Segunda avaliação

Prof. Thiago Gentil Ramires - Probabilidade e Estatística

Nome: Filipe Augusto Parreira Almeida

RA: 2320622

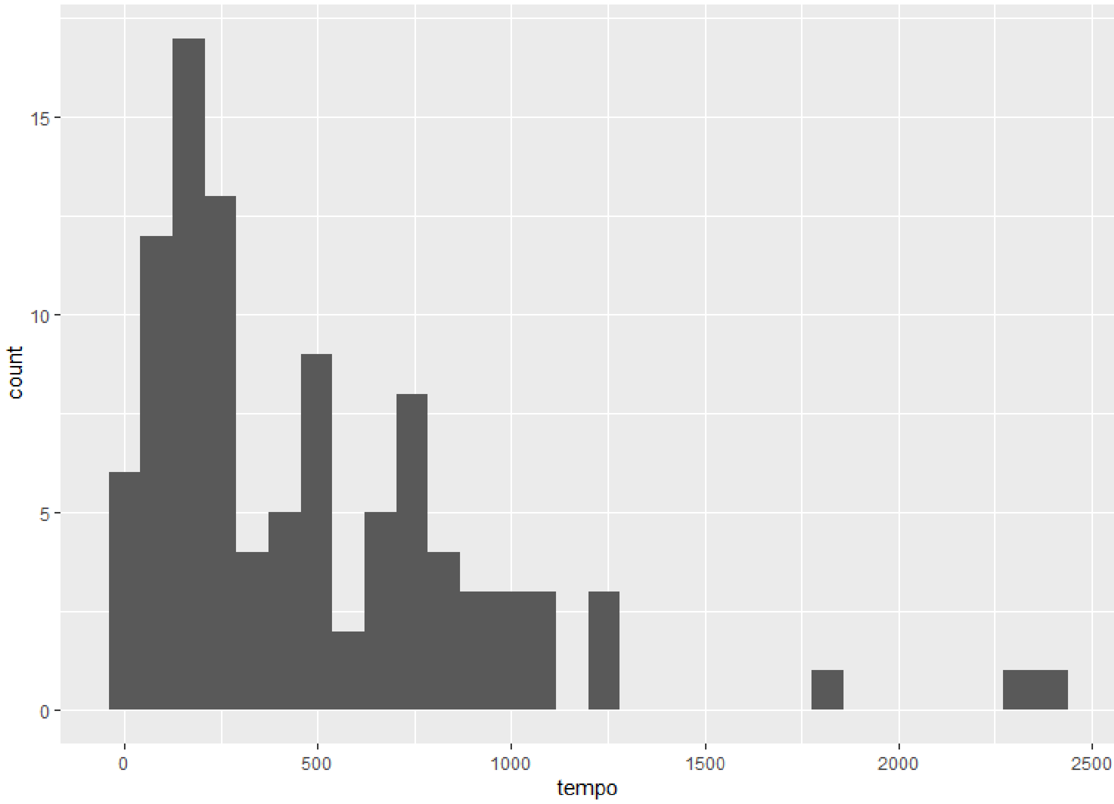
Disciplina matriculado(a): Probabilidade e Estatística – Eng. Comp.

Para todas as questões abaixo, **interprete os resultados e apresente os códigos e gráficos, quando necessário**. As bases de dados estão em anexo do Google Classroom, já salvas em CSV, com o separador decimal em Inglês, ou seja, as decimais estão separadas por ponto. **Cada questão vale 0,5 pontos.**

Questão 1) O tempo de duração (em horas) de 100 vigas metálicas, após teste de força, está apresentado no arquivo ex1.csv. Determine:

a) Qual o melhor modelo de probabilidade que representa tais tempos?

Resposta: Modelo Exponencial



Códigos:

dadosEx1 = read.csv("ex1.csv")

head(dadosEx1)

ggplot(dadosEx1, aes(tempo)) + geom_histogram()

b) Determine a probabilidade de uma viga durar menos de 300 horas.

Resposta: Probabilidade de aproximadamente 46,4% ou 0.464611.

Códigos:

```
mediaEx1 = mean(dadosEx1$tempo)

p1 = pexp(300, 1/mediaEx1)
```

c) Determine a probabilidade de uma viga durar entre 200 e 400 horas.

Resposta: Probabilidade de aproximandamente 22,4% ou 0.2246091

Códigos:

```
mediaEx1 = mean(dadosEx1$tempo)

p2 = pexp(400, 1/mediaEx1) - pexp(200, 1/mediaEx1)
```

d) Serão descartadas 70% das vigas com menor tempo de duração. Determine o tempo ideal de descarte, ou seja, o tempo que limita o descarte das vigas.

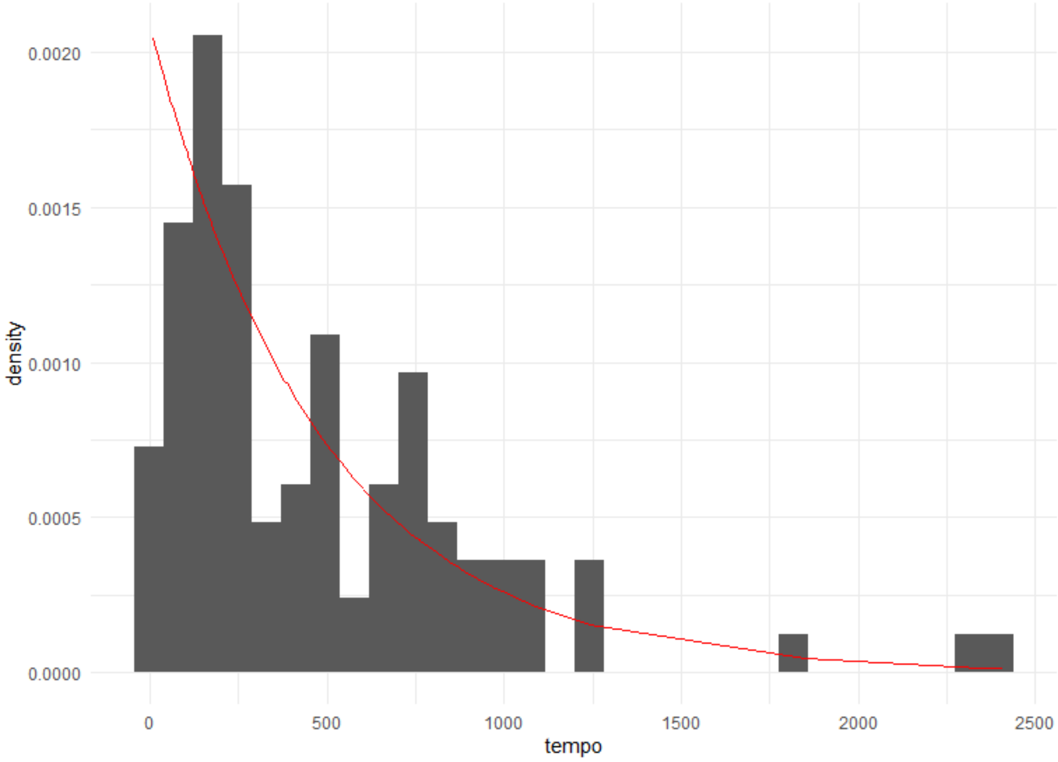
Resposta: O tempo que limita o descarte das vigas é de 578.1273

Códigos:

```
mediaEx1 = mean(dadosEx1$tempo)

p3 = qexp(0.7, 1/mediaEx1)
```

e) Apresente graficamente o histograma dos dados juntamente com o modelo escolhido (ajustado) na resposta A).

Resposta:



Códigos:

```
mediaEx1 = mean(dadosEx1$tempo)
tempo = dadosEx1$tempo
px = dexp(tempo, 1/mediaEx1);
dados = data.frame(tempo, px)
ggplot(dados, aes(tempo)) + geom_histogram(aes(y=..density..)) +
geom_line(aes(tempo, px), col='red') + theme_minimal()
```

Questão 2) Uma empresa está interessada em estudar o comportamento dos produtos eletrônicos produzidos por ela. Em um teste com 20 desses produtos produzidos, 6 apresentaram defeitos. Em uma semana foram produzidos 100 novos produtos. Determine:

a) A probabilidade de mais de 75 ou mais não apresentarem defeito?

Resposta: Levando em consideração o primeiro teste, pode-se concluir que 70% dos produtos não apresentam defeito, logo, 30% deles podem apresentar defeito, sendo assim, em 100 novos produtos a chance de 75 ou mais não apresentarem defeito é dado pela chance de nenhum apresentar defeito menos a chance de 25 ou menos apresentarem defeito, 83,6% ou 0.8368699.

Códigos:

```
p1 = 1 - pbinom(25, 100, 0.3)
```

b) A probabilidade de exatamente 70 não apresentarem defeito?

Resposta: A probabilidade de exatamente 70 não apresentarem defeito é de 8% ou 0.08678386.

Códigos:

```
p2 = dbinom(70, 100, 0.7)
```

c) A probabilidade de menos de 20 produtos apresentarem defeito?

Resposta: A probabilidade de menos de 20 produtos apresentarem defeito é de aproximadamente 1% ou 0.01646285

Códigos:

```
p3 = sum(dbinom(1:20, 100, 0.3))
```

d) Se no próximo mês forem produzidos 4000 produtos, quantos irão falhar em média?

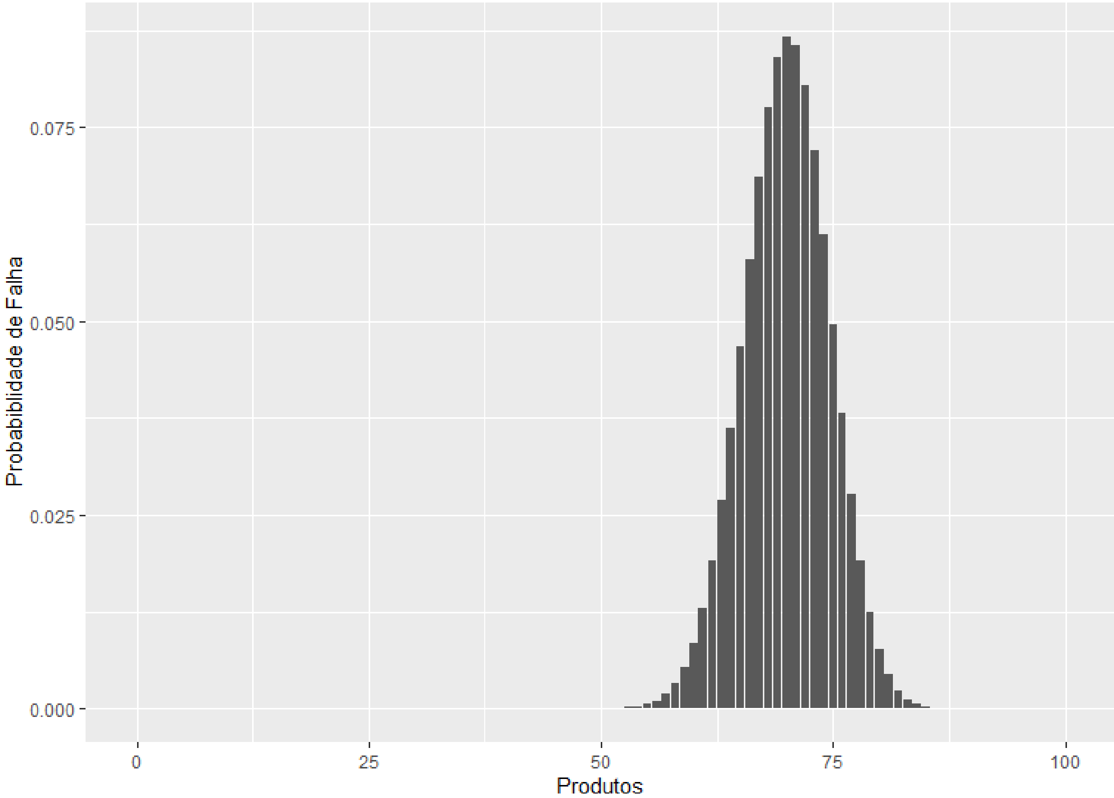
Resposta: Em média, na produção de 4000 produtos , 1200 iram falhar.

Códigos:

```
p4 = 4000 * 0.3
```

e) Apresente graficamente todas as probabilidades de não apresentar falha, considerando os n=100 produtos.

Resposta:



Códigos:

```
x = 0:100
px = dbinom(x, 100, 0.7)
dados = data.frame(x, px)
ggplot(dados, aes(x, px)) + geom_col() + labs(x = 'Produtos', y = 'Probabiblidade de Falha')
```

Questão 3) Um algoritmo de detecção de anomalias capta, em média, 20 erros por hora. Determine:

a) A probabilidade de detectar 15 erros em uma hora?

Resposta: A probabilidade de encontrar 15 erros em 1 hora é de aproximadamente 5% ou 0.05164885

Códigos:

```
p1 = dpois(15, 20)
```

b) A probabilidade de detectar entre 20 e 30 erros em uma hora?

Resposta: A probabilidade de detectar entre 20 e 30 erros em uma hora é de aproximadamente 42% ou 0.4274327

Códigos:

```
p2 = ppois(30, 20) - ppois(20, 20)
```

c) A probabilidade de detectar mais de 449 erros em um dia?

Resposta: A probabilidade de detectar mais de 449 erros em 24 horas é de aproximadamente 91% ou 0.919181

Códigos:

```
p3 = 1 - ppois(449, 20*24)
```

d) Um novo algoritmo foi testado, sendo que a média de detecção em uma hora foi de 30 erros. Esse algoritmo será adquirido pela empresa caso detecte na próxima hora mais de 34 erros. Determine a probabilidade da compra ser realizada.

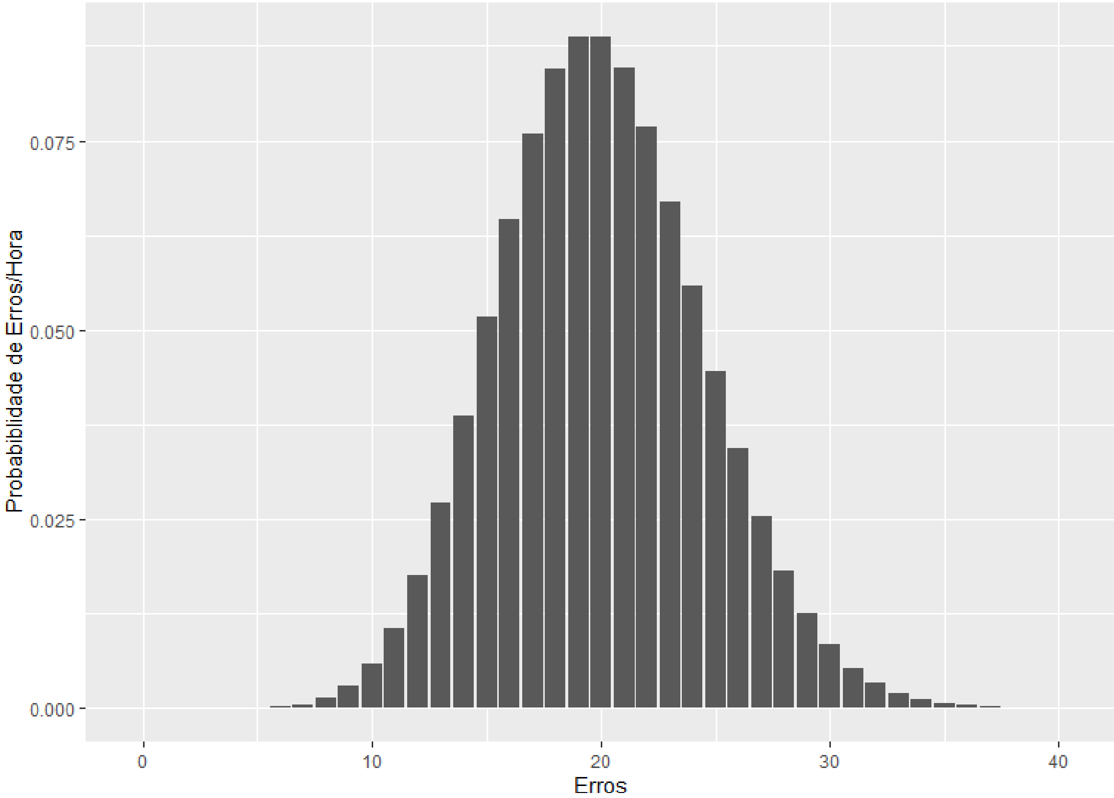
Resposta: A probabilidade da compra ser realizada é de aproximadamente 20% ou 0.2026917

Códigos:

```
p4 = 1 - ppois(34, 30)
```

e) Apresente o gráfico das probabilidades de detecção de erros do algoritmo, considerando o algoritmo com média de 20 erros por hora.

Resposta:



Códigos:

x = 0:40;

px = dpois(x, 20)

dados = data.frame(x, px)

ggplot(dados, aes(x, px)) + geom_col() + labs(x = 'Erros', y = 'Probabiblidade de Erros/Hora')

Questão 4) O peso de um produto, em Kg, foi determinado em uma amostra de tamanho 300 (Ver anexo ex4.csv). Determine:

a) Qual o melhor modelo de probabilidade que representa tais pesos?

Resposta: Modelo normal

Códigos:

```
dadosEx4 = read.csv("ex4.csv")

head(dadosEx4)

ggplot(dadosEx4, aes(dadosEx4$peso)) + geom_histogram()
```

b) Determine a probabilidade de um produto não pesar entre 45 e 55kg.

Resposta: A probabilidade de um produto não pesar entre 45 e 55 Kg é de aproximadamente 31% ou 0.3137118

Códigos:

```
media = mean(dadosEx4$peso)

desvio = sd(dadosEx4$peso)

p1 = 1 - (pnorm(55, media, desvio) - pnorm(45, media, desvio))
```

c) Determine a probabilidade de um peso ter exatamente 50Kg.

Resposta: A probabilidade de algo ser exatamente igual a um valor em um modelo continuo é sempre 0, logo, a probabilidade de um peso ser exatamente 50Kg é 0.

d) Os produtos serão classificados entre: leves (30% mais baixos), médios (pesos entre 30 a 70%) e pesados (os 30% maiores). Determine os limites dos pesos para realizar essa classificação.

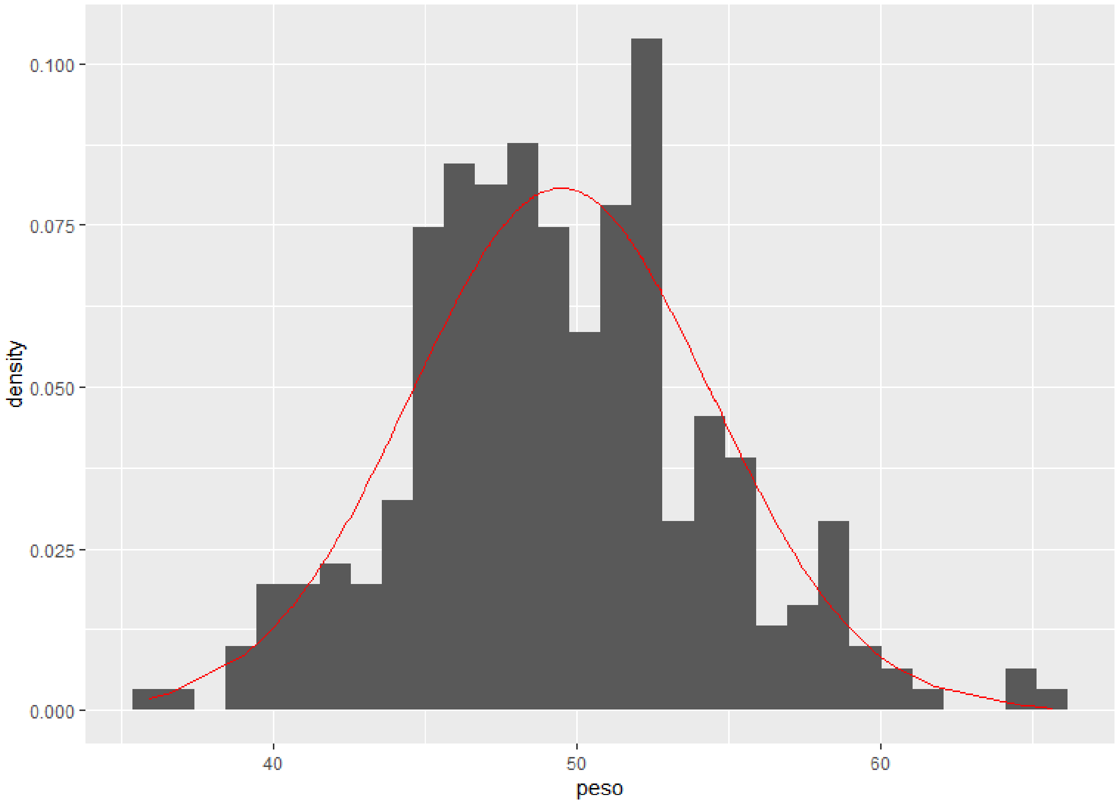
Resposta: O limites de peso são: Leve: 46.90295; Médio: 48.24108; Pesado: 52.08065

Códigos:

```
media = mean(dadosEx4$peso)

desvio = sd(dadosEx4$peso)

leve = qnorm(0.3, media, desvio)

medio = qnorm(0.4, media, desvio)

pesado = qnorm(0.7, media, desvio)
```

e) Apresente graficamente o histograma dos dados juntamente com o modelo escolhido (ajustado) na resposta A).

Resposta:



Códigos:

```
media = mean(dadosEx4$peso)

desvio = sd(dadosEx4$peso)

peso = dadosEx4$peso


px = dnorm(peso, media, desvio)

dados = data.frame(peso, px)


ggplot(dados, aes(peso)) + geom_histogram(aes(y = ..density..)) + 
geom_line(aes(peso, px), col = 'red')
```