COL341: Assignment 3

Note: random\_state has been set to 0 for consistent results

# Single-state

Faces at 1, Rest at 0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **.** | **3.1A (IG)** | **3.1A (GINI)** | **3.1B (GINI)** | **3.1B (IG)** |
| Training Time | 139.0977933 | 162.330637 | 3.567991257 | 2.245169163 |
|  |  | On | Train |  |
| Accuracy | 0.8555 | 0.8635 | 0.9885 | 0.999 |
| Precision of state 1 | 0.903935957 | 0.897602074 | 0.988749173 | 1 |
| Precision of state 0 | 0.710578842 | 0.748358862 | 0.987730061 | 0.996015936 |
| Recall of state 1 | 0.903333333 | 0.923333333 | 0.996 | 0.998666667 |
| Recall of state 0 | 0.712 | 0.684 | 0.966 | 1 |
|  |  | On | Validation |  |
| Accuracy | 0.825 | 0.8175 | 0.9275 | 0.9425 |
| Precision of state 1 | 0.888513514 | 0.874587459 | 0.935691318 | 0.969491525 |
| Precision of state 0 | 0.644230769 | 0.639175258 | 0.898876404 | 0.866666667 |
| Recall of state 1 | 0.876666667 | 0.883333333 | 0.97 | 0.953333333 |
| Recall of state 0 | 0.67 | 0.62 | 0.8 | 0.91 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **.** | **3.1C (Default)** | **3.1C (GridSearch)** | **3.1D (BestPruned)** | **3.1E (Default)** | **3.1E (GridSearch)** |
| Training Time | 0.015074968 | 0.01129365 | - | 5.681501865 | 6.592205048 |
|  |  | On | Train |  |  |
| Accuracy | 0.9295 | 0.8775 | 0.9755 | 1 | 1 |
| Precision of state 1 | 0.974842767 | 0.928327645 | 0.972638436 | 1 | 1 |
| Precision of state 0 | 0.815465729 | 0.738317757 | 0.984946237 | 1 | 1 |
| Recall of state 1 | 0.93 | 0.906666667 | 0.995333333 | 1 | 1 |
| Recall of state 0 | 0.928 | 0.79 | 0.916 | 1 | 1 |
|  |  | On | Validation |  |  |
| Accuracy | 0.885 | 0.8775 | 0.935 | 0.9725 | 0.9825 |
| Precision of state 1 | 0.934931507 | 0.93728223 | 0.936305732 | 0.964630225 | 0.977198697 |
| Precision of state 0 | 0.75 | 0.725663717 | 0.930232558 | 1 | 1 |
| Recall of state 1 | 0.91 | 0.896666667 | 0.98 | 1 | 1 |
| Recall of state 0 | 0.81 | 0.82 | 0.8 | 0.89 | 0.93 |

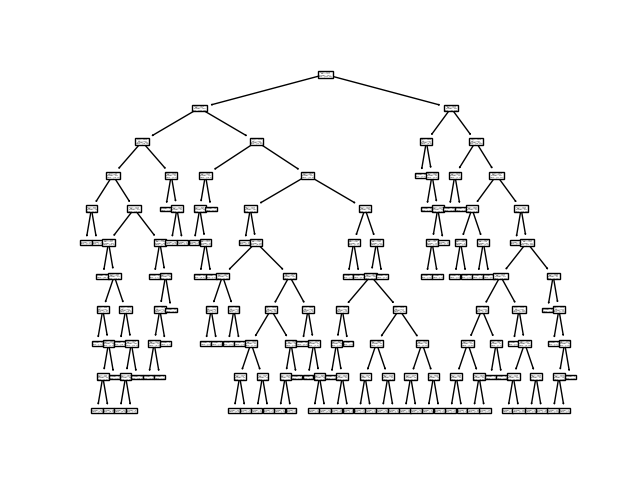
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **.** | **3.1F (Gradient)** | **3.1F (GradientGS)** | **3.1F (XGBoost)** | **3.1F (XGBoostGS)** |
| Training Time | 116.5799448 | Too long\* | 5.303102255 | 4.706798553 |
|  |  | On | Train |  |
| Accuracy | 1 | - | 1 | 1 |
| Precision of state 1 | 1 | - | 1 | 1 |
| Precision of state 0 | 1 | - | 1 | 1 |
| Recall of state 1 | 1 | - | 1 | 1 |
| Recall of state 0 | 1 | - | 1 | 1 |
|  |  | On | Validation |  |
| Accuracy | 0.98 | - | 0.9875 | 0.9775 |
| Precision of state 1 | 0.980263158 | - | 0.983606557 | 0.97704918 |
| Precision of state 0 | 0.979166667 | - | 1 | 0.978947368 |
| Recall of state 1 | 0.993333333 | - | 1 | 0.993333333 |
| Recall of state 0 | 0.94 | - | 0.95 | 0.93 |

\*- Insufficient processing power to compute under 8 hours

# Analysis of Sections

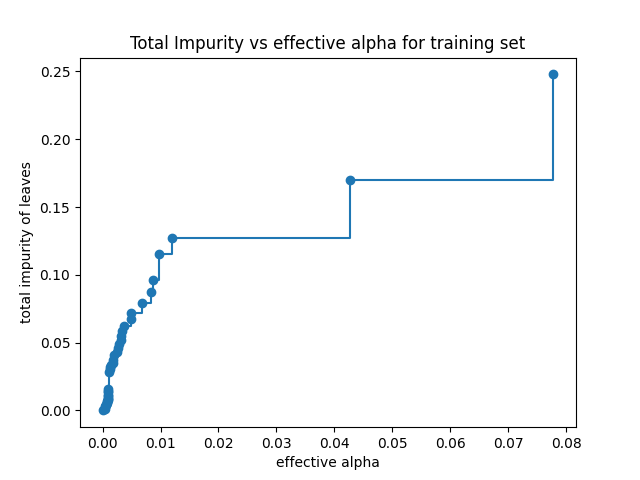
## 3.1c

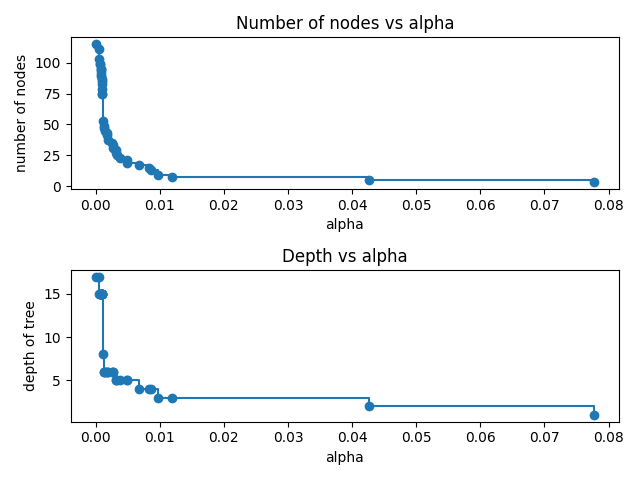
Visualised tree for SelectKBest

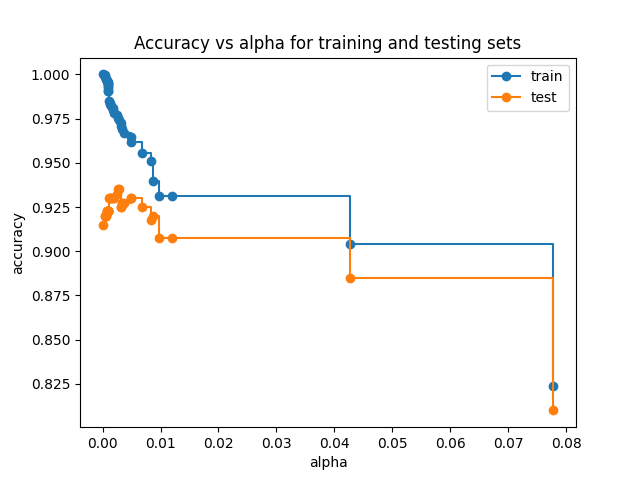


## 3.1d

Required graphs





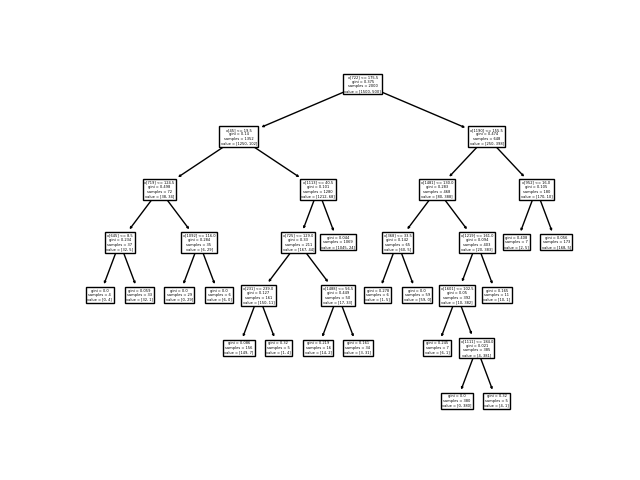


On training data, accuracy decreases with increase in alpha. We cannot conclude anything substantial from this. However, on testing data, accuracy first increases then decreases.

This clearly shows that post-pruning using ccp\_alpha can correct overfitting to some extent (for ideal selection of ccp\_alpha), however on increasing beyond this point accuracy is compromised.

ccp\_alpha controls a trade-off between complexity of a subtree and how well it fits to training data. With increase in alpha, effect of complexity becomes more prominent and in turn tree is pruned, thus reducing accuracy on training data.

Cost function is penalised by adding a term + ccp\_alpha\*|Complexity measure of tree|, similar to addition of penalty in linear regression to make ridge regression.

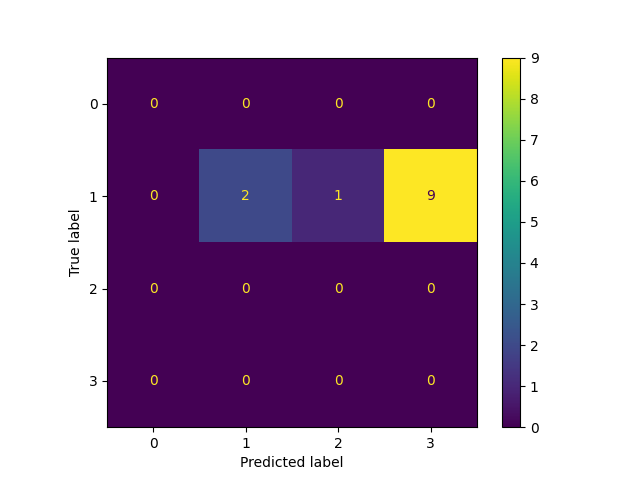


Visualisation of best-pruned tree tree on validation split

## 3.2g

Model used: XGBClassifier() with best parameters as learned by Grid-Search

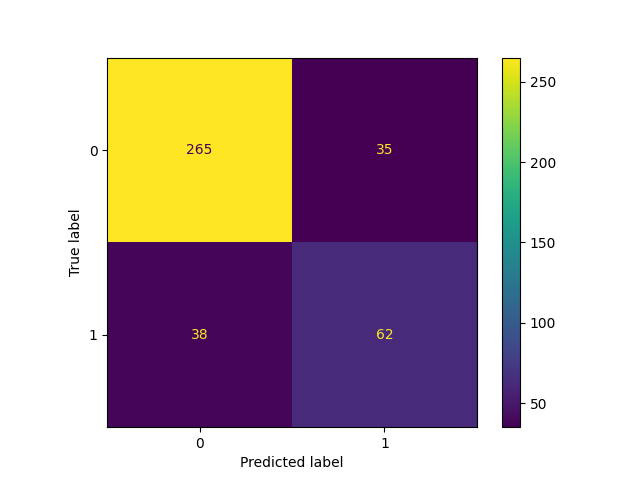
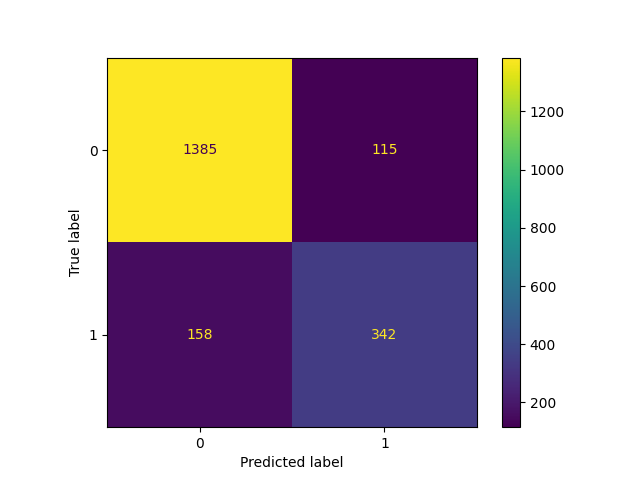
Images are variations of completely in frame, zoomed or misaligned (9 of my own, 3 of another).

Despite good performance of classifier on validation set, it classifies these images poorly (2 correctly, 1 incorrectly as airplane, 9 incorrectly as dogs). This could be due to bias in the selection of images for training and validation purposes.

# Confusion Matrix and Parameters (3.1g)

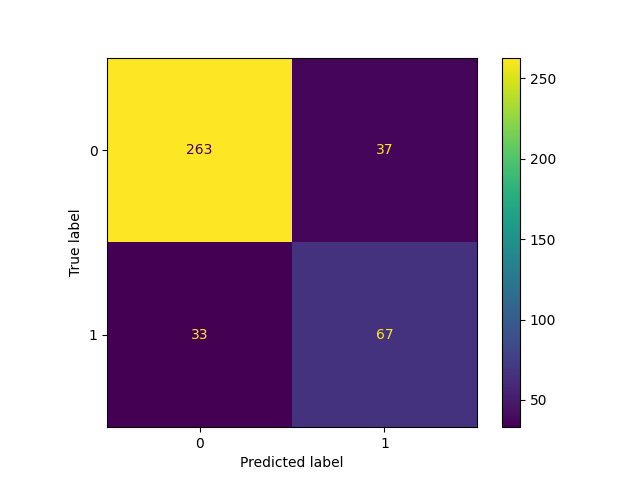
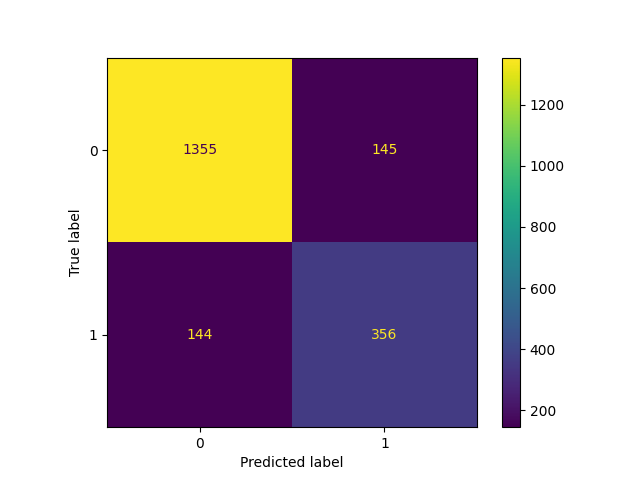
* Decision tree from scratch (3.1a)

GINI Index



Confusion Matrix on Train data Confusion Matrix on Validation data

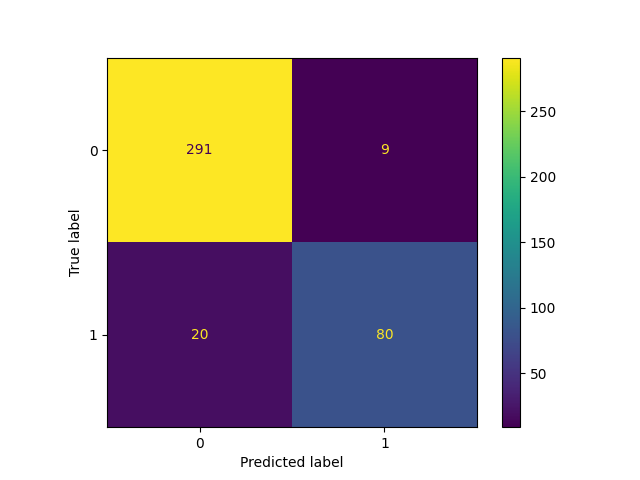
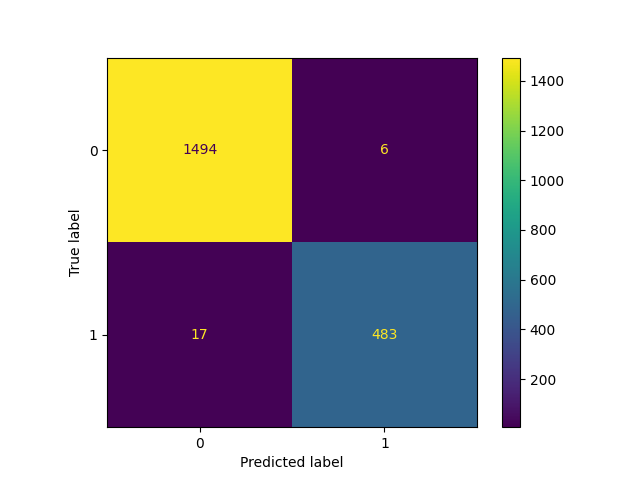
IG



Confusion Matrix on Train data Confusion Matrix on Validation data

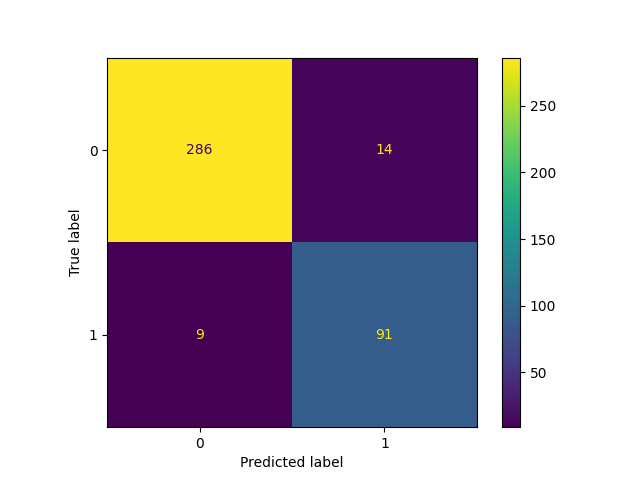
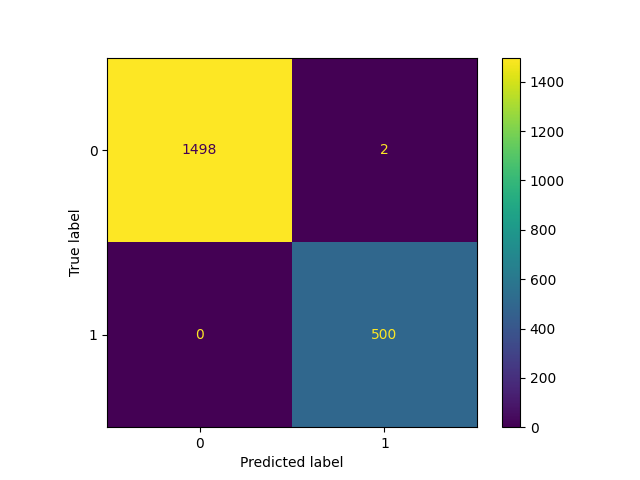
* Decision Tree sklearn (3.1b)

GINI Index



Confusion Matrix on Train data Confusion Matrix on Validation data

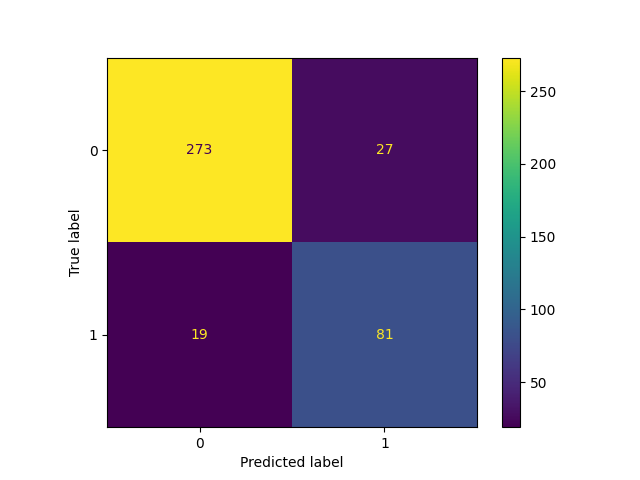
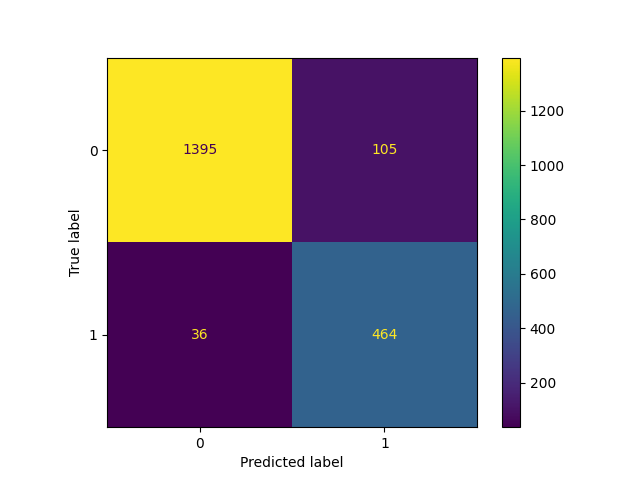
IG



Confusion Matrix on Train data Confusion Matrix on Validation data

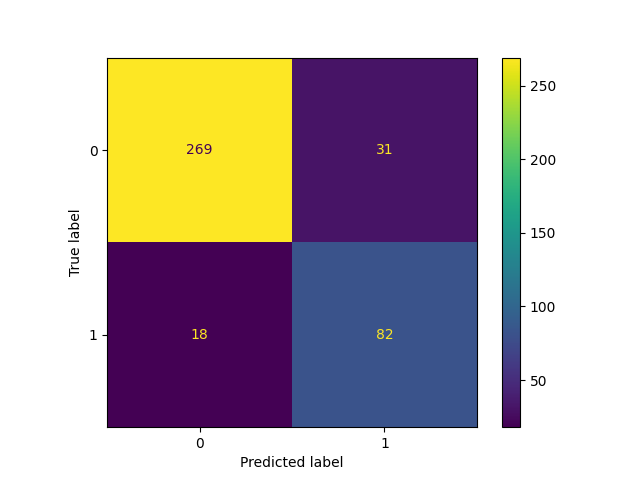
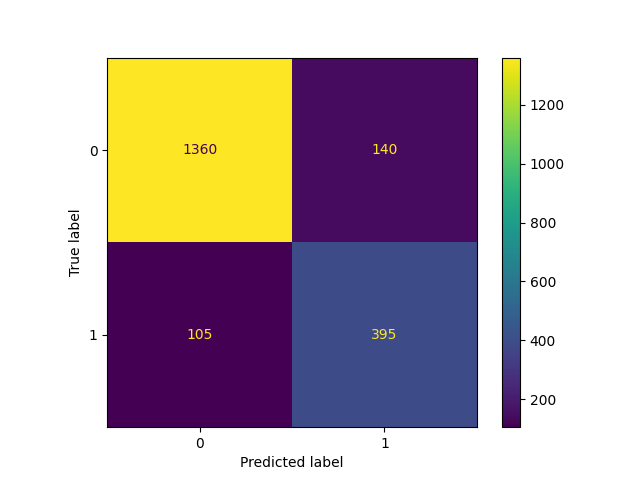
* Decision Tree Grid-Search (3.1c)

Default



Confusion Matrix on Train data Confusion Matrix on Validation data

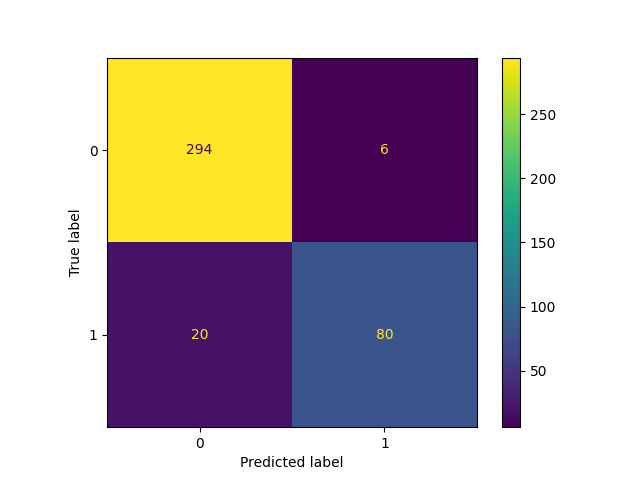
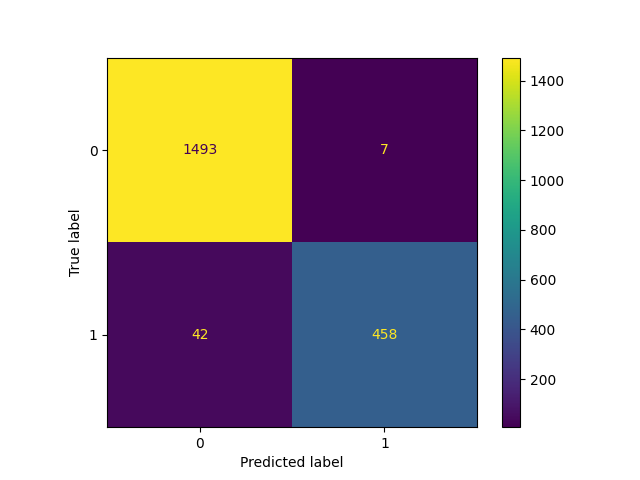
Best parameters: {'criterion': 'entropy', 'max\_depth': 5, 'min\_samples\_split': 4}



Confusion Matrix on Train data Confusion Matrix on Validation data

* Decision Tree Post Pruning with Cost Complexity Pruning (3.1d)

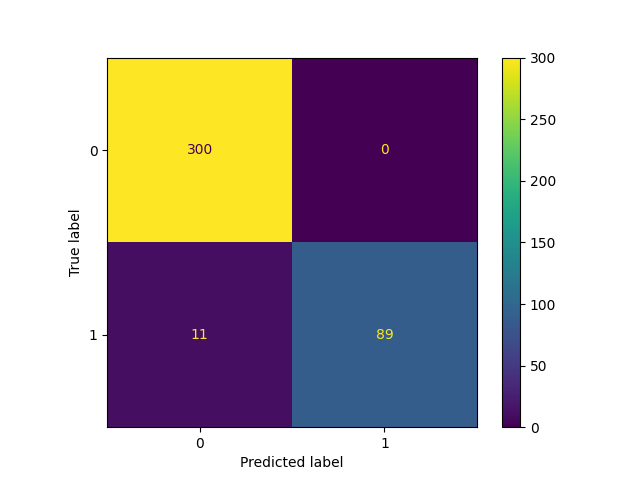
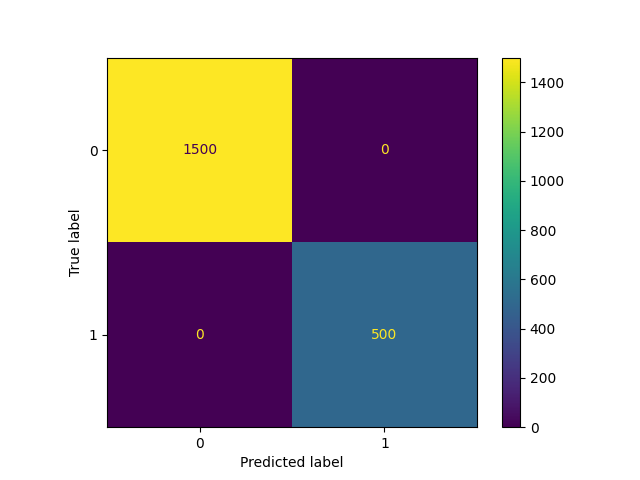
Best-performing tree on validation split



Confusion Matrix on Train data Confusion Matrix on Validation data

* Random forests (3.1e)

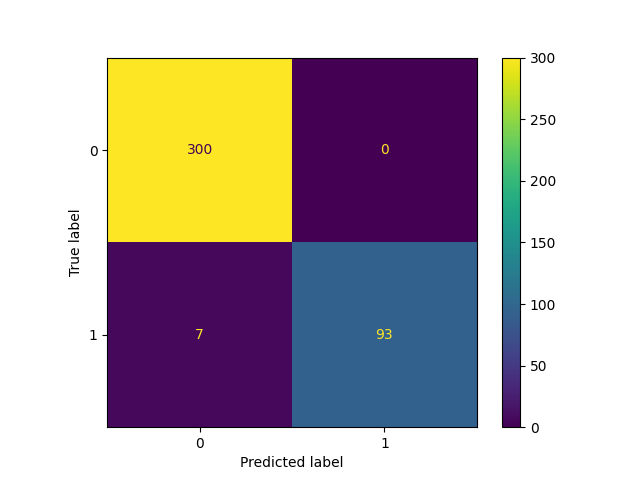
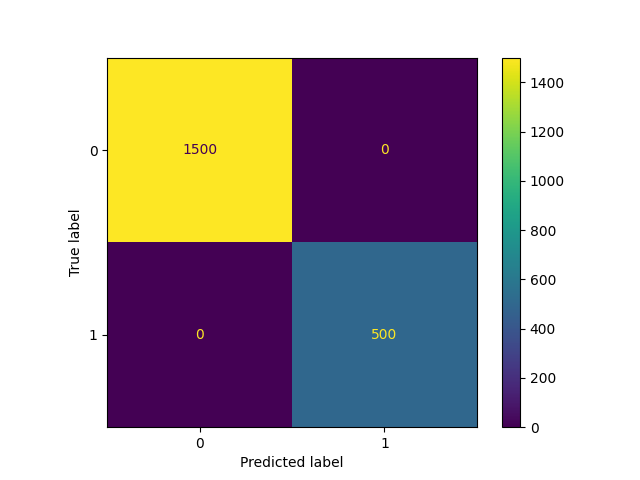
Default



Confusion Matrix on Train data Confusion Matrix on Validation data

Grid-Search Best Parameters

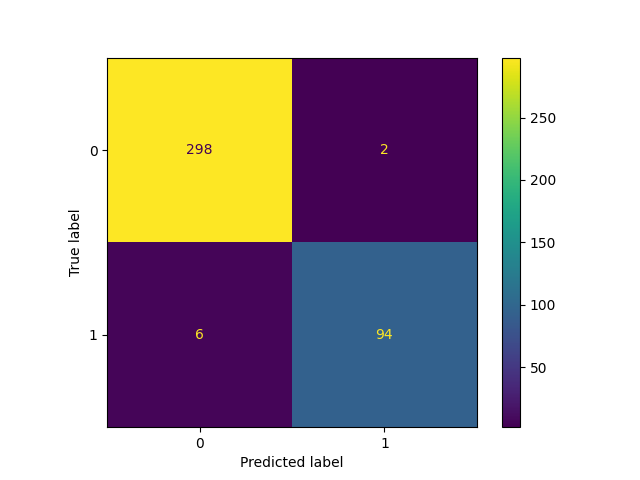
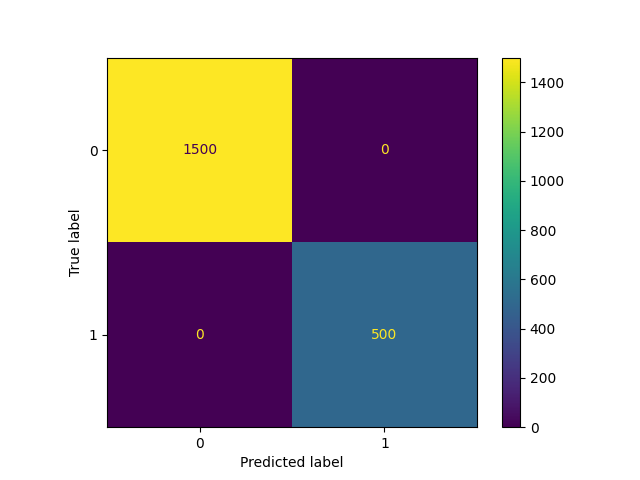
Best parameters: {'criterion': 'entropy', 'max\_depth': None, 'min\_samples\_split': 7, 'n\_estimators': 150}



Confusion Matrix on Train data Confusion Matrix on Validation data

* Gradient Boosted Trees and XGBoost (3.1f)

Gradient Boosted with Default

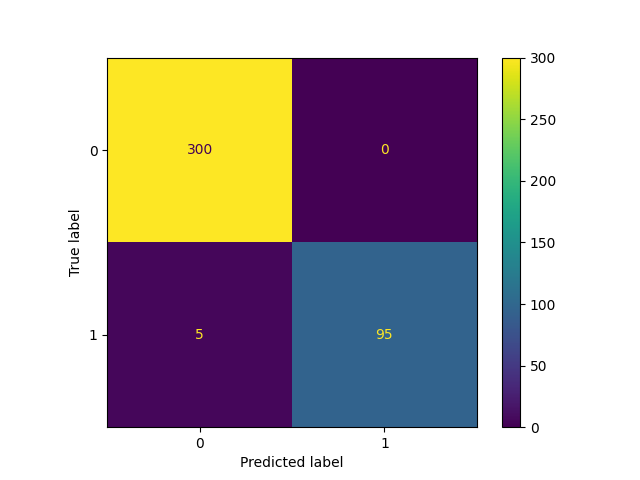
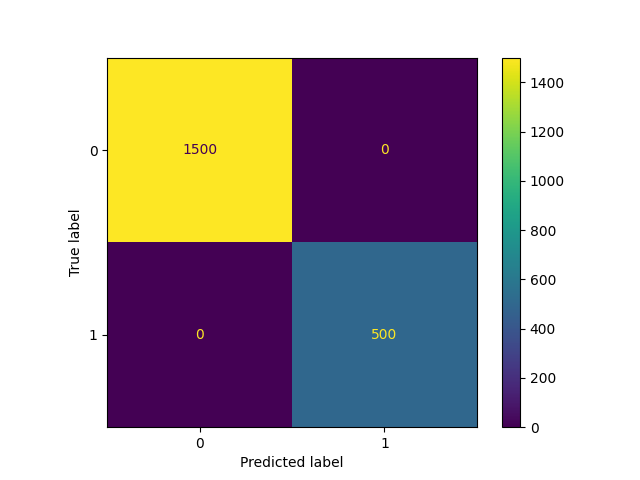


Confusion Matrix on Train data Confusion Matrix on Validation data

Gradient Boosted with Grid-Search Best Parameters

Note: Unable to complete regular, took ~8 hours.

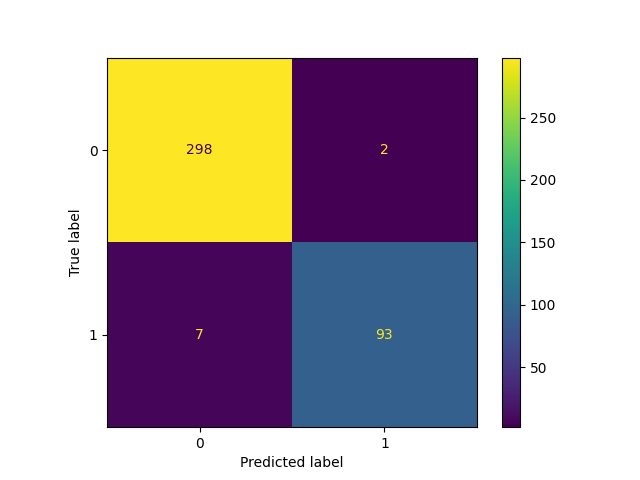
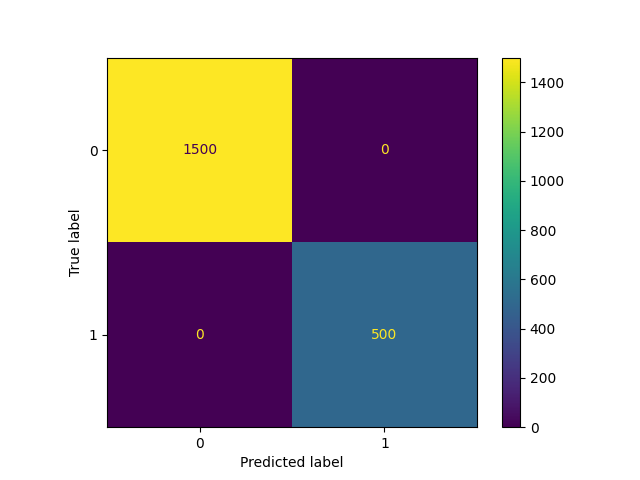
XGBoost with Default



Confusion Matrix on Train data Confusion Matrix on Validation data

XGBoost with Grid-Search Best Parameters

Best parameters: {'max\_depth': 6, 'n\_estimators': 40, 'subsample': 0.6}



Confusion Matrix on Train data Confusion Matrix on Validation data

# Multi-state

Cars at 0, Faces at 1, Airplanes at 2, Dogs at 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **.** | **3.2A (GINI)** | **3.2A (IG)** | **3.2B (Default)** | **3.2B (GridSearch)** |
| Training Time | 3.27023983 | 4.622252703 | 0.013673067 | 0.014748335 |
|  |  | On | Train |  |
| Accuracy | 0.969 | 0.971 | 0.8105 | 0.669 |
|  |  | On | Validation |  |
| Accuracy | 0.7425 | 0.7225 | 0.6325 | 0.6025 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **.** | **3.2C (BestPruned)** | **3.2D (Default)** | **3.2D (GridSearch)** | **3.2E (Gradient)** |
| Training Time | - | 9.960266113 | 9.01617837 | 704.1730013 |
|  | On | Train |  |  |
| Accuracy | 0.957 | 1 | 0.9985 | 1 |
|  | On | Validation |  |  |
| Accuracy | 0.7425 | 0.8725 | 0.8775 | 0.8925 |

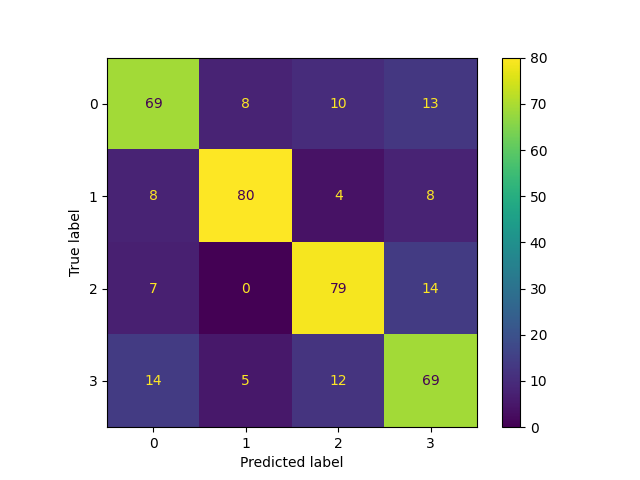
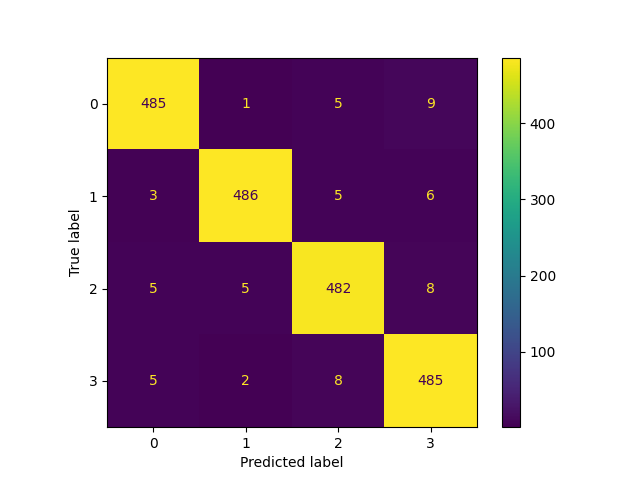
|  |  |  |  |
| --- | --- | --- | --- |
| **.** | **3.2E (GradientGS)** | **3.2E (XGBoost)** | **3.2E (XGBoostGS)** |
| Training Time | Too long\* | 28.95156932 | 27.03962278 |
|  | On | Train |  |
| Accuracy | - | 1 | 1 |
|  | On | Validation |  |
| Accuracy | - | 0.9025 | 0.91 |

\*- Insufficient processing power to compute under 8 hours

# Confusion Matrix and Parameters (3.2f)

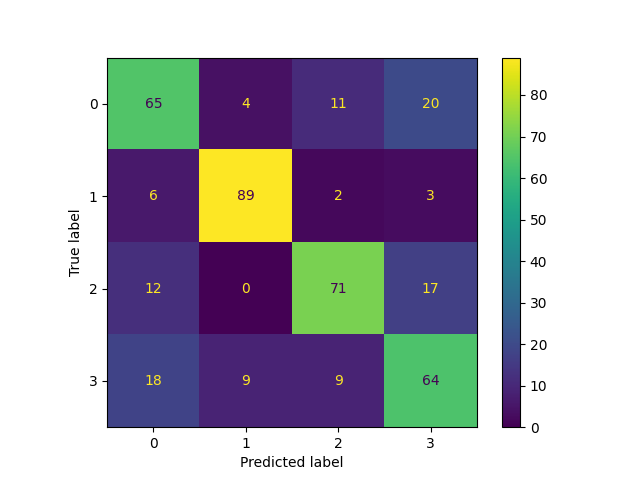
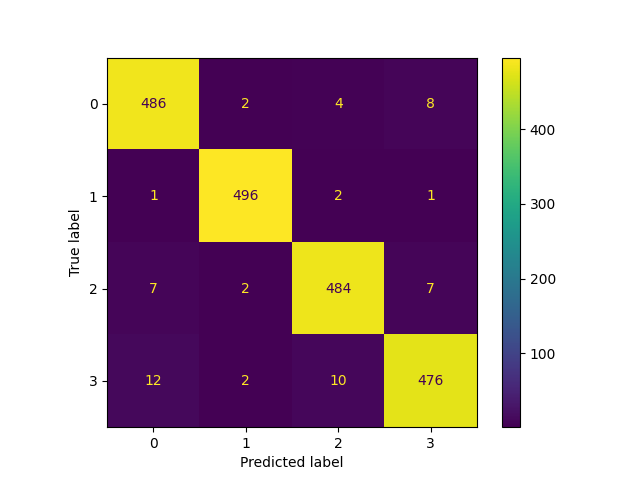
* Decision Tree sklearn (3.2a)

GINI Index



Confusion Matrix on Train data Confusion Matrix on Validation data

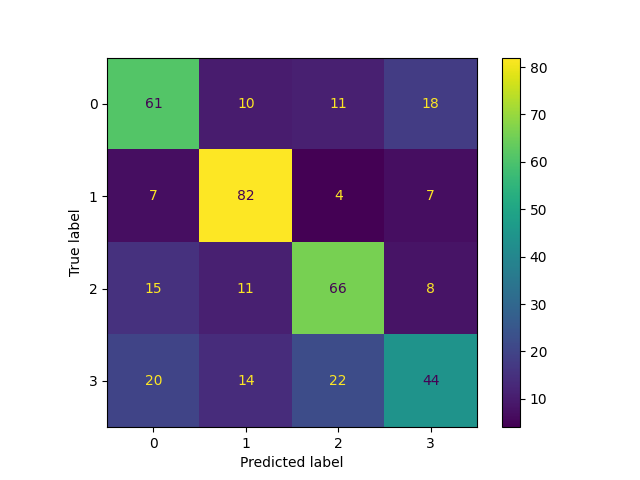
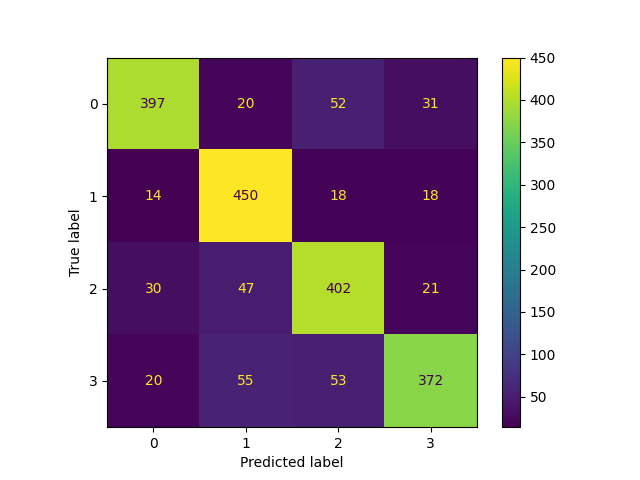
IG



Confusion Matrix on Train data Confusion Matrix on Validation data

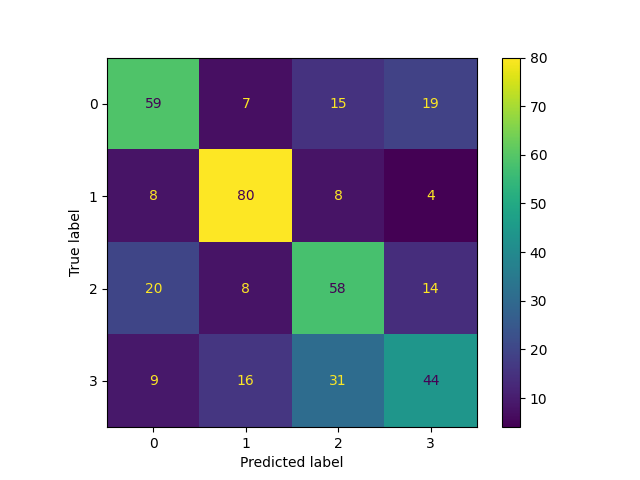
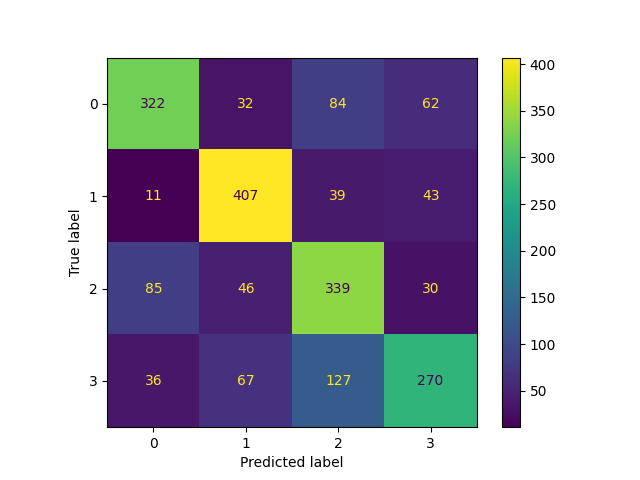
* Decision Tree Grid-Search (3.2b)

Default:



Confusion Matrix on Train data Confusion Matrix on Validation data

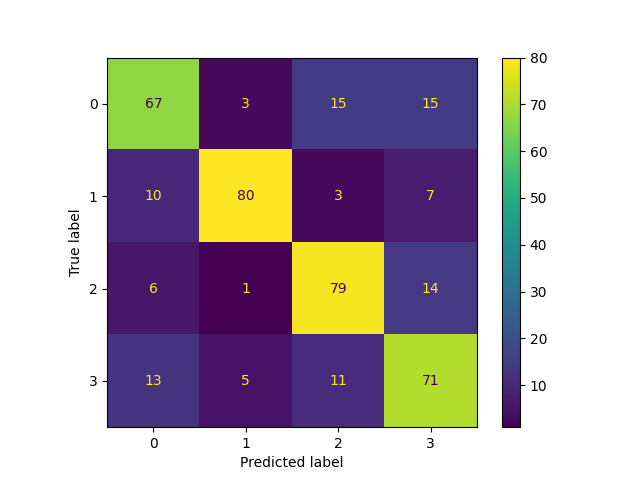
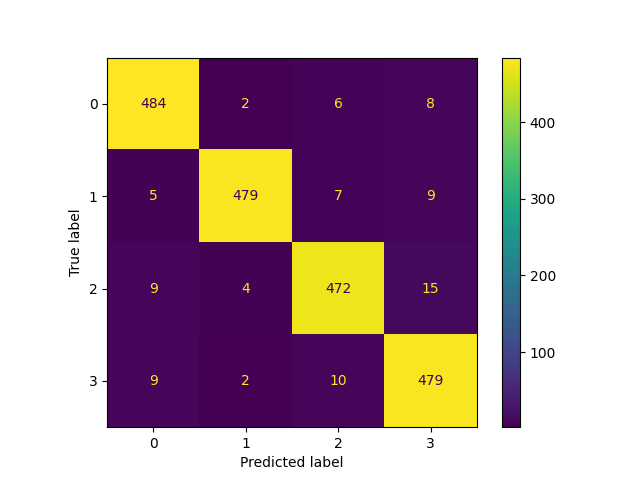
Best parameters: {'criterion': 'entropy', 'max\_depth': 5, 'min\_samples\_split': 4}



Confusion Matrix on Train data Confusion Matrix on Validation data

* Decision Tree Post Pruning with Cost Complexity Pruning (3.2c)

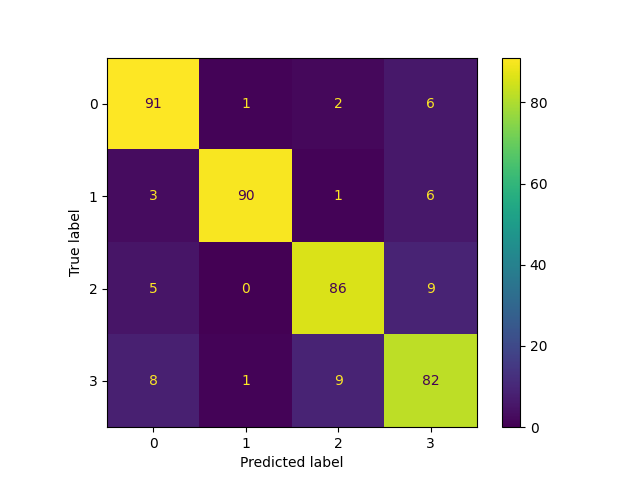
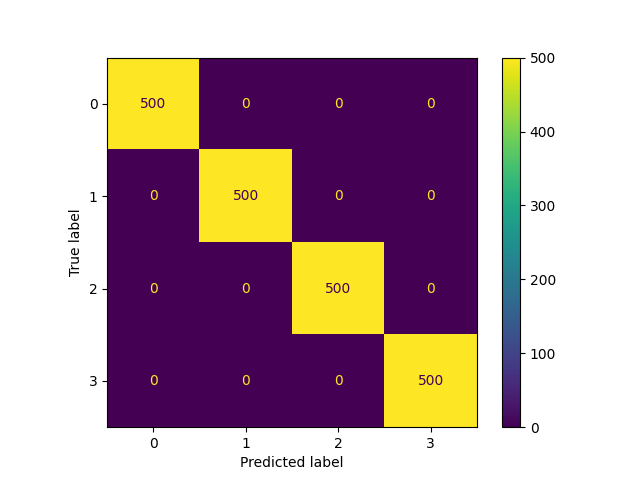
Best-performing tree



Confusion Matrix on Train data Confusion Matrix on Validation data

* Random forests (3.2d)

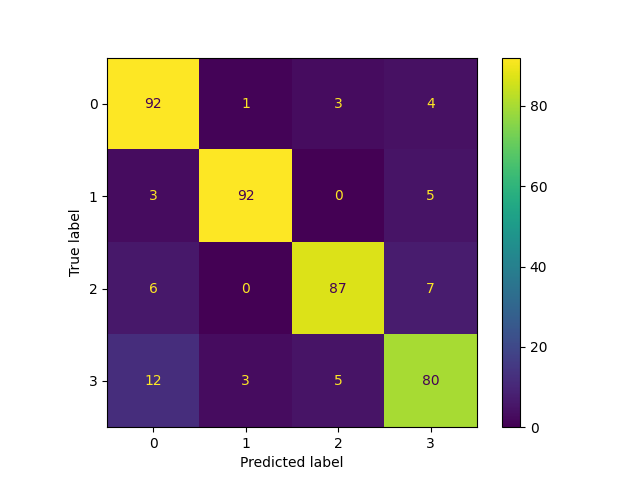
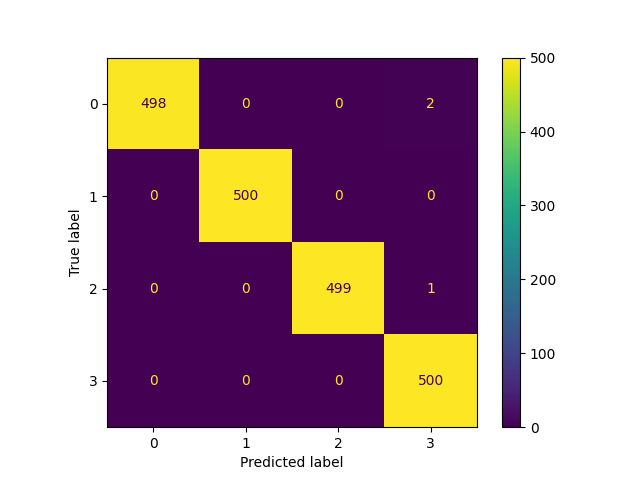
Default



Confusion Matrix on Train data Confusion Matrix on Validation data

Grid-Search Best Parameters

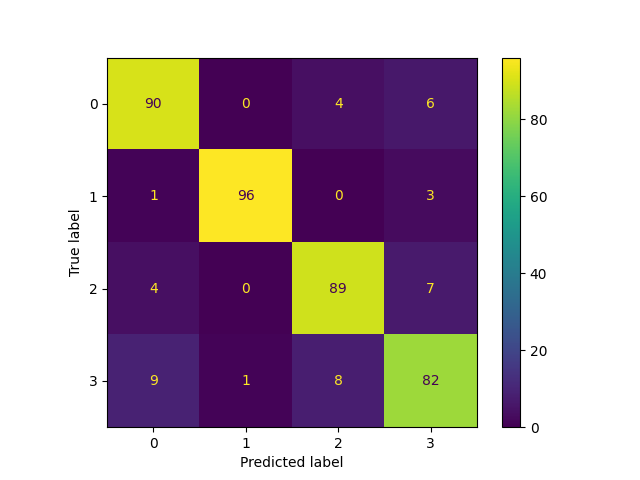
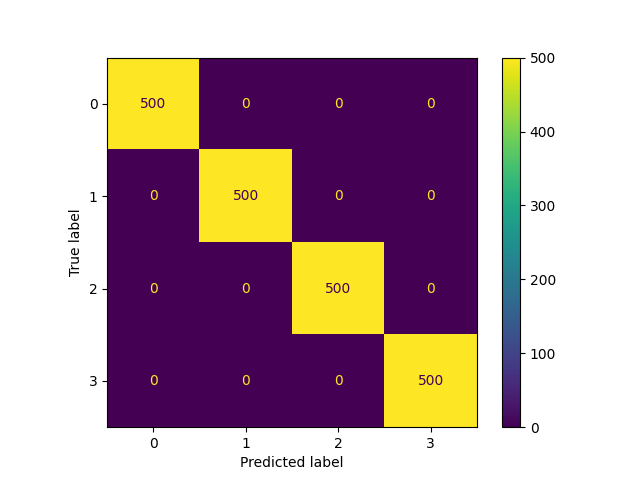
Best parameters: {'criterion': 'entropy', 'max\_depth': 10, 'min\_samples\_split': 10, 'n\_estimators': 100}



Confusion Matrix on Train data Confusion Matrix on Validation data

* Gradient Boosted Trees and XGBoost (3.2e)

Default

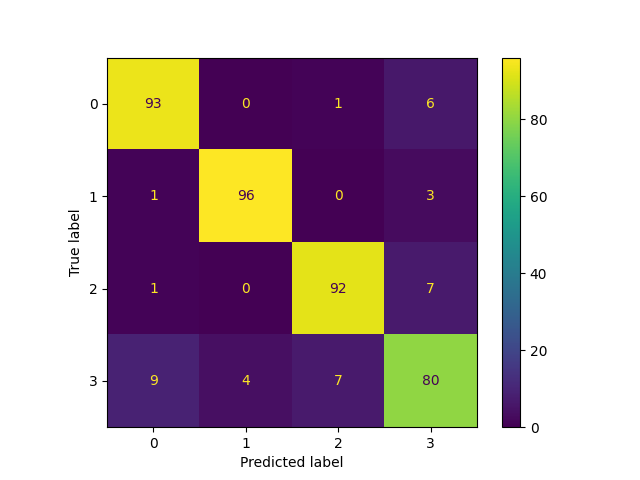
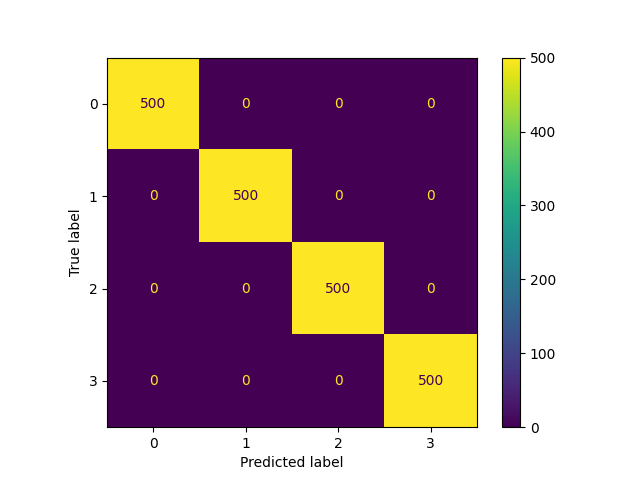


Confusion Matrix on Train data Confusion Matrix on Validation data

Gradient Boosted with Grid-Search Best Parameters

Note: Unable to complete, took ~8 hours

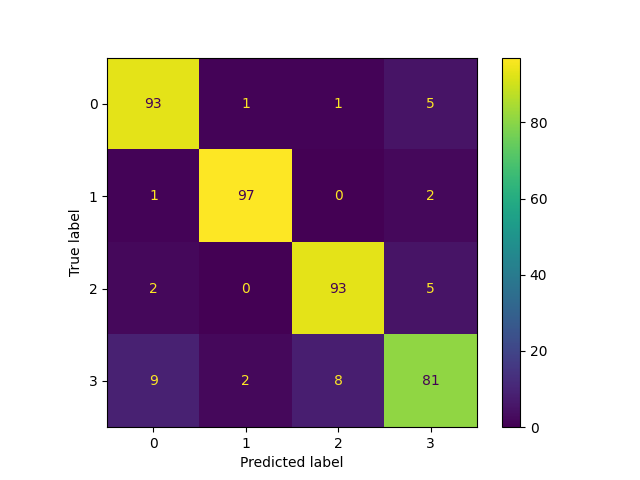
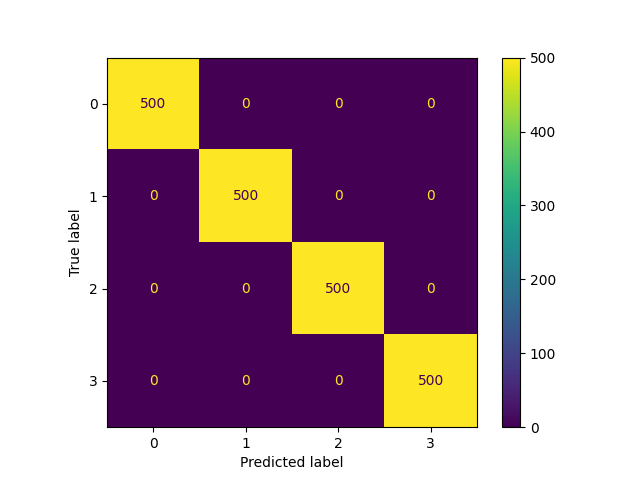
Default



Confusion Matrix on Train data Confusion Matrix on Validation data

XGBoost with Grid-Search Best Parameters

Best parameters: {'max\_depth': 10, 'n\_estimators': 50, 'subsample': 0.6}



Confusion Matrix on Train data Confusion Matrix on Validation data

# Glossary