**30.How to make prediction in simple linear regression in python?**

**Objective:**

* To predict the dependent variable for the corresponding independent variable using python.

**Process:**

* Import necessary libraries.
* Load the sample data set.
* Assign the independent(X) and dependent(y) variables.
* Train our data set in 70:30 manner.
* Pass the X\_train and Y\_train in to the linear model.
* Predict Y according to X.

**Input:**

* Data set(Sample data).

**Output:**

* Predicted value of Y according to X.

**Source code:**

#import the libraries

import pandas as pd

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LinearRegression

#Sample data set

data={'age':[25,26,25,23,30,29,23,24,26,25],

'rating':[4,3.6,2.5,2.25,4.5,4.4,3.9,3.5,3.7,3.2],

'bonus':[1300,1400,1250,1100,1500,1450,1150,1100,1250,1200],

'salary':[2500,2600,2400,2200,3000,2900,2300,2200,2450,2350],}

#create data frame

df=pd.DataFrame(data)

#Measuring descriptive statistics

print(df.describe())

#Take independent variable

X = df.iloc[:, :1].values

#Take dependent variable

y = df.iloc[:, 1].values

#Train our data

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.7, random\_state=0)

#building linear model

regressor = LinearRegression()

#Fit the variable to the linear model

t=regressor.fit(X\_train, y\_train)

#print the linear model results

print("Regression intercept is\n",regressor.intercept\_)

print("Regression coefficient is\n",regressor.coef\_)

#make the predictions

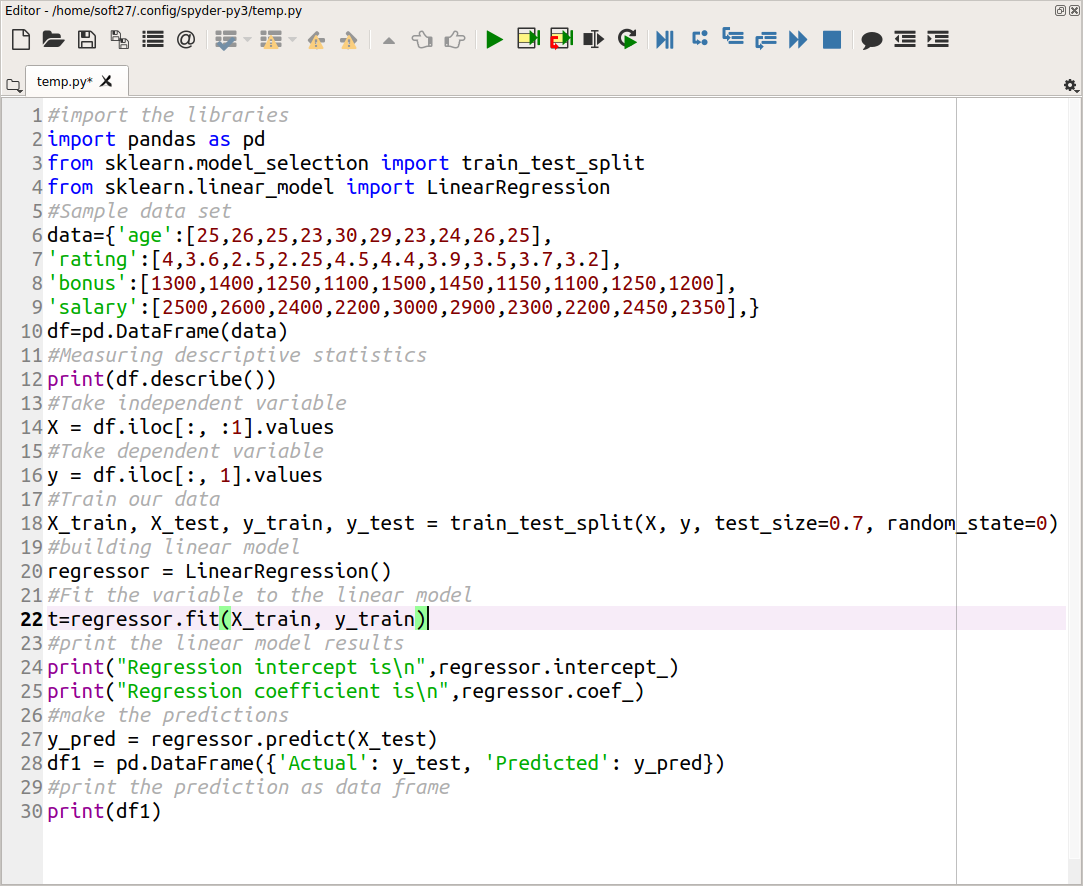
y\_pred = regressor.predict(X\_test)

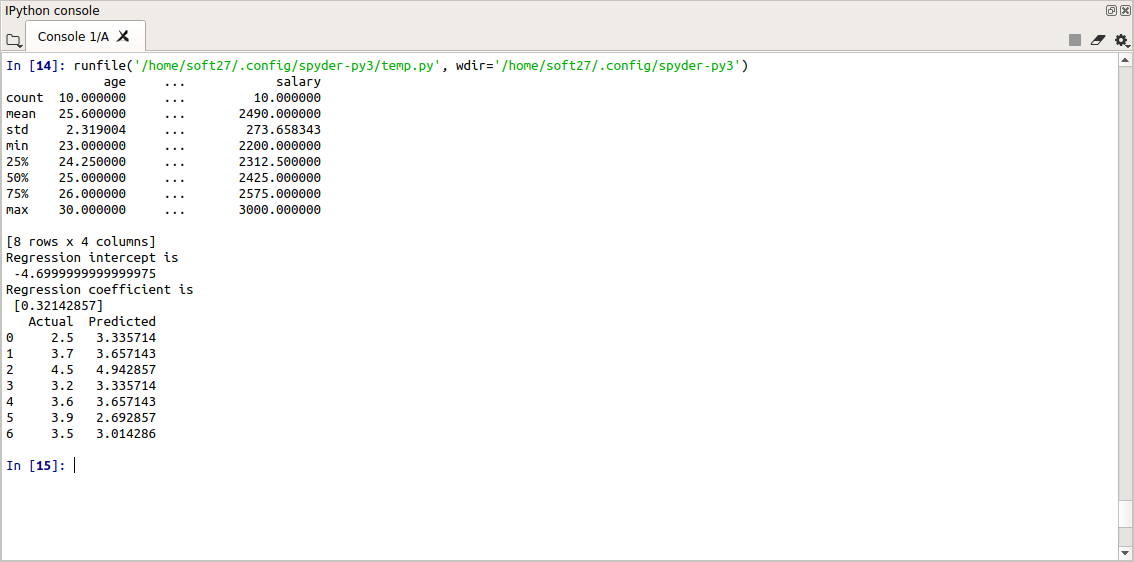
df1 = pd.DataFrame({'Actual': y\_test, 'Predicted': y\_pred})

#print the prediction as data frame

print(df1)

**Screen shot:**

****

****