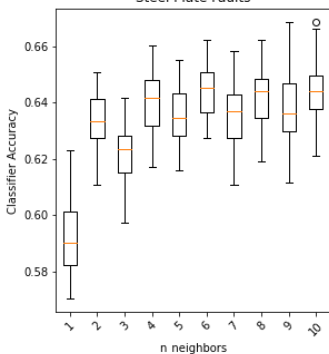
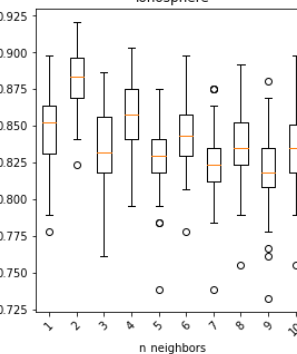
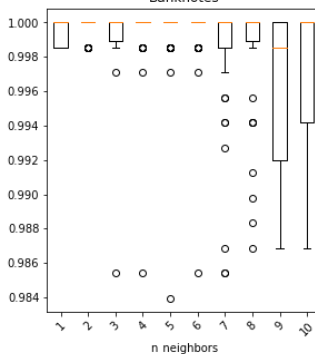
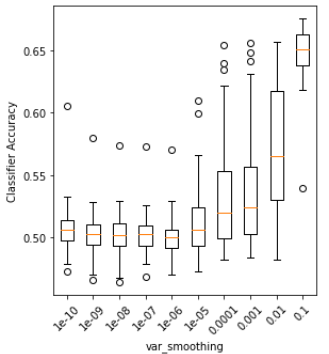
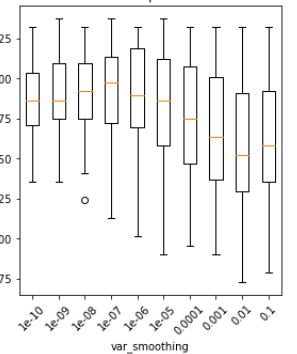
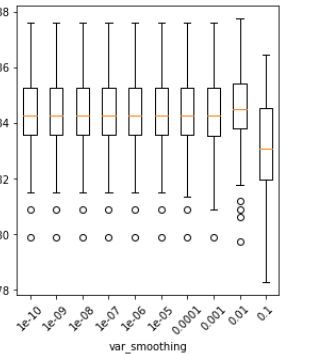
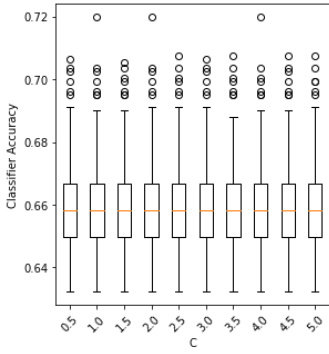
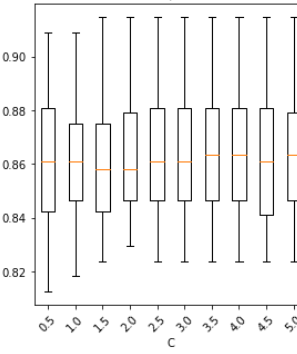
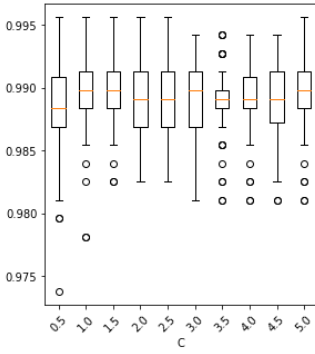
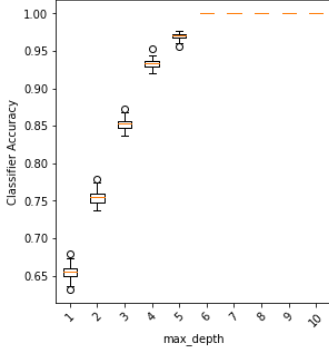
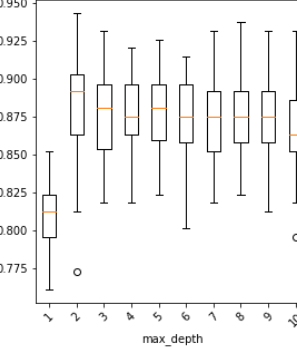
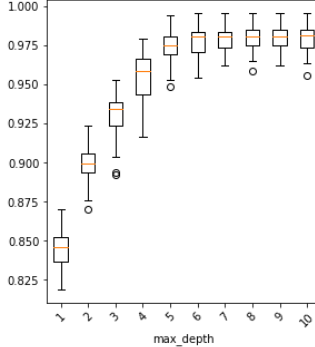
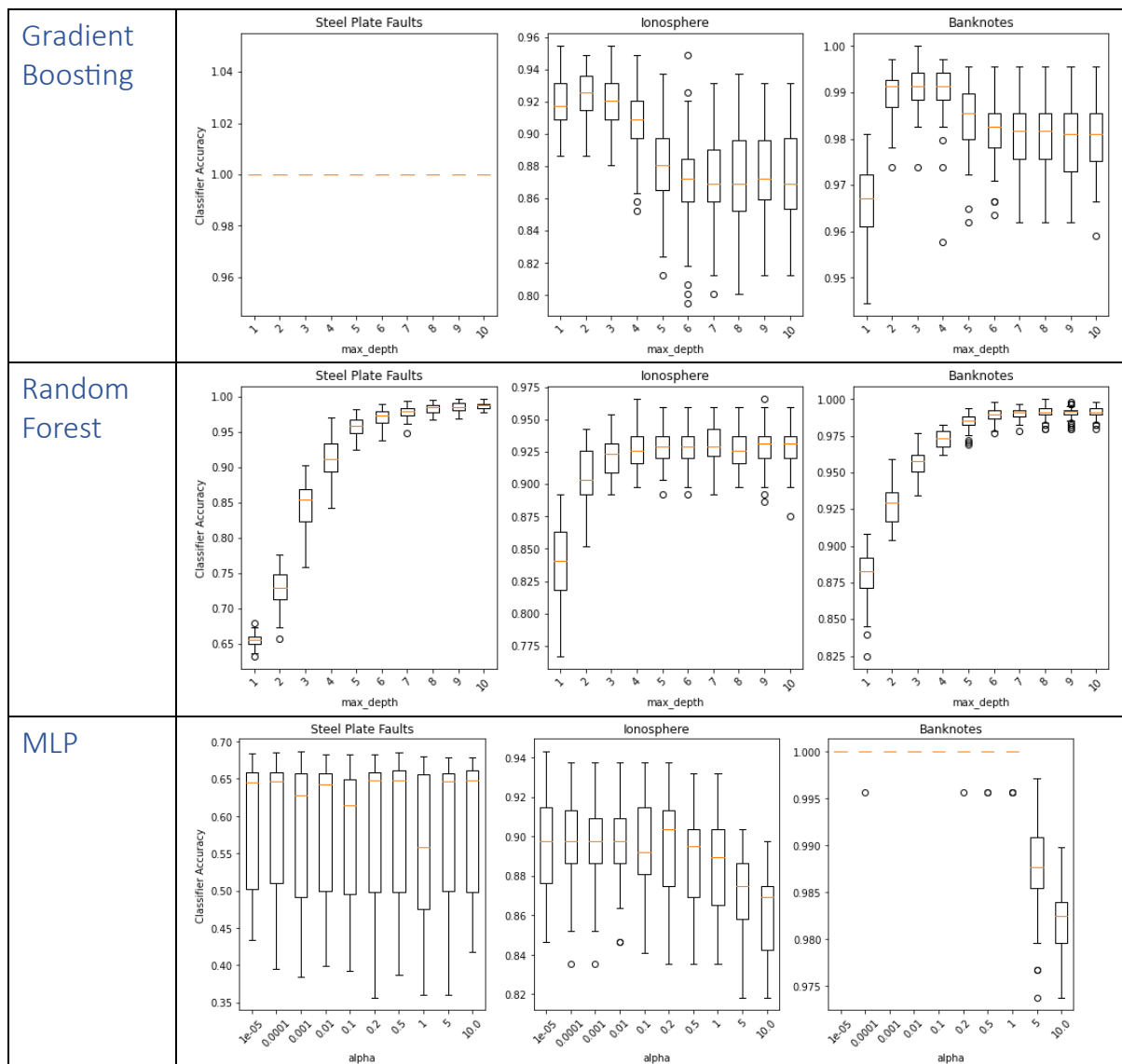


COMP309 Assignment 1

Part 1 – Classifiers

| | |
|---------------------|--|
| KNN |    |
| Naïve Bayes |    |
| Logistic Regression |    |
| Decision Tree |    |



Best Mean Value of Test Errors

| | Steel Plate Faults | Ionosphere | Banknotes |
|----------------------------|--------------------|------------|-----------|
| KNN | 0.6439 | 0.8806 | 0.9998 |
| Naïve Bayes | 0.6484 | 0.8911 | 0.8442 |
| Logistic Regression | 0.6627 | 0.8627 | 0.9893 |
| Decision Tree | 1.0 | 0.8807 | 0.9794 |
| Gradient Boosting | 1.0 | 0.9241 | 0.9909 |
| Random Forest | 0.9867 | 0.9300 | 0.9915 |
| MLP | 0.6027 | 0.9000 | 1.0 |

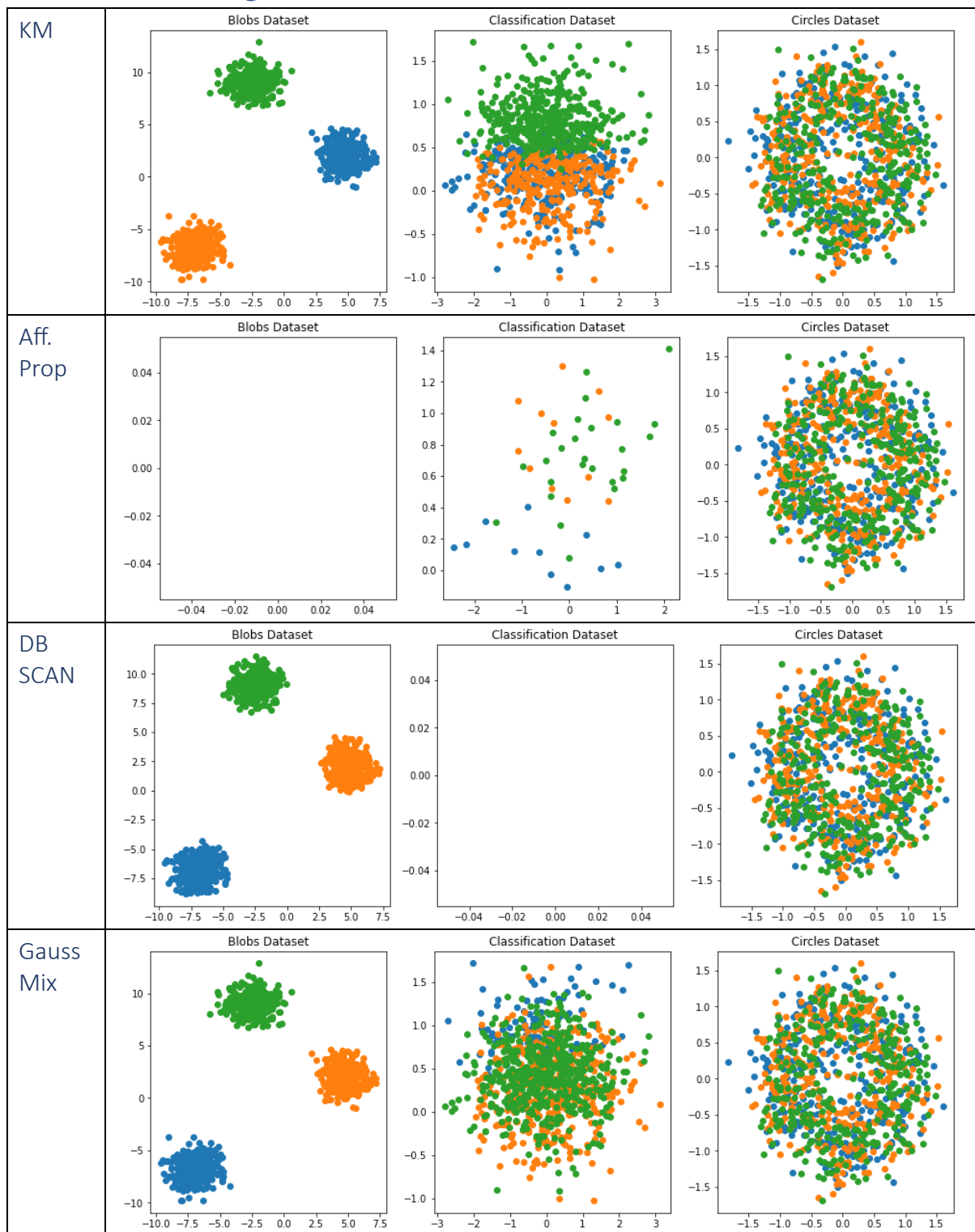
Hyperparameter Values for Best Test Errors

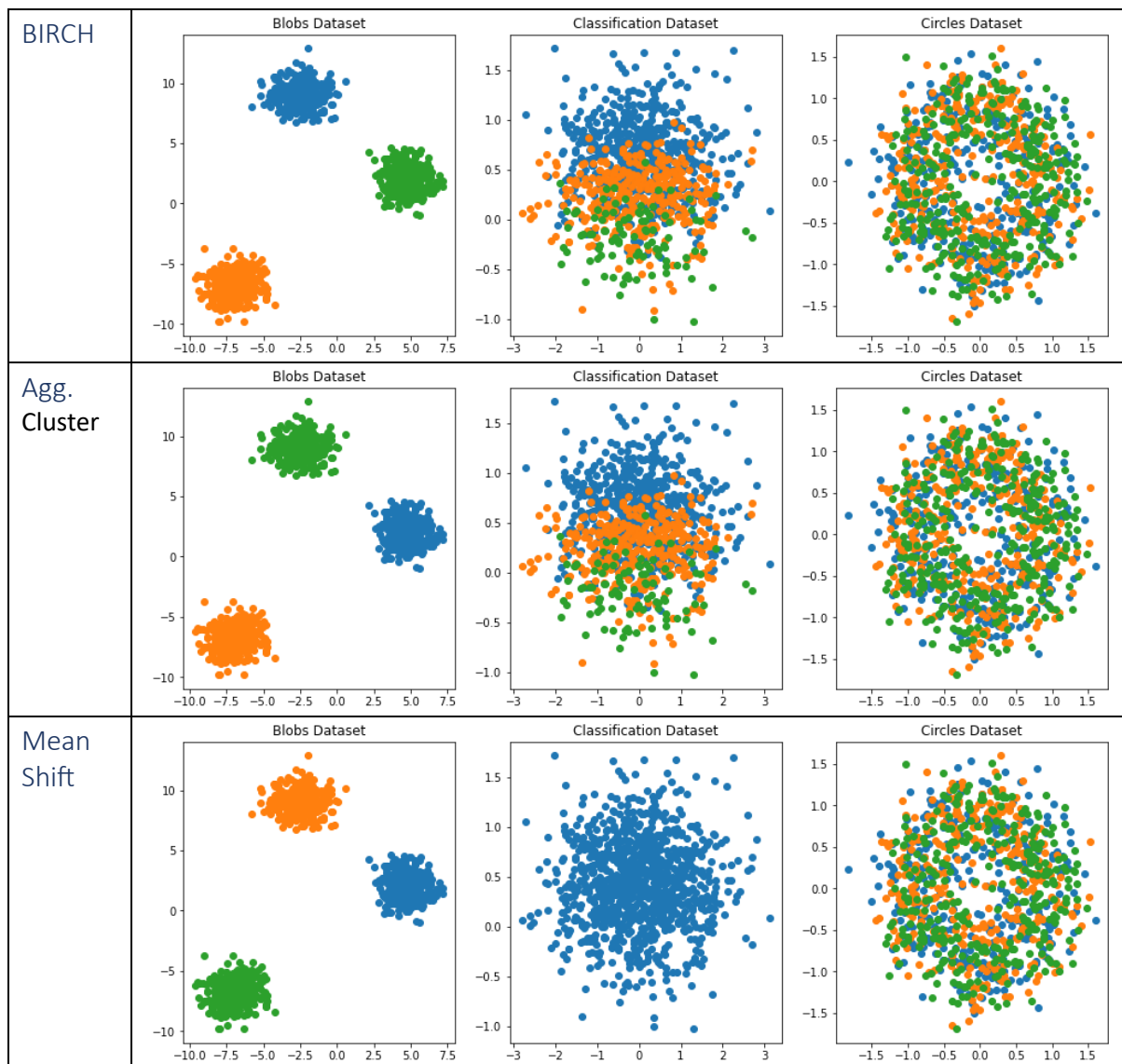
| | Steel Plate Faults | Ionosphere | Banknotes |
|---------------------------------------|------------------------------------|------------|-------------------------------|
| KNN (n_neighbours=) | 6 | 2 | 2 |
| Naïve Bayes (var_smoothing=) | 1e-1 | 1e-7 | 1e-2 |
| Logistic Regression (C=) | 1.0 | 5.0 | 1.0 |
| Decision Tree (max_depth=) | >= 6 | 2 | 9 |
| Gradient Boosting (max_depth=) | Inputs 1-10 (all values identical) | 2 | 3 |
| Random Forest (max_depth=) | 10 | 10 | 10 |
| MLP (alpha=) | 1e-4 | 1e-4 | 1e-05, 1e-3, 1e-2, 2e-1, 5e-1 |

The models which consistently had the best performance were the hierarchical clustering models – the decision tree, gradient boosting, and random forest classifiers. Not only did they have consistent high accuracy across all three datasets, but they displayed a clear trend of increasing accuracy with the control parameter. By eye, the relationship appears to be exponentially decaying towards an asymptote of full accuracy. Gradient boosting displayed a strange error of perfectly predicting the steel plates data across all control parameters. This is possibly due to rounding errors and the model was accurate for the dataset, which is a well behaved, labelled set with strong patterns and no missing values.

The model's accuracy which was most sensitive to the control parameter was KNN, which displays an erratic relationship to the parameter. It appears to be a weak positive relationship for the steel plates data, a weak negative relationship for the ionosphere data and mysteriously highly accurate and particularly uncorrelated to the banknotes data. It makes sense that KNN would be sensitive to the control parameter because its algorithm depends directly on the number of neighbours and changes in the distribution of data will dramatically affect which points the model uses to fit the training set. This contrasts with the MLP or other regression methods which use a hyperplane across the points and a regularization parameter which helps to reduce the effect of the distribution in vector space upon the accuracy of the model.

Part 2 – Clustering





Out of the seven algorithms, Affinity Propagation and DB Scanning proved least effective. Affinity propagation failed on the blobs and classification dataset, likely because of a too low number of iterations to find exemplars. DBScan failed on the “Classification” dataset, perhaps due to too many clusters of similar density. Mean shift also underperformed on the “classification” dataset, not identifying clusters in the data. This is likely because both DBScan and Mean Shift analyse density patterns in the data: the density for the “classification” set is uniform. Conversely, hierarchical (Agglomerative) clustering, K-Means and Birch all performed very well for this data set, identifying similar, distinct patterns in the data. Their implementations are similar: Birch is an alternative to mini-batch K-Means clustering, and all three rely on a tree-like structure (K-means in the sense that it branches from central nodes using an optimizing function).