Explaining Absenteeism at Workplace Predicted by a Neural Network

Harshit Trivedi

Research School of Computer Science,
The Australian National University, Canberra ACT - 2600, Australia
u6101164@anu.edu.au

Abstract. Artificial Neural Networks (ANNs) build upon the unique modelling of the human brain as a basis to develop algorithms to predict complex patterns. But, ANNs are often over-parameterized with redundant neurons resulting in unnecessary computations and memory usage [1]. This study has been performed to train a back propagation trained feed-forward neural network to predict the employees' absenteeism at workplace at a courier company during the time between July 2007 to July 2010. Further, an approach to building rules to filter underperforming neurons is introduced to increase the speed and accuracy performance of the network. Using this technique, the network performs with an accuracy of 58%, which is much better than its performance without pruning of the network. Hence, a better neural network model can be built using this technique to explain employees' absenteeism at workplace.

Keywords: artificial neural networks, absenteeism at workplace, network pruning, rules to remove under-performing neurons, ReLU

1 Introduction

This study was carried out on a dataset containing information about the employees of a courier company in Brazil and the reasons relating to the absenteeism from work between July 2007 to July 2010. This dataset and its related information has been drawn from the UCI Machine Learning Repository (http://archive.ics.uci.edu/ml/datasets/Absenteeism+at+work). The dataset has 740 rows of data and 20 distinctive features (that may/may not relate to their absence from work) collected for 36 different employees during the said time. These features range from reason for absence, age, Body Mass Index (BMI) to service time, number of children and number of pets among others. The final output is given as the number of hours an employee was absent for each instance of their absence. A sample of the raw data can be seen in the table below –

Table 1. Raw data for Absenteeism at Work (as obtained from UCI ML Repository [2]).

ID	Reason for Absence	Transportation Expense	Service Time	Age	Workload Average/day	Son	Social Drinker	…	Body Mass Index (BMI)	Absenteeism Time (in hr.)
11	26	289	13	33	239,554	2	1	…	30	4
36	0	118	18	50	239,554	1	1	…	31	0
3	23	179	18	38	239,554	0	1	…	31	2
7	7	279	14	39	239,554	2	1	…	24	4
11	23	289	13	33	239,554	2	1	…	30	2

On cursory analysis, the elementary reasons that come to mind for absenteeism at workplace can be deterioration of mental or physical health, childcare, petcare or eldercare, disengagement, family or social events, or injuries. According to a survey by Gallup-Healthways that recorded input from 94,000 workers across 14 major occupations, the total annual cost related to loss in productivity as a result of absenteeism at workplace totaled $84 billion [3]. So, explaining the causes of absenteeism can have a big impact on finding the reasons for this and then reducing these losses by addressing those reasons among employees.

So, this back-propagation neural network designed for this experiment is can be used to predict an employee's absenteeism time (in hours) given their other related parameters. This prediction can then be used to analyze the reasons for absenteeism among employees. It can provide them valuable feedback and a chance to implement corrective measures and better lifestyle for their own betterment.

2 Method

The main reason why Neural Networks are so popular these days is their ability to outperform nearly all other Machine Learning algorithms with their ability to work with incomplete information and having some amount of fault tolerance. But, Neural Networks also have some disadvantages too. These include a heavy dependency on the hardware of system and requiring parallel processing when dealing with heavy datasets or image classification tasks [4] [5]. So, in order to increase the speed and efficiency of a neural network when working with large datasets, pruning can be applied on the network to prune some features or neurons that are redundant or unhelpful in some sense to the overall output of the network [6].

Pruning a neural network is derived from the idea that not all parameters of a network contribute equally towards the output. So, pruning the least contributing parameters from the network is bound to improve its the network speed and accuracy to some extent [7]. Network Pruning, however, needs to be done carefully so as not to prune the network to the point of irreparable damage. So, network pruning should be an iterative process comprising of the following steps [7] [8]-

Step 1 - Fine-tuning the network until convergence on the target task.

Step 2 - Alternate iterations of pruning and further fine tuning.

Step 3 - Stop pruning after reaching the target trade-off between accuracy and pruning objective.

Although, this is a simple procedure, receiving a successful outcome relies on employing the appropriate pruning criterion.

There are many heuristic criteria like Minimum Weight, Activation, Mutual Information, Taylor Expansion, Relation to Optimal Brain Damage and Average Percentage of Zeros (APoZ) which can be used to identify the 'least important' parameters [7][8], where pruning by the magnitude of the kernel weights is considered one of the simplest possible criterions to work on.

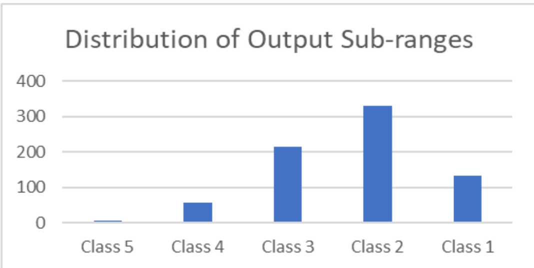
The actual pruning part of the process involves an iterative mechanism where we first evaluate the importance of each neuron, then remove the least important neuron, and then fine-tune the network and continue the pruning until the trade-off is reached [8].

The approach used by me to prune the network is derived from the study on producing a set of rules that filter the under-performing neurons depending upon the positive or negative correlation between some features and the final output of the pattern [9]. Using this approach, the effect of each input features can be classified on how it contributes to the output of that pattern. This can be easily done by taking an arithmetical mean of the contributing vector components for the cases where a characteristic pattern is found ON, and for cases where they are found OFF. The rules made by following this method can be useful in understanding the network's actions and can be used to find some helpful insight into the correlation between the network's features, and their positive and negative effects to the overall output of the network [9].

Characteristic patterns can be used to classify the output set into sub-ranges [10]. The reduction of output classes also plays a vital role in the cross validation of data, as it increases the network's probability of finding the right output class. So, I reduced the number of output classes from a possible 120 to 5. For my dataset, the 5 subranges for the values of the output neurons are –

Table 2. Classification of output into sub-ranges and distribution of patterns in each subrange.

Absenteeism Time (in hours)	Output Sub-Class	Frequency
More than 80	Class 5	6
Between 10 - 80	Class 4	57
Between 5 - 10	Class 3	216
Between 2 - 5	Class 2	329
Less than 2	Class 1	132



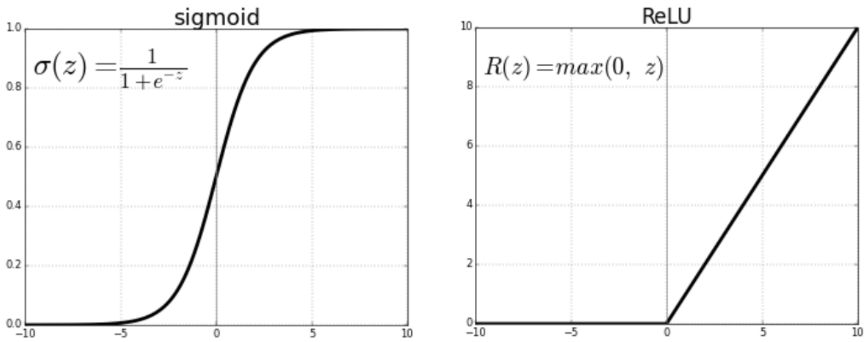
Based on the frequency distribution of each output sub-class, a set of rules can be formulated that highlight the features of the dataset which were instrumental in defining the results of that sub-class. The set of rules that can be formulated from the characteristic patterns to prune the under-performing input neurons are -

Table 3. Rule sets for the characteristic patterns observed in the dataset

Characteristic Pattern	Rule Set
ON 80+	(Social Drinker ≥ 0.66) AND (Social Smoker = 0)
ON Between 10 - 80	(BMI ≥ 25.84) AND (Height ≥ 175.05)
ON Between 5 - 10	(Age ≥ 35.95) AND (Service Time ≥ 11.87)
ON Between 2 - 5	(Transportation Expense ≥ 205) AND (Distance from Residence to Work ≥ 30.68)
ON Below 2	(Workload Average/day ≥ 278,444.96) AND (Age ≥ 38.22)

Besides this, a close study of the data revealed that a change in the activation function from Sigmoid function to ReLU (Rectified Linear Unit) function which has the benefits of sparsity and reduced likelihood of vanishing gradients would benefit the training of the network [10]. The constant gradient of ReLU results in faster learning, which was observed in the improved results.

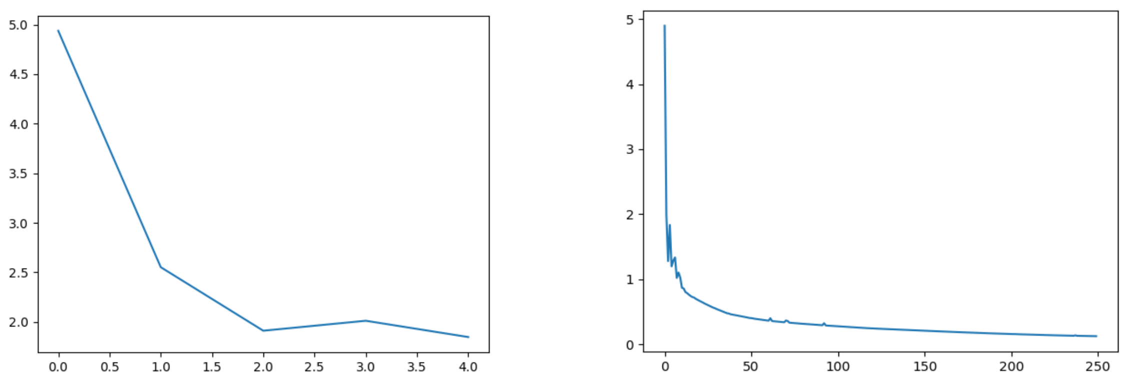
Figure 1. Output graphs for Logistic Sigmoid function v/s ReLU function



3 Results

The network showed an accuracy of 34% initially with a loss of 0.41 on the test data. After implementing the 5 sub-ranges for the output classes, the accuracy of the neural network jumped to 58% with a loss of just 0.15. This is a considerable improvement on the performances of the network and confirms my initial hypothesis.

Figure 2. Graphs depicting reduction in losses on the training of the dataset before and after implementing the output sub-ranges and adjusting the number of epochs



Similarly, the cross validation done by adjusting the number of epochs for the network training also positively impacted the overall accuracy of the network. Changing the activation function from logistic sigmoid function to the more prevalent ReLU function also brought about a positive change in the performance. Conversely, when I tried to slice the training dataset into two different datasets and assigned different activation functions namely ReLU and tanh functions to both, it didn't have the desired impact on the training. This is quite comprehensible, as ReLU is far more efficient (about 6 times) when converging to the local minima than tanh function. This is the reason almost all deep learning models use ReLU these days.

With the rule set developed in this study, it becomes easy to weed out the non-performing neurons in the network and prune the network to make it smaller and more efficient. For instance, it can be inferred from the last rule that employees having an average workload more than 278,445 and aged over 38 are more likely to be absent for at most 2 hours from work. Similarly, employees having transportation expenses of more than \$205 and whose residence is situated more than 30 kms from their workplace are more likely to take a leave of absence ranging from 2 – 5 hours from work. It was also observed that the employees who are aged 36 or older and have served the organization for more than 11 years tend to be absent from work for 5 – 10 hours in the said period. The employees falling under Class 4 (10 – 80 hours of absenteeism) were observed to have a BMI of 25 or more and height of over 175 cm. Although, there was no clear rule observed for employees absent for more than 80 hours, it can be said that most of them are social drinkers. These rules can be used to prune the network such that the effective patterns play an important role in the outcome of the network and the non-eminent patterns are diminished, which would subsequently also increase the speed of training the network on a large dataset.

However, the rule set devised for the study could not be implemented correctly in the experiments due to inadequate knowledge of implementation tools. So, it remains a topic for future study and development.

4 Conclusion

The cross validation of the dataset and assigning the appropriate activation functions to the network provides a good way to increase the efficiency and accuracy of the neural network. However, devising a set of rules from the characteristic features of the patterns is a much better way to make the network more efficient by pruning the under-performing neurons.

With this study, I have demonstrated how a set of rules can be defined to prune the patterns of a neural network, in order to make it more efficient and quick. This study can be very useful in determining the reasons for absenteeism at workplace, and how many hours of absenteeism is observed based on each of those reasons. Furthermore, employers can also take actions on some of these reasons to improve the productivity of their employees. For example, if we look at Class 2 and its corresponding rule, granting a travel allowance of about \$205 to the employees whose residence is situated more than 30 kms from their workplace would cover their transportation expense, and would bring down the absenteeism in this category. This would be very beneficial for productivity as the frequency of absenteeism in this class is the highest.

This method of pruning is not dependent upon the size or architecture of the network and is general in nature [7]. As the method couldn't be correctly implemented in the experiments, it will be a topic of my future research.

References

1. R. H. Campbell, M. Babaeizadeh and P. Smaragdis, "NoiseOut: A Simple way to Prune Neural Networks," University of Illinois at Urbana-Champaign, Urbana-Champaign, 2016.
2. University of California - Irvine , "UCI Machine Learning Repository - Absenteeism at Work Dataset," Center for Machine Learning and Intelligent Systems, 05 April 2018. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Absenteeism+at+work.
3. Investopedia, "The Causes And Costs of Absenteeism In Workplace," Forbes, 10 July 2013. [Online]. Available: https://www.forbes.com/sites/investopedia/2013/07/10/the-causes-and-costs-of-absenteeism-in-the-workplace/#266c38d73eb6
4. C. Louizos, M. Welling and D. P. Kingma, "Learning Sparse Neural Networks through L_0 Regularization," Cornell University Library, New York, 2017.
5. L. Theis, I. Korshunova, A. Tejani and F. Huszar, "Faster gaze prediction with dense networks and Fisher pruning," Cornell University Library, New York, 2018.
6. F. Huszar, "Pruning Neural Networks: 2 Recent Papers," inFERENCe, 6 February 2018. [Online]. Available: http://www.inference.vc/pruning-neural-networks-two-recent-papers/. [Accessed 22 May 2018].
7. Y. Le Cun, J. S. Denker and S. A. Solla, "Optimal Brain Damage," AT&T Bell Laboratories, Holmdel, New Jersey, 1990.
8. P. Molchanov, S. Tyree, T. Karras, T. Aila and J. Kautz, "Pruning Convolution Neural Networks for Resource Efficient Inference," in *ICLR*, 2017.

9. T. D. Gedeon and H. S. Turner, "Explaining Student Grades Predicted by a Neural Network," in *International Joint Conference on Neural Networks*, 1993.
10. T. Sainath, G. E. Hinton and G. E. Dahl, "Improving Deep Neural Networks for LVCSR Using Rectified Linear Units and Dropout," IBM T. J. Watson Research Center, Toronto.