Vinhos

Matheus, Murilo

pacotes

```
source("func.R")
# packages
packages <- c("Rcpp",
              "tidyverse",
              "magrittr",
              "tidyverse",
              "broom",
              "corrplot",
              "ca",
              "RColorBrewer",
              "gridExtra",
              "forcats",
              "rpart",
              "rpart.plot",
              "pROC",
              "randomForest",
              "caret",
              "xtable")

ipak(packages)
```

```
## Loading required package: Rcpp
```

```
## Loading required package: tidyverse
```

```
## Loading tidyverse: ggplot2
## Loading tidyverse: tibble
## Loading tidyverse: tidyr
## Loading tidyverse: readr
## Loading tidyverse: purrr
## Loading tidyverse: dplyr
```

```
## Conflicts with tidy packages ----------------------------------------------
```

```
## filter(): dplyr, stats
## lag():    dplyr, stats
```

```
## Loading required package: magrittr
```

```
##
## Attaching package: 'magrittr'
```

```
## The following object is masked from 'package:purrr':
##
##     set_names
```

```
## The following object is masked from 'package:tidyr':
##
##     extract
```

```
## Loading required package: broom
```

```
## Loading required package: corrplot

## Loading required package: ca

## Loading required package: RColorBrewer

## Loading required package: gridExtra

##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine

## Loading required package: forcats

## Loading required package: rpart

## Loading required package: rpart.plot

## Loading required package: pROC

## Type 'citation("pROC")' for a citation.

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var

## Loading required package: randomForest

## randomForest 4.6-12

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:gridExtra':
##
##     combine

## The following object is masked from 'package:dplyr':
##
##     combine

## The following object is masked from 'package:ggplot2':
##
##     margin

## Loading required package: caret

## Loading required package: lattice

##
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':
##
##     lift

## Loading required package: xtable
```

```
##         Rcpp    tidyverse     magrittr    tidyverse        broom
##         TRUE         TRUE         TRUE         TRUE         TRUE
##     corrplot           ca RColorBrewer    gridExtra      forcats
##         TRUE         TRUE         TRUE         TRUE         TRUE
##        rpart   rpart.plot         pROC randomForest        caret
##         TRUE         TRUE         TRUE         TRUE         TRUE
##       xtable
##         TRUE
```

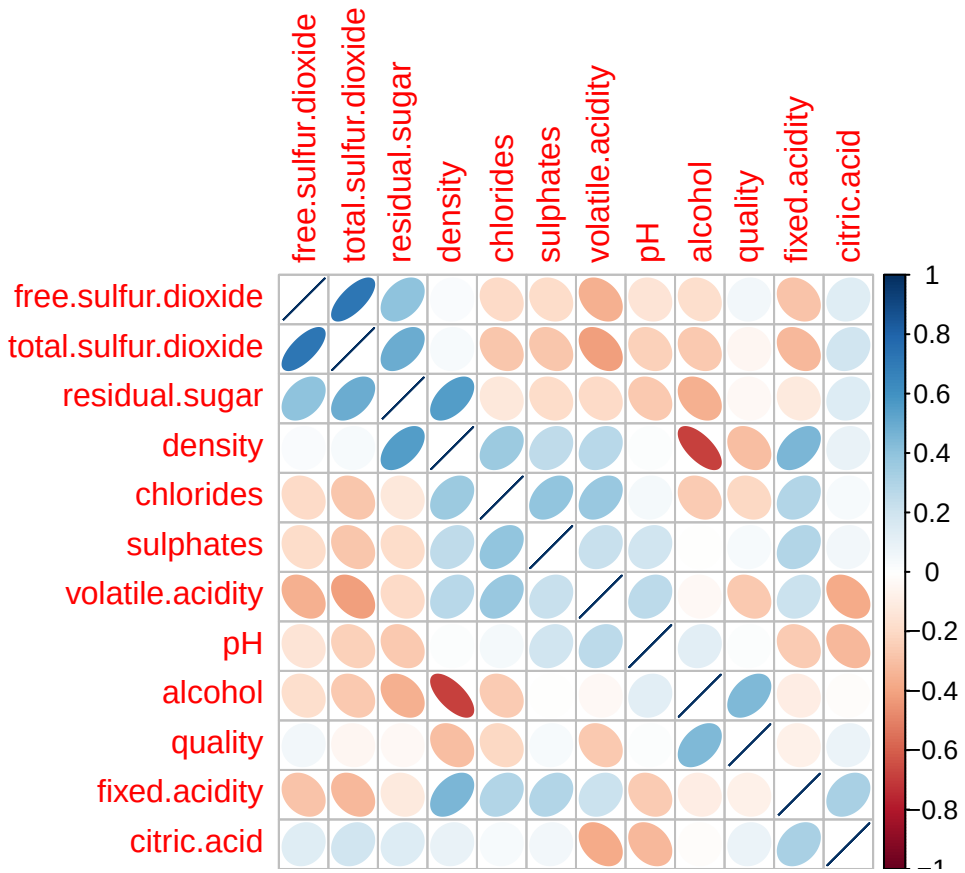
```
#library(devtools)
#install_github("vqv/ggbiplot")
```

Carregando os dados

```
##leitura dos dados
red <- read.csv("winequality/winequality-red.csv",header = T,sep = ";") %>%
  mutate(vinho = as.factor("red"))

white <- read.csv("winequality/winequality-white.csv",header = T,sep = ";") %>% mutate(vinho =as.factor
dados <- rbind(white,red)
```

Descritivas

```
dados %<>% mutate( quality = as.numeric(quality),
                   qualidade = lvls_reorder(map_chr(quality,cl),c(2,3,1)))

corrplot::corrplot(cor(dados[1:12]),method="ellipse",order = "hclust", addrect=NULL)
```

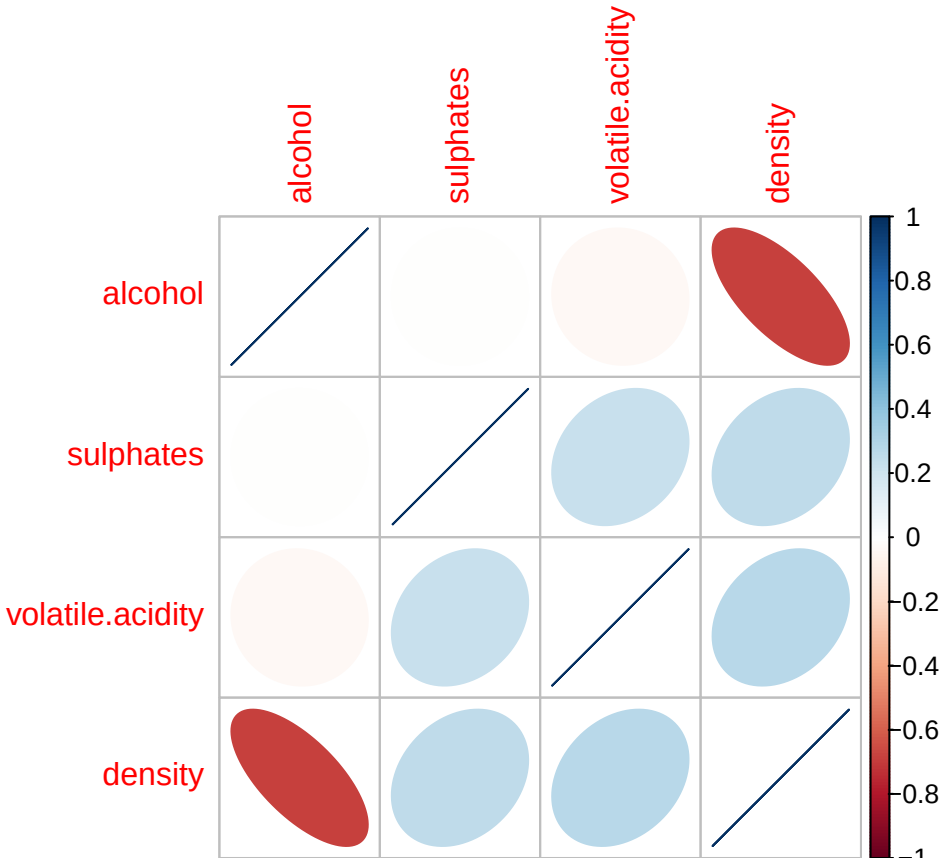



```
cbind(
cor(x=red[,1:11], y=red$quality),
cor(x=white[,1:11], y=white$quality))
```

```
                         [,1]          [,2]
```

fixed.acidity 0.12405165 -0.113662831 volatile.acidity -0.39055778 -0.194722969 citric.acid 0.22637251 -0.009209091 residual.sugar 0.01373164 -0.097576829 chlorides -0.12890656 -0.209934411 free.sulfur.dioxide -0.05065606 0.008158067 total.sulfur.dioxide -0.18510029 -0.174737218 density -0.17491923 -0.307123313 pH -0.05773139 0.099427246 sulphates 0.25139708 0.053677877 alcohol 0.47616632 0.435574715

```
indep <- c("vinho","volatile.acidity","sulphates","alcohol",
           "density","chlorides")
data <- dados %>% select(indep,quality,qualidade)
corrplot::corrplot(cor(data[2:5]),method="ellipse",order = "hclust", addrect=NULL)
```




```
set.seed(1)
sample_rows <- sample(1:nrow(data), nrow(data)*.75)
train <- data[sample_rows,]
test <- data[-sample_rows,]
glimpse(dados)
```

Observations: 6,497 Variables: 14 $ fixed.acidity 7.0, 6.3, 8.1, 7.2, 7.2, 8.1, 6.2, 7.0, 6... $ volatile.acidity 0.27, 0.30, 0.28, 0.23, 0.23, 0.28, 0.32,... $ citric.acid 0.36, 0.34, 0.40, 0.32, 0.32, 0.40, 0.16,... $ residual.sugar 20.70, 1.60, 6.90, 8.50, 8.50, 6.90, 7.00... $ chlorides 0.045, 0.049, 0.050, 0.058, 0.058, 0.050,... $ free.sulfur.dioxide 45, 14, 30, 47, 47, 30, 30, 45, 14, 28, 1... $ total.sulfur.dioxide 170, 132, 97, 186, 186, 97, 136, 170, 132... $ density 1.0010, 0.9940, 0.9951, 0.9956, 0.9956, 0... $ pH 3.00, 3.30, 3.26, 3.19, 3.19, 3.26, 3.18,... $ sulphates 0.45, 0.49, 0.44, 0.40, 0.40, 0.44, 0.47,... $ alcohol 8.8, 9.5, 10.1, 9.9, 9.9, 10.1, 9.6, 8.8,... $ quality 6, 6, 6, 6, 6, 6, 6, 6, 6, 5, 5, 5, 7,... $ vinho white, white, white, white, white, white... $ qualidade Medio, Medio, Medio, Medio, Medio, Medio...

```
dados %<>% mutate( quality = as.numeric(quality),
                   qualidade = lvls_reorder(map_chr(quality,cl),c(2,3,1)))
options(xtable.comment = FALSE)
descritivas <-
dados %>% select_if(.predicate = is.numeric) %>%
  gather(coluna,valores) %>%
  group_by(Coluna = coluna) %>%
  summarise_if(.predicate = function(x) is.numeric(x),
                   .funs = c(Media = "mean",
                      DP = "sd",
                      Var. = "var",
                      Minimo = "min",
```

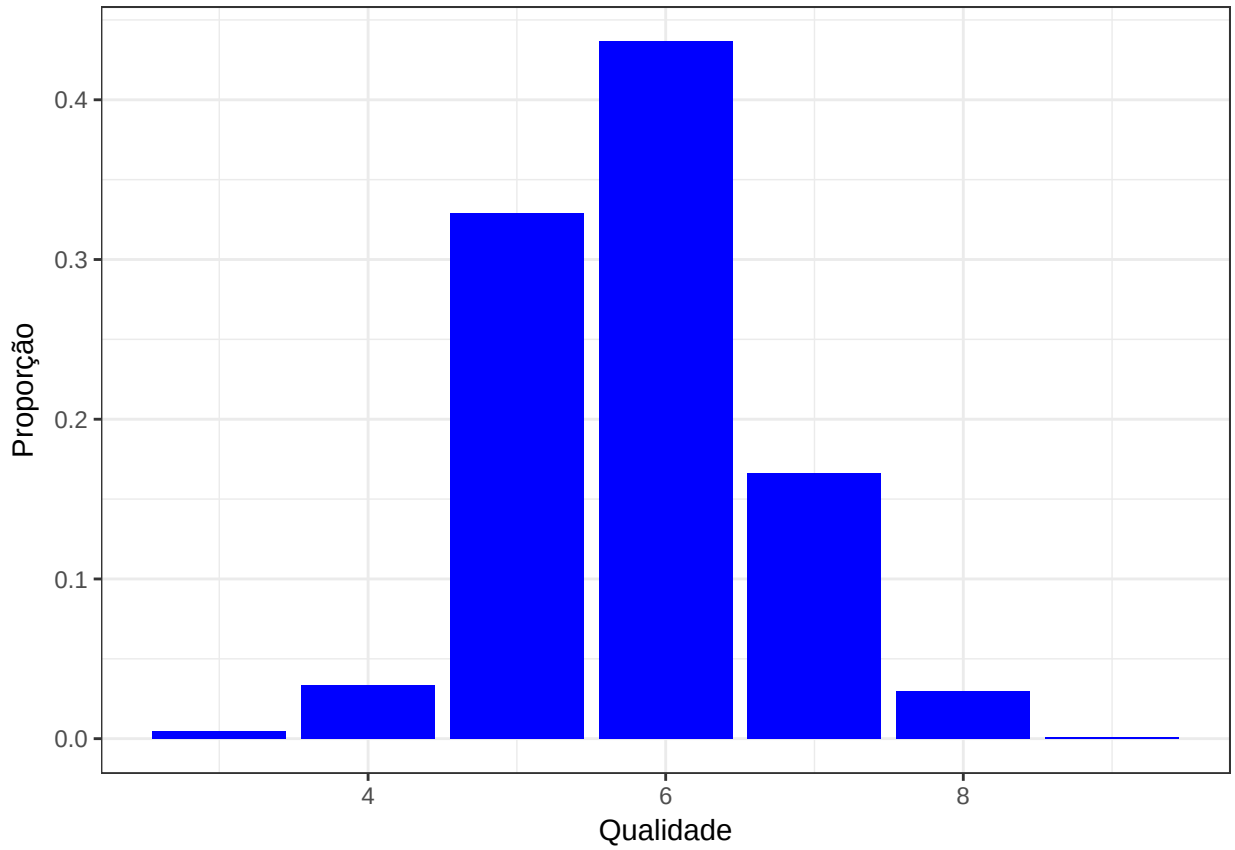
```
                      CV =  "cv",
                      Mediana = "median",
                      Maximo = "max")) %>%
  mutate_if(.predicate = is.numeric,funs(round(.,3)))
xtable(descritivas,caption = "Medidas Descritivas",digits = 3)
```

	Coluna	Media	DP	Var.	Minimo	CV	Mediana	Maximo
1	alcohol	10.492	1.193	1.423	8.000	11.368	10.300	14.900
2	chlorides	0.056	0.035	0.001	0.009	62.522	0.047	0.611
3	citric.acid	0.319	0.145	0.021	0.000	45.607	0.310	1.660
4	density	0.995	0.003	0.000	0.987	0.301	0.995	1.039
5	fixed.acidity	7.215	1.296	1.681	3.800	17.968	7.000	15.900
6	free.sulfur.dioxide	30.525	17.749	315.041	1.000	58.146	29.000	289.000
7	pH	3.219	0.161	0.026	2.720	4.996	3.210	4.010
8	quality	5.818	0.873	0.763	3.000	15.009	6.000	9.000
9	residual.sugar	5.443	4.758	22.637	0.600	87.408	3.000	65.800
10	sulphates	0.531	0.149	0.022	0.220	28.010	0.510	2.000
11	total.sulfur.dioxide	115.745	56.522	3194.720	6.000	48.833	118.000	440.000
12	volatile.acidity	0.340	0.165	0.027	0.080	48.470	0.290	1.580

Table 1: Medidas Descritivas

variavel resposta

```
ggplot(dados,aes(quality),fill = "blue") + stat_count(fill = "blue",aes(y = (..count..)/sum(..count..)))
  theme_bw() +
  labs(x = "Qualidade",y="Proporção")
```



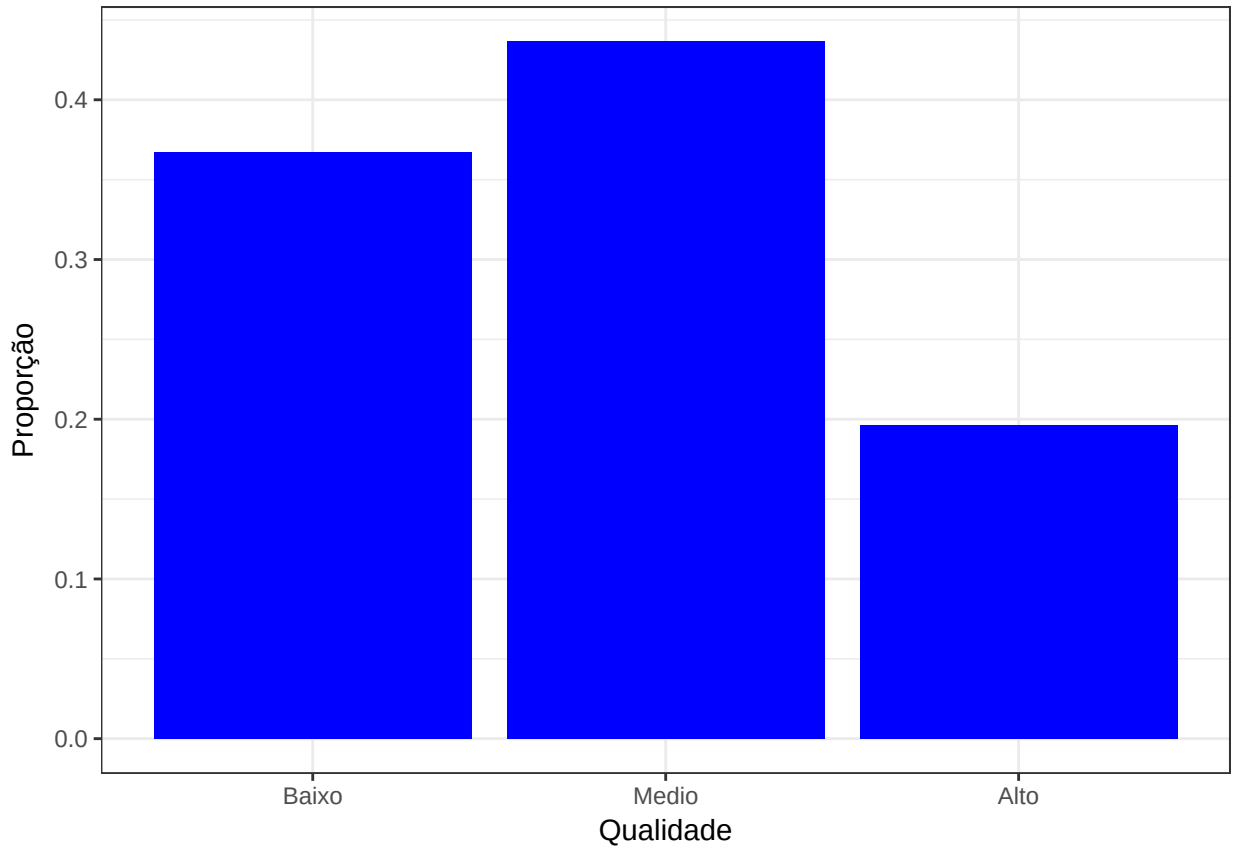

```
table(dados$quality)
```

```
##
##    3    4    5    6    7    8    9
##   30  216 2138 2836 1079  193    5
```

```
prop.table(table(dados$quality))
```

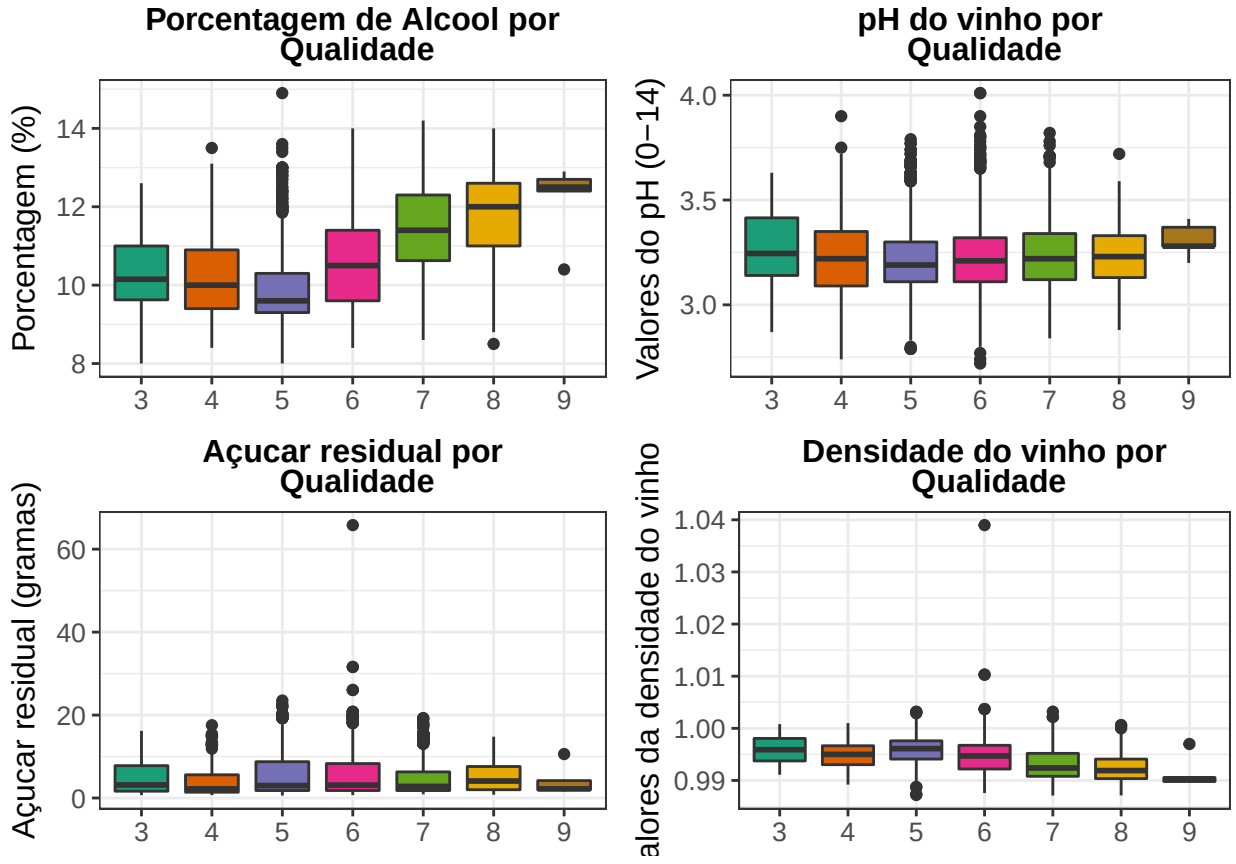
```
##
##           3           4           5           6           7           8
## 0.004617516 0.033246114 0.329074958 0.436509158 0.166076651 0.029706018
##           9
## 0.000769586
```

```
ggplot(dados,aes(qualidade),fill = "blue") + stat_count(fill = "blue",aes(y = (..count..)/sum(..count..)
  theme_bw() +
  labs(x = "Qualidade",y = "Proporção")
```




```r
#ANALISANDO POSSIVEIS VARIAVEIS INFLUENTES NA ANALISE NA QUALIDADE DO VINHO
#Começarei analisando o alcool, açucar, ph
myCol <- brewer.pal(8, "Dark2")
dados %<>%  mutate(quality = factor(quality))
p1 <- dados %>% ggplot(aes(quality,alcohol,fill = quality,group = quality)) +
  geom_boxplot(show.legend = F) +
  theme_bw() +
  labs(y="Porcentagem (%)", title = "Porcentagem de Alcool por\n Qualidade",x=NULL) +
  theme(plot.title = element_text(size = 12,lineheight=.8,
  face="bold",hjust = 0.5),text=element_text(size=12,  family="sans")) +
  scale_fill_manual(values = myCol)


p2 <- dados %>% ggplot(aes(quality,pH,fill = quality,group = quality)) +
  geom_boxplot(show.legend = F) +
  theme_bw() +
  labs(y="Valores do pH (0-14)", title = "pH do vinho por\n Qualidade",x=NULL) +
  theme(plot.title = element_text(size = 12,lineheight=.8,
  face="bold",hjust = 0.5),text=element_text(size=12,  family="sans")) +
  scale_fill_manual(values = myCol)


p3 <- dados %>% ggplot(aes(quality,residual.sugar,fill = quality,group = quality)) +
  geom_boxplot(show.legend = F) +
  theme_bw() +
  labs(y="Açucar residual (gramas)", title = "Açucar residual por\n Qualidade",x=NULL) +
```

```
  theme(plot.title = element_text(size = 12,lineheight=.8,
  face="bold",hjust = 0.5),text=element_text(size=12,  family="sans")) +
  scale_fill_manual(values = myCol)


p4 <- dados %>% ggplot(aes(quality,density,fill = quality,group = quality)) +
  geom_boxplot(show.legend = F) +
  theme_bw() +
  labs(y="Valores da densidade do vinho", title = "Densidade do vinho por\n Qualidade",x=NULL) +
  theme(plot.title = element_text(size = 12,lineheight=.8,
  face="bold",hjust = 0.5),text=element_text(size=12,  family="sans")) +
  scale_fill_manual(values = myCol)

grid.arrange(p1,p2,p3,p4)
```



Florestas

```
dados %<>% mutate(quality = as.numeric(quality))

##floresta Regressao
rf_model <- randomForest(quality ~. - qualidade,data = dados,  importance=TRUE ,ntree = 100)


predito <- round(predict(rf_model))
```

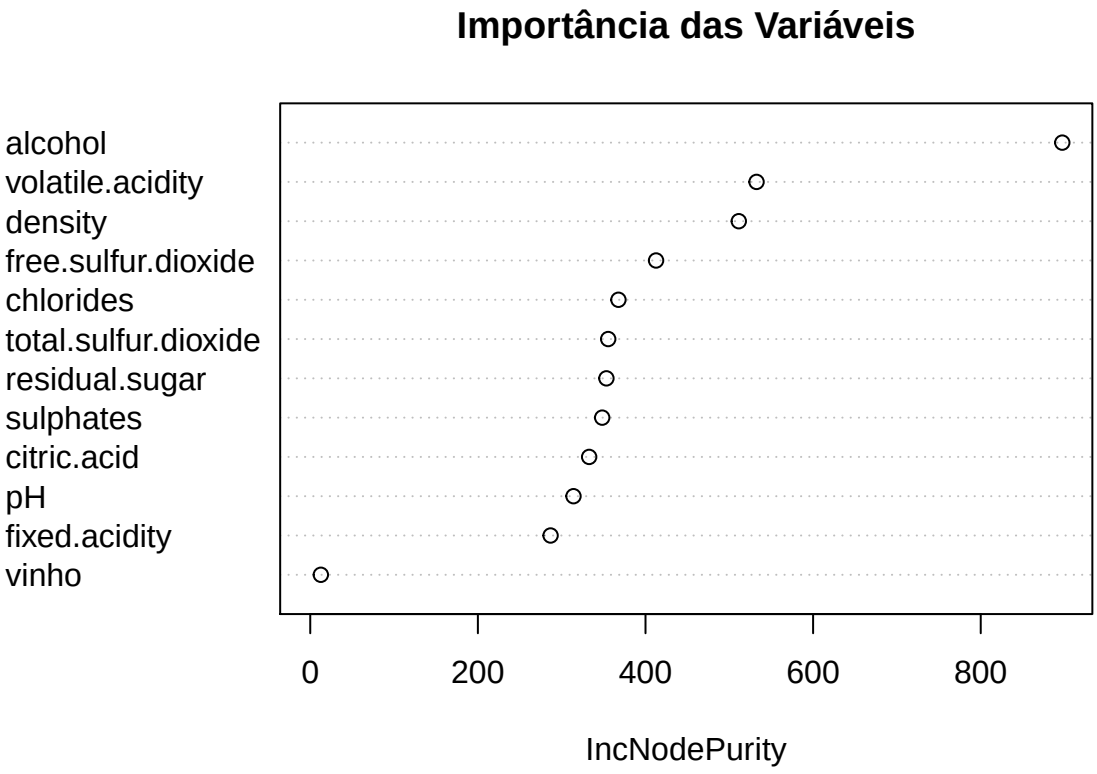
```r
table(predicao=predito, observado=dados$quality)
```

```
##          observado
## predicao    1    2    3    4    5    6    7
##        2    1   10    3    0    0    0    0
##        3   19  149 1556  385   14    2    0
##        4    9   57  561 2268  442   50    1
##        5    1    0   18  183  623  102    4
##        6    0    0    0    0    0   39    0
```

```r
mean(predito == dados$quality)
```

```
## [1] 0.6920117
```

```r
varImpPlot(rf_model,main = "Importância das Variáveis",type = 2)
```

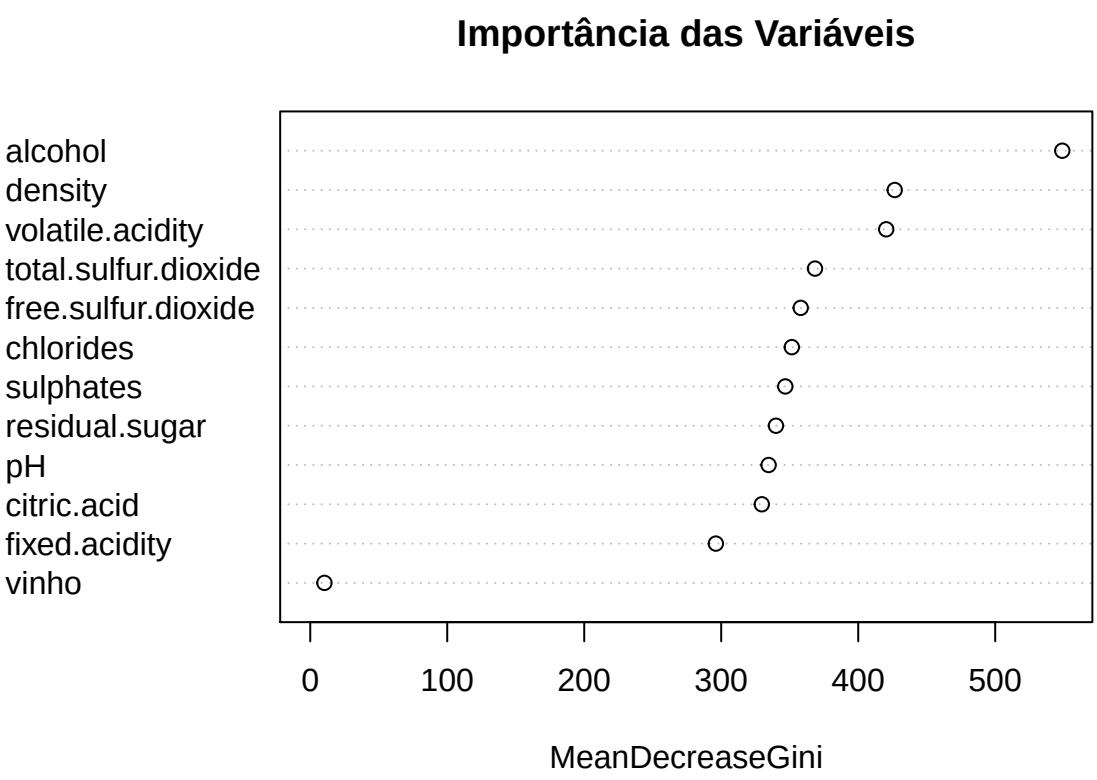



```r
##floresta 2.0 Classificação

rf_model <- randomForest(qualidade ~. - quality,data = dados,  importance = TRUE ,ntree = 100)


predito <- predict(rf_model)
confusionMatrix(predito,dados$qualidade)
```

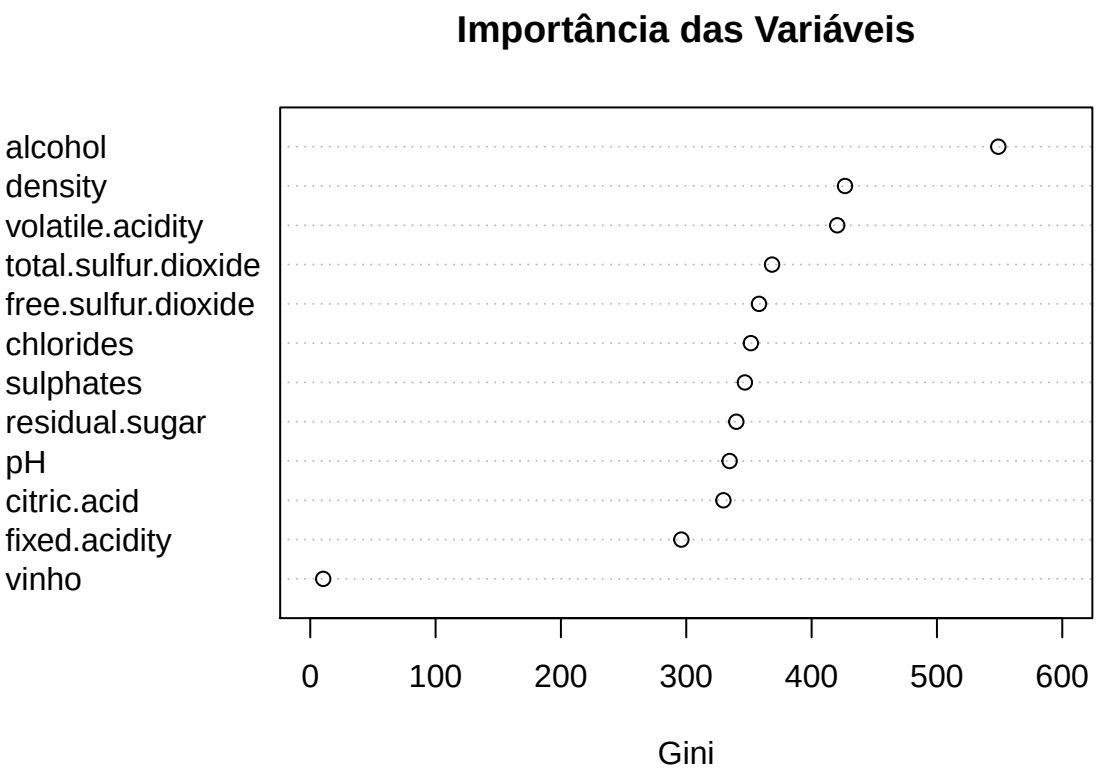
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction Baixo Medio Alto
```

```
##      Baixo  1854   466   37
##      Medio   505  2139  416
##      Alto     25   231  824
##
## Overall Statistics
##
##                Accuracy : 0.7414
##                  95% CI : (0.7306, 0.752)
##     No Information Rate : 0.4365
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5887
##  Mcnemar's Test P-Value : 2.854e-12
##
## Statistics by Class:
##
##                      Class: Baixo Class: Medio Class: Alto
## Sensitivity                0.7777       0.7542      0.6453
## Specificity                0.8777       0.7484      0.9510
## Pos Pred Value             0.7866       0.6990      0.7630
## Neg Pred Value             0.8720       0.7972      0.9164
## Prevalence                 0.3669       0.4365      0.1966
## Detection Rate             0.2854       0.3292      0.1268
## Detection Prevalence       0.3628       0.4710      0.1662
## Balanced Accuracy          0.8277       0.7513      0.7981
```

```r
varImpPlot(rf_model,main = "Importância das Variáveis",type = 2)
```




```r
impToPlot <- importance(rf_model, scale=FALSE)
dotchart(sort(impToPlot[,5]), xlim=c(0,600), xlab="Gini",main ="Importância das Variáveis" )
```

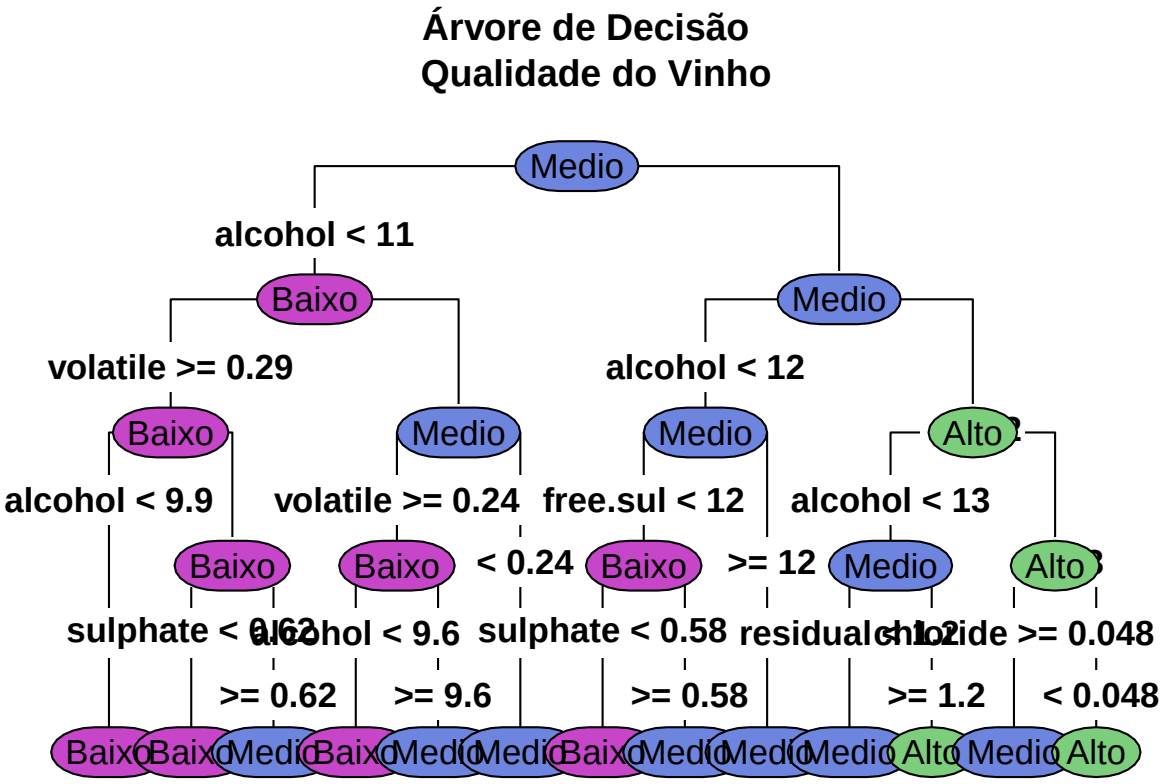



```
##Arvore Classificação
set.seed(1)
tree <- rpart(qualidade ~. - quality,data = dados, control = rpart.control(cp = 0,maxdepth = 4))

predito <- predict(tree,type = "class")
confusionMatrix(predito,dados$qualidade)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction Baixo Medio Alto
##      Baixo  1562   714   56
##      Medio   763  1697  682
##      Alto     59   425  539
##
## Overall Statistics
##
##                Accuracy : 0.5846
##                  95% CI : (0.5725, 0.5966)
##     No Information Rate : 0.4365
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.3366
##  Mcnemar's Test P-Value : 2.998e-13
##
## Statistics by Class:
```

```
##
##                      Class: Baixo Class: Medio Class: Alto
## Sensitivity                0.6552       0.5984     0.42208
## Specificity                0.8128       0.6053     0.90728
## Pos Pred Value             0.6698       0.5401     0.52688
## Neg Pred Value             0.8026       0.6605     0.86518
## Prevalence                 0.3669       0.4365     0.19655
## Detection Rate             0.2404       0.2612     0.08296
## Detection Prevalence       0.3589       0.4836     0.15746
## Balanced Accuracy          0.7340       0.6018     0.66468
```

```
prp(tree,type = 4,box.col=c("#C745C9","#6D85DE", "palegreen3")[tree$frame$yval],tweak = 1.5,fallen.leave
```




```
#USADO PCA
wine <- dados[,1:11]
# wine.pca <- prcomp(wine, scale. = TRUE)
# ggbiplot(wine.pca, obs.scale = 1, var.scale = 1,
#          groups = dados[,13], ellipse = TRUE, circle = TRUE) +
#   scale_color_discrete(name = '') +
#   theme(legend.direction = 'horizontal', legend.position = 'top') + theme_bw()


#A PARTIR DAQUI, DEVIDO A SEPARACAO DOS GRUPOS, RESOLVI ANALISAR SEPARADAMENTE
```

FLORESTA ALEATORIA VINHO BRANCO E VERMELHO

```
##
## Call:
##  randomForest(formula = rating ~ . - vinho - quality, data = dados)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 3
##
##         OOB estimate of  error rate: 1.91%
## Confusion matrix:
##       Ruim Bom Medio class.error
## Ruim  6291   0     8 0.001270043
## Bom      5   0     0 1.000000000
## Medio  111   0    82 0.575129534
```

```
##                       MeanDecreaseGini
## fixed.acidity                  26.71902
## volatile.acidity               30.57655
## citric.acid                    26.47440
## residual.sugar                 32.67645
## chlorides                      29.77715
## free.sulfur.dioxide            33.96036
## total.sulfur.dioxide           30.80699
## density                        34.16127
## pH                             29.35041
## sulphates                      32.61749
## alcohol                        35.61938
## qualidade                      41.25627
```

```
##        pred_RF
##         Ruim  Bom Medio
##   Ruim  6291    0     8
##   Bom      5    0     0
##   Medio  111    0    82
```

```
##
## Call:
##  randomForest(formula = rating ~ . - vinho - quality, data = dados)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 3
##
##         OOB estimate of  error rate: 1.91%
## Confusion matrix:
##       Ruim Bom Medio class.error
## Ruim  6291   0     8 0.001270043
## Bom      5   0     0 1.000000000
## Medio  111   0    82 0.575129534
```

```
##                       MeanDecreaseGini
## fixed.acidity                  69.89057
## volatile.acidity              103.60913
## citric.acid                    70.30478
## residual.sugar                 64.87859
## chlorides                      77.81598
```

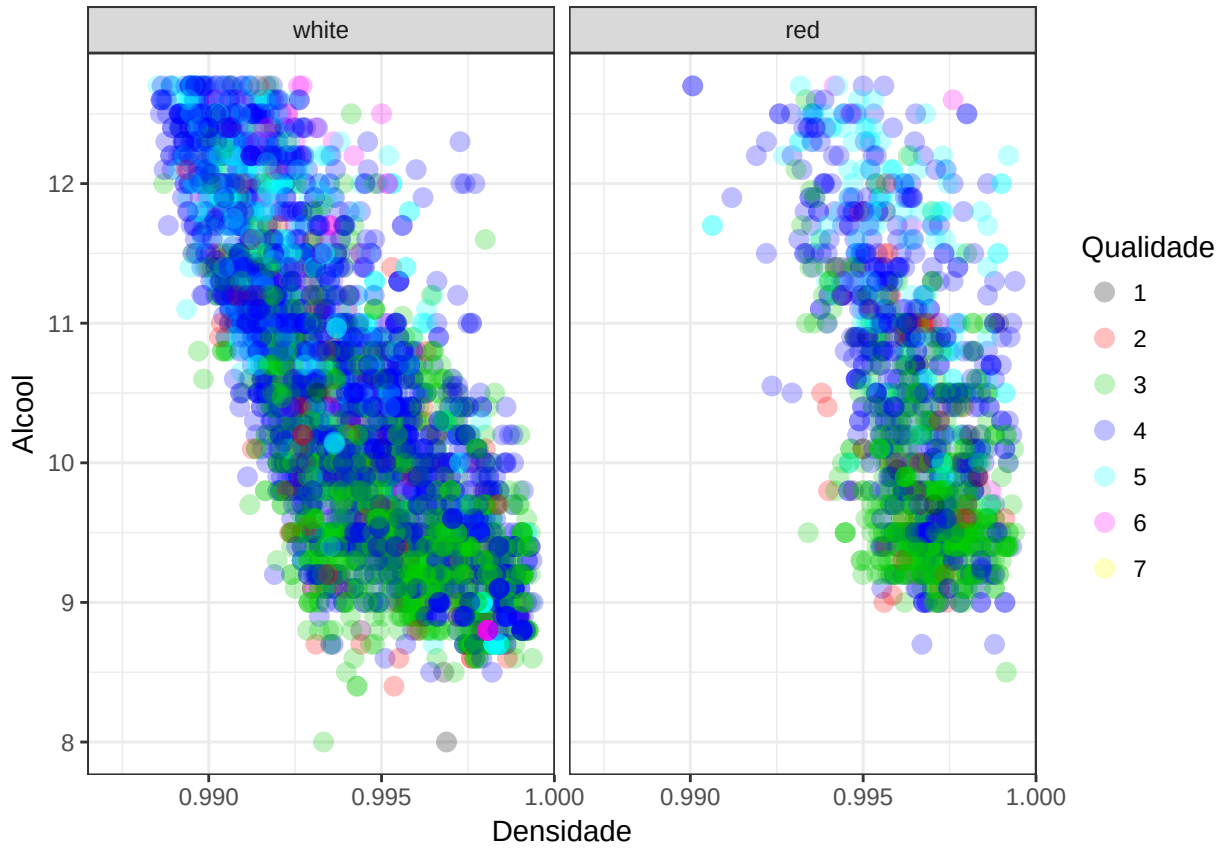
```
## free.sulfur.dioxide          63.24579
## total.sulfur.dioxide         97.24974
## density                      89.67449
## pH                           69.42025
## sulphates                   115.33664
## alcohol                     146.56616
```

```
##
## Call:
##  randomForest(formula = rating ~ . - vinho - quality, data = white)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 3
##
##         OOB estimate of  error rate: 25.17%
## Confusion matrix:
##       Ruim Bom Medio class.error
## Ruim  1227  16   397   0.2518293
## Bom     20 703   337   0.3367925
## Medio  294 169  1735   0.2106460
```

```
##                      MeanDecreaseGini
## fixed.acidity                225.6520
## volatile.acidity             314.6381
## citric.acid                  248.0415
## residual.sugar               275.1482
## chlorides                    268.2315
## free.sulfur.dioxide          295.3304
## total.sulfur.dioxide         278.8297
## density                      340.2162
## pH                           259.9758
## sulphates                    235.5267
## alcohol                      390.1216
```

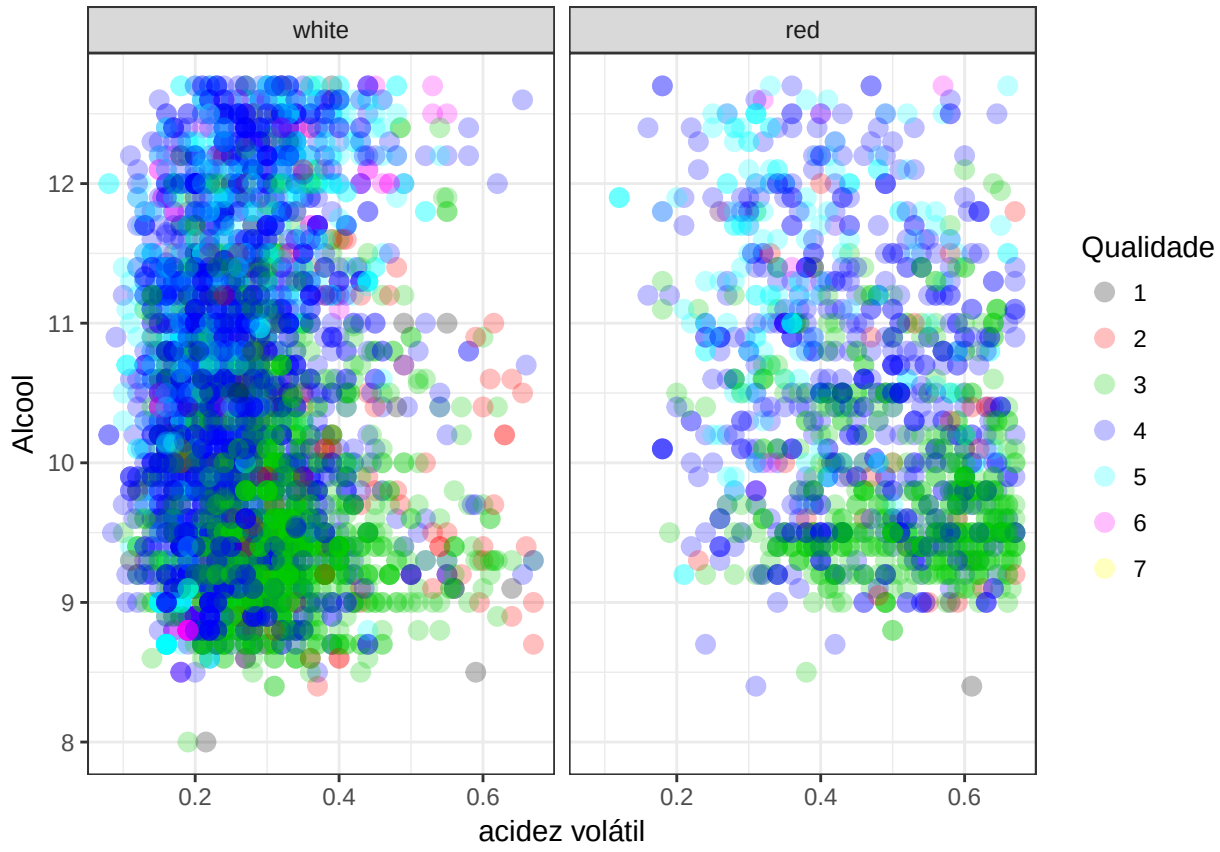
```
#ALCOOL VS DENSIDADE
dados$quality <- as.factor(dados$quality)

dados %>% ggplot(aes(x = density, y = alcohol, color = quality)) +
    facet_wrap(~vinho) +
    geom_point(size = 3, alpha = 1/4) +
    scale_color_identity(guide = 'legend') +
    ylim(min(dados$alcohol), quantile(dados$alcohol, 0.95)) +
    xlim(min(dados$density), quantile(dados$density, 0.95)) +
    theme_bw() +
    labs(y = "Alcool",x = "Densidade",color = "Qualidade")
```

```
## Warning: Removed 614 rows containing missing values (geom_point).
```



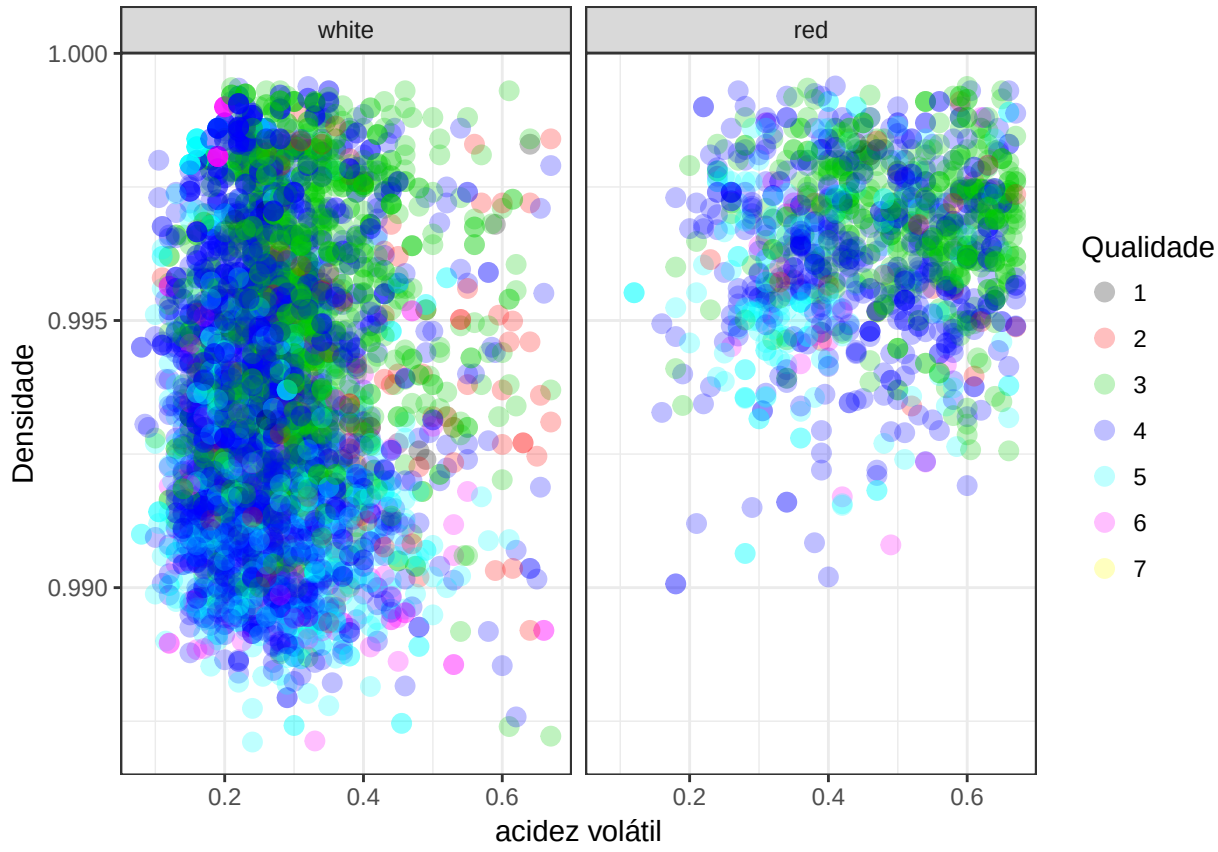

```
#ALCOOL VS ACIDEZ
dados$quality <- as.factor(dados$quality)

dados %>% ggplot(aes(x = volatile.acidity, y = alcohol, color = quality)) +
    facet_wrap(~vinho) +
    geom_point(size = 3, alpha = 1/4) +
    scale_color_identity(guide = 'legend') +
    ylim(min(dados$alcohol), quantile(dados$alcohol, 0.95)) +
    xlim(min(dados$volatile.acidity), quantile(dados$volatile.acidity, 0.95)) +
  theme_bw() +
     labs(y = "Alcool",x = "acidez volátil",color = "Qualidade")
```

```
## Warning: Removed 599 rows containing missing values (geom_point).
```



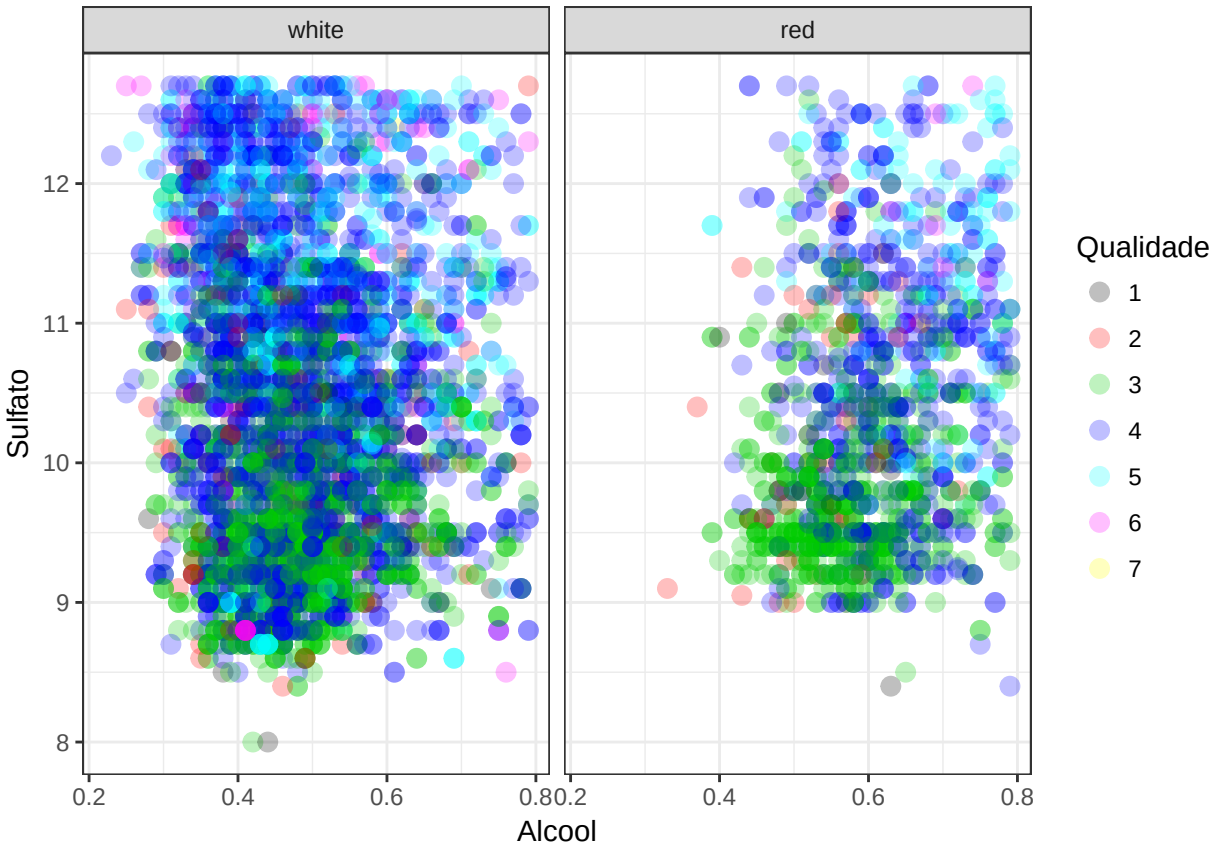

```
#volatile.acidity VS density
dados$quality <- as.factor(dados$quality)

dados %>% ggplot(aes(x = volatile.acidity, y = density, color = quality)) +
    facet_wrap(~vinho) +
    geom_point(size = 3, alpha = 1/4) +
    scale_color_identity(guide = 'legend') +
    ylim(min(dados$density), quantile(dados$density, 0.95)) +
    xlim(min(dados$volatile.acidity), quantile(dados$volatile.acidity, 0.95))  + theme_bw() +
   labs(y = "Densidade",x = "acidez volátil",color = "Qualidade")
```

```
## Warning: Removed 620 rows containing missing values (geom_point).
```




```
#SULFATO VS ALCOOL
dados$quality <- as.factor(dados$quality)

dados %>% ggplot(aes(x = sulphates, y = alcohol, color = quality)) +
    facet_wrap(~vinho) +
    geom_point(size = 3, alpha = 1/4) +
    scale_color_identity(guide = 'legend') +
    ylim(min(dados$alcohol), quantile(dados$alcohol, 0.95)) +
    xlim(min(dados$sulphates), quantile(dados$sulphates, 0.95)) +theme_bw() +
  labs(y = "Sulfato",x = "Alcool",color = "Qualidade")
```

```
## Warning: Removed 589 rows containing missing values (geom_point).
```


white
red
Sulfato
12
11
10
9
8
0.2
0.4
0.6
0.8
Alcool
Qualidade
1
2
3
4
5
6
7