Forest cover type predication

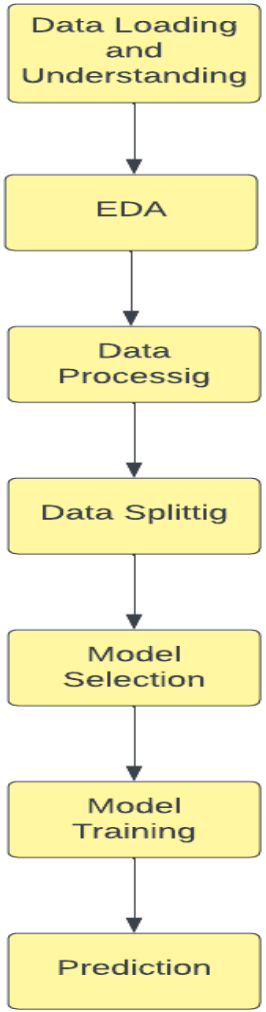
Hello!

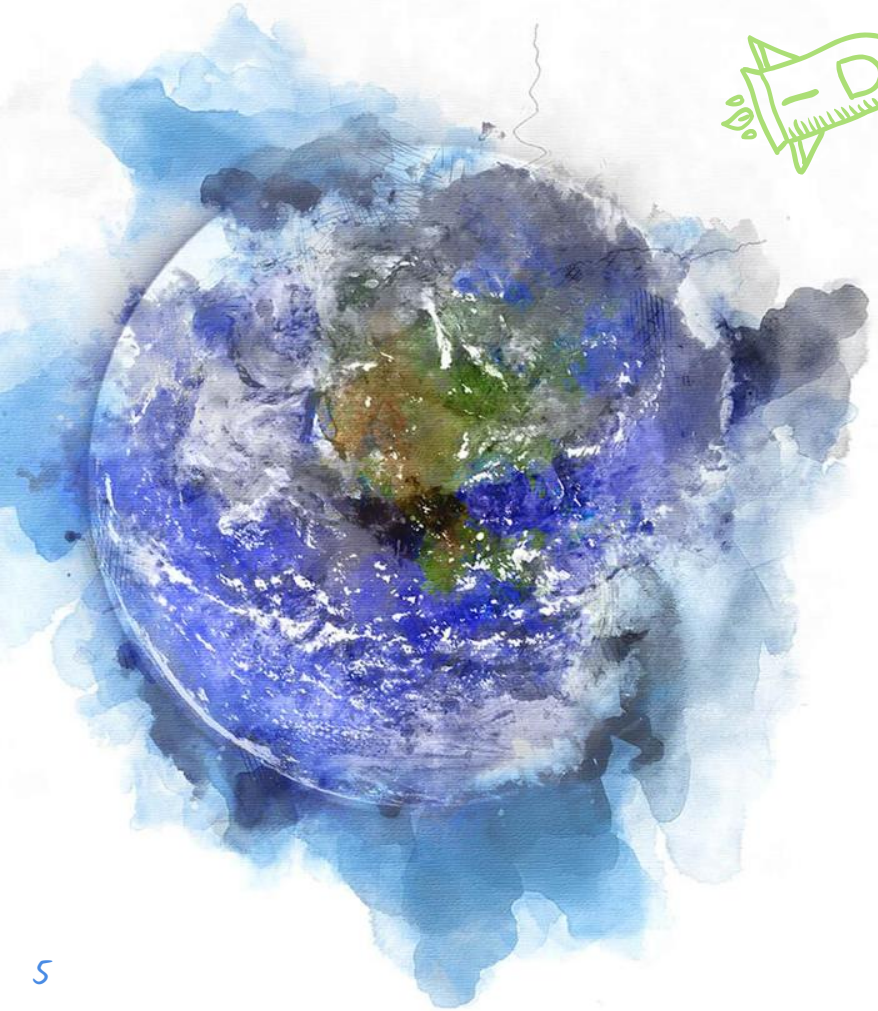
Group 16

Dina Ibrahim Mohammady
300389383
Nada Mohammed Zakaria
300389901
Sema Abdelnasser Helali
300389914

Problem Overview

The main objective is to predict the dominant forest cover type in 30 x 30 meter cells of the Roosevelt National Forest using cartographic variables, without employing remotely sensed data. The forest cover types in the four wilderness areas are primarily shaped by ecological processes due to minimal human-caused disturbances, emphasizing the significance of understanding the ecological relationships between cartographic features and forest cover types in these undisturbed wilderness areas.

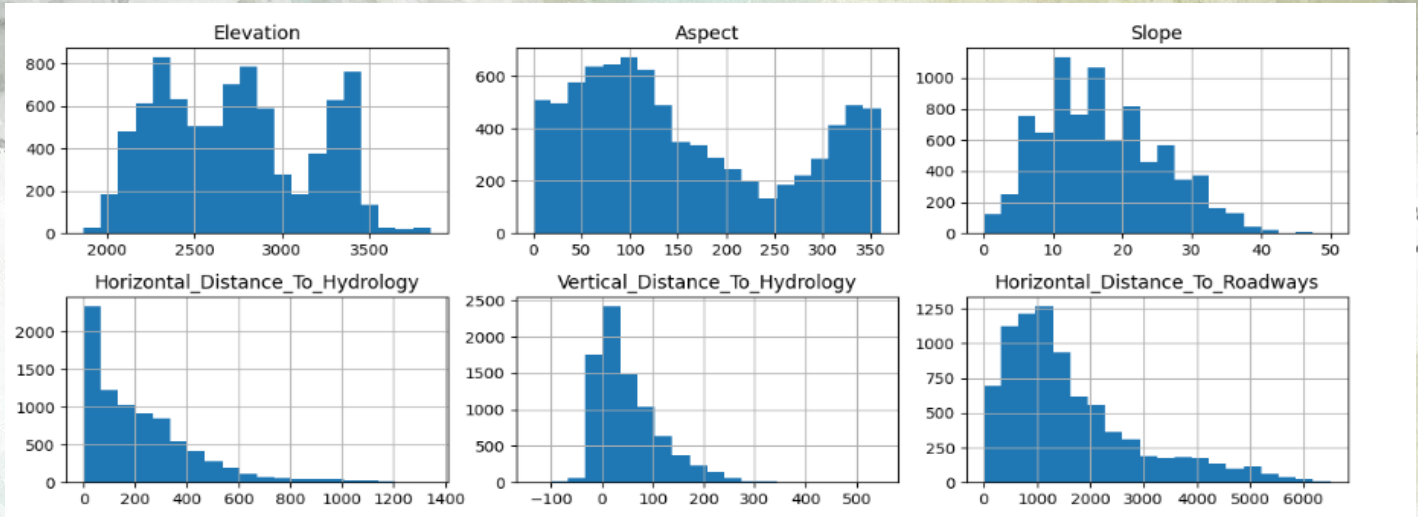


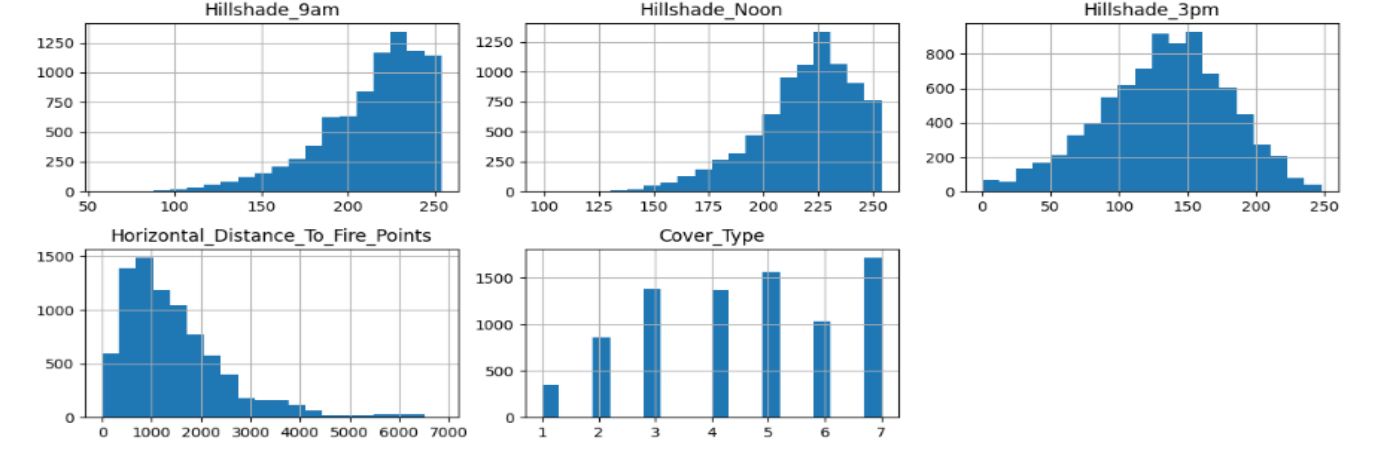


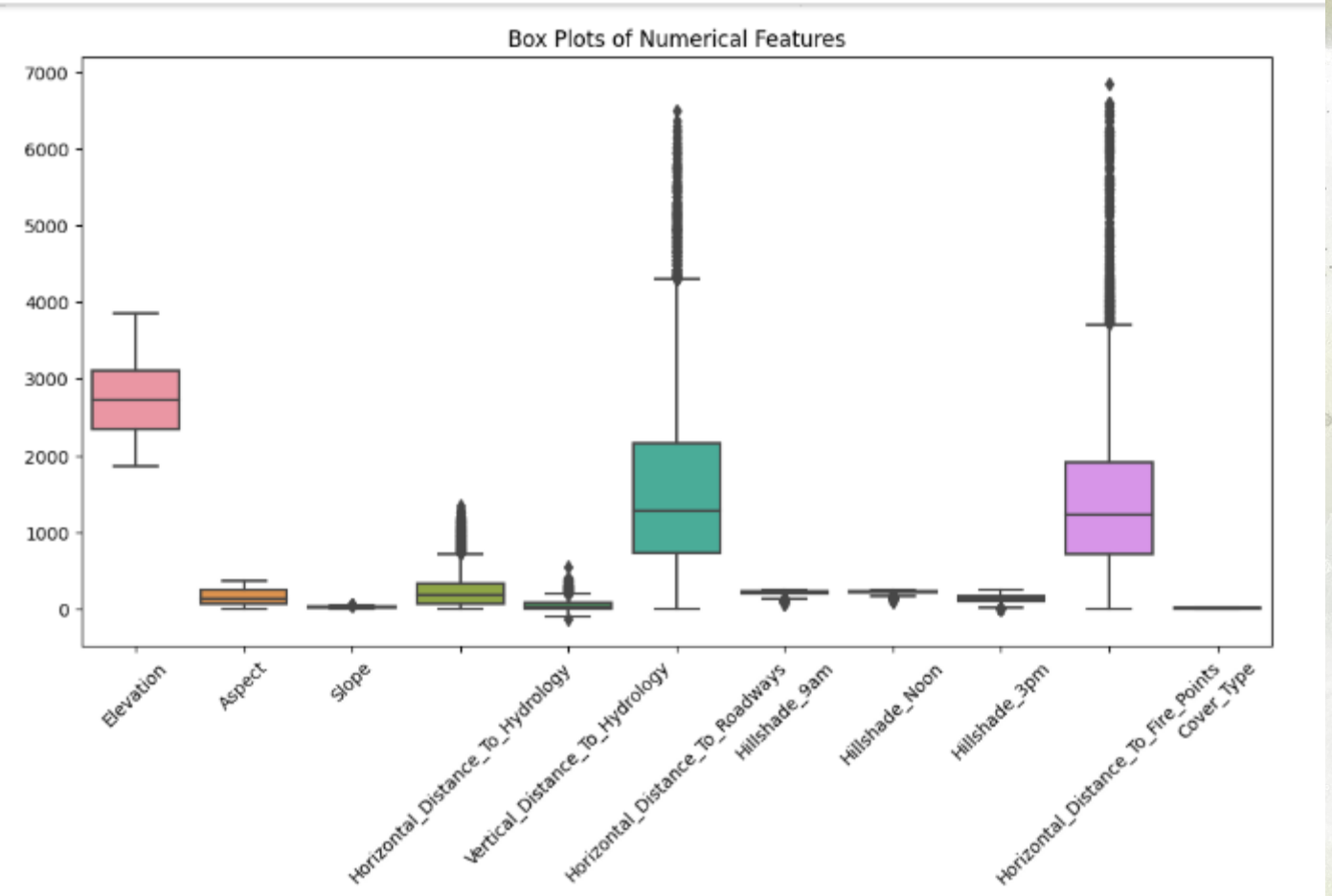
Exploratory Data Analysis

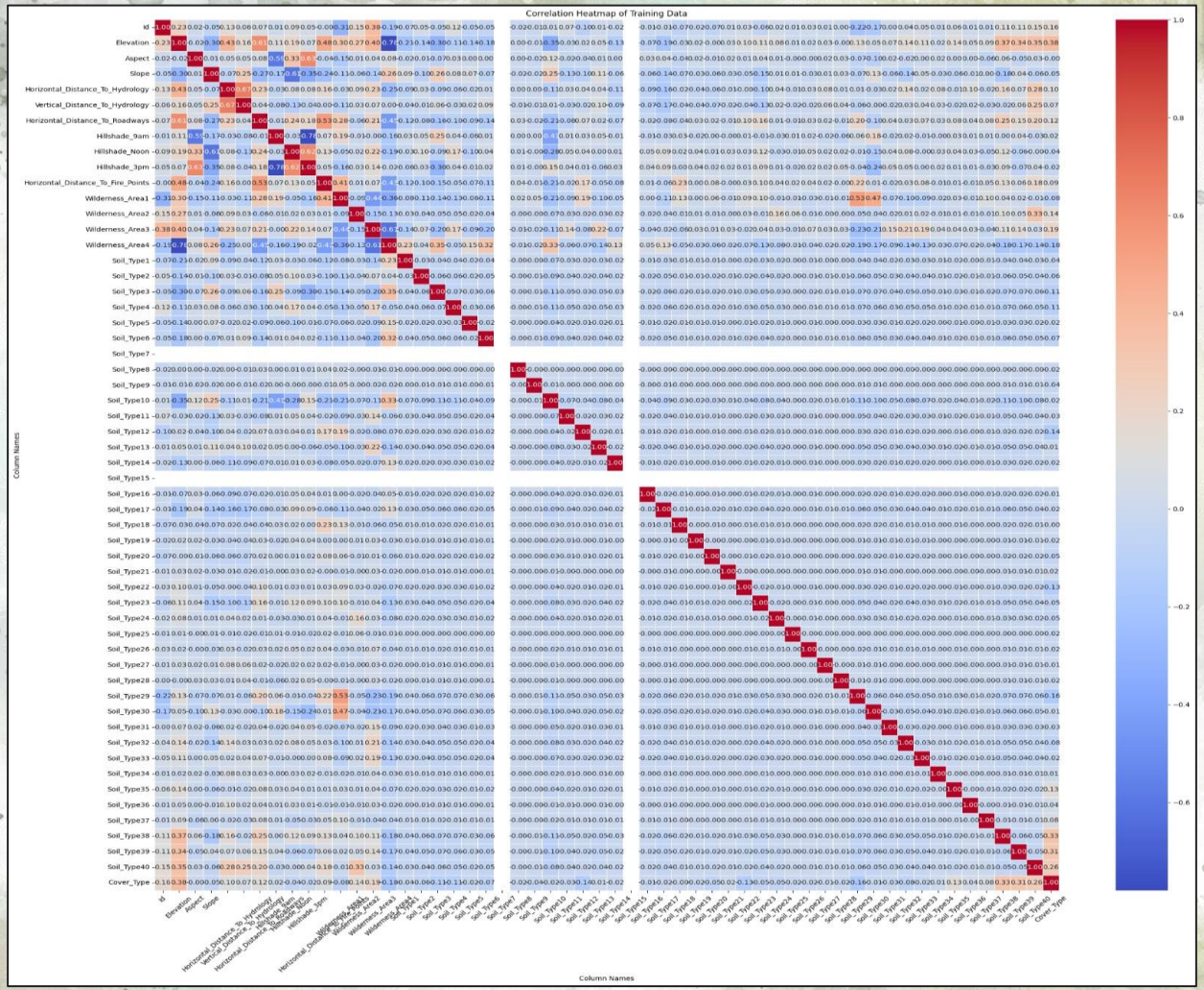
Dataset statistics

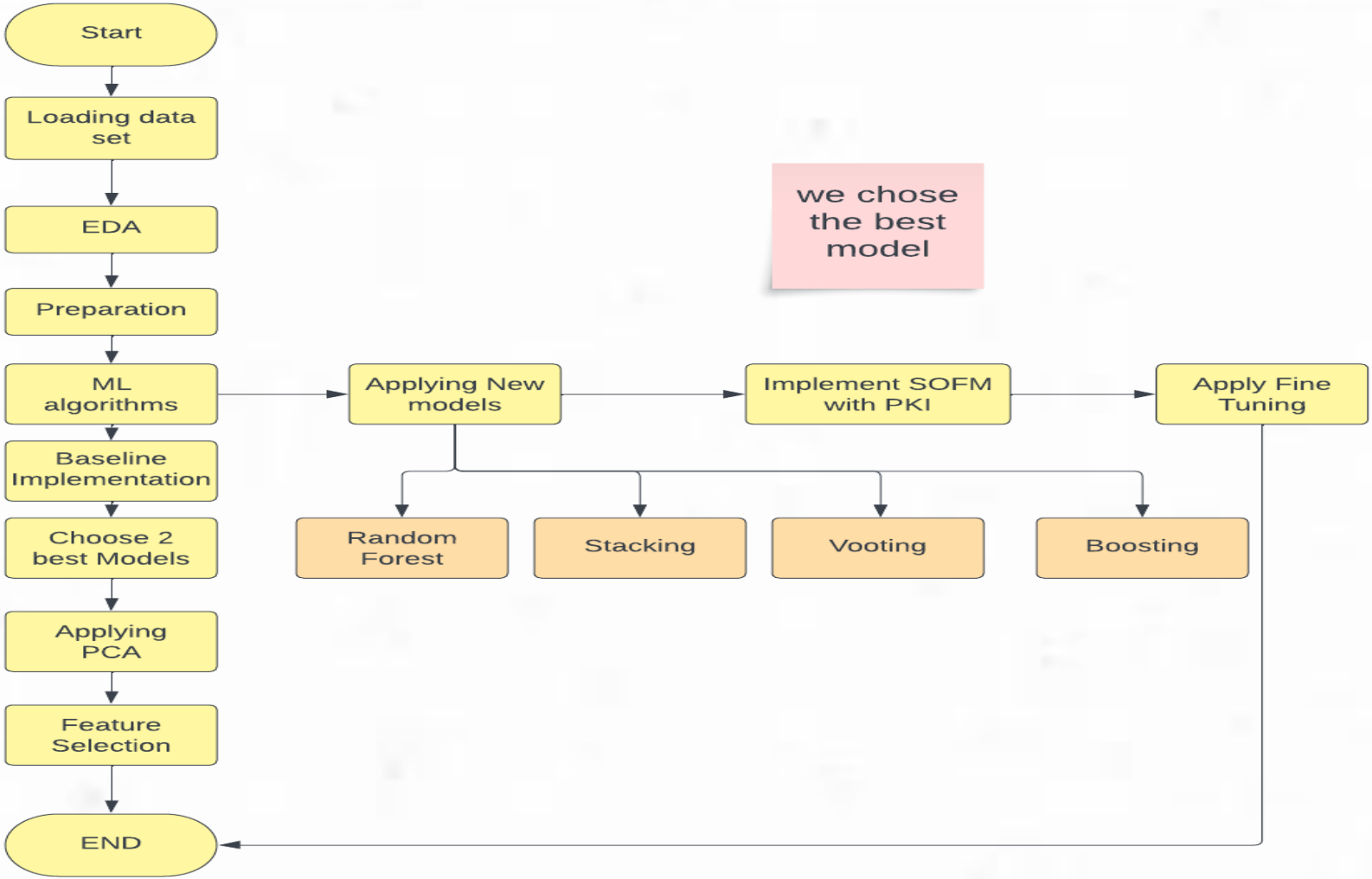
Number of variables	56
Number of observations	8286
Missing cells	0
Missing cells (%)	0.0%
Duplicate rows	0
Duplicate rows (%)	0.0%
Total size in memory	3.5 MiB
Average record size in memory	448.0 B

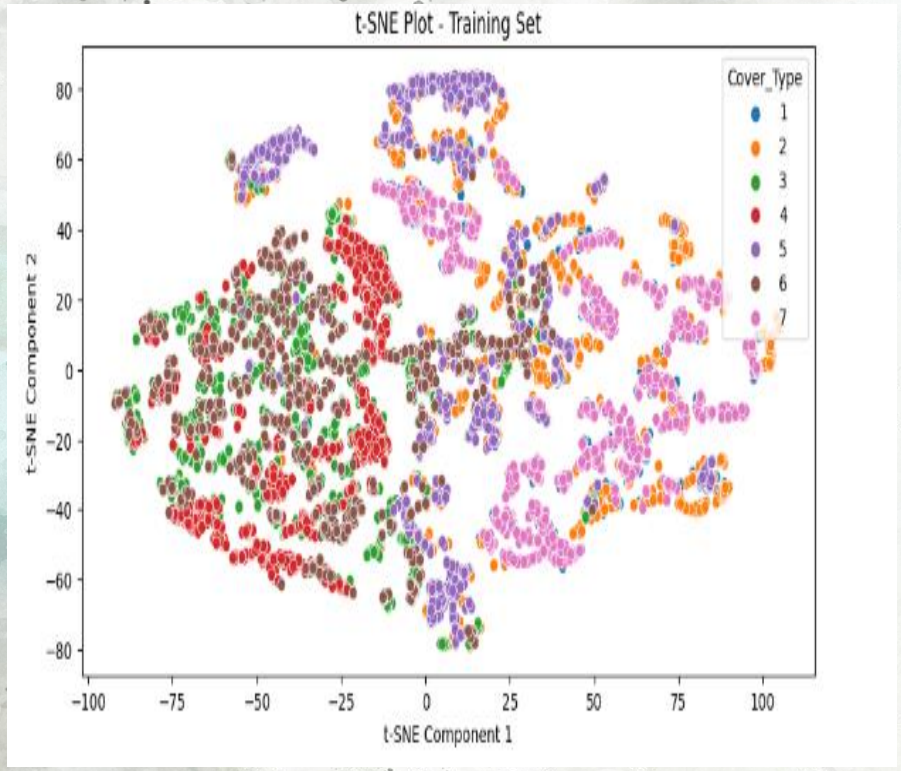

Elevation
Aspect
Slope
Horizontal_Distance_To_Hydrology
Vertical_Distance_To_Hydrology
Horizontal_Distance_To_Roadways

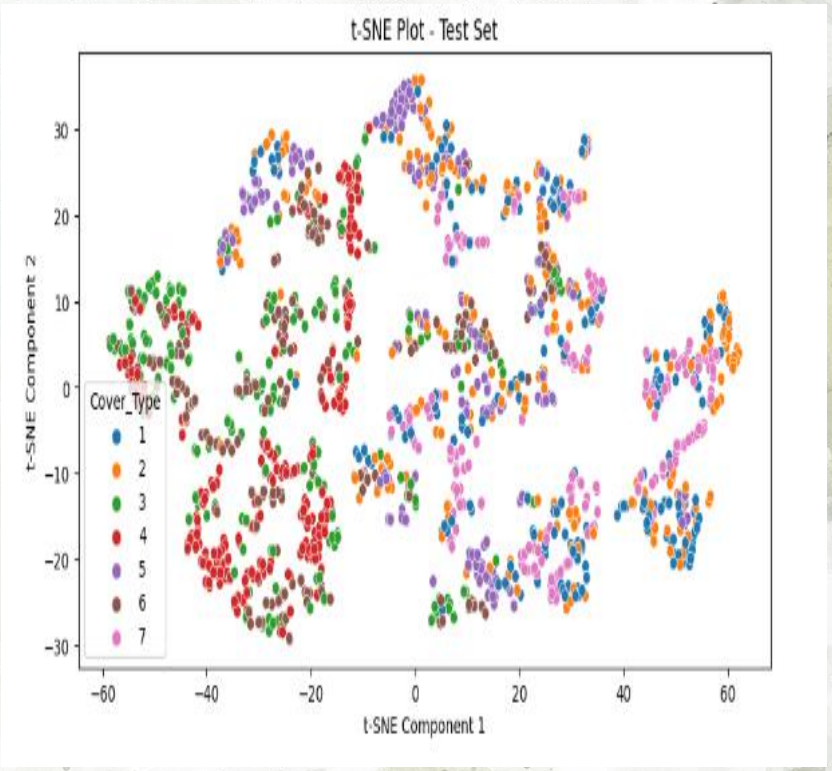

Hillshade_9am
Hillshade_Noon
Hillshade_3pm
Horizontal_Distance_To_Fire_Points
Cover_Type


Box Plots of Numerical Features
7000
6000
5000
4000
3000
2000
1000
0
Elevation
Aspect
Slope
Horizontal_Distance_To_Hydrology
Vertical_Distance_To_Hydrology
Horizontal_Distance_To_Roadways
Hillshade_9am
Hillshade_Noon
Hillshade_3pm
Horizontal_Distance_To_Fire_Points
Cover_Type


Correlation Heatmap of Training Data
Column Names
Column Names
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6

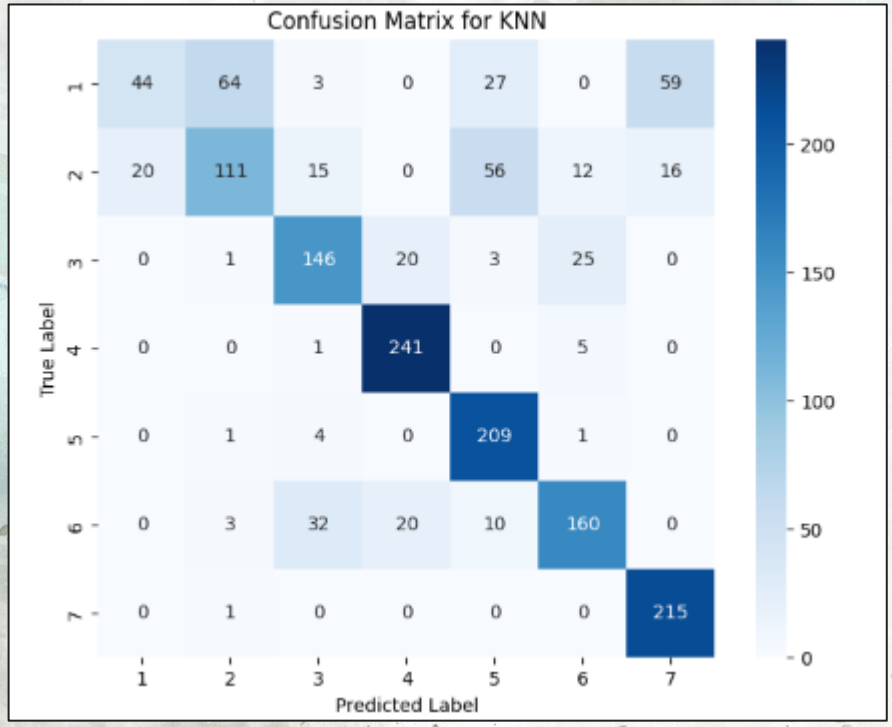

Start
Loading data set
EDA
Preparation
ML algorithms
Baseline Implementation
Choose 2 best Models
Applying PCA
Feature Selection
END
Applying New models
Random Forest
Stacking
Vooting
Boosting
Implement SOFM with PKI
Apply Fine Tuning
we chose the best model


t-SNE Plot - Training Set
Cover_Type
1
2
3
4
5
6
7
t-SNE Component 2
t-SNE Component 1
80
60
40
20
0
−20
−40
−60
−80
−100
−75
−50
−25
0
25
50
75
100


t-SNE Plot - Test Set
Cover_Type
1
2
3
4
5
6
7
t-SNE Component 2
t-SNE Component 1
30
20
10
0
−10
−20
−30
−60
−40
−20
0
20
40
60

Baseline Performance

For KNN

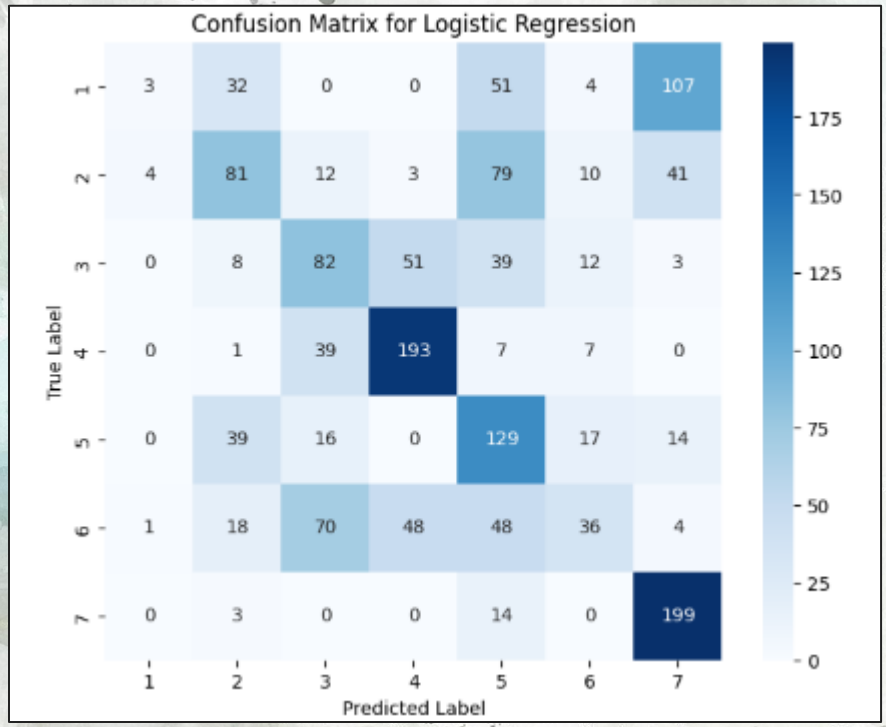


Training Accuracy for KNN: 0.8853

Testing Accuracy for KNN: 0.7384

Validation Accuracy for KNN: 0.7523

For Logistic Regression

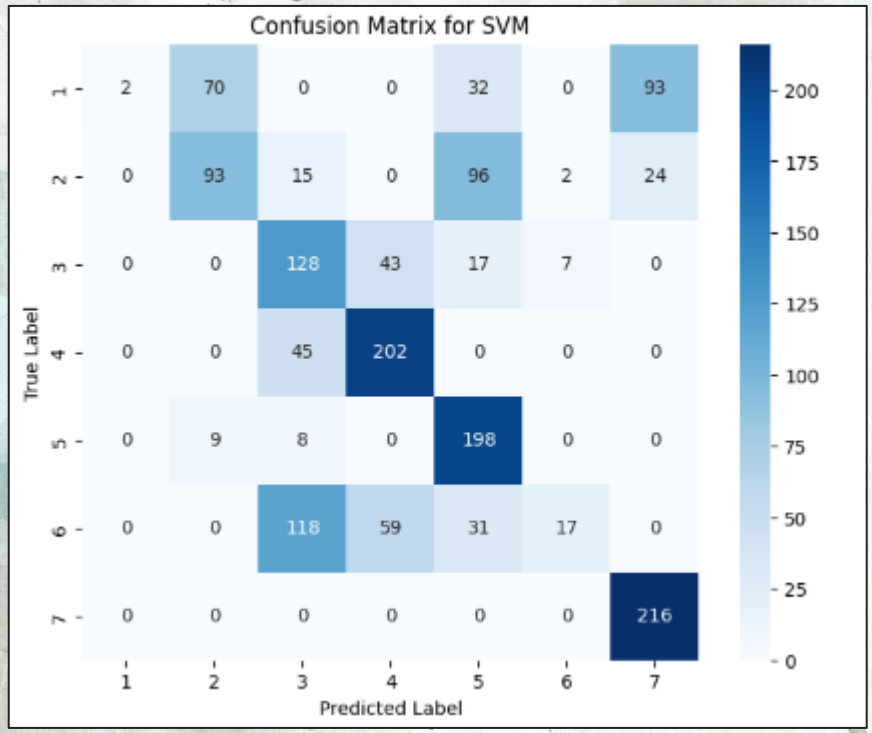


Training Accuracy for Logistic Regression: 0.5782

Testing Accuracy for Logistic Regression: 0.4741

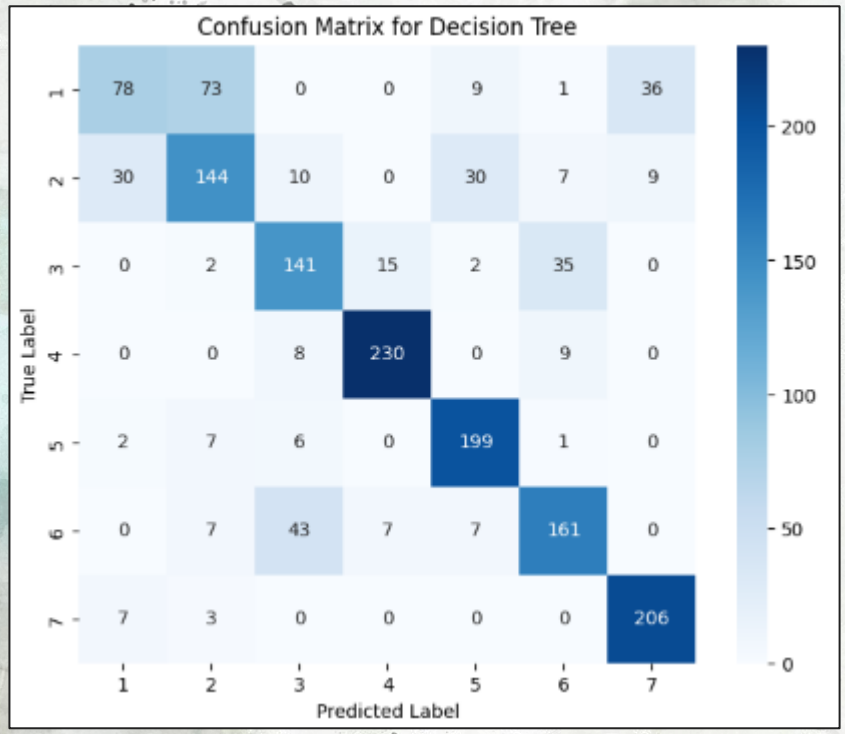
Validation Accuracy for Logistic Regression: 0.4773

For SVM



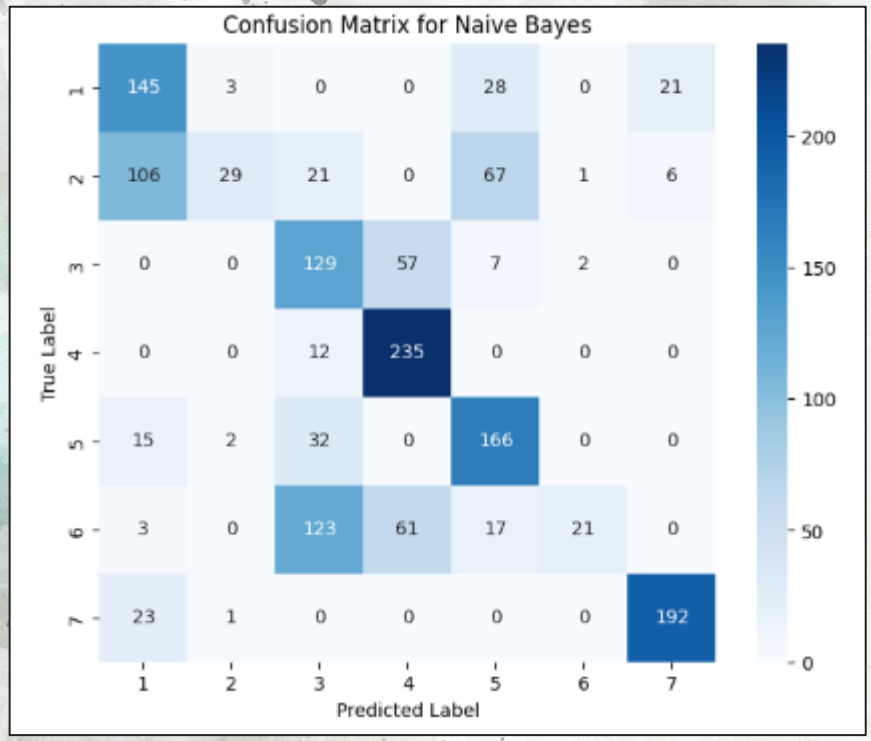

```
Training Accuracy for SVM: 0.6859
Testing Accuracy for SVM: 0.5613
Validation Accuracy for SVM: 0.5501
```

For Decision Tree



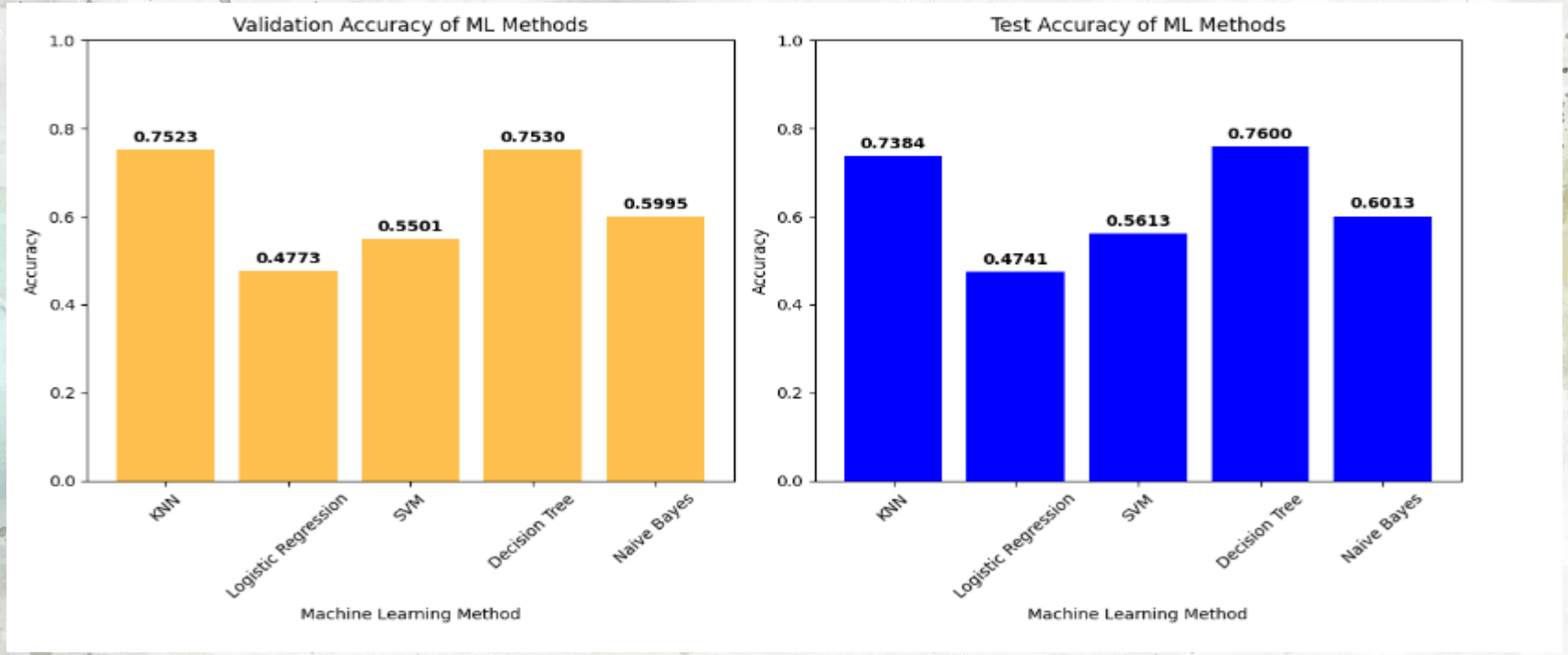

```
Training Accuracy for Decision Tree: 1.0000
Testing Accuracy for Decision Tree: 0.7600
Validation Accuracy for Decision Tree: 0.7530
```

For Naïve Bayes




```
Training Accuracy for Naive Bayes: 0.6417
Testing Accuracy for Naive Bayes: 0.6013
Validation Accuracy for Naive Bayes: 0.5995
```

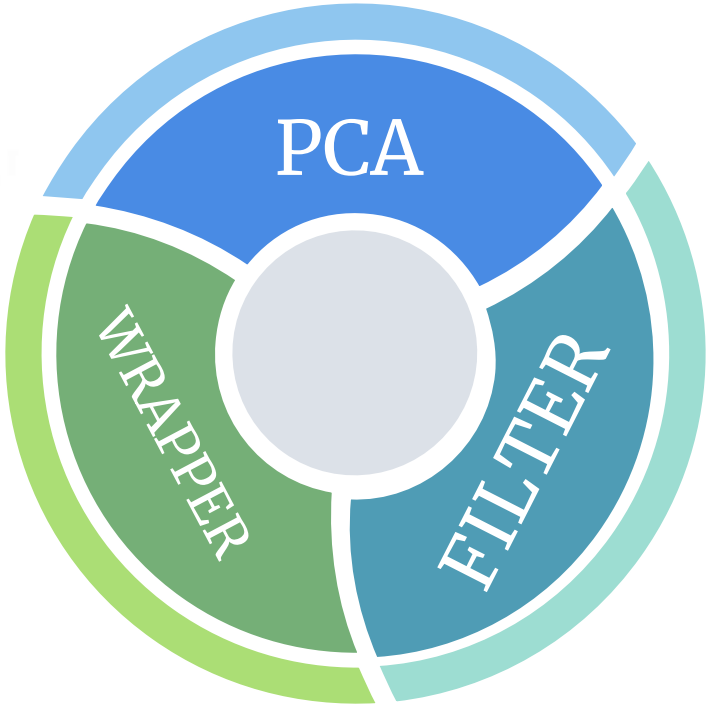
Accuracy Comparison



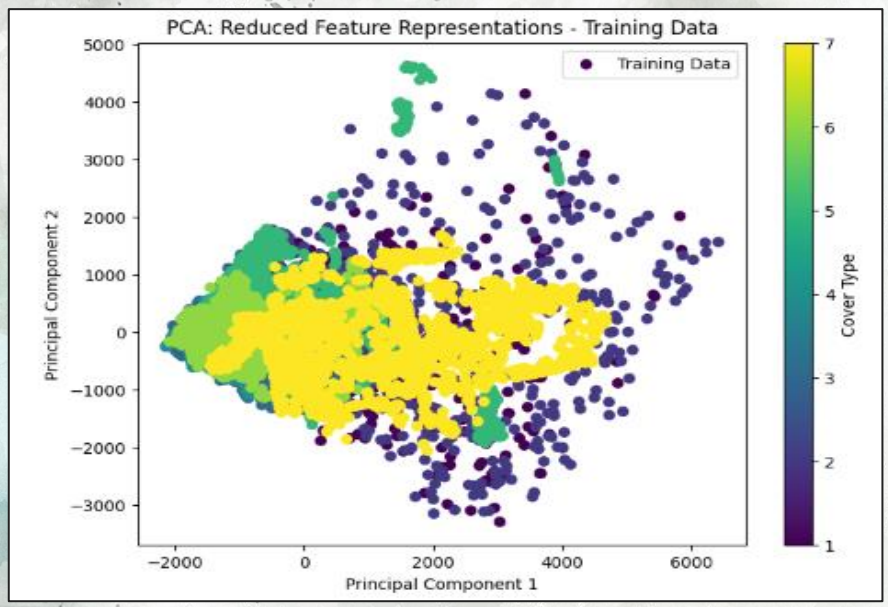
The Best 2 Baseline are Decision Tree and KNN

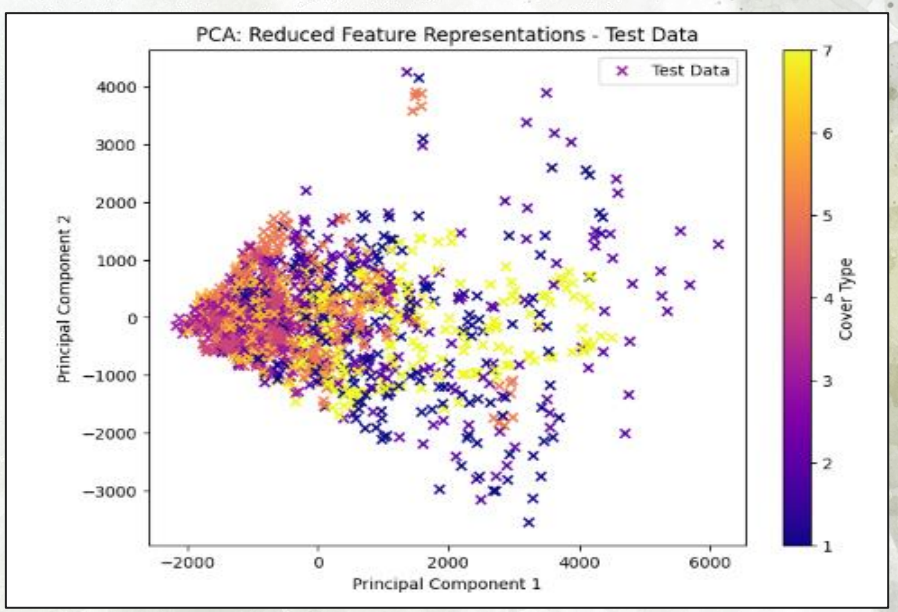
Improvement Strategy

Improvement Strategy

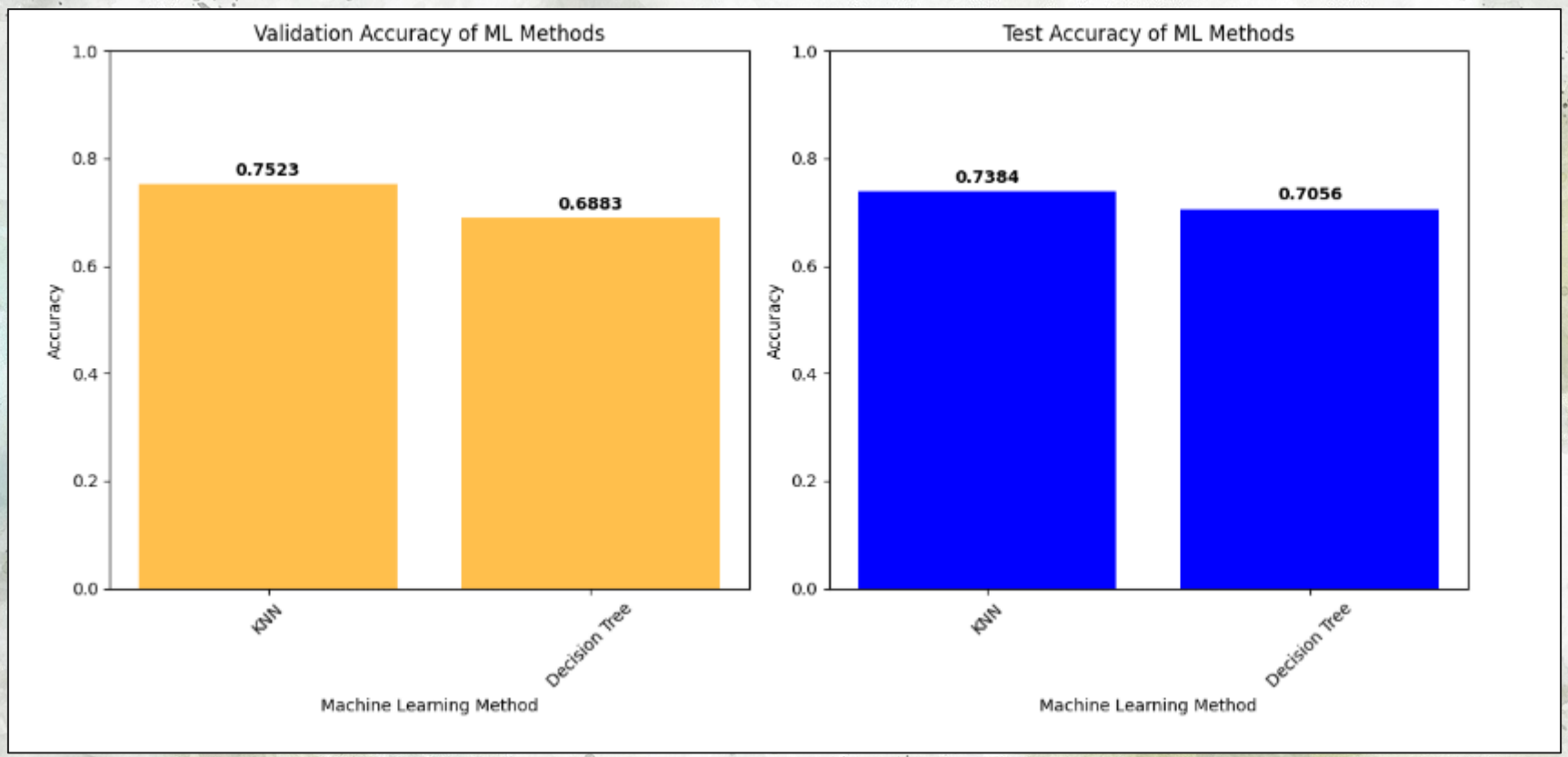


Applying PCA

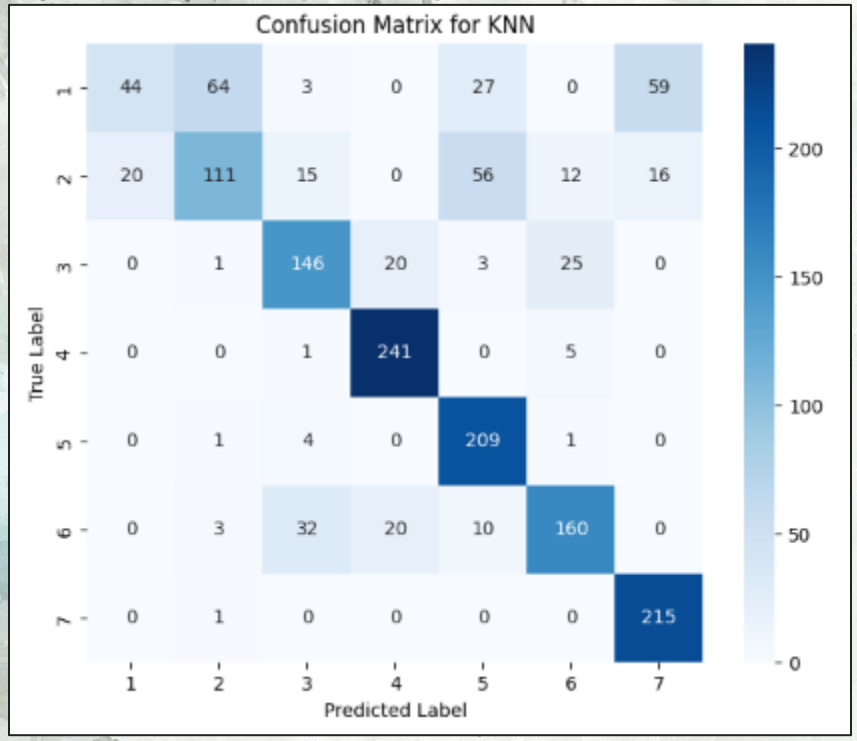


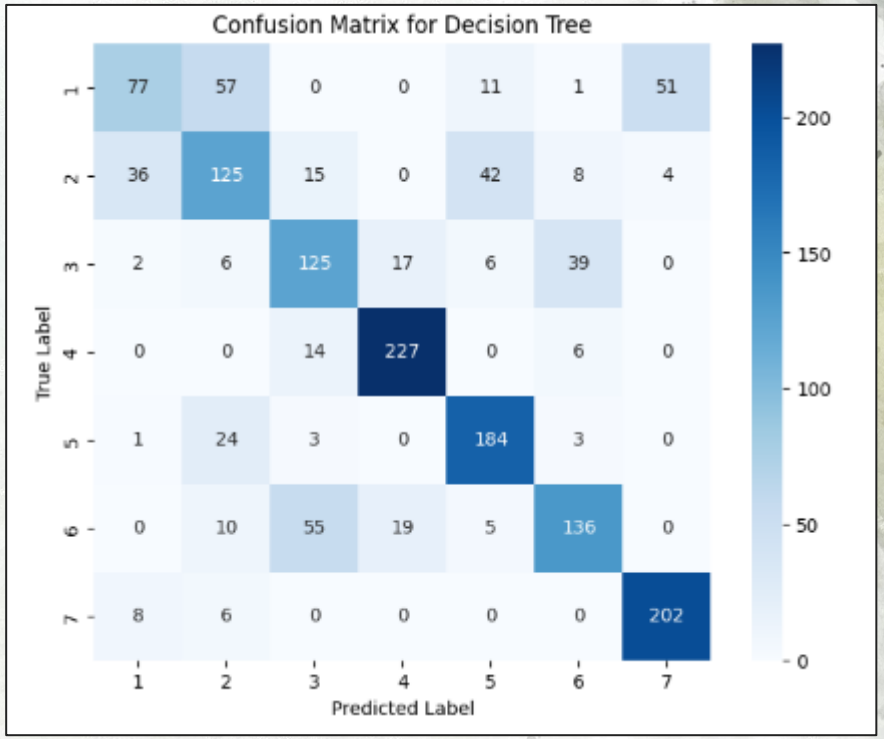


Comparing 2 models after PCA



Comparing 2 models after PCA

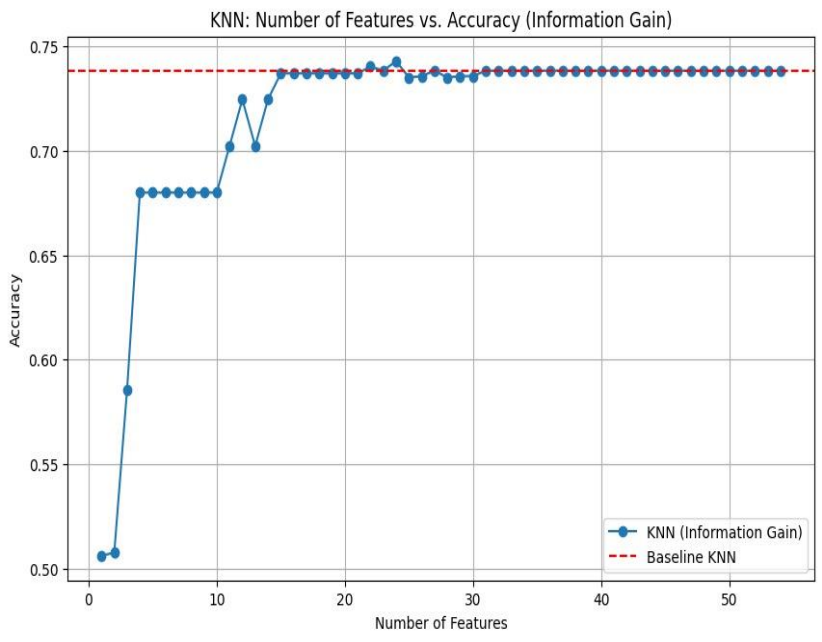


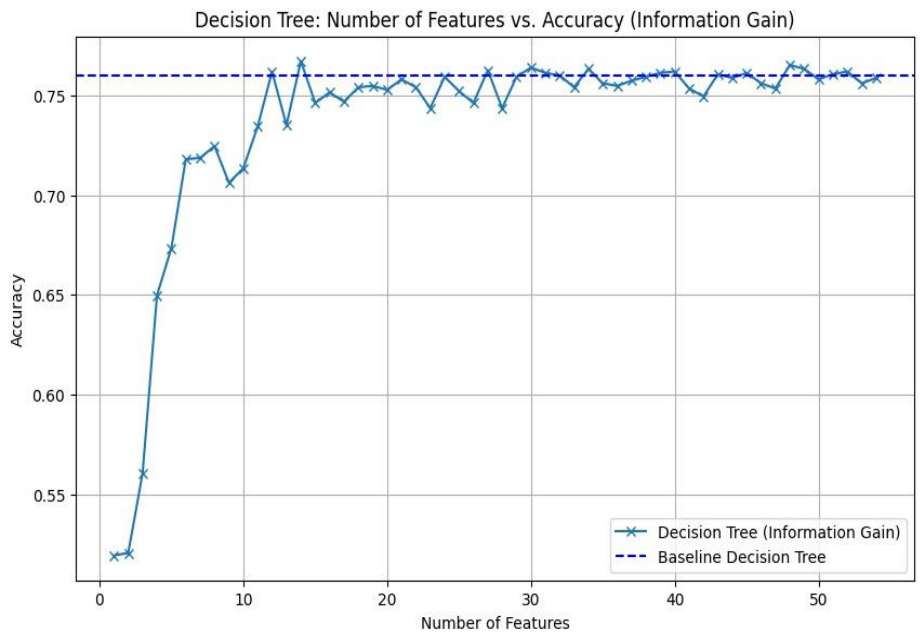



```
Training Accuracy for KNN: 0.8853
Testing Accuracy for KNN: 0.7384
Validation Accuracy for KNN: 0.7523
```

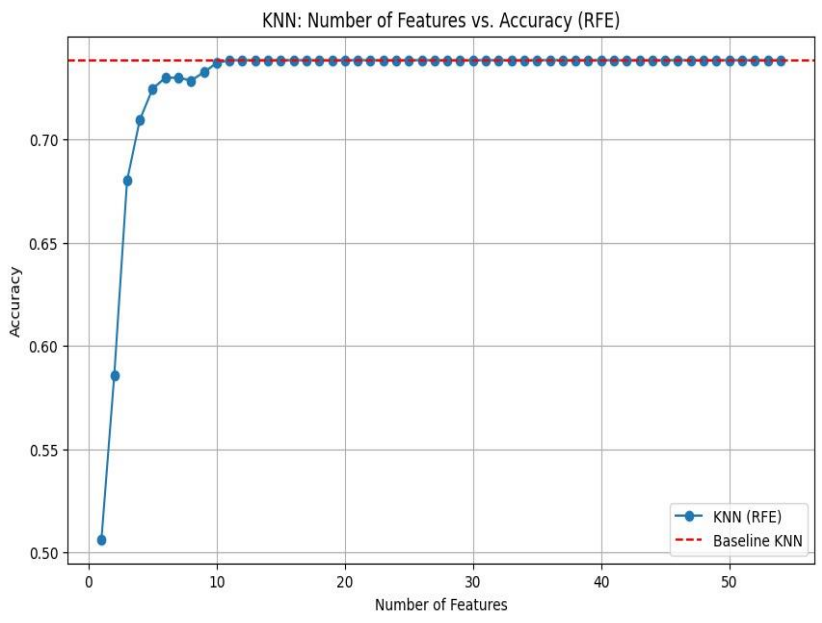
```
Training Accuracy for Decision Tree: 1.0000
Testing Accuracy for Decision Tree: 0.7056
Validation Accuracy for Decision Tree: 0.6883
```

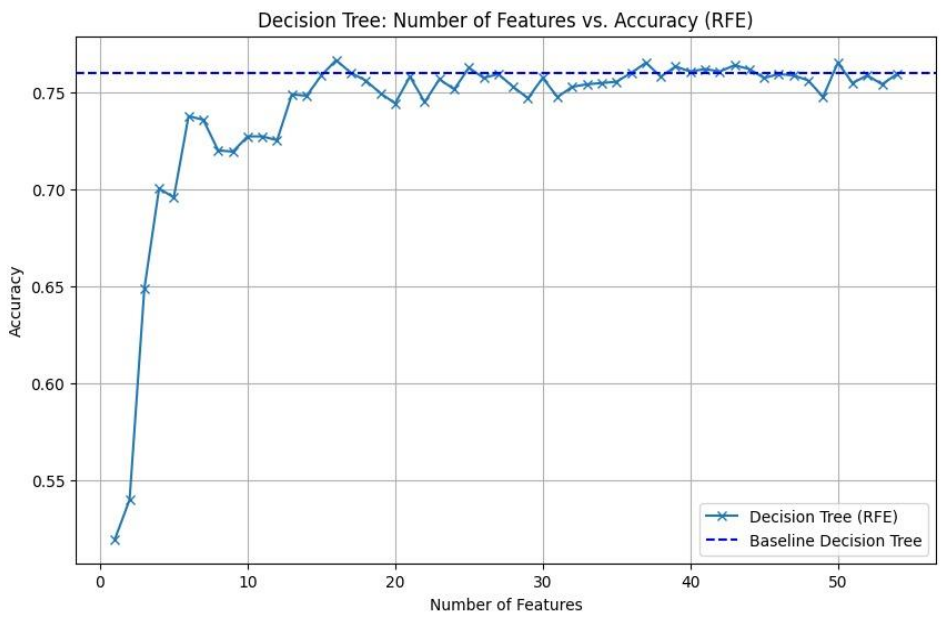
Applying Filter





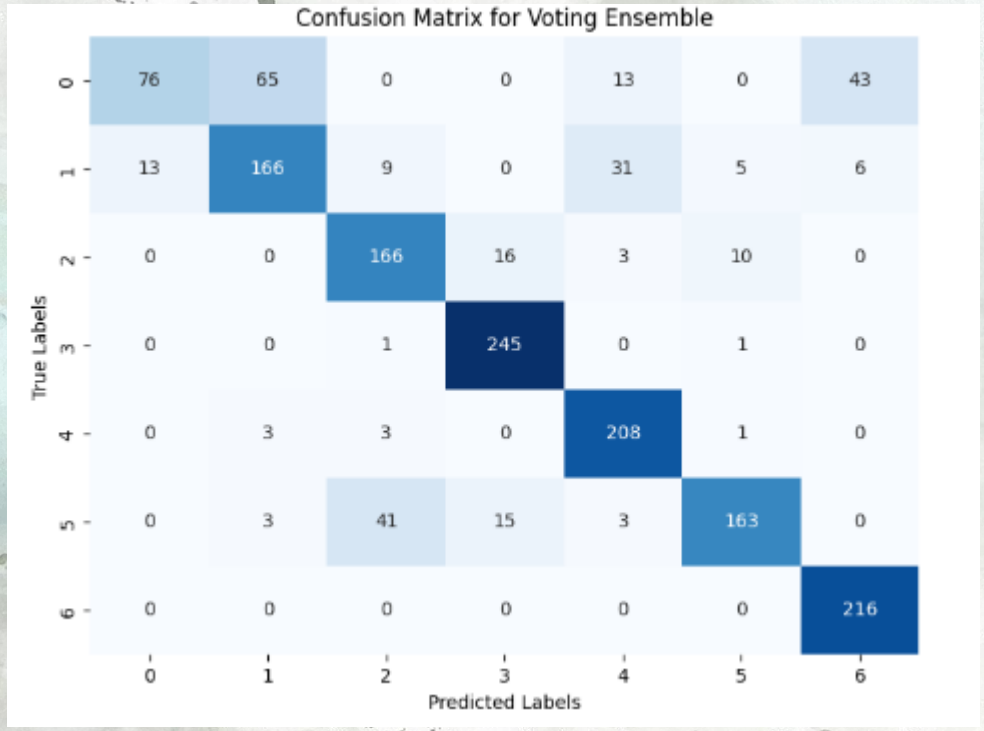
Applying Wrapper





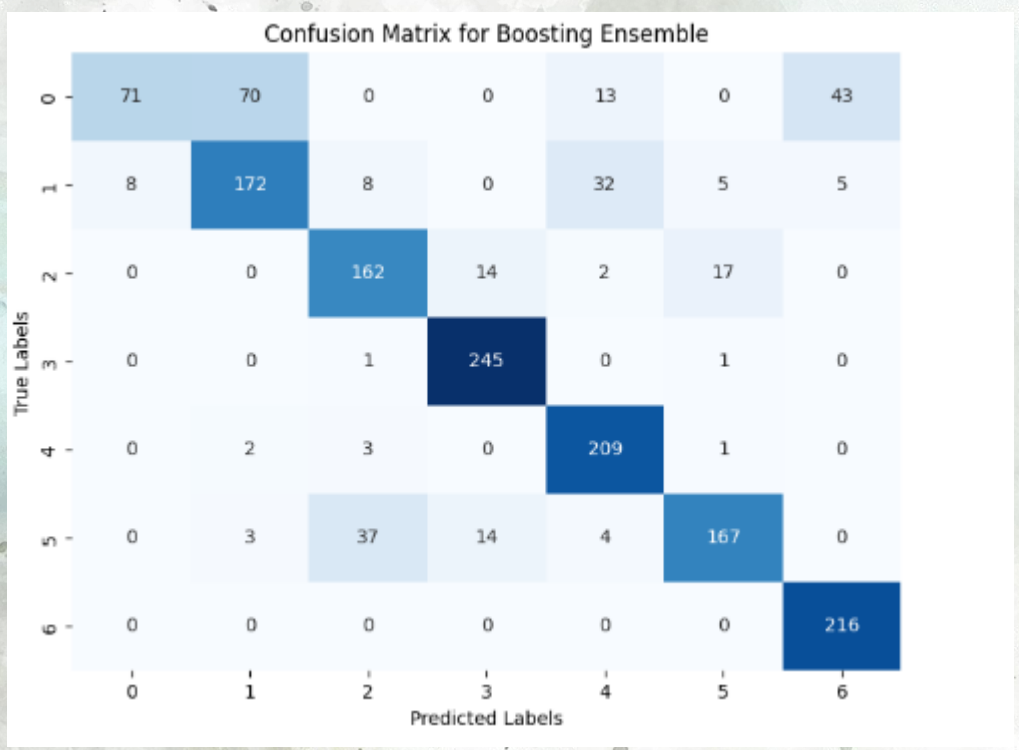
Applying New models

VOTING Model



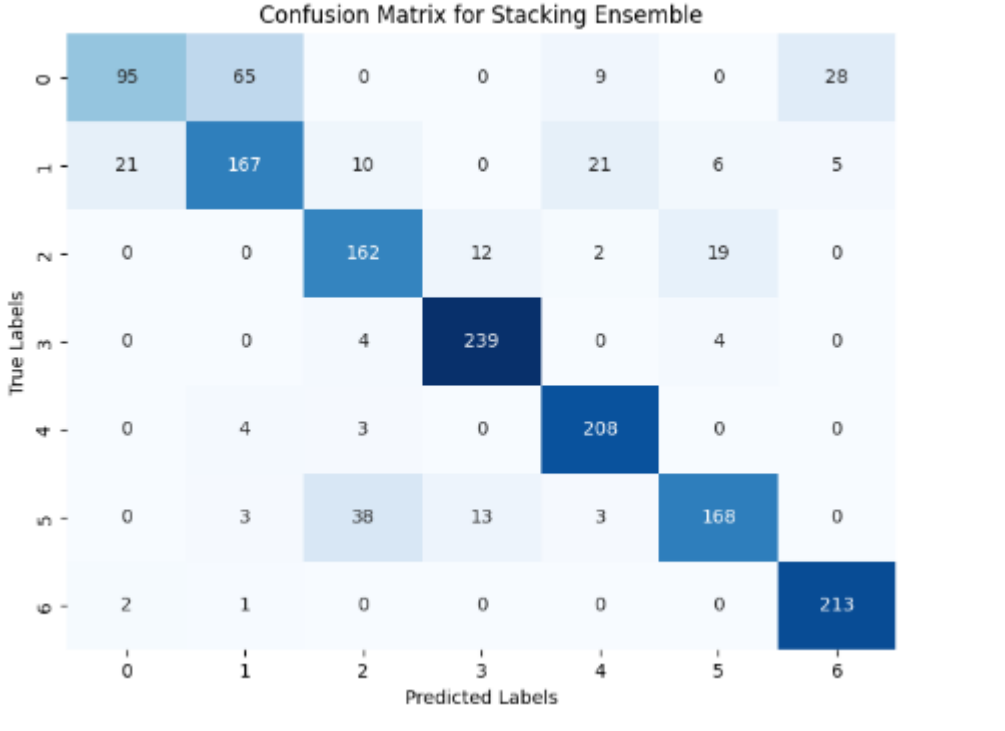
Accuracy of Voting Ensemble: 0.8131147540983606

Boosting Model



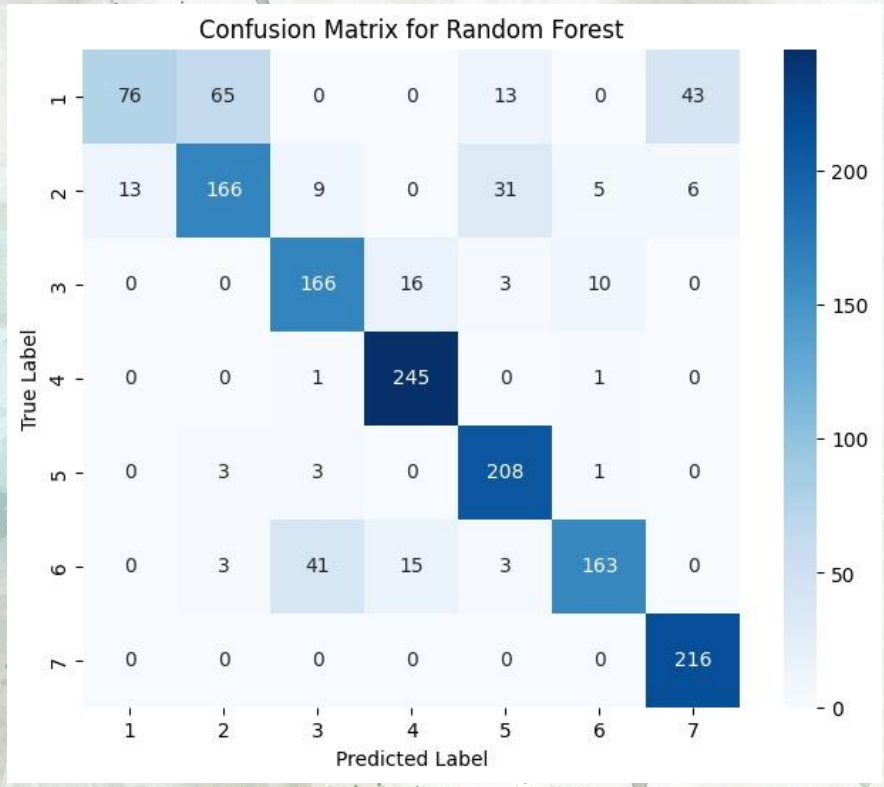
Accuracy of Boosting: 0.8144262295081968

Stacking Model

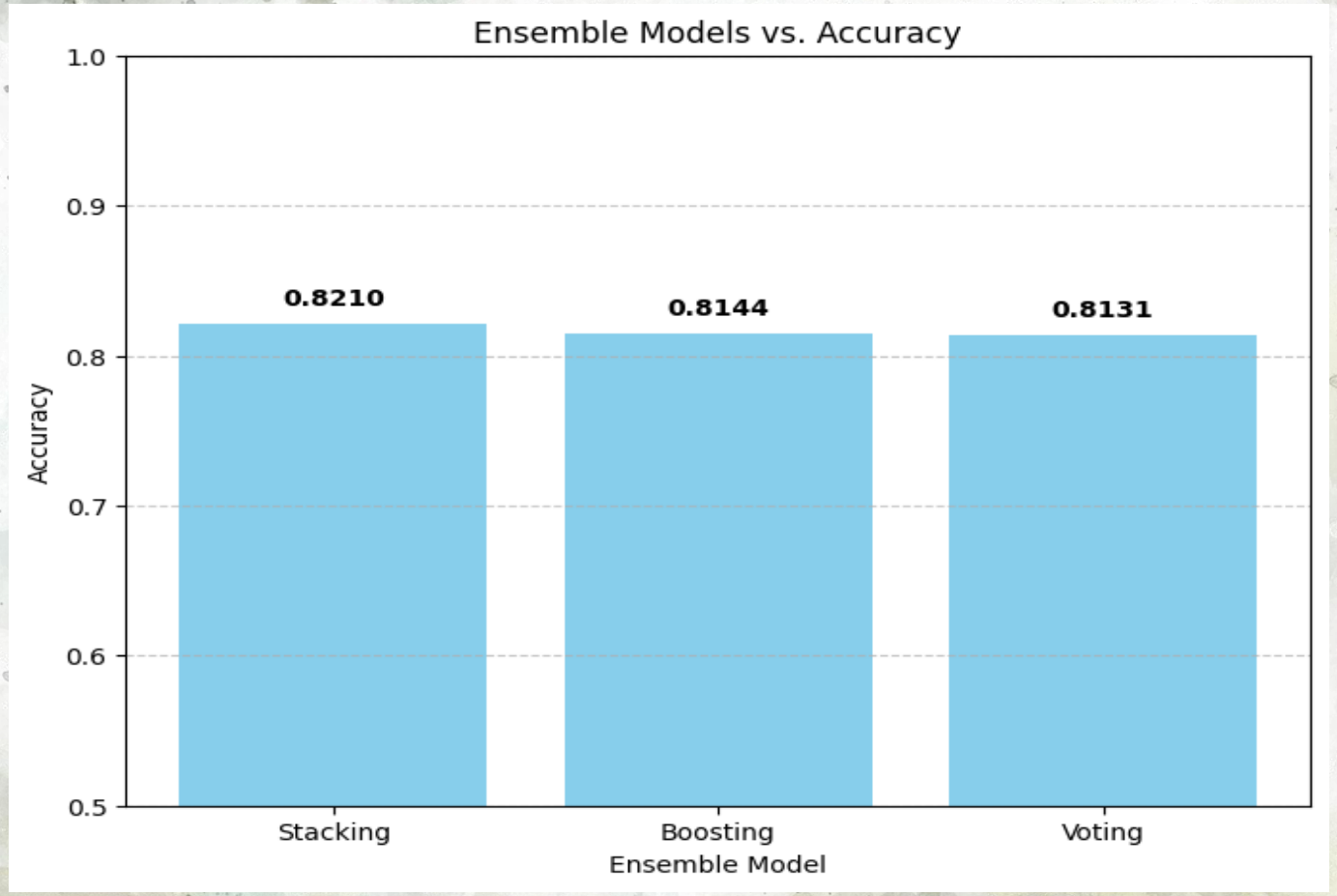



```
Accuracy of Stacking: 0.820983606557377
```

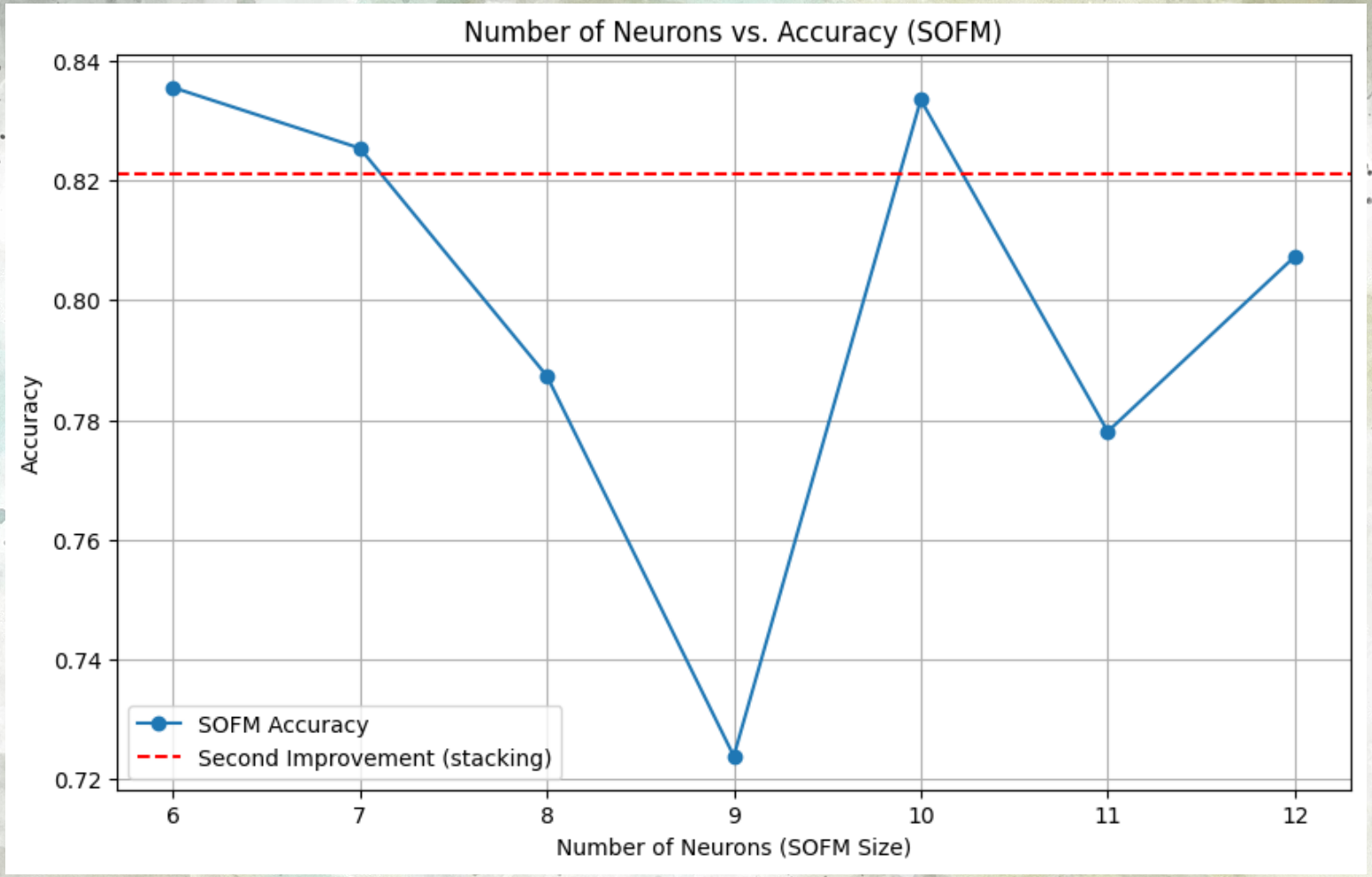
Random Forest Model

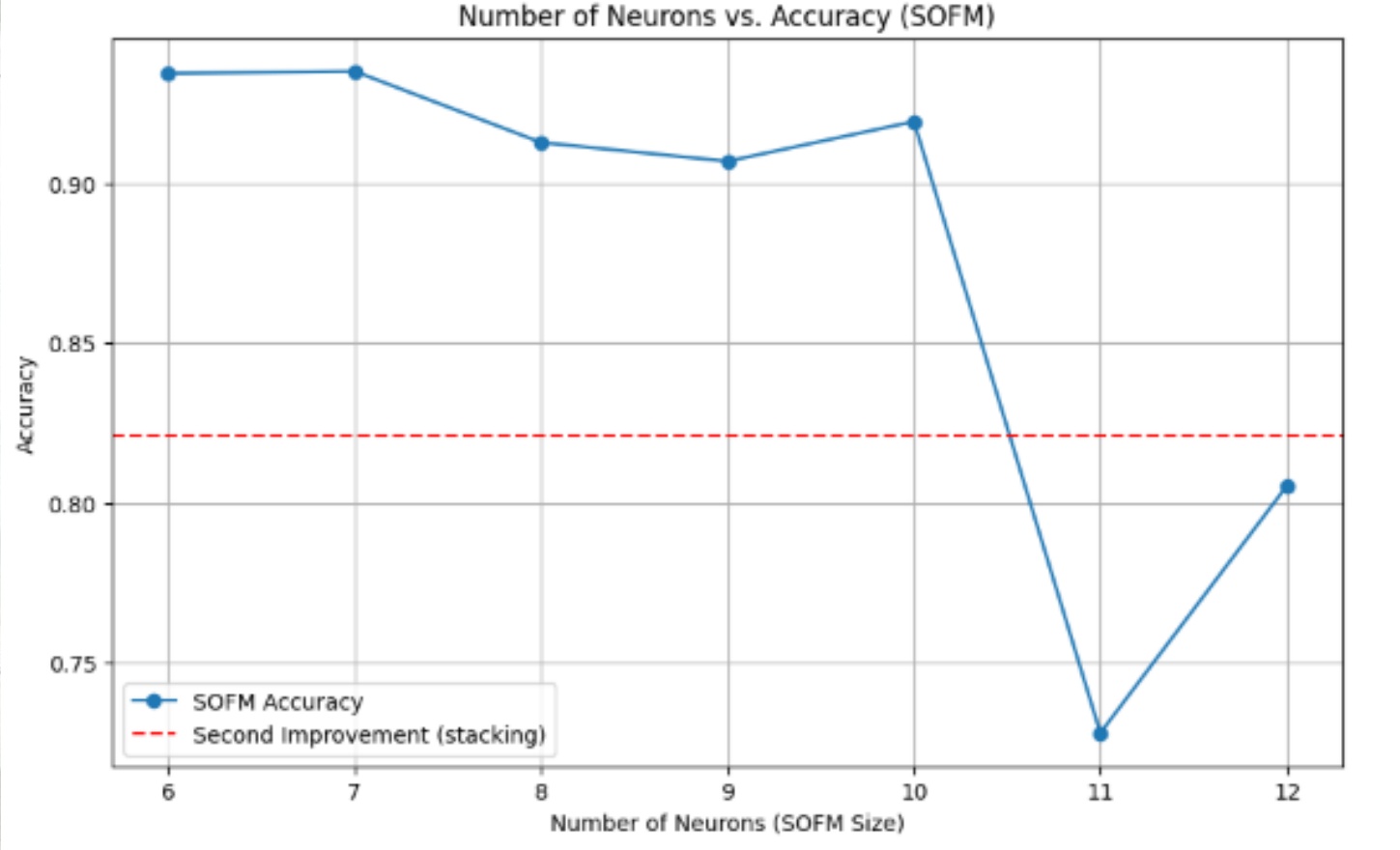


Testing Accuracy for Random Forest: 0.8131

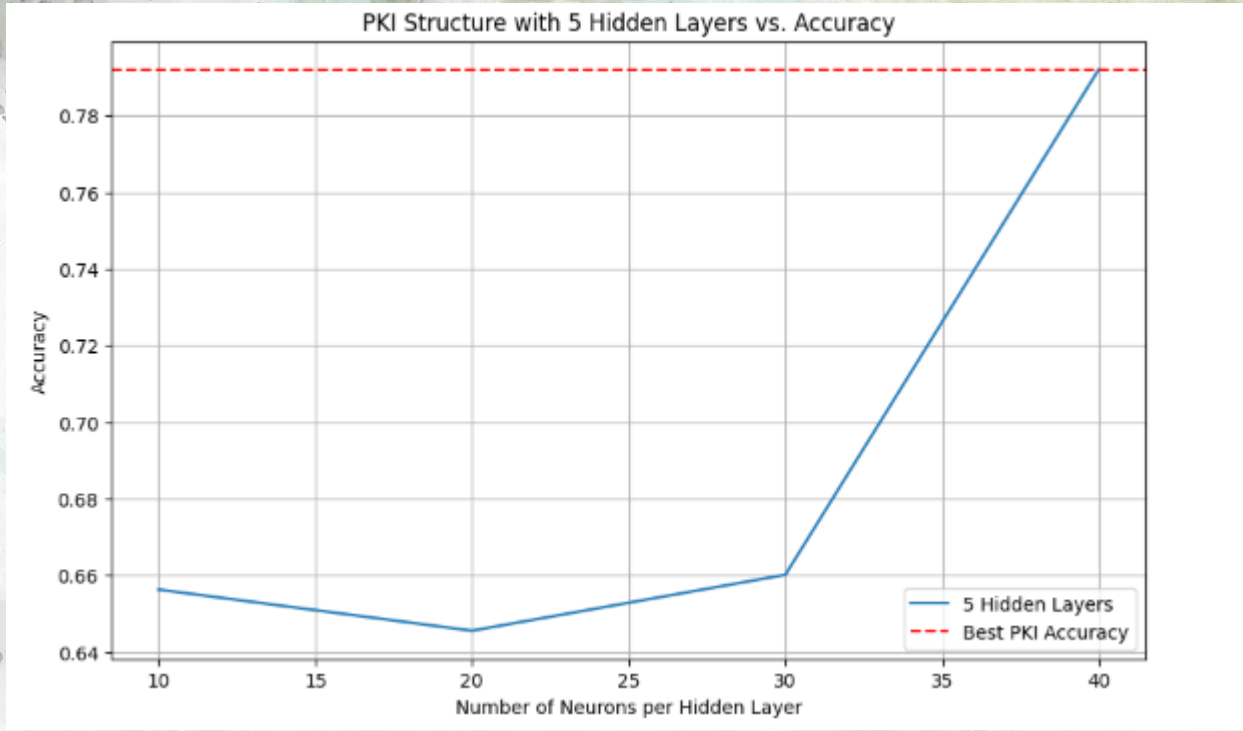

Ensemble Models vs. Accuracy
1.0
0.9
0.8
0.7
0.6
0.5
Accuracy
0.8210
0.8144
0.8131
Stacking
Boosting
Voting
Ensemble Model

Supervised & Unsupervised Combination via PKI


Number of Neurons vs. Accuracy (SOFM)
Accuracy
Number of Neurons (SOFM Size)
0.84
0.82
0.80
0.78
0.76
0.74
0.72
6
7
8
9
10
11
12
SOFM Accuracy
Second Improvement (stacking)


Number of Neurons vs. Accuracy (SOFM)
Accuracy
0.90
0.85
0.80
0.75
SOFM Accuracy
Second Improvement (stacking)
6
7
8
9
10
11
12
Number of Neurons (SOFM Size)

fine tuning



Best SOFM Structure: 10x10, Accuracy: 0.8425 Best PKI Structure: 5 Hidden Layers, 40 Neurons per Layer, Accuracy: 0.7921

conclusion

1. **First Improvement :** Decision Tree with Filter-based Feature Selection (SelectKBest): By selecting the most informative features using SelectKBest and applying Decision Tree, we improved the model's accuracy compared to the baseline.
2. **Second Improvement :**Random Forest: Utilizing Random Forest as an ensemble model further boosted the accuracy beyond the first improvement.

3. **Third Improvement** : PKI Model (SOFM + DNN): - Incorporating Self-Organizing Feature Map (SOFM) clustering information into a Deep Neural Network (DNN) model provided significant accuracy improvement compared to the previous strategies.

By sequentially applying these three improvement strategies, we achieved substantial enhancements in the supervised model's accuracy and performance, making it more capable of making accurate predictions on the target dataset.

Thanks!