Lição 10: Teste-t

Na Lição 9, vimos testes estatísticos univariados para variáveis nominais. Nesta lição, veremos um teste que se aplica a variáveis numéricas: o teste-t.

Antes de mais nada, carregue o pacote tidyverse.

```
library(tidyverse)
```

Vamos trabalhar com os dados das vogais pretônicas. Rode as linhas de comando para deixá-lo disponível, definindo como diretório de trabalho a pasta que contém o arquivo Pretonicas.csv.

```
# Definir diretório de trabalho

#setwd()

pretonicas <- read_csv("Pretonicas.csv",
                       col_types = cols(
                  AMOSTRA = col_factor(levels = c("PBSP", "SP2010")),
                  VOGAL = col_factor(levels = c("i", "e", "a", "o", "u
"))
                                        )
                       )
```

Cheque a estrutura desse dataframe.

```
str(pretonicas)

## spec_tbl_df [2,415 × 27] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ PALAVRA      : chr [1:2415] "fazer" "quatorze" "casou" "casado"
...
##  $ Transc.Fon   : chr [1:2415] "f<a>-'zer" "k<a>-'tor-ze" "k<a>-'zo
w" "k<a>-'za-do" ...
##  $ VOGAL        : Factor w/ 5 levels "i","e","a","o",..: 3 3 3 3 3
3 4 3 3 3 ...
##  $ F1           : num [1:2415] 487 686 731 621 845 ...
##  $ F2           : num [1:2415] 1666 1414 1168 1275 1574 ...
##  $ F1.NORM      : num [1:2415] 397 476 494 450 540 ...
##  $ F2.NORM      : num [1:2415] 1517 1386 1258 1314 1469 ...
##  $ CONT.PREC    : chr [1:2415] "f" "k" "k" "k" ...
##  $ CONT.SEG     : chr [1:2415] "z" "t" "z" "z" ...
##  $ VOGAL.SIL.SEG: chr [1:2415] "e" "o" "ow" "a" ...
##  $ F1.SIL.SEG   : num [1:2415] 498 462 529 842 509 ...
##  $ F2.SIL.SEG   : num [1:2415] 2001 1126 1009 1239 2351 ...
##  $ F1.SEG.NORM  : num [1:2415] 328 317 338 433 331 ...
##  $ F2.SEG.NORM  : num [1:2415] 1518 1095 1038 1149 1687 ...
##  $ VOGAL.TONICA : chr [1:2415] "e" "o" "ow" "a" ...
##  $ DIST.TONICA  : num [1:2415] 1 1 1 1 1 1 1 1 1 1 ...
```

```
##  $ ESTR.SIL.PRET: chr [1:2415] “CV” “CV” “CV” “CV” ...
##  $ Begin.Time.s : num [1:2415] 20.4 20.6 33.6 36.5 40.3 ...
##  $ End.Time.s   : num [1:2415] 20.4 20.6 33.6 36.5 40.4 ...
##  $ Duration.ms  : num [1:2415] 19.1 20.2 40.7 25.2 34.7 ...
##  $ AMOSTRA      : Factor w/ 2 levels “PBSP”,”SP2010”: 1 1 1 1 1 1 1
1 1 1 ...
##  $ PARTICIPANTE : chr [1:2415] “MartaS” “MartaS” “MartaS” “MartaS”
...
##  $ SEXO         : chr [1:2415] “feminino” “feminino” “feminino” “fe
minino” ...
##  $ IDADE        : num [1:2415] 32 32 32 32 32 32 32 32 32 32 ...
##  $ IDADE.CHEGADA: num [1:2415] 18 18 18 18 18 18 18 18 18 18 ...
##  $ ANOS.SP      : num [1:2415] 14 14 14 14 14 14 14 14 14 14 ...
##  $ CONTEXTO     : chr [1:2415] “ai aqui j\u0087 tem treze ano vai f
azer quatorze” “ai aqui j\u0087 tem treze ano vai fazer quatorze” “a\u
0092 depois ele voltou a gente casou e viemos” “que l\u0087 voc\u0090
s\u0097 podia sair se fosse casado n\u008e se fosse pra” ...
##  - attr(*, “spec”)=
##   .. cols(
##   ..   PALAVRA = col_character(),
##   ..   Transc.Fon = col_character(),
##   ..   VOGAL = col_factor(levels = c(“i”, “e”, “a”, “o”, “u”), orde
red = FALSE, include_na = FALSE),
##   ..   F1 = col_double(),
##   ..   F2 = col_double(),
##   ..   F1.NORM = col_double(),
##   ..   F2.NORM = col_double(),
##   ..   CONT.PREC = col_character(),
##   ..   CONT.SEG = col_character(),
##   ..   VOGAL.SIL.SEG = col_character(),
##   ..   F1.SIL.SEG = col_double(),
##   ..   F2.SIL.SEG = col_double(),
##   ..   F1.SEG.NORM = col_double(),
##   ..   F2.SEG.NORM = col_double(),
##   ..   VOGAL.TONICA = col_character(),
##   ..   DIST.TONICA = col_double(),
##   ..   ESTR.SIL.PRET = col_character(),
##   ..   Begin.Time.s = col_double(),
##   ..   End.Time.s = col_double(),
##   ..   Duration.ms = col_double(),
##   ..   AMOSTRA = col_factor(levels = c(“PBSP”, “SP2010”), ordered =
FALSE, include_na = FALSE),
##   ..   PARTICIPANTE = col_character(),
##   ..   SEXO = col_character(),
##   ..   IDADE = col_double(),
##   ..   IDADE.CHEGADA = col_double(),
##   ..   ANOS.SP = col_double(),
##   ..   CONTEXTO = col_character()
##   .. )
##  - attr(*, “problems”)=<externalptr>
```

Nesta lição, vamos trabalhar apenas com os dados da vogal /e/ pretônica. Por vezes será nos dados apenas de paraibanos ou de paulistanos, por vezes será para ambos os grupos. Crie então um subconjunto de dados chamado `PBSP_e`, com os dados da VOGAL /e/ na AMOSTRA PBSP.

```
PBSP_e <- filter(pretonicas, VOGAL == "e" & AMOSTRA == "PBSP")
```

Crie também um subconjunto de dados chamado `SP2010_e`, com os dados da VOGAL /e/ na AMOSTRA SP2010.

```
SP2010_e <- filter(pretonicas, VOGAL == "e" & AMOSTRA == "SP2010")
```

E crie um subconjunto de dados da vogal /e/ (com dados de PBSP e SP2010), chamado `VOGAL_e`, que vamos usar mais adiante.

```
VOGAL_e <- filter(pretonicas, VOGAL == "e")
```

Vamos agora fazer dois histogramas das medidas de F1.NORM (=altura das vogais) em uma mesma figura para compará-los. Complete a linha de comando neste ponto do *script*, substituindo, com calma, os valores df, VAR e VAR.

O dataframe é `VOGAL_e` – que contém dados de ambas as amostras –, o valor da variável em x é F1.NORM e da variável em `facet_grid()` é AMOSTRA, para que cada faceta contenha os dados de um dos grupos. A variável AMOSTRA vem antes de ~ para ordenar os gráficos um em cima do outro, e não lado a lado. O resultado se encontra na Figura 10.1.

```
ggplot(VOGAL_e, aes(x = F1.NORM)) +
  geom_histogram() +
  facet_grid(AMOSTRA ~ .)
```

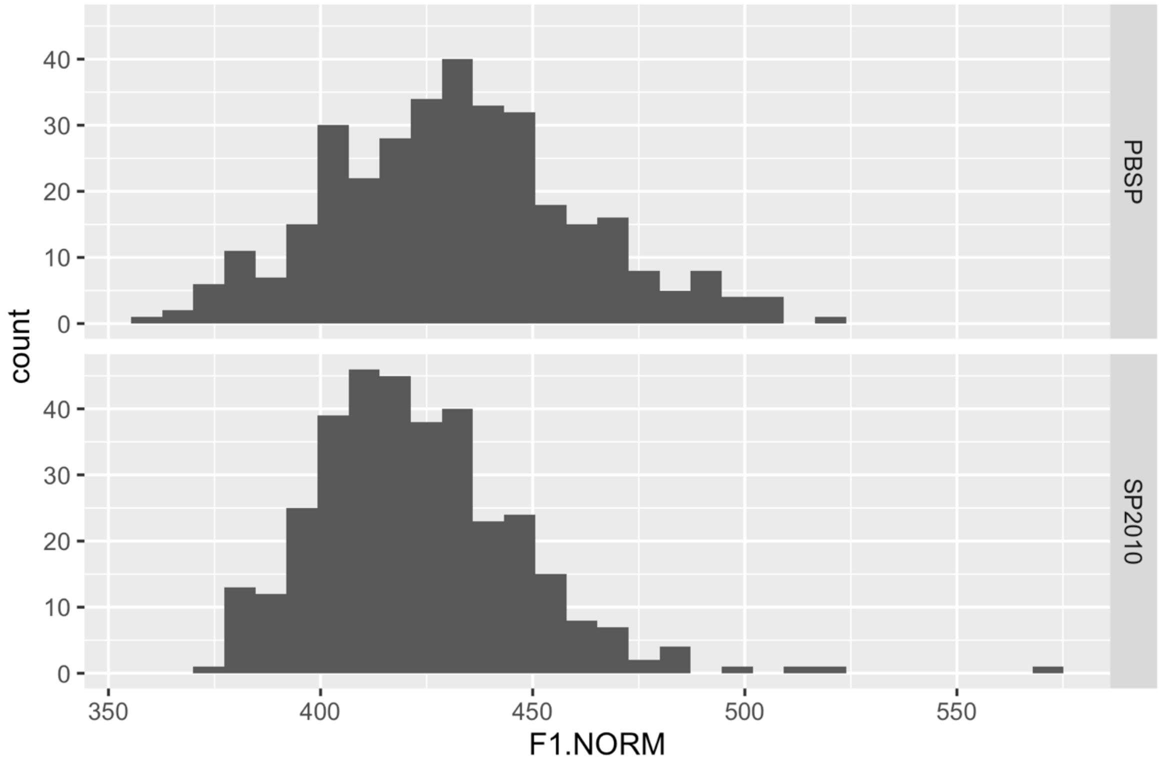

Figura 10.1: Distribuição das medidas de F1 normalizado da vogal /e/ nas amostras PBSP e SP2010. Fonte: própria.

Na figura, as medidas de F1.NORM para os paraibanos parecem ser, em geral, mais altas e mais dispersas do que para os paulistanos. Calcule a média e o desvio padrão de F1.NORM para PBSP e SP2010, e guarde os resultados num novo dataframe chamado `estatisticas`.

```
estatisticas <- VOGAL_e %>%
  group_by(AMOSTRA) %>%
  summarize(media_F1 = mean(F1.NORM),
            desvio_F1 = sd(F1.NORM)) %>%
  print()

## # A tibble: 2 × 3
##   AMOSTRA media_F1 desvio_F1
##   <fct>      <dbl>     <dbl>
## 1 PBSP        432.      29.5
## 2 SP2010      423.      24.9
```

Compare as médias de F1.NORM entre paraibanos e paulistanos: qual é maior?

- a média de F1.NORM de paraibanos
- a média de F1.NORM de paulistanos

Compare os valores de desvio padrão de F1.NORM entre paraibanos e paulistanos: qual é maior?

- o desvio padrão de F1.NORM de paraibanos
- o desvio padrão de F1.NORM de paulistanos

Como vimos na Lição 7, outro modo de visualizar a dispersão dos dados é por meio de boxplots. Faça um boxplot das medidas de F1.NORM por AMOSTRA no subconjunto de dados da VOGAL /e/ (Figura 10.2).

```
VOGAL_e %>%
  ggplot(., aes(x = AMOSTRA, y = F1.NORM)) +
  geom_boxplot(notch = TRUE) +
  scale_y_reverse()
```

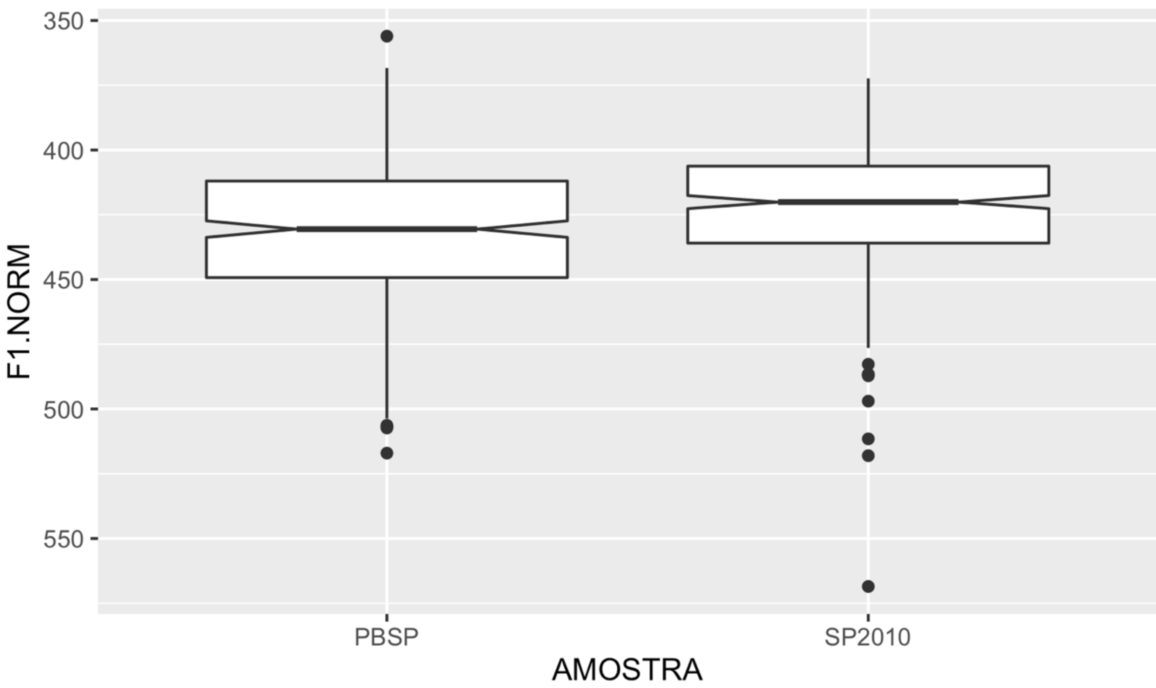

Figura 10.2: Boxplots das medidas de F1 normalizado da vogal /e/ nas amostras PBSP e SP2010. Fonte: própria.

Até aqui, refizemos os passos da Lição 7, pois esta teria sido sua trajetória de análise: começar com cálculos de medidas estatísticas básicas e visualização dos dados para estabelecer hipóteses. A partir dos histogramas e do boxplot, das medidas de média e desvio padrão, podemos querer verificar se as vogais médias pretônicas /e/ para paraibanos são significativamente mais baixas do que as dos paulistanos. Pelos entalhes nos boxplots, temos uma boa evidência de que as médias são significativamente diferentes; cabe agora um teste estatístico para reforçar essa evidência.

O teste-t – no R, `t.test()` – compara médias e variâncias de uma distribuição com uma distribuição esperada ou com a distribuição de outro grupo. Ele tem, contudo, um requisito: a distribuição dos dados deve seguir a distribuição normal (o formato da curva de sino), ou deve ser aplicada a uma amostra com grande número de dados. Caso a distribuição não seja normal ou a amostra seja pequena, aplica-se uma variante do teste-t, chamada Teste de Wilcoxon – no R, `wilcox.test()`. Aqui, vamos seguir a primeira condição – seguir a distribuição normal ou não. O teste-t é um teste paramétrico e o teste de Wilcoxon é um teste não paramétrico.

Olhando para os histogramas (volte a eles com as flechinhas azuis), você acha que a distribuição das medidas de F1.NORM da vogal /e/ para paraibanos segue a distribuição normal?

- sim
- não
- não sei!

E para os paulistanos?

- sim
- não
- não sei!

A inspeção gráfica é um bom primeiro passo, mas deixa margem para muita dúvida. O teste de Shapiro permite obter uma medida mais acurada do quanto a distribuição se aproxima da distribuição normal. No R, ele é feito com a função `shapiro.test()`, que se aplica a um vetor de dados numéricos. Aplique a função aos dados de F1.NORM do subconjunto `PBSP_e`.

```
shapiro.test(PBSP_e$F1.NORM)

##
##  Shapiro-Wilk normality test
##
```

```
## data:  PBSP_e$F1.NORM
## W = 0.99263, p-value = 0.09217
```

E aplique-a também aos de F1.NORM do subconjunto `SP2010_e`.

```
shapiro.test(SP2010_e$F1.NORM)

##
##  Shapiro-Wilk normality test
##
## data:  SP2010_e$F1.NORM
## W = 0.94706, p-value = 8.416e-10
```

Para aplicação do teste de Shapiro, usamos a função da instalação base do R. Só por curiosidade, também seria possível fazer isso por meio do dplyr; no entanto, como o resultado do teste de Shapiro é uma lista, também seria necessário guardar o resultado em uma lista (não um dataframe). A linha de comando neste ponto do *script*, já pronta, mostra como isso poderia ser feito, aplicando-se a função `list()` sobre o teste de Shapiro. Rode-a para ver o resultado.

```
shapiro <- pretonicas %>%
  filter(VOGAL == “e”) %>%
  group_by(AMOSTRA) %>%
  summarize(res = list(shapiro.test(F1.NORM)))
```

O resultado pode ser acessado por `shapiro$res`, em que `shapiro` é o objeto criado pelo comando acima e `res` é onde guardamos o resultado do teste de Shapiro. Rode essa linha de comando.

```
shapiro$res

## [[1]]
##
##  Shapiro-Wilk normality test
##
## data:  F1.NORM
## W = 0.99263, p-value = 0.09217
##
##
## [[2]]
##
##  Shapiro-Wilk normality test
##
## data:  F1.NORM
## W = 0.94706, p-value = 8.416e-10
```

O teste de Shapiro testa a hipótese *nula* que os dados vêm de uma distribuição normal. Isso significa que um $p < 0{,}05$ indica que a distribuição provavelmente *não* é normal. Veja o resultado do teste de Shapiro nos dados de `PBSP_e`. A que conclusão o pesquisador pode chegar?

- a distribuição é normal
- a distribuição não é normal

Qual teste então pode ser aplicado aos dados de `PBSP_e`?

- `t.test()`
- `wilcox.test()`

E sobre o resultado do teste de Shapiro nos dados de `SP2010_e`?

- a distribuição é normal
- a distribuição não é normal

Qual teste então pode ser aplicado aos dados de `SP2010_e`?

- `t.test()`
- `wilcox.test()`

Façamos isso. Vamos aplicar o teste-t à distribuição de F1.NORM dos dados de `PBSP_e`. Em sua forma mais simples, assim como no teste de proporções (Lição 9), pode-se verificar se a média dos dados observados é igual a uma média esperada. Neste caso, a função `t.test()` tem dois argumentos: (i) o vetor de dados numéricos com os dados observados; e (ii) o argumento mu, que especifica o valor esperado. Suponhamos que um estudo prévio sobre vogais médias pretônicas na fala de paraibanos tenha indicado uma média de 440 Hz para a vogal /e/; vamos usar esse valor para comparação.

```
t.test(PBSP_e$F1.NORM, mu = 440)

##
##  One Sample t-test
##
## data:  PBSP_e$F1.NORM
## t = -5.1475, df = 339, p-value = 4.482e-07
## alternative hypothesis: true mean is not equal to 440
## 95 percent confidence interval:
##  428.6064 434.9065
## sample estimates:
## mean of x
##  431.7564
```

Vejamos o resultado de um teste-t. Primeiro o R informa o teste que foi realizado (teste-t de uma amostra) e, logo em seguida, o conjunto de dados (`PBSP_e$F1.NORM`). A terceira linha informa a medida t, os graus de liberdade e o valor de significância. Isso é paralelo ao teste de qui-quadrado, visto na Lição 9, com a diferença que, em vez de uma tabela de distribuição de qui-quadrado, consulta-se uma tabela de distribuição t. Como variáveis numéricas podem ter um número muito maior de graus de liberdade, ela não é mostrada aqui (como na última lição), mas você pode facilmente encontrá-la na Internet.

Logo em seguida o R enuncia a hipótese alternativa e o intervalo de confiança da distribuição (*default* = 95%). Se quisesse mudar o intervalo de confiança, bastaria especificar o argumento conf.level. Veja que o intervalo de confiança do teste acima (428,6 Hz a 434,9 Hz) não contém a média esperada (mu = 440), de modo que a hipótese nula (= a média é igual a 440) pode ser rejeitada e a hipótese alternativa (= a média não é igual a 440) pode ser acatada. Por último, o R mostra a média da amostra (431,7 Hz), que é justamente o que havíamos calculado acima para as medidas de F1.NORM da vogal /e/ entre os paraibanos.

Assim como o teste de qui-quadrado, a aplicação do teste-t é extremamente simples. Mas não importa apenas saber como aplicá-lo, mas também *quando* aplicá-lo (a uma variável numérica) e *de onde saíram* as medidas estatísticas.

Para sua curiosidade, o valor-*t* é calculado de acordo com a fórmula na sequência. $\bar{X}$ é o valor da média da amostra (para nós, `estatisticas$media_F1[1]`) e μ é a média esperada (aqui, 440); S é o valor de desvio padrão da amostra (`estatisticas$desvio_F1[1]`) e n é o número de observações.

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

No R, como seria escrita a fórmula da figura de acordo com nossos dados?

- `(estatisticas$media_F1[1] - 440) / (estatisticas$desvio_F1[1] / sqrt(length(PBSP_e$F1.NORM)))`
- `(estatisticas$media_F1[1] - 440) / (estatisticas$desvio_F1[1] / sqrt(PBSP))`

- `(440 - estatisticas$media_F1[1]) / (estatisticas$desvio_F1[1] / length(PBSP_e$F1.NORM))`
- `estatisticas$media_F1[1] - 440 / estatisticas$desvio_F1[1] / sqrt(length(PBSP_e$F1.NORM))`

Execute a linha de comando `(estatisticas$media_F1[1] - 440) / (estatisticas$desvio_F1[1] / sqrt(length(PBSP_e$F1.NORM)))` para calcular o valor-*t*.

```
(estatisticas$media_F1[1] - 440) / (estatisticas$desvio_F1[1] / sqrt(l
ength(PBSP_e$F1.NORM)))

## [1] -5.147498
```

Compare este valor com o valor-*t* calculado no teste-t acima. É o mesmo, certo? Quanto aos graus de liberdade, ele é o número de observações - 1. Como há 340 dados de F1.NORM da vogal /e/ para paraibanos, df = 339. (Para outros tipos de teste-t que veremos adiante, o cálculo do valor-*t* e dos graus de liberdade é um pouco diferente. Para mais informações, veja http://www.statisticshowto.com/t-test/.)

Assim como no teste de proporções, é possível estabelecer hipóteses bi ou unidirecionais. Para as vogais médias pretônicas /e/ na fala de migrantes paraibanos, a expectativa é que esses valores se aproximem mais das medidas de F1 dos paulistanos – vogais relativamente mais altas, ou F1 mais baixo. Neste caso, o pesquisador poderia estabelecer uma hipótese unidirecional: a média de F1.NORM para paraibanos em São Paulo é mais baixa do que a média de F1.NORM para paraibanos não migrantes. No R, como esta hipótese seria colocada da função `t.test()`?

- `t.test(PBSP_e$F1.NORM, mu = 440, alternative = “less”)`
- `t.test(PBSP_e$F1.NORM, mu = 440, alternative = “greater”)`
- `t.test(PBSP_e$F1.NORM, mu = 440, alternative = “two.sided”)`

Faça agora o teste unidirecional, com alternative = “less”.

```
t.test(PBSP_e$F1.NORM, mu = 440, alternative = “less”)

##
##  One Sample t-test
##
## data:  PBSP_e$F1.NORM
## t = -5.1475, df = 339, p-value = 2.241e-07
```

```
## alternative hypothesis: true mean is less than 440
## 95 percent confidence interval:
##      -Inf 434.3978
## sample estimates:
## mean of x
##  431.7564
```

Evidentemente, a escolha de um teste uni ou bidirecional depende de suas questões de pesquisa, hipóteses e expectativas. O conhecimento necessário aí é a sua formação em Linguística!

Vamos agora aplicar o teste para as vogais dos paulistanos. Lembre-se que, neste caso, como a distribuição não é normal, devemos aplicar o teste de Wilcoxon, cuja função no R é `wilcox.test()` – atenção para o nome da função, que é diferente do nome do teste! Como primeiro argumento, use o vetor de dados de F1.NORM de `SP2010_e`, e como segundo argumento uma média hipotética mu = 410.

```
wilcox.test(SP2010_e$F1.NORM, mu = 410)

##
##  Wilcoxon signed rank test with continuity correction
##
## data:  SP2010_e$F1.NORM
## V = 46146, p-value < 2.2e-16
## alternative hypothesis: true location is not equal to 410
```

O resultado da função `wilcox.test()` é mais resumido, mas a interpretação é semelhante. O teste nos informa que as médias não são iguais, com probabilidade de se ter observado tal distribuição em caso de a hipótese nula ser verdadeira abaixo de 0,001. Para esta função, o *default* é não exibir os valores do intervalo de confiança e a média da amostra; caso queira vê-los, é necessário especificar o argumento conf.int = T. Faça isso agora a partir da linha de comando acima.

```
wilcox.test(SP2010_e$F1.NORM, mu = 410, conf.int = T)

##
##  Wilcoxon signed rank test with continuity correction
##
## data:  SP2010_e$F1.NORM
## V = 46146, p-value < 2.2e-16
## alternative hypothesis: true location is not equal to 410
## 95 percent confidence interval:
##  418.9230 424.0106
## sample estimates:
```

```
## (pseudo)median
##       421.4194
```

Note, contudo, que como se trata de uma distribuição não normal, as estimativas são menos precisas. Assim como na função `t.test()`, você também poderia estabelecer uma hipótese unidirecional com o argumento alternative, e mudar o intervalo de confiança com o argumento conf.level.

Nos exemplos acima, empregamos uma média esperada mu hipoteticamente vinda de algum estudo prévio, e analisamos os dados de `PBSP_e` e `SP2010_e` separadamente. Mas as funções `t.test()` e `wilcox.test()` também podem comparar a média e a variância/desvio padrão de dois grupos, como é justamente o nosso caso. Quando se quer comparar dois grupos, o primeiro argumento de ambas as funções é uma fórmula no formato y ~ x, em que y é a variável dependente e x é a variável independente. Memorize essa notação, pois ela será vista novamente mais adiante no curso. Aqui, queremos comparar F1.NORM ~ AMOSTRA. O segundo argumento é o conjunto de dados – para nós, `VOGAL_e`.

Para decidir qual dos testes aplicar (teste-t ou teste de Wilcoxon), façamos um teste de Shapiro sobre os dados de F1.NORM do subconjunto `VOGAL_e`.

```
shapiro.test(VOGAL_e$F1.NORM)

##
##  Shapiro-Wilk normality test
##
## data:  VOGAL_e$F1.NORM
## W = 0.97831, p-value = 1.505e-08
```

O teste nos informa que a distribuição dos dados não é normal e, portanto, é mais recomendado aplicar o teste de Wilcoxon. Essa decisão também poderia ter sido tomada pelo simples fato de que sabemos que a distribuição de uma das amostras (a dos paulistanos) não tem distribuição normal.

Digite então `wilcox.test(F1.NORM ~ AMOSTRA, data = VOGAL_e, conf.int = T)`.

```
wilcox.test(F1.NORM ~ AMOSTRA, data = VOGAL_e, conf.int = T)

##
##  Wilcoxon rank sum test with continuity correction
```

```
##
## data:  F1.NORM by AMOSTRA
## W = 70330, p-value = 9.2e-06
## alternative hypothesis: true location shift is not equal to 0
## 95 percent confidence interval:
##   5.181038 13.209983
## sample estimates:
## difference in location
##               9.148978
```

O resultado de um teste de Wilcoxon de duas amostras (= dois grupos) segue a mesma estrutura que já vimos acima; no entanto, uma diferença é que a estatística gerada é um valor W. Outra é o fato que a medida da comparação não são as médias em si, mas sim se a diferença entre elas é igual ou não a zero. É nesse sentido que se deve entender a hipótese alternativa (true location shift is not equal to 0) e os valores do intervalo de confiança. No teste feito acima, o intervalo de confiança, de 5,18 a 13,21, não contém zero, o que nos leva à rejeição da hipótese nula, com $p < 0{,}001$. A medida de estimativa ao final (9,15) é o valor da diferença estimada entre os grupos.

Veja que a estimativa da diferença entre as médias é um pouco diferente da diferença entre a média de PBSP e de SP2010. Digite `estatisticas$media_F1[1] - estatisticas$media_F1[2]` para comparar.

```
estatisticas$media_F1[1] - estatisticas$media_F1[2]
```

```
## [1] 8.790398
```

A diferença observada entre as médias é um pouco menor. Como os dados advêm de uma distribuição não normal, o teste de Wilcoxon ajustou a estimativa da diferença.

Por curiosidade, vamos aplicar a função `t.test()` ao mesmo conjunto de dados, para familiarização com o resultado fornecido pelo R. Digite `t.test(F1.NORM ~ AMOSTRA, data = VOGAL_e)`.

```
t.test(F1.NORM ~ AMOSTRA, data = VOGAL_e)
```

```
##
##  Welch Two Sample t-test
##
## data:  F1.NORM by AMOSTRA
## t = 4.2125, df = 660.96, p-value = 2.877e-05
## alternative hypothesis: true difference in means between group PBSP
and group SP2010 is not equal to 0
## 95 percent confidence interval:
```

```
##   4.692926 12.887869
## sample estimates:
##   mean in group PBSP mean in group SP2010
##             431.7564            422.9660
```

O teste-t de duas amostras também mede a diferença entre as médias e testa a hipótese nula que a diferença é igual a zero. Os valores do intervalo de confiança estimam que a diferença entre as medidas de F1.NORM para paraibanos e paulistanos está entre 4,7 Hz e 12,9 Hz, intervalo que não inclui zero – daí a diferença entre as médias ser significativa. Ao final, o R apresenta as médias do grupo 1 e do grupo 2.

Assim como fizemos no teste de uma amostra, podemos estabelecer uma hipótese unidirecional. Para as vogais pretônicas /e/ de paraibanos e de paulistanos, pode-se esperar que a média de F1.NORM de paraibanos seja maior que a de paulistanos, ou que a média de F1.NORM de paulistanos seja menor que a de paraibanos. Como você acha que o R determina a direção da comparação?

- pela ordem alfabética dos níveis
- pela ordem dos níveis no dataframe
- é uma ordem aleatória, que muda a cada teste

Exato! Pela ordem dos níveis, que pode ter sido definida pelo usuário ao importar os dados com `read_csv()`, ou, caso não tenham sido feitas modificações, a ordem em que os níveis aparecem no dataframe. É importante, portanto, checar a ordem dos níveis de AMOSTRA para bem interpretar o resultado do teste-t. Aplique a função `levels()` a esse vetor para termos certeza da ordem dos níveis de AMOSTRA.

```
levels(VOGAL_e$AMOSTRA)
## [1] “PBSP”   “SP2010”
```

Pela ordem no dataframe, o R vai comparar PBSP com SP2010, e não o contrário. Como esperamos que a média de F1.NORM seja maior para PBSP do que para SP2010, devemos estabelecer alternative = “greater” nas funções `wilcox.test()` ou `t.test()`. Copie a última linha de comando em que usamos a função `wilcox.test()`, cole-a no script e acrescente o argumento alternative = “greater”.

```
wilcox.test(F1.NORM ~ AMOSTRA, data = VOGAL_e, conf.int = T, alternati
ve = "greater")

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  F1.NORM by AMOSTRA
## W = 70330, p-value = 4.6e-06
## alternative hypothesis: true location shift is greater than 0
## 95 percent confidence interval:
##  5.834992      Inf
## sample estimates:
## difference in location
##               9.148978
```

Fizemos isso apenas para que você tenha em conta que sempre pode realizar testes bi ou unidirecionais. Na verdade, o resultado significativo do teste unidirecional era previsível. Lembra-se do exemplo da moeda (Lição 8)? Lá vimos que o teste bidirecional (como era o caso do juiz imparcial) é mais rigoroso para rejeição da hipótese nula do que o teste unidirecional (como era o seu caso). Se a hipótese nula foi rejeitada num teste bidirecional, ela também será no teste unidirecional que prevê a direção dos dados observados.

Um terceiro tipo de teste-t/teste de Wilcoxon é de amostras pareadas. Aqui, permita-me usar um novo conjunto de dados, pois os dados de vogais pretônicas de paraibanos e paulistanos não são adequados para um teste pareado. Vamos usar um exemplo retirado do curso Statistical Inference, do swirl – outro curso que recomendo fazer –, mas que não tem nada a ver com Linguística, infelizmente.

Um exemplo linguístico em que se poderia aplicar um teste-t pareado seria testar a eficácia de determinado método de ensino. Digamos que um grupo de 10 alunos de um curso de inglês fizesse uma prova, depois passasse por uma aula, e em seguida fizesse nova prova sobre o mesmo assunto. O pesquisador quer comparar se o desempenho do aluno – medido pelas provas – melhorou depois da aula. O pesquisador tem em mãos 20 notas de provas, mas cada uma dessas notas não é uma observação independente vinda da população. É mais provável que alunos que já tinham tido uma nota boa na primeira prova continuem com uma boa nota na segunda prova. Cada nota da prova 1 está associada a uma nota da prova 2. É aí que cabe fazer uma comparação de médias em que os dados são pareados.

No conjunto de dados não linguísticos com que vamos trabalhar, tem-se os resultados de um estudo médico sobre o sono, que testa o efeito de duas drogas que foram ministradas a 10 pacientes. Para mais informações, visite https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/sleep.html. Este conjunto de dados está disponível no dataframe sleep, que faz parte da instalação base do R. Digite `sleep` para vê-lo no Console.

```
sleep
```

```
##    extra group ID
## 1    0.7     1  1
## 2   -1.6     1  2
## 3   -0.2     1  3
## 4   -1.2     1  4
## 5   -0.1     1  5
## 6    3.4     1  6
## 7    3.7     1  7
## 8    0.8     1  8
## 9    0.0     1  9
## 10   2.0     1 10
## 11   1.9     2  1
## 12   0.8     2  2
## 13   1.1     2  3
## 14   0.1     2  4
## 15  -0.1     2  5
## 16   4.4     2  6
## 17   5.5     2  7
## 18   1.6     2  8
## 19   4.6     2  9
## 20   3.4     2 10
```

A primeira coluna mostra o resultado obtido após a aplicação das drogas; a segunda coluna indica qual foi a droga (grupo 1 ou grupo 2); e a terceira coluna identifica o paciente (de 1 a 10). Se fizermos simplesmente a média das medições da droga 1 e da droga 2, não podemos saber efetivamente se cada paciente melhorou ou não.

Para fazer um teste-t pareado, precisamos de dois vetores de mesma extensão e mesma ordem, para que os pares estejam devidamente ordenados. Crie primeiro um vetor chamado `g1`, com as medições da primeira coluna, das linhas 1 a 10.

```
g1 <- sleep[1:10, 1]
```

Crie agora um vetor chamado `g2`, com as medições da primeira coluna, das linhas 11 a 20.

```
g2 <- sleep[11:20, 1]
```

Aplique o teste de Shapiro a `g1` para verificar se a distribuição é normal.

```
shapiro.test(g1)

##
##  Shapiro-Wilk normality test
##
## data:  g1
## W = 0.92581, p-value = 0.4079
```

Aplique o teste de Shapiro a `g2` para verificar se a distribuição é normal.

```
shapiro.test(g2)

##
##  Shapiro-Wilk normality test
##
## data:  g2
## W = 0.9193, p-value = 0.3511
```

Como ambas as amostras têm distribuição normal, podemos aplicar `t.test()`. O teste-t pareado toma como argumentos os dois vetores e mais o argumento paired = T. Digite então `t.test(g1, g2, paired = T)` para ver o resultado.

```
t.test(g1, g2, paired = T)

##
##  Paired t-test
##
## data:  g1 and g2
## t = -4.0621, df = 9, p-value = 0.002833
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.4598858 -0.7001142
## sample estimates:
## mean of the differences
##                   -1.58
```

O R indica que a diferença é significativa (p = 0,0028), e que a diferença g1 - g2 é em média -1,58. Para comparar, veja o resultado de um teste-t caso não se especificasse que as amostras são pareadas. Digite `t.test(g1, g2)` – o que seria o mesmo que `t.test(extra ~ group, data = sleep)`.

```
t.test(g1, g2)
```

```
##
##  Welch Two Sample t-test
##
## data:  g1 and g2
## t = -1.8608, df = 17.776, p-value = 0.07939
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -3.3654832  0.2054832
## sample estimates:
## mean of x mean of y
##      0.75      2.33
```

Neste caso, o intervalo de confiança vai de -3,37 a 0,21, um intervalo que inclui o zero, de modo que a diferença entre as amostras poderia ter sido nula. A significância reflete isso: o valor-*p* calculado está acima de 0,05. Veja que deixar de especificar o fato de que os dados são pareados pode mudar o resultado do teste: no primeiro caso, o teste indicou que houve diferença, e no segundo caso não.

Ao reportar os resultados de um teste-t ou teste de Wilcoxon, é importante indicar qual medida estatística foi gerada (t ou W), os graus de liberdade, o valor-*p*, o tipo de teste (uma amostra, duas amostras, duas amostras pareadas), se o teste foi uni ou bicaudal, junto com os valores de médias e de erro padrão de cada distribuição. Não há uma função específica na instalação base do R para calcular o erro padrão, mas ele é facilmente calculável. Como vimos na Lição 6, essa medida é o valor do desvio padrão dividido pela raiz quadrada do número de observações – no R: `sd(x)/sqrt(length(x))`. O resultado do teste de Wilcoxon sobre a diferença entre a média de F1.NORM entre as amostras pode ser assim reportado: "Um teste de Wilcoxon bicaudal foi feito para comparar as médias de F1 normalizadas da vogal pretônica /e/ nas amostras de paraibanos e de paulistanos. Em média, a vogal /e/ de paraibanos é significativamente mais baixa (média = 431,7Hz, erro padrão = 1,60) do que a de paulistanos (média = 423Hz, erro padrão = 1,33), W = 70,330, $p < 0{,}001$.".

Vamos fazer uma pequena revisão antes de concluir esta lição. Quando se aplica um teste-t ou teste de Wilcoxon?

- quando se tem uma variável fatorial e se quer comparar as médias de dois grupos

- quando se tem uma variável nominal e se quer comparar as médias de dois grupos
- quando se tem uma variável numérica e se quer comparar as médias de dois grupos

Quando se aplica um teste-t (e não um teste de Wilcoxon)?

- quando a distribuição dos dados segue a distribuição normal
- quando a distribuição dos dados não segue a distribuição normal
- quando a distribuição dos dados segue a distribuição binomial
- quando a distribuição dos dados segue a distribuição de qui-quadrado

Além da inspeção de histogramas, como se pode determinar se uma distribuição é normal?

- por meio do teste-t, com a função `t.test()`
- por meio do teste de Shapiro, com a função `shapiro.test()`
- por meio do teste de Wilcoxon, com a função `wilcox.test()`

Quando se aplica um teste-t/teste de Wilcoxon de uma amostra?

- quando se quer comparar uma distribuição com uma média conhecida
- quando se quer comparar as distribuições entre dois grupos diferentes de nossos dados
- quando se quer comparar as distribuições entre dois grupos pareados de nossos dados

Quando se aplica um teste-t/teste de Wilcoxon de duas amostras?

- quando se quer comparar uma distribuição com uma média conhecida
- quando se quer comparar as distribuições entre dois grupos diferentes de nossos dados
- quando se quer comparar as distribuições entre dois grupos pareados de nossos dados

Quando se aplica um teste-t/teste de Wilcoxon pareado?

- quando se quer comparar uma distribuição com uma média conhecida
- quando se quer comparar as distribuições entre dois grupos diferentes de nossos dados
- quando se quer comparar as distribuições entre dois grupos pareados de nossos dados

Para saber mais

Recomendo a leitura do capítulo 5 de Dalgaard (2008) e do capítulo 5 de Levshina (2015) para reforçar os conceitos aqui aprendidos.

Exercícios

Para estes exercícios, você vai precisar do arquivo DadosRT-percepcao.csv. Os dados dessa planilha foram adaptados de um experimento de percepções sociolinguísticas sobre a pronúncia variável de /r/ em coda silábica como tepe ou como retroflexo. Todos os participantes eram habitantes da cidade de São Paulo (nativos e não nativos). A eles foram tocados quatro pequenos excertos de áudio, em que ouviam uma pessoa falando; sua tarefa era a de imaginar quem era o falante e responder um questionário sobre as características que imaginaram. Sete dessas características deveriam ser assinaladas dentro de uma escala de dez pontos, em que um extremo significa "pouco" e o outro extremo significa "muito": extrovertido; escolarizado; inteligente; formal; amigável; paulistano; ter sotaque. Havia também outra escala, em que um extremo significava morar num bairro mais periférico e o outro significava morar num bairro mais central. Os participantes foram divididos em dois grupos. Metade dos participantes ouviram os falantes Antonio e Luisa usando o retroflexo, e Daniela e Paulo usando o tepe. A outra metade ouviu os falantes Antonio e Luisa usando o tepe, e Daniela e Paulo usando o retroflexo. O interesse geral da pesquisa foi o de verificar se as percepções sobre um mesmo falante seriam alteradas ao serem ouvidos com o tepe ou com o retroflexo. Para mais detalhes, ver Oushiro (2015, 2019).

1. Carregue os dados de DadosRT-percepcao.csv em um dataframe chamado `dados`. Defina as colunas VARIANTE.OUVIDA e FALANTE como factor.
2. Cheque a estrutura dos `dados` para ver se foram carregados corretamente.
3. Aplique a função `View()` para visualizar a planilha de dados e se familiarizar com ela. Note, em especial, as colunas de variáveis numéricas: EXTROVERSAO, ESCOLARIZACAO, INTELIGENCIA etc. A planilha também contém variáveis que identificam a variante que foi ouvida, quem era o falante, e características sociais dos ouvintes/participantes.
4. Carregue o pacote tidyverse.
5. Faça um subconjunto de dados chamado `dados_R` para quando se ouviu a variante retroflexa. Veja o resultado de `str()` e de `View()` para saber como foram nomeadas a variável e a variante.
6. Faça um subconjunto de dados chamado `dados_T` para quando se ouviu a variante tepe. Veja o resultado de `str()` e de `View()` para saber como foram nomeadas a variável e a variante.
7. Faça um gráfico com dois histogramas da distribuição de "graus de paulistanidade" (PAULISTANIDADE) atribuídos aos falantes quando se ouviu o retroflexo e quando se ouviu o tepe. Para tanto, a partir do dataframe `dados`, (i) defina o parâmetro estético para x; (ii) use a geometria para histogramas com bindwidth = 1, alpha = 0.7 e cor das bordas das barras "gray"; e (iii) use `facet_grid()`, de modo que um histograma fique em cima do outro.
8. Examine os histogramas plotados. Em média, foram atribuídas maiores notas de PAULISTANIDADE quando se ouviu qual variante? Justifique sua resposta.
   a. tepe
   b. retroflexo
9. Os dados de PAULISTANIDADE quando se ouviu o retroflexo parecem seguir uma distribuição normal? Aplique o teste de Shapiro sobre esses dados.

10. De acordo com o teste acima, a distribuição das notas de PAULISTANIDADE quando se ouviu o retroflexo seguem uma distribuição normal? Justifique sua resposta.
11. Aplique o teste de Shapiro aos dados de PAULISTANIDADE quando se ouviu o tepe.
12. De acordo com o teste acima, a distribuição das notas de PAULISTANIDADE quando se ouviu o tepe seguem uma distribuição normal? Justifique sua resposta.
13. Com `ggplot()`, faça boxplots simples da distribuição dos dados de PAULISTANIDADE por VARIANTE.OUVIDA. Use o conjunto geral de `dados` e notch = T.
14. Calcule a média de notas atribuídas na escala de PAULISTANIDADE por VARIANTE.OUVIDA. Nomeie a coluna da medida como media.
15. Faça um teste estatístico para testar a hipótese nula de que a média da nota atribuída na escala de PAULISTANIDADE quando se ouviu o retroflexo é 5. Escolha o teste apropriado, segundo o resultado do teste de Shapiro, bem como o conjunto de dados. Estabeleça conf.int = T.
16. De acordo com o teste acima, qual deve ser a decisão do pesquisador?
    a. Rejeitar a hipótese alternativa e acatar a hipótese nula.
    b. Rejeitar a hipótese nula e acatar a hipótese alternativa.
    c. Refazer o teste.
17. Faça um teste estatístico para testar a hipótese nula de que média da nota atribuída na escala de PAULISTANIDADE é a mesma para quando se ouviu o tepe ou o retroflexo. Escolha o teste apropriado, segundo o resultado do teste de Shapiro, bem como o conjunto de dados. Estabeleça conf.int = T.
18. De acordo com o teste acima, a diferença entre as médias de PAULISTANIDADE para quando se ouviu tepe ou retroflexo é zero? Justifique sua resposta.
19. Qual é a diferença estimada entre as médias?
    a. 9.664e-15

   b. 0
   c. 1,199951
   d. 1,599976
   e. 1,900033

20. Com `ggplot()`, faça boxplots da distribuição das notas atribuídas ao grau de SOTAQUE por VARIANTE.OUVIDA e FALANTE. Para tanto, (i) escolha o dataframe apropriado; (ii) defina os parâmetros gráficos x e y; (iii) use a geometria de boxplots com notch = T; e (iv) use `facet_grid()` de modo que os falantes apareçam em facetas lado a lado no gráfico.
21. A figura permite comparar as notas atribuídas ao grau de SOTAQUE quando se ouviu o tepe ou o retroflexo para cada um dos quatro falantes. Para qual dos quatro falantes parece realmente não haver uma diferença significativa entre a nota atribuída a SOTAQUE para retroflexo ou tepe?
    a. Antonio
    b. Daniela
    c. Luisa
    d. Paulo
22. Na sequência, faremos quatro testes para analisar se a diferença entre as notas atribuídas ao grau de SOTAQUE é significativa a depender da VARIANTE.OUVIDA, para cada um dos quatro falantes. Como o teste se repetirá quatro vezes, é importante antes tomar uma importante medida. Qual é ela?
    a. fazer a correção de Bonferroni, ajustando o valor-*p* para 0,05/4
    b. fazer o teste-t para verificar se os dados seguem a distribuição normal
    c. fazer o teste de qui-quadrado para determinar os graus de liberdade
23. Calcule qual deve ser o valor-*p* para repetir o teste quatro vezes.
24. De acordo o valor-*p* determinado acima, qual deve ser o nível de confiança?
25. Faça um teste de Wilcoxon para testar a diferença entre as notas atribuídas ao grau de SOTAQUE a depender da VARIANTE.OUVIDA, apenas para o falante Antonio. Para tanto, com auxílio do pipe e a partir do dataframe dados, (i)

crie um subconjunto apenas dos dados do falante Antonio; e (ii) aplique o teste de Wilcoxon com a fórmula SOTAQUE ~ VARIANTE.OUVIDA e data = . (o ponto final, que indica o conjunto de dados criado à esquerda do pipe). Estabeleça o nível de confiança igual ao valor que você calculou na questão acima.

26. Faça o mesmo teste acima para a falante Daniela.
27. Faça o mesmo teste acima para o falante Paulo.
28. Faça o mesmo teste acima para a falante Luisa.
29. De acordo com os quatro testes acima, para qual ou quais falantes não há diferença significativa para o grau de SOTAQUE atribuído a depender da VARIANTE.OUVIDA?

Se quiser praticar mais, use os dados `pretonicas` ou os dados desta aula para elaborar novos testes. Para as pretônicas, por exemplo, você pode fazer e interpretar os testes para a vogal /o/. Para os dados do teste de percepções sociolinguísticas, você pode testar os efeitos de tepe/retroflexo sobre outras variáveis (ESCOLARIZACAO, CENTRALIDADE.BAIRRO etc.).