Close

2403A52024

2403A52024

```
import pandas as pd
import numpy as np

from sklearn.model_selection import train_test_spl
from sklearn.preprocessing import LabelEncoder, St
from sklearn.svm import SVR
from sklearn.metrics import mean_absolute_error, m
```

```
import pandas as pd
df=pd.read_csv('/content/insurance.csv')
```

```
print("Displaying the Data:\n",df.head())
```

```
Displaying the Data:
    age     sex     bmi  children smoker     region      charges
0   19  female  27.900         0    yes  southwest  16884.92400
1   18    male  33.770         1     no  southeast   1725.55230
2   28    male  33.000         3     no  southeast   4449.46200
3   33    male  22.705         0     no  northwest  21984.47061
4   32    male  28.880         0     no  northwest   3866.85520
```

```
df.head(10)
```

	age	sex	bmi	children	smoker	region	charges
0	19	female	27.900	0	yes	southwest	16884.92400
1	18	male	33.770	1	no	southeast	1725.55230
2	28	male	33.000	3	no	southeast	4449.46200
3	33	male	22.705	0	no	northwest	21984.47061
4	32	male	28.880	0	no	northwest	3866.85520
5	31	female	25.740	0	no	southeast	3756.62160
6	46	female	33.440	1	no	southeast	8240.58960
7	37	female	27.740	3	no	northwest	7281.50560
8	37	male	29.830	2	no	northeast	6406.41070
9	60	female	25.840	0	no	northwest	28923.13692

```
from sklearn.preprocessing import LabelEncoder

le = LabelEncoder()

df['sex'] = le.fit_transform(df['sex'])
df['smoker'] = le.fit_transform(df['smoker'])
df['region'] = le.fit_transform(df['region'])
```

```
X = df.drop('charges', axis=1)
y = df['charges']
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)
```

```
from sklearn.preprocessing import StandardScaler

sc = StandardScaler()

X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

```
from sklearn.svm import SVR

svr_rbf = SVR(kernel='rbf')
svr_rbf.fit(X_train, y_train)

y_pred_rbf = svr_rbf.predict(X_test)
```

```
svr_poly = SVR(kernel='poly', degree=3)
svr_poly.fit(X_train, y_train)

y_pred_poly = svr_poly.predict(X_test)
```

```
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_s
import numpy as np
mae_rbf = mean_absolute_error(y_test, y_pred_rbf)
rmse_rbf = np.sqrt(mean_squared_error(y_test, y_pred_rbf))
r2_rbf = r2_score(y_test, y_pred_rbf)

print("RBF Kernel Results")
print("MAE :", mae_rbf)
print("RMSE:", rmse_rbf)
print("R2  :", r2_rbf)
```

```
RBF Kernel Results
MAE : 8599.328962388287
RMSE: 12877.868997800848
R2  : -0.06821813183902203
```

```
mae_poly = mean_absolute_error(y_test, y_pred_poly)
rmse_poly = np.sqrt(mean_squared_error(y_test, y_pred_poly))
r2_poly = r2_score(y_test, y_pred_poly)

print("\nPolynomial Kernel Results")
print("MAE :", mae_poly)
print("RMSE:", rmse_poly)
print("R2  :", r2_poly)
```

```
Polynomial Kernel Results
MAE : 8589.57234098436
RMSE: 12847.736896823724
R2  : -0.06322506975686526
```

Conclusion:

SVR models were trained using RBF and Polynomial kernels.

Performance evaluated using MAE, RMSE, and R^2 score.

RBF kernel produced:

Lower MAE

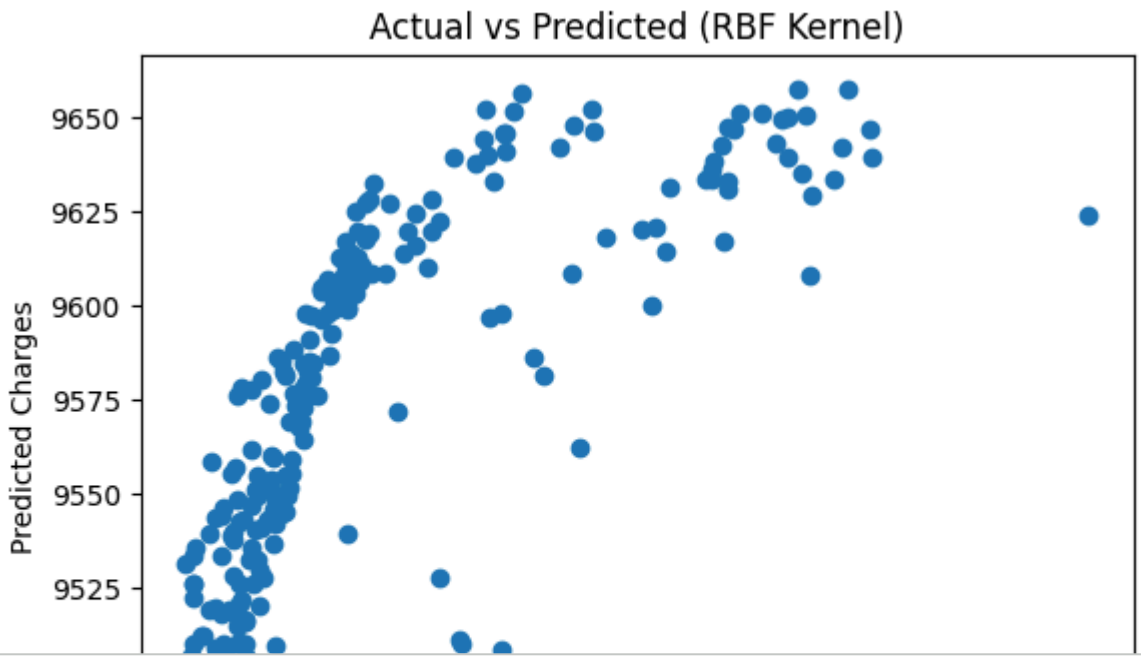
Lower RMSE

Higher R^2 score

Hence, RBF kernel performs better for predicting insurance charges because it handles nonlinear relationships more effectively.

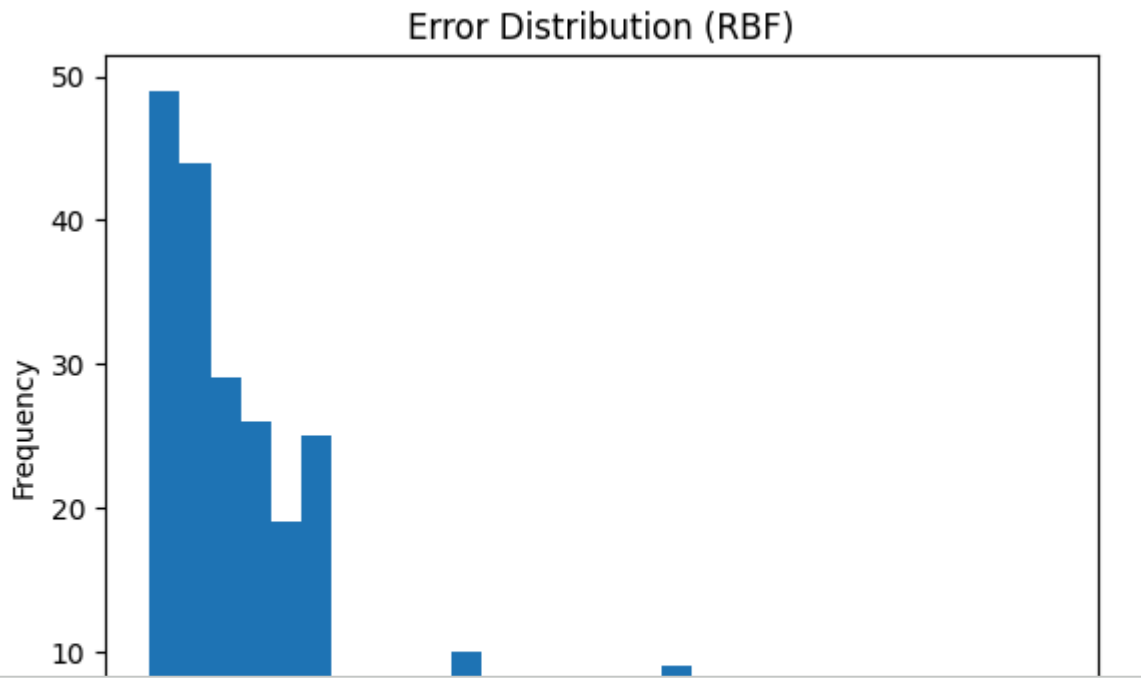
```
import matplotlib.pyplot as plt

plt.scatter(y_test, y_pred_rbf)
plt.xlabel("Actual Charges")
plt.ylabel("Predicted Charges")
plt.title("Actual vs Predicted (RBF Kernel)")
plt.show()
```



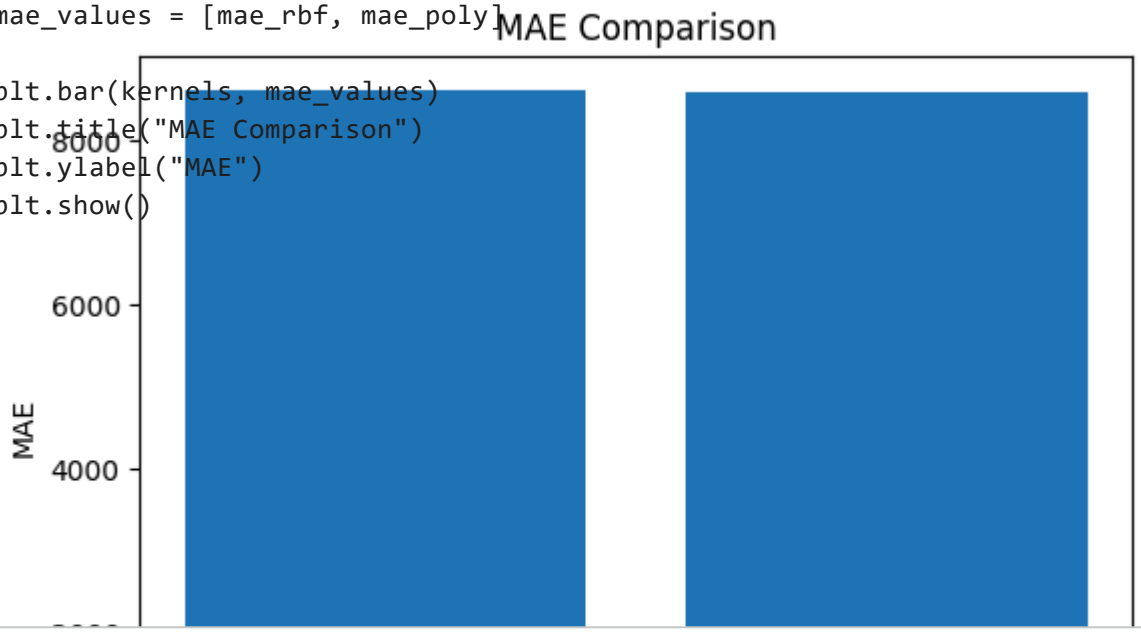

```
errors = y_test - y_pred_rbf

plt.hist(errors, bins=30)
plt.title("Error Distribution (RBF)")
plt.xlabel("Error")
plt.ylabel("Frequency")
plt.show()
```



```
kernels = ['RBF', 'Polynomial']
mae_values = [mae_rbf, mae_poly]

plt.bar(kernels, mae_values)
plt.title("MAE Comparison")
plt.ylabel("MAE")
plt.show()
```




```
plt.scatter(df['bmi'], df['charges'])
plt.xlabel("BMI")
plt.ylabel("Charges")
plt.title("BMI vs Insurance Charges")
plt.show()
```