International Journal of Statistical Analysis

Research Article Open Access

A Neural-Network Based Technique for Calculating Multi-Dimensional Integrals of an Arbitrary Function

Lee KC*

Department of Systems and Naval Mechatronic Engineering, National Cheng-Kung University, Tainan 70101, Taiwan

***Corresponding Author:** Kun-Chou Lee, Department of Systems and Naval Mechatronic Engineering, National Cheng-Kung University, Tainan 70101, Taiwan, E-mail: kclee@mail.ncku.edu.tw

Citation: Lee KC (2020) A Neural-Network Based Technique for Calculating Multi-Dimensional Integrals of an Arbitrary Function Project Zone International Journal of Statistical Analysis. V1(1): 1-3.

Received Date: Jan 10, 2020 **Accepted Date**: Feb 02, 2020 **Published Date**: Feb 05, 2020

1. Abstract

The neural network is one of the most important techniques in machine learning, which belong to artificial intelligence. In this paper, the architecture of a neural network is suitably extended to calculate multi-dimensional integrals of an arbitrary function. The final results are obtained without any numerical integration. The neural network is inherently a machine learning technique and thus serves as a black-box transformation. Our technique can treat not only deterministic but also random integrand. Only a small number of sampling points for the integrand are required to give accurate integration results.

2. Keywords: Neural-Network; Machine learning; Integration

3. Introduction

Recently, the machine learning [1] has attracted interest of researchers in different fields all over the world. The machine learning belongs to the scope of artificial intelligence. It teaches a computer to predict the response of a system by learning from experiences. The goal is to build an intelligent system. A machine learning technique is basically a black box, which can achieve both identification and regression. The term "black box" means that the relation between the input and output of a system is very complex. The neural network [2] is one of the most important techniques in machine learning. It is basically a black box that accepts certain input and produces certain output. The neural network is usually utilized to model the relation between the input and output of a complicated system. Recently, it has been applied to many fields of engineering, e.g., Electromagnetics [2]. For example, we have utilized neural networks as nonlinear models to treat different electromagnetic problems [3-5]. In [6], the architecture of a neural network is extended to calculate the derivative of the system output. The success of [6] then motivates us to develop an alternative extension of the neural network to compute the integral of the system output. The neural network utilized in this study is the RBF-NN (Radial Basis Function Neural Network) [2]. The goal is to estimate N-dimensional (N is a positive integer) integrals of an arbitrary function. Initially, the relation between the integrand (i.e., the function to be integrated) and its variables is modeled by an RBF-NN. This RBF-NN has one node in the output layer to represent the integrand and N nodes in the input layer to represent the integral variables. There still exist some nodes and Gaussian bases in the hidden layer for nonlinear mapping. After the RBF-NN is trained, the weights within the neural network are determined. The neural-network output becomes the linear combination of different Gaussian bases together with their weights. The original integration is then transformed into the linear combination of integrals on different Gaussian bases. The final results can be found from look-up tables of mathematical handbooks without any numerical integration. Numerical simulation shows that the results calculated by our neural-network based technique are consistent with those using ISML commercial subroutines. Due to the inherent black-box properties of neural networks, the proposed method has some benefits. It can treat not only deterministic but also random integrand. Only a small number of sampling points for the integrand are required to train the neural network, and then give accurate integration results. In other words, one does not need to know the overall characteristic of the integrand. These benefits will make the proposed method especially useful as the integrand is entirely obtained from practical measurement.

4. Formulations

Consider the N-dimensional integrals of an arbitrary function as

$$answer = \int_{a_N}^{b_N}\cdots\int_{a_2}^{b_2}\int_{a_1}^{b_1} func(\bar{x})\,dx_1\,dx_2\cdots dx_N \quad (1)$$

In (1), the integrand $func(\bar{x})$ is an arbitrary function and $\bar{x}$ = $(x_1, x_2, \cdots, x_N)$ contains components to represent the integral variables. Initially, an RBF-NN is utilized to model the nonlinear mapping of $\bar{x} \rightarrow func(\bar{x})$. The architecture of the RBF-NN is shown in Figure 1

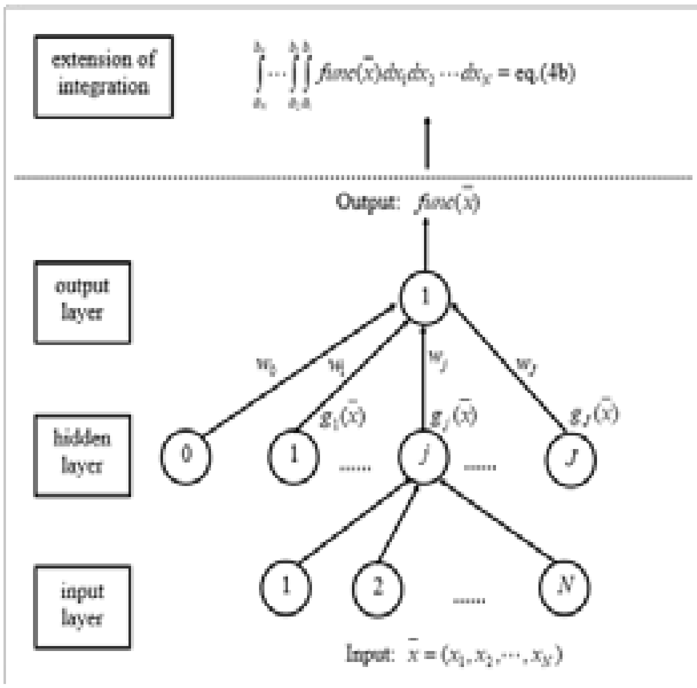

Figure 1: The RBF-NN architecture and its extension of integration.

There are N nodes in the input layer to represent the integral variables x_1, x_2, …, x_3. There is one node in the output layer to represent the integrand $func(\bar{x})$. There still exist J nodes in the hidden layer for nonlinear mapping. According to [2], the output of the RBF-NN can be expressed as

$$func(\bar{x}) = w_0 + \sum_{j=1}^{J} w_j g_j(\bar{x}), \quad (2)$$

where w_j and $g_j(\cdot)$ represent the weight and nonlinear transformation function of the j-th node in the hidden layer, respectively. In general, $g_j(\cdot)$ is given as [2]

$$g_j(\bar{x}) = e^{-\frac{\left|\bar{x}-\bar{v}_j\right|^2}{2\sigma^2}}, \quad (3)$$

where $\bar{v}_j$ is the mean corresponding to the j-th hidden node and σ^2 is the auto-covariance of the Gaussian function. The RBF-NN is trained by some learning samples of $\bar{x} \rightarrow func(\bar{x})$. The detailed training procedures are given in [2]. After the neural network is trained, all the weights w_j, j = 0, 1, …, J are determined. By substituting (2) and (3) into (1), we have

(4a)

$$answer = w_0\int_{a_N}^{b_N}\cdots\int_{a_2}^{b_2}\int_{a_1}^{b_1} 1\cdot dx_1\,dx_2\cdots dx_N + \sum_{j=1}^{J} w_j \int_{a_N}^{b_N}\cdots\int_{a_2}^{b_2}\int_{a_1}^{b_1} g_j(\bar{x})\cdot dx_1\,dx_2\cdots dx_N$$

$$= w_0\prod_{n=1}^{N}(b_n - a_n) + \frac{\pi^2\sigma^4}{4}\sum_{j=1}^{J} w_j \left\{\prod_{n=1}^{N}\left[erf\left(\frac{b_n - v_{j,n}}{\sqrt{2}\sigma}\right) - erf\left(\frac{a_n - v_{j,n}}{\sqrt{2}\sigma}\right)\right]\right\}.$$

where $erf(\cdot)$ is the error function and its value can be looked up from most mathematical handbooks. Therefore, the N-dimensional integrals in (1) is replaced by (4b) without any numerical integration. The final results of 4(b) can be obtained from look-up tables. In other words, the RBF-NN is extended to calculate multi-dimensional integrals in (1), as shown in Figure 1 (the part above the horizontal dotted line). It should be noted that the training work of the neural network is performed only once, i.e., during the mapping of $\bar{x} \rightarrow func(\bar{x})$. In general, the t**raining d**ata of $\bar{x} \rightarrow func(\bar{x})$ are easy to obtain.

5. Simulation

Without loss of generality, the simulation of three-dimensional integration is considered, i.e., N=3. The integrand in (1) is assumed to be

$$func(\bar{x}) = e^{-\sqrt{x_1^2+x_2^2+x_3^2}}\cdot\cos^2\left(\sqrt{x_1^2+x_2^2+x_3^2}\right) \quad (5)$$

For simplicity, the integral limits in (1) are assumed to be $a_2 = a_2 = a_3$ = -d and $b_1 = b_2 = b_3$ = d. There are 15 nodes in the hidden layer of the RBF-NN, i.e., J=15. The auto-covariance of the Gaussian basis is chosen to be $\sigma^2 = 0.5$. The neural network is trained by sampled data in intervals of $-d \le x_1, x_2, x_3 \le d$. In the sampling process, the x_n (n=1, 2, 3) is randomly sampled once for every 0.1 unit length. Following the above analyses, the calculated results of integration (answer) with respect to the integral limit d are shown in Figure 2. For comparison, the results of numerical integration calculated by IMSL (Fortran) commercial subroutines are also given. It shows that they are in good agreement. In Figure 2.

there are some ripples on the resultant curve of our neural-network based technique. This phenomenon is reasonable because the analytical integrand in (5) is replaced by a black-box neural network in the proposed technique. It should be noted that the proposed integration technique is still useful even though the

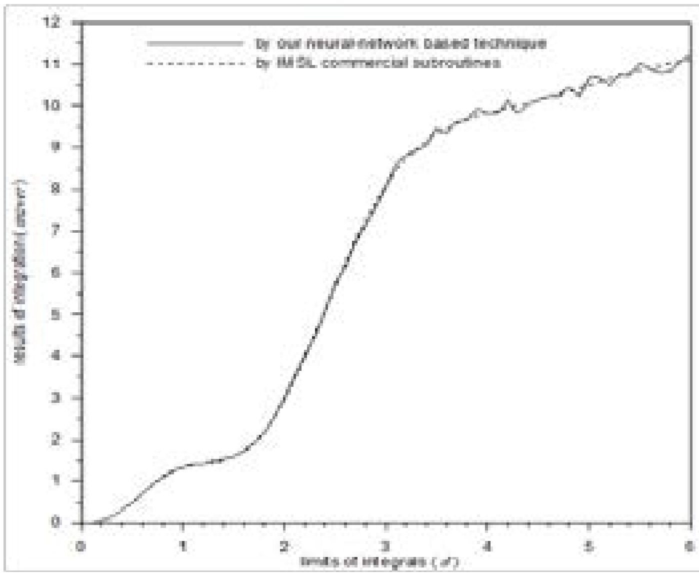

Figure 2: Calculated results of integration (answer) with respect to the integral limit (d) by our neural-network based technique and by IMSL commercial subroutines.

integrand is not deterministic. For example, if (5) is changed as

$$func(\bar{x}) = [e^{-\sqrt{x_1^2+x_2^2+x_3^2}} \cdot \cos^2(\sqrt{x_1^2+x_2^2+x_3^2})] \cdot (1+rand) \quad (6)$$

where $rand$ is a random variable with the probability uniformly distributed in the interval of [-0.05, 0.05]. The rand in (6) means there may exist $\pm 5\%$ random error with respect to (5). This random component often represents the uncertainty in practical measurement. In such a situation, procedures of the proposed integration technique are not affected at all since the neural network is inherently a black box. Following the same procedures as the previous example, the mean for the final integration result is almost the same as Figure 2 and is not shown again. In fact, no mathematical model is required for the integrand in our neural-network integration technique. The integrand may be entirely composed of discrete data from practical measurement. This is because the neural network requires only a small number of sampled data to serve as learning samples. The reason why we use the analytical model of (5) in the simulation is that the same problem can be calculated by commercial subroutines for comparison. The above simulation is implemented using personal computer with CPU of Intel(R) Core(TM) i7-4790 3.6GHz. All the above programs are coded using Fortran programming language.

6. Conclusions

In this study, the RBF-NN is successfully extended to calculate multi-dimensional integrals of an arbitrary function. The reason why we choose the RBF-NN model is that it contains bases (i.e., Gaussian bases) that are easy to implement multi-dimensional integrals. The RBF-NN is inherently a general regression model [7] and has been successfully utilized to treat nonlinear engineering problems [3-5]. Numerical simulation shows that the integration results calculated by the proposed method are consistent with those given by IMSL commercial subroutines. Since the neural network is inherently a black box, the proposed integration technique can treat not only deterministic but also random integrand. Only a small number of sampling points for the integrand are required to train the neural network, and then give accurate integration results. One does not need to know the overall characteristic of the integrand. The proposed method is especially useful as the integrand is entirely obtained from difficult measurement or complicated mathematics [8].

References

1. Deisenroth MP, Faisal AA, Ong CS. Mathematics for Machine Learning. Cambridge University Press, Cambridge, England, 2019.

2. Christodoulous C, Georgiopoulos M. Applications of Neural Networks in Electromagnetics. Artech House, Boston, 2001.

3. Lee KC. Mutual coupling analyses of antenna arrays by neural network models with radial basis functions. J Electromagnet Wave. 2003; 17: 1217-23.

4. Lee KC. A neural network based model for the two dimensional microwave imaging of cylinders. Int J RF Microw C E. 2004; 14: 398-403.

5. Lee KC, Lin TN. Application of neural networks to analyses of nonlinearly loaded antenna arrays including mutual coupling effects. IEEE Trans Antennas Propagat. 2005; 53: 1126-32.

6. Lee KC. Application of neural network and its extension of derivative to scattering from a nonlinearly loaded antenna. IEEE Trans Antennas Propagat. 2007; 55: 990-3.

7. Specht DF. A general regression neural network. IEEE Trans Neural Networks. 1991, 2: 568-76.

8. Lee KC. Impedance calculations for elements of sonar arrays by neural network based integration. IEEE Trans Aerosp Electron Syst. 2007; 43: 1065-70.