



ATIVIDADE 7 CA4322 TURMA 130

Nome [Hugo Emílio Nomura]

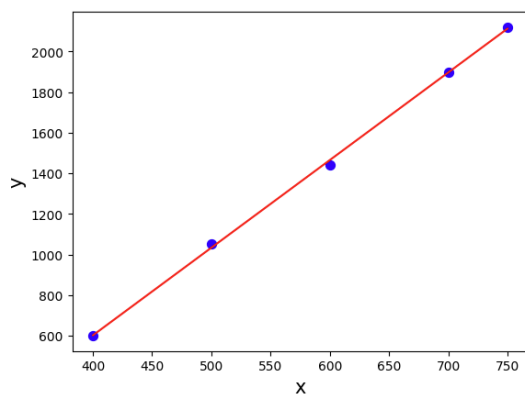
N. Matr [22.123.051-9]

- 1- A tabela abaixo refere-se à potência P (em hp), de um motor Diesel, em função do número de rotações r (em rpm). Pede-se:

$r(x)$	400	500	600	700	750
$P(y)$	600	1050	1440	1900	2120

Com base nos dados da tabela, calcular os coeficientes da regressão linear simples, o resíduo do ajuste, o gráfico dos pontos da tabela com a reta ajustada, o coeficiente de correlação de Pearson e o coeficiente de determinação R^2 .

Gráfico:



Comandos:

```
# coeficientes da regressão linear simples
n = np.size(x)
Sx= np.sum(x)
Sy=np.sum(y)
Sxy=np.sum(x*y)
Sxx=np.sum(x*x)
a1=(n*Sxy-Sx*Sy) / (n*Sxx-Sx**2) #Calcula o coeficiente a1 da reta
a0=(Sxx*Sy-Sxy*Sx) / (n*Sxx-Sx**2) #Calcula o coeficiente a0 da reta
print(a1)
print(a0)
```

```
#funcao que calcula o Residuo
def Residuo(x,y,b0,b1):
    n = len(y)
    RS = 0
    for i in range(0,n):
        y_pred=b0+b1*x[i]
        RS = RS + (y[i]-y_pred)**2
    return RS
```

```
print('RS:', Residuo(x,y,a0,a1))
```

```
# r2
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score

x = np.array([400, 500, 600, 700, 750])
y = np.array([600, 1050, 1440, 1900, 2120])
x = np.array(x)
x = x.reshape(-1, 1)
modelo = LinearRegression().fit(x, y)
R2 = r2_score(y, modelo.predict(x))
print('R2:', R2)
```

```
# Pearson
import numpy as np
x = np.array([400, 500, 600, 700, 750])
y = np.array([600, 1050, 1440, 1900, 2120])
correlacao = np.corrcoef(x, y)
print(correlacao)
```

```
# grafico
import numpy as np
import matplotlib.pyplot as plt

# define os dados
x = np.array([400, 500, 600, 700, 750])
y = np.array([600, 1050, 1440, 1900, 2120])

# mostra os dados
plt.scatter(x, y, color = "b", marker = "o", s = 50)
# prediz os valores
y_pred = a0 + a1*x
# mostra a reta de regressão
plt.plot(x, y_pred, color = "r")

plt.xlabel('x', fontsize = 15)
plt.ylabel('y', fontsize = 15)
plt.show(True)
```

Resultados:

```
a1 = 4.318292682926829
```

```
a2 = -1125.7926829268292
```

```
Resíduo = 972.56097560976
```

```
Pearson = 0.99968214
```

```
R2 = 0.9993643724670541
```

2) Uma corrida de um taxi é cobrada pela distância percorrida conforme a tabela abaixo. Determine: **(a)** O polinômio interpolador de Lagrange usando a função **(b)** Quanto o passageiro pagará se a distância for $d = 9$ Km usando a função polyfit.? Verificar também o resultado usando a função interp. **(c)** se o passageiro dispõe de R\$24,00 até que distância poderá ir?

x(Km)	1	7	14
y(\$reais)	10.5	17.7	35.2

Comandos:

```
# Lagrange
import sympy as sy
import numpy as np
from scipy.interpolate import lagrange
from numpy.polynomial.polynomial import Polynomial
from scipy.interpolate import lagrange
x=[1, 7, 14]
y=[10.5, 17.7, 35.2]
x = np.array(x)
y = np.array(y)
poly = lagrange(x, y)
Polynomial(poly).coef
```

```
# b
z = np.polyfit(x, y, 2)
p(9)
```

```
# c
np.roots(p-24)
```

Resultados:

```
Coeficientes do polinômio de Lagrange [ 0.1,  0.4, 10]
O passageiro pagará (utilizando polyfit) = 21.7
O passageiro pagará (utilizando interp) = 22.7
Com R$24,00 o passageiro poderá rodar 10km
```

3. A partir dos dados do dataset 'mtcars.csv', apresentar as primeiras 25 do dataset e determinar os coeficientes da regressão múltipla com as variáveis independentes gear, qsec e cyl e com a variável dependente hp. Indentificar a variável mais significativa e a menos significativa.

OBS: usar como leitura do dataset a seguinte linha:

```
data=pd.read_csv('mtcars.csv')
```

Comandos:

```
# ler o data set e printar as 25 primeiras linhas
import pandas as pd
import numpy as np
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt
from sklearn import datasets
df = pd.read_csv('mtcars.csv', header=(0))
print("Número de linhas e colunas:",df.shape)
df.head(25)

# coeficientes
import seaborn as sns
import statsmodels.formula.api as sm
model = sm.ols(formula="hp ~ gear+qsec+cyl ", data=df)
result = model.fit()
print(result.summary())
```

Resultados:

	model	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
0	Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
1	Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
2	Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
3	Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
4	Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
5	Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
6	Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
7	Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
8	Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
9	Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4
10	Merc 280C	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4
11	Merc 450SE	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3
12	Merc 450SL	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3
13	Merc 450SLC	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3
14	Cadillac Fleetwood	10.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4
15	Lincoln Continental	10.4	8	460.0	215	3.00	5.424	17.82	0	0	3	4
16	Chrysler Imperial	14.7	8	440.0	230	3.23	5.345	17.42	0	0	3	4
17	Fiat 128	32.4	4	78.7	66	4.08	2.200	19.47	1	1	4	1
18	Honda Civic	30.4	4	75.7	52	4.93	1.615	18.52	1	1	4	2
19	Toyota Corolla	33.9	4	71.1	65	4.22	1.835	19.90	1	1	4	1
20	Toyota Corona	21.5	4	120.1	97	3.70	2.465	20.01	1	0	3	1
21	Dodge Challenger	15.5	8	318.0	150	2.76	3.520	16.87	0	0	3	2
22	AMC Javelin	15.2	8	304.0	150	3.15	3.435	17.30	0	0	3	2
23	Camaro Z28	13.3	8	350.0	245	3.73	3.840	15.41	0	0	3	4
24	Pontiac Firebird	19.2	8	400.0	175	3.08	3.845	17.05	0	0	3	2
25	Fiat X1-9	27.3	4	79.0	66	4.08	1.935	18.90	1	1	4	1

	coef

Intercept	-127.6725
gear	29.6531
qsec	-3.2578
cyl	36.0666

A variável menos significativa é qsec e a mais é cyl

	coef	std err	t	P> t

Intercept	-127.6725	176.405	-0.724	0.475
gear	29.6531	12.901	2.298	0.029
qsec	-3.2578	5.748	-0.567	0.575
cyl	36.0666	6.457	5.585	0.000