

▼ Random Forest Regression

▼ Importing the libraries

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

▼ Importing the dataset

```
dataset = pd.read_csv('/content/Data.csv')
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, -1].values
```

▼ Splitting the dataset into the Training set and Test set

```
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 = 0)
```

▼ Training the Random Forest Regression model on the whole dataset

```
from sklearn.ensemble import RandomForestRegressor
regressor = RandomForestRegressor(n_estimators = 10, random_state = 0)
regressor.fit(X_train, y_train)

☐ RandomForestRegressor(bootstrap=True, ccp_alpha=0.0, criterion='mse',
                          max_depth=None, max_features='auto', max_leaf_nodes=None,
                          max_samples=None, min_impurity_decrease=0.0,
                          min_impurity_split=None, min_samples_leaf=1,
                          min_samples_split=2, min_weight_fraction_leaf=0.0,
                          n_estimators=10, n_jobs=None, oob_score=False,
                          random_state=0, verbose=0, warm_start=False)
```

▼ Predicting the Test set results

```
y_pred = regressor.predict(X_test)
np.set_printoptions(precision=2)
print(np.concatenate((y_pred.reshape(len(y_pred),1), y_test.reshape(len(y_test),1)),1))
```

```
↳ [[433.78 431.23]
    [457.99 460.01]
    [463.14 461.14]
    ...
    [470.16 473.26]
    [439.51 438.  ]
    [460.32 463.28]]
```

▼ Evaluating the Model Performance

```
from sklearn.metrics import r2_score
r2_score(y_test, y_pred)
```

```
↳ 0.9615980699813017
```