An Empirical Study of Supervised Learning Algorithms

Dalila Solis
University of California, San Diego
d1solis@ucsd.edu

Abstract

Over the past few years, there have been a plethora of machine-learning methods that have been developed. Briefly looking at the horizon of machine learning, it seems as though this never-ending list of methods poses a great challenge to researchers to essentially select the most efficient algorithm for certain tasks. In this paper, I specifically dive into an empirical comparison of Support Vector Machines (SVM), K-Nearest Neighbors (KNN), and Decision Trees on the Iris, Car Evaluation, and Zoo datasets. Our objective is to assess the performance of these algorithms under various training/testing split scenarios (20/80, 50/50, and 80/20) and to understand the impact of hyperparameter tuning on classification accuracy.

1. Introduction

For the final project, three different classifiers were utilized on three various datasets that were obtained from the UC Irvine Machine Learning Repository. The three datasets that were explored were the Iris Dataset (Fisher, R. A.. (1988)), the Car Evaluation Dataset (Bohanec, Marko. (1997)), and the Zoo Dataset (Forsyth, Richard. (1990)). These datasets are later referred to as IRIS, CAR, and ZOO in this paper.

The supervised learning algorithms that were used on the IRIS, CAR, and ZOO datasets were Support Vector Machines, K-Nearest Neighbors, and Decision Trees. The datasets were essentially loaded and processed through Visual Studio Code using Python programming language. This includes the use of necessary libraries such as Pandas, Numpy, and Scipy as well as Sklearn for implementing the machine-learning methods.

2. Method

This study's primary goal was to assess each learning algorithm's performance on a given dataset. Hyper-parameters and cross-validation were also implemented to obtain reliable estimates of a model's performance for each classification that was used. During training and testing, it was necessary to incorporate different data partitions such as 20% training and 80% testing, 50% training and 50% testing, as well as 80% training and 20% testing for three trials. This led to computing average scores to remove the potential of having accidental results.

3. Experiment

a. IRIS

The Iris dataset is a small classic dataset from 1936 by Fisher. It is one of the earliest known datasets used for evaluating classification methods. It represents three classes of 50 instances each, where each class refers to a type of iris plant—Setosa, Versicolour, or Virginica. The dataset

An Empirical Study of Supervised Learning Algorithms

Dalila Solis
University of California, San Diego
d1solis@ucsd.edu

contains four real-valued features: sepal length, sepal width, petal length, and petal width. While one class is linearly separable from the other two, the latter are not linearly separable from each other. The overall goal is to predict the class of the iris plant.

In addition to the dataset containing no missing values as well as the four real-valued features being in a suitable format already, this allowed for a general focus on using label encoding on the categorical target feature into numerical labels for easier data processing. Thus, the categories were labeled as such: “Iris Setosa” to (0), “Iris Versicolour” to (1), and “Iris Virginica” to (2).

The classifiers used on this dataset were SVM, KNN, and Decision Trees. Cross-validation was involved with the data partitions of 20/80, 50/50, and 80/20. Each partition was repeated three times to capture the differences in the results.

For the IRIS dataset, the performance of each classifier was recorded and displayed in an individual table under **Appendix A.1 Table 1**. This is where the results of the training, testing, and validation accuracies are held along with the most favorable hyperparameters for each partition. It is evident that from **Table 1**, Each split under SVM resulted in the best performance for this dataset in particular, thus concluding that the entirety of SVM held most suitable.

b. CAR

The Car Evaluation dataset is derived from a simple hierarchical decision model and evaluates cars based on various attributes, which include buying price, maintenance costs, technical characteristics, and safety features. The dataset contains 1728 instances with 6 features, removes the structural information, and directly relates car acceptability to six attributes: buying, maintenance, number of doors, capacity, luggage boot size, and safety. The dataset is categorized as multivariate and intended for classification.

Moreover, the dataset contained no missing values. So the next approach was to look at the seven features which were all categorial. It was essential to utilize ordinal encoding in order for the data to be formatted and be easier to process. As an example, we see that the categories for the ‘buying’ column are labeled as such: “vhigh” to (4), “high” to (3), “med” to (2), and “low” to (1). It was also necessary to transform our target feature using binary classification where our labels are “unacc”, “acc”, “good”, and “vgood” in which the first label is classified as false/negative which is denoted as (0) and the last three labels are classified as true/positive and are denoted as (1).

The classifiers used on this dataset were SVM, KNN, and Decision Trees. Cross-validation was involved with the data

An Empirical Study of Supervised Learning Algorithms

Dalila Solis
University of California, San Diego
d1solis@ucsd.edu

partitions of 20/80, 50/50, and 80/20. Each partition was repeated three times to capture the differences in the results.

For the CAR dataset, the performance of each classifier was recorded and displayed in an individual table under **Appendix A.1 Table 2**. This is where the results of the training, testing, and validation accuracies are held along with the most favorable hyperparameters for each partition. It is evident that from **Table 2**, 80/20 under Decision Trees resulted in the best performance for this dataset in particular and Decision Trees overall.

c. ZOO

The Zoo dataset is derived from a simple hierarchical decision model. It has 101 instances and 16 features, including 17 Boolean-valued attributes. This artificial dataset classifies animals into seven classes based on a variety of features such as hair, feathers, eggs, milk, etc. The classes represent mammals, birds, reptiles, fish, amphibians, insects, etc. Similar to CAR, the Zoo dataset is intended for classification.

Furthermore, since the dataset did not have any missing values, the shift of focus was primarily toward handling the binary, boolean, and categorical features. For features with binary values such as “hair” or “feathers”, ordinal encoding was the most suitable to maintain a specific order for each distinct category. Since binary classification is being dealt with, (0) as false and (1) as true were utilized. The “legs” feature was one-hot encoded to capture the different leg counts where six binary columns were created, each corresponding to a different leg count.

The classifiers used on this dataset were SVM, KNN, and Decision Trees. Cross-validation was involved with the data partitions of 20/80, 50/50, and 80/20. Each partition was repeated three times to capture the differences in the results.

For the ZOO dataset, the performance of each classifier was recorded and displayed in an individual table under **Appendix A.1 Table 3**. This is where the results of the training, testing, and validation accuracies are held along with the most favorable hyperparameters for each partition. It is evident that from **Table 3**, 80/20 under SVM resulted in the best performance for this dataset in particular and SVM overall.

4. Conclusion

From the results, it can be concluded that SVM was the most successful in terms of performance for the IRIS and ZOO datasets, though was deemed as unsuccessful for the CAR dataset. Meanwhile, KNN performed nearly as well in terms of accuracy for IRIS and ZOO datasets. We can also make the same statement for the Decision Tree classifier as it seemed to perform just as well for each dataset.

An Empirical Study of Supervised Learning Algorithms

Dalila Solis
University of California, San Diego
d1solis@ucsd.edu

An observation that can be made from all three datasets is how the performance of the classifiers depends on the ratio of training to testing data. It is clear that from the tables provided, having a balanced dataset could contribute to a better generalization through the use of classifiers and partitions. This also includes the number of instances and features in a particular dataset as well as whether or not the dataset has any clear missing values.

5. References

Bohanec, Marko. (1997). "Car Evaluation." UCI Machine Learning Repository, https://archive.ics.uci.edu/dataset/19/car+evaluation

Fisher, R. A. (1988). "Iris." UCI Machine Learning Repository, https://archive.ics.uci.edu/dataset/53/iris

Forsyth, Richard. (1990). "Zoo." UCI Machine Learning Repository, https://archive.ics.uci.edu/dataset/111/zoo

Dua, D. & Karra Taniskidou, E. (2017). "Welcome to the UC Irvine Machine Learning Repository." UCI Machine Learning Repository, https://archive.ics.uci.edu/

An Empirical Study of Supervised Learning Algorithms

Dalila Solis
University of California, San Diego
d1solis@ucsd.edu

A. Appendix

A.1. Table Results

Table 1: IRIS dataset

Classifier	Splits	Training Accuracy	Testing Accuracy	Validation Accuracy	Hyperparameters
SVM	20/80	1.0000	1.0000	1.0000	{'C': 1.0}
SVM	50/50	1.0000	1.0000	1.0000	{'C': 1.0}
SVM	80/20	1.0000	1.0000	1.0000	{'C': 1.0}
KNN	20/80	1.0000	1.0000	0.9667	{'n_neighbors': 5}
KNN	50/50	1.0000	1.0000	0.9867	{'n_neighbors': 5}
KNN	80/20	1.0000	1.0000	1.0000	{'n_neighbors': 5}
Decision Trees	20/80	1.0000	0.9333	0.9667	{Max Depth: None, Min Samples Split: 2, Criterion: 'gini'}
Decision Trees	50/50	1.0000	1.0000	1.0000	{Max Depth: None, Min Samples Split: 2, Criterion: 'gini'}
Decision Trees	80/20	1.0000	1.0000	1.0000	{Max Depth: None, Min Samples Split: 2, Criterion: 'gini'}

An Empirical Study of Supervised Learning Algorithms

Dalila Solis
University of California, San Diego
d1solis@ucsd.edu

Table 2: CAR dataset

Classifier	Splits	Training Accuracy	Testing Accuracy	Validation Accuracy	Hyperparameters
SVM	20/80	0.7594	0.7404	0.7507	{'C': 1.0}
SVM	50/50	0.7488	0.7523	0.7442	{'C': 1.0}
SVM	80/20	0.7554	0.7312	0.7467	{'C': 1.0}
KNN	20/80	0.9130	0.8532	0.8174	{'n_neighbors': 5}
KNN	50/50	0.9826	0.9398	0.9155	{'n_neighbors': 5}
KNN	80/20	0.9841	0.9595	0.9580	{'n_neighbors': 5}
Decision Trees	20/80	1.0000	0.9754	0.9536	{Max Depth: None, Min Samples Split: 2, Criterion: 'gini'}
Decision Trees	50/50	1.0000	0.9931	0.9780	{Max Depth: None, Min Samples Split: 2, Criterion: 'gini'}
Decision Trees	80/20	1.0000	0.9971	0.9899	{Max Depth: None, Min Samples Split: 2, Criterion: 'gini'}

An Empirical Study of Supervised Learning Algorithms

Dalila Solis
University of California, San Diego
d1solis@ucsd.edu

Table 3: ZOO dataset

Classifier	Splits	Training Accuracy	Testing Accuracy	Validation Accuracy	Hyperparameters
SVM	20/80	1.0000	0.9913	0.9485	{'C': 1.0}
SVM	50/50	1.0000	0.9861	0.9929	{'C': 1.0}
SVM	80/20	1.0000	1.0000	0.9956	{'C': 1.0}
KNN	20/80	1.0000	0.9870	0.9318	{'n_neighbors': 5}
KNN	50/50	0.9930	0.9861	0.9929	{'n_neighbors': 5}
KNN	80/20	0.9956	1.0000	0.9869	{'n_neighbors': 5}
Decision Trees	20/80	1.0000	0.9130	0.9318	{Max Depth: None, Min Samples Split: 2, Criterion: 'gini'}
Decision Trees	50/50	1.0000	0.9861	0.9788	{Max Depth: None, Min Samples Split: 2, Criterion: 'gini'}
Decision Trees	80/20	1.0000	1.0000	0.9868	{Max Depth: None, Min Samples Split: 2, Criterion: 'gini'}