Support vector Regression(SVR)

Importing Libraries

In [1]:

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

In [4]:

```
# Importing  database
db=pd.read_csv('position_salaries_dataset.csv')
x=db.iloc[:,1:-1].values
y=db.iloc[:,-1].values
```

In [5]:

```
print(x)
```

```
[[ 1]
 [ 2]
 [ 3]
 [ 4]
 [ 5]
 [ 6]
 [ 7]
 [ 8]
 [ 9]
 [10]]
```

In [6]:

```
print(y)
```

```
[   45000    50000    60000    80000   110000   150000   200000   300000   500000
 1000000]
```

In [7]:

```
#reshpape y
y=  y.reshape(len(y),1)
```

In [8]:

```
print(y)
```

```
[[  45000]
 [  50000]
 [  60000]
 [  80000]
 [ 110000]
 [ 150000]
 [ 200000]
 [ 300000]
 [ 500000]
 [1000000]]
```

Feature Scaling

In [9]:

```
from sklearn.preprocessing import StandardScaler

sc_x = StandardScaler()
sc_y = StandardScaler()
x = sc_x.fit_transform(x)
y = sc_y.fit_transform(y)
```

In [10]:

```
print (x)
```

```
[[-1.5666989 ]
 [-1.21854359]
 [-0.87038828]
 [-0.52223297]
 [-0.17407766]
 [ 0.17407766]
 [ 0.52223297]
 [ 0.87038828]
 [ 1.21854359]
 [ 1.5666989 ]]
```

In [11]:
```
print(y)
```

```
[[-0.72004253]
 [-0.70243757]
 [-0.66722767]
 [-0.59680786]
 [-0.49117815]
 [-0.35033854]
 [-0.17428902]
 [ 0.17781001]
 [ 0.88200808]
 [ 2.64250325]]
```

In [16]:
```
#Traioning the SVR model  on the whole dataset

from sklearn.svm import SVR

regressor = SVR (kernel ='rbf' )
regressor.fit(x, y)
```

```
/usr/local/lib/python3.10/site-packages/sklearn/utils/validation.py:1183: DataConv
ersionWarning: A column-vector y was passed when a 1d array was expected. Please c
hange the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

Out[16]:
```
▾ SVR

SVR()
```

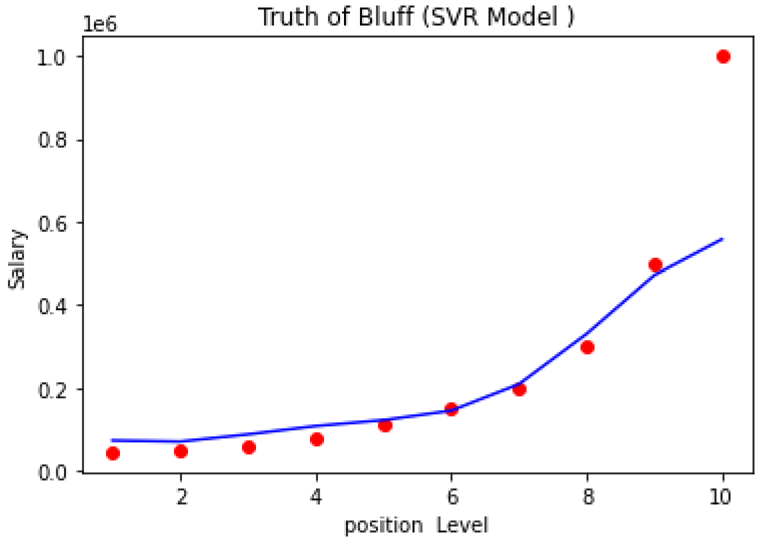
Predicting a new result

In [15]:
```
# Have to predict salary  of candidate having 6.5 years of exp
sc_y.inverse_transform(regressor.predict(sc_x.transform([[6.5]])).reshape(-1,1))
```

Out[15]:
```
array([[170370.0204065]])
```

Visualizing the SVR Result

In [18]:
```
plt.scatter(sc_x.inverse_transform(x), sc_y.inverse_transform (y), color = 'red')
plt.plot(sc_x.inverse_transform(x), sc_y.inverse_transform(regressor.predict(x).res
plt.title('Truth of Bluff (SVR Model )')
plt.xlabel('position  Level')
plt.ylabel('Salary')
plt.show()
```



In [25]:

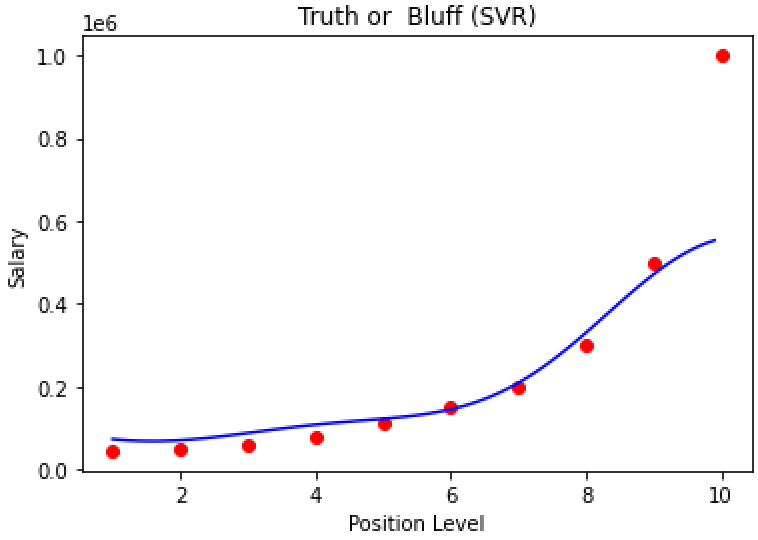
```
# Visualising  the SVR Results (for higher Solution  and  smoother curve)


x_grid = np.arange(min(sc_x.inverse_transform (x)), max(sc_x.inverse_transform(x)),
x_grid = x_grid.reshape((len(x_grid) , 1))
plt.scatter(sc_x.inverse_transform(x), sc_y.inverse_transform(y), color='red')


#Make Predictions on  the x_grid
predictions = regressor.predict(sc_x.transform(x_grid))

#Reshape the prediction array
predictions = predictions.reshape(-1, 1)

plt.plot(x_grid, sc_y.inverse_transform(predictions), color = 'blue' )
plt.title("Truth or  Bluff (SVR)" )
plt.xlabel("Position Level")
plt.ylabel("Salary")
plt.show()
```



In [ ]: