

Licenciatura Engenharia Informática

Ciência de Dados

Trabalho Data Analysis 2º Semestre

Executado por

Hugo Lopes 2349
Bernardo Silva 2348

Orientado por

Ricardo Ferreira

Entregue em
24/06/2022

1. Índice

1.	ÍNDICE	3
2.	ÍNDICE FIGURAS.....	4
3.	INTRODUÇÃO	5
4.	REGRESSÃO LINEAR.....	6
5.	CONCLUSÃO	10
6.	BIBLIOGRAFIA	11

2. Índice Figuras

Figura 1 - Dataset	6
Figura 2 - Ambiente virtual para instalar as bibliotecas.....	6
Figura 3 - Instalações das bibliotecas.....	6
Figura 4 - Correlação entre as variáveis	7
Figura 5 – código.....	7
Figura 6 - valor de β_0 , β_1 e E (Mean_Absolute_Error)	8
Figura 7 – Código para a construção do gráfico	8
Figura 8 - Gráfico	9

3. Introdução

Este trabalho foi realizado com o âmbito de solidificar os nossos conhecimentos na linguagem Python e como parâmetro de avaliação na disciplina Ciência de dados, nele vamos criar um modelo de regressão linear e explicar como fizemos e qual foi o nosso raciocínio para a construção do mesmo.

Neste trabalho tivemos que pensar em uma base de dados que conseguíssemos fazer uma relação entre os dados, de forma que fosse possível criar um gráfico com a mesma e utilizar as bibliotecas Numpy, Pandas, Sklearn, Seaborn, Matplotlib, Scipy.

4. Regressão Linear

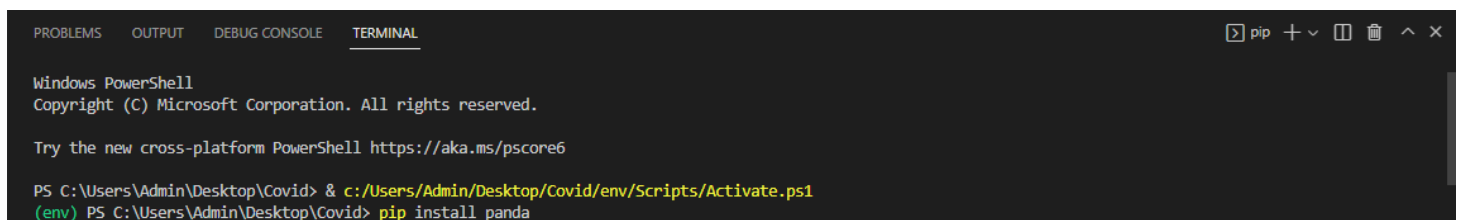
Começamos por escolher um dataset no site kaggle, um site que disponibiliza vários datasets, e entre muitas opções escolhemos um em que pudéssemos fazer alguma relação entre os seus campos.

Escolhemos um dataset sobre os casos de covid, em que a relação era entre os dias e os casos confirmados que haviam nesses dias.

Date	Confirmed
2021-01-01	4991.0
2021-01-02	5328.0
2021-01-03	4600.0
2021-01-04	3021.0
2021-01-05	5615.0
2021-01-06	6394.0

Figura 1- Dataset

Depois utilizamos o Visual Studio Code como IDE para o nosso código, tivemos de criar um ambiente virtual para instalar as bibliotecas que necessitamos para a realização deste trabalho (Numpy, Pandas, Seaborn, Matplotlib, Sklearn, Scipy).



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Admin\Desktop\Covid> & c:/Users/Admin/Desktop/Covid/env/Scripts/Activate.ps1
(env) PS C:\Users\Admin\Desktop\Covid> pip install panda
```

Figura 2 - Ambiente virtual para instalar as bibliotecas

Finalizadas as instalações das bibliotecas, tivemos que as importar no programa.

```
1 import pandas as pd
2 import numpy as np
3 from matplotlib import pyplot as plt
4 import seaborn as sns
5 import scipy.stats as st
6 from sklearn.linear_model import LinearRegression
7 from sklearn.metrics import mean_squared_error
```

Figura 3 - Instalações das bibliotecas

Agora com as importações feitas, começamos por fazer a correlação entre as variáveis do dataset.

```
data = pd.read_csv('Data.csv', usecols=['Date','Confirmed'])
print (data)
```

	Date	Confirmed
0	2021-01-01	4991.0
1	2021-01-02	5328.0
2	2021-01-03	4600.0
3	2021-01-04	3021.0
4	2021-01-05	5615.0
..
357	2021-12-24	2605.0
358	2021-12-25	2407.0

Figura 4 - Correlação entre as variáveis

Para nos facilitar os cálculos transformamos a data em os dias dos anos, por exemplo o dia 1 de janeiro corresponde ao 1 e o dia 1 de fevereiro corresponde a 32.

De seguida para a continuação do trabalho tivemos de saber a equação da reta estimada e para isso precisávamos de saber o valor de β_0 , β_1 e E.

```
data = pd.read_csv('Data.csv', usecols=['Date','Confirmed'])
data_values = data.values

k=0
for i in data_values:
    i[0]=str(k)
    k+=1
data_values = np.array(data_values,dtype=int)

X = data_values[:,0].reshape(-1,1)
Y = data_values[:,1].reshape(-1,1)
linear_regressor = LinearRegression()
Regressaol = linear_regressor.fit(X, Y)
Y_pred = linear_regressor.predict(X)

print("Valores:")
print("\tB0->" , Regressaol.intercept_[0])
print("\tB1->", format(Regressaol.coef_[0][0], '.10f'))
print("\tE->", mean_squared_error(Y,Y_pred))
```

Figura 5 – código

```
print("Valores:")
print("\tB0->" , Regressaol.intercept_[0])
print("\tB1->", format(Regressaol.coef_[0][0], '.10f'))
print("\tE->", mean_absolute_error(Y,Y_pred))
```

```
Valores:
      B0-> 11387.792246929364
      B1-> 5.4470884531
      E-> 7967.987480711498
```

Figura 6 - valor de β_0 , β_1 e E (Mean_Absolute_Error)

Equação da reta:

Com o *mean_squared_error*- $Y = \beta_0 + \beta_1 X + e \Leftrightarrow Y = 11387.792 + 5.447X + 96386745.814$

Com o *mean_absolute_error*- $Y = \beta_0 + \beta_1 X + e \Leftrightarrow Y = 11387.792 + 5.447X + 7967.987$

E por fim com estes dados já conseguimos fazer o gráfico com a reta estimada.

```
plt.scatter(X, Y)
plt.plot(X, Y_pred, color='red')
plt.ylabel('Numero de casos ')
plt.xlabel('Dias')
plt.legend(['Numero de casos por dia'])
plt.title('Casos Covid 2021')
plt.grid()
plt.show()
```

Figura 7 – Código para a construção do gráfico

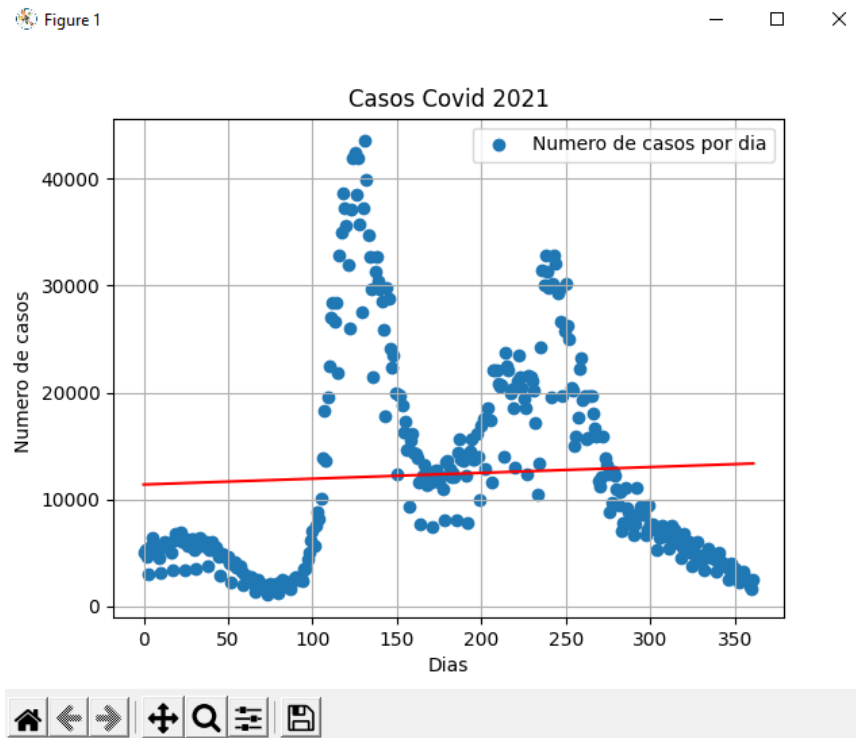


Figura 8 - Gráfico

Com este gráfico podemos ver que os dados estão muito dispersos logo podemos assumir que não existe uma correlação linear.

5. Conclusão

Com este projeto aprendemos a utilizar uma dataset para criar gráficos através da linguagem python, e a utilizar bibliotecas para a realização do mesmo.

As principais dificuldades que sentimos foi nas questões matemáticas para a criação dos gráficos e das equações, mas após alguma pesquisa conseguimos resolver o problema.

Este trabalho também nos proporcionou um aprofundamento na linguagem de python para além do que tínhamos dado na disciplina destinada a essa mesma linguagem.

6. Bibliografia

<https://medium.com/analytics-vidhya/simple-linear-regression-with-example-using-numpy-e7b984f0d15e> – Como calcular a regressão linear

<https://www.geeksforgeeks.org/python-mean-squared-error/> - Calcular o e

<https://towardsdatascience.com/linear-regression-in-6-lines-of-python-5e1d0cd05b8d> – Como calcular a regressão linear

https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html