Comparing Supervised Learning Algorithm Performance on Various Datasets

Rebecca Du
(rrdu@ucsd.edu)

Abstract

In this paper, multiple supervised learning algorithms are implemented to perform binary classification across different datasets of varying sizes. The algorithms are then compared to see which ones performed best. The findings of the classification experiments are further analyzed and future areas of research are proposed.

1. Introduction

In this final project, four datasets were used to assess the performance of the various supervised learning models. The datasets were obtained from the UC Irvine Machine Learning Repository (https://archive.ics.uci.edu/dataset/2/adult) and are listed as follows: Adult Income Prediction Dataset (Becker, 1996), Car Evaluation Dataset (Bohanec & Rajkovic, 1997), Heart Disease Prediction Dataset (from Cleveland) (Detrano et al, 1988), and Mushroom Prediction Dataset (Goethals et al, 1987). The datasets are henceforth referred to as ADULT, CAR, HEART, and MUSHROOM, respectively.

The code for the final project (including loading, formatting, processing, and classifying the datasets) was written in the Python programming language (https://www.python.org/). Various Python libraries were used to assist the coding process, including Numpy (http://www.numpy.org/), Pandas (https://pandas.pydata.org/), Scipy (https://www.scipy.org/), Sklearn (https://scikit-learn.org/). Additionally, the Matplotlib library (https://matplotlib.org/) was used to aid in data visualization. The code was written in Jupyter notebooks (https://jupyter.org/).

Four specific supervised learning algorithms were assessed in this project: Decision Tree classifier (DT), KNN classifier (KNN), Random Forest classifier (RF), and SVM Linear classification (SVM). These algorithms were coded based on frameworks provided in class and resources online.

The datasets and Jupyter notebooks used in this project can be found in the Github repository linked below.

https://github.com/dofutofu88/COGS-118A-Final-Project

2. Methodology

2.1 General Approach

The overall goal of this project was to follow the methodology outlined in Caruna & Niculescu-Mizil's paper - that is, to compare the performance of the aforementioned supervised learning algorithms on different datasets and assess which ones outperformed others, using the testing accuracy of the algorithms as the main point of comparison. The algorithms were tasked with solving one problem: to perform binary classification on the dataset and predict which of two categories each data point should go in. While ADULT was an inherently binary classification problem, the CAR, HEART, and MUSHROOM datasets were initially a multivariate classification problem that was converted to binary classification for consistency.

To ensure thorough testing, the four datasets were each partitioned according to three ratios of training to testing data - 20:80, 50:50, and 80:20. The optimal model of each classifier was obtained through Gridsearch. Each classifier was then evaluated on the previously mentioned partitions for three trials. Additionally, cross-validation was used to calculate testing accuracies and obtain the most optimal hyperparameters for each classifier. A heatmap function was defined to print out heatmaps of each classifier's training and testing accuracy with respect to the classifier's variables for every trial and partition. At the end of the training and testing rounds, the best training accuracy, average testing accuracy, and hyperparmeter for each classifier were recorded and printed out in a summary table. For the sake of conciseness, this description of the classification process is not repeated in the following sections describing each dataset. All datasets were classified using the same process mentioned above.

2.2 Adult Income (ADULT)

The ADULT dataset was created by scraping census income data from 1994. The goal of the dataset was to predict whether an adult's income would exceed $50,000 per year based on various parameters.

After cleaning up the dataset and dropping sensitive parameters (parameters such as race, which would be inappropriate to "rank", even if done arbitrarily), seven parameters remained. Two of the seven parameters were integer values (age and working hours per week), while the remaining five were categorical (workclass, education level, marital status, occupation, sex) and were thus ordinally encoded using the label encoder function of the Sklearn package. The dataset had an initial size of 48,842 instances but was cut down to 10,000 instances for the sake of brevity.

The dataset was inherently organized as a binary classification problem, with "Income" being the target variable. The outcomes were encoded as such: 0 represented a prediction of a salary below $50,000 per year and 1 represented a prediction of a salary above $50,000 per year.

2.3 Car Evaluation (CAR)

The goal of the CAR data is to assess whether or not the price of a car is acceptable given six features.

The features are buying price, maintenance, number of doors, number of persons the car can fit, trunk size, and safety. All of the six features were initially categorical and converted to integers through the use of the Sklearn label encoder function. The CAR dataset was of a smaller size, containing only 1,728 instances.

The CAR dataset was not inherently a binary classification problem; the target variable "Acceptability" was a ranked value on a four of 0 to 3, with 0 indicating unacceptable and 3 indicating high acceptability. To turn the dataset into a binary classification problem, all rankings above 0 (i.e. 1, 2, and 3) were encoded as acceptable and given the value 1, while 0 remained unacceptable and was given the value 0.

2.4 Heart Disease Prediction (HEART)

The HEART dataset was obtained from a Cleveland hospital database. The goal of the HEART dataset was to distinguish the presence of heart disease in a given patient.

The dataset has 13 features: age, sex (0 is female, 1 is male), cp (chest pain), trestbps (resting blood pressure), chol (serum cholesterol), fbs (if the fasting blood sugar > 120 mg/dl), restecg (resting ecg), thalach (maximum heart rate achieved), exang (exercise-induced angina), oldpeak (depression induced by exercise relative to rest), slope (the slope of the peak exercise segment), ca (number of major vessels colored by fluoroscopy), and thal (thalaseemia diagnosis). The features were a mix of categorical values (ca, thal) that had to be ordinally encoded by the Sklearn label encoder function and numerical values (age, sex, cp, trestbps, chol, fbs, restecg, thalach, exang, oldpeak, slope). The HEART dataset was the smallest, consisting of only 303 instances.

Like the CAR dataset, the target variable of the HEART dataset ("Num") was not inherently a binary classification problem. "Num" ranked the predicted presence of heart disease in a patient on a scale of 0 to 4, with 0 being no presence and 4 being a high presence of the disease. To change this to a binary classification problem, the values were manually encoded to either 0 (no presence of heart disease) or 1 (any presence of heart disease - i.e. 1, 2, 3, 4).

2.5 Mushroom Prediction (MUSHROOM)

The MUSHROOM dataset was obtained from the Audobon Society Field Guide, which described mushrooms in terms of physical characteristics. The goal of the MUSHROOM dataset was to determine whether a particular mushroom was poisonous or edible.

The MUSHROOM dataset had the greatest number of features across the four datasets tested, clocking in at 22 features. The features were: cap shape, cap surface, cap color, presence of bruises, odor, gill attachment, gill spacing, gill size, gill color, stalk shape, stalk root, stalk surface above ring, stalk surface below ring, stalk color above ring, stalk color below ring, veil type, veil color, ring number, ring type, spore print color, population, and habitat. All features save for two (cap-color and veil-type, which were binary) were categorical and thus ordinally encoded using the same Sklearn label encoder function as previous datasets. The MUSHROOM dataset was also the largest before being cut down - totaling 8,124 instances. The combination of many instances and features made it an ideal dataset to test the classifiers.

The MUSHROOM dataset's target feature was "poisonous", which was a categorical variable consisting of three classes: definitely edible, definitely poisonous, unknown edibility/not recommended. In labeling, the last trait was collapsed into the poisonous category so that the variable was encoded as such: 0 being edible, 1 being inedible (poisonous or unknown), thus allowing binary classification to be performed on the dataset.

3. Results

The performance of the classifiers in the training and testing sets is summarized in **Appendix A.1** in three tables:

- **Table 1** lists the best training accuracy for each classifier across the three partitions and trials, along with the optimal hyperparameter used to obtain the result.
- **Table 2** lists the average test accuracy for each classifier on the various datasets for each of the three partitions of training and testing data: 20:80, 50:50, and 80:20.
- **Table 3** summarizes the average test accuracy for each classifier on each dataset across all the aforementioned partitions.

As mentioned in Section 2, heatmaps were also created throughout the training and testing processes to track the performance of the classifiers. A sample heatmap is presented in **Appendix A.2**.

The data presented in **Table 1** indicates that the 20:80 partition of training to testing data resulted in the best training accuracy, with 9 of the 16 best training accuracies across the classifiers and datasets being from the 20:80 partition. Additionally, we can see that the KNN classifier performed the best in training for the ADULT, CAR, and MUSHROOM datasets, while the decision tree classifier performed the best in training for the HEART dataset.

Table 2 reveals that for testing the classifiers, the best partition was an 80:20 ratio of training to testing data, with 10 of the 16 best training accuracies across classifiers and datasets coming from that ratio.

From **Table 3**, it becomes evident that the random forest classifier performed the best for the ADULT and CAR datasets. For the HEART and MUSHROOM datasets, the SVM classifier performed the best (though the random forest classifier was a close second for both cases).

4. Discussion

For the most part, the results obtained from the classification processes were in line with what was discovered in Caruna & Niculescu-Mizil's paper, with the random forest classifier being one of the best-performing classifiers when measured based on testing accuracy. The influence of dataset size, feature number, and hyperparameter choice will be discussed in the following paragraphs.

First, to assess the influence of dataset size on classifier performance, we will compare the ADULT and CAR datasets. The ADULT dataset has approximately 10 times as many instances as the CAR dataset, and both datasets have approximately the same number of features (seven and six, respectively). The data reveals that the CAR dataset outperformed all other datasets in almost all the classifiers. Although this is not in line with the logic that the more data that is available to be used for training and testing, the better the model performance, it can be explained by the fact that only a portion of the ADULT dataset was used for brevity's sake. The portion that was chosen was of 10,000 randomly selected instances (a fourth of the entire dataset size), which may not provide as comprehensive a picture for model testing as using the entire dataset. In future experiments, it may be worthwhile to use the entire dataset to preserve trends in the data that might otherwise be lost by only using a fraction of the data.

Second, to observe the impact of parameter amount on classifier performance, the ADULT and MUSHROOM datasets will be compared. The two datasets have approximately the same size (10,000 instances and 8,124 instances). The ADULT dataset had seven parameters, while the MUSHROOM dataset had three times as many, totaling 21 parameters. The results indicate that the ADULT dataset outperformed the MUSHROOM dataset across all the classifiers. From this, the conclusion can be drawn that having too many parameters may negatively impact the performance of the classifier by causing overfitting.

Lastly, the data indicates that the optimal hyperparameter remains the same across all the datasets. Thus, the conclusion can be drawn that the classifiers reached the same result regarding the best hyperparameter for the datasets.

5. Bonus Points

Bonus points are requested for using more than the minimum required 3 datasets and 3 classifiers (4 datasets and 4 classifiers were used).

6. Conclusion

The results of the experiment were largely as expected and the same as what Caruna & Niculescu-Mizil discovered in their paper. The only anomaly that stood out was the underperformance of the ADULT dataset compared to the CAR dataset despite having many more instances, though this can be explained by not considering the entire dataset due to it being too large. Results indicated that the performance of a classifier on a given dataset is largely influenced by the size of the dataset, the number of features, and the hyperparameters that were chosen.

Future studies may choose to examine the performance of more classifiers on larger datasets, as some of the datasets used in this experiment were on the smaller side. Additionally, datasets with a clearer correlation between the features and labels could be chosen to better assess the performance of the classifiers.

References

Becker, B. & Kohavi, R. (1996). Adult. Retrieved December 12, 2023, from UCI Machine Learning Repository. https://doi.org/10.24432/C5XW20.

Bohanec, M. (1997). Car Evaluation. Retrieved December 12, 2023, from UCI Machine Learning Repository. https://doi.org/10.24432/C5JP48.

Caruna, R. & Niculescu-Mizil, A. (2006). *An Empirical Comparison of Supervised Learning Algorithms*

Janosi, A., Steinbrunn, W., Pfisterer, M. & Detrano, R. (1988). Heart Disease. Retrieved December 12, 2023, from UCI Machine Learning Repository. https://doi.org/10.24432/C52P4X.

Komatineni, S. & Miranda, V. (2018). *Using Supervised Learning Algorithms to Solve Several Problems*.

Mushroom. (1987). Retrieved December 12, 2023 from UCI Machine Learning Repository. https://doi.org/10.24432/C5959T.

A. Appendix

A.1. Tables

Table 1: Best Training Accuracy, Partition of Best Training Accuracy, and Best Hyperparameter

Best Training Accuracy, Partition of Best Training Accuracy, and Best Hyperparameter					
		Classifier			
		DT	KNN	RF	SVM
Dataset	ADULT	84.00% 20/80 D = 5	85.08% 50/50 K = 6	84.15% 20/80 D = 5	81.75% 20/80 C = 0.1
	CAR	93.99% 80/20 D = 5	97.48% 80/20 K = 5	96.30% 20/80 D = 5	74.42% 50/50 C = 1
	HEART	92.91% 20/80 D = 4	82.91% 20/80 K = 2	92.08% 20/80 D = 2	92.08% 20/80 C = 0.1
	MUSHROOM	67.17% 50/50 D = 5	75.29% 20/80 K = 4	68.75% 50/50 D = 5	68.58% 50/50 C = 0.1

Table 2: Average Test Accuracy Per Partition (20:80, 50:50, 80:20)

Average Test Accuracy Across Each Partition **1) 20:80 2) 50:50 3) 80:20**					
		Classifier			
		DT	**KNN**	**RF**	**SVM**
Dataset	**ADULT**	1) 82.36% 2) 82.46% 3) 82.73%	1) 78.40% 2) 79.57% 3) 80.76%	1) 82.30% 2) 82.42% 3) 82.75%	1) 81.34% 2) 81.28% 3) 81.46%
	CAR	1) 93.61% 2) 92.59% 3) 93.15%	1) 92.24% 2) 96.03% 3) 97.48%	1) 96.30% 2) 95.45% 3) 94.68%	1) 70.33% 2) 71.29% 3) 71.19%
	HEART	1) 77.36% 2) 75.00% 3) 77.59%	1) 58.84% 2) 60.52% 3) 57.37%	1) 80.79% 2) 82.89% 3) 84.15%	1) 77.22% 2) 82.01% 3) 90.32%
	MUSHROOM	1) 65.00% 2) 65.15% 3) 64.12%	1) 63.58% 2) 61.07% 3) 58.15%	1) 65.57% 2) 65.73% 3) 65.88%	1) 66.23% 2) 65.86% 3) 66.29%

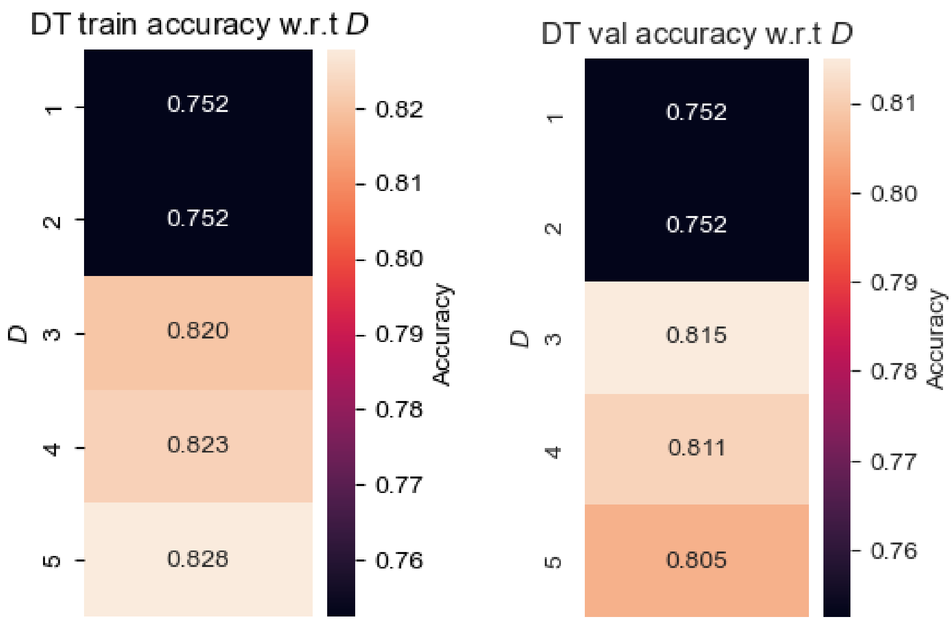
Table 3: Average Test Accuracy Across All Partitions

Average Test Accuracy Across All Partitions					
		Classifier			
		DT	**KNN**	**RF**	**SVM**
Dataset	**ADULT**	82.51%	79.57%	82.49%	81.36%
	CAR	93.11%	95.25%	95.47%	70.93%
	HEART	76.65%	58.91%	82.61%	83.18%
	MUSHROOM	64.75%	60.93%	65.72%	66.13%

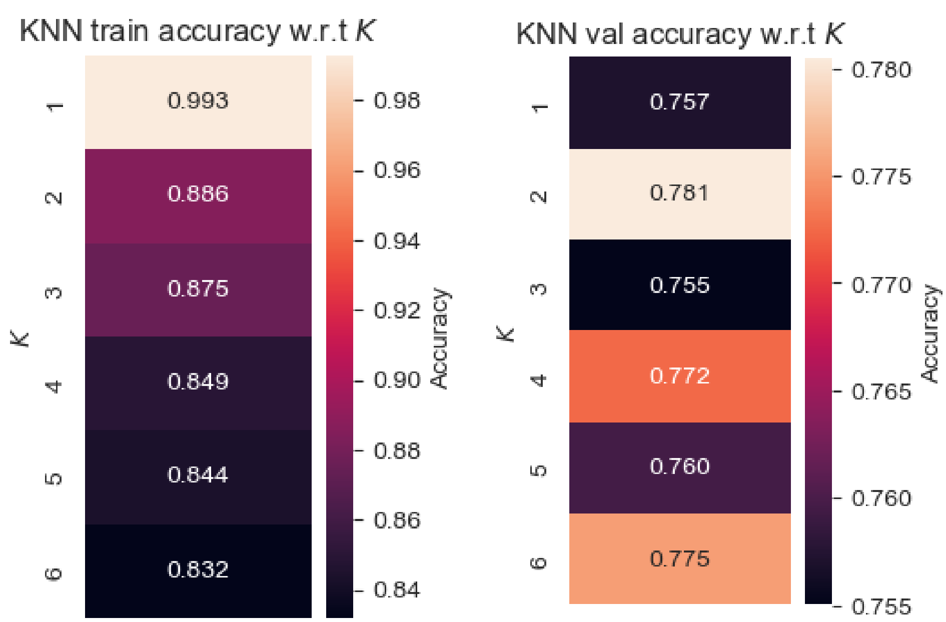
A.2. Sample Heatmaps

The following are heatmaps obtained from the ADULT dataset for each classifier performed on the 20/80 partition. The complete list of heatmaps for each dataset, classifier, and partition can be found in their respective Jupyter notebooks in the Github repository.

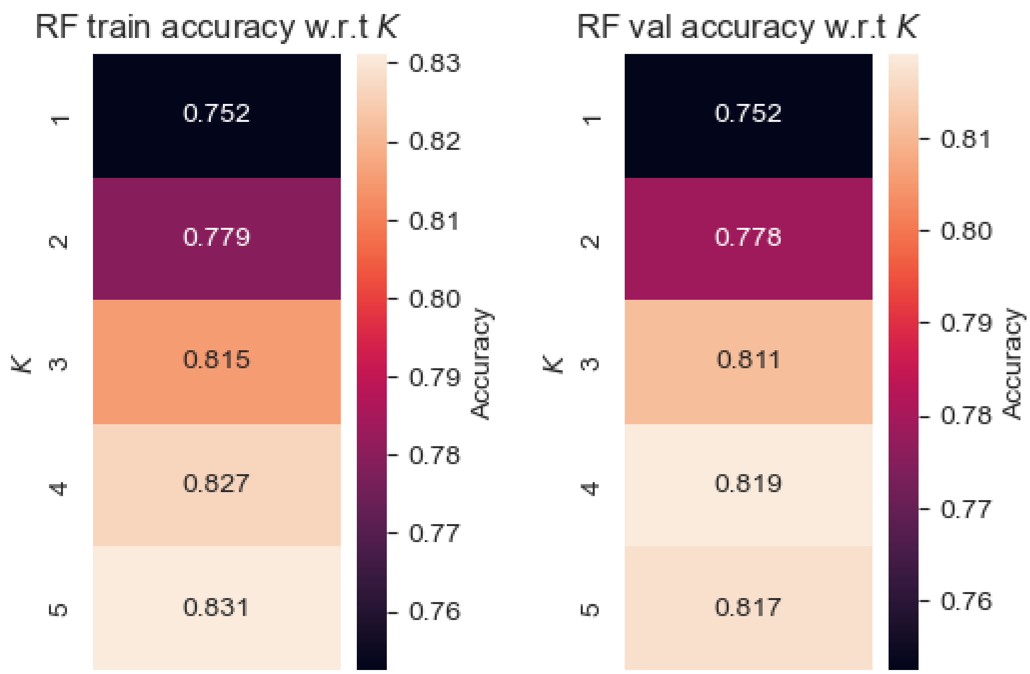
Decision Tree (DT)



KNN



Random Forest (RF)



SVM Linear