Assumptions- Linear Regression

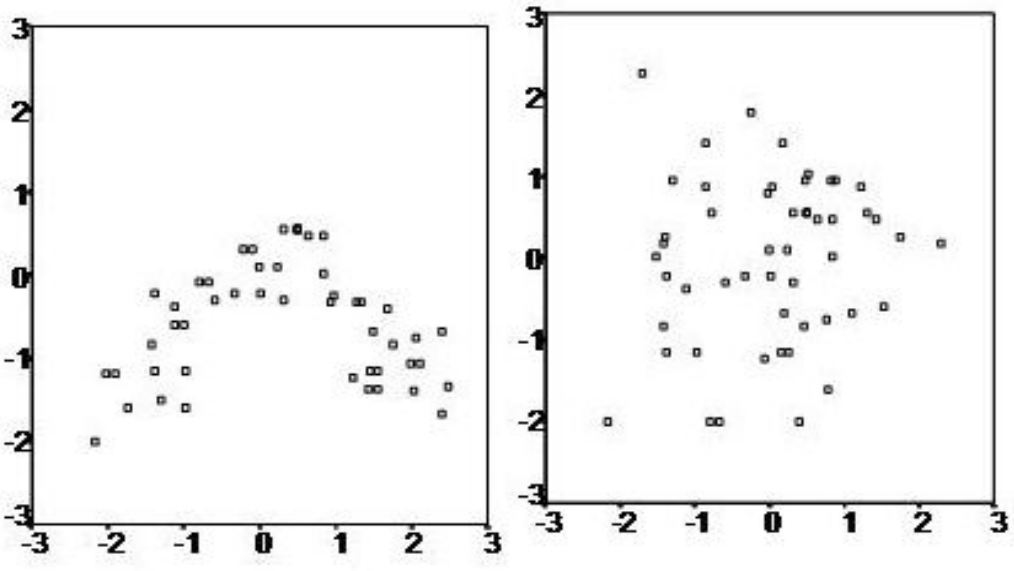
INTERNSHIPSTUDIO

- Linear regression is an analysis that assesses whether one or more predictor variables explain the dependent (criterion) variable.

- The regression has few key assumptions:
  - Linear relationship
  - Multivariate normality
  - No or little multicollinearity
  - No auto-correlation

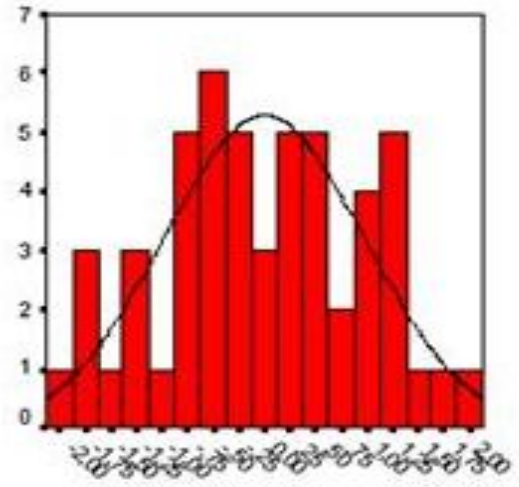
Assumptions of Linear Regression...

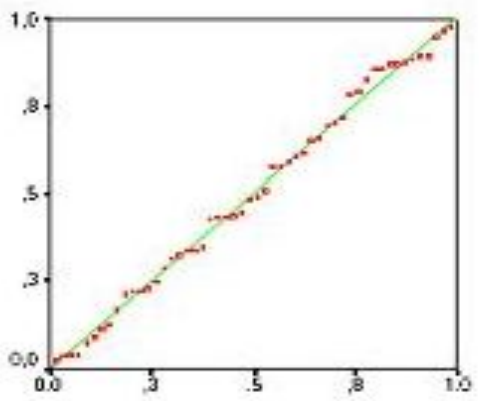
- First, linear regression needs the relationship between the independent and dependent variables to be linear.
- It is also important to check for outliers since linear regression is sensitive to outlier effects.
- The linearity assumption can best be tested with scatter plots, the following two examples depict two cases, where no and little linearity is present.



Assumptions of Linear Regression...

- Secondly, the linear regression analysis requires all variables to be multivariate normal.
- When the data is not normally distributed a non-linear transformation (e.g., log-transformation) might fix this issue.
- Thirdly, linear regression assumes that there is little or no multicollinearity in the data.
- Multicollinearity occurs when the independent variables are too highly correlated with each other.

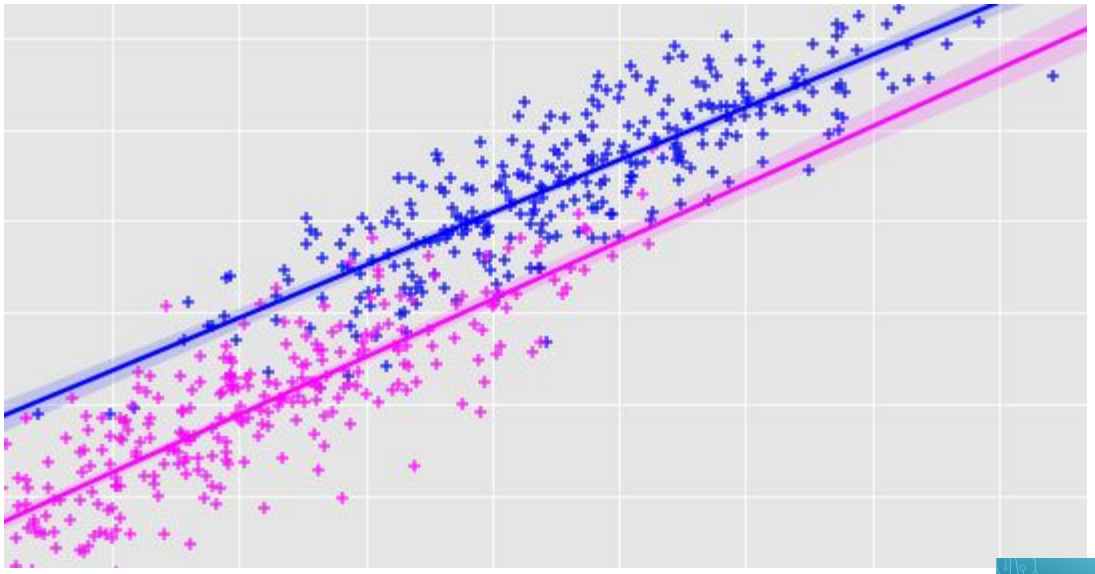




Multivariable Linear Regression

INTERNSHIPSTUDIO

- Multiple linear regression (MLR), also known simply as multiple regression, is a statistical technique that uses several explanatory variables to predict the outcome of a response variable.
- The goal of multiple linear regression (MLR) is to model the linear relationship between the independent variables and dependent variable.



Linear Regression

INTERNSHIPSTUDIO

Formula and Calcualtion of Multiple Linear Regression

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip} + \epsilon$$

where, for $i = n$ observations:

y_i = dependent variable

x_i = expanatory variables

β_0 = y-intercept (constant term)

β_p = slope coefficients for each explanatory variable

ϵ = the model's error term (also known as the residuals)

- Multiple regression is an extension of linear (OLS) regression that uses just one explanatory variable.
- MLR is used extensively in econometrics and financial inference.

Implementing MLR

INTERNSHIPSTUDIO

- import all the required libraries :

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as seabornInstance
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn import metrics
%matplotlib inline
```

- The following command imports the dataset from the file you uploded:

  dataset = pd.read_csv('winequality.csv')

- Let's explore the data a little bit by checking the number of rows and columns in it.

  dataset.shape

- To see the statistical details of the dataset, we can use describe():

  dataset.describe()

MLR- Data cleaning

INTERNSHIPSTUDIO

- Let us clean our data little bit, So first check which are the columns the contains NaN values in it :

  dataset.isnull().any()

- Once the above code is executed, all the columns should give False, In case for any column you find True result, then remove all the null values from that column using below code.

  dataset = dataset.fillna(method='ffill')

- Our next step is to divide the data into “attributes” and “labels”. X variable contains all the attributes/features and y variable contains labels.

```
X = dataset[['fixed acidity', 'volatile acidity', 'citric acid',
'residual sugar', 'chlorides', 'free sulfur dioxide', 'total sulfur
dioxide', 'density', 'pH', 'sulphates','alcohol']].values

y = dataset['quality'].values
```

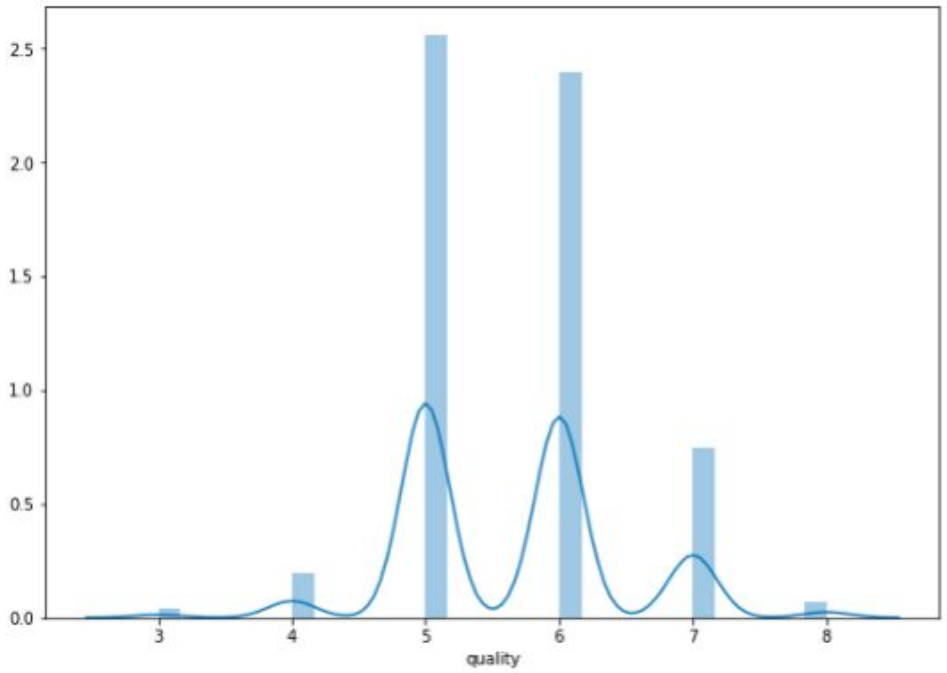
Implementing MLR....

Linear Regression

INTERNSHIPSTUDIO

- Let's check the average value of the “quality” column.

```
plt.figure(figsize=(15,10))
plt.tight_layout()
seabornInstance.distplot(dataset['quality'])
```



Average value of the quality of the wine.

Implementing Multivariable Linear Regression model....

INTERNSHIPSTUDIO

- As we can observe that most of the time the value is either 5 or 6.

- Next, we split 80% of the data to the training set while 20% of the data to test set using below code.

```
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=0)
```

- Now lets train our model.

```
regressor = LinearRegression()
regressor.fit(X_train, y_train)
```

Implementing Multivariable Linear Regression model….

INTERNSHIPSTUDIO

- *This means that for a unit increase in "density", there is a decrease of 31.51 units in the quality of the wine.*
- *Similarly, a unit decrease in "Chlorides" results in an increase of 1.87 units in the quality of the wine.*
- *We can see that the rest of the features have very little effect on the quality of the wine.*

- Now let's do prediction on test data.

```
y_pred = regressor.predict(X_test)
```

- Check the difference between the actual value and predicted value.

```
df = pd.DataFrame({'Actual': y_test, 'Predicted': y_pred})
df1 = df.head(25)
```

INTERNSHIPSTUDIO

1. Explain multivariable Linear Regression with key points?

2. Show the implementation of MLR with example?

3. Define all the variables in MLR

4. How do we print Mean Absolute Error, Mean Squared Error, Root Mean Squared Error ?