BASIC LINEAR METHODS FOR REGRESSION

Linear regression analysis is used to predict the value of a variable based on the value of another variable. The variable you want to predict is called the dependent variable. The variable you are using to predict the other variable's value is called the independent variable.

This form of analysis estimates the coefficients of the linear equation, involving one or more independent variables that best predict the value of the dependent variable. Linear regression fits a straight line or surface that minimizes the discrepancies between predicted and actual output values. There are simple linear regression calculators that use a “least squares” method to discover the best-fit line for a set of paired data. You then estimate the value of X (dependent variable) from Y (independent variable).

**You can perform the linear regression method**in a programs and environments including **☺**

* R linear regression
* MATLAB linear regression
* Sklearn linear regression
* Linear regression Python
* Excel linear regression

**Why linear regression is important ☺**

Linear-regression models are relatively simple and provide an easy-to-interpret mathematical formula that can generate predictions. Linear regression can be applied to various areas in business and academic study.

You’ll find that linear regression is used in everything from biological, behavioral, environmental and social sciences to business. Linear-regression models have become a proven way to scientifically and reliably predict the future. Because linear regression is a long-established statistical procedure, the properties of linear-regression models are well understood and can be trained very quickly.

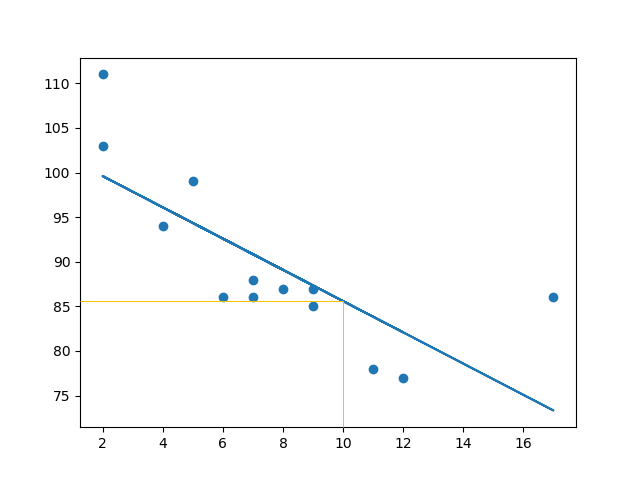
**Here’s how you can check for these assumptions:**

1. The variables measured at a continuous level. Examples of continuous variables are time, sales, weight and test scores.
2. Use a scatterplot to find out quickly if there is a linear relationship between those two variables.
3. The observations should be independent of each other (that is, there should be no dependency).
4. Your data should have no significant outliers.
5. Check for homoscedasticity- a statistical concept in which the variances along the best-fit linear-regression line remain similar all through that line.
6. The residuals (errors) of the best-fit regression line follow normal distribution.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13 | **from** **scipy** **import** stats  x = [**5**,**7**,**8**,**7**,**2**,**17**,**2**,**9**,**4**,**11**,**12**,**9**,**6**]  y = [**99**,**86**,**87**,**88**,**111**,**86**,**103**,**87**,**94**,**78**,**77**,**85**,**86**]  slope, intercept, r, p, std\_err = stats.linregress(x, y)  **def** **myfunc**(x):  **return** slope \* x + intercept  speed = myfunc(**10**)  **print**(speed) |

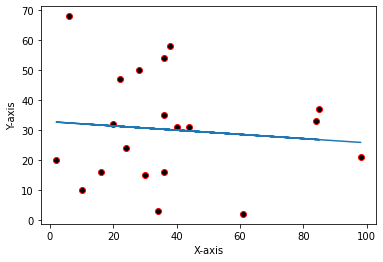
**Out\_put : 85.59308314937454**

predicted a speed at 85.6, which we also could read from the diagram



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18 | **import** **matplotlib.pyplot** **as** **plt**  **from** **scipy** **import** stats  x = [**98**,**40**,**34**,**36**,**85**,**20**,**36**,**44**,**38**,**10**,**16**,**24**,**28**,**84**,**6**,**2**,**36**,**61**,**22**,**30**]  y = [**21**,**31**,**3**,**35**,**37**,**32**,**54**,**31**,**58**,**10**,**16**,**24**,**50**,**33**,**68**,**20**,**16**,**2**,**47**,**15**]  slope, intercept, r, p, std\_err = stats.linregress(x, y)  **def** **myfunc**(x):  **return** slope \* x + intercept  mymodel = list(map(myfunc, x))  plt.scatter(x, y,edgecolor ="red",c="black")  plt.plot(x, mymodel)  plt.xlabel("X-axis")  plt.ylabel("Y-axis")  plt.show() |

**Out\_put :**

****

**Different approaches to solve linear regression models**

 There are many different methods that we can apply to our linear regression model in order to make it more efficient. But we will discuss the most common of them here.

1. **Gradient Descent**
2. **Least Square Method / Normal Equation Method**
3. **Adams Method**
4. **Singular Value Decomposition (SVD)**

**# We will learn these 4 methods in details as next blog ☺**