

T3.1 / N2 e Presença em aula 24

Atividade de Presença para Probest.

Para nossa análise, X são os casos e Y são as mortes.

Essa análise será direta ao ponto. Entregando os resultados omitindo o processo. Caso queira ver todo o processo de raciocínio, olhar o outro arquivo.

```
In [1]: import pandas as pd
from regressao_linear import RegressaoLinearSimples

# Leitura de Dados:
dados = pd.read_csv('./Dados semana 1 a 20 - Covid 2021 - Página1.csv')
```

Separamos os dados em X impares (Casos) e Y impares (Obitos)

```
In [2]: impares = dados.loc[dados.Semana % 2 != 0]
X_impares = impares.Casos
Y_impares = impares.Obitos
```

Foi criado um modelo de REgressaoLinear com todos os calculos para que possa ser reutilizado. Se quiser ver o arquivo estará sendo enviado com o código fonte. Porém, como disse anteriormente, o cálculo completo estará na análise grande.

```
In [3]: model = RegressaoLinearSimples()
model.fit(X_impares.values ,Y_impares.values)
```

Aqui Já podemos prever um valor de X:

```
In [4]: # Quando tiver 450 mil casos, as mortes poderão ser de:
model.prever(450_000)
```

Out[4]: 15433.629708967546

Vamos agora solucionar as questões:

2: Encontre a reta de regressão linear Y em função de X (para as semanas ímpares das 20 semanas), se ela existir.

```
In [5]: print('A formula da reta ficou como:', model.formula_str)
```

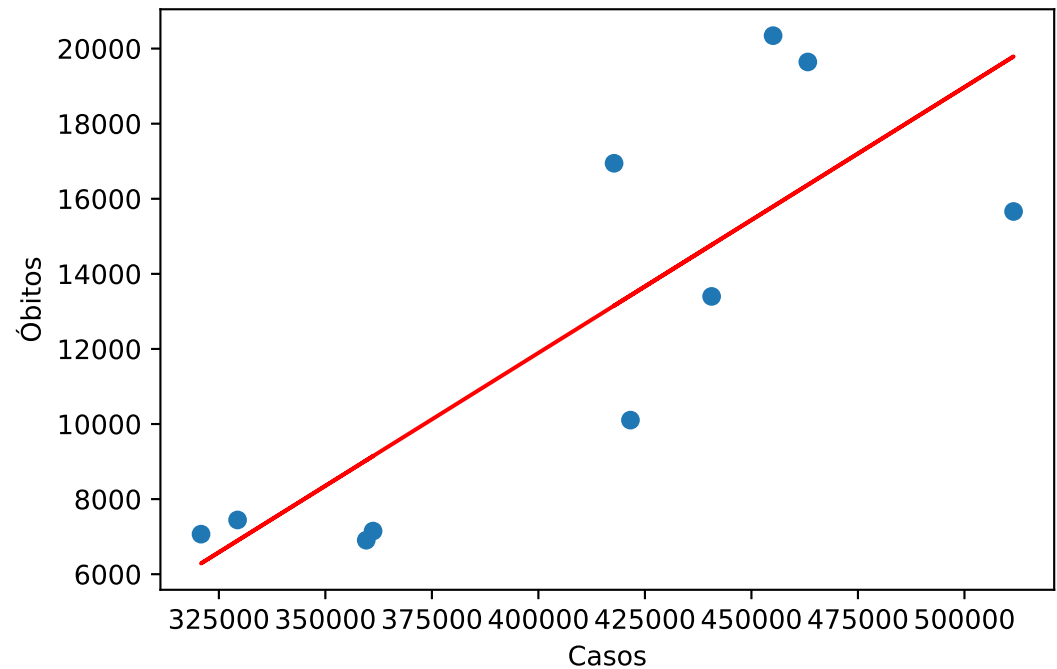
A formula da reta ficou como: Y = -16424.80179962504 + 0.07079651446353907 * X

PS: A questão 3 foi pego como referencia os dados impares, em excessão da 3.3, que pede 5 semanas pares.

3.1: Plote em um gráfico;

```
In [6]: import matplotlib.pyplot as plt
plt.scatter(X_impares, Y_impares)
plt.plot(X_impares,[model.prever(i) for i in X_impares], color='red')
plt.ylabel('Óbitos')
plt.xlabel('Casos')
```

Out[6]: Text(0.5, 0, 'Casos')



3.2 - Determine os coeficientes de determinação e correlação linear entre as variáveis X e Y;

```
In [13]: E_X, E_Y, E_X_Y = model.get_mean(X_impares.values, Y_impares.values)

previsoes = [model.prever(i) for i in X_impares]
previsoes_2 = [(p - E_Y)**2 for p in previsoes]

real = Y_impares.values
real_2 = [(r - E_Y)**2 for r in real]
```

```
In [14]: R_2 = sum(previsoes_2) / sum(real_2)
print('Coeficiente de Determinação:', R_2)
```

Coeficiente de Determinação: 0.6771566077941553

Sabemos que $R^2 = P^2$

Então:

```
In [15]: # p = raiz quadrada de R^2
from math import sqrt
p = sqrt(R_2)
print('Coeficiente de Correlação linear:', p)
```

Coeficiente de Correlação linear: 0.8228952593095644

3.3: Escolha 5 semanas pares e faça a previsão do número de mortes por semana, na reta ajustada de Y; determinando o erro cometido no processo para cada semana.

```
In [19]: pares = dados.loc[dados.Semana % 2 == 0]
pares_5 = pares[:5]

X_pares = pares_5.Casos
Y_pares = pares_5.Obitos

pares_5
```

Out[19]:

	Semana	Casos	Obitos
1	2	379061	6665
3	4	360721	7500
5	6	311959	7520
7	8	378084	8244
9	10	500099	12766

Para fazer as previsões, iremos usar da mesma forma que fizemos:

```
In [20]: modelo_par = RegressaoLinearSimples()
modelo_par.fit(X_pares.values, Y_pares.values)
```

```
In [21]: # Previsões e erros.
for index, caso in enumerate(X_pares):
    previsao_item = modelo_par.prever(caso)
    print(f'Valor real: {Y_pares.values[index]}. \nValor obtido: {previsao_item}. \nDiferença Absoluta: {previsao_item - Y_pares.values[index]}')
```

Valor real: 6665.
Valor obtido: 8323.433240465012.
Diferença Absoluta: 1658.4332404650122

Valor real: 7500.
Valor obtido: 7752.432580441369.
Diferença Absoluta: 252.43258044136928

Valor real: 7520.
Valor obtido: 6234.268230164769.
Diferença Absoluta: -1285.7317698352308

Valor real: 8244.
Valor obtido: 8293.015157321986.
Diferença Absoluta: 49.01515732198641

Valor real: 12766.
Valor obtido: 12091.850791606868.
Diferença Absoluta: -674.1492083931316

```
In [23]: print(f'Formula do Modelo PAR: {modelo_par.formula_str}')
```

Formula do Modelo PAR: Y = -3478.3160064936546 + 0.03113416903073296 * X

In []: