COGS 118A: Classification Model Comparison

Joshua Hong
A15950737
University of California, San Diego
jjhong@ucsd.edu

Abstract

This paper aims to be a partial replication of the results found in the CNM06 paper. We select a subset of algorithms and metrics from the CMN06 paper and apply them to novel datasets. Our results largely reinforce the conclusions in CNM06, as Random Forests and neural networks were the best performers across metrics and data sets while logistic regression and decision trees consistently performed more poorly.

1 Introduction

When exploring classification problems, often the choice of algorithm used can lead to significant differences in performance. Therefore, it becomes useful to have general knowledge on which learning algorithms perform well under broad settings.

Studies in the past have tackled the issue of comparing supervised learning algorithms. While some of these earlier studies, such as STATLOG (King et al., 1995), served to provide a broad overview of algorithms when they were performed, more recent developments have necessitated a revaluation of classic algorithms as well as newer algorithms. Caruana and Niculescu-Mizil (2006), referred to as CNM06, provides a more in depth analysis of learning algorithms on a variety of dataset and metrics. Building off of a smaller study which compared the AUC performance of algorithms (Caruana et al., 2004), Caruana and Niculescu-Mizil conclude that boosted trees and random forests perform the best, but also that there is great variability in algorithm performance from problem to problem.

This study aims to be a partial replication of the results of the CNM06 Study. To accomplish this, we select a subset of the algorithms and metrics explored in CNM06 and compare performance on novel data sets. These results would verify that the conclusions of the CNM06 Study are applicable beyond the data sets in the original study.

2 Methodology

2.1 Learning Algorithms

In this comparison study, we explore four different classification algorithms: logistic regression, decision trees, random forests, and multi-layer perceptron networks. We detail the specifics of each model uses as well as the search space for hyper parameters below.

Logistic Regression (LOGREG): We trained models with L1 and L2 regularization as well as models with no regularization, varying the regularization parameter by factors of 10 from 10^{-8} to 10^{4}.

Decision Trees (DT): We trained models with two different splitting criteria, gini impurity and entropy. We also varied the maximum depth of the trees, training models with depths of 1, 2, 3, 4, and 5.

Preprint. Under review.

Random Forests (RF): We trained models that had 1024 trees each. Within each model we varied the splitting criteria between gini impurity and entropy. We also varied the maximum features considered at each split, considering the values 1, 2, 4, 6, 8, 12, 16, and 20. We note that if the dataset being trained on had less than 20 features, then only the values less than the number of features were considered.

Multi-Layer Perceptron Networks (MLP): We trained models with varying sizes for the hidden layer: 1, 2, 4, 8, 32, and 128. We also varied the regularization parameter by factors of 10 from 10^{-8} to 10^{4}.

2.2 Performance Metrics

To compare the performance of our selected learning algorithms, we used 3 performance metrics: accuracy, F1, and ROC.

2.3 Data Sets

The algorithms were trained and tested on 4 data sets available from the UCI Machine Learning Repository: HTRU2, OCCUPANCY, ELECTRIC_GRID, and CREDIT_DEFAULT.

HTRU2: This data set consists of data on possible pulsar candidates, with the goal being to classify candidates into pulsar and non-pulsar classes. We label the pulsar class as the positive class and use the other 8 continuous variables as predictors for our algorithms.

OCCUPANCY: This data set consists of data used for binary classification of room occupancy. We use the occupied status as the positive class and use the other 5 continuous variables as predictors for our algorithms. We exclude the date attribute as it only serves as a time stamp for every data point and is not included in the relevant papers under the data set on the UCI repository.

ELECTRIC_GRID: This data set consists of data on electric grids where the goal is to determine the stability of a system. As every data point contains a continuous label as well as categorical label, we drop the continuous label and use only the categorical label for our binary classification task. The other attributes are used as predictors for our algorithms.

CREDIT_DEFAULT: This data set consists of data on credit card clients and aims predict the probability of a customer defaulting. We designate the positive class as a customer defaulting and we use the other 23 attributes as predictors for our algorithms.

Table 1: Description of Problems

Problem	#ATTR	TRAIN SIZE	TEST SIZE	%POZ
HTRU2	9	5000	12898	9%
OCCUPANCY	7	5000	15560	23%
ELECTRIC_GRID	14	5000	5000	36%
CREDIT_DEFAULT	24	5000	25000	22%

2.4 Training and Testing Framework

To investigate every combination of data set and algorithm, we first select a algorithm from the 4 being compared. Then we select one of the 4 data sets and split the data set into a training set with 5000 data points and a test set. On the resulting training set, we run a five-fold cross validation grid search to select the optimal hyper-parameters for each performance metric. Using the optimal hyper-parameters, we train another classifier with the training set and obtain it's performance by predicting on the test set.

For every data set, this process is repeated 4 additional times to obtain a total of 5 trials for every data set and algorithm combination. Therefore, the mean performance of an algorithm with regard to a specific performance metric will be the average of 20 test scores.

This framework is largely the same between each algorithm. However, trials that involve LOGREG or MLP have the data points standardized before being trained upon. Likewise, trials that involve DT or RF don't require standardization of features.

3 Experiments & Results

3.1 Performances by Metric

Table 2 shows the scores of each learning algorithm on each of the 3 metrics. For each algorithm and metric, we take the 5 trials conducted on the 4 data sets and average the testing metric performance of each trial. The last column, MEAN, is the average score over the 3 metrics for each algorithm.

Within the table, the algorithm with the best score for each metric is **bolded**. Other algorithms whose performance within the same metric is not statistically distinguishable from the best algorithm at $p = 0.05$ using an unpaired t-test on the 20 trials are marked with a $*$. Therefore, entries within the table that are not bolded or starred have a statistically significant worse performance than the best model for $p = 0.05$.

When averaging across all metrics, we observe that the model with the highest score is the multi-layer perceptron, followed closely by random forests. On the other hand, logistic regression and decision trees did not perform significantly lower, but have mean scores that are roughly 0.05 lower than the multi-layer perceptron.

When looking at individual metrics, we see similar results; the multi-layer perceptron achieves the highest average score for accuracy, F1 and ROC. While the multi-layer perceptron does perform the best on all of the metrics being investigated, we are unable to conclude that the difference in performance is statistically significant due to the low sample size and the closeness of the mean scores.

Table 2: Scores for each Learning Algorithm by Metric (Averaged over 4 problems)

Algorithm	ACC	F1	ROC	MEAN
LOGREG	0.899*	0.740*	0.825*	0.821*
DT	0.877*	0.740*	0.830*	0.815*
RF	0.924*	0.797*	0.862*	0.861*
MLP	**0.929**	**0.812**	**0.875**	**0.872**

Table 3: P-values for each Learning Algorithm by Metric

Algorithm	ACC	F1	ROC	MEAN
LOGREG	0.254	0.320	0.273	0.101
DT	0.094	0.303	0.316	0.064
RF	0.814	0.830	0.764	0.701
MLP	1	1	1	1

3.2 Performances by Problem

The average performance of each algorithm on each dataset can be found in Table 4. In this table, each entry is the average over the performance of the 3 metrics we are measuring.

Under the HTRU2 data set, we observe that the multi-layer perceptron algorithm performed the best. While random forests, logistic regression, and decision trees were measured to have a worse average performance, our significance tests reveal that there is insufficient evidence to demonstrate a statistical difference.

For the OCCUPANCY data set, we observe that the random forest algorithm performed the best. However, all 4 algorithms performed similarly with a score of around 0.98 and only decision trees performed significantly worse with a p-value less than 0.05.

For the ELECTRIC_GRID data set, we observe that the multi-layer perceptron algorithm performed better than logistic regression, decision trees, and random forests. This difference in performance is reflected in Table 5, as the p-values for the other algorithms are less than 0.05, indicating that the there is a statistically significant difference in performance.

For the CREDIT_DEFAULT data set, we observe that the random forest algorithm performed the best with a average score of 0.642. Comparatively, the other classification algorithms performed worse, but not to a significant degree as all of the computed p-values are greater than 0.05.

In general, we observe that the training performance of decision trees and random forests is much higher than the metric scores obtained during testing. On the other hand, the training scores of logistic regression and the multi-layer perceptron algorithm are much closer to their performance on the test set.

Table 4: Scores for each Learning Algorithm by Problem (Averaged over 3 metrics)

Algorithm	HTRU2	OCCUPANCY	ELECTRIC_GRID	CREDIT_DEFAULT	MEAN
LOGREG	0.917*	0.985*	0.780	0.601*	0.821*
DT	0.899*	0.980	0.804	0.578*	0.815*
RF	0.922*	**0.987**	0.893	**0.642**	0.861*
MLP	**0.926**	0.985*	**0.950**	0.627*	**0.872**

Table 5: P-values for each Learning Algorithm by Problem

Algorithm	HTRU2	OCCUPANCY	ELECTRIC_GRID	CREDIT_DEFAULT	MEAN
LOGREG	0.579	0.382	5.37e-17	0.517	0.101
DT	0.173	0.004	1.23e-17	0.236	0.064
RF	0.805	1	4.89e-12	1	0.701
MLP	1	0.376	1	0.785	1

4 Discussion

Overall, the conclusions reached by the comparisons performed in this study do seem to reflect the results of the study conducted by Caruana and Niculescu-Mizil (2006). The random forest and multi-layer perceptron algorithms performed the best on all 4 data sets, while logistic regression and decision trees performed noticeably worse. These results seem largely in line with the findings by Caruana, as random forests and uncalibrated neural nets were some of the best performers.

Furthermore, our investigation into the scores for each data set reveals that there is significant variability across different problems. We observe that while multi-layer perceptron networks had the highest average score across all data sets and metrics, random forests achieved a higher average metric score on the OCCUPANCY and CREDIT_DEFAULT data sets. Similarly, the multi-layer perceptron algorithm performed significantly better than other algorithms on the ELECTRIC_GRID data set.

While our smaller subset of metrics and trials decrease the statistical power of our results, the general findings within this study do seem to reflect and uphold the findings of the CNM06 study.

5 Bonus

This paper explores 4 algorithms instead of the required 3. While random forests and neural nets were some of the best performers in the CNM06 Study, decision trees and logistic regression performed more poorly. Beyond the selection of algorithms, all the data sets chosen are different from the data sets featured in the CNM06 Study. Additionally, ROC and PR curves for each algorithm and data set as well as heat maps of validation performance across hyper-parameter settings are included in the appendix.

References

Caruana, Rich, and Alexandru Niculescu-Mizil. "An empirical comparison of supervised learning algorithms." *In Proceedings of the 23rd international conference on Machine learning, pp. 161-168.* 2006.

Caruana, Rich, and Alexandru Niculescu-Mizil. "An Empirical Evaluation of Supervised Learning for ROC Area." In *ROCAI*, pp. 1-8. 2004.

King, R., C. Feng and A. Sutherland. "STALOG: Comparison of classification algorithms on large real-world problems." *Appl. Artif. Intell.* 9 (1995): 289-333.

Appendix

Table 6: Training scores for each Learning Algorithm by Problem (Averaged over 3 metrics)

Algorithm	HTRU2	OCCUPANCY	ELECTRIC_GRID	CREDIT_DEFAULT	MEAN
LOGREG	0.920	0.986	0.782	0.602	0.823
DT	1	1	1	0.999	0.999
RF	0.999	1	1	0.999	0.999
MLP	0.933	0.987	0.999	0.787	0.927

Table 7: Raw Scores for Logistic Regression

Dataset_Metric	Trial_1	Trial_2	Trial_3	Trial_4	Trial_5	MEAN
HTRU2_Accuracy	0.977051	0.977438	0.979144	0.978756	0.976896	0.977857
HTRU2_F1	0.861293	0.871863	0.877449	0.874427	0.867438	0.870494
HTRU2_ROC_AUC	0.896799	0.907240	0.908895	0.903697	0.901071	0.903541
OccupancyData_Accuracy	0.988817	0.988946	0.988689	0.988882	0.988560	0.988779
OccupancyData_F1	0.976099	0.976471	0.976054	0.976285	0.975868	0.976155
OccupancyData_ROC_AUC	0.990778	0.991060	0.990799	0.990821	0.990813	0.990854
ElectricGridData_Accuracy	0.816600	0.817800	0.816200	0.811600	0.815800	0.815600
ElectricGridData_F1	0.733663	0.733236	0.737653	0.728687	0.736631	0.733974
ElectricGridData_ROC_AUC	0.788600	0.791795	0.793297	0.788523	0.793934	0.791230
CreditDefaultData_Accuracy	0.809400	0.813240	0.814680	0.811320	0.810960	0.811920
CreditDefaultData_F1	0.359285	0.395129	0.386601	0.379587	0.371543	0.378429
CreditDefaultData_ROC_AUC	0.606132	0.620838	0.617517	0.614290	0.611055	0.613966

Table 8: Raw Scores for Decision Tree

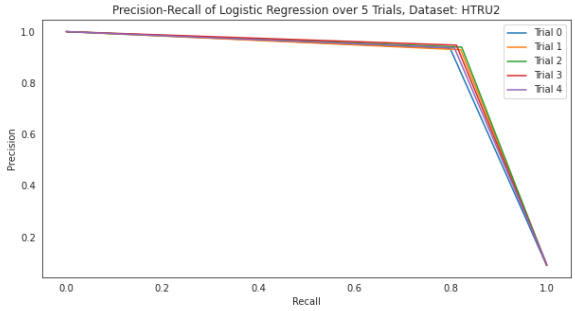
Dataset_Metric	Trial_1	Trial_2	Trial_3	Trial_4	Trial_5	MEAN
HTRU2_Accuracy	0.969298	0.966119	0.968832	0.967127	0.968522	0.967980
HTRU2_F1	0.833819	0.811130	0.821105	0.823727	0.818609	0.821678
HTRU2_ROC_AUC	0.907298	0.905820	0.914939	0.912640	0.899407	0.908021
OccupancyData_Accuracy	0.988496	0.986632	0.987468	0.987018	0.984897	0.986902
OccupancyData_F1	0.974997	0.970291	0.974753	0.973564	0.967532	0.972227
OccupancyData_ROC_AUC	0.984213	0.980935	0.985346	0.981261	0.976229	0.981597
ElectricGridData_Accuracy	0.826000	0.825400	0.829200	0.834600	0.832600	0.829560
ElectricGridData_F1	0.771881	0.759128	0.754761	0.775293	0.773481	0.766909
ElectricGridData_ROC_AUC	0.816110	0.809378	0.813502	0.820671	0.820097	0.815952
CreditDefaultData_Accuracy	0.722680	0.722400	0.719680	0.723520	0.721520	0.721960
CreditDefaultData_F1	0.399168	0.410879	0.395921	0.388952	0.393238	0.397632
CreditDefaultData_ROC_AUC	0.616566	0.619515	0.611620	0.606884	0.612003	0.613318

Table 9: Raw Scores for Random Forest

Dataset_Metric	Trial_1	Trial_2	Trial_3	Trial_4	Trial_5	MEAN
HTRU2_Accuracy	0.978369	0.978136	0.978601	0.979687	0.977748	0.978508
HTRU2_F1	0.876834	0.875671	0.877788	0.877717	0.876688	0.876940
HTRU2_ROC_AUC	0.914340	0.913012	0.906251	0.904930	0.921426	0.911992
OccupancyData_Accuracy	0.990746	0.991324	0.991260	0.991710	0.989781	0.990964
OccupancyData_F1	0.980446	0.980679	0.980811	0.981998	0.978478	0.980483
OccupancyData_ROC_AUC	0.988944	0.990436	0.989696	0.991945	0.989540	0.990112
ElectricGridData_Accuracy	0.909600	0.910200	0.915200	0.916400	0.911400	0.912560
ElectricGridData_F1	0.865655	0.873654	0.869642	0.877122	0.869565	0.871128
ElectricGridData_ROC_AUC	0.893918	0.894382	0.896077	0.898680	0.892565	0.895124
CreditDefaultData_Accuracy	0.813000	0.811280	0.813320	0.813960	0.811800	0.812672
CreditDefaultData_F1	0.457057	0.453513	0.472810	0.452004	0.464130	0.459903
CreditDefaultData_ROC_AUC	0.649691	0.650363	0.658389	0.650401	0.651084	0.651986

Table 10: Raw Scores for Multi-layer Perceptron

Dataset_Metric	Trial_1	Trial_2	Trial_3	Trial_4	Trial_5	MEAN
HTRU2_Accuracy	0.979144	0.979687	0.980772	0.980307	0.978446	0.979671
HTRU2_F1	0.881264	0.881847	0.889379	0.888193	0.872858	0.882708
HTRU2_ROC_AUC	0.917780	0.915351	0.919872	0.915552	0.915006	0.916712
OccupancyData_Accuracy	0.988046	0.988496	0.989396	0.988817	0.988882	0.988728
OccupancyData_F1	0.974742	0.975216	0.977285	0.976355	0.976335	0.975986
OccupancyData_ROC_AUC	0.990823	0.990278	0.991485	0.990929	0.991128	0.990929
ElectricGridData_Accuracy	0.958400	0.956800	0.956600	0.951200	0.956800	0.955960
ElectricGridData_F1	0.941048	0.938901	0.940850	0.937707	0.943759	0.940453
ElectricGridData_ROC_AUC	0.957136	0.958518	0.948667	0.953472	0.953534	0.954265
CreditDefaultData_Accuracy	0.797080	0.787720	0.794960	0.791680	0.795680	0.793424
CreditDefaultData_F1	0.430030	0.455709	0.445986	0.456371	0.447391	0.447097
CreditDefaultData_ROC_AUC	0.649328	0.621278	0.637414	0.640232	0.648776	0.639405



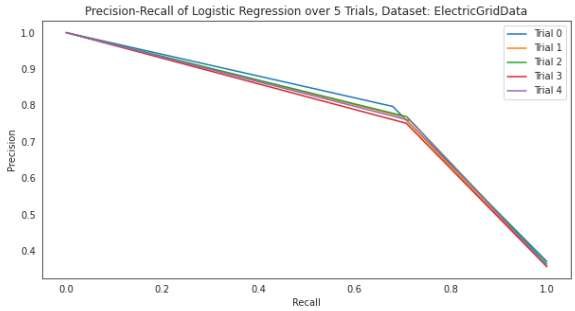
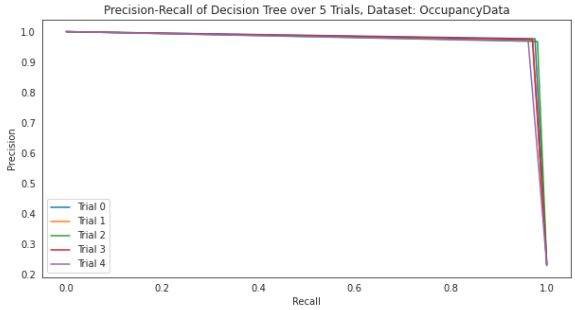


Figure 1: Precision-Recall Curves of Logistic Regression



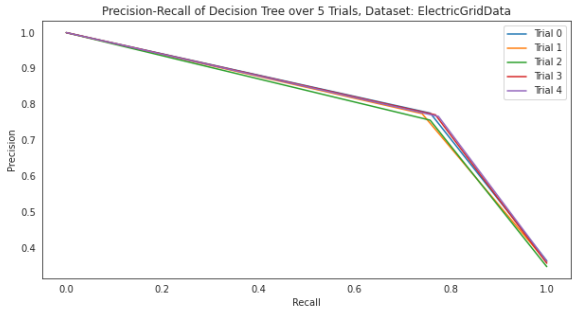
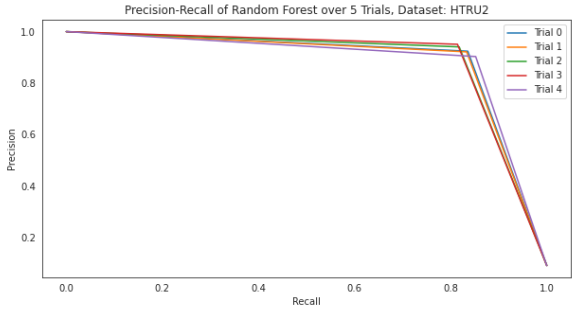
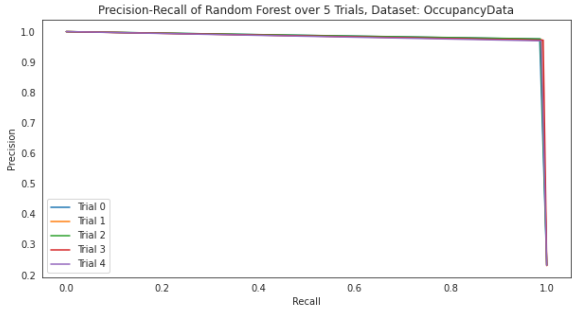
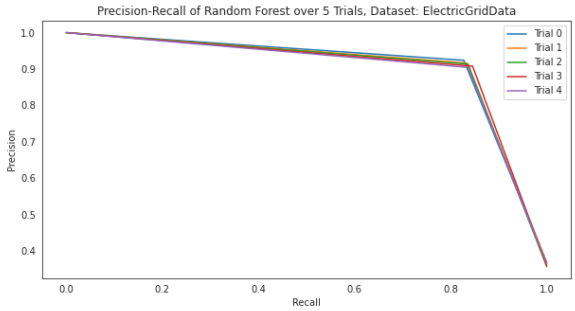


Figure 2: Precision-Recall Curves of Decision Trees







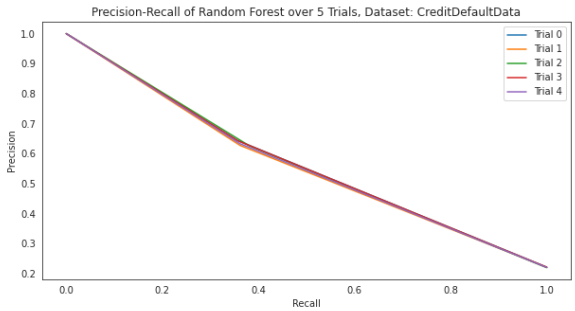
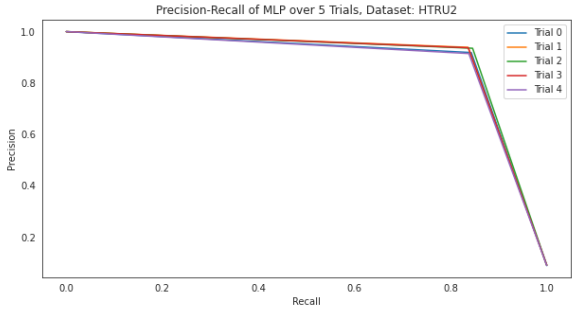
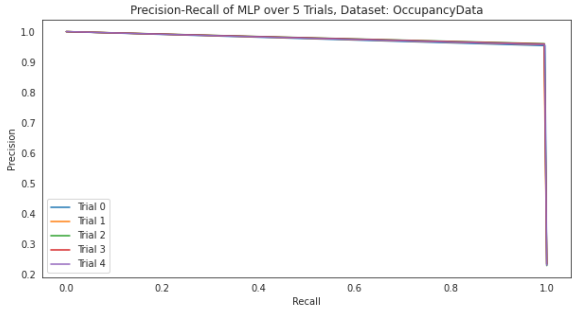
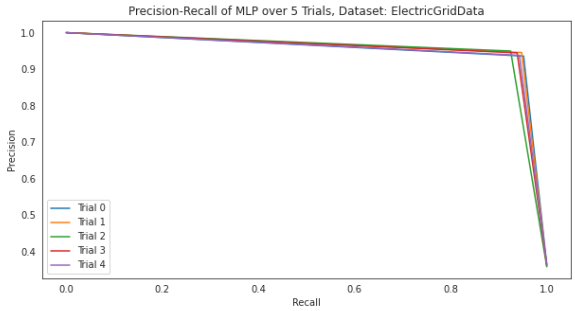


Figure 3: Precision-Recall Curves of Random Forest







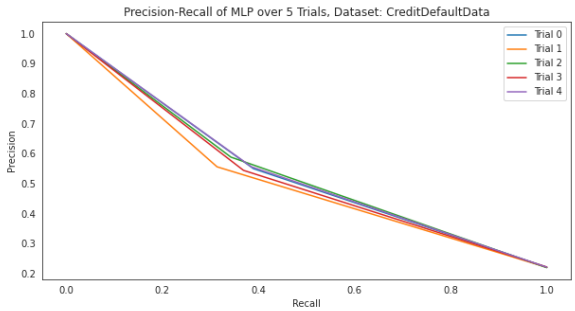
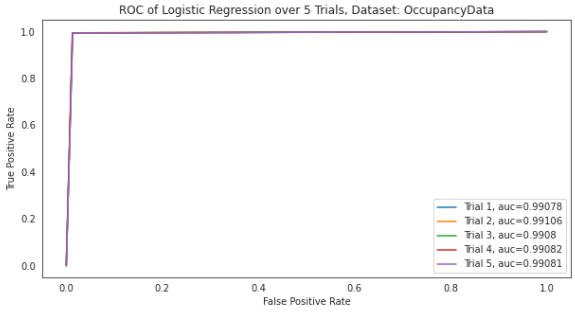
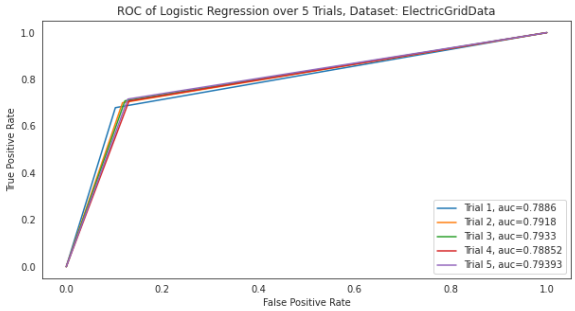


Figure 4: Precision-Recall Curves of MLP





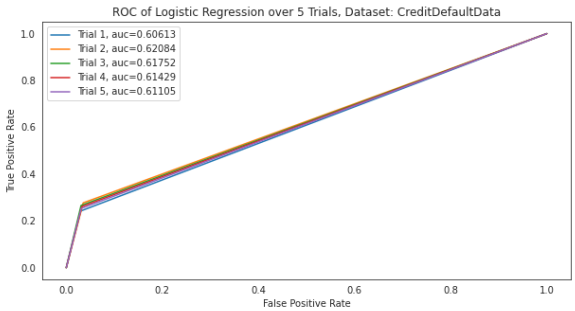
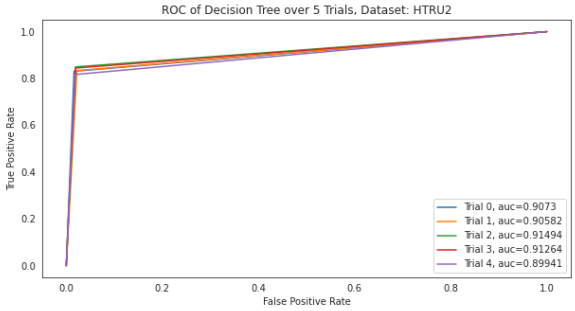
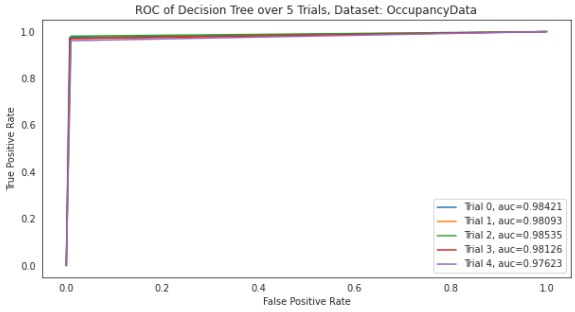
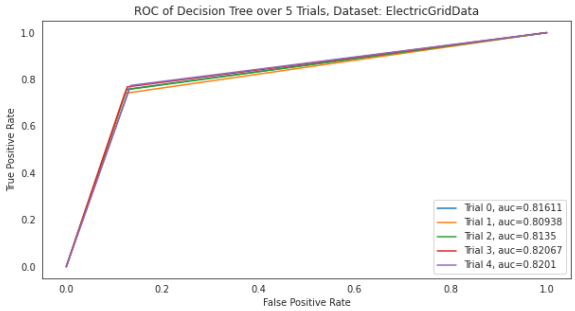


Figure 5: ROC Curves of Logistic Regression







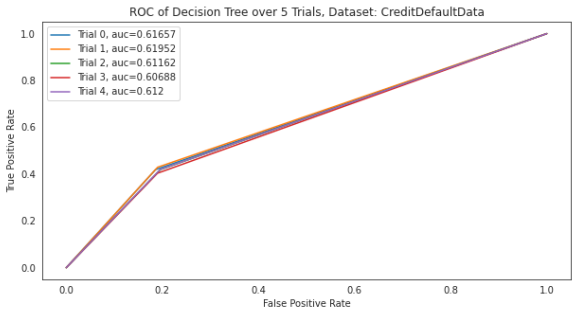
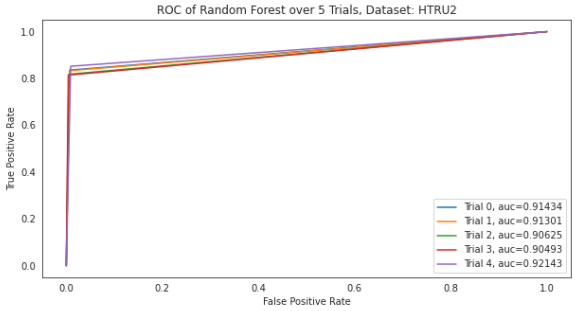
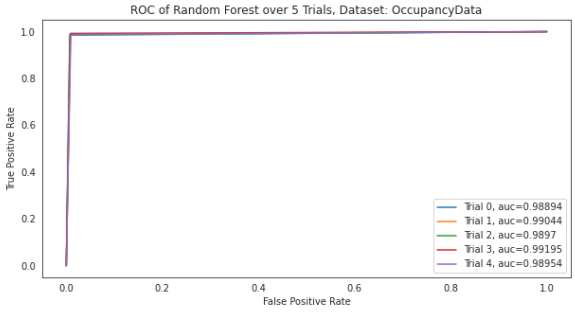
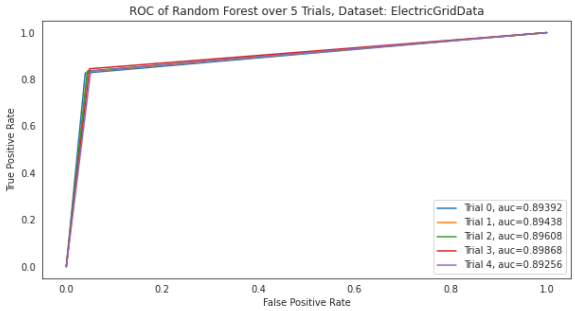


Figure 6: ROC Curves of Decision Tree







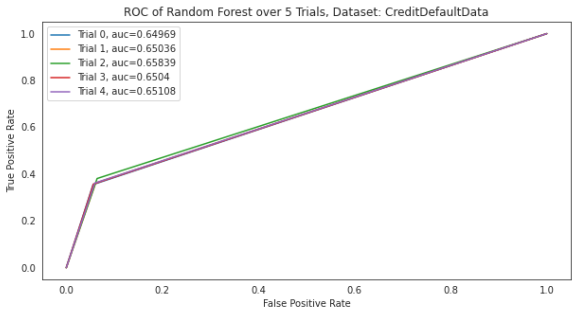
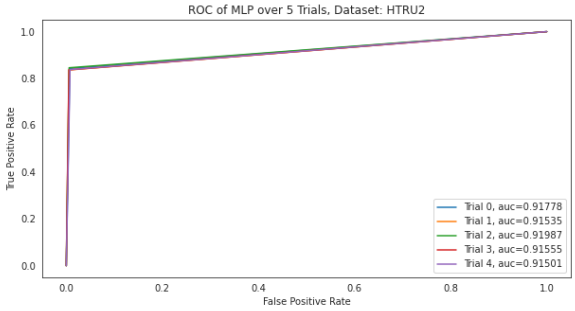
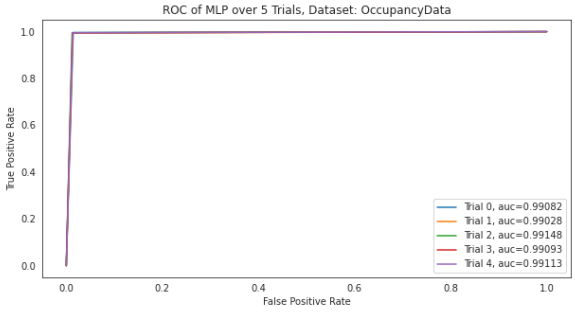
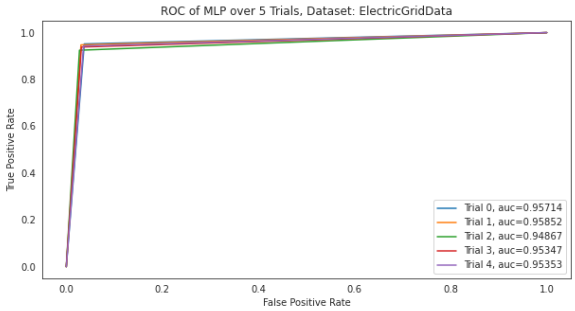


Figure 7: ROC Curves of Random Forest







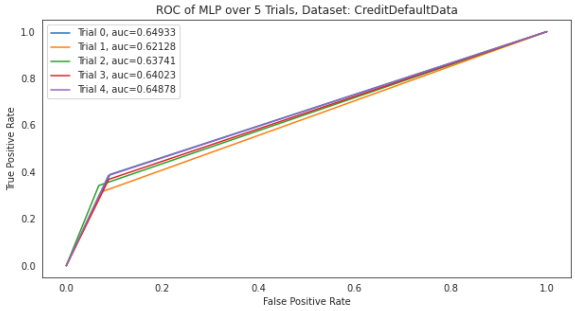


Figure 8: ROC Curves of MLP

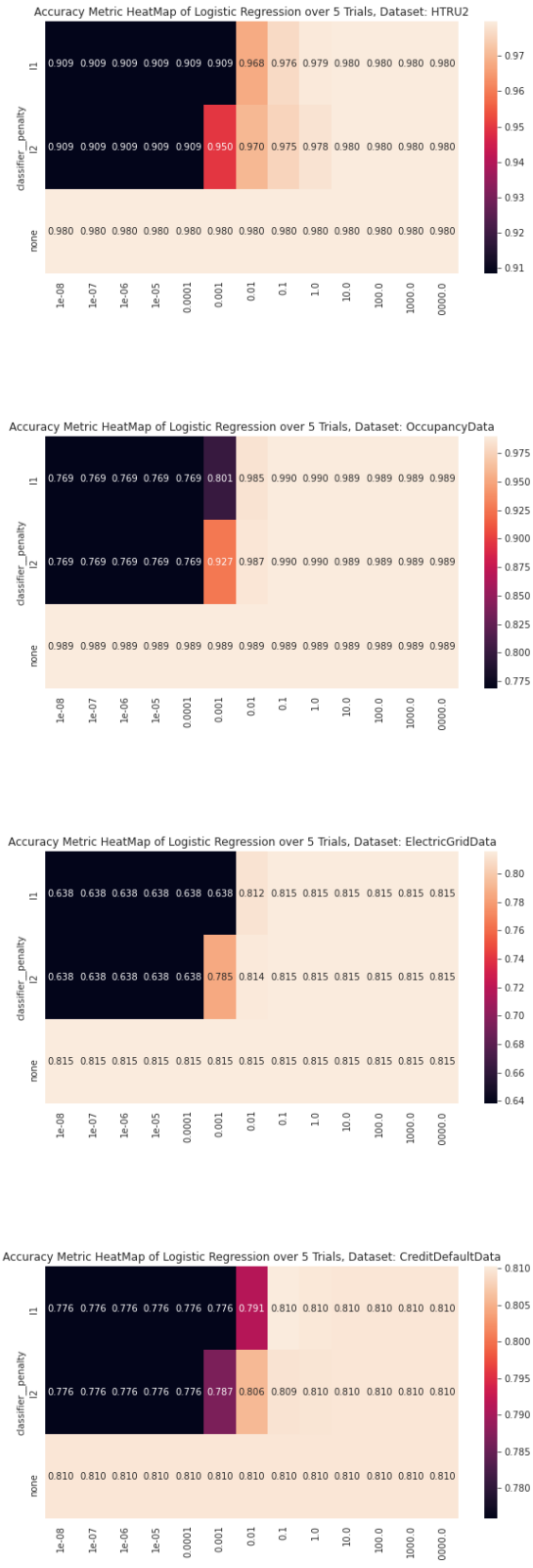


Figure 9: Accuracy Heat Maps for Logistic Regression

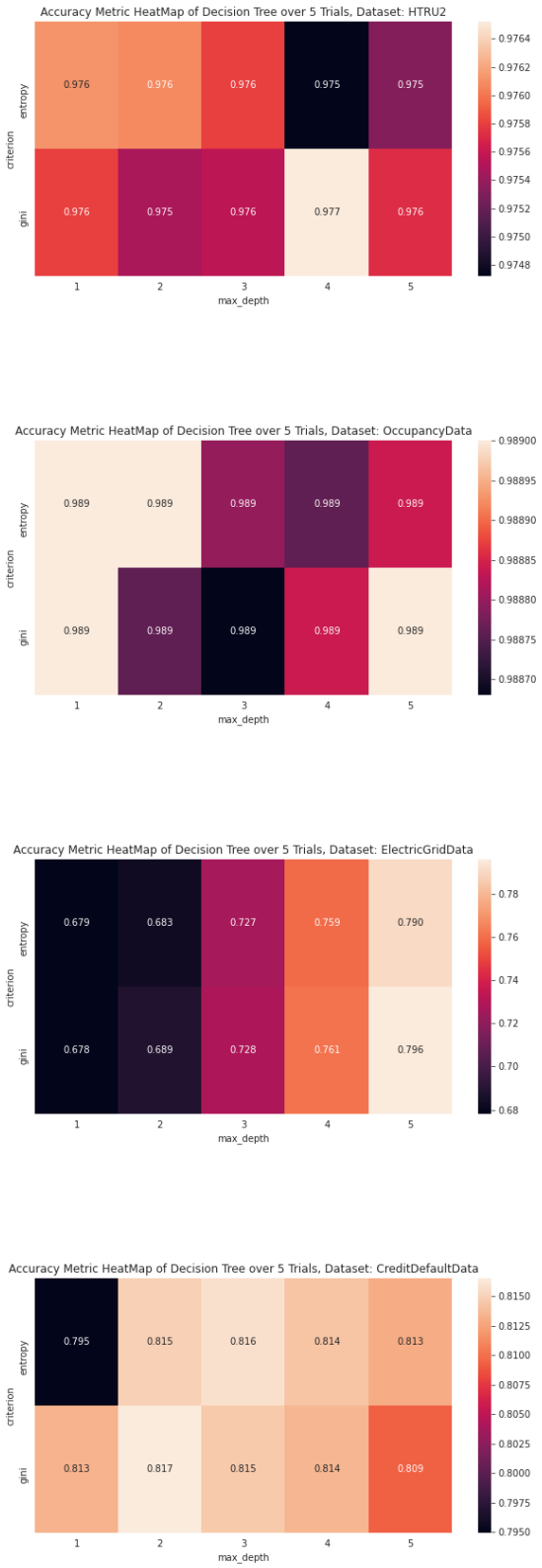


Figure 10: Accuracy Heat Maps for Decision Tree

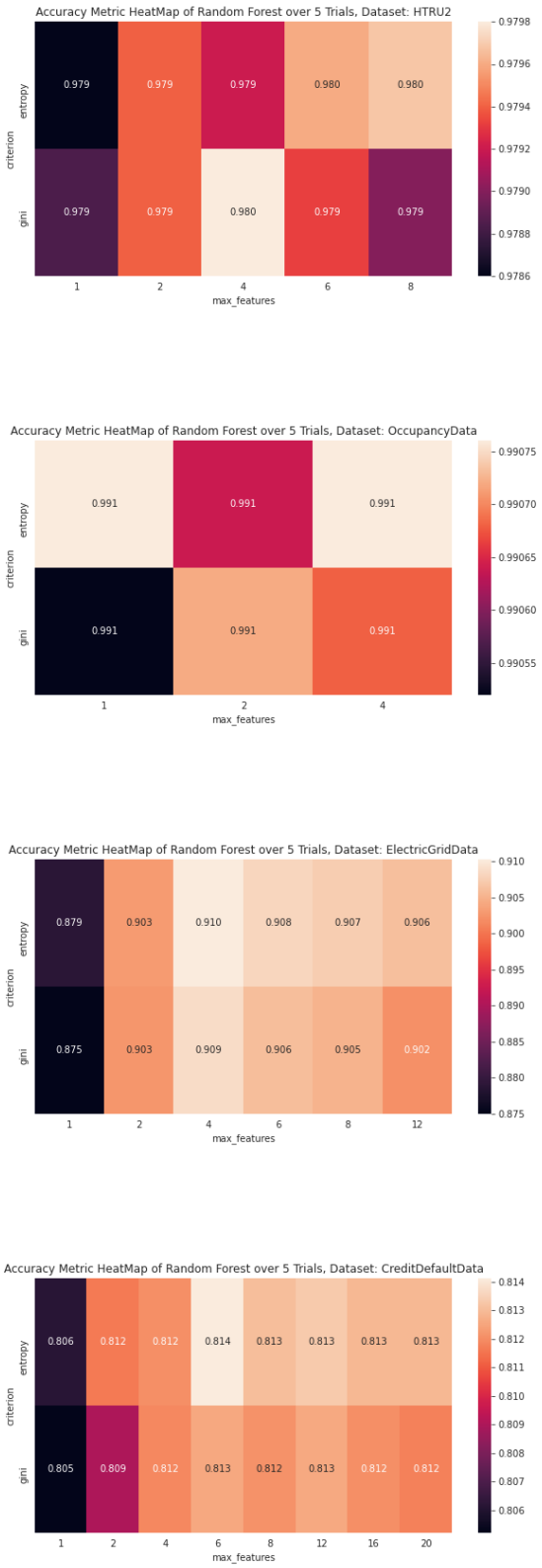


Figure 11: Accuracy Heat Maps for Random Forest

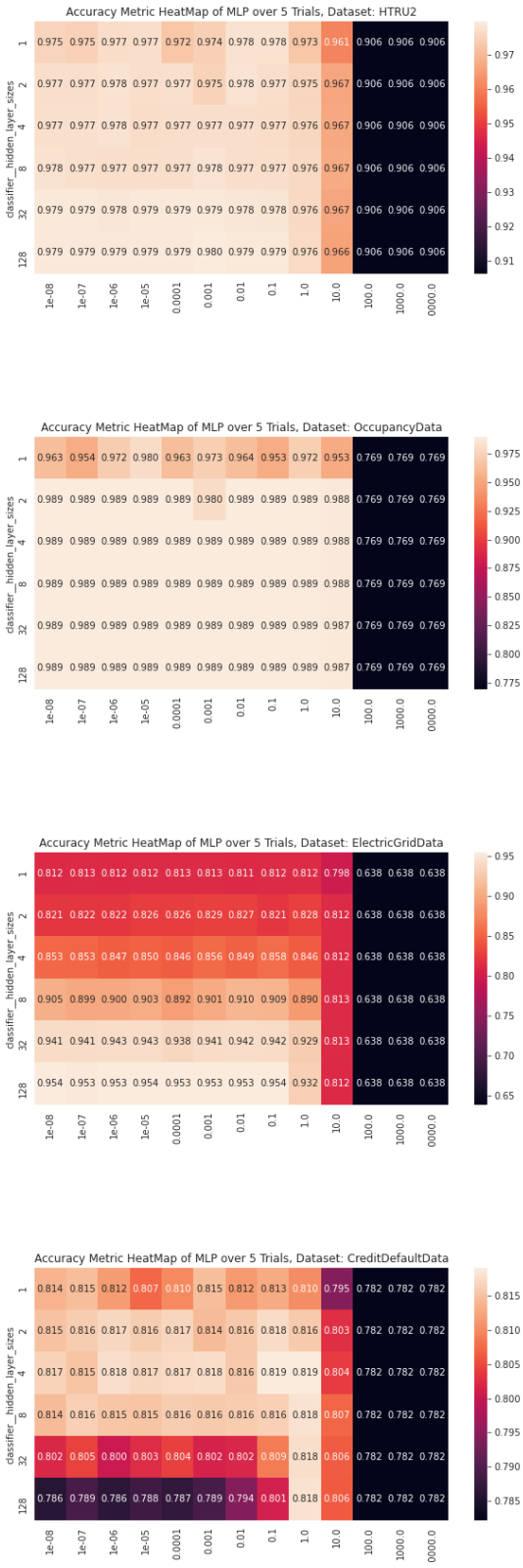


Figure 12: Accuracy Heat Maps for MLP

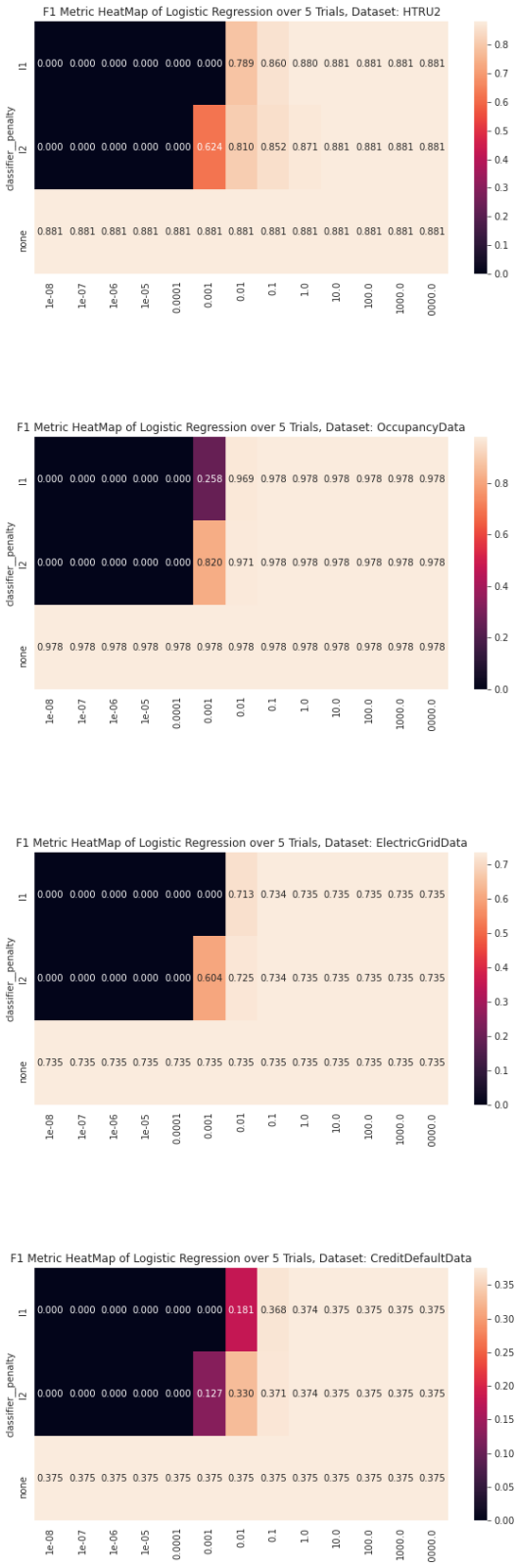


Figure 13: F1 Heat Maps for Logistic Regression

Figure 14: F1 Heat Maps for Decision Tree

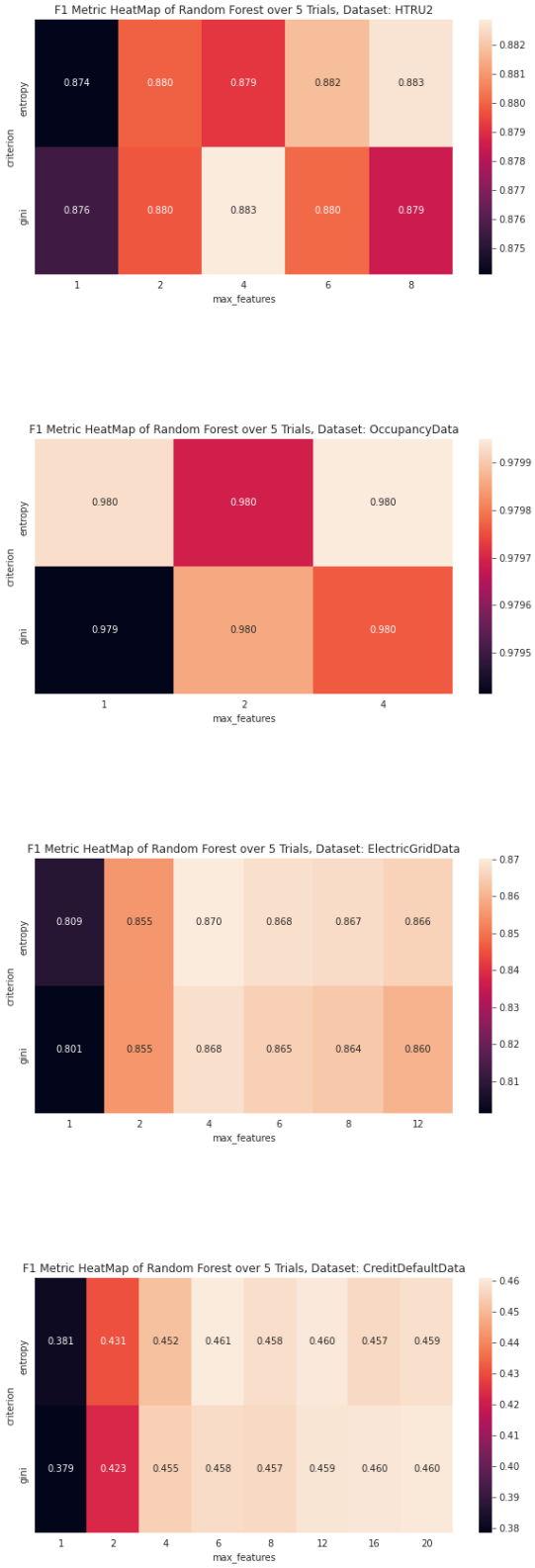


Figure 15: F1 Heat Maps for Random Forest

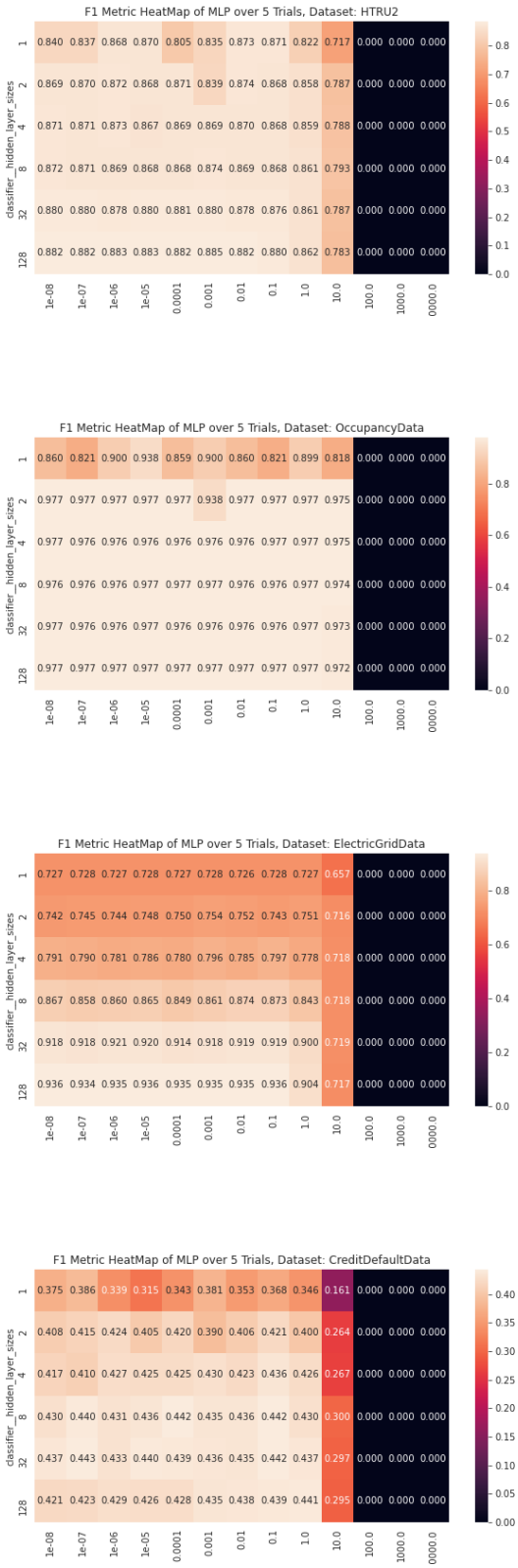


Figure 16: F1 Heat Maps for MLP

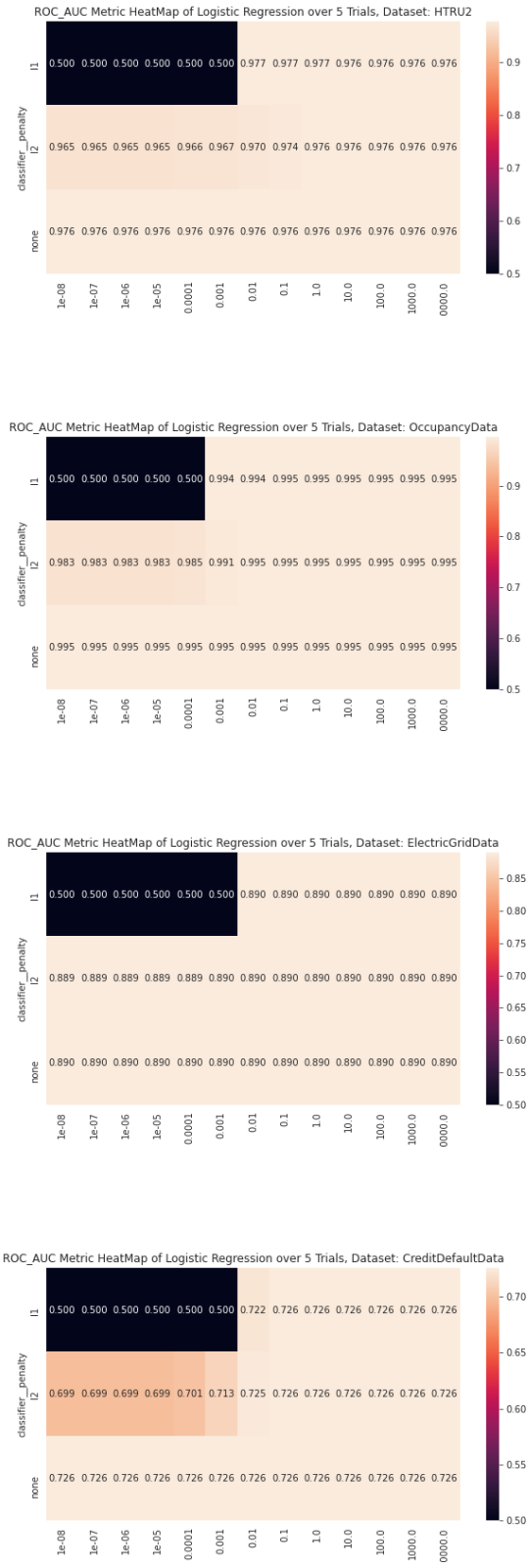
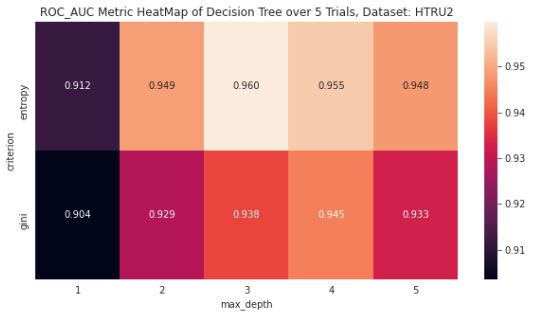
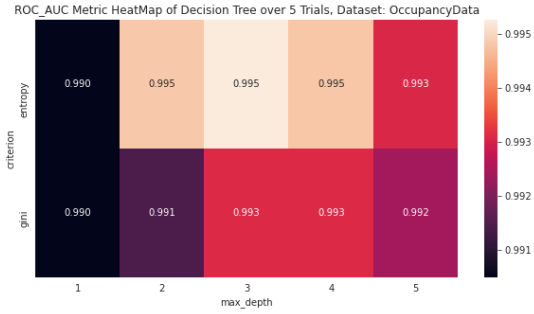
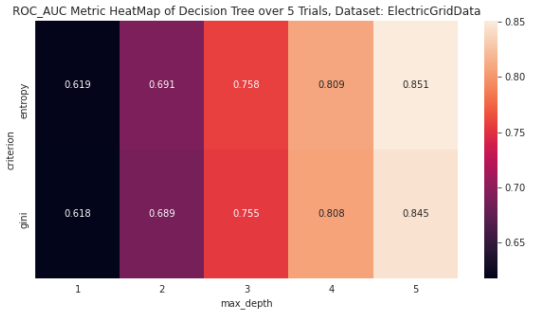


Figure 17: ROC Heat Maps for Logistic Regression







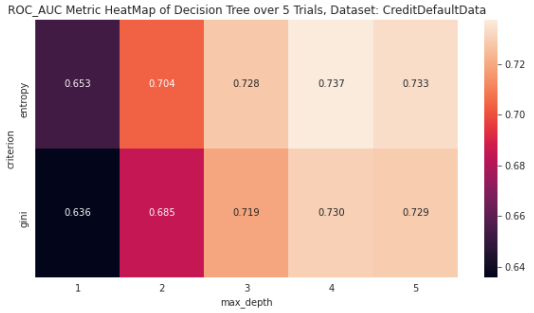
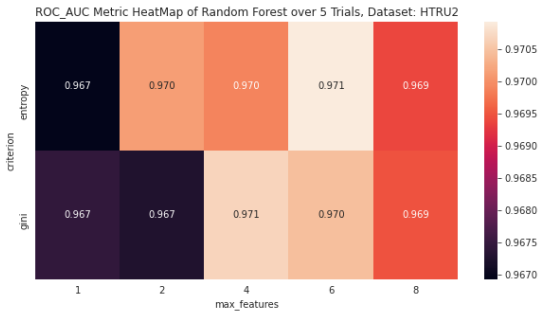
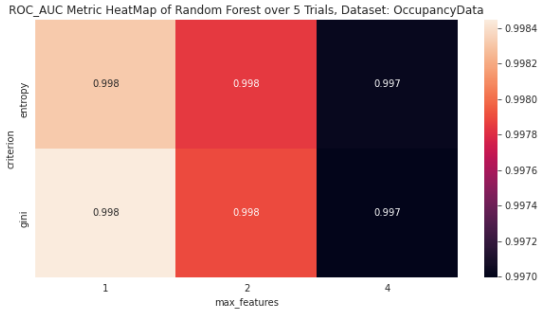
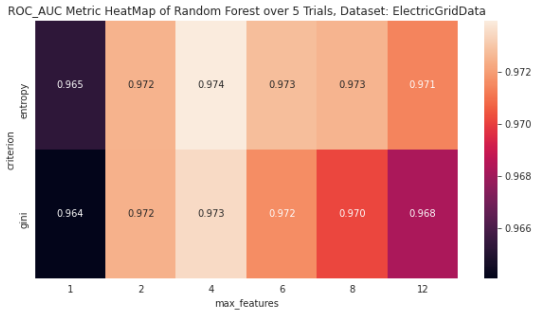


Figure 18: ROC Heat Maps for Decision Tree







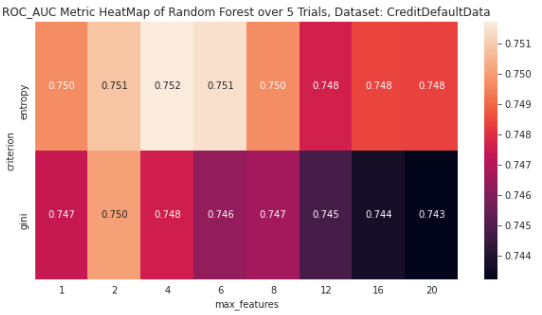


Figure 19: ROC Heat Maps for Random Forest

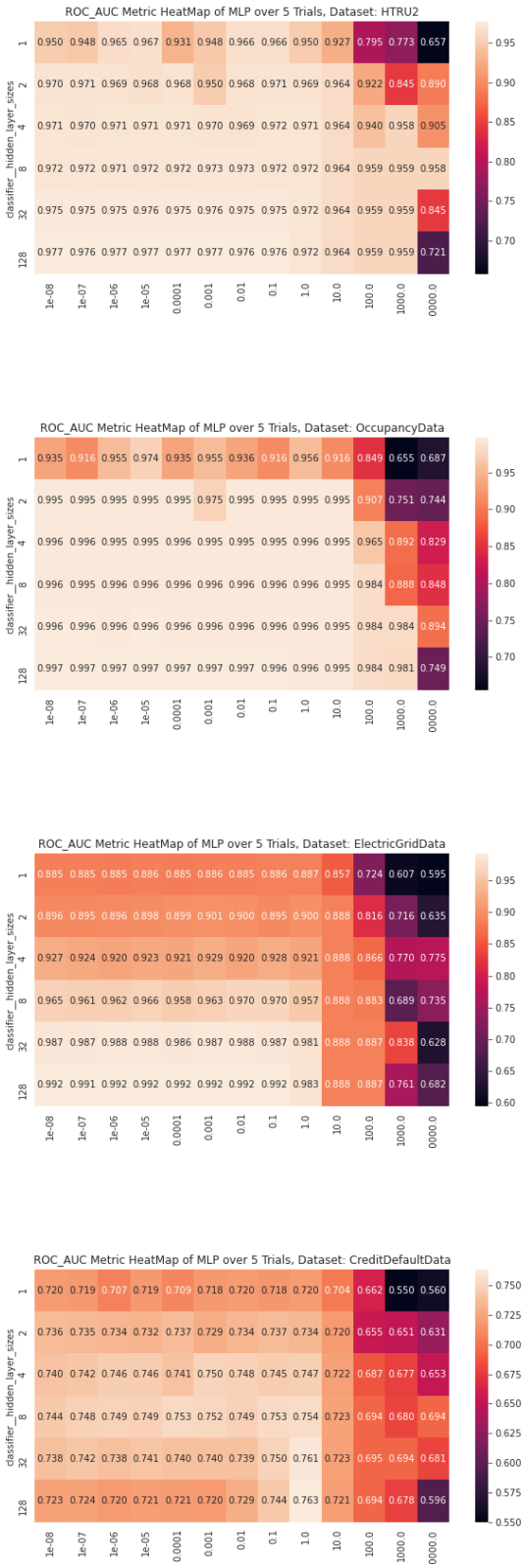


Figure 20: ROC Heat Maps for MLP