COL341 - Assignment 3

Valaya - 2019MT10731

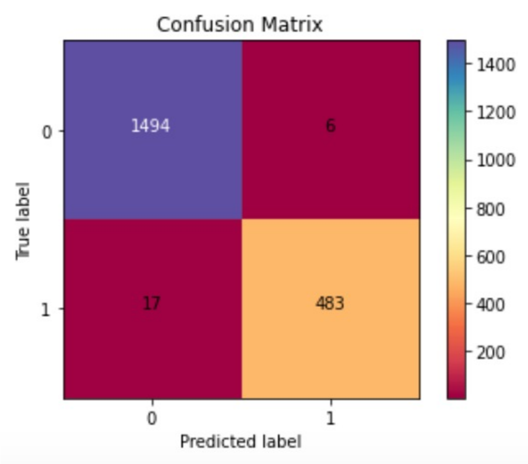
April 20, 2023

3.1 Binary Classification

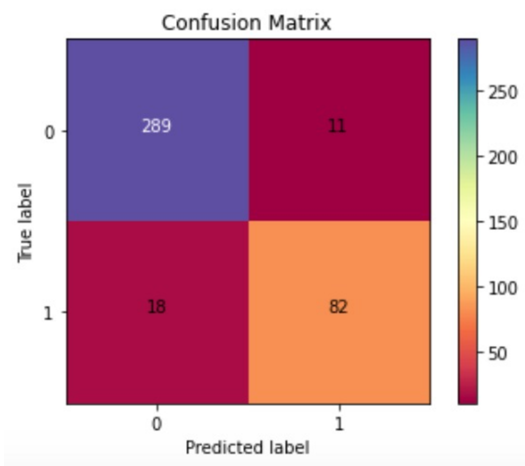
a) Decision Tree from scratch

1. Gini Index

	Train	Validation
Accuracy	0.9885	0.9275
Precision	0.9877	0.8817
Recall Value	0.966	0.82
Training Time	261.76 sec	



(a) Training Data

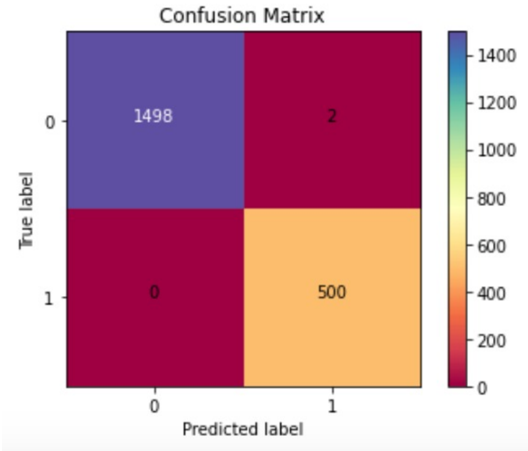


(b) Validation Data

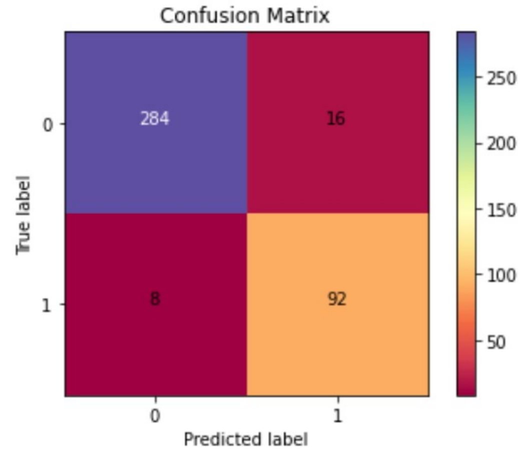
Figure 1: Confusion Matrices with gini criterion

2. Information Gain

	Train	Validation
Accuracy	0.999	0.94
Precision	0.996	0.8518
Recall Value	1.0	0.92
Training Time	568.46 sec	



(a) **Training Data**



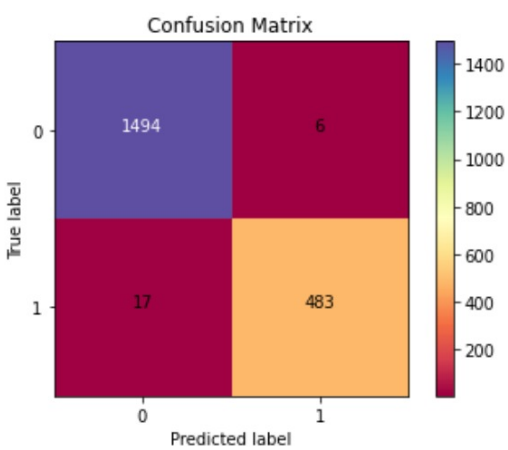
(b) **Validation Data**

Figure 2: Confusion Matrices with entropy criterion

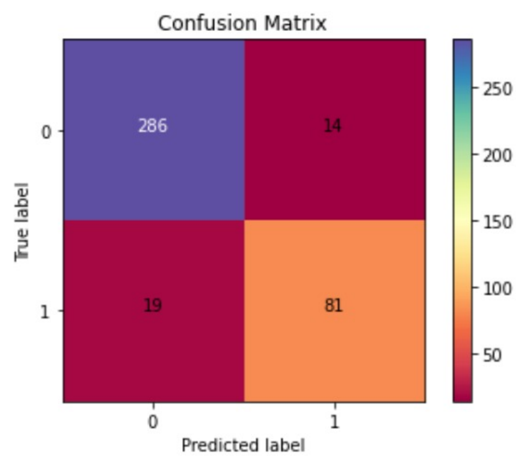
b) Decision Tree sklearn

1. Gini Index

	Train	Validation
Accuracy	0.9885	0.9175
Precision	0.9885	0.9175
Recall Value	0.9885	0.9175
Training Time	2.86 sec	



(a) **Training Data**

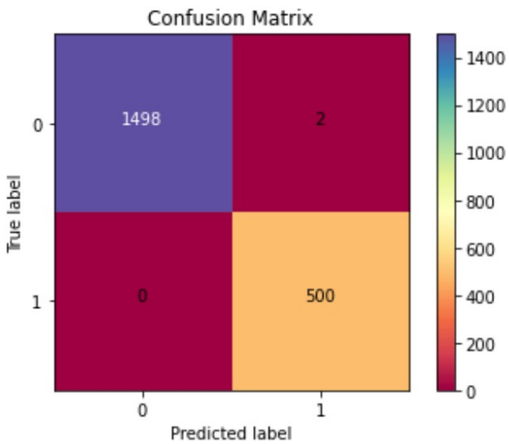


(b) **Validation Data**

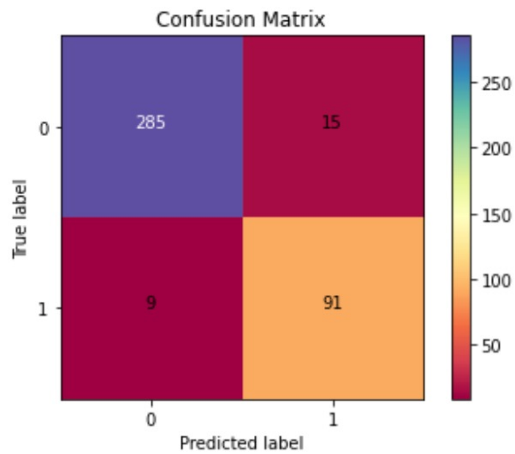
Figure 3: Confusion Matrices with entropy criterion

2. Information Gain

	Train	Validation
Accuracy	0.999	0.94
Precision	0.999	0.9417
Recall Value	0.999	0.94
Training Time	1.78 sec	



(a) Training Data



(b) Validation Data

Figure 4: Confusion Matrices with entropy criterion

The accuracy scores for both the model implementations (Gini index and information gain) are similar in our implementation from scratch and the sklearn implementation. However, the time taken by the sklearn model is significantly lesser (1-2 sec) compared to the time taken by our implementation form scratch (200-500 sec). The potential reasons for this could be -

- The code for the decision tree algorithm provided by sklearn is likely to be very efficient and optimized for performance
- Sklearn may use more efficient data structures for representing the decision tree
- Sklearn may use parallel processing techniques to speed up the algorithm

c) Decision Tree Grid-Search and Visualisation

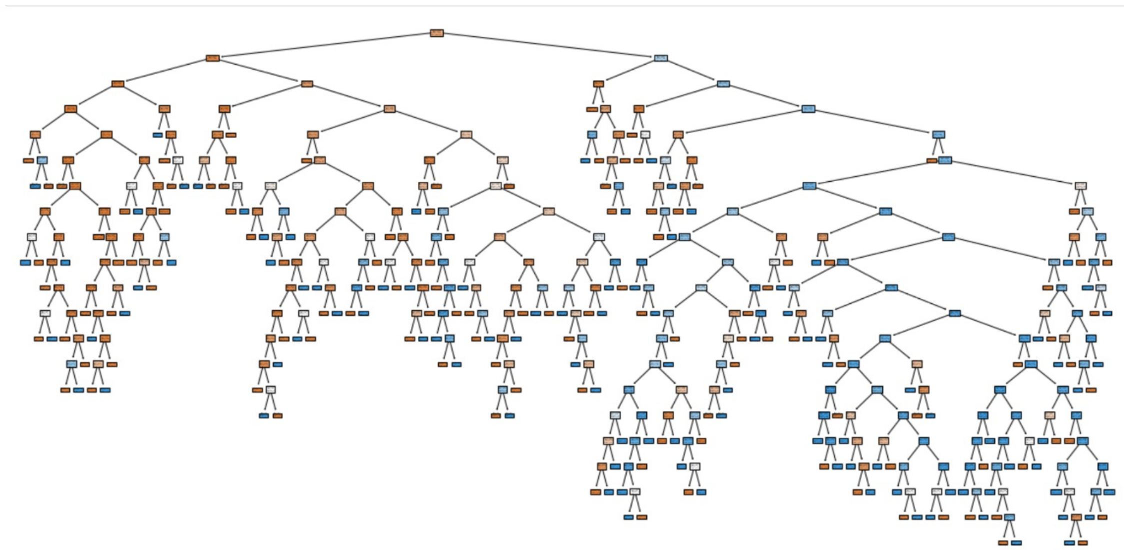
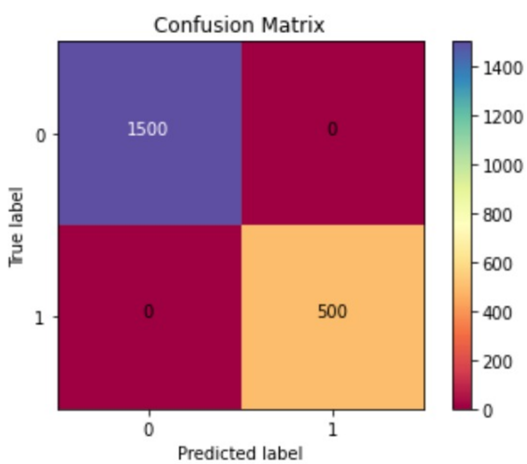


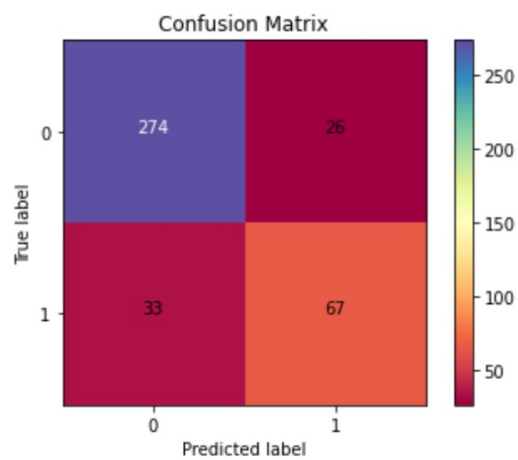
Figure 5: Tree Visualisation after Feature Selection

	Train	Validation
Accuracy	1.0000	0.8525
Precision	1.0000	0.8495
Recall Value	1.0000	0.8525

Table 1: Observations after feature selection



(a) Training Data

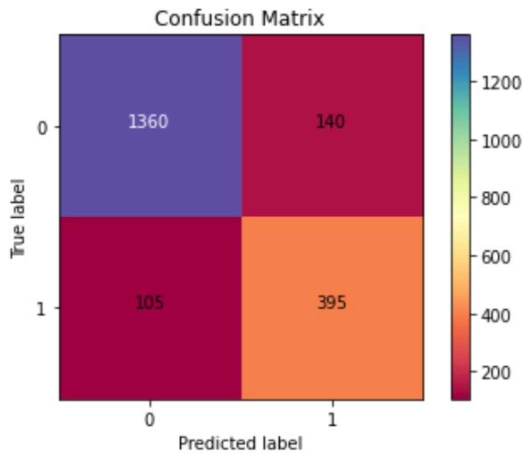


(b) Validation Data

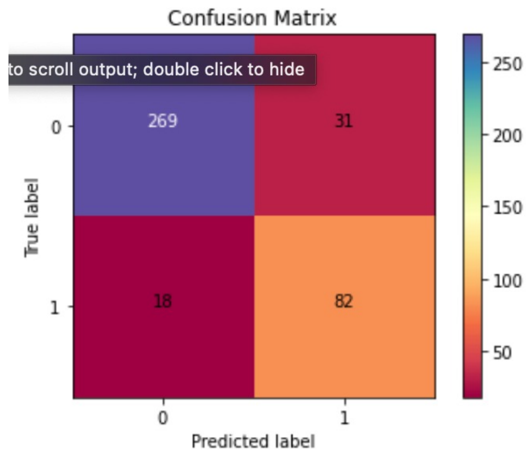
Figure 6: Confusion Matrices after Feature Selection

	Train	Validation
Accuracy	0.88	0.88
Precision	0.8808	0.8844
Recall Value	0.8775	0.8775

Table 2: Observations after grid search



(a) Training Data



(b) Validation Data

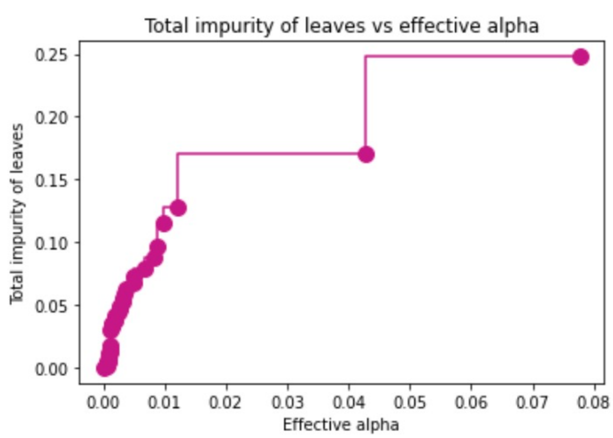
Figure 7: Confusion Matrices after grid search

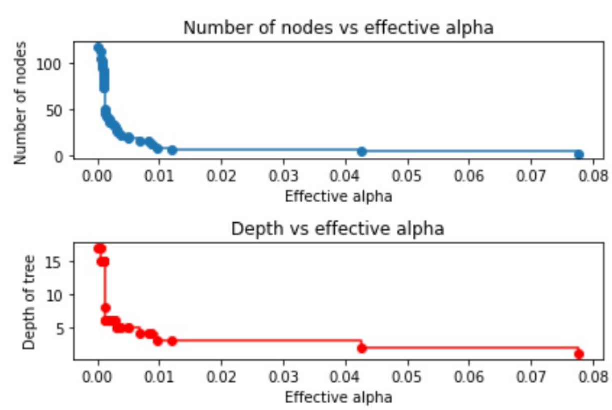
After feature selection, the performance metrics have reduced compared to the models implemented in parts a) and b) implying that 10 features aren't enough to capture the relevant information of the data.

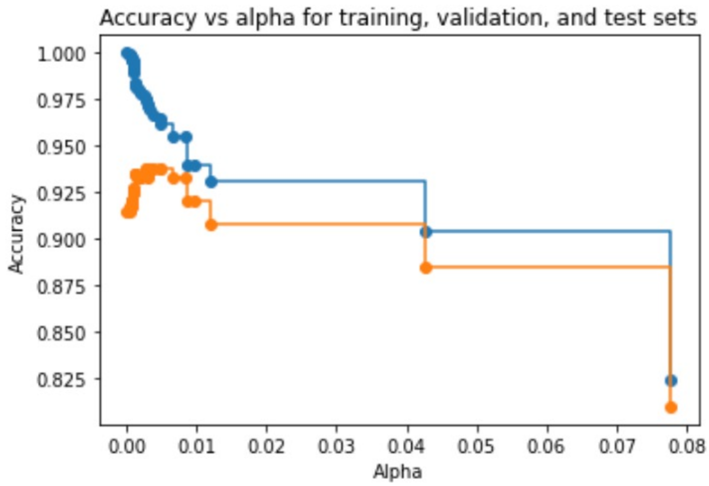
Best Parameters: criterion: 'entropy', max˙depth: 5, min˙samples˙split: 4

d) Decision Tree Post Pruning with Cost Complexity Pruning

A higher value of the cost complexity parameter ccp_alpha indicates that greater amount of pruning which implies a greater impurity at leaf nodes. As can be seen from the graphs, as the effective alpha increases, depth of the pruned tree and number of nodes decreases. A decrease in training accuracy can be observed with decreasing alpha, however in the case of validation, there's a slight increase in the beginning, followed by a decrease. A possible reason for this could be that the unpruned tree might have overfit, howvere pruning it a lot resulted in underfit.







	Train	Validation
Accuracy	0.9755	0.9375

Table 3: Training and validation Accuracy for Best Tree

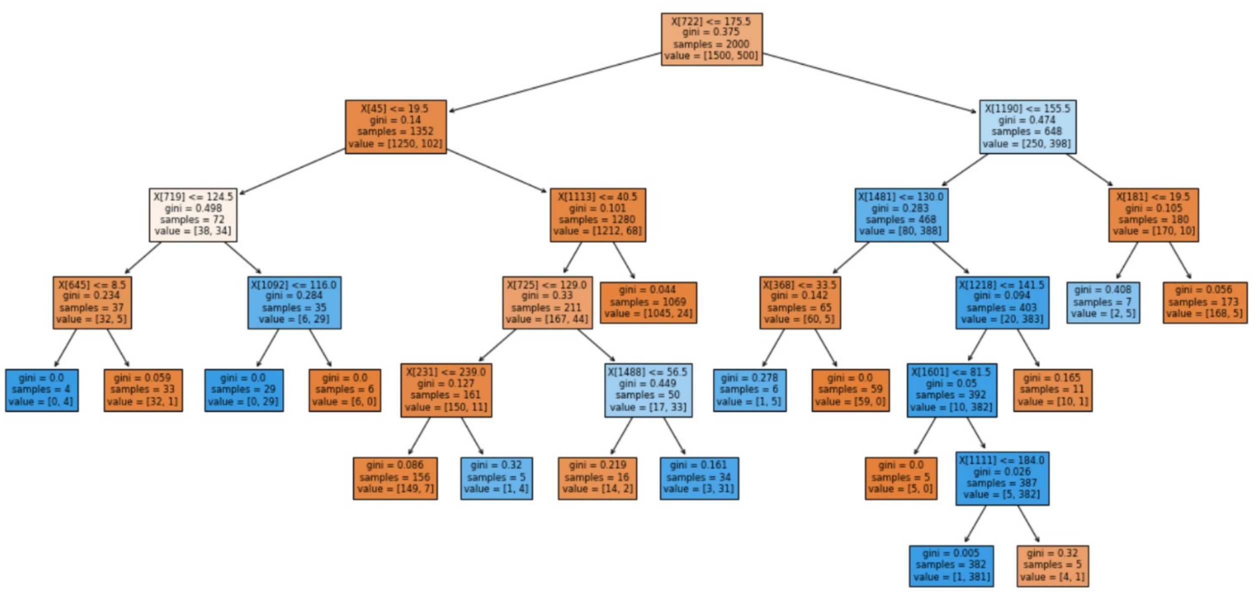
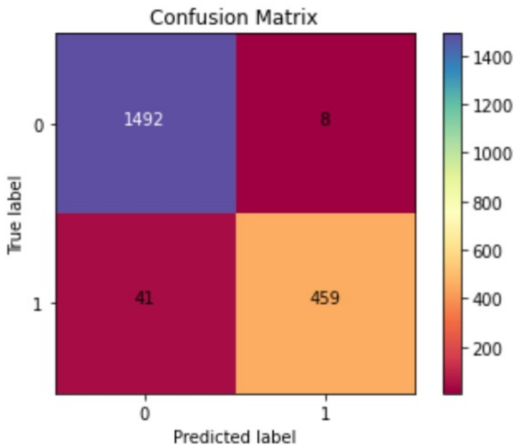
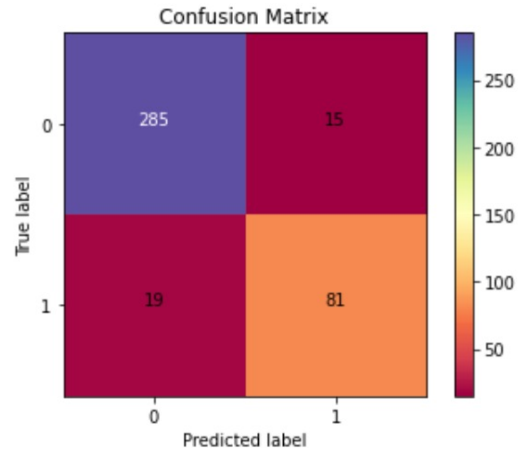


Figure 9: Best-Pruned-Tree Visualisation



(a) **Training Data**



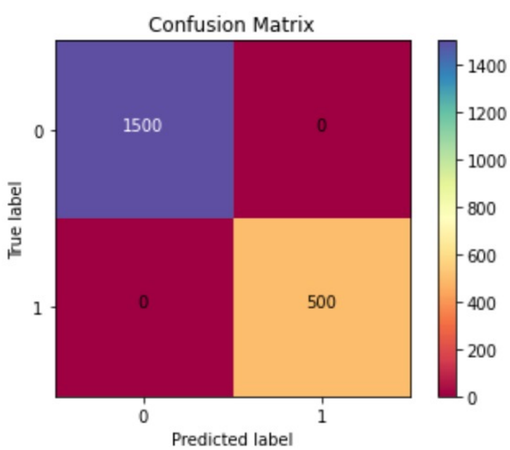
(b) **Validation Data**

Figure 10: Confusion Matrices of the Best Pruned Tree

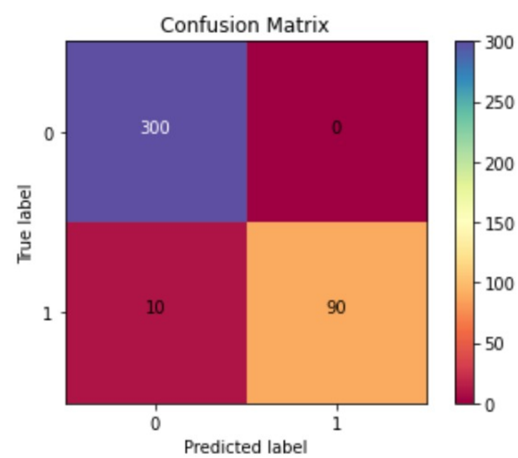
e) Random Forest

	Train	Validation
Accuracy	1.0	0.975
Precision	1.0	1.0
Recall Value	1.0	0.9

Table 4: Observations of Random Forest



(a) **Training Data**

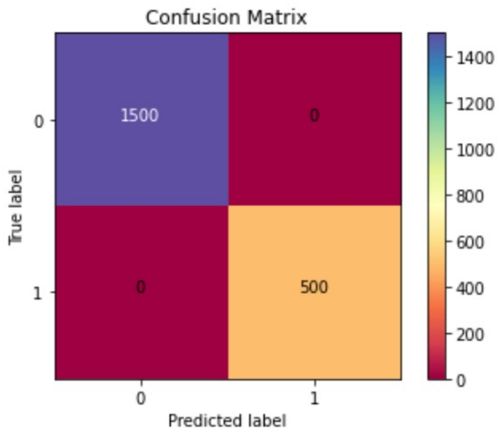


(b) **Validation Data**

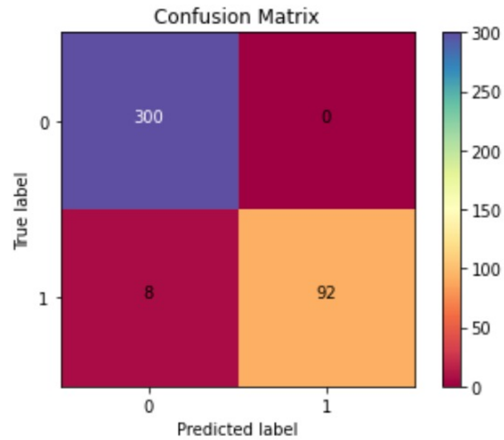
Figure 11: Confusion Matrices of Random Forest

	Train	Validation
Accuracy	1.0	0.98
Precision	1.0	1.0
Recall Value	1.0	0.92

Table 5: Observations of Random Forest after Grid Search



(a) **Training Data**



(b) **Validation Data**

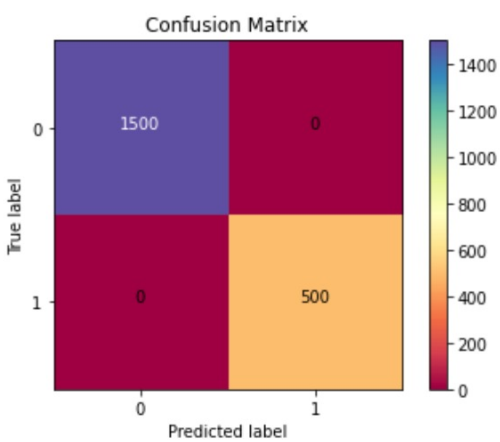
Figure 12: Confusion Matrices of Random Forest after Grid Search

Best parameters: criterion: 'entropy', max_depth: 7, min_samples_split: 5, n_estimators: 150

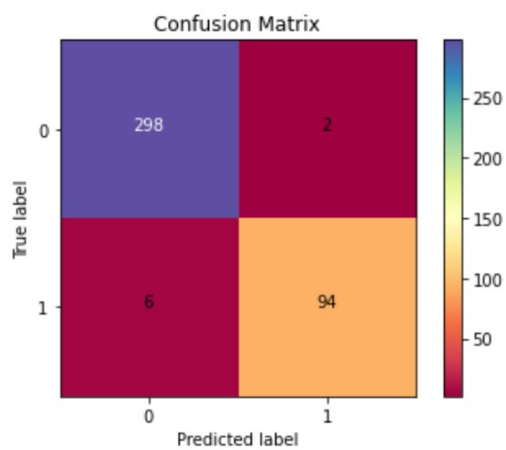
f) Gradient Boosted Trees and XGBoost

	Train	Validation
Accuracy	1.0	0.98
Precision	1.0	0.98
Recall Value	1.0	0.98
Training Time	86.32 sec	

Table 6: Observations of Gradient Boosted Trees



(a) **Training Data**

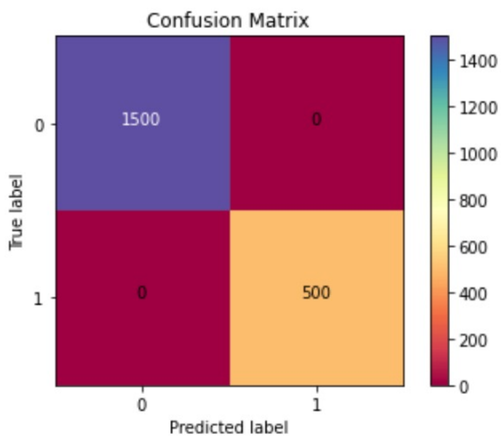


(b) **Validation Data**

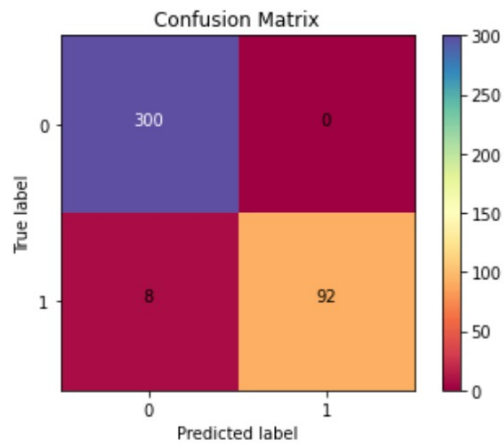
Figure 13: Confusion Matrices of Gradient Boosted Trees

	Train	Validation
Accuracy	1.0	0.985
Precision	1.0	1.0
Recall Value	1.0	0.94

Table 7: Observations of Gradient Boosted Trees after Grid Search



(a) **Training Data**



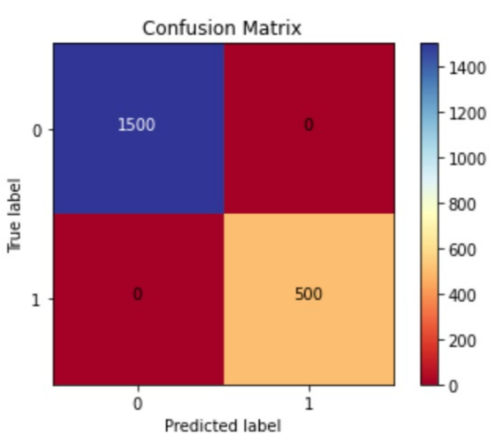
(b) **Validation Data**

Figure 14: Confusion Matrices of Gradient Boosted Trees after Grid Search

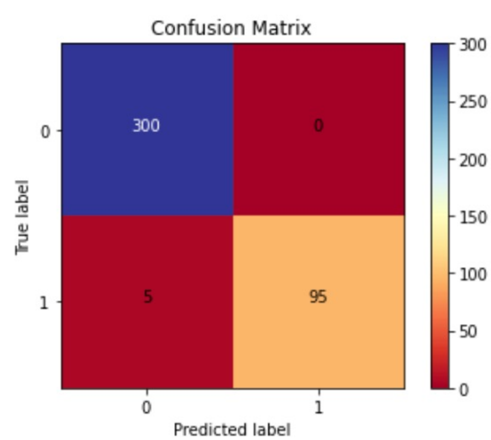
Best Parameters: max˙depth = 5, sub˙sample = 0.6, n˙estimators = 50

	Train	Validation
Accuracy	1.0	0.9875
Precision	1.0	1.0
Recall Value	1.0	0.95
Training Time	8.32 sec	

Table 8: Observations of XGBoost Trees



(a) **Training Data**

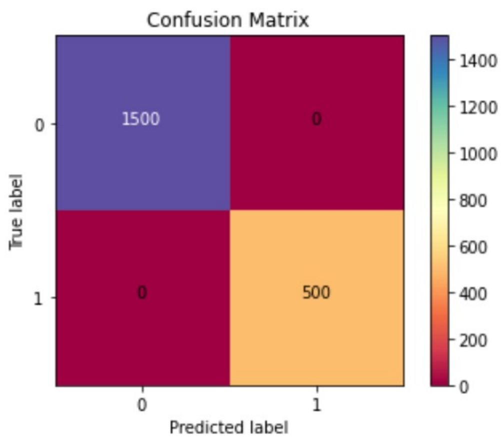


(b) **Validation Data**

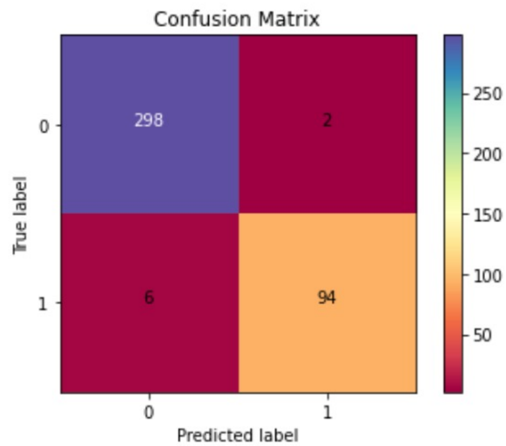
Figure 15: Confusion Matrices of XGBoost Trees

	Train	Validation
Accuracy	1.0	0.977
Precision	1.0	0.99
Recall Value	1.0	0.92

Table 9: Observations of XGBoosted Trees after Grid Search



(a) Training Data



(b) Validation Data

Figure 16: Confusion Matrices of XGBoosted Trees after Grid Search

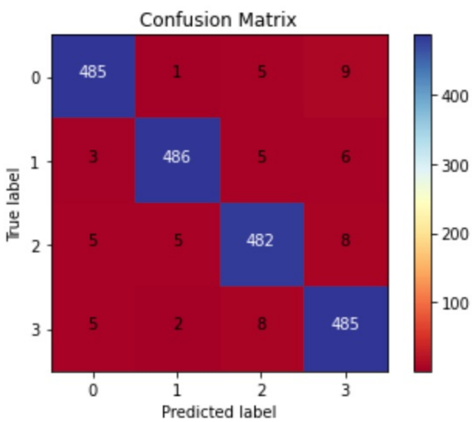
Best Parameters: max˙depth = 6, sub˙sample = 0.6, n˙estimators = 30

3.2 Multi-Class Classification

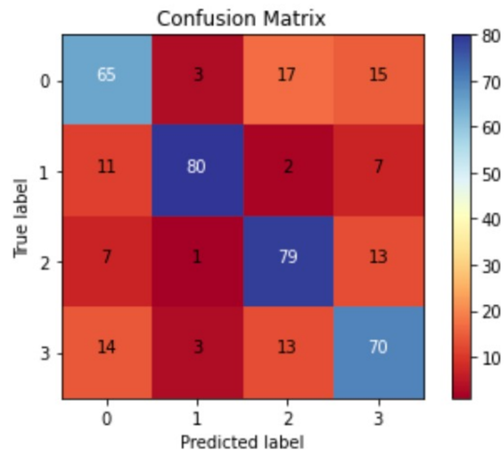
a) Decision Tree sklearn

1. Gini Index

	Train	Validation
Accuracy	0.969	0.74
Training Time	8.74 sec	



(a) Training Data

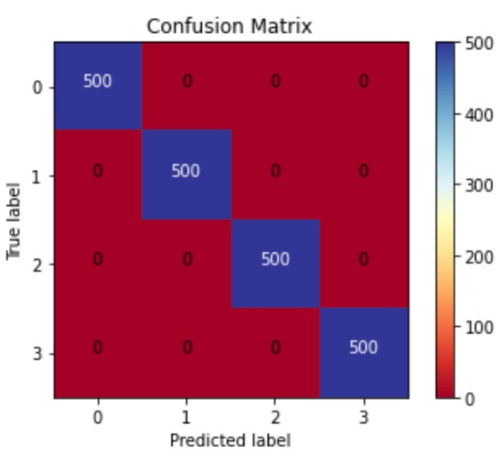


(b) Validation Data

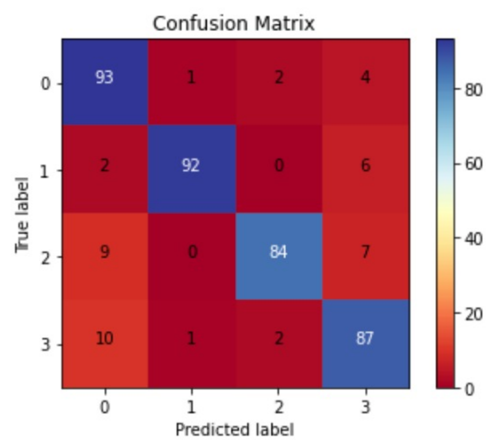
Figure 17: Confusion Matrices with gini criterion

2. Information Gain

	Train	Validation
Accuracy	0.97	0.7275
Training Time	11.3 sec	



(a) Training Data



(b) Validation Data

Figure 18: Confusion Matrices with entropy criterion

b) Decision Tree Grid Search and visualisation

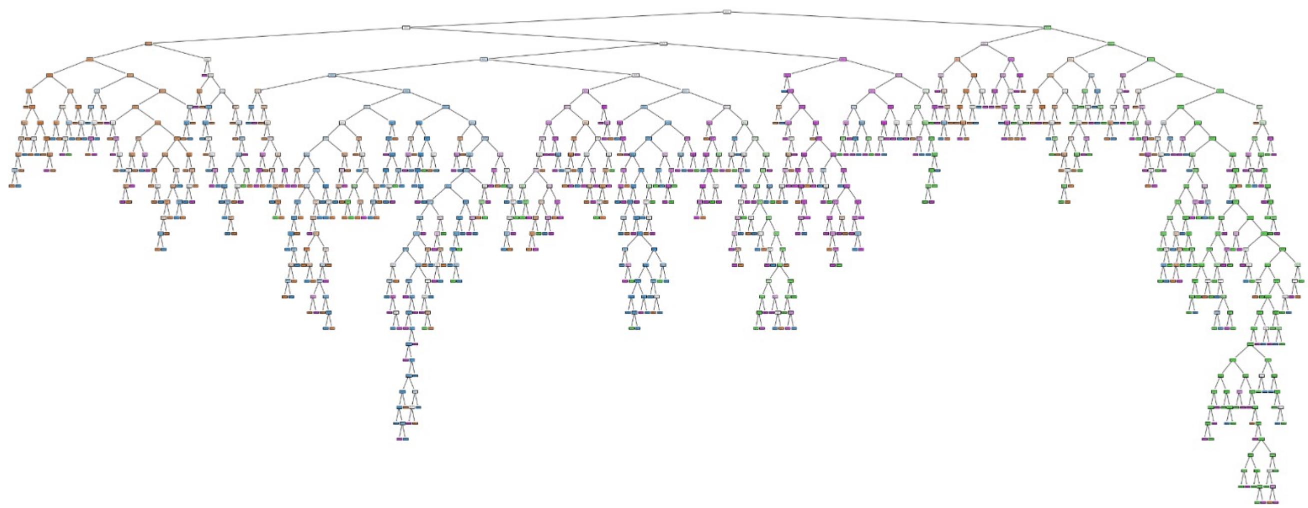
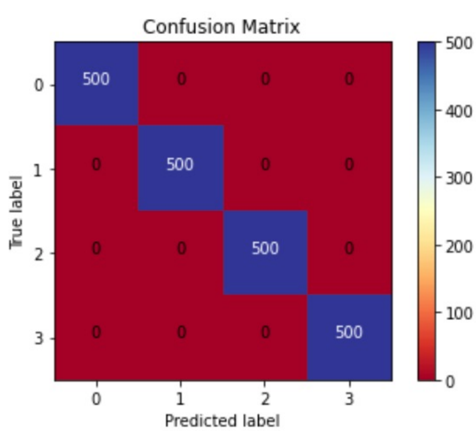


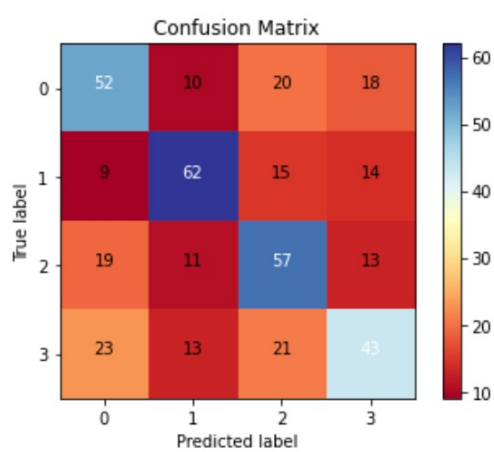
Figure 19: Tree Visualisation after Feature Selection

	Train	Validation
Accuracy	1.0000	0.535

Table 10: Observations after feature selection



(a) Training Data

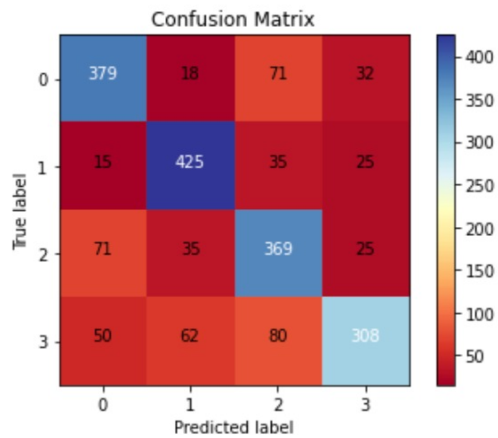


(b) Validation Data

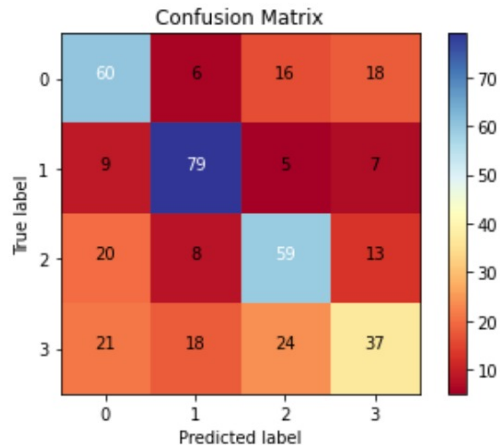
Figure 20: Confusion Matrices after Feature Selection

	Train	Validation
Accuracy	0.74	0.59
Training Time	0.05 sec	

Table 11: Observations after grid search



(a) Training Data



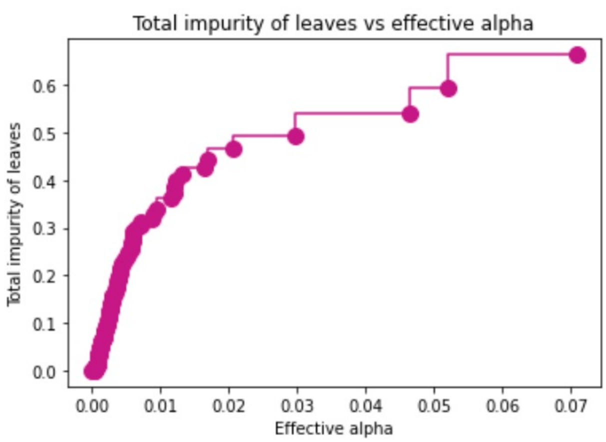
(b) Validation Data

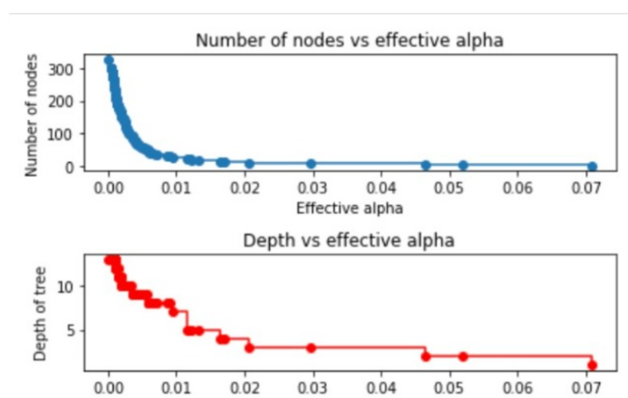
Figure 21: Confusion Matrices after grid search

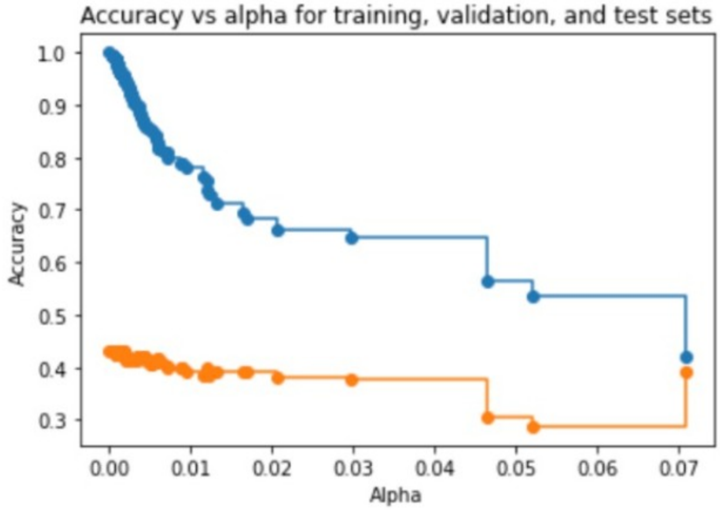
Best Parameters: criterion: 'entropy', max˙depth: 7, min˙samples˙split: 7

c) Decision Tree Post Pruning with Cost Complexity Pruning

Similar to binary, training accuracy decreases with alpha, and validation decreases and then increases. This suggests underfitting in the beginning, and overfitting with increased pruning.







	Train	Validation
Accuracy	0.9865	0.7375

Table 12: Training and validation Accuracy for Best Tree

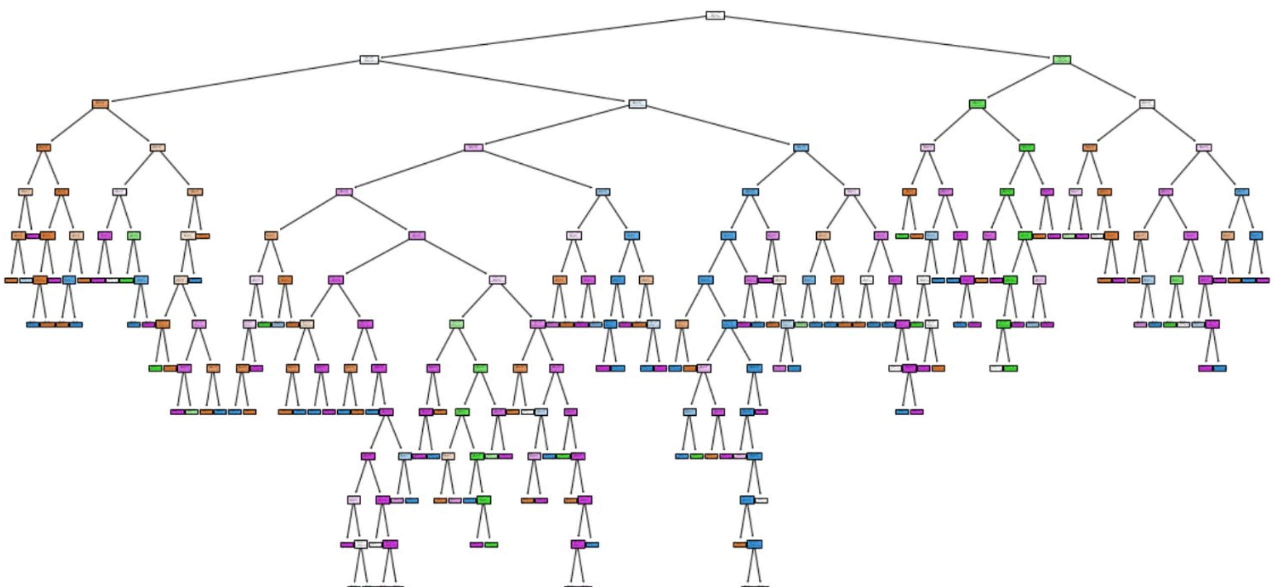
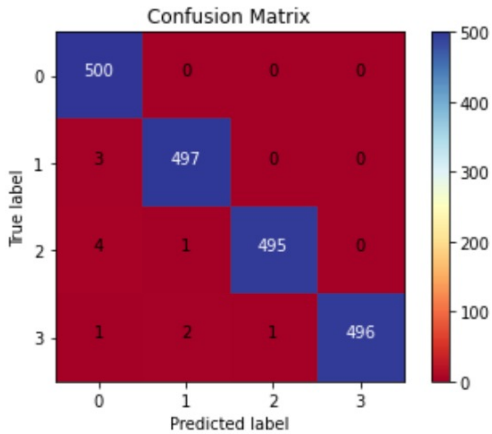
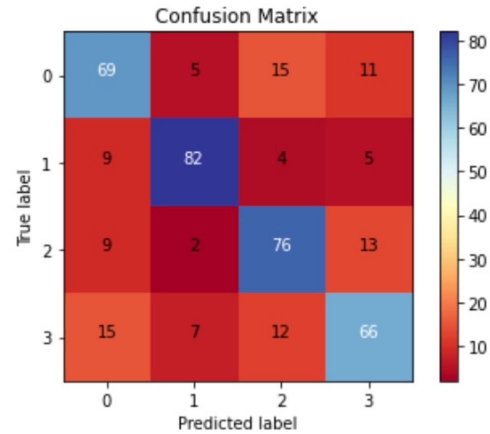


Figure 23: Best-Pruned-Tree Visualisation



(a) Training Data



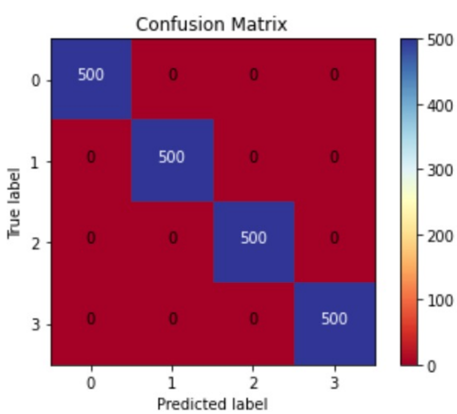
(b) Validation Data

Figure 24: Confusion Matrices of the Best Pruned Tree

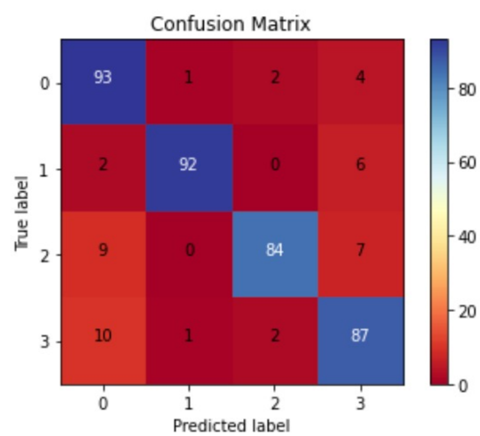
d) Random Forest

	Train	Validation
Accuracy	1.0	0.89

Table 13: Observations of Random Forest



(a) Training Data

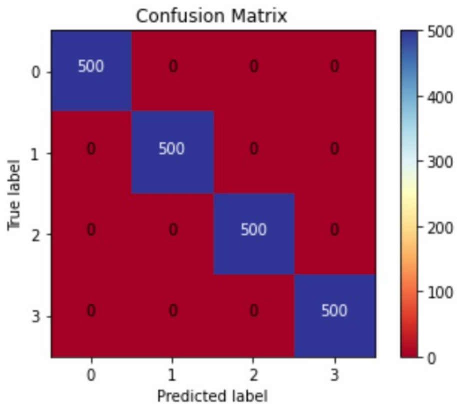


(b) Validation Data

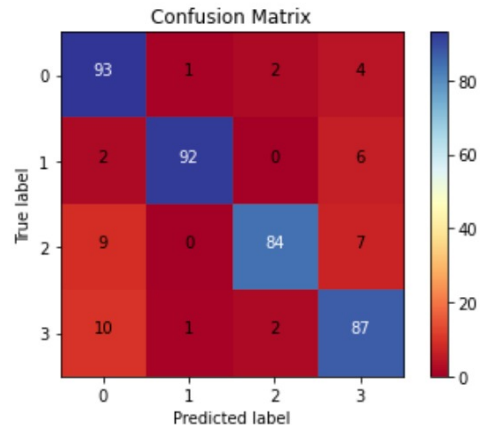
Figure 25: Confusion Matrices of Random Forest

	Train	Validation
Accuracy	1.0	0.88

Table 14: Observations of Random Forest after Grid Search



(a) **Training Data**



(b) **Validation Data**

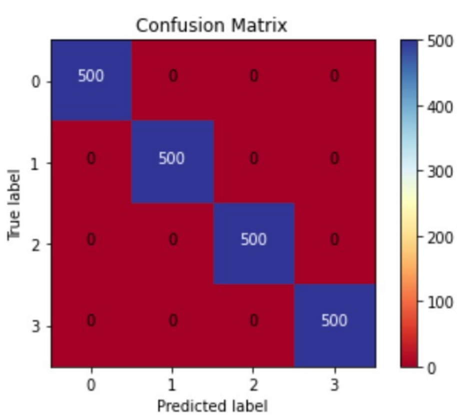
Figure 26: Confusion Matrices of Random Forest after Grid Search

Best parameters: criterion = 'entropy', max˙depth = None, min˙samples˙split = 10, n˙estimators = 200

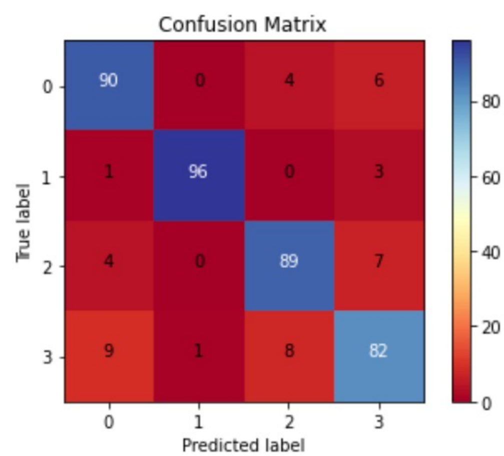
e) Gradient Boosted Trees and XGBoost

	Train	Validation
Accuracy	1.0	0.8925
Training Time	451.80 sec	

Table 15: Observations of Gradient Boosted Trees



(a) **Training Data**

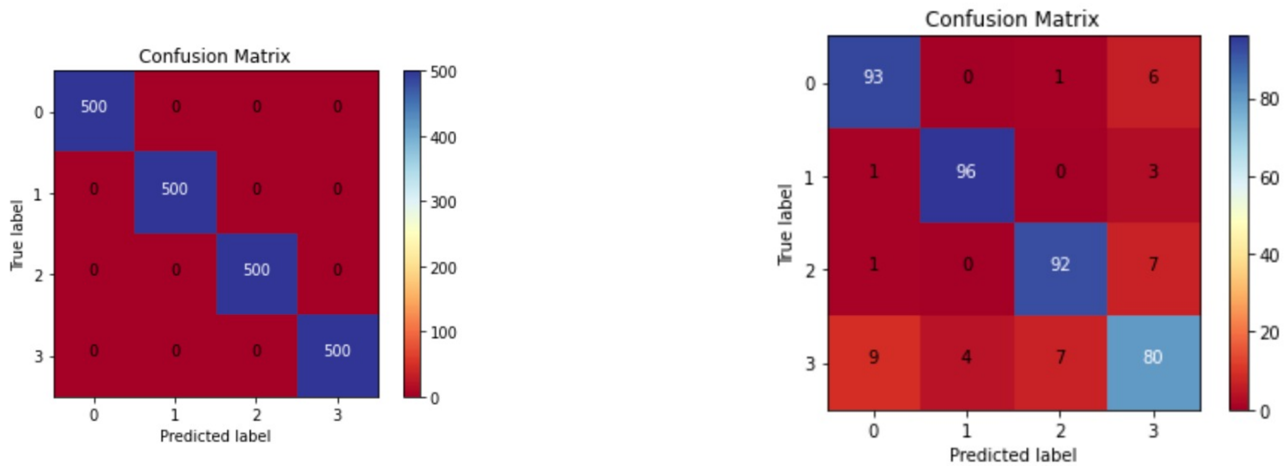


(b) **Validation Data**

Figure 27: Confusion Matrices of Gradient Boosted Trees

	Train	Validation
Accuracy	1.0	0.9025
Training Time	58.03 sec	

Table 16: Observations of XGBoost Trees

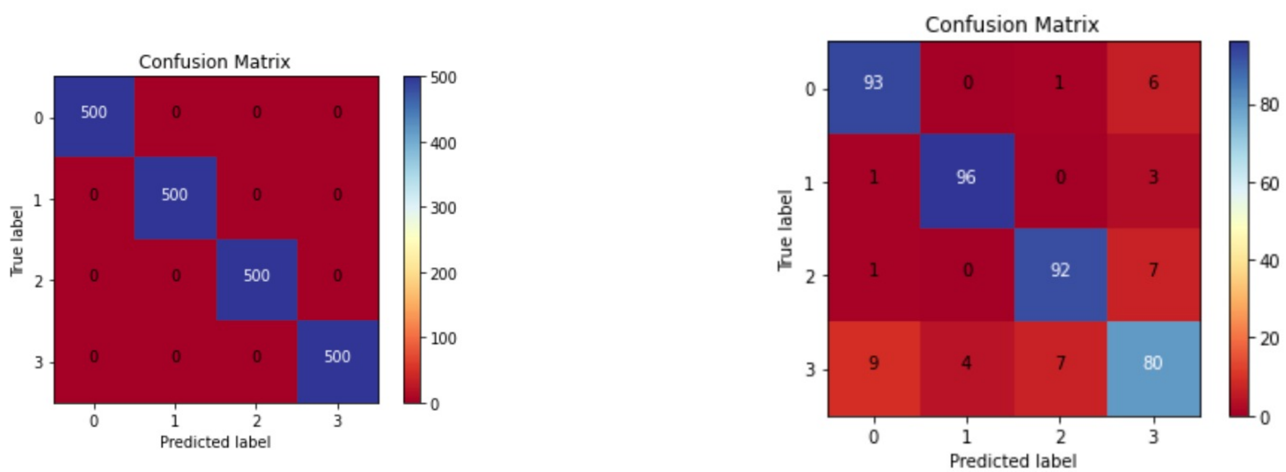


(a) Training Data **(b) Validation Data**

Figure 28: Confusion Matrices of XGBoost Trees

	Train	Validation
Accuracy	1.0	0.915

Table 17: Observations of XGBoosted Trees after Grid Search



(a) Training Data **(b) Validation Data**

Figure 29: Confusion Matrices of XGBoosted Trees after Grid Search

Best max˙depth = 7, sub˙sample = 0.6, n˙estimators = 50

f) Real-time Application

The predictions I'm getting on 10 of my pictures are - array([3, 3, 0, 0, 3, 3, 3, 1, 3, 0]) which implies 10% accuracy. Initially, I had used some pictures where I had my spectacles on and those pics were marked as 'cars' probably due to specs being considered as headlights. I removed those images and included only the ones where I didn't have my specs on. The less accuracy can also be attributed to the training data consisting of only extra-fair skinned people.