Trabalho 1 - SME0806 - Estatística Computacional

Universidade de São Paulo

Diego G. de Paulo (10857040)
Caio Henrique M. Schiavo (11810602)
Vitor Gratiere Torres (10284952)
Douglas Sudré Souza (10733820)
Bruno H.da S Justino (11031621)

21/05/2021

Contents

Introdução

Após a apresentação e explicação do conteúdo da disciplina, foi solicitado, pelo professor, um trabalho com base em algoritmos de amostras pseudo-aleatórias (Método da Transformação e Método da Rejeição) e a aplicação do Método de Monte Carlo para estimar, por simulações, parâmetros e funções de variáveis aleatórias.

A simulação de um modelo probabilístico consiste na geração de mecanismos estocásticos e, em seguida, na observação do fluxo resultante do modelo ao longo do tempo. No Método de Monte Carlo, nome originado pelo uso da aleatoriedade e da natureza repetitiva das atividades realizadas em cassinos em Monte Carlo, Mônaco, representa-se a solução de um problema como um parâmetro de uma população hipotética e, que usa uma sequência aleatória de números para construir uma amostra da população da qual estimativas estatísticas desse parâmetro possam ser obtidas. Halton (1970).

Neste trabalho, também nos deparamos com distribuições do tipo Log-Normal, caracterizada pela propriedade que os logaritmos dos valores seguem uma distribuição normal e pela forte assimetria positiva, dada pela ocorrência de uma grande quantidade de valores baixos e uma pequena quantidade de valores altos a muito altos. (Koch e Link, 1970, p. 213)

Exercício 1

Motivação

O intuito deste exercício é gerar uma amostra pseudo-aleatória de $f(x)$ dada por $f(x) \propto q(x) = e^{\frac{-|x|^3}{3}}$.

Metodologia

Para gerar tal amostra, foi selecionado o método da rejeição. Este método é descrito por:

A seleção de uma variável aleatória Y, com função de densidade dada por $g(y)$ amostrável. Além disso, há a suposição:

- $\frac{f(x)}{g(x)} \leq M$, $1 \leq M < \infty$

E, por recomendação, toma-se $M = max_x \left(\frac{f(x)}{g(x)}\right)$

Resolução

Para o exercício em questão, seleciona-se Y, tal que $Y \sim Laplace(0, 1)$ que tem a função de probabilidade dada por: $g(y) = \frac{1}{2}e^{-|y|}, y \in \mathbb{R}$. Para obter M, tem-se:

$$M = max_x \left(\frac{f(x)}{g(x)}\right) \iff \frac{d\left(\frac{e^{\frac{-|x|^3}{3}}}{\frac{1}{2}e^{-|x|}}\right)}{dx} = 0$$

Calculando a derivada de $\frac{f(x)}{g(x)}$:

$$\frac{d\left(\frac{f(x)}{g(x)}\right)}{dx} = \frac{d\left(\frac{e^{\frac{-|x|^3}{3}}}{\frac{1}{2}e^{-|x|}}\right)}{dx} = \frac{2e^{\frac{-|x|^3+3|x|}{3}}x(-x^2+1)}{|x|}$$

Igualando a zero:

$$\frac{2e^{\frac{-|x|^3+3|x|}{3}}x(-x^2+1)}{|x|} = 0$$

Como solução para esta equação tem-se: $x = -1, 0, 1$, afim de não postergar o cálculo e partir para o gerador de amostra pseudo-aleatória, seleciona-se, dos pontos críticos, apenas os pontos de máximo em $x = -1, 1$. Sendo assim obtém-se: $M = max_x \left(\frac{f(x)}{g(x)}\right) = 2e^{\frac{2}{3}}$. Finalizada a etapa de seleção das variáveis, segue a apliacação das etapas:

- 1º: Gerar uma amostra de Y
- 2º: Gerar $u \sim U(0, 1)$
- 3º: Se $u \leq \frac{f(y)}{Mg(y)}$ faça $x = y$, Caso contrário retornar ao primeiro passo.
- 4º: Repita os passos anteriores até obter n observações necessárias.

Gráficos

Abaixo, para melhor visualização dos resultados obtidos matematicamente, estão exibidos os gráficos das funções $f(x)$, $g(y)$, assim como os gráficos de $\frac{f(x)}{g(x)}$, para que os pontos critícos possam ser observados, e o gráfico sobreposto de $f(x)$ e $Mg(x)$, para notar o envolapamento de $f(x)$ por $Mg(x)$.

```
gx <- function(x) {

  return(0.5 * exp(-abs(x)))

}

fx <- function(x) {

  return(exp(-(abs(x)^3)/3))

}
```

```
fgx <- function(x){

  return(fx(x)/gx(x))

}

M <- fgx(-1)



M_gx <- function(x, M) {
  return(M * gx(x))
}
par(mfrow=c(2,2))
curve(fx, -5, 5, xlab = "x", ylab = "f(x)",
      col = "darkgrey", lwd = 2, main = "Gráfico de f(x)")

curve(gx, -5, 5, xlab = "y", ylab = "g(y)",
      col = "darkgrey", lwd = 2, main = "Gráfico de g(y)")

curve(fgx, -4, 4, ylab = "f(x)/g(x)", main = "Gráfico de f(x)/g(x)")
points(c(-1, 1), c(M, M), pch = 20, col = "blue")
abline(h = M, lty = 2, col = "blue")
segments(c(-1, 1), c(0, 0), c(-1, 1), c(M, M), lty = 2, col = "blue")

curve(M_gx(x, M), -5, 5, col = "red", ylab = "f(x) e M*g(x)", main = "Gráfico de f(x) e Mg(x)"
curve(fx(x), add = TRUE)
legend("topright", c("f(x)", "M*g(x)"), col = c("black", "red"),
```

```
lty = 1, bty = "n")
```

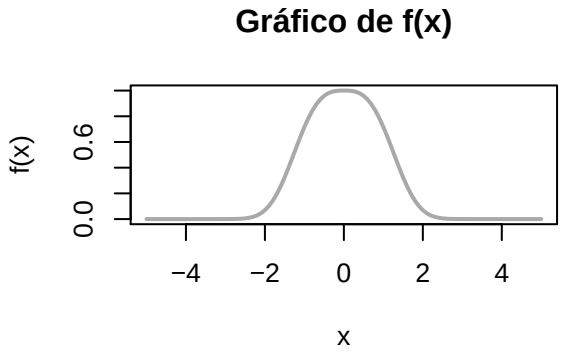


Gráfico de f(x)
f(x)
0.0
0.6
−4
−2
0
2
4
x

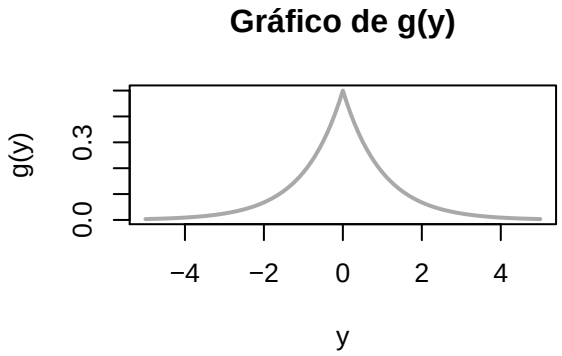


Gráfico de g(y)
g(y)
0.0
0.3
−4
−2
0
2
4
y

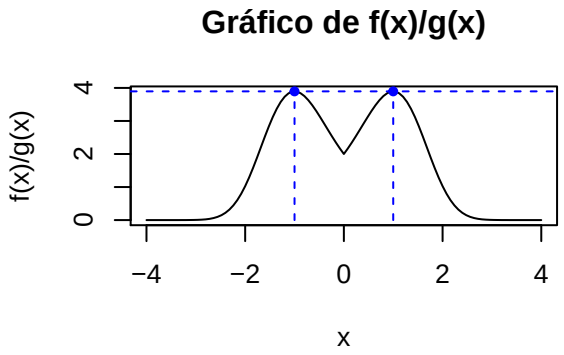


Gráfico de f(x)/g(x)
f(x)/g(x)
0
2
4
−4
−2
0
2
4
x

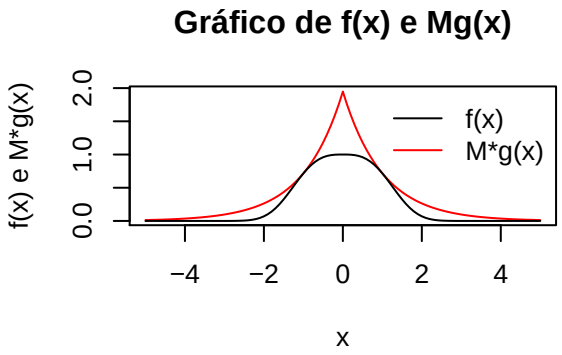
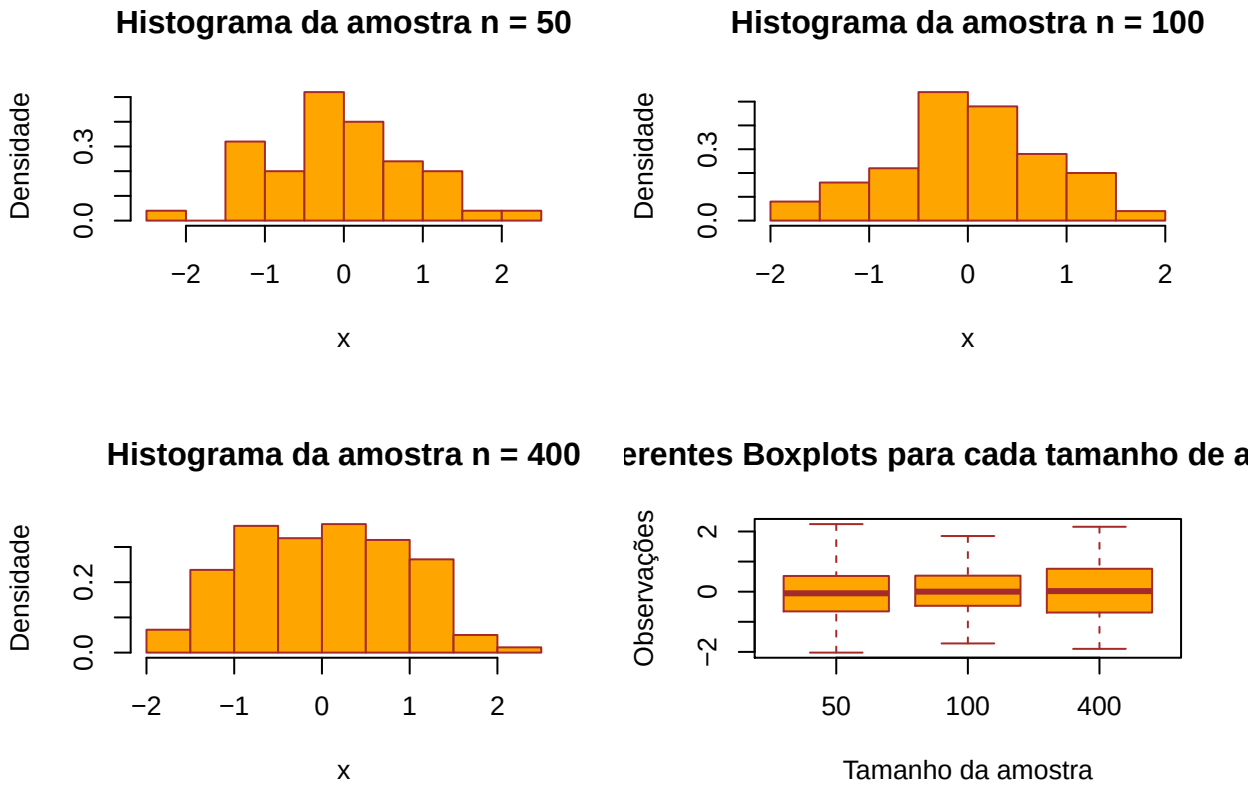


Gráfico de f(x) e Mg(x)
f(x) e M*g(x)
0.0
1.0
2.0
f(x)
M*g(x)
−4
−2
0
2
4
x

A seguir, é possível observar histogramas e boxplots para cada tamanho de amostra $n = (50, 100, 400)$ das amostras pseudo-aleatórias geradas da $f(x)$. É possível notar, observando o gráfico, que a medida em que se aumenta o tamanho da amostra mais próximo da simetria observada na função, dispõe-se a amostra.

```r
gerador <- function(n){

  nger <- n0 <- 0
  ax <- c()
  while (n0 < n) {
    rej <- TRUE
    while(rej) {
      nger <- nger + 1
      u <- runif(1)
      y <- ifelse(u <= 0.5, log(2 * u), -log(2 * (1 - u)))

      if (M * runif(1) <= fgx(y)) {
        n0 <- n0 + 1
        ax[n0] <- y
        rej <- FALSE
      }
    }
  }

  result <- list(sample = ax, n_gerado = nger)
  return(result)

}

amostra_gerada_50 <- gerador(50)
ax_50 <- amostra_gerada_50$sample

amostra_gerada_100 <- gerador(100)
ax_100 <- amostra_gerada_100$sample

amostra_gerada_400 <- gerador(400)
ax_400 <- amostra_gerada_400$sample

par(mfrow=c(2,2))
hist(ax_50, freq = FALSE, main = "Histograma da amostra n = 50",
     xlab = "x", ylab = "Densidade",
     col="orange", border="brown")
hist(ax_100, freq = FALSE, main = "Histograma da amostra n = 100",
     xlab = "x", ylab = "Densidade",
     col="orange", border="brown")
hist(ax_400, freq = FALSE, main = "Histograma da amostra n = 400",
     xlab = "x", ylab = "Densidade",
     col="orange", border="brown")
```

```r
amostra <- c(ax_50, ax_100, ax_400)
tamanho <- c(rep(50, length(ax_50)),
             rep(100, length(ax_100)),
             rep(400, length(ax_400)))

df <- data.frame(amostra, tamanho)

boxplot(amostra~tamanho,
data=df,
main="Diferentes Boxplots para cada tamanho de amostra",
xlab="Tamanho da amostra",
ylab="Observações",
col="orange",
border="brown"
)
```



Exercício 2

Motivação

Uma aproximação pode ser obtida utilizando simulações de amostras de (X, Y). Vamos gerar observações do par (X, Y) com as expressões que estão no enunciado.

Resolução

```
R <- 10000
S <- 200

mean_hat <- c()
for(i in 1:R){
  x <- rlnorm(S,0,1)
  erro <- rnorm(S,0,1)
  y <- exp(9+3*log(x)+erro)
  mean_hat[i] <- mean(y/x)
}
```

Estimação Pontual

Na estimação pontual desejamos encontrar um único valor numérico que esteja bastante próximo do verdadeiro valor do parâmetro.

```
mean(mean_hat)
```

```
## [1] 99854.96
```

Estimação Intervalar

Embora os estimadores pontuais especifiquem um único valor para o parâmetro, diferentes amostras levam a diferentes estimativas, pois o estimador é uma função de uma amostra aleatória. E, estimar um parâmetro através de um único valor não permite julgar a magnitude do erro que podemos estar cometendo.

Daí, surge a ideia de contruir um intervalo de valores que tenha uma alta probabilidade de conter o verdadeiro valor do parâmetro (denominado intervalo de confiança).

```
quantile(mean_hat, c(0.025, 0.975))
```

```
##      2.5%     97.5%
##  44054.94 255157.93
```

Poderiamos resolver de diversas outras formas, de maneira que, com o método utilizado, estamos 95% confiantes de que o intervalo de 43703.07 a 252929.58 realmente contém o verdadeiro valor de p.

Exercício 3

Motivação

Neste presente exercício busca-se trabalhar com a distribuição poisson e estudar os Erros do Tipo I e II para esta distribuição utilizando o Método de Monte Carlo.

Resolução

Para facilitar a compreensão, será indicado alguns aspectos importantes de tal distribuição. Dado $X \sim Poisson(\lambda)$, $X_1, ..., X_n$ é uma amostra aleatória de X de tamanho n.

- $\mathbb{P}(X = k) = \frac{e^{-\lambda}\lambda^x}{x!}$, para $\lambda > 0$ e $x = 0, 1, 2, ...$;
- $\mathbb{E}(X) = \lambda$;
- $\text{Var}(X) = \lambda$;
- $\hat{\lambda}_{EMV} = \frac{1}{n}\sum_{i=1}^{n} X_i = \bar{X}$, em que k_i é a i-ésima observação amostral de X_i;

Pela propriedade da eficiêcnia de um estimador de máxima verossimilhança, tem-se o resultado:

$$\sqrt{n}(\hat{\lambda}_{EMV} - \lambda) \xrightarrow{D} \mathcal{N}(0, \mathcal{IF}(\lambda)^{-1})$$

em que $\mathcal{IF}(\lambda) = \frac{1}{\lambda}$

Para o teste de hipótese:

- $H_0 : \lambda = 2$
- $H_A : \lambda > 2$

A Estatística do Teste T pode ser definida a partir do resultado da efiência do estimador de máxima verossimilhança.

$$T = \sqrt{n}(\hat{\lambda}_{EMV} - 2) \xrightarrow[\text{Sob } H_0]{D} \mathcal{N}\left(0, \frac{1}{2}\right)$$

```
R <- 10000
lambda <- 2
po_sample <- function(R, lambda, n){

  pvalue <- c()
  aceptance <- c()
  medias_v <- c()
  poder_dif <- seq(2.2, 4, length.out = 5)-2
  power_1 <- c()

  for (i in 1:R) {

    sample_p <- rpois(n, lambda)
    media <- mean(sample_p)
```

```
  p_value <- pnorm(media, mean = lambda, sd = sqrt(1/n*lambda))
  pvalue[i] <- p_value

  medias_v[i] <- media

  if (pvalue[i] >= 0.05) {
    aceptance[i] <- 1
  }
  else{
    aceptance[i] <- 0
  }
}

poder_1 <- power.t.test(n = n, delta = poder_dif[1],
                        sd = sqrt(1/n*lambda),
                        sig.level = .05,
                        alternative = "two.sided",
                        type = "one.sample")
power_1[1] <- (as.numeric(unlist(poder_1[5])))

poder_2 <- power.t.test(n = n, delta = poder_dif[2],
                        sd = sqrt(1/n*lambda),
                        sig.level = .05,
                        alternative = "one.sided",
                        type = "one.sample")
power_1[2] <- as.numeric(unlist(poder_2[5]))

poder_3 <- power.t.test(n = n, delta = poder_dif[3],
                        sd = sqrt(1/n*lambda),
                        sig.level = .05,
                        alternative = "one.sided",
                        type = "one.sample")
power_1[3] <- as.numeric(unlist(poder_3[5]))

poder_4 <- power.t.test(n = n, delta = poder_dif[4],
                        sd = sqrt(1/n*lambda),
                        sig.level = .05,
                        alternative = "one.sided",
                        type = "one.sample")
power_1[4] <- as.numeric(unlist(poder_4[5]))

poder_5 <- power.t.test(n = n, delta = poder_dif[5],
                        sd = sqrt(1/n*lambda),
                        sig.level = .05,
                        alternative = "one.sided",
                        type = "one.sample")
power_1[5] <- as.numeric(unlist(poder_5[5]))
```

```
  results <- list(pvalor = pvalue, aceita = aceptance,
                  media = medias_v, poder = power_1)

  return(results)

}


a <- po_sample(R, lambda, 10)
b <- po_sample(R, lambda, 30)
c <- po_sample(R, lambda, 75)
d <- po_sample(R, lambda, 100)
```

Tabelas

Na tabela subsequente, é possível notar a tendência esperada, conforme o o tamanho n da amostra aumenta, mais próximo de 95% fica a taxa de aceitação de H_0. Este resultado é esperado pois, pela definição do teste de hipótese estatístico, ao fixarmos $\alpha = 0.05$ espera-se que em 95% dos casos testados sejam em direção a aceitação de H_0 se a suposição de normalidade dos dados realmente é válida.

```
freqa <- table(a$aceita) * 100 / sum(table(a$aceita))
freqb <- table(b$aceita) * 100 / sum(table(b$aceita))
freqc <- table(c$aceita) * 100 / sum(table(c$aceita))
freqd <- table(d$aceita) * 100 / sum(table(d$aceita))

tabela <- rbind("Amostra n = 10" = freqa, "Amostra n = 30" = freqb,
                "Amostra n = 75" = freqc, "Amostra n = 100" = freqd)
tabela_1 <- data.frame(tabela)

knitr::kable(tabela_1, booktabs = T, align = "c",
             caption = "Tabela de Erro Tipo I (em %)",
             col.names = c("Rejeita Hipótese Nula", "Não Rejeita Hipótese Nula"),
             format = "latex", escape = F) %>%
kable_styling(position = "center", latex_options = c("hold_position"))
```

\begin{table}[!h]

\caption{Tabela de Erro Tipo I (em %)}

	Rejeita Hipótese Nula	Não Rejeita Hipótese Nula
Amostra n = 10	4.07	95.93
Amostra n = 30	4.93	95.07
Amostra n = 75	3.79	96.21
Amostra n = 100	4.76	95.24

\end{table}

Agora, partindo para análise do poder, notamos outro resultado esperado na tabela seguinte, a tendência apresentada é apresentada pelo gráfico da seguinte forma: conforme cresce a diferença entre o valor testado do paramêtro ($\lambda = 2$) e/ou cresce o tamanho da amostra maior é o poder do teste, ou seja, torna-se cada vez mais sensível aos desvios do valor de $\widehat{\lambda}_{EMV}$.

```
tabela_2 <- rbind("Amostra n = 10" = a$poder, "Amostra n = 30" = b$poder,
                  "Amostra n = 75" = c$poder, "Amostra n = 100" = d$poder)
p <- seq(2.2, 4, length.out = 5)
df2 <- data.frame(round(tabela_2, 2))

knitr::kable(df2, booktabs = T, align = "c",
             caption = "Tabela do Poder do Teste $\\lambda \\in[2,2;4]$",
             col.names = c(paste("$\\lambda =$", p[1]),
                           paste("$\\lambda =$", p[2]),
                           paste("$\\lambda =$", p[3]),
                           paste("$\\lambda =$", p[4]),
                           paste("$\\lambda =$", p[5])),
             format = "latex", escape = F) %>%
kable_styling(position = "center", latex_options = c("hold_position"))
```

Table 1: Tabela do Poder do Teste $\lambda \in [2,2;4]$

	$\lambda = 2.2$	$\lambda = 2.65$	$\lambda = 3.1$	$\lambda = 3.55$	$\lambda = 4$
Amostra n = 10	0.24	0.99	1	1	1
Amostra n = 30	0.98	1.00	1	1	1
Amostra n = 75	1.00	1.00	1	1	1
Amostra n = 100	1.00	1.00	1	1	1

Conclusão

Após a compreensão e resolução, houve um notório aumento em relação a percepção dos métodos de amostragem e geração de amostras por método Monte Carlo pelos elaboradores deste trabalho, concluindo que o mesmo foi fundamental para a melhora da visualização e entendimento dos resultados esperados de cada processo formulado durante a disciplina. Ao longo do trabalho, percebemos que poderíamos resolver as questões de diversas formas, como no caso da estimação, por exemplo. Pelo conteúdo da disciplina ser bem amplo, podemos associar a diversas outras disciplinas.

Apoio

Para elaborar o trabalho, o número 2021 foi selecionado como seed e, para os exercícios 2 e 3, 10000 foi a quantidade de amostras geradas para as estimações.