Supervised Learning Algorithm Comparison

Mark Choe A13917840
University of California, San Diego

Abstract

The main purpose of this paper is to compare and contrast the performances of different supervised learning algorithms upon different datasets. This report follows a previous study by . The supervised learning algorithms included in this study Caruana and Niculescu-Mizil. were a linear support vector classification (Linear SVC), a k-nearest neighbors algorithm (KNN), decision trees, random forests, and a multi-perceptron layer neural network (MLP). Grid search functions were also used in conjunction with these algorithms in order to optimize some of the parameters of these algorithms.

1. Introduction

As hinted at before, studies have already been conducted to compare and contrast different supervised learning algorithms, notably “An Empirical Comparison of Supervised Learning Algorithms” by Caruana and Niculescu-Mizil. However, slight changes in implementations and parameters can reveal new insights or reinforce previous findings. This understanding lays the foundation for this project.

This paper utilizes five different classifiers: linear SVCs (an SVM), k-nearest neighbors, decision trees, random forests, and a MLP neural network (similar to the classifiers used in the Caruana & Niculescu-Mizil study). These classifiers are applied to three different UCI datasets: ADULT, COV_TYPE, and LETTER (Dua & Karra, 2017). On top of being applied to each dataset, each classifier is fit and tested with three different splits of the dataset- the training set comprising 80% of the dataset, 50% of the dataset, and 20% of the dataset (differing from the splits used in the Caruana & Niculescu-Mizil study). Multiple trials of these runs were also conducted and averaged to provide the final results.

2. Data and Problem

As mentioned before, the three datasets used in this report were ADULT, COV_TYPE, and LETTER, which happen to be the same UCI datasets that the Caruana and Niculescu-Mizil study uses as well. All of these datasets do not contain missing values, but lack binary labels (ones and zeroes in the labels column).

The ADULT dataset consists of features pertaining to adult working people (race, marital status, education), and attempts to label whether or not an individual will make more than $50,000 or not. Converting these labels into a more usable binary classification was simply a matter of changing the “>50k” label into a one and the rest of the labels into a zero.

The COV_TYPE dataset consists of features pertaining to geographical features of a forest, and attempts to label the tree cover type of that forest. Following the example of the Caruana and Niculescu-Mizil study, the most common label is converted to a positive one and the rest of the labels are converted to a negative one in order to create a binary classification problem (Caruana & Niculescu-Mizil, 2006).

The LETTER dataset consists of features pertaining to the location and shape of pixel displays, and attempts to label a capital letter of the english alphabet. Following the example of the Caruana and Niculescu-Mizil study, the labels of the letters “A” through “M” are converted to a positive one and the rest of the labels are converted to a negative one in order to create a binary classification problem (Caruana & Niculescu-Mizil, 2006). Going further in-depth, this conversion is achieved by one-hot encoding the letters, taking the array location of the one from the one-hot encoding array, and finally separating the first thirteen letters from the rest of the letters.

Table 1: Description of Problem and Data

Datasets	Number of Datapoints	Number of Features	Number of Features (One-Hot)	Training Size: Testing Size 80/20 Split	Training Size: Testing Size 50/50 Split	Training Size: Testing Size 20/80 Split
ADULT	32561	14	104	5000:1250	3125:3125	1250:5000
COV_TYPE	581012	54	54	5000:1250	3125:3125	1250:5000
LETTER	20000	16	16	5000:1250	3125:3125	1250:5000

3. Method

When implementing an SVM, a linear SVC was used. The regularization parameter were varied in order to identify an optimal parameter for testing. Specifically, this list of Cs, the regularization parameter, varied by factors of ten and ranged from 10^{-5} to 10^{-1} (a slightly more narrow range than Caruana and Niculescu-Mizil study).

When implementing a KNN, the number of neighbors is varied in order to identify an optimal parameter for testing. Specifically, this list of nearest of neighbors varied by factors of one and ranged from one to twenty six (as per the Caruana and Niculescu-Mizil study). Other parameters tested in the Caruana and Niculescu-Mizil study were not varied, however.

When implementing the decision tree classifiers, the maximum depth of the decision tree was varied in order to identify an optimal parameter for testing. Specifically, this list of maximum depths varied by factors of five and ranged from five to twenty five. These maximum depths were very loosely based upon the Caruana and Niculescu-Mizil study, unlike the parameters of the other classifiers, because, although splitting criterion was mentioned by the study (related to the depth of the decision tree),

no concrete numbers for splitting seemed to be given. Other parameters tested in the Caruana and Niculescu-Mizil study were not varied.

When implementing the random forest classifiers, the number of trees per forest was set to 1024 and the size of the feature set considered by each set was slightly more narrow than the Caruana and Niculescu-Mizil study (with the exact values being 2, 4, 6, 8, 12, 16, 20).

When implementing the neural network, an MLP classifier with a stochastic gradient descent solver was used. The momentum parameter was varied by the same values as the Caruana and Niculescu-Mizil study - 0, 0.2, 0.5, and 0.9. However, the hidden units varied by factors of ten and ranged from 10^0 to 10^2 (a narrower range than Caruana and Niculescu-Mizil study).

4. Experiment

In an attempt to match the Caruana and Niculescu-Mizil study, 6250 data points were extracted from the training set so that the 80% training dataset comprised of 5000 data points like that of the Caruana and Niculescu-Mizil study. Before extracting these data points, however, the data points were shuffled, so that drawing the first 6250 data points did not bias the results. See Table 1 for further details of the problem.

After performing a grid search with a 5-fold cross-validation for optimal parameters (judged mostly based upon the validation training accuracy results), these parameters were then used for finding the test accuracies for the three trials conducted.

Tables 2, 3, and 4 contain the results of this study and corresponds to the ADULT, COV_TYPE, and LETTER datasets respectively. After the column labeling the classifier used, the first three columns of these tables refer to the training accuracies associated with the parameter eventually chosen for the different data splits. The following three columns refer to the validation accuracies associated with the parameter eventually chosen for the different data splits along with the parameter chosen. The following three columns refer to the testing accuracies of the three trials conducted for the different data splits. Finally, the last column averages the testing accuracies of the three trials.

Table 2: Accuracy Results for the ADULT Dataset

ADULT	Training 80/20 Split	Training 50/50 Split	Training 20/80 Split	Validation 80/20 Split; (Parameter Chosen)	Validation 50/50 Split; (Parameter Chosen)	Validation 20/80 Split; (Parameter Chosen)	Testing 80/20 Split (3 Trials)	Testing 50/50 Split (3 Trials)	Testing 20/80 Split (3 Trials)	Testing Averages: 80/20; 50/50; 20/80
Linear SVC	.7788	.7749	.7242	.778; (10^{-4})	.77344; (10^{-2})	.7256; (10^{-2})	.7832; .7904; .7696	.77312; .77664; .77376	.7886; .7742; .7718	.7811; .7745; .7782
KNN	.7942	0.7921	.794	.7908; (15)	.78496; (13)	.7912; (13)	.7896; .7896; .7896	.78368; .78368; .78368	.7742; .7742; .7742	.7896; .78368; .7742

Decision Tree	.8483	.8534	.8706	.839; (5)	.8352; (5)	.8392; (5)	.8592; .8608; .8584	.84576; .84608; .8464	.837; .837; .8364	.8595; .84608; .8368
Random Forest	.9818	.9243	.929	.847; (4)	.84832; (16)	.8352; (16)	.8648; .864; .8656	.86016; .8608; .86048	.849; .8484; .849	.8648; .86048; .8488
MLP	.77	.7702	.7788	.7692; (units = 100, momentum = 0)	.7693; (units = 100, momentum = 0)	.7792; (units = 100, momentum = 0)	.7696; .7696; .7664	.76896; .76864; .768	.7652; .7696; .7696	.7685; .7685; .7681

Table 3: Accuracy Results for the COV_TYPE Dataset

ADULT	Training 80/20 Split	Training 50/50 Split	Training 20/80 Split	Validation 80/20 Split; (Parameter Chosen)	Validation 50/50 Split; (Parameter Chosen)	Validation 20/80 Split; (Parameter Chosen)	Testing 80/20 Split (3 Trials)	Testing 50/50 Split (3 Trials)	Testing 20/80 Split (3 Trials)	Testing Averages: 80/20; 50/50; 20/80
Linear SVC	.6218	.6057	.5912	.6176; (10^{-5})	.59712; (10^{-5})	.5824; (10^{-5})	.6288; .6008; .624	.60288; .62624; .58848	.6022; .6004; .6004	.6179; .60464; .601
KNN	1	1	1	.7764; (1)	.7571; (1)	.7008; (1)	.784; .784; .784	.76064; .76064; .76064	.7098; .7098; .7098;	.784; .76064; .7098
Decision Tree	.8692	.87456	.8004	.7666; (10)	.7532; (10)	.7504; (5)	.7776; .7784; .7752	.75552; .7596; .75808	.7474; .7474; .7474	.7789; .7653; .7474
Random Forest	1	1	1	.8132; (2)	.80672; (2)	.78; (2)	.82; .8208; .816	.80896; .80928; .80928	.78; .781; .7784	.8189; .80917; .7798
MLP	.5419	.54712	.5034	.5406; (units = 100, momentum = 0)	.5376; (units = 100, momentum = 0)	.5216; (units = 100, momentum = 0)	.556; .5564; .56	.51904; .5168; .49216	.5626; .5136; .4784	.5575; .5093; .5182

Table 4: Accuracy Results for the LETTER Dataset

ADULT	Training 80/20 Split	Training 50/50 Split	Training 20/80 Split	Validation 80/20 Split; (Parameter Chosen)	Validation 50/50 Split; (Parameter Chosen)	Validation 20/80 Split; (Parameter Chosen)	Testing 80/20 Split (3 Trials)	Testing 50/50 Split (3 Trials)	Testing 20/80 Split (3 Trials)	Testing Averages: 80/20; 50/50; 20/80
Linear SVC	.7165	.7171	.7304	.7114; (10^{-1})	.70816; (10^{-1})	.716; (10^{-1})	.7136; .7128; .7136	.72192; .72096; .72096	.7136; .7138; .7148	.7133; .72128; .714
KNN	1	1	1	.9436; (1)	.92768; (1)	.8864; (1)	.9544; .9544; .9544	.94176; .94176; .94176	.9012; .9012; .9012	.9544; .94176; .9012
Decision Tree	.919	1	1	.848; (10)	.8496; (25)	.8112 (20)	.8336; .8376; .8312	.8608; .86688; .8624	.8288; .8352; .8308	.8341; .86336; .8316
Random Forest	1	1	1	.9388; (2)	.91872; (2)	.892 (2)	.9464; .948; .9488	.9296; .9298; .93024	.8782; .8772; .876	.9477; .9299; .8771
MLP	.8033	.8127	.75368	.7902; (units =	.78848; (units =	.7368; (units =	.872; .8656;	.83136; .84864;	.7156; .7312;	.8715; .7942;

				100, momentum = .9)	100, momentum = .9)	100, momentum = .9)	.8768	.72064	.7282	.725

Evidenced by Table 2, 3, and 4, decreasing the amount of training data, generally decreased the accuracy of the classifiers. With the minimal differences between the data splits, exceptions to this may have just arisen by chance (notably from randomizing the order of the data points). Additionally, when comparing to the results of the Caruana and Niculescu-Mizil study, these tables provide similar results (usually within 5% but sometimes within a tad more).

Even when considering that the Caruana and Niculescu-Mizil study utilized more than just accuracy as a metric for the results (of which the previously mentioned result differences could be attributed to), there are a couple notable exception to this - the neural network classifier and the decision tree. The MLP neural network consistently provided lower accuracies than that of the ANN of the Caruana and Niculescu-Mizil study. Although the gradient descent backprop and the momentum hyperparameter variations of the MLP matched that of the ANN, the hidden layer hyperparameter variations did not. The lower test accuracy of the MLP may be attributed to this difference. Additionally, when running the MLP, the maximum number of iterations was reached and the optimization could not converge, which also may have caused this difference in results. Regarding the decision tree classifier, due to the Caruana and Niculescu-Mizil study lacking concrete hyper parameters to follow, differences in results may just be attributed to differences in splitting criterion.

5. Conclusion

For the ADULT dataset, the decision tree and the random forest algorithms provided the least amount of testing error (greatest accuracy). For the COV_TYPE dataset, the KNN and random forest classifiers provided the greatest accuracy. For the LETTER dataset, the decision tree and random forest algorithms provided the greatest accuracy. These results largely matched that of the Caruana and Niculescu-Mizil study when not including classifiers not used in this project and keeping in mind the differences between the neural network and decision tree classifiers of the Caruana and Niculescu-Mizil study and this project. Just as the Caruana and Niculescu-Mizil study concluded, decision trees and random forest classifiers seemed to work best for these datasets (Caruana & Niculescu-Mizil, 2006).

6. References

Caruana R. and Niculescu-Mizil A. (2006). An Empirical Comparison of Supervised Learning Algorithms. Pittsburgh, PA: Proceedings of the 23rd International Conference on Machine Learning.

Dua, D. and Karra Taniskidou, E. (2017). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.