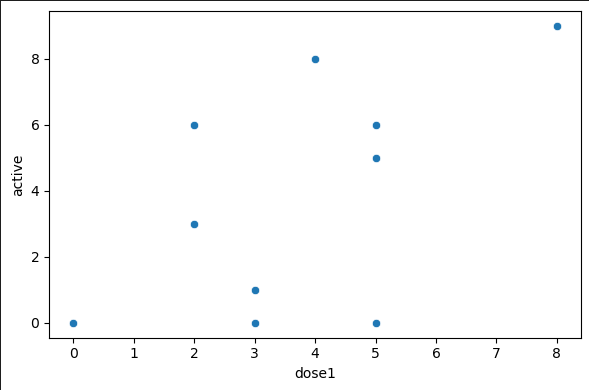
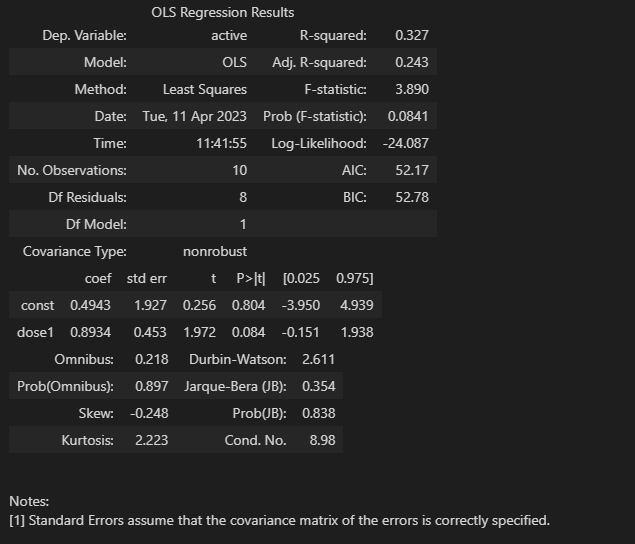
Linear regression

The below image is the scatter plot of dose1 with respect to active cases.Here on X axis dose1 is taken as it is independent feature.On y-axis we have taken active cases present with respect to dose1.



The below chart is

OLS (Ordinary Least Squares): Most common method to estimate the linear regression (It finds the line that minimizes the sum of square error (SSE))



From the OLS regression chart we have taken the coeff of dose1: 0.8934 and constant coeff : 0.4943 and put the value in the linear equation: y= mx + b where a is the dose1 coeff , and b is the constant.

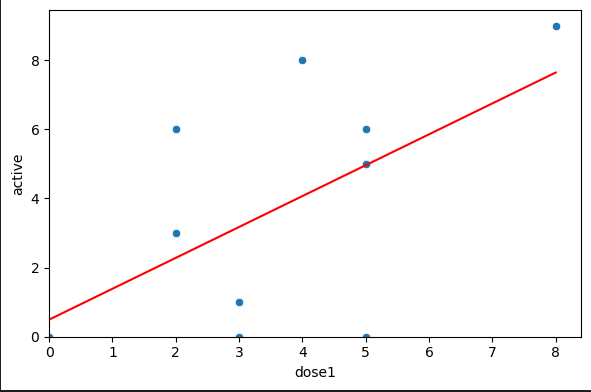
# dose1 coef: 0.8934

# Constant coef: 0.4943

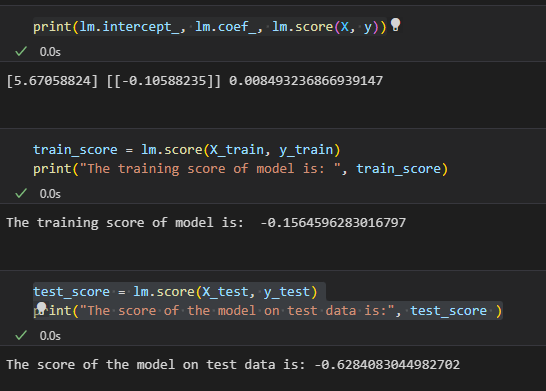
# Linear equation: 𝑦 = 𝑎𝑥 + 𝑏

y\_pred = 0.8934\*x['dose1'] + 0.4943

The graph belows shows the regression line which is represented by red colour.



Another method:





We have data of two features x, y. We want to fit an equation

y=ax+b

The least-squares best fit is

**Where ρ**

Is correlation coefficient, σ - standard deviation

Let us see, how it works on Iris:

>> iris= load('IRIS.DAT');

>> size(iris)

ans =

150 4

>> y=iris(:,1);

>> x=iris(:,2);

>> rho=corr(x,y)

rho =

-0.1176

>> plot(x,y,b\*)

plot(x,y,b\*)

|

Error: Unbalanced or unexpected parenthesis or bracket.

>> plot(x,y,'b\*')

>> a=rho\*std(y)/std(x)

a =

-0.2234

>> b=mean(y)-a\*mean(x)

b =

6.5262

>> det=rho^2

det =

0.0138

>> ii=[1:30];

>> yt=y';

>> yt(ii)

ans =

Columns 1 through 8

5.1000 4.4000 4.4000 5.0000 5.1000 4.9000 5.0000 4.6000

Columns 9 through 16

5.0000 4.8000 4.8000 5.0000 5.1000 5.0000 5.1000 4.9000

Columns 17 through 24

5.3000 4.3000 5.5000 4.8000 5.2000 4.8000 4.9000 4.6000

Columns 25 through 30

5.7000 5.7000 4.8000 5.2000 4.7000 4.5000

>> yc=a\*x(ii)+b;

>> yc'

ans =

Columns 1 through 8

5.7445 5.8115 5.8561 5.7445 5.6775 5.8338 5.8115 5.8115

Columns 9 through 16

5.7891 5.7668 5.8561 5.7445 5.7891 5.7668 5.6775 5.8561

Columns 17 through 24

5.6998 5.8561 5.7445 5.7668 5.7668 5.8338 5.7221 5.8338

Columns 25 through 30

5.5434 5.6775 5.8561 5.6104 5.8115 6.0125

>> yy(1,:)=yt(ii);

>> yy(2,:)=yc';

>> yy

yy =

Columns 1 through 8

5.1000 4.4000 4.4000 5.0000 5.1000 4.9000 5.0000 4.6000

5.7445 5.8115 5.8561 5.7445 5.6775 5.8338 5.8115 5.8115

Columns 9 through 16

5.0000 4.8000 4.8000 5.0000 5.1000 5.0000 5.1000 4.9000

5.7891 5.7668 5.8561 5.7445 5.7891 5.7668 5.6775 5.8561

Columns 17 through 24

5.3000 4.3000 5.5000 4.8000 5.2000 4.8000 4.9000 4.6000

5.6998 5.8561 5.7445 5.7668 5.7668 5.8338 5.7221 5.8338

Columns 25 through 30

5.7000 5.7000 4.8000 5.2000 4.7000 4.5000

5.5434 5.6775 5.8561 5.6104 5.8115 6.0125

>> abs(yy(1,:)-yy(2,:))

ans =

Columns 1 through 8

0.6445 1.4115 1.4561 0.7445 0.5775 0.9338 0.8115 1.2115

Columns 9 through 16

0.7891 0.9668 1.0561 0.7445 0.6891 0.7668 0.5775 0.9561

Columns 17 through 24

0.3998 1.5561 0.2445 0.9668 0.5668 1.0338 0.8221 1.2338

Columns 25 through 30

0.1566 0.0225 1.0561 0.4104 1.1115 1.5125

>> 100\*abs(yy(1,:)-yy(2,:))./yc'

ans =

Columns 1 through 8

11.2188 24.2876 24.8652 12.9596 10.1709 16.0068 13.9632 20.8461

Columns 9 through 16

13.6313 16.7649 18.0347 12.9596 11.9039 13.2967 10.1709 16.3271

Columns 17 through 24

7.0141 26.5728 4.2556 16.7649 9.8286 17.7209 14.3674 21.1492

Columns 25 through 30

2.8244 0.3972 18.0347 7.3157 19.1254 25.1558

>> re=100\*abs(yy(1,:)-yy(2,:))./yc'

re =

Columns 1 through 8

11.2188 24.2876 24.8652 12.9596 10.1709 16.0068 13.9632 20.8461

Columns 9 through 16

13.6313 16.7649 18.0347 12.9596 11.9039 13.2967 10.1709 16.3271

Columns 17 through 24

7.0141 26.5728 4.2556 16.7649 9.8286 17.7209 14.3674 21.1492

Columns 25 through 30

2.8244 0.3972 18.0347 7.3157 19.1254 25.1558

>> mean(er)

Undefined function or variable 'er'.

>> mean(re)

ans =

14.5978