An Evolutionary Algorithm based Parameter Estimation using Pima Indians Diabetes Dataset

Dr. Chandan Banerjee
Netaji Subhash Engineering College
Kolkata, India
chandanbanerjee1@gmail.com

Sayak Paul
Dept. of Information Technology
Netaji Subhash Engineering College
Kolkata, India
spsayakpaul@gmail.com

Moinak Ghoshal
Dept. of Information Technology
Netaji Subhash Engineering College
Kolkata, India
moinakghoshal1@gmail.com

***Abstract*—**Predictive modeling using the prowess of Machine Learning is getting stronger and smarter day by day. Often, these predictive models which are generally used to estimate specific values for a given problem are needed to be supplied with proper parameters. The parameters with which they are trained have to be valuably optimal so that the models yield good results. In this paper, a Neural Network is chosen as the predictive model which uses Pima Indians Diabetes dataset. For obtaining the optimal values for the parameters of the Neural Network, Evolutionary Algorithm based approach has been used, which not only resulted in a better execution time but also generated more optimal values as compared to the other existing methods. The values are compared with respect to their accuracy scores.

Keywords-*Neural Network; Batch Size; Pima Indian Diabetes; Evolutionary Algorithm; GridSearchCV*

*****

I. INTRODUCTION

Machine Learning, now a days has manifold applications. Be it in the grounds of Image Processing, be it in the grounds of Weather Prediction, be it any critical Stock Price Prediction, the branch of Machine Learning is yielding the state of the art results [1]. Behind the success of Machine Learning lies the power of Predictive Modeling which is also sometimes called as Statistical Modeling. Further these models are governed by various learning algorithms [2]. Now, almost all of the various learning algorithms have different sets of parameters with which they are trained. These parameters play an extremely important role in the generalization of these models so that they do not bear the problem of overfitting [3]. Therefore, obtaining the optimal values for these parameters is a very crucial task in the context of Predictive Modeling as well as Machine Learning. There exist few methods which can actually be used to find the optimal parameter values such as Randomized Search, Grid Search and Evolutionary Search [4]. In this paper, an Evolutionary Search based Parameter Estimation approach has been presented using the Pima Indian Diabetes dataset in which a Neural Network has been used as the Estimator. Further, a comparative study between the other Parameter Estimation methods has been shown to measure the performance of the proposed method in terms of execution time and accuracy.

This paper is divided into five sections. Section II describes the Related Works. Section III briefs about the Pima Indians Diabetes dataset. In Section IV the Proposed Method has been presented. Section V deals with the Experimental Results. Conclusion and Future Work has been given in Section VI.

II. RELATED WORK

In 1998, a global optimization approach for expensive black-box functions was proposed by Donald, Matthias and William [5]. In 2011, Frank, Holger and Kevin proposed sequential optimization technique for automatic algorithm configuration [6]. In 2011, Yasser, Thomas, Sara, Rich and Christina proposed a Distributed Tuning of machine learningalgorithms using Map Reduce clusters [7]. Jasper, Hugo and Ryan proposed a Bayesian based approach for the optimization of the model parameters in the year of 2012. In that paper, they incorporated Gaussian processes for constructing their Bayesian model [8]. In 2012, James and Yoshua proposed a Randomized search based method for parameter tuning which independently draws uniform density from the configuration space of the model [9].

III. DATASET DESCRIPTION

Pima Indians Diabetes dataset is a standard dataset for Machine Learning research purposes and it has been used by many researchers for different purposes.

The dataset contains 768 numbers of instances and 9 features including the class labels. All the values are numeric in the dataset. Figure 1 shows the feature names accordingly.

```
1. Number of times pregnant
2. Plasma glucose concentration a 2 hours in an oral glucose tolerance test
3. Diastolic blood pressure (mm Hg)
4. Triceps skin fold thickness (mm)
5. 2-Hour serum insulin (mu U/ml)
6. Body mass index (weight in kg/(height in m)^2)
7. Diabetes pedigree function
8. Age (years)
9. Class variable (0 or 1)
```

Figure 1. Feature names of the dataset

Figure 2 represents the class distribution of the dataset.

According to Figure 1, in the Class variable 0 indicates absence of Diabetes and 1 means presence of Diabetes.

Class Value	Number of instances
0	500
1	268

Figure 2. Class distribution of the dataset

The dataset contains missing values. Figure 3 shows the statistics of the missing values with respect to the feature numbers (as shown in Figure 1) [10].

1	5
2	35
3	227
4	374
5	11

Figure 3. Missing Values Statistics of the dataset

IV. PROPOSED WORK

The objective of this paper is to bring forth the potential of Evolutionary Algorithms in the context of parameter estimation for Predictive Models. A Neural Network has been chosen as the estimator for the course our work. An Evolutionary Search based method has been applied on the estimator and Pima Indians Diabetes dataset has been used. Following algorithm shows the steps involved in constructing the Neural Network as the estimator.

Input (Pima Indian Diabetes dataset)
Output (Dataset classified in 0 or 1)

Step 1: Preprocess the dataset.
Step 2: Normalize the dataset.
Step 3: Initialize a three layer Neural Network for the task of classification with its parameters randomly initialized.
Step 4: Train the Neural Network with the preprocessed and normalized version of the dataset.
Step 5: Cross validate the Neural Model for evaluating its classification accuracy.

After the above algorithm is being followed, an Evolutionary Search based method has been applied on the Neural Network model for obtaining its optimal parameter values. Following algorithm shows the steps involved in this process.

Input (Trained Neural Network with random parameter values)
Output (Optimal set of parameter values)

Step 1: Feed the Neural Network to the Evolutionary Search based method called EvolutionarySearchCV.
Step 2: Obtain the optimal parameter values from the method for which the Neural Network model yields the best result in terms of classification accuracy.

A. Preprocessing the dataset

In accordance with the Figure 3, it can be seen that the dataset contains missing values i.e. in some rows for a particular features there are no values present. Machine Learning algorithms often give very bad results if they are supplied with a dataset having missing values [11]. To address this issue we imputed the missing values with a technique called Mean Imputation [12].

B. Normalizing the dataset

The dataset contains numeric values which are not present in a uniform distribution. Due to this, a small change in a value of a particular feature may not impact the other variables having different range of values. To resolve this problem and to make the range of all feature values between 0 and 1 we further normalized the dataset with MinMax normalization technique [13].

C. Initialization of the Neural Network

A three layer Neural Network was initialized having one input layer, one hidden layer and one output layer. The input layer contains 12 neurons, the hidden layer contains 8 neurons and the output layer contains 1 neuron as this is a binary classification problem. All of the neurons are connected with all the other neurons of its preceding and succeeding layers. Figure 4 shows a sample three layer Neural Network architecture.

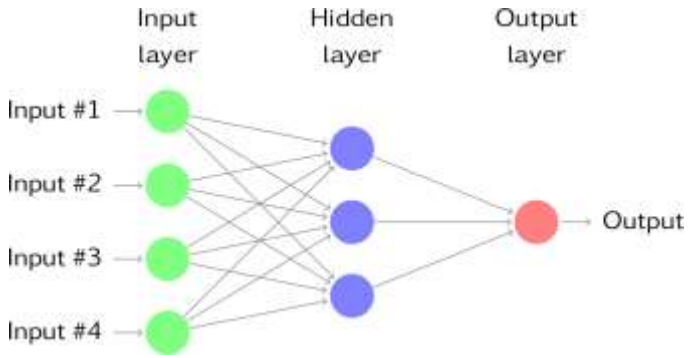


Figure 4. A Sample Neural Network architecture with three layers

The Neural Network is initialized with random set of values in its parameters. Some of the parameters are often called Hyperparameters as their values cannot be learned during the training of the Neural Network [14]. So these hyperparamters are selected for further estimation purpose. The Neural Network is optimized with Stochastic Gradient Descent [15].

D. Training and Cross-validation of the model

After being initialized, the Neural Network is trained with the dataset. As no separate testing set of the dataset is available Cross-validation method is applied to validate the classification accuracy of the Neural Network model [16].

E. EvolutionarySearchCV for parameter estimation

For the experimental purpose, hyperparameters like learning rate, batch size, number of epochs, regularization dropout have been chosen for tuning. These hyperpaamters have a great impact on the efficient training of a Neural Network.

The batch size is the number of patterns shown to the Network before its weights are updated. It is also an optimization in the training of the network, defining how many patterns to read at a time and keep in memory.

The number of epochs is the number of times that the entire training dataset is shown to the network during training.

Learning rate is used to control how much to update the weight at the end of each batch so the Gradients are not too astray.

Dropout is a regularization technique where randomly selected neurons are ignored during training to penalize for the large weights so that the model does not overfit [14].

Evolutionary algorithms (EA) tackle the optimization tasks by trying to incorporate the fundamentals behind natural evolution. EA are stochastic search and the optimization heuristics are derived from the classic evolution theory. The ground level idea is that if only those individualsof apopulation reproduce, which meet a certain selection criteria, and the other individuals of the population die, the population will converge to those individuals that best meet the selection criteria. If any distorted reproduction is added the population will begin to explore the search space and will move to individuals that have an increased selection probability and that exhibit this property to their descendants. These population dynamics obey the basic rule of the Darwinistic evolution theory [17].

For the experimental purpose Genetic Algorithm is used as the Evolutionary Algorithm. It follows the strategy of Selection, Crossover and Mutation simulating the biological processes of Gene-expression [18].

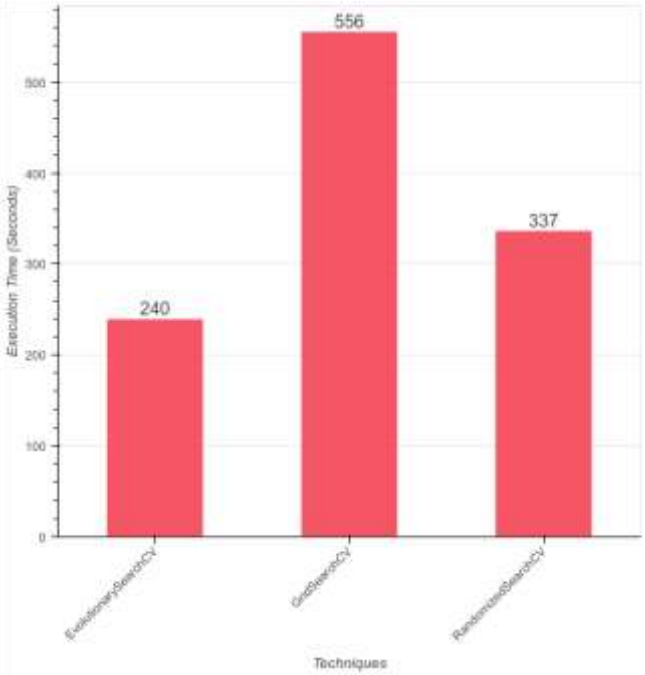
Now this method is used for finding the optimal values of the hyperparameters of the Neural Network mentioned above following the basic Evolution theory rather than trying out computations on an exhaustive search space. The method is called EvolutionarySearchCV [19].

IV. EXPERIMENTAL RESULTS

The Estimator performance evaluation metric that is used is Accuracy. Accuracy is the percentage of the correctly classified instances [1].

In Figure 5, a graph is shown to compare the execution times of EvolutionarySearchCV the other parameter estimation methods like RandomizedSearchCV and GridSearchCV.

Figure 5. Graph showing the comparison between the execution times of different Parameter Estimation methods including EvolutionarySearchCV



In Figure 6, a comparison between the accuracy scores of the above mentioned parameter stimation methods is presented.

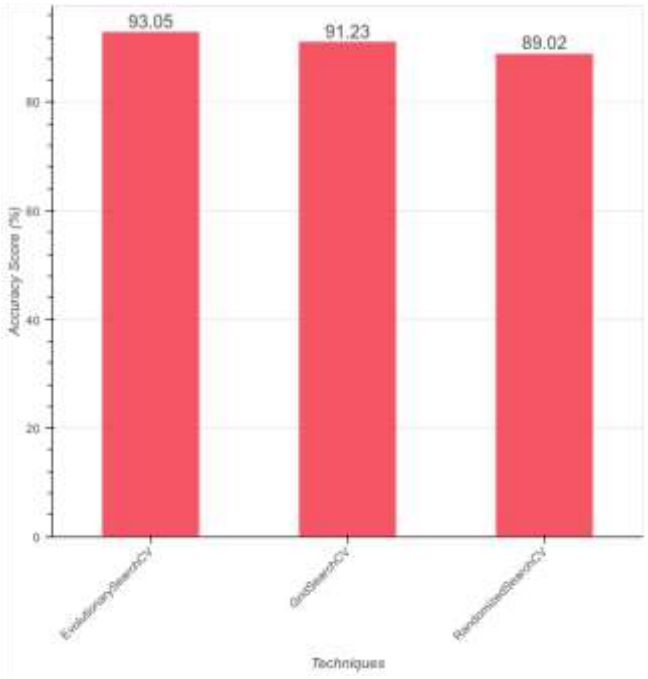


Figure 6. Graph showing the Accuracy Scores of the Parameter Estimation methods including EvoltionarySearchCV

Figure 7 shows the Accuracy scores of the neural Network model with Parameter Estimation and without Parameter Estimation.

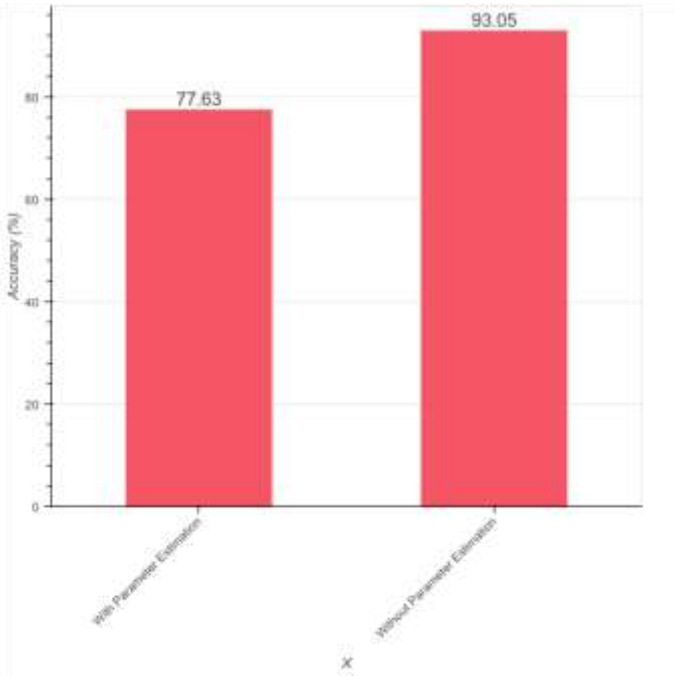


Figure 7. Graph showing the accuracy scores of the Neural Network model with and without Parameter Estimation

Following the optimal values of the above mentioned hyperparamaters obtained from EvolutionarySearchCV method:

Batch Size: 50
Number of epochs: 20
Learning Rate: 0.001
Dropout Regularization: 0.1

So, from the Figures 5, 6 and 7 it can be seen that with EvolutionarySearchCV technique the Parameter Estimation can be done more accurately and efficiently as compared to the other two techniques.

However, for the purpose of the experiment this technique has only been used on Neural Network estimator only. This technique can also be applied on other Estimators as well.

EvolutionarySearchCV is applied with the help of a Python library called sklearn-deap [19].

The Neural Network is formed using the Keras Python library [20].

For achieving the power of efficient computation Floydhub cloud resources were used [21].

VI. CONCLUSION AND FUTURE WORK

In this paper we constructed a Neural Network model using the Pima Indian Diabetes dataset. The dataset was preprocessed and normalized for resolving several issues. Further several hyperparameters of the Neural Network was tuned using the EvolutionarySearchCV method for obtaining their optimal values with respect to the problem.

In Future, we wish to extend this work by trying out the EvolutionarySearchCV technique on the parameters like Number of Neurons in each hidden layer, Optimization algorithm to be used etc. for even better results. Other Evolutionary Algorithmic paradigms such as Genetic Programming, Classifier Learning Systems can also be tried in the context of Parameter Estimation. The execution time may get increased in the commodity computers for relatively lower computation power. We wish to eliminate this limitation by wrapping our work in an API based service in Future so that the burden of training time gets discarded.

REFERENCES

[1] . M. Bishop, "Pattern Recognition and Machine Learning (Information Science and Statistics)", Springer Book, New York, NY, USA, pp. 130-131, 2007.

[2] J. Han, M. Kamber and J. Pei, "Data Mining: Concepts and Techniques", 3rd ed., MA: Morgan Kaufmann Publishers, USA, pp. 10 - 11, 2012.

[3] T. Seganran, "Programming Collective Intelligence", O'Reilly Publishers, pp. 102 - 103, 2007.

[4] scikit-learn official documentation on parameter estimation available on: http://scikit-learn.org/stable/modules/grid_search.html

[5] D. R. Jones, M. Schonlau and W. J, "Efficient global optimization ofexpensive black-box functions", Journal of Global Optimization, vol. 13. pp. 455-492, 1998.

[6] F. Hutter, H. H. Hoos and K. Leyton-Brown, "Sequential model-based optimization for general algorithm configuration", Lecture Notes in Computer Science (Springer), vol. 6683, pp. 507-523, 2011.

[7] Y. Ganjisaffar, T. Debeauvais, S. Javanmardi, R. Caruana, C. V. Lopes, "Distributed tuning of machine learning algorithms using MapReduce clusters", Proceedings of the Third Workshop on Large Scale Data Mining: Theory and Applications, 2011, doi: 10.1145/2002945.2002947

[8] J. Snoek, H. Larochelle and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms", Advances in Neural Information Processing Systems, vol. 25, pp. 2960- 2968, 2012.

[9] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization", Journal of Machine Learning Research, vol. 13, pp. 281-305, 2012.

[10] Pima Indians Diabetes dataset available on: https://archive.ics.uci.edu/ml/machine-learning-databases/pima-indians-diabetes/pima-indians-diabetes.names

[11] T. Hastie, R. Tibshirani, J Friedman, "The Elements of Statistical Learning", Springer Series in Statistics, pp. 201-205, 2001.

[12] Mean Imputation technique available on: http://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.Imputer.html

[13] Normalization technique available on: http://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.normalize.html

[14] I. Goodfellow, Y. Bengio and A. Courville, "Deep Learning", MIT Press, pp. 406-407, 2016, available on http://www.deeplearningbook.org

[15] Gradient Descent Wikipedia article available on: https://en.wikipedia.org/wiki/Gradient_descent

[16] Cross validation technique available on: http://scikit-learn.org/stable/modules/cross_validation.html

[17]Evolutionary Algortihm Wikipedia article: https://en.wikipedia.org/wiki/Evolutionary_algorithm

[18] X. Yu and M. Gen, "introduction to Evolutionary Algorithms", Springer Publishing Company, pp. 97 - 98, 2012.

[19] EvolutionarySearchCV available on: https://github.com/rsteca/sklearn-deap

[20] Keras official documentation available on: https://keras.io

[21] Floydhub official website: https://floydhub.com/welcome