Logistic Regression and Convolutional Neural Networks Performance Analysis based on Size of Dataset

[1]Kartik Chopra, [2]C. Srimathi
[1]Student, [2]Associate Professor,
[1]VIT University, Vellore, Tamil Nadu, India

***Abstract*— Machine learning is a method of data analysis that automates analytical model building. Using algorithms that iteratively learn from data, machine learning allows computers to find hidden insights without being explicitly programmed where to look. Deep Learning is a new area of Machine Learning research, which has been introduced with the objective of moving Machine Learning closer to one of its original goals: Artificial Intelligence. Various deep learning architectures such as deep neural networks, convolutional deep neural networks, deep belief networks and recurrent neural networks have been applied to fields like computer vision, automatic speech recognition, natural language processing, audio recognition and bioinformatics where they have been shown to produce state-of-the-art results on various tasks. Deep learning requires many hyperparameters to tune such as the number of layers, the number of predictor variables.**

Index Terms— **Artificial Intelligence, Deep Learning, Machine Learning.**

I. INTRODUCTION

This paper aims to instruct about two deep learning algorithms and to tell the performance of each algorithm on various sizes of data sets. The paper will focus on supervised learning algorithms such as – Logistic Regression and Deep Convolutional Networks. Logistic Regression being a probabilistic linear classifier is used here. The paper deals with a classification algorithm Convolutional Neural Networks used to train a classifier for classifying various images based on labels. Logistic Regression is used to identify predictor variables of a given dataset and then the model is run with Anova to find the dependent variables and to test for fitness. The paper will analyze the results of running the above two algorithms on datasets of different sizes. One small dataset and one large data set will be used. Time taken to run the algorithm, time taken to train, and time taken to predict the results, all will be calculated.

II. LITERATURE SURVEY

Deep learning (also known as deep structured learning, hierarchical learning or deep machine learning) is a class of machine learning algorithms that use a cascade of many layers of nonlinear processing units for feature extraction and transformation. Each successive layer uses the output from the previous layer as input. The algorithms may be supervised or unsupervised and applications include pattern analysis (unsupervised) and classification (supervised). They are based on the (unsupervised) learning of multiple levels of features or representations of the data. Higher level features are derived from lower level features to form a hierarchical representation. They are part of the broader machine learning field of learning representations of data. They learn multiple levels of representations that correspond to different levels of abstraction; the levels form a hierarchy of concepts. There are many models that exist that test the working of Logistic regression and Deep convolution networks.

A. Logistic Regression

Logistic regression is a statistical method for analyzing a dataset in which there are one or more independent variables that determine an outcome. The outcome is measured with a dichotomous variable (in which there are only two possible outcomes). In logistic regression, the dependent variable is binary or dichotomous, i.e. it only contains data coded as 1 (TRUE) or 0 (FALSE).

B. Convolutional Neural Networks

A Convolutional Neural Network (CNN) is comprised of one or more convolutional layers (often with a subsampling step) and then followed by one or more fully connected layers as in a standard multilayer neural network. The architecture of a CNN is designed to take advantage of the 2D structure of an input image (or other 2D input such as a speech signal). This is achieved with local connections and tied weights followed by some form of pooling which results in translation invariant features. Another benefit of CNNs is that they are easier to train and have many fewer parameters than fully connected networks with the same number of hidden units.

III. THE APPROACH

All the work has been done in R programming language. Python was used to download a dataset for CNN algorithm. The paper deals with implementing Logistic regression and CNN on various data sets. The first algorithm, Logistic Regression, is performed in

R using the glm() function. glm() function is used to predict binary outcome from the set of continuous predictor variables. The dataset used is the Titanic Survivor dataset. The aim of the model is to find the independent variables which can be used to predict whether a passenger survived or not. The dataset is a small dataset of around 900 values. Another dataset, the IMDb movie dataset of around 3000 values is used.

The second algorithm, Convolutional Neural Network is also performed in R using the mxnet library. MXNet is an open-source deep learning framework that allows to define, train, and deploy deep neural networks on a wide array of devices, from cloud infrastructure to mobile devices. It is highly scalable, allowing for fast model training, and supports a flexible programming model and multiple languages. CNN is performed on various image sets that have been converted into a csv file and these files are used as training and testing datasets. All the images have been preprocessed and turned to grayscale for having a single channel of color in the csv file. Also the image size has been reduced to 28x28 pixels since the machine the algorithm was run did not have a high end GPU.

The CNN model has various parameters like learning rate, batch size, number of rounds, momentum that have to be tweaked to get accurate results. Each of these parameters plays an important role. Learning rate is defined in the context of optimization, and minimizing the loss function of a neural network.

A. Prerequisites for Logistic Regression

The dataset had to be in csv file. The Titanic dataset, of around 900 values, had around 12 parameters, namely – Cabin, Age Embedded, Fare, Ticket, Patch, SibSp, Sex, Name, PClass, PassengerID and Survived. The IMDb dataset consisted of 14 parameters namely - num_critic_for_reviews, duration, director_facebook_likes, actor_1_facebook_likes, actor_2_facebook_likes, actor_3_facebook_likes, gross, num_voted, cast_total_facebook_likes, num_user_for_reviews, budget, imdb_score, movie_facebook_likes and success.

B. Prerequisites for Convolutional Neural Networks

One dataset, Olivetti Faces, of around 400 images was used. Another dataset of 25000 images was used. The second dataset comprised of 12500 of cats and another 12500 images of dogs. Both datasets were converted to grayscale and had 28x28 pixels for each image. Then a csv file was created using the pixel values of each image. The first csv file had 28*28*400 elements and the second csv file had 28*28*25000 elements.

C. The Model for Logistic Regression

Generalized linear models are fit using the glm() function in R. Here, glm() was used with the family ‘binomial’ and the link ’logit’. The ‘binomial’ family gives a response as a numerical vector with values between 0 and 1, interpreted as the proportion of successful cases. ‘Logit’ is used as a special link function in glm() to perform linear regression.

```
glm(Survived ~.,family=binomial(link='logit'),data=train)      (1)
glm(Success ~.,family=binomial(link='logit'),data=train)       (2)
```

D. The Model for Convolutional Neural Network

The CNN model for training was created by using the FeedForward.create() function in R under the MXNet library. This function was fed with a Neural Network model using the SoftmaxOutput() function. Parameters such as learning rate, num.round, momentum, array batch size were set accordingly. Before the model had to be trained, a neural net was formed comprising of two convolutional layers and two pooling layers. The activation function used was tanh. It is used to map any value to lie between [0,1].

```
model <- mx.model.FeedForward.create(NN_model, X = train_array, y = train_y,
ctx = device,
num.round = 50,
array.batch.size = 100,
learning.rate = 0.05,
momentum = 0.9,
wd = 0.00001,
eval.metric = mx.metric.accuracy,
epoch.end.callback = mx.callback.log.train.metric(100)) (3)
```

```
model<-mx.model.FeedForward.create(NN_model, X=train_array, y=train_y,
ctx=devices,
num.round = 480,
array.batch.size = 40,
learning.rate = 0.01,
```

momentum=0.9,
eval.metric = mx.metric.accuracy,
epoch.end.callback = mx.callback.log.train.metric(100)) (4)

IV. Performance Analysis

Each of the algorithm was run on MacBook having Ram size of 8gb and Intel Core M Processor 1.1GHz. The performance was measured on the basis of time, the size of dataset and the accuracy of the model.

A. Time Analysis

Both datasets, of sizes 900 and 3800 values, of logistic regression took less than 1 second to process the result. However, a major discrepancy was observed when performing CNN on two different datasets. The Olivetti Faces dataset of 400 images took about 13 minutes to train the model. The 'num.round' parameter for this dataset was set to be around 480, which meant that the entire dataset was gone over 480 times. But when the cats and dogs dataset of 25000 images was taken, the training time increased from 13 minutes to 1 hour 10 minutes even with a very small 'num.round' value of just 50.

B. Accuracy Measure

The logistic regression algorithm showed an accuracy of about 86% for the Titanic dataset and an accuracy of about 76%. The CNN algorithm displayed an accuracy of 97% for the Olivetti Faces dataset but showed an accuracy of about 65% for the cats and dogs dataset.

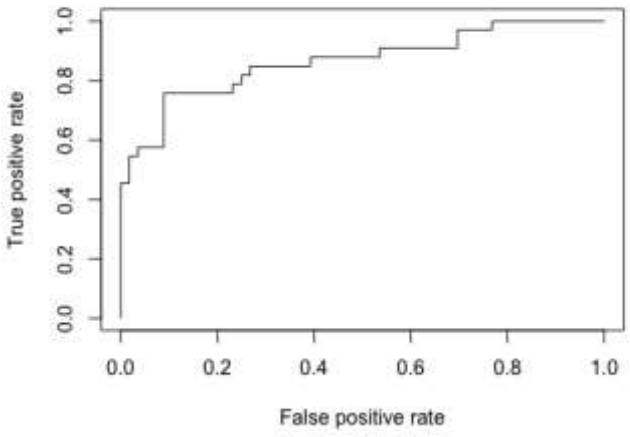

Figure 1 True Positive Rate vs False Positive Rate for Titanic Dataset

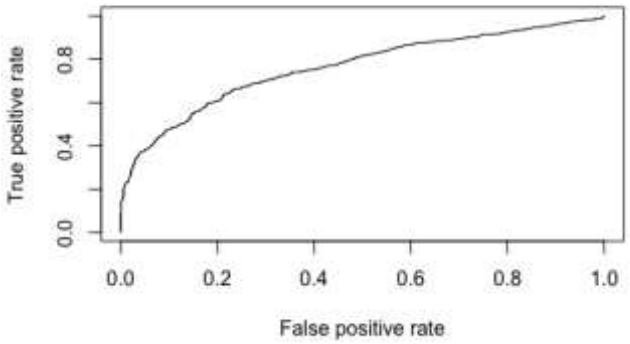

Figure 2 True Positive Rate vs False Positive for IMDb Dataset

V. GAPS AND LIMITATIONS

No model is 100% accurate. All these models showed accuracy between the range of 65% to 97% and the running time of these models is very high for large number of data sets. In logistic regression, identifying the correct independent variables is one of the major tasks. There are limited number of independent variables and wrongly identifying them ruins the model. In CNN, it has a high computational cost, and they are slow to train without a good GPU.

VI. CONCLUSION

In this paper, we performed two deep learning algorithms, logistic regression and convolutional neural network. We analyzed the time performance and accuracy of all the models based on different size of datasets.

REFERENCES

[1] S. S. Farfade, M. Saberian, and L.-J. Li. Multi-view face detection using deep convolutional neural networks. ICMR, 2015..

[2] D. Cires¸an, U. Meier, and J. Schmidhuber. Multi-column deep neural networks for image classification. Arxiv preprint arXiv:1202.2745, 2012.

[3] D.C. Cires¸an, U. Meier, J. Masci, L.M. Gambardella, and J. Schmidhuber. High-performance neural networks for visual object classification. Arxiv preprint arXiv:1102.0183, 2011.

[4] Archer, K. J. 2001. Goodness-of-fit tests for logisitic regression models developed using data collected from a complex sampling design. Ph.D. thesis, Ohio State University.

[5] Cramer, J.S., The Origins of Logistic Regression (December 2002). Tinbergen Institute Working Paper No. 2002-119/4.