Notebook Position_Salaries.csv ×

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('/Position_Salaries.csv')
X=dataset.iloc[:,1: -1].values
y=dataset.iloc[:, -1].values
print(X)
print(y)
```

Training the Linear Regression model on the whole dataset

```
from sklearn.linear_model import LinearRegression
lin_reg=LinearRegression()
lin_reg.fit(X,y)
```

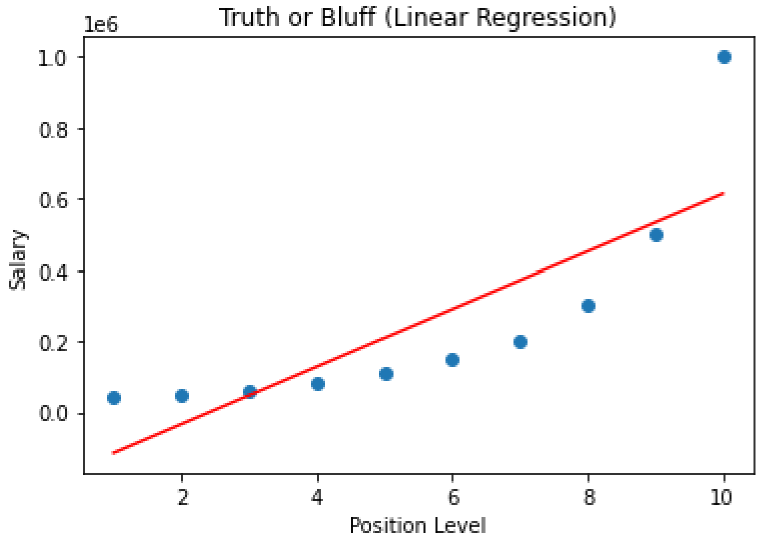
Training the Polynomial Regression model on the whole dataset

```
from sklearn.preprocessing import PolynomialFeatures
poly_reg=PolynomialFeatures(degree=4)
X_poly=poly_reg.fit_transform(X)
lin_reg_2=LinearRegression()
lin_reg_2.fit(X_poly, y)
```

```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=False)
```

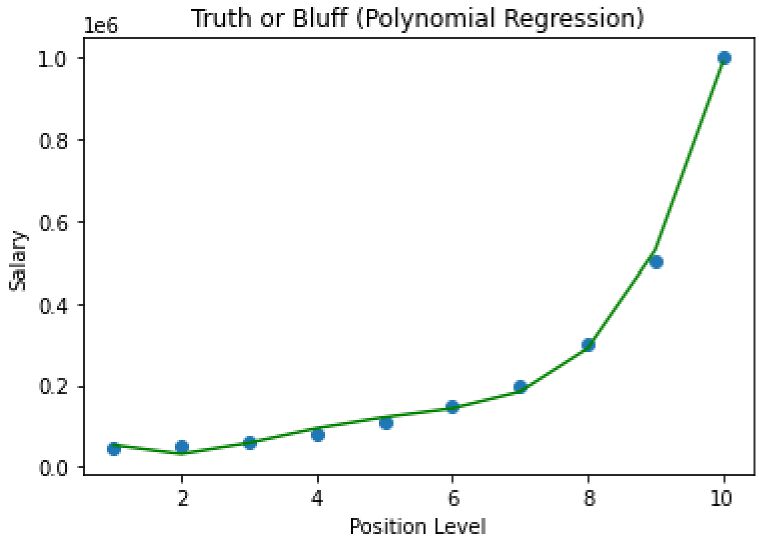
Visualising the Linear Regression results

```
plt.scatter(X,y)
plt.scatter(X,y)
plt.plot(X,lin_reg.predict(X), color='red')
plt.title('Truth or Bluff (Linear Regression)')
plt.xlabel("Position Level")
plt.ylabel("Salary")
plt.show()
```



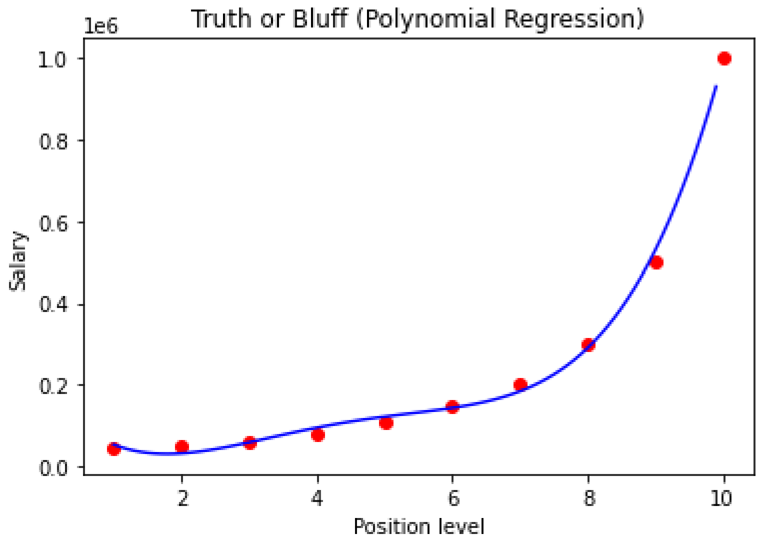
Visualising the Polynomial Regression results

```
plt.scatter(X,y)
plt.plot(X,lin_reg_2.predict(X_poly), color='green')
plt.title('Truth or Bluff (Polynomial Regression)')
plt.xlabel("Position Level")
plt.ylabel("Salary")
plt.show()
```



Visualising the Polynomial Regression results (for higher resolution and smoother curve)

```
X_grid = np.arange(min(X), max(X), 0.1)
X_grid = X_grid.reshape((len(X_grid), 1))
plt.scatter(X, y, color = 'red')
plt.plot(X_grid, lin_reg_2.predict(poly_reg.fit_transform(X_grid)), color = 'blue')
plt.title('Truth or Bluff (Polynomial Regression)')
plt.xlabel('Position level')
plt.ylabel('Salary')
plt.show()
```



Predicting a new result with Linear Regression

```
lin_reg.predict([[6.5]])
```

```
array([330378.78787879])
```

Predicting a new result with Polynomial Regression

```
lin_reg_2.predict(poly_reg.fit_transform([[6.5]]))
```

```
array([158862.45265155])
```