Analysis of applying Genetic Algorithm to Simple Neural Network Based on Bank Direct Marketing Dataset

Yitong Chen,

Research School of Computer Science,
Austrailian National University,
Acton ACT2601, Australia
u6147692@anu.edu.au

Abstract. Bank Marketing Dataset is imbalanced, noised and complicated, so simply applying MLP or Coscor is not practical to produce meaningful result. Using Three-Layer MLP as the basic Neural Network, this paper applies sampling on the dataset to resolve the imbalance and adopting GA to select informative attributes to form MLP's inputs. As a result, a clear improvement is achieved but significantly good result still requires further work.

Keywords: Multi-Layer Perception, Genetic Algorithm, Directed Marketing, Model Evaluation, Sampling.

1 Introduction

Previous research on Bank Marketing Dataset [1] found that the result of predicting whether a customer will subscribe a term deposit given attributes related to this customer's basic information and last contact of the current campaign using a simple Neural Network, even with the structure being constructed by Cascade-Correlation [2] method, is relatively trivial compared to the result generated by Moro S., Laureano. R. and Cortez P. [3] using simple statistical machine learning methods, such as Naïve Bayes [4], Decision Trees [5] and Support Vector Machines [6]. Though using different weights for different class in an unbalanced dataset can sometimes improve the performance of Neural Network, applying it to this model is either changing the outcome from mapping all the customer to positive class (subscribing a term deposit) to mapping them all to negative class or leading to a low precision and low recall outcome. Therefore, previously generated prediction models using Multi-Layer Perception (MLP) cannot provide any useful information for the bank to direct its marketing since whatever the result is, the proportion of positive class is not ideal for it to determine its marketing direction. What's more, adopting Cascade-Correlation structure to generate Neural Network with better-structure doesn't improve the result significantly, which suggests that the problem with this model may lie in the dataset instead of the Neural Network parameters. One possible explanation for this problem is that the original dataset contains certain amount of useless information while the customer's behaviour is not completely depending on the provided data, therefore, with a noised dataset and significant complex relation between input data and the output classification the prediction accuracy can be reasonably low.

In real world, some data are hard to collect, which means it may take long time to expand the dataset. Under this circumstance, data analysis should make the most from the existing dataset, even bearing its enormous noise and complicated relationship with desired target. As Genetic Algorithm (GA) [7] being capable of exploring significantly wide range of input combination and simulating annealing to getting out of local optimisation, in this research it is adopted to explore possibly better outcome from this Bank Marketing dataset.

Imitating the behaviour of chromosomes when producing offspring, the input combination is represented by a chromosome-like structure on which each 'gene' represents the selection state of an input attribute in GA and make them crossover to produce new generation of input combinations [7]. After several times of reproduction, new input combinations are expected to perform better than the original combination upon the same model [7].

However, in the previous research only the minimum version of the dataset is used to train the prediction model. In this paper, a larger dataset with exactly same attributes and an expanded dataset with some extra attributes are used to evaluate the outcome. As all the datasets used in this paper are imbalanced, it is significantly important to resolve this problem. Therefore, sampling is used to get a balanced sample from the original data for training to tackle this problem.

2 Method

2.1 Data Pre-processing

In this research, the pre-processing method is slightly different from that of the previous research. Strictly categorical attributes, including 'job' (type of job of the customer), 'marital' (marital status), 'education', 'contact' (contact communication type), 'month' (last contact month of year), 'day_of_week' (last contact day of the week) are all divided into categorical value attribute, for example, as 'contact' attribute has 'cellular' and 'telephone' categories, it is divided into 'cellular' attribute and 'telephone' attribute: customers using telephone to communicate has their 'telephone' attribute set to 1 and 'cellular' attribute set to 0. Meanwhile, as other categorical attributes only contain 'yes', 'no'

and/or ‘unknown’ categories, they are simply encoded to represent ‘yes’ by 1, ‘no’ by -1 and ‘unknown’ by 0 to indicate the status.

In the expanded dataset, extended social and economic context attributes are all numerical values, thus they are directly used without encoding.

Detailed meanings of attributes are given in the ‘.txt’ file with Bank Marketing Dataset.

After encoding, all the non-target attributes are normalized.

2.2 Implementation

2.2.1 Data Sampling

The datasets used in this research is extremely imbalanced with the ratio between negative class and positive class being 8:1 approximately. Aimed at resolving the imbalance of datasets, after separating datasets into training set and test set, new training sets are sampled from original training sets, having the same number of positive class and negative class. Two MLPs with the same hyper-parameters are trained separately with the original training sets and new training sets to evaluate the balancing effect of adopting sampling method.

2.2.2 Neural Networks

For simple Neural Networks, several Three-Layer full connection MLPs (containing 1 hidden layer) and Four-Layer full connection MLPs (containing 2 hidden layers) are constructed with different hyper-parameters. As the results for Four-layer connection MLPs don’t change significantly from those for Three-layer MLPs, in this paper only the results of Three-Layer MLP and corresponding GA are discussed.

Since the goal is to use GA to select informative attributes, the dimension of input data is changing. Therefore, number of input neurons is dynamically defined during evolution process. As a binary classification task, the number of output neurons in the Neural Network is 2. Adjacent layers are fully linear connected with sigmoid activation function.

2.2.3 Genetic Algorithm

The Genetic Algorithm used in this research is proposed by Goldberg, D.E. & Holland, J.H. [8]. Different selection of attributes is represented as bit-string consisting of 0 and 1. Each digit represents the selection of an attribute while 0 represents that it is not used. Then a population of datasets containing only partial selections of their attributes are used to train the Neural Network to generate a model for the prediction task with the harmonic mean of precision and recall - f1 score [9] to represent the performance of this Neural Network. To possibly produce better selection for better prediction result, a new population is generated in which individuals (a particular selection of attributes) are produced by crossing over two selections of the previous population. Since individuals in the previous population have different performance and the individuals with better performance tend to produce better offspring (new individuals), the probability of an individual being selected as a parent (original selection for cross over) is positively related to its performance [7]. However, simply using cross over method is easy to be caught into a local optimization and produce trivial result, the mutation method is adopted to prevent this situation [7]. In mutation period, every digit in the bit-sting representing a selection has a fixed little possibility to change, which introduces uncertainty in new population and makes it possible to explore wider selection range. After a certain frequency of reproducing, the offspring is highly possible to be the globally best selection under the evolution environment with proper parameters.

3 Result and Discussion

3.1 Evaluation

3.1.1 Evaluation Methods

On the purpose of demonstrating the self-learning process of MLP, all the losses calculated by loss function in training epochs are plotted for each MLP. Meanwhile, after the training and testing process, two confusion matrices are used to show the MLP’s learning outcome related to the training set and its prediction performance related to the test set.

F1 score is chosen for numerical evaluation in this research. As a commercial prediction task, the percentage of True Positive (TP) is significantly important. However, as in extreme situations both recall and precision [9] can have 100% accuracy with trivial result, for example, all the instances are mapped to positive class or only a tiny proportion of

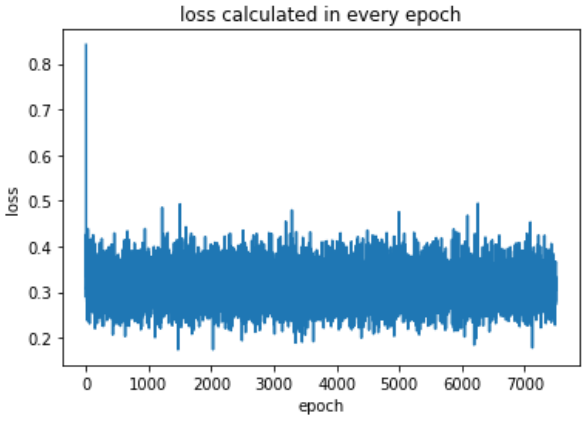
positive class is correctly labelled but no negative class is wrongly labelled, choosing their harmonic mean - f1 score can demonstrate the result much more meaningfully.

In GA, to visualize a generation's (population in a certain evolution step) performance, the image of a coordinate system where the x-axis is the numerical representation (mapping bit-string to numerical value) range of the selection and the y-axis is the f1 score. Points plotted in this coordinate system represent individuals' (selection of attributes represented by bit-string) performances.

3.1.2 Three-Layer Neural Network

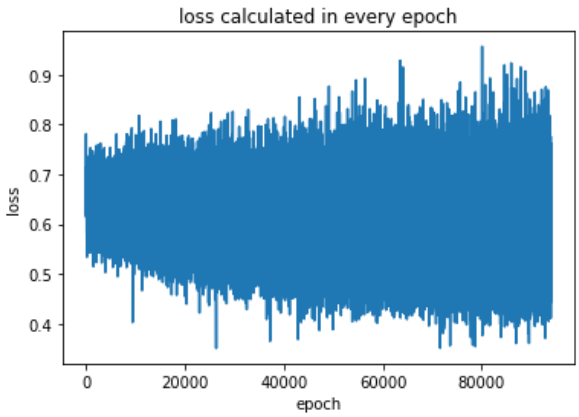
3.1.2.1 Three Layer MLP

At the stage of choosing Three Layer MLP's hyperparameters, after comparing different parameter set with learning rate ranging from 0.001 to 0.1, batch size ranging from 1 to 500, epochs ranging from 500 to 20000, hidden layer size from 80 to 300, it is found that the relatively better results for imbalanced dataset are generated with learning rate in range 0.008 to 0.02, batch size in range 180 to 350, epochs in range 300 to 600, hidden layer size being 280 to 450. The average training set accuracy is 89% while the average test set accuracy is 88% with f1 score being 16% in average. One outcome above the average is shown in fig. *1* (hidden layer size=300, epochs=500, batch size=250, learning rate=0.01). Differently, the relatively better results for balanced dataset (sample size=3000+3000) are generated with learning rate in range 0.001 to 0.01, batch size in range 32 to 150, epochs in range 300 to 600, hidden layer size being 150. The average training set accuracy is 68% while the average test set accuracy is 70% with f1 score being 30% in average. One outcome above average is shown in fig. *2* (hidden layer size=150, epochs=500, batch size=32, learning rate=0.001).




```
Training set Accuracy: 89.29 %
Confusion matrix for training:

 3190    24
  365    52
[torch.FloatTensor of size 2x2]

Testing Accuracy: 89.10 %
Testing Accuracy: 89.10 %
recall=12.50%, pricesion=68.42%, F=21.14%
Confusion matrix for testing:

 780    6
  91   13
```

Fig. 1. *MLP Performance on Imbalanced Dataset*

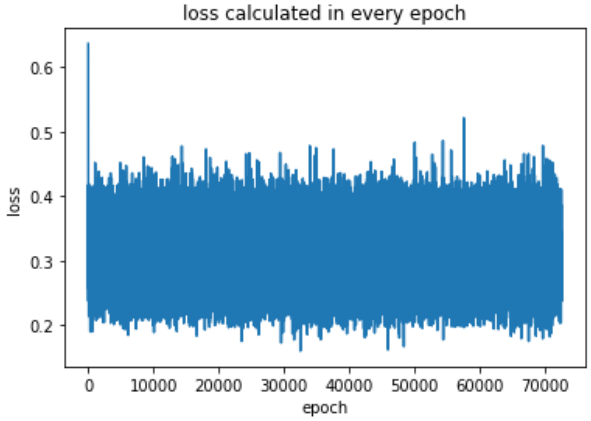



```
Training set Accuracy: 68.03 %
Confusion matrix for training:

 2180   820
 1098  1902
[torch.FloatTensor of size 2x2]

Testing Accuracy: 71.51 %
Testing Accuracy: 71.51 %
recall=67.31%, pricesion=24.22%, F=35.62%
Confusion matrix for testing:

 565  219
  34   70
```

Fig. 2. *MLP Performance on Balanced Dataset (sampled)*

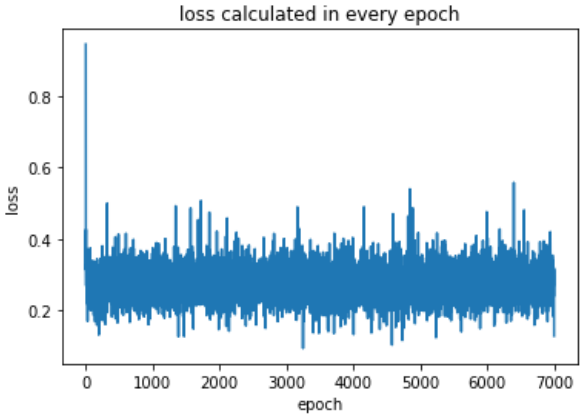
Comparing the result for imbalanced dataset and balanced dataset given their average statistics, it is found that though the testing accuracy is higher with the imbalanced dataset, its f1 score is lower since it maps almost all the customers to the negative class. As shown in the confusion matrix in fig. 1 for training set, the pattern of positive class is not properly learnt. Therefore, the sampling method is considered to help MLP learn patterns better in this situation despite the relatively low accuracy.

Aiming at checking if an enlarged dataset (simply containing more customers with no other added attributes) or an extended dataset (having more attributes, related to customers' social and economic context) is helping the MLP to learn better in this situation, the hyperparameters used to produce fig. *1* is used in MLPs training on the enlarged dataset and the extended dataset. The average training set accuracy and test set accuracy are not changed using the enlarged dataset, however, the average f1 score is increased by 4% compared to that of the original dataset. One average outcome is demonstrated in fig. *3*. As for the extended dataset, the average training set accuracy is 90% and the average test set accuracy is 91% with f1 score being 33% in average. One average outcome is demonstrated in fig. *4*.




```
Training set Accuracy: 89.01 %
Confusion matrix for training:

 31529    412
  3567    684
[torch.FloatTensor of size 2x2]

Testing Accuracy: 89.27 %
Testing Accuracy: 89.27 %
recall=17.44%, pricesion=61.99%, F=27.22%
Confusion matrix for testing:

 7870   111
  857   181
```

Fig. 3. *MLP Performance on enlarged Dataset*

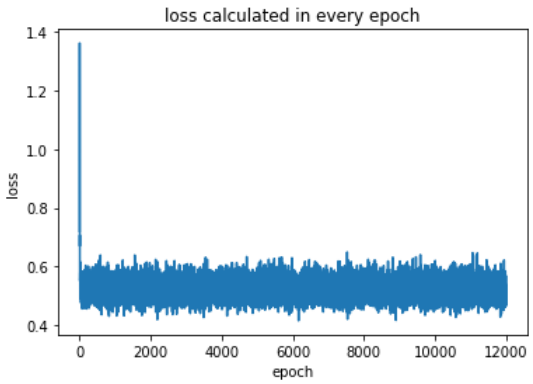



```
Training set Accuracy: 90.38 %
Confusion matrix for training:

 2910    43
  276    86
[torch.FloatTensor of size 2x2]

Testing Accuracy: 90.17 %
Testing Accuracy: 90.17 %
recall=23.60%, pricesion=65.62%, F=34.71%
Confusion matrix for testing:

 704   11
  68   21
```

Fig. 4. *MLP Performance on extended Dataset*

Comparing the results for imbalanced original dataset and enlarged dataset, it is found that the enlarged dataset is slightly better for MLP to learn patterns but the advantage gained from performance doesn't outweigh the disadvantage of vastly increased computational time. On the contrary, the comparison between training on the original dataset and training on the extended dataset shows that more relevant data can significantly improve the MLP's performance while not increasing the computational time by a great deal.

As a result, sampled extended dataset is chosen for further usage. The average testing accuracy is 77% and the average f1 score is 44% for this selected dataset and one average outcome is shown as fig. *5*. Though this dataset seems to perform best, the f1 score and accuracy are not ideal enough to produce perfectly meaningful information. Therefore, GA is used to search for informative attributes in this dataset and try to come up with a better prediction model.




```
Training set Accuracy: 75.00 %
Confusion matrix for training:

 2518   482
 1018  1982
[torch.FloatTensor of size 2x2]

Testing Accuracy: 80.25 %
Testing Accuracy: 80.25 %
recall=64.55%, pricesion=33.97%, F=44.51%
Confusion matrix for testing:

 648  138
  39   71
```

Fig. 5. *MLP Performance on sampled extended dataset*

3.1.3.2 Genetic Algorithm

With different hyperparameters for GA, it can perform significantly different results. As small population can drop good genes easily while large population requires a great deal of computational time, low crossover rate makes the evolution slow while extreme large cross over rate cannot store information from previous generation for stability, large mutation rate can destroy the evolution process while with nearly 0 mutation rate generations are easily caught in local optimization, after several testing to balance evolutional effect and computational time, the GA used in this research has 10 populations per generation, 0.8 crossover rate, 0.01 mutation rate and 20 generations with its DNA size set to be the number of input attributes. In each generation, datasets for training and testing are refreshed using sampling as to make the model applicable to the whole dataset instead of its subset. As before, f1 score is used to evaluate the MLP's performance, thus the fitness function is simply using f1 score.

Results for some generations in GA are shown in fig. 6. As a general trend, populations get similar as the evolution processing. Though the f1 score is not always increasing due to the different initialised values for MLP weights, it is getting stable and relatively higher than random cases.

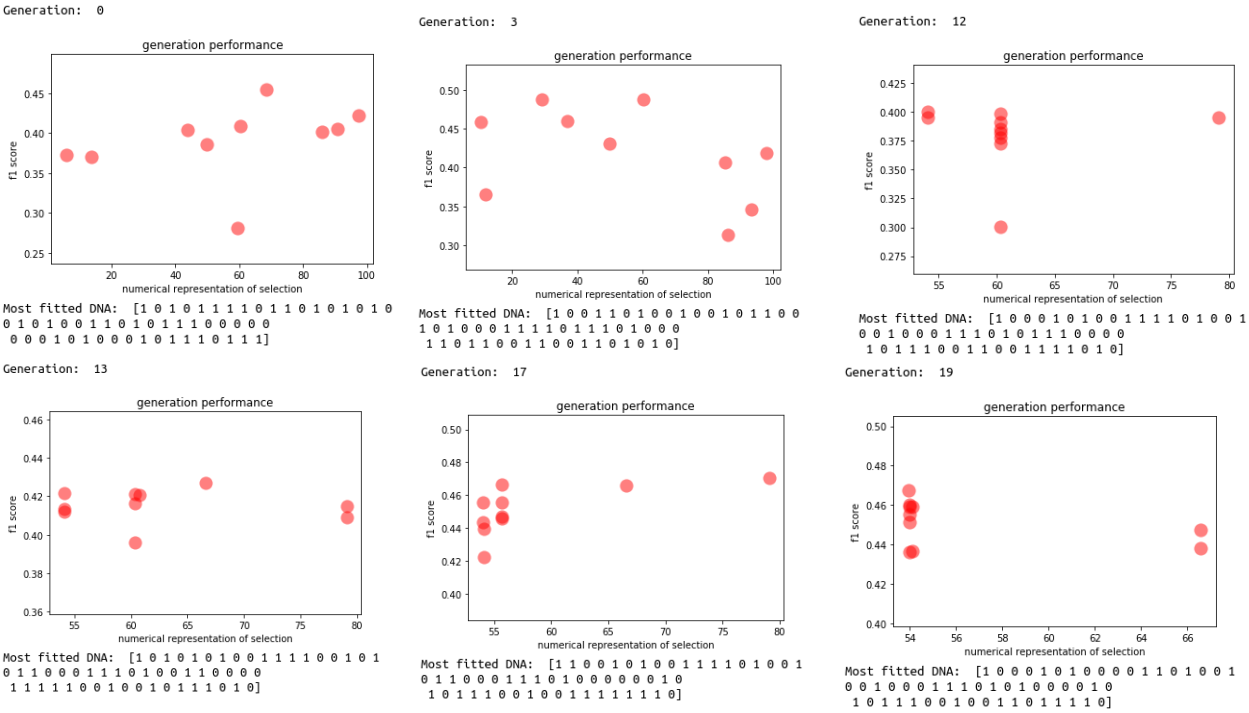


Fig. 6. *GA Generation Results*

From fig. *6* it is found that after 20 generation the f1 score is still not ideal and it is never larger than 55%, which indicates that the dataset is either noised a lot or the relationship is significantly complicated to be predicted perfectly using this model. However, the prediction information can still be useful as further campaign can be directed to customers of false positive class to convert them into true positive class since they get similar characteristics with those in the true positive class, which means they tend to have the potential to subscribe a term deposit in the future.

4 Conclusion and Future Work

Analysis of an imbalanced dataset usually requires adopting methods to resolve the imbalance to generate meaningful results. If the dataset doesn't have too much noise and its relationship between attributes and prediction classes is clear e.g. the glass classification dataset [10], simply set different weight for different class and sampling can both generate good result. However, for the Bank Marketing Dataset used in this research, due to its imbalance, noise and complicated relationship, even after sampling and weighting different class differently the result is still not ideal.

As an input attributes selection method, GA is only capable of showing which attribute combination generates the best outcome instead of practically improving the performance. However, this conclusion indicates that GA can be used to determine which data to be collected if the related attributes are numerous and collecting all of them for every entity is taking too much time and space.

To further analysis this dataset, suggestion is that using domain knowledge to analyse the relationship between attributes in attribute combinations selected by GA and the true classes. If the relationship is clear, then this relationship can be used to direct bank marketing. Assuming it is unclear, if new similar but different attributes can be collected, the GA and MLP method above can be adopted repeatedly to generate possibly better result.

Since sigmoid-like activation functions and linear activation functions are vastly used but they are all monotonically increasing functions, some complicated activation functions like e^x, x^2, can be tested on MLP. Intuitively adopting these complicated functions may widen MLP's ability to model complex relationships. If that is the case, then the powered MLP may also be better models for more complicated real world classification and regression tasks.

References

1. http://archive.ics.uci.edu/ml/datasets/Bank+Marketing
2. Fahlman S.E. & Labiere C.: The cascadecorrelation learning architecture, *Advances in Neural Information Processing Systems 2*. Morgan Kaufmann. Pp 525–532 (1989)
3. Moro S., Laureano. R. & Cortez P.: Using Data Mining for Bank Direct Marketing: An Application of the CRISP-DM Methodology, *Proceedings of the European Simulation and Modelling Conference - ESM'2011*, pp. 117-121, Guimaraes, Portugal, October, 2011. EUROSIS (2011)
4. Zhang, H.: "The Optimality of Naïve Bayes", *Proceedings of the 17th FLAIRS conference*, AAAI Press (2004)

5. Aptéa, C. & Weiss, S.: “Data mining with decision trees and decision rules”, *Future Generation Computer Systems 13, No.2-3*, 197–210 (1997)
6. Cortes, C. & Vapnik, V.: “Support Vector Networks”, *Machine Learning 20, No.3*, 273–297 (1995)
7. Andries, P. E.: *Computational Intelligence: An Introduction (Second Edition)*, pp 143-176 (2017)
8. Goldberg, D.E. & Holland, J.H.: *Machine Learning 3: 95*. https://doi.org/10.1023/A:1022602019183 (1988)
9. Sasaki, Y.: The truth of the F-measure. *Teach Tutor Mater*. (2007)
10. http://archive.ics.uci.edu/ml/datasets/Glass+Identification