Aprendizado de Máquina - Lista 01

Ricardo Marra - 19/0137576

February 6, 2022

[ ]:
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns

from sklearn.datasets import load_iris
```

Questão 01

X is a continuous-valued random variable with uniform density in $[-1, +1]$.

Para a função de densidade de probabilidade, temos que:

$$\int_{-\infty}^{\infty} f(x)\, dx = 1$$

No caso, como a densidade é uniforme: $f(x) = C$

E como a função está definida apenas no intervalo $[-1, 1]$:

$$\int_{-1}^{1} C\, dx = 1$$

$$C[1 - (-1)] = 1$$

$$2C = 1$$

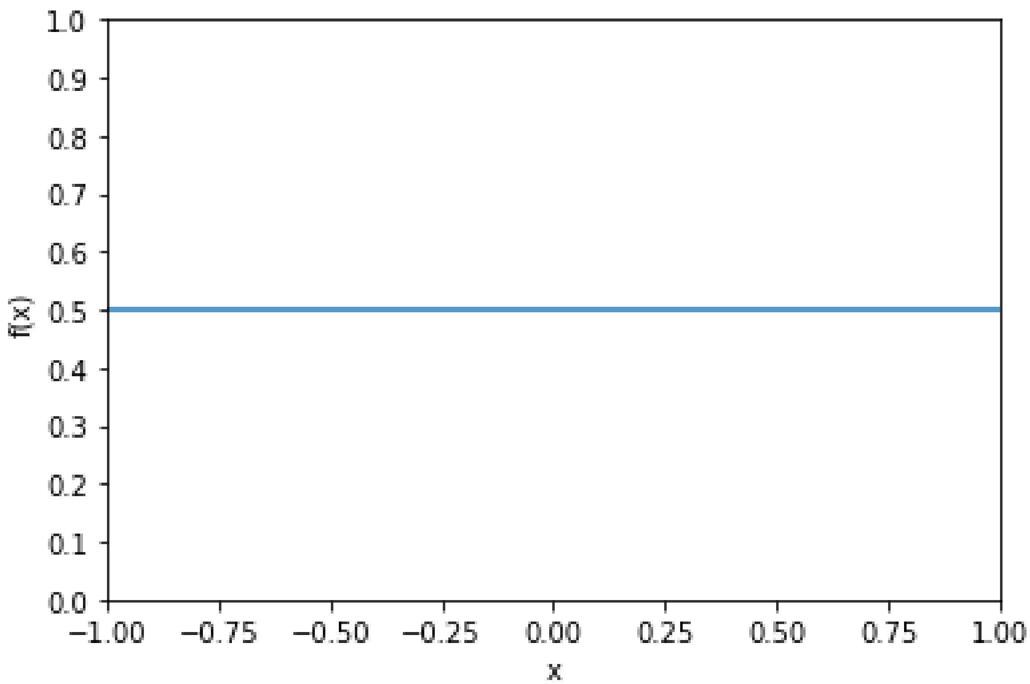
$$C = \frac{1}{2}$$

A área da função de densiddade de probabilidade é dada pela integral da função em seu domínio definido. Esta integral por definição é sempre igual a 1.

A curva é dada pela constante $\frac{1}{2}$ no domínio definido.

[ ]:
```
plt.plot([-1, 1], [0.5, 0.5])
plt.xlim([-1, 1])
plt.ylim([0, 1])
```

```
plt.yticks(np.arange(0, 1.1, step = 0.1))
plt.xlabel('x')
plt.ylabel('f(x)')
plt.show()
```



A função de distribuição cumulativa é dada por:

$$F(x) = \int_{-\infty}^{x} f(t)\, dt$$

Neste caso:

$$f(t) = \frac{1}{2},\ t \in [-1, 1]$$

Assim, temos que $t = 0$ no intervalo $[-\infty, -1]$.

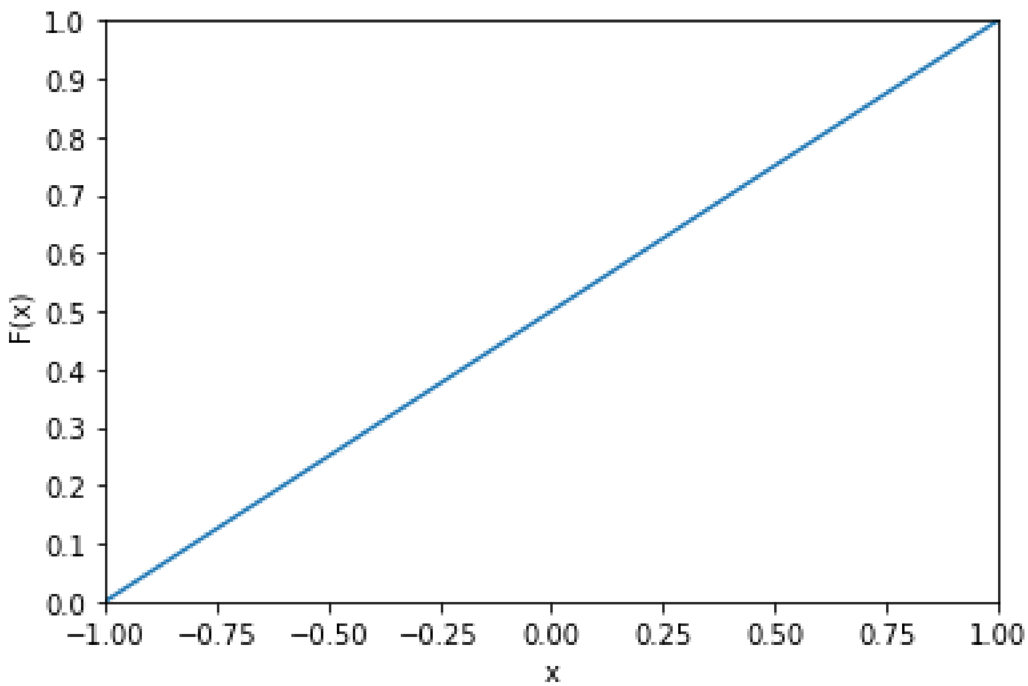
Logo

$$F(x) = \int_{-1}^{x} \frac{1}{2}\, dt$$

$$F(x) = \frac{1}{2}[x - (-1)]$$

$$F(x) = \frac{x+1}{2}$$

[ ]:
```
x = np.linspace(-1, 1, 100)
plt.plot(x, (x + 1)/2)
plt.xlim([-1, 1])
plt.ylim([0, 1])
plt.yticks(np.arange(0, 1.1, step = 0.1))
plt.xlabel('x')
plt.ylabel('F(x)')
plt.show()
```



(c) Calculate the probability of the event $X \in (-0.2, 0.2)$

Para calcular a probablidade do evento X em um intervalo, podemos usar a seguinte relação:

$$P(a \leq X \leq b) = F(b) - F(a)$$

Portanto:

$$P(-0.2 \leq X \leq 0.2) = F(0.2) - F(-0.2)$$

$$P(-0.2 \leq X \leq 0.2) = \frac{0.2+1}{2} - \frac{-0.2+1}{2}$$

$$P(-0.2 \leq X \leq 0.2) = 0.2$$

(d) Calculate the expected value $\mathbf{E}[X]$, the second $\mathbf{E}[X^2]$ and the fourth moment $\mathbf{E}[X^4]$ of the random variable. Calculate its variance $\mathrm{Var}[X]$, as well.

A esperança é definida como:

$$\mathbf{E}[X] = \int_{-\infty}^{\infty} x f(x)\, dx$$

Já seu segundo e quarto momento:

$$\mathbf{E}[X^2] = \int_{-\infty}^{\infty} x^2 f(x)\, dx$$

$$\mathbf{E}[X^4] = \int_{-\infty}^{\infty} x^4 f(x)\, dx$$

E a variância é dada por:

$$\mathrm{Var}[X] = \mathbf{E}[X^2] - \mathbf{E^2}[X]$$

Para $f(x) = \frac{1}{2} \forall x \in [-1, 1]$, temos:

Primeiro momento

$$\mathbf{E}[X] = \int_{-1}^{1} \frac{x}{2}\, dx = \frac{1}{4}[1^2 - (-1)^2]$$

$$\mathbf{E}[X] = 0$$

Segundo momento

$$\mathbf{E}[X^2] = \int_{-1}^{1} x^2 \frac{1}{2}\, dx = \frac{1}{6}[1^3 - (-1)^3]$$

$$\mathbf{E}[X^2] = \frac{1}{3}$$

Quarto momento

$$\mathbf{E}[X^4] = \int_{-1}^{1} x^4 \frac{1}{2}\, dx = \frac{1}{10}[1^5 - (-1)^5]$$

$$\mathbf{E}[X^2] = \frac{1}{5}$$

Variância

$$\mathrm{Var}[X] = \mathbf{E}[X^2] - \mathbf{E}^2[X]$$

$$\mathrm{Var}[X] = \frac{1}{3} - 0$$

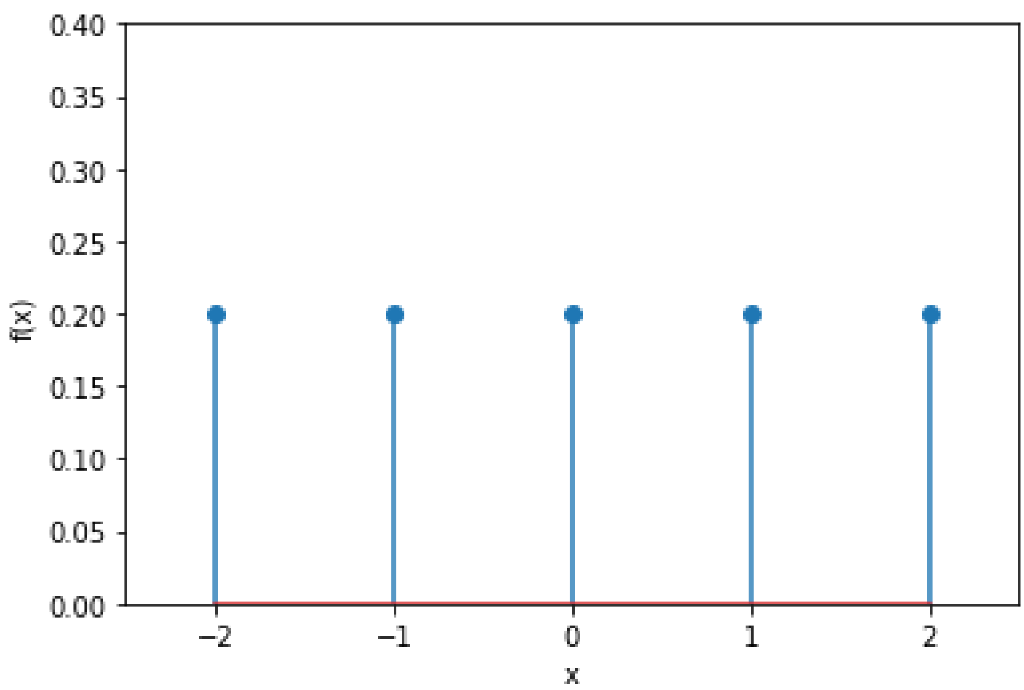
$$\mathrm{Var}[X] = \frac{1}{3}$$

Questão 02

X is a discrete-valued random variable with uniform distribution over the set $\{-2, -1, 0, 1, 2\}$. Draw its probability mass function and calculate $\mathbf{E}[X]$ and $\mathrm{Var}[X]$.

Como a distribuição é uniforme, portanto a probabilidade $p(x_i)$ dever ser igual para todo x_i, dado que a soma deve ser igual a 1.

[ ]:
```
plt.stem([-2, -1, 0, 1, 2], [0.2, 0.2, 0.2, 0.2, 0.2])
plt.xlim([-2.5, 2.5])
plt.ylim([0, 0.4])
plt.xlabel('x')
plt.ylabel('f(x)')
plt.show()
```



A esperança é dada por:

$$\mathbf{E}[X] = \sum_{i=1}^{n} x_i p(x_i)$$

E a variância por:

$$\text{Var}[X] = \mathbf{E}[X^2] - \mathbf{E^2}[X]$$

Portanto:

Primeiro momento

$$\mathbf{E}[X] = ((-2) \times 0.2) + ((-1) \times 0.2) + (0 \times 0.2) + (1 \times 0.2) + (2 \times 0.2)$$

$$\mathbf{E}[X] = 0.2 \times ((-2) + (-1) + 0 + 1 + 2)$$

$$\mathbf{E}[X] = 0$$

Segundo momento

$$\mathbf{E}[X^2] = ((-2)^2 \times (0.2)) + ((-1)^2 \times (0.2)) + (0^2 \times (0.2)) + (1^2 \times (0.2)) + (2^2 \times (0.2))$$

$$\mathbf{E}[X^2] = 0.2 \times ((-2)^2 + (-1)^2 + 0^2 + 1^2 + 2^2)$$

$$\mathbf{E}[X^2] = 0.2 \times 10$$

$$\mathbf{E}[X^2] = 2$$

Variância

$$\text{Var}[X] = \mathbf{E}[X^2] - \mathbf{E^2}[X]$$

$$\text{Var}[X] = 2$$

Questão 03

Based on the following estimators: - Sample mean: $m = \frac{1}{N}\sum_{t=1}^{N} x^t$ - Sample variance: $s^2 = \frac{1}{N}\sum_{t=1}^{N}(x^t - m)^2$ - Sample standard deviation: $s = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(x^t - m)^2}$ - Sample covariance: $s_{ij} = \frac{1}{N}\sum_{t=1}^{N}(x_i^t - m_i)(x_j^t - m_j)$ - Sample correlation coefficient: $r_{ij} = \frac{s_ij}{s_i s_j}$

Calculate r_{ij} between two features of a dataset. Choose features that you expect to be interconnected. Present all the aforementioned metrics, as well. Use a numerical software of your own choice and explain the steps of your solution.

Para esta questão, escolhi o dataset de Iris. Iris é um dataset clássico para aprendizado de técnicas de classificação, contendo 150 observações de 3 tipos de íris de uma planta. Suas features são a largura e comprimento da pétala e largura e comprimento da sépala da flor.

Primeiro, define-se as funções utilizadas para chegar na função do coeficiente de correlação.

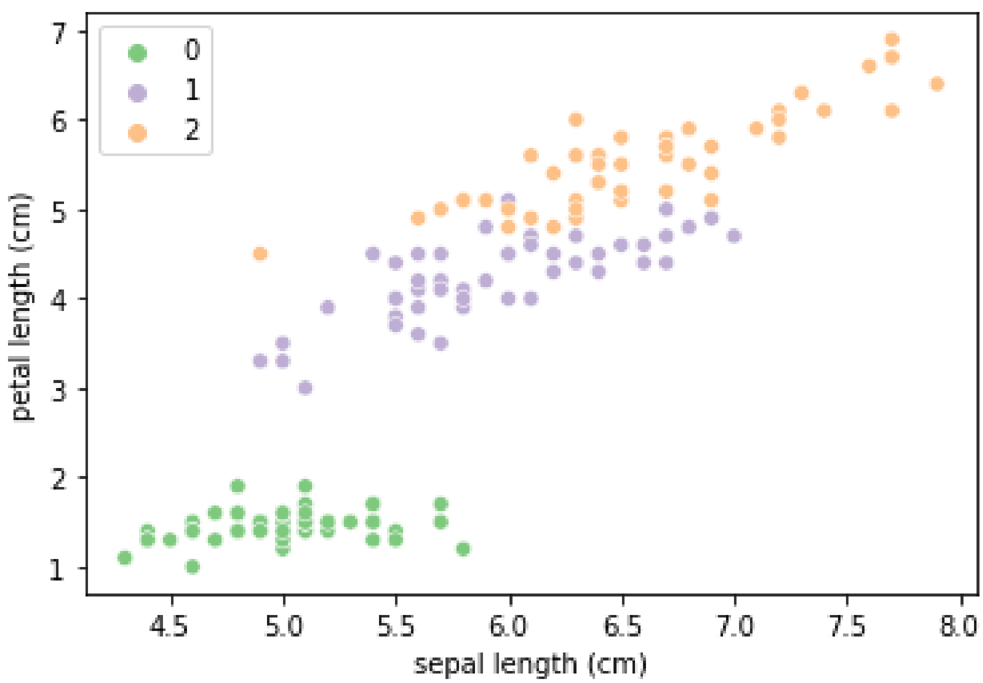
[ ]:
```
def sample_mean(sample):
    mean = 0

    for element in sample:
        mean += element

    mean = mean/len(sample)
    return mean

def sample_variance(sample):
    var = 0

    for element in sample:
        var += (element - sample_mean(sample))**2

    var = var/len(sample)
    return var

def sample_std(sample):
    return np.sqrt(sample_variance(sample))

def sample_cov(sample_i, sample_j):
    cov = 0

    for i in range(len(sample_i)):
        cov += (sample_i[i] - sample_mean(sample_i))*(sample_j[i] -␣
↪sample_mean(sample_j))

    cov = cov/len(sample_i)
    return cov

def sample_cor(sample_i, sample_j):
    return sample_cov(sample_i, sample_j)/
↪(sample_std(sample_i)*sample_std(sample_j))
```

Carrega-se o dataset para se observar as features.

```
[ ]: X = load_iris()['data']
     df = pd.DataFrame(X, columns = load_iris().feature_names)
     display(df.head(10))
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2
5	5.4	3.9	1.7	0.4
6	4.6	3.4	1.4	0.3
7	5.0	3.4	1.5	0.2
8	4.4	2.9	1.4	0.2
9	4.9	3.1	1.5	0.1

```
[ ]: sns.scatterplot(x = 'sepal length (cm)', y = 'petal length (cm)', data = df,␣
      ↪hue = load_iris().target, palette = 'Accent')
     plt.show()
```



É possível observar que o cumprimento entre as sépalas e as pétalas aparentam uma alta correlação.

```
[ ]: mean_sepal = sample_mean(X[:, 0])
     mean_petal = sample_mean(X[:, 2])
```

```
var_sepal = sample_variance(X[:, 0])
var_petal = sample_variance(X[:, 2])

std_sepal = sample_std(X[:, 0])
std_petal = sample_std(X[:, 2])

cov_sepal = sample_cov(X[:, 0], X[:, 2])

coef_corr = sample_cor(X[:, 0], X[:, 2])
```

Agora, podemos ovservar as métricas para as duas features.

[ ]:
```
print(f'Média Sépala: {mean_sepal}', f'Média Pétala: {mean_petal}')
print(f'Variância Sépala: {var_sepal}', f'Variância Pétala: {var_petal}')
print(f'Desvio Padrão Sépala: {std_sepal}', f'Desvio Padrão Pétala:␣
↪{std_petal}')

print(f'Covariância: {cov_sepal}')
print(f'Coefieciente de Correlação: {coef_corr}')
```

```
Média Sépala: 5.843333333333335 Média Pétala: 3.7580000000000027
Variância Sépala: 0.6811222222222222 Variância Pétala: 3.0955026666666674
Desvio Padrão Sépala: 0.8253012917851409 Desvio Padrão Pétala:
1.7594040657753032
Covariância: 1.2658199999999995
Coefieciente de Correlação: 0.8717537758865828
```

[ ]:
```

```