



Aprendizagem estatística para ciências.

Aplicações com software.

Felipe Coelho Argolo

Página intencionalmente deixada em branco.

Prefácio

Remember that all models are wrong; the practical question is how wrong do they have to be to not be useful
George Box & Norman R. Draper, *Empirical Model-Building and Response Surfaces*

Uma antiga aplicação da matemática é fazer inferências com base em observações de cenários parecidos. Civilizações antigas, como os babilônios, usavam interpolação linear para estimar informações. Fazendo o censo populacional com intervalo de anos, estimavam o valor dos anos não medidos, supondo que eles eram medidas centrais daquelas que ao seu redor. Métodos iterativos também foram usados para aproximar a raiz quadrada de números naturais ($\sqrt{2}$) e números irracionais π .

Essas técnicas deram fruto a abstrações mais gerais, aos campos da estatística e dos métodos numéricos. Em particular, o último século contou com a invenção do computador universal e dos processadores eletrônicos, impulsionando o poder de cálculos vertiginosamente.

O aperfeiçoamento teórico e instrumental trouxe ferramentas mais adequadas para cientistas e também algoritmos mais potentes para aplicações práticas.

Nos últimos anos, o campo ganhou forte notoriedade social e acadêmica em virtude dos resultados inéditos em problemas de predição com aplicação prática. Avanços em processamento de linguagem natural, visão computacional e algoritmos preditivos foram rapidamente aplicados pela indústria e por pesquisadores.

Uma descrição abrangente pode facilmente alcançar 1,000 páginas de texto sucinto, como o clássico ‘Deep Learning (Adaptive Computation and Machine Learning)’ de Goodfellow, Bengio and Courville. Outra obra de escopo e tamanho semelhante é a “Neural networks and learning machines”, de Simon Haykin.

Objetivos

Este texto oferece uma introdução intuitiva ao campo, contextualizando-o epistemologicamente. O campo de aprendizagem estatística tem definição pouco estabelecida. Abrange aspectos de matemática pura e aplicada. Com uma perspectiva mais geral, a matemática pura desenvolve abstrações básicas, descrevendo o comportamento de números, probabilidades, funções e outras entidades. Veremos que progressos fundamentais foram feitos por nomes como De Moivre, Euler e Gauss.

Em matemática aplicada, especialistas estudam a relação dessas abstrações com fenômenos observáveis. Estas pessoas empregam métodos quantitativos a contextos restritos: por exemplo, James Clerk Maxwell deduziu (1860) a distribuição estatística e velocidade de partículas em um gás ideal, conhecida como distribuição de Maxwell–Boltzmann. Em estatística, veremos a descoberta da distribuição t para as estimativas de uma média por William Gosset.

São exemplos de campos que fazem uso extenso das ferramentas descritas: neurociências (modelos lineares em fmri), psicometria (análise fatorial), ecologia, biologia molecular (testes estatísticos), ciências clínicas (meta-análises e inferência causal), economia, marketing, algotrading.

Este texto introduz e fornece um guia para aplicações práticas destas ferramentas a fenômenos observáveis. É destinados aos profissionais e pesquisadores trabalhando na fronteira entre matemática aplicada e ciências naturais.

O primeiro capítulo ilustra como o racional hipotético-dedutivo funciona para estudar teorias científicas. Aborda a relação entre ciências empíricas e três abstrações matemáticas: a distribuição normal, a distribuição t e o teorema do limite central. O segundo capítulo aborda correlações e modelos preditivos lineares. Um framework frequencista e linguagem R são usados para demonstrações de exemplos e exercícios.

O terceiro capítulo apresenta um racional diferente para os procedimentos. Usando o conceito de holismo epistemológico (van Quine), reproduzidos as análises anteriores usando inferência bayesiana. Fazemos perguntas diferentes para obter informações de nossos dados. No capítulo quatro, o foco está em modelos classificatórios e na função logística. Usamos R, Stan e um framework bayesiano para modelos simples e hierárquicos. Exploramos o poder das simulações através de Markov Chain Monte Carlo para obter estimativas

difíceis de tratar analiticamente.

O quarto capítulo ilustra o uso de grafos/redes para a construção de modelos preditivos. Os exemplos são de Support Vector Machine e Redes Neurais. Modelos são construídos do zero (from scratch) para ilustrar dois mecanismos importantes de otimização (gradient descent e back propagation).

Sumário

Capítulo 0 - Ferramentas: programação com estatística básica

- Computadores
- R : Curso rápido
 - Instalação, R e Rstudio
 - Tipos
 - Operadores úteis: <- , %>%
 - Funções
 - Vetores, loops e recursões
 - Matrizes e dataframes
 - Gramática dos gráficos e ggplot

Capítulo 1 - Os pássaros de Darwin e o método hipotético dedutivo

- Teorema do limite central e Distribuição normal
- Distribuição t
- Método hipotético-dedutivo e Testes de hipótese
- Valor p

Em construção:

Capítulo 2 - Expectativa de vida e assistência em saúde

- Relações lineares
- Coeficiente de correlação r de Pearson
- Regressão linear

Capítulo 3 - Contexto e inferência Bayesiana

- Intuições sobre distribuições probabilísticas
- Inferência Bayesiana para teste de diferenças e correlação linear
- Classificação
 - Regressão logística
 - Modelos hierárquicos
- Flexibilidade Bayesiana
 - Usando priors
 - O estimador Markov Chain Monte Carlo

Capítulo 4 - Redes neurais

- Support Vector Machines
- Gradient Descending
- Redes Neurais
 - Backpropagation
 - Deep learning (múltiplas camadas)

Capítulo 5 - Programação probabilística para contextos gerais

- Inferência Bayesiana para cosmologia
- Prevendo halos de matéria escura (Kaggle top solution)
- Redes neurais probabilísticas com PyMC3

Capítulo 6 - Ambientes desconhecidos

- Aprendizagem não supervisionada
- Redução de dimensões
- Clustering
- Aprendizagem semi-supervisionada
- Reinforcement learning

Pré-requisitos

Para uma leitura fluida do texto, recomenda-se a compreensão de rudimentos em probabilidade, estatística e cálculo (análise real). Os exemplos com ferramentas computacionais (exceto gráficos) usam sintaxe semelhante à matemática apresentada no texto. Assim, baixa familiaridade com linguagens de programação não é uma barreira.

Todos os exemplos podem ser reproduzidos usando software livre.

Leitura recomendada:

Neurociências

- Principles of neural science - Eric Kandel

Matemática pura e programação

- Better Explained (<https://betterexplained.com/>)
- What is mathematics - Courant & Robbins
- Fundamentos da matemática elementar - Iezzi (Vol. 5)
- MOOCs sobre estatística básica usando R (e.g.: <https://www.coursera.org/specializations/statistics>)
- Cálculo Diferencial e Integral - Piskounov =)
- <http://material.curso-r.com/>
- R Graphics Cookbook
- R Inferno
- Learn you a Haskell for Great Good
- Layered Grammar of Graphic - Hadley Wickham.
- The art of computer programming
- Algorithms unlocked
- Portais: statsexchange, stackoverflow, mathexchange, cross-validated.

Machine Learning

- An Introduction to Statistical Learning: with Applications in R
- Neural Networks and Learning Machines - Simon Haykin
- Stanford course on computer vision: <http://cs231n.stanford.edu/>
- Deep learning at Oxford 2015: (<https://www.youtube.com/watch?v=dV80NA1Eins&list=PLE6Wd9FR--EfW8dtjAuPoTuPcqmOV53Fu>)



Capítulo 0

Ferramentas

Programação com estatística básica

Master Foo and the Shell Tools¹

Um aprendiz do caminho Unix veio ao Mestre Foo e disse: “Estou confuso. Não é o caminho Unix que cada programa deve se concentrar em uma coisa e fazê-la bem?

Mestre Foo assentiu.

O aprendiz continuou: “Também não é do caminho Unix que a roda não deve ser reinventada? Mestre Foo assentiu novamente.

“Então, por que existem diversas ferramentas com capacidades similares em processamento de texto: sed, awk e Perl? Com qual delas posso praticar melhor o caminho Unix?”

Mestre Foo perguntou ao aprendiz: “Se você tem um arquivo de texto, qual ferramenta usaria para produzir uma cópia com algumas palavras trocadas por uma string de sua escolha?”

O aprendiz torceu o nariz e disse: “As expressões regulares de Perl seriam um excesso para tarefa tão simples. Eu não conheço awk, e venho escrevendo scripts sed nas últimas semanas. Como tenho experiência com sed, eu preferiria ele no momento. Mas se o trabalho precisa ser feito apenas uma vez, um editor de textos funcionaria.”

Mestre Foo assentiu e respondeu: “Quando você estiver com fome, coma; quando estiver com sede, beba; quando estiver cansado, durma.”

E, ao ouvir isso, o aprendiz foi iluminado.

¹<http://catb.org/esr/writings/unix-koans/shell-tools.html>

Computadores

Ao longo do texto, usaremos exemplos com software. Computadores são úteis para acelerar os cálculos necessárias para nossos objetivos.

Há milênios, o homem usa instrumentos, como ábacos e tabelas, para fazer operações extensas e precisas envolvendo grandes números. Dado um problema ou dado a ser computado, esses instrumentos mecanismos automatizam partes do processo devido à maneira como foram construídos. A principal diferença destas ferramentas para os computadores de hoje é que nossas máquinas podem ser programadas para fazer computações arbitrárias.

Ada Lovelace (*10 December 1815 – 27 November 1852*) foi a primeira a descobrir essa possibilidade. Estudando a Máquina Analítica de Charles Babbage, Ada concebeu uma maneira de realizar computações para as quais a máquina não havia sido desenhada originalmente. O programa concebido calculava os Números de Bernoulli. Discutivelmente, alterar a estrutura de máquinas mais simples também consiste em reprogramá-las.

Máquinas desse tempo pesavam toneladas e eram muito mais lentas. O avançar dos anos tornou a tecnologia mais acessível, ao ponto de possibilitar computadores pessoais de alta potência e baixo-custo. Além disso, ao invés de operações mecânicas complexas, podemos usar linguagens de programação que traduzem comandos baseados no inglês para instruções de máquina. Os programas aqui apresentados são escritos em R, Stan e Python. As três têm código aberto, podendo ser obtidas, instaladas e usadas sem pagamentos. Sendo um texto didático, as implementações com software priorizam legibilidade. Os três frameworks usam libs em C/C++/Fortran para otimizar computações e interface com GPU (graphic processing unit).

R será mais usada. É uma linguagem interpretada voltada à computação estatística, possuindo ferramentas úteis em sua biblioteca de base. Entre estas, funções para gerar e manipular distribuições probabilísticas. Sendo uma linguagem de ‘alto nível’, não temos sobrecarga cognitiva no programador com manejo de memória e hardware no código. A abstração de detalhes físicos, como registradores da CPU, são feitas automaticamente pelo interpretador. O ecossistema para visualização de dados possui poder e flexibilidade. A comunidade R cresce rápido e fluência nessa linguagem dá acesso a ferramentas muito diversas com bases grandes de suporte. Há suporte para estilo funcional e orientado a objetos.

Stan é uma linguagem/plataforma de domínio específico bastante popular entre estatísticos bayesianos. Possui ferramentas poderosas (e.g: Variational inference, MCMC com NUTS e HMC) para lidar com distribuições probabilísticas e inferência nesse contexto.

Python é uma linguagem de propósito geral. Bastante popular e dotada de uma base de usuários imensa. Linguagem de primeira escolha como interface de alto nível (wrapper) para a maioria das tecnologias de aplicação industrial (e.g: PyTorch, Pyro, Tensorflow). Com dois “dialetos” incompatíveis (2.X e 3.X) diferindo em mínimos detalhes, possui também uma variedade de opções que pode confundir iniciantes (e.g: pip vs. conda). Vamos precisar de Python (PyMC/Pyro) para combinar inferência Bayesiana e redes neurais.

R

Curso rápido

Programas de computador são importantes ao longo dos próximos capítulos para realizar cálculos, gerar dados e visualizações.

Felizmente, os programas que escreveremos são simples, de forma que não precisamos conhecer todos os recursos e características da linguagem R. Neste capítulo, entenderemos os instrumentos básicos para caminharmos.

Veremos diversas maneiras de escrever um programa para calcular a variância σ^2 de um conjunto de medidas.

Instalação

R

Instruções para download e instalação podem ser encontradas em:

<https://cloud.r-project.org/>

Em Windows, o processo costuma consistir em clicar no executável de instalação e concordar com os prompts.

Para Linux, envolve adicionar o CRAN à lista de repositórios e baixar o pacote *r-base* ou o código-fonte/tarball diretamente do website. Há inúmeros tutoriais explicando a instalação.

Rstudio

Com o R instalado, recomendo o uso do ambiente de desenvolvimento RStudio (<https://www.rstudio.com/>) para obter algumas facilidades. Entre elas: atalhos *vim*, editor com highlight de sintaxe, autocompletar, renderização em tempo real de animações e plots, visualização de datasets, ambiente de desenvolvimento, logs, suporte a markup languages, como Markdown, RMarkdown e Latex.²

Tipos

Primeiro, vamos conhecer as entidades básicas do R. Lidamos rotineiramente com vetores, que são células contíguas contendo dados. Os dados podem ser de tipos: lógico (verdadeiro/falso), caracteres, números inteiros, reais e complexos:

“logical”: a vector containing logical values (TRUE/FALSE) “integer”: a vector containing integer values (1,2,3,4...,23,26...)

“double”: a vector containing real values (3.14...)

“complex”: a vector containing complex values (2 +2i)

“character”: a vector containing character values (“string”)

Para saber o tipo de um objeto em R, use `typeof(objeto)`. Podemos acessar elementos de um vetor pelo seu índice, independente do tipo. Declaramos dois vetores, character e double.

```
>a <- c("banana", "terracota", "pie")
>b <- c(2.2, 4.4, 5.5)
> typeof(a)
[1] "character"
> typeof(b)
[1] "double"
```

²Este texto é escrito em Markdown e o código-fonte pode ser encontrado em <https://github.com/fargolo/stat-learn>

A função *combine*: `c(arg1,arg1,...)` combina argumentos em um vetor. Para nossas aplicações, vamos usar números reais (*double*) na maioria dos casos. Os tipos *integer*, *double* e *complex* fazem parte da classe dos números (*numeric*)

```
> class(b)
[1] "numeric"
```

Operadores

Além dos operadores clássicos (+,-,/,-, ...), usamos constantemente dois operadores pouco comuns: O operador “`<-`” atribui o valor da expressão à sua direita ao objeto à sua esquerda. É preferível ao operador “`=`” para evitar confusão ao passar argumentos de funções e fazer comparações lógicas.

```
> a <- 3
> a
[1] 3
```

O operador “`%>%`” da biblioteca *magrittr* fornece o resultado da expressão à sua esquerda como argumento para a expressão à sua direita. Evita aninhamento de expressões, tornando fluxos de computações mais legíveis.

As expressões a seguir são equivalentes.

```
> library(magrittr)
> result <- 3 %>% exp %>% exp
> result
[1] 528491311
> result == exp(exp(3))
[1] TRUE
```

Onde $\exp(a) = e^a$, $e \sim 2.72\dots$. A expressão “`3 %>% exp %>% exp`” equivale a “ $\exp(\exp(3))$ ”. Usando parênteses, partimos da última computação. Usando o pipe (`%>%`), começamos com a primeira operação.

Notem que para usar um recurso da biblioteca *magrittr*, carregamos ela usando o comando *library(magrittr)*. Para instalar uma biblioteca do repositório oficial (CRAN), usamos o comando *install.packages(magrittr)*.

Matrizes e data frames

R possui estruturas que ajudam a manipulação de dados estruturados como os que vemos comumente em ciências.

A mais simples é a lista. Uma lista é um conjunto de objeto de quaisquer tipos. Assim, podemos ter uma lista contendo vetores, doubles, matrizes e gráficos! Tudo em uma estrutura.

```
> mlist <- list(a = c(1,5,6,7), b = c("a","b","c","d"))
> mlist
$a
[1] 1 5 6 7
$b
[1] "a" "b" "c" "d"
> class(mlist)
[1] "list"
Podemos acessar estruturas internas pelo nome usando o operador $:
> typeof(mlist$a)
[1] "double"
> typeof(mlist$b)
[1] "character"
```

Outro tipo útil é composto pelas matrizes, que correspondem às matrizes da matemática, podendo também conter caracteres em suas células.

```
>matrix(data=c(mlist$a, mlist$b), ncol=2)
 [,1] [,2]
 [1,] "1"  "a"
 [2,] "5"  "b"
 [3,] "6"  "c"
 [4,] "7"  "d"
```

Podemos conduzir multiplicação de matrizes facilmente.

```
>mat_example <- matrix(c(.5,.25,.25,.5,0,.5,.25,.25,.5), nrow=3, byrow=TRUE)
> mat_example
 [,1] [,2] [,3]
 [1,] 0.50 0.25 0.25
 [2,] 0.50 0.00 0.50
 [3,] 0.25 0.25 0.50
> mat_example %*% c(1,0,1)
 [,1]
 [1,] 0.75
 [2,] 1.00
 [3,] 0.75
```

Por fim, data.frames são extensões das matrizes:

```
> mat_example %>% data.frame
      X1    X2    X3
1 0.50 0.25 0.25
2 0.50 0.00 0.50
3 0.25 0.25 0.50
```

Data frames são os objetos mais comumente tratados em R e seguem o formato tidy.

Cada variável corresponde a uma coluna.

Cada observação corresponde a uma linha.

Cada tipo de unidade observacional forma uma tabela.

Um exemplo visual torna as coisas mais fáceis. A seguir, temos uma variável categórica (País) e duas numéricas (Número de médicos por 1.000 habitantes em 2011 e Expectativa de vida ao nascer) em formato tidy:

Note que cada linha corresponde a apenas um país (observação) e cada coluna representa uma variável. Se queremos ver a observação 9, vamos à linha correspondente e podemos encontrar os valores: “Armenia” (País), “2.845” (Médicos/1.000 hab. em 2011) e “71” (Expectativa de vida ao nascer).

Para acessar o valor correspondente, usamos índices separados por vírgula. O primeiro espaço é reservado às linhas selecionadas e deve ser um vetor de números (linhas selecionadas) ou vetor com valores lógico do tamanho do dataset (valores com índices TRUE serão incluídos). O segundo espaço corresponde às colunas e deve conter índices numéricos ou nomes das variáveis.

```
>iris[1:5,c("Species",'Sepal.Length')] # primeiras 5 linhas com variaveis especies e sepal.length'
  Species Sepal.Length
1  setosa      5.1
2  setosa      4.9
3  setosa      4.7
4  setosa      4.6
5  setosa      5.0
```

Country	Doctors 2011	Life Expectancy at Birth
Aruba	NA	NA
Andorra	NA	83
Afghanistan	0.23400000	61
Angola	NA	52
Albania	1.11300000	74
Arab World	1.52685042	NA
United Arab Emirates	NA	77
Argentina	NA	76
Armenia	2.84500000	71

Figure 1: País, Número de médicos a cada 1000 habitantes em 2011 e Expectativa de vida ao nascer. “NA” corresponde a dados faltantes no R. Layout do RStudio Fonte: WHO

Gramática dos gráficos e ggplot

Uma das ferramentas de destaque no ecossistema R é a ggplot. Ela provê uma sintaxe bastante poderosa e flexível para plotar visualizações. O segredo está em seu design, que utiliza gramática de gráficos (**Grammar of GraphicsPlot**).

Bertin³ delineou essa abordagem, que consiste em mapear características dos dados a elementos visuais seguindo uma sintaxe consistente. A lib ggplot implementa uma gramática em camadas, possibilitando superposições para gráficos complexos.

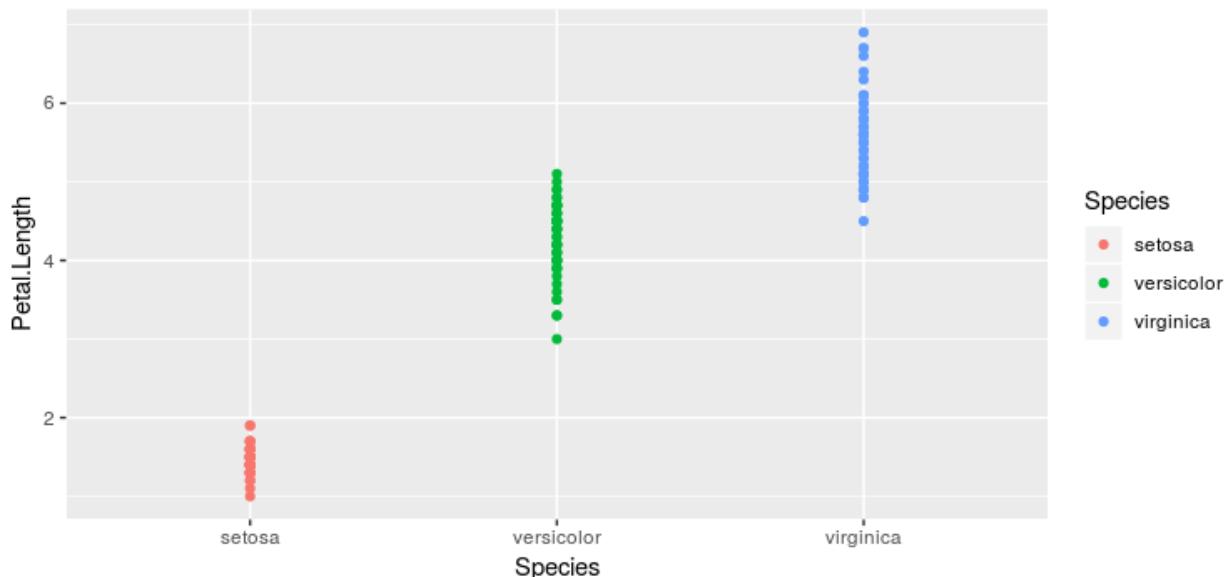
```
>head(sleep)
extra group ID
1 0.7 1 1
2 -1.6 1 2
3 -0.2 1 3
```

Para usarmos o ggplot, podemos declarar (1) o dataframe usado, (2) a relação entre medidas e parâmetros estéticos e (3) objetos geométricos. Parâmetros opcionais podem ser usados, aumentando o número de camadas ou criando transformações.

Assim, podemos plotar um histograma das medidas dos dois grupos com (1) dataset iris; (2) dimensão y: tamanho da pétala, cores:espécie, dimensão x: espécie; e (3) objeto geométrico: ponto.

Assim, teremos pontos com a altura (dimensão y) correspondente à medida da pétala e separados ao longo do eixo x por espécies. O ggplot automaticamente discretiza o eixo x.

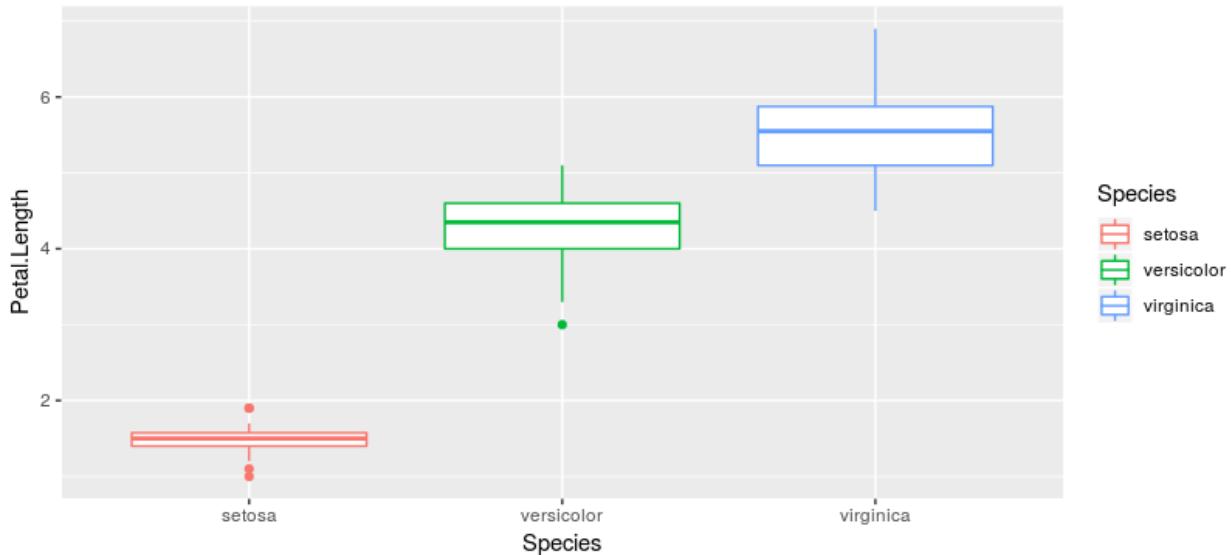
```
>library(ggplot2)
>ggplot(data=iris,aes(y=Petal.Length,x=Species,color=Species))+
  geom_point()
```



Para ilustrar a flexibilidade da biblioteca, note que mudando apenas o objeto geométrico (geom), obtemos um gráfico diferente, mantendo dados e relações (mappings) iguais :

```
>ggplot(data=iris,aes(y=Petal.Length,x=Species,color=Species))+
  geom_boxplot()
```

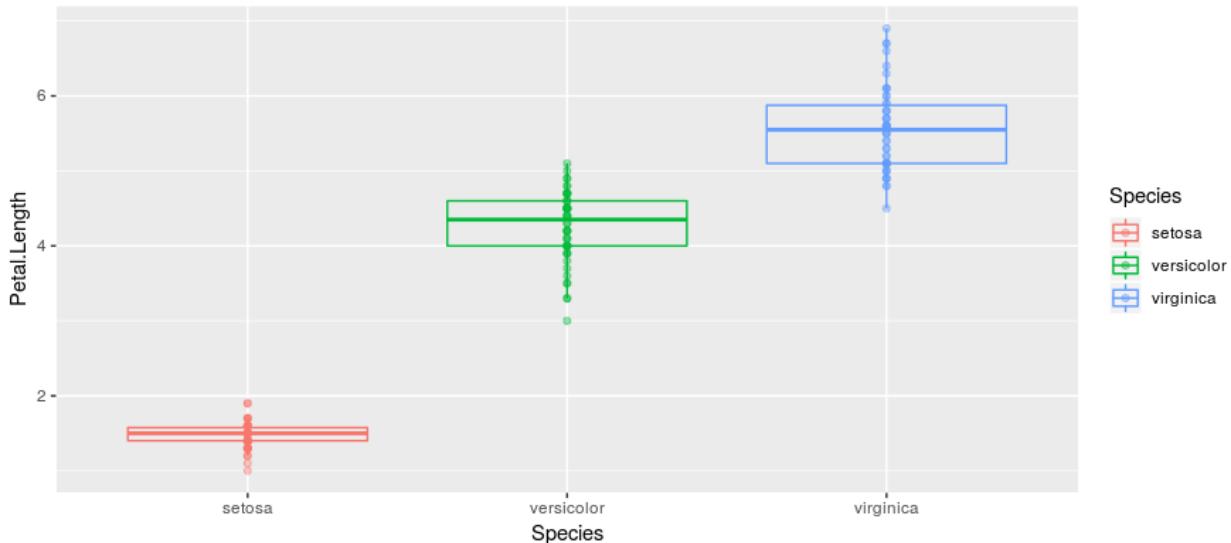
³Bertin, J. (1983),*Semiology of Graphics*, Madison, WI: University of Wisconsin Press



As figuras acima são conhecidas como boxplots. O centro correspondente à mediana (percentil 50), as bordas correspondem aos percentis 25 (inferior) e 75 (superior). Os fios, conhecidos como “bigodes”, estendem-se até $1,5^* \text{ IQR}$ (onde $\text{IQR} = \text{Percentil 75} - \text{Percentil 25}$).

É possível adicionar camadas e estas podem sobreescrivere informação de camadas anteriores. Isso torna a sintaxe do ggplot altamente modular. A seguir, superpomos pontos e boxplot:

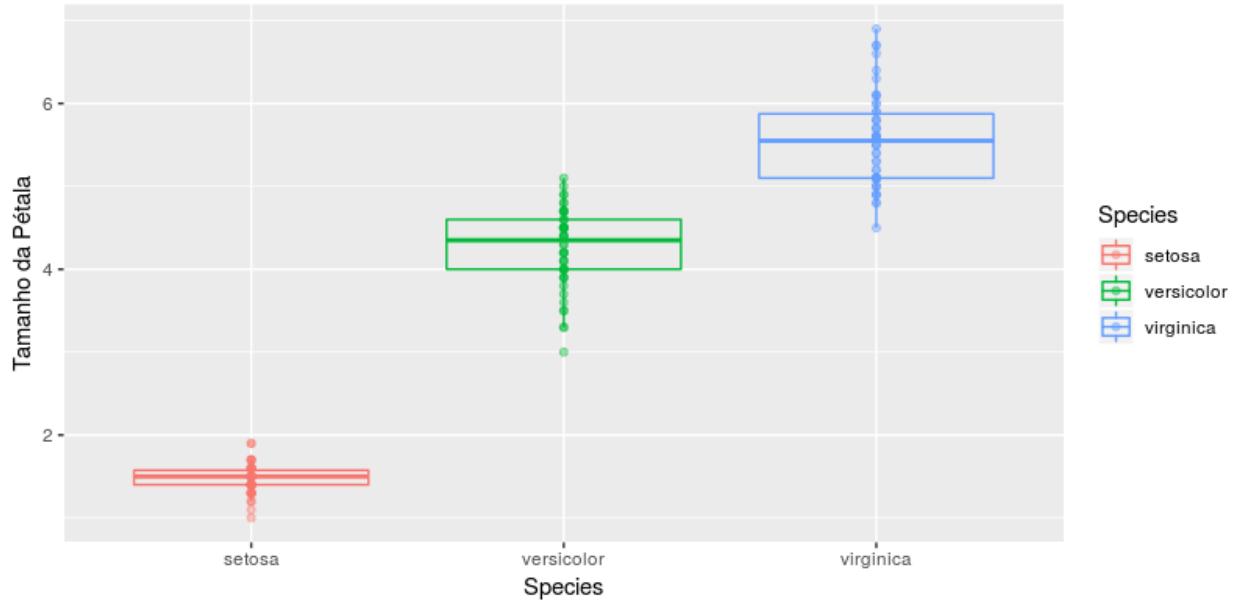
```
>ggplot(data=iris,aes(y=Petal.Length,x=Species,color=Species))+  
  geom_point(alpha=0.4)+ # camada 1  
  geom_boxplot(alpha=0) # camada 2
```



O parâmetro *alpha* regula a transparência dos objetos. Colocamos os boxplot com transparência total (*alpha*=0), dando visibilidade aos pontos (*alpha*=0.4). Adicionamos algum grau de transparência para que pontos superpostos sejam mais escuros que pontos individuais. Adicionaremos uma terceira camada, que substitui o rótulo do eixo y para uma legenda em português:

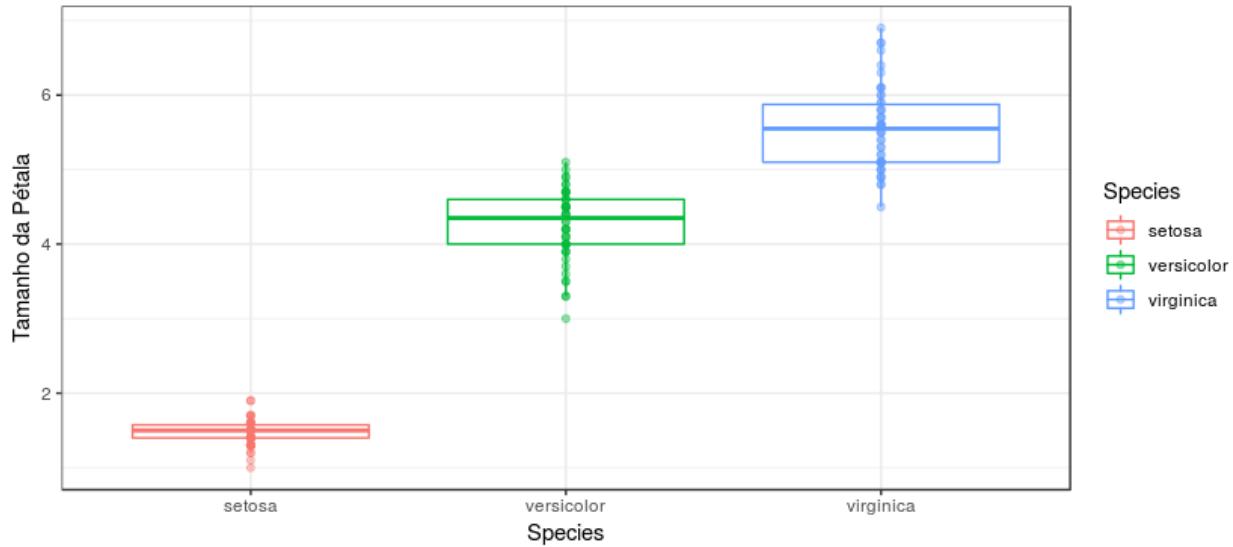
```
>ggplot(data=iris,aes(y=Petal.Length,x=Species,color=Species))+  
  geom_point(alpha=0.4)+ # camada 1  
  geom_boxplot(alpha=0)+ # camada 2
```

```
ylab("Tamanho da Pétala") # camada 3
```

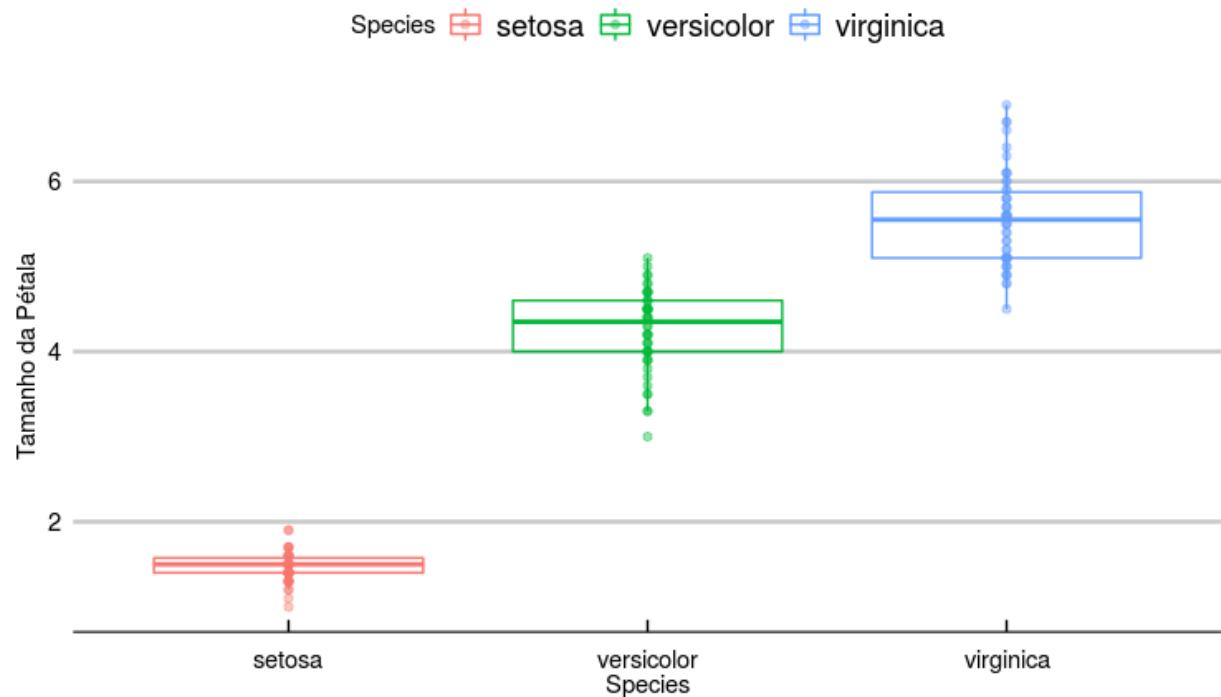


Ainda, existem temas prontos para mudar o estilo geral da imagem:

```
>ggplot(data=iris,aes(y=Petal.Length,x=Species,color=Species))+  
  geom_point(alpha=0.4)+ # camada 1  
  geom_boxplot(alpha=0)+ # camada 2  
  ylab("Tamanho da Pétala") # camada 3  
  theme_bw() # camada 4: tema
```



```
>ggplot(data=iris,aes(y=Petal.Length,x=Species,color=Species))+  
  geom_point(alpha=0.4)+ # camada 1  
  geom_boxplot(alpha=0)+ # camada 2  
  ylab("Tamanho da Pétala") # camada 3  
  theme_economist_white(gray_bg = F) # camada 4: tema
```



Funções

Uma das formas de escrever programas é através de funções.

Podemos declarar funções que (1) aceitam argumentos de entrada, (2) executam computações com esses argumentos e (3) devolvem resultados na saída.

Assim, podemos criar a função soma2, que recebe dois argumentos numéricos e retorna a soma de ambos.

```
>soma2 <- function(argumento1,argumento2){  
  return(argumento1+argumento2)  
}
```

Ao invocarmos soma2 com os argumentos 2 e 5, recebemos soma2(argumento1=2, argumento2=5) = 2+5 = 7.

```
>soma2(argumento1=2,argumento2=5)  
[1] 7
```

Podemos omitir o nome dos argumentos. Assim os objetos são passados na ordem de entrada.

```
>soma2(2,3)  
[1] 5
```

Por padrão, o valor retornado é mostrado no console.

R aceita em sua sintaxe que uma função seja argumento de outra numa mesma instrução:

```
>soma2(2, soma2(3,2) )  
[1] 7
```

A expressão acima é equivalente a $(2 + (3 + 2)) = 7$.

Podemos definir a função de média para um vetor de números, dado pela (1) soma dividida pelo (2) tamanho do vetor:

```

>mean_vec <- function(x){
  sum(x)/length(x)
}
>mean_vec(b) # Anteriormente definido por b <- c(2.2, 4.4, 5.5)
[1] 4.033333

```

`sum(x)` retorna a soma de todos os elementos do vetor `x`. `length(x)` retorna o tamanho (número de células) do vetor `x`.

A média é uma medida de tendência central para um conjunto de observações. É o ponto mais perto de todos os outros.

Muitas formas de calcular a variância

Também podemos calcular uma medida relacionada ao quanto nossos valores se afastam do centro.

Primeiro, calculamos uma distância entre cada elemento x e a média das observações μ . A noção de distância implica que ela deve ser um valor positivo. Supondo que x e μ são medidas num espaço ordenado, podemos usar o módulo da diferença entre os valores: $\|x - \mu\|$. Ainda, podemos usar o quadrado da diferença: $d_i = (x_i - \mu)^2$.

A variância σ^2 das observações é uma medida da dispersão de toda a amostra.

Para calcular σ^2 , somamos todas as distâncias d_i e dividimos o resultado por $n - 1$.

```

>var_2 <- function(x) sum((x - mean(x))^2) / (length(x) - 1)
>var_2 (b)
[1] 2.823333

```

Sendo proporcional às distâncias dos valores em relação à média, a variância σ^2 tende a ser maior quando os valores são muito distintos entre si:

```

>c <- c(100,200,1,45,-24)
>var_2(c)
[1] 7966.3

```

Outra medida de dispersão, dada nas unidades originais da medida observada, é o desvio-padrão σ , dado pela raiz da variância σ^2 .

```

>var_2(b) %>% sqrt
[1] 1.680278

```

O R possui funções embutidas para muitas aplicações estatísticas: `sd` (desvio-padrão), `var` (variância), `mean` (média)... Em especial, temos funções prontas para trabalhar com diversas distribuições probabilísticas de variáveis aleatórias. Para sortear 10 números de uma distribuição normal:

```

>rnorm(n=10, mean=0, sd=1)
[1] 0.2874490 0.2931469 3.1897423 1.7445002 3.3998010 -0.1482911
[7] 2.0257046 -0.6002109 -0.2840376 -0.7715565

```

Distribuição gamma.

```

>rgamma(n=10, shape=1)
[1] 1.1183441 1.2770135 1.0972053 1.4820536 2.3542620 0.8231831 0.5535210
[8] 5.0481559 0.2853060 0.1623315

```

Exponencial:

```

>rexp(n=10, rate = 1)
[1] 0.31657586 0.26676766 0.02288276 0.92801416 0.44006133 0.05238540
[7] 1.10213153 0.91931786 2.58807134 0.41825081

```

Vetores, loops e recursões

Anteriormente, definimos a função para calcular variância como:

```
>var_2 <- function(x) sum((x - mean(x))^2) / (length(x) - 1)
```

Isso só é possível porque o R aplica funções a vetores de maneira automática. Assim, a expressão $(x - mean(x))^2$ subtrai a média de cada elemento do vetor x.

Normalmente, é necessário usar estruturas recursivas para isso. O laço for (for loop) define uma sequência de tamanho n definido e repete um bloco de comandos n vezes. Se queremos imprimir números entre 1 e 10:

```
>for (i in 1:10) print(i)
[1] 1
[1] 2
[1] 3
[1] 4
[1] 5
[1] 6
[1] 7
[1] 8
[1] 9
[1] 10
```

A instrução avalia print(i) para valores i=1,2,3..,10 de forma repetida.

Vamos reescrever nossa função para calcular variância σ^2 usando um loop. Podemos definir um loop com o tamanho do vetor x e calcular o quadrado da diferença em cada elemento.

Assim,:;

```
var_3 <- function(x){
  accumulator <- numeric() #armazena distâncias
  for (i in 1:length(x)) # loop começa em 1 segue até o tamanho do vetor
    accumulator[i] <- (x[i] - mean(x))^2 # calcula e armazena distâncias.
  return (sum(accumulator) / (length(x) - 1)) #calcula media
}
```

Ambas definições apresentam o mesmo resultado que a implementação nativa do R:

```
> var(b)
[1] 2.823333
> var_2(b)
[1] 2.823333
> var_3(b)
[1] 2.823333
```

Ainda, uma maneira de manipular muitos elementos é através de funções de alta ordem. Estas funções recebem outras funções como argumentos. Um exemplo é a função map da lib purrr. Definimos uma função para a distância, $f(y) = (y - \mu)^2$, e aplicamos em todos os elementos. Só então, somamos os resultados e dividimos por n-1.

Tudo pode ser feito em apenas um pipe:

```
>map(.f = function(y) (y - mean(arg))^2, .x = arg) %>%
  unlist(.) %>% sum(.) / (length(arg) - 1)
```

Quando usamos o pipe, o caractere de ponto (.) se refere ao valor fornecido como entrada pela pipe anterior. Assim, sum(.), no exemplo, acima soma os valores passados pela função unlist(.), que por sua vez, transforma em vetor uma lista de valores passada pela função map. Nossa função pode ser escrita:

```

var_4 <- function(arg){
  purrr::map(.f = function(y) (y - mean(arg))^2, .x = arg) %>% # Define e aplica função
  unlist %>% sum(.) / (length(arg) - 1) # Soma as distâncias e divide por n-1
}
> var_4(b)
[1] 2.823333

```

Exercícios

1. Qual a diferença entre linguagens compiladas e interpretadas?
2. Um programa escrito em R pode ser escrito em qualquer outra linguagem. Esta afirmação é verdadeira? Por quê?
3. Cite 3 recursos que uma IDE fornece ao programador.
4. Modifique o tema de fundo do RStudio para um de cor escura (menos luz para os olhos :)).
5. Usando o operador `<-`, produza:
 - Um vetor com componentes do tipo logical
 - Dois vetores de 5 elementos do tipo double
 - A soma dos elementos nos vetores do item b.
 - A divisão entre elementos dos vetores do item b.
 - Aplique as funções sd, mean e var em amostras normais aleatórias de $n = 10, 30, 100$ e 300 . A função rnorm ($n, \text{mean}, \text{sd}$) pode ajudar. Compare os valores da distribuição de origem com os obtidos.
7. *UnLISP it!* Transforme as seguintes expressões, substituindo parênteses aninhados pelo operador pipe (%>%) quando julgar conveniente:
 - `round (mean (c(10 , 2, 3))`
 - `round (mean (rnorm (n = ceiling (runif (1,0,10))))`
 - `paste("a",seq(1:rnorm(n=mean(c(3,2,1,16)))))`
 - `round(nrow(iris) + exp(1), digits = ceiling(runif(1,0,10)))`
8. Usando o código das funções var_2 (vetorizado), var_3 (for loop) e var_4 (função de alta ordem map)
 - Escreva as funções correspondentes (sd_2, sd_3, sd_4) para desvio-padrão e compare com a função padrão do R (sd). Dica: Basta aplicar raiz quadrada ao valor final retornado anteriormente!
9. Usando o dataset iris
 - Selecione apenas os exemplos com tamanho de pétala maior que 4.
 - Selecione os 10 maiores exemplares. Suponha que o tamanho é dado pela média das 4 medidas fornecidas.
 - Calcule a média e o desvio-padrão para duas medidas em cada espécie.
 - Faça um scatterplot entre duas medidas
 - Adicione cores de acordo com a espécie

- Adicione o rótulo de texto a um dos pontos
- Mude títulos (principal, eixos x e y, legenda)
- Mude o tema de fundo. Dica: experimente os temas da lib ggthemes

Capítulo 1

Os pássaros de Darwin e o método hipotético-dedutivo.

Testes estatísticos e distribuições probabilísticas

Introdução

Charles Darwin observou que os pássaros fringilídeos nas ilhas de Galápagos apresentavam variedades de formato e tamanho dos bicos. Sua intuição sobre a origem das variedades a partir de um ancestral comum foi um dos argumentos mais contundentes do “On the Origin of Species” (1859). Essa história é o ponto de partida para este capítulo.

Estudamos a relação natural entre ciências empíricas e duas distribuições probabilísticas: a distribuição normal e a distribuição t, relacionadas entre si. A adoção da distribuição normal em trabalhos científicos é popular, porém os motivos são pouco entendidos. O Teorema do Limite Central é fundamental nesse contexto.

Usamos as distribuições citadas para estudar as medidas dos bicos dos tentilhões em pequenas amostras de cada ilha e fazer inferências sobre as populações. O racional de testes de hipótese é introduzido.

Pássaros em Galápagos

Em sua viagem pelo mundo a bordo do Beagle, Charles Darwin descreveu um grupo de pássaros que habita as Ilhas Galápagos, arquipélago localizado a aproximadamente 900 km da costa do Equador (América do Sul). A variedade em tamanhos dos bicos chamou atenção: “It is very remarkable that a nearly perfect gradation of structure in this one group can be traced in the form of the beak, from one exceeding in dimensions that of the largest gros-beak, to another differing but little from that of a warbler”.⁴ *The Voyage of the Beagle* (1839).

É interessante notar que a linguagem usada para denotar diferenças é eminentemente quantitativa (dimensions, largest, differing). Darwin não conduziu estudos quantitativos por razões práticas. Neste capítulo, simularemos o mesmo cenário empregando métodos estatísticos para comparar os pássaros.

Antes da publicação de *A origem das Espécies*, o caso dos fringilídeos (nome destas aves) já continha um embrião do processo de seleção natural. Na segunda edição, em 1845, ele especula sobre um grupo ancestral comum moldado por fins específicos:

“Seeing this gradation and diversity of structure in one small, intimately related group of birds, one might really fancy that from an original paucity of birds in this archipelago, one species had been taken and modified for different ends.”⁵

⁴É bastante notável que uma graduação quase perfeita na estrutura desse grupo possa ser traçada na forma do bico, desde um excedendo as dimensões do maior dos pardais bico-gordo, até outro diferindo pouco do papa-amoras.

⁵(...)[ao] ver esta graduação e diversidade em estrutura em um pequeno, intimamente relacionado grupo de pássaros, é possível imaginar que, a partir de poucos pássaros deste arquipélago, uma espécie foi escolhida e modificadas para certos fins. Darwin, Charles (1845), *Journal of researches into the natural history and geology of the countries visited during the voyage of H.M.S. Beagle round the world, under the Command of Capt. Fitz Roy, R.N* (2nd. ed.), London: John Murray

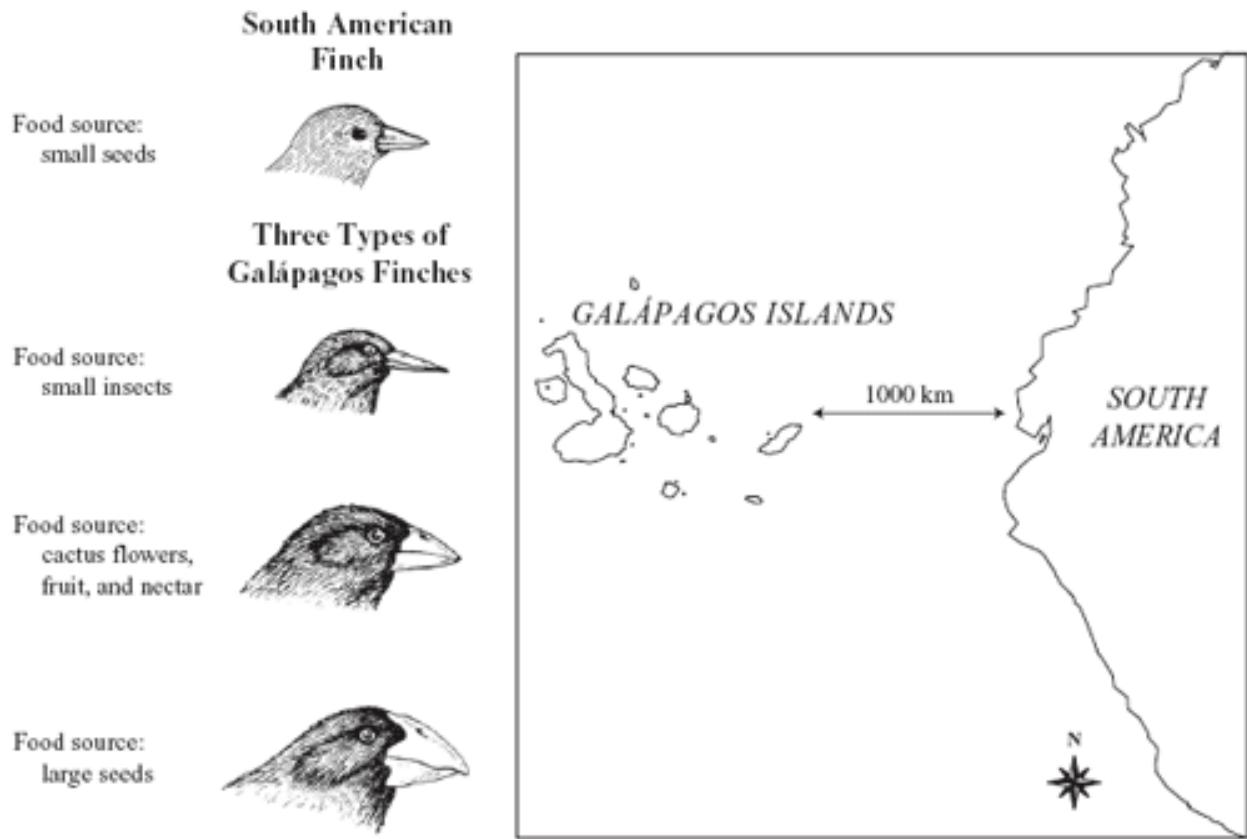


Figure 2: Figura 1. Fringilídeos em Galápagos.

Darwin observou que a variedade dos bicos era adaptada à dieta de cada grupo: frutas, nozes, insetos. Os de bico pontudo conseguem comer frutas e arilo da sementes do cacto, enquanto os de bico curto estraçalham a base do cacto e comem sua polpa.

A inspeção visual de um naturalista treinado foi capaz detectar essas nuances. Sob sua percepção, havia um total de 3 espécies em 4 ilhas: 1 na Ilha Charles, 1 na Ilha Albemarle e 1 nas ilhas James e Chatham. Inicialmente, notou que os pássaros eram semelhantes àqueles vistos no Chile. Darwin coletou 26 pássaros e os levou de volta para que um ornitólogo os estudasse com mais detalhe. John Gould sugeriu que os 26 pássaros representavam 12 espécies completamente novas, número que posteriormente passou para 25. Hoje, os taxonomistas sugerem um número de 15 espécies para os fringilídeos de Darwin.

Pensaremos como biólogos interessados em estudar quantitativamente o tamanho dos bicos. Usaremos estatística e probabilidades para testar hipóteses e fazer conclusões mais acuradas sobre as medidas, explorando diferenças entre os grupos de pássaros de Galápagos.

A distribuição normal e um curioso teorema

Em trabalhos empíricos, é comum a suposição de que medidas de uma variável aleatória vêm de uma população com distribuição normal. A seguir, vamos estudar o comportamento dessa função probabilística.

Abraham de Moivre (26 May 1667 – 27 November 1754), sem financiamento exclusivo para estudos e pesquisa, prestava serviços. Entre eles, cálculos de probabilidades em jogos de azar para clientes. Em 1733, de Moivre percebeu que as probabilidades de uma distribuição binomial, como o lançamento de moedas ($p=0.5$), aproximam-se de uma curva suave (contínua) à medida em que o n aumenta.

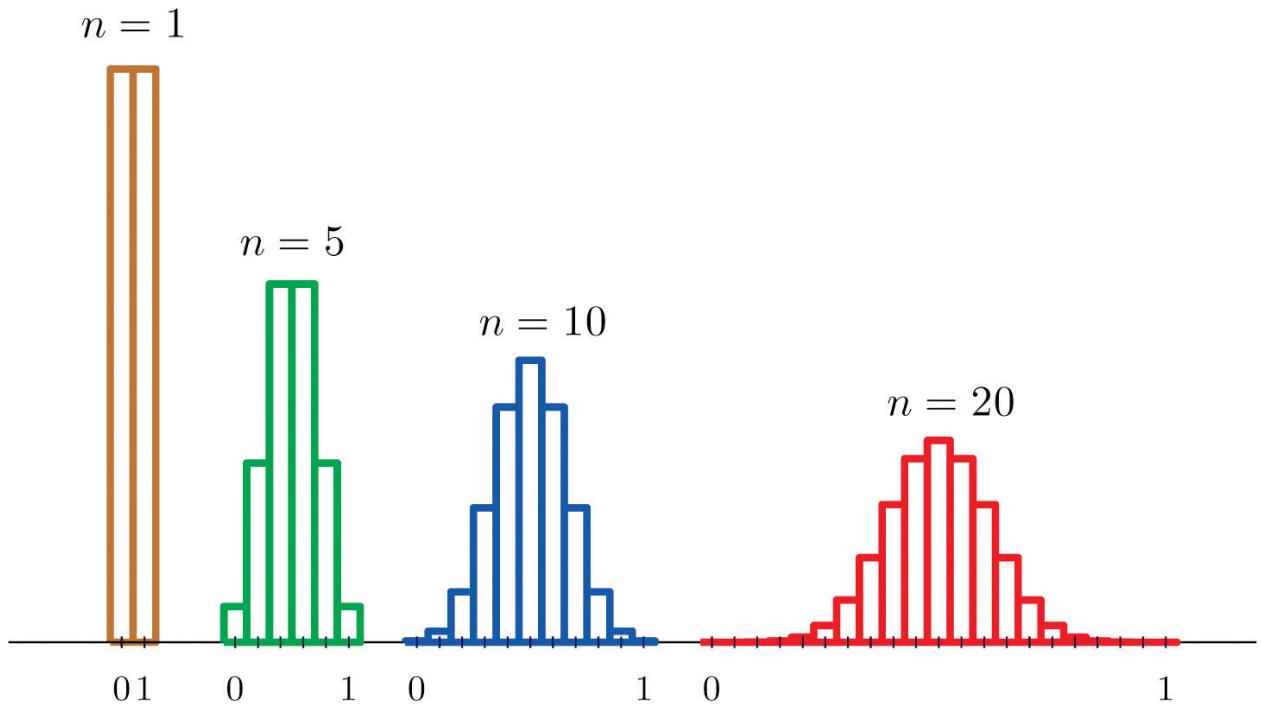


Figure 3: Figura 2. Distribuição binomial com $p = 0.5$ (e.g: lançamento de uma moeda). Para $n > 1$, valores extremos indicam resultados com apenas caras (cauda à esquerda, 0000...) ou coroas (cauda à direita, 1111...)

Para $n = 1$, temos uma distribuição uniforme $P(1) = P(0) = 0.5$. Para um número maior de lançamentos, os resultados mais frequentes são números parecidos de caras (0s) e coroas (1s).

Para $n = 10$, é muito mais provável obter um número de caras próximo a 5 (centro das curvas) que um resultado com 9 ou 10 lançamentos iguais. É possível demonstrar que aumentar o valor de n faz com que a distribuição se aproxime da seguinte curva contínua:

Algumas características nesta curva são desejáveis e/ou plausíveis para o estudo de fenômenos naturais: simetria; valores mais prováveis concentrados ao redor de um pico, valores extremos improváveis; distribuição governada por poucos parâmetros (média e variância).

De Moivre intuiu que a distribuição de binomiais com muitos lançamentos aproximava o de uma função suave. Mas qual curva? Parecia o de um número elevado aos valores de uma quadrática negativa: $P(x) = e^{-x^2}$. Em verdade, ele deduziu a solução para o problema das moedas ($p = \frac{1}{2}$). A seguinte expressão geral descreve a probabilidade $P(x)$ correspondente à curva que procuramos.

$$P(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$$

O número misterioso em questão é o número de Euler ($e \sim 2.72\dots$), que será melhor explicado no capítulo 3. Há um fator, $\frac{1}{\sqrt{2\pi}}$ (aproximadamente 0.4), multiplicando o resultado. Em R, podemos definir a função:

```
>mgauss <- function(x) 0.4*exp((-1)*(x^2)/2)
```

Em seguida, obter valores no intervalo $[-5, 5]$ e plotá-los:

```
>library(ggplot2)
>x_seq <- seq(-5,5,by = 0.1)
>ndist <- purrr::map(.f=mgauss,.x=x_seq) %>% unlist
>ggplot(data.frame(ndist,x_seq),aes(x=x_seq,y=ndist))+
  geom_point()+
```

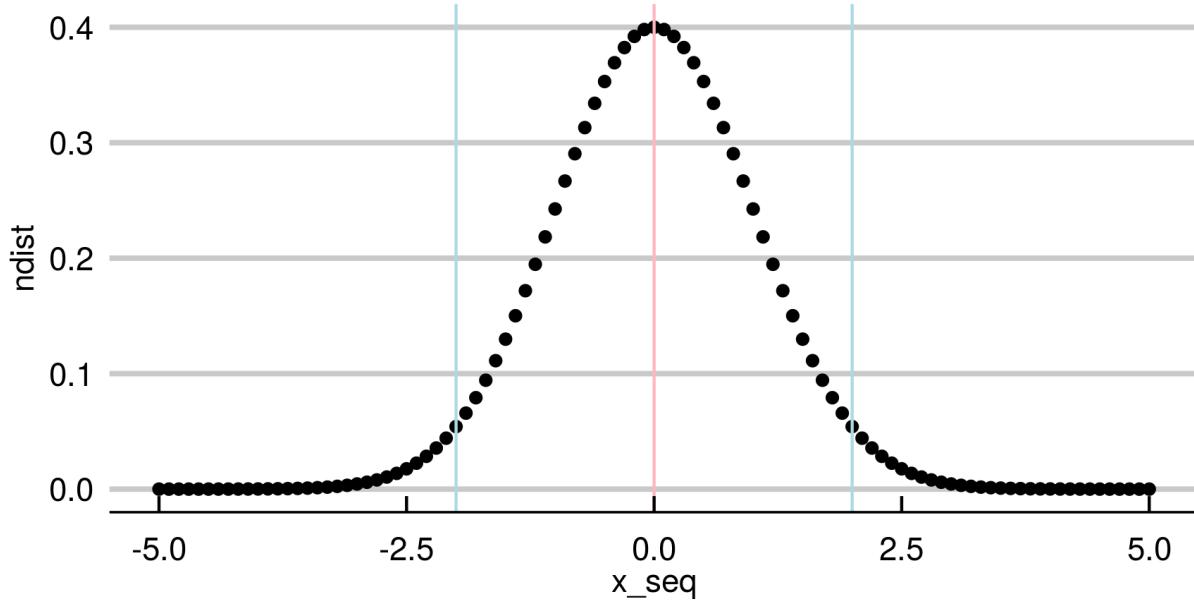
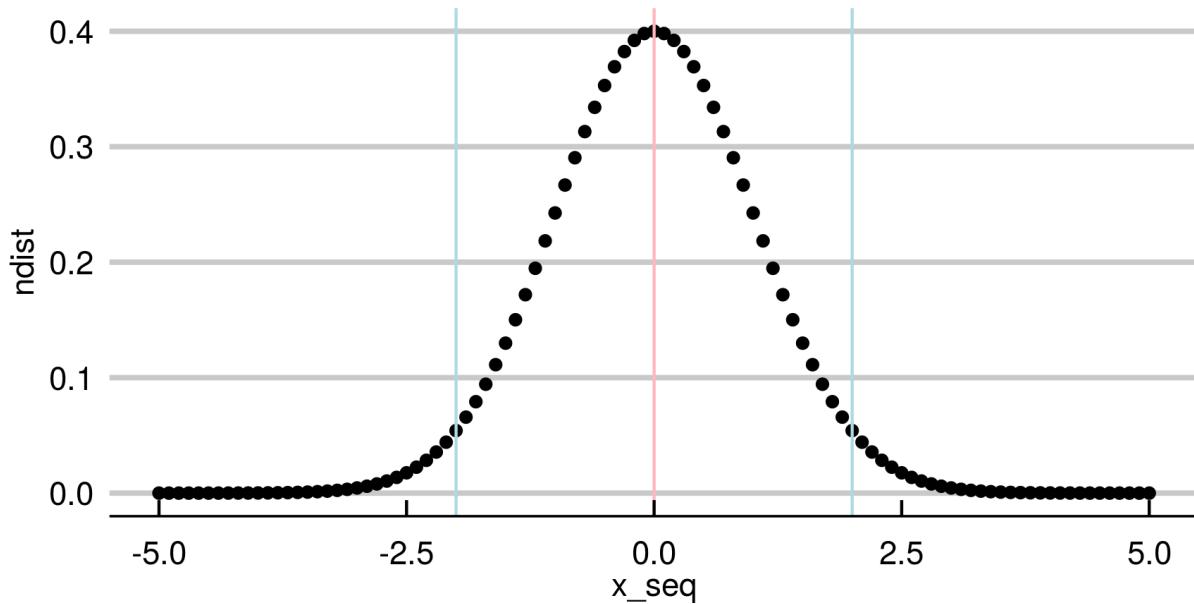


Figure 4: Figura 3: Distribuição normal (gaussiana), cujo formato lembra o de um sino

```
geom_vline(xintercept = 2,color="light blue")+
  geom_vline(xintercept = -2,color="light blue")+
  geom_vline(xintercept = 0,color="light pink")+
  theme_economist_white(gray_bg = F)
```



Observamos como a distribuição gaussiana se dá a partir da equação.

É evidente que $-x^2$ sempre retorna valores negativos. Nossa função gera valores entre 0 e 1 exponenciando e

$(e \sim 2.718\dots)$ a um fator negativo quadrático ($y = 0.4 * e^{-x^2/2}$).

Examinando o comportamento da equação, notamos que valores próximos à média ($x \sim \mu = 0$) fazem com que o expoente de e , dado por $-x^2/2$, se aproxime de 0, maximizando nossa função $f(0) = 0.4 * e^{-x^2/2} = 0.4 * e^0 = 0.4$). O valor obtido (0.4) corresponde ao topo da curva no gráfico acima (linha rosa).

Observamos a curva se aproximar do máximo simetricamente para valores próximos de 0.

Isso reflete diretamente o fato de que valores próximos à média serão mais prováveis e valores extremos menos prováveis.

Para comparação: $f(2) = 0.4 * e^{-2^2/2} = 0.4 * e^{-2} = 0.4 * 0.135 \sim 0.05$ (linha azul). A probabilidade de se obter o valor médio ($x = 0, p \sim 0.4$) é oito vezes maior que a probabilidade de obter o valor 2 ($x = 2; p = 0.05$). O termo quadrático torna a distribuição simétrica para valores opostos em relação à média. $P(x) = P(-x)$. Como calculamos $P(2)$ antes, sabemos que: $P(-2) = P(2) = 0.05$ para $\mu = 0\$$. É igualmente provável encontrar valores duas unidades maiores ou duas unidades menores que a média. Esses pontos estão marcados por uma linhas azuis na figura.

Podemos trabalhar com curvas normais com centros (média μ) deslocados para a esquerda ($\mu < 0$) ou para a direita ($\mu > 0$), subtraindo o termo de x em nosso expoente. Além disso, diferentes variâncias (σ^2) refletem a frequência de valores longe da média e o quanto distante dela eles são. Visualmente, determina o tamanho da base do sino na ilustração (Figura 3). Usamos a notação $N \sim (\mu, \sigma^2)$ para descrever uma distribuição gaussiana arbitrária:

$$P(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Poderíamos encontrar características desejáveis, como a simetria citada acima, em outras distribuições. Então, por que usamos uma equação mais complexa? Distribuições binomiais grandes e lançamentos de moedas são tão importantes?

O Teorema do Limite Central

A razão é o Teorema do Limite Central.

Se somarmos muitas distribuições de uma mesma família, a distribuição resultante se aproxima de uma normal.

Exemplos ajudam a ganhar intuição. Ao lançar um dado justo de 6 faces, temos probabilidade de $\frac{1}{6}$ em cada resultado.



Uma distribuição discreta uniforme, em que $P(1) = P(2) = P(3) = P(4) = P(5) = P(6)$ e definida para números naturais entre 1 e 6: $X \sim U_{discr}(1, 6)$.

A média para muitos lançamentos, ou valor esperado, é dado por:

$$E(X) = E(U(1, 6)) = (1 + 6)/2 = 3.5$$

Vamos fazer um experimento virtual usando 100 lançamentos de 11 dados.

O código em R para a seguir gera os dados e as visualizações de que precisamos:

```
>library(magrittr)
>library(ggthemes)
>library(ggplot2)
>source("multiplot.R")
>set.seed(2600)
>n_plots <- 12

>dice_fun <- function(n){runif(n, min=0, max=6) %>% ceiling} # Random samples
>data_mat <- replicate(n=n_plots-1,dice_fun(100)) # Replicate
>data_mat <- cbind(data_mat,rowSums(data_mat)) # Sum

>plot_list <- vector("list", n_plots) # Plot each distribution
>plot_list <- apply(X=data_mat, MARGIN=2, FUN=function(x)
  ggplot(data.frame(obs=x),aes(x=obs)) +
    geom_histogram(binwidth = 0.2) +
    ylab("")+xlab("")+
    theme_economist())

>m_plot <- multiplot(plotlist = plot_list,cols=n_plots/3)
```

```
# Multiplot function available at: http://www.cookbook-r.com/Graphs/Multiple_graphs_on_one_page_(gg
```

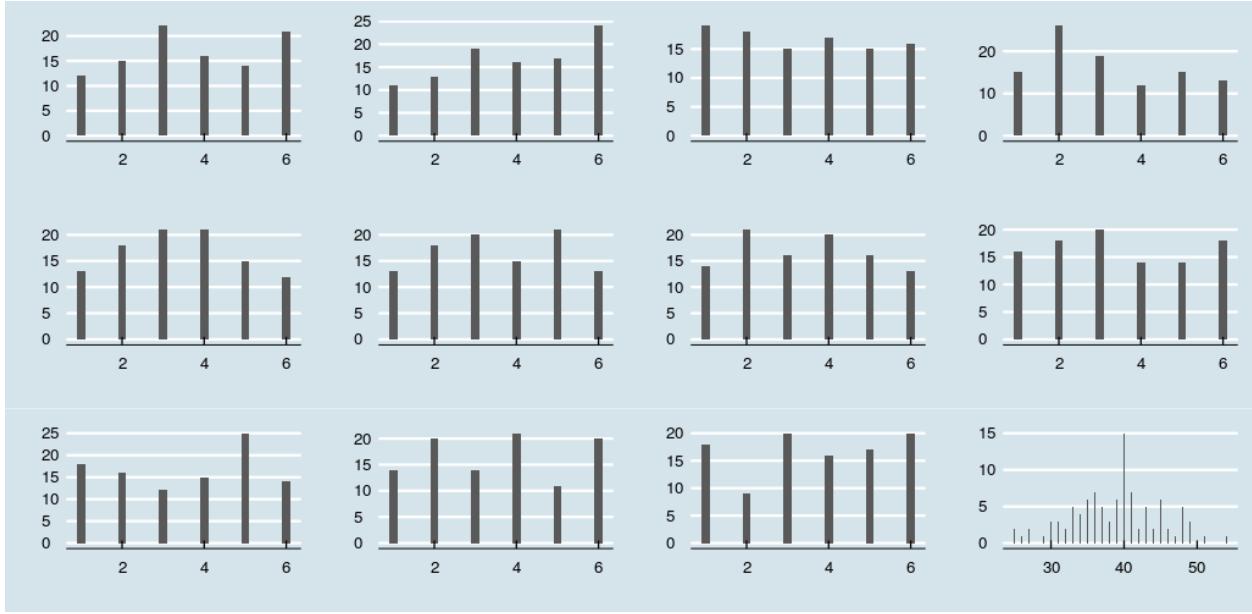


Figure 5: Figura 2 . Soma de amostras ($n=100$) de 11 distribuições uniformes correspondentes ao lançamento de dados honestos de 6 faces. O resultado está na célula inferior à direita.

Notamos que as barras estão distribuídas com alturas bastante parecidas nas 11 primeiras células. A frequência esperada para cada valor é $\sim 1/6$ do total de 100 lançamentos. $Freq(X_i) \sim \frac{1}{6} * 100 \sim 16.66$
Algo interessante ocorre com a soma das distribuições (canto inferior direito).

O valor esperado é, como diz a intuição, a soma dos valores esperados em cada amostra:
 $E(X) = \sum_{i=1}^{11} E(U_i \sim (1, 6)) = 11 * 3.5 = 38.5$

O valor 38.5 corresponde aproximadamente ao centro da distribuição resultante (Figura 2, canto inferior direito) Entretanto, a distribuição muda de forma! Sem muito esforço, é notável a semelhança com a curva normal, com valores extremos menos frequentes e simetricamente afastado da média (valor esperado).

É possível provar que a soma de muitas distribuições de uma mesma família converge para a distribuição normal em qualquer caso. Desde que estas sejam independentes. A esse resultado damos o nome de Teorema do Limite Central.⁶ Este resultado tem uma util importância para o estudo dos fenômenos naturais através de experimentos.

Ciência experimental e o Teorema do Limite Central

Muitos objetos de interesse para os cientistas são simplificações de fenômenos complexos. Um exemplo trivial está na cor da pele de seres humanos. Uma parte considerável depende do número de genes herdados relacionados à melanina. Eles se comportam de maneira aditiva. Assim, cada variante de gene extra pode contribuir para a cor final com X unidades na escala para medir pigmentação.

A cor de um indivíduo será influenciada pela soma dessas distribuições, o que é análogo à matemática descrita para os lançamentos de dados.

⁶Prova formal em <http://www.cs.toronto.edu/~yuvalf/CLT.pdf>



Podemos comparar grupos quanto a medidas fenotípicas finais (cor da pele) sem saber detalhes sobre as relações entre cada gene e seus mecanismos de expressão e regulação.

A distribuição final de melanina vem da soma de distribuições individuais semelhantes e tenderá a ser normal.

Como vimos, o mesmo é válido para quaisquer distribuições subjacentes: se elas forem gama, uniformes ou de Poisson, a distribuição da soma ainda tenderá à normalidade.

A figura 2 mostra a soma de distribuições uniformes para dados honestos, evidenciando que esta se aproxima de uma normal.

$$X \sim U_1(1, 6) + U_2(1, 6) + \dots + U_{11}(1, 6) = X \sim N(38.5, \sigma^2)$$

Vamos visualizar o mesmo processo para uma outra família de distribuições, gamma:

$$X \sim \gamma_1(\alpha, \beta) + \dots + \gamma_n(\alpha, \beta) = X \sim N(\mu', \sigma')$$

Para valores grandes de n:

```
>gamma_fun <- function(n){rgamma(n,1)}
>data_mat <- replicate(n=n_plots-1, gamma_fun(100))
>data_mat <- cbind(data_mat, rowSums(data_mat))

>plot_list <- vector("list", n_plots)
>plot_list <- apply(X=data_mat, MARGIN=2, FUN=function(x)
  ggplot(data.frame(obs=x), aes(x=obs)) +
    geom_histogram(binwidth = 0.2) +
    ylab("")+xlab("")+
    theme_economist())

>m_plot <- multiplot(plotlist = plot_list, cols=n_plots/3)
```

Novamente, verificamos que a soma começa a ser simétrica em torno da média, com formato de sinos (base alargada). Os fenômenos observáveis em nosso universo são naturalmente complexos. Especialmente em sistemas biológicos, há redundância de componentes e um objeto de interesse para cientistas é resultado da combinação de muitas variáveis subjacentes. O teorema do limite central permite que utilizemos distribuições normais para uma grande variedade de problemas. Ainda que as distribuições subjacentes sejam desconhecidas, a variável resultante de uma grande combinação será gaussiana em muitos casos.

A descoberta das equações que regem esses mecanismos de convergência foi um grande avanço para as ciências experimentais.

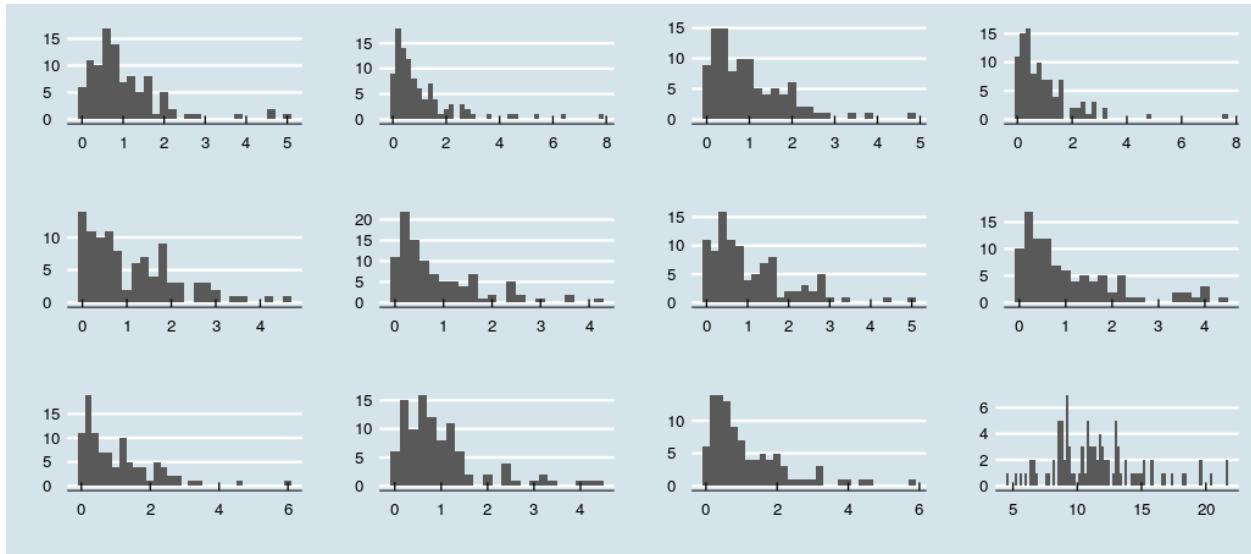


Figure 6: Figura 3. Soma de amostras ($n=100$) de 11 distribuições gama. O resultado está na célula inferior à direita. Função de densidade de probabilidade para distribuição gama: $f(x) = 1/\Gamma(\alpha) * \beta^\alpha * x^{\alpha-1} * e^{-\beta x}$, com $\alpha = \beta = 1$

Exercícios

1. Sobre a distribuição normal para uma variável aleatória, é verdadeiro (mais de uma possibilidade):
 - a. A soma da probabilidade de todos os valores possíveis é 1.
 - i. $\int_{-\infty}^{+\infty} f(x)dx = 1$.
 - b. É simétrica em relação à moda.
 - c. O valor esperado é dado por $1/\sigma\sqrt{2\pi}$.
 - d. 95% dos valores estão próximos à média.
 - e. Valores extremos são improváveis.
 - f. É unicamente determinada por variância σ^2 e média μ .
 - g. É contínua e diferenciável, facilitando cálculos.
 - h. Amostras pequenas resultam em distribuições t.
2. Usando o comando “?Distributions” acesse algumas distribuições disponíveis na biblioteca de base do R.
 - a. Plote o histograma da soma de 100 distribuições X^2 (`rchisq;n = 60`).
 - b. Faça o mesmo procedimento para 100 distribuições de outra família e tamanho à sua escolha.
 - c. Obtenha os valores de skewness e kurtosis para essas distribuições. Uma distribuição normal padrão ($\sigma^2=1;\mu=0$) possui skewness (assimetria) de 0 e kurtosis (volume dos valores mais extremos) de 3. Quais os encontrados por você?
 - d. Cite dois fenômenos naturais cuja distribuição estatística é conhecida e qual a distribuição correspondente.

Darwins's Finches

Mostraremos como a contribuição individual de genes com efeitos aditivo de distribuição uniforme resulta em medidas aproximadamente normais para os bicos das aves.

Vamos simular as medidas de bicos em 4 amostras ($n=150$) de pássaros.

O tamanho dos bicos é dado pelo efeito aditivo de muitos genes semelhantes, portanto esperamos que sua distribuição seja normal pelo Teorema do Limite Central.

Uma cópia do gene adiciona x milímetros ao tamanho final. O valor de x é sorteado de uma variável aleatória de distribuição uniforme, $X \sim U(0, 1)$.

Pássaros têm um número fixo de n de genes aditivos em cada amostra, sorteado no intervalo entre 80 e 100. A medida final dos bicos é dada pela soma efeitos dos n genes. Esse número é fixo em cada população e varia entre populações.

Para simular os dados com as condições acima:

```
>library(magrittr)
>library(ggthemes)
>library(ggplot2)
>set.seed(2600)

>n_birds <- 150 # sample_size
>genes_low <- 80 # lower bound on number of genes
>genes <- 100 # upper bound on number of genes
>n_islands <- 4 #samples

>unif_sum <- function(genes){
  replicate(n = genes,
            expr = runif(100, min=0, max = 1)) %>%
  rowSums
}

>generate_pop <- function(n_pop,n_genes){
  replicate(n=n_pop,
            expr = unif_sum(n_genes) %>% mean)
}

>galapagos_birds <- purrr::map(.f = function(x) generate_pop(n_pop=n_birds,
                                                               n_genes = x),
                                 .x = runif(n=n_islands, genes_low, genes) %>% ceiling) %>%
  unlist %>% matrix(nrow=n_birds,byrow=F) %>%
  data.frame
```

Verificando o histograma das medidas finais, notamos, mais uma vez, que as medidas se aproximam de uma gaussiana.

```
>my_alpha <- 0.5
>my_bins <- 50
>ggplot(data=galapagos_birds,aes(x=X1))+
  geom_histogram(alpha=my_alpha,bins = my_bins)+
  geom_histogram(data=galapagos_birds,aes(x=X2),fill="dark blue",
                 alpha=my_alpha,bins = my_bins)+
  geom_histogram(data=galapagos_birds,aes(x=X3),fill="dark red",
                 alpha=my_alpha,bins = my_bins)+
  geom_histogram(data=galapagos_birds,aes(x=X4),fill="dark green",
```

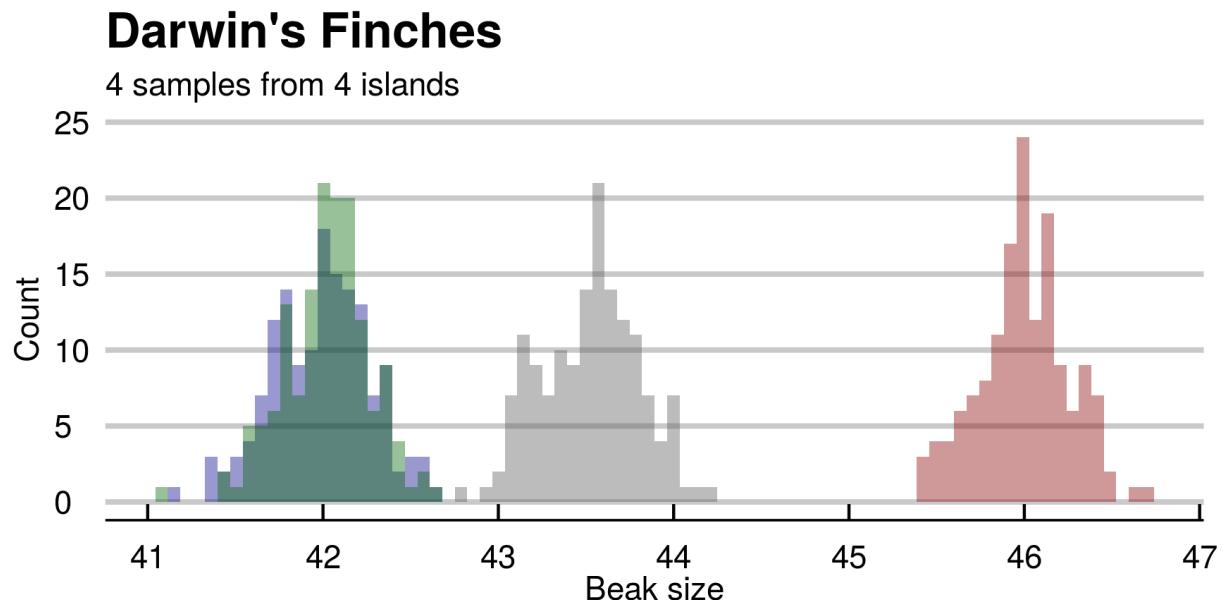


Figure 7: Figura 4. Distribuição das medidas de bicos em populações simuladas para genes com efeito aditivo.

```

alpha=my_alpha,bins = my_bins) +
xlab("Beak size") + ylab("Count") +
ggtitle("Darwin's Finches", subtitle = "4 samples from 4 islands") +
theme_economist_white(gray_bg = F)

```

Curiosamente, os números aleatórios gerados usando a semente sugerida (set.seed(2600) na linha 4) são semelhantes à suposição de Darwin: 4 ilhas (amostras) e três espécies (distribuições de bicos). Notamos que há duas amostras (verde, azul) de medidas bastante parecidas e outras duas separadas (cinza, vermelho). Supondo que medimos os bicos de algumas aves, como saber se os grupos são diferentes? Calculando as diferenças entre distribuições, podemos inferir se duas amostras têm o mesmo número de genes subjacentes! Para isso, usaremos um racional e algumas ferramentas novas.

Testes de hipótese

Filósofos da ciência estudam características no modus operandi de outros estudiosos. O que há em comum entre os procedimentos empregados por biólogos e geólogos? O que distingue Charles Darwin e Paul Dirac de John Dee e Edward Kelley? O que funciona em áreas distintas do conhecimento humano?

Adotamos a denominação coletiva de “ciências” para algumas áreas do conhecimento. Ainda, associamos a elas características em comum nos procedimentos e na estrutura interna. De alguma forma, científicidade comunica credibilidade. Nas últimas décadas, filósofos discutiram a validade do problema de demarcar ciência de pseudociência e não-ciência.⁷ Neste capítulo, vamos nos ater a um paradigma conceitual mais antigo e indiscutivelmente influente.

O método hipotético-dedutivo foi popularizado no século XX como uma bandeira de identificação associada ao trabalho científico. Um ciclo que consiste em formular hipóteses, desenhar experimentos, testar hipóteses falseáveis, verificar resultados e repetir o processo de forma iterativa.

O racional em usar hipóteses testáveis é de que proposições válidas sobre um sistema contém informações que ajudam a prevê-lo. Assim, “faz sol ou não amanhã” é uma proposição inútil, enquanto “faz sol amanhã” é uma proposição útil. Note que “faz sol amanhã” é uma hipótese testável (falseável), enquanto “faz sol ou não amanhã” é uma hipótese verdadeira independente das observações.

O exemplo é grosseiro, porém alguns ramos do conhecimento humano produziram hipóteses não-falseáveis. K. Popper, líder da revitalização do método hipotético dedutivo no século passado, atacou severamente o materialismo dialético de Karl Marx, assim como a teoria de evolução por seleção natural de Charles Darwin. Por exemplo, Marx previu que a revolução aconteceria em um país industrializado e outros eventos que não se concretizaram. Seus seguidores justificaram com hipóteses ad-hoc. A teoria da evolução por seleção natural de Darwin era amparada em muitos exemplos de reprodução impossível (recomposição da trajetória evolutiva em fósseis). A psicanálise também sofreu duras críticas, em virtude da irrefutabilidade de seus pilares centrais.

Para Popper, a dificuldade em gerar hipóteses testáveis e falseáveis sinalizava uma evidente fragilidade nas teorias, as quais não empregariam métodos científicos em seus avanços.

Uma maneira de formalizar essa ideia, incorporando o uso de ferramentas quantitativas, é através de probabilidades. Calculamos a probabilidade associada a observações, considerando o cenário de uma hipótese (falseável). Esse racional adequa ferramentas matemáticas robustas à plataforma epistemológica de Popper, sendo um modelo dominante de produção em ciências experimentais.

Em geral, os pesquisadores formulam uma hipótese base, chamada hipótese nula, que descreve o cenário menos interessante para o trabalho. Por exemplo, se estamos comparando dois grupos, A e B, quanto a uma intervenção, a hipótese nula costuma declarar que os grupos são iguais.

Queremos estudar o tamanho dos bicos de pássaros das ilhas A e B. A hipótese nula natural é: Não há diferença entre os bicos dos pássaros do tipo A e B.

Medimos o bico de alguns pássaros dos dois grupos e calculamos a probabilidade de encontrarmos essas medidas considerando que A e B são iguais. Se essa probabilidade for muito baixa, rejeitamos nossa hipótese.

Estruturando os passos:

1. Definimos a hipótese nula (H_0) e pelo menos uma hipótese alternativa(H_1).
 - H_0 : Pássaros das ilhas A e B possuem bicos de tamanho igual.
 - H_1 : Os pássaros possuem bicos de tamanho diferentes.

Então, podemos fazer um experimento, coletando medidas experimentais para o comprimento dos bicos. Essas medidas, junto a premissas matemáticas razoáveis, permitem especular: qual a probabilidade p de obter nossas observações considerando distribuições iguais entre A e B? Isto é, considerando H_0 verdade, nossos resultados seriam raros ou comuns?

⁷Massimo Pigliucci - Philosophy of Pseudoscience: Reconsidering the Demarcation Problem

Caso p seja menor que um limiar arbitrariamente pré-definido (convencionalmente, 0.05), rejeitamos H_0 . A probabilidade é pequena se H_0 for verdade. A interpretação do valor p não é muito intuitiva. Envolve mensurar quão improváveis são as observações em um cenário hipotético na vigência da hipótese nula. Sua tradução (errada) mais popular é de que representa “a chance de o resultado deste estudo estar errado”.

A domínio dos procedimentos hipotético-dedutivos nas ciências produziu resultados interessantes. No sentido proposto por Thomas Kuhn de ciência normal como acúmulo de evidências e testagem de hipóteses. O fantasma de desenhar um experimento com possibilidade imparcial de falha aguçou a percepção de pesquisadores para a falibilidade de ideias. O grau de sofisticação em reproduzibilidade de procedimentos foi amplificada.

Em contrapartida, o arcabouço acima é suficiente para produzir um trabalho científico críptico para leigos. Segundo receitas pré-definidas, um texto parece estar em conformação com os padrões acadêmicos, ainda que a hipótese elementar em torno do objeto de pesquisa seja simplória. Outro efeito colateral é a busca por valores p baixos, desprezando precedentes teóricos e premissas probabilísticas (múltiplos testes).

Um exemplo prático: Teste t para amostras independentes

Para testar estatisticamente se as medidas são diferentes, executaremos um teste t para comparação dos grupos.

A distribuição t surge quando queremos entender quão improváveis são nossas estimativas (μ') supondo uma média real hipotética (μ) de origem em uma variável de distribuição normal desconhecida.

Exemplo: Medimos os bicos de 30 pássaros. Obtivemos média amostral $\mu' = 38$ mm com desvio-padrão $\sigma' = 0.3$ mm. Supondo que a média real (μ) é de 40 mm, qual é a probabilidade obtermos $\mu' = 38$ mm em uma amostra aleatória (nossa situação)?

Entender a imprecisão da estimativa de uma média foi o eixo principal para a descrição dessa distribuição por William Gosset. Sob o pseudônimo Student, o estatístico, que trabalhava para a fábrica de cerveja Guinness, publicou na Biometrika (1908) o famoso texto The probable error of a mean.

Para entender a imprecisão, necessitamos de uma medida da dispersão dessas medidas.

Assumimos amostras retiradas de uma variável aleatória com distribuição normal e média μ . Podemos retirar j amostras de tamanho N e calcular a média dessas amostras $\mu'_1, \mu'_2, \dots, \mu'_j$. As médias amostrais μ'_j são estimativas da média real μ .

$$\mu'_1, \mu'_2, \dots, \mu'_j$$

Qual a dispersão das estimativas $\mu'_1, \mu'_2, \dots, \mu'_j$?

Para um conjunto de estimativas $\mu'_1, \mu'_2, \dots, \mu'_j$ chamamos ela de **erro padrão** (*standard error of the mean*), dado pelo desvio-padrão populacional dividido pela raiz quadrada do tamanho da família de amostras em questão (std. err. = σ/\sqrt{n}). Como não sabemos o desvio-padrão na população, usamos o valor de σ'_j das amostras. Student propôs o uso de uma quantidade para estimar a probabilidade de uma estimativa μ' dado centro μ . Essa quantidade pivotal é a razão entre (1) distância das estimativas e média real, $\mu' - \mu$, e (2) o erro padrão. A estatística t:

$$t = \frac{Z}{s} = (X - \mu)/\sqrt{n}$$

Assim, a estatística t para nosso exemplo ($\mu'=38$; $\mu= 40$; $n=30$; $\sigma'=0.3$) é:

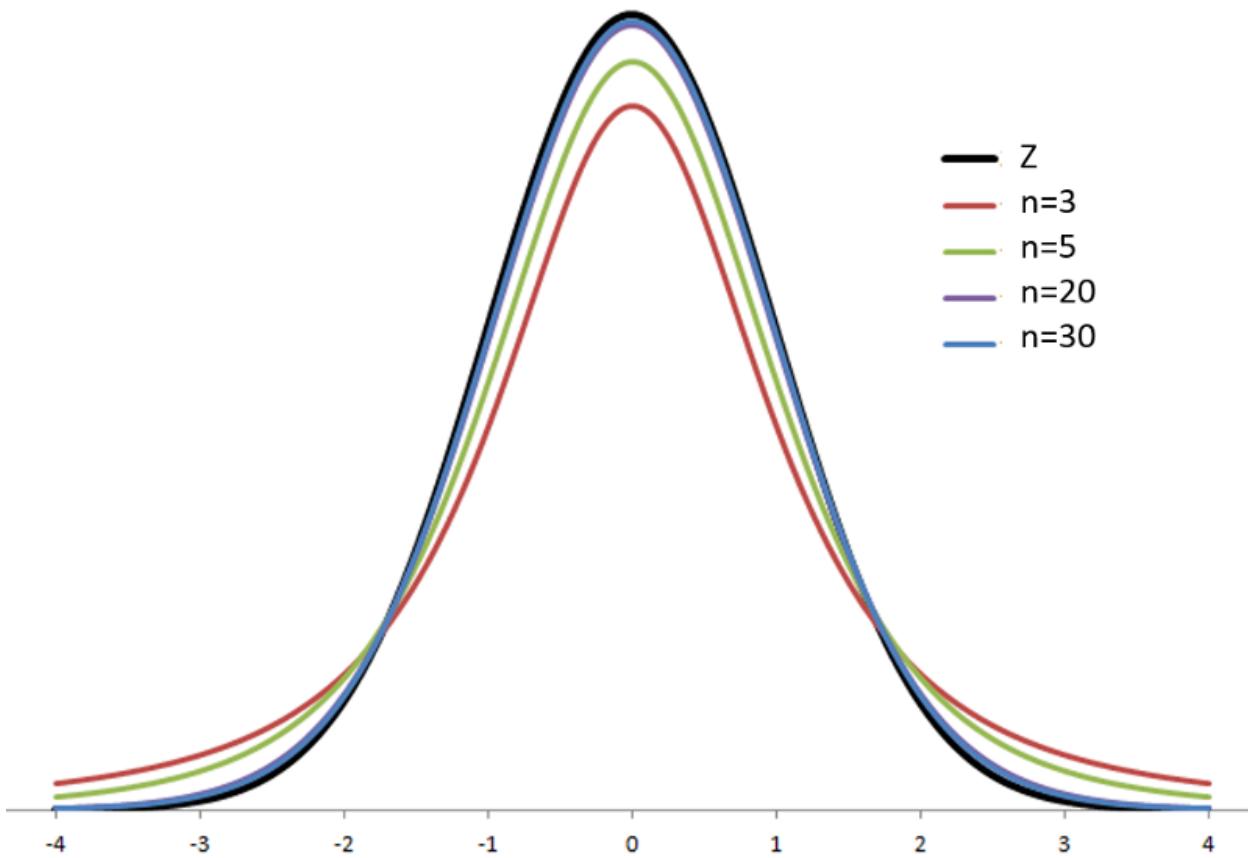
$$t = (38 - 40)/\sqrt{30}$$

Student (Gosset) mostrou que essa estatística segue uma distribuição probabilística (t de Student) definida:

$$f(t) = \frac{1}{\sqrt{\nu} B(\frac{1}{2}, \frac{\nu}{2})} \left(1 + \frac{t^2}{\nu}\right)^{-\frac{\nu+1}{2}}$$

B é a função Beta e v são graus de liberdade. Possui densidade parecida com a da distribuição normal, porém com probabilidades maiores para valores extremos. O parâmetro v (graus de liberdade) expressa essa

característica. Empiricamente é estimado pelo tamanho das amostras usadas na estimativa de μ' . Associamos uma amostra (tamanho n) retirada de uma população normal (tamanho arbitrariamente alto, $n \rightarrow \infty$) a uma distribuição t com $(n-1)$ graus de liberdade. Em nosso exemplo, $n=30$, então $v = 29$ ($n-1$).



Maiores valores correspondem a amostras maiores e fazem com que a distribuição t se aproxime de uma distribuição normal. Em um caso extremo, temos nsamples=npop. e as amostras são idênticas à distribuição de origem.

Sabendo a estatística t ($t = -36.51$) e os graus de liberdade para nossa família de amostras ($v = 29$), podemos usar a expressão $f(t)$ para saber a probabilidade de obtermos nossa média 38 mm se a média real for de 40 mm. Em R, a função pt retorna a probabilidade total de valores extremos menores que a estatística t fornecida.

```
>pt(36.51, df = 29)
[1] 4.262e-26
```

A nossa hipótese considera valores t extremos maiores (direita) ou menores (esquerda). Então consideraremos também o valor p associado aos extremos maiores que a estatística t simétrica positiva ($t_{min} = -36.51$; $t_{max} = 36.61$). É o que encontraríamos com a medida simétrica em relação à média de 42 mm.

A seguinte expressão deveria funcionar.

```
>(1- pt(36.51, df = 29)) + pt(-36.51, df = 29)
[1] 4.262e-26
```

Entretanto, o valor retornado parece idêntico ao de $pt(-36.51, df = 29)$. Acontece que o R aproxima a soma das probabilidades de valores menores que $t = 36.51$ para 1, zerando o primeiro componente da soma ($pt(36.51, df = 29)$). Sabendo que a distribuição t é simétrica, podemos fazer então usar o seguinte truque:

Additional hours' sleep gained by the use of hyoscyamine hydrobromide.

Patient	1 (Dextro-)	2 (Laevo-)	Difference (2-1)
1.	+ .7	+ 1.9	+ 1.2
2.	- 1.6	+ .8	+ 2.4
3.	- .2	+ 1.1	+ 1.3
4.	- 1.2	+ .1	+ 1.3
5.	- 1	- .1	0
6.	+ 3.4	+ 4.4	+ 1.0
7.	+ 3.7	+ 5.5	+ 1.8
8.	+ .8	+ 1.6	+ .8
9.	0	+ 4.6	+ 4.6
10.	+ 2.0	+ 3.4	+ 1.4
	Mean + .75	Mean + 2.33	Mean + 1.58
	S. D. 1.70	S. D. 1.90	S. D. 1.17

Figure 8: Retirado de The probable error of a mean, pag. 20. Os dados estão disponíveis na biblioteca de base do R, sob o nome ‘school’.

```
> 2*pt(-36.51, df = 29)
[1] 8.524e-26 # valor p
```

Nota

Uma percepção errônea comum sobre a distribuição t é de que ela é resultado amostras pequenas retiradas de uma variável aleatória normal. Qualquer amostra retirada de uma variável de distribuição normal terá, por definição, distribuição normal, ainda que seja composta por 1 ou 2 observações. O que segue distribuição t é a quantidade pivotal descrita acima.

Na sessão IX do artigo, Student (Gosset) demonstra como seu insight pode ser usado para testar o efeito de isômeros da escopolamina como indutora do sono.⁸ São usadas duas amostras (levo e dextro hidrobromido de hyoscyamina).

Usando dados de 10 pacientes que usaram ambas as substâncias e medidas da quantidade adicional de horas de sono observadas, “Student” calcula: (1) a probabilidade dos dados supondo média 0 em cada grupo e (2) a probabilidade dos dados supondo que a diferença das médias é 0.

O primeiro procedimento foi realizado com a medida dos bicos e é chamado teste t de amostra única. Hipotetizando um valor para a média (e.g. $\mu_{bico} = 40mm$; $\mu_{sonoadicional} = 0horas$), calculamos as probabilidades de nossa estimativa.

O segundo procedimento é chamado de teste t de amostras independentes. Hipotetizamos um valor para diferença de médias entre duas populações ($\mu_a - \mu_b = 0$) e calculamos a probabilidade de nossa estimativa. Exemplo prático: existe diferença de peso entre os bicos dos pássaros A e B?

⁸https://atmos.washington.edu/~robwood/teaching/451/student_in_biometrika_vol6_no1.pdf



Aplicação

Retornando ao nosso exemplo de Galápagos, faremos um teste t de amostras independentes.

1. As medidas em A e B são amostras de variáveis aleatórias com distribuição normal.
2. Definimos a hipótese nula e pelo menos uma hipótese alternativa.
 - H_0 : Pássaros das ilhas A e B possuem bicos de tamanho igual.
 - $\mu_a - \mu_b = 0$
- b. H_1 : Os pássaros possuem bicos de tamanho diferentes.

O procedimento é semelhante ao anterior. Calculamos uma quantidade intermediária que segue distribuição t usando a estimativa amostral da diferença e erro padrão associado. Então, podemos especular: qual a probabilidade p de alguém obter nossas observações considerando distribuições de médias iguais ($\mu_a = \mu_b$)? Esse teste infere a probabilidade para as populações de onde saíram as amostras.

Caso p seja menor que um limiar arbitrariamente pré-definido (convencionalmente, 0.05), rejeitamos H_0 . A probabilidade de observarmos os dados é pequena se H_0 for verdade.

Obtemos o valor p somando os valores de probabilidades correspondentes às diferenças obtidas ou valores mais extremos. Caso a diferença entre valores seja grande, o valor da estatística crescerá. Isso implica uma baixa probabilidade de observar aqueles resultados se as amostras fossem semelhantes (vindas da mesma distribuição).

Aplicação com software

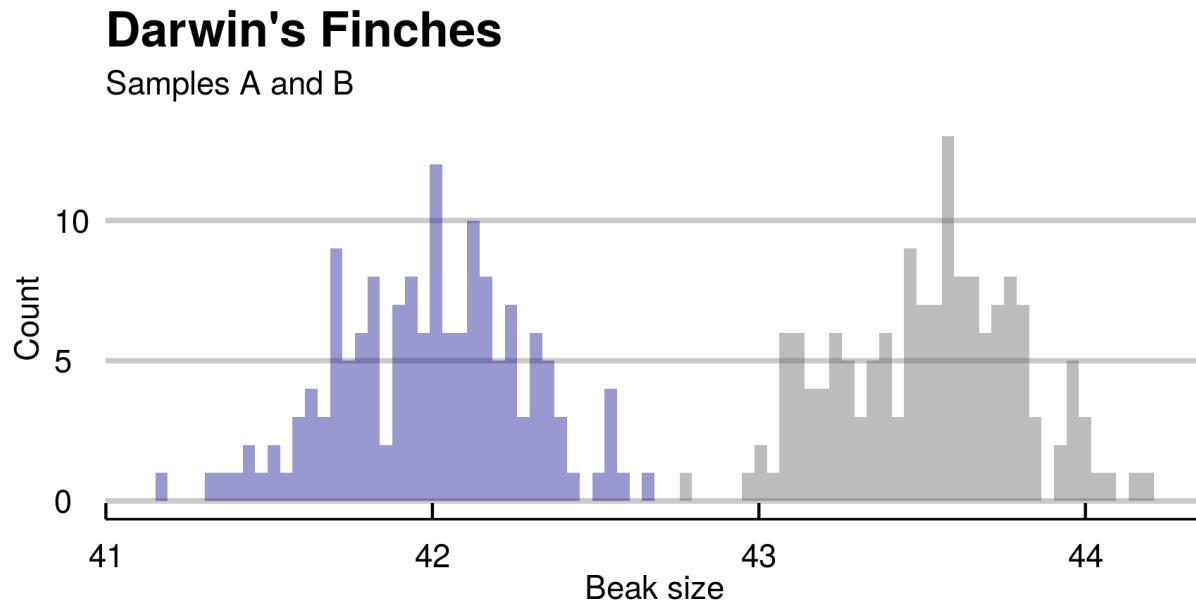
Vamos computar um teste t para 2 amostras independentes. A estatística t é calculada com algumas mudanças: os graus de liberdade são somados e o erro padrão é balanceado entre amostras.

$$t = \frac{X_1 - X_2}{s_p \sqrt{2/n}}; s_p = \sqrt{\frac{s_1^2 + s_2^2}{2}}$$

s_p corresponde ao desvio-padrão balanceado.

Considerando $(n_1 - 1) + (n_2 - 1)$ graus de liberdade, calculamos a estatística t e o valor p correspondente para nossos graus de liberdade. Usando as amostras criadas anteriormente, correspondentes às barras cinza (A) e azul(B), vamos plotar os histogramas.

```
>ggplot(data=galapagos_birds,aes(x=A))+  
  geom_histogram(alpha=my_alpha,bins = my_bins)+  
  geom_histogram(data=galapagos_birds,aes(x=B),fill="dark blue",  
                 alpha=my_alpha,bins = my_bins)+  
  xlab("Beak size")+ylab("Count")  
  ggtitle("Darwin's Finches",subtitle = "Samples A and B")  
  theme_economist_white(gray_bg = F)
```



```
# Ajustes nos dados  
>a <- galapagos_birds$X1  
>b <- galapagos_birds$X2  
>sd_a <- sd(a) #desvio-padrão  
>sd_b <- sd(b)
```

Aqui, ao invés de comparar as estimativas das médias de distribuição t para amostras A e B.

Calculamos a (1) Diferença esperada na vigência da hipótese nula ($diff_{H_0} = 0$), (2) estimativa da diferença ($diff = \mu_A - \mu_B$), graus de liberdade (df) e erro padrão balanceado (se_{pooled}) para a distribuição das diferenças de médias:

```
>expected_diff <- 0  
>mean_diff <- mean(a) - mean(b) #mean difference
```

```

>df_pool <- length(a) + length(b) - 2 # degrees of freedom
>se_pool <- sqrt(((length(a) - 1) * sd_a^2 + (length(b) - 1) * sd_b^2) /
    df_pool) # pooled std. error

```

A estatística t correspondente à diferença observada, considerando uma distribuição t com os parâmetros

```

>t <- (mean_diff - expected_diff) / (se_pool * sqrt(1/length(a) + 1/length(b))) # t-statistic

```

Valor p para hipótese **bicaudal** (resultados extremos considerando a possibilidade de a diferença ser maior ou menor)

```

>p <- 2*pt(-abs(t), df = df_pool) # two-sided p-value

```

Resultados

```

> result <- c(mean_diff, t, p, mean(a), mean(b))
> names(result) <- c("Difference of means", "t", "p-value", "Mean A", "Mean B")
> result
Difference of means      t
  1.533321e+00  4.728513e+01
  p-value          Mean A          Mean B
  1.532661e-140  4.352244e+01  4.198912e+01

```

As funções length, sqrt (raiz quadrada), sd(desvio-padrão) e mean(média) são evidentes e comuns na biblioteca base de outras linguagens.

Obtivemos um valor p significativo ($p < 0.001$) usando $n = 150$. Os graus de liberdade são 149 ($150 - 1$) em cada amostra, sendo 298 ao total.

Alternativamente, para fins práticos, o R possui em sua biblioteca de base uma função para automatizar o processo em 1 linha:

```

> t.test(a,b,var.equal = T)
Two Sample t-test / data: a and b
t = 47.285, df = 298, p-value < 2.2e-16
Alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval: 1.469506 1.597136
Sample estimates:
mean of x mean of y
43.52244 41.98912

```

Estatística t e graus de liberdade apresentados pela biblioteca base do R(base::t.test) são idênticos aos que encontramos realizando o procedimento ‘manualmente’.

Diante do valor p obtido, concluiríamos que a distribuição dos dados como observada é improvável se for verdade a hipótese nula H_0 de que a diferença entre amostras seja 0.

Exemplo de relatório

A diferença estimada entre tamanho médio dos bicos entre amostras A e B foi significativamente ($p < 0.05$) diferente de 0 ($t=47.28$; $df = 298$).

	Amostra A	Amostra B	valor p
Média(μ)	43,52	41,99	<0,001
Desvio-padrão(σ)	0,28	0,28	

Report example

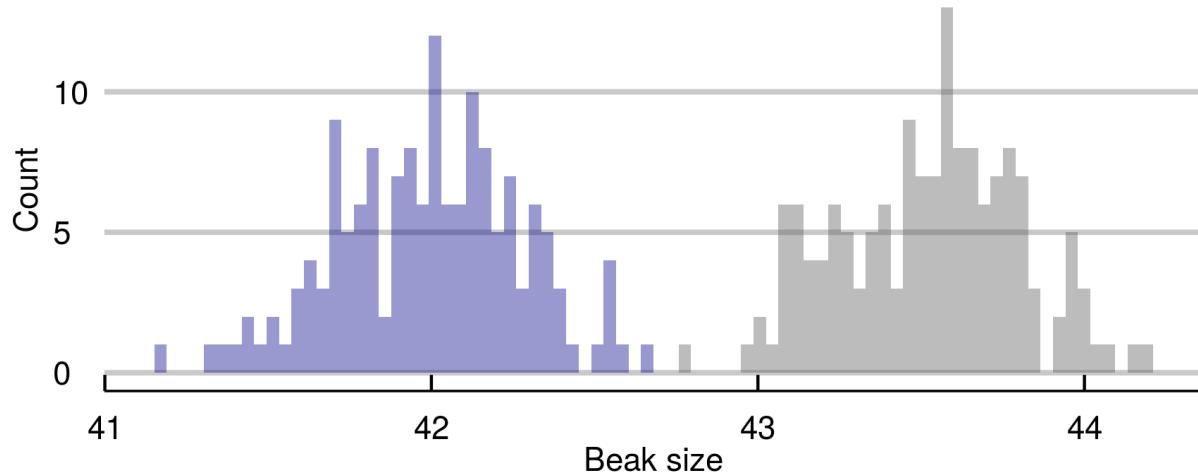
The estimated difference of beak mean sizes among samples A and B was significantly ($p < 0.05$) different

from zero ($t = 47.28$, $df = 298$)

	Amostra A	Amostra B	valor p
Mean (μ)	43,52	41,99	<0,001
Std. Dev. (σ)	0,28	0,28	

Darwin's Finches

Samples A and B



Exercícios

1. Usando o dataset simulado no capítulo:
 - a. Execute teste T para cada par de amostras
 - b. Quais testes apresentam $p < 0.05$?
 - i. Descreva estatística t, graus de liberdade e valor p.
 - * 1. Como são os graus de liberdade dos diversos testes?
 - * 2. Esses valores eram esperados para nossas amostras?
 - ii. Plote histogramas para todos os pares comparados em apenas um painel. Dica: grid.arrange
 - iii. Plote boxplots para uma das comparações.
 - iv. Plote violin plots (geom_violin) para uma das comparações.
 2. Usando o dataset iris
 - a. Escolha duas espécies e duas medidas.
 - b. Execute testes t para ambas as medidas
 - c. Reporte os resultados em uma tabela, incluindo média e desvio-padrão de ambas as medidas nas duas espécies.
 3. Os dados usados por Student para escopolamina estão incluídas na biblioteca de base do R.
 - a. Examine os dados invocando “sleep”: >sleep
 - i. Plote histogramas para as medidas em ambos os grupos
 - ii. Execute um teste t supondo média população zero ($\mu = 0$).
 - iii. Execute um teste t entre amostras, supondo têm a mesma média ($H_0 : \mu_1 = \mu_2$).
 4. Gerando a distribuição t:
 - a. Simule um conjunto de muitas medidas (sugestão: 100,000) a partir de uma distribuição normal($\mu = 0$, $\sigma=1$).
 - b. Retire 200 amostras de n=30 e salve as 200 médias (função sample).
 - c. Divida os valores por pelo erro padrão, σ/\sqrt{n} .
 - d. Retire 200 amostras de uma distribuição t com 29 graus de liberdade (função rt)
 - e. Plote o histograma superposto da distribuição obtida e da distribuição teórica