Lista 4 - Mineração

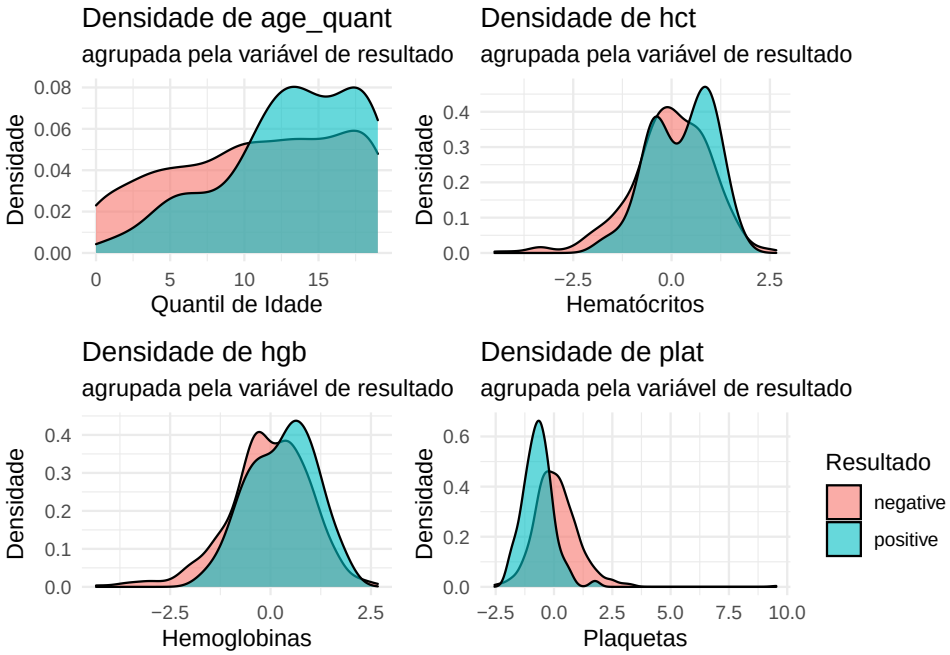
Victor Alves Dogo Martins, RA: 744878 Ana Beatriz Alves Monteiro, RA: 727838
Larissa Torres, RA: 631914

11-09-2022

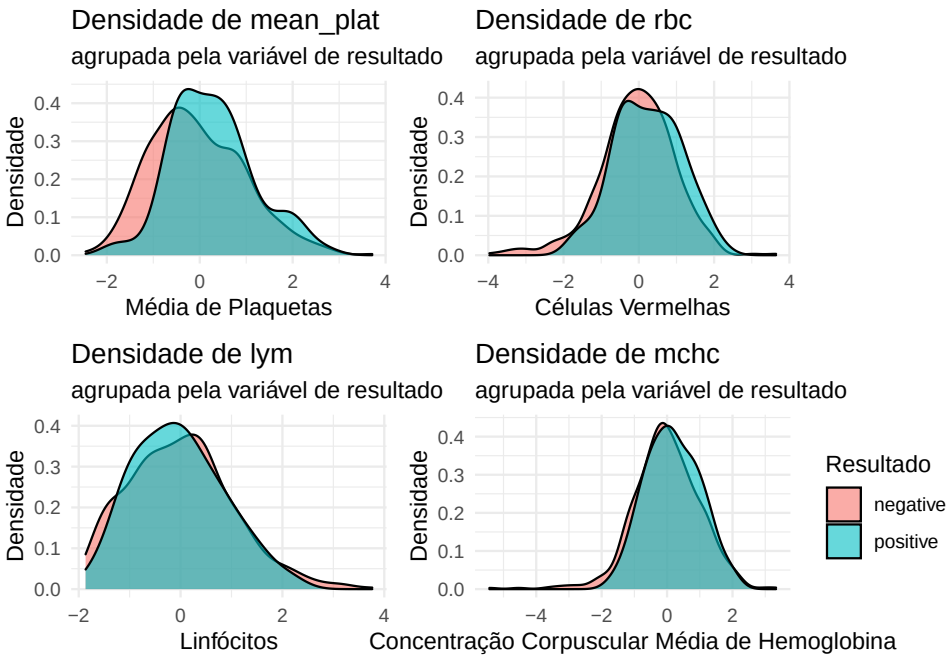
Item A

```
### Carregando pacotes

library(tidyverse)
library(knitr)
library(kableExtra)
library(patchwork)
library(rsample)
library(glmnet)
library(caret)
library(rpart)
library(rpart.plot)
library(randomForest)

### Lendo dados

df <- readr::read_csv('dados_covid.csv') |>
  rename(result=1, age_quant=2, hct=3, hgb=4,
         plat=5, mean_plat=6, rbc=7, lym=8,
         mchc=9, wbc=10, baso=11, mch=12,
         eos=13, mcv=14, mono=15, rdw=16)

### ITEM A: estimações de densidade continua das
### variaveis divididas por diagnostico

age_quant <- df |>
  ggplot()+
  aes(x=age_quant, fill=result)+
  geom_density(alpha=0.6)+
  theme_minimal()+
  theme(legend.position = 'none')+
  labs(x='Quantil de Idade', y='Densidade',
       title='Densidade de age_quant',
       subtitle = 'agrupada pela variável de resultado')

hct <- df |>
  ggplot()+
  aes(x=hct, fill=result)+
```

```
  geom_density(alpha=0.6)+
  theme_minimal()+
  theme(legend.position = 'none')+
  labs(x='Hematócritos', y='Densidade',
       title='Densidade de hct',
       subtitle = 'agrupada pela variável de resultado')

hgb <- df |>
  ggplot()+
  aes(x=hgb, fill=result)+
  geom_density(alpha=0.6)+
  theme_minimal()+
  theme(legend.position = 'none')+
  labs(x='Hemoglobinas', y='Densidade',
       title='Densidade de hgb',
       subtitle = 'agrupada pela variável de resultado')

plat <- df |>
  ggplot()+
  aes(x=plat, fill=result)+
  geom_density(alpha=0.6)+
  theme_minimal()+
  theme(legend.position = 'right')+
  labs(x='Plaquetas', y='Densidade',
       title='Densidade de plat',
       fill='Resultado',
       subtitle = 'agrupada pela variável de resultado')

(age_quant+hct)/(hgb+plat)
```

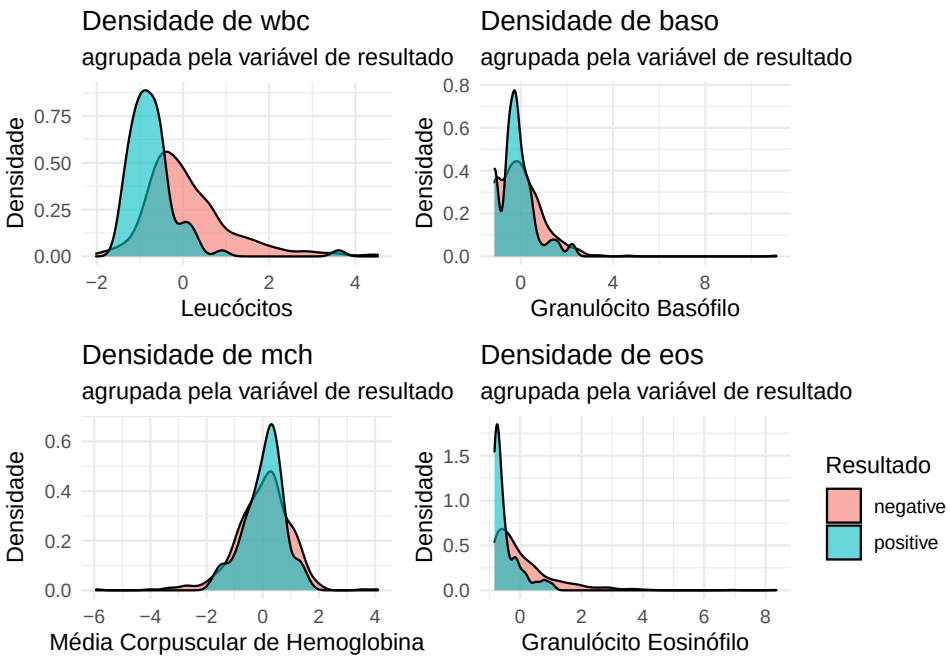


```r
mean_plat <- df |>
  ggplot()+
  aes(x=mean_plat, fill=result)+
  geom_density(alpha=0.6)+
  theme_minimal()+
  theme(legend.position = 'none')+
  labs(x='Média de Plaquetas', y='Densidade',
       title='Densidade de mean_plat',
       subtitle = 'agrupada pela variável de resultado')

rbc <- df |>
  ggplot()+
  aes(x=rbc, fill=result)+
  geom_density(alpha=0.6)+
  theme_minimal()+
  theme(legend.position = 'none')+
  labs(x='Células Vermelhas', y='Densidade',
       title='Densidade de rbc',
       subtitle = 'agrupada pela variável de resultado')

lym <- df |>
  ggplot()+
  aes(x=lym, fill=result)+
  geom_density(alpha=0.6)+
  theme_minimal()+
  theme(legend.position = 'none')+
  labs(x='Linfócitos', y='Densidade',
       title='Densidade de lym',
       subtitle = 'agrupada pela variável de resultado')

mchc <- df |>
  ggplot()+
  aes(x=mchc, fill=result)+
  geom_density(alpha=0.6)+
  theme_minimal()+
  theme(legend.position = 'right')+
  labs(x='Concentração Corpuscular Média de Hemoglobina', y='Densidade',
       title='Densidade de mchc',
       fill='Resultado',
       subtitle = 'agrupada pela variável de resultado')

(mean_plat+rbc)/(lym+mchc)
```



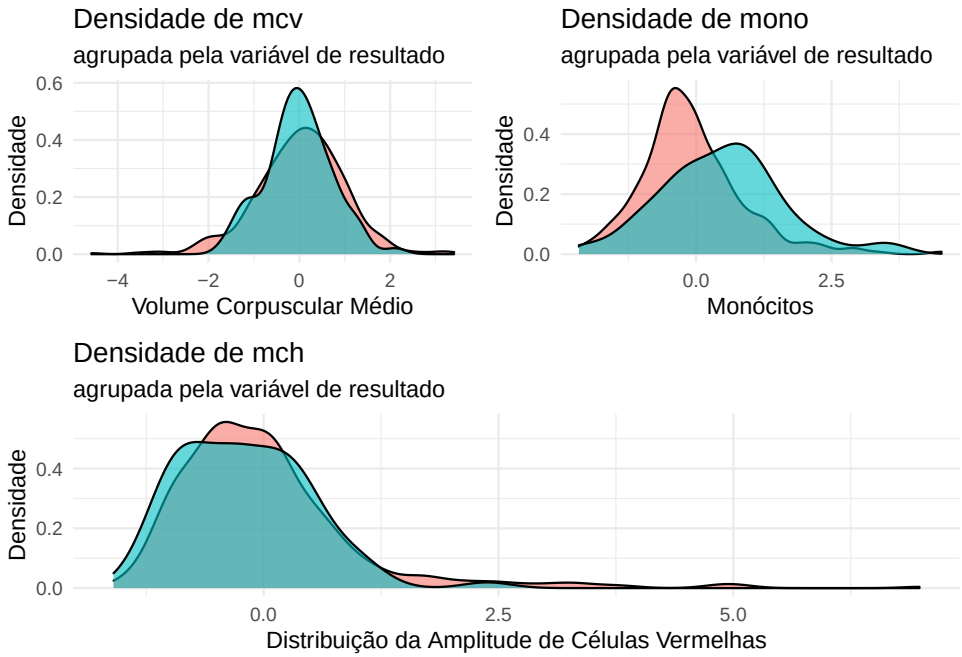

```
wbc <- df |>
  ggplot()+
  aes(x=wbc, fill=result)+
  geom_density(alpha=0.6)+
  theme_minimal()+
  theme(legend.position = 'none')+
  labs(x='Leucócitos', y='Densidade',
       title='Densidade de wbc',
       subtitle = 'agrupada pela variável de resultado')

baso <- df |>
  ggplot()+
  aes(x=baso, fill=result)+
  geom_density(alpha=0.6)+
  theme_minimal()+
  theme(legend.position = 'none')+
  labs(x='Granulócito Basófilo', y='Densidade',
       title='Densidade de baso',
       subtitle = 'agrupada pela variável de resultado')

mch <- df |>
  ggplot()+
  aes(x=mch, fill=result)+
  geom_density(alpha=0.6)+
  theme_minimal()+
  theme(legend.position = 'none')+
  labs(x='Média Corpuscular de Hemoglobina', y='Densidade',
       title='Densidade de mch',
       subtitle = 'agrupada pela variável de resultado')

eos <- df |>
```

```
  ggplot()+
  aes(x=eos, fill=result)+
  geom_density(alpha=0.6)+
  theme_minimal()+
  theme(legend.position = 'right')+
  labs(x='Granulócito Eosinófilo', y='Densidade',
       title='Densidade de eos',
       fill='Resultado',
       subtitle = 'agrupada pela variável de resultado')

(wbc+baso)/(mch+eos)
```




```
mcv <- df |>
  ggplot()+
  aes(x=mcv, fill=result)+
  geom_density(alpha=0.6)+
  theme_minimal()+
  theme(legend.position = 'none')+
  labs(x='Volume Corpuscular Médio', y='Densidade',
       title='Densidade de mcv',
       subtitle = 'agrupada pela variável de resultado')

mono <- df |>
  ggplot()+
  aes(x=mono, fill=result)+
  geom_density(alpha=0.6)+
  theme_minimal()+
  theme(legend.position = 'none')+
  labs(x='Monócitos', y='Densidade',
       title='Densidade de mono',
```

```
       subtitle = 'agrupada pela variável de resultado')

rdw <- df |>
  ggplot()+
  aes(x=rdw, fill=result)+
  geom_density(alpha=0.6)+
  theme_minimal()+
  theme(legend.position = 'none')+
  labs(x='Distribuição da Amplitude de Células Vermelhas', y='Densidade',
       title='Densidade de mch',
       subtitle = 'agrupada pela variável de resultado')

(mcv+mono)/rdw
```



Item B

```
### ITEM B: divisao dos dados

df <- df |>
  mutate(result=as.factor(ifelse(result=='negative',0,1)))

set.seed(57)

split <- initial_split(df, prop=0.6)

tre <- training(split)
tes <- testing(split)
```

```r
x_tre <- model.matrix(result~., tre)
y_tre <- pull(tre[,1])

x_tes <- model.matrix(result~., tes)
y_tes <- pull(tes[,1])
```

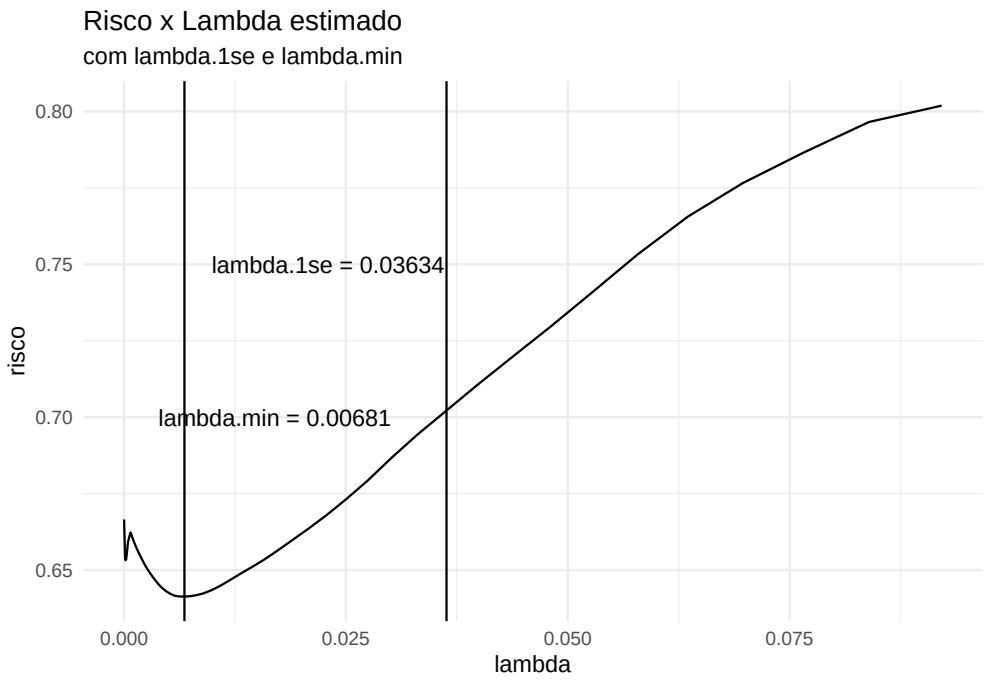
Item C

Plaquetas, Média de Plaquetas, Idade, Leucócitos, Eosinófilo, Basófilo, Média Corpuscular de Hemoglobina, Densidade de Mono, Volume Corpuscular Médio

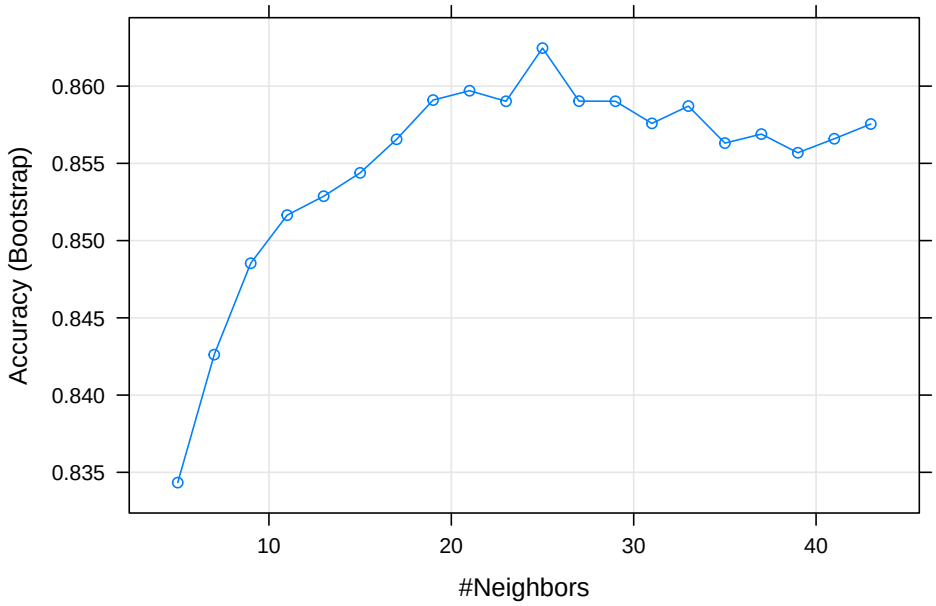
Item D

falar balanceamento

```r
### ITEM D

## Ajuste Lasso

cv_lasso <- cv.glmnet(x_tre, y_tre, alpha=1, family='binomial')
ajuste_lasso <- glmnet(x_tre, y_tre, alpha=1, lambda = cv_lasso$lambda.1se,
                       family='binomial')

## Erro x Lambda Lasso

tibble(
  lambda=cv_lasso$lambda,
  risco=cv_lasso$cvm
) |>
  ggplot()+
  aes(x=lambda, y=risco)+
  geom_line()+
  geom_vline(xintercept = cv_lasso$lambda.1se)+
  geom_vline(xintercept = cv_lasso$lambda.min)+
  annotate(geom = 'text', y=0.75, x=0.023,
           label=paste0('lambda.1se = ', round(ajuste_lasso$lambda,5)))+
  annotate(geom = 'text', y=0.7, x=0.017,
           label=paste0('lambda.min = ', round(cv_lasso$lambda.min,5)))+
  theme_minimal()+
  labs(title='Risco x Lambda estimado',
       subtitle = 'com lambda.1se e lambda.min')
```



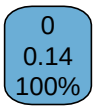

```
## Ajuste KNN

# Realizando calculo do melhor K

ajuste_knn <- train(
  x=x_tre,
  y=y_tre,
  method = 'knn',
  tuneLength = 20
)

# Plotando grafico de K vs Risco

plot(ajuste_knn)
```



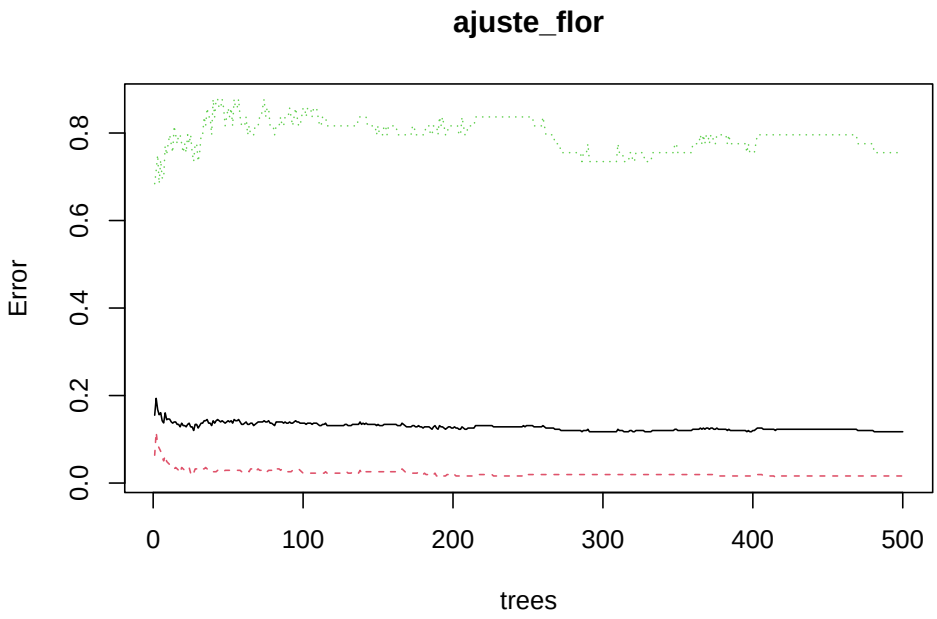

```
paste0('O melhor K é: ', ajuste_knn$bestTune)
```

```
## [1] "O melhor K é: 25"
```

```
## Arvore de Decisão

tre_arv <- data.frame(y_tre, x_tre[,-1])

ajuste_arv <- rpart(y_tre~., data=tre_arv)
melhor_cp <- ajuste_arv$cptable[which.min(ajuste_arv$cptable[,'xerror']),'CP']
ajuste_arv <- prune(ajuste_arv, cp=melhor_cp)

rpart.plot(ajuste_arv)
```




```
## Floresta Aleatória

# Realizando Ajuste

ajuste_flor <- randomForest(x=x_tre, y=y_tre, mtry = round(ncol(df)/3),
                            importance=TRUE)

# Ajuste x Numero de Arvores

plot(ajuste_flor)
```



Item E

Item F

Item G