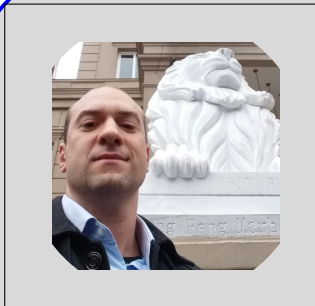


PROJETO INTEGRADOR DE COMPETÊNCIAS EM CIÊNCIA DE DADOS I - 40h_Turma_05_102020

Rodrigo Hermont Ozon*

Outubro, 2020



Sobre o Autor:



Rodrigo Hermont Ozon, economista e apaixonado por econometria, pelas aplicações de modelos econômicos a problemas reais e cotidianos vivenciados na sociedade e na realidade das empresas.

Seus contatos podem ser acessados em:

-  [Github](#)
-  [Linkedin](#)

Resumo

Este documento apresenta a resolução dos exercícios do curso Tecnológico em Ciência de Dados da Universidade Cruzeiro do Sul virtual para a disciplina de Projeto Integrador de Competência em Ciência de Dados I.

A produção deste .pdf utiliza a prática de pesquisa reproduzível utilizando o  e pode ser feito completamente de forma gratuita na plataforma Overleaf.com. Caso você queira ver um tutorial de como começar a integrar a linguagem \LaTeX com o  [consulte esse artigo que eu postei no meu !\[\]\(c244836fd67166dc60ebf5279a0f8377_img.jpg\)](#)

Como exercício criei uma página na web escrita em RMarkdown contendo as interatividades gráficas para esses exercícios aqui propostos. Caso tenha interesse, visite minha página em [meu perfil no Github !\[\]\(3211b5d1d968fc1665909b34f9f16010_img.jpg\)](#)

À minha amada esposa, Idiane *"Então o anjo do Senhor lhe apareceu, e lhe disse: O Senhor é contigo, homem valoroso."*

[Juízes 6:12](#)

Sumário

1	Atividade I	7
1.1	Resolução	7
1.1.1	Crie um gráfico de barras apresentando o resultado de cada item.	8
1.1.2	Liste os componentes da matriz dos dados.	9
1.1.3	Liste a soma e a média aritmética dessa matriz	10
1.1.4	Liste o produto dos elementos da matriz	10
2	Atividade II	12
2.1	Resolução	12
2.1.1	Exibir as equipes que são do país “Alemanha”.	13
2.1.2	Exibir quais os pilotos que são do mesmo país do piloto “Max Chilton”	13
2.1.3	Selecionar os pilotos que são da equipe “McLaren-Mercedes”	13
3	Atividade III	14
3.1	Resolução	14
3.1.1	Utilize a linguagem R para plotar X e Y em gráficos diferentes utilizando BOXPLOT.	15
3.1.2	Usando R, plote a comparação de X e Y no mesmo gráfico utilizando BOXPLOT.	17
4	Atividade IV	18
4.1	Resolução	19
4.1.1	Qual a média IMC dessa pequena amostra?	19
4.1.2	Boxplot comparando a peso M, peso Fe peso geral no mesmo gráfico.	21
4.1.3	Boxplot da altura geral	22

Lista de Figuras

1	Gráfico de barras para valores e títulos	8
2	Gráfico de barras ordenado para valores e títulos	10
3	Boxplots x e y	15
4	Boxplots x e y	16
5	Boxplot x e y	17
6	Densidades Kernel de X e de Y	18
7	Boxplot comparativo de peso x sexo x peso geral (média(peso))	21
8	Boxplot da Altura	22

Atenção:

Esse é um documento \LaTeX dinâmico, onde os *chunks* de código R rodam dentro dele.

O exercício de SQL (se fosse com dados reais) poderia ser rodado dentro do R, carregando o pacote RSQLite.

Também utilizei o ambiente `lstlisting` para economizar tempo na compilação do documento:

```
\begin{lstlisting}[language=R]
```

```
Os códigos R ficam todas dentro desse formato neste documento
```

```
\end{lstlisting}
```

Introdução

A disciplina de projeto integrador de competências visa, além de sistematizar os conhecimentos adquiridos, também desenvolver atitudes e estratégias de pensamento focadas na resolução de problemas práticos das áreas de futura atuação dos alunos, bem como proporcionar a transversalidade disciplinar utilizada na prática profissional no dia a dia.

Não obstante, é uma valiosa prática andragógica que oferece a necessária vivência do ambiente de trabalho, fundamental para o amadurecimento e mediante aplicação dos conhecimentos em situações reais, abordando sempre problemas de cunho rigorosamente prático.

Além disso, é de fundamental importância a integração entre as disciplinas, pois, no exercício da profissão, o discente estará exposto a desafios que projetam-se transversalmente às disciplinas e é importante saber a qual domínio pertence a solução e em quantas etapas esses desafios se dividem até que se encontre a resposta e a execução em fases mais simples e de seu domínio.

O objetivo é utilizar e evoluir em relação às novas competências recém adquiridas visando sua fixação e otimização no que tangem as seguintes disciplinas: Modelagem de Dados, Análise de Dados Exploratória e Linguagem de Banco de Dados.

Em conformidade ao descrito, logo a seguir serão passadas uma série de leituras de reforço que necessitarão de sua atenção e esforço para recordar e se preparar para os desafios dispostos em nossa trilha de aprendizagem.

Também é bom lembrar que a língua mundialmente falada em tecnologia da informação é Inglês, portanto, é importante você estar estudando esse idioma para evoluir profissionalmente nesse mercado. Claro, você pode utilizar o tradutor da Google, mas é importante praticar e dominar o idioma.

Colocamos links com ferramentas e leituras, cujas indicações são:

Para tanto utilizaremos a linguagem R mundialmente usada para manipulação estatística de dados. Isso com certeza ajudará a dar maior confiança em sua jornada.

Caso você ainda não tenha instalado, vamos fazer isso, porque você vai utilizar para realizar os exercícios. Vamos lá então:

Disponível em: <https://www.r-project.org/> aqui você encontra tudo sobre a linguagem e se quiser o direcionamento para baixar o instalador.

Link para baixar o instalador direto: Disponível em: <https://cran-r.c3sl.ufpr.br/>

Levamos em conta que talvez você ainda não esteja familiarizado com a linguagem, então resolvemos lhe dar um reforço: Aqui você encontra muita documentação oficial para te ajudar: disponível em: <https://cran.r-project.org/manuals.html>

Lembrando que apesar da linguagem de manipulação de dados R, temos que saber das fontes e de como as modelamos. Nesse tocante, a modelagem de dados é uma habilidade crucial para todo cientista de dados, esteja você fazendo projetos de pesquisa ou arquitetando um novo repositório para o armazenamento de dados em uma empresa. E, para isso, devemos lembrar que a modelagem de dados trata do processo de produzir um diagrama descritivo de relações entre vários tipos de informações que devem ser armazenadas em um banco de dados e cujo objetivo é criar o método mais eficiente de armazenar informações num banco de dados. Focaremos aqui no diagrama E-R (entidade/relacionamento), assim como a linguagem para manipulação de dados no banco de dados SQL, assim você poderá extrair e realizar junções de dados alvos para extração e submissão à linguagem R. Veja a leitura abaixo indicada na biblioteca virtual:

Para modelagem de dados E-R e linguagem SQL, recomendo a leitura de: MEDEIROS, L.F. Banco de Dados: Princípios e Prática. Curitiba: Ed. Intersaberes. 2013. Disponível na biblioteca virtual da Pearsons em:

[Acesse e baixe aqui: !\[\]\(21226b58c700e5231ab98d27101bac58_img.jpg\)](#)

Há uma série de ferramentas úteis em seu dia a dia para modelar dados de forma conceitual, uma das mais utilizadas é o Modelo BR, que você encontrará no seguinte link:

disponível em <https://sourceforge.net/projects/brmodelo30/>

Você pode preferir baixar o Workbench do MySQL se tiver mais familiarizado, para tanto é só você seguir as instruções da página de download. Workbench oferece:

- Design e modelagem de banco de dados
 - Desenvolvimento SQL
 - Administração de banco de dados
 - Migração de Banco de Dados: Disponível em: <https://dev.mysql.com/downloads/workbench/>
-

1 Atividade I

Você foi contratado pela empresa XPTO para a função de analista de dados. Como parte de seu trabalho, foi lhe confiada uma extração de dados contendo os seguintes valores: (25, 45, 28, 79, 74, 61, 12, 68, 93, 39, 100), sendo que cada valor na sua atual sequência atende pelos títulos: (alface, cenoura, pepino, chuchu, pimenta, couve, rúcula, cebola, alho, pimentão, alcachofra). Pediram para que você realize o seguinte, utilizando a linguagem R, e apresentar os resultados.

1. Crie um gráfico de barras apresentando o resultado de cada item.
2. Liste os componentes da matriz dos dados.
3. Liste o resultado da soma dessa matriz.
4. Liste a média aritmética dessa matriz.
5. Liste o produto dos elementos dessa matriz.

Instruções de envio

Não envie arquivo .zip 1 arquivo em PDF exibindo seu código; 1 arquivo com o código em R;

1.1 Resolução

Carrego os vetores de dados numéricos e qualitativos (nomes das verduras)

```
valores<-c(25, 45, 28, 79, 74, 61, 12, 68, 93, 39, 100)

titulos<-c("alface", "cenoura", "pepino", "chuchu", "pimenta", "couve", "rúcula", "cebola",
"alho", "pimentão", "alcachofra")

dados<-data.frame(titulos, valores) # Crio o conjunto de dados combinando titulos e valores

dados # Mostra o dataframe
```

	titulos	valores
1	alface	25
2	cenoura	45

```


3      pepino      28
4      chuchu     79
5      pimenta    74
6      couve      61
7      rúcula     12
8      cebola     68
9      alho       93
10     pimentão   39
11 alcachofra    100

str(dados) # Mostra a estrutura e o tipo de variáveis no dataset

'data.frame': 11 obs. of 2 variables:
 $ titulos: Factor w/ 11 levels "alcachofra","alface",...: 2 5 8 6 9 7 11 4 3 10 ...
 $ valores: num  25 45 28 79 74 61 12 68 93 39 ...

```

1.1.1 Crie um gráfico de barras apresentando o resultado de cada item.

Após informarmos ao  as variáveis, podemos criar um gráfico de barras:

```

library(ggplot2) # Pacote do R para graficos estaticos de qualidade

ggplot(data=dados,aes(x=titulos,y=valores))+
  geom_bar(position = "dodge", stat="identity")+
  geom_text(aes(label=valores), position=position_dodge(width=0.9),
    vjust=-0.25, size=2.5) #+

```

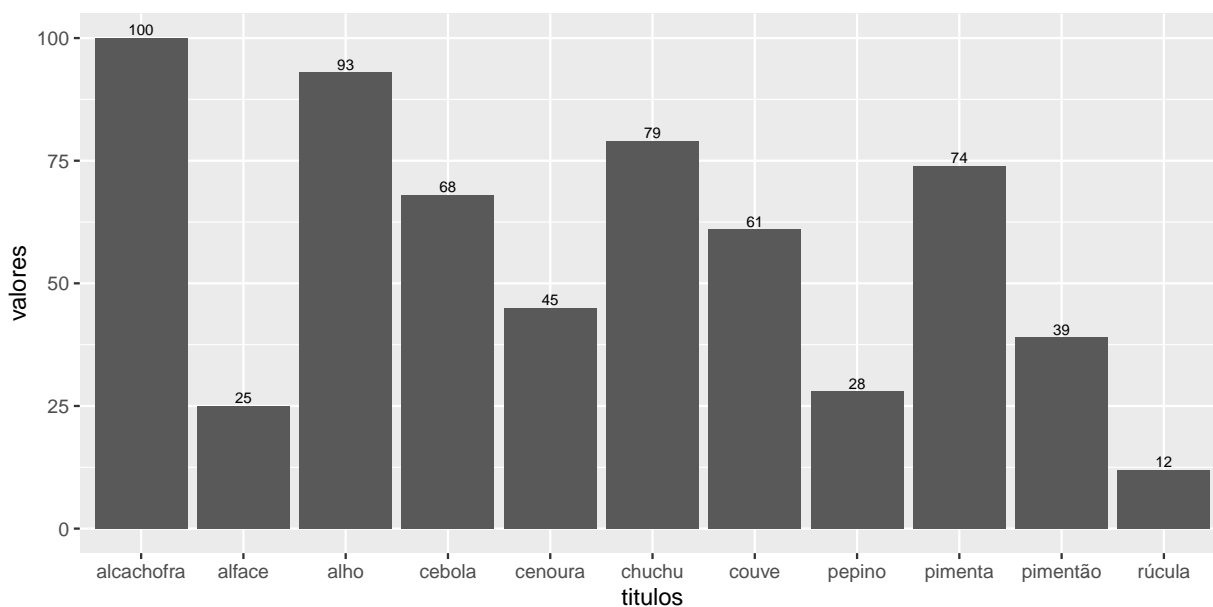


Figura 1: Gráfico de barras para valores e títulos


```
#Insere os rotulos acima das barras de tamanho 2.5
# ggtitle("Gráfico de barras para valores e títulos")+
# theme(plot.title = element_text(hjust = 0.5)) # Ajusta o titulo do grafico em cima no meio
```

1.1.2 Liste os componentes da matriz dos dados.

Para listarmos os componentes da matriz de dados, podemos simplesmente ordenar do maior para o menor:

```
library(dplyr) # Pacote para rodar as formulas com o operador pipe

dados%>%
group_by(titulos)%>%
arrange(desc(valores))

# A tibble: 11 x 2
# Groups:   titulos [11]
  titulos      valores
  <fct>         <dbl>
1 alcachofra     100
2 alho            93
3 chuchu         79
4 pimenta        74
5 cebola         68
6 couve          61
7 cenoura        45
8 pimentão       39
9 pepino         28
10 alface        25
11 rúcula        12
```

Poderíamos refazer o gráfico de barras ordenado:

```
ggplot(dados,aes(x= reorder(titulos,-valores),valores))+
geom_bar(position="dodge",stat ="identity")+
theme(axis.title.x=element_blank()+
  geom_text(aes(label=valores), position=position_dodge(width=0.9),
    vjust=-0.25, size=2.5)#+
```

```
# Insere os rotulos acima das barras de tamanho 2.5
#ggtitle("Gráfico de barras ordenado para valores e títulos")+
# theme(plot.title = element_text(hjust = 0.5))# Centraliza o titulo do grafico acima
```

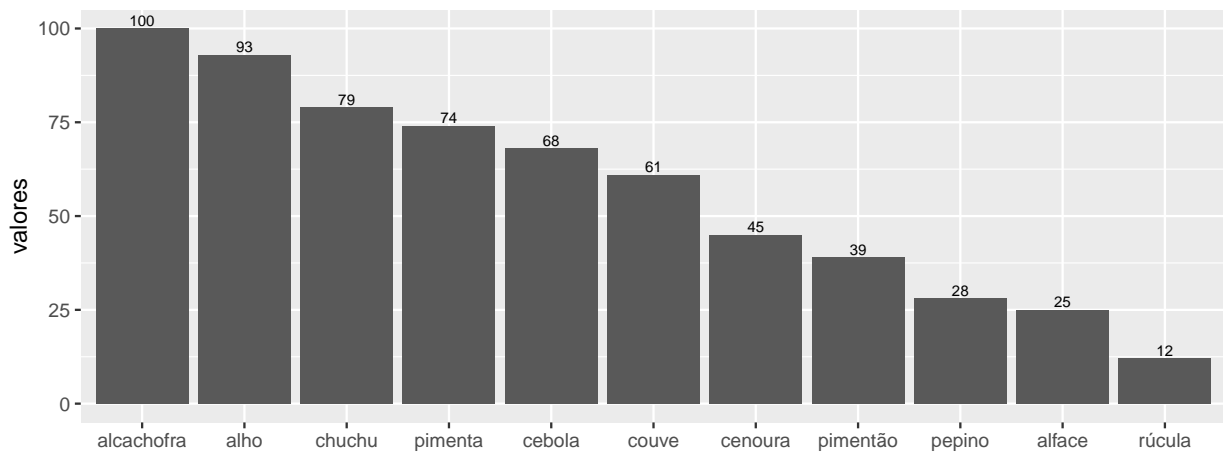


Figura 2: Gráfico de barras ordenado para valores e títulos

1.1.3 Liste a soma e a média aritmética dessa matriz

Para somarmos e também para obtermos a média aritmética do vetor de valores simplesmente fazemos

```
sum(dados$valores) # Soma


[1] 624

mean(dados$valores) # Media aritmetica

[1] 56.72727

round(mean(dados$valores),0) # Valores arredondados pois os valores sao discretos (inteiros)

[1] 57
```

Ou então aqui pelo \LaTeX pelo pacote de integração com o  knitr, simplesmente 624

Para inserir o valor da soma no texto utilize o comando:

o valor da soma eh `\ Sexpr{sum(dados$valores)}` ...

a média é de `\ Sexpr{mean(dados$valores)}` ...

Lembrando a contrabarra (backslash) precisa ser junto ao comando Sexpr.

1.1.4 Liste o produto dos elementos da matriz

Vamos obter o multiplicatório do vetor de valores, que é o mesmo que:

$$\prod_{i=1}^{11} \text{valores}_i = \text{valor}_1 \times \text{valor}_2 \times \dots \times \text{valor}_{11}$$

ou seja, a notação $\prod_{i=1}^{11}$ lê-se que do primeiro $i = 1$, índice de contagem da multiplicação até o 11º multiplique a variável valores.

```
print(prod(dados$valores), digits = 20) # Especifico o numero de digitos para numeros grandes
[1] 3324580966324800000
```

Ou então podemos multiplicar o valor do alface com o da cenoura fazendo:

```
alfacexcenoura<-dados[1,2]*dados[2,2]

alfacexcenoura

[1] 1125
```

Sendo que a instrução [1,2] significa primeira linha, da segunda coluna.

Se quiséssemos fazer o produto de todos os elementos do dataset, faríamos:

alface ×25, cenoura ×45, pepino ×28, chuchu ×79 ...

Ou seja, o vetor de nomes (títulos) multiplicado pelo vetor numérico de valores.

```
library(tidyr)
library(tidyverse)

multiplica.elementos<-paste(dados$titulos,"*",dados$valores)

multiplica.elementos # Mostra o resultado

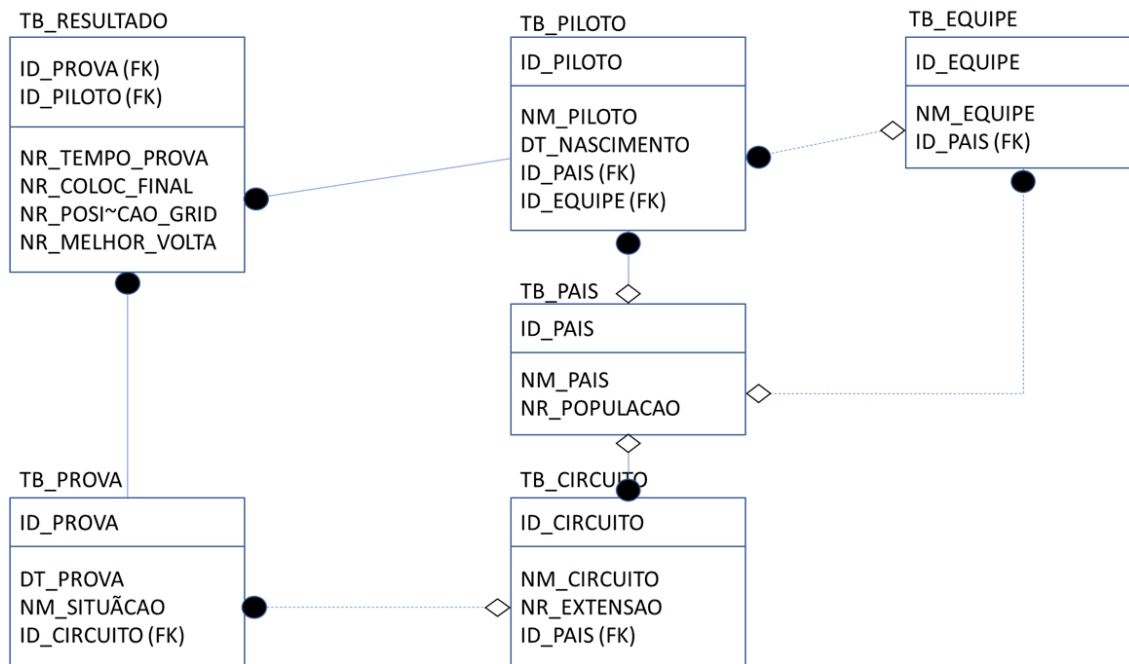
[1] "alface * 25"      "cenoura * 45"      "pepino * 28"      "chuchu * 79"
[5] "pimenta * 74"     "couve * 61"        "rúcula * 12"      "cebola * 68"
[9] "alho * 93"        "pimentão * 39"     "alcachofra * 100"

as.tibble(multiplica.elementos) # Mostra o resultado como um vetor

# A tibble: 11 x 1
  value
  <chr>
1 alface * 25
2 cenoura * 45
3 pepino * 28
4 chuchu * 79
5 pimenta * 74
6 couve * 61
7 rúcula * 12
8 cebola * 68
9 alho * 93
10 pimentão * 39
11 alcachofra * 100
```

2 Atividade II

Você trabalha há algum tempo como analista de dados numa empresa contratada pela Formula-1. O cientista de dados chefe pediu para que você, baseando-se no diagrama E-R abaixo, elabore os seguintes comandos SQL para satisfazer seus pedidos:



Obs.: (a dica aqui é que, para elaborar os comandos SQL, você não precisa acessar a base de dados, já que possui a modelagem (diagrama) que indica as chaves PK e FK, bem como os atributos de cada tabela e seus relacionamentos.)

1. Exibir as equipes que são do país “Alemanha”.
2. Exibir quais os pilotos que são do mesmo país do piloto “Max Chilton”.
3. Selecionar os pilotos que são da equipe “McLaren-Mercedes”.

2.1 Resolução

Poderíamos dar as instruções diretamente do **R** usando o SQLite, carregando primeiro os pacotes:

```
library(RSQLite)
library(DBI)
```

Como aqui não há necessidade de usarmos os comandos para conectarmos ao banco de dados estruturado, (deixo o chunk somente com os comentários de comandos)

```
1 conn <- dbConnect(SQLite(), "test.db")
2
3 dbListTables(conn) # Mostra todas as bases de dados disponiveis
4
5 conn
```

2.1.1 Exibir as equipes que são do país “Alemanha”.

Vamos exibir com comandos do SQLite quais as equipes são da Alemanha:

```
1
2 query <- "SELECT NM_EQUIPE
3           FROM TB_EQUIPE
4           INNER JOIN TB_PAIS ON NM_PAIS=Alemanha"
5
6 result <- dbGetQuery(conn, query)
7 str(result)
```

2.1.2 Exibir quais os pilotos que são do mesmo país do piloto “Max Chilton”

```
1
2 query<-"SELECT NM_PILOTO
3           FROM TB_PILOTO
4           INNER JOIN TB_PAIS
5           WHERE NM_PILOTO='Max Chilton'"
6
7 result <- dbGetQuery(conn, query)
8
9 result
```

2.1.3 Selecionar os pilotos que são da equipe “McLaren-Mercedes”

```
1
2 query<-"SELECT NM_PILOTO FROM TB_PILOTO
3           INNER JOIN TB_EQUIPE ON NM_EQUIPE_id='McLaren-Mercedes'"
4
5 result <- dbGetQuery(conn, query)
6
7 result
```

3 Atividade III

Você trabalha numa empresa de manipulação de dados e como analista de dados está acostumado a receber os mais diversos tipos de pedidos, desde procedures em SQL, rotinas de ETL e, também, como nesse caso, conjuntos de dados para gerar gráficos e entregar à área de marketing que está desenvolvendo um importante projeto, comparando faixas de markup entre produtos de concorrentes. Então, eles lhe passaram dois conjuntos de dados a saber:

$x = (5, 5, 5, 13, 7, 11, 11, 9, 8, 9)$

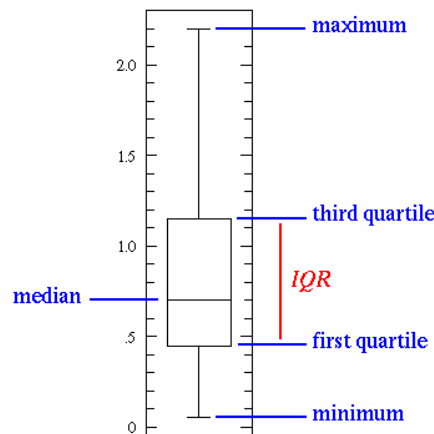
$y = (11, 8, 4, 5, 9, 5, 10, 5, 4, 10)$

Ambos possuem 10 elementos cada, representados por 2 vetores.

Pede-se que você prepare e apresente a visualização:

1. Utilize a linguagem R para plotar X e Y em gráficos diferentes utilizando BOXPLOT.
2. Usando R, plote a comparação de X e Y no mesmo gráfico utilizando BOXPLOT.

Obs.: como dica importante, aqui está o entendimento desse importante tipo de gráfico BOXPLOT que serve para a exibição de distribuição, recorde aqui a definição:



3.1 Resolução

Começo declarando os conjuntos de dados

```
x <- c(5,5,5,13,7,11,11,9,8,9)
```

```
y <- c(11,8,4,5,9,5,10,5,4,10)
```

```
xy<-data.frame(x,y)
```

```
xy
```

	x	y
1	5	11
2	5	8
3	5	4
4	13	5
5	7	9

```
6  11  5
7  11 10
8   9  5
9   8  4
10  9 10
```

3.1.1 Utilize a linguagem R para plotar X e Y em gráficos diferentes utilizando BOXPLOT.

```
require(gridExtra)

x<-ggplot(data=xy,aes(x="",y=x))+
  geom_boxplot()+
  theme(axis.title.x=element_blank())

y<-ggplot(data=xy,aes(x="",y=y))+
  geom_boxplot()+
  theme(axis.title.x=element_blank())

grid.arrange(x, y, ncol=2)
```

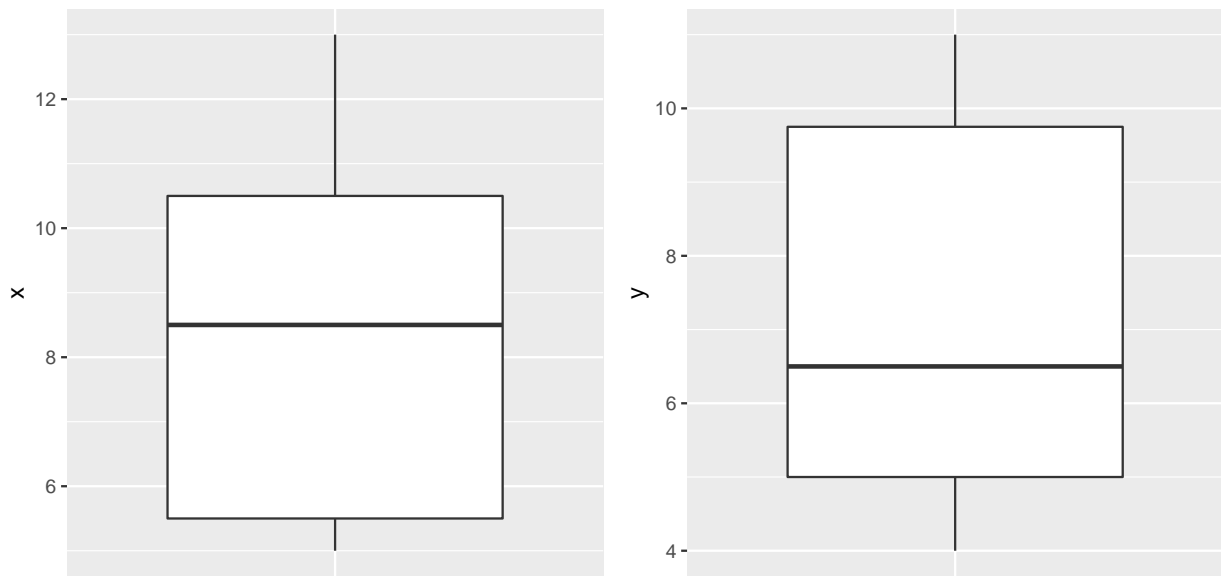


Figura 3: Boxplots x e y

O que também pode ser feito sem o pacote ggplot2:

```
par(mfrow=c(1,2))

boxplot.x<-boxplot(xy$x, main="Boxplot de X")

boxplot.y<-boxplot(xy$y, main="Boxplot de Y")
```

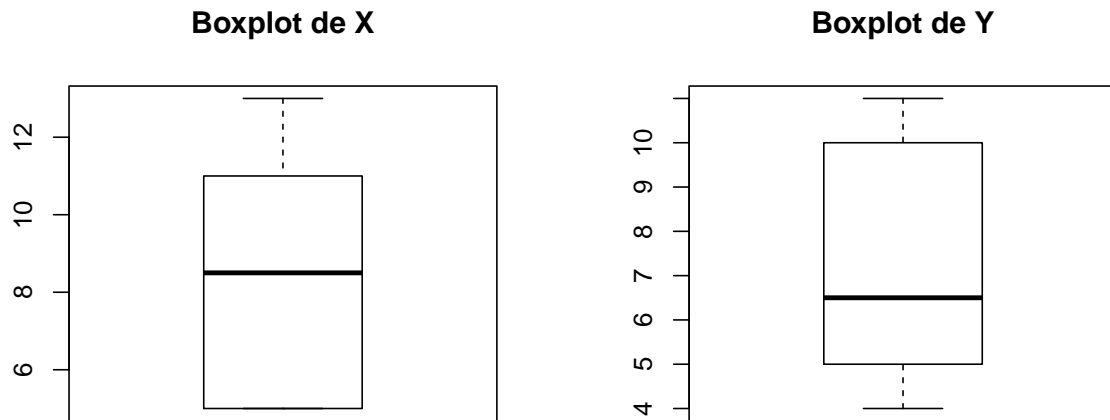


Figura 4: Boxplots x e y

A leitura dos boxplots pode ser resumida com o sumário estatístico para a mediana, máximo, mínimo, primeiro e terceiro quartil:

```
summary(xy)
```

x	y
Min. : 5.0	Min. : 4.00
1st Qu.: 5.5	1st Qu.: 5.00
Median : 8.5	Median : 6.50
Mean : 8.3	Mean : 7.10
3rd Qu.: 10.5	3rd Qu.: 9.75
Max. : 13.0	Max. : 11.00

A amplitude interquartil, pode ser facilmente obtida:

```
IQR.x<-quantile(xy$x,probs=0.75)-quantile(xy$x,probs=0.25)
```

```
IQR.y<-quantile(xy$y,probs=0.75)-quantile(xy$y,probs=0.25)
```

```
tibble(IQR.x,IQR.y)
```

```
# A tibble: 1 x 2
```

```
  IQR.x IQR.y
```

```
<dbl> <dbl>
```

```
1      5  4.75
```


3.1.2 Usando R, plote a comparação de X e Y no mesmo gráfico utilizando BOXPLOT.

```
ggplot(xy, aes(x=x, y=y)) +  
geom_boxplot()
```

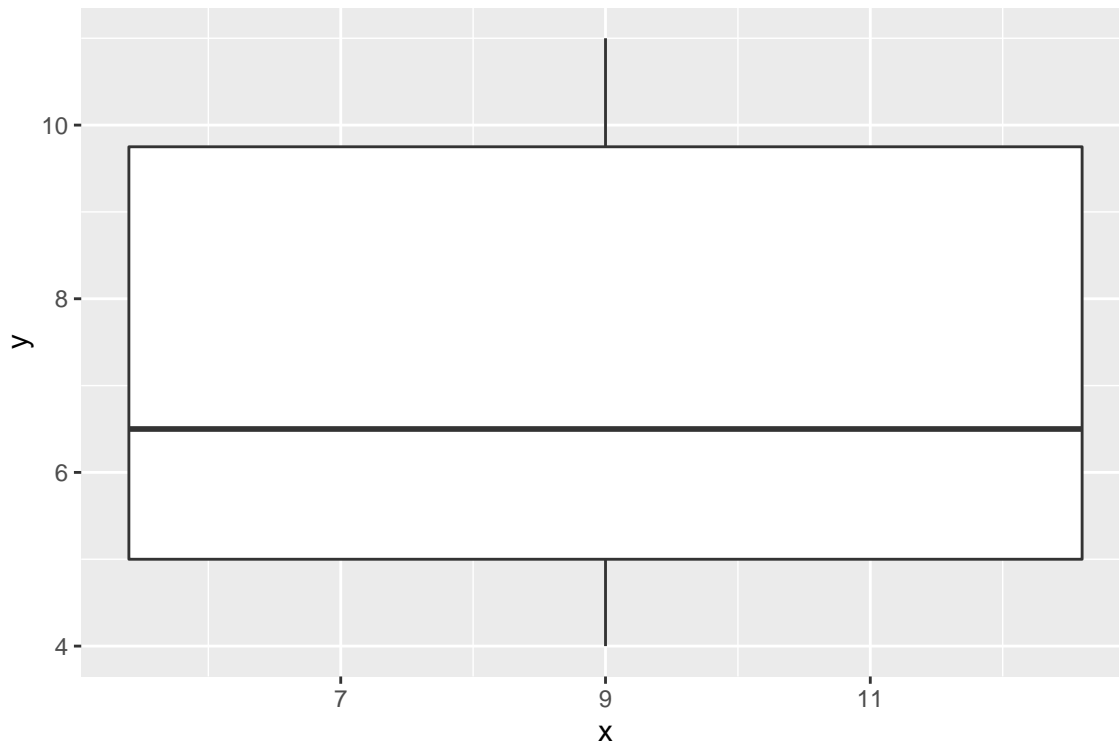


Figura 5: Boxplot x e y

Para facilitar a visualização das distribuições plotamos as densidades kernel, que é uma forma não paramétrica de estimar a função de densidade de probabilidade de uma variável aleatória:

```
par(mfrow=c(1,2))  
  
densx <- density(xy$x, kernel="epanechnikov")  
  
densy <- density(xy$y, kernel="epanechnikov")  
  
densidades <- c(densx, densy)  
  
plot(densx)  
  
plot(densy)
```

```
ensity.default(x = xy$x, kernel = "epanechniensity.default(x = xy$y, kernel = "epanechni
```

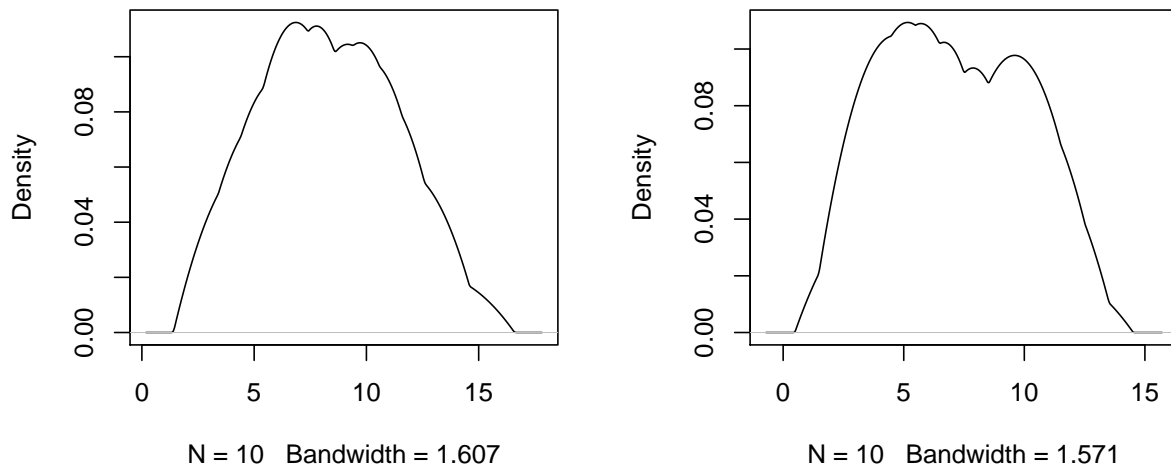


Figura 6: Densidades Kernel de X e de Y

4 Atividade IV

Como cientista de dados jr, trabalhando no IBGE, foi lhe passada uma amostra de dados retirada de um SGBD do CENSO, na forma de uma matriz, onde a primeira linha contém o nome dos atributos para facilitar sua identificação e, após isso, 6 linhas contendo alguns dados amostrais de uma população qualquer.

Segue a tabela:

Id	Turma	Sexo	Idade	Altura	Peso	Filhos	Fuma	Toler	Exerc	Cine	OpCine	TV	OpTV
1	A	F	17	1.6	60.5	2	NAO	P	0	1	B	16	R
2	A	F	18	1.69	55	1	NAO	M	0	1	B	7	R
3	A	M	18	1.85	72.8	2	NAO	P	5	2	M	15	R
4	A	M	25	1.85	80.9	2	NAO	P	5	2	B	20	R
5	A	F	19	1.58	55	1	NAO	M	2	2	B	5	R
6	A	M	19	1.76	60	3	NAO	M	2	1	B	2	R

Seu gerente pediu para que você, a partir desses dados, lhe passe, utilizando a linguagem R, as seguintes respostas:

1. Qual a média IMC dessa pequena amostra?
2. Boxplot comparando a peso M, peso Fe peso geral no mesmo gráfico.
3. Boxplot da altura geral.

Obs.: dica, a formula do IMC é a sigla para Índice de Massa Corporal, utilizada para classificar o peso do indivíduo em relação à sua altura e assim indicar se está dentro do peso ideal, acima ou abaixo do peso normal.

$$\text{IMC} = \text{peso} \div (\text{Altura} \times \text{Altura})$$

Há uma tabela para sabermos se o indivíduo está dentro ou fora dos padrões, mas isso não é importante nesse momento. Obter a medida sim!

4.1 Resolução

Primeiramente carrego a tabela para dentro do R: ([você baixar a tabela aqui](#))

```
tabela<-read.csv(file="AtividadeIV.csv",head=TRUE,sep=";")

str(tabela) # Checo se ele leu corretamente o dataset

'data.frame': 6 obs. of 14 variables:
 $ Id      : int  1 2 3 4 5 6
 $ Turma   : Factor w/ 1 level "A": 1 1 1 1 1 1
 $ Sexo    : Factor w/ 2 levels "F","M": 1 1 2 2 1 2
 $ Idade   : int  17 18 18 25 19 19
 $ Altura  : num  1.6 1.69 1.85 1.85 1.58 1.76
 $ Peso    : num  60.5 55 72.8 80.9 55 60
 $ Filhos  : int  2 1 2 2 1 3
 $ Fuma    : Factor w/ 1 level "NAO": 1 1 1 1 1 1
 $ Toler   : Factor w/ 2 levels "M","P": 2 1 2 2 1 1
 $ Exerc   : int  0 0 5 5 2 2
 $ Cine    : int  1 1 2 2 2 1
 $ OpCine  : Factor w/ 2 levels "B","M": 1 1 2 1 1 1
 $ TV      : int  16 7 15 20 5 2
 $ OpTV    : Factor w/ 1 level "R": 1 1 1 1 1 1
```

4.1.1 Qual a média IMC dessa pequena amostra?

Primeiramente, precisamos criar a variável IMC:

```
tabela<-tabela%>%
  mutate(IMC=Peso/Altura^2) #variavel IMC criada e inserida na tabela

str(tabela)

'data.frame': 6 obs. of 15 variables:
 $ Id      : int  1 2 3 4 5 6
 $ Turma   : Factor w/ 1 level "A": 1 1 1 1 1 1
 $ Sexo    : Factor w/ 2 levels "F","M": 1 1 2 2 1 2
 $ Idade   : int  17 18 18 25 19 19
 $ Altura  : num  1.6 1.69 1.85 1.85 1.58 1.76
 $ Peso    : num  60.5 55 72.8 80.9 55 60
 $ Filhos  : int  2 1 2 2 1 3
 $ Fuma    : Factor w/ 1 level "NAO": 1 1 1 1 1 1
 $ Toler   : Factor w/ 2 levels "M","P": 2 1 2 2 1 1
 $ Exerc   : int  0 0 5 5 2 2
 $ Cine    : int  1 1 2 2 2 1
 $ OpCine  : Factor w/ 2 levels "B","M": 1 1 2 1 1 1
 $ TV      : int  16 7 15 20 5 2
 $ OpTV    : Factor w/ 1 level "R": 1 1 1 1 1 1
 $ IMC     : num  23.6 19.3 21.3 23.6 22 ...
```

Calculamos a média:

```
mean(tabela$IMC)

[1] 21.53335
```

Esse resultado da média aponta para esse grupo de indivíduos dessa amostra como dentro dos padrões médios do IMC;

Caso você deseje saber se está dentro ou fora dos padrões, segue a tabela de classificação do IMC:

Tabela 1: Tabela de classificação do IMC

Resultado	Situação
Abaixo de 17	Muito abaixo do peso
Entre 17 e 18,49	Abaixo do peso
Entre 18,50 e 24,99	Peso normal
Entre 25 e 29,99	Acima do peso
Entre 30 e 34,99	Obesidade I
Entre 35 e 39,99	Obesidade II (severa)
Acima de 40	Obesidade III (mórbida)

Fonte: [Wikipedia](#)

Por mais que não se considere importante, sabermos se existe algum indivíduo dentro ou fora dos padrões, podemos criar uma variável(coluna) que mostra isso na tabela:

```
tabela<-tabela%>%
  mutate(Situacao = case_when(
    IMC <= 17 ~ "Muito abaixo do peso",
    IMC >= 17 & IMC <= 18.49 ~ "Abaixo do peso",
    IMC >= 18.5 & IMC <= 24.99 ~ "Peso Normal",
    IMC >= 25 & IMC <= 29.99 ~ "Acima do peso",
    IMC >= 30 & IMC <= 34.99 ~ "Obesidade I",
    IMC >= 35 & IMC <= 39.99 ~ "Obesidade II (severa)",
    IMC > 40 ~ "Obesidade III (mórbida)"))
```

Vamos dar mais uma olhada no dataset:

```
# Suprimo algumas colunas pra printar a tabela com o comando do R
```

```
tabelaselecionada<-tabela%>%
  select(Sexo,Idade,Altura,Peso,IMC,Situacao)
```

```
tabelaselecionada
```

	Sexo	Idade	Altura	Peso	IMC	Situacao
1	F	17	1.60	60.5	23.63281	Peso Normal
2	F	18	1.69	55.0	19.25703	Peso Normal
3	M	18	1.85	72.8	21.27100	Peso Normal
4	M	25	1.85	80.9	23.63769	Peso Normal
5	F	19	1.58	55.0	22.03173	Peso Normal
6	M	19	1.76	60.0	19.36983	Peso Normal

4.1.2 Boxplot comparando a peso M, peso Fe peso geral no mesmo gráfico.

```
ggplot(data=tabela,aes(x=Sexo,y=Peso, fill=mean(Peso)))+  
geom_boxplot()+  
geom_jitter(shape=5, position=position_jitter(0.2))
```

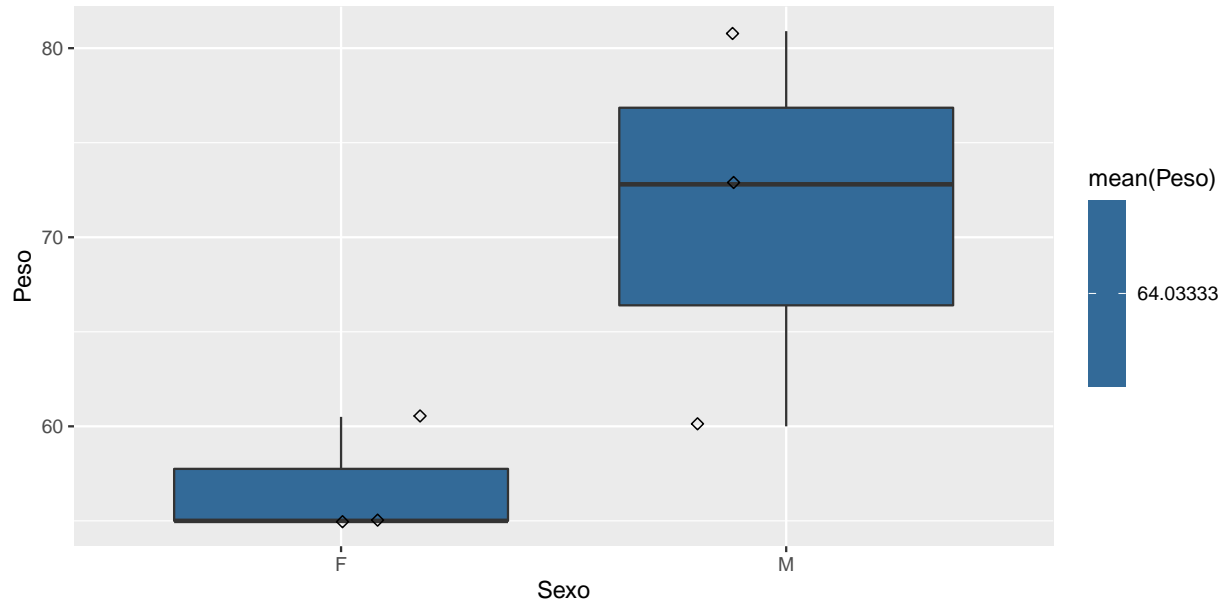


Figura 7: Boxplot comparativo de peso x sexo x peso geral (média(peso))

Ao observarmos as bolinhas geradas pela instrução jitter temos uma boa noção da dispersão dos pesos de uma maneira mais geral.

4.1.3 Boxplot da altura geral

```
ggplot(tabela,aes(x="",y=Altura))+  
geom_boxplot()+  
geom_jitter(shape=5, position=position_jitter(0.2))+# jitter para pontos sem sobreposicao  
scale_color_grey()
```

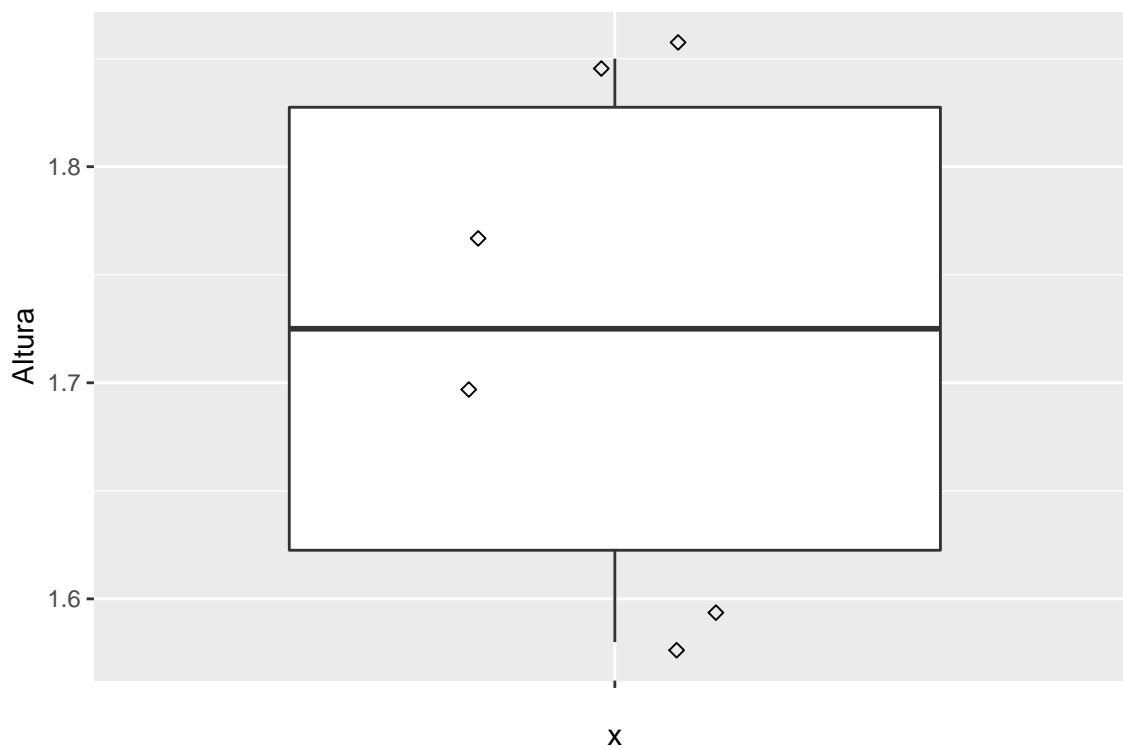


Figura 8: Boxplot da Altura

Referências

- [1] Gujarati, D.,N. **Basic Econometrics**, fourth edition, McGraw-Hill/Irwin, 2003.
- [2] Overleaf: Online \LaTeX Editor. Disponível em [Overleaf.com](https://overleaf.com)
- [3] Ozon, R.,H. **Modelagem de Volatilidade em Séries Financeiras**, *in* Wiki do Laboratório de Geoestatística e Informação, LEG/UFPR.
- [4] Ozon, R.H. **Análise econométrica em tempo real: Modelos família ARCH para dados financeiros** *in* rhozon.github.io .
- [5] Xie, Y. **Dynamic Documents with R and knitr** 2nd edition, 2015.