Categorical Datasets with 7 different Machine Learning Algorithms

Sohyun Park
CCID: sohyun2
Department of Computing Science
University of Alberta, Edmonton, Alberta
sohyun2@ualberta.ca

1 Introduction

Machine learning is currently one of the most popular technologies. It follows the process of building an algorithm for a model, training the model through a dataset, and verifying the model's performance through a test dataset. In this process called fine-tuning, the model has optimised parameters. In ordinary machine learning, we apply multiple models to one dataset and choose a model that shows the best performance. In contrast, in this project, I aim to track why these models are adequate for given datasets and the factors that optimise models to achieve the results.

For this object, I chose three datasets representing three areas that deep neural networks usually apply - Vision, Natural Language Processing(NLP and Audio Speech - instead of other machine learning methods. Unlike ordinary cases, I used seven machine learning techniques other than deep natural networks to the datasets: Linear Regression, Logistic Regression, Decision Tree, Support Vector Machine(SVM), Naive Bayes, k Nearest Neighbours(kNN) and Random Forest. Based on the distinct characteristics of each dataset, I found out more clearly what advantages and disadvantages exist when the seven algorithms are applied to each dataset. Additionally, the project has shown through experiments that we can achieve high performance without necessarily using deep natural networks when performing classification tasks with the three dataset types.

2 Background

Comparing various machine learning models for specific datasets forms the foundation of Uddin et al. [1], which applied six supervised machine learning algorithms on single disease prediction. It selected 48 articles employing different machine learning algorithms and found out the Support Vector Machine algorithm is used most frequently among the six models.

These days, models built using artificial neural networks are mainly adopted due to their strong performance. Regarding the vision classification task using the MNIST dataset, the model applying convolutional neural networks accomplished the state-of-the-art with 99.91% accuracy [2]. In natural language processing, the transformer method model yields 97.1% accuracy, achieving the state-of-the-art with the IMDB dataset [3]. Even though there is no benchmark paper for the FSDD dataset, Jain [4] indicates that the audio speech classification task has 72.8% accuracy with convolutional neural networks.

Being motivated by the previous works, this project applied seven machine learning algorithms including a Decision Tree and used three datasets for the experiments. One step forward, the project addresses why the model that performs best on a particular dataset could produce such results. Additionally, I intended to show that simple machine learning models can produce performance equivalent or similar to artificial neural networks by comparing the effects of machine learning models adopted through this project with state-of-art models.

3 Methodology

This project aims to determine which machine learning algorithm is the most suitable for analysing a specific dataset and why the algorithm works best for the dataset. I chose seven machine learning algorithms and three datasets to conduct the experiments, and calculated both accuracy and F1-score to evaluate model performance.

3.1 Models

Following 7 machine learning algorithms will be used for this project: Linear Regression, Logistic Regression, Decision Tree, Support Vector Machine(SVM), Navie Bayes, kNN, Random Forest.

3.1.1 Linear Regression

Linear regression attempts to model the relationship between two variables by fitting a linear equation to observed data. One variable is considered an explanatory variable, and the other is regarded as a dependent variable. Before attempting to provide a linear model to data, it should be determined whether or not there is a relationship between the variables of interest. If there appears to be no association between the proposed explanatory and dependent variables, then fitting a linear regression model to the data probably will not provide a good model. Linear regression has an equation of the form $Y = a + bX$, where X is the explanatory variable and Y is the dependent variable.

3.1.2 Logistic Regression

Logistic Regression is a classification algorithm that learns a function that approximates $P(Y|X)$, which means the probability of Y(target) given X(data). It makes the central assumption that $P(Y|X)$ can be approximated as a sigmoid function applied to a linear combination of input features. Mathematically, for a single training data point (x, y), Logistic Regression assumes:

$$P(Y = 1|X = x) = \sigma(z)$$

where $z = \theta_0 + \Sigma_{i=1}^{m}\theta_i x_i$. Using these equations for the probability of $Y|X$, we can create an algorithm that select values of θ that maximize that probability for all data.

3.1.3 Decision Tree

A decision tree is a non-linear model with separate lines dividing input space recursively into disjoint regions.

$$\chi = \cup_{i=0}^{n} R_i,\ s.t.\ R_i \cap R_j = \emptyset\ for\ i \neq j$$

The final regions are called terminal or leaf nodes. And the intermediate regions are called parent node R_p or Children nodes R_1 and R_2 split on feature X_j, which can be represented as mathematical form:

$$R_1 = X|X_j < t, X \in R_p$$
$$R_2 = X|X_j \geq t, X \in R_p$$

To predict the outcome in each leaf node, the average outcome of the training data in this node is used.

3.1.4 Support Vector Machine(SVM)

Support vector machines (SVMs) are linear classifiers based on the margin maximization principle. The SVM accomplishes the classification task by constructing the hyperplane in a higher-dimensional space that optimally separates the data into two categories. The SVM algorithm attempts to find the maximum margin between the two data categories and then determines the hyperplane in the middle of the maximum margin. Thus, the points nearest the decision boundary are located at the same distance from the optimal hyperplane.

3.1.5 Naive Bayes

Naive Bayes is an algorithm that utilizes the Bayes rule, $P(z|w) = \frac{P(z)P(w|z)}{P(w)}$, with a strong assumption that the attributes are conditionally independent, given the class. While this independence assumption is often violated in practice, Naive Bayes usually delivers competitive classification accuracy. For categorical attributes, the required probabilities $P(y)$ and $P(x|y)$ are normally derived from frequency counts stored in arrays whose values are calculated by a single pass through the training data at training time. These arrays can be updated as new data are acquired, supporting incremental learning. Probability estimates are usually derived from the frequency counts using smoothing functions such as the Laplace estimate or an m-estimate.

3.1.6 k Nearest Neighbours(kNN)

K Nearest neighbour is a machine learning method that aims at labelling previously unseen query objects while distinguishing two or more destination classes. The query object inherits the label from the closest sample object in the training set. The decision rule combines the labels from these k decision objects, either by simple majority voting or by any distance-based or frequency-based weighting scheme, to decide the predicted label for the query object.

3.1.7 Random Forest

A random forest is an ensemble of random Decision Tree classifiers that makes predictions by combining the predictions of the individual trees.

3.2 Datasets

In this project, three types of datasets are used. Modified National Institute of Standards and Technology database(MNIST) [5] is adopted for the vision classification task. It has a training set of 60,000 examples and a test set of 10,000 examples. To conduct NLP classification task, Large Movie Review Dataset(IMDB Dataset) [6] is used. This is a dataset for binary sentiment classification that provides 50,000 movie reviews. And I employ Free Spoken Digit Dataset(FSDD) [7] for Audio and Speech classification task. There are 3,000 recordings of 6 speakers in the dataset, 50 of each digit per speaker.

3.3 Metrics

I evaluate the model performance for each dataset using accuracy and F1-score. Accuracy is a good measure of success when the True Positives and True negatives are more important. We usually use accuracy for a dataset whose class distribution is similar. Meanwhile, F1-score is better when the False Negatives and False Positives should be focused on. If the data distribution is imbalanced, F1-score is preferred to use. This paper provides accuracy and F1-score as results of the experiments to consider both cases of data distribution(balanced or imbalanced).

4 Experiments

I trained the machine learning models on one machine without any GPU. The maximum time taken for the training step was approximately 527 seconds, while overall running time took about 937.068 seconds. The codes were written on Jupyter Notebook, which will be provided with this paper.

4.1 Vision

The provider of the MNIST dataset has already divided it into a training dataset and a test dataset. There are 60,000 examples in the training set and 10,000 examples in a test set. For the classification task, the model's input data is image converted into numerical values and normalised with value 256. The outputs are numbers in the range from 0 to 9.

There was no change from the basic setup under the scikit-learn framework regarding the settings of the adopted machine learning models. In contrast, n_neighbors value for k Nearest Neighbours model and max_depth value for Random Forest model set as ten because there are ten different values in the labels.

4.2 Natural Language Processing

I used the TfidfVectorizer function provided by the scikit-learn library to vectorise the natural languages in the IMDB dataset. The vectorising review data goes through the models as inputs with a pipeline, and the outputs are two kinds of labels: positive(1) or negative(0).

The dataset was divided with rates 8 to 2, where eight is for training data, and two is for test data. Regarding the settings of the models, there was no change from the basic setup under the scikit-learn framework. In contrast, n_neighbors value for k Nearest Neighbours model and max_depth value for Random Forest model set as two because there are two different labels(positive or negative) in the dataset.

4.3 Audio Speech

I used the librosa(==0.8.1) library for pre-processing the FSDD dataset. I extracted mfcc from the audio data and converted it to numerical values with the librosa library. And then, the converted mfcc values got reshaped, making it from 3-dimensional data to 2-dimensional data. I employed mfcc values going through the process as inputs of the models for the audio classification task, and the outputs are numbers in the range from 0 to 9.

The dataset was divided with rates 8 to 2, where eight is for training data, and two is for test data. Regarding the settings of the models, there was no change from the basic setup under the scikit-learn framework. In contrast, n_neighbors value for k Nearest Neighbours model and max_depth value for Random Forest model set as ten because there are ten different values in the labels.

5 Results

5.1 Vision

Considering the table below, Support Vector Machine(SVM) and k Nearest Neighbours(kNN) perform better than the other models with 97% accuracy. It is slightly less than 99.1, which is the accuracy of the CNNs based model that achieved state-of-the-art for the same task. Assuming that it is crucial to spend less time training the model, we can use SVM or kNN instead of the CNN model.

We can also see that the gap between the minimum value of the F1-score and the maximum value of the F1-score is relatively large for other models compared to SVM and kNN. Therefore, the performance of the two models is extraordinary. The two models have a common point: they are all distance-based models. It tells us that distance-based models work well with vision data, which is reasonable considering that CNN models focus on training neurons in certain regions.

MNIST dataset				
Models	Accuracy	F1-score(min)	F1-score(max)	time(s)
Linear Regression	0.26	0.00	0.51	3.53
Logistic Regression	0.93	0.88	0.97	18.84
Decision Tree	0.88	0.82	0.96	25.61
Support Vector Machine	**0.97**	**0.95**	**0.99**	526.36
Naive Bayes	0.56	0.09	0.90	**1.49**
k Nearest Neighbours	**0.97**	**0.95**	0.98	32.76
Random Forest	0.95	0.91	0.98	32.09

Table 1: Results produced with the 7 models with MNIST dataset

5.2 Natural Language Processing

The table below shows that Logistic Regression and SVM perform better than the other models with 90% accuracy. It is 7.1% lower than the state-of-the-art model constructed using a transformer.

Based on the results of the experiments, it is better to use a transformer-based model than Logistic Regression or SVM if the model performance is an essential factor for a task.

Since Logistic Regression is based on probability and SVM is based on the distance between vectors, it isn't easy to find something in common between the two models. Nevertheless, to find similarities, I focused on the fact that logistic regression used the sigmoid function to make the decision boundary of linear regression smoother. In the case of SVM, it finds a decision boundary located at the same distance among the particular points. Therefore, the given dataset is sensitive to the decision boundary that the model yields.

IMDB dataset				
Models	Accuracy	F1-score(0)	F1-score(1)	time(s)
Linear Regression	0.75	0.74	0.75	142.37
Logistic Regression	**0.90**	**0.90**	**0.90**	14.22
Decision Tree	0.72	0.72	0.73	100.45
Support Vector Machine	**0.90**	**0.90**	**0.90**	8.75
Naive Bayes	0.86	0.87	0.86	7.70
k Nearest Neighbours	0.75	0.77	0.72	**7.43**
Random Forest	0.82	0.81	0.82	12.01

Table 2: Results produced with the 7 models with IMDB dataset

5.3 Audio Speech

The table below indicates that Logistic Regression, SVM, and Random Forest work well with the dataset considering the accuracy, which is 99%. However, due to the accuracies provided by the models mostly exceeding 95%, which is higher than 72.8% given by the CNN based model, we can say that the dataset classified well with any algorithms except decision tree.

As 5.2, Logistic Regression and SVM performed well with the given dataset. The model that provides the lowest accuracy is Decision Tree, while Random Forest offers the best accuracy value. Overall, the dataset is also sensitive to the decision boundary, though it is not that influenced compared to the case of 5.2, considering the accuracy values that the models provide.

FSDD dataset				
Models	Accuracy	F1-score(min)	F1-score(max)	time(s)
Linear Regression	0.97	0.93	1.00	0.062
Logistic Regression	**0.99**	**0.98**	**1.00**	0.249
Decision Tree	0.87	0.82	0.91	0.873
Support Vector Machine	**0.99**	**0.98**	**1.00**	0.885
Naive Bayes	0.95	0.88	0.98	**0.004**
k Nearest Neighbours	0.98	0.96	1.00	0.016
Random Forest	**0.99**	0.97	1.00	1.369

Table 3: Results produced with the 7 models with FSDD dataset

6 Conclusion

By backtracking the process of finding the most efficient model for a specific dataset, this project made it possible to grasp the nature of machine learning models. I adopted seven non-neural network machine learning algorithms for three types of datasets: Vision, Natural Language Processing, and Audio Speech, which Neural Network Models usually accompany.

The results show how much accuracy the adopted models provide, and I compared it with the accuracy of the state-of-the-art model. It tells us that the chosen models perform similar to or even better than the neural network models for Vision and Audio data. Experimenting with basic algorithms whose structures are relatively well known made it easy to grasp which algorithm characteristics can effectively handle datasets. The experiments show how to choose a machine learning model suitable for datasets in more specific and essential aspects, not just relying on vague intuition.

References

[1] Shahadat Uddin, Arif Khan, Md Ekramul Hossain, and Mohammad Ali Moni. Comparing different supervised machine learning algorithms for disease prediction. *BMC Medical Informatics and Decision Making*, 19, 2019.

[2] Sanghyeon An, Minjun Lee, Sanglee Park, Heerin Yang, and Jungmin So. An ensemble of simple convolutional neural network models for mnist digit recognition, 2020.

[3] Siyu Ding, Junyuan Shang, Shuohuan Wang, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. Ernie-doc: A retrospective long-document modeling transformer, 2021.

[4] Royal Jain. Improving performance and inference on audio classification tasks using capsule networks, 2019.

[5] Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

[6] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/P11-1015`.

[7] Zohar Jackson, César Souza, Jason Flaks, Yuxin Pan, Hereman Nicolas, and Adhish Thite. Jakobovski/free-spoken-digit-dataset: v1.0.8, August 2018. URL `https://doi.org/10.5281/zenodo.1342401`.