UNIVERSIDADE DE SÃO PAULO
Gabarito: MAE0261 – Introdução à Análise de dados

Alex Monito Nhancololo

Critérios usados na correção

A correção da lista baseou-se nos seguintes aspectos: resolução correta dos exercícios e interpretação prática e não técnica dos resultados; organização, limpeza e boa estrutura; justificativas apresentadas sempre que necessárias; símbolos matemáticos e estatísticos escritos adequadamente; uso de régua em trabalhos feitos manualmente, em vez de simples traços à mão livre; presença obrigatória de legenda em todas as tabelas e gráficos; além de clareza e coerência na escrita, com linguagem objetiva e sem ambiguidades.

1 Introdução ao R e RStudio (mini tutorial)

Nota: Abra este documento em um computador e, onde aparecer **Clica aqui**, aproxime o cursor e clique: será aberto um vídeo.

1 O que é R?

R é uma linguagem de programação e um ambiente de software livre usado para analisar dados, criar gráficos e aplicar métodos estatísticos. Foi criado por Ross Ihaka e Robert Gentleman na Universidade de Auckland, Nova Zelândia, em 1993, inspirado na linguagem S. Hoje, é muito utilizado por estatísticos, cientistas de dados e pesquisadores em várias áreas, contando com uma grande comunidade que desenvolve pacotes e amplia suas funcionalidades.

1 O que é RStudio?

RStudio é um programa que facilita o uso do R. Ele oferece uma interface gráfica amigável, onde o usuário pode escrever e executar códigos, visualizar gráficos, organizar arquivos e acompanhar os objetos criados durante a análise. É um Ambiente de Desenvolvimento Integrado (IDE), ou seja, um espaço que reúne várias ferramentas para programar de forma mais prática e organizada. O RStudio foi desenvolvido pela empresa RStudio, PBC (atualmente Posit Software, PBC), fundada por Joseph J. Allaire, e teve sua primeira versão lançada em 2011.

Assista ao vídeo: **Clica aqui para ver o vídeo**. Para mais informações sobre R, você pode ler **Clica aqui** se quiser saber mais sobre R.

2 Instalação

Para usar o RStudio, você deve primeiro instalar o R.

2 Passo 1: Instalar o R

1. Acesse o site oficial do R: **Clica aqui** e assista ao vídeo colocado no final destas instruções.
2. Escolha o link de download para o seu sistema operacional (Linux, MacOS ou Windows) (veja a Figura 1, onde seleciono Windows porque é meu sistema operacional. Se não conhece o seu sistema operacional, veja o vídeo: **Clica aqui**. Se em algum lugar aparecer **Windows**, então você usa **Windows**).
3. Clique em "base"(ou na versão mais recente recomendada) para baixar o instalador principal.
4. Execute o arquivo baixado e siga as instruções, mantendo as configurações padrão.

Assista ao vídeo 1 da Fernanda Peres: **Clica aqui para ver o vídeo**.

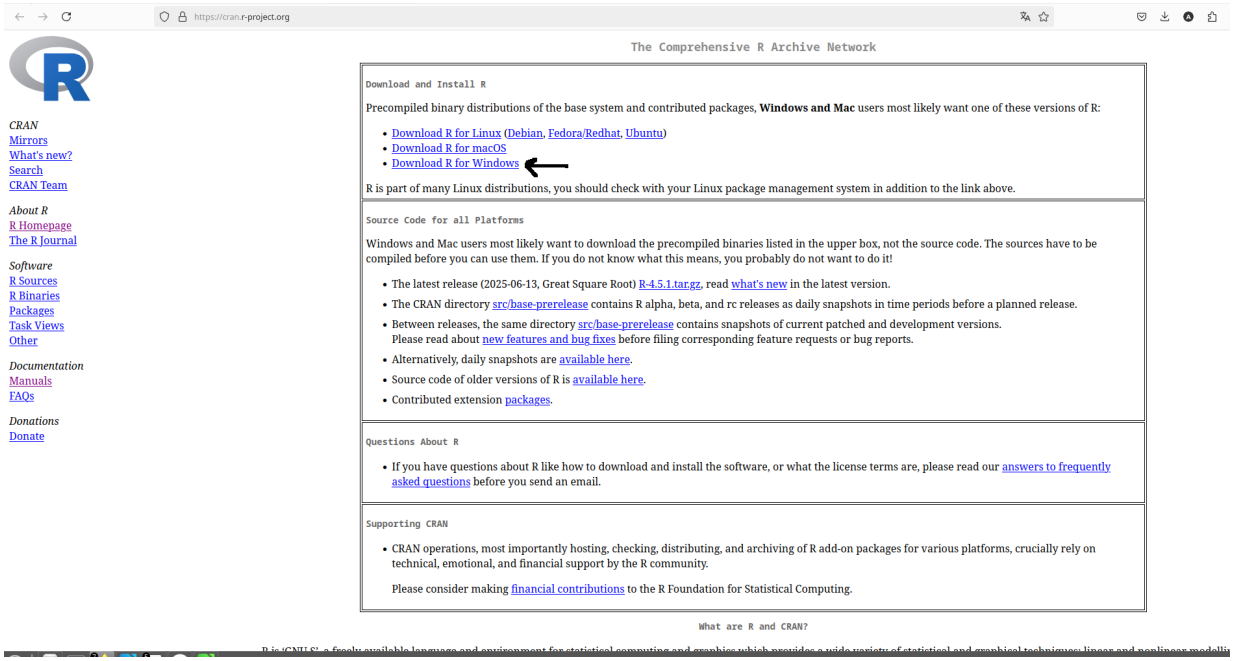


Figura 1: Página de download do R no CRAN.

2 Passo 2: Instalar o RStudio

1. Visite a página de download do RStudio: **Clica aqui**.
2. A versão gratuita "RStudio Desktop"é a recomendada para iniciantes. Clique no botão de download (Figura 2 mostra isso e o vídeo contém o tutorial).
3. O site geralmente detecta seu sistema operacional e sugere o instalador correto. Baixe-o.
4. Execute o instalador e siga as instruções, mantendo as opções padrão.

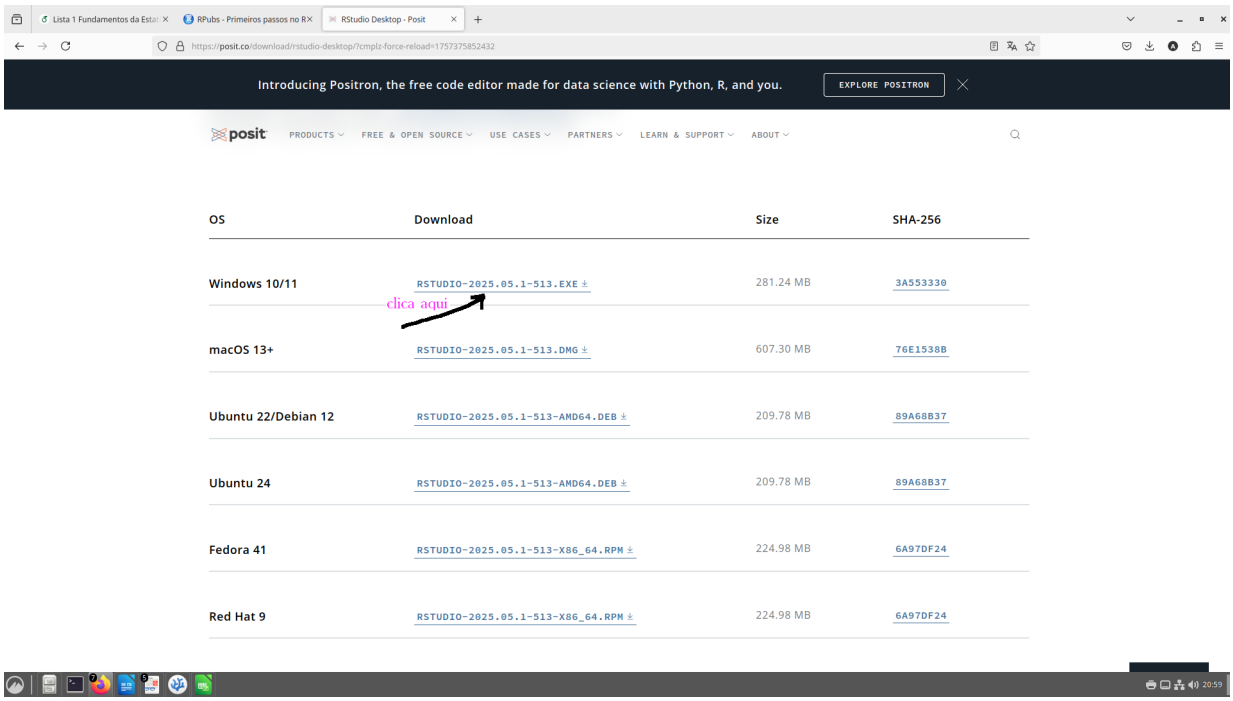


Figura 2: Página de download do RStudio.

Nota: Assim que instalar o R e o RStudio, sempre que quiser fazer análises, use o RStudio clicando nele duas vezes. Se não encontrar o RStudio na área de trabalho, pesquise por **Rstu** em uma das opções abaixo (1 ou 2) mostradas na Figura 3.

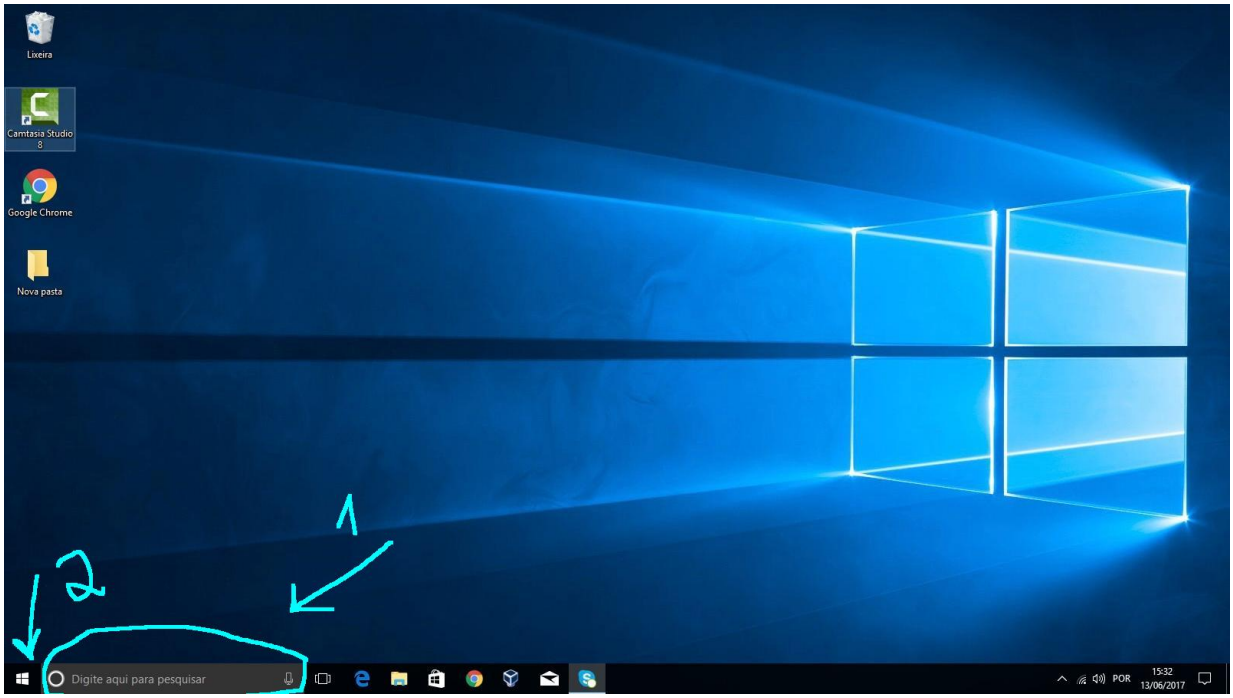


Figura 3: Encontrar aplicativo que sumiu.

Nota: O R diferencia letras minúsculas de maiúsculas (TOME CUIDADO). Para os próximos passos, assista aos vídeos da Fernanda Peres: **Clica aqui** para ver o Vídeo 2 (os demais também são interessantes).

Questões e resolução

Link dos dados **https://hbiostat.org/data/**, base de dados **Diabetes data**.
Questão 1: Apresente medidas descritivas para a variável quantitativa Glicose (stab.glu). Inclua média, mediana, quartis, o número de observações e medidas de variabilidade como o desvio padrão e o erro padrão. Interprete a mediana e o primeiro quartil.

Resolução

1) Medidas Descritivas para Glicose (stab.glu)

Para a variável quantitativa Glicose (stab.glu), calcularíamos as seguintes medidas descritivas:

- Média ($\bar{x}$): A soma de todas as observações x_i (isto é, todos os números da **stab.glu**) dividida pelo número total de observações n (quantidade de números existentes).

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} = \frac{82 + 97 + 92 + 93 + 90 + \ldots + 369 + 89 + 269 + 76 + 88}{403} = \frac{42989}{403} = 106.6725 \approx 106.67$$

- Mediana (Md): O valor que divide os dados ordenados, pelo meio.

$$\text{Mediana} = \begin{cases} x_{(\frac{n+1}{2})}, & \text{se } n \text{ é ímpar} \\ \dfrac{x_{(\frac{n}{2})} + x_{(\frac{n}{2}+1)}}{2}, & \text{se } n \text{ é par} \end{cases}$$

  Como n=403 é impar (se dividir por dois fica um número com vírgula, isto é número decimal), mediana é o número que está na posição (403+1)/2 = 202. Isto é, **mediana** = 89

- Quartis: Dividem um conjunto de dados ordenados em quatro partes iguais, cada uma contendo 25% das observações.

- Desvio padrão (s): Mede a dispersão dos dados em torno da média, isto é, o quão distantes em média os valores x_i estão de $\bar{x}$.

$$s = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}} = \sqrt{\frac{\sum_{i=1}^{403}(82 - 106.67)^2 + \ldots + (88 - 106.67)^2}{403 - 1}} = \sqrt{\frac{1132487}{402}} = \sqrt{2817.131} = 53.07665 \approx 53.08.$$

  Note que 2817.131 é a variância, isto é, desvio padrão é raiz quadrada da variância.

- Erro Padrão (EP): Mede a variabilidade da média amostral ($\bar{x}$) em relação à média populacional (μ), isto é, indica o quão preciso é o valor de $\bar{x}$ como estimador de μ.

$$EP = \frac{s}{\sqrt{n}} = \frac{53.07665}{\sqrt{403}} = \frac{53.07665}{20.1} = 2.64$$

Tabela 1: Estatísticas descritivas da variável Glicose (*stab.glu*)

Variável	n	Média	Desvio padrão	Mín	Q1	Mediana (Q2)	Q3	Máx	Assimetria	Curtose	Erro padrão
stab.glu	403	106.67	53.08	48	81.0	89.0	106.0	385	2.75	8.10	2.64

Nota: Clica aqui para acessar o livro **The Epidemiologist R Handbook**, ele é bom e com várias análises no R/Rstudio.
Interpretação:

- **Mediana (Q2 = 89 mg/dL):** metade (50%) das pessoas tinham glicemia igual ou abaixo de 89 mg/dL.
- **Primeiro quartil (Q1 = 81 mg/dL):** 25% das pessoas tinham glicemia igual ou inferior a 81 mg/dL.

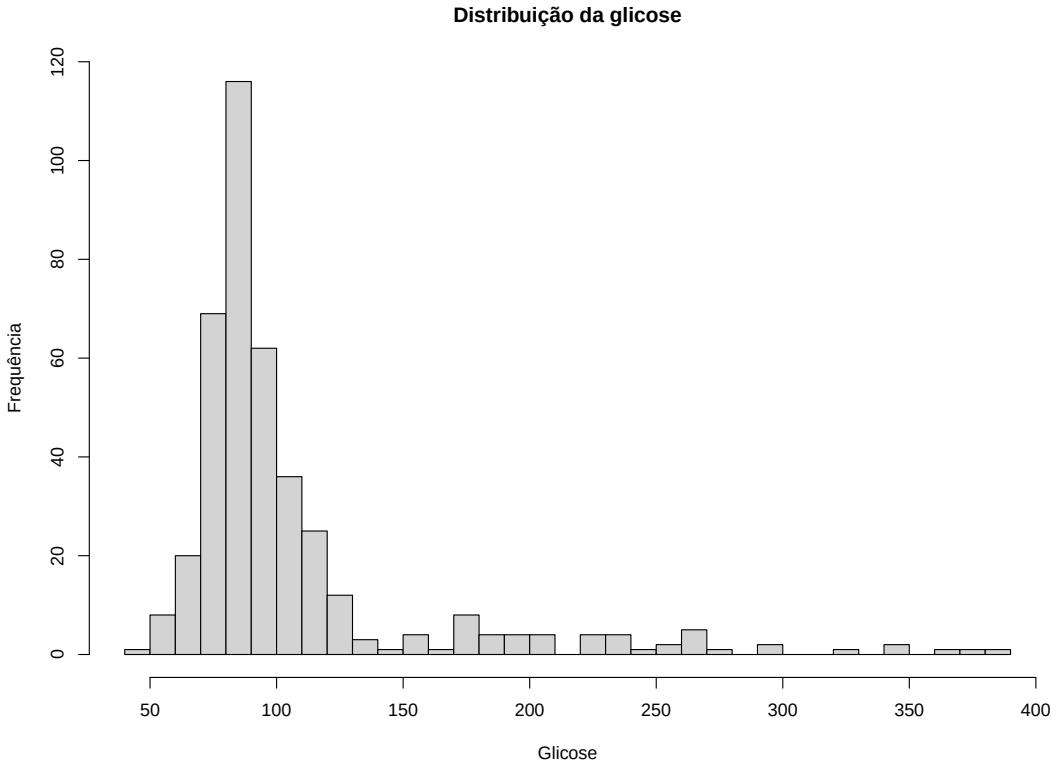
2) Histograma para a Variável Glicose

Um histograma seria construído para visualizar a distribuição dos dados de glicose. Caso feito manualmente os passos seriam:

1. Dividir a amplitude dos dados em intervalos de classe (bins) de igual tamanho.
2. Contar o número de observações (frequência) que caem em cada intervalo.
3. Desenhar barras para cada intervalo, onde a altura da barra corresponde à sua frequência.

Nota: Caso seja do seu interesse fazer manualmente, veja o vídeo: **Clica aqui**

Figura 4: Distribuição dos níveis de Glicose Sanguínea em Pacientes.



Interpretação: na Figura 4, observa-se que a maior parte das pessoas avaliadas apresenta glicose entre 80 e 110 mg/dL, faixa considerada dentro do esperado, mas há um grupo menor com valores mais altos, acima de 150 mg/dL, chegando em alguns casos a ultrapassar 300 mg/dL, o que mostra que embora a maioria esteja com níveis normais, existem indivíduos com glicemia bastante elevada que merecem atenção.
Note que neste gráfico temos assimetria à direita (cauda longa para valores altos), o que pode ser indício de ou valores atípicos (outliers).

3) Boxplot da Glicose por Gênero

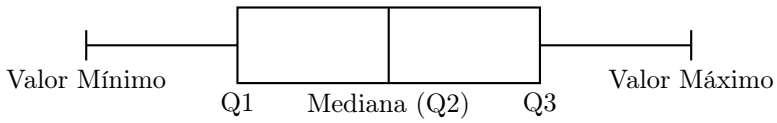
O boxplot é baseado no chamado **resumo de cinco números**: o valor mínimo, o primeiro quartil (Q_1), a mediana (Q_2), o terceiro quartil (Q_3) e o valor máximo.
Nota: Caso opte por construir manualmente, siga os passos abaixo: (1) veja o vídeo **Clica aqui** para manualmente e **clica aqui** e assista vídeo da Fernanda Peres.

Passo 1 - Organize os dados em ordem crescente, do menor para o maior valor. Em seguida, calcule: o valor mínimo, o primeiro quartil (Q_1), a mediana (Q_2), o terceiro quartil (Q_3) e o valor máximo.

Passo 2 - Cálculo do Intervalo Interquartil (IQR): $IQR = Q_3 - Q_1$

Passo 3 - Calcule os limites inferior (LI) e superior (LS): $LI = Q_1 - 1.5 \times IQR$ e $LS = Q_3 + 1.5 \times IQR$
Valores fora desses limites são considerados outliers.

Passo 4: Desenho do Boxplot



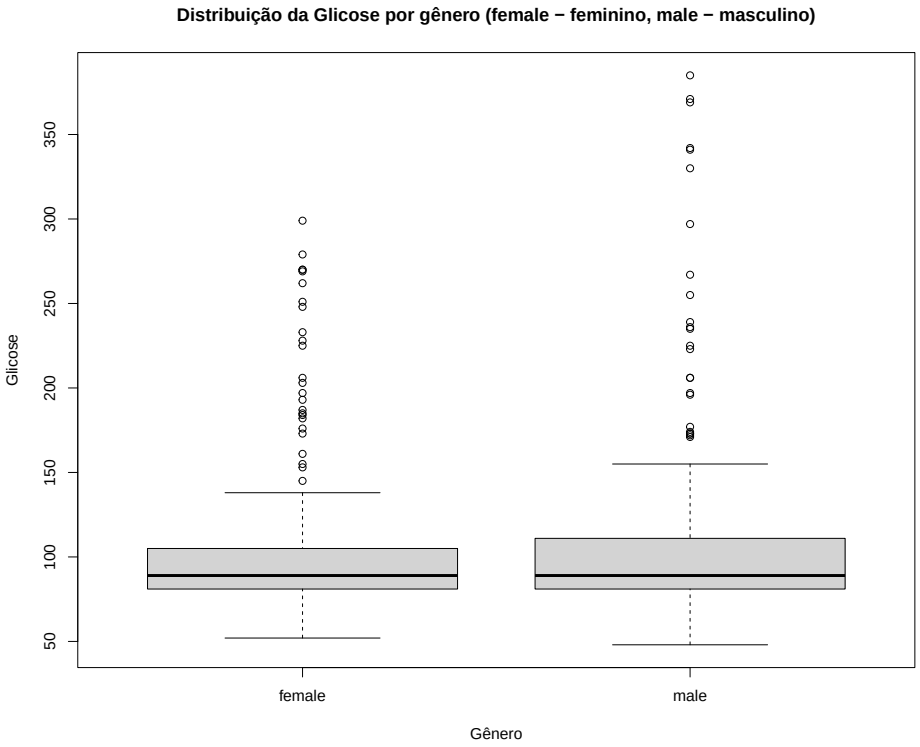


Figura 5: Figura 2: Boxplot dos níveis de Glicose sanguínea por gênero.

Interpretação:
A Figura 5 mostra como os níveis de glicose no sangue variam entre homens e mulheres no estudo. Cada círculo representa uma pessoa, e a "caixa" mostra onde a maior parte das pessoas está concentrada. A mediana (o valor do meio) dos níveis de glicose é parecida para ambos os gêneros. Isso significa que, de modo geral, homens e mulheres têm níveis de glicose similares. Existem pessoas com níveis de glicose muito altos (os círculos acima da caixa) em ambos os grupos. A maioria das pessoas tem glicose em uma faixa mais baixa, mas os homens parecem ter um pouco mais de casos com níveis muito altos do que as mulheres.

4) Exercício 3

Para comparar a distribuição de gênero entre os locais (Buckingham e Louisa), construiríamos uma tabela de contingência. Como os dados não estão disponíveis, segue uma tabela hipotética para ilustrar o cálculo.

Tabela 2: Distribuição de gênero por localização

Característica	**Buckingham** N = 200[1]	**Louisa** N = 203[1]	**Total** N = 403[1]
Gênero			
female	114 (57%)	120 (59%)	234 (58%)
male	86 (43%)	83 (41%)	169 (42%)

[1] n é total das observações existentes, seguido da porcentagem entre os parenteses(%)

Interpretação: Dos 403 participantes incluídos na análise, 234 ($\approx$ 58%) eram mulheres. Separando por local, Buckingham teve 114 mulheres em 200 participantes (57%) e Louisa teve 120 mulheres em 203 participantes (59%). A diferença entre as proporções de mulheres nos dois locais é muito pequena ($\approx$ 2%), o que indica que a composição por sexo é semelhante em Buckingham e em Louisa na amostra observada.

Exercício 5

	Teste positivo	Teste negativo	Total
Doente	231	27	258
Não doente	32	54	86
Total	263	81	344

Medidas de concordância

- **Sensibilidade (s):** Probabilidade de o teste ser positivo dado que a pessoa está doente.

$$\text{s} = \text{P(Teste positivo}| \text{ Doente)} = \frac{\text{total (Teste positivo} \cap \text{Doente)}}{\text{total Doente}} = \frac{231}{258} \approx 0{,}8953 = 89,53\%$$

  Interpretação: O exame tem uma alta sensibilidade. Isso significa que ele é muito eficaz em detectar a doença quando ela está presente. Cerca de 89,5% das pessoas doentes receberão um resultado positivo, o que é excelente caso o objectivo seja detectar se existe a doença.

- **Especificidade (e):** O exame tem uma capacidade limitada de identificar corretamente as pessoas que não têm a doença. Uma especificidade de 62,8% implica uma taxa de falsos positivos de 37,2% (1-0,6279). Ou seja, mais de um terço das pessoas saudáveis que fazem o teste receberão um resultado positivo (alarme falso). Isto é, é meio ruim se o objetivo é identificar se não existe a doença.

$$\text{e} = \text{P(Teste negativo} \mid \text{Não Doente)} = \frac{\text{total (Teste negativo} \cap \text{Não Doente)}}{\text{total Não Doente}} = \frac{54}{86} \approx 0{,}6279 = 62,79\%$$

  Interpretação: O teste identifica corretamente 62,79% dos indivíduos saudáveis.

Probabilidade de estar doente dado teste positivo:
Seja p a prevalência da doença.

$$\begin{aligned}\text{P(Doente} \mid \text{Teste Positivo)} &= \frac{\text{P(Doente} \cap \text{Teste Positivo)}}{\text{P(Teste Positivo)}} = \frac{\text{P(Teste Positivo} \mid \text{Doente)} \cdot \text{P(Doente)}}{\text{P(Teste Positivo} \mid \text{Doente)} + \text{ P(Teste Positivo} \mid \text{Não doente)}} \\ &= \frac{s \cdot p}{s \cdot p + (1-e) \cdot (1-p)}, \text{ em que } \mathbf{s} \text{ é sensibilidade e, } \mathbf{e} \text{ é especificidade calculados acima.} \\ &= \frac{0{,}8953 \cdot p}{0{,}8953 \cdot p + (1-0{,}6279) \cdot (1-p)}, \text{ em que } p \text{ foi fornecido no exercício}\end{aligned}$$

- Para prevalência da doença igual a 25% ($p = 0{,}25$):

$$\begin{aligned}\text{P(Doente} \mid \text{Teste Positivo)} &= \frac{0{,}8953 \cdot p}{0{,}8953 \cdot p + (1-0{,}6279) \cdot (1-p)} = \frac{0{,}8953 \cdot 0{,}25}{0{,}8953 \cdot 0{,}25 + (1-0{,}6279) \cdot (1-0{,}25)} \\ &= \frac{0{,}223825}{0{,}5029} \approx 0{,}4451 = 44,51\%\end{aligned}$$

  Interpretação: Com prevalência de 25%, a probabilidade de estar doente após um teste positivo é de aproximadamente 44,51%.

- Para prevalência de 75% ($p = 0{,}75$):

$$\begin{aligned}\text{P(Doente} \mid \text{Teste Positivo)} &= \frac{0{,}8953 \cdot p}{0{,}8953 \cdot p + (1-0{,}6279) \cdot (1-p)} = \frac{0{,}8953 \cdot 0{,}75}{0{,}8953 \cdot 0{,}75 + (1-0{,}6279) \cdot (1-0{,}75)} \\ &= \frac{0{,}671475}{0{,}7645} \approx 0{,}8783 = 87,83\%\end{aligned}$$

  Interpretação: Com prevalência de 75%, a probabilidade de estar doente após um teste positivo é de aproximadamente 87,83%.

Exercício 6

Seja X o número de canhotos em uma amostra de 10 pessoas. Temos $X \sim \text{Bin}(n = 10, p = 0.1)$, em que $0.1 = 10\%$. A fórmula da probabilidade binomial é:

$$P(X = k) = \binom{n}{k} p^k (1-p)^{n-k}$$

a) Nenhum canhoto ($k = 0$):

$$P(X = 0) = \binom{10}{0} (0.1)^0 (0.9)^{10-0} = \binom{10}{0} (0.1)^0 (0.9)^{10} = 1 \times 1 \times 0.348678 \approx 0.3487 = 34.87\%$$

b) Exatamente 1 canhoto ($k = 1$):

$$P(X = 1) = \binom{10}{1}(0.1)^1(0.9)^{10-1} = \binom{10}{1}(0.1)^1(0.9)^9 = 10 \times 0.1 \times 0.38742 \approx 0.3874 = 38.74\%$$

c) Exatamente 2 canhotos ($k = 2$):

$$P(X = 2) = \binom{10}{2}(0.1)^2(0.9)^{10-2} = \binom{10}{2}(0.1)^2(0.9)^8 = \frac{10 \times 9}{2} \times 0.01 \times 0.430467 \approx 45 \times 0.00430467 \approx 0.1937 = 19.37\%$$

d) Mais que 2 canhotos ($k > 2$):

$$P(X > 2) = 1 - P(X \leq 2) = 1 - [P(X = 0) + P(X = 1) + P(X = 2)]$$

$$P(X > 2) = 1 - (0.3487 + 0.3874 + 0.1937) = 1 - 0.9298 = 0.0702 = 7.02\%$$

Exercício 7

Seja $X \sim \mathrm{N}(\mu = 10, \sigma = 1)$. Usamos a transformação para a normal padrão $Z = \frac{X-\mu}{\sigma}$.
a) $P(X > 12)$**:**

$$Z = \frac{12 - 10}{1} = 2$$

$$P(X > 12) = P(Z > 2) = 1 - P(Z \leq 2)$$

Da tabela da normal padrão, $P(Z \leq 2) \approx 0.9772$.

$$P(X > 12) = 1 - 0.9772 = 0.0228$$

b) $P(8 < X < 12)$**:**

$$Z_1 = \frac{8 - 10}{1} = -2 \quad \text{e} \quad Z_2 = \frac{12 - 10}{1} = 2$$

$$P(8 < X < 12) = P(-2 < Z < 2) = P(Z < 2) - P(Z < -2)$$

$$P(-2 < Z < 2) = 0.9772 - 0.0228 = 0.9544$$

c) x tal que $P(X < x) = 0.02$:
Primeiro, encontramos o valor z na distribuição normal padrão tal que $P(Z < z) = 0.02$. Consultando a tabela Z (ou usando uma calculadora), encontramos $z \approx -2.054$. Agora, convertemos o valor z de volta para a escala de X:

$$x = \mu + z\sigma$$

$$x = 10 + (-2.054) \times 1 = 7.946$$

NOTA: É crucial ter atenção ao usar uma tabela de distribuição normal padrão (tabela Z), pois existem vários tipos. Antes de realizar seus cálculos, verifique sempre a área sob a curva que a sua tabela representa (Clica **aqui** e veja página 12-14). As mais comuns indicam a probabilidade acumulada à esquerda de um valor (P(Z < z)), mas outras podem mostrar a área a partir da média (P(0 < Z < z) **clica aqui**) ou a área na cauda direita (P(Z > z)). Para um guia visual sobre como ler uma tabela, você pode assistir ao vídeo clicando **aqui**.

3 Script R

```r
# ================================================================================
#                          Resolucao Lista 1
# ================================================================================
# Autor: Alex Monito Nhancololo
#
# Este script realiza uma analise exploratoria de dados de um conjunto de dados sobre diabetes.
# Os comentarios abaixo explicam o que cada parte do codigo faz.
# ================================================================================


# ------------------------------------------------------------------------
# Como rodar os comandos no RStudio
# ------------------------------------------------------------------------
# Para executar uma linha de codigo, coloque o cursor sobre ela e pressione CTRL + ENTER.


# ------------------------------------------------------------------------
# Instalacao dos pacotes (so precisa rodar uma unica vez)
# ------------------------------------------------------------------------
# Pacotes sao como "caixas de ferramentas" que adicionam novas funcoes ao R.
# A funcao install.packages() baixa e instala esses pacotes no seu computador.

install.packages("dplyr")      # Essencial para manipulacao de dados (filtros, selecao, etc.).
install.packages("tidyverse")  # Um "pacotao" que inclui o dplyr, ggplot2 e outras ferramentas uteis.
install.packages("flextable")  # Usado para criar tabelas com aparencia profissional.
install.packages("psych")      # Oferece funcoes avancadas de estatisticas descritivas.
install.packages("ggplot2")    # A ferramenta mais poderosa e popular para criar graficos em R.
install.packages("readxl")     # Para ler arquivos do Microsoft Excel (.xls, .xlsx).
install.packages("gtsummary")  # Cria tabelas de resumo estatistico prontas para publicacao.
install.packages("Hmisc")      # Outro pacote com muitas funcoes uteis para analise de dados.
install.packages("gt")         # Outra excelente ferramenta para construir tabelas.
install.packages("kableExtra") # Ajuda a formatar tabelas para diferentes formatos, como PDF.

'Observacao: Uma vez que um pacote e instalado, ele fica no seu computador para sempre.
Voce nao precisa instalar de novo, a menos que atualize o R.'


# ------------------------------------------------------------------------
# Carregando os pacotes (precisa rodar toda vez que iniciar o R)
# ------------------------------------------------------------------------
# A funcao library() "ativa" um pacote que ja foi instalado,
# tornando suas funcoes disponiveis para uso na sessao atual.

library(dplyr)
library(tidyverse)
library(flextable)
library(psych)
library(ggplot2)
library(readxl)
library(gtsummary)
library(Hmisc)
library(gt)
library(kableExtra)


# ------------------------------------------------------------------------
# Diretorio de trabalho
# ------------------------------------------------------------------------
# O diretorio de trabalho e a pasta no seu computador onde o R vai procurar
```

```r
# arquivos para ler e onde ele vai salvar arquivos por padrao.

getwd()
# getwd() significa "get working directory" (obter diretorio de trabalho).
# Essa funcao MOSTRA qual e a pasta atual.

setwd("/home/posmae/amnhancololo/Rede IME/Monitoria Introducao a analise de dados")
# setwd() significa "set working directory" (definir diretorio de trabalho).
# Essa funcao ALTERA a pasta de trabalho para o caminho que voce especificar.
# Dica: Sempre use barras normais "/"  ou "\\"e coloque o caminho entre aspas.


# ------------------------------------------------------------------------
# Importacao dos dados
# ------------------------------------------------------------------------
# Agora vamos carregar nosso conjunto de dados para dentro do R.

dados_diabete <- read_csv("diabetes.csv")
# A funcao read_csv() le um arquivo de texto com valores separados por virgula (.csv).
# Ela transforma o arquivo em um objeto no R chamado "data frame" (uma tabela).
# O simbolo "<-" e o operador de atribuicao. Ele "salva" a tabela lida
# em um objeto com o nome "dados_diabete".


# ------------------------------------------------------------------------
# Explorando os dados ("Conhecendo sua base de dados")
# ------------------------------------------------------------------------
# Antes de fazer qualquer analise, e crucial entender a estrutura e o conteudo dos seus
    dados.

glimpse(dados_diabete)
# A funcao glimpse() (do pacote dplyr) oferece uma "espiada" rapida nos dados.
# Mostra o numero de linhas e colunas, o nome e o tipo de cada coluna,
# e os primeiros valores de cada uma. E otima para um primeiro contato.

# ================= Exerc cio 1: Analise da Variavel "Glicose" (stab.glu)
    =================
# ------------------------------------------------------------------------
# Vamos focar em uma unica variavel para entender suas caracteristicas.

# Seleciona apenas a coluna "stab.glu" e a salva em um novo objeto chamado "Glicose".
Glicose <- dados_diabete["stab.glu"]

# A funcao str() (de "structure") mostra a estrutura do objeto.
# E util para confirmar o tipo de dado (numerico, texto, etc.).
str(Glicose)

# A funcao summary() e uma das mais uteis. Ela calcula as principais estatisticas
# descritivas para cada coluna: minimo, 1   quartil, mediana, media, 3   quartil e maximo.
summary(Glicose)

# A funcao describe() do pacote 'psych' da estatisticas ainda mais detalhadas,
# como desvio padrao, erro padrao, curtose e assimetria.
psych::describe(Glicose)

# A funcao describe() do pacote 'Hmisc' oferece outra visao descritiva.
Hmisc::describe(Glicose)

# A funcao quantile() calcula os percentis que voce especificar.
# Util para entender a distribuicao dos dados em pontos especificos.
quantile(Glicose$stab.glu, probs = c(0.05, 0.10, 0.25, 0.5, 0.75, 0.90, 0.95), na.rm =
    TRUE)
# O "Glicose$stab.glu" acessa a coluna "stab.glu" dentro do objeto "Glicose".
# "na.rm = TRUE" instrui a funcao a ignorar valores ausentes (NA) no calculo.

# A funcao head() mostra as 5 primeiras linhas dos dados.
# Bom para verificar se a importacao ocorreu como esperado.
head(Glicose, 5)

```

```r
# A funcao tail() mostra as 5 ultimas linhas.
tail(Glicose, 5)

# A funcao nrow() conta o numero de linhas (casos/observacoes) no seu conjunto de dados.
n <- nrow(Glicose); n   # Salva o numero de linhas em 'n' e depois o exibe.

# --- Calculos Manuais para Entendimento ---
# Embora o R tenha funcoes prontas, calcular "na mao" ajuda a fixar os conceitos.

# A funcao sum() calcula a soma de todos os valores na coluna.
soma <- sum(Glicose$stab.glu, na.rm = TRUE); soma
# A media e simplesmente a soma dividida pelo numero de observacoes.
media <- soma / n; media

# A mediana e o valor do meio quando os dados estao ordenados.
posicao <- (n + 1) / 2; posicao  # Encontra a posicao do valor do meio.
sort(Glicose$stab.glu)[posicao]   # sort() ordena os dados, e [posicao] seleciona o valor naquela posicao.

# A funcao mean() faz o calculo da media diretamente.
mean(Glicose$stab.glu) # Comparando com o nosso calculo manual.

# Calculo da variancia.
variancia <- sum((Glicose$stab.glu - mean(Glicose$stab.glu))^2) / (n - 1); variancia
# A funcao sqrt() calcula a raiz quadrada (square root). O desvio padrao e a raiz da variancia.
desvio_padrao <- sqrt(variancia); desvio_padrao
# O erro padrao e o desvio padrao dividido pela raiz quadrada do tamanho da amostra.
erro_padrao <- desvio_padrao / sqrt(n); erro_padrao


# ================= Exerc cio 2: Histograma (Visualizando a distribuicao) =================
# A funcao hist() cria um histograma, um grafico de barras que mostra a frequencia
# com que os valores aparecem em diferentes intervalos.
hist(Glicose$stab.glu,
     main = "Distribuicao da Glicose", # Titulo do grafico
     xlab = "Glicose",                 # Rotulo do eixo X
     ylab = "Frequencia",              # Rotulo do eixo Y
     breaks = 40)                      # Numero de barras/intervalos a serem exibidos


# ================= Exerc cio 3: Boxplot (Comparando distribuicoes) =================
# A funcao boxplot() cria um diagrama de caixa. E excelente para comparar
# a distribuicao de uma variavel numerica entre diferentes grupos.
boxplot(stab.glu ~ gender, data = dados_diabete,
        # A formula "stab.glu ~ gender" significa: "mostre a distribuicao de stab.glu para cada grupo em gender".
        main = "Distribuicao da Glicose por genero", # Titulo
        xlab = "Genero", ylab = "Glicose")          # Rotulos dos eixos


# ================= Exerc cio 4: Tabela de frequencias =================
# Aqui usamos o poder do 'tidyverse' e do 'gtsummary' para criar uma tabela profissional.

tabela_genero <- dados_diabete %>%
  # O operador pipe (%>%) pega o resultado do que esta a esquerda e o "passa" como
  # primeiro argumento para a funcao a direita. Ele torna o codigo mais legivel.

  select(location, gender) %>%
  # A funcao select() escolhe apenas as colunas que nos interessam: "location" e "gender".

  tbl_summary(
  # A funcao tbl_summary() (do gtsummary) cria a tabela de resumo.
    by = location, # Agrupa os resultados pela variavel "location".
    statistic = list(all_categorical() ~ "{n} ({p}%)"), # Define o que mostrar: contagem (n) e porcentagem (p).
    missing = "no" # Nao mostra uma linha para dados ausentes.
  ) %>%
```

```
190
191   add_overall() %>%
192   # A funcao add_overall() adiciona uma coluna "Overall" com os totais gerais.
193
194   bold_labels()
195   # A funcao bold_labels() coloca os nomes das variaveis em negrito, melhorando a estetica
        .
196
197 # Finalmente, exibimos a tabela que criamos.
198 tabela_genero
```