Analysis of Classification Algorithms on a Linearly Separable Dataset

1. INTRODUCTION

Machine Learning classification algorithms are being increasingly employed for tasks such as bank-failure prediction [1], data mining [2] and even disease detection [3]. However, with the development of a large variety of supervised learning (classification) models in the past few decades, deciding on which one to use for a given dataset is an arduous task, and it is crucial to understand the inner mechanisms of these algorithms to determine the ideal one. Previous researches have shown a comparative analysis of classification algorithms on various datasets. For instance, Dogan and Tanrikulu address the robustness of classifiers by evaluating the accuracy and CPU time of the algorithms on various datasets [4]. Another study by Rani and Jyothi evaluate the performance of supervised techniques on diabetes, nutrition and mushroom dataset [5]. While their findings provide an analysis of these algorithms on the datasets, there is little explanation as to why some algorithms performed better than the others. This paper focuses on how linear separability of the dataset affects the prediction accuracy of the popular linear and nonlinear classification algorithms while justifying the results at the same time.

The rest of this paper is structured as follows. In Section 2, an overview of the various classification algorithms along with their benefits and drawbacks is presented. Section 3 introduces the methodology of the experiment giving information on the dataset and the implementation process. All the results are provided in Section 4, and finally, conclusions are drawn in Section 5.

2. OVERVIEW OF THE METHODS

2.1 Logistic Regression

Logistic Regression originated in the 19th Century to understand the growth of populations and the progression of chemical reactions. It is the most common linear classification technique and uses the maximum likelihood estimation to fit the data. It employs the Sigmoid function to assign real valued independent variables a probability between 0 and 1. However, certain assumptions make logistic regression effective only for certain types of datasets. It assumes that the dependent variable is binary, and the covariates are independent of each other. Moreover, the covariates also need to be linearly related to the log odds [1] according to the following formula:

$$\sum_{j=1}^{m} \beta_j x_j + \beta_0 = ln \frac{P(y=1)}{P(y=0)} \quad (1)$$

2.2 K Nearest Neighbor (KNN)

K-nearest neighbor (KNN) algorithm is a classification technique mostly used for pattern recognition. It is a lazy learning algorithm as it approximates locally rather than globally [6]. This local approximation works in the sense that a point is assigned the classification value based on the majority of the label values of K nearest points. The term 'nearest' can refer to different kinds of distances like Manhattan distance (L1 norm) or Euclidean distance (L2 norm). The major advantage of this algorithm is that it is fast and efficient because of the local approximations it performs. However, a drawback of KNN is that the value of K must be pre-determined which may not always be intuitive. Besides, they work well only in the case of spherical data and perform very poorly if the data is in manifolds or is linear.

2.3 Support Vector Machines (SVM)

Support Vector Machine, also popularly known as SVM, is a classification algorithm that can have exceptional performance on linear as well as non-linear datasets depending on the kernel it uses. They integrate the theory-driven statistical methods with data-driven machine learning methods [7]. For linear datasets, the most commonly used choice is linear kernel which is just an affine classifier in the number of dimensions of dataset. In the case of non-linear datasets, Gaussian (rbf) kernel is the optimal choice since it acts as a linear classifier in an infinite-dimensional space [8]. Because of their adaptability to any kind of datasets, they are used in a variety of domains involving text categorization, image classification, hand-written character recognition, and biological sciences.

2.4 Random Forest

Random Forests are a complex and powerful class of classifiers that are highly non-linear. They are essentially a group of decision trees that label the data based on the majority vote of the component trees. Moreover, they employ tree bagging and boosting to reduce the variance and the bias respectively [9]. The major advantage of Random Forest is the notion of variable importance they employ. Because of variable importance, they can explain which covariates account for the majority of explanation of the decision. However, their drawback is that they are very slow and often uninterpretable. Moreover, they depend on parameters like the number of decision trees in the forest and the splitting criteria of the tree which need to be determined manually and can lead to overfitting if not tuned carefully.

2.5 Neural Network

Neural Networks are inspired by the human brain and are extensively used for learning purposes. Each neuron in the network is represented by an activation function that takes multiple inputs from previous layers (neurons) and outputs a single value. These inputs are weighted, and, by backpropagation, these weights are readjusted so that the error is minimized during learning [10]. Neural network is also at the heart of the field of deep learning where it is used for various tasks such as computer vision, natural language processing, speech recognition and many more. They are a powerful set of algorithms having strong adaptability to all kinds of datasets and massive parallel computing ability [11]. However, they are often complex, hard to interpret and require large amounts of data to learn.

3. METHODOLOGY

3.1 Dataset

The dataset [12][13] used in the experiment contains 13910 measurements from 16 sensors exposed to 6 gases, namely Ammonia, Acetaldehyde, Acetone, Ethylene, Ethanol, and Toluene, at varying concentrations ranging from 5 to 1000 ppmv [14]. The data was gathered between January 2008 and February 2011 at the ChemoSignals Laboratory in the BioCircuits Institute, University of California San Diego, and the entire dataset has been divided into 10 batch files depending on the months of data collection. Each chemical sensor gives rise to 8 features which represent the steady-state feature and the increasing/decaying transient portion of the sensor response. Since each measurement comes from 16 different sensors, every data point in the dataset

has 128 features from the sensors, the gas (represented by an integer) on which the experiment is being performed, and its concentration.

3.2 Implementation

Each batch file of the dataset is initially apportioned into a training set and a test set using an 80-20 split. Any missing values in the dataset are replaced by the mean of that feature and every feature is normalized according to the following formula so that there is no bias involved:

$$Value_{new} = \frac{Value_{old} - mean}{Std.Dev} \quad (2)$$

Once the data has been preprocessed, various machine learning techniques as described in Section 2 are applied. As mentioned before, logistic regression assumes that the dependent variable is binary; however, this dataset contains 6 possible values of the dependent variable. Therefore, the logistic regression model needs to be modified to a multinomial logistic regression model that employs Softmax function rather than the Sigmoid function [15]. Two variants of KNN are also tested using the number of neighbors (N) as 5 and 9. This is done to analyze the effect of parameter N on the learning of data. Similarly, two forms of SVM classifiers, namely Linear and Kernel, are used to understand the consequence of linearity of data on the performance of the algorithms. Also, a couple of instances of the Random Forest classification models are implemented by varying the number of decision trees from 10 to 50 in the model. Lastly, a three-layered neural network is implemented where the first two layers use the Rectified Linear Unit (ReLU) activation function and the last layer employs the Softmax activation function [16].

All these algorithms are run independently of each other and the results obtained are compared with the original labels using a confusion matrix [17]. The confusion matrix is the most common

technique for evaluating the performance of a supervised machine learning algorithm and Table 1 shows the organization of a confusion matrix.

Table 1. Confusion Matrix for a 2-class classifier

	Actual Positive	Actual Negative
Predicted Positive	True Positive	False Positive
Predicted Negative	False Negative	True Negative

Once the confusion matrix is generated for every algorithm, the misclassification error and the number of misclassifications defined by the following formulas are used to quantify the efficiency of the algorithm:

$$Misclassification\ error = \frac{False\ positives + False\ Negatives}{Total\ points} \quad (3)$$

$$Misclassifications = False\ positives + False\ Negatives \quad (4)$$

There are two components to each of the accuracy metric: false positives (Type 1 Error), where the classifier predicts positive when it is negative, and false negatives (Type 2 Error), where the classifier predicts negative even though it belongs to the class (is positive). Depending on the experiment, one aims at reducing either the Type 1 Error or the Type 2 Error. However, in our experiment, we treat both the errors equally and try to reduce their sum as encapsulated by the misclassifications metric.

4. RESULTS

Table 2 presents the misclassification error of each algorithm on the 10 batch files of the dataset. To indicate the performance of each algorithm on the individual batch files, the size of the test data for every batch file along with the total number of false positives and false negatives (misclassifications) are also shown.

Table 2. Misclassifications and Misclassification Error for the Classification Algorithms

Batch	Test Size	Logistic Regression	KNN N=5	KNN N=9	Linear SVM	Kernel SVM	Random Forest (n=10)	Random Forest (n=50)	Neural Network
1	89	4	3	7	3	8	4	2	2
2	249	2	1	2	1	1	0	0	0
3	318	4	3	5	1	5	1	2	2
4	33	0	0	2	0	0	0	0	0
5	40	1	1	1	1	1	2	1	0
6	460	4	5	6	3	5	9	5	4
7	723	2	3	3	2	3	2	2	2
8	59	3	1	3	2	3	1	1	2
9	94	0	0	0	0	0	0	0	0
10	720	4	15	21	4	34	14	11	5
	2785	24	32	50	17	60	33	24	17
	Misclassification Error	0.86	1.15	1.80	0.61	2.19	1.18	0.86	0.61

The results show that Linear SVM and Neural Networks performed best on the dataset with a misclassification error of just 0.61% while Kernel SVM had the worst performance among the algorithms and had a misclassification error of 2.15%. While these error percentages seem low, the actual performance of an algorithm depends on the dataset on which it is being used. A misclassification error of 25% on some hard to classify dataset can make the technique a good

classifier on that dataset; while a 2% classification error on an easily separable dataset may prove the algorithm to be a weak classifier on that dataset. Therefore, rather than evaluating using misclassification error, it is better to use actual misclassifications in the given experiment.

Based on the misclassifications metric, it can be seen that Linear SVM and Neural Network had the best performance with only 17 incorrect predictions out of a total of 2785 observations. This suggests that the data is indeed linearly separable and algorithms like Kernel SVM and KNN should have worse performance. The number of wrong predictions increases to 60 in the case of Kernel SVM indicating that the algorithm is overfitting the test data. Similarly, in KNN, the misclassifications increase from 32 to 50 on increasing the value of N from 5 to 9. On increasing the number of trees from 10 to 50 in Random Forest, the misclassification decreased from 33 to 24. However, the algorithm with 50 trees was much slower than the one with 10 trees which explains the tradeoff between accuracy and speed in the case of the Random Forest algorithm. Finally, Logistic Regression performs well with only 24 misclassifications suggesting a high possibility of the covariates being independent.

5. CONCLUSION

Overall, this paper proposes an empirical study on different classification algorithms such as Logistic Regression, KNN, SVM, Random Forest, and Neural Networks and evaluates their performance on the Gas Sensor Array Drift Dataset. While previous research [12] has focused on the clustering (unsupervised) algorithms on the same dataset, the main aim of this paper is to analyze the effectiveness of popular classification (supervised) algorithms depending on the

characteristics of the dataset like linearity and the parameter selection for the algorithms. The performance is evaluated using the Confusion Matrix with misclassifications as the metric.

Since the dataset is linearly separable, techniques such as Linear SVM, Neural Network and Logistic Regression perform better than the non-linear algorithms. While Random Forest with 50 trees has a good performance, it is slow because of the complexity added by a large number of trees. Nonlinear algorithms such as Kernel SVM and KNN with N=9 perform the worst with 60 and 50 misclassifications respectively. These findings suggest that nonlinear techniques tend to overfit when applied on a linearly separable test data set.

However, the limitation of this research is that it considers only the linearity of data and not other characteristics like the number of features and the number of observations. Further research can be conducted on analyzing these attributes of the dataset as well. Moreover, other supervised algorithms like Naïve Bayes, Adaboost and Perceptron can be tested as well to provide further generalization on the performance of classification algorithms on any dataset.

REFERENCES

[1] Hong Hanh Le, Jean-Laurent Viviani. Predicting bank failure: An improvement by implementing a machine-learning approach to classical financial ratios, Research in International Business and Finance, Volume 44, 2018, Pages 16-25, ISSN 0275-5319

[2] AL-Nabi, Delveen Luqman Abd and Shereen Shukri Ahmed. “Survey on Classification Algorithms for Data Mining:(Comparison and Evaluation).” (2013).

[3] Sharmila, Leoni & Dharuman, C & Venkatesan, Perumal. (2017). Disease Classification Using Machine Learning Algorithms-A Comparative Study. International Journal of Pure and Applied Mathematics. 114. 1-10.

[4] Dogan, N. & Tanrikulu, Z. Inf Technol Manag (2013) 14: 105. https://doi.org/10.1007/s10799-012-0135-8

[5] A. S. Rani and S. Jyothi, "Performance analysis of classification algorithms under different datasets," *2016 3rd International Conference on Computing for Sustainable Global Development*, New Delhi, 2016, pp. 1584-1589.

[6] Zhang, Zhongheng. "Introduction to machine learning: k-nearest neighbors." Annals of translational medicine vol. 4,11 (2016): 218. doi:10.21037/atm.2016.03.37

[7] Min, J.H., Lee, Y.C., 2005. Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters. Expert Syst. Appl. 28, 603–614.

[8] Corinna Cortes and Vladimir Vapnik. 1995. Support-Vector Networks. *Mach. Learn.* 20, 3 (September 1995), 273-297.

[9] Leo Breiman. 2001. Random Forests. *Mach. Learn.* 45, 1 (October 2001), 5-32.

[10] Srivastava, Nitish et al. "Dropout: a simple way to prevent neural networks from overfitting." J. Mach. Learn. Res. 15 (2014): 1929-1958.

[11] Ticknor, J., 2013. A Bayesian regularized artificial neural network for stock market forecasting. Expert Syst. Appl. 40 (14), 5501–5506

[12] A Vergara, S Vembu, T Ayhan, M Ryan, M Homer, R Huerta. "Chemical gas sensor drift compensation using classifier ensembles." Sensors and Actuators B: Chemical 166 (2012): 320-329.

[13] I Rodriguez-Lujan, J Fonollosa, A Vergara, M Homer, R Huerta. "On the calibration of sensor arrays for pattern recognition using the minimal number of experiments." Chemometrics and Intelligent Laboratory Systems 130 (2014): 123-134.

[14] Sathishkumar, E. & Kuttiyannan, Dr. Thangavel & Arul, D & Daniel, Arul. (2013). Effective Clustering Algorithm for Gas Sensor Array Drift Dataset. 3.

[15] B. Krishnapuram, L. Carin, M. A. T. Figueiredo and A. J. Hartemink, "Sparse multinomial logistic regression: fast algorithms and generalization bounds," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 6, pp. 957-968, June 2005. doi: 10.1109/TPAMI.2005.127

[16] Agarap, Abien Fred. "Deep Learning using Rectified Linear Units (ReLU)." ArXiv abs/1803.08375 (2018): n. pag.

[17] Kohavi, R. and Provost, F. (1998) Glossary of terms. Machine Learning—Special Issue on Applications of Machine Learning and the Knowledge Discovery Process. Machine Learning, 30, 271-274. https://doi.org/10.1023/A:1017181826899