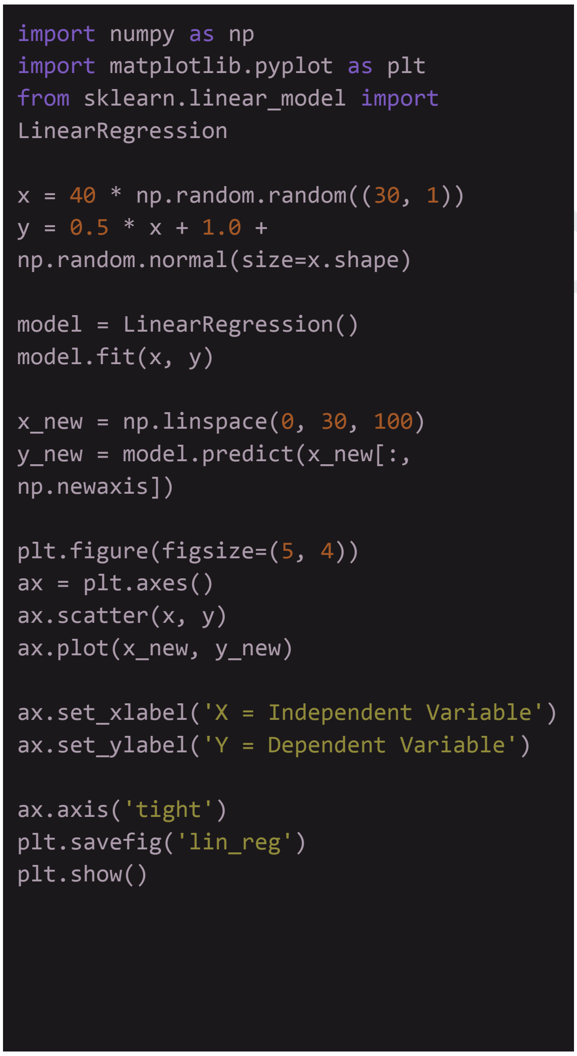
# Linear Regression

A machine learning approach called linear regression is used to predict the value of continuous response variables. Because the value of the response/target variables must be provided and utilised for training the models,

The predictive analytics issues that are solved using linear regression models are referred to as supervised learning problems. Additionally, keep in mind that the term "continuous" refers to the response variable's numerical character and ability to accept an unlimited number of values. A category of parametric models includes linear regression models.

For data that are linear in character, linear regression models perform exceptionally well. In other words, the target/response/dependent variable is linearly related to the predictor/independent variables in the data set. The linear relationship between the response and the predictor variable is illustrated by the following.1



```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import
LinearRegression

x = 40 * np.random.random((30, 1))
y = 0.5 * x + 1.0 +
np.random.normal(size=x.shape)

model = LinearRegression()
model.fit(x, y)

x_new = np.linspace(0, 30, 100)
y_new = model.predict(x_new[:,
np.newaxis])

plt.figure(figsize=(5, 4))
ax = plt.axes()
ax.scatter(x, y)
ax.plot(x_new, y_new)

ax.set_xlabel('X = Independent Variable')
ax.set_ylabel('Y = Dependent Variable')

ax.axis('tight')
plt.savefig('lin_reg')
plt.show()
```

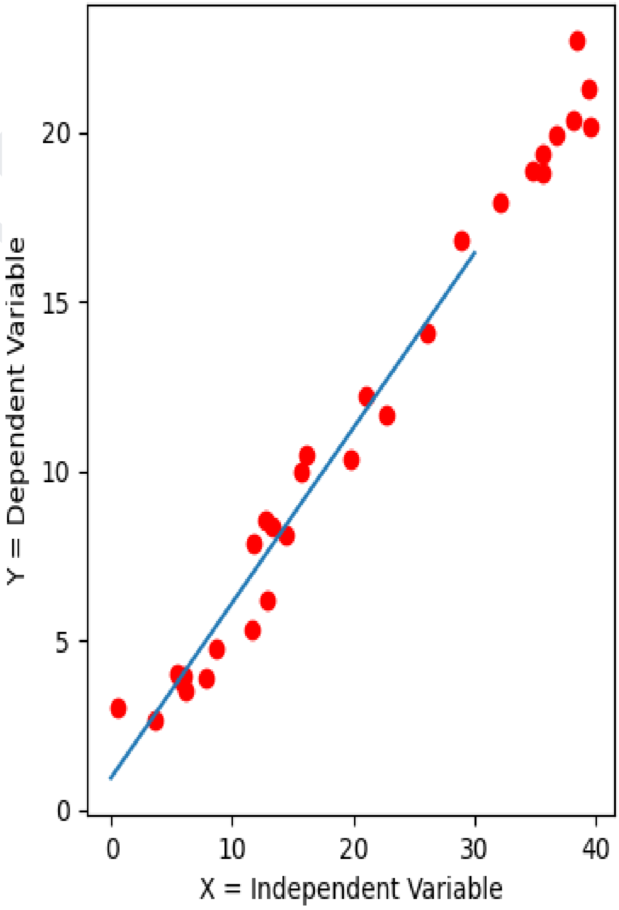


**Fig 1 Linear Regrassion Model**

The blue line in the above diagram is termed as best-fit line and can be found by training the model such as Y = mX + c

Linear regression models are of two different kinds.

1. Simple linear regression

2. Multiple linear regression.

**1. Simple linear regression:** A linear regression is referred to as simple linear regression when there is only one independent or predictor variable, as there is in this example with Y = mX + c

**2. Multiple linear regression:** is used when there are several independent or predictor variables, such as in the equation Y=w1x1+w2x2+...+wnxn is called multiple linear regression.

## Linear Regression Concepts:

**Residual Error:** The difference between the actual and anticipated values is referred to as residual error. When visualising in terms of best fit line, the positive residual error is when the real value is greater than the best fit line, and the negative residual error is when the actual value is less than the best fit line. The diagram below depicts the same situation.

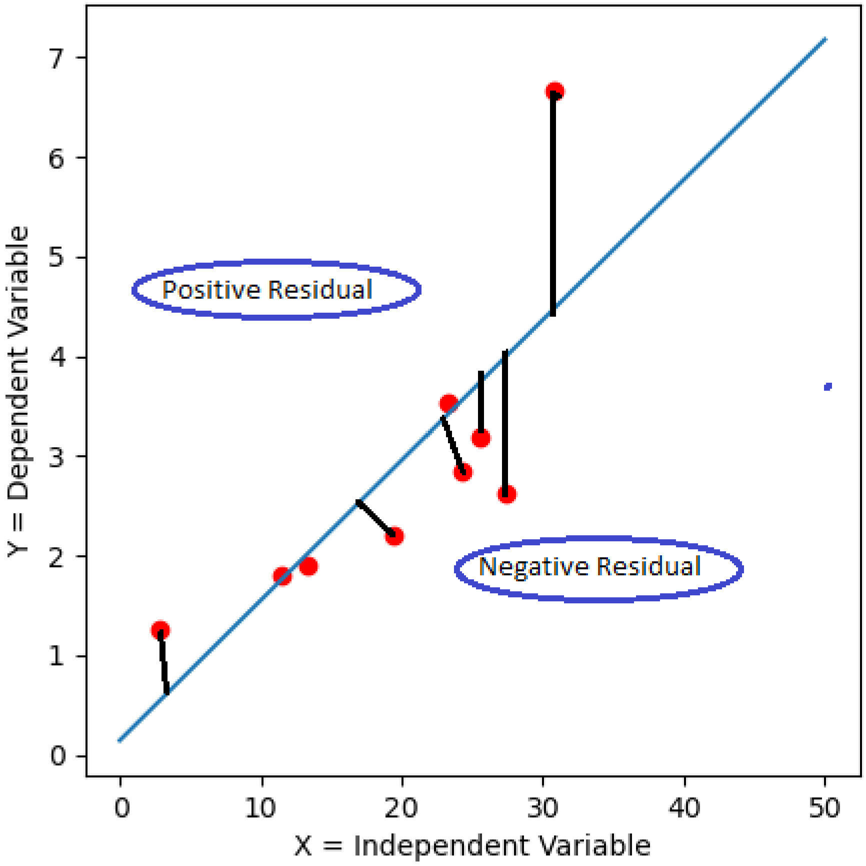


**Fig 2. Residual error – Linear Regression.**

The following are the key concept must be account when dealing with regrassion problem.

**Sum of Square Total (SST):** Squares Total is calculated as the total of the squared differences between the response variable's actual values and their mean. It is also known as the response's variance. Remember that variance is calculated as the product of the squared difference between each observation and the mean of all observations. Other names for it include Total Sum of Squares (TSS).

**Sum of Square Error (SSE):** The total of the squared difference between the actual value and the anticipated value is known as the sum of square error, also known as sum of square residual error. Residual Sum of Squares is another name for it.

**Sum of Sqare Regrassion (SSR):**The squared difference between the projected value and the mean of the actual values is what is referred to as a sum of square regression, or SSR. ESS (Explained Sum of Squares) is another name for it.

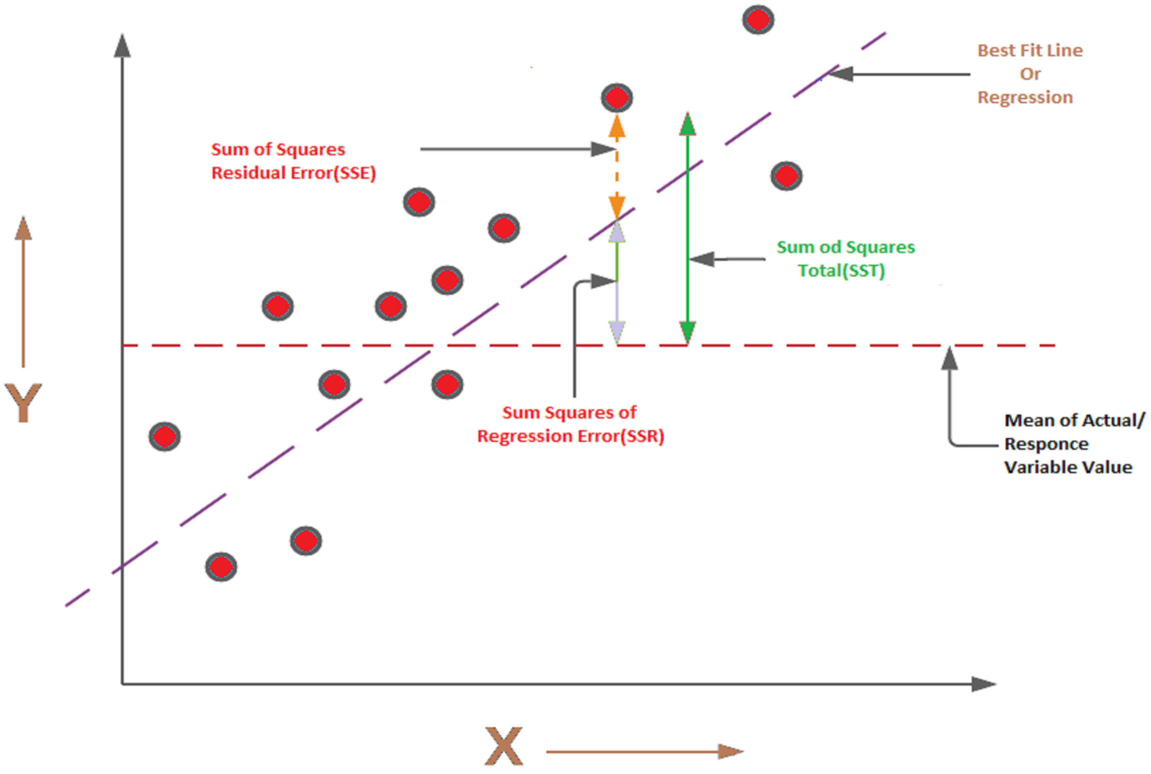


**Fig 3. SSR, SSE and SST representation in relation to Linear Regression**

### Relation Between SSR, SSE and SST

From the fig 3 we can see the relation ship between the SSR, SSR and SST.

$$\mathbf{SST = SSR + SSE}$$

**R-squared** is a measurement of how well the regression or best fit line fits the data. We also called bast fit line as "coefficient of determination". The Ratio Sum of Squares Regression (SSR) to Sum of Squares Total (SST) is how mathematics describes it.

$$\text{R-Squared} = SSR / SST = (SST - SSE) / SST = 1 - (SSE / SST)$$

A statistical evaluation of a linear regression model's quality of fit is the R-squared value. It reflects the percentage of the dependent variable's variation that the independent variables in the regression equation can account for.

The R-squared value is between 0 and 1, or between 0% and 100%. A higher R-squared value denotes that the independent variables may explain a greater percentage of the variance in the dependent variable. In other words, a higher R-squared value denotes that the regression line better matches the data points and the model can predict the dependent variable with more accuracy.

It's crucial to remember, though, that R-squared shouldn't be the only factor in determining how good a regression model is. Other things to think about are the importance of the independent variables, whether the model fits the data, and any possible problems like multicollinearity or outliers.

Furthermore, while a high R-squared value suggests a strong fit to the training data, it doesn't imply that the model will perform well on fresh, untried data. Overfitting can provide a high R-squared value but poor prediction performance when a model captures noise or random oscillations in the training data instead of the underlying correlations. In order to evaluate the model's performance, it is crucial to use methods like cross-validation or to evaluate it on a different test dataset.

In conclusion, a higher R-squared value often implies a better fit of the regression line to the data, but care should be taken in how you interpret it and you should also take other aspects into account when evaluating the model's overall quality and predictive power.

**Python Code Example:**
Here is the Python code for linear regression, which uses a housing dataset to train a regression model to forecast home prices. In the code provided below, pay close attention to some of the following

```
import warnings
warnings.filterwarnings('ignore')
import numpy as np
import pandas as pd
from sklearn.linear_model import LinearRegression
from sklearn.pipeline import make_pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score
from sklearn import datasets
#
# Load the Sklearn Boston Dataset
#
# load_boston() returns sklearn.utils.Bunch
boston = datasets.load_boston()
df = pd.DataFrame(data=boston.data,
                  columns=boston.feature_names)
X = boston.data
y = boston.target
#
# Create a training and test split
#
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
random_state=42)
#
# Fit a pipeline using Training dataset and related labels
#
pipeline = make_pipeline(StandardScaler(), LinearRegression())
pipeline.fit(X_train, y_train)
#
# Calculate the predicted value for training and test dataset
#
y_train_pred = pipeline.predict(X_train)
y_test_pred = pipeline.predict(X_test)
#
# Mean Squared Error
#
print('MSE train: %.2f, test: %.2f' % (mean_squared_error(y_train,
y_train_pred),
                mean_squared_error(y_test, y_test_pred)))
#
# R-Squared
#
print('R^2 train: %.2f, test: %.2f' % (r2_score(y_train, y_train_pred),
r2_score(y_test, y_test_pred)))
```

## Summary

- Linear regression is a supervised machine learning algorithm used to predict the value of the continuous random variables.
- When there is just one predictor or independent variable, it is called simple linear regression.
- When there are two or more predictors or independent variables, it is called multiple linear regression
- R-Squared is a metric that can be used to evaluate the linear regression model performance. It explains the variability of the response variable which is explained by the regression model. The higher the R-squared value, the better the variability explained by the regression model. However, one would need to take caution.
- R-Squared can be expressed as a function of SSE (Sum of Squares Residual Error) and SST (Sum of Squares Total)

Written by
Sanket Wade
sanketwade95@gmail.com