

SVR - Support Vector Regressor

In [2]: *#importing the libraries*

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

In [3]: *# importing the dataset*

```
dataset=pd.read_csv(r"C:\Users\Jan Saida\Downloads\emp_sal.csv")
dataset
```

Out[3]:

	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

In [4]: `x=dataset.iloc[:, 1:2].values` *#independent variable*
`y=dataset.iloc[:, 2].values` *#dependent variable*

In [5]: x

```
Out[5]: array([[ 1],
               [ 2],
               [ 3],
               [ 4],
               [ 5],
               [ 6],
               [ 7],
               [ 8],
               [ 9],
               [10]], dtype=int64)
```

In [6]: y

```
Out[6]: array([ 45000,  50000,  60000,  80000, 110000, 150000, 200000,
                300000,  500000, 1000000], dtype=int64)
```

```
In [7]: # Fitting SVR to the dataset
        #imported the svr class from SKLEARN.SVM library

        from sklearn.svm import SVR
```

```
In [8]: #creating regressor object & for now understand kernal is use for linear,polynomial or non-linear svr which we will see indept

        regressor=SVR(kernel='poly',degree=5)
        regressor
```

```
Out[8]: ▼          SVR          ⓘ ?
        SVR(degree=5, kernel='poly')
```

```
In [9]: # Fitting the SVR regressor

        regressor.fit(x,y)
```

Out[9]:

SVR
SVR(degree=5, kernel='poly')

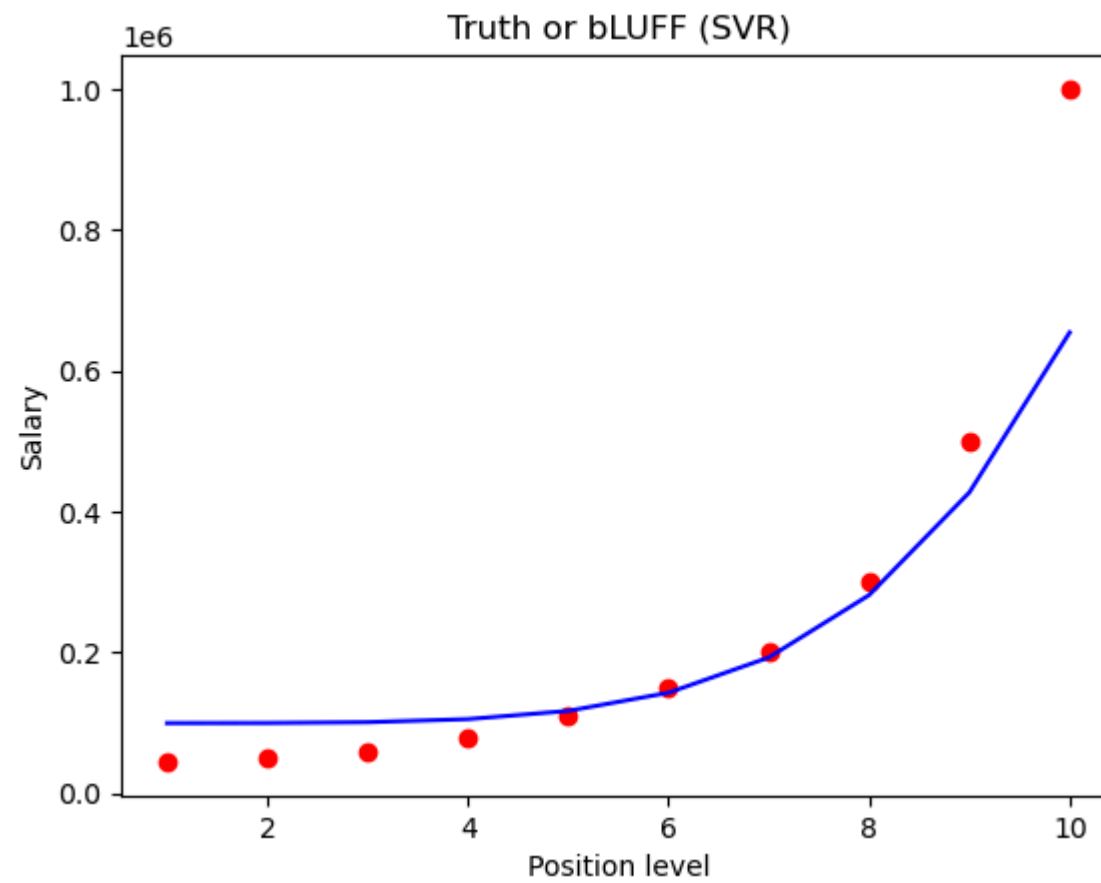
In [10]: *# Predicting a new result*

```
y_pred=regressor.predict([[6.5]])  
y_pred
```

Out[10]: array([164079.01344549])

In [11]: *# Visualizing the SVR results*

```
plt.scatter(x,y,color='red')  
plt.plot(x,regressor.predict(x),color='blue')  
plt.title('Truth or bLUFF (SVR)')  
plt.xlabel('Position level')  
plt.ylabel('Salary')  
plt.show()
```



In []: