Universidade Federal de Minas Gerais - UFMG
Instituto de Ciências Exatas - ICEx
Departamento de Estatística - DEST
Programa de Monitoria de Graduação (PMG)

Análises em Bioestatística Básica:

Uma introdução ao software R

Caio Coelho (Monitor)
Laura Alexandria (Monitora)
Leonardo Paes (Monitor)
Cristiano de Carvalho Santos (Orientador)

2020

Sumário

Capítulo 1

Introdução

O objetivo desta apostila é auxiliar estudantes das áreas de Ciencias Biológicas, Saúde e afins nos seus primeiros passos em análises de dados. A finalidade é passar uma ideia de como resolvemos problemas de Bioestatística utilizando o R, RStudio e o R Commander, que são apresentados a seguir.

1.1 Software R

O **software R** é um ambiente computacional e uma linguagem de programação que vem progressivamente se especializando em manipulação, análise e visualização gráfica de dados. Na atualidade é considerado um dos melhores ambientes computacionais para essa finalidade.

1.1.1 Como instalar o R

Passo a passo para instalar o R:

1. Crie uma pasta no seu computador no local que você preferir com o nome que você escolher. Essa será a pasta onde o R será instalado.
2. faça o download do instalador. Para isso, acesse o link: https://cran.r-project.org/bin/windows/base/ e clique em "Download R x.x.x for Windows", em que x.x.x é o número da versão mais recente disponível.
3. Salve o arquivo na pasta que você criou.
4. Clique no arquivo duas vezes com o botão esquerdo. Ele pedirá para você selecionar a linguagem da instalação. Escolha um idioma e clique em "OK".
5. Clique em "Avançar".
6. Nessa etapa, você precisará escolher a pasta de instalação. Selecione a pasta que você criou.
7. Continue clicando em "Avançar" e, ao fim da instalação, em "Concluir".

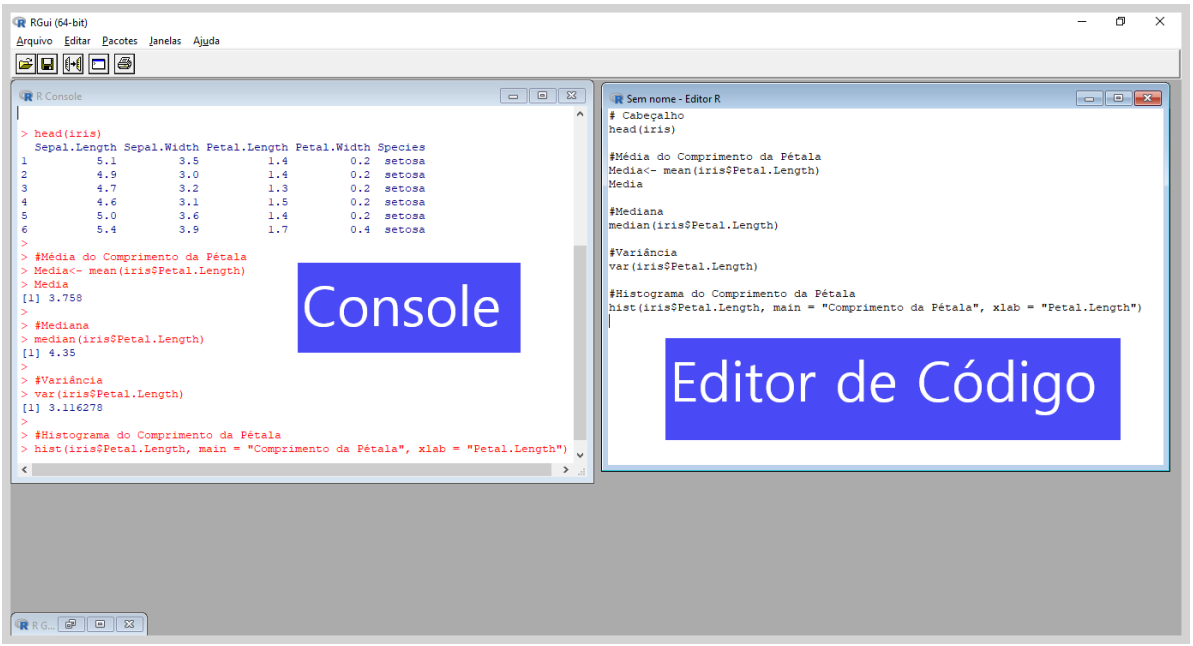
Pronto! O R está instalado no seu computador!

1.1.2 A interface do R

A seguir vemos a interface do R. Na Esquerda temos o Console, onde se visualiza os resultados dos comandos executados. À direita temos o Editor de Código, onde você cria seu script. Ao abrir o R, inicialmente você irá visualizar apenas o console disponível. Para abrir o editor clique em

"Arquivo -> Novo Script", ou "Arquivo -> Abrir Script"

caso já tenha um script de análises anteriores salvo.



Para executar um comando no R, digite o comando desejado no editor e clique em simultaneamente nas teclas “Ctrl” e “R” ou, caso queira executar o comando diretamente no cosole, apenas digite o código e aperte enter. Os comandos executados no console são temporários, já o código que você cria no editor pode ser salvo e reutilizado futuramente. O histórico dos comandos já excutados aparece no console e apertando o botão de seta para cima no teclado "↑ "você consegue acessá-lo para executar os comandos novamente.

Para salvar um script com suas análises clique no ícone em forma de disquete e selecione a pasta na qual deseja salvar seu código. Mais adiante veremos como construir um script.

1.2 RStudio

O **RStudio** é um ambiente de desenvolvimento de códigos em linguagem R. Ele funciona como uma espécie de interface, para utilizá-lo é necessário já possuir o R instalado em seu computador. A principal vantagem de se utilizar o RStudio é que ele permite que o usuário realize suas análises de maneira muito mais organizada e intuitiva.

1.2.1 Como instalar o RStudio

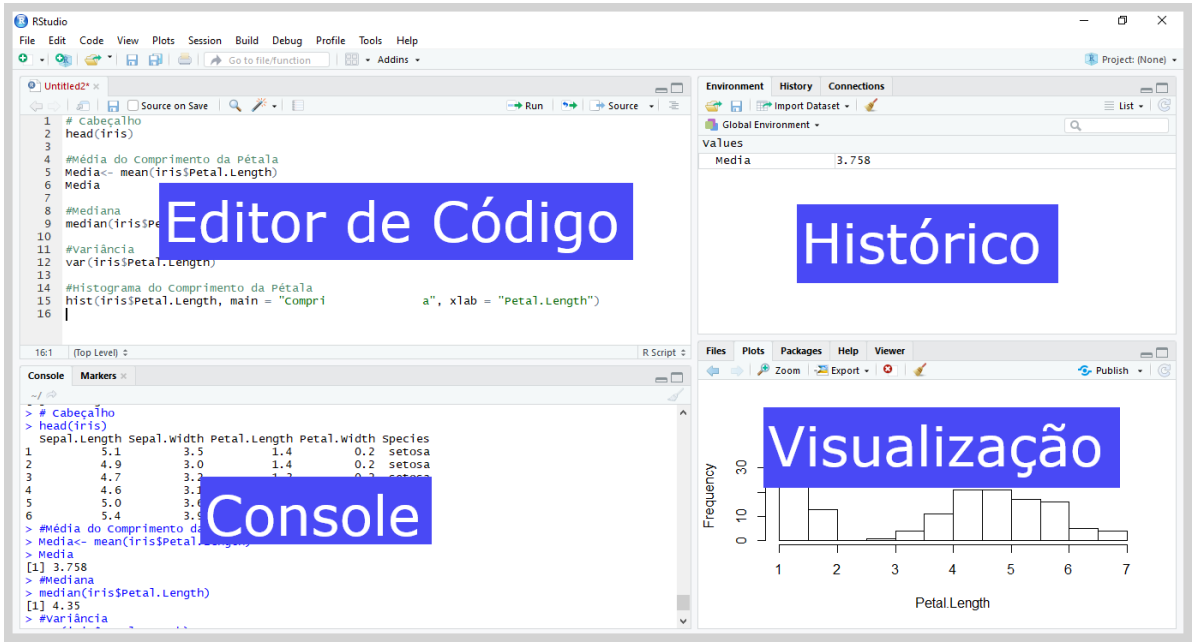
Passo a passo para instalar o RStudio:

1. Acesse o link: https://www.rstudio.com/products/rstudio/download/#download
2. Na lista com título *"Installers for Supported Platforms"* selecione a opção que condiz com o seu sistema operacional e faça o download.
3. Abra o instalador e siga as instruções, clicando no botão "Avançar".

Pronto! O RStudio está instalado no seu computador e pronto para o uso.

1.2.2 A interface do RStudio

Na imagem abaixo vemos a interface do RStudio, onde podemos ter uma ideia geral da aparência do software e entender como usar cada uma de suas partes funcionam.



O Rstudio contém:

- **Editor de Código:** Nessa parte você escreve os comandos que deseja realizar no R, é onde você cria seu *script*. Para processar o seu código você pode selecionar a linha com o comando e apertar simultaneamente as teclas "Ctrl" e "Enter" ou clicar no botão "Run", que aparece no editor.
- **Console:** É onde vemos os resultados dos comandos que executamos, também chamados de *outputs*. Você também pode executar partes do seu código diretamente no console, porém os comandos não ficam salvos, são apenas temporários.
- **Histórico:** É a parte do R voltada para a organização das suas análises, nela ficam salvas as variáveis que você criar, os dados que você ler ou qualquer outro resultado que você julgue importante de ser salvo.
- **Visualização:** Nessa parte você pode visualizar os gráficos que criar por meio dos comandos. Além disso, aqui você também pode consultar o *Help*, que serve como guia de instruções do R, ou instalar pacotes que serão utilizados em suas análises.

1.3 R Commander

O **R Commander** nada mais é do que um interface gráfica para o usuário que, como o nome sugere, é baseada em comandos. Ou seja, sua estrutura é em forma de menu, onde o usuário seleciona as análises que deseja fazer apenas clicando nas opções. Para utilizá-lo também é necessário ja possuir o R instalado em seu computador, uma vez que o R Commander é um pacote do R.

1.3.1 Como instalar o R Commander

O R Commander é um pacote do R. Um pacote é um conjunto de funções, bancos de dados e códigos que servem como ferramentas adicionais. O propósito do R Commander é simplificar a maneira de se utilizar o R. Para instalá-lo existem duas formas que serão apresentadas abaixo. Para utilizá-lo é recomendado não utilizar o RStudio, apenas o R, pois utilizar o primeiro pode acarretar problemas na instalação e utilização do R Commander.

- **1º método de instalação**

No seu editor de comandos ou no console do R (não utilizar o RStudio), execute o código abaixo:

```
install.packages('Rcmdr')
```

O código acima é um exemplo de execução da função *install.packages*, que serve para instalar novos pacotes no R, entre parenteses temos o argumento da função que remete ao nome do pacote que desejamos instalar.

- **2º método de instalação**

Na aba superior do R, clique em "Pacotes -> Instalar Pacote(s)..." irá abrir uma janela na qual você deve selecionar uma das opções de "Brazil", clique em "OK", Encontre o nome do

pacote desejado ("Rcmdr") e clique em "OK" novamente.

- **Execução**

Quando desejamos utilizar um pacote que foi instalado precisamos carregá-lo no R, para isso utilizamos o comando *library*, veja abaixo como carregamos o pacote Rcmdr, basta executar a função no console ou no editor de códigos do R.

```
library(Rcmdr)
```

Carregar um pacote faz com que várias funcionalidades estejam disponíveis para o usuário, em geral as utilizamos por meio de funções. A função do Rcomander que torna possível utilizá-lo é a mostrada abaixo, basta executá-la em seu editor de código e pronto, o Rcomander está disponível para uso.

```
Commander()
```

É importante ressaltar que, uma vez que um pacote já foi instalado não há necessidade de o instalar novamente sempre que quiser utilizá-lo. A única coisa que você precisa fazer é carregá-lo, ou seja, basta usar a função *library*.

1.3.2 A interface do R Commander

Abaixo vemos a interface do R Commander, você pode notar que ela é bem intuitiva. Na parte superior temos o menu principal, onde você pode selecionar os comandos que deseja executar clicando nas opções. Abaixo temos o console, cuja função é a mesma do R, apresentar os comandos já executados ao clicar nas opções e executar funções temporárias. Os resultados aparecem abaixo, bem como possíveis mensagens de alerta que aparecem na extremidade inferior.

1.4 BancosBio

Essa apostila vem acompanhada de um conjunto de banco de dados que foram reunidos com o objetivo de exemplificar os tópicos que abordaremos mais adiante, e também com o objetivo de se colocar em prática o que foi aprendido. Nós recomendamos fortemente que, ao aprender um novo tópico nos capítulos que seguem, você tente refazer o que foi feito utilizando as mesmas funções que serão ensinadas porém em outro conjunto de dados, pois é na prática que realmente se aprende.

O conjunto de dados BancosBio foi contruído em formato de um pacote, portanto é necessário instalá-lo para começar a utilizar. Já vimos com instalar um pacote no R quando falamos a respeito do R Commander. Para instalar de arquivos locais no R, o usuário deve ir em "Pacotes > Install package(s) from local files...". Agora introduziremos um novo método de instalação no RStudio, que se refere à como instalar pacotes em formatos de arquivos zipados. Antes de tudo, acesse o link: http://www.est.ufmg.br/~monitoria/material.html, baixe o arquivo "*BancosBio_0.1.0.tar*" e salve na pasta que você preferir.

Na parte de visualização do Rstudio clique em "Packages -> Install", na aba "Install from" selecione "Package Archive File", irá abrir uma janela (Caso não abra sozinho clique em "Browse..."), procure pelo pacote na pasta que você o salvou, clique em "Open-> Install".

Uma vez que o pacote já está instalado, para começar a utilizar basta carregá-lo. Fazemos isso da mesma forma que mostramos anteriormente, basta executar o comando abaixo.

```
library(BancosBio)
```

O pacote BancosBio é composto pelos bancos de dados: aids, bacteria, cancer, carangueijos, diabetes, estudante, hello, hormonios, hospital, musculo, nephro, tennis, e valid. Além desses, também são disponibilizados dois bancos adicionais, seeds (formato .txt) e heart (formato .xlsx) que também estão disponíveis no link acima. A razão desses não serem inclusos no pacote é que assim você pode aprender como trabalhar com dados de fontes externas. Cada um desses bancos de dados representa um estudo real nas áreas da saúde ou das ciências biológicas.

Você pode saber mais a respeito de cada um desse bancos de dados consultando o *Help* do R. Para fazer isso você pode explorar a aba "Help" na parte de visualização do RStudio ou simplesmente executar o comando *?* acompanhado do nome da função, comando ou banco de dados para o qual você deseja mais informações. Por exemplo, caso estajamos interessados em saber um pouco mais a respeito do banco de dados "bacteria", simplesmente excutamos o seguinte comando:

```
?bacteria
```

Note que na aba "Help" da parte de visualização, aparece uma breve descrição a respeito do banco de dados e de seu formato. Como exercício, execute *?median* em seu console ou editor de códigos do R e descubra qual o objetivo desse comando.

Capítulo 2

Como ler um banco de dados

Existem diversas maneiras de se ler bancos de dados no R, podendo ser esses bancos de dados de diversos formatos. Os formatos mais comuns são ".xls", ".xlsx", ".txt", ".csv", entre outros. Nessa seção mostraremos como ler alguns formatos de bancos de dados no R/Rstudio e no R Commander, no primeiro caso usando funções e no segundo por meio dos comandos.

Antes disso, precisamos introduzir a noção de diretório, que nada mais é do que a pasta na qual você está trabalhando, onde seus dados estão salvos e etc. Em geral, o R entende que seu diretório é a pasta na qual você salvou seu script. Para descobrir o seu diretório use o comando:

```
getwd()
```

Ele retornará qual sua pasta de trabalho. Isso significa que todos os arquivos que estiverem salvos nessa pasta serão facilmente encontrados pelo R. Para alterar seu diretório, caso seja necessário, você pode usar o comando:

```
setwd("C:/Desktop/Analise/Banco de dados")
```

Entre parênteses, temos por exemplo, um diretório na pasta "Banco de dados" que está salvo em "Analises" que por sua vez está na área de trabalho do computador (Desktop), nesse exemplo podemos carregar as bases de dados salvas na pasta por meio dos métodos que serão apresentados a seguir.

Também existe um método alternativo de se alterar o diretório no RStudio, aperte "Ctrl+Shift+H" e escolha seu novo diretório na janela que surgirá na sua tela.

2.1 Utilizando o R

Como dissemos anteriormente, existem diversas maneiras de se ler um banco de dados no R, podendo esse banco de dados estar em vários formatos. A seguir mostraremos como ler dados que estão salvos em formato de planilha no Excel (xlsx), formato de texto (txt) e formato csv.

2.1.1 Arquivos Excel

Para ler dados de uma planilha excel é necessário previamente instalar um pacote específico, uma vez que essa funcionalidade não está inicialmente disponível no R. Para isso recomendamos instalar o pacote *readxl*. Para isto, use o comando abaixo ou algum dos outros métodos de instalação citados anteriormente. Lembre-se que sempre que você quiser utilizar um pacote que já foi instalado é necessário apenas carregá-lo.

```
install.packages("readxl")
library(readxl)
```

No pacote *readxl* existe uma função chamada *read_excel()* que, como o nome sugere, tem o objetivo de ler arquivos em excel que estão salvos no seu diretório, consulte o *Help* para mais detalhes. Para trabalhar com um banco de dados no R não basta simplesmente excutar um comando, é preciso também armazenar os dados que você deseja analisar na memória do R. Para isso precisamos criar uma variável, que é simplesmente um espaço na memória no qual guardamos valores, bancos de dados ou qualquer análise que julgarmos importante. O comando abaixo lê o banco de dados *heart* (que deve estar salvo no diretório de trabalho) e o armazena na variável *dados*. Para atribuir algum objeto à uma variável usamos o sinal de igualdade.

```
dados = read_excel("heart.xlsx")
```

Como pode ser visto acima, a função *read_excel* é bem simples de ser usada, basta colocar o nome do arquivo com a extensão (.xlsx) entre aspas e armazenar na variável que deseja. Note que, caso você esteja usando o RStudio ao invés do R, irá aparecer na parte de histórico a variável que você criou, clicando nela você pode visualizar os dados foram lidos. Outra alternativa é usar o comando abaixo.

```
View(dados)
```

2.1.2 Arquivos ".csv" ou ".txt"

Para ler arquivos em formato ".csv" ou ".txt" o procedimento é mais simples, uma vez que não é necessário instalar nenhum outro pacote adicional. Usamos a função *read.table()*, veja abaixo:

```
dados2 = read.table("seeds.txt", header = TRUE)
```

Os argumentos passados para a função *read.table* são o nome do arquivo com a extensão e o argumento *header = TRUE*, que indica que seus dados têm cabeçalho, isto é, as colunas possuem um nome específico. Também pode ser necessário indicar mais argumentos na hora de executar uma função, para saber mais sobre isso consulte o *Help*. Para dados em formato ".csv" podemos usar a mesma função, basta ficar atento para extensão do arquivo e o delimitador das colunas.

No R Commander

Ler um conjunto de dados no R Commander é bem simples e intuitivo. Clique em:

Dados > Importar arquivos de dados

selecione o formato do arquivo no qual seus dados estão salvos, irá abrir uma janela onde você deve indicar o nome que gostaria para a variável onde seus dados serão alocados e marcar as opções de acordo com o seu banco de dados. Após isso, clique em "Ok"e irá surgir uma janela na qual você deve localizar a pasta onde seus dados estão salvos, selecione o arquivo com os dados e clique em "Abrir". Pronto, seus dados já estão disponíveis no R Commander para serem analisados. Você pode visualizá-los e/ou editá-los clicando em "Editar conjunto de dados".

Capítulo 3

Análises Descritivas

Antes de qualquer análise estatística é fundamental realizar uma boa análise exploratória dos seus dados. Fazemos isso por meio de análises descritivas, que consistem em criar tabelas de frequências, gráficos e medidas de resumo. Nos tópicos a seguir mostraremos como realizar esse tipo de análise no R/RStudio e no R Commander.

Usaremos o banco de dados *cancer* do pacote BancosBio para ilustrar as aplicações das funções que serão apresentadas nos tópicos seguintes. Esse banco de dados contém informações a respeito de 137 pacientes que foram diagnosticado com câncer de pulmão (para mais detalher consulte o *Help* deste banco de dados).

3.1 Tabelas de Frequência

A frequência absoluta de uma observação é o número de vezes em que tal observação apareceu na amostra. Uma tabela de frequências contém as informações a respeito das frequências dos diferentes níveis de uma variável. Por exemplo, considere a variável "treat" do banco de dados "cancer", que indica se o indivíduo está submetido à um tratamento padrão contra o cancer ou à um tratamento novo. Podemos estar interessados em saber a quantidade de indivíduos submetida a cada tipo de tratamento. Para isso usamos o comando *table*. Veja:

```
table(cancer$treat)
```

```
 1  2
69 68
```

Como podemos ver, 69 pacientes foram submetidos ao tratamento 1 (Padrão) e 68 ao tratamento 2 (Teste). Caso estejamos interessados na frequência relativa temos duas opções, a primeira é simplesmente dividir o resultado pelo total de pacientes e a segunda é usar a função *prop.table*.Perceba que o cifrão usado entre o nome do banco de dados e o nome da variável é responsável por filtrar apenas as observações da variável em questão, ou seja, só foram usados os dados sobre o Tipo de tratamento na execução da função. Veja abaixo o código com as duas opções:

```r
# Opcao 1:
Tab = table(cancer$treat)  # salva a tabela em Tab
total = sum(Tab) #O comando sum soma os valores da tabela

Tab/total  # A frequencia absoluta dividida pelo total é a frequencia relativa
```

```
        1         2
0.5036496 0.4963504
```

```r
# Opcao 2:
prop.table(Tab)
```

```
        1         2
0.5036496 0.4963504
```

Note que as duas opções resultam no mesmo resultado, logo você pode usar a que preferir. Experimente executar o código acima para treinar, note que todo o texto escrito após o # será desconsiderado pelo R. Isso ocorre pois o texto é considerado apenas como um comentário, uma maneira de lembrar o que foi feito no momento em que você criou o script.

Também podemos criar uma tabela de contingência, que registra a frequência de observações de 2 ou mais variáveis categóricas. Fazemos isso novamente utilizando o comando *table*. Suponha que estamos interessados em investigar se existe uma associação entre o tipo de tratamento e o fato de o paciente ter feito uma terapia prévia ou não, observe a tabela abaixo:

```r
table(cancer$treat, cancer$prior)
```

```
    0 10
  1 48 21
  2 49 19
```

Na linha 1 temos os indivíduos que foram submetidos ao tratamento 1, são 69 indivíduos, dos quais 48 não realizaram tratamento prévio (categoria 0) e 21 realizaram (categoria 1). Da mesma forma interpretamos a linha 2, dos 68 indivíduos submetidos ao tratamento 2, 49 não realizaram tratamento prévio e 19 realizaram. Também podemos analisar a tabela com as frequências relativas, veja abaixo:

```r
tab=table(cancer$treat, cancer$prior)
prop.table(tab, margin = 1)
```

```
           0         10
  1 0.6956522 0.3043478
  2 0.7205882 0.2794118
```

O argumento *margin = 1* indica que a proporção deve somar 100% em cada linha, ou seja,

dado que o indivíduo está realizando determidado tratamento qual a chance dele ter realizado um tratamento anterior ou não. Se você indicar *margin = 2* o total de 100% será obtido em cada coluna e caso esse argumento não seja informado a proporção será distribuída em toda tabela.

No R Commander

Primeiro você deve assegurar que as variáveis são do tipo fator, que é como o R interpreta as variáveis categóricas. Para isso vá em

Dados -> Modificação de variáveis no conjunto de dados -> Converter variável numérica para fator

Irá surgir uma janela onde você deve selecionar a variável que deseja converter e pode optar por definir o nome das categorias ou usar números, caso a variável desejada não apareça na lista significa que o R já reconheceu que essa variável é categórica. Após informar ao R que a variável é categórica vá em

Estatísticas -> Resumos-> Distribuições de frequência

ou

Estatísticas -> Tabelas de Contingência -> Tabela de dupla entrada.

Note que a aba "Estatísticas"permite configurar como você deseja calcular as frequências relativas.

3.1.1 Tabelas de Frequência para Variáveis Contínuas

Em algumas situações, como quando estiver trabalhando com variáveis númericas, será necessário contar o número de observações que estão dentro de um intervalo. Para isso, teremos que categorizar a variável em questão e depois colocar os valores em uma tabela de frequência. Para categorizar uma variável contínua no R, você pode usar o comando *cut* e os argumentos desta função são as observações, um argumento chamado *breaks*, em que se armazenam os limites dos intervalos e o argumento *labels*, que são os nomes que os elementos receberão. Confira:

```
categorizada<-cut(1:30, breaks = c(0,10,20,30),
                  labels = c("1-10","11-20","21-30"))
categorizada
```

```
 [1] 1-10  1-10  1-10  1-10  1-10  1-10  1-10  1-10  1-10  1-10  11-20 11-20
[13] 11-20 11-20 11-20 11-20 11-20 11-20 11-20 11-20 21-30 21-30 21-30 21-30
[25] 21-30 21-30 21-30 21-30 21-30 21-30
```

```
Levels: 1-10 11-20 21-30
```

Agora, aplicando a tabela, temos:

```
table(categorizada)
```

```
categorizada
 1-10 11-20 21-30
   10    10    10
```

No R Commander

Para categorizar uma variável numérica no R Commander é necessário ir em
Dados > Modificação de variáveis no conjunto de dados... > agrupar em classes uma variável numérica (para criar fator)...
Na janela que abrirá é possível escolher a variável que deseja categorizar, o nome da nova variável, o número de classes e como elas serão separadas e posteriormente o nome de cada classe. Após a categorização, para tabelar as frequências basta seguir o passo a passo anterior com a nova variável criada.

3.2 Medidas de Resumo

As medidas de resumo são algumas informações que nos ajudam a resumir importantes características da amostra em um único valor. No decorrer da seção será mostrado como calcular tais medidas no R e suas respectivas interpretações.

3.2.1 Média

A média é uma das mais conhecidas medidas de resumo. Ela é uma medida que revela a tendência central dos dados observados. Para calculá-la, utiliza-se a seguinte fórmula:

$$\overline{x} = \frac{\sum_{i=1}^{n} x_i}{n} = \frac{x_1 + ... + x_n}{n}.$$

No R, o cálculo pode ser simplificado com a função *mean*, que recebe como argumento o banco de dados a ser utilizado. Para ilustrar duas formas de calcular a média, considere que um fazendeiro deseja estimar a produção de leite por vaca em sua fazenda em um determinado dia. Para isso, ele mediu a produção de 9 vacas obtendo os valores 23, 24, 45, 53, 62, 74, 80, 82 ,89. No R podemos calcular a média fazendo:

```
media = (23+45+67+24+89+74+82+80+53+62)/10
media
```

```
[1] 59.9
```

```
amostra = c(23, 45, 67, 24, 89, 74, 82 ,80 ,53 ,62)
mean(amostra)
```

```
[1] 59.9
```

Como conclusão temos que a quantidade média de leite produzido por vaca nessa dia é de 59.9 litros.

Algumas vezes necessitamso de calcular a média de uma variável para diferentes grupos e para isso é possível utilizar a função *by*, que possui como argumentos a varíavel de interesse, seguida da variável categórica que determina os grupos e a função que desejamos utilizar para calcular a quantidade de interesse. No exemplo mostrado abaixo é possível compreender melhor. Nesse caso, queremos comparar a média de idade dos indivíduos do status 0 e 1 (O que significa???). Calculamos a média de idade dos dois grupos com o seguinte comando

```
by(cancer$age, cancer$status, mean)
```

```
cancer$status: 0
[1] 55.44444
------------------------------------------------------------
cancer$status: 1
[1] 58.50781
```

Logo, a média de idade daqueles de status 0 é 55.44444, enquanto que daqueles de status 1 é 58.50781. A função *by* também pode ser aplicada em conjunto com as outras medidas de resumo que serão vistas a seguir.

3.2.2 Mediana

A mediana é uma medida de tendência central cujo objetivo é mostrar identificar a posição central do banco de dados, isto é, ela é definida como o valor que é maior ou igual a pelo menos metade dos valores observados na amostra e, ao mesmo tempo, é menor ou igual a pelo menos metade dessa mesma amostra. A mediana também é conhecida como segundo quartil ou percentil 50. No R é possível utilizar as funções *median* e *quantile*. A função *median* recebe como argumento o conjunto de dados e a função *quantile* recebe o vetor com os valores da amostra e o argumento *probs* que determina a proporção considerada no percentil desejado, no caso deve igualado a 0.5. Considere o exemplo a seguir:

- Uma médica anotou o número de dias que cinco de seus pacientes levaram para se recuperarem de uma infermidade. Os dados obtidos foram 2, 8, 26, 9, 8 e 14 dias. Ela poderá calcular a mediana dessa amostra com os seguintes comandos:

```
amostra = c(2,8,26,9,8, 14)
median(amostra)
```

```
[1] 8.5
```

```
quantile(amostra, probs = 0.5)
```

```
50%
8.5
```

Neste exemplo, vemos que o número mediano dos dias que os pacientes levaram para se curar é igual a 8,5, ou seja, pelo menos 50% das observações são menores que 8,5 e pelo menos 50% das observação são maiores que 8,5 dias. Note que a mediana é igual a média dos dois valores centrais da amostra ordenada, procedimento utilizado quando o tamanho da amostra é par.

3.2.3 Quartis

Assim como foi feito com a mediana, para calcularmos os demais quartis de uma amostra, o primeiro e o terceiro, podemos utilizar novamente o comando *quantile*, mas dessa vez atribuindo ao argumento *probs* o valor 0.25 para o primeiro quartil e 0.75 para o terceiro. Se um valor para este argumento não for definido, a função retornará uma tabela com o menor valor, o primeiro, segundo e terceiro quartil e o maior valor da amostra. Para entender melhor, considere o exemplo a seguir:

- Uma nutricionista coletou o número de frutas que cinco dos seus pacientes comeu durante uma semana. Ela utiliza os seguintes comandos para encontrar os valores do primeiro e do terceiro quartis:

```
amostra = c(2,8,26,9,14)
quantile(amostra, probs = 0.25)
```

```
25%
  8
```

```
quantile(amostra, probs=0.75)
```

```
75%
 14
```

Ou seja o primeiro quartil da amostra é 8, isto é, podemos dizer que pelo menos um quarto da amostra observada come até 8 frutas por semana. Já o terceiro quartil é igual a 14, ou seja, pelo menos três quartos da amostra consomem no máximo 14 frutas por semana.

Alternativamente, podemos utilizar:

```
amostra = c(2,8,26,9,14)
quantile(amostra)
```

```
  0%  25%  50%  75% 100%
   2    8    9   14   26
```

Note que o primeiro e o último valores exibidos são o mínimo e máximo da amostra.

3.2.4 Função Summary

Você deve ter reparado que no R Commander, com apenas um comando é possível obter uma tabela com algumas das principais medidas de resumo. Sem utilizarmos o R commander isto também possível utilizando a função *summary*. Considere o banco dados *cancer* disponível no pacote BancosBio. Podemos obter os principais resumos sobre a idade dos pacientes com o seguinte comando:

```
summary(cancer$age)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  34.00   51.00   62.00   58.31   66.00   81.00
```

Avaliando os resultados percebe-se que a idade média dos pacientes com câncer é igual a 58.31 anos e que a mediana é igual a 62 anos. Ao avaliar o primeiro e terceiro quartis, que foram iguais a respectivamente 51 anos e 66 anos, em conjunto dos valores mínimos e máximos da amostra, é possível concluir que pelo menos 25% das observações da amostra são menores que 51, bem como pelo menos 25% da amostra está no intervalo entre 66 e 81 anos.

3.2.5 Variância e Desvio Padrão

Nessa seção discutiremos como o Desvio Padrão e a Variância podem ser calculados no R. O Desvio Padrão é a distância média entre os valores observados na amostra e a média da mesma, como pode-se ver na fórmula abaixo:

$$s = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n-1}} = \sqrt{\frac{(x_1 - \overline{x})^2 + ... + (x_n - \overline{x})^2}{n-1}}$$

Já a Variância, é o quadrado do desvio padrão, isto é, $Var(X) = s^2$. Ambas medidas são usadas para avaliar o quanto os valores de uma amostra variam, no caso, quanto maior a variância ou o desvio, maior será a variabilidade. Lembre-se que ambas medidas só podem assumir valores positivos. Para calculá-las no R, é possível usar as funções *sd* e *var*, respectivamente para calcular desvio padrão e variância, ou, a partir de uma função é possível encontrar ambas medidas, dado as relações mostradas anteriormente. Veja:

```
sd(cancer$stime)
```

```
[1] 157.8167
```

O desvio padrão do tempo de acompanhamento dos pacientes com câncer é 157.8167 dias.

```
var(cancer$stime)
```

```
[1] 24906.12
```

Enquanto a variância é igual a 24906.12 dias.

```
(sd(cancer$stime))^2
```

```
[1] 24906.12
```

Repare que esse valor é exatamente igual ao anterior.

3.2.6 Escore Padronizado

O escore padronizado pode ser utilizada para verificar se uma observação do banco de dados é um valor atípico. Com ele também podemos comparar dois valores de diferentes amostras, que têm médias e desvios padrões diferentes, avaliando quantos desvios padrões de distância tal observação se encontra da média de sua amostra. O escore padronizado é calculado segundo a fórmula a seguir:

$$z_i = \frac{x_i - \overline{x}}{s}$$

Para entendermos como obter os escores padronizados de uma amostra, considere o seguinte exemplo: Em uma colheita, cinco macieiras foram escolhidas para contar o número de maçãs que cada uma gerou, obtendo uma amostra de 44, 5, 7, 3 e 42. Para avaliar melhor a distância da maior observação das demais, calculou-se seu escore padronizado com o seguinte código:

```
amostra = c(44,5,7,3,42)
z = (44-mean(amostra))/sd(amostra)
z
```

```
[1] 1.140206
```

Assim, o valor padronizado de 44 é 1.140206. Isto é, a macieira que gerou 44 maçãs está a 1.140206 desvios da média da amostra. Para calcurmos o escore padronizado para toda a amostra podemos fazer:

```
z = (amostra-mean(amostra))/sd(amostra)
z
```

```
[1]  1.1402056 -0.7281985 -0.6323829 -0.8240141  1.0443900
```

Também podemos utilizar a função *scale*. O argumento pedido pela função é o banco de dados, e, nesse caso, todos os valores da amostra são padronizados. Observe os próximos comandos.

```
scale(amostra)
```

```
           [,1]
[1,]  1.1402056
[2,] -0.7281985
[3,] -0.6323829
[4,] -0.8240141
```

```
[5,]  1.0443900
attr(,"scaled:center")
[1] 20.2
attr(,"scaled:scale")
[1] 20.87343
```

3.2.7 Coeficiente de Variação

Nas seções anteriores vimos como calcular a variância e o desvio padrão, que são utilizados para avaliar a variação entre os valores de uma amostra. Porém quando deseja-se comparar a variação em amostras cujos elementos estão em escalas de grandezas muito diferentes ou com unidades de medidas diferentes, como por exemplo, se você quiser comparar a variação no peso de um grupo formigas e a variação no peso de um grupo de elefantes, será mais adequado utilizar o coeficiente de variação. O coeficiente de variação, nos diz qual a porcentatem do desvio padrão em comparação ao valor da média amostral, e sua fórmula é

$$CV = \frac{s * 100}{\overline{x}}\%$$

Para exemplificar, considerare o exemplo anterior das macieiras. O coeficiente de variação desta amostra é dado por

```
cv = (sd(amostra)*100)/mean(amostra)
cv
```

```
[1] 103.3338
```

Logo, o coeficiente de variação dessa amostra é igual a 103,3338%, ou seja, o desvio padrão representa 103,3338% do valor da média amostral.

Para efeito de comparação, agora vamos calcular o coeficiente de variação da quantidade de bananas geradas nas mudas de bananeiras na mesma fazenda do exemplo anterior. Neste caso temos:

```
bananas=c(20, 23, 32, 38, 37)
cv2 = (sd(bananas)*100)/mean(bananas)
cv2
```

```
[1] 27.18251
```

O coeficiente de variação para a amostra de produção de banana foi igual a 27,18251%, ou seja, existe maior variação na produção de maçãs do que de bananas.

3.2.8 Coeficiente de Correlação de Pearson

O Coeficiente de Correlação de Pearson é utilizado quando queremos mensurar o quanto duas variáveis estão relacionadas, ou seja, se o fato de uma das variáveis aumentar faz com que a outra aumente ou diminua. Ele é representado pela letra r, em que -1<r<1. Quanto maior o módulo de r, maior a correlação, que pode ser negativa ou positiva. A fórmula do Coeficiente de Correlação de Pearson é:

$$r = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \overline{y})^2}}$$

Que pode ser também calculado com o comando *cor*, cujos argumentos serão as variáveis que deseja estudar. Veja o exemplo a seguir, em que estamos avaliando a idade e o Escore de Karnofsky do desempenho do paciente:

```
cor(cancer$age,cancer$Karn)
```

```
[1] -0.09498481
```

Observe que o Coeficiente de Correlação está muito próxima de zero, logo, não há evidência de que as variáveis estudadas sejam correlacionadas.

3.2.9 Medidas de resumo no R Commander

No R Commander há duas principais formas de obter medidas resumo de interesse e nas duas formas é possível obter os resumos para diferentes grupos formados de acordo com uma variável categorica. Para habilitar as janelas de opções, inicialmente é necessário carregar algum banco de dados. Na primeira alternativa, em que podemos obter média, desvio padrão, percentis (quartis), coeficiente de variação, entre outras coisas. Siga os passos a seguir:

Vá em

Estatísticas > Resumos > Resumos numéricos...,

escolher a variável em questão e as medidas de interesse. O software retornará uma tabela com os resultados.

Na segunda alternativa é obrigatória a escolha de uma variável categorica para a separação de grupos e podemos obter média, mediana e desvio padrão. Procedemos de acordo com os comandos a seguir:

Clique em

Estatísticas > Resumos > Tabelas de Estatísticas...

e selecionar a medida desejada, a variável que será utilizada como fator e a variável para qual tem o interesse de calcular o resumo.

Se temos o interesse em calcular escores padronizados podemos seguir os seguintes passos:

Vá em

Dados > Modificação de variáveis no conjunto de dados... > Padronizar variáveis.

Note que novas colunas serão incluídas no banco de dados contendo os escores padronizados para as variáveis escolhidas.

Para calcular o Coeficiente de Correlação de Pearson no R Commander, você deve seguir as seguintes especificações:

Clique em

Estatísticas > Resumos > Matriz de Correlação

quando a janela de escolha abrir, se atente em selecionar a segunda variável com a tecla Ctrl pressionada e de escolher o tipo de correlação "Produto-momento de Pearson". O resultado retornado é uma matriz cujos os valores contidos na 2° e 3° células, que devem ser o mesmo, representam o valor do coeficiente de correlação.

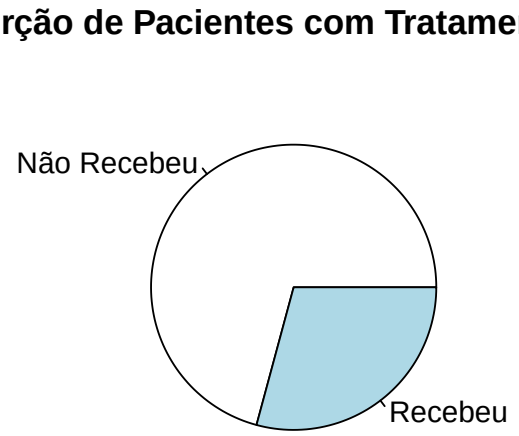
3.3 Análise Gráfica

Análise gráfica remete à parte visual da análise descritiva. Por meio de gráficos podemos identificar padrões ou entender melhor o comportamento de uma variável no banco de dados, ou até mesmo, o comportamento conjunto de duas ou mais variáveis. A seguir mostraremos como fazer no R os principais tipos de gráficos e explicaremos um pouco sobre a finalidade de cada um deles. O banco de dados *"cancer"* será utilizado na construção da maioria dos gráficos vistos a seguir.

3.3.1 Gráfico de Setores

O gráfico de setores é utilizado para analisarmos a distribuição de frequências de variáveis qualitativas. Ele é recomendado para variáveis categóricas com poucas categorias. No software R, o gráfico de setores é construido com o uso da função *pie* em conjunto com as funções *table* e *prop.table*. Como pode ser observado no exemplo, em que estuda-se a proporção de pacientes que receberam tratamento prévio, o argumento do comando deve ser a tabela de frequências absolutas ou relativas da variável a ser estudada. Pode-se notar também que foram utilizados outros argumentos, que podem ser usados na grande maioria das contruções gráficas, sendo eles o argumento *labels* para definir os nomes das categórias na legenda e o *main* para definir o título do gráfico. Veja a seguir:

```
pie(prop.table(table(cancer$prior)), labels = c("Não Recebeu","Recebeu"),
    main = "Proporção de Pacientes com Tratamento Prévio")
```

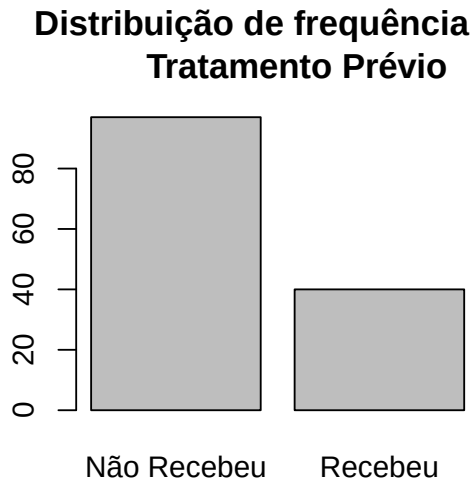


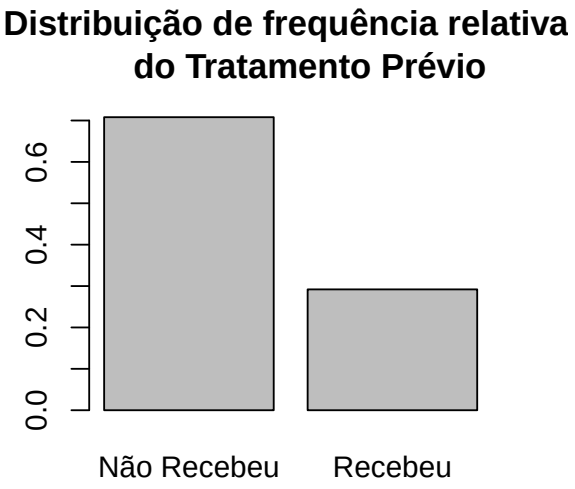
3.3.2 Gráfico de Barras

O Gráfico de barras pode ser utilizado para variáveis qualitativas e quantitativas discretas, e deve escolhido em detrimento do gráficos de setores quando a variável apresenta uma ordenação nas suas respostas, como ocorre com variáveis qualitativas ordinais e quantitivas discretas. Para criá-lo, deve-se utilizar o comando *barplot* também em conjunto com as funções *table* e *prop.table*. No exemplo a seguir, ilustramos o uso dos gráficos com as frequências absoluta e relativa.

```
par(mfrow=c(1,2)) ## Utilizado para colocar dois gráficos lado a lado
barplot(table(cancer$prior), names.arg = c("Não Recebeu","Recebeu"),
    main= "Distribuição de frequência do
    Tratamento Prévio")

barplot(prop.table(table(cancer$prior)), names.arg = c("Não Recebeu","Recebeu"),
    main= "Distribuição de frequência relativa
    do Tratamento Prévio")
```



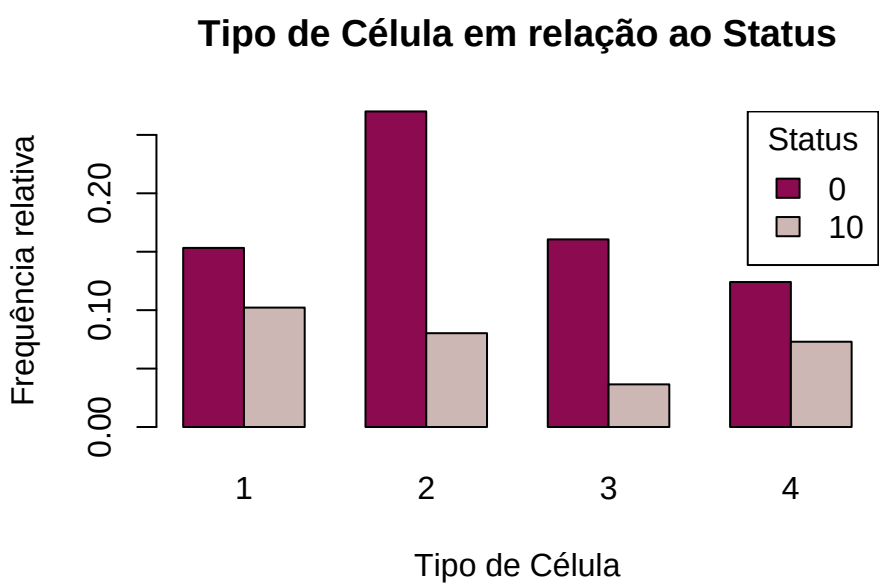


A partir da análise destes gráficos, vemos que a maior parte dos pacientes que não receberem tratamento prévio, totalizando 70% deles.

Também podemos estar interessados em construir um gráfico de barras para visualizar a frequência de duas variáveis conjuntamente. Considere que estamos interessados em observar o total de pacientes que possuem cada tipo de célula dentro do grupo dos pacientes que pertecem ao Status 0 ou 10. Veja abaixo como podemos realizar essa análise:

```
barplot(prop.table(table(cancer$prior,cancer$cell)),
main = "Tipo de Célula em relação ao Status",
beside = T, xlab = "Tipo de Célula", col = c("deeppink4","mistyrose3"),
ylab = "Frequência relativa")

legend("topright", legend = levels(cancer$prior),
       fill = c("deeppink4", "mistyrose3"), title ="Status")
```



Avaliando o gráfico, temos que a célula do tipo 2 é a que concentra a maior frequência os indivíduos de status 0, enquanto que a célula do tipo 1, possui o maior número de indivíduos de status 10.

Note que na construção do gráfico acima utilizamos, na função *barplot*, os seguintes argumentos:

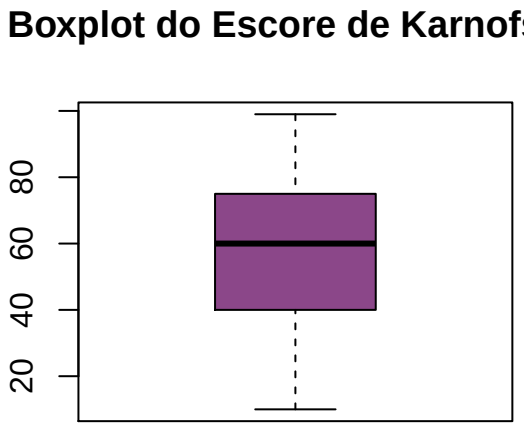
- *beside = TRUE* - para as barras ficarem uma do lado da outra;
- *xlab* - para nomear o eixo x;
- *col* - para determinar as cores do gráfico.

O código acima possui a função *legend* que responsável pela criação da legenda. Essa função pode ser usada em outros gráficos do R. Em seus argumentos foi especificado a posição da legenda (*"topright"*), os nomes das categorias (*legend*), as cores na legenda (*fill*) e o título da legenda (*title*). Observe que os nomes das cores levam no R aparecem em inglês. Para conferir todas as cores disponíveis basta usar as funções *colors()* ou *colours()*.

3.3.3 Boxplots

O boxplot é um gráfico para variáveis quantitativas que pode ser utilizado para comparação de tendência central e variabiladade entre grupos, avaliação de assimetria na distribuição dos dados e presença de valores atípicos na amostra. Para criar um boxplot utilizamos a função *boxplot*, na qual o primeiro argumento será o vetor númerico com a amostra da variável que você deseja visualizar. Considere o exemplo a seguir, em que se deseja avaliar a simetria do Escore de desempenho de Karnofsky dos pacientes.

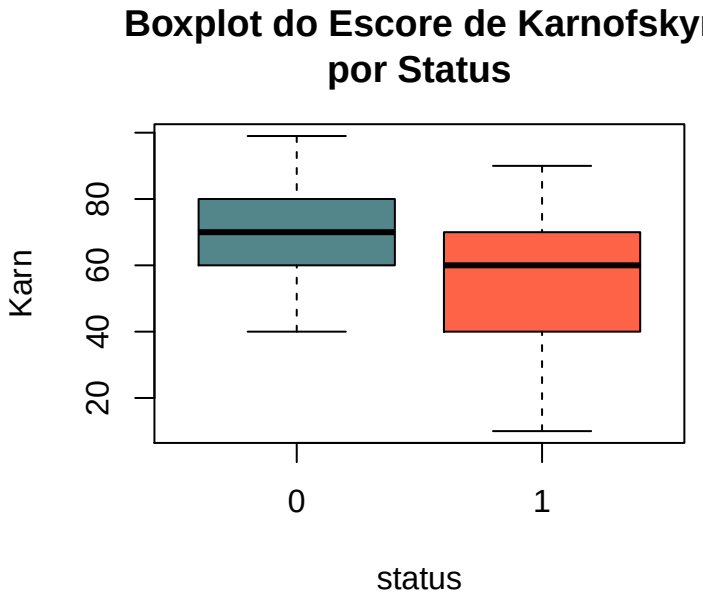
```
boxplot(cancer$Karn, main = "Boxplot do Escore de Karnofsky", col = "orchid4")
```



Neste gráfico fica perceptível que os valores do Escore de Karnofsky dos pacientes são consideravelmente simétricos, já que uma quantidade próxima de pacientes se encotra a cima e abaixo da mediana. Também pode-se concluir que os 1°, 2° e 3° são aproximadamente 40,60 e 75, respectivamente. Nenhum valor atípico é observado no conjunto de dados.

No exemplo abaixo, comparamos o Escore de Karnofsky segundo os dois níveis da variável Status.

```
boxplot(cancer$Karn ~ cancer$status, main ="Boxplot do Escore de Karnofskyr
por Status", ylab = "Karn", xlab = "status", col = c("cadetblue4","tomato1" ))
```



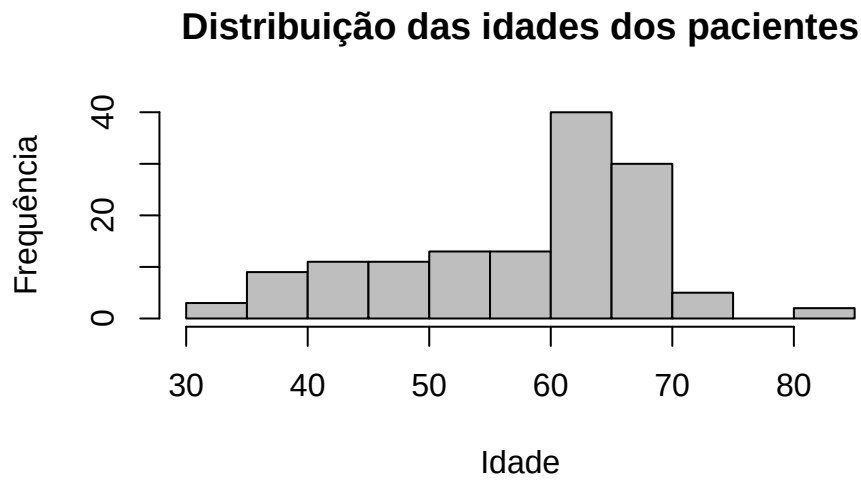
Ao analisar esse gráfico percebe-se que a todos os quartis e os valores de mínimo e máximo do Escore de Karnofsky do desempenho dos pacientes do status 0 são maiores que as mesmas medidas dos indivíduos do status 1. Observa-se também que há menor variabilidade no Escore do

conjunto dos indivíduos de status 0, em comparação aos de status 1, bem como maior simetria.

3.3.4 Histograma

O histograma é gráfico gráfico adequado para estudarmos a distribuição de frequências das respostas de uma variável contínua. Neste gráfico, a altura de barras podem representar a frequência absoluta, relativa ou densidade dos intervalos construídos para a variável. Para criar um histograma no R usamos o comando *hist*. Abaixo ilustramos o uso desta função com a variável "age" do banco de dados "cancer". No primeiro exemplo, passamos como argumento da função apenas os valores da variável de interesse, para isso usamos a expressão *cancer$age* que representa os valores localizados na coluna "age" do banco de dados "cancer". Veja abaixo:

```
hist(cancer$age, main =  "Distribuição das idades dos pacientes", xlab = "Idade",
     ylab = "Frequência", col = "grey")
```



De acordo com o gráfico gerado, é possivel observar que a maior parte dos pacientes têm idade entre 60 e 70 anos e pouquíssimos têm menos que 30 anos ou mais qu 90 anos.

A função *hist* também possui outros argumentos importantes como, por exemplo, o argumento *freq* que se receber *FALSE* fará com que o histograma seja construído com a densidade de cada intervalo. Similarmente, podemos utilizar *prob=TRUE*. Neste caso, a área total de todas as barras do histograma será igual à 1.

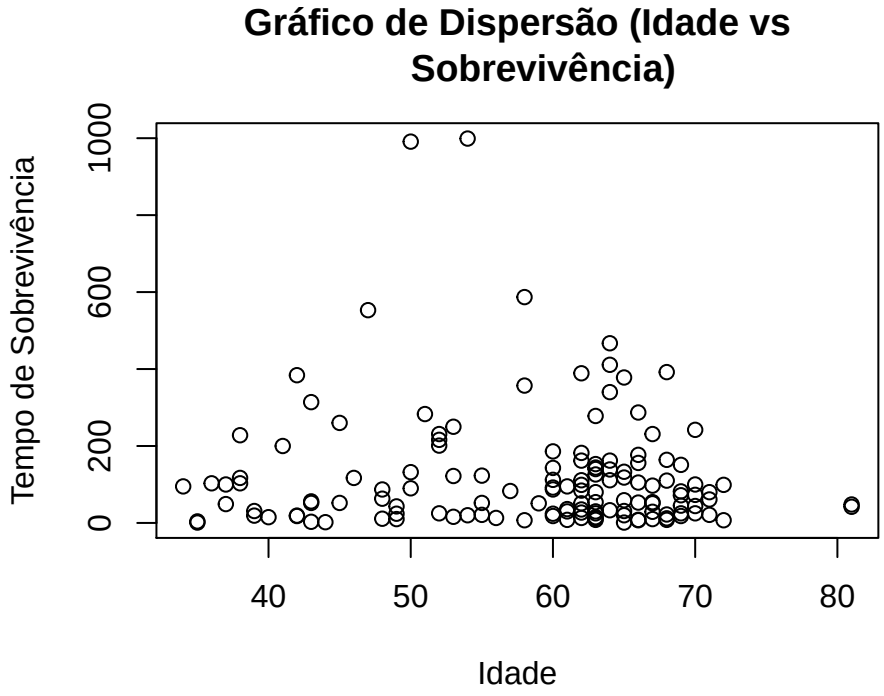
3.3.5 Diagrama de dispersão

Em geral, o diagrama de dispersão é utilizado para avaliarmos se duas variáveis numéricas estão relacionadas. Neste tipo de representação gráfica, utiliza-se um plano cartesiano em que

cada eixo representa uma das variáveis envolvidas e cada indivíduo da amostra é representado por um ponto, cujas as coordenadas equivalem ao valor que as variáveis assumem para este indivíduo.

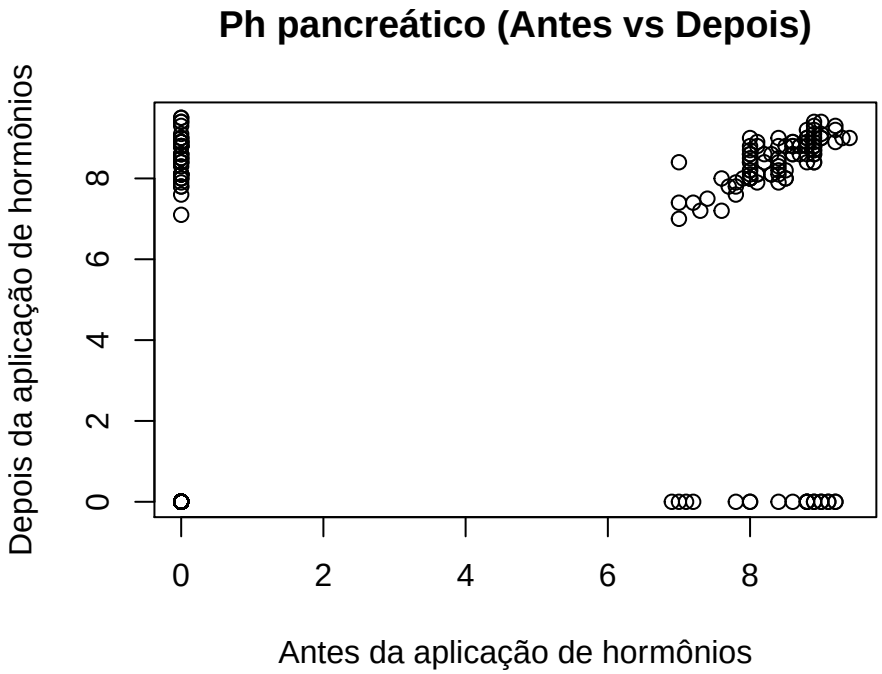
É possível gerá-lo a partir da função *plot*, cujos argumentos são as variáveis representadas nos eixos x e y, respectivamente. No seguinte exemplo, estuda-se uma possível correlação entre a Idade e o Tempo de Sobrevivência em indivíduos com câncer. Os pontos, em geral, se concentram na base do gráfico e não é possível dizer que as variáveis Idade e Tempo de Sobrevivência possuem alguma correlação.

```
plot(cancer$age, cancer$stime, main = "Gráfico de Dispersão (Idade vs
     Sobrevivência)", ylab ="Tempo de Sobrevivência" , xlab = "Idade")
```



Agora vamos observar a relação entre o Ph da secreção pancreática antes e depois da aplicação de hormônios, variáveis disponíveis no banco de dados *hormonios* do pacote BancosBio.

```
plot(hormonios$Panphpr, hormonios$Panphpt, main = "Ph pancreático (Antes vs Depois)",
 ylab = "Depois da aplicação de hormônios", xlab = "Antes da aplicação de hormônios")
```



Nesse caso, podemos ver que alguns indivíduos não tiveram secreção pancreática antes ou depois, e para os outros indivíduos existe uma relação linear positva entre as duas varíaveis, isto é, indivíduos com maior Ph antes da aplicação dos hormônios tendem a ter maior Ph permanecerá após a aplicação.

3.3.6 Análise gráfica no R Commander

No R commander os gráficos de interesse podem ser feito simplesmente acessando o menu "Gráficos" e escolhendo o gráfico desejado. Note que, dependendo do gráfico escolhido, é possível fazer gráficos separando por grupos e configurar inumeras opções, como nomes dos eixos, título, escolha da escala (frequência absoluta, relativa, densidade),etc. Abaixo exemplificamos como fazer um gráfico de barras.

Depois de carregar o banco de dados no R Commander, basta seguir em
Gráficos > Gráficos de Barras.
Escolha a variável desejada e, caso seja de interesse, escolha uma variável para definir grupos.
No caso de um gráfico com duas variáveis, podemos melhorar a visualização em
Opções > Estilo de barras agrupadas > Lado a lado (paralelo).

3.4 Exercícios

Os exercícios a seguir devem ser feitos exclusivamente no software R ou no RStudio. Os bancos de dados utilizados estão no pacote BancosBio (ver seção 1.4), consulte o hep para detalhes de cada um dos bancos de dados.

1. Considerando o banco de dados *bacteria*, obtenha tabelas de contigência com frequências relativa e absoluta para as variáveis que indicam a utilização de princípio ativo e nível das complicações. Mostre como calcular a frequência relativa pelo total geral, por linhas e por colunas.

Utilizando o banco de dados *diabetes* faça as questões 2 e 3.

2. Para a variável idade, faça uma tabela de frequência com 6 intervalos de idade de amplitude de 10 anos cada. Obtenha também a frequência relativa de cada intervalo.
3. Obtenha a proporção de pessoas que possuem diabetes.

4. Com base nos dados apresentados no banco *tennis*, compare gráficamente os sexos masculino e feminino com respeito o número de contusões no cotovelo. Quais gráficos são mais adequados para esta comparação.

Considere o banco de dados *hormonios* para as questões 5 a 8.

5. Mostre qual foi o hormônio foi mais utilizado.
6. Encontre o 1°, 2° e 3° quartil da quantidade de secreção pancreática antes da aplicação de hormônio.
7. Calcule a dose média para cada um dos diferentes hormônios.
8. Calcule o Coeficiente de Correlação linear entre a secreção biliar e o ph biliar antes e depois da aplicação de hormônios.

9. Calcule a média da variável *Wr.Hand* do banco de dados *estudantes*.

Utilize o banco de dados *caraguejos* para as questões 10 e 11.

10. Compare a largura da carapaça em caranguejos em termos de tendência central.
11. Existe uma relação entre o comprimento e a largura da carapaça do caranguejo?
12. Encontre o Coeficiente de Variação do tamanho do lobo frontal em caranguejos.

13. Calcule a variância e o desvio padrão da idade dos pacientes presentes no banco de dados *cancer*.

Com o banco de dados *hospital*, faça as questões 14 a 16.

14. Calcule o coeficente de variação do tempo duração de estadia no hospital?

15. Pode-se dizer que há uma relação entre a temperatura e a número de glóbulos brancos de um paciente ao ser admitido num hospital?
16. Deseja-se comparar a temperatura representada pela variável *Temp* dos indivíduos que receberam e não receberam antibiótico. Quais gráficos podem ser utilizados nessa situação? Construa o gráfico de sua preferência e interprete-o.

Considere o banco de dados *aids* para fazer as questões 17 e 19.

17. Compare em termos de variabilidade da idade em que os pacientes foram diagnosticados com aids.
18. Construa um gráfico de setores para investigar a variável sexo. Qual gênero foi mais acometido pela doença?
19. Construa um gráfico de barras para comparar as variáveis *sex* e *status*. Você diria que existe alguma associação entre essas duas variáveis?

20. Calcule o escore padronizado das observações da variável *Samp_sz* do banco de dados *nephro*. Existem valores atípicos?

21. Utilizando o banco de dados *valid*, a gordura saturada dos pacientes tem maior variação do que a gordura total?

22. Considere o banco de dados *estudante*. A pesquisadora deseja encontrar o intervalo de idade em que a maioria dos respondentes se encontra. Com quais gráficos é possível encontrar essa informação? E qual é esse intervalo?

23. Utilizando o banco de dados *seeds* (Disponível em formato .txt), crie tabelas de frequência absoluta e relativa para a variável *Variedade* que indica o tipo do grão de trigo. Comente os resultados.

Capítulo 4

Distribuições de Probabilidade

Além das análises utilizando estatísticas descritivas, o R contém pacotes que nos permitem trabalhar com probabilidades. Ao longo das disciplinas de estatística, vamos nos deparar com diversos exercícios de probabilidade, e o R é uma boa ferramenta para se conferir cálculos feitos nestes exercícios. Além do mais, a probabilidade no R amplia nossas possibilidades no âmbito dos estudos em que queremos checar o quão provável é a ocorrência de um evento, mas nem sempre conseguimos encontrar tais probabilidades efetuando cálculos à mão.

Neste tópico, para exemplificar, vamos abordar três distribuições conhecidas, são elas: Poison (Variável discreta), Binomial (Variável discreta) e Normal (Variável contínua). No R encontramos funções que, dados os parâmetros de cada distribuição, calculam probabilidades, quantis e geram valores de cada uma das distribuições de probabilidade.

4.1 Distribuição Binomial

Um dos principais modelos discretos de probabilidade é o modelo Binomial. Quando possuímos uma sequência de n experimentos independentes entre si, com variáveis binárias como resposta (sucesso ou fracasso), dada uma determinada probabilidade constante de sucesso, podemos obter a probabilidade de ocorrência de k sucessos através da distribuição binomial.

Podemos exemplificar o modelo binomial do seguinte modo: Na Pesquisa Nacional de Saúde do Escolar do ano de 2015 feita pelo IBGE, aferiu-se que dentre os alunos brasileiros de 9º ano, cerca de 79% receberam orientação sobre prevenção de gravidez na escola. Imagine uma amostra da população dos alunos de 9º ano. Se escolhermos aleatoriamente 10 pessoas nesta faixa escolar, podemos muito bem encontrar 10 pessoas que receberam orientações sobre a gravidez na escola. Pode ser também que, na nossa amostra, nenhum aluno tanha recebido tais orientações. Mas como mensurar as probabiliidades desses eventos? Eles são prováveis ou não?

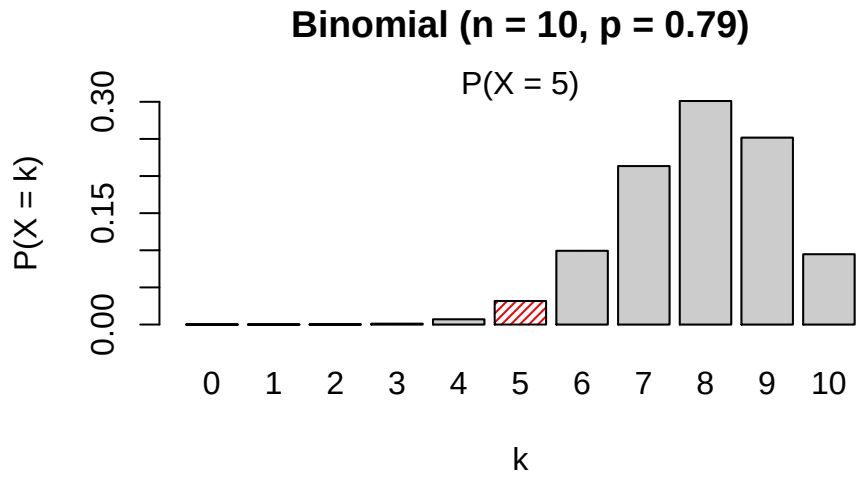
Podemos perfeitamente calcular as probabilidades desejadas usando o modelo binomial: neste caso, assumindo que a exposição referente à prevenção de gravidez seja independente para cada indivíduo da nossa amostra, nosso n seria igual a 10 e p seria $0,79$. Assim, probabilidades

referentes a número X de indivíduos que receberam orientações podem ser calculados com o auxílio da seguinte formula:

$$P(X = k) = \frac{n!}{k!(n-k)!}\, p^k (1-p)^{n-k}.$$

4.1.1 Cálculo de probabilidades

Suponha que na nossa amostra de 10 alunos de 9º ano, $n = 10$, estejamos interessados na probabilidade de encontrarmos 5 estudantes que obtiveram orientações sobre gravidez, $k = 5$, e lembrando que $p = 0,79$. Calculamos a probabilidade $P(X = 5)$ da seguinte maneira:




```
dbinom(x = 5, size = 10, prob = 0.79)
```

```
[1] 0.03166886
```

E para $P(X = 9)$?

```
dbinom(x = 9, size = 10, prob = 0.79)
```

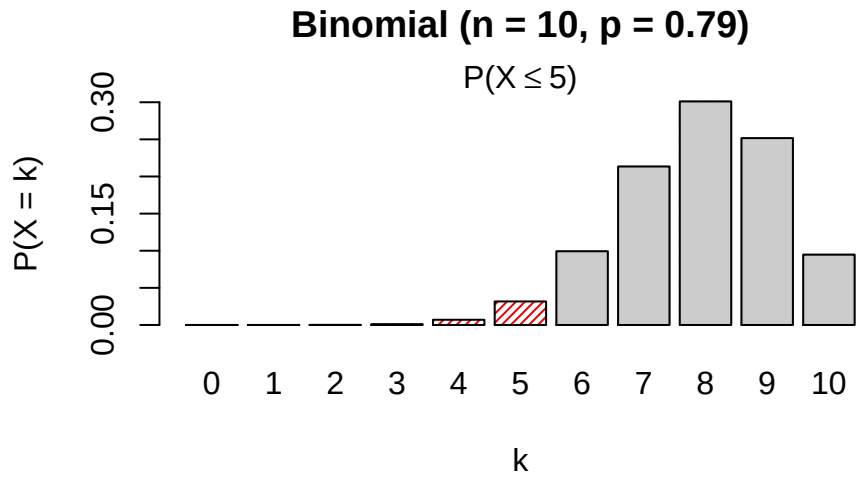
```
[1] 0.2516884
```

Repare que o argumento size em dbinom representa o número de replicações do experimento (resposta de cada aluno da amostra), referente ao parâmetro n da binomial.

4.1.2 Cálculo de probabilidades acumuladas

Agora, suponha o mesmo contexto do exemplo anterior. Porém, desta vez, queremos encontrar a probabilidade de que no maximo cinco pessoas da nossa amostra obtiveram orientações sobre gravidez, ou seja, estamos interessados na probabilidade acumulada da binomial com

$n = 10$ e $p = 0,79$ avaliada no ponto $k = 5$. No exemplo anterior estávamos interessados na probabilidade da ocorrência de 5 sucessos, e agora, buscamos a probabilidade da ocorrência de 5, 4, 3, 2, 1 e 0 sucessos, $P(X \leq 5)$. A figura abaixo ilustra a probabilidade desejada.



Podemos encontrar a probabilidade desejada fazendo:

```
pbinom(q = 5, size = 10, prob = 0.79)
```

```
[1] 0.03986239
```

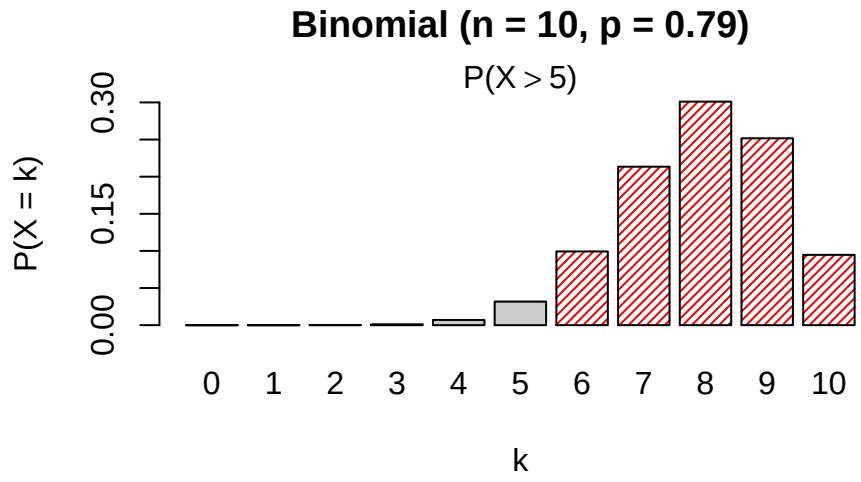
Como $P(X \leq 5) = P(X = 5) + P(X = 4) + P(X = 3) + P(X = 2) + P(X = 1) + P(X = 0)$, repare que se calcularmos as probabilidades separadamente utilizando dbinom e somarmos todas, devemos encontrar o mesmo resultado obtido com o comando pbinom.

Veja:

```
dbinom(x = 5, size = 10, prob = 0.79) + dbinom(x = 4, size = 10, prob = 0.79) +
dbinom(x = 3, size = 10, prob = 0.79) + dbinom(x = 2, size = 10, prob = 0.79) +
dbinom(x = 1, size = 10, prob = 0.79) + dbinom(x = 0, size = 10, prob = 0.79)
```

```
[1] 0.03986239
```

E se quisermos obter $P(X > 5)$? Neste caso, queremos descobrir a probabilidade de mais de cinco pessoas da nossa amostra terem recebido orientações sobre gravidez, como vemos na figura abaixo.



Sabemos que $P(X > 5) = 1 - P(X \leq 5)$, então:

```
1 - pbinom(q = 5, size = 10, prob = 0.79)
```

```
[1] 0.9601376
```

O comando pbinom nos fornece uma facilidade que nos permite obter a mesma probabilidade encontrada acima de forma mais direta. Basta utilizar o parâmetro lower.tail como "FALSE", Veja:

```
pbinom(q = 5, size = 10, prob = 0.79, lower.tail = FALSE)
```

```
[1] 0.9601376
```

Se o comando lower.tail não for especificado, o R vai interpretá-lo como "TRUE", calculando a probabilidade dos valores acumulados à esquerda, $P(X \leq 5)$. Definindo o parâmetro como "FALSE", o R retorna a probabilidade dos valores acumulados à direita, $P(X > 5)$.

4.1.3 Cálculo de quantis da distribuição

A função qbinom nos fornece o caminho "inverso" da função pbinom. Quando utilizamos qbinom estamos interessados em obter os quantis da distribuição binomial.Se queremos obter o quantil referente ao percentil de ordem $0,75$, estamos interessados num determinado valor de sucessos tal que a probabilidade acumulada neste valor seja de pelo menos 75%. Para calcular tal valor fazemos

```
qbinom(p = 0.75, size = 10, prob = 0.79)
```

```
[1] 9
```

Nos exemplos anteriores, mostramos que na binomial de parâmetros $n = 10$ e $p = 0,79$, a probabilidade de no máximo 5 sucessos é $0,03986239$. Portanto, se utilizarmos a função

qbinom(0.03986239, size = 10, prob = 0.79), devemos obter o valor 5 como resposta. Veja abaixo

```
qbinom(p = 0.03986239, size = 10, prob = 0.79)
```

```
[1] 5
```

4.1.4 Geração de amostras aleatórias seguindo modelo binomial

Por motivo de curiosidade, vamos introduzir a função rbinom para exemplificar um experimento modelado pela distribuiçao Binomial. Suponha que um aluno esteja fazendo uma prova de 20 questões independentes entre si e a probabilidade do estudante acertar cada questão é de $0,4$. Utilizando a função *rbinom*, é como se estivessemos simulando a aplicação da prova para o aluno em questão. Portanto, nosso experimento conterá 20 replicações de um experimento bernoulli (acertar ou errar cada questão) e a probabilidade de sucesso em cada replicação é $p = 0,4$. Utilizamos o código:

```
rbinom(n = 1, size = 20, prob = 0.4)
```

```
[1] 6
```

Com esta função simulamos um experimento binomial ($n = 1$) com 20 replicações de experimentos bernoulli ($size = 20$) e obtivemos 10 sucessos em nosso experimento, isto é, o estudante considerado teria acertado 10 questões da prova.

E se desejassemos simular a aplicação da prova de 20 questões para 5 alunos, considerando que todos eles tem probabilidade constante de $0,4$ de acertar cada questão na avaliação? Nosso experimento seria simulado da seguinte maneira:

```
rbinom(n = 5, size = 20, prob = 0.4)
```

```
[1]  7  4 10 10  7
```

Neste caso o parâmetro $n = 5$ se refere ao número de experimentos bernoulli, isto é, número de alunos que participam da prova. O resultados obtidos com a execução da função representam o número de acertos de cada aluno na avaliação com 20 questões.

4.2 Distribuição Poisson

É comum encontrarmos problemas em que o número de ocorrência de determinado evento é modelado a partir da distribuição Poisson. Neste tópico apresentaremos as funções bem similares às que vimos na seção anterior, mas adequada a distribuição Poisson, são elas: ppois, dpois e qpois.

As probabilidades de eventos envolvendo a distribuição de Poisson, com uma taxa de ocorrência

λ, podem ser calculadas através da seguinte função de probabilidade:

$$P(X = k) = \frac{e^{-\lambda}\lambda^k}{x!}.$$

4.2.1 Cálculo de probabilidades

Assim como em dbinom, a função dpois nos retorna a probabilidade associada a um valor de X. Uma diferença entre dbinom e dpois são os parâmetros que passamos para a função, devido à característica de cada modelo. No caso da binomial, devemos informar o tamanho do experimento e a probabilidade de sucesso, na Poisson precisamos informar apenas o valor do parâmetro λ.

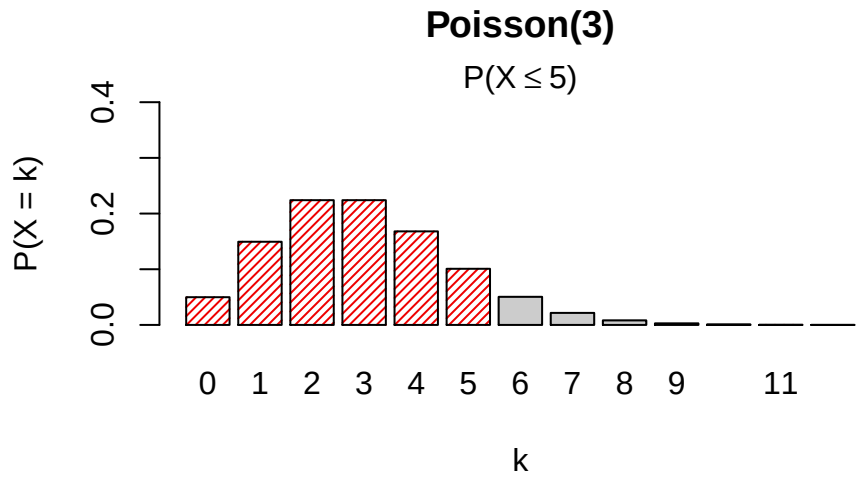
Para exemplificarmos, suponha que num consultório de fisioterapia cheguem em média 3 pacientes a cada uma hora. Suponha também que o número de pacientes que chegam neste consultorio é uma variável aleatória com distribuição Poisson. Estamos interessados na probabilidade de, num dia qualquer, chegarem 5 pacientes em uma hora no consultório. Usando a função dpois, considerando $\lambda = 3$, calculamos o valor de $P(X = 5)$ com:

```
dpois(x = 5, lambda = 3)
```

```
[1] 0.1008188
```

4.2.2 Cálculo de probabilidades acumuladas

A função ppois, por sua vez, retorna o valor da prababilidade acumulada avaliada em um valor k qualquer. No nosso exemplo, suponha que desejamos calcular a probabilade de no máximo 5 pacientes irem ao consultório ($P(X \leq 5)$). Desejamos calcular a probabilidade destacada no gráfico a seguir.



Para calcular tal probabilidade bastaria utilizar o comando ppois da seguinte maneira:

```r
ppois(q = 5, lambda = 3)
```

```
[1] 0.9160821
```

Assim como fizemos com a função pbinom, também podemos encontrar a probabilidade desejada somando várias probabilidade obtidas com a função dpois. Veja a seguir:

```r
dpois(x = 5, lambda = 3) + dpois(x = 4, lambda = 3) + dpois(x = 3, lambda = 3) +
dpois(x = 2, lambda = 3) + dpois(x = 1, lambda = 3) + dpois(x = 0, lambda = 3)
```

```
[1] 0.9160821
```

Se o interesse for calcular $P(X > 5)$, podemos fazer:

```r
ppois(q = 5, lambda = 3, lower.tail = FALSE)
```

```
[1] 0.08391794
```

A utilização da do parâmetro *lower.tail* foi discutida anteriormente na subseção referente a função pbinom.

4.2.3 Cálculo de quantis da distribuição

Continuando com o consultório de fisioterapia abordado anteriormente, suponha que queiramos encontrar um número x de pacientes tal que, com pelo menos 95% de probabilidade, a quantidade de visitantes no consultório seja menor ou igual do que x. Desta forma, estamos interessados em obter o quantil (percentil) de ordem 95% da distribuição Poisson com $\lambda = 3$. Utilizamos a função qpois para encontrar o quantil desejado:

```r
qpois(p = 0.95, lambda = 3)
```

```
[1] 6
```

Observamos que em pelo menos 95% dos dias o consultório recebe no máximo 6 pessoas.

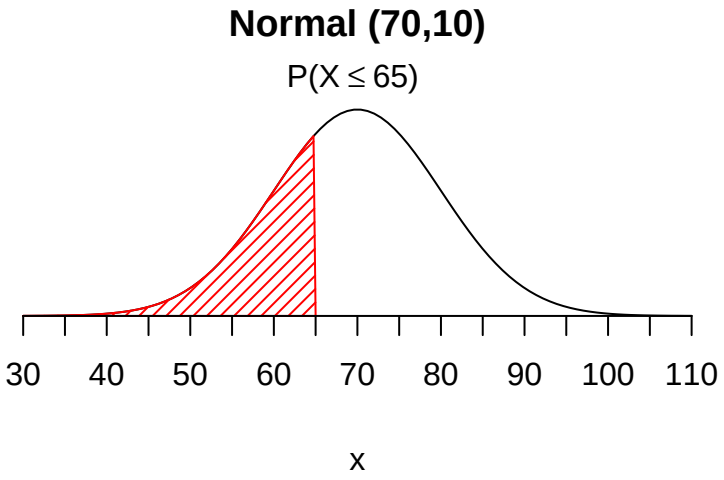
4.3 Distribuição Normal

A distribuição normal é amplamente utilizada na prática estatística para modelar uma série de eventos da natureza. Com ela é possível obtermos a probabilidade de ocorrência de eventos referentes às algumas variáveis contínuas. Por exemplo, se queremos descobrir a probabilidade de encontrarmos alunos com um peso menor do que 70kg. Sabemos que a variável aleatória peso é contínua, e se a variável peso seguir uma distribuição Normal, podemos obter facilmente o valor da probabilidade deseja utilizando o R.

No dia a dia das disciplinas de estatística, geralmente precisamos padronizar a variávle e ter em mãos a tabela da distribuição normal padrão para encontrarmos as probabilidades desejadas nos exercícios. Com o auxílio do R conseguimos obter tais probabilidades sem a necessidade de padronização. Nas subseções seguintes ilustramos o uso das funções ligadas a distribuição normal.

4.3.1 Cálculo de probabilidade acumulada e de intervalos

Suponha que um pesquisador esteja interessado em estudar um arbusto da região do cerrado. Sabe-se que o diâmetro do tronco deste arbusto é uma variável aleatória com distribuição Normal com média 70 e desvio padrão 10, X~Normal $(\mu = 70, \sigma^2 = 100)$. Queremos obter a probabilidade do pesquisador encontrar um arbusto com um tronco de diâmetro menor do que 65 cm, $P(X \leq 65)$, ou seja, a área representada no gráfico abaixo:



Se desenvolvessemos essa operação passo a passo, como deve ser feito para a utilização da tabela da distribuição normal padrão, deveríamos usar a formula

$$Z = \frac{X - \mu}{\sigma}.$$

Usando o R como calculadora, tal transformação seria calculando a padronização e com o resultado (-0.5) e buscaríamos a probabilidade $P(Z \leq -0.5)$ na $Normal(\mu = 0, \sigma^2 = 1)$. Veja:

```
(65 - 70)/10
```

```
[1] -0.5
```

```
pnorm(q = -0.5, mean = 0, sd = 1)
```

```
[1] 0.3085375
```

Também poderíamos utilizar a função pnorm sem definir os parâmetros *mean* e *sd* pois o R considera a distribuição Normal padrão. Veja abaixo:

```
pnorm(q = -0.5)
```

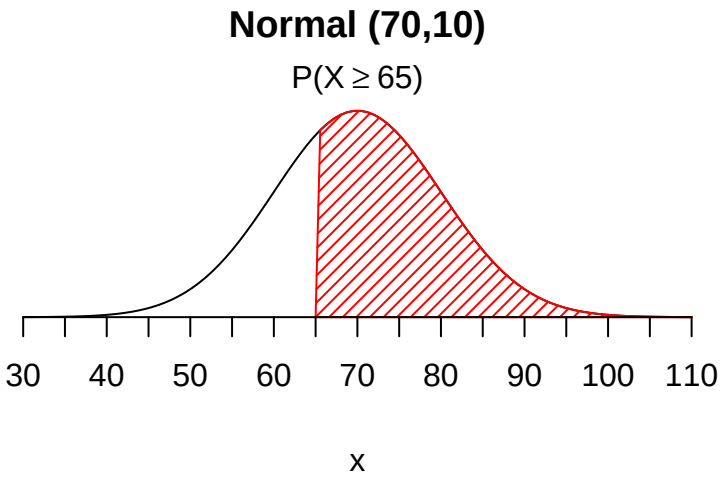
```
[1] 0.3085375
```

Podemos obter a probabilidade desejada diretamente ao informar os valores dos parâmetros de média e desvio padrão da distribuição, como é feito abaixo:

```
pnorm(q = 65, mean = 70, sd = 10)
```

```
[1] 0.3085375
```

E se quisermos obter a probabilidade $P(X > 65)$, área ilustrada no gráfico abaixo, basta fazermos como foi mostrado na função pbinom, definindo o parâmetro *lower.tail* como "FALSE" ou simplesmente "F". Também podemos utilizar o conceito de probabilidade de evento complementar. Observe a seguir:




```
pnorm(q = 65, mean = 70, sd = 10, lower.tail = F)
```

```
[1] 0.6914625
```

```
1 - pnorm(q = 65, mean = 70, sd = 10)
```

```
[1] 0.6914625
```

Algumas vezes, temos o interesse na probabilidade da variável aleatória X assumir valor em um intervalo, por exemplo, $P(50 \leq X \leq 65)$. Nestes casos, podemos utilizar que para variáveis contínuas temos que

$$P(50 \leq X \leq 65) = P(X \leq 65) - P(X \leq 50)$$

e, portanto, tal probabilidade pode ser calculada com

```
pnorm(q = 65, mean = 70, sd = 10) - pnorm(q = 50, mean = 70, sd = 10)
```

```
[1] 0.2857874
```

4.3.2 Cálculo de quantis da distribuição

Em alguns exercícios de probabilidade encontramos problemas similares a, dada uma variável $X \sim Normal(\mu = 10, \sigma^2 = 25)$, pedem o valor de x tal que a $P(X \leq x) = 0,8$. Nestas situações, possuímos o valor da probabilidade, mas não temos o valor de x. Em outras palavras, queremos encontrar o percentil de ordem 80 da distribuição $Normal(10, 25)$. Para resolver este problema, utilizamos a função qnorm:

```
qnorm(p = 0.8, mean = 10, sd = 5)
```

```
[1] 14.20811
```

Agora, dada uma variável $Y \sim Normal(\mu = 0, \sigma^2 = 1)$, qual o valor de y que corresponde à $P(Y > y) = 0,7$? Este valor pode ser obtido com o seguinte comando:

```
qnorm(p = 0.7, mean = 0, sd = 1, lower.tail = F)
```

```
[1] -0.5244005
```

Outro exemplo, muito utilizado na resolução de testes de hipóteses é encontrar z tal que $P(Z \leq z) = 0,975$, em que Z segue distribuição Normal padrão. Neste caso, temos

```
qnorm(0.975)
```

```
[1] 1.959964
```

4.3.3 Geração de amostras aleatórias seguindo o modelo Normal

Em provas e listas de exercícios, nos deparamos com amostras retiradas de uma determinada distribuição normal e nos perguntamos:

De onde são coletadas tais amostras, como os professores obtêm estes valores?

Podemos gerar amostras de qualquer distribuição normal com facilidade utilizando a função rnorm:

```
rnorm(n = 15, mean = 10, sd = 5)
```

```
 [1]  9.901430 20.368740  3.036020 18.930698  4.386243  3.900417  1.961474
 [8]  5.947010 18.242702 17.472476 13.610932 14.469717  5.758917  4.748408
[15]  9.986745
```

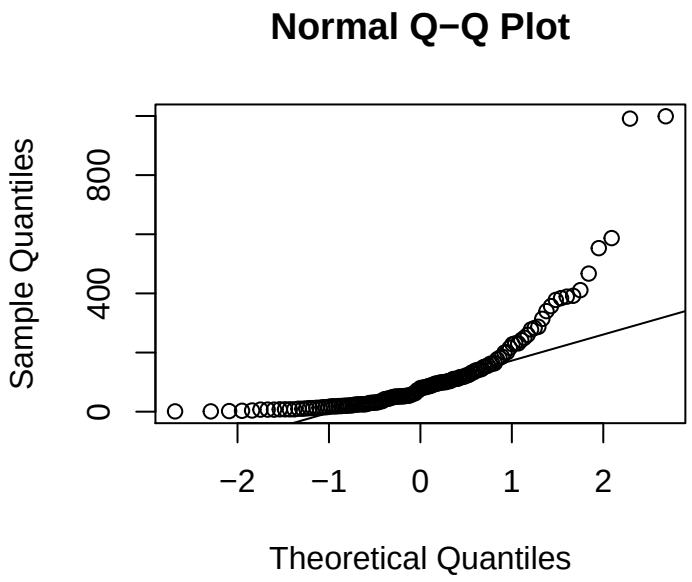
Através do comando acima, coletamos 15 valores oriundos de uma distribuição Normal com média $\mu = 10$ e variância $\sigma^2 = 25$.

4.3.4 Avaliando normalidade dos dados com qqnorm

Nas práticas profissionais, é comum encontrar problemas em que devemos checar a normalidade dos dados que temos em mãos para que possamos aplicar testes que demandam o pressuposto de normalidade. Para tal, utilizaremos a função qqnorm, uma ferramenta gráfica que nos permite checar se determinada amostra se aproxima de uma distribuição Normal, ao comparar

quantis empíricos (observados na amostra) com os quantis teóricos da distribuição. Novamente iremos usar o banco de dados *cancer* do pacote BancosBio para exemplificar o processo. Selecionamos a variável stime, que representa o tempo de sobrevivência ou acompanhamento em dias. No gráfico apresentado abaixo, os pontos são referentes a cada observação amostral e suas coordenadas representam os valores dos quantis empíricos e teóricos da distribuição Normal padrão. A linha representa a situação em há total concordância entre quantis dos dados e os quantis da distribuição Normal, isto é, se os pontos ficam próximos da reta, podemos dizer que a distribuição Normal é adequada para representar o comportamento daquele conjunto de dados. Veja abaixo como utilizar tal função.

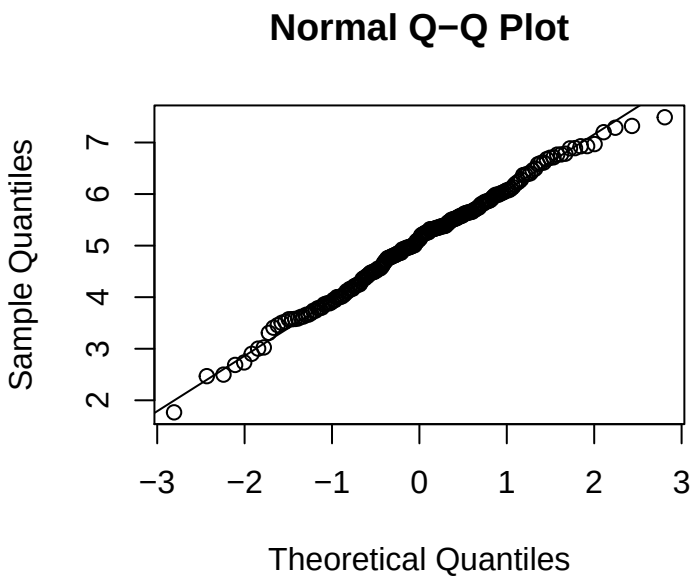
```
qqnorm(cancer$stime)
qqline(cancer$stime, distribution = qnorm)
```



Observando o gráfico, nota-se que os pontos não estão tão próximas da reta, principalmente no que diz respeito as extremidades da reta. Portanto, temos evidências à favor da não normalidade da variável stime do banco de dados câncer.

Para efeito de comparação, vamos gerar uma amostra oriunda de uma distribuição normal e avaliar a normalidade desta amostra. Para tal consideramos a função rnorm como pode ser visto a seguir,

```
amostra = rnorm(200, mean = 5, sd = 1)
qqnorm(amostra); qqline(amostra)
```



Desta vez, nota-se que os pontos se aproximam bastante da reta, ou seja, temos evidências à favor do fato de que a amostra foi coletada de uma população normal, o que condiz com o que já sabíamos sobre a amostra devido a maneira como os dados foram gerados.

4.4 Outras distribuições

Além das distribuições abordadas nesta apostila, o R fornece funções similares às que discutimos para outras distribuições de probabilidade. Para obter informações sobre cada comando, utilize o help das funções desejadas. Abaixo listamos as funções para algumas das distribuições mais comuns:

- **Distribuição Geométrica**: pgeom, dgeom, qgeom
- **Distribuição Binomial Negativa**: pnbinom, dnbinom, qnbinom
- **Distribuição Hipergeométrica**: phyper, dhyper, qhyper
- **Distribuição Qui-Quadrado**: pchisq, qchisq
- **Distribuição T de Student**: pt, qt

4.5 Distribuições de Probabilidade no R Commander

As funções apresentadas neste capítulo também podem ser utilizadas no R Commander, e de maneira bem simples. Para executar as funções desejadas basta clicar em "Distribuições", depois escolher o tipo de distribuição desejada (Contínua ou Discreta), em seguida selecionar a distribuição desejada (Normal, Binomial, Poisson etc), por fim, clicar na função que será processada e configurar os parâmetros necessários.

Por exemplo, suponha que desejamos calcular um quantil da distribuição Normal. Podemos seguir os passos abaixo:

vá em Distribuições -> Distribuições Contínuas -> Distribuição Normal -> Quantis da Normal... Depois de seguir esses passos, basta definir os parâmetros da distribuição e o quantil desejado.

Se por outra lado, desejamos calcular uma probabilidade acumulada referente a uma variável aleatória com distribuição Binomial, seguimos os seguintes passos:

vá em Distribuições -> Distribuições Discretas -> Distribuição Binomial -> Probabilidades das caudas da Binomial... e configure os parâmetros de forma adequada a probabilidade desejada.

4.6 Exercícios

Todos os exercícios que seguem devem ser feitos exclusivamente no R ou RStudio.

1. Para $X \sim Binomial(n = 20,\ p = 0,3)$, calcule:
   a. $P(X \leq 3)$
   b. $P(X = 3)$
   c. $P(5 < X \leq 15)$
   d. $P(12 < X < 18)$
   e. $P(X > 10|X \geq 3)$

2. Para $X \sim Poisson(\lambda = 100)$, calcule:
   a. $P(50 < X \leq 70)$
   b. $P(X > 200)$
   c. $P(X < 100|X > 50)$
   d. $P(50 < X < 100)$
   e. $P(X \leq a) = 0,2$. Qual o valor de a?

3. Para X $\sim$ Normal($\mu = 100$, $\sigma = 10$), calcule:
   a. $P(-3\sigma \geq X \leq 3\sigma)$
   b. $P(X > x) = 0,875$. Qual o valor de x?
   c. $P(100 - a < X < 100 + a) = 0.8$. Qual o valor de a?
   d. $P(X > 50|X \leq 100)$
   e. $P(X \geq 50|X \leq 100)$

4. Um dado não viciado é jogado 6 vezes em sequência. Levando em consideração que as jogadas são independentes umas das outras, calcule a probabilidade de se obter o número 4 duas vezes.

5. Considere o mesmo experimento do exercicio 4. Desta vez, calcule a probabilidade de se obter um número menor do que 2 três vezes.

6. Suponha que ao longo dos anos em uma dada escola, 4 entre 36 estudantes obtém nota nos exames de biologia. Suponha que uma nova turma com 36 estudantes seja escolhida ao acaso. Qual a probabilidade de 10 alunos tirarem nota máxima em um exame de biologia. Suponha também que as notas dos alunos são independentes entre si.

7. Num hospital veterinário, a taxa média de entrada de animais é de 10 por hora. Fazendo as suposições necessárias, calcule:
   a. A probabilidade de chegarem 27 animais em três horas neste hospital.
   b. A probabilidade de chegarem 10 animais em apenas meia hora.
   c. A probabilidade de não chegar nenhum animal em seis horas.

8. Num centro de saúde, estima-se que, em média, 2 pacientes solicitem os enfermeiros a cada meia hora. Dado que em 15 minutos não houve nenhuma solicitação, calcule a probabilidade de que ocorram 3 solicitações nos próximos 15 minutos.

9. Sabe-se que o tempo de cura de um tratamento contra determinado tipo de doença segue uma aproximadamente distribuição normal com $\mu = 60$ dias e desvio padrão de $\sigma = 15$ dias. Calcule a probabilidade de um paciente atingir a cura em no máximo 40 dias.

10. Assumindo que o tamanho de determinada planta tenha distribuição normal com media e desvio padrão dados, respectivamente, por $\mu = 10$ cm e $\sigma = 3$ cm, calcule:
    a. A probabilidade de encontrar uma planta com menos de 5 cm.
    b. A probabilidade de encontrar uma planta que tenha entre 6cm e 10 cm.
    c. A probabilidade de que em uma amostra de 20 plantas, 15 tenham mais do que 12 cm.
    d. A probabilidade de que em uma amostra de 50 plantas, todas tenham entre 9 cm e 12 cm.

Capítulo 5

Intervalo de Confiança e Teste de Hipóteses

5.1 Intervalos de Confiança

Quando queremos estimar um determinado parâmetro a partir de uma amostra, uma média por exemplo, temos duas opções:

a. Podemos fornecer uma estimativa pontual. No caso de uma média populacional, a estimativa pontual é uma média amostral;

b. Podemos fornecer uma estimativa intervalar. No caso da média, calculamos a média amostral e adicionamos uma margem de erro a nossa estimativa.

Suponha que desejemos fazer uma pesquisa com os alunos da UFMG no intuito de estimar o peso médio de todos os estudantes da universidade. Sabemos que se calculássemos o peso de todos os estudantes, encontraríamos o peso médio real (média populacional) e não haveria nenhum erro associado a nossa média. Entretando, não é simples obter informações de uma população inteira, como é o caso de todos os estudantes da UFMG. Portanto, a metodologia consiste em obter uma amostra de alunos e então estimar a média da população em questão. Mas devemos ter em mente o fato de que, quando obtemos diferentes amostras e calculamos uma média, muito provavelmente, iremos encontrar diferentes valores para cada uma das amostras. Neste sentido, a estimativa intervalar, ao incluir uma margem de erro na estimativa pontual obtida, leva em consideração a variabilidade amostral e inclui a informação sobre o nível de incerteza da estimativa (denotado por nível de confiança).

Intervalo de Confiança para a Média

Para obter uma estimativa intervalar para a média μ de uma população utilizamos informações tais como a média amostral, o desvio padrão populacional ou amostral, e algumas informações referentes às distribuições Normal e t-Student. Existem duas alternativas para a construção do intervalo de confiança neste caso e são dada por:

a. Quando a amostra observada segue distribuição normal e conhecemos o valor do desvio

padrão populacional σ, o intervalo de $(1-\alpha)100$ de confiança é dado por

$$IC_{(1-\alpha)} = \left[\bar{x} - z_{1-\alpha/2}\frac{\sigma}{\sqrt{n}}, \quad \bar{x} + z_{1-\alpha/2}\frac{\sigma}{\sqrt{n}}\right],$$

em que encontramos $z_{1-\alpha/2}$ na tabela da distribuição normal padrão tal que $P(Z \leq z_{1-\alpha/2}) = 1 - \alpha/2$.

Ainda que os dados observados não sigam distribuição normal, se o tamanho amostral for grande o suficiente podemos utilizar este intervalo devido ao fato de que a média amostral converge para a distribuição normal (Teorema Central do Limite).

b. Se os dados observados tem distribuição normal, mas a variância populacional não é conhecida, devemos utilizar a variância amostral para estimar a variância populacional (ver Variância e Desvio Padrão) e o intervalo de $(1-\alpha)100$ de confiança é dado por

$$IC_{(1-\alpha)} = \left[\bar{x} - t_{1-\alpha/2}\left(\frac{s}{\sqrt{n}}\right), \quad \bar{x} + t_{1-\alpha/2}\left(\frac{s}{\sqrt{n}}\right)\right],$$

em que encontramos $t_{1-\alpha/2}$ na tabela da distribuição t-Student com $n-1$ graus de liberdade tal que $P(T \leq t_{1-\alpha/2}) = 1 - \alpha/2$.

Exemplo 1

Suponha que queiramos investigar o nível médio de glicose dos alunos de uma turma de Bioestatística. Considere também que o nível de glicose desta população (turma de estudantes) segue uma distribuição Normal. Sendo assim, coletamos o nível de glicose de 15 alunos e obtivemos uma média amostral $\bar{x} = 96{,}8$ mg/dL. Sabemos também que, nesta turma, o desvio padrão populacional para a medida de glicose é $\sigma = 5$ mg/dL.

Visto que conhecemos o desvio padrão da medida na população, utilizaremos a primeira formulação para encontrar o intervalo de $\gamma = 0,95 = 1 - \alpha$ de confiança. Também sabemos que $\bar{x} = 96.8$ e $\sigma = 5$. Desta forma, fazemos

$$IC_{0.95}(\mu) = \left[96.8 - z_{1-\alpha/2}\left(\frac{5}{\sqrt{15}}\right) \quad , \quad 96.8 + z_{1-\alpha/2}\left(\frac{5}{\sqrt{15}}\right)\right].$$

Para obter $z_{1-\alpha/2}$, encontramos que $1 - (\alpha/2) = 1 - (0,05/2) = 0,975$. Assim, podemos obter $z_{1-(\alpha/2)} = z_{0.975}$, o percentil de ordem 97,5% da distribuição normal padrão (veja função qnorm), através de:

```
qnorm(p = 0.975, m = 0, s = 1)
```

```
[1] 1.959964
```

Arredondando para 3 casas decimais (no R podemos utilizar round(x = 1.959964, digits = 3) para obter o arredondamento com 3 casas decimais) obtemos $z_{0.975} = 1,96$. Logo o intervalo

de 95% de confiança para μ é dado por

$$IC_{95\%}(\mu) = \left[96,8 - 1,96\left(\frac{5}{\sqrt{15}}\right), \quad 96,8 + 1,96\left(\frac{5}{\sqrt{15}}\right)\right].$$

Podemos calcular os limites do intervalo no R fazendo

```
96.8 - 1.96*(5/sqrt(15))
```

```
[1] 94.26965
```

```
96.8 + 1.96*(5/sqrt(15))
```

```
[1] 99.33035
```

Assim, obtemos $IC_{95\%}(\mu) = [94,27;\ 99.33]$. Ou seja, com 95% de confiança, o nível médio de glicose dos alunos de uma turma de Bioestatística está entre 94,27 mg/dL e 99,33 mg/dL.

Exemplo 2

Agora, para exemplificar a segunda formulação do intervalo de confiança, utilizaremos o banco de dados *Caranguejos* do pacote BancosBio. Gostaríamos de fazer uma estimativa intervalar para o tamanho do lobo frontal de caranguejos em geral, considerando as observações da variável *FL* do banco de dados como uma amostra aleatória.

Não conhecemos a variância populacional do tamanho do lobo frontal de carangueijos e por isso devemos estimar estem parâmetros através da amostra que contém 200 observações de caranguejos. Note abaixo como podemos encontrar o tamanho amostral, a média e o desvio padrão amostral.

```
length(caranguejos$FL)
```

```
[1] 200
```

```
mean(caranguejos$FL)
```

```
[1] 15.583
```

```
round( sd(caranguejos$FL), digits = 3)
```

```
[1] 3.495
```

Como foi mostrado anteriormente, $1 - (\alpha/2) = 0,975$. Encontramos o valor $t_{0,975}$, através do percentil de ordem 97,5% da distribuição t-Student com $n - 1 = 199$ graus de liberdade e para tal, consideramos o resultado abaixo

```
# Arredondando para 3 casas decimais
round( qt(p = 0.975, df = 199), digits = 3)
```

```
[1] 1.972
```

Neste caso o intervalo de 95% de confiança para μ é dado por

$$IC_{(0.95)}(\mu) = \left[15,583 - 1,972\left(\frac{3,495325}{\sqrt{200}}\right), \quad 15,583 + 1,972\left(\frac{3,495325}{\sqrt{200}}\right)\right],$$

que pode ser calculado por

```
15.583  - 1.972 * (3.495 / sqrt(200))
```

```
[1] 15.09565
```

```
15.583  + 1.972 * (3.495 / sqrt(200))
```

```
[1] 16.07035
```

Desta forma obtemos que $IC_{(0.95)}(\mu) = [15,096, \quad 16,070]$. Com 95% de confiança temos que o tamanho médio do lobo frontal de carangueijos está entre 15,096 e 16,04.

Intervalo de Confiança para Proporções

Agora, vamos investigar como fazer um intervalo de confiança para uma proporção populacional. Para exemplificar, suponha que uma pesquisadora queira estimar a proporção de pessoas de determinada cidade que levaram seus animais domésticos para vacinarem contra a raiva. Para isto a pesquisadora selecionou 200 indivíduos aleatoriamente e os entrevistou. Dos 200 entrevistados, apenas 167 disseram que levaram os animais para vacinar. Neste caso, a estimativa pontual é a proporção amotral dada por

$$\hat{p} = \frac{167}{200} = 0,835.$$

A pesquisadora decidiu fazer um intervalo com 99% de confiança para estimar a verdadeira parcela de pessoas da cidade que cumpriram as orientações sanitárias e participaram da campanha de imunização contra a raiva. No caso de uma proporção populacional, o intervalo de confiança é obtido com

$$IC_{(1-\alpha)}(p) = \Big[\hat{p} - E, \quad \hat{p} + E\Big],$$

em que E é a margem de erro que pode ser obtida com duas diferentes abordagens.

Com a abordagem otimista, estimamos a margem de erro por $E = z_{1-(\alpha/2)}\left(\frac{\sqrt{\hat{p}(1-\hat{p})}}{\sqrt{n}}\right)$, em que encontramos $z_{1-\alpha/2}$ na tabela da distribuição normal padrão tal que $P(Z \leq z_{1-\alpha/2}) = 1-\alpha/2$. Este nome de otimista decorre do fato o intervalo de confiança é consequência do fato que o teorema Central do Limite garante que para tamanho amostral suficientemente grande, $\hat{p} \sim N\left(p, \frac{p(1-p)}{n}\right)$. Nesta abordagem estimamos $\frac{p(1-p)}{n}$ por $\frac{\hat{p}(1-\hat{p})}{n}$. A segunda abordagem consiste em estimar $\frac{p(1-p)}{n}$ por $\frac{0,25}{n}$, que surge ao consideramos a maior variância possível (obtida quando $p = 0,5$). Neste caso, dizemos que utilizamos a abordagem conservadora.

Para obter o intervalo de 99% de confiança, como é o interesse da pesquisadora, precisamos obter $z_{1-(\alpha/2)} = z_{0.995}$. Isto pode ser feito através da função qnorm, como é apresentado abaixo.

```
#Arredondando para 3 casas decimais
round( qnorm(p = 0.995, m = 0, s = 1), digits = 3 )
```

```
[1] 2.576
```

Neste caso, vamos mostrar o intervalo obtido apenas com a abordagem conservatora. Com esta abordagem, o intervalo de 99% confiança para proporção populacional de pessoas que participaram da campanha de imunização contra a raiva é dado por

$$IC_{(0.99)}(p) = \left[0,835 - 2,576\left(\frac{\sqrt{0,25}}{\sqrt{200}}\right), \quad 0.835 + 2,576\left(\frac{0,25}{\sqrt{200}}\right)\right].$$

No R, obtemos o intervalo como mostrado abaixo.

```
0.835 - 2.576*( sqrt(0.25) / sqrt (200) )
```

```
[1] 0.7439246
```

```
0.835 + 2.576*( sqrt(0.25) / sqrt (200) )
```

```
[1] 0.9260754
```

Desta forma, obtemos $IC_{(0.99)}(p) = [0,744,\ 0,926]$. Com 99% de confiança, a proporção populacional de pessoas que participaram da campanha de imunização contra a raiva está entre 74,4% e 92,6%.

5.2 Teste de Hipóteses

Em muitas situações práticas, é comum que tenhamos que decidir se uma afirmação a respeito de determinado assunto é verdadeira ou não, em outras palavras, devemos investigar nosso problema e checar se existem evidências que favorecem tal afirmação.

No contexto estatístico, as afirmações são chamadas de hipóteses, e o procedimento de tomada de decisão é chamado de teste de hipóteses. Ou seja, uma hipótese estatística é uma afirmação sobre os parâmetros de uma ou mais populações.

Por exemplo, considere a variável *FL* (lobo frontal do caranguejo) do banco de dados *Caranguejos* que mede o tamanho do lobo frontal do animal. Suponha que você é um pesquisador ou uma pesquisadora da área e acredita que o valor médio de *FL* é de 15 mm Então você opta por coletar uma amostra de caranguejos e verificar se o valor hipotético é plausível. Neste caso, você elabora as seguintes hipóteses:

$$H_0 : \mu = 15 \ \times \ H_1 : \mu \neq 15.$$

A hipótese H_0 é chamada de hipótese nula, enquanto H_1 é chamada de hipótese alternativa. O procedimento de teste consiste em avaliar se os dados corroboram ou não com a hipótese nula. Quando os dados apontam que a hipótese nula é improvável, dizemos que rejeitamos a hipótese nula. Um aspecto importante é que ao formularmos as hipóteses, o sinal de igualdade sempre deve estar na hipótese nula.

5.2.1 Teste de Hipóteses para a Média de Uma População

Suponha que $X_1, X_2, ..., X_n$ é uma amostra aleatória proveniente de uma distribuição normal com média μ e variância σ^2 desconhecidas. Neste contexto, a estatística de teste

$$T = \frac{\bar{X} - \mu}{S/\sqrt{n}}$$

terá distribuição t-Student com n-1 graus de liberdade.

Fixo um nível de significância, podemos decidir pela rejeição ou não da hipótese nula com base na região crítica dada por

$$RC = \{t_{obs} < -t_{1-\alpha/2} \quad \text{ou} \quad t_{obs} > t_{1-\alpha/2}\},$$

em que $P(T \leq t_{1-\alpha/2}) = 1 - \alpha/2$.

Rejeitamos a hipótese nula se o valor observado da estatística de teste pertence à região crítica. Também podemos simplesmente avaliar se o p-valor do teste é menor que o nível de significância, e se esse for o caso, rejeitamos a hipótese nula. Uma outra alternativa para o teste de hipóteses é checar se o valor de hipótese nula está contido no intervalo de confiança fornecido pelo teste: caso o valor de média de H_0 esteja dentro do intervalo, temos um indício de que a hipótese em questão é verdadeira; caso contrário, temos um indício de que a hipótese nula é falsa. É importante frisar que o teste de hipóteses utilizado intervalo de confiança pode ser sempre aplicado somente em problemas que envolvem a média. No caso, da proporção, o intervalo de confiança nem sempre vai nos dar uma informação válida.

Considere o exemplo citado anteriormente, em que desejamos testar uma hipótese a respeito da variável *FL* medida nos caranguejos. Abaixo iremos calcular a estatística de teste.

```
xbar = mean(caranguejos$FL) # Média amostral
S = sd(caranguejos$FL)  # Desvio padrão amostral
n= length(caranguejos$FL) # Tamanho da amostra

To = (xbar -15)/(S/sqrt(n)) # Estatística de teste
To
```

```
[1] 2.358826
```

O valor acima é o valor observado da estatística de teste. Agora precisamos compará-la com o valor de referência da distribuição t, podemos fazer isso usando a tabela ou o comando abaixo:

```r
a= 0.05 # nível de significancia de 5%
qt(1-a/2, n-1)
```

```
[1] 1.971957
```

Como $|t_{obs}| > t_{\alpha/2,n-1}$ concluímos que devemos rejeitar H_0, ou seja, existem evidência amostrais que apontam que a média de FL é diferente de 15. Também poderíamos ter optado por testar essa hipótese avaliando o p-valor da estatística de teste. Veja abaixo:

```r
2*pt(To, n-1,lower.tail = F) #2 vezes a area acima da estatística de teste
```

```
[1] 0.01930187
```

Note que o p-valor é menor que o nível de significância, portanto a conclusão é que devemos rejeitar H_0. Acima mostramos como se constrói passo a passo um teste t no R, porém tudo isso já está implementado na função abaixo:

```r
t.test(caranguejos$FL, mu = 15)
```

```
    One Sample t-test

data:  caranguejos$FL
t = 2.3588, df = 199, p-value = 0.0193
alternative hypothesis: true mean is not equal to 15
95 percent confidence interval:
 15.09562 16.07038
sample estimates:
mean of x
   15.583
```

Os resultados que essa função retornam são os mesmos calculados acima e, adicionalmente, ela também retorna o intervalo de 95% de confiança para a média ("95 percent confidence interval"). Você pode alterar o nível de confiança usando o argumento *conf.level*. Além disso, também podemos testar hipoteses unilaterais alterando o argumento *alternative*. Nesta situação escolhemos *alternative* = *less* quando a hipótese alternativa contém o sinal $<$ (e a H_0 contém $\geq$) e *alternative* = *greater* se a hipótese alternativa contém o sinal $>$.

No R Commander

Para realizar um teste para média no R Commander clique em:

Estatísticas > Médias > Teste t para uma amostra

Selecione a variável que deseja testar, especifique a hipótese alternativa, a hipótese nula e o nível de confiança.

5.2.2 Teste de Hipóteses para a Diferença das Médias de Duas Populações

Suponha que $X_1, X_2, ..., X_n$ e $Y_1, Y_2, ..., Y_m$ são duas amostras aleatórias independentes que seguem uma distribuição normal. Essas amostras são provenientes de dois grupos distintos, os quais desejamos comparar.

Na maioria do casos, quando se coletam os dados, os pesquisadores não têm uma informação inicial a respeito da variância dos dados. Logo, os dois grupos podem ter a mesma variância, ou não (independente se a média dos grupos é a mesma). Portanto, a seguir consideraremos dois cenários: um em que as variâncias dos dois grupos são consideradas iguais, e outro em que consideramos que as variâncias podem ser diferentes.

Novamente, considere a variável *FL* do banco de dados *Caranguejos*. Imagine que desejemos avaliar se a média da variável *FL* (lobo frontal) entre machos ($\bar{X}$) e fêmeas ($\bar{Y}$) é a mesma. Uma pesquisadora acredita que os machos são maiores que as fêmeas e então, levantamos as seguintes hipóteses do teste unilateral:

$$H_0 : \mu_x \leq \mu_y \quad \times \quad H_1 : \mu_x > \mu_y.$$

A seguir consideramos as duas situações sobre das variâncias populacionais serem iguais ou não.

Amostras com Variâncias iguais

Quando assumimos que a variâncias dos dois grupos são iguais a estatística de teste tem distribuição t-Student com n+m-2 graus de liberdade e é dada da seguinte maneira:

$$T = \frac{(\bar{X} - \bar{Y}) - (\mu_x - \mu_y)}{S_p\sqrt{\frac{1}{n} + \frac{1}{m}}}.$$

No R podemos calcular o valor observado para a estatística de teste como no apresentado a abaixo.

```
# Seleciona os carangueijos machos
X = caranguejos$FL[caranguejos$sex == "M"]

# Seleciona as carangueijos femeas
Y = caranguejos$FL[caranguejos$sex == "F"]

# Identifica o tamanho das amostras
```

```r
n = length(X)
m = length(Y)

# Média amostral dos grupos
xbar = mean(X)
ybar = mean(Y)

# Variancias
Sx2 = var(X)
Sy2 = var(Y)
Sp2 = ( (n-1)*Sx2 + (m-1)*Sy2 )/( n+m-2 )

# Desvio padrao
Sp = sqrt(Sp2)

# Estatística de teste
To = (xbar - ybar)/( Sp*sqrt((1/n) + (1/m)) )
To
```

```
[1] 0.6099835
```

Agora precisamos comparar o valor observado da estatística de teste com o valor de referência da distribuição t-Student ($RC = \{t_{obs} > t_{1-\alpha}\}$, com $P(T \leq t_{1-\alpha}) = 1 - \alpha$), ou seja, o quantil da distribuição t-Student relacionado ao valor crítico no teste de hipóteses. Podemos fazer a comparação utilizando a tabela t ou o comando da função qt, supondo um nível de significância de 1%.

```r
a = 0.01 # nível de significancia de 1%
qt(1-a, n+m-2) # Valor que deixa uma area de 99% abaixo
```

```
[1] 2.345328
```

Como o valor observado da estatística de teste é menor que o valor crítico, concluímos que não devemos rejeitar a hipótese nula e, assim, com 1% de significância, a diferença de médias observada não pode ser considerada significativamente maior do que zero. Portanto, temos indícios de que caranguejos machos, em média, não possuem tamanho do lobo frontal maior do que das fêmeas.

Vejamos agora qual o p-valor do teste no código abaixo.

```r
pt(To, n+m-2,lower.tail = F)
```

```
[1] 0.2712861
```

Como o p-valor é maior que o nível de significância estabelecido não rejeitamos a hipótese nula. Vejamos agora como realizar toda análise acima por meio de apenas da função t.test.

```r
t.test(x=X, y=Y, alternative = "greater", var.equal = TRUE, conf.level = 0.99)
```

```
    Two Sample t-test

data:  X and Y
t = 0.60998, df = 198, p-value = 0.2713
alternative hypothesis: true difference in means is greater than 0
99 percent confidence interval:
 -0.8591611        Inf
sample estimates:
mean of x mean of y
   15.734    15.432
```

Os argumentos indicados na função acima são os grupos (x e y), a hipótese alternativa (alternative = "greater") que indica que a media do grupo x é maior que a do grupo y. Também informamos que as variâncias dos grupos são iguais (var.equal = TRUE) e informamos que o nível de confiança desejado para o intervalo é de 99% (conf.level = 0.99).

Amostras com Variâncias diferentes

Quando é verificado que as variâncias dos dois grupos são diferentes, devemos usar outra estatística de teste que também possui distribuição t-Student. A diferença ocorrem em como as variâncias populacionais são estimadas e no valor dos graus de liberdade da distribuição t-Student.

No Help do R temos que o padrão do comando *t.test* é o argumento *var.equal = FALSE*, isso significa que a função *t.test* assume que as variâncias são desconhecidas e diferentes. Para fazer o teste no exemplo anterior considerando variâncias populacionais diferentes podemos simplesmente executar o seguinte comando

```r
t.test(x=X, y=Y, alternative = "greater")
```

Na prática, não conhecemos o valor real das variâncias para decidir se elas são iguais, mas podemos aplicar um teste de hipóteses para variâncias populacionais para decidir a respeito disso. Para mais detalhes consulte a função *var.test*.

No R Commander

Para realizar um teste para comparação da média de duas populações clique em:

Estatísticas > Médias > Teste t para amostras independentes.

Selecione a variável que distingue os elementos entre grupos e a variável que deseja comparar. Você também pode alterar o tipo de hipótese alternativa e o nível de confiança clicando em Opções.

Caso o R Commander não permita a escolha de uma variável categorica para definir os grupos devido a esta variável possuir códigos numéricos, você deve informá-lo que uma de suas variáveis é categórica e distingue os indivíduos em grupos diferentes. Para isso clique em:

Dados > Modificação de variáveis no conjunto de dados > Converter variável numérica para fator.

Selecione a variável categórica desejada, você pode alterar os nomes de cada grupo ou simplesmente usar números.

5.2.3 Teste de Hipóteses para a Diferença de Médias populacionais com amostras pareadas

No caso de amostras pareadas (ou dependentes), também podemos utilizar a função *t.test* para realizar o teste de hipóteses desejado. Neste caso, teremos que especificar o argumento *alternative* definindo as hipóteses escolhidas e especificar o argumento *paired = TRUE*.

No R Commander

Para realizar um teste para comparação da média de duas populações com amostras dependentes, clique em:

Estatísticas > Médias > Teste t (dados pareados).

Selecione a variável que distingue os elementos entre grupos e a variável que deseja comparar. Você também pode alterar o tipo de hipótese alternativa e o nível de confiança clicando em Opções.

5.2.4 Teste de Hipóteses para Proporção

Testes para proporção são utilizados quando desejamos comparar se a proporção de indivíduos com determinada classificação é igual a algum valor hipotético. Podemos pensar na proporção de indivíduos com determinada característica, ou com determinado tipo de doença, por exemplo. Além disso, teste de hipóteses para proporções também são comumente utilizados em pesquisas de opinião pública.

Quando testamos afirmativas referentes a proporções, temos as seguintes hipóteses: $\$H_0 : p = p_0 \quad \times \quad H_1 : p \neq p_0$.

Sob H_0, para n (tamanho amostral) consideravelmente grande, podemos definir a distribuição amostral de $\hat{p}$ como aproximadamente Normal, isto é,

$$\hat{p} \sim N\left(p, \frac{p(1-p)}{n}\right).$$

Disto, obtemos que $Z = \frac{\hat{p}-p}{\sqrt{\frac{p(1-p)}{n}}} \sim N(0,1)$.

Por exemplo, considere que um médico que trabalha com pacientes com câncer desconfia que 60% dos pacientes de todo o hospital, que estão em tratamento, nunca realizaram um tratamento prévio. Então ele formula as seguintes hipóteses: $\$H_0 : p = 0,6 \quad \times \quad H_1 : p \neq 0,6$.

Neste caso, p é a proporção de pacientes que não realizaram um tratamento anteriormente. Para descobrir se sua hipótese realmente faz sentido, ele coleta uma amostra de pacientes e observa quantos realizaram tratamento anterior. Essas informações estão disponíveis no banco de dados *Cancer* e são apresentadas abaixo:

```
table(cancer$prior)
```

```
 0 10
97 40
```

Na amostra, pode ser observado que 97 dos 137 pacientes não haviam realizado tratamento anterior (variável codificada com 0 no banco de dados), o que corresponde a uma proporção amostral de 70,8%.

Considerando um nível de significância $\alpha = 0,05$, damos continuidade calculando o valor observado da estatística de teste a seguir:

```
Z = ((97/137)-0.6)/sqrt((0.6*(1-0.6))/137)
Z
```

```
[1] 2.581046
```

O valor da estatística de teste foi $z_o = 2,58$. Com base neste valor, calculamos o p-valor do teste dado por:

```r
pnorm(Z, lower.tail = F)*2
```

```
[1] 0.009850134
```

Como o p-valor do teste é menor que o nível de significância decidimos por rejeitar a hipótese nula. Ou seja, existem evidências de que a proporção de pacientes no hospital que não realizaram tratamento prévio é diferente de 60%. Vemos abaixo o comando que realiza todo o teste para uma proporção populacional.

```r
prop.test(x = 97, n = 137, p = 0.6, correct = F)
```

```
    1-sample proportions test without continuity correction

data:  97 out of 137, null probability 0.6
X-squared = 6.6618, df = 1, p-value = 0.00985
alternative hypothesis: true p is not equal to 0.6
95 percent confidence interval:
 0.6270518 0.7776586
sample estimates:
        p
0.7080292
```

Note que o p-valor do teste bilateral foi igual ao que encontramos anteriormente.

O argumento *correct* indica se a função se deve ou não ser usada com correção de continuidade, normalmente utilizada quando aproximamos uma variável aleatória discreta por uma variável aleatória contínua. Os demais argumentos são o número de elementos na amostra que possuem a característica avaliada, o tamanho da amostra e o valor do parâmetro definido em H_0.

O teste também nos retorna o intervalo de 95% de confiança para o valor da proporção populacional, que nesse caso resultou no intervalo de 62,7% a 77,7%, ou seja, com uma confiança de 95%, a proporção de pacientes que realizou tratamento prévio está entre 62,7% e 77,7%.

A função *prop.test* também pode ser utilizada para testar hipóteses envolvendo duas proporções populacionais. Consulte *help(prop.test)* para ver os detalhes desta função.

No R Commander

Para realizar um teste para proporção no R Commander clique em:

Estatísticas > Frequências/Proporções > Teste de Frequência/Proporção (1 amostra)

Selecione a variável que deseja testar. Na aba de opções você pode alterar a hipótese alternativa, o valor da hipótese nula, o nível de confiança e o tipo do teste.

5.2.5 Teste Qui-Quadrado

O teste qui-quadrado pode ser utilizado para testar hipóteses de independência, homogeneidade e adequação de distribuições. Apresentamos como este teste pode ser utilizado no contexto de independência através do banco de dados *Estudante*, considere as variáveis *Smoke* e *Sex*. A variável *Smoke* indica a frequência que o estudante fuma, e então, suponha que você deseja investigar se o hábito de fumar é diferente entre estudantes do sexo masculino e feminino. Para responder a essa pergunta são formuladas as seguintes hipóteses:

$$\begin{cases} H_o : \text{Sexo e Fumo são variáveis independentes} \\ H_1 : \text{Sexo e Fumo não são variáveis independentes.} \end{cases}$$

Para testar essas hipóteses os dados são organizados na tabela de contingência abaixo.

```
Tab=table(estudante$Sex,estudante$Smoke) #Tabela com as frequencias
addmargins(Tab) #Adiciona os totais
```

```
        Heavy Never Occas Regul Sum
  Female     5    99     9     5 118
  Male       6    89    10    12 117
  Sum       11   188    19    17 235
```

No teste qui-quadrado, utilizamos a seguinte estatística de teste:

$$\chi^2 = \sum_{j=1}^{r}\sum_{i=1}^{s} \frac{(o_{ij} - e_{ij})^2}{e_{ij}},$$

em que o_{ij} é a frequência observada para cada classe, e_{ij} é a frequência esperada de cada classe, r é o número de linhas da tabela e s o número de colunas. Note que quando as frequências observadas são muito próximas das esperadas o valor da estatística de teste é próximo de zero. Já quando os valores observados são bem distantes do esperado, o valor da estatística de teste aumenta.

Para testar a hipótese usamos o fato que χ^2 segue uma distribuição Qui-quadrado com $(r-1)(s-1)$ graus de liberdade. O valor esperado e_{ij} é obtido realizando o seguinte cálculo:

$$e_{ij} = \frac{\text{Total da linha } i \quad \times \quad \text{Total da coluna } j}{Total\ geral}.$$

Considerando a tabela acima temos, por exemplo, $e_{11} = \frac{118 \text{ X } 11}{235} = 5,52$ e $\frac{(5-5,52)^2}{5,52} = 0,049$.

Fazemos esse cálculo para todos outros termos, encontramos o seguinte valor observado para a estatística de teste $\sum_{j=1}^{r}\sum_{i=1}^{s}\frac{(o_{ij}-e_{ij})^2}{e_{ij}} = \frac{(5-5,52)^2}{5,52} + ... + \frac{(12-8,46)^2}{8,46} = 3,5536$.

Sob o nível de significância de $\alpha = 0.05$, podemos encontrar a região crítica do teste pelo comando *qchisq*, ao obter o quantil 0,95 da distribuição Qui-quadrado:

```
qchisq(p = 0.95, df = 3)
```

```
[1] 7.814728
```

Como o valor observado não pertence à região crítica, não rejeitamos H_0, ou seja, não existem evidências para rejeitar a hipótese de que Sexo e Fumo sejam variáveis independentes.

Para realizar essa mesma análise por meio de um único comando usamos o código abaixo:

```
chisq.test(estudante$Sex, estudante$Smoke)
```

```
    Pearson's Chi-squared test

data:  estudante$Sex and estudante$Smoke
X-squared = 3.5536, df = 3, p-value = 0.3139
```

O comando acima nos retorna o valor da estatística de teste igual a $3,5536$ e o respectivo p-valor. Considerando um nível de significância de $\alpha = 0,05$, vemos que o p-valor é maior que α e, portanto, não devemos rejeitar a hipótese nula.

No R Commander

Para realizar um teste Qui-quadrado no R Commander clique em:

Estatísticas > Tabelas de Contingência > Tabela de dupla entrada.

Selecione as variáveis categóricas que se deseja testar a associação ou independência e clique em "Ok".

5.2.6 Testes de normalidade

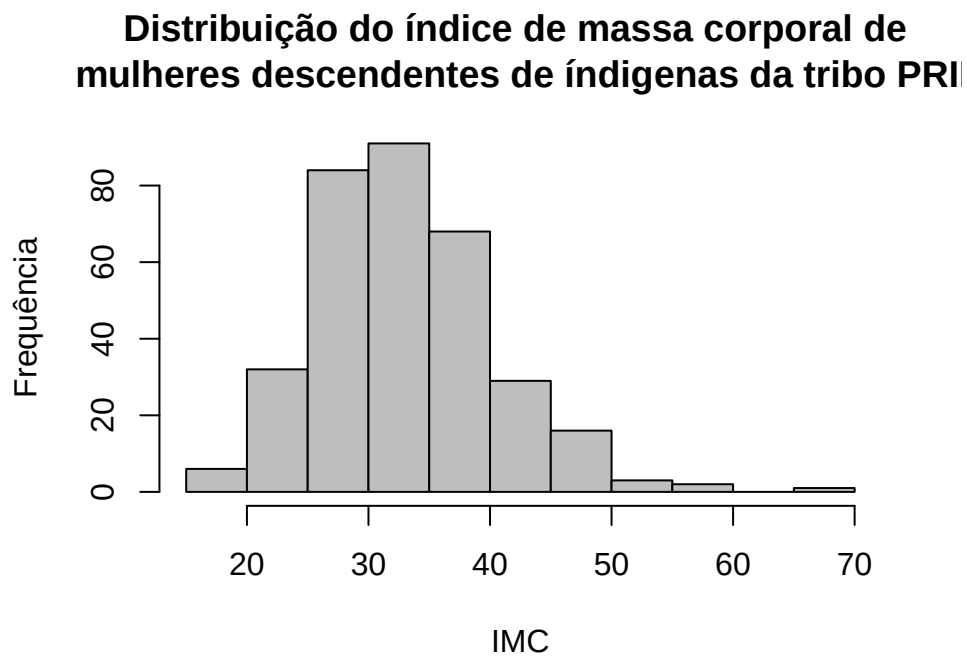
Vimos anteriormente que em muitas metodologias é necessário assumir a suposição de normalidade dos dados. Um primeiro passo para avaliar o pressuposto de normalidade dos dados é fazer um histograma e avaliar se o comportamento é similar a uma curva em forma de sino. Também vimos, na Seção 4.3.4, que podemos contruir um gráfico qqnorm para avaliar este pressuposto. Além disso, existem vários testes de hipóteses para testar se existem evidências amostrais para concluirmos que os dados seguem distribuição Normal. Nestes testes, as hipóteses são:

$$\begin{cases} H_o : \text{A variável de interesse segue distribuição Normal} \\ H_1 : \text{A variável de interesse segue qualquer outra distribuição de probabilidade.} \end{cases}$$

Aqui iremos apresentar os comandos para realizar seis diferentes testes de normalidade, sendo eles os testes de Kolmogorov-Smirnov, Shapiro-Wilk, Lilliefors, Cramér-von Mises, Shapiro-Francia, Anderson-Darling e Qui-quadrado de Pearson. Destes testes apresentados, os cinco últimos estão implementados no pacote "nortest".

Para ilustrar o uso das funções para realizar os testes de normalidade, iremos usar o banco de dados *diabetes* do pacote BancosBio para exemplificar o processo e selecionamos a variável bmi, que representa o índice de massa corporal. Inicialmente apresentaos um histograma destes dados:

```
hist(diabetes$bmi, main =  "Distribuição do índice de massa corporal de
     mulheres descendentes de índigenas da tribo PRIMA",
     xlab = "IMC", ylab = "Frequência", col = "grey")
```



Observando o histograma percebemos que o conjunto de dados é unimodal e apresenta assimetria à direita, trazendo indícios de que a normalidade pode não ser adequada, principalmente levando em conta que o banco de dados tem 532 observações.

Abaixo apresentamos os comandos necessárias para executar os testes de normalidade citados. Note que para o teste de Kolmogorov-Smirnov é necessário especificar os parâmetros de média e desvio padrão da distribuição. Isto pode ser feito considerando as estimativas obtidas a partir da média amostral e da variância amostral.

```
library(nortest)
dados <- diabetes$bmi
media <- mean(dados)
dp <- sd(dados)

#------------------------------------------------------------------------
ks.test(dados, "pnorm", mean = media, sd = dp) # Kolmogorov-Smirnov
```

```
    One-sample Kolmogorov-Smirnov test

data:  dados
D = 0.053239, p-value = 0.3035
alternative hypothesis: two-sided
```

```
#------------------------------------------------------------------------
shapiro.test(dados) # Shapiro-Wilk
```

```
    Shapiro-Wilk normality test

data:  dados
W = 0.96679, p-value = 6.917e-07
```

```
#------------------------------------------------------------------------
lillie.test(dados) # Lilliefors
```

```
    Lilliefors (Kolmogorov-Smirnov) normality test

data:  dados
D = 0.053239, p-value = 0.02428
```

```
#------------------------------------------------------------------------
cvm.test(dados) # Cramér-von Mises
```

```
    Cramer-von Mises normality test
```

```
data:  dados
W = 0.25515, p-value = 0.001112
```

```
#-----------------------------------------------------------------------
sf.test(dados) # Shapiro-Francia
```

```
    Shapiro-Francia normality test

data:  dados
W = 0.9659, p-value = 2.146e-06
```

```
#-----------------------------------------------------------------------
ad.test(dados) # Anderson-Darling
```

```
    Anderson-Darling normality test

data:  dados
A = 1.6618, p-value = 0.0002849
```

```
#-----------------------------------------------------------------------
pearson.test(dados) # Qui-quadrado de Pearson
```

```
    Pearson chi-square normality test

data:  dados
P = 26.645, p-value = 0.08591
```

```
#-----------------------------------------------------------------------
```

Observando os resultados e considerando um nível de significância de 5%, vemos que todos os testes não levam a mesma conclusão, mas a maioria deles leva a rejeição da hipóteses nula de normalidade dos dados. Apenas os testes de Kolmogorov-Smirnov e Qui-quadrado de Pearson levaram a não rejeição da hipótese de normalidade.

No R Commander

Para fazer um teste de normalidade no R Commander vá em

Estatísticas -> Resumos-> Test of normality.

Após percorrer este caminho é possível escolher qual teste quer realizar, além de ser possível estratificar o teste de normalidade, isto é, realizar o teste em subconjuntos do banco de dados definidos a partir de uma variável categórica.

5.3 Exercícios

Todos os exercícios que seguem devem ser feitos exclusivamente no R ou RStudio.

1. Considere uma amostra qualquer de tamanho $n = 17$, com média amostral $\bar{x} = 31,7$. Sabemos que o desvio padrão populacional é $\sigma = 3,8$. Calcule um intervalo de confiança para a média, com 5% de significância.

2. Construa um intervalo de confiança de 95% para a média populacional da variável bp do banco de dados *Diabetes*.

3. Construa um intervalo de confiança com 1% de significância para a média da variável bp do banco de dados *Diabetes*. Compare com o intervalo obtido no exercício anterior.

4. Considere uma pesquisa feita com uma amostra aleatória com 150 pessoas para avaliar a quantidade de fumantes em determinada cidade. Considerando que 115 pessoas disseram ser não fumantes. Construa um intervalo de confiança com 98% de confiança para a verdadeira proporção de fumantes da cidade em questão.

5. Na seção 5.3.3 aprendemos a utilizar a função *rnorm* para gerar aleatóriamente dados provenientes de uma distribuição normal. Utilizando essa função, gere três amostras de uma distribuição normal com média $\mu = 100$ e desvio padrão $\sigma = 25$. A primeira amostra gerada de tamanho $n = 10$, a segunda de tamanho $n = 30$ e a terceira de tamanho $n = 100$. Para cada uma das amostras, considerando um nível de significância $\alpha = 0,05$ e variância desconhecida, teste as seguintes hipóteses:

   a. $H_0 : \mu = 110$ vs $H_1 : \mu \neq 110$
   b. $H_0 : \mu \geq 110$ vs $H_1 : \mu < 110$
   c. $H_0 : \mu \leq 110$ vs $H_1 : \mu > 110$

6. Considerando o banco de dados *Cancer*, pede-se:

   a. Um médico que trabalha diretamente com os pacientes acredita que a média da idade dos pacientes em tratamento é de 50 anos. Formule as hipóteses e realize o teste adequado para verificar se o médico tem razão. Utilize $\alpha = 5\%$.

   b. O médico também desconfia que a média da variável Karn, escore de Karnofsky do desempenho do paciente, é de 60 pontos. Formule as hipóteses e realize o teste adequado para verificar se o médico tem razão. Conclua utilizando o intervalo de confiança e compare os resultados em dois níveis confiança diferentes.

7. Um professor de Estatística I coletou informações de seus alunos por meio de um questionário, os resultados estão no banco de dados "estudante". Usando os testes adequados teste as hipóteses abaixo levantadas pelo professor. Utilize $\alpha = 5\%$.

   a. Os alunos do sexo masculino, em média, são maiores que as alunas do sexo feminino.
      a. A pulsação das alunas, em média, é maior que a dos alunos.

b. Em geral os alunos do sexo masculino se exercitam mais que as alunas do sexo feminino.
c. 20% dos alunos são fumantes. Dica: Você pode encontrar o número total de não fumantes usando o código abaixo.

```
# Retorna TRUE (1) se o aluno nunca fumou ou FALSE (0)
# Se o aluno fuma com alguma frequência
Fumantes <- estudante$Smoke == "Never"

# Soma o total de não fumantes
sum(Fumantes, na.rm = T)
# na.rm = T desconsidera os alunos que não responderam
```

8. Interessado em estudar os efeitos da diabetes no corpo humano, um pesquisador coletou informações de 332 pacientes, os resultados estão disponíveis no banco de dados *Diabetes*. Agora seu papel é ajudar os pesquisadores a responderem as seguintes dúvidas:

   a. Qual a proporção de diabéticos na população de estudos? Forneça um intervalo de 99% de confiança.
   b. Existe diferença na média de pressão do sangue entre os pacientes que possuem e os que não possuem diabetes? a. Existe diferença na média do índice de massa corporal entre os pacientes que possuem e os que não possuem diabetes. a Ter diabetes altera a concentração de glucose?

9. Considerando o banco de dados aids, utilize o teste de hipóteses adequado para investigar se existe associação entre o sexo do paciente e a categoria de transmissão da doença.

10. Considerando o banco de dados hospital, suponha que um enfermeiro que trabalha na triagem dos pacientes acredita que 90% dos pacientes admitidos não recebem antibióticos. Formule as hipóteses e realize o teste adequado para verificar se o enfermeiro tem razão. Interprete e comente os resultados.

11. O banco de dados seeds.txt contém diversas informações a respeito de três variedades de trigo. Considerando esse banco de dados, pede-se:

a. Leia o banco de dados no ambiente do R.
b. Compare as variedades 1 e 2 em relação a variável “Area”.
c. Compare as variedades 3 e 2 em relação a variável “Compactidade”.
d. Compare as variedades 1 e 3 em relação a variável “Perimetro”.

Capítulo 6

Apêndice

I. Código referente ao gráfico da página 26

```
binomial = dbinom(c(0:10), size = 10, prob = 0.7) # atribui a uma variável
# a probabilidade de ocorrência de todos os valores possíves de uma
# binomial de tamanho 10 (11 valores)

names(binomial) = 0:10 # para cada valor de probabilidade,
# atribui-se o nome referente
# ao valor ao qual a probabilidade está associada

cols = rep("grey80", 11) # cria-se um vetor de 11 entradas repetidas
# para atibuir a cor grey80 a cada
# barra do gráfico da binomial

densities = rep(1000,11) #cria-se um vetor de 11 entradas repetidas
# para atibuir a densidade 1000 a cada barra do gráfico da binomial.
# Uma densidade baixa nos retornará um gráfico com barras hachuradas

cols[6] = "red2" # a posição 6 do vetor cols agora será red2 e não grey80,
# pois estamos interessados em destacar a sexta barra do gráfico

densities[6] = 30 # a sexta barra do gráfico terá uma densidade menor,
# de modo que ela fique hachurada e se diferencie das demais

barplot(binomial, col = cols, main =  "Binomial (n = 10, p =0.7)",
xlab = "k", ylab = "P(X = k)", ylim = c(0,0.3), density = densities)
# barplot cria um grafico de barras,
# main eh o titulo do grafico, xlab o nome do eixo x,
# ylab o nome do eixo y, ylim sao os limites do eixo y,
# col e density sao as cores e densidades que atribuimos anteriormente
```

```
mtext(text =("P(X = 5)")) # este comando adiciona um pequeno texto
# na parte superior do grafico
```

II. Código referente ao gráfico da página 30

```
poison = dpois(c(0:12), lambda = 3) # atribui a uma variável
# a probabilidade de ocorrência de todos os valores possíves de uma
# distribuição Poisson de tamanho 10 (11 valores)
names(poison) = 0:12 # para cada valor de probabilidade,
# atribui-se o nome referente
# ao valor ao qual a probabilidade está associada

cols = rep("grey80", 11) # cria-se um vetor de 11 entradas repetidas
# para atibuir a cor grey80 a cada
# barra do gráfico da binomial

densities = rep(1000,11) #cria-se um vetor de 11 entradas repetidas
# para atibuir a densidade 1000 a cada barra do gráfico da binomial.
# Uma densidade baixa nos retornará um gráfico com barras hachuradas

cols[0:6] = "red2" # as posições de 0 a 6 do vetor cols
# agora será red2 e não grey80,
# pois estamos interessados em destacar a primeira,
# segunda, terceira, quarta, quinta e sexta barra do gráfico

densities[0:6] = 30 # a 1a, 2a, 3a, 4a, 5a e 6a barra do gráfico
# terão uma densidade menor,
# de modo que elas fiquem hachuradas e se diferenciem das demais

barplot(poison, col = cols, main =  "Poisson(3)",
xlab = "k", ylab = "P(X = k)", ylim = c(0,0.4), density = densities)
#  barplot cria um grafico de barras,
# main eh o titulo do grafico, xlab o nome do eixo x,
# ylab o nome do eixo y, ylim sao os limites do eixo y,
# col e density sao as cores e densidades que atribuimos anteriormente

mtext(text = expression("P(X" <= "5)")) # este comando adiciona um
# pequeno texto na parte superior do grafico
```

III. Código referente ao gráfico da página 32

```
x = seq(-4,4, length=100)*10+70 #Cria-se uma sequência de tamanho 100
# com valores de -4 a 4.
```

```r
# Cada valor é transformado de acordo com a Normal de média 100
# e desvio padrão 10.
# A escolha de -4 a 4 se deve pela falcilidade de reconhecer a amplitude
# do intervalo como se estivéssemos trabalhando com a normal padrão.
# Ao multiplicar por 10 e somar 70, é como se estivéssemos transformando
# para a normal de nosso interesse.

y = dnorm(x, 70, 10) # Para cada valor que x assume,
# atribui-se um valor no eixo y
# através da função dnorm. Os valores de dnorm não são probabilidades.

plot(x, y, type="l", xlab="X", ylab="", main="Normal (70,10)", axes= F)
# O comando plot cria um gráfico associando os valres de x e de y.
# Type = "l" define que os pontos serão ligados por uma linha.
# Xlab atribui um nome ao eixo x.
# Ylab atribui nome ao  eixo y, mas neste caso, deixaremos sem nome.
# Main atribui nome ao gráfico.
# Axes = F exclui a margem e os eixos do gráficos
# (optamos por deixar assim por motivo estético).

i = (x >= (-4*10+70)) & (x <= 65) #Nesta parte, temos uma variável i
# a qual são atribuídos valores TRUE ou FALSE (1 ou 0)
# para a condição que definimos.
# Esta condição foi definida de acordo com a área do gráfico
# que queremos destacar.
# i será TRUE se os valores de x
# forem maiores ou iguais a -4*10+70 e menores ou iguais a 65.
# Ou seja, queremos
# selecionar os valores menores do que 65.

polygon(x = c((-4*10+70), x[i], 65), y = c(0, y[i],0),
        col = "red", density = 20)
# o comando polygon nos permitirá colorir a área interessada,
# utilizando os valores  da condição que definimos acima.

# O valores de X serão associados aos valores de Y.
# Como a área é definida a partir de -4*10+70,
# associaremos este valor a y = 0.
# Como a área tem  o 65 como limitante superior, associamos o 65 a Y = 0.
# Os valores de X[i] para i = TRUE  serão associados aos valores de Y[i]
# para i = TRUE, pois definimos assim a nossa regiãod e interesse.
# Col define a cor que vamos utilizar para colorir a área e density define
# a densidade do colorido (uma densidade alta, colore a área por inteiro
# e uma densidade baixa deixa a área hachurada).
```

```r
mtext(text = expression("P(X" <= "65) "),side = 3) # este comando adiciona
# um pequeno texto na parte superior do grafico

axis(side = 1, at=seq(from = -4*10+70, to = 4*10+70, by = 5), pos=0)
# Como excluímos a margem do gráfico,
# adicionaremos um eixo x de nossa preferência
# localizado em side = 1 (parte inferior)
# e na altura 0 através do comando pos = 0.
# O eixo irá de -4*10+70 até 4*10+70 de 5 em 5 valores.
```

Referências

- R Core Team (2019). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.
- Venables, W. N. & Ripley, B. D. (2002) Modern Applied Statistics with S. Fourth Edition. Springer, New York. ISBN 0-387-95457-0
- Fox, J., and Bouchet-Valat, M. (2019). Rcmdr: R Commander. R package version 2.6-0.