International Journal of Engineering & Technology, 7 (4) (2018) 4411-4414

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET
doi: 16740
Research paper

Predictive analysis using resilient and traditional backpropagation algorithm

Shriansh pandey [1] *, Stuti Bajpai [2], Addepalli VN. Krishna [3]

[1] 4th Year, CSE, Christ Deemed to BE University
[2] 3rd Year, CSE, Christ Deemed to BE University
[3] Professor, CSE, Christ Deemed to BE University
**Corresponding Author E-Mail: Shrianshpandey05@gmail.com*

Abstract

Artificial neural networks can be used in many applications like analysis, manipulation, predictions on the given statistical data. Neural-net is the function which is used to train the multi-layered perceptron when discussed about the regression analysis ,i.e to maximize the func-tional relationship between predictors(input variables) and the response variables(output). Hence neural networks can be used as an exten-sion of generalized linear models(supervised learning). This paper deals with the brief introduction about multi-layered perceptron and the validation of resilient backpropagation and the traditional backpropagation algorithms. A real time bank dataset is considered and predictions is being done on the Churning rates of Customers by the above two algorithms. It is observed that resilient backpropagation algorithm gives more accurate and precise results in comparison with traditional backpropagation algorithm by the number of steps taken by both the algorithms in convergence of the whole neural network and the error rate of both the algorithms.

Keywords: *Artificial Neural Networks; Neural Net; Perceptron's; Rprop; Backdrop*

1. Introduction

Neural networks is a platform which is built to train the multi-layered perceptron in context of Regression analysis. The relation-ship between the covariates and the response variables is very useful and of great heed. For example when performing the simu-lation of some complex problem , the possible risk factors and their effects on the problem analyzed to find the risk factors that can be used to evolve the hindrance strategies. This Artificial neu-ral networks (ANN) can be used to implement almost at every convoluted functional relation. In ANN it is not mandatory to state the relationship between input variables (covariate) and output (response variables) for example as linear combination. Thus, Artificial neural network is a very constructive demographic tool. Any particular dataset in ANN is used to train the neural network and the resemblance of relationship adapts it's parameters by it-erative method. Neural network is also used in generalizing the linear models.

The package NEURALNET [1] in R programing caries a resilient function , used to train the feed forward neural networks means to resemble the functional relationship in a particular situation. It can supervise the random number of input and output variables and also the hidden layers and neurons even when the multiplexity is high with high estimation costs. As a result of this the iteration can stop early because of maximum iteration steps, defined by human , reached before the clustering of algorithm. The neuralnet package provides a function to envision the results. For example plot() function is used to visualize the whole graph of covariates , hidden layers , response variables. The resilient Backpropagation algo-rithm is one of the fastest algorithm(three different versions are implemented and traditional backpropagation used for the compar-ison purpose) which is used for the computation purpose , training of neural network in order to perform the predictive analysis of a particular dataset, to solve the multiplex/complex problems. In resilient backpropagation algorithm the user or can define the hidden layers , which can minimize the computational costs by including more layers which leads to more accuracy and lesser and lesser neurons per layer. Neuralnet compact the gap regarding the algorithms provided for a neural networks in R.

2. Literature survey

[3]The central idea of this article is to introduce supervised learn-ing in multi-layer precetrons based on technique of gradient de-scent. The study of different adoptive studies are discussed too in the process. The article also puts forward behaviour of certain learning procedure on certain problems hence giving idea on vari-ous properties of these algorithms. [2]A direct approach to fram-ing and modelling non normal data. This is applicable when on the normal data the usual regression assumptions are not applied. Since non-normal data is encountered quiet frequently this ap-proach comes very handy. Hence the use of generalized linear model is spreading in various fields such as Environmetrics. [4]In this paper it has been mentioned about the new algorithm which is known as RPROP . the resilient backpropagation algorithm.. it has been mentioned that RPROP is simple and easy to implement and compute the local learning scheme by modifying the update values of each weights as per the nature of sequence of sign of partial derivatives in each dimension of weight space. RPROP is having a very a important features the Robust nature and the time conver-gence. [7]It has been mentioned about the novel robustification techniques for analysing and demonstrating the effects of covari-ates and response variables of artificial neural networks. [1] In this paper it has been mentioned about the brief instigation of multi-

layered perceptron and the resilient backpropagation algorithm. And an explanation has been done to explain the application of neuralnet package, which can be used for building the ANN and also while applying the RPROP algorithm. Hence an attempt has been made to compare the two algorithms, Traditional and Resilient backpropagation on a same dataset to find out the better algorithm for the given dataset.

3. Methodology

Neural net majorly centered on [8], which are very relevant on analyzing the functional relationships. The structure of an MLP is a graph corresponding to it , that means it includes vertices and edges and in terms of neural networks it known as neurons and synapses. Neurons are structured in layers and are connected with synapses. There are separate neurons associated with covariates (inputs) & response variables (output). And the layers which are between theses inputs and outputs are denoted as hidden layers because they're not directly visible. Here is an illustration of neural network with 1 hidden layer and 2 hidden neurons. This diagram/model shows the relationships between 1 covariate X and the response variable Y.

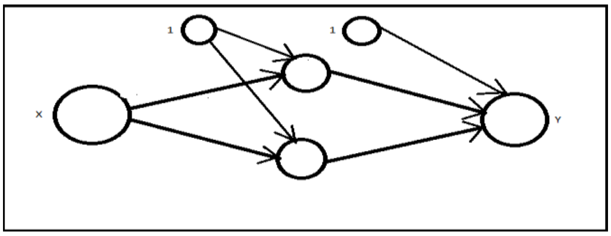

Fig. 1: Example of Neural Network with [1] Covariate and 1 Response Variable.

To every synapses there's a weight attached with it and stipulate the upshot of corressponding neuron , with the help of this all the data passes the whole neural network as signal. The signals are operated firstly by integration function , integrating all the incoming signals and then they are processed by the activation function by transforming the output of neuron.
The basic MLP(multi-layered perceptron) caries an input layer of N covariates and output layer with 1 response variable and that calculates the function:

$$o(X) = f(Wo + \sum_{i=1}^{n} WiXi) = f(Wo + W^{T}X) \quad (1)$$

Where WO = intercepts
W = (W_1,W_2,W_3,……,W_n) [vector of all synaptic weights without intercepts].
X = (X_1,X_2,X_3…….,X_n) [vector of covariates i.e inputs].
The given function is mathematically synonyms to the generalized linear model(GLM) with an associative function of f^{-1}. Hence , all the calculated weights are the regression parameters of the generalized linear model.
[9]exhibited that 1 hidden layer is adequate to model any piecewise steady function. But according to [10] to increase the computational resilience we can increase the number of hidden layers. So multi-layered perceptron with M numbers of hidden neurons in a hidden layer operates the following function

$$o(X) = f(Wo + \sum_{j=1}^{J} Wj\,.\,f(Woj + \sum_{i=1}^{n} Wij.\,Xi) \quad (2)$$

$$O\,(X) = f(Wo+\sum_{j=0}^{J} Wj.\,f(Woj + Wj^{T}X)) \quad (3)$$

Where,

W_o= intercept of output neuron.

W_{oj}= intercept of Jth hidden neuron.

W_j= (W_{1j},W_{2j},W_{3j},……,Wnj) [vector of all synaptic weight synaptic weights related to synapses which are leading to Jth hidden neuron.

X=(X_1,X_2,X_3,……,X_n) [vectors of all covariates].
Formally asserted that all hidden neurons response variables(output neurons) compute an outputs f(g(Z_o,Z_1,Z_2,Z_3,……,Z_n)) = f(g(Z)) from the output of all foregoing neurons Z_o,Z_1,Z_2,………..,Z_n , where g:R^{k+1} denotes R which resembles the integration function. And , f:R denotes R resembles the activation function. The integration function is also given as

$$G(Z)=W_oZ_o\sum_{i=1}^{k} WiZi \quad (4)$$

$$G\,(Z) = W_o+W^tZ \quad (5)$$

and the activation is given as differential , nonlinear , non descending function like logistic function ($f(u) = 1/1+e^{-u}$) . Neuralnet uses same integration function and activation function for all the neurons.

3.1. Supervised learning

Package neuralnet anchor on supervised learning algorithm. Throughout the training process neural networks are contour to the data by learning algorithms. These learning algorithms are designated by the manipulation of the given/actual output which is collated to the predicted output and all the parameters (which are basically weights) according to this comparison. All the weights are assigned to the neurons initially some random variables from the dataset through standard normal distribution. There are some steps which are followed during the iterative training procedure.
First it calculate the response variable o(X) for input X and current weights. Then it calculates iteratively until the predicted value(o) does not match with the observed value Y.

Then the error function E is the sum of squared errors. Which is given as:

$$E = 1/2\sum_{l=1}^{L}\sum_{h=1}^{H}(Olh - Ylh)^{2} \quad (6)$$

Calculate the difference among observed and predicted output.

l = 1, 2, 3, L

This error function do the listing of observations means make the pair of input and output.

h= 1, 2, 3, H denotes the output nodes.

By the help of learning algorithm all weights are adjusted.

3.2. Backpropagation algorithm

The algorithm is that we start a neural network say single layer or multi-layer we start by presenting it with a training sample, then we present the training sample, compute the updates of activation values of all the units, until we determine the activation values of the output layer units. Then compare the output we have received with the training sample outputs in order to compute the error. After having estimated the error, our object/function is to re-adjust the weights so that the error decreases, by doing so the new activation values will be closure to the real values. Having adjusted the weights for hour training sample we will again resume with another training sample and again readjust the weights.
Every time we use the proportionality factor called "alpha" which is the training rate to decide on how much we're going to adjust. And because we're using the gradient descent then we'll more likely to come down in global minimum. Then we have to contin-

ue these steps again and again in the form of iterations ideally when the results are not varying much.

3.3. Resilient backpropagation algorithm

Resilient backpropagation algorithm is similar with regular neural networks except things:
Training with Rprop is faster than the training regular backpropagation.
Rprop does not require any specific free parameter values like we have in regular backpropagation that is the learning rate α.
The Rprop is very complex to implement unlike regular backpropagation.

4. Analysis

4.1. Problem statement

We have taken a bank problem in which there's a dataset of bank customers. Which has been collected by the bank after doing some sort of surveys. The bank has been seeing the unusual churn(churn is basically defined as when people leave the company) rates. The found out that customers are leaving the bank at very high and unusual rates. So , the bank want to understand that what the particular problem is behind this high churning rates.
So that they can take measures and solve the particular problem. Bank operates in three different Europe countries that are France , Spain , Germany. So they took the sample of 250 customers and they measure everything that they knew about them. They measured the following terms:

1) Customer ID
2) Surname
3) Credit score
4) Geography
5) Gender
6) Age
7) Tenure (how long they have been to this bank)
8) Balance
9) No. of products (loan , credit card and extra)
10) Whether they have credit card or not.
11) Is active member
12) Estimated salary
13) Exited

Hence, our goal is to create a demographic segmentation model to tell the bank that which of the customers among the existing customers are at high risk of leaving the bank. So here two demographic segmentation models has been created 1st by using Rprop(resilient backpropagation) and 2nd by using backprop (traditional backpropagation) and the comparison has been done.
The neuralnet package has been used for both the algorithms that is "RPROP" and "BACKPROP" in order to illustrate the correlation between the covariates (credit score , age , tenure , balance , number of products, has credit card , is active member , estimated salary) and the response variable(Exited). As we know that output that is binary hence the activation function is designated as logistic function and the error function is adapted as cross-entropy and also the linear.output is mentioned as FALSE so that the response variable is mapped by the activation function to the interval of [0,1].

RowNumb	Customer	Surname	CreditSco	Geograph	Gender	Age	Tenure	Balance	NumOfPr	HasCrCard	IsActiveM	Estimated	Exited
1	15634602	Hargrave	619	France	Female	42	2	0	1	1	1	101348.9	1
2	15647311	Hill	608	Spain	Female	41	1	83807.86	1	0	1	112542.6	0
3	15619304	Onio	502	France	Female	42	8	159660.8	3	1	0	113931.6	1
4	15701354	Boni	699	France	Female	39	1	0	2	0	0	93826.63	0
5	15737888	Mitchell	850	Spain	Female	43	2	125510.8	1	1	1	79084.1	0
6	15574012	Chu	645	Spain	Male	44	8	113755.8	2	1	0	149756.7	1
7	15592531	Bartlett	822	France	Male	50	7	0	2	1	1	10062.8	0
8	15656148	Obinna	376	Germany	Female	29	4	115046.7	4	1	0	119346.9	1
9	15792365	He	501	France	Male	44	4	142051.1	2	0	1	74940.5	0

Fig. 2: The Above Figure Contains the Sample Dataset, Which Has Been Used for This Analysis.

Table 1: Result Matrix for Resilient Backprop

Error	133.06654008190
Reached threshold	0.00998261631
Steps	11552
Intercept.to.1layhid1	-1.51323170713
creditScore.to.1layhid1	-1.06379922870
Age.to.1layhid1	1.82475948779
Tenure.to.1layhid1	1.47399153641
Balance.to.1layhid1	0.11075939753
Numofproducts.to.1layhid1	0.51550200066
HasCrCard.to.1layhid1	-1.14474533519
IsActiveMember.to.1layhid1	0.31383581840
EstimatedSalary.to.1layhid1	-0.39462216152
Intercept.to.1layhid2	1.22227815438
CreditScore.to.1layhid2	-1.39141209154
Age.to.1layhid1	2.96461291492
Tenure.to.1layhid2	0.99587842308
Balance.to.1layhid2	-0.31576672839
NumofProduct.to.1layhid2	0.44028788576
HasCrCard.to.1layhid2	0.63295290805
IsActiveMember.to.1layhid2	1.83720117243
EstimatedSalary.to.1layhid2	1.23392972466
Intercept.to.Exited	2.18445009871
1layhid.1.to.Exited	-2.88073889180
1layhid.2.to.Exited	-3.44885703354

Table 2: Sample Results for Resilient Backdrop

Credit Score	Age	Tenure	Balance	Exited
619	42	2	0	0.194587
608	41	1	83807.86	0.360789
502	42	8	15966.80	0.231479
699	39	1	0	0.194582

4.1.1. The result analysis of resilient backpropagation

The results in table 1 and table 2 has been generated by applying the resilient backpropagation algorithm. The above result exhibit that the training process involved 11552 steps until all the absolute partial derivatives of the error function were smaller than the 0.01 (it is the default value of threshold).
The estimated weights range from -1.51 to 2.18 lets take a case where the intercepts of the first hidden layer are -1.51 & 1.22 and the weights are predominant to the first hidden neuron are estimated as (-1.063, 1.824, 1.473, 0.110, 0.515, -1.144, 0.313, -0.394) for the covariates (CreditScore , age , tenure , balance , NumofProducts , HasCrCard , IsActiveMember , EstimatedSalary) respectively.
If the Error function is identical to the negative log likelihood function , then the error refers to the likelihood as is used for example to calculate Akaike's Information Criterion (AIC) [10].
In this case, the entity Sample result is a list consisting of only one element relating to one calculated depiction . If more than one depiction were calculated, the output would be saved each in a separate list of elements. This approach is identical for all values that change with depiction apart from sample result that is retain as matrix with one column for each replication. We can analyze that person having credit score 619 is having the probability of leaving the bank as 19.99% whereas the person with credit score 608 and age of 41 is having the probability of 36..07% to leave the bank and so on. And this is how we can analyze the probabilities of all the customer. However in table 2 it is just a sample result which is being mentioned.

4.1.2. The result analysis of traditional backpropagation

The results in table 3 and table 4 has been generated by the traditional backpropagation algorithm.
By implementing traditional backpropagation the iteration steps got converges with only 82 steps.
The estimated weights range from 0.21 to -1.15 lets take a case where the intercepts of the first hidden layer are 0.21 & 0.43 and the weights cardinal to the first hidden neuron are estimated as (0.55 , -1.04, -0.10, -0.58, -0.61, -0.77, 0.50, 0.45) for the covariates (CreditScore , age , tenure , balance , NumofProducts , HasCrCard , IsActiveMember , EstimatedSalary) respectively.

So In this comparison, the demographic simulation is less satisfying than the model which was attained by the resilient backpropagation algorithm.
From this Sample Result of Traditional Backpropagation we can analyze that same customer which was having a credit score of 619 and showing the probability of 19.99% in Resilient Backpropagation, that same customer is having the probability of 20.20% in Traditional Backpropagation. Whereas the customer with 850 credit score was showing the probability of 23.07% in case of Resilient Backpropagation, is having the probability of 24.52% in case of Traditional Backpropagation.

Table 3: Result Matrix of Traditional Backdrop

Error	133.885183852993
Reached threshold	0.009556460
Steps	82
Intercept.to.1layhid1	0.212308249893
creditScore.to.1layhid1	0.555444270461
Age.to.1layhid1	-1.044120967456
Tenure.to.1layhid1	-0.107727305929
Balance.to.1layhid1	-0.582406109411
Numofproducts.to.1layhid1	-0.614607096847
HasCrCard.to.1layhid1	-0.779761734795
IsActiveMember.to.1layhid1	0.501420136088
EstimatedSalary.to.1layhid1	0.456570166795
Intercept.to.1layhid2	0.438913939263
CreditScore.to.1layhid2	0.495055572958
Age.to.1layhid1	-0.434042774069
Tenure.to.1layhid2	-0.051055478570
Balance.to.1layhid2	-0.994076490094
NumofProduct.to.1layhid2	0.452296246305
HasCrCard.to.1layhid2	-0.190883899146
IsActiveMember.to.1layhid2	-0.686645243223
EstimatedSalary.to.1layhid2	-0.379216176763
Intercept.to.Exited	-1.157165488099
1layhid.1.to.Exited	-0.109845405842
1layhid.2.to.Exited	0.694363482203

Table 4: Sample Result for Traditional Backdrop

Credit Score	Age	Tenure	Balance	Exited
619	42	2	0	0.202025
608	41	1	83807.86	0.245234
502	42	8	15966.80	0.245234
699	39	1	0	0.199928

4.1.3. The plot comparison between resilient back prop and traditional back prop

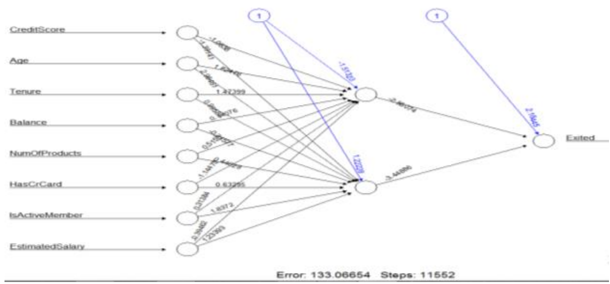

Fig. 3: The Resilient Backpropagation obtains the Above Plot of Trained Neural Network, Which Includes Synaptic Weights and the Basic Information about the Training Process.

It resembles the construction of trained neural networks that is commonly known as network topology. This figure consist of the trained synaptic weights, all intercepts as well as the basic information about the training process like the gross error and the number of steps needed to coincide. For the larger neural networks the size of the figure and also related to each neuron can be diagnosed using the parameters dimension and the radius Respectively.

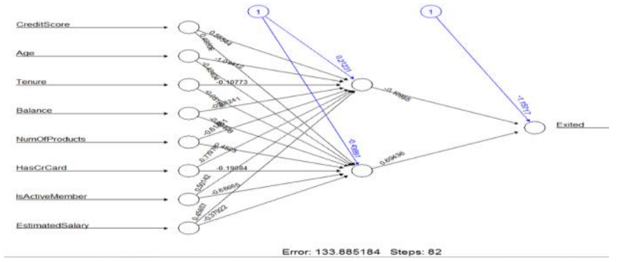

Fig. 4: the Traditional Backpropagation obtains the above Plot of Trained Neural Network Which Includes Synaptic Weights and the Basic Information about the Training Process.

5. Conclusion

This paper gave a brief review about the comparison between the two neural networks algorithm i.e Backpropagation algorithm and the Resilient backpropagation algorithm. The Resilient Backpropagation algorithm is contrive from the traditional backpropagation algorithm which refine the weights of neural network in order to find the local minima of the error function. And hence the Gradient of error function that's is dE/dW is deliberated with respect to weights in order to find roots. The traditional backpropagation may be defined as when we compute the error in output layer and backpropagate that error in to the hidden layer. So , we have some heuristic for the errors that are there in the hidden layer. In short, we can say that Resilient backpropagation and the traditional backpropagation algorithm are same except that resilient backpropagation do not require the specific free parameter values. Like learning rate (α) Unlike we use in traditional backpropagation. Training in with Resilient backpropagation is often faster than training with regular backpropagation and results of Resilient backpropagation are quite accurate when compared with Traditional backpropagation algorithm because of low error rate in comparison with Traditional Backpropagation. And the number of steps taken by Rprop to converge is more than Backprop. So the prediction is more precise.

Acknowledgement

We would like to acknowledge the help and support which was given to us by our co-author Professor Dr Addepalli VN Krishna. He has made a great contribution by supporting us.

References

[1] S. Fritsch and F.Gunther (2008), Training of neural network: R foundation for statistical computing. R package version 1.2
[2] P.McCullagh and J. Nelder *Generalized linear models chapman hall, London, 1983. https://doi.org/10.1007/978-1-4899-3244-0.*
[3] M.Riedmiller Advanced supervised learning in multi-layer perceptron from backpropagation to adaptive learning algorithms. International Journal of computer standards and interface, (1994), 16:265-278.
[4] M.Riedmiller and H. Braun(1993) A direct method for faster backpropagation learning: the rprop algorithm proceedings of the IEEE international conference on neural networks(ICNN), 1:586-591,
[5] W. Schiffmann, M. Joost, and R. Werner (1994) Optimization of the backpropagation algorithm for training multilayer perceptrons. Technical report, University of Koblenz, Institute of Physics,
[6] A. Kumar and D. Zhang (2006), Personal recognition using hand shape and texture. IEEE Transactions on Image Processing, 15:2454–2461, https://doi.org/10.1109/TIP.2006.875214.
[7] O. Intrator and N. Intrator(2001), Interpreting neuralnetwork results: a simulation study. Computational Statistics & Data Analysis, 37:373–393, https://doi.org/10.1016/S0167-9473(01)00016-0.
[8] C. Bishop (1995), Neural networks for pattern recognition, *Oxford university press, New York.*
[9] K. Hornik, M.Stichcombe, H.White (1989), Multi-layer feedforward network and universal approximators.2:359-366.
[10] Frauke Günther and Stefan Fritsch. neuralnet: Training of Neural Networks,2010 The R Journal Vol. 2/1.