# Lição 13: Regressão Linear Parte 2

**N.B.**: Rode as linhas de comando a seguir antes de iniciar esta lição. Defina como diretório de trabalho aquele que contém o arquivo Pretonicas.csv.

```
# Definir diretório de trabalho

#setwd()

# Importar planilha de dados

pretonicas <- read_csv("Pretonicas.csv",
                       col_types = cols(.default = col_factor(),
                                        VOGAL = col_factor(levels = c(
"i", "e", "a", "o", "u")),
                                        F1 = col_double(),
                                        F2 = col_double(),
                                        F1.NORM = col_double(),
                                        F2.NORM = col_double(),
                                        F1.SIL.SEG = col_double(),
                                        F2.SIL.SEG = col_double(),
                                        F1.SEG.NORM = col_double(),
                                        F2.SEG.NORM = col_double(),
                                        DIST.TONICA = col_double(),
                                        Begin.Time.s = col_double(),
                                        End.Time.s = col_double(),
                                        Duration.ms = col_double(),
                                        IDADE = col_integer(),
                                        IDADE.CHEGADA = col_integer(),
                                        ANOS.SP = col_integer()
                                        )
                       )

pretonicas$CONT.PREC <-  fct_collapse(pretonicas$CONT.PREC,
                          dental.alveolar = c("t", "d", "n", "l"),
                          labial = c("p", "b", "m", "f", "v"),
                          palatal.sibilante = c("S", "Z", "L", "s", "z
"),
                          velar = c("k", "g"),
                          vibrante = c("h", "R")
                                    )

pretonicas$CONT.PREC <- fct_relevel(pretonicas$CONT.PREC, "dental.alve
olar", "labial", "palatal.sibilante", "velar", "vibrante")

pretonicas$CONT.SEG <- fct_collapse(pretonicas$CONT.SEG,
                     dental.alveolar = c("t", "d", "n", "l"),
                     labial = c("p", "b", "m", "f", "v"),
                     palatal.sibilante = c("S", "Z", "L", "N", "s", "z
```

```
"),
                    velar = c("k", "g"),
                    vibrante = c("r", "h", "R")
                                )

pretonicas$CONT.SEG <- fct_relevel(pretonicas$CONT.SEG, "dental.alveol
ar", "labial", "palatal.sibilante", "velar", "vibrante")

### Criar subconjunto de dados da vogal /e/ pretonica

VOGAL_e <- filter(pretonicas, VOGAL == "e") %>%
  droplevels()

### Retirar valores atípicos

VOGAL_e2 <- filter(VOGAL_e, F1.NORM < 500)
```

Esta lição dá continuidade ao tópico Regressão Linear, iniciado na lição anterior. Ali, criamos modelos relativamente simples de regressão linear com variáveis previsoras binária (AMOSTRA), com mais de 2 fatores (CONT.SEG) e numérica (F1.SEG.NORM), bem como modelos com duas variáveis sem e com interação, com vistas a treinar a leitura dos resultados. Entretanto, análises de regressão linear por meio da função `lm()` permitem incluir mais variáveis previsoras, e um modelo completo deve abarcar tantas variáveis pertinentes quanto possível – são esses modelos completos que, preferencialmente, você reportará em suas publicações.

Por outro lado, ao criar modelos multivariados, é importante ter em mente o princípio da Navalha de Occam, também conhecido como Princípio da Simplicidade ou Princípio da Parcimônia. Ele estabelece que teorias mais simples são preferíveis a teorias mais complexas, e que não se deve agregar hipóteses desnecessárias a uma teoria. Transpondo tal princípio a nossos modelos estatísticos, poderíamos imaginar que um modelo que inclui 20 variáveis previsoras se aproxima mais da realidade; contudo, se 15 dessas variáveis contribuem pouco para fazer previsões a respeito da variável resposta, um modelo com 5 variáveis é preferível por ser mais simples.

É daí também que surge o interesse em modelos multivariados: o efeito de uma variável previsora pode se mostrar pertinente em uma análise univariada, mas se revelar não tão influente quando considerada frente a outras variáveis. A análise multivariada

pode indiciar que o efeito de determinada variável é apenas superficial ou pequeno em vista de outros efeitos. Cabe então a pergunta: como decidir quais e quantas variáveis previsoras incluir num modelo linear?

A decisão sobre quais variáveis têm efeito de fato para descrever, explicar e prever o comportamento da variável resposta cabe inequivocadamente ao pesquisador. Cada variável previsora é uma hipótese a respeito da variável resposta, e as hipóteses, como já vimos, devem ter por base a teoria e a literatura sobre determinado assunto – em outras palavras, devem ser bem motivadas. Mas o pesquisador sempre pode propor novas correlações ou descobrir que um efeito que se mostrou significativo em um conjunto de dados não é em outro conjunto.

Para chegar a um modelo satisfatório dos dados, há duas abordagens possíveis: começar por um modelo estatístico simples e a ele acrescentar novas variáveis previsoras, uma a uma; ou começar com um modelo complexo, com todas as variáveis previsoras, e procurar eliminar aquelas que têm pouca ou nenhuma influência na variável resposta. O R tem algumas funções que facilitam a construção de tais modelos, dentre as quais estão as funções `step()` e `drop1()`.

Nesta lição, vamos precisar de quatro pacotes: tidyverse, car, lme4 e lmerTest. Carregue-os com as linhas de comando a seguir.

```
library(tidyverse)
library(car)
library(lme4)
library(lmerTest)
```

Nesta lição, vamos usar o dataframe `VOGAL_e2`, com que trabalhamos na lição passada. Veja sua estrutura com `str()`.

```
str(VOGAL_e2)

## spec_tbl_df [677 × 27] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ PALAVRA      : Factor w/ 259 levels “diferente”,”melhor”,..: 1 1
2 3 1 4 5 6 7 8 ...
##  $ Transc.Fon   : Factor w/ 259 levels “d<i>-f<e>-'RE-te”,..: 1 1 2
3 1 4 5 6 7 8 ...
##  $ VOGAL        : Factor w/ 1 level “e”: 1 1 1 1 1 1 1 1 1 1 ...
##  $ F1           : num [1:677] 613 656 573 735 567 ...
##  $ F2           : num [1:677] 2014 1848 2413 1656 1375 ...
##  $ F1.NORM      : num [1:677] 447 464 431 496 429 ...
```

```
##  $ F2.NORM      : num [1:677] 1698 1611 1905 1512 1366 ...
##  $ CONT.PREC    : Factor w/ 5 levels “dental.alveolar”,..: 2 2 2 5
2 4 2 3 2 1 ...
##  $ CONT.SEG     : Factor w/ 5 levels “dental.alveolar”,..: 5 5 3 2
5 5 1 5 3 2 ...
##  $ VOGAL.SIL.SEG: Factor w/ 11 levels “a”,”aw”,”A”,”\u0097”,..: 5 5
4 7 5 5 1 5 1 5 ...
##  $ F1.SIL.SEG   : num [1:677] 569 524 686 652 661 ...
##  $ F2.SIL.SEG   : num [1:677] 1674 2428 1497 2159 1865 ...
##  $ F1.SEG.NORM  : num [1:677] 350 336 385 375 378 ...
##  $ F2.SEG.NORM  : num [1:677] 1360 1724 1274 1594 1452 ...
##  $ VOGAL.TONICA : Factor w/ 14 levels “e”,”o”,”ow”,”a”,..: 9 9 2 1
9 9 4 9 4 9 ...
##  $ DIST.TONICA  : num [1:677] 1 1 1 1 1 1 1 1 1 2 ...
##  $ ESTR.SIL.PRET: Factor w/ 5 levels “CV”,”CVs”,”CCV”,..: 1 1 1 1 1
1 1 1 1 1 ...
##  $ Begin.Time.s : num [1:677] 219 226 576 584 614 ...
##  $ End.Time.s   : num [1:677] 219 226 576 584 614 ...
##  $ Duration.ms  : num [1:677] 10.4 11.6 30.3 17.5 17.1 ...
##  $ AMOSTRA      : Factor w/ 2 levels “PBSP”,”SP2010”: 1 1 1 1 1 1 1
1 1 1 ...
##  $ PARTICIPANTE : Factor w/ 14 levels “MartaS”,”JosaneV”,..: 1 1 1
1 1 1 1 1 1 1 ...
##  $ SEXO         : Factor w/ 2 levels “feminino”,”masculino”: 1 1 1
1 1 1 1 1 1 1 ...
##  $ IDADE        : int [1:677] 32 32 32 32 32 32 32 32 32 32 ...
##  $ IDADE.CHEGADA: int [1:677] 18 18 18 18 18 18 18 18 18 18 ...
##  $ ANOS.SP      : int [1:677] 14 14 14 14 14 14 14 14 14 14 ...
##  $ CONTEXTO     : Factor w/ 632 levels “diferente o clima de eu com
ele”,..: 1 2 3 4 5 6 7 8 9 10 ...
##  - attr(*, “spec”)=
##   .. cols(
##   ..   .default = col_factor(),
##   ..   PALAVRA = col_factor(levels = NULL, ordered = FALSE, include
_na = FALSE),
##   ..   Transc.Fon = col_factor(levels = NULL, ordered = FALSE, incl
ude_na = FALSE),
##   ..   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_factor(levels = NULL, ordered = FALSE, inclu
de_na = FALSE),
##   ..   CONT.SEG = col_factor(levels = NULL, ordered = FALSE, includ
e_na = FALSE),
##   ..   VOGAL.SIL.SEG = col_factor(levels = NULL, ordered = FALSE, i
nclude_na = FALSE),
##   ..   F1.SIL.SEG = col_double(),
##   ..   F2.SIL.SEG = col_double(),
##   ..   F1.SEG.NORM = col_double(),
##   ..   F2.SEG.NORM = col_double(),
##   ..   VOGAL.TONICA = col_factor(levels = NULL, ordered = FALSE, in
```

```
clude_na = FALSE),
##   ..   DIST.TONICA = col_double(),
##   ..   ESTR.SIL.PRET = col_factor(levels = NULL, ordered = FALSE, i
nclude_na = FALSE),
##   ..   Begin.Time.s = col_double(),
##   ..   End.Time.s = col_double(),
##   ..   Duration.ms = col_double(),
##   ..   AMOSTRA = col_factor(levels = NULL, ordered = FALSE, include
_na = FALSE),
##   ..   PARTICIPANTE = col_factor(levels = NULL, ordered = FALSE, in
clude_na = FALSE),
##   ..   SEXO = col_factor(levels = NULL, ordered = FALSE, include_na
= FALSE),
##   ..   IDADE = col_integer(),
##   ..   IDADE.CHEGADA = col_integer(),
##   ..   ANOS.SP = col_integer(),
##   ..   CONTEXTO = col_factor(levels = NULL, ordered = FALSE, includ
e_na = FALSE)
##   .. )
##  - attr(*, "problems")=<externalptr>
```

Imagine que um pesquisador tenha levantado a hipótese de que altura da vogal /e/ pretônica depende das variáveis AMOSTRA, SEXO, F1.SEG.NORM, CONT.PREC e CONT.SEG. A partir de `VOGAL_e2`, crie primeiro um modelo linear chamado `mod`, com F1.NORM como variável resposta e as variáveis AMOSTRA + SEXO + F1.SEG.NORM + CONT.PREC + CONT.SEG como variáveis previsoras.

```
mod <- lm(F1.NORM ~
            AMOSTRA +
            SEXO +
            F1.SEG.NORM +
            CONT.PREC +
            CONT.SEG,
          data = VOGAL_e2)
```

Veja o resultado de `mod` com `summary()`.

```
summary(mod)
```

```
##
## Call:
## lm(formula = F1.NORM ~ AMOSTRA + SEXO + F1.SEG.NORM + CONT.PREC +
##     CONT.SEG, data = VOGAL_e2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -60.457 -16.874  -0.574  14.869  70.772
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)               397.14855   10.42949  38.079  < 2e-16 **
```

```
*
## AMOSTRASP2010               -8.46124    1.93283  -4.378 1.39e-05 **
*
## SEXOmasculino               -2.80518    1.95100  -1.438 0.150958
## F1.SEG.NORM                  0.11337    0.02459   4.611 4.80e-06 **
*
## CONT.PREClabial             -0.56978    2.88994  -0.197 0.843762
## CONT.PRECpalatal.sibilante  -8.27567    2.96849  -2.788 0.005458 **
## CONT.PRECvelar              -1.31555    7.46441  -0.176 0.860157
## CONT.PRECvibrante            3.17324    3.38029   0.939 0.348201
## CONT.SEGlabial             -11.78880    4.14795  -2.842 0.004619 **
## CONT.SEGpalatal.sibilante  -14.49915    3.96319  -3.658 0.000274 **
*
## CONT.SEGvelar               -6.72795    3.56755  -1.886 0.059748 .
## CONT.SEGvibrante            -3.72273    3.60498  -1.033 0.302137
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 24.61 on 665 degrees of freedom
## Multiple R-squared:  0.101,  Adjusted R-squared:  0.08617
## F-statistic: 6.795 on 11 and 665 DF,  p-value: 6.897e-11
```

O resultado do modelo inclui todas as variáveis previsoras (e respectivos fatores, em caso de variáveis nominais), suas estimativas e valores de significância em relação ao coeficiente linear (Intercept). Como visto na lição passada, sua leitura depende do conhecimento de quais são os níveis de referência.

Para a variável AMOSTRA, qual é o nível de referência? Veja o resultado de `str()`, se necessário.

- PBSP
- SP2010

Para a variável SEXO, qual é o nível de referência?

- feminino
- masculino

Para a variável F1.SEG.NORM, qual é o nível de referência?

- F1.SEG.NORM = 0 Hz
- F1.SEG.NORM = 400 Hz
- F1.SEG.NORM = 600 Hz

Para a variável CONT.PREC, qual é o nível de referência?

- dental.alveolar
- labial
- palatal.sibilante
- velar
- vibrante

Para a variável CONT.SEG, qual é o nível de referência?

- dental.alveolar
- labial
- palatal.sibilante
- velar
- vibrante

Os respectivos valores de referência são o zero ou o primeiro nível de acordo com a ordem em que aparecem na planilha ou de acordo com o especificado pelo usuário no momento da importação dos dados. Em `mod`, o coeficiente linear 397,14855 se refere, portanto, à estimativa do valor de F1.NORM para falantes paraibanos do sexo feminino, quando F1.SEG.NORM é zero e quando o contexto precedente e o contexto seguinte são consoantes dental-alveolares. As estimativas para todos os outros cenários possíveis podem ser deduzidos a partir da soma dos valores dos coeficientes angulares.

Vemos que a estimativa de F1.NORM para AMOSTRASP2010 difere significativamente em relação a PBSP; a cada unidade de F1.NORM; quando CONT.PREC é palatal-sibilante (em relação à dental-alveolar); e quando CONT.SEG é labial ou palatal-sibilante (em relação à dental-alveolar). Não há diferença significativa para as estimativas de homens e mulheres, e em relação aos demais contextos precedentes ou seguintes em relação à dental-alveolar. Sobre a variável SEXO, um pesquisador pode se perguntar se é pertinente manter essa variável no modelo, uma vez que parece não estar contribuindo muito para a estimativa da variável resposta. Sobre as variáveis CONT.PREC e CONT.SEG, o pesquisador pode se perguntar se as variáveis são relevantes de modo global, já que

apenas um ou dois fatores de cada variável mostram diferenças significativas em relação ao nível de referência.

A função `step()` compara diferentes modelos com e sem a inclusão de diferentes variáveis e reporta, ao final, quais variáveis devem ser mantidas. Para isso, baseia-se no AIC (Akaike Information Criterion), que penaliza o modelo se tem muitas variáveis – de modo semelhante ao $R^2$ ajustado. A função pode ser aplicada em três direções: (i) forward; (ii) backward; e (iii) both. Comecemos com a opção "forward".

Para isso, primeiro precisamos criar um modelo que não inclui qualquer variável previsora. Crie então um modelo chamado `m0`, com a seguinte linha de comando: `m0 <- lm(F1.NORM ~ 1, data = VOGAL_e2)`.

```
m0 <- lm(F1.NORM ~ 1, data = VOGAL_e2)
```

Vamos agora aplicar a função `step()`, com os seguintes argumentos: (i) `m0`; (ii) direction = "forward"; (iii) scope = ~ AMOSTRA + SEXO + F1.SEG.NORM + CONT.PREC + CONT.SEG. Guarde o resultado num objeto chamado `m.fw`.

```
m.fw <- step(m0, direction = "forward", scope = ~ AMOSTRA + SEXO + F1.
SEG.NORM + CONT.PREC + CONT.SEG)

## Start:  AIC=4399.27
## F1.NORM ~ 1
##
##                Df Sum of Sq    RSS    AIC
## + F1.SEG.NORM  1   15538.4 432617 4377.4
## + AMOSTRA      1   11939.5 436216 4383.0
## + CONT.SEG     4   10796.0 437359 4390.8
## + CONT.PREC    4    6421.6 441733 4397.5
## <none>                     448155 4399.3
## + SEXO         1     336.6 447818 4400.8
##
## Step:  AIC=4377.38
## F1.NORM ~ F1.SEG.NORM
##
##              Df Sum of Sq    RSS    AIC
## + AMOSTRA     1   10793.0 421824 4362.3
## + CONT.SEG    4    9127.0 423490 4370.9
## + CONT.PREC   4    6876.3 425740 4374.5
## <none>                    432617 4377.4
## + SEXO        1     422.5 432194 4378.7
##
## Step:  AIC=4362.27
## F1.NORM ~ F1.SEG.NORM + AMOSTRA
##
##              Df Sum of Sq    RSS    AIC
```

```
## + CONT.SEG   4    9685.2 412138 4354.5
## + CONT.PREC  4    7346.8 414477 4358.4
## <none>                   421824 4362.3
## + SEXO       1     395.9 421428 4363.6
##
## Step:  AIC=4354.55
## F1.NORM ~ F1.SEG.NORM + AMOSTRA + CONT.SEG
##
##             Df Sum of Sq    RSS    AIC
## + CONT.PREC  4    8010.5 404128 4349.3
## <none>                   412138 4354.5
## + SEXO       1     903.8 411235 4355.1
##
## Step:  AIC=4349.26
## F1.NORM ~ F1.SEG.NORM + AMOSTRA + CONT.SEG + CONT.PREC
##
##        Df Sum of Sq    RSS    AIC
## + SEXO  1    1252.4 402875 4349.2
## <none>              404128 4349.3
##
## Step:  AIC=4349.16
## F1.NORM ~ F1.SEG.NORM + AMOSTRA + CONT.SEG + CONT.PREC + SEXO
```

Acima, especificamos que queremos partir do modelo `m0`, sem variáveis previsoras, e avaliar se a adição de novas variáveis melhora seu poder explanatório. O R então calculou o AIC para diferentes modelos e incluiu as variáveis F1.NORM, AMOSTRA, CONT.SEG, CONT.PREC e SEXO – nessa ordem. Essa ordem de seleção indica a importância relativa de cada variável para explicar a variação em F1.NORM. Digite agora `m.fw` para ver o cálculo de coeficientes de cada previsor.

```
m.fw

##
## Call:
## lm(formula = F1.NORM ~ F1.SEG.NORM + AMOSTRA + CONT.SEG + CONT.PREC
+
##     SEXO, data = VOGAL_e2)
##
## Coefficients:
##                (Intercept)                F1.SEG.NORM
##                   397.1486                     0.1134
##              AMOSTRASP2010             CONT.SEGlabial
##                   -8.4612                   -11.7888
##  CONT.SEGpalatal.sibilante              CONT.SEGvelar
##                  -14.4992                    -6.7280
##           CONT.SEGvibrante            CONT.PREClabial
##                   -3.7227                    -0.5698
## CONT.PRECpalatal.sibilante             CONT.PRECvelar
##                   -8.2757                    -1.3156
```

```
##          CONT.PRECvibrante                SEXOmasculino
##                     3.1732                      -2.8052
```

Vamos agora fazer um modelo "de trás para frente". A função `step()` toma novamente como argumentos o modelo de qual se quer partir – aqui usaremos `mod`, o modelo completo feito acima – e a direção – direction = "backward". Neste caso, não é necessário especificar o argumento scope. Digite então `m.bw <- step(mod, direction = "backward")`.

```
m.bw <- step(mod, direction = "backward")

## Start:  AIC=4349.16
## F1.NORM ~ AMOSTRA + SEXO + F1.SEG.NORM + CONT.PREC + CONT.SEG
##
##                Df Sum of Sq    RSS    AIC
## <none>                      402875 4349.2
## - SEXO          1    1252.4 404128 4349.3
## - CONT.PREC     4    8359.2 411235 4355.1
## - CONT.SEG      4   11162.6 414038 4359.7
## - AMOSTRA       1   11609.9 414485 4366.4
## - F1.SEG.NORM   1   12881.8 415757 4368.5
```

No modelo "de trás para frente", o R começa com o modelo completo e tenta excluir variáveis uma a uma. Nenhuma variável foi excluída, e o R as manteve na ordem em que foram especificadas dentro do modelo inicial – AMOSTRA, SEXO, F1.SEG.NORM, CONT.PREC e CONT.SEG. O importante aqui é checar se as mesmas variáveis que foram incluídas no modelo "para frente" também são incluídas no modelo "de trás para frente". Caso isso não ocorra, é um sinal de que há interação entre variáveis do modelo. Cabe ao pesquisador encontrá-la(s) e incluir a interação num novo modelo.

Veja também o resultado guardado em `m.bw`.

```
m.bw

##
## Call:
## lm(formula = F1.NORM ~ AMOSTRA + SEXO + F1.SEG.NORM + CONT.PREC +
##     CONT.SEG, data = VOGAL_e2)
##
## Coefficients:
##                (Intercept)                AMOSTRASP2010
##                   397.1486                      -8.4612
##              SEXOmasculino                  F1.SEG.NORM
##                   -2.8052                       0.1134
##           CONT.PREClabial  CONT.PRECpalatal.sibilante
##                   -0.5698                      -8.2757
```

```
##             CONT.PRECvelar           CONT.PRECvibrante
##                    -1.3156                      3.1732
##             CONT.SEGlabial   CONT.SEGpalatal.sibilante
##                   -11.7888                    -14.4992
##              CONT.SEGvelar            CONT.SEGvibrante
##                    -6.7280                    -3.7227
```

Note que os coeficientes angulares são os mesmos calculados para o modelo forward. Por fim, apliquemos a direção "both". Neste caso, o programa começa executando o mesmo que a direção "forward" mas, toda vez que inclui uma nova variável, ele tenta excluir alguma variável que possa não mais estar contribuindo para o modelo. A partir da linha de comando em que você criou `m.fw`, mude o nome do objeto para `m.both` e apague o argumento `direction`; não é necessário especificá-lo pois este é o valor *default* da função.

```
m.both <- step(m0, scope = ~ AMOSTRA + SEXO + F1.SEG.NORM + CONT.PREC
+ CONT.SEG)

## Start:  AIC=4399.27
## F1.NORM ~ 1
##
##               Df Sum of Sq    RSS    AIC
## + F1.SEG.NORM  1   15538.4 432617 4377.4
## + AMOSTRA      1   11939.5 436216 4383.0
## + CONT.SEG     4   10796.0 437359 4390.8
## + CONT.PREC    4    6421.6 441733 4397.5
## <none>                     448155 4399.3
## + SEXO         1     336.6 447818 4400.8
##
## Step:  AIC=4377.38
## F1.NORM ~ F1.SEG.NORM
##
##               Df Sum of Sq    RSS    AIC
## + AMOSTRA      1   10793.0 421824 4362.3
## + CONT.SEG     4    9127.0 423490 4370.9
## + CONT.PREC    4    6876.3 425740 4374.5
## <none>                     432617 4377.4
## + SEXO         1     422.5 432194 4378.7
## - F1.SEG.NORM  1   15538.4 448155 4399.3
##
## Step:  AIC=4362.27
## F1.NORM ~ F1.SEG.NORM + AMOSTRA
##
##               Df Sum of Sq    RSS    AIC
## + CONT.SEG     4    9685.2 412138 4354.5
## + CONT.PREC    4    7346.8 414477 4358.4
## <none>                     421824 4362.3
## + SEXO         1     395.9 421428 4363.6
## - AMOSTRA      1   10793.0 432617 4377.4
```

```
## - F1.SEG.NORM  1   14391.9 436216 4383.0
##
## Step:  AIC=4354.55
## F1.NORM ~ F1.SEG.NORM + AMOSTRA + CONT.SEG
##
##               Df Sum of Sq    RSS    AIC
## + CONT.PREC    4    8010.5 404128 4349.3
## <none>                     412138 4354.5
## + SEXO         1     903.8 411235 4355.1
## - CONT.SEG     4    9685.2 421824 4362.3
## - AMOSTRA      1   11351.2 423490 4370.9
## - F1.SEG.NORM  1   12965.0 425103 4373.5
##
## Step:  AIC=4349.26
## F1.NORM ~ F1.SEG.NORM + AMOSTRA + CONT.SEG + CONT.PREC
##
##               Df Sum of Sq    RSS    AIC
## + SEXO         1    1252.4 402875 4349.2
## <none>                     404128 4349.3
## - CONT.PREC    4    8010.5 412138 4354.5
## - CONT.SEG     4   10348.9 414477 4358.4
## - AMOSTRA      1   11627.3 415755 4366.5
## - F1.SEG.NORM  1   12739.3 416867 4368.3
##
## Step:  AIC=4349.16
## F1.NORM ~ F1.SEG.NORM + AMOSTRA + CONT.SEG + CONT.PREC + SEXO
##
##               Df Sum of Sq    RSS    AIC
## <none>                     402875 4349.2
## - SEXO         1    1252.4 404128 4349.3
## - CONT.PREC    4    8359.2 411235 4355.1
## - CONT.SEG     4   11162.6 414038 4359.7
## - AMOSTRA      1   11609.9 414485 4366.4
## - F1.SEG.NORM  1   12881.8 415757 4368.5
```

E veja os valores de coeficiente angular no resultado guardado em `m.both`.

```
m.both

##
## Call:
## lm(formula = F1.NORM ~ F1.SEG.NORM + AMOSTRA + CONT.SEG + CONT.PREC
+
##     SEXO, data = VOGAL_e2)
##
## Coefficients:
##                (Intercept)                 F1.SEG.NORM
##                   397.1486                      0.1134
##              AMOSTRASP2010              CONT.SEGlabial
##                   -8.4612                   -11.7888
##  CONT.SEGpalatal.sibilante               CONT.SEGvelar
##                   -14.4992                    -6.7280
##           CONT.SEGvibrante             CONT.PREClabial
##                   -3.7227                    -0.5698
```

```
## CONT.PRECpalatal.sibilante              CONT.PRECvelar
##                    -8.2757                     -1.3156
##          CONT.PRECvibrante               SEXOmasculino
##                     3.1732                     -2.8052
```

Novamente, o que se espera é que as variáveis selecionadas sejam as mesmas e que os coeficientes angulares também tenham os mesmos valores. Caso isso não ocorra, há uma forte evidência de que as variáveis não são independentes entre si e deve-se verificar se há interações entre as variáveis do modelo.

Outra função que permite avaliar se vale a pena manter determinada variável no modelo estatístico é `drop1()`. A função requer um modelo com inclusão de todas as variáveis previsoras que se quer testar – no nosso caso, `mod` – e o tipo de teste a se aplicar – aqui vamos usar "F", que toma por base a estatística-F para comparar modelos (aquela que aparece ao pé do resultado de `summary()`; outra opção seria o teste "Chisq"). Digite então `drop1(mod, test = "F")`.

```
drop1(mod, test = "F")
```

```
## Single term deletions
##
## Model:
## F1.NORM ~ AMOSTRA + SEXO + F1.SEG.NORM + CONT.PREC + CONT.SEG
##             Df Sum of Sq    RSS    AIC F value    Pr(>F)
## <none>                   402875 4349.2
## AMOSTRA      1   11609.9 414485 4366.4 19.1637 1.393e-05 ***
## SEXO         1    1252.4 404128 4349.3  2.0673  0.150958
## F1.SEG.NORM  1   12881.8 415757 4368.5 21.2632 4.802e-06 ***
## CONT.PREC    4    8359.2 411235 4355.1  3.4495  0.008400 **
## CONT.SEG     4   11162.6 414038 4359.7  4.6063  0.001127 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

O resultado de `drop1()` apresenta um valor de significância para cada variável previsora incluída no modelo (assim como outras medidas estatísticas). Aqui vemos que SEXO não se correlaciona significativamente com F1.NORM – diferentemente do resultado da função `step()`. Cabe ao pesquisador decidir se ele seguirá o resultado da função `step()` ou `drop1()`. Os resultados podem diferir porque cada teste se baseia em um critério diferente (AIC para `step()`; estatística-F ou qui-quadrado para `drop1()`). Por mais frustrante que isso possa ser, não há aqui fórmula mágica para lhe dizer qual decisão

tomar. Mas esse fato é importante para nos lembrar de que a análise estatística não dá todas as respostas, pois está sempre sujeita à interpretação do pesquisador.

Com `drop1()`, o pesquisador pode ainda atualizar o modelo e continuar aplicando a função para tentar excluir mais uma variável, pois a função exclui as variáveis previsoras uma de cada vez. Aqui, não a aplicaremos mais pois todas as variáveis remanescentes são significativas.

Temos então boas evidências de que as variáveis F1.SEG.NORM, AMOSTRA, CONT.SEG e CONT.PREC se correlacionam com a altura da vogal /e/ pretônica, mas evidências não tão fortes quanto ao papel da variável SEXO. Vamos criar um modelo linear chamado `modelo` com a inclusão apenas das variáveis selecionadas tanto em `step()` quanto em `drop1()` – ou seja, excluindo a variável SEXO –, nos dados de `VOGAL_e2`.

```
modelo <- lm(F1.NORM ~ F1.SEG.NORM + AMOSTRA + CONT.SEG + CONT.PREC, d
ata = VOGAL_e2)
```

Após chegar a um modelo de seus dados, ainda é necessário fazer alguns testes para checar se o modelo não viola os pressupostos básicos de modelos lineares. A checagem desses pressupostos é importante para se certificar de que seus resultados são confiáveis.

O primeiro deles é fácil: a variável resposta deve ser numérica e contínua. Numa variável contínua, a relação entre os valores das observações é quantitativa; por exemplo, é possível dizer que uma vogal com 300 Hz de F1 tem a metade do valor de uma vogal com 600 Hz. Para variáveis nominais, aplicamos outros tipos de teste (regressão logística – que veremos nas Lições 14 e 15 –, multinomial ou de Poisson – que não serão vistas neste curso).

Uma segunda checagem a fazer é verificar se a relação entre a variável resposta e uma variável previsora numérica é de fato linear. Isso pode ser feito por meio da função `plot()` – como fizemos na Lição 11 – ou ainda por meio da função `crPlot()` do pacote car.

Em nosso modelo, apenas F1.SEG.NORM é uma variável numérica. Digite então `crPlot(modelo, var = "F1.SEG.NORM")` para checar se a relação entre essa variável e a variável resposta é linear.

```
crPlot(modelo, var = "F1.SEG.NORM")
```

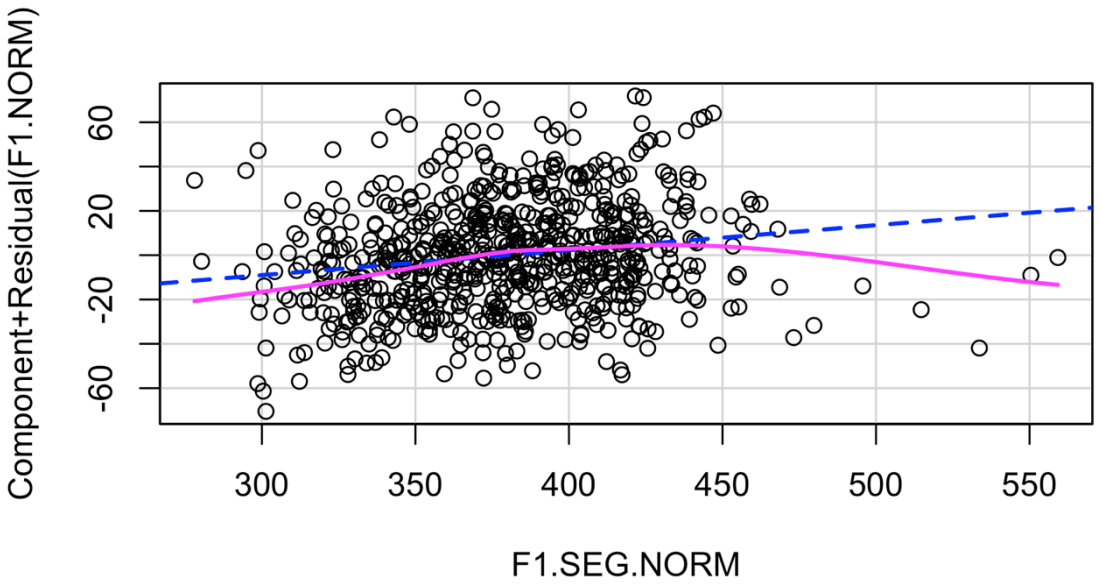


*Figura 13.1: Plot dos valores previstos e observados de F1.SEG.NORM no modelo. Fonte: própria.*

Na figura plotada, a linha pontilhada corresponde aos valores previstos pelo modelo, e a linha contínua corresponde a uma linha de regressão suavizada que segue mais fielmente a distribuição observada. Se a linha contínua não segue a linha pontilhada, isso é um sinal de que o pressuposto de relação de linearidade entre as variáveis não é cumprido. Na Figura 13.1, vemos que a linha contínua mais se distancia da linha pontilhada nos maiores valores de F1.SEG.NORM. No início da análise, na lição anterior, retiramos dados acima de 500 Hz apenas para a variável resposta (F1.NORM), mas não para a vogal da sílaba seguinte. Esses correspondem apenas a quatro dados, de modo que retirá-los não causará grande impacto na amostra, ao mesmo tempo que melhorará o modelo. Crie então um novo subconjunto de dados, chamado `VOGAL_e3`, com os dados de F1.SEG.NORM abaixo de 500 Hz.

```
VOGAL_e3 <- filter(VOGAL_e2, F1.SEG.NORM < 500)
```

Crie agora um novo modelo linear, chamado `modelo2`, com a mesma fórmula F1.NORM ~ AMOSTRA + F1.SEG.NORM + CONT.PREC + CONT.SEG, no conjunto `VOGAL_e3`.

```
modelo2 <- lm(F1.NORM ~ AMOSTRA + F1.SEG.NORM + CONT.PREC + CONT.SEG,
data = VOGAL_e3)
```

E aplique a função `crPlot()` a `modelo2`, com var = "F1.SEG.NORM", para checar se a relação entre as variáveis se aproxima mais da linearidade.

```
crPlot(modelo2, var = "F1.SEG.NORM")
```

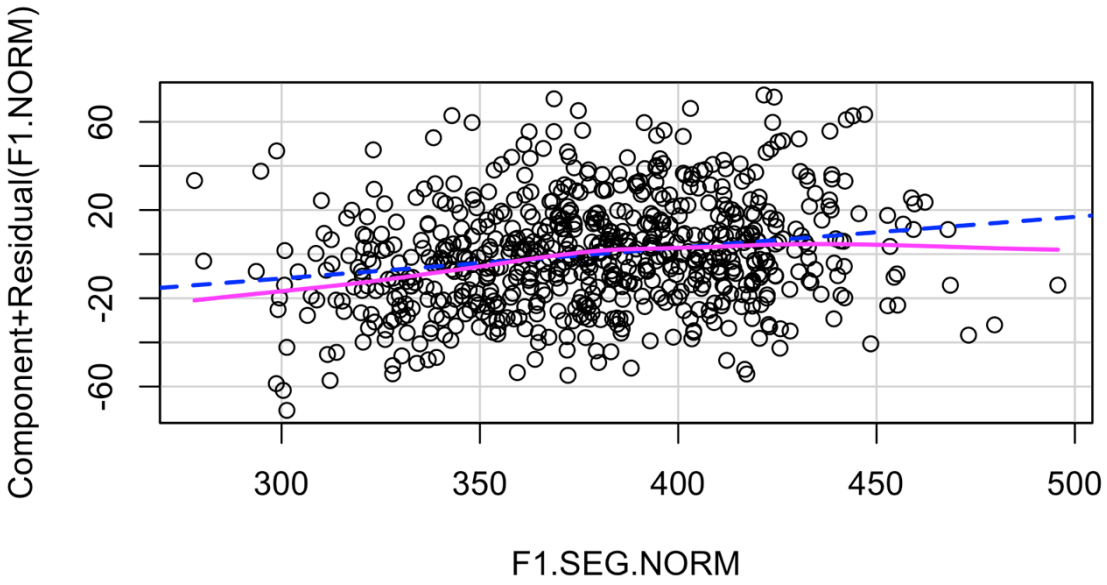


*Figura 13.2: Plot dos valores previstos e observados de F1.SEG.NORM no modelo2. Fonte: própria.*

Em relação à Figura 13.1, a linha contínua na Figura 13.2 se aproxima muito mais da linha pontilhada.

Outro pressuposto fundamental num modelo linear é que as variáveis previsoras não sejam dependentes entre si. Tal dependência é chamada de multicolinearidade, que ocorre quando algumas variáveis previsoras do modelo se referem a um mesmo efeito. No conjunto original de pretônicas, por exemplo, F1.SEG.NORM se refere à altura da vogal da sílaba seguinte; há ainda outra variável na planilha, chamada VOGAL.SIL.SEG, que

codifica a vogal em vez de seu valor de F1. Trata-se de uma mesma variável, vista de duas maneiras (a primeira, contínua; a segunda, nominal), de modo que não devem ser incluídas em um mesmo modelo. Em geral, o próprio pesquisador é capaz de prever que duas variáveis não são independentes entre si e já não as incluir num mesmo teste estatístico.

Mas há uma função do pacote car, `vif()`, que permite checar se variáveis incluídas num modelo são colineares. Aplique-a a `modelo2`.

```
vif(modelo2)
```

```
##                 GVIF Df GVIF^(1/(2*Df))
## AMOSTRA     1.046156  1        1.022818
## F1.SEG.NORM 1.038512  1        1.019074
## CONT.PREC   1.973176  4        1.088669
## CONT.SEG    1.978711  4        1.089050
```

Quando duas variáveis são colineares, os valores de GVIF (da primeira coluna) e GVIF-ajustado (da terceira coluna) são altos, acima de 5. Nosso modelo está ok quanto a esse critério, já que todos os valores são abaixo de 2. Caso verifique valores acima de 5, você deve considerar não incluir as variáveis colineares no mesmo modelo.

Outro pressuposto já foi mencionado na lição anterior: a distribuição dos resíduos é normal, com valores simétricos e mediana zero. Digite `summary(modelo2)` para visualizar o resultado de nosso último modelo.

```
summary(modelo2)
```

```
##
## Call:
## lm(formula = F1.NORM ~ AMOSTRA + F1.SEG.NORM + CONT.PREC + CONT.SEG
,
##     data = VOGAL_e3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -59.972 -16.741  -1.065  14.618  71.855
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                384.43394   10.83542  35.479  < 2e-16 **
*
## AMOSTRASP2010               -8.26394    1.93126  -4.279 2.15e-05 **
*
## F1.SEG.NORM                  0.13958    0.02592   5.385 1.01e-07 **
*
```

```
## CONT.PREClabial             -0.07849    2.87425  -0.027 0.978223
## CONT.PRECpalatal.sibilante  -8.02946    2.95385  -2.718 0.006734 **
## CONT.PRECvelar              -1.14571    7.42957  -0.154 0.877492
## CONT.PRECvibrante            3.65342    3.38749   1.079 0.281201
## CONT.SEGlabial             -10.12030    4.10273  -2.467 0.013888 *
## CONT.SEGpalatal.sibilante  -13.55855    3.92429  -3.455 0.000585 **
*
## CONT.SEGvelar               -6.02371    3.53098  -1.706 0.088484 .
## CONT.SEGvibrante            -2.44226    3.59664  -0.679 0.497350
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 24.49 on 662 degrees of freedom
## Multiple R-squared:  0.1099, Adjusted R-squared:  0.09646
## F-statistic: 8.174 on 10 and 662 DF,  p-value: 1.524e-12
```

Vemos nos resíduos que a distribuição não parece ser normal, principalmente pelos valores Min (-59,972) e Max (71,855). Já vimos, em lições anteriores, uma função para verificar numericamente se uma distribuição é normal. Qual é ela?

- `bonferroni.test()`
- `chisq.test()`
- `shapiro.test()`
- `t.test()`

Aplique então o teste de Shapiro a `modelo2$residuals`.

```
shapiro.test(modelo2$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  modelo2$residuals
## W = 0.99349, p-value = 0.005183
```

O que informa o teste acima?

- a distribuição dos resíduos de `modelo2` é normal
- a distribuição dos resíduos de `modelo2` não é normal

A assunção de normalidade dos resíduos torna-se menos importante quando se trabalha com uma amostra grande. Neste caso, em que temos mais de 600 dados, podemos ficar razoavelmente seguros de que a não normalidade dos resíduos não afeta as estimativas de maneira danosa. Em amostras menores, a não normalidade dos resíduos

pode ser mais problemática, visto que cada observação tem maior peso quanto menor for a amostra (Lição 6).

Por fim, é importante avaliar se cada observação é independente uma das outras. Idealmente, cada dado coletado da população deveria ter a mesma chance de entrar na amostra. Em estudos linguísticos, isso raramente é o caso. Nos dados de vogais pretônicas, em que estamos trabalhando, as mais de 674 ocorrências de /e/ pretônico advêm de 14 falantes (7 paraibanos e 7 paulistanos), o que significa que vieram de um conjunto pequeno da população. De cada falante, foram extraídos cerca de 50 dados da vogal /e/, o que significa que os dados em cada subconjunto não são independentes uns dos outros.

Grande parte da variabilidade nos dados linguísticos se deve ao próprio falante. Na estatística, este tipo de variável é chamada de *efeito aleatório*, pois normalmente muda a cada amostra. Efeitos aleatórios se contrapõem a efeitos *fixos*, que podem ser facilmente reproduzidos em outros estudos. Por exemplo, a variável SEXO é um efeito fixo, pois seus níveis – feminino e masculino – podem ser facilmente reproduzidos em nova amostra de falantes paraibanos e paulistanos. Se escolhêssemos novos falantes aleatoriamente, provavelmente teríamos homens e mulheres na nova amostra. Por outro lado, essa mesma nova amostragem dificilmente conteria os exatos 14 falantes da primeira amostra. Considerando-se que muito da variabilidade nos dados vem dos próprios indivíduos, é importante levar em conta sua contribuição para o resultado final dos modelos estatísticos.

Outro efeito aleatório comum em estudos linguísticos é o *item lexical*. Certas palavras podem ter comportamento idiossincrático, independentemente de condicionamentos mais gerais como classe morfológica, contexto fonológico precedente a um segmento alvo, contexto fonológico seguinte, função sintática etc. Seguindo o mesmo raciocínio acima, uma nova amostra aleatória de dados linguísticos (coletados da mesma maneira) muito provavelmente conterá substantivos, verbos, advérbios..., mas dificilmente conterá exatamente os mesmos itens lexicais que compõem a amostra original.

Efeitos aleatórios, quando existirem, *sempre* devem ser incluídos nos modelos estatísticos. Modelos que incluem tanto efeitos fixos quanto efeitos aleatórios são chamados de *modelos de efeitos mistos*. O motivo de termos deixado os efeitos aleatórios para o final é bastante prático: você verá daqui a pouco que esses modelos demoram um bocado para rodar no R. Minha recomendação, portanto, é que você chegue a um modelo satisfatório com efeitos fixos, cheque se esse modelo não viola os pressupostos de modelos lineares (fazendo novos ajustes, se necessários) e, apenas como última etapa, inclua os efeitos aleatórios. Este é um excelente exemplar da expressão "por último, mas não menos importante": qualquer estudo que tenha efeitos aleatórios deve incluí-los na modelagem estatística.

Os modelos de efeitos mistos são implementados no R por meio do pacote lme4, que carregamos no início desta lição. O pacote lmerTest, também já carregado, fornece valores-*p* para as estimativas.

A função `lmer()` toma os mesmos argumentos de `lm()`, com a diferença de que requer a inclusão de efeitos aleatórios. Estas são indicadas dentro da fórmula com a notação (1|varaleatoria). Vamos então criar um modelo de efeitos mistos a partir do último modelo criado (modelo2). A partir daquela linha de comando, modifique (i) o nome do objeto para `mod1.lmer`; (ii) a função para lmer; e (iii) inclua duas novas variáveis na fórmula: + (1|PARTICIPANTE) + (1|PALAVRA).

```
mod1.lmer <- lmer(F1.NORM ~
                    AMOSTRA +
                    F1.SEG.NORM +
                    CONT.PREC +
                    CONT.SEG +
                    (1|PARTICIPANTE) +
                    (1|PALAVRA),
                  data = VOGAL_e3)
```

Aplique a função `summary()` a `mod1.lmer`.

```
summary(mod1.lmer)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method
[
## lmerModLmerTest]
## Formula:
## F1.NORM ~ AMOSTRA + F1.SEG.NORM + CONT.PREC + CONT.SEG + (1 |
```

```
##     PARTICIPANTE) + (1 | PALAVRA)
##    Data: VOGAL_e3
##
## REML criterion at convergence: 6166
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.37177 -0.67890 -0.03253  0.58279  2.88123
##
## Random effects:
##  Groups       Name        Variance Std.Dev.
##  PALAVRA      (Intercept)  29.805   5.459
##  PARTICIPANTE (Intercept)   9.184   3.030
##  Residual                 565.113  23.772
## Number of obs: 673, groups:  PALAVRA, 255; PARTICIPANTE, 14
##
## Fixed effects:
##                             Estimate Std. Error        df t value
## (Intercept)                379.20012   11.37177 414.46165  33.346
## AMOSTRASP2010               -8.06663    2.52093  12.06047  -3.200
## F1.SEG.NORM                  0.14891    0.02704 463.76920   5.508
## CONT.PREClabial              0.54952    3.22933 134.99159   0.170
## CONT.PRECpalatal.sibilante  -5.90882    3.32399 128.45128  -1.778
## CONT.PRECvelar               0.69095    7.79936 262.83540   0.089
## CONT.PRECvibrante            4.90485    3.62336 232.44699   1.354
## CONT.SEGlabial              -9.49738    4.39146 230.83428  -2.163
## CONT.SEGpalatal.sibilante  -12.53629    4.11419 302.56280  -3.047
## CONT.SEGvelar               -5.57322    3.86767 170.43261  -1.441
## CONT.SEGvibrante            -2.27535    3.83996 242.30288  -0.593
##                            Pr(>|t|)
## (Intercept)                 < 2e-16 ***
## AMOSTRASP2010               0.00759 **
## F1.SEG.NORM                6.04e-08 ***
## CONT.PREClabial             0.86514
## CONT.PRECpalatal.sibilante  0.07783 .
## CONT.PRECvelar              0.92947
## CONT.PRECvibrante           0.17716
## CONT.SEGlabial              0.03159 *
## CONT.SEGpalatal.sibilante   0.00251 **
## CONT.SEGvelar               0.15143
## CONT.SEGvibrante            0.55404
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) AMOSTR F1.SEG CONT.PRECl CONT.PREC. CONT.PRECvl
## AMOSTRASP20 -0.170
## F1.SEG.NORM -0.924  0.040
## CONT.PREClb -0.159  0.066 -0.033
## CONT.PRECp. -0.203  0.086  0.013  0.523
## CONT.PRECvl -0.094  0.064  0.034  0.220      0.183
## CONT.PRECvb -0.229  0.111  0.034  0.452      0.440      0.193
## CONT.SEGlbl -0.329  0.048  0.059  0.331      0.280      0.035
## CONT.SEGpl. -0.298 -0.011  0.095  0.055      0.132     -0.061
```

```
## CONT.SEGvlr -0.255  0.027 -0.004  0.150      0.106      0.045
## CONT.SEGvbr -0.316 -0.001  0.084 -0.091      0.101     -0.032
##             CONT.PRECvb CONT.SEGl CONT.SEG. CONT.SEGvl
## AMOSTRASP20
## F1.SEG.NORM
## CONT.PRECl
## CONT.PRECp.
## CONT.PRECvl
## CONT.PRECvb
## CONT.SEGlbl  0.286
## CONT.SEGpl. -0.018       0.545
## CONT.SEGvlr  0.199       0.621     0.582
## CONT.SEGvbr  0.229       0.581     0.590     0.660
```

O principal resultado a checar num modelo de efeitos mistos é se os mesmos efeitos fixos continuam a ser correlacionados após a inclusão dos efeitos aleatórios. Ao ver os coeficientes de `mod1.lmer`, percebemos que a variável CONT.PREC deixa de ser significativamente correlacionada com F1.NORM. Isso é sinal de que o efeito anteriormente observado se deve a alguns falantes ou a alguns itens lexicais específicos (mais provavelmente este último caso, já que se trata de uma variável linguística) e, tendo-os em conta como efeito aleatório, pode-se chegar à conclusão de que CONT.PREC não tem um efeito verdadeiro sobre a variável resposta.

Podemos também aplicar a função `step()`, na direção "de trás para frente", para verificar se a variável CONT.PREC é excluída desse novo modelo (as direções "forward" e "both" não funcionarão aqui, já que a função `lmer()` não permite criar um modelo sem efeitos aleatórios). Digite então `m.bw.lmer <- step(mod1.lmer, direction = "backward")`.

```
m.bw.lmer <- step(mod1.lmer, direction = "backward")
```

Veja o resultado de `m.bw.lmer`.

```
m.bw.lmer

## Backward reduced random-effect table:
##
##                    Eliminated npar  logLik    AIC    LRT Df
## <none>                             14 -3083.0 6194.0
## (1 | PARTICIPANTE)          1   13 -3084.0 6194.1 2.0447  1
## (1 | PALAVRA)               0   12 -3085.8 6195.6 3.4823  1
##                    Pr(>Chisq)
## <none>
## (1 | PARTICIPANTE)    0.15273
## (1 | PALAVRA)         0.06203 .
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Backward reduced fixed-effect table:
## Degrees of freedom method: Satterthwaite
##
##             Eliminated  Sum Sq Mean Sq NumDF  DenDF F value    Pr(>
F)
## CONT.PREC            1  5456.1  1364.0     4 180.78  2.3711   0.054
14
## AMOSTRA              0  9896.0  9896.0     1 649.74 17.2840 3.650e-
05
## F1.SEG.NORM          0 16752.3 16752.3     1 504.77 29.2589 9.794e-
08
## CONT.SEG             0  6040.5  1510.1     4 175.21  2.6375   0.035
63
##
## CONT.PREC   .
## AMOSTRA     ***
## F1.SEG.NORM ***
## CONT.SEG    *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Model found:
## F1.NORM ~ AMOSTRA + F1.SEG.NORM + CONT.SEG + (1 | PALAVRA)
```

As primeiras linhas do resultado mostram as variáveis eliminadas e mantidas na coluna Eliminated. Dos efeitos fixos, CONT.PREC de fato se mostra não correlacionada com F1.NORM, de modo que podemos retirá-la do modelo final. Dos efeitos aleatórios, PARTICIPANTE foi eliminado; isso se deve ao fato de que a normalização de Lobanov tem justamente o objetivo de minimizar diferenças que se devem a características individuais. Em qualquer outra análise que não envolva normalização, a inclusão da variável falante/participante é fundamental.

Cabe comentar, por fim, como apresentar os resultados de uma análise multivariada de regressão linear. Na Tabela 13.1 se apresentam duas tabelas de Gries (2019, p. 229) como exemplos: é importante reportar todos os valores da tabela de coeficientes, bem como as medidas estatísticas que aparecem ao pé do resultado de `summary()`.

*Tabela 13.1: Exemplo de tabela de resultados (1).*

| | Soma de Quadrados | Estimativa | Erro Padrão | t | p |
|---|---|---|---|---|---|
| Intercepto | 23,61 | 2,75 | 1,52 | 1,8 | 0,08 |
| GERMAN | 2931,69 | 1,75 | 0,09 | 20,1 | <0,001 |
| CLASS | 3010,30 | -8,72 | 0,43 | -20,37 | <0,001 |
| Resíduo | 558,68 | | | | |
| $R^2$ | $R^2$ Múltiplo = 0,974 | | | $F_{2,77} = 1416$ | $p < 0,001$ |
| | $R^2$ Ajustado = 0,973 | | | | |

*Fonte: Gries (2019, p.229).*

Eis outro exemplo na Tabela 13.2: este é do artigo de Walker et al (2014), publicado na revista *Language Variation and Change*. Ao reportar modelos de efeitos mistos, é relevante indicar os efeitos aleatórios incluídos no modelo.

*Tabela 13.2: Exemplo de tabela de resultados (2).*

| | Estimate | *SE* | *t* value | *p* value |
|---|---|---|---|---|
| Intercept | -.09334 | .1316 | -.709 | .478 |
| Speaker = Puerto Rican | .16994 | .16247 | 1.046 | .296 |
| Variant = [s] | .32958 | .05556 | 5.932 | <.001 |
| Participant = Puerto Rican | -.20599 | .06993 | -2.946 | .003 |
| Speaker = Puerto Rican: Variant = [s] | -.23736 | .07228 | -3.284 | .001 |

*Note*: random effects = (1 + speaker nationality * variant | participant) + (1 + variant | speaker)

*Fonte: Walker et al (2014, p.179).*

E na Tabela 13.3 está um modelo elaborado por Elisa Battisti – um modelo que eu mesma passei a adotar em meus trabalhos por ser particularmente informativo e elegante. Para variáveis previsoras fatoriais (todas as variáveis, neste caso), além das estatísticas geradas pelo modelo, inclui-se uma coluna com o número de dados do valor de aplicação e do total para a respectiva variante da linha. Ao pé da tabela, apresenta-se o modelo completo que foi testado.

*Tabela 13.3: Exempo de tabela de resultados (3).*

Análise de regressão de efeitos mistos de haplologia com Renda Domiciliar (N = 864)
*Intercepto* = -1.087

| Variável | Estimativa | Erro padrão | Valor-*z* | *p* | Apl./N |
|---|---|---|---|---|---|
| Zona | | | | | |
| Centro (v. referência) | | | | | 65/264 (25%) |
| Leste | -0,093 | 0,488 | -0,191 | 0,848 | 25/163 (15%) |
| Norte | 2,134 | 0,900 | 2,370 | 0,018 * | 39/200 (19%) |
| Sul | 1,006 | 0,358 | 2,811 | 0,005 ** | 106/247 (43%) |
| Renda Domiciliar | | | | | |
| a (v. referência) | | | | | 110/347 (31%) |
| b1 | -0,024 | 0,352 | -0,071 | 0,943 | 83/251 (33%) |
| b2 | -3,680 | 0,998 | -3,681 | <0,001 *** | 6/70 (8%) |
| c1 | -2,281 | 0,771 | -2,975 | 0,003 ** | 36/196 (18%) |
| Sílaba um | | | | | |
| CCV (v. referência) | | | | | 20/43 (46%) |
| CV | -0,344 | 0,632 | -0,545 | 0,586 | 215/821 (26%) |
| Sílaba dois | | | | | |
| CV (v. referência) | | | | | 209/728 (29%) |
| CVC | -0,654 | 0,326 | -2,006 | 0,045 * | 26/136 (19%) |
| Tonicidade | | | | | |
| Átona-átona (v. referência) | | | | | 183/617 (30%) |
| Átona-tônica | -0,297 | 0,274 | -1,087 | 0,277 | 52/247 (21%) |

Modelo 2. HAPLOLOGIA ~ ZONA + RENDA.DOMIC + SILABAUM + SILABADOIS2 + TONICIDADE + (1|PALAVDIR) + (1|PALAVESQ) + (1|INDIVIDUO)

*Fonte: Battisti (c.p.)*

Cabe ainda ressaltar que o melhor modo de reportar os seus resultados deve seguir modelos de sua área de pesquisa específica. O melhor jeito de saber isso é *lendo artigos de periódicos renomados da sua área.*

Nesta e na última lição, vimos como realizar e ler os resultados de modelos lineares, as funções `step()` e `drop1()` que auxiliam o pesquisador a escolher as variáveis a serem incluídas ou retiradas do modelo, e uma lista de pressupostos que devem ser checados a fim de avaliar quão confiáveis são os resultados. Em especial, enfatiza-se a importância de se realizar uma análise de efeitos mistos, com a inclusão de efeitos

aleatórios, para que se possa confirmar se os efeitos fixos de fato têm influência na variável resposta. O resultado final a ser reportado deve ser, idealmente, aquele do modelo de efeitos mistos.

Deixei disponível no Anexo B um *script* com uma sugestão de roteiro de análise para um conjunto de dados de uma variável numérica, como é o caso das vogais pretônicas. Trata-se efetivamente apenas de uma sugestão – cabe a você decidir as análises que efetivamente acabará fazendo. A ideia desse roteiro é juntar o conteúdo que foi aqui apresentado ao longo de diversas lições, mas que, na prática, serão realizados em sequência por você.

**Para saber mais**

Recomendo fortemente a leitura dos capítulos 7 e 8 de Levshina (2015) para se aprofundar nos preceitos da análise de regressão linear. Esses capítulos apresentam os passos e os pressupostos de modelos lineares de modo bastante detalhado.

**Exercícios**

Nesta lista de exercícios, você vai desenvolver uma análise semelhante à que fizemos na Lição 13, mas agora sobre a vogal /o/ pretônica. Primeiro, carregue os dados da planilha Pretonicas.csv e crie um subconjunto de dados da vogal /o/ pretônica. Recodifique as variáveis CONT.PREC e CONT.SEG com os mesmos critérios empregados na recodificação dessas variáveis para a vogal /e/ pretônica.

Imagine que você tenha levantado a hipótese de que a altura da vogal /o/ pretônica (F1.NORM) depende das variáveis AMOSTRA, SEXO, CONT.PREC, CONT.SEG, F1.SEG.NORM, ESTR.SIL.PRET. A última variável ainda não foi apresentada: ESTR.SIL.PRET codifica a estrutura da sílaba em que se encontra a vogal pretônica (CV; CCV; CVr; CVs – em que C = consoante, V = vogal, r = /r/ em coda, s = /s/ em coda).

1. Entre quais pares de variáveis há colinearidade? Considere: (i) AMOSTRA; (ii) SEXO; (iii) CONT.PREC; (iv) CONT.SEG; (v) F1.SEG.NORM; (vi) ESTR.SIL.PRET.

a. entre (iii)-(vi) e entre (iv)-(vi)
b. entre (i)-(v) e entre (v)-(vi)
c. entre (iii)-(iv) e entre (iii)-(vi)
d. entre (iv)-(v) e entre (ii)-(v)

2. Entre qual par de variáveis há interação? Considere: (i) AMOSTRA; (ii) SEXO; (iii) CONT.PREC; (iv) CONT.SEG; (v) F1.SEG.NORM; (vi) ESTR.SIL.PRET.
    a. entre (iii) e (iv)
    b. entre (i) e (ii)
    c. entre (i) e (v)
    d. entre (ii) e (vi)
    e. entre (iv) e (v)
3. A partir dos dados da vogal /o/ pretônica, crie um modelo com F1.NORM como variável resposta e com todas as variáveis previsoras acima, exceto aquela que é colinear a duas outras variáveis. Inclua a interação identificada na questão 2. Quanto da variação em F1.NORM é explicada por esse modelo?
    a. 8,6%
    b. 22,9%
    c. 24,7%
4. Qual variável não apresenta correlação significativa com F1.NORM?
    a. AMOSTRA
    b. CONT.PREC
    c. CONT.SEG
    d. ESTR.SIL.PRET
    e. F1.SEG.NORM
    f. SEXO
    g. nenhuma
    h. todas

5. Calcule a estimativa de F1.NORM na fala de paraibanas quando a vogal /o/ pretônica é precedida de consoante velar e seguida de consoante palatal-sibilante (p.ex., no item lexical "gostaria").
6. Use as funções `step()` e `drop1()` para checar se as variáveis do modelo criado acima devem ser mantidas. Os testes com `step()` e `drop1()` concordam quanto às variáveis que devem ser mantidas? Justifique sua resposta.
7. Os testes "forward", "backward" e "both" de `step()` concordam quanto às variáveis que devem ser mantidas? Justifique sua resposta.
8. A função `crPlot()`, do pacote car, infelizmente não se aplica a modelos com interação. Crie um novo modelo com a inclusão das mesmas variáveis do último teste, mas sem a inclusão de interação entre variáveis. Escolha se você vai manter a variável excluída por `drop1()` ou não. Em seguida, aplique a função `crPlot()` para testar se há linearidade entre a variável resposta e variável previsora numérica do modelo.
9. Em qual intervalo da variável previsora há menor concordância entre os valores previstos e os valores observados?
    a. entre 300 Hz e 350 Hz
    b. entre 350 Hz e 400 Hz
    c. entre 400 hz e 450 Hz
    d. entre 450 Hz e 500 Hz
10. Crie um modelo de efeitos mistos, com as variáveis aleatórias PARTICIPANTE e PALAVRA, e com a inclusão das mesmas variáveis do penúltimo teste – ou seja, com a interação entre as duas variáveis identificada na questão 2. Não se assuste pois o R mostrará alguns avisos de que certas combinações entre fatores de variáveis não existem.
11. Qual variável a seguir não apresenta correlação significativa com F1.NORM no modelo de efeitos mistos?

a. AMOSTRA
b. CONT.PREC
c. CONT.SEG
d. F1.SEG.NORM
e. interação CONT.PREC:CONT.SEG
f. todas as variáveis são significativamente correlacionadas

12. A partir do modelo de efeitos mistos, calcule a estimativa da altura da vogal /o/ pretônica (F1.NORM) quando o F1 da vogal da sílaba seguinte é 450 Hz.