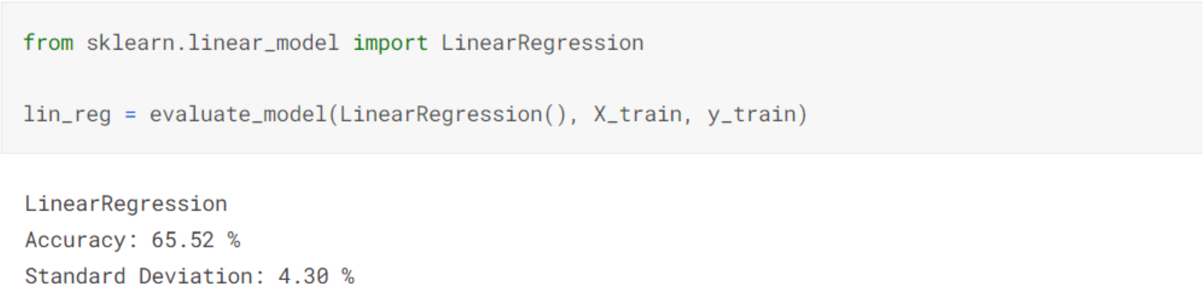
Random Forest Provides a much more accurate result as compared to other machine Learning models

It has the below mentioned advantages:

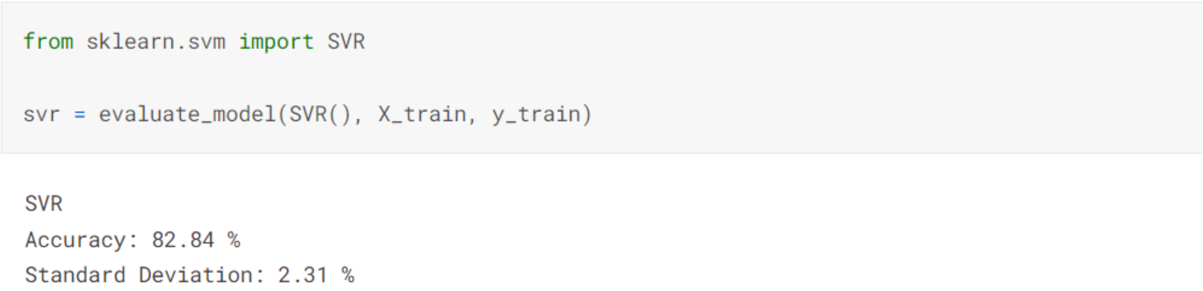
1. It reduces overfitting in decision trees and helps to improve the accuracy
2. It is flexible to both classification and regression problems
3. It works well with both categorical and continuous values
4. It automates missing values present in the data
5. Normalising of data is not required as it uses a rule-based approach



```
from sklearn.linear_model import LinearRegression

lin_reg = evaluate_model(LinearRegression(), X_train, y_train)
```

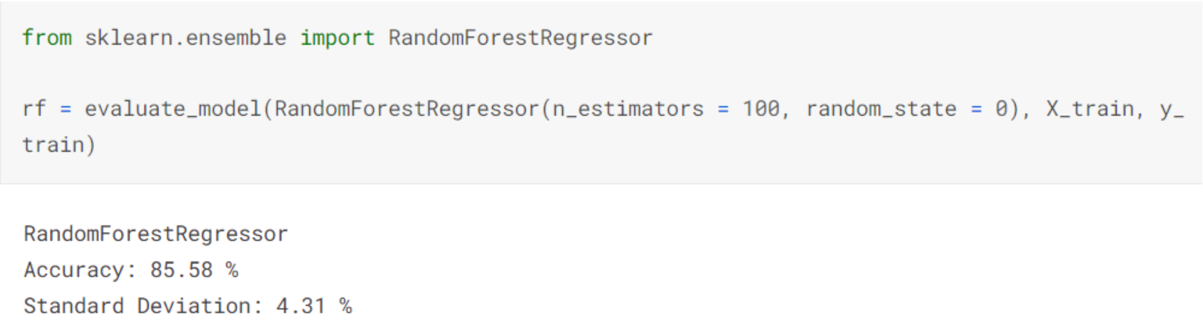
```
LinearRegression
Accuracy: 65.52 %
Standard Deviation: 4.30 %
```



```
from sklearn.svm import SVR

svr = evaluate_model(SVR(), X_train, y_train)
```

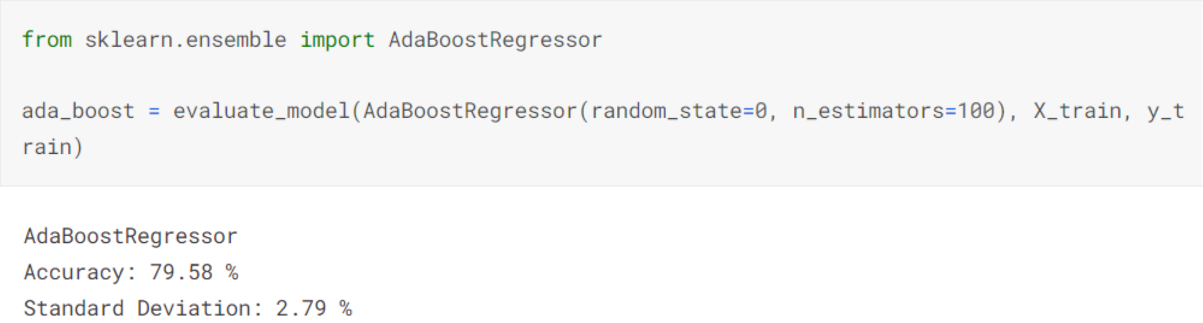
```
SVR
Accuracy: 82.84 %
Standard Deviation: 2.31 %
```



```
from sklearn.ensemble import RandomForestRegressor

rf = evaluate_model(RandomForestRegressor(n_estimators = 100, random_state = 0), X_train, y_train)
```

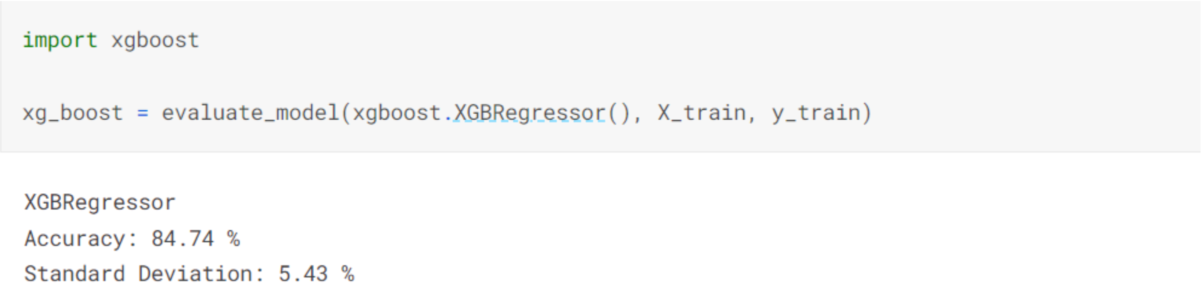
```
RandomForestRegressor
Accuracy: 85.58 %
Standard Deviation: 4.31 %
```



```
from sklearn.ensemble import AdaBoostRegressor

ada_boost = evaluate_model(AdaBoostRegressor(random_state=0, n_estimators=100), X_train, y_train)
```

```
AdaBoostRegressor
Accuracy: 79.58 %
Standard Deviation: 2.79 %
```



```
import xgboost

xg_boost = evaluate_model(xgboost.XGBRegressor(), X_train, y_train)
```

```
XGBRegressor
Accuracy: 84.74 %
Standard Deviation: 5.43 %
```

When compared to other models it has yielded has much accurate result of 86% and above and on the other hand Linear regression model yields 66%~ accuracy,SVR yields 83%~ accuracy,AdaBoostRegressor yields 80%,XGBRegressor yields 85%