

Bioestística usando o R

Petronio Fagundes de Oliveira Filho

2023-03-16

1 Introdução

1.1 Importância da Bioestatística

Os indivíduos variam em relação as suas características biológicas, psicológicas e sociais na saúde e na doença. Esta variabilidade gera uma grande quantidade de incertezas.

A Bioestatística, estatística aplicada às ciências biológicas e da saúde, é a ferramenta utilizada pelos pesquisadores para trabalhar com essas incertezas advindas da variabilidade. Várias definições foram escritas para a estatística, uma dela é a seguinte (1):

Estatística é a disciplina interessada com o tratamento dos dados numéricos obtidos a partir de grupos de indivíduos

A Bioestatística lida com a variabilidade humana utilizando técnicas estatísticas quantitativas (2) que ajudam a diminuir a ignorância em relação a esta diversidade. A compreensão da variabilidade humana torna a medicina mais ciência, diminuindo as incertezas, na tentativa de verificar se os resultados encontrados de fato existem ou são apenas obra do acaso.

Na década de 1990, houve um acesso maior aos computadores. Os profissionais da saúde não estatísticos passaram a ter mais interesse no campo da bioestatística. Isto gerou uma onda que facilitou o aparecimento de novas ferramentas estatísticas de ponta. Apesar disso, o conhecimento da Bioestatística permanece restrito aos especialistas na área.

Nos últimos anos, os pacotes de software foram aprimorados, tornando-se mais amigáveis e diminuindo significativamente o pânico ao se defrontar com uma série de números uma vez que a maioria deles exige apenas conhecimento básico de matemática.

Para a tomada de decisão em saúde é fundamental o acúmulo de conhecimento adquirido através da prática clínica, geradora da experiência do profissional, do intercâmbio com os pares e da análise adequada das evidências científicas publicadas em periódicos de qualidade. Para atingir este objetivo, é fundamental o conhecimento de bioestatística, incluindo aqui que o pensamento que deve nortear os profissionais da saúde ao lidar com o ser humano é o pensamento probabilístico.

1.2 Pílulas históricas da Estatística

A história deve começar em algum lugar, mas a história não tem começo (3)

Entretanto, é natural, que se trace as raízes voltando ao passado, tanto quanto possível. Alguns referem-se à curiosidade em relação ao registro de dados à dinastia Shang, na China, possivelmente no século XIII a.c,

com a realização de censos populacionais. Há relatos bíblicos de possíveis censos realizados por Moisés (1491 a.C.) e por Davi (1017 a.C.).

Os romanos e os gregos já realizavam censos por volta do século VIII a IV a.C. Em 578-534 a.C., o imperador *Servo Túlio* mandou realizar um censo de população masculina adulta e suas propriedades que serviu para estabelecer o recrutamento para o exército, para o exercício dos direitos políticos e para o pagamento de impostos. Os romanos fizeram 72 censos entre 555 a.C. e 72 d.C. A punição para quem não respondia, geralmente era a morte! Na Idade Média, na Europa, existem registros de diversos censos: durante o domínio muçulmano, na Península Ibérica, nos séculos VII a XV; no reinado de Carlos Magno (712-814) e ainda o maior registro estatístico feito na época, o *Domesday Book* (Figura 1), realizado na Inglaterra, por Guilherme I (3) , o Conquistador, onde registravam nascimentos, mortes, batismos e casamentos. Houve, também, recenseamentos nas repúblicas italianas no século XII ao XIII (4).

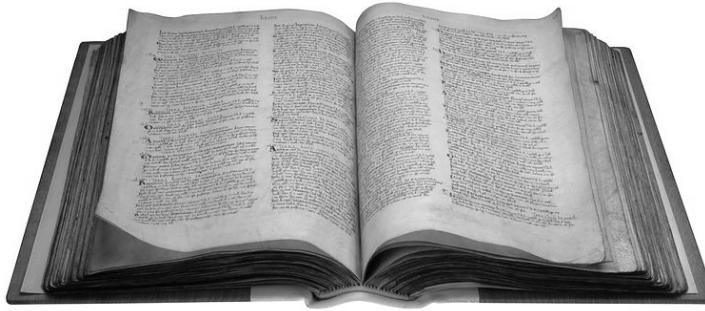


Figura 1: Domesday Book

Gottfried Achenwall (29/10/1719 – 01/05/1772) foi um filósofo alemão, historiador, economista e estatístico. Os economistas alemães reivindicam para ele o título de “Pai da Estatística”. Isto é contestado por escritores ingleses que reivindicam este título para *William Petty* (27/05/1623 – 16/12/1687) que foi um filósofo e cientista britânico que propôs a utilização de métodos quantitativos, que denominou de aritmética política, para a análise da riqueza de um país.

John Graunt (24/04/1620 – 18/04/1674) foi um cientista britânico a quem se deve vários estudos demográficos ingleses. Foi o precursor da construção de Tábuas de Mortalidade. Realizou estudos com *William Petty*. Em 1791, *Sir John Sinclair* (1754 – 1835) concebeu um plano de uma pesquisa empírica na Escócia para fornecer informações estatísticas. Foi a primeira vez que o termo estatística foi usado em inglês.

Girolamo Cardano (24/09/1501 – 21/09/1576) foi um médico, matemático, físico e filósofo italiano. É tido como o primeiro a introduzir ideias gerais da teoria das equações algébricas e as primeiras regras da probabilidade, descritas no livro *Liber de Ludo Aleae*, publicado em 1663. Descreveu pela primeira vez a clínica da febre tifoide. Foi amigo de Leonardo da Vinci.

Pierre-Simon Laplace, Marquês Laplace (23/03/1749 – 05/03/1927) foi um matemático, astrônomo e físico francês. Embora conduzisse pesquisas substanciais sobre física, outro tema principal dos esforços de sua vida foi a teoria das probabilidades. Em seu *Essai philosophique sur les probabilités*, Laplace projetou um sistema matemático de raciocínio indutivo baseado em probabilidades, que hoje coincidem com as ideias bayesianas.

Antoine Gombaud, conhecido como Chevalier de Méré (1607- 1684) foi um nobre e jogador. Como não tinha mais sucesso nos jogos de azar, buscou ajuda de *Blaise Pascal* (19/06/1623 – 19/08/1662), matemático, físico francês, que se correspondeu com *Pierre Fermat* (matemático e cientista francês), nascendo desta colaboração a teoria matemática das probabilidades (1812). *Blaise Pascal* foi mais tarde chamado de o Pai da Teoria das Probabilidades.

A moderna teoria das probabilidades foi atribuída a *Abraham De Moivre* (25/05/1667 – 27/11/1754), matemático francês, que adquiriu fama por seus estudos na trigonometria, teoria das probabilidades e pela equação da curva normal. Em 1742, *Thomas Bayes* (1701 – 07/04/1761, matemático e pastor presbiteriano,

inglês, desenvolveu o Teorema de Bayes que descreve a probabilidade de um evento ocorrer, baseado em um conhecimento *a priori*.

Adrien-Marie Legendre (18/09/1752 – 10/01/1833) foi um matemático francês. Em 1783, tornou-se membro adjunto da *Academie des Sciences*, instituição que esteve na vanguarda dos desenvolvimentos científicos dos séculos XVII e XVIII. Fez importantes contribuições à estatística, à teoria dos números e à álgebra abstrata.

Johann Carl Friedrich Gauss (30/04/1777 – 23/02/1855) foi um matemático, astrônomo e físico alemão (Figura 2) que contribuiu em diversas áreas das ciências como teoria dos números, estatística, geometria diferencial, eletrostática, astronomia e ótica. Muitos referem-se a ele como o Príncipe da Matemática, o mais notável dos matemáticos. Descobriu o método dos mínimos quadrados e a lei de Gauss da distribuição normal de erros e sua curva em formato de sino, hoje tão familiar para todos que trabalham com estatística.



Figura 2: Johann Carl Friedrich Gauss

Lambert Adolphe Jacques Quêtelet (22/02/1796 – 17/02/1874) foi um astrônomo, matemático, demógrafo e estatístico francês. Seu trabalho se concentrou em estatística social, criando regras de determinação de propensão ao crime

Francis Galton (16/02/1822 – 17/01/1911) foi um antropólogo, matemático e estatístico inglês. Entre muitos artigos e livros, criou o conceito estatístico de correlação e da regressão à média. Ele foi o primeiro a aplicar métodos estatísticos para o estudo das diferenças e herança humanas de inteligência. Criou o conceito de eugenia e afirmava que era possível a melhoria da espécie por seleção artificial. Acreditava que a raça humana poderia ser melhorada caso fossem evitados relacionamentos indesejáveis. Isto acompanhava o pensamento burguês europeu da época. Criou a psicometria, onde desenvolveu testes de inteligência para selecionar homens e mulheres brilhantes. Esta teoria teve papel importante na formação do fascismo e nazismo (5).

William Farr (30/11/1807 – 14/04/1883) foi um médico sanitário e estatístico inglês, nascido na vila de Kenley, Shropshire. Foi o primeiro investigador a examinar séries temporais de morbimortalidade para longos períodos e, assim, considerado o criador da Estatística da Saúde Pública Moderna. Seus relatórios foram fundamentais para o desencadeamento das reformas sanitárias britânicas, em meados e final do século XIX (6).

Florence Nightingale (12/05/1820 – 13/08/1910) foi uma enfermeira (Figura 3) que ficou famosa por ser pioneira no tratamento de feridos, durante a Guerra da Criméia (7). Ficou conhecida na história pelo apelido de “A dama da lâmpada”, pelo fato de servir-se de uma lamparina para auxiliar no cuidado aos feridos durante a noite. Também contribuiu no campo da Estatística, sendo pioneira na utilização de métodos de representação visual de informações, como por exemplo gráfico de setores (habitualmente conhecido como gráfico do tipo “pizza”)



Figura 3: Florence Nightingale

John Snow (York, 15/03/1813 - Londres, 15/03/1858) foi um médico inglês (Figura 4), considerado pai da Epidemiologia Moderna. Recebeu, em 1853, o título de Sir após ter anestesiado a rainha Vitória no parto sem dor de seu oitavo filho, Leopoldo de Albany. Este fato ajudou a divulgar a técnica entre os médicos da época. Demonstrou que a cólera era causada pelo consumo de águas contaminadas com matérias fecais, ao comprovar que os casos dessa doença se agrupavam em determinados locais da cidade de Londres, em 1854, onde havia fontes dessas águas (6).

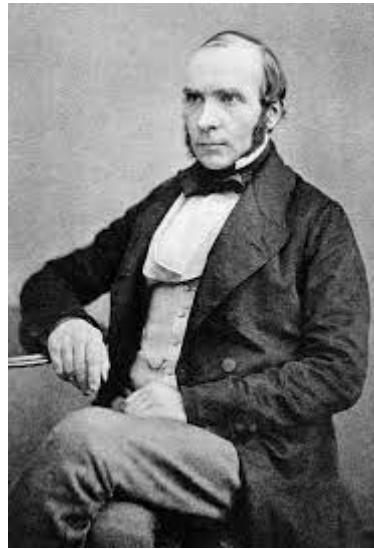


Figura 4: John Snow

Karl Pearson (27/03/1857 – 27/04/1936) foi um importante estatístico inglês, fundador do Departamento de Estatística Aplicada da *University College London* em 1911. Juntamente com Weldon e Galton fundou, em 1901, a revista *Biometrika* com o objetivo era desenvolver as teorias estatísticas, editada até os dias de hoje. O trabalho de Pearson como estatístico fundamentou muitos métodos estatísticos de uso comum, nos dias atuais: regressão linear e o coeficiente de correlação, teste do qui-quadrado de Pearson, classificação das distribuições (8).

Charles Edward Spearman (10/09/1863 – 17/09/1945) foi um psicólogo inglês conhecido pelo seu trabalho

na área da estatística, como um pioneiro da análise fatorial e pelo coeficiente de correlação de postos de Spearman. Ele também fez bons trabalhos de modelos da inteligência humana.

William Sealy Gosset (13/07/1876 – 16/10/1937) foi um químico e estatístico inglês (Figura 5). Em 1907, enquanto trabalhava químico da cervejaria experimental de Arthur Guinness & Son, criou a distribuição t que usou para identificar a melhor variedade de cevada, trabalhando com pequenas amostras. A cervejaria Guinness tinha uma política que proibia que seus empregados publicassem suas descobertas em seu próprio nome. Ele, então, usou o pseudônimo “Student” e o teste é chamado “t de Student” em sua homenagem (9).



Figura 5: William Sealy Gosset

Ronald Aylmer Fisher (17/02/1890 – 29/07/1962) foi um estatístico, biólogo e geneticista inglês. Em 1919, Fisher se envolveu com pesquisa agrícola no centro de experimentos de *Rothamsted Research*, em Harpenden, Inglaterra, e desenvolveu novas metodologias e teoria no ramo de experimentos (10). Durante sua vida, Fisher (Figura 6) escreveu 7 livros e publicou cerca de 400 artigos acadêmicos em estatística e genética . Em um dos seus livros, *The design of Experiments* (1935), Fisher relata um experimento que surgiu de uma pergunta curiosa: o gosto do chá muda de acordo com a ordem em que as ervas e o leite são colocados? Essa simples questão resultou em um estudo pioneiro na área e serviu de sustentação para análise da aleatorização de dados experimentais (9). Ronald A. Fisher foi descrito (11) como “um gênio que criou praticamente sozinho os fundamentos para o moderno pensamento estatístico”. Era muito temperamental. Seus atritos com outros estatísticos ficaram famosos, entre eles encontra-se ninguém menos do que Karl Pearson, outro notável estatístico.



Figura 6: Ronald A. Fisher

Austin Bradford Hill (08/07/1897 – 18 /04/1991) foi um epidemiologista e estatístico inglês (Figura 7), pioneiro no estudo do acaso nos ensaios clínicos e, juntamente com Richard Doll, foi o primeiro a demonstrar a ligação entre o uso do cigarro e o câncer de pulmão. Hill é amplamente conhecido pelos Critérios de Hill, conjunto de critérios para a determinação de uma associação causal (12).



Figura 7: Bradford Hill

John Wilder Tukey (16/06/1915 – 26/07/2000) foi um estatístico norte-americano. Desenvolveu uma filosofia para a análise de dados que mudou a maneira de pensar dos estatísticos, sugerindo que se faça uma visualização dos dados, interpretando o formato, centro, dispersão, presença de valores atípicos, summarizar numericamente e por fim escolher um modelo matemático. Foi o criador do boxplot e introduziu a palavra “bit” como uma contração do termo *binary digit*.

Douglas G. Altman (12 /07/1948 – 03/06/2018) foi um estatístico inglês (Figura 8), conhecido por seu trabalho em melhorar a confiabilidade dos artigos de pesquisa médica (13) e por artigos altamente citados sobre metodologia estatística. Ele foi professor de estatística em medicina na Universidade de Oxford. Há praticamente 30 anos, Altman (14) escreveu um artigo sobre problema da qualidade da pesquisa em medicina que causou um grande impacto e permanece válido até hoje. Nesta publicação ele afirma:

A má qualidade de muitas pesquisas médicas é amplamente reconhecida, mas, de forma perturbadora, os líderes da profissão médica parecem apenas minimamente preocupados com o problema e não fazem nenhum esforço aparente para encontrar uma solução.

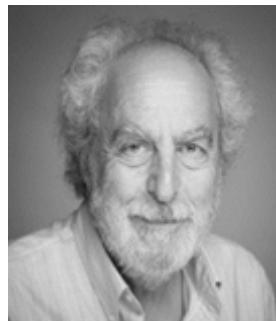


Figura 8: Douglas G. Altman

1.3 História resumida do R

O R é uma linguagem e um ambiente de desenvolvimento voltado fundamentalmente para a computação estatística. Foi inspirado em duas linguagens: S (John Chambers, do Bell Labs) que forneceu a sintaxe e

Scheme (Hal Abelson e Gerald Sussman) implementou e forneceu a semântica.

O nome R provém em parte das iniciais dos criadores, *George Ross Ihaka* e *Robert Gentleman* (Figura 9), e também de um jogo figurado com a linguagem S. Em 29 de Fevereiro de 2000, o software foi considerado com funcionalidades e estável o suficiente para a versão 1.0.

O R é um projeto GNU ¹. Software Livre significa que os usuários têm liberdade para executar, copiar, distribuir, estudar, alterar e melhorar o software. Foi desenvolvido em um esforço colaborativo de pessoas em vários locais do mundo (15).

O projeto R fornece uma grande variedade de técnicas estatísticas e gráficas. É uma linguagem e um ambiente similar ao S. A linguagem do S que também é uma linguagem de computador voltada para cálculos estatísticos. Um dos pontos fortes de R é a facilidade com que produções gráficas de qualidade podem ser produzidas. O R é também altamente expansível com o uso dos pacotes, que são bibliotecas para sub-rotinas específicas ou áreas de estudo específicas. Um conjunto de pacotes é incluído com a instalação de R e muito outros estão disponíveis na rede de distribuição do R - *Comprehensive R Archive Network* (CRAN) (16).



Figura 9: Robert Gentleman (E) e George Ross (D)

A linguagem R é largamente usada entre estatísticos e analistas de dados para desenvolver softwares de estatística e análise de dados. Pesquisas e levantamentos com profissionais da área mostram que a popularidade do R aumentou substancialmente nos últimos anos (17).

¹ Esta sigla está associada ao animal gnu africano, símbolo de software de distribuição livre, quer dizer is Not Unix, sigla recursiva muito comum entre nerds!

2 Natureza dos Dados

2.1 Variáveis e Dados

As pesquisas manuseiam dados referentes às variáveis que estão sendo estudadas. *Variável* é toda característica ou condição de interesse que pode de ser mensurada ou observada em cada elemento de uma amostra ou população. Como o próprio nome diz, seus valores são possíveis variar de um indivíduo a outro ou no mesmo indivíduo. Em contraste com a variável, o valor de uma constante é fixo. As variáveis podem ter valores numéricos ou não numéricos. O resultado da mensuração ou observação de uma variável é denominado *dado*.

A Tabela 1 mostra um conjunto de variáveis e suas medidas (dados) de um grupo de pacientes internados em uma determinada UTI. O termo medida deve ser entendido num sentido amplo, pois não é possível “medir” o sexo (observação) ou o estado geral (critérios) de alguém, ao contrário do peso e da pressão arterial que podem ser mensurados com instrumentos.

Tabela 1: Variáveis e dados.

Id	Nome	Idade	Sexo	PAS	PAD	Estado Geral
1	João	45	masculino	140	90	bom
2	Maria	32	feminino	110	70	regular
3	Pedro	27	masculino	120	80	grave
4	Teresa	18	feminino	100	60	bom

2.2 População e Amostra

Na pesquisa em saúde, a não ser quando se realiza um censo, coleta-se dados de um subconjunto de indivíduos denominado de amostra, pertencente a um grupo maior, conhecido como população. A população de interesse é, geralmente, chamada de população-alvo. A amostra para ser representativa da população deve ter as mesmas características desta. A partir dos dados encontrados na amostra, presume-se o resultado é condizente com a população. Este processo é denominado de inferência estatística. O interesse na amostra não está propriamente nela, mas na informação que ela fornece ao investigador sobre a população de onde ela provém. A amostra fornece estimativas (estatísticas) da população (Figura 10).

População ou **população-alvo** consiste em todos os elementos (indivíduos, itens, objetos) cujas características estão sendo estudadas.

Amostra é a parte, subconjunto, da população selecionada para estudo.

Em decorrência do acaso, diferentes amostras de uma mesma população fornecem resultados diferentes. Este fato deve ser levado em consideração ao usar uma amostra para fazer inferência sobre uma população. Este fenômeno é denominado de **variação amostral** ou **erro amostral** e é a essência da estatística. O grau de certeza na inferência estatística depende da representatividade da amostra.

O processo de obtenção da amostra é chamado de **amostragem**. Mesmo que este processo seja adequado, a amostra nunca será uma cópia perfeita da população de onde ela foi extraída. Desta forma, em qualquer conclusão baseada em dados de uma amostra, sempre haverá o que é conhecido como erro amostral. Este erro deve ser tratado estatisticamente tendo em mente a teoria da amostragem, baseada em probabilidades.

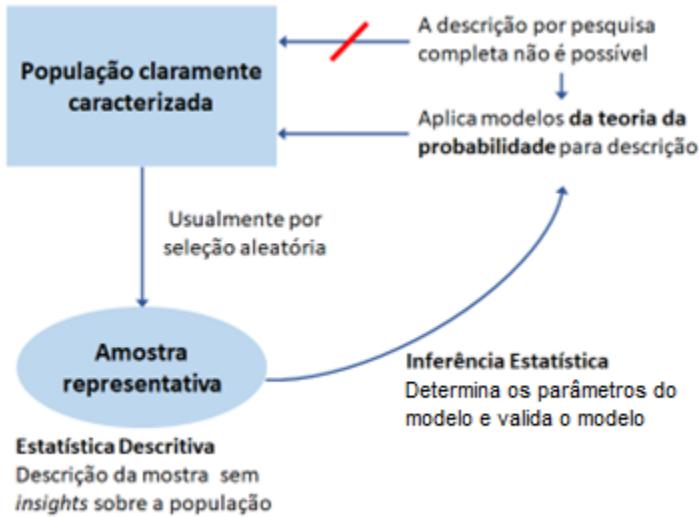


Figura 10: População, amostra e inferência estatística

2.3 Estimativas e Parâmetros

Estimativa é uma característica que resume os dados de uma amostra (estatística amostral) e o **parâmetro** é uma característica estabelecida para toda a população. Os valores dos parâmetros são normalmente desconhecidos, porque é inviável medir uma população inteira. A estimativa é um valor aproximado do parâmetro. As estimativas são representadas por letras romanas e os parâmetros por letras gregas. Por exemplo, a media da população é representada por μ e a média da amostra por \bar{x} ; o desvio padrão da população é denotado σ e o desvio padrão da amostra por s .

Na maioria dos estudos, são utilizadas amostras que fornecem estimativas que, para serem representativas da população, devem ser probabilísticas. Ou seja, a amostra deve ser recrutada de forma aleatória, permitindo que cada um dos membros da população tenha a mesma probabilidade de ser incluído na amostra. Além disso, uma amostra deve ter um tamanho adequado para permitir inferências válidas.

2.4 Escalas de medição

Em um estudo científico, há necessidade de registrar os dados para que eles representem acuradamente as variáveis observadas. Este registro de valores necessita de escalas de medição. **Mensuração** ou **medição** é o processo de atribuir números ou rótulos a objetos, pessoas, estados ou eventos de acordo com regras específicas para representar quantidades ou qualidades dos dados. Para a mensuração das variáveis são usadas as escalas nominal, ordinal, intervalar e de razão (18).

2.4.1 Escala Nominal

As escalas nominais são meramente classificativas, permitindo descrever as variáveis ou designar os sujeitos, sem recurso à quantificação. É o nível mais elementar de representação. São usados nomes, números ou outros símbolos para designar a variável. Os números, quando usados, representam códigos e como tal não permitem operações matemáticas. As variáveis nominais não podem ser ordenadas. Podem apenas ser comparadas utilizando as relações de igualdade ou de diferença, através de **contagens**. Os números atribuídos às variáveis servem como identificação, ou para associá-la a uma dada categoria. As categorias de uma escala nominal são exaustivas e mutuamente exclusivas. Quando existem duas categorias, a variável é dita **dicotômica** e com três ou mais categorias, **polítômicas**.

Os nomes e símbolos que designam as categorias podem ser intercambiáveis sem alterar a informação essencial.

Exemplos: Tipos sanguíneos: A, B, AB, O; variáveis dicotômicas: morto/vivo, homem/mulher, sim/não; cor dos olhos, etc.

2.4.2 Escala Ordinal

As variáveis são medidas em uma escala ordinal quando ocorre uma ordem, crescente ou decrescente, inerente entre as categorias, estabelecida sob determinado critério. A diferença entre as categorias não é necessariamente igual e nem sempre mensuráveis. Geralmente, designam-se os valores de uma escala ordinal em termos de numerais ou postos (*ranks*), sendo estes apenas modos diferentes de expressar o mesmo tipo de dados. Também não faz sentido realizar operações matemática com variáveis ordinais. Pode-se continuar a usar contagem.

Exemplos: classe social (baixa, média, alta); estado geral do paciente: bom, regular, mau; estágios do câncer: 0, 1, 2, 3 e 4; escore de Apgar: 0, 1, 2... 10.

2.4.3 Escala Intervalar

Uma escala intervalar contém todas as características das escalas ordinais com a diferença de que se conhece as distâncias entre quaisquer números. Em outras palavras, existe um espectro ordenado com intervalos quantificáveis. Este tipo de escala permite que se verifique a ordem e a diferença entre as variáveis, porém não tem um zero verdadeiro, o zero é arbitrário.

O exemplo clássico é a mensuração da temperatura, usando as escalas de: Celsius ou Fahrenheit. Aqui é legítimo ordenar, fazer soma ou médias. No entanto, 0°C não significa ausência de temperatura, portanto a operação divisão não é possível. Uma temperatura de 40°C não é o dobro de 20°C . Se 40°C e 20°C forem transformados para a escala Fahrenheit, passarão, respectivamente, para 104°F e 68°F e, sem dúvida, 104 não é o dobro de 68 !

2.4.4 Escala de Razão

Há um espectro ordenado com intervalos quantificáveis como na escala intervalar. Entretanto, as medidas iniciam a partir de um zero verdadeiro e a escala tem intervalos iguais, permitindo as comparações de magnitude entre os valores. Refletem a quantidade real de uma variável, permitindo qualquer operação matemática.

Os dados tanto na escala intervalar como na de razão, podem ser contínuos ou discretos. Dados contínuos necessitam de instrumentos para a sua mensuração e assumem qualquer valor em um certo intervalo. Por exemplo, o tempo para terminar qualquer tarefa pode assumir qualquer valor, 10 min, 20 min, 35 min, etc., de acordo com o tipo de tarefa. Outros exemplos: peso, dosagem de colesterol, glicemia.

Dados discretos possuem valores iguais a números inteiros, não existindo valores intermediários. A mensuração é feita através da contagem. Por exemplo: número de filhos, número de fraturas, número de pessoas.

2.5 Tipos de Variáveis

A primeira etapa na descrição e análise dos dados é classificar as variáveis, pois a apresentação dos dados e os métodos estatísticos variam de acordo com os seus tipos. As variáveis, primariamente, podem ser divididas em dois tipos: numéricas ou quantitativas e categóricas ou qualitativas (19).

2.5.1 Variáveis Numéricas

As variáveis numéricas são classificadas em dois tipos de acordo com a escala de mensuração: continuas e discretas.

As **variáveis contínuas** são aquelas cujos dados foram mensurados em uma escala intervalar ou de razão, podendo assumir, como visto, qualquer valor dentro de um intervalo de números reais, dependendo da precisão do instrumento de medição. O tratamento estatístico tanto para variável intervalar como de a razão é o mesmo. A diferença entre elas está na presença do zero absoluto. As variáveis numéricas contínuas têm unidade de medida. Por exemplo, um menino de 4 anos tem 104 cm.

Uma variável numérica é considerada **discreta** quando é apenas possível quantificar os resultados possíveis através do processo de contagem. Também têm unidade de medida – *número de elementos*. Por exemplo, o número de fraturas, o número de acidentes.

2.5.2 Variáveis Categóricas

As variáveis categóricas ou qualitativas são de dois tipos: nominal e ordinal, de acordo com a escala de mensuração. Um tipo particularmente comum é uma variável binária (ou variável dicotômica), que tem apenas dois valores possíveis. Por exemplo, o sexo é masculino ou feminino. Este tipo de variável é bastante utilizado na área da saúde, em Epidemiologia. As variáveis nominais não têm quaisquer unidades de medida e a nominação das categorias é completamente arbitrária e pertencer a uma categoria não significa ter maior importância do que pertencer à outra. Uma variável ordinal tem uma ordem inerente ou hierarquia entre as categorias. Do mesmo modo que as variáveis nominais, as variáveis ordinais não têm unidades de medida. Entretanto, a ordenação das categorias não é arbitrária. Assim, é possível ordená-las de modo lógico. Um exemplo comum de uma variável categórica ordinal é a classe social, que tem um ordenamento natural da maioria dos mais desfavorecidos para os mais ricos. As escalas, como a escore de Apgar e a escala de coma de Glasgow (20), também são variáveis ordinais. Mesmo que pareçam numéricas, elas apenas mostram uma ordem no estado dos pacientes. O escore de Apgar (21) é uma escala, desenvolvida para a avaliação clínica do recém-nascido imediatamente após o nascimento. Originalmente, a escala foi usada para avaliar a adaptação imediata do recém-nascido à vida extrauterina. A pontuação pode variar de zero a 10. Uma pontuação igual ou maior do que oito, indica um recém-nascido normal. Uma pontuação de sete ou menos pode significar depressão do sistema nervoso e abaixo de quatro, depressão grave.

As variáveis ordinais, da mesma forma que as nominais, não são números reais e não convém aplicar as regras da aritmética básica para estes tipos de dados. Este fato gera uma limitação na análise dos dados.

2.5.3 Como identificar o tipo da variável?

A maneira mais fácil de dizer se os dados são numéricos é verificar se eles têm unidades ligadas a eles, tais como: g, mm, °C, ml, número de úlceras de pressão, número de mortes e assim por diante. Se não, podem ser ordinais ou nominais – ordinais se os valores podem ser colocados em ordem. A Figura 11 é uma ajuda para o reconhecimento do tipo de variável (22).



Figura 11: Caminho para identificar o tipo de variável

2.5.4 Variáveis Dependentes e Independentes

De um modo geral as pesquisas são realizadas para testar as hipóteses dos pesquisadores e, para isso, eles medem variáveis com a finalidade de compará-las. A maioria das hipóteses podem ser expressas por duas variáveis: uma variável explicativa ou preditora e uma variável desfecho (19).

A **variável preditora** ou explanatória é a que se acredita ser a causa e também é conhecida como variável independente, porque o seu valor não depende de outras variáveis. Em Epidemiologia, é com frequência referida como exposição ou fator de risco.

A **variável desfecho** é aquela que é o efeito, consequência ou resultado da ação de outra variável, por isso, também chamada de variável dependente. Em um estudo que tenta verificar se o tabagismo, durante a gestação, pode interferir no peso do recém-nascido, tem o fumo (variável categórica) como variável preditora (exposição ou fator de risco) e o peso do recém-nascido (variável numérica contínua) como variável desfecho

3 Produção dos Dados

3.1 Processo de Pesquisa

A pesquisa é um processo de construção do conhecimento. O objetivo deste processo é gerar um novo conhecimento e/ou confirmar ou refutar algum conhecimento prévio. A pesquisa é um processo de aprendizagem tanto do pesquisador quanto da sociedade que se beneficiará deste novo conhecimento. Para ser chamada de científica, a pesquisa deve obedecer aos princípios consagrados pela ciência (23).

A pesquisa nasce de uma dúvida do pesquisador, de algum questionamento que ele considerou interessante sobre o mundo, ou seja, de algo que se costuma chamar de pergunta ou questão da pesquisa. Existem vários motivos que geram questões de pesquisa:

- Avaliação crítica de pesquisas realizadas por outros pesquisadores.
- Condução de uma pesquisa primária com a finalidade de responder uma questão (ou questões), gerando um novo conhecimento ou ampliação do conhecimento existente.
- Para obter habilidades de pesquisa ou experiência, com frequência como parte de um programa educacional.
- Testar a viabilidade de um projeto ou técnica de pesquisa.

3.1.1 Questão de Pesquisa

A pesquisa visa estabelecer novos conhecimentos em torno de um tema específico. O tema de pesquisa pode surgir do próprio interesse ou experiência do pesquisador, ou partir da encomenda de alguma instituição financiadora. Algumas vezes, a pesquisa se origina de outros estudos realizados pelo próprio pesquisador ou outros pesquisadores.

À medida que a ideia da pesquisa cresce, o pesquisador estabelece uma pergunta de pesquisa específica ou um conjunto de questões que ele deseja responder. Algumas vezes, o tema da pesquisa é tão amplo que o pesquisador tem que ter cuidado para não se perder do seu objetivo. Este objetivo é que vai guiá-lo no estabelecimento da pergunta ou perguntas a serem respondidas no estudo. Estes questionamentos são conhecidos como **questão de pesquisa** ou **pergunta de partida**.

O foco da questão de pesquisa pode ser na descrição de um fenômeno clínico. Neste caso a pergunta é dita **descritiva**, por exemplo, pesquisa de prevalência de uma enfermidade, proporção de utilização de um serviço de saúde, características de um teste, etc. Quando a pergunta busca a explicação para um fenômeno, ela é dita **analítica**, por exemplo, comparação entre dois fenômenos. Em geral, perguntas analíticas são mais interessantes. Entretanto, as perguntas descritivas são fundamentais no início de um estudo analítico.

Uma boa pergunta de pesquisa deve ter as seguintes características (24):

- **Factível:** o pesquisador deve conhecer desde o início os limites e problemas práticos que podem interferir na pesquisa. A viabilidade está relacionada com o tamanho amostral, com o domínio técnico adequado, com o tempo e custos envolvidos e com um foco dirigido estritamente aos objetivos mais importantes.
- **Interessante:** a questão de pesquisa deve despertar o interesse não apenas do pesquisador, mas também de seus pares e agentes financiadores.
- **Nova:** a pesquisa deve ser inovadora, original, em algum sentido, para que o estudo seja uma contribuição ao conhecimento ou amplie um conhecimento existente;
- **Ética:** se o estudo impõe riscos físicos ou invasão de privacidade ou não traz nenhuma informação nova, o pesquisador deve suspendê-lo. É importante discutir previamente com pesquisadores mais experientes ou com algum representante do Comitê de Ética em Pesquisa da instituição.

- **Relevante:** nenhuma das características da questão de pesquisa é mais importante do que a sua relevância. Para isto basta pensar nos benefícios que os resultados da pesquisa trarão à Medicina atual.

Ou seja, antes de dedicar tempo e esforço para escrever um projeto de pesquisa deve-se avaliar se a questão de pesquisa é FINER (Factível, Interessante, Nova, Ética e Relevante).

3.1.2 Hipótese de Pesquisa

Uma vez estabelecida a(s) pergunta(s) de pesquisa adequada(s), os pesquisadores formulam hipóteses para serem testadas. Enquanto a pergunta de pesquisa possa ser um pouco vaga em sua natureza como: “existe uma relação entre o tipo psicológico e a capacidade de parar de usar drogas?” Uma hipótese de pesquisa, necessita ser precisa. Há necessidade de especificar qual o tipo psicológico está relacionado à habilidade de parar de usar drogas.

A precisão da hipótese é fundamental em um projeto de pesquisa, pois ela determinará o delineamento de pesquisa a ser seguido pelo pesquisador e as técnicas estatísticas apropriadas para a análise dos dados. A fonte e o tipo de dados são determinados pela característica do delineamento recomendado pela hipótese de pesquisa.

O objetivo da pesquisa, usando o método científico, é refutar ou não as hipóteses de pesquisa. Se a hipótese do pesquisador não for rejeitada, houve a geração de um novo conhecimento.

3.2 Processo de Amostragem

Após o estabelecimento das hipóteses a serem testadas, há necessidade de coletar os dados. Uma vez que é praticamente impossível analisar toda a população que constitui a **população-alvo**, extraí-se uma **amostra** desta população. Este processo é denominado de **amostragem** (25).

Uma amostra deve ser representativa da população, ou seja, deve ter características semelhantes às da população e ser fidedigna. A fidedignidade está relacionada à precisão dos dados que sofrem influência dos instrumentos de aferição, questionários não validados e falhas humanas. Uma amostra inadequada ameaça a validade da pesquisa. Os dados coletados de maneira não aleatória são chamados de **evidência anedótica**. O nível de confiança nos resultados de uma pesquisa está diretamente relacionado à qualidade da amostra. A amostra deve ser representativa.

Uma amostra deve conter apenas dados úteis que permitam a resposta da pergunta de pesquisa, evitando desperdício e fuga dos objetivos traçados. A aleatoriedade provoca uma diferença entre o resultado da amostra e o verdadeiro valor da população que é denominada **erro amostral**. Não importa quão bem a amostra seja coletada, os erros amostrais irão sempre ocorrer. Entretanto, não existe técnica estatística que salve amostras coletadas incorretamente, tendenciosas!

$$\text{MÉDIA NA POPULAÇÃO} - \text{MÉDIA NA AMOSTRA} = \text{ERRO AMOSTRAL}$$

3.2.1 Amostras probabilística

Para evitar vieses, erros sistemáticos, que favorecem determinados desfechos, o ideal é coletar uma amostra probabilística. A amostra probabilística adota o princípio da **equiprobabilidade**, isto é, “todos os sujeitos da população têm a mesma probabilidade de fazerem parte da amostra”. Esta probabilidade é conhecida e diferente de zero. As amostras probabilísticas têm o potencial de ser possível a generalização para a população; ser imparcial e com menor erro amostral.

Amostra aleatória simples: é a mais utilizada pois garante representatividade da amostra junto à população. A amostra aleatória simples não emprega nenhum critério particular para a definição da amostra.

O mecanismo mais comum de obter este tipo de amostra é por um simples sorteio, em geral, usando programas de computador.

Amostra aleatória estratificada: quando a população é constituída por subpopulações ou estratos e é razoável supor que a variável de interesse apresenta comportamento diferente nos diferentes estratos, pode-se usar este tipo de amostragem. Neste caso, a amostra deve ter a mesma estratificação da população para ser representativa. Um exemplo comum de estratificação é o nível socioeconômico. A partir do momento que os estratos estão definidos se procede uma amostra aleatória simples de cada estrato.

Amostra aleatória sistemática: as unidades amostrais são selecionadas a partir de um esquema rígido preestabelecido de sistematização que tem o propósito de abranger toda a população-alvo. Para isso, ordenase os indivíduos da população (por exemplo, um grande arquivo com 20000 fichas) e calcula-se uma constante conveniente, $c = N/n$, onde N é tamanho da população e n é o tamanho da amostra. Se $n = 500$, a constante será 40, ou seja, será selecionado aleatoriamente o primeiro membro da amostra (k), de maneira que k seja menor do que a constante e maior do que 1. A partir daí os sucessivos membros serão: $k + c; k + 2c; k + 3c; \dots$ até atingir n .

Amostra aleatória por conglomerados (*clusters*): este tipo de amostra é utilizada quando dentro da população são identificados agrupamentos (*clusters*) naturais, por exemplo, espaços, vilas, etc. Neste tipo de amostragem o elemento focal não é o sujeito, mas o *cluster*. Identificados estes, sorteiam-se os conglomerados e se analisa todos os indivíduos dos conglomerados sorteados.

3.2.2 Amostras não probabilísticas

Na amostragem não aleatória ou intencionada há uma escolha deliberada da amostra, subordinada a objetivos específicos do pesquisador. Não há garantia de representatividade da população. É importante averiguar, neste tipo de amostragem, a presença de *conflictos de interesse*.

Amostra de conveniência: é uma técnica comum onde é selecionada uma mostra que esteja acessível. Em outras palavras, os indivíduos são recrutados porque eles estão prontamente disponíveis. Neste tipo de amostra há incapacidade de fazer afirmações gerais com rigor estatístico sobre a população.

Amostra por cotas: é uma versão não probabilística da amostra estratificada. Tem três etapas:

- 1) Segmentação, onde se divide em grupos, por exemplo, sexo, classe social, região, etc.;
- 2) Definição do tamanho das cotas;
- 3) Seleção por meio de amostras de conveniência.

Amostra de resposta voluntária: o pesquisador solicita aos membros de uma população-alvo para que eles participem da amostra e as pessoas decidem se entram ou não. Esses tipos de amostras são enviesados porque as pessoas podem ter interesses particulares ou opiniões negativas e tendem a querer participar.

3.2.3 Tamanho amostral

A determinação do tamanho de uma amostra é de suma importância, pois amostras desnecessariamente grandes acarretam desperdício de tempo e de dinheiro e amostras muito pequenas podem levar a resultados não confiáveis, ameaçando a validade da pesquisa.

Não existe um número estabelecido para o tamanho da amostra. Há uma solução para cada caso. O tamanho da amostra depende (26):

- do tipo de problema;
- do tipo de variável;

- da magnitude do erro estatístico aceito pelo pesquisador;
- da diferença minimamente importante entre os grupos;
- da probabilidade de que a amostra identifique uma diferença verdadeira: Poder estatístico;
- do tempo, dinheiro e pessoal disponível, bem como da dificuldade em se obterem dados e da complexidade da pesquisa.

O tamanho amostral mínimo é determinado por fórmulas estatísticas complexas. Os cálculos são muito pesados, mas agora, felizmente, existem programas de computador disponíveis que realizam este trabalho, por exemplo o **G-Power3** (27). Além disso, é possível acessar um site que fornece informações e ferramentas para o cálculo amostral em pesquisas da área da saúde². Existem tabelas extensas para calcular o número de participantes (28) para um determinado nível de poder (e vice-versa).

3.3 Principais Delineamentos de Pesquisa

Em geral, a pesquisa clínica, é dividida em dois tipos de investigação. O primeiro é aquele em que o observador apenas observa o doente, as características da sua doença e sua evolução, sem atuar de modo a modificar qualquer aspecto que esteja estudando. Trata-se de **estudo observacional**.

O segundo corresponde aos **estudos experimentais**, onde o pesquisador não se limita a observar, mas promove uma intervenção com o objetivo de conhecer os efeitos dessa sobre os participantes da pesquisa. A intervenção pode ser a prescrição de um medicamento, uma dieta, atividade física ou repouso, ou simplesmente, o estabelecimento de um programa de atenção à saúde.

Os estudos podem ser também classificados em primários ou secundários ou integrativos (29). Estudos primários correspondem a pesquisas originais que constituem a maioria das publicações encontradas nas revistas médicas. Estudos secundários são aqueles que procuram summarizar e extrair conclusões de estudos primários

- *Estudos Primários*
 - Estudos Observacionais
 - * Relato de Caso e Série de Casos
 - * Estudo Transversal
 - * Estudo Caso-controle
 - * Estudo de Coorte
 - Estudos Experimentais
 - * Experimento laboratorial
 - * Ensaio Clínico
- *Estudos Secundários*
 - Revisões não sistemáticas
 - Revisões Sistemáticas
 - Diretrizes (*Guidelines*)
 - Análise de decisão
 - Análise Econômica

3.3.1 Elementos básicos de um delineamento de pesquisa

Os estudos contêm três elementos básicos:

1. Variáveis componentes: Nas investigações das relações entre as variáveis identificam-se pelo menos duas variáveis nos estudos epidemiológicos.

²<http://calculoamostral.bauru.usp.br/calculoamostral/index.php>

- a. *Desfecho*: Aquilo que vai acontecer durante uma investigação na mensuração da condição de saúde-doença. Sinônimo: variável dependente.
 - b. *Exposição*: O fator que precede o desfecho. Sinônimos: fator em estudo, variável preditora, variável independente.
2. Temporalidade: Quanto ao tempo os estudos podem ser contemporâneos, retrospectivos e prospectivos, de acordo como os dados são obtidos em relação ao momento atual.
 3. Enfoque: Um estudo pode ter vários enfoques. Na maioria deles, na área médica, eles relacionam-se à prevenção, ao diagnóstico, à terapêutica e ao prognóstico.

3.4 Estudos Observacionais

3.4.1 Relato de Caso ou Série de casos

No relato de caso, descrevem-se casos raros, eventos não comuns ou inesperados, doenças desconhecidas ou raras. Um evento notável deve ser identificado. Um relato de caso tem a descrição de até dez casos. Acima deste número tem-se uma série de casos (30).

Metodologicamente, faz-se um relato descritivo simples de características interessantes observadas em um paciente ou grupo de pacientes. Os indivíduos são acompanhados em um espaço de tempo curto e não possuem participantes-controles. A coleta dos dados é, na maioria das vezes, retrospectiva.

Uma série de casos não é planejada e não envolve quaisquer hipóteses investigativas. Pode ser empregada como precursor de outros estudos.

3.4.2 Estudos Transversais ou Seccionais

Os estudos transversais são também conhecidos como estudos seccionais. Este tipo de estudo fornece a informação sobre a prevalência, ou seja, a proporção dos indivíduos que tem a doença ou condição clínica em um determinado momento. Por este motivo são também conhecidos como estudos de prevalência (31).

Observam dados coletados em um grupo de indivíduos em um único momento, sem um período de seguimento. O desfecho e exposição são avaliados no mesmo momento no tempo. Os dados são coletados apenas uma vez para cada indivíduo, podendo ser em dias diferentes em diferentes sujeitos. As informações são, em geral, obtidas em um curto espaço de tempo.

É um estudo estático, representa a “fotografia” de um momento. Entretanto, se as variáveis preditora e de desfecho são definidas apenas com base nas hipóteses causa-efeito do investigador e não no delineamento do estudo, é possível também examinar associações.

Os estudos de corte transversal, de um modo geral, são desenhados para determinar “*O que está acontecendo?*”. São usados para:

- Determinar a prevalência de uma doença, como a prevalência de HIV em gestantes.
- Pesquisar atitudes ou opiniões em relação a um determinado assunto (pesquisa de satisfação)
- Verificar interrelações entre variáveis, como observação das características de fumantes pesados em relação ao sexo, idade, etc.
- Enquetes

Cuidados na interpretação de dados de estudos transversais

- 1) *Efeito temporal*

Como os dados (exposição e desfecho) são coletados no mesmo momento, fica difícil estabelecer qualquer relação temporal entre eles (dilema ovo/galinha). Por exemplo, não é possível estabelecer uma relação de causalidade entre hipertensão e doença cardíaca se os dados são coletados de forma a ficar impossível saber que surgiu em primeiro lugar.

2) Estudos transversais repetidos

Os estudos transversais, algumas vezes, são repetidos em outro momento ou em outros locais com a finalidade de verificar variabilidade nos achados. Por exemplo, medir a prevalência de uma doença em momentos diferentes ou em diferentes locais. Os indivíduos serão um pouco diferentes, devendo-se interpretar as diferenças destes resultados com cautela.

3) Estudos transversais que parecem longitudinais

Uma armadilha comum é confundir um estudo seccional com um longitudinal porque os dados foram coletados através do tempo até completar o tamanho amostral previsto. O importante é que os dados (variável preditora e desfecho) foram coletados somente uma vez para cada indivíduo e no mesmo momento. Isto gera uma interpretação errônea se analisarmos como um estudo longitudinal.

Análise dos Estudos Transversais

Quando se compara a prevalência de doença em expostos e não expostos, a medida de associação usada é a *Razão de Prevalência Pontual* (RPP).

3.4.3 Estudos Caso-Controle

Para examinar a possível associação de uma exposição a uma determinada doença, identifica-se um grupo de doentes (casos) e, com a finalidade de comparação, um grupo de pessoas sem a doença (controles) e determina-se a chance (*odds*) de exposição e não exposição entre casos e entre controles.

Os estudos caso-controle, portanto, partem da presença ou ausência de um desfecho e após olham para trás no tempo (retrospectivamente) para detectar possíveis fatores de risco (Figura 12)(32). Analisam o que aconteceu e são usados para investigar fatores de risco de doenças raras onde um estudo prospectivo seria muito longo para identificar uma quantidade suficiente de casos.

É útil também para investigar surtos agudos (infecção alimentar) para identificar se existe ou não associação entre a exposição e o desfecho investigado. Com frequência, os estudos caso-controle são o primeiro passo na busca de uma etiologia quando há suspeita de que alguma de várias exposições esteja associada a uma determinada doença.

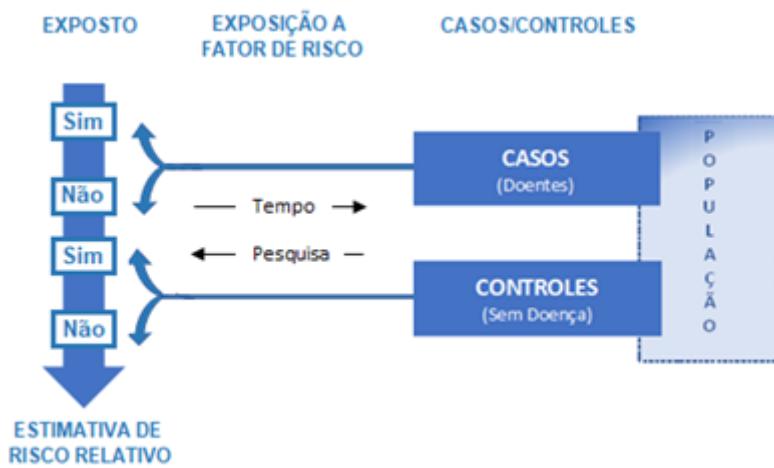


Figura 12: Desenho de um estudo caso controle.

Seleção dos casos

Os casos podem ser selecionados de várias fontes, incluindo indivíduos hospitalizados, de consultórios ou clínicas, principalmente quando registros adequados são mantidos.

Muitos problemas podem ocorrer na seleção de casos, neste tipo de estudo. Se os casos forem selecionados de um único hospital, quaisquer fatores de risco identificados podem ser apenas daquele hospital, em decorrência do padrão de referência e nível de atendimento (um hospital terciário que apenas atende um determinado convênio, por exemplo, o Sistema Único de Saúde). Por isso, devem ser utilizados casos procedentes de vários hospitais da comunidade, pois aí os casos pertenceriam a diferentes grupos sociais e diferentes graus de gravidade da doença.

Casos incidentes ou prevalentes

Os casos usados nos estudos caso-controle podem ser casos incidentes (recém-diagnosticados) ou casos prevalentes da doença (pessoas que apresentaram a doença em algum período).

O problema do uso de casos incidentes é que há necessidade de se esperar que novos casos sejam diagnosticados e isto pode requerer muito tempo. Enquanto os casos prevalentes já estão disponíveis havendo um maior número disponível para o estudo. Em ambos os modelos existem problemas, pois nos casos prevalentes algumas pessoas podem morrer logo após o diagnóstico e estarem pouco representadas no estudo. Por outro lado, nos casos incidentes, serão excluídos os pacientes que morreram antes do diagnóstico ser feito. Não existe uma solução fácil para este problema, mas é importante lembrar-se destas questões ao interpretar os resultados e tirar conclusões do estudo.

Seleção dos controles

Da mesma forma do que nos estudos experimentais, a escolha dos controles afeta a comparação com os casos (33). A escolha dos controles inclui:

- Pacientes do mesmo hospital, mas com condições ou doenças não relacionadas;
- Pacientes pareados um a um em relação a fatores prognósticos, tais como sexo e idade;
- Uma amostra aleatória originária da mesma população de onde provêm os casos.

Sem dúvida, o melhor grupo controle é a terceira opção, mas esta é raramente possível. Por este motivo, alguns estudos caso-controle incluem mais de um grupo controle para tornar o estudo mais robusto

Controles pareados

O emparelhamento é definido como processo de seleção dos controles para que sejam semelhantes aos casos em algumas características como, por exemplo, idade, gênero, raça, condição socioeconômica e ocupação.

Controles emparelhados são bastante comuns. O autor deve ter o cuidado de especificar cuidadosamente o modo como houve o pareamento. Por exemplo, "emparelhado por idade dentro de dois anos" mostra a amplitude do pareamento. É difícil realizar o emparelhamento para muitos fatores, pois um pareamento seguro não existe. Em um delineamento pareado, a análise estatística deve levar em conta o emparelhamento e os fatores usados por ele. Onde um indivíduo em um par tiver um dado perdido, ambos devem ser omitidos da análise estatística.

Estudos caso-controle aninhados

Um delineamento do tipo caso-controle aninhado é um estudo de caso-controle "aninhado" em um estudo de coorte (34). É um excelente desenho para variáveis preditoras que são caras para medir e que podem ser avaliadas no final do estudo em indivíduos que desenvolvem o resultado durante o estudo (casos) e em uma amostra daqueles que não o fazem (controles).

O investigador começa com uma coorte adequada (Figura 13) (35) com casos suficientes ao final do acompanhamento para fornecer poder adequado para responder à pergunta de pesquisa. No final do estudo, aplica critérios que definem o resultado de interesse para identificar todos aqueles que desenvolveram o resultado (casos). Em seguida, seleciona uma amostra aleatória dos indivíduos que não desenvolveram o resultado (controles).

A principal razão para usar delineamentos caso-controle aninhado é reduzir o trabalho e o custo na coleta de dados. A principal desvantagem desse projeto é que muitas questões e circunstâncias da pesquisa não são passíveis de armazenamento para posterior análise.

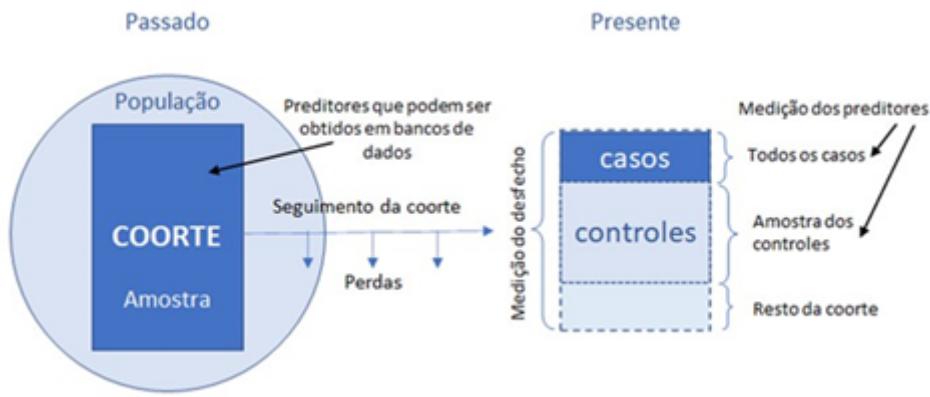


Figura 13: Desenho de um estudo caso-controle aninhado.

Estudo caso-controle de base populacional

São os estudos caso-controle onde os casos e controles são uma amostra completa ou probabilística de uma população definida.

Limitações dos estudos caso-controle

Várias limitações podem afetar os estudos caso-controle:

- A escolha do grupo controle afeta as comparações entre casos e controles;
- Os dados da exposição ao fator de risco são coletados retrospectivamente e dependem da memória dos participantes, registros médicos e, portanto, podem ser incompletos, sem acurácia ou enviesados (viés de memória);
- Se o processo que conduz à identificação dos casos está relacionado a um possível fator de risco, a interpretação dos resultados será difícil (viés averiguação).
 - Por exemplo: suponha que os casos sejam mulheres jovens com hipertensão selecionadas de uma clínica de contracepção. Nesta situação, um possível fator de risco, o anticoncepcional oral (ACO), estará vinculado à seleção dos casos e, desta forma, o uso de ACO será mais comum entre os casos do que entre os controles populacionais.

Análise dos Estudos Caso-controle

A principal estratégia de análise é o cálculo da *odds ratio* (Razão de Chances), que pode ser interpretado como uma estimativa do Risco Relativo.

O Risco Relativo somente pode ser calculado quando é possível o cálculo da incidência (ver capítulo 18). Nos estudos caso-controle, isso não é possível, pois aqui o estudo começa com casos e controles em vez de indivíduos expostos e não expostos ao fator de risco. Desta maneira, se comparam as *odds* (chance) de uma exposição passada a um fator de risco suspeitado em indivíduos doentes e em controles não doentes. Esta relação é denominada de *odds ratio*.

3.4.4 Estudos de Coorte

Os estudos de coorte são considerados o padrão-ouro dos estudos observacionais. Seu nome se originou das coortes dos soldados romanos, cada uma delas constituída por 480 a 600 legionários. As coortes romanas

eram distintas entre si e tinham sua identidade determinada por, ao menos, uma característica comum entre os indivíduos de cada grupo. Podia ser por características estratégicas no campo de batalha, por uma cor presente na indumentária, ou outras. Em Epidemiologia, o termo coorte permaneceu com significado semelhante.

Em um estudo de coorte, um grupo de pacientes sadios (coorte), expostos ou não a um suspeitado fator de risco, é seguido através do tempo para determinar a incidência da doença em questão em cada um dos grupos (36).

Neste modelo de estudo, a característica comum aos dois grupos é a exposição. Tem-se uma coorte de expostos e uma coorte de não expostos que são acompanhadas por um período de tempo que permita o aparecimento do desfecho. No final do estudo, compara-se a incidência do desfecho (doença) entre os expostos com a incidência do desfecho entre os não expostos. Se existe uma associação positiva entre a exposição e o desfecho, se espera que a incidência do desfecho entre os expostos seja maior do que a incidência de desfecho entre não expostos.

Um esquema simplificado de um estudo de coorte é mostrado na Figura 14(37).

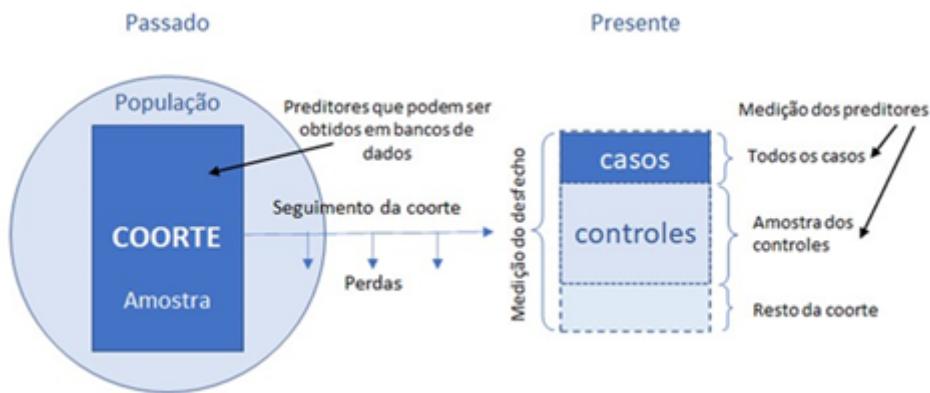


Figura 14: Desenho de um estudo de coorte sobre risco.

Observar que como se identifica novos casos (incidência) à medida que eles ocorrem, é possível determinar uma relação temporal entre a exposição e a doença, isto é, se a exposição precedeu o início da doença. Isto é *fundamental para estabelecer uma relação causal entre a exposição e a doença*.

Os estudos de coorte têm semelhança com os ensaios clínicos randomizados. Ambos os estudos comparam grupos expostos a grupos não expostos. Não havendo possibilidade de realizar a randomização, por exemplo, por motivos éticos quando a exposição é sabidamente prejudicial, é indicado um estudo de coorte. A diferença fundamental, portanto, é a ausência de randomização nos estudos de coorte.

Existem duas maneiras básicas para formar os grupos:

- 1) Seleciona-se a população-alvo baseado no fato dos indivíduos estarem expostos ou não ao fator em estudo (Figura 14);
- 2) Ou seleciona-se a população-alvo antes que qualquer um dos seus membros se torne exposto, ou antes, que a exposição seja identificada (Figura 15). Um exemplo típico deste modelo é o Estudo de Framingham (38).



Figura 15: Desenho de uma coorte com grupos expostos e não expostos. @david2019gordis.

Tipos de estudo de coorte

De acordo com as características do seguimento, as coortes podem ser:

- 1) **Estudo de Coorte Prospectivo** (Coorte Concorrente ou Longitudinal), onde os grupos são montados no presente, coletados os dados basais deles e continua-se a coletar dados com o passar do tempo até a doença se desenvolver ou não.
- 2) **Estudo de Coorte Retrospectivo ou Histórico** (Coorte não concorrente), onde a exposição é avaliada em dados passados e o desfecho (doença ou não) é verificado no momento do início do estudo. O problema aqui é que a averiguação da exposição depende dos registros pregressos.
- 3) **Estudo de Coorte Misto** (Prospectivo e Retrospectivo), onde a exposição é verificada em registros objetivos no passado (como em uma coorte histórica) e o seguimento e a medida do desfecho se fazem no futuro.

Vieses em estudos de coorte

Os potenciais vieses nos estudos de coorte são os seguintes:

- 1) **Viés de confusão** – é a grande ameaça dos estudos observacionais. O confundimento causa um erro sistemático na inferência, podendo aumentar ou diminuir uma associação observada entre exposição e doença. Uma variável funciona como fator de confusão quando ela está associada com a exposição e ao mesmo tempo com a doença. Ela não deve fazer parte da cadeia causal da exposição à doença. Por exemplo, num estudo sobre fatores de risco, uma associação entre o hábito de beber café e a doença coronária é detectada. Porém, se não for considerado o fato de que os fumantes bebem mais café do que os não-fumantes, pode-se chegar à errônea conclusão de que o café é um fator de risco independente para doença coronária, o que não corresponde à realidade. Neste caso, o café é um fator de confusão e não um fator causal independente para a doença coronária (39).
- 2) **Viés na avaliação dos desfechos** – este viés pode ocorrer quando o pesquisador que avalia o desfecho também sabe sobre o *status* de exposição dos sujeitos da pesquisa. Evita-se este problema “cegando” a pessoa que faz a avaliação da doença.
- 3) **Viés de informação** – ocorrem principalmente em estudos históricos onde as informações dependem de registros passados e podem ser diferentes entre as pessoas expostas e não expostas.
- 4) **Viés de não resposta e perdas de acompanhamento** – a não participação e as perdas podem introduzir um grande viés, alterando o cálculo da incidência nos expostos e entre os não expostos.

- 5) **Viés de análise** – se os estatísticos tiverem alguma hipótese em relação aos dados que estão analisando, eles podem introduzir vieses em suas análises.

Análise dos estudos de coorte

Para verificar se existe associação entre certo desfecho (doença) e uma determinada exposição calcula-se o **Risco Relativo (RR)**. Este é definido como a razão entre a incidência (risco) em expostos e a incidência (risco) em não expostos (ver capítulo 18).

Vantagens e desvantagens dos estudos de coorte

1) Vantagens

- Adequado para exposições raras
- Bom poder para testar hipóteses
- Importante em estudos etiológicos e prognósticos
- Salienta os múltiplos desfechos de uma exposição

2) Desvantagens

- Inadequado em desfechos raros
- Perdas no seguimento levam a viés de seleção
- Demorado/elevado custo

3.5 Ensaio Clínico

Experimentos são estudos nos quais o pesquisador *manipula a variável preditora* (intervenção) e observa o efeito no desfecho que está sendo avaliado ao longo do tempo. A abordagem experimental, especificamente, o ensaio clínico randomizado controlado é a ferramenta de escolha para comparar terapêuticas ou intervenções.

Os estudos experimentais podem também comparar os cuidados prestados por serviços de saúde, programas de educação em saúde e estratégias administrativas. Os estudos experimentais realizados com seres humanos são denominados de **ensaios clínicos**.

Nos ensaios clínicos não controlados os indivíduos servem como seus próprios controles (antes-e-depois). Os resultados destes estudos estão sujeitos vários problemas:

- **Melhora previsível.** Paciente melhora espontaneamente e não pelo tratamento.
- **Flutuação na gravidade da doença.**
- **Efeito Hawthorne:** o indivíduo melhora pela atenção e não pela terapêutica (40).
- **Regressão à média:** uma limitação importante surge quando se quer avaliar a evolução de um grupo que tenha sido selecionado por estar no extremo de uma distribuição sem que haja um grupo controle. Empiricamente, observa-se que indivíduos que se encontram num determinado momento, em um dos extremos de uma distribuição, tendem a estarem menos distantes da média em um momento posterior, sem que qualquer intervenção tenha sido desenvolvida. Este fenômeno é conhecido como efeito de *regressão à média*. Por exemplo: uma pessoa com uma doença crônica tem dias piores e outros melhores. Se ela é medicada com gotas homeopáticas ou faz uso de florais nos dias em que se sente excepcionalmente mal vai notar que é frequente uma melhora, seguindo estes “tratamentos”. Não que eles funcionem, mas pela regressão à média (41).

3.5.1 Características de um ensaio clínico

Um ensaio clínico deve ter algumas características fundamentais (Figura 16)(42):

- 1) Os indivíduos devem ser designados por randomização para os grupos de comparação.
 - A randomização é a melhor abordagem no delineamento de um ensaio clínico (43).
 - Randomizar significa sortear (por meio de computadores, tábua de números aleatórios) os indivíduos para decidir a alocação dos mesmos em um dos grupos de estudo. O elemento decisivo da randomização é a imprevisibilidade da próxima alocação.
- 2) O pesquisador compara o grupo de estudo com um grupo controle apropriado.
- 3) O investigador manipula a variável independente (preditora).



Figura 16: Estrutura de um ensaio clínico randomizado.

3.5.2 Elementos básicos de um ensaio clínico

Seleção dos participantes

Os pesquisadores devem determinar e explicar detalhadamente os critérios de inclusão e de exclusão:

- Objetivos dos critérios de inclusão e exclusão
 - Restringir a heterogeneidade da amostra
 - Diminuir o número de variáveis independentes
 - Fazer com que exista uma chance maior de que as diferenças nos desfechos estejam relacionadas aos tratamentos
 - Melhorar a *validade interna*, ou seja, o grau em que os resultados do estudo são consistentes para aquela amostra particular de indivíduos. Esta validade depende basicamente do rigor metodológico usado para delinejar o ensaio clínico, podendo ser ameaçada por dois tipos de erros: sistemático ou aleatório.
 - Tornar a generalização (validade externa) mais precisa. Entretanto deve-se ter cuidado com critérios de inclusão e exclusão muito rígidos, pois podem diminuir esta capacidade de generalização

O grau de detalhamento deve ser suficientemente preciso para permitir que outros reproduzam o estudo. O tamanho da amostra deve ser claramente determinado pelo poder do teste estatístico. Poder é a habilidade de o teste estatístico detectar diferenças entre os grupos, dado que tais diferenças existam na população em

estudo. Lembrar que resultados não significativos podem ser apenas uma evidência para um inadequado tamanho amostral.

O grupo controle deve ser selecionado utilizando-se os mesmos critérios do grupo experimental. Prestar atenção em possíveis armadilhas que podem gerar vieses:

- Uso de grupo controle histórico (não concorrente);
- Grupo controle selecionado de outros locais (outras clínicas, outros hospitais).

O grupo controle adequado é um grupo controle concorrente, tratado no mesmo momento e no mesmo local do grupo experimental. O característico é o grupo controle não receber tratamento. Mais comumente recebem um placebo, indistinguível do tratamento experimental, mas sem componente ativo. Mesmo assim, pode haver melhora dos participantes do grupo controle (Efeito Placebo) (44). Quando não for ético suspender o tratamento e administrar placebo, o grupo controle pode ser constituído por indivíduos que recebem o tratamento padrão.

Alocação

A alocação deve ser aleatória. A randomização é a principal técnica para reduzir o viés, criando grupos homogêneos. Como foi visto, é uma das características fundamentais dos ensaios clínicos. O poder da randomização depende da ocultação da sequência de alocação.

A randomização pode ser:

- **Completa:** os indivíduos que obedecem ao critério de inclusão e exclusão são randomizados de modo que todos têm a mesma probabilidade de pertencer a cada um dos grupos. Isto maximiza o poder. Pode ser feita por blocos para assegurar a igualdade numérica dos grupos (estudos multicêntricos).
- **Estratificada:** os participantes são estratificados de acordo com possíveis variáveis de confusão (gravidade da doença, idade, sexo, etc.) e a randomização é realizada dentro de cada estrato.
- **Randomização e alocação desigual:** os sujeitos têm uma maior probabilidade de ser randomizados em um grupo (em geral, grupo experimental) do que o outro (comparação). Este tipo de estudo tem menor poder.

Condução/Seguimento/Avaliação

Em um ensaio clínico deve estar assegurado de que o estudo tenha um tempo de seguimento adequado, pois nem todos os indivíduos participam conforme o plano original. Podem ocorrer perdas de alguns pacientes durante o acompanhamento, seja porque com o tempo se constata que eles não têm a doença em estudo ou porque não aderiram ao tratamento ou intervenção e abandonaram o estudo. Quanto maior o número de pacientes perdidos e menos informações sobre eles, menos confiança pode ser colocada nos resultados do estudo. De um modo geral, não se deve tolerar perdas que sejam maiores que a incidência do desfecho no estudo. Uma regra simples é que perdas menores que 5% produzem pouco viés e perdas maiores que 20% são uma ameaça importante à validade do estudo. As perdas entre 5 e 20% devem ser avaliadas com cuidado, se possível utilizando-se uma análise de sensibilidade (pior cenário), principalmente se as perdas forem diferentes nos grupos pelo maior risco de viés.

Neste tipo de análise, nos estudos com resultado positivo, todos os pacientes perdidos no grupo experimental, inicialmente, são considerados como tendo o desfecho. Posteriormente, analisa-se como se nenhum dos indivíduos perdidos no grupo controle atingiu o desfecho. Se o resultado permanecer positivo, as perdas não afetaram a validade do estudo. Estudos sem relato adequado ou nenhum relato de perdas ou exclusões devem ser avaliados com muito cuidado.

Outro aspecto importante, no seguimento dos sujeitos da pesquisa, é o tratamento igual de todos os grupos. Para garantir este princípio, utiliza-se da técnica de **cegamento** ou **mascaramento** (45). Esta técnica impede que os participantes da pesquisa (pesquisadores, avaliadores e participantes) tomem conhecimento

de qual grupo de tratamento o participante se encontra. Este conhecimento antecipado pode influenciar as expectativas, as opiniões e as crenças em relação aos resultados do estudo. O cegamento tem como principal finalidade a eliminação do viés de aferição, além de melhorar a adesão ao tratamento, reduzir as perdas de seguimento e diminuir o viés causado por co-intervenções (assistência suplementar maior para um dos grupos).

Quando o cegamento ocorre nos pacientes e nos pesquisadores, diz-se que o estudo é **duplo-cego**. Se ele também incluir os avaliadores do estudo, ele é **triplo cego**. Um ensaio clínico em que não há cegamento é dito aberto (*open label*, no caso de estudos com fármacos).

A avaliação dos desfechos também pode afetar os resultados. É importante garantir-se que aqueles que registram os desfechos estejam cegados em relação a que grupo o sujeito da pesquisa pertence. Os autores devem estabelecer regras cuidadosas para decidir se um desfecho ocorreu ou não e despender esforços iguais para identificar desfechos para todos os pacientes no estudo.

Intenção de tratar

Os pesquisadores violam a randomização se omitirem da análise os pacientes que não receberam a intervenção designada ou, pior ainda, contarem eventos que ocorreram nos sujeitos não aderentes que foram designados para a intervenção contra o grupo controle. Os sujeitos de uma pesquisa, para evitar tal viés, devem ser analisados dentro do grupo para o qual eles foram alocados pela randomização (46). Este princípio é denominado **intenção de tratar**.

Análise da magnitude do efeito

Calcula-se uma série de estimativas quantitativas para analisar a magnitude do efeito da intervenção em um ensaio clínico. Entre elas, destacam-se o **Risco Relativo**, **Redução Relativa do Risco**, **Número Necessário para Tratar** que serão estudados no capítulo 18.

Outro método para avaliar resultados de um ensaio clínico para dados de tempo até o evento é a **análise de sobrevida**. Esta fornece informação sobre a rapidez com que os eventos ocorrem. A curva de sobrevida pode utilizar dados de pacientes acompanhados por diferentes períodos de tempo.

3.5.3 Ensaios clínicos de equivalência e não inferioridade

Ensaios clínicos controlados com placebo são ideais para avaliar a eficácia de um tratamento. Eles permitem o controle do efeito placebo e são mais eficientes, exigindo um menor número de pacientes para detectar um efeito do tratamento. Um ensaio clínico placebo controlado é eticamente justificado se não existe tratamento padrão, se o tratamento padrão não se mostrou eficaz, não há riscos associados com o retardo no tratamento e se a possibilidade de se retirar do estudo está incluída no protocolo. Sempre que possível e justificado, os ensaios clínicos placebo controlados devem ser a primeira escolha para avaliação de um tratamento.

Dado que um grande número de tratamentos eficazes comprovados está disponível, ensaios clínicos controlados por placebo são, muitas vezes, antiéticos. Nestas situações, ensaios clínicos com controle ativo são geralmente apropriados.

Se o objetivo do ensaio clínico é testar se um novo tratamento é similar em eficácia a um tratamento já existente, ele é denominado de **Estudo de Equivalência**. O Ensaio Clínico é delineado de maneira que possa demonstrar que, dentro limites aceitáveis, os dois tratamentos são igualmente eficazes. Existe equivalência quando a diferença observada entre os dois tratamentos for menor que a máxima diferença aceitável, determinada previamente. Estes limites devem ser clinicamente apropriados. Se condição em investigação for muito grave, os limites para a equivalência devem ser estreitados. Quanto menor forem os limites de equivalência, maior o tamanho amostral. Este delineamento é útil se o novo tratamento trouxer benefícios, tais como menores efeitos colaterais, facilidade no uso e ser mais barato.

Em muitos estudos com controle ativo, os pesquisadores desejam comprovar que o tratamento em estudo, no mínimo, não é substancialmente pior que o tratamento controle. Estes estudos são chamados de **Estudos de Não Inferioridade**. Um aspecto importante do delineamento e da interpretação desses estudos é a

determinação da margem de não inferioridade. Os estudos de não inferioridade devem demonstrar, pelo menos, que o tratamento em estudo tem alguma eficácia, não inferior ao tratamento padrão. A análise dos estudos de não inferioridade é, por natureza, unidirecional.

Quando um ensaio clínico busca evidenciar que um tratamento é melhor do que outro ele é denominado **Estudos de Superioridade**. Quando o ensaio clínico é delineado, ele deve ter uma hipótese bilateral e o tamanho da amostra definido de maneira que haja alto poder estatístico para detectar uma diferença clinicamente significativa entre os dois tratamentos. Os ensaios clínicos clássicos têm esta característica. Entretanto, nos dias atuais, este desenho de estudo pode não ser eticamente possível, uma vez que é pouco provável que não exista um tratamento com algum benefício comprovado. A comparação, portanto, deverá ser feita com o tratamento já existente, provando que o tratamento em estudo é similar ou, pelo menos, não seja inferior (47).

3.5.4 Outros tipos de ensaios clínicos

Ensaio clínico com delineamento cruzado

No delineamento cruzado (*crossover design*), os sujeitos da pesquisa são randomizados para um grupo e depois mudados para o outro grupo (Figura 17). Cada sujeito serve como seu próprio controle, diminuindo a variabilidade intragrupo, aumentando o poder e consequentemente, reduzindo o erro β (erro que ocorre quando a análise estatística dos dados não consegue rejeitar uma hipótese, no caso desta hipótese ser falsa). É um tipo de delineamento bastante atrativo e útil (48).

A maior desvantagem é o efeito residual (*carryover*), por isso os estudos cruzados devem ter um período de *washout*, período sem nenhum tratamento. Este período de tempo deve ser suficiente para a eliminação da droga para se ter certeza de que nenhum efeito da terapia permaneceu. Também pode haver um viés de acordo com a ordem de administração das terapias, pois os pacientes podem reagir de modo diferente como resultado do entusiasmo no início do tratamento que pode diminuir com o tempo.

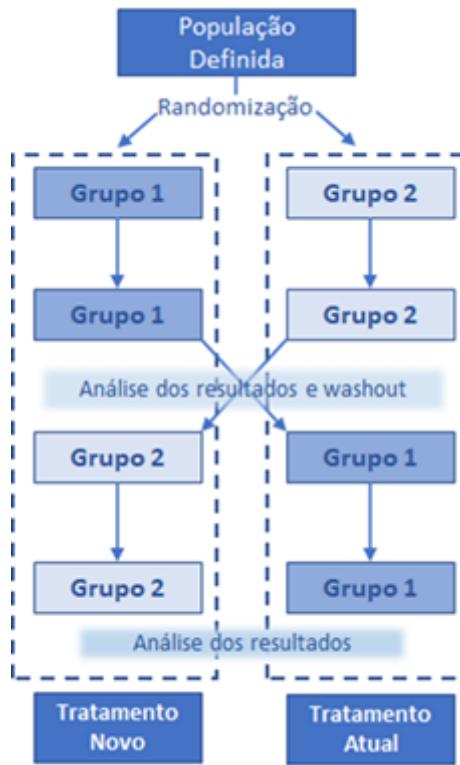


Figura 17: Ensaio clínico randomizado com delineamento cruzado.

Delineamento Fatorial

Uma variação interessante de ensaio clínico é o *delineamento fatorial*. Este tipo de estudo permite que sejam testadas duas drogas em apenas um estudo, assumindo que os desfechos antecipados para as duas são diferentes e que seus modos de ação são independentes. Este desenho de estudo gera economia.

Um exemplo de delineamento fatorial é observado no *Physician's Health Study* onde usando um delineamento fatorial 2 x 2 foi testada a aspirina para a prevenção primária de doença cardiovascular (49), e betacaroteno para a prevenção primária de câncer.

No estudo da prevenção primária do câncer, os autores concluíram, após 12 anos de suplementação de betacaroteno, que o mesmo não produziu nem benefícios e nem prejuízos em termos de incidência de câncer (50).

3.5.5 Fases de um ensaio clínico

Para a realização de um ensaio clínico, a intervenção deve passar por várias fases (51).

Fase Não Clínica

Antes de começar a testar novos tratamentos em seres humanos, os cientistas testam as substâncias em laboratórios (*in vitro*) e em animais de experimentação. O objetivo principal desta fase é verificar como esta substância se comporta em um organismo. Assim, após esta fase se pode verificar se o medicamento é seguro para ser testado em seres humanos. Todo este processo é regido por leis da bioética em pesquisa em animais.

Fase Clínica

A fase clínica é a fase de testes em seres humanos. Esta etapa é constituída por quatro fases consecutivas e somente depois de finalizadas todas as fases, a droga poderá ser autorizada para comercialização e disponibilizada para uso em seres humanos. As sucessivas fases dentro da fase clínica são:

- *Fase I* - Um estudo de fase I testa a droga pela primeira vez. O objetivo principal é avaliar a segurança do produto investigado. Nesta fase, o medicamento é testado em pequenos grupos (10 – 30 pessoas), geralmente, de voluntários sadios. Podemos ter exceções se estivermos avaliando medicamentos para câncer ou portadores de HIV-AIDS. Se a droga se mostrar segura, é possível ir para a Fase II.
- *Fase II* - Nesta fase, o número de pacientes é maior (70 - 100). O objetivo é avaliar a eficácia da medicação, isto é, se ela funciona para tratar determinada doença, e também conseguir informações mais detalhadas sobre a segurança (toxicidade). Somente se os resultados forem bons é que o medicamento será estudado como um estudo clínico fase III.
- *Fase III* - Nesta fase, o novo tratamento é comparado com o tratamento padrão existente. São os ensaios clínicos. O número de pacientes aumenta e depende da hipótese (em geral, 100 a 1.000). Devem de preferência utilizar desfechos clínicos, grupo controle, além de serem randomizados e duplo-cegos.
- *Fase IV* - Estes estudos são realizados para se confirmar que os resultados obtidos na fase III são aplicáveis a grande parte dos doentes. Nesta fase, o medicamento já foi aprovado para ser comercializado. A vantagem dos estudos fase IV é que eles permitem acompanhar os efeitos dos medicamentos em longo prazo. É uma fase de vigilância pós-comercialização.

4 Ambiente do R

4.1 Instalação do R básico

Para usar o R, há necessidade de carregar o programa básico que contém a sua linguagem de programação. O sistema é formado por um programa básico, *Graphical User Interface* (R-Gui) e muitos pacotes com procedimentos adicionais.

O [site](#) oficial do R fornece as versões atualizadas do software e informações sobre este sofisticado projeto de computação estatística.

Para baixar o R, usa-se um “CRAN Mirror”, clicando em CRAN (*Comprehensive R Archive Network*) na margem esquerda, abaixo de *Download*. O CRAN é central no uso do R: é o local de onde se carrega o software e todos os pacotes necessários para instalar e para expandir o R.

Em vez de ter um único local, o CRAN é “espelhado” em diferentes locais do mundo. “Espelhado” significa simplesmente que existem versões idênticas do CRAN distribuídas por todo o mundo. É possível baixar o R diretamente da [nuvem](#) ou escolher uma origem mais próxima do seu local de atuação. No Brasil, encontram-se várias opções, como a [Universidade Federal do Paraná](#), [Fundação Oswaldo Cruz](#), [RJ](#), [Universidade de São Paulo](#), [São Paulo](#) e [Universidade de São Paulo, Piracicaba](#).

Após escolher uma das alternativas acima (pode ser qualquer uma delas) surgirá a página *The Comprehensive R Archive Network* com as opções para escolher o sistema operacional. Escolha o sistema de acordo com o seu computador (Windows, macOS ou Linux). Ao clicar em uma dessas opções, se o sistema operacional escolhido é o Windows, aparecerá a página *R for Windows*. Nesta, deve-se clicar em [base](#). No caso de outros sistemas operacionais, seguir as orientações mostradas no site do R.

Clicando em [base](#), haverá um redirecionamento para a página onde aparece a versão do R para o Windows mais atual. Clique no link que diz *Download R-...for Window* para baixar o instalador em um diretório do computador, em geral *Downloads*.

Para instalar o programa básico, basta executar o instalador *R-...-win.exe* baixado no diretório. Ao fazer isso, aparece na tela do computador, no canto esquerdo, em baixo, o arquivo salvo. Execute este arquivo com um clique sobre ele. Aparecerá uma janela perguntando “*Deseja permitir que este aplicativo faça alterações no seu dispositivo?*”. Clique em *Sim*. A seguir o instalador pedirá para escolher o Idioma. Selecione Português Brasileiro.

Em sequência aparecerão informações sobre o diretório no qual o R será instalado em seu computador. Recomenda-se aceitar a configuração padrão sugerida pelo instalador do software.

A próxima janela pedirá para personalizar os componentes que serão instalados. Recomenda-se usar as configurações sugeridas pelo instalador que irá reconhecer automaticamente a arquitetura do seu sistema Windows (32 e/ou 64 bits).

A partir daqui, siga as recomendações padrão propostas pelo instalador até completar a instalação, clicando em *Concluir*.

O R não precisa ser iniciado, pois o software que será usado, neste livro, é o *RStudio*. Este, para ser executado, necessita ter o R instalado no computador. Ou seja, o R é o programa “cérebro” necessário para as análises de dados que serão realizadas. Ele precisa estar instalado para permitir o funcionamento do *RStudio*.

4.2 RStudio

O **RStudio** é um membro ativo da comunidade R. Foi fundado em 2009 por Joseph J. Allaire, engenheiro de software americano. O RStudio, inspirado pelas inovações dos usuários de R em ciência, educação e indústria, desenvolveu ferramentas gratuitas e abertas para facilitar o uso do R.

O RStudio é um projeto filiado à *Foundation for Open Access Statistics* (FOAS). A FOAS trabalha para garantir o sucesso do projeto R. Eles promovem o uso e o desenvolvimento de software livre para estatísticas, como a linguagem R e o ambiente para estatísticas computacionais. Junto está o *R Consortium* que é uma colaboração entre a Fundação R, RStudio, Microsoft, TIBCO, Google, Oracle, HP e outros.

O RStudio é patrocinado para financiar e inspirar ideias que permitirão que o R se torne uma plataforma ainda melhor para a ciência.

4.2.1 Instalação do R Studio

Para instalar o RStudio, acessar o [site](#) e clicar em *Download* para obter a versão desejada. Recomenda-se a versão *RStudio Desktop – Open Source License* que é gratuita. Esta versão entrega as ferramentas integradas para o R.

A seguir, aparecerão os instaladores disponíveis, conforme a plataforma suportada pelo seu computador. As mais utilizadas são Windows e Mac OS X. Neste livro, como base, serão mostrados os passos para a plataforma Windows³.

Em sequência, executar o instalador baixado RStudio-2022.07.2-576.exe⁴ e seguir as suas instruções.

4.2.2 Iniciando o RStudio

Para iniciar o RStudio basta clicar no ícone indicativo (Figura 18) que se encontra no menu *Iniciar* do Windows.



Figura 18: Ícone do RStudio

O RStudio abre como mostrado na Figura 19. O RStudio é uma interface mais funcional e amigável para o R. Contém um conjunto de ferramentas integradas projetadas para ajudá-lo a ser mais produtivo com o R.

³A instalação para Mac OS X pode ser facilmente obtida em busca do Google. Depois de instalado, o uso do RStudio não difere do Windows

⁴Versão disponível em 23/10/2022

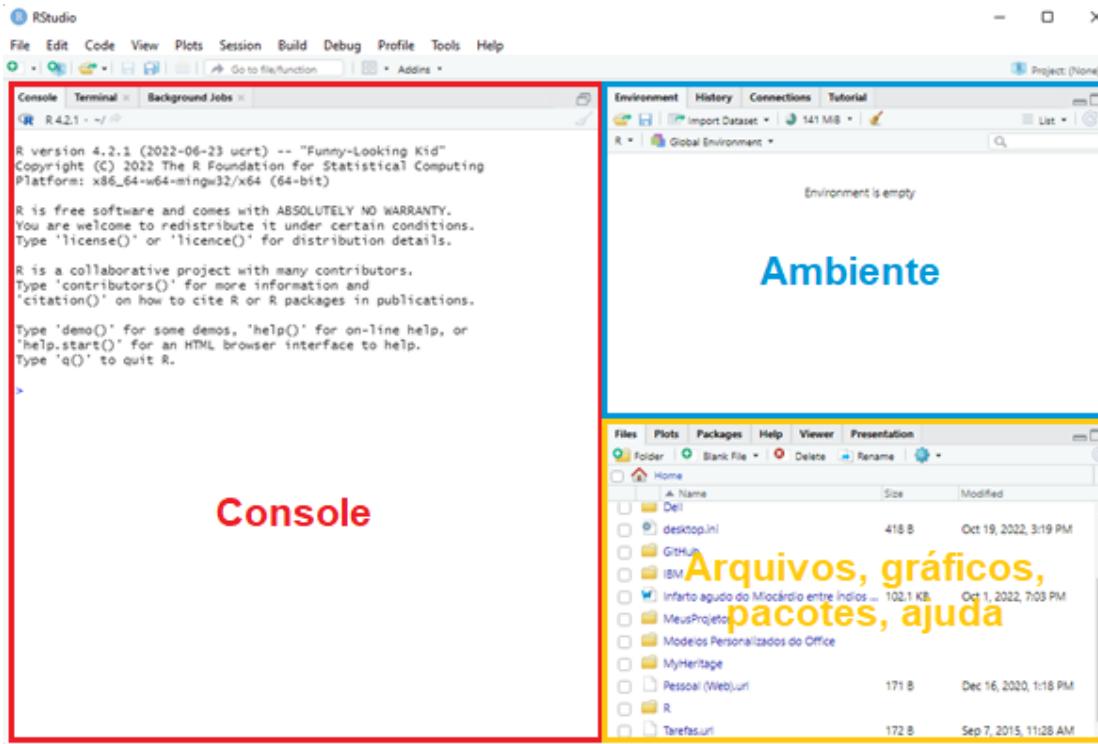


Figura 19: Tela inicial do RStudio

Inclui um console, editor texto que suporta execução direta de códigos e uma variedade de ferramentas robustas para plotagem, exibição de histórico, depuração e gerenciamento de seu espaço de trabalho incluídos em uma interface que está, inicialmente, dividida em 3 painéis:

1. *Console*
2. *Environment, History, Connections, Tutorial*
3. *Files, Plots, Packages, Help*

Console e R Script

Do lado esquerdo fica o **Console** (Figura 19, em vermelho), onde os comandos podem ser digitados e onde aparecem os resultados da execução dos comandos. Ao abrir o RStudio, aparece no *Console* uma série de informações sobre o R, como versão em uso e, por último, o diretório onde está armazenado o espaço de trabalho (*workspace*). Estas informações podem ser facilmente apagadas, clicando na barra de ferramentas, no menu *Edit*, e após em *Clear Console* ou, usando as teclas *Ctrl+L*.

O *Console* é a principal parte do R. Aqui é onde o R realmente executa o comando. No início do *Console*, existe um caractere (`>`). Este é um *prompt* que informa que o R está pronto para receber um novo código. Pode-se digitar o código diretamente no *Console* após o *prompt* e obter uma resposta imediata. Por exemplo, se for digitado `1 + 1` e pressionado *Enter*, o R imediatamente gera uma saída de `2` (Figura 20).

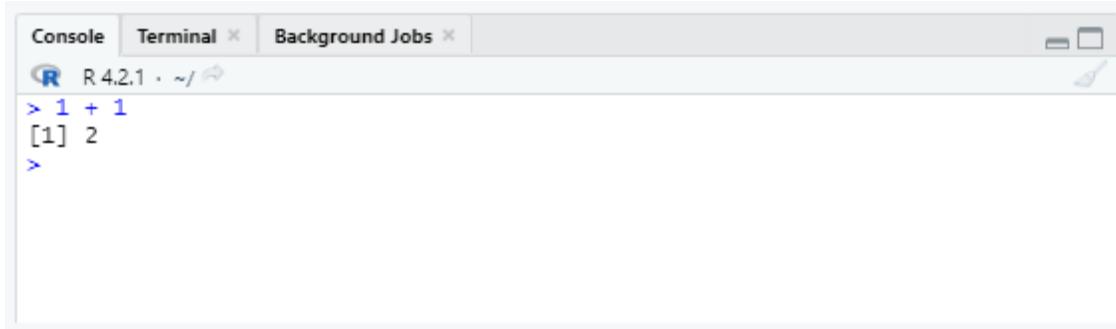


Figura 20: Console

Recomenda-se que a maior parte dos comandos sejam digitados no bloco de notas do RStudio, o *R Script*. Reservar o *Console* apenas para depurar ou fazer análises e cálculos rápidos. A razão para isso é simples: se o comando for digitado diretamente no *Console*, ele não será salvo e se for cometido um erro na digitação, haverá necessidade de digitar tudo novamente. Portanto, é melhor escrever os comandos no *R Script* e, quando estiver pronto para executar, enviando para o *Console*.

O *R Script* é o quarto painel do *RStudio* e seu bloco de notas. Ele é criado através do menu *File > New File > R Script* ou clicando no **botão verde** com o sinal (+), na barra de ferramentas de acesso rápido, na parte superior à esquerda. Ao criar um novo *R Script* será aberto o painel do bloco de notas (Figura 21, em verde).

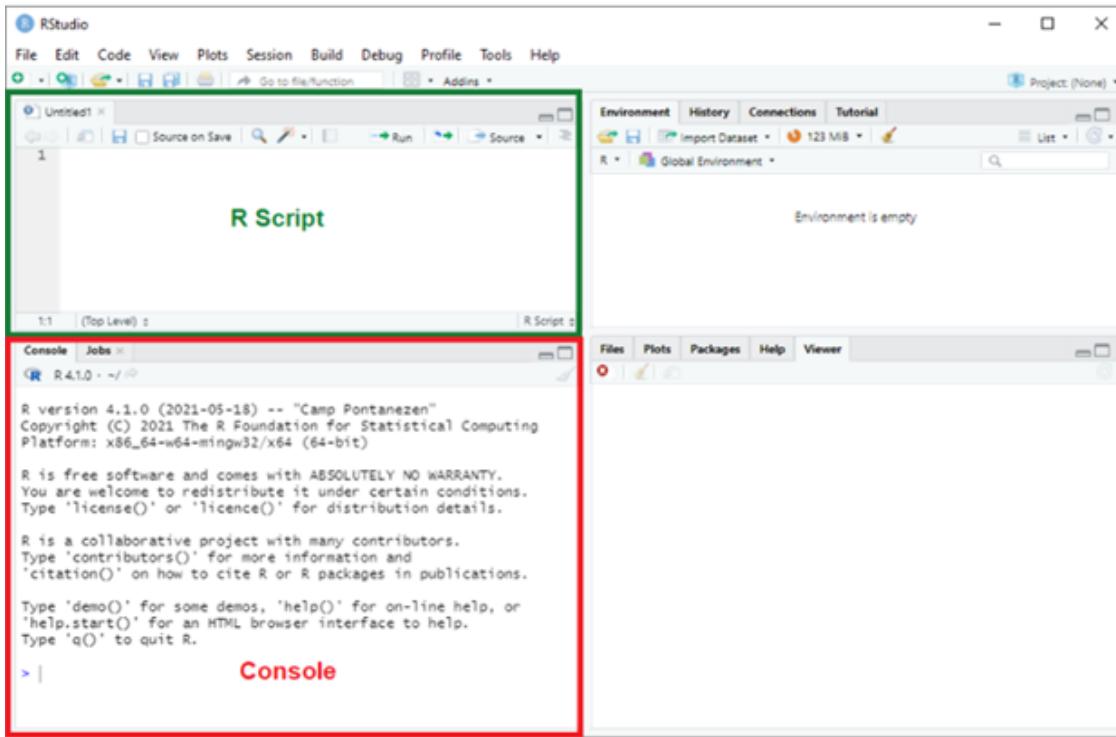


Figura 21: R Script

Um diferencial do *RStudio* é que os comandos são autocompletáveis. Basta começar a escrever o comando, inserindo 3 ou mais caracteres, por exemplo, `summ` referente a função `summary ()`, usada para sumarizar um conjunto de dados, e surge um menu de opções, facilitando a digitação (Figura 22).

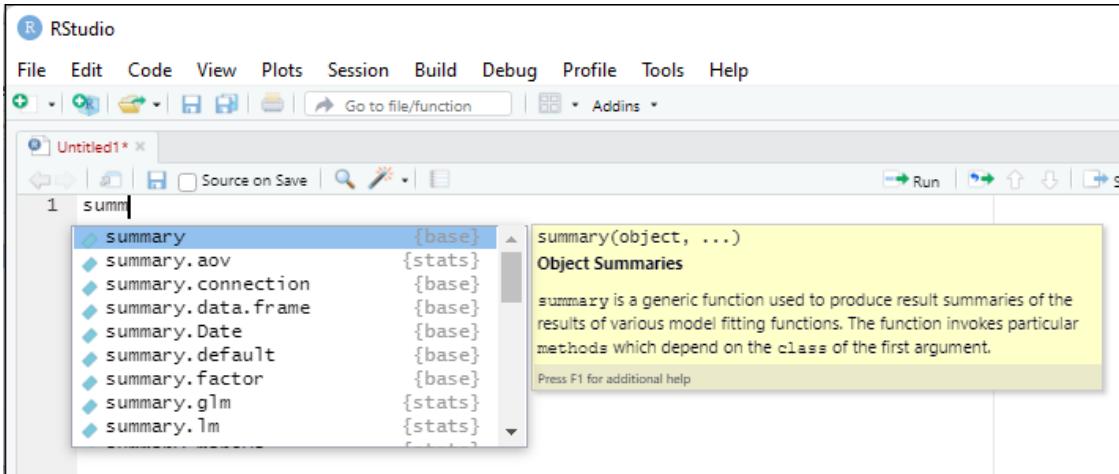


Figura 22: Menu autocompletável

Após digitar no *Console*, para que seja executado o comando há necessidade de clicar na tecla *Enter*; no *RScript*, clicar em *Run*, acima, na barra, ou usar o atalho *Ctrl + Enter*. Textos podem ser copiados e colados no script e linhas em branco podem ser inseridas. Além disso, no final da sua sessão, é possível salvar o arquivo, que poderá ser recarregado no futuro, se precisar refazer a análise.

Os *scripts* do R são apenas arquivos de texto com a extensão (.R). Quando se cria um *R Script*, aparece como *Sem título* (*Untitled*). Antes de começar a digitar um novo *script* no R *Sem título*, recomenda-se salvar o arquivo com um novo nome de arquivo. Dessa forma, se algo no computador falhar durante o trabalho, o R terá o código protegido.

Ao digitar o código em um *script*, o R não executa o código enquanto se digita. Para que o R realmente avalie o código digitado, há necessidade de primeiro enviar o código para o *Console*, clicando no botão *Run* ou usando a tecla de atalho *Crtl+Enter*. Cada linha é marcada no início por um número em sequência.

Além da digitação de comandos, o *R Script* permite fazer comentários onde tudo que for escrito após o símbolo `#` são considerados apenas como comentários. Os comentários são literais, escritos diretamente para explicar o comando executado. São repetidos na saída do Console sem não aparecer nos resultados.

Ambiente, História, Conexão e Tutorial

No lado superior direito há um painel com quatro abas (Figura 19, em azul):

- 1) **Ambiente** (Environment) - onde ficam armazenados os objetos criados, as bases de dados importadas, etc., na sessão ativa. É possível visualizar informações como o número de observações e linhas dos bancos de dados ativos. A guia também tem algumas ações clicáveis, como *Import Dataset*, que permite importar arquivos csv, Excel, SPSS, etc.
- 2) **História** (History) - onde fica o histórico dos comandos executados no *Console*. Estes comandos podem ser pesquisados nesta guia. Os comandos são exibidos em ordem (mais recentes na parte inferior) e agrupados por bloco de tempo.
- 3) **Conexões** (Connections) - mostra todas as conexões feitas com fontes de dados suportadas e permite saber quais conexões estão ativas no momento. O *RStudio* suporta múltiplas conexões de banco de dados simultâneas.
- 4) **Tutorial** - a partir da versão 1.3, o *R Script* ganhou um painel Tutorial dedicado, usado para executar tutoriais que ajudarão você a aprender e dominar a linguagem de programação R. Na primeira vez que se abre o programa, clicando nesta aba, o *RStudio* solicita que seja instalado o pacote `learnr` (Figura 23). Isto permite acesso a vários tutoriais úteis que merecem ser explorados

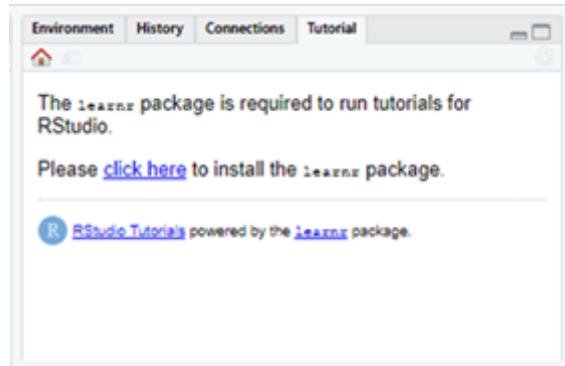


Figura 23: Tutoriais do RStudio

Arquivos, Gráficos, Pacotes e Ajuda

No lado direito, abaixo, existem outras abas muito úteis (Figura Figura 19, em amarelo):

- 1) **Arquivos (Files)** - esta guia dá acesso ao diretório onde se encontram os seus arquivos. Um bom recurso do painel *Files* é que se pode usá-lo para definir seu diretório de trabalho. Para isso, clique em *More* e depois em *Set As Working Directory*.
- 2) **Gráficos (Plots)** - local onde ficam os gráficos gerados. Existem botões para abrir o gráfico em uma janela separada e exportar o gráfico como um *.pdf* ou *.jpeg*.
- 3) **Pacotes (Packages)** - mostra uma lista de todos os pacotes R instalados no seu computador e indica se eles estão atualmente carregados ou não. Pacotes que estão sendo executados na sessão atual, estão marcados, enquanto aqueles que estão instalados, mas ainda inativos, estão desmarcados.
- 4) **Ajuda (Help)** - menu de ajuda para as funções R. Você pode digitar o nome de uma função na janela de pesquisa (por exemplo, `histogram` ou usar o `?hist`), no *Console* ou no *R Script*, para procurar ajuda sobre uma função (Figura 24). A Ajuda no *R Studio* pode também ser acessada no menu *Help* da barra de ferramentas onde existem várias opções. Para complementar, alguns livros são muito úteis, como o *R Cookbook* (52) ou *Using R for introductory statistics* (53). No entanto, na maioria das vezes a forma mais prática de conseguir ajuda com uma dúvida específica é a busca em fóruns na internet, como o *Stack Overflow*: <https://stackoverflow.com/>.

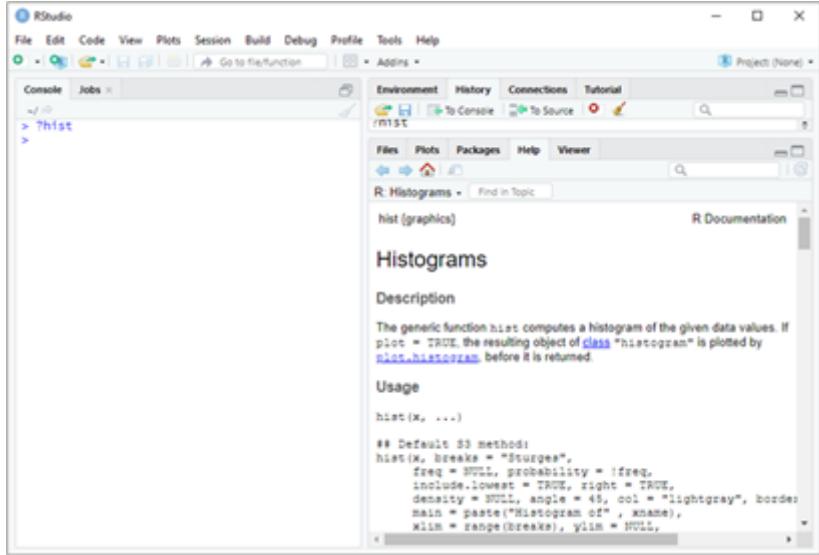


Figura 24: Ajuda do RStudio

4.3 Pacotes

Para que o R cumpra a sua função de dialogar com o usuário para realizar análises estatística e construir gráficos, ele necessita ter instalado pacotes.

Quando se instala o R básico, ele vem com vários pacotes que permitem uma grande quantidade de análises. Entretanto, à medida que se utiliza o R, torna-se necessário instalar novos pacotes criados pela comunidade do R. Esses novos pacotes contêm novas funções e novos comandos que aumentarão a funcionalidade do R.

Um pacote é uma coleção de funções, dados e documentação que expande os recursos do R base. O uso dos pacotes é a chave para o uso bem-sucedido do R. Eles são instalados à medida que o trabalho com o R exigir.

4.3.1 Repertório de pacotes

Quando se identifica a necessidade de um novo pacote, há necessidade de saber onde ele se encontra. O principal repositório de pacotes é o CRAN (*Comprehensive R Archive Network*), já comentado anteriormente. Para acessar este repositório, use o [link](#) e escolha um espelho (*0-Cloud* ou o mais próximo geograficamente). Depois que o pacote for instalado, ele será mantido em sua biblioteca (*library*) R associada à sua versão principal atual do R. Haverá necessidade de atualizar e reinstalar os pacotes sempre que atualizar uma versão principal do R.

Estando na página do CRAN, no menu, à esquerda, clique em *Packages*. Isto o colocará na página dos *Contributed Packages*, onde a maioria dos pacotes podem ser encontrados em *Table of available packages, sorted by name*. Também é possível clicar em *CRAN Task Views*, onde encontramos os pacotes separados por tópicos.

4.3.2 Instalação de um novo pacote

Instalar um pacote significa simplesmente baixar o código do pacote em um computador pessoal. Existem duas maneiras principais de instalar novos pacotes. O método mais comum é baixá-los do CRAN, usando a função `install.packages()`. Dentro dos parênteses, como argumento, coloca-se entre aspas (duplas ou simples) o nome do pacote. Como visto, deve-se, de preferência, digitar o comando no *R Script*. Por exemplo, será instalado o pacote `ggplot2` que contém múltiplas funções gráficas como abaixo:

```
install.packages("ggplot2")
library(ggplot2)
```

Para carregar o pacote, isto é, para fazer com que suas funções se tornem ativas para uso na sessão, deve-se usar a função `library()`, como mostrado no comando acima. Se o *RStudio* for fechado e reaberto, o pacote deverá ser novamente ativado. Observe que a função `library()` não requer que o nome do pacote seja digitado entre aspas. Isto acontece porque antes de o pacote ser instalado o R não o reconhece, portanto, há necessidade de indicar o nome (caracteres), para que o R procure na internet, por exemplo, o que ele deve baixar. Já, depois de instalado, o pacote é um objeto conhecido pelo R, logo as aspas não são mais necessárias.

Uma outra maneira de instalar pacotes no R, é usar o botão **Install**, localizado na aba *Packages*, no painel inferior, à direita. Clicando em **Install**, abre-se a caixa de diálogo da Figura 25. Digitar em *Packages* o nome do pacote (`ggplot2`) e o *RStudio* completará com opções para achar o pacote. Clicar em `ggplot2` e verifique se *Install dependencies* foi selecionado. A seguir clicar em *Install* e aguardar aparecer no *Console* a mensagem que o pacote foi instalado com sucesso.

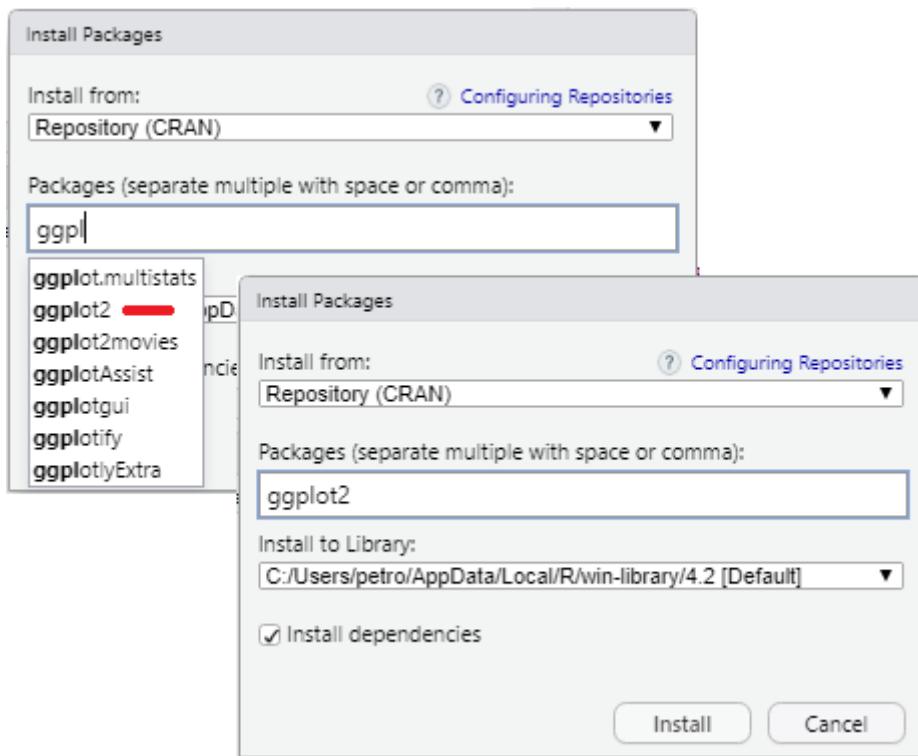


Figura 25: Instalação do pacote 'ggplot2' usando a caixa de diálogo 'Install Packages'

4.3.3 Atualização dos pacotes

Periodicamente, há necessidade de atualizar os pacotes instalados. Essa necessidade advém do fato que, com o tempo, os autores de pacotes lançarão novas versões com correções de defeitos e novos recursos e, geralmente, é uma boa ideia manter-se atualizado. Para realizar a atualização proceda da seguinte maneira:

```
# atualiza todos os pacotes disponíveis, solicitando permissão
update.packages()
```

```
# atualiza, sem solicitações de permissão/esclarecimento  
update.packages(ask = FALSE)  
  
# atualiza um pacote específico  
update.packages("ggplot2")
```

4.3.4 Instalando e carregando mais de um pacote

Para carregar mais de um pacote simultaneamente, pode-se usar uma das funções: `libraries()` ou `packages()` do pacote `easypackages`. Em primeiro lugar, instalar e carregar o pacote:

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

Posteriormente, basta usar uma das funções do `easypackages`:

```
libraries("readxl", "dplyr", "ggplot2", "car")
```

Outro pacote que gerencia pacotes do R é o `pacman`. Este pacote tem a função `p_load()` que instala e carrega um ou mais pacotes. Usar esta função, escrevendo o nome dos pacotes sem necessidade de aspas:

```
install.packages("pacman")  
library(pacman)  
  
p_load(readxl, dplyr, ggplot2, car)
```

Ou, escrever diretamente:

```
pacman::p_load(readxl, dplyr, ggplot2, car)
```

O pacote `pacman` tem outras funções, entre elas a função `p_update()` que atualiza o pacote e , se usada sem especificar o pacote , atualiza todos. Para saber mais sobre o pacote `pacman`, use a ajuda.

```
p_update(readxl, dplyr, ggplot2, car)
```

4.3.5 Citação de pacotes em publicações?**

No R existe um comando que mostra como citar o R ou um de seus pacotes. Basta digitar a função `citation()` no *Console* ou no *R Script* e observar a saída. Para um pacote específico, basta colocar o nome do pacote entre aspas, na função.

```
citation()  
  
##  
## To cite R in publications use:  
##  
##   R Core Team (2022). R: A language and environment for statistical  
##   computing. R Foundation for Statistical Computing, Vienna, Austria.  
##   URL https://www.R-project.org/.
```

```

## A BibTeX entry for LaTeX users is
##
## @Manual{,
##   title = {R: A Language and Environment for Statistical Computing},
##   author = {{R Core Team}},
##   organization = {R Foundation for Statistical Computing},
##   address = {Vienna, Austria},
##   year = {2022},
##   url = {https://www.R-project.org/},
## }
##
## We have invested a lot of time and effort in creating R, please cite it
## when using it for data analysis. See also 'citation("pkgnname")' for
## citing R packages.

citation ("ggplot2")

##
## To cite ggplot2 in publications, please use:
##
## H. Wickham. ggplot2: Elegant Graphics for Data Analysis.
## Springer-Verlag New York, 2016.
##
## A BibTeX entry for LaTeX users is
##
## @Book{,
##   author = {Hadley Wickham},
##   title = {ggplot2: Elegant Graphics for Data Analysis},
##   publisher = {Springer-Verlag New York},
##   year = {2016},
##   isbn = {978-3-319-24277-4},
##   url = {https://ggplot2.tidyverse.org},
## }

```

4.4 Diretório de trabalho

O diretório de trabalho (**Working Directory**) é uma pasta onde o R lê e salva arquivos. Deve-se criar um diretório de trabalho para a sessão . Para isso, no *RStudio* siga o caminho: *Session > Set Working Directory > Choose Directory* ou use o atalho *Ctrl + Shift + H* e escolha o diretório desejado ou crie um novo.

Ao finalizar, aparecerá no *Console* (Figura 26):

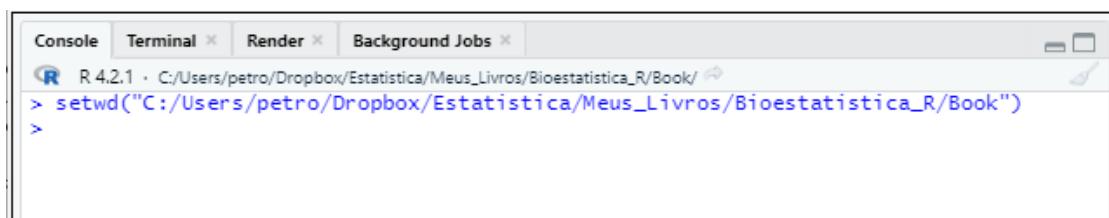


Figura 26: Diretório de trabalho

Note que o R usou a função `setwd()` que significa “definir diretório de trabalho”. Também é possível usar esta função diretamente no *R Script* ou no *Console*, digitando conforme o caminho do diretório.

Para saber qual é o diretório de trabalho que está sendo usado pelo R pode-se executar a função `getwd()`. A saída no *Console* mostrará o diretório de trabalho usado, portanto é recomendado que se faça isso no início da sessão para verificar se há ou não necessidade de modificar o diretório.

4.5 Projeto

Uma funcionalidade importante do *RStudio* é a possibilidade de se criar projetos. Um projeto nada mais é do que uma pasta no seu computador. Nessa pasta, estarão todos os arquivos que serão usados ou criados na sua análise.

A principal razão de se utilizar projetos é simplesmente *organização*. Com eles, fica muito mais fácil importar conjunto de dados para dentro do R, criar análises reproduutíveis e compartilhar o trabalho realizado.

Ao se começar uma nova análise, é interessante criar um Novo Projeto. Para isso, clicar *File > New Project* ou clicar no menu que está na parte superior, à direita, *Project (none) > New Project....* Abrirá a janela da Figura 27.

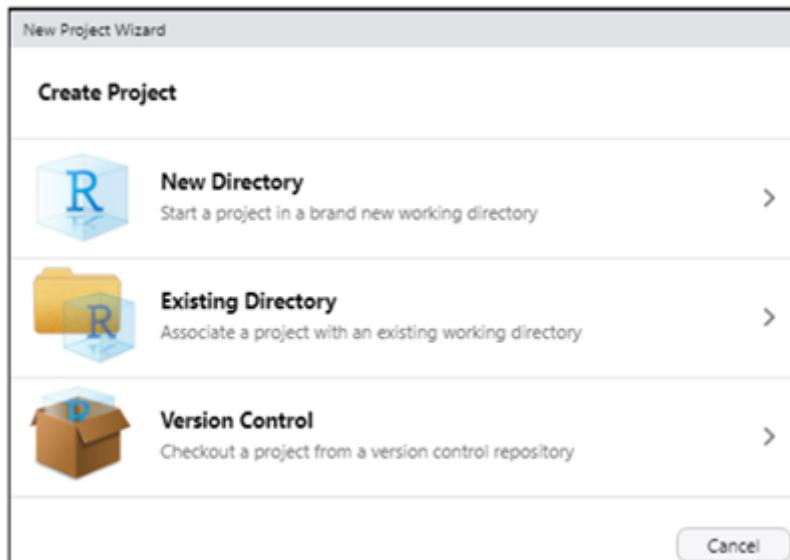


Figura 27: Assistente de novo projeto.

Clique em *New Directory* para criar um novo diretório. Por exemplo, para as aulas de Bioestatística, pode-se criar um diretório com, por exemplo, o nome *bioestatistica* (evite usar acentos, maiúsculas ou caracteres especiais) ou qualquer outro nome.

Quaisquer documentos Excel ou arquivos de texto associados podem ser salvos nesta nova pasta e facilmente acessados de dentro de R, indo ao menu *Project (none) > Open Project....* A partir daí, é possível realizar análises de dados ou produzir visualizações com seus dados importados.

Quando um projeto estiver aberto no *RStudio*, o seu nome aparecerá no canto superior direito da tela. Na aba *Files*, aparecerão todos os arquivos contidos no projeto. Quando se clica no nome do projeto, abre um menu que torna muito fácil a navegação pelos projetos existentes. Basta clicar em qualquer um deles para trocar de projeto, isto é, deixar de trabalhar em uma análise e começar a trabalhar em outra.

4.6 O R como calculadora

O R pode ser utilizado para uma série de operações matemáticas desde as mais simples às mais complexas. Para isso, basta digitar no Console ou no R Script, usando os operadores.

4.6.1 Operadores

Operadores são usados para realizar operações em variáveis e valores.

Por exemplo, o operador `+` (adição) é usado para somar dois valores:

```
10 + 5
```

```
## [1] 15
```

Operadores aritméticos

No R, você pode usar operadores aritméticos para realizar operações matemáticas comuns.

```
# Adição  
10 + 5
```

```
## [1] 15
```

```
# Subtração  
10 - 5
```

```
## [1] 5
```

```
# Multiplicação  
10 * 5
```

```
## [1] 50
```

```
# Divisão  
10 / 5
```

```
## [1] 2
```

```
# Potência  
10 ^ 5
```

```
## [1] 1e+05
```

```
# Divisão modular (divisão com resto)  
10 %% 3
```

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

```
# Divisão inteiro  
10 %/% 3
```

```
## [1] 3
```

O resultado da exponenciação é exibido como notação científica, onde $e + 05$ significa 10^5 .

Operadores de atribuição

Operadores de atribuição são usados para atribuir valores a variáveis, como será visto na seção *Objetos*, adiante.

Operadores de comparação

São usados para comparar dois valores.

```
# Igualdade  
3 == 3
```

```
## [1] TRUE
```

```
3 == 4
```

```
## [1] FALSE
```

```
# Não igual (diferente)  
3 != 4
```

```
## [1] TRUE
```

```
# Maior  
6 > 3
```

```
## [1] TRUE
```

```
# Menor  
3 < 4
```

```
## [1] TRUE
```

```
# Maior ou igual  
5 >= 3
```

```
## [1] TRUE
```

```
# Menor ou igual  
3 <= 4
```

```
## [1] TRUE
```

Observe que, na linguagem R, o sinal de igualdade é escrito com duplo =.

Operadores lógicos

Operadores lógicos são usados para combinar declarações condicionais:

```
# Conjunção lógica E, retorna TRUE se ambos elementos são verdadeiros  
6 == 6 & 7 == 8
```

```
## [1] FALSE
```

```
# Conjunção lógica E, retorna TRUE se ambos elementos são verdadeiros  
2 * 3 && 1 * 6
```

```
## [1] TRUE
```

```
# Conjunção lógica OU, retorna TRUE se um dos elementos é verdadeiro  
(2 * 2) | sqrt(16)
```

```
## [1] TRUE
```

```
6 == 6 | 7 == 8
```

```
## [1] TRUE
```

```
# Conjunção lógica NÃO, retorna FALSE se o elemento é verdadeiro  
!6==6
```

```
## [1] FALSE
```

```
!2==4
```

```
## [1] TRUE
```

Logarítmico

```
# Logarítmico natural (base e)  
log (10)
```

```
## [1] 2.302585
```

```
# Logarítmico base 10  
log10 (10)
```

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

Raiz quadrada

```
sqrt (81)
```

```
## [1] 9
```

Resultado absoluto

```
abs (3 - 6)
```

```
## [1] 3
```

4.7 Objetos

O R permite salvar valores dentro de um *objeto*. Os objetos são criados utilizando o *operador de atribuição* `<-`. Para digitar este operador, basta teclar o sinal *menor que* (`<`), seguido de *hífen* (`-`) , sem espaços. Existe um atalho que é pressionar (**Alt**) + (`-`). O símbolo `=` pode ser usado no lugar de `<-`, mas não é recomendado.

Objeto é um pequeno espaço na memória do computador onde o R armazenará um valor ou o resultado de um comando, utilizando um nome arbitrariamente definido. Tudo criado pelo R pode se constituir em um objeto, por exemplo: uma variável, uma operação aritmética, um gráfico, uma matriz ou um modelo estatístico. Através de um objeto torna-se simples acessar os dados armazenados na memória. Ao criar um objeto, se faz uma declaração. Isto significa que se está afirmando, por exemplo, que uma determinada operação aritmética irá, agora, tornar-se um objeto que irá armazenar um determinado valor. As declarações são feitas uma em cada linha do *R Script*.

Os objetos devem receber um nome e é obrigatório que ele comece por uma letra (ou um ponto) e não é permitido o uso do hífen. Pode-se usar o ponto e *underlines* para separar palavras. Deve ser evitado o uso de nomes que sejam de objetos do sistema, ou outros objetos já criados, funções ou constantes. Por exemplo, não deve ser utilizado: `c, q, r, s, t, C, D, F, I, T, diff, exp, log, mean, pi, range, rank, var, NA, NaN, NULL, FALSE, TRUE, break, else, if, break, function, in, while` que devem ser reservados, pois têm significados especiais.

Quando se usa um objeto com o nome `pi`, ele assumirá outro valor diferente de 3,141593. Preservando este nome, toda vez que usarmos a palavra `pi`, o R assume o valor pré-estabelecido. Além disso, o R faz a diferença entre letras maiúsculas e minúsculas. Ou seja, `soma` é um objeto diferente de `Soma` e ambos são diferentes de `SOMA`.

Para exibir o conteúdo de um objeto, basta digitar seu nome no *R Script* ou no *Console* e executar. Em análises mais extensas, verificar se já há um objeto com o mesmo nome, pois seus valores serão substituídos ao executar o novo objeto. Para saber se já existe um objeto com o nome definido, digite as primeiras letras do objeto criado e o *R Studio* listará, usando a sua função de autocompletar, tudo que começar com essas letras no arquivo. Assim ficará fácil verificar se já existe um objeto com o nome desejado.

No comando abaixo, é criado um objeto que receberá a soma de dez números, utilizando a função `sum()`. O objeto foi denominado de `soma`. Para exibir o valor contido no objeto `soma`, é necessário digitar `soma` no *R Script* ou *Console* e executar:

```
soma <- sum (2, 3, 12, 15, 21, 4, 8, 7, 13, 21)
soma
```

```
## [1] 106
```

4.8 Funções

A função é uma orientação ao R para que ele execute algum procedimento específico, por isso, em geral, têm nomes sugestivos do que elas realizam. Por exemplo, a função `mean ()` realiza a média aritmética de uma série de números colocados entre parênteses. O resultado, como regra geral, deve ser colocados em um objeto que será armazenado na memória do computador.

Esta série de números pode antes ser armazenada por um objeto, nomeado `dados`, posteriormente, se usa a função `mean()` com este objeto `dados`. O resultado da função `mean`, exibido no *Console*, será recebido por outro objeto `media_dados` que será colocado na memória do computador.

```
dados <- c(3, 5, 7, 9, 6, 7)
media_dados <- mean(dados)
media_dados
```

```
## [1] 6.166667
```

As funções podem ser criadas pelo pesquisador, de acordo com as suas necessidades. Entretanto, na maioria das vezes, elas são encontradas prontas, fazendo parte de um pacote. Pacotes contêm muitas funções que para serem executadas necessitam que o pacote esteja instalado e carregado. As funções para exercerem a sua ação devem receber dentro delas (entre parênteses) os argumentos que elas exigem. Os argumentos de uma função são sempre separados por vírgulas.

Para se saber quais argumentos necessários para uma determinada função basta consultar a ajuda, onde se encontrará a documentação da mesma. Para isso basta digitar no *Console*, no caso da função `mean()`, `help(mean)` ou `?mean`:

```
help(mean)
```

O resultado deste comando aparecerá na aba *Help*, na parte inferior, à direita (Figura 28):

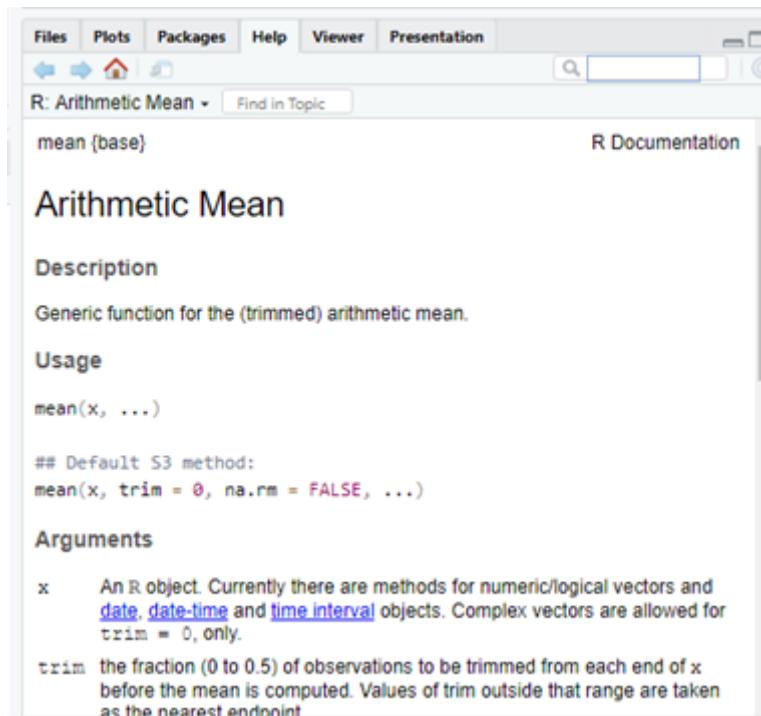


Figura 28: Ajuda para Média Aritmética.

Os principais argumentos da função `mean()` são:

- `x` → vetor numérico

- **trim** → fração das observações (varia de 0 a 0,5) extraída de cada extremidade de x para calcular a média aparada
- **na.rm** → valor lógico (TRUE ou FALSE) que indicam se os valores ausentes (NA) devem ser removidos antes que o cálculo continue

Este último argumento é muito importante quando, na sequência de valores existe algum não informado ou inexistente. No R, els são denominados de valores ausentes (*missing values*) e denotados por **NA**(*Not Available*).

Por exemplo, em uma coleta de uma série de valores, correspondentes ao peso de 15 recém-nascidos, havendo a “falta” de um dos registros, ao calcular a média com a função **mean()**, ela retornará NA.

```
pesoRN <- c (3340, 3345, 3750, 3650, 3220, 4070, NA, 3970, 3060, 3180,
              2865, 2815, 3245, 2051, 2630)
mean (pesoRN)
```

```
## [1] NA
```

Colocando o argumento **na.rm = TRUE**, para remover os valores faltantes, a função retornará a média aritmética sem este valor:

```
mean (pesoRN, na.rm = TRUE)
```

```
## [1] 3227.929
```

4.8.1 Criando funções

No R, é possível criar funções pessoais que podem simplificar um código e, eventualmente, diminuir o tempo de execução das análises.

Fórmula geral

As funções têm uma fórmula geral:

```
nome_da_funcao <- function (x){transformar x}
```

Por exemplo, a área de um circulo é igual a $\pi \times raio^2$. Para calcular a área do círculo, pode-se criar uma função que faça este trabalho:

```
area.circ <- function(r){
  area <- pi*r^2
  return(area)
}
```

Ao executar essa função, é possível usá-la para calcular a área de um círculo, cujo raio é igual a 5 cm:

```
r = 5
area.circ(5)
```

```
## [1] 78.53982
```

Outros exemplos

O Indice de Massa Corporal é igual ao peso (kg) dividido pela $altura^2$, em metros. Uma função para fazer este cálculo é:

```
imc <- function(peso, altura){  
  res <- peso/altura^2  
  return(res)  
}
```

Logo, o IMC de um indivíduo que tenha 67 kg e 1,7 m é:

```
peso <- 67  
altura <- 1.70  
imc(67, 1.70)
```

```
## [1] 23.18339
```

Ativação de uma função criada

Para ativar uma função previamente criada, usa-se a função nativa `source ()`. O argumento desta função é o caminho (no exemplo, é o diretório do autor) onde se encontra a função buscada, por exemplo, a função `imc()` criada acima:

```
source('C:/Users/petro/Dropbox/Estatistica/Bioestatistica_usando_R/Funcoes/imc.R')
```

4.9 Classes

São os atributos de um objeto e o seu conhecimento é de suma importância. A partir do conhecimento do tipo de classe que as funções sabem o que extamente fazer com um objeto. Por exemplo, não é possível somar duas letras e se for feita a tentativa de somar “a” e “b”, O R retorna um erro: `Error in "a" + "b": non-numeric argument to binary operator.`

No R, os textos são escritos entre aspas simples ou duplas. As aspas servem para diferenciar nomes (objetos, funções, pacotes) de textos (letras e palavras). Os textos são muito comuns em variáveis categóricas e são popularmente chamados de *strings* ou *character*. Além desta classe, o R tem outras classes básicas que são a *numeric* e a *logical*. Um objeto de qualquer uma dessas classes é chamado de *objeto atômico*. Esse nome se deve ao fato de essas classes não se misturarem (54).

Para saber qual o tipo de classe que um objeto pertence, basta usar a função `class ()`.

```
idade <- c(3, 5, 7, 9, 6, 7)  
class (idade)  
  
## [1] "numeric"  
  
nome <- c("Pedro", "Maria", "Margarida", "Alice", "João", "Luís")  
class(nome)  
  
## [1] "character"
```

4.10 Vetores

Um **vetor** é uma variável com um ou mais valores do mesmo tipo. Por exemplo, o número de filhos em 10 famílias foi 4, 5, 3, 2, 2, 1, 2, 1, 3 e 2. O vetor nomeado de `n.filhos` é um objeto numérico de comprimento = 10. A maneira mais fácil de criar um vetor em R é concatenar (ligar) os 10 valores, usando a função `c()` assim:

```
n.filhos <- c(4, 5, 3, 2, 2, 1, 2, 1, 3, 2)
```

```
## [1] 4 5 3 2 2 1 2 1 3 2
```

Como os vetores são conjuntos *indexados*, pode-se dizer que cada valor dentro de um vetor tem uma **posição**. Essa posição é dada pela ordem em que os elementos foram colocados no momento em que o vetor foi criado. Isso nos permite acessar individualmente cada valor de um vetor (54).

Para acessar um determinado valor, basta colocar a posição do mesmo entre colchetes `[]`. Se há interesse em conhecer o número de filhos da quinta família, procede-se da seguinte forma:

```
n.filhos[5]
```

```
## [1] 2
```

Se houver tentativa de acessar um valor inexiste, o R retorna `NA`.

```
n.filhos[11]
```

```
## [1] NA
```

Se houver necessidade de excluir um dos elementos, basta colocar entre colchetes a posição do mesmo com sinal negativo. Por exemplo, para excluir o valor correspondente a sexta família, usa-se:

```
n.filhos[-6]
```

```
## [1] 4 5 3 2 2 2 1 3 2
```

Observa-se que o valor 1 foi excluído da série de elementos.

Quando são colocados elementos em um vetor que pertençam a classes diferentes, o R promove o que se denomina de **coerção**, pois o vetor pode ter apenas uma classe de objeto. Dessa forma, as classes mais fortes reprimem as mais fracas. Por exemplo, sempre que for misturado números e texto em um vetor, os números serão considerados como texto:

```
vetor <- c(12, 15, 4, 6, "A", "D")
```

```
## [1] "12" "15" "4"   "6"   "A"   "D"
```

Observe que, agora, todos os elementos do vetor passaram a ser textos.

4.10.1 Tipos de vetores

Dado um vetor, pode-se determinar seu tipo com `typeof()`, ou verificar se é um tipo específico com uma das funções: `is.character()`, `'is.double()'`, `is.integer()`, `is.logical()`:

```
n.filhos <- c(4, 5, 3, 2, 2, 1, 2, 1, 3, 2)
typeof(n.filhos)
```

```
## [1] "double"
```

```
is.numeric(n.filhos)
```

```
## [1] TRUE
```

As expressões do tipo *character* devem aparecer entre aspas duplas ou simples. Os números no R são geralmente tratados como objetos numéricos (números reais de dupla precisão). Mesmo números inteiros são tratados como numéricos. Para fazer um número inteiro ser tratado como objeto inteiro, deve-se utilizar a letra L após o número.

Os valores lógicos (ou booleanos) são TRUE ou FALSE. T ou F também são aceitos.

```
n.filhos <- c(4L, 5L, 3L, 2L, 2L, 1L, 2L, 1L, 3L, 2L)
typeof(n.filhos)
```

```
## [1] "integer"
```

```
is.numeric(n.filhos)
```

```
## [1] TRUE
```

```
is.double(n.filhos)
```

```
## [1] FALSE
```

```
nomes <- c('Maria', 'João', 'Manuel', 'Petronio', 'José')
typeof(nomes)
```

```
## [1] "character"
```

```
is.numeric(nomes)
```

```
## [1] FALSE
```

```
is.double(nomes)
```

```
## [1] FALSE
```

```
altura <- c(1.60, 1.78, 1.55, 1.67, 1.69)
typeof(altura)
```

```
## [1] "double"
```

```
is.numeric(altura)
```

```
## [1] TRUE
```

```
is.double(altura)
```

```
## [1] TRUE
```

4.11 Data frames

Data frame são objetos de dados genéricos de R, usados para armazenar os dados tabulares, onde os dados são organizados de maneira lógica em um formato de linha-e-coluna semelhante ao de uma planilha do Excel. O data frame é uma estrutura bidimensional. Estas dimensões podem ser encontradas com a função `dim()`. Os Data frames podem ser formados com objetos criados previamente, desde que tenham o mesmo comprimento (55).

Abaixo serão criadas algumas variáveis, todas relacionadas ao nascimento de 15 bebês:

```
id <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15)
pesoRN <- c (3340,3345,3750,3650,3220,4070,3380,3970,3060,3180,
             2865,2815,3245,2051,2630)
compRN <- c (50,48,52,48,50,51,50,51,47,47,47,47,49,51,50,44)
sexo <- c (2,2,2,1,1,2,1,1,2,2,1,1,2,1,1,2,1,1)
tipoParto <- c (1,1,2,1,2,2,1,2,1,1,1,2,1,1,1,2,1,1)
idadeMae <- c (40,19,26,19,32,24,27,20,21,19,23,36,21,23,23)
```

Tem-se um grupo de variáveis isoladas. Seria útil reuni-las em um só objeto, usando a função `data.frame()`. Este novo objeto receberá o nome de `dadosNeonatos`.

```
dadosNeonatos <- data.frame (id,
                               pesoRN,
                               compRN,
                               sexo,
                               tipoParto,
                               idadeMae)
```

Ao ser executado o comando retornará um novo objeto da classe `data.frame`:

```
class (dadosNeonatos)
```

```
## [1] "data.frame"
```

Havendo necessidade de acrescentar outra variável no banco de dados `dadosNeonatos`, por exemplo, os dados da ida ou não dos recém-nascidos para a UTI. Para isso, será atribuido a um vetor, contendo a situação dos 15 recém-nascidos, o nome de `utiNeo` e para relacioná-lo a uma coluna do dataframe `dadosNeonatos`, será usado o símbolo `$`, como mostrado abaixo⁵:

⁵A variável criada, `utiNeo`, possui dois níveis: 1 = sim; 2 = não, referente se o bebê foi ou não para a UTI.

```
dadosNeonatos$utiNeo <- c (2,2,2,2,1,2,1,2,2,2,2,1,2,2,2)
```

Para observar o novo banco de dados, pode-se usar a função `str()` do R base. Digitar no *R Script*:

```
str (dadosNeonatos)
```

```
## 'data.frame':    15 obs. of  7 variables:
## $ id        : num  1 2 3 4 5 6 7 8 9 10 ...
## $ pesoRN    : num  3340 3345 3750 3650 3220 ...
## $ compRN    : num  50 48 52 48 50 51 50 51 47 47 ...
## $ sexo       : num  2 2 2 1 1 1 2 1 1 1 ...
## $ tipoParto : num  1 1 2 1 2 2 1 2 1 1 ...
## $ idadeMae  : num  40 19 26 19 32 24 27 20 21 19 ...
## $ utiNeo    : num  2 2 2 2 1 2 1 2 2 2 ...
```

Observando a saída da função, verifica-se que o dataframe contém 15 linhas e 6 colunas e que todas as variáveis estão como variáveis *numéricas*, mas as variáveis `sexo`, `tipoParto` são variáveis *categóricas*, bem como a variável `utiNeo`, acrescentada depois. Há necessidade de fazer uma transformação dessas variáveis.

4.12 Fatores

Os fatores, no R, são usados para trabalhar com variáveis categóricas. São variáveis usadas para categorizar e armazenar os dados, tendo um número limitado de valores diferentes.

Um fator armazena os dados como um vetor de valores inteiros. O fator em R também é conhecido como uma variável categórica que armazena valores de dados de *string* e inteiros como níveis. O fator é usado principalmente em modelagem estatística e análise exploratória de dados com R (56).

4.12.1 Criando fatores

No data frame `dadosNeonatos`, criado anteriormente, contém três variáveis (`sexo`, `tipoParto` e `utiNeo`) que estão como variáveis numéricas. É possível, desta forma, realizar operações aritméticas com elas. Isto, obviamente, seria um absurdo. Assim, é necessário transformá-las em fatores. Para isso, é usada a função `factor()`, nativa do R. Os principais argumentos desta função são:

- `x` → vetor numérico
- `levels` → vetor opcional dos valores que `x` pode assumir
- `labels` → vetor de caracteres dos rótulos para os níveis, na mesma ordem
- `ordered` → vetor lógico (TRUE ou FALSE). Se TRUE, os níveis dos fatores são assumidos como ordenados

No exemplo, as variáveis não têm uma ordem lógica, então, o argumento `ordered` não será usado.

```
dadosNeonatos$utiNeo <- factor (dadosNeonatos$utiNeo,
                                    levels = c(1,2),
                                    labels = c('sim','não'))
dadosNeonatos$tipoParto <- factor(dadosNeonatos$tipoParto,
                                    levels = c(1,2),
                                    labels = c("normal","cesareo"))
dadosNeonatos$sexo <- factor (dadosNeonatos$sexo,
                                levels = c(1,2),
                                labels = c("M","F"))
```

Após a transformação, executa-se novamente a função `str()` para ver como ficou o dataframe:

```
str(dadosNeonatos)
```

```
## 'data.frame': 15 obs. of 7 variables:  
## $ id : num 1 2 3 4 5 6 7 8 9 10 ...  
## $ pesoRN : num 3340 3345 3750 3650 3220 ...  
## $ compRN : num 50 48 52 48 50 51 50 51 47 47 ...  
## $ sexo : Factor w/ 2 levels "M","F": 2 2 2 1 1 1 2 1 1 1 ...  
## $ tipoParto: Factor w/ 2 levels "normal","cesareo": 1 1 2 1 2 2 1 2 1 1 ...  
## $ idadeMae : num 40 19 26 19 32 24 27 20 21 19 ...  
## $ utiNeo : Factor w/ 2 levels "sim","não": 2 2 2 2 1 2 1 2 2 2 ...
```

Agora, as três variáveis passaram a ser fatores e as outras mantiveram-se numéricas.

Desta forma, é possível trabalhar com ela fazendo, por exemplo, uma contagem da frequência do tipo de parto, usando a função `table()`:

```
table(dadosNeonatos$tipoParto)
```

```
##  
## normal cesareo  
##      10      5
```

Ou seja, aproximadamente 70% dos partos desta amostra são normais.

4.12.2 Salvando o dataframe criado

O data frame, criado e modificado anteriormente, pode ser salvo para uso posterior no diretório de trabalho.

Para isso existe a função `save()`, fornecendo como argumentos o data frame a ser salvo e o nome do arquivo (`file =`) entre aspas. Por convenção, esta função salva com a extensão `.RData` que deve ser digitada, pois o R não a adiciona automaticamente.

```
save(dadosNeonatos, file = "dadosNeonatos.RData")
```

Este comando colocará o arquivo no diretório de trabalho em uso. Portanto, se o objetivo é salvar em outro local, deve ser informado ao R qual o novo diretório.

Para carregar o objeto salvo anteriormente com o comando `save()`, usa-se a função `load()`. Se o arquivo a ser lido não estiver no diretório de trabalho da sessão, há necessidade de especificar o caminho até o arquivo:

```
load("dadosNeonatos.RData")
```

Ou, indicando o diretório onde está o arquivo:

```
load("C:/Users/petro/Dropbox/Estatistica/Meus_Livros/Bioestatistica_R/Book/dadosNeonatos.RData")
```

É possível salvar em outro tipo de extensão como Excel (`.xlsx`), Valores Separados por Vírgula (`.csv`), etc. O procedimento é o mesmo, mudando a função. Para salvar em uma extensão `.xlsx`, utiliza-se a função `write_xlsx()` do pacote `writexl` (57):

```
writexl::write_xlsx(dadosNeonatos, "dadosNeonatos.xlsx")
```

Para salvar com a extensão .csv, usar a função `write.csv()` ou `write.csv2()` que faz parte do pacote `utils`, incluído no R base. A primeira função, usa `.` para a separação dos decimais e `,` para separar as variáveis; a segunda função usa `,` para os decimais e `;` para separar as variáveis, convenção do Excel para algumas localidades, como o Brasil (58). Portanto, uma maneira de salvar o arquivo é:

```
write.csv2 (dadosNeonatos, "dadosNeonatos.csv")
```

5 Manipulando os dados no R Studio

5.1 Importando dados de outros softwares

Foi visto. quando estudou-se os dataframe, que é possível inserir dados diretamente no R. Entretanto, se o conjunto de dados for muito extenso, torna-se complicado. Desta forma, é melhor importar os dados de outro software, como o Excel, SPSS, etc. A recomendação é que se construa o banco de dados, por exemplo, no Excel, e depois exporte o arquivo em um formato que o R reconheça – .xlsx, .csv, .sav, por exemplo.

5.1.1 Importando dados de um arquivo CSV

O formato CSV significa *Comma Separated Values*, ou seja, é um arquivo de valores separados por vírgula. Esse formato de armazenamento é simples e agrupa informações de arquivos de texto em planilhas. É possível gerar um arquivo .csv, a partir de uma planilha do Excel, usando o menu **salvar como** e escolher CSV.

As funções `read.csv()` e `read.csv2()`, incluídas no R base, podem ser utilizadas para importar arquivos CSV. Existe uma pequena diferença entre elas. Dois argumentos dessas funções têm padrão diferentes em cada uma. São eles: `sep` (separador de colunas) e `dec` (separador de decimais). Em `read.csv()`, o padrão é `sep = ","` e `dec = "."` e em `read.csv2()` o padrão é `sep = ";"` e `dec = ","`. Portanto, quando se importa um arquivo .csv, é importante saber qual a sua estrutura. Verificar se os decimais estão separados por *ponto* ou por *vírgula* e se as colunas (variáveis), por *vírgula* ou *ponto e vírgula*.

Quando se usa o `read.csv()` há necessidade de informar o separador e o decimal, pois senão ele usará o padrão inglês e o arquivo não será lido. Já com `read.csv2()`, que o usa o padrão brasileiro, não há necessidade de informar ao R qual o separador de colunas e nem o separador dos decimais.

Além disso, é necessário saber em que diretório do computador está o arquivo para informar ao comando. Recomenda-se colocar o arquivo na pasta do diretório de trabalho, pois assim basta apenas colocar o nome do arquivo na função de leitura dos dados. Caso contrário, tem-se que se usar todo o caminho.

Como exemplo, será importado o arquivo `dadosNeonatos.csv` que se encontra no diretório de trabalho do autor, salvo anteriormente. A estrutura deste arquivo mostra que as colunas estão separadas por ponto-e-vírgula e, portanto, a leitura dos dados será feita com a função `read.csv2()` sem informar o diretório completo e colocando os dados em um objeto de nome `neonatos`:

```
neonatos <- read.csv2("dadosNeonatos.csv")  
  
str(neonatos)  
  
## 'data.frame': 15 obs. of 7 variables:  
## $ id : int 1 2 3 4 5 6 7 8 9 10 ...  
## $ pesoRN : int 3340 3345 3750 3650 3220 4070 3380 3970 3060 3180 ...  
## $ compRN : int 50 48 52 48 50 51 50 51 47 47 ...  
## $ sexo : chr "F" "F" "F" "M" ...  
## $ tipoParto: chr "normal" "normal" "cesareo" "normal" ...  
## $ idadeMae : int 40 19 26 19 32 24 27 20 21 19 ...  
## $ utiNeo : chr "n\xe3o" "n\xe3o" "n\xe3o" "n\xe3o" ...
```

Recentemente, foi desenvolvido o pacote `readr`, incluído no conjunto de pacotes `tidyverse`(59), para lidar rapidamente com a leitura de grandes arquivos. O pacote fornece substituições para funções como `read.csv()`. As funções `read_csv()` e `read_csv2()` oferecidas pelo `readr` são análogas às do R base. Entretanto, são muito mais rápidas e fornecem mais recursos, como um método compacto para especificar tipos de coluna.

Uma leitura típica para `read_csv2()` terá a seguinte aparência. Será criado um outro objeto de nome `recemNascidos` apenas para facilitar, didaticamente, ele é exatamente igual ao `neonatos`:

```

library(readr)
recemNascidos <- read_csv2("dadosNeonatos.csv")

## i Using ',',',' as decimal and ',.' as grouping mark. Use `read_delim()` for more control.

## Rows: 15 Columns: 7
## -- Column specification -----
## Delimiter: ;
## chr (3): sexo, tipoParto, utiNeo
## dbl (4): id, pesoRN, compRN, idadeMae
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

str(recemNascidos)

## #> #> spc_tbl_ [15 x 7] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## #>   $ id      : num [1:15] 1 2 3 4 5 6 7 8 9 10 ...
## #>   $ pesoRN  : num [1:15] 3340 3345 3750 3650 3220 ...
## #>   $ compRN  : num [1:15] 50 48 52 48 50 51 50 51 47 47 ...
## #>   $ sexo     : chr [1:15] "F" "F" "F" "M" ...
## #>   $ tipoParto: chr [1:15] "normal" "normal" "cesareo" "normal" ...
## #>   $ idadeMae : num [1:15] 40 19 26 19 32 24 27 20 21 19 ...
## #>   $ utiNeo   : chr [1:15] "n\xe3o" "n\xe3o" "n\xe3o" "n\xe3o" ...
## #> - attr(*, "spec")=
## #>   .. cols(
## #>     .. id = col_double(),
## #>     .. pesoRN = col_double(),
## #>     .. compRN = col_double(),
## #>     .. sexo = col_character(),
## #>     .. tipoParto = col_character(),
## #>     .. idadeMae = col_double(),
## #>     .. utiNeo = col_character()
## #>   .. )
## #> - attr(*, "problems")=<externalptr>

```

5.1.2 Importando um arquivo do Excel

O pacote `readxl`, pertencente ao conjunto de pacotes do `tidyverse`, facilita a obtenção de dados do Excel para o R, através da função `read_excel()`. Esta função tem o argumento `sheet` = , que deve ser usado indicando o número ou o nome da planilha, colocado entre aspas. Este argumento é importante se houver mais de uma planilha, caso contrário, ele é opcional. Para saber os outros argumentos da função, coloque o cursor dentro da função e aperte a tecla Tab (Figura 29). Isto abrirá mostrando um menu com os argumentos:

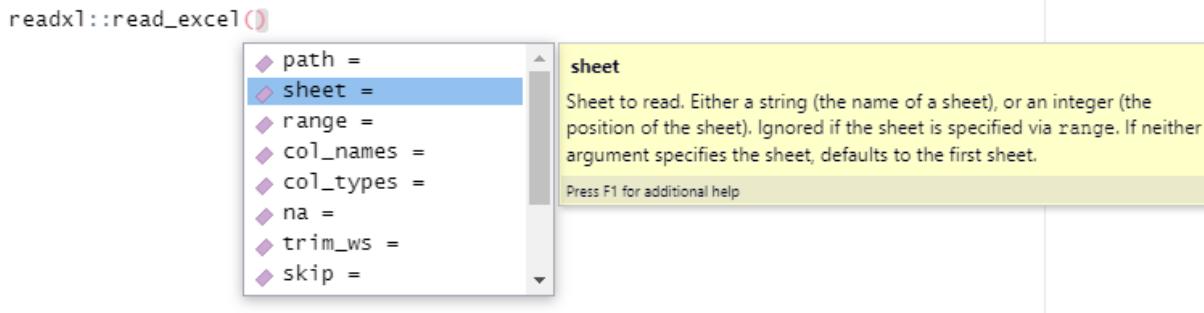


Figura 29: Argumentos da função para importar arquivos xlsx

Será feita a leitura do mesmo arquivo, usado na leitura de dados `csv`, apenas será atribuído a um objeto com outro nome (`recemNatos`):

```

library(readxl)

recemNatos <- read_excel("dadosNeonatos.xlsx")

str(recemNatos)

## # tibble [15 x 7] (S3: tbl_df/tbl/data.frame)
## $ id      : num [1:15] 1 2 3 4 5 6 7 8 9 10 ...
## $ pesoRN  : num [1:15] 3340 3345 3750 3650 3220 ...
## $ compRN  : num [1:15] 50 48 52 48 50 51 50 51 47 47 ...
## $ sexo    : chr [1:15] "F" "F" "F" "M" ...
## $ tipoParto: chr [1:15] "normal" "normal" "cesareo" "normal" ...
## $ idadeMae : num [1:15] 40 19 26 19 32 24 27 20 21 19 ...
## $ utiNeo   : chr [1:15] "não" "não" "não" "não" ...

```

5.1.3 Importando arquivos com o RStudio

O RStudio permite importar arquivos sem a necessidade de digitar comandos, que, para alguns podem ser tediosos.

Na tela inicial do RStudio, à direita, na parte superior, clique na aba *Environment* e em *Import Dataset*. Esta ação abre um menu que permite importar arquivos .csv, Excel, SPSS, etc.

Por exemplo, para importar o arquivo `dadosNeonatos.xlsx`, clicar em *From Excel...* Abre uma janela com uma caixa de diálogo. Clicar no botão *Browse...*, localizado em cima à direita, para buscar o arquivo `dadosNeonatos.xlsx`. Assim que o arquivo for aberto, ele mostra uma *preview* do arquivo e, em baixo, à direita mostra uma *preview* do código (Figura 30), igual ao digitado anteriormente, que cria um objeto denominado `dadosNeonatos`, nome do objeto escolhido pelo R, mas pode ser modificado na janela, à esquerda, *Import Option* em *Name*, onde pode-se digitar qualquer nome. Após encerrar as escolhas, clicar em *Import*. É um caminho diferente para fazer o mesmo. Este é um dos fascínios do R!

Figura 30: Importando arquivos do excel com o RStudio.

5.2 Dataframe e tibble

A maneira mais comum de armazenar dados no R é usar `data.frames` ou `tibble`.

`Tibble` é um novo tipo de data frame. É como se fosse um data frame mais moderno. Ele mantém muitos recursos importantes do data frame original, mas remove muitos dos recursos desatualizados.

Os `tibbles` são outro recurso incrível adicionado ao R por Hadley Wickham, através do `Tidyverse`, conjunto de pacotes que formam um conjunto básico de funções que facilitam a manipulação e representação gráfica dos dados (59). Para saber mais sobre tibble, veja vignette('tibbles').

A maioria dos pacotes do R usa dataframes tradicionais, entretanto é possível transformá-los para `tibble`, usando a função `as_tibble()`, incluída no pacote `tidyverse` (60). O único propósito deste pacote é simplificar o processo de criação de `tidy` `data`(dados organizados).

O conceito de `tidy data`, introduzido por Wickman (61), se refere à estrutura dos dados organizados de maneira que cada linha é uma observação, cada coluna representa variáveis e cada entrada nas células do dataframe são os valores.

A transformação de um data frame tradicional em um tibble, é um procedimento recomendável, em função da maior flexibilidade destes.

Como exemplo deste procedimento, será usado o famoso conjunto de dados da flor iris (62) que fornece as medidas em centímetros das variáveis comprimento e largura da sepala e comprimento e largura da pétala, respectivamente, para 50 flores de cada uma das 3 espécies de íris (*Iris setosa*, *versicolor* e *virginica*). Este conjunto de dados encontra-se no pacote `datasets` no R base. Para visualizar os dados, será usado a função `str()`, também do R base, que mostra a estrutura interna de um objeto:

```
str(iris)
```

```

## 'data.frame':   150 obs. of  5 variables:
## $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
## $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
## $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
## $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
## $ Species      : Factor w/ 3 levels "setosa","versicolor",...: 1 1 1 1 1 1 1 1 1 1 ...

```

Observa-se que é um conjunto de dados da classe `data.frame`, contendo 150 observações de 5 variáveis (colunas). Fazendo a coerção para um `tibble`, tem-se:

```

library(tidyr)
as_tibble(iris)

```

```

## # A tibble: 150 x 5
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
##       <dbl>       <dbl>       <dbl>       <dbl>   <fct>
## 1         5.1         3.5         1.4         0.2 setosa
## 2         4.9         3           1.4         0.2 setosa
## 3         4.7         3.2         1.3         0.2 setosa
## 4         4.6         3.1         1.5         0.2 setosa
## 5         5           3.6         1.4         0.2 setosa
## 6         5.4         3.9         1.7         0.4 setosa
## 7         4.6         3.4         1.4         0.3 setosa
## 8         5           3.4         1.5         0.2 setosa
## 9         4.4         2.9         1.4         0.2 setosa
## 10        4.9         3.1         1.5         0.1 setosa
## # ... with 140 more rows

```

5.3 Pacote `dplyr`

O pacote `dplyr` é comumente usado para limpar e trabalhar com dados (63). No nível mais básico, as funções do pacote referem-se a “verbos” de manipulação de dados, como `select`, `filter`, `mutate`, `arrange`, `summarize`, entre outros, que permitem encadear várias etapas em algumas linhas de código, como será visto adiante.

O pacote `dplyr` é adequado para trabalhar com um único conjunto de dados, bem como para obter resultados complexos em grandes conjuntos de dados. As funções `dplyr` são processadas mais rápido do que as funções R base.

Para trabalhar na manipulação dos dados serão usados alguns pacotes, já mencionados anteriormente, `readxl(64)` e `dplyr`, e o conjunto de dados `dadosMater.xlsx`. Para obter estes dados, clique [aqui](#) e faça o download para o seu diretório de trabalho.

```

library(readxl)
library(dplyr)

```

```

mater <- read_excel("dadosMater.xlsx")

```

A função `read_excel()` carrega o arquivo e o coloca em objeto que foi, arbitrariamente, chamado de `mater`⁶.

⁶ATENÇÃO: O comando para carregar o conjunto de dados somente funciona, sem colocar o caminho completo, se o trabalho está sendo realizado no diretório de trabalho.

```
as_tibble(mater)
```

```
## # A tibble: 1,368 x 30
##       id idadeMae altura peso ganhoP~1 anosEst cor eCivil renda fumo quant~2
##   <dbl>    <dbl>  <dbl> <dbl>    <dbl>    <dbl> <dbl> <dbl> <dbl> <dbl>    <dbl>
## 1     1      42   1.65  69.9     3.9     3     2     1   1.45     2     0
## 2     2      29   1.66   78     16.5    11     1     2   2.41     2     0
## 3     3      19   1.72   81      5     9     2     1   1.93     2     0
## 4     4      31   1.55   74     43      5     2     2   1.45     2     0
## 5     5      34   1.6    60     15      7     2     2   0.48     2     0
## 6     6      29   1.5    60    11.4     8     2     2   0.96     1    10
## 7     7      30   1.54  75.5    10.5     4     1     2   1.2      1    20
## 8     8      34   1.63   61      9     6     1     2   2.41     2     0
## 9     9      17   1.68   57     15     10     1     2   2.17     2     0
## 10    10     32   1.5    70    11.4     1     2     2   0.72     2     0
## # ... with 1,358 more rows, 19 more variables: prenatal <dbl>, para <dbl>,
## #   droga <dbl>, ig <dbl>, tipoParto <dbl>, pesoPla <dbl>, sexo <dbl>,
## #   pesoRN <dbl>, compRN <dbl>, pcRN <dbl>, apgar1 <dbl>, apgar5 <dbl>,
## #   utiNeo <dbl>, obito <dbl>, hiv <dbl>, sifilis <dbl>, rubeola <dbl>,
## #   toxo <dbl>, infCong <dbl>, and abbreviated variable names 1: ganhoPeso,
## #   2: quantFumo
```

Por padrão, a função retorna as dez primeiras linhas. Além disso, colunas que não couberem na largura da tela serão omitidas. Também são apresentadas a dimensão da tabela e as classes de cada coluna. Observa-se que ele tem 1368 linhas (observações) e 30 colunas (variáveis). Além disso, verifica-se que todas as variáveis estão como numéricas (`dbl`) e, certamente, algumas, dependendo do objetivo na análise, precisarão ser transformadas.

O significado de cada uma das variáveis do arquivo `dadosMater.xlsx`⁷ são mostrados abaixo.

- **id** → identificação do participante
- **idadeMae** → idade da parturiente em anos
- **altura** → altura da parturiente em metros
- **peso** → peso da parturiente em kg
- **ganhoPeso** → aumento de peso durante a gestação
- **anosEst** → anos de estudo completos
- **cor** → cor declarada pela parturiente: 1 = branca; 2 = não branca
- **eCivil** → estado civil: 1 = solteira; 2 = casada ou companheira
- **renda** → renda familiar em salários mínimos
- **fumo** → tabagismo: 1 = sim; 2 = não
- **quantFumo** → quantidade de cigarros fumados diariamente

⁷Conjunto de dados coletados na maternidade-escola do Hospital Geral de Caxias do Sul

- **prenatal** → realizou pelo menos 6 consultas no pré-natal? 1 = sim; 2 = não
 - **para** → número de filhos paridos
 - **droga** → drogadição? 1 = sim; 2 = não
 - **ig** → idade gestacional em semanas
 - **tipoParto** → tipo de parto: 1 = normal; 2 = cesareana
 - **pesoPla** → peso da placenta em gramas
 - **sexo** → sexo do recém-nascido (RN): 1 = masc; 2 = fem
 - **pesoRN** → peso do RN em gramas
 - **compRN** → comprimento do RN em cm
 - **pcRN** → perímetrocefálico dorecém-nascido em cm
 - **apgar1** → escore de Apgar no primeiro minuto
 - **apgar5** → escore de Apgar no quinto minuto
 - **utiNeo** → RN necessitou de terapia intesiva? 1 = sim; 2 = não
 - **obito** → obito no período neonatal? 1 = sim; 2 = não
 - **hiv** → parturiente portadora de HIV? 1 = sim; 2 = não
 - **sifilis** → parturiente portadora de sífilis? 1 = sim; 2 = não
 - **rubeola** → parturiente portadora de rubéola? 1 = sim; 2 = não
 - **toxo** → parturiente portadora de toxoplasmose? 1 = sim; 2 = não
 - **infCong** → parturiente portadora de alguma infecção congênita? 1 = sim; 2 = não

5.3.1 Função select()

A função `select ()` é usada para escolher com quais colunas (variáveis) entrarão na análise. Ela recebe os nomes das colunas como argumentos e cria um novo banco de dados usando as colunas selecionadas. A função `select ()` pode ser combinada com outras funções, como `filter ()`.

Por exemplo, um novo banco de dados será criado (`mater1`), contendo as mesmas 1368 linhas, mas apenas com as variáveis `idadeMae`, `altura`, `peso`, `anosEst`, `renda`, `ig`, `fumo`, `pesoRN`, `sexo`. Consulte a ajuda (`?select()`) para obter maiores informações em relação aos argumentos da função:

```
mater1 <- select(mater, idadeMae, altura, peso, anosEst, renda, ig, tipoParto, fumo, pesoRN, sexo)
```

Para visualizar este novo banco de dados, pode-se usar a função `str()`:

```
str(mater1)
```

```
## $ idadeMae : num [1:1368] 42 29 19 31 34 29 30 34 17 32 ...
## $ altura   : num [1:1368] 1.65 1.66 1.72 1.55 1.6 1.5 1.54 1.63 1.68 1.5 ...
## $ peso     : num [1:1368] 69.9 78 81 74 60 60 75.5 61 57 70 ...
## $ anosEst  : num [1:1368] 3 11 9 5 7 8 4 6 10 1 ...
## $ renda    : num [1:1368] 1.45 2.41 1.93 1.45 0.48 0.96 1.2 2.41 2.17 0.72 ...
## $ ig       : num [1:1368] 29 33 33 33 33 33 33 33 34 34 ...
## $ tipoParto: num [1:1368] 2 2 1 1 2 1 2 1 1 2 ...
## $ fumo     : num [1:1368] 2 2 2 2 2 1 1 2 2 2 ...
## $ pesoRN   : num [1:1368] 1035 2300 1580 1840 2475 ...
## $ sexo     : num [1:1368] 2 2 2 2 2 2 2 2 2 2 ...
```

Como mostrado anteriormente, muitas variáveis numéricas do `mater`, na realidade, são fatores e necessitam de serem modificadas. Entretanto, das selecionadas, para constituir o novo banco de dados, apenas `tipoParto`, `fumo` e `sexo` necessitam serem transformadas para fator:

```
mater1$tipoParto <- factor(mater1$tipoParto,
                            levels = c(1,2),
                            labels = c("normal","cesareo"))

mater1$fumo <- factor (mater1$fumo,
                        levels = c(1,2),
                        labels = c('sim','não'))

mater1$sexo <- factor (mater1$sexo,
                        levels = c(1,2),
                        labels = c("masc","fem"))
```

Usando, de novo, a função `str()`, é possível observar a transformação:

Se houver necessidade de se excluir alguma variável (coluna), basta colocar o sinal de subtração (-) antes do nome da variável:

```
mater2 <- select(mater1, -altura)
```

```
str(mater2)
```

```
## tibble [1,368 x 9] (S3: tbl_df/tbl/data.frame)
## $ idadeMae : num [1:1368] 42 29 19 31 34 29 30 34 17 32 ...
```

```

## $ peso      : num [1:1368] 69.9 78 81 74 60 60 75.5 61 57 70 ...
## $ anosEst   : num [1:1368] 3 11 9 5 7 8 4 6 10 1 ...
## $ renda     : num [1:1368] 1.45 2.41 1.93 1.45 0.48 0.96 1.2 2.41 2.17 0.72 ...
## $ ig        : num [1:1368] 29 33 33 33 33 33 33 34 34 ...
## $ tipoParto: Factor w/ 2 levels "normal","cesareo": 2 2 1 1 2 1 2 1 1 2 ...
## $ fumo      : Factor w/ 2 levels "sim","não": 2 2 2 2 2 1 1 2 2 2 ...
## $ pesoRN    : num [1:1368] 1035 2300 1580 1840 2475 ...
## $ sexo      : Factor w/ 2 levels "masc","fem": 2 2 2 2 2 2 2 2 2 2 ...

```

5.3.2 Função filter()

A função `filter()` é usada para criar um subconjunto de dados que obedecam determinadas condições lógicas: & (e), | (ou) e ! (não). Por exemplo:

- `y & !x` → seleciona `y` e não `x`
- `x & !y` → seleciona `x` e não `y`
- `x | !x` → seleciona `x` ou `y`
- `x & !x` → seleciona `x` e `y`

Um recém-nascido é dito a termo quando a duração da gestação é igual a 37 a 42 semanas incompletas. Se quisermos extrair do banco de dados `mater2` os recém-nascidos a termo, pode-se usar a função `filter()`:

```
mater3 <- filter (mater2, ig>=37 & ig<42)
```

Para exibir o resultado, execute a função `str()`:

```
str(mater3)
```

```

## tibble [1,085 x 9] (S3: tbl_df/tbl/data.frame)
## $ idadeMae : num [1:1085] 28 31 27 28 18 28 22 28 25 14 ...
## $ peso      : num [1:1085] 48.5 65 60 47 65.5 72 65 74 70 56.7 ...
## $ anosEst   : num [1:1085] 6 5 8 8 7 11 6 5 9 6 ...
## $ renda     : num [1:1085] 3.13 0.72 2.41 1.69 1.93 1.92 2.65 2.53 0.48 1.92 ...
## $ ig        : num [1:1085] 37 37 37 38 39 39 39 39 39 39 ...
## $ tipoParto: Factor w/ 2 levels "normal","cesareo": 1 2 2 1 1 2 2 1 1 1 ...
## $ fumo      : Factor w/ 2 levels "sim","não": 2 2 1 2 1 1 2 2 2 2 ...
## $ pesoRN    : num [1:1085] 3285 3100 3100 2800 3270 ...
## $ sexo      : Factor w/ 2 levels "masc","fem": 1 1 1 1 1 1 1 1 1 1 ...

```

Observe que, agora, o conjunto de dados `mater3` tem 1085 linhas, número de recém-nascidos a termo do banco de dados original `mater` (1368). Logo, os recém nascidos a termo correspondem a 79.3% dos nascimentos, nesta maternidade.

Outro exemplo

Para selecionar apenas os meninos, codificados como "`masc`", procede-se da seguinte maneira⁸:

```
meninos <- filter (mater1, sexo == 'masc')
```

⁸Lembrar que o sinal de igualdade, no R, é duplo =

```

str(meninos)

## # tibble [731 x 10] (S3: tbl_df/tbl/data.frame)
## $ idadeMae : num [1:731] 19 27 28 31 27 28 18 28 22 28 ...
## $ altura   : num [1:731] 1.53 1.75 1.5 1.55 1.6 1.58 1.76 1.63 1.54 1.55 ...
## $ peso     : num [1:731] 70 62 48.5 65 60 47 65.5 72 65 74 ...
## $ anosEst  : num [1:731] 7 11 6 5 8 8 7 11 6 5 ...
## $ renda    : num [1:731] 0.92 2.41 3.13 0.72 2.41 1.69 1.93 1.92 2.65 2.53 ...
## $ ig       : num [1:731] 36 36 37 37 37 38 39 39 39 39 ...
## $ tipoParto: Factor w/ 2 levels "normal","cesareo": 2 2 1 2 2 1 1 2 2 1 ...
## $ fumo     : Factor w/ 2 levels "sim","não": 2 2 2 2 1 2 1 1 2 2 ...
## $ pesoRN   : num [1:731] 2160 2800 3285 3100 3100 ...
## $ sexo     : Factor w/ 2 levels "masc","fem": 1 1 1 1 1 1 1 1 1 1 ...

```

O banco de dados `meninos` é constituídos por 731 meninos. Isto representa 53.4% dos nascimentos.

Uma outra maneira de se fazer o mesmo é usar a função `grepl()`, dentro da função `filter()`. Ela é usada para pesquisar a correspondência de padrões. No código a seguir, pesquisa-se os registros em que a variável `sexo` contém "fem", correspondentes às meninas.

```
meninas <- filter (mater1, grepl("fem", sexo))
```

```
str(meninas)
```

```

## # tibble [637 x 10] (S3: tbl_df/tbl/data.frame)
## $ idadeMae : num [1:637] 42 29 19 31 34 29 30 34 17 32 ...
## $ altura   : num [1:637] 1.65 1.66 1.72 1.55 1.6 1.5 1.54 1.63 1.68 1.5 ...
## $ peso     : num [1:637] 69.9 78 81 74 60 60 75.5 61 57 70 ...
## $ anosEst  : num [1:637] 3 11 9 5 7 8 4 6 10 1 ...
## $ renda    : num [1:637] 1.45 2.41 1.93 1.45 0.48 0.96 1.2 2.41 2.17 0.72 ...
## $ ig       : num [1:637] 29 33 33 33 33 33 33 34 34 34 ...
## $ tipoParto: Factor w/ 2 levels "normal","cesareo": 2 2 1 1 2 1 2 1 1 2 ...
## $ fumo     : Factor w/ 2 levels "sim","não": 2 2 2 2 2 1 1 2 2 2 ...
## $ pesoRN   : num [1:637] 1035 2300 1580 1840 2475 ...
## $ sexo     : Factor w/ 2 levels "masc","fem": 2 2 2 2 2 2 2 2 2 2 ...

```

5.3.3 Função `mutate()`

Esta função tem a finalidade de computar ou anexar uma ou mais colunas (variáveis) novas.

O Índice de Massa Corporal (IMC) é igual a

$$IMC = \frac{peso}{altura^2}$$

Será acrescentado a variável `imc`, no banco de dados `mater1`, usando a função `mutate()`:

```
mater1 <- mutate(mater1, imc = peso/altura^2)
```

Para ver esta variável presente no banco de dados, executar:

```

str (mater1)

## # tibble [1,368 x 11] (S3: tbl_df/tbl/data.frame)
## $ idadeMae : num [1:1368] 42 29 19 31 34 29 30 34 17 32 ...
## $ altura   : num [1:1368] 1.65 1.66 1.72 1.55 1.6 1.5 1.54 1.63 1.68 1.5 ...
## $ peso     : num [1:1368] 69.9 78 81 74 60 60 75.5 61 57 70 ...
## $ anosEst  : num [1:1368] 3 11 9 5 7 8 4 6 10 1 ...
## $ renda    : num [1:1368] 1.45 2.41 1.93 1.45 0.48 0.96 1.2 2.41 2.17 0.72 ...
## $ ig       : num [1:1368] 29 33 33 33 33 33 33 34 34 ...
## $ tipoParto: Factor w/ 2 levels "normal","cesareo": 2 2 1 1 2 1 2 1 1 2 ...
## $ fumo     : Factor w/ 2 levels "sim","não": 2 2 2 2 2 1 1 2 2 2 ...
## $ pesoRN   : num [1:1368] 1035 2300 1580 1840 2475 ...
## $ sexo     : Factor w/ 2 levels "masc","fem": 2 2 2 2 2 2 2 2 2 2 ...
## $ imc      : num [1:1368] 25.7 28.3 27.4 30.8 23.4 ...

```

Lembrar que este banco de dados `mater5` é igual ao `mater1`, subconjunto do banco de dados original `mater`, apenas com acréscimo da variável `imc`.

5.3.4 Função `sample_n()`

Função usada para selecionar de forma aleatória linhas de um dataframe. Sempre consulte a ajuda (`?sample_n`) para obter informações das funções. Os seus argumentos básicos são:

- `tbl` → dataframe
- `size` → número de linhas para selecionar
- `replace` → amostra com ou sem reposição?. Padrão = FALSE

Uma amostra de 20 neonatos selecionados do banco de dados `meninos` pode ser selecionada do seguinte modo:

```
meninos1 <- sample_n(meninos, 20)
```

Usando a função `str()`, verifica-se a estrutura deste pequeno conjunto de dados que pode ser considerado uma miniatura do original (2.7%).

```
str(meninos1)
```

```

## # tibble [20 x 10] (S3: tbl_df/tbl/data.frame)
## $ idadeMae : num [1:20] 36 29 22 24 18 26 37 23 32 31 ...
## $ altura   : num [1:20] 1.6 1.63 1.6 1.56 1.68 1.58 1.65 1.65 1.76 ...
## $ peso     : num [1:20] 68.5 66 55 61 50 70 70 56 60 77 ...
## $ anosEst  : num [1:20] 5 3 8 9 8 11 5 8 8 8 ...
## $ renda    : num [1:20] 4.82 2.07 1.92 2.41 1.69 3.61 2.89 1.57 0.96 1.69 ...
## $ ig       : num [1:20] 28 40 40 40 38 36 40 41 34 39 ...
## $ tipoParto: Factor w/ 2 levels "normal","cesareo": 2 1 2 1 2 1 1 1 1 2 ...
## $ fumo     : Factor w/ 2 levels "sim","não": 2 2 2 2 2 2 2 2 1 2 ...
## $ pesoRN   : num [1:20] 1905 3340 3150 3160 3115 ...
## $ sexo     : Factor w/ 2 levels "masc","fem": 1 1 1 1 1 1 1 1 1 1 ...

```

Uma outra função semelhante a esta é `sample_frac()`. Ela usa os mesmos argumentos que a `sample_n()`, modificando o argumento `size`, onde se informa a fração desejada até 1 (100%). Por exemplo, para se ter uma amostra de tamanho semelhante a anterior, há necessidade de selecionar, aproximadamente, uma fração de 0.027 da amostra.

```

meninos2 <- sample_frac(meninos, 0.027)

str(meninos2)

## # tibble [20 x 10] (S3: tbl_df/tbl/data.frame)
## $ idadeMae : num [1:20] 34 40 32 28 19 22 22 35 26 26 ...
## $ altura   : num [1:20] 1.57 1.7 1.66 1.54 1.64 1.55 1.65 1.5 1.65 1.52 ...
## $ peso     : num [1:20] 58 47.5 75 77 110 53 65 49 90 54 ...
## $ anosEst  : num [1:20] 11 4 5 11 6 10 6 2 11 7 ...
## $ renda    : num [1:20] 1.69 1.45 1.2 4.82 2.41 2.41 1.93 1.2 0.96 1.93 ...
## $ ig       : num [1:20] 40 39 41 41 40 36 40 30 39 39 ...
## $ tipoParto: Factor w/ 2 levels "normal","cesareo": 1 2 1 1 1 1 2 1 2 2 ...
## $ fumo     : Factor w/ 2 levels "sim","não": 2 2 2 2 2 2 1 1 2 1 ...
## $ pesoRN   : num [1:20] 3420 3350 4370 3645 3040 ...
## $ sexo     : Factor w/ 2 levels "masc","fem": 1 1 1 1 1 1 1 1 1 1 ...

```

É importante mencionar que toda vez que estas funções forem executadas elas irão gerar amostras diferentes. Então, por exemplo, não devemos esperar que a média dos pesos dos recém-nascidos de amostras diferentes sejam iguais. No capítulo sobre Distribuições Amostrais, este assunto voltará à cena.

As funções `sample_n()` e `sample_frac()` estão com os dias contados, pois foram substituídas por `slice_sample()` do conjunto de funções que acompanham a função `slice()`

5.3.5 Função `slice()`

Esta função é usada para selecionar um subconjunto linhas com base em seus locais inteiros. Permite selecionar, remover e duplicar linhas. Para os exemplos, será usado o conjunto de dados `meninos`, criado acima.

Selecionando um subconjunto de uma linha específica

```

# Selecionando a linha 10
meninos %>%
  slice(10)

## # A tibble: 1 x 10
##   idadeMae altura  peso anosEst renda    ig tipoParto fumo  pesoRN sexo
##       <dbl>  <dbl> <dbl>   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1      28    1.55    74      5  2.53    39 normal    não    3650 masc

# Selecionando várias linhas, por exemplo, linhas de 1 a 5
meninos %>%
  slice(1:5)

## # A tibble: 5 x 10
##   idadeMae altura  peso anosEst renda    ig tipoParto fumo  pesoRN sexo
##       <dbl>  <dbl> <dbl>   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1      19    1.53    70      7  0.92    36 cesareo  não    2160 masc
## 2      27    1.75    62     11  2.41    36 cesareo  não    2800 masc
## 3      28    1.5     48.5    6  3.13    37 normal  não    3285 masc
## 4      31    1.55    65      5  0.72    37 cesareo  não    3100 masc
## 5      27    1.6     60      8  2.41    37 cesareo  sim    3100 masc

```

É possível também selecionar linhas de acordo com determinado grupo, usando a função `group_by()`, incluído no pacote `dplyr`.

```
meninos %>%
  group_by(fumo) %>%
  slice (1)

## # A tibble: 2 x 10
## # Groups:   fumo [2]
##   idadeMae altura peso anosEst renda    ig tipoParto fumo  pesoRN sexo
##   <dbl>   <dbl> <dbl>   <dbl> <dbl> <dbl> <fct>   <fct> <dbl> <fct>
## 1     27     1.6   60      8  2.41    37 cesareo sim    3100 masc
## 2     19     1.53  70      7  0.92    36 cesareo não   2160 masc
```

A função `slice()` é acompanhada por vários auxiliares para casos de uso comuns:

- `slice_head()` e `slice_tail()` selecionam a primeira ou a última linha;
- `slice_sample()` seleciona linhas aleatoriamente;
- `slice_min()` e `slice_max()` selecionam linhas com valores mais altos ou mais baixos de uma variável.

Selecionando um subconjunto de forma aleatória

A função `slice_sample()` substitui a `sample_n()`:

```
meninos3 <- meninos %>% slice_sample(n = 20)
meninos3
```

```
## # A tibble: 20 x 10
##   idadeMae altura peso anosEst renda    ig tipoParto fumo  pesoRN sexo
##   <dbl>   <dbl> <dbl>   <dbl> <dbl> <dbl> <fct>   <fct> <dbl> <fct>
## 1     20     1.6   70      7  0.6    40 normal não   3320 masc
## 2     30     1.53  57      8  1.69   40 cesareo não   3275 masc
## 3     33     1.53  58      8  1.92   24 normal não   750  masc
## 4     17     1.6   47      5  1.92   26 normal não   785  masc
## 5     27     1.52  80      6  1.69   39 normal sim    3400 masc
## 6     19     1.55  53      9  1.69   40 normal não   3370 masc
## 7     20     1.76  73.5    11 1.69   40 normal não   3380 masc
## 8     22     1.62  69      8  2.17   39 normal sim    2930 masc
## 9     30     1.72  64.2    10 2.89   37 normal não   3330 masc
## 10    27     1.66  66      8  2.24   40 normal não   3265 masc
## 11    31     1.56  65      8  1.64   40 normal não   3635 masc
## 12    25     1.59  58      6  1.2    39 normal não   3610 masc
## 13    39     1.54  77      11 0.72   39 cesareo não   3715 masc
## 14    29     1.64  76      11 3.61   35 cesareo não   1750 masc
## 15    35     1.55  53      7  3.61   39 normal não   3395 masc
## 16    32     1.55  70      4  1.22   39 normal sim    3410 masc
## 17    35     1.65  59.5    11 1.93   40 normal não   3660 masc
## 18    36     1.62  71      11 4.82   34 cesareo não   2675 masc
## 19    22     1.54  65      6  2.65   39 cesareo não   3365 masc
## 20    21     1.62  65      11 1.93   39 normal sim    2955 masc
```

Para maiores informações em relação a estas funções consulte a ajuda (`?slice()`).

5.3.6 Função arrange()

Ordena as linhas pelos valores de uma coluna de forma ascendente ou descentente.

Voltando a amostra meninos1, será colocado em ordem crescente a variável pesoRN:

```
arrange(meninos1, pesoRN)
```

```
## # A tibble: 20 x 10
##   idadeMae altura peso anosEst renda    ig tipoParto fumo  pesoRN sexo
##   <dbl>   <dbl> <dbl>   <dbl> <dbl> <dbl> <fct>   <fct> <dbl> <fct>
## 1     23   1.55  47      6  1.2    27 cesareo  não  1050 masc
## 2     36   1.6   68.5     5  4.82   28 cesareo  não  1905 masc
## 3     32   1.65  60      8  0.96   34 normal  sim   1990 masc
## 4     14   1.58  50      7  1.69   37 normal  não  2525 masc
## 5     21   1.64  52      9  1.93   37 normal  não  2680 masc
## 6     25   1.58  66     10  1.92   37 cesareo  não  2755 masc
## 7     26   1.65  60     11  1.92   39 normal  sim   2825 masc
## 8     31   1.76  77      8  1.69   39 cesareo  não  2860 masc
## 9     18   1.55  47      9  1.2    38 normal  não  2995 masc
## 10    18   1.56  50      8  1.69   38 cesareo  não  3115 masc
## 11    22   1.6   55      8  1.92   40 cesareo  não  3150 masc
## 12    24   1.6   61      9  2.41   40 normal  não  3160 masc
## 13    26   1.68  70     11  3.61   36 normal  não  3290 masc
## 14    43   1.65  63      8  4.82   40 normal  não  3300 masc
## 15    29   1.63  66      3  2.07   40 normal  não  3340 masc
## 16    18   1.59  65     11  2.17   39 normal  não  3380 masc
## 17    19   1.69  65      8  2.89   40 cesareo  não  3430 masc
## 18    37   1.58  70      5  2.89   40 normal  não  3445 masc
## 19    25   1.6   60     11  2.41   40 cesareo  não  3860 masc
## 20    23   1.65  56      8  1.57   41 normal  não  4045 masc
```

Para a ordem decrescente, colocar a função desc(), dentro da função arrange()

```
arrange(meninos1, desc(pesoRN))
```

```
## # A tibble: 20 x 10
##   idadeMae altura peso anosEst renda    ig tipoParto fumo  pesoRN sexo
##   <dbl>   <dbl> <dbl>   <dbl> <dbl> <dbl> <fct>   <fct> <dbl> <fct>
## 1     23   1.65  56      8  1.57   41 normal  não  4045 masc
## 2     25   1.6   60     11  2.41   40 cesareo  não  3860 masc
## 3     37   1.58  70      5  2.89   40 normal  não  3445 masc
## 4     19   1.69  65      8  2.89   40 cesareo  não  3430 masc
## 5     18   1.59  65     11  2.17   39 normal  não  3380 masc
## 6     29   1.63  66      3  2.07   40 normal  não  3340 masc
## 7     43   1.65  63      8  4.82   40 normal  não  3300 masc
## 8     26   1.68  70     11  3.61   36 normal  não  3290 masc
## 9     24   1.6   61      9  2.41   40 normal  não  3160 masc
## 10    22   1.6   55      8  1.92   40 cesareo  não  3150 masc
## 11    18   1.56  50      8  1.69   38 cesareo  não  3115 masc
## 12    18   1.55  47      9  1.2    38 normal  não  2995 masc
## 13    31   1.76  77      8  1.69   39 cesareo  não  2860 masc
## 14    26   1.65  60     10  1.92   39 normal  sim   2825 masc
```

```

## 15      25   1.58   66       5   1.92    37 cesareo  não  2755 masc
## 16      21   1.64   52       9   1.93    37 normal  não  2680 masc
## 17      14   1.58   50       7   1.69    37 normal  não  2525 masc
## 18      32   1.65   60       8   0.96    34 normal  sim   1990 masc
## 19      36   1.6    68.5     5   4.82    28 cesareo  não  1905 masc
## 20      23   1.55   47       6   1.2     27 cesareo  não  1050 masc

```

5.3.7 Função count()

Permite contar rapidamente os valores únicos de uma ou mais variáveis. Esta função tem os seguintes argumentos:

- **x** → dataframe
- **wt** → pode ser NULL (padrão) ou uma variável
- **sort** → padrão = FALSE; se TRUE, mostrará os maiores grupos no topo
- **name** → O nome da nova coluna na saída; padrão = NULL

Quando o argumento **name** é omitido, a função retorna *n* como nome padrão.

Usando o dataframe **mater1**, a função **count()** irá contar o número de parturientes fumantes, variável dicotômica **fumo**:

```
count(mater1, fumo)
```

```

## # A tibble: 2 x 2
##   fumo     n
##   <fct> <int>
## 1 sim     301
## 2 não    1067

```

5.3.8 Operador pipe %>%

O operador pipe **%>%** pode ser usado para inserir um valor ou um objeto no primeiro argumento de uma função. Ele pode ser acionado digitando **%>%** ou usando o atalho **ctrl+shift+M**. Em vez de passar o argumento para a função separadamente, é possível escrever o valor ou objeto e, em seguida, usar o **pipe** para convertê-lo como o argumento da função na mesma linha. Funciona como se o **pipe** jogasse o objeto dentro da função seguinte.

Vários comando foram utilizados, manipulando o banco de dados **mater**. Alguns orocedimentos, serão mostrados, usando, agora, o operador pipe.

Em primeiro lugar, serão selecionadas algumas colunas do dataframe **mater**; acresida a variável **imc**; selecionado os recém-nascidos a termo do sexo masculino, que no banco de dados **mater** é igual a 1. Tudo em um só comando!

```

meusDados <- mater %>%
  select(idadeMae, altura, peso, anosEst, renda,
         ig, tipoParto, fumo, pesoRN, sexo) %>%
  mutate(imc = peso/altura^2) %>%
  filter (ig>=37 & ig<42, sexo == 1)

```

```

str(meusDados)

## # tibble [592 x 11] (S3: tbl_df/tbl/data.frame)
## $ idadeMae : num [1:592] 28 31 27 28 18 28 22 28 25 14 ...
## $ altura   : num [1:592] 1.5 1.55 1.6 1.58 1.76 1.63 1.54 1.55 1.56 1.51 ...
## $ peso     : num [1:592] 48.5 65 60 47 65.5 72 65 74 70 56.7 ...
## $ anosEst  : num [1:592] 6 5 8 8 7 11 6 5 9 6 ...
## $ renda    : num [1:592] 3.13 0.72 2.41 1.69 1.93 1.92 2.65 2.53 0.48 1.92 ...
## $ ig       : num [1:592] 37 37 37 38 39 39 39 39 39 39 ...
## $ tipoParto: num [1:592] 1 2 2 1 1 2 2 1 1 1 ...
## $ fumo     : num [1:592] 2 2 1 2 1 1 2 2 2 2 ...
## $ pesoRN   : num [1:592] 3285 3100 3100 2800 3270 ...
## $ sexo     : num [1:592] 1 1 1 1 1 1 1 1 1 1 ...
## $ imc      : num [1:592] 21.6 27.1 23.4 18.8 21.1 ...

```

Observe que o dataframe `mater` aparece apenas no início e, como ele é um argumento das outras funções, ele é transferido, automaticamente, não havendo necessidade de escrever dentro na função.

No final, retornará um novo dataframe que foi colocado em objeto, denominado `meuDados`, o qual contém informações de todos os 592 meninos, nascidos a termo e de suas mães.

5.4 Manipulação de datas

Originalmente, todos os que trabalham com o R queixavam-se de como era frustrante trabalhar com datas. Era um processo que causava grande perda de tempo nas análises. O pacote lubridate foi criado para simplificar ao máximo a leitura de datas e extração de informações dessas datas.

Antes de usar, há necessidade de instalar e carregar o pacote.

```
install.packages("lubridate")
```

```
library(lubridate)
```

A função mais importante para leitura de dados no lubridate é a `ymd()`. Essa função serve para ler qualquer data de uma *string* no formato YYYY-MM-DD.

Para iniciar, será registrada uma data qualquer: Observe que o R registrou esta dada como um objeto da classe numérica.

```

data.hoje <- "29/10/2022"
class(data.hoje)

```

```
## [1] "character"
```

Para converter esta data da classe `character` para a classe `date`, usar a função `dmy()`:

```

data.hoje <- dmy(data.hoje)
class(data.hoje)

```

```
## [1] "Date"
```

Uma grande facilidade que essas funções trazem é poder criar objetos com classe `date` a partir de números e `character` em diversos formatos.

```
dmy(29102022)
```

```
## [1] "2022-10-29"
```

```
dmy("29/10/2022")
```

```
## [1] "2022-10-29"
```

```
dmy("29102022")
```

```
## [1] "2022-10-29"
```

```
dmy("29.10.2022")
```

```
## [1] "2022-10-29"
```

Se além da data, houver necessidade de especificar o horário, basta usar `dmy_h()`, `dmy_hm()` e `dmy_hms()`. Se for usado o padrão americano, pode ser usado `ymd()`.

O `lubridate` traz diversas funções para extrair os componentes de um objeto da classe `date`.

- `second()` - extrai os segundos.
- `minute()` - extrai os minutos.
- `hour()` - extrai a hora.
- `wday()` - extrai o dia da semana.
- `mday()` - extrai o dia do mês.
- `month()` - extrai o mês.
- `year()` - extrai o ano.

Por exemplo,

```
dn <- dmy("04/10/1947")
year(dn)
```

```
## [1] 1947
```

Para acrescentar um horário a ao objeto data de nascimento (dn):

```
hour(dn) <- 04
dn
```

```
## [1] "1947-10-04 04:00:00 UTC"
```

Data e horário do dia em que essa página foi editada pela última vez.

```
today()  
  
## [1] "2023-03-16"  
  
now()  
  
## [1] "2023-03-16 18:46:38 -03"
```

5.4.1 Operações com datas

Intervalos

Intervalos podem ser salvos em objetos com classe `interval`.

```
inicio <- dmy("01/01/2022")  
final <- dmy("29/10/2022")  
  
periodo <- interval(inicio, final)  
periodo  
  
## [1] 2022-01-01 UTC--2022-10-29 UTC
```

```
class(periodo)
```

```
## [1] "Interval"  
## attr(,"package")  
## [1] "lubridate"
```

Aritmética com datas

```
# Somando datas  
  
today() + ddays(60)      # hoje + 60 dias
```

```
## [1] "2023-05-15"
```

```
today() + dyears(1)      # hoje + 1 ano
```

```
## [1] "2024-03-15 06:00:00 UTC"
```

```
# Duração de um intervalo  
  
intervalo <- dmy("10-01-2022") %--% dmy("17-10-2022")  
intervalo
```

```
## [1] 2022-01-10 UTC--2022-10-17 UTC
```

```
intervalo/ddays(1)      # Número de dias  
  
## [1] 280  
  
intervalo/dmonths(1)   # Número de meses  
  
## [1] 9.199179  
  
intervalo / dweeks(1) # Número de semanas  
  
## [1] 40  
  
as.period(intervalo)  
  
## [1] "9m 7d 0H 0M 0S"
```

Para mais informações sobre o `lubridate`, consulte a ajuda do pacote.

6 Descrevendo os dados

Nos relatórios ou artigos científicos, a comunicação dos resultados é feita através da combinação de medidas resumidoras e visualização dos dados por meio de tabelas e gráficos.

6.1 Dados brutos

Habitualmente, costuma-se armazenar os dados em bancos de dados (dataframes ou tibbles). Entretanto, eles estão registrados de forma aleatória e não classificada. Ao se visualizar um dataframe, é difícil responder perguntas em relação a qualquer variável, principalmente, em grandes banco de dados. Eles se constituem uma lista, um rol de valores colocados na ordem em que foram obtidos. Parecem um jogo de quebra cabeça antes de serem organizados e resumidos! São denominados de *dados brutos* ou, também, de dados não agrupados.

6.2 Medidas resumidoras

As maneiras mais usadas para resumir o conjunto de dados são:

- Primeiro, um valor em torno do qual os dados têm uma tendência para se reunir ou se agrupar, denominado de medida sumária de localização ou medida de tendência central.
- Em segundo lugar, um valor que mede o grau em que os dados se dispersam, denominado de medida de dispersão ou variabilidade.

Para trabalhar nesta seção, serão necessários os seguintes pacotes:

```
pacman::p_load(dplyr, readxl)
```

E o arquivo `dadosMater15.xlsx`, amostra de 15 recém-nascidos do banco de dados original (`dadosMater.xlsx`) que pode ser obtido [aqui](#) e baixado para o seu diretório de trabalho.

Agora, vamos criar um objeto, `mater15`, para receber os dados, a partir do diretório de trabalho, executando o seguinte código:

```
mater15 <- read_excel("dadosMater15.xlsx")
```

6.2.1 Medidas de tendência central

6.2.1.1 Média A média (\bar{x}) é a mais usada medida de tendência central. Ela é calculada pela razão entre a soma de todas as observações de um conjunto de dados e o total de observações. A média é mais adequada para medidas numéricas simétricas.

$$\bar{x} = \frac{\sum(x_1 + x_2 + x_3 + \dots + x_n)}{n}$$

Se no conjunto de dados houver algum valor ausente (*missing*), o comando mostra o resultado como NA (*not available*). Para corrigir isto, basta colocar o argumento `na.rm = TRUE` na função `mean()`. Assim, o R vai retornar a média, ignorando os valores ausentes. Recomenda-se sempre usar o argumento.

A média aritmética dos pesos dos recém-nascidos (`pesoRN`) do arquivo `dadosMater15.xlsx` é calculado por:

```
mean (mater15$pesoRN, na.rm = TRUE)
```

```
## [1] 3238.067
```

6.2.1.2 Mediana A mediana (M_d) representa o valor central em uma série ordenada de valores. Assim, metade dos valores será igual ou menor que o valor mediano e a outra metade igual ou maior do que ele. No R, usa-se a função `median()` para calcular o valor da mediana. Vamos utilizar a variável `mater15$apgar1`. Como o Apgar é um escore, a medida resumidora mais adequada é a mediana.

```
median (mater15$apgar1, na.rm = TRUE)
```

```
## [1] 8
```

6.2.1.3 Moda Moda (M_o) é o valor que ocorre com maior frequência em um conjunto de dados. Tem o menor nível de sofisticação. É usada primariamente para dados nominais porque há simplesmente contagem dos valores. Ao contrário das outras medidas de tendência central, a moda não informa nada sobre a ordem das variáveis ou variação dentro das variáveis.

O R não tem uma função embutida padrão para calcular a moda. Portanto, há necessidade de ser criada uma função de usuário para calcular a moda.

```
moda <- function(x) {  
  z <- table(as.vector(x))  
  names(z)[z == max(z)]}
```

Usando esta função pode-se calcular a moda para a variável `mater15$apgar1`.

```
moda (mater15$apgar1)
```

```
## [1] "8"
```

6.2.1.4 Quantil Uma medida de localização bastante utilizada são os `quantis` que são pontos estabelecidos em intervalos regulares que dividem a amostra em subconjuntos iguais. Se estes subconjuntos são em número de 100, são denominados de `percentis`; se são em número de 10, são os `decis` e em número de 4, são os `quartis`. A função apropriada no R para obter o quantil é `quantile()`.

Para determinar os três quartis do peso dos recém-nascidos (`mater15$pexoRN`), usa-se:

```
quantile (mater15$pexoRN, c (0.25, 0.50, 0.75))
```

```
##    25%    50%    75%  
## 2962.5 3245.0 3515.0
```

Observe que o percentil 50º é igual a mediana. O percentil 75º é o ponto do conjunto de dados onde 75% dos recém-nascidos têm um peso inferior a 3515g e 25% está acima deste valor.

6.2.1.5 Média aparada As médias aparadas são estimadores robustos da tendência central. Para calcular uma média aparada, é removida uma quantidade predeterminada de observações em cada lado de uma distribuição e realizada a média das observações restantes. Um exemplo de média aparada é a própria mediana.

A base R tem como calcular a média aparada acrescentando o argumento `trim =`, proporção a ser aparada. Se for aparado 20%, usa-se `trim = 0.2`. Isto significa que serão removidos 20% dos dados dos dois extremos. No caso da amostra de 15 recém-nascidos, serão removidos três valores mais baixos e três valores mais altos, passando a mostra a ter 9 valores, e a média aparada será a média destes 9 valores.

O comando para obter a média aparada é:

```
mean (mater15$pesoRN, na.rm = TRUE, trim = 0.20)
```

```
## [1] 3253.889
```

6.2.2 Medidas de Dispersão

6.2.2.1 Amplitude A amplitude de um grupo de medições é definida como a diferença entre a maior observação e a menor.

No conjunto de dados dos pesos dos recém-nascidos, a amplitude pode ser obtida, no R, com a função `range()`, que retorna o valor mínimo e o máximo.

```
range (mater15$pesoRN, na.rm = TRUE)
```

```
## [1] 2051 4070
```

6.2.2.2 Intervalo Interquartil A intervalo interquartil (IIQ), também conhecido como amplitude interquartil (AIQ) é uma forma de média aparada. É simplesmente a diferença entre o terceiro e o primeiro quartil, ou seja, a diferença entre o percentil 75 e o percentil 25. Considere a escolaridade (`anosEst`) das parturientes da amostra `dadosMater15.xlsx`. Os percentis 25 e 75 são obtidos por:

```
quantile (mater15$anosEst, c(0.25,0.75))
```

```
## 25% 75%
##   6    8
```

Também pode ser usada a função `summary ()`:

```
summary(mater15$anosEst)
```

```
##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##        4       6       7       7       8      11
```

Portanto, o IIQ está entre 6 a 8 anos de estudo ou, $8 - 6 = 2$ anos de estudos completos. Em outras palavras, 50% das mulheres desta amostra têm de 6 a 8 anos de estudo.

6.2.2.3 Variância e Desvio Padrão A variância e o desvio padrão fornecem uma indicação de quanto aglomerados em torno da média os dados de uma amostra estão. Estes tipos de medidas representam desvios (erros) da média. Quando se verifica o desvio de cada valor (x) em relação à média \bar{x} , os desvios positivos se anulam com os negativos, resultando em uma soma igual a zero.

A consequência deste fato é que não é possível resumir os desvios numa única medida de variabilidade. Para se chegar a uma medida de variabilidade há necessidade de se eliminar os sinais, antes de somar todos os desvios em relação à média.

Uma maneira de se fazer isso é elevar todas as diferenças ao quadrado. Assim, se obtém o desvio em relação à média elevado ao quadrado. A soma destes valores é denominada de *Soma dos Quadrados (SQ) dos Desvios* ou *Soma dos Erros ao Quadrado*. Se o interesse é apenas saber o erro ou desvio médio, divide-se por n (tamanho da amostra). No entanto, em geral o interesse se concentra em usar o desvio ou erro na amostra para estimar o erro na população. Dessa maneira, divide-se a Soma dos Quadrados por $n - 1$. Essa medida é conhecida como variância (s^2). O divisor, $n-1$, é denominado de *graus de liberdade (gl)* associados à variância.

Os graus de liberdade representam o número de desvios que estão livres para variar. É um conceito de difícil explicação. Suponha uma maternidade há 50 anos atrás, quando não havia alojamento conjunto. Nessa época era comum os recém-nascidos normais ficarem em um berçário. A cada horário de amamentação eles eram levados para os quartos de suas mães para mamar. Posteriormente, eram trazidos para o berçário e colocados nos berços até a próxima mamada. Suponha que, em um determinado momento, havia 15 bebês e que, no berçário, existiam 15 berços (postos) para colocá-los durante o intervalo das mamadas. Quando o primeiro recém-nascido chega, a enfermeira poderá escolher qualquer um dos berços para o colocar. Depois, quando o próximo recém-nascido chegar, ela terá 14 opções de escolha, pois um dos berços está ocupado. Ainda existe uma boa liberdade de escolha. No entanto, à medida que os recém-nascidos forem sendo trazidos para o berçário, chegará a um ponto em que 14 berços estarão ocupados. Agora, a enfermeira não terá liberdade de escolha, pois só resta um berço. Nesse exemplo existem 14 graus de liberdade. Para o último recém-nascido não houve liberdade de escolha (65). Portanto os graus de liberdade são um menos o tamanho da amostra ($n - 1$).

A variância é a razão entre a soma dos quadrados e as observações realizadas menos um.

$$s^2 = \frac{\sum(x_i - \bar{x})^2}{n - 1}$$

No R existem as funções `sd()` e `var()`, também incluídas no R base, que facilmente calculam essas medidas de dispersão.

Usando a variável `mater15$pesoRN`, tem-se:

```
var(mater15$pesoRN, na.rm =TRUE)
```

```
## [1] 273861.8
```

O desvio padrão é a raiz quadrada da variância: $s = \sqrt{var}$

```
sqrt (var(mater15$pesoRN))
```

```
## [1] 523.318
```

Ou,

```
sd (mater15$pesoRN, na.rm = TRUE)
```

```
## [1] 523.318
```

A variância e desvio padrão são medidas de variabilidade. Representam quanto bem a média representa os dados. Informa se ela está funcionando bem como modelo. Pequenos desvios padrão mostram que existe pouca variabilidade nos dados, que eles se aproximam da média. Quando existe um grande desvio padrão, a média não é muito precisa para representar os dados.

O desvio padrão, além de medir a precisão com que a média representa os dados, também informa sobre o formato dos dados e por isso é uma medida de dispersão. Em uma amostra onde desvio padrão é pequeno, os dados se agrupam próximo a média e o formato da distribuição fica mais pontiagudo (curva em azul, 31). Nesse caso a média representa bem os dados. Em outra amostra, com a mesma média anterior, mas com os dados mais dispersos entorno da média, o desvio padrão é maior e o formato da distribuição fica achatado (curva verde, na Figura 31). Nesse caso a média não é uma boa representação dos dados.

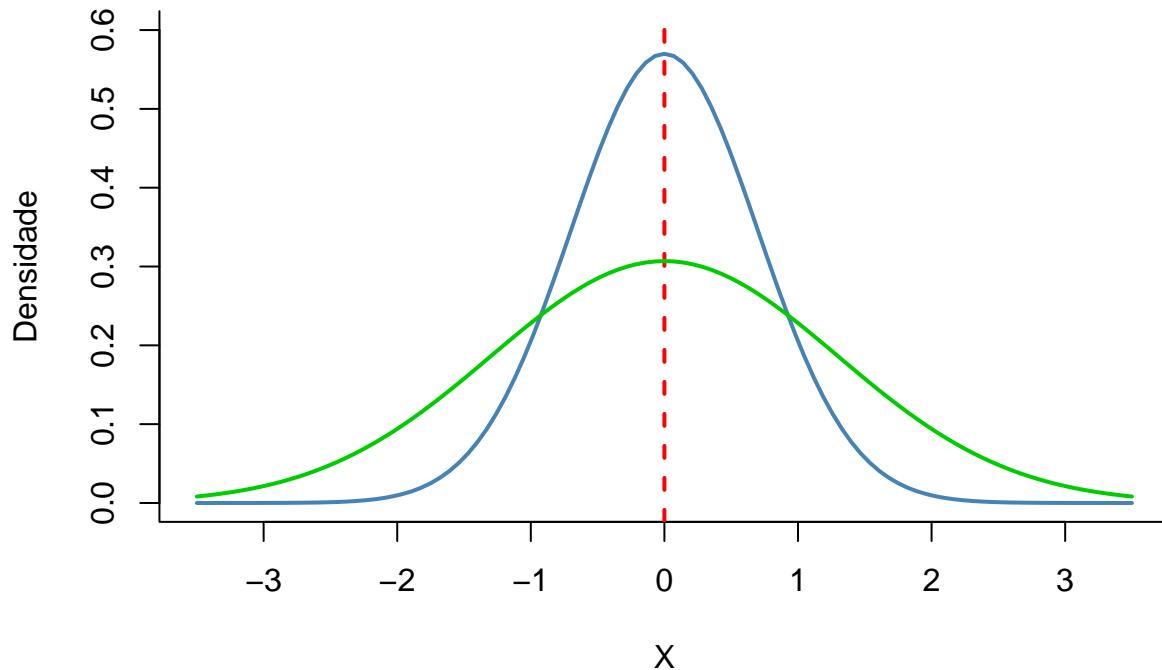


Figura 31: Dispersão dos dados em torno da média.

6.2.2.4 Coeficiente de Variação O desvio padrão por si só tem limitações. Um desvio padrão de duas unidades pode ser considerado pequeno para um conjunto de valores cuja média é 100. Entretanto, se a média for 5, ele se torna muito grande. Além disso, o desvio padrão por ser expresso na mesma unidade dos dados, não permite aplicá-lo na comparação de dois ou mais conjunto de dados que têm unidades diferentes. Para eliminar essas limitações, é possível caracterizar a dispersão ou variabilidade dos dados em termos relativos, usando uma medida denominada Coeficiente de Variação (CV), também conhecido como Desvio Padrão

Relativo ou Coeficiente de Variação de Pearson. É expresso, em geral como uma porcentagem, sendo definido como a razão do desvio padrão pela média:

$$CV = \frac{s}{\bar{x}}$$

Multiplicando o valor da equação por 100 tem-se o CV percentual.
O R não possui uma função específica para calcular o CV.

Foi criada uma função específica para isso, já multiplicada por 100.

```
coef_var <- function (valores) {  
  (sd(valores, na.rm=T) / mean(valores, na.rm=T))*100}
```

Portanto, o CV da variável `mater15$pesoRN` é igual a:

```
coef_var (mater15$pesoRN)
```

```
## [1] 16.16144
```

Se usarmos outra variável do banco de dados, por exemplo, `mater15$idadeMae`, o CV será igual a:

```
coef_var (mater15$idadeMae)
```

```
## [1] 25.83343
```

O peso do recém-nascido tem um CV = 16.2 e a idade materna um CV = 25.8, mostrando que esta tem uma maior variabilidade. Quanto menor o desvio padrão, menor o CV e, consequentemente, menor a variabilidade. Um CV $\geq 50\%$, sugere que a variável tem uma distribuição assimétrica.

6.2.3 Escolha da medida resumidora

A seleção da medida de tendência central mais adequada depende de vários fatores, incluindo a natureza dos dados e o propósito da sumarização.

O tipo da variável tem substancial influência na escolha da medida de tendência central a ser usada. A moda é mais apropriada para dados nominais e seu uso com variáveis ordinais resulta em uma perda no poder em termos de informação que se poderia obter dos dados.

A mediana é mais adequada para variáveis ordinais, embora possa ser usada para variáveis contínuas, especialmente quando a distribuição dos dados é assimétrica. A mediana não deveria ser usada com dados nominais porque os postos assumidos não podem ser obtidos com dados de nível nominal.

Finalmente, a média somente deve ser usada com dados contínuos simétricos, se houver assimetria a mediana deve ser preferida.

As medidas de dispersão devem estar associadas a uma medida de tendência central. Elas caracterizam a variabilidade dos dados na amostra. Com dados ordinais usar a amplitude ou o intervalo interquartil. O desvio padrão não é apropriado em dados ordinais devido à natureza não numérica destes.

Com os dados numéricos deve-se usar o desvio padrão, que utiliza toda a informação nos dados, ou o intervalo interquartil (IIQ). Quando os dados forem simétricos, usar a média acompanhada do desvio padrão, caso contrário, usar a mediana e o IIQ. Não misturar e combinar medidas {[22]}.

6.3 Tabelas

Existem muitas maneira de criar tabelas no R. Para mostrar como construir as tabelas, será feita a leitura do conjunto de dados (`dadosMater.xlsx`) da maternidade-escola do Hospital Geral de Caxias do Sul, RS, já mostrado quando se discutiu o pacote `dplyr`. A partir do diretório de trabalho:

```
library(readxl)
mater <- read_excel("dadosMater.xlsx")
```

6.3.1 Tabelas de Frequências

Tabela de frequência para dados categóricos

Uma maneira concisa que permite observar a variável e extrair informação sobre o seu comportamento, é a utilização de uma *tabela de frequência*. A tabela de frequência deve ser simples, clara e objetiva, ou seja, não deve ter um volume muito grande de informações. Deve ser autoexplicativa, não deve haver necessidade de ler o texto para entendê-la.

A tabela de frequência (Tabela 3) agrupa os dados por categorias ou classes, contabilizando o número ocorrências em cada categoria. O número de observações em uma determinada classe recebe o nome de *frequência absoluta* (f). Além da frequência absoluta, costuma aparecer a frequência relativa (fr) que representa a proporção da classe em relação ao número total de observações (n), calculada por $fr = \frac{f}{n}$, a frequência percentual (fp), obtida pela multiplicação da frequência relativa por 100 e a frequência acumulada, que é a soma de todas as classes até a classe atual, podendo ser frequência acumulada absoluta (F), frequência acumulada relativa (Fr) ou frequência acumulada percentual (Fp).

Em uma tabela, os dados são apresentados em colunas verticais indicadoras e linhas horizontais. Nas linhas aparecem as categorias e nas colunas as frequências, constituindo o corpo da tabela. O cabeçalho indica a natureza do conteúdo de cada coluna. No cruzamento das colunas e das linhas, tem-se as caselas ou casas.

Existem algumas recomendações na construção de uma tabela de frequência (66):

- deve ter um título na parte superior que responda as perguntas: “o que? quando? onde?” relativas ao fato estudado;
- deve ter um rodapé, na parte inferior da tabela, onde se coloca notas necessárias e a fonte dos dados;
- as colunas externas da tabela devem ser abertas, o emprego de linhas verticais para a separação das colunas no corpo da tabela é opcional;
- Na parte superior e inferior, as tabelas devem ser fechadas por linhas horizontais;
- Nenhuma casela deve ficar vazia, apresentando um número ou um símbolo. Se não se dispuser do dado, colocar reticências ... e a presença de um X representa que o dado foi omitido para evitar a identificação.

Se os dados forem nominais, a ordenação das categorias é arbitrária, costuma-se colocar em primeiro lugar a maior frequência (Tabela 2) , colocando-os em categorias ordenadas (67).

Construção da tabela de frequência

Com frequência há necessidade se recodificar uma variável numérica para categórica. Por exemplo, que as gestantes são, classicamente, subdivididas em menores de 20 anos (adolescentes), 20 a 35 anos e maiores de 35 anos. No conjunto `dadosMater`, a variável `idadeMae` é uma variável numérica onde a mulher mais jovem tem 13 anos e mais velha tem 46. Não existe uma variável com a idade categorizada. Será feita uma transformação desta variável numérica em categórica com três níveis: < 20 anos, 20 a 35 anos e >35 anos.

Será usada a função `cut()` do pacote do R base. Esta função tem vários argumentos:

- `x` → vetor numérico

Tabela 2: Distribuição de frequência de drogadição em parturientes do Hospital Geral de Caxias do Sul, RS, 2008.

Drogadição	f	fr	fp	Fp
Não drogaditas	904	0,955	95,5	95,5
Medicamentos	23	0,024	2,4	97,9
Álcool	17	0,018	1,8	99,7
Crack	2	0,002	0,2	99,9
Cocaína	1	0,001	0,1	100,0
Total	947	1,000	100,0	

- **breaks** → vetor numérico de dois ou mais pontos de corte exclusivos ou um único número (maior ou igual a 2) dando o número de intervalos nos quais x deve ser subdividido
- **labels** → rótulos para os níveis das categorias resultante. Por padrão, os rótulos são construídos usando a notação de intervalo $(a, b]$ (aberto à esquerda e fechado à direita).
- **include.lowest** → valor lógico, se o menor valor será incluído, ou o maior, se **right = TRUE**. Padrão = **include.lowest=TRUE**
- **right** → valor lógico indicando se o intervalo deve ser fechado à direita e aberto a esquerda. Padrão = **right = FALSE**.
- **ordered_result** → valor lógico indicando se o resultado deve ser um fator ordenado.

```
mater$idadeCateg <- cut(mater$idadeMae,
                         breaks = c(13, 20, 36, 46),
                         labels = c("<20a", "20-35a", ">35a"),
                         include.lowest = TRUE,
                         right = FALSE,
                         ordered_result =TRUE)
```

Neste exemplo, foi usado **right = FALSE**, em consequência, o intervalo 13 a 20, incluirá o 13 (menor idade) e excluirá o 20, o intervalo 20 a 36, incluirá o 20 e excluirá o 36 e o último intervalo incluirá o 36 e excluirá o 46, que é o valor mais alto. Em função disso, foi incluído mais um argumento **include.lowest=TRUE**, para incluir o valor 46.

Para se verificar como ficou a distribuição de frequência absoluta, constroi-se uma tabela, inicialmente com a função **table()**:

```
f_abs <- table (mater$idadeCateg)
f_abs
```

```
##
##   <20a 20-35a  >35a
##   219    992    157
```

O cálculo das frequências relativas pode ser dada pela função **prop.table()**, usando a frequência absoluta e a função **round()** para arredondar os valores para 3 dígitos:

```
f_rel <- round(prop.table(f_abs), 3)
f_rel
```

```
##
##   <20a 20-35a  >35a
##   0.160  0.725  0.115
```

Multiplicando por 100 a f_rel, tem-se a frequência percentual:

```
f_perc <- round(f_rel*100, 2)
f_perc

## 
##   <20a 20-35a    >35a
##   16.0    72.5    11.5

f_abs <- c (f_abs, sum(f_abs))
f_rel <- c (f_rel, sum (f_rel))
f_perc <- c (f_perc, sum (f_perc))

tab1 <- cbind(f_abs,
               f_rel ,
               f_perc)

tab1 <- as.data.frame(tab1)
row.names(tab1)[4] <- "Total"
colnames(tab1) <- c("Frequênci", "Freq.Relativa", "Freq.Percentual")
tab1

##          Frequênci Freq.Relativa Freq.Percentual
## <20a           219       0.160        16.0
## 20-35a         992       0.725        72.5
## >35a           157       0.115        11.5
## Total          1368      1.000       100.0
```

A função `kable()` do `knitr(68)` pode ser usada, retornando uma tabela muito simples e profissional (Tabela 3). Como a função somente trabalha com matrizes e dataframes, a tabela `tab1` deve ser um `data.frame`.

Para melhorar o aspecto da tabela, pode-se acrescentar funções do pacote `kableExtra (69)` com a sintaxe pipe (`%>%`), como exemplo, `kable_styling()`.

```
knitr::kable(tab1,
             booktabs = TRUE,
             format = 'latex',
             caption = "Distribuição das puérperas por faixa etária, Hospital Geral de Caxias do Sul, R",
             format.args = list(decimal.mark = ",")) %>%
kable_styling(full_width = T,
              latex_options = "hold_position")
```

Tabela 3: Distribuição das puérperas por faixa etária, Hospital Geral de Caxias do Sul, RS, 2008.

	Frequênci	Freq.Relativa	Freq.Percentual
<20a	219	0,160	16,0
20-35a	992	0,725	72,5
>35a	157	0,115	11,5
Total	1368	1,000	100,0

Tabela de frequência para dados numéricos

Como fazer a distribuição de frequência de uma variável contínua sem um critério pré-determinado para as classes?

Como exemplo, será usado, agora, o IMC pré-gestacional das parturientes do banco de dados `dadosMater.xlsx`). Esta variável não existe, tem-se apenas o peso e a altura e, portanto, com estes dados ela pode ser criada:

```
mater$imc <- round(mater$peso/mater$altura^2, 1)
```

Após, segue-se os seguintes passos:

1. *Estabelecimento do número de classes (k):*

Antes, as classes foram estabelecidas de acordo com algum critério. Em geral, quando não há um padrão pré-determinado, o número de classes é estabelecido de acordo com o tamanho da amostra. Este número pode ser escolhido lembrando-se das oscilações que ocorrem nos dados e do interesse do pesquisador em mostrar seus dados. Não existe uma regra totalmente eficiente para determinar o número de classes. É importante ter bom senso, de maneira que seja possível ver como os valores se distribuem.

Para a maioria dos dados, é recomendado e 8 a 20 classes, isto é, $8 \leq k \leq 20$. Com poucas classes, perde-se precisão e, com muitas classes, a tabela torna-se muito extensa. Baseado na regra de Sturges , é sugerido usar a recomendação da Figura 32 (70).

Número de observações (n)	Número de classes (k)
1	1
2	2
3 a 5	3
6 a 11	4
12 a 23	5
24 a 46	6
47 a 93	7
94 a 187	8
188 a 376	9
377 a 756	10

Figura 32: Número de classes baseado em Sturges

Para a variável `imc`, como existem 1368 observações, deve-se usar ao redor de 10 classes. Executando a função `nclass.Sturges ()`, abaixo, o número de classes é igual a:

```
k <- nclass.Sturges (mater$imc)  
k
```

```
## [1] 12
```

```
; k
```

2. *Amplitude e limites das classes:*

A classe possui um limite inferior e um limite superior. O importante é que os limites dos intervalos sejam mutuamente exclusivos, isto é, cada valor deve ser representado em um único intervalo. Além disso, os intervalos devem ser exaustivos, isto é, devem conter todos os valores possíveis entre o valor mínimo e o máximo. O recomendado é que as classes sejam homogêneas, ou seja, tenham a mesma amplitude. A amplitude dos valores pode ser obtida com a função `range()`:

```
amplitude <- range(mater$imc)
amplitude
```

```
## [1] 11.8 48.7
```

Usando esta amplitude dos dados, é possível ter a largura (amplitude) das classes (`h`), usando a diferença entre o mínimo e máximo e dividindo pelo número de classes (`k`):

```
h <- round(diff(amplitude)/k, 0)
h
```

```
## [1] 3
```

A fórmula é apenas a diferença absoluta dos limites inferior e superior dividida pelo número de classes, arredondada com o a função `round()` com 1 dígito decimal.

A partir desses dados, é possível construir as classes. A primeira classe será o valor mínimo de 11,8, que pode ser arredondado para 11,8 até 14,8 (11,8 + 3) exclusive; a segunda classe será 14,8 até 17,8 (14,8 + 3) e assim por diante.

3. Construção da tabela:

Pode-se construir a tabela, usando a função `table()` e dentro desta a função `cut()` e dentro dela a função `seq(limite inferior, limite superior, l = número de classes)`.

```
nutriCateg <- table(cut(mater$imc,
                           right = TRUE,
                           include.lowest = TRUE,
                           seq(11.8, 48.7, l = k + 1)))
nutriCateg
```

```
##
## [11.8,14.9]  (14.9,18]      (18,21]    (21,24.1] (24.1,27.2] (27.2,30.3]
##          2        46        258        480        237        176
## (30.3,33.3] (33.3,36.4] (36.4,39.5] (39.5,42.6] (42.6,45.6] (45.6,48.7]
##          87        39        22         12          5          4
```

Preste atenção! Estes comandos que vão gerar a tabelatêm o argumento `right = TRUE` (padrão). Neste caso, ao contrário do comentado anteriormente, onde foi usado `right = FALSE`, os símbolos aparecem como `[]` (na tabela) e significa que o limite inferior da classe foi excluído (aberto à esquerda) e o superior foi incluído (fechado à direita). Aqui, também foi introduzido o argumento `include.lowest = TRUE` para incluir o valor mínimo dos dados (11,8), e a representação gráfica fica `[]`.

Olhando a saída do objeto `nutriCateg`, ela parece pouco esclarecedora e, no caso do IMC, talvez fosse melhor usar outro critério. Como por exemplo o que define o estado nutricional no 1º trimestre de gestação e classifica as gestantes em *baixo peso* ($IMC < 18,5 \text{ kg/m}^2$), *peso adequado* ($18,5 \leq IMC \leq 24,9 \text{ kg/m}^2$),

sobrepeso ($25,0 \leq \text{IMC} \leq 29,9 \text{ kg/m}^2$) e *obesidade* ($\text{IMC} \geq 30 \text{ kg/m}^2$). Assim, é recomendado um ganho de peso total adequado de 12,5 kg a 18 kg para as gestantes classificadas como baixo peso; de 11,5 kg a 16,0 kg para as classificadas como peso adequado; de 7,0 a 11,5 kg nas classificadas com sobrepeso; e de 5,0 a 9,0 kg nas obesas (71). Desta forma, tem-se uma tabela que melhor define este grupo de mulheres quanto ao estado nutricional.

```
mater$estNutri <- cut(mater$imc,
                      breaks = c(11.8, 18.5, 25, 30, 48.7),
                      labels = c("Baixo Peso", "Peso adequado", "Sobrepeso", "Obesidade"),
                      include.lowest = TRUE,
                      right = FALSE,
                      ordered_result =TRUE)

f.abs <- table (mater$estNutri)
f.rel <- round(prop.table(f.abs), 3)
f.perc <- round(f.rel*100, 2)

f.abs <- c (f.abs, sum(f.abs))
f.rel <- c (f.rel, sum (f.rel))
f.perc <- c (f.perc, sum (f.perc))

tab2 <- cbind(f.abs,
               f.rel ,
               f.perc)

tab2 <- as.data.frame(tab2)
row.names(tab2)[5] <- "Total"
colnames(tab2) <- c("FrequênciA", "Freq.Relativa", "Freq.Percentual")
tab2

##          FrequênciA Freq.Relativa Freq.Percentual
## Baixo Peso        67      0.049           4.9
## Peso adequado    791      0.578          57.8
## Sobrepeso         335      0.245          24.5
## Obesidade         175      0.128          12.8
## Total             1368      1.000         100.0
```

Colocando em um formato mais científico, tem-se uma tabela (Tabela 4) bem mais elegante sobre o estado nutricional pré-gestacional:

```
knitr::kable(tab2,
             booktabs = TRUE,
             format = 'latex',
             caption = "Estado nutricional pré-gestacional das parturientes, HGCS, 2008.",
             format.args = list(decimal.mark = ",")) %>%
kable_styling(full_width = TRUE,
              latex_options = "hold_position")
```

6.3.2 Tabelas de contingência

As tabelas de contingência, também chamadas tabelas cruzadas, são bastante usadas em estatísticas epidemiológicas para resumir a relação entre duas ou mais variáveis categóricas.

Tabela 4: Estado nutricional pré-gestacional das parturientes, HGCS, 2008.

	Frequência	Freq.Relativa	Freq.Percentual
Baixo Peso	67	0,049	4,9
Peso adequado	791	0,578	57,8
Sobrepeso	335	0,245	24,5
Obesidade	175	0,128	12,8
Total	1368	1,000	100,0

Uma tabela de contingência é um tipo especial de tabela de distribuição de frequência, onde duas variáveis são mostradas simultaneamente. Por exemplo, um pesquisador pode estar interessado em saber se o hábito de fumar na gestação aumenta o risco de o recém-nascido precisar de cuidados intensivos.

Existem duas variáveis `fumo` (fumo na gestação) e `utiNeo` (necessidade de cuidados intensivos neonatais) no banco de dados `dadosMater.xlsx`. Cada uma dessas variáveis tem duas alternativas, `sim` e `não`, por isso a tabela de cruzamento é denominada tabela de contingência 2 x 2. No arquivo, estão registradas como variáveis numéricas , 1 e 2, e devem ser transformadas para fatores (1 = sim e 2 = não)⁹, usando a função `factor()`.

```
mater$fumo <- factor (mater$fumo,
                      ordered = TRUE,
                      levels = c (1,2),
                      labels = c ("sim", "não"))
mater$utiNeo <- factor (mater$utiNeo,
                        ordered = TRUE,
                        levels = c (1,2),
                        labels = c ("sim", "não"))
```

Basta agora, usar a função `with()` junto com a função `table(variável da linha, variável das colunas)`. Por convenção, costuma-se colocar a variável explicativa ou explanatória nas linhas (`fumo`) e o desfecho nas colunas (`utiNeo`):

```
tabFumo <- with(data = mater, table(fumo, utiNeo))
tabFumo
```

```
##      utiNeo
## fumo  sim não
##   sim  71 230
##   não 204 863
```

Para ter a soma das margens, usar a função `addmargins (tabela, margin = c (1,2), FUN = sum)` do pacote `stats`, incluído na instalação básica do R. A função adiciona a soma das linhas (1) e das colunas (2) às margens da tabela (tabFumo). Para melhorar visualmente, pode-se colocar `sum` em um objeto denominado de `Total`.

```
Total <- sum
addmargins (tabFumo, margin = c(1,2), FUN = sum)

## Margins computed over dimensions
## in the following order:
## 1: fumo
## 2: utiNeo
```

⁹Poderiam ser transformados em fatores sem trocar os rótulos e manter os números 1 e 2, como se fossem palavras. O autor prefere usar nomes.

```

##      utiNeo
## fumo  sim  não  sum
##   sim    71   230  301
##   não   204   863 1067
##   sum   275  1093 1368

```

6.4 Gráficos

Para descrever os dados e visualizar o que está acontecendo, recomenda-se utilizar um gráfico adequado. O que é adequado depende principalmente do tipo de dados, bem como das características particulares do que se quer explorar. Além disso, um gráfico em um relatório sempre é um fator de “impacto”. Ou seja, pode ter um efeito positivo no leitor ou fazê-lo abandonar a leitura. Finalmente, um gráfico de frequência pode ser utilizado para ilustrar, explicar uma situação complexa onde palavras ou uma tabela podem ser confusos, extensos ou de outro modo insuficiente. Por outro lado, deve-se evitar usar gráficos onde poucas palavras expressam claramente o que se quer mostrar. Aconselha-se que, ao analisar os dados, é importante inspecioná-los como se fossem uma imagem, uma fotografia, ver como eles se parecem, qual o seu aspecto, e só então pensar em interpretar os aspectos vitais da estatística (72).

O R básico fornece uma grande variedade de funções para visualizar dados, elas de uma maneira relativamente simples permitem a construção de gráficos que facilitam a interpretação tanto de variáveis categórica como contínuas. Para gráficos mais sofisticados existe um pacote denominado `ggplot2` (73). Este pacote é uma ferramenta extremamente versátil. É um pouco mais complexo e exige mais tempo para dominá-lo, mas, uma vez que se aprenda o básico sobre ele, oferece uma estrutura extremamente flexível para exibir os dados. Inicialmente, serão usadas as funções do R básico e, posteriormente, será feita uma introdução ao `ggplot2`.

6.4.1 Gráfico de setores

Também conhecido como gráfico de *pizza*. Cada segmento (fatia) do gráfico de pizza deve ser proporcional à frequência da categoria que representa. A desvantagem do gráfico de pizza é que ele só pode representar uma variável, portanto, há necessidade de um gráfico separado para cada variável que se deseja representar. Além disso, um gráfico de pizza pode perder clareza se ele é usado para representar mais do que quatro ou cinco categorias. Na maioria das vezes, em um artigo ou relatório não há necessidade de se usar este tipo de gráfico. As tabelas são muito melhores. Segundo Edward Tufte, professor emérito de estatística, *design* gráfico e economia política na Universidade de Yale, o único gráfico pior do que um gráfico de pizza são vários deles (74)! Ele é usado mais no mundo dos negócios. Como regra, não use gráfico de pizza!

Em uma consulta, entre estudantes de Medicina, foi perguntado a sua opinião em relação a este tipo de gráfico. A pergunta feita foi: “O que você sente ao ver um gráfico de pizza em um artigo científico?” As alternativas para a resposta eram quatro (ódio, irritação, indiferença, amor). O resultado do inquérito está na Tabela 5.

Tabela 5: Sentimento dos alunos de Medicina em relação ao gráfico de pizza, UCS, 2012.

Sentimento	f	fr	fp	Fp
Odeiam	6	0,15	15	15
Não gostam	12	0,30	30	45
Indiferentes	14	0,35	35	80
Amam	8	0,20	20	100
Total	40	1,00	100	

No R base, pacote `graphics`, existe a função `pie()` para obter um gráfico de setores simples. Esta função usa os seguintes argumentos básicos, consulte a ajuda do R para outras informações:

- **x** → vetor numérico não negativo
- **labels** → caracteres que fornecem nomes para as fatias. Para rótulos vazios ou NA (após coerção para caractere), nenhum rótulo ou linha indicadora é desenhada
- **radius** → A pizza é desenhada centralizada em um quadrado cujos lados variam de -1 a +1. Se os caracteres que rotulam as fatias forem longos, pode ser necessário usar um raio menor. O padrão é 0,8.
- **density** → Densidade das linhas de sombreamento, em linhas por polegada. O padrão é NULL significa que nenhuma linha de sombreamento é desenhada. Valores não positivos de densidade também inibem o desenho de linhas sombreadas
- **col** → Vetor de cores a ser usado no preenchimento ou sombreamento das fatias. Se estiver faltando, um conjunto de 6 cores pastel é usado

Os valores da coluna de frequência absoluta (f) da Tabela 5 serão usados como o argumento x . Ele informa a área (proporção de cada fatia). Os rótulos das fatias são escritos com a função concatenar `c()`. As cores também podem ser estabelecidas com a função concatenar, mas para introduzir o pacote `RColorBrewer`, será usada 4 cores da sua paleta `RdBu`, mistura de vermelho e azul (Figura 33).

```
library(RColorBrewer)

pie(x = c(6, 12, 14, 8),
    labels = c("Odeiam", "Não gostam", "Indiferentes", "Amam"),
    col = brewer.pal(n = 4, name = "RdBu"))
```

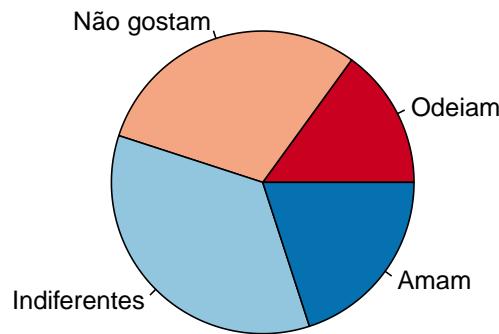


Figura 33: Gráfico de Pizza: Opinião dos estudantes de Medicina.

A função `pie3D()` permite construir um gráfico de setores em três dimensões. Para isso, há necessidade de instalar o pacote `plotrix` (75). Os argumentos são praticamente os mesmos do gráfico simples. Acrescenta-se `radius = 0.9` que muda o raio da pizza e `explode = 0.1` que determina o afastamento das fatias (0, as

mantém juntas). Além disso, como o gráfico exibe rótulos com textos muito grandes, usa-se o argumento `labelcex = 1` e coloca-se um título com o argumento `main` (Figura 34).

```
library (plotrix)

pie3D(x = c(6, 12, 14, 8),
      labels = c("Odeiam", "Não gostam", "Indiferentes", "Amam"),
      radius = 0.9,
      explode = 0.1,
      col = brewer.pal(n = 4, name = "RdBu"),
      labelcex = 1)
```

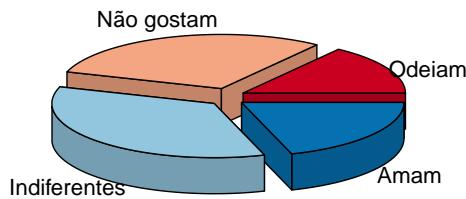


Figura 34: Gráfico de Pizza: Opinião dos estudantes de Medicina.

6.4.2 Gráfico de barras

Os gráficos de barra exibem a distribuição (frequências) de uma variável categórica através de barras verticais ou horizontais, ou sobrepostas (76).

Assim como o gráfico de setores, o gráfico de barras é utilizado para representar a frequência absoluta ou percentual de diferentes categorias. As barras são proporcionais as frequências. A forma mais simples de solicitar um gráfico de barra no R é digitar a função `barplot()` do pacote básico. Esta função é específica para desenhar gráficos de barras horizontais e verticais e usa os seguintes argumentos:

- **height** → um vetor ou matriz de valores que descreve as barras que constituem o gráfico
- **width** → especifica largura das barras, com padrão de 1, opcional
- **space** → a quantidade de espaço (como uma fração da largura média da barra) restante antes de cada barra. Pode ser fornecido como um único número ou um número por barra
- **beside** → argumento lógico para especificar se colunas devem ser mostradas lado a lado

- **col** → cores das barras componentes das barras, por padrão é usado *grey* (cinza)
- **border** → cor das bordas das barras
- ... → outros argumentos. Consulte a ajuda do R.

Para a construção do gráfico de barras simples da Figura 35, foi utilizada a variável `idadeCateg`, anteriormente criada, a partir do conjunto de dados `dadosMater.xlsx`.

```
barplot(table(mater$idadeCateg))
```

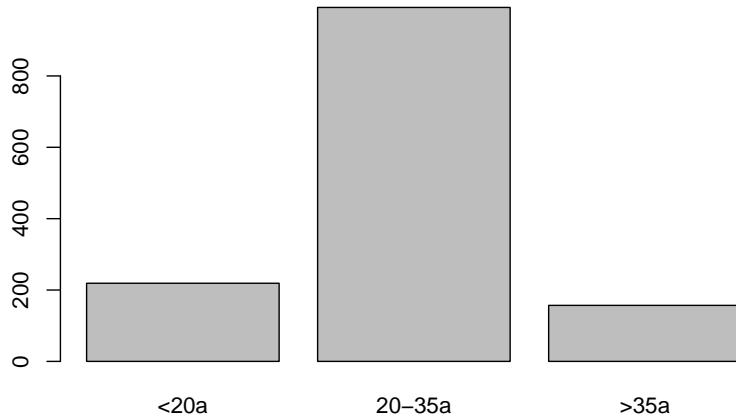


Figura 35: Gráfico de barra simples.

Observando a Figura 35, verifica-se que não existem rótulos nos eixos *x* e *y* e o eixo *y* tem um tamanho inferior a barra mais alta. Estes e outros problemas podem ser resolvidos modificando-se ou acrescentando outros argumentos na função `barplot()`. Existem vários argumentos e para conhecê-los melhor pesquise no Help do RStudio. Em um gráfico de barra simples são suficientes as seguintes modificações que irão resultar na Figura 36:

- Para corrigir a amplitude do eixo *y*, existe o argumento `ylim = c(lim inf, lim sup)`. Na Tabela 4 ,observa-se que a frequência máxima é de 992, assim estende-se até 1000, bem próximo da frequência da categoria, acrescentando `ylim = c(0,1000)`, separado por vírgulas de outros argumentos.
- Para os rótulos se utiliza os argumentos `ylab = ("Frequência")` e `xlab = ("Faixa Etária")`. Também, pode ser incluído um título no gráfico com o argumento `main = "Título"`. Observe que os títulos estão entre aspas.
- Para modificar o tamanho das letras dos eixos *x* e *y*, que estão pouco visíveis, existe o argumento `cex.lab = 1`, que é o padrão. Para aumentar em 30%, por exemplo, usar `cex.lab = 1.3`. Os nomes tem padrão `cex.names = 1`, para modificar pode-se usar 1.3, 1.5, etc. Se nada for modificado, o R imprime o padrão.
- Para a cor das barras, use o argumento `col = ("cor")`. Escolha a cor entre as 657 opções, ou deixe o padrão cinza (*grey*). O argumento `col.axis = "cor"` controla a cor dos valores dos eixos.
- Para modificar a borda das barras que por padrão é preta, é possível mudar, usando o argumento `border = "cor"`. Sem borda basta colocar 0 (zero), no lugar da cor.

- Para colocar as barras na posição horizontal, pode ser utilizado o argumento `horiz = TRUE`. Lembrar de inverter as barras. Ou seja, a variável `x` passa a ser `y` e vice-versa.
- O argumento `las = 1` faz o texto do eixo `y` ficar horizontal
- A função `box(bty = "L")`, colocada após, é opcional, faz os eixos se encontrarem em 0.

```
barplot(table(mater$idadeCateg),
       ylim = c(0,1000),
       col= "tomato",
       border = "black",
       ylab= "Frequênciá absoluta",
       xlab = "Faixa etária",
       cex.lab = 1.2,
       las = 1)
box(bty = "L")
```

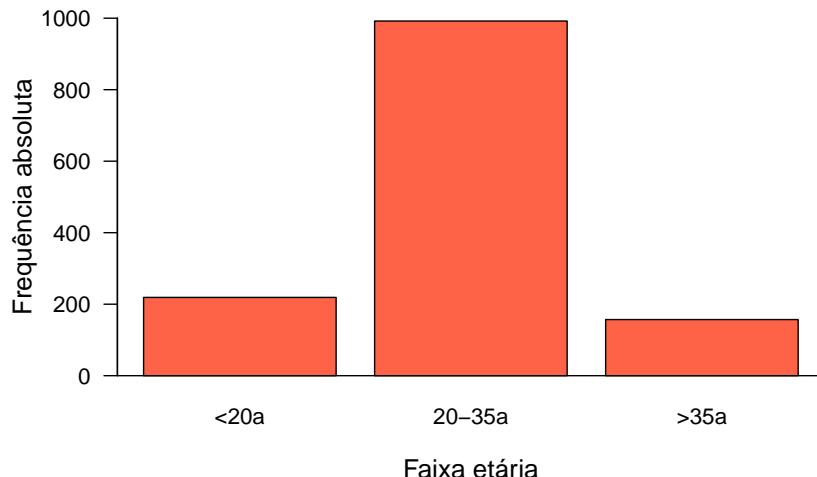


Figura 36: Gráfico de barra simples modificado.

Para que as barras fiquem horizontais como na Figura 37, usa-se o argumento `horiz=TRUE`:

```
barplot(table(mater$idadeCateg),
       xlim = c(0,1000),
       col= "steelblue",
       border = "black",
       ylab= "Faixa Etária",
       xlab = "Frequênciá absoluta",
       cex.lab = 1.2,
       horiz=TRUE)
box(bty = "L")
```

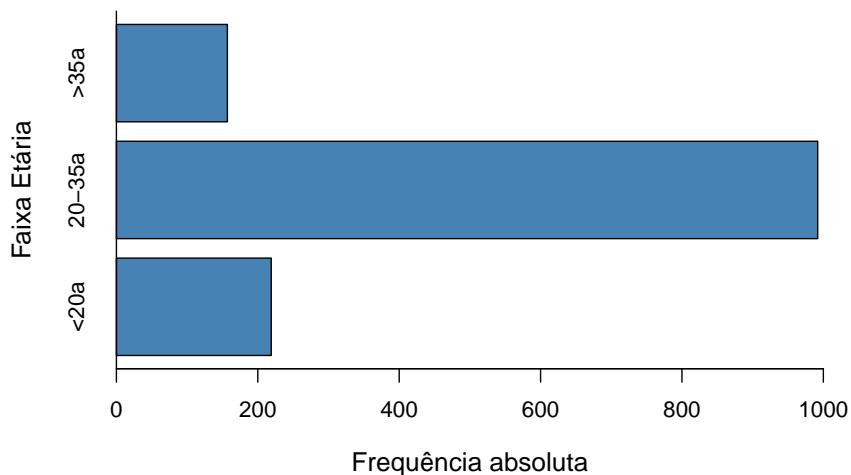


Figura 37: Gráfico com barras horizontais.

Além disso, é possível fazer outras alterações para tornar o gráfico mais informativo . Por exemplo, pode-se colocar as frequência de cada barra no topo das mesmas (Figura 38):

- **1º Passo:** Criar um gráfico de barras , colocando-o em um objeto *x*, que conterá a coordenada X do centro de cada uma das barras. Para verificar isso, basta executar o objeto *x*;
- **2º Passo:** colocar a tabela `table(mater$idadeCateg)` com um objeto *y* da classe matriz;
- **3º Passo:** usar a função `text()` para colocar os valores.

```

x <- barplot(table(mater$idadeCateg),
             ylim = c(0,1000),
             col= "springgreen",
             border = "black",
             ylab = "Frequência absoluta",
             xlab = "Faixa etária",
             cex.lab = 1.2,
             las = 1)
box(bty = "L")

y <- as.matrix(table(mater$idadeCateg))

text (x, y, labels = as.character(y), adj = c(0.5, 2), col = "black")

```

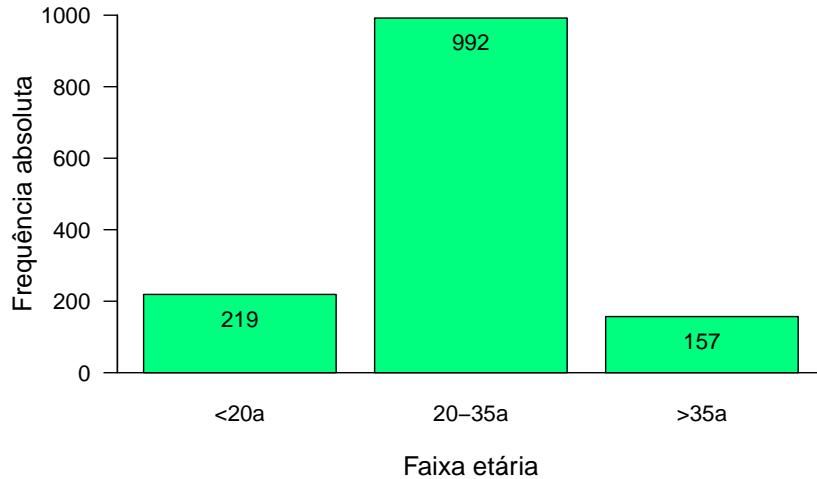


Figura 38: Gráfico de barra simples com frequências no topo.

Gráfico de barras empilhadas

Para este tipo de apresentação são utilizados, praticamente, os mesmos argumentos vistos para gerar um gráfico de barra simples. Como existem duas variáveis, há necessidade de avisar ao R como elas devem aparecer. Para isso, entra o argumento `beside = FALSE`, que informa que as barras não estarão uma ao lado da outra e sim empilhadas (Figura 39). O padrão é as barras ficarem uma ao lado da outra.

Acrescenta-se uma legenda com a função `legend()` na parte superior esquerda (`topleft`). O argumento `bty = "n"` informa que será removido o quadro ao redor da legenda e `fill = c("dimgrey", "salmon")` são as cores das barras.

As duas variáveis a serem visualizadas são o *habito tabagista* entre as puérperas de acordo com a *idade*. No conjunto de dados `dadosMater.xlsx`, o hábito tabagista está registrado na variável `fumo`, vista quando se estudou tabelas de contingência. Aqui se construirá uma tabela 3 x 2, `tabFumo2`:

```
tabFumo2 <- table(mater$fumo, mater$idadeCateg)

barplot(tabFumo2,
        beside = FALSE,
        ylim = c(0, 1000),
        xlab="Faixa Etária",
        ylab = "Frequência",
        col = c ("dimgrey", "cadetblue1"),
        cex.lab = 1,
        cex.axis = 1,
        cex.names = 1,
        las = 1)
box(bty = "L")
legend ("topleft",
        legend = c("Fumantes", "Não Fumantes"),
        fill = c("dimgrey", "cadetblue1"),
        bty="n",
        cex = 1)
```

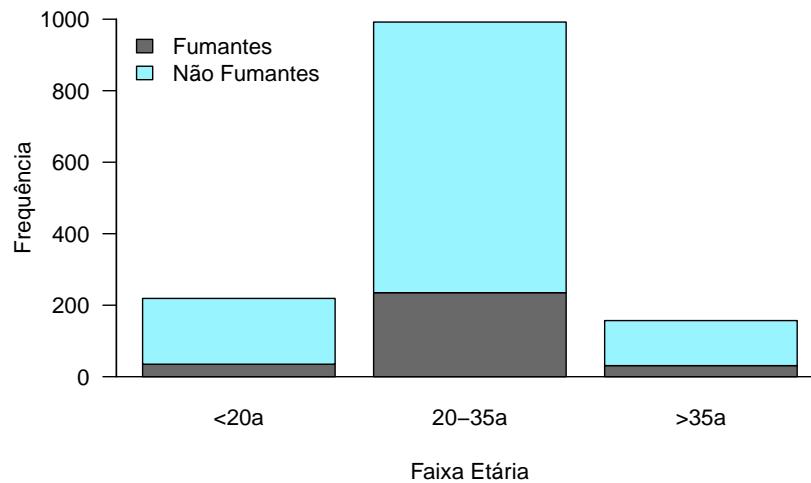


Figura 39: Gráfico de barras empilhadas.

Gráfico de barras lado a lado

É igual a anterior, apenas com o argumento `beside = TRUE` (Figura 40) .

```
barplot(tabFumo2,
        beside = TRUE,
        ylim = c(0, 1000),
        xlab="Faixa Etária",
        ylab = "Frequência",
        col = c ("dimgrey", "cadetblue1"),
        cex.lab = 1,
        cex.axis = 1,
        cex.names = 1,
        las = 1)
box(bty = "L")
legend ("topleft",
        legend = c("Fumantes", "Não Fumantes"),
        fill = c("dimgrey", "cadetblue1"),
        bty="n",
        cex = 1)
```

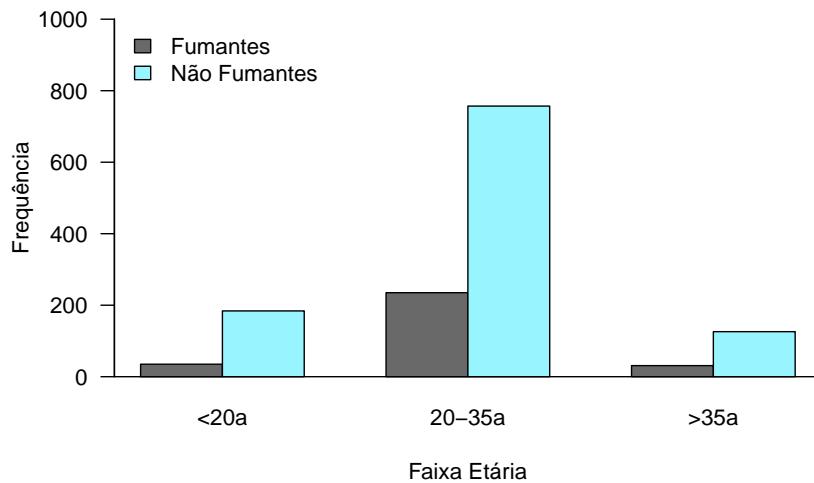


Figura 40: Gráfico de barras lado a lado

Gráfico de barras para uma variável discreta

A variável `mater$para`, número de filhos anteriores ao atual, é uma variável numérica discreta e, para representá-la, o mais adequado é usar um gráfico de barras simples Figura 41.

```
tab_filhos<- table (mater$para)

barplot (tab_filhos,
         col = "tomato",
         xlab="Número de filhos anteriores ao atual",
         ylab = "Frequência",
         ylim = c(0, 500),
         cex.lab = 1,
         cex.axis = 1,
         cex.names = 1,
         las = 1)
box(bty = "L")
```

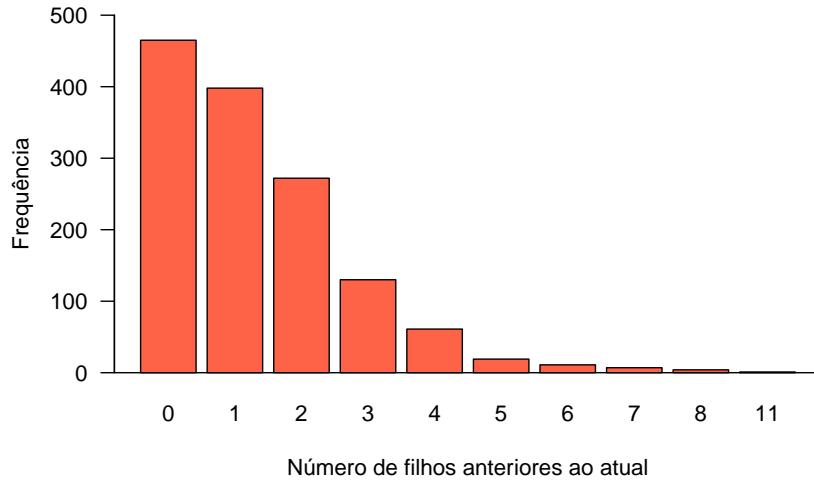


Figura 41: Gráfico de barras para uma variável discreta

6.4.3 Gráfico de barra de erro

O *gráfico de barra de erro* é um tipo de gráfico barra acrescido de uma medida de dispersão: desvio padrão, intervalos de confiança ou erro padrão. As barras de erro dão uma ideia geral de quão precisa é uma medição ou, inversamente, quão longe o valor observado está do valor verdadeiro.

Continuando a usar o arquivo `dadosMater.xlsx`, será selecionada uma amostra de recém-nascidos a termo, definido pela OMS como o nascido de 37 semanas completas a 42 semanas incompletas (259 a 293 dias). A partir destes dados, será construído um gráfico de barra de erro dos recém-nascidos do sexo masculino e feminino.

Inicialmente, deve ser instalado e carregado o pacote `Hmisc` (77), necessário para fornecer a função `errbar()` que irá construir o gráfico de barra de erro.

```
library (Hmisc)
```

A seguir, carregar o arquivo `dadosMater.xlsx` e transformar em fator a variável `sexo`, usando os rótulos `masc` e `fem`.

```
mater <- read_excel("dadosMater.xlsx")

mater$sexo <- factor(mater$sexo,
                      labels = c('masc', 'fem'))
```

Em sequência, selecionar as variáveis necessárias ao objetivo (`ig`, `pesoRN` e `sexo`), usando a função `select()`; separar com a função `filter()` os recém-nascidos a termo (`ig >= 37 & ig < 42`) e calcular, através da função `summarise()`, as medidas resumidoras, separando os grupos com a função `group_by()`, de acordo com o `sexo`. Todas essas funções são pertencentes ao pacote `dplyr`:

```
mater <- mater %>% select(ig, pesoRN, sexo) %>%
  filter(ig >= 37 & ig < 42) %>%
```

```

group_by(sexo) %>%
summarise(n = n(),
  media = mean(pesoRN, na.rm = T),
  dp = sd(pesoRN, na.rm = T),
  ep = dp/sqrt(n),
  l_inf = media - 1.96*dp,
  l_sup = media + 1.96*dp)

```

No próximo passo, constroi-se um objeto, denominado **barras**, que irá conter as médias dos pesos dos recém-nascidos masculinos e femininos, que representam a altura das barras. Usando este objeto, constroi-se um gráfico de barras que será recebido por outro objeto, **bp**.

Finalmente, coloca-se os limites inferiores e superiores para cada sexo, usando os valores calculados pela função **summarise()** que junto com o objeto **bp** constituem-se de argumentos da função **errbar()** (Figura 42). Veja maiores detalhes na ajuda do R (**?errbar**).

```

barras <- c(mater$media[1], mater$media[2])

bp <- barplot(barras,
  ylim=c(0,4200),
  ylab = "Peso do Recém-nascido (g)",
  cex.lab = 1,
  cex.axis = 0.8,
  cex.names = 1,
  space = c(0,0.5),
  names.arg=c("Meninos", "Meninas"),
  col = c("lightblue", "pink2"),
  las = 1)
box(bty = "L")

lim_inf <- c(mater$l_inf[1], mater$l_inf[2])
round(lim_inf, 2)

## [1] 2376.18 2249.69

lim_sup <- c(mater$l_sup[1], mater$l_sup[2])
round(lim_sup, 2)

## [1] 4172.08 4044.09

errbar(bp, barras, lim_inf, lim_sup, add = T, xlab = NULL)

```

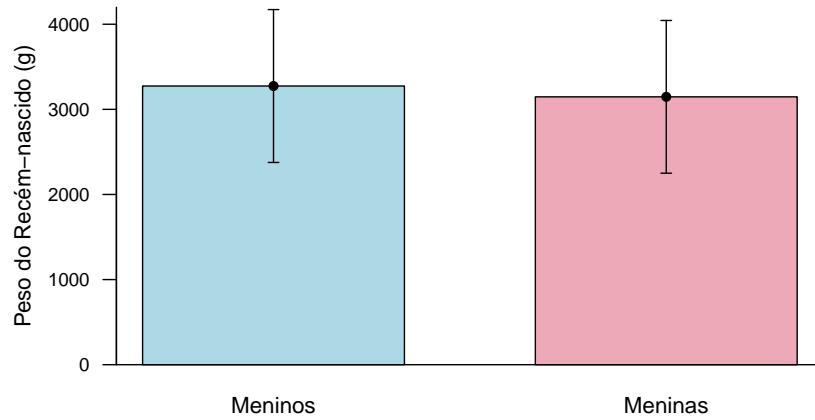


Figura 42: Gráfico de barras de erro

6.4.4 Histograma

O *histograma* é uma ferramenta gráfica que fornece informações sobre o formato da distribuição e dispersão dos dados, permitindo verificar se existe ou não simetria. É usado para dados contínuos.

No histograma as frequências observadas são representadas por intervalos de classes de ocorrência que estão no eixo *x* e a altura das barras, representando a frequência de cada intervalo, no eixo *y*. A área de cada barra é proporcional à porcentagem de observações de cada intervalo.

O R base possui uma função, denominada de `hist()` que constrói o histograma e possui vários argumentos:

- **`x`** → um vetor numérico usado na construção do histograma
- **`breaks`** → especifica o número de barras
- **`freq`** → lógico; se `TRUE` (padrão), o histograma é uma representação de frequências; se `FALSE`, densidades de probabilidade, densidade de componentes, são plotados
- **`col`** → cor a ser usada para preencher as barras. O padrão de `NULL` produz barras não preenchidas
- **`border`** → cor da borda ao redor das barras. O padrão é usar a cor de primeiro plano padrão
- **`main, xlab, ylab`** → rótulo do título, do eixo *x* e do eixo *y*. Para remover o rótulo usar `NULL`.
- **`xlim, ylim`** → limites do eixo *x* e do eixo *y*.

Histograma Simples

Os dados para a construção do histograma serão provenientes da variável `altura` do arquivo `dadosMater.xlsx`.

```
mater <- read_excel("dadosMater.xlsx")
```

Para construir o histograma básico da Figura 43, executa-se o comando:

```
hist(mater$altura)
```

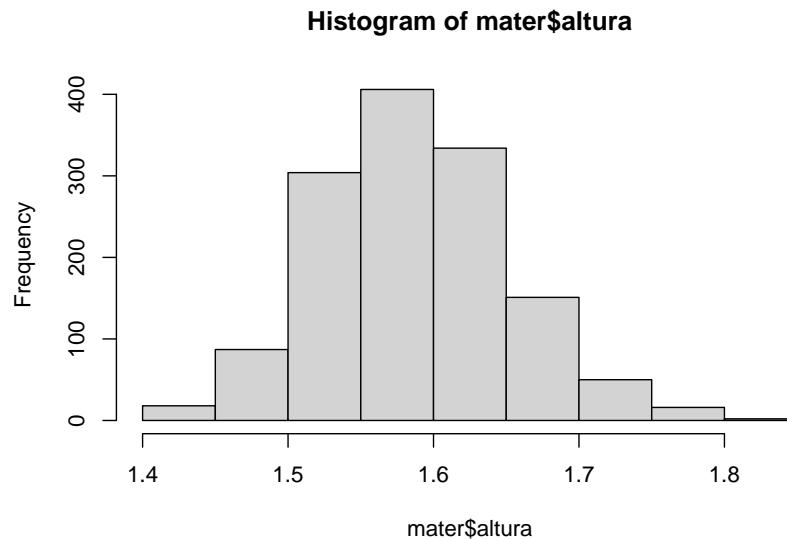


Figura 43: Histograma básico

Observando o histograma gerado, observam-se alguns problemas que devem ser melhorados para tornar a sua aparência mais agradável.

- O rótulo dos eixo x está com o nome da variável e do eixo y está em inglês;
- O título do histograma está em inglês e repete o eixo x . Pode ser removido.
- O eixo y tem um limite superior menor do que a barra mais alta;
- O gráfico está na cor cinza, que conforme o interesse pode ser modificada;
- O número de barras pode ser modificado com o argumento **breaks**. Existe uma função no R que permite calcular o número de intervalos, usando a *regra de Sturges* (`nclass.Sturges()`). Entretanto, na maioria das vezes, é o objetivo do estudo quem determina o número de barras e, também, porque nem sempre o R obedece ao argumento.

É importante saber o limite inferior e superior da variável, para construir o eixo x :

```
min(mater$altura, na.rm = TRUE)
```

```
## [1] 1.4
```

```
max(mater$altura, na.rm = TRUE)
```

```
## [1] 1.85
```

```
nclass.Sturges(mater$altura)
```

```
## [1] 12
```

Acrescentado argumentos, modifica-se o aspecto do histograma (Figura 44):

```
hist(mater$altura,
      breaks = 12,
      ylim = c(0, 450),
      xlim = c(1.4, 1.9),
      main= NULL,
      ylab = "Frequência",
      xlab = "Altura da gestante (metros)",
      col = "tomato",
      las = 1)
box(bty = "L")
```

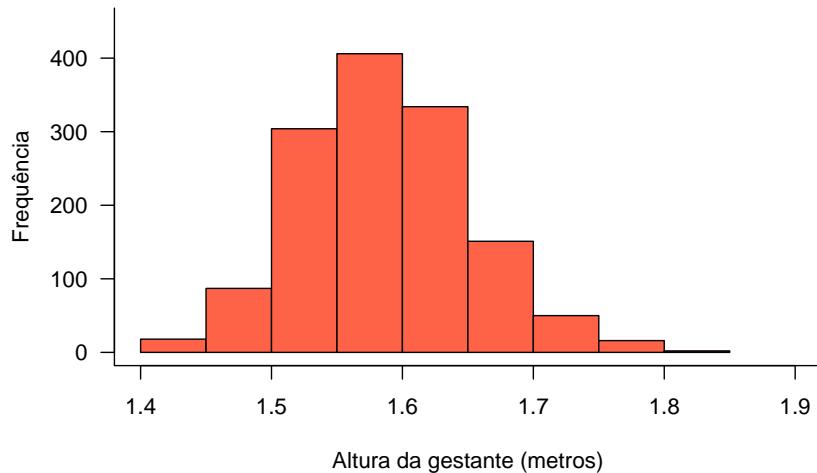


Figura 44: Histograma modificado

Observe que o formato do histograma é igual ao anterior, mudando a cor das barras, o limite do eixo y e os rótulos dos eixos. O R não modificou o número de barras. Ou seja, não obedeceu à modificação do argumento `breaks = 12`. Ele escolheu o que ele achou mais adequado!

Histograma com curva normal sobreposta

Eventualmente, para melhor comparar a distribuição dos dados, usamos uma curva normal sobreposta que servirá de indicador (Figura 45). A distribuição normal será discutida mais adiante.

1º Passo:

Construir um histograma de densidade, que é a proporção de todas as observações que se enquadram dentro do intervalo. Na função `hist()`, modificar o argumento para `freq = FALSE`.

2º Passo:

Adicionar uma curva normal ao histograma, usando a função `curve()`. Calcular antes a média e o desvio padrão da variável `mater$altura`.

```
mu <- mean(mater$altura, na.rm = TRUE)
dp <- sd(mater$altura, na.rm = TRUE)
```

```

hist(mater$altura,
      ylim = c (0, 6),
      xlim = c (1.4, 1.9),
      main= NULL,
      ylab = "Densidade",
      xlab = "Altura da gestante (metros)",
      col ="steelblue",
      freq = FALSE,
      border = "white")
box (bty = "L")

curve (dnorm (x,
              mean=mu,
              sd=dp),
       col="red",
       lty=1,
       lwd=2,
       add=TRUE)

```

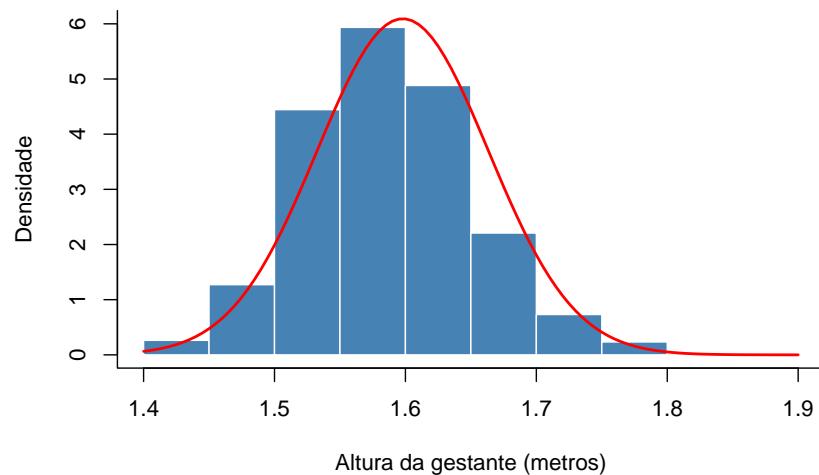


Figura 45: Histograma com curva normal sobreposta

Componentes do Histograma

Ao se criar um objeto *h* da classe `histogram` (Figura 46), pode-se verificar uma lista de componentes do mesmo.

```

h <- hist(mater$altura,
           breaks = 8,
           ylim = c (0, 450),
           xlim = c (1.4, 1.9),
           main= NULL,
           ylab = "Frequência",

```

```

xlab = "Altura da gestante (metros)",
col ="seagreen2",
freq = TRUE,
border = "white")
box (bty = "L")

```

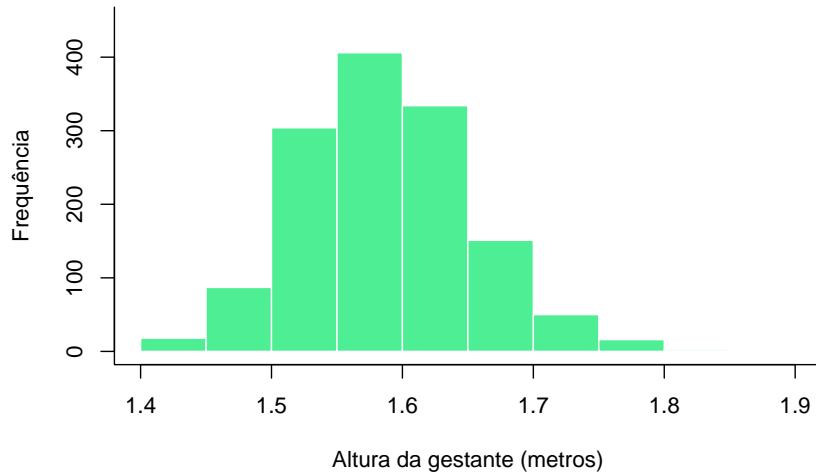


Figura 46: Histograma da altura da gestante

h

```

## $breaks
## [1] 1.40 1.45 1.50 1.55 1.60 1.65 1.70 1.75 1.80 1.85
##
## $counts
## [1] 18 87 304 406 334 151 50 16 2
##
## $density
## [1] 0.26315789 1.27192982 4.44444444 5.93567251 4.88304094 2.20760234 0.73099415
## [8] 0.23391813 0.02923977
##
## $mids
## [1] 1.425 1.475 1.525 1.575 1.625 1.675 1.725 1.775 1.825
##
## $xname
## [1] "mater$altura"
##
## $equidist
## [1] TRUE
##
## attr(,"class")
## [1] "histogram"

```

Estes componentes podem ser usados para outras análises.

Construção de um histograma usando os componentes

Pode-se colocar os valores correspondentes às barras usando os componentes do histograma (Figura 47).

```
hist(mater$altura,
      breaks = 8,
      ylim = c(0, 450),
      xlim = c(1.4, 1.9),
      main= NULL,
      ylab = "Frequênciac",
      xlab = "Altura da gestante (metros)",
      col = "salmon")
box (bty = "L")

text (h$mid, h$counts, labels = h$counts, adj= c(0.5, -0.5))
```

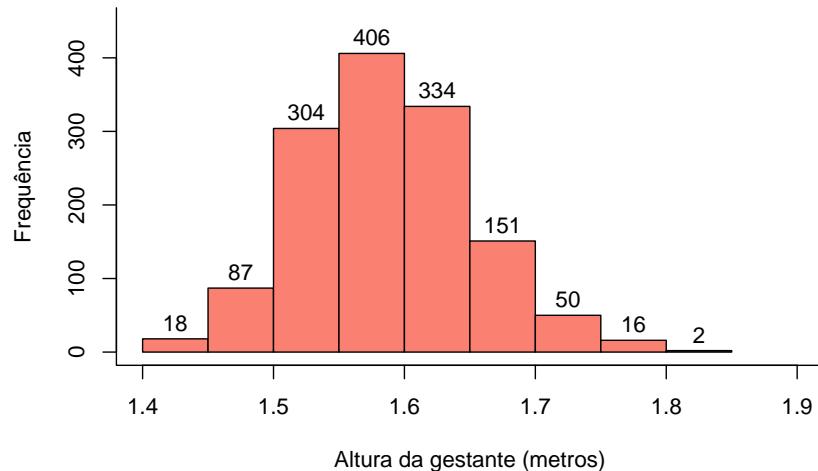


Figura 47: Histograma com frequência no topo

6.4.5 Boxplot

O *boxplot* descreve a distribuição de uma variável contínua exibindo o resumo de cinco números: mínimo, 1º quartil (percentil 25), mediana (percentil 50), 3ª quartil (percentil 75) e máximo (Figura 48). Pode também apresentar observações atípicas (*outliers*), valores fora do intervalo de $\pm 1,5$ o intervalo interquartil, em geral, representados por (o). Valores que estão acima ou abaixo de 3 vezes o IIQ são considerados extremos, representados por (*).

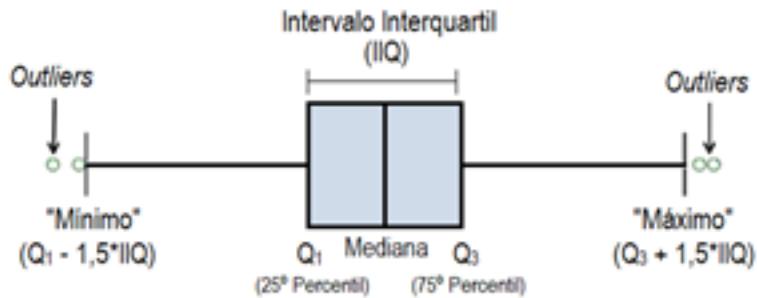


Figura 48: Boxplot

Continuando a usar o arquivo `dadosMater.xlsx`, será selecionada uma amostra de recém-nascidos a termo da mesma maneira como foi feito para construção do gráfico de barra de erro.

```
mater <- read_excel("dadosMater.xlsx")

mater$sexo <- factor(mater$sexo,
                      labels = c('masc', 'fem'))
```

O R possui uma função no pacote básico denominada `boxplot()` que constrói o gráfico da Figura 49.

```
boxplot (mater$pesoRN)
```

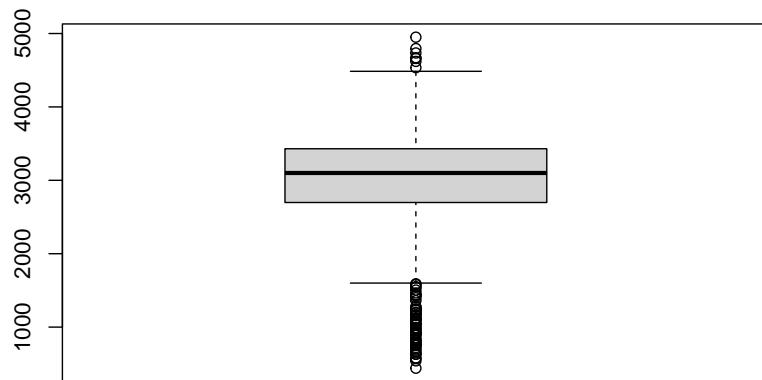


Figura 49: Boxplot simples

Este boxplot pode ser modificado (Figura 50), alterando alguns argumentos como colocação de um título no gráfico, e rótulos nos eixos e mudança na cor. Os argumentos `cex.lab`, `cex.axis` e `cex.names` estabelecem o tamanho das fontes. Por exemplo, para aumentar em 20%, usamos 1.2.

```

boxplot (mater$pesoRN,
         col = "lightblue2",
         main = "RN a termo",
         ylab = "Peso do Recém-nascido (g)",
         border = "black",
         cex.lab = 1,
         cex.axis = 1,
         cex.names = 1,
         las = 1)

```

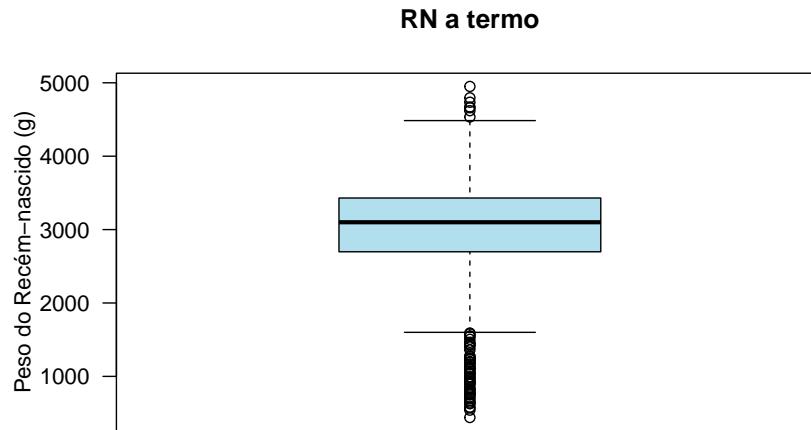


Figura 50: Boxplot modificado

Estatísticas do boxplot

A função `boxplot.stats()` do pacote `grDevices` fornece as estatísticas do boxplot, facilitando a interpretação do mesmo, de modo semelhante ao visto para o histograma.

```
boxplot.stats (mater$pesoRN)
```

```

## $stats
## [1] 1600.0 2697.5 3100.0 3430.0 4485.0
##
## $n
## [1] 1368
##
## $conf
## [1] 3068.709 3131.291
##
## $out
## [1] 1035 1580 1440 750 1030 1098 1240 785 440 810 540 1075 1505 1040 830
## [16] 1270 580 920 900 814 1195 915 1280 570 1050 765 1195 930 1160 1240
## [31] 850 1200 1550 4735 4950 4535 4670 1425 4660 4795 890 1110 700 670 1360

```

```
## [46] 1120 630 955 770 1160 1232 980 1240 1590 650 630 750 980 720 1110
## [61] 920 1105 995 1170 1465 1400 1440 1245 1545 4620
```

- \$stats = é o resumo dos 5 números: mínimo, percentil 25, mediana, percentil 75 e máximo
- \$n = nº de obs;
- \$conf = limite inf/sup do entalhe se houver;
- \$out = são os outliers

Múltiplos boxplots

Os boxplots são muito usados na comparação de grupos. A necessidade mais comum é ordenar as categorias de acordo com o aumento da mediana, mas isto é opcional. Permite identificar rapidamente qual grupo tem o maior valor e como as categorias são classificadas (Figura 51).

```
boxplot (mater$pesoRN ~ mater$sexo,
          col = c("lightblue2", "pink"),
          ylab = "Peso do Recém-nascido (g)",
          xlab = "Sexo",
          ylim = c(1000, 5000),
          border = "black",
          cex.lab = 1,
          cex.axis = 1,
          cex.names = 1,
          las = 1)
```

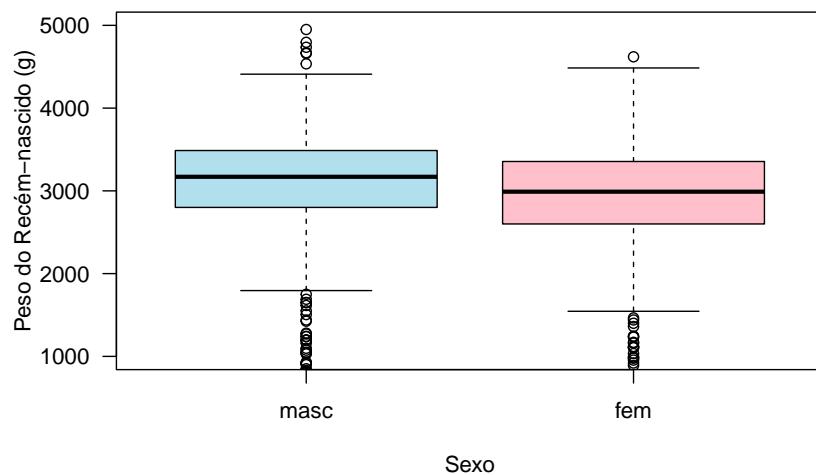


Figura 51: Múltiplos boxplots

Pode-se fazer um entalhe (*notch*) que podem ser interpretados como um intervalo de comparação em torno dos valores medianos (Figura 52). É calculado pela fórmula : $\text{mediana} \pm 1.57 \times \text{IIQ}/\sqrt{n}$. No nosso exemplo, observe que o entalhe nos meninos está um pouco acima do das meninas..

```
boxplot (mater$pesoRN ~ mater$sexo,
         col = c("lightblue2", "pink"),
         ylab = "Peso do Recém-nascido (g)",
         xlab = "Sexo",
         ylim = c(1000, 5000),
         border = "black",
         cex.lab = 1,
         cex.axis = 1,
         cex.names = 1,
         las = 1,
         notch = TRUE)
```

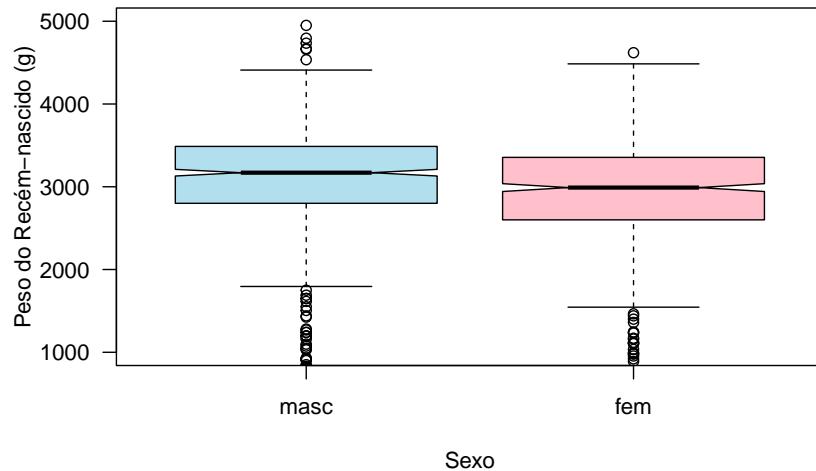


Figura 52: Boxplots com entalhes

Boxplots com stripcharts

A função `stripchart()` permite criar um gráfico de dispersão unidimensional sobre o boxplot (Figura 53). Você também pode personalizar o símbolo (pontos) para criar o gráfico, a largura da linha e sua cor com os argumentos `pch`, `lwd` e `col`, respectivamente. Alguns símbolos, como `pch = 21` a `25` permitem que você modifique a cor de fundo do símbolo com o argumento `bg`. O argumento `vertical = TRUE`, coloca os pontos na vertical sobreposto ao boxplot, quando o argumento `add = TRUE`. O argumento `cex = 0.3` é o tamanho dos pontos e `method = "jitter"`, espalha os pontos.

```
boxplot (mater$pesoRN ~ mater$sexo,
         col = c("lightblue2", "pink"),
         ylab = "Peso do Recém-nascido (g)",
         xlab = "Sexo",
         border = "black",
         cex.lab = 1,
         cex.axis = 1,
         cex.names = 1,
         pch = 20,
         cex = 0.8,
```

```

    las = 1,
    outline = TRUE)

stripchart(mater$pesoRN ~ mater$sexo,
           method = "jitter",
           main=NULL,
           col = c("blue", "red"),
           vertical=TRUE,
           pch=16,
           cex = 0.3,
           add = TRUE)

```

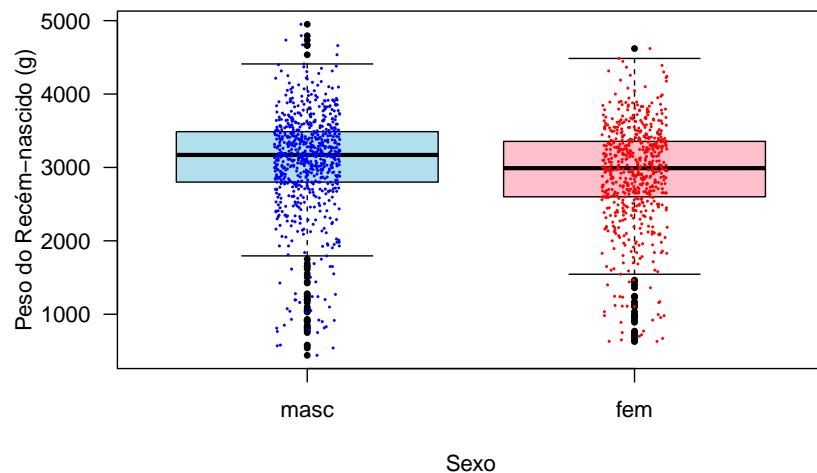


Figura 53: Boxplots com dispersão unidimensional

Boxplots horizontais

Para criar um boxplot horizontal (Figura 54), usamos o argumento `horizontal = TRUE` e invertemos os rotulos dos eixos *x* e *y*.

```

boxplot (mater$pesoRN ~ mater$sexo,
         col = c("lightblue2", "pink2"),
         xlab = "Peso do Recém-nascido (g)",
         ylab = "Sexo",
         horizontal = TRUE,
         border = "black",
         cex.lab = 1,
         cex.axis = 1,
         cex.names = 1,
         pch = 20,
         cex = 0.8)

```

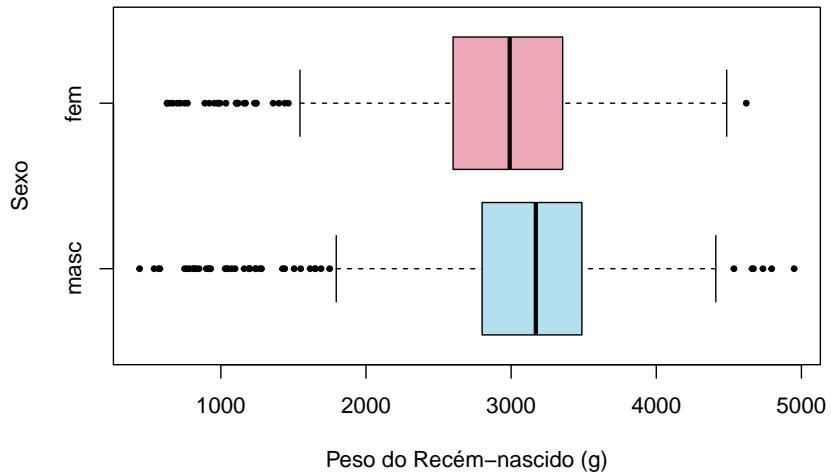


Figura 54: Boxplots horizontais

6.4.6 Gráfico de Dispersão

Um gráfico de dispersão (*Scatterplot*) exibe a relação entre duas variáveis numéricas. Cada ponto representa uma observação. Suas posições nos eixos x (horizontal) e y (vertical) representam os valores das duas variáveis.

O R Base é uma boa opção para construir um gráfico de dispersão, usando a função `plot()`. Ambas as variáveis numéricas do banco de dados devem ser especificadas nos argumentos x e y .

Será construído um gráfico de dispersão (Figura 55) do comprimento e o peso dos recém-nascidos a termo. Com o conjunto de dados `dadosMater.xlsx`, usado até aqui, selecionamos as variáveis `compRN`, `pesoRN`, `sexo` e `ig`, incluindo apenas os neonatos a termo ($37 \leq \text{idade gestacional} < 42$ semanas)

```
mater <- read_excel("dadosMater.xlsx")

mater$sexo <- factor(mater$sexo,
                      labels = c('masc', 'fem'))

mater <- mater %>% select(ig, pesoRN, compRN, sexo) %>%
  filter(ig >= 37 & ig < 42)

plot (x = mater$compRN,
      y = mater$pesoRN,
      ylab = "Peso de Recém-nascido (g)",
      xlab = "Comprimento do Recém-nascido (cm)",
      cex.axis = 0.8,
      las = 1)
```

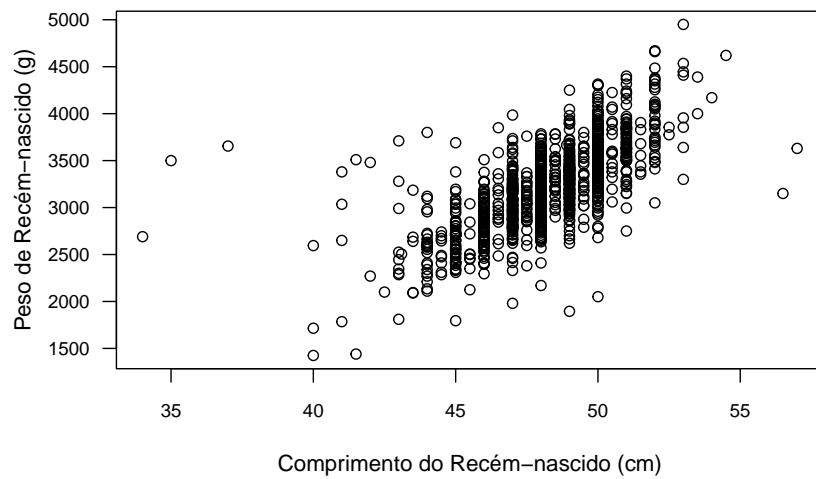


Figura 55: Gráfico de dispersão

Este mesmo gráfico pode ser obtido, usando uma fórmula $y \sim x$ e acrescentando o argumento `bty = "L"` (Figura 56). Este argumento permite personalizar a caixa ao redor do gráfico.

- **o**: caixa completa (parâmetro padrão),
- **n**: sem caixa
- **7**: superior + direita
- **L**: inferior + esquerda
- **C**: superior + esquerda + inferior
- **U**: esquerda + inferior + direita

```
plot (pesoRN ~ compRN,
      data = mater,
      ylab = "Peso de Recém-nascido (g)",
      xlab = "Comprimento do Recém-nascido (cm)",
      cex.axis = 0.8,
      las = 1,
      bty = "L")
```

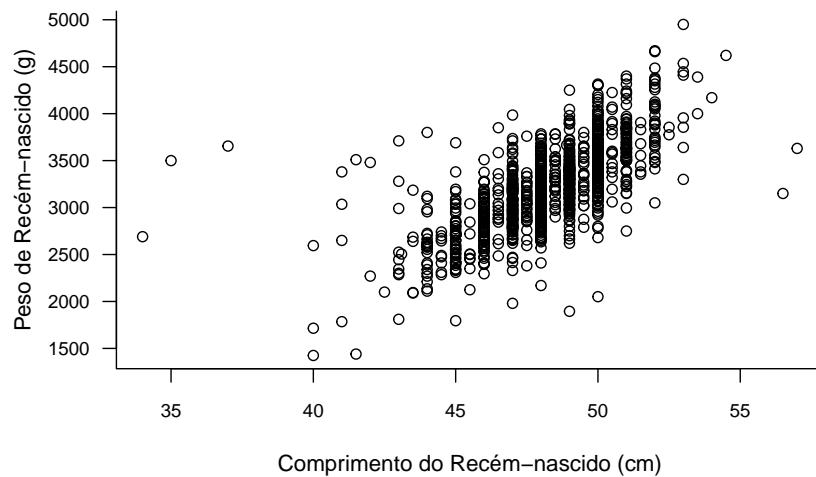


Figura 56: Gráfico de dispersão

Como em qualquer outro gráfico, este também pode ser melhorado em seu aspecto, tornando os pontos sólidos e coloridos. O argumento `pch` estabelece o tipo de pontos (Figura 57).

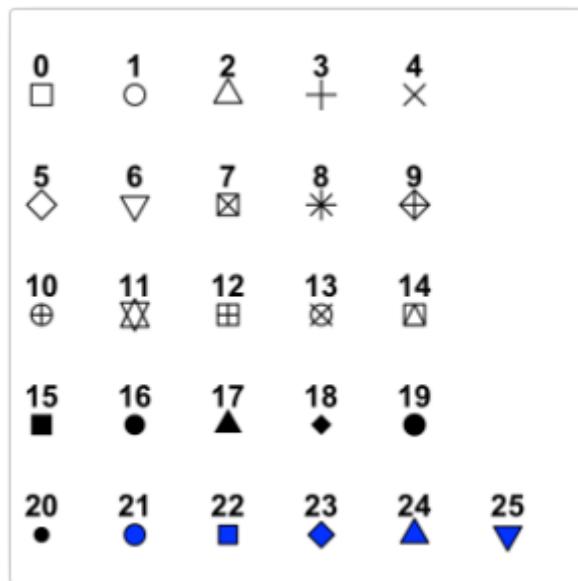


Figura 57: Argumento pch

Além disso, como os pontos estão aglomerados, devido a quantidade, é possível tentar espalhá-los, usando a função `jitter()` na variável `compRN` (Figura 58). O argumento 10 é variável e significa o grau de espalhamento:

```
plot (jitter(mater$compRN, 10),
      mater$pesoRN,
```

```

col = "steelblue",
ylab = "Peso de Recém-nascido (g)",
xlab = "Comprimento do Recém-nascido (cm)",
las = 1,
bty = "L",
pch = 19,
cex = 1,
cex.lab = 1.1,
cex.axis = 0.8)

```

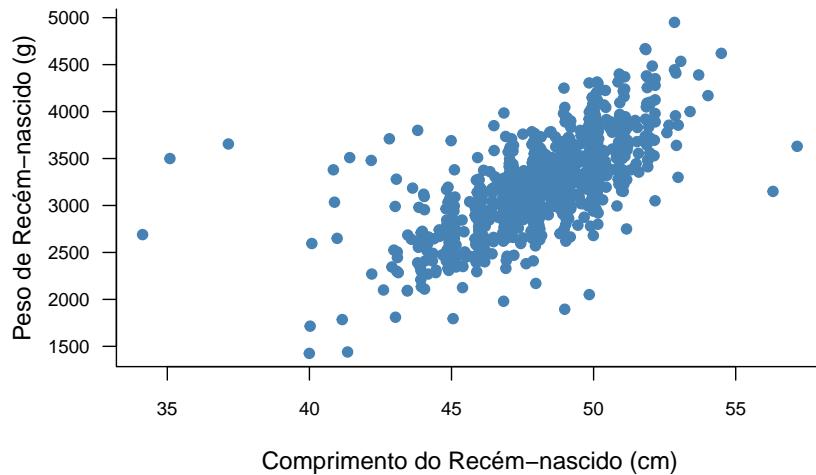


Figura 58: Gráfico de dispersão com *jitter*

Mapeamento dos pontos de acordo com uma variável categórica

Inicialmente, será criado um vetor para representar as cores, de acordo com o sexo (meninos = azul; meninas = vermelho). Usa-se a função `unclass()` para discriminar os sexos (Figura 59). Acrescenta-se uma legenda para ilustrar a separação.

```

cores <- c("dodgerblue3", "tomato")

plot(x = jitter(mater$compRN, 10),
      y = mater$pesoRN,
      bg = cores[unclass(mater$sexo)],
      ylab = "Peso de Recém-nascido (g)",
      xlab = "Comprimento do Recém-nascido (cm)",
      las = 1,
      bty = "L",
      cex = 1.5,
      pch=21,
      cex.lab = 1,
      cex.axis = 0.8)

legend (legend = c("Meninos", "Meninas"),

```

```

fill = cores,
bty="n",
cex = 1,
"topleft")

```

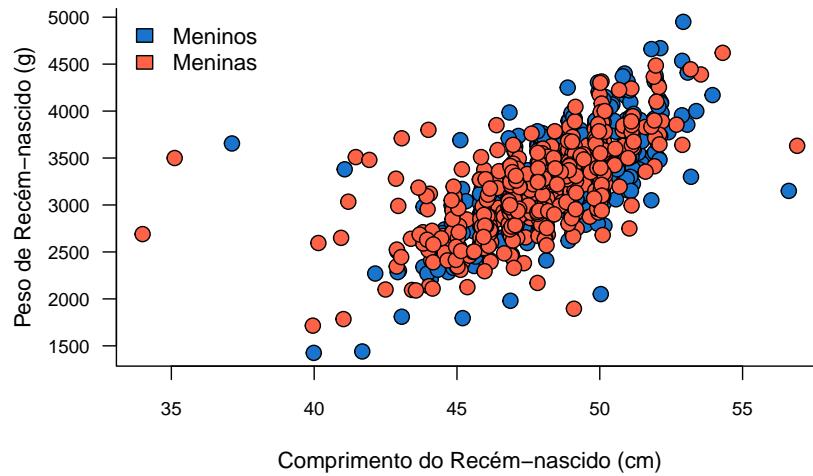


Figura 59: Mapeamento dos pontos de acordo com uma variável categórica

Adição da reta de ajuste

Uma linha reta de ajuste dos dados (Figura 60) pode ser acrescentada usando a função `abline ()`, associada a função `lm ()`. Um modelo típico `lm` (*linear model*) tem o formato *resposta* (*y*) ~ *preditor* (*x*). Mais detalhes sobre o modelo de ajuste linear na regressão linear.

```

# Construção do gráfico de dispersão
plot (jitter(mater$compRN,10),
      mater$pesoRN,
      col = "gray40",
      bg = "darkturquoise",
      ylab = "Peso de Recém-nascido (g)",
      xlab = "Comprimento do Recém-nascido (cm)",
      las = 1,
      bty = "L",
      pch = 21,
      cex = 1.3,
      cex.lab = 1,
      cex.axis = 0.8)

# Criação do modelo de ajuste
modelo <- lm (mater$pesoRN ~ mater$compRN)

# Adição da reta, usando o modelo
abline (modelo,
        col="red",

```

```

lwd=2,
lty = 2)

```

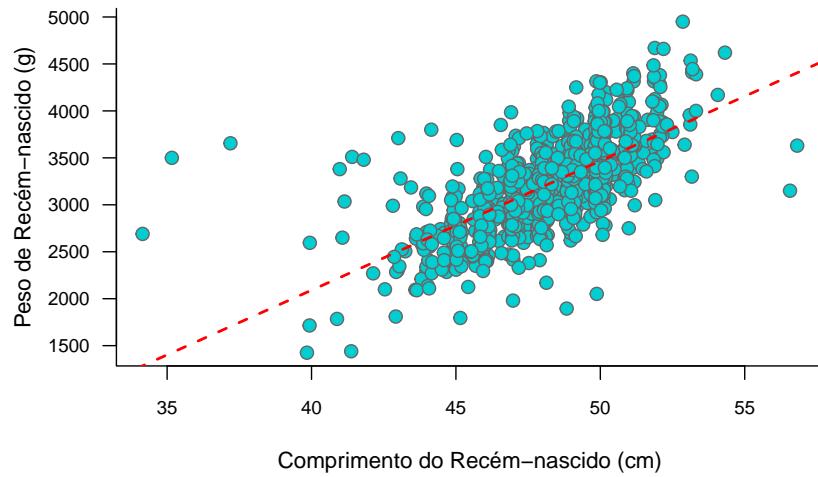


Figura 60: Gráfico de dispersão com reta de ajuste

Ao executar o modelo, se obtém os parâmetros para a construção da equação da regressão linear:

```
summary(modelo)
```

```

##
## Call:
## lm(formula = mater$pesoRN ~ mater$compRN)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1434.56  -218.40   -19.56  177.76 2097.87
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3416.451    215.821 -15.83 <2e-16 ***
## mater$compRN   137.674     4.475  30.77 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 337.7 on 1083 degrees of freedom
## Multiple R-squared:  0.4664, Adjusted R-squared:  0.4659
## F-statistic: 946.6 on 1 and 1083 DF,  p-value: < 2.2e-16

```

A equação de predição da regressão linear permite que ao conhecer o valor do comprimento é possível prever o peso do recem-nascido:

$$\hat{y} = b_0 + b_1 \times x$$

Desta forma, substituindo pelos valores contidos nas estimativas da tabela dos coeficientes do sumário do modelo, um bebê com 50 cm terá um peso de aproximadamente:

$$\hat{y} = -3416.45 + 137.67 \times 50 = 3467.05$$

OBSERVAÇÃO: Para maiores detalhes sobre os parâmetros dos gráficos no R, consulte [aqui](#).

6.5 Introdução ao ggplot2

O R tem vários sistemas para fazer gráficos e, na maioria das vezes, eles são suficientes. Entretanto, o surgimento do `ggplot2` (78) trouxe a possibilidade de serem construídos gráficos mais elegantes e versáteis. Além disso, torna o processo mais rápido, baseado uma sofisticada gramática (79).

O gráfico é construído, usando função `ggplot()`, a partir de alguns elementos básicos:

- **Dados:** os dados brutos que você deseja representar graficamente.
- **Geometria geoms:** As formas geométricas que irão representar os dados.
- **Estética aes:** Estética dos objetos geométricos e estatísticos, como posição, cor, tamanho, forma e transparência
- **Escala scales:** Mapas entre os dados e as dimensões estéticas, como intervalo de dados para plotar largura ou valores de fator para cores.
- **Transformações estatísticas stats:** resumos estatísticos dos dados, como quantis, curvas ajustadas e somas.
- **Sistemas de coordenadas coordinates Systems:** A transformação usada para mapear coordenadas de dados no plano do retângulo de dados.
- **Facetas facetting:** A organização dos dados em uma grade de gráficos.
- **Temas visuais themes:** Os padrões visuais gerais de um gráfico, como plano de fundo, grades, eixos, tipo de letra padrão, tamanhos e cores.

O número de elementos pode variar dependendo de como você os agrupa e da pergunta a ser respondida.

6.5.1 Dados

Antes de trabalhar com os dados, há necessidade de instalar e carregar alguns pacotes. O pacote `pacman` (80) será utilizado pela sua versatilidade em buscar e carregar múltiplos pacotes. Será utilizada a função `p_load()`, deste pacote:

```
if(!require(pacman)){install.packages("pacman")}

## Carregando pacotes exigidos: pacman

pacman::p_load(readxl, ggplot2, dplyr, knitr, kableExtra, ggpubr, scales,forcats)
```

Com os pacotes devidamente ativos, o conjunto de dados `dadosMater.xlsx`, já bastante conhecido, é carregado:

```
mater <- read_excel("dadosMater.xlsx")
```

Este banco de dados é bastante extenso e contém muitas variáveis que não serão usadas. Por isso, ele será reduzido, usando a função `select ()` do pacote `dplyr` para criar um novo objeto com o nome `dados`:

```

dados <- mater %>%
  select(idadeMae, anosEst, peso, renda, ig, tipoParto,
         fumo, pesoRN, compRN, sexo) %>%
  filter(ig >= 37 & ig < 42)

```

Após este processo, algumas variáveis serão transformadas:

Criação de das variáveis idadeCateg e escolaCateg

A partir da variável idadeMae e usando a função cut() será criada a variável idadeCateg:

```

dados$idadeCateg <- cut(dados$idadeMae,
                         breaks = c(13, 20, 36, 46),
                         labels = c("<20a", "20-35a", ">35a"),
                         include.lowest = TRUE,
                         right = FALSE,
                         ordered_result =TRUE)

```

A variável escolaCateg será criada da mesma maneira, a partir da variável anosest. Até 9 anos de estudos completos: *ensino fundamental*; de 10 a 12 anos, o *ensino médio* e a partir de 13 anos de estudo, o ensino superior.

```

dados$escolaCateg <- cut (dados$anosEst,
                           breaks= c (0,10,13,18),
                           right = FALSE,
                           labels = c("Fundamental",
                                     "Médio",
                                     "Superior"),
                           include.lowest = TRUE,
                           ordered_result =TRUE)

```

Transformação de variáveis numéricas em fator

As variáveis numéricas do banco que são categóricas serão modificadas para fatores:

```

dados$fumo <- factor (dados$fumo,
                      levels = c(1, 2),
                      labels = c("sim", "não"))

dados$tipoParto <- factor (dados$tipoParto,
                           levels = c(1, 2),
                           labels = c("normal", "cesareo"))

dados$sexo <- factor (dados$sexo,
                      levels = c(1, 2),
                      labels = c("masc", "fem"))

```

Desta forma, os dados tem a seguinte configuração:

```

str(dados)

## # tibble [1,085 x 12] (S3: tbl_df/tbl/data.frame)
## $ idadeMae    : num [1:1085] 28 31 27 28 18 28 22 28 25 14 ...

```

```

## $ anosEst    : num [1:1085] 6 5 8 8 7 11 6 5 9 6 ...
## $ peso       : num [1:1085] 48.5 65 60 47 65.5 72 65 74 70 56.7 ...
## $ renda      : num [1:1085] 3.13 0.72 2.41 1.69 1.93 1.92 2.65 2.53 0.48 1.92 ...
## $ ig         : num [1:1085] 37 37 37 38 39 39 39 39 39 39 ...
## $ tipoParto   : Factor w/ 2 levels "normal","cesareo": 1 2 2 1 1 2 2 1 1 1 ...
## $ fumo        : Factor w/ 2 levels "sim","não": 2 2 1 2 1 1 2 2 2 2 ...
## $ pesoRN     : num [1:1085] 3285 3100 3100 2800 3270 ...
## $ compRN     : num [1:1085] 48.5 47 47 48 49 41.5 50 48 46 50 ...
## $ sexo        : Factor w/ 2 levels "masc","fem": 1 1 1 1 1 1 1 1 1 1 ...
## $ idadeCateg : Ord.factor w/ 3 levels "<20a"<"20-35a"<...: 2 2 2 2 1 2 2 2 2 1 ...
## $ escolaCateg: Ord.factor w/ 3 levels "Fundamental"<...: 1 1 1 1 1 2 1 1 1 1 ...

```

As variáveis `idadeMae` e `anosEst` não são mais necessárias e serão removidas:

```

dados <-dados %>%
  select(-idadeMae, -anosEst)

```

Como forma de treinamento, os dados serão exibidos de uma forma, visualmente, mais elegante e em uma apresentação mais amigável (Tabela 6). A função `kable()`, do pacote `knitr`, e a função `kable_styling()` do pacote `kableExtra`, já vistas anteriormente, cumprem este papel. A função `kable()` pode usar a função `head()` embutida. Ao executar os códigos serão exibido apenas 10 linhas do banco de dados (se não for especificado, mostra apenas 6 linhas). Isto evita uma poluição visual:

Tabela 6: Dados do arquivo ‘dadosMater.xlsx’ resumidos

peso	renda	ig	tipoParto	fumo	pesoRN	compRN	sexo	idadeCateg	escolaCateg
48.5	3.13	37	normal	não	3285	48.5	masc	20-35a	Fundamental
65.0	0.72	37	cesareo	não	3100	47.0	masc	20-35a	Fundamental
60.0	2.41	37	cesareo	sim	3100	47.0	masc	20-35a	Fundamental
47.0	1.69	38	normal	não	2800	48.0	masc	20-35a	Fundamental
65.5	1.93	39	normal	sim	3270	49.0	masc	<20a	Fundamental
72.0	1.92	39	cesareo	sim	1440	41.5	masc	20-35a	Médio
65.0	2.65	39	cesareo	não	3365	50.0	masc	20-35a	Fundamental
74.0	2.53	39	normal	não	3650	48.0	masc	20-35a	Fundamental
70.0	0.48	39	normal	não	2605	46.0	masc	20-35a	Fundamental
56.7	1.92	39	normal	não	3200	50.0	masc	<20a	Fundamental

O argumento `full_width =FALSE`, reduz a largura da tabela e a `bootstrap_options` = admite vários opções além da `basic`, isoladas ou combinadas:

- `striped`: adiciona listras zebreadas à tabela;
- `hover`: adiciona cor de fundo cinza nas linhas da tabela;
- `condensed`: torna a tabela mais compacta;
- `responsive`: faz rolagem horizontal quando há menos de 768 px (20,32 cm)

6.5.2 ggplot

A sintaxe do `ggplot2` é diferente do R básico. De acordo com os elementos básicos, um `ggplot` padrão precisa de três informações que devem ser especificadas: os *dados*, a *estética* e a *geometria*. Essas são as camadas principais.

Vamos construir um `ggplot` padrão (Figura 61) que será recebido por um objeto `g`, usando `data = dados` e a estética (`aes`) usará, no eixo *x*, a variável `compRN` e, no eixo *y*, a variável `pesoRN`.

```
g <- ggplot (data = dados, aes (x = compRN, y = pesoRN))  
g
```

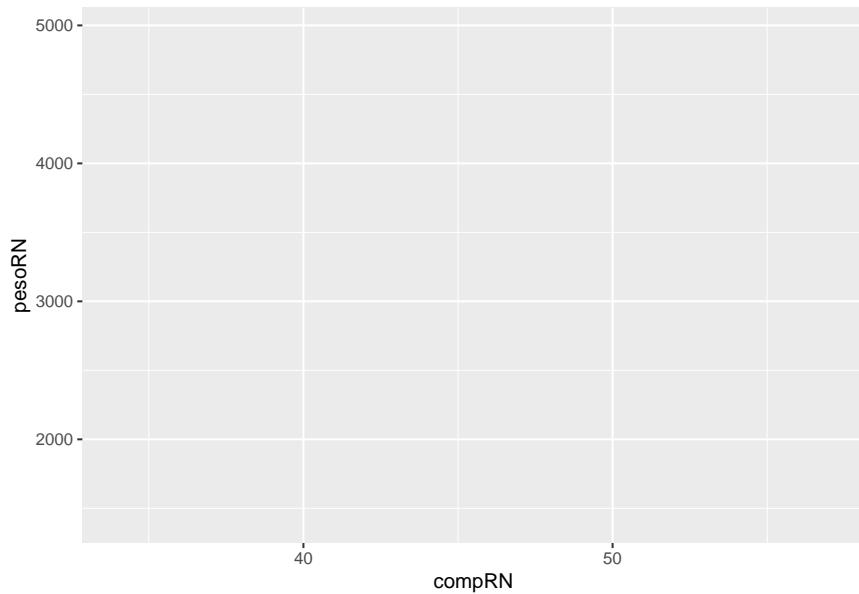


Figura 61: Gráfico ggplot padrão

A este gráfico básico “vazio” adiciona-se uma camada que especifique o tipo de geometria desejada (Figura 62). Será utilizada a `geom_point()` que retorna um gráfico de dispersão.

```
g + geom_point()
```

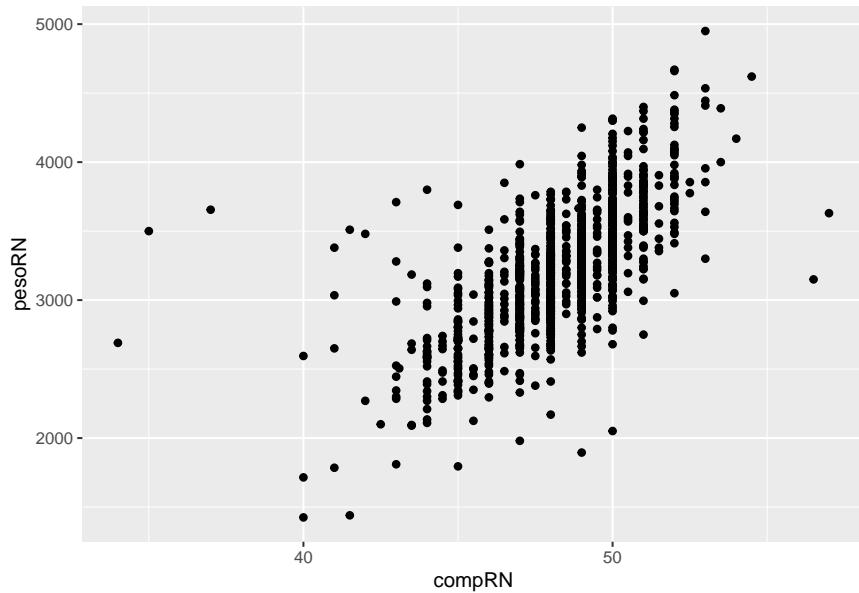


Figura 62: Gráfico de dispersão

A estética (`aes`) pode ser definida tanto na camada `ggplot` como na `geom`. Especificando no `ggplot`, esta `aes` será usada em todos os `geoms` usados. Usando no `geom`, servirá apenas para ele. No exemplo, é indiferente o local de uso da `aes`, o resultado será o mesmo, pois temos apenas um `geom`.

6.5.3 Tipos de `geoms`

Encontra-se uma grande possibilidade de geometrias, de acordo com o tipo de gráfico que será plotado. Elas podem ser visualizadas [aqui](#). Por exemplo:

Histograma

Para a construção de um histograma, usaremos a variável `pesoRN` e o `geom_histogram()`. Aqui, há necessidade apenas do eixo `x`, pois existe uma única variável onde se observa a sua distribuição (Figura 63):

```
ggplot(data = dados) +  
  geom_histogram(aes(x = pesoRN))
```

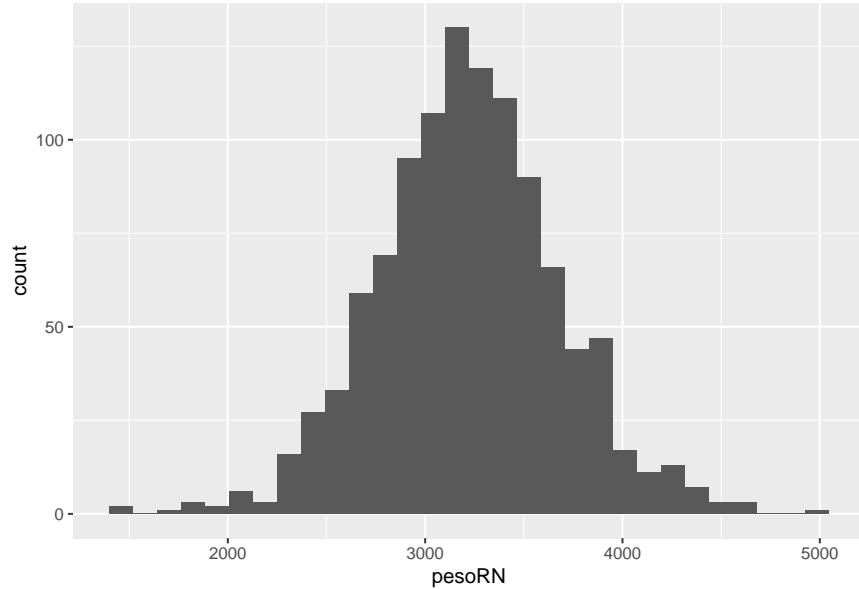


Figura 63: Histograma no 'ggplot2'

Gráfico de barras

Sera construído um gráfico de barras (Figura 64) da variável `idadeCateg`, usando o `geom_bar()`.

```
ggplot(data = dados) +  
  geom_bar(aes(x = idadeCateg, y = after_stat(count/sum(count))))
```

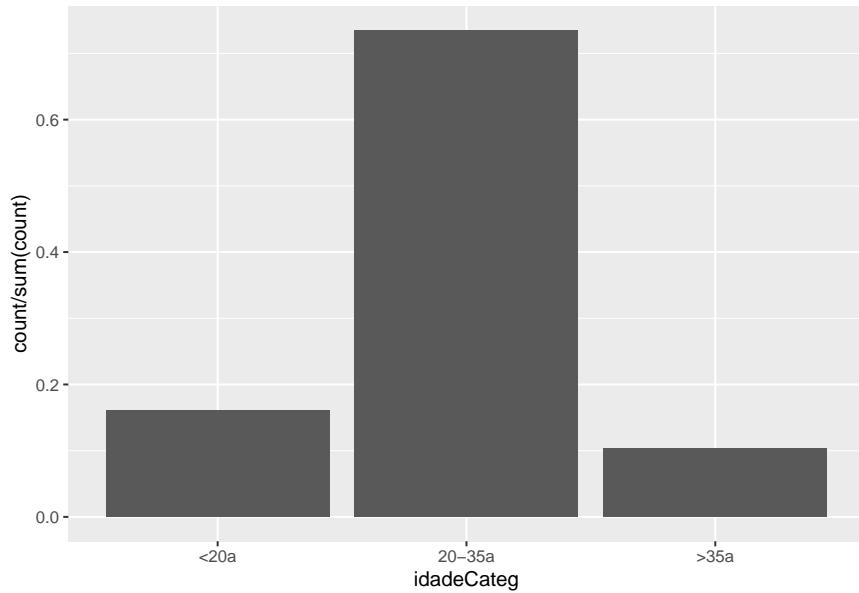


Figura 64: Gráfico de barras no 'ggplot2'

Boxplot

Com o `geom_boxplot()`, serão construídos boxplots (Figura 65) comparando os pesos dos neonatos por sexo..

```
ggplot(data = dados) +
  geom_boxplot(aes(x = sexo, y = pesoRN))
```

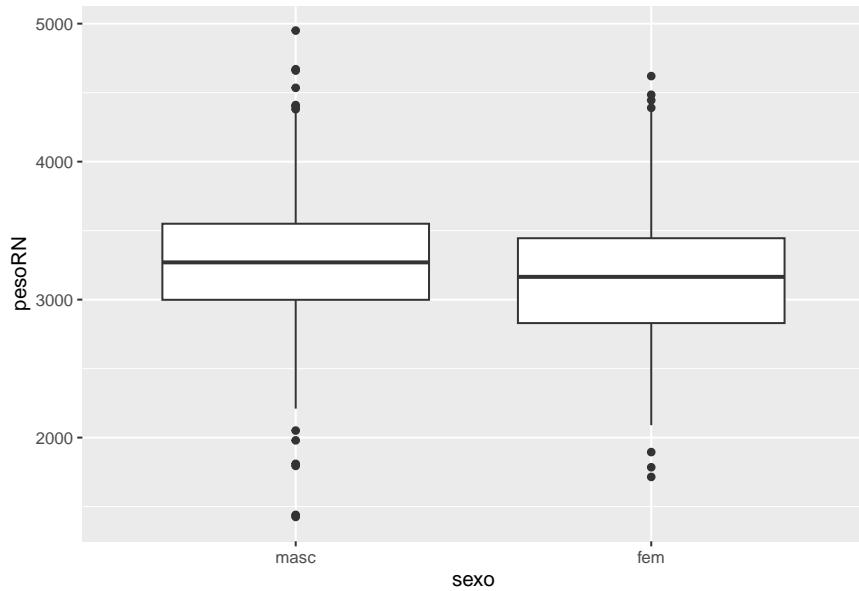


Figura 65: Boxplot no 'ggplot2'

Gráfico de linhas

Para obter os dados clique aqui e baixar para o seu diretório de trabalho o arquivo `dadosObitos.xlsx`. Este conjunto de dados é constituído pelos óbitos por COVID-19 no Rio Grande do Sul, 2020-2022.

Crie o objeto `obitos` para receber o banco de dados, a partir do diretório de trabalho.

```
obitos <- read_excel("dadosObitos.xlsx")
str(obitos)

## # tibble [25 x 2] (S3:tbl_df/tbl/data.frame)
## $ data : POSIXct[1:25], format: "2020-03-01" "2020-04-01" ...
## $ obitos: num [1:25] 4 60 182 440 1391 ...
```

Para a construção do gráfico de linha (Figura 66), será usado o `geom_line()`.

```
ggplot(data = obitos) +
  geom_line(aes(x = data, y = obitos))
```

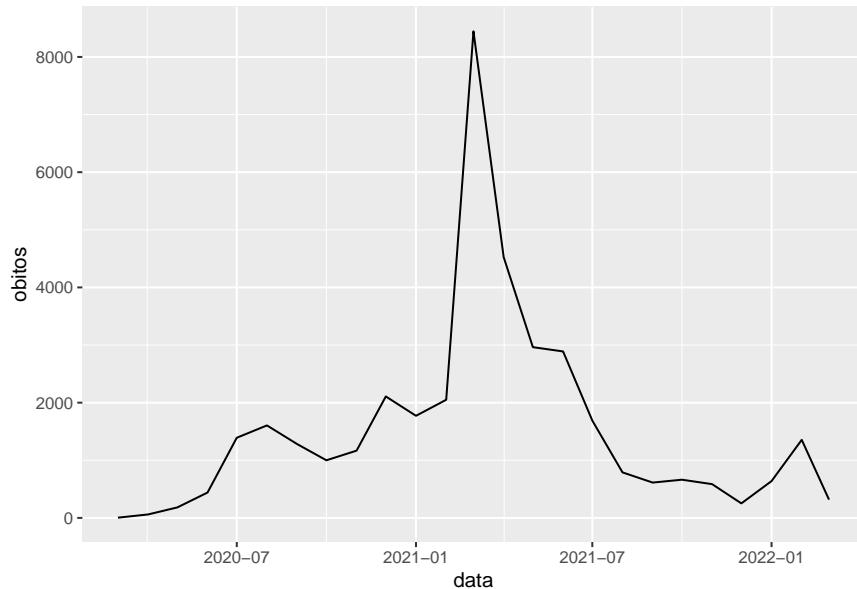


Figura 66: Gráfico de linha no 'ggplot2'

6.5.4 Modificando argumentos do geom

Agora, serão modificados alguns argumentos no `geom`. Cada tipo de `geom` possibilita alterações específicas.

Inicialmente, serão realizadas modificações no gráfico de dispersão, construído acima com o `geom_point`, modificando a cor, de acordo com o sexo do recém-nascido (Figura 67):

```
ggplot(data = dados) +
  geom_point(aes(x = compRN, y = pesoRN,
                 color = sexo))
```

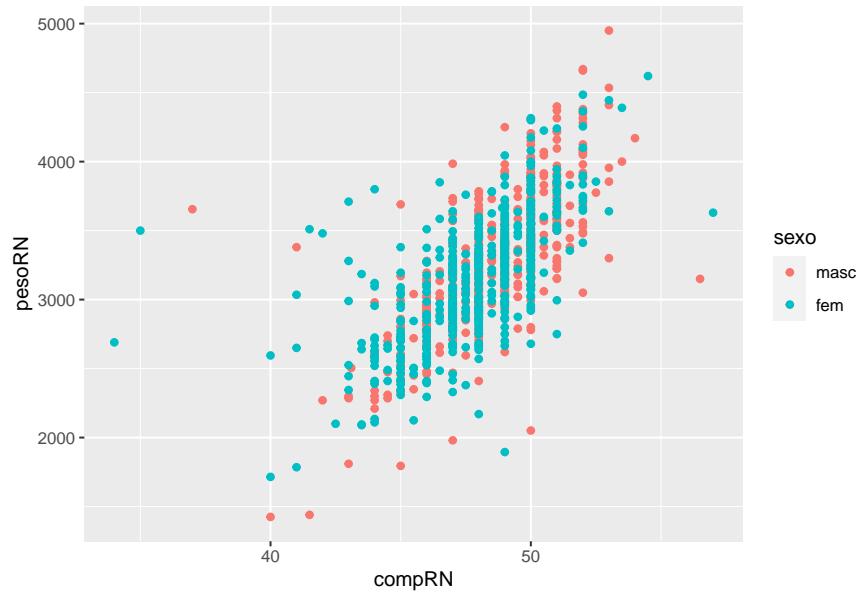


Figura 67: Gráfico de dispersão modificado com cores de acordo com o sexo

Como o argumento `cor` foi colocado dentro da `aes`, ele será definido por uma variável, no caso `sexo`. Neste caso, cada um dos sexos serão representados por pontos coloridos diferentes. A escolha da cor foi automática pelo `ggplot2` entregando duas cores conforme o padrão da sua paleta.

O tamanho dos pontos pode ser alterado com o argumento `size` (Figura 68).

```
ggplot(data = dados) +
  geom_point(aes(x = compRN, y = pesoRN,
                 color = sexo,
                 size = sexo))
```

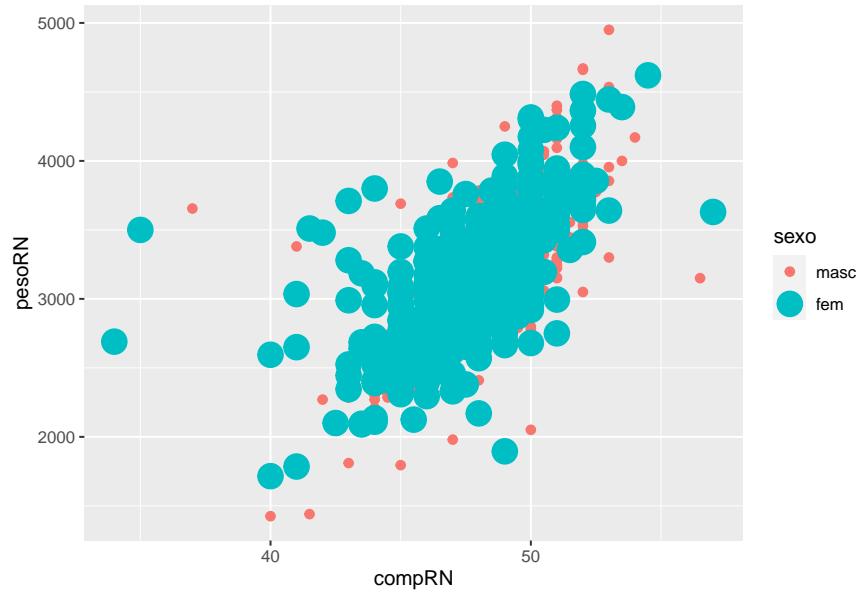


Figura 68: Gráfico de dispersão com cores e tamanhos diferentes dos pontos, de acordo com o sexo

Ficou meio bagunçado, pois os pontos se sobrepõem e dificulta a visualização e inclusive o R libera um aviso informando que isto não é recomendado.

O formato pode ser modificado com o argumento `shape`(Figura 69):

```
ggplot(data = dados) +  
  geom_point(aes(x = compRN, y = pesoRN,  
                 color = sexo,  
                 shape = sexo,  
                 size = sexo))
```

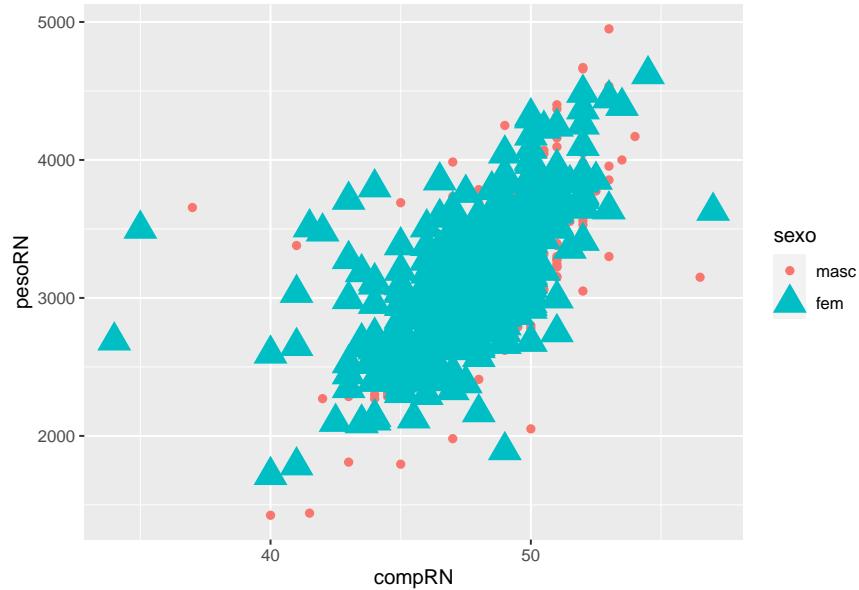


Figura 69: Gráfico de dispersão com formatos dos pontos diferentes, de acordo sexo

Agora, ficou muito pior! Não teria, logicamente, muito sentido modificar tudo ao mesmo tempo, com a mesma variável. Foi realizado, aqui, apenas para mostrar a possibilidade do `ggplot2`.

Estes argumentos foram modificados dentro da estética (`aes`), usando o nome da variável. Entretanto, também é possível colocar os argumentos fora da `aes`. Nesta situação (Figura 70), colocando a cor fora da `aes`, há necessidade de escolher uma determinada cor para os pontos.

```
ggplot(data = dados) +  
  geom_point(aes(x = compRN,  
                 y = pesoRN,  
                 shape = sexo),  
             color = "steelblue",  
             size = 3)
```

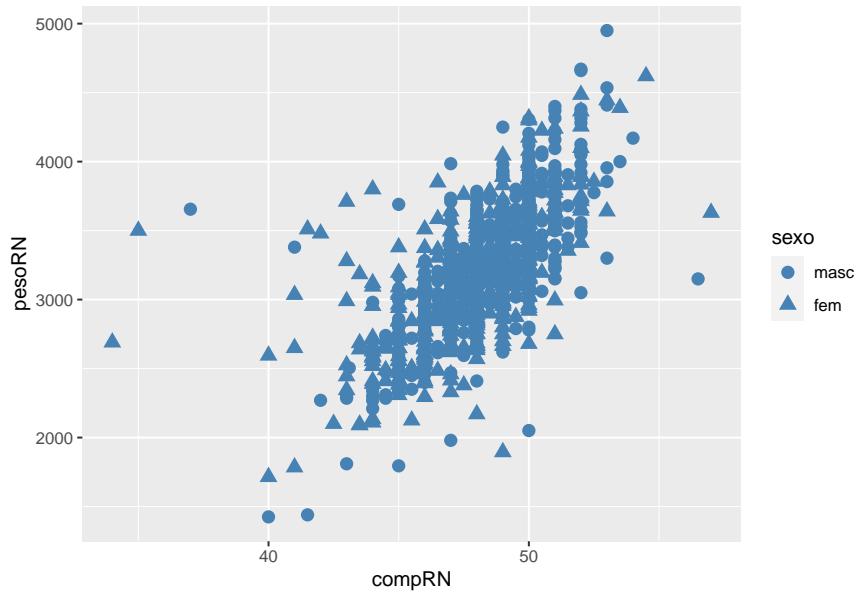


Figura 70: Gráfico de dispersão com alteração do formato

O formato (`shape`) pode ser colocado fora do `aes` (Figura 71) e [aqui](#) se encontram vários formatos que podem ser usados no `ggplot2`.

```
ggplot(data = dados) +
  geom_point(aes(x = compRN,
                 y = pesoRN),
             color = "tomato",
             size = 2,
             shape = 25)
```

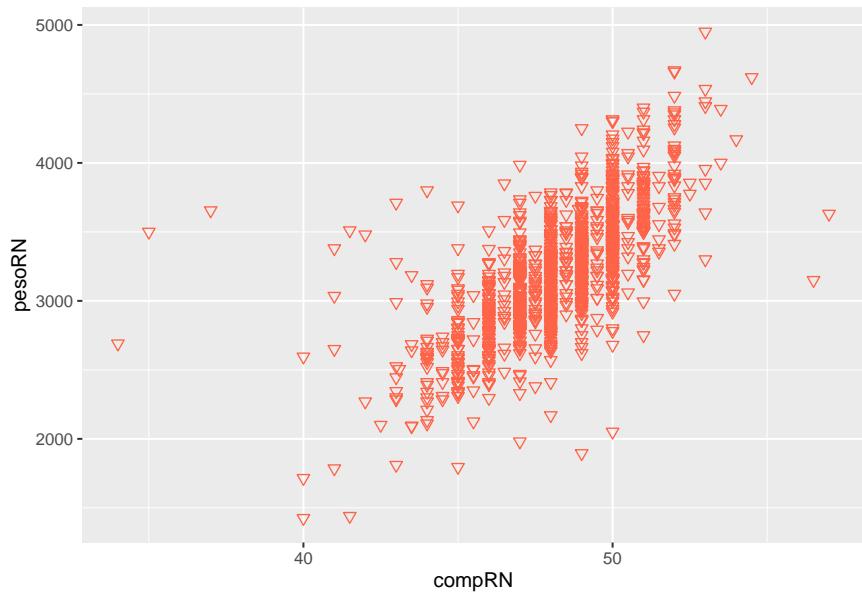


Figura 71: Gráfico de dispersão por sexo com alteração do argumento ‘color’ e ‘shape’

6.5.5 Reta de ajuste em um gráfico de dispersão

Acrescenta-se uma nova camada , chamada `geom_smooth`, além da `geom_point`. Os argumentos são o `method = "lm"` que irá ajustar uma reta aos pontos. Também é possível colocar um intervalo de mais ou menos um erro padrão para a reta com o argumento `se = TRUE`. No exemplo (Figura 72), foi usado `se = FALSE`. Além disso, foi solicitado que cor da reta seja preta (`color = "black"`), reduzido o seu tamanho (`size = 0.5`) e estabelecido que a reta seja tracejada (`linetype = "dashed"`):

```
ggplot(data = dados, aes(x = compRN, y = pesoRN)) +  
  geom_point(color = "tomato",  
             size = 3) +  
  geom_smooth(method = "lm",  
              se = FALSE,  
              color = "black",  
              linewidth = 0.5,  
              linetype = "dashed")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

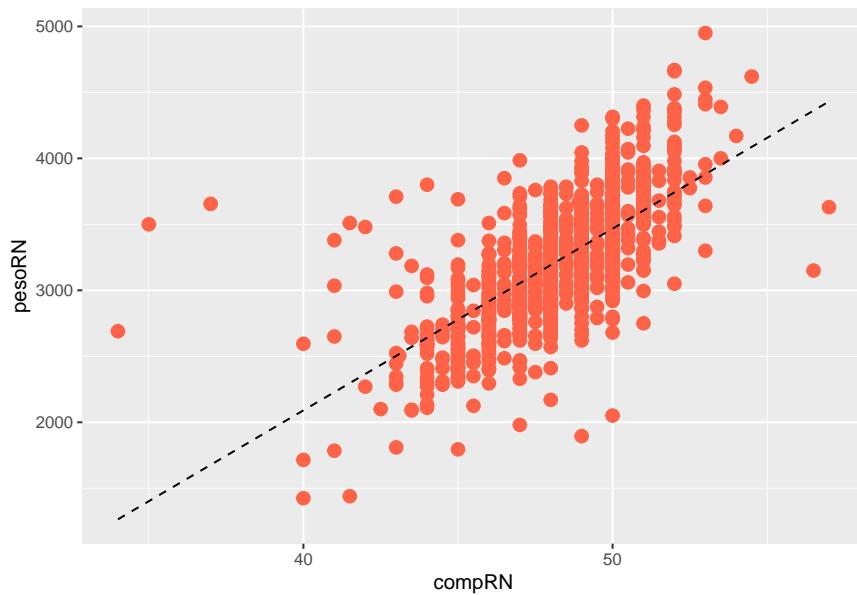


Figura 72: Gráfico de dispersão com reta de ajuste

Quando se usa mais de um `geom`, é importante a ordem em que eles são escritos, pois, como cada um deles é uma camada, elas se sobrepõem e podem se confundir.

6.5.6 Filtrando dados para o gráfico

Faz-se isso, usando a função `filter()` do pacote `dplyr`. Será construído um gráfico igual ao anterior, filtrando apenas o sexo masculino (Figura 73):

```

dados %>% filter (sexo == "masc") %>%
  ggplot(aes(x = compRN, y = pesoRN)) +
  geom_point(color = "steelblue",
             size = 3) +
  geom_smooth(method = "lm",
              se = FALSE,
              color = "black",
              size = 0.5,
              linetype = "dashed")

## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.

## `geom_smooth()` using formula = 'y ~ x'

```

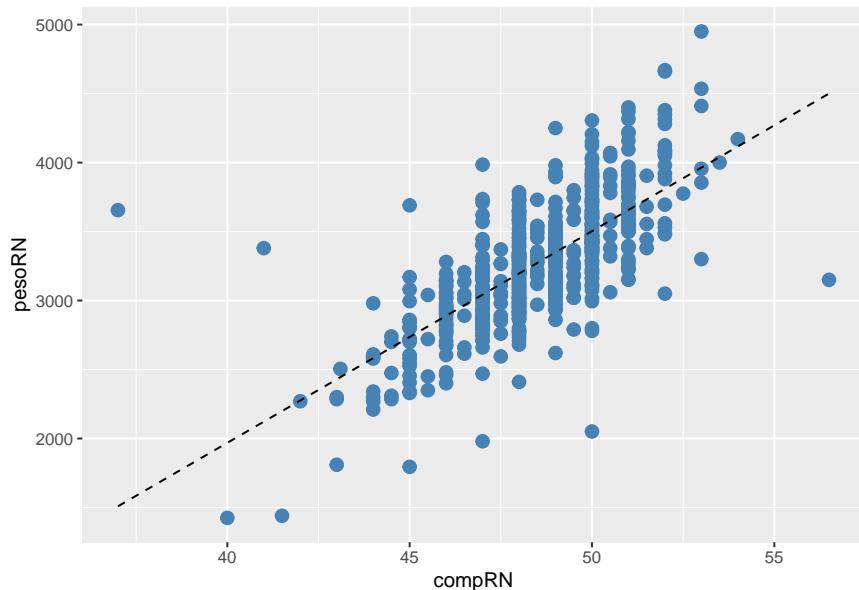


Figura 73: Gráfico de dispersão com reta de ajuste para o sexo masculino

6.5.7 Resumo dos dados usando o geom

Inicialmente, serão usado os argumentos `stat = "summary"` e `fun = "mean"` dentro do `geom_point()` (Figura 74):

```

ggplot(data = dados, aes(x = sexo, y = pesoRN, color = sexo)) +
  geom_point(stat = "summary",
             fun = "mean",
             size = 3,
             show.legend = FALSE)

```

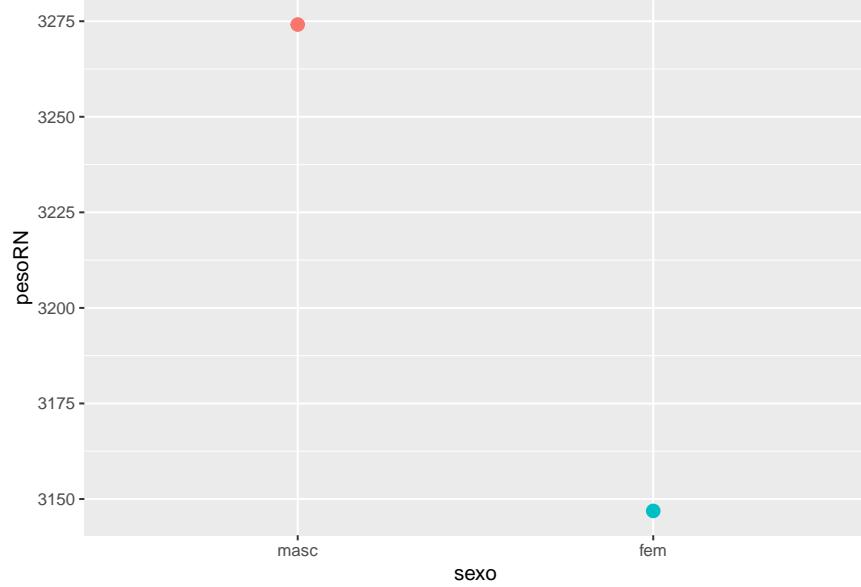


Figura 74: Gráfico resumo, mostrando as médias por sexo

O argumento `show.legend = FALSE` foi acrescentado para evitar o aparecimento da legenda, indicando quem é `masc` ou `fem`, pois o eixo x já mostra.

Pode-se chegar ao mesmo resultado, usando a função `stat_summary()` para acrescentar estatísticas de resumo.

6.5.8 Incluindo barras de erro

Aqui, usamos uma camada `geom_error_bar()`, com os argumentos `stat = summary` e `fun.data = "mean_se"`. Este último argumento fornece a média e o erro padrão e o `width = 0.1`, o tamanho da barra horizontal, gerando o gráfico da Figura 75:

```
ggplot(data = dados, aes(x = sexo, y = pesoRN, color = sexo)) +
  geom_point(stat = "summary",
             fun = "mean",
             size = 3,
             show.legend = FALSE) +
  geom_errorbar(stat = "summary",
                fun.data = "mean_se",
                width = 0.1,
                show.legend = FALSE)
```

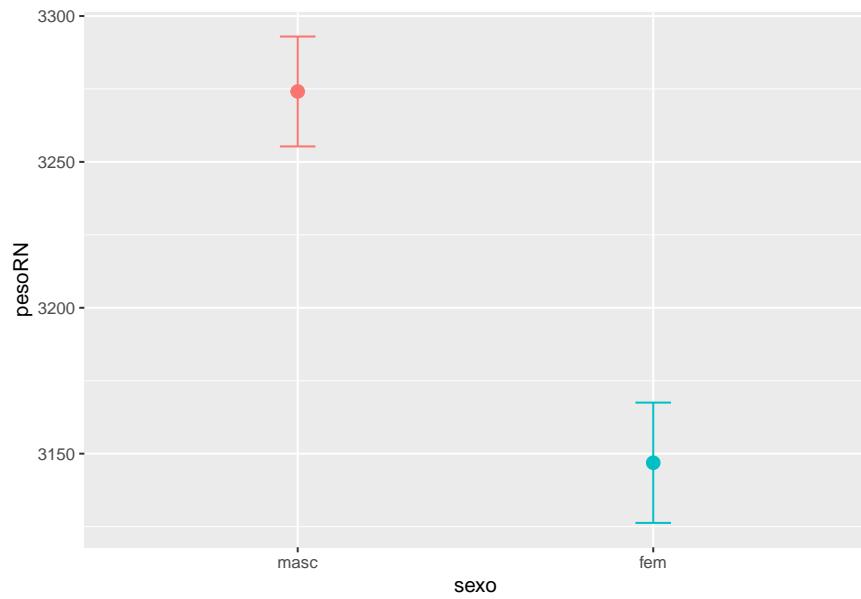


Figura 75: Gráfico de barra de erro no ‘ggplot2’

Usando o pacote `ggbpubr` consegue-se criar um gráfico semelhante (Figura 76) ao anterior (média \pm se), mas com o intervalo de confiança (média $\pm 1.96 \times se$).

```
ggplot(data = dados, aes(x = sexo, y = pesoRN, color = sexo)) +
  geom_point(stat = "summary",
             fun = "mean",
             size = 3,
             show.legend = FALSE) +
  geom_errorbar(stat = "summary",
                fun.data = "mean_ci",
                width = 0.1,
                show.legend = FALSE)
```

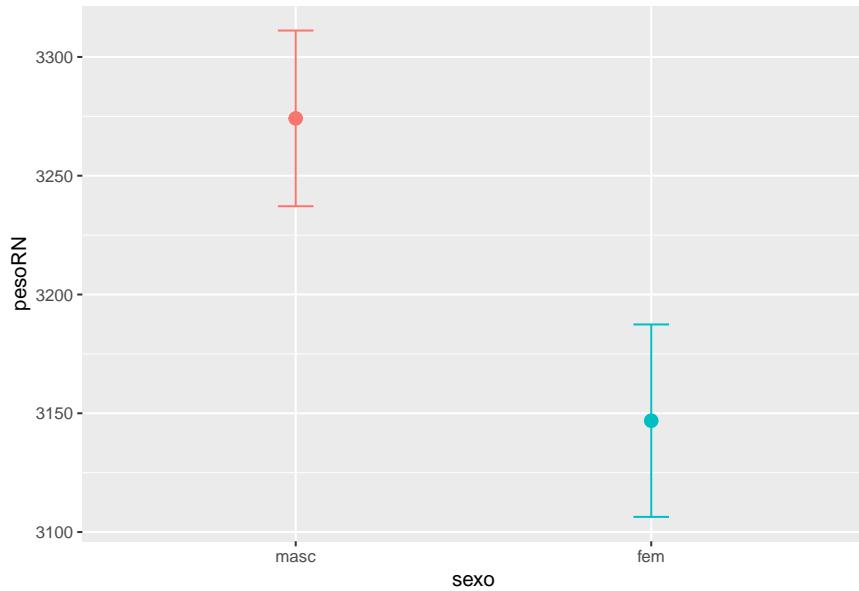


Figura 76: Gráfico de barra de erro no ‘ggpubr’

6.5.9 Incluindo mais de um grupo no gráfico

Agora, será construído o mesmo gráfico do peso dos recém nascidos por sexo, levando em consideração o tabagismo materno (Figura 77). Em primeiro lugar, filtra-se pelo tabagismo, presente ou ausente. Depois seguindo a mesma programação anterior, separando as cores pelo tabagismo (`fumo`). Coloca-se também o argumento `position = position_dodge(0.4)` para que não haja sobreposição das barras no gráfico e exibe-se a legenda (`show.legend = TRUE`):

```
dados %>% filter(fumo %in% c("sim", "não")) %>%
  ggplot(aes(x = sexo, y = pesoRN, color = fumo)) +
  geom_point(stat = "summary",
             fun = "mean",
             position = position_dodge(0.4),
             size = 3,
             show.legend = TRUE) +
  geom_errorbar(stat = "summary",
                fun.data = "mean_ci",
                width = 0.1,
                show.legend = TRUE,
                position = position_dodge(0.4))
```

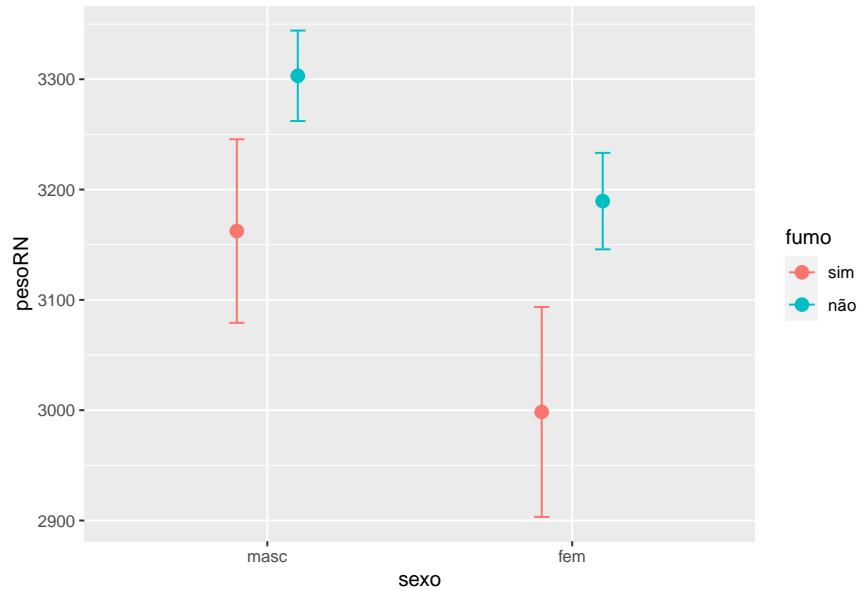


Figura 77: Gráfico do peso do neonato de acordo com sexo e tabagismo materno

6.5.10 Modificando o tema

O tema padrão do `ggplot2` é uma aparência acinzentada que pode ser modificada pela definição de outro tema integrado, como o `theme_bw()` (Figura 78) que é uma variação de `theme_grey()`, padrão, que usa um fundo branco e linhas finas de grade cinza. Outro tema interessante é o `theme_classic()` que é um tema de aparência clássica, com linhas dos eixos `x` e `y` e sem linhas de grade, já usado no gráfico anterior. Para ver outras possibilidades clique [aqui](#).

```
ggplot(data = dados) +
  geom_boxplot(aes(x = sexo,
                    y = pesoRN)) +
  theme_bw()
```

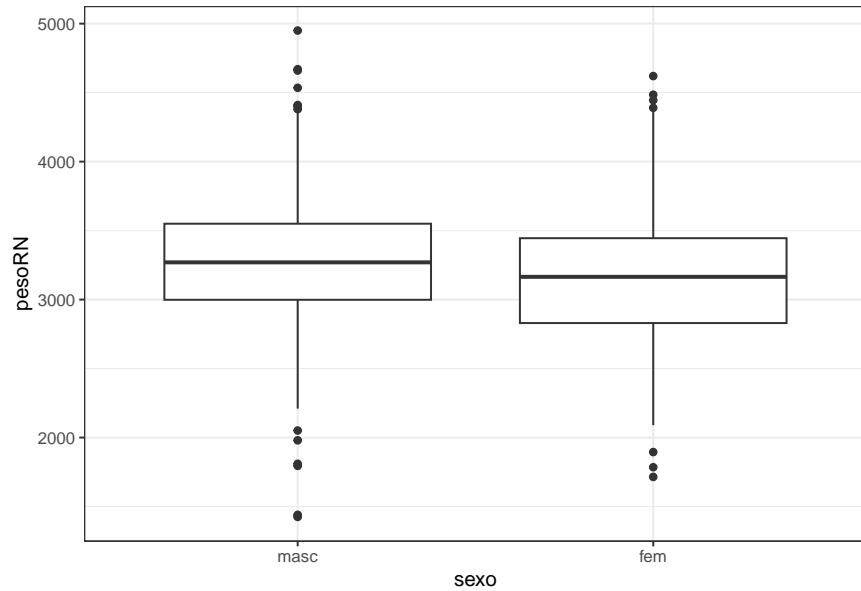


Figura 78: Boxplots do peso do neonato de acordo com sexo

Ou, adicionando cores aos boxplots e removendo a legenda, em uma nova camada, com o argumento `legend.position="none"` da função `theme ()` (Figura 79):

```
ggplot(data = dados) +
  geom_boxplot(aes(x = sexo,
                    y = pesoRN,
                    color = sexo)) +
  theme_classic() +
  theme(legend.position="none")
```

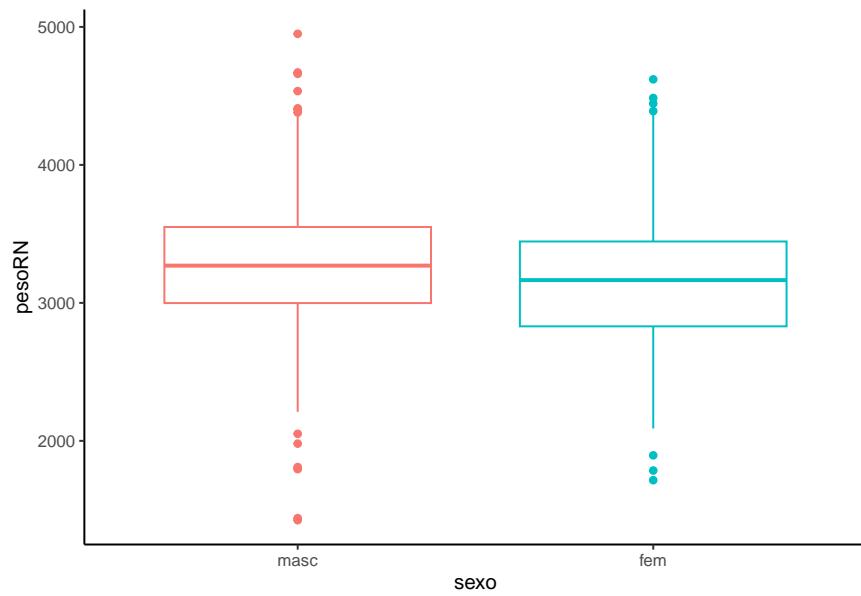


Figura 79: Boxplots com outro tema e sem legenda

6.5.11 Trabalhando com os eixos

Rótulo dos eixos

Para adicionar ou modificar os rótulos dos eixos, adiciona-se `labs()`, escrevendo em cada rótulo (*x* e *y*) os seus respectivos rótulos (Figura 80):

```
ggplot(data = dados) +  
  geom_boxplot(aes(x = sexo,  
                  y = pesoRN,  
                  color = sexo)) +  
  labs (x = "Sexo do recém-nascido",  
        y = "Peso do recém-nascido (g)") +  
  theme_classic() +  
  theme(legend.position="none")
```

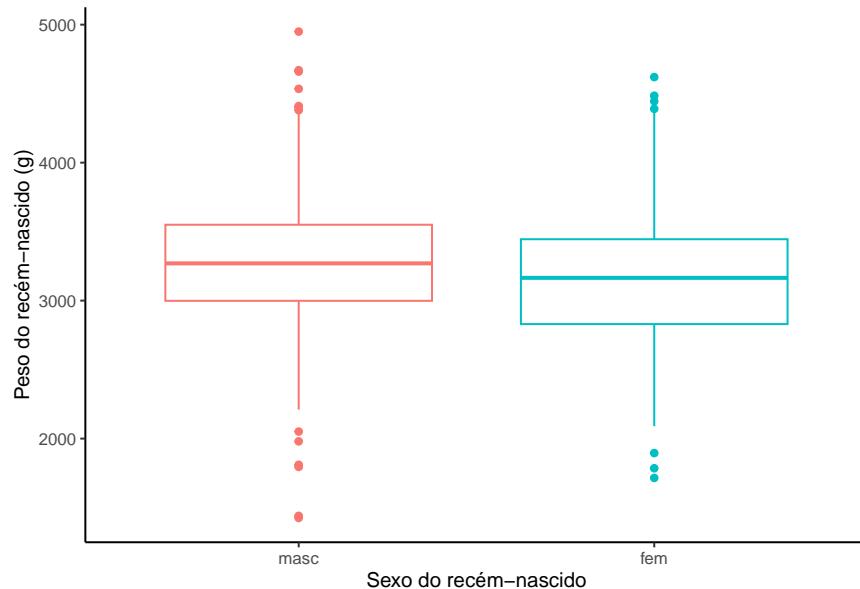


Figura 80: Boxplots com rótulos dos eixos modificados

Modificando o espaço entre o eixo e os rótulos do eixo

A função `theme()` é um comando essencial para modificar elementos específicos do tema (textos e títulos, caixas, símbolos, planos de fundo,etc.). É bastante usado!

Por enquanto, serão modificados elementos de texto (Figura 81). É possível alterar as propriedades de todos ou alguns especificamente (aqui os títulos dos eixos), substituindo o `element_text()` padrão com `theme()`:

```
ggplot(data = dados) +  
  geom_boxplot(aes(x = sexo,  
                  y = pesoRN,  
                  color = sexo)) +  
  labs (x = "Sexo do recém-nascido",  
        y = "Peso do recém-nascido (g)") +  
  theme(axis.title.x = element_text(vjust = 0, size = 15),
```

```

    axis.title.y = element_text(vjust = 2, size = 15)) +
theme_classic() +
theme(legend.position="none")

```

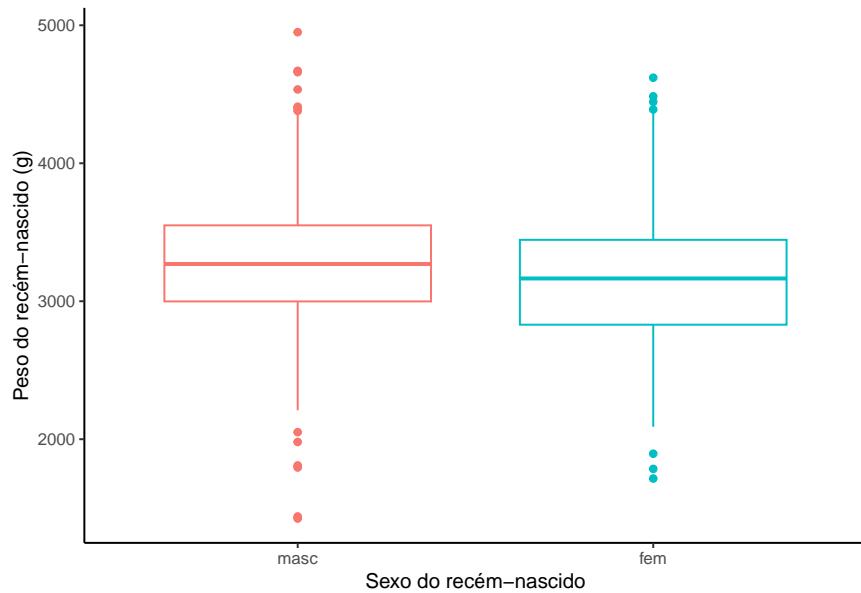


Figura 81: Boxplots com elementos dos textos modificados

O argumento `vjust` se refere ao alinhamento vertical, que geralmente varia entre 0 e 1, mas pode-se especificar valores fora desse intervalo. Pode-se, também, alterar a distância especificando a margem de ambos os elementos de texto (Figura 82):

```

ggplot(data = dados) +
  geom_boxplot(aes(x = sexo,
                   y = pesoRN,
                   color = sexo)) +
  labs (x = "Sexo do recém-nascido",
        y = "Peso do recém-nascido (g)") +
  theme(axis.title.x = element_text(margin = margin (t = 10), size = 15),
        axis.title.y = element_text(margin = margin (r = 10), size = 15)) +
  theme_classic() +
  theme(legend.position="none")

```

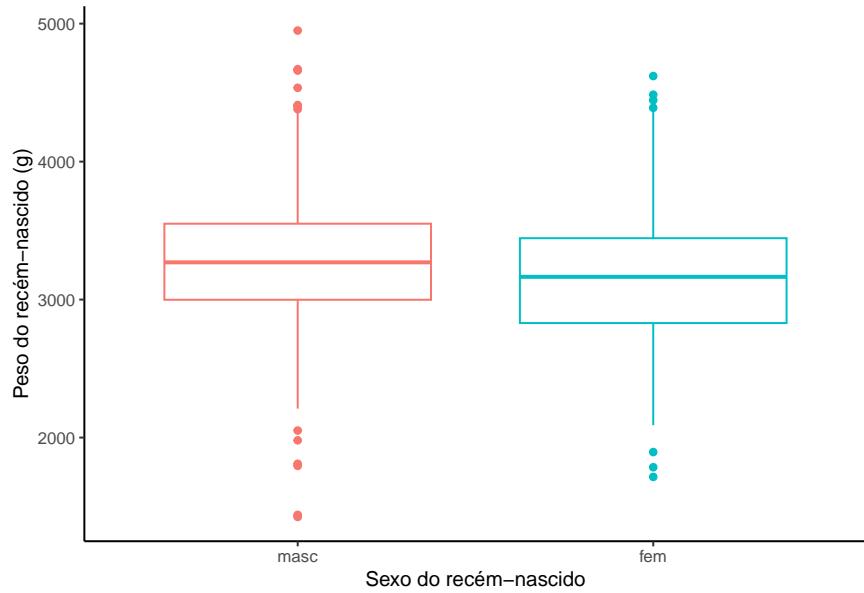


Figura 82: Boxplots com outras modificações

Os rótulos *t* e *r* dentro da função `margin()` referem-se ao topo e à direita, respectivamente. É possível especificar as quatro margens como `margin(t, r, b, l)`. Observe que, agora, há necessidade de alterar a margem direita para modificar o espaço no eixo *y*, não a margem inferior.

Acrescentando um título

Pode-se adicionar um título através da função `ggtitle()` (Figura 83):

```
ggplot(data = obitos) +
  geom_line(aes(x = data,
                 y = obitos)) +
  labs(x = "Data (ano/mês)",
       y = "Nº de mortes") +
  ggtitle ("Mortes por COVID-19 - SES/RS, 2020-22") +
  theme_classic()
```

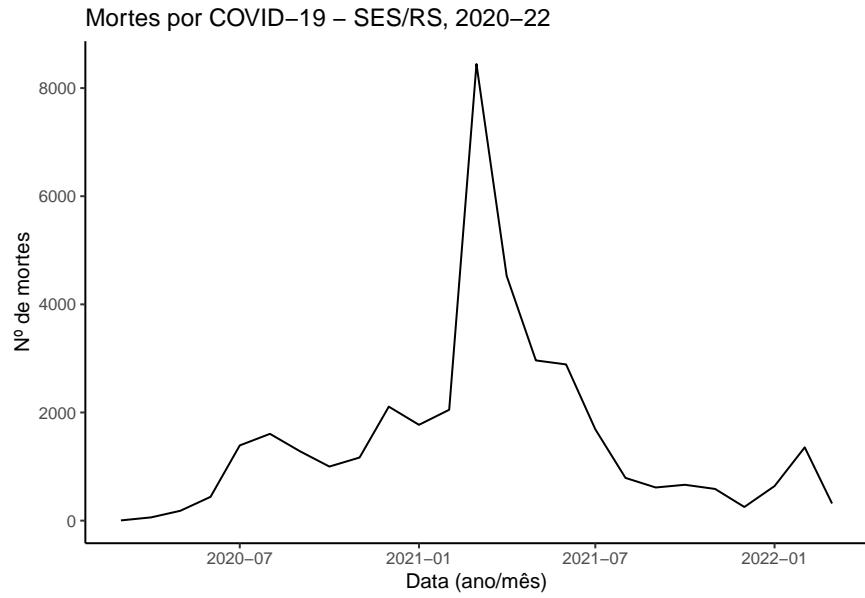


Figura 83: Boxplots com elementos dos textos modificados

Uma outra maneira de colocar título, subtítulo e fonte no gráfico é (Figura 84):

```
ggplot(data = obitos) +
  geom_line(aes(x = data,
                y = obitos)) +
  labs(x = "Data (ano/mês)",
       y = "Nº de mortes",
       title = "Mortes por COVID-19",
       subtitle = "RS - 2020-2022",
       caption = "Fonte: SES") +
  theme_classic()
```

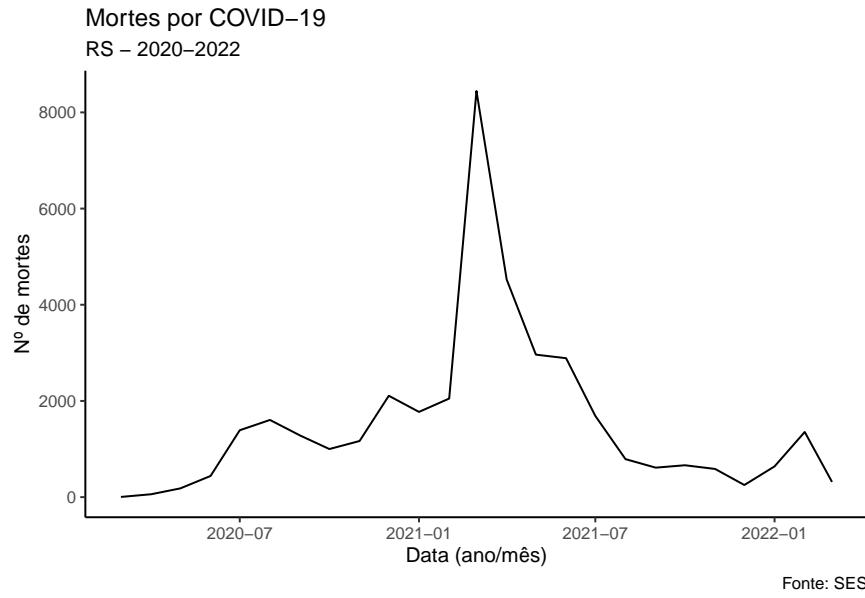


Figura 84: Boxplots com elementos dos textos modificados

Outro exemplo, modificando o tamanho, estilo e tipo de fonte (Figura 85). Isto é feito, adicionando a camada `theme()` com argumentos `plot.title`, `plot.subtitle`, etc. Para maiores detalhes consulte [aqui](#).

```
ggplot(data = dados, aes(x = sexo, y = pesoRN, color = sexo)) +
  geom_point(stat = "summary",
             fun = "mean",
             size = 3,
             show.legend = FALSE) +
  geom_errorbar(stat = "summary",
                fun.data = "mean_ci",
                width = 0.1,
                show.legend = FALSE) +
  labs(x = "Sexo do recém-nascido",
       y = "Peso do recém-nascido (g)",
       title = "Peso do RN por sexo",
       subtitle = "Maternidade do HGCS",
       caption = "Fonte: Autor") +
  theme_classic() +
  theme (plot.title = element_text(size = 13,
                                    face = "bold",
                                    color = "navy"),
         plot.subtitle = element_text(size = 11,
                                    face = "bold",
                                    color = "steelblue"))
```

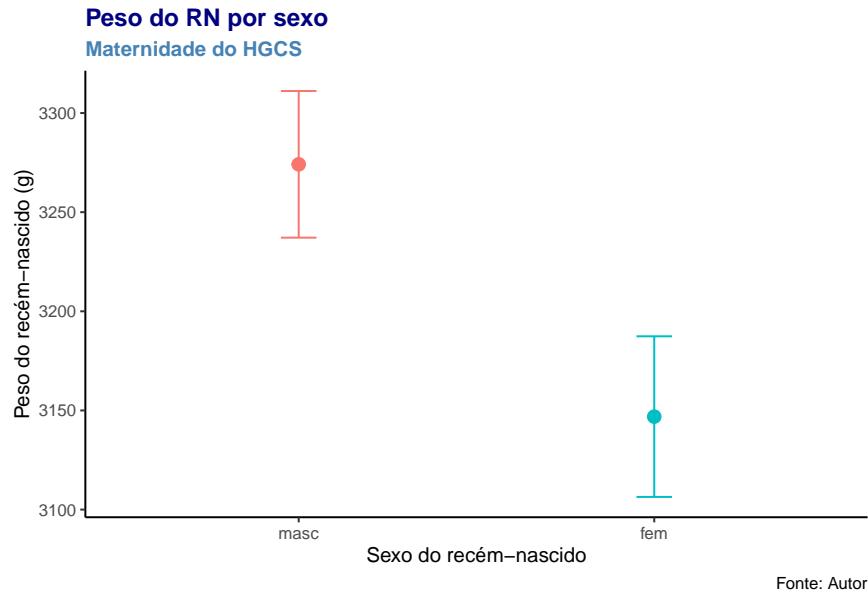


Figura 85: Boxplots com elementos dos textos modificados

Modificando os limites dos eixos

O sistema de coordenadas cartesianas é o tipo de sistema de coordenadas mais familiar e comum. Definir limites no sistema de coordenadas ampliará o gráfico (como se você estivesse olhando para ele com uma lupa) e não alterará os dados subjacentes, como definir limites em uma escala. Para realizar este trabalho, vamos usar a função `coord_cartesian ()` ou `scale_y_continuous ()` ou `scale_x_continuous ()`.

Será usado aqui o gráfico já construído acima (Figura 72) com pequenas alterações e vamos armazená-lo em um objeto, denominado `gd` (gráfico de dispersão). Isto facilita a repetição do gráfico em outros códigos, pois basta escrever `gd` e executar (Figura 86).

```
gd <- ggplot(data = dados, aes(x = comprRN, y = pesoRN)) +
  geom_point(color = "tomato",
             size = 3) +
  geom_smooth(method = "lm",
              se = FALSE,
              color = "black",
              size = 0.8,
              linetype = "dashed") +
  labs(x = "Comprimento do RN (cm)", y = "Peso do RN (g)",
       title = "Gráfico de Dispersão",
       caption = "Fonte: Autor") +
  theme_classic()
gd
```

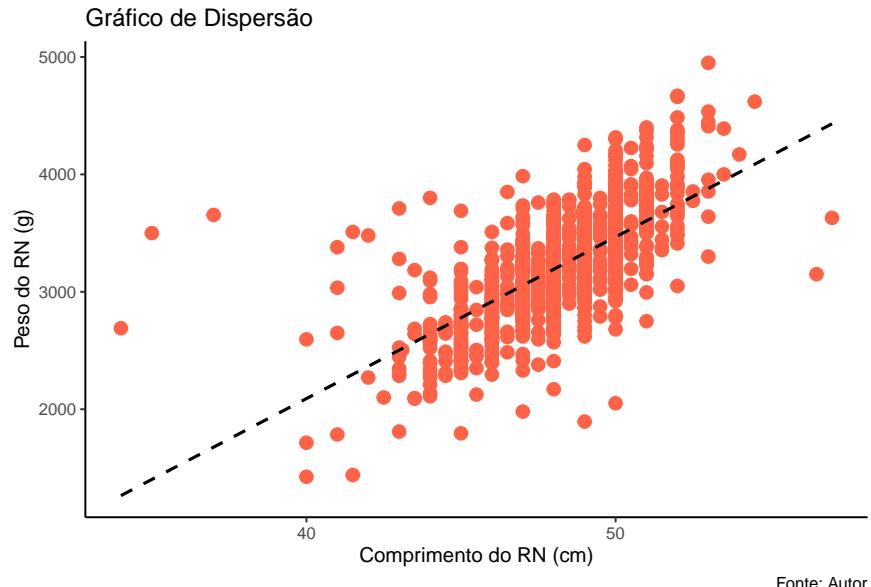


Figura 86: Gráfico de dispersão

1. Função `coord_cartesian()`

- `xlim`, `ylim` → limites dos eixos x e y
- `expand` → Se TRUE, o padrão, adiciona um pequeno fator de expansão aos limites para garantir que dados e eixos não se sobreponham. Se FALSE, os limites são tirados exatamente dos dados ou `xlim/ylim` (Figura 87).

```
gd + coord_cartesian(ylim = c(3500, 4000),
                      xlim = c(45, 55),
                      expand = TRUE)
```

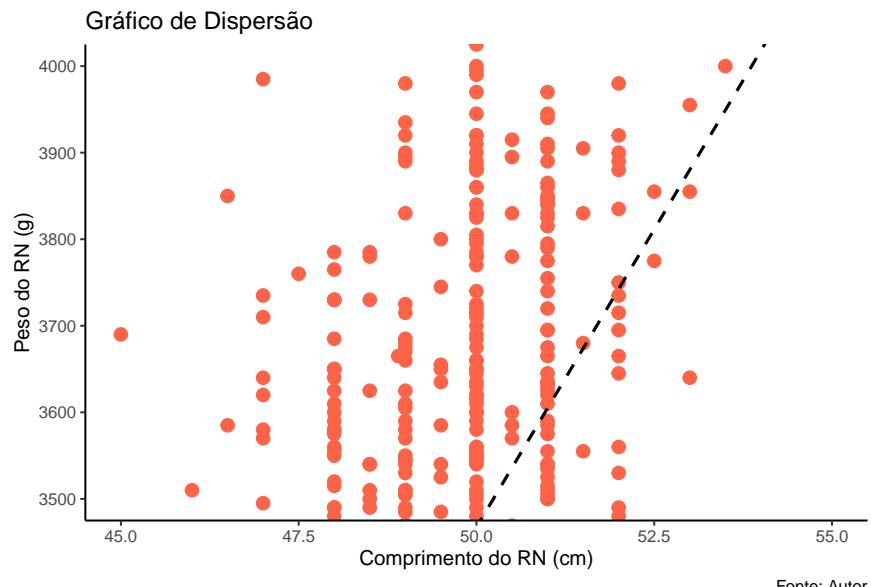


Figura 87: Gráfico de dispersão expandido

Observe que é como se fizesse um zoom no gráfico nos limites estabelecidos. Há um corte um pouco acima dos limites. No eixo y, um pouco acima de 4000 e um pouco abaixo de 3500 e, no eixo x, um pouco à esquerda de 45 e um pouco à direita de 55. Isto aconteceu, porque colocamos `expand = TRUE`. Para extrair esta margem, colocar `expand = FALSE`:

2. Usando escalas de posição contínuas

- Pode-se usar a função `scale_y_continuous()` e `scale_x_continuous()` para fazer algo parecido com a `coord_cartesian()` (Figura 88):

```
gd + scale_x_continuous(limits = c(45, 55)) +
  scale_y_continuous (limits = c(3500, 4000))
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

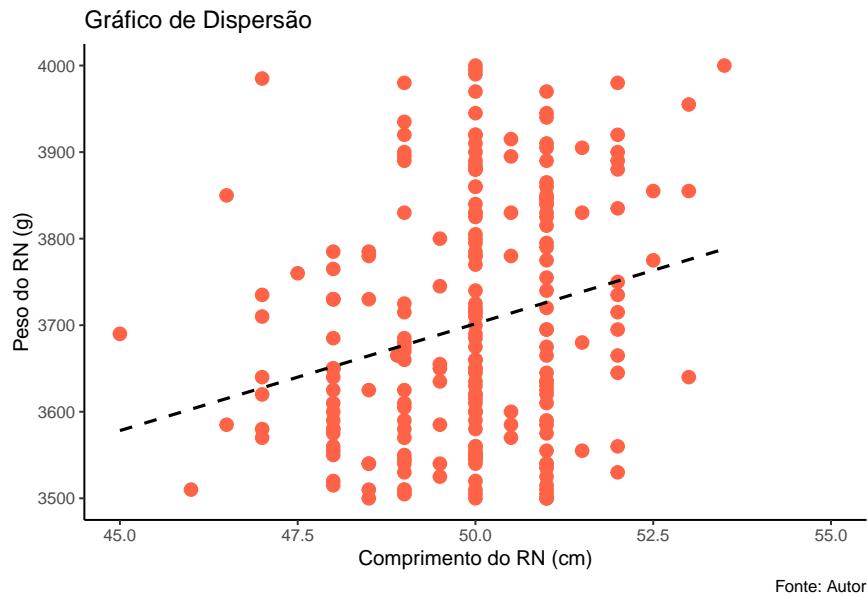


Figura 88: Gráfico de dispersão expandido

A função removeu os casos que estão fora dos limites estabelecidos. No caso, a mensagem do R mostra que foram removidos 1132 casos. O gráfico foi construído sem estes casos e, no gráfico anterior, houve apenas uma aproximação (um zoom) dentro dos limites. Portanto, houve um impacto importante no gráfico.

Se os dados não forem em escala contínua, é possível escolher outra escala, por exemplo `scale_y_discrete()`.

Modificando a expansão

Mostrou-se que é possível interferir na expansão da margem com o argumento `expand = TRUE` ou `FALSE` que pode ser usado também com a função `scale_y_continuous()`. Agora, será visto um exemplo de um gráfico de barras das faixas etárias das gestantes (Figura 89):

```

gb <- ggplot(data = dados) +
  geom_bar(aes(x = idadeCateg, y = after_stat(count/sum(count)),
               fill = idadeCateg),
            show.legend = FALSE) +
  labs(y = "Frequência", x = "Faixa Etária da Parturiente")
gb

```

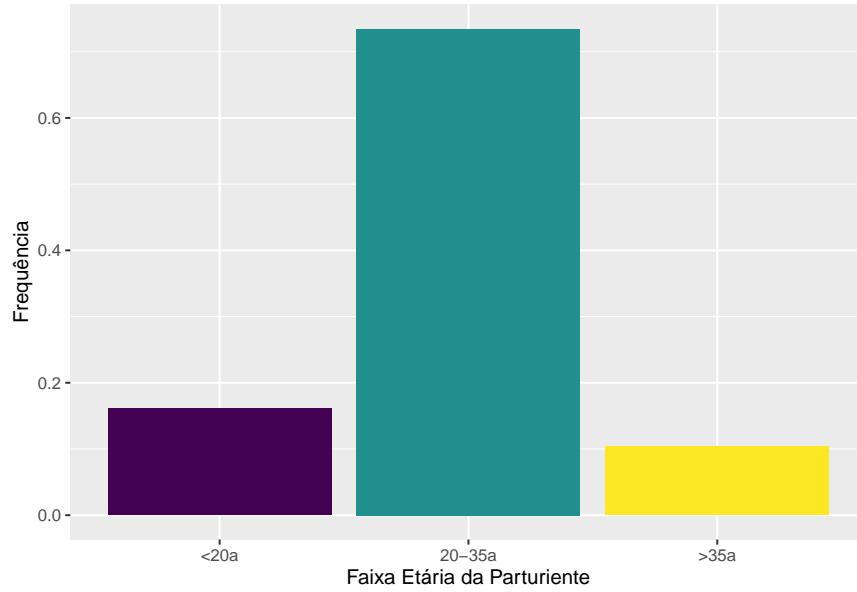


Figura 89: Gráfico de barras

Observe que abaixo do 0 (zero) existe uma expansão (Figura 90). Para que as barras tenham início exatamente no 0, pode-se empregar a função `scale_y_continuous()` com o argumento `expand = expansion(add = c(0,50))`, significando que não se expande nada abaixo do 0 e se adiciona 50 unidades para cima, criando uma margem superior.

```

gb + scale_y_continuous (expand = expansion(add = c(0,0.05)))

```

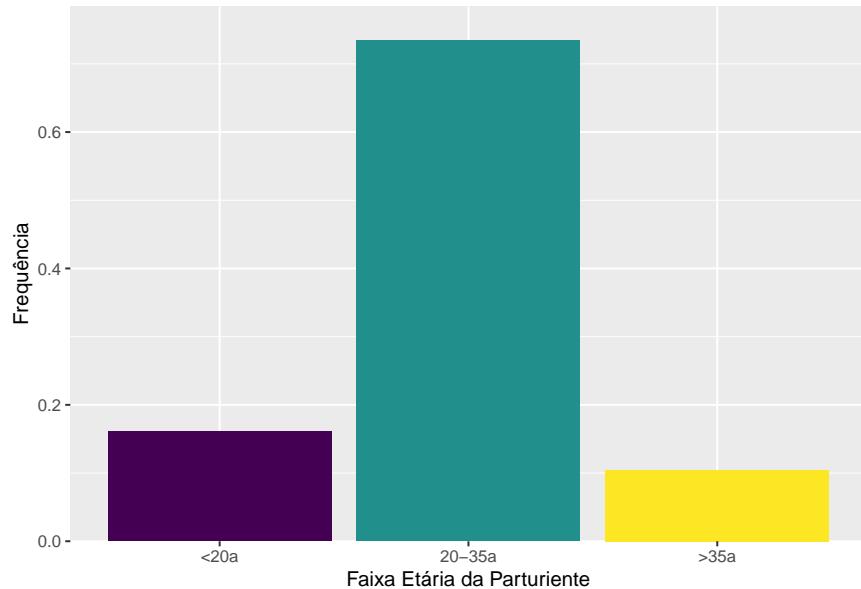


Figura 90: Gráfico de barras com expansão

Isto também poderia ser feito com `mult` no lugar do `add` (Figura 91), representando o multiplicador que se coloca acima e abaixo:

```
gb + scale_y_continuous (expand = expansion(mult = c(0,0.05)))
```

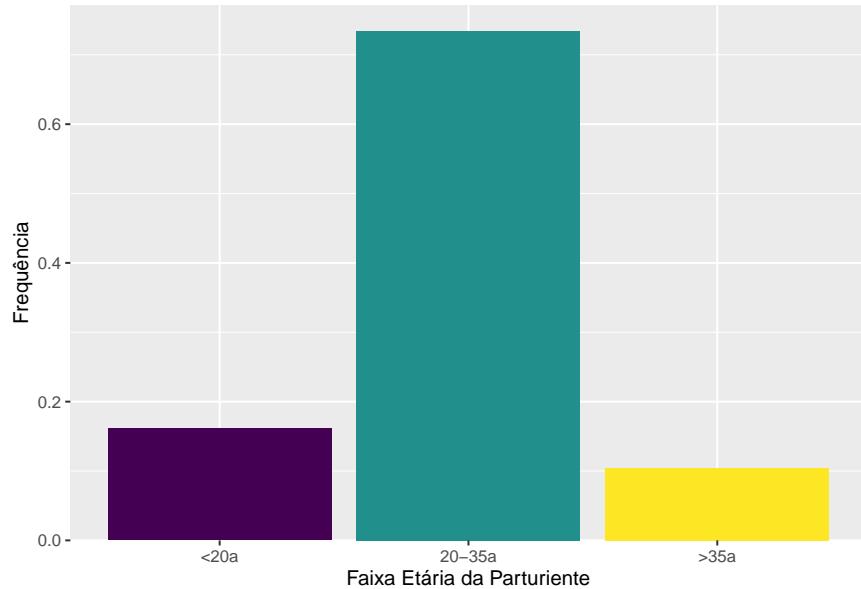


Figura 91: Gráfico de barras, igual a anterior

Usando a proporção ou percentagem nos eixos

Para usar a proporção no eixo y do gráfico anterior, devemos modificar a estética deste eixo, usando `y = after_stat(count/sum(count))` (Figura 92)

```

gbp <- ggplot(data = dados) +
  geom_bar(aes(x = idadeCateg, y = after_stat(count/sum(count))),
            fill = idadeCateg),
  show.legend = FALSE) +
  labs(y = "Frequência",
       x = "Faixa Etária da Parturiente",
       caption = "Fonte: Autor")
gbp

```

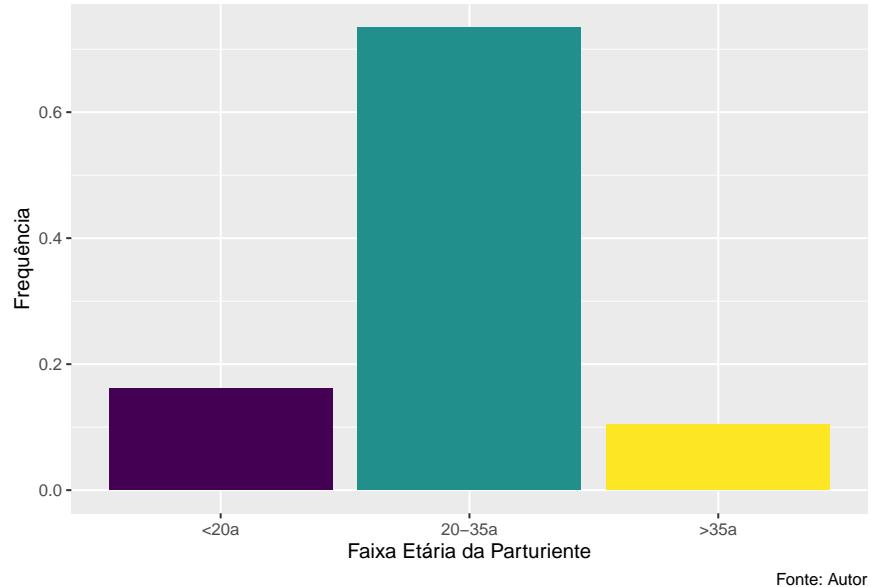


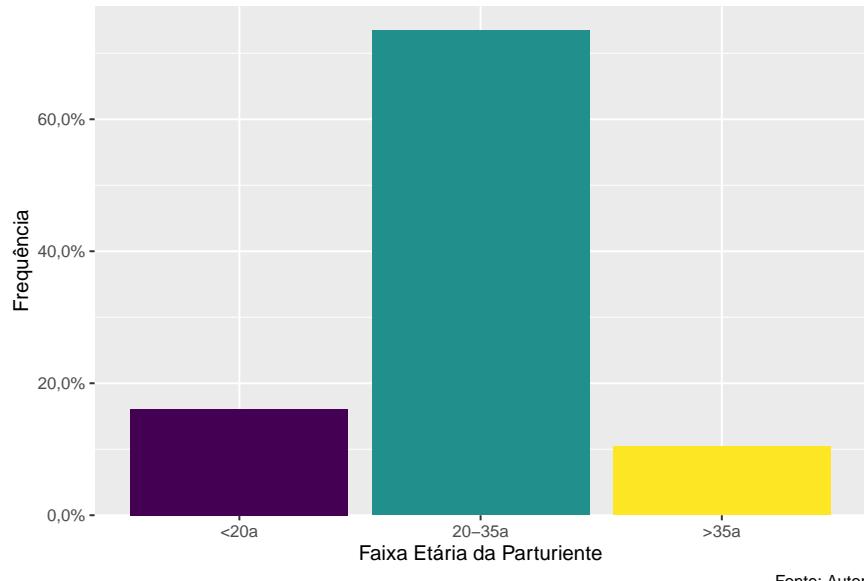
Figura 92: Gráfico de barras com proporções no eixo y

Para ter a percentagem (Figura 93), empregar a função `percent_format()` do pacote `scales` (81):

```

gbp + scale_y_continuous (expand = expansion(mult = c(0,0.05)),
                           labels = percent_format (accuracy = 0.1,
                           decimal.mark = ","))

```



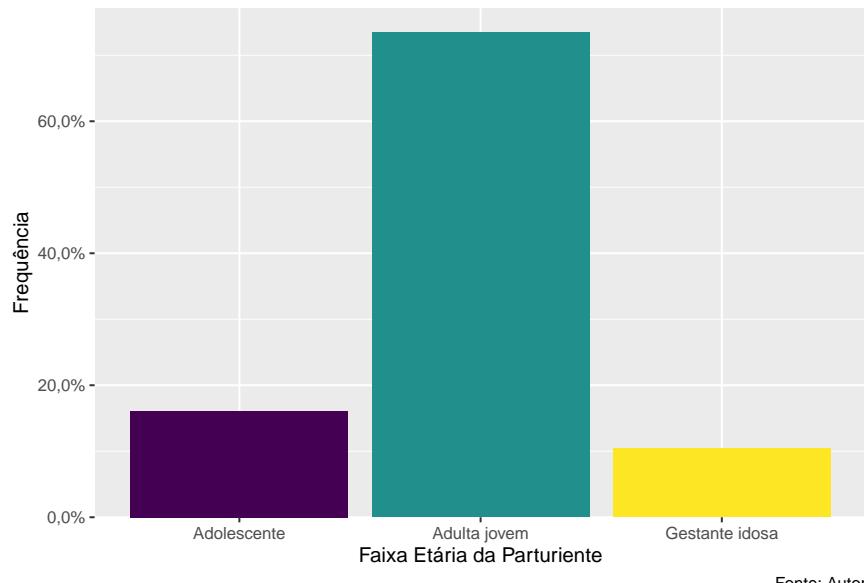
Fonte: Autor

Figura 93: Gráfico de barras com percentagens no eixo y

Mudando o nome e a ordem dos rótulos do eixo x

No gráfico acima, temos a faixa etária dividida em <20a, 20–35a e >35a. Pode haver interesse em mudar para adolescentes, adultas jovens e gestante idosa (Figura 94). Para fazer isso, sem mudar o banco de dados, simplesmente modifica-se os rótulos no argumento `labels` da função `scale_x_discrete()`:

```
gbp <- ggplot(data = dados) +
  geom_bar(aes(x = idadeCateg, y = after_stat(count/sum(count)),
               fill = idadeCateg),
           show.legend = FALSE) +
  labs(y = "Frequência",
       x = "Faixa Etária da Parturiente",
       caption = "Fonte: Autor") +
  scale_y_continuous (expand = expansion(mult = c(0,0.05)),
                     labels = percent_format (accuracy = 0.1,
                                              decimal.mark = ",")) +
  scale_x_discrete (labels = c("Adolescente", "Adulta jovem", "Gestante idosa"))
gbp
```



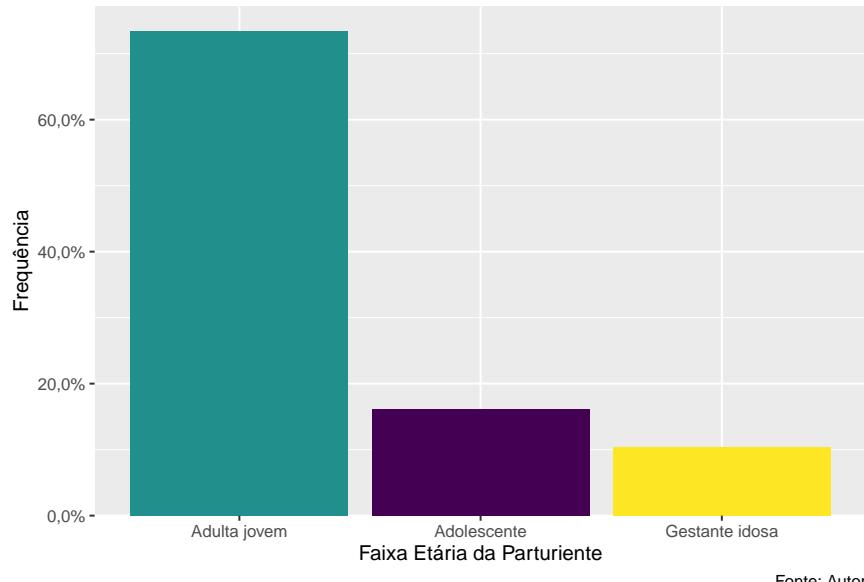
Fonte: Autor

Figura 94: Gráfico de barras com eixo x modificado

É viável, também, mudar esta ordem no eixo x, da maior frequência para a menor (Figura 95):

```
gbp + scale_x_discrete (limits = c("20-35a", "<20a", ">35a"),
                         labels = c("Adulta jovem", "Adolescente", "Gestante idosa"))

## Scale for x is already present.
## Adding another scale for x, which will replace the existing scale.
```



Fonte: Autor

Figura 95: Gráfico de barras com eixo x modificado

Pode-se conseguir o mesmo resultado anterior com a função `fct_infreq()` do pacote `forcats`, colocada na estética do `geom_bar()`.

```
ggplot(data = dados) +
  geom_bar(aes(x = fct_infreq(idadeCateg), y = after_stat(count/sum(count))),
           fill = idadeCateg,
           show.legend = FALSE) +
  labs(y = "Frequência",
       x = "Faixa Etária da Parturiente",
       caption = "Fonte: Autor") +
  scale_y_continuous (expand = expansion(mult = c(0,0.05)),
                      labels = percent_format(accuracy = 0.1,
                                              decimal.mark = ",")) +
  scale_x_discrete (labels = c("Adulta jovem", "Adolescente", "Gestante idosa"))
```

Moficando os intervalos dos valores do eixo

O gráfico de linha de mortes por COVID no RS, 2020-2022, visto anteriormente (Figura 66), será atribuído a um objeto `gl`:

```
gl <- ggplot(data = obitos) +
  geom_line(aes(x = data, y = obitos)) +
  labs(x = "Ano (mês)", y = "Nº de mortes") +
  theme_classic()
gl
```

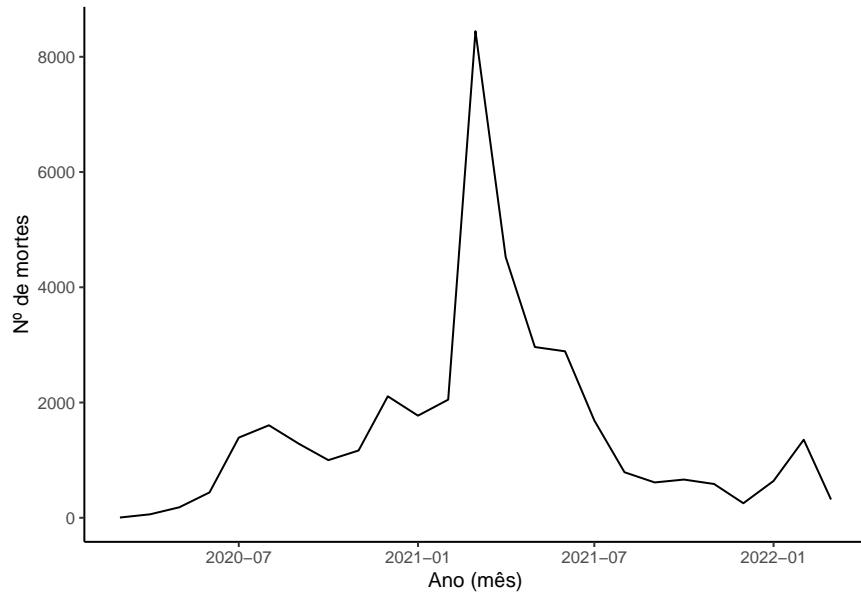


Figura 96: Mortes por COVID-19, 2020-2022, RS.

Observe (Figura 96) que, no eixo y , os óbitos estão registrados a cada 2000 e, no eixo x , foram marcadas apenas 4 datas. Podemos modificar isso adicionando duas camadas, usando as funções `scale_y_continuous()` e `scale_x_datetime()`:

```
gl + scale_y_continuous(n.breaks = 10) +
  scale_x_datetime(date_breaks = "4 month",
                  date_labels = "%Y (%b)")
```

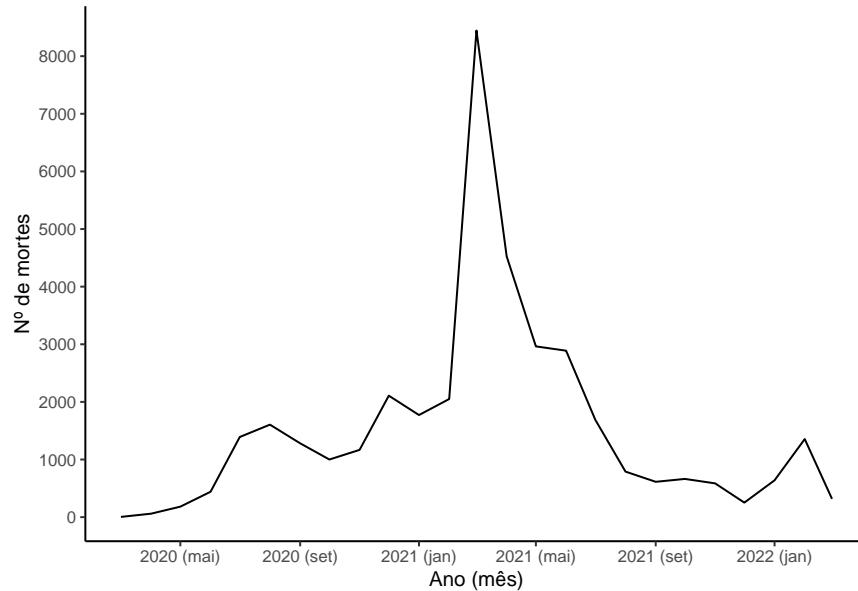


Figura 97: Mortes por COVID-19, 2020-2022, RS.

O aspecto do gráfico mudou um pouco (Figura 97). Agora, existem marcações no eixo *y* a cada 1000 mortes e o registro do tempo aparece a cada 4 meses, conforme estabelecido no argumento `date_breaks = "4 month"` e o formato foi modificado com o argumento `date_labels = "%Y %b"`. Neste, `%Y` significa o ano e `%b` significa o mês abreviado (Jan-Dec). Para ver como customizar as datas, veja [aqui](#)

6.5.12 Modificação das cores

Voltando a usar um gráfico de barra, já visto anteriormente (Figura 94), da distribuição da idade da gestante por faixa etária, onde `ggplot2` escolheu as cores porque foi colocado o argumento `fill = idadeCateg`.

Essas cores não foram escolhidas pelo autor e é possível mudá-las, usando uma paleta própria. por exemplo, adicionando uma camada com a função `scale_fill_manual()` (Figura 98):

```
ggplot(data = dados) +
  geom_bar(aes(x = idadeCateg, y = after_stat(count/sum(count))),
           fill = idadeCateg),
  show.legend = FALSE) +
  labs(y = "Frequência",
       x = "Faixa Etária da Parturiente",
       caption = "Fonte: Autor") +
  scale_y_continuous (expand = expansion(mult = c(0,0.05)),
                     labels = percent_format (accuracy = 0.1,
                                              decimal.mark = ",")) +
  scale_x_discrete (labels = c("Adolescente", "Adulta jovem", "Gestante idosa")) +
  scale_fill_manual(values = c("steelblue", "navy", "lightblue"))
```

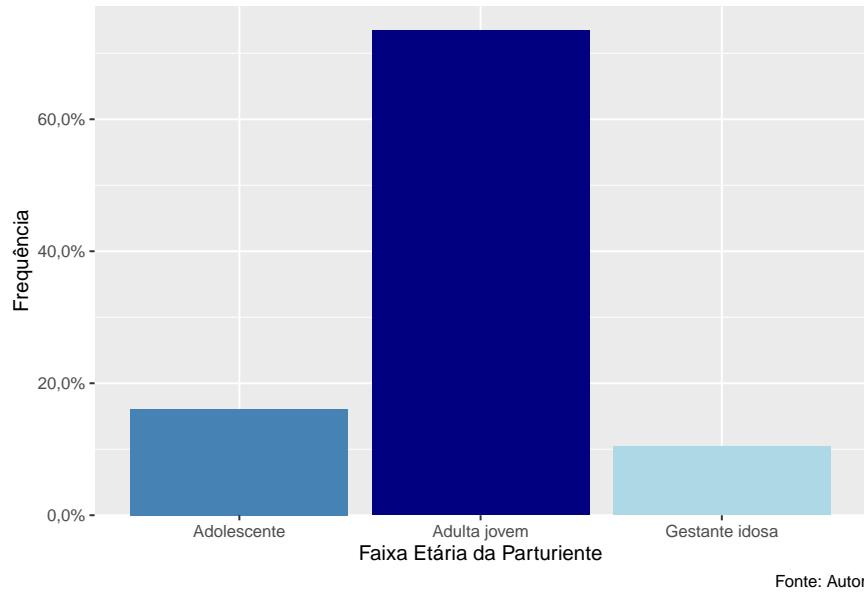


Figura 98: Frequência da faixa etária das parturientes da Maternidade do HCCS, 2008.

As cores podem ser escolhidas [aqui](#).

Além de escrever o nome das cores aceitas pelo ggplot2, é possível usar o sistema hexadecimal que utiliza números e letras. Ver [aqui](#) o gerador de paletas. Copiar o código e colocar antes o símbolo #.

É possível, também, usar uma paleta de pacotes do R, como o pacote RColorBrewer(Figura 99)

```
ggplot(data = dados) +
  geom_bar(aes(x = idadeCateg, y = after_stat(count/sum(count)),
               fill = idadeCateg),
           show.legend = FALSE) +
  labs(y = "Frequência",
       x = "Faixa Etária da Parturiente",
       caption = "Fonte: Autor") +
  scale_y_continuous (expand = expansion(mult = c(0,0.05)),
                     labels = percent_format (accuracy = 0.1,
                                              decimal.mark = ",")) +
  scale_x_discrete (labels = c("Adolescente", "Adulta jovem", "Gestante idosa")) +
  scale_fill_brewer(palette = "Dark2")
```

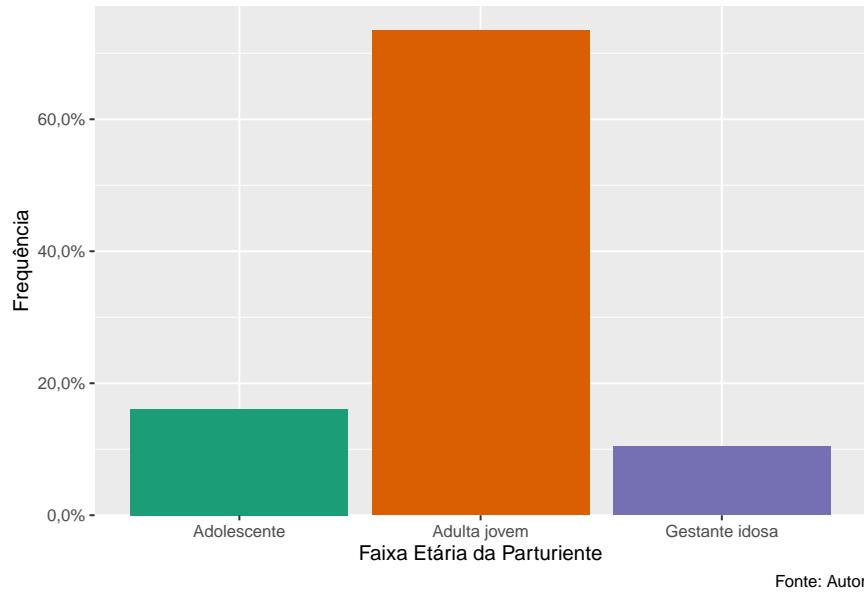


Figura 99: Frequência da faixa etária das parturientes da Maternidade do HCCS, 2008.

Usando o pacote `ggsci`, pode-se escolher o padrão de algumas revistas médicas como a JAMA, Lancet, etc (Figura 100). Para maiores detalhes acesse [aqui](#)

```
pacman::p_load(ggsci)

ggplot(data = dados) +
  geom_bar(aes(x = idadeCateg, y = after_stat(count/sum(count)),
               fill = idadeCateg),
           show.legend = FALSE) +
  labs(y = "Frequência",
       x = "Faixa Etária da Parturiente",
       caption = "Fonte: Autor") +
  scale_y_continuous (expand = expansion(mult = c(0,0.05)),
                     labels = percent_format (accuracy = 0.1,
                                              decimal.mark = ",")) +
  scale_x_discrete (labels = c("Adolescente", "Adulta jovem", "Gestante idosa")) +
  scale_fill_lancet()
```

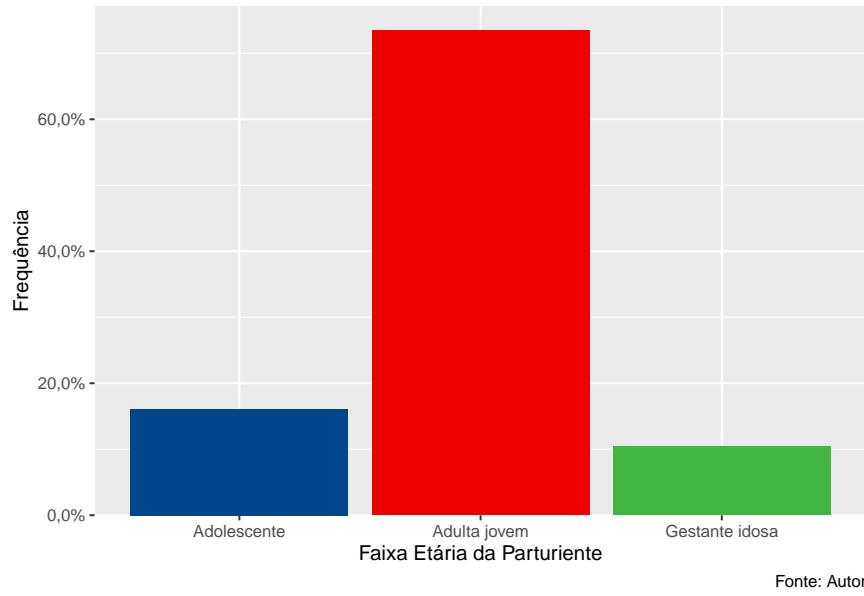


Figura 100: Frequência da faixa etária das parturientes da Maternidade do HCCS, 2008.

6.5.13 Exemplo final: Gráfico de barra de erro com colunas

Será construído um gráfico de barras de erro para visualizar a influência do sexo e do tabagismo materno no peso do recém-nascido (Figura 101). Será incluído a representação das colunas (barras) e as barras de erro com intervalo de confiança de 95%, calculado usando $\text{média} \pm \text{margem de erro}$, onde margem de erro = $1.96 \times \text{erro padrão}$. Estes conceitos serão discutidos em outros capítulos.

Em primeiro lugar, faz-se um resumo dos dados que serão usados no gráfico:

```
resumo <- dados %>%
  group_by(sexo, fumo) %>%
  dplyr::summarise(n = n(),
    media = mean(pesoRN, na.rm = TRUE),
    dp = sd(pesoRN, na.rm = TRUE),
    me = 1.96 * dp/sqrt(n))
resumo

## # A tibble: 4 x 6
## # Groups:   sexo [2]
##   sexo   fumo     n   media     dp     me
##   <fct> <fct> <int> <dbl> <dbl> <dbl>
## 1 masc   sim     122 3162.  464.  82.4
## 2 masc   não    470 3303.  453.  40.9
## 3 fem    sim     110 2998.  503.  94.1
## 4 fem    não    383 3190.  435.  43.6
```

Onde, dp = desvio padrão e me = margem de erro. O objeto `resumo` pertence a classe `data.frame` e será empregado na construção do gráfico:

```

ggplot(resumo,
       aes(x=sexo, y=media, fill=fumo)) +
  geom_bar(stat="identity", color="black",
           position=position_dodge()) +
  geom_errorbar(aes(ymin=media-me, ymax=media+me), width=.2,
                position=position_dodge(.9)) +
  labs(x="Sexo",
       y = "Peso do RN (g)",
       fill = "Tabagismo",
       caption = "RN = Recém-nascido")+
  theme_classic() +
  scale_fill_manual(values=c('gray80','darkslategray1'))

```

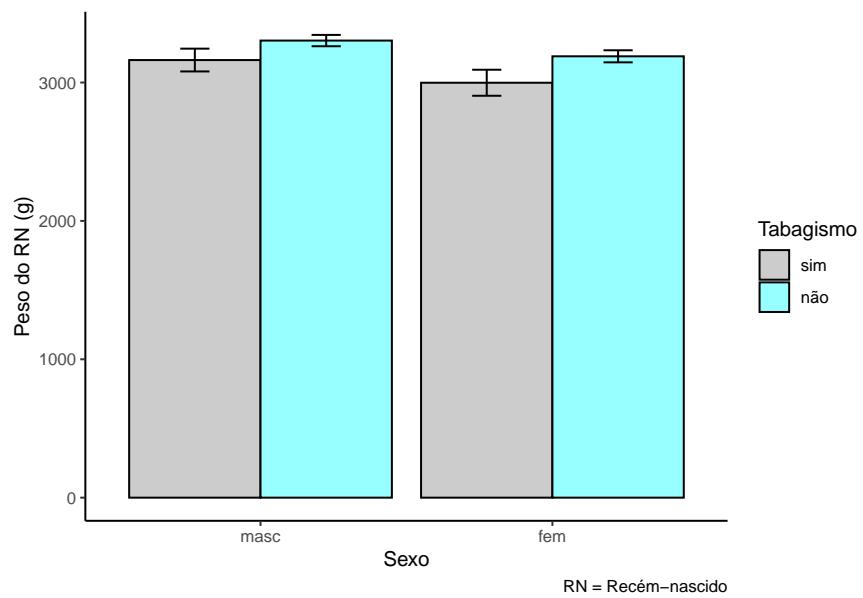


Figura 101: Influência do sexo e tabagismo materno no peso ao nascer.

Observe que o rótulo da legenda foi determinado com `fill = "Tabagismo"`, porque a cor das barras foi estabelecida na estética do `ggplot` com o mesmo argumento.

OBS.: Clique em [ggplot2::cheat sheet](#) para obter a planilha de dicas do `ggplot2`.

7 Introdução à Probabilidade

7.1 Introdução

A teoria das probabilidades é a base sobre a qual a estatística é desenvolvida. Os jogos de azar deram um grande impulso ao conhecimento da teoria da probabilidade, principalmente, pelo trabalho de Blaise Pascal (1623-1662) em parceria com Pierre de Fermat (1601-1665), estimulados por um nobre francês, Antoine Gombaud, conhecido como Chevalier de Mère, inveterado jogador, que estava cansado dos resultados negativos em suas apostas (82).

A Teoria das probabilidades permite que seja possível modelar populações, experimentos ou qualquer situação que possa ser considerada aleatória. Estes modelos possibilitam fazer inferência sobre populações a partir da observação de uma amostra dessa população. Ao usar apenas uma parte da população, inevitavelmente, é cometido um erro o *erro amostral*. Este erro amostral pode ser dimensionado pela teoria das probabilidades.

Existem duas interpretações alternativas de probabilidades: a frequentista e a bayesiana (83). Neste livro, será discutida, basicamente, a definição de probabilidade frequentista. O processo bayesiano de formulação de um modelo probabilístico faz uso do conhecimento subjetivo, estabelecendo uma especificação *a priori*, combinado com a informação objetiva ou empírica. A teoria bayesiana é a estrutura integradora dessas duas fontes de informação, derivando como resultado a distribuição *a posteriori* dos parâmetros de interesse. No capítulo sobre análise de testes diagnósticos, será abordado alguns aspectos relacionados a teoria bayesiana.

7.2 Processo aleatório

Um processo ou experimento é dito *aleatório* quando em uma situação se sabe quais os resultados que podem acontecer, mas não se sabe qual resultado particular irá acontecer. Por exemplo, quando uma moeda é lançada, se conhece que a probabilidade de o desfecho cara ocorrer é de 50%, mas se desconhece o que irá ocorrer até que a moeda esteja no chão.

O número de caras que podem surgir em vários lançamentos da moeda é chamado de *variável aleatória*, ou seja, uma variável que pode assumir mais de um valor com determinadas probabilidades (84). Da mesma forma, um dado lançado pode mostrar seis faces, numeradas de um a seis, com igual probabilidade de 16,7%. Portanto, quando a probabilidade é associada a todos os conjuntos de valores possíveis de uma variável, diz-se que ela é aleatória. O conjunto de todos os possíveis resultados de um experimento aleatório é denominado *espaço amostral*.

Na área da saúde, trabalha-se com uma infinidade de variáveis aleatórias, por exemplo, o número de filhos de uma mulher, o número de mortos diários em uma epidemia, o número de vacinados em uma campanha, etc. Essas variáveis são a variáveis aleatórias discretas, pois apenas permitem ser quantificadas por processo de contagem. Por outro lado, o peso, a altura de uma mulher são ditos variáveis aleatórias contínuas, pois podem assumir qualquer valor real entre uma medida e outra, dependendo da precisão do aparelho usado.

Em geral, Variáveis aleatórias são representadas por letras maiúsculas, como X, Y e Z e sua a probabilidade pode ser denotada por:

$$P[X] \quad ou \quad P[X = x]$$

7.3 Definição frequentista de probabilidade

A probabilidade se relaciona a eventos futuros ou que ainda não ocorreram, desta forma a probabilidade pode ser entendida como uma medida de incerteza em relação ao evento. A probabilidade de um evento ocorrer, em determinadas circunstâncias, pode ser definida como a proporção de vezes que o evento é observado quando o experimento é repetido um número infinitamente grande de vezes (83).

A chamada *Lei dos Grandes Números* diz que à medida que múltiplas observações são coletadas, a proporção observada de ocorrências de um determinado desfecho, após n ensaios, converge para a probabilidade real P desse desfecho. Ou seja, quanto mais vezes for repetido uma experiência, a melhor estimativa de probabilidade tende a ocorrer.

O resultado dos comandos abaixo simulam 1000 lançamentos de uma moeda, mostrando que quando chega próximo de 300 lançamentos, a probabilidade se mantém praticamente constante em torno de 50% (Figura 102).

```
x=1:1000
y=cumsum (sample (0:1,1000, rep=TRUE))
```

```
plot (x,y/1:1000,
      ylab="Probabilidade", xlab = "Lançamentos da moeda",
      ylim=c (0.3,0.8), xlim=c (0,1000),
      pch=16,
      col="steelblue")
abline(h = 0.5, col = "red", lty = 2)
```

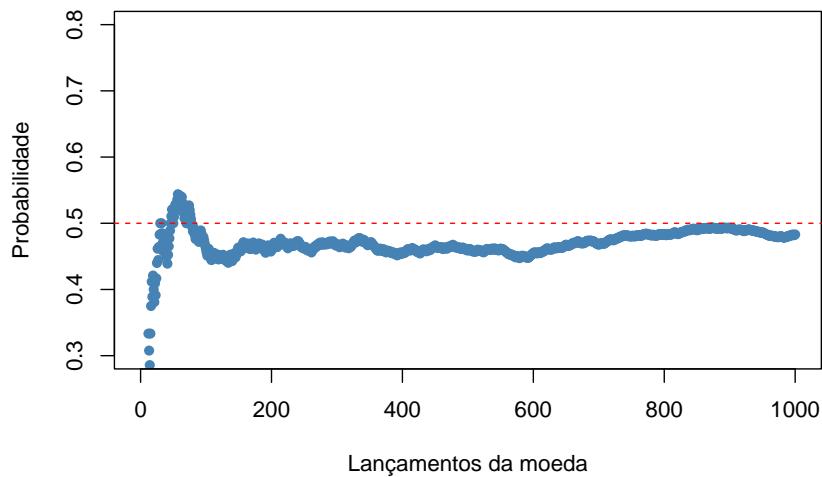


Figura 102: Simulação do lançamento de 1000 moedas.

A definição frequentista também pode ser aplicada a uma medida contínua como a altura de mulheres. No conjunto de dados dadosMater.xlsx, encontra-se o registro da altura de 1368 mulheres. Essas alturas serão selecionadas e colocadas em objeto wh (*women height*):

```
wh <- read_excel("dadosMater.xlsx") %>%
  select(altura)
summary (wh)
```

```
##      altura
##  Min.   :1.400
##  1st Qu.:1.550
```

```

## Median :1.600
## Mean   :1.598
## 3rd Qu.:1.650
## Max.   :1.850

```

A mediana da altura das gestantes é 1,60 m. Ou seja, metade dessas mulheres têm uma altura acima de 1,60 m. Em um longo conjunto de sorteios, a probabilidade de uma mulher ter altura acima de 1,60 m é 50%. O percentil 75 (3º quartil) é igual a 1,65 m, a probabilidade de estar acima deste valor, portanto, é 25%. É possível encontrar a probabilidade de a altura estar acima, abaixo ou entre quaisquer valores. Quando se faz a mensuração de uma variável contínua, fica-se limitado ao método usado, portanto, quando se diz que uma mulher tem 1,60 m, significa dizer que está entre 159,5 e 165,5 m, dependendo da precisão do instrumento de medição.

Logo, o interesse está na probabilidade de a variável aleatória assumir valores entre certos limites. A probabilidade de encontrar um valor exatamente de 1,60 m é quase igual a zero (na realidade, 2.7×10^{-124}).

Como se verá adiante, isto pode ser facilmente calculado no R:

```

# Distância do valor de 1,60 e a média em desvios padrão (escores Z)
z <- (1.60 - mean(wh$altura))/sd(wh$altura)

# Probabilidade
pnorm (z, mean(wh$altura),sd(wh$altura))

## [1] 7.387473e-127

```

7.4 Propriedades das probabilidades

As seguintes propriedades simples decorrem da definição de probabilidade.

Sendo E um evento aleatório, a $P[E]$ está entre 0 e 1, ou seja $0 \leq P[E] \leq 1$. Quando o evento certamente não ocorre, a probabilidade é 0, quando sempre ocorre a probabilidade é 1. Quando a probabilidade for igual a 0,50 tem-se máxima incerteza.

1. Regra de adição (regra do “ou”)

Dois eventos A e B são mutuamente exclusivos, ou seja, quando A acontece, B não pode acontecer. Então, a probabilidade de que um ou outro aconteça é a soma de suas probabilidades. Por exemplo, um dado lançado pode mostrar um ou dois, mas não ambos. A probabilidade de mostrar um ou dois é igual a $1/6 + 1/6 = 1/3$.

$$P[A \cup B] = P[A] + P[B]$$

Se A e B não são mutuamente exclusivos, ou seja, quando A acontece pode também ocorrer B. Por exemplo, o nascimento de uma menina pode ser concomitante com o fato de ser branca.

$$P[A \cup B] = P[A] + P[B] - P[A \cap B]$$

2. Regra de multiplicação (regra do “e”)

Suponha que dois eventos (A e B) sejam independentes, ou seja, saber que um aconteceu não nos diz nada sobre se o outro aconteceu. Então, a probabilidade de que ambos aconteçam é o produto de suas probabilidades. Por exemplo, suponha que jogamos duas moedas. Uma moeda não influencia a outra,

portanto os resultados dos dois lançamentos são independentes e a probabilidade de ocorrerem duas caras é $0,50 \times 0,50 = 0,25$.

$$P[A \text{ e } B] = P[A] \times P[B]$$

Se os eventos são dependentes, a probabilidade que ambos aconteçam é igual a:

$$P[A \text{ e } B] = P[A] \times P[B|A]$$

7.5 Distribuição de Probabilidades

Um conjunto de eventos que são mutuamente excludentes e que inclui todos os eventos que podem acontecer, é chamado de exaustivo. A soma de suas probabilidades é 1. O conjunto dessas probabilidades constitui uma *distribuição de probabilidade*.

Existem diversos modelos probabilísticos que procuram descrever vários tipos de variáveis aleatórias discretas ou contínuas. Estas distribuições também são chamadas de *modelos probabilísticos estocástico* que são definidas por duas funções matemáticas: a *função de probabilidade* (fp) para variáveis discretas, que atribui a cada valor a sua probabilidade de ocorrência ($P(X=x)$) e *função densidade de probabilidade* (fdp) para variáveis contínuas.

A função de probabilidade é a função que atribui probabilidades a cada um dos possíveis valores da variável aleatória discreta, usando, em geral, as frequências relativas, apresentadas em uma tabela de frequência. O *modelo de Bernoulli* ou *Binomial* e o *modelo de Poisson* são exemplos de modelo probabilístico de variáveis discretas.

A função densidade de probabilidade é a função que atribui probabilidade a qualquer intervalo de número reais, ou seja, um conjunto de valores não enumerável (infinito). Não é possível atribuir probabilidades para um determinado valor, é possível apenas para um intervalo. Por exemplo, o peso dos recém-nascidos. Para atribuir probabilidade a intervalos de valores é utilizada uma função e as probabilidades são representadas por áreas. Existem diversos modelos contínuos de probabilidade, mas o mais importante deles, é o *modelo normal*, também conhecido como *modelo gaussiano*.

7.6 Distribuição Normal

O *modelo probabilístico normal* ou *gaussiano* é extremamente importante em estatística, pois serve como um fundamento para técnicas de inferência. Variáveis como os pesos dos recém-nascidos a termo, as alturas das mulheres adultas, a renda familiar em reais e muitas outras variáveis, na natureza, se ajustam ao modelo da distribuição normal.

O modelo de distribuição normal sempre descreve uma curva simétrica, unimodal e em forma de sino (Figura 103).

```
x <- seq(-4, 4, length=1000)
y <- dnorm(x)
plot(x,y,
      type = "l",
      lwd = 2,
      axes = TRUE,
      xlab = "X", ylab = "Densidade de Probabilidades")
```

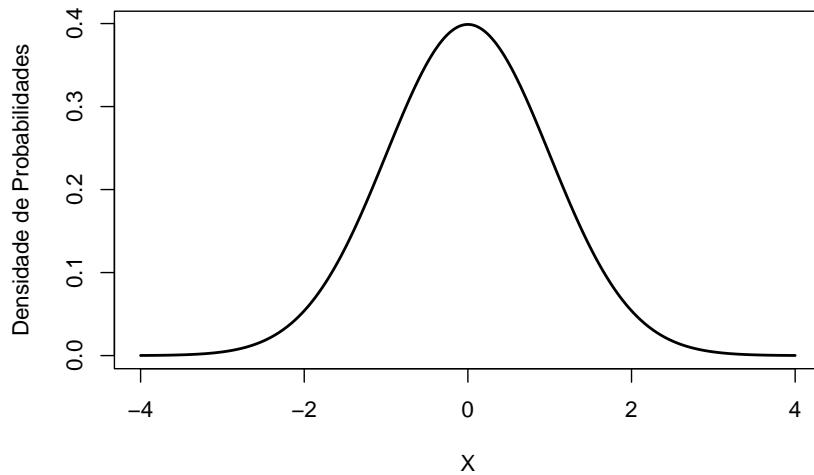


Figura 103: Curva normal.

No entanto, essas curvas podem parecer diferentes dependendo dos detalhes do modelo. Especificamente, o modelo de distribuição normal pode ser ajustado usando dois parâmetros: *média* e *desvio padrão*.

Como é fácil prever, alterar a média desloca a curva de sino para a esquerda ou para a direita, enquanto a alteração do desvio padrão estende ou achata a curva, ou seja, muda a dispersão da distribuição.

A Figura 104, mostra a distribuição normal com média 0 e desvio padrão 1, na curva à direita, a distribuição normal com média 1.5 e desvio padrão 1. Sobrepondo-se à curva da esquerda observa-se uma curva mais achatada (verde) que tem média 0 e desvio padrão 1.5. Observa-se, como mencionado, que modificando os parâmetros da curva, altera-se a posição ou o formato da curva.

```
curve (dnorm (x,
               mean=0,
               sd=1),
      col="dodgerblue3",
      lty=1,
      lwd=2,
      ylim = c(0, 0.4),
      xlim = c(-4.5, 4.5),
      ylab = "Densidade",
      xlab = "X",
      bty = "n")
box(bty = "L")
abline (v= 0, lwd = 1, lty = 2, col = "dodgerblue3")

curve (dnorm (x,
               mean=0,
               sd=1.5),
      col="darkolivegreen3",
      lty=1,
      lwd=2,
      add=T)
```

```

curve (dnorm (x,
               mean=1.5,
               sd=1),
       col="firebrick3",
       lty=1,
       lwd=2,
       add=T)
abline (v= 0, lwd = 1, lty = 2, col = "firebrick3")

```

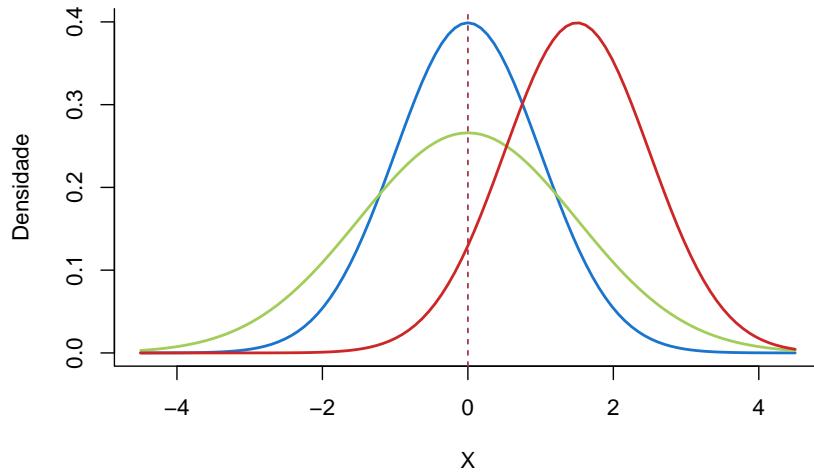


Figura 104: Curvas normais com modificação dos parâmetros.

7.6.1 Características da distribuição normal

A curva normal apresenta as seguintes características:

- A média e o desvio padrão descrevem exatamente uma distribuição normal, eles são chamados de parâmetros da distribuição. Se uma distribuição normal tem média μ e desvio padrão σ , pode-se escrever a distribuição como $N(\mu, \sigma)$. As três distribuições do gráfico anterior. podem ser escritas como:

$$N(\mu = 0, \sigma = 1) , N(\mu = 0, \sigma = 1.5) \text{ e } N(\mu = 1.5, \sigma = 1)$$

- Na distribuição normal, a média, a mediana e a moda coincidem.
- A curva normal é simétrica em torno da média (μ).
- As extremidades da curva, em ambos os lados da média, se estendem cada vez mais próximas do eixo x (abscissa) sem jamais tocá-lo. É assintótica.
- Os pontos de inflexão da curva são $\mu - \sigma$ e $\mu + \sigma$.
- A área total sob a curva é 1 ou 100%.

7.6.2 Distribuição normal padronizada

Cada variável aleatória contínua tem a sua média e seu desvio padrão e, portanto, a sua curva normal correspondente.

Para facilitar a comparação entre variáveis, foi criado o conceito de **curva normal padronizada**, que é uma curva normal com média 0 e desvio padrão 1. A distribuição normal padrão também pode ser chamada de *distribuição normal centrada ou reduzida*.

Para calcular probabilidades associadas a distribuição normal, costuma-se converter a variável aleatória original X , em unidades reduzidas ou padronizadas, denominadas de **escore Z **.

Esta transformação é realizada pela equação que indica o número de desvios padrão envolvidos no afastamento do valor x em relação à média:

$$z = \frac{x - \mu}{\sigma}$$

onde:

- z → escore z
- x → valor qualquer da variável aleatória X
- μ → média da variável X
- σ → desvio padrão da variável X

Qualquer distribuição de uma variável aleatória normal pode ser padronizada, usando o escore z . Isto permite que se calcule a probabilidade de se encontrar determinados intervalos de valores (85).

A altura das puérperas do conjunto de dados `dadosMater.xlsx` tem as seguintes medidas resumidoras:

```
library (dplyr)

mater <- readxl::read_excel("dadosMater.xlsx") %>%
  select(altura) %>%
  dplyr::summarise(n = n(),
                   media = mean(altura, na.rm = TRUE),
                   dp = sd(altura, na.rm = TRUE),
                   min = min(altura, na.rm = TRUE),
                   max = max(altura, na.rm = TRUE))
mater

## # A tibble: 1 x 5
##       n   media     dp    min    max
##   <int>   <dbl>   <dbl>   <dbl>   <dbl>
## 1 1368    1.60 0.0655    1.4    1.85
```

Desta forma, pode-se verificar quantos desvios padrão uma mulher que mede 1,725m, pertencente a esta população, está afastada da média de 1.598m. Assim:

```
z <- (1.725 - mater$media)/mater$dp
z
```

```
## [1] 1.940054
```

Esta mulher está distante praticamente 2 desvios padrão acima da média da sua população. Portanto, ela é considerada alta. Por que?

Para responder a essa pergunta, há necessidade de calcular a probabilidade de encontrar uma mulher com esta altura, nesta população.

No R, existem as funções `dnorm()`, `pnorm()` e `qnorm()`, que permitem calcular a *densidade de probabilidade*, *distribuição cumulativa* e *função quantílica da distribuição normal* para um conjunto de valores. Além disso, a função `rnorm()` permite obter observações aleatórias que seguem uma distribuição normal (86).

Função `pnorm()`

A função `pnorm()` fornece a *Função de Distribuição Cumulativa* (CDF) da distribuição Normal, que é a probabilidade de que a variável X contenha um valor menor ou igual a x .

- **Sintaxe:** `pnorm(q, mean = 0, sd = 1, lower.tail = TRUE)`
- **Argumentos:**
 - q : vetor de quantis
 - $mean$: média
 - sd : desvio padrão
 - $lower.tail$: Se `TRUE`, as probabilidades são ($P \leq x$), caso contrário $P(X > x)$

Se for usado $mean = 0$ e $sd = 1$, o valor de $q = z$, caso contrário, toma-se os valores da média, o desvio padrão da população e o valor de x .

Com esta função pode-se responder a pergunta feita anteriormente em relação a probabilidade de encontrar uma mulher com mais de 1,725m, equivalente a 1.9400545 desvios padrão acima da média, em uma população com média = 1.5979678 e desvio padrão = 0.0654787.

```
p <- pnorm(z, mean = 0, sd = 1, lower.tail = FALSE)  
p
```

```
## [1] 0.02618654
```

Ou, usando os valores:

```
pnorm(1.725, mean = mater$media, sd = mater$dp, lower.tail = FALSE)
```

```
## [1] 0.02618654
```

Observa-se que, nesta população, apenas 2.6% das mulheres têm acima de 1,725m, razão de considerar-se uma mulher acima deste valor como sendo alta. Ou seja, é pouco provável encontrar mulheres acima dessa altura, nesta população. A Figura 105 representa com clareza esta pequena probabilidade.

```
source("normal_area.R")  
normal_area(media = 0, dp = 1, linf = 1.94, lsup = 3, cor = "tomato", lwd = 2 )  
text(2.6, 0.05, "2.6%")
```

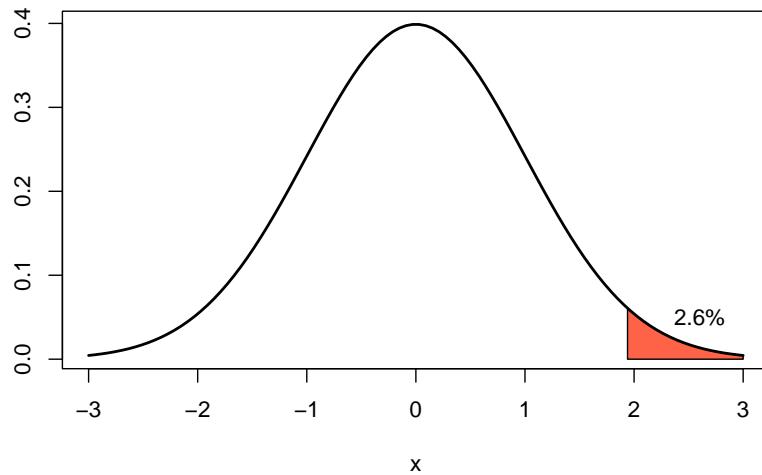


Figura 105: Probabilidade de encontrar mulheres com mais de 1,725m

A função `normal_area()` é uma função criada para desenhar a uma curva normal com a área da probabilidade desejada colorida. Ela pode ser obtida [aqui](#) e baixada no seu diretório. Foi usada a função `text()` para escrever o valor da probabilidade.

Função `qnorm()`

A função `qnorm()` permite encontrar o quantil (percentil) q para qualquer probabilidade p . Portanto, a função `qnorm` é o inverso da função `pnorm`. A sintaxe do `qnorm` é a seguinte:

- **Sintaxe:** `qnorm(p, mean = 0, sd = 1, lower.tail = TRUE)`
- **Argumentos:**
 - p : vetor de probabilidades
 - `mean`: média
 - `sd`: desvio padrão
 - `lower.tail`: Se `TRUE`, as probabilidades são ($P \leq x$), caso contrário $P(X > x)$

No exemplo anterior, a probabilidade de se encontrar mulheres, na maternidade, com mais de 1,725m foi de 2.6%. Poderia ser calculado com a função `qnorm()` qual o escore z correspondente:

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

```
## [1] 1.940054
```

Outro exemplo, na mediana ($p = 0,5$), o escore z é igual a:

```
qnorm(0.50, mean = 0, sd = 1)
```

```
## [1] 0
```

Função `dnorm()`

Essa função retorna o valor da função de densidade de probabilidade (pdf) da distribuição normal dada uma certa variável aleatória X , uma média populacional μ e o desvio padrão populacional σ .

- **Sintaxe:** `dnorm(x, mean = 0, sd = 1)`
- **Argumentos:**
 - x : vetor de quantis
 - $mean$: média
 - sd : desvio padrão

Embora x represente a variável independente da pdf para a distribuição normal, também é útil pensar em x como um escore z . Por exemplo, a densidade de probabilidade quando $x = 0$ é igual:

```
dnorm(x = 0, mean = 0, sd = 1)
```

```
## [1] 0.3989423
```

Agora, para melhor compreensão, será mostrado o que foi dito, representando a função de densidade de probabilidade da distribuição normal com o `dnorm()`.

Inicialmente, será construído um vetor de escores z :

```
escores_z <- seq(-3,3, by = 0.1)
escores_z
```

```
## [1] -3.0 -2.9 -2.8 -2.7 -2.6 -2.5 -2.4 -2.3 -2.2 -2.1 -2.0 -1.9 -1.8 -1.7 -1.6
## [16] -1.5 -1.4 -1.3 -1.2 -1.1 -1.0 -0.9 -0.8 -0.7 -0.6 -0.5 -0.4 -0.3 -0.2 -0.1
## [31] 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0 1.1 1.2 1.3 1.4
## [46] 1.5 1.6 1.7 1.8 1.9 2.0 2.1 2.2 2.3 2.4 2.5 2.6 2.7 2.8 2.9
## [61] 3.0
```

Um objeto, `valores_d`, receberá os valores das densidades de probabilidade gerados com a função `dnorm()`, usando os `escores_z`:

```
valores_d <- dnorm(escores_z, mean = 0, sd = 1)
```

Estes valores serão plotados para construir a curva normal (Figura 106):

```
plot(valores_d,
      type = "l",                                # Tipo de gráfico em linha
      lwd = 2,                                    # Espessura da linha 2x padrão
      col = "steelblue",                           # Cor da linha
      xaxt = "n",                                 # Eixo x sem rótulos
      ylab = "Densidade de Probabilidade",
      xlab = "Escores z")

# Rótulos do eixo x
axis(1, at = which(valores_d == dnorm(0)), labels = c(0))
axis(1, at=which(valores_d == dnorm(1)), labels=c(-1, 1))
axis(1, at=which(valores_d == dnorm(2)), labels=c(-2, 2))
axis(1, at=which(valores_d == dnorm(3)), labels=c(-3, 3))
```

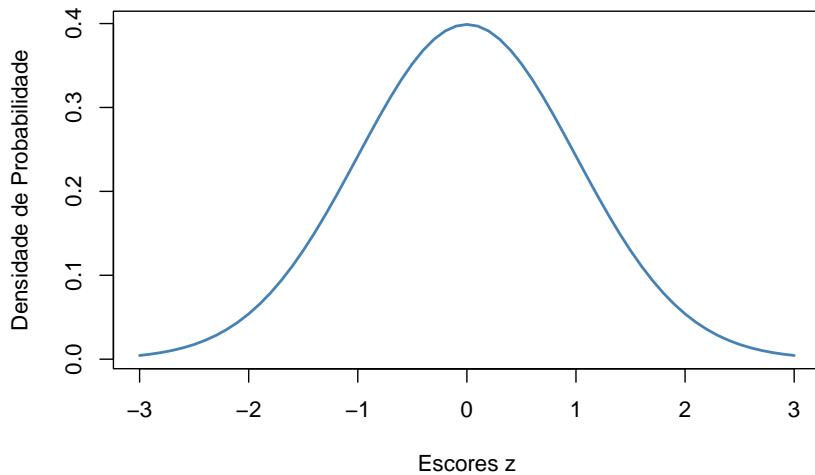


Figura 106: Função densidade de probabilidade.

Como se pode ver, `dnorm()` fornece a “altura” do pdf da distribuição normal em qualquer escore z que se forneça como argumento.

Função `rnorm()`

A função `rnorm()` gera n números aleatórios com distribuição normal com média μ e desvio padrão σ . A sintaxe da função `rnorm()` no R é a seguinte:

- **Sintaxe:** `qnorm(n, mean = 0, sd = 1)`
- **Argumentos:**
 - n : número de observações a serem geradas
 - $mean$: média
 - sd : desvio padrão

Com esta função é possível, por exemplo, gerar 10 observações de uma distribuição normal:

```
rnorm(10)
```

```
## [1] 1.3982554 0.2799928 -0.7454283 1.0739554 -2.0961584 0.1211288
## [7] 0.8555645 0.2833571 -0.5407017 -1.1418455
```

No entanto, deve-se notar que, se não especificar uma “semente” (`seed`), a saída não será reproduzível:

```
rnorm(10)
```

```
## [1] -0.69788312 -1.28414234 -1.46902384 -0.54235084 0.32410527 -0.03439035
## [7] 0.41482006 0.85759091 -0.80023932 -1.09969537
```

Pode-se usar a função `set.seed()` para tornar o código reproduzível. O valor da “semente” (número) não é importante desde que seja consistente na sua utilização. O que é verdadeiramente importante é que o código seja reproduzido fielmente.

Para ilustrar, será construído dois conjuntos de 10 números que serão recebidos pelos objetos `x` e `y`. Para gerar o conjunto de números `x`, sera usado o número 123 como “semente”. A “semente” funciona como uma espécie de marca. Para o `y` não será usado a `set.seed()`.

```
n <- 10

set.seed (123)
x <- rnorm (n)
x

## [1] -0.56047565 -0.23017749  1.55870831  0.07050839  0.12928774  1.71506499
## [7]  0.46091621 -1.26506123 -0.68685285 -0.44566197

y <- rnorm(n)
y

## [1]  1.2240818  0.3598138  0.4007715  0.1106827 -0.5558411  1.7869131
## [7]  0.4978505 -1.9666172  0.7013559 -0.4727914
```

Comparando os conjuntos com a função `identical()` do R base, observa-se que os conjuntos são diferentes:

```
identical(x, y)
```

```
## [1] FALSE
```

Agora, repetindo os mesmos comandos, mas usando antes a mesma “semente”:

```
set.seed (123)
x <- rnorm (n)
x

## [1] -0.56047565 -0.23017749  1.55870831  0.07050839  0.12928774  1.71506499
## [7]  0.46091621 -1.26506123 -0.68685285 -0.44566197

set.seed (123)
y <- rnorm(n)
y

## [1] -0.56047565 -0.23017749  1.55870831  0.07050839  0.12928774  1.71506499
## [7]  0.46091621 -1.26506123 -0.68685285 -0.44566197

identical(x, y)
```

```
## [1] TRUE
```

Observa-se, agora, que temos conjuntos idênticos.

Agora, para usar `rnorm()`, serão gerados três vetores diferentes de números aleatórios de uma distribuição normal.

```

set.seed(1234)
n10 <- rnorm(10, mean = 0, sd = 1)
n100 <- rnorm(100, mean = 0, sd = 1)
n10000 <- rnorm(10000, mean = 0, sd = 1)

```

A seguir, serão construídos histogramas (Figura 107), onde se pode observar que, aumentando o número de observações, tem-se gráficos que irão progressivamente se aproximando da verdadeira função de densidade normal.

```

# Este comando coloca os gráficos em uma mesma linha, o argumento mfrow(c(1,3)) diz ao R para construir
par(mfrow=c(1,3))

# Histogramas
hist(n10, breaks = 5, main = "n =10", ylab = "Frequência")
hist(n100, breaks = 20, main = "n =100", ylab = "Frequência")
hist(n10000, breaks = 50, main = "n =10000", ylab = "Frequência")

```

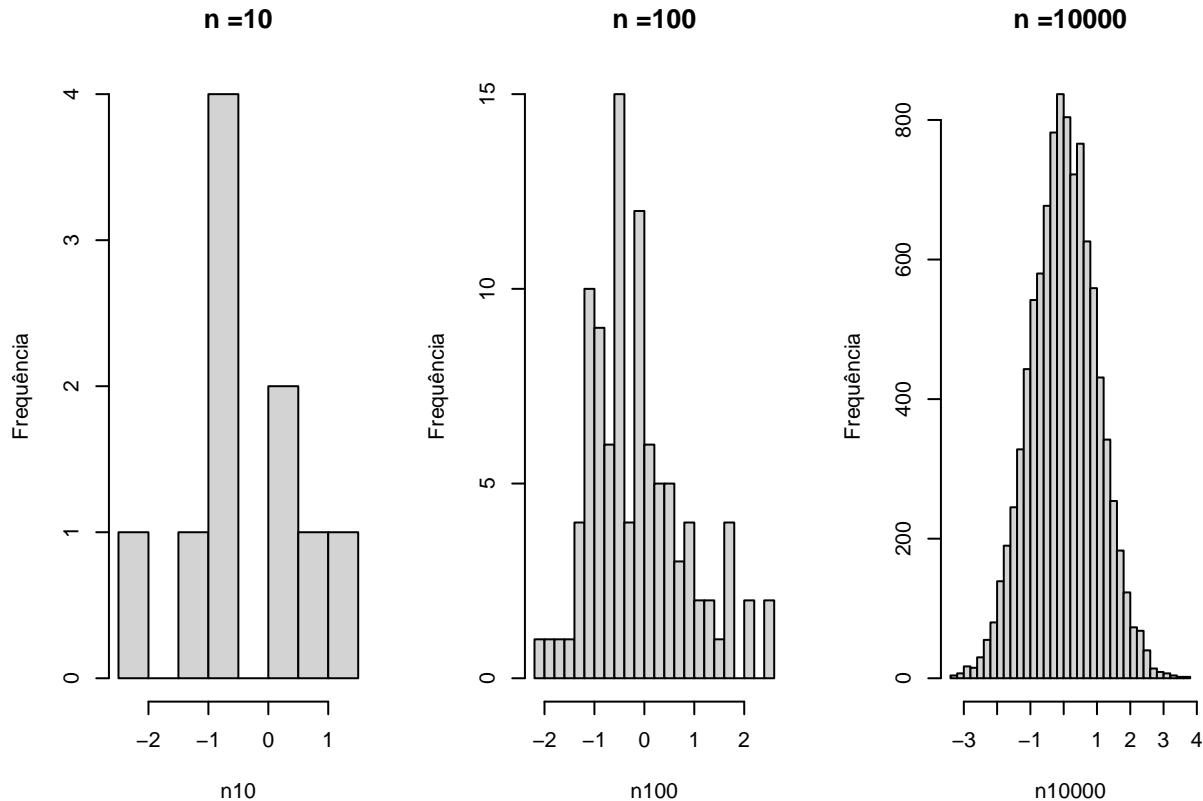


Figura 107: Histogramas construídos com amostras geradas pela função rnorm.

```

# Restaura as configurações basais de plotagem
par(mfrow=c(1,1))

```

7.6.3 Regra Empírica 68-95-99.7

A regra empírica diz que, se uma população de um conjunto de dados tem uma distribuição normal com média 0 e desvio padrão 1 ($X \sim \text{Norm}(0,1)$) pode-se afirmar que aproximadamente, 68%, 95% e 99,7% dos valores encontram-se, respectivamente, dentro de ± 1 , ± 2 e ± 3 desvio padrão acima e abaixo média.

Esta regra pode ser usada para descrever uma população e ajudar a decidir se uma amostra de dados veio de uma distribuição normal. Se uma amostra é grande o suficiente e a observação do histograma tem um formato parecido com um sino, é possível verificar se os dados seguem as especificações 68-95-99,7%. Se sim, é razoável concluir que os dados vieram de uma distribuição normal.

Usando a amostra dos recém-nascidos a termo da maternidade-escola do Hospital Geral de Caxias do Sul e foi observado o histograma com uma curva normal sobreposta, tem-se

```
# Selecionando os recém-nascidos a termo
mater <- readxl::read_excel("dadosMater.xlsx")%>%
  filter(ig >= 37 & ig < 42)

# Média dos pesos dos recém-nascidos a termo
media <- mean(mater$pesoRN, na.rm =TRUE)
media

## [1] 3216.316

# Desvio padrão dos pesos dos recém-nascidos a termo
dp <- sd(mater$pesoRN, na.rm =TRUE)
dp

## [1] 462.1205
```

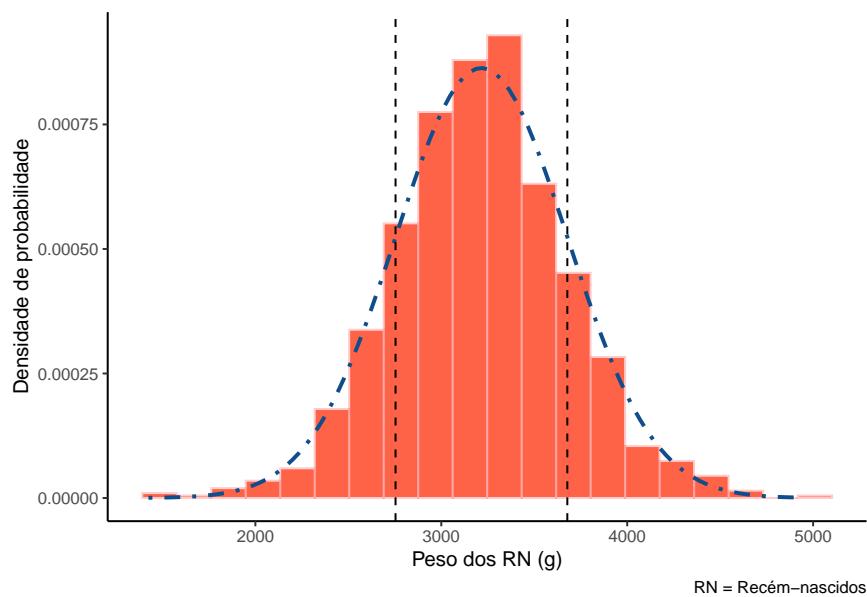


Figura 108: Histograma com curva normal sobreposta

e pode-se aceitar que a distribuição é aproximadamente normal (Figura 108). Consequentemente, 68% desses bebês pesam entre 2754.2 e 3678.4g (média ± 1 desvio padrão). As linhas verticais tracejadas são apenas para melhorar a visualização, pois não se necessita, praticamente, de nenhum cálculo.

7.6.4 Exercitando o raciocínio com a curva normal

1. Suponha-se que em uma determinada região existam duas populações etnicamente diferentes onde as mulheres têm as seguintes medidas de altura: *população 1* tem $\mu = 160$ cm e $\sigma = 6,6$ cm e a *população 2* tem $\mu = 139$ cm e $\sigma = 6,6$ cm. Essas duas populações vivem misturadas e têm o mesmo aspecto físico, podendo ser distinguidas apenas geneticamente.

- A qual população pertence uma mulher de 150 cm?

Probabilidade de pertencer à População 1

```
x <- 150
mu1 <- 160
sigma1 <- 6.6

z1 <- (x - mu1)/sigma1
z1
```

```
## [1] -1.515152
```

```
p1 <- pnorm (z1)
p1
```

```
## [1] 0.06486702
```

Ou seja, na população 1, apenas 6.5% das mulheres tem altura abaixo de 1,50, 93.5% é mais alta do que este valor.

Probabilidade de pertencer à População 2

```
x <- 150
mu2 <- 140
sigma2 <- 6.6

z2 <- (x - mu2)/sigma2
z2
```

```
## [1] 1.515152
```

```
p2 <- pnorm (z2)
p2
```

```
## [1] 0.935133
```

Na população 2, -92.5% das mulheres têm altura acima de 150 cm. Este valor valor está 1.52 desvios padrão distante da média. Isto significa que se ela pertence a população 2, ela seria considerada alta, quer dizer, praticamente 6.5% das mulheres desta população são menores do que ela.

No gráfico abaixo, pode-se visualizar a posição de uma mulher de 1,50 m (linha vermelha tracejada) em relação às duas populações (109).

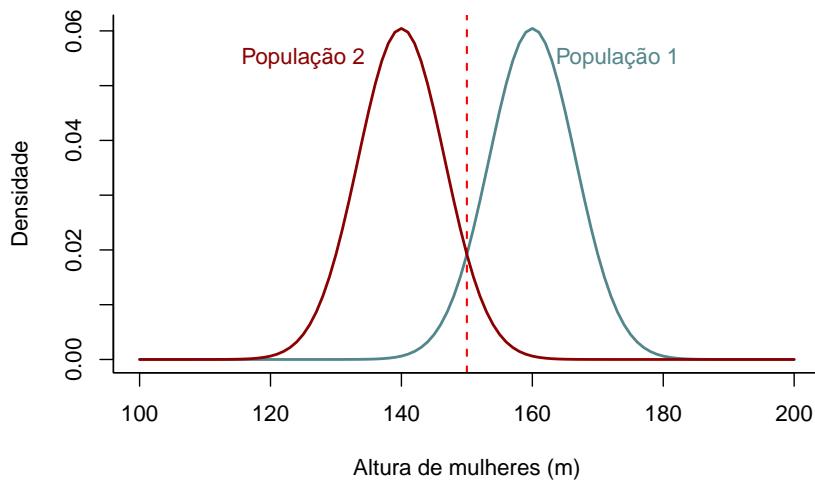


Figura 109: Posição de uma mulher com 1,50m comparando duas populações

Concluindo, ela pode pertencer a qualquer uma das populações. Pode ser uma mulher alta da população 2 ou uma “baixinha” da população 1!

2. Usando as populações do exercício 1, qual a probabilidade de se encontrar mulheres, em qualquer das populações, abaixo do escore $z = -1.96$?

```
pnorm (-1.96)
```

```
## [1] 0.0249979
```

3. Usando as populações do exercício 1, qual a probabilidade de se encontrar mulheres, em qualquer das populações, acima do escore $z = 1.96$?

```
pnorm (1.96, lower.tail = FALSE)
```

```
## [1] 0.0249979
```

Em outras palavras, se forem observadas as respostas das perguntas 2 e 3, chega-se a conclusão que entre os escores $z = -1.96$ e 1.96 encontram-se 95% das mulheres de qualquer população cujo parâmetro tem distribuição normal. Na “regra empírica 68-95-99.7” usou-se o valor de 1,96 arredondado para 2.

4. Qual é o escore z do 50º percentil da distribuição normal?

```
qnorm (0.50)
```

```
## [1] 0
```

5. Qual o escore z para o 97,5º percentil da distribuição normal?

```
qnorm (0.975)
```

```
## [1] 1.959964
```

7.7 Distribuição Binomial

A distribuição normal padrão é apenas um dos exemplos de distribuição de probabilidade. Uma boa parte das situações se ajustam a ela. Entretanto, diversas situações reais muitas vezes se aproximam de outras distribuições estocásticas definidas por algumas hipóteses. Daí a importância de se conhecer e manipular algumas destas distribuições. Entre elas, a **distribuição binomial**.

Quando um experimento aleatório resulta em um de dois, mutuamente exclusivos, desfechos, tais como vivo/morto, positivo/negativo, sim/não, masculino/feminino é denominado de *Ensaio de Bernoulli*. Recebeu esta denominação em homenagem ao matemático suíço, Jacob Bernoulli (1654-1705), considerado fundador do cálculo e da teoria da probabilidade (87).

A distribuição de frequências que descreve as proporções de um ensaio de Bernoulli, chama-se *Distribuição Binomial*. A probabilidade binomial dá a probabilidade de determinado desfecho ocorrer em determinado número de ensaios independentes. Uma sequência de ensaios de Bernoulli forma um *Processo de Bernoulli*.

A distribuição binomial é importante para variáveis discretas. Existem poucas condições que precisam ser atendidas antes se considere uma variável aleatória para distribuição binomial:

- Cada ensaio resulta em um de dois desfechos, mutuamente exclusivos, denominados, arbitrariamente, de sucesso e fracasso;
 - A probabilidade de sucesso é fixa, igual a p , constante em cada ensaio, e a probabilidade de fracasso é igual a $1 - p$;
 - O número de repetições n em um ensaio é fixo.
- Os ensaios são independentes

A distribuição binomial é na verdade uma família de distribuições, cujos membros são definidos pelos valores de n e p (parâmetros da distribuição binomial).

A probabilidade de sucesso¹⁰, em uma distribuição binomial, é dada pela fórmula:

$$P(X = x) = C \times p^x \times (1 - p)^{n-x}$$

onde n = ensaios, x = sucessos, p = probabilidade de um sucesso e C representa o número possível de combinações em um ensaio.

O número de combinações, C de x sucessos entre n repetições podem ser computado pela fórmula

$$C = \frac{n!}{x!(n-x)!}$$

ou, no R, com a função `choose (n, x)`.

O modelo de distribuição binomial trata de encontrar a probabilidade de sucesso de um evento que tem apenas dois resultados possíveis em uma série de experimentos. Usando dados de uma distribuição binomial, é possível calcular os valores esperados de uma variável aleatória conforme ela passa por tentativas independentes.

¹⁰ Sucesso, aqui, não está no sentido de vitória, êxito, triunfo, glória e sim como obter o desfecho esperado. Por exemplo, se uma moeda é lançada e se espera obter cara, sucesso significa um resultado igual a cara.

Em outras palavras, é possível prever o número exato de caras ou coroas que se deve esperar ao jogar uma moeda um certo número de vezes.

Também, pode-se usar a probabilidade binomial cumulativa para encontrar a probabilidade de obter um determinado intervalo de resultados. Por exemplo, saber a probabilidade do nascimento de até três meninos em 10 nascimentos consecutivos quando a probabilidade de nascer um menino é 0,50.

O R tem quatro funções embutidas para gerar distribuição binomial. Elas são descritas a seguir.

Função `pbinom()`

Esta função retorna o valor da *função de densidade cumulativa* (cdf) da distribuição binomial dada uma certa variável aleatória q , número de tentativas (`size`) e probabilidade de sucesso em cada tentativa (`prob`).

- **Sintaxe:** `pbinom(q, size, prob, lower.tail = TRUE)`
- **Argumentos:**
 - q : vetor de quantis
 - $size$: número de ensaios
 - $prob$: probabilidade de sucesso em cada ensaio
 - $lower.tail$: Se `TRUE`, as probabilidades são $(P \leq x)$, caso contrário $P(X > x)$

Por exemplo, qual é a probabilidade de nascer até três meninos em cinco nascimentos, sabendo que a probabilidade de nascer um menino é igual a 0,50?

```
pbinom (3, 5, 0.50)
```

```
## [1] 0.8125
```

Isso corresponde a soma das probabilidades de nascer nenhum menino, um menino, dois meninos e três meninos (Figura 110). Isto é calculado pela equação $P(X = x)$, vista anteriormente:

```
n = 5
p = 0.50
x <- 0:5
# Probabilidades de meninos
Fx <- (factorial(n)/(factorial(x)*factorial(n-x)))* p^x *(1-p)^(n-x)
Fx
```

```
## [1] 0.03125 0.15625 0.31250 0.31250 0.15625 0.03125
```

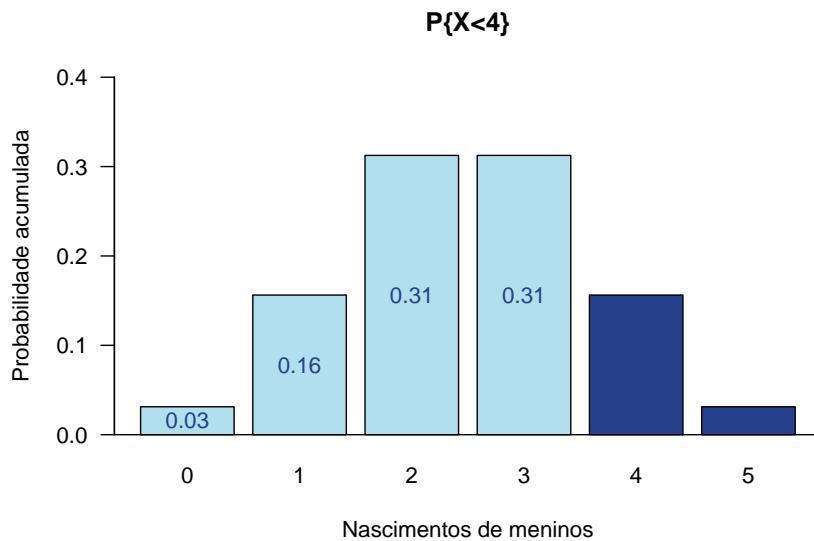


Figura 110: Distribuição binomial, mostrando a $P(x < 4)$ com $n = 5$ e $p = 0.50$

```
## [1] 0.8125
```

Função `qbinom()`

Esta função retorna o valor da função de densidade cumulativa inversa (cdf) da distribuição binomial dada uma certa variável aleatória q , número de tentativas (`size`) e probabilidade de sucesso em cada tentativa (`prob`). Com o uso desta função, podemos descobrir o quantil da distribuição binomial.

- **Sintaxe:** `qbinom(p, size, prob, lower.tail = TRUE)`
- **Argumentos:**
 - `p`: probabilidade ou vetor de probabilidades
 - `size`: numero de ensaios
 - `prob`: probabilidade de sucesso em cada ensaio
 - `lower.tail`: Se `TRUE`, as probabilidades são $(P \leq x)$, caso contrário $P(X > x)$

Por exemplo, quantos meninos nascerão em 5 partos com 81.25% de probabilidade cumulativa?

```
qbinom (0.8125, size = 5, prob = 0.50)
```

```
## [1] 3
```

Função `rbinom()`

A função `rbinom()` permite extrair n observações aleatórias de uma distribuição binomial. Os argumentos da função são descritos abaixo:

- **Sintaxe:** `qbinom(n, size, prob)`
- **Argumentos:**

- *n*: número de observações aleatórias a ser gerado
- *size*: numero de ensaios
- *prob*: probabilidade de sucesso em cada ensaio

Se há necessidade de fazer uma simulação de 1000 amostras aleatoriamente, de tamanho 5 e a probabilidade de nascer menino (0,50):

```
menino <- rbinom(n = 1000, size = 5, prob = 0.5)
mean(menino)
```

```
## [1] 2.536
```

No entanto, se não for especificado uma “semente” (*seed*) antes de executar a função, será obtido um conjunto diferente de observações aleatórias a cada execução e , portanto, a média a cada execução será diferente. Para tornar a saída reproduzível, pode-se definir uma “semente” da seguinte maneira:

```
set.seed(23)
menino <- rbinom(n = 1000, size = 5, prob = 0.5)
mean(menino)
```

```
## [1] 2.515
```

Quanto maior o número de variáveis aleatória criadas, mais próximo a média do número de sucessos estará do número esperado de sucessos que é igual ao número de sucessos vezes a probabilidade de sucesso em cada ensaio (2.5)

Função `dbinom()`

Essa função retorna o valor da função de densidade de probabilidade (pdf) da distribuição binomial dada uma determinada variável aleatória X, número de tentativas (*size*) e probabilidade de sucesso em cada tentativa (*prob*). A função tem a seguinte sintaxe:

- **Sintaxe:** `dbinom(x, size, prob)`
- **Argumentos:**
 - *x*: vetor de números
 - *size*: numero de ensaios
 - *prob*: probabilidade de sucesso em cada ensaio

A função é usada para encontrar a probabilidade de um determinado valor para dados que seguem a distribuição binomial, ou seja, encontra $P(X=x)$, probabilidade de x sucessos em tentativas de tamanho (*size*) n quando a probabilidade (p) de sucesso é prob. Obtém o mesmo resultado da fórmula:

$$P(X = x) = C \times p^x \times (1 - p)^{n-x}$$

Por exemplo, no nascimento de uma criança, as duas possibilidades, menino ou menina, são mutuamente excludentes e esses são os únicos eventos que podem acontecer. A probabilidade de nascimento de menino, como visto, é 0,50, qual seria a probabilidade de nascerem 4 meninos em 5 partos consecutivos (Figura 111)?

```
dbinom(4, size = 5, prob = 0.50)
```

```
## [1] 0.15625
```

```
# Probabilidades de nascer meninos em 5 nascimentos
Fx <- dbinom(0:5, 5, 0.50)
Fx
```

```
## [1] 0.03125 0.15625 0.31250 0.31250 0.15625 0.03125
```

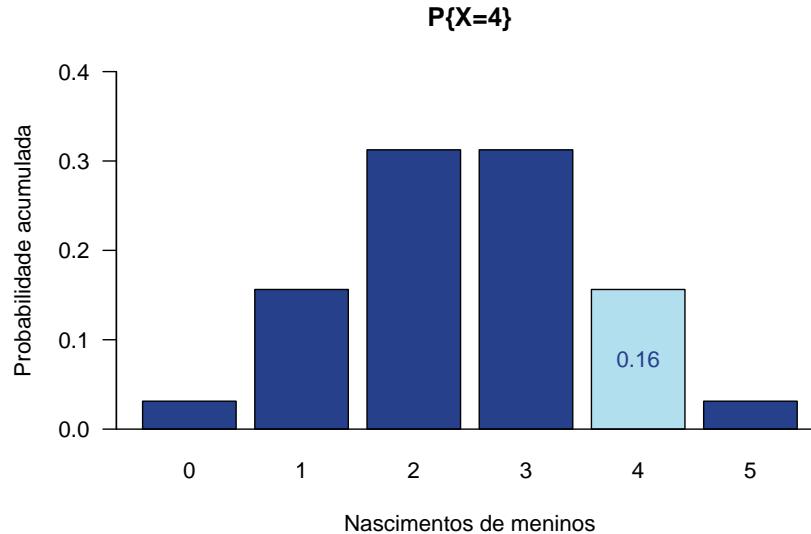


Figura 111: Distribuição binomial para P (x = 4) com n = 5 e p = 0,50

7.7.1 Média e desvio padrão da distribuição binomial

Quando o número de repetições é grande, geralmente há necessidade de resumir as probabilidades. A distribuição binomial pode ser descrita por sua *média* e *variância*.

A média é o valor médio da variável aleatória em um longo número de repetições. É também chamada de *valor esperado* ou *expectativa*. A expectativa de uma variável aleatória X, geralmente, é denotada por $E(X)$ e obtida pela multiplicação do número de ensaios independentes (n) pela probabilidade (p) de sucesso em cada ensaio:

$$\mu = E(X) = n \times p$$

Portanto, a expectativa (esperança) de nascimento de meninos em 5 partos é $E(X) = 5 \times 0,50 = 2,5$, como visto na função `rbinom()`. Observe que o valor esperado de uma variável aleatória discreta não tem um valor que a variável aleatória pode realmente assumir.

Por exemplo, para o número médio de meninos em um parto, ou não se tem menino ou se tem 1 menino, cada uma possibilidade com probabilidade de 0,50 e o valor esperado é $(0 \times 0,50) + (1 \times 0,50) = 0,50$. O número de meninos deve ser 0 ou 1, mas o valor esperado é a metade, a média que se obteria no longo prazo.

A variância de uma variável aleatória discreta X é igual a

$$\sigma^2 = var(X) = n \times p \times (1 - p)$$

Consequentemente, o desvio padrão é igual a

$$\sigma = \sqrt{var(X)} = \sqrt{n \times p \times (1 - p)}$$

Para o exemplo de 5 nascimentos, a média foi de 2,5 meninos e o desvio padrão

$$\sigma = \sqrt{5 \times 0.50 \times (1 - 0.50)} = \sqrt{2.5 \times 0.50} = 1.12$$

Portanto, se espera que ocorram em média 2,5 ($\sigma = 1.12$) nascimentos de meninos em 5 partos.

7.8 Distribuição de Poisson

A distribuição de Poisson é utilizada para descrever a probabilidade do número de ocorrências em um intervalo contínuo (de tempo ou espaço). No caso da distribuição binomial, a variável de interesse é o número de sucessos em um intervalo discreto (n ensaios de Bernoulli).

A unidade de medida (tempo ou espaço) é uma variável contínua, mas a variável aleatória, o número de ocorrências}, é discreta. Esta distribuição segue as mesmas premissas da distribuição binomial:

- as tentativas são independentes;
- a variável aleatória é o número de eventos em cada amostra;
- a probabilidade é constante em cada intervalo

Ela é utilizada para modelar eventos discretos que ocorrem com pouca frequência no tempo ou espaço, por isso é algumas vezes denominada de *distribuição de eventos raros*. Pode-se usar a distribuição de Poisson como uma aproximação da distribuição Binomial quando n , o número de tentativas, for grande e p ou $(1 - p)$ for pequeno (eventos raros).

Um bom princípio básico é usar a distribuição de Poisson quando $n \geq 20$ e $n \times p$ ou $n \times (1 - p) < 5\%$ (88). Nessas condições, a probabilidade que uma variável aleatória X adote um valor x é

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

onde λ (lambda) representa o número de ocorrências de um evento em um intervalo de tempo e é conhecida como parâmetro da distribuição de Poisson e é igual em média a $n \times p$.

No R, essa probabilidade é dada pela função `dpois(x, lambda)`.

Exemplo: Suponha que a probabilidade de uma puérpera ter infecção congênita (rubéola) seja igual a 0,0009. Qual seria a probabilidade, em uma população de 6000 gestantes, de que 5 estejam infectadas?

```
p <- 0.0009
x <- 5
n <- 6000
lambda <- n * p
P <- dpois(x, lambda)
round (P, 3)
```

```
## [1] 0.173
```

Portanto, a probabilidade de se encontrar 5 mulheres com infecção congênita é de aproximadamente 17%.

8 Assimetria e Curtose

8.1 Assimetria

A assimetria analisa a proximidade ou o afastamento de um conjunto de dados quantitativos em relação à distribuição normal. Mede o grau de afastamento de uma distribuição em relação a um eixo central (geralmente a média).

Quando a curva é simétrica, a média, a mediana e a moda coincidem, num mesmo ponto, havendo um perfeito equilíbrio na distribuição. Quando o equilíbrio não acontece, isto é, a média, a mediana e a moda recaem em pontos diferentes da distribuição esta será assimétrica; enviesada a direita ou esquerda. podendo-se caracterizar como curvas assimétricas à direita ou à esquerda. Quando a distribuição é assimétrica à esquerda ou assimetria negativa, a cauda da curva localiza-se à esquerda, desviando a média para este lado (Figura 112). Na assimetria positiva, ocorre o contrário, a cauda está localizada à direita e da mesma forma a média (89).

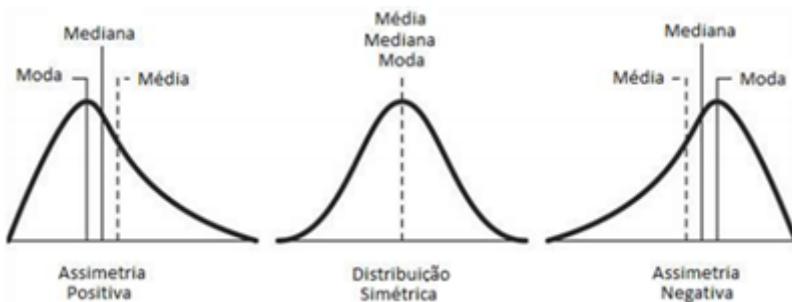


Figura 112: Assimetria

O R dispõe de diversas maneiras para o cálculo do *coeficiente de assimetria* (*g1*). O coeficiente de assimetria é um método numérico estatístico para medir a assimetria da distribuição ou conjunto de dados. Ele fala sobre a posição da maioria dos valores de dados na distribuição em torno do valor médio.

Quando a *assimetria* = 0, tem-se uma distribuição simétrica e a média, a mediana e a moda coincidem; quando a *assimetria* < 0, *mdia* < *mediana* < *moda*, a distribuição tem *assimetria negativa* e quando a *assimetria* > 0, *mdia* > *mediana* > *moda*, a distribuição tem *assimetria positiva*. A Tabela 7 sugere uma forma de interpretar o coeficiente de assimetria(90).

Tabela 7: Interpretação do *g1*

<i>g1</i>	assimetria
-1 a +1	leve
-1 a -2 e +1 a +2	moderada
-2 a -3 e +2 a +3	importante
< -3 ou > +3	grave

Como exemplo, de avaliação do coeficiente de assimetria, será usada a distribuição da variável altura, correspondente a altura em metros de 1368 parturientes da Maternidade do HGCS (*dadosMater.xlsx*), já mostrado anteriormente (Figura 45), repetido aqui com um boxplot sobreposto (Figura 113)

```
# Dados
mater <- readxl::read_excel("dadosMater.xlsx")
```

```

# Estruturação do layout do gráfico
layout(matrix(c(1,2), nrow = 2 , ncol =1, byrow = TRUE), heights = c(1, 8))

# Boxplot
par (mar=c (0, 4.3, 1.1, 2))
boxplot (mater$altura,
          horizontal = TRUE,
          ylim = c (1.4, 1.9),
          xaxt = "n",
          col = "lightblue",
          frame = FALSE)

#Histograma
par (mar=c (4, 4.3, 1.1, 2))
hist (mater$altura,
      breaks=15,
      col = "lightblue",
      border = "black",
      main = "",
      xlab = "Altura das Puérperas (m)",
      ylab = "Frequência",
      xlim = c(1.4,1.9),
      las = 1)
box(bty = "L")

```

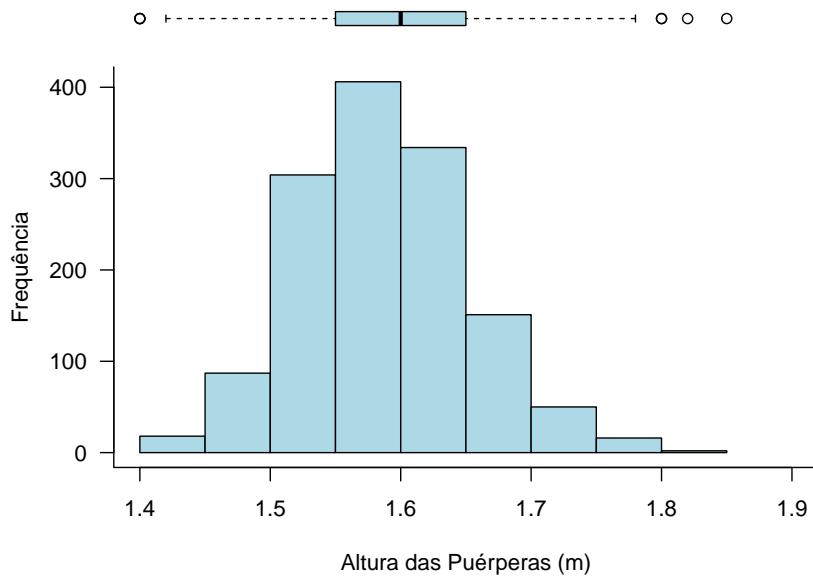


Figura 113: Posição de uma mulher com 1,50m comparando duas populações

```

# Restauração do padrão
par (mar = c(5, 4, 4, 2) + 0.1)

```

Observando o formato da distribuição no histograma e do boxplot, conclui-se que a variável `altura` tem uma assimetria positiva, provocada por alguns *outliers*, como uma mulher com altura de 1,85m. Para examinar

os `outliers`, pode-se verificar as estatísticas do boxplot, que entregam as estatísticas dos 5 números (min, P25, mediana, P75 e max), o total de observações, o limite inferior e superior do intervalo de confiança de 955 e os valores aípicos (`outliers`):

```
boxplot.stats(mater$altura)

## $stats
## [1] 1.42 1.55 1.60 1.65 1.78
##
## $n
## [1] 1368
##
## $conf
## [1] 1.595728 1.604272
##
## $out
## [1] 1.40 1.82 1.80 1.40 1.40 1.85 1.80
```

O formato formato dos dados se ajusta bem, como visto na Figura 113, ao modelo normal e os pressupostos deste modelo poderiam ser aplicados a estes dados.

O valor da assimetria (`skewness`) pode ser obtida com a função `skewness()` do pacote `moments` (91).

```
moments::skewness(mater$altura)
```

```
## [1] 0.1812196
```

O resultado da saída, confirmam a impressão visual, a variável altura tem uma distribuição praticamente simétrica. Esta conclusão também pode ser feita, analisando as medidas resumidoras dessa variável:

```
media <- mean(mater$altura, na.rm = TRUE)
round(media, 2)
```

```
## [1] 1.6
```

```
dp <- sd(mater$altura, na.rm = TRUE)
round(dp, 3)
```

```
## [1] 0.065
```

```
mediana <- median (mater$altura, na.rm = TRUE)
mediana
```

```
## [1] 1.6
```

```
coefVar <- dp/media
round(coefVar, 3)
```

```
## [1] 0.041
```

Os resultados mostram um desvio padrão pequeno, média igual à mediana e um coeficiente de variação igual a 0.041, muito próximo de 0, características consideradas pertencentes a uma amostra que provavelmente se ajusta à distribuição normal.

8.2 Curtose

É o grau de achatamento de uma distribuição, em relação a distribuição normal. A curtose indica como o pico e as caudas de uma distribuição diferem da distribuição normal. A assimetria mede essencialmente a simetria da distribuição, enquanto a curtose determina o peso das caudas da distribuição. Portanto, é uma medida dos tamanhos combinados das duas caudas; mede a quantidade de probabilidade nas caudas.

Uma curtose em excesso é uma medida que compara a curtose de uma distribuição com a curtose de uma distribuição normal. A curtose de uma distribuição normal é igual a 3. Portanto, o excesso de curtose é determinado subtraindo 3 da curtose:

$$\text{Excesso de curtose} = \text{curtose} - 3$$

Os dados que seguem uma *distribuição mesocúrtica* mostram um excesso de curtose de zero ou próximo de zero. Isso significa que se os dados seguem uma distribuição normal, eles seguem uma distribuição mesocúrtica. A *distribuição leptocúrtica* mostra caudas pesadas em ambos os lados, indicando grandes valores discrepantes. Uma *distribuição leptocúrtica* manifesta uma curtose excessiva positiva. Uma *distribuição platicúrtica* mostra uma curtose excessiva negativa, revela uma distribuição com cauda plana (Figura 114).

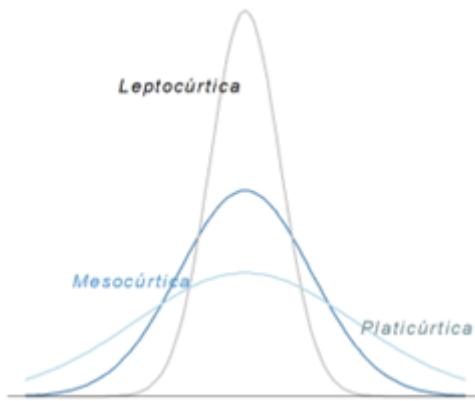


Figura 114: Assimetria

A função `kurtosis()`, também do pacote `moments`, pode ser usada para o cálculo do coeficiente de curtose e do resultado deve-se subtrair o valor 3.

```
moments::kurtosis(mater$altura)
```

```
## [1] 3.124257
```

Se o coeficiente de curtose é maior do que 3, há um excesso de curtose e a distribuição dos dados é leptocúrtica com um pico mais acentuado no gráfico. O resultado do exemplo aponta para uma distribuição leptocúrtica, pois existe um pequeno excesso de curtose (`moments::kurtosis(mater$altura) - 3`). Quando o coeficiente de curtose é menor do que 3, a distribuição é platicúrtica e a curva fica mais achatada. Quando o coeficiente de curtose é igual a 3, ou próximo de 3, a distribuição é mesocúrtica, como a distribuição normal.

8.3 Testando o raciocínio

1. Criar um conjunto de dados com distribuição normal com média 0 e desvio padrão 1 e n = 100000

```
# Criação da variável n100000
n100000 <- rnorm(100000, mean = 0, sd = 1)
```

2. Construa um histograma (Figura 115) com curva normal sobreposta:

```
ggplot() +
  geom_histogram(aes(x = n100000,
                     y = ..density..),
                 bins = 20,
                 fill='tomato',
                 col=alpha('red',0.2)) +
  geom_function(fun=dnorm,
                args=list(mean=0,sd=1),
                col='dodgerblue4',
                lwd=1,
                lty=2) +
  labs(x='X',
       y='Densidade de probabilidade',
       caption = "PFOF")+
  theme_bw()
```

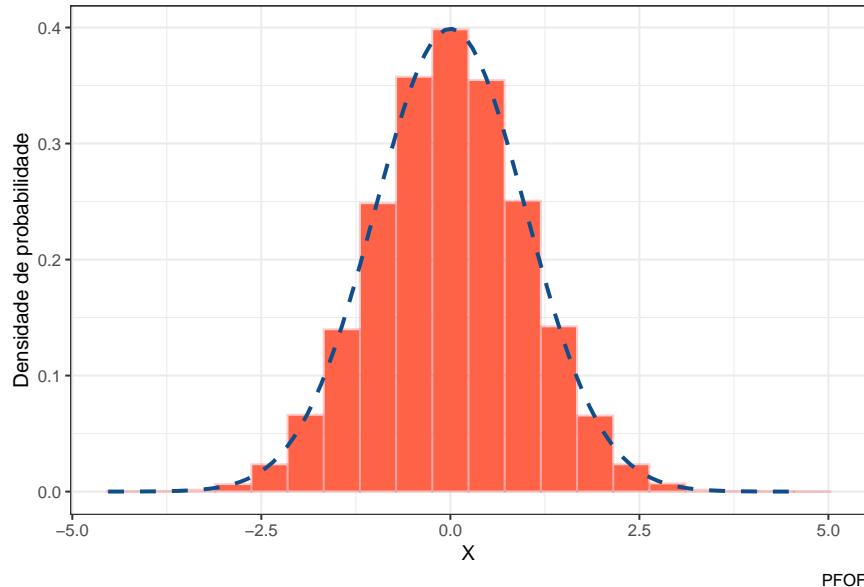


Figura 115: Histograma com curva normal

3. Observe a skewness e a kurtosis

```
print(moments::skewness(n100000))
```

```
## [1] 0.0007816012
```

```
print(moments::kurtosis(n100000))
```

```
## [1] 3.032317
```

Como era de se esperar, usando a `rnorm()`, a distribuição é um exemplo de distribuição normal, $skewness \approx 0$ e $kurtosis \approx 3$. Observe que a cada vez que os comandos forem executados, os resultados serão discretamente diferentes. Para evitar isso, deve-se usar `set.seed()`. Faça o teste!

9 Distribuições Amostrais

9.1 Distribuições populacional e amostral

Estatísticas da amostra, como a média, a mediana, a moda e o desvio padrão, são medidas numéricas de resumo calculadas para dados de uma amostra. Por outro lado, as mesmas medidas numéricas de resumo calculadas para dados populacionais são chamadas de *parâmetros populacionais*.

Um parâmetro populacional é sempre uma constante, enquanto uma estatística de amostra é sempre uma variável aleatória. Como cada variável aleatória deve possuir uma distribuição de probabilidade, cada estatística de amostra possui uma distribuição de probabilidade. A distribuição de probabilidade de uma estatística de amostra é mais comumente chamada de *distribuição amostral*. Os conceitos abordados neste capítulo são a base da *estatística inferencial*.

9.1.1 Distribuição populacional

A distribuição populacional é a distribuição de probabilidade derivada das informações sobre todos os elementos de uma população.

O conjunto de dados de 1368 observações de puérperas e recém-nascidos da Maternidade-escola do Hospital Geral de Caxias do Sul, RS, será, para fins didáticos, considerado a *população*. O gráfico, abaixo, mostra a distribuição da altura das puérperas dessa ‘população’ (Figura 116).

```
mater <- readxl::read_excel("dadosMater.xlsx")
media = mean(mater$altura, na.rm = TRUE)
dp = sd(mater$altura, na.rm = TRUE)
```

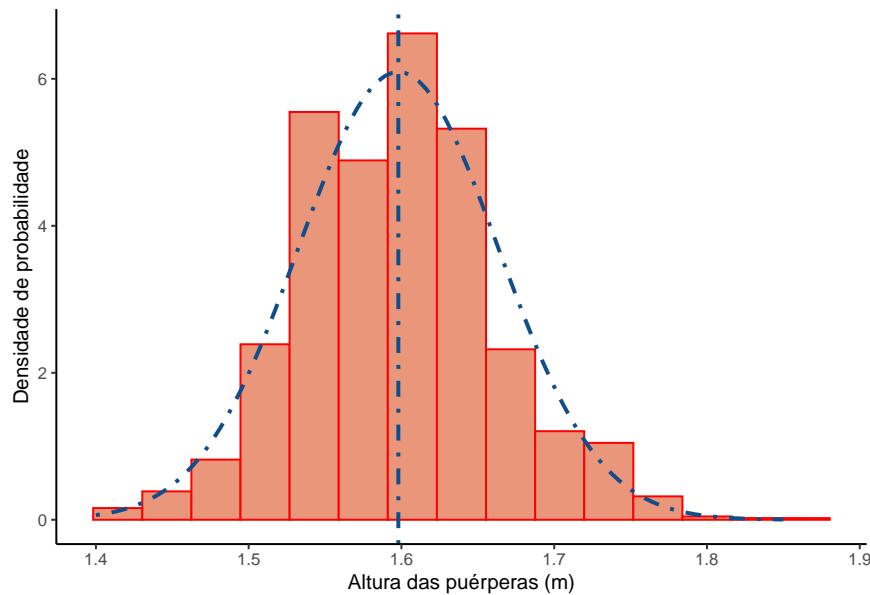


Figura 116: Histograma da altura de 1368 puérperas

Os valores da média e do desvio padrão calculados para essa população fornecem os valores dos parâmetros populacionais μ e σ . Esses valores são $\mu = 1.598\text{m}$ e $\sigma = 0.065\text{m}$.

9.1.2 Distribuição amostral

Conforme mencionado no início deste capítulo, o valor de um parâmetro da população é sempre constante. Por exemplo, para qualquer conjunto de dados populacionais, há apenas um valor para a média populacional, μ .

No entanto, não se pode dizer o mesmo sobre a média amostral. Amostras diferentes do mesmo tamanho, retiradas da mesma população, produzem valores diferentes da média amostral, \bar{x} . O valor da média amostral, para qualquer amostra, dependerá dos elementos incluídos nessa amostra. Em decorrência, a média amostral é uma variável aleatória. Portanto, como outras variáveis aleatórias, a média amostral possui uma distribuição de probabilidade, que é mais comumente chamada de *distribuição amostral da média*.

Outras estatísticas de amostra, como mediana, moda e desvio padrão, também possuem distribuições amostrais. Em geral, a distribuição de probabilidades de uma amostra é denominada de distribuição amostral.

Voltando à variável altura das puérperas da Maternidade do HGCS, convencionada a priori como a população de interesse. Isso raramente acontece na vida real. Reunir informação sobre uma população inteira costuma ser muito custoso ou impossível. Por essa razão, a prática é selecionar apenas uma amostra da população e a usar para compreender as suas características.

Usando a função `slice_sample()` do pacote `dplyr`, será extraída uma amostra de $n = 30$ da população e calculada a média e o desvio padrão:

```
library(dplyr)
mater <- readxl::read_excel("dadosMater.xlsx") %>%
  dplyr::select(altura)

amostra1 <- mater %>% dplyr::slice_sample(n = 30)

media1 <- mean(amostra1$altura, na.rm =TRUE)
dp1 <- sd(amostra1$altura, na.rm =TRUE)
print(c(media1, dp1))

## [1] 1.59600000 0.07636573
```

A `amostra1` tem média igual a \$1.596 e desvio padrão igual a 0.076. Se este processo for repetido várias vezes, ninguém ficará surpreso se, a cada amostra aleatória, a média amostral for diferente das anteriores, gerando médias e desvios padrão diferentes.

```
amostra2 <- mater %>% dplyr::slice_sample(n = 30)

media2 <- mean(amostra2$altura, na.rm =TRUE)
dp2 <- sd(amostra2$altura, na.rm =TRUE)
print(c(media2, dp2))

## [1] 1.58266667 0.07362221
```

À medida que o número de amostras possíveis forem aumentando, elas constituem uma distribuição cuja média, média das médias, $\bar{x}_{\bar{x}}$, é igual a média populacional, μ . Essa distribuição, no caso da média, recebe o nome de *distribuição amostral das médias*.

Agora, para exemplificar este conceito, serão geradas 5000 amostras e calculada a média de cada uma das amostras de $n = 30$ que constituirão a distribuição, mostrada no gráfico da Figura 117.

```

# extraindo 5000 amostras
amostras5000 <- rep (0, 5000)
for (i in 1:5000) {
  amostra <- mater %>% dplyr::slice_sample (n = 30)
  amostras5000 [i] <- mean(amostra$altura)
}
# Media e desvio padrão das 5000 amostras
round (mean (amostras5000), digits = 3)

## [1] 1.598

round (sd (amostras5000), digits = 3)

## [1] 0.012

```

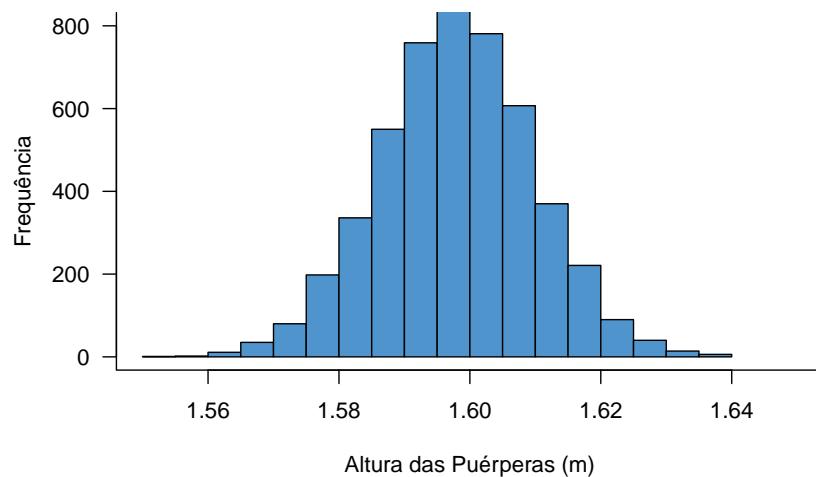


Figura 117: Distribuição amostral das médias de 5000 amostras de $n = 30$

Se a média, $\bar{x}_{\bar{x}}$, dessas 5000 amostras de $n = 30$, for comparada com a média populacional, μ , observa-se que até 3 dígitos decimais não há uma diferença. Entretanto, o desvio padrão é bem menor (0.012) que o da população (0.065).

9.2 Erros amostrais e não amostrais

Normalmente, amostras diferentes selecionadas da mesma população darão resultados diferentes porque contêm elementos diferentes. Isso é evidente nas medias das amostra1 e amostra2, 1.596m e 1.583m, respectivamente, comparadas com a média da população igual a 1.598m .

```

erro1 <- abs(mean(amostra1$altura, na.rm =TRUE) - mean(mater$altura, na.rm =TRUE))
round(erro1, 3)

```

```

## [1] 0.002

erro2 <- abs(mean(amostra2$altura, na.rm =TRUE) - mean(mater$altura, na.rm =TRUE))
round(erro2, 3)

## [1] 0.015

```

Se outras amostras forem extraídas, o resultado obtido de qualquer amostra geralmente será diferente do resultado obtido da população correspondente. A diferença (erro) entre o valor de uma estatística amostral obtida de uma amostra e o valor do parâmetro populacional correspondente, é chamada de *erro amostral*. Observe que essa diferença representa o erro amostral apenas se a amostra for aleatória e não houver nenhum erro não amostral. Caso contrário, apenas uma parte dessa diferença será devido ao erro de amostragem.

$$\mu = \bar{x}_i + \text{erro amostral}$$

É importante lembrar que o erro amostral ocorre devido ao acaso. Os erros que ocorrem por outros motivos, como erros cometidos durante a coleta, registro e tabulação dos dados, são chamados de *erros não amostrais*. Esses erros ocorrem, em geral, por causa de erros humanos e não por acaso.

9.3 Média e desvio padrão da média

A média e o desvio padrão calculados para a distribuição amostral da média são chamados de *média* ($\mu_{\bar{x}}$) e *desvio padrão* ($\sigma_{\bar{x}}$) da *média*. Na verdade, a média e o desvio padrão da média são, respectivamente, a média e o desvio padrão das médias de todas as amostras do mesmo tamanho selecionadas de uma população. O desvio padrão da média é, comumente, chamado de *erro padrão da média* ($\sigma_{\bar{x}}$).

A média amostral, \bar{x} , é chamada de *estimador* da média da população, μ . Quando o valor esperado (ou média) de uma estatística amostral é igual ao valor do parâmetro populacional correspondente, essa estatística amostral é considerada um estimador não enviesado, consistente.

Para a média amostral \bar{x} , $\mu_{\bar{x}} = \mu$. Logo, \bar{x} , é um estimador imparcial de μ . Esta é uma propriedade muito importante que um estimador deve possuir. No entanto, o desvio padrão da média, $\sigma_{\bar{x}}$, não é igual ao desvio padrão, σ , da distribuição populacional (a menos que $n = 1$). O desvio padrão da média amostral é igual ao desvio padrão da população dividido pela raiz quadrada do tamanho amostral:

$$\sigma_{\bar{x}} = \frac{\sigma}{\sqrt{n}}$$

A dispersão da distribuição amostral da média é menor do que a dispersão da distribuição populacional correspondente, como mostrado acima. Em outras palavras, $\sigma_{\bar{x}} < \sigma$. Isso é visível na fórmula do $\sigma_{\bar{x}}$. Quando n é maior que 1, o que geralmente é verdadeiro, o denominador em $\frac{\sigma}{\sqrt{n}}$ é maior que 1. Desta forma, $\sigma_{\bar{x}}$ é menor que σ . O desvio padrão da distribuição amostral da média diminui à medida que o tamanho amostral aumenta.

Sempre que o n for grande, em geral > 30 (92), pode ser assumido que a distribuição será uma curva normal e que o desvio padrão da amostra (s) é um estimador não enviesado do desvio padrão populacional (σ). Então, o erro padrão da média ($\sigma_{\bar{x}}$) pode ser estimado pelo $EP_{\bar{x}}$:

$$EP_{\bar{x}} = \frac{s}{\sqrt{n}}$$

9.4 Teorema do Limite Central

Na maioria das vezes, a população da qual as amostras são extraídas não é normalmente distribuída. Em tais casos, a forma da distribuição amostral de X é inferida de um teorema muito importante chamado *teorema do limite central*. De acordo com este teorema para um grande tamanho de amostra (> 30), a distribuição amostral da média é aproximadamente normal, independentemente da forma da distribuição da população (92). Esta aproximação tornar-se-á mais acurada à medida que aumenta o tamanho amostral:

- a média da distribuição amostral, $\mu_{\bar{x}}$, é igual a média populacional, μ ;
- desvio padrão da distribuição amostral, $\sigma_{\bar{x}}$, é igual a $\frac{\sigma}{\sqrt{n}}$;
- o erro padrão da média, $\sigma_{\bar{x}}$, é sempre menor que o desvio padrão populacional, σ (Figura 118).

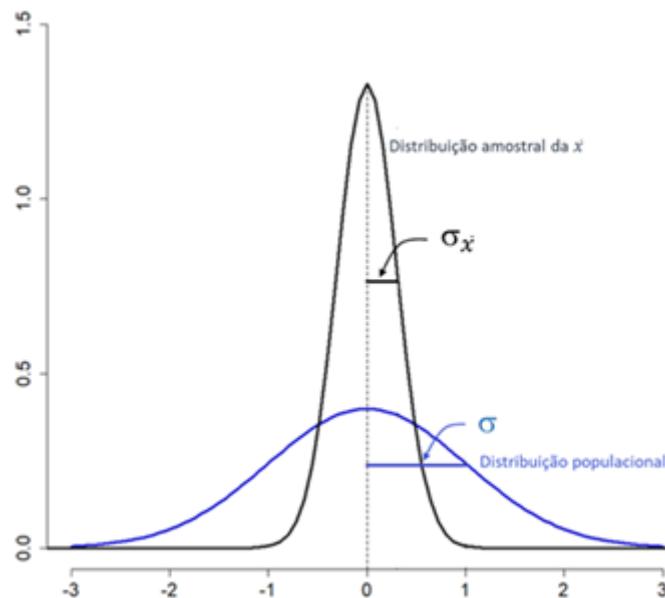


Figura 118: Erro padrão versus desvio padrão.

Se for tomado como exemplo a variável `renda`, do conjunto de dados `dadosMater.xlsx`, que representa a renda familiar em salários mínimos (SM). Como foi feito anteriormente, suponha que essa variável seja a população de estudo. Ela tem as seguintes medidas resumidoras e de assimetria:

```
mater <- readxl::read_excel("dadosMater.xlsx")

media.sm <- mean (mater$renda, na.rm = TRUE)
media.sm

## [1] 2.224949

dp.sm <- sd(mater$renda, na.rm = TRUE)
dp.sm

## [1] 1.226359
```

```
mediana.sm <- median(mater$renda, na.rm = TRUE)
mediana.sm
```

```
## [1] 1.92
```

```
print(moments::skewness(mater$renda))
```

```
## [1] 2.22336
```

```
print(moments::kurtosis(mater$renda))
```

```
## [1] 11.22392
```

O desvio padrão é grande em relação à média, com um coeficiente de variação de 55.1185153% e uma mediana < média. Estas métricas junto com os coeficientes de assimetria e curtose apontam para a assimetria positiva da variável `renda familiar`. O gráfico da Figura 119 confirma esta afirmação:

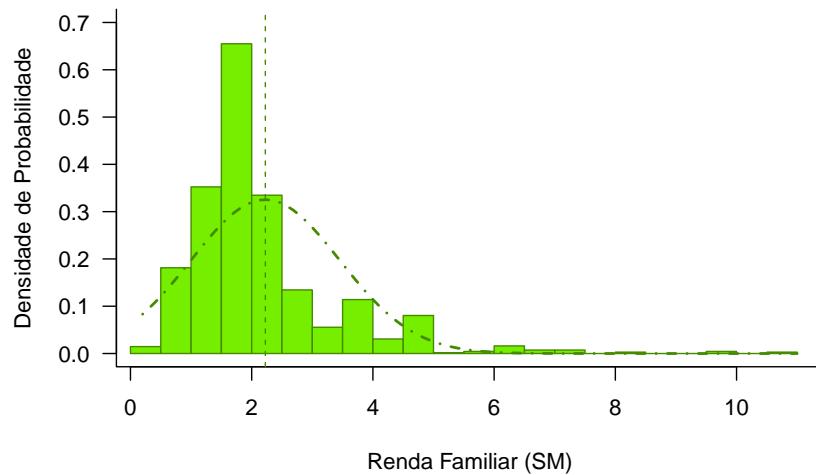


Figura 119: Distribuição assimétrica positiva

Os valores da média e do desvio padrão calculados para a distribuição de probabilidade dessa população fornecem os valores dos parâmetros populacionais μ e σ . Esses valores são $\mu = 2.225\text{m}$ e $\sigma = 1.226\text{m}$.

Se extraímos múltiplas amostras dessa população, observa-se a modificação do formato da distribuição à medida que aumenta o tamanho amostral, se aproximando progressivamente do modelo normal, com um número grande de amostras.

```
# extraindo 1000 amostras
amostras1000 <- rep(0, 1000)
for (i in 1:1000) {
  amostra.sm <- sample (mater$renda, 30)
  amostras1000 [i] <- mean(amostra.sm)
```

```

}

# Media e desvio padrão das 1000 amostras
round (mean (amostras1000), digits = 3)

```

```
## [1] 2.221
```

```
round (sd (amostras1000), digits = 3)
```

```
## [1] 0.22
```

```
round (median(amostras1000), digits = 3)
```

```
## [1] 2.218
```

```
print(moments::skewness(amostras1000))
```

```
## [1] 0.2404874
```

```
print(moments::kurtosis(amostras1000))
```

```
## [1] 2.857316
```

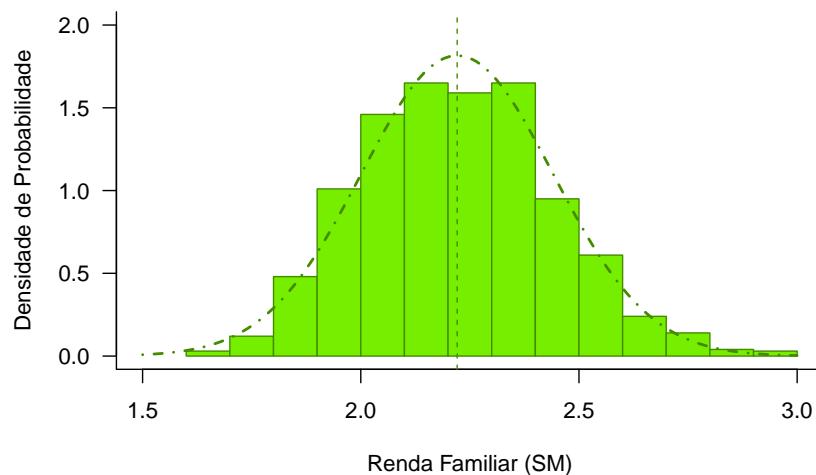


Figura 120: Distribuição praticamente normal

Ou seja, extraíndo-se 1000 amostras de $n = 30$ e calculando as mesmas métricas anteriores, verifica-se que, agora, a variável está com distribuição praticamente normal (Figura 120).

9.5 Proporções populacional e amostral

O conceito de proporção é o mesmo que o conceito de frequência relativa e o conceito de probabilidade de sucesso em um experimento binomial, discutidos anteriormente, na distribuição binomial.

A frequência relativa de uma categoria ou classe dá a proporção da amostra ou população que pertence a essa categoria ou classe. Da mesma forma, a probabilidade de sucesso em um experimento binomial representa a proporção da amostra ou população que possui uma determinada característica.

A proporção populacional, representada por p , é obtida considerando a razão entre o número de elementos em uma população com uma característica específica e o número total de elementos na população. A proporção amostral, denotada por \hat{p} (pronuncia-se p-chapéu), fornece uma proporção semelhante para uma amostra.

$$p = \frac{X}{N} \quad e \quad \hat{p} = \frac{x}{n}$$

onde,

- N = número total de elementos em uma população
- n = número total de elementos em uma amostra
- X = número de elementos na população que possui determinada característica
- x = número de elementos na amostra que possui determinada característica

Como no caso da média, a diferença entre a proporção amostral e a proporção populacional correspondente, determina o *erro amostral*, assumindo que a amostra é aleatória e nenhum erro não amostral foi cometido. Ou seja,

$$\text{erro amostral} = \hat{p} - p$$

No conjunto de dados `dadosMater.xlsx`, pserá verificado a proporção de fumantes com:

```
mater <- readxl::read_excel("dadosMater.xlsx")

mater$fumo <- factor (mater$fumo,
                       levels = c (1,2),
                       label = c ("sim", "não"))

fumo <- with(mater, table(fumo))
fr.fumo <- prop.table(fumo)
```

Assim, a proporção de gestantes fumantes foi de 0.22.

Considerando que este resultado fosse desconhecido e que as mulheres da maternidade do HGCS fosse a população, para verificar a proporção de mulheres fumantes, se extrairá uma amostra de $n = 100$ (Tabela 8).

```
amostra.fumo <- mater %>% dplyr::slice_sample(n = 100)

tabagismo <- with(amostra.fumo, table(fumo))
fr <- prop.table(tabagismo)
fp <- fr*100

tab.fumo <- cbind(n = tabagismo,
                   fr = round(fr, 2),
```

```

fp = round(fp, 2))

tab.fumo <- as.data.frame(tab.fumo)

```

Tabela 8: Proporção de Tabagismo nas gestantes

	n	Freq. Relativa	Freq. Percentual
sim	23	0.23	23
não	77	0.77	77

A proporção de uma amostra é uma variável aleatória: varia de amostra para amostra de uma forma que não pode ser prevista com certeza. Foi visto que esta variável aleatória é escrita como \hat{p} . E que tem uma média $\mu_{\hat{p}}$ e um desvio padrão $\sigma_{\hat{p}}$.

Suponha que amostras aleatórias de tamanho n sejam retiradas de uma população na qual a proporção com uma característica de interesse seja p . A média e o desvio padrão da proporção amostral \hat{p} satisfazem

$$\mu_{\hat{p}} = p \quad e \quad \sigma_{\hat{p}} = \sqrt{\frac{pq}{n}}$$

O Teorema do Limite Central também se aplica aqui. No entanto, a condição de que a amostra seja grande é um pouco mais complicada do que apenas ter um tamanho de pelo menos 30.

9.5.1 Distribuição amostral da proporção amostral

Para amostras grandes, a proporção amostral é aproximadamente normalmente distribuída, com média $\mu_{\hat{p}} = p$ e desvio padrão $\sigma_{\hat{p}} = \sqrt{\frac{pq}{n}}$.

Uma amostra é grande se o intervalo $[p - 3\sigma_{\hat{p}}, p + 3\sigma_{\hat{p}}]$ estiver totalmente dentro do intervalo $[0,1]$.

Na prática, p não é conhecido, portanto, $\sigma_{\hat{p}}$ também não é. Nesse caso, para verificar se a amostra é suficientemente grande, substitui-se o valor de p pelo valor conhecido de \hat{p} . Isso significa verificar se o intervalo encontra-se totalmente dentro do intervalo $[0,1]$, usando:

$$\left(\hat{p} - 3 \times \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}, \quad \hat{p} + 3 \times \sqrt{\frac{\hat{p}(1-\hat{p})}{n}} \right)$$

Transportando os dados da amostra de gestantes, para a fórmula e usando o R para o cálculo, tem-se:

```

p.chapeu <- tab.fumo[1,2]
n <- tab.fumo[1,1] + tab.fumo[2,1]

li <- p.chapeu - 3*sqrt((p.chapeu*(1-p.chapeu))/n)
li

## [1] 0.1037502

ls <- p.chapeu + 3*sqrt((p.chapeu*(1-p.chapeu))/n)
ls

## [1] 0.3562498

```

Dessa forma, tem-se que a amostra de $n = 100$ é aceitável para uma população onde $p = 0.22$

10 Estimação

10.1 Introdução

A estatística inferencial é a parte da estatística que usa os resultados da amostra para tomar decisões e tirar conclusões sobre a população de onde a amostra foi retirada. A estimação e o teste de hipóteses, tomados em conjunto, constituem a *inferência estatística*.

Estimação é um procedimento pelo qual um valor ou valores numéricos são atribuídos a um parâmetro populacional com base nas informações de uma amostra. Na estatística inferencial, μ é chamada, como anteriormente visto, de média populacional e p é chamada de proporção populacional. Existem muitos outros parâmetros populacionais, como mediana, moda, variância e desvio padrão.

Se houvesse possibilidade de realizar um *censo* (pesquisa incluindo toda a população de interesse), não haveria necessidade dos procedimentos de estimação. Seria equivalente ao que ocorre em uma eleição, basta contar os votos, para declarar os vencedores da eleição. No entanto, em saúde, realizar censo é um procedimento caro, demorado ou virtualmente impossível. Portanto, geralmente é utilizada uma amostra da população e calculada o valor das estatísticas da amostra apropriada. Baseado nessas estatísticas, é atribuído valores ao parâmetro.

Os valores atribuídos a um parâmetro populacional com base no valor de uma estatística amostral são chamados de *estimativa* do parâmetro populacional. A estatística da amostra usada para estimar um parâmetro da população é chamada de *estimador*. Assim, a média da amostra, \bar{x} , é um estimador da média da população, μ ; e a proporção da amostra, \hat{p} , é um estimador da proporção da população, p .

10.2 Pacotes necessários para este capítulo

```
pacman::p_load(readxl,  
                 dplyr,  
                 knitr,  
                 kableExtra,  
                 ggplot2,  
                 Rmisc,  
                 DescTools)
```

10.3 Estimativa Pontual e Intervalo de Confiança

Se for selecionada uma amostra e calculado o valor da estatística amostral, esse valor fornece a estimativa do parâmetro populacional correspondente.

Voltando a considerar, para fins didáticos, o dataset `dadosMater.xlsx` como se fosse uma população, serão calculadas as medidas sumarizadora básicas da variável `pesoRN`:

```
mater <- read_excel("dadosMater.xlsx") %>%  
  filter(ig>=37 & ig<42)  
  
# Média populacional  
mu <- round(mean(mater$pesoRN, na.rm = TRUE))  
mu  
  
## [1] 3216
```

```
# Desvio padrão populacional
dp_pop <- round(sd(mater$pesoRN, na.rm = TRUE))
dp_pop
```

```
## [1] 462
```

Agora, será extraída da população uma amostra de $n = 30$ ¹¹ e calculado os mesmas medidas resumidoras;

```
mater30 <- mater %>% slice_sample(n = 30)

# Média amostral
media30 <- round(mean(mater30$pesoRN, na.rm = TRUE))
media30
```

```
## [1] 3139
```

```
# Desvio padrão amostral
dp30 <- round(sd(mater30$pesoRN, na.rm = TRUE))
dp30
```

```
## [1] 567
```

O valor de 3139g é a média amostral, \bar{x} , usada como um estimativa da μ , é denominada de *estimativa pontual*. Como já mencionado anteriormente, espera-se que cada amostra selecionada produza um valor diferente da estatística amostral.

Assim, o valor atribuído a uma média populacional, μ , com base em uma estimativa pontual depende de qual das amostras está sendo usada. Consequentemente, a estimativa pontual atribui um valor a μ que quase sempre difere da mesma.

Para melhorar a precisão, usa-se uma estimativa de intervalo. Em vez de atribuir um único valor para o parâmetro populacional, é construído um intervalo, acrescentando ou subtraindo um valor, chamado de *margem de erro*, à estimativa pontual.

Este procedimento é conhecido como *estimação por intervalo* e o intervalo construído, estabelecendo um limite inferior e um limite superior em torno da estimativa amostral, é denominado de *intervalo de confiança*. Desta forma, é possível afirmar que o intervalo de confiança, provavelmente, contém o parâmetro populacional correspondente (Figura 121).

¹¹É importante lembrar que toda vez que for extraída uma nova amostra de tamanho $n = 30$, o resultado será um conjunto de números diferentes e, em consequência, a média será diferente.

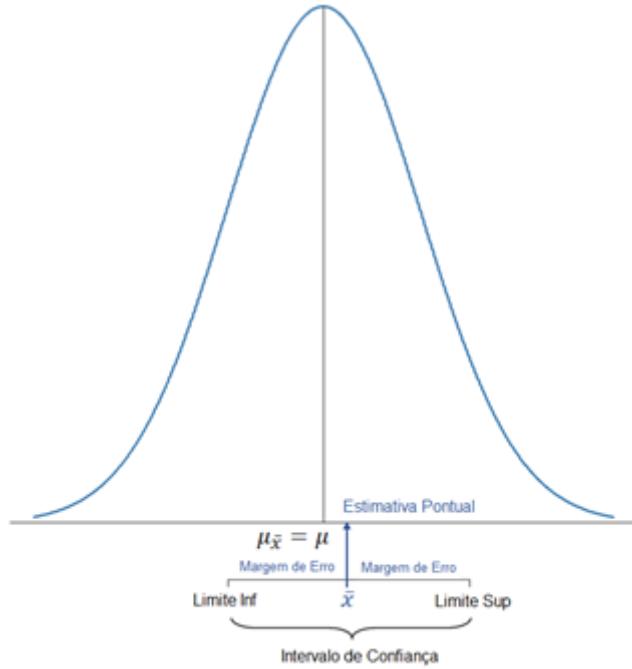


Figura 121: Intervalo de Confiança.

A construção do intervalo de confiança depende da obtenção da margem de erro. Este processo necessita de dois fatores:

- do desvio padrão da distribuição amostral, $\sigma_{\bar{x}} = \frac{\sigma}{\sqrt{n}}$, que em decorrência do Teorema do Limite Central, pode ser escrito $EP_{\bar{x}} = \frac{s}{\sqrt{n}}$;
- do nível de confiança atribuído ao intervalo.

Primeiro, quanto maior for o desvio padrão de \bar{x} , maior será a margem de erro subtraída e adicionada à estimativa pontual. Consequentemente, o intervalo de confiança se modifica de acordo com a margem de erro. Quanto maior a margem de erro mais amplo o intervalo de confiança.

Em segundo lugar, a quantidade subtraída e adicionada à estimativa se modifica de acordo o *nível de confiança*. Para ter uma maior confiança, deve-se aumentar a margem de erro, de acordo com a probabilidade declarada. Quanto maior o nível de confiança (NC), maior a probabilidade. O nível de confiança é mostrado como $(1 - \alpha) \times 100\%$, onde α é o *nível de significância*. Tradicionalmente, o valor de α é igual a 0,05, mas qualquer outro valor pode ser usado.

10.4 Estimação da média populacional: σ conhecido

A *margem de erro* para a estimativa da média populacional, μ , quando se conhece o desvio padrão populacional, σ , e $n \geq 30$ ou, mesmo que $n < 30$, mas a população de onde amostra foi selecionada tem distribuição normal, é a quantidade que é subtraída ou adicionada ao valor da média da amostra, \bar{x} , para obter o intervalo de confiança para μ . Desta forma, a margem de erro é igual a:

$$\text{margem de erro (me)} = z_{(1-\frac{\alpha}{2})} \times \sigma_{\bar{x}}$$

Ou,

$$me = z_{(1-\frac{\alpha}{2})} \times \frac{\sigma}{\sqrt{n}}$$

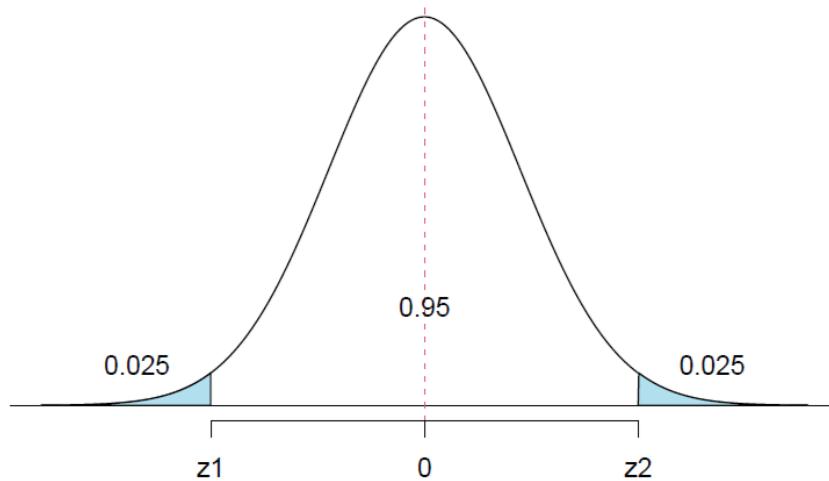
Logo, o intervalo de confiança para a média populacional, μ , para um nível de confiança $(1 - \alpha)100\%$, é igual a:

$$IC_{(1-\alpha)}(\mu) \rightarrow \bar{x} \pm me$$

Se objetivo é construir um intervalo de confiança de 95%, a última equação passa a ser:

$$IC_{(1-\alpha)}(\mu) \rightarrow \bar{x} \pm z_{(0,975)} \times me$$

Onde z é o valor crítico para o nível de confiança escolhido, obtido da tabela de distribuição normal padrão, e me é a margem de erro ($z_{0,975} \times \text{erro padrao}$). Um intervalo de confiança de 95% significa que a área total sob a curva normal entre dois pontos em torno da média populacional, μ , é igual a 95%, ou 0,95. A área das caudas é α , ou seja, cada cauda é igual a $\frac{\alpha}{2}$.



Para encontrar o valor de z para um nível de confiança de 95%, primeiro encontram-se as áreas à esquerda desses dois pontos, z_1 e z_2 . Esses dois valores de z serão iguais, mas com sinais opostos. A área total sob a curva é igual a 1. A área entre z_1 e z_2 é igual a $1 - \alpha = 0,95$.

A área a esquerda de z_1 é igual a 0,025 e a área a esquerda de z_2 é igual a $1 - 0,025 = 0,975$. No R, o valor z_1 e z_2 podem facilmente ser obtidos com a função `qnorm()`:

```
qnorm(0.025)
```

```
## [1] -1.959964
```

```
qnorm(0.975)
```

```
## [1] 1.959964
```

Dessa maneira, para uma confiança de 95%, é usado um $z = 1.96$, onde:

$$p(-1,96 \leq z \leq 1,96) = 0,95$$

Desta forma, usando a média da amostra de 30 recém-nascidos a termo igual a 3216g e o desvio padrão populacional é igual a 462g. Então, pode-se escrever que o intervalo de confiança de 95% para o peso do recém-nascido a termo da Maternidade do HGCS é:

$$\begin{aligned} IC_{95\%}(\mu) &\rightarrow \bar{x} \pm (1.96 \times \sigma_{\bar{x}}) \\ IC_{95\%}(\mu) &\rightarrow \bar{x} \pm (1.96 \times \frac{\sigma}{\sqrt{n}}) \end{aligned}$$

```
n = length(mater30$pesoRN)
me <- 1.96 * dp_pop/sqrt(n)
round(me)
```

```
## [1] 165
```

```
round (media30)
```

```
## [1] 3139
```

```
lim_inf <- media30 - me
lim_sup <- media30 + me
ic95 <- c(lim_inf, lim_sup)
round(ic95)
```

```
## [1] 2974 3304
```

O R não tem um comando específico para encontrar os intervalos de confiança para a média dos dados normais quando o desvio padrão da população é conhecido. Então, foi criada uma função para calcular o intervalo de confiança, conhecendo-se o desvio padrão da população¹². Esta função necessita dos seguintes argumentos:

- $x \rightarrow$ conjunto de números da amostra
- $s \rightarrow$ desvio padrão populacional
- $nc \rightarrow$ nível de confiança. Padrão: nc = 0.95

```
IC_z <- function (x, s, nc = 0.95)
{
  `>%` <- magrittr::`>%
  n <- length(x)
  me <- abs(qnorm((1-nc)/2))* dp_pop/sqrt(n)
  df_out <- data.frame( tamanho_amostral = n,
                        media_amostral = mean(x),
                        dp_amostral = sd(x),
                        margem_erro = me,
                        'IC limite inferior'=mean(x) - me,
                        'IC limite superior'=mean(x) + me) %>%
  tidyr::pivot_longer(names_to = "Medidas", values_to ="valores", 1:6 )
  return(df_out)
}
```

¹²Situação rara! A regra é este ser desconhecido.

```
IC_z(x = mater30$pesoRN, s = dp_pop, )
```

```
## # A tibble: 6 x 2
##   Medidas      valores
##   <chr>        <dbl>
## 1 tamanho_amostral     30
## 2 media_amostral    3139.
## 3 dp_amostral       567.
## 4 margem_erro        165.
## 5 IC.limite.inferior 2974.
## 6 IC.limite.superior 3304.
```

Portanto, tem-se uma confiança de 95% de que a média do peso dos recém-nascidos a termo da Maternidade do HGCS está entre 2974 e 3304g.

Estes limites também podem ser estendidos para outros níveis de confiança. Por exemplo, para um nível de confiança de 99% $(1 - \frac{\alpha}{2}) = (1 - \frac{0.01}{2}) = 0,995$, o z crítico é igual:

```
round(qnorm(0.995), 2)
```

```
## [1] 2.58
```

Substituindo na fórmula, usando o R:

```
l_inf <- mu - (2.58 * dp_pop/sqrt(n))
l_sup <- mu + (2.58 * dp_pop/sqrt(n))
ic99 <- c(l_inf, l_sup)
round(ic99)
```

```
## [1] 2998 3434
```

Agora, com uma confiança de 99% o intervalo fica entre 2998 e 3434g.

Observando o IC95% e o IC99%, verifica-se que a amplitude do intervalo aumentou com o crescimento da confiança de 95% para 99%, porque houve um aumento na margem de erro.

10.4.1 Interpretação do intervalo de confiança

Se fossem extraídas todas as possíveis amostras de tamanho 30 da população de recém-nascidos a termo e construído para cada uma delas um intervalo de confiança de 95% em torno de cada média amostral, espera-se que 95% desses intervalos incluirão a média populacional e 5% não incluirão.

O IC95% informa sobre a *precisão* com que a média amostral estima a média populacional desconhecida ¹³.

Na Figura ??, são mostradas 20 amostras diferentes de tamanho $n = 30$, retiradas da mesma população de recém-nascidos a termo da Maternidade do HGCS. Junto aparecem os intervalos de confiança de 95% construídos em torno dessas amostras. Observa-se que apenas uma amostra (em vermelho) não inclui a média populacional (linha tracejada vertical em azul). Pode-se afirmar com 95% de confiança que se forem extraídas muitas amostras do mesmo tamanho de uma população e construído intervalos de confiança de 95% em torno das médias dessas amostras, 95% desses intervalos de confiança incluirão a média populacional.

¹³Anteriormente, mostrou-se a media populacional por uma questão didática. A regra é não se conhecer a média populacional, razão da importância do intervalo de confiança

ICs de sucesso: 95 %

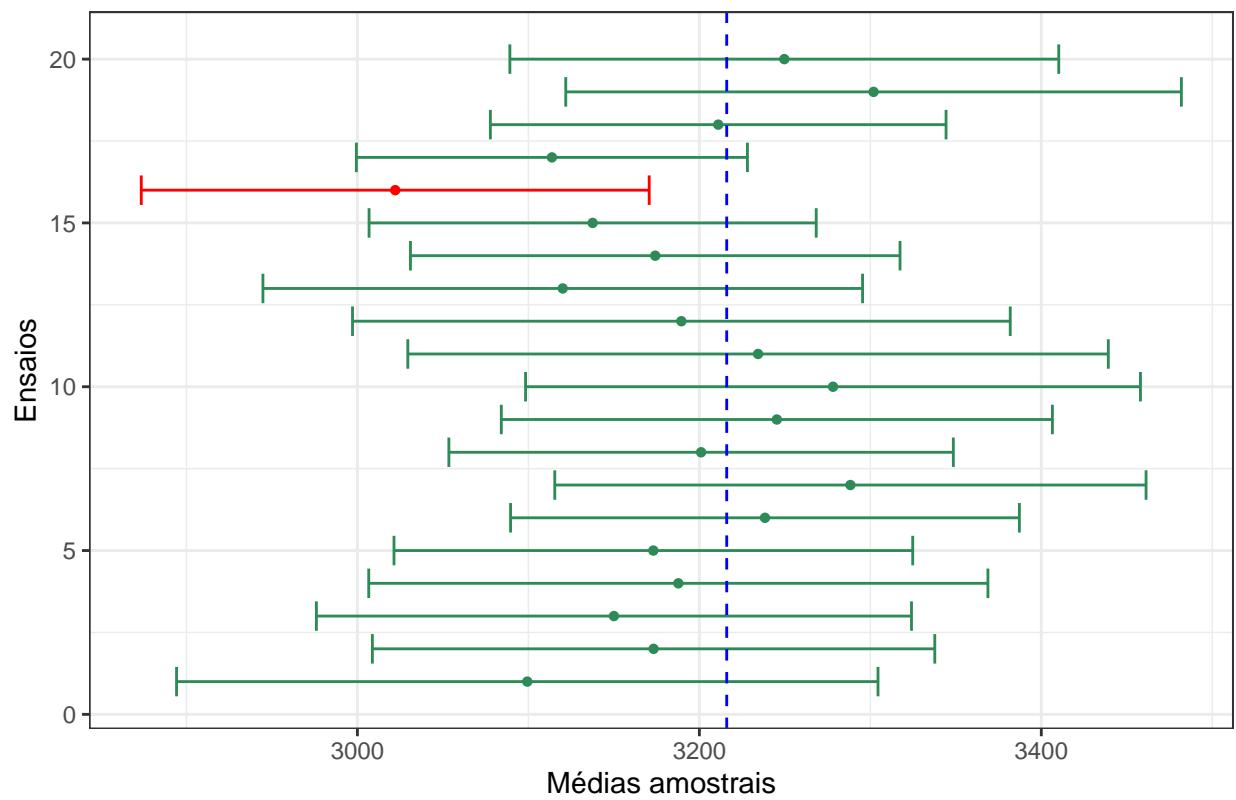


Figura 122: IC 95% - 20 amostras de $n = 30$

10.5 Estimação da média populacional: σ desconhecido

Com amostras pequenas, usar o modelo normal para construir intervalos de confiança, gera um erro, pois os pressupostos do teorema do limite central não são respeitados. Quando o desvio padrão populacional, σ , é desconhecido e o tamanho amostral é pequeno (< 30), a estimativa da média populacional é feita usando a *distribuição t*.

10.5.1 Distribuição t

A distribuição t , desenvolvida por *William Sealy Gosset*, em 1908, é semelhante à distribuição normal. Como a curva de distribuição normal, a curva de distribuição t é unimodal, simétrica (em forma de sino) em torno da média e nunca encontra o eixo horizontal. A área total sob uma curva de distribuição t é 1 ou 100%. A curva da distribuição t é mais plana do que a curva de distribuição normal padrão. Em outras palavras, ela é mais achatada e mais espalhada. No entanto, conforme o tamanho da amostra aumenta, a distribuição t aproxima-se da distribuição normal padrão.

O formato de uma curva de distribuição t particular depende do número de graus de liberdade. O número de graus de liberdade (gl) para uma distribuição t é igual ao tamanho da amostra menos um, ou seja, $gl = n - 1$.

O número de graus de liberdade é o único parâmetro da distribuição t . Há uma diferente distribuição t para cada número de graus de liberdade, portanto, a distribuição t se constitui em uma família de distribuições (Figura 123).

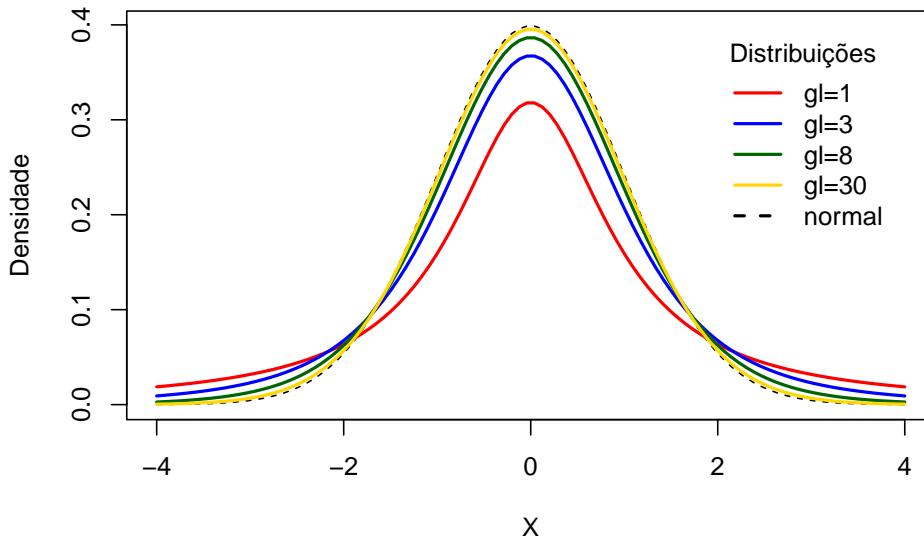


Figura 123: Curvas de distribuição t conforme o grau de liberdade e a distribuição normal.

Como a distribuição normal padrão, a média da distribuição padrão t é 0. Entretanto, ao contrário da distribuição normal padrão, cujo desvio padrão é 1, o desvio padrão de uma distribuição t é $\sqrt{\frac{gl}{gl-2}}$, para $gl > 2$, que sempre é maior do que 1. Assim, o desvio padrão de uma distribuição t é maior do que o desvio padrão da distribuição normal padrão.

Os valores de t_{crtico} podem ser obtidos usando a função `qt()` que usa os seguintes argumentos:

- $p \rightarrow$ probabilidade, igual a $1 - \frac{\alpha}{2}$, considerando-se bicaudal e $1 - \alpha$ quando unicaudal;
- $df \rightarrow$ graus de liberdade;
- $lower.tail \rightarrow$ lógico; se TRUE, informa a probabilidade da cauda inferior. O padrão é TRUE

Assim, o valor do t_{crtico} para $gl = 10$ é:

```
alpha <- 0.05
p <- 1 - (alpha/2)
gl = 10
t <- qt(0.975, 10, lower.tail = TRUE)
round(t, digits = 2)
```

```
## [1] 2.23
```

Dessa forma, a área compreendida entre ± 2.23 é igual a 95% (Figura 124):

$$p(-2,23 \leq t \leq 2,23) = 0,95$$

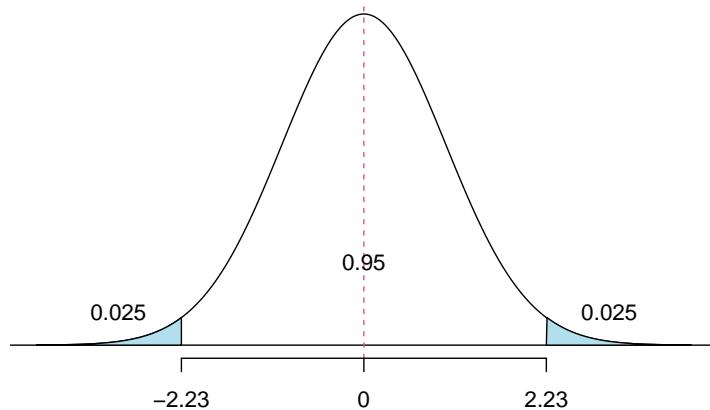


Figura 124: Distribuição t com $gl = 10$, bilateral.

Quando se considera apenas uma das caudas (unicaudal ou unilateral), o valor do t_{crtico} para $gl = 10$ é

```
t1 <- qt(0.95, 10, lower.tail = TRUE)
round(t1, digits = 2)
```

```
## [1] 1.81
```

Assim, a área abaixo de 1.81 é igual a 95% (Figura 125)

$$p(t \leq 1,81) = 0,95$$

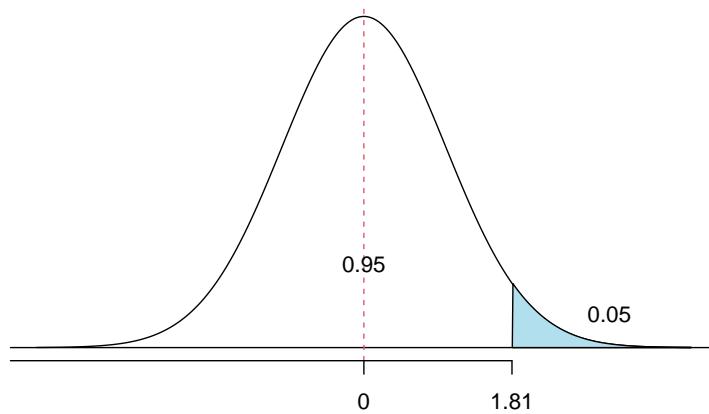


Figura 125: Distribuição t com $gl = 10$, unilateral.

10.5.2 Cálculo do intervalo de confiança com σ desconhecido

Dados

Neste momento, será usada a variável `mater$altura`, altura das mulheres em metros, pertencente ao conjunto de dados `dadosMater.xlsx`:

```
mater <- read_excel("dadosMater.xlsx")
dados <- mater %>% select (altura)
```

O conjunto de dados resultante contém a altura de 1368 mulheres, que para fins didáticos será considerada a população de estudo. Será extraída uma amostra¹⁴ de 30 mulheres e fazer inferência para essa população, *supondo que os parâmetros dessa população sejam desconhecidos*.

```
amostra <- sample (dados$altura, 30)
```

A maneira mais intuitiva de estimar a média da população com base na amostra, é, simplesmente, calcular a média e o desvio padrão da amostra.

```
media <- round(mean(amostra), 3)
media
```

```
## [1] 1.566
```

```
dp <- round(sd(amostra), 3)
dp
```

```
## [1] 0.073
```

¹⁴Lembrar que se for retirada outra amostra da mesma população de $n = 30$, a média e o desvio padrão serão diferentes!

Para termos maior precisão, vamos calcular os intervalos de confiança para esta estimativa, usando a distribuição t .

Cálculo manual do IC

Quando o desvio padrão da população (σ) não é conhecido, pode-se usar o seu estimador que é o desvio padrão da amostra (s), respeitando os pressupostos (93). Então, o erro padrão da média ($\sigma_{\bar{x}}$) pode ser estimado pelo $EP_{\bar{x}}$.

$$EP_{\bar{x}} = \frac{s}{\sqrt{n}}$$

O intervalo de confiança para a μ para um nível de confiança (NC) igual a $(1-\alpha) \times 100\%$ é igual a:

$$IC_{NC}(\mu) \rightarrow x \pm (t_{(1-\frac{\alpha}{2})} \times \frac{s}{\sqrt{n}})$$

Quando o tamanho amostral é grande, o valor de t se aproxima do valor de z , portanto, em situações em que não se conhece o desvio padrão populacional, não há muita diferença se houver uma aproximação de t para z (Tabela 9).

Tabela 9: Comparação dos valores z e t(gl).

z	n	gl	t
1,96	5	4	2,57
1,96	10	9	2,23
1,96	30	29	2,04
1,96	50	49	2,01
1,96	100	99	1,98
1,96	200	199	1,97
1,96	500	499	1,96
1,96	1000	999	1,96

Tem-se uma amostra de $n = 30$, $\bar{x} = 1.566$ e $s = 0.073$. Serão necessários os outros dados para obter o intervalo de confiança, usando uma distribuição t bicaudal e um nível de significância $\alpha = 0.05$:

```
n <- 30
alpha <- 0.05
p <- 1 - alpha/2

gl <- n - 1

tc <- qt(p, gl, lower.tail = TRUE)
round(tc, 3)
```

```
## [1] 2.045
```

```
EP <- round(dp/sqrt(n), 3)
EP
```

```
## [1] 0.013
```

Onde p é a probabilidade decorrente do nível de significância, gl são os graus de liberdade e EP é o erro padrão. Com estes dados, pode-se calcular o intervalo de confiança de 95%:

```
ic <- c((media - tc * EP), (media + tc * EP))
ic <- round(ic, 2)
ic

## [1] 1.54 1.59
```

Ou seja, temos uma confiança 95% de que a média populacional desconhecida se encontra entre 1.54, 1.59m

Cálculo usando uma função do R

O R possui algumas funções que calculam o intervalo de confiança para variáveis numéricas, baseadas na distribuição t . Entre elas, a função `CI()`, incluída no pacote `Rmisc`:

```
ic <- CI(amostra, ci = 0.95)
round(ic, 2)

## upper mean lower
## 1.59 1.57 1.54
```

10.6 Intervalo de Confiança para uma proporção populacional

Com frequência , necessitanos estimar uma proporção populacional. Como não é possível realizar um censo cada vez que se necessita encontrar o valor de uma proporção da população, normalmente se obtém resultados de pesquisas por amostragem. Portanto, para levar em conta a variabilidade nos resultados obtidos em diferentes pesquisas por amostragem, precisamos conhecer os procedimentos para estimar uma proporção da população.

10.6.1 Dados para estimar a proporção populacional

Vamos usar uma amostra aleatória de $n = 60$ da variável `mater$fumo` para estimar a proporção de mulheres fumantes no conjunto de dados `dadosMater.xlsx`.

```
dados <- read_excel("dadosMater.xlsx") %>%
  select (fumo) %>%
  slice_sample(n = 60)

head(dados)

## # A tibble: 6 x 1
##       fumo
##   <dbl>
## 1     2
## 2     2
## 3     1
## 4     2
## 5     2
## 6     2
```

A seleção mostra que temos 60 observações da variável `fumo` e que a mesma é do tipo numérica (`dbl`), 1 = fumante e 2 = não fumante. Portanto, ela é uma variável categórica e deve ser transformada para fator.

```
dados$fumo <- factor (dados$fumo,
                        levels = c (1,2),
                        label = c ("sim", "não"))
```

10.6.2 Cálculo da estimativa pontual da proporção

Nesta amostra, a proporção de fumantes é:

```
tab <- table(dados$fumo)
tabFumo <- round (prop.table (tab), 3)
tabFumo
```

```
##
##    sim    não
## 0.233 0.767
```

10.6.3 Cálculo do intervalo de confiança para a proporção

Cálculo Manual com Aproximação Normal

1^a etapa: verificar a premissa de que quando a proporção populacional é desconhecida a proporção pontual (\hat{p}) e o seu complemento ($\hat{q} = 1 - \hat{p}$) multiplicados, cada um, por n , devem ser maior do que 5.

```
n <- 60
(tabFumo) * n
```

```
##
##    sim    não
## 13.98 46.02
```

Ambos valores são maiores do que 5.

2^a Etapa: O intervalo pode ser estimado pela distribuição normal. O z_{crtico} é calculado:

```
alpha <- 0.05
p <- 1 - alpha/2
zc <- qnorm (p, mean = 0, sd = 1)
zc <- round(zc, 2)
zc
```

```
## [1] 1.96
```

3^a Etapa: Cálculo do erro padrão da proporção ($\sqrt{\frac{\hat{p}\times\hat{q}}{n}}$) e da margem de erro:

```
prop <- tabFumo [1]
EP <- sqrt((prop * (1 - prop))/n)
me <- zc * EP
me
```

```
##      sim
## 0.1069685
```

4^a Etapa: Intervalo de confiança

```
ic_prop <- c((prop - me), (prop + me))
round(ic_prop, 3)
```

```
##      sim      sim
## 0.126  0.340
```

Cálculo usando uma função

O chamado *Intervalo de Confiança Exato* corrigem as deficiências da aproximação normal. O R tem uma função para este cálculo: `BinomCI()` do pacote `DescTools`(94). É preferível usar o método de Clopper e Pearson que fornece o IC exato.

Os argumentos da função `BinomCI()` são:

- $x \rightarrow$ é o número de desfechos, sucessos;
- $n \rightarrow$ é o tamanho da amostra, número de ensaios;
- $p \rightarrow$ probabilidade, hipótese nula; se ignorada o padrão é 0,50;
- $conf.level \rightarrow$ nível de confiança, o padrão é 0.95;
- $method \rightarrow$ possui vários métodos para calcular intervalos de confiança para uma proporção binomial como: “clopper-pearson” (exact interval), “wilson”, “wald”, “agresti-coull”, “jeffreys”, “modified wilson”, “modified jeffreys”, “arcsine”, “logit”, “witting”, “pratt”. O método padrão é o de “wilson”. Qualquer outro método, há necessidade de solicitar;
- $sides \rightarrow$ hipótese alternativa padrão “two.sided” (bilateral), mas pode ser “right” ou “left” (unilateral a direita ou a esquerda, respectivamente).

```
IC <- BinomCI (tab[1],
                 n,
                 conf.level = 0.95,
                 method = "clopper-pearson")
round(IC, 3)
```

```
##      est lwr.ci upr.ci
## [1,] 0.233  0.134    0.36
```

Observe que existe uma pequena diferença entre os valores da aproximação normal e o exato, com método de “clopper-pearson”.

11 Teste de Hipóteses

Teste de hipóteses é um dos procedimentos básicos para a inferência estatística. Em um teste de hipóteses, testa-se uma teoria ou crença sobre um parâmetro populacional (95). Na maioria das vezes, como mencionado anteriormente, obtém-se informações a partir de uma amostra em função da impossibilidade ou dificuldade de se conseguir essas informações a partir da população. Portanto, extrapolar ou estender os resultados, obtidos de uma amostra, para a população, significa aceitá-los como representações adequadas da mesma.

Sabe-se que as estimativas amostrais diferem dos valores reais (populacionais) e o objetivo dos testes de hipóteses é estabelecer a probabilidade de essa diferença ser explicada pelo acaso. O teste de hipóteses fornece um sistema referencial para a tomada de decisão sobre a adequação ou não dos dados amostrais serem representativos de uma população. Este sistema referencial é a distribuição de probabilidade do evento observado (96).

11.1 Pacotes necessários para este capítulo

```
pacman::p_load(readxl,  
                 dplyr,  
                 BSDA)
```

11.2 Exemplo

Considere o exemplo dos recém-nascidos a termo da Maternidade do HGCS.

```
mater <- readxl::read_excel("dadosMater.xlsx") %>%  
  filter(ig>=37 & ig<42)  
  
mater$sexo <- factor(mater$sexo,  
                      levels = c(1, 2),  
                      labels = c("masc", "fem"))  
  
# Desvios padrão da variável pesoRN dos neonatos a termo (neonatos)  
neonatos <- mater %>% group_by(sexo) %>%  
  dplyr::summarise (n = n(),  
                    media = mean(pesoRN, na.rm = TRUE),  
                    sigma = sd(pesoRN, na.rm = TRUE))  
neonatos  
  
## # A tibble: 2 x 4  
##   sexo      n  media sigma  
##   <fct> <int> <dbl> <dbl>  
## 1 masc     592  3274.  458.  
## 2 fem      493  3147.  458.
```

Suponha que se faça uma afirmação de que existe uma diferença significativa entre os pesos ao nascer de meninos e meninas. Para confirmar isso, foi extraída uma amostra de 100 casos do conjunto `mater`, sem reposição, considerada como a população de estudo.¹⁵

¹⁵Conhecer os parâmetros da população, é muito raro. Aqui isto aconteceu, artificialmente, para fins didáticos.

```
set.seed (123)
mater100 <- mater %>% slice_sample(n = 100)
```

Este conjunto de dados contém 100 observações de 30 variáveis. Como serão usadas apenas as variáveis `sexo` e `pesoRN`, ele será reduzido e a seguir serão obtidas as medidas summarizadoras por sexo:

```
neonatos100 <- mater100 %>% select(sexo, pesoRN) %>%
  group_by(sexo) %>%
  dplyr::summarise(n = n(),
                    media = mean (pesoRN, na.rm = TRUE),
                    dp = sd (pesoRN, na.rm = TRUE))
neonatos100
```

```
## # A tibble: 2 x 4
##   sexo     n  media    dp
##   <fct> <int> <dbl> <dbl>
## 1 masc     57 3319.  428.
## 2 fem      43 3227.  515.
```

Esta amostra de 57 meninos e 43 meninas, informa que os meninos têm, em média, 3319.0350877g ao nascer e as meninas 3227.0930233g. Esta diferença de peso entre os sexos pode ter ocorrido devido ao acaso. Portanto, há necessidade de realizar um teste de hipóteses para tomar uma decisão sobre o parâmetro populacional (97). Esta diferença é grande o suficiente para rejeitar a hipótese de igualdade entre os pesos e concluir há uma diferença real entre eles?

11.3 Hipótese nula e alternativa

Uma *hipótese estatística* é qualquer consideração (suposição) feita em relação a um parâmetro populacional. Por meio do teste de hipóteses, se verifica se tais considerações são ou não compatíveis com os dados disponíveis.

No teste de hipóteses (TH), existem dois tipos de hipóteses, definidas como:

Hipótese nula(H_0): hipótese que afirma a não existência de diferença entre os grupos e, portanto, a diferença observada é atribuível ao acaso. É a hipótese a ser testada, aquela que se busca afastar. É escrita como:

$$H_0 : \mu_1 = \mu_2 \quad \text{ou} \quad \mu_1 - \mu_2 = 0$$

Hipótese alternativa (H_A): é a hipótese contrária, como o nome diz, alternativa à H_0 . Representa a posição de uma nova perspectiva, a conclusão que será apoiada se H_0 for rejeitada. Ela supõe que realmente existe uma diferença entre os grupos. É a hipótese que o pesquisador pretende comprovar. É escrita, em geral, simplesmente como havendo uma diferença entre os grupos, sem indicar uma direção, *hipótese bilateral ou bicaudal*:

$$H_A : \mu_1 \neq \mu_2 \quad \text{ou} \quad \mu_1 - \mu_2 \neq 0$$

Ou, se houver uma suspeita, através de um conhecimento prévio, apontar uma direção para a diferença, ou seja, usar uma *hipótese unilateral* ou *monocaudal*. Neste caso existe duas possibilidades:

- 1) Unilateral à direita:

$$H_A : \mu_1 > \mu_2 \quad ou \quad \mu_1 - \mu_2 > 0$$

Consequentemente,

$$H_0 : \mu_1 \leq \mu_2 \quad ou \quad \mu_1 - \mu_2 \leq 0$$

2) Unilateral à esquerda:

$$H_A : \mu_1 < \mu_2 \quad ou \quad \mu_1 - \mu_2 < 0$$

Consequentemente,

$$H_0 : \mu_1 \geq \mu_2 \quad ou \quad \mu_1 - \mu_2 \geq 0$$

A H_0 e H_A são opostas e mutuamente exclusivas. No teste de hipótese calcula-se a probabilidade de obter os resultados encontrados caso não haja efeito na população, ou seja, caso a H_0 seja verdadeira. Portanto, o TH é um teste de significância para a H_0 .

Exemplo (continuação)

Voltando aos recém-nascidos, as hipóteses seriam escritas da seguinte maneira, considerando uma H_A bilateral:

$$H_0 : \mu_{meninos} = \mu_{meninas} \quad ou \quad \mu_{meninos} - \mu_{meninas} = 0$$

$$H_A : \mu_{meninos} \neq \mu_{meninas} \quad ou \quad \mu_{meninos} - \mu_{meninas} \neq 0$$

11.4 Escolha do teste estatístico e regra de decisão

O teste estatístico escolhido depende do tipo de distribuição da variável, por exemplo, teste Z, teste t, teste F, qui-quadrado (χ^2). A decisão para rejeitar ou não a H_0 depende da magnitude do teste estatístico.

É fundamental verificar, para cada teste estatístico os seus pressupostos. Para a maioria dos testes, deve-se verificar a distribuição (normalidade), igualdade das variâncias entre os grupos (homoscedasticidade), independência entre os grupos, tipo de correlação, etc.

Realizado o teste estatístico, para rejeitar ou não rejeitar a H_0 , partindo do pressuposto de que ela é verdadeira, há necessidade de determinar uma *regra de decisão* que permita uma declaração fundamentada. Essa regra de decisão cria duas regiões, uma *região de rejeição* e uma *região de não rejeição* da H_0 , demarcadas por um *valor crítico*.

Este valor de referência é determinado pelo *nível de significância*, α , e deve ser explicitamente mencionado *antes* de se iniciar a pesquisa, pois é baseado nele que se fundamentam as conclusões da mesma. O nível de significância corresponde a probabilidade de rejeitar uma hipótese nula verdadeira. Quando a hipótese alternativa não tem uma direção definida, a área de rejeição, α , é colocada nas duas caudas (Figura 126, superior), dividindo a probabilidade ($\frac{\alpha}{2}$); quando houver indicação prévia de um sentido, a área de rejeição ficará a direita (Figura 126, inferior) ou a esquerda dependendo da direção escolhida.

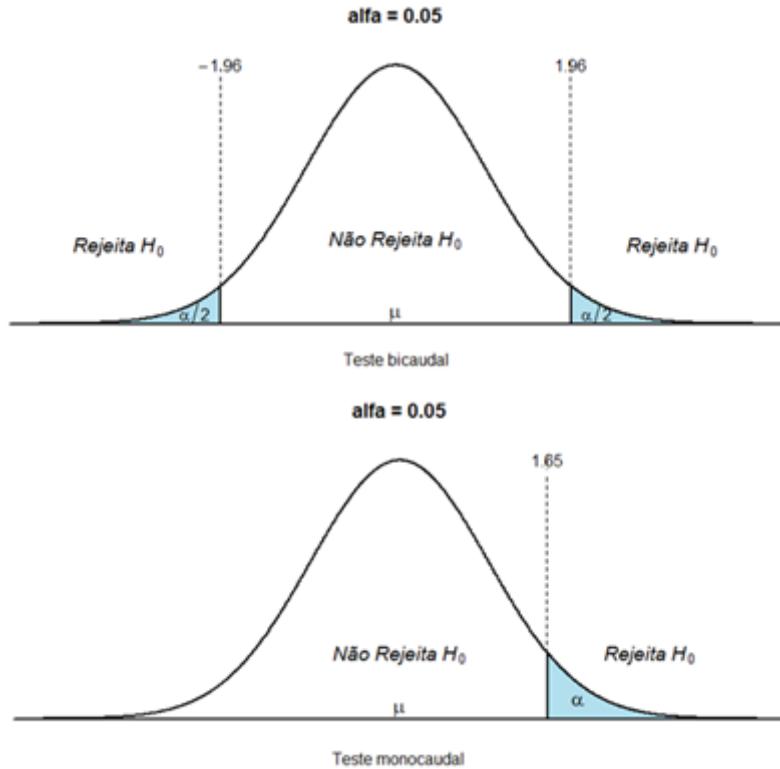


Figura 126: Regiões bicaudais (acima) e moncaudal à direita (abaixo) de rejeição e não rejeição da hipótese

Como se observa, ao se tomar uma decisão existe a possibilidade de se cometer *erros*. O primeiro erro é denominado de *erro tipo I* e ocorre quando, baseado na regra de decisão escolhida, uma hipótese nula verdadeira é rejeitada. Nesse caso, tem-se um resultado *falso positivo*. Há uma conclusão de que existe um efeito quando na verdade ele não existe. A probabilidade de cometer esse tipo de erro é α , o mesmo usado como nível de significância no estabelecimento da regra de decisão.

$$P(\text{rejeitar } H_0 | H_0 \text{ verdadeira}) = \alpha$$

Qual o valor de α que pode representar forte evidencia contra H_0 , reduzindo a possibilidade de erro tipo I?

O valor de α escolhido, apesar de arbitrário, deve corresponder a importância do que se pretende demonstrar, quanto mais importante, menor deve ser o valor de α . Nesses casos, não se quer rejeitar incorretamente H_0 mais de 5% das vezes. Isso corresponde ao nível de significância mais usado de 0,05 ($\alpha = 0,05$). Em algumas situações também são utilizados 0,01 e 0,10. Como mencionado, o valor de α deve ser escolhido antes de iniciar o estudo.

Existe uma outra possibilidade de erro, denominado de *erro tipo II*, que ocorre quando a hipótese nula é realmente falsa, mas com base na regra de decisão escolhida, não se rejeita essa hipótese nula. Nesse caso, o resultado é um *falso negativo*; não se conseguiu encontrar um efeito que realmente existe. A probabilidade de cometer esse tipo de erro é chamada de β .

$$P(\text{no rejeitar } H_0 | H_0 \text{ falsa}) = \beta$$

Na construção de um teste de hipótese, o erro tipo II é considerado menos grave que o erro tipo I. Entretanto, ele é bastante importante. Tradicionalmente, adota-se o limite de 0,10 a 0,20 para o erro tipo II.

Abaixo (Figura 127) estão resumidas as possíveis consequências na tomada de decisão em um teste de hipótese (98).

Erros tipo I e Tipo II		Realidade Populacional	
		Há diferença (H_0 Falsa)	Não há diferença (H_0 Verdadeira)
Decisão de acordo com o teste (baseada na amostra)	Teste significativo (Rejeita H_0)	Decisão correta $(1 - \beta) = \text{Poder do teste}$	Erro tipo I $\alpha = P$ (erro tipo I) = nível de significância
	Teste não significativo (Não rejeita H_0)	Erro tipo II $\beta = P$ (erro tipo II)	Decisão correta $1 - \alpha = \text{Confiança}$

Figura 127: Tomada de decisão e erros.

Exemplo (continuação)

O nível de significância escolhido será $\alpha = 0,05$ e como a amostra mater100 é constituída por 57 meninos e 43 meninas, onde os meninos têm média, 3319.0350877g ao nascer e as meninas 3227.0930233g. Esta amostra é proveniente de uma população cujo peso dos recém-nascidos têm distribuição normal. Em função destas características, para testar a hipótese de que não existe diferenças entre os pesos ao nascer, de acordo com o sexo, será usado o teste z .

Para um teste z , para um $\alpha = 0,05$, o valor crítico, para um teste bilateral, é igual a:

```
round (qnorm(0.975), 2)
```

```
## [1] 1.96
```

Ou seja, com é bilateral, o z_{crtico} é igual a ± 1.96 . Desta forma, se o resultado do teste $z_{calculado}$ for igual ou maior do que o z_{crtico} , rejeita-se a hipótese de igualdade entre os pesos dos recém-nascidos, masculinos e femininos

$$\begin{aligned}|z_{calculado}| < |z_{crtico}| &\rightarrow \text{não se rejeita } H_0 \\|z_{calculado}| \geq |z_{crtico}| &\rightarrow \text{rejeita - se } H_0\end{aligned}$$

O teste estatístico é usado para saber se a diferença média obtida através da amostra se afasta de forma significativa da diferença média populacional. Para isso é usado o erro padrão da média ($\sigma_{\bar{x}}$) que padroniza essa diferença em números de erros padrão. O teste z é calculado pela fórmula:

$$z = \frac{\bar{x}_1 - \bar{x}_2}{\sigma_{\bar{x}_1 - \bar{x}_2}} = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}}$$

No R base não há um teste z , há necessidade de instalar e carregar o pacote BSDA e usar a sua função `z.test()`.

```

meninos <- mater100 %>% filter(sexo == "masc")
meninas <- mater100 %>% filter(sexo == "fem")
dp1 <- neonatos100$dp[1]
dp2 <- neonatos100$dp[2]

teste_z <- z.test(meninos$pesoRN,
                   meninas$pesoRN,
                   alternative = "two.sided",
                   sigma.x = dp1,
                   sigma.y = dp2,
                   conf.level = 0.95)
teste_z

##
##  Two-sample z-Test
##
## data: meninos$pesoRN and meninas$pesoRN
## z = 0.94909, p-value = 0.3426
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -97.92831 281.81244
## sample estimates:
## mean of x mean of y
## 3319.035 3227.093

```

O $|z_{calculado}| < |z_{critico}|$, donde se conclui que não é possível rejeitar a H_0 e, portanto, não existe uma diferença estatisticamente significativa entre os pesos dos recém-nascidos, com 95% de confiança.

11.5 Valor P

Nas seções anteriores, foi discutido um procedimento onde se encontrou o valor de probabilidade tal que uma dada hipótese nula é rejeitada ou não é rejeitada, de acordo com o nível de significância, H_0 , fixado, pelo pesquisador, no início da pesquisa.

Essa abordagem do valor de probabilidade, mais comumente chamada de abordagem do valor P , fornece esse valor. Uma vez realizada a pesquisa, o pesquisador calcula a *probabilidade de obter um resultado tão ou mais extremo que o observado, uma vez que a hipótese nula é verdadeira*. O valor P também é conhecido como *nível descritivo do teste* (99).

O objetivo de um teste estatístico é transformar em probabilidade a magnitude do desvio verificado em relação ao valor esperado, fornecendo o valor P . A partir daí pode-se, também, definir a regra de decisão, usando esse valor P . Toma-se o valor predeterminado (em geral, 0,05) de α e, então, compara-se o valor P com α e toma-se a decisão. Usando essa abordagem, rejeita-se a H_0 se o valor $P < \alpha$ e não se rejeita se o valor $P > \alpha$. Costuma-se dizer que se o valor $P < \alpha$, o resultado é significativo e não significativo quando $P > \alpha$.

Uma boa parte dos pesquisadores, principalmente no início da carreira, ficam empolgados pelo conhecimento do valor P . Entretanto, deve ser sempre lembrado que encontrar o valor P não é o único foco da pesquisa. O foco deve estar dirigido ao *tamanho do efeito* (*effect size*). O valor P obtido pelo teste estatístico, vai informar apenas sobre a probabilidade de se cometer erro ao rejeitar ou não rejeitar a hipóteses nula.

Exemplo (continuação)

O teste realizado, `z.test()`, fornece o valor P de 0,3426. Este valor poderia ser obtido através da função `pnorm()` com o argumento `lower.tail = FALSE`. Além disso, como o teste é bilateral, o resultado deve ser multiplicado por 2. .

```
pnorm(0.94909, lower.tail = FALSE) * 2
```

```
## [1] 0.3425748
```

Concluindo, esta é uma probabilidade muito maior do que $\alpha = 0,05$, e ,em consequência, comete-se em erro, considerado grande, se a hipótese nula for rejeitada.

11.6 Poder do teste

Ao se planejar uma pesquisa é fixado previamente o valor máximo de alfa (α), probabilidade de erro tipo I, que será aceito e o valor de beta (β), probabilidade de erro tipo II, é calculado para a situação específica da pesquisa.

Considera-se *poder do teste estatístico* a probabilidade de o teste rejeitar uma hipótese nula quando ela é realmente falsa. Corresponde, na Figura 128 , a região à direita da linha vertical azul, isto é, a área de rejeição da hipótese nula, para um teste monocaudal à direita.

$$\text{Poder do teste} = P(\text{rejeitar } H_0 | H_0 \text{ falsa})$$

$$\text{Poder do teste} = 1 - P(\text{no rejeitar } H_0 | H_0 \text{ falsa})$$

$$\text{Poder do teste} = 1 - \beta$$

É comum dar pouca atenção à probabilidade de cometer um erro tipo II. Isto em geral acontece porque o único valor atribuído pelo investigador, antes de iniciar o teste, é o nível de significância, α , correspondente ao risco aceitável de rejeitar a hipótese nula.

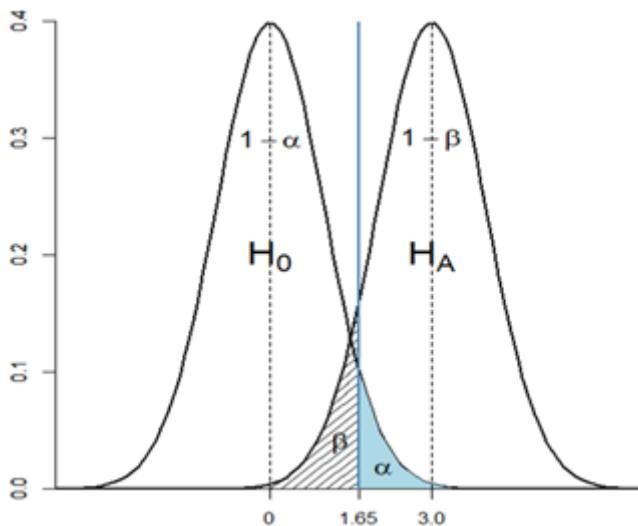


Figura 128: Nível de significância, probabilidade de erro tipo II, poder e nível de confiança em um teste de hipótese e a região de rejeição da hipótese nula (à direita da linha vertical azul).

Por outro lado, β pode assumir um de muitos valores. Suponha um estudo onde se deseja testar a hipótese nula de que algum parâmetro da população é igual a algum valor especificado. Se a hipótese nula for falsa e não se conseguir rejeitá-la, comete-se um erro do tipo II. Se o valor hipotético do parâmetro não for o valor verdadeiro, o valor de β depende de vários fatores:

1. o valor verdadeiro do parâmetro de interesse,
2. o valor hipotético do parâmetro,
3. o valor de α ,
4. o tamanho da amostra, n .

Para α e n fixos, então, antes de realizar um teste de hipótese, é possível obter muitos valores de β , calculando muitos valores para o parâmetro de interesse, dado que o valor hipotético é falso.

Para um determinado teste de hipótese, é interessante saber quão bem o teste controla os erros do tipo II. Se H_0 é de fato falsa, é importante saber a probabilidade de rejeitá-la. O poder de um teste fornece essa informação desejada. Pode ser calculado para qualquer valor alternativo do parâmetro sobre o qual se testa uma hipótese. Portanto, poder é a probabilidade de agir corretamente quando H_0 for falsa. Isso quer dizer que o poder é uma função da hipótese alternativa assumida como verdadeira.

11.7 Resumo do teste de hipóteses: Passos principais

11.7.1 Primeiro Passo: Dados

É extremamente importante conhecer a natureza dos dados. A sua descrição adequada, o objetivo e o delineamento da pesquisa indicarão o teste estatístico a ser usado.

11.7.2 Segundo Passo: Estabelecer as hipóteses estatísticas

As hipóteses, nula e alternativa, que devem ser claramente estabelecidas. A H_0 deve ter uma indicação de igualdade ($=$, \geq ou \leq) e é a hipótese a ser testada.

A H_0 e a H_A são hipóteses complementares. Ao estabelecer as hipóteses deve-se definir o tipo de hipótese alternativa:

- a) apontando simplesmente a existência de uma diferença e ela é bilateral; ou
- b) apontando que existe uma diferença e o sentido do desvio desejado (maior ou menor que a referência) e aí ela é unilateral. O teste unilateral só deve ser usado se houver informações prévias sobre o problema.

Finalmente, ter sempre em mente que a rejeição ou não de uma H_0 não implica em uma prova irrefutável.

11.7.3 Terceiro passo: Escolha do teste estatístico e regra de decisão

Escolher o teste a ser usado como mencionado acima. Verificar os seus pressupostos ¹⁶, executar o teste e comparar o resultado com o valor crítico predeterminado pelo nível de significância, α . Para um teste z , por exemplo, para um $\alpha = 0,05$, o valor crítico, para um teste bilateral, é igual a $\pm 1,96$, como visto anteriormente.

11.7.4 Quarto passo: Conclusão

A conclusão do teste, baseada no resultado do cálculo do teste estatístico, declara se este é significativo ou não, ou seja rejeita ou não a hipótese nula, conforme a regra de decisão, previamente estabelecida.

¹⁶Ao estudar cada um dos testes será comentado com mais detalhes sobre estes pressupostos

11.7.5 Quinto passo: Valor P

Calcular o valor P , probabilidade de a amostra observada, ou qualquer uma mais extrema que ela, ter sido gerada dentro das condições da hipótese nula. No exemplo visto, esta probabilidade é igual a 34.3%.

12 Comparação entre duas médias

Esta seção discute como testar uma hipótese sobre a diferença entre as médias de duas populações, $\mu_1 - \mu_2$, assumindo que os desvios padrão, σ_1 e σ_2 , dessas populações são desconhecidos, mas são considerados iguais.

O teste usado para realizar essa função é o *teste t de Student* (ou simplesmente *teste t*). Portanto, compara duas médias e verifica se a diferença entre elas é estatisticamente significativa. Em outras palavras, permite que se avalie se a diferença entre as médias é real ou é decorrente do acaso.

A necessidade de determinar se duas médias populacionais, representadas por amostras aleatórias representativas dessas populações, são diferentes entre si é uma situação extremamente frequente em pesquisas científicas. Por exemplo, se um grupo experimental difere de um grupo controle, se um grupo difere antes de depois de uma intervenção ou um *teste t* para uma amostra.

12.1 Pacotes necessários para este capítulo

```
pacman::p_load(readxl,  
                 dplyr,  
                 knitr,  
                 kableExtra,  
                 ggplot2,  
                 ggpubr,  
                 ggsci,  
                 car,  
                 rstatix,  
                 lsr,  
                 tidyverse)
```

12.2 Teste *t* para comparar amostras independentes

O teste *t* é usado quando há duas condições experimentais e participantes diferentes foram designados para cada condição.

12.2.1 Dados do exemplo

Será usado um conjunto de dados constituído por medidas de altura (em metros) de dois grupos de mulheres, pertencentes a duas regiões geográficas diferentes.

Estes dados podem ser obtidos [aqui](#). Faça o download e salve no seu diretório de trabalho.

12.2.1.1 Leitura dos dados Para a leitura dos dados, será usado a função `read_excel()` incluído no pacote `readxl` que deve ser instalado e carregado. Os dados serão recebidos por um objeto que será denominado de `dados`:

```
dados <- read_excel("dadosPop.xlsx")
```

Para visualizar os dados, pode-se usar a função `glimpse()` do pacote `dplyr`:

```
glimpse(dados)
```

```

## Rows: 60
## Columns: 3
## $ id      <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, ~
## $ altura <dbl> 1.50, 1.56, 1.63, 1.66, 1.60, 1.65, 1.49, 1.60, 1.56, 1.58, 1.5~
## $ pop     <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~

```

Observa-se que existem 60 mulheres, sendo 30 moradoras na região 1 e 30 na região 2. A variável `id` é a variável de identificação das mulheres (variável numérica), `altura` é uma variável numérica que corresponde a medida da altura em metros e `pop` é uma variável categórica, onde 1 é corresponde às mulheres da região 1 e 2 as da região2. No dataset, esta variável encontra-se como uma variável numérica e deverá ser transformada em fator.

Para uma visualização mais elegante e em uma apresentação mais amigável, pode-se usar a função `kable()` do pacote `knitr` e a função `kable_styling()` do pacote `kableExtra`. A função `kable()`, neste exemplo, usa a função `head()` embutida. Ao executar os códigos serão exibido apenas 10 linhas do banco de dados (se não for especificado, mostra apenas 6 linhas). Isto evita uma poluição visual (Tabela 10):

```

kable(head(dados, 10),
      booktabs = TRUE,
      col.names = c("Id", "Altura", "Pop"),
      caption = "Altura (em metros) de dois grupos de mulheres.") %>%
kable_styling(full_width = FALSE,
              latex_options = "hold_position") %>%
column_spec(1, width = "0.5in") %>%
column_spec(2, width = "1in") %>%
column_spec(3, width = "1in")

```

Tabela 10: Altura (em metros) de dois grupos de mulheres.

Id	Altura	Pop
1	1.50	1
2	1.56	1
3	1.63	1
4	1.66	1
5	1.60	1
6	1.65	1
7	1.49	1
8	1.60	1
9	1.56	1
10	1.58	1

12.2.1.2 Exploração e resumo dos dados

Inicialmente, a variável `pop` será transformada em fator:

```

dados$pop <- as.factor(dados$pop)

```

A seguir, calcular a média e o desvio padrão da variável `altura` de acordo com `pop`, usando a função `group_by()` e `summarise` do pacote `dplyr`

```

resumo <- dados %>%
  group_by(pop) %>%
  dplyr:: summarise(n = n(),
                     media = mean(altura, na.rm = TRUE),
                     dp = sd(altura, na.rm = TRUE),
                     mediana = median(altura, na.rm = TRUE),
                     me = 1.96 * dp/sqrt(n))
resumo

## # A tibble: 2 x 6
##   pop      n  media    dp mediana    me
##   <fct> <int> <dbl> <dbl>  <dbl> <dbl>
## 1 1        30  1.60 0.0621    1.6  0.0222
## 2 2        30  1.39 0.0736    1.38 0.0263

```

Para melhor observar os dados, ainda é possível construir um gráfico do tipo boxplot (Figura 129), usando o pacote `ggplot2`:

```

ggplot(data = dados) +
  geom_boxplot(aes(x = pop,
                   y = altura,
                   fill = pop)) +
  labs (x = "Populações",
        y = "Altura (m)") +
  theme_classic() +
  theme(legend.position="none")

```

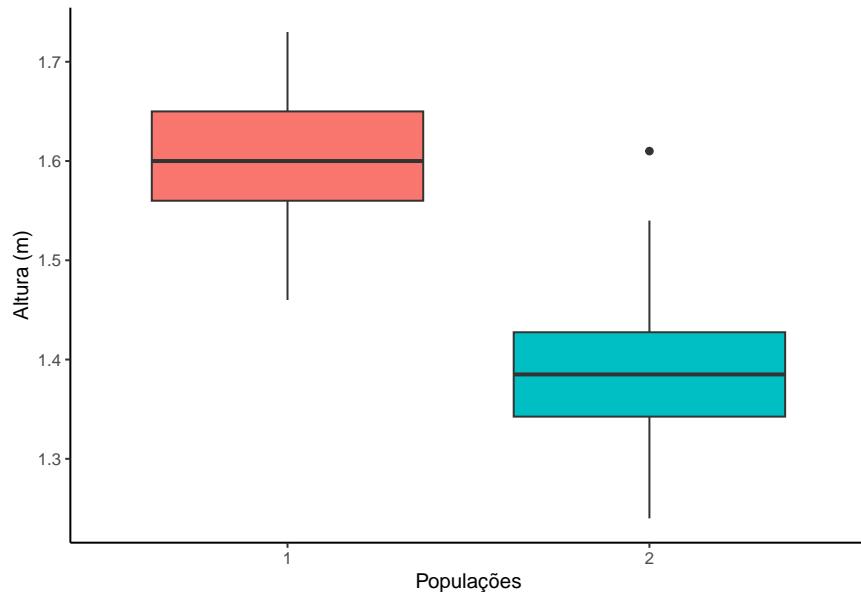


Figura 129: Boxplot dos dados

12.2.2 Definição das hipóteses estatísticas

As hipóteses comparam as médias dos dois grupos. Para um teste bicaudal, considerando pop1 a população da região 1 e pop2 a população da região 2, as hipóteses são escritas como:

$$H_0 : \mu_{pop1} = \mu_{pop2}$$
$$H_A : \mu_{pop1} \neq \mu_{pop2}$$

12.2.3 Definição da regra de decisão

O nível significância, α , escolhido é igual a 0.05. A distribuição t é dependente dos graus de liberdade, dados por:

No exemplo,

```
n1 <- resumo$n[1]  
n1
```

```
## [1] 30
```

```
n2 <- resumo$n[2]  
n2
```

```
## [1] 30
```

```
g1 <- n1 + n2 -2  
g1
```

```
## [1] 58
```

Para um $\alpha = 0,05$, o valor crítico de t para $g1 = 58$ para uma hipótese alternativa bicaudal é obtido com a função `qt(p, df)`, onde $df = g1$ e $p = 1 - \alpha/2$

```
alpha <- 0.05  
round (qt((1-alpha/2), g1), 3)
```

```
## [1] 2.002
```

Portanto, se

$$|t_{calculado}| < |t_{crtico}| \rightarrow \text{no se rejeita } H_0$$
$$|t_{calculado}| \geq |t_{crtico}| \rightarrow \text{rejeita se } H_0$$

12.2.4 Teste estatístico

12.2.4.1 Lógica do teste t O teste t compara as médias de duas amostras independentes, usando o erro padrão como métrica da diferença entre essas médias. Quanto maior o valor de t , maior a probabilidade de que as amostras pertençam a populações diferentes, correndo nessas circunstâncias a rejeição da hipótese nula (100).

Calcula-se o teste t com a seguinte equação:

$$t = \frac{(\bar{x}_1 - \bar{x}_2) - (\mu_1 - \mu_2)}{EP_d}$$

Onde EP_d é o erro padrão da diferença entre a médias $\bar{x}_1 - \bar{x}_2$. Se a hipótese nula for verdadeira, as amostras foram retiradas da mesma população e, portanto, $\mu_1 - \mu_2 = 0$. Assim, a equação fica:

$$t = \frac{(\bar{x}_1 - \bar{x}_2)}{EP_d}$$

O erro padrão da diferença $\bar{x}_1 - \bar{x}_2$ é calculado de maneiras diferentes:

- 1) Se a variâncias nos dois grupos forem iguais, usa-se:

$$EP_d = \sqrt{s_o^2 \left(\frac{1}{n_1} + \frac{1}{n_2} \right)}$$

Onde s_o^2 é a variância combinada ou conjugada que é, simplesmente, a média ponderada das variâncias dos grupos:

$$s_o^2 = \frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{(n_1 - 1) + (n_2 - 1)}$$

Quando os grupos têm o mesmo tamanho ($n_1 = n_2$), s_o^2 é simplesmente a média aritmética da variância dos grupos:

$$s_o^2 = \frac{s_1^2 + s_2^2}{2}$$

$$EP_d = \sqrt{\frac{2s_o^2}{n}}$$

- 2) Se as variâncias dos dois grupos forem diferentes:

$$EP_d = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$$

Esta explicação da lógica e dedução da estatística de teste serve para uma melhor compreensão de como o teste funciona, mas para executar um teste t não há necessidade disso, basta saber como encaminhar ao R e como interpretar o resultado fornecido por ele.

12.2.4.2 Pressupostos do teste t

O teste t assume que:

1. As amostras são independentes;
2. Deve haver distribuição normal. Entretanto, quando as amostras são grandes (teorema do limite central), isso não é muito importante;
3. Exista homocedasticidade, ou seja, as variâncias dos grupos devem ser iguais.

Violar o pressuposto de número 3 tem importância se os tamanhos dos grupos forem diferentes. Se os grupos tiverem o mesmo tamanho e a amostra for grande, este pressuposto torna-se menos importante, não importando muito se esta hipótese foi violada (101). O pressuposto tem mais importância em grupos pequenos e desiguais. Existe um teste, denominado *teste t de Welch* que corrige essa violação. É possível portanto, esquecer esse pressuposto e fazer o teste de Welch sempre.

Avaliação da normalidade

Uma boa parte dos procedimentos estatísticos são testes paramétricos¹⁷ com base na distribuição normal. Ou seja, se assume que a distribuição dos dados segue o modelo da distribuição normal. Se essa suposição não for atendida, a lógica por trás do teste de hipóteses pode ser violada.

Pode-se verificar a normalidade de maneira visual, observando o comportamento dos dados através de gráficos como o *histograma* (Figura 130) e o *gráfico Q-Q* (Figura 131). É útil também sobrepor uma distribuição normal no histograma, para fins de comparação.

```
mu1 <- resumo$media[1]
dp1 <- resumo$dp[1]
mu2 <- resumo$media[2]
dp2 <- resumo$dp[2]

par(mfrow=c(1,2))

hist(dados$altura[dados$pop == "1"],
      ylim = c(0, 7),
      xlim = c(1.4, 1.9),
      main= "População 1",
      ylab = "Densidade",
      xlab = "Altura (metros)",
      col ="steelblue",
      freq = FALSE,
      border = "white")
box (bty = "L")

curve (dnorm (x,
              mean=mu1,
              sd=dp1),
       col="red",
       lty=1,
       lwd=2,
       add=TRUE)

hist(dados$altura[dados$pop == "2"],
      ylim = c(0, 7),
```

¹⁷Teste paramétricos são testes estatísticos que se baseiam nos padrões da distribuição populacional da variável em estudo, por exemplo, a distribuição normal é descrita por dois parâmetros – média e desvio padrão – que são suficientes para se conhecer as probabilidades. Os testes que não requerem a especificação da forma de distribuição da população, ou seja, têm distribuição livre, são denominados de não paramétricos.

```

xlim = c (1.1, 1.7),
main= "População 2",
ylab = "Densidade",
xlab = "Altura (metros)",
col ="steelblue",
freq = FALSE,
border = "white")
box (bty = "L")

curve (dnorm (x,
               mean=mu2,
               sd=dp2),
       col="red",
       lty=1,
       lwd=2,
       add=TRUE)

```

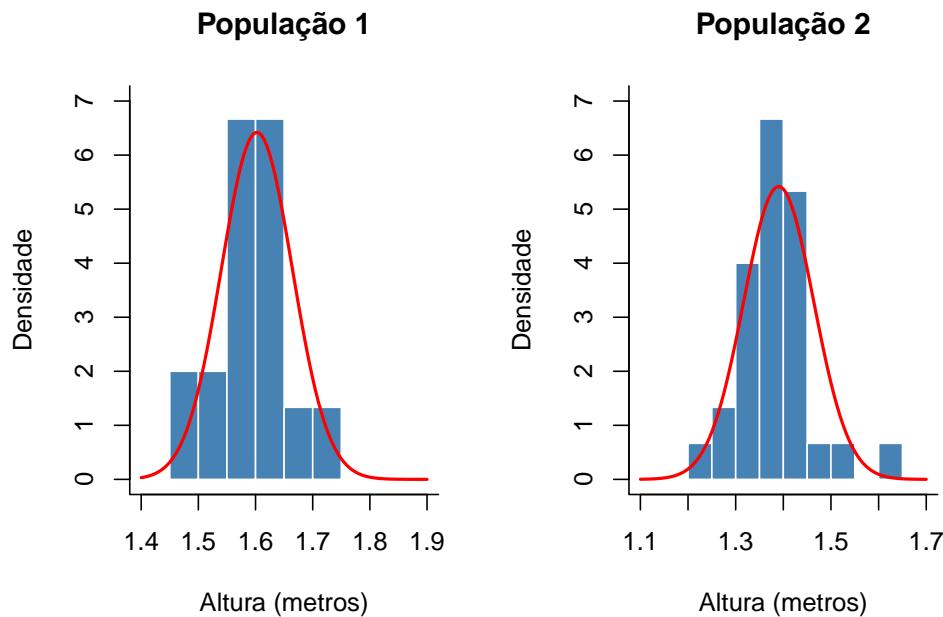


Figura 130: Histogramas da altura das populações

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

O gráfico *QQ* (ou gráfico quantil-quantil) desenha a correlação entre uma determinada amostra e a distribuição normal. Uma linha de referência de 45 graus também é plotada. Um gráfico Q-Q é um gráfico de dispersão criado plotando dois conjuntos de quantis um contra o outro. Se ambos os conjuntos de quantis vierem da mesma distribuição, veremos os pontos formando uma linha aproximadamente reta.

Se os valores caírem na diagonal do gráfico, a variável é normalmente distribuída. Os desvios da diagonal mostram desvios da normalidade. Para desenhar um gráfico Q-Q pode ser usado a função `ggqqplot ()` do pacote `ggnpubr` que produz um gráfico QQ normal com uma linha de referência, acompanhada de área sombreada, correspondente ao IC95%.

```
ggqqplot(dados, x = "altura", color = "pop") +
  labs(y = "Altura (m)",
       x = "Quantis teóricos")
```

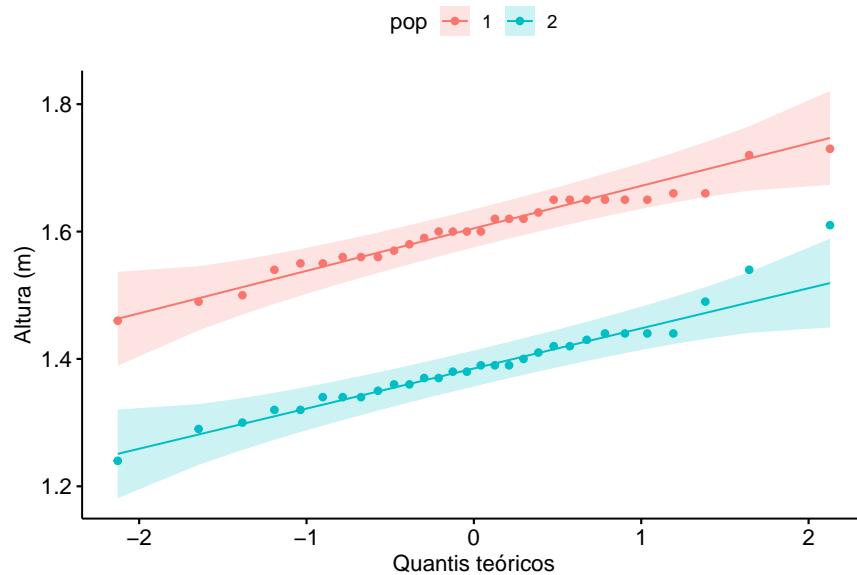


Figura 131: Gráficos Q-Q

Observando os gráficos, verifica-se que a variável `altura` tem uma distribuição visualmente normal aceitável, pois o histograma se ajusta à curva normal e os gráficos Q-Q mostram que os dados seguem a linha diagonal.

Outra maneira de analisar a normalidade é verificar se a distribuição como um todo se desvia de uma distribuição normal comparável. Para isso, usam-se *testes estatísticos de normalidade*. Os dois principais são o *teste de Shapiro-Wilk* e o *teste de Kolmogorov-Smirnov (K-S)*.

Esses testes comparam os dados da amostra com um conjunto de valores normalmente distribuídos com a mesma média e desvio padrão. Se o teste não for significativo ($P > 0,05$), informa-se que a distribuição da amostra não é significativamente diferente de uma distribuição normal. Se, no entanto, o teste for significativo ($P \leq 0,05$), a distribuição em questão será significativamente diferente de uma distribuição normal.

O método de Shapiro-Wilk é amplamente recomendado para teste de normalidade (102), (103), (104).

```
sw <- dados %>%
  dplyr::group_by(pop) %>%
  shapiro_test(altura)
sw
```

```
## # A tibble: 2 x 4
##   pop   variable statistic     p
##   <fct> <chr>      <dbl> <dbl>
## 1 1     altura      0.971 0.553
## 2 2     altura      0.948 0.154
```

Os resultados mostram que ambos valores P do teste, 0.553 e 0.154, estão acima de 0,05, corroborando com a não rejeição da normalidade dos dados.

Homogeneidade da Variância

Na visualização da Figura 132 nos dois grupos de mulheres, observa-se que há, entre os limites inferior e superior, uma dispersão das medidas em torno da região central que vai progressivamente diminuindo. Esta dispersão parece ser semelhante nos grupos. Isto sugere que haja *homogeneidade das variâncias*.

Portanto, homogeneidade da variância é o pressuposto de que a dispersão das medidas é aproximadamente igual em diferentes grupos de casos, ou que a dispersão dos valores são aproximadamente iguais em pontos diferentes da variável preditora.

```
ggplot(dados, aes(x = pop, y = altura)) +  
  geom_jitter(aes(shape = pop, color = pop),  
              position = position_jitter(0.2),  
              size = 1) +  
  stat_summary(aes(color = pop),  
              fun.data= "mean_sdl",  
              fun.args = list(mult=1),  
              geom = "pointrange") +  
  labs(x = "População",  
       y = 'Altura (m)') +  
  theme_classic() +  
  scale_fill_nejm() +  
  theme(legend.position="none")
```

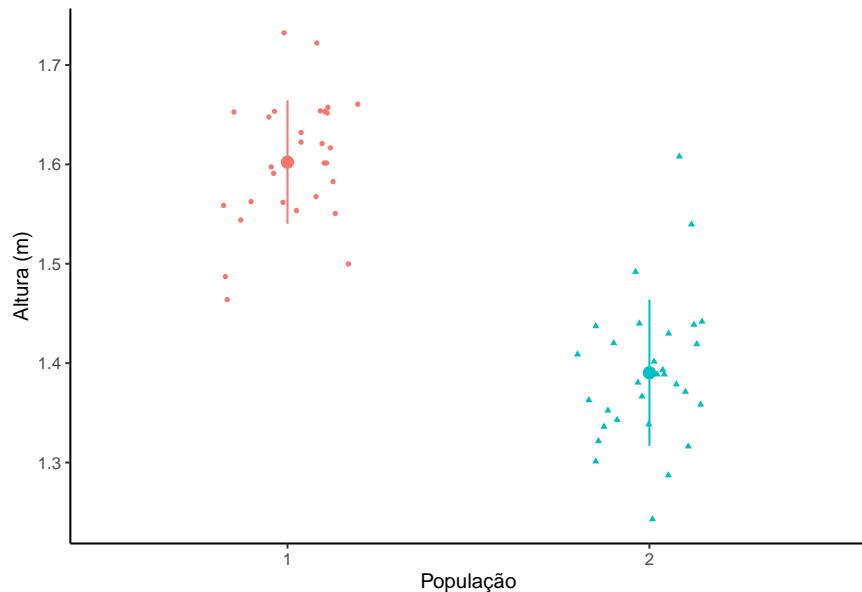


Figura 132: Gráfico mostrando a dispersão dos dados

Ao comparar grupos, essa suposição pode ser testada com o *teste de Levene*. Neste teste, a H_0 é todas as variâncias são iguais. No R, a função que calcula o teste é `leveneTest()` do pacote `car` (105). Os argumentos são:

- $y \rightarrow$ variável de resposta para o método padrão ou um objeto `lm` ou `fórmula`. Se y for um objeto de modelo linear ou uma fórmula, as variáveis do lado direito do modelo devem ser todas fatores e devem ser completamente cruzadas;
- $group \rightarrow$ fator que define os grupos;

- *center* → O nome de uma função para calcular o centro de cada grupo; `mean` fornece o teste de Levene original; o padrão, `median`, fornece um teste mais robusto;
- *data* → conjunto de dados para avaliar a `formula`.

```
levene <- leveneTest(altura~pop, center = mean, data = dados)
levene
```

```
## Levene's Test for Homogeneity of Variance (center = mean)
##          Df F value Pr(>F)
## group    1  0.1599 0.6907
##          58
```

O valor $P > 0,05$, confirma a impressão visual dos boxplots de que os grupos têm homogeneidade das variâncias, portanto a hipótese nula de igualdade das variâncias não pode ser rejeitada.

Um outro teste que compara duas variâncias poderia ser usado. É o teste F que pode ser calculado com a função `var.test()` do pacote `stats`, incluído no R base. Seus argumentos podem ser consultados na ajuda do R.

```
var.test(altura~pop, alternative = "two.sided", data = dados)
```

```
##
##  F test to compare two variances
##
## data: altura by pop
## F = 0.71253, num df = 29, denom df = 29, p-value = 0.3667
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.3391409 1.4970303
## sample estimates:
## ratio of variances
## 0.7125337
```

O resultado mostra que a conclusão sobre a homogeneidade das variâncias é a mesma.

12.2.4.3 Execução do teste *t* para grupos independentes A análise dos pressupostos mostra que os mesmos não foram violados. A próxima etapa é executar o teste, usando a função `t.test()`, incluído no pacote `stats` que tem os seguintes argumentos:

- *x* → um vetor numérico de valores de dados;
- *y* → um vetor numérico de valores de dados;
- *alternative* → especificar a hipótese alternativa, deve ser `two.sided` (padrão), `greater` ou `less`. Basta especificar apenas a letra inicial;
- *mu* → número que indica o valor verdadeiro da média (ou diferença nas médias se for realizar um teste de duas amostras);
- *paired* → indicador lógico (padrão é FALSE) para um teste t emparelhado (`paired = TRUE`);
- *var.equal* → indicador lógico que indica se as duas variâncias devem ser tratadas como iguais. Se TRUE, então a variância combinada é usada para estimar a variância, caso contrário, a aproximação de Welch (para os graus de liberdade) é usada;
- *conf.level* → nível de confiança do intervalo, o padrão é 0.95;
- *formula* → fórmula tipo $y \sim x$ onde y é uma variável numérica que fornece os valores dos dados e x um fator com dois níveis que fornecem os grupos correspondentes;

- *data* → opcional, matriz ou banco de dados contendo as variáveis da fórmula;
- *subset* → vetor opcional que especifica um subconjunto de observações a ser usado;
- *na.action* → uma função que indica o que deve acontecer quando os dados contêm NAs.

Método 1

Dados apresentados em dois vetores numéricos x e y .

```
t.test(dados$altura[dados$pop == "1"],
       dados$altura[dados$pop == "2"],
       var.equal = TRUE)

##
## Two Sample t-test
##
## data: dados$altura[dados$pop == "1"] and dados$altura[dados$pop == "2"]
## t = 12.056, df = 58, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.1767997 0.2472003
## sample estimates:
## mean of x mean of y
## 1.602333 1.390333
```

Método 2

Emprega uma fórmula (*variável resposta ~ grupo*) com os dados do banco de dados:

```
teste <- t.test(altura~pop,
                var.equal = TRUE,
                data = dados)
teste

##
## Two Sample t-test
##
## data: altura by pop
## t = 12.056, df = 58, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group 1 and group 2 is not equal to 0
## 95 percent confidence interval:
##  0.1767997 0.2472003
## sample estimates:
## mean in group 1 mean in group 2
## 1.602333 1.390333
```

O resultado de ambos os métodos são exatamente iguais. Ele entrega as seguintes informações:

- t é o valor estatístico do teste t (12.056),
- df são os graus de liberdade ($gl = 58$),
- $p\text{-value}$ é o nível de significância do teste t (valor P).
- $conf.int$ é o IC95% da diferença média (0.177, 0.247);
- $sample\ estimates$ estimativas da amostra são o valor médio dos grupos da amostra (1.602, 1.39).

12.2.5 Conclusão

Como $|t_{calculado}| = 12.056 > |t_{0,05;58}| = 2.002$, rejeita-se H_0 . Observa-se que o valor P é muito pequeno ($< 0,0001$) e, portanto, a diferença observada nas médias dos dois grupos deve ser assumida como significativa. Assim, as duas populações são diferentes e deve-se admitir que as amostras são procedentes de populações com médias de altura diferentes, com probabilidade de erro extremamente pequena. A estimativa da diferença média ($\mu_1 - \mu_2$) é fornecida pelo intervalo de confiança de 95% (0.177, 0.247). Observe que o valor zero não está contido no intervalo e isto confirma a não significância estatística da diferença.

Concluindo, a altura das mulheres da população 1 e a altura das mulheres da população 2 são diferentes, a diferença ($\mu_1 - \mu_2$) encontrada é estatisticamente significativa ($t = 12.056$, $gl = 58$, $P < 0,0001$), com uma confiança de 95%.

Esta conclusão pode ser visualizada em um gráfico:

- 1) Calcular o teste estatístico com a função `t_test()` do pacote `rstatix` (106) e colocar em um objeto denominado `t_teste`:

```
t_teste <- t_test(dados,
                    altura~pop,
                    p.adjust.method = "none",
                    paired = FALSE,
                    var.equal = TRUE,
                    alternative = "two.sided",
                    mu = 0,
                    conf.level = 0.95,
                    detailed = TRUE)
```

- 2) Construir um boxplot (Figura 133) exibindo o valor P

```
p <- ggplot(dados, aes(x=pop, y=altura)) +
  geom_boxplot(aes(color = pop)) +
  scale_color_nejm() +
  theme_classic() +
  theme(legend.position="none")
p +
  labs(x = "Região",
       y = 'Altura (m)',
       title = 'Comparação da Altura de Mulheres',
       subtitle = get_test_label(stat.test = t_teste,
                                correction = "none",
                                detailed = TRUE,
                                type = "expression",
                                p.col = "p",),
       caption = 'petronioliveira@gmail.com')
```

- 3) Ou um gráfico de barra de erro (Figura 134) com IC95%

```
ggplot(resumo,
       aes(x=pop, y=media, fill=pop)) +
  geom_point(stat = "summary",
             fun = "mean",
```

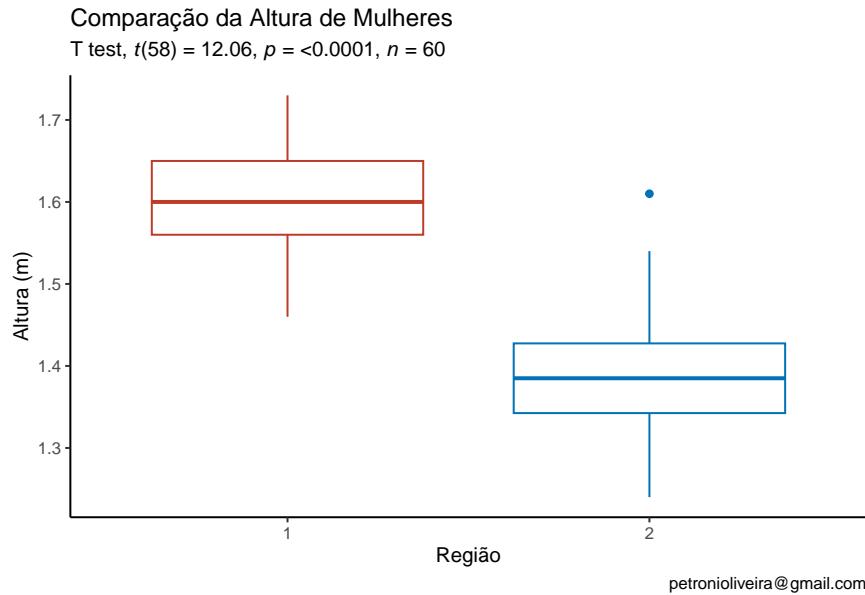


Figura 133: Boxplots comparando os dois grupos

```

size = 1,
show.legend = FALSE) +
geom_bar(width = 0.5, stat="identity", color= "black",
position=position_dodge()) +
geom_errorbar(aes(ymin=media-me, ymax=media+me), width=.1,
position=position_dodge(width = 0.2)) +
labs(title="Comparação da Altura de Mulheres",
subtitle = get_test_label(stat.test = t_teste,
correction = "none",
detailed = TRUE,
type = "expression"),
x="Região",
y = "Altura (m)",
caption = "petronioliveira@gmail.com")+
theme_classic() +
scale_fill_nejm(alpha = 0.5) +
theme(legend.position="none")

```

12.2.6 Tamanho do Efeito

A significância estatística deve ter uma atenção relativa do pesquisador, pois ela apenas mede a probabilidade de rejeitar uma hipótese nula, uma vez que ela seja verdadeira. Ajudam a determinar, em uma pesquisa, a significância dos resultados encontrados em relação à hipótese nula, mas não informam nada em relação a magnitude do efeito. Por exemplo, mostra se determinado tratamento afeta as pessoas, mas não dizem quanto isso afeta.

O tamanho do efeito (*effect size*) é uma medida quantitativa da magnitude do efeito. Quanto maior o tamanho do efeito, mais forte é a relação entre duas variáveis. É possível observar o tamanho do efeito ao comparar dois grupos quaisquer para ver quão substancialmente diferentes eles são.

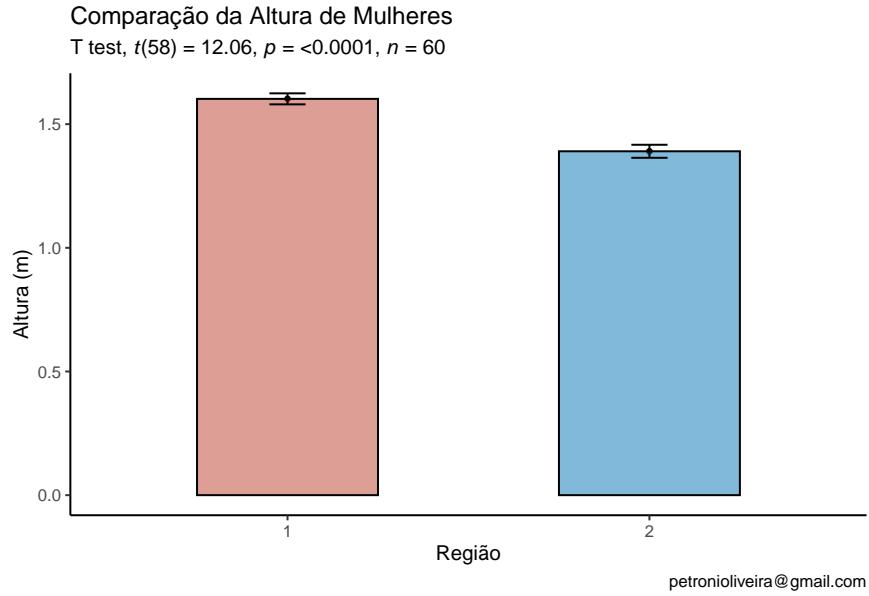


Figura 134: Gráfico de barra de erro comparando os dois grupos

Normalmente, em ensaios clínicos tem-se um grupo de tratamento e um grupo de controle. O grupo de tratamento é uma intervenção que se espera efetue um resultado específico. O valor do tamanho do efeito mostrará se a terapia teve um efeito pequeno, médio ou grande. Isso tem mais relevância do que simplesmente informar o tamanho do valor P .

12.2.6.1 d de Cohen Também conhecida como *diferença média padronizada*, o d de Cohen (107) (108) é uma medida adequada e bastante popular para encontrar a magnitude do efeito na comparação entre duas médias.

Para calcular a diferença média padronizada se verifica a diferença entre as médias dos dois grupos e se divide pelo desvio padrão conjugado:

$$d = \frac{(\bar{x}_1 - \bar{x}_2)}{s_o}$$

Onde,

$$s_o = \sqrt{\frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_1 + n_2 - 2}}$$

Voltando ao exemplo da altura das mulheres em duas regiões, o d de Cohen é calculado, usando a função `cohensD()` do pacote `lsr` que usa os seguintes argumentos:

- $x \rightarrow$ um vetor numérico de valores de dados, variável preditora;
- $y \rightarrow$ um vetor numérico de valores de dados, variável resposta;
- *formula* → Fórmula na forma variável *resposta* ~ *grupo*;
- *data* → dataframe ou matriz;
- *method* → Qual versão da estatística d devemos calcular? Os valores possíveis são *pooled*(padrão), *x.sd*, *y.sd*, *corrected*, *raw*, *paired* e *unequal*;

- μ → O valor “nulo” contra o qual o tamanho do efeito deve ser medido. Quase sempre é 0 (padrão); raramente especificado.

Assim, o d de Cohen pode ser obtido da seguinte forma:

```
d <- lsR::cohensD (altura ~ pop, data = dados)
d
## [1] 3.112768
```

Bastante simples! Agora, como interpretar este resultado de $d = 1,1$ (arredondado)? Sua interpretação não é intuitiva, recomenda-se usar a Tabela 11 para interpretar (107).

Tabela 11: Tamanho do Efeito

d	Interpretação
< 0,2	insignificante
0,2 < 0,5	pequeno
0,5 < 0,8	médio
$\geq 0,8$	grande

Dessa forma, as alturas das mulheres diferem significativamente ($P < 0,0001$) de acordo com a região, sendo que as mulheres da região 1 são bem mais altas do que as da região 2 e a magnitude dessa diferença é grande ($d = 3,1$).

12.3 Teste t para grupos pareados

Um *teste t pareado* é usado para estimar se as médias de duas medidas relacionadas são significativamente diferentes uma da outra. Esse teste é usado quando duas variáveis contínuas são relacionadas porque são coletadas do mesmo participante em momentos diferentes (antes e depois), de locais diferentes na mesma pessoa ao mesmo tempo ou de casos e seus controles correspondentes.

12.3.1 Dados do exemplo

O banco de dados é constituído por uma amostra de 15 escolares portadores de asma não controlada. Fizeram avaliação da sua função pulmonar no início do uso de um novo corticoide inalatório. Após 60 dias, repetiram a avaliação da função pulmonar. Para baixar o banco de dados, clique [aqui](#). Faça o download para o seu diretório de trabalho.

12.3.1.1 Leitura e transformação dos dados Crie um objeto `dados` para recebê-lo, a partir do diretório de trabalho, executando o seguinte código:

```
dados <- read_excel("dadosPar.xlsx")
```

A função `read_excel()` do pacote `readxl` abre o arquivo `dadosPar.xlsx`. Eles podem ser visualizados, usando a função `head()` que mostra estrutura e o formato dos dados, exibindo as 6 primeiras linhas:

```
head(dados)

## # A tibble: 6 x 3
##       id basal final
##   <dbl> <dbl> <dbl>
## 1     1   1.3   1.53
## 2     2   1.47  1.63
## 3     3   2.06  2.35
## 4     4   1.95  2.7
## 5     5   1.47  2.01
## 6     6   1.13  1.53
```

O dataframe `dados` está no formato amplo (*wide*). Será transformado para o formato longo (*long*), usando a função `pivot_longer()` do pacote `tidyverse`. Este processo não é obrigatório, mas vamos realizá-lo para fins de treinamento. O novo banco de dados será atribuído ao denominado `dadosL`. A função `pivot_longer ()` necessita dos seguintes argumentos:

- `dados` → dataframe a ser pivotado, transformado;
- `cols` → colunas a serem transformadas no formato longo;
- `names_to` → Especifica o nome da coluna a ser criada a partir dos dados armazenados nos nomes das colunas de dados;
- `values_to` → Especifica o nome da coluna a ser criada a partir dos dados armazenados nos valores das células;
- ... → possui outros argumentos. Ver ajuda.

```
dadosL <- dados %>%
  pivot_longer(c(basal, final),
               names_to = "momento",
               values_to = "medidas")
```

A estrutura do banco de dados `dadosL` é:

```
head(dadosL)
```

```
## # A tibble: 6 x 3
##       id momento medidas
##   <dbl> <chr>      <dbl>
## 1     1 basal        1.3
## 2     1 final       1.53
## 3     2 basal        1.47
## 4     2 final       1.63
## 5     3 basal        2.06
## 6     3 final       2.35
```

12.3.1.2 Medidas Resumidoras Para resumir as variáveis, serão usadas as funções `group_by ()`, `summarise ()` e `mutate ()` do pacote `dplyr`, aplicadas ao formato longo `dadosL`:

```
resumo <- dadosL %>%
  group_by(momento) %>%
  dplyr::summarise(n = n(),
```

```

    media = mean(medidas, na.rm = TRUE),
    dp = sd (medidas, na.rm = TRUE),
    mediana = median (medidas, na.rm = TRUE),
    IIQ = IQR (medidas, na.rm =TRUE),
    ep = dp/sqrt(n),
    me = ep * qt(1 - (0.05/2), n - 1))
resumo

```

```

## # A tibble: 2 x 8
##   momento     n  media    dp mediana   IIQ    ep    me
##   <chr>    <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 basal      15  1.31  0.427   1.26  0.48  0.110 0.236
## 2 final      15  1.69  0.471   1.59  0.38  0.122 0.261

```

12.3.1.3 Visualização dos dados

1) Tabela

É possível exibir os dados, tanto o banco de dados `dados` como o `dadosL`, de uma maneira mais elegante, usando a função `kable()` do pacote `knitr` e a função `kable_styling()` do pacote `kableExtra`. A função `kable ()` usa a função `head()` embutida. Ao executar os códigos, se não for especificado, é mostrado apenas 6 linhas. Será mostrado o formato amplo e todas as suas 15 linhas:

```

kable(head(dados, 15),
  booktabs = TRUE,
  col.names = c("Id", "Basal", "Final"),
  caption = "Função pulmonar de 15 escolares asmáticos antes-e-depois \\\\ do uso de um corticoide ina
kable_styling(position = "center",
  latex_options = "hold_position") %>%
  column_spec(2, width = "1.5in") %>%
  column_spec(3, width = "1.5in")

```

2) Gráficos

Apenas, por uma questão didática, serão apresentadas várias maneiras de mostrar os dados visualmente. Podem ser usados qualquer um dos tipos a seguir, pois todos dão, praticamente, a mesma informação.

Gráfico de barra de erro

```

resumo %>%
  ggplot(aes(x=momento, y=media, fill=momento)) +
  geom_bar(stat="identity", width = 0.4, color="black") +
  geom_point() +
  geom_errorbar(aes(ymin=media-me, ymax=media+me), width=0.1,
    position=position_dodge(.9)) +
  labs(title="Basal x Final - IC95%" ,
    x="Momento", y = "Volume Forçado em 1 seg (L)")+
  theme_classic() +
  theme(legend.position="none") +
  scale_fill_manual(values=c("cyan4","cyan3"))

```

Tabela 12: Função pulmonar de 15 escolares asmáticos antes-e-depois do uso de um corticoide inalatório

Id	Basal	Final
1	1.30	1.53
2	1.47	1.63
3	2.06	2.35
4	1.95	2.70
5	1.47	2.01
6	1.13	1.53
7	1.48	1.66
8	0.94	1.59
9	1.05	1.50
10	0.87	1.61
11	0.75	1.17
12	1.26	1.30
13	1.21	1.41
14	0.78	1.00
15	1.99	2.37

Neste gráfico (Figura 135), a altura da barra representa a média do *Volume Forçado em 1 seg* (VEF1) nos diferentes momentos (basal e final). O erro corresponde a margem de erro (me) a partir do ponto (média), ou seja, é o intervalo de confiança de 95%.

Boxplot

```
dadosL %>%
  ggplot(aes(x = momento, y = medidas, fill = momento)) +
  geom_errorbar(stat = "boxplot", width = 0.1) +
  geom_boxplot(outlier.color = "red",
               outlier.shape = 1,
               outlier.size = 1) +
  scale_fill_manual(values = c("cyan4","cyan3")) +
  ylab("Volume Forçado em 1 seg (L)") +
  xlab("Momento") +
  stat_summary(fun = mean,
              geom = "point",
              shape = 19, size = 2, color="red") +
  theme_classic() +
  theme(text = element_text(size = 12)) +
  theme(legend.position = "none")
```

A altura da caixa dos boxplots (Figura 136) é o intervalo interquartil (IIQ) e corresponde a 50% dos dados. A linha que corta horizontalmente a caixa é a mediana. Os bigodes da caixa (whiskers) em suas extremidades são os limites inferior e superior dos dados, excluindo os valores atípicos (outliers), representado no boxplot final por um ponto vermelho, acima do limite superior. Os pontos em vermelho (dentro das caixas) representam as médias.

Gráfico de linha

```
resumo %>%
  ggplot(aes(x=momento, y=media, group=1)) +
  geom_line(linetype ='dashed') +
```

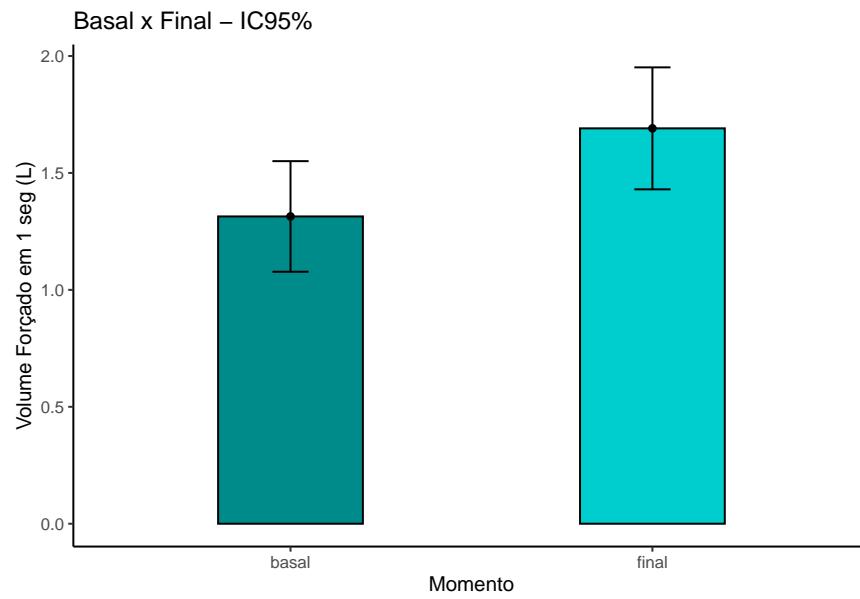


Figura 135: Gráfico de barra de erro comparando o grupo antes-e-depois

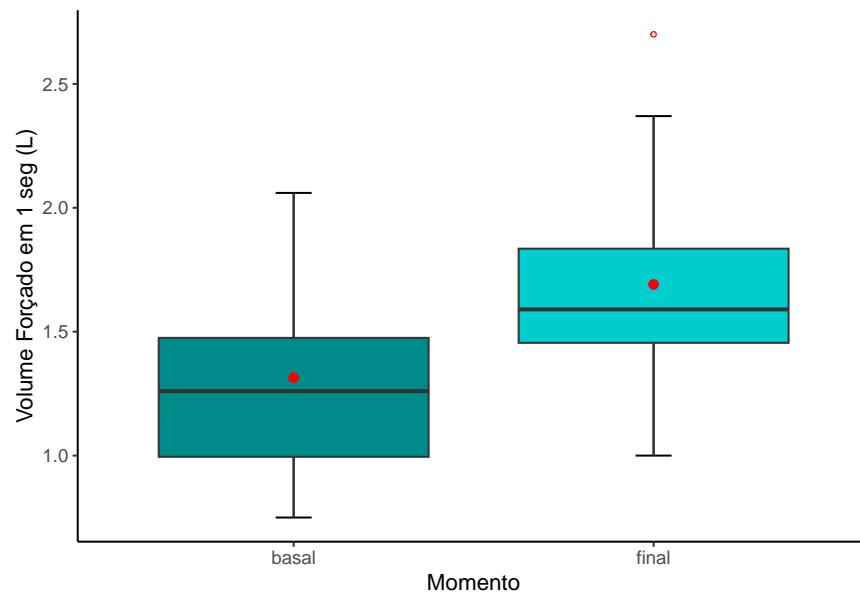


Figura 136: Boxplots comparando o grupo antes-e-depois

```

geom_errorbar(aes(ymin=media - me,
                   ymax=media + me),
               width=0.1,
               linewidth = 1,
               col = c("cyan4","cyan3")) +
geom_point(size = 3, color = c("cyan4", "cyan3")) +
theme_classic()+
labs(x='Momento',
     y='Volume Forçado em 1 seg (L)')

```

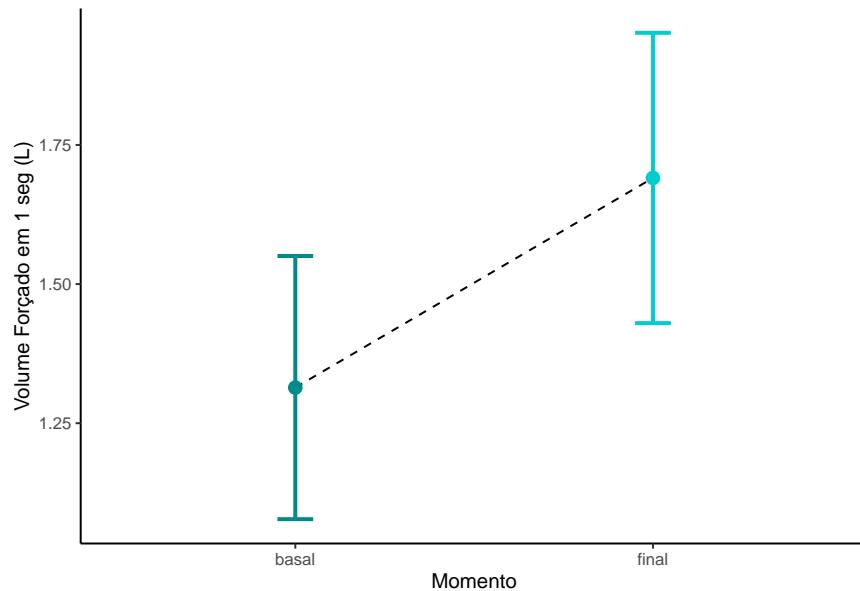


Figura 137: Gráfico de linha comparando o grupo antes-e-depois

Este gráfico de linha (Figura 137) com representação da margem de erro tem a mesma interpretação do gráfico de barra de erro. A escolha do tipo de gráfico depende da ênfase do autor sobre os dados.

12.3.1.4 Criação de uma variável que represente a diferença entre as médias A diferença entre as média basal e final será atribuída ao nome D. Esta ação será realizada, utilizando o banco de dados amplo (dados):

```

dados$D <- dados$basal - dados$final
head (dados)

```

```

## # A tibble: 6 x 4
##       id basal final      D
##   <dbl> <dbl> <dbl> <dbl>
## 1     1    1.3  1.53 -0.23
## 2     2    1.47 1.63 -0.16
## 3     3    2.06 2.35 -0.29
## 4     4    1.95 2.7  -0.75
## 5     5    1.47 2.01 -0.54
## 6     6    1.13 1.53 -0.4

```

Atenção, agora, o banco de dados apresenta uma nova variável `dif`, pois o foco do teste t pareado é essa diferença entre as médias, basal e final, a média das diferenças.

Resumo da variável D

Ao resumo será atribuído ao nome `sumario` (sem acento):

```
sumario <- dados %>%
  dplyr::summarise(media = mean (D),
                    dp = sd (D),
                    mediana = median (D),
                    IIQ = IQR (D),
                    min = min (D),
                    max = max (D))
sumario

## # A tibble: 1 x 6
##   media     dp mediana    IIQ   min     max
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 -0.377  0.218   -0.38  0.285 -0.75 -0.0400
```

Existe uma diferença de 0.38L entre o VEF1 basal e o final. A pergunta que se faz é: Esta diferença tem significância estatística? Os gráficos sugerem que sim!

12.3.2 Definição das hipóteses estatísticas

Será usado um teste bicaudal. Se a intervenção não produz efeito, então:

$$H_0 : \mu_D = 0$$

Se a intervenção produz efeito, então:

$$H_A : \mu_D \neq 0$$

Regra de decisão

O nível significância, α , escolhido é igual a 0,05. A distribuição da estatística do teste, sob a H_0 , é a distribuição t que é dependente dos graus de liberdade. O número de graus de liberdade é igual ao número de observações menos 1, neste caso são o número de pares menos 1.

$$gl = n - 1$$

Onde n é o número de pares. No exemplo, $n = 15$, consequentemente:

$$gl = 15 - 1 = 14$$

Para um $\alpha = 0,05$, o valor crítico de t para $gl = 14$ para uma hipótese alternativa bicaudal:

```
alpha <- 0.05
round(qt(1 - alpha/2, 14), 3)
```

```
## [1] 2.145
```

Portanto, se

$$|t_{calculado}| < |t_{critico}| \rightarrow \text{no rejeitar } H_0$$

$$|t_{calculado}| > |t_{critico}| \rightarrow \text{rejeitar } H_0$$

12.3.3 Teste estatístico

12.3.3.1 Lógica do teste A estatística do teste t dependente é a mesma do teste t independente r dada por:

$$T = \frac{\bar{D} - \mu_D}{EP_D}$$

Como na equação do teste t para amostras independentes, sob a hipótese nula igual a zero, $\mu_D = 0$, assim, a equação fica:

$$T = \frac{\bar{D}}{EP_D}$$

A estimativa do erro padrão das diferenças é dada por:

$$EP_D = \frac{s_D}{\sqrt{n}}$$

Como não existe informação sobre a variância das diferenças, seu valor será estimado por s_D^2 que é dado por:

$$s_D^2 = \frac{\sum(D_i - \bar{D})^2}{n - 1}$$

Onde D_i são as diferença individuais ($x_1 - y_1, x_2 - y_2, \dots, x_n - y_n$). Desta forma, o desvio padrão das diferenças, s_D , é:

$$s_D = \sqrt{\frac{\sum(D_i - \bar{D})^2}{n - 1}}$$

Da mesma maneira que no teste t para grupos independentes, essa demonstração serve para uma melhor compreensão de como o teste funciona, mas para executar este teste t não há necessidade disso, basta saber como encaminhar ao R, como será visto adiante.

12.3.3.2 Pressupostos do teste O teste t pareado assume que os seguintes pressupostos devem ser atendidos:

- (1) Os dados devem ser dependentes;
- (2) A variável desfecho deve estar em uma escala contínua;
- (3) As diferenças entre os pares devem ter distribuição normal.

Ao usar um teste t pareado, a variação entre os pares de medidas é a estatística mais importante e a variação entre os participantes, como no teste t de duas amostras independentes, é de pouco interesse, não havendo necessidade de se verificar se as variâncias dos grupos são iguais.

Para testar o pressuposto de *normalidade* das diferenças, usa-se a variável criada da diferença entre os pares, D . Verifica-se a normalidade dessa variável com o teste Shapiro-Wilk, usando a função `shapiro.test()` do pacote `stats`, incluída no R base.

```
shapiro.test (dados$D)
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data: dados$D  
## W = 0.94216, p-value = 0.4103
```

Além disso, um gráfico Q-Q (Figura 138) pode ser usado para avaliar a normalidade, com a função `ggqqplot()` do pacote `ggpubr` que produz um gráfico QQ normal com uma linha de referência, acompanhada de área sombreada, correspondente ao IC95%

```
ggqqplot (dados$D) +  
  labs(y = "Diferença Basal-Inicial",  
    x = "Quantis teóricos")
```

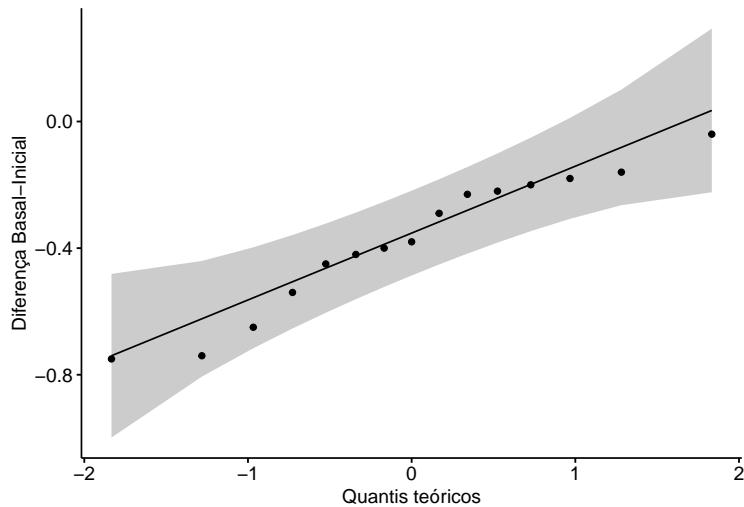


Figura 138: Gráfico Q-Q para avaliar a normalidade

Os resultados do teste de Shapiro-Wilk e gráfico QQ, mostram que a H_0 de normalidade da variável D não é rejeitada, apesar de haver uma pequena assimetria à esquerda que não impede o prosseguimento da análise.

12.3.3.3 Execução do teste estatístico O cálculo do teste t pareado usa a mesma função do teste t para amostras independentes, `t.test()`, mudando o argumento `paired =TRUE`(padrão) por `paired =FALSE`. Assim:

```

teste_par <- t.test(dados$basal,
                     dados$final,
                     alternative = "two.sided",
                     paired = TRUE,
                     conf.level = 0.95,
                     var.equal=TRUE)
teste_par

## 
## Paired t-test
##
## data: dados$basal and dados$final
## t = -6.6969, df = 14, p-value = 1.016e-05
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
## -0.4973000 -0.2560333
## sample estimates:
## mean difference
## -0.3766667

```

- t é o valor estatístico do teste t pareado,
- df são os graus de liberdade ,
- $p\text{-value}$ é o valor P do teste t.
- $conf.int$ é o IC95% da diferença média;
- $sample\ estimates$ é estimativa da diferença média

Também pode ser calculado usando a fórmula $y \sim x$ com os dados no formato longo (`dadosL`):

```

t.test(formula = medidas ~ momento,
       data = dadosL,
       alternative = 'two.sided',
       paired = TRUE)

## 
## Paired t-test
##
## data: medidas by momento
## t = -6.6969, df = 14, p-value = 1.016e-05
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
## -0.4973000 -0.2560333
## sample estimates:
## mean difference
## -0.3766667

```

12.3.4 Conclusão

Conclui-se que o VEF1 dos escolares asmáticos se modificou significativamente entre o início e após 60 dias do uso de um novo medicamento com uma confiança de 95%. A diferença ($\mu_{basal} - \mu_{final}$) encontrada é estatisticamente significativa ($t = -6.6969$, $gl = 14$, $P = 1.0156211 \times 10^{-5}$), com uma confiança de 95%.

Observe que o intervalo de confiança de 95% da diferença de -0.38 está todo abaixo de zero (-0.5, -0.26), confirmindo a significância.

12.3.5 Tamanho do Efeito

O tamanho do efeito pode ser determinado, também, com o teste d de Cohen, usando a função `cohensD()` do pacote `lsr`:

```
d_par <- lsr::cohensD (dados$basal, dados$final)
d_par
```

```
## [1] 0.8379499
```

Dessa forma, o uso do novo corticoide inalatório modificou significativamente o VEF1 dos escolares asmáticos com o uso de um novo corticoide inalatório ($P < 0,00001$), mostrando um aumento deste e que a magnitude dessa diferença é grande ($d = 0,84$).

Os resultados poderiam ser apresentados usando um gráfico de linha (Figura 139), aproveitando a função `t_test()` do pacote `rstatix` e colocar em um objeto denominado `t_par`.

```
t_par <- t_test(dadosL,
  medidas~momento,
  p.adjust.method = "none",
  paired = TRUE,
  alternative = "two.sided",
  mu = 0,
  conf.level = 0.95,
  detailed = TRUE)

resumo %>%
  ggplot(aes(x=momento, y=media, group=1)) +
  geom_line(linetype ='dashed') +
  geom_errorbar(aes(ymin=media - me,
                     ymax=media + me),
                width=0.1,
                size = 1,
                col = c("cyan4","cyan3")) +
  geom_point(size = 2) +
  theme_classic()+
  labs(title="Avaliação do Uso de Corticosteroide Inalatório",
       subtitle = get_test_label(stat.test = t_par,
                                 correction = "none",
                                 detailed = TRUE,
                                 type = "expression"),
       x="Momento",
       y = "Volume Forçado em 1 seg (L)",
       caption = "d Cohen = 0,84")+
  theme_classic() +
  theme(legend.position="none")
```

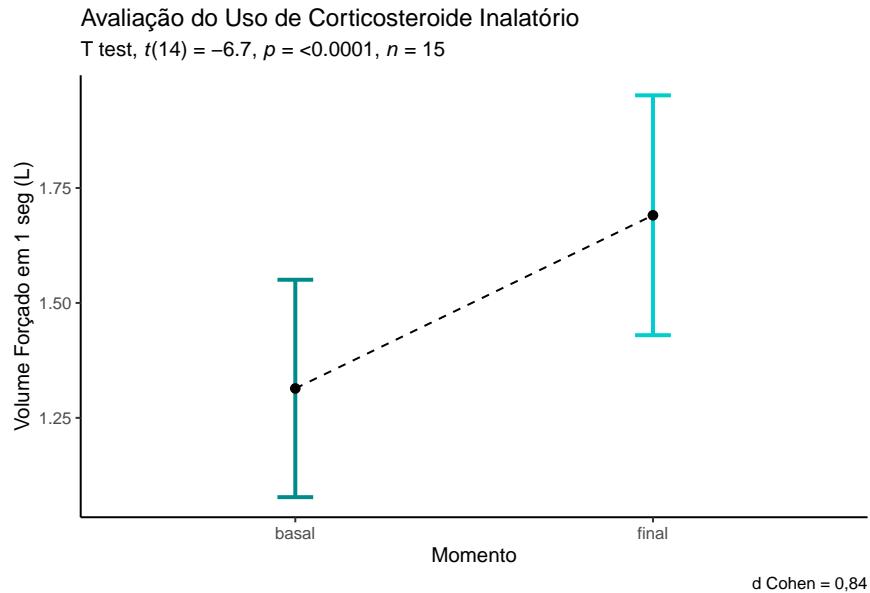


Figura 139: Gráfico de linha comparando o grupo antes-e-depois

13 Análise de Variância

13.1 Pacotes necessários para este capítulo

```
pacman::p_load(readxl,
  dplyr,
  fastGraph,
  knitr,
  kableExtra,
  ggplot2,
  ggpubr,
  ggsci,
  car,
  rstatix,
  effectsize,
  emmeans)
```

13.2 ANOVA de um fator

A *análise de variância (ANOVA) de um fator*, também conhecida como ANOVA de uma via, é uma extensão do teste t independente para comparar duas médias em uma situação em que há mais de dois grupos. Dito de outra forma, o teste t para uso com duas amostras independentes é um caso especial da análise de variância de uma via.

A ANOVA de um fator compara o efeito de uma variável preditora (variável independente, fator) sobre uma variável contínua (desfecho). Por exemplo, verificar se a intensidade do tabagismo na gestação (não fumantes, fumantes leves, moderados ou pesados) afetam o peso dos recém-nascidos. O gráfico de boxplots parece mostrar que sim (Figura 140).

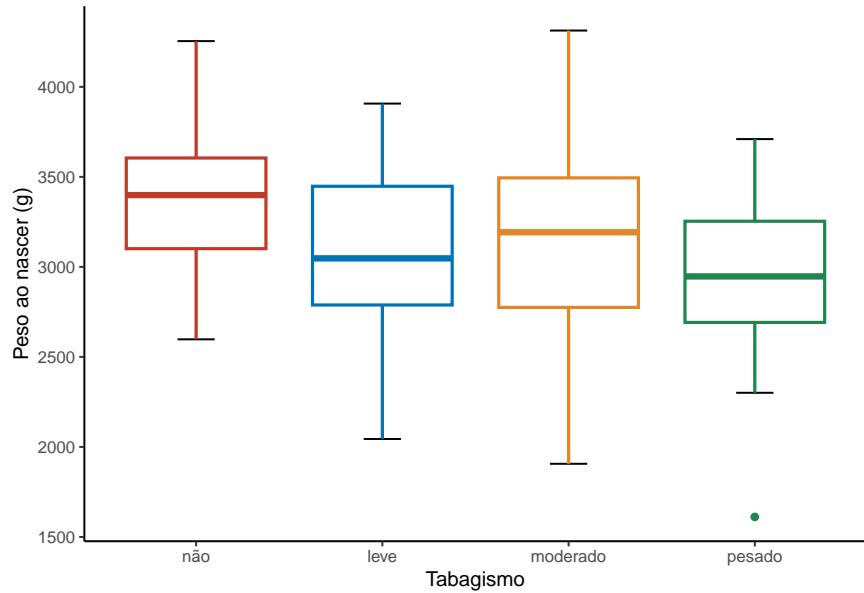


Figura 140: Impacto do tabagismo materno no peso ao nascer

Inicialmente, para analisar os grupos, se ficaria tentado a fazer comparações por pares usando um teste t de amostras independentes. Com existem quatro grupos, é possível compará-los realizando seis teste, grupo 1 versus grupo 2, grupo 1 versus grupo 3, grupo 1 versus grupo 4, grupo 2 versus grupo 3, grupo 2 versus grupo 4 e grupo 3 versus grupo 4. Se os dados têm k grupos são necessários $\frac{k!}{2!(k-2)!}$ testes.

A probabilidade de um erro do tipo I não ocorrer para cada teste t é de 0,95 (isto é, $1 - 0,05$), supondo um $\alpha = 0,05$. Os três testes são independentes; portanto, a probabilidade de um erro do tipo I não ocorrer nos seis testes é de $(0,95)^6 = 0,735$. Portanto, a probabilidade de ocorrer pelo menos um erro do tipo I nos seis testes t de duas amostras é de $1 - 0,735$ ou 0,265 (26,5%), o que é mais alto do que o nível de significância definido de 0,05 (109).

Portanto, uma ANOVA de um fator é usada para verificar as diferenças entre vários grupos dentro de um fator, reduzindo assim o número de comparações em pares e a probabilidade de ocorrer um erro tipo I.

13.2.1 Lógica do Modelo da ANOVA

O procedimento de ANOVA é utilizado para testar a hipótese nula de que as médias de três¹⁸ ou mais populações são as mesmas contra hipótese alternativa de que nem todas as médias são iguais.

No capítulo de comparação de duas médias, foi usado um teste para comparar duas variâncias, denominado de *teste F*. Este teste, é uma razão entre duas variâncias e recebeu este nome em homenagem a Sir Ronald Aylmer Fisher.

A variância é uma medida de dispersão que mensura como os dados estão espalhados em torno da média. Quanto maior o seu valor, maior a dispersão.

Considere a Figura 141, onde está representada a distribuição de uma variável X em três grupos independentes. Pode-se, claramente, distinguir observações provenientes dessas distribuições, pois a sobreposição delas é pequena. Cada uma dela se dispersa pouco em torno da média.

Agora, observe o Figura 142, onde a distribuição da variável X é mostrada, mantendo as mesmas médias, mas com variâncias maiores. Isto torna claro que se o objetivo é distinguir observações provenientes desses

¹⁸Pode ser usada também para comparar a média de duas populações e o resultado será o mesmo de um teste t para amostras independentes.

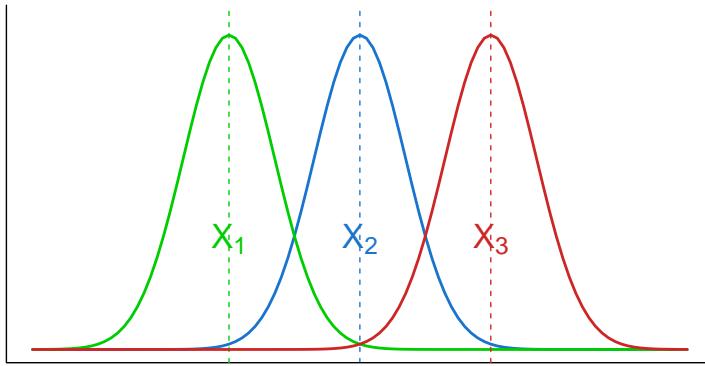


Figura 141: Três distribuições diferentes

grupos não basta avaliar suas médias, há necessidade de comparar a variação entre os grupos com a variação dentro de cada grupo (110).

Se a variação entre os grupos for grande quando comparada à variação dentro de cada grupo, aumenta a probabilidade de reconhecer a proveniência das observações (Figura 141). Entretanto, se a variação entre os grupos for pequena comparada à variação dentro do grupo, torna difícil a distinção de observações provenientes dos grupos (Figura 142).

Portanto, usar o teste F para determinar se as médias de grupo são iguais é apenas uma questão de incluir as variâncias corretas na razão. Na ANOVA com um fator, a estatística F é a razão dos estimadores das variâncias entre e dentro dos grupos.

$$F = \frac{\text{varância ENTRE os grupos}}{\text{varância DENTRO dos grupos}}$$

Quando o valor de F fica próximo de 1, significa que as variâncias são muito próximas; quando F é significativamente maior do que 1, é possível distinguir os indivíduos de diferentes grupos. Ou seja, se o objetivo for mostrar que as médias são diferentes, será bom que a variância dentro dos grupos seja baixa. Pode-se pensar na variância dentro do grupo como o ruído que pode obscurecer a diferença entre os sons (as médias). No gráfico A, o valor de F seria alto e em B seria baixo.

Como saber se o valor de F é alto o suficiente? Um único valor F é difícil de interpretar sozinho. Há necessidade de colocá-lo em um contexto maior antes que seja possível interpretá-lo. Para fazer isso, usa-se a distribuição F para calcular as probabilidades.

13.2.2 Distribuição F

A razão entre a variabilidade entre os grupos e a variabilidade dentro do grupo segue uma distribuição F quando a hipótese nula é verdadeira. Quando se realiza uma ANOVA com um fator obtém-se um valor F . No entanto, se forem extraídas várias amostras aleatórias do mesmo tamanho da mesma população e fosse repetida a mesma análise, o resultado seriam muitos valores F diferentes, constituindo uma distribuição amostral, denominada de *distribuição F* .

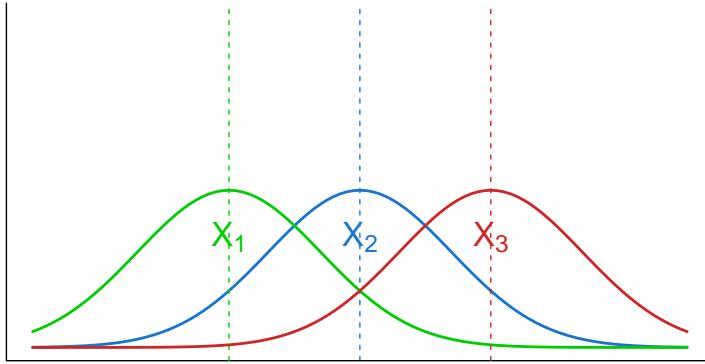


Figura 142: Distribuições com mesmas médias da figura anterior, mas variâncias maiores

Dessa forma, como a distribuição F assume que a hipótese nula é verdadeira, é possível colocar o resultado de qualquer valor F , resultante do teste de ANOVA, e determinar quão consistente ele é com a hipótese nula e calcular a probabilidade. A probabilidade que se quer calcular é a probabilidade de observar uma estatística F que é pelo menos tão alta quanto o valor que o estudo obteve. Essa probabilidade permite determinar quão comum ou raro é o valor F , sob a suposição de que a hipótese nula é verdadeira. Se a probabilidade for pequena o suficiente, pode-se concluir que dados são inconsistentes com a hipótese nula. Como já foi mostrado em outros momentos, essa probabilidade é o valor P .

O formato de uma curva de distribuição F depende do número de graus de liberdade. No entanto, a distribuição F tem dois números de graus de liberdade: *graus de liberdade para o numerador* (variância entre) e *graus de liberdade para o denominador* (variância dentro). Esses dois graus de liberdade são os parâmetros da distribuição F . Cada combinação de graus de liberdade fornece uma curva de distribuição F diferente. As unidades de uma distribuição F são denotadas por F , que assume apenas valores positivos. Como as distribuições normal, t e qui-quadrado (veja no capítulo específico), a distribuição F é uma distribuição contínua. A forma de uma curva de distribuição F é inclinada para à direita, mas a assimetria diminui à medida que o número de graus de liberdade aumenta, conforme observado na Figura 143.

As principais funções para interagir com a distribuição F são `df()`, `pf()`, `qf()`, `rf()`. A função `df()` fornece a densidade, a função `pf()` fornece a função de distribuição, a função `qf()` fornece a função quantil e a função `rf()` gera valores de densidade aleatórios.

Usa-se `df()` para calcular a densidade no valor de 1 de uma curva F com `gl1=10` e `gl2=20`:

```
df(1, df1 = 10, df2 = 20)
```

```
## [1] 0.7143568
```

Ou seja, ao se observar a curva acima da cor verde, quando $x = 1$, $y = 0,7$, de densidade de probabilidade.

Usa-se `pf()` para calcular a área sob a curva para o intervalo $[0,1.5]$ e o intervalo $[1.5, +\infty]$ de uma curva F com `gl1=10` e `gl2=20`. Além disso, pode-se perguntar ao R se a soma dos intervalos $[0,1.5]$ e $[1.5, +\infty]$ é igual a 1.

Distribuição F

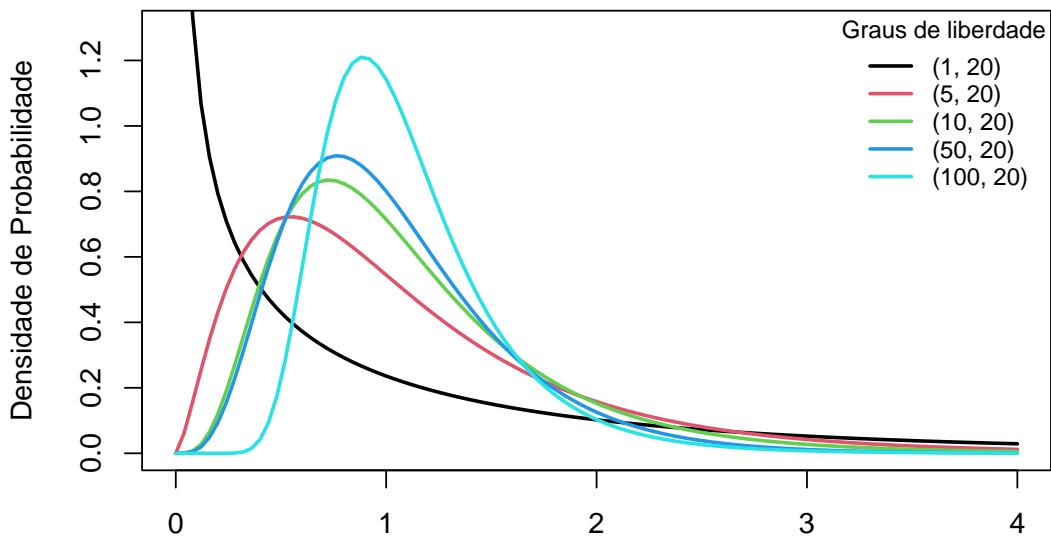


Figura 143: Distribuições F.

```
x = 1.5
g11 = 10
g12 = 20
# Interval$(0, 1.5)
p1 <- pf(x, df = g11, df2 = g12, lower.tail = TRUE)
p1
```

```
## [1] 0.7890535
```

```
# intervalo$[1.5, +inf)
p2 <- pf(x, df = g11, df2 = g12, lower.tail = FALSE)
p2
```

```
## [1] 0.2109465
```

```
p1 + p2 == 1
```

```
## [1] TRUE
```

Usa-se o `qf()` para calcular o quantil para uma determinada área (= probabilidade) sob a curva para uma curva F com $g11=10$ e $g12=20$ que corresponde a $q=0,5$. Defini-se `lower.tail = TRUE` para obter a área para o intervalo $[0,q]$.

```

q <- 0.50
gl1=10
gl2=20
Fc <- round(qf(q, df1 = gl1, df2 = gl2, lower.tail = TRUE), 2)
Fc
## [1] 0.97

```

Observando a Figura 144, construído com função `shadeDist()` do pacote `fastGraph` (111) , verifica-se que a área sob a curva abaixo de 0,97 é igual a 50%. Consulte a ajuda para maiores detalhes dos argumentos da função.

```

shadeDist (xshade = Fc,
           ddist = "df",
           parm1 = gl1,
           parm2 = gl2,
           lower.tail = TRUE,
           digits.prob = 2,
           digits.xtic = 2,
           col=c("gray1","steelblue"))

```

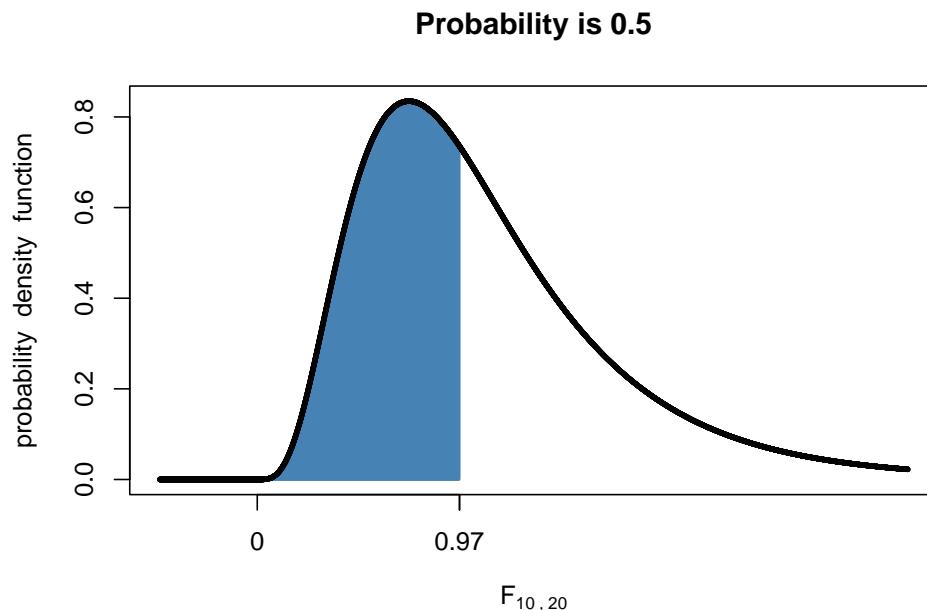


Figura 144: Distribuição F (10,20) = 0.97.

Usar-se-á a função `rf()` para gerar 100.000 valores aleatórios da distribuição F com $gl1=10$ e $gl2=20$. Em seguida, plota-se um histograma (Figura 145) e compara-se com a função de densidade de probabilidade da distribuição F com $gl1=10$ e $gl2=20$ (linha vermelha).

```

x <- rf(100000, df1 = 10, df2 = 20)
hist(x,

```

```

breaks = 'Scott',
freq = FALSE,
xlim = c(0,3),
ylim = c(0,1),
ylab = "Densidade",
xlab = '',
main = 'Histograma para uma distribuição F(10,20)',
cex.main=0.9)

curve(df(x, df1 = 10, df2 = 20), from = 0, to = 4, n = 5000, col= 'red', lwd=2, add = T)

```

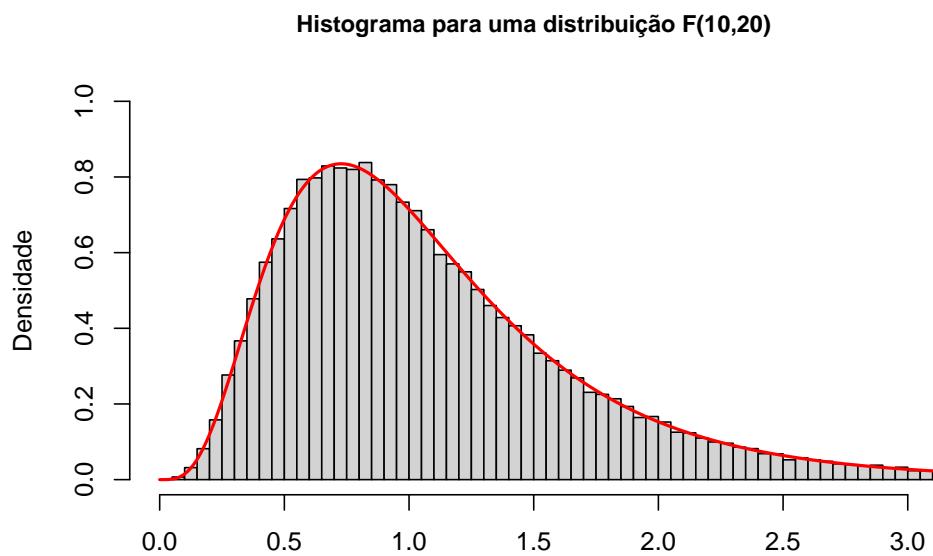


Figura 145: Histograma de uma distribuição F (10,20)

13.2.3 Dados do exemplo

Para testar a hipótese de que a intensidade do tabagismo materno tem efeito sobre o peso do recém-nascido, foram selecionados aleatoriamente 200 recém-nascidos classificados em quatro grupos de $n = 50$ cada grupo, conforme a quantidade de cigarros fumados por dia por suas mães. Estes dados estão no arquivo `dadosFumo.xlsx`.

- **Grupo 1:** recém-nascidos de mães não fumantes;
- **Grupo 2:** recém-nascidos de mães que fumavam até 10 cigarros/dia – categorizado como tabagismo leve;
- **Grupo 3:** recém-nascidos de mães que fumavam de 11 a 19 cigarros/dia – categorizado como tabagismo moderado;
- **Grupo 4:** recém-nascidos de mães que fumavam ≥ 20 cigarros por dia – categorizado como tabagismo pesado.

Para baixar o banco de dados, clique [aqui](#). Salve o mesmo no seu diretório de trabalho.

13.2.3.1 Leitura dos dados A leitura será feita com a função `read_excel()` do pacote `readxl` e serão atribuídos a um objeto de nome `dados` e verificada a sua estrutura com a função `head()`.

```
dados <- readxl::read_excel("dadosFumo.xlsx")

head (dados)

## # A tibble: 6 x 3
##       id pesoRN fumo
##   <dbl>  <dbl> <dbl>
## 1     1    3458.    1
## 2     2    2723.    1
## 3     3    4125.    1
## 4     4    2905.    1
## 5     5    3608.    1
## 6     6    3383.    1
```

13.2.3.2 Exploração e resumo dos dados Como a variável `fumo` encontra-se como uma variável numérica, será transformada em fator que é a sua verdadeira classe com 4 níveis.

```
dados$fumo <- factor (dados$fumo,
                      ordered = TRUE,
                      levels = c(1, 2, 3, 4),
                      labels = c ("nao",
                                 "leve",
                                 "moderado",
                                 "pesado"))

class (dados$fumo)

## [1] "ordered" "factor"
```

As medidas resumidoras serão obtidas, usando as funções `group_by ()` e `summarise ()` do pacote `dplyr`.

```
alpha = 0.05
resumo <- dados %>%
  group_by(fumo) %>%
  dplyr::summarise(n = n(),
                   media = mean(pesoRN, na.rm = TRUE),
                   dp = sd (pesoRN, na.rm = TRUE),
                   ep = dp/sqrt(n),
                   me = qt ((1-alpha/2), n-1)*ep,
                   IC_Inf = media - me,
                   IC_sup = media + me)
resumo

## # A tibble: 4 x 8
##   fumo      n media    dp    ep    me IC_Inf IC_sup
##   <ord>    <int> <dbl> <dbl> <dbl> <dbl>  <dbl>
## 1 nao      50 3395.  405.  57.3  115.  3280.  3510.
## 2 leve     50 3102.  431.  60.9  122.  2980.  3225.
## 3 moderado 50 3151.  495.  70.0  141.  3011.  3292.
## 4 pesado   50 2954.  443.  62.6  126.  2828.  3080.
```

13.2.3.3 Visualização gráfica dos dados Os boxplots (Figura 146) são uma maneira interessante de visualizar os dados:

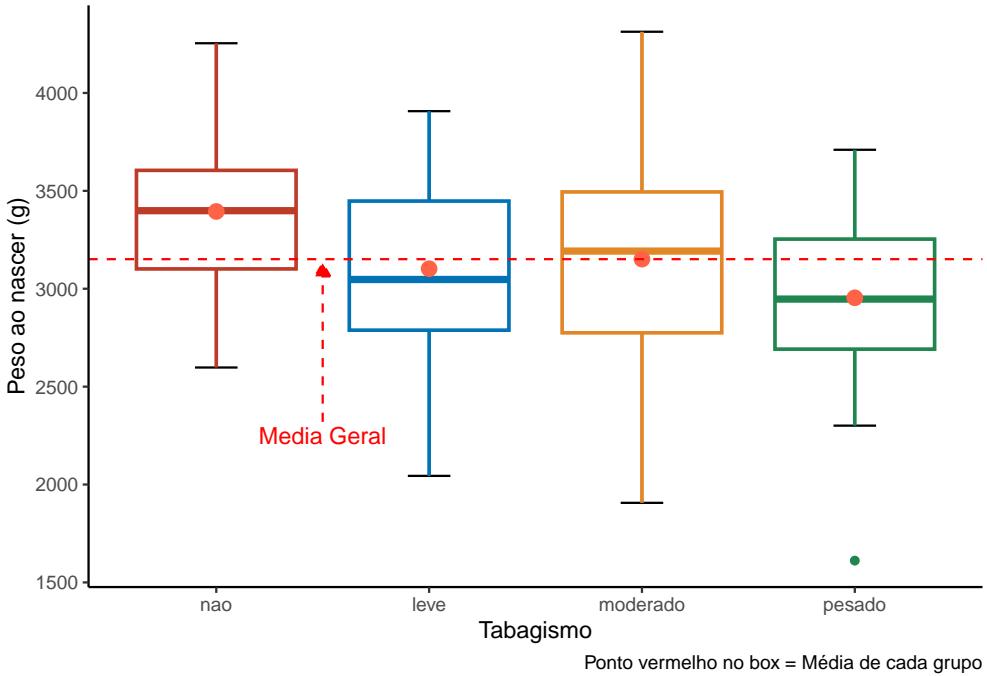


Figura 146: Boxplots do impacto do tabagismo materno no peso ao nascer

Observa-se que há uma tendência de o peso ao nascer diminuir à medida que quantidade de cigarros fumados aumenta. Entretanto, esta diferença pode ser pelo acaso.

13.2.4 Definição das hipóteses estatísticas

Para testar a igualdade entre as médias, será usado um teste bicaudal:

$$H_0 : \mu_1 = \mu_2 = \mu_3 = \mu_4$$

Contra a hipótese alternativa, H_A , de que, pelo menos, uma das médias é diferente das demais.

13.2.5 Definição da regra de decisão

O nível significância, α , geralmente escolhido é igual a 0,05. A distribuição da estatística do teste, sob a H_0 , é a distribuição F . O número de graus de liberdade total ($n-1$) é dividido em dois componentes:

- Grau de liberdade do numerador (ENTRE) é dado por $gl_E = k - 1$, onde k é o número de grupos.
- Grau de liberdade do denominador (DENTRO ou residual) é dado por $gl_D = n - k$, onde, $n = \sum n_i$.

O teste ANOVA de uma via é sempre unilateral à direita com a região de rejeição na cauda direita da curva de distribuição F . No exemplo, para um $\alpha = 0,05$

```

alpha <- 0.05
k <- length(resumo$media)
n <- nrow(dados)
g1E <- k - 1
g1E

```

```
## [1] 3
```

```

g1D <- n - k
g1D

```

```
## [1] 196
```

Com esses dados, usando a função `qf()` calcula-se o valor crítico de F (Figura 147) que é igual:

```

Fc <- qf(1 - alpha, g1E, g1D)
round(Fc, 2)

```

```
## [1] 2.65
```

Portanto, se

$$|F_{calculado}| < |F_{crtico}| \rightarrow \text{no se rejeita } H_0$$

$$|F_{calculado}| \geq |F_{crtico}| \rightarrow \text{rejeita - se } H_0$$

13.2.6 Teste Estatístico

A estatística de teste é obtida calculando duas estimativas da variância populacional, σ^2 : a *variância entre os grupos* (s_E^2) e a *variância dentro dos grupos* (s_D^2).

A variância entre os grupos também é chamada de *quadrado médio entre os grupos* (QM_E) e é igual a soma dos quadrados entre (SQ_E) ou do fator dividida pelos graus de liberdade entre:

$$QM_E = \frac{SQ_E}{gl_E}$$

A variância dentro dos grupos é também denominada de *quadrado médio dentro dos grupos* ou residual (QM_D) e é igual a soma dos quadrados dentro dividida pelos graus de liberdade dentro:

$$QM_D = \frac{SQ_D}{gl_D}$$

A variância entre os grupos, QM_E , dá uma estimativa de σ^2 com base na variação entre as médias das amostras extraídas de diferentes populações. Para o exemplo das quatro categorias de tabagismo durante a gestação, o QM_E será baseado nos valores das médias dos pesos dos recém-nascidos nos quatro grupos diferentes. Se as médias de todas as populações em consideração forem iguais, as médias das respectivas amostras ainda serão diferentes, mas a variação entre elas deverá ser pequena e, consequentemente, espera-se que o valor do QM_E seja pequeno. No entanto, se as médias das populações consideradas não são todas

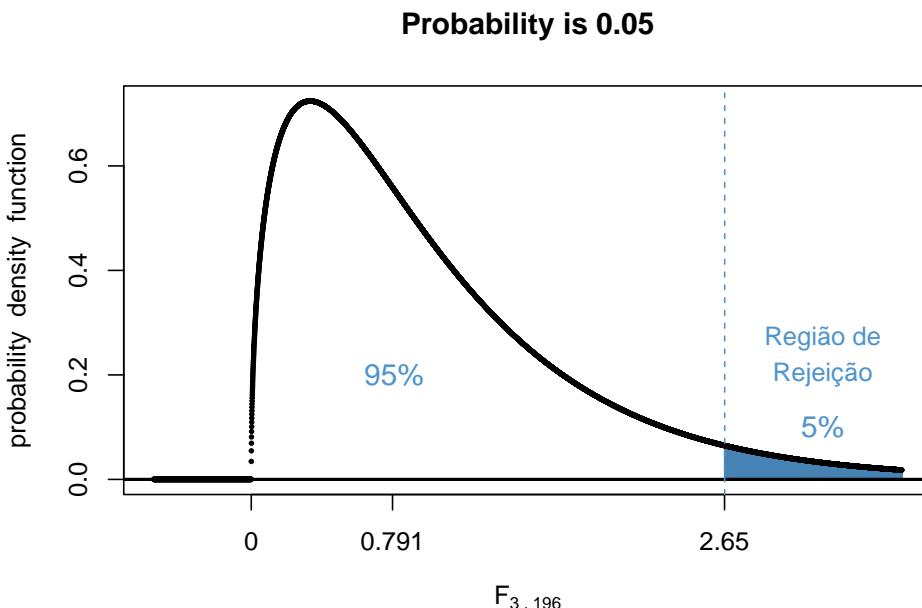


Figura 147: Curva da Distribuição F $3,196 = 2,65$

iguais, espera-se que a variação entre as médias das respectivas amostras seja grande e, consequentemente, o valor de QM_E seja grande.

A variância dentro das amostras, QM_D , dá uma estimativa de σ^2 com base na variação dos dados de diferentes amostras. Para o exemplo das quatro categorias de tabagismo durante a gestação, o QM_D será baseado nas médias individuais dos pesos dos recém-nascidos incluídos nas quatro amostras retiradas de quatro populações. O conceito de QM_D é semelhante ao conceito de desvio padrão conjugado ou agrupado, s_o , para duas amostras.

A estatística de teste é simplesmente, a razão das variâncias entre e dentro do grupo. Dessa maneira,

$$F = \frac{s_E^2}{s_D^2} = \frac{\frac{SQ_E}{gl_E}}{\frac{SQ_D}{gl_D}} = \frac{QM_E}{QM_D}$$

13.2.6.1 Pressupostos do teste Ao realizar um teste de ANOVA de um fator deve-se assumir que:

1. As populações das quais as amostras são retiradas são normalmente distribuídas;
2. As populações das quais as amostras são retiradas têm a mesma variância (homocedasticidade);
3. Amostras aleatórias e independentes;
4. Todos os grupos devem ter tamanho amostral adequado. Grupos com menos de 10 participantes são problemáticos por reduzirem a precisão da média. Na prática, deve-se evitar menos de 30 participantes. A relação entre os grupos não deve ser maior do que 1:4 (112);
5. Não devem existir valores atípicos (*outliers*);
6. A mensuração dos dados deve ser em nível intervalar ou de razão.

Portanto, antes iniciar com o teste de hipótese, verifica-se se as suposições mencionadas para o teste de hipótese ANOVA unidirecional foram atendidas. As amostras são amostras aleatórias e independentes. Isto já é um bom começo!

Avaliação da normalidade

Verifica-se a premissa de normalidade, usando o teste de Shapiro-Wilk para os múltiplos grupos e desenhandando um gráfico de probabilidade normal (gráficos Q-Q) para cada grupo.

```
dados %>%
  dplyr::group_by(fumo) %>%
  shapiro_test(pesoRN)
```

```
## # A tibble: 4 x 4
##   fumo     variable statistic      p
##   <ord>    <chr>        <dbl> <dbl>
## 1 nao      pesoRN       0.976 0.385
## 2 leve     pesoRN       0.979 0.499
## 3 moderado pesoRN       0.985 0.776
## 4 pesado   pesoRN       0.971 0.257
```

Para o gráfico Q-Q (Figura 148) pode ser usado a função `ggqqplot()` do pacote `ggpubr` que produz um gráfico QQ normal com uma linha de referência, acompanhada de área sombreada, correspondente ao IC95%.

```
ggqqplot(dados, x="pesoRN", facet.by = "fumo") +
  labs(y = "Peso ao nascer (g) (m)",
       x = "Quantis teóricos")
```

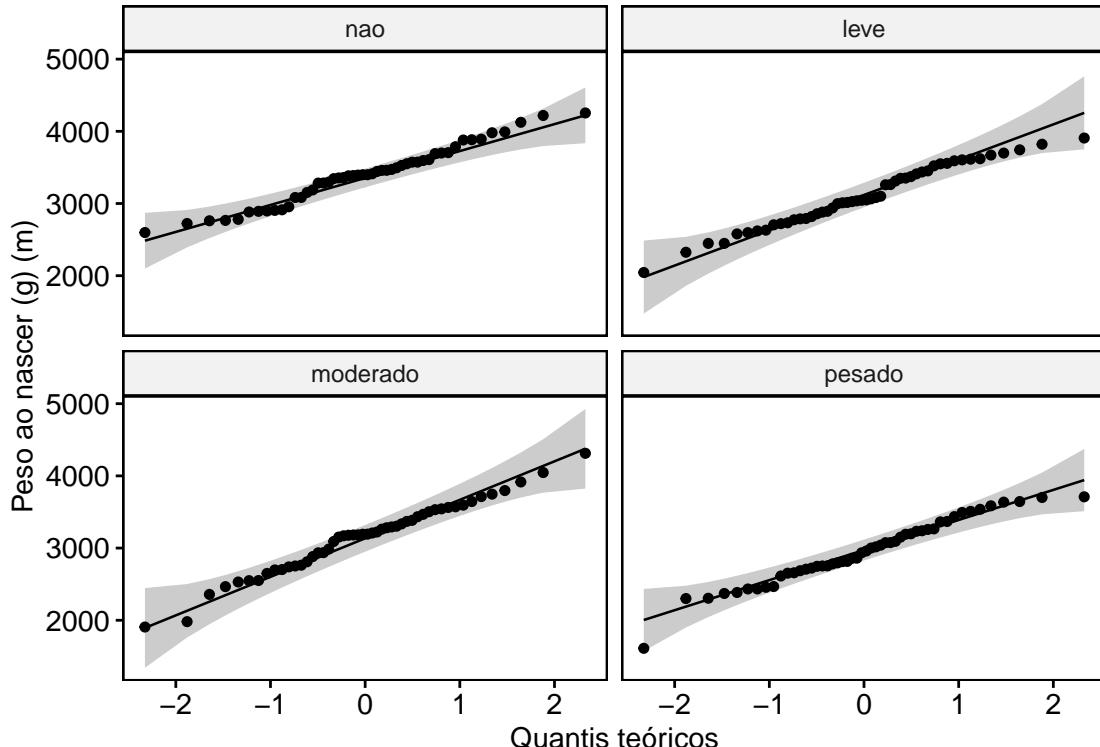


Figura 148: Gráficos Q-Q

O resultado do teste de Shapiro-Wilk entregou todos os resultados com valor P acima de 0.05 e os gráficos Q-Q, não são perfeitos, mas pode-se assumir que os dados para cada grupo caem aproximadamente em uma linha reta.

Avaliação da homogeneidade das variâncias

Em seguida, testa-se a suposição de que as variâncias são iguais, usando o Teste de Levene através da função `leveneTest ()` do pacote ‘car’.

```
leveneTest(pesoRN~fumo, center = mean, data = dados)
```

```
## Levene's Test for Homogeneity of Variance (center = mean)
##          Df F value Pr(>F)
## group     3  0.6306 0.5961
##          196
```

Verificação da presença de outliers

Pode-se aqui, além de verificar nos boxplots, usar a função `by_group()` do pacote `dplyr` junto com a função `identify_outliers()` do pacote `rstatix`:

```
dados %>%
  group_by(fumo) %>%
  identify_outliers(pesoRN)
```

```
## # A tibble: 1 x 5
##   fumo      id pesoRN is.outlier is.extreme
##   <ord>    <dbl> <dbl> <lgl>     <lgl>
## 1 pesado    168  1611. TRUE      FALSE
```

Como mostrado nos boxplots, existe um valor atípico, ou seja, está abaixo de 1,5 IIQ. Entretanto, ele não é extremo ($> 3 \text{ IIQ}$).

Da mesma maneira que no teste t , os pressupostos têm mais importância em grupos pequenos e desiguais. Para o exemplo em análise, os pressupostos foram verificados e pode-se assumir que os grupos são independentes e as médias têm distribuição normal e existe homocedasticidade, além disso, os grupos têm o mesmo tamanho ($n = 50$). Portanto, a análise pode ser continuada.

O que fazer se os pressupostos são violados?

Se a homogeneidade da variância é o problema, um teste possível de ser implementado no R é o *F de Welch*, aplicando a função `welch.test()`, incluída no pacote `onewaytests` (113). Existem também testes não paramétricos, como o *Teste de Kruskal-Wallis*, que será visto mais adiante.

13.2.6.2 Execução do teste estatístico É perfeitamente possível realizar um teste de hipótese ANOVA unidirecional no R manualmente. Entretanto, é uma proeza cansativa! Incrível, Usando o R para obter o mesmo resultado faz-se o processo em apenas uma linha de código!

Para realizar um teste de hipótese ANOVA unidirecional no R, aplica-se a função `aov()` do R base. Esta função espera a chamada notação de fórmula, portanto, os dados são incluídos separando as duas variáveis de interesse por `~` (til). Além disso, os dados no qual as variáveis especificadas na fórmula são encontradas. Além da fórmula e dos dados, a função `aov()` tem outros argumentos:

- *effect.size* → tamanho do efeito a ser calculado e mostrado nos resultados da ANOVA. Os valores permitidos podem ser “ges” (eta ao quadrado) ou “pes” (eta parcial ao quadrado) ou ambos. O padrão é “ges”;
- *contrasts* → uma lista de contrastes a ser usada para alguns dos fatores da fórmula

```

modelo.aov <- aov(pesoRN ~ fumo, dados)

sumario <- summary(modelo.aov)
sumario

##           Df   Sum Sq Mean Sq F value    Pr(>F)
## fumo       3 5030606 1676869   8.482 2.52e-05 ***
## Residuals 196 38748837 197698
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

A saída é bem reduzida, relatando as informações específicas da *Tabela da ANOVA*, a estatística F junto com o valor P e os graus de liberdade, soma dos quadrados (*Sum Sq*) e quadrados médios (*Mean Sq*), que com frequência se necessita para o para o relatório do modelo.

A variância entre os grupos também é chamada de **quadrado médio entre os grupos** e é igual a soma dos quadrados entre ou do fator dividida pelos graus de liberdade entre. A variância dentro dos grupos é também denominada de **quadrado médio dentro dos grupos ou residual** e é igual a soma dos quadrados dentro dividida pelos graus de liberdade dentro.

A ANOVA detectou um efeito significativo do fator, que neste caso é o **fumo**, o valor $P < 0,0001$.

Pode-se simplesmente relatar isso e encerrar, mas é provável que se queira saber quais grupos diferem uns dos outros. Lembre-se de que não se pode apenas inferir isso a partir de uma visão dos dados, mas felizmente existem testes estatísticos para ajudar a entender as diferenças dos grupos.

13.2.7 Testes *post-hoc*

Os testes de comparações múltiplas constituem-se em uma análise após a realização da ANOVA. Se houve uma diferença, indicada pela ANOVA, os testes de comparações múltiplas ou também conhecidos como *teste post hoc*, ajudam a quantificar as diferenças entre os grupos para determinar quais grupos diferem significativamente uns dos outros.

Aqui será usado o *HSD de Tukey*, que é conservador. *HSD* vem da expressão em inglês - *Honest Significant Difference*. Este teste requer um objeto `aov` no qual executa seu procedimento, que chamaremos de `mc`. O procedimento de Tukey HSD executará uma comparação de pares de todas as combinações possíveis dos grupos e testará esses pares para diferenças significativas entre suas médias, tudo enquanto ajusta o valor P a um limite superior de significância para compensar o fato de que muitos testes estatísticos estão sendo realizados e a probabilidade de um falso positivo aumenta com o aumento do número de testes. A função a ser usada é a `tukey_hsd()`, do pacote `rstatix`.

```

pwc <- tukey_hsd (modelo.aov)
pwc

## # A tibble: 6 x 9
##   term group1  group2  null.value estimate conf.low conf.high   p.adj p.adj~1
## * <chr> <chr>     <chr>        <dbl>    <dbl>    <dbl>    <dbl> <dbl> <chr>
## 1 fumo  nao      leve         0     -292.    -523.    -61.8 6.54e-3 **
## 2 fumo  nao      moderado    0     -243.    -474.    -12.8 3.41e-2 *
## 3 fumo  nao      pesado     0     -441.    -671.    -210. 9.15e-6 ****
## 4 fumo  leve     moderado    0      49.0    -181.    279. 9.46e-1 ns
## 5 fumo  leve     pesado     0     -149.    -379.    81.9 3.42e-1 ns
## 6 fumo  moderado pesado     0     -198.    -428.    32.8 1.21e-1 ns
## # ... with abbreviated variable name 1: p.adj.signif

```

Com base nos valores $P < 0,05$ tem-se três combinações de grupos que diferem: leve–não, moderado–não e pesado–não. Isto mostra que o grupo que difere é o das mães não fumantes.

Pode-se visualizar isso na Figura 149 obtida com a função `plot()`, usando os resultados da função `TukeyHSD()` disponível no **R** base. Esta função gera o teste de Tukey com as diferença entre os pares e os intervalos de confiança que permitem a construção do gráfico. A função `par()` é empregada para adaptar as margens da figura ao tamanho da mesma e depois é usada novamente para retornar ao padrão `par(mar=c(5.1, 4.1, 4.1, 2.1))`. O argumento `mar` é um vetor numérico que define os tamanhos das margens na seguinte ordem: inferior, esquerda, superior e direita.

```
par(mar=c(3,8,3,3))
plot(TukeyHSD(modelo.aov, conf.level = 0.95), las = 1)
```

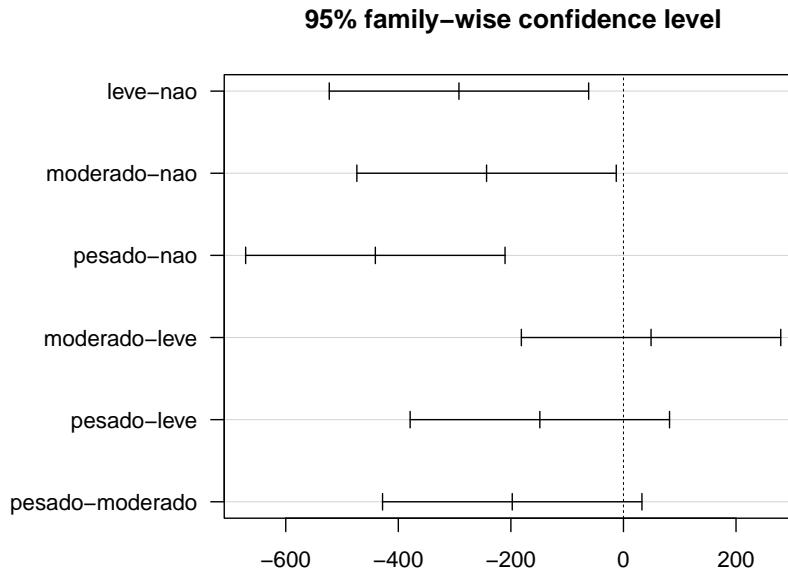


Figura 149: Gráficos do Teste de Tukey

```
par(mar=c(5.1, 4.1, 4.1, 2.1))
```

13.2.8 Tamanho do efeito

Uma das medidas de tamanho de efeito mais comumente relatadas para a ANOVA é o **eta ao quadrado** (η^2), que é um índice da força da associação entre um fator e uma variável dependente. Eta ao quadrado é a proporção da variação total atribuível ao fator. É calculado como a razão da variância do fator para a variância total e os valores variam de 0 a 1.

Esta medida pode ser obtida com o pacote `effectsize` (114), usando a função `eta_squared()` com um objeto da classe tipo `modelo.aov`.

```
effectsize::eta_squared (modelo.aov, partial = FALSE)
```

```
## # Effect Size for ANOVA (Type I)
##
## Parameter | Eta2 |      95% CI
```

```

## -----
## fumo      | 0.11 | [0.05, 1.00]
##
## - One-sided CIs: upper bound fixed at [1.00].

```

O *eta quadrado* é uma estimativa tendenciosa da força da associação, na medida em que superestima os efeitos, especialmente para amostras pequenas. Uma outra medida do tamanho do efeito menos tendenciosa é o ômega ao quadrado (ω^2). O ômega ao quadrado é uma medida corrigida, menos enviesada e menos inflacionada. Ela pode ser calculada com a função `omega_squared()`, também do pacote `effectsize`:

```
effectsize::omega_squared (modelo.aov, partial = FALSE)
```

```

## # Effect Size for ANOVA (Type I)
## 
## Parameter | Omega2 |      95% CI
## -----
## fumo      | 0.10 | [0.04, 1.00]
##
## - One-sided CIs: upper bound fixed at [1.00].

```

Apesar de ser controverso, pode-se seguir a orientação da Tabela 13, para a interpretação (115):

Tabela 13: Interpretação do Tamanho do Efeito

Resultado	Effectsize
0,01	pequeno
0,06	médio
0,14	grande

13.2.9 Conclusão

O peso dos recém-nascidos foi estatisticamente diferente entre os diferentes grupos, $F(3, 196) = 8,48$, $P = 0.0000252$, $\eta^2 = 0,11$.

As análises *post-hoc* de Tukey revelaram que o peso dos recém-nascidos a termo no grupo das gestantes não fumantes apresentou uma diferença estatisticamente significativa do grupo de tabagismo leve (-292 g, IC95%: -523 a -62 g; $P = 0,0065$); do grupo de tabagismo moderado (-243 g, IC95%: -474 a -13 g; $P = 0,0341$) e do grupo de tabagismo pesado (-441 g, IC95%: -671 a -210 g; $P < 0,0001$), mas entre os grupos de fumantes não houve diferença estatisticamente significativa.

13.2.9.1 Apresentação dos resultados Serão apresentados boxplots (Figura 150), com `ggboxplot ()`, do pacote `ggpubr`, utilizando, para cores, a `pallete = "jama"`, do pacote `ggsci`. Para adicionar teste estatístico, usou-se a função `get_test_label()` e para o teste *post hoc*, a função `get_pwc_label()`, ambas do pacote `rstatix`.

```

modelo.aov <- aov(pesoRN ~ fumo,
                    dados)

tab.aov <- anova_test(dados,
                      pesoRN ~ fumo,

```

```

type = 2)

pwc <- tukey_hsd(dados,pesoRN~fumo)

pwc <- pwc %>% add_xy_position (x = "fumo")
ggboxplot (dados,
  x = "fumo",
  y = "pesoRN",
  xlab = "Tabagismo na gestação",
  ylab = "Peso do recém-nascido (g)",
  color = "fumo",
  palette = "jama",
  fill = "fumo",
  alpha = 0.3,
  size = 1.0) +
  stat_pvalue_manual (pwc,
    hide.ns = TRUE) +
  labs (subtitle = get_test_label (tab.aov, detailed = TRUE),
    caption = get_pwc_label(pwc)) +
  theme(legend.position = "none") +
  theme (text = element_text (size = 12))

```

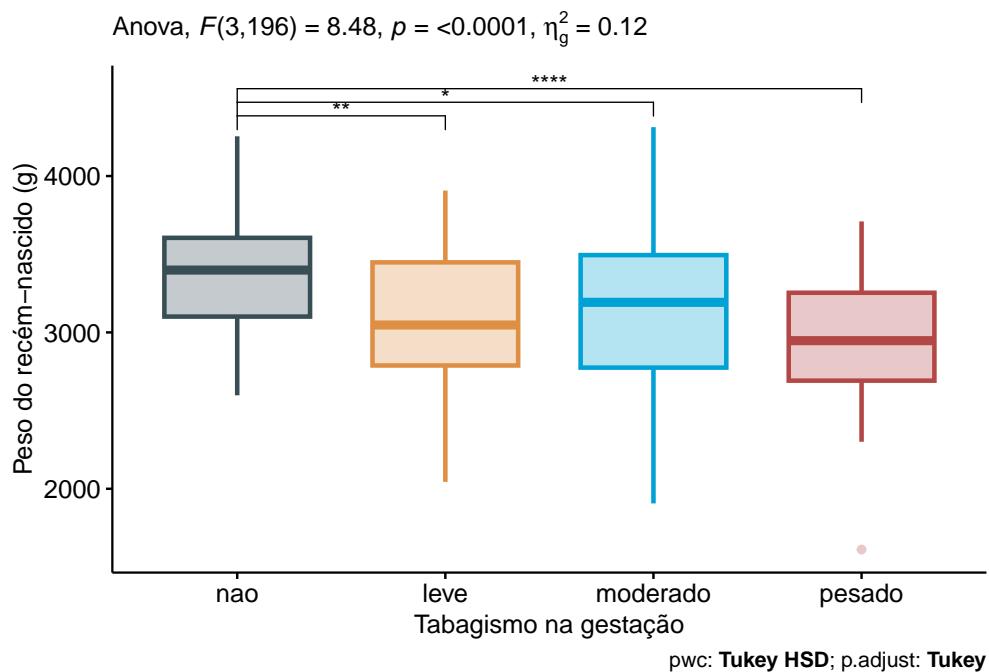


Figura 150: Efeito do tabagismo na gestação sobre o peso do recém-nascido.

13.3 ANOVA de dois fatores

A *ANOVA de dois fatores* é uma extensão da ANOVA de um fator. Neste tipo de ANOVA, ao invés de observar o efeito de um fator sobre a variável desfecho contínua, é analisado simultaneamente o efeito de duas variáveis de agrupamento. Outros sinônimos para a ANOVA de dois fatores são: *ANOVA factorial* ou

ANOVA de duas vias.

Quando se tem dois ou mais fatores, além de observar o efeito desses fatores sobre a variável desfecho, há necessidade de verificar se eles não interagem entre si. Portanto, é um objetivo importante da ANOVA factorial avaliar se há um efeito de *interação* estatisticamente significativo entre os fatores.

13.3.1 Dados do exemplo

O conjunto de dados `dadosMemoria.xlsx` que contém informações de um teste de memória realizado em homens e mulheres, após o consumo de álcool, categorizado em três grupos (nenhum, 3 latas e 6 latas de cerveja tipo *pilsen* com 4,5% de álcool). O grupo sem consumo de álcool (cerveja sem álcool) serve como controle. Após o consumo de álcool, foi avaliada a memória para a realização de uma tarefa cognitiva.

Neste exemplo, modificado de Andy Field (116), o efeito do álcool sobre a memória do indivíduo é a variável focal, a principal preocupação. Acredita-se que o efeito de álcool depende de outro fator, sexo, que são chamados de variáveis moderadoras.

Para baixar o banco de dados, clique [aqui](#). Salve o mesmo no seu diretório de trabalho.

13.3.1.1 Leitura dos dados A leitura será feita com a função `read_excel()` do pacote `readxl` e serão atribuídos a um objeto de nome `dados` e verificada a sua estrutura com a função `head()`.

```
dados <- readxl::read_excel("dadosMemoria.xlsx")  
head(dados)
```

```
## # A tibble: 6 x 3  
##   sexo     alcool  escore  
##   <chr>    <chr>    <dbl>  
## 1 Feminino nenhum      65  
## 2 Feminino nenhum      70  
## 3 Feminino nenhum      60  
## 4 Feminino nenhum      60  
## 5 Feminino nenhum      60  
## 6 Feminino nenhum      55
```

13.3.1.2 Exploração e sumarização dos dados Observando o resultado da função `head()`, verifica-se que as variáveis `alcool` e `sexo` estão como `<chr>` e ideal é que estejam como fatores. Portanto, vamos Colocar as categorias do consumo de álcool como fator e em uma ordem lógica (nenhum consumo, três latas e 6 latas). A variável `sexo` será apenas colocada como fator porque não tem uma ordem lógica. As demais variáveis, `id` (identificação) e `escore`(escore de memória) podem permanecer com `dbl` (numérica).

```
dados$alcool <- factor(dados$alcool,  
                        levels = c("nenhum",  
                                  "3 latas",  
                                  "6 latas"))  
dados$sexo <- as.factor(dados$sexo)
```

A sumarização dos dados será feita com as funções `group_by()` e `summarise()` do pacote `dplyr` para a variável `escore` por grupos, `sexo` e `alcool`.

```

alpha <- 0.05
resumo <- dados %>%
  dplyr::group_by(sexo, alcool) %>%
  dplyr::summarise(n = n(),
    media = mean(escore, na.rm=TRUE),
    dp = sd(escore, na.rm=TRUE),
    ep = dp/sqrt(n),
    me = qt((1 - alpha/2),n-1)*ep,
    linf = media - me,
    lsup = media + me)
resumo

## # A tibble: 6 x 9
## # Groups:   sexo [2]
##   sexo     alcool     n  media    dp    ep    me  llinf  lsup
##   <fct>    <fct>  <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 Feminino  nenhum     8  60.6  4.96  1.75  4.14  56.5  64.8
## 2 Feminino  3 latas    8  62.5  6.55  2.31  5.47  57.0  68.0
## 3 Feminino  6 latas    8  57.5  7.07  2.5   5.91  51.6  63.4
## 4 Masculino nenhum     8  66.9 10.3   3.65  8.64  58.2  75.5
## 5 Masculino 3 latas    8  66.9 12.5   4.43 10.5   56.4  77.3
## 6 Masculino 6 latas    8  35.6 10.8   3.83  9.06  26.6  44.7

```

Os dados estão estruturados com um desenho onde as células tem um formato 2 x 3 com os fatores `sexo` e `alcool` e 8 indivíduos em cada célula. O fator `sexo` tem dois níveis (feminino e masculino) e o fator `alcool` tem três níveis (nenhum, 3 latas e 6 latas). Observe que o desenho é *balanceado*, pois todas as células têm o mesmo número de indivíduos. Esta estrutura é o caso mais simples; desenhos não balanceados são mais complexos.

13.3.1.3 Visualização gráfica dos dados Para visualizar os dados, será construído um gráfico com boxplots (Figura 151), usando o pacote `ggpubr`(117), com a função `ggboxplot()`, que fornece algumas funções fáceis de usar para criar e personalizar gráficos prontos para publicação baseados em ‘`ggplot2`’. O boxplot irá plotar os dados agrupados pelas combinações dos níveis dos dois fatores.

```

ggboxplot (dados,
  bxp.errorbar = TRUE,
  bxp.errorbar.width = 0.2,
  x = "alcool",
  y = "escore",
  color = "black",
  fill = "sexo",
  palette = "bmj",
  ylab = "Escore da Memória",
  xlab = "",
  legend.title = "Sexo",
  legend = "top") +
  theme (text = element_text (size = 12))

```

Além dos boxplot, é interessante desenhar um gráfico de linhas (Figura 152) que plota a média (ou outro resumo) da variável escore (resposta) para combinações bidimensionais de fatores, ilustrando assim possíveis interações. Aqui, pode-se usar a função `gglime()`, também pertencente ao interessante pacote `ggpubr`.

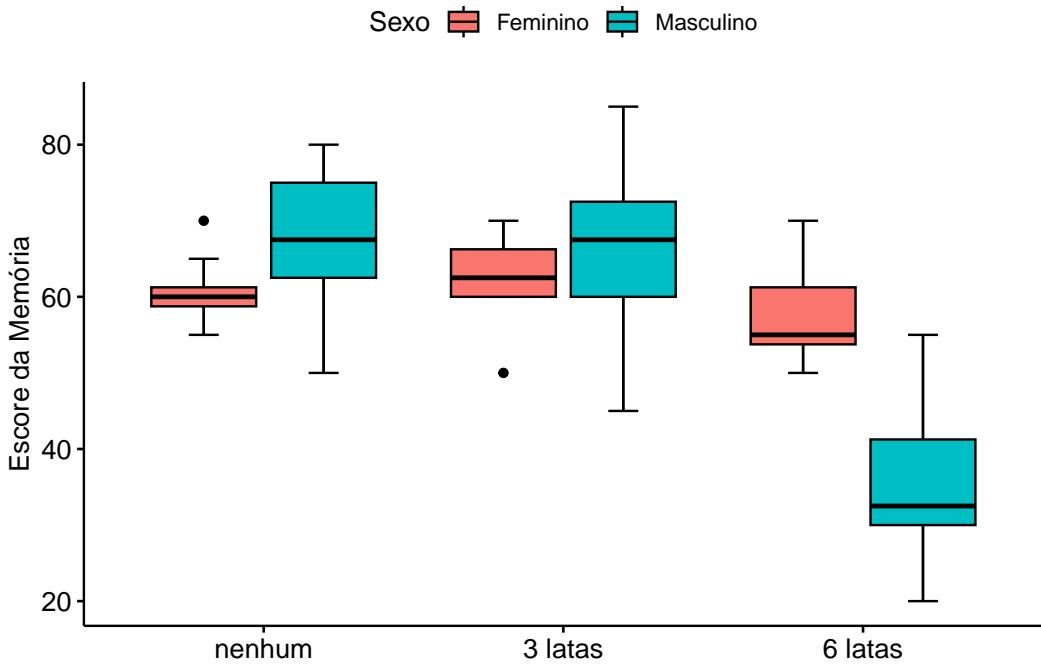


Figura 151: Efeito do álcool na memória de acordo com o sexo.

```
ggline(dados,
       x = "alcool",
       y = "escore",
       color = "sexo",
       size = 1,
       add = c("mean_ci"),
       palette = c("red", "dodgerblue4"))
```

Os gráficos sugerem um possível efeito do álcool sobre a memória, bem como uma interação entre os sexos.

13.3.2 Hipóteses estatísticas

Serão testadas três possibilidades de hipótese nula:

1. Não há diferença nas médias do fator `alcool`.
2. Não há diferença nas médias do fator `sexo`.
3. Os fatores `alcool` e `sexo` não interagem de forma alguma.

A essas se contrapõe a hipótese alternativa, H_A , de que pelo menos uma das médias é diferente dentro de cada um dos fatores e que existe interação entre eles.

13.3.3 Pressupostos do modelo

Para usar uma ANOVA de duas vias, os dados devem atender a certos pressupostos. A ANOVA de duas vias faz todas as suposições usuais de um teste paramétrico de diferença:

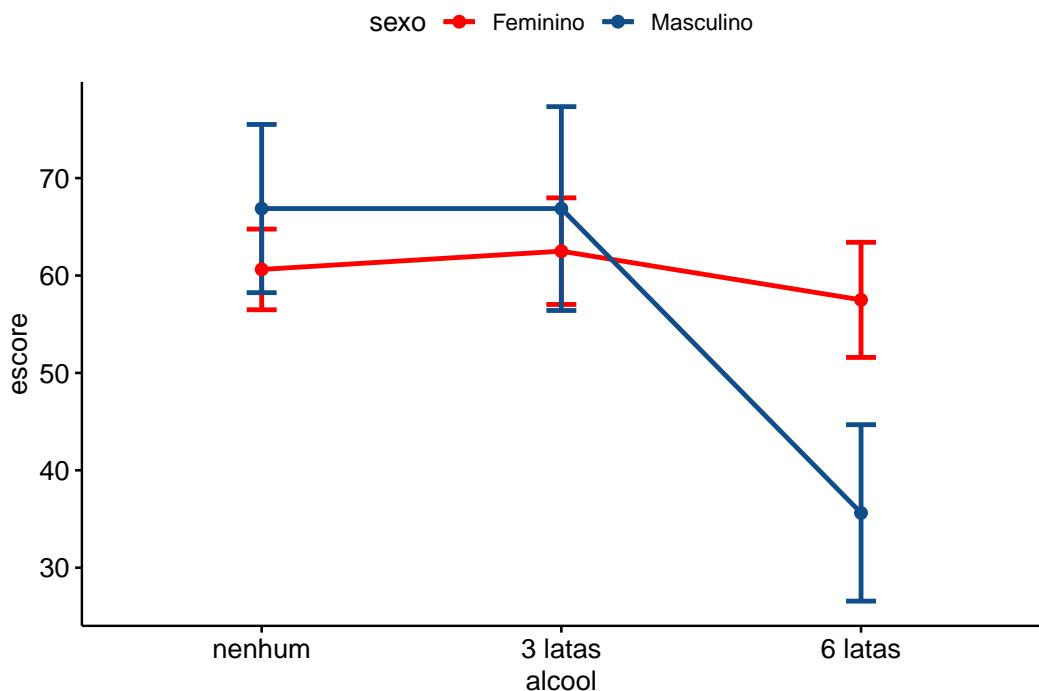


Figura 152: Efeito do álcool na memória de acordo com o sexo.

1. Independência de observações

As variáveis respostas não devem ser dependentes umas das outras (ou seja, uma não deve causar a outra). Isso é impossível de testar com variáveis categóricas - só pode ser garantido por um bom projeto experimental.

Além disso, a variável dependente deve representar observações únicas - não devem ser agrupadas em locais ou indivíduos. Se esta premissa for violada, você pode incluir uma variável de bloqueio e/ou usar uma ANOVA de medidas repetidas.

2. Normalidade

Variável desfecho normalmente distribuída em todos os grupos.

3. Ausência de valores atípicos (*outliers*)

Um valor aberrante ou valor atípico, é uma observação que apresenta um grande afastamento das demais da série, $\pm 1,5$ o intervalo interquartil (IIQ) e extremo se estiver ± 3 IIQ. A existência de outliers implica, tipicamente, em prejuízos à interpretação dos resultados.

4. Homogeneidade de variância (homocedasticidade)

A variação em torno da média para cada grupo sendo comparado deve ser semelhante entre todos os grupos. Se os dados não atenderem a essa suposição, é possível usar uma alternativa não paramétrica, como o teste de Kruskal-Wallis.

13.3.4 Verificação dos pressupostos nos dados brutos

Existe uma discussão se os pressupostos devem ser avaliados nos dados brutos ou apenas nos resíduos. Aqui serão realizadas as duas abordagens que frequentemente resultam no mesmo resultado.

13.3.4.1 Normalidade A variável dependente (`escore`) deve apresentar distribuição aproximadamente normal dentro de cada grupo. Os grupos aqui serão formados pela combinação das duas variáveis independentes (`sexo` e `alcool`). A normalidade será avaliada pelo teste de Shapiro-Wilk, com a função `shapiro_test()` do pacote `rstatix` (106), separando os grupos com a função `group_by()` do pacote `dplyr`, encadeadas com o operador `pipe (%>%)`:

```
dados %>%
  group_by (sexo, alcool) %>%
  shapiro_test (escore)
```

```
## # A tibble: 6 x 5
##   sexo     alcool   variable statistic      p
##   <fct>    <fct>    <chr>     <dbl> <dbl>
## 1 Feminino nenhum escore     0.872 0.156
## 2 Feminino 3 latas escore    0.899 0.283
## 3 Feminino 6 latas escore    0.897 0.273
## 4 Masculino nenhum escore    0.941 0.622
## 5 Masculino 3 latas escore    0.967 0.870
## 6 Masculino 6 latas escore    0.951 0.720
```

Os resultados suportam a conclusão de não rejeição da hipótese nula de que os dados se ajustam a distribuição normal.

13.3.4.2 Pesquisa de valores atípicos A forma mais simples de verificar a presença de um valor atípico é observar o boxplot, mostrado anteriormente. Se observa a presença de valores atípicos entre as mulheres que não ingeriram álcool e nas que ingeriram 3 latas de cerveja. Agora, para confirmar esse achado, será usado a função `identify_outliers ()`, do pacote `rstatix`:

```
dados %>%
  group_by (sexo, alcool) %>%
  identify_outliers(escore)
```

```
## # A tibble: 2 x 5
##   sexo     alcool   escore is.outlier is.extreme
##   <fct>    <fct>    <dbl> <lgl>      <lgl>
## 1 Feminino nenhum     70 TRUE       TRUE
## 2 Feminino 3 latas    50 TRUE       FALSE
```

A Saída do teste confirma a existência dos dois valores atípicos, sendo um deles extremo, entretanto como estes valores são possíveis e, relativamente, próximos da média do sexo feminino, portanto, causam pouca preocupação, principalmente porque o teste de ANOVA é bastante robusto.

13.3.4.3 Verificação da homogeneidade das variâncias Para verificar a homocedasticidade, como os dados têm distribuição normal, é possível usar o teste de Levene, o `leveneTest()` do pacote `car` (105).

```

leveneTest (escore ~ sexo*alcool,
            data = dados,
            center = mean)

## Levene's Test for Homogeneity of Variance (center = mean)
##          Df F value Pr(>F)
## group    5  1.5268 0.2021
##          42

```

13.3.5 Verificação dos pressupostos nos resíduos

O modelo da ANOVA pode ser considerado como um modelo de regressão. Desta forma, este modelo de regressão vai usar os dados brutos para criar um modelo de previsão para esses dados. Este modelo de regressão não é perfeito, existe uma diferença entre os valores previstos e os valores observados, são os resíduos. Faz sentido, então, preocupar-se com os resíduos quando se analisa fatores tentando explicar uma variável dependente contínua, como na ANOVA, pensando em uma regressão linear simples.

A ANOVA prevê que todos os valores do grupo sejam iguais a média do grupo. Ou seja, um homem que ingere 3 latas de cerveja tem um valor de seu escore de memória igual ao deste grupo. Por este motivo, fazer a análise dos resíduos é praticamente o mesmo que a análise dos valores brutos.

Para analisar os resíduos (diferença entre os valores observados e o previsto pelo modelo), em primeiro lugar se constrói o modelo da ANOVA com efeito da interação, usando a função `lm()` do pacote `stats`, incluído no R base:

```

mod.int.lm <- lm(formula = escore ~ alcool * sexo,
                  data = dados)

```

Ao se executar o comando, tem-se a impressão que nada ocorreu, entretanto foi criado o *modelo da ANOVA* com uma série de variáveis, entre elas os resíduos (`residuals`). Para observar os resíduos, basta digitar:

```
mod.int.lm$residuals
```

```

##      1       2       3       4       5       6       7       8       9      10
##  4.375  9.375 -0.625 -0.625 -0.625 -5.625 -0.625 -5.625  7.500  2.500
##     11      12      13      14      15      16      17      18      19      20
## -2.500  7.500  2.500 -2.500 -2.500 -12.500 -2.500  7.500 12.500 -2.500
##     21      22      23      24      25      26      27      28      29      30
## -2.500  2.500 -7.500 -7.500 -16.875 -11.875 13.125 -1.875  3.125  8.125
##     31      32      33      34      35      36      37      38      39      40
##  8.125 -1.875 -21.875 -6.875 18.125 -1.875  3.125  3.125 13.125 -6.875
##     41      42      43      44      45      46      47      48
## -5.625 -5.625 -5.625 19.375 -0.625 -15.625  9.375  4.375

```

Para obter um resumo estatístico dos resíduos:

```
summary(mod.int.lm$residuals)
```

```

##      Min. 1st Qu. Median   Mean 3rd Qu.   Max.
## -21.875 -5.625 -0.625  0.000  5.156 19.375

```

13.3.5.1 Avaliação da normalidade dos resíduos Uma das suposições de uma ANOVA é que os resíduos são normalmente distribuídos. A normalidade dos resíduos, inicialmente, será verificada, usando o teste de Shapiro-Wilk com a função `shapiro.test()`, também pertencente ao pacote `stats`.

```
shapiro_test (mod.int.lm$residuals)

## # A tibble: 1 x 3
##   variable      statistic p.value
##   <chr>          <dbl>    <dbl>
## 1 mod.int.lm$residuals  0.982   0.664
```

O teste entrega um valor $P > 0.05$, indicando que não é possível rejeitar H_0 de normalidade dos resíduos.

Uma outra maneira comum de verificar essa suposição é criando um *gráfico Q-Q*. Se os resíduos forem normalmente distribuídos, os pontos em um gráfico Q-Q ficarão em uma linha diagonal reta. Este gráfico (Figura 153) pode ser construído com a função `ggqqplot()` do pacote `ggnetwork`.

```
ggqqplot(mod.int.lm$residuals)
```

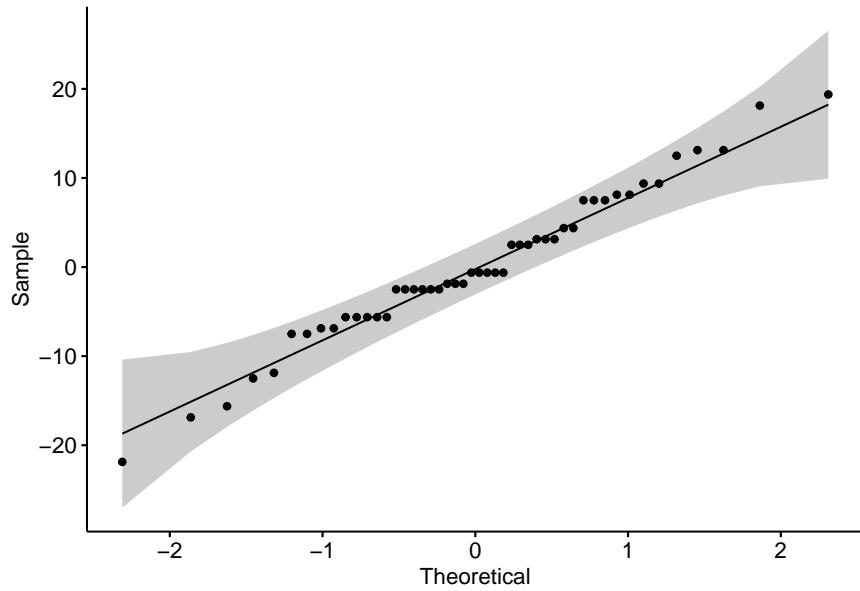


Figura 153: Normalidade dos resíduos - QQ plot.

O gráfico QQ de normalidade, mostra que os resíduos seguem aproximadamente uma linha reta, permitindo assumir a normalidade dos mesmos.

13.3.5.2 Pesquisa de valores atípicos nos resíduos Para a verificação da presença de valores atípicos entre os resíduos, cria-se uma variável que será denominada de `residuos` (observe o banco de dados para ver o acréscimo dessa variável):

```
dados$residuos <- mod.int.lm$residuals
glimpse (dados)
```

```

## Rows: 48
## Columns: 4
## $ sexo      <fct> Feminino, Feminino, Feminino, Feminino, Feminino, Feminino, F-
## $ alcool    <fct> nenhum, nenhum, nenhum, nenhum, nenhum, nenhum, nenhu-
## $ escore    <dbl> 65, 70, 60, 60, 60, 55, 60, 55, 70, 65, 60, 70, 65, 60, 60, 5-
## $ residuos  <dbl> 4.375, 9.375, -0.625, -0.625, -0.625, -5.625, -0.625, -

```

Para identificar os *outliers*, usa-se função `identify_outliers()` do pacote `rstatix`:

```

dados %>% group_by(sexo, alcool) %>
  identify_outliers(residuos)

```

```

## # A tibble: 2 x 6
##   sexo     alcool   escore residuos is.outlier is.extreme
##   <fct>    <fct>    <dbl>   <dbl>   <lgl>     <lgl>
## 1 Feminino nenhum     70     9.38  TRUE      TRUE
## 2 Feminino 3 latas    50    -12.5  TRUE      FALSE

```

Observando os resultados com os dados brutos, verifica-se que eles são iguais aos atuais, confirmando que tanto faz avaliar os dados brutos como os resíduos.

13.3.5.3 Verificação da homogeneidade da variância nos resíduos A verificação da homogeneidade da variância entre os resíduos pode ser feita com o teste de Levene, como feito com os dados brutos.

```

leveneTest (residuos ~ sexo*alcool,
             data = dados,
             center = mean)

```

```

## Levene's Test for Homogeneity of Variance (center = mean)
##          Df F value Pr(>F)
## group      5  1.5268 0.2021
##             42

```

Uma outra maneira de avaliar a homogeneidade da variância, é construir um gráfico diagnóstico¹⁹ (Figura 154) do modelo com a função `plot()`, tipo 1, resíduos versus ajustes (*Residuals vs Fitted*).

```

plot(mod.int.lm, 1)

```

Não há correlações óbvias entre resíduos e valores ajustados (a média de cada grupo) no gráfico abaixo, onde a linha vermelha tracejada segue praticamente uma linha horizontal em torno de 0, o que é bom. Como resultado, pode-se, assim como no teste de Levene, assumir que as variâncias são homogêneas.

Verifica-se o mesmo ocorrido com a normalidade, os resultados nos resíduos não diferem daqueles realizados com os dados brutos.

13.3.6 Realização do teste de ANOVA de dois fatores

Inicialmente, será conduzida uma ANOVA de duas vias, incluindo na fórmula da função `aov()`, do pacote `stats`, a variável desfecho `escore` e as variáveis independentes somadas (+), `alcool` e `sexo`. A *Tabela da ANOVA*, pode ser obtida a partir do modelo, usando a função `summary()`:

¹⁹Outros gráficos diagnósticos podem ser obtidos para analisar resíduos em um modelo de regressão (118)

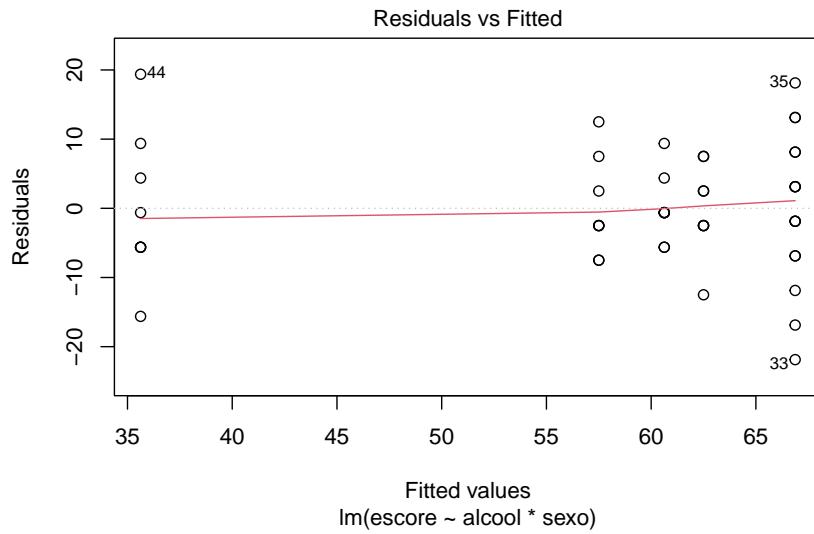


Figura 154: Resíduos versus ajuste

```
mod.aov <- aov(formula = escore ~ alcool + sexo,
                  data = dados)
summary (mod.aov)

##           Df Sum Sq Mean Sq F value    Pr(>F)
## alcool      2   3332   1666.1  13.413 2.83e-05 ***
## sexo        1     169    168.7   1.358      0.25
## Residuals  44   5466    124.2
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

O efeito principal da ingestão do álcool sobre a memória é significativo ($P < 0,0001$). Em relação ao efeito principal do sexo, ele é não significativo ($P > 0,05$).

Se um efeito significativo de um fator foi encontrado, pode-se fazer *testes post-hoc* para testar a diferença entre cada par de níveis da variável independente. Existem muitos tipos de comparações pareadas que fazem suposições diferentes. Um dos testes post-hoc mais comuns para ANOVA padrão é o teste de *Diferença Honestamente Significativa* (HSD) de Tukey. Ele pode ser realizado com a função `HSDTukey()` do pacote `stats`:

```
TukeyHSD(mod.aov)
```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = escore ~ alcool + sexo, data = dados)
##
## $alcool
##            diff      lwr      upr      p adj
## 3 latas-nenhum  0.9375 -8.620051 10.495051 0.9692998
```

```

## 6 latas-nenhum -17.1875 -26.745051 -7.629949 0.0002228
## 6 latas-3 latas -18.1250 -27.682551 -8.567449 0.0001043
##
## $sexo
##           diff      lwr      upr     p adj
## Masculino-Feminino -3.75 -10.23421 2.734212 0.2500789

```

O teste post-hoc evidenciou que quanto maior a quantidade álcool, maior o efeito (6 latas de cerveja > 3 latas > cerveja sem álcool). Não houve diferença nos sexos, entretanto, como este é um modelo aditivo, sem verificar interações, não é possível dizer se o efeito do álcool depende do sexo, ou seja, que haja interação entre as variáveis independentes.

Quase sempre útil combinar uma tabela de resumo da ANOVA com uma tabela de resumo de regressão. Como foi mencionado, a ANOVA é um caso especial de regressão. Logo, se obtém os mesmos resultados com um objeto de regressão e com um objeto ANOVA. No entanto, o formato dos resultados é diferente e frequentemente mais fácil de interpretar.

```

mod.lm <- lm(formula = escore ~ alcool + sexo,
              data = dados)
summary(mod.lm)

##
## Call:
## lm(formula = escore ~ alcool + sexo, data = dados)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -24.6875  -6.5625  -0.1562   6.7188  22.1875
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 65.6250    3.2174  20.397 < 2e-16 ***
## alcool3 latas  0.9375    3.9405   0.238   0.813
## alcool6 latas -17.1875   3.9405  -4.362 7.67e-05 ***
## sexoMasculino -3.7500    3.2174  -1.166   0.250
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.15 on 44 degrees of freedom
## Multiple R-squared:  0.3905, Adjusted R-squared:  0.3489
## F-statistic: 9.395 on 3 and 44 DF,  p-value: 6.471e-05

```

Como é possível ver, a tabela de regressão não nos fornece testes para cada variável como a tabela ANOVA. Em vez disso, ela nos diz o quanto diferente cada nível de uma variável independente é de um valor padrão. Pode-se dizer qual valor de uma variável independente é a variável padrão apenas vendo qual valor está faltando na tabela. Nesse caso, não aparece o coeficiente para o **alcool-nenhum**, então esse é o valor padrão.

O intercepto na tabela nos informa a média do valor padrão. Nesse caso, o escore médio do **alcool-nenhum** foi de 65,6. Os coeficientes para os outros níveis informam que **alcool-3latas** tem, em média, um escore de memória 0,9 maior do que o **alcool-nenhum**, e **alcool-6latas**, em média, reduz o escore de memória 17,2 em relação ao **alcool-nenhum**. Não surpreendentemente, essas são as mesmas diferenças observadas no teste Tukey HSD!

13.3.6.1 ANOVA com interação As interações entre variáveis testam se o efeito de uma variável depende ou não de outra variável. Por exemplo, é possível usar uma interação para responder à pergunta: o efeito do álcool depende do sexo do indivíduo?

Se for analisado com as informações que se tem até agora, pode-se dizer que o efeito principal do sexo não é significativo ($P = 0,250$). Portanto, ignorando quanto de álcool foi ingerido pelo indivíduo, o sexo do indivíduo não influencia o escore de memória. Em outras palavras, ignorando outros efeitos, homens e mulheres sofrem o mesmo efeito do álcool. Entretanto, em presença de interação, não faz sentido interpretar os efeitos principais. Dessa forma, quando há possibilidade de dependência de uma variável em relação à outra, é imperativo incluir a interação no modelo.

Para incluir termos de interação em uma ANOVA, basta usar um asterisco (*) em vez do sinal de mais (+) entre os termos em sua fórmula.

Observe que quando se inclui um termo de interação em um objeto de regressão, o R incluirá automaticamente os efeitos principais. Será repetida a ANOVA anterior a mesmas duas variáveis independentes, mas agora incluindo a interação entre `sexo` e `alcool`.

```
mod.int.aov <- aov(formula = escore ~ alcool * sexo,
                     data = dados)
summary(mod.int.aov)

##           Df Sum Sq Mean Sq F value    Pr(>F)
## alcool      2   3332   1666.1  20.065 7.65e-07 ***
## sexo        1     169    168.7   2.032    0.161
## alcool:sexo 2   1978    989.1  11.911 7.99e-05 ***
## Residuals  42   3488     83.0
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Parece que realmente existe uma interação significativa entre `sexo` e `alcool`. Em outras palavras, o efeito do álcool depende do sexo do bebedor. Isso faz sentido, dado o gráfico de linha plotado na visualização dos dados.

Para entender a natureza da diferença, observe os coeficientes de regressão do objeto de regressão, criado quando se verificou os pressupostos da ANOVA. Usar a função `summary()` com o modelo `mod.int.lm`:

```
summary(mod.int.lm)

##
## Call:
## lm(formula = escore ~ alcool * sexo, data = dados)
##
## Residuals:
##       Min     1Q Median     3Q    Max
## -21.875 -5.625 -0.625  5.156 19.375
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  60.625     3.222  18.818 < 2e-16 ***
## alcool3 latas      1.875     4.556   0.412   0.683  
## alcool6 latas     -3.125     4.556  -0.686   0.497  
## sexoMasculino    6.250     4.556   1.372   0.177  
## alcool3 latas:sexoMasculino -1.875     6.443  -0.291   0.772  
## alcool6 latas:sexoMasculino -28.125    6.443  -4.365 8.12e-05 ***
```

```

## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.112 on 42 degrees of freedom
## Multiple R-squared: 0.6111, Adjusted R-squared: 0.5648
## F-statistic: 13.2 on 5 and 42 DF, p-value: 9.609e-08

```

Novamente, para interpretar esta tabela, primeiro há necessidade de saber quais são os valores padrão. Pode-se dizer isso pelos coeficientes que estão “faltando” na tabela. Como não aparece termo `sexoFeminino`, por exemplo, isso significa que ele é o padrão. Então, o intercepto é o valor médio do escore de memória das mulheres e, pode-se interpretar o coeficiente `sexoMasculino(6,3)` como a diferença do escore de memória dos homens em relação às mulheres. Como esta diferença é positiva, ele está levemente superior. Considerando-se como padrão `sexoMasculino:nenhum`, observa-se que o escore de memória médio dos homens que ingerem 6 latas de cerveja reduz em 28.

Os termos de interação dizem como o efeito dos álcool muda quando quem o ingere é do sexo masculino ou feminino. O efeito do álcool na memória das mulheres é pequeno, ficando praticamente estável nas três condições. Por outro lado, os homens permanecem estáveis no seu escore de memória quando quantidades pequenas de álcool são ingeridas, declina rapidamente quando ingerem 6 latas de cerveja.

13.3.6.2 Tipos de ANOVA Existem três abordagens essencialmente distintas para fazer uma ANOVA - chamadas, Tipo 1, 2 e 3 (ou Tipo I, II e III). Esses tipos diferem em como calculam a variabilidade, especificamente, as somas dos quadrados.

Se os dados forem relativamente balanceados, o que significa que há números relativamente iguais de observações em cada grupo, todos os três tipos fornecerão a mesma resposta. No entanto, se seus dados estiverem desbalanceados, o que significa que alguns grupos de dados têm muito mais observações do que outros, você precisará usar o Tipo II (2) ou o Tipo III (3).

Para verificar se os dados estão平衡ados, pode-se usar a seguinte função:

```

with(dados,
  table(sexo, alcool))

```

```

##           alcool
## sexo      nenhum 3 latas 6 latas
##   Feminino     8     8     8
##   Masculino    8     8     8

```

Os resultados mostram o mesmo número de indivíduos em todas as células, portanto, não importa qual o tipo de ANOVA a ser usado. Os resultados serão iguais.

Serão mostrados os três tipos, utilizando um modelo de regressão (`mod.int.lm`), criado anteriormente. Este modelo será inserido como argumento principal para `aov()` para uma ANOVA Tipo I ou `Anova()` do pacote `car` para uma ANOVA Tipo II ou Tipo III:

ANOVA Tipo I

A ANOVA tipo I testa primeiro o efeito de um fator, seguido do efeito do outro fator dado que se conhece o primeiro, seguido pela interação entre eles, dado que os efeitos principais já são conhecidos. Esta ordem natural (fator A -> fator B -> A*B) é a razão desta ANOVA ser conhecida também como soma de quadrados sequencial.

No exemplo do efeito do álcool na memória de homens e mulheres, tem-se:

```

memoria.I <- aov(mod.int.lm)
summary (memoria.I)

##          Df Sum Sq Mean Sq F value    Pr(>F)
## alcool      2   3332   1666.1  20.065 7.65e-07 ***
## sexo        1     169    168.7   2.032    0.161
## alcool:sexo 2   1978    989.1  11.911 7.99e-05 ***
## Residuals   42   3488     83.0
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

ANOVA Tipo II

Este tipo de ANOVA testa o efeito de um dos fatores principais dado que o outro já é conhecido. Assim, assume-se a não significância da interação. Existe a sugestão de se testar a interação, conhecendo-se o efeito dos fatores individualmente. Se de fato a interação for não significativa, então o tipo II é estatisticamente mais poderoso que o tipo III.

```

memoria.II <- car::Anova(mod.int.lm, type = 2)
memoria.II

```

```

## Anova Table (Type II tests)
##
## Response: escore
##          Sum Sq Df F value    Pr(>F)
## alcool      3332.3  2 20.0654 7.649e-07 ***
## sexo         168.8  1  2.0323    0.1614
## alcool:sexo 1978.1  2 11.9113 7.987e-05 ***
## Residuals   3487.5 42
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

ANOVA Tipo III

Este tipo de ANOVA só é valido quando a interação é significativa. Entretanto, em muitos casos, quando existe interação, não há interesse nos efeitos principais isoladamente.

A ANOVA Tipo III necessita que se faça modificação nos contrastes:

```

memoria.III <- car::Anova(lm(formula = escore ~ sexo*alcool,
                               data = dados,
                               contrasts=list(sexo=contr.sum, alcool=contr.poly)),
                               type = 3)
memoria.III

```

```

## Anova Table (Type III tests)
##
## Response: escore
##          Sum Sq Df F value    Pr(>F)
## (Intercept) 163333  1 1967.0251 < 2.2e-16 ***
## sexo         169    1    2.0323    0.1614
## alcool      3332   2   20.0654 7.649e-07 ***
## sexo:alcool 1978   2   11.9113 7.987e-05 ***
## Residuals   3487  42
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

13.3.7 Testes post-hoc

Uma interação entre os dois fatores significativa indica que o impacto que um fator tem na variável desfecho ou resposta depende do nível do outro fator e vice-versa. Portanto, pode-se decompor uma interação de dois fatores significativa em:

1. *Efeito principal simples*: executar o modelo de um fator da primeira variável em cada nível da segunda variável,
2. *Comparações de pares simples (Simple pairwise comparisons)*: se o efeito principal simples for significativo, executar várias comparações de pares para determinar quais grupos são diferentes.

Para uma interação de dois fatores não significativa, há necessidade de determinar se há algum efeito principal estatisticamente significativo na saída ANOVA. Um efeito principal significativo pode ser seguido por comparações de pares entre grupos.

Efeitos principais simples

No exemplo usado nesta seção, poder-se-ia, portanto, investigar o efeito do consumo de álcool no sexo ou investigar o efeito do sexo em todos os níveis da variável consumo de álcool.

Aqui, será executada uma ANOVA de um fator do consumo de álcool em cada nível de sexo. Observe que, se as premissas da ANOVA de duas vias foram atendidas (por exemplo, homogeneidade de variâncias), é melhor usar o termo de erro geral (da ANOVA dois fatores) como entrada no modelo ANOVA de um fator. Isso tornará mais fácil detectar quaisquer diferenças estatisticamente significativas, caso existam (119) (120). Se o pressuposto da homocedasticidade for violado, pode-se considerar a execução da ANOVA de um fator separadas com termos de erro separados.

No exemplo do efeito do álcool na memória de homens e mulheres, serão agrupados os dados por sexo e analisados os efeitos principais simples do nível do consumo de álcool no escore de memória. O argumento `erro` é usado para especificar o modelo ANOVA, no caso modelo `lm` (`mod.int.lm`), na função `anova_test()` do pacote `rstatix`:

```
dados %>%
  dplyr::group_by(sexo) %>%
  rstatix::anova_test(escore~alcool, error = mod.int.lm)
```

```
## # A tibble: 2 x 8
##   sexo     Effect   DFn    DFd      F          p `p<.05`    ges
## * <fct>    <chr>   <dbl> <dbl>    <dbl>    <dbl> <chr>    <dbl>
## 1 Feminino alcool     2     42  0.615  0.546      ""     0.028
## 2 Masculino alcool    2     42 31.4   0.0000000465 **     0.599
```

O resultado mostra que o efeito principal simples do consumo de álcool no escore de memória foi estatisticamente significativo para homens ($P = 4,65 \times 10^{-9}$) e não significativo para as mulheres ($P = 0,546$). Em outras palavras, há uma diferença estatisticamente significativa no escore médio de memória entre homens com o consumo de álcool $F(2, 42) = 31,4$, $P < 0,0001$. Esta conclusão não é válida para mulheres, $F(2, 542) = 0,615$, $P = 0,546$.

Comparações por pares

Um efeito principal simples, estatisticamente significativo, pode ser seguido por múltiplas comparações de pares para determinar quais médias de grupo são diferentes. Agora, serão realizadas múltiplas comparações entre pares nos diferentes grupos de consumo de álcool por sexo.

Pode-se executar e interpretar todas as comparações de pares possíveis usando um ajuste de Bonferroni. Isso pode ser feito facilmente usando a função `emmeans_test()`, incluída no pacote `rstatix` (médias marginais estimadas, também conhecidas como médias dos mínimos quadrados ou médias ajustadas (121)).

Serão comparados os escores dos diferentes sexos por níveis de consumo de álcool:

```
pwc <- dados %>%
  group_by(alcool) %>%
  emmeans_test (escore ~ sexo,
                 p.adjust.method = "bonferroni")
pwc

## # A tibble: 3 x 10
##   alcool term .y.   group1   group2      df statistic      p   p.adj p.adj~2
## * <chr>  <chr> <chr> <chr>    <dbl>     <dbl>     <dbl>     <dbl> <chr>
## 1 3 latas sexo  escore Feminino Masculino    42   -0.960 3.42e-1 3.42e-1 ns
## 2 6 latas sexo  escore Feminino Masculino    42    4.80  2.02e-5 2.02e-5 ****
## 3 nenhum sexo  escore Feminino Masculino    42   -1.37 1.77e-1 1.77e-1 ns
## # ... with abbreviated variable names 1: statistic, 2: p.adj.signif
```

Agora, serão comparados os escores de diferentes níveis de consumo de álcool por sexo

```
pwc.1 <- dados %>%
  group_by(sexo) %>%
  emmeans_test (escore ~ alcool,
                 p.adjust.method = "bonferroni")
pwc.1

## # A tibble: 6 x 10
##   sexo     term   .y.   group1   group2      df statistic      p   p.adj p.adj~1
## * <chr>  <chr> <chr> <chr>    <dbl>     <dbl>     <dbl>     <dbl> <chr>
## 1 Feminino alcool escore nenhum 3 latas    42  -4.12e-1 6.83e-1 1  e+0 ns
## 2 Feminino alcool escore nenhum 6 latas    42   6.86e-1 4.97e-1 1  e+0 ns
## 3 Feminino alcool escore 3 latas 6 latas    42   1.10e+0 2.79e-1 8.36e-1 ns
## 4 Masculino alcool escore nenhum 3 latas    42  -4.39e-16 1  e+0 1  e+0 ns
## 5 Masculino alcool escore nenhum 6 latas    42   6.86e+0 2.31e-8 6.94e-8 ****
## 6 Masculino alcool escore 3 latas 6 latas    42   6.86e+0 2.31e-8 6.94e-8 ****
## # ... with abbreviated variable name 1: p.adj.signif
```

As Saída exibe resultados onde aparece que o consumo de álcool não afetou a memória das mulheres, mas o consumo de 6 latas de cerveja diminuiu o escore de memória dos homens quando comparados com homens que consumiram cerveja sem álcool ou que ingeriram apenas 3 latas de cerveja.

Teste de Tukey de múltiplas comparações

Para completar os cruzamentos da análise, pode-se ainda usar o teste de Tukey de múltiplas comparações, denominado HSD de Tukey, através da função `TukeyHSD()`, do pacote `stats`. Esta função usa como argumento um modelo `aov`, no exemplo, `mod.int.aov`.

```
TukeyHSD(mod.int.aov)
```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
```

```

## 
## Fit: aov(formula = escore ~ alcool * sexo, data = dados)
## 
## $alcool
##          diff      lwr      upr     p adj
## 3 latas-nenhum  0.9375 -6.889643  8.764643 0.9544456
## 6 latas-nenhum -17.1875 -25.014643 -9.360357 0.0000105
## 6 latas-3 latas -18.1250 -25.952143 -10.297857 0.0000040
## 
## $sexo
##          diff      lwr      upr     p adj
## Masculino-Feminino -3.75 -9.058607 1.558607 0.1613818
## 
## $`alcool:sexo`
##          diff      lwr      upr     p adj
## 3 latas:Feminino-nenhum:Masculino  1.875 -11.726381 15.476381 0.9983764
## 6 latas:Feminino-nenhum:Masculino -3.125 -16.726381 10.476381 0.9825753
## nemum:Masculino-nenhum:Feminino   6.250 -7.351381 19.851381 0.7432243
## 3 latas:Masculino-nenhum:Feminino  6.250 -7.351381 19.851381 0.7432243
## 6 latas:Masculino-nenhum:Feminino -25.000 -38.601381 -11.398619 0.0000306
## 6 latas:Feminino-3 latas:Feminino -5.000 -18.601381  8.601381 0.8796489
## nemum:Masculino-3 latas:Feminino  4.375 -9.226381 17.976381 0.9277939
## 3 latas:Masculino-3 latas:Feminino  4.375 -9.226381 17.976381 0.9277939
## 6 latas:Masculino-3 latas:Feminino -26.875 -40.476381 -13.273619 0.0000080
## nemum:Masculino-6 latas:Feminino  9.375 -4.226381 22.976381 0.3286654
## 3 latas:Masculino-6 latas:Feminino  9.375 -4.226381 22.976381 0.3286654
## 6 latas:Masculino-6 latas:Feminino -21.875 -35.476381 -8.273619 0.0002776
## 3 latas:Masculino-nenhum:Masculino  0.000 -13.601381 13.601381 1.0000000
## 6 latas:Masculino-nenhum:Masculino -31.250 -44.851381 -17.648619 0.0000003
## 6 latas:Masculino-3 latas:Masculino -31.250 -44.851381 -17.648619 0.0000003

```

A saída mostra que o fato de ser homem ou mulher não afetou a memória, não havendo diferença significativa entre os sexos. O importante aqui é observar a interação. Com relação ao consumo do álcool, um efeito significativo após o consumo de 6 latas de cerveja, ocorreu no sexo masculino.

13.3.8 Relatando os resultados de uma ANOVA de dois fatores

Pode-se relatar os resultados da ANOVA de dois fatores da seguinte maneira:

- (1) Uma ANOVA de dois fatores foi realizada para avaliar se a memória de homens e mulheres era afetada pelo consumo do álcool aviado em três níveis:
 - Não consumiram álcool
 - Consumiram 3 latas de cerveja (~ 1L)
 - Consumiram 6 latas de cerveja (~ 2L)
- (2) Os dados são apresentados como média e desvio padrão, na Tabela 14.
- (3) O efeito principal do sexo na memória foi não significativo ($F(1,42) = 2,03, P = 0,1614$).
- (4) Houve um efeito principal significativo de acordo com a quantidade de álcool consumida na memória dos participantes ($F(2,42) = 20,07, P < 0,0001$).

Tabela 14: Efeito do Álcool sobre a Memória - Escore médio (desvio padrão)

Sexo	Sem_alcool	Um_litro	Dois_litros	Valor.P
Feminino	60,6 (5,0)	62,5 (6,6)	57,5 (7,0)	0,546
Masculino	66,9 (10,3)	66,9 (12,5)	35,6 (10,8)	<0,0001
Valor P	0,793	0,927	0,0003	

^a Um litro de cerveja (4,5%) = 5 unidades de álcool

- (5) As análises posteriores (médias marginais estimadas com correção de Bonferroni) revelaram que a memória não foi afetada nas mulheres pelo consumo de álcool, mas o consumo de 6 latas de cerveja afetou a memória dos homens quando comparados os homens que não consumiram álcool ou que consumiram até 3 latas de cerveja.

- (6) Visualização dos resultados:

Serão apresentados gráficos de barra de erro (Figura 155), com `ggbarplot()`, do pacote `ggpubr`, utilizando, para cores tonalidades de cinza. Para adicionar teste estatístico, usou-se a função `get_test_label()` e para o teste *post hoc*, a função `get_pwc_label()`, ambas do pacote `rstatix`.

```
bp <- ggbarplot(dados,
  x = "alcool", y = "escore",
  add = "mean_ci",
  error.plot = "upper_errorbar",
  fill = "sexo",
  palette = c("gray60", "gray40"),
  position = position_dodge(0.8)) +
  theme(legend.key.size = unit(0.3, 'cm')) +
  theme(legend.position = "right")

pwc <- pwc %>%
  add_xy_position(fun = "mean_ci",
    x = "alcool",
    dodge = 0.8)

anova <- anova_test(mod.int.aov)

bp + stat_pvalue_manual(pwc,
  label = "p.adj.signif",
  tip.length = 0.01,
  y.position = 85) +
  labs (x = "Ingestão de álcool",
    y = "Média escore de memória",
    subtitle = get_test_label (anova, detailed = TRUE),
    caption = get_pwc_label(pwc))
```

Uma opção, é apresentar os resultados como um gráfico de linhas. já (Figura 156) mostrado anteriormente, usando a função `gglime()` do pacote `ggpubr`:

```
gl <- gglime(dados,
  x = "alcool",
  y = "escore",
  color = "sexo",
```

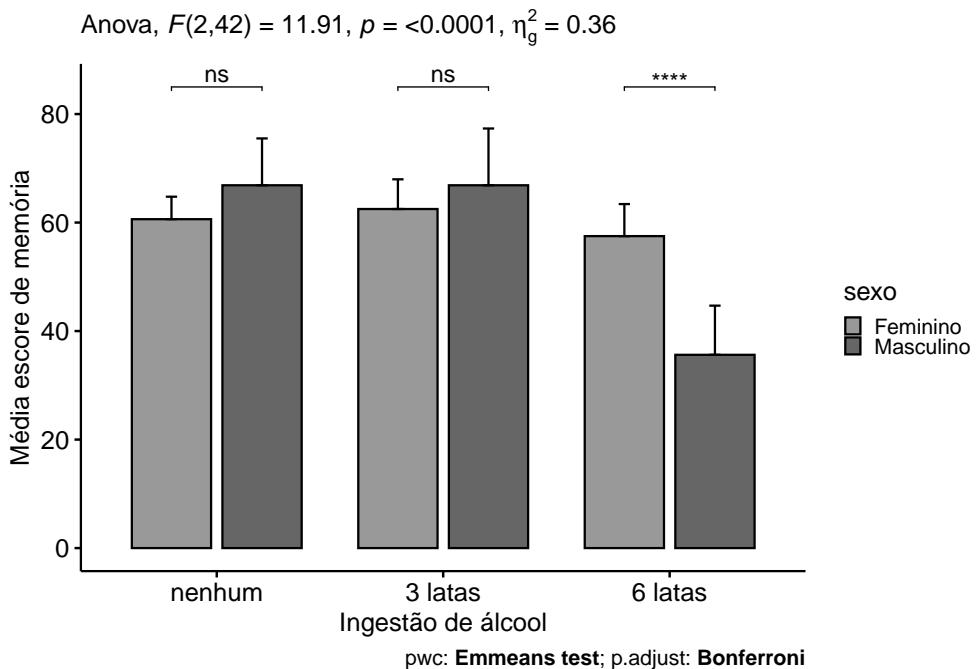


Figura 155: Efeito do álcool na memória de acordo com o sexo.

```

size = 0.7,
add = c("mean_ci"),
palette = c("darkred", "dodgerblue4"))

gl + stat_pvalue_manual(pwc,
                        label = "p.adj.signif",
                        tip.length = 0.01,
                        y.position = 85) +
  labs (x = "Ingestão de álcool",
        y = "Média escore de memória",
        subtitle = get_test_label (anova, detailed = TRUE),
        caption = get_pwc_label(pwc))

```

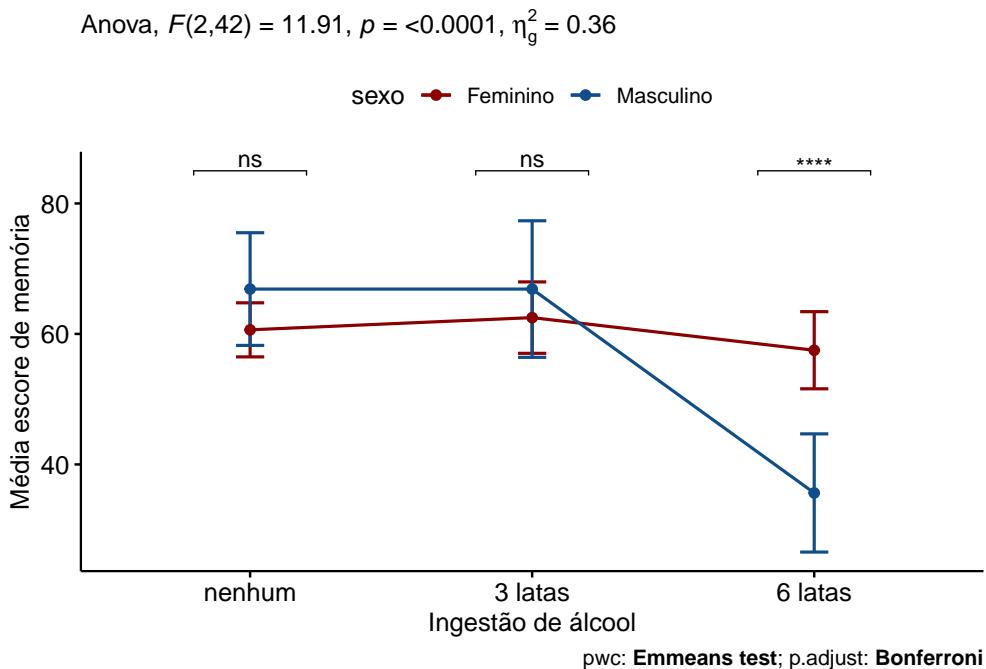


Figura 156: Efeito do álcool na memória de acordo com o sexo.

14 ANOVA de medidas repetidas

14.1 ANOVA de medidas repetidas de um fator

14.1.1 Pacotes necessários

Instalar e carregar os seguintes pacotes:

```
pacman::p_load(readxl, rstatix, dplyr, ggplot2, ggpibr, tidyverse, ggsci)
```

14.1.2 Preparação dos dados

O conjunto de dados contém a pontuação de autoestima de 10 indivíduos em três pontos de tempo durante uma dieta específica para determinar se sua autoestima melhorou. A autoestima foi determinada por uma escala (122) (123), cujos resultados variam de 0 a 30 pontos. Valores entre 15 e 25 caracterizam uma autoestima muito boa; abaixo de 15 é considerada baixa autoestima.

Os dados podem ser obtidos [aqui](#). Baixe no seu diretório de trabalho e carregue com a função `read_excel()` do pacote `readxl`:

```
dados <- read_excel("dadosAutoestima.xlsx")
head(dados)
```

```
## # A tibble: 6 x 4
##       id     t1     t2     t3
##   <dbl> <dbl> <dbl> <dbl>
## 1      1 12.0  15.5  21.3
```

```

## 2      2  7.67 20.7 18.9
## 3      3  9.73 13.3 29.3
## 4      4 10.3 14.1 25.0
## 5      5  8.61 11.7 19.4
## 6      6  6.14 16.0 20.0

```

Os dados se encontram no formato amplo e para realizar a ANOVA de medidas repetidas, o *R* necessita que os dados estejam no formato longo. Para fazer esta transformação será usada a função `gather()` do pacote `tidyverse` (60). Nesta função, no argumento `data`, coloca-se o nome do conjunto de dados; em `key`, há necessidade de nomear a coluna a ser criada que receberá as colunas do formato amplo que serão reunidas. No argumento `value`, nomear a coluna que receberá os valores e em `values_to`, especificar o nome da variável no formato longo que conterá os valores. A variável `id` e a nova variável `tempo` devem ser convertida para fatores:

```

dadosL <- dados %>%
  gather(key = "tempo", value = "escore", t1, t2, t3) %>%
  convert_as_factor(id, tempo)

head(dadosL)

```

```

## # A tibble: 6 x 3
##   id    tempo escore
##   <fct> <fct>  <dbl>
## 1 1     t1     12.0
## 2 2     t1     7.67
## 3 3     t1     9.73
## 4 4     t1     10.3
## 5 5     t1     8.61
## 6 6     t1     6.14

```

14.1.3 Sumarização dos dados

Calcule algumas estatísticas resumidas dos escores de autoestima por grupos (`tempo`): média e desvio padrão, usando a funções `group_by()` e `get_summary_stats()` do pacote `rstatix`:

```

dadosL %>%
  rstatix::group_by(tempo) %>%
  get_summary_stats(escore, type = "mean_sd")

## # A tibble: 3 x 5
##   tempo variable     n   mean     sd
##   <fct> <fct>   <dbl> <dbl> <dbl>
## 1 t1    escore     10  9.42  1.66
## 2 t2    escore     10 14.8   2.59
## 3 t3    escore     10 22.9   3.43

```

14.1.4 Visualização dos dados

A visualização pode ser obtida com um conjunto de boxplots (Figura 157)com as cores do *BMJ* ou um gráfico de linha, acrescido de barras de erro. Estes gráficos permitem visualizar a variação dos escores com o tempo.

```

ggboxplot (dadosL,
            bxp.errorbar = TRUE,
            bxp.errorbar.width = 0.1,
            x = "tempo",
            y = "escore",
            color = "black",
            fill = "tempo",
            ylab = "Escore de Autoestima",
            xlab = "Tempo",
            legend = "none") +
scale_fill_grey(start=0.95, end=0.6) +
theme (text = element_text (size = 12))

```

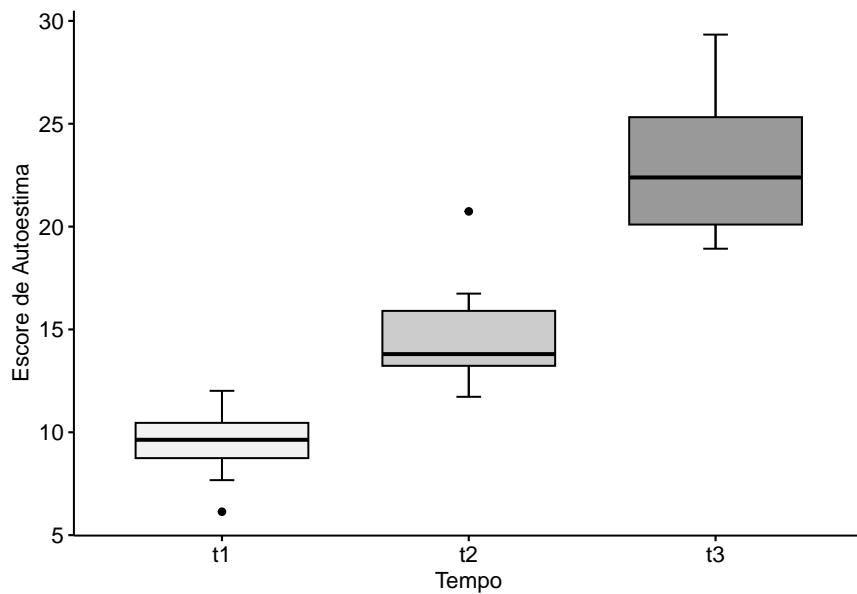


Figura 157: Impacto de uma dieta específica na autoestima.

O gráfico de linha (Figura 158) mostra bem o comportamento dos escores com o tempo:

```

ggline(dadosL,
        x = "tempo",
        y = "escore",
        color = "darkblue",
        size = 0.7,
        linetype = "dashed",
        add = c("mean_ci")) +
ylab("Escore de Autoestima") +
xlab("Tempo")

```

14.1.5 Avaliação dos pressupostos

A ANOVA de medidas repetidas faz as seguintes suposições sobre os dados:

1. A amostra foi selecionada aleatoriamente da população;

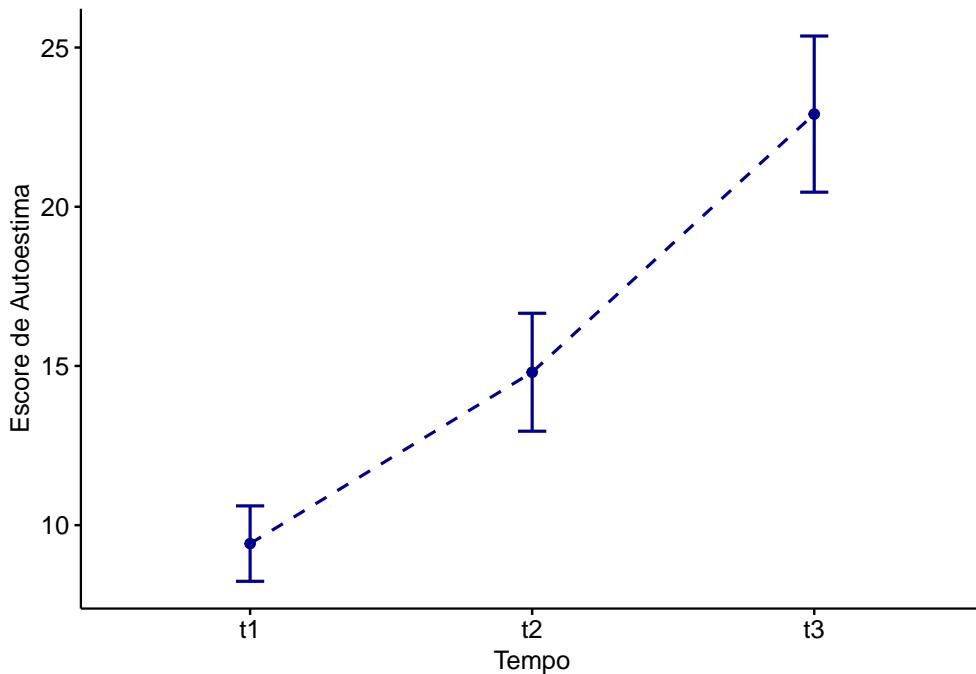


Figura 158: Impacto de uma dieta específica na autoestima.

2. A variável dependente é normalmente distribuída na população para cada nível do fator dentro dos sujeitos;
3. Não deve existir outliers extremos;
4. Existência de esfericidade

14.1.5.1 Identificação de valores atípicos Não deve haver valores atípicos em nenhuma célula do delineamento. Isso pode ser verificado visualizando os dados nos boxplots, mostrados anteriormente, onde se observa a presença de dois *outliers*, um no t1 e outro em t2. Além disso, pode-se verificar a presença de valores atípicos, usando a função `identify_outliers()` do pacote `rstatix`.

```
dadosL %>%
  group_by(tempo) %>%
  identify_outliers(escore)

## # A tibble: 2 x 5
##   tempo id    escore is.outlier is.extreme
##   <dbl> <dbl> <dbl>     <lgl>      <lgl>
## 1 t1     6     6.14 TRUE      FALSE
## 2 t2     2     20.7  TRUE      FALSE
```

A saída confirma a presença de dois valores atípicos, em t1 e em t2. Entretanto, eles não são extremos, não estão afastados acima de 3 intervalos interquartis e, provavelmente, não trarão problemas, apesar da amostra ser pequena.

14.1.5.2 Avaliação da normalidade Para testar a hipótese de normalidade dos dados, será utilizado o teste de Shapiro-Wilk através da função `shapiro_test()`, do pacote `rstatix` e a função `group_by()`, incluída no pacote `dplyr` ou `rstatix`, junto com o operador pipe (%>%):

```

dadosL %>%
  group_by (tempo) %>%
  shapiro_test(escore)

```

```

## # A tibble: 3 x 4
##   tempo variable statistic     p
##   <fct>  <chr>      <dbl> <dbl>
## 1 t1     escore     0.967 0.859
## 2 t2     escore     0.876 0.117
## 3 t3     escore     0.923 0.380

```

Os resultados da Saída mostram que os escores de autoestima estão normalmente distribuídos em cada momento.

É possível construir um gráfico QQ (Figura 159) para cada um dos momentos, usando a função `ggqqplot ()` do pacote `ggnetwork`, consulte a vinheta do pacote para maiores detalhes. Foi utilizado também o argumento `faced.by`, que divide em painéis, organizando-os como uma grade, de acordo com o momento (`t1`, `t2` e `t3`).

```

ggqqplot(dadosL,
          x = "escore",
          facet.by = "tempo",
          color = "tempo",
          palette = get_palette("Dark2", 3),
          legend = "none")

```

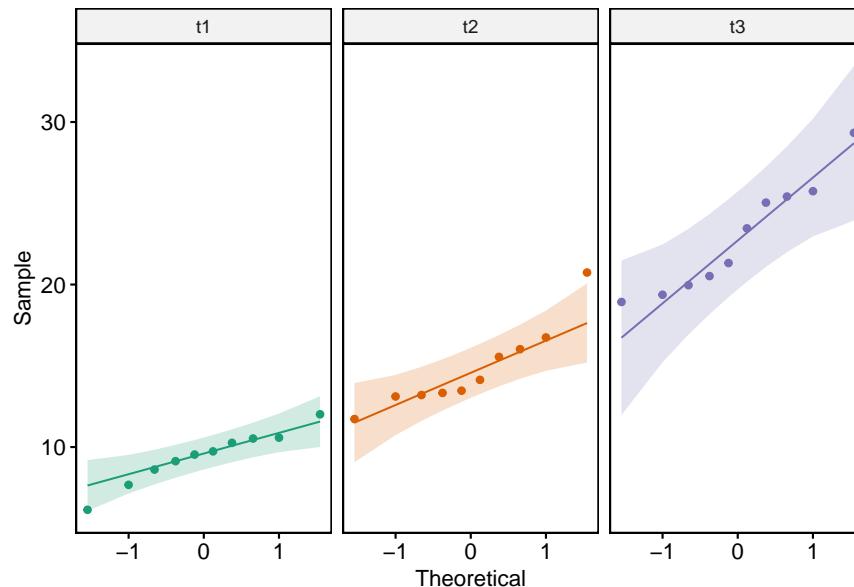


Figura 159: Gráfico QQ para verificar a normalidade

Observando o gráfico, como quase todos os pontos caem aproximadamente ao longo da linha de referência, pode-se assumir a normalidade dos escores em todos os momento

14.1.5.3 Esfericidade A violação da suposição de esfericidade pode distorcer os cálculos de variância resultantes de um teste ANOVA de medidas repetidas mais liberal (ou seja, um aumento na taxa de erro

Tipo I). Nesse caso, a ANOVA de medidas repetidas deve ser corrigida apropriadamente dependendo do grau em que a esfericidade foi violada. Na relação entre os escores, há necessidade de pressupor que exista esfericidade (ϵ - épsilon), também chamada de circularidade, grosseiramente semelhante à homocedasticidade da ANOVA de uma via. A ANOVA de medidas repetidas pressupõe que as variâncias das diferenças entre todas as combinações de condições relacionadas (ou níveis de grupo) são iguais. A melhor maneira de verificarla é calcular as diferenças entre os pares de escores em todas as combinações dos níveis de tratamento.

O teste de esfericidade de Mauchly é usado para avaliar se a suposição de esfericidade é atendida ou não. Isso é relatado automaticamente ao usar a função `anova_test()` do pacote `rstatix`. Se o teste resulta em um valor P menor do que 0,05, pode-se concluir de que há uma diferença significativa entre as variâncias das diferenças.

O principal problema da violação da condição de esfericidade é a ocorrência de testes F não exatos e liberais, com consequente perda do poder do teste. Existem várias correções que podem ser aplicadas para produzir uma razão F válida, através do ajuste dos graus de liberdade.

As correções mais frequentemente preconizadas são o ϵ de Greenhouse-Geisser (GGe) e o ϵ de Huynh-Feldt (HFe). Huynh e Feldt (124) relataram que quando a correção ϵ de Greenhouse-Geisser é $> 0,75$ muitas hipóteses nulas falsas deixam de ser rejeitadas, isto é, o teste é muito conservador, propondo outra correção dos graus de liberdade. É recomendado o uso da correção de Greenhouse-Geisser para o ajuste dos graus de liberdade quando $\epsilon < 0,75$ ou nada se sabe a respeito da esfericidade (125). Avaliando o poder destes testes, Muller (126) verificou que a correção de Greenhouse-Geisser fornece um controle adicional do erro Tipo I, enquanto o poder é maximizado.

A verificação da esfericidade é realizada junto com a realização do modelo de ANOVA, realizadi com a função `anova_test()` do `rstatix`.

14.1.6 Cálculo ANOVA de medidas repetidas

Usa-se a função `anova_test()`, do pacote `rstatix`, para o cálculo da ANOVA de medidas repetidas, criando um modelo que será atribuído ao objeto `mod.anova`:

```
mod.anova <- anova_test(data = dadosL, dv = escore, wid = id, within = tempo)
mod.anova
```

```
## ANOVA Table (type III tests)
##
## $ANOVA
##   Effect DFn DFd      F      p p<.05    ges
## 1  tempo    2   18 55.463 2.02e-08     * 0.829
##
## $`Mauchly's Test for Sphericity`
##   Effect      W      p p<.05
## 1  tempo 0.551 0.092
##
## $`Sphericity Corrections`
##   Effect    GGe    DF[GG]    p[GG] p[GG]<.05    HFe      DF[HF]    p[HF]
## 1  tempo 0.69 1.38, 12.42 2.16e-06     * 0.774 1.55, 13.94 6.04e-07
##   p[HF]<.05
## 1          *
```

Em primeiro lugar, observar o Teste de Mauchly para a esfericidade. Verifica-se que efeito do tempo tem um valor $P = 0,092$, ou seja, $> 0,05$ e, portanto, não houve violação da esfericidade e não há necessidade de observar as correções do ϵ de Greenhouse-Geisser (GGe) ou o ϵ de Huynh-Feldt (HFe).

Desta forma, pode-se dizer que houve uma modificação significativa no escore de autoestima, à medida que o tempo passou ($F(2, 18) = 55,5$, $P < 0.0001$, $\eta^2 = 0,83$).

Usando a função `get_anova_table()` do pacote `rstatix` para extrair a tabela ANOVA, a correção de esfericidade Greenhouse-Geisser é aplicada automaticamente aos fatores que violam a suposição de esfericidade.

```
get_anova_table(mod.anova)
```

```
## ANOVA Table (type III tests)
##
##   Effect DFn DFd      F      p p<.05    ges
## 1  tempo     2 18 55.463 2.02e-08    * 0.829
```

Onde,

- **F** Indica que se está comparando com uma distribuição F (teste F); (2, 18) indica os graus de liberdade no numerador (DFn) e no denominador (DFd), respectivamente; 55,5 indica o valor da estatística F obtido.
- **p** especifica o valor P .
- **ges** é o tamanho do efeito generalizado (quantidade de variabilidade devido ao fator dentro dos assuntos), η^2 .

14.1.7 Testes post hoc

É possível fazer comparações por pares, realizando vários testes t pareados entre os níveis do fator (tempo). Os valores P são ajustados usando o método de correção de testes múltiplos de Bonferroni.

```
pwc <- dadosL %>%
  pairwise_t_test(
    escore ~ tempo, paired = TRUE,
    p.adjust.method = "bonferroni"
  )
pwc

## # A tibble: 3 x 10
##   .y.   group1 group2   n1   n2 statistic     df      p    p.adj p.adj.~1
## * <chr> <chr>  <chr> <int> <int>    <dbl> <dbl>    <dbl> <dbl> <chr>
## 1 escore t1     t2      10    10    -4.97     9 0.000773  0.002   **
## 2 escore t1     t3      10    10    -13.2     9 0.000000334 0.000001 ****
## 3 escore t2     t3      10    10    -4.87     9 0.000887  0.003   **
```

Todas as diferenças pareadas são estatisticamente significativas.

14.1.8 Relatando os resultados da ANOVA de medidas repetidas unifatorial

Pode-se relatar de forma simples:

1. O escore de autoestima se modificou de forma significativa de acordo com a passagem do tempo, $F(2, 18) = 55,5$, $p < 0,0001$, eta quadrado generalizado = 0,82.

2. Análises post hoc, com um ajuste de Bonferroni, revelaram que todas as diferenças pareadas, entre os pontos de tempo, foram estatisticamente diferentes ($P < 0,05$).

Uma opção de apresentação gráfica, é o gráfico de linhas (Figura 160), junto com os teste estatísticos:

```
gl <- ggline(dadosL,
  x = "tempo",
  y = "escore",
  color = "darkblue",
  size = 0.7,
  linetype = "dashed",
  add = c("mean_ci"))

gl + stat_pvalue_manual(pwc,
  label = "p.adj",
  tip.length = 0.00,
  y.position = c(23, 29, 26)) +
  labs (x = "Tempo",
  y = "Escore de autoestima",
  subtitle = get_test_label (mod.anova, detailed = TRUE),
  caption = get_pwc_label(pwc))
```

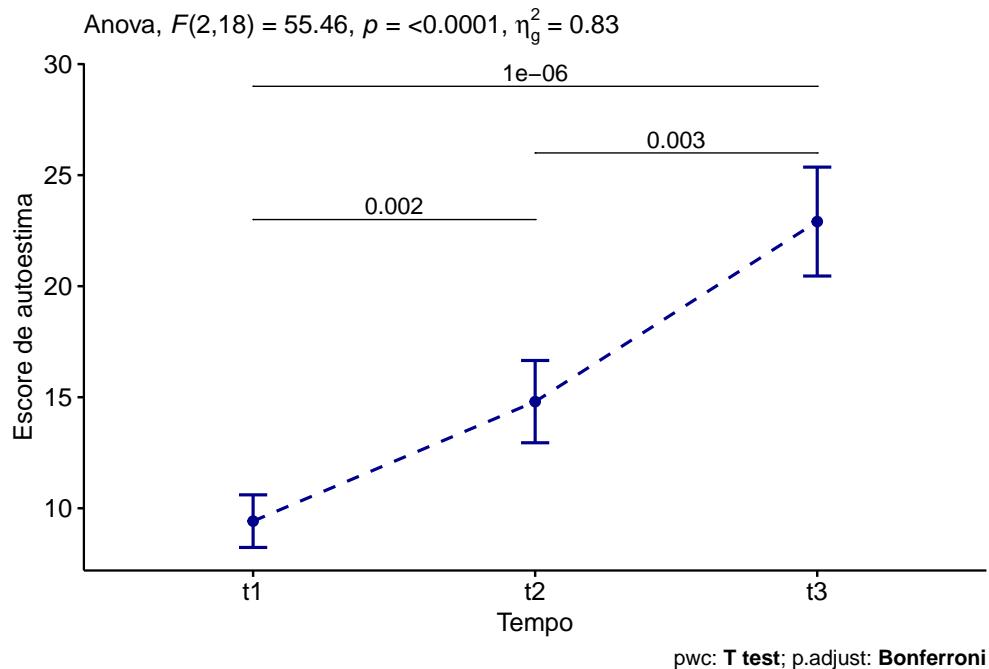


Figura 160: Impacto de uma dieta específica na autoestima.

14.2 ANOVA de medidas repetidas de dois fatores

14.2.1 Preparação dos dados

O conjunto de dados de `dadosAutoestima2.xlsx` contém as medidas dos escores de autoestima de 12 indivíduos inscritos em 2 ensaios clínicos sucessivos de curto prazo (4 semanas): placebo e dieta especial.

Os dados podem ser obtidos [aqui](#). Baixe no seu diretório de trabalho.

Cada participante participou dos dois ensaios. A ordem das tentativas foi equilibrada e foi permitido tempo suficiente entre os ensaios para permitir que quaisquer efeitos dos ensaios anteriores se dissipassem (*washout*).

O escore de autoestima foi registrado em três momentos: no início (t1), no meio (t2) e no final (t3) dos ensaios.

A questão é investigar se esse tratamento dietético de curto prazo pode induzir um aumento significativo do escore de autoestima ao longo do tempo. Em outras palavras, se quer saber se há interação significativa entre dieta e tempo no escore de autoestima.

A ANOVA de medidas repetidas bidirecional pode ser realizada para determinar se existe uma interação significativa entre dieta e tempo no escore de autoestima.

14.2.2 Leitura dos dados

A leitura dos dados será feita com a função `read_excel()` do pacote `readxl`:

```
autoestima <- readxl::read_excel("dadosAutoestima2.xlsx")
head(autoestima)
```

```
## # A tibble: 6 x 5
##       id tratamento     t1     t2     t3
##   <dbl> <chr>      <dbl>  <dbl>  <dbl>
## 1     1 dieta      25.2   25.8   26.4
## 2     2 dieta      30.0   29.7   29.1
## 3     3 dieta      27.3   27.3   27.6
## 4     4 dieta      27.3   27.6   28.5
## 5     5 dieta      22.2   22.8   21.6
## 6     6 dieta      22.8   22.5   22.8
```

Após, será exibida uma linha aleatória por grupo de tratamento, usando a função `sample_n_by()`, do pacote `rstatix`:

```
set.seed(123)
autoestima %>% sample_n_by(tratamento, size = 1)

## # A tibble: 2 x 5
##       id tratamento     t1     t2     t3
##   <dbl> <chr>      <dbl>  <dbl>  <dbl>
## 1     3 dieta      27.3   27.3   27.6
## 2     3 placebo    27.9   27.6   26.7
```

14.2.3 Transformação dos dados

Os dados autoestima estão no formato amplo e as colunas t1, t2 e t3 devem ser reunidas, em uma única variável denominada `tempo`, transformando o formato amplo em longo. A seguir, converter em fator esta nova variável `tempo` e a variável identificadora `id`:

```
autoestimaL <- autoestima %>%
  gather(key = "tempo", value = "escore", t1, t2, t3) %>%
  convert_as_factor(id, tratamento, tempo)
```

Explorar o novo conjunto de dados no formato longo:

```
autoestimaL %>% sample_n_by(tratamento, tempo, size = 1)
```

```
## # A tibble: 6 x 4
##   id    tratamento tempo  escore
##   <fct> <fct>     <fct>   <dbl>
## 1 10    dieta      t1      27
## 2 2    dieta      t2      29.7
## 3 6    dieta      t3      22.8
## 4 11   placebo    t1      27.6
## 5 5    placebo    t2      21.9
## 6 4    placebo    t3      26.7
```

Neste exemplo, o efeito do “tempo” no escore de autoestima é nossa variável focal, nossa principal preocupação.

No entanto, pensa-se que o efeito “tempo” será diferente se o tratamento for realizado ou não. Nesse cenário, a variável “tratamento” é considerada como variável moderadora.

14.2.4 Sumarização dos dados

Os dados serão por `tratamento` e `tempo` e, em seguida, serão calculadas algumas estatísticas resumidas da variável de `escore`: média e sd (desvio padrão).

```
autoestimaL %>%
  group_by(tratamento, tempo) %>%
  get_summary_stats(escore, type = "mean_sd")
```

```
## # A tibble: 6 x 6
##   tratamento tempo variable     n   mean     sd
##   <fct>       <fct> <fct>   <dbl> <dbl> <dbl>
## 1 dieta        t1    escore     12  26.3  2.29
## 2 dieta        t2    escore     12  26.4  2.23
## 3 dieta        t3    escore     12  26.3  2.44
## 4 placebo      t1    escore     12  26.4  2.42
## 5 placebo      t2    escore     12  25.2  3.07
## 6 placebo      t3    escore     12  23.6  3.16
```

14.2.5 Visualização dos dados

Serão criados boxplots (Figura 161) do escore coloridos pelos grupos de tratamento, com cores da paleta do NEJM:

```
ggboxplot (autoestimaL,
           bxp.errorbar = TRUE,
           bxp.errorbar.width = 0.1,
           x = "tempo",
           y = "escore",
           color = "black",
           fill = "tratamento",
```

```

      ylab = "Escore de Autoestima",
      xlab = "Tempo")+
scale_fill_jco() +
theme (text = element_text (size = 12))

```

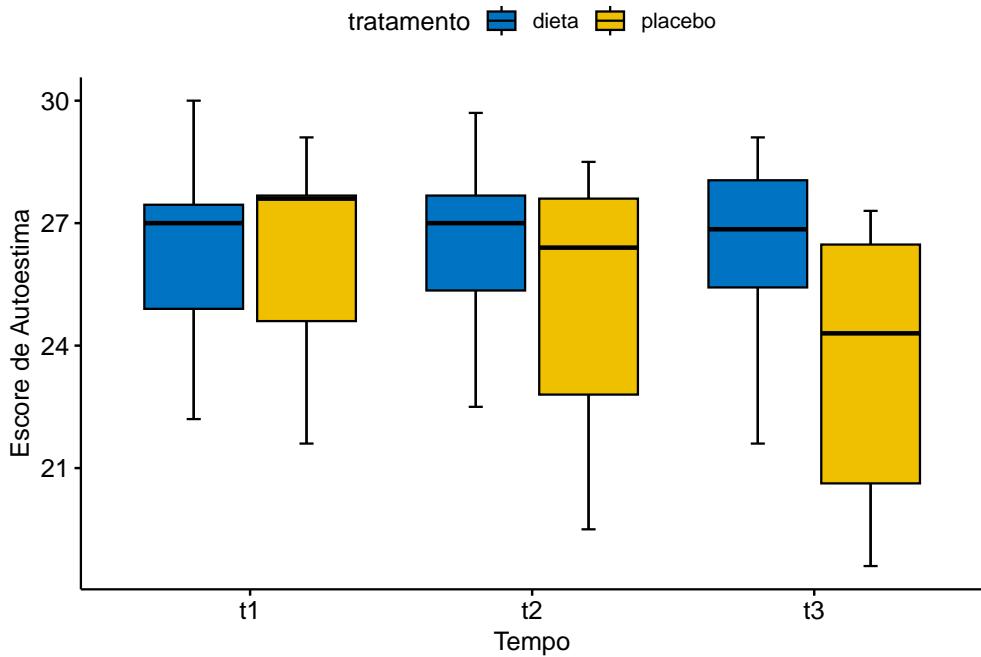


Figura 161: Impacto de uma dieta específica na autoestima.

14.2.6 Avaliação dos pressupostos

Os pressupostos são os mesmos da ANOVA de um fator.

14.2.6.1 Identificação dos outliers A observação dos boxplots mostra que não existem valores atípicos. Estes *outliers* seão analisados, usando função `identify_outliers()` do pacote `rstatix`.

```

autoestimaL %>%
  group_by(tratamento, tempo) %>%
  identify_outliers(escore)

```

```

## [1] tratamento tempo      id      escore      is.outlier is.extreme
## <0 linhas> (ou row.names de comprimento 0)

```

A saída confirma a ausências de valores atípicos.

14.2.6.2 Avaliação da normalidade Para testar a hipótese de normalidade dos dados, será utilizado o teste de Shapiro-Wilk através da função `shapiro_test()`, do pacote `rstatix` e a função `group_by()`, incluída no pacote `dplyr` ou `rstatix`, junto com o operador `pipe (%>%)`:

```
autoestimaL %>%
  group_by (tratamento, tempo) %>%
  shapiro_test(escore)
```

```
## # A tibble: 6 x 5
##   tratamento tempo variable statistic     p
##   <fct>      <fct>  <chr>       <dbl>   <dbl>
## 1 dieta       t1     escore      0.919  0.279
## 2 dieta       t2     escore      0.923  0.316
## 3 dieta       t3     escore      0.886  0.104
## 4 placebo     t1     escore      0.828  0.0200
## 5 placebo     t2     escore      0.868  0.0618
## 6 placebo     t3     escore      0.887  0.107
```

Os resultados da Saída mostram que os escores de autoestima estão normalmente distribuídos em cada momento, havendo uma exceção: o grupo placebo, no momento t1.

É possível construir um gráfico QQ (Figura 162) para cada um dos momentos, usando a função `ggqqplot()` do pacote `ggpubr`, consulte a vinheta do pacote para maiores detalhes. Foi utilizado também a função `facet_grid()`, do `ggplot2`, que divide em painéis, organizando-os como uma grade, de acordo com o momento (t1, t2 e t3).

```
ggqqplot(data = autoestimaL,
          x = "escore",
          color = "tempo",
          palette = get_palette("Dark2", 3),
          legend = "none",
          ggtheme = theme_bw()
        ) +
  facet_grid(tempo~tratamento)
```

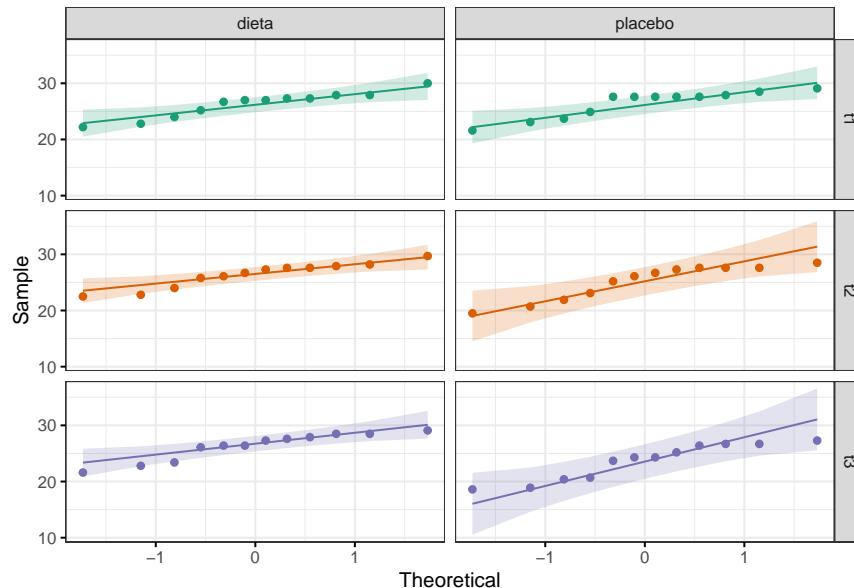


Figura 162: Gráfico QQ para verificar a normalidade

Observando o gráfico, como quase todos os pontos caem aproximadamente ao longo da linha de referência, pode-se seguir a análise, pois não há muito problema.

14.2.6.3 Esferecideade

A esferecideade será avaliada junto com a construção do modelo.

14.2.7 Cálculo da ANOVA de medidas de repetidas

É realizado da mesma maneira do que a ANOVA de medidas repetidas de uma via:

```
mod.anova2 <- anova_test(data = autoestimaL,
                           dv = escore,
                           wid = id,
                           within = c(tratamento, tempo))
mod.anova2

## ANOVA Table (type III tests)
##
## $ANOVA
##          Effect DFn DFd      F      p p<.05   ges
## 1        tratamento  1  11 15.541 2.00e-03    * 0.059
## 2            tempo   2  22 27.369 1.08e-06    * 0.049
## 3 tratamento:tempo  2  22 30.424 4.63e-07    * 0.050
##
## $`Mauchly's Test for Sphericity`
##          Effect      W      p p<.05
## 1            tempo 0.469 0.023    *
## 2 tratamento:tempo 0.616 0.089
##
## $`Sphericity Corrections`
##          Effect   GGe      DF[GG]      p[GG] p[GG]<.05   HFe      DF[HF]
## 1            tempo 0.653 1.31, 14.37 5.03e-05    * 0.705 1.41, 15.52
## 2 tratamento:tempo 0.723 1.45, 15.9 1.25e-05    * 0.803 1.61, 17.66
##          p[HF] p[HF]<.05
## 1 2.81e-05    *
## 2 4.82e-06    *

get_anova_table(mod.anova2)

## ANOVA Table (type III tests)
##
##          Effect DFn DFd      F      p p<.05   ges
## 1        tratamento 1.00 11.00 15.541 2.00e-03    * 0.059
## 2            tempo 1.31 14.37 27.369 5.03e-05    * 0.049
## 3 tratamento:tempo 2.00 22.00 30.424 4.63e-07    * 0.050
```

Existe uma interação estatisticamente significativa entre o tratamento e o tempo, $F(2, 22) = 30,4$, $p < 0,0001$.

14.2.8 Teste post hoc

Uma *interação significativa entre os dois fatores* indica que o impacto que um fator (tratamento) tem na variável desfecho (escore de autoestima) depende do nível do outro fator (tempo”), e vice-versa. Assim, é possível decompor a interação entre os dois fatores significativa em:

- *Efeito principal simples*: executar o modelo unifatorial da primeira variável (tratamento) em cada nível da segunda variável (tempo),
- *Comparações simples pareadas*: se o efeito principal simples for significativo, executar várias comparações pareadas para determinar quais grupos são diferentes.

Para uma *interação não significativa entre os dois fatores*, há necessidade de determinar se tem algum *efeito principal* estatisticamente significativo da saída ANOVA.

14.2.8.1 Procedimento para uma interação significativa entre os dois fatores

Inicialmente, será verificado o

Efeito do tratamento

No exemplo, será analisado o efeito do tratamento no escore de auto-estima em cada momento no tempo. Note que como o tratamento tem apenas dois níveis (dieta e placebo), o teste de ANOVA e teste *t* pareado fornecem os mesmos resultados.

Efeito do tratamento em cada ponto de tempo

```
tratamento <- autoestimaL %>%
  group_by(tempo) %>%
  anova_test(dv = escore, wid = id, within = tratamento) %>%
  get_anova_table() %>%
  adjust_pvalue(method = "bonferroni")
tratamento

## # A tibble: 3 x 9
##   tempo Effect      DFn     DFd       F     p `p<.05`     ges     p.adj
##   <fct> <chr>    <dbl>    <dbl>    <dbl> <dbl> <chr>    <dbl>    <dbl>
## 1 t1   tratamento 1     11  0.376  0.552   ""    0.000767 1
## 2 t2   tratamento 1     11  9.03   0.012   "*"   0.052    0.036
## 3 t3   tratamento 1     11 30.9   0.00017 **    0.199    0.00051
```

Comparações pareadas entre os grupos de tratamentos

```
pwc <- autoestimaL %>%
  group_by(tempo) %>%
  pairwise_t_test(escore ~ tratamento,
                  paired = TRUE,
                  p.adjust.method = "bonferroni")
pwc

## # A tibble: 3 x 11
##   tempo .y. group1 group2     n1     n2 statis~1      df      p     p.adj p.adj~2
##   * <fct> <chr> <chr> <int> <int>    <dbl> <dbl> <dbl> <dbl> <chr>
## 1 t1   escore dieta placebo    12     12   -0.613    11  0.552   0.552   ns
## 2 t2   escore dieta placebo    12     12     3.00    11  0.012   0.012   *
## 3 t3   escore dieta placebo    12     12     5.56    11  0.00017 0.00017 ***
## # ... with abbreviated variable names 1: statistic, 2: p.adj.signif
```

Considerando o valor P ajustado de Bonferroni ($p.adj$), pode-se observar que o efeito principal simples do tratamento não foi significativo no ponto de tempo t1 ($P = 1$). Torna-se significativo em t2 ($p = 0,036$) e t3 ($p = 0,00051$).

Comparações pareadas mostram que o escore médio de autoestima foi significativamente diferente entre o grupo placebo e dieta em t2 ($P = 0,12$) e t3 ($P = 0,00017$), mas não em t1 ($P = 0,55$).

Efeito do tempo

Observe que também é possível realizar a mesma análise para a variável `tempo` em cada nível de tratamento. Esta análise, necessariamente, não precisa ser feita!

Efeito do tempo em cada nível de tratamento

```
tempo <- autoestimaL %>%
  group_by(tratamento) %>%
  anova_test(dv = escore, wid = id, within = tempo) %>%
  get_anova_table() %>%
  adjust_pvalue(method = "bonferroni")
tempo

## # A tibble: 2 x 9
##   tratamento Effect  DFn    DFd      F      p `p<.05` ges     p.adj
##   <fct>      <chr> <dbl> <dbl>    <dbl>    <dbl> <chr>    <dbl>    <dbl>
## 1 dieta       tempo     2     22  0.078  0.925     ""    0.000197 1
## 2 placebo     tempo     2     22 39.7   0.00000005 **     0.145   0.0000001
```

Comparações pareadas entre pontos no tempo

```
pwc2 <- autoestimaL %>%
  group_by(tratamento) %>%
  pairwise_t_test(
    escore ~ tempo, paired = TRUE,
    p.adjust.method = "bonferroni"
  )
pwc2

## # A tibble: 6 x 11
##   tratam~1 .y.  group1 group2    n1    n2 statis~2    df      p  p.adj p.adj~3
##   * <fct>   <chr> <chr> <chr> <int> <int>    <dbl> <dbl>    <dbl> <dbl> <chr>
## 1 dieta    esco~ t1    t2     12    12 -0.522    11 6.12e-1 1  e+0 ns
## 2 dieta    esco~ t1    t3     12    12 -0.102    11 9.21e-1 1  e+0 ns
## 3 dieta    esco~ t2    t3     12    12  0.283    11 7.82e-1 1  e+0 ns
## 4 placebo   esco~ t1    t2     12    12  4.53     11 8.58e-4 3  e-3 **
## 5 placebo   esco~ t1    t3     12    12  6.91     11 2.55e-5 7.65e-5 ****
## 6 placebo   esco~ t2    t3     12    12  6.49     11 4.49e-5 1.35e-4 ***
## # ... with abbreviated variable names 1: tratamento, 2: statistic,
## #   3: p.adj.signif
```

Após a execução do código, verifica-se que o efeito do tempo é significativo apenas para o placebo, $F(2, 22) = 39,7$, $p < 0,0001$. As comparações pareadas mostram que todas as comparações entre os pontos de tempo foram estatisticamente significativas para o placebo.

14.2.8.2 Procedimento para uma interação não significativa entre os dois fatores Se a interação não for significativa, é preciso interpretar os efeitos principais para cada uma das duas variáveis: “tratamento” e “tempo”. Um efeito principal significativo pode ser acompanhado com comparações pareadas.

No exemplo, (consulte a tabela ANOVA em `mod.anova2`), houve efeitos principais estatisticamente significativos do tratamento ($F(1, 11) = 15,5, P = 0,002$) e tempo ($F(2, 22) = 27,4, p < 0,0001$) no escore de autoestima.

Comparações para a variável tratamento

```
autoestimaL %>%
  pairwise_t_test(escore ~ tratamento,
                  paired = TRUE,
                  p.adjust.method = "bonferroni")

## # A tibble: 1 x 10
##   .y.    group1 group2     n1     n2 statistic      df      p    p.adj p.adj.si~1
## * <chr>  <chr>  <chr>  <int>  <int>     <dbl> <dbl>  <dbl>  <dbl> <chr>
## 1 escore dieta placebo     36     36     4.35     35 0.000113 0.000113 ***
## # ... with abbreviated variable name 1: p.adj.signif
```

Comparações para a variável tempo

```
autoestimaL %>%
  pairwise_t_test(escore ~ tempo,
                  paired = TRUE,
                  p.adjust.method = "bonferroni")

## # A tibble: 3 x 10
##   .y.    group1 group2     n1     n2 statistic      df      p    p.adj p.adj.sig~1
## * <chr>  <chr>  <chr>  <int>  <int>     <dbl> <dbl>  <dbl>  <dbl> <chr>
## 1 escore t1     t2       24     24     2.86     23 0.009 0.027 *
## 2 escore t1     t3       24     24     3.70     23 0.001 0.004 **
## 3 escore t2     t3       24     24     3.75     23 0.001 0.003 **
```

14.2.9 Relatando os resultados da ANOVA de medidas repetidas de dois fatores

O resultado pode ser relatado da seguinte forma:

Uma ANOVA de medidas repetidas de dois fatores foi realizada para avaliar o efeito de diferentes tratamentos dietéticos ao longo do tempo no escore de autoestima.

Houve uma interação estatisticamente significativa entre “tratamento” e “tempo” no escore de autoestima, $F(2, 22) = 30,4, p < 0,0001$. Portanto, o efeito da variável “tratamento” foi analisado em cada ponto de tempo. Os valores de P foram ajustados usando o método de correção de testes múltiplos de Bonferroni. O efeito da variável “tratamento” foi significativo em t2 ($P = 0,036$) e t3 ($P = 0,00051$), mas não no ponto de tempo t1 ($P = 1$).

Comparações pareadas, usando o teste t pareado, mostram que o escore médio de autoestima foi significativamente diferente entre os ensaios placebo e dieta nos pontos de tempo t2 ($P = 0,012$) e t3 ($P = 0,00017$), mas não em t1 ($P = 0,55$).

```

bxp <- ggboxplot (autoestimaL,
                   bxp.errorbar = TRUE,
                   bxp.errorbar.width = 0.1,
                   x = "tempo",
                   y = "escore",
                   color = "black",
                   fill = "tratamento",
                   ylab = "Escore de Autoestima",
                   xlab = "Tempo")+
  scale_fill_grey(start=0.95, end=0.6) +
  theme(legend.position="right") +
  theme (text = element_text (size = 12))

pwc <- pwc %>% add_xy_position(x = "tempo")
bxp +
  stat_pvalue_manual(pwc,
                     label = "p.adj",
                     tip.length = 0.01,
                     hide.ns = FALSE,
                     y.position = c(32, 32, 32)) +
  labs(subtitle = get_test_label(mod.anova2, detailed = TRUE),
       caption = get_pwc_label(pwc))

```

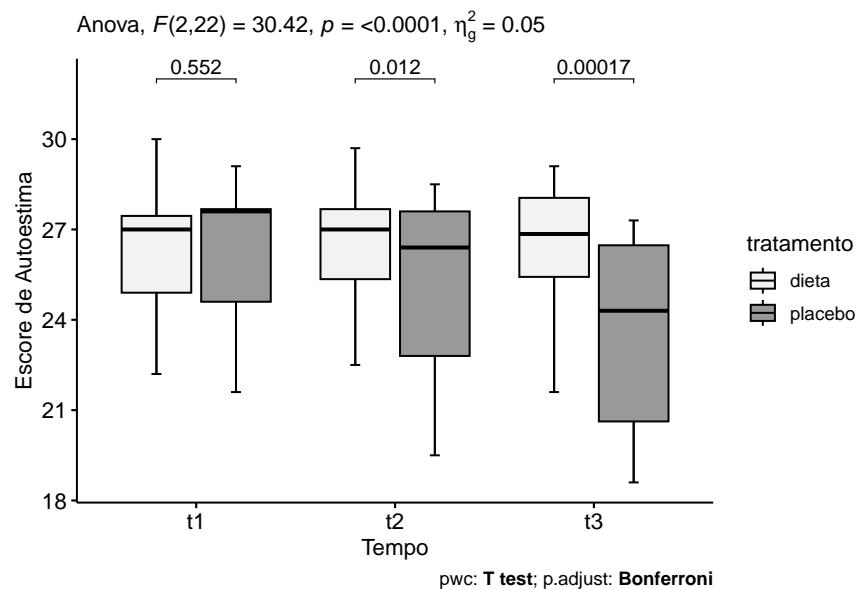


Figura 163: Avaliação da autoestima no decorrer do tempo

14.2.9.1 Visualização: boxplots com valores P (Figura 163)

15 Correlação e Regressão

15.1 Correlação

A correlação é usada para avaliar a força e a direção da relação entre duas variáveis numéricas contínuas, normalmente distribuídas. A maneira mais comum de mostrar a relação entre duas variáveis quantitativas é através de um diagrama ou gráfico de dispersão (*scatterplot*). A Figura 164 exibe um exemplo de um gráfico de dispersão, onde se observa um padrão geral que sugere uma relação entre o estriol urinário (mg/24h) e o peso fetal em uma gravidez normal (127).

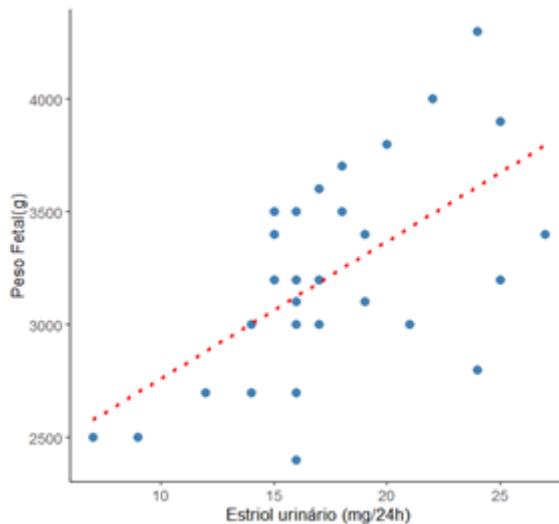


Figura 164: Correlação da excreção de estriol urinário e peso fetal

O gráfico de dispersão mostra que os valores de uma variável aparecem no eixo horizontal 'x' e os valores da outra variável aparecem no eixo vertical y. Cada indivíduo nos dados aparece como o ponto no gráfico fixado pelos valores de ambas as variáveis para aquele indivíduo. Normalmente, eixo x é a *variável explicativa* (ou variável explanatória ou independente) e y a *variável desfecho* (variável resposta ou dependente).

Em um diagrama de dispersão deve-se procurar o padrão geral e desvios marcantes desse padrão. Verifica-se o padrão geral, observando a direção, a forma e força do relacionamento. Um tipo importante de desvio é um valor atípico, um valor individual que está fora do padrão geral do relacionamento.

A Figura 164 mostra uma clara direção do padrão geral que se move da esquerda inferior para a direita superior. Este comportamento é denominado de *correlação positiva* entre as variáveis. A forma do relacionamento é aproximadamente uma linha reta com uma leve curva para a direita à medida que se move para cima. A força de uma correlação em um gráfico de dispersão é determinada pela proximidade dos pontos em uma forma clara. No caso, quanto mais se aproxima de uma reta, mais forte é a associação, no caso de uma correlação linear. Duas variáveis estão *negativamente associadas* quando se comportam de forma oposta ao da Figura 164.

Obviamente, nem todos os diagramas de dispersão mostram uma direção clara que permita descrever como correlação positiva ou negativa e não tem uma forma linear, sugerindo que não há correlação, como a Figura 165.

15.1.1 Coeficiente de correlação de Pearson

A correlação é quantificada pelo *Coeficiente de Correlação Linear de Pearson*. Este coeficiente paramétrico, denotado por r , é um número adimensional, independente das unidades usadas para medir as variáveis x e

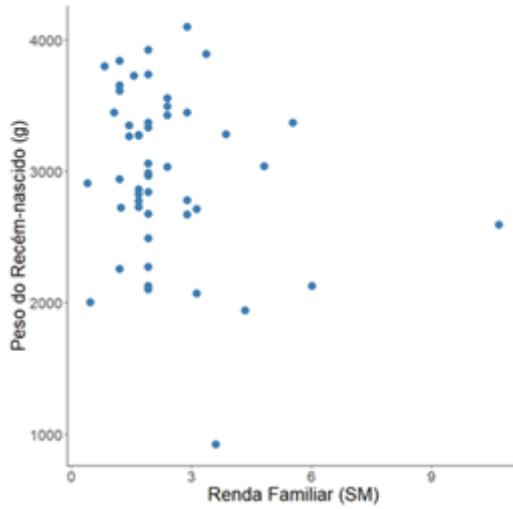


Figura 165: Gráfico de dispersão sugerindo ausência de correlação.

y .

Suponha que se tenha dados sobre as variáveis x e y para n indivíduos. Os valores para o primeiro indivíduo são x_1 e y_1 , os valores para o segundo indivíduo são x_2 e y_2 e assim por diante. As médias e desvios padrão das duas variáveis são \bar{x} e s_x para os valores de x e \bar{y} e s_y para os valores de y . A correlação r entre x e y é dada pela equação:

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2} \times \sqrt{\sum (y_i - \bar{y})^2}}$$

O Coeficiente de Correlação, r , apresenta as seguintes características:

- É um valor numérico que varia de -1 a +1 (Figura 166):
 - Quando $r = -1$, há uma correlação linear negativa ou inversa perfeita;
 - Quando $r = +1$, há uma correlação linear positiva ou direta perfeita;
 - Quando $r = 0$, não há correlação entre as variáveis.

- Quanto mais os pontos se aproximam de uma linha reta, maior a magnitude de r .
- O coeficiente de correlação r é calculado para uma amostra e é uma estimativa do coeficiente de correlação da população ρ (leia-se rô).
- A correlação não faz distinção entre variáveis explicativas e variáveis resposta. Apesar de haver uma recomendação para que x seja a variável explanatória e y a variável desfecho. Não faz diferença qual variável será chamada chama de x e qual de y no cálculo da correlação.
- Como r usa os valores padronizados das observações, r não muda se as unidades de medida de x , y ou ambos são modificados. A correlação r em si não tem unidade de medida; é apenas um número.

15.1.2 Pacotes necessários

```
pacman::p_load(readxl,
                 dplyr,
                 rstatix,
```

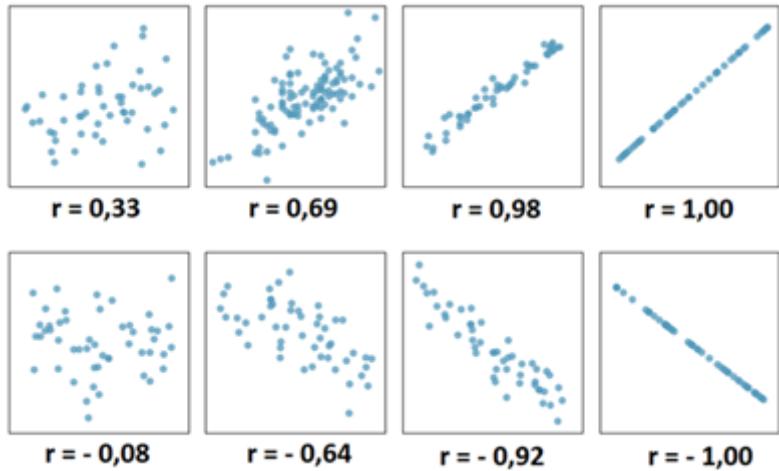


Figura 166: Coeficiente de Correlação.

```
ggplot2,
ggpubr,
ggsci,
car,
lmtest,
kableExtra)
```

15.1.3 Dados do exemplo

Está claro que existe uma relação entre a idade de crianças e a sua altura (comprimento). Vamos usar os dados coletados em um ambulatório pediátrico de 40 crianças entre 18 e 36 meses (20 meninos e 20 meninas). Os dados estão no banco de dados `dadosReg.xlsx`. Para baixar o banco de dados, clique [aqui](#). Salve o arquivo no seu diretório de trabalho.

15.1.3.1 Leitura e visualização dos dados O conjunto de dados será atribuído ao objeto `dados`.

```
dados <- read_excel("dadosReg.xlsx")
```

Observe os dados com a função `glimpse()`, do pacote `dplyr`:

```
glimpse(dados)
```

```
## #> #> Rows: 40
## #> Columns: 5
## #> $ id      <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, ~
## #> $ idade   <dbl> 18, 18, 19, 19, 20, 20, 21, 21, 22, 22, 23, 23, 24, 24, 25, 25, ~
## #> $ comp    <dbl> 80.0, 80.0, 83.0, 82.0, 84.0, 81.0, 84.5, 84.0, 85.0, 82.5, 86.~
## #> $ irmaos  <dbl> 0, 0, 2, 0, 0, 1, 1, 0, 1, 2, 0, 1, 0, 0, 0, 0, 1, 0, 1, 1, ~
## #> $ sexo    <chr> "masc", "fem", "masc", "fem", "masc", "fem", "masc", "fem", "ma~
```

De acordo com uma das exigências da correlação, as variáveis `idade` e `comp` pertencem a classe das variáveis numéricas. A variável `sexo` foi lida como um variável numérica e será transformada em fator:

```
dados$sexo <- as.factor(dados$sexo)
```

Uma das exigências da correlação é de que as variáveis sejam numéricas e, portanto, não será necessário nenhum tipo de transformação.

15.1.3.2 Medidas resumidoras As medidas resumidoras serão calculadas, usando a função `get_summary_stats()` do pacote `rstatix` que necessita dos seguintes argumentos:

```
dados %>%
  get_summary_stats(idade,
    comp,
    type = "mean_sd")
```



```
## # A tibble: 2 x 4
##   variable     n   mean     sd
##   <fct>     <dbl> <dbl> <dbl>
## 1 idade       40   27.0  5.41
## 2 comp        40   90.2  6.00
```

15.1.3.3 Visualização dos dados Aqui, será usado o gráfico de dispersão (Figura 167), usando a função `geom_point` do pacote `ggplot2`:

```
ggplot(dados,
  aes(x=idade, y=comp, color = sexo)) +
  geom_point(size = 3,
    shape = 19) +
  xlab("Idade (meses)") +
  ylab ("Comprimento(cm)") +
  theme_classic() +
  theme(text = element_text(size = 12,
    color = NULL,
    face = "bold"))
```

A separação dos pontos por sexo não muda a análise e foi realizada apenas para treinamento.

15.1.4 Pressupostos da Correlação

A primeira e mais importante etapa antes de analisar os dados, usando a correlação de Pearson é verificar se é apropriado usar este teste estatístico.

Vamos discutir sete pressupostos, três estão relacionados com o projeto do estudo e como as variáveis foram medidas (pressuposto 1, 2 e 3) e quatro que se relacionam com as características dos dados (pesupostos 4, 5, 6 e 7) (128).

1. Variáveis numéricas contínuas

As duas variáveis devem ser medidas em uma escala contínua (são medidas no nível intervalar ou de razão). No exemplo, tanto a variável `idade` como o comprimento (`comp`) são variáveis contínuas, como verificado acima.

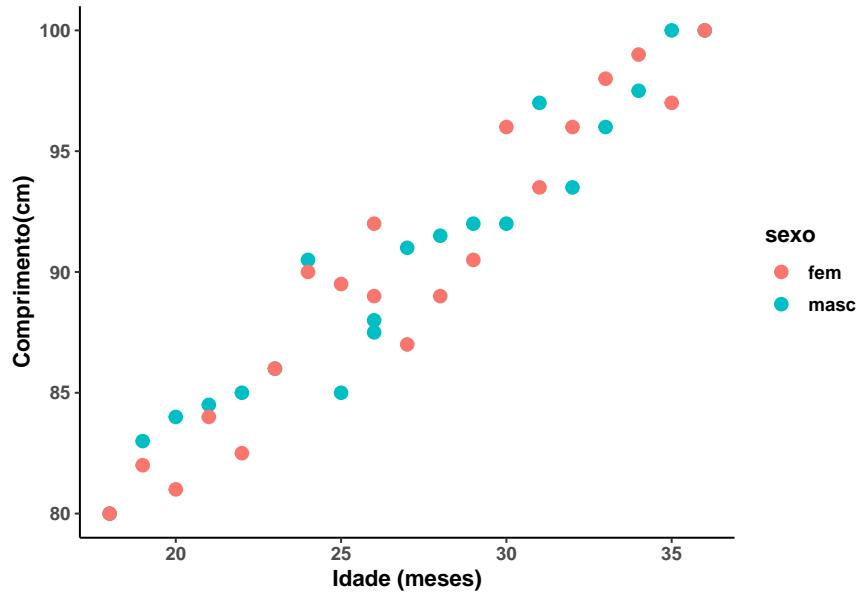


Figura 167: Correlação entre a idade e a altura de uma criança.

2. Variáveis devem estar como pares

As duas variáveis contínuas devem ser emparelhadas, o que significa que cada caso (por exemplo, cada participante) tem dois valores: um para cada variável.

3. Independência das observações

Deve haver independência de casos, o que significa que as duas observações para um caso (por exemplo, a idade e o comprimento) devem ser independentes das duas observações para qualquer outro caso.

Se estes pressupostos forem atendidos, avalia-se os outros pressupostos:

4. Relação linear entre as variáveis

O coeficiente de correlação de Pearson é uma medida da força de uma associação linear entre duas variáveis. Dito de outra forma, ele determina se há um componente linear de associação entre duas variáveis contínuas. Por esse motivo, verifica-se a relação entre duas variáveis, em um gráfico de dispersão, para ver se a execução de uma correlação de Pearson é a melhor escolha como medida de associação.

A variável `idade` é colocada como variável preditora (eixo x) e `comp` como desfecho (eixo y). O gráfico de dispersão anterior, mostra uma nítida correlação linear.

5. Normalidade das variáveis

Para verificar se as variáveis têm distribuição normal, é possível usar o teste de Shapiro-Wilk, usando a função `shapiro.test()`:

```
shapiro.test(dados$idade)
```

```

##  

## Shapiro-Wilk normality test  

##  

## data: dados$idade  

## W = 0.9582, p-value = 0.1453

```

```
shapiro.test(dados$comp)
```

```

##  

## Shapiro-Wilk normality test  

##  

## data: dados$comp  

## W = 0.95782, p-value = 0.141

```

O teste de Shapiro-Wilk de ambas as variáveis retorna um valor $P > 0,05$, indicando que não é possível rejeitar a H_0 ; os dados seguem a distribuição normal, portanto o pressuposto foi atendido.

O teste estatístico fornece uma prova, mas também pode-se fazer uma verificação visual rápida da normalidade dos dados, usando a função `ggqqplot()` do pacote `ggpubr` que produz um gráfico quantil-quantil, onde os pontos devem cair próximos da linha de referência (Figuras 168 e 169).

```
ggqqplot(dados,  

         x = "idade",  

         color = "steelblue",  

         xlab = "Quantis normais",  

         ylab = "Idade (meses)")
```

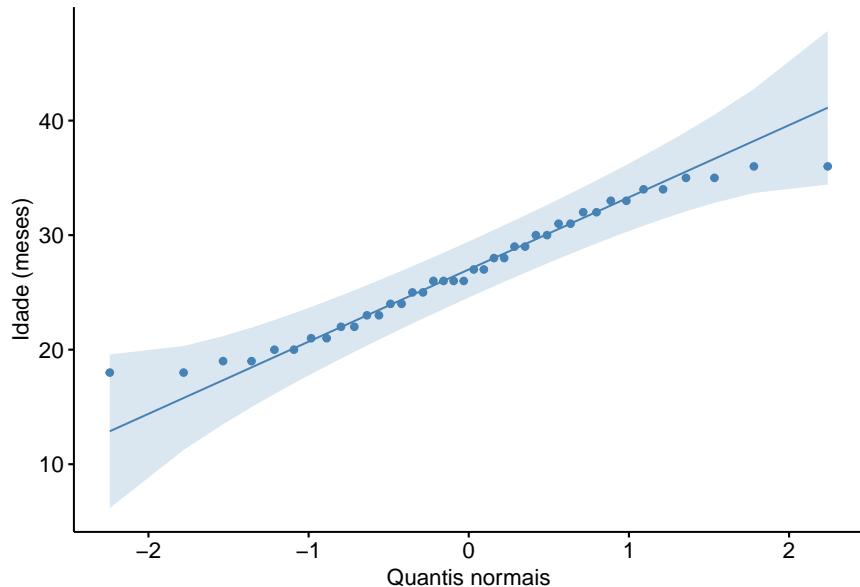


Figura 168: Gráfico QQ da idade.

```
ggqqplot(dados,  

         x = "comp",  

         color = "steelblue",
```

```
xlab = "Quantis normais",
ylab = "Idade (meses)"
```

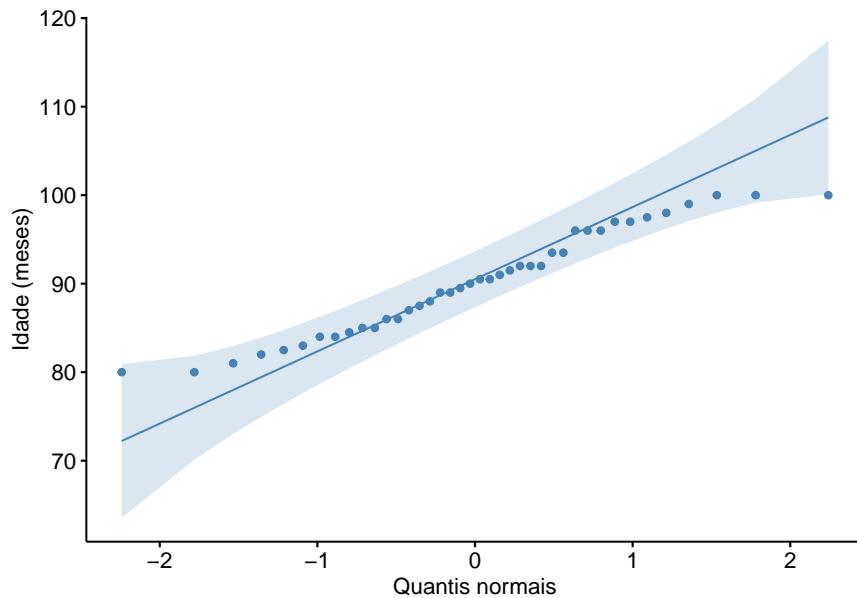


Figura 169: Gráfico QQ do comprimento

Como todos os pontos caem aproximadamente ao longo da linha de referência, pode-se assumir que ambas variáveis têm distribuição que se ajusta à normal.

6. Pesquisa de valores atípicos

A presença de *outliers* pode ser verificada construindo boxplots (Figura 170) com as variáveis `idade` e `comp`

```
par (mfrow=c (1,2))
boxplot(dados$idade,
        ylab = "Idade (meses)",
        col = "lightgreen")
boxplot(dados$comp,
        ylab = "Comprimento (cm)",
        col = "salmon")
```

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

Não se observa valores atípicos nos boxplots. Isto também pode ser verificado com as estatísticas dos boxplots que retorna:

- *stats*: um vetor de tamanho 5, contendo o *whisker* inferior, 1º quartil, a mediana, a 3º quartil e o *whisker* superior;
- *n*: o número de observações na amostra;
- *conf*: os extremos inferior e superior do *entalhe*. Os entalhes (se solicitados) se estendem a $\pm 1,58 \times IQR/\sqrt{n}$;
- *out*: os valores de quaisquer pontos de dados que estão além dos extremos dos *whisker*.

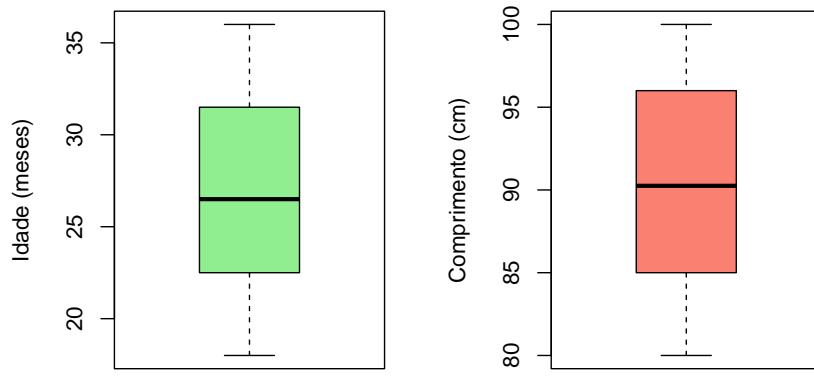


Figura 170: Pesquisa de valores atípicos com boxplots

```
boxplot.stats (dados$idade)
```

```
## $stats
## [1] 18.0 22.5 26.5 31.5 36.0
##
## $n
## [1] 40
##
## $conf
## [1] 24.25162 28.74838
##
## $out
## numeric(0)
```

```
boxplot.stats (dados$comp)
```

```
## $stats
## [1] 80.00 85.00 90.25 96.00 100.00
##
## $n
## [1] 40
##
## $conf
## [1] 87.50198 92.99802
##
## $out
## numeric(0)
```

7. Homoscedasticidade

A homocedasticidade assume que os dados são igualmente distribuídos sobre a linha de regressão. Descreve uma situação na qual o resíduo é o mesmo em todos os valores das variáveis independentes. A heterocedasticidade (a violação da homocedasticidade) está presente quando o tamanho dos resíduos difere entre os valores de uma variável independente.

O impacto de violar o pressuposto da homocedasticidade é uma questão de grau, aumentando à medida que a heterocedasticidade aumenta. Desse dorma, avalia-se a homocedasticidade, observando os resíduos.

Uma correlação linear pode ser descrita por uma reta. Em uma correlação linear perfeita, a reta passa por todos os pontos. Normalmente, não é possível traçar uma reta que passe por todos os pontos. A melhor reta é aquela que promove o melhor ajuste, ou seja, é aquela cuja distância dos pontos até a reta é a menor possível. Os **resíduos** são a diferença entre o valor observado e o valor previsto pelo melhor ajuste, estabelecido pelo modelo de regressão linear.

Construção do modelo:

```
mod_reg <- lm(comp ~ idade, dados)
```

Análise gráfica da homoscedasticidade (Figura 171):

```
par(mfrow=c(1,2))
plot(mod_reg, which=c(1,3))
```

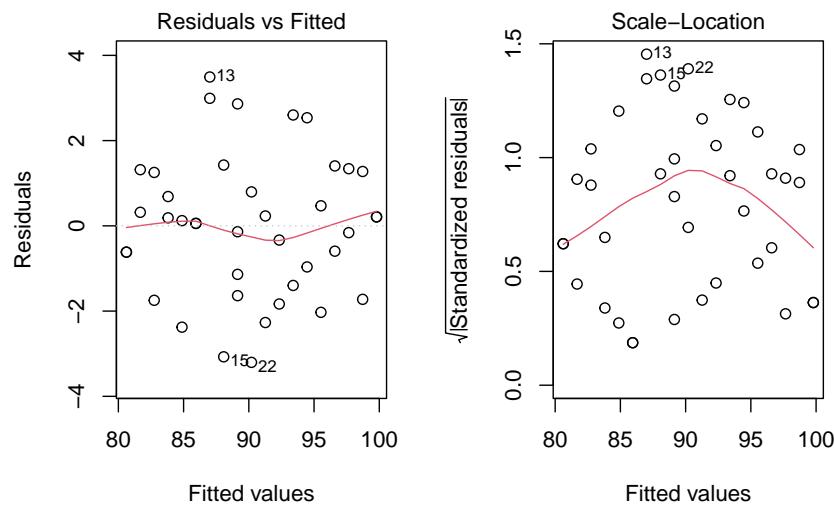


Figura 171: Gráficos diagnósticos 1 e 3

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

Os gráficos de diagnóstico dos resíduos são apresentados de quatro maneiras diferentes e, para maiores detalhes consulte [aqui](#). O primeiro e terceiro tipo serão avaliados:

Gráfico 1: *valores previstos × resíduos*

No gráfico 1, tem-se os resíduos em função dos valores estimados. Pode-se utilizar este gráfico para observar a independência e a homocedasticidade, se os resíduos se distribuem de maneira razoavelmente aleatória e com mesma amplitude em torno do zero.

Gráfico 3: valores previstos × resíduos padronizados

No gráfico 3, pode ser avaliado da mesma maneira que o 1, observando a aleatoriedade e amplitude, desta vez dos resíduos padronizados. É bom se uma linha horizontal é observada com pontos de distribuição igualmente distribuídos (aleatoriamente), o que não é o caso. Permite verificar se há *outliers* - valores de resíduos padronizados acima de 3 ou abaixo de -3.

Embora o gráfico possa dar uma ideia sobre homocedasticidade, às vezes, um teste mais formal é preferido. Existem vários testes para isso, mas aqui será utilizado o *Teste de Breusch-Pagan*. A H_0 e a H_A podem ser consideradas como:

H_0 : Homocedasticidade. Os resíduos têm variância constante sobre o modelo verdadeiro.

H_A : Heterocedasticidade. Os resíduos não têm variância constante sobre o modelo verdadeiro.

Se o valor $P > 0,05$ não se rejeita a H_0 de homocedasticidade. O teste de Breusch-Pagan é encontrado na função `bptest()`, incluída no pacote `lmtest`:

```
bptest(mod_reg)
```

```
##  
## studentized Breusch-Pagan test  
##  
## data: mod_reg  
## BP = 0.049988, df = 1, p-value = 0.8231
```

Os resultados não indicam heteroscedasticidade e isso é bom. Desta forma, pode-se aplicar a equação final de predição.

15.1.5 Execução do teste de correlação

1. Coeficiente de correlação de Pearson (r)

- O coeficiente de correlação, r , é calculado para uma amostra e é uma estimativa do coeficiente de correlação da população ρ ($r\hat{\rho}$).
- A correlação não faz distinção entre variáveis explicativas e variáveis resposta. Apesar de haver uma recomendação para que x seja a variável explanatória e y a variável desfecho. Não faz diferença qual variável será chamada chama de x e qual de y no cálculo da correlação.
- Como o r usa os valores padronizados das observações, não muda nada se as unidades de medida de x , y ou ambas são modificadas. A correlação r em si não tem unidade de medida; é apenas um número.

O cálculo pode ser realizado com a função `cor.test()` do R base que usa os seguintes argumentos:

Argumento	Significado
<code>x</code>	dados da variável x ;
<code>y</code>	dados da variável y ;
<code>method</code>	Pode ser usado dos seguintes “pearson”, “kendall” ou “spearman”; exact
<code>alternative</code>	hipótese alternativa “two.sided” (bilateral) ou “greater” ou “less” (unilateral a direita ou a esquerda, respectivamente);
<code>conf.level</code>	nível de confiança. Padrão 0.95.

```

r <- cor.test(dados$idade,
              dados$comp,
              method = "pearson")
r

##
## Pearson's product-moment correlation
##
## data: dados$idade and dados$comp
## t = 21.445, df = 38, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.9271449 0.9793782
## sample estimates:
##        cor
## 0.9610805

```

Interpretação do resultado

A saída do Coeficiente de Correlação de Pearson (r) é igual 0.96 (IC95%: 0.93, 0.98) o que corresponde a uma correlação linear muito forte (Tabela 16, ignorando o sinal) entre a idade e o comprimento de crianças (129).

O coeficiente refere a existência de correlação linear, mas não especifica se a relação é de causa e efeito. O valor P especifica se a correlação é igual a zero (H_0) ou diferente de zero (H_A). No caso, ela é diferente de zero.

O importante é a magnitude do r , entretanto, o coeficiente r e o valor P devem ser interpretados em conjunto. Se o valor $P > 0,05$, mesmo que r seja diferente de zero, a correlação não deveria ser interpretada.

Tabela 16: Interpretação do Coeficiente de Correlação

Coeficiente de Correlação	Interpretação
0,0 < 0,3	desprezível
0,3 < 0,5	fraca
0,5 < 0,7	moderada
0,7 < 0,9	forte
0,9 < 1,0	muito forte
1,0	perfeita

Talvez a melhor maneira de interpretar a correlação linear é elevar o valor do r ao quadrado para obter o *Coeficiente de Determinação* (R^2). No exemplo usado, tem-se que o R^2 é igual a $0,96^2 = 0,922$, então, 92,2% da variação do comprimento da criança (y) podem ser explicados pela variação da sua idade (x), fato mais ou menos óbvio!.

2. Coeficiente de correlação de Spearman (ρ)

Se os pressupostos são violados é recomendado o uso de correlação não paramétrica, incluindo testes de correlação baseados em postos de Spearman e Kendall (130).

Usa-se a mesma função, usada para a correlação de Pearson, mudando o argumento `method`:

```

rho <- cor.test(dados$idade,
                 dados$comp,
                 method = "spearman")
rho

##
##  Spearman's rank correlation rho
##
## data:  dados$idade and dados$comp
## S = 448.01, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.957973

```

3. Coeficiente de correlação de Kendall (τ)

O coeficiente de correlação de postos de Kendall ou estatística tau de Kendall é usado para estimar uma medida de associação baseada em postos:

```

r <- cor.test(dados$idade,
               dados$comp,
               method = "kendall")
r

##
## Kendall's rank correlation tau
##
## data:  dados$idade and dados$comp
## z = 7.5731, p-value = 3.644e-14
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##      tau
## 0.8532019

```

15.2 Regressão Linear Simples

A regressão linear simples, assim como a correlação, é uma técnica usada para explorar a natureza da relação entre duas variáveis aleatórias contínuas. A principal diferença entre esses dois métodos analíticos é que a regressão permite investigar a alteração em uma variável, chamada *resposta*, correspondente a uma determinada alteração em outra, conhecida como variável *explicativa*. A regressão é um modelo matemático que permite a *predição* de uma variável resposta a partir de uma outra variável explicativa. A análise de correlação quantifica a força da relação entre as variáveis, tratando-as simetricamente (131).

A regressão linear simples é chamada assim, porque se tem apenas uma variável independente. Se houver mais de uma variável independente, é chamada de *regressão múltipla*.

A representação matemática do modelo de regressão linear populacional é descrita pela equação da reta de melhor ajuste em um conjunto de pares de dados (x, y) em um gráfico de dispersão de pontos.

$$y = \beta_0 + \beta_1 x$$

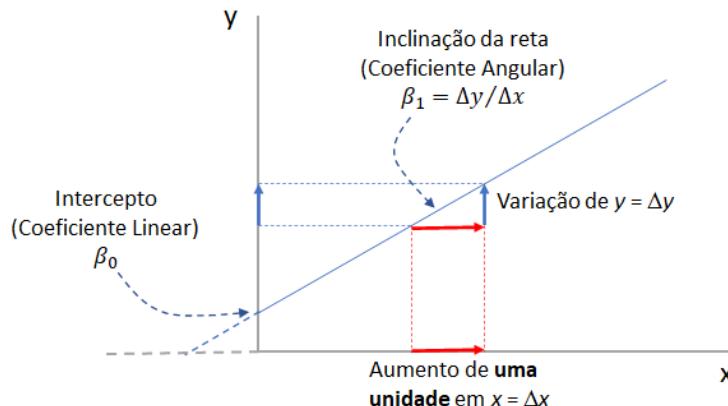


Figura 172: Reta de regressão, coeficiente angular e coeficiente linear.

A inclinação da reta de regressão (β_1) determina a variação de y para cada unidade de variação de x e recebe o nome de *coeficiente angular* ou *de regressão*. O ponto de interceptação da reta com y quando x é igual a zero é β_0 e é denominado de *coeficiente linear* (Figura 172). A equação da reta de regressão amostral que estima a reta de regressão populacional é igual a:

$$\hat{y} = b_0 + b_1 x$$

A reta do diagrama de dispersão da Figura 172 é a melhor reta de ajuste aos dados.

15.2.1 Resíduos

No exemplo usado no início desta seção, verificou-se que existe uma correlação linear entre a idade e o comprimento de crianças, usando uma amostra de 40 crianças entre 18 e 36 meses. A correlação de Pearson foi muito forte ($r = 0,96$, $P < 0,00001$). Esta relação linear pode ser descrita pela reta, mostrada na Figura 173.

```
ggplot(dados,
       aes(x = idade,
            y = comp,
            color = "tomato")) +
  geom_smooth(method = "lm",
              se = FALSE,
              color = "steelblue") +
  geom_point() +
  theme_classic() +
  xlab("Idade (meses)") +
  ylab("Comprimento (cm)") +
  theme(text = element_text(size = 12)) +
  theme(legend.position = "none")
```

Não é possível traçar uma reta que passe por todos os pontos. Esta reta ideal descreveria uma correlação perfeita, que não é o caso. Pode haver várias retas, a reta calculada pela regressão linear é aquela que promove o melhor ajuste, ou seja, é aquela cuja distância dos pontos até a reta é a menor possível.

Os resíduos são a diferença entre o valor observado e o valor previsto pelo modelo de regressão linear, construído anteriormente (`mod_reg`). A técnica estatística para achar a melhor reta que ajusta um conjunto

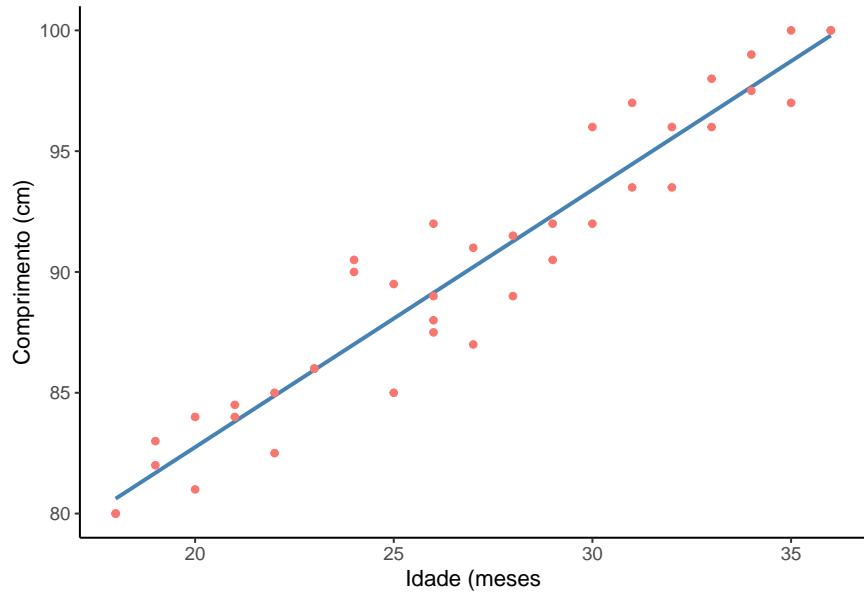


Figura 173: Reta de regressão

de dados é denominada de método dos mínimos quadrados (*Ordinary Least Square*). A melhor reta ajustada é aquela em que a soma dos quadrados da distância de cada ponto (soma dos quadrados residual) em relação à reta é minimizada.

Para se obter os resíduos, graficamente, pode ser usar os seguintes comandos que resultam na Figura 174.

```
# Obter e salvar os valores preditos e residuais
dados$previsto <- predict(mod_reg)

dados$residuos <- residuals(mod_reg)

# Construção do gráfico com os resíduos
ggplot(dados,
        aes(x = idade,
            y = comp)) +
  geom_smooth(method = "lm",
              se = FALSE,
              color = "steelblue") +
  geom_segment(aes(xend = idade,
                   yend = previsto),
               linewidth = 0.7,
               linetype = "dotted") +
  geom_point(aes(y = previsto),
             shape = 19,
             colour = "red") +
  geom_point() +
  theme_classic() +
  xlab("Idade (meses") +
  ylab("Comprimento (cm)") +
  theme(text = element_text(size = 12))
```

Uma boa maneira de testar a qualidade do ajuste do modelo é observar os resíduos (132) ou as diferenças

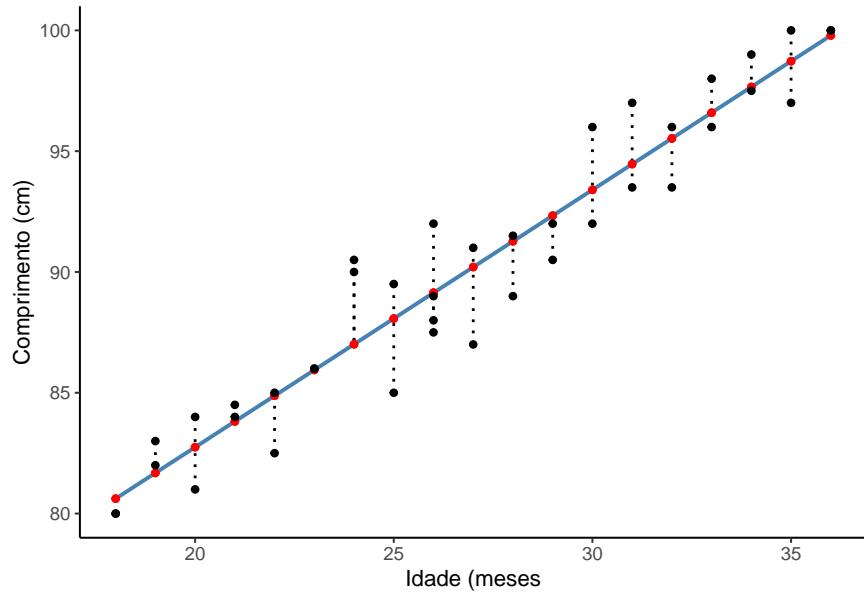


Figura 174: Resíduos

entre os valores reais (pontos pretos) e os valores previstos (pontos vermelhos). A reta de regressão, em azul no gráfico, representa os valores previstos. A linha vertical pontilhada da linhareta até o valor dos dados observados é o **resíduo**.

A ideia aqui é que a soma dos resíduos seja aproximadamente zero ou o mais baixo possível. Na vida real, a maioria dos casos não seguirá uma linha perfeitamente reta, portanto, resíduos são esperados. Na saída do resumo da função `lm()` em (`mod_reg$residuals`), você pode ver estatísticas descritivas sobre os resíduos do modelo (`residuals`), elas mostram como os resíduos são aproximadamente zero. Pode-se observar isso, usando a função `summary()` e `sum()`:

```
summary(mod_reg$residuals)
```

```
##      Min.   1st Qu.    Median     Mean   3rd Qu.   Max. 
## -3.20326 -1.20326  0.08994  0.00000  1.25849  3.49221
```

```
sum(mod_reg$residuals)
```

```
## [1] -1.69309e-15
```

Onde os resultados são praticamente iguais a zero.

15.2.2 Análise dos pressupostos do modelo de regressão

A análise exploratória do conjunto de dados foi feita quando do estudo da Correlação. Assim como a correlação, a regressão linear faz várias suposições sobre os dados.

15.2.2.1 Gráficos diagnósticos Os gráficos de diagnóstico da regressão (Figura 175) podem ser criados usando a função `plot()` do R base, como mostrado para a correlação:

O modelo de regressão, anteriormente criado, `mod_reg`, entra como argumento da função:

```
par(mfrow=c(2,2))
plot(mod_reg)
```

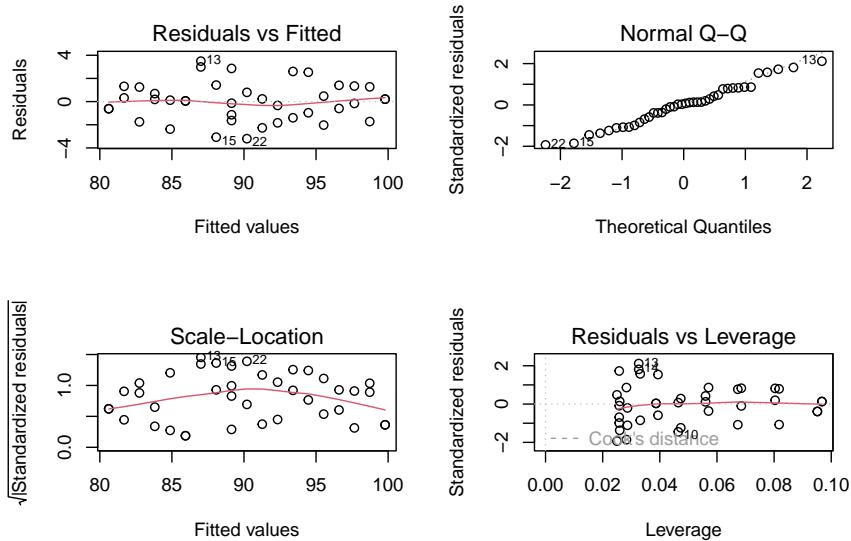


Figura 175: Gráficos diagnósticos

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

Os gráficos de diagnóstico mostram resíduos de quatro maneiras diferentes:

1. *Resíduos vs. ajustados (Residuals vs Fitted)*. Usado para verificar os pressupostos de relação linear. Uma linha horizontal, sem padrões distintos é um indicativo de uma relação linear, o que é bom. Os dados do exemplo (linha azul) afastam-se muito pouco do zero, mas a acompanham e não se observa nenhum padrão distinto, como uma parábola por exemplo.
2. *Q-Q plot*. Usado para examinar se os resíduos são normalmente distribuídos. É bom se os pontos residuais seguirem a linha reta tracejada. É possível dizer que os resíduos seguem a linha diagonal, com pequenos desvios toleráveis.
3. *Localização da dispersão (scale-location)*. Usado para verificar a homogeneidade de variância dos resíduos (homocedasticidade). Uma linha horizontal com pontos igualmente dispersos é uma boa indicação de homocedasticidade. No exemplo usado, os resíduos parecem estar dispersos e a linha azul não está próxima do zero, sugerindo um problema com a homocedasticidade, entretanto, não está acima de 3.
4. *Resíduos vs. alavancagem (leverage)*. Usado para identificar casos influentes, ou seja, valores extremos que podem influenciar os resultados da regressão quando incluídos ou excluídos da análise. Nem todos os *outliers* são influentes na análise de regressão linear. Mesmo que os dados tenham valores extremos, eles podem não ser influentes para determinar uma linha de regressão. Isso significa que os resultados não seriam muito diferentes, incluindo ou não esses valores. Por outro lado, alguns casos podem ser muito influentes, mesmo que pareçam estar dentro de uma faixa razoável de valores. Outra forma de colocar, é que eles não se entendem com a tendência na maioria dos casos. Ao contrário dos outros gráficos, desta vez os padrões não são relevantes. Deve-se estar atento aos valores distantes no canto superior direito ou no canto inferior direito. Esses pontos são os lugares onde os casos podem ter

influência contra uma linha de regressão. Procurar casos fora de uma linha tracejada, *distância de Cook*. Quando os casos estão fora da distância de Cook (o que significa que têm pontuações altas de distância de Cook), os casos são influentes para os resultados da regressão. Os resultados da regressão serão alterados se excluirmos esses casos.

A aparência dos gráficos do exemplo mostra que não há nenhum caso influente. Pouco se observa as linhas de distância de Cook (uma linha tracejada) porque todos os casos estão bem dentro das linhas de distância de Cook.

15.2.2.2 Avaliação da normalidade Além de observar o gráfico QQ plot, é possível realizar um teste estatístico de normalidade dos resíduos como, por exemplo o teste de Shapiro-Wilk. Ele pode ser executado, usando a função shapiro.test() do pacote stats, incluído no R base.

Ao ser criado o modelo de regressão (`mod_reg`), ele fornece uma série de variáveis que pode ser listada da seguinte maneira:

```
ls(mod_reg)
```

```
## [1] "assign"      "call"        "coefficients" "df.residual"
## [5] "effects"     "fitted.values" "model"       "qr"
## [9] "rank"        "residuals"    "terms"       "xlevels"
```

Usando a variável `residuals`, confirma-se o observado no QQPlot de que os resíduos apresentam distribuição normal, pois o valor de $P > 0,05$.

```
shapiro.test (mod_reg$residuals)
```

```
##
## Shapiro-Wilk normality test
##
## data: mod_reg$residuals
## W = 0.97906, p-value = 0.6547
```

15.2.2.3 Outliers nos resíduos Existe uma função pode ser usada para verificar valores atípicos nos resíduos da regressão para modelos lineares como `rstandard()` do pacote `stats`, que analisa os resíduos padronizados.

A função padroniza todos os resíduos e inclui no objeto `residuos_p`. Para analisá-los, faz-se um sumário, usando a função `summary()`. Esta função exibirá os a estatística dos 5 números mais a média para os resíduos padronizados:

```
residuos_p <- rstandard(mod_reg)
summary(residuos_p)
```

```
##      Min.   1st Qu.   Median   Mean   3rd Qu.   Max.
## -1.9327846 -0.7271178  0.0548028  0.0006059  0.7779208  2.1154118
```

Em uma amostra normalmente distribuída, ao redor de 95% dos valores estão entre $-1,96$ e $+1,96$, 99% deve estar entre $-2,58$ e $+2,58$ e quase todos (99,9%) deve se situar entre $-3,09$ e $+3,09$.

Portanto, resíduos padronizados com um valor absoluto maior que 3 são motivo de preocupação porque em uma amostra média é improvável que aconteça um valor tão alto por acaso (133).

Se a saída da função `rstandard()` for comparada com o eixo y do gráfico Residuals vs Leverage, dos gráficos diagnósticos, verifica-se valores semelhantes que variam abaixo de 3 e acima de -3, indicando que não há outliers influenciando e a mediana está próxima de zero.

15.2.2.4 Homocedasticidade dos resíduos A variância constante, homogênea, é frequentemente chamada de homocedasticidade. Por outro lado, a variância não constante é chamada de heterocedasticidade. Foi visto que é possível usar um gráfico de **resíduos versus ajustados** (previstos) e o **scale-location** (gráficos diagnósticos) para avaliar esse pressuposto. Embora o gráfico possa dar uma ideia sobre homocedasticidade, às vezes, um teste mais formal é preferido. Existem vários testes para isso, mas aqui será utilizado o **Teste de Breusch-Pagan**. A H_0 e a H_A podem ser consideradas como,

H_0 : *Homocedasticidade*. Os resíduos têm variação constante sobre o modelo verdadeiro.

H_A : *Heterocedasticidade*. Os resíduos não têm variação constante sobre o modelo verdadeiro.

Se o valor $P > 0,05$ não se rejeita a H_0 de homocedasticidade. O teste de Breusch-Pagan é encontrado na função **bptes()**, incluída no pacote **lmtest(134)**.

```
lmtest::bptest(mod_reg)
```

```
##  
## studentized Breusch-Pagan test  
##  
## data: mod_reg  
## BP = 0.049988, df = 1, p-value = 0.8231
```

Os resultados da saída do teste exibem um valor $P > 0,05$, indicando que não se deve rejeitar a homocedasticidade, mostrando que a pequena alteração no gráfico **Scale-location** e **Residuals vs Fitted** não é preocupante.

O problema mais sério associado à heterocedasticidade é o fato de que os erros padrão são tendenciosos. Como o erro padrão é fundamental para a realização de testes de significância e cálculo de intervalos de confiança, os erros padrão tendenciosos levam a conclusões incorretas sobre a significância dos coeficientes de regressão. No geral, no entanto, a violação da suposição de homocedasticidade deve ser bastante grave para apresentar um grande problema, dada a natureza robusta da regressão pelo método **ordinary least-squares**. No entanto, é importante que a equação final de predição seja aplicada apenas a populações com as mesmas características da amostra do estudo.

15.2.2.5 Independência dos resíduos Os resíduos no modelo devem ser independentes, ou seja, não devem ser correlacionados entre si. Para verificar isso, pode-se executar o *teste Durbin-Watson* (**teste dw**), utilizando a função **durbinWatsonTest()** do pacote 'car'. O teste retorna um valor entre 0 e 4. Um valor maior que 2 indica uma correlação negativa entre resíduos adjacentes, enquanto um valor menor que 2 indica uma correlação positiva. Se o valor for dois, é provável que exista independência. Existe uma sugestão de que valores abaixo de 1 ou mais de 3 são um motivo definitivo de preocupação (133). É importante mencionar que o teste tem como pressuposto a normalidade dos dados.

```
durbinWatsonTest(mod_reg)
```

```
## lag Autocorrelation D-W Statistic p-value  
##   1      -0.1044054     2.204843   0.642  
## Alternative hypothesis: rho != 0
```

Como na saída do teste o valor $P > 0,05$ e a estatística DW é igual a 2,2, não se rejeita a hipótese nula de independência ($\rho = 0$).

15.2.3 Tamanho amostral na regressão

O tamanho da amostra deve ser suficiente para suportar o modelo de regressão. É importante coletar dados suficientes para obter um modelo de regressão confiável. O tamanho da amostra necessário para suportar um modelo depende do valor do coeficiente de correlação do modelo (no caso da correlação linear simples é o r de Pearson) e do número de variáveis incluídas.

A Tabela 17 (135) mostra o número de participantes necessários em modelos com 1 a 4 preditores independentes. Como se observa, o requisito de tamanho da amostra aumenta com o número de variáveis preditoras.

Tabela 17: Tamanho amostral para regressão

Valor r	1 variável preditora	2 variáveis preditoras	3 variáveis preditoras	4 variáveis preditoras
0.2	190	230	265	290
0.3	80	100	115	125
0.4	45	55	65	70

Existem muitas regras práticas, sugerindo o tamanho da amostra. Uma delas, diz que se deve ter 10 a 15 casos por variável preditora no modelo. Entretanto, essas regras podem ser duvidosas e o melhor é calcular o tamanho amostral baseado no tamanho do efeito, usando, por exemplo o site StatToDo (CHANG, 2014)

15.2.4 Realização da regressão linear

Após analisar os pressupostos do modelo de regressão do exemplo, verificou-se que as variáveis `idade` e `comprimento` da criança têm relação linear, que os resíduos do modelo têm distribuição normal, que existe homoscedasticidade e que não há pontos influentes. E, portanto, o modelo permite que se realize uma análise de regressão linear para avaliar a relação entre as variáveis independentes e dependentes.

Para realizar uma análise de regressão linear simples e verificar os resultados, há necessidade de executar dois comandos. O primeiro, que cria o modelo linear já foi realizado na análise dos gráficos e será repetido aqui. O segundo, imprime o resumo do modelo com a função `summary()`:

```
mod_reg <- lm (comp ~ idade, dados)
summary (mod_reg)

##
## Call:
## lm(formula = comp ~ idade, data = dados)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -3.2033 -1.2033  0.0899  1.2585  3.4922 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 61.44408   1.36466  45.02   <2e-16 ***
## idade        1.06515   0.04967  21.45   <2e-16 ***
## ---      
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1.678 on 38 degrees of freedom
```

```
## Multiple R-squared:  0.9237, Adjusted R-squared:  0.9217
## F-statistic: 459.9 on 1 and 38 DF,  p-value: < 2.2e-16
```

A saída da função `summary()` primeiro apresenta como o modelo foi obtido e, em seguida, resume os resíduos do modelo.

Os *Coeficientes* mostram:

1. As estimativas (*Estimate*) para os parâmetros do modelo - o valor do intercepto y (neste caso, 61,44) e o efeito estimado da idade sobre o comprimento (1,1). Isto significa que para cada unidade de aumento na idade se espera um aumento de 1,1 cm no comprimento.
 2. O erro padrão dos valores estimados (*Std. Error*).
 3. A estatística de teste (*t value*)
 4. O valor P ($\Pr(|t|)$), também conhecido como a probabilidade de encontrar a estatística t fornecida se a hipótese nula de nenhuma correlação for verdadeira.
- As três linhas finais são os diagnósticos do modelo - o mais importante a observar é o valor P (aqui é $2,2 \times 10^{-16}$), que indicará se o modelo se ajusta bem aos dados.

A partir desses resultados, pode-se dizer que existe uma correlação positiva significativa entre idade e comprimento (valor $P < 0,001$), com um aumento de 1,1 cm no comprimento para cada aumento de 1 mês na idade, possibilitando a previsão comprimento da criança pela idade.

Estes dados são empregados para formular a equação do modelo de regressão da seguinte maneira:

$$\hat{y} = 61,44 + 1,1x$$

O erro padrão das estimativas são fornecidos. Esses dados permitem calcular o IC95%. Ou pode-se usar a função `confint()` do pacote `stats`, que será colocada dentro da função `round()` para arredondar os valores até um dígito.

```
round (confint (mod_reg, level = 0.95), 1)
```

```
##           2.5 % 97.5 %
## (Intercept) 58.7   64.2
## idade       1.0    1.2
```

Dessa forma, é possível prever que uma criança de 30 meses, de acordo com o modelo, terá o seguinte comprimento:

```
comp_30m <- 61.4 + 1.1 *30
comp_30m
```

```
## [1] 94.4
```

```
lim.sup <- 64.2 + 1.2*30
lim.inf <- 58.7 + 1.0*30
print (c(lim.inf, lim.sup))
```

```
## [1] 88.7 100.2
```

Ou seja, espera-se que uma criança tenha, aos 30 meses de idade, um comprimento médio de 94,4 cm (IC95%: 88,7-100,2)

15.2.5 Visualização dos resultados

Será obtido um gráfico de dispersão com a reta de regressão e seu intervalo de confiança de 95% (Figura 176). Além disso, adicionou-se a equação do modelo de regressão (o R arredondou os valores), juntamente com o coeficiente de determinação R^2 .

```
ggplot (dados, aes (x = idade, y = comp)) +
  geom_point (size = 2) +
  geom_smooth (method = "lm", se = TRUE, color = "steelblue") +
  stat_regressao_equation (label.y = 100, aes (label = ..eq.label..)) +
  stat_regressao_equation (label.y = 99, aes (label = ..rr.label..)) +
  theme_classic () +
  xlab ("Idade (meses)") +
  ylab ("Comprimento(cm)") +
  theme (text = element_text (size = 12))
```

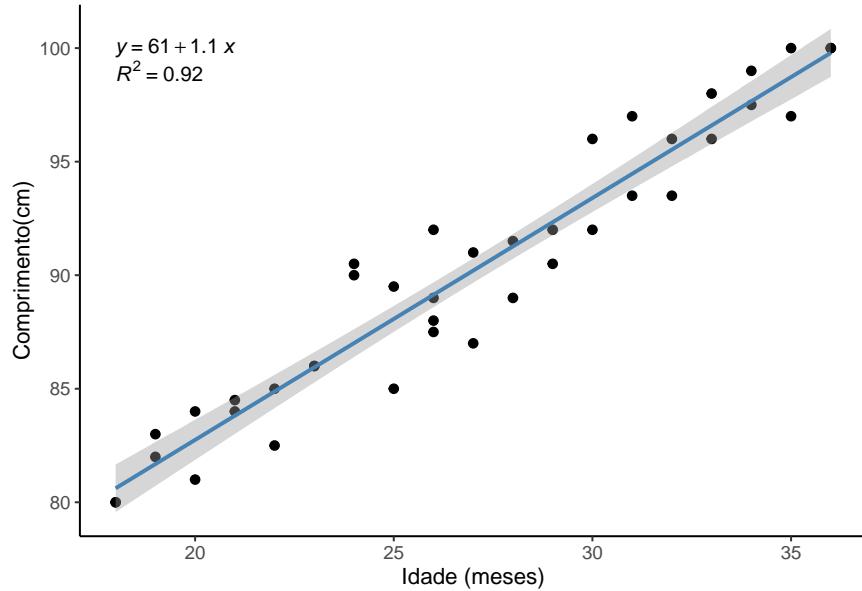


Figura 176: Resultado da regressão linear

No gráfico, o intervalo de previsão médio de 95% em torno da reta de regressão é um intervalo de confiança de 95%, ou seja, a área na qual há 95% de certeza de que a reta de regressão verdadeira se encontra (136). Esta banda de intervalo é levemente curvada porque os erros na estimativa do intercepto e da inclinação são incluídos em adição ao erro na previsão da variável desfecho.

16 Análise de Dados Categóricos

16.1 Pacotes necessários

```
pacman::p_load(readxl,  
                 dplyr,  
                 knitr,  
                 kableExtra,  
                 ggplot2,  
                 gmodels,  
                 expss,  
                 nhstplot,  
                 DescTools,  
                 summarytools,  
                 rstatix)
```

16.2 Qui-Quadrado

Dois testes de hipótese são proeminentes na pesquisa na área da saúde. Um é o teste t de duas amostras, que é usado para testar a igualdade de duas médias populacionais independentes. O segundo é o teste qui-quadrado (denotado por χ^2). O teste é denominado teste qui-quadrado porque usa a distribuição qui-quadrado ou χ^2 .

16.2.1 Distribuição qui-quadrado

Se uma variável X é normalmente distribuída, então a variável X^2 tem uma distribuição qui-quadrado (137). A distribuição qui-quadrado com k categorias é a distribuição de uma soma dos quadrados de k variáveis aleatórias independentes com distribuição normal. O número de categorias determina o número de graus de liberdade. O formato da distribuição qui-quadrado depende desses graus de liberdade.

Em geral, ela é assimétrica com apenas valores positivos, iniciando em zero. A assimetria diminui à medida que aumentam os graus de liberdade. Para cada grau de liberdade tem-se curvas de distribuição diferentes.

```
curve(dchisq(x, df = 5), from = 0, to = 60, col = "royalblue", lwd =2, bty = "n")  
curve(dchisq(x, df = 10), from = 0, to = 60, col = "red", lwd =2, add = T)  
curve(dchisq(x, df = 15), from = 0, to = 40, col = "orange", lwd =2, add = T)  
curve(dchisq(x, df = 20), from = 0, to = 40, col = "cyan", lwd =2, add = T)  
curve(dchisq(x, df = 30), from = 0, to = 60, col = "green3", lwd =2, add = T)  
box(bty = "L")  
legend (legend=c ("gl = 05", "gl = 10", "gl = 15", "gl = 20", "gl = 30"),  
       fill = c ("royalblue", "red", "orange", "cyan", "green3"),  
       bty="n",  
       cex = 1,  
       x ="right")
```

A distribuição χ^2 converge para a distribuição normal à medida que os graus de liberdade aumentam, de acordo com o teorema do limite central, entretanto esta convergência é lenta (Figura 177).

A distribuição qui-quadrado tem duas aplicações comuns: primeiro, como um teste para saber se duas variáveis categóricas são independentes ou não (*Teste de independência ou associação*); segundo, o teste de qualidade do ajuste do qui-quadrado (*Teste de aderência ou ajuste*) que é usado para comparar uma determinada distribuição com uma distribuição conhecida.

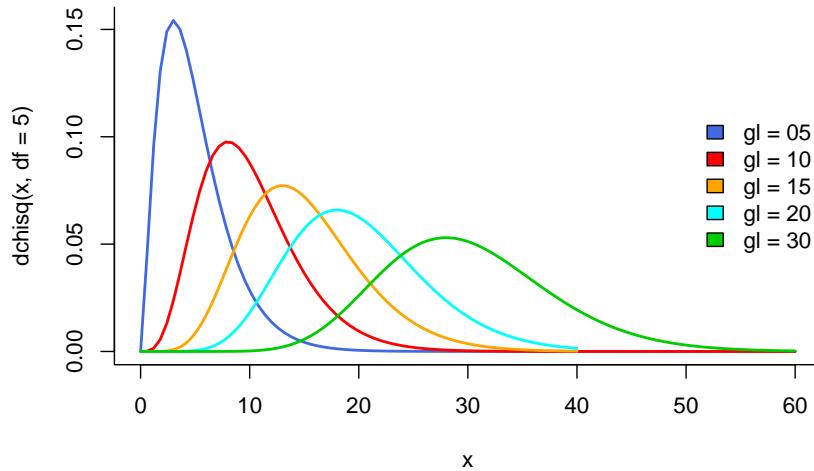


Figura 177: Distribuição do qui-quadrado.

16.2.2 Estatística do qui-quadrado

O cálculo da estatística χ^2 é baseado nas frequências existentes nas células da tabela de contingência. Em primeiro lugar, calcula-se as frequências que se espera em cada célula caso a hipótese nula seja verdadeira (*frequências esperadas*). Em segundo lugar, usando a equação geral, o teste mede o grau de discrepância entre o conjunto de *frequências observadas* (O) e o conjunto de frequências esperadas (E).

$$\chi^2 = \sum \left[\frac{(O_i - E_i)^2}{E_i} \right]$$

Se O_i é muito semelhante ao E_i , então o χ^2 é baixo; se O_i é muito diferente em relação ao E_i , então o χ^2 é alto.

As *frequências observadas* são o número de sujeitos ou objetos na amostra que se enquadram nas várias categorias da variável de interesse. As *frequências esperadas* são o número de sujeitos ou objetos na amostra que seria esperado observar se hipótese nula fosse verdadeira.

$$E = \frac{\text{total coluna} \times \text{total linha}}{\text{total geral}}$$

Por exemplo, suponha a Tabela 18:

Tabela 18: Acidentes automobilísticos

Sexo	Acidentes	Sem acidentes	Total
Homens	16	44	60
Mulheres	4	36	40
Total	20	80	100

Usando os dados da Tabela 18, o número de acidentes esperados para os homens será:

```
esperado <- (20*60)/100  
esperado
```

```
## [1] 12
```

O número de acidentes esperado para os homens é igual a 12, entretanto ocorreram 16. Houve uma diferença. Esta diferença é calculada para todas as células e será o importante no cálculo do qui-0quadrado.

16.2.2.1 Restrições ao qui-quadrado

1. Regra Geral

O teste pode ser usado, se a frequência observada em cada célula for maior ou igual a 5 e a frequência esperada for maior ou igual a 5.

2. Tabela 2×2 ($gl = 1$)

Neste caso, é recomendada a *Correção de Continuidade de Yates*, mesmo quando o n for grande.

3. Tabela $l \times c$

O teste pode ser usado se o número de células com frequência esperada inferior a 5 for menor do que 20% do total das células e nenhuma frequência esperada é igual a zero.

4. n pequeno

Neste caso, é preconizado o *Teste Exato de Fisher*.

16.2.2.2 Valor crítico do qui-quadrado A estatística de teste (que em certo sentido é a diferença entre as frequências observadas e esperadas) deve ser comparada a um valor crítico para determinar se a diferença é grande ou pequena. Não se pode dizer se uma estatística de teste é grande ou pequena sem colocá-la em perspectiva com o valor crítico. Se a estatística de teste estiver acima do valor crítico, significa que a probabilidade de observar tal diferença entre as frequências observadas e esperadas é improvável.

O valor crítico pode ser encontrado na tabela estatística da distribuição Qui-quadrado e depende do nível de significância, denotado α , e dos graus de liberdade, denotado gl . O nível de significância geralmente é igual a 5%. Os graus de liberdade para um teste de Qui-quadrado de independência são encontrados da seguinte forma:

$$gl = (\text{numero de linhas} - 1) \times (\text{numero de colunas} - 1)$$

Em uma tabela de contingência 2×2 , como a Tabela 18, tem $gl = (2 - 1) \times (2 - 1) = 1$. Basta agora obter o valor crítico com a função `qchisq()`:

```
alpha <- 0.05  
gl = 1  
qchisq (1-alpha, gl)
```

```
## [1] 3.841459
```

Este valor é comparado com o $\chi^2_{calculado}$ para um nível de significância de 5%. Se ele é maior, rejeita-se se a H_0 ; caso contrário, não se rejeita. Para obter o valor P , pode-se usar a função `pchisq()`, onde, como argumento, coloca-se o valor do $\chi^2_{calculado}$, os graus de liberdade e acrescenta-se `lower.tail = FALSE` para obter a probabilidade da cauda superior, uma vez que a distribuição do qui-quadrado é unilateral à direita.

Na Tabela 18, o valor crítico é igual a 3,84 e o $\chi^2_{calculado}$ é igual a 4,17, logo a valor P é igual a:

```
pchisq (4.17, 1, lower.tail = FALSE)
```

```
## [1] 0.0411458
```

Dessa forma, concluimos, com uma confiança de 95%, que os homens têm uma proporção maior de acidentes comparados às mulheres ($\chi^2(1) = 4,17; P = 0,041$). Observe na Figura 178 que o $\chi^2_{calculado}$ localiza-se a direita da linha vertical, na área vermelha de rejeição da H_0 .

```
plotchisqtest(chisq = 3.84,
               df = 1,
               colorleft = "aliceblue",
               colorright = "red",
               ylabel = "Densidade de probabilidade sob a hipótese nula")
```

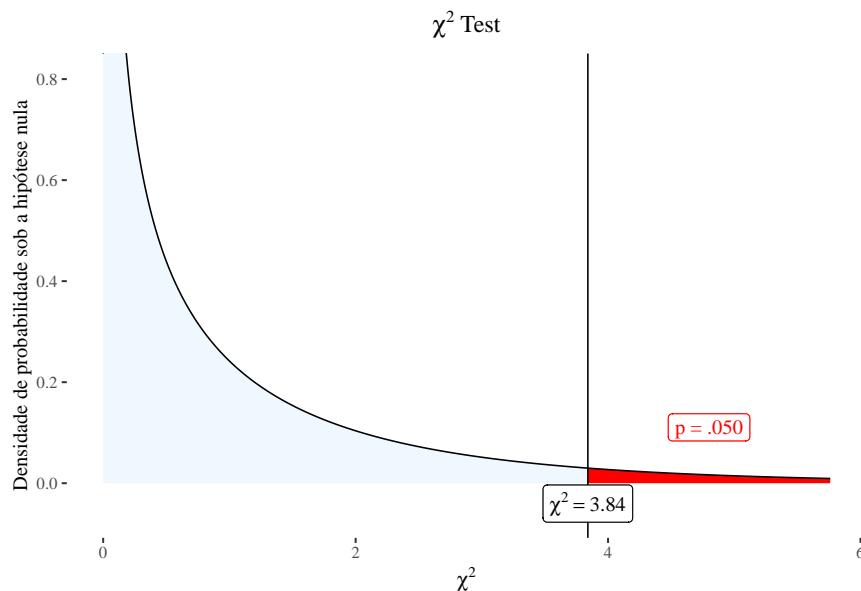


Figura 178: Distribuição do qui-quadrado, gl = 1, alpha = 0,05.

O gráfico foi criado com a função `plotchisqtest()` do pacote `nhstplot`, pacote simples e conveniente para representar graficamente os testes de significância de hipótese nula mais comuns, como testes F , testes t e testes z (138).

16.3 Qui-quadrado de independência ou associação

16.3.1 Carregar, explorar e preparar os dados

Inicialmente, vamos tornar ativo o banco de dados `dadosMater.xlsx`. Para baixar o banco de dados, clique [aqui](#). Salve o mesmo no seu diretório de trabalho.

```
dados <- read_excel ("dadosMater.xlsx")
```

Adicione a este arquivo uma variável denominada `baixoPeso` usando a função `ifelse ()`:

```
dados$baixoPeso <- ifelse(dados$pesoRN < "2500", "1", "2")
```

Onde 1 = sim e 2 = não, ou seja, com peso de nascimento $< 2500\text{g}$ ou $\geq 2500\text{g}$.

O próximo passo é selecionar, deste arquivo, apenas esta variável criada e a variável `fumo`, porque o objetivo da análise será verificar se existe associação entre tabagismo na gestação e baixo peso ao nascer ($< 2500\text{g}$).

```
dados <- dados %>% select (fumo, baixoPeso)
```

A seguir, será extraída uma amostra deste banco de dados com $n = 300$, usando a função `sample_n()` do pacote `dplyr`. A função `set.seed()` apenas garante que os dados selecionados aleatoriamente se mantenham os mesmos em outros sorteios:

```
set.seed(123)
dados <- sample_n(dados, 300)

str(dados)

## tibble [300 x 2] (S3:tbl_df/tbl/data.frame)
## $ fumo      : num [1:300] 2 2 1 2 2 2 1 2 2 2 ...
## $ baixoPeso: chr [1:300] "2" "2" "1" "2" ...
```

Temos agora um conjunto de dados com duas colunas: `fumo`, como uma variável numérica e `baixoPeso`, como caractere. Ambas devem ser transformadas em fator e, onde temos 1 e 2, mudar os rótulos para “sim” e “não” e mantendo a ordem sim” e “não”.

```
dados$fumo <- factor(dados$fumo,
                      levels = c(1, 2),
                      labels = c("sim", "não"))

dados$baixoPeso <- factor(dados$baixoPeso,
                           levels = c(1, 2),
                           labels = c("sim", "não"))

str (dados)

## tibble [300 x 2] (S3:tbl_df/tbl/data.frame)
## $ fumo      : Factor w/ 2 levels "sim","não": 2 2 1 2 2 2 1 2 2 2 ...
## $ baixoPeso: Factor w/ 2 levels "sim","não": 2 2 1 2 1 2 2 2 2 2 ...
```

Tabelas

Com os dados, será construída uma tabela com a função `table()`:

```
tab <- with(data = dados, table(fumo, baixoPeso))
addmargins(tab)
```

```

##      baixoPeso
## fumo  sim não Sum
##   sim  17  52  69
##   não  19 212 231
##   Sum  36 264 300

```

As prevalências de baixo peso por categoria de tabagismo:

```

fumantes <- tab[1,1]/(tab[1,1]+ tab[1,2])
fumantes

## [1] 0.2463768

não.fumantes <- tab[2,1]/(tab[2,1]+ tab[2,2])
não.fumantes

## [1] 0.08225108

```

Visualização gráfica

Será construído um gráfico de barras empilhadas (Figura 179):

```

ggplot(dados) +
  aes (x = fumo, fill = baixoPeso) +
  geom_bar () +
  scale_fill_manual(values = c("gray", "salmon")) +
  labs (title = NULL,
        x = "Tabagismo",
        y = "Frequência") +
  annotate("text", x="sim", y=62, label= "24,6%") +
  annotate("text", x = "não", y=223, label = "8,2%") +
  theme_classic () +
  theme (text = element_text (size = 12)) +
  labs(fill = "Peso ao nascer < 2500g")

```

Observa-se que 24,6% das gestantes fumantes geram bebês com baixo peso, enquanto que entre as não fumantes este percentual cai três vezes, indo para 8,4%. É uma diferença grande! Aqui, quase se tem certeza que ela é significativa.

16.3.2 Hipóteses estatísticas

H_0 : a proporção de baixo peso é igual nos dois grupos (fumantes e não fumantes); não há associação entre as variáveis.

H_A : a proporção de baixo peso é diferente nos dois grupos (fumantes e não fumantes); existe associação entre as variáveis.

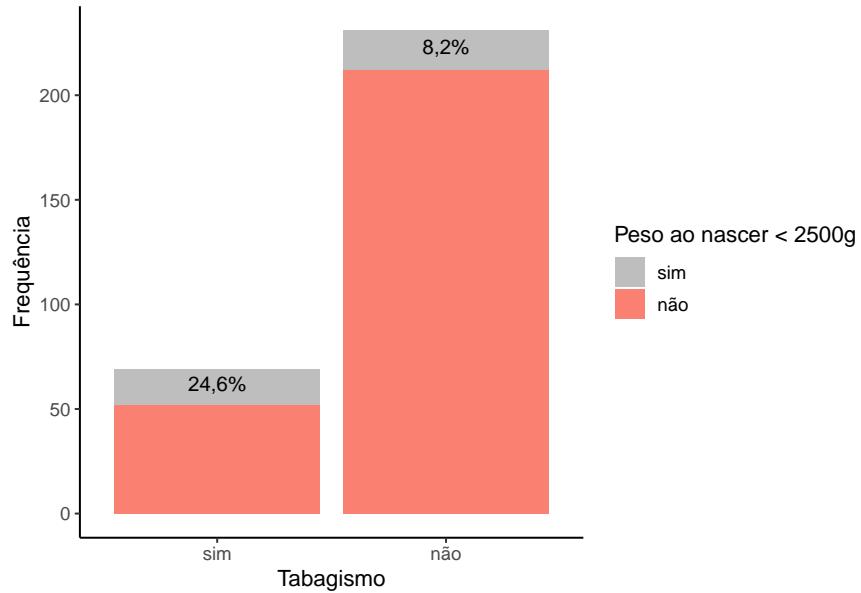


Figura 179: Gráfico de barras empilhadas: tabagismo vs baixo peso ao nascer.

16.3.3 Cálculo do Qui-quadrado de Pearson no R

Para este exemplo, o χ^2 irá verificar se existe uma associação entre as variáveis `fumo` e `baixoPeso`, assumindo um $\alpha = 0,05$ que equivale a um valor crítico de 3,84 com um grau de liberdade, em uma tabela 2×2 . A função `chisq.test()` libera o qui-quadrado com a correção de Yates, pois usa o argumento `correct = TRUE` por padrão.

Quando não se está trabalhando com uma tabela de 2×2 e a regra geral for obedecida e o n for grande, pode-se usar o qui-quadrado de Pearson sem correção.

Para executar a função `chisq.test()`, basta colocar como argumento as variáveis `fumo` e `baixoPeso` ou construir antes uma tabela de contingência com a função `table()` e depois colocá-la como argumento. Como tabela `tab` já existe:

```
teste <- chisq.test(tab)
teste
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab
## X-squared = 12.043, df = 1, p-value = 0.0005198
```

A saída do teste exibe tudo que é necessário: o título do teste, as variáveis usadas (as da tabela `tab`) , a estatística de teste, os graus de liberdade e o valor P do teste. É possível recuperar a estatística de teste χ^2 , os valores esperados e o valor P com:

```
teste$statistic
## X-squared
## 12.04314
```

```

teste$expected

##      baixoPeso
## fumo    sim   não
##   sim  8.28 60.72
##   não 27.72 203.28

```

```
teste$p.value
```

```
## [1] 0.000519832
```

Se um aviso como **"Chi-squared approximation may be incorrect"** (Aproximação qui-quadrado pode estar incorreta) aparecer, significa que as menores frequências esperadas são inferiores a 5. Para evitar esse problema, é possível usar uma das seguintes opções:

- reunir alguns níveis (especialmente aqueles com um pequeno número de observações) para aumentar o número de observações nos subgrupos, ou
- usar o teste exato de Fisher.

Existe uma função no pacote **gmodels** (139), muito interessante, onde se encontra a função **CrossTable()** que imprime, além de uma tabela de frequência com as probabilidades e outros teste, como o teste χ^2 , o teste exato de Fisher e o teste de McNemar com e sem correção de continuidade. Consulte a ajuda para melhor estudar esta elegante função! Neste momento, será explorado apenas o qui-quadrado e os valores esperados com três dígitos:

```

CrossTable (dados$fumo,
            dados$baixoPeso,
            digits = 3,
            prop.chisq = FALSE,
            prop.t = FALSE,
            chisq = TRUE,
            expected = TRUE)

```

```

##
##
##      Cell Contents
## |-----|-----|
## |           N |-----|
## |           Expected N |-----|
## |           N / Row Total |-----|
## |           N / Col Total |-----|
## |-----|
##
##
## Total Observations in Table:  300
##
##
##           | dados$baixoPeso
##   dados$fumo |     sim |     não | Row Total |
## -----|-----|-----|-----|
##   sim |     17 |     52 |     69 |

```

```

##          |    8.280 |    60.720 |      |
##          |    0.246 |     0.754 |    0.230 |
##          |    0.472 |     0.197 |      |
## -----
##       não |      19 |     212 |    231 |
##          | 27.720 | 203.280 |      |
##          |    0.082 |     0.918 |    0.770 |
##          |    0.528 |     0.803 |      |
## -----
## Column Total |      36 |     264 |    300 |
##          |    0.120 |     0.880 |      |
## -----
## 
## 
## Statistics for All Table Factors
## 
## Pearson's Chi-squared test
## -----
## Chi^2 = 13.55281      d.f. = 1      p = 0.0002319443
## 
## Pearson's Chi-squared test with Yates' continuity correction
## -----
## Chi^2 = 12.04314      d.f. = 1      p = 0.000519832
## 
## 
```

Observe que a saída mostra em cada célula da tabela, o número de casos, o número esperado, a percentagem por linha (n^o de casos/total da linha) e a percentagem por coluna (n^o de casos/total da coluna). Por último, exibe o qui-quadrado de Pearson com e sem coreção de continuidade de Yates.

16.3.4 Conclusão

Usando a correção de continuidade de Yates, pois é uma tabela 2×2 , vê-se que o valor P é menor que o nível de significância de 5% e, consequentemente, rejeita-se a hipótese nula e conclui-se que existe uma associação significativa entre tabagismo na gestação e o baixo peso ao nascimento ($\chi^2_{com\ corre\acute{o}\ de\ Yates}(1) = 12; P = 0,0005$).

Além disso, no relato dos resultados pode-se apresentar uma tabela ou em um gráfico.

Tabela Para a construção da tabela, pode-se usar a função `ctable()` do pacote `summarytools(140)` para obter uma tabela com todos os dados a serem exibidos. O argumento `prop = "r"` exibe os percentuais das linhas (“c”, nas colunas). Na realidade, são maneiras diferente de se obter o mesmo resultado.

```
ctable(dados$fumo, dados$baixoPeso,  
       prop = "r",  
       chisq = TRUE,  
       headings = FALSE)
```

```
##  
## -----  
##          baixoPeso      sim      não      Total  
##  fumo  
##    sim           17 (24.6%)   52 (75.4%)   69 (100.0%)
```

```

##      não          19 ( 8.2%)   212 (91.8%)   231 (100.0%)
##  Total          36 (12.0%)   264 (88.0%)   300 (100.0%)
## -----
## -----
##   Chi.squared    df   p.value
## -----
##     12.0431      1     5e-04
## -----

```

Para a apresentação dos resultados, é interessante calcular os intervalos de confiança para cada uma das proporções e apresentar junto a uma tabela. Para isso, a função `BinomCI()`, vista quando se estudou distribuição binomial, cumpre um papel satisfatório:

Baixo peso entre os fumantes

```
BinomCI(17, 69,
        conf.level = 0.95,
        method = "clopper-peerson")
```

```

##           est     lwr.ci     upr.ci
## [1,] 0.2463768 0.1505497 0.3649049
```

Baixo peso entre os não fumantes

```
BinomCI(19, 231,
        conf.level = 0.95,
        method = "clopper-peerson")
```

```

##           est     lwr.ci     upr.ci
## [1,] 0.08225108 0.05024649 0.1254658
```

Estes dados podem ser colocados em uma tabela, como a Tabela 19:

Tabela 19: Efeito do tabagismo materno no peso ao nascer

	Fumantes	Não fumantes	Valor P
Baixo Peso	17/69	19/231	0.00052
IC95%	15,1-36,5	5,0-12,5	

Gráfico

Uma boa apresentação seria com gráficos de barras empilhadas (Figura 180), acompanhado dos percentuais e do tipo de teste realizado, usando a função `get_test_label()` que necessita do teste calculado com a função `chisq_test()` do pacote `rstatix`, já discutido em outras ocasiões.

```
teste_r <- rstatix::chisq_test(tab, correct = T)
teste_r
```

```

## # A tibble: 1 x 6
##       n statistic      p    df method      p.signif
## * <int>     <dbl>   <dbl> <int> <chr>      <chr>
## 1    300      12.0  0.00052     1 Chi-square test ***
```

```

ggplot(dados) +
  aes (x = fumo, fill = baixoPeso) +
  geom_bar () +
  scale_fill_manual(values = c("gray", "gray30")) +
  labs (title = NULL,
        subtitle = get_test_label (teste_r, detailed = TRUE),
        x = "Tabagismo",
        y = "Frequência") +
  annotate("text", x="sim", y=62, label= "24,6% (15,0-36,5)") +
  annotate("text", x = "não", y=223, label = "8,2% (5,0-12,5)") +
  theme_classic () +
  theme (text = element_text (size = 12)) +
  labs(fill = "Peso ao nascer < 2500g")

```

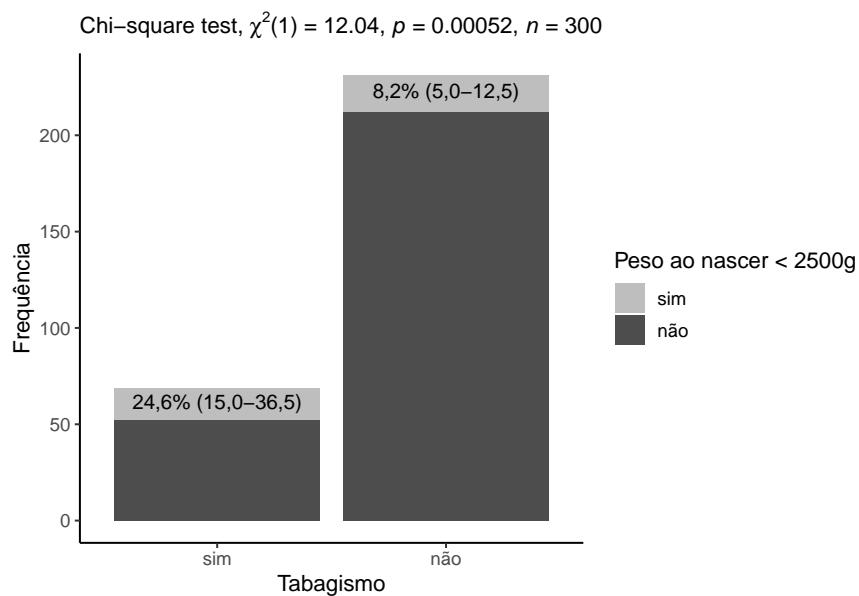


Figura 180: Gráfico de barras empilhadas: tabagismo vs baixo peso ao nascer.

16.4 Teste Aderência ou do Melhor Ajuste

O teste de qualidade de ajuste do qui-quadrado (*chi-square goodness of fit*) é usado para comparar a distribuição observada com uma distribuição esperada, em uma situação em que se tem duas ou mais categorias em dados discretos. Em outras palavras, ele compara várias proporções observadas com as probabilidades esperadas (141).

16.4.1 Dados

Há uma dúvida se o número de pacientes que procura uma determinada Unidade de Pronto Atendimento (UPA) é aproximadamente o mesmo em todos os dias da semana. Esta é uma informação importante sob o ponto de vista administrativo. Para se atingir este objetivo registrou-se o número de pacientes que procurou a UPA por dia da semana.

O número de atendimentos nos sete dias (de segunda-feira à domingo) da semana está representada pela *frequência observada*, *freq_obs*:

```
freq_obs <- c(20, 17, 22, 21, 26, 33, 36)
freq_obs
```

```
## [1] 20 17 22 21 26 33 36
```

O total de atendimentos durante uma semana é igual a:

```
soma <- sum(freq_obs)
soma
```

```
## [1] 175
```

Assim, a frequência esperada diária é igual a soma total dos atendimentos dividido pelo número observações (no caso, dias da semana), representada por k:

```
k = 7
freq_esp <- soma/k
freq_esp
```

```
## [1] 25
```

onde k é o número de células (número de dias na semana).

Com estes valores , pode-se criar um vetor, p, com as proporções dos atendimentos diárias esperados:

```
p <- c(freq_esp/soma, freq_esp/soma, freq_esp/soma, freq_esp/soma, freq_esp/soma, freq_esp/soma, freq_esp/soma)
```

```
## [1] 0.1428571 0.1428571 0.1428571 0.1428571 0.1428571 0.1428571 0.1428571
```

16.4.2 Hipóteses estatísticas

H_0 : a distribuição das frequências observadas (O) é igual a distribuição de frequências esperadas (E)

H_A : a distribuição das frequências observadas (O) não é igual a distribuição de frequências esperadas (E)

16.4.3 Cálculo do teste estatístico

Vamos assumir um $\alpha = 0,05$. Os graus de liberdade são calculados como o número de células (k) menos 1: $gl = (k - 1)$. O χ^2_{crtico} pode ser encontrado usando:

```
alpha = 0.05
k = 7
gl = k - 1
qchisq (1 - alpha, gl)
```

```
## [1] 12.59159
```

Em outras palavras, se o $\chi^2_{calculado} > \chi^2_{critico}$, rejeita-se a H_0 . Na Figura 181, o resultado tem que ficar acima da linha vertical vermelha para que a hipótese nula seja rejeitada. Se cair fora da área de rejeição, abaixo da linha vertical vermelha, aceita-se a hipótese nula.

```
plotchisqtest(chisq = 12.6,
              df = 6,
              colorleft = "aliceblue",
              colorright = "red",
              ylabel = "Densidade de probabilidade",
              colorcut = "red",)
```

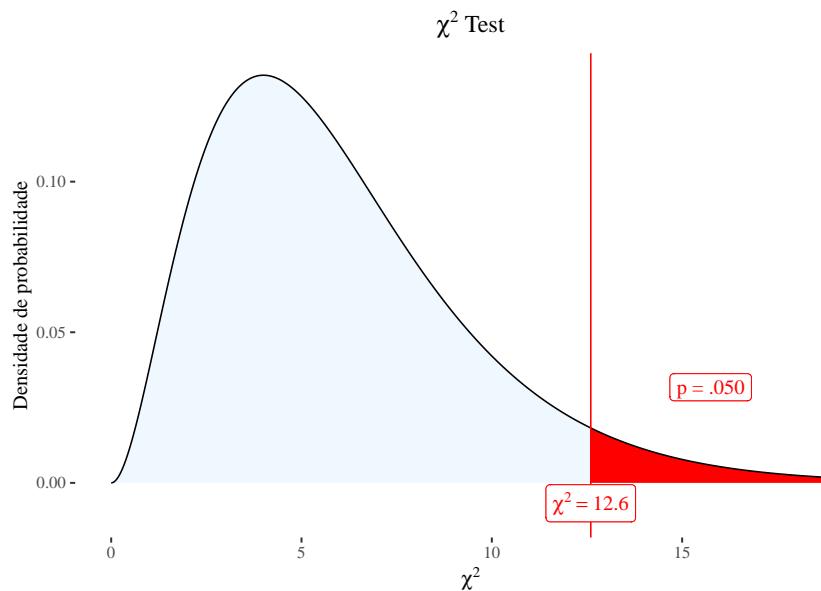


Figura 181: Distribuição do qui-quadrado, gl = 6, alpha = 0,05.

A estatística do teste pode ser encontrada, usando a função `chisq.test()`:

```
chisq.test (x = freq_obs, p = p)
```

```
##  
## Chi-squared test for given probabilities  
##  
## data: freq_obs  
## X-squared = 12, df = 6, p-value = 0.06197
```

16.4.4 Conclusão

Observando a Saída do teste do qui-quadrado, verifica-se que o $\chi^2_{calculado} < \chi^2_{critico}$, portanto, não se rejeita a H_0 e conclui-se que, nesta amostra, com uma confiança de 95%, que a frequência observada de pacientes à UPA é igual a esperada ($P = 0,062$). Lembrando que, neste caso, como temos um valor P limitrofe, existindo a possibilidade de se estar aceitando uma H_0 falsa e cometendo um erro tipo II. Seria recomendado, aumentar o tamanho amostral em uma nova coleta, usando estes dados como um piloto para o cálculo amostral.

16.5 Qui-quadrado de Pearson para tabelas extensas

Utilizados para tabelas quando o número de grupos, k , é superior a 2. Por exemplo, verificar se existe uma tendência de maior taxa de infecção nos neonatos que permanecem mais tempo hospitalizados. Será usado o banco de dados `dadosCirurgia.xlsx` que pode ser encontrado [aqui](#). Salve o mesmo no seu diretório de trabalho. Este banco de dados contém 144 recém-nascidos submetidos a diferentes procedimentos cirúrgicos. A variável tempo de hospitalização (`tempohosp`) é contínua e assimétrica. Então, para que possa ser usada aqui, será categorizada por quartis. A variável `infec` (presença de infecção) é uma variável dicotômica (`sim`, `não`).

16.5.1 Carregar o banco de dados

```
cirurgia <- read_excel ("dadosCirurgia.xlsx")
```

16.5.2 Exploração e manipulação do banco de dados

Vamos observar o banco de dados com a função `glimpse()` do pacote `dplyr`:

```
glimpse (cirurgia)
```

```
## #> #> Rows: 144
## #> #> Columns: 7
## #> #> $ id      <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 1~
## #> #> $ sexo    <chr> "masc", "masc", "masc", "masc", "fem", "fem", "fem", ~
## #> #> $ peso     <dbl> 2020, 1850, 2540, 1150, 2900, 2480, 2225, 3245, 2390, 2872, ~
## #> #> $ ig       <dbl> 36, 30, 38, 31, 36, 37, 38, 39, 38, 39, 40, 35, 38, 36, 36, ~
## #> #> $ tempohosp <dbl> 37, 37, 37, 46, 37, 36, 30, 18, 25, 14, 14, 9, 17, 15, 17, 1~
## #> #> $ infec    <chr> "não", "não", "sim", "sim", "não", "sim", "não", "não", "sim~
## #> #> $ cirurgia  <chr> "abdominal", "abdominal", "abdominal", "outra", "abdominal", ~
```

Caracteristicamente a variável tempo de hospitalização é assimétrica e vamos transformá-la em categorias:

```
summary (cirurgia$tempohosp)
```

```
##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##      1.00   20.75  27.50   37.42  42.00   245.00
```

O sumário da variável nos dá orientação para a categorização:

```
cirurgia$tempo <- cut(cirurgia$tempohosp,
                        breaks = c(1, 20.75, 27.50, 42.00, 245),
                        labels = c("<= 21", "22-28", "29-42", ">42"),
                        right = FALSE,
                        include.lowest = TRUE)

tab1 <- table (cirurgia$tempo)
tab1
```

```

## 
## <= 21 22-28 29-42 >42
##   36     36     35     37

```

Agora, a variável `cirurgia$infec` será colocada como um fator:

```

cirurgia$infec <- factor(cirurgia$infec, levels = c("sim", "não"))
table(cirurgia$infec)

```

```

## 
## sim não
## 56 88

```

Vamos cruzar a variável `tempo` com a variável `infec` em uma tabela:

```

tab2 <- table (cirurgia$tempo, cirurgia$infec)
addmargins(tab2)

```

```

## 
##           sim não Sum
## <= 21    9 27 36
## 22-28   11 25 36
## 29-42   14 21 35
## >42     22 15 37
## Sum     56 88 144

```

16.5.3 Hipóteses estatísticas

H_0 : A presença de infecção não altera o tempo de hospitalização.
 H_A : A presença de infecção altera o tempo de hospitalização.

16.6 Cálculo da estatística do teste

O teste estatístico χ^2 será calculado usando a função `CrossTable()` do pacote `gmodels`, já mencionada anteriormente:

```

CrossTable(cirurgia$tempo,
           cirurgia$infec,
           digits = 2,
           expected = TRUE,
           prop.chisq = FALSE,
           prop.t = FALSE,
           chisq = TRUE,
           mcnemar = FALSE,
           fisher = FALSE)

```

```

## 
##           Cell Contents
## |-----|

```

```

## | N |
## | Expected N |
## | N / Row Total |
## | N / Col Total |
## |-----|
## 
## 
## Total Observations in Table: 144
## 
## 
##          | cirurgia$infec
## cirurgia$tempo |      sim |     não | Row Total |
## -----|-----|-----|-----|
##      <= 21 |      9 |     27 |     36 |
##           | 14.00 | 22.00 |      |
##           | 0.25 | 0.75 | 0.25 |
##           | 0.16 | 0.31 |      |
## -----|-----|-----|-----|
##     22-28 |     11 |     25 |     36 |
##           | 14.00 | 22.00 |      |
##           | 0.31 | 0.69 | 0.25 |
##           | 0.20 | 0.28 |      |
## -----|-----|-----|-----|
##     29-42 |     14 |     21 |     35 |
##           | 13.61 | 21.39 |      |
##           | 0.40 | 0.60 | 0.24 |
##           | 0.25 | 0.24 |      |
## -----|-----|-----|-----|
##      >42 |     22 |     15 |     37 |
##           | 14.39 | 22.61 |      |
##           | 0.59 | 0.41 | 0.26 |
##           | 0.39 | 0.17 |      |
## -----|-----|-----|-----|
##   Column Total |     56 |     88 |    144 |
##           | 0.39 | 0.61 |      |
## -----|-----|-----|-----|
## 
## 
## Statistics for All Table Factors
## 
## 
## Pearson's Chi-squared test
## -----
## Chi^2 = 10.58013     d.f. = 3     p = 0.01422704
## 
## 
## 

```

Na Saída, observa-se que o percentual de neonatos infectados aumenta com o tempo de hospitalização (16% para o grupo do menor quartil do tempo de hospitalização e 39% para o grupo do maior quartil) com $\chi^2(3) = 10,58; P = 0,014$.

16.7 Conclusão

A partir desses resultados, pode-se inferir que a menor taxa de infecção está no grupo do primeiro quartil e é significativamente diferente em relação a taxa de infecção do maior quartil, mas sem indicação para os grupos intermediários. É útil fazer o teste de tendência linear (*Linear-by-linear Association*). Para isso, pode-se usar a função `lbl_test ()` do pacote `coin`.

```
coin::lbl_test (cirurgia$tempo ~ cirurgia$infec)

##
##  Asymptotic Linear-by-Linear Association Test
##
## data:  cirurgia$tempo (ordered) by cirurgia$infec (sim, não)
## Z = 3.1231, p-value = 0.001789
## alternative hypothesis: two.sided
```

Este teste indica uma tendência significativa para a presença de infecção à medida que aumenta o tempo de hospitalização ($P = 0,0018$).

16.8 Teste exato de Fisher

O teste do qui-quadrado não é um método apropriado de análise se a amostra é pequena. Por exemplo, se n for menor que 20 ou se n estiver entre 20 e 40 e uma das frequências esperadas for menor que 5, o teste do qui-quadrado deve ser evitado. Nesta situação, é recomendado o **teste exato de Fisher**.

16.8.1 Dados

Um estudo estabeleceu como objetivo verificar se a asma não controlada é um fator de risco para a procura da emergência. Foram acompanhados 16 escolares asmáticos durante um ano com relação ao número de visitas à emergência de acordo com o controle da sua asma.

16.8.1.1 Entrando com os dados Vamos criar dois vetores com os dados e após criar um dataframe denominado `dadosControle`:

```
emerg <- c (1,1,2,2,2,2,1,1,1,1,1,1,1,2)
controle <- c (1,1,1,1,1,1,2,2,2,2,2,2,2,2)
```

Onde 1 = sim e 2 = não

```
dadosControle <- data.frame(emerg, controle)
```

As variáveis numéricas serão transformadas em fatores:

```
dadosControle$emerg <- factor (dadosControle$emerg,
                                ordered=TRUE,
                                levels = c (1,2),
                                labels = c ('sim', 'não'))
dadosControle$controle <- factor (dadosControle$controle,
                                   ordered=TRUE,
                                   levels = c (1,2),
```

```

            labels = c ('sim', 'não'))

glimpse (dadosControle)

## # Rows: 16
## # Columns: 2
## $ emerg    <ord> sim, sim, não, não, não, não, sim, sim, sim, sim, s~
## $ controle <ord> sim, sim, sim, sim, sim, sim, não, não, não, não, n~

```

16.8.2 Hipóteses estatísticas

H_0 : as variáveis são independentes, não há relação entre as duas variáveis categóricas. .
 H_A : as variáveis são dependentes, existe uma relação entre as duas variáveis categóricas.

16.8.3 Execução do teste estatístico

O teste exato de Fisher é usado quando há pelo menos uma célula na tabela de contingência das frequências esperadas abaixo de 5. Para recuperar as frequências esperadas, use a função `chisq.test()` junto com `$expected`:

```

chisq.test (dadosControle$controle, dadosControle$emerg)$expected

## Warning in chisq.test(dadosControle$controle, dadosControle$emerg): Aproximação
## do qui-quadrado pode estar incorreta

##                               dadosControle$emerg
## dadosControle$controle  sim   não
##                         sim 4.375 2.625
##                         não 5.625 3.375

```

A Saída mostra a presença de três células com valores abaixo de 5, indicando a necessidade de se usar o teste de Fisher.

Pode-se usar a função `fisher.test()`, colocando como argumento uma tabela de contingência 2×2 :

```

tab3 <- table (dadosControle$controle, dadosControle$emerg)
addmargins(tab3)

```

```

##                               sim   não Sum
##     sim      2     5    7
##     não      8     1    9
##     Sum     10    6   16

```

```

fisher.test (tab3)

##
##  Fisher's Exact Test for Count Data
##
##  data:  tab3

```

```

## p-value = 0.03497
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
## 0.0009525702 0.9912282442
## sample estimates:
## odds ratio
## 0.06464255

```

16.9 Conclusão

O valor P é menor que o nível de significância de 5%, previamente estabelecido, e, portanto, deve-se rejeitar a hipótese nula. No contexto, rejeitar a hipótese nula para o teste exato de independência de Fisher significa que há uma associação significativa entre as duas variáveis categóricas (controle da asma e visitas à emergência).

16.10 Teste de Macnemar

É um teste estatístico não paramétrico aplicável nos estudos tipo “antes-e-depois” em que cada indivíduo é utilizado como seu próprio controle e a medida é efetuada em escala nominal. O *teste de McNemar* é usado para determinar se há uma diferença estatisticamente significativa nas proporções entre os dados emparelhados.

As medidas coletadas nesses tipos de projetos de estudo não são independentes e, portanto, os testes do Qui-quadrado não podem ser usados porque os pressupostos serão violados.

O teste de McNemar é usado para avaliar se há uma mudança significativa nas proporções ao longo do tempo para dados emparelhados ou se há uma diferença significativa nas proporções entre casos e controles. O resultado de interesse é a mudança dentro da pessoa (ou diferenças dentro do par) e não há variáveis explicativas.

O teste é calculado examinando o número de respostas que são *concordantes* para positivo (sim em ambas as ocasiões) e negativo (não em ambas as ocasiões) e o número de pares *disconcordantes* (sim e não, ou não e sim). Os pares concordantes não fornecem informações sobre as diferenças e não são usados na avaliação. Em vez disso, deve-se concentrar nos pares discordantes, que podem ser divididos em dois tipos: um par discordante do tipo *sim – não* e um par discordante tipo *não – sim* (142).

16.10.1 Pressupostos do teste de McNemar

Os pressupostos para o teste de McNemar são:

1. A variável desfecho é binária, dicotômica;
2. Cada participante é representado na tabela apenas uma vez;
3. A diferença entre as proporções emparelhadas é o resultado de interesse;
4. O teste de McNemar pode não ser confiável se houver contagens baixas nas células “discordantes”. Existe recomendação de que a soma dessas células seja ≥ 20 (143).

16.10.2 Dados

Em uma universidade, um professor de bioestatística comparou as atitudes de 200 estudantes de Medicina em relação à confiança que eles depositam na análise estatística antes e depois da conclusão da disciplina. A pergunta feita foi: Confiam na análise estatística utilizada nos periódicos médicos? As respostas podem ser resumidas na Tabela 20:

Tabela 20: Confiança na análise estatística após término da disciplina

Pré-teste	Pós_Sim	Pós_Não	Total
Sim	20 (a)	8 (b)	28
Não	22 (c)	150 (d)	172
Total	42	158	200

16.10.3 Hipóteses estatísticas

Considerando as caselas a, b, c e da Tabela 20, a hipótese nula de homogeneidade marginal indica que as duas probabilidades marginais para cada resultado são as mesmas, isto é,

$$p_a + p_b = p_a + p_c$$

e

$$p_c + p_d = p_b + p_d$$

Assim, a hipótese nula e a hipótese alternativa são:

H_0 : a proporção de alunos que respondem sim no pré-teste e no pós-teste é a mesma

H_A : a proporção de alunos que respondem sim no pré-teste e no pós-teste não é a mesma

16.10.4 Cálculo do teste de McNemar

16.10.4.1 Lógica do teste O teste estatístico de McNemar, com correção de continuidade, é obtido utilizando a equação:

$$\chi^2 = \frac{(|b - c| - 1)^2}{b + c}$$

Sob a hipótese nula, com um número suficientemente grande de discordantes (células b e c), o χ^2 tem uma distribuição qui-quadrado com um grau de liberdade. Se o resultado é significativo, isto é, fornece evidências suficientes para rejeitar a hipótese nula, significa que as proporções marginais são significativamente diferentes umas das outras.

Substituindo os dados da Tabela na Equação, tem-se:

```
a <- 20
b <- 8
c <- 22
d <- 150
chi <- ((abs(b - c) - 1)^2)/(b + c)
chi
```

```
## [1] 5.633333
```

Assumindo um $\alpha = 0,05$, pode-se obter valor crítico para o χ^2 para $gl = 1$, usando a função `qchisq()`, do pacote `stats`:

```
alpha = 0.05
qchisq(1 - alpha, 1)
```

```
## [1] 3.841459
```

Desta maneira, rejeita-se a H_0 , pois o $\chi^2_{calculado} > \chi^2_{critico}$. O valor P pode ser conseguido com a função `pchisq()`:

```
pchisq (5.633, 1, lower.tail = FALSE)
```

```
## [1] 0.01762544
```

16.10.4.2 Cálculo do teste de McNemar no R Carregar o arquivo `dadosBioestatistica.xlsx`, que pode ser encontrado [aqui](#). Este conjunto de dados contém os dados da tabela acima.

```
dados <- readxl::read_excel("dadosBioestatistica.xlsx")
```

Tabela de contingência

```
dados$preteste <- factor(dados$preteste, levels = c("sim", "não"))
dados$postteste <- factor(dados$postteste, levels = c("sim", "não"))
```

```
tb <- table(dados$preteste, dados$postteste,
             dnn = c("Pré-teste", "Pós-teste"))
```

```
tb
```

```
##          Pós-teste
## Pré-teste sim não
##       sim   20    8
##       não   22 150
```

Usando a função `mcnemar.test()`, do pacote `stats`, pode-se obter a estatística do teste:

```
mcnemar.test (tb,
               correct = TRUE)
```

```
##
## McNemar's Chi-squared test with continuity correction
##
## data: tb
## McNemar's chi-squared = 5.6333, df = 1, p-value = 0.01762
```

O resultado do teste de McNemar com correção de continuidade é exatamente igual ao calculado manualmente.

Pode-se também usar a função `CrossTable()`, do pacote `gmodels`, como feito no cálculo do qui-quadrado de Pearson, que imprime, além de uma tabela de frequência com as probabilidades, o teste de McNemar com e sem correção de continuidade.

```

gmodels::CrossTable (dados$preteste,
                     dados$postteste,
                     digits = 2,
                     prop.chisq = FALSE,
                     prop.t = FALSE,
                     mcnemar = TRUE)

## 
## Cell Contents
## |-----|
## |           N |
## |           N / Row Total |
## |           N / Col Total |
## |-----|
## 
## 
## Total Observations in Table: 200
## 
## 
##          | dados$postteste
## dados$preteste |      sim |     não | Row Total |
## -----|-----|-----|-----|
##       sim |    20 |     8 |    28 |
##       | 0.71 | 0.29 | 0.14 |
##       | 0.48 | 0.05 |   |
## -----|-----|-----|-----|
##       não |   22 |  150 |  172 |
##       | 0.13 | 0.87 | 0.86 |
##       | 0.52 | 0.95 |   |
## -----|-----|-----|-----|
##   Column Total |    42 |  158 |  200 |
##   | 0.21 | 0.79 |   |
## -----|-----|-----|-----|
## 
## 
## McNemar's Chi-squared test
## -----
## Chi^2 =  6.533333    d.f. =  1    p =  0.01058714
## 
## McNemar's Chi-squared test with continuity correction
## -----
## Chi^2 =  5.633333    d.f. =  1    p =  0.01762209
## 
## 

```

16.10.5 Conclusão

Houve uma modificação estatisticamente significativa na opinião dos alunos após o curso de Bioestatística em relação à confiança nas análises estatísticas (86% no pré-teste de respostas não x 79% no pós-teste, $\chi^2 = 5,63, gl = 1, P = 0,018$). Alguns alunos (14) mudaram de opinião em relação a sua confiança nas análises estatísticas dos periódicos médicos.

17 Métodos não paramétricos

17.1 Pacotes necessários

```
pacman::p_load (readxl,  
                 knitr,  
                 kableExtra,  
                 dplyr,  
                 ggplot2,  
                 ggpubr,  
                 rstatix,  
                 ggsci,  
                 confintr,  
                 coin,  
                 tidyrr)
```

17.2 Distribuição livre

A maioria dos testes estatísticos que discutidos são testes paramétricos. Nestes, o interesse estava focado em estimar ou testar uma hipótese sobre um ou mais parâmetros populacionais. Além disso, o aspecto central desses procedimentos era o conhecimento da forma funcional da população da qual foram retiradas as amostras que forneceram a base para a inferência. Por exemplo, o teste t de Student para amostras independentes e a ANOVA são baseados no pressuposto de que os dados foram amostrados de populações que têm distribuição normal.

Os **testes não paramétricos** não fazem suposições em relação à distribuição da população. Não têm, portanto, os pressupostos restritivos, comuns nos testes paramétricos. Têm *distribuição livre*. São baseados em uma ideia simples de ordenação por postos, do valor mais baixo ao mais alto. Analisam somente os postos, ignorando os valores. Podem ser usados tanto com variáveis ordinais como quantitativas numéricas.

17.3 Postos

Os métodos estatísticos não paramétricos não lidam diretamente com os valores observados. Em função disso, para poder usar a informação fornecida pelas observações, sem trabalhar diretamente com os valores observados, utiliza-se os postos das observações. Posto (*rank*) de uma observação é a sua posição em relação aos demais valores.

A atribuição dos postos de uma variável é realizada da seguinte maneira:

1. Colocam-se as observações em ordem crescente;
2. Associam-se valores, correspondendo às suas posições relativas na amostra. O primeiro elemento recebe o valor 1, o segundo o valor 2 e, assim por diante, até que a maior observação receba o valor n ;
3. Se todas as observações são distintas, os postos são iguais aos valores associados às observações no passo anterior.
4. Para observações iguais (empates), associam-se postos iguais à média das suas posições relativas na amostra.

Por exemplo, suponha uma amostra contendo os escores de Apgar no primeiro minuto de 10 recém-nascidos a termo (Tabela 21). Em primeiro lugar, os valores são colocados em ordem crescente e, após, atribui-se postos aos valores. Observe que os postos atribuídos aos valores das posições 3 e 4 são iguais e correspondentes a média de 3 e 4, que é igual a 3,5. O mesmo ocorreu com os outros valores onde houve empate. A soma

dos postos, no exemplo, é igual a 55. Para verificar a correção do cálculo, haja ou não empates, a soma dos postos será sempre $\frac{n \times (n+1)}{2}$. No exemplo, $n = 10$, logo $\frac{10 \times (10+1)}{2} = 55$.

Tabela 21: Construção dos postos

Apgar 1	Ordem	Posto
4	1	1.0
5	2	2.0
7	3	3.0
8	4	4.5
8	5	4.5
9	6	6.0
10	7	8.0
10	8	8.0
10	9	8.0
11	10	10.0

17.4 Teste de Mann-Whitney

O teste de *Mann-Whitney* é usado para analisar a diferença na variável dependente (desfecho) para dois grupos independentes. O teste classifica todos os valores dependentes, ou seja, o valor mais baixo obtém o posto um e, em seguida, usa a soma dos postos de cada grupo no cálculo da estatística de teste.

É o substituto do teste t para amostras independentes quando os pressupostos deste teste são violados. Para a aplicação do teste de Mann-Whitney a variável de interesse deve ser ordinal ou numérica. Este teste é equivalente ao desenvolvido por Frank Wilcoxon (1892 – 1965), assim algumas vezes é denominado de *Wilcoxon Rank Sum Test*. O R usa esta denominação e é importante não confundir com o teste não paramétrico para amostra pareadas, discutido mais adiante.

17.4.1 Dados

Será usado o banco de dados `dadosCirurgia.xlsx` que pode ser encontrado [aqui](#). Salve o mesmo no seu diretório de trabalho. Este banco de dados contém 144 recém-nascidos submetidos a diferentes procedimentos cirúrgicos.

As variáveis disponíveis são:

- **id**: identificação do neonato;
- **sexo**: sexo do recém-nascido, `fem` e `masc`;
- **peso**: peso do neonato em gramas;
- **tempohosp**: tempo de hospitalização em dias;
- **infec**: presença de infecção secundária: sim e não;
- **cirurgia**: tipo de cirurgia: abdominal, cardíaca, outra.

A questão de pesquisa a ser respondida é:

Existe diferença no tempo de hospitalização (`tempohosp`) dos recém-nascidos de acordo com a presença ou não de infecção (`infec`)?

17.4.1.1 Leitura dos dados Os dados serão lidos com a função `read_excel()` do pacote `readxl`:

```
dados <- read_excel ("dadosCirurgia.xlsx")
glimpse(dados)

## # Rows: 144
## # Columns: 7
## $ id      <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 1~
## $ sexo     <chr> "masc", "masc", "masc", "masc", "fem", "fem", "fem", ~
## $ peso     <dbl> 2020, 1850, 2540, 1150, 2900, 2480, 2225, 3245, 2390, 2872, ~
## $ ig       <dbl> 36, 30, 38, 31, 36, 37, 38, 39, 38, 39, 40, 35, 38, 36, 36, ~
## $ tempohosp <dbl> 37, 37, 37, 46, 37, 36, 30, 18, 25, 14, 14, 9, 17, 15, 17, 1~
## $ infec    <chr> "não", "não", "sim", "sim", "não", "sim", "não", "sim", ~
## $ cirurgia <chr> "abdominal", "abdominal", "abdominal", "outra", "abdominal", ~
```

17.4.1.2 Exploração e visualização dos dados A função `summary()` cumpre um bom papel para observar o dados:

```
summary(dados)
```

```
##      id          sexo         peso          ig
## Min.   : 1.00  Length:144   Min.   :1150  Min.   :30.00
## 1st Qu.: 36.75 Class  :character  1st Qu.:2094  1st Qu.:35.00
## Median : 72.50 Mode   :character  Median :2455   Median :36.00
## Mean   : 72.50                   Mean   :2469   Mean   :36.51
## 3rd Qu.:108.25                   3rd Qu.:2840  3rd Qu.:38.00
## Max.  :144.00                   Max.  :3545   Max.  :41.00
##      tempohosp      infec        cirurgia
## Min.   : 1.00  Length:144   Length:144
## 1st Qu.: 20.75 Class  :character  Class  :character
## Median : 27.50 Mode   :character  Mode   :character
## Mean   : 37.42
## 3rd Qu.: 42.00
## Max.  :245.00
```

A variável `infec` aparece como caractere e será transformada como fator:

```
dados$infec <- as.factor(dados$infec)
```

```
glimpse(dados)
```

```
## # Rows: 144
## # Columns: 7
## $ id      <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 1~
## $ sexo     <chr> "masc", "masc", "masc", "masc", "fem", "fem", "fem", ~
## $ peso     <dbl> 2020, 1850, 2540, 1150, 2900, 2480, 2225, 3245, 2390, 2872, ~
## $ ig       <dbl> 36, 30, 38, 31, 36, 37, 38, 39, 38, 39, 40, 35, 38, 36, 36, ~
## $ tempohosp <dbl> 37, 37, 37, 46, 37, 36, 30, 18, 25, 14, 14, 9, 17, 15, 17, 1~
## $ infec    <fct> não, não, sim, sim, não, sim, não, sim, não, não, ~
## $ cirurgia <chr> "abdominal", "abdominal", "abdominal", "outra", "abdominal", ~
```

Os boxplots (Figura 182) são uma boa maneira de visualizar os dados:

```
ggplot (dados,
  aes (x=infec,
       y =tempohosp,
       color = infec,
       fill = infec,
       alpha = 0.5)) +
  geom_boxplot (outlier.colour = "black",
                outlier.size=1.5,
                color = "black") +
  scale_color_nejm () +
  scale_fill_nejm () +
  theme_classic () +
  theme (legend.position = "none") +
  ylab ("Tempo de hospitalização (dias)") +
  xlab ("Presença de infecção") +
  theme (text = element_text (size = 12))
```

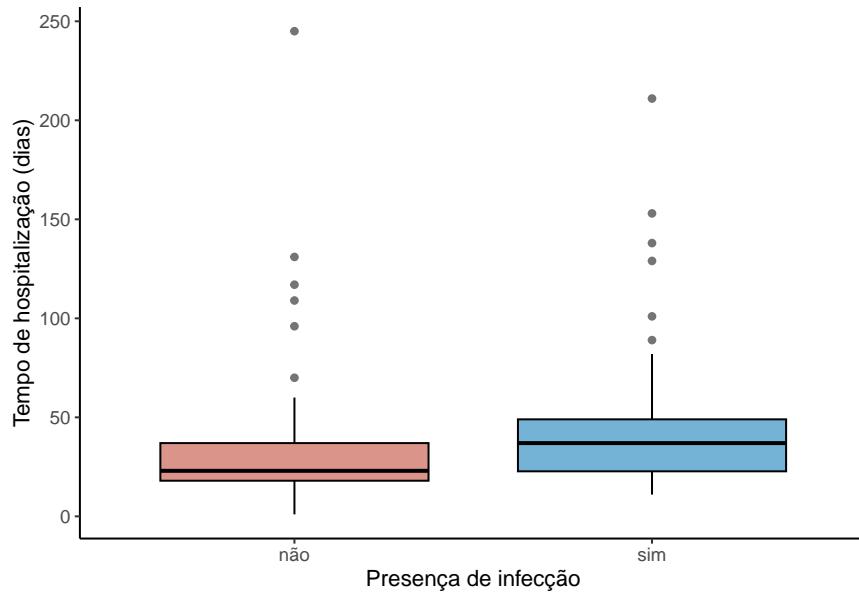


Figura 182: Impacto da infecção no tempo de hospitalização.

Os boxplots exibem uma série de valores atípicos, indicando que existe uma assimetria em ambos os grupos. A normalidade da variável também pode ser verificada usando o teste de Shapiro-Wilk.

```
dados %>%
  group_by(infec) %>%
  shapiro_test(tempohosp)
```

```
## # A tibble: 2 x 4
##   infec variable  statistic      p
##   <fct>  <chr>     <dbl>    <dbl>
## 1 não    tempohosp  0.565 9.87e-15
## 2 sim    tempohosp  0.692 1.47e- 9
```

17.4.1.3 Sumarização dos dados Como a variável `tempohosp` é assimétrica conforme mostrado acima, onde ambos os valores P são menores do que 0,05, será realizado um sumário numérico com a obtenção da mediana e IIQ. Isto será feito através da função `group_by()` e `get_summary_stats()`, incluídas no pacote `dplyr` e `rstatix`, respectivamente.

```
resumo <- dados %>%
  group_by(infec) %>%
  dplyr::summarise(n = n(),
                   mediana = median(tempohosp, na.rm = TRUE),
                   p25=quantile(tempohosp, probs = 0.25, na.rm = TRUE),
                   p75=quantile(tempohosp, probs = 0.75, na.rm = TRUE))
resumo

## # A tibble: 2 x 5
##   infec     n mediana    p25    p75
##   <fct> <int>   <dbl> <dbl>   <dbl>
## 1 não      88     23    18     37
## 2 sim      56     37   22.8    49
```

Os dados mostram que a mediana de tempo de internação dos neonatos infectados é bem maior do que os não infectados. A diferença mediana e os intervalos de confiança desta diferença são:

```
infectado <- dplyr::filter(dados, infec == "sim") %>%
  select(tempohosp)
sem_infec <- dplyr::filter(dados, infec == "não") %>%
  select(tempohosp)

dif_mediana <- median(infectado$tempohosp) - median(sem_infec$tempohosp)
dif_mediana
```

```
## [1] 14
```

```
ci_quantile_diff(infectado, sem_infec)
```

```
##
## Two-sided 95% bootstrap confidence interval for the population value of
## 50% quantile(x) - 50% quantile(y) based on 9999 bootstrap replications
## and the bca method
##
## Sample estimate: 14
## Confidence interval:
##  2.5% 97.5%
##  4.5 19.5
```

Para calcular intervalos de confiança para a diferença das medianas, usou-se a função `ci_quantile_diff()` do pacote `confintr` (144).

Dessa forma, a mediana da diferença entre os dois grupos foi de 14 dias (IC95%; 5 - 20).

17.4.2 Hipóteses estatísticas

Da mesma maneira que o teste t , as hipóteses estabelecidas comparam dois grupos independentes. Se não houver diferença entre os grupos, ou seja, os grupos são provenientes de uma mesma população, as somas dos postos em cada grupo devem ficar próximas. Desta forma,

$H_0 \rightarrow$ as duas populações são iguais

$H_A \rightarrow$ as duas populações não são iguais

Não foi escrita a hipótese nula como sendo de que as médias (ou as medianas) são iguais, pois o tese não usa as medidas de posição tradicionais e sim os postos.

17.4.3 Pressupostos do teste de Mann_Whitney

O teste de Mann-Whitney é baseado nos seguintes pressupostos:

1. Os dados são aleatórios;
2. As amostras são de dois grupos independentes;
3. Um dos grupos é denominado de 1 e o outro de 2;
4. A variável a ser comparada nos grupos deve ser ordenável;
5. O grupo 1 será o grupo de menor tamanho e, se tiverem o mesmo tamanho, o grupo 1 é aquele cuja soma dos postos é a menor.

17.4.4 Execução do teste estatístico

17.4.4.1 Lógica do teste U de Mann-Whitney De acordo com as hipóteses estabelecidas, o teste é bicaudal. Se as observações nos dois grupos forem provenientes da mesma população, a soma dos postos em cada grupo devem ficar próximas.

Para calcular o teste, procede-se da seguinte maneira:

1. Deve haver uma variável que identifique o grupo a que pertence cada uma das observações. No exemplo proposto, a variável desfecho é `tempohosp` e a variável agrupadora é `infec`, categorizada como `sim` e `não`.
2. Ordenar de forma crescente todos os valores da variável `tempohosp`, sem levar em consideração a que grupo pertence. Para realizar este procedimento, será usada a função `rank()` do R base com o método para empates igual à média dos valores empataos (`ties.method="average"`). Ao executar a função, será criada uma nova variável, denotada `posto`.

```
dados$posto <- rank(dados$tempohosp, ties.method = "average")
```

```
head(dados)
```

```
## # A tibble: 6 x 8
##       id sexo   peso    ig tempohosp infec cirurgia  posto
##     <dbl> <chr> <dbl> <dbl>      <dbl> <fct> <chr>     <dbl>
## 1     1 masc   2020     36        37 não    abdominal  94.5
## 2     2 masc   1850     30        37 não    abdominal  94.5
## 3     3 masc   2540     38        37 sim    abdominal  94.5
## 4     4 masc   1150     31        46 sim    outra     120
## 5     5 masc   2900     36        37 não    abdominal  94.5
## 6     6 fem    2480     37        36 sim    abdominal  91
```

3. Verificar o tamanho (n) de cada grupo (presença ou não de infecção) e somar os postos em cada um dos grupos, usando a função `group_by()` junto com a função `summarise()`,

```
resumo <- dados %>%
  dplyr::group_by(infec) %>%
  dplyr::summarise(n = n(),
                    soma = sum(posto))
resumo

## # A tibble: 2 x 3
##   infec     n   soma
##   <fct> <int> <dbl>
## 1 não      88  5564.
## 2 sim      56  4876.
```

4. Denominar de *grupo_1* o grupo com menor soma:

```
grupo_1 <- min(resumo$soma)
grupo_1
```

```
## [1] 4875.5
```

5. Denotar o *grupo_1* como T

```
T <- grupo_1
```

Consequentemente,

```
n1 <- 56
n2 <- 88
```

6. Calcular a estatística do teste, usando a fórmula preconizada por Altman (145):

$$U = n_1 \times n_2 + \left[\frac{n_1 \times (n_1 + 1)}{2} \right] - T$$

```
U <- (n1*n2 + ((n1*(n1 + 1))/2)) - T
U
```

```
## [1] 1648.5
```

Obs.: O U de Mann-Whitney aparece no teste de Wilcoxon como W, eles são iguais

7. Se $n_1, n_2 \geq 10$, a distribuição da estatística do teste pode ser aproximada por uma distribuição normal com média igual a

$$\mu_U = \left[\frac{n_S \times (n_L + 1)}{2} \right]$$

onde n_S e n_L , são, respectivamente, o grupo de menor e maior tamanho. No exemplo, n_1 e n_2 .

```
m_U <- (n1*(n1+n2+1))/2  
m_U
```

```
## [1] 4060
```

E desvio padrão igual a

$$\sigma_U = \sqrt{\frac{n_L \times \sigma_U}{6}}$$

```
dp_U <- sqrt((n2*m_U)/6)  
dp_U
```

```
## [1] 244.0219
```

Os resultados fornecem os dados para calcular a estatística Z_U com correção de continuidade e, a partir dela, calcular o valor P .

$$Z_U = \frac{(T - 0,5) - \mu_U}{\sigma_U}$$

```
Z_U <- ((T - 0.5) - m_U)/dp_U  
round(Z_U, 2)
```

```
## [1] 3.34
```

8. Finalmente, calcula-se o valor P , usando a função `pnorm()`, multiplicada por 2, pois o teste é bicaudal.

```
P <- pnorm(Z_U, lower.tail = FALSE) * 2  
round(P, 4)
```

```
## [1] 8e-04
```

Na prática, não há necessidade de fazer todos esses cálculos, pois o R calcula facilmente o teste. Os cálculos foram mostrados para melhorar o entendimento de como o teste de Mann-Whitney funciona.

17.4.4.2 Cálculo do U de Mann-Whitney no R O cálculo será realizado, usando a função `wilcox.test()` do pacote `stats`, incluído no R base. Este teste necessita alguns argumentos, consulte `?wilcox.test` para maiores detalhes.

```
teste1 <- wilcox.test(formula = tempohosp ~ infec, data = dados)  
teste1
```

```
##  
## Wilcoxon rank sum test with continuity correction  
##  
## data: tempohosp by infec  
## W = 1648.5, p-value = 0.0008302  
## alternative hypothesis: true location shift is not equal to 0
```

O teste também pode ser realizado com a função `wilcox_test()` do pacote `rstatix` que fornece o mesmo resultado.

```
teste2 <- wilcox_test(formula = tempohosp ~ infec, data = dados)
teste2
```

```
## 
##   Asymptotic Wilcoxon-Mann-Whitney Test
##
## data:  tempohosp by infec (não, sim)
## Z = -3.3446, p-value = 0.0008241
## alternative hypothesis: true mu is not equal to 0
```

Ambos teste entregam o mesmo resultado, mostrando uma diferença estatisticamente significativa ($P = 0,00083$) entre os tempos de hospitalização dos recém-nascidos que realizaram cirurgia no período neonatal que se infectaram ou não (diferença mediana = 14 dias (4,5 - 20)).

17.4.5 Tamanho do efeito

É interessante calcular o tamanho do efeito, a magnitude do efeito. O tamanho do efeito r é calculado como a estatística Z_U dividida pela raiz quadrada do tamanho da amostra (n).

$$r = \frac{Z_U}{\sqrt{n}}$$

O valor de Z_U é igual a 3.3398648, logo

```
r <- Z_U/sqrt(n1+n2)
round(r,3)
```

```
## [1] 0.278
```

O R possui a função `wilcox_effsize()` do pacote `rstatix` e necessita também do pacote `coin` (146) , instalado para calcular a estatística r . A Saída exibirá junto a magnitude o efeito, que no caso é pequena (veja Tabela 22).

```
wilcox_effsize(dados, tempohosp~infec)
```

```
## # A tibble: 1 x 7
##   .y.     group1 group2 effsize    n1    n2 magnitude
## * <chr>   <chr>   <chr>   <dbl> <int> <int> <ord>
## 1 tempohosp não     sim      0.279    88    56 small
```

17.4.6 Conclusão

O valor P do teste é 8.3819198×10^{-4} , bem abaixo do nível de significância estabelecido ($\alpha = 0,05$). Pode-se concluir que o tempo de hospitalização nos dois grupos é estatisticamente diferente. Entretanto, a magnitude dessa diferença é pequena ($0,10 < 0,30, 0,30 < 0,50, >= 0,50$).

Isto pode ser visualizado no gráfico (Figura 183):

Tabela 22: Interpretação do valor r (sem considerar o sinal)

Valor r	Magnitude
$0,10 < 0,30$	pequeno
$0,30 < 0,50$	médio
$\geq 0,50$	grande

```

bxp <- ggplot (dados,
  aes (x=infec,
       y =tempohosp,
       color = infec)) +
  geom_boxplot (outlier.colour = "black",
                outlier.size=1.5) +
  theme_classic () +
  scale_color_nejm () +
  theme (legend.position = "none") +
  ylab ("Tempo de hospitalização (dias)") +
  xlab ("Presença de infecção") +
  theme (text = element_text (size = 12))

teste <- dados %>%
  rstatix::wilcox_test(tempohosp ~ infec) %>%
  add_significance()

teste <- teste %>% add_xy_position(x = "infec")
bxp +
  stat_pvalue_manual(teste, tip.length = 0) +
  labs(subtitle = get_test_label(teste, detailed = TRUE))

```

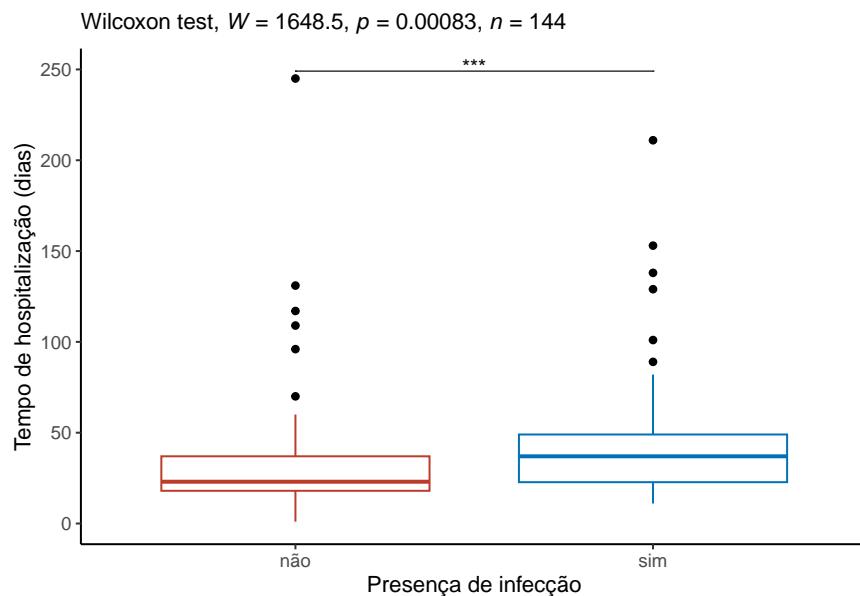


Figura 183: Impacto da infecção no tempo de hospitalização.

17.5 Teste de Wilcoxon

O teste de Wilcoxon, também conhecido como teste dos postos com sinais de Wilcoxon (*Wilcoxon Signed-Rank Test*), é um teste não paramétrico utilizado em situações em que existem dois conjuntos de dados emparelhados, ou seja, dados provenientes do mesmo participante. O teste não examina os dois grupos individualmente; em vez disso, ele se concentra na diferença existente entre cada par de observações. É um equivalente não paramétrico do teste t pareado.

17.5.1 Dados

Para verificar se a realização de exercícios aeróbicos modifica a função respiratória de 10 escolares asmáticos, foi medido o Pico de Fluxo Expiratório Máximo (*Peak Flow Meter*) no início e no final do programa, após 120 dias. O Pico de Fluxo Expiratório Máximo (PFE) serve como uma forma simples de avaliar a força e a velocidade de saída do ar de dentro dos pulmões. É medido em L/min. Os resultados do estudo tem apenas três variáveis, `id`, `basal` e `final`.

```
id <- c(1:10)
basal <- c(120, 200, 140, 200, 110, 240, 150, 120, 250, 190)
final <- c(220, 300, 230, 180, 300, 330, 230, 250, 300, 200)

dados <- data.frame(id, basal, final)

glimpse (dados)

## #> #> #> Rows: 10
## #> #> #> Columns: 3
## #> #> #> $ id      <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
## #> #> #> $ basal   <dbl> 120, 200, 140, 200, 110, 240, 150, 120, 250, 190
## #> #> #> $ final   <dbl> 220, 300, 230, 180, 300, 330, 230, 250, 300, 200
```

A questão de pesquisa a ser respondida, portanto, é:

Existe diferença entre as medidas iniciais e finais do PFE dos escolares asmáticos que entraram em um programa de exercícios aeróbicos?

17.5.1.1 Exploração e transformação dos dados Os dados estão no formato amplo com as variáveis `basal` e `final` classificadas como numéricas. Será transformado para o formato longo, usando a função `pivot_longer()` do pacote `tidyverse`. Este processo é opcional, mas, como foi feito com o teste t pareado, será repetido aqui:

```
dadosL <- dados %>%
  pivot_longer(c(basal, final),
               names_to = "momento",
               values_to = "medidas")
```

O conjunto de dados `dadosL` passa a ter a seguinte estrutura:

```
glimpse(dadosL)
```

```
## #> #> #> Rows: 20
## #> #> #> Columns: 3
## #> #> #> $ id      <int> 1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 9, 9, 10, 10
## #> #> #> $ momento <chr> "basal", "final", "basal", "final", "basal", "final", "basal", ~
## #> #> #> $ medidas <dbl> 120, 220, 200, 300, 140, 230, 200, 180, 110, 300, 240, 330, 15~
```

17.5.1.2 Medidas resumidoras Como o número de participantes é de apenas 10, a medida de posição mais adequada para resumir os dados é mediana e a medida de dispersão é o intervalo interquartil (IIQ). Para isso, se fará uso das funções `group_by()` e `summarise()` do pacote `dplyr`:

```
resumo <- dadosL %>%
  group_by(momento) %>%
  dplyr::summarise(n = n(),
    mediana = median (medidas, na.rm = TRUE),
    p25=quantile(medidas, probs = 0.25, na.rm = TRUE),
    p75=quantile(medidas, probs = 0.75, na.rm = TRUE),
    media = mean (medidas, na.rm = TRUE),
    dp = sd (medidas, na.rm = TRUE),
    ep = dp/sqrt(n),
    me = ep * qt(1 - (0.05/2), n - 1))
resumo

## # A tibble: 2 x 9
##   momento     n mediana   p25   p75 media     dp     ep     me
##   <chr>     <int>   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 basal       10     170    125    200   172  50.9  16.1  36.4
## 2 final       10     240    222.   300   254  50.4  15.9  36.0
```

17.5.1.3 Visualização dos dados Boxplot (Figura 184)

```
dadosL %>%
  ggplot(aes(x = momento, y = medidas, fill = momento)) +
  geom_errorbar(stat = "boxplot", width = 0.1) +
  geom_boxplot (outlier.color = "red",
    outlier.shape = 1,
    outlier.size = 1) +
  scale_fill_manual(values = c("cyan4","cyan3")) +
  ylab("Volume Forçado em 1 seg (L)") +
  xlab("Momento") +
  theme_classic() +
  theme(text = element_text(size = 12)) +
  theme(legend.position = "none")
```

Gráfico de linha (Figura 185)

```
resumo %>%
  ggplot(aes(x=momento, y=media, group=1)) +
  geom_point(size = 2) +
  geom_line(linetype ='dashed') +
  geom_errorbar(aes(ymin=media - me,
    ymax=media + me),
    width=0.1,
    size = 1,
    col = c("cyan4","cyan3")) +
  theme_classic()+
  labs(x='Momento',
    y='Volume Forçado em 1 seg (L)')
```

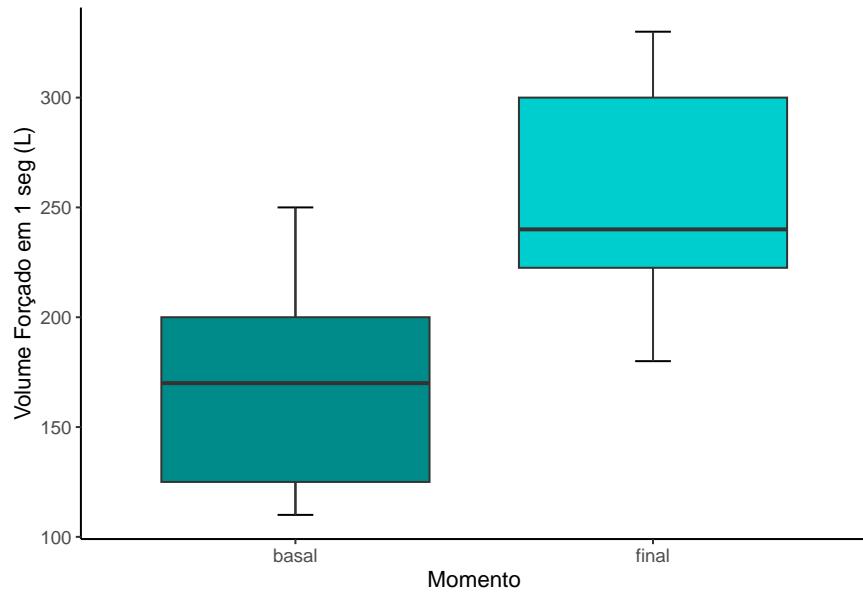


Figura 184: Impacto de exercícios aeróbicos na função respiratória de 10 escolares asmáticos.

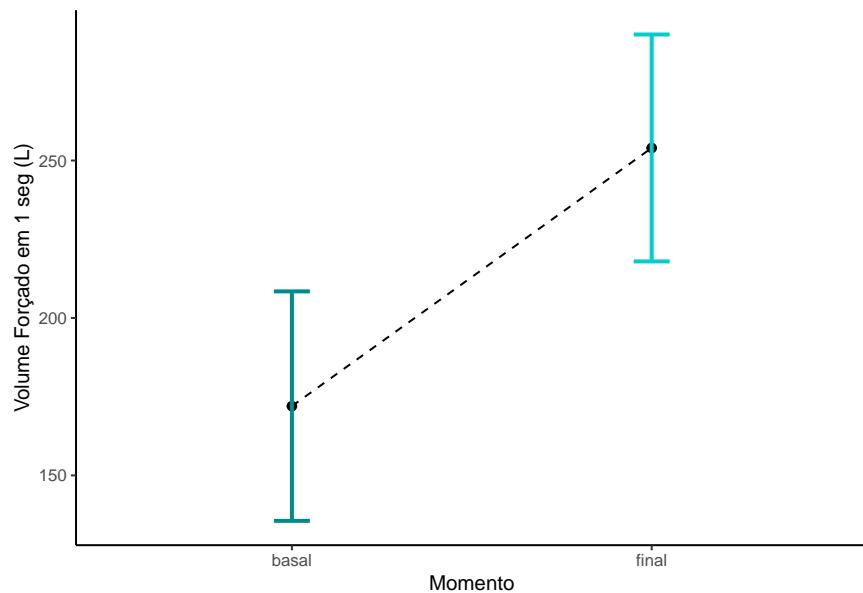


Figura 185: Impacto de exercícios aeróbicos na função respiratória de 10 escolares asmáticos.

17.5.1.4 Criação de uma variável que represente a diferença entre os momentos A diferença entre as média basal e final será atribuída ao nome D. Esta ação será realizada, utilizando o banco de dados amplo (dados):

```
dados$D <- dados$final - dados$basal  
head (dados)
```

```
##   id basal final   D  
## 1  1    120    220 100  
## 2  2    200    300 100  
## 3  3    140    230  90  
## 4  4    200    180 -20  
## 5  5    110    300 190  
## 6  6    240    330  90
```

Resumo da variável D

Ao resumo será atribuído ao nome `sumario` (sem acento):

```
sumario <- dados %>%  
  dplyr::summarise(n = n (),  
                    mediana = median (D, na.rm = TRUE),  
                    p25=quantile(D, probs = 0.25, na.rm = TRUE),  
                    p75=quantile(D, probs = 0.75, na.rm = TRUE))  
sumario  
  
##     n mediana  p25 p75  
## 1 10      90 57.5 100
```

O sinal negativo demonstra que houve um aumento do PFM do momento basal para o final.

17.5.2 Definição das hipóteses estatísticas

Da mesma maneira que o teste t pareado, as hipóteses estabelecidas comparam dois grupos dependentes. O teste de Wilcoxon é usado para avaliar a hipótese nula de que a distribuição das diferenças entre os grupos tem uma diferença mediana igual a 0.

$$H_0 \rightarrow D_i = 0$$

$$H_A \rightarrow D_i \neq 0$$

Note que a H_A estabelece que a diferença pode aumentar ou diminuir. Logo, o teste é bicaudal.

17.5.3 Execução do teste estatístico

17.5.3.1 Lógica do teste de Wilcoxon

1. A ideia do teste é verificar se as diferenças positivas são maiores ou menores, em grandeza absoluta, que as diferenças negativas. Para isso, foi criada, anteriormente, a variável D. Agora, será criada outra variável, iguala a variável D, apenas ignorando o sinal, denominada `D_abs`, diferença absoluta entre as variáveis `final` e `basal`.

```
dados$D_abs <- abs(dados$final - dados$basal)
```

2. Excluir os casos com diferença igual a 0 (zero). Para isso, uma maneira possível é extrair um subconjunto de dados do conjunto principal (`dados`), criando um conjunto de dados com a função `filter()` do pacote `dplyr`, que receberá o nome de `dados1`. O argumento `D_abs != 0` significa todas as diferenças absolutas diferentes de 0:

```
dados1 <- dados %>% dplyr::filter(D_abs != 0)
```

Observe que como não há diferenças zeradas. Ou seja, o novo conjunto de dados continua o mesmo. O que pode ser confirmado, executando a função `glimpse()`:

```
glimpse (dados1)
```

```
## #> #> #> #> #>
```

```
## #> Rows: 10
## #> Columns: 5
## #> #> $ id      <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
## #> #> $ basal   <dbl> 120, 200, 140, 200, 110, 240, 150, 120, 250, 190
## #> #> $ final   <dbl> 220, 300, 230, 180, 300, 330, 230, 250, 300, 200
## #> #> $ D       <dbl> 100, 100, 90, -20, 190, 90, 80, 130, 50, 10
## #> #> $ D_abs   <dbl> 100, 100, 90, 20, 190, 90, 80, 130, 50, 10
```

3. Ordenar de forma crescente todos os valores da variável `D_abs` do banco de dados `dados1`, usando a função `arrange()` do pacote `dplyr`:

```
dados1 <- dados1 %>% arrange(dados1$D_abs)
```

4. Estabelecer postos para os valores ordenados da variável `D_abs`, do conjunto de dados `dados1`, fazendo a média das ordens quando houver empate. A execução deste comando cria uma nova variável, chamada `ponto`:

```
dados1$ponto <- rank(dados1$D_abs)
```

5. Estabelecer sinais para os postos, criando dois subconjuntos de dados do conjuntos `dados1`, um com os escolares com postos positivos (`pos`) e outros com postos negativos(`neg`):

```
neg <- dados1 %>% dplyr::filter(D < 0)
pos <- dados1 %>% dplyr::filter(D > 0)
```

6. Somar todos os postos (variável `ponto`) em cada um dos subconjuntos criados (`neg` e `pos`):

```
soma_neg <- sum(neg$ponto)
soma_pos <- sum(pos$ponto)
print (c(soma_neg, soma_pos))
```

```
## [1] 2 53
```

7. Atribuir a menor soma à estatística do teste, denotada T :

```
T <- min (soma_neg : soma_pos)
T
```

```
## [1] 2
```

No teste `wilcox.test`, $T = V$.

8. Para dados com tamanhos grandes (> 20 pares), a significância de T pode ser determinada (147), considerando que a distribuição de T tem aproximadamente distribuição normal com média igual a

$$\mu_T = \frac{n \times (n + 1)}{4}$$

onde n é o tamanho da amostra.

```
n <- sumario$n
n
```

```
## [1] 10
```

```
mu_T <- (n * (n + 1))/4
mu_T
```

```
## [1] 27.5
```

E desvio padrão igual a:

$$\sigma_T = \sqrt{\frac{n(n + 1) \times (2n + 1)}{24}}$$

```
dp_T <- sqrt ((n*(n + 1)) * (2 * n + 1) /24)
dp_T
```

```
## [1] 9.810708
```

Os resultados da execução das equações fornecem os dados para calcular a estatística Z_T com correção de continuidade e, a partir dela, calcular o valor P .

$$Z_T = \frac{|T - \mu_T| - 0,5}{\sigma_T}$$

```
Z_T <- (abs(T - mu_T) - 0.5)/dp_T
Z_T
```

```
## [1] 2.548236
```

9. Concluindo, o valor da estatística de teste T é superior ao $Z_{crtico} = 1,96$, para um $\alpha = 0,05$. Desse forma, não rejeita-se a H_0 . Há diferença significativa entre o PFE basal e o PFE final, neste grupo de escolares asmáticos.
10. O valor P pode ser obtido com a função `pnorm()` e multiplicando o resultado por 2, pois o teste é bilateral.

```
P <- pnorm (Z_T, lower.tail = FALSE) * 2
round(P, 4)
```

```
## [1] 0.0108
```

O valor $P < 0,05$, logo, rejeita-se a H_0 .

Como já dito anteriormente, na prática, não há necessidade de fazer todos esses cálculos, pois o R calcula facilmente o teste. Eles são apenas uma demonstração de como o teste funciona.

17.5.3.2 Cálculo do teste de Wilcoxon no R Usando o conjunto de dados no formato longo (dadosL), calcula-se o teste, usando a função `wilcox.test()`, do pacote `stats`:

```
teste1 <- wilcox.test(medidas ~ momento, data = dadosL, paired = TRUE)
```

```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): não é
## possível computar o valor de p exato com o de desempate
```

```
teste1
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  medidas by momento
## V = 2, p-value = 0.01072
## alternative hypothesis: true location shift is not equal to 0
```

O pacote `rstatix` também tem uma função que permite chegar ao mesmo resultado:

```
teste2 <- dadosL %>%
  rstatix::wilcox_test(medidas ~ momento, paired = TRUE) %>%
  add_significance()
teste2
```

```
## # A tibble: 1 x 8
##   .y.    group1 group2   n1   n2 statistic   p p.signif
##   <chr>  <chr>  <chr>  <int> <int>    <dbl> <dbl> <chr>
## 1 medidas basal final     10    10      2 0.0107 *

```

Os resultados são iguais, apenas são formas diferentes de se executar o teste de Wilcoxon.

17.5.4 Tamanho do efeito

O tamanho do efeito pode ser calculado da mesma forma que para o teste de Mann-Whitney, usando a mesma equação e os dados obtidos acima, onde 2.548236 e 10 tem-se

$$r = \frac{Z_T}{\sqrt{n}}$$

Pode-se usar também a função `wilcox_effsize()` para calcular a estatística r . A Saída exibe junto a magnitude o efeito, que no caso é grande ($> 0,5$ como mostra a Tabela 2 do teste de Mann-Whitney).

```

dadosL %>%
  wilcox_effsize(medidas ~ momento, paired = TRUE)

## # A tibble: 1 x 7
##   .y.    group1 group2 effsize    n1    n2 magnitude
## * <chr>  <chr>  <chr>    <dbl> <int> <int> <ord>
## 1 medidas basal final     0.823    10    10 large

```

17.5.5 Conclusão

Assumindo um $\alpha = 0,05$, se o valor P , obtido pelo teste, for menor do que 0,05, rejeita-se a hipótese nula ($V = 2$, $P = 0,01$, $n = 10$).

Pode-se concluir que existe diferença nas medidas do pico de fluxo expiratório máximo no início e no fim do programa de exercícios aeróbicos realizados pelos escolares asmáticos e a magnitude do efeito foi grande ($r = 0,82$).

Isto pode ser visualizado na Figura 186:

```

bxp <- ggplot (dadosL,
               aes (x=momento,
                     y =medidas,
                     color = momento)) +
  geom_boxplot (outlier.colour = "black",
                outlier.size=1.5) +
  theme_classic () +
  scale_color_nejm () +
  theme (legend.position = "none") +
  ylab ("Dosagem do Cortisol (microgramas/L)") +
  xlab ("Momento") +
  theme (text = element_text (size = 12))

teste <- dadosL %>%
  rstatix::wilcox_test (medidas ~ momento, paired = TRUE) %>%
  add_significance ()
teste <- teste %>% add_xy_position ()

bxp +
  stat_pvalue_manual (teste,
                      tip.length = 0) +
  labs (subtitle = get_test_label (stat.test = teste,
                                 detailed = TRUE))

```

17.6 Teste de Kruskal-Wallis

Quando os pressupostos subjacentes a ANOVA não são atendidos, é possível usar o teste não paramétrico de Kruskal-Wallis (KW) para testar a hipótese de que os parâmetros de localização são iguais. Pode ser considerado uma extensão do teste de Wilcoxon-Mann-Whitney.

Enquanto a ANOVA depende da hipótese de que todas as populações são independentes e normalmente distribuídas, o teste de Kruskal-Wallis exige apenas amostras aleatórias independentes provenientes de suas

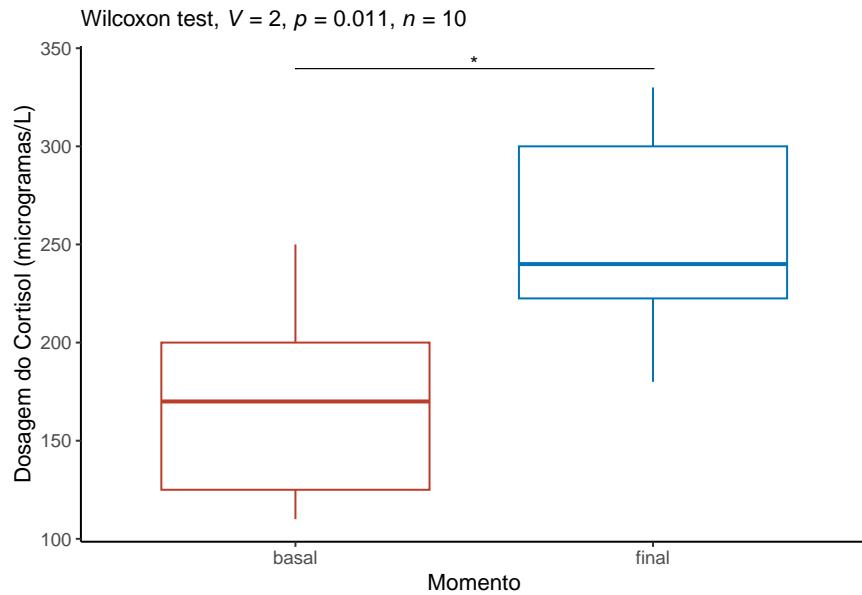


Figura 186: Impacto de exercícios aeróbicos na função respiratória de 10 escolares asmáticos.

respectivas populações. Entretanto, este teste somente deve ser aplicado se a amostra for pequena e/ou os pressupostos para a ANOVA forem seriamente violados.

O teste não usa diretamente medições de quantidade conhecida, utiliza, como outros testes não paramétricos, os postos dos valores analisados. Em função disso, é também conhecido como

17.6.1 Dados

Um experimento foi realizado para verificar se o álcool ou o café afetam os tempos de reação ao dirigir (148). O estudo tem três grupos diferentes de participantes: 10 bebendo água (controle), 10 bebendo cerveja contendo duas unidades de álcool e 10 bebendo café. O tempo de reação em uma simulação de direção foi medido para cada participante.

Os dados encontram-se no arquivo `dadosResposta.xlsx`. Clique [aqui](#) para baixar e, após, salve o mesmo no seu diretório de trabalho.

As variáveis são:

- **id**: identificação do participante;
- **tempo**: tempo de reação na simulação de direção em segundos;
- **bebida**: três grupo: água, álcool e café.

O estudo pretende verificar se existe diferença no tempo de reação dos participantes em um teste de direção com a ingestão de água, café e álcool.

```
dados <- read_excel ("dadosResposta.xlsx")
```

17.6.1.1 Leitura dos dados

17.6.1.2 Exploração e visualização dos dados

Visão geral das variáveis com a função `summary()`:

```
summary (dados)

##      id          tempo         bebida
##  Min.   : 1.00   Min.   :0.3700  Length:30
##  1st Qu.: 8.25   1st Qu.:0.9575  Class :character
##  Median :14.50   Median :1.5450  Mode  :character
##  Mean   :14.83   Mean   :1.6500
##  3rd Qu.:21.75   3rd Qu.:1.9700
##  Max.   :30.00   Max.   :3.4700
```

A variável `bebida` encontra-se como caracter e deve ser fator:

```
dados$bebida <- factor(dados$bebida,
                         levels = c("agua", "cafe", "alcool"))

glimpse (dados)

## #> #> Rows: 30
## #> #> Columns: 3
## #> #> $ id      <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 11, 12, 13, 14, 15, 16, 17, 9, 21, 22, ~
## #> #> $ tempo   <dbl> 0.37, 0.38, 0.61, 0.78, 0.83, 0.86, 0.90, 0.95, 0.98, 1.11, 1.2~
## #> #> $ bebida <fct> agua, agua, agua, agua, agua, agua, agua, agua, agua, cafe, cafe, caf~
```

Os dados serão observados visualmente através de boxplots (Figura 187), usando a função `ggplot()` do pacote `ggplot2`, com cores do `nejm` (*New England Journal of Medicine*) o pacote `ggsci`.

```
ggplot(dados,
       aes(x=bebida,
           y =tempo,
           color = bebida,
           fill = bebida,
           alpha = 0.5)) +
  geom_boxplot(outlier.colour = "black",
               outlier.size=1.5,
               color = "black") +
  scale_color_nejm() +
  scale_fill_nejm() +
  theme_classic() +
  theme(legend.position = "none") +
  ylab ("Tempo (seg)") +
  xlab ("Bebida") +
  theme(text = element_text(size = 12))
```

Os boxplots exibem dados com medianas visualmente diferentes, bigodes diferentes e grupos com presença de *outliers*. Para verificar o impacto desses achados, pode-se usar a função `identify_outliers()`, do pacote `rstatix` que confirma na sua Saída a presença de *outliers* no grupo agua e cafe, sendo dois extremos.

```
dados %>% group_by(bebida) %>%
  identify_outliers(tempo)
```

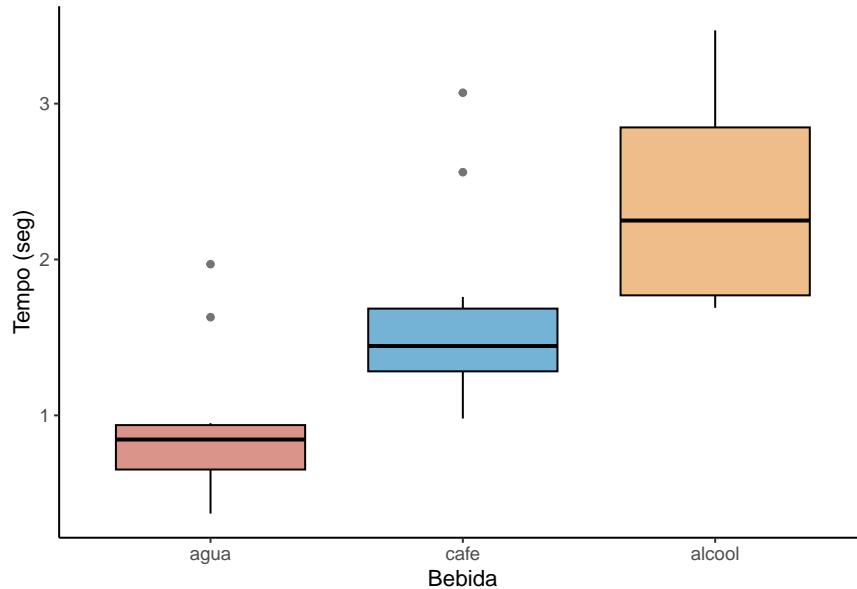


Figura 187: Impacto do tipo de bebida no tempo de reação ao dirigir.

```
## # A tibble: 4 x 5
##   bebida    id tempo is.outlier is.extreme
##   <fct>   <dbl> <dbl>   <lgl>
## 1 agua      9  1.63 TRUE
## 2 agua     10  1.97 TRUE
## 3 cafe     19  2.56 TRUE
## 4 cafe     20  3.07 TRUE
```

17.6.1.3 Sumarização e avaliação da normalidade dos dados Para avaliar a normalidade será usado o teste de Shapiro-Wilk, com a função `shapiro_test()` e a função `group_by()` do pacote `dplyr`:

```
dados %>% group_by (bebida) %>%
  shapiro_test (tempo)
```

```
## # A tibble: 3 x 4
##   bebida variable statistic      p
##   <fct>   <chr>     <dbl>   <dbl>
## 1 agua     tempo     0.863  0.0837
## 2 cafe     tempo     0.815  0.0220
## 3 alcool   tempo     0.875  0.114
```

A variável `cafe` tem uma distribuição que não se ajusta a distribuição normal.

Para completar a exploração dos dados, será solicitado, usando as funções `group_by ()` e `summarise`, do pacote `dplyr`, medidas de localização e dispersão adequadas para variáveis bem assimétricas.

```
resumo <- dados %>%
  group_by(bebida) %>%
  dplyr::summarise(n = n(),
                   mediana = median (tempo, na.rm = TRUE),
```

```

p25=quantile(tempo, probs = 0.25, na.rm = TRUE),
p75=quantile(tempo, probs = 0.75, na.rm = TRUE))
resumo

```

```

## # A tibble: 3 x 5
##   bebida     n mediana   p25   p75
##   <fct>   <int>   <dbl> <dbl> <dbl>
## 1 agua      10    0.845  0.653  0.937
## 2 cafe      10    1.44   1.28   1.68
## 3 alcool    10    2.25   1.77   2.85

```

17.6.2 Hipóteses estatísticas

Se não houver diferença entre os grupos, ou seja, os grupos são provenientes de uma mesma população, as somas dos postos em cada grupo devem ficar próximas. Desta forma,

$H_0 \rightarrow$ as populações são iguais

$H_A \rightarrow$ pelo menos uma das populações tende a exibir valores diferentes do que as outras populações

17.6.3 Pressupostos do teste

O teste de Kruskal-Wallis pressupõe as seguintes condições para o seu adequado uso:

1. As amostras são amostras aleatórias independentes de suas respectivas populações;
2. A escala de medição utilizada é pelo menos ordinal e, se houver apenas três grupos, deve haver pelo menos 5 casos em cada grupo;
3. As distribuições dos valores nas populações amostradas são idênticas, exceto pela possibilidade de que uma ou mais das populações sejam compostas por valores que tendem a ser maiores do que os das outras populações.

17.6.4 Execução do teste estatístico

17.6.4.1 Lógica do teste de Kruskall-Wallis A teoria do teste Kruskal-Wallis é semelhante à do teste de Mann-Whitney, ou seja, tem como base a soma dos postos. Em primeiro lugar, os escores são ordenados do menor para o maior, independentemente do grupo que pertençam.

O menor recebe o posto 1 e assim por diante. Após a atribuição dos postos, soma-se os postos por grupo. A soma dos postos de cada grupo é representada por $R_1, R_2, R_3, \dots, R_i$. A estatística do teste, H , é calculada com a equação (149):

$$H = \frac{12}{N \times (N+1)} \sum_{i=1}^k \frac{R_i^2}{n_i} - 3 \times (N+1)$$

onde n_i é o número de observações no grupo i , $N = \sum_{i=1}^k n_i$ (o número total de observações em todos os k grupos) e R_i é a soma dos postos das n_i observações no grupo i .

Uma boa verificação (mas não uma garantia) de que os postos foram atribuídos corretamente é ver se a soma de todos os postos é igual a $\frac{N \times (N+1)}{2}$.

1. Criar a variável `posto` com os postos ordenados de forma crescente, independente do grupo, como realizado no teste de Mann-Whitney:

```

dados$posto <- rank(dados$tempo, ties.method = "average")

head(dados)

## # A tibble: 6 x 4
##       id tempo bebida posto
##   <dbl> <dbl> <fct>   <dbl>
## 1     1  0.37 agua      1
## 2     2  0.38 agua      2
## 3     3  0.61 agua      3
## 4     4  0.78 agua      4
## 5     5  0.83 agua      5
## 6     6  0.86 agua      6

```

2. Somar os postos de cada grupo separadamente:

```

resumo <- dados %>%
  dplyr::group_by(bebida) %>%
  dplyr::summarise(n = n(),
                    soma = sum(posto))
resumo

## # A tibble: 3 x 3
##   bebida     n   soma
##   <fct>   <int> <dbl>
## 1 agua      10  74.5
## 2 cafe      10 157
## 3 alcool    10 234.

```

3. Cálculo da estatística do teste H

```

N <- 30
n <- 10
R_agua <- resumo[1,3]
R_alcool <- resumo[2,3]
R_cafe <- resumo[3,3]

H <- (12/(N*(N+1))) * ((R_agua^2/n) + (R_alcool^2/n) + (R_cafe^2/n)) - (3*(N+1))
H

##       soma
## 1 16.31806

```

4. Cálculo do Valor P

Se existir três grupos, com cinco ou menos participantes em cada grupo, há necessidade de usar a tabela especial para tamanhos de amostra pequenos (150). Se você tiver mais de cinco participantes por grupo, trate H como qui-quadrado. A estatística H é estatisticamente significativa se for igual ou maior que o valor crítico qui-quadrado para o grau de liberdade específico, igual a $k - 1$. Aqui, tem-se 10 participantes por grupo e, assumindo um $\alpha = 0,05$, o $H_{critico}$ é igual a:

```

alpha <- 0.05
k <- 3
gl = k - 1
H_critico <- qchisq(1 - alpha, gl)
H_critico

```

```
## [1] 5.991465
```

Uma vez que o $H_{calculado} = 16,3$ é maior que $H_{critico} = 6,0$, rejeita-se a H_0 . O valor P obtido através da função pchisq():

```

H <- 16.32
pchisq(H, 2, lower.tail = FALSE)

```

```
## [1] 0.0002858624
```

Novamente, o R tem funções que fazem facilmente esses cálculos. Eles são colocados aqui apenas para ilustrar o raciocínio de como o teste funciona, para os mais curiosos!

17.6.4.2 Teste de Kruskal-Wallis no R Calcula-se o teste, usando a função `kruskal.test()` do pacote `stats`, incluído no R base. Esta função usa os seguintes argumentos:

- **x** → um vetor numérico de valores de dados;
- **g** → um vetor ou objeto fator que fornece o grupo para os elementos correspondentes de **x**. Ignorado com um aviso se **x** for uma lista;
- **formula** → fórmula do tipo *resposta ~ grupo*;
- **data** → conjunto de dados (matriz ou dataframe) contendo as variáveis da fórmula;
- ... → outros argumentos.

```
kruskal.test (tempo ~ bebida, data = dados)
```

```

##
##  Kruskal-Wallis rank sum test
##
## data: tempo by bebida
## Kruskal-Wallis chi-squared = 16.322, df = 2, p-value = 0.0002856

```

17.6.5 Tamanho do efeito

O eta quadrado (η^2), com base na estatística H , pode ser usado como a medida do tamanho do efeito do teste de Kruskal-Wallis. É calculado pela equação:

$$\eta_H^2 = \frac{(H - k + 1)}{(N - k)}$$

onde H é a estatística obtida no teste de Kruskal-Wallis; k é o número de grupos; N é o número total de observações (151).

A estimativa eta ao quadrado assume valores de 0 a 1 e, multiplicada por 100, indica a porcentagem de variância na variável dependente explicada pela variável independente. Pode ser obtido no R com a função `kruskal_effsize()` do pacote `rstatix`:

```

dados %>% kruskal_effsize (tempo~bebida)

## # A tibble: 1 x 5
##   .y.      n effsize method  magnitude
## * <chr> <int>  <dbl> <chr>    <ord>
## 1 tempo     30    0.530 eta2[H] large

```

Um efeito $\geq 0,14$ é considerado grande e $< 0,06$ é pequeno (115).

17.6.6 Testes post hoc

A partir do resultado do teste de Kruskal-Wallis, sabe-se que há uma diferença significativa entre os grupos, mas não se sabe quais pares de grupos são diferentes.

Um teste de Kruskal-Wallis significativo é geralmente seguido pelo teste de Dunn (152) para identificar quais grupos são diferentes.

Para realizar as múltiplas comparações, no R, pode ser usada a função `dunn_test()` com de ajuste de P pelo método de Bonferroni, incluído no pacote `rstatix`:

```

pwc <- dados %>%
  dunn_test (tempo ~ bebida, p.adjust.method = "bonferroni")
pwc

## # A tibble: 3 x 9
##   .y. group1 group2   n1   n2 statistic      p   p.adj p.adj.signif
## * <chr> <chr>  <chr> <int> <int>    <dbl>    <dbl>    <dbl> <chr>
## 1 tempo agua   cafe     10    10    2.10  0.0361  0.108   ns
## 2 tempo agua   alcool   10    10    4.04  0.0000537 0.000161 *** 
## 3 tempo cafe   alcool   10    10    1.94  0.0520  0.156   ns

```

A saída do teste de Dunn, mostra que existe uma diferença estatisticamente significativa apenas entre a água e o álcool.

17.6.7 Conclusão

Um teste de Kruskal-Wallis foi realizado para comparar os tempos de reação em uma simulação de direção após beber água, café ou álcool. Houve evidência de uma diferença ($P = 0,00029$) de pelo menos um par de grupos (Figura 188).

O teste de comparações de pares, usando o teste de Dunn, foi realizado para os três pares de grupos. Houve evidencia de diferença entre o grupo que consumiu duas unidades de álcool e o grupo que ingeriu água (P ajustado (Bonferroni) = 0,00016). Entre os demais pares não houve diferença significativa. O tempo mediano de reação para o grupo que recebeu água foi de 0,84 (0,65 – 0,94) segundos, em comparação com 2,25(1,77 – 2,85) segundos no grupo que bebeu cerveja equivalente a duas unidades de álcool, enquanto para o café foi de 1,45(1,28 – 1,69) segundos.

```

bxp <- ggplot (dados,
               aes (x=bebida,
                     y =tempo,
                     color = bebida)) +
  geom_boxplot (outlier.colour = "black",

```

```

        outlier.size=1.5) +
scale_color_nejm () +
scale_fill_nejm () +
theme_classic () +
theme (legend.position = "none") +
ylab ("Tempo (seg)") +
xlab ("Tipo de Bebida") +
theme (text = element_text (size = 13))

teste <- rstatix::kruskal_test(data = dados, formula = tempo ~ bebida)

pwc <- pwc %>% add_xy_position()

bxp +
  stat_pvalue_manual(pwc, hide.ns = FALSE) +
  labs(subtitle = get_test_label(teste,
                                 detailed = TRUE),
       caption = get_pwc_label(pwc))

```

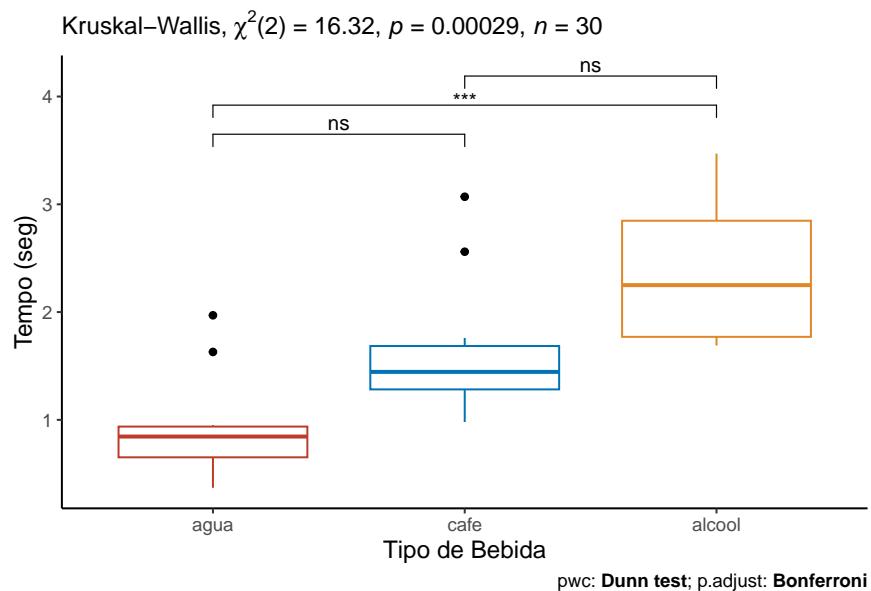


Figura 188: Impacto do tipo de bebida no tempo de reação ao dirigir.

18 Estatística em Epidemiologia

18.1 Pacotes necessários

```
pacman::p_load(readxl,  
                 dplyr,  
                 ggplot2,  
                 gmodels,  
                 DescTools,  
                 kableExtra,  
                 epiR,  
                 epitools,  
                 pROC,  
                 vcd,  
                 MKmisc,  
                 BiocManager,  
                 survival,  
                 survminer,  
                 mlbench,  
                 cowplot,  
                 car,  
                 MASS)
```

18.2 Raciocínio bayesiano no diagnóstico médico

O processo diagnóstico é o centro da atenção da atividade médica na busca de reduzir as incertezas e reconhecer a que classe pertence determinado paciente. Portanto, é extremamente importante saber quanto bem os testes diagnósticos podem prever que um indivíduo é portador de certa condição ou doença. Entende-se aqui como teste diagnóstico todo o processo diagnótico, desde o exame clínico até o mais sofisticado exame de imagem ou laboratorial. A ideia é saber como o teste diagnóstico se comporta para separar um “doente” e um “não doente”; qual a sua validade neste processo?

Deve-se sempre ter em mente que o estabelecimento do diagnóstico é um processo imperfeito que resulta em uma probabilidade ao invés de uma certeza de estar correto. Ou seja, cada vez mais os médicos têm que aplicar as leis da probabilidade na avaliação de testes diagnósticos e sinais clínicos.

A abordagem bayesiana denomina de *probabilidade a priori* a probabilidade estabelecida inicialmente, baseada apenas na experiência do médico, em seu conhecimento em relação a doença suspeitada. Diante de uma evidência de doença, pode ser solicitado um teste diagnóstico. Quando ele recebe um teste positivo para uma doença, a probabilidade muda, passa a ser uma probabilidade condicional, probabilidade da doença dado que o teste é positivo, denominada *probabilidade a posteriori*.

Um teste que define corretamente quem é doente e quem não é doente é denominado de *padrão-ouro* ou *padrão de referência*. Algumas vezes, o teste padrão de referência é simples e barato. Outras vezes, é caro, difícil de obter, tecnicamente complexo, arriscado ou pouco prático. Algumas vezes, não há padrão-ouro. Em função dessas limitações, outros testes são usados e, como consequência, podem ocorrer erros. Em outras palavras, no processo diagnóstico podem ocorrer *falsos positivos* e *falsos negativos*.

Esta incerteza, na utilização de testes diagnósticos, gera a necessidade de o médico conferir a probabilidade de falsos positivos e falsos negativos na elaboração de um diagnóstico ao receber o resultado positivo ou negativo de um exame. Uma maneira simples de mostrar as relações de um teste diagnóstico e o verdadeiro diagnóstico, é mostrada na tabela 2×2 (Figura 189).

		Padrão - ouro	
		Doença presente	
Teste positivo	Doença presente	Verdadeiro positivo (a)	Falso positivo (b)
	Doença ausente	Falso negativo (c)	Verdadeiro negativo (d)
		(a + c)	(b + d)
			(a + b)
			(c + d)

Figura 189: Falsos positivos e falsos negativos

18.2.1 Sensibilidade e Especificidade

As estatísticas mais utilizadas para descrever a validade dos testes de diagnóstico em contextos clínicos são sensibilidade e a especificidade.

Sensibilidade é a habilidade do teste em identificar corretamente quem tem a doença. É a taxa de verdadeiros positivos (VP) de um teste e corresponde a probabilidade de um indivíduo com a doença ter um teste positivo.

Um teste sensível raramente deixará passar pessoas que tenham a doença. Testes com sensibilidade alta são úteis para *excluir* a presença de uma doença. Isto é, um teste negativo exclui virtualmente a possibilidade de o paciente ter a doença de interesse, pois tem pouca probabilidade de produzir resultados falsos negativos. Isto pode ser lembrado pelo mnemônico *SnNout*, do inglês: *High Sensivity, a Negative result rules out the diagnosis* (153).

Especificidade é a habilidade do teste em identificar corretamente quem não tem a doença. É a taxa de verdadeiros negativos (VN) de um teste e corresponde a probabilidade de um indivíduo sem a doença ter um teste negativo. Um teste específico raramente classificará de forma errônea indivíduos sendo portadores da doença quando eles não são. Os testes muito específicos são usados para *confirmar* a presença da doença. Se o teste é altamente específico, um teste positivo sugere fortemente a presença da doença de interesse.

De forma similar que a sensibilidade pode-se usar o mnemônico *SpPin*, do inglês: *High Specificity, a Positive result rules in the diagnosis* (153).

Estas estatísticas de diagnóstico podem ser calculadas a partir das equações, cujas letras representam as caselas da tabela 2×2 , acima;

$$\text{Sensibilidade} = \frac{a}{(a + c)} \quad \text{Especificidade} = \frac{b}{(b + d)}$$

A taxa de falsos negativos (TFN) é a proporção de indivíduos que têm a doença e que têm um resultado de teste negativo e a taxa de falsos positivos (TFP) é a proporção de pacientes que não possuem a doença e que apresentam resultados positivos. Podem ser expressas pelas equações:

$$TFN = \frac{c}{(a + c)} \quad \text{ou} \quad (1 - \text{sensibilidade})$$

$$TFP = \frac{b}{(b + d)} \quad \text{ou} \quad (1 - \text{especificidade})$$

Idealmente, um teste de diagnóstico deveria ter altos níveis de sensibilidade e especificidade. No entanto, isso não é possível, pois existe um balanço entre sensibilidade e especificidade. À medida que a especificidade

aumenta, a sensibilidade diminui e vice-versa. As curvas ROC, que serão discutidas mais adiante neste capítulo, podem ser usadas para identificar um ponto de corte em uma medição contínua que maximize a sensibilidade e a especificidade.

Quando um clínico tem um paciente cujo teste apresentou resultado positivo, a pergunta mais importante é a seguinte: dado que o teste é positivo, qual é a probabilidade de o paciente ter a doença? A sensibilidade do teste não responde a este questionamento, mas sim a probabilidade de um resultado positivo, dado que o paciente tem a doença (154).

18.2.1.1 Exemplo O conjunto de dados `dadosApendicite.xlsx` contém informações de 156 pacientes que realizaram ultrassonografia abdominal para o diagnóstico de apendicite aguda. Para obter arquivo, clique [aqui](#) e salve o mesmo em seu diretório de trabalho.

Foram avaliados pacientes com diagnóstico clínico de apendicite aguda, submetidos à ultrassonografia abdominal e apendicectomia laparoscópica, acompanhado de estudo anatomo-patológico dos apêndices extirpados (155). Será avaliado o teste diagnóstico usado.

Leitura e observação do conjunto de dados

Será usado a função `read_excel()` do pacote `readxl` e a função `glimpse()` do pacote `dplyr`:

```
dados <- read_excel ("dadosApendicite.xlsx")
glimpse(dados)
```

```
## Rows: 156
## Columns: 3
## $ id      <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, ~
## $ apendicite <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
## $ eco      <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
```

As variáveis apendicite e eco foram exibidas como variáveis numéricas e serão transformadas em fatores:

```
dados$apendicite <- factor(dados$apendicite,
                            levels = c(1,2),
                            labels = c("Presente",
                                      "Ausente"))

dados$eco <- factor(dados$eco,
                      levels = c(1,2),
                      labels = c("Positivo",
                                "Negativo"))
```

Construção de uma tabela de contingência 2×2

```
tab_ap <- table (dados$eco,
                  dados$apendicite,
                  dnn = c ("Eco", "Apendicite"))
tab_ap

##          Apendicite
## Eco      Presente Ausente
##   Positivo     85      7
##   Negativo     46     18
```

Cálculo da sensibilidade e da especificidade

Pode-se usar a função `epi.tests()` do pacote `epiR` (156) que calcula, junto com os intervalos de confiança, a prevalência aparente e verdadeira, sensibilidade, especificidade, valores preditivos positivos e negativos e razões de probabilidade positivas e negativas a partir de dados de contagem fornecidos em uma tabela 2×2 . Utiliza os argumentos

- `dat` → dados sob a forma de vetor ou matriz
- `conf.level` → magnitude do intervalo de confiança, entre 0 e 1.

Os resultados serão atribuídos a um objeto de nome `diag`:

```
diag <- epiR::epi.tests(tab_ap,
                         conf.level = 0.95)
print(diag)

##          Outcome +    Outcome -     Total
## Test +      85           7       92
## Test -      46          18       64
## Total       131          25      156
##
## Point estimates and 95% CIs:
## -----
## Apparent prevalence *          0.59 (0.51, 0.67)
## True prevalence *             0.84 (0.77, 0.89)
## Sensitivity *                 0.65 (0.56, 0.73)
## Specificity *                 0.72 (0.51, 0.88)
## Positive predictive value *   0.92 (0.85, 0.97)
## Negative predictive value *   0.28 (0.18, 0.41)
## Positive likelihood ratio     2.32 (1.22, 4.40)
## Negative likelihood ratio    0.49 (0.35, 0.68)
## False T+ proportion for true D- * 0.28 (0.12, 0.49)
## False T- proportion for true D+ * 0.35 (0.27, 0.44)
## False T+ proportion for T+ *   0.08 (0.03, 0.15)
## False T- proportion for T- *   0.72 (0.59, 0.82)
## Correctly classified proportion * 0.66 (0.58, 0.73)
##
## * Exact CIs
```

Assim, a sensibilidade é igual a 65% (IC95%: 56 – 73%) e a especificidade é igual a 72% (IC95%: 51 – 88%). Isto significa que um indivíduo com apendicite aguda tem 65% de probabilidade de ter uma ecografia alterada; um indivíduo sem apendicite aguda tem 72% de probabilidade de ter uma ecografia normal. O objetivo do teste de diagnóstico é usá-lo para fazer um diagnóstico, então há necessidade de saber a probabilidade que o teste fornece para um diagnóstico correto. A sensibilidade e a especificidade não fornecem esta informação. Para atingir esse objetivo, usa-se o *valor preditivo* (157).

18.2.2 Valor Preditivo

O propósito de um teste diagnóstico é usar seus resultados para fazer um diagnóstico, portanto, é necessário conhecer a probabilidade de que o resultado do teste forneça o diagnóstico correto (157).

Os valores preditivos positivo e negativo descrevem a probabilidade de um paciente ter doença, uma vez que os resultados de seus testes são conhecidos.

O **valor preditivo positivo** (VPP) de um teste é definido como a proporção de pessoas com um resultado de teste positivo que realmente têm a doença.

O **valor preditivo negativo** (VPN) é a proporção de pacientes com resultados de teste negativos que não têm doença.

Como a sensibilidade e a especificidade, estas estatísticas de diagnóstico também podem ser calculadas a partir da tabela 2×2 , mostrada no início:

$$VPP = \frac{a}{(a + b)} \quad VPN = \frac{d}{(c + d)}$$

Observando os resultados anteriores da função `epi.tests()`, verifica-se que 92% (85/92) dos indivíduos que tiveram teste positivo (ultrassonografia alterada) tinham doença (apendicite aguda).

Isso significa que seu VPP é igual a 92% (IC95%: 18 – 41%), ou dito de outra forma, uma pessoa com ultrassonografia positiva tem 92% de probabilidade de ter a apendicite aguda. O VPP é também conhecido como *probabilidade pós-teste* de doença dado um teste positivo.

Dos 64 pacientes que tiveram ultrassonografia sem alterações, 18 não apresentaram apendicite aguda, portanto, um VPN de 28% (IC95%: 56 – 73%). Isso significa que uma pessoa quem tem um teste negativo tem 28,1% de probabilidade de não ter apendicite aguda.

Entretanto, essas proporções são de validade limitada. Os valores preditivos de um teste, na prática clínica, dependem criticamente da prevalência da anormalidade nos pacientes testados. No estudo, a prevalência de apendicite aguda é igual a

$$\frac{\text{total de casos de apendicite aguda}}{\text{total de casos no estudo}} = \frac{131}{156} = 0,84 \text{ ou } 84\% \text{ (IC}_{95\%}: 77 \text{ a } 89\%)$$

Levando-se em consideração que a prevalência de apendicite aguda na população é de 7% [(158), mantendo a sensibilidade (64%) e a especificidade (72%) da ultrassonografia, entre 156 pacientes, selecionados aleatoriamente, se esperaria encontrar aproximadamente 11 casos (7% de 156) de apendicite aguda. Para facilitar a compreensão, observe a a tabela 2×2 (Figura 190):

		Anatomopatológico		
		Doença presente	Doença ausente	
Eco positiva	Doença presente	7 (a)	41 (b)	48
	Doença ausente	4 (c)	104 (d)	
		11	145	108

Figura 190: Prevalencia e valor preditivo

O VPP e o VPN são iguais a:

```
a <- 7
b <- 41
c <- 4
d <- 104
vpp = a/(a + b)
round(vpp, 3)*100
```

```
## [1] 14.6
```

```
vpn = d/(c + d)
round(vpn, 3)*100
```

```
## [1] 96.3
```

Ao se comparar o VPP obtido, agora, com o VPP do estudo, observa-se que o mesmo diminuiu bastante, de 92% para 14,6%. O contrário ocorre com a VPN que aumenta substancialmente de 28% para 96,3%, mostrando claramente a influência da prevalência.

Se a prevalência diminui, o VPP diminui e o VPN aumenta. Portanto, será errado aplicar diretamente os valores preditivos publicados de um teste às suas próprias populações, quando a prevalência da doença em sua população for diferente da prevalência da doença na população em que o estudo publicado foi realizado. Um teste pode ser útil em um lugar e não ter validade em outro onde a prevalência é muito baixa.

Pode-se chegar aos mesmos resultados, usando as equações:

$$VPP = \frac{sens \times prev}{(sens \times prev) + [(1 - espec) \times (1 - prev)]}$$

$$VPN = \frac{espec \times (1 - prev)}{[(1 - sens) \times prev] + [espec \times (1 - prev)]}$$

A prevalência pode ser interpretada como a probabilidade antes da realização do teste, conhecida como *probabilidade pré-teste*. A diferença entre as probabilidades pré e pós-teste é uma forma de avaliar a utilidade do teste. Esta diferença pode ser mensurada pela *razão de probabilidade (likelihood ratio)*.

18.2.3 Razão de Probabilidade

A *Razão de Probabilidades (likelihood ratio)* é uma forma alternativa de descrever o desempenho de um teste diagnóstico. Alguns autores a denominam de razão de verossimilhança.

A razão de probabilidades para um resultado de teste é definida como a razão entre a probabilidade de observar aquele resultado em indivíduos com a doença em questão e a probabilidade desse resultado em indivíduos sem a doença (159).

Razões de probabilidade são, clinicamente, mais úteis do que sensibilidade e especificidade. Fornecem um resumo de quantas vezes mais (ou menos) a probabilidade de os indivíduos com a doença apresentarem aquele resultado específico do que os indivíduos sem a doença, e também podem ser usados para calcular a probabilidade de doença para pacientes individuais (160). Cada vez mais as razões de probabilidade estão se tornando populares para relatar a utilidade dos testes de diagnóstico.

Quando os resultados do teste são relatados como sendo positivos ou negativos, dois tipos de razões de probabilidades podem ser descritos, a razão de probabilidades para um teste positivo (denotada LR+) e a razão de probabilidades para um teste negativo (denotada LR-).

A razão de probabilidades para um teste positivo é definida como a probabilidade de um indivíduo com doença ter um teste positivo dividida pela probabilidade de um indivíduo sem doença ter um teste positivo. A fórmula para calcular LR+ é

$$LR(+)=\frac{\text{probabilidade de um indivíduo COM DOENA ter teste+}}{\text{probabilidade de um indivíduo SEM DOENA ter teste+}}$$

Ou,

$$LR(+) = \frac{sensibilidade}{1 - especificidade}$$

Razão de probabilidades positiva maior que 1 significa que um teste positivo tem mais probabilidade de ocorrer em pessoas com a doença do que em pessoas sem a doença. De um modo geral, para os indivíduos que apresentam um resultado positivo, $LR (+) > 10$ aumenta significativamente a probabilidade de doença (“confirma” a doença), enquanto $LR (+) < 0,1$, virtualmente, exclui a probabilidade de uma pessoa ter a doença (161).

Usando os dados da do objeto `diag`, obtido com a função `epi.tests()` do pacote epiR, tem-se que a LR (+) da ultrassonografia para o diagnóstico de apendicite aguda é igual 2.32 (IC95%: 1,22 – 4,40). Significa que uma pessoa com apendicite aguda tem cerca de 2,32 vezes mais probabilidade de ter um teste positivo do que uma pessoa que não tem a doença.

A razão de probabilidade negativa é definida como a probabilidade de um indivíduo com doença ter um teste negativo dividido pela probabilidade de um indivíduo sem doença ter um teste negativo. A fórmula para calcular a LR– é:

$$LR(-) = \frac{probabilidade\ de\ um\ individuo\ COM\ DOENA\ ter\ teste-}{probabilidade\ de\ um\ individuo\ SEM\ DOENA\ ter\ teste-}$$

Ou,

$$LR(+) = \frac{1 - sensibilidade}{especificidade}$$

Razão de probabilidade negativa menor que 1 significa que um teste negativo é menos provável de ocorrer em pessoas com a doença do que em pessoas sem a doença. Um LR muito baixo (abaixo de 0,1) praticamente exclui a chance de que uma pessoa tenha a doença (161).

Voltando aos dados anteriores, a LR (-) para a ultrassonografia é igual a 0.49 (IC95%: 0.35 - 0.68). Significa que a probabilidade de ter um teste negativo para indivíduos com doença A é 0,49 vezes ou cerca de metade daqueles sem a doença. Dito de outra forma, os indivíduos sem a doença têm cerca o dobro probabilidade de ter um teste negativo do que os indivíduos com a doença.

18.2.3.1 Estimando a probabilidade de doença Uma grande vantagem das razões de probabilidade é que elas podem ser usadas para ajudar o médico a adaptar a sensibilidade e a especificidade dos testes aos pacientes individuais. Ao se atender um paciente em uma clínica, pode-se decidir realizar um teste específico, após uma anamnese e um exame físico. A decisão de fazer o teste baseia-se nos sintomas e sinais do paciente e na experiência pessoal. Existe suspeita de um determinado diagnóstico e o objetivo é excluir ou confirmar esse diagnóstico. Antes de solicitar o teste, geralmente existe uma estimativa aproximada da probabilidade do paciente de ter essa doença, conhecida como probabilidade pré-teste ou *a priori*, que geralmente é estimada com base na experiência pessoal do médico, dados de prevalência local e publicações científicas.

A razão mais importante pela qual um teste é realizado é tentar modificar a probabilidade de doença. Um teste positivo pode aumentar a probabilidade pré-teste e um teste negativo pode reduzir a probabilidade pré-teste. A probabilidade pós-teste de doença é o que mais interessa aos médicos e pacientes, pois isso pode ajudar a decidir se devem confirmar, descartar um diagnóstico ou realizar outros testes.

Os resultados dos testes clínicos são geralmente usados não para fazer ou excluir categoricamente um diagnóstico, mas para modificar a probabilidade do pré-teste a fim de gerar a probabilidade do pós-teste. O teorema de Bayes é uma relação matemática que permite estimar a probabilidade pós-teste.

Para se compreender este conceito, é importante entender a diferença entre probabilidade e odds (162).

Probabilidade é a proporção de pessoas que apresentam uma determinada característica (teste positivo, sinal clínico). *Odds* (chance) representa a razão entre duas características complementares, ou seja, a probabilidade de um evento dividido pela probabilidade de não evento ($1 - \text{evento}$). Ambos contêm as mesmas informações de maneiras diferentes. Por exemplo, usando os dados da tabela `tab_ap`

```
addmargins(tab_ap)
```

```
##               Apendicite
## Eco          Presente Ausente Sum
##   Positivo      85       7   92
##   Negativo      46      18   64
##   Sum           131      25  156
```

verifica-se que a probabilidade (p) de ultrassonografia positiva é de

```
p = 92/156
p
```

```
## [1] 0.5897436
```

e que o *odds* da ultrassonografia positiva²⁰ é

```
odds = (92/156)/(64/156)
odds
```

```
## [1] 1.4375
```

Pode-se transformar a *odds* em probabilidade e vice-versa da seguinte maneira:

$$p = \frac{\text{odds}}{1 + \text{odds}}$$

```
p = 1.44/(1 + 1.44)
round(p, 2)
```

```
## [1] 0.59
```

$$\text{odds} = \frac{p}{1 - p}$$

```
odds = 0.59/(1-0.59)
round(odds, 2)
```

```
## [1] 1.44
```

²⁰Existem duas maneiras de descrever uma estimativa de odds: ou como um número isolado, por exemplo, 0,25, subentendendo que expressa uma razão, 0,25: 1,0, ou de forma clara como uma razão 1:4. Ou seja, para cada indivíduo com o fator existem quatro sem o fator. Tradicional e comumente usados no mundo das apostas em corridas de cavalos.

Pelo teorema de Bayes, sabendo-se a *probabilidade a priori* ou *probabilidade pré-teste*, é possível obter a *probabilidade pós-teste* ou *a posteriori*, usando a razão de probabilidades.

Para atingir este objetivo, basta multiplicar o *odds pré-teste* pela razão de probabilidades:

$$odds_{pos-teste} = odds_{pre-teste} \times LR$$

Para encontrar a probabilidade pós-teste, basta converter o odds pós-teste em probabilidade.

Voltando da `tab_ap`, foi verificado que o LR (+) é igual a 2,32 e a prevalência de apendicite aguda é em torno de 7% pode-se prever a probabilidade de haver apendicite aguda, diante de uma ultrassonografia alterada:

```
prev <- 0.07
LR <- 2.32

odds_pre <- 0.07/(1 -0.07)

odds_pos <- odds_pre * LR

p_pos <- odds_pos/(odds_pos +1)

round(p_pos, 2)

## [1] 0.15
```

Ou, em outras palavras, diante de um teste positivo, a probabilidade de o paciente ter apendicite aguda passa de 7% antes do teste para praticamente 15%!

Estes cálculos podem ser simplificados, utilizando o nomograma de Fagan (163), extremamente fácil de se usar (164), pois basta unir a probabilidade pré-teste ao LR que a reta apontará para a probabilidade pós-teste (Figura 191).

18.2.4 Curva ROC

Nem sempre o resultado de um teste é dicotômico (positivo/negativo). Com frequência, trabalha-se com variáveis contínuas (pressão arterial, glicemia, dosagem do sódio, dosagens hormonais, etc.). Neste caso, não há um resultado “positivo” ou “negativo”. Um “ponto de corte” precisa ser criado, para definir quem será considerado positivo ou negativo.

A escolha do ponto de corte depende das consequências de um resultado falso positivo ou de um falso negativo. Falsos positivos estão associados com custos (emocional ou financeiro) e com a dificuldade de “desrotular” alguém que recebeu o rótulo de “positivo”. Resultados falsos negativos podem “tranquilizar” pessoas doentes que não são seguidas ou tratadas precocemente.

A distribuição dos níveis glicêmicos em diabéticos e não diabéticos não tem um ponto de corte bem nítido. As duas populações se sobrepõem (Figura 192), gerando falso positivos ou falso negativos, dependendo do ponto de corte escolhido (162).

Suponha que ao se examinar uma população fosse escolhido o ponto de corte de 80mg/dL, haveria um aumento no número de indivíduos com teste positivo com uma taxa de falsos positivos elevada, diminuindo a especificidade do teste. Se, por outro lado, o ponto de corte fosse elevado para 200mg/dL, o número de falsos negativos teria um grande aumento, reduzindo a sensibilidade. Esta oscilação entre a sensibilidade e a especificidade ocorre pelo fato de a localização do ponto de corte ser uma decisão arbitrária num contínuo entre o normal e anormal.

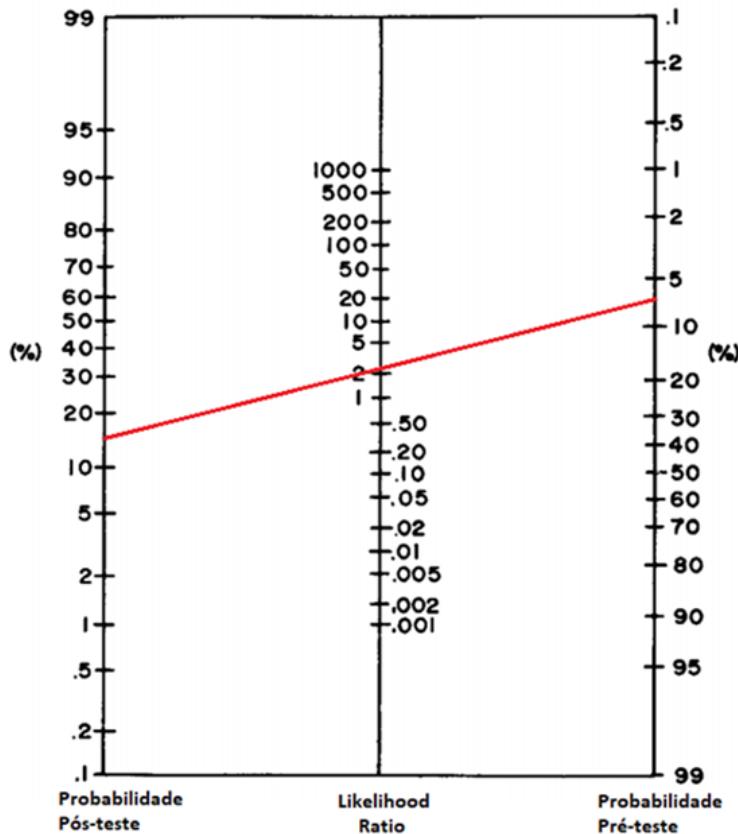


Figura 191: Nomograma de Fagan

Ao se escolher um ponto de corte deve-se fazer um balanço entre a sensibilidade e a especificidade, levando em conta as consequências da escolha. Por exemplo, a triagem para fenilcetonúria em recém-nascidos valoriza a sensibilidade em vez de especificidade; o custo da perda de um caso é alto, pois existe tratamento eficaz. Uma desvantagem é que ocorre um grande número de testes falso positivos que causam angústia e a realização de mais testes.

Em contraste, a triagem para o câncer de mama deve favorecer a especificidade sobre a sensibilidade, uma vez que uma avaliação mais aprofundada daquelas com teste positivo, implica em biopsias dispendiosas e invasivas.

As curvas ROC (*Receiver Operating Characteristic*) são uma ferramenta inestimável para encontrar o ponto de corte em uma medida com distribuição contínua que melhor prediz se uma condição está presente, por exemplo, se pacientes são positivos ou negativos para a presença de uma doença (165). As curvas ROC são usadas para encontrar um ponto de corte que separa um resultado de teste “normal” de um “anormal” quando o resultado do teste é uma medida contínua. As curvas ROC são traçadas calculando a sensibilidade e a especificidade do teste na predição do diagnóstico para cada valor da medida. A curva permite determinar um ponto de corte para a medição que maximiza a taxa de verdadeiros positivos (sensibilidade) e minimiza a taxa de falsos positivos ($1 -$ especificidade) e, portanto, maximiza a razão de probabilidades (*likelihood ratio*).

18.2.4.1 Exemplo O conjunto de dados `dadosTestes.xlsx` contém informações para os resultados hipotéticos de três testes bioquímicos diferentes e uma variável (`doença`) que indica se foi confirmada a doença (*padrão-ouro*). Para obter arquivo, clique [aqui](#) e salve o mesmo em seu diretório de trabalho.

Leitura e observação dos dados

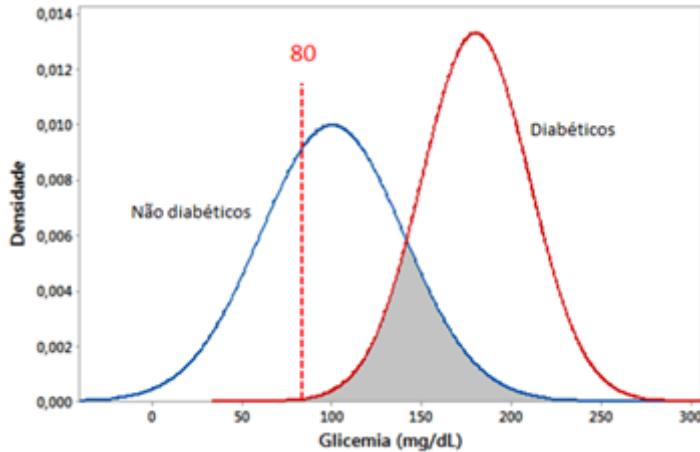


Figura 192: Populações de indivíduos normais e diabéticos

```

dados <- read_excel("dadosTestes.xlsx")
dados$doenca <- as.factor(dados$doenca)

```

As curvas ROC são usadas para avaliar qual teste é mais útil para prever quais pacientes serão positivos para a doença. A hipótese nula é que a área sob a curva ROC é igual a 0,5, ou seja, a habilidade do teste para identificar casos positivos e negativos é a esperada por acaso.

As Figuras 193, 194 e 195 mostram a quantidade de sobreposição na distribuição da medição dos testes bioquímicos contínuos em ambos os grupos doença positiva e doença negativa. No Teste 1, a sobreposição é completa e não haverá um ponto de corte que separe efetivamente os dois grupos. Nos Testes 2 e 3, há uma maior separação das medidas de teste entre os grupos, particularmente para Teste 3.

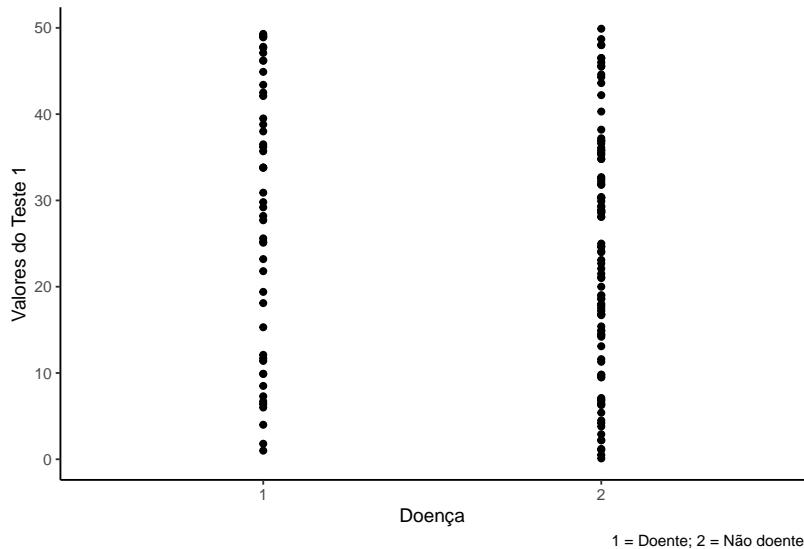


Figura 193: Gráfico de dispersão: Teste 1 vs doença.

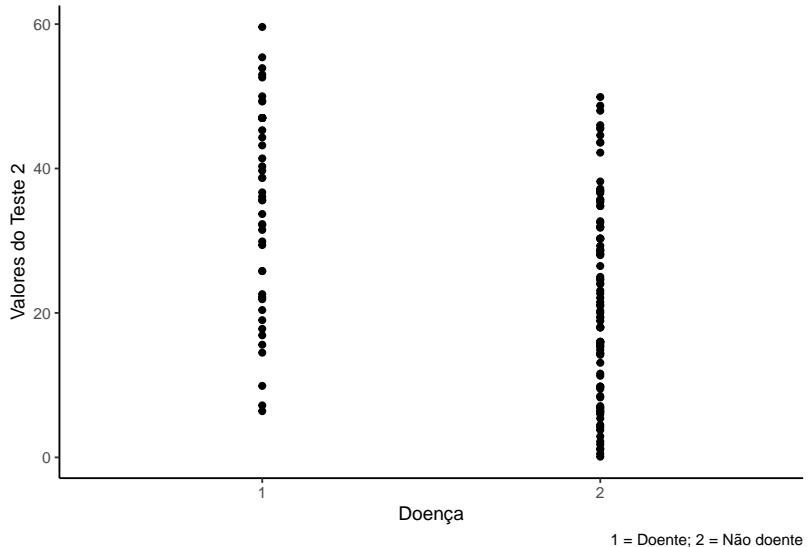


Figura 194: Gráfico de dispersão: Teste 2 vs doença.

18.2.4.2 Validade dos testes A validade dos testes, na distinção entre os grupos doença-positivo e doença-negativo, pode ser quantificada pelas curvas ROC, usando a função `roc` o pacote `pROC` (166). Este pacote tem várias funções:

- **auc**: calcula a área da curva ROC;
- **ci**: calcula o intervalo de confiança da curva ROC;
- **ci.auc**: calcula o intervalo de confiança da AUC;
- **ci.se**: calcula o intervalo de confiança de sensibilidades em determinadas especificidades;
- **ci.sp**: calcula o intervalo de confiança de especificidades em determinadas sensibilidades;
- **ci.thresholds**: calcula o intervalo de confiança dos limites;
- **coords**: Retorna as coordenadas (sensibilidades, especificidades, pontos de corte) de uma curva ROC;
- **roc**: Constrói uma curva ROC;
- **roc.test**: Compara a AUC de duas curvas ROC correlacionadas;
- **smooth**: suaviza a curva ROC

Construção das curvas ROC

Usando mencionada com os argumentos variável resposta (`doenca`), variável preditora (`teste3`, `teste2` e `teste1`), indicação de que o gráfico deve ser desenhado (`plot = TRUE`). Como por padrão o gráfico é plotado com a sensibilidade no eixo x e a especificidade no eixo y , deve-se acrescentar o argumento `legacy.axes = TRUE` para aparecer o seu complemento, os falsos positivos ($1 - \text{especificidade}$).

Além desse, pode-se usar vários outros argumentos como: `print.auc = TRUE`, que imprime no gráfico a AUC e `ci` que imprime junto o intervalo de confiança da AUC. Para que a sensibilidade e especificidade apareça como uma percentagem, deve-se usar o argumento `percent = TRUE`, pois o padrão é `FALSE`. Os demais argumentos são os rótulos dos eixos, cor da curva, largura da curva (`lwd`).

Estes comandos geram a Figura 196, que exibe o desempenho diagnóstico dos três testes.

Interpretação do resultado

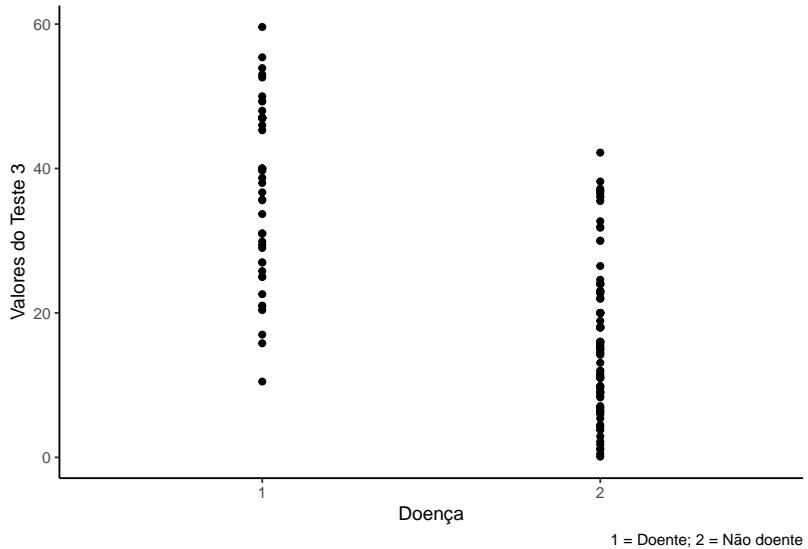


Figura 195: Gráfico de dispersão: Teste 3 vs doença.

Em uma curva ROC, a sensibilidade é calculada usando cada valor do teste no conjunto de dados como um ponto de corte e é plotada em relação à $(1 - \text{especificidade})$ correspondente nesse ponto, como mostrado na Figura 196.

Assim, a curva são os Verdadeiros Positivos (VP) plotados em relação aos Falsos Positivos (FP), calculados usando cada valor do teste como ponto de corte. A reta diagonal indica onde o teste cairia se os resultados não fossem melhores do que o acaso para predizer a presença de uma doença. O Teste 1 está próximo desta reta, confirmando que ele tem pouca capacidade de discriminar os pacientes doentes e não doentes.

A área abaixo da reta diagonal é equivalente a 0,5 da área total. Quanto maior a área sob a curva ROC, mais útil é o teste para predizer os pacientes que têm a doença. Uma curva que cai substancialmente abaixo da linha diagonal indica que o teste tem pouca capacidade de diagnosticar a doença. Quando há uma separação perfeita dos valores dos dois grupos, isto é, sem sobreposição das distribuições, a área sob a curva ROC é igual a 1 (a curva ROC alcançará o canto superior esquerdo do gráfico).

A área sob a curva (*Area Under the Curve – AUC*) e seu intervalo de confiança de 95% podem ser obtidos com os comandos usados na construção da Figura 196 ou separadamente usando as funções `auc()` e `ci.auc()` do pacote `pROC`.

```
auc (roc1)
```

```
## Area under the curve: 0.5891
```

```
ci.auc (roc1)
```

```
## 95% CI: 0.4856-0.681 (2000 stratified bootstrap replicates)
```

```
auc(roc2)
```

```
## Area under the curve: 0.7616
```

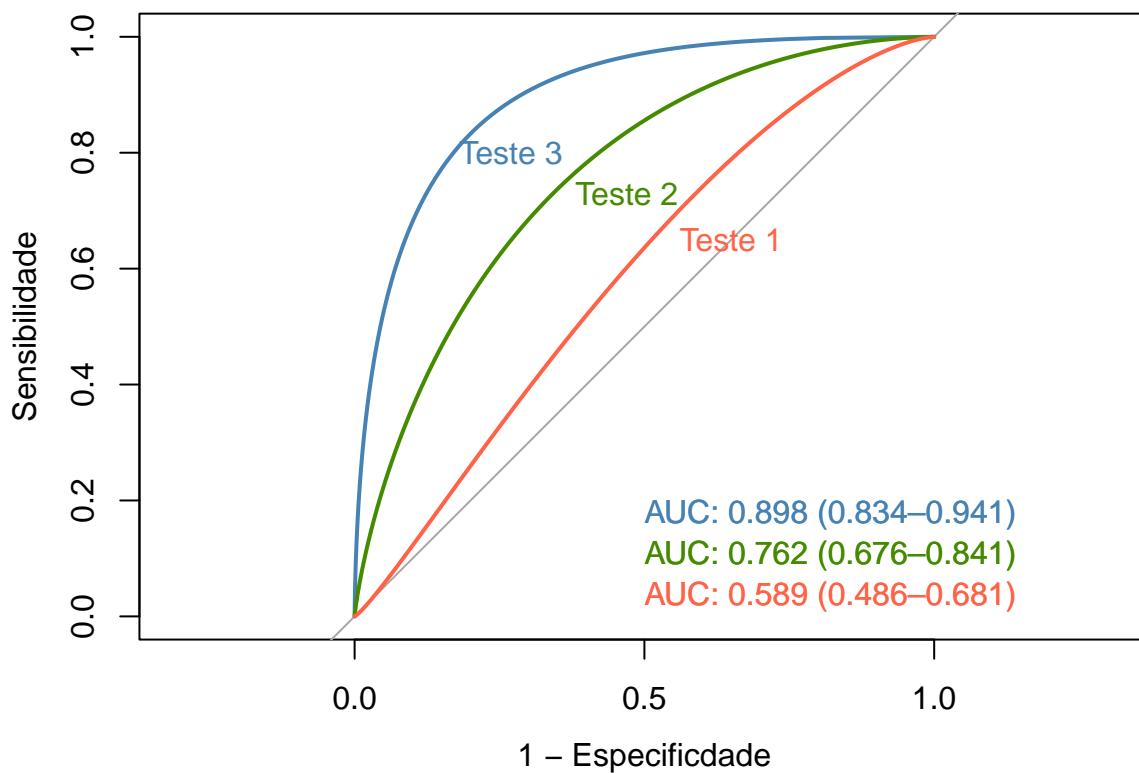


Figura 196: Curvas ROC para os Testes 1, 2 e 3.

```

ci.auc(roc2)

## 95% CI: 0.6743–0.8385 (2000 stratified bootstrap replicates)

auc (roc3)

## Area under the curve: 0.898

ci.auc(roc3)

## 95% CI: 0.8337–0.9409 (2000 stratified bootstrap replicates)

```

A acurácia geral de um teste pode ser descrita como a área sob a curva; quanto maior for a área, melhor será o teste. Na Figura 196, o Teste 3 tem uma AUC maior que os outros dois testes. Usa-se a seguinte estimativa (Tabela 23) para avaliar a acurácia de um teste ou da capacidade de identificar corretamente uma condição usando curva ROC (167):

Desta forma, o Teste 3 pode ser considerado um bom teste e o Teste 1 é um teste ruim.

Comparando duas curvas

Pode-se comparar duas curvas ROC com a função `roc.test()`, por exemplo, comparando as curvas dos Teste 3 e 2 (168):

Tabela 23: Acurácia do teste diagnóstico

AUC	Qualidade do Teste
>0,90	excelente
0,80 a 0,90	muito bom
0,70 a 0,80	bom
0,60 a 0,70	suficiente
0,50 a 0,60	ruim
<0,50	ignorar teste

```
roc.test(roc3, roc2)
```

```
##
##  Bootstrap test for two correlated ROC curves
##
## data:  roc3 and roc2
## D = 4.6643, boot.n = 2000, boot.stratified = 1, p-value = 3.097e-06
## alternative hypothesis: true difference in AUC is not equal to 0
## sample estimates:
## Smoothed AUC of roc1 Smoothed AUC of roc2
##           0.8980454          0.7616201
```

O Teste 3 tem uma AUC que o caracteriza como um bom teste e o teste de DeLong, entregue na saída do `roc.test()`, resultou que a diferença entre ele o Teste 2 é estatisticamente significativa ($P < 0,0001$).

18.2.4.3 Melhor ponto de corte O melhor ponto de corte (*Best Critical Value*), que às vezes é chamado de *ponto de diagnóstico ótimo* ou *de Youden*, é o ponto da curva mais próximo da parte superior do eixo y (Figura 196, Teste 3). Este é o ponto em que a taxa de verdadeiros positivos é otimizada e a de taxa de falsos positivos é minimizada. O melhor ponto de corte para o Teste 3 é mostrado na Figura 197. Este melhor ponto de corte pode ser identificado a partir dos pontos de coordenadas da curva, usando a função `roc()` com os seguintes argumentos:

```
##
## Call:
## roc.default(response = dados$doenca, predictor = dados$teste3,      ci = TRUE, plot = TRUE, threshold)
##
## Data: dados$teste3 in 48 controls (dados$doenca 1) > 97 cases (dados$doenca 2).
## Area under the curve: 0.8973
## 95% CI: 0.8444-0.9502 (DeLong)
```

Assim, para o *Teste 3*, o ponto de corte ideal é 24,8, onde a especificidade é igual a 0,854 e a sensibilidade é igual 0,845. Estes dados, fornecem um LR para um resultado positivo igual a:

$$LR (+) = \frac{0.845}{(1 - 0.854)} = 5,79$$

As coordenadas da curva ROC pode ser obtida com a seguinte programação, a partir de uma sensibilidade e especificidade acima de 0 (zero):

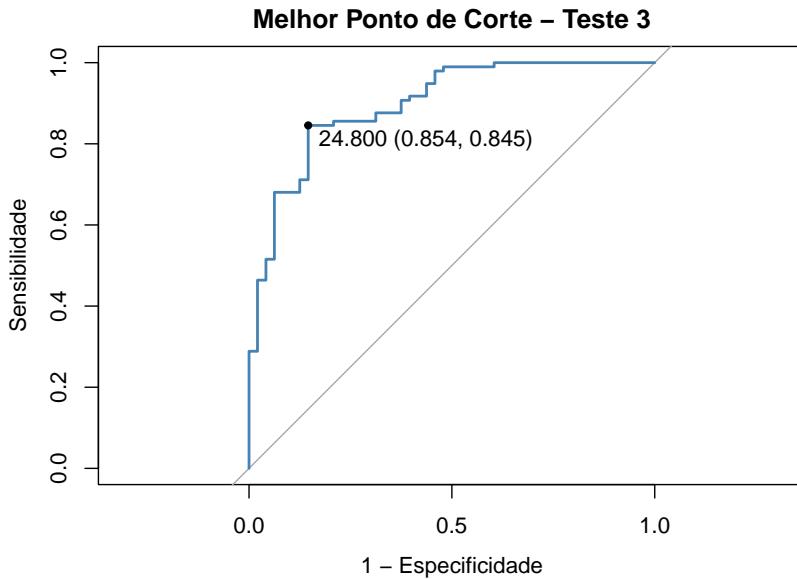


Figura 197: Curvas ROC para os Testes 1, 2 e 3.

```
coordenadas <- dados %>% roc(doenca, teste3) %>% coords (transpose = F)
head(coordenadas, 10)
```

```
##      threshold specificity sensitivity
## 1        Inf    0.000000000      1
## 2       57.50   0.020833333      1
## 3       54.65   0.041666667      1
## 4       53.45   0.062500000      1
## 5       52.80   0.083333333      1
## 6       51.30   0.104166667      1
## 7       49.65   0.125000000      1
## 8       48.65   0.166666667      1
## 9       47.50   0.187500000      1
## 10      46.50   0.354166667      1
```

A estatística J de Youden (169) é calculada deduzindo 1 a partir da soma de sensibilidade e especificidade do teste e não é expressa como porcentagem, mas como parte de um número inteiro: $(\text{sensibilidade} + \text{especificidade}) - 1$. A estatística J de Youden no melhor ponto de corte do Teste 3 é igual a $(0,845 + 0,854) - 1 = 0,699$.

Este é o maior valor de todos os valores das coordenadas (91 valores) usadas (ponto de corte 39 na Saída 15.4).

```
youden <- max(coordenadas$sensitivity + coordenadas$specificity) - 1
youden
```

```
## [1] 0.6995275
```

A Figura 197 mostra o ponto de corte ideal. Ele também pode ser obtido com a função `coords()` do pacote `pRoc`

```

roc3 <- dados %>% roc(doenca, teste3)
coords(roc3, x = "best", ret="threshold", transpose = FALSE,
       best.method="youden")

```

```

##   threshold
## 1      24.8

```

O método para obter o melhor ponto de corte (`best.method`) pode ser pelo método `e youden` ou `closest.topleft`. No exemplo, o resultado é o mesmo. Para maiores detalhes consulte a ajuda da função (`?coord`).

18.3 Estatística *kappa*

A estatística de concordância *kappa* (k) de Cohen é utilizada para descrever a concordância entre dois ou mais avaliadores quando realizam uma avaliação nominal ou ordinal de uma mesma amostra (170). A estatística *kappa* corrige a chance do acaso nas avaliações e é obtida pela fórmula igual a:

$$k = \frac{p_o - p_e}{1 - p_e}$$

Onde p_o = proporção observada de concordância e p_e = proporção esperada de concordância apenas pelo acaso.

Por exemplo, dois radiologistas podem revisar independentemente uma série de radiografias do tórax de pacientes para determinar a presença ou ausência de pneumonia. Para avaliar o grau de concordância entre as classificações dos dois médicos, pode ser relatado o percentual de concordância entre os avaliadores (por exemplo, 50% dos avaliadores responderam “sim” nas duas ocasiões). No entanto, esse percentual pode ser enganoso, pois não leva em conta o nível de concordância entre os dois avaliadores que pode ocorrer por acaso. A estatística *kappa* pode ser usada para avaliar a concordância das respostas para dois ou mais avaliadores após considerar a concordância casual. Portanto, a estatística *kappa* é uma estimativa da proporção de concordância entre avaliadores que excede a concordância que ocorreria por acaso.

A interpretação dos valores de *kappa* é mostrada na Tabela 24 (171). Quando a proporção observada de concordância é menor que a esperada por acaso, o *kappa* terá um valor negativo indicando não concordância. Um valor de *kappa* igual a 0 indica que a concordância observada é igual à concordância casual.

O teste de hipóteses (calcula o valor P) testa a hipótese de que a concordância entre os dois avaliadores seja puramente aleatória. Quando o valor P é menor que 0,05, rejeitamos a hipótese de que a concordância foi puramente aleatória. As premissas para o *kappa* de Cohen são que os participantes ou itens a serem classificados são independentes e também que os avaliadores e categorias são independentes.

Tabela 24: Valor Kappa e nível de concordância correspondente

Valor kappa	Concordância
<0,00	pobre
0,00 - 0,20	leve
0,21 - 0,40	razoável
0,41 - 0,60	moderada
0,61 - 0,80	substancial
0,81 - 1,00	quase perfeita

Existem diferentes tipos de estatísticas *kappa*. Para dados com três ou mais categorias possíveis (por exemplo, concordo, concordo parcialmente, discordo) ou para dados categóricos ordenados, o *kappa* ponderado deve

ser usado para que as respostas que estão mais distantes da concordância tenham maior peso do que aquelas próximas à concordância. No exemplo usado, as categorias possíveis são dicotômicas (sim e não), portanto, o kappa não ponderado (*unweighted*) e o ponderado (*weighted*) retornam o mesmo resultado.

18.3.1 Exemplo

O arquivo `dadosPneumonia.xlsx` contém os dados de 54 crianças com suspeita de pneumonia, cujas radiografias foram avaliadas por dois radiologistas. O objetivo foi medir a concordância diagnóstica dos dois profissionais. Para o cálculo do coeficiente *kappa* será usada a função `Kappa()` do pacote `vcd` (172). Usa os seguintes argumentos:

- *x* → matriz ou tabela
- *weights* → matriz especificada pelo usuário com as mesmas dimensões de *x*, desnecessário para *kappa* não ponderado.

Na impressão do *kappa* pode-se usar `print(k, digits = 3, CI = TRUE, level = 0.95)`. Onde *k* é o coeficiente de *kappa*, calculado pela função `Kappa()`, *CI* é o intervalo de confiança e o nível de confiança padrão é 95%.

18.3.1.1 Baixar e carregar o conjunto de dados O conjunto de dados `dadosPneumonia.xlsx` pode ser obtido [aqui](#). Após salvar o arquivo em seu diretório, ele pode ser carregado com a função `read_excel()` do pacote `readxl`:

```
dados <- readxl::read_excel("dadosPneumonia.xlsx")
```

18.3.1.2 Construção da tabela necessária para o cálculo Cruzar os resultados da opinião dos radiologistas, variáveis `rx1` e `rx2`:

```
dados$rx1 <- factor(dados$rx1,
                      ordered=TRUE,
                      levels = c("sim", "não"))
dados$rx2 <- factor(dados$rx2,
                      ordered=TRUE,
                      levels = c("sim", "não"))

tabk <- table (dados$rx1,
                dados$rx2,
                dnn = c ("Radiologista 1", "Radiologista 2"))
tabk

##           Radiologista 2
## Radiologista 1 sim não
##       sim   32   5
##       não    3  14
```

18.3.1.3 Cálculo do *kappa* O *kappa* é dado por:

```
k <- vcd::Kappa(tabk)
print (k,
       digits= 3,
       CI=TRUE,
       level=0.95)
```

```

##           value    ASE     z Pr(>|z|) lower upper
## Unweighted 0.667 0.107 6.21 5.42e-10 0.456 0.878
## Weighted   0.667 0.107 6.21 5.42e-10 0.456 0.878

```

A saída exibe o *kappa* pontual os intervalos de confiança de 95%, podendo-se concluir desses resultados que existe uma boa confiabilidade nos resultados dos radiologistas (substancial, de acordo com a Tabela 24).

18.4 Medidas de frequênciа

18.4.1 Prevalência

A prevalência, ou mais adequadamente, a *prevalência pontual* de uma doença é a proporção da população portadora da doença em um determinado ponto do tempo. É uma medida instantânea por excelência e fornece uma medida estática da frequência da doença. É também conhecida como *tакса de prevalência* e é expressa em percentagem ou por 10^n habitantes. As medidas de prevalência geram informações úteis para o planejamento e administração de serviços de saúde.

A *prevalência por período* descreve os casos que estavam presentes em qualquer momento durante um determinado período de tempo. Descreve o número total de casos de uma doença que se sabe haver existido durante um período de tempo.

Um tipo especial de prevalência de período é a prevalência ao longo da vida, que mede a frequência cumulativa ao longo da vida de um resultado até o momento presente (ou seja, a proporção de pessoas que tiveram o evento em qualquer momento no passado).

As doenças, quanto a sua duração, podem ser agudas e de longa duração ou crônicas. A prevalência é proporcional ao tempo de duração da doença. Hipoteticamente, se o surgimento de novos casos de doença ocorre em ritmo constante e igual para doenças agudas e crônicas, estas últimas acumularão casos, aumentando a prevalência. As doenças agudas tenderão a manter uma prevalência constante. A terapêutica, diminuindo o tempo de duração das doenças, também reduz a prevalência. A prevalência é dada pela razão:

$$\text{Prevalencia} = \frac{\text{Número de casos conhecidos da doença}}{\text{Total da População}} \times 10^n$$

18.4.1.1 Exemplo Como exemplo, será verificada a frequência de tabagismo entre as puérperas da maternidade do HGCS. O banco de dados `dadosMater.xlsx` contém informação de 1368 nascimentos e pode ser obtido [aqui](#). Depois de salvo em seu diretório de trabalho, ele pode ser carregado com a função `read_excel()` do pacote `readxl`.

```
dados <- readxl::read_excel ("dadosMater.xlsx")
```

Inicialmente, verifica-se quantas fumantes existem. O conjunto de dados contém uma variável `fumo`, onde 1 = fumante e 2 = não fumante. Portanto, há necessidade de transformar a variável numérica em um fator:

```

dados$fumo <- factor (dados$fumo,
                      ordered = TRUE,
                      levels = c(1,2),
                      labels = c("fumante", "não fumante"))

tabFumo <- table(dados$fumo)
tabFumo

```

```

##          fumante não fumante
##            301           1067

```

Além de relatar a estimativa pontual da frequência da doença, é importante fornecer uma indicação da incerteza em torno dessa estimativa pontual. A função `epi.conf()`, do pacote `epiR` (156), permite calcular intervalos de confiança para prevalência, motivo da escolha dessa função.

A função `epi.conf()` usa os seguintes argumentos:

- `dat` → matriz ou tabela;
- `ctype` → tipo de intervalo de confiança a ser calculado. Opções: `mean.single`, `mean.unpair`, `mean.pair`, `prop.single`, `prop.unpaired`, `prevalence`, `inc.risk`, `inc.rate`, `odds` e `smr` (*standardized mortality rate*);
- `method` → método a ser usado. Quando `ctype = "inc.risk"` ou `ctype = "prevalence"`, as opções `exact`, `wilson` e `fleiss`. Quando `ctype = "inc.rate"` as opções são `exact` e `byar`;
- `N` → tamanho da população;
- `conf.level` → magnitude do intervalo de confiança retornado. Deve ser um único número entre 0 e 1.

Construção da matriz

Com os dados da `tabFumo`, constroi-se uma matriz de duas colunas:

```

n1 <- 301
N1 <- 301 + 1067
mat1 <- as.matrix(cbind (n1, N1))
mat1

```

```

##      n1    N1
## [1,] 301 1368

```

Cálculo da prevalência

usando a função `epiR()`, tem-se:

```

epiR::epi.conf(mat1,
                 ctype = "prevalence",
                 method = "exact",
                 conf.level = 0.95)

##          est      lower     upper
## 1 0.2200292 0.1983313 0.2429365

```

A Saída mostra que a prevalência de fumantes entre as puérperas do HGCS é igual a 22,0% (IC95%: 19,8 – 24,3%).

18.4.2 Incidência

A incidência fornece uma medida da frequência com que os indivíduos suscetíveis se tornam casos de doenças, à medida que são observados ao longo do tempo.

Um caso de incidente ocorre quando um indivíduo deixa de ser suscetível e passa a ser doente. A contagem de casos de incidentes é o número de tais eventos que ocorrem em uma população durante um período de acompanhamento definido. Existem duas maneiras de expressar a incidência:

A *incidência cumulativa* (risco) é a proporção de indivíduos inicialmente suscetíveis em uma população que se tornam novos casos durante um período de acompanhamento definido.

Para calcular a incidência cumulativa, é necessário primeiro identificar os doentes e após acompanhar por um determinado tempo os não doentes (Figura 198).



Figura 198: Incidência

A *taxa de incidência* (densidade de incidência ou taxa de incidência) é o número de novos casos da doença que ocorrem por unidade de tempo em risco durante um período de acompanhamento definido. Este período é expresso como *pessoas-tempo* (pessoas-ano, por exemplo).

O conceito de pessoas-tempo pode ser ilustrado com o seguinte exemplo: a Figura 199 representa um estudo epidemiológico hipotético com duração de cinco anos, onde D é o desfecho e C representa os sujeitos que deixaram o estudo por migração ou morte (censurados) por causa não relacionada ao desfecho

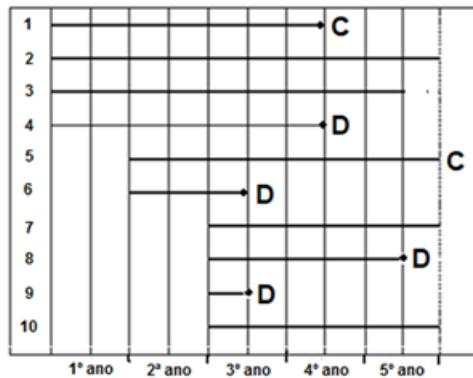


Figura 199: Estudo epidemiológico hipotético.

Nesse estudo hipotético, o indivíduo 1 permaneceu no estudo 3,5 anos; o indivíduo 2,5 anos; o indivíduo 3, 4,5 anos e, assim por diante, totalizando 32,5 pessoas-anos. Em outras palavras, ocorreram 4 desfechos durante os 5 anos do estudo, consequentemente, a taxa de incidência (TI) foi de

$$TI = \frac{4}{32,5} \times 1000 = \frac{123}{1000 \text{ pessoas} - \text{ano}}$$

Isto significa que se fossem acompanhadas 1000 pessoas por um ano, 123 delas apresentariam o desfecho D.

18.4.2.1 Exemplo Aparentemente, pessoas cegas tem uma menor incidência de câncer e esse efeito parece ser mais pronunciado em pessoas totalmente cegas do que em pessoas com deficiência visual grave.

Para testar essa hipótese, foi identificada uma coorte de 1.567 pessoas totalmente cegas e 13.292 sujeitos com deficiência visual grave. As informações sobre a incidência de câncer foram obtidas do Registro Sueco de Câncer (173). Foram diagnosticados de 136 casos de câncer em 22050 pessoas-ano em risco totalmente cegas e 1709 casos de câncer em 127650 pessoas-anos em risco com deficiência visual grave.

A taxa de incidência pode ser calculada, usando-se a mesma função `epi.conf()`, usada para o cálculo da prevalência, mudando o argumento `ctype = "prevalence"` para `ctype = "inc.rate"`, conforme recomendado:

Pessoas totalmente cegas

Inicialmente, controi-se a matriz:

```
n2 <- 136
N2 <- 22050
mat2 <- as.matrix(cbind (n2, N2))
mat2
```

```
##      n2      N2
## [1,] 136 22050
```

Logo, a incidência de câncer nos totalmente cegos é:

```
epiR::epi.conf(mat2,
  ctype = "inc.rate",
  method = "exact",
  conf.level = 0.95)*1000
```

```
##      est      lower      upper
## n2 6.1678 5.174806 7.295817
```

Pessoas com grave deficiência visual

Inicialmente, controi-se a matriz:

```
n3 <- 1709
N3 <- 127650
mat3 <- as.matrix(cbind (n3, N3))
mat3
```

```
##      n3      N3
## [1,] 1709 127650
```

Logo, a incidência de câncer nos com grave deficiência visual é:

```
epiR::epi.conf(mat3,
  ctype = "inc.rate",
  method = "exact",
  conf.level = 0.95)*1000
```

```
##      est      lower      upper
## n3 13.38817 12.76088 14.03832
```

As Saídas mostram que para cada 1000 pessoas cegas (a função foi multiplicada por 1000) acompanhadas por um ano, ocorreu 6,2 ((IC95%: 5,2 – 7,3) casos de câncer. Uma taxa de incidência, praticamente, metade da taxa de incidências das pessoas com deficiência visual grave. Os IC95% não são coincidentes, o que significa que essa diferença é significativa. Houve, na amostra, uma redução da incidência de câncer entre os indivíduos totalmente cegos, sugerindo que a melatonina possa ser um fator protetor contra o câncer.

18.4.3 Relação entre prevalência e incidência

A incidência é uma medida de risco. A prevalência, por não levar em consideração o tempo de duração da doença (t), não tem esta capacidade. Em uma população onde a situação da doença encontra-se em estado estacionário (ou seja, sem grandes migrações ou mudanças ao longo do tempo na incidência/prevalência), a relação entre prevalência e incidência e duração da doença pode ser expressa pela seguinte fórmula (174):

$$prevalencia\ pontual = incidencia \times prevalencia$$

Por exemplo, se a incidência da doença for de 0,8% ao ano e sua duração média (sobrevida após o diagnóstico) for de 10 anos, a prevalência pontual será de aproximadamente 8%.

18.5 Medidas de associação

18.5.1 Odds Ratio

Odds Ratio (OR) é a razão entre dois *odds*. A *Odds Ratio*, traduzida como *Razão de Chances*, está associada, usualmente, com estudos retrospectivos tipo caso-controle com desfechos dicotômicos.

A *odds ratio* (OR) expressa a *odds* de exposição entre os que têm o desfecho (casos) pela *odds* de exposição nos livres de desfecho (controles).

Usando a Figura 200 e a fórmula $odds = \frac{p}{1-p}$, tem-se:

		Doença	Sem Doença	
		a	b	$a + b$
Expostos	Não expostos	c	d	$c + d$
		$a + c$	$b + d$	

Figura 200: Tabela de contingência 2 x 2

$$odds_{exp\ casos} = \frac{\frac{a}{a+c}}{1 - \frac{a}{a+c}} = \frac{a}{c}$$

$$odds_{exp\ controles} = \frac{\frac{b}{b+d}}{1 - \frac{b}{b+d}} = \frac{b}{d}$$

Portanto, a OR é igual a:

$$OR = \frac{odds_{exp\ casos}}{odds_{exp\ controles}} = \frac{\frac{a}{c}}{\frac{b}{d}} = \frac{a \times d}{c \times b}$$

18.5.1.1 Exemplo Em um estudo de caso-controle hipotético, a distribuição das exposições entre os casos e um grupo de pessoas saudáveis (“controles”) é comparada entre si. Os casos correspondem a um tipo raro de câncer, onde se suspeita que exista uma associação à exposição a um determinado fator de risco.

Os dados desse estudo hipotético estão no arquivo `dadosCasoControle.xlsx`. O conjunto de dados pode ser obtido [aqui](#). Depois de salvo em seu diretório de trabalho, ele pode ser carregado com a função `read_excel()` do pacote `readxl`.

```
cc <- readxl::read_excel ("dadosCasoControle.xlsx")
```

Colocar os níveis das variáveis `cc$exposto` e `cc$desfecho` como fatores e na ordem `sim`, `não`, uma vez que o R coloca em ordem alfabética (`não`, `sim`):

```
cc$exposto <- factor (cc$exposto,
                      levels = c("sim", "não"))

cc$desfecho <- factor (cc$desfecho,
                       levels = c("sim", "não"))
```

Construir uma tabela 2×2 :

```
tab_cc <- table (cc$exposto,
                  cc$desfecho,
                  dnn = c("Exposição", "Desfecho"))
tab_cc

##             Desfecho
## Exposição sim não
##       sim   48   20
##     não    12   40
```

A OR será obtida utilizando a função `epi.2by2()` do pacote `epiR` (156). Esta função tem os seguintes argumentos:

- `dat` → tabela de contingência 2×2 ;
- `method` → as opções são “cohort.count”, “cohort.time”, “case.control” ou “cross.sectional”.;
- `conf.level` → padrão = 0.95;
- `units` → multiplicador para incidência e prevalência;
- `outcome` → indicação de como a variável desfecho é representada na tabela de contingência (“as.columns” ou “as.rows”).

```
epiR::epi.2by2(tab_cc,
                 method = "case.control",
                 conf.level = 0.95,
                 units = 100,
                 outcome = "as.columns")
```

	Outcome +	Outcome -	Total	Odds
## Exposed +	48	20	68	2.40 (1.43 to 1.45)
## Exposed -	12	40	52	0.30 (0.13 to 0.53)
## Total	60	60	120	1.00 (0.69 to 1.45)
##				
## Point estimates and 95% CIs:				
## -----				
## Odds ratio			8.00 (3.49, 18.34)	

```

## Attrib fraction (est) in the exposed (%)      87.24 (69.26, 95.03)
## Attrib fraction (est) in the population (%)   70.00 (48.69, 82.46)
## -----
## Uncorrected chi2 test that OR = 1: chi2(1) = 26.606 Pr>chi2 = <0.001
## Fisher exact test that OR = 1: Pr>chi2 = <0.001
## Wald confidence limits
## CI: confidence interval

```

A Saída exibe os dados em uma tabela 2×2 , mostrando as odds e os IC95% e outras estatísticas epidemiológicas relacionadas.

A OR varia de zero ao infinito. Quando o valor da OR se aproxima de 1, a doença e o fator de risco não estão associados. Acima de 1 significa um que existe associação e valores menores de 1 indicam uma associação negativa (efeito protetor).

No exemplo hipotético, os indivíduos que se expuseram ao fator de risco têm uma chance 8 vezes maior de apresentar este tipo de câncer. O valor P do qui-quadrado é altamente significativo ($P < 0,001$).

18.5.2 Risco Relativo

O *Risco relativo* (RR) é a razão entre a incidência de desfecho em indivíduos expostos e a incidência de desfecho em indivíduos não expostos. O RR estima a magnitude da associação entre a exposição e o desfecho (doença). Em outras palavras, compara a probabilidade de ocorrência do desfecho entre os indivíduos expostos com a probabilidade de ocorrência do desfecho nos indivíduos não expostos.

A partir da tabela de contingência 2×2 (Figura 200), tem-se que o estimador do RR é dado por:

$$RR = \frac{incidencia_{exp}}{incidencia_{no\ exp}} = \frac{\frac{a}{a+b}}{\frac{c}{c+d}}$$

18.5.2.1 Exemplo Em 1940, ocorreu um surto de gastroenterite, após um jantar, em uma igreja, na cidade de Lycoming, Condado de Oswego, Nova York. Das 80 pessoas presentes, 75 foram entrevistadas. Quarenta e seis relataram doença gastrointestinal, atendendo à definição de caso.

As taxas de ataque (incidência) foram calculadas para aqueles que comeram e não comeram cada um dos 14 itens alimentares consumidos na ceia (175). O pacote `epitools` (176) contém os dados desta investigação no arquivo `oswego`.

```

data(oswego)
dplyr::glimpse(oswego)

## Rows: 75
## Columns: 21
## $ id                  <int> 2, 3, 4, 6, 7, 8, 9, 10, 14, 16, 17, 18, 20, 21, 2~
## $ age                 <int> 52, 65, 59, 63, 70, 40, 15, 33, 10, 32, 62, 36, 33~
## $ sex                 <chr> "F", "M", "F", "F", "M", "F", "F", "M", "F", ~
## $ meal.time            <chr> "8:00 PM", "6:30 PM", "6:30 PM", "7:30 PM", "7:30 ~
## $ ill                  <chr> "Y", "Y", "Y", "Y", "Y", "Y", "Y", "Y", "Y", ~
## $ onset.date           <chr> "4/19", "4/19", "4/19", "4/18", "4/18", "4/19", "4~
## $ onset.time            <chr> "12:30 AM", "12:30 AM", "12:30 AM", "10:30 PM", "1~
## $ baked.ham             <chr> "Y", "Y", "Y", "Y", "N", "N", "Y", "N", "Y", ~
## $ spinach               <chr> "Y", "Y", "Y", "Y", "N", "N", "Y", "N", "Y", ~
## $ mashed.potato          <chr> "Y", "Y", "N", "N", "Y", "N", "Y", "N", "N", "N", ~

```

```

## $ cabbage.salad      <chr> "N", "Y", "N", "Y", "N", "N", "N", "N", "N", "N", "N", ~
## $ jello             <chr> "N", "N", "N", "Y", "Y", "N", "N", "N", "N", "N", "N", ~
## $ rolls              <chr> "Y", "N", "N", "N", "Y", "N", "N", "Y", "N", "Y", "Y", ~
## $ brown.bread        <chr> "N", "N", "N", "N", "Y", "N", "N", "Y", "N", "N", "N", ~
## $ milk               <chr> "N", ~
## $ coffee              <chr> "Y", "Y", "Y", "N", "Y", "N", "N", "N", "N", "N", "Y", ~
## $ water              <chr> "N", "N", "N", "Y", "Y", "N", "N", "N", "Y", "N", "N", ~
## $ cakes              <chr> "N", "N", "Y", "N", "N", "N", "Y", "N", "N", "Y", "Y", ~
## $ vanilla.ice.cream  <chr> "Y", "Y", "Y", "Y", "Y", "N", "Y", "Y", "Y", "Y", "Y", ~
## $ chocolate.ice.cream <chr> "N", "Y", "Y", "N", "Y", "Y", "Y", "Y", "Y", "Y", "Y", ~
## $ fruit.salad         <chr> "N", ~

```

Observe que existem 75 observações de 21 variáveis, algumas características dos indivíduos como idade, sexo, etc. Importante para a análise é a variável `ill` (Y – sim, doente; N – não doente) e a variáveis relacionadas aos alimentos ingeridos durante o jantar na igreja. O sorvete de baunilha foi considerado o principal responsável pelo surtoo.

A seguir, as variáveis `oswego$vanilla.ice.cream` e `oswego$ill`²¹ serão transformadas em fator e os níveis colocados na ordem Y, N, uma vez que o R coloca em ordem alfabética ('N, Y') :

```

oswego$ill <- factor (oswego$ill,
                      levels = c ("Y", "N"))
oswego$vanilla.ice.cream <- factor (oswego$vanilla.ice.cream,
                                       levels = c ("Y", "N"))

```

Agora será construída uma tabela para o cálculo do RR:

```

tab_vanilla <- table (oswego$vanilla.ice.cream,
                       oswego$ill,
                       dnn = c ("Vanilla", "Ill"))
tab_vanilla

```

```

##           Ill
## Vanilla   Y  N
##          Y 43 11
##          N  3 18

```

O RR será obtido, utilizando a função `epi.2by2()` do pacote `epiR`, cujos argumentos foram mostrados no cálculo da OR, mudando a tabela para `tab_vanilla` e `method = "cohort.count"`:

```

epiR::epi.2by2(tab_vanilla,
                 method = "cohort.count",
                 conf.level = 0.95,
                 units = 100,
                 outcome = "as.columns")

```

	Outcome +	Outcome -	Total	Inc risk *
## Exposed +	43	11	54	79.63 (66.47 to 89.37)
## Exposed -	3	18	21	14.29 (79.63 to 79.63)
## Total	46	29	75	61.33 (49.38 to 72.36)
##				

²¹Foi mantido o nome das variáveis em inglês, pois no banco de dados `oswego` elas estão nessa língua.

```

## Point estimates and 95% CIs:
## -----
## Inc risk ratio           5.57 (1.94, 16.03)
## Odds ratio              23.45 (5.84, 94.18)
## Attrib risk in the exposed *   65.34 (46.92, 83.77)
## Attrib fraction in the exposed (%) 82.06 (48.41, 93.76)
## Attrib risk in the population *   47.05 (28.46, 65.63)
## Attrib fraction in the population (%) 76.71 (37.11, 91.37)
## -----
## Uncorrected chi2 test that OR = 1: chi2(1) = 27.223 Pr>chi2 = <0.001
## Fisher exact test that OR = 1: Pr>chi2 = <0.001
## Wald confidence limits
## CI: confidence interval
## * Outcomes per 100 population units

```

Os resultados da Saída indicam que os indivíduos que ingeriram sorvete de baunilha ($n = 54$) tiveram um risco maior de desenvolver gastroenterite aguda quando comparado aos que não ingeriram ($n = 21$). Dividindo o risco dos indivíduos expostos (incidência = 79,6) pelo risco dos não expostos (incidência = 14,3), encontra-se o RR = 5,57.

Quanto maior o RR mais forte é a associação entre a doença em questão e a exposição ao fator de risco. Um RR = 1 indica que a doença e a exposição ao fator de risco não estão associadas. Valores < 1 indicam uma associação negativa entre o fator de risco e a doença (efeito protetor).

18.5.3 Odds Ratio vs Risco Relativo

A OR não deve ser entendida como uma medida aproximada do RR, exceto para doenças raras (doenças, em geral com prevalência menor do que 10%). Caso contrário, a OR tenderá a superestimar a magnitude da associação e o OR afasta-se da hipótese nula da não associação (OR = 1), independentemente de ser um fator de risco ou de proteção. A discrepância (d)²² entre as estimativas do RR e OR pode ser definido como a razão entre o OR e o RR estimados (177). Em outras palavras, a discrepância corresponde a uma proporção do RR (178).

$$d = \frac{1 - p_{no\ exp}}{1 - p_{exp}} = \frac{\frac{c}{c+d}}{\frac{a}{a+b}}$$

Logo,

$$OR = RR \times d$$

Para finalizar, uma comparação entre OR e RR é mostrada na Tabela 25 (179).

18.5.4 Razão de Prevalência

Quando dados transversais estão disponíveis, muitas vezes as associações são avaliadas, usando a *razão de prevalências pontuais* (RPP). Tem o mesmo princípio das duas medidas anteriores, a razão de prevalência (RPP) compara a prevalência do desfecho entre os expostos com a prevalência do desfecho entre os não expostos.

Matematicamente, a RPP é calculada de maneira semelhante ao RR. Apenas, deve-se ter em mente que o desfecho e a exposição foram medidos no mesmo momento, enquanto que para o cálculo do RR há necessidade de calcular a incidência.

²²em inglês, *built-in bias*

Tabela 25: Força de associação do RR comparado com o OR.

OR	RR	Magnitude
1,0	1,0	insignificante
1,5	1,2	pequena
3,5	1,9	moderada
9,0	3,0	grande
32	5,7	muito grande
360	19	quase perfeita
infinito	infinito	perfeita

Usando uma tabela de contingência 2 x 2 (Figura 200), tem-se:

$$RPP = \frac{\text{prevalncia de doença}_{exp}}{\text{prevalncia de doença}_{no\ exp}} = \frac{\frac{a}{a+b}}{\frac{c}{c+d}}$$

Também é possível verificar a prevalência de exposição entre doentes e não doentes:

$$RPP = \frac{\text{prevalncia de exposição}_{doentes}}{\text{prevalncia de exposição}_{não\ doentes}} = \frac{\frac{a}{a+c}}{\frac{b}{b+d}}$$

18.5.4.1 Exemplo Em um estudo transversal (180), foi verificada a prevalência de infecções congênitas entre as puérperas com idade igual ou acima de 20 anos comparadas às mulheres com menos de 20 anos (adolescentes). A hipótese foi de que as adolescentes tinham uma prevalência maior de infecções.

Parte dos dados estão no arquivo `dadosMater.xlsx`, que contém, como já mencionado, informações de 1368 nascimentos. Entre essas, tem-se a idade das mães (`idadeMae`) e se foi diagnosticada infecção congênita (`infCong`).

O arquivo pode ser obtido [aqui](#). Depois de salvo em seu diretório de trabalho, ele pode ser carregado com a função `read_excel()` do pacote `readxl`.

```
dados <- readxl::read_excel ("dadosMater.xlsx")
```

A partir da variável `idadeMae`, criar a variável `faixaEtaria`, dividindo as parturientes em menores de 20 anos (adolescentes) e ≥ 20 anos. Para isso, usou-se a função `cut()` do pacote base. Revise os argumentos desta função.

```
dados$faixaEtaria <- cut (dados$idadeMae,
                           breaks=c(13, 20, 46),
                           labels = c("<20a", "≥20a"),
                           right = FALSE,
                           include.lowest = TRUE)
```

A variável `infCong` encontra-se como uma variável numérica e deve ser transformada emfator:

```
dados$infCong <- factor (dados$infCong,
                           ordered = TRUE,
                           levels = c (1,2),
                           labels = c ("sim", "não"))
```

Após estes procedimentos, constroi-se uma tabela 2×2 :

```
tab_infCong <- table(dados$faixaEtaria,
                      dados$infCong,
                      dnn = c("Faixa Etária", "Inf. Cong."))
tab_infCong
```

```
##           Inf. Cong.
## Faixa Etária sim não
##      <20a    7 212
##     =>20a   119 1030
```

Cálculo da RPP

Usando a tabela `tab_infCong` com a função `epi.2by2()` do pacote `epiR`, cujos argumentos foram mostrados no cálculo da OR e RR, mudando a tabela para `tab_infCong` e `method = "cross.sectional"`:

```
epiR::epi.2by2(tab_infCong,
                 method = "cross.sectional",
                 conf.level = 0.95,
                 units = 100,
                 outcome = "as.columns")
```

	Outcome +	Outcome -	Total	Prevalence *
## Exposed +	7	212	219	3.20 (1.29 to 6.47)
## Exposed -	119	1030	1149	10.36 (8.65 to 12.26)
## Total	126	1242	1368	9.21 (7.73 to 10.87)

```
## 
## Point estimates and 95% CIs:
## -----
## Prevalence ratio                  0.31 (0.15, 0.65)
## Odds ratio                        0.29 (0.13, 0.62)
## Attrib prevalence in the exposed * -7.16 (-10.08, -4.24)
## Attrib fraction in the exposed (%) -224.02 (-584.89, -53.29)
## Attrib prevalence in the population * -1.15 (-3.48, 1.19)
## Attrib fraction in the population (%) -12.45 (-17.53, -7.58)
## -----
## Uncorrected chi2 test that OR = 1: chi2(1) = 11.278 Pr>chi2 = <0.001
## Fisher exact test that OR = 1: Pr>chi2 = <0.001
## Wald confidence limits
## CI: confidence interval
## * Outcomes per 100 population units
```

A Saída exibe várias informações. Foi feita a hipótese de uma maior prevalência entre as mulheres com menos de 20 anos. Por este motivo, elas aparecem como as expostas (`Exposed +`) e tem uma prevalência de 3,20/100, enquanto que as mulheres com mais de 20 anos tiveram uma prevalência de 10,36/100. Isto mostra que a razão de prevalência é igual a 0,31, ou seja, < 1 , sugerindo que ao contrário da hipótese inicial, as adolescentes têm, neste estudo, uma menor prevalência de infecções congênitas.

18.6 Medidas de impacto

18.6.1 Risco Atribuível

O *Risco Atribuível* (RA) possui características de medida de impacto. O RA, ao invés de concentrar-se na associação em si, refere-se mais às consequências e às repercussões da exposição sobre a ocorrência do desfecho.

O RA é a medida do excesso ou acréscimo absoluto de risco que pode ser atribuído à exposição (181). Com o RA é possível estimar o número de casos que podem ser prevenidos se a exposição for eliminada e assim estimar a magnitude do impacto, em termos de saúde pública, imposto por esta exposição.

O risco de desenvolver o desfecho (incidência) está aumentado em RA nos indivíduos expostos em comparação com os que não estão expostos. Nos estudos de coorte, costuma-se usar mais a expressão *Risco Atribuível* ou *Diferença de Risco*. Nos ensaios clínicos, usa-se mais a expressão *Redução do Risco Absoluto* (RRA), pois se espera que a intervenção reduza o risco.

Calcula-se o RA ou a RRA pela diferença absoluta entre as incidências dos expostos e não expostos:

$$RA = |I_{expostos} - I_{no\ expostos}|$$

Utilizando a tabela de contingência 2×2 (Figura 200), o RA fica expresso da seguinte maneira:

$$RA = \left| \frac{a}{a+b} - \frac{c}{c+d} \right|$$

No exemplo mostrado no Risco Relativo, o RA pode ser calculado usando a mesma tabela de contingência, repetida aqui para facilitar a compreensão (Figura 201):

	Desfecho +	Desfecho -	Total
Exposição +	43	11	54
Exposição -	3	18	21
Total	46	29	75

Figura 201: Taxa de ataque de gastrorenterite com sorvete de baunilha - Oswego

Logo,

$$RA = \left| \frac{43}{43+11} - \frac{3}{3+18} \right| = |0,796 - 0,143| = 0,653$$

O risco atribuível na exposição mede o excesso de risco associado a uma determinada categoria de exposição. Por exemplo, com base no exemplo, a incidência cumulativa de gastrorenterite aguda entre os indivíduos que comeram o sorvete de baunilha é de 79,6% e para os que não ingeriram o sorvete (categoria de referência ou não exposta) foi de 14,3%. Desta forma, o risco excessivo associado à exposição $79,6 - 14,3 = 65,3\%$. Ou seja, assumindo uma associação causal (sem confusão ou viés), a não ocorrência da festa diminuiria o risco no grupo exposto de 79,6% para 14,3%.

O RA expresso em relação à incidência nos expostos e apresentado em percentual é denominado de *Risco Atribuível Proporcional* (RAP) ou *Fração Atribuível nos Expostos*.

O RAP informa qual a proporção de desfecho, expresso em percentagem, entre os expostos que poderia ter sido prevenida se a exposição fosse eliminada. É dado pela fórmula:

$$RAP = \left(\frac{I_{expostos} - I_{no\ expostos}}{I_{expostos}} \right) \times 100$$

No exemplo do surto de gastrorenterite aguda no jantar da igreja de Oswego, tem-se:

$$RAP = \left(\frac{0,796 - 0,143}{0,796} \right) \times 100 = 82,06\%$$

Se a causalidade foi estabelecida, essa medida pode ser interpretada como a porcentagem do risco total de gastrorenterite aguda que é atribuível à ingesta de sorvete de baunilha.

Outra maneira de se chegar a este mesmo resultado é através do RR, usando a seguinte fórmula

$$RAP = \left(\frac{I_{expostos} - I_{no\ expostos}}{I_{expostos}} \right) \times 100$$

$$RAP = \left(\frac{I_{expostos}}{I_{expostos}} - \frac{I_{no\ expostos}}{I_{expostos}} \right) \times 100$$

$$RAP = \left(1 - \frac{1}{\frac{I_{expostos}}{I_{no\ expostos}}} \right) \times 100$$

$$RAP = \left(1 - \frac{1}{RR} \right) \times 100$$

$$RAP = \left(\frac{RR - 1}{RR} \right) \times 100$$

No exemplo, o RR é igual a 5,57, logo:

$$RAP = \left(\frac{5,57 - 1}{5,57} \right) \times 100 = 82,05\%$$

18.6.2 Redução Relativa do Risco

Quando se avalia um tratamento ou alguma intervenção onde se supõe que haja uma redução do risco, por exemplo, uso da aspirina para reduzir infarto agudo de miocárdio, o termo Risco Atribuível é substituído por Redução do Risco Atribuível e é calculado da mesma forma visto na equação do Risco Atribuível.

Neste caso, ao invés de usar o Risco Atribuível Proporcional (RAP), onde se pressupõe que a exposição é um fator de risco para a doença e o $RR > 1$, usa-se a *Redução Relativa do Risco*, pois a exposição é supostamente um fator protetor e o $RR < 1$, como se espera que ocorra nos ensaios clínicos.

Esta medida análoga ao RAP é também chamada de *Eficácia*, definida como a proporção da incidência nos indivíduos não tratados (por exemplo, o grupo de controle) que é reduzida pela intervenção (182).

O cálculo da Redução Relativa do Risco (RRR) é semelhante ao Risco Atribuível Proporcional (RAP), onde a incidência nos expostos é a incidência no grupo que recebeu a intervenção (ou taxa de eventos no grupo tratamento) e a incidência nos não expostos é incidência nos controles (ou taxa de eventos nos controles – TEC). Como se supõe que a incidência nos controles seja maior que a incidência no grupo de tratamento, a equação fica:

$$RRR = \left(\frac{I_{controle} - I_{tratamento}}{I_{controle}} \right) \times 100$$

Alternativamente, a RRR pode ser estimada pela equação:

$$RRR = (1 - RR) \times 100$$

O *Physicians' Health Study* (49) é um ensaio clínico randomizado controlado, duplo cego, desenhado com o objetivo de determinar se uma dose baixa de aspirina (325 mg a cada 48 horas) diminui a mortalidade cardiovascular e se o betacaroteno reduz a incidência de câncer. Participaram deste estudo 22071 indivíduos por uma média de 60,2 meses.

O estudo do componente aspirina mostrou os seguintes resultados (Figura 202):

		Desfechos por ano		Total
		IAM	Sem IAM	
Aspirina		139 (a)	10898 (b)	11037
	Placebo	239 (c)	10795 (d)	11034

Figura 202: Physicians' Health Study, componente aspirina e IAM.

A incidência cumulativa de Infarto Agudo de Miocárdio (IAM) em ambos os grupos foi:

$$Incidencia_{aspirina} = \frac{139}{11037} = 0,0126$$

$$Incidencia_{placebo} = \frac{239}{11034} = 0,0217$$

$$RR = \frac{0,0126}{0,0217} = 0,58$$

Logo, a RRR é igual a:

$$RRR = (1 - 0,58) \times 100 = 42\%$$

Ou seja, houve uma redução de 42% no risco de IAM no grupo que usou aspirina e a conclusão dos autores foi que este ensaio clínico demonstrou, em relação à prevenção primária de doença cardiovascular, uma diminuição no risco de IAM.

Estes cálculos podem ser realizados com a função `risks()` do pacote `MKmisc` (183). Esta função calcula o risco relativo (RR), odds ratio (OR), redução relativa do risco (RRR) e outras estatísticas epidemiológicas, como RAR, NNT.

A função `risks()` usa como argumento:

- $p0$ → incidência do desfecho de interesse no grupo não exposto;
- $p1$ → incidência do desfecho de interesse no grupo exposto.

Antes de executar a função, deve-se instalar o pacote **BiocManager** para instalar o pacote **limma**:

```
BiocManager::install("limma")
```

A função **risks()** será usada dentro da função **round()** para reduzir o número de dígitos decimais:

```
p0 <- 0.0217
p1 <- 0.0126
round(MKmisc::risks(p0,p1), 4)
```

```
##      p0      p1      RR      OR      RRR      ARR      NNT
##  0.0217  0.0126  0.5806  0.5753  0.4194  0.0091 109.8901
```

18.6.3 Número Necessário para Tratar

Os resultados da função **risks()** entrega junto o *Número Necessário para Tratar* (NNT) que deve ser arredondado para o número inteiro mais próximo (no caso, 110) e significa a estimativa do número de indivíduos que devem receber uma intervenção terapêutica, durante um período específico de tempo, para evitar um efeito adverso ou produzir um desfecho positivo.

O NNT equivale à recíproca do RRA (Redução do Risco Absoluto ou Diferença de Risco):

$$NNT = \frac{1}{RRA} = \frac{1}{I_{controle} - I_{tratamento}}$$

No exemplo do *Physicians' Health Study*, o RRA igual a:

$$RA = |I_{expostos} - I_{no\ expostos}| = |0,0126 - 0,0217| = 0,0091$$

$$NNT = \frac{1}{0,0091} = 109,89 \simeq 110$$

Pode-se calcular os IC95%, calculando o NNT para os limites do RRA usando a seguinte equação (184):

$$IC_{95\%} \longrightarrow RRA \pm z_{(1-\alpha/2)} \times EP_{RRA}$$

Onde,

$$EP_{RRA} = \sqrt{\frac{p_0(1-p_0)}{n_1} + \frac{p_1(1-p_1)}{n_2}}$$

Usando os dados do *Physicians' Health Study*, pode-se criar um *script* no *RStudio* para os cálculos:

Vetor dos dados

```
a <- 139
b <- 10898
c <- 239
d <- 10795
dados <- c(a, b, c, d)
```

*Matriz dos dados*²³

```
mat_iam <- matrix (dados, byrow = TRUE, nrow = 2)
tratamento <- c ("aspirina", "placebo")
desfecho <- c ("IAM", "s/IAM")
rownames (mat_iam) <- tratamento
colnames (mat_iam) <- desfecho
mat_iam

##           IAM s/IAM
## aspirina 139 10898
## placebo   239 10795
```

Cálculo das incidências no grupo tratamento e no grupo placebo

Na matriz o que está entre colchetes [1,1] significa: linha 1 e coluna 1, ou seja, o valor 139.

```
n1 <- mat_iam [1,1] + mat_iam [1,2]
n1
```

```
## [1] 11037
```

```
p1 <- mat_iam [1,1] / n1
round (p1, 4)
```

```
## [1] 0.0126
```

```
n0 <- mat_iam [2,1] + mat_iam [2,2]
n0
```

```
## [1] 11034
```

```
p0 <- mat_iam [2,1] / n0
round (p0, 4)
```

```
## [1] 0.0217
```

Os resultados da matriz de dados e o cálculo das incidências $p0$ (incidência no grupo placebo) e $p1$ (incidência no grupo de tratamento) já eram conhecidos e foram repetidos apenas para entrar na programação do cálculo do IC95%.

Cálculo do erro padrão da RRA

```
RRA <- abs(p0 - p1)
round(RRA, 4)
```

```
## [1] 0.0091
```

²³Aproveite para revisar como construir matriz

```

NNT <- 1/RRA
round(NNT, 0)

## [1] 110

alpha <- 0.05
z <- qnorm (1 - (alpha/2))
round (z, 3)

## [1] 1.96

EP_RRA <- sqrt(((p0*(1-p0)) / n0) + (((p1*(1-p1)) / n1)))
round (EP_RRA, 4)

## [1] 0.0017

```

Cálculo do intervalo de confiança de 95% da RRA

ls = limite superior li = limite inferior

```

li_RRA <- RRA - (z * EP_RRA)
round (li_RRA, 4)

```

```
## [1] 0.0056
```

```

ls_RRA <- RRA + (z * EP_RRA)
round (ls_RRA, 4)

```

```
## [1] 0.0125
```

Portando, ao Redução Absoluta do Risco foi igual a 0,0091 (0,0056-0,0125). A partir destes resultados, pode-se calcular o intervalo de confiança para o NNT.

Cálculo do intervalo de confiança de 95% do NNT

```

li_NNT <- 1/ls_RRA
round (li_NNT, 0)

```

```
## [1] 80
```

```

ls_NNT <- 1/li_RRA
round (ls_NNT, 0)

```

```
## [1] 177
```

Concluindo, o uso da aspirina no *Physicians' Health Study* reduziu o risco de infarto agudo do miocárdio em 42% (RRR), ou seja, foi eficaz. Por outro lado, para ter este impacto será necessário tratar 110 (IC95%: 80-177) pacientes para que um tenha benefício. Este NNT é grande; o ideal é um NNT < 10. Apesar disso, como a aspirina tem baixo custo e se benefícios suplantarem os efeitos adversos, seu uso pode estar justificado.

18.6.4 Número Necessário para Causar Dano

Deve-se comparar o NNT com o *Número Necessário para causar Dano* (NNN), em inglês, *Number Needed to Harm* (NNH). Deve ser interpretado como o número de pacientes tratados para que um deles apresente um efeito adverso.

O NNN é calculado pela recíproca do aumento do risco absoluto (ARA), equivalente a diferença de risco ou redução do risco absoluto:

$$NNN = \frac{1}{ARA} = \frac{1}{I_{expostos} - I_{no\ expostos}}$$

18.6.4.1 Exemplo No *Physicians' Health Study* sobre o uso de aspirina na prevenção de IAM, foi verificado também os efeitos colaterais da aspirina, como acidentes vasculares cerebrais (AVC), Figura 203.

Desfechos por ano			
	AVC	Sem AVC	Total
Aspirina	119	10918	11037
Placebo	98	10936	11034

Figura 203: Physicians' Health Study, componente aspirina e AVC.

Cálculo das incidências

```
p0 <- 98/11034
round(p0, 4)
```

```
## [1] 0.0089
```

```
p1 <- 119/11037
round(p1, 4)
```

```
## [1] 0.0108
```

Para o cálculo do NNN, usa-se a função `risk()`, como mencionado antes:

```
p0 <- 0.0089
p1 <- 0.0108
round(MKmisc::risks (p0, p1), 4)
```

```
##      p0      p1      RR      OR      RRI      ARI      NNN
## 0.0089  0.0108  1.2135  1.2158  0.2135  0.0019  526.3158
```

Os resultados mostram que o NNN (NNH) é igual a 526. Ou seja, para evitar um IAM há necessidade de tratar 110 pacientes e a cada 526 tratados espera-se um caso de AVC, havendo um benefício bem maior quando comparado ao risco de AVC.

18.7 Análise de sobrevida

A análise de sobrevida é utilizada quando se pretende investigar o tempo entre o início de um estudo e a ocorrência subsequente de um evento que modifica o estado de saúde do indivíduo. É bastante usada em estudos sobre câncer, por exemplo, analisando o tempo desde a cirurgia até a morte, o tempo desde o início do tratamento até a progressão da doença, o tempo desde a resposta até a recorrência da doença. Ela também é usada para medir a ocorrência de outros eventos como o tempo desde a infecção pelo vírus da imunodeficiência humana (HIV) até o desenvolvimento da Síndrome de Imunodeficiência Adquirida (SIDA), o tempo de hospitalização, tempo de amamentação, etc.

O interesse está centrado na verificação do efeito dos fatores de risco ou de prognóstico sobre o tempo de sobrevida de um indivíduo ou de um grupo, bem como definir as probabilidades de sobrevida em diversos momentos no seguimento do grupo. Considera-se tempo de sobrevida, ou simplesmente sobrevida, o tempo a entre a entrada do indivíduo no estudo e a ocorrência do evento de interesse. Com relação aos dados relacionados ao tempo, podem ocorrer problemas. O tempo para um evento geralmente não tem distribuição normal. Além disso, nem sempre se pode esperar até que o evento ocorra em todos os pacientes e alguns pacientes abandonam o estudo mais cedo. Todos devem ser considerados dados e as análises de sobrevida contornam esses problemas.

Em estudos de sobrevida, os indivíduos são observados até a ocorrência de um evento final que, geralmente, corresponde à morte, ou à variação de um parâmetro biológico ou outro evento que indique a modificação do estado inicial (cura, recorrência, retorno ao trabalho, etc.) O evento final é denominado de *falha*, por referir-se, em geral, a algo indesejável.

18.7.1 Dados Censurados

Quando, em um estudo de sobrevida, os pacientes que saem do estudo ou que não vivenciam o evento são chamados de observações *censuradas*.

Esses tempos de sobrevida censurados subestimam o verdadeiro (mas desconhecido) tempo para o evento. Quando o evento (supondo que ocorreria) está além do final do período de acompanhamento, a censura costuma ser chamada de censura à direita.

A censura também pode ocorrer quando se observa a presença de um evento, mas não se sabe onde começou. Por exemplo, considere um estudo que investigue o tempo para a recorrência de um câncer após a remoção cirúrgica do tumor primário. Se os pacientes forem examinados 3 meses após a cirurgia e já tinham recorrência, então o tempo de sobrevida será censurado a esquerda, porque o tempo real (desconhecido) de recorrência ocorreu menos de 3 meses após a cirurgia.

Os dados de tempo do evento também podem ser censurados em intervalos, o que significa que os indivíduos entram e saem da observação. Se considerarmos o exemplo anterior e os pacientes também forem examinados aos 6 meses, aqueles que estão livres da doença aos 3 meses e perdem o acompanhamento entre 3 e 6 meses são considerados censurados no intervalo. A maioria dos dados de sobrevivência incluem observações censuradas à direita (185).

18.7.2 Método de Kaplan-Meier

O método de Kaplan-Meier (KM) é um método não paramétrico usado para estimar a probabilidade de sobrevivência a partir dos tempos de sobrevivência observados (186).

A função de sobrevida é a probabilidade de sobreviver a pelo menos um determinado ponto no tempo e o gráfico desta probabilidade é a curva de sobrevida. O método de sobrevida de Kaplan-Meier pode ser usado para comparar as curvas de sobrevida de dois ou mais grupos, como comparar um grupo tratado a um grupo não tratado (placebo), ou homens comparados a mulheres.

A curva de sobrevida KM, um gráfico da probabilidade de sobrevida de Kaplan-Meier em relação ao tempo, fornece um resumo útil dos dados que podem ser usados para estimar medidas como a mediana de sobrevida.

18.7.2.1 Pressupostos do método de Kaplan-Meier Os pressupostos para o uso da análise de sobrevida são as seguintes (187):

- os participantes devem ser independentes, ou seja, cada participante aparece apenas uma vez no grupo;
- os grupos devem ser independentes, ou seja, cada participante está apenas em um grupo;
- todos os participantes são livres de eventos quando se inscrevem no estudo;
- a medição do tempo até o evento deve ser precisa;
- o ponto inicial e o evento são claramente definidos;
- as perspectivas de sobrevida dos participantes permanecem constantes, ou seja, os participantes inscritos no início ou no final do estudo devem ter as mesmas perspectivas de sobrevida;
- a probabilidade de censura não está relacionada à probabilidade do evento.

Como em todas as análises, se o número total de pacientes em qualquer grupo for pequeno, digamos menos de 30 participantes em cada grupo, os erros padrão em torno das estatísticas resumidas serão grandes e, portanto, as estimativas de sobrevida serão imprecisas. Para estudos de sobrevida, recomenda-se fazer o cálculo do tamanho amostral previamente. O R dispõe de um pacote que possibilita este cálculo, o powerSurvEpi (188).

18.7.2.2 Exemplo O arquivo `dadosSobrevida.xlsx` contém as informações de 60 pacientes selecionados para um ensaio clínico randomizado hipotético de dois tratamentos nos quais 32 pacientes receberam o novo tratamento e 28 pacientes receberam o tratamento padrão. Para obter o arquivo, clique [aqui](#) e salve o mesmo em seu diretório de trabalho.

Destes pacientes, 33 eram mulheres e 27 homens. Durante o estudo (65 meses), um total de 21 pacientes morreram (7 mulheres e 14 homens).

Carregar o conjunto de dados

A partir do diretório de trabalho, carregue para um objeto que será denominado de sobrevida, usando a função `read_excel()` do pacote `readxl` e observe os dados com a função `head()`.

```
sobrevida <- readxl::read_excel("dadosSobrevida.xlsx")
head (sobrevida)
```

```
## # A tibble: 6 x 5
##       id evento tempo sexo  grupo
##   <dbl>  <dbl> <dbl> <chr> <chr>
## 1     22      0     5  fem   novo 
## 2     21      0     7  masc  novo 
## 3     19      0     8  fem   novo 
## 4     13      0     9  fem   novo 
## 5     50      1     9  masc  novo 
## 6     20      1    12  masc  novo
```

A Saída exibe um banco de dados com cinco variáveis:

- *id* → Identificação do indivíduo
- *evento* → Desfecho. 0 = censurado; 1 = morte
- *tempo* → Sobrevida em meses
- *sexo* → 1 = masculino; 2 = feminino
- *grupo* → Grupo de tratamento: 1 = nova droga; 2 = padrão

Construir uma tabela tratamento vs evento

```
table (sobrevida$grupo,
       sobrevida$evento,
       dnn = c("Tratamento", "Evento"))
```

```
##           Evento
## Tratamento 0  1
##      novo   24  8
##     padrão 15 13
```

A saída mostra o número em cada grupo, o número de eventos e o número censurados. Houve menos eventos, mas mais pacientes censurados no grupo do tratamento novo.

Calcular as estimativas de sobrevida de Kaplan-Meier para a construção da Curva de Sobrevida de cada tratamento

Para isso, usa-se a função `survfit()` do pacote `survival(189)`. Seus principais argumentos incluem:

- objeto de sobrevida, criado usando a função `Surv()` aninhada na função `survfit()`
- e o conjunto de dados contendo as variáveis.

Para a construção da tabela e da curva de sobrevida, digite e execute o seguinte:

```
tabsurv <- survfit (Surv (tempo, evento) ~ grupo, data = sobrevida)
summary(tabsurv)
```

```
## Call: survfit(formula = Surv(tempo, evento) ~ grupo, data = sobrevida)
##
##           grupo=novo
##   time n.risk n.event survival std.err lower 95% CI upper 95% CI
##    9     29      1   0.966  0.0339   0.9013   1.000
##   12     27      1   0.930  0.0479   0.8404   1.000
##   15     26      1   0.894  0.0579   0.7874   1.000
##   16     25      1   0.858  0.0657   0.7387   0.997
##   32     15      1   0.801  0.0826   0.6545   0.980
##   36     13      1   0.739  0.0965   0.5725   0.955
##   40     11      1   0.672  0.1086   0.4897   0.923
##   58      2      1   0.336  0.2438   0.0811   1.000
##
##           grupo=padrão
##   time n.risk n.event survival std.err lower 95% CI upper 95% CI
##    1     28      3   0.893  0.0585   0.7853   1.000
##    2     25      1   0.857  0.0661   0.7369   0.997
##    3     24      1   0.821  0.0724   0.6911   0.976
##    4     23      2   0.750  0.0818   0.6056   0.929
##    7     20      1   0.712  0.0859   0.5625   0.902
##   17     19      1   0.675  0.0892   0.5210   0.875
##   21     17      2   0.596  0.0947   0.4361   0.813
##   38      9      1   0.529  0.1048   0.3592   0.780
##   52      2      1   0.265  0.1944   0.0628   1.000
```

A Tabela de sobrevida é uma tabela descritiva com a coluna *time*, indicando o dia em que o evento ocorreu. A coluna *n.risk* indica o número de pacientes sob risco naquele momento. A coluna denominada *n.event*

indica o número total de pacientes que sofreram o evento desde o início do estudo até o momento avaliado. A coluna *survival* indica a proporção de pacientes que sobreviveram desde o início do estudo até aquele momento. Por exemplo, a sobrevida cumulativa é de 0,801 aos 32 meses no grupo tratamento novo e de 0,529 aos 38 meses no grupo tratamento padrão.

O método Kaplan-Meier produz uma única estatística resumida do tempo de sobrevida, isto é, a média ou mediana. O *tempo médio de sobrevida* é estimado a partir dos tempos observados e é mostrado para cada grupo na tabela de médias e medianas para o tempo de sobrevida.

A *sobrevida média* é calculada como a soma do tempo dividido pelo número de pacientes que permanecem sem censura. Essa estatística pode ser usada para indicar o período de tempo em que um paciente pode sobreviver. O *tempo mediano de sobrevida* é o ponto em que metade dos pacientes experimentou o evento. Se a curva de sobrevida não cair para 0,5 (ou seja, probabilidade de sobrevida de 50%), o tempo mediano de sobrevida não poderá ser calculado.

Estes dados podem ser visualizados na Saída, obtida com o comando:

```
summary(tabsurv)$table
```

	records	n.max	n.start	events	rmean	se(rmean)	median	0.95LCL
## grupo=novo	32	32	32	8	49.92533	4.078218	58	40
## grupo=padrão	28	28	28	13	36.62437	5.403382	52	21
## 0.95UCL								
## grupo=novo		NA						
## grupo=padrão		NA						

Visualização da curva de sobrevida

Pode-se visualizar a curva (Figura 204) de uma maneira simples, utilizando a função `plot()` do pacote básico do R:

```
plot (tabsurv, col = c ("steelblue", "rosybrown"), lwd = 2)
legend (legend = c ("Tratamento novo", "Tratamento padrão"),
        fill = c ("steelblue", "rosybrown"),
        bty="n",
        cex = 1,
        'bottomleft')
```

Outra maneira, mais sofisticada, de produzir a curva de KM é usando a função `ggsurvplot()`, incluída no pacote `survminer` (190) que utiliza o pacote `ggplot2` (Figura 205)

Com essa função é possível mostrar:

- os intervalos de confiança de 95% da função de sobrevida, usando o argumento `conf.int = TRUE`;
- o número e/ou a porcentagem de indivíduos em risco por tempo, utilizando a opção `risk.table`. Os valores permitidos para a `risk.table` incluem:
 - `TRUE` ou `FALSE` especificando se deve mostrar ou não a tabela de risco. O padrão é `FALSE`.
 - `absolute` ou `percentage`: para mostrar o número absoluto e o percentual de sujeitos em risco por tempo, respectivamente. Use `abs_pct` para mostrar o número absoluto e a porcentagem.
 - o valor *P* do teste Log-Rank comparando os grupos usando `pval = TRUE`.
 - linha horizontal/vertical na sobrevida mediana usando o argumento `surv.median.line`. Os valores permitidos incluem um de `c("nenhum", "hv", "h", "v")`. Onde *v* = vertical, *h* = horizontal.

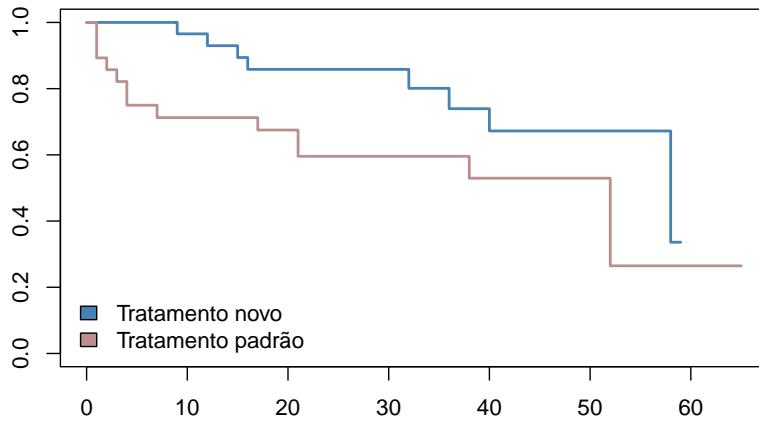


Figura 204: Curva de sobrevida comparando dois grupos de tratamento.

```
ggsurvplot (tabsurv,
            pval = TRUE,
            conf.int = TRUE,
            risk.table = "abs_pct",
            risk.table.col = "strata",
            surv.median.line = "hv",
            ggtheme = theme_bw (),
            legend.labs = c ("Tratamento Novo",
                           "Tratamento padrão"),
            palette = c ("steelblue", "rosybrown"))
```

O teste Log Rank pondera todos os pontos de tempo igualmente e é a estatística de sobrevida mais usada (191). O teste de log rank é um teste não paramétrico, que não faz suposições sobre as distribuições de sobrevivência. Os pressupostos deste teste são os mesmos do método de Kaplan-Meier. No exemplo, o valor P do teste é fornecido na Figura 205 e é igual a 0,083, ou seja, acima de 0,05, indicando não rejeição da H_0 . A hipótese nula diz que não há diferença na sobrevivência entre os dois grupos.

Essencialmente, o teste de log rank compara o número observado de eventos em cada grupo com o que seria esperado se a hipótese nula fosse verdadeira (ou seja, se as curvas de sobrevivência fossem idênticas). A estatística de log rank é aproximadamente distribuída como uma estatística de teste qui-quadrado.

A função `survdiff()`, também do pacote `survival`, pode ser usada para calcular o teste de log-rank comparando duas ou mais curvas de sobrevida e pode ser usado da seguinte forma:

```
dif_sobrevida <- survdiff (Surv (tempo, evento) ~ grupo, data = sobrevida)
dif_sobrevida
```

```
## Call:
## survdiff(formula = Surv(tempo, evento) ~ grupo, data = sobrevida)
##
##          N Observed Expected (O-E)^2/E (O-E)^2/V
```

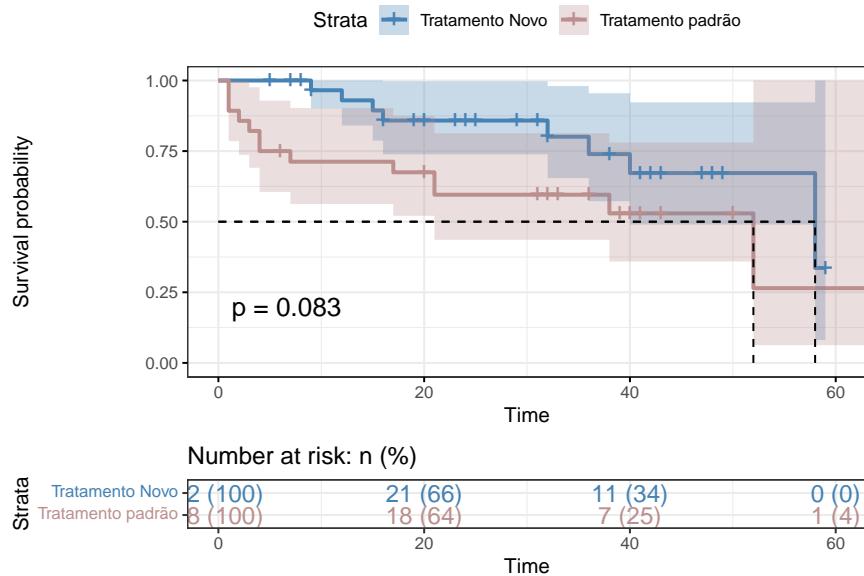


Figura 205: Curva de sobrevida comparando dois grupos de tratamento, usando ggsurvplot().

```
## grupo=novo      32      8    11.91      1.28      3.01
## grupo=padrão   28     13     9.09      1.68      3.01
##
## Chisq= 3  on 1 degrees of freedom, p= 0.08
```

A suposição de que o risco de um evento em um grupo em comparação com o outro grupo não muda ao longo do tempo é chamado de risco proporcional. Se as curvas de sobrevida se cruzam, isso sugere que os riscos não são proporcionais. Nessa situação, o teste log rank será menos poderoso e um teste alternativo deve ser considerado, como a *Regressão de Cox* ou *Modelo de Riscos Proporcionais*.

Regressão de Cox ou Modelo de Riscos Proporcionais

O modelo tem como objetivo a examinar simultaneamente como os fatores especificados influenciam a taxa de ocorrência de um determinado evento (por exemplo, infecção, morte) em um determinado ponto no tempo. Essa taxa é referida como *hazard ratio*.

Geralmente, as variáveis preditoras (ou fatores) são denominadas covariáveis. O modelo de Cox é expresso pela função de risco denotada por $h(t)$. Pode ser interpretada como o risco de morrer no tempo t e estimada da seguinte forma:

$$h(t) = h_0(t) \times e^{(b_1 x_1 + b_2 x_2 + \dots + b_n x_n)}$$

Onde,

- t é o tempo de sobrevida, indica que o risco varia com o tempo;
- $h(t)$ é a função de risco (*hazard*) determinada por um conjunto de n covariáveis (x_1, x_2, \dots, x_n)
- Os *coeficientes* (b_1, b_2, \dots, b_n) medem o tamanho do efeito das covariáveis
- $h(0)$ é o risco basal, o valor do risco se todos os x_i fossem iguais a zero ($\exp(0)$ é igual a 1).

As quantidades $\exp(b_i)$ são chamadas de *hazard ratio* (HR). Uma hazard ratio acima de 1 indica uma covariável que está positivamente associada à probabilidade do evento e, portanto, negativamente associada ao tempo de sobrevida.

Resumindo,

- HR = 1: Sem efeito
- HR <1: Redução do risco
- HR> 1: Aumento do risco

Para calcular o modelo de Cox no R serão utilizados os mesmos dados da do arquivo `dadosSobrevida.xlsx`.

O pacote `survival` tem uma função para calcular o modelo de Cox, `coxph()`, que usa os argumentos:

- *formula* → é o modelo linear com um objeto de sobrevida como variável desfecho. O objeto de sobrevida é criado usando a função `Surv()` como segue: `Surv(tempo, evento)`.
- **data*** → um banco de dados contendo as variáveis

```
mod.cox <- coxph (Surv (tempo, evento) ~ grupo, data = sobrevida)
mod.cox
```

```
## Call:
## coxph(formula = Surv(tempo, evento) ~ grupo, data = sobrevida)
##
##          coef  exp(coef)   se(coef)      z      p
## grupopadrão 0.7698     2.1593    0.4505  1.709 0.0875
##
## Likelihood ratio test=3.03  on 1 df, p=0.08171
## n= 60, number of events= 21
```

A função `summary()` fornece um relatório mais completo:

```
summary(mod.cox)
```

```
## Call:
## coxph(formula = Surv(tempo, evento) ~ grupo, data = sobrevida)
##
## n= 60, number of events= 21
##
##          coef  exp(coef)   se(coef)      z Pr(>|z|)
## grupopadrão 0.7698     2.1593    0.4505  1.709 0.0875 .
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##          exp(coef) exp(-coef) lower .95 upper .95
## grupopadrão     2.159      0.4631    0.893     5.221
##
## Concordance= 0.637  (se = 0.054 )
## Likelihood ratio test= 3.03  on 1 df,  p=0.08
## Wald test           = 2.92  on 1 df,  p=0.09
## Score (logrank) test = 3.06  on 1 df,  p=0.08
```

Os resultados da regressão de Cox, podem ser interpretados da seguinte forma:

- *Significância estatística.* A coluna marcada com *z* fornece o valor da estatística Wald. Corresponde à razão de cada coeficiente de regressão para seu erro padrão ($z = \frac{coef}{EP_{coef}}$). A estatística Wald avalia se o coeficiente beta (β) de uma determinada variável é estatisticamente diferente de 0. A partir da saída, pode-se concluir que não há diferença estatisticamente significativa entre os grupos ($P = 0,0875$).

- *Coeficientes de regressão.* A seguir deve-se observar, no modelo de Cox, o sinal dos coeficientes de regressão (coef). Um sinal positivo significa que o *hazard* (risco) é maior e, portanto, pior o prognóstico, para sujeitos com valores mais elevados dessa variável. No exemplo, a variável grupo é codificada como 1=novo, 2=padrão. O resumo do modelo de Cox fornece a *hazard ratio* (HR) para o segundo grupo em relação ao primeiro grupo, ou seja, tratamento padrão versus tratamento novo. O coeficiente beta para grupo = 0,7698 indica que os indivíduos do tratamento padrão têm maior risco de morte (taxas de sobrevivência mais baixas) do que os do grupo tratamento novo, nesses dados. Entretanto, esta diferença não é estatisticamente significativa.
- *Hazard ratios.* Os coeficientes exponenciados ($\exp(\text{coef}) = \exp(0,7698) = 2,1593$), também conhecidos como *hazard ratio*, fornecem o tamanho do efeito das covariáveis. Por exemplo, ser do grupo padrão aumenta o risco por um fator de 2,1593. Se esta diferença fosse significativa ($P < 0,05$), pertencer ao grupo padrão estaria associado a um mau prognóstico.
- *Intervalos de confiança das taxas de risco.* O resultado do resumo também fornece intervalos de confiança de 95% para a razão de risco ($\exp(\text{coef})$), limite inferior de 95% = 0,893, limite superior de 95% = 5,221, mostrando a não significância estatística, pois cruza o 1.
- *Significância estatística global do modelo.* Finalmente, a saída fornece valores de P para três testes alternativos para significância geral do modelo: O teste de razão de verossimilhança (*Likelihood ratio test*), teste de Wald e a estatística *logrank*. Esses três métodos são equivalentes. Para um tamanho amostral grande, eles darão resultados semelhantes. Para n pequeno, eles podem diferir um pouco. O teste de razão de verossimilhança tem melhor comportamento para tamanhos de amostra pequenos, por isso é geralmente preferido.

18.8 Regressão logística binária

A *regressão logística* (também conhecida como *regressão logit* ou *modelo logit*) foi desenvolvida como um a extensão do modelo linear pelo estatístico David Cox em 1958. Pertence a uma família, denominada *Modelo Linear Generalizado* (GLM) e é um modelo de regressão em que o desfecho Y é categórico.

A regressão logística permite estimar a probabilidade de uma resposta categórica com base em uma ou mais variáveis preditoras (X). Possibilita informar se a presença de um preditor aumenta (ou diminui) a probabilidade de um determinado desfecho em uma porcentagem específica. No caso em que Y é binário - ou seja, assume apenas dois valores, 0 e 1, que representam desfechos como aprovação/reprovação, sim/não, vivo/morto ou saudável/doente, tem-se a *regressão logística binária*.

Na regressão logística binária, as variáveis que afetam a probabilidade do resultado são medidas como *Odds Ratio*, que são chamadas de *Odds Ratios ajustadas* (192).

Na regressão linear, os valores das variáveis desfecho são preditos a partir de uma ou mais variáveis explicativas. Na regressão logística, uma vez que o desfecho é binário, a probabilidade de o desfecho ocorrer é calculada com base nos valores das variáveis explicativas. A regressão logística é semelhante à regressão linear na medida em que uma equação de regressão pode ser usada para prever a probabilidade de ocorrência de um desfecho. No entanto, a equação de regressão logística é expressa em termos logarítmicos (ou logits) e, portanto, os coeficientes de regressão devem ser convertidos para serem interpretados.

Embora as variáveis explicativas ou preditores no modelo possam ser variáveis contínuas ou categóricas, a regressão logística é mais adequada para medir os efeitos das exposições ou variáveis explicativas que são variáveis binárias. Variáveis contínuas podem ser incluídas, mas a regressão logística produzirá uma estimativa de risco para cada unidade de medida. Assim, a suposição de que o efeito de risco é linear sobre cada unidade da variável deve ser atendida e a relação não deve ser curva ou ter um ponto de corte sobre o qual o efeito ocorre. Além disso, as interações entre variáveis explicativas podem ser incluídas (192). Os casos em que a variável dependente tem mais de duas categorias de resultados podem ser analisados com *regressão logística multinomial*, não mostrada neste livro.

18.8.1 Pressupostos da regressão logística

O método de regressão logística assume que:

- O desfecho é uma variável binária ou dicotômica como sim vs não, positivo vs negativo, 1 vs 0.
- Existe uma relação linear entre o logit do desfecho e cada variável preditora. A função logit é $\text{logit}(P) = \log P \left(\frac{P}{(1-P)} \right)$, onde P é a probabilidade do desfecho. Na regressão linear, é assumido que o desfecho tem uma correlação linear com os preditores. Na regressão logística, o desfecho é categórico e, portanto, essa suposição é violada. Por isso que se usa o log (ou logit) dos dados. A suposição de linearidade na regressão logística, portanto, é que existe uma correlação linear entre quaisquer preditores contínuos e o logit da variável de desfecho.
- Os casos devem ser independentes. Os dados dos casos não devem ser relacionados; por exemplo, não pode medir as mesmas pessoas em pontos diferentes no tempo (medidas repetidas)
- Não há valores influentes (valores extremos ou *outliers*) nos preditores contínuos
- Não há altas intercorrelações (ou seja, multicolinearidade) entre os preditores.

Para melhorar a precisão de seu modelo, você deve certificar-se de que essas suposições sejam verdadeiras para seus dados.

18.8.2 Dados para a regressão logística

Os dados foram obtidos do banco de dados `PimaIndiansDiabetes2`, incluído no pacote `mlbench` (193) que contém 768 observações sobre 9 variáveis. Este conjunto de dados é originalmente do *Instituto Nacional de Diabetes e Doenças Digestivas e Renais*. São dados de mulheres com 21 anos ou mais com herança indígena Pima.

As variáveis traduzidas são:

- *gesta* → Número de vezes que engravidou
- *glicose* → Concentração de glicose plasmática após 2 horas de um teste oral de tolerância à glicose (mg%)
- *pd* → Pressão arterial diastólica (mm Hg)
- *tríceps* → Espessura da dobra cutânea do tríceps (mm)
- *insulina* → Insulina sérica após 2 horas (mu U/ml)
- *imc* → Índice de massa corporal (peso em kg/(altura em m)²)
- *pedigree* → Função de pedigree de diabetes
- *idade* → Idade (anos)
- *diabetes* → 0 = neg, 1 = pos

Depois de removidos os dados omissos, o banco de dados ficou como encontrado em `dadosPima.xlsx`. Estes dados serão utilizados para executar uma regressão logística binária para verificar se alguma dessas variáveis podem prever a presença de diabetes em mulheres com herança Pima.

Para obter arquivo, clique [aqui](#) e salve o mesmo em seu diretório de trabalho.

18.8.3 Preparação dos dados

Carregar os dados, usando a função `read_excel()` do pacote `readxl`:

```
dados <- readxl::read_excel("dadosPima.xlsx")
```

Para observar a estrutura dos dados, pode-se usar a função `glimpse()` do pacote `dplyr`:

```
dplyr::glimpse(dados)

## #> #> Rows: 392
## #> Columns: 9
## #> #> $ gesta      <dbl> 1, 0, 3, 2, 1, 5, 0, 1, 1, 3, 11, 10, 1, 13, 3, 3, 4, 4, 3, 9~
## #> #> $ glicose    <dbl> 89, 137, 78, 197, 189, 166, 118, 103, 115, 126, 143, 125, 97, ~
## #> #> $ pd         <dbl> 66, 40, 50, 70, 60, 72, 84, 30, 70, 88, 94, 70, 66, 82, 76, 5~
## #> #> $ triceps    <dbl> 23, 35, 32, 45, 23, 19, 47, 38, 30, 41, 33, 26, 15, 19, 36, 1~
## #> #> $ insulina   <dbl> 94, 168, 88, 543, 846, 175, 230, 83, 96, 235, 146, 115, 140, ~
## #> #> $ imc        <dbl> 28.1, 43.1, 31.0, 30.5, 30.1, 25.8, 45.8, 43.3, 34.6, 39.3, 3~
## #> #> $ pedigree   <dbl> 0.167, 2.288, 0.248, 0.158, 0.398, 0.587, 0.551, 0.183, 0.529~
## #> #> $ idade       <dbl> 21, 33, 26, 53, 59, 51, 31, 33, 32, 27, 51, 41, 22, 57, 28, 2~
## #> #> $ diabetes    <dbl> 0, 1, 1, 1, 1, 1, 0, 1, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0~
```

Para realizar a regressão logística, os dados devem ser inseridos como na regressão linear: organizados em colunas (uma representando cada variável). Olhando a saída da função `glimpse`, nota-se que as variáveis categóricas foram carregadas como `<dbl>` (numéricas). A variável diabetes foi codificada como 1 (evento ocorreu) e 0 (evento não ocorreu); neste caso, 1 representa ter diabetes e 0 representa uma ausência de diabetes. Como o R codifica os fatores na ordem 0 e 1, mantém-se assim e transforma-se a variável diabetes em fator.

```
dados$diabetes <- as.factor(dados$diabetes)
```

As variáveis idade e a variável gesta (número de gravidezes) são altamente assimétricas (Figura: 206):

```
# Este comando coloca os gráficos em uma mesma linha, em duas colunas:
par(mfrow=c(1,2))

# Histogramas
hist(dados$idade, breaks = 8, main = "", ylab = "Frequência", xlab = "Idade (anos)")
hist(dados$gesta, breaks = 12, main = "", ylab = "Frequência", xlab = "Nº de gravidezes")
```

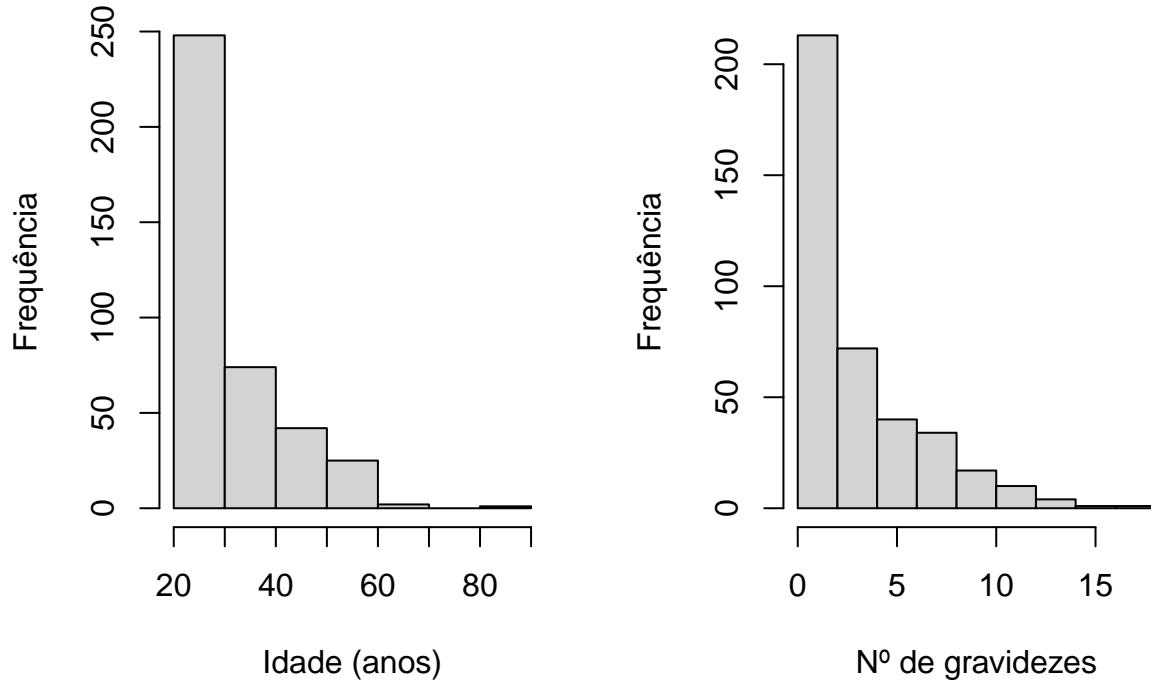


Figura 206: Histogramas das variáveis idade e número de gravidezes.

```
# Restaura as configurações basais de plotagem
par(mfrow=c(1,1))
```

Por isso, elas serão categorizadas como fatores com a função `cut()`, consultar a seção que trata da construção de tabelas de frequência:

```
dados$idadeCateg <- cut (dados$idade,
                         breaks= c (20,31,41,51,81),
                         labels = c ("20-30", "31-40", "41-50", ">50"),
                         right = FALSE,
                         include.lowest = TRUE)

dados$gestaCateg <- cut (dados$gesta,
                          breaks= c (0,6,11,17),
                          labels = c ("0-5", "6-10", ">10"),
                          right = FALSE,
                          include.lowest = TRUE)
```

Após a visualização e modificações das variáveis, construído um novo banco de dados (`dados2`), retirando as variáveis contínuas `idade` e `gesta` e mantendo as variáveis categóricas `idadeCateg` e `gestaCateg`:

```

dados2 <- dados %>%
  dplyr::select(-idade, -gesta)
glimpse(dados2)

## # Rows: 392
## # Columns: 9
## $ glicose    <dbl> 89, 137, 78, 197, 189, 166, 118, 103, 115, 126, 143, 125, 9~
## $ pd          <dbl> 66, 40, 50, 70, 60, 72, 84, 30, 70, 88, 94, 70, 66, 82, 76, ~
## $ triceps    <dbl> 23, 35, 32, 45, 23, 19, 47, 38, 30, 41, 33, 26, 15, 19, 36, ~
## $ insulina   <dbl> 94, 168, 88, 543, 846, 175, 230, 83, 96, 235, 146, 115, 140~
## $ imc         <dbl> 28.1, 43.1, 31.0, 30.5, 30.1, 25.8, 45.8, 43.3, 34.6, 39.3, ~
## $ pedigree   <dbl> 0.167, 2.288, 0.248, 0.158, 0.398, 0.587, 0.551, 0.183, 0.5~
## $ diabetes    <fct> 0, 1, 1, 1, 1, 1, 0, 1, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0, 1, ~
## $ idadeCateg <fct> 20-30, 31-40, 20-30, >50, >50, >50, 31-40, 31-40, 31-40, 20~
## $ gestaCateg <fct> 0-5, 0-5, 0-5, 0-5, 0-5, 0-5, 0-5, 0-5, 0-5, >10, 6-10~

```

18.8.4 Criação do modelo de regressão (*enter*)

Agora, pode-se prosseguir com a análise com uma regressão logística binária, usando a função `glm()` do pacote `stats` que pode usar vários argumentos:

- *formula* → objeto da classe “formula”. Um preditor típico tem a forma “resposta ~ preditor” em que resposta, na regressão logística binária, é uma variável dicotômica e o preditor pode ser uma série de variáveis numéricas ou categóricas;
- *family* → uma descrição da distribuição de erro e função de linka ser usada no modelo `glm`, pode ser uma *string* que nomeia uma função de *family*. O padrão é `family = gaussian()`. No caso da regressão logística binária ‘`family = binomial()`’ ou `family = binomial(link = "logit")`. Para outras informações, use `help(glm)` ou `help(family)`’;
- *data* → banco de dados.
- ... → ...

A função `glm()` diz ao R para executar um modelo linear generalizado. Dentro dos parênteses, são fornecidas informações importantes sobre o modelo. À esquerda do til (~) está a variável dependente: ‘’. Deve ser codificada como 0 e 1 para a função `glm` ler como binária. Após o til (~), são listadas as variáveis preditoras. Quando depois do sinal til é colocado um ponto (~.), significa a presença de todas as variáveis preditoras. Quando é colocado o asterisco (*) entre duas variáveis preditoras, isto indica que não se deseja apenas o efeito principal, mas também um termo de interação entre elas. No exemplo, não foi pedido essa análise. Finalmente, após a vírgula, especifica-se que a distribuição é binomial. A função link padrão em `glm` para uma variável desfecho binomial é o *logit*, portanto, não há necessidade de especificar no modelo.

Inicialmente, será feita uma regressão logística do tipo entrada forçada (*enter*), método padrão de conduzir uma regressão que consiste em simplesmente colocar todos os preditores no modelo de regressão em um bloco e estimar parâmetros para cada um (194). O conjunto de dados `dados2` será usado no modelo com todos os preditores dentro da função. O objeto criado será denominado `mod1`.

```

mod1 <- glm(diabetes ~ .,
             data = dados2,
             family = binomial())

```

Para ver o modelo gerado, há necessidade de executar a função `summary()`:

```

summary(mod1)

##
## Call:
## glm(formula = diabetes ~ ., family = binomial(), data = dados2)
##
## Deviance Residuals:
##    Min      1Q   Median      3Q     Max 
## -2.8862 -0.6414 -0.3546  0.5900  2.6253 
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)    
## (Intercept) -8.8655465  1.1691937 -7.583 3.39e-14 ***
## glicose      0.0392471  0.0059152  6.635 3.25e-11 ***
## pd            -0.0045595  0.0119562 -0.381  0.70294  
## triceps       0.0147965  0.0173948  0.851  0.39497  
## insulina     -0.0006838  0.0013559 -0.504  0.61402  
## imc           0.0630043  0.0273349  2.305  0.02117 *  
## pedigree      1.0165275  0.4388864  2.316  0.02055 *  
## idadeCateg31-40 0.8544642  0.3767035  2.268  0.02331 *  
## idadeCateg41-50 1.5745534  0.5203173  3.026  0.00248 ** 
## idadeCateg>50  1.3840205  0.6367486  2.174  0.02974 *  
## gestaCateg6-10 -0.2430053  0.4202225 -0.578  0.56308  
## gestaCateg>10  0.8284550  0.7673085  1.080  0.28028  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 498.10 on 391 degrees of freedom
## Residual deviance: 337.98 on 380 degrees of freedom
## AIC: 361.98
##
## Number of Fisher Scoring iterations: 5

```

Essas estatísticas resumidas ajudam a entender melhor o modelo, fornecendo-nos as seguintes informações:

- Distribuição dos desvios residuais;
- Estimativas do Intercepto e inclinação junto com o erro padrão, valor z (estatística de Wald) e valor P ;
- Valor AIC e
- Desvio residual e desvio nulo.

Em uma regressão logística, a resposta que está sendo modelada é o `log(odds)` ou `logit` de que o desfecho é igual a 1. Os coeficientes de regressão fornecem a mudança no `log(odds)` no desfecho para a mudança de uma unidade na variável preditora, mantendo todas as outras variáveis preditivas constantes (195).

Nas *variáveis contínuas*, para cada aumento de uma unidade na concentração da glicose, por exemplo, o `log(odds)` de ser diabético “1 = pos” (versus ser diabético “0 = neg”) aumenta em 0,039. Da mesma forma, para um aumento de unidade na pressão diastólica, a probabilidade de `log(odds)` de ser diabético “1 = pos” (versus ser diabético “0 = neg”) diminui em 0,0045, pois o coeficiente é negativo.

Para *variáveis categóricas*, o desempenho de cada categoria é avaliado em relação a uma categoria de base. A categoria de base para a variável `idadeCateg` é 20–30 (primeira categoria que aparece quando se observa os níveis²⁴) e para `gestaCateg` é 0–5. A interpretação de tais variáveis é a seguinte:

Estar na faixa etária de 31–40 anos, versus faixa etária de 20–30, muda o `log(odds)` de ser diabético “1 = pos” (versus ser diabético “0 = neg”) em 0,854. Estar na faixa de 6–10 gravidezes, versus 0–5 gravidezes, muda o `log(odds)` de ser diabético “pos” (versus ser diabético “neg”) em -0,24.

Como o `log(odds)` é difícil de interpretar, é possível exponenciar o `log(odds)` para colocar os resultados em uma escala de *odds ratios*. O R calcula as *odds ratios* e seus IC95, usando a função `exp()` e, dentro dela, a função `coef()` – do coeficiente b para as variáveis preditoras – e a função `confint()`, que fornece os intervalos de confiança. A função `cbind()` combinará as duas. A função `round()` é dispensável, foi usada apenas para que o resultado tenha 5 dígitos:

```
round (exp(cbind(OR = coef(mod1), confint(mod1))), 5)
```

```
## Waiting for profiling to be done...
```

	OR	2.5 %	97.5 %
## (Intercept)	0.00014	0.00001	0.00127
## glicose	1.04003	1.02846	1.05267
## pd	0.99545	0.97246	1.01935
## triceps	1.01491	0.98076	1.05019
## insulina	0.99932	0.99667	1.00200
## imc	1.06503	1.00999	1.12487
## pedigree	2.76358	1.18886	6.64511
## idadeCateg31-40	2.35011	1.11963	4.92734
## idadeCateg41-50	4.82858	1.75113	13.60151
## idadeCateg>50	3.99091	1.17784	14.52935
## gestaCateg6-10	0.78427	0.33953	1.77546
## gestaCateg>10	2.28978	0.53772	11.29542

A função `exp()` transformou o `log(odds)` em *odds*. A Saída mostra que a chance de ser diabético aumenta em um fator de 1,04 para um aumento de uma unidade na concentração de glicose, mantendo todas as outras variáveis constantes. Todo o IC95% encontra-se acima de 1. Isto ocorre com a `triceps` (prega cutânea tricipital), com o `imc`, `pedigree` (herança Pima), `idade acima de 30 anos`, indicando haver significância estatística, para essas variáveis, até este momento da análise.

18.8.5 Criação do modelo passo-a-passo (*stepwise*)

Existem vários métodos para seleção de variáveis, aqui será usado o método passo a passo (*stepwise*). O procedimento de seleção é realizado automaticamente por pacotes estatísticos como o pacote MASS (196) através da função `stepAIC()` que utiliza o AIC (Critério de Informação de Akaike²⁵) para a seleção (197). Esta função usa os seguintes argumentos:

- *object* → um objeto que representa um modelo de uma classe apropriada;
- *direction* → o modo de pesquisa passo a passo pode ser “backward” (para trás), “forward” (para frente)
`mod2 <- stepAIC(mod1, direction = "backward")` `summary(mod2)`

²⁴Se houver uma justificativa, esta referência pode ser modificada, usando a função `relevel()` do pacote `stats`. Por exemplo, para mudar a referência da variável `idadeCateg` para 31–40, deve-se executar: `dados$idadeCateg <- relevel(dadosPima$idadeCateg, ref = "31-40")`.

²⁵<https://www.scribbr.com/statistics/akaike-information-criterion/>

e) ou “both” (ambos) que é o padrão . Se o argumento scope estiver faltando, o padrão para a direção kwaré “bacd”;

- *scope* → define a gama de modelos examinados na pesquisa passo a passo
- ... → para maiores informações, use o *help*.

Execute o comando abaixo para realizar a seleção do modelo, que será recebido pelo objeto `mod2`:

```
mod2 <- stepAIC(mod1, direction = "backward")
```

```
## Start: AIC=361.98
## diabetes ~ glicose + pd + triceps + insulina + imc + pedigree +
##           idadeCateg + gestaCateg
##
##             Df Deviance   AIC
## - pd      1  338.13 360.13
## - insulina 1  338.24 360.24
## - gestaCateg 2  340.36 360.36
## - triceps  1  338.71 360.71
## <none>        337.98 361.98
## - imc      1  343.41 365.41
## - pedigree  1  343.61 365.61
## - idadeCateg 3  349.12 367.12
## - glicose   1  391.66 413.66
##
## Step: AIC=360.13
## diabetes ~ glicose + triceps + insulina + imc + pedigree + idadeCateg +
##           gestaCateg
##
##             Df Deviance   AIC
## - insulina 1  338.36 358.36
## - gestaCateg 2  340.43 358.43
## - triceps  1  338.84 358.84
## <none>        338.13 360.13
## - imc      1  343.49 363.49
## - pedigree  1  343.89 363.89
## - idadeCateg 3  349.25 365.25
## - glicose   1  391.88 411.88
##
## Step: AIC=358.36
## diabetes ~ glicose + triceps + imc + pedigree + idadeCateg +
##           gestaCateg
##
##             Df Deviance   AIC
## - gestaCateg 2  340.79 356.79
## - triceps   1  339.10 357.10
## <none>        338.36 358.36
## - imc       1  343.49 361.49
## - pedigree  1  344.00 362.00
## - idadeCateg 3  349.53 363.53
## - glicose   1  406.62 424.62
##
## Step: AIC=356.79
```

```

## diabetes ~ glicose + triceps + imc + pedigree + idadeCateg
##
##          Df Deviance    AIC
## - triceps     1   341.42 355.42
## <none>          340.79 356.79
## - imc         1   346.32 360.32
## - pedigree    1   346.82 360.82
## - idadeCateg  3   361.77 371.77
## - glicose     1   408.50 422.50
##
## Step:  AIC=355.42
## diabetes ~ glicose + imc + pedigree + idadeCateg
##
##          Df Deviance    AIC
## <none>          341.42 355.42
## - pedigree    1   347.78 359.78
## - imc         1   355.10 367.10
## - idadeCateg  3   363.70 371.70
## - glicose     1   409.44 421.44

summary (mod2)

##
## Call:
## glm(formula = diabetes ~ glicose + imc + pedigree + idadeCateg,
##      family = binomial(), data = dados2)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.9485 -0.6356 -0.3625  0.6233  2.6759
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.995423  1.008994 -8.915 < 2e-16 ***
## glicose      0.037429  0.005086  7.360 1.84e-13 ***
## imc          0.072975  0.020377  3.581 0.000342 ***
## pedigree     1.055869  0.430381  2.453 0.014154 *
## idadeCateg31-40 0.818513  0.334366  2.448 0.014367 *
## idadeCateg41-50 1.633945  0.399650  4.088 4.34e-05 ***
## idadeCateg>50  1.352995  0.528396  2.561 0.010450 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 498.10 on 391 degrees of freedom
## Residual deviance: 341.42 on 385 degrees of freedom
## AIC: 355.42
##
## Number of Fisher Scoring iterations: 5

```

Depois de implementar a função `stepAIC()`, restam agora quatro variáveis independentes - `glicose`, `imc`, `pedigree` e `idadeCateg`. De todos os modelos possíveis, este modelo (`mod2`) possui o valor mínimo de AIC. Além disso, as variáveis selecionadas têm valor $P < 0,05$.

Observe o cálculo das *odds ratios* com o `mod2` e compare com o `mod1`

```
round(exp(cbind(OR = coef(mod2), confint(mod2))), 5)

## Waiting for profiling to be done...

##          OR   2.5 % 97.5 %
## (Intercept) 0.00012 0.00002 0.00082
## glicose     1.03814 1.02820 1.04897
## imc         1.07570 1.03450 1.12092
## pedigree    2.87447 1.25945 6.80648
## idadeCateg31-40 2.26713 1.17331 4.36901
## idadeCateg41-50 5.12405 2.35800 11.37105
## idadeCateg>50  3.86900 1.40027 11.30915
```

Na Saída, observa-se que ter idade entre 41 a 50 anos tem uma chance 5,1 (IC95%: 2,4 – 11,4) maior de ser diabético comparado com a faixa etária 20-30 anos. As *odds ratios* ajustados da regressão logística binária fornecem uma estimativa que não é enviesada por confusão.

18.8.6 Uso do modelo para fazer previsões

A regressão logística fornece uma curva de probabilidade em forma de S (Figura 207). Pode ser utilizada a função `predict()`, uma função R genérica para fazer previsões a partir de modelos de ajuste. Se nenhum conjunto de dados for fornecido à função `predict()`, as probabilidades são calculadas a partir dos dados que foram usados para ajustar o modelo de regressão logística, no caso o `mod2`. A função `predict()` toma como argumentos o modelo de regressão e os valores da variável preditora (198). Para a função `mutate()` e o operador `pipe`, consulte `help()`.

- *object* → um objeto que representa um modelo de regressão
- *data* → banco de dados
- *type* → “reponse” ou “terms”
- ... →...

Esta função verifica o impacto da variação dos níveis de uma variável preditora sobre a probabilidade do desfecho. Por exemplo, verifica o impacto da concentração sérica de glicose na probabilidade de ser diabético.

```
predito <- predict(mod2, data=dados2, type = "response")
dados2 %>%
  mutate(predito = ifelse(diabetes == "1", 1, 0)) %>%
  ggplot(aes(glicose, predito)) +
  geom_point() +
  geom_smooth(method = "glm", method.args = list(family = "binomial")) +
  labs(title = NULL,
       x = "Glicemia (mg%)",
       y = "Probabilidade de diabetes")

## `geom_smooth()` using formula = 'y ~ x'
```

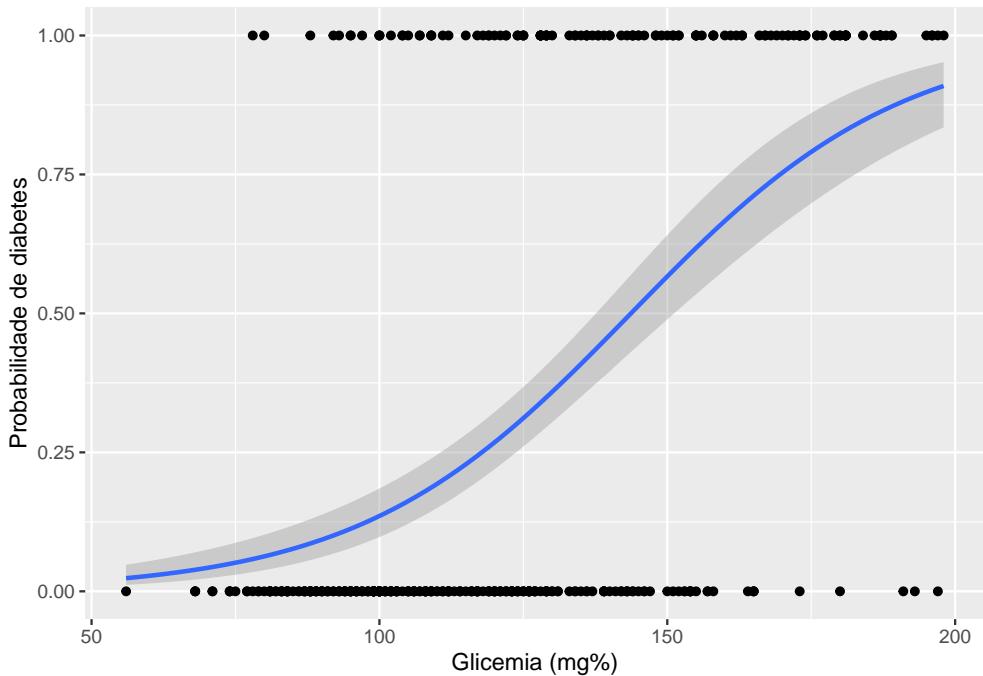


Figura 207: Probabilidade de diabetes de acordo com a glicemia em mulheres de herança Pima.

Uma vez que os coeficientes para a `glicose` foram estimados para o `mod2`, é uma questão simples calcular a probabilidade de diabetes para qualquer concentração da glicose. Por exemplo, qual a probabilidade de uma mulher da etnia Pima ser diabética com uma glicose de 180 mg%?

Em primeiro lugar constrói-se um modelo logístico novo, usando apenas a `glicose` como preditor:

```
mod3 <- glm(diabetes ~ glicose,
              family = binomial(link="logit"),
              data = dados2)
summary(mod3)

##
## Call:
## glm(formula = diabetes ~ glicose, family = binomial(link = "logit"),
##      data = dados2)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.1728   -0.7475   -0.4789    0.7153    2.3860
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.095521  0.629787 -9.679   <2e-16 ***
## glicose     0.042421  0.004761  8.911   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```

##      Null deviance: 498.10  on 391  degrees of freedom
## Residual deviance: 386.67  on 390  degrees of freedom
## AIC: 390.67
##
## Number of Fisher Scoring iterations: 4

```

A seguir, utiliza-se a `glicose = c(90, 120, 150, 180)`, criando assim um novo *dataframe*, chamado `glicemia`.

```

glicemia <- data.frame (glicose = c(90, 120, 150, 180))
glicemia

```

```

##   glicose
## 1     90
## 2    120
## 3    150
## 4    180

```

A partir daí calcula-se as predições, colocando, no dataframe `glicemia`, uma variável de probabilidade de predição (`pred.prob`):

```

glicemia$pred.prob = predict(mod3,
                             newdata=glicemia,
                             type="response")
glicemia

```

```

##   glicose pred.prob
## 1     90  0.09299244
## 2    120  0.26795891
## 3    150  0.56651015
## 4    180  0.82350197

```

A Saída mostra que a probabilidade de uma mulher de herança Pima ter diabete com uma glicemia de 90 mg% é 9,2%. Esta probabilidade sobe para 82,3% quando a glicemia é de 180 mg%.

Esta ferramenta permite também que o pesquisador faça a comparação das probabilidades usando outros preditores. Para isso basta acrescentar ao modelo de regressão logística outras variáveis, por exemplo, IMC igual a 25.

```

mod4 <- glm(diabetes ~ glicose + imc,
             family = binomial(link="logit"),
             data = dados2)
summary(mod4)

```

```

##
## Call:
## glm(formula = diabetes ~ glicose + imc, family = binomial(link = "logit"),
##      data = dados2)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.2112  -0.7396  -0.4114   0.7009   2.4306

```

```

## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.301460   0.927405 -8.951 < 2e-16 ***
## glicose      0.040713   0.004825  8.437 < 2e-16 ***
## imc         0.071794   0.019606  3.662 0.00025 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
## Null deviance: 498.10 on 391 degrees of freedom
## Residual deviance: 372.12 on 389 degrees of freedom
## AIC: 378.12
## 
## Number of Fisher Scoring iterations: 4

```

Acrescentar o ICM = 25 aos dados:

```

dados3 <- data.frame (glicose = c(90, 120, 150, 180),
                      imc = c(25, 25, 25, 25))
dados3

```

```

##   glicose imc
## 1      90  25
## 2     120  25
## 3     150  25
## 4     180  25

```

A seguir, faz-se as predições:

```

dados3$pred.prob = predict(mod4,
                           newdata=dados3,
                           type="response")
dados3

##   glicose imc pred.prob
## 1      90  25 0.05507376
## 2     120  25 0.16506181
## 3     150  25 0.40139839
## 4     180  25 0.69460853

```

Ou seja, uma mulher com herança Pima e um ICM = 25 kg/m² e uma glicemia de 180 mg% tem 69,4% de probabilidade de ser diabética.

Qual a probabilidade de ter diabetes mudando o IMC para 30?

Tente responder, seguindo os passos mostrados!

18.8.7 Avaliação do modelo

18.8.7.1 Linearidade Esta suposição pode ser testada examinando se o termo de interação entre o preditor e sua log transformação é significativo (199). Portanto há necessidade de criar os termos de interação de cada uma das variáveis independentes contínuas que estão no mod2 com seu log, usando a função log().

Glicose

```
dados2$glicoseInt <- log(dados2$glicose)*dados2$glicose
```

IMC

```
dados2$imcInt <- log(dados2$imc)*dados2$imc
```

Pedigree

```
dados2$pedigreeInt <- log(dados2$pedigree)*dados2$pedigree
```

Essas variáveis foram adicionadas ao conjunto de dados `dados2`. podem ser observadas com a função `glimpse()`.

```
dplyr::glimpse(dados2)
```

```
## Rows: 392
## Columns: 12
## $ glicose      <dbl> 89, 137, 78, 197, 189, 166, 118, 103, 115, 126, 143, 125, ~
## $ pd           <dbl> 66, 40, 50, 70, 60, 72, 84, 30, 70, 88, 94, 70, 66, 82, 76~
## $ triceps     <dbl> 23, 35, 32, 45, 23, 19, 47, 38, 30, 41, 33, 26, 15, 19, 36~
## $ insulina    <dbl> 94, 168, 88, 543, 846, 175, 230, 83, 96, 235, 146, 115, 14~
## $ imc          <dbl> 28.1, 43.1, 31.0, 30.5, 30.1, 25.8, 45.8, 43.3, 34.6, 39.3~
## $ pedigree     <dbl> 0.167, 2.288, 0.248, 0.158, 0.398, 0.587, 0.551, 0.183, 0.~
## $ diabetes     <fct> 0, 1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1~
## $ idadeCateg   <fct> 20-30, 31-40, 20-30, >50, >50, >50, 31-40, 31-40, 31-40, 2~
## $ gestaCateg   <fct> 0-5, 0-5, 0-5, 0-5, 0-5, 0-5, 0-5, 0-5, 0-5, >10, 6-1~
## $ glicoseInt   <dbl> 399.4886, 674.0374, 339.8233, 1040.7911, 990.6902, 848.590~
## $ imcInt        <dbl> 93.73513, 162.20784, 106.45360, 104.24066, 102.47621, 83.8~
## $ pedigreeInt   <dbl> -0.29889016, 1.89372743, -0.34579298, -0.29153532, -0.3666~
```

Para criar `dados4`, as variáveis `pd`, `insulina`, `triceps` e `gestaCateg` do `dados2` são removidas, pois mostraram-se não dados2 %>% quando se criou o mod2 no método passo-a-passo (`stepwise`):

```
dados4 <- dados2 %>%
  dplyr::select(-c(gestaCateg, pd, insulina, triceps))
names(dados4)
```

```
## [1] "glicose"      "imc"          "pedigree"      "diabetes"      "idadeCateg"
## [6] "glicoseInt"   "imcInt"       "pedigreeInt"
```

O conjunto de dados `dados4` será usado para criar um quinto modelo, `mod5`, para examinar o comportamento da interação entre o preditor e sua log transformação:

```
mod5 <- glm(diabetes~., family = binomial, data = dados4)
summary(mod5)
```

```

## 
## Call:
## glm(formula = diabetes ~ ., family = binomial, data = dados4)
## 
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max 
## -2.5954  -0.6326  -0.3411   0.6109   2.7501 
## 
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)    
## (Intercept) -17.80900  7.05884 -2.523  0.01164 *  
## glicose      0.09919  0.23220  0.427  0.66924    
## imc         0.90065  0.57720  1.560  0.11867    
## pedigree     2.08633  0.77570  2.690  0.00715 ** 
## idadeCateg31-40 0.81470  0.33792  2.411  0.01591 *  
## idadeCateg41-50 1.62746  0.40570  4.011 6.04e-05 *** 
## idadeCateg>50  1.37610  0.54080  2.545  0.01094 *  
## glicoseInt   -0.01047  0.03948 -0.265  0.79091    
## imcInt        -0.17971  0.12529 -1.434  0.15149    
## pedigreeInt   -1.62801  0.95970 -1.696  0.08981 .  
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
## Null deviance: 498.10  on 391  degrees of freedom 
## Residual deviance: 336.65  on 382  degrees of freedom 
## AIC: 356.65 
## 
## Number of Fisher Scoring iterations: 5

```

A Saída exibe a parte que testa a suposição de linearidade dos logit. O interesse está centrado apenas nos termos de interação. Qualquer interação que seja significativa indica que o efeito principal violou o pressuposto de linearidade do logit. Todas as interações têm valores de significância (valor P) maiores que 0,05, indicando que o pressuposto de linearidade do logit foi atendido.

18.8.7.2 Multicolinearidade A regressão logística é propensa ao efeito do viés da colinearidade e é essencial testar a colinearidade, após uma análise de regressão logística (200).

Os problemas de multicolinearidade consistem em incluir, no modelo, diferentes variáveis que possuem relação preditiva semelhante com o desfecho. Isso pode ser avaliado para cada preditor calculando o valor VIF (*Variance Inflation Factor*), usando a função `vif()` do pacote `car` (105) para o `mod2` que tem mais de dois termos.

```
car::vif(mod2)
```

```

##              GVIF Df GVIF^(1/(2*Df))
## glicose     1.050569  1      1.024973
## imc        1.023676  1      1.011769
## pedigree    1.026294  1      1.013062
## idadeCateg 1.088364  3      1.014213

```

Qualquer variável com um valor VIF alto (acima de 5 ou 10) deve ser removida do modelo. A Saída mostra

que todos os valores estão abaixo de 10. Isso leva a um modelo mais simples sem comprometer a precisão do modelo, o que é bom.

18.8.7.3 Estatística R^2 R ao quadrado é uma métrica útil para regressão linear simples e múltipla, mas não tem o mesmo significado na regressão logística.

Os estatísticos descobriram uma variedade de análogos de R ao quadrado para regressão logística que eles referem, coletivamente, como *pseudo R ao quadrado*. Não têm a mesma interpretação, na medida em que não são simplesmente a proporção da variância explicada pelo modelo.

Infelizmente, existem muitas maneiras diferentes de calcular um R^2 para regressão logística e nenhum consenso sobre qual é a melhor. Os dois métodos mais frequentemente relatados em softwares estatísticos são um proposto por McFadden (1974) e outro por Cox-Snell (1989). No entanto, também é bastante relatado o de Nagelkerke. Os valores mais altos indicam um melhor ajuste do modelo.

O R^2 de McFadden é uma versão, baseada no log-likelihood para o modelo somente com o intercepto e o modelo estimado completo. O R^2 de Cox e Snell é baseado no log-likelihood para o modelo em comparação com o log-likelihood para um modelo basal. No entanto, com resultados categóricos, tem um valor máximo teórico inferior a 1, mesmo para um modelo “perfeito”. O R^2 de Nagelkerke é uma versão ajustada do R^2 de Cox e Snell que ajusta a escala da estatística para cobrir todo o intervalo de 0 a 1.

Em seguida, será calculado os vários valores de R^2 , usando a função `PseudoR2()` do pacote `DescTools`:

```
PseudoR2(mod2, which =c("Nagelkerke", "McFadden", "CoxSnell"))
```

```
## Nagelkerke  McFadden  CoxSnell
##  0.4580199  0.3145606  0.3294780
```

Como se verifica, na Saída, todos os valores de R^2 diferem ligeiramente, mas podem ser usados como medidas de tamanho de efeito para o modelo.

Então, basicamente, o pseudo R quadrado pode ser interpretado como R^2 , mas não se espera que seja tão grande. Uma regra prática, bastante útil é que o pseudo R quadrado de McFadden variando de 0,2 a 0,4 indica um ajuste muito bom do modelo (201).

18.8.7.4 Pesquisa de valores atípicos e pontos de alavancagem Um *outlier* é uma observação que não é bem prevista pelo modelo de regressão ajustado (ou seja, tem um grande resíduo positivo ou negativo). Uma observação com um alto valor de alavancagem possui uma combinação incomum de valores preditores. Ou seja, é um outlier no espaço do preditor. O valor da variável dependente não é usado para calcular a alavancagem de uma observação.

Uma observação influente é uma observação que tem um impacto desproporcional na determinação dos parâmetros do modelo. As observações influentes são identificadas usando uma estatística chamada *distância de Cook* ou *D de Cook*.

A identificação dos *outliers* é feita essencialmente através dos resíduos padronizados, com a função `rstandard()`:

```
residuos_p <- rstandard(mod2)
summary(residuos_p)
```

```
##      Min. 1st Qu. Median Mean 3rd Qu. Max.
## -2.96350 -0.63838 -0.36367 -0.08147  0.63008  2.68035
```

Em uma amostra normalmente distribuída, ao redor de 95% dos valores estão entre $-1,96$ e $+1,96$, 99% deve estar entre $-2,58$ e $+2,58$ e quase todos (99,9%) devem situar-se entre $-3,09$ e $+3,09$. Portanto, resíduos padronizados com um valor absoluto maior que 3 são motivo de preocupação porque em uma amostra média é improvável que aconteça um valor tão alto por acaso (Field, 2012). Na Saída, observa-se que os valores estão dentro de -3 e $+3$.

Essa avaliação pode ser acompanhada de um gráfico diagnóstico (veja seção da regressão linear para maiores detalhes), usando a função `plot()` para a função modelo `mod2`.

```
plot (mod2, which = 5)
```

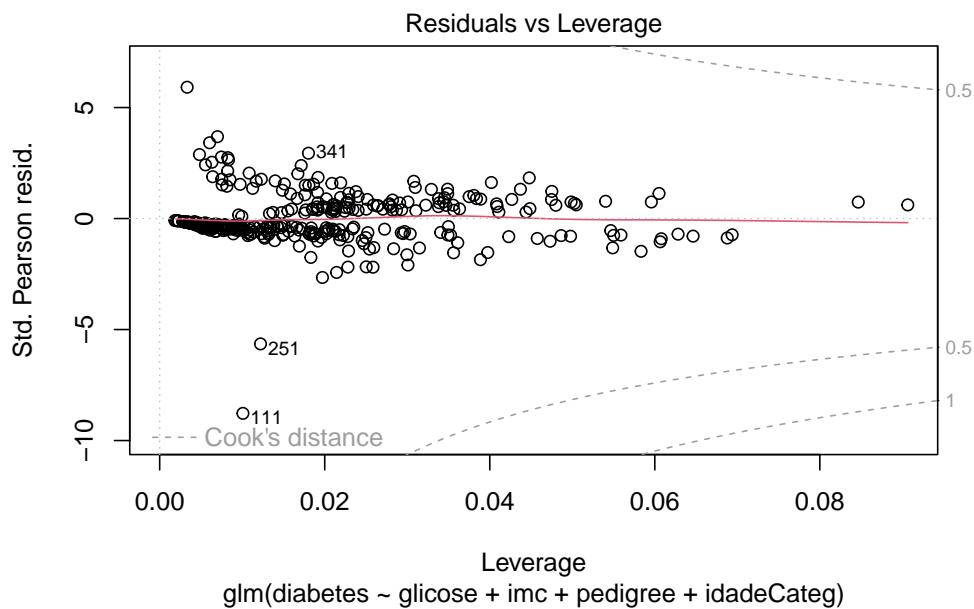


Figura 208: Gráfico diagnóstico dos resíduos e pontos de alavancagem.

São produzidos vários gráficos com a função `plot()`, mas o foco é o gráfico 5 (Figura 208) que confirma os achados de não existirem valores atípicos que comprometam o ajuste do modelo.

Para a análise dos pontos influentes, pode-se verificar os pontos de alavancagem (`leverage`) com a função `hatvalues()` do pacote `stats`, incluído no R base.

```
hat <- hatvalues(mod2)
summary (hat)
```

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	0.001847	0.005862	0.013332	0.017857	0.023846	0.090653

Os valores de alavancagem podem estar entre 0 (indicando que o caso não tem qualquer influência) e 1 (indicando que o caso tem grande influência). Se nenhum caso exercer influência indevida sobre o modelo, se espera que todos os valores de alavancagem estivessem próximos do valor médio. Alguns autores (202), recomendam investigar casos com valores superiores a duas vezes a média ($2 \times 0,0179 = 0,0358$) como ponto

de corte para identificar casos com influência indevida. Alguns valores estão acima de duas vezes a média. Entretanto, o maior valor está bem longe do valor igual a 1.

É interessante fazer a análise, observando junto a *distância de Cook* para ver se um ponto é um *outlier* significativo.

```
cook <- cooks.distance(mod2)
summary(cook)

##      Min.    1st Qu.     Median      Mean    3rd Qu.      Max.
## 2.220e-06 8.955e-05 5.703e-04 2.868e-03 2.908e-03 1.121e-01
```

Se a distância de Cook é < 1 , não há necessidade real de excluir esse ponto, uma vez que não tem um grande efeito na análise de regressão (203). Na Figura 208, verifica-se que os casos mais distantes não alcançam a distância de Cook.

18.8.7.5 Matriz de confusão É uma representação tabular de valores observados versus valores previstos. Ajuda a quantificar precisão (ou acurácia) do modelo. Agora será realizada uma comparação dos valores observados de “diabetes” com os valores previstos.

Inicialmente, será criada uma variável correspondente aos valores previstos (`mod2$fitted.values`) classificando como “pos” se o valor ajustado exceder 0,5, caso contrário, “neg”.

```
dados2$predito <- ifelse(mod2$fitted.values > 0.5, "pos", "neg")
```

Pode-se avaliar o desempenho do modelo (acurácia) com a comparação tabular entre os valores observados e os previstos:

```
tabDiabetes <- table(dados2$diabetes, dados2$predito)
rownames(tabDiabetes) <- c("Obs.neg", "Obs.pos")
colnames(tabDiabetes) <- c("Pred.neg", "Pred.pos")
tabDiabetes

##
##          Pred.neg Pred.pos
## Obs.neg      237      25
## Obs.pos       48      82
```

A *acurácia* é obtida pela soma dos valores das caselas na diagonal dividido pelo total.

```
acuracia <- sum(diag(tabDiabetes))/sum(tabDiabetes)
acuracia
```

```
## [1] 0.8137755
```

A partir da matriz de confusão a acurácia do modelo é de 81,4%.

18.8.7.6 Curva ROC A curva ROC permite explicar o desempenho do modelo avaliando a sensibilidade versus especificidade.

Para se obter a curva ROC será usada a função `roc()` do pacote `pROC` (veja Estatísticas Diagnósticas).

Os comandos abaixo exibem em seus resultados (Figura 209) a curva ROC e a área sob a curva (AUC) e seus IC95%. Quanto maior a área sob a curva melhor é o poder de predição do modelo (204).

```
roc (dados2$diabetes,
      mod2$fitted.values,
      plot=TRUE,
      legacy.axes=TRUE,
      print.auc=TRUE,
      print.auc.y = 0.2,
      ci = TRUE,
      ylab="Sensibilidade",
      xlab="1 - Especificidade",
      col="steelblue",
      lwd=2)
```

```
## Setting levels: control = 0, case = 1

## Setting direction: controls < cases
```

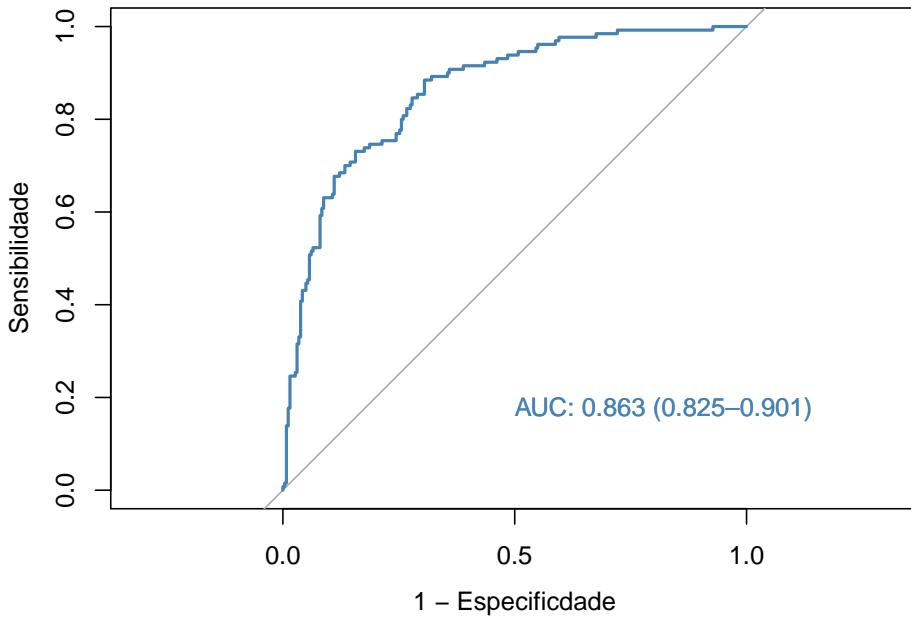


Figura 209: Curva ROC do modelo de regressão.

```
##
## Call:
## roc.default(response = dados2$diabetes, predictor = mod2$fitted.values,      ci = TRUE, plot = TRUE,
## ##
```

```
## Data: mod2$fitted.values in 262 controls (dados2$diabetes 0) < 130 cases (dados2$diabetes 1).
## Area under the curve: 0.8631
## 95% CI: 0.8254-0.9007 (DeLong)
```

A Figura 209 exibe uma a curva ROC do `mod2` onde se observa que AUC = 0,863. Isto significa que a performance do modelo como preditor é muito boa (ver Tabela 23).

Referências

1. Armitage P, Berry G, Matthews JNS. Statistical methods in medical research. John Wiley & Sons; 2008.
2. Massad E, Silveira PSP, Menezes RX de, Ortega NRS. Métodos quantitativos em medicina. Editora Manole Ltda; 2004.
3. Kendall MG. Studies in the history of probability and statistics. Where shall the history of statistics begin? *Biometrika*. 1960;47(3/4):447–9.
4. Breve História dos Censos [Internet]. Instituto Nacional de Estatística. Statistics Portugal; 2014. Disponível em: https://censo.ine.pt/xportal/xmain?xpid=CENSOS&xpgid=censos_bhistoria
5. Salgado-Neto G, Salgado A. Sir Francis Galton e os extremos superiores da curva normal. *Revista de Ciências Humanas*. 2011;45(1):223–39.
6. Stolley PD, Lasky T. The Beginnings of Epidemiology. Em: Investigating Disease Patterns. Scientific American Library; 2000. p. 23–49.
7. Editors History com. Florence Nightingale. <https://www.history.com/topics/womens-history/florence-nightingale-1>;
8. Moore DS. Topics in Inferency. Em: The basic practice of statistics. W.H. Freeman; 2000. p. 417.
9. Salsburg D. Uma senhora toma chá... Em: Uma senhora toma chá. Zahar; 2009. p. 17–23.
10. Hald A. Biography of Fisher. Em: A History of Parametric Statistics Inference from Bernoulli to Fisher, 1713–1935. John Wiley & Sons; 2007. p. 159–63.
11. Kruskal W. The Significance of Fisher: A Review of R.A. Fisher: The Life of a Scientist. *Journal of the American Statistical Association* [Internet]. 1980;75(372):1019–30. Disponível em: <https://doi.org/10.1080/01621459.1980.10477590>
12. Stolley PD, Lasky T. Lung Cancer: New Methods of Studying Disease. Em: Investigating Disease Patterns. Scientific American Library; 2000. p. 51–79.
13. Matthews R, Chalmers I, Rothwell P, Douglas G Altman: statistician, researcher, and driving force behind global initiatives to improve the reliability of health research. British Medical Journal Publishing Group; 2018.
14. Altman DG. The scandal of poor medical research. Vol. 308, Bmj. British Medical Journal Publishing Group; 1994. p. 283–4.
15. R Core Team. The R Project for Statistical Computing | What is R? Disponível em: <https://www.r-project.org/about.html>; 2022.
16. R Core Team. The R Project for Statistical Computing | CRAN Mirrors. Disponível em: <https://cran.r-project.org/mirrors.html>; 2022.
17. Front Matter. Em: The R Book [Internet]. John Wiley & Sons, Ltd; 2012. p. i–xxiv. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118448908>
18. Oliveira Filho PF de. Natureza dos Dados. Em: Epidemiologia e Bioestatística—Fundamentos para a Leitura Crítica. 2^a edição. Editora Rubio; 2022. p. 3–6.
19. Kirkwood BR, Sterne JA. Defining the Data. Em: Essential Medical Statistics. Second Edition. Blackwell Science Company; 2003. p. 9–14.
20. Sternbach GL. The Glasgow coma scale. *The Journal of emergency medicine*. 2000;19(1):67–71.
21. Pediatrics AA of, Fetus C on, Newborn, Obstetricians AC of, Gynecologists, Obstetric Practice C on. The apgar score. *Pediatrics*. 2006;117(4):1444–7.
22. Bowers D. First things first—the nature of data. Em: Medical Statistics from Scratch. Second Edition. John Wiley; Sons; 2008. p. 3–13.

23. Ribeiro Mendes F. O que é um trabalho científico. Em: Iniciação Científica. Autonomia Editora; 2012. p. 17–26.
24. Hulley SB, Cummings SR, Browner WS, Grady DG, Newman TB. Elaborando a questão de pesquisa e desenvolvendo o plano de estudo. Em: Delineando a pesquisa clínica. Quarta Edição. Artmed Editora; 2015. p. 15–24.
25. McCombes S. Sampling Methods [Internet]. <https://www.scribbr.com/methodology/sampling-methods/>. scribbr.com Team; 2019. Disponível em: <https://www.scribbr.com/>
26. Callegari-Jacques SM. Amostras. Em: Bioestatística: princípios e aplicações. Artmed Editora; 2003. p. 146–7.
27. Faul F, Erdfelder E, Lang A-G, Buchner A. G* Power 3: A flexible statistical power analysis program for the social, behavioral, and biomedical sciences. Behavior research methods. 2007;39(2):175–91.
28. Cohen J. Statistical power analysis for the behavioral sciences. Lawrence Erlbaum Associates; 1988.
29. Grimes DA, Schulz KF. An overview of clinical research: the lay of the land. The Lancet. 2002;359(9300):57–61.
30. Fletcher RH, Fletcher SW, Fletcher GS. Prognóstico. Em: Epidemiologia Clínica: Elementos Essenciais. Artmed Editora; 2014. p. 108–9.
31. Grimes DA, Schulz KF. Descriptive studies: what they can and cannot do. The Lancet. 2002;359(9301):145–9.
32. Fletcher RH, Fletcher SW, Fletcher GS. Risco: da doença à exposição. Em: Epidemiologia Clínica: Elementos Essenciais. Artmed Editora; 2014. p. 88.
33. Grimes DA, Schulz KF. Compared to what? Finding controls for case-control studies. The Lancet. 2005;365(9468):1429–33.
34. Ernster VL. Nested case-control studies. Preventive Medicine. 1994;23(5):587–90.
35. Newman TB, Browner WS, Cummings SR, Hulley SB. Delineando estudos de caso-controle. Em: Delineando a pesquisa clínica. Quarta Edição. Artmed Editora; 2015. p. 111.
36. Grimes DA, Schulz KF. Cohort studies: marching towards outcomes. The Lancet. 2002;359(9303):341–5.
37. Fletcher RH, Fletcher SW, Fletcher GS. Risco: da doença à exposição. Em: Epidemiologia Clínica: Elementos Essenciais. Artmed Editora; 2014. p. 68.
38. Kannel WB, McGee DL. Diabetes and cardiovascular risk factors: the Framingham study. Circulation. 1979;59(1):8–13.
39. Coutinho M. Princípios de epidemiologia clínica aplicada a cardiologia. Arquivos Brasileiros de Cardiologia. 1998;71:109–16.
40. McCambridge J, Witton J, Elbourne DR. Systematic review of the Hawthorne effect: new concepts are needed to study research participation effects. Journal of Clinical Epidemiology. 2014;67(3):267–77.
41. Bland JM, Altman DG. Statistic Notes: Regression towards the mean. BMJ. 1994;308(6942):1499.
42. Fletcher RH, Fletcher SW, Fletcher GS. Tratamento. Em: Epidemiologia Clínica: Elementos Essenciais. Artmed Editora; 2014. p. 143.
43. Kabisch M, Ruckes C, Seibert-Grafe M, Blettner M. Randomized controlled trials: part 17 of a series on evaluation of scientific publications. Deutsches Ärzteblatt International. 2011;108(39):663.
44. Elander G, Hermerén G. Placebo effect and randomized clinical trials. Theoretical Medicine. 1995;16(2):171–82.
45. Schulz KF, Grimes DA. Blinding in randomised trials: hiding who got what. The Lancet. 2002;359(9307):696–700.
46. Montori VM, Guyatt GH. Intention-to-treat principle. CMAJ. 2001;165(10):1339–41.

47. Christensen E. Methodology of superiority vs. equivalence trials and non-inferiority trials. *Journal of hepatology*. 2007;46(5):947–54.
48. Health Improvement O for, Disparities. Crossover randomised controlled trial: comparative studies [Internet]. Office for Health Improvement and Disparities. UK Health improvement; 2020. Disponível em: <https://www.gov.uk/guidance/crossover-randomised-controlled-trial-comparative-studies>
49. Physicians' Health Study Research Group* SC of the. Final report on the aspirin component of the ongoing Physicians' Health Study. *New England Journal of Medicine*. 1989;321(3):129–35.
50. Hennekens CH, Buring JE, et al. Lack of effect of long-term supplementation with beta carotene on the incidence of malignant neoplasms and cardiovascular disease. *New England Journal of Medicine*. 1996;334(18):1145–9.
51. Stanley K. Design of randomized controlled trials. *Circulation*. 2007;115(9):1164–9.
52. Chang W. Cookbook for R. Cookbook for R. <http://www.cookbook-r.com>; 2021.
53. Verzani J. Using R for introductory statistics. Chapman; Hall/CRC; 2004.
54. Damiani A, Milz B, Lente C, al et. Ciência de Dados em R [Internet]. R6 Consultoria; 2015. Disponível em: <https://livro.curso-r.com/index.html>
55. Zuur AF, Ieno EN, Meesters EH. Getting Data into R. Em: A Beginner's Guide to R. Springer; 2009. p. 29–56.
56. Wickham H, Grolemund G. 15 Factors|R for data science [Internet]. Welcome | R for Data Science. O'Reilly; 2017. Disponível em: <https://r4ds.had.co.nz/factors.html>
57. Ooms J. writexl: Export Data Frames to Excel 'xlsx' Format [Internet]. 2022. Disponível em: <https://CRAN.R-project.org/package=writexl>
58. Team RC. write.table: Data Output/CSV files [Internet]. DataCamp; 2022. Disponível em: <https://www.rdocumentation.org/packages/utils/versions/3.6.2/topics/write.table>
59. Wickham H, Averick M, Bryan J, Chang W, et al. Welcome to the Tidyverse. *Journal of open source software*. 2019;4(43):1686.
60. Wickham H, Girlich M. tidyR: Tidy Messy Data [Internet]. 2022. Disponível em: <https://CRAN.R-project.org/package=tidyr>
61. Wickham H. Tidy Data. *Journal of Statistical Software*. 2014;59(10):11–23.
62. Fisher RA. The use of multiple measurements in taxonomic problems. *Annals of eugenics*. 1936;7(2):179–88.
63. Wickham H, François R, Henry L, Müller K, et al. dplyr: A grammar of data manipulation. R package version 04. 2015;3:156.
64. Wickham H, Bryan J. readxl: Read excel files. R package version. 2019;1(1):1–3.
65. Field A, Miles J, Field Z. Everithing you ever wanted to know about statistics (well, sort of). Em: Discovering statistics using R. Sage Publications, Ltd; 2012. p. 38.
66. Arango HG. Organização dos dados em tabelas. Em: Bioestatística: teórica e computacional. 3^a edição. Guanabara Koogan; 2009. p. 32–57.
67. Oliveira Filho PF de. Tabelas. Em: Epidemiologia e Bioestatística-Fundamentos para a Leitura Crítica. 2^a edição. Editora Rubio; 2022. p. 9–12.
68. Xie Y. knitr: a comprehensive tool for reproducible research in R. Em: Implementing reproducible research. Chapman; Hall/CRC; 2018. p. 3–31.
69. Zhu H et al. Construct complex table with “kable” and Pipe syntax [Internet]. 2021. Disponível em: <https://CRAN.R-project.org/package=kableExtra>

70. Arango HG. Números de classes e Intervalo de Classes. Em: Bioestatística teórica e computacional. Terceira edição. Guanabara Koogan, RJ; 2009. p. 35–40.
71. Rasmussen KM, Yaktine AL, et al. Weight gain during pregnancy: reexamining the guidelines. 2009;
72. Field A, Miles J, Field Z. Exploring data with graphs. Em: Discovering statistics using R. Sage Publications, Ltd; 2012. p. 117.
73. Wickham H. Getting Started with ggplot2. Em: ggplot2. Second edition. Springer; 2016. p. 11–31.
74. Tufte ER. Aesthetics and Technique in Data Graphical Design. Em: The Visual Display of Quantitative Information. Second edition. Graphics Press; 2001. p. 178.
75. Lemon J, Bolker B, Oom S, et al. Package “plotrix”. Vienna: R Development Core Team. 2015;
76. Kabacoff RI. Basic graphs. Em: R in Action: Data analysis and graphics with R. Manning Publications Co.; 2011. p. 120–4.
77. Harrell FE, Dupont C. Hmisc: Harrell Miscellaneous [Internet]. R package version. 2022. Disponível em: <https://cran.r-project.org/web/packages/Hmisc/index.html>
78. Wickham H. ggplot2: Elegant Graphics for Data Analysis [Internet]. Springer-Verlag New York; 2016. Disponível em: <https://ggplot2.tidyverse.org>
79. Wickham H. A layered grammar of graphics. Journal of Computational and Graphical Statistics. 2010;19(1):3–28.
80. Rinker TW, Kurkiewicz D. pacman: Package Management for R [Internet]. Buffalo, New York; 2018. Disponível em: <http://github.com/trinker/pacman>
81. Wickham H, Seidel D. scales: Scale Functions for Visualization [Internet]. 2022. Disponível em: <https://CRAN.R-project.org/package=scales>
82. Debnath L, Basu K. A short history of probability theory and its applications. International Journal of Mathematical Education in Science and Technology. 2015;46(1):13–39.
83. Menezes RX de. Introdução à Probabilidade. Em: Massad E, Menezes RX de, Silveira PSP, Ortega NRS, organizadores. Métodos Quantitativos em Medicina. Barueri, São Paulo: Editora Manole Ltda.; 2004. p. 151–87.
84. Pagano M, Kimberly G. Theoretical Probability Distributions. Em: Principles of Biostatistics. Second Edition. CRC Press; 2000. p. 162.
85. Gonzalez JCS. Normal distribution in R [Internet]. R CODER. 2021. Disponível em: <https://rcoder.com/>
86. Jain S. A Guide to dnorm, pnorm, rnorm, and qnorm in R [Internet]. GeeksforGeeks. 2022. Disponível em: <https://www.geeksforgeeks.org/>
87. Robertson E, O'Connor J. Jacob (Jacques) Bernoulli [Internet]. Maths History. School of Mathematics; Statistics, University of St Andrews; 2022. Disponível em: https://mathshistory.st-andrews.ac.uk/Biographies/Bernoulli_Jacob/
88. Fisher LD, Van Belle G. Poisson Random Variables. Em: Biostatistics: A Methodology for the Health Sciences. New York, NY: John Wiley & Sons; 1993. p. 211–8.
89. Peat J, Barton B. Descriptive statistics. Em: Medical statistics : a guide to SPSS, data analysis, and critical appraisal. New York, NY: John Wiley & Sons; 2014. p. 24–51.
90. George D, Mallory P. Descriptive Statistics. Em: IBM SPSS Statistics 26 Step by Step: A Simple Guide and Reference. New York, NY: Taylor & Francis Group; 2020. p. 114–20.
91. Komsta L. Moments, Cumulants, Skewness, Kurtosis and Related Tests [Internet]. CRAN. 2022. Disponível em: <https://cran.r-project.org/web/packages/>
92. Pagano M, Gavreau K. The Central Limit Theorem. Em: Principles of Biostatistics. Second Edition. Pacific Grove, CA: Duxbury; 2000. p. 197–8.

93. Motulsky H. The Theory of Confidence Intervals. Em: *Intuitive Biostatistics: A Nonmathematical Guide to Statistical Thinking*. Second Edition. New York, NY: Oxford University Press; 2010. p. 96–102.
94. Signorell A et al. DescTools: Tools for Descriptive Statistics [Internet]. 2022. Disponível em: <https://cran.r-project.org/package=DescTools>
95. Kelen GD, Brown CB, Ashton J. Statistical reasoning in clinical trials: hypothesis testing. *Am J Emerg Med.* 1988;1(1):52–61.
96. Menezes RX de, Burattini MN. Testes de Hipótese e intervalos de Confiança. Em: Massad E, Menezes RX de, Silveira PSP, Ortega NRS, organizadores. *Métodos Quantitativos em Medicina*. Barueri, São Paulo: Editora Manole Ltda.; 2004. p. 225–41.
97. Guyatt G, Jaeschke R, Heddle N, et al. Basic statistics for clinicians: 1. Hypothesis testing. *CMAJ: Canadian Medical Association Journal*. 1995;152(1):27.
98. Fletcher RH, Fletcher SW, Fletcher GS. Acaso. Em: *Epidemiologia Clínica: Elementos Essenciais*. Quinta Edição. Artmed Editora; 2014. p. 108–9.
99. Menezes RX de, Burattini MN. Testes de Hipótese e intervalos de Confiança. Em: Massad E, Menezes RX de, Silveira PSP, Ortega NRS, organizadores. *Métodos Quantitativos em Medicina*. Barueri, São Paulo: Editora Manole Ltda.; 2004. p. 225–41.
100. Pagano M, Kimberly G. Comparison of Two Means. Em: *Principles of Biostatistics*. Second Edition. CRC Press; 2000. p. 262–72.
101. Zimmerman DW. A note on preliminary tests of equality of variances. *Br J Math Stat Psychol.* 2004;57(1):173–81.
102. Razali NM, Wah YB, et al. Power comparisons of shapiro-wilk, kolmogorov-smirnov, lilliefors and anderson-darling tests. *Journal of statistical modeling and analytics*. 2011;2(1):21–33.
103. Ghasemi A, Zahediasl S. Normality tests for statistical analysis: a guide for non-statisticians. *International journal of endocrinology and metabolism*. 2012;10(2):486.
104. Yap BW, Sim CH. Comparisons of various types of normality tests. *Journal of Statistical Computation and Simulation*. 2011;81(12):2141–55.
105. Fox J, Weisberg S. *An R Companion to Applied Regression* [Internet]. Third. Thousand Oaks CA: Sage; 2019. Disponível em: <https://socialsciences.mcmaster.ca/jfox/Books/Companion/>
106. Kassambara A. rstatix: Pipe-Friendly Framework for Basic Statistical Tests [Internet]. 2022. Disponível em: <https://CRAN.R-project.org/package=rstatix>
107. Cohen J. *Statistical power analysis for the behavioral sciences*. 2nd Edition. Routledge; 1988.
108. Lindenau JD, Guimaraes LSP. Calculating the Effect Size in SPSS. *Revista HCPA* [Internet]. 2012;32(3):363–81. Disponível em: <https://seer.ufrgs.br/hcpa>
109. Field A, Miles J, Field Z. Comparing several means: ANOVA (GML 1). Em: *Discovering Statistics Using R*. Sage Publications, Ltd; 2012. p. 399–400.
110. Menezes RX de. Análise de Variância. Em: Massad E, Menezes RX de, Silveira PSP, Ortega NRS, organizadores. *Métodos Quantitativos em Medicina*. Barueri, São Paulo: Editora Manole Ltda.; 2004. p. 297–300.
111. Garren ST. Package fastgraph [Internet]. CRAN. Comprehensive R Archive Network (CRAN); 1919. Disponível em: <https://CRAN.R-project.org/package=fastGraph>
112. Peat J, Barton B. Continuous variables: analysis of variance. Em: *Medical statistics : a guide to SPSS, data analysis, and critical appraisal*. New York, NY: John Wiley & Sons; 2014. p. 114.
113. Dag O, Dolgun A, Konar NM. Onewaytests: An R Package for One-Way Tests in Independent Groups Designs. *R Journal*. 2018;10(1):175–99.
114. Ben-Shachar MS, Lüdecke D, Makowski D. effectsize: Estimation of effect size indices and standardized parameters. *Journal of Open Source Software*. 2020;5(56):2815.

115. Watson P. Rules of thumb on magnitudes of effect sizes [Internet]. MRC Cognition and Brain Sciences Unit. Cambridge University; 2021. Disponível em: <https://imaging.mrc-cbu.cam.ac.uk/statswiki/FAQ/effectSize>
116. Field A, Miles J, Field Z. Factorial ANOVA (GLM3). Em: Discovering statistics using R. Sage Publications, Ltd; 2012. p. 513–4.
117. Kassambara A. `ggpubr::ggplot2` based publication ready plots [R package ggpqr version 0.5.0] [Internet]. The Comprehensive R Archive Network. Comprehensive R Archive Network (CRAN); 2022. Disponível em: <https://cloud.r-project.org/web/packages/ggpqr/index.html>
118. Patterson R, Coffman J, Goldstein-Greenwood J, Others. Understanding Diagnostic Plots for Linear Regression Analysis [Internet]. Research Data Services + Sciences. University of Virginia Library; 2015. Disponível em: <https://data.library.virginia.edu/diagnostic-plots/>
119. Wickens TD, Keppel G. Two-way factorial experiments. Em: Design and analysis: A researcher's handbook. Pearson Prentice-Hall; 2004. p. 193–286.
120. Maxwell SE, Delaney HD, Kelley K. Two-way Between-Subject Factorial Designs. Em: Designing experiments and analyzing data: A model comparison perspective. Third Edition. Routledge; 2017. p. 312–82.
121. Lenth R, Singmann H, Love J, Buerkner P, Herve M. Emmeans: Estimated marginal means, aka least-squares means. R package version. 2018;1(1):3.
122. Rosenberg M. Society and the adolescent self-image. Princeton university press; 2015.
123. Dini G, Quaresma M, Ferreira L, et al. Adaptação cultural e validação da versão brasileira da escala de autoestima de Rosenberg. Revista Brasileira de Cirurgia Plástica. 2001;19(1):41–52.
124. Huynh H, Feldt LS. Estimation of the Box correction for degrees of freedom from sample data in randomized block and split-plot designs. Journal of Educational Statistics. 1976;1(1):69–82.
125. Girden ER. Two-Factor Studies with Repeated Measures on Both Factors. Em: ANOVA: repeated measures. Sage; 1992. p. 31–40. (QASS Series; vol. 84).
126. Muller KE, Barton CN. Approximate power for repeated-measures ANOVA lacking sphericity. Journal of the American Statistical Association. 1989;84(406):549–55.
127. Greene Jr JW, Touchstone JC. Urinary estriol as an index of placental function. A study of 279 cases. Obstetrical & Gynecological Survey. 1963;18(3):356–9.
128. Kassambara A. Correlation Test between two variables in R [Internet]. STHDA - Statistical tools for high-throughput data analysis. 2021. Disponível em: <http://www.sthda.com/english/wiki/correlation-test-between-two-variables-in-r>
129. Schober P, Boer C, Schwarte LA. Correlation coefficients: appropriate use and interpretation. Anesthesia & Analgesia. 2018;126(5):1763–8.
130. De Winter JC, Gosling SD, Potter J. Comparing the Pearson and Spearman correlation coefficients across distributions and sample sizes: A tutorial using simulations and empirical data. Psychological methods. 2016;21(3):273.
131. Sedgwick P. Correlation versus linear regression. BMJ. 2013;346.
132. Kim H-Y. Statistical notes for clinical researchers: simple linear regression 3–residual analysis. Restorative dentistry & endodontics. 2019;44(1).
133. Field A, Miles J, Field Z. Regression. Em: Discovering statistics using R. Sage Publications, Ltd; 2012. p. 266–76.
134. Hothorn T, Zeileis A, Farebrother RW, et al. Package “`lmtest`”. Testing linear regression models <https://cran.r-project.org/web/packages/lmtest/lmtest.pdf> Accessed. 2015;6.
135. Peat J, Barton B. Correlation and regression. Em: Medical statistics : a guide to SPSS, data analysis, and critical appraisal. New York, NY: John Wiley & Sons; 2014. p. 209.

136. Altman DG, Gardner MJ. Statistics in Medicine: Calculating confidence intervals for regression and correlation. *British Medical Journal (Clinical research ed)*. 1988;296(6631):1238.
137. Altman DG. Comparing groups: categorical data. Em: *Practical Statistics for Medical Research*. London: Chapman & Hall/CRC; 1991. p. 244–7.
138. Myszkowski N. nhstplot package [Internet]. RDocumentation. 2020. Disponível em: <https://rdrr.io/r/nhstplot/>
139. Warnes GR, Bolker B, et al. Gmodels: Various R programming tools for model fitting [Internet]. CRAN R Project. 2022. Disponível em: <https://rdrr.io/cran/gmodels/>
140. Comtois D. summarytools: Tools to Quickly and Neatly Summarize Data [Internet]. CRAN R Project. 2022. Disponível em: <https://github.com/dcomtois/summarytools>
141. Daniel WW, Cross CL. The chi-square distribution and analysis of frequencies. Em: *Practical Statistics for Medical Research*. Hoboken, NJ: John Wiley & Sons, Inc; 2013. p. 604–19.
142. Eliasziw M, Donner A. Application of the McNemar test to non-independent matched pair data. *Statistics in medicine*. 1991;10(12):1981–91.
143. Rosner B. Hypothesis Testing: Categorical Data. Em: *Fundamentals of Biostatistics*. Seventh Edition. Boston: Cengage; 2011. p. 377.
144. Mayer M. confintr: Confidence Intervals [Internet]. 2022. Disponível em: <https://CRAN.R-project.org/package=confintr>
145. Altman DG. Comparing groups: continuos data. Em: *Practical Statistics for Medical Research*. London: Chapman & Hall/CRC; 1991. p. 194–7.
146. Hothorn T, Hornik K, Van De Wiel MA, Zeileis A. A lego system for conditional inference. *The American Statistician*. 2006;60(3):257–63.
147. Zar JH. Paired-Sample Hypotheses. Em: *Biostatistical Analysis*. Edinburgh: Pearson; 2014. p. 189–98.
148. Karadimitriou SM, Marshall E. Kruskal-Wallis in R [Internet]. Statistics Support for Students. Loughborough; Coventry Universities; 2020. Disponível em: <https://www.statstutor.ac.uk/>
149. Zar JH. Nonparametric Analysis of Variance. Em: *Biostatistical Analysis*. Edinburgh: Pearson; 2014. p. 226–30.
150. Kanji GK. The Kruskal-Wallis test. Em: *100 Statistical Tests*. 3rd Edition. London: Sage publications; 2006. p. 220.
151. Tomczak M, Tomczak E. The need to report effect size estimates revisited. An overview of some recommended measures of effect size. *Trends in sport sciences*. 2014;1(21):19–25.
152. Dunn OJ. Multiple comparisons using rank sums. *Technometrics*. 1964;6(3):241–52.
153. Straus SE, Glasziou P, et al. Diagnosis and screening. Em: *Evidence-Based Medicine: How to Practice and Teach EBM*. Fifth Edition. Edinburgh: Elsevier; 2019. p. 185–218.
154. Altman DG, Bland JM. Diagnostic tests. 1: Sensitivity and specificity. *BMJ: British Medical Journal*. 1994;308(6943):1552.
155. Peixoto R de O, Nunes TA, Gomes CA. Indices diagnósticos da ultrassonografia abdominal na apendicite aguda: influência do gênero e constituição física, tempo evolutivo da doença e experiência do radiologista. *Revista do Colégio Brasileiro de Cirurgiões*. 2011;38:105–11.
156. Mark, Sergeant E, et al. epiR: Tools for the Analysis of Epidemiological Data [Internet]. 2022. Disponível em: <https://CRAN.R-project.org/package=epiR>
157. Altman DG, Bland JM. Diagnostic tests 2: predictive values. *Bmj*. 1994;309(6947):102.
158. Pereira Lima A, Vieira FJ, Oliveira GP de M, et al. Perfil clínico-epidemiológico da apendicite aguda: análise retrospectiva de 638 casos. *Revista do Colegio Brasileiro de Cirurgiões*. 2016;43:248–53.

159. Halkin A, Reichman J, Schwaber M, Paltiel O, Brezis M. Likelihood ratios: getting diagnostic testing into perspective. *QJM: monthly journal of the Association of Physicians*. 1998;91(4):247–58.
160. Deeks JJ, Altman DG. Diagnostic tests 4: likelihood ratios. *Bmj*. 2004;329(7458):168–9.
161. Guyatt G, Rennie D, et al. Diagnostic Tests. Em: User's Guides to Medical Literature: A Manual for Evidence-Based Clinical Practice. 3rd Edition. New York: JAMA; 2015. p. 607–31.
162. Oliveira Filho PF de. Testes Diagnósticos. Em: Epidemiologia e Bioestatística: Fundamentos para a leitura crítica. Segunda Edição. Rio de Janeiro: Editora Rubio; 2022. p. 89–105.
163. Fagan T. Nomogram for Bayes's theorem. *New England Journal of Medicine*. 1975;293:257.
164. Caraguel CG, Vanderstichel R. The two-step Fagan's nomogram: ad hoc interpretation of a diagnostic test result without calculation. *BMJ Evidence-Based Medicine*. 2013;18(4):125–8.
165. Altman DG, Bland JM. Diagnostic tests 3: receiver operating characteristic plots. *BMJ: British Medical Journal*. 1994;309(6948):188.
166. Robin X, Turck N, Hainard A, et al. pROC: an open-source package for R and S+ to analyze and compare ROC curves. *BMC Bioinformatics*. 2011;12(1):1–8.
167. Borges LSR. Diagnostic accuracy measures in cardiovascular research. *Int J Cardiovasc Sci*. 2016;29(3):218–22.
168. DeLong ER, DeLong DM, Clarke-Pearson DL. Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. *Biometrics*. 1988;44:837–45.
169. Youden WJ. Index for rating diagnostic tests. *Cancer*. 1950;3(1):32–5.
170. Cohen J. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*. 1960;20(1):37–46.
171. Landis JR, Koch GG. The measurement of observer agreement for categorical data. *Biometrics*. 1977;159–74.
172. Zeileis A, Meyer D, Hornik K. Residual-based shadings for visualizing (conditional) independence. *Journal of Computational and Graphical Statistics*. 2007;16(3):507–25.
173. Feychtting M, Osterlund B, Ahlbom A. Reduced cancer incidence among the blind. *Epidemiology*. 1998;9(5):490–4.
174. Szklo M, Nieto FJ. Measuring Disease Occurrence. Em: Epidemiology: beyond the basics. Fourth Edition. Burlington, MA: Jones & Bartlett Learning; 2019. p. 80–1.
175. Gross M. Oswego County revisited. *Public health reports*. 1976;91(2):168.
176. Aragon TJ. epitools: Epidemiology Tools [Internet]. 2020. Disponível em: <https://CRAN.R-project.org/package=epitools>
177. Szklo M, Nieto FJ. Measuring Associations Between Exposures and Outcomes. Em: Epidemiology: beyond the basics. Fourth Edition. Burlington, MA: Jones & Bartlett Learning; 2019. p. 88–94.
178. Davies HTO, Crombie IK, Tavakoli M. When can odds ratios mislead? *BMJ*. 1998;316(7136):989–91.
179. Hopkins WG. A Scale of Magnitudes for Effect Statistics [Internet]. A New View of Statistics. Sportscience; 2016. Disponível em: <http://www.sportsci.org/resource/stats/index.html>
180. Madi JM, Souza R da S de, Araujo BF de, Oliveira Filho PF, et al. Prevalence of toxoplasmosis, HIV, syphilis and rubella in a population of puerperal women using Whatman 903® filter paper. *The Brazilian Journal of Infectious Diseases*. 2010;14(1):24–9.
181. Szklo M, Nieto FJ. Measuring Associations Between Exposures and Outcomes. Em: Epidemiology: beyond the basics. Fourth Edition. Burlington, MA: Jones & Bartlett Learning; 2019. p. 84–102.
182. Szklo M, Nieto FJ. Measuring Associations Between Exposures and Outcomes. Em: Epidemiology: beyond the basics. Fourth Edition. Burlington, MA: Jones & Bartlett Learning; 2019. p. 97–8.

183. Kohl M. Package “MKmisc” [Internet]. 2019. Disponível em: <https://github.com/stamats/MKmisc>
184. Bender R. Calculating confidence intervals for the number needed to treat. *Controlled clinical trials*. 2001;22(2):102–10.
185. Clark TG, Bradburn MJ, Love SB, Altman DG. Survival analysis Part I: basic concepts and first analyses. *British Journal of Cancer*. 2003;89(2):232–8.
186. Kaplan EL, Meier P. Nonparametric Estimation from Incomplete Observations. *Journal of the American Statistical Association*. 1958;53(282):457–81.
187. Peat J, Barton B. Survival analyses. Em: *Medical statistics : a guide to SPSS, data analysis, and critical appraisal*. New York, NY: John Wiley & Sons; 2014. p. 352–3.
188. Qiu W, Chavarro J, Lazarus R, Rosner B, Ma J. powerSurvEpi: Power and Sample Size Calculation for Survival Analysis of Epidemiological Studies. R package version 00. 2015;9.
189. Therneau T et al. A package for Survival Analysis in R. R package version. 2015;2(7).
190. Kassambara A, Kosinski M, Biecek P, Fabian S. Survminer: Drawing Survival Curves Using ggplot2. URL <https://CRAN.R-project.org/package=survminer> R package version 04. 2021;9.
191. Bland JM, Altman DG. The logrank test. *BMJ*. 2004;328(7447):1073.
192. Peat J, Barton B. Adjusted odds ratios. Em: *Medical statistics : a guide to SPSS, data analysis, and critical appraisal*. New York, NY: John Wiley & Sons; 2014. p. 298–308.
193. Leisch F, Dimitriadou E. Package mlbench [Internet]. CRAN. Comprehensive R Archive Network (CRAN); 2009. Disponível em: <https://cran.r-project.org/web/packages/mlbench/index.html>
194. Field A, Miles J, Field Z. Logistic regression. Em: *Discovering statistics using R*. Sage Publications, Ltd; 2012. p. 320.
195. Hosmer Jr DW, Lemeshow S, Sturdivant RX. Interpretation of Fitted Logistic Regression Model. Em: *Applied logistic regression*. Third Edition. John Wiley & Sons; 2013. p. 49–64.
196. Ripley B, Venables B, Bates DM, et al. Support Functions and Datasets for Venables and Ripley’s MASS [Internet]. CRAN: The R Project for Statistical Computing. 2023. Disponível em: <https://cran.r-project.org/web/packages/MASS/MASS.pdf>
197. Field A, Miles J, Field Z. Logistic regression. Em: *Discovering statistics using R*. Sage Publications, Ltd; 2012. p. 318.
198. Kabacoff RI. Generalized linear models. Em: *R in Action: Data analysis and graphics with R*. Manning Publications Co.; 2011. p. 317–22.
199. Field A, Miles J, Field Z. Logistic regression. Em: *Discovering statistics using R*. Sage Publications, Ltd; 2012. p. 344–5.
200. Mansfield ER, Helms BP. Detecting multicollinearity. *The American Statistician*. 1982;36(3a):158–60.
201. McFadden D. Quantitative methods for analysing travel behaviour of individuals: some recent developments. Em: *Behavioural travel modelling*. Routledge; 2021. p. 279–318.
202. Hoaglin DC, Welsch RE. The hat matrix in regression and ANOVA. *The American Statistician*. 1978;32(1):17–22.
203. Stevens JP. Outliers and influential data points in regression analysis. *Psychological bulletin*. 1984;95(2):334.
204. Fawcett T. An introduction to ROC analysis. *Pattern recognition letters*. 2006;27(8):861–74.