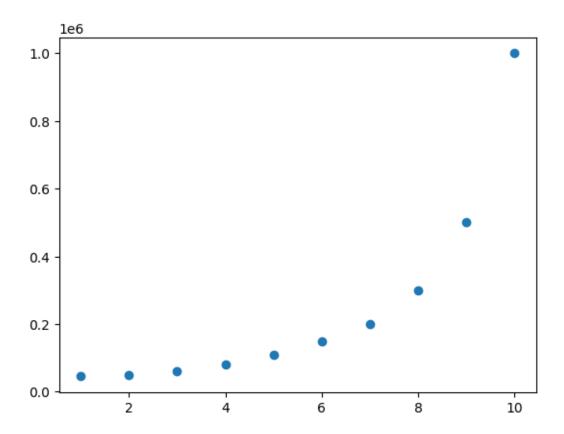# polynomial-regression

November 8, 2024

```python
[2]: import numpy as np
     import matplotlib.pyplot as plt
     import pandas as pd
     from sklearn.metrics import mean_absolute_error,mean_squared_error,r2_score


     import warnings
     warnings.filterwarnings("ignore")
```

```python
[3]: df=pd.read_csv('/emp_sal.csv')
```

```python
[4]: df
```

```
[4]:              Position  Level   Salary
     0  Jr Software Engineer      1    45000
     1  Sr Software Engineer      2    50000
     2             Team Lead      3    60000
     3               Manager      4    80000
     4            Sr manager      5   110000
     5        Region Manager      6   150000
     6                   AVP      7   200000
     7                    VP      8   300000
     8                   CTO      9   500000
     9                   CEO     10  1000000
```

```python
[7]: plt.scatter(X,y)
```

```
[7]: <matplotlib.collections.PathCollection at 0x7c8ab71ccee0>
```

```
[6]: X = df.iloc[:, 1:2].values

     y = df.iloc[:, 2].values
```
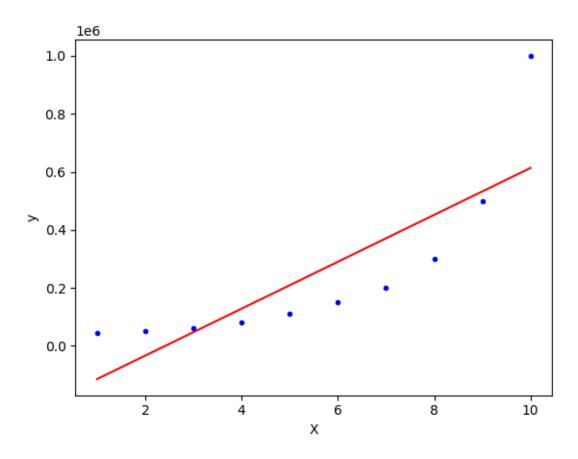
```
[17]: from sklearn.linear_model import LinearRegression
      lin_reg = LinearRegression()
      lin_reg.fit(X, y)
```

```
[17]: LinearRegression()
```

```
[18]: y_pred = lin_reg.predict(X)
```
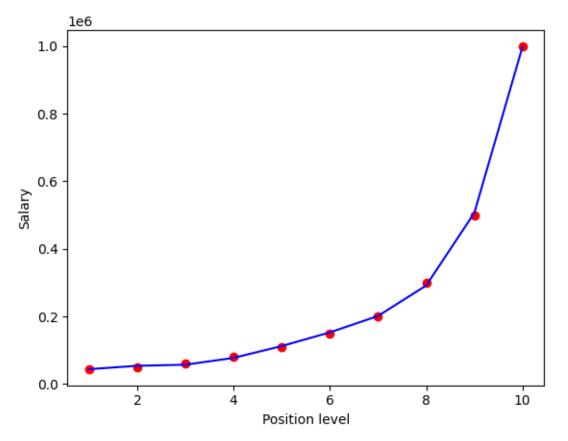
```
[19]: print("r2:",r2_score(y_pred,y))
```

```
r2: 0.5053238120653423
```

```
[20]: plt.plot(X,lin_reg.predict(X),color='r')
      plt.plot(X, y, "b.")
      plt.xlabel("X")
      plt.ylabel("y")
      plt.show()
```

1 Applying Polynomial Linear Regression

```
[10]: from sklearn.preprocessing import PolynomialFeatures,StandardScaler
      poly_reg=PolynomialFeatures(degree=5)
```

```
[11]: X_poly = poly_reg.fit_transform(X)
```

```
[12]: print(X[1])
      print(X_poly[1])
```

```
[2]
[ 1.  2.  4.  8. 16. 32.]
```

```
[21]: lin_reg = LinearRegression()
      lin_reg.fit(X_poly, y)
```

```
[21]: LinearRegression()
```

```
[22]: y_pred = lin_reg.predict(X_poly)
```

```
[23]: print("r2:",r2_score(y_pred,y))
```

r2: 0.9997968614530858

```
[24]: print(y)
      print(y_pred)
```

```
[  45000    50000    60000    80000   110000   150000   200000   300000   500000
 1000000]
[ 43895.10489603   53836.82983718   57065.26806579   77286.71328745
 112263.40326412 152736.59673702 201349.65034962 291571.09557057
 506617.71561662 998377.6223756 ]
```

```
[25]: plt.scatter(X, y, color = 'red')
      plt.plot(X, lin_reg.predict(poly_reg.fit_transform(X)), color = 'blue')
      plt.xlabel('Position level')
      plt.ylabel('Salary')
      plt.show()
```



```
[25]:
```