Analysis of Supervised Machine Learning Models

Alec Slim

Abstract

There have been many researches done on supervised machine learning models in the past few years . This paper provides a detailed analysis of the implementation and performance of three supervised machine learning algorithms on multiple datasets. This paper focuses on comparing the performance and accuracy of three different classifiers: Random Forest, K-Nearest Neighbors(KNN) and Support Vector Machines(SVM). The data sets used came from the UCI Machine Learning Repository and include the Occupancy Detection, Abalone, and Wine Quality datasets. The results were evaluated using metrics such as accuracy and cross-validation scores. The findings hope to help select the appropriate models for the binary classification use case.

1. Introduction

This study is inspired by Rich Caruana and Alexandru Niculescu-Mizil's study "An Empirical Comparison of Supervised Learning Algorithms". The three models were chosen from their list of models used but does not contain the same datasets allowing a new prospective on the topic. Supervised machine learning techniques have become essential in solving classification and regression problems. These algorithms allow for predictive modeling based on labeled datasets, where the goal is to create a function that is able to accurately predict the class of given data. This study evaluates the performance of three popular classifiers: Random Forest, KNN, and SVM. These algorithms are very popular used due to their distinct strengths. Random Forest excels in handling large datasets with complex structures, KNN offers simplicity and effectiveness for smaller datasets, and SVM provides powerful classification capabilities in high-dimensional spaces.

2. Method

2.1. Data Sets

The datasets used in this analysis originally were not made for the use case of binary classification but were transformed to make it possible.The OCCUPANCY Detection dataset already had binary targets so there was no change made there. The WINE QUALITY data set contained values from 1-10 as its targets so it was changed to a value of 1 when the target was greater than or equal to 7 and 0 otherwise. The ABALONE dataset contained ring counts(+1.5 for age) as its target, which were changed to 1 if the age was less than 10 and 0 otherwise. Irrelevant features such as the date were removed to focus the analysis on other readings. Missing values were cleaned to ensure integrity in the data.

2.2. Learning Algorithms

This section gives a description of all the learning algorithms that were used and their parameters

2.2.1. RANDOM FOREST

I used the Sci-kit Learn implementation of random forest. The number of trees ranged from values of 100, 150, 200. The low number of trees was chosen due to resources and time limitations, but it is known that higher values can often yield better results. More accurate results can be made with higher values but should be cautious of the possibility of over fitting. The max depth ranged from 5, 10, 15 and the minimum sample split ranged from 5, 10, 15. The max features parameter contained 'sqrt' and 'log2'.

2.2.2. K-NEAREST NEIGHBORS

I used the Sci-kit Learn implementation of K-Nearest Neighbors. The number of neighbors ranged from 2, 5, 7, 10 while the metric parameter included euclidean and manhattan as the options.

2.2.3. SUPPORT VECTOR MACHINE

I used the Sci-kit Learn implementation of Support Vector Machines. The C parameter ranged from 0.1, 1, 10. The kernel parameter contained linear and rbf as the options. The gamma values were either scale or auto.

2.3. Performance metrics

I used accuracy as my performance metric to determine how well the models performed. The three accuracies that I reported are training, testing, and cross validation accuracy. each accuracy is an average of three trials in order to avoid any unusual outputs. Out of the three accuracies cross validation is the most important one to look at as it gives the best insight into the real accuracy of the model.

2.4. Experiment

Each algorithm was trained, tested, and validated on all of the datasets and also on three separate train/test splits: 80/20, 50/50, and 20/80. The inclusion of multiple train/test splits allows for a better analysis of how the dataset size and composition affects the model performance. For each of the trials the hyper parameters were first optimized using cross validation on the training set and then the model was trained on the entire training data. From there the models got tested on the test data and was cross validated to achieve each of the performance metrics. All the steps after obtaining the optimal hyper parameters were repeated three times and the metrics were averaged before being returned. Along with the performance metric a bar graph of the top 10 best hyper parameters and a heat map of the hyper parameters were made.

2.4.1. PERFORMANCE BY DATA SET

The performance of all the trials can be seen in the chart below. For the occupation dataset random forest and knn had very similar results by both achieving the same validation scores for all data splits. SVM did not get similar scores except on the 20 percent split. For the ABALONE dataset the random forest model performed the best with a validation score of 0.848 while knn came in second and svm in last. For the WINE dataset random forest was the best performing with an accuracy of 0.875 on the 80 percent split.

Table 1. Accuracy scores of each model by data split

Model	OCC 80	OCC 50	OCC 20	ABAL 80	ABAL 50	ABAL 20	WINE 80	WINE 50	WINE 20
RF(TRAIN)	0.998	0.998	0.998	0.909	0.863	0.883	0.990	0.990	0.947
RF(VAL)	0.992	0.991	0.988	0.848	0.847	0.839	0.875	0.867	0.845
RF(TEST)	0.993	0.992	0.991	0.847	0.839	0.837	0.880	0.860	0.837
KNN(TRAIN)	0.994	0.993	0.990	0.865	0.876	0.861	0.930	0.924	0.848
KNN(VAL)	0.992	0.991	0.988	0.844	0.840	0.838	0.854	0.846	0.836
KNN(TEST)	0.993	0.991	0.990	0.848	0.839	0.824	0.838	0.836	0.822
SVM(TRAIN)	0.989	0.989	0.988	0.852	0.845	0.849	0.880	0.891	0.846
SVM(VAL)	0.989	0.989	0.988	0.844	0.845	0.847	0.837	0.837	0.829
SVM(TEST)	0.990	0.989	0.989	0.850	0.835	0.830	0.849	0.840	0.823

2.5. Conclusion

The results conclude that random forest performed the best with knn coming as a close second and svm being a lacking model compared to the other two for this case. Random forest can be seen as a good general choice for binary classification while knn can be the best option under certain circumstances and svm is seen as unfavorable unless a specific case that requires it arises. The results of this experiment match research trials that have been done which was expected. The accuracy of the models can be further improved upon given more resources and time. The addition of more datasets and models is another way this can be expanded upon to gain a wider understanding of what models perform the best for binary classification. Future studies could also explore the possibility of integrating feature engineering techniques and advanced ensemble methods to further optimize the performance of the models.

References

Candanedo, Luis. 2016. Occupancy Detection . UCI Machine Learning Repository. https://doi.org/10.24432/C5X01N.

Nash, Warwick, Tracy Sellers, Simon Talbot, Andrew Cawthorn, and Wes Ford. 1994. Abalone. UCI Machine Learning Repository. https://doi.org/10.24432/C55C7W.

Cortez, Paulo, A. Cerdeira, F. Almeida, T. Matos, and J. Reis. 2009. Wine Quality. UCI Machine Learning Repository. https://doi.org/10.24432/C56S3T.

Caruana, Rich, and Alexandru Niculescu-Mizil. ”An Empirical Comparison of Supervised Learning Algorithms.” *Proceedings of the 23rd International Conference on Machine Learning (ICML 2006)*, Pittsburgh, PA, USA, June 25–29, 2006, 161–168. Association for Computing Machinery (ACM). https://doi.org/10.1145/1143844.1143865.