Winequality final

Sai Supriya

2022-12-09

## Importing

library(readr)

## Warning: package 'readr' was built under R version 4.1.2

wines <- read\_csv("Downloads/Machine Learning2(Udemy)/winequality.csv")

## Rows: 6497 Columns: 13  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (1): type  
## dbl (12): fixed acidity, volatile acidity, citric acid, residual sugar, chlo...  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

str(wines)

## spec\_tbl\_df [6,497 × 13] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ type : chr [1:6497] "red" "red" "red" "red" ...  
## $ fixed acidity : num [1:6497] 7.4 7.8 7.8 11.2 7.4 7.4 7.9 7.3 7.8 7.5 ...  
## $ volatile acidity : num [1:6497] 0.7 0.88 0.76 0.28 0.7 0.66 0.6 0.65 0.58 0.5 ...  
## $ citric acid : num [1:6497] 0 0 0.04 0.56 0 0 0.06 0 0.02 0.36 ...  
## $ residual sugar : num [1:6497] 1.9 2.6 2.3 1.9 1.9 1.8 1.6 1.2 2 6.1 ...  
## $ chlorides : num [1:6497] 0.076 0.098 0.092 0.075 0.076 0.075 0.069 0.065 0.073 0.071 ...  
## $ free sulfur dioxide : num [1:6497] 11 25 15 17 11 13 15 15 9 17 ...  
## $ total sulfur dioxide: num [1:6497] 34 67 54 60 34 40 59 21 18 102 ...  
## $ density : num [1:6497] 0.998 0.997 0.997 0.998 0.998 ...  
## $ pH : num [1:6497] 3.51 3.2 3.26 3.16 3.51 3.51 3.3 3.39 3.36 3.35 ...  
## $ sulphates : num [1:6497] 0.56 0.68 0.65 0.58 0.56 0.56 0.46 0.47 0.57 0.8 ...  
## $ alcohol : num [1:6497] 9.4 9.8 9.8 9.8 9.4 9.4 9.4 10 9.5 10.5 ...  
## $ quality : num [1:6497] 5 5 5 6 5 5 5 7 7 5 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. type = col\_character(),  
## .. `fixed acidity` = col\_double(),  
## .. `volatile acidity` = col\_double(),  
## .. `citric acid` = col\_double(),  
## .. `residual sugar` = col\_double(),  
## .. chlorides = col\_double(),  
## .. `free sulfur dioxide` = col\_double(),  
## .. `total sulfur dioxide` = col\_double(),  
## .. density = col\_double(),  
## .. pH = col\_double(),  
## .. sulphates = col\_double(),  
## .. alcohol = col\_double(),  
## .. quality = col\_double()  
## .. )  
## - attr(\*, "problems")=<externalptr>

summary(wines)

## type fixed acidity volatile acidity citric acid   
## Length:6497 Min. : 3.800 Min. :0.0800 Min. :0.0000   
## Class :character 1st Qu.: 6.400 1st Qu.:0.2300 1st Qu.:0.2500   
## Mode :character Median : 7.000 Median :0.2900 Median :0.3100   
## Mean : 7.215 Mean :0.3397 Mean :0.3186   
## 3rd Qu.: 7.700 3rd Qu.:0.4000 3rd Qu.:0.3900   
## Max. :15.900 Max. :1.5800 Max. :1.6600   
## residual sugar chlorides free sulfur dioxide total sulfur dioxide  
## Min. : 0.600 Min. :0.00900 Min. : 1.00 Min. : 6.0   
## 1st Qu.: 1.800 1st Qu.:0.03800 1st Qu.: 17.00 1st Qu.: 77.0   
## Median : 3.000 Median :0.04700 Median : 29.00 Median :118.0   
## Mean : 5.443 Mean :0.05603 Mean : 30.53 Mean :115.7   
## 3rd Qu.: 8.100 3rd Qu.:0.06500 3rd Qu.: 41.00 3rd Qu.:156.0   
## Max. :65.800 Max. :0.61100 Max. :289.00 Max. :440.0   
## density pH sulphates alcohol   
## Min. :0.9871 Min. :2.720 Min. :0.2200 Min. : 8.00   
## 1st Qu.:0.9923 1st Qu.:3.110 1st Qu.:0.4300 1st Qu.: 9.50   
## Median :0.9949 Median :3.210 Median :0.5100 Median :10.30   
## Mean :0.9947 Mean :3.219 Mean :0.5313 Mean :10.49   
## 3rd Qu.:0.9970 3rd Qu.:3.320 3rd Qu.:0.6000 3rd Qu.:11.30   
## Max. :1.0390 Max. :4.010 Max. :2.0000 Max. :14.90   
## quality   
## Min. :3.000   
## 1st Qu.:5.000   
## Median :6.000   
## Mean :5.818   
## 3rd Qu.:6.000   
## Max. :9.000

###We will cluster the wines by characteristics

##Data preprocessing

# Removing Unwanted columns like Characters and numbers from the given dataset.

#From the summary statistics it is observed that the total sulphur dioxide the maximum and minimum contents are to be citric acid and volatile acidity.And the other variable factor here is residual sugar where max and min values are 65 and 0 from the given data. ## Removing missing NA

wines1<- na.omit(wines)

## Creating matric

wines2<- cbind(wines1$`fixed acidity`, wines1$`volatile acidity`, wines1$`citric acid`,wines1$`residual sugar`, wines1$chlorides, wines1$`free sulfur dioxide`, wines1$`total sulfur dioxide`, wines1$density, wines1$pH, wines1$sulphates, wines1$alcohol,wines1$quality)

## Standardising the cluster variables

wines3<- scale(wines2)

##Name the columns

colnames(wines3)<- c("fixed.acidity","volatile.acidity", "citric.acid","residual.sugar","chlorides","free.sulfur.dioxide","total.sulfur.dioxide","density","pH","sulphates", "alcohol","quality")  
  
str(wines3)

## num [1:6497, 1:12] 0.142 0.451 0.451 3.074 0.142 ...  
## - attr(\*, "scaled:center")= num [1:12] 7.215 0.34 0.319 5.443 0.056 ...  
## - attr(\*, "scaled:scale")= num [1:12] 1.296 0.165 0.145 4.758 0.035 ...  
## - attr(\*, "dimnames")=List of 2  
## ..$ : NULL  
## ..$ : chr [1:12] "fixed.acidity" "volatile.acidity" "citric.acid" "residual.sugar" ...

wines3<- as.data.frame(wines3)

print("---Data preview---")

## [1] "---Data preview---"

head(wines3)

## fixed.acidity volatile.acidity citric.acid residual.sugar chlorides  
## 1 0.1424623 2.1886645 -2.192664 -0.7447208 0.5699140  
## 2 0.4510010 3.2819823 -2.192664 -0.5975941 1.1978825  
## 3 0.4510010 2.5531038 -1.917405 -0.6606484 1.0266184  
## 4 3.0735801 -0.3624106 1.660957 -0.7447208 0.5413699  
## 5 0.1424623 2.1886645 -2.192664 -0.7447208 0.5699140  
## 6 0.1424623 1.9457049 -2.192664 -0.7657389 0.5413699  
## free.sulfur.dioxide total.sulfur.dioxide density pH sulphates  
## 1 -1.1000552 -1.4462472 1.0349132 1.8129500 0.1930819  
## 2 -0.3112961 -0.8624022 0.7014323 -0.1150642 0.9995017  
## 3 -0.8746955 -1.0924018 0.7681285 0.2580999 0.7978967  
## 4 -0.7620156 -0.9862481 1.1016093 -0.3638402 0.3274852  
## 5 -1.1000552 -1.4462472 1.0349132 1.8129500 0.1930819  
## 6 -0.9873753 -1.3400936 1.0349132 1.8129500 0.1930819  
## alcohol quality  
## 1 -0.9153937 -0.9371575  
## 2 -0.5800235 -0.9371575  
## 3 -0.5800235 -0.9371575  
## 4 -0.5800235 0.2079830  
## 5 -0.9153937 -0.9371575  
## 6 -0.9153937 -0.9371575

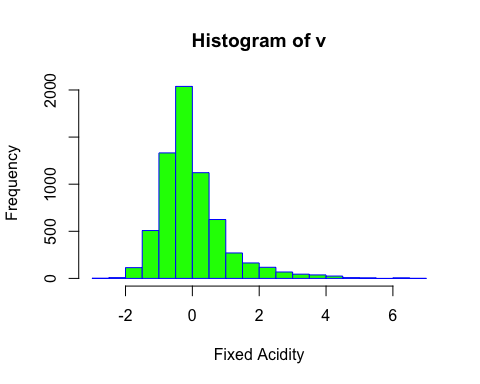
#Number of rows to identify how big is the data we are dealing with  
print("---Number of rows---")

## [1] "---Number of rows---"

nrow(wines3)

## [1] 6497

v = wines3$fixed.acidity  
hist(v,xlab = "Fixed Acidity",col = "Green",border = "blue")



library("ggplot2")

## Warning: package 'ggplot2' was built under R version 4.1.2

gg=ggplot(wines3,aes(x=quality,y=alcohol))+geom\_point(aes(col= density))+geom\_smooth(method="loess",se=F)+labs(subtitle = "Quality Vs Alcohol",y="alcohol",x="quality",title="Scatterplot",caption = "Source: Wine\_prediction")  
plot(gg)

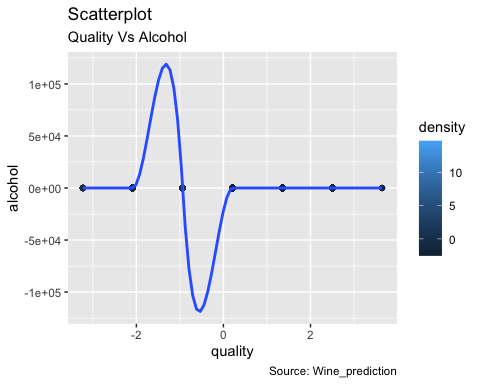
## `geom\_smooth()` using formula 'y ~ x'

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =  
## parametric, : pseudoinverse used at -0.93716

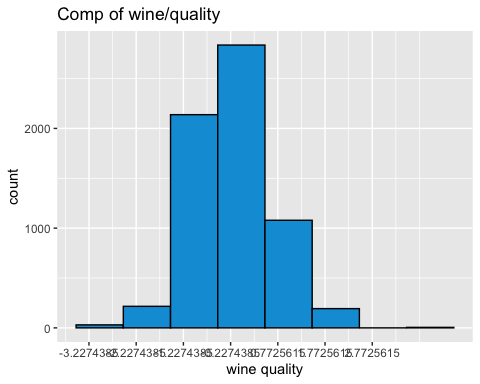
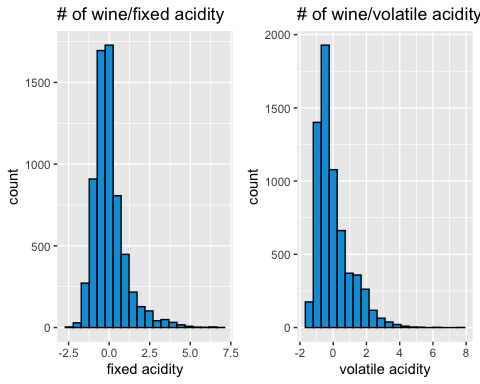
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =  
## parametric, : neighborhood radius 1.1451

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =  
## parametric, : reciprocal condition number 1.4277e-15

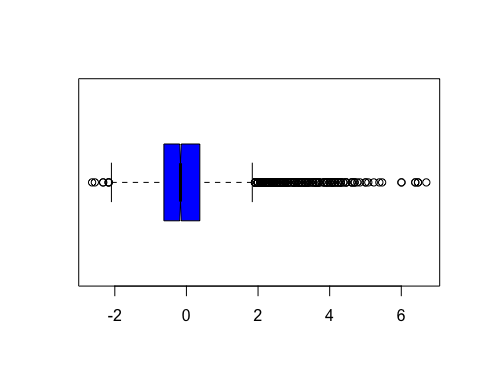
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric =  
## parametric, : There are other near singularities as well. 1.3113



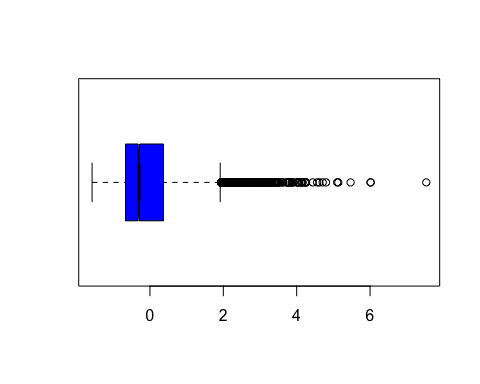
ggplot(data=wines3, aes(x=quality)) +   
 geom\_histogram(binwidth = 1, color = 'black', fill = '#099DD9') +  
 scale\_x\_continuous(breaks = seq(min(wines3$quality), max(wines3$quality), 1)) +   
 ggtitle("Comp of wine/quality") +   
 xlab('wine quality')

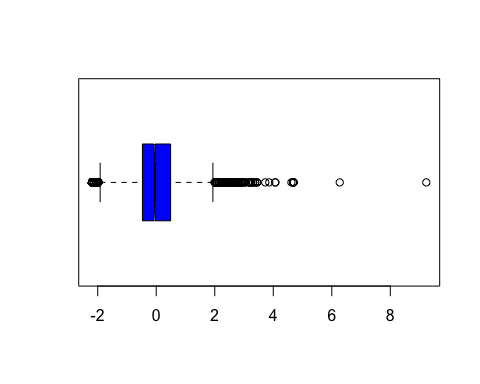
p11<- boxplot(wines3$fixed.acidity,horizontal = T,col = "blue", notch = T)



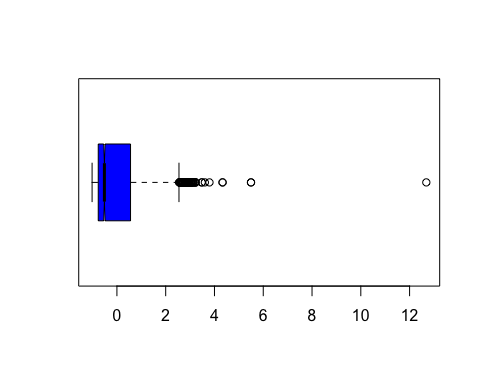
p21<- boxplot(wines3$volatile.acidity,horizontal = T,col = "blue", notch = T)



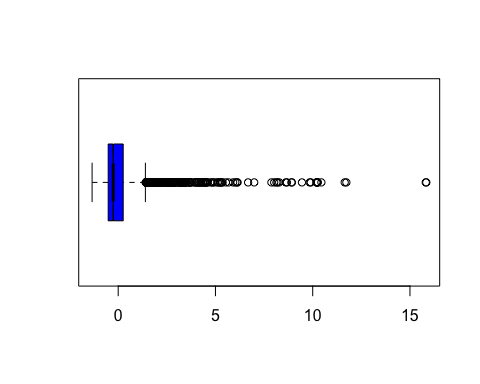
p31 <- boxplot(wines3$citric.acid,horizontal = T,col = "blue", notch = T)



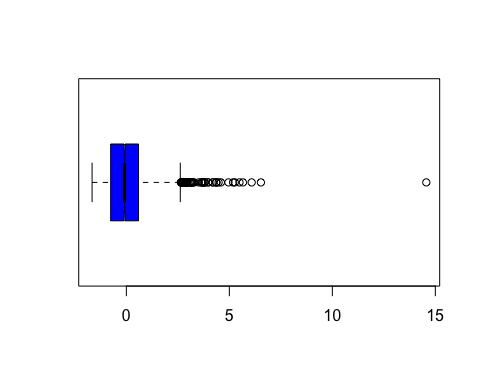
p41 <- boxplot(wines3$residual.sugar,horizontal = T,col = "blue", notch = T)



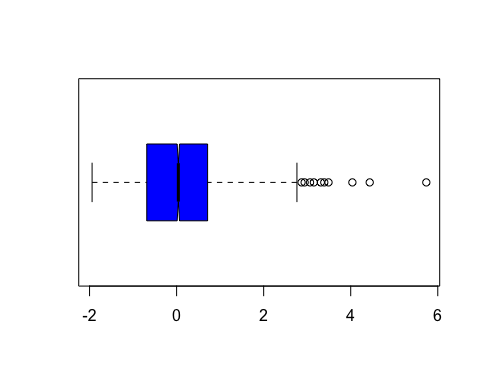
p51 <- boxplot(wines3$chlorides,horizontal = T,col = "blue", notch = T)



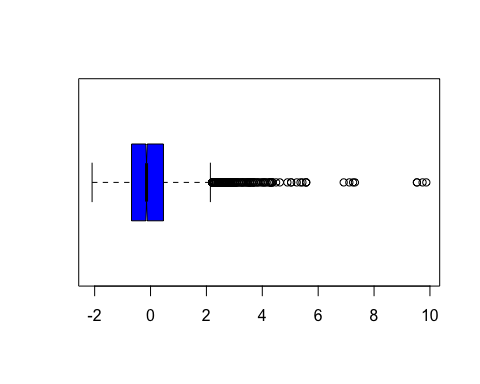
p61 <- boxplot(wines3$free.sulfur.dioxide,horizontal = T,col = "blue", notch = T)



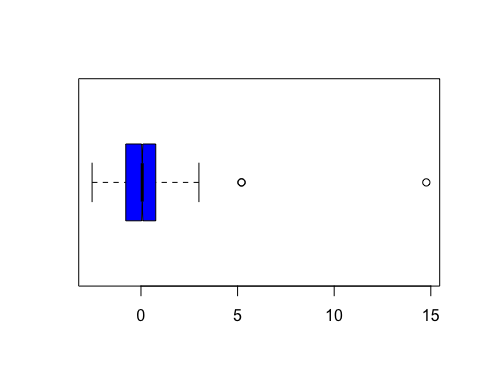
p71 <- boxplot(wines3$total.sulfur.dioxide,horizontal = T,col = "blue", notch = T)



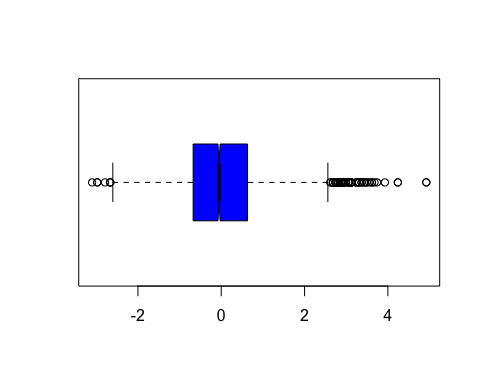
p81 <-boxplot(wines3$sulphates,horizontal = T,col = "blue", notch = T)



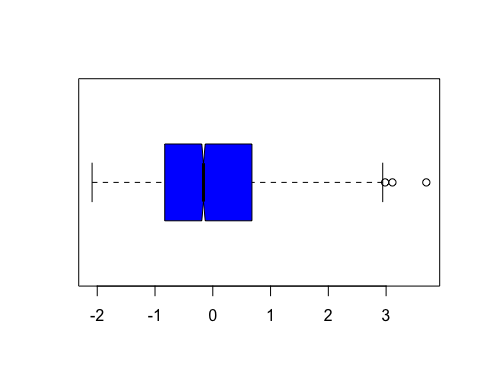
p91 <-boxplot(wines3$density,horizontal = T,col = "blue", notch = T)



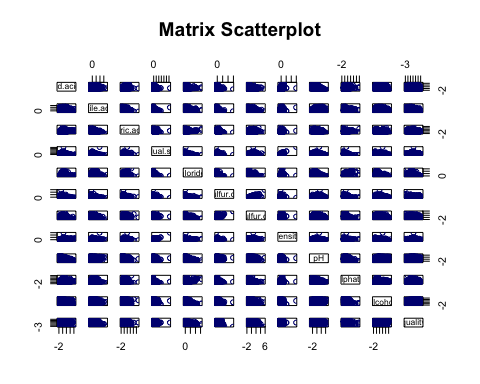
p101 <-boxplot(wines3$pH,horizontal = T,col = "blue", notch = T)



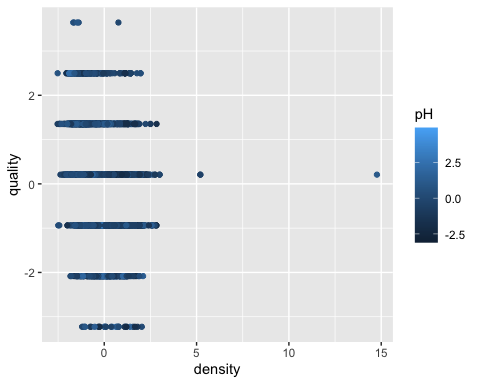
p111 <-boxplot(wines3$alcohol,horizontal = T,col = "blue", notch = T)



plot(wines3, col="navy", main="Matrix Scatterplot")



ggplot(wines3, aes(x = density, y = quality)) + geom\_point(aes(color = pH))



library("psych")

## Warning: package 'psych' was built under R version 4.1.2

##   
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':  
##   
## %+%, alpha

describe(wines3)

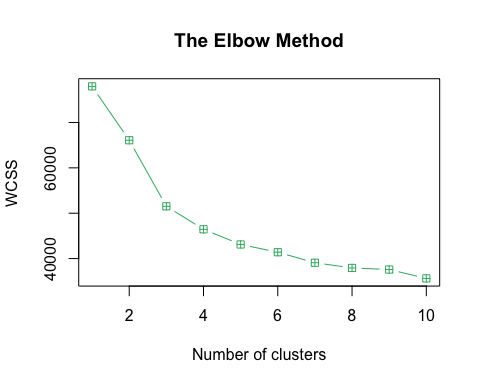
## vars n mean sd median trimmed mad min max range  
## fixed.acidity 1 6497 0 1 -0.17 -0.12 0.69 -2.63 6.70 9.33  
## volatile.acidity 2 6497 0 1 -0.30 -0.14 0.72 -1.58 7.53 9.11  
## citric.acid 3 6497 0 1 -0.06 -0.01 0.71 -2.19 9.23 11.42  
## residual.sugar 4 6497 0 1 -0.51 -0.16 0.53 -1.02 12.69 13.70  
## chlorides 5 6497 0 1 -0.26 -0.15 0.47 -1.34 15.84 17.18  
## free.sulfur.dioxide 6 6497 0 1 -0.09 -0.07 1.00 -1.66 14.56 16.23  
## total.sulfur.dioxide 7 6497 0 1 0.04 0.00 1.02 -1.94 5.74 7.68  
## density 8 6497 0 1 0.06 0.00 1.14 -2.53 14.77 17.30  
## pH 9 6497 0 1 -0.05 -0.03 1.01 -3.10 4.92 8.02  
## sulphates 10 6497 0 1 -0.14 -0.10 0.80 -2.09 9.87 11.96  
## alcohol 11 6497 0 1 -0.16 -0.07 1.12 -2.09 3.70 5.79  
## quality 12 6497 0 1 0.21 -0.04 1.70 -3.23 3.64 6.87  
## skew kurtosis se  
## fixed.acidity 1.72 5.05 0.01  
## volatile.acidity 1.49 2.82 0.01  
## citric.acid 0.47 2.39 0.01  
## residual.sugar 1.43 4.35 0.01  
## chlorides 5.40 50.84 0.01  
## free.sulfur.dioxide 1.22 7.90 0.01  
## total.sulfur.dioxide 0.00 -0.37 0.01  
## density 0.50 6.60 0.01  
## pH 0.39 0.37 0.01  
## sulphates 1.80 8.64 0.01  
## alcohol 0.57 -0.53 0.01  
## quality 0.19 0.23 0.01

## K means with 3 clusters

# We selected 3clusters of the data as a sample, storing 6497 observations in the sample data, using the seed 6, a random 1-digit number,where doing the sampling with a precise and chosen data gives an accurate results and provides the correct set of findings in determining the clusters.We also want to set the seed so that we ensure reproducibility with this code:

# Using the elbow method to find the optimal number of clusters

set.seed(6)  
wcss = vector()  
for (i in 1:10) wcss[i] = sum(kmeans(wines3, i)$withinss)  
plot(1:10,  
 wcss,  
 type = 'b',  
 main = paste('The Elbow Method'),  
 xlab = 'Number of clusters',  
 ylab = 'WCSS',col="mediumseagreen", pch=12)

 #Thus, elbow method can be used to assess individual observations, or the average elbow method can be used to assess the choice of k. which gives k = 5 the optimal number of cluster that can be formed is to be 5 clusters.

The distance metric is one of the commonly used metrics to compare results across different K values. When the number of clusters, K is increased, the distance from centroid to data points will be decreased and will reach a point where K is the same as the number of data points.

In this the elbow method analysis is used to choose an optimal value for n\_clusters. Where we have found the optimal number of clusters formed are 5.

# Fitting K-Means to the dataset

set.seed(29)  
kmeans = kmeans(x = wines3, centers = 5)  
y\_kmeans = kmeans$cluster

#K Mean Clustering- i used k mean clustering to generate groups with similar characteristics and used large data scale the number of groups is represented by k,and i used Silhouette method to get optimal numbers OF clusters ‘K’ The optimal number of clusters K=5.

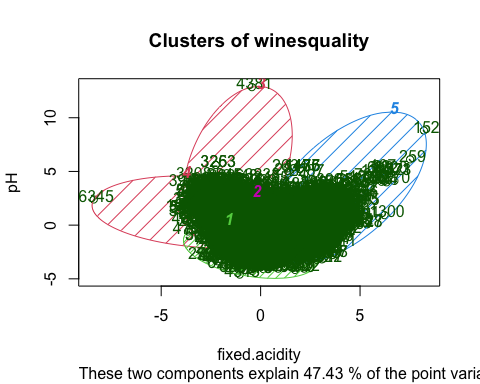
fit <- kmeans(wines3,5 )

#running kmeans for mydata dataset with k=5 and storing the result in fit.

library(cluster)

## Warning: package 'cluster' was built under R version 4.1.2

clusplot(wines3,  
 y\_kmeans,  
 lines = 0,  
 shade = TRUE,  
 color = TRUE,  
 labels = 2,  
 plotchar = FALSE,  
 span = TRUE,  
 main = paste('Clusters of winesquality'),  
 xlab = 'fixed.acidity',  
 ylab = 'pH')



summary(wines3)

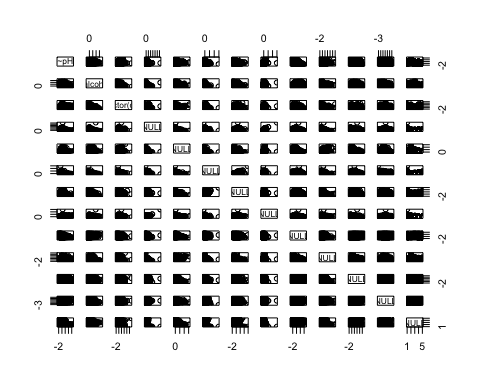
## fixed.acidity volatile.acidity citric.acid residual.sugar   
## Min. :-2.6344 Min. :-1.5772 Min. :-2.19266 Min. :-1.0180   
## 1st Qu.:-0.6289 1st Qu.:-0.6661 1st Qu.:-0.47230 1st Qu.:-0.7657   
## Median :-0.1661 Median :-0.3017 Median :-0.05941 Median :-0.5135   
## Mean : 0.0000 Mean : 0.0000 Mean : 0.00000 Mean : 0.0000   
## 3rd Qu.: 0.3739 3rd Qu.: 0.3665 3rd Qu.: 0.49111 3rd Qu.: 0.5584   
## Max. : 6.6989 Max. : 7.5338 Max. : 9.23057 Max. :12.6858   
## chlorides free.sulfur.dioxide total.sulfur.dioxide density   
## Min. :-1.3425 Min. :-1.66345 Min. :-1.9416 Min. :-2.53000   
## 1st Qu.:-0.5148 1st Qu.:-0.76202 1st Qu.:-0.6855 1st Qu.:-0.78589   
## Median :-0.2579 Median :-0.08594 Median : 0.0399 Median : 0.06448   
## Mean : 0.0000 Mean : 0.00000 Mean : 0.0000 Mean : 0.00000   
## 3rd Qu.: 0.2559 3rd Qu.: 0.59014 3rd Qu.: 0.7122 3rd Qu.: 0.76479   
## Max. :15.8410 Max. :14.56245 Max. : 5.7368 Max. :14.76765   
## pH sulphates alcohol quality   
## Min. :-3.10038 Min. :-2.0918 Min. :-2.0892 Min. :-3.2274   
## 1st Qu.:-0.67481 1st Qu.:-0.6805 1st Qu.:-0.8316 1st Qu.:-0.9372   
## Median :-0.05287 Median :-0.1429 Median :-0.1608 Median : 0.2080   
## Mean : 0.00000 Mean : 0.0000 Mean : 0.0000 Mean : 0.0000   
## 3rd Qu.: 0.63126 3rd Qu.: 0.4619 3rd Qu.: 0.6776 3rd Qu.: 0.2080   
## Max. : 4.92265 Max. : 9.8701 Max. : 3.6959 Max. : 3.6434

##fit$cluster will give the cluster in which the obs go to.Lets Store it in mydata dataset with header cluster.

wines3$cluster=fit$cluster  
#or same as:  
wines3$cluster=fit$cluster

##Making Pair wise profiling plots and labelling wines with respect to its ingredients:

library(ggplot2)  
plot(wines3,aes(pH,alcohol,color=as.factor(cluster)))+geom\_point()



## NULL

library(cluster)  
  
  
wine\_df2 <- wines3[, -5]  
set.seed(240) # Setting seed  
kmeans.re <- kmeans(wine\_df2, centers = 5, nstart = 10)  
kmeans.re

## K-means clustering with 5 clusters of sizes 1589, 1649, 603, 985, 1671  
##   
## Cluster means:  
## fixed.acidity volatile.acidity citric.acid residual.sugar free.sulfur.dioxide  
## 1 -0.1846683 -0.3564474 0.29319560 1.3843044 0.94550277  
## 2 -0.4561617 -0.3797596 0.03181511 -0.3856796 0.04339946  
## 3 2.0838769 0.4322947 1.02553445 -0.5709293 -0.91739421  
## 4 0.1058135 1.6872984 -1.24413920 -0.6192310 -0.78606731  
## 5 -0.1886029 -0.4368905 0.05309929 -0.3647284 -0.14751921  
## total.sulfur.dioxide density pH sulphates alcohol  
## 1 1.02642311 0.8794364 -0.42078939 -0.2449341 -0.84490933  
## 2 -0.04019979 -1.1399987 0.01731967 -0.2407335 1.17049279  
## 3 -1.31976067 0.9752074 -0.07050727 1.4407145 0.09609325  
## 4 -1.12582850 0.5132103 0.94103422 0.4272302 -0.25927961  
## 5 0.20350682 -0.3657263 -0.14621701 -0.3012584 -0.23347426  
## quality cluster  
## 1 -0.24099276 1.004405  
## 2 0.94548409 4.998787  
## 3 0.09783685 2.000000  
## 4 -0.52792960 2.995939  
## 5 -0.42797771 3.994614  
##   
## Clustering vector:  
## [1] 4 4 4 3 4 4 4 4 4 4 4 4 4 3 4 4 3 3 4 3 3 4 3 4 4 4 4 3 4 4 4 4 4 4 4 4 4  
## [38] 3 4 4 4 4 4 4 4 4 4 3 4 5 4 4 4 5 4 4 3 4 4 4 3 3 4 4 4 4 4 4 3 4 4 4 4 4  
## [75] 3 3 3 4 4 4 4 3 4 4 5 4 3 4 3 4 5 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 3 4 3 3 4  
## [112] 4 4 3 4 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 4 4 4 4 4 4 4 4 4 4 4 2 4 2 3 4 4  
## [149] 4 3 3 3 4 4 5 5 5 5 4 4 4 4 4 4 4 3 4 4 4 3 4 4 4 4 4 4 4 4 4 4 4 3 4 4 4  
## [186] 3 5 4 5 5 4 4 4 4 4 4 4 3 4 4 3 3 4 4 4 3 3 4 4 3 3 4 3 4 4 5 4 4 4 4 5 4  
## [223] 4 4 4 4 3 4 4 4 2 4 4 4 4 4 4 4 4 4 3 3 4 3 3 4 4 4 4 4 3 4 3 4 4 4 3 4 3  
## [260] 3 4 4 4 4 3 3 4 3 4 3 4 3 3 4 4 4 4 3 3 3 3 3 4 3 4 4 3 4 4 3 4 3 3 4 3 3  
## [297] 3 4 4 4 4 3 4 4 3 3 4 3 3 5 3 4 3 5 4 4 4 4 3 4 3 4 4 3 3 3 3 3 3 3 3 3 4  
## [334] 4 4 3 3 4 3 3 3 3 3 3 3 4 4 3 3 4 3 4 4 3 2 4 3 3 3 3 4 3 3 3 3 3 3 3 3 3  
## [371] 4 3 3 4 3 3 3 3 3 3 3 3 3 3 4 4 4 4 4 3 4 3 3 3 3 3 4 3 3 4 4 5 3 3 4 3 3  
## [408] 3 3 3 3 3 4 3 4 4 3 4 3 4 3 4 4 3 4 4 4 4 3 3 3 4 3 3 3 3 4 3 3 4 3 3 3 3  
## [445] 4 4 3 3 4 3 3 3 4 3 5 3 3 4 3 3 3 4 3 3 3 3 3 3 3 4 3 3 3 3 3 4 3 3 4 4 3  
## [482] 3 3 3 3 3 3 3 3 3 4 3 3 4 2 3 4 5 3 4 4 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3  
## [519] 3 3 3 3 3 3 3 3 3 3 3 4 3 3 3 3 3 3 4 4 3 3 4 3 4 3 3 3 4 3 3 3 4 3 3 4 3  
## [556] 3 3 3 3 3 3 3 3 3 3 3 4 4 3 4 3 4 3 3 3 3 4 3 3 3 3 3 3 3 3 4 3 4 2 3 3 2  
## [593] 3 3 4 3 3 3 4 3 4 3 4 3 4 4 3 3 3 4 3 3 4 3 3 3 3 3 3 3 4 4 4 4 4 4 4 4 4  
## [630] 4 4 3 4 4 5 4 4 4 4 3 3 3 3 3 3 4 4 4 3 5 3 4 3 3 3 4 3 3 4 4 4 4 4 3 3 4  
## [667] 3 3 3 3 4 4 4 4 3 3 3 4 4 3 3 4 4 4 4 4 4 4 4 3 4 4 3 4 4 4 4 4 4 3 3 4 4  
## [704] 5 4 4 4 4 4 3 3 4 4 4 4 4 4 4 4 4 4 4 4 3 4 4 4 4 4 4 3 4 4 4 4 4 4 4 4 4  
## [741] 4 4 4 3 3 4 3 3 4 4 4 4 4 4 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 4 4  
## [778] 4 4 4 4 4 4 4 4 3 3 3 3 4 3 4 4 4 3 3 3 3 3 3 4 4 4 4 4 2 2 2 4 4 4 3 3 4  
## [815] 3 3 3 3 4 4 4 2 4 4 4 4 2 4 4 4 4 4 3 3 4 4 2 2 3 4 3 4 3 4 3 4 4 4 4 4 3  
## [852] 3 3 3 3 4 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 3 4 4 4 4 4 4 3 4 4 4 4 3  
## [889] 4 3 4 4 3 4 4 4 3 4 3 4 3 4 4 4 4 4 4 4 4 2 3 3 3 3 2 3 4 4 3 4 3 3 4 4 3  
## [926] 3 3 4 3 3 4 4 4 4 4 3 3 3 2 4 3 3 3 3 3 3 3 3 3 3 3 3 3 3 4 3 3 3 4 4 3 4  
## [963] 4 3 3 3 3 4 3 4 3 3 3 3 3 4 4 4 2 3 3 4 2 3 3 4 3 4 4 3 4 4 4 4 3 4 4 4 4  
## [1000] 4 4 3 3 2 4 2 3 3 3 3 3 3 4 4 4 3 3 2 2 4 3 3 4 3 4 4 2 4 5 4 4 4 4 4 4 3  
## [1037] 2 4 3 3 4 4 3 3 2 4 4 4 3 3 4 3 4 3 4 4 3 4 3 3 3 3 3 3 4 4 4 3 3 4 3 4 4  
## [1074] 4 4 3 3 3 3 2 3 2 4 3 4 4 3 5 3 3 3 3 4 3 4 3 4 3 3 3 3 4 4 4 3 4 2 3 4 3  
## [1111] 4 4 3 3 2 4 4 4 2 4 2 4 4 3 4 2 2 4 3 3 4 5 2 4 3 2 3 3 4 4 4 3 4 4 4 3 4  
## [1148] 3 3 3 3 4 4 3 4 4 3 2 3 3 3 3 3 4 4 3 3 3 2 4 3 4 3 4 4 4 4 4 4 3 3 3 3 4  
## [1185] 4 4 4 4 4 4 3 4 2 4 4 4 4 4 5 4 4 2 3 4 3 3 3 3 3 3 4 4 4 3 3 3 4 2 3 3 3  
## [1222] 3 4 3 3 4 4 4 2 4 3 4 4 3 4 5 4 4 4 4 4 3 3 4 1 4 4 4 4 4 4 4 4 4 4 4 4 4  
## [1259] 4 4 3 4 3 4 3 4 4 3 4 2 2 4 4 4 4 4 3 4 4 3 4 4 4 4 4 3 2 4 4 4 4 4 4 4 4  
## [1296] 4 4 4 4 4 4 4 3 5 4 4 4 4 4 4 4 4 4 4 4 4 4 3 4 3 3 4 3 3 4 4 4 4 4 4 4 4  
## [1333] 4 4 4 4 4 4 4 4 4 4 4 4 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 4 3 4 4 4 4 3 4  
## [1370] 4 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 4 3  
## [1407] 3 4 3 4 4 4 3 3 3 4 3 2 4 4 4 4 4 4 3 3 2 4 4 3 4 4 4 4 3 3 3 4 4 4 2 4 4  
## [1444] 4 4 4 4 4 4 2 2 3 4 4 3 4 4 4 3 3 4 4 4 4 4 4 4 4 4 4 4 4 3 4 3 2 3 2 4 3  
## [1481] 4 3 4 3 4 4 4 4 4 4 2 4 4 4 4 4 4 4 4 4 4 4 4 4 3 4 4 3 3 3 4 4 4 4 4 4 4  
## [1518] 4 3 4 4 4 4 4 4 4 4 4 3 4 4 4 4 4 4 4 4 4 4 4 4 2 4 3 3 4 4 4 3 2 4 4 4 4  
## [1555] 4 4 4 4 1 4 4 4 4 4 4 4 5 4 4 4 3 4 4 4 1 4 3 4 4 4 4 4 4 4 2 3 3 4 4 4 4  
## [1592] 4 4 4 4 4 4 4 5 1 5 5 1 1 5 5 1 5 5 5 5 5 2 1 2 5 2 5 5 2 2 2 4 5 1 5 5 2  
## [1629] 2 1 5 5 5 1 2 5 2 1 1 5 5 5 5 5 5 5 1 5 5 2 2 2 2 5 5 1 1 5 5 5 1 5 5 5 5  
## [1666] 2 5 2 5 1 1 5 5 2 5 2 2 5 5 5 1 5 1 1 1 1 1 1 1 1 1 2 2 2 1 1 2 5 1 1 1 1  
## [1703] 1 1 1 1 1 1 5 1 1 1 1 1 4 2 2 1 1 5 1 5 5 5 2 5 1 2 5 5 1 1 1 1 1 5 1 2 2  
## [1740] 5 1 5 5 5 2 5 4 2 5 2 1 5 5 3 1 1 2 2 2 2 1 5 1 1 1 1 2 1 5 2 5 5 2 1 2 5  
## [1777] 1 4 5 5 5 1 1 1 1 1 5 2 1 1 1 5 5 1 1 1 1 1 1 1 1 1 2 5 1 5 3 4 5 5 2 5 5  
## [1814] 2 1 5 1 1 5 5 1 2 5 5 1 5 1 5 1 4 1 1 1 1 1 1 1 2 1 1 5 2 5 1 2 2 2 2 5 5  
## [1851] 1 1 5 5 2 2 2 2 5 5 1 5 1 5 1 1 4 4 4 1 4 1 5 1 1 1 5 5 2 2 2 5 1 1 1 1 1  
## [1888] 1 1 1 5 1 5 5 5 1 2 1 5 4 5 4 2 5 5 5 5 2 5 2 2 5 5 5 5 5 5 2 5 2 5 5 5 5  
## [1925] 1 1 5 1 5 2 5 2 5 5 2 2 1 5 2 1 5 1 5 5 5 2 1 1 1 2 2 5 2 5 1 1 5 5 5 1 5  
## [1962] 5 1 2 2 5 5 5 5 5 5 4 5 2 2 2 2 5 2 2 1 1 5 2 2 2 1 5 1 1 5 4 2 1 5 5 2 1  
## [1999] 2 2 1 5 1 2 5 2 5 5 5 1 1 5 5 1 1 5 1 5 5 2 1 1 1 1 1 1 1 5 1 1 5 1 4 2 2  
## [2036] 5 2 1 5 2 5 2 1 1 5 2 1 1 2 1 5 2 2 2 1 2 5 5 1 5 1 1 5 1 1 1 5 1 1 1 5 2  
## [2073] 5 5 1 2 5 5 5 1 2 1 1 1 5 5 5 5 1 2 2 1 2 5 2 1 2 5 1 1 1 5 5 1 1 5 2 2 2  
## [2110] 5 2 5 2 5 5 5 5 2 5 5 5 2 2 5 5 5 2 2 2 5 1 1 1 1 1 1 1 5 5 5 1 5 5 1 5 5  
## [2147] 2 2 1 1 2 2 1 2 2 1 5 1 2 1 5 1 2 5 1 5 5 5 5 1 2 1 2 2 5 5 2 2 2 1 5 5 5  
## [2184] 2 5 1 2 2 5 1 2 1 5 5 2 1 5 2 2 5 5 5 5 5 5 5 5 1 4 2 5 1 1 1 1 2 5 5 5 1  
## [2221] 1 5 5 5 2 4 1 1 5 1 1 5 1 2 5 2 1 5 5 2 1 1 1 1 1 5 1 5 1 1 1 1 1 2 2 5 5  
## [2258] 5 1 5 2 4 5 1 5 2 1 5 1 1 2 2 5 1 1 1 2 2 2 5 5 1 1 5 1 2 1 1 2 5 5 5 5 1  
## [2295] 1 2 1 1 1 1 2 5 5 2 5 1 5 1 5 1 2 2 1 5 2 1 1 2 2 1 5 2 2 2 5 2 5 5 5 1 1  
## [2332] 2 1 1 5 1 1 2 5 5 2 5 1 5 2 1 2 1 2 2 1 1 1 5 2 1 1 2 2 1 1 1 1 5 5 1 2 1  
## [2369] 1 5 5 5 1 2 2 5 1 1 1 4 1 1 1 1 1 1 1 2 1 1 2 5 2 5 1 1 1 5 2 1 1 1 1 5 1  
## [2406] 1 1 1 1 1 5 5 1 1 5 5 1 5 1 2 1 5 5 5 1 5 2 2 2 5 5 2 2 5 2 2 2 2 2 2 1 1  
## [2443] 2 2 2 1 5 1 2 2 1 5 1 5 2 1 1 1 1 2 1 5 2 5 2 5 1 1 5 1 1 5 5 5 2 2 5 5 2  
## [2480] 2 5 2 2 1 2 2 2 2 2 5 2 5 2 2 5 1 5 5 1 2 1 1 1 5 5 5 2 1 1 5 5 5 5 5 5 5  
## [2517] 5 2 5 5 1 1 1 2 5 4 5 1 1 2 1 1 1 1 1 5 1 1 1 1 1 2 2 5 2 1 5 4 5 5 1 5 5  
## [2554] 1 5 5 2 2 1 1 2 1 2 1 5 5 1 2 2 2 2 1 2 2 5 5 1 2 5 5 2 2 2 5 5 5 5 5 2 5  
## [2591] 5 1 5 5 5 5 1 1 5 5 1 1 2 2 5 2 4 2 5 5 2 5 1 1 5 1 5 5 5 2 2 2 1 2 5 5 4  
## [2628] 2 1 5 1 1 1 1 5 5 4 5 5 4 5 5 1 2 5 5 5 5 2 1 5 5 3 5 5 5 1 5 5 2 1 5 5 5  
## [2665] 1 1 2 2 5 2 1 2 5 1 2 1 5 1 5 5 5 5 1 2 1 2 2 1 2 1 1 5 5 1 2 1 2 1 2 5 2  
## [2702] 5 5 1 2 2 2 2 5 2 2 1 5 4 2 4 1 4 1 1 5 2 5 5 1 2 2 5 2 5 5 1 5 5 1 2 2 5  
## [2739] 5 5 1 5 5 5 1 5 5 5 1 1 1 4 2 5 5 1 1 1 1 5 1 5 1 5 5 5 5 5 5 5 5 2 5 1 1  
## [2776] 1 5 5 5 5 5 5 5 2 1 5 2 5 5 2 2 1 1 1 1 2 1 1 2 2 2 1 2 2 5 1 1 2 2 5 5 5  
## [2813] 2 5 5 2 1 2 2 5 2 5 5 2 2 2 5 2 5 1 2 2 5 5 5 2 5 2 5 5 2 2 2 5 1 5 2 2 1  
## [2850] 2 5 5 2 5 5 5 1 1 1 1 5 5 1 5 2 2 1 1 1 1 5 5 5 1 5 1 1 5 5 2 2 1 2 2 2 2  
## [2887] 2 2 2 5 1 5 5 5 1 5 2 2 5 1 5 1 1 1 2 2 5 5 3 1 5 5 5 1 1 5 5 5 1 2 2 5 5  
## [2924] 5 2 1 5 2 5 5 5 1 2 2 5 2 1 1 1 5 5 1 1 2 2 2 2 2 5 2 2 5 1 5 5 1 1 2 2 5  
## [2961] 2 2 5 5 2 2 5 1 1 5 2 1 1 2 2 5 5 2 5 2 2 5 5 5 4 2 2 2 5 5 2 2 5 2 2 2 1  
## [2998] 1 2 1 1 2 2 5 5 2 2 1 5 2 2 2 5 2 4 5 1 5 2 1 2 5 1 5 2 2 5 2 5 5 5 2 2 5  
## [3035] 1 1 2 1 1 1 5 5 2 1 2 2 5 5 2 1 5 1 1 1 1 2 2 2 5 2 5 5 5 2 5 2 2 5 5 1 1  
## [3072] 5 5 5 5 1 5 5 5 5 5 5 5 5 5 5 1 1 1 5 5 1 2 2 2 4 2 2 2 2 5 5 2 2 5 1 1 1  
## [3109] 5 5 5 2 1 2 5 1 1 2 5 5 5 5 1 1 1 3 1 5 1 1 2 1 5 1 5 5 5 5 5 1 5 5 5 5 5  
## [3146] 5 5 2 5 5 5 2 2 2 2 1 5 5 5 2 5 2 2 5 5 1 5 1 5 5 1 1 5 5 5 5 5 1 1 1 5 2  
## [3183] 1 1 1 5 1 1 2 2 5 5 5 5 1 2 1 5 2 5 1 2 2 5 2 2 5 1 1 2 5 2 2 1 5 1 5 1 2  
## [3220] 2 5 1 5 2 5 1 5 5 5 5 5 2 2 1 5 5 5 1 5 5 5 1 1 1 1 5 2 5 5 2 1 5 1 5 5 5  
## [3257] 5 1 1 1 1 2 1 1 2 5 5 2 5 5 2 5 5 1 1 2 2 5 5 1 1 1 1 1 1 1 1 1 2 5 1 2 1  
## [3294] 5 5 1 2 2 2 5 1 5 5 5 2 5 1 4 1 5 5 5 5 5 2 5 5 5 5 2 2 1 2 5 5 2 1 5 2 5  
## [3331] 1 2 5 1 2 5 1 5 5 5 2 1 1 5 1 5 1 5 5 1 2 1 2 2 1 1 1 5 1 1 5 2 5 1 1 1 5  
## [3368] 5 2 1 5 1 1 5 1 1 5 2 2 5 5 1 3 2 1 2 2 5 2 5 5 5 1 5 5 5 2 1 5 1 5 1 5 1  
## [3405] 1 5 1 1 1 5 5 2 5 2 2 2 5 2 2 2 5 2 1 1 5 1 1 1 2 2 2 2 2 5 1 5 2 1 5 5 2  
## [3442] 1 3 2 1 1 1 1 2 2 2 2 5 2 1 4 5 5 1 5 1 2 1 5 1 1 1 2 1 5 5 1 1 5 1 5 5 1  
## [3479] 1 1 1 1 1 5 1 5 1 1 5 1 1 1 1 5 1 1 1 5 2 1 5 1 5 1 1 5 2 2 1 1 5 5 5 2 2  
## [3516] 1 5 1 1 1 1 5 5 5 5 5 5 5 1 1 1 5 1 5 1 1 2 5 1 1 1 1 1 5 1 1 5 2 5 1 4 2  
## [3553] 5 1 1 5 5 1 5 5 5 5 1 1 1 5 1 5 2 5 2 5 1 1 1 1 5 5 1 1 1 1 1 1 1 5 1 1 2  
## [3590] 5 1 5 2 1 1 1 1 1 1 5 5 5 2 5 1 1 5 5 2 2 5 5 1 5 5 2 5 2 1 5 5 1 5 1 5 5  
## [3627] 1 5 1 5 5 2 1 5 5 5 5 5 5 1 5 5 5 1 5 1 5 5 1 3 1 1 5 5 5 1 2 1 5 1 5 5 5  
## [3664] 1 5 5 5 2 2 5 5 5 1 1 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 1 1 4 1 1 1 5 1 1 5 1  
## [3701] 2 1 5 5 5 1 1 1 1 1 1 1 5 1 1 5 5 2 5 1 2 1 1 2 1 1 1 4 5 5 1 1 2 5 1 5 1  
## [3738] 5 1 1 1 1 2 1 1 5 1 1 2 2 1 5 5 1 1 5 2 5 5 5 2 5 2 1 5 5 1 1 1 1 1 5 1 1  
## [3775] 1 5 5 1 5 1 5 1 5 5 1 5 5 5 1 2 1 5 1 5 5 2 2 1 1 5 5 1 5 5 1 1 2 5 2 5 5  
## [3812] 2 5 5 1 5 5 5 2 5 1 2 1 2 1 1 1 1 5 1 5 5 5 5 5 1 1 5 5 1 1 1 1 1 2 1 5 5  
## [3849] 1 1 2 1 1 5 5 5 5 1 5 1 5 2 1 1 1 5 1 5 1 2 1 5 5 2 1 1 5 1 5 5 5 1 1 1 1  
## [3886] 5 5 1 2 2 5 5 2 2 1 2 1 2 2 2 1 1 1 5 2 5 1 5 5 1 2 5 2 2 1 2 1 1 2 2 4 5  
## [3923] 5 1 5 1 5 1 5 1 5 1 2 1 1 1 1 1 5 5 1 2 5 5 1 1 5 2 1 1 1 5 2 2 2 5 5 1 5  
## [3960] 1 5 5 1 5 5 1 1 1 5 2 1 5 5 2 2 2 5 1 5 5 5 2 2 2 1 5 5 5 2 2 5 5 5 1 5 1  
## [3997] 2 2 5 5 5 1 2 5 1 2 5 1 5 1 1 1 5 4 5 5 1 5 1 2 1 4 5 4 5 2 1 2 1 1 1 1 1  
## [4034] 1 5 1 5 1 1 5 2 1 5 1 5 5 5 5 5 1 1 2 2 1 1 5 5 1 1 1 5 5 1 5 1 1 5 5 5 5  
## [4071] 2 5 1 2 4 5 2 1 1 5 5 1 1 1 1 1 2 2 1 5 5 1 1 1 5 5 1 1 5 1 1 5 5 5 1 1 2  
## [4108] 1 1 1 1 5 5 2 2 5 1 2 1 1 2 2 5 1 2 2 5 5 1 5 5 5 1 2 5 1 5 5 5 1 5 1 1 5  
## [4145] 5 5 5 1 1 5 1 5 2 2 2 1 2 2 5 2 1 2 5 5 1 2 5 1 5 5 2 5 5 5 1 1 1 2 5 1 1  
## [4182] 1 1 1 1 1 5 1 4 1 2 5 1 2 5 5 2 5 1 2 1 2 2 2 2 5 1 1 2 1 1 2 1 2 1 1 1 2  
## [4219] 5 1 1 2 1 2 1 2 2 1 5 5 2 1 1 1 2 1 1 2 2 2 1 5 5 1 2 2 5 1 1 1 5 2 1 1 1  
## [4256] 1 2 1 2 5 2 2 2 2 5 2 2 3 1 2 2 1 2 2 2 2 2 5 5 2 1 2 5 2 2 2 1 1 5 5 5 5  
## [4293] 5 5 2 1 5 5 5 2 1 1 1 1 1 1 1 1 1 1 5 1 1 1 1 1 1 5 5 2 1 2 5 5 1 5 5 1 2  
## [4330] 1 1 5 2 1 1 5 5 1 2 1 1 5 5 5 2 5 1 2 1 2 2 2 2 5 1 1 5 5 5 5 5 5 2 1 5 2  
## [4367] 1 2 5 1 5 2 2 2 2 2 2 2 5 2 1 5 1 1 1 2 1 1 1 5 1 1 2 1 2 2 1 2 2 1 1 5 2  
## [4404] 2 5 1 1 1 2 2 5 5 2 2 5 5 2 1 2 5 1 5 2 1 5 1 2 2 1 1 1 2 2 2 5 1 1 2 2 2  
## [4441] 5 5 5 5 2 2 2 5 1 1 5 5 2 5 2 2 2 2 5 5 5 2 2 2 2 2 2 5 2 2 2 2 2 2 2 1 1  
## [4478] 2 2 5 2 5 2 2 2 5 5 5 2 2 5 1 2 2 5 1 1 5 1 5 2 1 2 2 5 1 2 2 2 1 1 2 5 2  
## [4515] 2 2 2 2 5 5 5 2 5 1 1 2 2 2 1 2 2 1 5 5 5 5 1 2 2 5 5 1 2 5 2 2 2 1 5 5 5  
## [4552] 5 5 2 5 2 2 5 2 1 5 1 2 5 2 1 1 5 2 5 5 5 1 1 5 2 1 1 2 2 5 1 2 2 2 2 2 2  
## [4589] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 5 2 2 1 1 5 1 2 2 5 5 5 4 1 1 2  
## [4626] 5 1 5 2 2 1 5 1 1 2 5 1 2 1 1 1 1 5 1 5 2 5 2 1 1 1 1 5 2 2 2 2 1 5 1 1 2  
## [4663] 1 1 1 1 5 2 2 2 2 1 2 5 2 5 1 2 5 5 1 2 2 2 5 2 1 2 2 2 1 2 2 3 3 2 5 2 2  
## [4700] 2 5 5 2 5 5 2 5 1 5 2 2 5 1 2 2 2 2 2 5 1 5 2 2 1 1 2 5 2 2 2 1 2 2 2 1 5  
## [4737] 2 5 5 1 2 2 5 1 1 1 1 1 1 2 1 2 2 2 2 1 2 2 5 2 1 2 2 2 2 2 2 2 5 5 2 2 2  
## [4774] 2 1 2 5 5 5 2 2 5 2 2 2 5 2 2 5 2 2 2 5 5 5 2 2 2 2 2 2 2 2 1 2 2 5 5 1 5  
## [4811] 1 5 5 2 2 5 5 5 2 3 2 2 1 2 2 5 1 1 2 1 2 5 2 5 2 2 2 5 2 2 2 2 2 2 5 2 2  
## [4848] 2 2 2 1 2 2 5 1 1 1 1 1 1 2 1 2 5 1 2 5 2 1 5 2 2 2 5 5 2 2 1 5 5 2 2 5 2  
## [4885] 2 2 2 5 1 5 2 1 2 5 1 1 1 2 2 5 2 2 2 2 2 2 1 5 2 2 2 2 2 1 2 2 2 2 5 1 2  
## [4922] 2 2 2 2 5 2 5 2 1 1 2 5 2 1 1 1 5 5 2 5 5 1 1 1 1 5 5 5 5 2 2 2 2 2 1 2 2  
## [4959] 2 2 5 5 2 2 2 2 2 5 2 2 2 2 2 5 1 5 2 2 1 2 2 2 2 5 2 2 1 2 2 2 2 2 2 1 1  
## [4996] 1 5 1 2 5 5 5 5 2 5 2 1 1 5 2 5 1 1 2 2 1 2 1 2 1 2 2 1 2 1 1 1 2 1 1 1 5  
## [5033] 2 2 2 2 5 1 5 2 5 2 2 1 2 1 5 2 2 2 2 5 2 2 2 5 2 2 1 1 2 2 2 1 2 2 1 2 1  
## [5070] 2 1 5 2 5 1 2 2 1 2 2 5 2 2 2 5 2 5 2 2 2 5 2 2 5 2 2 1 5 2 2 5 1 2 2 2 5  
## [5107] 2 1 1 1 2 2 2 2 2 2 2 5 2 1 2 1 1 2 2 2 1 4 2 1 1 2 1 1 1 2 5 2 2 2 2 1 2  
## [5144] 2 1 1 1 2 2 5 1 1 1 2 5 2 2 1 5 2 5 2 5 2 5 2 2 2 5 2 4 2 2 5 2 1 2 5 2 5  
## [5181] 2 2 5 2 2 2 1 1 1 2 5 2 2 2 2 2 1 2 1 1 2 5 2 2 1 1 2 1 1 1 2 5 1 1 2 1 5  
## [5218] 1 1 1 1 2 1 2 1 5 1 1 1 5 5 1 5 2 2 2 2 2 5 5 5 5 5 2 2 2 2 5 2 5 1 5 1 5  
## [5255] 2 2 2 2 2 2 5 4 2 2 2 1 2 5 2 1 2 2 2 5 5 2 2 1 2 1 2 2 1 2 1 5 5 2 2 2 5  
## [5292] 2 2 1 2 2 1 2 5 5 1 1 1 1 1 2 2 1 5 2 1 1 1 5 1 5 5 1 1 2 5 5 2 2 2 2 2 2  
## [5329] 5 1 1 5 2 1 2 5 5 5 1 1 1 1 1 1 1 5 1 1 5 1 5 5 5 2 5 5 5 5 5 5 2 5 5 2 1  
## [5366] 1 1 1 1 5 1 1 2 1 2 5 2 2 2 1 5 5 1 1 2 1 1 1 1 1 1 1 1 5 1 1 5 5 5 5 5 5  
## [5403] 5 5 5 5 5 5 5 1 5 5 5 1 2 5 2 2 2 5 1 2 1 2 2 2 5 2 2 2 5 5 1 5 5 2 2 5 5  
## [5440] 2 1 2 2 2 1 2 1 5 5 2 2 2 2 2 2 2 5 2 1 1 1 1 1 2 2 5 1 1 1 1 1 1 1 1 2 1  
## [5477] 5 1 4 1 1 5 2 1 2 2 2 1 1 2 5 5 5 2 2 5 1 2 5 5 4 2 2 2 2 2 2 2 2 2 5 2 5  
## [5514] 2 2 2 1 2 2 5 1 2 5 5 5 5 5 1 1 2 2 2 5 5 1 5 1 2 5 1 5 5 5 5 5 2 2 5 1 5  
## [5551] 1 2 2 1 2 5 5 5 5 1 1 2 2 1 5 5 1 1 5 1 5 5 1 1 2 2 2 2 5 1 2 2 5 2 5 5 1  
## [5588] 2 5 1 5 2 2 2 5 2 2 5 5 5 5 1 2 2 2 2 5 1 5 2 2 5 1 1 5 1 2 1 5 5 5 1 1 2  
## [5625] 2 1 5 2 5 5 5 1 5 1 5 5 5 5 4 5 1 1 1 1 1 1 1 1 2 1 2 1 1 1 2 1 2 1 2 5 5  
## [5662] 2 5 2 2 2 5 2 1 5 2 1 5 5 1 5 2 2 5 1 1 5 5 2 2 5 5 2 5 1 5 5 1 2 1 5 5 5  
## [5699] 5 1 1 1 2 2 2 5 1 2 2 5 2 2 2 2 2 1 1 2 2 1 1 1 2 2 5 5 1 1 2 2 1 1 5 2 5  
## [5736] 2 1 2 2 1 1 5 1 1 1 1 2 1 2 1 2 1 5 1 5 1 1 1 1 1 1 1 5 5 2 2 2 2 2 1 2 2  
## [5773] 5 1 2 5 5 2 1 1 2 1 2 2 1 5 2 2 5 2 5 5 2 5 2 2 5 1 2 2 2 2 5 1 2 1 2 2 2  
## [5810] 5 1 5 5 1 1 1 5 2 1 2 2 5 4 2 1 1 2 1 5 5 2 2 5 5 1 1 5 1 2 1 2 2 2 1 1 2  
## [5847] 5 2 5 1 5 1 4 5 5 5 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 1 1 1 1 2 5 5 1 1 1 1 5  
## [5884] 5 5 5 5 5 1 1 1 5 1 5 1 5 5 2 1 1 1 1 2 2 5 1 2 2 2 1 5 2 2 2 2 4 2 2 2 1  
## [5921] 5 2 2 2 1 1 1 5 1 1 1 1 1 1 1 1 1 1 2 1 1 2 5 5 5 5 2 1 1 2 2 5 2 5 2 5 2  
## [5958] 1 1 5 2 2 1 5 1 2 1 5 5 5 5 2 2 5 5 5 2 5 5 5 2 2 2 1 1 2 2 5 5 1 1 1 1 1  
## [5995] 1 1 1 1 1 1 2 2 5 1 1 2 2 1 2 5 2 2 2 2 1 5 5 1 1 1 1 5 5 1 5 5 1 2 2 1 2  
## [6032] 2 1 5 5 2 5 5 5 1 2 1 5 2 2 2 2 5 1 1 1 5 2 1 1 1 1 2 5 1 5 2 2 1 1 5 5 2  
## [6069] 2 2 2 5 5 1 2 2 1 1 5 1 1 2 5 5 2 5 2 2 2 2 2 2 1 2 1 2 1 5 2 5 1 2 2 5 5  
## [6106] 2 2 5 5 2 2 2 2 1 2 2 5 5 1 1 5 5 1 1 1 1 5 5 1 1 1 2 1 2 1 1 5 5 5 2 2 5  
## [6143] 5 2 2 2 2 1 5 2 2 2 2 2 1 5 1 5 2 2 2 2 2 2 5 1 5 2 5 2 2 2 2 2 2 5 1 2 1  
## [6180] 1 2 5 1 2 2 2 5 5 2 5 1 2 5 2 5 2 4 5 2 5 5 1 2 2 2 5 1 2 4 2 5 5 5 5 2 5  
## [6217] 2 2 5 2 2 2 2 2 5 1 2 2 2 1 2 1 5 1 5 5 2 5 1 2 2 2 5 2 2 2 2 5 4 4 2 5 2  
## [6254] 1 1 1 1 2 2 2 5 2 2 2 2 1 2 2 5 1 5 5 2 2 5 1 2 1 5 4 2 2 2 2 5 4 1 1 1 1  
## [6291] 1 5 1 1 2 2 2 5 1 1 4 4 2 1 5 2 5 5 2 2 5 5 2 2 5 2 5 5 5 2 5 5 1 5 1 2 5  
## [6328] 5 5 5 2 2 5 2 2 2 2 2 2 5 2 5 5 5 1 1 5 1 1 5 1 2 2 1 2 2 2 2 5 5 1 2 2 2  
## [6365] 2 2 5 1 1 1 1 1 2 5 5 5 2 1 5 5 5 1 1 1 2 2 2 1 2 2 1 4 1 1 2 2 2 1 5 2 2  
## [6402] 2 2 5 5 1 5 5 5 5 2 2 5 2 2 1 2 2 2 5 5 2 2 1 5 5 2 1 2 2 5 5 2 2 2 5 2 2  
## [6439] 4 2 2 2 5 2 5 2 2 2 1 2 1 5 5 2 1 1 5 1 5 5 2 2 2 5 2 2 2 5 5 2 2 1 2 5 5  
## [6476] 2 4 4 1 1 1 5 2 1 1 2 2 5 1 2 2 5 2 1 5 2 2  
##   
## Within cluster sum of squares by cluster:  
## [1] 9627.063 9064.628 4977.893 5708.210 9240.290  
## (between\_SS / total\_SS = 55.3 %)  
##   
## Available components:  
##   
## [1] "cluster" "centers" "totss" "withinss" "tot.withinss"  
## [6] "betweenss" "size" "iter" "ifault"

Now we can start interpreting the cluster results:

Cluster 1: 1.It looks to be a higher residual and high with respect to citric acid and good with sulphur dioxide content.

Cluster 2: It represents above average in pH , sulphur dioxide and citric acid. Quality percent high compared to others.

Cluster 3 is dominant in the fixed acidity volatile acidity citric acid, good in density, sulphates and quality.

Cluster 4 is high volatile acidity.

Cluster 5 has low levels in all of its contents.

The clusters for white wine were found to be primarily distinguished by alcohol content and pH, with one cluster having a significantly higher alcohol content and lower pH compared to the others. The other two clusters had similar alcohol content and pH levels.

Overall, the k means clustering analysis showed that there are distinct groups within the winequality dataset, with the primary differences being alcohol content and pH levels. These findings could potentially be used to inform the production and marketing of different types of wine.

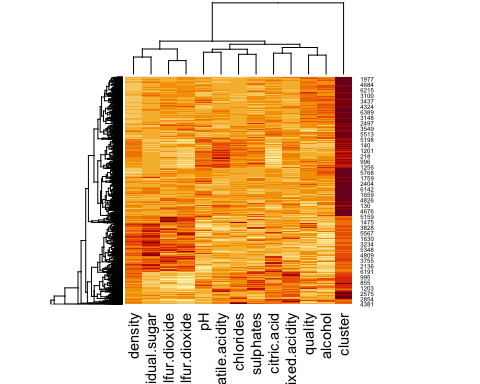
kmeans.re$centers

## fixed.acidity volatile.acidity citric.acid residual.sugar free.sulfur.dioxide  
## 1 -0.1846683 -0.3564474 0.29319560 1.3843044 0.94550277  
## 2 -0.4561617 -0.3797596 0.03181511 -0.3856796 0.04339946  
## 3 2.0838769 0.4322947 1.02553445 -0.5709293 -0.91739421  
## 4 0.1058135 1.6872984 -1.24413920 -0.6192310 -0.78606731  
## 5 -0.1886029 -0.4368905 0.05309929 -0.3647284 -0.14751921  
## total.sulfur.dioxide density pH sulphates alcohol  
## 1 1.02642311 0.8794364 -0.42078939 -0.2449341 -0.84490933  
## 2 -0.04019979 -1.1399987 0.01731967 -0.2407335 1.17049279  
## 3 -1.31976067 0.9752074 -0.07050727 1.4407145 0.09609325  
## 4 -1.12582850 0.5132103 0.94103422 0.4272302 -0.25927961  
## 5 0.20350682 -0.3657263 -0.14621701 -0.3012584 -0.23347426  
## quality cluster  
## 1 -0.24099276 1.004405  
## 2 0.94548409 4.998787  
## 3 0.09783685 2.000000  
## 4 -0.52792960 2.995939  
## 5 -0.42797771 3.994614

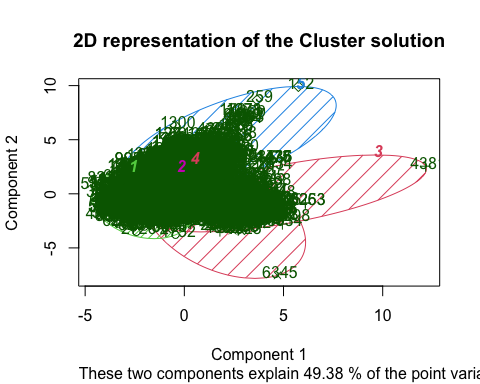
kmeans.re$centers[, c("total.sulfur.dioxide", "free.sulfur.dioxide")]

## total.sulfur.dioxide free.sulfur.dioxide  
## 1 1.02642311 0.94550277  
## 2 -0.04019979 0.04339946  
## 3 -1.31976067 -0.91739421  
## 4 -1.12582850 -0.78606731  
## 5 0.20350682 -0.14751921

heatmap(as.matrix(wines3))



library(cluster)  
clusplot(wines3, y\_kmeans, main='2D representation of the Cluster solution',  
 color=TRUE, shade=TRUE,  
 labels=2, lines=0)



In order to evaluate the clustering performance we build a confusion matrix:

kmeans.re$cluster

## [1] 4 4 4 3 4 4 4 4 4 4 4 4 4 3 4 4 3 3 4 3 3 4 3 4 4 4 4 3 4 4 4 4 4 4 4 4 4  
## [38] 3 4 4 4 4 4 4 4 4 4 3 4 5 4 4 4 5 4 4 3 4 4 4 3 3 4 4 4 4 4 4 3 4 4 4 4 4  
## [75] 3 3 3 4 4 4 4 3 4 4 5 4 3 4 3 4 5 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 3 4 3 3 4  
## [112] 4 4 3 4 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 4 4 4 4 4 4 4 4 4 4 4 2 4 2 3 4 4  
## [149] 4 3 3 3 4 4 5 5 5 5 4 4 4 4 4 4 4 3 4 4 4 3 4 4 4 4 4 4 4 4 4 4 4 3 4 4 4  
## [186] 3 5 4 5 5 4 4 4 4 4 4 4 3 4 4 3 3 4 4 4 3 3 4 4 3 3 4 3 4 4 5 4 4 4 4 5 4  
## [223] 4 4 4 4 3 4 4 4 2 4 4 4 4 4 4 4 4 4 3 3 4 3 3 4 4 4 4 4 3 4 3 4 4 4 3 4 3  
## [260] 3 4 4 4 4 3 3 4 3 4 3 4 3 3 4 4 4 4 3 3 3 3 3 4 3 4 4 3 4 4 3 4 3 3 4 3 3  
## [297] 3 4 4 4 4 3 4 4 3 3 4 3 3 5 3 4 3 5 4 4 4 4 3 4 3 4 4 3 3 3 3 3 3 3 3 3 4  
## [334] 4 4 3 3 4 3 3 3 3 3 3 3 4 4 3 3 4 3 4 4 3 2 4 3 3 3 3 4 3 3 3 3 3 3 3 3 3  
## [371] 4 3 3 4 3 3 3 3 3 3 3 3 3 3 4 4 4 4 4 3 4 3 3 3 3 3 4 3 3 4 4 5 3 3 4 3 3  
## [408] 3 3 3 3 3 4 3 4 4 3 4 3 4 3 4 4 3 4 4 4 4 3 3 3 4 3 3 3 3 4 3 3 4 3 3 3 3  
## [445] 4 4 3 3 4 3 3 3 4 3 5 3 3 4 3 3 3 4 3 3 3 3 3 3 3 4 3 3 3 3 3 4 3 3 4 4 3  
## [482] 3 3 3 3 3 3 3 3 3 4 3 3 4 2 3 4 5 3 4 4 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3  
## [519] 3 3 3 3 3 3 3 3 3 3 3 4 3 3 3 3 3 3 4 4 3 3 4 3 4 3 3 3 4 3 3 3 4 3 3 4 3  
## [556] 3 3 3 3 3 3 3 3 3 3 3 4 4 3 4 3 4 3 3 3 3 4 3 3 3 3 3 3 3 3 4 3 4 2 3 3 2  
## [593] 3 3 4 3 3 3 4 3 4 3 4 3 4 4 3 3 3 4 3 3 4 3 3 3 3 3 3 3 4 4 4 4 4 4 4 4 4  
## [630] 4 4 3 4 4 5 4 4 4 4 3 3 3 3 3 3 4 4 4 3 5 3 4 3 3 3 4 3 3 4 4 4 4 4 3 3 4  
## [667] 3 3 3 3 4 4 4 4 3 3 3 4 4 3 3 4 4 4 4 4 4 4 4 3 4 4 3 4 4 4 4 4 4 3 3 4 4  
## [704] 5 4 4 4 4 4 3 3 4 4 4 4 4 4 4 4 4 4 4 4 3 4 4 4 4 4 4 3 4 4 4 4 4 4 4 4 4  
## [741] 4 4 4 3 3 4 3 3 4 4 4 4 4 4 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 4 4  
## [778] 4 4 4 4 4 4 4 4 3 3 3 3 4 3 4 4 4 3 3 3 3 3 3 4 4 4 4 4 2 2 2 4 4 4 3 3 4  
## [815] 3 3 3 3 4 4 4 2 4 4 4 4 2 4 4 4 4 4 3 3 4 4 2 2 3 4 3 4 3 4 3 4 4 4 4 4 3  
## [852] 3 3 3 3 4 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 3 4 4 4 4 4 4 3 4 4 4 4 3  
## [889] 4 3 4 4 3 4 4 4 3 4 3 4 3 4 4 4 4 4 4 4 4 2 3 3 3 3 2 3 4 4 3 4 3 3 4 4 3  
## [926] 3 3 4 3 3 4 4 4 4 4 3 3 3 2 4 3 3 3 3 3 3 3 3 3 3 3 3 3 3 4 3 3 3 4 4 3 4  
## [963] 4 3 3 3 3 4 3 4 3 3 3 3 3 4 4 4 2 3 3 4 2 3 3 4 3 4 4 3 4 4 4 4 3 4 4 4 4  
## [1000] 4 4 3 3 2 4 2 3 3 3 3 3 3 4 4 4 3 3 2 2 4 3 3 4 3 4 4 2 4 5 4 4 4 4 4 4 3  
## [1037] 2 4 3 3 4 4 3 3 2 4 4 4 3 3 4 3 4 3 4 4 3 4 3 3 3 3 3 3 4 4 4 3 3 4 3 4 4  
## [1074] 4 4 3 3 3 3 2 3 2 4 3 4 4 3 5 3 3 3 3 4 3 4 3 4 3 3 3 3 4 4 4 3 4 2 3 4 3  
## [1111] 4 4 3 3 2 4 4 4 2 4 2 4 4 3 4 2 2 4 3 3 4 5 2 4 3 2 3 3 4 4 4 3 4 4 4 3 4  
## [1148] 3 3 3 3 4 4 3 4 4 3 2 3 3 3 3 3 4 4 3 3 3 2 4 3 4 3 4 4 4 4 4 4 3 3 3 3 4  
## [1185] 4 4 4 4 4 4 3 4 2 4 4 4 4 4 5 4 4 2 3 4 3 3 3 3 3 3 4 4 4 3 3 3 4 2 3 3 3  
## [1222] 3 4 3 3 4 4 4 2 4 3 4 4 3 4 5 4 4 4 4 4 3 3 4 1 4 4 4 4 4 4 4 4 4 4 4 4 4  
## [1259] 4 4 3 4 3 4 3 4 4 3 4 2 2 4 4 4 4 4 3 4 4 3 4 4 4 4 4 3 2 4 4 4 4 4 4 4 4  
## [1296] 4 4 4 4 4 4 4 3 5 4 4 4 4 4 4 4 4 4 4 4 4 4 3 4 3 3 4 3 3 4 4 4 4 4 4 4 4  
## [1333] 4 4 4 4 4 4 4 4 4 4 4 4 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 4 3 4 4 4 4 3 4  
## [1370] 4 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 3 4 3  
## [1407] 3 4 3 4 4 4 3 3 3 4 3 2 4 4 4 4 4 4 3 3 2 4 4 3 4 4 4 4 3 3 3 4 4 4 2 4 4  
## [1444] 4 4 4 4 4 4 2 2 3 4 4 3 4 4 4 3 3 4 4 4 4 4 4 4 4 4 4 4 4 3 4 3 2 3 2 4 3  
## [1481] 4 3 4 3 4 4 4 4 4 4 2 4 4 4 4 4 4 4 4 4 4 4 4 4 3 4 4 3 3 3 4 4 4 4 4 4 4  
## [1518] 4 3 4 4 4 4 4 4 4 4 4 3 4 4 4 4 4 4 4 4 4 4 4 4 2 4 3 3 4 4 4 3 2 4 4 4 4  
## [1555] 4 4 4 4 1 4 4 4 4 4 4 4 5 4 4 4 3 4 4 4 1 4 3 4 4 4 4 4 4 4 2 3 3 4 4 4 4  
## [1592] 4 4 4 4 4 4 4 5 1 5 5 1 1 5 5 1 5 5 5 5 5 2 1 2 5 2 5 5 2 2 2 4 5 1 5 5 2  
## [1629] 2 1 5 5 5 1 2 5 2 1 1 5 5 5 5 5 5 5 1 5 5 2 2 2 2 5 5 1 1 5 5 5 1 5 5 5 5  
## [1666] 2 5 2 5 1 1 5 5 2 5 2 2 5 5 5 1 5 1 1 1 1 1 1 1 1 1 2 2 2 1 1 2 5 1 1 1 1  
## [1703] 1 1 1 1 1 1 5 1 1 1 1 1 4 2 2 1 1 5 1 5 5 5 2 5 1 2 5 5 1 1 1 1 1 5 1 2 2  
## [1740] 5 1 5 5 5 2 5 4 2 5 2 1 5 5 3 1 1 2 2 2 2 1 5 1 1 1 1 2 1 5 2 5 5 2 1 2 5  
## [1777] 1 4 5 5 5 1 1 1 1 1 5 2 1 1 1 5 5 1 1 1 1 1 1 1 1 1 2 5 1 5 3 4 5 5 2 5 5  
## [1814] 2 1 5 1 1 5 5 1 2 5 5 1 5 1 5 1 4 1 1 1 1 1 1 1 2 1 1 5 2 5 1 2 2 2 2 5 5  
## [1851] 1 1 5 5 2 2 2 2 5 5 1 5 1 5 1 1 4 4 4 1 4 1 5 1 1 1 5 5 2 2 2 5 1 1 1 1 1  
## [1888] 1 1 1 5 1 5 5 5 1 2 1 5 4 5 4 2 5 5 5 5 2 5 2 2 5 5 5 5 5 5 2 5 2 5 5 5 5  
## [1925] 1 1 5 1 5 2 5 2 5 5 2 2 1 5 2 1 5 1 5 5 5 2 1 1 1 2 2 5 2 5 1 1 5 5 5 1 5  
## [1962] 5 1 2 2 5 5 5 5 5 5 4 5 2 2 2 2 5 2 2 1 1 5 2 2 2 1 5 1 1 5 4 2 1 5 5 2 1  
## [1999] 2 2 1 5 1 2 5 2 5 5 5 1 1 5 5 1 1 5 1 5 5 2 1 1 1 1 1 1 1 5 1 1 5 1 4 2 2  
## [2036] 5 2 1 5 2 5 2 1 1 5 2 1 1 2 1 5 2 2 2 1 2 5 5 1 5 1 1 5 1 1 1 5 1 1 1 5 2  
## [2073] 5 5 1 2 5 5 5 1 2 1 1 1 5 5 5 5 1 2 2 1 2 5 2 1 2 5 1 1 1 5 5 1 1 5 2 2 2  
## [2110] 5 2 5 2 5 5 5 5 2 5 5 5 2 2 5 5 5 2 2 2 5 1 1 1 1 1 1 1 5 5 5 1 5 5 1 5 5  
## [2147] 2 2 1 1 2 2 1 2 2 1 5 1 2 1 5 1 2 5 1 5 5 5 5 1 2 1 2 2 5 5 2 2 2 1 5 5 5  
## [2184] 2 5 1 2 2 5 1 2 1 5 5 2 1 5 2 2 5 5 5 5 5 5 5 5 1 4 2 5 1 1 1 1 2 5 5 5 1  
## [2221] 1 5 5 5 2 4 1 1 5 1 1 5 1 2 5 2 1 5 5 2 1 1 1 1 1 5 1 5 1 1 1 1 1 2 2 5 5  
## [2258] 5 1 5 2 4 5 1 5 2 1 5 1 1 2 2 5 1 1 1 2 2 2 5 5 1 1 5 1 2 1 1 2 5 5 5 5 1  
## [2295] 1 2 1 1 1 1 2 5 5 2 5 1 5 1 5 1 2 2 1 5 2 1 1 2 2 1 5 2 2 2 5 2 5 5 5 1 1  
## [2332] 2 1 1 5 1 1 2 5 5 2 5 1 5 2 1 2 1 2 2 1 1 1 5 2 1 1 2 2 1 1 1 1 5 5 1 2 1  
## [2369] 1 5 5 5 1 2 2 5 1 1 1 4 1 1 1 1 1 1 1 2 1 1 2 5 2 5 1 1 1 5 2 1 1 1 1 5 1  
## [2406] 1 1 1 1 1 5 5 1 1 5 5 1 5 1 2 1 5 5 5 1 5 2 2 2 5 5 2 2 5 2 2 2 2 2 2 1 1  
## [2443] 2 2 2 1 5 1 2 2 1 5 1 5 2 1 1 1 1 2 1 5 2 5 2 5 1 1 5 1 1 5 5 5 2 2 5 5 2  
## [2480] 2 5 2 2 1 2 2 2 2 2 5 2 5 2 2 5 1 5 5 1 2 1 1 1 5 5 5 2 1 1 5 5 5 5 5 5 5  
## [2517] 5 2 5 5 1 1 1 2 5 4 5 1 1 2 1 1 1 1 1 5 1 1 1 1 1 2 2 5 2 1 5 4 5 5 1 5 5  
## [2554] 1 5 5 2 2 1 1 2 1 2 1 5 5 1 2 2 2 2 1 2 2 5 5 1 2 5 5 2 2 2 5 5 5 5 5 2 5  
## [2591] 5 1 5 5 5 5 1 1 5 5 1 1 2 2 5 2 4 2 5 5 2 5 1 1 5 1 5 5 5 2 2 2 1 2 5 5 4  
## [2628] 2 1 5 1 1 1 1 5 5 4 5 5 4 5 5 1 2 5 5 5 5 2 1 5 5 3 5 5 5 1 5 5 2 1 5 5 5  
## [2665] 1 1 2 2 5 2 1 2 5 1 2 1 5 1 5 5 5 5 1 2 1 2 2 1 2 1 1 5 5 1 2 1 2 1 2 5 2  
## [2702] 5 5 1 2 2 2 2 5 2 2 1 5 4 2 4 1 4 1 1 5 2 5 5 1 2 2 5 2 5 5 1 5 5 1 2 2 5  
## [2739] 5 5 1 5 5 5 1 5 5 5 1 1 1 4 2 5 5 1 1 1 1 5 1 5 1 5 5 5 5 5 5 5 5 2 5 1 1  
## [2776] 1 5 5 5 5 5 5 5 2 1 5 2 5 5 2 2 1 1 1 1 2 1 1 2 2 2 1 2 2 5 1 1 2 2 5 5 5  
## [2813] 2 5 5 2 1 2 2 5 2 5 5 2 2 2 5 2 5 1 2 2 5 5 5 2 5 2 5 5 2 2 2 5 1 5 2 2 1  
## [2850] 2 5 5 2 5 5 5 1 1 1 1 5 5 1 5 2 2 1 1 1 1 5 5 5 1 5 1 1 5 5 2 2 1 2 2 2 2  
## [2887] 2 2 2 5 1 5 5 5 1 5 2 2 5 1 5 1 1 1 2 2 5 5 3 1 5 5 5 1 1 5 5 5 1 2 2 5 5  
## [2924] 5 2 1 5 2 5 5 5 1 2 2 5 2 1 1 1 5 5 1 1 2 2 2 2 2 5 2 2 5 1 5 5 1 1 2 2 5  
## [2961] 2 2 5 5 2 2 5 1 1 5 2 1 1 2 2 5 5 2 5 2 2 5 5 5 4 2 2 2 5 5 2 2 5 2 2 2 1  
## [2998] 1 2 1 1 2 2 5 5 2 2 1 5 2 2 2 5 2 4 5 1 5 2 1 2 5 1 5 2 2 5 2 5 5 5 2 2 5  
## [3035] 1 1 2 1 1 1 5 5 2 1 2 2 5 5 2 1 5 1 1 1 1 2 2 2 5 2 5 5 5 2 5 2 2 5 5 1 1  
## [3072] 5 5 5 5 1 5 5 5 5 5 5 5 5 5 5 1 1 1 5 5 1 2 2 2 4 2 2 2 2 5 5 2 2 5 1 1 1  
## [3109] 5 5 5 2 1 2 5 1 1 2 5 5 5 5 1 1 1 3 1 5 1 1 2 1 5 1 5 5 5 5 5 1 5 5 5 5 5  
## [3146] 5 5 2 5 5 5 2 2 2 2 1 5 5 5 2 5 2 2 5 5 1 5 1 5 5 1 1 5 5 5 5 5 1 1 1 5 2  
## [3183] 1 1 1 5 1 1 2 2 5 5 5 5 1 2 1 5 2 5 1 2 2 5 2 2 5 1 1 2 5 2 2 1 5 1 5 1 2  
## [3220] 2 5 1 5 2 5 1 5 5 5 5 5 2 2 1 5 5 5 1 5 5 5 1 1 1 1 5 2 5 5 2 1 5 1 5 5 5  
## [3257] 5 1 1 1 1 2 1 1 2 5 5 2 5 5 2 5 5 1 1 2 2 5 5 1 1 1 1 1 1 1 1 1 2 5 1 2 1  
## [3294] 5 5 1 2 2 2 5 1 5 5 5 2 5 1 4 1 5 5 5 5 5 2 5 5 5 5 2 2 1 2 5 5 2 1 5 2 5  
## [3331] 1 2 5 1 2 5 1 5 5 5 2 1 1 5 1 5 1 5 5 1 2 1 2 2 1 1 1 5 1 1 5 2 5 1 1 1 5  
## [3368] 5 2 1 5 1 1 5 1 1 5 2 2 5 5 1 3 2 1 2 2 5 2 5 5 5 1 5 5 5 2 1 5 1 5 1 5 1  
## [3405] 1 5 1 1 1 5 5 2 5 2 2 2 5 2 2 2 5 2 1 1 5 1 1 1 2 2 2 2 2 5 1 5 2 1 5 5 2  
## [3442] 1 3 2 1 1 1 1 2 2 2 2 5 2 1 4 5 5 1 5 1 2 1 5 1 1 1 2 1 5 5 1 1 5 1 5 5 1  
## [3479] 1 1 1 1 1 5 1 5 1 1 5 1 1 1 1 5 1 1 1 5 2 1 5 1 5 1 1 5 2 2 1 1 5 5 5 2 2  
## [3516] 1 5 1 1 1 1 5 5 5 5 5 5 5 1 1 1 5 1 5 1 1 2 5 1 1 1 1 1 5 1 1 5 2 5 1 4 2  
## [3553] 5 1 1 5 5 1 5 5 5 5 1 1 1 5 1 5 2 5 2 5 1 1 1 1 5 5 1 1 1 1 1 1 1 5 1 1 2  
## [3590] 5 1 5 2 1 1 1 1 1 1 5 5 5 2 5 1 1 5 5 2 2 5 5 1 5 5 2 5 2 1 5 5 1 5 1 5 5  
## [3627] 1 5 1 5 5 2 1 5 5 5 5 5 5 1 5 5 5 1 5 1 5 5 1 3 1 1 5 5 5 1 2 1 5 1 5 5 5  
## [3664] 1 5 5 5 2 2 5 5 5 1 1 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 1 1 4 1 1 1 5 1 1 5 1  
## [3701] 2 1 5 5 5 1 1 1 1 1 1 1 5 1 1 5 5 2 5 1 2 1 1 2 1 1 1 4 5 5 1 1 2 5 1 5 1  
## [3738] 5 1 1 1 1 2 1 1 5 1 1 2 2 1 5 5 1 1 5 2 5 5 5 2 5 2 1 5 5 1 1 1 1 1 5 1 1  
## [3775] 1 5 5 1 5 1 5 1 5 5 1 5 5 5 1 2 1 5 1 5 5 2 2 1 1 5 5 1 5 5 1 1 2 5 2 5 5  
## [3812] 2 5 5 1 5 5 5 2 5 1 2 1 2 1 1 1 1 5 1 5 5 5 5 5 1 1 5 5 1 1 1 1 1 2 1 5 5  
## [3849] 1 1 2 1 1 5 5 5 5 1 5 1 5 2 1 1 1 5 1 5 1 2 1 5 5 2 1 1 5 1 5 5 5 1 1 1 1  
## [3886] 5 5 1 2 2 5 5 2 2 1 2 1 2 2 2 1 1 1 5 2 5 1 5 5 1 2 5 2 2 1 2 1 1 2 2 4 5  
## [3923] 5 1 5 1 5 1 5 1 5 1 2 1 1 1 1 1 5 5 1 2 5 5 1 1 5 2 1 1 1 5 2 2 2 5 5 1 5  
## [3960] 1 5 5 1 5 5 1 1 1 5 2 1 5 5 2 2 2 5 1 5 5 5 2 2 2 1 5 5 5 2 2 5 5 5 1 5 1  
## [3997] 2 2 5 5 5 1 2 5 1 2 5 1 5 1 1 1 5 4 5 5 1 5 1 2 1 4 5 4 5 2 1 2 1 1 1 1 1  
## [4034] 1 5 1 5 1 1 5 2 1 5 1 5 5 5 5 5 1 1 2 2 1 1 5 5 1 1 1 5 5 1 5 1 1 5 5 5 5  
## [4071] 2 5 1 2 4 5 2 1 1 5 5 1 1 1 1 1 2 2 1 5 5 1 1 1 5 5 1 1 5 1 1 5 5 5 1 1 2  
## [4108] 1 1 1 1 5 5 2 2 5 1 2 1 1 2 2 5 1 2 2 5 5 1 5 5 5 1 2 5 1 5 5 5 1 5 1 1 5  
## [4145] 5 5 5 1 1 5 1 5 2 2 2 1 2 2 5 2 1 2 5 5 1 2 5 1 5 5 2 5 5 5 1 1 1 2 5 1 1  
## [4182] 1 1 1 1 1 5 1 4 1 2 5 1 2 5 5 2 5 1 2 1 2 2 2 2 5 1 1 2 1 1 2 1 2 1 1 1 2  
## [4219] 5 1 1 2 1 2 1 2 2 1 5 5 2 1 1 1 2 1 1 2 2 2 1 5 5 1 2 2 5 1 1 1 5 2 1 1 1  
## [4256] 1 2 1 2 5 2 2 2 2 5 2 2 3 1 2 2 1 2 2 2 2 2 5 5 2 1 2 5 2 2 2 1 1 5 5 5 5  
## [4293] 5 5 2 1 5 5 5 2 1 1 1 1 1 1 1 1 1 1 5 1 1 1 1 1 1 5 5 2 1 2 5 5 1 5 5 1 2  
## [4330] 1 1 5 2 1 1 5 5 1 2 1 1 5 5 5 2 5 1 2 1 2 2 2 2 5 1 1 5 5 5 5 5 5 2 1 5 2  
## [4367] 1 2 5 1 5 2 2 2 2 2 2 2 5 2 1 5 1 1 1 2 1 1 1 5 1 1 2 1 2 2 1 2 2 1 1 5 2  
## [4404] 2 5 1 1 1 2 2 5 5 2 2 5 5 2 1 2 5 1 5 2 1 5 1 2 2 1 1 1 2 2 2 5 1 1 2 2 2  
## [4441] 5 5 5 5 2 2 2 5 1 1 5 5 2 5 2 2 2 2 5 5 5 2 2 2 2 2 2 5 2 2 2 2 2 2 2 1 1  
## [4478] 2 2 5 2 5 2 2 2 5 5 5 2 2 5 1 2 2 5 1 1 5 1 5 2 1 2 2 5 1 2 2 2 1 1 2 5 2  
## [4515] 2 2 2 2 5 5 5 2 5 1 1 2 2 2 1 2 2 1 5 5 5 5 1 2 2 5 5 1 2 5 2 2 2 1 5 5 5  
## [4552] 5 5 2 5 2 2 5 2 1 5 1 2 5 2 1 1 5 2 5 5 5 1 1 5 2 1 1 2 2 5 1 2 2 2 2 2 2  
## [4589] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 5 2 2 1 1 5 1 2 2 5 5 5 4 1 1 2  
## [4626] 5 1 5 2 2 1 5 1 1 2 5 1 2 1 1 1 1 5 1 5 2 5 2 1 1 1 1 5 2 2 2 2 1 5 1 1 2  
## [4663] 1 1 1 1 5 2 2 2 2 1 2 5 2 5 1 2 5 5 1 2 2 2 5 2 1 2 2 2 1 2 2 3 3 2 5 2 2  
## [4700] 2 5 5 2 5 5 2 5 1 5 2 2 5 1 2 2 2 2 2 5 1 5 2 2 1 1 2 5 2 2 2 1 2 2 2 1 5  
## [4737] 2 5 5 1 2 2 5 1 1 1 1 1 1 2 1 2 2 2 2 1 2 2 5 2 1 2 2 2 2 2 2 2 5 5 2 2 2  
## [4774] 2 1 2 5 5 5 2 2 5 2 2 2 5 2 2 5 2 2 2 5 5 5 2 2 2 2 2 2 2 2 1 2 2 5 5 1 5  
## [4811] 1 5 5 2 2 5 5 5 2 3 2 2 1 2 2 5 1 1 2 1 2 5 2 5 2 2 2 5 2 2 2 2 2 2 5 2 2  
## [4848] 2 2 2 1 2 2 5 1 1 1 1 1 1 2 1 2 5 1 2 5 2 1 5 2 2 2 5 5 2 2 1 5 5 2 2 5 2  
## [4885] 2 2 2 5 1 5 2 1 2 5 1 1 1 2 2 5 2 2 2 2 2 2 1 5 2 2 2 2 2 1 2 2 2 2 5 1 2  
## [4922] 2 2 2 2 5 2 5 2 1 1 2 5 2 1 1 1 5 5 2 5 5 1 1 1 1 5 5 5 5 2 2 2 2 2 1 2 2  
## [4959] 2 2 5 5 2 2 2 2 2 5 2 2 2 2 2 5 1 5 2 2 1 2 2 2 2 5 2 2 1 2 2 2 2 2 2 1 1  
## [4996] 1 5 1 2 5 5 5 5 2 5 2 1 1 5 2 5 1 1 2 2 1 2 1 2 1 2 2 1 2 1 1 1 2 1 1 1 5  
## [5033] 2 2 2 2 5 1 5 2 5 2 2 1 2 1 5 2 2 2 2 5 2 2 2 5 2 2 1 1 2 2 2 1 2 2 1 2 1  
## [5070] 2 1 5 2 5 1 2 2 1 2 2 5 2 2 2 5 2 5 2 2 2 5 2 2 5 2 2 1 5 2 2 5 1 2 2 2 5  
## [5107] 2 1 1 1 2 2 2 2 2 2 2 5 2 1 2 1 1 2 2 2 1 4 2 1 1 2 1 1 1 2 5 2 2 2 2 1 2  
## [5144] 2 1 1 1 2 2 5 1 1 1 2 5 2 2 1 5 2 5 2 5 2 5 2 2 2 5 2 4 2 2 5 2 1 2 5 2 5  
## [5181] 2 2 5 2 2 2 1 1 1 2 5 2 2 2 2 2 1 2 1 1 2 5 2 2 1 1 2 1 1 1 2 5 1 1 2 1 5  
## [5218] 1 1 1 1 2 1 2 1 5 1 1 1 5 5 1 5 2 2 2 2 2 5 5 5 5 5 2 2 2 2 5 2 5 1 5 1 5  
## [5255] 2 2 2 2 2 2 5 4 2 2 2 1 2 5 2 1 2 2 2 5 5 2 2 1 2 1 2 2 1 2 1 5 5 2 2 2 5  
## [5292] 2 2 1 2 2 1 2 5 5 1 1 1 1 1 2 2 1 5 2 1 1 1 5 1 5 5 1 1 2 5 5 2 2 2 2 2 2  
## [5329] 5 1 1 5 2 1 2 5 5 5 1 1 1 1 1 1 1 5 1 1 5 1 5 5 5 2 5 5 5 5 5 5 2 5 5 2 1  
## [5366] 1 1 1 1 5 1 1 2 1 2 5 2 2 2 1 5 5 1 1 2 1 1 1 1 1 1 1 1 5 1 1 5 5 5 5 5 5  
## [5403] 5 5 5 5 5 5 5 1 5 5 5 1 2 5 2 2 2 5 1 2 1 2 2 2 5 2 2 2 5 5 1 5 5 2 2 5 5  
## [5440] 2 1 2 2 2 1 2 1 5 5 2 2 2 2 2 2 2 5 2 1 1 1 1 1 2 2 5 1 1 1 1 1 1 1 1 2 1  
## [5477] 5 1 4 1 1 5 2 1 2 2 2 1 1 2 5 5 5 2 2 5 1 2 5 5 4 2 2 2 2 2 2 2 2 2 5 2 5  
## [5514] 2 2 2 1 2 2 5 1 2 5 5 5 5 5 1 1 2 2 2 5 5 1 5 1 2 5 1 5 5 5 5 5 2 2 5 1 5  
## [5551] 1 2 2 1 2 5 5 5 5 1 1 2 2 1 5 5 1 1 5 1 5 5 1 1 2 2 2 2 5 1 2 2 5 2 5 5 1  
## [5588] 2 5 1 5 2 2 2 5 2 2 5 5 5 5 1 2 2 2 2 5 1 5 2 2 5 1 1 5 1 2 1 5 5 5 1 1 2  
## [5625] 2 1 5 2 5 5 5 1 5 1 5 5 5 5 4 5 1 1 1 1 1 1 1 1 2 1 2 1 1 1 2 1 2 1 2 5 5  
## [5662] 2 5 2 2 2 5 2 1 5 2 1 5 5 1 5 2 2 5 1 1 5 5 2 2 5 5 2 5 1 5 5 1 2 1 5 5 5  
## [5699] 5 1 1 1 2 2 2 5 1 2 2 5 2 2 2 2 2 1 1 2 2 1 1 1 2 2 5 5 1 1 2 2 1 1 5 2 5  
## [5736] 2 1 2 2 1 1 5 1 1 1 1 2 1 2 1 2 1 5 1 5 1 1 1 1 1 1 1 5 5 2 2 2 2 2 1 2 2  
## [5773] 5 1 2 5 5 2 1 1 2 1 2 2 1 5 2 2 5 2 5 5 2 5 2 2 5 1 2 2 2 2 5 1 2 1 2 2 2  
## [5810] 5 1 5 5 1 1 1 5 2 1 2 2 5 4 2 1 1 2 1 5 5 2 2 5 5 1 1 5 1 2 1 2 2 2 1 1 2  
## [5847] 5 2 5 1 5 1 4 5 5 5 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 1 1 1 1 2 5 5 1 1 1 1 5  
## [5884] 5 5 5 5 5 1 1 1 5 1 5 1 5 5 2 1 1 1 1 2 2 5 1 2 2 2 1 5 2 2 2 2 4 2 2 2 1  
## [5921] 5 2 2 2 1 1 1 5 1 1 1 1 1 1 1 1 1 1 2 1 1 2 5 5 5 5 2 1 1 2 2 5 2 5 2 5 2  
## [5958] 1 1 5 2 2 1 5 1 2 1 5 5 5 5 2 2 5 5 5 2 5 5 5 2 2 2 1 1 2 2 5 5 1 1 1 1 1  
## [5995] 1 1 1 1 1 1 2 2 5 1 1 2 2 1 2 5 2 2 2 2 1 5 5 1 1 1 1 5 5 1 5 5 1 2 2 1 2  
## [6032] 2 1 5 5 2 5 5 5 1 2 1 5 2 2 2 2 5 1 1 1 5 2 1 1 1 1 2 5 1 5 2 2 1 1 5 5 2  
## [6069] 2 2 2 5 5 1 2 2 1 1 5 1 1 2 5 5 2 5 2 2 2 2 2 2 1 2 1 2 1 5 2 5 1 2 2 5 5  
## [6106] 2 2 5 5 2 2 2 2 1 2 2 5 5 1 1 5 5 1 1 1 1 5 5 1 1 1 2 1 2 1 1 5 5 5 2 2 5  
## [6143] 5 2 2 2 2 1 5 2 2 2 2 2 1 5 1 5 2 2 2 2 2 2 5 1 5 2 5 2 2 2 2 2 2 5 1 2 1  
## [6180] 1 2 5 1 2 2 2 5 5 2 5 1 2 5 2 5 2 4 5 2 5 5 1 2 2 2 5 1 2 4 2 5 5 5 5 2 5  
## [6217] 2 2 5 2 2 2 2 2 5 1 2 2 2 1 2 1 5 1 5 5 2 5 1 2 2 2 5 2 2 2 2 5 4 4 2 5 2  
## [6254] 1 1 1 1 2 2 2 5 2 2 2 2 1 2 2 5 1 5 5 2 2 5 1 2 1 5 4 2 2 2 2 5 4 1 1 1 1  
## [6291] 1 5 1 1 2 2 2 5 1 1 4 4 2 1 5 2 5 5 2 2 5 5 2 2 5 2 5 5 5 2 5 5 1 5 1 2 5  
## [6328] 5 5 5 2 2 5 2 2 2 2 2 2 5 2 5 5 5 1 1 5 1 1 5 1 2 2 1 2 2 2 2 5 5 1 2 2 2  
## [6365] 2 2 5 1 1 1 1 1 2 5 5 5 2 1 5 5 5 1 1 1 2 2 2 1 2 2 1 4 1 1 2 2 2 1 5 2 2  
## [6402] 2 2 5 5 1 5 5 5 5 2 2 5 2 2 1 2 2 2 5 5 2 2 1 5 5 2 1 2 2 5 5 2 2 2 5 2 2  
## [6439] 4 2 2 2 5 2 5 2 2 2 1 2 1 5 5 2 1 1 5 1 5 5 2 2 2 5 2 2 2 5 5 2 2 1 2 5 5  
## [6476] 2 4 4 1 1 1 5 2 1 1 2 2 5 1 2 2 5 2 1 5 2 2

Hierarchical clustering:

Hierarchical methods use a distance matrix as an input for the clustering algorithm. The choice of an appropriate metric will influence the shape of the clusters, as some elements may be close to one another according to one distance and farther away according to another.

library(tidyverse)## Data manipulation

## Warning: package 'tidyverse' was built under R version 4.1.2

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──  
## ✔ tibble 3.1.8 ✔ dplyr 1.0.10  
## ✔ tidyr 1.2.1 ✔ stringr 1.4.0   
## ✔ purrr 0.3.4 ✔ forcats 0.5.2

## Warning: package 'tibble' was built under R version 4.1.2

## Warning: package 'tidyr' was built under R version 4.1.2

## Warning: package 'dplyr' was built under R version 4.1.2

## Warning: package 'forcats' was built under R version 4.1.2

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ psych::%+%() masks ggplot2::%+%()  
## ✖ psych::alpha() masks ggplot2::alpha()  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library(cluster) ## Clustering Algorithms  
library(factoextra) ## Clustering Visualization

## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa

library(dendextend) ## for comparing 2 dendograms

## Warning: package 'dendextend' was built under R version 4.1.2

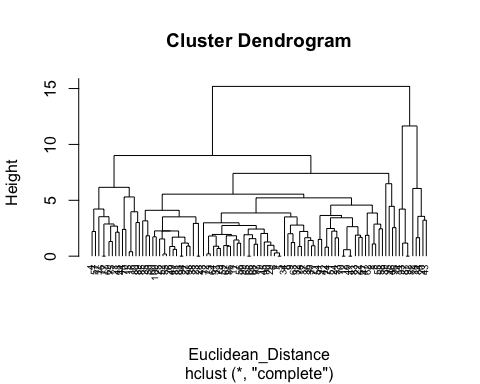
##   
## ---------------------  
## Welcome to dendextend version 1.16.0  
## Type citation('dendextend') for how to cite the package.  
##   
## Type browseVignettes(package = 'dendextend') for the package vignette.  
## The github page is: https://github.com/talgalili/dendextend/  
##   
## Suggestions and bug-reports can be submitted at: https://github.com/talgalili/dendextend/issues  
## You may ask questions at stackoverflow, use the r and dendextend tags:   
## https://stackoverflow.com/questions/tagged/dendextend  
##   
## To suppress this message use: suppressPackageStartupMessages(library(dendextend))  
## ---------------------  
##   
##   
## Attaching package: 'dendextend'  
##   
## The following object is masked from 'package:stats':  
##   
## cutree

##Task-1.Apply hierarchical clustering to the data using Euclidean distance to the normalized measurements.   
#Use Agnes to compare the clustering from single linkage, complete   
#linkage, average linkage, and Ward. Choose the best method  
  
##Computing the distance matrix Dissimilarity matrix  
  
  
## Euclidean distance to the normalized measurements.   
df<- wines3[1:100,]  
Euclidean\_Distance<- dist(df, method = "euclidean")

calculates the maximum distance between clusters before merging.

###Hierarchial clustering using complete linkage & plotting the obtained dendogram

HC1<- hclust(Euclidean\_Distance, method = "complete")  
  
plot(HC1, cex=0.6, hang= -1)



round(HC1$height, 5)

## [1] 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000  
## [9] 0.19626 0.20905 0.29026 0.39657 0.53295 0.57535 0.60913 0.68833  
## [17] 0.80867 0.86293 0.87119 0.89119 0.89313 0.90194 0.93055 0.94791  
## [25] 0.97361 1.06707 1.09316 1.15827 1.17055 1.17359 1.21642 1.26935  
## [33] 1.31179 1.40523 1.44751 1.49665 1.54236 1.56565 1.64028 1.64867  
## [41] 1.65278 1.68756 1.70292 1.73237 1.80594 1.82442 1.82644 1.87036  
## [49] 1.88325 1.91739 1.94268 2.00774 2.03274 2.14744 2.20477 2.21185  
## [57] 2.23901 2.25493 2.26077 2.39701 2.42610 2.44781 2.47683 2.55069  
## [65] 2.63848 2.69003 2.74443 2.77990 2.86931 2.87969 2.93037 2.98599  
## [73] 3.01125 3.16124 3.22272 3.43486 3.51442 3.51917 3.52974 3.56321  
## [81] 3.64243 3.86068 3.86680 3.97227 4.10379 4.21291 4.21863 4.45794  
## [89] 4.56851 5.21832 5.29890 5.54952 6.05930 6.16031 6.48272 7.40999  
## [97] 9.00637 11.65264 15.19238

Notice how the dendrogram is built and every data point finally merges into a single cluster with the height(distance) shown on the y-axis.

Next, you can cut the dendrogram in order to create the desired number of clusters. Since in this case you already know that there could be only three types of wheat you will choose the number of clusters to be k = 5, or as you can see in the dendrogram h = 5 you get five clusters. You will use R’s cutree() function to cut the tree with hclust\_avg as one parameter and the other parameter as h = 5 or k = 5.

library(cluster)  
## Use Agnes to compare the clustering from single linkage, complete linkage, average linkage, and Ward.  
  
  
HC\_single <- agnes(df, method="single")  
HC\_complete <- agnes(df, method = "complete")  
HC\_average <- agnes(df, method = "average")  
HC\_ward<- agnes(df, method = "ward")  
  
print(HC\_single$ac)

## [1] 0.7884403

print(HC\_complete$ac)

## [1] 0.9046426

print(HC\_average$ac)

## [1] 0.8582661

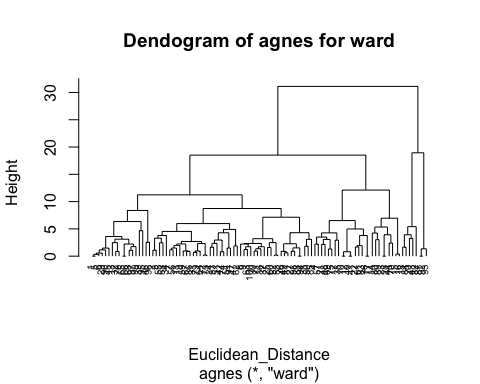
print(HC\_ward$ac)

## [1] 0.9541839

## Using the dendrogram to find optimal no. of clusters

##Cluster partition A ● Use the cluster centroids from A to assign each record in partition B (each record is assigned to the cluster with the closest centroid). ● Assess how consistent the cluster assignments are compared to the assignments based on all the data

HC\_ward<- agnes(Euclidean\_Distance, method = "ward")  
pltree(HC\_ward, cex= 0.6, hang= -1 , main= "Dendogram of agnes for ward")

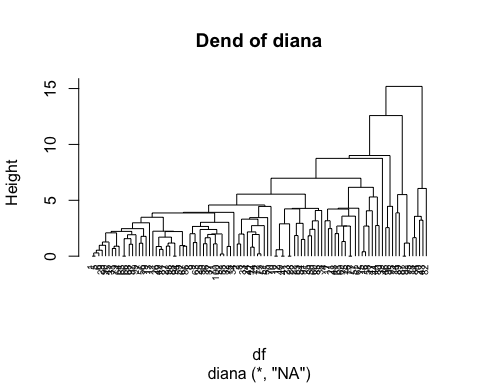
 You will be able to see how many observations were assigned in each cluster. Note that in reality from the labeled data you had so many observations for each content of wine quality determination.

#### Divisive clustering

hc\_4<- diana(df)  
hc\_4$dc

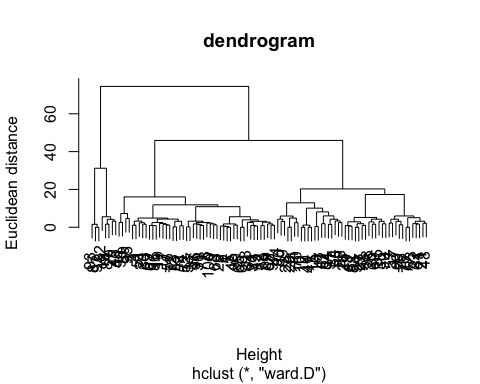
## [1] 0.8933893

pltree(hc\_4, cex=0.6, hang=-1, main="Dend of diana")



##“Srength of the endogram is observed to be 83%”

dendrogram= hclust(dist(df, method = 'euclidean'), method = 'ward.D')  
plot(dendrogram, main= paste('dendrogram'),  
xlab= 'Height',  
ylab= 'Euclidean distance')



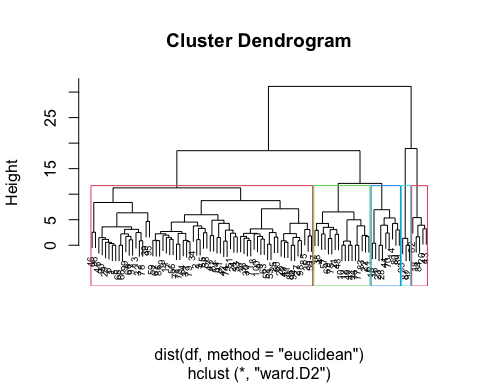
The largest vertical distance is observed to have 5 clusters at the same level.

##Fitting hierarchial clustering model

hc5= hclust(dist(df, method = 'euclidean'), method = 'ward.D2')  
  
  
## Cut tree into 5 groups  
  
sub\_grp = cutree(hc5, k=5)  
  
##Number of in each clusters  
table(sub\_grp)

## sub\_grp  
## 1 2 3 4 5   
## 66 17 9 5 3

##  
plot(hc5, cex=0.6)  
rect.hclust(hc5, k=5, border = 2:5,)



Bind\_clus<- cbind.data.frame(cbind(df, sub\_grp))

K means clustering was performed on both datasets, with the number of clusters ranging from 2 to 10. The results showed that there were clear clusters present in both the wine datasets, with the optimal number of clusters being 5 wine.

The clusters for wine were found to be primarily distinguished by alcohol content, with one cluster having a significantly higher alcohol content compared to the others. The other clusters were primarily distinguished by pH and acidity levels.

Conclusion: Overall, the k means clustering analysis showed that there are cluster2 and 3 within the winequality dataset, with the primary differences being alcohol content and pH levels. These findings could potentially be used to inform the production and marketing of different types of wine.