Atividade 01 - Introdução ao Aprendizado Estatístico *

Marcel Dantas de Quintela

Atividade apresentada como parte das avaliações da disciplina de Introdução ao Aprendizado Estatístico, ministrada pela Profa. Mariane Barros Alves para o curso de Especialização em Ciência e Dados do Instituto de Matemática da Universidade Federal do Rio de Janeiro

---

Atividade 01

Assuma que a variável aleatória Y denote custos de cancelamento de contratos (em milhares de Reais) registrados por uma empresa. O arquivo **custos.txt** contém 1000 registros de custos de cancelamento, selecionados aleatoriamente da base de dados da empresa. Na primeira coluna, estão os custos na escala original e, na segunda coluna do arquivo, tem-se o logaritmo natural de cada custo, que será usado como veremos adiante.

Obs: Você pode ler o arquivo de dados e conferir se a leitura foi correta usando as linhas decomandoR:

```
dados<-matrix(scan(file="custos.txt"),ncol=2,byrow=T) #Leitura do arquivo
head(dados) #Visualização das primeiras linhas do arquivo lido
```

```
##          [,1]     [,2]
## [1,] 6.846861 1.923790
## [2,] 6.947157 1.938333
## [3,] 5.788298 1.755838
## [4,] 6.927557 1.935507
## [5,] 9.035326 2.201142
## [6,] 8.192294 2.103194
```

```
dim(dados) #Conferindo a dimensão dos do arquivo
```

```
## [1] 1000    2
```

ETAPA 1 (20%): Especificação de um modelo observacional

a. Construa um histograma dos custos de cancelamento na sua escala original, ou seja, os dados contidos na primeira coluna do arquivo.

b. Admita que se deseje modelar os custos de cancelamento usando um modelo Lognormal(μ, σ^2). Justifique brevemente a escolha desse modelo, com base na visualização do histograma do item (a). Caso não conheça o modelo Lognormal, faça uma pesquisa sobre o aspecto de sua função densidade de probabilidade. Enuncie pelo menos um outro modelo probabilístico que você julga que poderia ser usado para descrição desses dados, justificando.

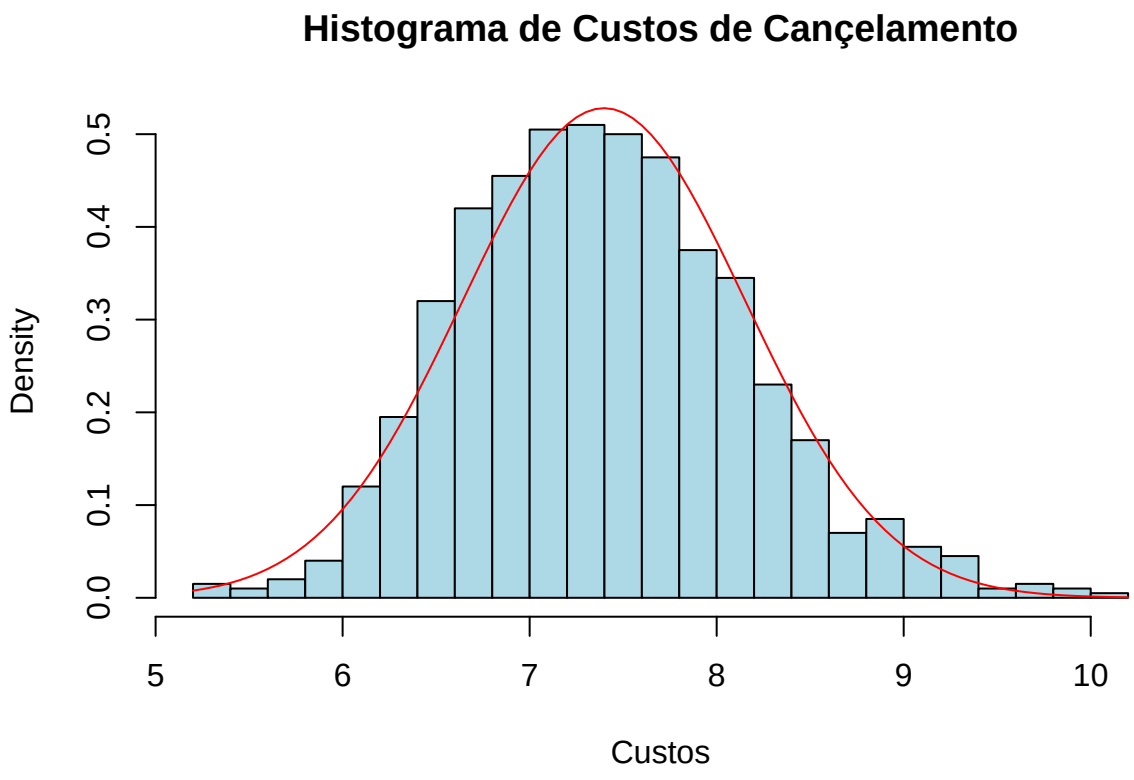
*Replication files are available on the author's Github account (http://github.com/svmiller). **Current version**: fevereiro 24, 2021; **Corresponding author**: svmille@clemson.edu.

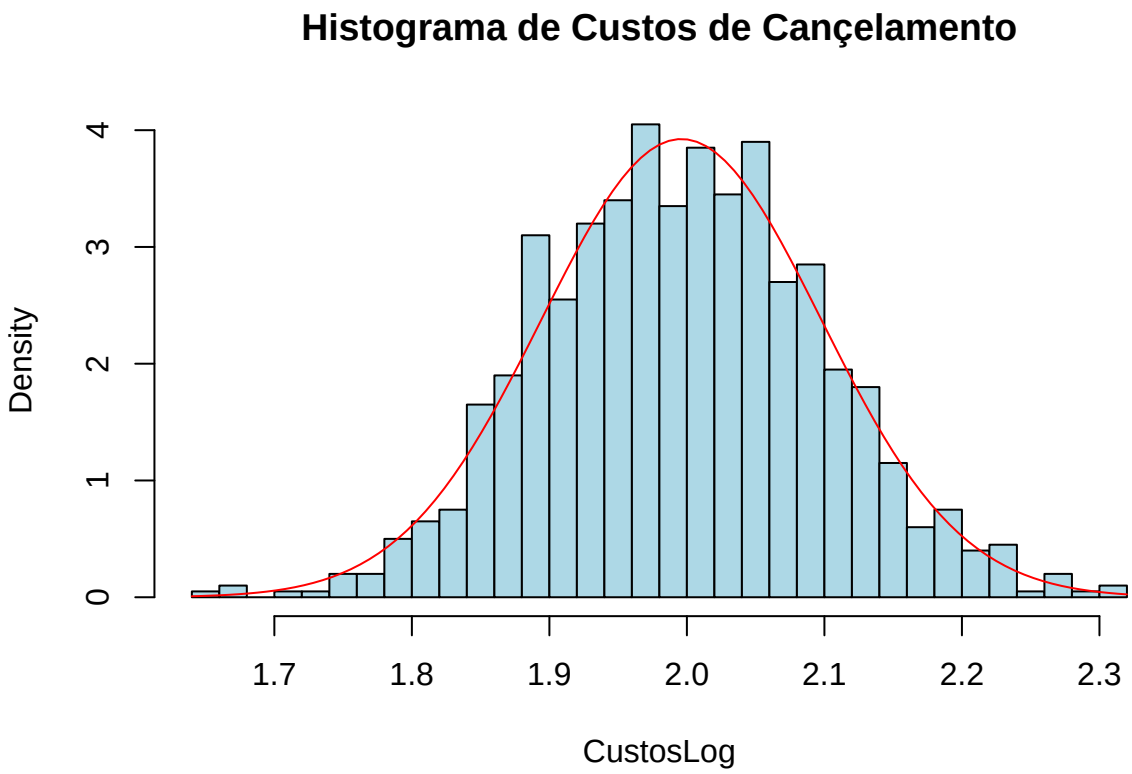
A partir de agora, nos fixaremos no estudo do modelo Lognormal(μ, σ^2) para os custos de cancelamento.

Obs: Dizer que Y (custos) tem distribuição Lognormal(μ, σ^2) é equivalente a dizer que $Z = ln(Y)$ tem distribuição Normal(μ, σ^2). Passamos a estudar a informação contida nos dados com respeito a esses dois parâmetros, μ e σ^2, que são idênticos no modelo Lognormal ou Normal.

```
dados<-as.data.frame(dados) #transformar em dataframe
names(dados)<-c("Custos","CustosLog")

for(i in 1:2){
  hist(dados[,i],
       breaks = 30,
       main="Histograma de Custos de Cançelamento",
       xlab=names(dados)[i],
       right=FALSE,
       prob=TRUE,
       col="light blue")
  curve(dnorm(x,mean(dados[,i]),sd(dados[,i])),add=T, col="red")
}
```





A distribuição Lognormal é indicada para modelar variáveis com distribuições assimétricas à direita. Distribuição assimetricas positiva é caracterizada pela ocorrência de uma grande quantidade de valores baixos e uma pequena quantidade de valores altos a muito altos. Esses valores podem ser interpretados como anômalos e podem ser excluídos, equivocadamente, podendo levar a uma subestimação na avaliação desta distibuição. Assim a distribuição lognirmal pode ser usada na modelagem de variáveis não negativas com assimetria positiva.

Outro modelo que poderia ser usado para descrever estes dados seria o modelo Gama. Dado seu domínio em $\Re^+$ pode se usada para modelar valores de dados positivos que são assimétricos à direita e maiores que 0. O modelo Gama também se mostra bem versátil, uma vez que variando seus parâmetros de forma e de escala , pode-se obter varias formas de densidade da Gama.

ETAPA 2 (40 %): Obtenção numérica de estimas de máxima verossimilhança e seu uso para alimentar o modelo probabilístico:

c. Obtenha estimativas de máxima verossimilhança para (μ, σ^2).

Obs1: Utilize, a título de exemplo, o código R apresentado em aula para obtenção de estimativas de máxima verossimilhança para o caso Gama e adapte-o para o problema que estamos tratando. Considere verossimilhança Normal, alimentada pelos dados transformados Z=ln(Y) (segunda coluna da base de dados).

Obs2: No R, as distribuições Normal e Lognormal são parametrizadas por μ e não por σ^2, então a estimativa que você vai obter é para σ (desvio-padrão da Normal). Esteja atento para esse detalhe.

Obs3: Você precisará informar valores iniciais para a rotina de otimização da verossimilhança. Inicialize o algoritmo de otimização com par de valores (1,1). Repita o procedimento algumas vezes, fazendo uma análise de sensibilidade dos seus resultados ao valor inicial do algoritmo de otimização.

```
rm(x)
n<- length(dados[,2])
x<-data.frame(CustosLog=c(mean(dados[,2]),
                          ((n-1)*var(dados[,2])/n)^.5,
                          (n-1)*var(dados[,2])/n),
              row.names = c("mu","sd","var"))

#tamanho amostral
a <- NULL
b <- NULL

# -logverossimilhança
theta<-c(a,b)
neglogvero<-function(theta){-sum(log(dnorm(dados[,2],theta[1],theta[2])))}

#resultado:
o<-matrix(c(1, 0.5, 1  , 5 ,4 , 2,
            1, 1  , 2 , 1.1 ,2 , 10),6,2)
for (i in 1:length(o[,1])) {
  saida<-nlm(neglogvero,p=o[i,])
#estimativas de maxima verossimilhanca
  x<-cbind(x,Vero=with(saida,c(estimate[1],estimate[2], estimate[2]^2)))
  names(x)[dim(x)[2]]<-paste0("Vero [",o[i,1],",",o[i,2],"]")
}
as.data.frame(t(x))
```

```
##                     mu         sd          var
## CustosLog     1.995955  0.1016070   0.01032399
## Vero [1,1]    1.995955  0.1016071   0.01032400
## Vero [0.5,1]  1.995954  0.1016065   0.01032388
## Vero [1,2]    1.995955  0.1016071   0.01032400
## Vero [5,1.1] -2.920043 16.4018371 269.02025957
## Vero [4,2]    1.995954  0.1016065   0.01032388
## Vero [2,10]   1.995954  0.1016065   0.01032388
```

d. Utilizando o modelo observacional com valores dos parâmetros substituídos por suas estimativas de máxima verossimilhança, determine a probabilidade de que um custo de cancelamento de contrato futuro, nessa empresa, seja superior a 9 (mil Reais).

```
#P(X>9)
pnorm(log(9),mean=x[1,2],sd=x[2,2],lower.tail = FALSE)
```

```
## [1] 0.02380341
```

ETAPA 3 (40%): Estudo visual do comportamento da função de verossimilhança:

Nesta etapa, desejamos visualizar o comportamento do gráfico da função de verossimilhança perfilada para μ . Note que se trabalharmos com os dados transformados $Z = ln(Y)$, teremos verossimilhança obtida a partir da densidade normal, cuja forma analítica é bem conhecida. Expresse analiticamente a função de verossimilhança, notando que esta pode ser escrita em função das estatísticas $t1 = \sum_{i=1}^{n} z_i$ e $t2 = \sum_{i=1}^{n} z_i^2$. Então, a exemplo do que foi feito em aula, escreva a expressão da função de verossimilhança em função de $t1$ e $t2$. Note que essas duas estatísticas contêm toda a informação necessária, vinda dos dados, para avaliarmos a função de verossimilhança (são chamadas estatísticas suficientes – para estimar os parâmetros de interesse, não precisamos guardar toda a amostra observada – é suficiente guardar os valores dessas estatísticas). Agora, tome os dados transformados Z, disponíveis na segunda coluna do arquivo custos.txt e, com esses dados, no R:

e. Faça uma grade para o parâmetro μ, com valores variando de 0 a 5.
f. Faça o gráfico da função de verossimilhança normal (alimentada pelos dados transformados, Z), contra a grade de valores para μ, fixando o valor de σ^2 na estimativa de máxima verossimilhança obtida na etapa 2.
g. Comente o gráficos e relacione seus comentários à estimativa de μ obtida na etapa 2.

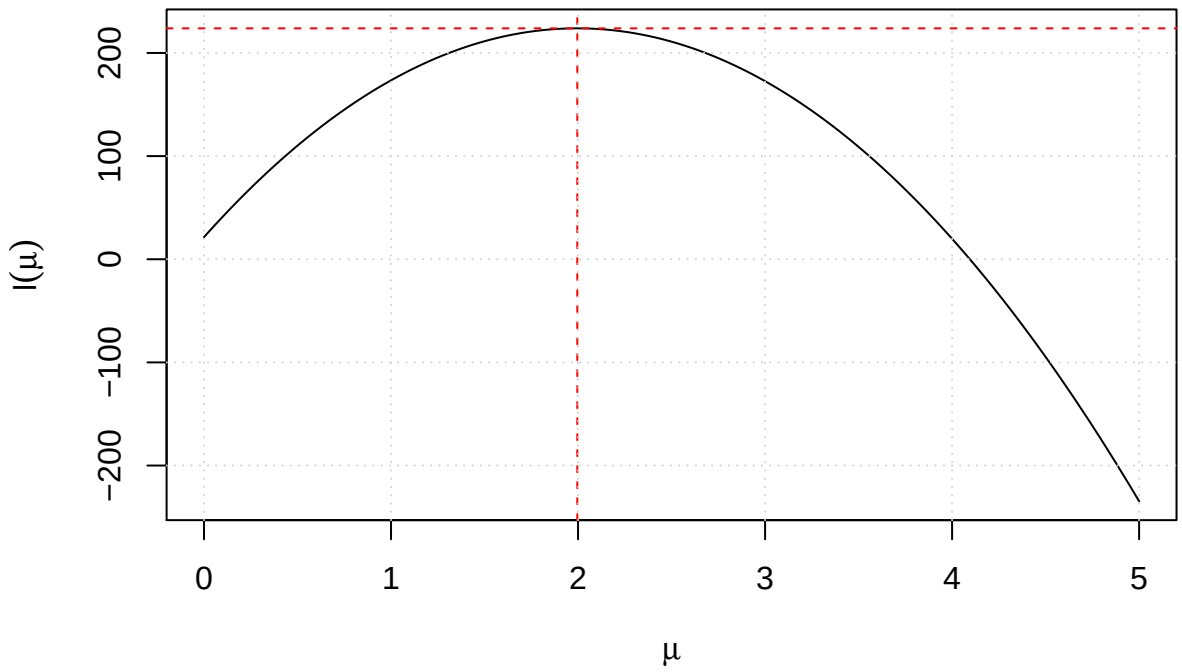
Obs: Fizemos, em aula, gráficos exploratórios do comportamento da função de verossimilhança para os modelos Bernoulli e Poisson. Os códigos R usados naqueles exemplos podem ser facilmente adaptados para a execução da etapa 3.

```
y<-dados[,2]
n <- length(y)
a <- NULL
b <- NULL

t1<-sum(y)
t2<-sum(y^2)

logvero<-function(a,b){-n/2*log(2*pi*b)-1/2*b*(t2-2*a*t1+n*a^2)}
#Definindo o grid para \mu
grade<-seq(0,5,0.01)

#gráfico de \mu x L(\mu) fixando o \sigma
plot(grade,logvero(grade,x[2,2]),
     main="",
     xlab = expression(mu),
     ylab = expression(l(mu)),
     type="l")
abline(v=x[1,2],lty=2,col="red")
abline(h=max(logvero(grade,x[2,2])),lty=2,col="red")
grid()
```


l(μ)
200
100
0
−100
−200
0
1
2
3
4
5
μ