Polynomial Regression

Importing the libraries

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

Importing the dataset

```
dataset = pd.read_csv('Data.csv')
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, -1].values
```

Splitting the dataset into the Training set and Test set

```
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)
```

```
print(X_train)
```

```
[[  11.22   43.13 1017.24   80.9 ]
 [  13.67   54.3  1015.92   75.42]
 [  32.84   77.95 1014.68   45.8 ]
 ...
 [  16.81   38.52 1018.26   75.21]
 [  12.8    41.16 1022.43   86.19]
 [  32.32   67.9  1006.08   37.93]]
```

```
print(y_train)
```

```
[473.93 467.87 431.97 ... 459.01 462.72 428.12]
```


```
print(y_train)
```

```
[473.93 467.87 431.97 ... 459.01 462.72 428.12]
```


Training the Polynomial Regression model on the Training set

```
from sklearn.preprocessing import PolynomialFeatures
from sklearn.linear_model import LinearRegression
poly_reg = PolynomialFeatures(degree = 4)
X_poly = poly_reg.fit_transform(X_train)
regressor = LinearRegression()
regressor.fit(X_poly, y_train)
```

```
LinearRegression
LinearRegression()
```

Predicting the Test set results

```
y_pred = regressor.predict(poly_reg.transform(X_test))
np.set_printoptions(precision=2)
print(np.concatenate((y_pred.reshape(len(y_pred),1), y_test.reshape(len(y_test),1)),1))
```

```
[[434.16 431.23]
 [458.26 460.01]
 [460.72 461.14]
 ...
 [469.49 473.26]
 [438.53 438.  ]
 [461.62 463.28]]
```

Evaluating the Model Performance

```
from sklearn.metrics import r2_score
r2_score(y_test, y_pred)
```

```
0.9455261542316076
```