Lista Prática 3

Exercício 1. *Neste exercício, usaremos as funções* `runif()` *e* `rnorm()` *para gerar números aleatórios com distribuições uniforme e normal, respectivamente.*

a) *Gere o vetor* `x` *com 10.000 números aleatórios a partir de uma distribuição uniforme no intervalo [0, 10]. Qual é a média e o desvio padrão de x?*

b) *Gere o vetor* `z` *com 10.000 números aleatórios usando $z = 2x + \tilde{u}, \quad \tilde{u} \sim N(0, 4^2)$. Qual é a correlação entre x e z?*

c) *Gere o vetor $\tilde{\varepsilon}$ (*`e_til`*) com 10.000 números aleatórios a partir de uma distribuição $N(0, 6^2)$. Qual é a correlação entre $\tilde{\varepsilon}$ e cada uma das demais variáveis x e z? Além disso, verifique a correlação entre x e a soma $3z + \tilde{\varepsilon}$.*

d) *Gere o vetor* `y`*, considerando o seguinte modelo real:*

$$y = \tilde{\beta}_0 + \tilde{\beta}_1 x + \tilde{\beta}_2 z + \tilde{\varepsilon}, \tag{1.1}$$

em que $\tilde{\beta}_0 = 10$, $\tilde{\beta}_1 = 2$ e $\tilde{\beta}_2 = 3$. Agora, estime por MQO o seguinte modelo empírico:

$$y = \beta_0 + \beta_1 x + \varepsilon. \tag{1.2}$$

A estimação conseguiu recuperar $\hat{\beta}_0 \approx \tilde{\beta}_0$ e $\hat{\beta}_1 \approx \tilde{\beta}_1$? Explique.

e) *Obtenha os resíduos de MQO, $\hat{\varepsilon}$, e verifique se valem os seguintes momentos amostrais (sujeitas a algum erro de arredondamento):*

$$\frac{1}{N}\sum_{i=1}^{N} \hat{\varepsilon}_i = 0 \qquad e \qquad \frac{1}{N}\sum_{i=1}^{N} x_i \hat{\varepsilon}_i = 0$$

f) *O que os resultados do item (e) dizem sobre as condições de momento populacionais $E(\varepsilon) = 0$ e $E(x\varepsilon) = 0$?*

g) *Denote como <u>Caso I</u> o modelo real visto até agora, em que $\tilde{\beta}_2 = 3$ e $z = 2x + \tilde{u}$, em que $\tilde{u} \sim N(0, 4^2)$. Gere novamente observações z e y, e estime por MQO o modelo empírico (1.2) para cada um dos seguintes modelos reais:*

- *<u>Caso II</u>: $\tilde{\beta}_2 = -3$ e $z = 2x + \tilde{u}, \quad \tilde{u} \sim N(0, 4^2)$*
- *<u>Caso III</u>: $\tilde{\beta}_2 = 3$ e $z = -2x + \tilde{u}, \quad \tilde{u} \sim N(0, 4^2)$*
- *<u>Caso IV</u>: $\tilde{\beta}_2 = -3$ e $z = -2x + \tilde{u}, \quad \tilde{u} \sim N(0, 4^2)$*

Considerando os sinais do parâmetro da variável omitida z, $\tilde{\beta}_2$, e da sua covariância com x, $cov(x, z)$, em quais casos a estimativa do parâmetro de x é sobre-estimada ($\hat{\beta}_1 > \tilde{\beta}_1$)? E em quais é sub-estimada ($\hat{\beta}_1 < \tilde{\beta}_1$)?

Resposta:

a) $\bar{x} \approx 5$ e $dp(x) \approx 2,9$

```
N = 10000
x = runif(N, 0, 10)
mean(x)
sd(x)
```

```
[1] 5.048937
[1] 2.886117
```

b) $\bar{z} \approx 10,1$, $dp(z) \approx 7$ e $corr(x, z) \approx 0,82$

```
z = 2*x + rnorm(N, 0, 4)
mean(z)
sd(z)
cor(x, z)
```

```
[1] 10.12043
[1] 6.990625
[1] 0.819311
```

c) $\bar{\tilde{\varepsilon}} \approx 0$, $dp(\tilde{\varepsilon}) \approx 6$, $corr(x, \tilde{\varepsilon}) \approx 0$, $corr(z, \tilde{\varepsilon}) \approx 0$ e $corr(x, 3z + \tilde{\varepsilon}) \approx 0,79$

```
e_til = rnorm(N, 0, 6)
cor(x, e_til)
cor(z, e_til)
cor(x, 3*z + e_til)
```

```
[1] 0.04037083
[1] 5.95413
[1] 0.008887763
[1] 0.009301469
[1] 0.6326533
```

d) $\bar{y} \approx 50$ e $dp(y) \approx 26$

```
y = 10 + 2*x + 3*z + e_til
mean(y)
sd(y)
```

```
[1] 50.52809
[1] 26.62365
```

Agora, estimando o modelo empírico (1.2), temos

```
fit = lm(y ~ x)
fit
```

```
(Intercept)            x
     10.430        7.942
```

Como $\hat{\beta}_0 \approx 10,4 \approx 10 = \tilde{\beta}_0$ e $\hat{\beta}_1 \approx 7,9 > 2 = \tilde{\beta}_1$, a estimação conseguiu recuperar apenas $\tilde{\beta}_0$ do modelo real, enquanto $\hat{\beta}_1$ é viesado (sobre-estimado) e não recuperou $\tilde{\beta}_1$.

Isto se dá pelo viés de variável omitida, pois o modelo empírico (1.2) não incluiu z como covariada, logo $\varepsilon = 3z + \tilde{\varepsilon}$. Portanto, não é válida a hipótese $E(x\varepsilon) = 0$, o que compromete as estimativas da regressão por MQO. De fato, vimos no item (b) que $corr(x, 3z + \tilde{\varepsilon}) \approx 0,79 \neq 0$.

e) $\sum_{i=1}^{n} \hat{\varepsilon}_i \approx 0, \quad \sum_{i=1}^{n} x_i \hat{\varepsilon}_i \approx 0$

```r
sum(fit$resid)
sum(fit$resid * x)
```

```
[1] 2.039452e-12
[1] 5.794154e-11
```

f) O que se pode dizer é que o estimador de MQO faz com que $\sum_{i=1}^{n} \hat{\varepsilon}_i$ e $\sum_{i=1}^{n} x_i \hat{\varepsilon}_i$ sejam sempre iguais a zero. Isto ocorre, pois o MQO escolhe $\hat{\beta}_0$ e $\hat{\beta}_1$ que resultem em resíduos $\hat{\varepsilon}$ $(= y - \hat{y})$ que satisfaçam essas contrapartidas amostrais. No entanto, isto não quer dizer que as hipóteses $E(\varepsilon) = 0$ e $E(x\varepsilon) = 0$ sejam verdadeiras. Se $E(x\varepsilon) \neq 0$, então a estimação de $\hat{\beta}_1$ será viesada.

g) II $\hat{\beta}_1 < \tilde{\beta}_1 = 2$, quando $\tilde{\beta}_2 < 0$ e $cov(x, z) > 0$

```r
z = 2*x + rnorm(10000, 0, 4)
y = 10 + 2*x - 3*z + u
cor(x, z)
lm(y ~ x)
```

```
[1] 0.8245678

(Intercept)            x
     10.338       -4.067
```

III $\hat{\beta}_1 < \tilde{\beta}_1 = 2$, quando $\tilde{\beta}_2 > 0$ e $cov(x, z) < 0$

```r
z = -2*x + rnorm(10000, 0, 4)
y = 10 + 2*x + 3*z + e_til
cor(x, z)
lm(y ~ x)
```

```
[1] -0.8263252

(Intercept)            x
      10.14        -4.04
```

IV $\hat{\beta}_1 > \tilde{\beta}_1 = 2$, quando $\tilde{\beta}_2 < 0$ e $cov(x, z) < 0$

```r
z = -2*x + rnorm(10000, 0, 4)
y = 10 + 2*x - 3*z + e_til
cor(x, z)
lm(y ~ x)
```

```
[1] -0.819522

(Intercept)            x
     10.166        7.997
```

Portanto, assumindo os sinais de $\tilde{\beta}_2$ e $cov(x,z)$, conseguimos ao menos analisar que, em relação ao valor verdadeiro $\tilde{\beta}_1$, a estimativa

- $\hat{\beta}_1$ é sobre-estimada ($\hat{\beta}_1 > \tilde{\beta}_1$), se $\tilde{\beta}_2.cov(x,z) > 0$, e
- $\hat{\beta}_1$ é sub-estimada ($\hat{\beta}_1 < \tilde{\beta}_1$), se $\tilde{\beta}_2.cov(x,z) < 0$.

□

Exercício 2. *Neste exercício, usaremos a base de dados de Papke (1995), que possui informações sobre a participação e contribuição em planos previdência privada de empresas nos EUA, chamada de 401k:*

```
data(k401k, package="wooldridge")
```

- `prate`: *é o percentual de trabalhadores contribuindo ativamente à previdência privada.*
- `mrate`: *é a taxa de "generosidade" da empresa, isto é, a razão de quanto a empresa contribui para a previdência privada de seu funcionário.*
- `totemp`: *é número total de funcionários.*

Queremos saber a relação entre a taxa de participação de funcionários (`prate`) e a taxa de generosidade da empresa (`mrate`).

a) Estime analiticamente (sem usar a função `lm()`) o modelo:

$$prate = \beta_0 + \beta_1 mrate + \beta_2 totemp + \varepsilon$$

b) Usando operações matriciais, adapte as funções objetivo da seção de Otimização para o caso multivariado[1]. Depois, usando `optimx::opm()`, obtenha as estimativas que otimizam essas funções objetivo por

(i) Minimização da soma do quadrado dos resíduos

(ii) Método Generalizado dos Momentos (GMM), cujos momentos amostrais são

$$\begin{bmatrix} \sum_{i=1}^N \hat{\varepsilon}_i \\ \sum_{i=1}^N mrate_i.\hat{\varepsilon}_i \\ \sum_{i=1}^N totemp_i.\hat{\varepsilon}_i \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

(iii) Máxima Verossimilhança (ML)

Resposta:

[1] Lembre-se de transformar os objetos em vetores/matrizes usando `matrix()` ou `as.matrix()` antes de fazer as operações matriciais.

a)

```r
y = as.matrix(k401k[,"prate"])
X = as.matrix(cbind(1, k401k[,c("mrate", "totemp")]))

N = nrow(k401k)
K = ncol(X) - 1

bhat = solve(t(X) %*% X) %*% t(X) %*% y
yhat = X %*% bhat
ehat = y - yhat

sig2hat = as.numeric( t(ehat) %*% ehat / (N-K-1) )
Vbhat = sig2hat * solve( t(X) %*% X )
se_bhat = sqrt( diag(Vbhat) )

t_bhat = bhat / se_bhat
p_bhat = 2 * pt(-abs(t_bhat), N-K-1)
cv = qt(1 - .05/2, N-K-1)
ci = cbind(bhat - cv*se_bhat, bhat + cv*se_bhat)

round(data.frame(bhat, se_bhat, t_bhat, p_bhat, ci), 4)
```

```
          bhat se_bhat   t_bhat p_bhat      X1      X2
1      83.4307  0.5793 144.0246 0.0000 82.2944 84.5670
mrate   5.8303  0.5262  11.0798 0.0000  4.7981  6.8624
totemp -0.0001  0.0000  -2.5498 0.0109 -0.0002  0.0000
```

(i) Minimização de função perda

```r
resid_quad = function(params, fn_args) {
  # Extraindo argumentos da lista fn_args
  yname = fn_args[[1]]
  Xnames = fn_args[[2]]
  dta = fn_args[[3]]

  # Extraindo as variáveis da base em vetores
  y = as.matrix(dta[,yname])
  X = as.matrix(cbind(1, dta[,Xnames]))

  bhat = matrix(params, ncol=1)
  yhat = X %*% bhat # valores ajustados
  ehat = y - yhat # desvios = observados - ajustados
  sum(ehat^2)
}

# Otimização
min_loss = optimx::opm(par=c(0,0,0), fn=resid_quad,
      fn_args=list("prate", c("mrate", "totemp"), k401k),
      method=c("Nelder-Mead", "BFGS", "nlminb"))

round(min_loss, 4)
```

```
                 p1     p2     p3    value fevals gevals convergence
Nelder-Mead 83.4291 5.8347 -1e-04 394707.7    272     NA           0
BFGS        83.4307 5.8303 -1e-04 394707.6     52      6           0
nlminb      83.4307 5.8303 -1e-04 394707.6     60     64           0
```

(ii) GMM

```r
mom_ols1 = function(theta, fn_args) {
  # No gmm(), só pode ter 1 input dos argumentos dessa função
  # Extraindo argumentos da lista fn_args
  yname = fn_args[[1]]
  Xnames = fn_args[[2]]
  dta = fn_args[[3]]

  # Extraindo as variáveis da base em vetores
  y = as.matrix(dta[,yname])
  X = as.matrix(cbind(1, dta[,Xnames]))

  bhat = matrix(theta, ncol=1)
  yhat = X %*% bhat
  ehat = y - yhat

  ## Vetor de momentos
  m = abs(t(X) %*% ehat)

  # Ponderação
  W = diag(length(theta)) # matriz de pesos iguais
  as.numeric(t(m) %*% W %*% m)
}

# Otimização
gmm1 = optimx::opm(par=c(0,0,0), fn=mom_ols1,
  fn_args=list("prate", c("mrate", "totemp"), k401k),
  method=c("Nelder-Mead", "BFGS", "nlminb"))

round(gmm1, 4)
```

```
                   p1     p2      p3        value fevals gevals convergence
Nelder-Mead    0.0754 0.0792  0.0022 2.513261e+10     82     NA           0
BFGS          83.4307 5.8303 -0.0001 0.000000e+00    201     11           0
nlminb        83.4299 5.8309 -0.0001 2.592000e-01    108     95           0
```

(iii) Máxima Log-Verossimilhança

```r
loglik1 = function(theta, fn_args) {
  # Extraindo argumentos da lista fn_args
  yname = fn_args[[1]]
  Xnames = fn_args[[2]]
  dta = fn_args[[3]]

  # Extraindo as variáveis da base em vetores
  y = as.matrix(dta[,yname])
  X = as.matrix(cbind(1, dta[,Xnames]))

  bhat = matrix(theta[1:(length(theta)-1)], ncol=1)
  sighat = theta[length(theta)]
  yhat = X %*% bhat
  log_ypdf = dnorm(y, mean = yhat, sd = sighat, log = TRUE)

  ## Calculando a log-verossimilhanca
  loglik = sum(log_ypdf)

  ## Retornando o negativo da log-verossimilanca
  -loglik # Negativo, pois mle2() minimiza e queremos maximizar
}

```

```r
mle1 = optimx::opm(par=c(0,0,0,100), fn=loglik1,
      fn_args=list("prate", c("mrate", "totemp"), k401k),
      method=c("Nelder-Mead", "BFGS", "nlminb"))
round(mle1, 4)
```

```
                  p1     p2     p3      p4    value fevals gevals convergence
Nelder-Mead 83.4318 5.8306 -1e-04 16.0414 6433.706    267     NA           0
BFGS        83.4308 5.8302 -1e-04 16.0408 6433.706     95     25           0
nlminb      83.4307 5.8303 -1e-04 16.0408 6433.706     37    107           0
```

$\square$

Exercício 3. *Neste exercício, usaremos a base de dados* `wage1` *do pacote* `wooldridge` *e pode ser carregada no R usando o comando:*

```r
data(wage1, package="wooldridge")
```

Queremos saber a relação entre anos de estudo (`educ`) e o logaritmo da renda das pessoas (`lwage`), considerando seus sexos/gêneros.

a) Gere duas bases a partir de `wage1`: uma apenas com mulheres (`wage_female`) e outra apenas com homens (`wage_male`), e estime os seguintes modelos (sem interceptos):

$$coninc = \beta_F . educ + \varepsilon \qquad \text{(base com mulheres)}$$
$$coninc = \beta_M . educ + \varepsilon \qquad \text{(base com homens)}$$

Quais são as estimativas para β_F e β_M?

b) Usando a base completa (`wage1`), plote um gráfico de dispersão (scatterplot) entre anos de estudo × renda, colorindo os pontos de acordo com o sexo da pessoa. Também, adicione as retas das regressões feitas no item (a) com cores distintas.

c) Na base completa (`wage1`), regrida um único modelo em que, com as estimativas obtidas, possamos calcular $\hat{\beta}_F$ e $\hat{\beta}_M$ encontrados no item (a). A diferença entre $\hat{\beta}_F$ e $\hat{\beta}_M$ é estatisticamente significante?

Resposta:

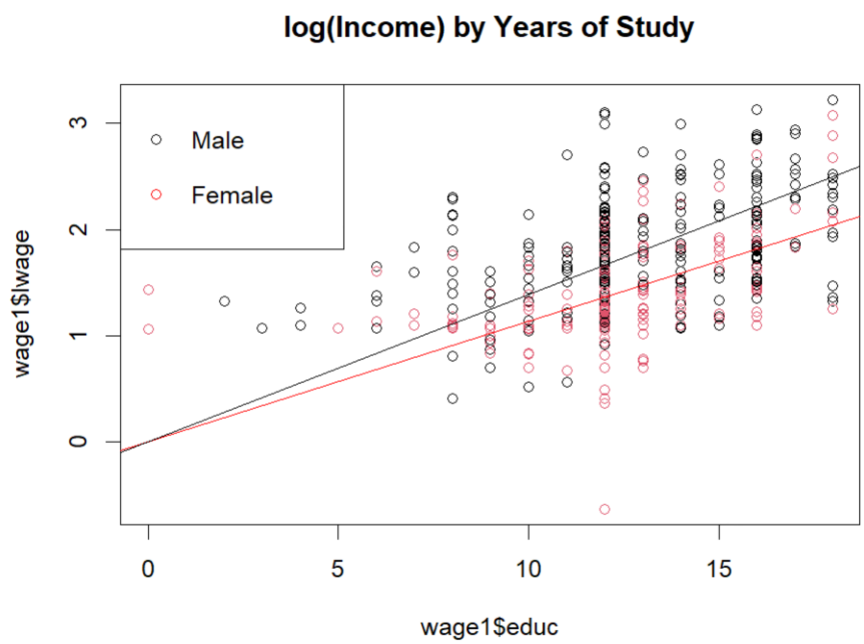
a) $\hat{\beta}_F = 0.1135$ e $\hat{\beta}_M = 0.1385$

```r
# Gerando 2 bases com trab. em tempo integral de mulheres e de homens
wage_male = wage1[wage1$female==0,]
wage_female = wage1[wage1$female==1,]

# Estimando os modelos sem constante
reg_male = lm(lwage ~ 0 + educ, wage_male)
reg_female = lm(lwage ~ 0 + educ, wage_female)

# Mostrando os betas estimados
reg_male$coef
reg_female$coef
```

```
0.1384512
0.1135278
```

b)

```r
# Plotando anos de educação X renda
plot(wage1$educ, wage1$lwage, col=wage1$female+1,
     main="log(Income) by Years of Study")
# Plotando as retas das regressões estimadas
abline(reg_female, col="red")
abline(reg_male, col="black")
legend("topleft", pch=1, col=c("black", "red"),
       legend=c("Male", "Female"))
```



c) Vamos estimar o modelo que, além da covariada de anos de estudo, vamos incluir a interação entre anos de estudo `educ` e a *dummy* `female`:

$$\text{coninc} = \beta_1.\text{educ} + \beta_2.\text{educ.female} + \varepsilon$$

```r
# Estimando o modelo com interação entre anos de educ. e dummy mulher
reg_full = lm(lwage ~ 0 + educ + educ:female, wage1)
summary(reg_full)
```

```
              Estimate Std. Error t value Pr(>|t|)
educ          0.138451   0.002161  64.077  < 2e-16 ***
educ:female -0.024923   0.003197  -7.797 3.45e-14 ***
---
Signif. codes:  0  ***  0.001  **  0.01  *  0.05  .  0.1
```

Note que $\hat{\beta}_M = \hat{\beta}_1$ e $\hat{\beta}_F = \hat{\beta}_1 + \hat{\beta}_2$. Além disso, a diferença entre as estimativas, $\hat{\beta}_2 = \hat{\beta}_F + \hat{\beta}_M$, é estatisticamente significante a $0,1\%$.

□

Exercício 4. *Analisando **apenas trabalhadores em tempo integral e parcial ("Working Fulltime" e "Working Parttime")**, verifique a possível discriminação na renda em relação aos povos hispânicos povos hispânicos (mexicanos, porto riquenhos, cubanos, etc.). Para isto use a base de dados General Social Survey (GSS)[2], que pode ser carregada no R usando o comando:*

```r
load(url("https://fhnishida.netlify.app/project/rec5004/gss.Rdata"))
```

a) *Regrida a renda (`coninc`) em relação às dummies dos povos hispânicos e liste os que possuem diferença significativa de renda em relação aos não-hispânicos. Utilize como variáveis de controle: o sexo, a idade, a idade^2, a raça/cor de pele, os anos de estudo (`educ`) e o status de trabalho (`wrkstat`).*

b) *Note que a regressão anterior "jogou fora" quase 70% das observações por causa de valores ausentes (NA's) das variáveis utilizadas, sobretudo de `hispanic`. Caso essas informações estejam faltando aleatoriamente (missing at random), pode ser razoável considerar o resultado do item (a) como representativo de toda população trabalhadora nos EUA. Para verificar isso, siga os passos abaixo:*

   i. *Para facilitar, selecione apenas as colunas/variáveis usadas na regressão do item (a)*
   ii. *Crie uma variável dummy `missing` que é igual a 1 se houver pelo menos um `NA` na linha/observação, e igual a 0 caso contrário.*
   iii. *Crie variáveis dummies para as variáveis categóricas `sex` e `race`.*
   iv. *Usando regressões de diferença de médias, verifique a significância das diferenças entre as observações retiradas da regressão do item (a) (`missing==1 / NA`), e as que foram mantidas (`missing==0 / nonNA`)*
   v. *Crie e analise a seguinte tabela com os resultados:*

	nonNA	NA	diferença	p-valor
coninc	-	-	-	-
sex_female	-	-	-	-
age	-	-	-	-
race_white	-	-	-	-
race_black	-	-	-	-
educ	-	-	-	-

Resposta:

a) Mexicanos a 0,1%, Filipinos a 5%, Argentinos e Porto Riquenhos a 10%:

```r
# Filtrando base com trabalhadores fulltime e parttime
gss = gss[gss$wrkstat %in% c("Working Fulltime", "Working Parttime"),]

# definindo Not Hispanic como referencia
gss$hispanic = relevel(gss$hispanic, ref="Not Hispanic")

```

[2]Disponibilizado por Bryan Wheeler (2014)

```r
# regressão
reg = lm(coninc ~ hispanic + sex + age + I(age^2) +
           race + educ + wrkstat, gss)
summary(reg)
```

```
                                   Estimate  Std. Error t value Pr(>|t|)
(Intercept)                       -76942.435   4104.084 -18.748  < 2e-16 ***
hispanicMexican, Mexican American  -6158.952   1711.843  -3.598 0.000322 ***
hispanicPuerto Rican               -5831.574   3195.995  -1.825 0.068082 .
hispanicCuban                       -727.544   5755.621  -0.126 0.899413
hispanicSalvadorian                -6341.522   6006.062  -1.056 0.291060
hispanicGuatemalan                 -1289.261   7959.050  -0.162 0.871319
hispanicPanamanian                -18409.278  12850.695  -1.433 0.152016
hispanicNicaraguan                -13707.093  13653.226  -1.004 0.315428
hispanicCosta Rican                15386.564  22241.821   0.692 0.489088
hispanicCentral American           -6364.602  11644.985  -0.547 0.584698
hispanicHonduran                   -3260.999   9975.247  -0.327 0.743743
hispanicDominican                 -11471.848   8454.734  -1.357 0.174857
hispanicWest Indian                 3739.035  38519.432   0.097 0.922674
hispanicPeruvian                   -6149.380  10701.734  -0.575 0.565564
hispanicEquadorian                -10500.778  10714.220  -0.980 0.327070
hispanicColumbian                  -5315.638  10301.585  -0.516 0.605864
hispanicVenezuelan                  4602.093  13621.146   0.338 0.735473
hispanicArgentinian                47851.664  27233.777   1.757 0.078935 .
hispanicSpanish                    -7198.788   5416.767  -1.329 0.183883
hispanicBasque                    -14484.157  38494.135  -0.376 0.706724
hispanicFilipino/A                 41203.692  19262.145   2.139 0.032451 *
hispanicLatin American            -12548.006  22230.069  -0.564 0.572453
hispanicSouth American             -5027.299  12859.208  -0.391 0.695843
hispanicLatino/A                  -14379.493  15748.125  -0.913 0.361215
hispanicHispanic                  -18581.959  22247.698  -0.835 0.403607
hispanicOther, Not Specified      -19508.177  14556.634  -1.340 0.180223
sexFemale                          -8196.156    766.173 -10.698  < 2e-16 ***
age                                 2946.157    176.359  16.705  < 2e-16 ***
I(age^2)                             -27.439      1.988 -13.804  < 2e-16 ***
raceBlack                         -14506.432   1119.148 -12.962  < 2e-16 ***
raceOther                           2511.628   1474.706   1.703 0.088572 .
educ                                5041.284    135.727  37.143  < 2e-16 ***
wrkstatWorking Parttime            -6199.141   1047.625  -5.917 3.38e-09 ***
---
Signif. codes:  0  ***  0.001  **  0.01  *  0.05  .  0.1

Residual standard error: 38490 on 10422 degrees of freedom
(23594 observations deleted due to missingness) <<<<<<<<<<<<<<<<<<<<<<<<<<<<<
Multiple R-squared:  0.2148,  Adjusted R-squared:  0.2124
F-statistic: 89.09 on 32 and 10422 DF,  p-value: < 2.2e-16
```

b) Note que 23.594 das 34.049 observações (69,3%) foram excluídas da regressão por causa de missing values (NA's). Então, vamos comparar as diferenças das médias de algumas variáveis para as duas amostras:

```r
library(dplyr)
gss = gss %>% select(coninc, hispanic, sex, age, race, educ, wrkstat) %>%
  mutate(
    missing = ifelse(is.na(coninc) | is.na(hispanic) | is.na(sex) | is.na(age)
    | is.na(race) | is.na(educ) | is.na(wrkstat), 1, 0)
  )

```

```
gss %>% is.na() %>% apply(2, sum) # NA's por variável
sum(gss$missing) # qtd de observ. com NA's
sum(gss$missing) / nrow(gss) # % NA's
```

```
  coninc hispanic      sex      age     race     educ  wrkstat  missing
    2600    22434        0      109        0       75        0        0
[1] 23594
[1] 0.6929425
```

Como pode ser visto acima, grande parte dos NA's ocorreram na variável `hispanic`. Em `gss`, criaremos a variável binária `missing`, que indica se a observação possui algum missing value nas variáveis utilizadas na regressão. Também, incluiremos dummies das variáveis `sex`, `race`.

```
# criando variáveis dummies
gss = fastDummies::dummy_cols(gss, c("sex", "race"))
names(gss)
```

```
 [1] "coninc"     "hispanic"   "sex"        "age"        "race"
 [6] "educ"       "wrkstat"    "missing"    "sex_Male"   "sex_Female"
[11] "race_White" "race_Black" "race_Other"
```

Agora, calcularemos as diferenças de média das variáveis de renda, sexo, idade, raça e anos de educação:

```
# diferenças de médias
vars = c("coninc", "sex_Female", "age", "race_White", "race_Black", "educ")

resultados = matrix(0, length(vars), 4) %>% .data.frame()
colnames(resultados) = c("nonNA", "NA", "dif", "p-valor")
rownames(resultados) = vars

# loop para fazer regressão e preencher tabela
for (var in vars) {
  reg = lm(eval(parse(text=var)) ~ missing, gss) %>% summary() %>% coef()
  resultados[var, 1] = reg["(Intercept)", "Estimate"]
  resultados[var, 2] = reg["(Intercept)", "Estimate"] + reg["missing", "
    Estimate"]
  resultados[var, 3] = reg["missing", "Estimate"]
  resultados[var, 4] = reg["missing", "Pr(>|t|)"]
}
resultados %>% round(3)
```

```
               nonNA        NA       dif p-valor
coninc     56103.154 48686.478 -7416.676   0.000
sex_Female     0.494     0.475    -0.019   0.002
age           41.707    39.877    -1.830   0.000
race_White     0.765     0.834     0.070   0.000
race_Black     0.137     0.130    -0.007   0.066
educ          13.881    13.210    -0.671   0.000
```

Note que apenas a proporção de negros é estatisticamente igual a 5% de significância. Logo, as observações com valores ausentes na variável `hispanic` não pa

Exercício 5. *O seguinte modelo pode ser usado para estudar se gastos na campanha afetam os resultados eleitorais:*

$$voteA = \beta_0 + \beta_1 \log(expendA) + \beta_2 \log(expendB) + \beta_3 prtystrA + \varepsilon$$

em que:

- ***voteA*** *é o percentual de votos recebidos pelo Candidato A*
- ***expendA*** *e* ***expendB*** *são gastos de campanha pelos Candidatos A e B, respectivamente*
- ***prtystrA*** *é uma medida de força do partido do Candidato A*

a) Declare a hipótese nula: o impacto nos votos do Candidato A por um aumento em 1% nos seus gastos é anulado por um aumento em 1% de gastos de B.

b) Usando a base de dados ***vote1*** *do pacote* ***wooldridge****, estime o modelo acima.*

c) Faça o teste hipótese do item (a). Dica: Você pode obter as variâncias e covariâncias das estimativas usando ***vcov()*** *no objeto de regressão gerado por* ***lm()****.*

Resposta:

a) H_0: $\beta_1 + \beta_2 = 0$

b) Usando `lm()`:

```
reg = lm(voteA ~ log(expendA) + log(expendB) + prtystrA, data=vote1)
round(summary(reg)$coef, 4)
```

```
             Estimate Std. Error  t value Pr(>|t|)
(Intercept)   45.0789     3.9263  11.4813   0.0000
log(expendA)   6.0833     0.3821  15.9187   0.0000
log(expendB)  -6.6154     0.3788 -17.4632   0.0000
prtystrA       0.1520     0.0620   2.4502   0.0153
```

c) (1) É possível fazer o teste de hipótese "na mão" usando

$$t = \frac{\hat{\beta}_1 + \hat{\beta}_2}{\sqrt{var(\hat{\beta}_1) + var(\hat{\beta}_2) + cov(\hat{\beta}_1, \hat{\beta}_2)}} = \frac{-0,532}{\sqrt{0,146 + 0,144 + 2(-0,003)}} = \frac{-0,532}{0,533} \approx 1$$

```
vcov(reg) # matriz de variâncias-covariâncias do estimador
```

```
             (Intercept) log(expendA) log(expendB) prtystrA
(Intercept)      15.4159      -0.3949      -0.8741  -0.1762
log(expendA)     -0.3949       0.1460      -0.0027  -0.0065
log(expendB)     -0.8741      -0.0027       0.1435   0.0036
prtystrA         -0.1762      -0.0065       0.0036   0.0038
```

```
var_b1 = vcov(reg)[2,2]
var_b2 = vcov(reg)[3,3]
cov_b1b2 = vcov(reg)[2,3]
sqrt( var_b1 + var_b2 + 2*(cov_b1b2) )
```

```
[1] 0.5330858
```

(2) Também é possível fazer o teste em uma regressão definindo $\theta = \beta_1 + \beta_2 \iff \beta_1 = \theta - \beta_2$. Substituindo β_1 no modelo e isolando θ segue que:

$$\begin{aligned} voteA &= \beta_0 + \beta_1 \log(expendA) + \beta_2 \log(expendB) + \beta_3 prtystrA + \varepsilon \\ &= \beta_0 + (\theta - \beta_2) \log(expendA) + \beta_2 \log(expendB) + \beta_3 prtystrA + \varepsilon \\ &= \beta_0 + \theta \log(expendA) + \beta_2[\log(expendB) - \log(expendA)] + \beta_3 prtystrA + \varepsilon \end{aligned}$$

```
reg2 = lm(voteA ~ log(expendA) + I(log(expendB)-log(expendA)) + prtystrA,
          data=vote1)
round(summary(reg2)$coef, 4)
```

```
                             Estimate Std. Error  t value Pr(>|t|)
(Intercept)                   45.0789     3.9263  11.4813   0.0000
log(expendA)                  -0.5321     0.5331  -0.9982   0.3196
I(log(expendB) - log(expendA)) -6.6154     0.3788 -17.4632   0.0000
prtystrA                       0.1520     0.0620   2.4502   0.0153
```

Agora, basta verificar a estatística t (ou p-valor) de $\hat{\theta}$.

□

Exercício 6. *Use a base de dados sobre salários na liga americana de beisebol* `mlb1`*, do pacote* `wooldridge`*, para este exercício e considere o modelo:*

$$\log(salary) = \beta_0 + \beta_1 years + \beta_2 gamesyr + \beta_3 bavg + \beta_4 hrunsyr + \varepsilon$$

em que:

- `salary` *é o salário do jogador*
- `years` *é a quantidade de anos como jogador profissional*
- `gamesyr` *é a média de jogos por ano do jogador*
- `bavg` *é o percentual de rebatida*
- `hrunsyr` *é a média de home runs por ano*

a) *Adicione* `runsyr` *(corridas por ano),* `fldperc` *(percentual de defesa), e* `sbasesyr` *(bases roubadas por ano) ao modelo. Quais destes fatores são individualmente significantes?*

b) *A partir do modelo modelo do item (a), teste a significância conjunta de* `bavg`*,* `fldperc`*, e* `sbasesyr`*. Faça "na mão" os dois testes possíveis para este caso e analise-os.*

Resposta:

a) Apenas `runsyr` é estatisticamente significante.

```
reg3 = lm(log(salary) ~ years + gamesyr + bavg + hrunsyr + runsyr
          + fldperc + sbasesyr, mlb1)
round(summary(reg3)$coef, 4)
```

```
               Estimate Std. Error t value Pr(>|t|)
(Intercept)  10.4083     2.0033  5.1957   0.0000
years         0.0700     0.0120  5.8440   0.0000
gamesyr       0.0079     0.0027  2.9504   0.0034
bavg          0.0005     0.0011  0.4798   0.6317
hrunsyr       0.0232     0.0086  2.6867   0.0076
runsyr        0.0174     0.0051  3.4344   0.0007
fldperc       0.0010     0.0020  0.5164   0.6059
sbasesyr     -0.0064     0.0052 -1.2382   0.2165
```

b) (1) Primeiro, vamos testar as $G = 3$ restrições conjuntamente ($\beta_3 = \beta_6 = \beta_7 = 0$) por meio do teste de Wald:

$$w(\hat{\boldsymbol{\beta}}) = \left[\boldsymbol{R}\hat{\boldsymbol{\beta}} - \boldsymbol{h}\right]' \left[\boldsymbol{R}\boldsymbol{V}_{\hat{\boldsymbol{\beta}}}\boldsymbol{R}'\right]^{-1} \left[\boldsymbol{R}\hat{\boldsymbol{\beta}} - \boldsymbol{h}\right] \sim \chi^2_{(G)}$$

```
bhat = matrix(reg3$coef, ncol=1)
Vbhat = vcov(reg3)
G = 3 # Número de restrições
# Matriz das restrições
R = matrix(c(0, 0, 0, 1, 0, 0, 0, 0,
             0, 0, 0, 0, 0, 0, 1, 0,
             0, 0, 0, 0, 0, 0, 0, 1),
           nrow=G, byrow=TRUE)
# Vetor de constantes h
h = matrix(c(0, 0, 0),
           nrow=3, ncol=1)
# Estatística de Wald
w = t( R %*% bhat - h ) %*% solve( R %*% Vbhat %*% t(R) ) %*% (R %*% bhat - h)
1 - pchisq(w, df=G) # p-valor
```

```
[1,] 0.5610675
```

(2) Agora, verificaremos pelo teste F

$$F = \frac{\text{SSR}_r - \text{SSR}_{ur}}{\text{SSR}_{ur}} . \frac{N - K - 1}{G} = \frac{R^2_{ur} - R^2_r}{1 - R^2_{ur}} . \frac{N - K - 1}{G}$$

```
N = nrow(mlb1)
K = length(reg3$coef) - 1
reg3r = lm(log(salary) ~ years + gamesyr + hrunsyr + runsyr, mlb1)
r2ur = summary(reg3)$r.squared
r2r = summary(reg3r)$r.squared
F = ( r2ur - r2r ) / (1 - r2ur) * (N-K-1) /  G
1 - pf(F, G, N-K-1) # p-valor
```

```
[1] 0.5617088
```

Portanto, não rejeitamos a hipótese nula $\beta_3 = \beta_6 = \beta_7 = 0$

□