Explainable Artificial Intelligence Model: Analysis of Neural Network Parameters

Sandip Kumar Pal, Amol A. Bhave, and Kingshuk Banerjee

1 Introduction

In recent years, there has been growing interest of extracting patterns from data using artificial neural network (ANN)-based modelling techniques. The use of these models in the real-life scenarios is becoming primary focus area across different industries and data analytics practitioners. It is already established that the ANN-based models provide a flexible framework to build the models with increased predictive performance for the large and complex data. But unfortunately, due to high degree of complexity of ANN models, the interpretability of the results can be significantly reduced, and it has been named as "black box" in this community. For example, in banking system to detect the fraud or a robo-advisor for securities consulting or for opening a new account in compliance with the KYC method, there are no mechanisms in place which make the results understandable. The risk with this type of complex computing machines is that customers or bank employees are left with a series of questions after a consultancy or decision which the banks themselves cannot answer: "Why did you recommend this share?", "Why was this person rejected as a customer?", "How does the machine classify this transaction as terror financing or money laundering?". Naturally, industries are more and more focusing on the transparency and understanding of AI when deploying artificial intelligence and complex learning systems.

S. K. Pal (✉) · A. A. Bhave
Cognitive Business and Decision Support, IBM India, Bengaluru, India
e-mail: sandipkumar.pal@gmail.com

A. A. Bhave
e-mail: amobhave@in.ibm.com

K. Banerjee
Cognitive Computing and Analytics, IBM Global Business Services, Bengaluru, India
e-mail: kingshukb@in.ibm.com

A. K. Laha (ed.), *Applied Advanced Analytics*, Springer Proceedings in Business and Economics, https://doi.org/10.1007/978-981-33-6656-5_4

Probably, this has opened a new direction of research works to develop various approaches to understand the model behaviour and the explainability of the model structure. Recently, Joel et al. (2018) has developed explainable neural network model based on additive index models to learn interpretable network connectivity. But it is not still enough to understand the significance of the features used in the model and the model is well specified or not.

In this article, we will express the neural network (NN) model as nonlinear regression model and use statistical measures to interpret the model parameters and the model specification based on certain assumptions. We will consider only multilayer perceptron (MLP) networks which is a very flexible class of statistical procedures. We have arranged this article as: (a) explain the structure of MLP as feed-forward neural network in terms of nonlinear regression model, (b) the estimation of the parameters, (c) properties of parameters and their asymptotic distribution, (d) simulation study and conclusion.

2 Transparent Neural Network Model (TRANN)

In this article, we have considered the MLP structure given in Fig. 1. Each neural network can be expressed as a function of explaining variable $X = [x_1, x_2, \ldots, x_p]$ and the network weights $\omega = (\gamma', \beta', b')$ where α' is the weights between input and hidden layers, β' is the weights between hidden and output layers and b' is the bias of the network. This network is having the following functional form

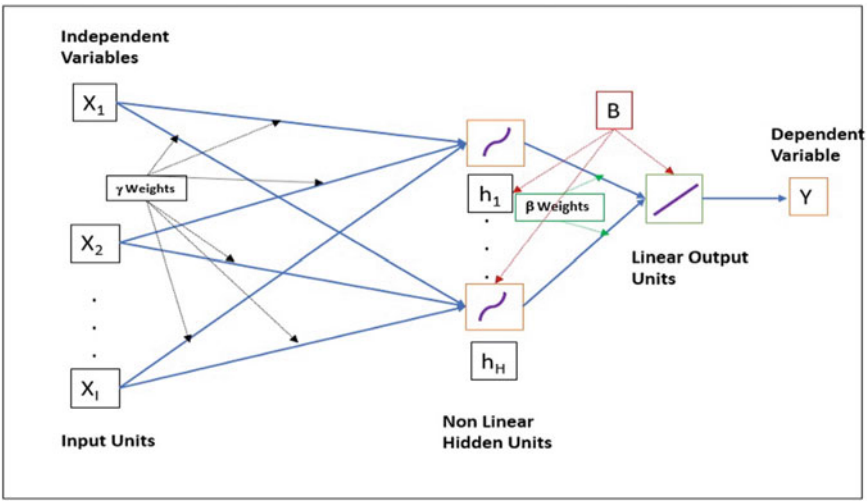

Fig. 1 A multilayer perceptron neural network: MLP network with three layers

$$F(X,\omega) = \sum_{h=1}^{H} \beta_h g(\sum_{i=1}^{I} \gamma_{hi} x_i + b_h) + b_{00} \tag{1}$$

where the scalars I and H denote the number of input and hidden layers of the network and g is a nonlinear transfer function. The transfer function g can be considered as either logistic function or the hyperbolic tangent function. In this paper, we have considered logistic transfer function for all the calculation. Let us assume that Y is dependent variable and we can write Y as a nonlinear regression form

$$Y = F(X,\omega) + \epsilon \tag{2}$$

where ϵ is $i.i.d$ normal distribution with $E[\epsilon] = 0$, $\mathrm{E}[\epsilon\epsilon'] = \sigma I$. Now, Eq. (2) can be interpreted as parametric nonlinear regression of Y on X. So based on the given data, we will be able to estimate all the network parameters. Now the most important question is what would be the right architecture of the network, how we can identify the number of hidden units in the network and how to measure the importance of those parameters. The aim is always to identify an optimum network with small number of hidden units which can well approximate the unknown function (Sarle 1995). Therefore, it is important to derive a methodology not only to select an appropriate network but also to explain the network well for a given problem.

In the network literature, available and pursued approaches are $regularization$, *stopped-training* and $pruning$ (Reed 1993). In $regularization$ methods, we can minimize the network error (e.g. sum of error square) along with a penalty term to choose the network weights. In the *stopped-training* data set, the training data set split into training and validation data set. The training algorithm is stopped when the model errors in the validation set begin to grow during the training of the network, basically stopping the estimation when the model is overparameterized or overfitted. It may not be seen as sensible estimates of the parameters as the growing validation error would be an indication to reduce the network complexity. In the $pruning$ method, the network parameters are chosen based on the "significant" contribution to the overall network performance. However, the "significance" is not judged by based on any theoretical construct but more like a measure of a factor of importance.

The main issue with $regularization$, *stopped-training* and $pruning$ is that they are highly judgemental in nature which makes the model building process difficult to reconstruct. In transparent neural network (TRANN), we are going to explain the statistical construct of the parameters' estimation and their properties through which we explain the statistical importance of the network weights and will address well the model misspecification problem. In the next section, we will describe the statistical concept to estimate the network parameters and their properties. We have done a simulation study to justify our claim.

3 TraNN Parameter Estimation

In general, the estimation of parameters of a nonlinear regression model cannot be determined analytically and needs to apply the numerical procedures to find the optima of the nonlinear functions. This is a standard problem in numerical mathematics. In order to estimate the parameters, we minimized squared error, $SE = \sum_{t=1}^{T}(Y_t - F(X_t, \omega))^2$, and applied backpropagation method to estimate the parameters. Backpropagation is the most widely used algorithm for supervised learning with multi-layered feed-forward networks. The repeated application of chain rule has been used to compute the influence of each weight in the network with respect to an error function SE in the backpropagation algorithm (Rumelhart et al. 1986) as:

$$\frac{\partial SE}{\partial \omega_{ij}} = \frac{\partial SE}{\partial s_i}\frac{\partial s_i}{\partial \text{net}_i}\frac{\partial \text{net}_i}{\partial \omega_{ij}} \tag{3}$$

where ω_{ij} is the weight from neuron j to neuron i, s_i is the output, and net_i is the weighted sum of the inputs of neuron i. Once the partial derivatives of each weight are known, then minimizing the error function can be achieved by performing

$$\breve{\omega}_{t+1} = \breve{\omega}_t - \eta_t[-\nabla F(X_t, \breve{\omega}_t)]'[Y_t - F(X, \breve{\omega}_t)], t = 1, 2, \ldots, T \tag{4}$$

Based on the assumptions of the nonlinear regression model (2) and under some regularity conditions for F, it can be proven (White 1989) that the parameter estimator $\hat{\omega}$ is consistent with asymptotic normal distribution. White ((White, 1989)) had shown that the parameter estimator an asymptotically equivalent estimator can be obtained from the backpropagation estimator using Eq. (4) when η_t is proportional to t^{-1} as

$$\hat{\omega}_{t+1} = \breve{\omega}_t + \left[\sum_{t=1}^{T} \nabla F(X_t, \breve{\omega}_t)' \nabla F(X_t, \breve{\omega}_t)\right]^{-1} \times \sum_{t=1}^{T} \nabla F(X_t, \breve{\omega}_t)'[Y_t - F(X, \breve{\omega}_t)], t = 1, 2, \ldots, T \tag{5}$$

In that case, the usual hypothesis test like Wald test or the LM test for nonlinear models can be applied. Neural network belongs to the class of misspecified models as it does not map to the unknown function exactly but approximates. The application of asymptotic standard test is still valid as the misspecification can be taken care through covariance matrix calculation of the parameters (White 1994). The estimated parameters $\hat{\omega}$ are normally distributed with mean ω^* and covariance matrix $\frac{1}{T}C$. The parameter vector ω^* can be considered as best projection of the misspecified model onto the true model which lead to:

$$\sqrt{T}(\hat{\omega} - \omega^*) \sim N(0, C) \tag{6}$$

where the T denotes the number of observations. As per the theory of misspecified model (Anders 2002), the covariance matrix can be calculated as

$$\frac{1}{T} = A^{-1} B A^{-1} \tag{7}$$

where the matrix A and B can be expressed as $A \equiv E[\nabla^2 SE_t]$ and $B \equiv E[\nabla SE_t \nabla SE_t^{'}]$. SE_t denotes the squared error contribution of the tth observations, and ∇ is the gradient with respect to the weights.

4 TRANN Model Parameter Test for Significance

The hypothesis tests for significance of the parameters are an instrument for any statistical models. In TRANN, we are finding and eliminating redundant inputs from the feed-forward single layered network through statistical test of significance. This will help to understand the network well and will be able to explain to network connection with mathematical evidence. This will help to provide a transparency to the model as well. The case of irrelevant hidden units occurs when identical optimal network performance can be achieved with fewer hidden units. For any regression method, the value of t-statistic plays an important role for hypothesis testing whereas it is overlooked in neural networks. The non-significant parameters can be removed from the network, and the network can be uniquely defined (White 1989). This is valid for linear regression as well as neural networks. Here, we estimate the t-statistic as

$$\frac{\hat{\omega}_k - \omega_{H_0}(k)}{\hat{\sigma}_k} \tag{8}$$

where $\omega_{H_0}(k)$ denotes the value or the restrictions to be tested under null hypothesis H_0. The $\hat{\sigma}_k$ is the estimated standard deviation of the estimated parameter $\hat{\omega}_k$. Later, we have estimated the variance–covariance matrix $\hat{C}$ where the diagonal elements are ω_k and the $\hat{C}$ can be estimated as

$$\frac{1}{T}\hat{C} = \hat{A}^{-1} \hat{B} \hat{A}^{-1} \tag{9}$$

$$\hat{A}^{-1} = \frac{1}{T} \sum_{t=1}^{T} \frac{\partial^2 SE_t}{\partial \hat{\omega} \partial \hat{\omega}'} \quad \text{and} \quad \hat{B}^{-1} = \sum_{t=1}^{T} \hat{\epsilon}_t^2 \left(\frac{\partial F(t, \hat{\omega})}{\partial \hat{\omega}}\right)\left(\frac{\partial F(t, \hat{\omega})}{\partial \hat{\omega}}\right)' \tag{10}$$

where $\hat{\epsilon}_t^2$ is the square of estimated error for tth sample.

Equation (6) implies that asymptotic distribution of the network parameters is normally distributed and it is possible to perform the test of significance of each parameter using the estimated covariance matrix $\hat{C}$. Then, both Wald test and LM test are applicable as per the theory of misspecified model (Anders 2002).

5 Simulation Study

We have performed a simulation study to establish the estimation methods and hypothesis test of significance with a 8-2-1 feed-forward network where we have considered eight input variables, one hidden layer with two hidden units and one output layer. Therefore, as per the structure of Eq. (1), the network model contains 21 parameters and we have set the parameter values as $b' = (b_{00} : 0.91, b_1 : -0.276, b_2 : 0.276)$
$\beta' = (\beta_1 : 0.942, \beta_2 : 0.284)$
$\gamma' = (\gamma_{11} = -1.8567, \gamma_{21} = -0.0185, \gamma_{31} = -0.135), \gamma_{41} = 0.743, \gamma_{51} = 0.954, \gamma_{61} = 1.38, \gamma_{71} = 1.67, \gamma_{81} = 0.512, \gamma_{12} = 1.8567, \gamma_{22} = 0.0185, \gamma_{32} = 0.135, \gamma_{42} = -0.743, \gamma_{52} = -0.954, \gamma_{62} = -1.38, \gamma_{72} = -1.67, \gamma_{82} = -0.512)$
and the error term ϵ is generated from normal distribution with mean zero and standard deviation 0.001. In the model, the independent variables $X = [x_1, ..., x_8]$ are drawn from exponential distribution. We have generated 100,000 samples using the above parameters, and then we have taken multiple sets of 5000 random sample of observations out of 100,000 observations and derived the estimates of the parameters and confidence intervals. We are calling this method as bootstrap method. The estimated values of the parameters, standard errors, confidence interval, t-values and p-values through bootstrapping method are given in Table 1. The results based on the asymptotic properties of the estimates are given in Table 2 based on Eq. (9). Both the methods are establishing the test of significance of parameters under null hypothesis $H_0 : \omega = 0$.

Table 1 Results using bootstrapping method

Coefficients	Estimates	Std. error	t value	95% C.I.	Pr[> |t|]
b_{00}	0.917	1.089E−03	842.057	[0.916 , 0.920]	< 0.001
b_1	−0.283	4.613E−03	−61.348	[−0.289 , −0.272]	< 0.001
b_2	0.300	1.312E−02	22.866	[0.284 , 0.327]	< 0.001
β_1	0.936	1.433E−03	653.175	[0.934 , 0.938]	< 0.001
β_2	0.279	1.086E−03	256.906	[0.278 , 0.281]	< 0.001
γ_{11}	−1.854	7.246E−03	−255.865	[−1.861, −1.833]	< 0.001
γ_{21}	−0.025	3.732E−03	−6.699	[−0.031, −0.017]	< 0.001
γ_{31}	−0.142	3.050E−03	−46.557	[−0.147 , −0.137]	< 0.001
γ_{41}	0.736	3.217E−03	228.785	[0.731 , 0.741]	< 0.001
γ_{51}	0.947	4.142E−03	228.634	[0.941 , 0.952]	< 0.001
γ_{61}	1.373	4.771E−03	287.78	[1.365 , 1.380]	< 0.001
γ_{71}	2.133	3.488E−02	61.153	[2.075 , 2.182]	< 0.001
γ_{81}	0.019	3.294E−03	5.768	[0.015 , 0.025]	< 0.001
γ_{12}	1.873	2.402E−02	77.977	[1.816 , 1.910]	< 0.001
γ_{22}	0.042	1.401E−02	2.998	[0.025 , 0.070]	< 0.001
γ_{32}	0.160	1.233E−02	12.976	[0.143 , 0.185]	< 0.001
γ_{42}	−0.730	1.081E−02	−67.53	[−0.746 , -0.705]	< 0.001
γ_{52}	−0.941	1.347E−02	−69.859	[−0.959 , −0.911]	< 0.001
γ_{62}	−1.375	1.544E−02	−89.054	[−1.398 , −1.341]	< 0.001
γ_{72}	−2.039	1.333E−01	−15.296	[−2.197 , −1.796]	< 0.001
γ_{82}	−0.065	1.170E−02	−5.556	[−0.08 , −0.042]	< 0.001

Table 2 Results using asymptotic properties

Coefficients	Estimates	Std. error	t value	Pr[$>\|t\|$]
b_{00}	0.917	1.646E−04	5573.017	<0.001
b_1	−0.283	3.715E−04	−761.616	<0.001
b_2	0.305	1.356E−03	224.886	<0.001
β_1	0.936	1.657E−04	5649.898	<0.001
β_2	0.279	1.646E−04	1695.213	<0.001
γ_{11}	−1.855	7.034E−05	−26371.921	<0.001
γ_{21}	−0.025	4.436E−05	−574.123	<0.001
γ_{31}	−0.142	3.441E−05	−4125.615	<0.001
γ_{41}	0.732	2.823E−05	25,930.712	<0.001
γ_{51}	0.943	4.694E−05	20,096.352	<0.001
γ_{61}	1.373	7.827E−05	17,541.977	<0.001
γ_{71}	2.151	9.378E−05	22,930.994	<0.001
γ_{81}	0.021	1.652E−05	1300.632	<0.001
γ_{12}	1.885	2.446E−04	7704.932	<0.001
γ_{22}	0.042	1.637E−04	255.681	<0.001
γ_{32}	0.161	1.273E−04	1266.425	<0.001
γ_{42}	−0.717	1.077E−04	−6654.568	<0.001
γ_{52}	−0.932	1.749E−04	−5325.529	<0.001
γ_{62}	−1.377	2.837E−04	−4855.127	<0.001
γ_{72}	−2.103	3.496E−04	−6016.115	<0.001
γ_{82}	−0.075	5.922E−05	−1265.38	<0.001

6 Conclusion

Neural networks are a very flexible class of assumptions about the structural form of the unknown function F. In this paper, we have used nonlinear regression technique to explain the network through statistical analysis. The statistical procedures usable for model building in neural networks are significance test of parameters through which an optimal network architecture can be established. In our opinion, the transparent neural network is a major requirement to perform a diagnosis of neural network architecture which not only approximates the unknown function but also explains the network features well through the statistical nonlinear modelling assumptions. As a next step, we would like to investigate more on the deep neural networks based on the similar concepts.

Acknowledgements We use this opportunity to express our gratitude to everyone who supported us in this work. We are thankful for their intellectual guidance, invaluable constructive criticism and friendly advice during this project work. We are sincerely grateful to them for sharing their truthful and illuminating views on a number of issues related to the project. We express our warm thanks to our colleagues Koushik Khan and Sachin Verma for their support to write code in Python

and R. We would also like to thank Prof. Debasis Kundu from IIT Kanpur who provided the valuable references and suggestions for this work.

References

Anders, U. (2002). Statistical model building for neural networks. In *963 Statistical Model Building for Neural Networks*.

Joel, V., et al. (2018). Explainable neural networks based on additive index model. arXiv.

Reed, R. (1993). Pruning algorithms—A survey. *IEEE Transactions on Neural Networks*, *4*, 740–747.

Rumelhart, D. E., et al. (1986). A direct adaptive method for faster backpropagation learning-the rprop algorithm. Parallel distributed Processing.

Sarle, W. S. (1995). Stopped training and other remedies for overfitting. In *Proceedings of the 27th Symposium on the Interface*.

White, H. (1994). *Estimation, inference and specification analysis*. Cambridge University Press.

White, H. (1989). Learning in neural networks: A statistical perspective. *Neural Computation*, *1*, 425–464.