



# CE319- Estatística Multivariada

## Listas de Exercícios

Luiz Henrique - 20213026

Mateus Souza - 20207154

Abril/2024

# 1.

Os dados apresentados no arquivo 'football.txt', disponível na página da disciplina, são informações de 335 atletas da liga profissional de futebol americano. As variáveis são as seguintes:

- Atleta - nome do atleta;
- Universidade - nome da universidade;
- Posição - posição em que joga;
- Pontuação - score;
- Altura - em polegadas;
- Envergadura - em polegadas;
- Peso - em libras;
- Comp.mao - comprimento da mão em polegadas;
- Tempo.40 - tempo para percorrer 40 jardas (segundos);
- Força - resultado do teste de força;
- Pulo.vert - distância vertical do salto;
- Pulo.hor - distância horizontal do salto;
- Corrida - tempo (em segundos);
- Corrida2 - tempo (em segundos).

Faça uma análise descritiva dos dados. Utilize gráficos, obtenha as medidas descritivas vistas em aula. Converta as variáveis para unidades de medida adotadas no Brasil.

```
#install.packages("ggplot2")
#install.packages("reshape2")
#install.packages("aplpack")
#install.packages("plotly")
#install.packages("ellipse")
#install.packages("lattice")
#install.packages("psych")
library(ggplot2)
library(reshape2)
library(aplpack)
library(plotly)
library(ellipse)
library(lattice)
library(psych)

#importação
df1 <- read.table(file = "UD02_Base_Football.txt", header = TRUE, encoding = "latin1")
head(df1)
```

##	Atleta	Universidade	Posição	Pontuação	Altura	Envergadura	Peso
## 1	AbbrederisJared	Wisconsin	WO	5.20	73	31.375	195
## 2	AdamsDavante	FresnoSt.	WO	5.96	73	32.625	212
## 3	AlexanderMo	UtahSt.	S	4.90	73	32.625	220
## 4	AllenRicardo	Purdue	DB	5.10	71	30.000	187
## 5	AmaroJace	TexasTech	TE	5.40	77	34.000	265
## 6	AndrewsAntonio	WesternKentucky	RB	5.16	70	31.250	225

```
##      Comp.mao Tempo.40 Força Pulo.vert Pulo.hor Corrida Corrida2
## 1      9.625      4.50      4      30.5      117      6.80      4.08
## 2      9.000      4.56     14      39.5      123      6.82      4.30
## 3      8.875      4.54     NA      38.0      123      7.05      4.51
## 4      9.250      4.61     13      35.5      117      NA       4.15
## 5      9.000      4.74     28      33.0      118      7.42      4.30
## 6      9.500      4.82     20      29.5      106      7.24      4.49
```

*#conversão de medidas*

```
df1$Altura <- df1$Altura * 2.54
df1$Envergadura <- df1$Envergadura * 2.54
df1$Peso <- df1$Peso * 0.453592
df1$Comp.mao <- df1$Comp.mao * 2.54
head(df1)
```

```
##      Atleta      Universidade Posição Pontuação Altura Envergadura
## 1 AbbrederisJared      Wisconsin      W0      5.20 185.42      79.6925
## 2 AdamsDavante      FresnoSt.      W0      5.96 185.42      82.8675
## 3 AlexanderMo      UtahSt.      S      4.90 185.42      82.8675
## 4 AllenRicardo      Purdue      DB      5.10 180.34      76.2000
## 5 AmaroJace      TexasTech      TE      5.40 195.58      86.3600
## 6 AndrewsAntonio WesternKentucky      RB      5.16 177.80      79.3750
##      Peso Comp.mao Tempo.40 Força Pulo.vert Pulo.hor Corrida Corrida2
## 1 88.45044 24.4475      4.50      4      30.5      117      6.80      4.08
## 2 96.16150 22.8600      4.56     14      39.5      123      6.82      4.30
## 3 99.79024 22.5425      4.54     NA      38.0      123      7.05      4.51
## 4 84.82170 23.4950      4.61     13      35.5      117      NA       4.15
## 5 120.20188 22.8600      4.74     28      33.0      118      7.42      4.30
## 6 102.05820 24.1300      4.82     20      29.5      106      7.24      4.49
```

*#descritiva*

```
summary(df1[, -c(1:3)])
```

```
##      Pontuação      Altura      Envergadura      Peso
## Min.      :4.500   Min.      :170.2   Min.      :72.39   Min.      : 73.94
## 1st Qu.:5.100   1st Qu.:182.6   1st Qu.:79.38   1st Qu.: 92.99
## Median :5.250   Median :188.0   Median :82.23   Median :105.23
## Mean      :5.323   Mean      :187.3   Mean      :82.07   Mean      :109.85
## 3rd Qu.:5.420   3rd Qu.:193.0   3rd Qu.:84.77   3rd Qu.:123.15
## Max.      :7.500   Max.      :203.2   Max.      :93.34   Max.      :159.66
##
##      Comp.mao      Tempo.40      Força      Pulo.vert      Pulo.hor
## Min.      :20.32   Min.      :4.260   Min.      : 4.00   Min.      :20.50   Min.      : 88
## 1st Qu.:23.50   1st Qu.:4.530   1st Qu.:15.00   1st Qu.:29.00   1st Qu.:109
## Median :24.45   Median :4.680   Median :20.00   Median :33.00   Median :116
## Mean      :24.33   Mean      :4.764   Mean      :20.28   Mean      :32.66   Mean      :115
## 3rd Qu.:25.40   3rd Qu.:4.950   3rd Qu.:25.00   3rd Qu.:36.00   3rd Qu.:122
## Max.      :28.26   Max.      :5.590   Max.      :42.00   Max.      :42.00   Max.      :134
##      NA's      :67      NA's      :90      NA's      :64      NA's      :74
##      Corrida      Corrida2
## Min.      :6.470   Min.      :3.810
## 1st Qu.:6.980   1st Qu.:4.190
## Median :7.165   Median :4.300
## Mean      :7.250   Mean      :4.334
## 3rd Qu.:7.455   3rd Qu.:4.470
## Max.      :8.290   Max.      :5.160
```

```
## NA's :113      NA's :148
```

```
#matriz de covariância
```

```
matriz_cov <- cov(df1[, -c(1:3)], use = "pairwise.complete.obs")
```

```
matriz_cov_long <- melt(matriz_cov)
```

```
ggplot(matriz_cov_long, aes(Var2, Var1, fill = value)) +
```

```
  geom_tile(color = "white") +
```

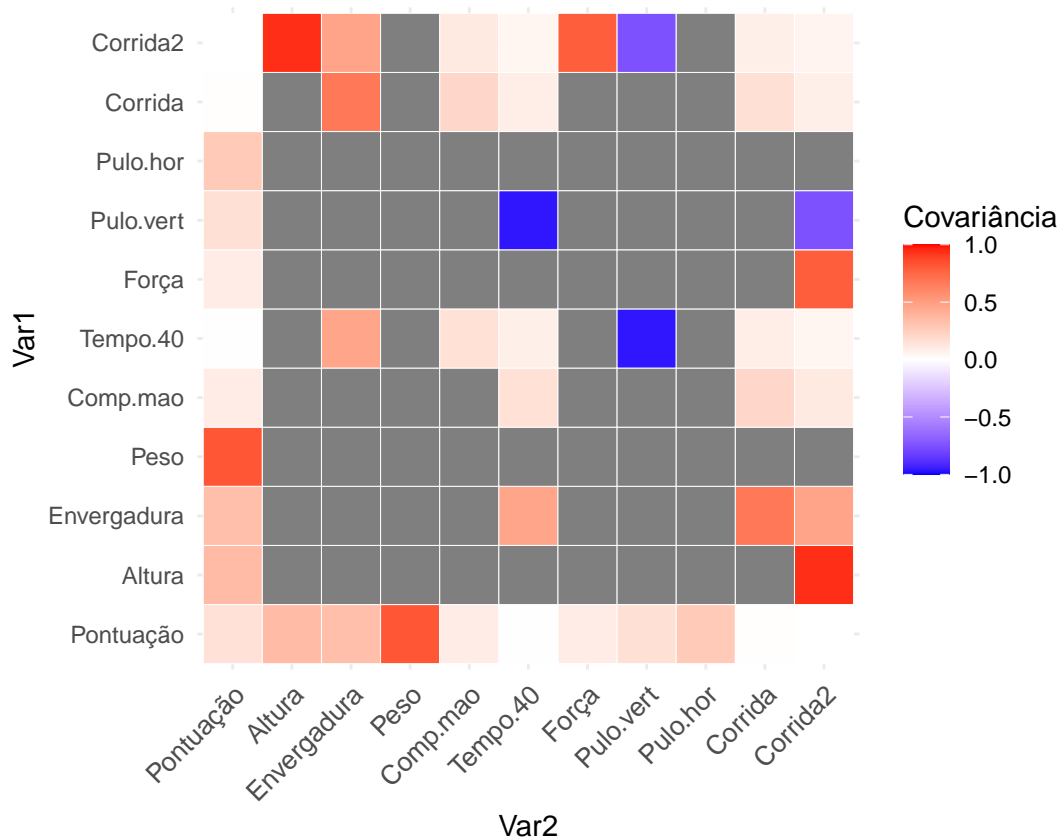
```
  scale_fill_gradient2(low = "blue", mid = "white", high = "red", midpoint = 0, limit = c(-1, 1), space = "srgb",
```

```
                      name = "Covariância") +
```

```
  theme_minimal() +
```

```
  theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 10, hjust = 1)) +
```

```
  coord_fixed()
```



```
#matriz de correlação
```

```
matriz_cor <- cor(df1[, -c(1:3)], use = "pairwise.complete.obs")
```

```
matriz_cor_long <- melt(matriz_cor)
```

```
ggplot(matriz_cor_long, aes(Var2, Var1, fill = value)) +
```

```
  geom_tile(color = "white") +
```

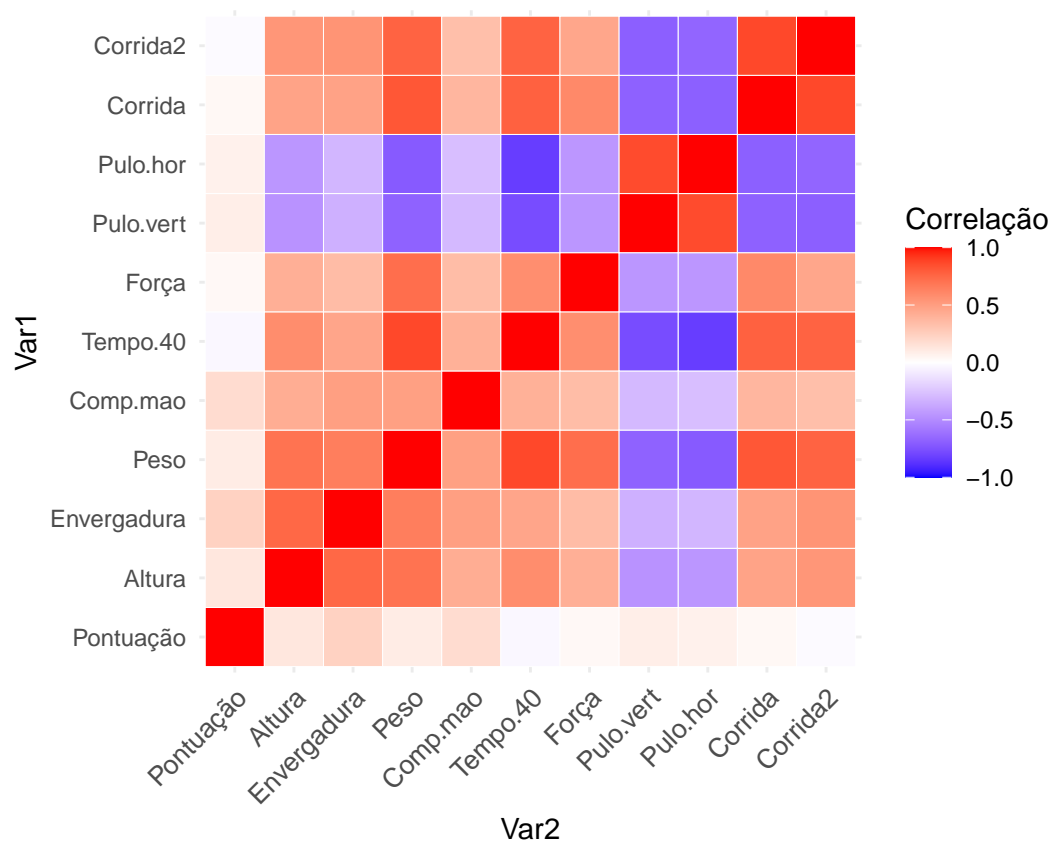
```
  scale_fill_gradient2(low = "blue", mid = "white", high = "red", midpoint = 0, limit = c(-1, 1), space = "srgb",
```

```
                      name = "Correlação") +
```

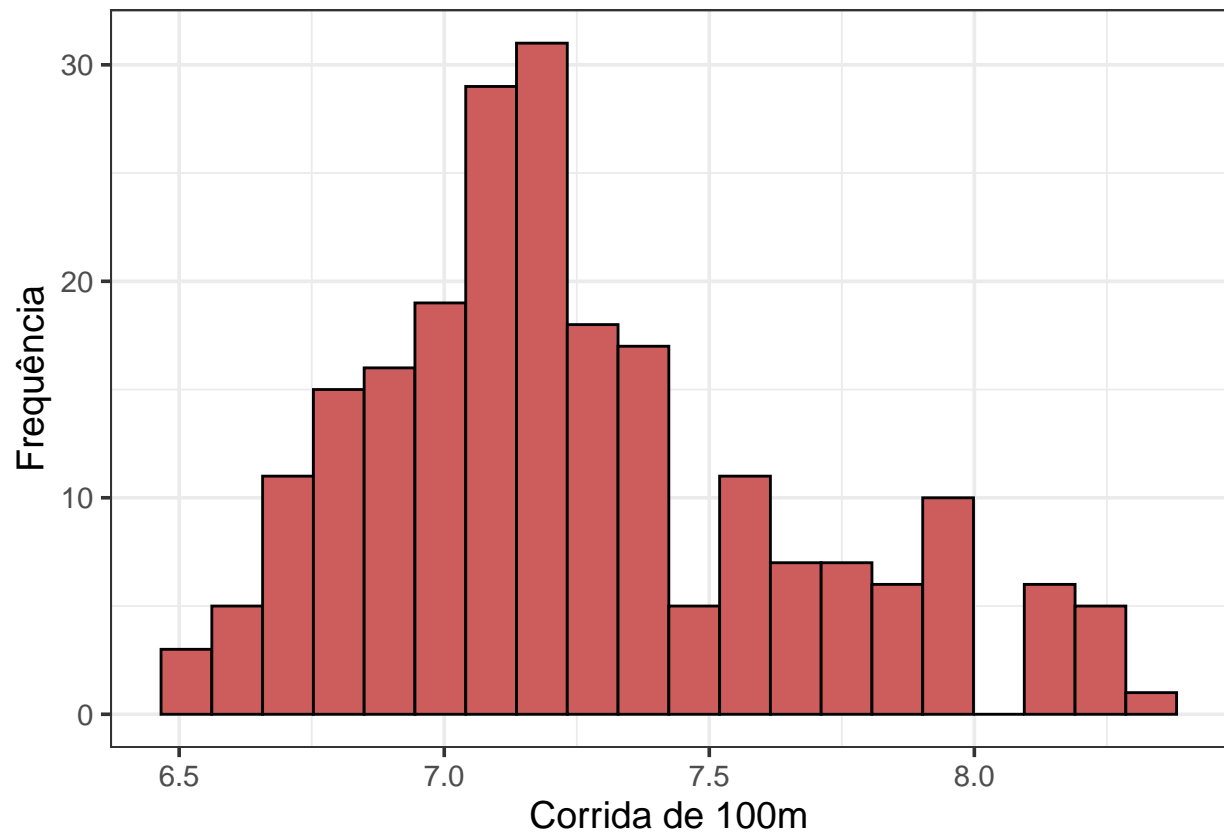
```
  theme_minimal() +
```

```
  theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 10, hjust = 1)) +
```

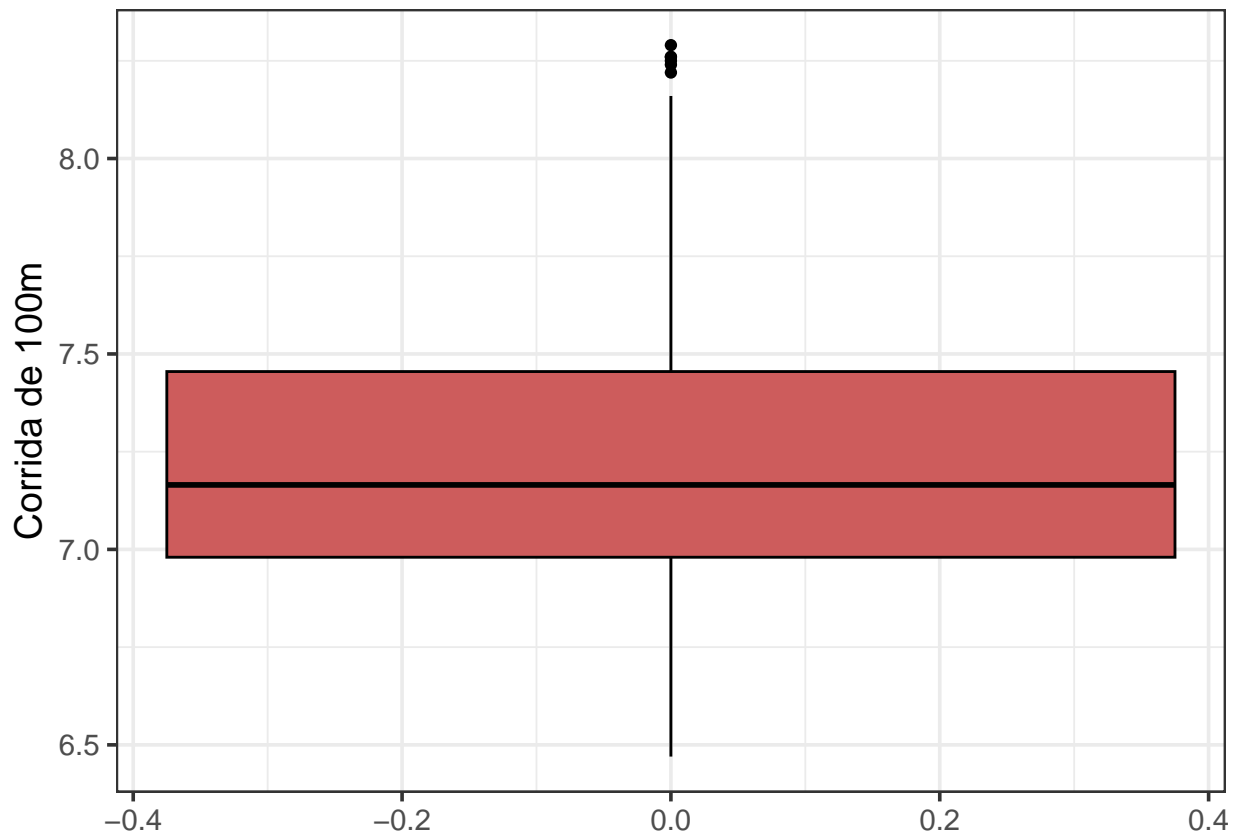
```
  coord_fixed()
```



```
#histograma
ggplot(df1[, -c(1:3)], aes(x = Corrida)) +
  geom_histogram(bins = 20, fill = 'indianred', color = 'black') +
  labs(x = "Corrida de 100m", y = "Frequência") +
  theme_bw(base_size = 14)
```



```
#boxplot
ggplot(df1[, -c(1:3)], aes(y = Corrida)) +
  geom_boxplot(fill = 'indianred', color = 'black') +
  labs(y = "Corrida de 100m") +
  theme_bw(base_size = 14)
```



```
#boxplot
#p <- plot_ly(df1[, -c(1:3)], x = ~Pontuação, y = ~"Atleta", type = "box",
#           marker = list(color = 'indianred')) %>%
# layout(title = "Pontuações nas 10 provas para cada atleta",
#        xaxis = list(title = "Pontuação"),
#        yaxis = list(title = "Atleta"),
#        showlegend = FALSE)
#p
#3d
#p3 <- plot_ly(df1[, -c(1:3)], x = ~Corrida, y = ~Corrida2, z = ~Pulo.hor,
#           type = "scatter3d", mode = "markers",
#           marker = list(color = ~Pulo.hor, size = ~Peso, colorscale = "Viridis")) %>%
# layout(title = "Gráfico em três dimensões para Corrida de 100m, de 400m e salto em distância",
#        scene = list(
#          xaxis = list(title = "Corrida de 100m"),
#          yaxis = list(title = "Corrida de 400m"),
#          zaxis = list(title = "Salto em distância")
#        ))
#p3
```

## 2.

Considere a matriz de dados:

$$X = \begin{pmatrix} -1 & 3 & -2 \\ 2 & 4 & 2 \\ 5 & 2 & 3 \end{pmatrix}$$

a) Calcule a matriz de desvios (resíduos)  $x - 1\bar{x}'$ . Essa matriz tem rank completo? Justifique.

```

X <- matrix(c(-1, 3, -2, 2, 4, 2, 5, 2, 3), nrow = 3, byrow = TRUE)
X

##      [,1] [,2] [,3]
## [1,]   -1    3   -2
## [2,]    2    4    2
## [3,]    5    2    3

media_colunas <- colMeans(X)
colMeans(X)

## [1] 2 3 1

matriz_medias_t <- matrix(rep(t(colMeans(X)),3), nrow = 3, byrow = TRUE)
matriz_medias_t

##      [,1] [,2] [,3]
## [1,]    2    3    1
## [2,]    2    3    1
## [3,]    2    3    1

matriz_desvios <- X - matriz_medias_t
matriz_desvios

##      [,1] [,2] [,3]
## [1,]   -3    0   -3
## [2,]    0    1    1
## [3,]    3   -1    2

det_matriz_desvios <- det(matriz_desvios)
det_matriz_desvios

## [1] 0

if (det_matriz_desvios != 0) {
  print("A matriz tem rank completo.")
} else {
  print("A matriz não tem rank completo.")
}

## [1] "A matriz não tem rank completo."

```

b) Determine  $S$  e calcule a variância generalizada. Interprete-a geometricamente.

```

S <- (1/(nrow(X)-1)) * t(X - matriz_medias_t) %*% (X - matriz_medias_t)
S

##      [,1] [,2] [,3]
## [1,]  9.0 -1.5  7.5
## [2,] -1.5  1.0 -0.5
## [3,]  7.5 -0.5  7.0

det_S <- det(S)
det_S

## [1] 5.995204e-15

# Determinante da matriz de covariância amostral S
det_S <- det(S)

# Volume gerado pelos vetores e1, e2, ..., ep (autovetores de S)

```



```

volume <- sqrt(abs(det_S))

# Número de observações
n <- nrow(X)

# Número de variáveis
p <- ncol(X)

# Variância generalizada
variancia_generalizada <- (volume^2) / ((n - 1)^p)

# Exibindo os resultados
cat("Determinante da matriz de covariância (|S|):", det_S, "\n")

## Determinante da matriz de covariância (|S|): 5.995204e-15
cat("Volume gerado pelos vetores (autovetores de S):", volume, "\n")

## Volume gerado pelos vetores (autovetores de S): 7.74287e-08
cat("Número de observações (n):", n, "\n")

## Número de observações (n): 3
cat("Número de variáveis (p):", p, "\n")

## Número de variáveis (p): 3
cat("Variância generalizada:", variancia_generalizada, "\n")

## Variância generalizada: 7.494005e-16
c) Usando os resultados do item anterior, calcule a variância total.
# Calculando a variância total
variancia_total <- variancia_generalizada * ((n - 1)^p)

# Exibindo o resultado
cat("Variância Total:", variancia_total, "\n")

## Variância Total: 5.995204e-15

```

### 3.

Os dados do arquivo 'radioterapia.txt', disponível na página da disciplina, referem-se a escores atribuídos a pacientes com câncer, tratados com radioterapia, segundo as seguintes variáveis:

- x1: escore relativo ao número de sintomas;
- x2: escore referente às atividades (numa escala de 1-5);
- x3: escore de sono (1-5);
- x4: escore de alimentação (1-3);
- x5: escore de apetite (1-5);
- x6: escore de reações na pele (0-3).

Faça uma análise descritiva desses dados, usando gráficos, médias, variâncias e covariâncias, correlações. . .

```
df2 <- read.table(file = "UD02_Base_Radioterapia.txt", header = TRUE, encoding = "latin1")
head(df2)
```

```
##      Symptoms Activity Sleep   Eat Appetite Skinreact
## 1      0.889      1.389 1.555 2.222    1.945         1
## 2      2.813      1.437 0.999 2.312    2.312         2
## 3      1.454      1.091 2.364 2.455    2.909         3
## 4      0.294      0.941 1.059 2.000    1.000         1
## 5      2.727      2.545 2.819 2.727    4.091         0
## 6      3.937      1.250 1.937 2.937    3.749         1
```

```
summary(df2)
```

```
##      Symptoms      Activity      Sleep      Eat
## Min.   : 0.000   Min.   :0.941   Min.   :0.666   Min.   :1.286
## 1st Qu.: 1.887   1st Qu.:1.111   1st Qu.:1.564   1st Qu.:2.000
## Median : 3.404   Median :1.641   Median :2.178   Median :2.139
## Mean   : 3.542   Mean   :1.809   Mean   :2.138   Mean   :2.209
## 3rd Qu.: 5.178   3rd Qu.:2.323   3rd Qu.:2.712   3rd Qu.:2.440
## Max.   :10.461   Max.   :4.000   Max.   :4.000   Max.   :2.937
##      Appetite      Skinreact
## Min.   :1.000   Min.   :0.000
## 1st Qu.:1.924   1st Qu.:1.000
## Median :2.500   Median :1.000
## Mean   :2.575   Mean   :1.276
## 3rd Qu.:3.272   3rd Qu.:2.000
## Max.   :5.000   Max.   :3.000
```

```
#matriz de covariância
```

```
matriz_cov <- cov(df2, use = "pairwise.complete.obs")
```

```
matriz_cov_long <- melt(matriz_cov)
```

```
ggplot(matriz_cov_long, aes(Var2, Var1, fill = value)) +
```

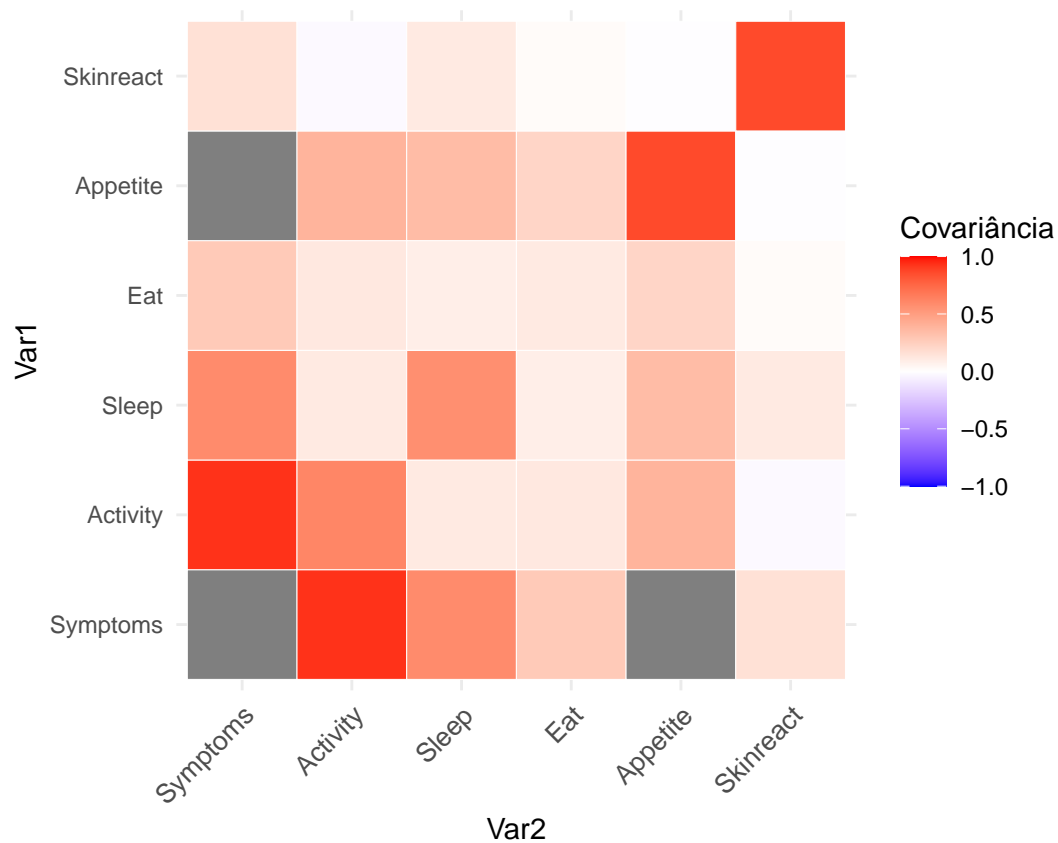
```
  geom_tile(color = "white") +
```

```
  scale_fill_gradient2(low = "blue", mid = "white", high = "red", midpoint = 0, limit = c(-1, 1), space
    name = "Covariância") +
```

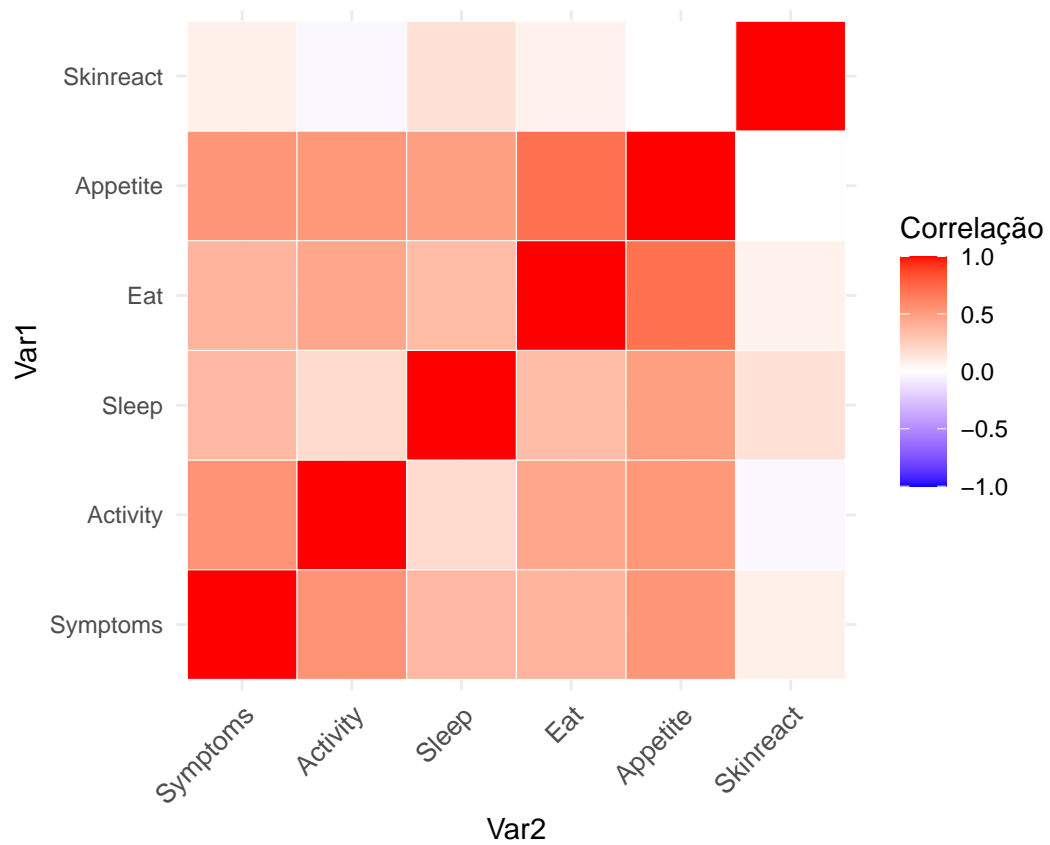
```
  theme_minimal() +
```

```
  theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 10, hjust = 1)) +
```

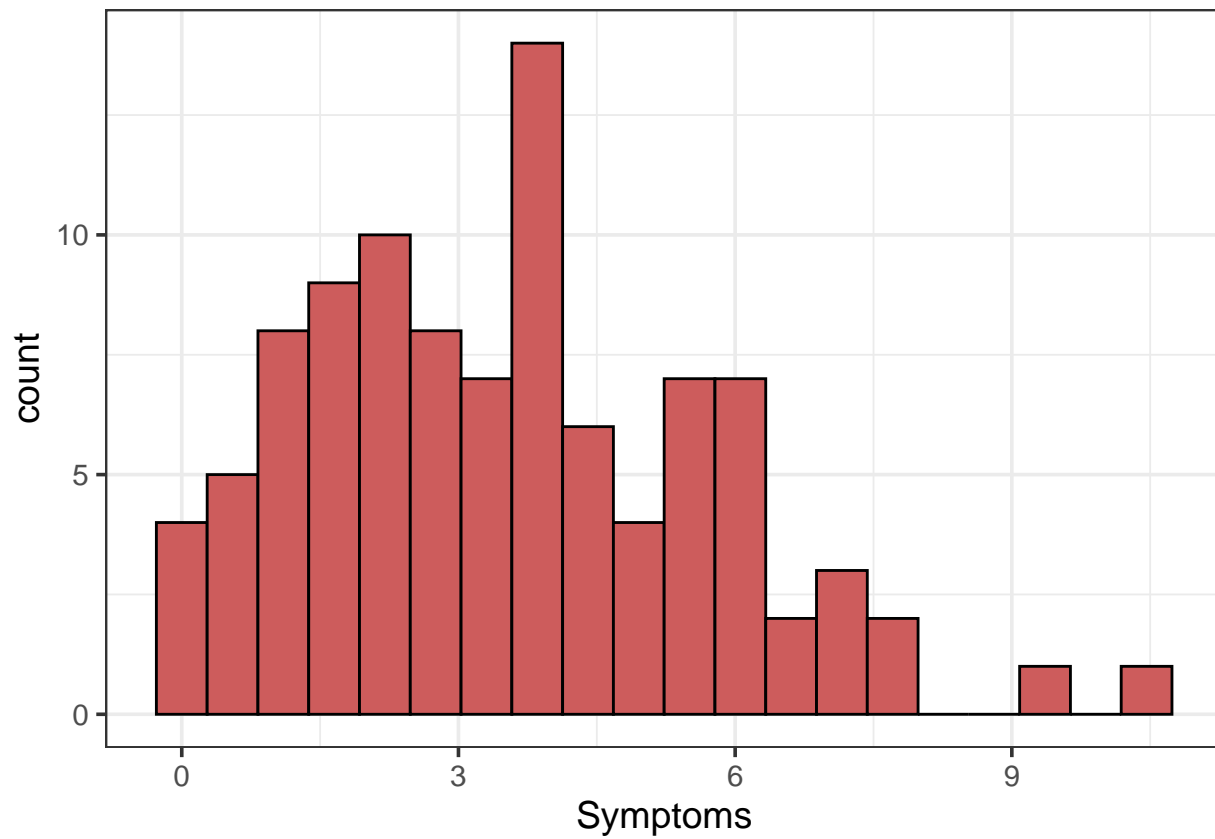
```
  coord_fixed()
```



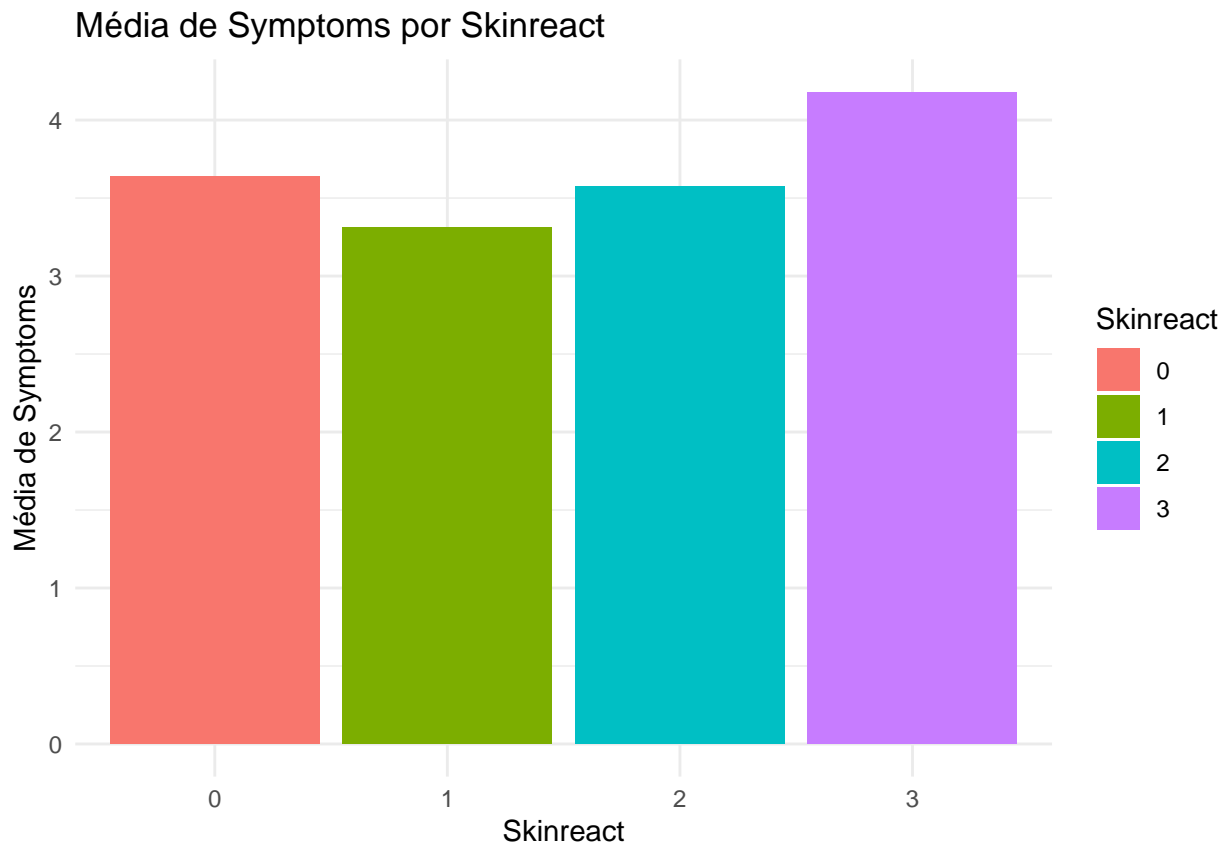
```
#matriz de correlação
matriz_cor <- cor(df2, use = "pairwise.complete.obs")
matriz_cor_long <- melt(matriz_cor)
ggplot(matriz_cor_long, aes(Var2, Var1, fill = value)) +
  geom_tile(color = "white") +
  scale_fill_gradient2(low = "blue", mid = "white", high = "red", midpoint = 0, limit = c(-1, 1), space
    name = "Correlação") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 10, hjust = 1)) +
  coord_fixed()
```



```
#histograma
ggplot(df2, aes(x = Symptoms)) +
  geom_histogram(bins = 20, fill = 'indianred', color = 'black') +
  labs(x = "Symptoms") +
  theme_bw(base_size = 14)
```



```
#barplot symptoms por skinreact
ggplot(df2, aes(x = factor(Skinreact), y = Symptoms, fill = factor(Skinreact))) +
  geom_bar(stat = "summary", fun = "mean", position = "dodge") +
  labs(x = "Skinreact", y = "Média de Symptoms", fill = "Skinreact") +
  ggtitle("Média de Symptoms por Skinreact") +
  theme_minimal()
```



#### 4.

Os dados contidos no arquivo ossos.txt, disponível na página da disciplina, referem-se ao tamanho (cm) de determinado osso em crianças. Cada criança teve o osso medido semestralmente a partir dos oito anos de idade, de tal forma que  $y_1$  se refere à medida aos 8 anos de idade,  $y_2$  aos oito e meio,  $y_3$  aos nove e  $y_4$  aos nove e meio.

a) Usando o R, obtenha o vetor de médias, a matriz de covariâncias e a matriz de correlações amostrais;

```
df3 <- read.table(file = "UD02_Base_Ossos.txt", header = TRUE, encoding = "latin1")
df3 <- df3[,-c(1)]
head(df3)
```

```
##      y1    y2    y3    y4
## 1 47.8 48.8 49.0 49.7
## 2 46.4 47.3 47.7 48.4
## 3 46.3 46.8 47.8 48.5
## 4 45.1 45.3 46.1 47.2
## 5 47.6 48.5 48.9 49.3
## 6 52.5 53.2 53.3 53.7
```

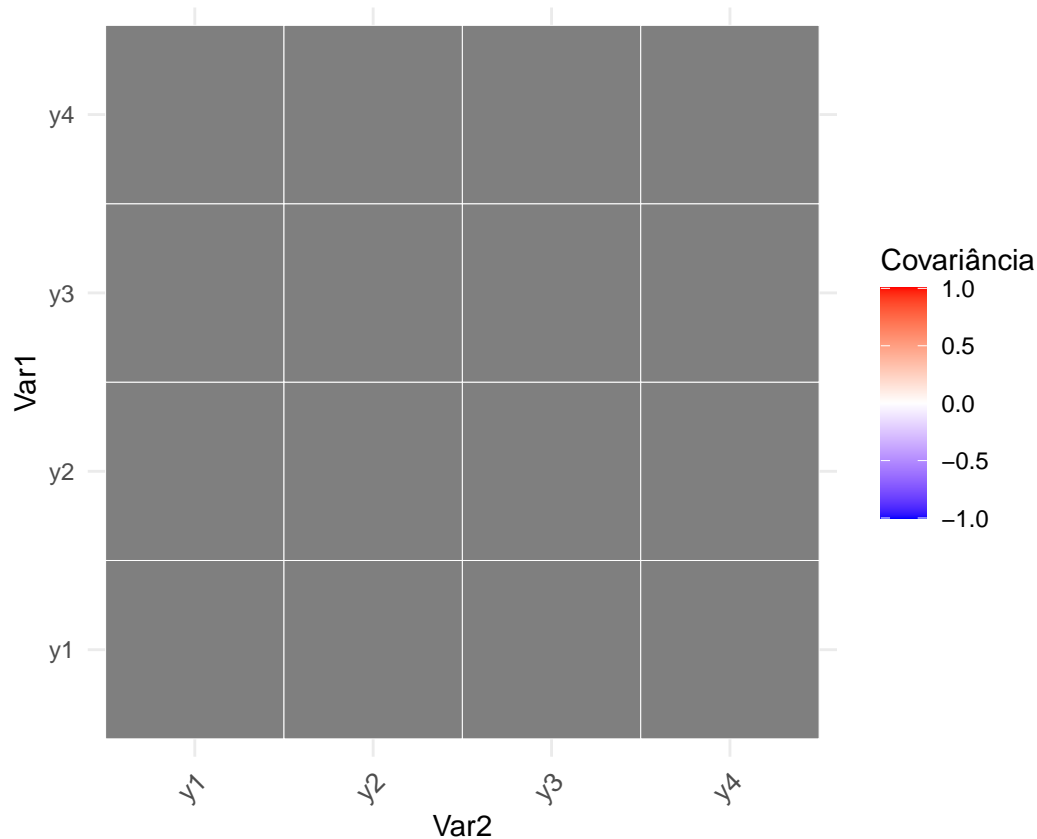
```
df3 <- as.matrix(df3)
#vetor médias
media_colunas <- colMeans(df3)
media_colunas
```

```
##      y1      y2      y3      y4
## 48.655 49.625 50.570 51.450
```

```

#matriz de covariância
matriz_cov <- cov(df3, use = "pairwise.complete.obs")
matriz_cov_long <- melt(matriz_cov)
ggplot(matriz_cov_long, aes(Var2, Var1, fill = value)) +
  geom_tile(color = "white") +
  scale_fill_gradient2(low = "blue", mid = "white", high = "red", midpoint = 0, limit = c(-1, 1), space
    name = "Covariância") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 10, hjust = 1)) +
  coord_fixed()

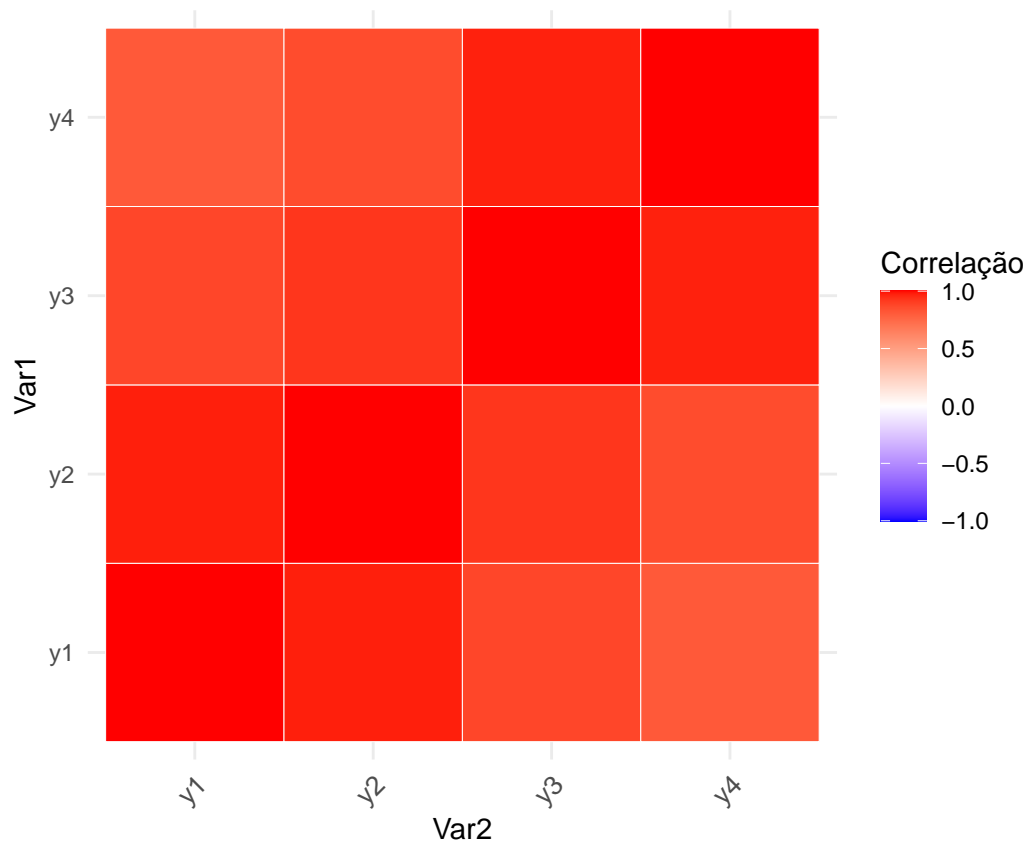
```



```

#matriz de correlação
matriz_cor <- cor(df3, use = "pairwise.complete.obs")
matriz_cor_long <- melt(matriz_cor)
ggplot(matriz_cor_long, aes(Var2, Var1, fill = value)) +
  geom_tile(color = "white") +
  scale_fill_gradient2(low = "blue", mid = "white", high = "red", midpoint = 0, limit = c(-1, 1), space
    name = "Correlação") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 10, hjust = 1)) +
  coord_fixed()

```



b) Obtenha a variância generalizada e a variância total;

```
matriz_medias_t <- matrix(rep(media_colunas,nrow(df3)), nrow = nrow(df3), byrow = TRUE)
matriz_medias_t
```

```
##      [,1] [,2] [,3] [,4]
## [1,] 48.655 49.625 50.57 51.45
## [2,] 48.655 49.625 50.57 51.45
## [3,] 48.655 49.625 50.57 51.45
## [4,] 48.655 49.625 50.57 51.45
## [5,] 48.655 49.625 50.57 51.45
## [6,] 48.655 49.625 50.57 51.45
## [7,] 48.655 49.625 50.57 51.45
## [8,] 48.655 49.625 50.57 51.45
## [9,] 48.655 49.625 50.57 51.45
## [10,] 48.655 49.625 50.57 51.45
## [11,] 48.655 49.625 50.57 51.45
## [12,] 48.655 49.625 50.57 51.45
## [13,] 48.655 49.625 50.57 51.45
## [14,] 48.655 49.625 50.57 51.45
## [15,] 48.655 49.625 50.57 51.45
## [16,] 48.655 49.625 50.57 51.45
## [17,] 48.655 49.625 50.57 51.45
## [18,] 48.655 49.625 50.57 51.45
## [19,] 48.655 49.625 50.57 51.45
## [20,] 48.655 49.625 50.57 51.45
```



```
matriz_desvios <- df3 - matriz_medias_t
matriz_desvios
```

```
##           y1      y2      y3      y4
## [1,] -0.855 -0.825 -1.57 -1.75
## [2,] -2.255 -2.325 -2.87 -3.05
## [3,] -2.355 -2.825 -2.77 -2.95
## [4,] -3.555 -4.325 -4.47 -4.25
## [5,] -1.055 -1.125 -1.67 -2.15
## [6,]  3.845  3.575  2.73  2.25
## [7,]  2.545  3.375  3.73  3.05
## [8,]  1.145  0.375 -0.27  1.25
## [9,] -0.555  1.175  1.73  2.95
## [10,] -3.655 -2.625 -3.27 -3.15
## [11,]  2.545  1.775  1.03  0.45
## [12,] -0.155 -0.425  2.43  4.05
## [13,]  3.445  3.175  3.13  3.55
## [14,] -0.455 -0.725 -1.27 -1.65
## [15,]  0.945  0.775  0.63  0.35
## [16,]  2.045  2.075  2.13  1.85
## [17,] -1.455 -1.925 -2.17 -1.95
## [18,]  4.645  4.975  4.53  3.85
## [19,] -2.455 -2.125 -2.47 -3.05
## [20,] -2.355 -2.025  0.73  0.35
```

```
S <- (1/(nrow(df3)-1)) * t(df3 - matriz_medias_t) %*% (df3 - matriz_medias_t)
S
```

```
##           y1      y2      y3      y4
## y1 6.329974 6.189079 5.777000 5.548158
## y2 6.189079 6.449342 6.153421 5.923421
## y3 5.777000 6.153421 6.918000 6.946316
## y4 5.548158 5.923421 6.946316 7.464737
```

```
det_S <- det(S)
det_S
```

```
## [1] 1.068328
```

```
# Determinante da matriz de covariância amostral S
```

```
det_S <- det(S)
```

```
# Volume gerado pelos vetores e1, e2, ..., ep (autovetores de S)
```

```
volume <- sqrt(abs(det_S))
```

```
# Número de observações
```

```
n <- nrow(df3)
```

```
# Número de variáveis
```

```
p <- ncol(df3)
```

```
# Variância generalizada
```

```
variancia_generalizada <- (volume^2) / ((n - 1)^p)
```

```
# Exibindo os resultados
```

```
cat("Determinante da matriz de covariância (|S|):", det_S, "\n")
```

```
## Determinante da matriz de covariância (|S|): 1.068328
cat("Volume gerado pelos vetores (autovetores de S):", volume, "\n")
```

```
## Volume gerado pelos vetores (autovetores de S): 1.033599
cat("Número de observações (n):", n, "\n")
```

```
## Número de observações (n): 20
cat("Número de variáveis (p):", p, "\n")
```

```
## Número de variáveis (p): 4
cat("Variância generalizada:", variancia_generalizada, "\n")
```

```
## Variância generalizada: 8.197664e-06
# Calculando a variância total
variancia_total <- variancia_generalizada * ((n - 1)^p)

# Exibindo o resultado
cat("Variância Total:", variancia_total, "\n")
```

```
## Variância Total: 1.068328
```

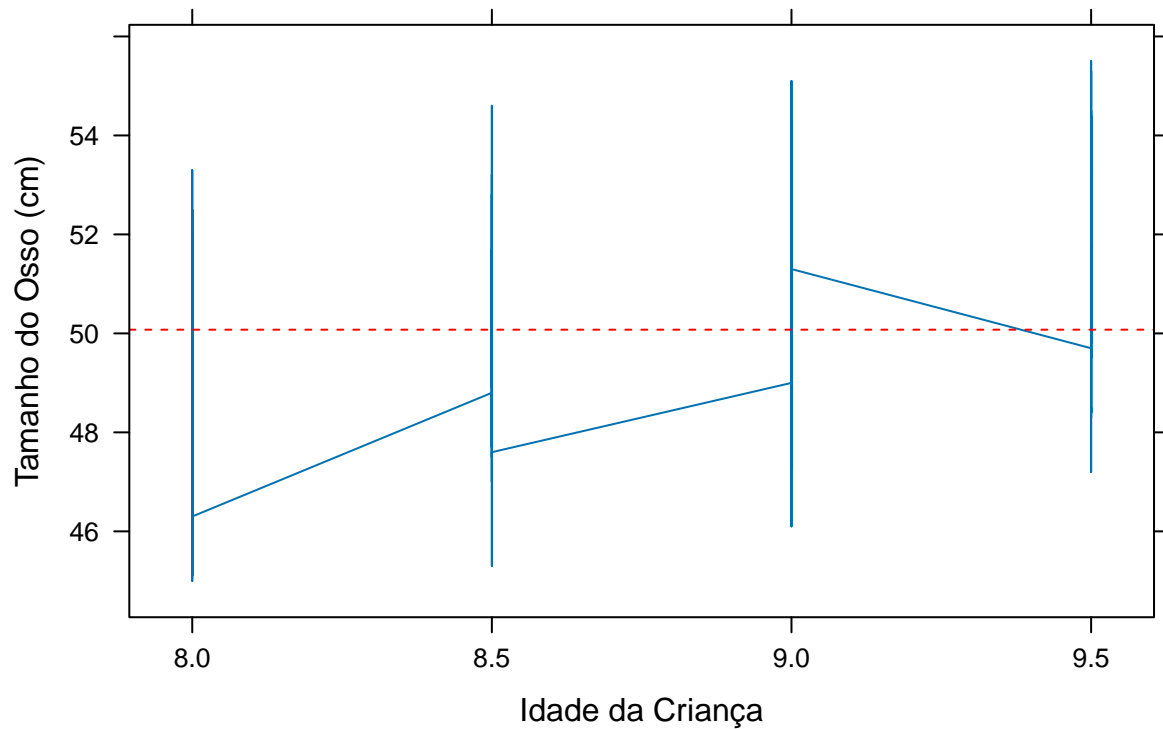
- c) Usando a função `xyplot` da biblioteca `lattice`, faça um gráfico com o tamanho do osso no eixo vertical, a idade da criança no eixo horizontal e cada criança representada por uma linha, que une os quatro pontos correspondentes às suas quatro medidas. Adicione ao gráfico a linha média.

```
df3 <- read.table(file = "UD02_Base_Ossos.txt", header = TRUE, encoding = "latin1")
df3 <- df3[,-c(1)]
dataxy <- data.frame(idade = c(rep(8,20),rep(8.5,20),rep(9,20),rep(9.5,20)), valores = c(df3$y1,df3$y2,df3$y3,df3$y4))

# Criar o gráfico com xyplot
p <- xyplot(valores ~ idade , data = dataxy, type = "l", lwd = 1,
  xlab = "Idade da Criança", ylab = "Tamanho do Osso (cm)",
  main = "Tamanho do Osso em Crianças",
  panel = function(x, y, ...) {
    panel.xyplot(x, y, ...)
    panel.abline(h = mean(y, na.rm = TRUE), col = "red", lty = 2)
  })

# Mostrar o gráfico
print(p)
```

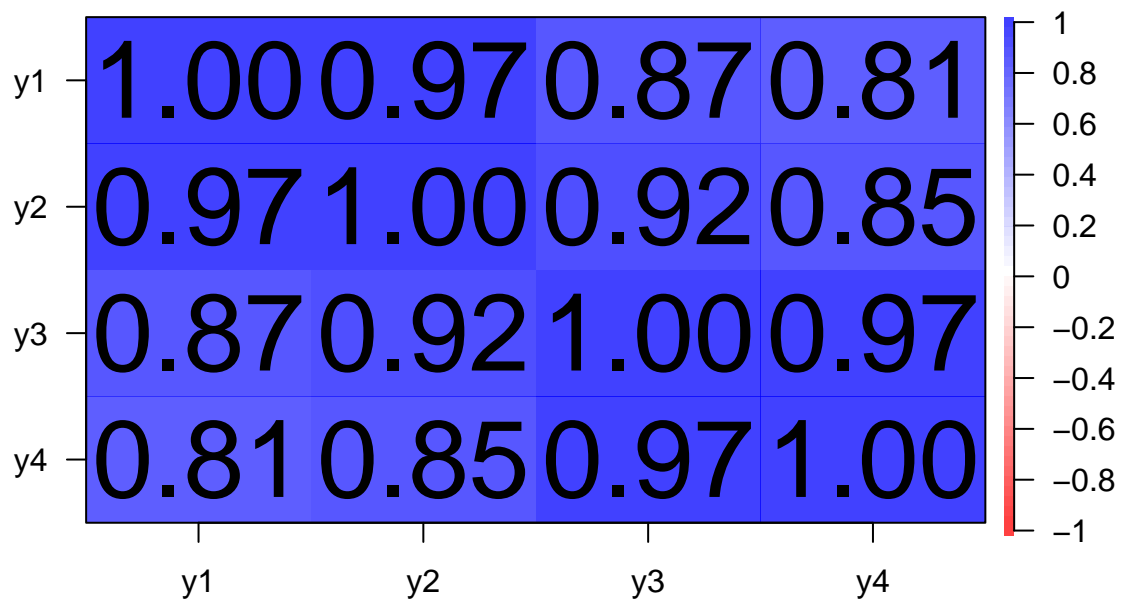
## Tamanho do Osso em Crianças



d) Faça um correlograma. Analise os resultados obtidos.

```
cor.plot(df3)
```

## Correlation plot from data



## 5.

Os dados presentes no arquivo 'glicose.txt', disponível na página da disciplina, referem-se a seis medidas de glicose no sangue, tomadas em 50 mulheres. Os y's representam os níveis de glicose registrados nas três ocasiões; os x's são as medidas de glicose tomadas uma hora após a ingestão de açúcar.

- Obtenha o vetor de médias, a matriz de covariâncias e a matriz de correlações amostrais para todas as seis variáveis;
- Considerando a partição das variáveis originais em  $(x; y)$ , re-apresente os resultados do item anterior, destacando, no vetor e nas matrizes, os vetores de média e as matrizes de covariâncias originadas pela partição;
- Construa gráficos 3d para  $x$  e para  $y$ . Experimente diferentes ângulos para a visualização dos resultados.

## 6.

Considere as duas matrizes de covariâncias:

$$\Sigma_1 = \begin{pmatrix} 14 & 8 & 3 \\ 8 & 5 & 2 \\ 3 & 2 & 1 \end{pmatrix}$$

$$\Sigma_2 = \begin{pmatrix} 6 & 6 & 1 \\ 6 & 8 & 2 \\ 1 & 2 & 1 \end{pmatrix}$$

Mostre que a variância generalizada da população 2 é maior do que a da população 1, embora a variância total seja maior para a população 1. Comente com relação a esse fato, baseado nas covariâncias e nas correlações.

```
# Declarando a matriz X1
X1 <- matrix(c(14, 8, 3, 8, 5, 2, 3, 2, 1), nrow = 3, ncol = 3, byrow = TRUE)
X1

##      [,1] [,2] [,3]
## [1,]  14   8   3
## [2,]   8   5   2
## [3,]   3   2   1

X2 <- matrix(c(6, 6, 1, 6, 8, 2, 1, 2, 1), nrow = 3, ncol = 3, byrow = TRUE)
X2

##      [,1] [,2] [,3]
## [1,]   6   6   1
## [2,]   6   8   2
## [3,]   1   2   1

media_colunas <- colMeans(X1)
matriz_medias_t <- matrix(rep(media_colunas,nrow(X1)), nrow = nrow(X1), byrow = TRUE)
matriz_desvios <- X1 - matriz_medias_t
S <- (1/(nrow(X1)-1)) * t(matriz_desvios) %*% (matriz_desvios)
det_S <- det(S)
volume <- sqrt(abs(det_S))
n <- nrow(X1)
p <- ncol(X1)
variancia_generalizada <- (volume^2) / ((n - 1)^p)
cat("Variância generalizada X1:", variancia_generalizada, "\n")

## Variância generalizada X1: 4.634964e-18
```

```

variancia_total <- variancia_generalizada * ((n - 1)^p)
cat("Variância Total X2:", variancia_total, "\n")

## Variância Total X2: 3.707971e-17

media_colunas <- colMeans(X2)
matriz_medias_t <- matrix(rep(media_colunas,nrow(X2)), nrow = nrow(X2), byrow = TRUE)
matriz_desvios <- X2 - matriz_medias_t
S <- (1/(nrow(X2)-1)) * t(matriz_desvios) %*% (matriz_desvios)
det_S <- det(S)
volume <- sqrt(abs(det_S))
n <- nrow(X2)
p <- ncol(X2)
variancia_generalizada <- (volume^2) / ((n - 1)^p)
cat("Variância generalizada X2:", variancia_generalizada, "\n")

## Variância generalizada X2: 0

variancia_total <- variancia_generalizada * ((n - 1)^p)
cat("Variância Total X2:", variancia_total, "\n")

## Variância Total X2: 0

```

## 7.

Considere os dois seguintes vetores de dados:

$$y_1 = \begin{pmatrix} 10 \\ 8 \\ 2 \end{pmatrix}$$

$$y_2 = \begin{pmatrix} 6 \\ 5 \\ 4 \end{pmatrix}$$

Considerando esse par de observações, calcule as seguintes medidas de distância sem usar as funções específicas do R:

- Distância Euclideana;
- Distância de Manhattan;
- Distância de Chebyshev;
- Distância de Canberra;
- Distância de Minkowski.

```

# Vetores y1 e y2
y1 <- c(10, 8, 2)
y2 <- c(6, 5, 4)

# Cálculo da distância Euclidiana
dist_euclidiana <- sqrt(sum((y1 - y2)^2))
dist_euclidiana

## [1] 5.385165

# Cálculo da distância de Manhattan
dist_manhattan <- sum(abs(y1 - y2))
dist_manhattan

```

```
## [1] 9
# Cálculo da distância de Chebyshev
dist_chebyshev <- max(abs(y1 - y2))
dist_chebyshev

## [1] 4
# Cálculo da distância de Canberra
dist_canberra <- sum(abs(y1 - y2) / (abs(y1) + abs(y2)))
dist_canberra

## [1] 0.8141026
# Cálculo da distância de Minkowski com p = 3
p <- 3
dist_minkowski <- (sum(abs(y1 - y2)^p))^(1/p)
dist_minkowski

## [1] 4.626065
```

## 8.

Considere os dois seguintes vetores de dados binários:

$$y_1 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \qquad y_2 = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}$$

Considerando esse par de observações, calcule as seguintes medidas de similaridades sem usar as funções específicas do R: a) Índice de empates simples; b) Índice de Jaccard; c) Índice de Dice-Sorensen; d) Índice de Oshiai.

```
# Vetores y1 e y2
y1 <- c(0, 0, 1, 0, 1, 1, 1, 1)
y2 <- c(0, 1, 1, 0, 0, 1, 0, 1)

# Cálculo do índice de empates simples
num_empates <- sum(y1 == y2)
num_total <- length(y1)
indice_empates_simples <- num_empates / num_total
indice_empates_simples

## [1] 0.625

# Cálculo do índice de Jaccard
num_empates <- sum(y1 == y2)
num_nao_empates <- sum(y1 != y2)
indice_jaccard <- num_empates / (num_empates + num_nao_empates)
indice_jaccard

## [1] 0.625
```

```
# Cálculo do índice de Dice-Sorensen
num_empates <- sum(y1 == y2)
num_elem_y1 <- sum(y1)
num_elem_y2 <- sum(y2)
indice_dice_sorensen <- 2 * num_empates / (num_empates + num_elem_y1 + num_elem_y2)
indice_dice_sorensen
```

```
## [1] 0.7142857
```

```
# Cálculo do índice de Ochiai
num_empates <- sum(y1 == y2)
sqrt_num_elem_y1_y2 <- sqrt(sum(y1) * sum(y2))
indice_ochiai <- num_empates / sqrt_num_elem_y1_y2
indice_ochiai
```

```
## [1] 1.118034
```

## 9.

Considere a base de dados USArrests. Usando as funções específicas do R, obtenha matrizes de distância usando as medidas estudadas em sala de aula. Produza visualizações das matrizes de distâncias usando mapas de calor. Compare os resultados.