Neural Network on Prediction for Potential Customers of Bank

Xiaohan Wang

Research School of Computer Science, Australian National University

Abstract. A neural network, which can classify customers whether they will subscribe the term deposit in bank according to some of their personal information and contact information with bank, is implemented in the work. For increasing the accuracy of the neural network, some changes have been attempted, such as sampling method, the topology of neural network, the number of epoch and other probably relevant factors. The genetic algorithm is used to find the most proper value for the number of hidden units. Bimodal distribution removal is a technique to remove outliers in training data and has also been implemented when the neural network fits training data. The result, that the accuracy of the neural network is 89.25%, is little worse compared with that in a research paper reporting results on the same data set. Finally, conclusion about this work is drawn and some suggestion for future work is made.

Keywords: neural network, classify, accuracy, bimodal distribution removal, genetic algorithm

1 Introduction

Unlike other common companies, banks have access to phone number of customers, and their phone calls usually will not be felt as annoying as that from business companies, so phone calls are a practicable method for bank to promote products. S. Moro, P. Cortez and P. Rita (2014) believe that if some methods are based on accessible information and can help predict whether the customer will subscribe the product, the bank can choose to make a phone call with customers who are more likely to subscribe the term deposit than others. This will dramatically increase working efficiency on marketing campaigns, and for access to information about customers and contact between customers and banks, the Bank Marketing data set is chosen.

However, selecting sample is a huge problem because there are only 4,640 instances whose targets are ‘yes’ (target ‘yes’ means this customer will subscribe the term deposit) in the whole data set 41,188 instances. The decision of what kind of the neural network topology is also difficult to made, and with method bimodal distribution removal, the patterns which should be removed is hardly to figure out.

To model the neural network, firstly a neural network with one hidden layer is established, and then one hidden layer changes to two. Secondly, back-propagation method is introduced into modeling the neural network. Thirdly, the change on sampling method is made from simple random sample to stratified sample. Fourthly, the number of hidden layers changes back to one, and the genetic algorithm is used to find the most proper value for the number of hidden units. Finally, bimodal distribution removal is taken to remove outliers from training data. For each time after a major change, some other settings such as the number of epoch and learning rate are adjusted for several times to find a better model.

The final neural network achieves the accuracy for approximate 89.25% and can hardly predicts the patterns in class whose frequency is low.

2 Method:

2.1 Data Obtain

The dataset that has been chosen is Bank Marketing (with social/economic context), which is created by Sérgio Moro (ISCTE-IUL), Paulo Cortez (Univ. Minho) and Paulo Rita (ISCTE-IUL) in 2014, and this dataset is from the UCI Machine Learning Repository.

There are 22 features in Bank Marketing dataset: age, job, marital, education, default, housing, loan, contact, month, day_of_week, duration, campaign, pdays, previous, poutcome, emp.var.rate, cons.price.idx, cons.conf.idx, euribor3m, nr.employed and y. Except the last one, all features are predictors; and the last one feature 'y' is the target. Target 'y' has only two values 'yes' and 'no', which means all these instances are in the two classes.

2.2 Raw Data Processing

These features will be used for fitting the neural network, therefore some features which are not in numeric type need to be transformed into which in numeric type. The accurate transformations are below:
'job' valued with 0, 1, …, 11 (from 'unknown', 'admin', …, 'unemployed')
'marital' valued with 0, 1, …, 3 (from 'unknown', 'divorced', …, 'single')
'education' valued with 0, 1, …, 7 (from 'unknown', 'basic.4y', …, 'university.degree')
'default' valued with 0, 1, 2 (from 'unknown', 'no', 'yes')
'housing' valued with 0, 1, 2 (from 'unknown', 'no', 'yes')
'loan' valued with 0, 1, 2 (from 'unknown', 'no', 'yes')
'contact' valued with 0, 1 (from 'telephone', 'cellular')
'month' valued with 1, 2, …, 12 (from 'jan', 'feb', …, 'dec')
'day_of_week' valued with 1, 2, …, 5 (from 'mon', 'tue', …, 'fri')
'poutcome' valued with 0, 1, 2 (from 'nonexistent', 'success', 'failure')
'y' valued with 1, 0 (from 'yes', 'no')
and after that, for all the features, every value of them has been normalized by the formula: x = (x – x.mean()) / s.std(), which means each value of the feature equals to itself subtracting the mean value of this feature and then dividing standard deviation of this feature. This operation can balance the influence of all features on modelling the neural network.

2.3 Sampling

The sampling method used in this neural work is stratified sampling instead of simple random sampling, because simple random sampling can hardly select out patterns in class 'yes'. In the whole data set, there are 36,548 instances in class 'no' and 4,640 instances in class 'yes', and if the sample data is directly selected from the original data set by simple random sampling, the probability is pretty high that nearly none patterns in class 'yes' are selected into the sample data when sample data selects few instances from data set. This leads that the neural network nearly only fits the data in class 'no' and is not sensitive to instances in class 'yes'.

Stratified sampling can effectively avoid this problem, with selecting instances in different class according to their ratio to the whole data set. Firstly, the data set is partitioned into two groups according to the two classes of all instances. Secondly, instances in different groups are selected out in proportion, and in this case the ratio of instances in class 'yes' to instances in class 'no' is approximately 1 to 8. Finally, the obtained sample data set contains 80 instances in class 'yes' and 640 instances in class 'no', and the percentage of each class in sample data set is the same with that in original data set. In addition, 400 patterns are separately and randomly selected as the testing data.

2.4 Cross-validation

The normalized data set is randomly divided into 10 groups, and the neural network fits these data for 10 times. For each time, one piece of data is for testing the model, and for all ten-time testing, the testing data is different with the rest data for training.

Then the mean accuracy of the neural network is calculated from ten accuracies obtained from previous testing. Finally, the neural network fits from new data for testing, and an accuracy is obtained. The method 10-fold cross-validation can prevent over-fitting, and only wastes 10 percent of the data, however the training process costs 10 times more time than before because of this method.

It has been used in the early stage of the experiment, and in this survey, it helps find which classes the removed patterns belong to. Because of expensive computational complexity, when the genetic algorithm is used in the later stage of the experiment, the cross-validation method has not been used anymore.

2.5 Neural Network

In the neural network, the number of input neurons and output neurons is the same with the number of features (20) and target (2) respectively, and the number of hidden neurons is 20. For several attempts, 10, 30, 50 hidden neurons also have been used, however in the later stage of the experiment, the genetic algorithm is used to find the most proper number of hidden units. The back-propagation function can dramatically increase the accuracy of the model, so it is used when training the neural network. In addition, because the genetic algorithm and the method bimodal distribution removal dramatically increase the computational complexity, the decision of implementing 2-hidden-layer neural network is abandoned, and the number of epoch is also adjusted to 2000, but influenced by bimodal distribution removal, in fact the training stops before that.

2.6 Genetic algorithm

In the experiment, the genetic algorithm is chosen to help find the best number of hidden units in the neural network. The fitness function helps determine the fitness degree of the number of hidden units, and the selection function is used to randomly select the most proper values. The crossover function is used to produce new values for the number of hidden units based on the previously selected value. The mutation function is used to randomly make some changes to these new values. In addition, for simplicity, in a generation, when their numbers of hidden units are the same, the results are directly obtained without training and equal to the previous one.

In the experiment, the sizes of population and generation are 10 and 5 respectively, and the probabilities of DNA crossover and mutation are 0.8 and 0.1 respectively.

2.7 Bimodal distribution removal (Slade, P., & Gedeon, T. D., 1993)

Normally, when the neural network has been trained for 200-500 epoch, outliers can be gradually identified by the neural network because their extremely higher error than the that of other patterns. After removal of outliers from input patterns, the neural network is less influenced by them, and can have higher accuracy than before.

The formula for bimodal distribution removal to figure out outliers is

error >= mean(error) + a * std(error) where 0 <= a <= 1

When error of the pattern is larger than the mean of the errors for all the patterns plus the standard deviation of the errors multiplying a ratio, this pattern will be removed. When the variance of errors becomes approximately 0.1, patterns which are regarded as outliers can start to be removed, and the removal action is taken for each 50-epoch until variance of errors decreases to below 0.01. (Slade, P., & Gedeon, T. D., 1993)

However, in practice, it is difficult to decide which input patterns to remove, and the two reasons are:

Firstly, the condition of starting patterns removal is not clear. When the variance of errors of all patterns approximately equals 0.1 or below, the removal can start, and it normally happens within 200-500 epoch. The distribution of error frequencies is presented in Figure 1 and Figure 2. In practice, from 200 to 500, each number of

epoch with 50-epoch increment has been tried, and it shows little influence, so the epoch 300 is selected as the condition of starting outlier removal.

Figure 1: the frequency of loss (when the number of epochs is 200)

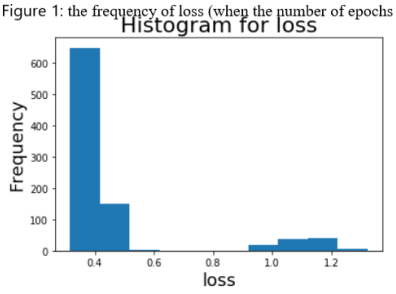
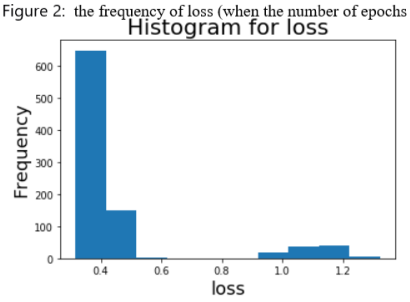


Figure 2: the frequency of loss (when the number of epochs is 500)



Secondly, it is difficult to decide the value of the ratio that is mentioned in the formula even if it is in a little range from 0 to 1. The 9 attempts with the value 0.1, 0.2, …, 0.9 have been taken, and it finds that the value below 0.5 is worse than that above 0.5. The value 0.9 is selected in the training.

In addition, because the error of each pattern is required by the method bimodal distribution removal, the neural network is set with one hidden layer instead of two hidden layers for speeding up training.

3 Results and Discussion:

The accuracy in the experiment is approximately 89.25%, which is little worse than that (91.4%) from the research paper “A Data-Driven Approach to Predict the Success of Bank Telemarketing”, published by S. Moro, P. Cortez and P. Rita in 2014.

For the genetic algorithm, it finds that changes of numbers of hidden units have little influence on the accuracy of the neural network in the experiment. Even though randomness exists in training attempts, the value 11 for the number of hidden units can help the neural network achieve slightly better accuracy than others according to the results in generation 2, 3 and 4 (Figure 3 for the results is below). Without the implementation of the bimodal distribution removal, the models can achieve accuracy from 89.25% to 90.25% in the experiment. When bimodal distribution removal technique is used, it shows that all values of the model accuracy are 89.25% no matter how many hidden units the neural network has (Figure 4 for the results is below). The conclusion can be drawn that the technique bimodal distribution removal has negative influence on training the neural network in the experiment.

Figure 3: the results without the technique (bimodal distribution removal):

Generation:	**0**	**1**	**2**	**3**	**4**
Number of hidden units – Accuracy (%):	5 – 89.25	5 – 89.25 4 – 89.25 15 – 90.25	15 – 89.25 3 – 89.25 5 – 89.25 11 – 90.25 20 – 89.75 14 – 89.25	11 – 90.00 1 – 89.25 20 – 89.25 17 – 90.75 14 – 89.25	6 – 89.25 13 – 89.25 11 – 89.75 2 – 89.25 18 – 89.75 16 – 89.25 9 – 89.25

(notice: in each generation, the same results are not presented.)

Figure 4: the results with the technique (bimodal distribution removal):

Generation:	**0**	**1**	**2**	**3**	**4**
Number of hidden units – Accuracy (%):	10 – 89.25	10 – 89.25 20 – 89.25 7 – 89.25 9 – 89.25 15 – 89.25	15 – 89.25 10 – 89.25 18 – 89.25 5 – 89.25 9 – 89.25 14 – 89.25 20 – 89.25	4 - 89.25 10 – 89.25 7 – 89.25 18 – 89.25 5 – 89.25 20 – 89.25 19 – 89.25 1 – 89.25	18 – 89.25 5 – 89.25 2 – 89.25 11 – 89.25 19 – 89.25 20 – 89.25 3 – 89.25 4 – 89.25

(notice: in each generation, the same results are not presented.)

Before the implementation of bimodal distribution removal, the models can achieve accuracy in different values, and can predict different target classes (some confusion matrixes are provided below).

When the accuracy is 89.25%:

Actual ~~Predicted~~	Class ‘yes’	Class ‘no’
Class ‘yes’	0	43
Class ‘no’	0	357

When the accuracy is 90.00%:

Actual ~~Predicted~~	Class ‘yes’	Class ‘no’
Class ‘yes’	6	37
Class ‘no’	3	354

When the accuracy is 90.75%:

Actual ~~Predicted~~	Class ‘yes’	Class ‘no’
Class ‘yes’	6	37
Class ‘no’	0	357

However, with the implementation of bimodal distribution removal, the models can hardly predict different target classes. In the experiment, they achieve accuracy in only one value 89.25%.

When the accuracy is 89.25%:

Actual ~~Predicted~~	Class ‘yes’	Class ‘no’
Class ‘yes’	0	43
Class ‘no’	0	357

It is worth being mentioned that sometimes these models with implementation of BDR (bimodal distribution removal) can also predict patterns in different classes and achieve accuracy in different values. When this situation happens, the neural network has slightly higher accuracy compared with that without implementation of BDR.

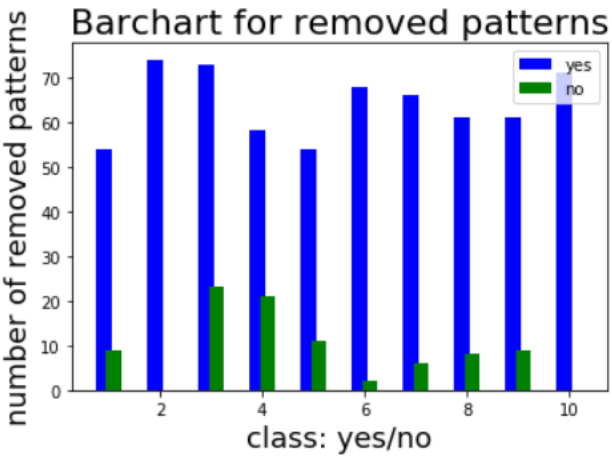
In some situation, bimodal distribution removal is little helpful and even have an adverse impact on training neural network, because some patterns that are not outliers might be regarded as outliers by the model by mistake and are removed.

As what has been mentioned before, in the whole dataset, there are 36,548 instances in class ‘no’ and only 4,640 instances in class ‘yes’. Even if stratified sampling can effectively avoid the situation that nearly no instances in class ‘yes’ are selected into sample data, in stratified sample the 1-to-8 ratio of instances in two classes is still significantly unbalanced, which causes the neural network is more sensitive to class ‘no’ than to class ‘yes’. Patterns in class ‘yes’ always cause higher errors than that in ‘no’ class because the neural network is more familiar with patterns in class ‘no’ which hold a large proportion in input patterns.

Therefore, in each training sample dataset, if there are no outliers which can cause extremely high errors, the most of patterns with little higher errors than others will be removed, but in fact, these removed patterns are mostly probable with target ‘yes’ (see Figure 5), and the number of patterns in class ‘yes’ becomes smaller than before even if it is originally small. The consequence is that trained neural network only has fit patterns in class ‘no’ and losses ability to classify input in a certain extent.

However, if outliers exist in sample training dataset and can cause extremely high errors, these outliers will be removed by BDR. The accuracy of the neural network increases slightly, but this situation happens for few times, because the sample training dataset only contains 720 patterns and has little probability of containing outliers.

Figure 5: the removed patterns in classes ‘yes’ and ‘no’ with 10-fold cross-validation



In addition, about computational complexity, for each step in an epoch, a batch of patterns could have been input together to train the neural network, and a total error of those patterns in the same batch would be calculated. However, the method bimodal distribution removal requires the error of each patterns to identify outliers. Therefore, only one patterns can be input into the neural network for each step if bimodal distribution removal is implemented, and this causes the increase of computational complexity.

On the one hand, the main problem is that sometimes the neural network can only figure out patterns in class ‘no’ which take a huge proportion in dataset, and the high accuracy is because most of the patterns in testing data are in class ‘no’. The technique bimodal distribution removal also contributes to the situation as the removed patterns are mostly in class ‘yes’. In addition, little feature analysis has been done when feature selection, it might indirectly increase the complexity of modelling the neural network and has some adverse impacts on the accuracy of the model.

On the anther hand, several techniques increase the performance of the neural network, such as stratified sampling, back-propagation. Stratified sampling helps avoid the situation that the number of instances selected into sample data is extremely

small because of its low ratio in the whole dataset, and in a certain extent it also reduces the instability of the sampling quality. Back-propagation dramatically increases the performance of the neural network by optimizing the values of weights.

4 Conclusion and Future Work

In the survey, the neural network is trained for predicting the potential customers of the banks. The Bank Marketing dataset is selected and processed by transformation and normalization. The sampling method is stratified sampling, and the genetic algorithm is used to find the most proper number of hidden units for the neural network. The technique bimodal distribution removal is used to remove the outliers in training data. The cross-validation method is used when finding which patterns are removed, but not used in later stage of the experiment because the genetic algorithm dramatically increases the computational complexity. The final neural network has the accuracy for approximate 89.25%.

Combined the Bank Marketing dataset, bimodal distribution removal method and other factors, the neural network finally obtained does not performed well because of its insensitivity of instances whose frequency is much lower than others. Therefore, the main problem that needs an appropriate solution is how to handle the significantly unbalanced ratio between patterns in different classes. Besides stratified sampling, some more appropriate methods of sampling can be found. New topology of the neural network can be specifically designed for handing the problem about unbalanced ratio of instances in classes. Feature selection with genetic algorithms may improve the performance of the neural network. For the technique of outlier removal, bimodal distribution removal needs some necessary improvement. For example, a more accurate formula, which calculates the condition which patterns should be removed, can be raised by some accessible methods.

Reference

Moro, S., Cortez, P., & Rita, P. (2014). A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems, 62*, 22-31. doi:10.1016/j.dss.2014.03.001

Moro, S., Cortez, P., & Rita, P. (2014). Bank Marketing Data Set. UCI Machine Learning Repository. Retrieved from: https://archive.ics.uci.edu/ml/datasets/bank+marketing.

Slade, P., & Gedeon, T. D. (1993, June). Bimodal distribution removal. In International Workshop on Artificial Neural Networks (pp. 249-254). Springer, Berlin, Heidelberg.