MOST ACCURATE NON-LINEAR APPROXIMATION OF STANDARD NORMAL DISTRIBUTION INTEGRAL BASED ON ARTIFICIAL NEURAL NETWORKS

Massoud Sokouti[1], Ramin Sadeghi[1*], Saeid Pashazadeh[2], Saeed Eslami Hasan Abadi[3], Morteza Ghojazadeh[4], and Babak Sokouti[5*]

Received: August 23, 2017; Revised: October 13, 2017; Accepted: October 18, 2017

Abstract

Approximating the cumulative distribution function values of a standard normal distribution with the highest accuracies still remains a challenging task. For this purpose, the non-linear prediction formulas based on artificial neural networks are applicable to the non-linear nature of a standard normal distribution integral. In this study, a dataset consisting of almost real integral values of a standard normal distribution was prepared ranging from -5 to 10 by increments of 0.01. The dataset was used to train 16 artificial neural networks each of which was repeated 100 times to reach the best performance among them by considering the number of neurons, including 1, 2, 3, 5, 15, 25, 35, and 45. The test dataset was constructed ranging from -10 to 10 by increments of 0.001 without including the training dataset. Two different types of ANN models were considered in which their transfer functions of the hidden layers were hyperbolic tangent and those of the output layers were either hyperbolic tangent or linear (purelin). Three evaluation metrics, the mean squared error (MSE), absolute error (AE), and relative error (RE) were used to compare the results of the proposed models and another 7 accurate literature approximation formulas. The results of the predicted points against their almost real values were illustrated and their measurement metric values were calculated and compared with those of the 7 literature formulas. The highest accuracies with 8 to 9 digits of accuracy were achieved by the 2 proposed equations based on ANN models using

[1] *Nuclear Medicine Research Center, Mashhad University of Medical Sciences, Mashhad, Iran. E-mail: raminsadeghi1355@yahoo.com*
[2] *Department of Computer and Electrical Engineering, University of Tabriz, Tabriz, Iran.*
[3] *Department of Medical Informatics, Faculty of Medicine, Mashhad University of Medical Sciences, Mashhad, Iran.*
[4] *Research Center for Evidence-Based Medicine, Tabriz University of Medical Sciences, Tabriz, Iran.*
[5] *Biotechnology Research Center, Tabriz University of Medical Sciences, Tabriz, Iran. Tel: +9841 33364038; Fax: +98 41 33370420. E-mail: b.sokouti@gmail.com; sokoutib@tbzmed.ac.ir*
[*] *Corresponding author*

only 15 neurons with the measurement metrics MSE = 2.15E-17, AE = 1.03E-08, RE = 1.04E-08, point = 2.89, and MSE = 4.91E-18, AE = 4.51E-09, RE = 3.23E-06, point = -2.99 in the interval -10 to 10, respectively. In conclusion, the 2 ANN-based equations with 15 neurons were superior in terms of properties, including optimization, less absolute error, and less computational costs. However, for simple calculations, the ANN-based equation with 2 neurons using 2 hyperbolic tangent transfer functions at their hidden and output layers can also be used.

Keywords: Artificial neural network, standard normal distribution, approximation, cumulative distribution function, non-linear model

Introduction

In the realm of statistics in sciences such as engineering and natural, social, and computer sciences, the problem of approximating the values of normal distribution, one of the well-known continuous probability distributions, has attracted statistics researchers worldwide (Bowling *et al.*, 2009; Casella and Berger, 2001). The applications of the normal distribution were vastly visited in approximating the quantities as representatives of the sum of many independent processes like measurement errors (Bowling *et al.*, 2009; Lyon, 2014). The mathematical formula of the normal curve was firstly developed by De Moivre, in 1733, as a rationale for normal probability law (Johnson *et al.*, 1994; Le Cam and Grace, 2000). However, Stigler stated that De Moivre only presented a rule for approximating binomial coefficients which never included the probability of the density function (Stigler, 1986). Normal distribution is mostly known as the Gaussian distribution to support the least squares which was developed by Carl Friedrich Gauss in 1809 (Gauss, 2004). Additionally, this is known as Laplace's second law. Although Gauss suggested the normal distribution law first, Laplace presented the fundamental central limit theorem which emphasized the normal distribution in 1810 (Stigler, 1986). The normal distribution is mostly applied in hypothesis testing to check a null hypothesis. In other words, the null hypothesis test is satisfied when the overall view of plotted points almost forms a straight line.

Generally, the calculation of different types of density functions of normal distributions can be performed in the form of the probability density function (PDF) of the normal distribution, PDF of the standard normal distribution, cumulative density function (CDF) of the normal distribution, and CDF of the standard normal distribution and their formulas are highlighted in this section. In the following formula, the PDF of normal distribution is calculated in which x is a random variable with mean μ and variance σ^2, and the PDF of the normal distribution is calculated as follows:

$$f(x)=\frac{1}{\sqrt{2\pi\sigma}}e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}, \ -\infty<x<\infty, \ \mu\in R, \ \sigma>0 \quad (1)$$

For calculating the PDF of the standard normal distribution, let $\mu = 0$, $\sigma = 1$, as follows:

$$f(x)=\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}x}, \quad -\infty<x<\infty \quad (2)$$

Moreover, the CDF of the normal distribution is

$$P(X<x)=\int_{-\infty}^{x}\frac{1}{\sqrt{2\pi\sigma}}e^{-\frac{1}{2}\left(\frac{y-\mu}{\sigma}\right)^2}dy \quad (3)$$

and finally, by using Equations 2 and 3, the

$$\varphi(x)=\int_{-\infty}^{x}\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}(y)^2}dy \quad (4)$$

From a survey of the literature, there is no clear solution for computing the infinite numerical integral of the PDF (i.e., Equation 2) which resulted in $\varphi(x)$ (i.e., Equation 4). Additionally, the standard normal distribution table is commonly used for calculating these corresponding values. In this case, if the x value is not present in the table, the probability between 2 x values should be calculated, which certainly does not result in that much of an accurate value with respect to the compared existing errors. To overcome this shortcoming, several works were developed for approximating the cumulative normal distribution with less accurate results (Aludaat and Alodat, 2008; Bowling *et al*., 2009; Yun, 2009). There are 2 known approaches for approximating the cumulative normal distribution. The first approach is based on numerical algorithms focusing on high precision, whereas the second approach uses ad-hoc approximations which do not increase the precision. According to these specifications (i.e., speed and precision), the numerical algorithms include complex computations while ad-hoc approximations have low computational costs by using simple formulas. The classification of these formulas was discussed in detail by Waissi and Rossin (1996) and Bryc (2002). The formulas were divided into 4 types of approximations: series expansion, sigmoid approximation, orthogonal expansion, and ad-hoc approximation. In some cases, some exceptions can be supposed since some numerical algorithms are as simple as a pocket calculator, while some ad-hoc algorithms are so accurate but they require moderate calculations. Table 1 lists the studies carried out on improving the calculation of Equation 4 in terms of decreasing the absolute error.

The aim of this study is to propose several non-linear models based on a multilayer perceptron (MLP) neural network (with various numbers of neurons in their hidden layers) for extracting and demonstrating improved equations to calculate Equation 4 with the least absolute error. To the best of the authors' knowledge, this is the first study carried out for optimizing the calculation of Equation 4 using different MLP neural networks in terms of their numbers of neurons.

Materials and Methods

Dataset

For constructing the input and output data for training the artificial neural network in order to propose a non-linear predictive model, a range of $-5 \leq x \leq 10$ with increasing steps of 0.01 was used. However, the Matlab R2013a programming environment was used to do the calculation of Equation 4 based on the above-mentioned data range. According to the references (Hart, 1978; Greene, 1993; Andrews, 1997), Equation 4 was calculated based on the following formulas:

$$\varphi(x) \approx \mathrm{erfc}(-x/\sqrt{2})/2 \qquad (5)$$

$$\mathrm{erfc}(x) = 1 - \mathrm{erf}(x)$$

$$\text{where } \mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt, \qquad (6)$$

$$\mathrm{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2} dt$$

For calculating the values for *erfc* and *erf,* the approximation methodology in Cody (1969) was used since it had less errors and was more accurate in comparison to other approximation methods which are as below based on their dividing ranges:

$$\begin{cases} \mathrm{erf}(x) \cong x R_{lm}(x^2) & |x| \leq 0.5 \\ \mathrm{erfc}(x) \cong e^{-x^2} R_{lm}(x) & 0.46875 \leq x \leq 4 \\ \mathrm{erfc}(x) \cong \dfrac{e^{-x^2}}{x} \left\{ \dfrac{1}{\sqrt{\pi}} + \dfrac{1}{x^2} R_{lm}(1/x^2) \right\} & x \geq 4 \end{cases} \qquad (7)$$

where $R_{lm} = -100 \log_{10} \max \left| \frac{f(x) - f_{lm}(x)}{f(x)} \right|$

Table 1. Literature review of most of the important equations considering their absolute errors

Founder	Equation	Range	Max. Absolute Error
Laplace (Laplace, 1812; Zelen and Severo, 1970)	$\varphi(x) \approx 0.5 + \frac{1}{\sqrt{2\pi}}\sum_{k=0}^{N}(-1)^k \frac{x^{2k+1}}{k!\,2^k(2k+1)}$	NA	NA
Laplace (Johnson and Kotz, 1970; Zelen and Severo, 1970)	$\varphi(x)\approx 0.5+\frac{1}{\sqrt{2\pi}}e^{-z^2/2}\sum_{=k0}^{\infty}\frac{x^{2+k1}}{(2+k1)!!}$	6<x<∞	22 significant digits of above equation
McConnell (McConnell, 1990)	$\varphi(x) \approx 1-\frac{1}{\sqrt{2\pi}}e^{-z^2/2}\sum_{j=1}^{5}\frac{b_j}{(1+px)^j}$ $=p0.2316419$, $b_1=0.31938153$, $b_2-=0.356563782$, $b_3=1.781477937$, $b_4=-1.821255978$, $b_5=1.330274428$	0≥x<∞	7.5×10^{-8}
Revfeim (Revfeim, 1990)	$\varphi(x)\approx 1-1/\left(1+e^{\Sigma_{=k0}^{\infty}a_k x^{2+k1}}\right)$	-∞<x<∞	NA
Page (Page, 1977)	$\varphi(x) \approx 1-1/\left(1+e^{\Sigma_{k=0}^{\infty}a_k x^{2k+1}}\right)$, $a_1x+a_2x^3 = 1.5976x+0.070565992x^3$	0≥x<∞	1.4×10^{-4}
Waissi and Rossin (Waissi and Rossin, 1996)	$\varphi(x)\approx 1-1/\left(1+e^{\Sigma_{=k0}^{\infty}a_k x^{2+k1}}\right)$, $a_1+za_2z^3+a_3z^5=1.595208466+z0.07412366556z^3+0.0007809431668z^5$	0≥x<8	4.3×10^{-5}
Lin (Lin, 1990)	$\varphi(x)\approx 1-1/\left(1+e^{\left(\frac{4.2x\pi}{9x-}\right)}\right)$		6.8×10^{-3}
Laplace (Laplace, 1812)	$\varphi(x) = 1-\frac{1}{\sqrt{2\pi}}e^{-x^2/2}\cfrac{x}{x+\cfrac{1}{x+\cfrac{2}{x+\cfrac{3}{\dots}}}}$	0≤ x<9	10^{-4}
Bryc (Bryc, 2002)	$\varphi(x)\approx 1-\frac{+x3.333}{\sqrt{2\pi x^2+7.32+x2\times3.333}}e^{x-^2/2}$	0≥x<∞	7.1×10^{-4}
Bryc (Bryc, 2002)	$\varphi(x)=1-\frac{x^2+5.575192695+x12.77436324}{\sqrt{2\pi x^3+14.38718147x^2+31.53531977+x2\times12.77436324}}e^{x-^2/2}$	0≥x<∞	1.9×10^{-5}
Kerridge and Cook (Kerridge and Cook, 1976)	$\varphi(x) = 0.5+\sqrt{\frac{2}{\pi}}e^{-x^2/2}\sum_{n=0}^{\infty}(x/2)^{2n+1}\frac{H_{2n}\left(\frac{x}{2}\right)}{(2n+1)!}$	0≤x< ∞	NA
Strecock; Moran (Moran, 1980; Strecock, 1968)	$\varphi(x) \approx 0.5+\frac{1}{\pi}\left(\frac{x}{3\sqrt{2}}+\sum_{n=1}^{12}\frac{1}{n}e^{-n^2/9}\sin(nx\sqrt{2}/3)\right)$	NA	4.36×10^{-4}
Divgi (Divgi, 1979)	$\varphi(x) \approx \frac{1}{1+e^{-1.7x}}$, $\varphi(x) \approx \frac{1}{1+e^{-(1.526x(1+0.1034x)}}$	0≤x≤7	0.009561, 0.002097
Hart (Hart, 1957)	$\varphi(x) \approx 1-\left(\frac{1}{\sqrt{2\pi}}\frac{e^{-x^2/2}}{x+0.8e^{-0.4x}}\right)$	-∞<x<∞	4.3×10^{-3}
Hart ((Hart, 1966)	$\varphi(x) = 1-\frac{e^{-x^2/2}}{\sqrt{2\pi}x}\left(1-\frac{\sqrt{1+bx^2}/(1+ax^2)}{P_0x+\sqrt{P_0^2x^2+e^{-x^2/2}\sqrt{1+bx^2}/(1+ax^2)}}\right)$ where $a=\left(1+\sqrt{1-2\pi^2+6\pi}\right)/2\pi$, $=b2\pi a^2$, $P_0=\sqrt{\pi/2}$	0≤x<∞	5.4×10^{-5}
Bagby (Bagby, 1995(	$\varphi(x)=\frac{1}{2}+\frac{1}{2}\left(1-\frac{1}{30}\left(7e^{-x^2/2}+16e^{-x^2)2-\sqrt{2}(}+\left(7+\frac{\pi}{4}x^2\right)e^{-x^2}\right)\right)^{1/2}$	0≤x<∞	3.04×10^{-5}
Johnson and Kotz (Johnson and Kotz, 1970)	$\varphi(x)\approx 0.5+0.5\sqrt{1-e^{x-^2/2}}$	0≤x<∞	0.0277
Zelen and Severo (Zelen and Severo, 1970)	$\varphi(x)\approx 0.5+0.5\sqrt{1-e^{-2x^2/\pi}}$	0≤x<∞	0.0031
Hamaker (Hamaker, 1978)	$\varphi(x)\approx 0.5+0.5\sqrt{1-e^{(-0.806x(1-0.018x))}}$	0≤x<∞	0.1145
Lin (Lin, 1989)	$\varphi(x) \approx 1-0.5e^{-0.416x-0.717x^2}$	0≤x<∞	0.0329
Norton (Norton, 1989)	$\varphi(x)\approx 1-0.5e^{-1.2x^{0.8}}$	0≤x<∞	0.0658

Table 1. Literature review of most of the important equations considering their absolute errors (continued)

Founder	Equation	Range	Max .Absolute Error
Shore (Shore, 2005)	$\varphi(x)\approx\frac{1g+(x-)g-(x)}{2}g(x)=e^{\log-2\times e^{\frac{a}{1/s_1}\times\left(\left((1+s_1x)^{1/s_1}-1\right)+s_2x\right)}}$, $l=-0.61228883,\ s_1=-0.11105481,\ s_2=0.44334159,\ a=-6.37309208$	-∞<x<∞	6.6072×10^{-7}
Aludaat and Alodat (Aludaat and Alodat, 2008)	$\varphi(x)\approx0.5+0.5\sqrt{1-e^{-\sqrt{\frac{\pi}{8}}x^2}}$	0≤x<∞	0.00197323
Winitzki (Winitzki, 2008)	$\varphi(x)\approx\sqrt{\left[1-\frac{3}{4}e^{-(x^2)(4/\pi+0.14x^2)/(1+0.14x^2)}\right]}$	0≤x<4	0.0488
Bowling (Bowling et al., 2009)	$\varphi(x)\approx\frac{1}{1+e^{-1.702x}}$, $\varphi(x)\approx\frac{1}{1+e^{-(0.07056x^3+1.5976x}}$	0≤x<∞	0.0095, 0.00014
Yun (Yun, 2009)	$\varphi(x)\approx0.5\left(1+\tanh\left(\frac{r}{2j}\left(\frac{1}{(1-x/a)^j}-\frac{1}{(1+x/a)^j}\right)\right)\right)$	0≤x≤a, j≥2, $a=\sqrt{\frac{\pi}{2}}$	8.9×10^{-4}
Yerukala et al. (Yerukala et al., 2011)	$\varphi(x)\approx0.5-1.136\tanh(-0.2695x)+2.47\tanh(0.5416x)-3.013\tanh(0.4134x)$	-3≤x≤3	0.0013
Vazquez-Leal et al.(Vazquz-Leal et al., 2012)	$\varphi(x)\approx\frac{1}{e^{-\frac{358x}{23}+111\arctan\left(\frac{37x}{294}\right)}+1}$	-∞<x<∞	$8.2933\times\times10^{-5}$
Soranzo and Epure (Soranzo and Epure, 2012)	$\varphi(x)\approx0.5+0.5\sqrt{1-e^{-x^2\frac{17+x^2}{26.694+2x^2}}}$	0≤x<∞	$4\times\times10^{-5}$
Soranzo and Epure (Soranzo and Epure, 2012)	$\varphi(x)\approx0.5+0.5\sqrt{1-e^{\frac{-1.273547x^2-0.0743968x^4}{2+0.1480931x^2+0.000258x^4}}}$	0≤x<∞	$1.18\times\times10^{-5}$
Choudhury (Choudhury, 2014)	$\varphi(x)\approx1-\frac{1}{\sqrt{2\pi}}\times\frac{e^{-x^2/2}}{0.226+0.64+0.33\sqrt{x^2+3}}$	0<x<∞	$1.9296\times\times10^{-4}$
Olabiyi and Annamalai (Olabiyi and Annamalai, 2012a, 2012b)	$\varphi(x)\approx1-0.24015e^{-0.5616x^2}$	0≤x<∞	0.2599
Soranzo and Epure ()	$\varphi(x)\approx2^{-22^{1-41^{x/10}}}$	0≤x<∞	0.00013
Boiroju (Boiroju and Rao, 2014)	$\varphi(x)\approx\frac{1}{1+e^{-y(z)}}$, where $y(z)=-0.50644467+0.5\begin{pmatrix}0.506445+10.4467\tanh(1.3448+0.3264x)\\+9.8475\tanh(-1.3519+0.3376x)+1.5976x+0.07056992x^3\end{pmatrix}$	-5≤x≤5	$5.3\times\times10^{-5}$

The maximum relative errors for *erf(x)* and *erfc(x)* are between $6a10^{-19}$ and $3n10^{-20}$.

Moreover, a test dataset was constructed in the range of -10 $\leq x \leq$ 10 with increasing steps of 0.01and was used where 25% of the data was unseen. However, for robust evaluation of the equations, a second test dataset was also generated in the range of -10$\leq x \leq$ 10 with increasing steps of 0.001 where 100% of the data was unseen by excluding the training dataset.

Artificial Neural Network Model

A multilayer perceptron (MLP) artificial neural network (ANN) model was used for approximating the cumulative standard normal distribution values (Sokouti *et al.*, 2011) and it was implemented and run in

Matlab R2013a using the *nntool* toolbox. Considering the properties of an ANN model, a feed-forward backpropagation model with a total of 3 layers (i. e. , input, hidden, and output layers) and the total number of neurons 1, (1, 2, 3, 5, 15, 25, 35, 45), for 1 of these layers was considered, respectively. The training, adaptive learning, and performance functions were set to *trainlm*, *learngdm*, and *mse*. In this study, the ANN model was used to derive and propose a non-linear equation. Two approaches were carried out in which training and test datasets construct 75% and 25% of the data in the range of -10 to 10. In the second approach, a 100% unseen dataset was used to further evaluate the models' performance, as mentioned in the data preparation section. Our approach was carried out in 2 groups. In the first group, the *tansig* and *purelin* functions were set for the hidden and output layers, respectively; and, in the second group, the transfer functions of both layers were set to *tansig*. The formula which could be derived from the settings of the first group is as shown in Equation 8 and the one for the second group is Equation 9. In both equations, *n* is the number of neurons used in the hidden layer. For this purpose, it was essential for the unknown values for Equations 8 and 9 to be extracted using the MATLAB commands.

$$\varphi(x)=\frac{\left(W2_{1\times n}\times\left(\tanh\left(W1_{n\times1}\times\left(\frac{(x+10)}{10}-1\right)+b1_{n\times1}\right)\right)+b2\right)+1}{2} \quad (8)$$

$$\varphi(x)=\frac{\tanh\left(W2_{1\times n}\times\left(\tanh\left(W1_{n\times1}\times\left(\frac{(x+10)}{10}-1\right)+b1_{n\times1}\right)\right)+b2\right)+1}{2} \quad (9)$$

Both groups of ANN models were trained using 1, 2, 3, 5, 15, 25, 35, and 45 neurons; moreover, the number of training processes carried out for each number of neurons was performed 100 times with the maximum number of 100000 iterations (i.e., the maximum number of epochs) in order to select the highest performance model. Then, the equation for the selected model would be extracted and used for several applications by having the generalizability property.

In the next section, the approximation results based on the ANN models will be illustrated, compared, and discussed.

Evaluation Process

For the evaluation of the selected ANN-based equations, 3 measurement criteria comprising mean square error (MSE), absolute error (AE), and relative error (RE) were denoted by Equations10, 11, and 12.

$$MSE=\frac{\sum_{i=1}^{n}\left(y_{prd_i}-y_{obs_i}\right)^2}{n} \quad (10)$$

where *n* is the number of values in the working range with the step of 0.01

$$AE=MAX\left|y_{prd_i}-y_{obs_i}\right|_{i=1}^{n} \quad (11)$$

$$RE=\frac{AE}{\left|y_{obs_i}\right|} \quad (12)$$

where *i* = the maximum point of the AE.

Moreover, the approximation formulas of the literature studies with the least absolute errors were also considered for comparison purposes using the increment steps of 0.01 for their intervals. The target approximation formulas were selected from 7 studies which were those from McConnell (1990); Shore (2005); Bowling *et al.* (2009); Yerukala *et al.* (2011); Soranzo and Epure (2012); Vazquez-Leal *et al.* (2012); Boiroju and Rao (2014).

Results and Discussion

As a result, 8 equations with the least absolute errors were selected and derived for each group (i.e., a total of 16 formulas/models for both groups) according to their corresponding number of neurons. Then, the weight and bias values of the trained ANN models were retrieved in order to obtain the coefficients for Equations 8 and 9 for the first and second groups. Considering the first group, the values for the corresponding weight and bias for the ANN models by using the number of neurons at the hidden layer (i.e., 1, 2, 3, 5, 15, 25, 35, and 45) are listed as below which can be easily substituted in Equation 8:

(i) one-neuron ANN model

$W1_{1\times1}$=2.715480527669424e-02 $b1_{1\times1}$=-6.970935405902314e-02,
$W2_{1\times1}$=3.147717705629710e + 02, b2= 2.190685416357931e+01

(ii) two-neuron ANN model

$$W1_{2\times1}=\begin{pmatrix}3.118958400789535e+00\\-3.122725462709437e+00\end{pmatrix},\ b1_{2\times1}=\begin{pmatrix}-1.453891497303654e+00\\-1.453496632847387e+00\end{pmatrix},$$
$$W2^{T}_{2\times1}=\begin{pmatrix}6.517192665597936e+00\\-6.497284492506964e+00\end{pmatrix},\ b2=1.835090646691703e-02$$

(iii) Three-neuron ANN model

$$W1_{3\times1}=\begin{pmatrix}3.168190544327077e+00\\3.020368833799329e+00\\-1.406952232091600e+01\end{pmatrix},\ b1_{3\times1}=\begin{pmatrix}-1.455384028321100e+00\\1.499978451145009e+00\\-4.224083487682318e+00\end{pmatrix},\ W2^{T}_{15\times1}=\begin{pmatrix}6.305097648958258e+00\\7.450666248186394e+00\\2.165194576730191e-02\end{pmatrix},\ b2=-1.068174984538483e+00$$

(iv) Five-neuron ANN model

$$W1_{5\times1}=\begin{pmatrix}1.063882706170648e+01\\-6.877911683527201e+00\\2.113290704626592e+00\\3.686267380329537e+00\\-9.678436356617475e+00\end{pmatrix},\ b1_{5\times1}=\begin{pmatrix}-4.663178797640711e+00\\1.546146671735263e+00\\-2.932148256457400e+00\\1.427085961268888e+00\\-4.366814913835770e+00\end{pmatrix},$$
$$W2_{1\times5}=(-4.192290550236102e\text{-}01\quad -9.571451208144678e\text{-}02\quad 1.961415821616574e+02\quad 4.196531261749107e+00\quad -6.073282378158161e\text{-}01)$$
$$b2 = 1.903527548722836e + 02$$

(v) Fifteen-neuron ANN model

$$W1_{15\times1}=\begin{pmatrix}-2.099951453293109e+01\\-2.099894905183933e+01\\2.103407820598740e+01\\-2.060058363881875e+01\\1.877744096029708e+01\\-3.396871983469291e+00\\4.731874479294938e+00\\-3.342324275683668e+00\\-4.700920534686953e+00\\-3.436482999075139e+00\\1.896828976490556e+01\\2.053971999810092e+01\\2.105795699136218e+01\\2.096175909506985e+01\\-2.087428228133542e+01\end{pmatrix},\ b1_{15\times1}=\begin{pmatrix}2.100044103941327e+01\\1.800102306108889e+01\\-1.494086531220070e+01\\1.029195882333148e+01\\-7.885806196523440e+00\\1.825779033224735e+00\\-1.226664266071044e+00\\-4.275629611555986e-03\\-1.223372505821335e+00\\-1.844731564209750e+00\\7.975831031393682e+00\\1.029346985635356e+01\\1.491498429970975e+01\\1.804267110247749e+01\\-2.112541653100507e+01\end{pmatrix},\ W2^{T}_{15\times1}=\begin{pmatrix}-9.559763352548241e-02\\9.470991726411977e-02\\3.566607606073758e-01\\-9.887175720630514e-02\\8.957414270820524e-03\\-6.666919384335695e+00\\6.029675415840060e-01\\-5.649342020953396e-01\\-6.385694769055340e-01\\-6.560197692372093e+00\\8.937203059250859e-03\\1.027825998793328e-01\\1.428708433478357e-01\\9.886558087675777e-02\\-2.462140497531903e-01\end{pmatrix},\ b2=-7.677657130816705e-02$$

(vi) Twenty-five-neuron ANN model

$$W1_{25\times1}=\begin{pmatrix}3.495101687352134e+01\\-3.501265821217525e+01\\3.500637888094200e+01\\-3.489086799553467e+01\\3.501546679094547e+01\\-3.583103119203559e+01\\-3.493501149175996e+01\\-3.238190156542316e+01\\2.249692771751524e+01\\-7.145987516717788e+00\\-8.400425955772050e+00\\9.263064302879608e+00\\2.631250619740159e+00\\3.045399396614363e+00\\6.335351511757824e+00\\-1.242651959039748e+01\\2.157783563051200e+01\\3.125490490810157e+01\\-3.447218646848484e+01\\3.555884064866255e+01\\-3.574003251066216e+01\\-3.497833775654460e+01\\3.482480210924418e+01\\3.517028602790893e+01\\3.497480284989059e+01\end{pmatrix},\ b1_{25\times1}=\begin{pmatrix}-3.504876024642099e+01\\3.206968791236843e+01\\-2.915883542270238e+01\\2.638537212152881e+01\\-2.330311097464608e+01\\1.854890921017881e+01\\1.617182324318609e+01\\1.211759788368103e+01\\-6.711286733161215e+00\\1.920096048671841e+00\\1.549194833413913e+00\\-1.093267147521347e+00\\-1.579485041394357e+00\\1.575765721658392e+00\\1.398711205761712e+00\\-3.983223573119117e+00\\7.456150075313463e+00\\1.312499170285375e+01\\-1.591649370355811e+01\\1.823407218771302e+01\\-2.206225056903883e+01\\-2.627874237982053e+01\\2.936778866182593e+01\\3.189088050101411e+01\\3.502562889914148e+01\end{pmatrix},\ W2^{T}_{25\times1}=\begin{pmatrix}-1.224978336913008e-01\\8.849345852859619e-02\\-4.537735081808653e-02\\3.929165914610170e-01\\3.361763820000490e-01\\-3.197250967587279e-02\\-5.103977800420922e-03\\3.492549167071435e-04\\1.631294581291284e-04\\-9.073658192641233e-02\\-1.755919835314916e-02\\2.081135359994731e-03\\1.014748229602313e+01\\7.544288843809484e+00\\7.600930380266434e-02\\8.402094807176229e-03\\-1.639984949207469e-03\\7.028716786460763e-03\\-2.630488131061969e-02\\9.466125227087033e-02\\-1.520407838103999e+00\\-2.438886688350208e-01\\-5.490436353690448e-01\\7.927458684627193e-01\\1.462410495539894e-01\end{pmatrix},\ b_2=-1.155776631483248e-01$$

(vii) Thirty-five-neuron ANN model

$$W1_{35\times 1} = \begin{pmatrix} 4.894603684396202e+01 \\ -4.906830744692746e+01 \\ -4.900229977582411e+01 \\ -4.900127723485228e+01 \\ 4.900509679486439e+01 \\ 4.903085963876935e+01 \\ 4.901261655480346e+01 \\ 4.994196665631488e+01 \\ -4.932057697345421e+01 \\ 4.812308143164857e+01 \\ 4.593959932937125e+01 \\ -4.580070966537956e+01 \\ 4.292789592098706e+01 \\ -3.984472868760076e+01 \\ -2.408109693242951e+01 \\ 7.303307853124354e+00 \\ -8.563548754226844e+00 \\ -2.892343944654531e+00 \\ -2.882450757291813e+00 \\ -7.222463748214143e+00 \\ -9.875665197092308e+00 \\ 2.831942423159738e+01 \\ -3.936087137759864e+01 \\ 4.399531967608473e+01 \\ 4.676497620064986e+01 \\ -4.816140471435644e+01 \\ -4.945553252826797e+01 \\ 5.018296375041263e+01 \\ 4.902711800686014e+01 \\ 4.903458743532799e+01 \\ 4.904641904081962e+01 \\ -4.903339437173212e+01 \\ 4.880369020638671e+01 \\ -4.900653123151764e+01 \\ -4.898472835889896e+01 \end{pmatrix}, b1_{35\times 1} = \begin{pmatrix} -4.905438632248451e+01 \\ 4.604447301905857e+01 \\ 4.323201318864403e+01 \\ 4.035040169846835e+01 \\ -3.746352223461011e+01 \\ -3.454320899441987e+01 \\ -3.168470132132026e+01 \\ -2.689924753940786e+01 \\ 2.442002964940688e+01 \\ -2.219980303799607e+01 \\ -1.851447166705629e+01 \\ 1.975184803078586e+01 \\ -1.568536637131599e+01 \\ 1.462457423522689e+01 \\ 6.034292599517634e+00 \\ -1.709416469232908e+00 \\ 1.297158699753149e+00 \\ 1.540216209055020e+00 \\ -1.556091848673539e+00 \\ -1.427451961687515e+00 \\ -2.473974743746167e+00 \\ 7.231092537065845e+00 \\ -1.238592519599751e+01 \\ 1.511499818213857e+01 \\ 1.892436058402301e+01 \\ -2.105224733619110e+01 \\ -2.335871706178789e+01 \\ 2.589840739725394e+01 \\ 3.165886166230509e+01 \\ 3.453950823529853e+01 \\ 3.741009717512179e+01 \\ -4.031215315978130e+01 \\ 4.345263395704699e+01 \\ -4.611054466934503e+01 \\ -4.901507550210645e+01 \end{pmatrix}, W2^{T}_{35\times 1} = \begin{pmatrix} -2.862939042222665e-01 \\ -1.231054552979993e-01 \\ 1.428265513786771e-01 \\ -1.918786710742536e-02 \\ -2.754088330030437e-03 \\ -7.334819905975451e-02 \\ 2.223625366473631e-01 \\ 1.075524923071320e-01 \\ -4.119822745914029e-02 \\ 1.900491085387314e-02 \\ 3.709817567753663e-03 \\ -9.380358045509013e-03 \\ -5.639897915098600e-03 \\ -7.813286511586947e-03 \\ -4.553914032151346e-05 \\ 5.439442137614849e-02 \\ -4.643692991935546e-03 \\ -8.096733563323472e+00 \\ -8.327504685927949e+00 \\ -4.417793327985704e-02 \\ -1.386117875980643e-02 \\ 6.148298679779344e-05 \\ 2.097371319953842e-04 \\ -3.625772870086452e-04 \\ 2.075239550764812e-03 \\ -7.622262461061762e-03 \\ -2.114953422173903e-02 \\ 6.756481033924583e-02 \\ 3.568915666282725e-01 \\ 5.483015145889498e-01 \\ -1.551774438618304e-01 \\ 9.905747392603461e-01 \\ -9.224651615176318e-01 \\ -4.383023280153533e-01 \\ -2.305305273830525e-01 \end{pmatrix},$$

$$, b_2 = 2.083092216181480e-01$$

(viii) Forty-five-neuron ANN model

$$W1_{45\times 1} = \begin{pmatrix} 6.299820908825659e+01 \\ 6.300664417469459e+01 \\ 6.299988952521794e+01 \\ 6.300045892413652e+01 \\ -6.299789609013054e+01 \\ -6.300038002522746e+01 \\ 6.299997457310309e+01 \\ 6.299981078330410e+01 \\ 6.305113500261507e+01 \\ 6.441360107837251e+01 \\ 6.391218051484091e+01 \\ 6.317587012799323e+01 \\ -6.221096322749710e+01 \\ 6.113993038515120e+01 \\ 6.006026163653103e+01 \\ -5.883835024755179e+01 \\ 5.686420736226727e+01 \\ 5.318637402232434e+01 \\ -4.478183191507309e+01 \\ 3.952837469788874e+01 \\ -3.952844550166083e+01 \\ -4.835988731688460e+00 \\ -7.300499499994583e+00 \\ -2.455523120939708e+00 \\ -3.107939427566630e+00 \\ 4.996859821849774e+00 \\ -3.803577734274298e+01 \\ -5.038124271254003e+01 \\ -5.565542689739049e+01 \\ 5.796138247568506e+01 \\ -5.931998232239937e+01 \\ 6.059989445807423e+01 \\ 6.183142264134702e+01 \\ 6.217688391291404e+01 \\ -6.367366188471953e+01 \\ 6.490490613582580e+01 \\ 6.301104951340047e+01 \\ -6.299249619879775e+01 \\ -6.300235826225556e+01 \\ -6.300447184077225e+01 \\ 6.299738013146234e+01 \\ -6.299924955064373e+01 \\ 6.300309112416672e+01 \\ 6.299144611740773e+01 \\ 6.300284736607412e+01 \end{pmatrix}, b1_{45\times 1} = \begin{pmatrix} -6.300180382313137e+01 \\ -6.012933092653415e+01 \\ -5.727284748932777e+01 \\ -5.440855561175093e+01 \\ 5.154798388103821e+01 \\ 4.868132751244812e+01 \\ -4.581821319295474e+01 \\ -4.295481394072186e+01 \\ -4.000673998039649e+01 \\ -3.438803537078636e+01 \\ -3.164544632861275e+01 \\ -2.942162129397179e+01 \\ 2.740016753037419e+01 \\ -2.549654638277299e+01 \\ -2.371368657479505e+01 \\ 2.197080989100559e+01 \\ -2.001299090134468e+01 \\ -1.752402460799438e+01 \\ 1.256451521145570e+01 \\ -8.497721861929646e+00 \\ 8.497736163632856e+00 \\ 5.298977528604295e-01 \\ 1.683279730019613e+00 \\ 1.428755414193281e+00 \\ -1.775126059746317e+00 \\ 1.292788449286010e+00 \\ -9.553398255015406e+00 \\ -1.378351491063045e+01 \\ -1.642084680027054e+01 \\ 1.832628075368289e+01 \\ -2.002129402304928e+01 \\ 2.179785089273308e+01 \\ 2.374058389838297e+01 \\ 2.744647528536162e+01 \\ -3.019682489666826e+01 \\ 3.330693667686552e+01 \\ 4.007203505327105e+01 \\ -4.296498294815211e+01 \\ -4.581392314780771e+01 \\ -4.867524523329695e+01 \\ 5.154867719301232e+01 \\ -5.440992032440295e+01 \\ 5.726930146729534e+01 \\ 6.014528209239725e+01 \\ 6.299710351707639e+01 \end{pmatrix}, W2^{T}_{45\times 1} = \begin{pmatrix} -8.838650593148859e-02 \\ -3.016796975492030e-01 \\ -2.002445436275839e-02 \\ -1.126309432272745e-01 \\ -1.986393620812464e-01 \\ 2.071859925290123e-02 \\ -5.162234547809259e-02 \\ 4.645272259807390e-03 \\ 2.910022048651167e-01 \\ 6.018995815318233e-02 \\ 2.489987102186065e-02 \\ 1.229296874281943e-02 \\ -6.671327484278657e-03 \\ 3.627299433920321e-03 \\ 1.881003252312628e-03 \\ -8.983519497372880e-04 \\ 3.793258082955183e-04 \\ 1.111571167967181e-04 \\ 1.517889860655479e-05 \\ 1.377643705541364e+00 \\ 1.377634327278756e+00 \\ 1.290905623844537e-01 \\ -2.250661126804125e-02 \\ -1.025206173935034e+01 \\ -8.017633155353273e+00 \\ 4.297390164155873e-01 \\ 2.594336310399616e-05 \\ 5.790206994197330e-05 \\ 1.249803721588920e-04 \\ -2.337973447085913e-04 \\ 3.774297168531949e-04 \\ -5.316005405276355e-04 \\ -5.657346750597833e-04 \\ 2.450082277118672e-03 \\ -9.102086503433544e-03 \\ 3.249390362372376e-02 \\ 2.886116071122238e-01 \\ -2.390330303327478e-01 \\ -3.475152965636427e-02 \\ -1.841367366691523e-01 \\ 1.095335378363268e-01 \\ -2.589428265060553e-02 \\ 1.205813148015499e-01 \\ 2.666096551478816e-01 \\ -2.318849066649910e-01 \end{pmatrix},$$

$$b_2 = 8.842440121887855e-02$$

(viii) Forty-five-neuron ANN model

$$W1_{45\times1}=\begin{pmatrix}6.251920637135326e+01\\-6.296725196416787e+01\\-6.300613036822146e+01\\-6.313752867171398e+01\\6.945875353802238e+01\\-6.713002694934082e+01\\6.449850676190505e+01\\-6.234055012353785e+01\\-6.002275882305337e+01\\-5.775203808720978e+01\\-5.564019467596143e+01\\5.381208570877544e+01\\5.097863689480813e+01\\-5.104750600360681e+01\\-4.516982193921947e+01\\2.333233112079833e+01\\1.707674196040219e+01\\-1.651574233792203e+01\\-1.602836284617875e+01\\1.551546658035705e+01\\-1.498587035185923e+01\\-1.454097912613093e+01\\-1.450917735543179e+01\\1.497154519307319e+01\\-1.557306029012081e+01\\1.616105830505685e+01\\1.670962149171479e+01\\1.722756563925411e+01\\1.777929708640031e+01\\1.930728651910226e+01\\-4.502477360026503e+01\\5.052973234934350e+01\\5.299029709982894e+01\\-5.174565000272506e+01\\-5.500610272287304e+01\\-5.737566591259179e+01\\-5.945789650275933e+01\\6.200648293742452e+01\\-6.498061724458810e+01\\-6.750984841134816e+01\\-6.490060348146332e+01\\6.305720784835496e+01\\-6.301337211094374e+01\\6.307194026282598e+01\\6.284892753058401e+01\end{pmatrix},\ b1_{45\times1}=\begin{pmatrix}-6.348028996182568e+01\\6.017020701668122e+01\\5.726591041091906e+01\\5.424726926724066e+01\\-4.221479273491081e+01\\3.746658871507015e+01\\-3.358892044445777e+01\\3.047162734762460e+01\\2.767020476790678e+01\\2.514102129763418e+01\\2.286365416335795e+01\\-2.083374522728584e+01\\-1.851484105175050e+01\\1.619477783807295e+01\\1.334314231459959e+01\\-6.237121978889193e+00\\-3.813419901132540e+00\\3.014805773143980e+00\\2.326480804213734e+00\\-1.686975139123624e+00\\1.082031235876330e+00\\5.060243560524390e-01\\-4.614251840160455e-02\\6.027860013289010e-01\\-1.182240749014621e+00\\1.791018376060459e+00\\2.435056635468220e+00\\3.124577707453511e+00\\3.894227260873322e+00\\5.046787294812350e+00\\-1.321297674366736e+01\\1.592220613538732e+01\\1.785209861605638e+01\\-1.992183652233739e+01\\-2.261613033007880e+01\\-2.512234227280589e+01\\-2.771591338922249e+01\\3.082832105258233e+01\\-3.465983412913204e+01\\-3.935399206199843e+01\\-4.909074193721585e+01\\5.433875431015268e+01\\-5.725619509413212e+01\\6.006006536585354e+01\\6.315142657918010e+01\end{pmatrix},\ W2^{T}_{45\times1}=\begin{pmatrix}1.248841358358896e-09\\4.299454823728459e-10\\-1.727264801189385e-09\\7.880700836970386e-09\\-1.435183598677820e-07\\3.909162142508086e-07\\-9.950774034853025e-07\\2.065660334943432e-06\\3.722233907129872e-06\\6.189169173370438e-06\\9.070420987098902e-06\\-1.135751028961183e-05\\-1.055270942704919e-05\\-1.370734322623192e-05\\-2.489173504895097e-05\\4.045145023360564e-04\\4.809279368857999e-03\\-1.486393979807095e-02\\-3.520502623316605e-02\\6.968209862089886e-02\\-1.196219353870882e-01\\-1.771056663848668e-01\\-1.959646105923796e-01\\1.559121444781965e-01\\-1.059247559453889e-01\\6.413836855146690e-02\\3.405208082694868e-02\\1.539777595219648e-02\\5.632132232275810e-03\\1.253487949370736e-03\\-2.602603800646506e-05\\2.206616861712783e-05\\1.077113189163187e-05\\6.099651013212306e-06\\5.950268514953016e-06\\4.215390321376564e-06\\2.507364941811252e-06\\-1.251081600214034e-06\\5.573206670486175e-07\\1.917433362519798e-07\\1.364678128065421e-08\\1.178889811580595e-09\\3.240201369917928e-10\\1.019650482029663e-10\\-1.118315022407471e-10\end{pmatrix},\ b_2=1.238727250940022e-09$$

Moreover, a comparison of the obtained ANN-based equations considering their numbers of hidden layer neurons and the exact function of Equations 6 and 7 is illustrated in Figure 1.

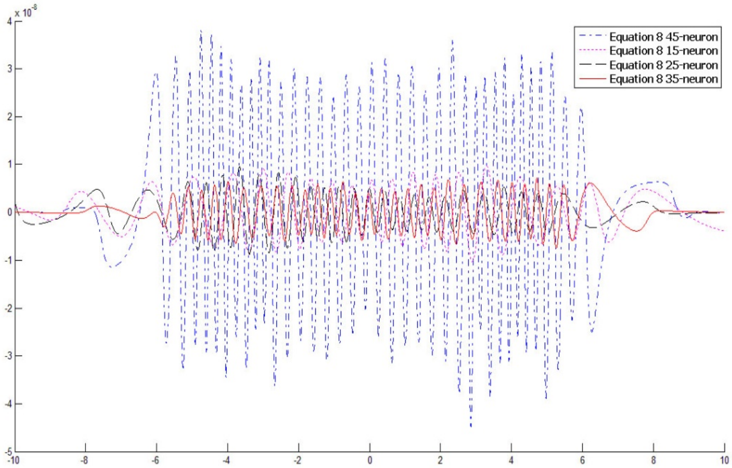

Figure 1. Comparison of Equation 8 with (a) 1 neuron, (b) 2 neurons, (c) 3 neurons, (d) 5 neurons, (e) 15 neurons, (f) 25 neurons, (g) 35 neurons, (h) 45 neurons and Equations 6 and 7

Considering the second group, the values for the corresponding weight and bias for the ANN models by using the number of neurons at the hidden layer (i.e., 1, 2, 3, 5, 15, 25, 35, and 45) are listed as below which can also be easily substituted in Equation 9:

(i) one-neuron ANN model

$W1_{1\times1}$=-8.452008183094593e+00, $b1_{1\times1}$= 2.189027205970751e-05, $W2^{T}_{5\times1}$=-1.003022781796761e+00, b_2=-1.975844687545507e-04

(ii) two-neuron ANN model

$$W1_{2\times1}=\begin{pmatrix}-6.418135485365346e+00\\ 6.323020889526632e+00\end{pmatrix},\ b1_{2\times1}=\begin{pmatrix}-2.011370885823009e-03\\ 2.116065214422967e-03\end{pmatrix},$$

$$W2^{T}_{2\times1}=\begin{pmatrix}-1.930290371262969e+01\\ -1.830434366379824e+01\end{pmatrix},\ b2=9.849211789332542e-06$$

(iii) three-neuron ANN model

$$W1_{3\times1}=\begin{pmatrix}-6.589885632031264e+00\\ -6.285905098526706e+00\\ 6.591416874435057e+00\end{pmatrix},\ b1_{3\times1}=\begin{pmatrix}1.049786186931366e+00\\ 9.751071010889175e-05\\ 1.050573101309483e+00\end{pmatrix},$$

$$W2^{T}_{3\times1}=\begin{pmatrix}2.283452873274368e-01\\ -1.456522804872633e+00\\ -2.280731924998766e-01\end{pmatrix},\ b2 = -4.152066768161014e - 06\mathrm{We}$$

(iv) five-neuron ANN model

$$W1_{5\times1}=\begin{pmatrix}-5.549629629452675e+00\\ 6.466940348204287e+00\\ -6.071210644803074e+00\\ 6.466012054436762e+00\\ -5.546999767281810e+00\end{pmatrix},\ b1_{5\times1}=\begin{pmatrix}2.691647038201131e+00\\ -9.907003656592450e-01\\ -7.241684822501341e-05\\ 9.902213100983933e-01\\ -2.692414814423433e+00\end{pmatrix},$$

$$W2^{T}_{5\times1}=\begin{pmatrix}8.141292960474813e-04\\ -2.861829630283413e-01\\ -1.574226940266170e+00\\ -2.864156050549468e-01\\ 8.094652324425476e-04\end{pmatrix},\ b_2=5.597539669242772e-08$$

(v) fifteen-neuron ANN model

$$W1_{15\times1}=\begin{pmatrix}-2.405925769573003e+01\\ -2.210750628539557e+01\\ -9.719481155198993e+00\\ 9.954399033095426e+00\\ -8.594483799998265e+00\\ 9.229804943977715e+00\\ 8.741306191823089e+00\\ -8.734267451457470e+00\\ -8.814120549225477e+00\\ 1.044707110836989e+01\\ -8.603783451846537e+00\\ 1.016397018970783e+01\\ 1.009266325189857e+01\\ 2.153823110254939e+01\\ -2.252211279865299e+01\end{pmatrix},\ b1_{15\times1}=\begin{pmatrix}1.532291485842314e+01\\ 1.218462762906055e+01\\ 2.990540078092301e+00\\ -2.540260713516699e+00\\ 1.670250251010084e+00\\ -8.654701941329216e-01\\ -3.779279239101866e-01\\ -6.860847232528856e-02\\ -5.297222621963846e-01\\ 1.159407859347714e+00\\ -1.725869242980181e+00\\ 2.690625425183765e+00\\ 3.216179931581035e+00\\ 1.241190570083128e+01\\ -1.905125110847143e+01\end{pmatrix},\ W2^{T}_{15\times1}=\begin{pmatrix}1.331253307181319e-07\\ 2.920228279738527e-07\\ 1.000979783315226e-03\\ -3.892495123207327e-03\\ 3.222135278172105e-02\\ 7.827930195833303e-02\\ 3.198658240175948e-01\\ -3.975623322499561e-01\\ -2.559397048838933e-01\\ 1.941068453055924e-02\\ 3.062611352213085e-02\\ -2.733966182074436e-03\\ -5.823860070034497e-04\\ -1.581918130589407e-07\\ -2.076106749919385e-08\end{pmatrix},\ b_2= -2.4162711355783$$

(vi) twenty five-neuron ANN model

$$W1_{25\times1}=\begin{pmatrix}3.491731358507697e+01\\3.701148402073014e+01\\-3.852049201681941e+01\\3.248820159618543e+01\\2.556490510484232e+01\\-1.151230911571230e+01\\-9.483182350673888e+00\\1.537621228952060e+01\\1.466034498823569e+01\\1.401296801299310e+01\\-1.341797581553035e+01\\9.488672565761647e+00\\9.665198580907246e+00\\-1.268863550824714e+01\\1.396387692004485e+01\\-1.488922887382066e+01\\-1.568694642037862e+01\\1.656335575409575e+01\\9.738473742099560e+00\\1.843430282128609e+01\\2.675620468646865e+01\\-3.287485196896664e+01\\3.919291753241863e+01\\3.807197973374723e+01\\3.464980774858503e+01\end{pmatrix}, b1_{25\times1}=\begin{pmatrix}-3.508178989906209e+01\\-2.968174528627151e+01\\1.858198474619686e+01\\-1.425134317607212e+01\\-1.011617755655233e+01\\3.766982441676595e+00\\2.356087846883949e+00\\-2.856129531494328e+00\\-2.219050691645519e+00\\-1.636135419811063e+00\\1.112117990902848e+00\\-3.735356981285690e-01\\2.224756345550862e-01\\-8.501179884133270e-01\\1.425478368015262e+00\\-2.032857555819264e+00\\-2.683803775766942e+00\\3.407412878918974e+00\\2.605085342907332e+00\\6.574710255114608e+00\\1.081511197336140e+01\\-1.467907528151538e+01\\1.923068642143514e+01\\2.797398839321572e+01\\3.532698988051439e+01\end{pmatrix}, W2^{T}_{25\times1}=\begin{pmatrix}6.799012417876193e-10\\-3.977349578743652e-09\\2.190337850088068e-07\\-8.539738643578402e-07\\-2.299762041020506e-06\\2.882407190449310e-04\\9.300181134917486e-03\\1.874755254354686e-03\\9.134419902695087e-03\\2.164327162241903e-02\\-2.593847027179655e-02\\4.083859508348985e-01\\4.237243747335662e-01\\-6.113006055672730e-02\\3.659821741323770e-02\\-1.854484006279905e-02\\-6.798850674344503e-03\\1.299494405693479e-03\\-5.449281810526827e-03\\-2.506844563177652e-05\\-4.660483151265039e-06\\1.522351429250283e-06\\-3.475902563919815e-07\\-2.325440557070588e-08\\-2.551008238418834e-08\end{pmatrix}, b_2=2.262178014674500e-08$$

(vii) thirty five-neuron ANN model

$$W1_{35\times1}=\begin{pmatrix}-4.909676539703168e+01\\-4.890534877087131e+01\\5.149692643571975e+01\\5.322757366448631e+01\\-4.838806669279031e+01\\-4.313537881479173e+01\\-3.796848536981272e+01\\3.323887970665783e+01\\2.757483934212110e+01\\1.275464551953557e+01\\1.639299155390428e+01\\1.604230057301373e+01\\-1.563142742490947e+01\\1.507528963260691e+01\\-1.432884906658316e+01\\-1.290520364017659e+01\\1.293436952842840e+01\\-1.348580508914150e+01\\1.415786063623743e+01\\-1.480925100485531e+01\\-1.538274396494530e+01\\1.587432316896475e+01\\-1.627464395016685e+01\\1.656174899059143e+01\\-1.673802374147956e+01\\1.361124056171664e+01\\2.990080110164914e+01\\-3.447779692780378e+01\\-3.811158903112487e+01\\-4.280328400308675e+01\\4.786044972783644e+01\\-5.253267502215736e+01\\-5.124352524292047e+01\\-4.892605751128640e+01\\4.908096853592451e+01\end{pmatrix}, b1_{35\times1}=\begin{pmatrix}4.890087721027262e+01\\4.621432443999160e+01\\-4.017663690463689e+01\\-2.952361513462488e+01\\2.488539073447668e+01\\2.066759731812871e+01\\1.692268506679432e+01\\-1.371307419440914e+01\\-1.042828236563116e+01\\-4.090167643574963e+00\\-3.935309186789562e+00\\-3.274447099714844e+00\\2.646390789320564e+00\\-2.037890320487771e+00\\1.447456170482213e+00\\8.089515118206784e-01\\-2.827504867326262e-01\\-2.156103882938890e-01\\7.266303371832777e-01\\-1.261574311840839e+00\\-1.821110559260861e+00\\2.404990334468343e+00\\-3.010702953182356e+00\\3.633075255674909e+00\\-4.270477712063262e+00\\4.591027985631411e+00\\1.173936954312582e+01\\-1.473593015825151e+01\\-1.762072436025166e+01\\-2.135001514180100e+01\\2.577029404534148e+01\\-3.071208000741853e+01\\-4.050785772436588e+01\\-4.618708610281051e+01\\4.891513356551904e+01\end{pmatrix}, W2^{T}_{35\times1}=\begin{pmatrix}-1.727638129284520e-10\\1.854508796671710e-10\\5.952177462464403e-09\\-1.298269771359952e-07\\4.497266751328284e-07\\1.174182993745439e-06\\2.638730414269710e-06\\-4.959555997209649e-06\\-7.727418343547418e-06\\-4.546466180607047e-04\\1.189877425803286e-03\\5.404989614125410e-03\\-1.503401738966334e-02\\3.299172430698340e-02\\-6.217230052948015e-02\\-1.744201517665080e-01\\2.145433602428661e-01\\-1.831071093500976e-01\\1.326857621337295e-01\\-8.628065969769419e-02\\-5.059428079418625e-02\\2.605610569760441e-02\\-1.139413457439836e-02\\3.951046159697602e-03\\-8.693376225601990e-04\\-2.136131886282420e-04\\-4.237338575018372e-06\\2.987991932653093e-06\\1.509780585669798e-06\\5.794178363451960e-07\\-1.785982650973138e-07\\3.269774202907536e-08\\-1.988902624595084e-09\\9.769774796867984e-11\\8.378335893844551e-11\end{pmatrix}, b_2=4.285508266870725e-11$$

(viii) forty five-neuron ANN model

$$W1_{45\times1}=\begin{pmatrix}6.251920637135326e+01\\-6.296725196416787e+01\\-6.300613036822146e+01\\-6.313752867171398e+01\\6.945875353802238e+01\\-6.713002694934082e+01\\6.449850676190505e+01\\-6.234055012353785e+01\\-6.002275882305337e+01\\-5.775203808720978e+01\\-5.564019467596143e+01\\5.381208570877544e+01\\5.097863689480813e+01\\-5.104750600360681e+01\\-4.516982193921947e+01\\2.333233112079833e+01\\1.707674196040219e+01\\-1.651574233792203e+01\\-1.602836284617875e+01\\1.551546658035705e+01\\-1.498587035185923e+01\\-1.454097912613093e+01\\-1.450917735543179e+01\\1.497154519307319e+01\\-1.557306029012081e+01\\1.616105830505685e+01\\1.670962149171479e+01\\1.722756563925411e+01\\1.777929708640031e+01\\1.930728651910226e+01\\-4.502477360026503e+01\\5.052973234934350e+01\\5.299029709982894e+01\\-5.174565000272506e+01\\-5.500610272287304e+01\\-5.737566591259179e+01\\-5.945789650275933e+01\\6.200648293742452e+01\\-6.498061724458810e+01\\-6.750984841134816e+01\\-6.490060348146332e+01\\6.305720784835496e+01\\-6.301337211094374e+01\\6.307194026282598e+01\\6.284892753058401e+01\end{pmatrix},\ b1_{45\times1}=\begin{pmatrix}-6.348028996182568e+01\\6.017020701668122e+01\\5.726591041091906e+01\\5.424726926724066e+01\\-4.221479273491081e+01\\3.746658871507015e+01\\-3.358892044445777e+01\\3.047162734762460e+01\\2.767020476790678e+01\\2.514102129763418e+01\\2.286365416335795e+01\\-2.083374522728584e+01\\-1.851484105175050e+01\\1.619477783807295e+01\\1.334314231459959e+01\\-6.237121978889193e+00\\-3.813419901132540e+00\\3.014805773143980e+00\\2.326480804213734e+00\\-1.686975139123624e+00\\1.082031235876330e+00\\5.060243560524390e-01\\-4.614251840160455e-02\\6.027860013289010e-01\\-1.182240749014621e+00\\1.791018376060459e+00\\2.435056635468220e+00\\3.124577707453511e+00\\3.894227260873322e+00\\5.046787294812350e+00\\-1.321297674366736e+01\\1.592220613538732e+01\\1.785209861605638e+01\\-1.992183652233739e+01\\-2.261613033007880e+01\\-2.512234227280589e+01\\-2.771591338922249e+01\\3.082832105258233e+01\\-3.465983412913204e+01\\-3.935399206199843e+01\\-4.909074193721585e+01\\5.433875431015268e+01\\-5.725619509413212e+01\\6.006006536585354e+01\\6.315142657918010e+01\end{pmatrix},\ W2^{T}_{45\times1}=\begin{pmatrix}1.248841358358896e-09\\4.299454823728459e-10\\-1.727264801189385e-09\\7.880700836970386e-09\\-1.435183598677820e-07\\3.909162142508086e-07\\-9.950774034853025e-07\\2.065660334943432e-06\\3.722233907129872e-06\\6.189169173370438e-06\\9.070420987098902e-06\\-1.135751028961183e-05\\-1.055270942704919e-05\\-1.370734322623192e-05\\-2.489173504895097e-05\\4.045145023360564e-04\\4.809279368857999e-03\\-1.486393979807095e-02\\-3.520502623316605e-02\\6.968209862089886e-02\\-1.196219353870882e-01\\-1.771056663848668e-01\\-1.959646105923796e-01\\1.559121444781965e-01\\-1.059247559453889e-01\\6.413836855146690e-02\\3.405208082694868e-02\\1.539777595219648e-02\\5.632132232275810e-03\\1.253487949370736e-03\\-2.602603800646506e-05\\2.206616861712783e-05\\1.077113189163187e-05\\6.099651013212306e-06\\5.950268514953016e-06\\4.215390321376564e-06\\2.507364941811252e-06\\-1.251081600214034e-06\\5.573206670486175e-07\\1.917433362519798e-07\\1.364678128065421e-08\\1.178889811580595e-09\\3.240201369917928e-10\\1.019650482029663e-10\\-1.118315022407471e-10\end{pmatrix},\ b_2=1.238727250940022e-09$$

Moreover, a comparison of the obtained ANN-based equations considering their numbers of hidden layer neurons and the exact function of Equations 6 and 7 is illustrated in Figure 2.

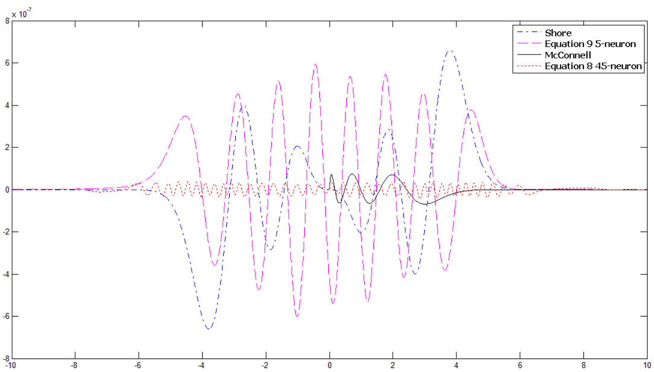

Figure 2. Comparison of Equation 9 with (a) 1 neuron, (b) 2 neurons, (c) 3 neurons, (d) 5 neurons, (e) 15 neurons, (f) 25 neurons, (g) 35 neurons, (h) 45 neurons and Equations 6 and 7

Next, the results obtained from the proposed models/formulas and other outstanding studies in terms of three evaluation metrics including Mean Square Error (MSE), Absolute Error (AE), and Relative Error (RE) will be compared and discussed. These are summed up in Table 2 which consists of the results of approximation formulas of studies presented by Bowling (Bowling *et al.*, 2009), Yerukala (Yerukala *et al.*, 2011), Vazquez (Leal *et al.*, 2012), Kumar (Boiroju and Rao, 2014), Soranzo (Soranzo and Epure 2012), Shore (Shore, 2005), and McConnell (McConnell, 1990). Moreover, the results of the proposed formulas in two groups (i. e., based on equations 8 and 9) using 1, 2, 3 ,5, 15, 25, 35, and 45 neurons in the hidden layer are included. In Table 2, Bowling presented an approximation expression with MSE =1.57E-05, AE = 9.49E-03 and RE = 1.33E-02 at point 0. 57 which works in the interval $0 \leq x \leq 10$. A better approximation model in comparison to the Bowling's (Bowling *et al.*, 2009) is the one based on equation 9 with one neuron in the interval $-\infty \leq x \leq +\infty$. The performance of this model achieves MSE = 1. 55E-05, AE = 9. 46E-03 and RE = 9.66E-03 at point 2.04 in the interval $0 \leq x \leq 10$ and MSE = 1.57E-05, AE = 9.52E-03 and RE = 4.60E-01 at point -2.04 in the interval $-10 \leq x \leq 10$. Another approximation model is based on equation 8 with one neuron in the interval $-\infty \leq x \leq +\infty$. This formula's performance achieves MSE = 1. 38E-05, AE = 8.84E-03 and RE = 9.04E-03 at point 0. 57 in the interval $0 \leq x \leq 10$ and MSE =1.39E-05, AE = 8.95E-03 and RE = 3.15E-02 at point -0.57 in the interval $-10 \leq x \leq 10$. The next model which is proposed based on equation 8 with two neurons in the interval $-\infty \leq x \leq +\infty$. The measurement metrics for this formula are MSE = 1. 81E-06, AE = 2.92E-03 and RE =3.05E-03 at point 1.73 in the interval $-10 \leq x \leq 10$. The next one is Yerukala's approach which is an ANN based model with three neurons at its hidden layer in the interval $-3 \leq x \leq 3$. For keeping up with

Table 2. Comparisons of the proposed ANN models based on equations 8 and 9, and other seven outstanding formulas in terms of three measurement metrics including MSE, AE, RE.

Methods	MSE	AE	RE	Max. AE observed at Point	Range
Bowling (Bowling *et al.*, 2009)	1.57E-05	9.49E-03	1.33E-02	0.57	$0 \leq x \leq 10$
Equation 8 1-neuron	1.39E-05	8.95E-03	3.15E-02	-0.57	$-10 \leq x \leq 10$
Equation 9 1-neuron	1.57E-05	9.52E-03	4.60E-01	-2.04	$-10 \leq x \leq 10$
Equation 8 2-neuron	1.81E-06	2.92E-03	3.05E-03	1.73	$-10 \leq x \leq 10$
Yerukala (3-neuron) (Yerukala *et al.*, 2011)	8.68E-07	1.25E-03	3.02E-01	-2.64	$-3 \leq x \leq 3$
Equation 8 3-neuron	3.07E-09	1.05E-04	3.13E-04	-.43	$-10 \leq x \leq 10$
Vazquez (Leal *et al.*, 2012)	5.25E-10	8.29E-05	5.47E-04	-1.03	$-10 \leq x \leq 10$
Kumar(2-neuron) (Boiroju and Rao, 2014)	2.89E-10	5.30E-05	2.64E-04	-0.84	$-5 \leq x \leq 5$
Soranzo(Soranzo and Epure 2012)	2.58E-11	1.13E-05	1.30E-05	1.13	$0 \leq x \leq 10$
Equation 8 5-neuron	1.27E-11	8.28E-06	7.93E-04	-2.31	$-10 \leq x \leq 10$
Equation 9 2-neuron	7.23E-12	6.97E-06	1.90E-05	-0.34	$-10 \leq x \leq 10$
Equation 9 3-neuron	3.92E-12	5.89E-06	5.89E-06	3.51	$-10 \leq x \leq 10$
Shore (Shore, 2005)	4.74E-14	6.61E-07	8.77E-03	-3.79	$-10 \leq x \leq 10$
Equation 9 5-neuron	5.70E-14	6.02E-07	3.91E-06	-1.02	$-10 \leq x \leq 10$
McConnell (McConnell, 1990)	9.54E-16	7.47E-08	9.77E-08	0.72	$0 \leq x \leq 10$
Equation 8 45-neuron	2.99E-16	4.50E-08	4.50E-08	2.86	$-10 \leq x \leq 10$
Equation 8 15-neuron	2.15E-17	1.03E-08	1.04E-08	2.89	$-10 \leq x \leq 10$
Equation 8 25-neuron	1.17E-17	9.57E-09	7.59E-05	-3.66	$-10 \leq x \leq 10$
Equation 8 35-neuron	1.05E-17	7.48E-09	7.48E-09	5.26	$-10 \leq x \leq 10$
Equation 9 15-neuron	4.91E-18	4.51E-09	3.23E-06	-2.99	$-10 \leq x \leq 10$
Equation 9 25-neuron	1.00E-18	2.57E-09	4.18E-09	0.29	$-10 \leq x \leq 10$
Equation 9 35-neuron	1.34E-18	3.10E-09	6.67E-08	-1.68	$-10 \leq x \leq 10$
Equation 9 45-neuron	3.70E-19	1.91E-09	2.25E-05	-3.76	$-10 \leq x \leq 10$

this model, our proposed ANN model with three neurons based on equation 8 for the interval $-\infty \leq x \leq +\infty$ has achieved MSE =3.07E-09, AE = 1.05E-04 and RE = 3.13E-04 at point -0.43 in the interval $-10 \leq x \leq 10$. In Figure 3, the error plots of Bowling, equation 9 with one neuron, equation 8 with one neuron, equation 8 with two neurons, Yerukala (Yerukala *et al.* , 2011) , and equation 8 with three neurons are illustrated. Surely, not all of the methods comply with the range of $-10\leq x \leq10$.

After the abovementioned methods, Vazquez's formula (Leal *et al.* , 2012) achieves better results than our three neuron ANN model based on *tanh* and *purelin* transfer functions which works in the interval $-\infty \leq x \leq +\infty$ and the results are extracted for the interval $-10 \leq x \leq 10$. The next better ANN model with tanh and standard logistic transfer functions at hidden and output layers which used two neurons in its hidden layer was proposed by Kumar (Boiroju and Rao, 2014) in the interval $-5 \leq x \leq +5$. After that,

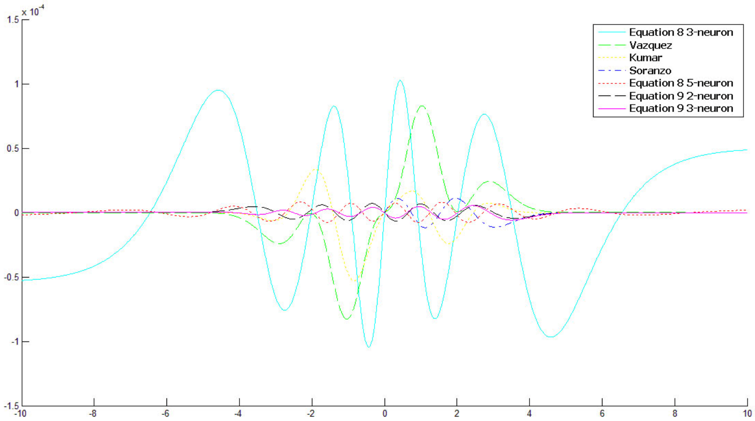

Figure 3. Error comparisons of Bowling, Equation 9 with 1 neuron, Equation 8 with 1 neuron, Equation 8 with 2 neurons, Yerukala, and Equation 8 with 3 neurons

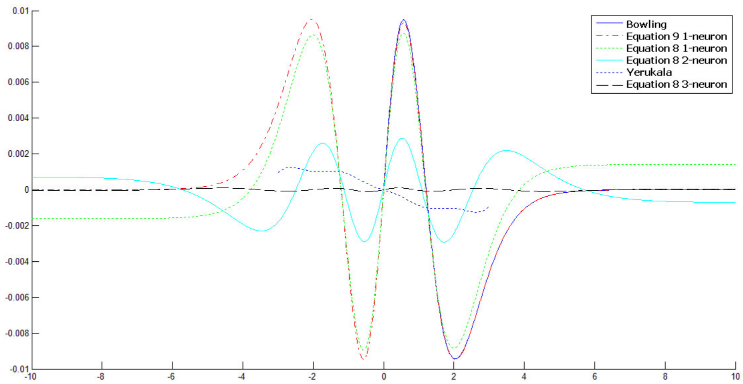

Figure 4. Error comparisons of Equation 8 with 3 neurons, Vazquez-Leal, Boiroju, Soranzo, Equation 8 with 5 neurons, Equation 9 with 2 neurons, and Equation 8 with 3 neurons

the next approximation equation presented by Soranzo (Soranzo and Epure 2012) in the interval $-\infty \leq x \leq +\infty$ was evaluated in the interval $-10 \leq x \leq 10$. Our next proposed ANN models based on equation 8 with five, and based on equation 9 with two and three neurons stand better than Soranzo's which work in the interval $-\infty \leq x \leq +\infty$ and are evaluated in the interval $-10 \leq x \leq 10$.

In Figure 4, the error plots of ANN models based on equation 8 with three and five neurons and based on equation 9 with two and three neurons, Vazquez (Leal *et al.*, 2012), Kumar (Boiroju and Rao, 2014), Soranzo (Soranzo and Epure 2012) are illustrated.

Three other formulas which stand next are related to Shore's (Shore, 2005), our proposed ANN model based on equation 9 with five neurons, and McConnell's (McConnell, 1990). However, the ranges of them are $-\infty \leq x \leq +\infty$, $-\infty \leq x \leq +\infty$, and $0 \leq x \leq +\infty$, respectively and the evaluation intervals for them are as $-10 \leq x \leq 10$, $-10 \leq x \leq 10$, $0 \leq x \leq 10$. After the McConnell' s, the ANN model based on Equation 8 with 45 neurons stands first by using the interval $-10 \leq x \leq 10$. In Figure 5, the error plots related to these four methods are illustrated in the interval $-10 \leq x \leq 10$, however, McConnell's only supports $0 \leq x \leq 10$.

Up to this point, our proposed ANN model based on Equation 8 with 45 neurons reaches the best performance in comparison to other existing formulas. From now on, other ANN models which are different in terms of their neurons and transfer functions will be evaluated in order to propose the optimum ANN model for predicting the equation 4. As it can be deduced from Table 2 and Figure 6, the ANN models based on

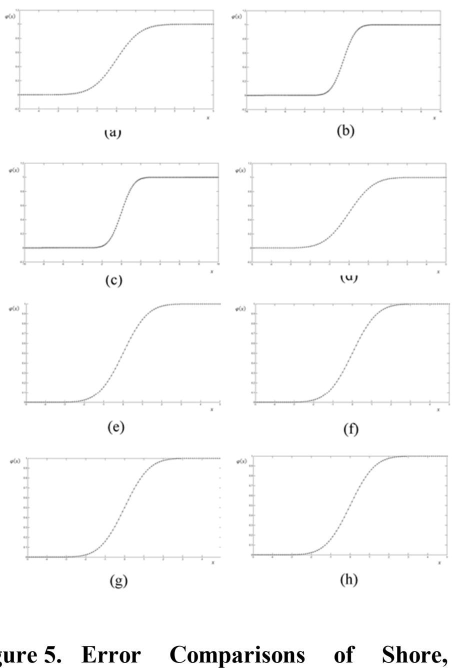

Figure 5. Error Comparisons of Shore, Equation 9 with 5 neurons, McConnell, and Equation 8 with 45 neurons in intervals $-10 \leq x \leq 10$

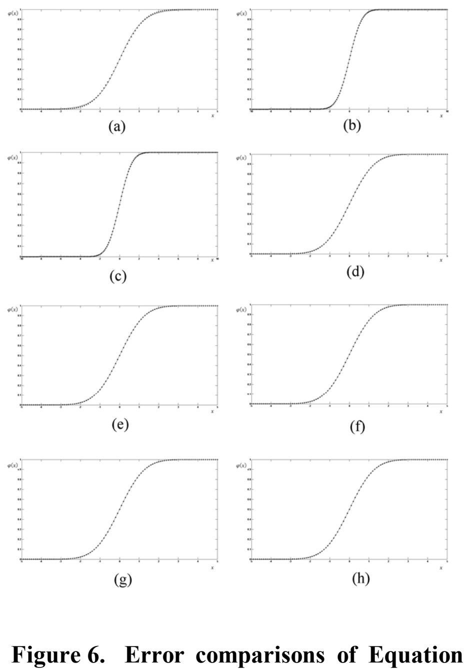

Figure 6. Error comparisons of Equation 8 with 45 neurons, Equation 8 with 15 neurons, Equation 8 with 25 neurons, and Equation 8 with 35 neurons in intervals $-10 \leq x \leq 10$

equation 8 with 45, 15, 25, and 35 neurons are ranked orderly considering their absolute error values. So, if the computational costs are so important for us, the ANN model based on Equation 8 with 15 neurons will be more than enough for the purpose of both optimization and having less absolute error values (i.e., MSE = 2.15E-17, AE = 1.03E-08 and RE = 1.04E-08 at point 2.89 in the interval $-10 \le x \le 10$). On the other hand, from Table 2 and Figure 7, it can found out that the ANN models based on Equation 9 with 15, 25, 35, and 45 neurons are ranked orderly considering their absolute error values. Again, taking in to account that less number of computational commands, optimization and less MSE and AE are important, the ANN model based on Equation 8 with 15 neurons can selected which has achieved MSE = 4.91E-18, AE = 4.51E-09 and RE = 3.23E-06 at point -2.99 in the interval $-10 \le x \le 10$.

To further evaluate the extracted formulas based on Equations 8 and 9 from ANN model, the results for a 100% unseen dataset with increment step of 0.001 are demonstrated in Table 3. The outcomes of the second unseen test dataset are illustrative of the fact that the measurement metrics values for these equations are almost the same as their corresponding equations from Table 2. Hence, the generalizability of the equations is satisfied. Moreover, during the experiment, additional hidden layers did not improve the performance of the equations; however, using more hidden layers will increase the computational costs as well.

Last but not the least, when an equation for a best performance ANN model is extracted, there is no need for training the ANN model for several times again, and hence the equation can be used for future researches and its reproducibility results will be guaranteed.

Conclusions

In this study, we have proposed 16 ANN models based on derived equations from these non-linear black boxes for predicting the values of the cumulative distribution function of standard normal distribution. The (tanh, tanh) and (tanh, purelin) transfer functions are used in their hidden layers and output layers, respectively. The proposed models, especially with 15 neurons on both types of ANN models, showed superior performance in comparison to the literature approximation formulas whether they are based on ANN

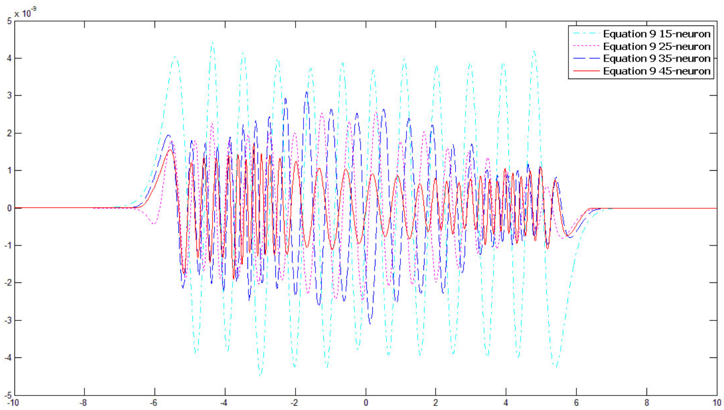

Figure 7. Error comparisons of Equation 9 with 15 neurons, Equation 9 with 25 neurons, Equation 9 with 35 neurons, and Equation 9 with 45 neurons in intervals $-10 \le x \le 10$

models or simple formulas in the interval -10 ≤ x ≤10. Moreover, this is the first study to include all types of formulas which include different types of mathematical equations. Additionally, by considering the ANN models with 15 neurons, the optimization, the most accuracy, and less absolute errors of about 8 to 9 decimal points of accuracy are the properties that will make ANN-based formulas to be more accurate and close to the real computational values. However, if the 6 decimal points of accuracy is still of interest to researchers, our ANN model based on the tanh and tanh transfer functions with only 2 neurons can be used even by simple calculators. The outcomes of this study represented as approximation equations can be beneficial for being embedded in statistics software environments such as SPSS, STATA, and R, and comprehensive Meta-analysis, OpenMeta[Analyst] , and Meta-mums tools.

Acknowledgments

This study is part of a Ph. D. thesis (No. 931507) and was approved and supported by the Research Council of Mashhad University of Medical Sciences, Mashhad, Iran and carried out in the Nuclear Medicine Research Center of Mashhad University of Medical Sciences, Mashhad, Iran.

References

Aludaat, K. M. and Alodat, M. T. (2008) . A note on approximating the normal distribution function. Applied Mathematical Sciences, 2(9):425-429.

Andrews, L.C. (1997). Special Functions of Mathematics for Engineers. 2nd ed. SPIE Publications, The International Society for Optical Engineering, Bellingham, WA, USA, 504p.

Bagby, R. J. (1995) . Calculating normal probabilities. Am. Math. Mon., 102(1):46-48.

Boiroju, N. K. and Rao, K. R. (2014) . Logistic approximation to standard normal distribution function. Assam Statistical Review, 28(1):27-40.

Bowling, S. R., Khasawneh, M. T., Aewkuekool, S., and Cho, B.R. (2009). A logistic approximation to the cumulative normal distribution. J. Indus. Eng. Manag., 2(1):114-127.

Bryc, W. (2002) . A uniform approximation to the right normal tail integral. Appl. Math. Comput., 127(2-3):365-374.

Casella, G. and Berger, R.L. (2001). Statistical Inference. 2nd ed. Duxbury Press, Pacific Grove, CA, USA, 660p.

Choudhury, A. (2014) . A simple approximation to the area under standard normal curve. Mathematics and Statistics, 2(3):147-149.

Cody, W.J. (1969). Rational Chebyshev approxiamtions for the error function. Math. Comput. , 23(107):631-637.

Divgi, D. R. (1979) . Calculation of univariate and bivariate normal probability functions. The Annals of Statistics, 7(4):903-910.

Gauss, K. F. (2004) . Theory of the Motion of the Heavenly Bodies Moving About the Sun in Conic Sections. Davis, C. H. (translator) . Dover Publications, Mineola, NY, USA, 400p.

Greene, W. H. (1993) . Econometric Analysis. 5th ed. Prentice Hall, Upper Saddle River, NJ, USA, 1026p.

Hamaker, H. C. (1978) . Approximating the cumulative normal distribution and its inverse. Appl. Statist., 27:76-77.

Hart, J.F. (1978). Computer Approximations. John Wiley & Sons, Inc., New York, NY, USA, 343p.

Hart, R. G. (1957) . A formula for the approximation of definite integrals of the normal distribution function. Mathematical Tables and Other Aids to Computation, 11(60):265-268.

Hart, R.G. (1966). A closed approximation related to the error function. Math. Comp., 20:600-602.

Johnson, N. L., Kotz, S., and Balakrishnan, N. (1994). Continuous Univariate Distributions: Vol. 1. 2nd ed. Wiley-Interscience, Hoboken, NJ, USA, 761p.

Johnson, N. L. and Kotz, S. (1970) . Distributions in Statistics: Continuous Univariate Distributions: Vol. 1. John Wiley & Sons, Inc., New York, NY, USA, 333p.

Kerridge, D. F. and Cook, G. W. (1976) . Yet another series for the normal integral. Biometrika, 63: 401-403.

Laplace, P. S. (1812) . Théorie analytique des probabilitiés. Courcier, Paris, 612p. Le Cam, L. and Yang, G. L. (2000) . Asymptotics in Statistics: Some Basic Concepts. 2nd ed. Springer-Verlag, New York, NY, USA, 287p.

Lin, J. T. (1989) . Approximating the normal tail probability and its inverse for use on a pocket calculator. Appl. Stat.-J Roy. St. C, 38(1):69-70.

Lin, J.T. (1990). A simpler logistic approximation to the normal tail probability and its inverse. Appl. Stat. J. Roy. St. C, 39:255-257.

Lyon, A. (2014). Why are normal distributions normal? Brit. J. Philos. Sci., 65(3):621-649.

McConnell, C.R. (1990). Pocket computer approximation for areas under the standard normal curve. Am. Stat., 44:63.

Moran, P.A.P. (1980). Calculation of the normal distribution function. Biometrika, 67:675-676.

Olabiyi, O. and Annamalai, A. (2012a). Invertible exponential-type approximations for the Gaussian probability integral Q(x) with applications. IEEE Wireless Communications Letters, 1(5):544-547.

Olabiyi, O. and Annamalai, A. (2012b). New exponential-type approximations for the erfc(.) and erfcp(.) functions with applications. 8th International Wireless Communications and Mobile Computing Conference; August 27-31; Limassol, Cyprus, p. 1,221-1,226.

Page, E. (1977). Approximations to the cumulative normal function and its inverse for use on a pocket calculator. Appl. Statist., 26:75-76.

Norton, R. M. (1989) . Pocket-calculator approximation for areas under the standard normal curve. Am. Stat., 43(1):24-26.

Revfeim, K. J. A. (1990) . More approximations for the cumulative and inverse normal distribution. Am. Stat., 44:63.

Shore, H. (2005) . Accurate RMM-based approximations for the CDF of the normal distribution. Commun. Stat.-Theor. M., 34:507-513.

Sokouti, B. , Sokouti, M. , and Haghipour, S. (2011) . A non-linear system's response identification using artificial neural networks. Elektron. Elektrotech. , 113(7):63-66.

Soranzo, A. Epure, E. (2012). Simply explicitly invertible approximations to 4 decimals of error function and normal cumulative distribution function. Available from: http://arxiv.org/abs/1201.1320v1. Accessed date:

Soranzo, A. and Epure, E. (2012) . Practical explicitly invertible approximation to 4 decimals of normal cumulative distribution function modifying Winitzki's approximation of erf. Available from: http://arxiv.org/abs/1211.6403. Accessed date:

Soranzo, A. and Epure, E. (2014). Very simply explicitly Iivertible approximations of normal cumulative and normal quantile function. Applied Mathematical Sciences, 8(87):4,323-4,341.

Stigler, S.M. (1986). The History of Statistics: The Measurement of Uncertainty before 1900. Harvard University Press, Cambridge, MA, USA, 432p.

Strecock, A.J. (1968). On the calculation of the inverse of the error function. Math. Comput., 22:144-158.

Vazquez-Leal, V. , Castaneda-Sheissa, R. , Filobello-Nino, U., Sarmiento-Reyes, A., and Oreal, J.S. (2012). High accurate simple approximation of normal distribution integral. Math. Probl. Eng., 2012:1-22.

Waissi, G.R. and Rossin, D.F. (1996). A sigmoid approximation of the standard normal integral. Appl. Math. Comput., 77:91-95.

Winitzki, S. (2008). A handy approximation for the error function and its inverse. Available from: http://sites.google.com/site/winitzki/sergei-winitzkis-files/erf-approx.pdf. Accessed date:

Yerukala, R., Boiroju, N.K., and Reddy, M.K. (2011). An approximation to the cdf of standard normal distribution. International Journal of Mathematical Archive, 2(7):1,077-1,079.

Yun, B.I. (2009). Approximation to the cumulative normal distribution using hyperbolic tangent based functions. J. Korean Math. Soc., 46(6):1267-1276.

Zelen, M. and Severo, N.C. (1970). Probability functions. In: Handbook of Mathematical Functions. Abramowitz, M. and Stegun, I. A. , eds. Dover Publications, Mineola, NY, USA, 1046p.