Modelos probabilísticos para variáveis aleatórias contínuas

Resolução dos exemplos e exercícios utilizando o Python

Distribuição Exponencial

Distribuição Exponencial para o cálculo de ***P(0≤X≤x) = P(X≤x)***

expon.cdf(x,scale=media)

Distribuição Exponencial para o cálculo de ***P(X>x)***

expon.sf(x,scale=media)

onde:

alfa= parâmetro da distribuição exponencial que representa o inverso da média.

media = 1/alfa

Importante:

Para usar as funções de cálculo de probabilidade para a distribuição exponencial no Python é necessário primeiramente que você importe a função expon*:*

from scipy.stats import expon

Exemplo 1

A vida útil de uma lâmpada é modelada através da distribuição exponencial com parâmetro 1/8000.

a) Calcule o tempo médio de duração dessas lâmpadas.
X: vida útil de uma lâmpada
X ~ Exp (1/8000).

$$E(X) = \frac{1}{\alpha} = \frac{1}{1/8000} = 8000 \text{horas}$$

b) Calcule a probabilidade de que uma lâmpada dure pelo menos 4000 horas.
Como a distribuição exponencial não tem um limite superior mas, tem o zero como limite inferior, para realizar o cálculo de que a lâmpada dure pelo menos 4000 horas iremos utilizar o complementar, ou seja,

$$P(X \geq 4000) = 1 - P(0 < X < 4000) = 1 - \left(e^{-\frac{1}{8000}0} - e^{-\frac{1}{8000}4000} \right) = 0{,}6065$$

c) Sabe-se que o fabricante garante a reposição de uma lâmpada caso ela dure menos de 50 horas. Determine a probabilidade de haver troca por defeito na fabricação.

$$P(X < 50) = P(0 < X < 50) = e^{-\frac{1}{8000}0} - e^{-\frac{1}{8000}50} = 0{,}0062$$

d) Uma lâmpada é colocada em teste. Calcule a probabilidade de que ela dure pelo menos 10000 horas, sabendo-se que ela já está em funcionamento a pelo menos 6000 horas.
Para resolver essa probabilidade condicional podemos utilizar a propriedade de falta de memória da distribuição exponencial:

$$P(X \geq 10000 | X \geq 6000) = P(X \geq 4000) = 0{,}6065$$

Utilizando o Python:

```
from scipy.stats import expon

# Parâmetro da distribuição exponencial
lambd = 1/8000

# Tempo médio de duração (esperança da distribuição exponencial)
tempo_medio = 1 / lambd
print("a. Tempo médio de duração das lâmpadas:", tempo_medio, "horas")

# Probabilidade de que uma lâmpada dure pelo menos 4000 horas
probabilidade_4000_horas = expon.sf(4000, scale=1/lambd)
print("b. Probabilidade de uma lâmpada durar pelo menos 4000 horas:", probabilidade_4000_horas)

# Probabilidade de haver troca por defeito na fabricação
probabilidade_troca_defeito = expon.cdf(50, scale=1/lambd)
print("c. Probabilidade de haver troca por defeito na fabricação:", probabilidade_troca_defeito)

# Probabilidade de uma lâmpada durar pelo menos 10000 horas dado que ela já durou 6000 horas
probabilidade_10000_dado_6000 = expon.sf(10000 - 6000, scale=1/lambd)
print("d. Probabilidade de uma lâmpada durar pelo menos 10000 horas, dado que ela já está em
funcionamento há pelo menos 6000 horas:", probabilidade_10000_dado_6000)
```

a. Tempo médio de duração das lâmpadas: 8000.0 horas
b. Probabilidade de uma lâmpada durar pelo menos 4000 horas: 0.6065306597126334
c. Probabilidade de haver troca por defeito na fabricação: 0.00623050937660527
d. Probabilidade de uma lâmpada durar pelo menos 10000 horas, dado que ela já está em funcionamento há pelo menos 6000 horas: 0.6065306597126334

Exercício 1

A vida de certa marca de lâmpada tem uma distribuição aproximadamente exponencial com média de 1000 horas.

a) Determinar a porcentagem das lâmpadas que queimarão antes de 1000 horas.
b) Após quantas horas terão queimado 50% das lâmpadas?

Respostas: a) 0,6312 b) 693,1472 horas

Iniciando o raciocínio:
X: tempo de duração de uma lâmpada (em horas)
E(X)= 1000 horas → Duração média
α=1/1000 horas → Parâmetro da distribuição exponencial
X pode ser descrita pela distribuição exponencial.

a) Determinar a porcentagem das lâmpadas que queimarão antes de 1000 horas.
   P(X<1000) = ?
b) Após quantas horas terão queimado 50% das lâmpadas?
   P(X< b) = 0,5 → Preciso descobrir o valor de b, que corresponde a quantidade de horas abaixo da qual 50% das lâmpadas estarão queimadas.

Utilizando o Python:

```
from scipy.stats import expon

# Média da distribuição exponencial
media = 1000

# Parâmetro da distribuição exponencial (lambda)
lambd = 1 / media

# a) Porcentagem das lâmpadas que queimarão antes de 1000 horas
porcentagem_antes_1000_horas = expon.cdf(1000, scale=media) * 100
print("Porcentagem das lâmpadas que queimarão antes de 1000 horas:", porcentagem_antes_1000_horas,
"%")

# b) Após quantas horas terão queimado 50% das lâmpadas?
horas_50_percent = expon.ppf(0.5, scale=media)
print("Após", horas_50_percent, "horas terão queimado 50% das lâmpadas.")
```

```
Porcentagem das lâmpadas que queimarão antes de 1000 horas: 63.212055882855765 %
Após 693.1471805599452 horas terão queimado 50% das lâmpadas.
```

Observe que usamos uma função nova '.ppf'. Qual é o objetivo dela? Realizar cálculos inversos! Você fornece a probabilidade acumulada até um determinado valor x e ela te retorna com o valor de x correspondente, ou seja, você informa que a P(X≤ b) = 0,5 e a função de retorna o valor de b.

Exercício 2

Uma fábrica utiliza dois métodos para a produção de lâmpadas. 70% das lâmpadas são produzidas pelo método A e as demais pelo método B. A duração da lâmpada depende do método pelo qual ela foi produzida, sendo que as produzidas pelo método A seguem uma distribuição exponencial com parâmetro 1/80 e as do método B seguem uma exponencial de parâmetro 1/100. Qual a probabilidade de que, se escolhermos uma lâmpada ao acaso, ela dure mais de 100 horas?

Resposta: 0,31

Iniciando o raciocínio:
X: tempo de duração de uma lâmpada (em horas)
X pode ser descrita pela distribuição exponencial.

- Método A:
    - P(A) = 0,70 → Probabilidade de ser produzida pelo método A.
    - E(X)= 80 horas → Duração média
    - α=1/80 horas → Parâmetro da distribuição exponencial

- Método B:
    - P(B) = 1- P(A) = 0,30 → Probabilidade de ser produzida pelo método B.
    - E(X)= 100 horas → Duração média
    - α=1/100 horas → Parâmetro da distribuição exponencial

Pergunta: "...Qual a probabilidade de que, se escolhermos uma lâmpada ao acaso, ela dure mais de 100 horas?" → Para responder a essa pergunta você precisa considerar os dois métodos de produção e a probabilidade de que a lâmpada dure mais de 100 horas em cada método de produção.

Utilizando o Python:

```
from scipy.stats import expon

# Probabilidade de escolher uma lâmpada produzida pelo método A
prob_A = 0.7

# Parâmetros das distribuições exponenciais para os métodos A e B
parametro_A = 1 / 80
parametro_B = 1 / 100

# Calculando a probabilidade de que uma lâmpada dure mais de 100 horas
prob_duracao_mais_100_horas = prob_A * expon.sf(100, scale=1/parametro_A) + (1 - prob_A) *
expon.sf(100, scale=1/parametro_B)
print("Probabilidade de que uma lâmpada dure mais de 100 horas:", prob_duracao_mais_100_horas)
```

Probabilidade de que uma lâmpada dure mais de 100 horas: 0.3109171901535658

Distribuição Normal

*Distribuição Normal para o cálculo de probabilidade **P(X≤x)***

norm.cdf(x, m, s)

*Distribuição Normal para o cálculo de probabilidade **P(X>x)***

norm.sf(x, m, s)

*Cálculo inverso: Informa o valor de x a partir de uma probabilidade **acumulada***

norm.ppf(p, m, s)

onde:

m= média
s= desvio padrão
p = representa a probabilidade acumulada até x

Importante:

Para usar as funções de cálculo de probabilidade para a distribuição normal no Python é necessário primeiramente que você importe a função norm*:*
from scipy.stats import norm

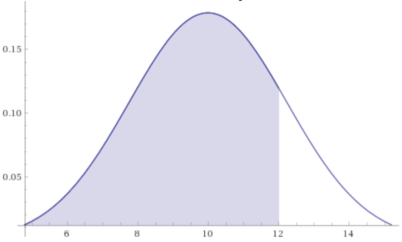
Exemplo 3

Suponha que as medidas da corrente em um pedaço de fio sigam a distribuição normal, com um média de 10 miliamperes e uma variância de 5 miliamperes. Qual a probabilidade:

Como temos uma variável (X: medida da corrente em um pedaço de fio) com distribuição normal com μ=10 e σ^2=5, é necessário padroniza-la para poder consultar as probabilidades disponíveis na tabela da distribuição normal padrão. A padronização de uma variável X ~ N (μ, σ^2) em uma variável Z ~ N (0, 1) é realizada efetuando o seguinte cálculo:

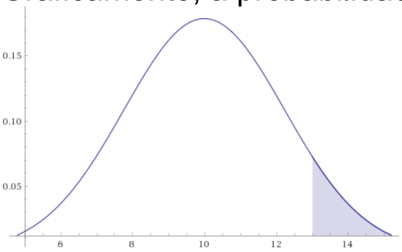
$$z = \frac{x - \mu}{\sigma}$$

a) Da medida da corrente ser de no máximo 12 miliamperes.
Graficamente, a probabilidade desejada pode ser representada da seguinte maneira:



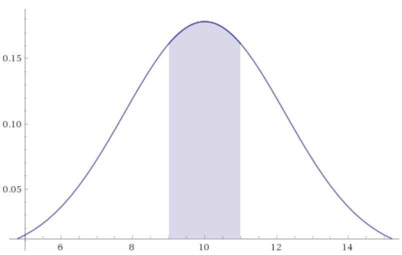
$$P(X \leq 12) = P\left(Z \leq \frac{12 - 10}{\sqrt{5}}\right) = P(Z \leq 0{,}89) = 0{,}5 + 0{,}3133 = 0{,}8133$$

b) Da medida da corrente ser de pelo menos 13 miliamperes.
Graficamente, a probabilidade desejada pode ser representada da seguinte maneira:



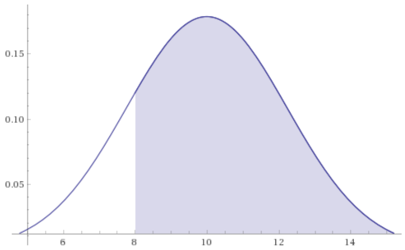
$$P(X \geq 13) = P\left(Z \geq \frac{13 - 10}{\sqrt{5}}\right) = P(Z \geq 1{,}34) = 0{,}5 - 0{,}4099 = 0{,}0901$$

c) Um valor entre 9 e 11 miliamperes.
Graficamente, a probabilidade desejada pode ser representada da seguinte maneira:



$$P(9 < X < 11) = P\left(\frac{9 - 10}{\sqrt{5}} < Z < \frac{11 - 10}{\sqrt{5}}\right) = P(-0{,}45 < Z < +0{,}45)$$
$$= 0{,}1736 + 0{,}1736 = 0{,}3472$$

d) Maior do que 8 miliamperes.
Graficamente, a probabilidade desejada pode ser representada da seguinte maneira:



$$P(X > 8) = P\left(Z > \frac{8-10}{\sqrt{5}}\right) = P(Z > -0{,}89) = 0{,}5 + 0{,}3133 = 0{,}8133$$

Utilizando o Pyhton:

```
from scipy.stats import norm

# Média e variância da distribuição normal
media = 10
variancia = 5

# Desvio padrão é a raiz quadrada da variância
desvio_padrao = variancia ** 0.5

# a) Probabilidade da medida da corrente ser de no máximo 12 miliamperes
prob_a = norm.cdf(12, media, desvio_padrao)
print("a) Probabilidade da medida da corrente ser de no máximo 12 miliamperes:", prob_a)

# b) Probabilidade da medida da corrente ser de pelo menos 13 miliamperes
prob_b = 1 - norm.cdf(13, media, desvio_padrao)
print("b) Probabilidade da medida da corrente ser de pelo menos 13 miliamperes:", prob_b)

# c) Probabilidade de um valor entre 9 e 11 miliamperes
prob_c = norm.cdf(11, media, desvio_padrao) - norm.cdf(9, media, desvio_padrao)
print("c) Probabilidade de um valor entre 9 e 11 miliamperes:", prob_c)

# d) Probabilidade de ser maior do que 8 miliamperes
prob_d = 1 - norm.cdf(8, media, desvio_padrao)
print("d) Probabilidade de ser maior do que 8 miliamperes:", prob_d)


a) Probabilidade da medida da corrente ser de no máximo 12 miliamperes: 0.8144533152386513
b) Probabilidade da medida da corrente ser de pelo menos 13 miliamperes: 0.08985624743949994
c) Probabilidade de um valor entre 9 e 11 miliamperes: 0.34527915398142306
d) Probabilidade de ser maior do que 8 miliamperes: 0.8144533152386513
```

Exercício 3

Considere que a pontuação obtida por diferentes candidatos em um concurso público segue uma distribuição aproximadamente normal, com média igual a 140 pontos e desvio padrão igual a 20 pontos. Suponha que um candidato é escolhido ao acaso. Calcule as probabilidades a seguir:

a) Apresentar uma pontuação entre 140 e 165,6.
b) Apresentar uma pontuação entre 127,4 e 140.
c) Apresentar uma pontuação entre 117,2 e 157.
d) Apresentar uma pontuação inferior a 127.
e) Apresentar uma pontuação superior a 174,2.
f) Apresentar uma pontuação inferior a 167,4.
g) Apresentar uma pontuação entre 155,4 e 168,4.

Respostas: a) 0,3997 b) 0,2357 c) 0,6752 d) 0,2578 e) 0,0436 f) 0,9147 g) 0,1428

Utilizando o Python:

```
from scipy.stats import norm

# Média e desvio padrão da distribuição normal
media = 140
desvio_padrao = 20

# a) Probabilidade de apresentar uma pontuação entre 140 e 165,6
prob_a = norm.cdf(165.6, media, desvio_padrao) - norm.cdf(140, media, desvio_padrao)
print("a) Probabilidade de apresentar uma pontuação entre 140 e 165,6:", prob_a)

# b) Probabilidade de apresentar uma pontuação entre 127,4 e 140
prob_b = norm.cdf(140, media, desvio_padrao) - norm.cdf(127.4, media, desvio_padrao)
print("b) Probabilidade de apresentar uma pontuação entre 127,4 e 140:", prob_b)

# c) Probabilidade de apresentar uma pontuação entre 117,2 e 157
prob_c = norm.cdf(157, media, desvio_padrao) - norm.cdf(117.2, media, desvio_padrao)
print("c) Probabilidade de apresentar uma pontuação entre 117,2 e 157:", prob_c)

# d) Probabilidade de apresentar uma pontuação inferior a 127
prob_d = norm.cdf(127, media, desvio_padrao)
print("d) Probabilidade de apresentar uma pontuação inferior a 127:", prob_d)

# e) Probabilidade de apresentar uma pontuação superior a 174,2
prob_e = 1 - norm.cdf(174.2, media, desvio_padrao)
print("e) Probabilidade de apresentar uma pontuação superior a 174,2:", prob_e)

# f) Probabilidade de apresentar uma pontuação inferior a 167,4
prob_f = norm.cdf(167.4, media, desvio_padrao)
print("f) Probabilidade de apresentar uma pontuação inferior a 167,4:", prob_f)

# g) Probabilidade de apresentar uma pontuação entre 155,4 e 168,4
prob_g = norm.cdf(168.4, media, desvio_padrao) - norm.cdf(155.4, media, desvio_padrao)
print("g) Probabilidade de apresentar uma pontuação entre 155,4 e 168,4:", prob_g)
```

```
a) Probabilidade de apresentar uma pontuação entre 140 e 165,6: 0.39972743204555794
b) Probabilidade de apresentar uma pontuação entre 127,4 e 140: 0.23565270788432235
c) Probabilidade de apresentar uma pontuação entre 117,2 e 157: 0.6751943063145094
d) Probabilidade de apresentar uma pontuação inferior a 127: 0.2578461108058647
e) Probabilidade de apresentar uma pontuação superior a 174,2: 0.043632936524031996
```

f) Probabilidade de apresentar uma pontuação inferior a 167,4: 0.914656549178033
g) Probabilidade de apresentar uma pontuação entre 155,4 e 168,4: 0.14284610581610324

Exercício 4

As vendas diárias de um mercado de bairro seguem, aproximadamente, uma distribuição normal, com média igual a R$5.000,00 e desvio padrão igual a R$2.000,00. Calcule a probabilidade de que, em um determinado dia, as vendas:

a) Sejam superiores a R$3.500,00?
b) Sejam inferiores a R$3.000,00?
c) Estejam entre R$3.800,00 e R$5.300,00?
d) Estejam entre R$2.100,00 e 7.800,00?

Respostas: a) 0,7734 b) 0,1587 c) 0,2854 d) 0,8457

Utilizando o Python:

```
from scipy.stats import norm

# Média e desvio padrão das vendas diárias
media_vendas = 5000
desvio_padrao_vendas = 2000

# a) Probabilidade de que as vendas sejam superiores a R$3.500,00
prob_a = 1 - norm.cdf(3500, media_vendas, desvio_padrao_vendas)
print("a) Probabilidade de que as vendas sejam superiores a R$3.500,00:", prob_a)

# b) Probabilidade de que as vendas sejam inferiores a R$3.000,00
prob_b = norm.cdf(3000, media_vendas, desvio_padrao_vendas)
print("b) Probabilidade de que as vendas sejam inferiores a R$3.000,00:", prob_b)

# c) Probabilidade de que as vendas estejam entre R$3.800,00 e R$5.300,00
prob_c = norm.cdf(5300, media_vendas, desvio_padrao_vendas) - norm.cdf(3800, media_vendas,
desvio_padrao_vendas)
print("c) Probabilidade de que as vendas estejam entre R$3.800,00 e R$5.300,00:", prob_c)

# d) Probabilidade de que as vendas estejam entre R$2.100,00 e R$7.800,00
prob_d = norm.cdf(7800, media_vendas, desvio_padrao_vendas) - norm.cdf(2100, media_vendas,
desvio_padrao_vendas)
print("d) Probabilidade de que as vendas estejam entre R$2.100,00 e R$7.800,00:", prob_d)
```

```
a) Probabilidade de que as vendas sejam superiores a R$3.500,00: 0.7733726476231317
b) Probabilidade de que as vendas sejam inferiores a R$3.000,00: 0.15865525393145707
c) Probabilidade de que as vendas estejam entre R$3.800,00 e R$5.300,00: 0.2853645746201689
d) Probabilidade de que as vendas estejam entre R$2.100,00 e R$7.800,00: 0.8457140811565805
```