**Before Feature Engineering**

**KNN:**

* **Test** set accuracy: 0.55
* **Hyper Parameters:** n\_neighbors=10 , weights='uniform'
* **Summary**
  + I tested by changing 'n\_neighbours', and apparently above 10 gave the highest test accuracy

**Decision Tree:**

* Accuracy on **training** set: 0.991
* Accuracy on **test** set: 0.704
* **Hyper Parameters:** max\_depth=10 , max\_features=10
* **Summary**
  + By changing 'max\_deph' to 10 and 'max\_features' to 10(Almost Sqaure Root of total features), I got the best I could from changing parameters

**Random Forest:**

* Accuracy on **training** set: 1.000
* Accuracy on **test** set: 0.873
* **Hyper Parameters:** n\_estimators=80 , max\_features=16
* **Summary**
  + Setting n\_estimators to 80 and max\_features to 16 100 gave a good accuracy near 90%

**Naive Bayes:**

* Accuracy on **training** set: 0.848
* Accuracy on **test** set: 0.803
* **Hyper Parameters:** No Parameters
* **No Summary**

**SVM:**

* Accuracy on **training** set: 1.000
* Accuracy on **test** set: 0.549
* **Hyper Parameters:** C = 1, gamma = 1
* **Summary**
  + Test Accuracy didn’t change no matter what parameter

**After Feature Engineering**

**Univariate statistics:  
Percentile = 10**

KNN:

Test set accuracy: 0.63 Before: 0.55

Decision Tree:

Accuracy on training set: 1.000 Before: 0.991

Accuracy on test set: 0.986 Before: 0.704

Random Forest:

Accuracy on training set: 1.000 Before: 1.000

Accuracy on test set: 0.972 Before: 0.873

Naive Bayes:

Accuracy on training set: 0.886 Before: 0.848

Accuracy on test set: 0.845 Before: 0.803

SVM:

Accuracy on training set: 1.000 Before: 1.000

Accuracy on test set: 0.915 Before: 0.549

Decision Tree Regressor:

Accuracy on training set: 0.998 Before: 1.000

Accuracy on test set: 0.998 Before: 0.877

Random Forest Regressor:

Accuracy on training set: 0.990 Before: 0.980

Accuracy on test set: 0.982 Before: 0.921

**Model Based Selection:**

**Random Forest Classifier used as model with “**n\_estimators=10”

KNN:

Test set accuracy: 0.55

Decision Tree:

Accuracy on training set: 1.000

Accuracy on test set: 0.901

Random Forest:

Accuracy on training set: 1.000

Accuracy on test set: 0.803

Naive Bayes:

Accuracy on training set: 0.915

Accuracy on test set: 0.803

SVM:

Accuracy on training set: 1.000

Accuracy on test set: 0.549

Decision Tree Regressor:

Accuracy on training set: 1.000

Accuracy on test set: 0.886

Random Forest Regressor:

Accuracy on training set: 0.980

Accuracy on test set: 0.911

**Recursive Feature Elimination (Iterative Feature Selection):**

**Random Forest Classifier used with “**n\_estimators=100**” and “**n\_features\_to\_select=16**”**

KNN:

Test set accuracy: 0.56

Decision Tree:

Accuracy on training set: 1.000

Accuracy on test set: 0.930

Random Forest:

Accuracy on training set: 1.000

Accuracy on test set: 0.930

Naive Bayes:

Accuracy on training set: 0.910

Accuracy on test set: 0.873

SVM:

Accuracy on training set: 1.000

Accuracy on test set: 0.549

Decision Tree Regressor:

Accuracy on training set: 1.000

Accuracy on test set: 0.962

Random Forest Regressor:

Accuracy on training set: 0.985

Accuracy on test set: 0.949

**Conclusion:**

By Observing the training and testing accuracies during all the above mentioned procedures

(Data also available in the excel file), the best one is **Decision Tree Regressor after Feature Selection via Univariate Selection with Percentile = 10.**

Training Accuracy of 99.8%

Testing accuracy of 99.8%