Multivariate Analysis 1 Final Project

Yi Chen

5/1/2020

## Introdcution

This project will show the example of using R to carry out some simple multivariate analyses, with a focus on principal component analysis (PCA) and linear discriminate analysis (LDA). Whole project will guide the reader from the exploratory data analysis to more advanced statistical methodologies. For every technique covered in this project, the corresponding research question, method theory (including assumption and limitation), and R code will be provided.

## Data Set

The data sets in this project come from the UCI Machine Learning Repository (<http://archive.ics.uci.edu/ml>). In particular, the wine recognition data (<http://archive.ics.uci.edu/ml/machine-learning-databases/wine/wine.data>) is updated at Sept 21, 1998 by C.Blake. The original source of the data set comes from Forina, M. et al (1991).

The observations in the data set are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines.

## load the data  
  
wine <- read.table("http://archive.ics.uci.edu/ml/machine-learning-databases/wine/wine.data",sep=",")  
colnames(wine) <- c("type", "Alcohol","Malic acid","Ash","Alcalinity","Magnesium","Phenols","Flavanoids","Nonflavanoid phenols","Proanthocyanins","Color intensity","Hue","OD280/OD315","Proline")  
head(wine,5)

## type Alcohol Malic acid Ash Alcalinity Magnesium Phenols Flavanoids  
## 1 1 14.23 1.71 2.43 15.6 127 2.80 3.06  
## 2 1 13.20 1.78 2.14 11.2 100 2.65 2.76  
## 3 1 13.16 2.36 2.67 18.6 101 2.80 3.24  
## 4 1 14.37 1.95 2.50 16.8 113 3.85 3.49  
## 5 1 13.24 2.59 2.87 21.0 118 2.80 2.69  
## Nonflavanoid phenols Proanthocyanins Color intensity Hue OD280/OD315 Proline  
## 1 0.28 2.29 5.64 1.04 3.92 1065  
## 2 0.26 1.28 4.38 1.05 3.40 1050  
## 3 0.30 2.81 5.68 1.03 3.17 1185  
## 4 0.24 2.18 7.80 0.86 3.45 1480  
## 5 0.39 1.82 4.32 1.04 2.93 735

dim(wine)

## [1] 178 14

summary(wine)

## type Alcohol Malic acid Ash   
## Min. :1.000 Min. :11.03 Min. :0.740 Min. :1.360   
## 1st Qu.:1.000 1st Qu.:12.36 1st Qu.:1.603 1st Qu.:2.210   
## Median :2.000 Median :13.05 Median :1.865 Median :2.360   
## Mean :1.938 Mean :13.00 Mean :2.336 Mean :2.367   
## 3rd Qu.:3.000 3rd Qu.:13.68 3rd Qu.:3.083 3rd Qu.:2.558   
## Max. :3.000 Max. :14.83 Max. :5.800 Max. :3.230   
## Alcalinity Magnesium Phenols Flavanoids   
## Min. :10.60 Min. : 70.00 Min. :0.980 Min. :0.340   
## 1st Qu.:17.20 1st Qu.: 88.00 1st Qu.:1.742 1st Qu.:1.205   
## Median :19.50 Median : 98.00 Median :2.355 Median :2.135   
## Mean :19.49 Mean : 99.74 Mean :2.295 Mean :2.029   
## 3rd Qu.:21.50 3rd Qu.:107.00 3rd Qu.:2.800 3rd Qu.:2.875   
## Max. :30.00 Max. :162.00 Max. :3.880 Max. :5.080   
## Nonflavanoid phenols Proanthocyanins Color intensity Hue   
## Min. :0.1300 Min. :0.410 Min. : 1.280 Min. :0.4800   
## 1st Qu.:0.2700 1st Qu.:1.250 1st Qu.: 3.220 1st Qu.:0.7825   
## Median :0.3400 Median :1.555 Median : 4.690 Median :0.9650   
## Mean :0.3619 Mean :1.591 Mean : 5.058 Mean :0.9574   
## 3rd Qu.:0.4375 3rd Qu.:1.950 3rd Qu.: 6.200 3rd Qu.:1.1200   
## Max. :0.6600 Max. :3.580 Max. :13.000 Max. :1.7100   
## OD280/OD315 Proline   
## Min. :1.270 Min. : 278.0   
## 1st Qu.:1.938 1st Qu.: 500.5   
## Median :2.780 Median : 673.5   
## Mean :2.612 Mean : 746.9   
## 3rd Qu.:3.170 3rd Qu.: 985.0   
## Max. :4.000 Max. :1680.0

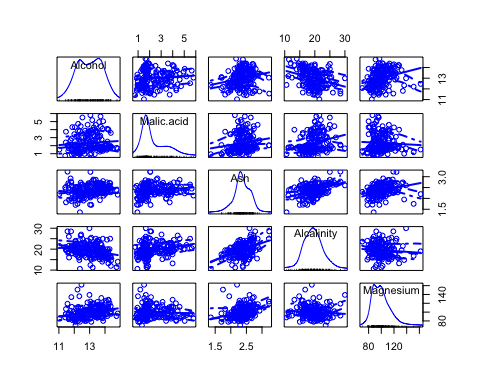
As we can see, the first column type indicates which wine type the sample belongs to. There are 178 samples with 13 distinct attributes. All 13 attributes are continuous variables. There is no missing value in the data set.

## EDA through Visualization

#### Matrix Scatter Plot

In multivariate analysis, a common way to understand the structure of the data set through visulization is matrix scatterplot. For simplicity, I only use the first 5 attribution in the plot.

library(car)  
scatterplotMatrix(wine[2:6])

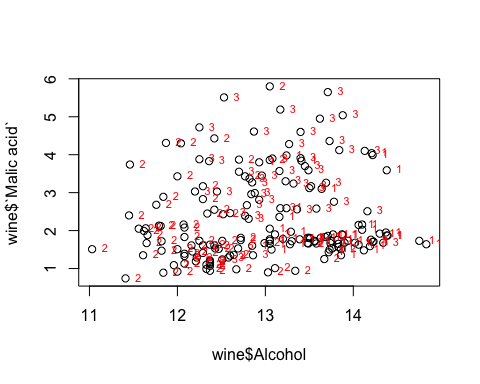


In this matrix scatterplot, the diagonal cells show histograms of each of the variables, in this case the concentrations of the first five chemicals. Each of the off-diagonal cells is a scatterplot of two of the five chemicals. As we can see, there is a certain level of correlation between the pair of the attributes. Meanwhile, the distribution of most attributes approximately follow the normal distribution. But the distribution of Alcohol and Malic.acid is more different from the normal distribution.

#### Scatter Plot with group label.

Using the scatter plot give us a more detail information about how two attribute are correlated with each other. In particular, I label each simple points with their corresponding wine type. In this way, we can see who the data allocate in different two dimensional space across different wine types. I choose the attribute of alcohol and malic acid as an example.

plot(wine$Alcohol, wine$`Malic acid`)  
text(wine$Alcohol, wine$`Malic acid`, wine$type, cex=0.7, pos=4, col="red")



In general, there is already a pattern in classification. The second type of wine are located at the range with lower value of alcohol. For the first and third type of wine, they are located at the space with high alcohol but low malic acid and high malic acid. However, there are many points from different types of wine overlap with each other. This indicates that there is a limiation in using just two dimension space for classifying the wine type.

## Discriptive Analysis

#### mean and standard deviation of each attribute

In order to have a deeper understanding of the data set. The first thing we need to know is the basic discriptive statistics for each attribution.

round(sapply(wine[2:14], mean),2)

## Alcohol Malic acid Ash   
## 13.00 2.34 2.37   
## Alcalinity Magnesium Phenols   
## 19.49 99.74 2.30   
## Flavanoids Nonflavanoid phenols Proanthocyanins   
## 2.03 0.36 1.59   
## Color intensity Hue OD280/OD315   
## 5.06 0.96 2.61   
## Proline   
## 746.89

round(sapply(wine[2:14], sd),2)

## Alcohol Malic acid Ash   
## 0.81 1.12 0.27   
## Alcalinity Magnesium Phenols   
## 3.34 14.28 0.63   
## Flavanoids Nonflavanoid phenols Proanthocyanins   
## 1.00 0.12 0.57   
## Color intensity Hue OD280/OD315   
## 2.32 0.23 0.71   
## Proline   
## 314.91

Clearly, we can see there is a big difference in the scale of 13 attribute. This indicates that there is a need for normalize the data before the analysis. Meanwhile, scale the data also reshape the distribution of the data so that it is closer to the normal distribution, which is the assumption for many multivariate analysis models.

#### mean and standard deviation difference across wine type

Secondly, we may also interested in explore the difference of mean and standard deviation across different groups.

printMeanAndSdByGroup <- function(variables,groupvariable)  
 {  
 # find the names of the variables  
 variablenames <- c(names(groupvariable),names(as.data.frame(variables)))  
 # within each group, find the mean of each variable  
 groupvariable <- groupvariable[,1] # ensures groupvariable is not a list  
 means <- aggregate(as.matrix(variables) ~ groupvariable, FUN = mean)  
 names(means) <- variablenames  
 print(paste("Means:"))  
 print(means)  
 # within each group, find the standard deviation of each variable:  
 sds <- aggregate(as.matrix(variables) ~ groupvariable, FUN = sd)  
 names(sds) <- variablenames  
 print(paste("Standard deviations:"))  
 print(sds)  
}  
  
printMeanAndSdByGroup(wine[2:14],wine[1])

## [1] "Means:"  
## type Alcohol Malic acid Ash Alcalinity Magnesium Phenols Flavanoids  
## 1 1 13.74475 2.010678 2.455593 17.03729 106.3390 2.840169 2.9823729  
## 2 2 12.27873 1.932676 2.244789 20.23803 94.5493 2.258873 2.0808451  
## 3 3 13.15375 3.333750 2.437083 21.41667 99.3125 1.678750 0.7814583  
## Nonflavanoid phenols Proanthocyanins Color intensity Hue OD280/OD315  
## 1 0.290000 1.899322 5.528305 1.0620339 3.157797  
## 2 0.363662 1.630282 3.086620 1.0562817 2.785352  
## 3 0.447500 1.153542 7.396250 0.6827083 1.683542  
## Proline  
## 1 1115.7119  
## 2 519.5070  
## 3 629.8958  
## [1] "Standard deviations:"  
## type Alcohol Malic acid Ash Alcalinity Magnesium Phenols Flavanoids  
## 1 1 0.4621254 0.6885489 0.2271660 2.546322 10.49895 0.3389614 0.3974936  
## 2 2 0.5379642 1.0155687 0.3154673 3.349770 16.75350 0.5453611 0.7057008  
## 3 3 0.5302413 1.0879057 0.1846902 2.258161 10.89047 0.3569709 0.2935041  
## Nonflavanoid phenols Proanthocyanins Color intensity Hue OD280/OD315  
## 1 0.07004924 0.4121092 1.2385728 0.1164826 0.3570766  
## 2 0.12396128 0.6020678 0.9249293 0.2029368 0.4965735  
## 3 0.12413959 0.4088359 2.3109421 0.1144411 0.2721114  
## Proline  
## 1 221.5208  
## 2 157.2112  
## 3 115.0970

As an example, I will compare the mean difference between the first and second wine type. There is small mean different between the first and second wine type in terms of hue and ash, but big difference in terms of profline and color density. This indicates that different attribute has different power in distinguish the wine type. For example, ash and nonflavanoid phenols are simiar across different types of wine. But flavanoids is much lower in the thrid type of wine than the other two type.

#### Between-groups Variance and Within-groups Variance for a Variable

Further, we can calculate the within group variance, between group variance, and Separation of each variable. Similar to the idea in ANOVA, these information help us to identify which variable is more useful in classify the wine type and which varible is more consistant across different wine type.

The within group variance can be calculated using the formula:

, where g is the number of group, n is the number of total sample.

WithinGroupsVariance <- function(variable,groupvariable){  
 n\_total <- length(variable)  
 n\_group <- length(unique(groupvariable))  
 total\_variance <- 0  
 for (g in 1:n\_group){  
 group\_data <- variable[groupvariable==g]  
 group\_mean <- mean(group\_data)  
 for (i in group\_data){  
 total\_variance <- total\_variance + (i - group\_mean)^2  
 }  
 }  
 return(total\_variance/(n\_total-n\_group))  
}  
WithinGroupsVariance(wine$Alcohol,wine$type)

## [1] 0.2620525

As the example, the within group variande for alcohol is about 0.26. Simiarly, we can calculate the between group variance using the formula below,

, where represent the number of sample in group and is the group mean.

BetweenGroupsVariance <- function(variable,groupvariable){  
 n\_group <- length(unique(groupvariable))  
 grand\_mean <- mean(variable)  
 total\_variance <- 0  
 for (g in 1:n\_group){  
 group\_data <- variable[groupvariable==g]  
 group\_mean <- mean(group\_data)  
 ng <- length(group\_data)  
 total\_variance <- total\_variance + ng\*(group\_mean - grand\_mean)^2  
 }  
 return(total\_variance/(n\_group-1))  
}  
BetweenGroupsVariance(wine$Alcohol,wine$type)

## [1] 35.39742

As the example, the within group variande for alcohol is about 35.39. We can calculate the “separation” achieved by a variable as its between-groups variance devided by its within-groups variance.

35.39742/0.2620525

## [1] 135.0776

The separation indicate that much more variance in the data set can be explianed by the group difference than the within group uncertainty. A high value of separation means the variboe is useful in classfication since the variance difference across group is significant. Similarly, we can calculate all variables in the data set.

for (i in 2:14){  
 between <- BetweenGroupsVariance(wine[,i],wine$type)  
 within <- WithinGroupsVariance(wine[,i],wine$type)  
 variablename <- colnames(wine)[i]  
 sep <- between / within  
 print(paste("variable",variablename,",Wv=",within,",Bv=",between,",Sep=",sep))  
}

## [1] "variable Alcohol ,Wv= 0.262052469153907 ,Bv= 35.3974249602692 ,Sep= 135.0776242428"  
## [1] "variable Malic acid ,Wv= 0.88754679674658 ,Bv= 32.7890184869213 ,Sep= 36.9434249631837"  
## [1] "variable Ash ,Wv= 0.0660721013425184 ,Bv= 0.879611357248741 ,Sep= 13.312901199991"  
## [1] "variable Alcalinity ,Wv= 8.00681118121155 ,Bv= 286.41674636309 ,Sep= 35.7716374073093"  
## [1] "variable Magnesium ,Wv= 180.65777316441 ,Bv= 2245.50102788939 ,Sep= 12.4295843381499"  
## [1] "variable Phenols ,Wv= 0.191270475224227 ,Bv= 17.9283572942847 ,Sep= 93.7330096203672"  
## [1] "variable Flavanoids ,Wv= 0.274707514337437 ,Bv= 64.2611950235641 ,Sep= 233.925872681549"  
## [1] "variable Nonflavanoid phenols ,Wv= 0.0119117022132797 ,Bv= 0.328470157461624 ,Sep= 27.5754171469659"  
## [1] "variable Proanthocyanins ,Wv= 0.246172943795542 ,Bv= 7.45199550777775 ,Sep= 30.2713831702276"  
## [1] "variable Color intensity ,Wv= 2.28492308133354 ,Bv= 275.708000822304 ,Sep= 120.664018441003"  
## [1] "variable Hue ,Wv= 0.0244876469432414 ,Bv= 2.48100991493829 ,Sep= 101.3167953903"  
## [1] "variable OD280/OD315 ,Wv= 0.160778729560982 ,Bv= 30.5435083544253 ,Sep= 189.972320578889"  
## [1] "variable Proline ,Wv= 29707.6818705169 ,Bv= 6176832.32228483 ,Sep= 207.920373902178"

As we can see from the analysis, the variable Flavanoids and Proline have the high separation, while Magnesium and Nonflavanoid phenols have the low separation.

## Principal Component Analysis

#### Summary of PCA

Principal component analysis (PCA) simplifies the complexity in high-dimensional data while retaining trends and patterns. In other word, PCA is a technique for feature extraction — so it combines input variables in a specific way, then we can drop the “least important” variables while still retaining the most valuable parts of all of the variable. It reduce the high dimension of data space into a lower dimension principal component space through orthogonal linear transformation. The new coordinate system created by the principaal component has greatest variance by some scalar projection of the data comes to lie on the first coordinate (called the first principal component), the second greatest variance on the second coordinate, and so on.

The biggest assunmption of PCA is the linear assumption. PCA can find the orthogonal projections of the dataset that contains the highest variance if the data set is linear correlated. Otherwise, it cannot help to reduce the dimension. In other word, the underlying structure of the data must be linear, patterns that are highly correlated may be unresolved because all PCs are uncorrelated, and the goal is to maximize variance and not necessarily to find clusters. Second, PCA rely on the orthogonal tranformations. Howeever, it may not be the best data transformation in terms of extracting the feature. Third, PCA is scale variant. This means that the scale of the data set will change the result of the PCA. Finally, PCA assume no missing value in the data set.

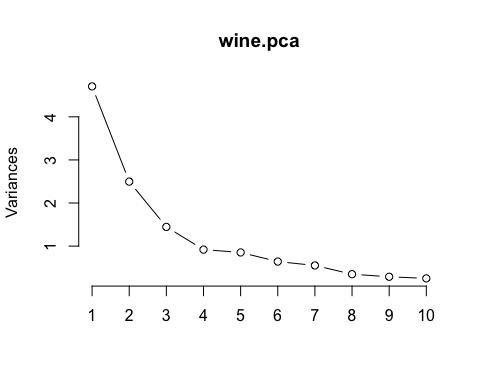
#### PCA Data Analysis

To carry out a principal component analysis (PCA) on a multivariate data set, the first step is often to standardise the variables. This is necessary if the input variables have very different variances, which is true in this case as the concentrations of the 13 chemicals have very different variances (see above).

scale\_imput <-scale(wine[,2:14])  
wine.pca <- prcomp(scale\_imput)   
summary(wine.pca)

## Importance of components:  
## PC1 PC2 PC3 PC4 PC5 PC6 PC7  
## Standard deviation 2.169 1.5802 1.2025 0.95863 0.92370 0.80103 0.74231  
## Proportion of Variance 0.362 0.1921 0.1112 0.07069 0.06563 0.04936 0.04239  
## Cumulative Proportion 0.362 0.5541 0.6653 0.73599 0.80162 0.85098 0.89337  
## PC8 PC9 PC10 PC11 PC12 PC13  
## Standard deviation 0.59034 0.53748 0.5009 0.47517 0.41082 0.32152  
## Proportion of Variance 0.02681 0.02222 0.0193 0.01737 0.01298 0.00795  
## Cumulative Proportion 0.92018 0.94240 0.9617 0.97907 0.99205 1.00000

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



The most obvious change in slope in the scree plot occurs at component four, which is the “elbow” of the scree plot. Therefore, it cound be argued based on the basis of the scree plot that the first three components should be retained.

cumsum((wine.pca$sdev)^2)/sum((wine.pca$sdev)^2)

## [1] 0.3619885 0.5540634 0.6652997 0.7359900 0.8016229 0.8509812 0.8933680  
## [8] 0.9201754 0.9423970 0.9616972 0.9790655 0.9920479 1.0000000

As we can see the first four principal component keeps about 73.59% variance. There are some other rules to we can pick to determine the number of principal component to keep. For example, if we need to ensure that at least 80% of the total variance should be maintained. Thus, first five principal components should be picked. Another rule is to pick the principal component which has the variance bigger than 1. In this way, we have to keep the first three principal component.

(wine.pca$sdev)^2

## [1] 4.7058503 2.4969737 1.4460720 0.9189739 0.8532282 0.6416570 0.5510283  
## [8] 0.3484974 0.2888799 0.2509025 0.2257886 0.1687702 0.1033779

Take the first principal’s loading is shown as below.

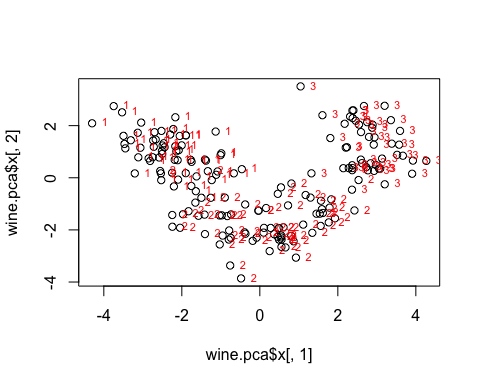
round(wine.pca$rotation[,1],2)

## Alcohol Malic acid Ash   
## -0.14 0.25 0.00   
## Alcalinity Magnesium Phenols   
## 0.24 -0.14 -0.39   
## Flavanoids Nonflavanoid phenols Proanthocyanins   
## -0.42 0.30 -0.31   
## Color intensity Hue OD280/OD315   
## 0.09 -0.30 -0.38   
## Proline   
## -0.29

This indicates that the first principal component rely more on Flavanoids, Phenols, and OD280/OD315. The variable Ash almost contribute zero to the first principal component. The loading of the principal component is helpful in interpretating the possible meaning of the principal component. In this study, I am not familiar with chemical knowledge of these attributes. So, I will not take the risk to summary the meaning of the principal component.

Finllay, scatterplot of the first two principal components, and label the data points with the cultivar that the wine samples come from will be take as an example.

plot(wine.pca$x[,1],wine.pca$x[,2]) # make a scatterplot  
text(wine.pca$x[,1],wine.pca$x[,2], wine$type, cex=0.7, pos=4, col="red")



The scatterplot shows the first principal component on the x-axis, and the second principal component on the y-axis. We can see from the scatterplot that wine samples of cultivar 1 have much lower values of the first principal component than wine samples of cultivar 3. Therefore, the first principal component separates wine samples of cultivars 1 from those of cultivar 3.

We can also see that wine samples of cultivar 2 have much higher values of the second principal component than wine samples of cultivars 1 and 3. Therefore, the second principal component separates samples of cultivar 2 from samples of cultivars 1 and 3.

Therefore, the first two principal components are reasonably useful for distinguishing wine samples of the three different cultivars.

## Linear Discriminate Analysis

#### Summary of LDA

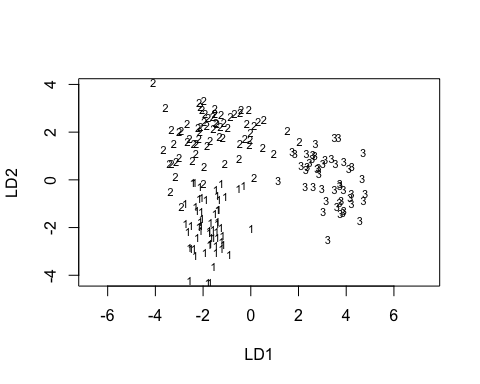
Linear discriminate analysis (LDA) is a method to find a linear combination of features that characterizes or separates two or more classes of objects or events. The resulting combination may be used as a linear classifier, or, more commonly, for dimensionality reduction before later classification.

LDA assume that the approaches the problem by assuming that the conditional probability density functions from both classes are both normally distributed with differnt mean vector and the same variance-covariance matrix (can also be different). If the varianes are assumed to be the same, it is called homoscedasticity assumption. Besides, LDA also take the multicollinearity and indepdence assumption. Predictive power can decrease with an increased correlation between predictor variables. Meanwhile, simples are assumed to be randomly sampled, and a participant’s score on one variable is assumed to be independent of scores on that variable for all other participants. Thus, if the data does not follow the multivariate normal distribution, the variables have high multicollinearlity or the samples are not independent, the result of LDA will be biased. Besides, the LDA also rely on the complete data set and binary class.

#### LDA analysis

If we want to separate the wines by cultivar, the wines come from three different cultivars, so the number of groups is 3, and the number of variables is four principal component I calculated in the last section. The maximum number of useful discriminant functions that can separate the wines by cultivar is the minimum of 3-1 and 4, and so in this case it is the minimum of 2 and 13, which is 2. The variance is assumed to be the same.

library(MASS)  
wine.lda <- lda(x = wine.pca$x[,1:4], grouping = wine$type)  
plot(wine.lda)



As we can see that the two discriminant functions seperate the three wine type well. To see the loading of the discrimiant functions, we can:

wine.lda$scaling

## LD1 LD2  
## PC1 0.9606948 0.4427462  
## PC2 0.7622075 -1.0176477  
## PC3 -0.2018024 0.2830970  
## PC4 -0.1167712 -0.1659482

As we can see, the first discrimiant function rely more on the first principal component, while the second discriminant function rely more on the second principal component.

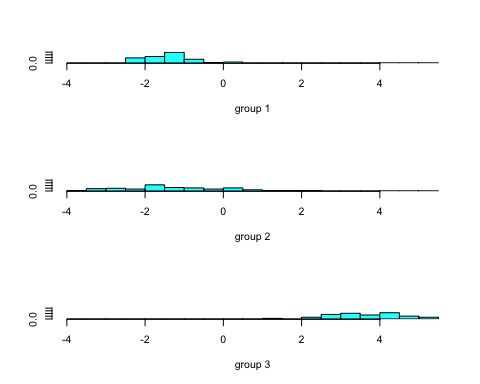
table(Predicted=predict(wine.lda)$class, Actual=wine$type)

## Actual  
## Predicted 1 2 3  
## 1 59 3 0  
## 2 0 66 0  
## 3 0 2 48

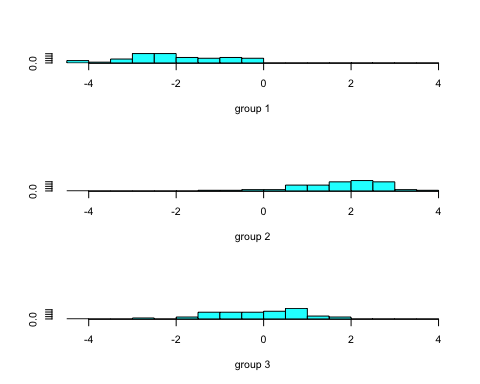
As we can see from the table above, most of the data are predicted correctly. Only 2 points belongs to the range 3 are predicted to be in range 2. The overall aggrement rate is .

To futher analysis the separation achieved by each discriminant function. Stacked histogram can be used.

wine.pred <- predict(wine.lda, wine.pca$x[,1:4])  
ldahist(data = wine.pred$x[,1], g=wine$type)



ldahist(data = wine.pred$x[,2], g=wine$type)



As we can see, the frist discriminant function has the power to distinguish the type 1 and 2 with type 3. However, the difference between 1 and 2 are limiated. The differences between the 3 wine types are more clear seperated in the second discriminant function.

## Reference

Forina, M. et al. 1991. PARVUS - An Extendible Package for Data Exploration, Classification and Correlation. Institute of Pharmaceutical and Food Analysis and Technologies, *Via Brigata Salerno*, 16147 Genoa, Italy.