

```

# import numpy, matplotlib, pandas
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

#read the data set
dataset =pd.read_csv(r'C:\Users\DELL\NIT\3MAR\Investment.csv')
dataset
# check the data set having any missing values
dataset.isnull().sum()

# creating independent variables in x using iloc(selecting rows)
x = dataset.iloc[:, :-1] # target variable
#y= dataset.iloc[:, -1] # we can call like this also
y= dataset.iloc[:,4]
# getting dummy variables (Convert categorical variable into dummy/indicator variables.)
x = pd.get_dummies(x,dtype=int)

from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x,y,test_size=0.2,random_state=0)

from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(x_train, y_train)

y_pred =regressor.predict(x_test)

m_slope = regressor.coef_
print(m_slope)

c_inter = regressor.intercept_
print(c_inter)

x = np.append( arr = np.ones((50,1)).astype(int), values=x, axis=1)

#x = np.append(arr=np.full((50,1), 42467.5), values=x, axis=1)

import statsmodels.api as sm
x_opt = x[:,[0,1,2,3,4,5]]
#OrdinaryLeastSquares
regressor_OLS= sm.OLS(endog=y,exog=x_opt).fit()
regressor_OLS.summary()

# Backward elimination based on p value p=0.05

import statsmodels.api as sm
x_opt = x[:,[0,1,2,3,5]]
#OrdinaryLeastSquares
regressor_OLS= sm.OLS(endog=y,exog=x_opt).fit()
regressor_OLS.summary()

import statsmodels.api as sm
x_opt = x[:,[0,1,2,3]]
#OrdinaryLeastSquares
regressor_OLS= sm.OLS(endog=y,exog=x_opt).fit()
regressor_OLS.summary()

```

```
import statsmodels.api as sm
x_opt = x[:,[0,1,3]]
#OrdinaryLeastSquares
regressor_OLS= sm.OLS(endog=y,exog=x_opt).fit()
regressor_OLS.summary()
```

```
import statsmodels.api as sm
x_opt = x[:,[0,1]]
#OrdinaryLeastSquares
regressor_OLS= sm.OLS(endog=y,exog=x_opt).fit()
regressor_OLS.summary()
```