SME0806 - Estatística Computacional - Trabalho 2

Matheus Victal Cerqueira

17/06/2021

Alunos:

- Aline Fernanda da Conceição, 9437275
- Diego J Talarico Ferreira, 3166561
- Matheus Victal Cerqueira, 10276661
- Murilo Henrique Soave, 10688813
- Nelson Calsolari Neto, 10277022

Docente: Professor Dr. Mário de Castro

Introdução

O presente documento se trata de uma solução para os exercícios propostos no Trabalho 2 da disciplina SME0806 - Estatística Computacional, oferecida pelo Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo no primeiro semestre de 2021. As temáticas abordadas são métodos computacionais de reamostragem para a obtenção de estimativas pontuais e intervalares. O nível de confiança das estimativas intervalares aqui realizadas foi $\gamma = 95\%$ e a seed, 2112.

Desenvolvimento e Metodologia

Exercício 1

No presente exercício, quer-se analisar o coeficiente de Gini para o Produto Interno Bruto (PIB) *per capita* dos municípios do estado de São Paulo. Para tal análise, foi considerado que os valores observados para tais municípios correspondiam à uma amostra de observações e em tal amostra performou-se um processo de reamostragem baseado em *bootstrap* para obter-se estimativas pontuais e intervalares para o coeficiente de desigualdade. O sistema de referência utilizado foi o de uma pesquisa de PIB municipal no estado de Sâo Paulo realizada pelo Instituto Brasileiro de Geografia e Estatística - IBGE; Fundação Seade em 2018. A função para obter o coeficiente de Gini foi implementada em R e encontra-se abaixo.

```
# Coeficiente de Gini
Gini <- function(v){ # função recebe vetor de interesse v
  mu <- mean(v)
  n <- length(v)
  m_sub <- outer(v,v, FUN = '-')# obtém uma matriz com todas as subtrações
  #possíveis entre os termos de v (xi-xj); i,j = {1,2,...,n}
  mod_m_sub <- abs(m_sub) # obtém a matriz com o módulo das subtrações
  somatorio <- sum(mod_m_sub) # obtém o somatório dos elementos de mod_m_sub
  gini <- somatorio/(2*n^2*mu) # cálculo do coeficiente de Gini
```

```
  return(gini)
}
```

Com a função para obtenção do coeficiente de Gini em mãos, pode-se performar o processo de reamostragem bootstrap para realizar estimativas. Como serão feitas estimativas intervalares, será necessária a obtenção da dsitribuição *t bootstrap*, o que demanda um procedimento de bootstrap iterado de dois níveis, como mostrado abaixo.

Bootstrap iterado

```
data <- read.csv("dadosIBGE_f.csv",
                 header = TRUE,
                 encoding = "UTF-8") # dados dos municípios de SP, IBGE

data <- data.frame(data)

PIBpc <- data$PIBpc # valores do PIB per capita para todos os municípios


## Bootstrap iterado

set.seed(2112)
n <- length(PIBpc) # número de observações (municípios: 645)
G <- Gini(PIBpc) # estimativa pontual do coef de Gini
B1 <- 200 # Número de amostras bootstrap no primeiro nível
B2 <- 100 # Número de amostras bootstrap no segundo nível
t <- c() # vetor vazio para receber os valores de t* (t bootstrap)

system.time({ # Obter tempo de compilação do segmento
  # Nível 1 (N1)
  ge1 <- c() #vetor para valores do Gini estimado para cada amostra boostrap(N1)

  for(b1 in 1:B1){ # laço para cada amostra bootstrap obtida no nível 1
    # Amostra bootstrap (nível 1)
    ab1 <- sample(PIBpc, n, replace = TRUE)
    # Estimativa bootstrap (*)
    ge1[b1] <- Gini(ab1)

    # Nível 2 (N2)
    ge2 <- c() #recebe valores do Gini estimado para cada amostra boostrap(N2)

    for(b2 in 1:B2){
      # Amostra bootstrap (nível 2)
      ab2 <- sample(ab1, n, replace = TRUE)
      # Estimativa bootstrap (**)
      ge2[b2] <- Gini(ab2)
    }

    t[b1] <- (ge1[b1] - G)/sd(ge2) # obtanção dos valores da distribuição t*
  }
})
```

```
##    user  system elapsed
##  153.09   68.64  227.36
```

Agora, tem-se em mãos a distribuição estimativas do coefifiente de Gini em amostras bootstrap, a dsitribuição t^*, e a estimativa G obtida pela aplicação direta da função Gini no vetor de dados $PIBpc$, quantidades necessárias para a obtenção das estimativas de interesse.

Obtenção de estimativas pontuais e intervalares

```
# Estimativas pontuais
epb <- sd(ge1) # erro padrão bootstrap
gb_cvies <- 2 * G - mean(ge1)  # estimativa bootstrap com correção de viés
gb_vies <- mean(ge1) # estimativa bootstrap
est_Gini <- G # estimativa direta com a função Gini sobre o vetor PIBpc
```

```
## Erro padrão bootstrap: 0.01611714
```

```
## Estimativa direta: 0.3262675
```

```
## Estimativa Bootstrap (sem correção de viés): 0.3245067
```

```
## Estimativa Bootstrap (com correção de viés): 0.3280282
```

É notório que as estimativas estão relativamente próximas. Sendo que as estimativas bootstrap com correção de viés e sem correção de viés apresentaram diferenças similares, em módulo, em relação à estimativa denominada direta.

```
# Estimativas intervalares
conf <- .95 # nível de confiança de 95%
coefq <- c(1-conf,1+conf)/2 # coeficientes de probabilidade dos quantis
cat("Intervalo percentil: IC[G,",100*conf,"%] = [",
    quantile(ge1, probs = coefq, type = 6),"]" )
```

```
## Intervalo percentil: IC[G, 95 %] = [ 0.2909147 0.3565151 ]
```

```
qs12 <- quantile(t, probs = coefq, type = 6) # quantis da distribuição de t*
ictb <- G - qs12[2:1]*epb # intervalo de confiança usando q12
cat("Intervalo t* (t bootstrap): IC[G,",100*conf,"%] = [",ictb,"]" )
```

```
## Intervalo t* (t bootstrap): IC[G, 95 %] = [ 0.2995715 0.375705 ]
```

O intervalo percentil é de obtenção computacionalmente menos intensa do que o pela dsitribuição t^*. Porém, recomenda-se a obtenção de ambas, quando viável, como feito neste documento. A amplitude dos intervalos aqui obtidos foi bem parecida, com um leve deslocamento para a diretia do intervalo t^* em relação ao intervalo percentil.

Nas estimativas encontradas, o índice de Gini apresentou um valor médio menor do que observamos pelo cálculo realizado pelo IBGE. Isso se deve pela forma em que os dados foram coletados. Aqui consideramos a renda per capita enquanto na estimativa do IBGE utiliza-se os dados de renda das famílias a partir da PNAD nos dois casos utiliza-se o bootstrap para estimar as medidas de precisão.

O último índice de Gini calculado pelo IBGE para o estado de SP foi 0,533 (PNAD 2016/2017 https://agenciadenoticias.ibge.gov.br/media/com_mediaibge/arquivos/bd466f98f27dac67181148ebe5d960de.pdf) enquanto o valor máximo estimado a partir do intervalo t bootstrap foi de 0,3749.

Exercício 2

O exercício 2 sugere a utilização de uma medida de associação para a comparação entre colunas referentes à categoria "Valor Adicionado" e, após a identificação das colunas mais fortemente asociadas, a obtenção de estimativas pontuais e intervalares para o valor da medida escolhida entre tais colunas. Aqui, optou-se pela correlação de Pearson, a qual já está implementada em R e é muito bem otimizada, permitindo a obtenção de mais amostras bootstrap em tempo viável.

```
cor(data[,c(2,3,4,5,6)]) # matriz de correlações das colunas de "Valor Adicionado"
```

```
##                Agropecuaria  Industria     Adm_pub T_ex_adm_pub         Total
## Agropecuaria  1.0000000000 0.01031106 0.006243629 -0.005827323 -0.0006927772
## Industria     0.0103110586 1.00000000 0.921608723  0.899280372  0.9220296264
## Adm_pub       0.0062436293 0.92160872 1.000000000  0.992062090  0.9951054886
## T_ex_adm_pub -0.0058273227 0.89928037 0.992062090  1.000000000  0.9984369962
## Total        -0.0006927772 0.92202963 0.995105489  0.998436996  1.0000000000
```

As colunas mais fortemente associadas pela correlação de Pearson são: "Total" e "Total exceto administração pública". Então a estimativa para a correlação de Pearson será feita emr elação à elas. Novamente, aplica-se o bootstrap iterado para a obtenção dos valores necessários para a obtenção das estimativas de interesse.

```
x <- data$T_ex_adm_pub # vetor de dados de Total exceto administração pública
y <- data$Total # vetor de dados de Total

set.seed(2112)
n <- length(data$Municipios) # número de observações (municípios: 645)
pearson <- cor(x,y) # estimativa direta do pearson
B1 <- 2000 # Número de amostras bootstrap no primeiro nível 1
B2 <-  100 # Número de amostras bootstrao no segundo nível 2
t <- c() # vetor vazio para receber os valores de t* (boostrap)

system.time({ # Obter tempo de compilação do segmento
  # Nível 1 (N1)
  pe1 <- c() #vetor para valores do pearson estimada para cada amostra
  # boostrap(N1)

  for(b1 in 1:B1){
    # Amostra bootstrap dos indices para os vetores x e y (nível 1)
    indic1 <- sample(n, n, replace = TRUE)
    # Estimativa bootstrap (*)
    pe1[b1] <- cor(x[indic1],y[indic1])

    # Nível 2 (N2)
    pe2 <- c() #recebe valores da cor estimada para cada amostra boostrap(N2)

    for(b2 in 1:B2){
      # Amostra bootstrap dos índices para os vetores x e y, a partir de
      # indic1 (nível 2)
      indic2 <- sample(indic1, n, replace = TRUE)
      # Estimativa bootstrap (**)
      pe2[b2] <- cor(x[indic2],y[indic2])
    }
```

```
    t[b1] <- (pe1[b1] - pearson)/sd(pe2) # valores de t*
  }
})
```

```
##    user  system elapsed
##   49.32    0.03   51.03
```

```
# Estimativas pontuais
epb <- sd(pe1) # erro padrão bootstrap
pb_vies <- mean(pe1) # estimativa bootstrap
est_Pearson <- pearson # estimativa direta
```

```
## Erro padrão bootstrap: 0.0093782
```

```
## Estimativa direta: 0.998437
```

```
## Estimativa Bootstrap (sem correção de viés): 0.9922386
```

A estimativa com correção de viés resultou em um valor levemente acima de 1, o que não é possível para a correlação de Pearson. Assim, removemos tal estimativa do relatório, mas, devido ao alto valor da correlação, era esperado que erros amostrais tivessem algum efeito do gênero.

Também utilizamos as estimativas a partir do bootstrap para obter maiores precisões em intervalos de confiança, com o segue abaixo.

```
# Estimativas intervalares
conf <- .95 # nível de confiança de 95%
coefq <- c(1-conf,1+conf)/2 # coeficientes de probabilidade dos quantis
cat("Intervalo percentil: IC[pearson,",100*conf,"%] = [",
    quantile(pe1, probs = coefq, type = 6),"]" )
```

```
## Intervalo percentil: IC[pearson, 95 %] = [ 0.9723363 0.9996156 ]
```

```
qs12 <- quantile(t, probs = coefq, type = 6) # quantis da distribuição de t*
ictb <- pearson - qs12[2:1]*epb # intervalo de confiança usando qs12
cat("Intervalo t* (t bootstrap): IC[pearson,",100*conf,"%] = [",ictb,"]" )
```

```
## Intervalo t* (t bootstrap): IC[pearson, 95 %] = [ 0.9943835 1.043458 ]
```

O intervalo t bootstrap teve seu valor superior maior do que um, o que não corresponde à um valor possível. Assim, podeíamos truncá-lo de forma a ter:

$$IC[\rho, 95\%] \approx [0.9943835; 1]$$

A amplitude dos intervalos aqui obtidos foi relativamente parecida, com um leve deslocamento para a direta do intervalo t^* em relação ao intervalo percentil. Porém, o intervalo t^* é notavelmente mais preciso, como esperado pelos resultados teóricos. Também observamos que a correlação entre os valores adicionados foi alta em todas as formas estimadas. O que é explicado dado que os totais (tanto bruto quanto com a exceção do valor de administração pública) são dependentes da soma dos demais valores.