# Lista 4 - MAE0330

*Guilherme NºUSP: 8943160 e Leonardo NºUSP: 9793436*

## Exercício 1

Sejam as densidades

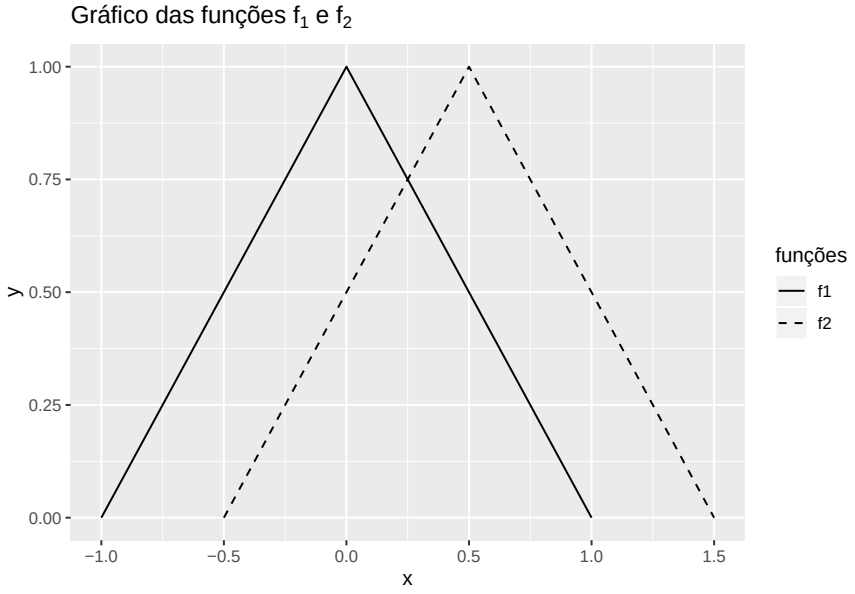
$$f_1(x) = (1 - \mid x \mid),\ \mid x \mid \leq 1$$

e

$$f_2(x) = (1 - \mid x - 0.5 \mid),\ -0.5 \leq x \leq 1.5$$

(a) Faça o gráfico das densidades.

**Resolução**




(b) Obtenha as regiões de classificação quando $p_1 = p_2$ e $c(1|2) = c(2|1)$.

**Resolução**

Como temos $p_1 = p_2$ e $c(1|2) = c(2|1)$ então as regiões de classificação ficam:

$$R_1 = \frac{f_1(x)}{f_2(x)} \geq 1$$

e

$$R_2 = \frac{f_1(x)}{f_2(x)} < 1$$

Para a região $R_1$, temos:

$$R_1 = \frac{1-|x|}{1-|x-0.5|} \geq 1 \Rightarrow \frac{1-|x|-1+|x-0.5|}{1-|x-0.5|} \geq 0$$

Domínio de x em $\frac{f_1(x)}{f_2(x)}$ : $\{x \in \mathbb{R} : -0.5 < x \leq 1\}$

Para resolver a inequação precisamos remover o módulo e dividir as regiões em cada caso, para $-0.5 < x \leq 0$:

$$\frac{x-(x-0.5)}{1+(x-0.5)} \geq 0 \Rightarrow \frac{0.5}{x+0.5} \geq 0 \Rightarrow (-0.5; 0]$$

Para o caso $0 < x \leq 0.5$:

$$\frac{-x-(x-0.5)}{1+(x-0.5)} \geq 0 \Rightarrow \frac{-2x+0.5}{x+0.5} \geq 0 \Rightarrow -2x+0.5 \geq 0 \Rightarrow [0; 0.25]$$

Para o caso $0.5 < x \leq 1$:

$$\frac{-x-x-0.5}{1-x+0.5} \geq 0 \Rightarrow \frac{-0.5}{1.5-x} \geq 0 \Rightarrow x > 1.5 \ (fora\ do\ dominio)$$

Fazendo a união dos intervalos dentro do domínio, temos que a região é dada por $R_1 : -0.5 < x \leq 0.25$

Para a região $R_2$, pelo fato de que temos apenas duas populações, basta fazer o conjunto completar da região $R_1$, ou seja, $R_2 : 0.25 < x < 1$

(c) Obtenha as regiões de classificação quando $p_1 = 0.2$ e $c(1|2) = c(2|1)$.

**Resolução**

Como temos $p_1 = 0.2 \Rightarrow p_2 = 0.8$ e $c(1|2) = c(2|1)$ então as regioões de classificação ficam:

$$R_1 = \frac{f_1(x)}{f_2(x)} \geq \frac{p_2}{p_1} \Rightarrow \frac{f_1(x)}{f_2(x)} \geq \frac{0.8}{0.2} \Rightarrow \frac{f_1(x)}{f_2(x)} \geq 4$$

e

$$R_2 = \frac{f_1(x)}{f_2(x)} < 4$$

Para a região $R_1$, temos:

$$R_1 = \frac{1-|x|}{1-|x-0.5|} \geq 4 \Rightarrow \frac{1-|x|-4+4|x-0.5|}{1-|x-0.5|} \geq 0$$

Domínio de x em $\frac{f_1(x)}{f_2(x)}$ : $\{x \in \mathbb{R} : -0.5 < x \leq 1\}$

Para resolver a inequação precisamos remover o módulo e dividir as regiões em cada caso, para $-0.5 < x \leq 0$:

$$\frac{-4(x-0.5)-3+x}{1+(x-0.5)} \geq 0 \Rightarrow \frac{-3x-1}{x+0.5} \geq 0 \Rightarrow -3x-1 \geq 0 \Rightarrow \left[-0.5; -\frac{1}{3}\right]$$

Para o caso $0 < x \leq 0.5$:

$$\frac{-4(x-0.5)-x-3}{1+x-0.5} \geq 0 \Rightarrow \frac{-5x-1}{0.5+x} \geq 0 \Rightarrow x < -0.2 \ (fora\ do\ dominio)$$

Para o caso $0.5 \leq x \leq 1$:

$$\frac{4(x-0.5)-x-3}{1-x+0.5} \geq 0 \Rightarrow \frac{3x-5}{1.5-x} \geq 0 \Rightarrow x \geq \frac{5}{3} \ (fora\ do\ dominio)$$

Fazendo a união dos intervalos dentro do domínio, temos que a região é dada por $R_1 : \left\{ -\frac{1}{2} < x \leq -\frac{1}{3} \right\}$

Para a região $R_2$, pelo fato de que temos apenas duas populações, basta fazer o conjunto completar da região $R_1$, ou seja, $R_2 : \left\{ -\frac{1}{3} < x \leq 1 \right\}$

# Exercício 2

Considere três populações bivariadas com a mesma matriz de covariância, e médias dadas por: $\mu_1^T = (0; 0)$, $\mu_2^T = (0; -1)$ e $\mu_3^T = (1; 0)$. A matriz de covariância comum é:

$$\Sigma = \begin{bmatrix} 5 & -2 \\ -2 & 1 \end{bmatrix}$$

(a) Obtenha as regiões de classificação das observações pelo método de Fisher.

**Resolução**

Primeiramente obtemos os autovetores e autovalores da matriz $\Sigma^{-1}B$:

```
## eigen() decomposition
## $values
## [1] 5.27008323 0.06325011
##
## $vectors
##            [,1]       [,2]
## [1,] -0.3898619 -0.7953761
## [2,] -0.9208733  0.6061163
```

E agora, obtemos os autovalores e autovetores de $\Sigma^{-\frac{1}{2}} B \Sigma^{-\frac{1}{2}}$ a fim de obter autovetores com a padronização $v_l^T \Sigma v_l = 1$ e $v_l^T \Sigma v_k = 0 \ \forall \ k \neq l$:

```
## eigen() decomposition
## $values
## [1] 5.27008323 0.06325011
##
## $vectors
##           [,1]       [,2]
## [1,] 0.4241554 -0.9055894
## [2,] 0.9055894  0.4241554
```

Por fim, obtemos os resultados para as funções discriminantes, seja $v_1, v_2$ autovetores de $\Sigma^{-\frac{1}{2}} B \Sigma^{-\frac{1}{2}}$:

$$\Sigma^{-1/2} v_1 = \begin{pmatrix} 0.7071 & 0.7071 \\ 0.7071 & 2.121 \end{pmatrix} (0.4241, 0.9056) = \begin{pmatrix} 0.940 \\ 2.221 \end{pmatrix}$$

E

$$\Sigma^{-1/2}v_2 = \begin{pmatrix} 0.7071 & 0.7071 \\ 0.7071 & 2.121 \end{pmatrix}(-0.905, 0.4241) = \begin{pmatrix} -0.340 \\ 0.259 \end{pmatrix}$$

Dessa forma temos as funções discriminantes $y_j = (\Sigma^{-1/2}v_j)^T x_0$:

$$y_1 = 0.940 * x_1 + 2.221 * x_2$$

$$y_2 = -0.340 * x_1 + 0.259 * x_2$$

Em que $x_0 = (x_1, x_2)$, assim com essas funções é possível classificar novas observações.

(b) Suponha que as probabilidades a priori das três populações são 1/2, 1/3 e 1/6, respectivamente. Obtenha as funções discriminantes pelo método geral. Faça suposições necessárias.

**Resolução**

Supondo normalidade dos dados, custos iguais e matriz de covariâncias iguais ($\Sigma_1 = \Sigma_2 = \Sigma_3 = \Sigma$):

$$d_1 = \mu_1^T * \Sigma^{-1}x_0 - \frac{1}{2} * \mu_1^T * \Sigma^{-1} * \mu_1 + ln(p_1)$$

$$= \begin{pmatrix} 0 \\ 0 \end{pmatrix}^T \begin{pmatrix} 5 & -2 \\ -2 & 1 \end{pmatrix}^{-1} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \frac{1}{2} \begin{pmatrix} 0 \\ 0 \end{pmatrix}^T \begin{pmatrix} 5 & -2 \\ -2 & 1 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ 0 \end{pmatrix} + ln(\frac{1}{2}) = ln(\frac{1}{2}) = -0.693$$

$$d_2 = \mu_2^T * \Sigma^{-1}x_0 - \frac{1}{2} * \mu_2^T * \Sigma^{-1} * \mu_2 + ln(p_2) = -2x_1 - 5x_2 - 2.5 + ln(\frac{1}{3}) = -2x_1 - 5x_2 - 3.599$$

$$d_3 = \mu_3^T * \Sigma^{-1}x_0 - \frac{1}{2} * \mu_3^T * \Sigma^{-1} * \mu_3 + ln(p_3) = x_1 + 2x_2 - 0.5 + ln(\frac{1}{6}) = x_1 + 2x_2 - 2.291$$

Em que o vetor $x_0^T = (x_1, x_2)$

Ao obter os respectivos $d_1, d_2$ e $d_3$, $x_0$ é alocado em $\pi_k$ se $d_i$ é máximo com $i = 1, 2, 3$.

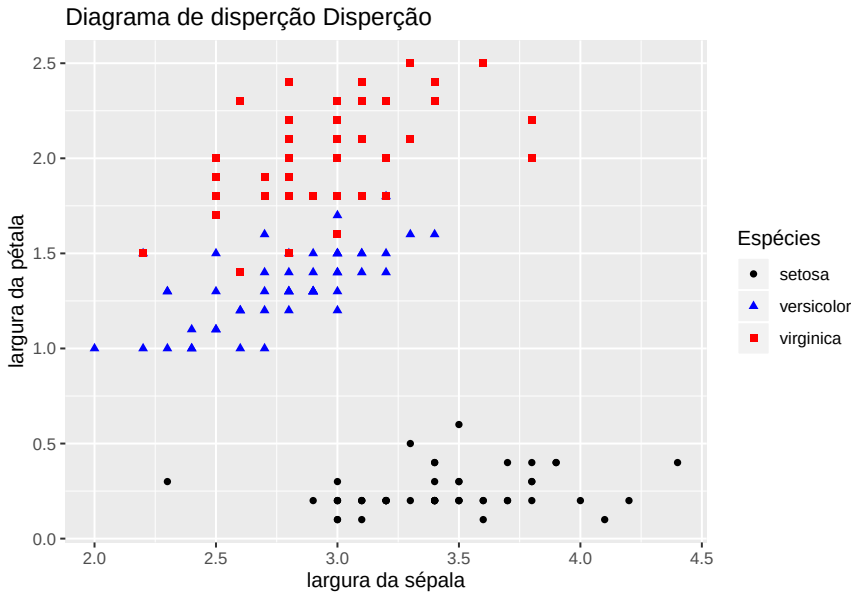
# Exercício 3

Os dados no arquivo T11-5.DAT são bastante conhecidos e referem-se a medidas da pétala e sépala de amostras de três diferentes espécies de flores do gênero iris (iris setosa, iris versicolor e iris virginica). O arquivo contém 5 colunas: comprimento da sépala ($X_1$ ), largura da sépala ($X_2$), comprimento da pétala ($X_3$), largura da pétala ($X_4$) e, por fim, a espécie da flor (1 - setosa; 2 - versicolor; 3 - virginica). Considere as variáveis $X_2$ : largura da sépala e $X_4$ : largura da pétala.

(a) Faça o diagrama de dispersão de $X_2$ e $X_4$ , diferenciando as observações das três espécies diferentes.

**Resolução**

Segue o diagrama de disperção das variáveis $X_2$ e $X_4$ diferenciando as observações das três espécies diferentes:




Onde podemos notar uma boa separação entre as espécies.

(b) Assumindo que as distribuição de $X_2$ e $X_4$ seja normal bivariada, obtenha os escores discrimantes quadráticos com $p_1 = p_2 = p_3$ . Classifique uma nova observação $x_0^T = (3,5; 1,75)$ em uma das três populações.

**Resolução**

Os Escores discriminates quadráticos são dados por:

$$\hat{d_1} = -\frac{1}{2} ln(|S_1|) - \frac{1}{2}(x_0 - \bar{x_1})^T S_1^{-1}(x_0 - \bar{x_1}) + ln(p_1)$$

$$= -\frac{1}{2} ln\left( \begin{vmatrix} 0.1437 & 0.0093 \\ 0.0093 & 0.0111 \end{vmatrix} \right) - \frac{1}{2} \begin{pmatrix} x_1 - 3.428 \\ x_2 - 0.246 \end{pmatrix}^T \begin{pmatrix} 0.1437 & 0.0093 \\ 0.0093 & 0.0111 \end{pmatrix}^{-1} \begin{pmatrix} x_1 - 3.428 \\ x_2 - 0.246 \end{pmatrix}$$

$$= -3.679x_1^2 + 23.708x_1 - 47.598x_2^2 + 6.16x_1x_2 + 2.301x_2 - 38.77$$

$$\hat{d_2} = -\frac{1}{2} ln(|S_2|) - \frac{1}{2}(x_0 - \bar{x_2})^T S_2^{-1}(x_0 - \bar{x_2}) + ln(p_2)$$

$$= -9.08x_1^2 + 24.937x_1 - 22.868x_2^2 + 19.138x_1x_2 + 7.634x_2 - 37.623$$

$$\hat{d_3} = -\frac{1}{2} ln(|S_3|) - \frac{1}{2}(x_0 - \bar{x_3})^T S_3^{-1}(x_0 - \bar{x_3}) + ln(p_3)$$

$$= -6.765x_1^2 + 22.936x_1 - 9.325x_2^2 + 8.54x_1x_2 + 12.387x_2 - 45.158$$

Em que o vetor $x_0^T = (x_1, x_2)$

Ajustando modelo via pacote MASS, temos:

```
## Call:
## qda(Species ~ Sepal.Width + Petal.Width, data = iris, prior = c(1,
##     1, 1)/3)
##
## Prior probabilities of groups:
##     setosa versicolor  virginica
##  0.3333333  0.3333333  0.3333333
##
## Group means:
##            Sepal.Width Petal.Width
## setosa           3.428       0.246
## versicolor       2.770       1.326
## virginica        2.974       2.026
```

Antes de classificarmos a nova observação, classificaremos nossa própria amostra para avaliar se o modelo está acurado:

Table 1: Matriz de confusão

| | setosa | versicolor | virginica |
|---|---|---|---|
| setosa | 50 | 0 | 0 |
| versicolor | 0 | 46 | 4 |
| virginica | 0 | 3 | 47 |

É possível notar que existem 7 classificações erradas. Classificando a observação $x_0^T = (3,5; 1,75)$:

```
## $setosa
## [1] -104.8719
##
## $versicolor
## [1] -1.055387
##
## $virginica
## [1] -2.325402
```

Pelos escores discrimantes quadráticos obtidos acima, nota-se que a população classificada para a observação é a *versicolor*, pois tem o maior dos escores.

(c) Assumindo que as distribuição de $X_2$ e $X_4$ seja normal bivariada com a mesma matriz de covariância, calcule os escores discriminantes lineares com $p_1 = p_2 = p_3$ . Classifique uma nova observação $x_0^T = (3,5; 1,75)$ em uma das três populações. Compare os resultados com o item anterior. Qual dos métodos de classificação você escolheria? Justifique.

**Resolução**

Ajustando o modelo:

```
## Call:
## lda(Species ~ Sepal.Width + Petal.Width, data = iris, prior = c(1,
##     1, 1)/3)
##
## Prior probabilities of groups:
##     setosa versicolor  virginica
##  0.3333333  0.3333333  0.3333333
##
## Group means:
##            Sepal.Width Petal.Width
## setosa           3.428       0.246
## versicolor       2.770       1.326
## virginica        2.974       2.026
##
## Coefficients of linear discriminants:
##                   LD1       LD2
## Sepal.Width -1.986964 2.6800746
## Petal.Width  5.477136 0.8169648
##
## Proportion of trace:
##    LD1    LD2
## 0.9884 0.0116
```

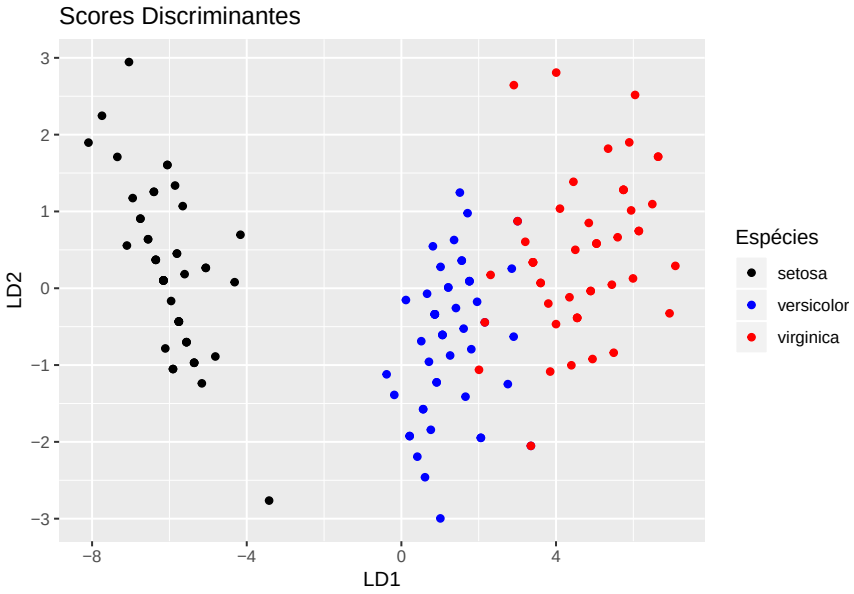
Antes de classificarmos a nova observação, classificaremos nossa própria amostra para avaliar se o modelo está acurado:

Table 2: Matriz de confusão

| | setosa | versicolor | virginica |
|---|---|---|---|
| setosa | 50 | 0 | 0 |
| versicolor | 0 | 49 | 1 |
| virginica | 0 | 4 | 46 |

Observa-se que o modelo errou 5 observações apenas, com esta informação e a tabela obtida no item anterior, o melhor método seria os escores discriminantes lineares pois há simplicidade, melhor intrepretação e menos erros de classificação.

É possível realizar um gráfico de dispersão para as duas funções discriminates, diferenciando as observações das três espécies diferentes:

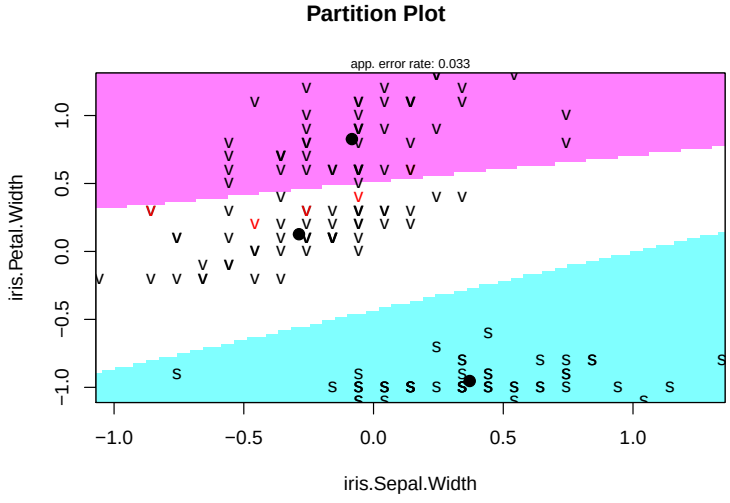



Por fim, é predito à qual população a observação $x_0^T = (3,5; 1,75)$ pertence:

```
## $class
## [1] versicolor
## Levels: setosa versicolor virginica
##
## $posterior
##         setosa versicolor virginica
## 1 3.209389e-14  0.7187594 0.2812406
##
## $x
##        LD1      LD2
## 1 2.136514 1.636255
```

Nota-se que a classificação não foi alterada do item anterior, logo a observação é classificada como versicolor.

Por fim, é possível realizar as regiões de classificação das espécies:



Em que, a região rosa representa a espécie virginica, a branca representa a espécie versicolor e por fim, de azul, a espécie setosa.

# Exercício 4

Gere dois grupos de 100 observações de uma distribuição normal bivariada com mesma matriz de covariância $\Sigma$ e vetores de médias $\mu_j$ , j = 1, 2, diferentes para os grupos (indique a semente adotada na simulação).

(a) Divida os dados em uma subamostra de "treinamento/estimação" (cerca de 70% dos dados) e uma de validação/predição (o restante das observações).

### Resolução

Fixando a semente 5678, segue o código que gera os dados e particiona os mesmos em treino (70%) e teste(30%):

```
library(caret)
library(MASS)
set.seed(5678)

sigma <- matrix(c(20,5,5,10),2,2)
mu1 <- c(5,5)
mu2 <- c(0,0)
n <- 100
sim_bnorm1 <- mvrnorm(n, mu1, sigma)
sim_bnorm2 <- mvrnorm(n, mu2, sigma)

y <- c(rep("1",100),rep("2",100))

df <- data.frame(rbind(sim_bnorm1,sim_bnorm2),y)

# item a

sample <- createDataPartition(y=df$y, p=0.7, list=FALSE)
train <- df[sample, ]
test <- df[-sample, ]
```

(b) Obtenha a função discriminante linear de Fisher e o critério de classificação. Faça as suposições necessárias. Obtenha a matriz de classificação usando os dados de validação. Comente.

### Resolução

Supondo que temos matriz de covariâncias iguais ($\Sigma_1 = \Sigma_2 = \Sigma$) entre as populações $\pi_1$ e $\pi_2$, além disso, tomando $X = (X_1, X_2)^T$ com:

$$\begin{cases} X_1 \sim N_2\left(\begin{pmatrix}5\\5\end{pmatrix}, \begin{pmatrix}20 & 5\\5 & 10\end{pmatrix}\right) \\ X_2 \sim N_2\left(\begin{pmatrix}0\\0\end{pmatrix}, \begin{pmatrix}20 & 5\\5 & 10\end{pmatrix}\right) \end{cases}$$

Associando $X_1$ a população $\pi_1$ e $X_2$ a população $\pi_2$, assim temos que:

```
## Call:
## lda(y ~ X1 + X2, data = train, prior = c(1/2, 1/2), CV = FALSE)
##
## Prior probabilities of groups:
##   1   2
## 0.5 0.5
##
## Group means:
##          X1        X2
## 1 4.9772844 4.8727213
## 2 0.3815526 0.1298852
##
## Coefficients of linear discriminants:
##            LD1
## X1 -0.06939064
## X2 -0.28762309
```

Como visto acima a função discriminante de Fisher é

$$Y = 0.07X_1 + 0.29X_2$$

E o critério de classificação é dado por

$$\hat{m} = \frac{1}{2}(\bar{x_1} - \bar{x_2})S_p^{-1}(\bar{x_1} + \bar{x_2}) = \frac{1}{2}\left[ \begin{pmatrix} 4.60 \\ 4.74 \end{pmatrix}^T \begin{pmatrix} 27.74 & 9.5 \\ 9.5 & 14.42 \end{pmatrix}^{-1} (4.60\ 4.74) \right] =$$

```
## [1] 0.893
```

Assim para cada observação nova $\hat{y}$ inseridos na função $Y$ se for maior ou igual a 0.893 classificamos o objeto na população $\pi_1$ e caso contrário classificamos o objeto na população $\pi_2$.

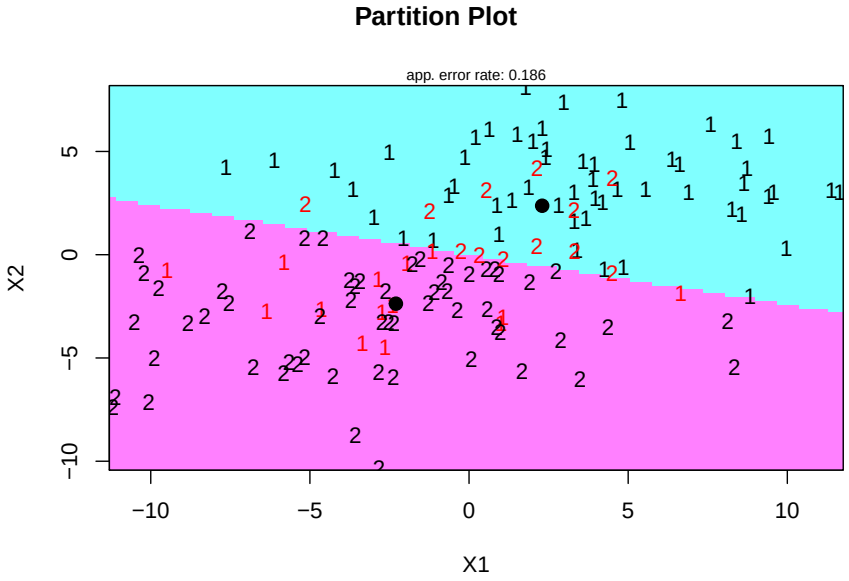
Fazendo a matriz de classificação com os dados de teste:

Table 3: Matriz de confusão

| | 1 | 2 |
|---|---|---|
| 1 | 28 | 2 |
| 2 | 5 | 25 |

Em que podemos notar que existe até uma boa classificação com uma acurácia de 88%, uma sensibilidade de 84% e por fim uma especificidade de 93%.

Constuindo o gráfico de classificação, temos:




O azul representa a região da população 1 e o rosa a população 2.

(c) Obtenha a função discriminante supondo $c(2|1) = 50$ e $c(1|2) = 100$, em que $c(i|j)$ é o custo de classificar uma observação em $\pi_i$ quando ela é na verdade de $\pi_j$ . Ainda, suponha que 20% da população total pertence a $\pi_1$. Compare os critérios de classificação com e sem essas informações.

**Resolução**

Considerando que $c(2|1) = 50$ e $c(1|2) = 100$, além disso que as probabilidades a prioris de pertecer a $\pi_1$ é $p_1 = 20\%$ e de pertecer a $\pi_2$ é $p_2 = 80\%$, assim a função discriminante é dada por:

$$Y = 0.07X_1 + 0.29X_2$$

Porém o critério de classificação fica modificado, sendo:

$$\hat{c} = \frac{1}{2}(\bar{x_1} - \bar{x_2})S_p^{-1}(\bar{x_1} + \bar{x_2}) + log\left(\frac{C(1|2)}{C(2|1)}\frac{p_2}{p_1}\right) =$$

$$\frac{1}{2}\left[\begin{pmatrix}4.60\\4.74\end{pmatrix}^T\begin{pmatrix}27.74 & 9.5\\9.5 & 14.42\end{pmatrix}^{-1}(4.60\ 4.74)\right] + log\left(\frac{100}{50}\frac{0.8}{0.2}\right) =$$

```
## [1] 2.973
```

```
## Call:
## lda(y ~ X1 + X2, data = train, prior = c(1, 8)/9, CV = FALSE)
##
## Prior probabilities of groups:
##         1         2
## 0.1111111 0.8888889
##
```

```
## Group means:
##          X1        X2
## 1 4.9772844 4.8727213
## 2 0.3815526 0.1298852
##
## Coefficients of linear discriminants:
##            LD1
## X1 -0.06939064
## X2 -0.28762309
```

Assim para cada observação nova $\hat{y}$ inseridos na função $Y$ se for maior ou igual a 2.973 classificamos o objeto na população $\pi_1$ e caso contrário classificamos o objeto na população $\pi_2$, porém fazendo uma comparação com o critério do exercício anterior onde não haviamos fixado custos e nem probabilidades a prioris, este favorece muito mais a classificação para a população $\pi_2$. Para fazer o ajuste do modelo com a função *lda* do pacote $MASS$, foi preciso fazer uma modificação na priori, pois tinhamos que adicionar os custos, e nos argumentos da função *lda* não tem entreda para custos, para isso: Das razão dos custos e da razão das prioris, temos:

$$\frac{100}{50}\frac{0.8}{0.2} = 8$$

Agora, basta encontar a razão $\frac{p}{1-p} = 8$, com $0 < p < 1$ logo:

$$\frac{p}{1-p} = 8 \Rightarrow p = 8 - 8p \Rightarrow 8 = 9p \Rightarrow p = \frac{8}{9} \; e \; 1 - p = \frac{1}{9}$$

Assim entrando com as probabilidades a priori como acima, estamos levando em consideração os cutos. Obtendo a seguinte matriz de classificação:

Table 4: Matriz de confusão

| | 1 | 2 |
|---|---|---|
| 1 | 18 | 12 |
| 2 | 1 | 29 |

Em que calculando métricas como acurácia, sensibilidade e especificidade, temos:

| Acurácia | Sensibilidade | Especificidade |
|---|---|---|
| 78% | 95% | 70% |

Logo podemos notar uma baixa acurácia em relação ao modelo do exercício anterior, embora tenhamos uma boa sensibilidade e boa especificidade, o modelo praticamente "chuta" todas as observação como sendo da população $\pi_2$, o que para um classificador não é bom.

d) Gere 20 novas observações da seguinte maneira:

- Associe essa nova observação a $\pi_1$ com probabilidade 0,2 e a $\pi_2$ com probabilidade 0,8;
- Se a nova observação pertence a $\pi_1$ , gere uma observação de uma normal bivariada com média $\mu_1$ e matriz de covariância $\Sigma$;
- Se a nova observação pertence a $\pi_2$ , gere valores de uma normal bivariada com média $\mu_2$ e matriz de covariância $\Sigma$.

Aplique as regras de classificação de (b) e (c) nessas novas observações. Compare as proporções de classificações erradas e corretas obtidas com as classificações de (b) e (c). Discuta os resultados.

**Resolução**

Gerando os dados conforme o exercício pede temos:

```
set.seed(5678)

n <- 20
p1 <- 0.2

y <- data.frame(y=rbinom(n,1,p1))
y[which(y[,1]==0),] <- "2"

sim_bnorm.d <- matrix(NA,20,2)

for (i in 1:n){
  if (y[i,]=="1"){
    sim_bnorm.d[i,] <- mvrnorm(1, mu1, sigma)
  }
  else{
    sim_bnorm.d[i,] <- mvrnorm(1, mu2, sigma)
  }
}

df1 <- data.frame(sim_bnorm.d,y)
```

Table 5: Matriz gerada

| X1 | X2 | y |
|---|---|---|
| 2.89 | 5.31 | 2 |
| 3.58 | 0.29 | 2 |
| 8.40 | 5.05 | 2 |
| -3.55 | -1.54 | 2 |
| -2.25 | 2.68 | 2 |
| -2.92 | -5.22 | 2 |
| 4.52 | -6.01 | 2 |
| -7.33 | -8.48 | 2 |
| 1.83 | -1.52 | 2 |
| -0.97 | 1.91 | 1 |
| 3.89 | -0.08 | 2 |
| 9.22 | 7.80 | 1 |
| 6.03 | 2.88 | 2 |
| 6.82 | -1.23 | 2 |
| 5.34 | 6.47 | 1 |
| -5.26 | 4.16 | 2 |
| 0.42 | 1.93 | 2 |
| 1.64 | -2.85 | 2 |
| 1.53 | 0.16 | 1 |
| -0.27 | -1.53 | 2 |

Fazendo as matrizes de confusão dos modelos ajustados nos item (b) e (c) respectivamente, temos:

Em que podemos notar como a função discriminate de Fisher não classificou bem os novos dados, enquanto

Table 6: Matriz de confusão item (b)

| | 1 | 2 |
|---|---|---|
| 1 | 2 | 2 |
| 2 | 3 | 13 |

Table 7: Matriz de confusão item (c)

| | 1 | 2 |
|---|---|---|
| 1 | 2 | 2 |
| 2 | 0 | 16 |

o modelo ajustado no item (c) ajustou melhor, que pode ser explicado pela forma pelas proporções amostrais que mostram que existe mais de uma população do que em outra e no modelo do item (c), isso foi levado em consideração, logo assim como esperado a classificação dele é melhor.

# Exercício 5

Considere os dados no arquivo primate.scapulae.txt utilizados na lista 2. Relembre que esses dados são referentes a medidas feitas na escápula de cinco diferentes gêneros de primatas Hominoidea (Hylobates, Pong, Pan, Gorilla e Homo). As medidas estão nas variáveis AD.BD, AD.CD, EA.CD, Dx.CD, SH.ACR, EAD, $\beta$ e $\gamma$. As cinco primeiras medidas são índices e as três últimas são ângulos. O ângulo $\gamma$ não está disponível para os primatas Homo e, portanto, não deve ser usado na análise (relembre que as medidas faltantes não estão representadas por NA nos dados). Com auxílio computacional, considerando apenas as 7 medidas das escápulas disponíveis, obtenha a melhor regra de classificação dentre as discutidas em sala. Utilize as taxas de classificação incorreta e correta para comparação entre os métodos. Discuta os resultados.

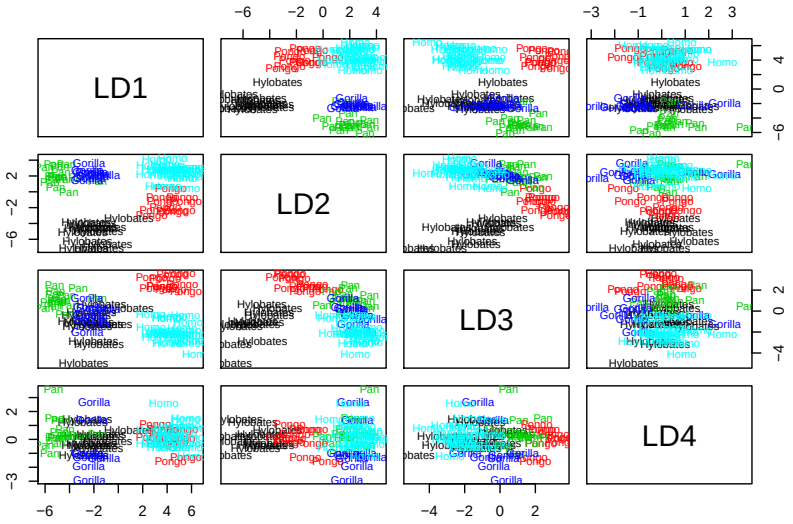
**Resolução**

Utilizaremos a técnica relativa ao exercício anterior, dividindo a base de dados em 70% de treino e 30% de teste com a mesma semente (5678). Em seguida, cria-se um modelo linear com as 7 variáveis, além disso supondo normalidade dos dados, matriz de covariâncias, custos e prioris iguais, temos:

```
## Call:
## lda(class ~ AD.BD + AD.CD + EA.CD + Dx.CD + SH.ACR + EAD + beta,
##     data = train, prior = c(1/5, 1/5, 1/5, 1/5, 1/5))
##
## Prior probabilities of groups:
##   Gorilla      Homo Hylobates       Pan     Pongo
##       0.2       0.2       0.2       0.2       0.2
##
## Group means:
##              AD.BD     AD.CD    EA.CD    Dx.CD   SH.ACR       EAD     beta
## Gorilla   68.94111  86.65556 65.72778 13.09111 59.74444  98.44444 30.88889
## Homo      42.70929  38.46071 77.46250 13.10143 67.41429 109.07143 47.21429
## Hylobates 60.31083 145.64167 55.72917 12.40917 64.62500 117.83333 15.75000
## Pan       75.69733  91.78000 57.93667 13.30667 54.56667 100.40000 24.13333
## Pongo     33.02300  34.88000 80.76000 12.34500 56.82000 123.00000 32.00000
##
## Coefficients of linear discriminants:
##                LD1         LD2         LD3          LD4
## AD.BD  -0.14354171  0.14943498  0.06317621 -0.198048886
## AD.CD   0.02248074 -0.04155105 -0.06229599 -0.028141243
## EA.CD   0.08086913  0.02036518  0.05518152 -0.272135514
## Dx.CD   0.19966631  0.20141020 -0.52806981  1.416233560
## SH.ACR  0.01757088 -0.02798669 -0.08553230 -0.002667064
```

```
## EAD     0.04731309 -0.05003809  0.02885531 -0.110587559
## beta    0.15051625  0.16116719 -0.20308632 -0.072830508
##
## Proportion of trace:
##    LD1    LD2    LD3    LD4
## 0.5219 0.3607 0.1108 0.0067
```

É possível verificar que as duas primeiras funções discriminantes são o suficiente para o modelo, possuindo uma proporção de 90.23%. A partir disso, é feito o gráfico de dispersão das funções discriminantes:




Como já discutido anteriormente, as duas primeiras funções já são bem suficientes para a análise. No seu respectivo gráfico, é de se notar que a espécie Hylobates está bem distante das demais espécies, as espécies Pongo e Homo se confundem um pouco, assim como as espécies Pan e Gorilla. Construindo a matriz de confusão:

Table 8: Matriz de confusão

| | Gorilla | Homo | Hylobates | Pan | Pongo |
|---|---|---|---|---|---|
| Gorilla | 4 | 0 | 0 | 1 | 0 |
| Homo | 0 | 12 | 0 | 0 | 0 |
| Hylobates | 0 | 0 | 4 | 0 | 0 |
| Pan | 0 | 0 | 0 | 5 | 0 |
| Pongo | 0 | 0 | 0 | 0 | 5 |

Para esta semente utilizada, há apenas um erro de classificação para o modelo. A seguir, utilizaremos a função quadrática que supõe normalidade dos dados, custos e prioris iguais:

```
## Call:
## qda(class ~ AD.BD + AD.CD + EA.CD + Dx.CD + SH.ACR + EAD + beta,
##     data = train, prior = c(1, 1, 1, 1, 1)/5)
##
## Prior probabilities of groups:
```

```
##   Gorilla      Homo Hylobates       Pan     Pongo
##       0.2       0.2       0.2       0.2       0.2
##
## Group means:
##               AD.BD     AD.CD    EA.CD    Dx.CD   SH.ACR       EAD     beta
## Gorilla   68.94111  86.65556 65.72778 13.09111 59.74444  98.44444 30.88889
## Homo      42.70929  38.46071 77.46250 13.10143 67.41429 109.07143 47.21429
## Hylobates 60.31083 145.64167 55.72917 12.40917 64.62500 117.83333 15.75000
## Pan       75.69733  91.78000 57.93667 13.30667 54.56667 100.40000 24.13333
## Pongo     33.02300  34.88000 80.76000 12.34500 56.82000 123.00000 32.00000
```

Obtemos então a matriz de confusão:

Table 9: Matriz de confusão

| | Gorilla | Homo | Hylobates | Pan | Pongo |
|---|---|---|---|---|---|
| Gorilla | 0 | 0 | 0 | 5 | 0 |
| Homo | 0 | 12 | 0 | 0 | 0 |
| Hylobates | 0 | 0 | 4 | 0 | 0 |
| Pan | 0 | 0 | 0 | 5 | 0 |
| Pongo | 0 | 0 | 0 | 0 | 5 |

Para esta semente, existem 5 erros de classificação observados.

Ao utilizar outras sementes, é possível notar que para a função linear a matriz de confusão não muda muito, já a quadrática possui mais alterações. Isso é um ponto negativo para a função quadrática, além da falta de interpretação das funções. Desse modo a função escolhida seria a linear.

# Exercício 6

Considere o arquivo de dados Carseats disponível no pacote ISLR no R. A descrição dos dados pode ser obtida digitando-se ?Carseats após o carregamento do pacote. Assuma que o interesse está em predizer vendas (Sales - variável contínua) usando árvore de regressão.

(a) Divida os dados em dados de treinamento e teste, deixando 70% das observações no banco de dados de treinamento.

### Resolução

Como não é possível a vizualização da divisão da base de dados, segue abaixo o código para realizar a divisão, fixando a mesma semente do exercício 4 (5678):

```r
library(ISLR)
library(caret)

data("Carseats")
attach(Carseats)

set.seed(5678)
indice_treino <- createDataPartition(y=Carseats$Sales,
```

```
                                    p=0.7, list=FALSE)
treino <- Carseats[indice_treino,]
teste <- Carseats[-indice_treino,]
```

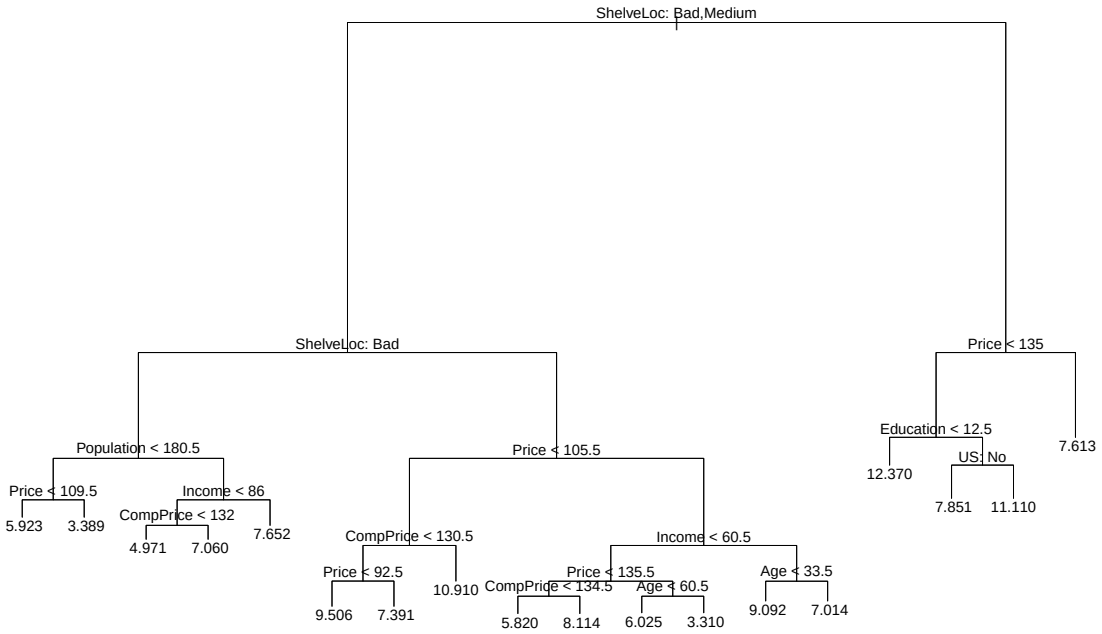
(b) Ajuste uma árvore de regressão nos dados de treinamento. Faça um gráfico da árvore e interprete.

**Resolução**

Ajustando o modelo de árvore de regressão, temos o seguinte resultado:

```
##
## Regression tree:
## tree(formula = Sales ~ ., data = treino)
## Variables actually used in tree construction:
## [1] "ShelveLoc"  "Population" "Price"      "Income"     "CompPrice"
## [6] "Age"        "Education"  "US"
## Number of terminal nodes:  18
## Residual mean deviance:  2.428 = 638.5 / 263
## Distribution of residuals:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -4.03300 -1.00200 -0.09091  0.00000  1.01900  4.01800
```

E o gráfico:




Nota-se que é possível montar 4 populações com essa árvore: Uma com locais de pratileiras boas, uma com locais de pratileiras ruins, uma com locais de pratileiras medianos e preços menores do que 105.5 e por fim uma com locais de pratileiras medianos e preços maiores do que 105.5.

(c) Obtenha as somas de quadrados dos erros de predição dos dados de treinamento e depois nos dados de teste.

**Resolução**

A soma de quadrados dos erros de predição dos dados de treinamento é:

$$\sum_{i=1}^{281}(\hat{y}_{treino} - y_{treino})^2 = 638.536$$

E para os dados de teste é:

$$\sum_{i=1}^{119}(\hat{y}_{teste} - y_{teste})^2 = 631.067$$

Em que podemos notar que a soma dos quadrados dos erros é praticamente equivalente, nos dando um indício que não houve *overfitting* com nosso modelo de árvore de regressão.

(d) Ajuste um modelo de regressão no banco de treinamento (o melhor que você encontrar para predição). Faça a predição dos valores para os dados de teste e compare com os resultados da árvore.

Observação: Códigos em R para obtenção de árvores de regressão podem ser encontrados no texto disponível no site o "Material de Apio" no e-disciplinas.

**Resolução**

Ajustando o modelo com todas as variáveis, certamento o modelo saturado não é o melhor, para isso executaremos o algotimos *stepwise* para a seleção de variáveis:

```
##
## Direction:  backward/forward
## Criterion:  BIC
##
## Start:  AIC=59.06
## Sales ~ CompPrice + Income + Advertising + Population + Price +
##     ShelveLoc + Age + Education + Urban + US
##
##               Df Sum of Sq     RSS    AIC
## - Urban        1      0.11  272.64  53.53
## - Population   1      0.21  272.73  53.63
## - Education    1      2.05  274.58  55.53
## - US           1      2.79  275.32  56.28
## <none>                      272.53  59.06
## - Income       1     58.27  330.80 107.87
## - Advertising  1    106.34  378.87 146.00
## - Age          1    149.52  422.05 176.32
## - CompPrice    1    372.59  645.12 295.56
## - ShelveLoc    2    734.54 1007.07 415.06
## - Price        1    861.30 1133.82 454.02
##
## Step:  AIC=53.53
## Sales ~ CompPrice + Income + Advertising + Population + Price +
```

```
##     ShelveLoc + Age + Education + US
##
##               Df Sum of Sq     RSS    AIC
## - Population   1      0.19  272.83  48.09
## - Education    1      2.06  274.69  50.00
## - US           1      2.75  275.39  50.71
## <none>                      272.64  53.53
## + Urban        1      0.11  272.53  59.06
## - Income       1     58.65  331.29 102.64
## - Advertising  1    106.95  379.58 140.89
## - Age          1    149.57  422.20 170.79
## - CompPrice    1    379.77  652.40 293.07
## - ShelveLoc    2    738.36 1010.99 410.52
## - Price        1    863.49 1136.13 448.95
##
## Step:  AIC=48.09
## Sales ~ CompPrice + Income + Advertising + Price + ShelveLoc +
##     Age + Education + US
##
##               Df Sum of Sq     RSS    AIC
## - Education    1      2.15  274.98  44.66
## - US           1      3.05  275.88  45.58
## <none>                      272.83  48.09
## + Population   1      0.19  272.64  53.53
## + Urban        1      0.09  272.73  53.63
## - Income       1     58.46  331.29  97.01
## - Advertising  1    119.78  392.61 144.73
## - Age          1    149.75  422.58 165.40
## - CompPrice    1    382.32  655.15 288.61
## - ShelveLoc    2    738.17 1011.00 404.88
## - Price        1    863.30 1136.13 443.31
##
## Step:  AIC=44.66
## Sales ~ CompPrice + Income + Advertising + Price + ShelveLoc +
##     Age + US
##
##               Df Sum of Sq     RSS    AIC
## - US           1      2.82  277.80  41.89
## <none>                      274.98  44.66
## + Education    1      2.15  272.83  48.09
## + Population   1      0.29  274.69  50.00
## + Urban        1      0.09  274.88  50.20
## - Income       1     59.24  334.21  93.84
## - Advertising  1    119.98  394.95 140.76
## - Age          1    148.23  423.20 160.18
## - CompPrice    1    380.77  655.74 283.23
## - ShelveLoc    2    747.43 1022.41 402.40
## - Price        1    872.85 1147.83 440.55
##
## Step:  AIC=41.89
## Sales ~ CompPrice + Income + Advertising + Price + ShelveLoc +
##     Age
##
##               Df Sum of Sq     RSS    AIC
```

```
## <none>                     277.80  41.89
## + US           1      2.82  274.98  44.66
## + Education    1      1.92  275.88  45.58
## + Population   1      0.61  277.19  46.91
## + Urban        1      0.05  277.75  47.48
## - Income       1     59.00  336.80  90.37
## - Age          1    148.46  426.26 156.56
## - Advertising  1    181.00  458.80 177.23
## - CompPrice    1    384.70  662.50 280.47
## - ShelveLoc    2    745.37 1023.17 396.97
## - Price        1    873.25 1151.05 435.70
```

Em que chegamos a o seguinte modelo final:

```
##
## Call:
## lm(formula = Sales ~ CompPrice + Income + Advertising + Price +
##     ShelveLoc + Age, data = treino)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.81451 -0.65416  0.01379  0.66918  2.90280
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       4.951866   0.600650   8.244 7.06e-15 ***
## CompPrice         0.095088   0.004890  19.444  < 2e-16 ***
## Income            0.016806   0.002207   7.615 4.34e-13 ***
## Advertising       0.122949   0.009219  13.337  < 2e-16 ***
## Price            -0.095235   0.003251 -29.294  < 2e-16 ***
## ShelveLocGood     4.878146   0.181082  26.939  < 2e-16 ***
## ShelveLocMedium   2.017831   0.149183  13.526  < 2e-16 ***
## Age              -0.045019   0.003727 -12.079  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.009 on 273 degrees of freedom
## Multiple R-squared:  0.8732, Adjusted R-squared:  0.8699
## F-statistic: 268.5 on 7 and 273 DF,  p-value: < 2.2e-16
```

Para avaliar se a árvore está prevendo bem utilizando o cálulo da somas de quadrados dos erros de predição, que no caso para a árvore é:

$$\sum_{i=1}^{119}(\hat{y}_{tree} - y_{teste})^2 = 631.067$$

E para a regressão múltipla temos:

$$\sum_{i=1}^{119}(\hat{y}_{reg} - y_{teste})^2 = 131.941$$

Em que podemos notar que o modelo de regressão múltipla é melhor, pois se tem uma soma de quadrados dos erros menor que o da árvore, que provavelmente sofreu o aumento devido ao *overfitting*.

# Exercício 7

Considere 51 objetos $O_1, O_2, ..., O_{51}$ organizados em uma linha reta, sendo que o j-ésimo objeto está localizado em um ponto com coordenada igual a j. Defina a medida de similaridade $s_{ij}$ entre os objetos $O_i$ e $O_j$ por:

$$s_{ij} = \begin{cases} 9, \ se \ i = j \\ 8, \ se \ 1 \leq |i-j| \leq 3 \\ 7, \ se \ 4 \leq |i-j| \leq 6 \\ \vdots \\ 1, \ se \ 22 \leq |i-j| \leq 24 \\ 0, \ se \ |i-j| \geq 25 \end{cases}$$

Converta as similaridades em dissimilaridades $\delta_{ij}$ pela transformação

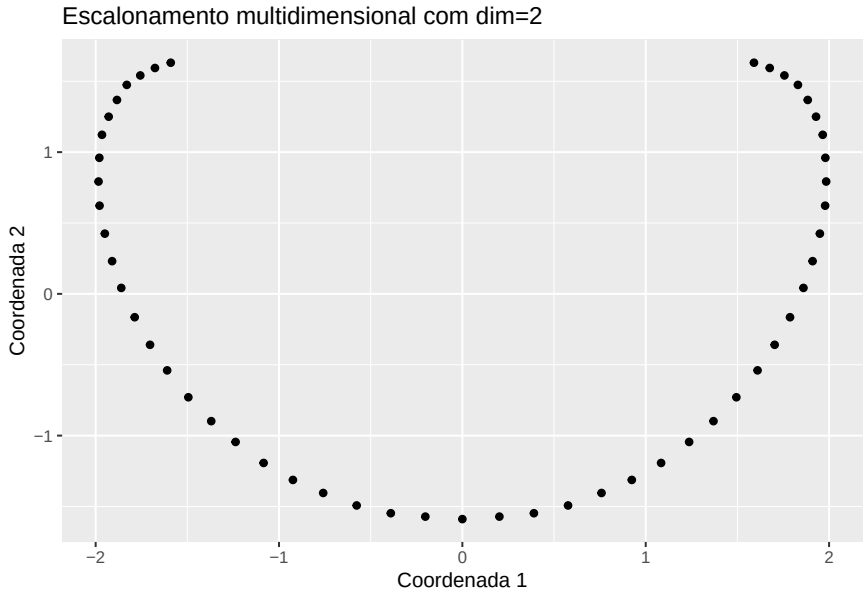
$$\sqrt{s_{ii} + s_{jj} - 2s_{ij}}$$

Utilize o método de escalonamento multidimensional clássico nesta matriz de dissimilaridade obtida. Faça o gráfico da solução obtida em duas dimensões e interprete o resultado.

**Resolução**

Utilizando o método de escalonamento multidimensional clássico na matriz de dissimilaridade $\delta$, temos:

```
##        Length Class  Mode
## points 102    -none- numeric
## eig     51    -none- numeric
## x        0    -none- NULL
## ac       1    -none- numeric
## GOF      2    -none- numeric
```




Em que podemos notar que o escalonamento multidimensional nos dá um formato de um arco.

# Exercício 8

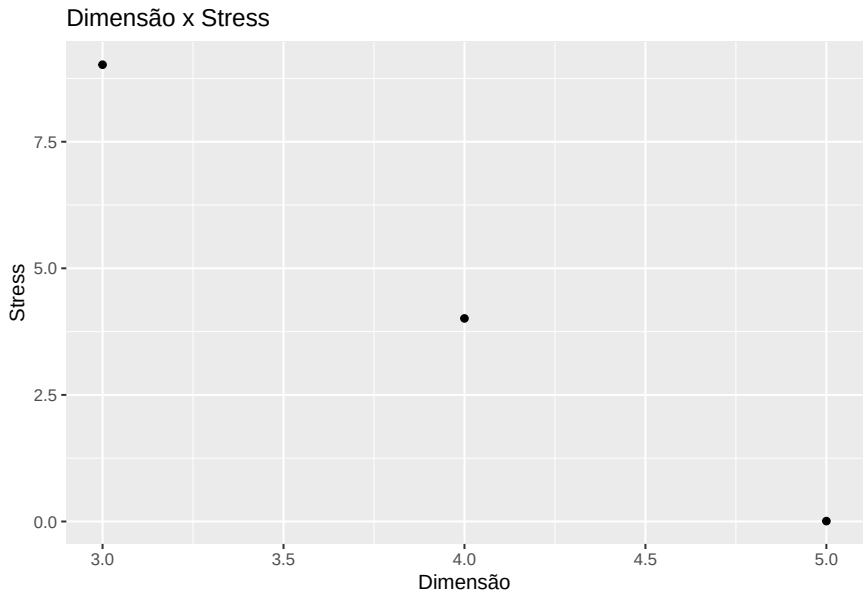
A tabela a seguir apresenta as distâncias entre sítios arqueológicos de diferentes períodos. As distâncias foram calculadas com base em frequências de diferentes tipos de cerâmicas encontradas nos sítios.

| Sítio | Sítio Arqueológico | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | P1980918 (1) | P1931131 (2) | P1550960 (3) | P1530987 (4) | P1361024 (5) | P1351005 (6) | P1340945 (7) | P1311137 (8) | P1301062 (9) |
| (1) | 0 | | | | | | | | |
| (2) | 2.202 | 0 | | | | | | | |
| (3) | 1.004 | 2.025 | 0 | | | | | | |
| (4) | 1.108 | 1.943 | 0.233 | 0 | | | | | |
| (5) | 1.122 | 1.870 | 0.719 | 0.541 | 0 | | | | |
| (6) | 0.914 | 2.070 | 0.719 | 0.679 | 0.539 | 0 | | | |
| (7) | 0.914 | 2.186 | 0.452 | 0.681 | 1.102 | 0.916 | 0 | | |
| (8) | 2.056 | 2.055 | 1.986 | 1.990 | 1.963 | 2.056 | 2.027 | 0 | |
| (9) | 1.608 | 1.722 | 1.358 | 1.168 | 0.681 | 1.005 | 1.719 | 1.991 | 0 |

(a) Dadas as distâncias, utilizando escalonamento multidimensional não-métrico, obtenha o *stress* para $q = 3, 4$ e 5 dimensões. Faça um gráfico do stress mínimo versus q. Discuta o número de dimensões que é necessário para uma boa representação dos dados.

**Resolução**

Utilizando o algoritmo de Kruskal-Shepard, obtermos o seguinte gráfico:




Em que podemos notar que a dimensão 5 é a que teve o menor *stress* (0.007%) que é "perfeito" (segundo uma tabela de qualidade dada em aula), se tratando que nossos dados tem dimensão 9, 5 é um bom número de dimensões para reduzir, porém ainda não é possível fazer algum tipo de vizualização, se o objetivo for vizualização, é mais recomendado usar a dimensão 2 ou 3. No caso é mais conveniente a dimensão 3 que tem *stress*=9% que é classificado como "bom".

(b) Obtenha as coordenadas dos pontos em duas dimensões e faça o gráfico.

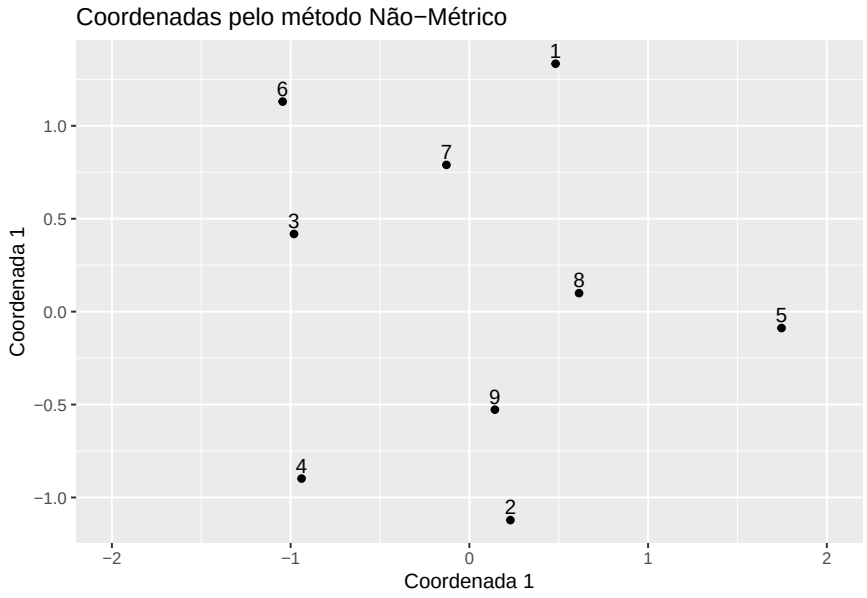
**Resolução**

As coordenadas dos pontos em duas dimensões é:

Table 10: Escalonamento multidimensional, método não métrico

| X1 | X2 |
|---:|---:|
| 0.4823335 | 1.3341391 |
| 0.2296609 | -1.1217133 |
| -0.9811661 | 0.4181129 |
| -0.9387605 | -0.8982619 |
| 1.7468288 | -0.0882723 |
| -1.0445662 | 1.1303209 |
| -0.1283079 | 0.7896873 |
| 0.6137201 | 0.0993932 |
| 0.1429663 | -0.5274815 |

E seu gráfico:




Em que é possível notar a formação de alguns grupos como os sítios 1,3,6 e 7 como um grupo, os ítios 2,4,8 e 9 como outro grupo e o sítio 5 como um grupo de apenas um elemento.

(c) Obtenha as coordenadas dos pontos em duas dimensões utilizando o escalonamento multidimensional clássico e faça o gráfico.

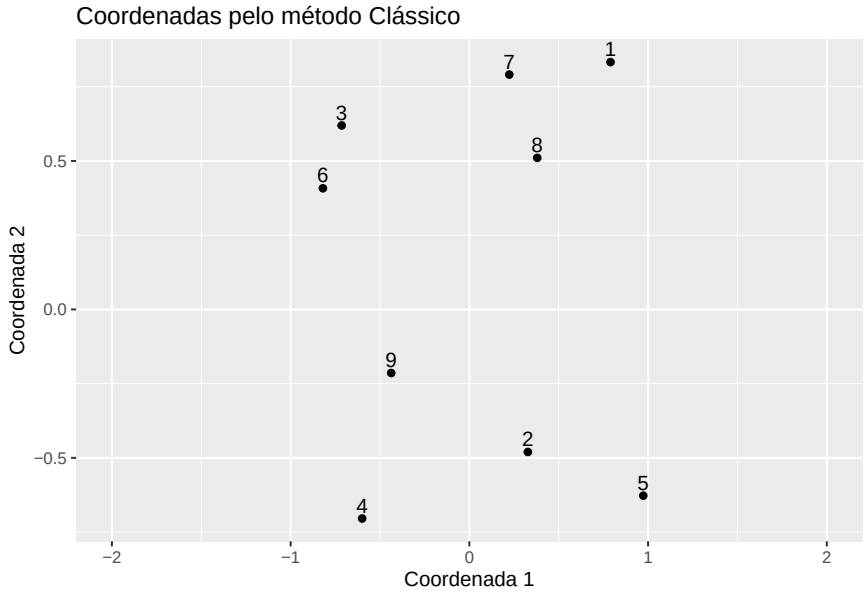
**Resolução**

As coordenadas dos pontos em duas dimensões pelo método clássico é:

Table 11: Escalonamento multidimensional, método clásico

| X1 | X2 |
|---|---|
| 0.7899265 | 0.8329469 |
| 0.3273636 | -0.4801290 |
| -0.7144570 | 0.6195736 |
| -0.5996989 | -0.7042560 |
| 0.9730717 | -0.6272868 |
| -0.8193115 | 0.4080191 |
| 0.2233671 | 0.7908823 |
| 0.3798227 | 0.5104016 |
| -0.4373754 | -0.2142272 |

E seu gráfico:




Da mesma forma que o exercício anterior é possível notar a formação de alguns grupos como os sítios 1,3,6,7 e 8 como um grupo, os ítios 2,4,5 e 9 como outro grupo, diferenciando do exercício anterior.

# Códigos

```
library(ggplot2)
library(MASS)
library(ISLR)
library(caret)
library(tree)
library(RcmdrMisc)
library(smacof)
library(expm)
```

```r
# Exercício 1

# item a

data <- data.frame(x=c(-1, -0.5, 0, 0.5, 1, 1.5),
                   y=c(0,0,1,1,0,0),
                   l=c("f1","f2","f1","f2","f1","f2"))

ggplot(data, aes(x = x,y=y,group=l)) +
  geom_line(aes(linetype=l))+
  scale_y_continuous(limits = c(0,1)) +
  scale_x_continuous(limits = c(-1,1.5)) +
  scale_linetype_manual(name = "funções", values = c("solid","dashed")) +
  labs(x = "x",
       y = "y",
       title = expression(paste("Gráfico das funções ",f[1]," e ",f[2])))

# Exercício 2

# item a

mu1 <- c(0,0); mu2 <- c(0,-1); mu3 <- c(1,0)
Sigma <- matrix(c(5,-2,-2,1),2,2)

m.g <- c(1/3,-1/3)
B <- (mu1-m.g)%*%t((mu1-m.g)) + (mu2-m.g)%*%t((mu2-m.g)) + (mu3-m.g)%*%t((mu3-m.g))

eigen(solve(Sigma)%*%B)

WBW <- eigen(sqrtm(solve(Sigma))%*%B%*%sqrtm(solve(Sigma)))
f1 <- WBW$vectors[,1]
f2 <- WBW$vectors[,2]
f1t <- sqrtm(solve(Sigma))%*%f1
f2t <- sqrtm(solve(Sigma))%*%f2

# Exercício 3

data("iris")
attach(iris)

# item a

ggplot(iris,aes(x=Sepal.Width, y=Petal.Width)) +
  geom_point(aes(shape=Species, color=Species)) +
  scale_color_manual(values=c('black','blue','red')) +
  labs(x="largura da sépala",
       y="largura da pétala",
       title="Diagrama de disperção Disperção",
       colour="Espécies",
       shape="Espécies")

# item b
```

```r
iris.qda <- qda(Species~Sepal.Width+Petal.Width, data = iris,
                prior=c(1,1,1)/3,CV=FALSE)
qda.pred <- predict(iris.qda)
table(iris$Species,qda.pred$class)

dados <- data.frame(Sepal.Width,Petal.Width,Species)

# Matrizes de covariâncias
cov.set <- cov(subset(x = dados,Species=="setosa")[-3])
cov.ver <- cov(subset(x = dados,Species=="versicolor")[-3])
cov.vir <- cov(subset(x = dados,Species=="virginica")[-3])

# Médias
aux1 <- tapply(dados[,1],Species,mean)
aux2 <- tapply(dados[,2],Species,mean)

m.set <- c(aux1[1],aux2[1])
m.ver <- c(aux1[2],aux2[2])
m.vir <- c(aux1[3],aux2[3])

# Predição
Scores <- function(x,mu,s,p=1/3){
  as.numeric(-1/2*log(det(s))-1/2*t(x-mu)%*%solve(s)%*%(x-mu) + log(p))
}

x0 <- c(3.5,1.75)
list("setosa"=Scores(x0,m.set,cov.set),"versicolor"=Scores(x0,m.ver,cov.ver),
     "virginica"=Scores(x0,m.vir,cov.vir))

# item c

iris.lda <- lda(Species~Sepal.Width+Petal.Width, data = iris,prior=c(1,1,1)/3)
iris.lda

lda.pred <- predict(iris.lda)
table(iris$Species,lda.pred$class)

iris1 <- data.frame(iris$Sepal.Width,iris$Petal.Width)
lda.data <- cbind(iris1, lda.pred)
ggplot(lda.data, aes(x.LD1, x.LD2)) +
  geom_point(aes(color = Species)) +
  scale_color_manual(values=c('black','blue','red')) +
  labs(x="LD1",
       y="LD2",
       title="Scores Discriminantes",
       colour="Espécies")

x0 <- data.frame(3.5,1.75)
names(x0) <- c("Sepal.Width","Petal.Width")
lda.predx0 <- predict(iris.lda,newdata=x0)
lda.predx0 # predição

# Gráfico de separação
```

```r
mu.k <- iris.lda$means
mu <- colMeans(mu.k)
dscores <- scale(iris1[,1:2], center=mu, scale=F)
partimat(x=dscores[,2:1], grouping=iris$Species, method="lda")

detach(iris)

# Exercício 4

set.seed(5678)

sigma <- matrix(c(20,5,5,10),2,2)
mu1 <- c(5,5)
mu2 <- c(0,0)
n <- 100
sim_bnorm1 <- mvrnorm(n, mu1, sigma)
sim_bnorm2 <- mvrnorm(n, mu2, sigma)

y <- c(rep("1",100),rep("2",100))

df <- data.frame(rbind(sim_bnorm1,sim_bnorm2),y)

# item a

sample <- createDataPartition(y=df$y, p=0.7, list=F)
train <- df[sample, ]
test <- df[-sample, ]

# item b

disc.linear1 <- lda(y~X1+X2, data = train, prior=c(1/2,1/2), CV=FALSE)
disc.linear1

lda.pred <- predict(disc.linear1,newdata=test)
table(test$y,lda.pred$class)

# gráfico de separação
mu.k <- disc.linear1$means
mu <- colMeans(mu.k)
dscores <- scale(train[,1:2], center=mu, scale=F)
partimat(x=dscores[,2:1], grouping=train$y, method="lda")

# item c

disc.linear2 <- lda(y~X1+X2, data = train, prior=c(1,8)/9, CV=FALSE)

lda.pred <- predict(disc.linear2,newdata=test)
table(test$y,lda.pred$class)

# item d

set.seed(5678)
```

```r
n <- 20
p1 <- 0.2

y <- data.frame(y=rbinom(n,1,p1))
y[which(y[,1]==0),] <- 2

sim_bnorm.d <- matrix(NA,n,2)

for (i in 1:n){
  if (y[i,]=="1"){
    sim_bnorm.d[i,] <- mvrnorm(1, mu1, sigma)
  }
  else{
    sim_bnorm.d[i,] <- mvrnorm(1, mu2, sigma)
  }
}

df1 <- data.frame(sim_bnorm.d,y)

lda.pred.d <- predict(disc.linear1,newdata=df1)
table(df1$y,lda.pred.d$class)

lda.pred.d2 <- predict(disc.linear2,newdata=df1)
table(df1$y,lda.pred.d2$class)

# Exercício 5

primate <- read.csv("primate.scapulae.txt", sep="")
attach(primate)
set.seed(5678)

sample <- createDataPartition(y=primate$classdigit, p=0.7,list=FALSE)
train <- primate[sample, ]
test <- primate[-sample, ]

model1 <- lda(class~AD.BD+AD.CD+EA.CD+Dx.CD+SH.ACR+EAD+beta, data = train,
              prior=c(1,1,1,1,1)/5)
model1

# classificações
plot(model1,col=as.integer(train$classdigit))

# matriz de confusão
lda.pred <- predict(model1,newdata=test)
table(test$class,lda.pred$class)

# Modelo quadrático

model2 <- qda(class~AD.BD+AD.CD+EA.CD+Dx.CD+SH.ACR+EAD+beta, data = train,
              prior=c(1,1,1,1,1)/5)
model2

# matriz de confusão
```

```r
qda.pred <- predict(model2,newdata=test)
table(test$class,qda.pred$class)

# Exercício 6

data("Carseats")
attach(Carseats)

# item a

set.seed(5678)
indice_treino <- createDataPartition(y=Carseats$Sales, p=0.7, list=FALSE)
treino <- Carseats[indice_treino,]
teste <- Carseats[-indice_treino,]

# item b

tree.reg <- tree(Sales ~ ., treino)
summary(tree.reg)

# gráfico da árvore
plot(tree.reg)
text(tree.reg, pretty = 0)

# valores de preditos e real (teste)
y.hat.teste <- predict(tree.reg, teste)
y.teste <- teste$Sales

# valores de preditos e real (treino)
y.hat.treino <- predict(tree.reg, treino)
y.treino <- treino$Sales

# soma de quadrado  dados teste
sum((y.hat.teste - y.teste)^2)

# soma de quadrado  dados teste
sum((y.hat.treino - y.treino)^2)

# item d

mod1 <- lm(Sales ~ ., treino)
step <- stepwise(mod1) # stepwise

mod.final <- lm(formula = Sales ~ CompPrice + Income + Advertising + Price +
                  ShelveLoc + Age, data = treino)
summary(mod.final)

# soma de quadrados arvore
sum((y.hat.teste - y.teste)^2)

# soma de quadrados regressão
pred.reg <- predict(mod.final,teste)
sum((pred.reg - y.teste)^2)
```

```r
# exercício 7

M <- matrix(NA,51,51)

# Matriz de similaridades
for (i in 1:51) {
  for (j in 1:51) {
    if (i==j) M[i,j]=9
    else if (abs(i-j) >=1 && abs(i-j) <=3) M[i,j]=8
    else if (abs(i-j) >=4 && abs(i-j) <=6) M[i,j]=7
    else if (abs(i-j) >=7 && abs(i-j) <=9) M[i,j]=6
    else if (abs(i-j) >=10 && abs(i-j) <=12) M[i,j]=5
    else if (abs(i-j) >=13 && abs(i-j) <=15) M[i,j]=4
    else if (abs(i-j) >=16 && abs(i-j) <=18) M[i,j]=3
    else if (abs(i-j) >=19 && abs(i-j) <=21) M[i,j]=2
    else if (abs(i-j) >=22 && abs(i-j) <=24) M[i,j]=1
    else M[i,j]=0
  }
}

Delta <- matrix(NA,51,51)

# Matriz de dissimilaridades
for (i in 1:51) {
  for (j in 1:51) {
    Delta[i,j] = sqrt(M[i,i]+M[j,j]-2*M[i,j])
  }
}

EM.m <- cmdscale(Delta,k=2, eig=TRUE)
summary(EM.m)

df <- data.frame(x=EM.m$points[,1],y=EM.m$points[,2])

ggplot(df,aes(x=x,y=y)) +
  geom_point() +
  labs(x='Coordenada 1',
       y='Coordenada 2',
       title='Escalonamento multidimensional com dim=2')

# Exercício 8

# criando uma matriz simétrica
D.aux <- c(0    ,
           2.202 , 0     ,
           1.004 , 2.025 , 0     ,
           1.108 , 1.943 , 0.233 , 0     ,
           1.122 , 1.870 , 0.719 , 0.541 , 0     ,
           0.914 , 2.070 , 0.719 , 0.679 , 0.539 , 0     ,
           0.914 , 2.186 , 0.452 , 0.681 , 1.102 , 0.916 , 0     ,
           2.056 , 2.055 , 1.986 , 1.990 , 1.963 , 2.056 , 2.027 , 0     ,
           1.608 , 1.722 , 1.358 , 1.168 , 0.681 , 1.005 , 1.719 , 1.991 , 0)
D <- matrix(0,9,9)
```

```r
D[upper.tri(D, diag=TRUE)] <- D.aux
D[lower.tri(D,diag=FALSE)] <- D[upper.tri(D,diag=FALSE)]

# item a

# Método Shepard-Kruskal
EM.nm_3 <- isoMDS(D, k=3, trace=FALSE)
EM.nm_4 <- isoMDS(D, k=4, trace=FALSE)
EM.nm_5 <- isoMDS(D, k=5, maxit = 60, trace=FALSE)
iso_stress <- data.frame(stress=c(EM.nm_3$stress,EM.nm_4$stress,EM.nm_5$stress),
                         dim=c(3,4,5))

ggplot(iso_stress , aes(x=dim,y=stress)) +
  geom_point() +
  labs(x="Dimensão",
       y="Stress",
       title="Dimensão x Stress")

# item b

EM.nm_2 <- isoMDS(D, k=2)
df1 <- data.frame(EM.nm_2$points)

ggplot(df1, aes(x=X1,y=X2)) +
  geom_point() +
  scale_x_continuous(limits = c(-2,2)) +
  geom_text(aes(label=as.character(seq(1:9)), hjust=0.5, vjust=-0.4)) +
  labs(x="Coordenada 1",
       y="Coordenada 2",
       title="Coordenadas pelo método Não-Métrico")

# item c

EM.C <- cmdscale(D, k=2, eig=TRUE)

df2 <- data.frame(EM.C$points)

ggplot(df2, aes(x=X1,y=X2)) +
  geom_point() +
  scale_x_continuous(limits = c(-2,2)) +
  geom_text(aes(label=as.character(seq(1:9)), hjust=0.5, vjust=-0.4)) +
  labs(x="Coordenada 1",
       y="Coordenada 2",
       title="Coordenadas pelo método Clássico")
```