

Polynomial Regression Algorithm

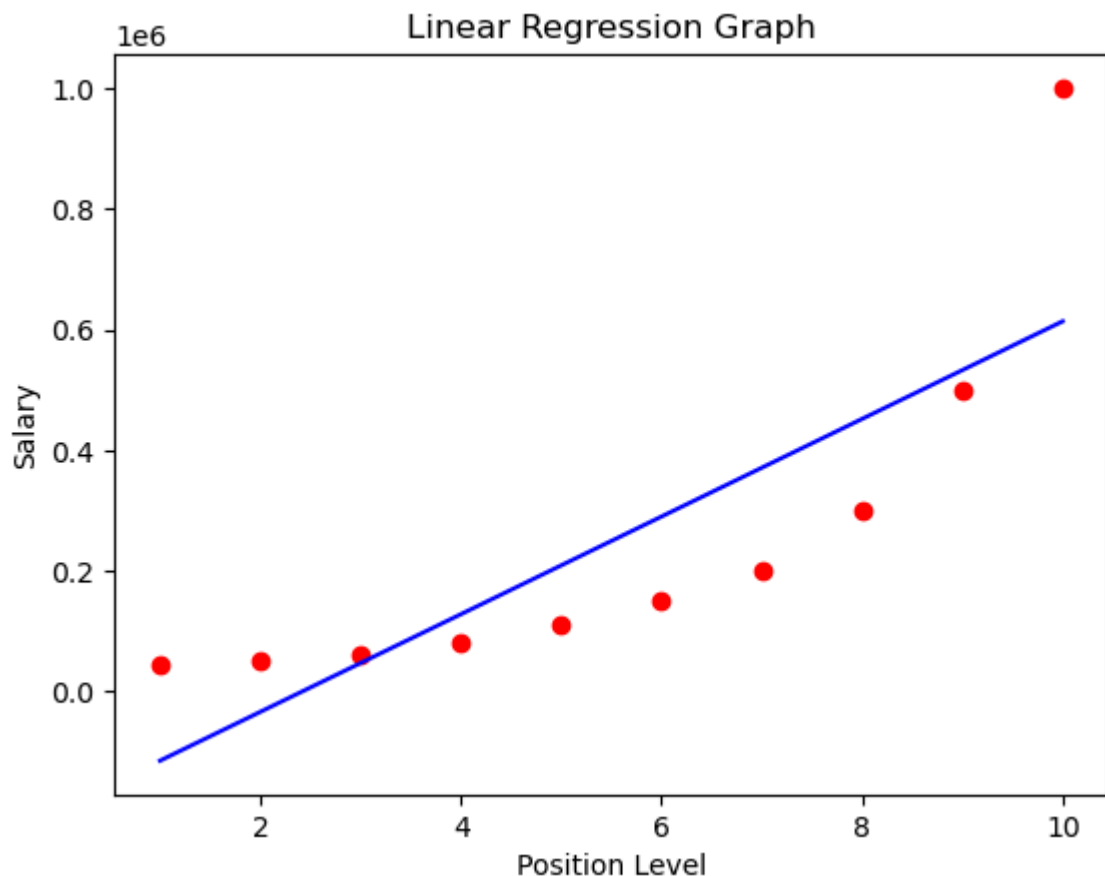
```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression, Ridge, Lasso
from sklearn.metrics import mean_squared_error
from sklearn.metrics import r2_score
from sklearn.datasets import fetch_openml
from sklearn.model_selection import cross_val_score
from sklearn import preprocessing
import pickle
```

```
In [2]: data = pd.read_csv(r"C:\Users\JANHAVI\Downloads\emp_sal.csv")
```

```
In [3]: x = data.iloc[:, 1:2].values
y = data.iloc[:, 2] .values
```

```
In [4]: from sklearn.linear_model import LinearRegression
lin_reg = LinearRegression()
lin_reg.fit(x, y)

plt.scatter(x, y, color = 'red') # Real salary data (training)
plt.plot(x, lin_reg.predict(x), color = 'blue') # Predicted regression line
plt.title('Linear Regression Graph')
plt.xlabel('Position Level')
plt.ylabel('Salary')
plt.show()
```



```
In [5]: lin_model_pred = lin_reg.predict([[6.5]])
print(lin_model_pred)
```

```
[330378.78787879]
```

```
In [6]: from sklearn.preprocessing import PolynomialFeatures
poly_reg = PolynomialFeatures(degree = 5)
x_poly = poly_reg.fit_transform(x)
```

```
In [7]: poly_reg.fit(x_poly,y)

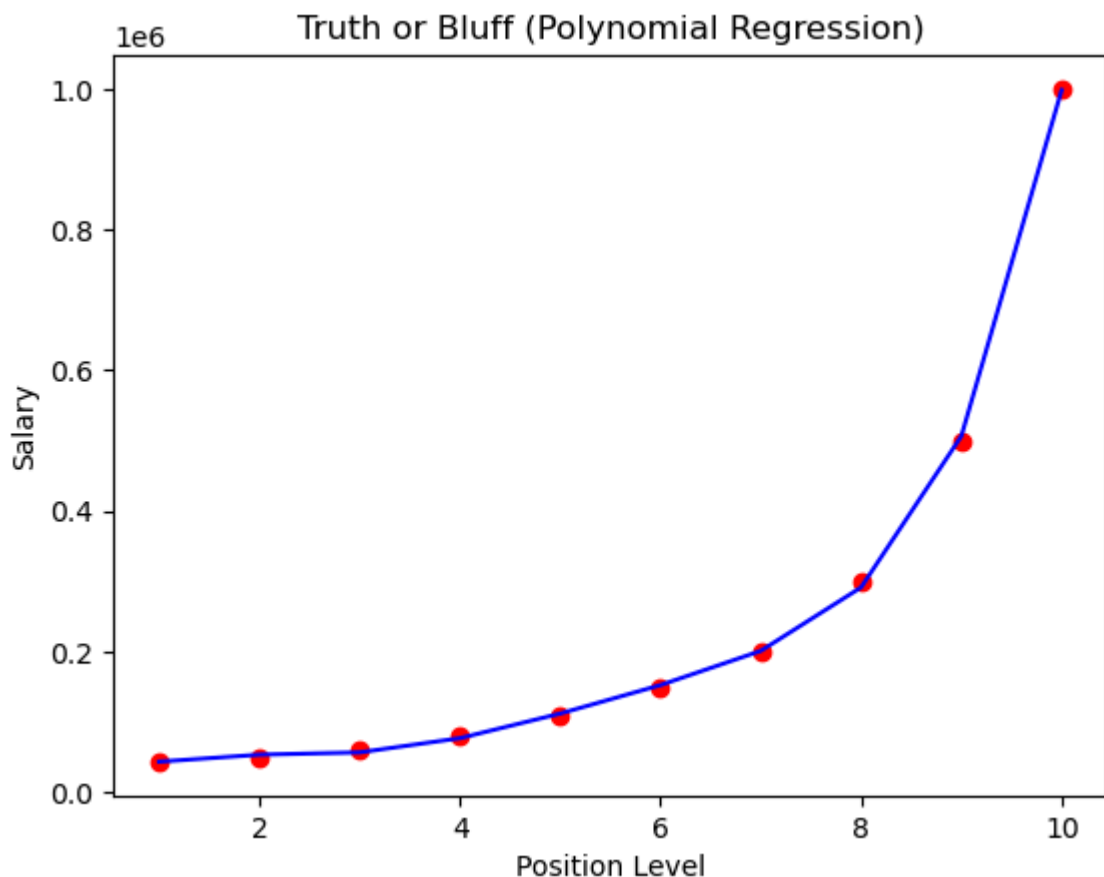
lin_reg_2 = LinearRegression()
lin_reg_2.fit(poly_reg.fit_transform(x), y)
```

```
Out[7]:
```

▼ LinearRegression ⓘ ?

► Parameters

```
In [8]: plt.scatter(x, y, color = 'red') # Real salary data (testing)
plt.plot(x, lin_reg_2.predict(poly_reg.fit_transform(x)), color = 'blue') # Regres
plt.title('Truth or Bluff (Polynomial Regression)')
plt.xlabel('Position Level')
plt.ylabel('Salary')
plt.show()
```



```
In [9]: lin_model_pred = lin_reg.predict([[6.5]])
print(lin_model_pred)
```

```
[330378.78787879]
```

```
In [ ]:
```