An Experimental Evaluation of Supervised Learning Algorithms

Based on "An Empirical Comparison of Supervised Learning Algorithms"
Rich Caruana and Alexandru Niculescu-Mizil, 2006
Adapted by Ella Lucas for UC San Diego, 2019

Abstract

Supervised learning techniques have provided a means to classify never before seen instances of data based on trends observed from input features. This task requires an algorithm to correctly generalize information in a "supervised" environment, which just means that the classifier is given both inputs and outputs, which are then labeled. Machine learning evolves rapidly, and like any fast-moving division of science there is a demand for periodic evaluation of the methodologies that have been put to practice. Caruana and Niculescu-Mizil published a large-scale comparison of ten supervised learning methods in 2006, and their research inspired the work contained. The following report contains an evaluation of four supervised learning algorithms: logistic regression, decision tree, multi-layer perceptron (MLP), and k nearest neighbor (KNN). These learning methods are evaluated based on testing accuracies.

1. Introduction

As of 2006, when the inspiration for this analysis was published, it was noted that the machine learning community was lacking in comprehensive experimental comparison of classifiers, and that such a task is an ongoing process considering the rate at which algorithms are developed. This report is a modern survey of the classical methodologies.

The four algorithms were evaluated on three binary classification problems sourced from the UCI Machine Learning Repository using accuracy as the performance metric. For each algorithm, k-fold cross-validation is implemented to locate the best hyper-parameters, signifying that the parameter space was thoroughly explored to ensure a fair comparison between algorithms.

Each dataset was shuffled and partitioned into three divisions prior to model fitting: 80% training 20% testing, 50% training 50% testing, and 20% training 80% testing.

The empirical results are *consistent* with those of Caruana and Niculescu-Mizil's findings, however they by no means mirror results of prior studies due to natural variations in data selection and pre-processing, model implementation, and metrics for performance evaluation. This survey will point out and explain significant variations from the prior study to demonstrate consistency.

To preview the experimental findings, k nearest neighbors gave the best average performance across all three test problems and all three partitions, closely followed by logistic regression, with decision tree and the multi-layer perceptron achieving the same trailing accuracy score.

These results were highly influenced by the Occupancy Detection dataset, which was a significantly easier classification task.

2. Data & Problem Description

The algorithms are compared on 3 binary classification problems.

The Adult Data Set (ADULT) is used to train the algorithms to predict whether an individuals income exceeds $50K/yr based on census data. It contains 48,842 instances and 14 attributes, making it substantially large.

The Wine Quality Data Set (WINE) is used to train algorithms to predict the quality of wine based on physicochemical tests. Wines are scaled from 1-12 in terms of quality. For the purposes of this survey, the classifiers are trained to predict whether a given wine is of "good" of "bad" quality based on the sign of its label. It contains 4,898 instances and 12 attributes.

The Occupancy Detection Data Set (OCC) is used to train algorithms to predict whether a room is occupied of not based on temperature, humidity, light, and CO2. It contains 20,560 instances and 7 attributes.

ADULT, WINE, and OCC are sourced from the UCI Repository (Blake & Merz, 1998). WINE was converted to a binary problem by treating the top 50% of wines as positive and the bottom 50% of wines as negative. ADULT and OCC contain nominal features, which were converted to numeric via label encoding. ADULT and WINE were normalized, OCC was not due to the high interdependence of its features, which lended better performance without normalizing.

Furthermore, data pre-processing incorporated specific partitions of the data to be used for training and testing.

See tables below for descriptions of each problem.

	Train Size	Test Size
80/20 Split	39073	9769
50/50 Split	24421	24421
20/80 Split	9768	39074

Table 1.1 Adult

	Train Size	Test Size
80/20 Split	16448	4112
50/50 Split	10280	10280
20/80 Split	4112	16448

Table 1.2 Occupancy

	Train Size	Test Size
80/20 Split	5197	1300
50/50 Split	3248	3249
20/80 Split	1299	5198

Table 1.3 Wine

3. Method Description

3.1 Learning Algorithms

This section introduces the algorithms used and their corresponding parameters. All parameters were explored using sci-kit-learn's grid search cross validation (3-fold), which also lends experimental results (covered in section 4).

Logistic Regression (LR): trained regularized model built using sci-kit-learn's libraries. 'Liblinear' solver used to support both L1 and L2 regularization. Model was set to handle binary data (multi-class default). 'C', the inverse regularization parameter, is varied by factors of 10.

Decision Tree (tree): built decision tree classifier to measure split quality with 'entropy' function using sci-kit learn's libraries. The max depth parameter varies from 1-5.

Multi-Layer Perceptron (MLP/NN): built MLP (deep ANN) using sci-kit learn's libraries. The activation parameter varies between available functions (identity, logistic, tanh, relu). The initial learning rate, which specifies the step-size in updating weights, is set at .02 and the maximum number of iterations is set at 2,000.

K-Nearest Neighbors (KNN): built KNN using ski-kit learn's libraries. Five values of K are used ranging between 1-5.

3.2 Performance Metric

The above classifiers were judged based on test accuracy, ranging from [0,1]. Test accuracy is obtained using cross-validation and reported for each classifier, dataset, and partition.

4. Experiments

The data corresponding to each classification problem is trained, validated, and tested on each of the four classifiers. Cross validation is used to find the best hyper-parameters and to report average accuracy scores. Accuracy scores are averaged over three trials for three different partitions.

Table 2.1 shows the normalized accuracy score for each algorithm. For each problem, in order to compute this accuracy, the best parameter settings were determined (based on the chosen hyper-parameters specified in Section 3) through cross-validation. The metric outcome is the normalized score for each model on the test data. Each entry in the table averages these scores across all trials and test problems. Higher scores indicate better performance. The models in this table are sorted by mean overall test accuracy score.

Classifier	Avg. Test Accuracy
KNN	**0.9**
LR	0.87
tree	0.85
NN	0.85

Table 2.1 Mean Test Accuracy for Each Classifier, Averaged Over Each Partition and Problem

As is discernible from the table, the strongest model was k nearest neighbors, with the others closely following behind. The results reported by Caruana and Niculescu-Mizil differed slightly in rank from what was found in this experimental process. To ensure that this discrepancy was not due to an error in model implementation, different parameter settings and data partitions were tested to observe the effect on accuracy. This difference is much more likely attributed to the OCC dataset, which acted as an outlier thanks to its extremely high performance, and caused differences from the paper due to the fact that it is a novel dataset unique to this analysis. Average performance for the MLP is equivalent between the two studies.

Table 2.2 shows the averaged accuracy score for each algorithm on each classification problem. This provides a more granular view of model performance and highlights the above point that the Occupancy Detection problem was navigated with ease by the classifiers, misleading overall averages.

Classifier	ADULT	WINE	OCC
LR	0.81	0.71	**0.99**
tree	0.81	0.71	**0.99**
NN	0.85	0.76	**0.94**
KNN	0.87	0.82	**0.99**

Table 2.2 Test Accuracy Averaged over Each Trial and Partition for each Classifier

The ADULT dataset is the only used in both this experimentation and that of the source paper. This is a significant motivator behind any differences in overall results considering that the data associated with every classification problem has its own requirements for pre-processing prior to

model fitting, and will be approached differently depending on the classifier selected and model parameters. This is the very premise of the No Free Lunch Theorem, suggesting that there is no single superior learning algorithm. Despite the rank of classification algorithms in Caruana and Niculescu-Mizil's study, it is not empirical to assume that the same algorithm that achieved high performance on one problem in one setting will be the best in another problem setting, even when the process is designed to be similar.

With the No Free Lunch Theorem noted, there are still consistencies between the study environments apparent in the results. On the ADULT dataset, the neural networks (MLP) performance between both studies is highly consistent.

Logistic regression and decision tree models underperformed slightly compared to the source study on the ADULT data, while the KNN model performed significantly better.

Table 2.3 shows the test accuracy results organized by classification problem, averaged over all trials and partitions for each classifier. This demonstrates the best classification algorithm for each problem, another means of analyzing experimental outcome.

Data	LR	tree	NN	KNN
ADULT	0.81	0.83	0.85	**0.87**
WINE	0.71	0.74	0.76	**0.82**
OCC	**0.99**	**0.99**	0.94	**0.99**

Table 2.3 Mean Test Accuracy for Each Classification Problem, by Each Algorithm

KNN was highly suited for all of the classification problems tested in this study. It produced the best results on the ADULT and WINE datasets, and shares almost perfect accuracy on the OCC dataset with LR and decision tree.

The optimal model on the ADULT dataset from the source paper was a boosted stump, which was not tested here.

Consider Table 2.4 for accuracy averaged over all trials, partitions, and algorithms for each classification problem. This report lends insight to how effectively each problem was approached accounting for all aspects of the experimental environment.

Dataset	Avg. Test Accuracy
ADULT	0.83
WINE	0.74
OCC	0.99

Table 2.4 Mean Accuracy for Each Problem over All Partitions

The Occupancy Detection problem is approached with apparent ease by all of the classifiers. During experimentation, this success rate was initially surprising in comparison to the other problems. Several steps were taken to further examine this result before it was reported as a truth and analyzed.

Conceptually, model accuracy decreases when the size of the training dataset decreases, assuming proper implementation. Sometimes on an easy classification problem, the effect of 80/20, 50/50, and 20/80 train/test splits may not result in a large difference between accuracy scores. Regardless, forcing the training set to be *very* small (1%, or 10%) should lead to visibly declining accuracies. This was tested on the OCC set and accuracy decreased accordingly.

Secondly, external research on the Occupancy Detection problem proved that those who have worked with the data in the past find it conducive to the same "nearly perfect" classification achievements seen here. In a 2018 blog post in Deep Learning for Time Series, Jason Brownlee implemented logistic regression on the same dataset with an accuracy of .99 (Brownlee, 2018).

Table 4.5 shows the accuracy averaged over each trial, classifier, and dataset for each partition. This demonstrates the effects of partitioning data in the pre-processing stage for training and testing a model.

Train/Test Split	Avg. Test Accuracy
80/20	**0.86**
50/50	0.85
20/80	0.85

Table 2.5 Mean Accuracy for Each Partition

As expected, providing more training data to a given model, regardless of the classification problem, parameters, number of features, etc. lends a higher test accuracy.

5. Conclusion

Results have been shown by both studies to be highly dependent on the problem choice, analytical metrics, and classifier choice. It is noted in Section 5 ("Bootstrap Analysis") of Caruana and Niculescu-Mizil's study that the omission (or hypothetically, inclusion) of a single problem and its corresponding dataset could significantly alter rankings.

This is why the two studies are *comparable* but not equivalent. Section 4 pointed out areas where similarities and differences arose, with an explanation as to why based on the experimental design.

Had there been more time and computational space, this study would like to have incorporated another dataset and boosted the decision tree classifier to observe the effect on performance.

Overall, the state of the field is an expansive space and is growing rapidly. Despite when a learning method was first developed and put to practice, major improvements have been proposed to enhance even the classical designs. This is apparent through the use of Sci-Kit Learn's libraries throughout this experimentation. Sci-Kit Learn powered the development of each of the classifiers, performing grid search cross-validation within the classifier's definitions making the design comprehensive, efficient, and effective.

All of the learning methods used in this study achieved excellent performance, particularly KNN (best classifier, mean accuracy = 0.90) and LR (mean accuracy = 0.87), demonstrating that model complexity is not always correlated with enhanced performance. The adaption of MLP and decision tree both obtained a mean accuracy of 0.85.

Partitioning the data so that the model is trained on 80% of the set and tested on the remaining 20% yields the best mean accuracy (0.86). The other partitions used in the study were 50% training data, 50% testing data (mean accuracy = 0.85) and 20% training 80% testing (mean accuracy = 0.85). It is not unusual that the variance between the partition accuracies is low given the size and difficulty of the classification problems used in this study.

Consider the accuracy results reported within are often *averages* across three trials (to account for randomness), three partitioning sequences, four classifiers, and three datasets. With so many components to the experimentation, averages can be misleading at times but overall are the most effective way to summarize the results. Section 4 breaks down those averages in as many ways as possible to account for this and to show the variability across problems and classifiers.

References

Blake, C., & Merz, C. (1998). UCI repository of machine learning databases.

Brownlee, J. (2018). How to predict room occupancy based on environmental factors. *Deep Learning for Time Series.*

Caruana, R., & Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. *Department of Computer Science, Cornell University.*

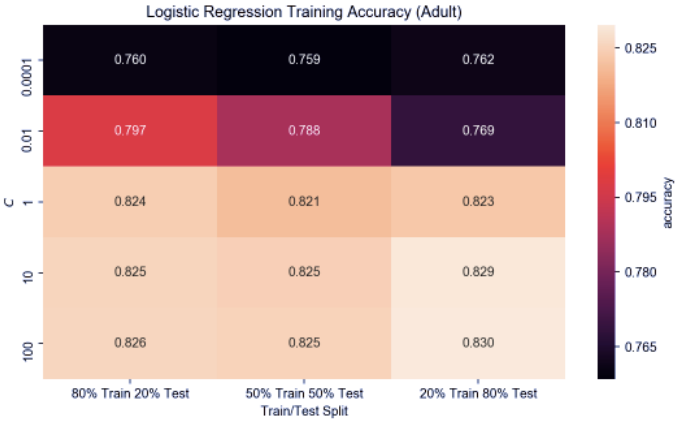
Pedregosa *et al.*, (2011). Scikit-learn: machine learning in python. *JMLR 12.*

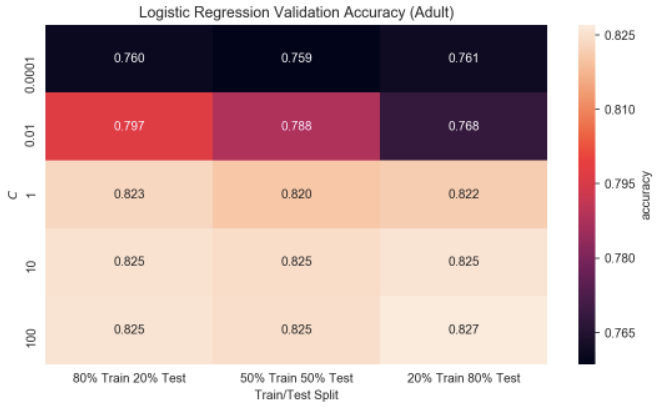
Zhuowen, T. (2019). Supervised machine learning algorithms: class notes & homework assignments. *Department of Cognitive Science, University of California San Diego.*

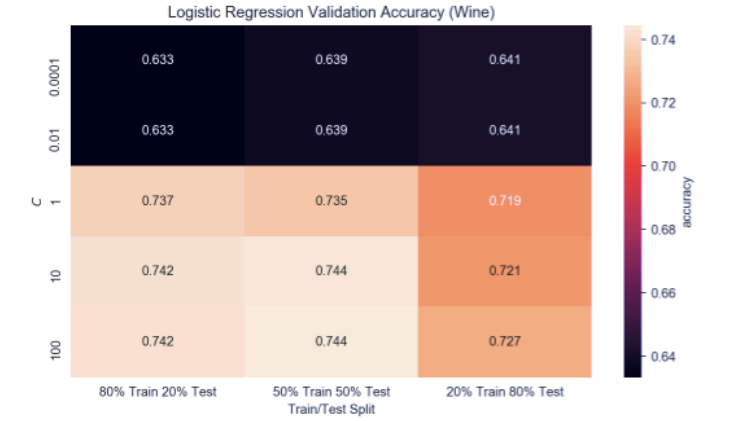
6. Appendix

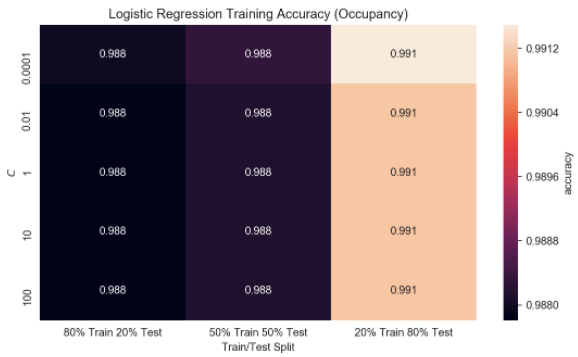
Heat maps for each classifier, problem, partition, for each hyper-parameter passed for the respective model.

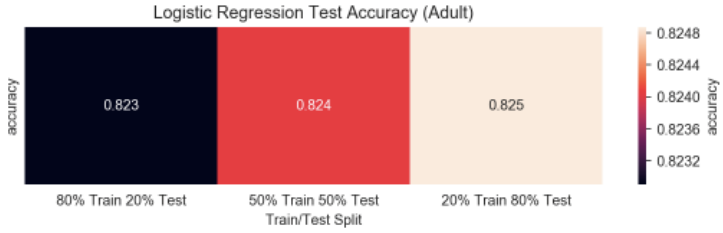
6.1 Logistic Regression

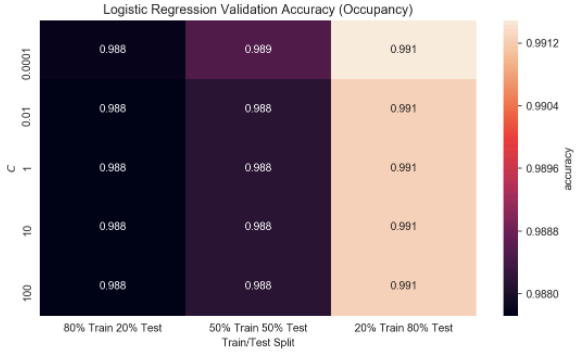


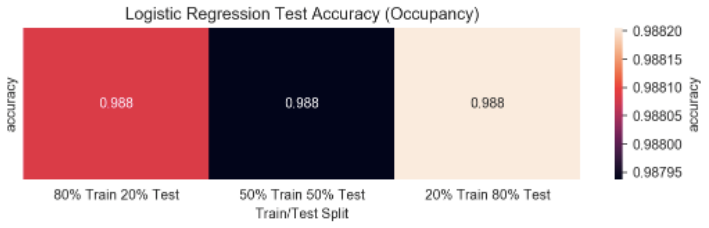


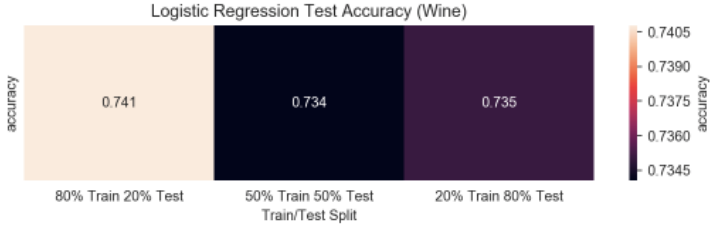




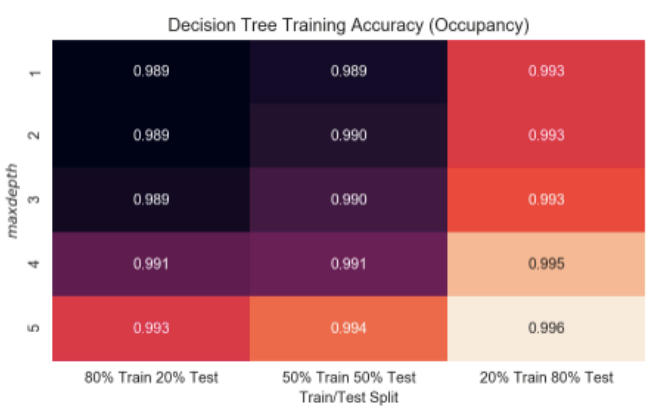


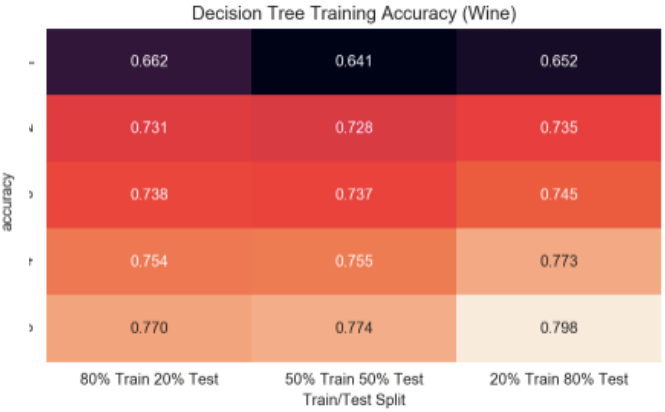


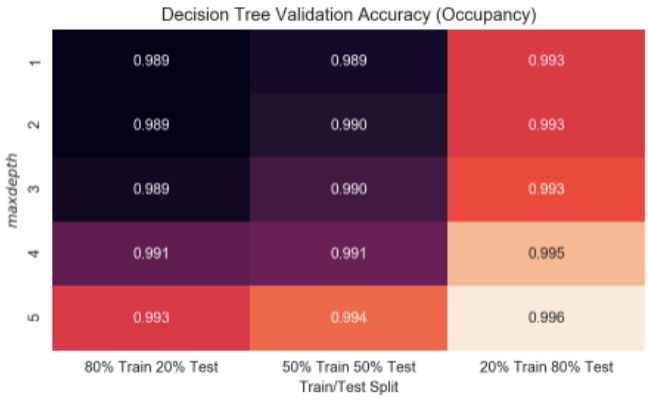


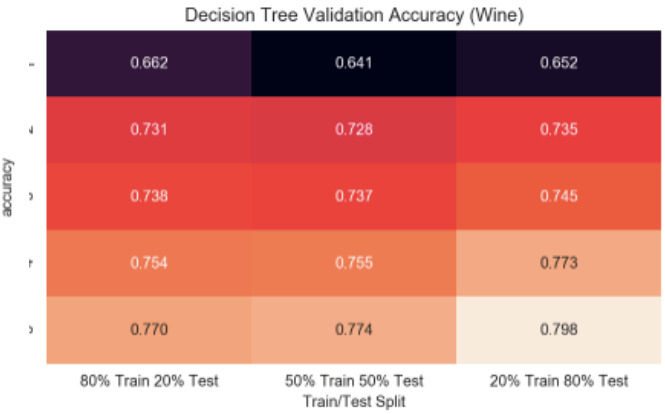


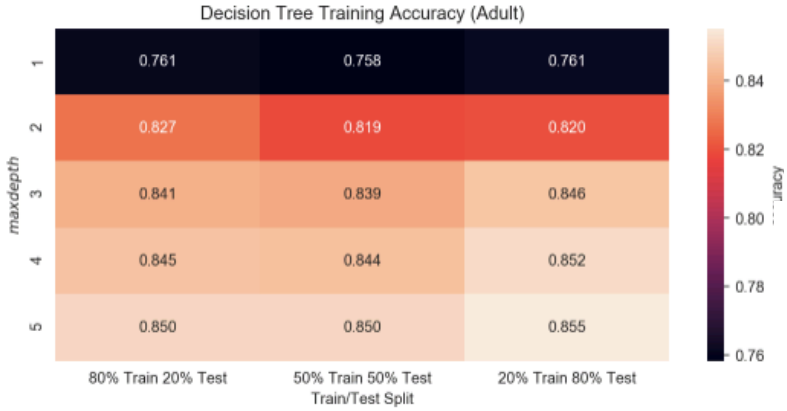
6.2 Decision Tree

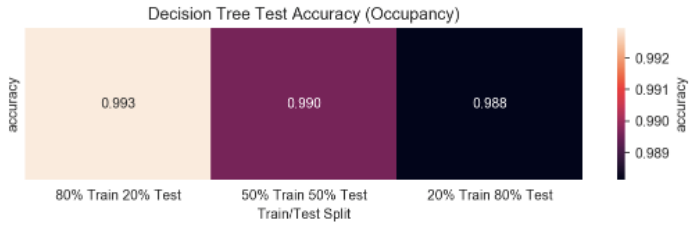


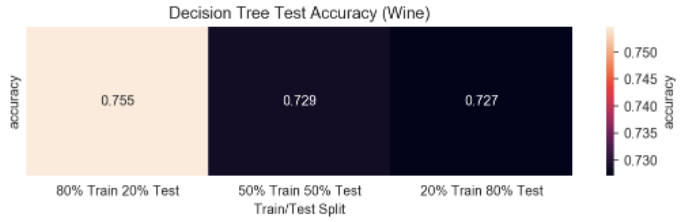


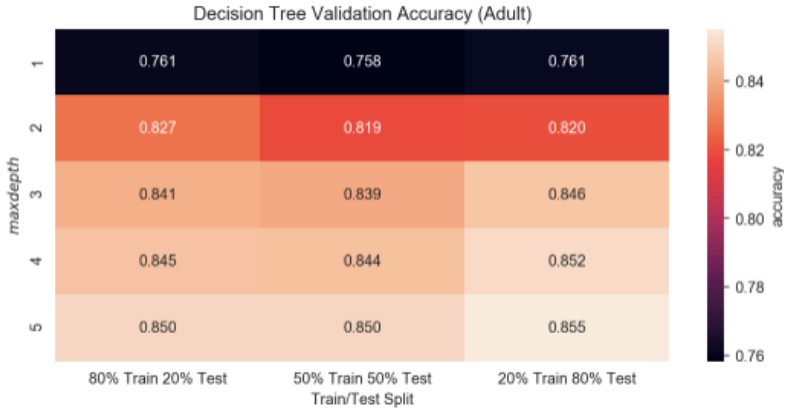


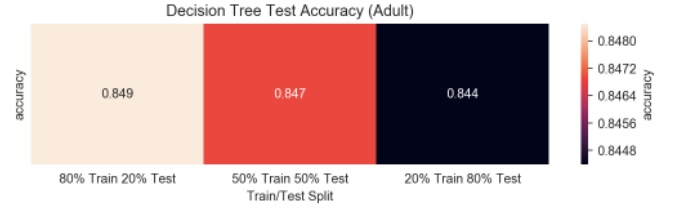




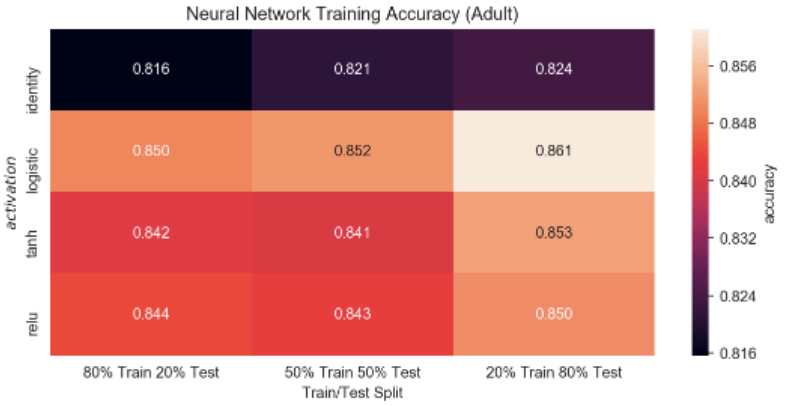


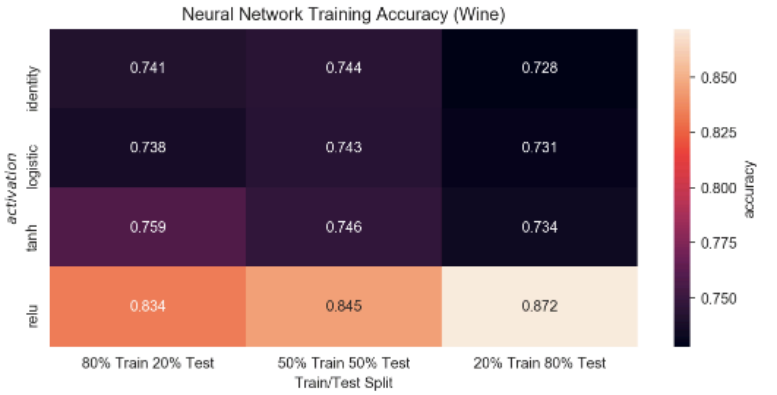


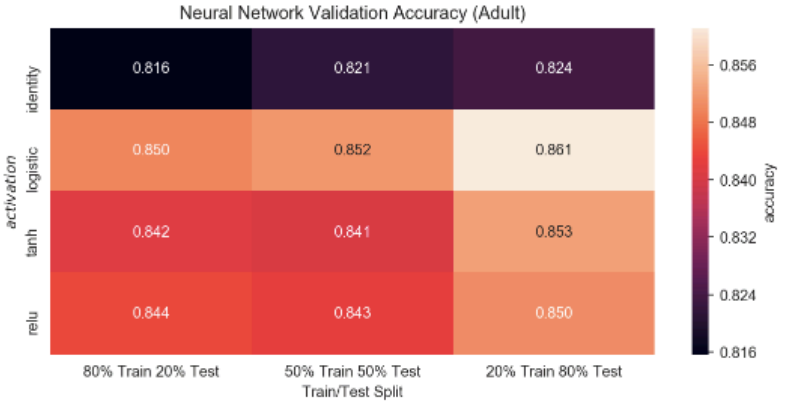


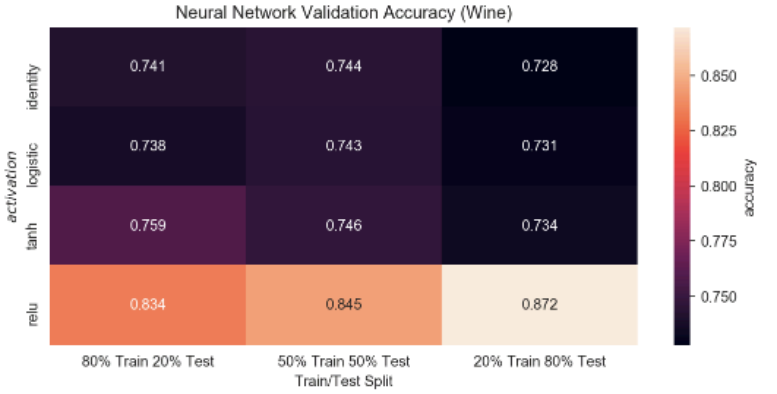


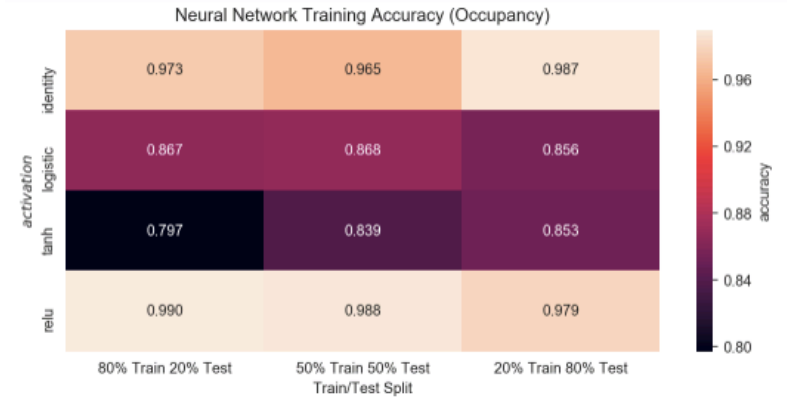
6.3 Multi-Layer Perceptron

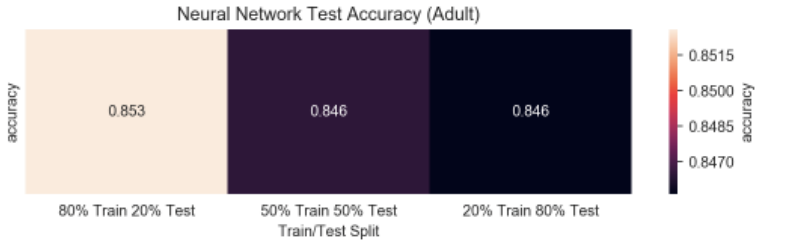


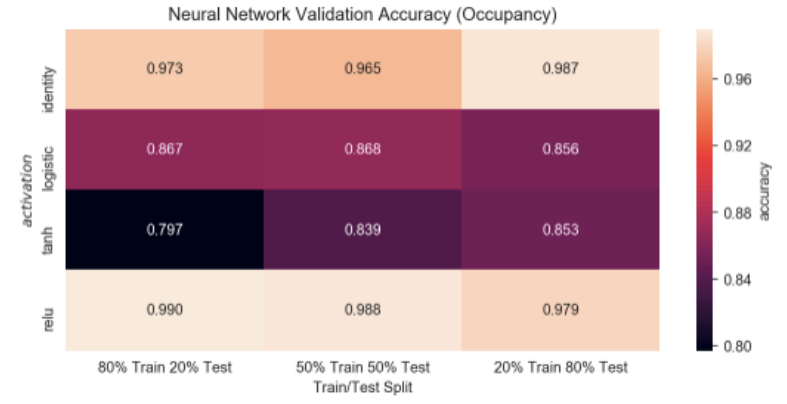


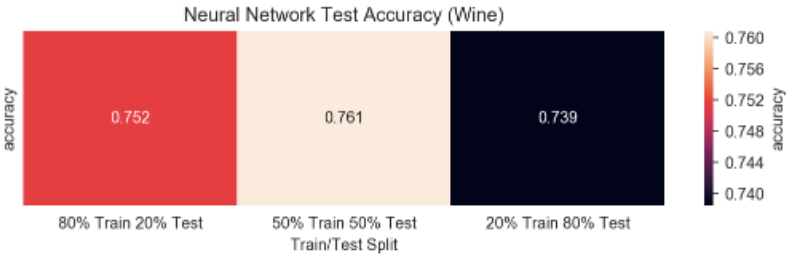


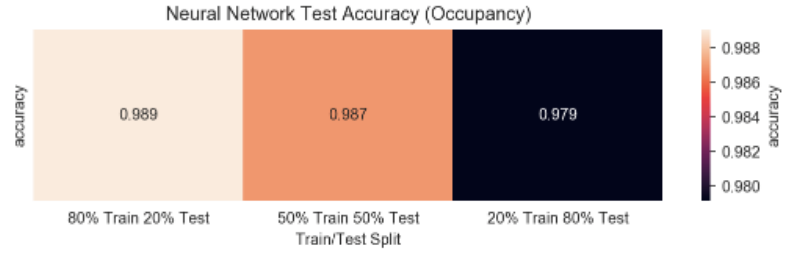




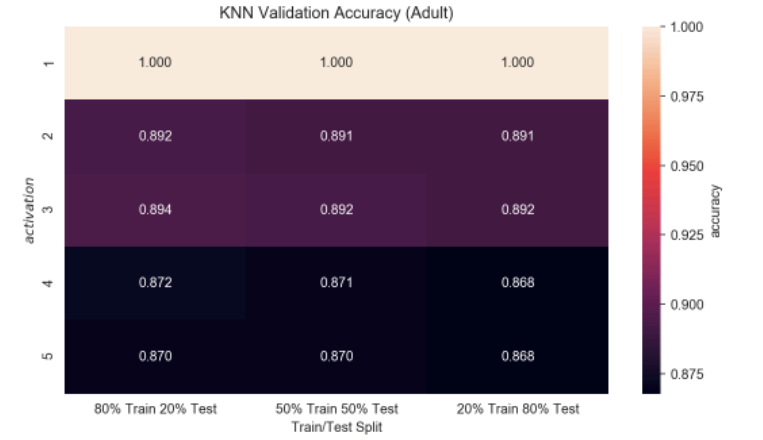


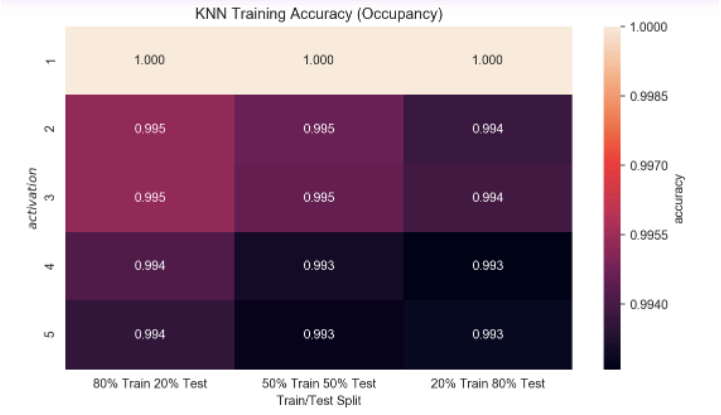


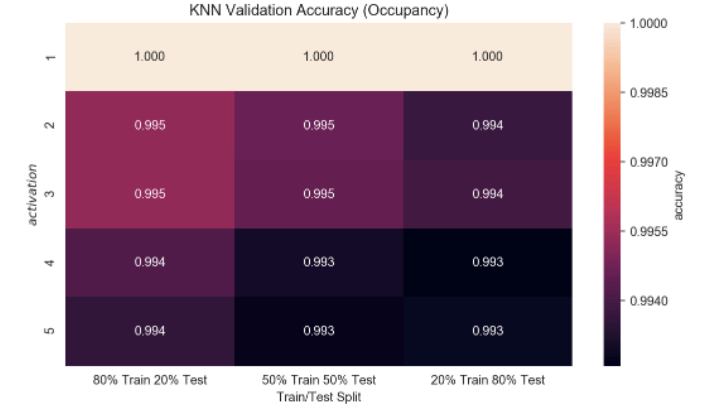


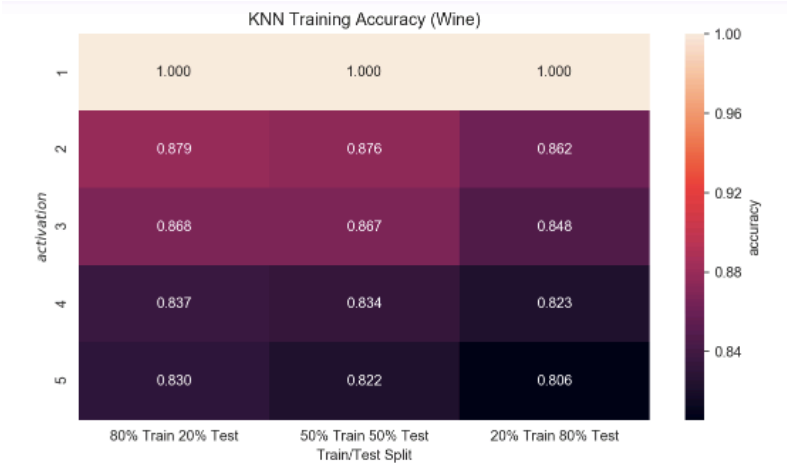


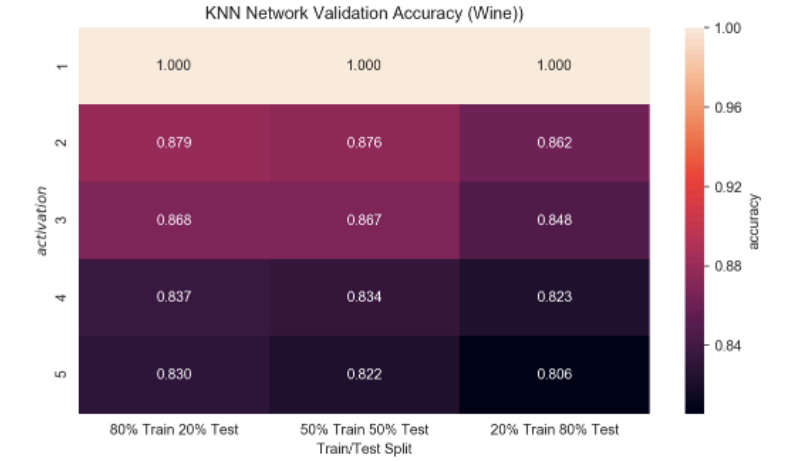
6.4 K Nearest Neighbors

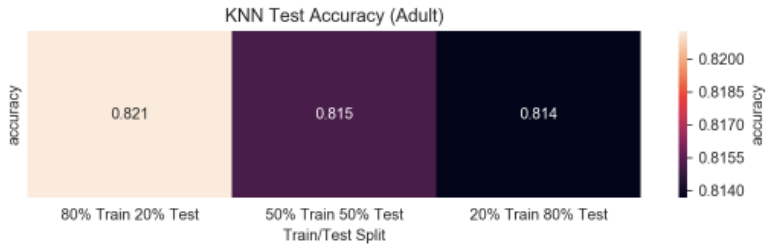


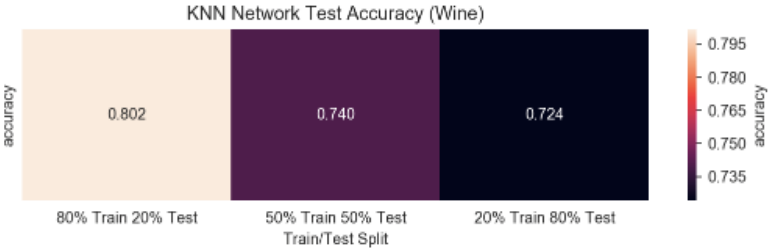


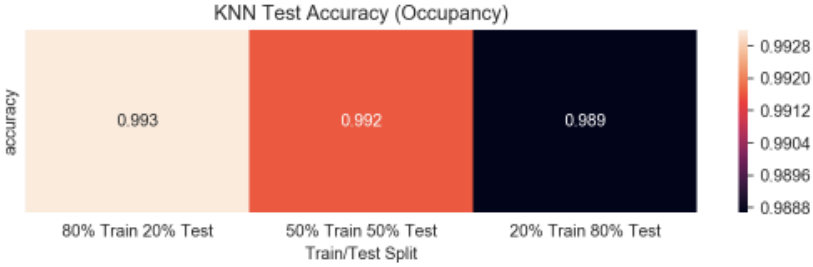












Bonus

The above analysis and comparison of the supervised learning algorithms logistic regression, decision tree, multi-layer perceptron, and k nearest neighbors on the classification problems income range, wine quality, and occupancy detection, partitioned three ways (80/20, 50/50, 20/80 train/test), and iterated three times was extended beyond basic requirements in several ways.

The experiment was conducted on an additional classifier, k nearest neighbors. This demonstration of additional effort was done strategically to solidify course concepts. Homework 6, particularly the implementation of KNN, was a challenge for me and I was unable to implement it successfully at the time. I wanted to develop KNN as my additional classifier in this study to challenge myself to learn about its functionality and make up for my previous difficulties. This proved rewarding considering the excellent performance of KNN in this study.

The ADULT data contained qualifies as a large dataset, making computations more expensive.

Furthermore, the classification algorithms implemented performed nearly perfectly on the Occupancy Detection problem.

Finally, as an additional demonstration of my experimental methodology, I have featured an Appendix containing the heat maps reporting the training, validation, and testing accuracies for each partition, problem, and classifier for each of the hyper-parameters.