τ = 45%
Dimension 2. λ = 3.441. τ = 21%
1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0
Sweet.7
Caramelic.8
Grainy.3
Flavor impact.3
Rancid.4
Mouthfeel.7
Burnt.8
Bitter.3
Fermented.3
Buttery.4
Malty.7
Rhum.1
Toasted.8
Yeasty.8
Yeasty.4
Flavor impact.7
Viscosity.1
Acetic acid.3
C8 Acid.4
Malty.1
Hop.1
Acid.3
Viscosity.7
Hop.7
Cardboard.8
Alcohol.3
Hop.3
Musk.1
Anisic.8
Bready.8
Cereal.3
Caramelic.1
Oily.1
Buttery.3
Fruity.1
Aroma impact.8
Phenyl acetate.1
Cereal.8
Apple.7
Guayacol.6
Oily.8
Banana.1
C10 Acid.1
Buttery.8
Yeasty.7
Hop.2
Ripe.6
Mettalic.8
Anisic.4
Sweet.2
Pheolic.1
Guayacol.4
Carbonated.6
Catty.6
Flavor impact.4
Malty.8
C10 Acid.4
Toasted.1
Yeasty.1
C10 Acid.8
Rancid.1
Aroma impact.6
Bready.3
Hop.8
Brandy.1
Ripe.8
Astringent.4
Earthy.4
Burnt.1
Oily.7
Burnt.3
Carbonated.4
Apple.6
Bitter.4
C8 Acid.7
Acid.6
Toasted.4
Guayacol.3
Anisic.6
Toasted.7
Pheolic.6
Anisic.5
Bitter.2
Grainy.4
Acid.1
Hop.4
Earthy.1
Bitter.6
Hop.6
Earthy.7
Astringent.2
Brandy.4
Guayacol.5
Rhum.6
Burnt.2
Fermented.7
Aroma impact.4
Bready.4
Sweet.4
Acetic acid.8
Mettalic.7
Yeasty.2
Cardboard.2
Aroma impact.5
Acid.4
Cardboard.3
Burnt.7
Astringent.8
Catty.1
Banana.6
Grainy.5
Cereal.1
Burnt.4
Alcohol.4
Acetic acid.4
Bitter.8
Fruity.3
Burnt.6
Cereal.4
Athyl acetate.4
Catty.7
Rancid.7
Malty.4
Cereal.6
Malty.3
Astringent.6
Rhum.4
Oily.6
Musk.7
Banana.5
Grainy.1
Toasted.3
C10 Acid.7
Sweet.6
Acid.5
Yeasty.6
Grainy.6
Mettalic.6
Acetic acid.6
Pheolic.4
Acetic acid.1
Cardboard.1
Carbonated.1
Mettalic.1
Bready.6
Toasted.6



8.7 Conclusion

Dimension one: Difference between traditional beers and proposal beers meant to mimic non-alcoholic beers. Judges scores agreed along dimension one. Dimension two: Non alcoholic vs alcoholic beers. Judge 4 vs. Judge 5 and 2.