3dqdc7vch

March 27, 2024

Import Datasets

```python
[1]: import numpy as np
     import pandas as pd
     import matplotlib.pyplot as plt
```

Read Dataset

```python
[2]: dataset=pd.read_csv('/content/Position_Salaries.csv')
```

```python
[3]: dataset
```

```
[3]:           Position  Level   Salary
     0   Business Analyst      1    45000
     1   Junior Consultant      2    50000
     2   Senior Consultant      3    60000
     3            Manager      4    80000
     4    Country Manager      5   110000
     5     Region Manager      6   150000
     6            Partner      7   200000
     7     Senior Partner      8   300000
     8            C-level      9   500000
     9                CEO     10  1000000
```

```python
[4]: x=dataset.iloc[:,1:2].values
```

```python
[5]: y=dataset.iloc[:,2:].values
```

```python
[6]: #check on how the variable look like
     x
```

```
[6]: array([[ 1],
            [ 2],
            [ 3],
            [ 4],
            [ 5],
            [ 6],
            [ 7],
```
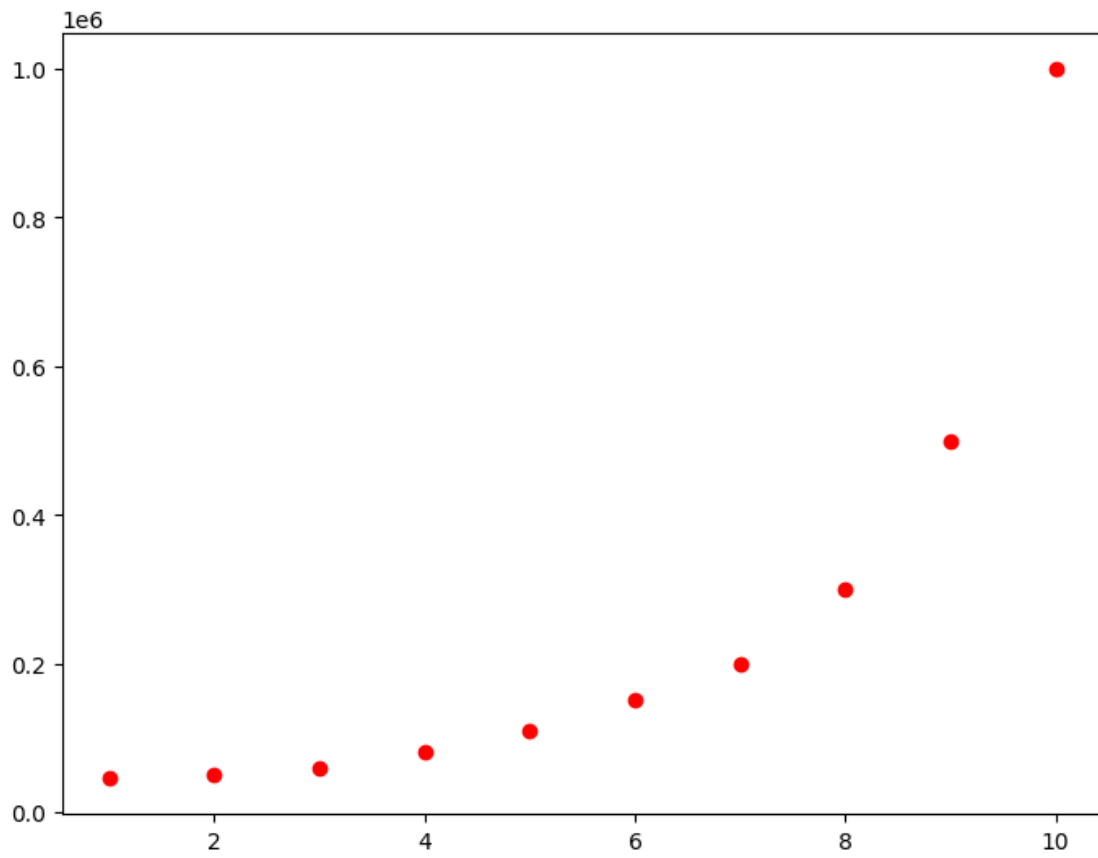
```
       [ 8],
       [ 9],
       [10]])
```

[7]: `y`

```
[7]: array([[  45000],
            [  50000],
            [  60000],
            [  80000],
            [ 110000],
            [ 150000],
            [ 200000],
            [ 300000],
            [ 500000],
            [1000000]])
```

[8]:
```python
#check the pattern of the data and what model you have to apply
fig=plt.figure()
ax=fig.add_axes([0,0,1,1])
ax.scatter(x,y,color='r')
```

[8]: `<matplotlib.collections.PathCollection at 0x7f9649e34a60>`

```
[9]:  from sklearn.preprocessing import PolynomialFeatures
      from sklearn.linear_model import LinearRegression
```
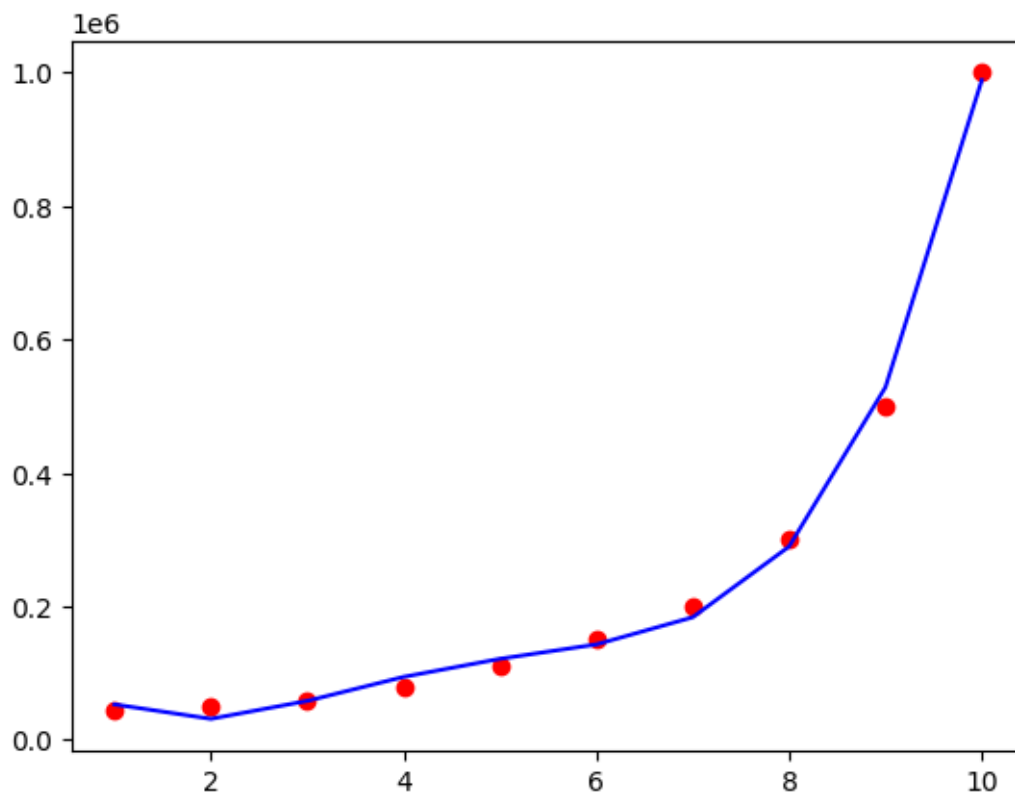
```
[16]: poly=PolynomialFeatures(degree=4)
      x_poly=poly.fit_transform(x)
```

```
[17]: pilreg=LinearRegression()
      pilreg.fit(x_poly,y)
```

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

```
[18]: plt.scatter(x,y,color='red')
      plt.plot(x,pilreg.predict(poly.fit_transform(x)),color='blue')
```

```
[18]: [<matplotlib.lines.Line2D at 0x7f963d794340>]
```

```
[19]: #predicting real value from our dataset
      pilreg.predict(poly.fit_transform([[10]]))
```

```
[19]: array([[988916.08391612]])
```