Exercise 1

Tobias Raidl, 11717659

2023-10-13

Setup Dataset

Standardizing the dataset (not the type column) in order to make means and scales meaningful.

```
library(pgmm)
library(dplyr)
```

```
## Warning: Paket 'dplyr' wurde unter R Version 4.1.3 erstellt

##
## Attache Paket: 'dplyr'

## Die folgenden Objekte sind maskiert von 'package:stats':
##
##     filter, lag

## Die folgenden Objekte sind maskiert von 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
data(wine)
wine_scaled = data.frame(scale(dplyr::select(wine, -Type)))
wine = cbind("Type"=wine$Type, wine_scaled)
```

1.

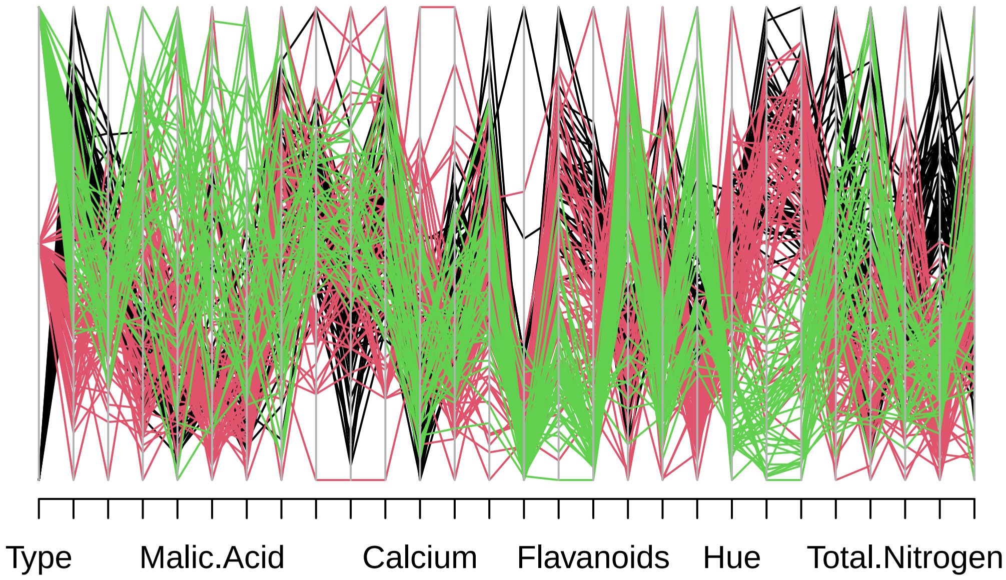
By using a parallel coordinate system, colored according to each observations TYPE, you can estimate the explanatory power of each variable according to its TYPE. If the colored lines are clustered well it corresponds to a high explanatory power. This only holds if TYPE is perceived as dependent variable.

```
library(MASS)
```

```
##
## Attache Paket: 'MASS'

## Das folgende Objekt ist maskiert 'package:dplyr':
##
##     select
```

```
parcoord(wine, col=wine$Type)
```



2.

The two variables with the highest difference in mean when aggregated by TYPE are Flavanoids and OD280.OD315.of.Diluted.Wines. I previously scaled the means for each variable so that it actually conveys information. Therefore it makes sense to compare the means.

```
grouped_means = wine %>%
  group_by(Type) %>%
  summarise_all(list(mean))
grouped_means
```

```
## # A tibble: 3 x 28
##    Type Alcohol Sugar.free.Extract Fixed.Acidity Tartaric.Acid Malic.Acid
##   <dbl>   <dbl>              <dbl>         <dbl>         <dbl>      <dbl>
## 1     1   0.917              0.661        -0.477        -0.568     -0.292
## 2     2  -0.889             -0.471        -0.246        -0.171     -0.361
## 3     3   0.189             -0.116         0.950         0.951      0.893
## # i 22 more variables: Uronic.Acids <dbl>, pH <dbl>, Ash <dbl>,
## #   Alcalinity.of.Ash <dbl>, Potassium <dbl>, Calcium <dbl>, Magnesium <dbl>,
## #   Phosphate <dbl>, Chloride <dbl>, Total.Phenols <dbl>, Flavanoids <dbl>,
## #   Non.flavanoid.Phenols <dbl>, Proanthocyanins <dbl>, Color.Intensity <dbl>,
## #   Hue <dbl>, OD280.OD315.of.Diluted.Wines <dbl>,
## #   OD280.OD315.of.Flavanoids <dbl>, Glycerol <dbl>, X2.3.Butanediol <dbl>,
## #   Total.Nitrogen <dbl>, Proline <dbl>, Methanol <dbl>
```

```
grouped_means_diff = grouped_means %>%
  summarise_all(list(function(x) diff(range(x)))) %>%
  dplyr::select(-Type)
grouped_means_diff
```

```
## # A tibble: 1 x 27
##   Alcohol Sugar.free.Extract Fixed.Acidity Tartaric.Acid Malic.Acid Uronic.Acids
##     <dbl>              <dbl>         <dbl>         <dbl>      <dbl>        <dbl>
## 1    1.81               1.13          1.43          1.52       1.25         1.44
## # i 21 more variables: pH <dbl>, Ash <dbl>, Alcalinity.of.Ash <dbl>,
## #   Potassium <dbl>, Calcium <dbl>, Magnesium <dbl>, Phosphate <dbl>,
## #   Chloride <dbl>, Total.Phenols <dbl>, Flavanoids <dbl>,
## #   Non.flavanoid.Phenols <dbl>, Proanthocyanins <dbl>, Color.Intensity <dbl>,
## #   Hue <dbl>, OD280.OD315.of.Diluted.Wines <dbl>,
## #   OD280.OD315.of.Flavanoids <dbl>, Glycerol <dbl>, X2.3.Butanediol <dbl>,
## #   Total.Nitrogen <dbl>, Proline <dbl>, Methanol <dbl>
```

3.

The highest difference in variance in between groups can be observed in the variable Chloride Tartaric.Acid. These values convey information aswell, as standardization has been done previously.

```
grouped_variances = wine %>%
  group_by(Type) %>%
  summarise_all(list(var))
grouped_variances
```

```
## # A tibble: 3 x 28
##    Type Alcohol Sugar.free.Extract Fixed.Acidity Tartaric.Acid Malic.Acid
##   <dbl>   <dbl>              <dbl>         <dbl>         <dbl>      <dbl>
## 1     1   0.324              0.570         0.362         0.258      0.380
## 2     2   0.439              0.770         0.888         0.460      0.826
## 3     3   0.427              1.02          0.698         1.39       0.948
## # i 22 more variables: Uronic.Acids <dbl>, pH <dbl>, Ash <dbl>,
## #   Alcalinity.of.Ash <dbl>, Potassium <dbl>, Calcium <dbl>, Magnesium <dbl>,
## #   Phosphate <dbl>, Chloride <dbl>, Total.Phenols <dbl>, Flavanoids <dbl>,
## #   Non.flavanoid.Phenols <dbl>, Proanthocyanins <dbl>, Color.Intensity <dbl>,
## #   Hue <dbl>, OD280.OD315.of.Diluted.Wines <dbl>,
## #   OD280.OD315.of.Flavanoids <dbl>, Glycerol <dbl>, X2.3.Butanediol <dbl>,
## #   Total.Nitrogen <dbl>, Proline <dbl>, Methanol <dbl>
```

```
grouped_variances_diff = grouped_variances %>%
  summarise_all(list(function(x) diff(range(x)))) %>%
  dplyr::select(-Type)
grouped_variances_diff
```

```
## # A tibble: 1 x 27
##   Alcohol Sugar.free.Extract Fixed.Acidity Tartaric.Acid Malic.Acid Uronic.Acids
##     <dbl>              <dbl>         <dbl>         <dbl>      <dbl>        <dbl>
## 1   0.115              0.446         0.526          1.13      0.568        0.922
## # i 21 more variables: pH <dbl>, Ash <dbl>, Alcalinity.of.Ash <dbl>,
## #   Potassium <dbl>, Calcium <dbl>, Magnesium <dbl>, Phosphate <dbl>,
## #   Chloride <dbl>, Total.Phenols <dbl>, Flavanoids <dbl>,
## #   Non.flavanoid.Phenols <dbl>, Proanthocyanins <dbl>, Color.Intensity <dbl>,
## #   Hue <dbl>, OD280.OD315.of.Diluted.Wines <dbl>,
## #   OD280.OD315.of.Flavanoids <dbl>, Glycerol <dbl>, X2.3.Butanediol <dbl>,
## #   Total.Nitrogen <dbl>, Proline <dbl>, Methanol <dbl>
```

4.

Both of the variables seem to cluster well, therefore provide good explanatory power for the Type variable if Type is said to be dependent. They would probably fit well for fitting a model in order to predict Type values of unknown data.

```r
highest_diff_variable =
↪  colnames(grouped_means_diff)[apply(grouped_means_diff,1,which.max)]
temp = grouped_means_diff %>%
  dplyr::select(-highest_diff_variable)
```

Mean difference

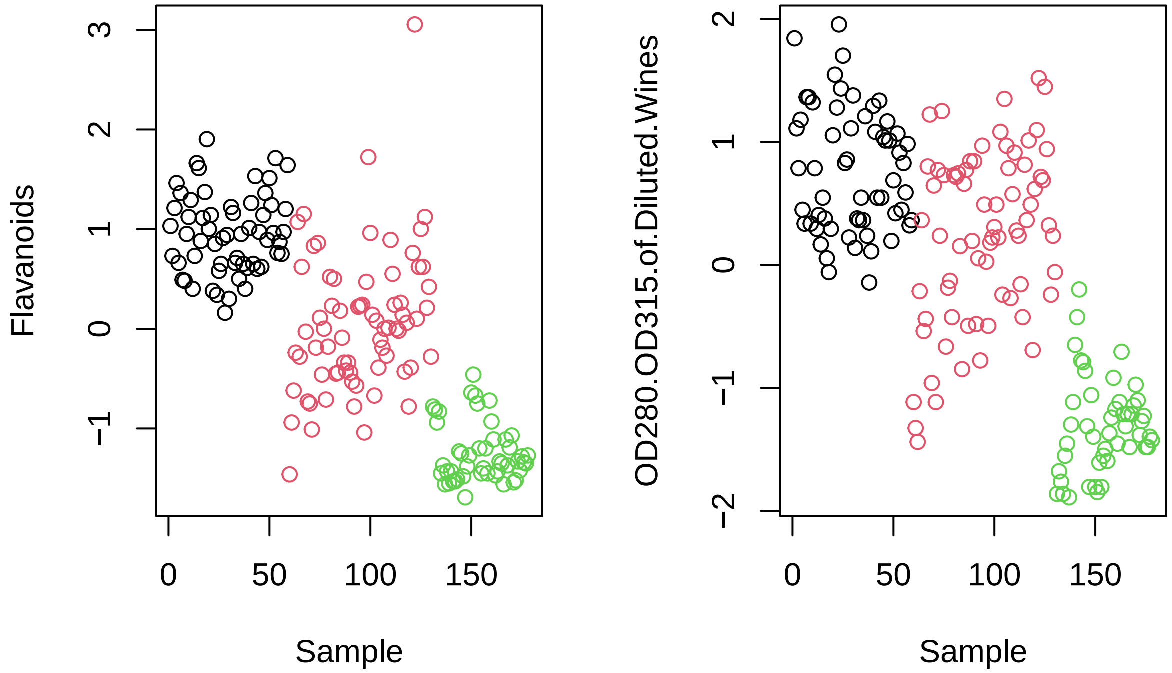
```
## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.
## i Please use `all_of()` or `any_of()` instead.
##   # Was:
##   data %>% select(highest_diff_variable)
##
##   # Now:
##   data %>% select(all_of(highest_diff_variable))
##
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```r
second_highest_diff_variable = colnames(temp)[apply(temp,1,which.max)]

cat(paste("Variables with highest mean difference grouped by Type are:\n",
↪  highest_diff_variable, "\n", second_highest_diff_variable))
```

```
## Variables with highest mean difference grouped by Type are:
##  Flavanoids
##  OD280.OD315.of.Diluted.Wines
```

```
par(mfrow = c(1, 2))
plot(y = wine[[highest_diff_variable]], x = seq(1: nrow(wine)), col=wine$Type,
ylab = highest_diff_variable, xlab = "Sample")
plot(y = wine[[second_highest_diff_variable]], x = seq(1:nrow(wine)), col=wine$Type,
ylab = second_highest_diff_variable, xlab = "Sample")
```



#### Variance difference

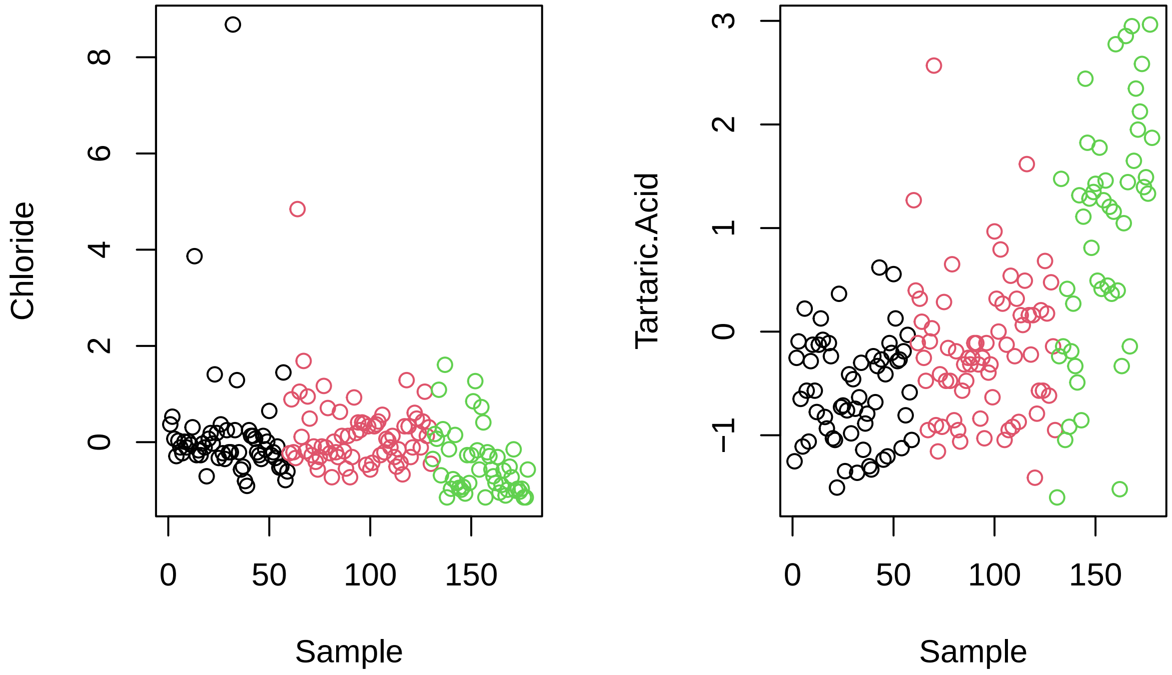
```
highest_diff_variable =
↪  colnames(grouped_variances_diff)[apply(grouped_variances_diff,1,which.max)]
temp = grouped_variances_diff %>%
  dplyr::select(-highest_diff_variable)
second_highest_diff_variable = colnames(temp)[apply(temp,1,which.max)]

cat(paste("Variables with highest var difference grouped by Type are:\n",
↪  highest_diff_variable, "\n", second_highest_diff_variable))
```

```
## Variables with highest var difference grouped by Type are:
##  Chloride
##  Tartaric.Acid
```

```
par(mfrow = c(1, 2))
plot(y = wine[[highest_diff_variable]], x = seq(1: nrow(wine)), col=wine$Type,
ylab = highest_diff_variable, xlab = "Sample")
```

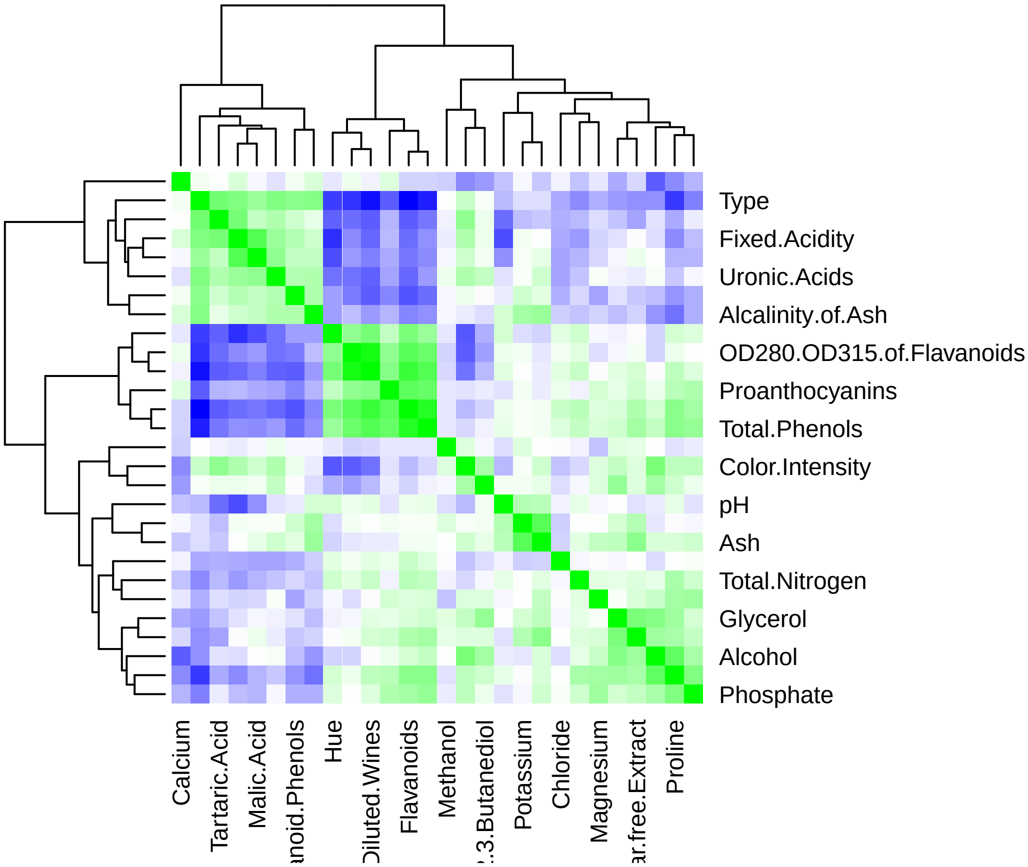
```r
plot(y = wine[[second_highest_diff_variable]], x = seq(1:nrow(wine)), col=wine$Type,
ylab = second_highest_diff_variable, xlab = "Sample")
```



5.

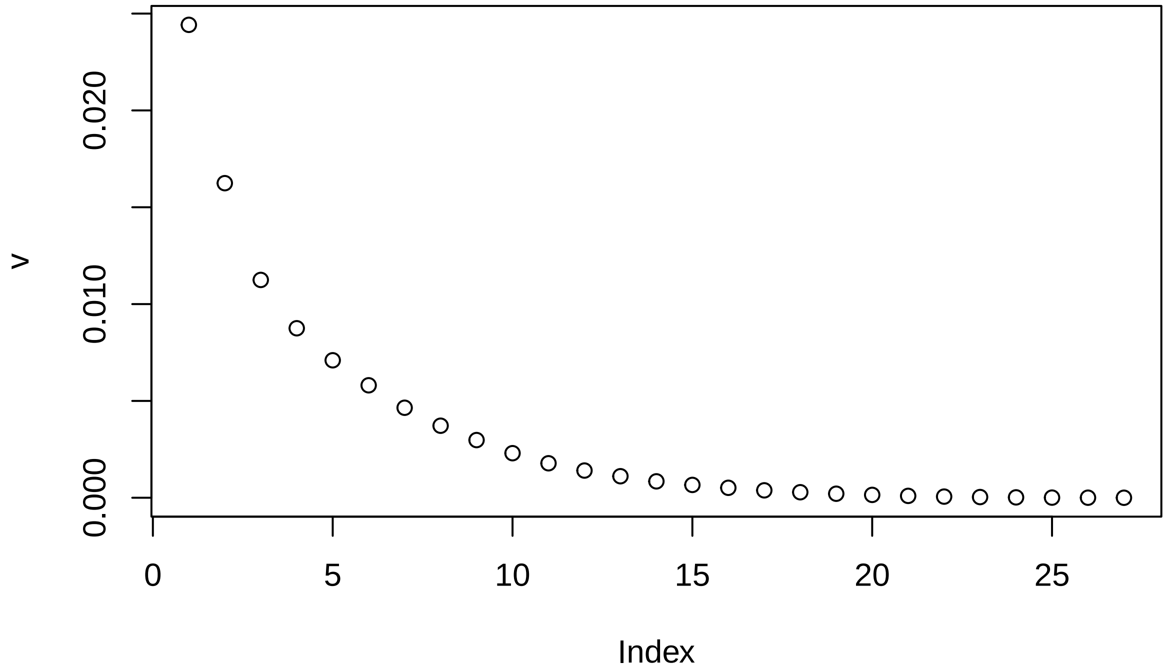
I chose a color palette with 3 different hue marks, as the interval to be mapped in color is correlation. The first mar (1) is mapped to blue, the second mark (0) is mapped to white and the third mark (-1) to green. 1 corresponds with strong positive correlation, 0 with no correlation and -1 with strong negative correlation.

```r
cor_mat = cor(wine)
#print(cor_mat)
heatmap(cor_mat, symm = TRUE,
col = colorRampPalette(c("blue", "white", "green"))(100))
```



6.

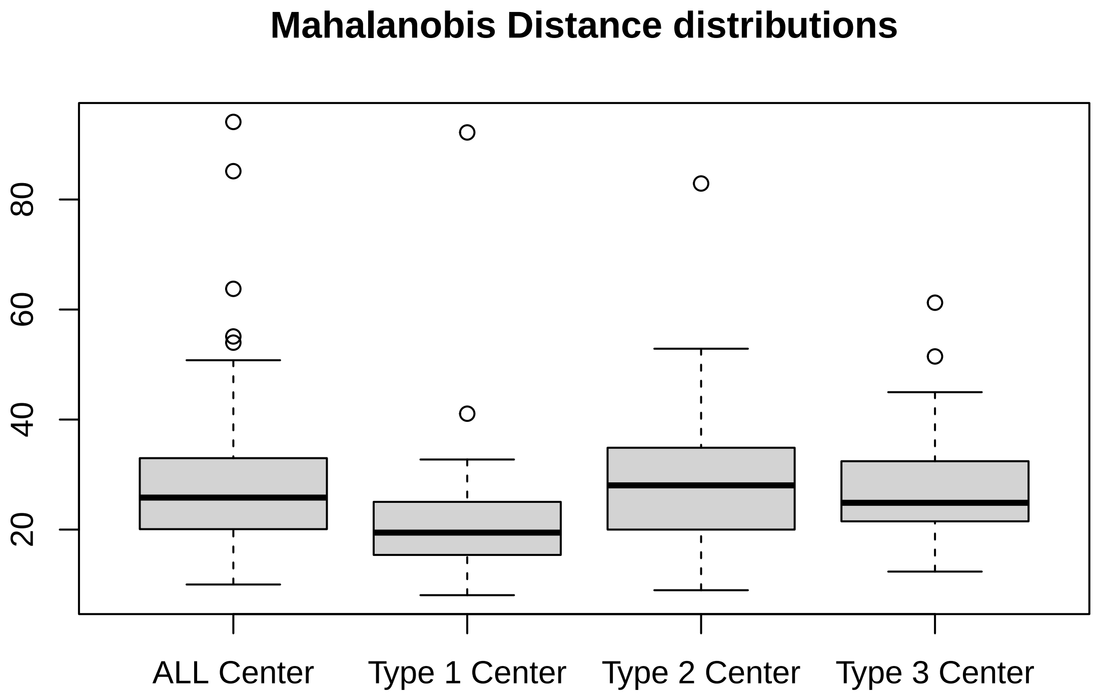
```
eigen_error = function(k){
  temp = eigen(cor_mat)
  eigenvalues = temp$values[1:k]
  eigenvectors = temp$vectors[,1:k]
  eigenvalue_matrix = diag(eigenvalues)
  eigenvector_matrix = rbind(eigenvectors)
  recycled_cor_mat = eigenvector_matrix %*% eigenvalue_matrix %*% t(eigenvector_matrix)
  error = mean((cor_mat-recycled_cor_mat)^2)
  return(error)
}
error = eigen_error(5)
v = 2:ncol(wine)
v = unlist(lapply(v, FUN=eigen_error))
plot(v)
```



7

Type 1 has a lower mean. This could be a result of Type 1 observations generally lying inbetween Type 2 and Type 3 on most variables. It could also be caused by too few observations being not able to estimate the underlying distributions properly.

```
types <- split(wine, wine$Type)
cov_mat = cov(wine)
maha <- c(
as.data.frame(mahalanobis(wine, colMeans(wine), cov(wine))),
as.data.frame(mahalanobis(types[[1]], colMeans(types[[1]]), cov_mat)),
as.data.frame(mahalanobis(types[[2]], colMeans(types[[2]]), cov_mat)),
as.data.frame(mahalanobis(types[[3]], colMeans(types[[3]]), cov_mat))
)
boxplot(maha, main="Mahalanobis Distance distributions", names = c('ALL Center', 'Type 1
↪  Center', 'Type 2 Center', 'Type 3 Center'))
```


Mahalanobis Distance distributions
80
60
40
20
ALL Center
Type 1 Center
Type 2 Center
Type 3 Center