

## Ficha laboratorial – Regressão – parte 1 (R. Linear)

**Ex1.** Corre o código abaixo. O ficheiro “RegressData001.p” encontra-se no Moodle.

```
1 import numpy as np
2 from matplotlib import pyplot as plt
3 import pickle
```

```
1 file = open('RegressData001.p', 'rb')
2 D = pickle.load(file)
3 file.close()
```

```
1 x=D['x']
2 y=D['y']
```

```
1 #Projeção dos dados
2 plt.figure()
3 plt.plot(x,y, 'b+')
4 plt.grid()
5 plt.show()
```

Responde à Q1 do formulário que se encontra em:

<https://forms.gle/N2VuUhM1WWLuQhf6>

**Ex.2** Corre os dois códigos apresentados abaixo.

```
1 #Cálculo das matrizes de optimização da regressão
2 n_points=x.shape[1]
3 X=np.vstack((np.ones((1,n_points)),x))
4 Rx=np.dot(X,X.T)
5 rxy=np.dot(X,y.T)
```

```
1 #Estimar wopt para a reta de regressão
2 w=np.dot(np.linalg.pinv(Rx),rxy)
3 print(w)
```

Responde à Q2 e Q3 do referido formulário.

**Ex. 3** Corre os dois códigos apresentados abaixo.

```
1 #Dois pontos para desenhar a reta de regressão
2 x_min=np.min(x)
3 x_max=np.max(x)
4 p1=np.array([x_min-1,x_max+1])
5 p1_temp=np.vstack((np.ones([1,2]),p1))
6 p2=np.dot(w.T,p1_temp)
7 P=np.vstack((p1,p2))
```

```
1 #Desenhar a reta de regressão
2 plt.figure()
3 plt.plot(x,y, 'b+')
4 plt.plot(P[0,:],P[1,:], 'r-')
5 plt.grid()
6 plt.show()
```

**Ex. 4** Corre o código apresentado abaixo.

```
1 #Cálculo e gráfico dos erros de regressão
2 yh=np.dot(w.T,X)
3 erros=y-yh
4 x.shape
5 plt.figure()
6 plt.axis((x_min-1.,x_max+1.,np.min(erros)*3,np.max(erros)*3))
7 plt.plot(x,erros,'k.')
8 plt.grid()
9 plt.show()
```

Responde à Q4 do referido formulário.

**EX. 5** Corre o código apresentado abaixo.

```
1 #Cálculo do coeficiente de regressão
2 my=np.mean(y)
3 SQtot=np.sum((y-my)**2)
4 SQexp=np.sum((yh-my)**2)
5 R2=SQexp/SQtot
6
```

Responde à Q5 do referido formulário.