

Class Task

Linear Regression:

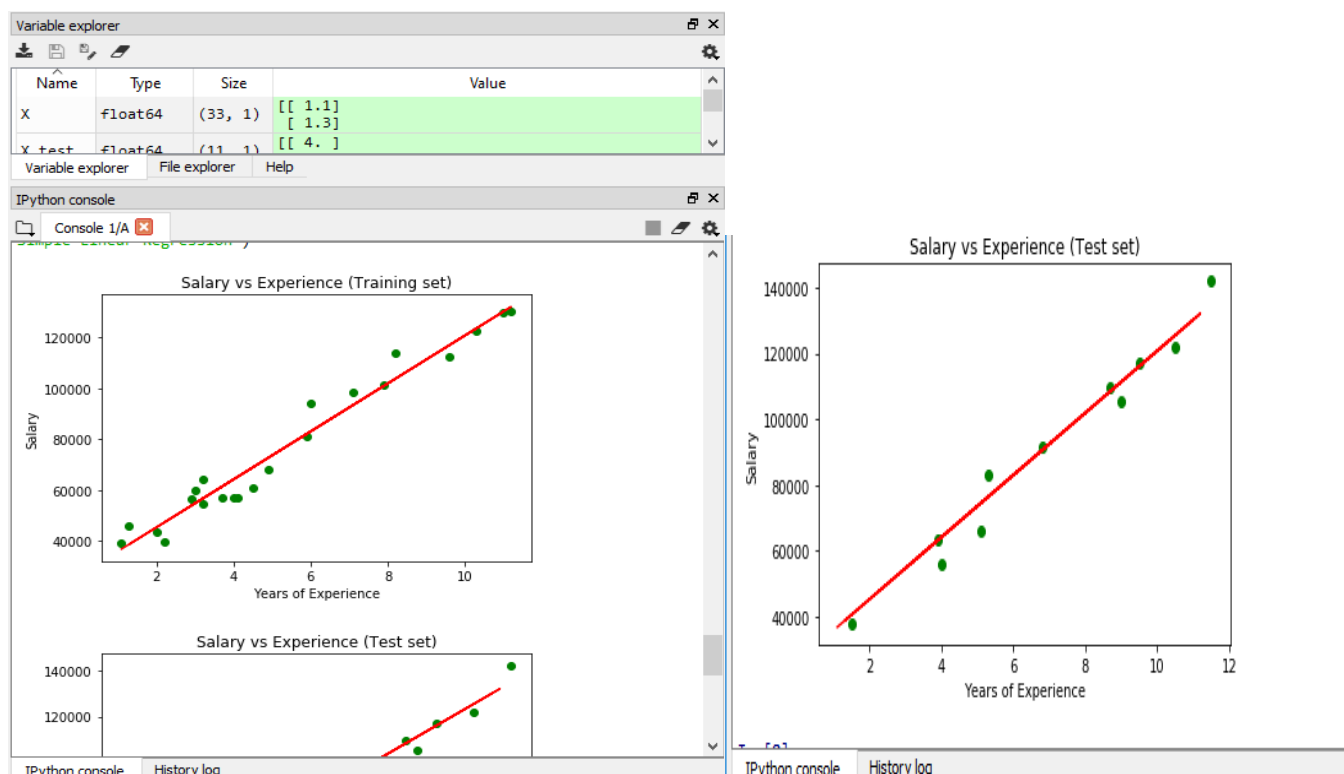
-Plot colour changed

-New Values added

```
Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help

temp.py simple_linear_regression.py

2
3 # Importing the libraries
4 import numpy as np
5 import matplotlib.pyplot as plt
6 import pandas as pd
7
8 # Importing the dataset
9 dataset = pd.read_csv('Salary_Data.csv')
10 X = dataset.iloc[:, :-1].values
11 y = dataset.iloc[:, 1].values
12
13 # Splitting the dataset into the Training set and Test set
14 from sklearn.model_selection import train_test_split
15 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 1/3, random_state = 0)
16
17 # Fitting Simple Linear Regression to the Training set
18 from sklearn.linear_model import LinearRegression
19 regressor = LinearRegression()
20 regressor.fit(X_train, y_train)
21
22 # Predicting the Test set results
23 y_pred = regressor.predict(X_test)
24
25 # Visualising the Training set results
26 plt.scatter(X_train, y_train, color = 'green')
27 plt.plot(X_train, regressor.predict(X_train), color = 'red')
28 plt.title('Salary vs Experience (Training set)')
29 plt.xlabel('Years of Experience')
30 plt.ylabel('Salary')
31 plt.show()
32
33 # Visualising the Test set results
34 plt.scatter(X_test, y_test, color = 'green')
35 plt.plot(X_train, regressor.predict(X_train), color = 'red')
36 plt.title('Salary vs Experience (Test set)')
37 plt.xlabel('Years of Experience')
38 plt.ylabel('Salary')
39 plt.show()
```



Polynomial Regression:

-new values added

-change of degree

```
temp.py | simple_linear_regression.py | polynomial_regression.py
8 # Importing the dataset
9 dataset = pd.read_csv('Position_Salaries.csv')
10 X = dataset.iloc[:, 1:2].values
11 y = dataset.iloc[:, 2].values
12
13 # Splitting the dataset into the Training set and Test set
14 """from sklearn.cross_validation import train_test_split
15 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 0)"""
16
17
18 # Fitting Linear Regression to the dataset
19 from sklearn.linear_model import LinearRegression
20 lin_reg = LinearRegression()
21 lin_reg.fit(X, y)
22
23 # Fitting Polynomial Regression to the dataset
24 from sklearn.preprocessing import PolynomialFeatures
25 poly_reg = PolynomialFeatures(degree = 7)
26 X_poly = poly_reg.fit_transform(X)
27 poly_reg.fit(X_poly, y)
28 lin_reg_2 = LinearRegression()
29 lin_reg_2.fit(X_poly, y)
30
31 # Visualising the Linear Regression results
32 plt.scatter(X, y, color = 'green')
33 plt.plot(X, lin_reg.predict(X), color = 'orange')
34 plt.title('Truth or Bluff (Linear Regression)')
35 plt.xlabel('Position level')
36 plt.ylabel('Salary')
37 plt.show()
38
39 # Visualising the Polynomial Regression results
40 plt.scatter(X, y, color = 'green')
41 plt.plot(X, lin_reg_2.predict(poly_reg.fit_transform(X)), color = 'orange')
42 plt.title('Truth or Bluff (Polynomial Regression)')
43 plt.xlabel('Position level')
44 plt.ylabel('Salary')
45 plt.show()

24 from sklearn.preprocessing import PolynomialFeatures
25 poly_reg = PolynomialFeatures(degree = 7)
26 X_poly = poly_reg.fit_transform(X)
27 poly_reg.fit(X_poly, y)
28 lin_reg_2 = LinearRegression()
29 lin_reg_2.fit(X_poly, y)
30
31 # Visualising the Linear Regression results
32 plt.scatter(X, y, color = 'green')
33 plt.plot(X, lin_reg.predict(X), color = 'orange')
34 plt.title('Truth or Bluff (Linear Regression)')
35 plt.xlabel('Position level')
36 plt.ylabel('Salary')
37 plt.show()
38
39 # Visualising the Polynomial Regression results
40 plt.scatter(X, y, color = 'green')
41 plt.plot(X, lin_reg_2.predict(poly_reg.fit_transform(X)), color = 'orange')
42 plt.title('Truth or Bluff (Polynomial Regression)')
43 plt.xlabel('Position level')
44 plt.ylabel('Salary')
45 plt.show()
46
47 # Visualising the Polynomial Regression results (for higher resolution and smoother curve)
48 X_grid = np.arange(min(X), max(X), 0.1)
49 X_grid = X_grid.reshape((len(X_grid), 1))
50 plt.scatter(X, y, color = 'red')
51 plt.plot(X_grid, lin_reg_2.predict(poly_reg.fit_transform(X_grid)), color = 'blue')
52 plt.title('Truth or Bluff (Polynomial Regression)')
53 plt.xlabel('Position level')
54 plt.ylabel('Salary')
55 plt.show()
56
57 # Predicting a new result with Linear Regression
58 lin_reg.predict([[6.5]])
59
60 # Predicting a new result with Polynomial Regression
61 lin_reg_2.predict(poly_reg.fit_transform([[6.5]]))
```

Name	Type	Size	Value
X	int64	(13, 1)	$\begin{bmatrix} 1 \\ 2 \end{bmatrix}$
X_grid	float64	(120, 1)	$\begin{bmatrix} 1. \\ 2. \end{bmatrix}$

Variable explorer | File explorer | Help

IPython console

Console 1/A

In [22]: runfile('C:/Users/Mominah Baig/Desktop/ML/Part 2 - Regression-20200322T064230Z-001/Part 2 - Regression/2 - Polynomial Regression/polynomial_regression.py', wdir='C:/Users/Mominah Baig/Desktop/ML/Part 2 - Regression-20200322T064230Z-001/Part 2 - Regression/2 - Polynomial Regression')

