International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Application of Artificial Neural Networks to Forecast ITK Inhibitor Activity Data

Rama Devi Chalasani [*1], **Radhika Y** [2]

Submitted: 17/10/2022 **Revised:** 16/12/2022 **Accepted:** 27/12/2022

Abstract*:* An innovative method of testing the artificial neural network's effectiveness for ITK inhibitor data prediction was used. As a comparison, a multiple linear regression model was also developed. Using back propagation training, a multilayer perceptron MLP neural network was given the bioactivity estimate task. It was determined that there were enough buried neurons and that the learning rate was enough based on changes in RMSE. Thus, the final neural network consists of one output variable as the output layer, six input variables, eight hidden neurons, three nodes for bias accounting, and a 0.55 learning rate. To assess the robustness of the neural network model, test set data were forecasted, and forecast accuracy was measured.

Keywords: *Back propagation, hidden layer, Neurons, Neural Network*.

1. Introduction

This Due to their high processing speed and capability to handle complex non-linear functions, artificial neural networks (ANNs) have emerged as one of the most successful technologies in recent years. ANNs are used extensively in a variety of fields, including engineering [3], measurement and control [2], and sensors [1]. In order for connected information processing units to convert input into output, they need a model called a "neural network," which is characterized by an activation function [4]. It has always been done to compare the genetic makeup of neural networks with the human nervous system. Information is transmitted by interconnected units, much like how it does in human neurons. The neural network's first layer receives the raw input, processes it, and then sends the result to the hidden layers. The last layer, which creates the output, receives the processed information from the hidden layer [5]. The most fundamental type of neural network is the perceptron. A weighted summation and an activation function are used by a perceptron to process multidimensional information. The perceptron model's inability to handle non-linearity is a significant drawback. This restriction is removed by a multilayered neural network, which also aids in the resolution of non-linear issues. The output layer is connected to the hidden layer through the input layer. The connections are weighted, and the weights are optimized via a learning rule [6].

The primary goal of the research is to create a neural network model that estimates the biological activity of ITK inhibitors in order to evaluate the model's prediction power using particular, significant physco-chemical characteristics rather than experimental data. The two machine learning techniques that deal with function approximation issues are neural networks and support vector machines [7]. To predict activity data of physiologically relevant inhibitors against several protein targets, a variety of regression approaches have been used and proposed. The dimensionality of the dataset increases as the number of independent variables rises, making regression models built on a machine learning concept appear to be simpler, but controlling numerous aspects and understanding domain knowledge remain challenging. Creating a more straightforward prediction model, perhaps one that uses an empirical methodology, is therefore imperative [8]. In terms of empirical methods, multilayer perceptrons (MLP), a type of artificial neural network (ANN), have been used extensively in the domains of bioinformatics during the past few decades [9]. In this investigation, we assess the efficacy of neural networks in predicting ITK inhibitor activity data.

2. Materials And Methods

Using the back propagation method, neural net software was employed to train neural networks. Resilient back propagation (RPROP) is a technique that can be used with or without weight backtracking [10, 11], as well as the modified globally convergent version (GRPROP) developed by Anastasiadis et al. [12]. By letting users

[1] *Research Scholar, Department of Computer Science and Engineering, GITAM School of Technology, GITAM(Deemed to-be University) Visakhapatnam-530045, A.P, India.*
[2] *Professor, Department of Computer Science and Engineering, GITAM School of Technology, GITAM(Deemed to-be University) Visakhapatnam-530045, A.P, India*
Corresponding Author Email: ramadevi21vg@gmail.com

choose their own activation and error functions, the function offers versatile options. Additionally, generalized weights are implemented [13] through their calculation. The neural network can use the "backprop," "rprop+," "rprop-," "sag," or "slr" algorithms. Back propagation is referred to as "backprop," resilient back propagation is referred to as "rprop+" and "rprop-," and the modified globally convergent method is used in "sag" and "slr," respectively (grprop). A learning rate—either the learning rate associated with the smallest absolute gradient (sag) or the smallest learning rate (slr) itself—is modified as part of the globally convergent algorithm, which is based on robust back propagation without weight backtracking.

2.1 Dataset

The dataset for neural network analysis was a collection of 133 ITK inhibitor data from the results of multivariate regression analysis. The dataset was simply split into training and test sets, as shown in Table 1, with bioactivity acting as the dependent variable and six independent variables (Balaban index, logP, LUMO, HOMO, KC3 index, and shape index, respectively) chosen as the independent variables [14]. These variables describe the relationship between the response variable and the change in property values.

Table 1: ITK inhibitor training and test set data.

				Training Set		
activity	**BALABAN**	**LOGP**	**LUMO**	**HOMO**	**KC3**	**SHAPE**
-0.10607	1.26758	1.7469	-0.83198	-8.46474	3.39466	6.1269
-0.60264	1.09088	3.9511	-0.65749	-8.49199	3.50835	7.40238
0.011762	1.07498	3.5897	-0.58275	-8.45258	3.31011	7.16509
-0.00581	1.09088	3.8545	-0.49918	-8.33252	3.50835	7.96828
0.307417	1.07498	3.4931	-0.51726	-8.36368	3.31011	7.73093
-0.21542	1.27798	1.4997	-0.60697	-8.4041	3.10599	5.88807
0.071021	1.2752	1.908	-0.6324	-8.47381	3.10599	6.39383
-0.11919	1.27719	2.2694	-0.57413	-8.37499	3.51423	6.63225
0.300448	1.0553	3.6846	-0.6371	-8.47916	3.31011	7.67591
0.287611	1.07309	3.199	-0.50533	-8.37269	3.59878	8.15593
0.029969	1.25813	3.1322	-0.62515	-8.46272	3.39466	7.68693
-1.71238	1.08549	4.1983	-0.94919	-8.51629	3.79703	7.64039
-0.74179	1.07012	3.8369	-0.97376	-8.50695	3.59878	7.40238
-0.62054	1.08549	4.1017	-0.90497	-8.45064	3.79703	8.20638
-1.69808	1.28997	1.8975	-0.52544	-8.81458	2.13435	7.42828
-2.21908	1.28189	2.6058	-0.53495	-8.97438	2.13435	7.50228
-2.72048	1.27434	2.3531	-0.5249	-8.95243	2.33847	8.25666
-2.13636	1.2801	3.3976	-0.52906	-8.95976	2.42302	8.1119
-2.06593	1.2801	2.7453	-0.82046	-9.04527	2.42302	7.68087
-1.70939	1.28372	2.7453	-0.50005	-8.78236	2.42302	7.68087
-3.09565	1.2967	3.073	-0.17675	-8.93134	2.33947	7.73496
-1.4344	1.28997	0.874801	-0.95814	-9.09837	2.13435	7.38148
-1.49069	1.28189	1.7588	-0.46428	-8.70377	2.13435	7.76101
-3.65154	1.54713	1.7763	-0.53487	-8.9766	2.30102	6.94553
-2.25535	1.29945	2.9483	-0.21375	-8.8672	2.07798	8.05816

-2.53876	1.31015	3.3464	-0.77193	-9.0172	2.07798	8.41538
-1.64129	1.29716	3.7401	-0.88868	-9.03766	2.36666	8.67633
-2.10408	1.30775	4.1382	-0.93898	-9.03583	2.36666	9.03661
-2.16347	1.28189	2.7024	-0.56805	-8.73479	2.13435	6.91128
-2.03164	1.29681	2.144	-0.95517	-8.94298	2.01478	7.98461
-1.88687	1.30674	2.3905	-0.92366	-8.9407	2.29514	8.21582
-1.81604	1.28751	2.1492	-0.88734	-8.99535	2.13435	7.98461
-1.4609	1.28054	2.5455	-0.89697	-8.96169	2.13435	8.55479
-2.5088	1.2652	3.8503	-0.90043	-8.86662	1.7261	6.93611
-2.80119	1.12762	5.0675	-0.93322	-8.90224	1.93023	7.68592
-1.90875	1.2652	2.5961	-0.66216	-8.77583	1.7261	6.90508
-1.60784	1.27608	2.6482	-0.90693	-8.89762	1.7261	7.45743
-2.62081	1.27608	2.3007	-0.8565	-8.88731	1.7261	7.45743
-2.50794	1.28997	1.8975	-0.96046	-8.97064	2.13435	7.42828
-2.67356	1.52085	0.9858	-0.90684	-8.76119	1.41612	4.58915
-0.64103	1.28997	1.6134	-0.60404	-8.43434	2.13435	7.12725
-1.04978	1.28189	1.818	-0.6762	-8.51866	2.13435	7.76101
-0.855	1.28189	2.665	-0.49816	-8.42218	2.13435	7.50228
-0.67562	1.28372	3.4568	-0.71358	-8.47871	2.42302	8.1119
-0.1985	1.2801	3.4568	-0.7897	-8.48865	2.42302	8.1119
-0.28513	1.2801	3.183	-0.43036	-8.35614	2.42302	7.96142
-0.67908	1.28102	2.4123	-0.09543	-8.2552	2.33847	8.25666
-0.75826	1.27434	2.4123	-0.4087	-8.41383	2.33847	8.25666
-1.41247	1.28372	3.1322	-0.6359	-8.47246	2.42302	7.73496
-1.25874	1.28997	1.9567	-0.86522	-8.41884	2.13435	7.42828
-1.20835	1.2801	3.1322	-0.09647	-8.30285	2.42302	7.73496
-0.27655	1.28102	2.53	-0.65463	-8.46496	2.33847	7.88787
-0.27655	1.27434	2.53	-0.8457	-8.47154	2.33847	7.88787
-1.52238	1.27287	1.9735	-0.77731	-8.43864	2.63435	8.26046
-2.04998	1.18918	3.6717	-1.13856	-8.4535	1.98659	6.33283
-0.86463	1.25719	4.8791	-1.11953	-8.43565	1.93023	7.94663
-1.19106	1.25222	4.4273	-1.11724	-8.42493	2.23859	7.09057
-0.48202	1.12181	5.7	-1.12411	-8.46655	2.13435	8.14399
-1.67831	1.12308	5.4473	-1.13821	-8.45702	2.33847	8.88345
-0.87956	1.26589	3.559	-1.03258	-8.50194	1.93023	8.4804
-1.52473	1.26637	2.3487	-0.85795	-8.61768	2.33847	7.34606
-0.64144	1.28505	2.7765	-0.82453	-8.43437	2.2189	8.44328

-0.84567	1.29802	3.023	-1.10143	-8.53537	2.49926	8.67395
-1.50775	1.1242	3.4521	-1.02214	-8.46359	2.42302	8.90883
-0.43279	1.11075	3.5042	-1.11652	-8.46122	2.42302	9.45957
-0.43275	1.1326	2.4629	-1.10592	-8.54341	2.33259	9.2
-1.00015	1.13325	2.8243	-1.10753	-8.5407	2.62127	9.43237
-0.5323	1.29704	2.8725	-0.84435	-8.55392	2.28211	8.44328
-1.87563	1.32171	3.2856	-0.7875	-8.82661	2.59047	8.67395
-1.37774	1.52639	0.6545	-1.13808	-8.54863	2.31269	6.549
-2.12685	1.52843	1.2827	-1.31081	-8.60159	2.21506	7.10276
-1.97844	1.53574	1.8451	-0.77413	-8.44031	2.50514	7.33373
-1.29674	1.29037	1.3411	-1.12847	-8.53095	2.33847	6.30554
-0.551	1.28921	1.7374	-0.80488	-8.53199	2.33847	6.81842
-0.96992	1.27434	2.5674	-1.12666	-8.5531	2.33847	7.29772
-0.15162	1.27434	2.53	-0.77256	-8.43386	2.33847	7.88787
-0.35081	1.28189	1.818	-0.82922	-8.47782	2.13435	7.76101
-0.64201	1.28997	0.934001	-0.88141	-8.50452	2.13435	7.38148
-1.19391	1.2268	3.9775	-1.10931	-8.75121	1.90892	8.52657
-1.53333	1.47045	2.3295	-1.13417	-8.83484	1.70479	8.35404
-1.71778	1.18237	2.5134	-1.12838	-8.82231	1.90892	9.08199
-1.334	1.2392	4.7131	-0.69814	-8.67145	1.90304	8.61412
-0.96346	1.24157	4.0601	-0.94311	-8.71582	1.81848	8.52657
-1.30109	1.2455	2.4641	-0.95824	-8.76279	1.61436	8.02367
-1.19542	1.23926	3.3481	-0.66473	-8.72431	1.61436	8.42545
-1.32519	1.23509	4.0601	-1.11645	-8.72935	1.81848	8.52657
-1.27424	1.24361	3.0311	-0.76663	-8.56313	2.22673	7.91494
-1.17233	1.09096	4.511	-1.2464	-8.89489	2.02261	8.58563
-1.08033	1.22583	3.7818	-1.12105	-8.73857	1.81848	7.9504
-1.33639	1.09369	2.915	-1.10031	-8.91465	1.81849	8.0852
-1.27719	1.2209	2.1027	-0.98399	-8.81722	1.41612	7.80056
-0.08511	1.2209	2.1993	-0.89591	-8.33401	1.41612	7.15524
-1.80735	1.48362	2.1136	-1.13289	-8.80379	1.41612	6.96325
-1.30218	1.21296	3.6987	-1.12324	-8.81087	1.62024	8.30363
-1.32426	1.21296	3.7953	-1.18437	-8.36369	1.62024	7.66475
-1.90023	1.28189	3.1332	-0.38474	-8.44161	2.13435	6.91128
-2.71402	1.09412	4.8176	-0.47078	-8.43816	2.46768	7.90825
-3.08954	1.15343	4.1354	-0.51713	-8.44388	2.40652	7.07671
-1.66933	1.12184	4.1354	-0.59074	-8.43217	2.46768	7.07671

-1.41427	1.28997	2.4249	-0.50908	-8.43755	2.13435	6.8336
-1.40672	1.28997	2.6607	-0.39326	-8.3917	2.13435	6.8336
-2.18328	1.28997	1.7589	-0.69007	-8.5025	2.13435	7.0816
-1.76574	1.28997	1.5889	-0.76896	-8.53348	2.13435	7.0816
-2.28918	1.28997	1.9927	-0.78192	-8.66806	2.13435	6.78908
-1.86641	1.28997	1.4156	-0.60993	-8.54527	2.13435	6.78908
-1.89124	1.28997	1.2456	-0.68865	-8.53768	2.13435	6.78908
-1.63597	1.28997	1.4022	-0.9087	-8.69184	2.13435	6.78908
-0.82917	1.28648	2.5577	-0.69444	-8.50205	2.42302	7.28563
-0.89744	1.28648	2.8583	-0.74794	-8.51795	2.42302	7.43018
-0.83596	1.27657	1.4976	-0.96971	-8.57262	2.63435	7.58818
-1.48203	1.08094	3.4105	-0.94644	-8.56378	2.66013	8.34595
-1.87824	1.06848	3.7966	-0.61675	-8.44727	2.5426	8.3252
-1.28733	1.05073	3.8915	-0.4863	-8.43947	2.5426	8.86054
-1.31011	1.08925	3.8735	-0.71483	-8.4432	2.46768	7.83112
-0.99862	1.08925	3.4256	-0.84731	-8.47274	2.46768	8.07208
-0.76621	1.09992	3.0763	-0.99062	-8.50343	2.46768	7.99825
-0.73694	1.09436	2.2358	-0.91769	-8.43903	2.75636	8.20462
-0.01167	1.09992	1.9651	-0.91097	-8.5135	2.46768	7.7125
-1.6217	1.09992	2.4852	-0.38262	-8.23103	2.46768	7.85093
-1.59569	1.08925	1.817	-0.49858	-8.37667	2.46768	8.34746
-0.64318	1.09495	2.8915	-1.28495	-8.49203	2.60377	8.45603
-0.21163	1.09373	3.166	-1.03364	-8.53224	2.6728	8.25585
-0.93285	1.2801	3.6512	-0.58081	-8.49654	2.42302	7.36109
-0.89934	1.27434	2.9982	-0.83904	-8.57572	2.33847	7.29772
-1.90399	1.27434	2.8805	-0.40552	-8.41085	2.33847	7.65006
-1.13133	1.2801	3.925	-0.57696	-8.52532	2.42302	7.50475

Test Set

activity	**BALABAN**	**LOGP**	**LUMO**	**HOMO**	**KC3**	**SHAPE**
-1.5733	1.27287	2.5594	-1.43531	-8.89371	2.63435	8.17491
-1.97552	1.28102	2.4708	-1.06036	-9.10845	2.33847	7.88787
-2.115	1.29775	3.1365	-0.71665	-8.91311	2.01478	8.55479
-0.85055	1.27904	2.0541	-1.01295	-8.62032	2.33847	7.21907
-1.01507	1.2689	2.6882	-0.94374	-8.55525	2.8268	8.3625
-0.61456	1.09992	2.733	-0.9535	-8.5341	2.46768	7.7125
-0.5013	1.09436	2.4481	-1.00614	-8.49699	2.75636	7.94854

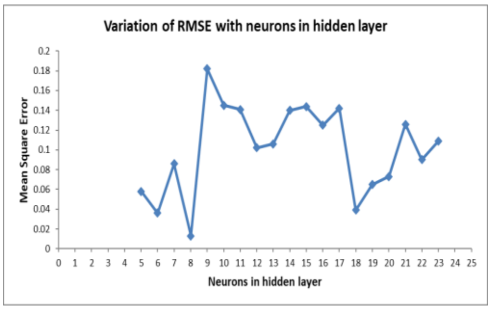
3. **Results and Discussion**

The processed data was normalized using the min-max method, which converts the data into a common range and eliminates the scaling influence from every variable. The min-max method preserves the variables' initial distribution, in contrast to Z-score normalization, median, and MAD methods. The 133 data points in the normalized data set, which includes six variables (BALABAN, LOGP, LUMO, HOMO, KC3 and SHAPE), have been randomly split into training (95%) and testing (5%) groups. For the bioactivity estimate challenge, a multilayer perceptron MLP neural network trained in back propagation was selected. A function that can adequately relate all the variables under consideration is required due to the nonlinear association between input and output data for a certain network. As a result, nonlinear transfer functions are needed to add nonlinearity into a network for the hidden neurons in the network. According to reports, an MLP network that has enough neurons satisfies the universal approximation property [15] [16].

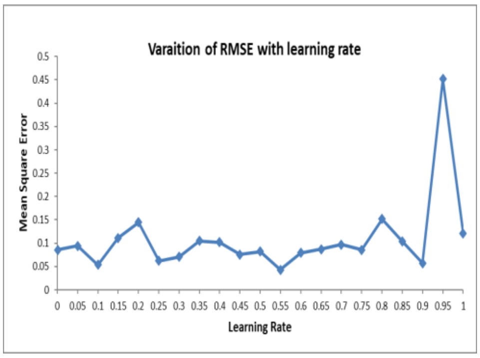
For issues involving function approximation, several neural networks are accessible. In contrast to Radial Basis Function (RBF) networks and Generalized Regression neural networks (GRNN), a Multilayer Perceptron (MLP) neural network trained using back propagation was chosen because it can effectively learn big data sets [17]. It has been demonstrated that MLP works well for establishing nonlinear relationships between sets of variables and is useful for solving problems involving function approximation. Since the back propagation learning method is more widely used, significant research has been done to speed up its convergence because the fundamental algorithm is too slow for the majority of actual applications. Along with back propagation, other algorithms were proposed, including globally convergent algorithms with the shortest absolute gradient (sag) or the smallest learning rate, robust back propagation with and without weight backtracking, and (slr). All of these methods were put to the test on a training set to attain the lowest RMSE possible in order to evaluate how well each training algorithm performed.

When a particular training algorithm failed to produce the desired results on an MLP in some circumstances, it may have been because the learning rule's attempt to reproduce the values of the network parameters failed to converge and the given network was unable to carry out the desired function, possibly as a result of not having enough hidden neurons [18]. The necessary hidden neurons, however, have not yet been identified theoretically. Due to underfitting and excessive statistical bias, if the hidden neurons are few, there will be a high training error and high generalization error. On the other hand, if the hidden neurons are significantly larger than the variables, there may be a low training error, but there will still be a high generalization error because of overfitting and high variation. Without training multiple networks and calculating their individual generalization errors, it is typically impossible to establish the optimal number of hidden neurons [19].

The neural network must meet certain conditions, including a maximum number of training epochs, a learning rate, and a minimum number of nodes in a single hidden layer [20]. In this study, the generalization (testing) RMSE error was varied sequentially from 5 to 23 hidden neurons [21] [22] and the learning rate was varied from 0.01 to 1.0 in increments of 0.05 [23]. We calculated the mean square error (MSE) for each configuration between the model output and the measured data. The maximum MSE-measured model performance is shown in Figure 1 by the optimal hidden layer neuron count and learning rate. The optimal learning rate and the quantity of neurons in the hidden layer were determined by trial and error.



A



B

Fig. 1: Mean square error calculation between data and output from variations with the number of neurons in the hidden layer (A) and variations with the learning rate (B).

Finding the optimal number of hidden neurons—i.e., the number at which the testing RMSE falls—was the criteria used. The ideal learning parameters are determined by repeatedly training a neural network with the optimal number of hidden neurons after it has been reached. The final neural network therefore includes six input variables, eight hidden neurons, three nodes a 0.55 learning rate, and one output variable as the output layer for bias accounting (Figure 2).

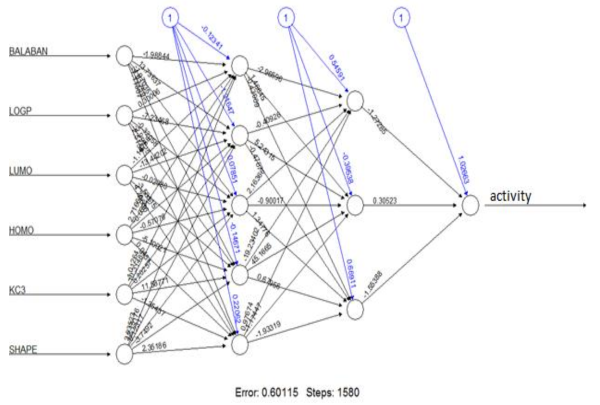


Fig. 2: Presentation of neural network with back propagation algorithm.

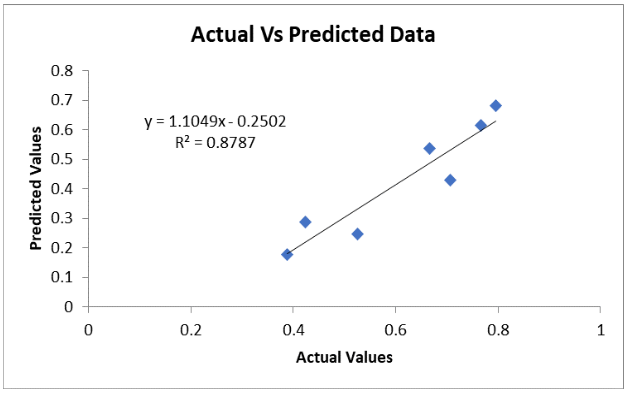


Fig. 3: Prediction of test set data by neural network**.**

According to test set data, which was anticipated as a tool to assess the robustness of neural network models, the most crucial feature of a model should be its capacity to generalize. Overfitting, however, prevents the generalization of models [24]. The model's capacity to generalize shows that it has the potential to perform well when tested using test set data that were not used to train the model. Therefore, the data set was divided into two sets at random, with 5% of the data used to test the model and 95% of the data used to train the model, which calculates the gradient and modifies network parameters like weights and biases. The model training process was stopped when the network attempted to overfit the data or made a mistake on the dataset, and the weights were initialized randomly. Figure 3 shows that the model correctly predicted the test set 78% of the time, with a correlation value of 0.8787, supporting the neural network model's capacity for prediction.

It should be mentioned that when building a strong neural network model, it's crucial to consider the appropriate amount of layers, neurons in the hidden layer, learning rates, and model training epochs. Since mean square error calculation was used in this instance, the number of hidden layers and learning rates were optimized. In addition, the model might not accurately represent nonlinearity if the hidden layer's neurons are not taken into account. The model may, however, become overfit if too many neurons are used, which prevents it from being generalized [25].

Additionally, a multivariate linear association analysis on the training data produced a low r2 value of 0.49, a superior F-statistic, and low p-values, suggesting that the proposed model predicts activity with lower accuracy than neural net model but with higher predictive capacity on test data. 0.24 was shown as an error on a cross-validated mean square error plot in Figure 4. Given that Figure 5 shows that the RMSE values are below 0.3, a subset of the training set was plotted to examine how the RMSE varied with the duration of the training set.

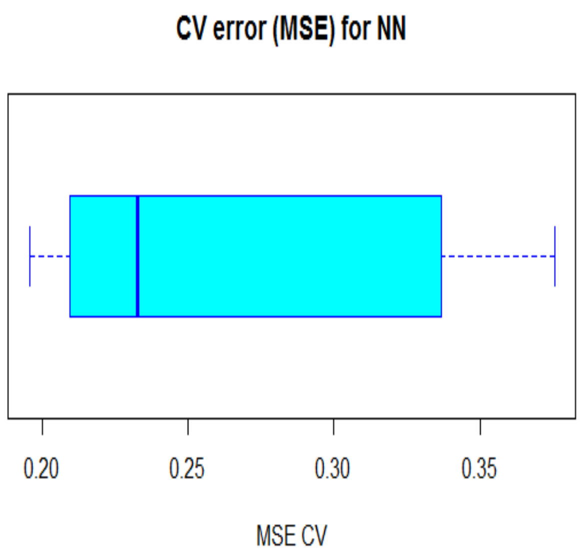


Fig. 4: Cross-validated mean square error of the dataset.

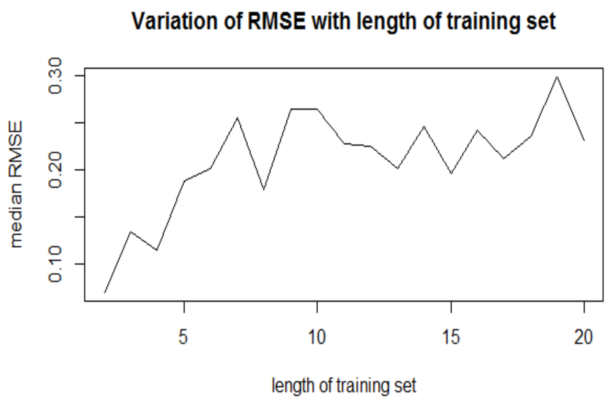


Fig. 5: Variation of RMSE with length of the training set.

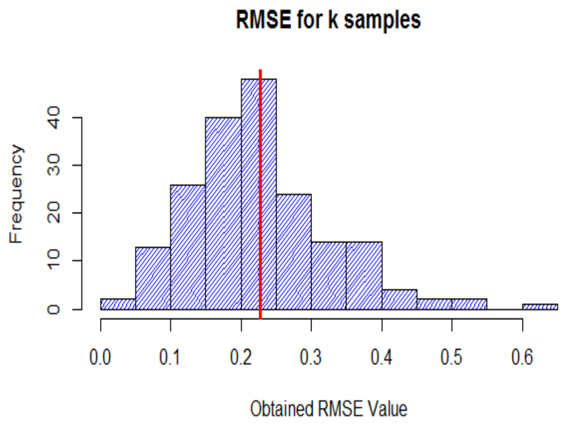


Fig. 6: Obtained RMSE values for k sample size.

The histogram in figure 6 shows the distribution of the dataset's RMSE while also highlighting the average RMSE (vertical red line) over k distinct samples. The activity variable's obtained RMSE ranges from 0 to 0.6. These levels of RMSE are considered to be modest.

4. Conclusion

The neural network model and the linear regression model both provided a respectable level of accuracy for the training and test set data of the ITK inhibitors used in this experiment. Here, 8 neurons in the hidden layer were proposed by a network growth technique, and the mean square error calculation indicated that the learning rate was 0.55. Additionally, a correlation coefficient of 0.8787 and a prediction accuracy of 78% for the test set provided by the neural network model supported its predictive capabilities.

Acknowledgements

Authors are declaring thanks to Department of Computer Science and Engineering, GITAM School of Technology, GITAM(Deemed to-be University) Visakhapatnam, A.P, India for partially supported for completion of this research work.

Author contributions

RamaDevi Chalasani: Conceptualization, Methodology, Software, Field study, Visualization, Investigation, Writing-Reviewing and Editing.

Radhika Y: Data curation, Writing-Original draft preparation, Software, Validation.,

Conflicts of interest

The authors declare no conflicts of interest.

References

López-Molina, T.; Pérez-Méndez, A.; Rivas-Echeverría, F. Data analysis techniques for neural networks-based virtual sensors. Proceedings of the 8th WSEAS International Conference on Neural Networks, Vancouver, Canada, June 18–20, 2007; 8, pp. 76–83.

Huang, Y. Advances in artificial neural networks methodological development and application. Algorithm 2009, 2, 973–1007.

Chen, Y.J.; Chen, Y.M.; Wang, C.B.; Chu, H.C.; Tsai, T.N. Developing a multi-layer reference design retrieval technology for knowledge management in engineering design. Expert Syst. Appl 2005, 29, 839–866.

K.P. Singh, A. Basant, A. Malik, G. Jain. Artificial neural network modeling of the river water quality-case study. Ecol. model., 220 (2009), pp. 888-895.

C. Bishop. Neural Networks for Pattern Recognition.Oxford University Press, New York (1995).

G.D. Garson. Interpreting neural network connection weights.Artif.Intell. Expert, 6 (1991), pp. 47-51.

Chuang, C.C.; Su, S.F.; Jeng, J.T.; Hsiao, C.C. Robust support vector regression networks for function approximation with outliers. IEEE Trans. Neural Networks 2002, 13, 1322–1330.

Keun Young Lee et al. Application of an artificial neural network (ANN) model for predicting mosquito abundances in urban areas. Ecological Informatics Volume 36, November 2016, Pages 172-180.

J. Dopazo, W. Huaichun, J.M. Carazo. A new type of unsupervised growing neural network for biological sequence classification that adopts the topology of a phylogenetic tree.Lect. Notes Comput. Sci., 1240 (1997), pp. 932-941.

Riedmiller M. (1994) *Rprop - Description and Implementation Details.* Technical Report. University of Karlsruhe.

Riedmiller M. and Braun H. (1993) *A direct adaptive method for faster backpropagation learning: The RPROP algorithm.* Proceedings of the IEEE International Conference on Neural Networks (ICNN), pages 586-591. San Francisco.

Anastasiadis A. et. al. (2005) New globally convergent training scheme based on the resilient propagation algorithm. Neurocomputing 64, pages 253-270.

Intrator O. and Intrator N. (1993) *Using Neural Nets for Interpretation of Nonlinear Models.*Proceedings of the Statistical Computing Section, 244-249 San Francisco: American Statistical Society (eds).

Rama Devi Ch, and Radhika Y. 2019.A machine learning access for selection of influential variables of several ITK inhibitors using regression research. International Journal of Recent Technology and Engineering, Volume-8 Issue

-2S8,pp. 1867-1875.
Bishop, C.M. Neural Networks for Pattern Recognition; Oxford University Press: New York, NY, USA, 1995.
Huang, G.B.; Chen, L.; Sien, C.K. Universal approximation using incremental constructive feedforward networks with random hidden nodes. IEEE Trans. Neural Networks 2006, 17, 879–892.
Garcia-Cuesta, E.; Galvan, I.M.; de Castro, A.J. Multilayer perceptron as inverse model in a ground-based remote sensing temperature retrieval problem. J. Eng. Appl. Artif. Intell 2008, 21, 26–34.
Ahmed M. Wefky et al. Alternative Sensor System and MLP Neural Network for Vehicle Pedal Activity Estimation. Sensors 2010, 10(4), 3798-3814.
Haykin, S. Neural Networks: A Comprehensive Foundation; MacMillan College Publishing: New York, NY, USA, 1994] [Bishop, C.M. Neural Networks for Pattern Recognition; Oxford University Press: New York, NY, USA, 1995
D.J. Hill, B.S. Minsker Anomaly detection in streaming environmental sensor data: a data-driven modeling approach. Environ. Model Softw., 25 (2010), pp. 1014-1022
Haykin, S. Neural Networks: A Comprehensive Foundation; MacMillan College Publishing: New York, NY, USA, 1994
Bishop, C.M. Neural Networks for Pattern Recognition; Oxford University Press: New York, NY, USA, 1995
Keun YoungLee et al. Application of an artificial neural network (ANN) model for predicting mosquito abundances in urban areas. Ecological Informatics Volume 36, November 2016, Pages 172-180
U. Schlink, S. Doring, E. Pelikan, G. Nunnari, G. Cawley, H. Junninen, A. Greig, R. Foxall, K. Eben, T. Chatterton, J. Vondracek, M. Richter, M. Dostal, L. Bertucco, M. Kolehmainen, M. Doyle A rigorous inter-comparison of ground-level ozone predictions. Atmos. Environ., 37 (2003), pp. 3237-3253
D.J. Hill, B.S. Minsker. Anomaly detection in streaming environmental sensor data: a data-driven modeling approach. Environ. Model Softw., 25 (2010), pp. 1014-1022