```
# Importing numpy as np
import numpy as np
```

```
# Consider the Y vector to be a variable labeled "Missed Days at Work";
# there are 12 rows (objects)

Y=np.array([1,0,1,4,3,2,5,6,9,13,15,16])
print(Y)
Y.shape
```

```
[ 1  0  1  4  3  2  5  6  9 13 15 16]
(12,)
```

```
# Consider column 2 of X to be a variable labeled "Attitude Toward Work" - a 1 to 13 point
# rating scale where 1 is extremely favorable and 13 is extremely unfavorable;
# and consider column 3 of X to be a variable labeled "Years in Present Position";
# X has 12 rows and 3 columns;  the first column is all 1's

X=np.array([[1,1,1],
           [1,2,1],
           [1,2,2],
           [1,3,2],
           [1,5,4],
           [1,5,6],
           [1,6,5],
           [1,7,4],
           [1,10,8],
           [1,11,7],
           [1,11,9],
           [1,12,10]])
print(X)
X.shape
```

```
[[ 1  1  1]
 [ 1  2  1]
 [ 1  2  2]
 [ 1  3  2]
 [ 1  5  4]
 [ 1  5  6]
 [ 1  6  5]
 [ 1  7  4]
 [ 1 10  8]
 [ 1 11  7]
 [ 1 11  9]
 [ 1 12 10]]
(12, 3)
```

```
# Transpose of X
# Here I am taking XT as X-Transpose
```

```
XT=X.T
print(XT)
XT.shape
```

```
[[ 1  1  1  1  1  1  1  1  1  1  1  1]
 [ 1  2  2  3  5  5  6  7 10 11 11 12]
 [ 1  1  2  2  4  6  5  4  8  7  9 10]]
(3, 12)
```

```
# Multiplication of X-Transpose and X
# Here I am taking XM as multiplication of X-Transpose
XM=XT@X
print(XM)
XM.shape
```

```
[[ 12  75  59]
 [ 75 639 497]
 [ 59 497 397]]
(3, 3)
```

```
# Matrix Inverse
MI=np.linalg.inv((XT)@X)
print(MI)
```

```
[[ 0.3169944  -0.0214686  -0.02023368]
 [-0.0214686   0.06093854 -0.07309775]
 [-0.02023368 -0.07309775  0.09703619]]
```

```
#calculating betas
#for finding betas we have to multiple Matrix Inverse and X Transpose and Y
betas=MI@XT@Y
print(betas)
betas.shape
```

```
[-2.2630379   1.54972927 -0.2385295 ]
(3,)
```

```
# Calculating Betas
# Here is another way of finding betas
# betas=np.linalg.inv((XT)@X)@(X.T)@Y
# print(betas)
# betas.shape
```

```
# Calculating 3 beta values
```

```
beta_0=betas[0]
print("Beta 0 :", beta_0)
```

```
beta_1=betas[1]
print("Beta 1 :", beta_1)

beta_2=betas[2]
print("Beta 2 :", beta_2)
```

```
Beta 0 : -2.2630379025362557
Beta 1 : 1.5497292675976007
Beta 2 : -0.2385294955827837
```

Regression Equation : $Y\hat{}= \beta0\hat{} + \beta1\hat{} * X1 + \beta2\hat{} * X2 + \varepsilon$

Y^ = -2.2630379 + 1.54972927 * X1 + -0.2385295 * X2 + ε

Colab paid products - Cancel contracts here

0s completed at 9:38 PM