Universidade de Brasília

DEPARTAMENTO DE ESTATÍSTICA

18 julho 2023

Projeto - Análise de Discriminantes e Classificação, com validação cruzada Jackknife e Hold-out, utilizando modelos bayesianos.

Prof. Dr. Donald Matthew Pianto

Aluno: Bruno Gondim Toledo

Matrícula: 15/0167636

Estatística Computacional

1º/2023

```
if (!require("pacman")) install.packages("pacman")
```

```
## Carregando pacotes exigidos: pacman
```

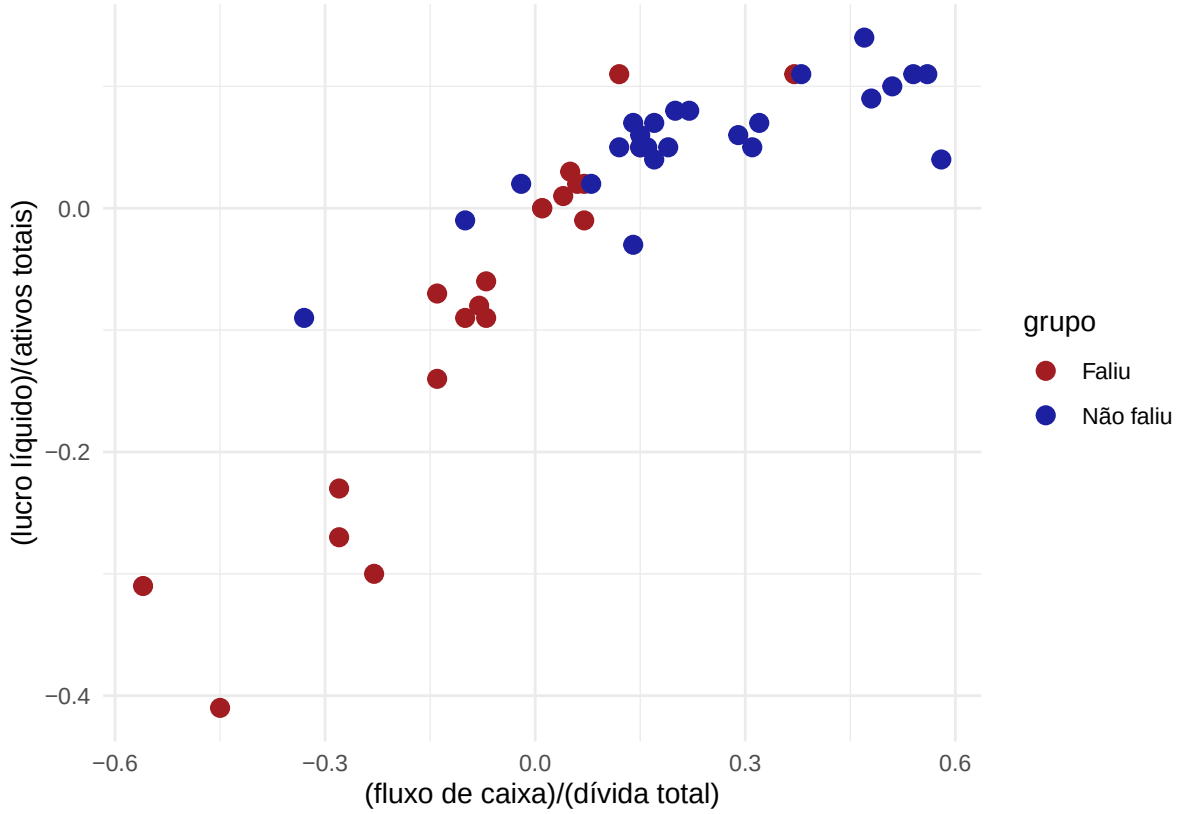
```
p_load(knitr,effectsize,DescTools,tidyverse,MASS,klaR,knitr,cowplot,nlme,
       Rchoice,AICcmodavg,mdscore,questionr,mda,mvnTest,gclus,mclust,caTools)
M <- 150167636
```

Johnson e Wichern - Exercício 11.24.

```
dados <- read_table("dados/T11-4-BankruptcyData.DAT.txt",
                    col_names = FALSE, col_types = cols(X6 = col_skip()))
dados$X5 <- factor(dados$X5)
```

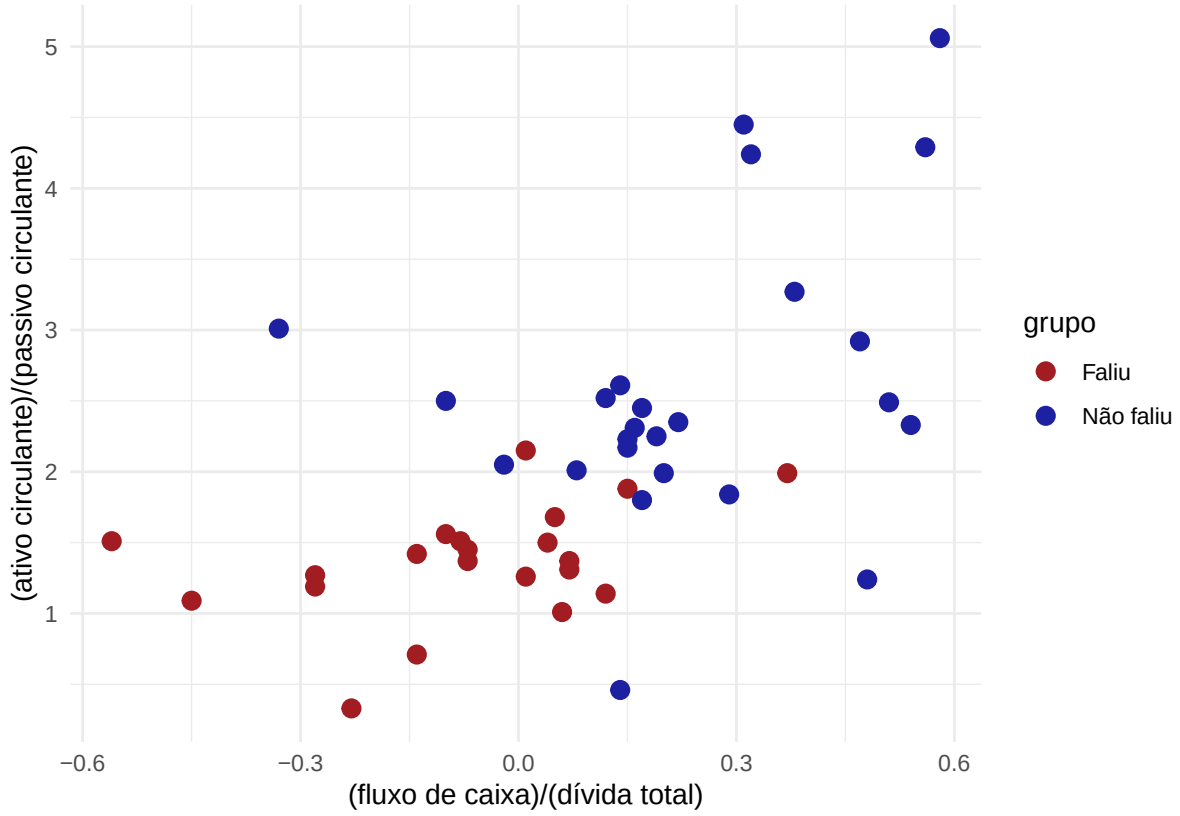
a)

```
dados <- dados |>
  mutate(grupo = ifelse(X5 == 0,"Faliu","Não faliu"))

ggplot(dados, aes(x = X1, y = X2, color = grupo)) +
  geom_point(size = 3) +
  labs(
    x = "(fluxo de caixa)/(dívida total)",
    y = "(lucro líquido)/(ativos totais)"
  ) +
  scale_color_manual(values = c("#A11D21", "#1D21A1")) +
  theme_minimal()
```

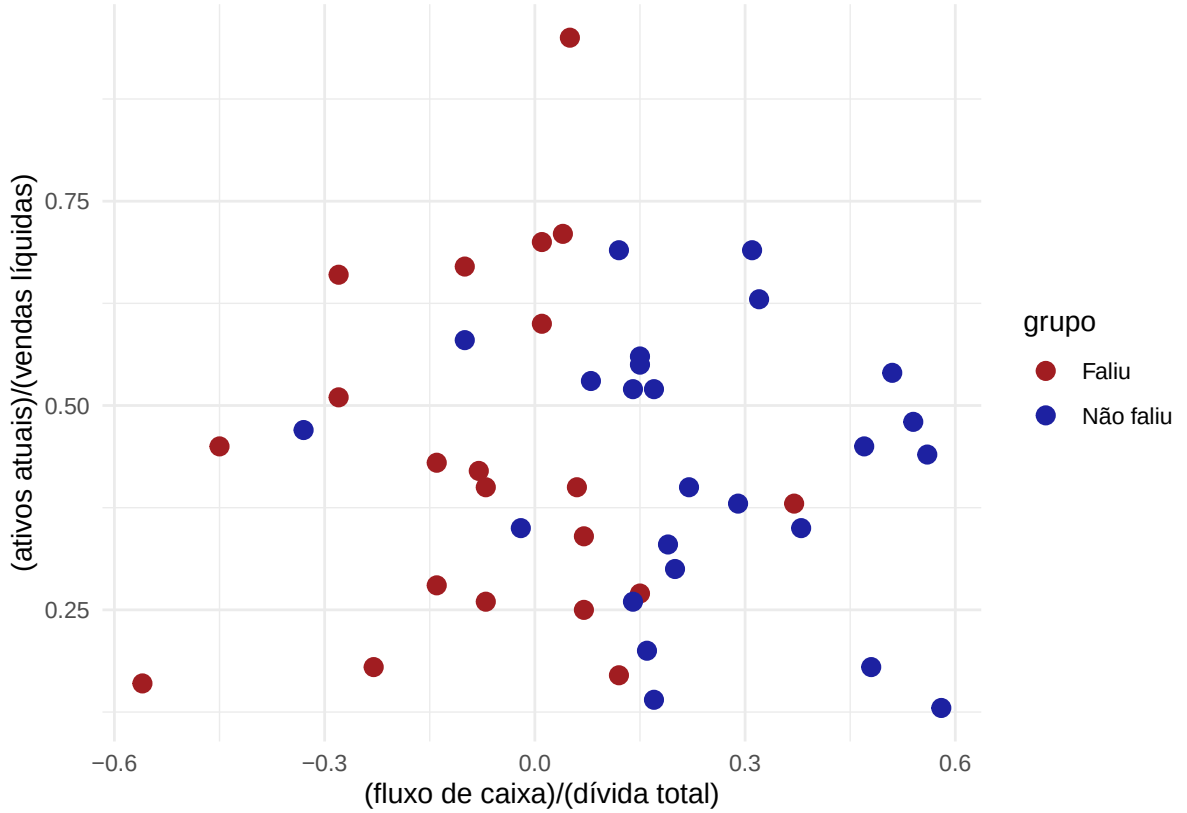



```
ggplot(dados, aes(x = X1, y = X3, color = grupo)) +
  geom_point(size = 3) +
```

```
  labs(
    x = "(fluxo de caixa)/(dívida total)",
    y = "(ativo circulante)/(passivo circulante)") +
  scale_color_manual(values = c("#A11D21", "#1D21A1")) +
  theme_minimal()
```




```
ggplot(dados, aes(x = X1, y = X4, color = grupo)) +
  geom_point(size = 3) +
  labs(
    x = "(fluxo de caixa)/(dívida total)",
    y = "(ativos atuais)/(vendas líquidas)") +
  scale_color_manual(values = c("#A11D21", "#1D21A1")) +
  theme_minimal()
```



Em todos os gráficos, os pontos lembram a forma de elipsoides. Portanto, graficamente, não é possível rejeitar a normalidade bivariada dos dados.

b)

```
falidos <- dados |>
  filter(X5 == 0) |>
  dplyr::select(X1,X2) |>
  summarise_all(mean)
falidos <- as.matrix(falidos)
colnames(falidos) <- NULL

ativos <- dados |>
  filter(X5==1) |>
  dplyr::select(X1,X2) |>
  summarise_all(mean)
ativos <- as.matrix(ativos)
colnames(ativos) <- NULL

xb1 <- falidos
xb2 <- ativos

falidos <- dados |>
  filter(X5 == 0) |>
  dplyr::select(X1,X2)

ativos <- dados |>
  filter(X5==1) |>
  dplyr::select(X1,X2)

S1 <- cov(falidos)
```

```
S2 <- cov(ativos)
```

Considerando 1 como o grupo de empresas que faliram (falidos) e 2 como o grupo de empresas que não faliram ainda (ativos), temos os vetores de média μ_1', μ_2' dados respectivamente por: [-0.0690476, -0.0814286],[0.2352, 0.0556], e matrizes de covariância $S_1 =$

	X1	X2
X1	0.0441290	0.0284764
X2	0.0284764	0.0210029

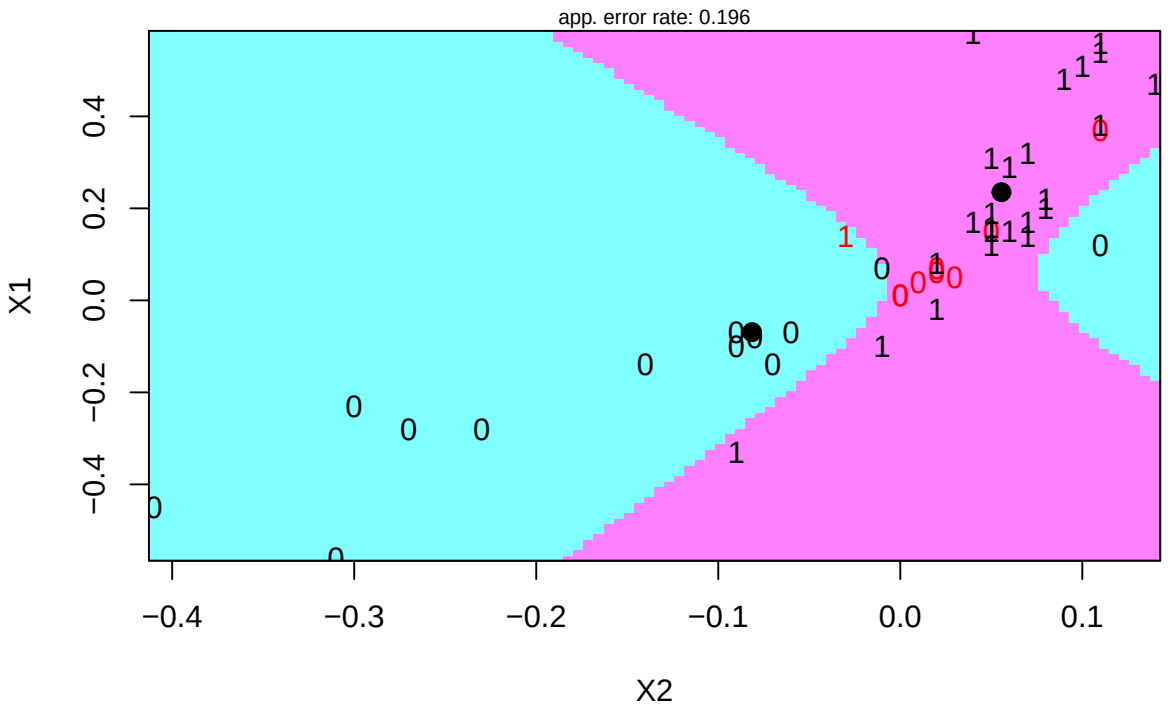
e $S_2 =$

	X1	X2
X1	0.0470510	0.0085072
X2	0.0085072	0.0023757

c)

Como para este conjunto não rejeitamos a hipótese de normalidade multivariada (apesar de termos feito apenas análise gráfica), e, apesar de não termos testado a igualdade das variâncias, elas aparentam ser diferentes; portanto a abordagem mais adequada para este caso é a análise discriminante quadrática abaixo. No caso, foram definido custos e prioris iguais para ambos os grupos.

```
dados <- dados[,-6]
gqda <- qda(X5~X1+X2, data = dados,prior =c(.5,.5))

gqdap1 <- predict(gqda)
gqctable1 <- table(dados$X5, gqdap1$class)

partimat(X5~X1+X2, data=dados, method="qda",
         plot.matrix = F, imageplot = T,prec=100)
```




```
# Com validação cruzada
gqdaVC <- qda(X5~X1+X2, data = dados,prior =c(.5,.5),CV=T)
```

Matriz de confusão:

	0	1
0	13	8
1	1	24

Proporção de classificações corretas em cada grupo:

	x
0	0.6190476
1	0.9600000

Proporção total de classificação correta: 0.8043478

d)

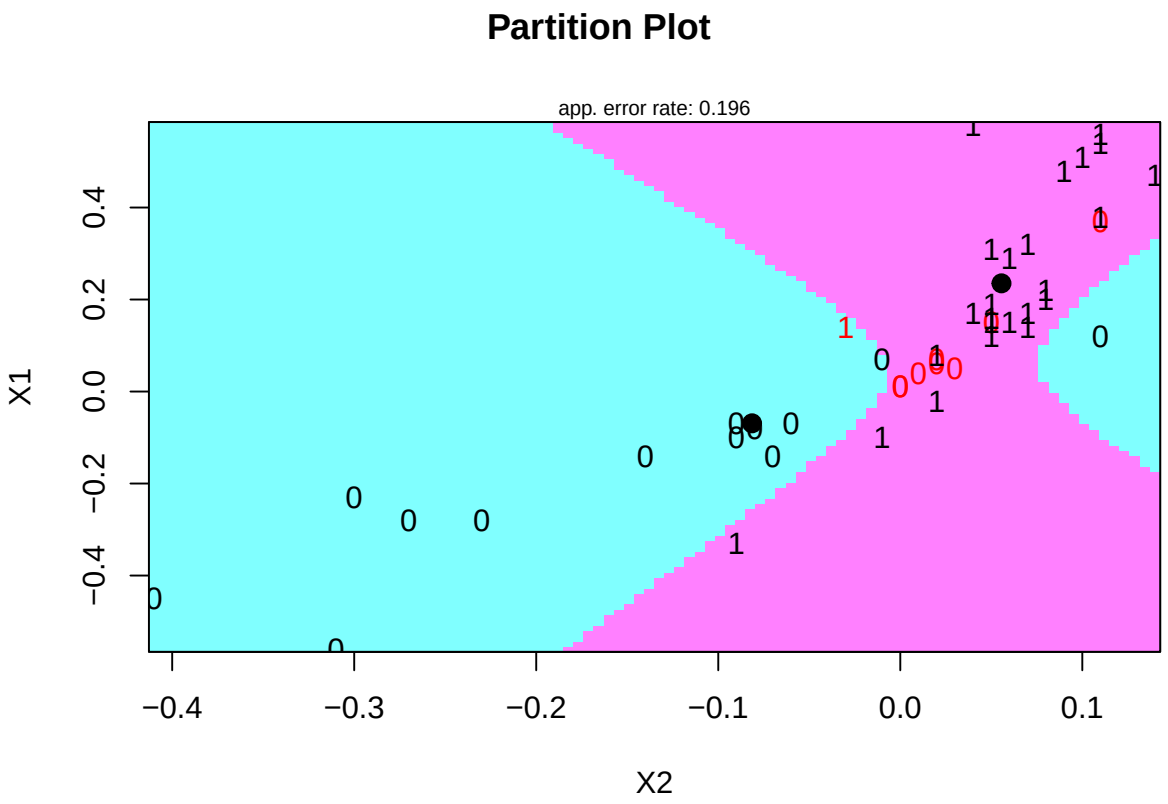
```
# Matrizes de confusão:
M <- table(dados$X5, gqdap1$class)
MCV <- table(dados$X5, gqdaVC$class)

# APER e \hat{E}APR:
APER <- (sum(M)-sum(diag(M)))/sum(M) # APER x_1,x_2
E_APR <- (sum(MCV)-sum(diag(MCV)))/sum(MCV) # \hat{E} APR x_1,x_2
```

O erro aparente (APER) deste conjunto foi calculado como sendo 0.1956522; enquanto que a estimação da taxa de erro aparente ($\hat{E}(AER)$) foi calculada como 0.2173913. Notamos que apesar de o erro estimado

via validação cruzada Jackknife ter sido maior que o erro aparente, esta é uma estimativa mais robusta em comparação com o resultado sem validação cruzada.

e)

```
#dados <- dados[,-6]
gqda <- qda(X5~X1+X2, data = dados,prior =c(.05,.95))

gqdap1 <- predict(gqda)
gqctable1 <- table(dados$X5, gqdap1$class)

partimat(X5~X1+X2, data=dados, method="qda",
         plot.matrix = F, imageplot = T,prec=100)
```




```
# Com validação cruzada
gqdaVC <- qda(X5~X1+X2, data = dados,prior =c(.05,.95),CV=T)

# Matrizes de confusão:
M <- table(dados$X5, gqdap1$class)
MCV <- table(dados$X5, gqdaVC$class)

# APER e \hat{E}APR:
APER <- (sum(M)-sum(diag(M)))/sum(M) # APER x_1,x_2
E_APR <- (sum(MCV)-sum(diag(MCV)))/sum(MCV) # \hat{E} APR x_1,x_2
```

Matriz de confusão:

	0	1
0	9	12
1	0	25

Proporção de classificações corretas em cada grupo:

	x
0	0.4285714
1	1.0000000

Proporção total de classificação correta: 0.7391304

O erro aparente (APER):0.2608696

Estimativa da taxa de erro aparente ($\hat{E}(AER)$): 0.2608696

Analisando os APER e $\hat{E}(AER)$, concluímos que as prioris iguais ($p_1 = 0,5; p_2 = 0,5$) tem um erro de classificação inferior se comparado as prioris desiguais ($p_1 = 0,05; p_2 = 0,95$). Neste caso, notamos que tanto o APER quanto o $\hat{E}(AER)$ deram resultados idênticos.

f)

```
#xb1
#xb2
al <- t(t(xb1-xb2)) %*% solve(cov_pooled(ativos,falidos))
m <- t(t(xb1-xb2)) %*% solve(cov_pooled(ativos,falidos)) %*% t(xb1+xb2)

pop1 <- falidos |>
  rowwise() |>
  mutate(M = al %*% c(X1, X2)) |>
  mutate(pop = ifelse(M > m,"p1","p2")) |>
  pull()
pop1 <- factor(pop1)

pop2 <- ativos |>
  rowwise() |>
  mutate(M = al %*% c(X1, X2)) |>
  mutate(pop = ifelse(M > m,"p1","p2")) |>
  pull()
pop2 <- factor(pop2)
#summary(pop1)
#summary(pop2)

# APER:
APER <- 8/46

# AVALIAÇÃO: Como as matrizes S_1 e S_2 aparentam ser diferentes, esta técnica não é a mais adequada
```

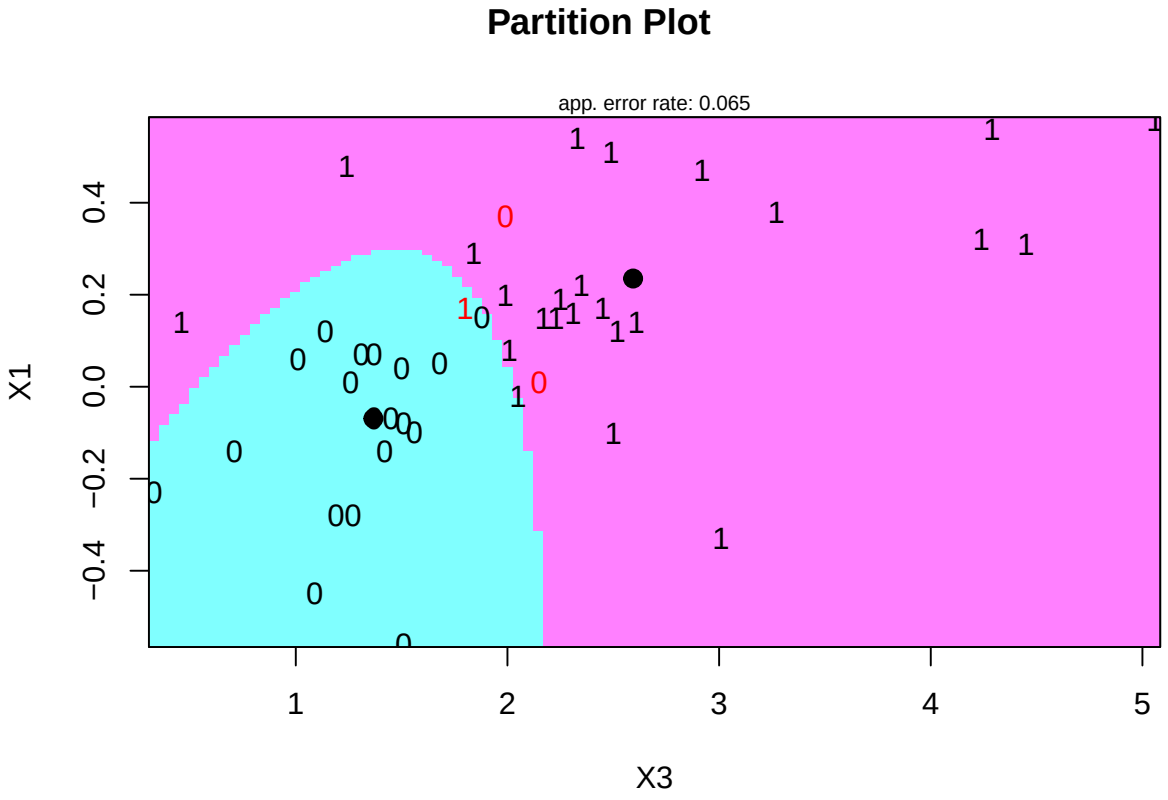
Como as matrizes S_1 e S_2 aparentam ser diferentes, esta técnica não aparenta ser a mais adequada. Entretanto, tomando como base apenas a performance do APER = 0.173913, até que a classificação por discriminantes lineares não ficou ruim, com resultados até melhores do que os obtido pelos discriminantes quadráticos.

g)

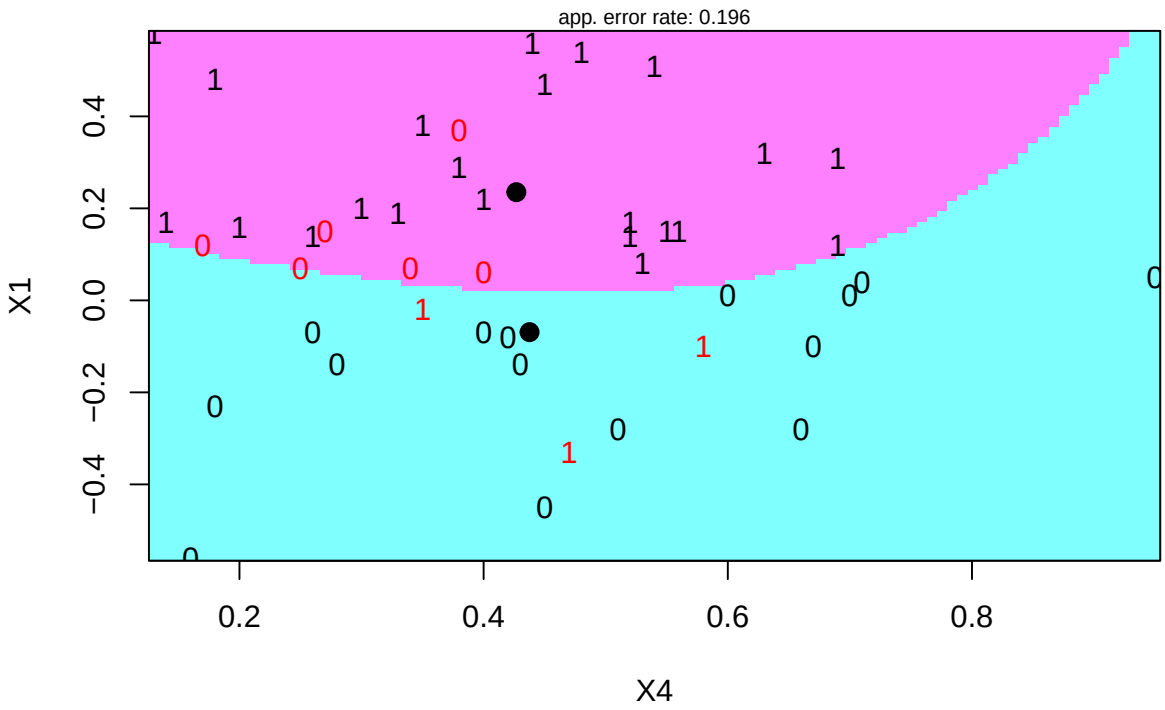
```
#b)
# x1,x3
falidos2 <- dados |>
  filter(X5 == 0) |>
  dplyr::select(X1,X3) |>
  summarise_all(mean)
falidos2 <- as.matrix(falidos2)
```

```r
colnames(falidos2) <- NULL

ativos2 <- dados |>
  filter(X5==1) |>
  dplyr::select(X1,X3) |>
  summarise_all(mean)
ativos2 <- as.matrix(ativos2)
colnames(ativos2) <- NULL

xb12 <- falidos2
xb22 <- ativos2

falidos2 <- dados |>
  filter(X5 == 0) |>
  dplyr::select(X1,X3)

ativos2 <- dados |>
  filter(X5==1) |>
  dplyr::select(X1,X3)

S12 <- cov(falidos2)
S22 <- cov(ativos2)

#x1,x4

falidos3 <- dados |>
  filter(X5 == 0) |>
  dplyr::select(X1,X4) |>
  summarise_all(mean)
falidos3 <- as.matrix(falidos3)
colnames(falidos3) <- NULL

ativos3 <- dados |>
  filter(X5==1) |>
  dplyr::select(X1,X4) |>
  summarise_all(mean)
ativos3 <- as.matrix(ativos3)
colnames(ativos3) <- NULL

xb13 <- falidos3
xb23 <- ativos3

falidos3 <- dados |>
  filter(X5 == 0) |>
  dplyr::select(X1,X4)

ativos3 <- dados |>
  filter(X5==1) |>
  dplyr::select(X1,X4)

S13 <- cov(falidos3)
S23 <- cov(ativos3)

#c)
# x_1,x_3 ----

gqda <- qda(X5~X1+X3, data = dados,prior =c(.5,.5))
```

```r
gqdap2 <- predict(gqda)
gqctable2 <- table(dados$X5, gqdap2$class)
prop2 <- diag(prop.table(gqctable2,1)) # prop de classif. correta no grupo
propt2 <- sum(diag(prop.table(gqctable2))) # prop total de classf. correta

partimat(X5~X1+X3, data=dados, method="qda",
         plot.matrix = F, imageplot = T,prec=100)
```




```r
# Com validação cruzada
gqdaVC2 <- qda(X5~X1+X3, data = dados,prior =c(.5,.5),CV=T)

# Matrizes de confusão:
M2 <- table(dados$X5, gqdap2$class)
MCV2 <- table(dados$X5, gqdaVC2$class)

# APER e \hat{E}APR:
APER2 <- (sum(M2)-sum(diag(M2)))/sum(M2) # APER x_1,x_2
E_APR2 <- (sum(MCV2)-sum(diag(MCV2)))/sum(MCV2) # \hat{E} APR x_1,x_2

# x_1,x_4 ----

gqda <- qda(X5~X1+X4, data = dados,prior =c(.5,.5))

gqdap3 <- predict(gqda)
gqctable3 <- table(dados$X5, gqdap3$class)
prop3 <- (diag(prop.table(gqctable3,1))) # prop de classif. correta no grupo
propt3 <- (sum(diag(prop.table(gqctable3)))) # prop total de classf. correta

partimat(X5~X1+X4, data=dados, method="qda",
         plot.matrix = F, imageplot = T,prec=100)
```




```
# Com validação cruzada
gqdaVC3 <- qda(X5~X1+X4, data = dados,prior =c(.5,.5),CV=T)

# Matrizes de confusão:
M3 <- table(dados$X5, gqdap3$class)
MCV3 <- table(dados$X5, gqdaVC3$class)

# APER e \hat{E}APR:
APER3 <- (sum(M3)-sum(diag(M3)))/sum(M3) # APER x_1,x_2
E_APR3 <- (sum(MCV3)-sum(diag(MCV3)))/sum(MCV3) # \hat{E} APR x_1,x_2

#e)
# x_1,x_3 ----

gqda <- qda(X5~X1+X3, data = dados,prior =c(.05,.95))

gqdap4 <- predict(gqda)
gqctable4 <- table(dados$X5, gqdap4$class)
prop4 <- (diag(prop.table(gqctable4,1))) # prop de classif. correta no grupo
propt4 <- (sum(diag(prop.table(gqctable4)))) # prop total de classf. correta

# Com validação cruzada
gqdaVC4 <- qda(X5~X1+X3, data = dados,prior =c(.05,.95),CV=T)

# Matrizes de confusão:
M4 <- table(dados$X5, gqdap4$class)
MCV4 <- table(dados$X5, gqdaVC4$class)

# APER e \hat{E}APR:
APER4 <- (sum(M4)-sum(diag(M4)))/sum(M4) # APER x_1,x_2
E_APR4 <- (sum(MCV4)-sum(diag(MCV4)))/sum(MCV4) # \hat{E} APR x_1,x_2
```

```
# x_1,x_4 ----

gqda <- qda(X5~X1+X4, data = dados,prior =c(.05,.95))

gqdap5 <- predict(gqda)
gqctable5 <- table(dados$X5, gqdap5$class)
prop5 <- (diag(prop.table(gqctable5,1))) # prop de classif. correta no grupo
propt5 <- (sum(diag(prop.table(gqctable5)))) # prop total de classf. correta

# Com validação cruzada
gqdaVC5 <- qda(X5~X1+X4, data = dados,prior =c(.05,.95),CV=T)

# Matrizes de confusão:
M5 <- table(dados$X5, gqdap5$class)
MCV5 <- table(dados$X5, gqdaVC5$class)

# APER e \hat{E}APR:
APER5 <- (sum(M5)-sum(diag(M5)))/sum(M5) # APER x_1,x_2
E_APR5 <- (sum(MCV5)-sum(diag(MCV5)))/sum(MCV5) # \hat{E} APR x_1,x_2
```

Vetores de média e matrizes de covariância para as variáveis (x1,x3):

Vetor média μ_1' = -0.0690476, 1.3666667

Vetor Média μ_3' = 0.2352, 2.5936

Matriz de covariância S_1 =

	X1	X3
X1	0.0441290	0.0344933
X3	0.0344933	0.1643033

Matriz de covariância S_3 =

	X1	X3
X1	0.0470510	0.0749305
X3	0.0749305	1.0467740

Análise discriminante quadrática, com prioris = (0,5;0,5), utilizando as variáveis (x1,x3):

Matriz de confusão:

	0	1
0	19	2
1	3	22

Proporção de classificações corretas em cada grupo:

	x
0	0.9047619
1	0.8800000

Proporção total de classificação correta: 0.8913043

Erro aparente (APER):0.1086957

Estimativa da taxa de erro aparente ($\hat{E}(AER)$): 0.1304348

Análise discriminante quadrática, com prioris = (0,05;0,95), utilizando as variáveis (x1,x3):

Matriz de confusão:

	0	1
0	4	17
1	0	25

Proporção de classificações corretas em cada grupo:

	x
0	0.1904762
1	1.0000000

Proporção total de classificação correta: 0.6304348

Erro aparente (APER):0.3695652

Estimativa da taxa de erro aparente ($\hat{E}(AER)$): 0.3913043

Vetores de média e matrizes de covariância para as variáveis (x1,x4):

Vetor média μ_1' = -0.0690476, 0.437619

Vetor Média μ_3' = 0.2352, 0.4268

Matriz de covariância S_1 =

	X1	X4
X1	0.0441290	0.0041474
X4	0.0041474	0.0445790

Matriz de covariância S_3 =

	X1	X4
X1	0.0470510	-0.0067035
X4	-0.0067035	0.0263810

Análise discriminante quadrática, com prioris = (0,5;0,5), utilizando as variáveis (x1,x4):

Matriz de confusão:

	0	1
0	17	4
1	4	21

Proporção de classificações corretas em cada grupo:

	x
0	0.8095238
1	0.8400000

Proporção total de classificação correta: 0.826087

Erro aparente (APER):0.173913

Estimativa da taxa de erro aparente ($\hat{E}(AER)$): 0.2173913

Análise discriminante quadrática, com prioris = (0,05;0,95), utilizando as variáveis (x1,x4):

Matriz de confusão:

	0	1
0	3	18
1	0	25

Proporção de classificações corretas em cada grupo:

	x
0	0.1428571
1	1.0000000

Proporção total de classificação correta: 0.6086957

Erro aparente (APER):0.3913043

Estimativa da taxa de erro aparente ($\hat{E}(AER)$): 0.4565217

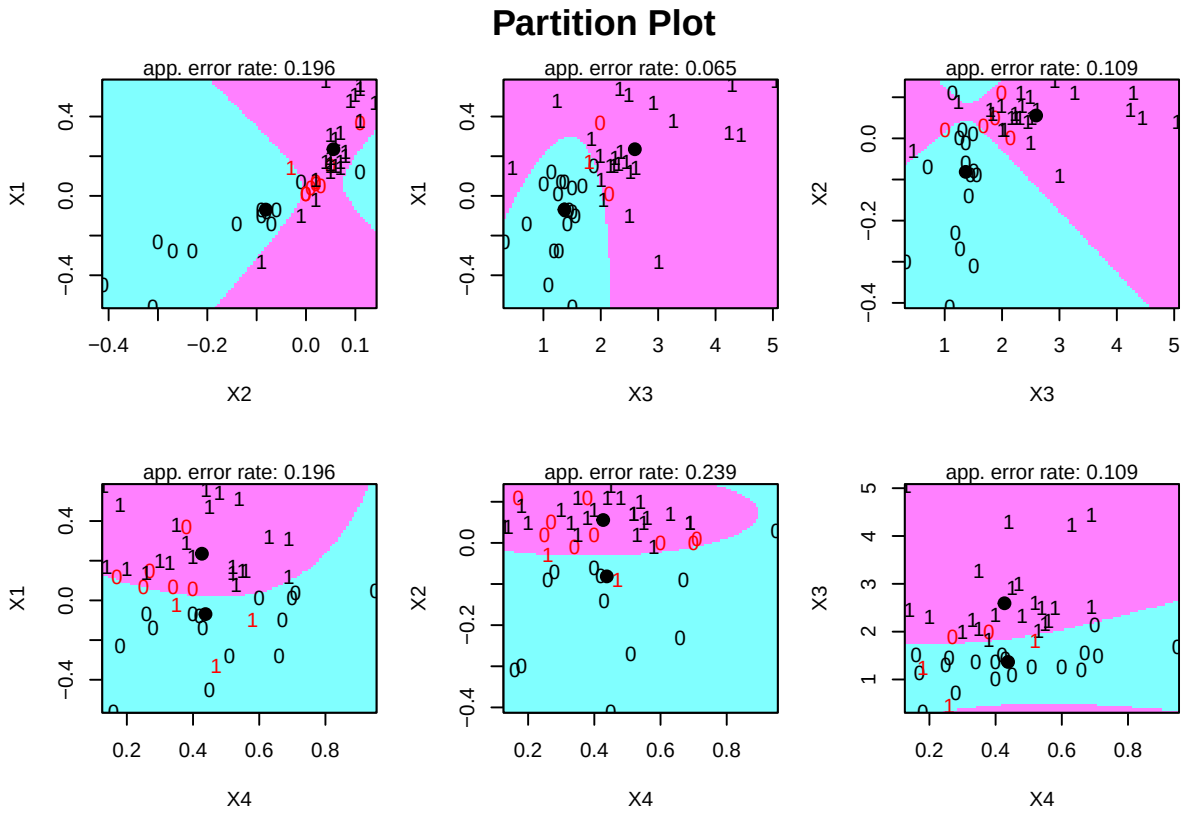
Conclusões:

De fato, os resultados encontrados foram bastante distintos para cada caso. Analisando somente os APER e $\hat{E}(AER)$, notamos que a análise em que foi observado o menor valor de ambos foi a análise executada utilizando as variáveis (x_1, x_3), com prioris iguais $(0,5; 0,5)$, enquanto que os maiores valores foram observados para o modelo em que utilizei as variáveis (x_1, x_4) com prioris desiguais $(0,05; 0,95)$. O modelo que menos variou estas duas estatísticas para ambas as prioris testadas $(0,5; 0,5)$ e $(0,05; 0,95)$ foi o modelo inicialmente testado com as variáveis (x_1, x_2). Com base nisso, podemos concluir que tanto a escolha das variáveis quanto a escolha das prioris, influenciam bastante na qualidade do modelo final.

h)

```
# x1,x2,x3,x4

falidos4 <- dados |>
  filter(X5 == 0) |>
  dplyr::select(X1,X2,X3,X4) |>
  summarise_all(mean)
falidos4 <- as.matrix(falidos4)
colnames(falidos4) <- NULL

ativos4 <- dados |>
  filter(X5==1) |>
  dplyr::select(X1,X2,X3,X4) |>
  summarise_all(mean)
ativos4 <- as.matrix(ativos4)
```

```r
colnames(ativos4) <- NULL

xb14 <- falidos4
xb24 <- ativos4

falidos4 <- dados |>
  filter(X5 == 0) |>
  dplyr::select(X1,X2,X3,X4)

ativos4 <- dados |>
  filter(X5==1) |>
  dplyr::select(X1,X2,X3,X4)

S14 <- cov(falidos4)
S24 <- cov(ativos4)

# Modelo c priori igual

gqda6 <- qda(X5~X1+X2+X3+X4, data = dados,prior =c(.5,.5))

gqdap6 <- predict(gqda6)
gqctable6 <- table(dados$X5, gqdap6$class)
prop6 <- diag(prop.table(gqctable6,1)) # prop de classif. correta no grupo
propt6 <- (sum(diag(prop.table(gqctable6)))) # prop total de classf. correta

partimat(X5~X1+X2+X3+X4, data=dados, method="qda",
         plot.matrix = F, imageplot = T,prec=100)
```




```r
# Com validação cruzada
gqdaVC6 <- qda(X5~X1+X2+X3+X4, data = dados,prior =c(.5,.5),CV=T)
```

```r
# Matrizes de confusão:
M6 <- table(dados$X5, gqdap6$class)
MCV6 <- table(dados$X5, gqdaVC6$class)

# APER e \hat{E}APR:
APER6 <- (sum(M6)-sum(diag(M6)))/sum(M6) # APER x_1,x_2
E_APR6 <- (sum(MCV6)-sum(diag(MCV6)))/sum(MCV6) # \hat{E} APR x_1,x_2


# Alterando a priori
gqda7 <- qda(X5~X1+X2+X3+X4, data = dados,prior =c(.05,.95))

gqdap7 <- predict(gqda7)
gqctable7 <- table(dados$X5, gqdap7$class)
prop7 <- (diag(prop.table(gqctable7,1))) # prop de classif. correta no grupo
propt7 <- (sum(diag(prop.table(gqctable7)))) # prop total de classf. correta

# Com validação cruzada
gqdaVC7 <- qda(X5~X1+X2+X3+X4, data = dados,prior =c(.05,.95),CV=T)

# Matrizes de confusão:
M7 <- table(dados$X5, gqdap7$class)
MCV7 <- table(dados$X5, gqdaVC7$class)

# APER e \hat{E}APR:
APER7 <- (sum(M7)-sum(diag(M7)))/sum(M7) # APER x_1,x_2
E_APR7 <- (sum(MCV7)-sum(diag(MCV7)))/sum(MCV7) # \hat{E} APR x_1,x_2
```

Vetores de média e matrizes de covariância para as variáveis (x1,x2,x3,x4):

Vetor média μ_1' = -0.0690476, -0.0814286, 1.3666667, 0.437619

Vetor Média μ_3' = 0.2352, 0.0556, 2.5936, 0.4268

Matriz de covariância S_1 =

	X1	X2	X3	X4
X1	0.0441290	0.0284764	0.0344933	0.0041474
X2	0.0284764	0.0210029	0.0260200	0.0034414
X3	0.0344933	0.0260200	0.1643033	0.0327817
X4	0.0041474	0.0034414	0.0327817	0.0445790

Matriz de covariância S_3 =

	X1	X2	X3	X4
X1	0.0470510	0.0085072	0.0749305	-0.0067035
X2	0.0085072	0.0023757	0.0085832	0.0001853
X3	0.0749305	0.0085832	1.0467740	0.0326328
X4	-0.0067035	0.0001853	0.0326328	0.0263810

Análise discriminante quadrática, com prioris = (0,5;0,5), utilizando as variáveis (x1,x2,x3,x4):

Matriz de confusão:

	0	1
0	19	2
1	1	24

Proporção de classificações corretas em cada grupo:

	x
0	0.9047619
1	0.9600000

Proporção total de classificação correta: 0.9347826

Erro aparente (APER):0.0652174

Estimativa da taxa de erro aparente ($\hat{E}(AER)$): 0.1086957

Análise discriminante quadrática, com prioris = (0,05;0,95), utilizando as variáveis (x1,x2,x3,x4):

Matriz de confusão:

	0	1
0	12	9
1	0	25

Proporção de classificações corretas em cada grupo:

	x
0	0.5714286
1	1.0000000

Proporção total de classificação correta: 0.8043478

Erro aparente (APER):0.1956522

Estimativa da taxa de erro aparente ($\hat{E}(AER)$): 0.2391304

No caso da inclusão de todas as 4 variáveis, o classificador com prioris iguais produziu as melhores classificações (menores APER e $\hat{E}(AER)$). Também neste caso, o classificador com prioris $(0,05; 0,95)$ produziu um APER significativamente maior que o mesmo modelo com prioris iguais, porém foram os menores valores se comparados com os valores observados nos demais modelos com prioris $(0,05; 0,95)$.

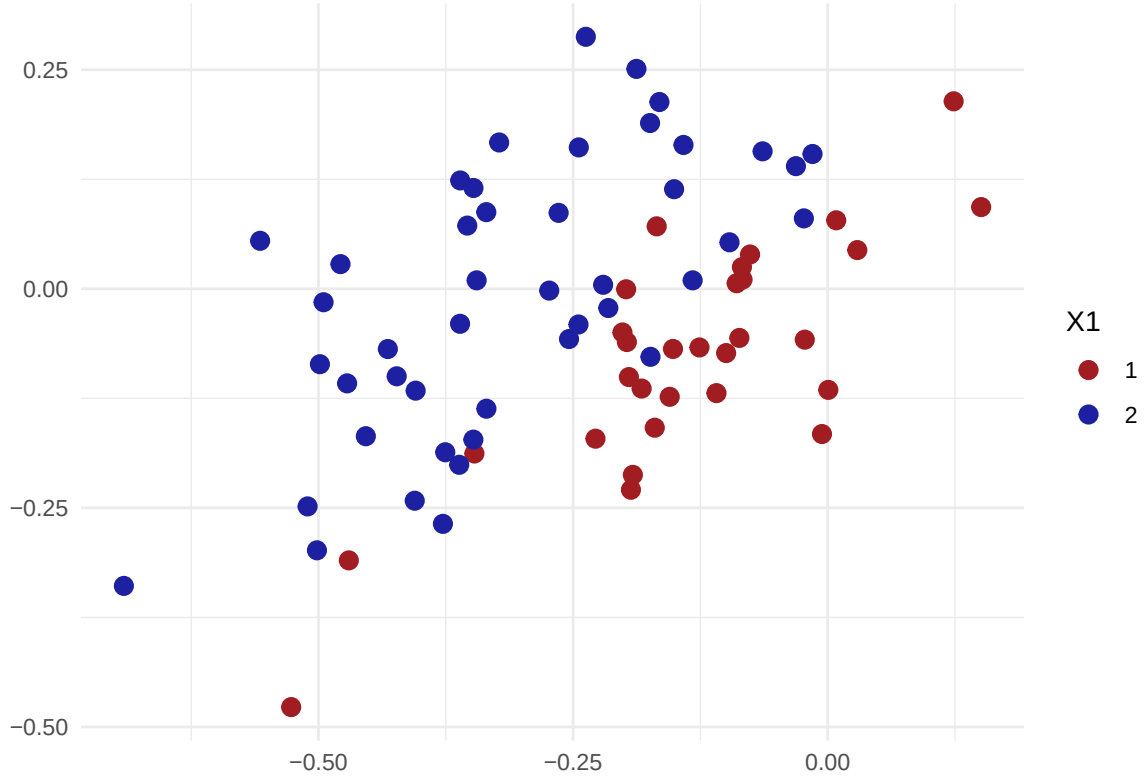
Conclusões:

Isto nos leva a acreditar que a inclusão de mais variáveis foi bom para o modelo, produzindo os menores erros aparentes. Entretando, a diferença não foi tão substantiva assim, então, deve-se considerar questões como verba para coleta de tantas variáveis, complexidade da análise e viabilidade de novas coletas caso deseje-se seguir com o modelo mais preciso.

Johnson e Wichern - Exercício 11.32.

a)

```r
ggplot(dados, aes(x = X2, y = X3, color = X1)) +
  geom_point(size = 3) +
  labs(
    x = "",
    y = ""
  ) +
  scale_color_manual(values = c("#A11D21", "#1D21A1")) +
  theme_minimal()
```



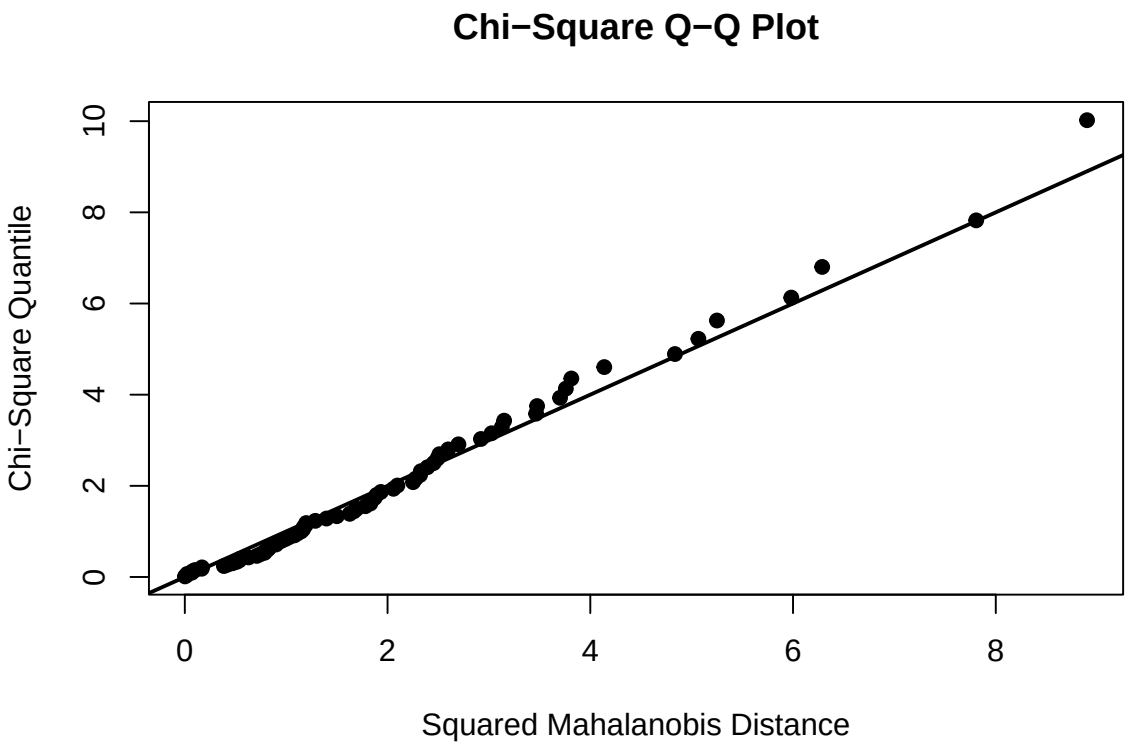

```r
shapiro.test(dados$X2)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  dados$X2
## W = 0.98496, p-value = 0.5185
```

```r
shapiro.test(dados$X3)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  dados$X3
## W = 0.99255, p-value = 0.9428
```

```r
AD.test(dados[,2:3], qqplot = TRUE)
```




```
##             Anderson-Darling test for Multivariate Normality
##
##   data : dados[, 2:3]
##
##   AD              : 0.7583493
##   p-value         : 0.2362764
##
##   Result  : Data are multivariate normal (sig.level = 0.05)
```

Através da análise visual, não é possível rejeitar a normalidade bivariada, visto que os pontos aparentam formar uma elipsoide. Foi realizado ainda testes de Shapiro-Wilk nas duas marginais, que também não rejeitaram a normalidade; univariada, neste caso. Foi ainda utilizado o teste de Anderson-Darling para normalidade multivariada do pacote `mvnTest`, que também não rejeitou a normalidade multivariada. Portanto, não temos evidências para rejeitar a hipótese de normalidade multivariada dos dados.

b)

```r
medias1 <- dados |>
  filter(X1 == 1) |>
  dplyr::select(X2,X3) |>
  summarise_all(mean)
medias1 <- as.matrix(medias1)
colnames(medias1) <- NULL

medias2 <- dados |>
  filter(X1==2) |>
  dplyr::select(X2,X3) |>
  summarise_all(mean)
medias2 <- as.matrix(medias2)
colnames(medias2) <- NULL
```

```r
xb1 <- medias1
xb2 <- medias2

grupo1 <- dados |>
  filter(X1 == 1) |>
  dplyr::select(X2,X3)

grupo2 <- dados |>
  filter(X1==2) |>
  dplyr::select(X2,X3)

al <- t(t(medias1-medias2)) %*% solve(cov_pooled(grupo1,grupo2))

m <- .5*(al%*%t(medias1)+al%*%t(medias2))

pop1 <- grupo1 |>
  rowwise() |>
  mutate(M = al %*% c(X2, X3)) |>
  mutate(pop = ifelse(M > m,"p1","p2")) |>
  pull()
pop1 <- factor(pop1)

pop2 <- grupo2 |>
  rowwise() |>
  mutate(M = al %*% c(X2, X3)) |>
  mutate(pop = ifelse(M > m,"p1","p2")) |>
  pull()
pop2 <- factor(pop2)

#table(pop1)
#table(pop2)

# Então, a matriz de confusão será:
mc <- t(matrix(c(27,3,
                 8,37),2,2, dimnames = list(c("pop1", "pop2"), c("pop1", "pop2"))))

#11/sum(mc) # APER

# Fazendo por funções prontas:

lda <- lda(X1~X2+X3, data = dados,prior =c(.5,.5))

gldap <- predict(lda)
glctable <- table(dados$X1, gldap$class)
prop <- (diag(prop.table(glctable,1))) # prop de classif. correta no grupo
propt <- (sum(diag(prop.table(glctable)))) # prop total de classf. correta

# Validação hold-out
#table(dados$X1)
set.seed(M)
split <- sample.split(dados$X1, SplitRatio = 0.3)
train <- subset(dados, split==T)
test <- subset(dados, split==F)

lda1 <- lda(X1~X2+X3, data = train,prior =c(.5,.5))

PT <- predict(lda1, newdata = test, type = "response")
glctable <- table(test$X1, PT$x >= 0.5)
```

Matriz de confusão:

FALSE	TRUE
20	1
12	19

Proporção de classificações corretas em cada grupo:

x
0.9523810
0.6129032

Proporção total de classificação correta: 0.75

Com isso, temos que a taxa de erro do modelo pontual é de 25%. Esta é relativamente maior do que a encontrada pelos outros métodos de validação utilizados até agora.

c)

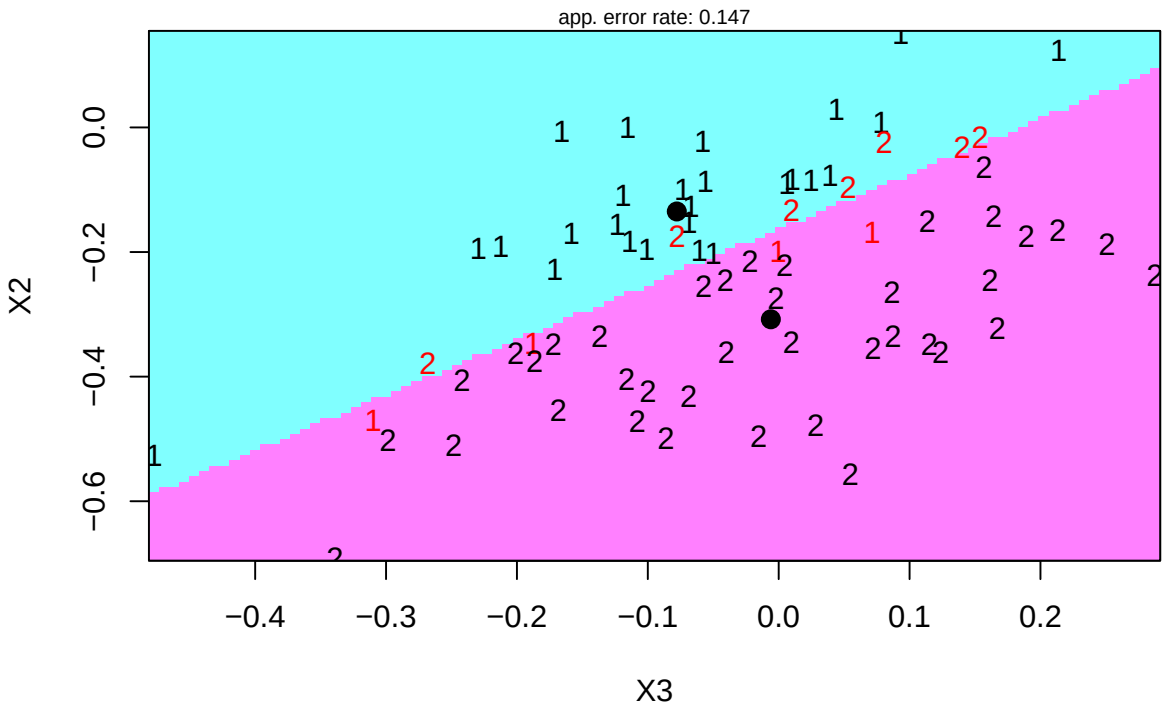
```
pop3 <- df |>
  rowwise() |>
  mutate(M = al %*% c(X2, X3)) |>
  mutate(pop = ifelse(M > m,"p1","p2")) |>
  pull()
pop3 <- factor(pop3)
kable(table(pop3))
```

pop3	Freq
p1	10

Todas as 10 novas observações foram classificadas como percentence à população π_1

d)

```
LDA <- lda(X1~., data = dados,prior=c(.75,.25))

LDAp1 <- predict(LDA)
LDAtable1 <- table(dados$X1, LDAp1$class)
prop <- (diag(prop.table(LDAtable1,1))) # prop de classif. correta no grupo
propt <- (sum(diag(prop.table(LDAtable1)))) # prop total de classf. correta

partimat(X1~X2+X3, data=dados, method="lda",
         plot.matrix = F, imageplot = T,prec=100)
```




```
# Validação hold-out
#table(dados$X1)
set.seed(M)
split <- sample.split(dados$X1, SplitRatio = 0.3)
train <- subset(dados, split==T)
test <- subset(dados, split==F)

lda1 <- lda(X1~X2+X3, data = train,prior =c(.75,.25))

PT <- predict(lda1, newdata = test, type = "response")
glctable <- table(test$X1, PT$x >= 0.5)


pred <- LDA |>
  predict(df)
```

Matriz de confusão:

FALSE	TRUE
19	2
5	26

Proporção de classificações corretas em cada grupo:

x
0.9047619
0.8387097

Proporção total de classificação correta: 0.8653846

Com isso, temos que a taxa de erro do modelo pontual é de 13.4615385%. Percebemos que a taxa de erro caiu consideravelmente ao escolher esta outra priori.

Além disso, todas as 10 novas observações foram novamente classificadas como pertencente à população π_1. Este é um resultado que não impressiona, visto que já haviam sido classificados assim com a priori igual, então era de se esperar que confirmasse este resultado com uma priori maior para a população 1.