ORIGINAL ARTICLE

OPEN ACCESS

Enhancing Cancer Dynamics Simulation Leveraging Levenberg-Marquardt Backpropagation Neural Network with Numerical Computing Framework

Zawar Hussain [1], **Muhammad Sulaiman** [2], **Fizan Ahmad** [3]

Authors' Affiliations:

[1]Department of Mathematics, Abdul Wali Khan University, Mardan, KPK, Pakistan
Email: z.math1998@gmail.com

[2]Department of Mathematics, Abdul Wali Khan University, Mardan, KPK, Pakistan
Email: msulaiman@awkum.edu.pk

[3]Department of Mathematics, Abdul Wali Khan University, Mardan, KPK, Pakistan
Email: ahmadfaizan1616@gmail.com

Article History:

Submitted: July 12, 2026
Revised: November 19, 2026
Accepted: November 23, 2026
Published online: Nov 25, 2026

Corresponding author(s):

Zawar Hussain
Email: z.math1998@gmail.com

ABSTRACT

In the current study, a set of four coupled nonlinear ordinary differential equations is investigated for understanding the time-dependent dynamics of interactions between cancer cells, the immune system, anti-PD-L1 antibodies, and interleukin-10 (IL-10). From a mathematical perspective, the solution of two different parameter regions of the proposed model is performed using the widely used fourth-order Runge-Kutta (RK4) method for generating high-quality reference solutions. These numerical solutions are considered as supervised data for training the Levenberg-Marquardt backpropagation neural network (LMB-NN) to solve the solution profile of the coupled nonlinear ordinary differential equation system. MSE, regression analysis, and error histogram analysis are the basic tools for evaluating the performance of the neural network using independent training, validation, and testing data sets. Since the average MSE is of the order 10^{-12} - 10^{-15}, it can be said that there is very good agreement between the solutions of the RK4 method and the LMB-NN. The mean values of the test MSE for the examined examples are 4.90×10^{-12}, 1.52×10^{-14}, and 3.75×10^{-15}, respectively. Moreover, the error distributions, along with the regression analysis, provide additional arguments proving the very good numerical agreement between the neural network predictions and the solutions obtained via the RK4 method. The above results show the excellent performance of the LMB-NN, which is the supervised data-driven surrogate model, in predicting the numerical solutions of the ODE model of the specified cancer-immune system. Biological validation, parameter estimation, treatment optimization, and clinical prognosis are out of the scope of this work.

Keywords: Mathematical modeling; Cancer dynamic; Nonlinear ODEs LMB-NN; RK4 method; Optimization Algorithm

(MSC2020) Codes: 92C60, 92C50, 92C55, 68T05.

1. | INTRODUCTION

A sophisticated disease, cancer results from the unrestrained growth of malignant cells intimately related to tumor cell-host immune system interactions. The immune system's ability to spot strange cells and start a fit immunological reaction is a crucial part of cancer monitoring. But cancer cells can come up with defenses that help them elude immune detection and inhibit anti-tumor responses. These characteristics have propelled the development of cancer immunotherapies meant to help or restore immune-mediated tumor control. Dendritic cells (DCs) are essential for antitumor immunity among the many cellular components involved in it because of their ability to catch and present tumor-related antigens (TAAs) to T lymphocytes. Dendritic-cell vaccination (DCV) is therefore being investigated as an immunotherapy approach whereby DCs are loaded with tumor-associated antigens to stimulate tumor-specific immune responses. Because activated CD8+ T cells can spot and eliminate cancer cells, cytotoxic T lymphocyte interaction with antigen-presenting DCs is essential. The effectiveness of this response is mostly controlled by multiple linked biological processes, including immune suppression, immune-cell activation, tumor development, and cytokine control. Such interactions create a nonlinear dynamical system where changes in one population might influence the temporal evolution of others. The immune response's inherent and adaptive parts work together. The innate immune system provides an instant, non-specific defense, including cells like neutrophils, macrophages, and natural killer cells, while the adaptive immune system generates antigen-specific responses mostly mediated by B and T lymphocytes [1]. Inside T lymphocytes, CD4+ helper cells and CD8+ cytotoxic cells carry out different but related jobs in tumor elimination and immunological control. Immune checkpoint systems make the interaction between T lymphocytes and tumor cells even more challenging. Specifically, the programmed cell death protein 1 (PD-1)/programmed death-ligand 1 (PD-L1) pathway helps to regulate T-cell activity [2]. Earlier studies on PD-1 and PD-L1 control of cancer immunity have been reviewed [3], [4]. The interaction between PD-L1 and PD-1 helps tumor immune evasion to be suppressed and T-cell activity to be suppressed [5]. Therefore, immune-checkpoint blockade has been investigated as a means of restoring or increasing immune response against cancer [6].

Cytokines are another control tool in the tumor-immune microenvironment. Immunoregulatory cytokine IL-10 controls immune and inflammatory responses [7]. Its impact on immune-cell activity [9] and tumor development [8], especially antigen-stimulated CD8+ T cells [10], has been investigated in earlier studies. These results show that immune regulation and tumor development cannot be considered independently. Rather, a linked system of tumor cells, immune cells, checkpoint processes, and cytokine-mediated effects is created whose temporal behavior may be represented by coupled mathematical equations. With mathematical modeling, these biological

ties may be converted into quantitative dynamical systems. Time-dependent processes in biology, medicine, and other scientific fields have been modeled using ordinary differential equations (ODEs). To study tumor development, immunological responses, treatment effectiveness, and the interactions between immune cell types and malignant cells, mathematical models have been developed in cancer research. Using models based on differential equations, one can investigate the temporal behavior of interacting biological populations. [12] These models could also provide a framework for examining how changes in model parameters affect the tumor-immune dynamics [13]. Several mathematical models have been put forward to describe the immune reactions and cancer development. ODE-based methods are extremely useful when the main aim is to show how interacting populations change over time [15]. More complex biological mechanisms can lead to parameter-dependent solution behavior and highly coupled nonlinear equations [16]. Therefore, obtaining an exact numerical answer to the mathematical system that arises is a key component of computational analysis [17]. If the governing equations have major nonlinearities, rigidity, singular behavior, or several interacting time scales, the numerical treatment might grow more complex [18]. To approximate the solutions to nonlinear biological and cancer models, earlier numerical studies have used conventional computational techniques. [19] Even if these approaches offer insightful reference solutions for mathematical analysis, their utility may be affected by the features of the underlying equations, parameter values, and numerical discretization. Thus, the growing complexity of nonlinear biological models pushes one to look into alternative computational techniques able to exactly predict solution trajectories [21]. Numerical techniques based on neural networks [22] provide one such computational framework since they can estimate nonlinear functions and solution trajectories for nonlinear differential equations. For training neural networks in nonlinear mathematical problems especially, optimization techniques depending on the Levenberg-Marquardt approach have been applied. Research into many mathematical and biological systems has shown that these methods may be used as numerical approximation tools [25].

Backpropagation neural networks (LMBNNs) based on Levenberg-Marquardt have been employed before in bioinformatics and related biological systems [29–31]. Their use has also been recorded for fractional-order systems [32,33], nonlinear nanofluid models [34–37], and higher-order systems of differential equations [38–42]. These investigations demonstrate how useful LMBNN-based approaches are for several kinds of nonlinear mathematical issues. Still, most of the studies represented by [29–42] look at nanofluid models, higher-order differential equations, fractional-order systems, and bioinformatics. The numerical prediction of the coupled cancer-immune dynamical system covered in this research has not been properly investigated using an LMBNN method. More specifically, it is still needed to find out if an LMBNN can precisely imitate the solution trajectories

produced by the nonlinear cancer-immune model with respect to a pertinent reference numerical solution. This illustrates the specific computational gap filled in this study. Moreover, one must distinguish between numerical approximation and clinical prediction. The mathematical model is solved with the neural network used in this study; it is not meant to forecast one patient's clinical result or the success of a cancer therapy. Similarly represented by the agreement between the LMBNN solution and a reference numerical solution is the numerical validation of the computational approach, not the experimental, biological, or clinical validation of the cancer model.

This paper therefore investigates the given nonlinear cancer-immune ODE system with an LMBNN-based numerical technique. The paper focuses on assessing the numerical approximation of the model solution and its precision relative to a reference solution. The goals are to establish the coupled cancer-immune dynamics, create the LMBNN approximation, evaluate the resultant solution using graphical comparisons and quantitative error indicators, and ascertain the capability of the suggested approach as a numerical tool for the nonlinear system. Therefore, the contribution is mathematical and computer-based, with the results examined in terms of numerical accuracy rather than clinical prediction or treatment validation for particular patients.

2. | LITERATURE REVIEW

Recent cancer-immune modelling studies indicate that mathematical models are becoming more often used to simulate tumor growth, T-cell activity, immune-checkpoint signaling, and treatment responses. Previous editions of Lai and Friedman's mathematical models of tumor-immune interactions included cancer immunization and immune-checkpoint suppression; later ODE-based studies considered radiation with anti-PD-L1 medication [18,19]. Though they plainly depict biological interactions, these mechanistic models have complexity, dependence on parameters, and model-specific assumptions that make numerical solution and parameter study very challenging computationally. Recent research also demonstrates the scope of mathematical cancer-immune modelling feasible. Considering effector and worn-out cytotoxic T lymphocytes, neoantigen production, draining lymph nodes, and the impact of radiation, Takayanagi et al. (2025) built a mechanistic model of the cancer-immunity cycle and then compared the predictions of the model with animal study results [43]. Using molecular data and mathematical modeling, Sakai et al. looked at when immune-checkpoint inhibitors should be administered together with radiation therapy, concentrating on PD-1/PD-L1 and related ligand-receptor interactions [44]. Goodin et al. created a three-dimensional mathematical model for pancreatic ductal adenocarcinoma to investigate anti-PD-L1 and CD8+ T-cell-activating nanotherapy using experimental and in situ data [45]. Li et al. [46] created a multiscale tumor-immune model for colorectal cancer by combining anti-PD-L1 therapy and cancer immunization and applying approximate Bayesian analysis to represent treatment-

response heterogeneity. These studies point to a move toward more comprehensive and empirically driven cancer-immune models. The present study's scope differs from these latest mechanistic and multiscale approaches. Using four state variables CD8+ T lymphocytes, cancer cells, IL-10, and anti-PD-L1 antibodies it focuses on numerical approximation of the resulting coupled nonlinear ODE trajectories. Unlike past studies that use RK4-generated trajectories as numerical reference data [43–46], the current one calibrates models against experimental or patient data. Therefore, the contribution here is computational rather than clinical; it examines if an LMB-NN can reproduce the numerical solution pathways of the given model across the tested parameter settings. Already used for nonlinear mathematical and engineering challenges including bioinformatics, fractional-order systems, nanofluid models, and higher-order differential systems, LMB-NN has moreover, lately neural-network research has integrated the Levenberg–Marquardt method with neural-network approximation for nonlinear coupled differential systems and revealed relative L2-error-based evaluation [47].

Unlike conventional numerical integration, LMB-NN provides a learned input-output representation that could be evaluated after training at locations along the shown trajectory range. The principal computational activity of LMB-NN for the current study is this one. Given that RK4 trajectories are the reference targets, it does not inherently offer more stability or accuracy than RK4. Research gap. Although much research on mechanistic cancer-immune modelling and separate applications of LMB-NN to nonlinear differential systems exists, supervised LMB-NN approximation of the current four-component cancer-immune ODE system has not been systematically investigated within the technique used here. Given the chosen tumor-growth and carrying-capacity scenarios, can LMB-NN faithfully rebuild numerically generated trajectories? Though small in focus on numerical trajectory estimation, this analysis helps to address that question.

3. | PROBLEM FORMULATION

Developing cancer and tumour-preventing treatment techniques is crucial to better lifestyle and quality of life, as well as gamma-gaining advantages in society and the economy. Cancer is difficult to treat because it affects several systems and has one of the highest mortality rates in the modern day. Therefore, research on the disease with an emphasis on the many components of the immune system is vital for better cancer management. This study also creates a fresh mathematical model by adding anti-PD-L1 and IL-10 into the equation to increase CD8+T cell output both quantitatively and qualitatively (see Fig. 1). Therefore, a first-of-its-kind system which follows [24] is used to examine the interaction among CD8+T lymphocytes, cancer cells, IL-10, and anti-PD-L1.

$$\frac{dT}{dt} = a + bI10CT\left(1 - \frac{T}{p}\right) - cT, \tag{1}$$

$$\frac{dC}{dt} = dC(1 - Cq) - eCTI10 - zCTZ, \quad (2)$$

$$\frac{dI_{10}}{dt} = -fI_{10}, \quad (3)$$

$$\frac{dZ}{dt} = -\gamma Z. \quad (4)$$

The parameters T, C, I, and Z represent the CD8+T lymphocytes, cancer cells, IL-10 cytokine, and anti-PD-L1 antibodies, respectively. The symbols a and b represent the initial density of CD8+T cells and the reproduction rate of CD8+T cells due to IL-10, respectively. The parameters c and p denote the ratio of CD8+T cell death and the ratio of tumour growth, respectively. The symbols q and e represent the carrying capacity of cancer cells. The parameters e and z represent the ratio of death of cancer cells due to the anti-PD-L1 antibodies.

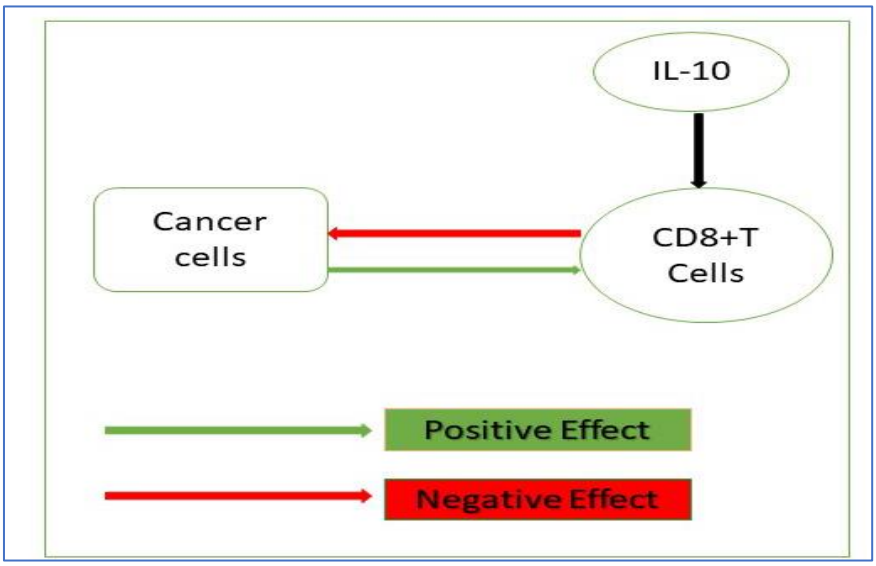


Figure 1. IL-10 interleukin and anti-PD-L1 inhibitor interaction with CD8+T lymphocytes in cancer cells.

Analysis of the system:

We can write inequality (1) as a kind of equation.

$$-cT \leq \frac{dT}{dC}.$$

We then obtained

$$T(t) = T_0 e^{-\alpha}.$$

T (0), an integral constant with a positive value, serves as the T cells' initial value. As t approaches infinity, the result is T (t) > 0. T (t) afterward is always positive. It is written as an instance of an inequality derived from equation (2).

$$-\mathrm{eCTI10} - \mathrm{zCTZ} \leq \frac{dC}{dt}.$$

And get

$$C(t) = C0e - (eTI10 + zTZ)t.$$

Since the limit t causes C (t) > 0, the initial value of cancer cells, C (0), is positive and tends to rise with time. As a result, C (t) always has a positive value. We can be described as an incidence of inequality by applying Eq (3).

$$0 \leq \frac{dI_{10}}{dt}.$$

Then, I obtained I10 > 0. From Eq. (4)

$$\frac{dZ}{dt} = -\gamma Z.$$

We write it as an inequality

$$0 \leq \frac{dI_{10}}{dt},$$

and get Z > 0. So, T (t), C (t), I10(t), and Z(t) are always positive numbers. The system's solution is shown to be positive by the equations (5) through (11) before them. In this work, we solve a set of ODEs and locate the system's solution at 1001 different locations. The carrying capacity of cancer cells (q) and the tumor growth ratio (d), both of which are given in the problem, are the two parameters that are the subject of this discussion. Additionally, the issue is resolved using neural networks, and the outcomes are evaluated against the reference solution using various rotational parameter values.

4. | DESIGN FRAMEWORK AND NUMERICAL/COMPUTATIONAL METHODOLOGY

The computational procedure involves four stages: (1) numerical solution of the four-dimensional cancer dynamics ODE system; (2) construction of the reference data set; (3) supervised learning of LMB-NN according to the reference trajectories; and (4) testing of the obtained neural network by using numerical points that are not utilized during the training stage. For obtaining 1001 reference points in the range of t ∈ [0,200], the set of ODEs is numerically calculated via Mathematica ND Solve under d=0.4, d=0.6, and d=0.8 (family I) and q=1.5, q=1.7, and q=2.0 (family II). These values of d and q are assumed as sensitivity scenarios in the model parametrization, but not as calibrated values from the experiments. The reference data set consists of 70%, 15%, and 15% train, validate, and test sets. The inputs and outputs of the network include one scalar input (time t) and four scalar outputs T, C, I10, and Z, respectively. As already noted, the above MATLAB realization of the LMB-NN architecture uses one hidden layer having 25 neurons. The above statement "four hidden layers" needs to be corrected as: one input node, one hidden layer having 25 neurons, and four output nodes.

- The training fraction is seventy percent; the fractions used for validation and testing are fifteen percent of the reference points each.
- The testing set contains 15% of the reference points and is only needed for evaluation of numerical trajectory reconstruction.

- The validation set will help in keeping track of the performance of the neural network and contains 15% of the reference points.

The number of hidden neurons is set to be equal to twenty-five, like in the discussed test case. This is a design decision in the algorithm because hyperparameter optimization is not performed here. The first thing that happens is the generation of reference numerical trajectories that are processed by the MATLAB Neural Network Toolbox. In LMB-NN, the task is to learn a non-linear function of time to the four state variables, while the reference numerical trajectories are generated using RK4/NDSolve.

Table 1. Every case and scenario mentioned in the paperwork

Scenarios	Cases	Parameters
S1	C1	d=0.4
	C2	d=0.6
	C3	d=0.8
S2	C1	q=1.5
	C2	q=1.7
	C3	q=2.0

5. | MATHEMATICAL PRELIMINARIES

5.1. Network Mapping and Notation

The scalar input time variable is represented by $x(t) = t$, and the four-dimensional state vector is

$$y(t) = [T(t), C(t), I_{10}(t), Z(t)]^T.$$

A nonlinear mapping from the time input to the four state outputs is defined by the LMB-NN:

$$\hat{y}(t;\theta): \mathbb{R} \to \mathbb{R}^4.$$

Here, the network's trainable weights and biases are gathered by θ. The four state variables at each of the N reference time samples are included in the target matrix. The vector residual is equal to

$$(e(\theta) = vec(y_{\text{ref}} - \hat{y}(\theta)) \in \mathbb{R}^{4N}).$$

If P is the total number of trainable parameters, then

$$g = J^T e \in \mathbb{R}^P$$

and the Jacobian is

$$J = \frac{\partial e}{\partial \theta} \in \mathbb{R}^{4N \times P}.$$

Therefore,

$$MSE(\theta) = \frac{1}{4N} \|e(\theta)\|_2^2.$$

Although the training objective provided here is the joint four-output MSE, individual-variable MSE values can also be obtained by restricting e to the corresponding component. The state vector y(t) is four-dimensional, although each state component T, C, I10, and Z is scalar.

$$\Delta\theta = -(J^T J + \mu I)^{-1} J^T e$$

is the Levenberg–Marquardt update, and

$\theta_{\text{new}} = \theta + \Delta\theta$, where μ>0 is the damping value and $I \in \mathbb{R}^{P\times P}$is the identity matrix.

When $J^T J + \mu I$

is nonsingular, the update is numerically well-defined; conditioning and numerical precision are required for the positive damping term to regularize the normal equations.

The scalar time input is denoted by $x \in R$, the reference state vector by $y \in \mathbb{R}^4$, the network prediction by $\hat{y} \in \mathbb{R}^4$, trainable weight matrices by W, trainable bias vectors by b, the concatenated vector of weights and biases by $\theta \in \mathbb{R}^P$, the stacked residual vector by $e \in \mathbb{R}^{4N}$, the Jacobian by $J \in \mathbb{R}^{4N\times P}$, the least-squares gradient quantity by $g = J^T e \in \mathbb{R}^P$, and the LM damping parameter is μ>0.

5.2. Conditions and Assumptions for the LMB Method.

The problem of nonlinear least squares under supervision with network differentiable output functions and a finite reference dataset is solved by the LMB technique. Both the input and the target dataset are supposed to be finite and well-ordered. The weights and biases are the optimization variables, and the architecture of the network is constant during the whole training phase. The LM step is not the mathematical stability proof of the cancer ODE system, but the optimization step.

5.3. Data Partition.

Seventy percent of the 1001 numerical reference points on t∈ [0,200] are used for training, 15% for validation, and 15% for testing. The test dataset allows evaluation of held-out point reconstruction within the same distribution of numerical data because all three datasets were generated using the same RK4 trajectory distribution. It does not prove extrapolation to any unknown patients, treatments, or biological conditions and is not independent clinical validation.

5.4. Mathematical Analysis

Error measures. Let $y_{\text{ref},i}$ *and* $\hat{y}_i$denote the reference and LMB-NN state vectors at the ith test point. The pointwise absolute error is

$$E_i = \left\| y_{\text{ref},i} - \hat{y}_i \right\|_2.$$

For all four outputs and N evaluation points,

$$MSE = \frac{1}{4N}\sum_{i=1}^{N}\sum_{j=1}^{4}\left(y_{\text{ref},ij} - \hat{y}_{ij}\right)^2,$$

$$RMSE = \overset{MAE=\frac{1}{4N}\sum_{i=1}^{N}\sum_{j=1}^{4}\left|y_{\text{ref},ij}-\hat{y}_{ij}\right|}{\sqrt{MSE}}.$$

When a maximum error is reported,

$$MaxE = \max_{i,j}\left|y_{\text{ref},ij} - \hat{y}_{ij}\right|.$$

This is a completely numeric error reconstruction metric; relative errors if normalization relative to some quantity is employed. Neural network approximation converges. The aim of training and training-validation and test MSEs provides a means to assess the convergence numerically.

Although regression and error histograms provide additional evidence for similarity to the numeric reference trajectory, the MSEs provided here reach about 10-12 to 10-15 in several cases provided above. The similarity of training and validation errors means there was no divergence between training and validation in this particular case because of the current data distribution in tests. This is an empirical way of assessing convergence specific to a particular case. Positive analysis provides knowledge of the non-negativity of state variables for the current initial conditions and parameters. The reference trajectories are obtained utilizing the identical RK4 output grid with $\Delta t=0.02$. The current investigation does not make any assertions regarding the full stability analysis of all equilibria via Lyapunov, bifurcation, or eigenvalue approaches. A full stability analysis of equilibrium states would involve the determination of equilibria for the four-dimensional nonlinear system and the analysis of the Jacobian matrix for the system at each parameter value setting. This can be viewed as a limitation and not as a calculation result not done Reference-solution Limitation. The measured LMB-NN errors of approximation are the errors of approximation of the numerical reference solution by the neural network generated from the RK4/NDSolve data generation process. Hence, the results should not be interpreted as an independent analysis of the RK4 discretization errors or any proof of higher stability/accuracy/efficiency of LMB-NN relative to RK4.

6. | RESULTS AND COMPUTATIONAL VALIDATION

In relation to each of the six parameter scenarios that were evaluated, the outcomes measure the level at which LMB-NN can replicate the reference trajectories that are produced through the Runge-Kutta method. 70% of the total of 1001 points within the range of $t \in [0,200]$ of reference data were used for training, 15% for validation, and 15% for testing. The extremely low MSE values in table 09 indicate a high degree of numerical agreement with the reference trajectories. However, such low errors cannot confirm that LMB-NN is more precise, stable, and effective than RK4/NDSolve since it is through the same RK4-generated trajectories that the objectives for training and held-out testing are set.

The reference data (LMB) of neural networks are generated via the Runge-Kutta numerical method while varying these two parameters. There are several graphs in these, namely fitness, performance, state transition, error histogram, and regression graphs. As indicated earlier, the solution input varies between 0 and 200 at a constant step size of 0.02 using the NDSolve command in the Mathematica environment. There were 1001 reference points, of which 70% were allocated to training, and 15% to testing and validation. The LMB-designed neural network is depicted in Figure 2 below, and the presented problem was solved using the LMB neural network approach.

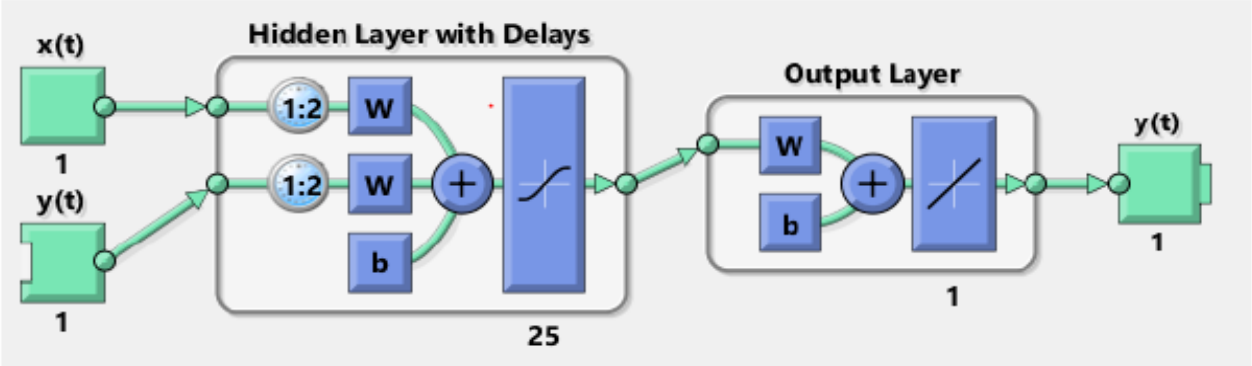


Figure 2. LMB designed the neural network structure of the given model.

Figures 3–17 show the neural network for nonlinear systems that LMB created. Figures 6 through 11 show the results of the mentioned model (1-4) employing the performance and transition states. The performance graphs for the proposed model show the mean squared error (MSE) values for training, testing, best curve, and validation for each application. The network's best performances for (1) at epochs 1000, 1000,1000,1000,1000, and 1000 lie around 10^{-12}, 10^{-11}, 10^{-10}, 10^{-10}, 10^{-13}, 10^{-10}, respectively.

Similarly, for other equations the network's best performances at epochs 1000, 1000,1000, 1000, 1000, and 1000 lies around 10^{-14}, 10^{-12}, 10^{-09}, 10^{-10}, 10^{-14}, 10^{-12} and epochs 167,1000,1000,1000,572,1000 were 10^{-15}, 0^{-14}, 10^{-14}, 10^{-12}, 10^{-13},10^{-11}. The step size (Mu) of the designed LMB process and the gradient-based values for the specified system [7.1583×10^{-16}, 6.129×10^{-06}, 2.0763×10^{-05}, 1.4558×10^{-05}, 3.8017×10^{-05}, 3.1448×10^{-05}] while for other scenarios are [9.6881×10-07, 3.4421×10^{-05}, 8.7595×10^{-05}, 1.3136×10^{-05}] and [8.0548×10^{-08},4.9788×10^{-07},1.7609×10^{-06}, 5.3017×10^{-06},9.9812×10^{-08}, 2.4667×10^{-06}] similarly mu values for all scenarios are [10^{-08}, 10^{-07}, 10^{-06}, 10^{-06}, 10^{-09}, 10^{-06}], [10^{-09}, 10^{-07}, 10^{-05}, 10^{-06}, 10^{-09}, 10^{-08}] and [10^{-10}, 10^{-10}, 10^{-10}, 10^{-08}, 10^{-08}, 10^{-07}] and are shown in all the state transitions figures.

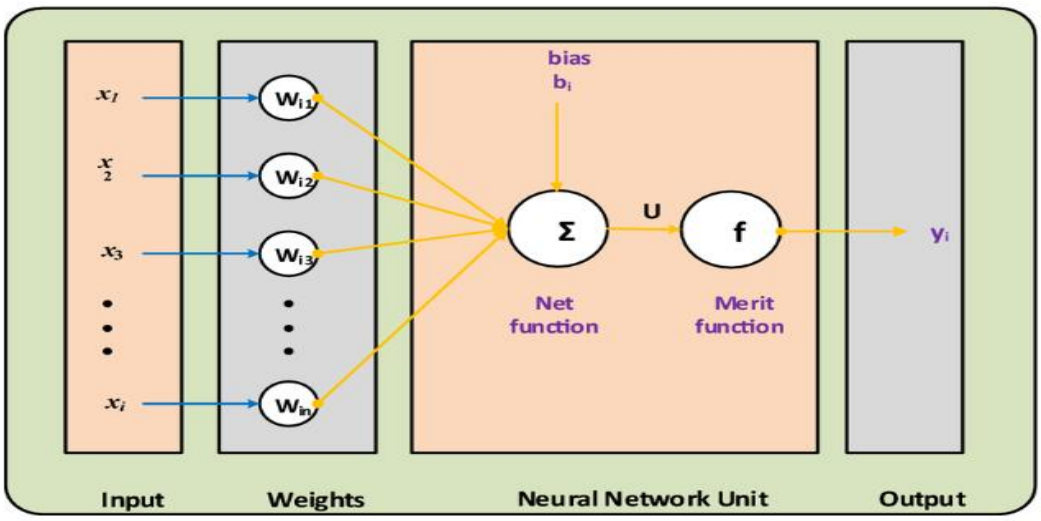
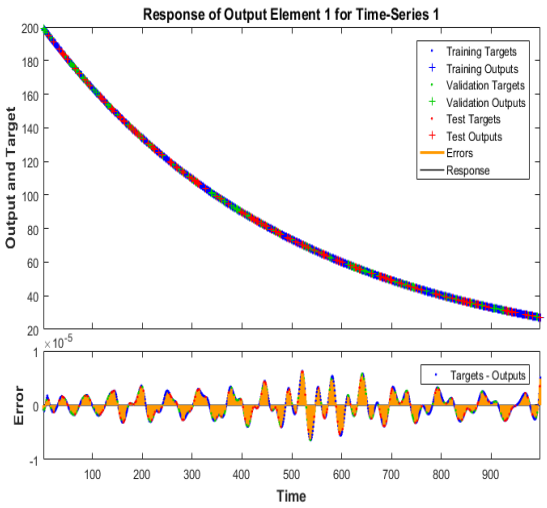


Figure 3. A model system designed based on a single neuron.

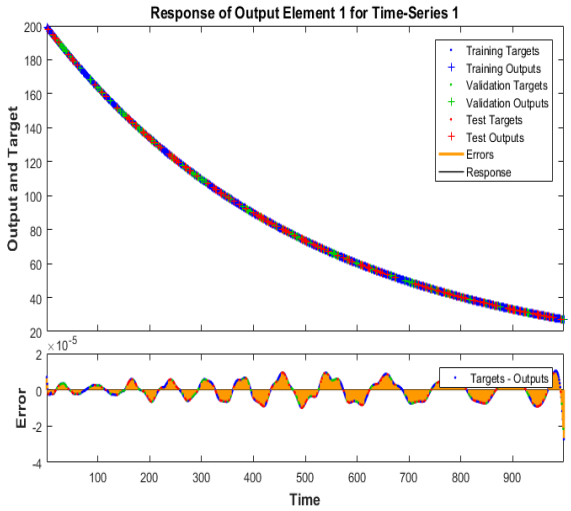
For each system scenario, these graphs show the convergence and precision rates of the LMB neural network. The comparisons between the LMB neural network and the numerical RK4 results for each case of the given model in Figures 6 through 11 attest to the precision of the fitting plots. The system's maximum error values for the LMB neural network vary from 10^{-06} to 10^{-09} when training, testing, and validating inputs are used. However, Figures 12, 13, and 14 show the error histogram values. Figures 15 to 17 in Figure 15 illustrate the regression values for each scenario of the aforementioned systems. The study was created by applying regression analysis after using correlation analyses as a starting point. The modelling, validation, testing, and training were all carried out as effectively as feasible, as seen by the correlation coefficients "R" being close to unity. The regression study in general showed the accuracy of the LMB neural network in solving the model. For the relevant systems, Table 8 demonstrates that convergence values in the MSE sense were also obtained for training, testing, validation, performance, backpropagation, performed epochs, and complexity.

Table 2. The values of the parameters for the (1)–(4) model, where c = cells and h = hour.

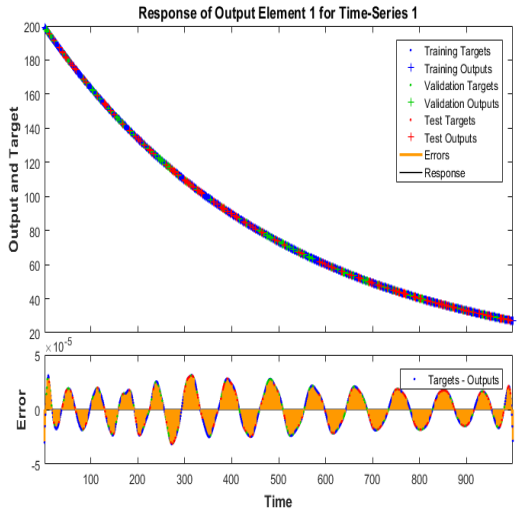
parameters	values (unit)	Reference
a	0.0001 (h−1 c−1 m m3)	[43]
b	0.175 (h−1 c−1 m m3)	estimated
c	0.005 (h−1)	[43]
p	1 (m c−1 m3)	[43]
d	0.02 (h−1 c−1 m m3)	[43]
q	1 (m c−1 m3)	[43]
e	0.15 (h−1)	estimated
z	1 (h−1 c−1 m m3)	estimated
f	0.01 (h−1)	estimated
γ	0.001925 (h−1)	estimated



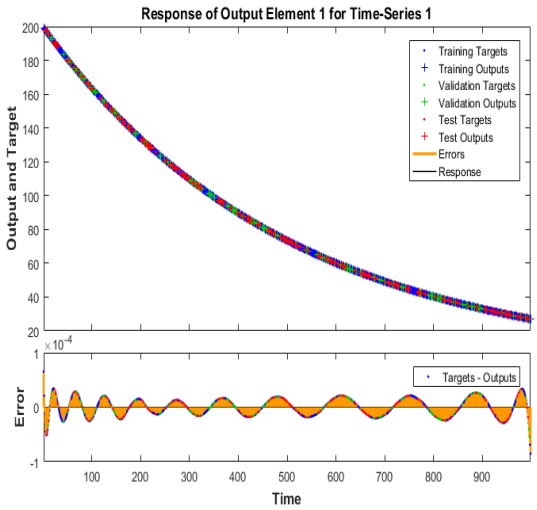
(a) Fitness 1



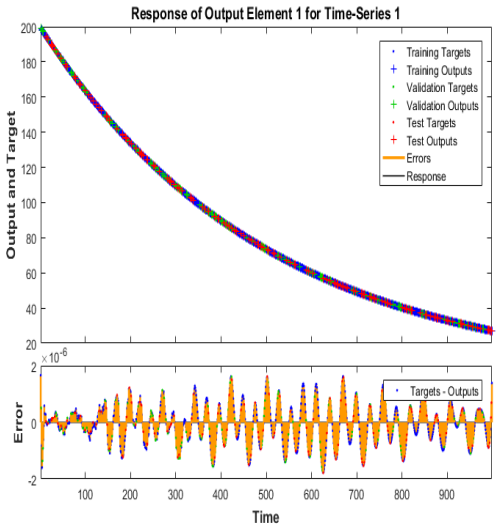
(b) Fitness 2



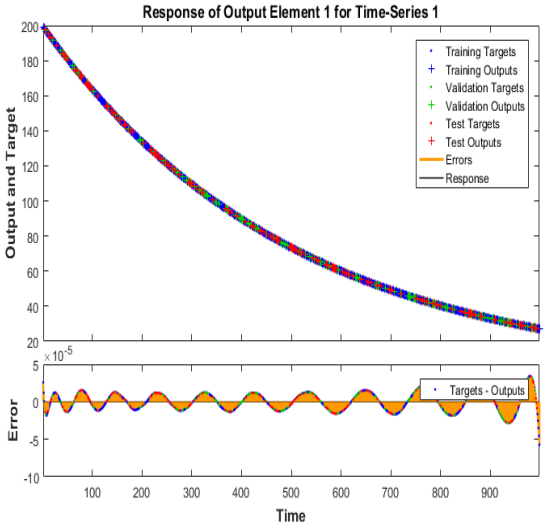
(c) Fitness 3



(d) Fitness 4



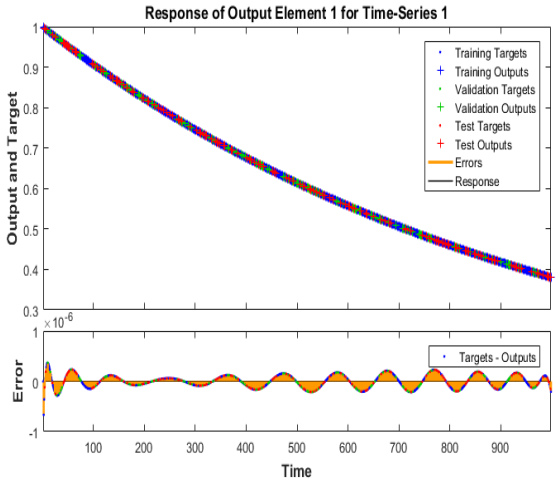
(e) Fitness5



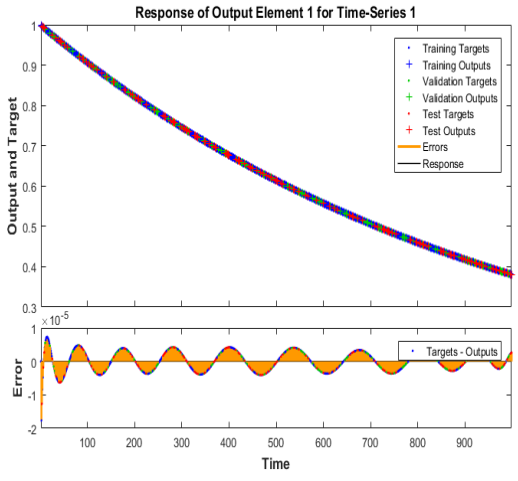
(f) Fitness6

Figure 3. By using the LMB-NN technique, we obtained the Fitness graphs of the IL-10 cytokine (I10)

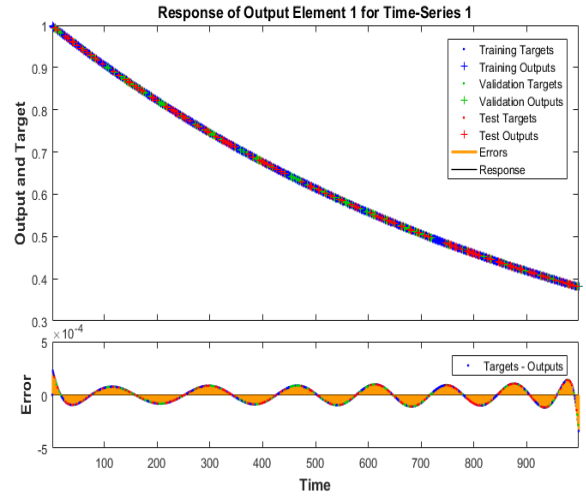
Fitness graphs are obtained according to each value of scenarios 1 and 2.



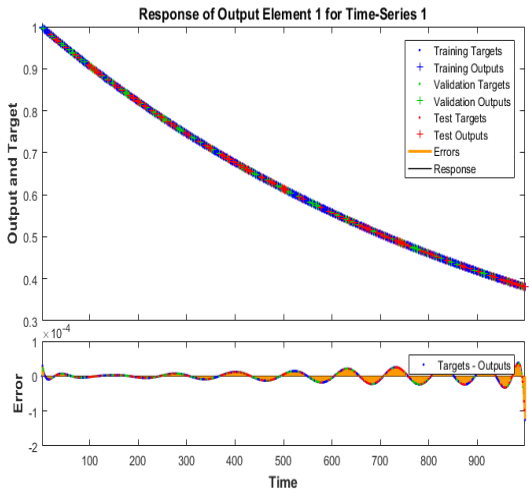
(a) Fitness 1



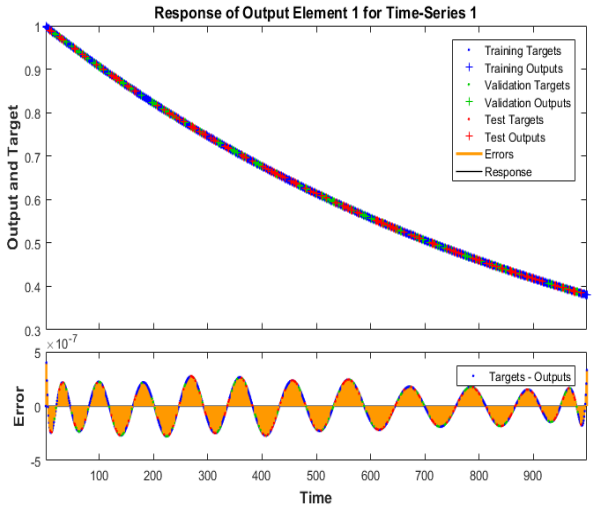
(b) Fitness 2



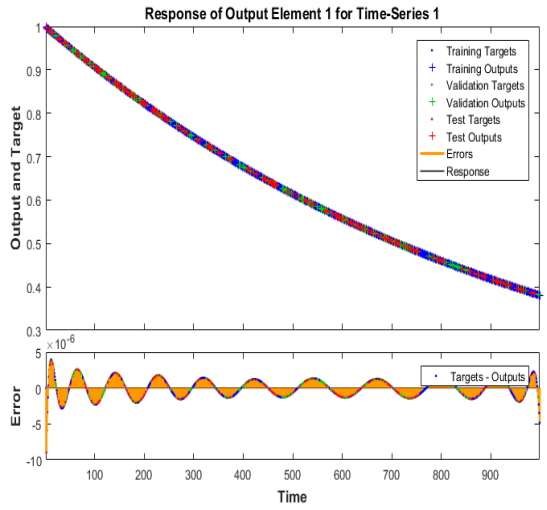
(c) Fitness3



(d) Fitness4

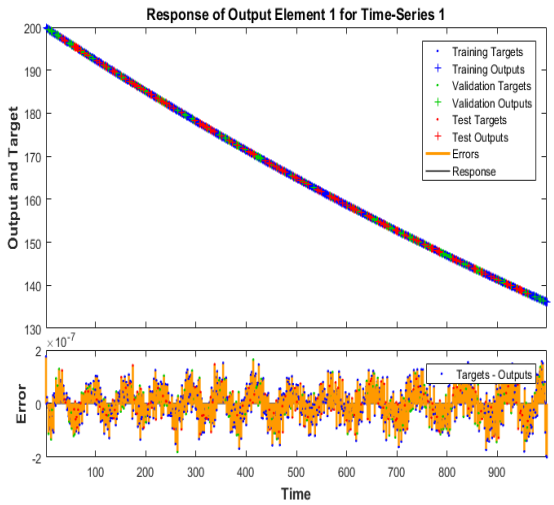


(e) Fitness 5

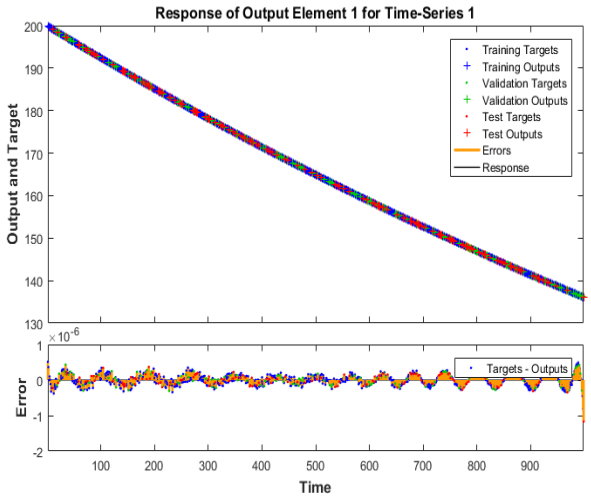


(f) Fitness 6

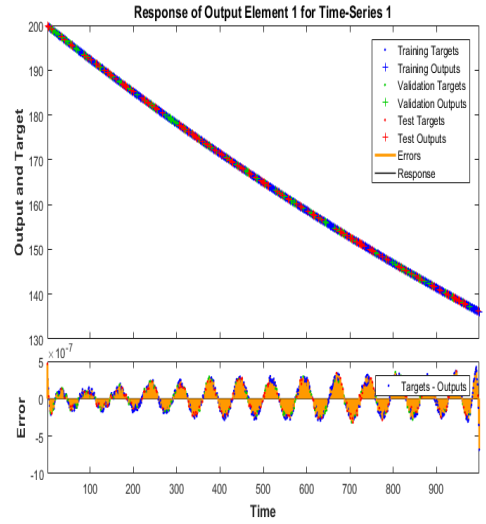
Figure 4. LMB-NN fitness graphs of CD8+T lymphocytes are obtained according to each value, giving Scenarios of table 01 and parameter values of table02.



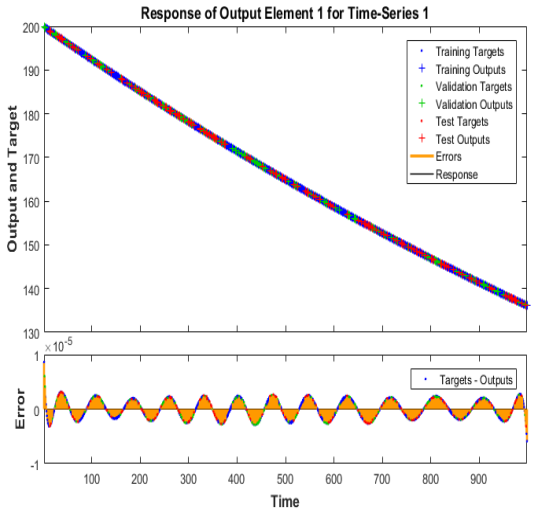
(a) Fitness 1



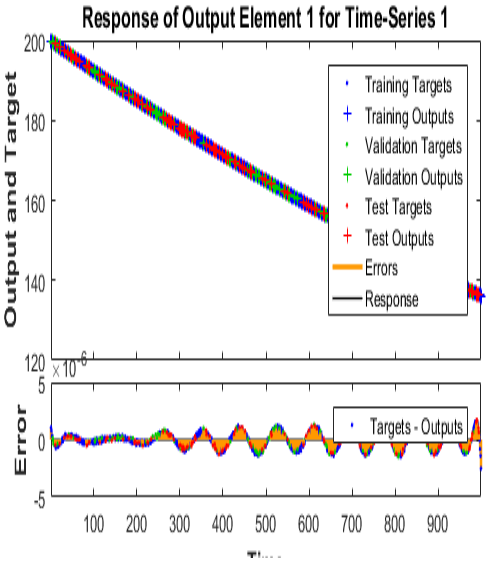
(b) Fitness2



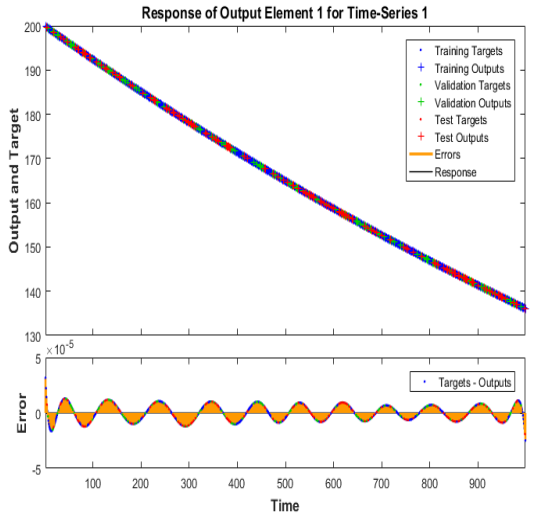
(c) Fitness 3



(d) Fitness 4



(e) Fitness5

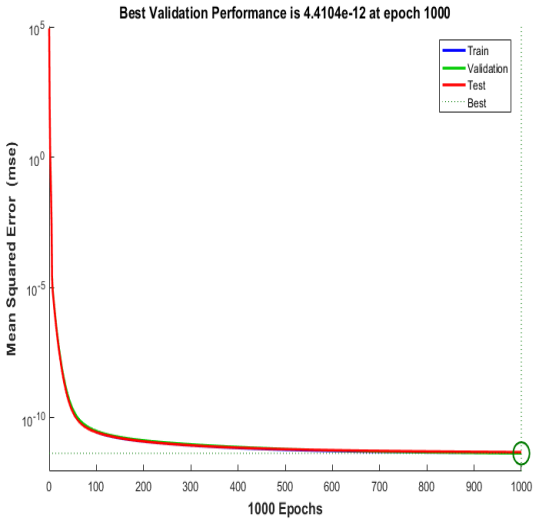


(**f**) Fitness6

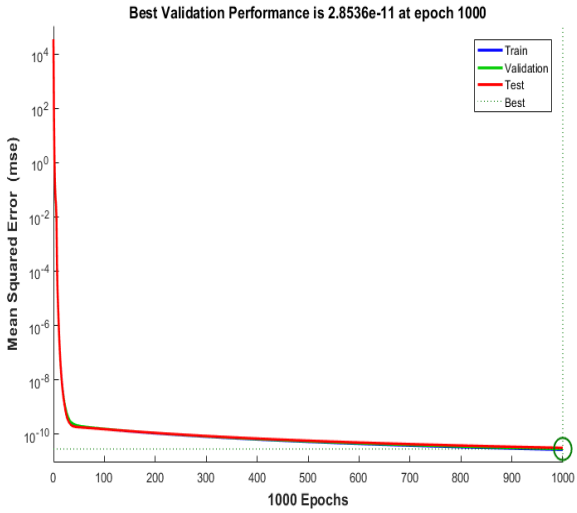
Figure 5. By using the LMB-NN Fitness graph of anti-PD-L1 is obtained according to the varying Parameters of both scenarios as shown in Table 01 and parameters values in Table 02.

Table 3. Comparative analysis of LMB-NN and RK4 method for scenario 01 case 01.

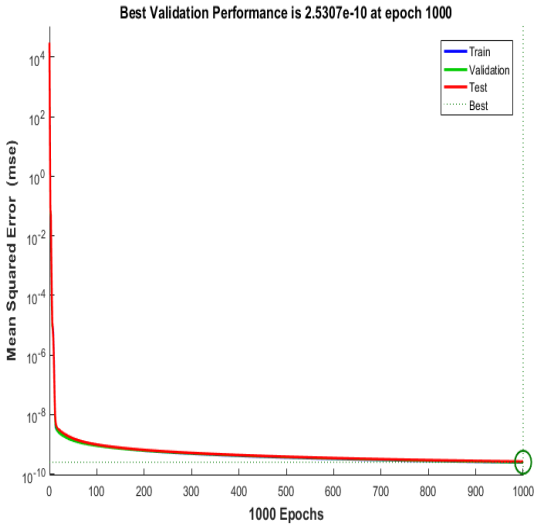
	T[t]			**I[t]**			**Z[t]**		
t	RK4	LMB-NN	Error	RK4	LMB-NN	Error	RK4	LMB-NN	Error
0	200	199.201598	8.12E-07	200	0.99812358	4.03E-10	200	199.846059	1.76E-07
20	163.746148	163.092472	4.37E-07	0.906815	0.90504293	1.15E-07	192.4463	192.298214	4.95E-09
40	134.064002	133.528814	2.95E-06	0.822423	0.8208199	4.26E-08	185.178	185.035439	7.07E-08
60	109.762324	109.324152	1.93E-07	0.746062	0.74461185	8.11E-08	178.1841	178.046965	6.40E-08
80	89.8657913	89.5070457	1.39E-07	0.676968	0.67565593	8.84E-08	171.4544	171.322435	7.38E-08
100	73.5758861	73.2821701	1.88E-07	0.61445	0.61326197	3.13E-08	164.9789	164.851879	5.85E-09
120	60.2388418	59.9983729	4.90E-06	0.55788	0.55680558	2.52E-08	158.7479	158.625704	2.56E-08
140	49.3193952	49.1225087	3.04E-06	0.506694	0.5057218	6.28E-09	152.7523	152.634682	9.91E-08
160	40.379306	40.2181096	1.52E-06	0.460379	0.45949937	1.16E-07	146.9831	146.86993	4.05E-08
180	33.0597802	32.927805	4.11E-07	0.418472	0.41767551	1.40E-07	141.4318	141.322903	3.20E-08
200	27.0670603	27.05407	3.00E-07	0.380552	0.3802451	3.12E-07	136.0901	136.0101	3.34E-07



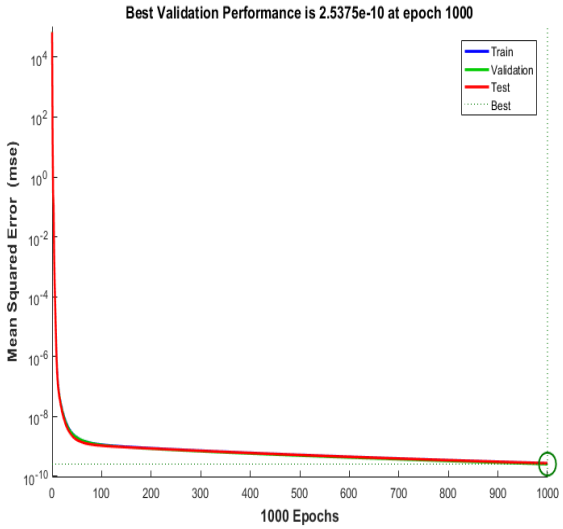
(a) Performance 1



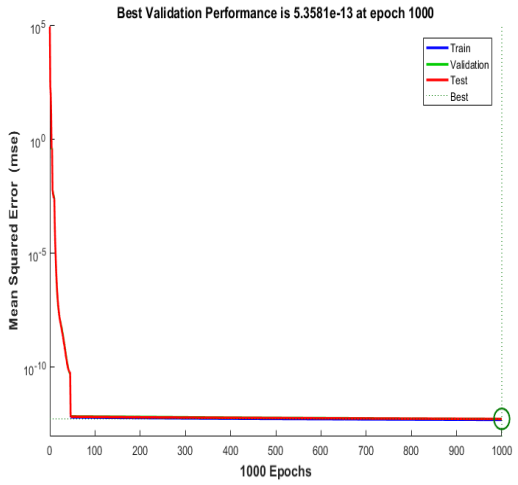
(b) Performance 2



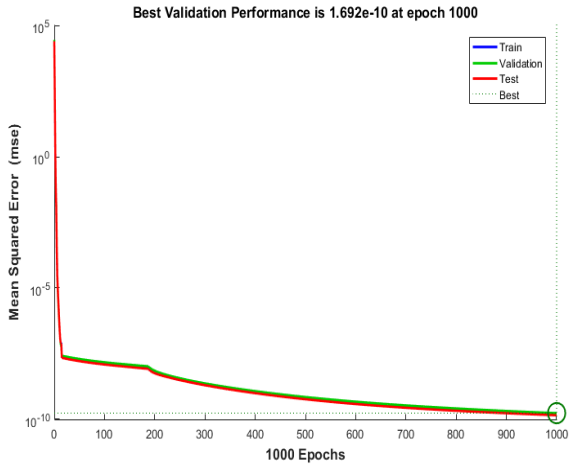
(c) Performance3



(d) Performance



(e) Performance5

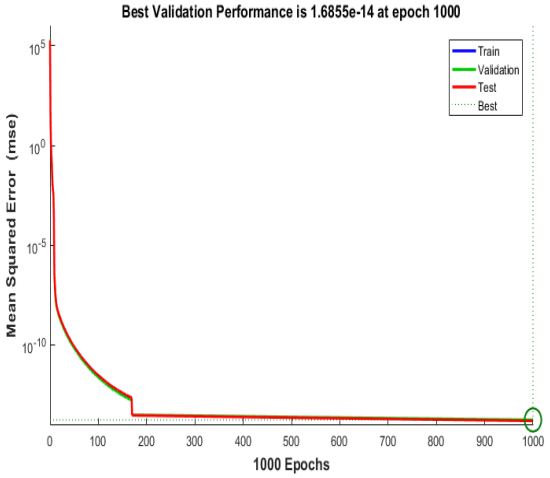


(f) Performance6

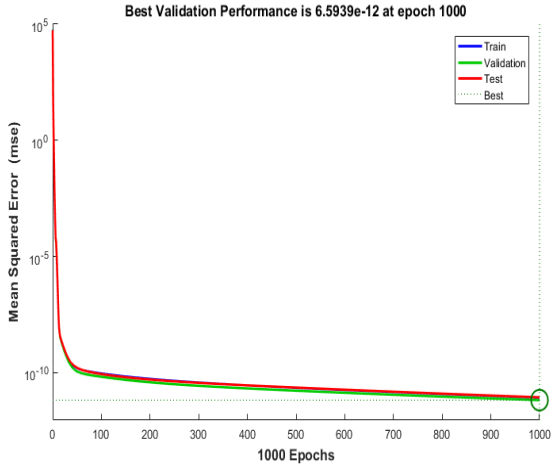
Figure 6. We were able to plot the performance of the IL-10 cytokine (I10) utilizing the LMB-NN. Each graph displays a different value for each scenario of Table 01 and parameters of 02.

Table 4. Comparative analysis of LMB-NN and RK4 method for scenario 1, case 2.

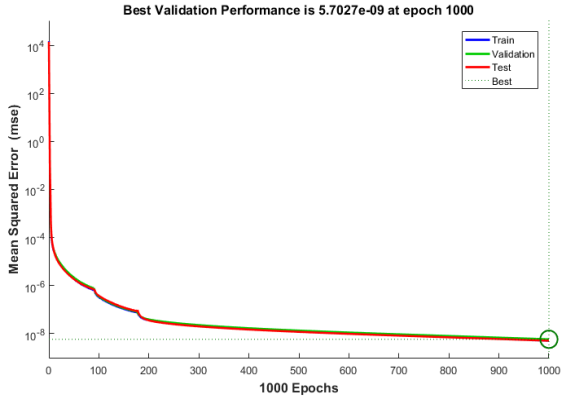
	T[t]			I[t]			Z[t]		
t	**RK4**	**LMB-NN**	**Error**	**RK4**	**LMB-NN**	**Error**	**RK4**	**LMB-NN**	**Error**
0	200	199.2015905	7.01E-06	200	0.9981236	4.03E-10	200	199.84606	5.11E-07
20	163.7461484	163.0924696	2.38E-06	0.906814673	0.9050429	1.15E-07	192.4463409	192.29821	3.44E-07
40	134.0640025	133.528824	6.36E-06	0.822423098	0.8208199	4.26E-08	185.1779707	185.03544	1.47E-07
60	109.7623242	109.3241471	4.81E-06	0.746062444	0.7446119	8.11E-08	178.1841144	178.04697	9.88E-08
80	89.86579151	89.5070511	5.01E-06	0.676968468	0.6756559	8.84E-08	171.4544042	171.32244	1.14E-07
100	73.57588622	73.28217684	6.71E-06	0.614449651	0.613262	3.13E-08	164.9788636	164.85188	1.42E-07
120	60.23884211	59.99837526	6.88E-06	0.557880288	0.5568056	2.52E-08	158.7478932	158.6257	8.74E-08
140	49.31939534	49.12251881	7.05E-06	0.506694071	0.5057203	1.38E-06	152.7522559	152.63468	1.00E-07
160	40.37930585	40.2181071	3.96E-06	0.460379006	0.4594965	2.62E-06	146.9830636	146.86993	2.84E-07
180	33.05978041	32.92779692	8.71E-06	0.418471403	0.4176754	1.67E-07	141.4317638	141.3229	2.41E-07
200	27.06706043	27.03401	2.87E-05	0.380551836	0.380143	2.19E-07	136.0901272	136.0101	3.12E-07



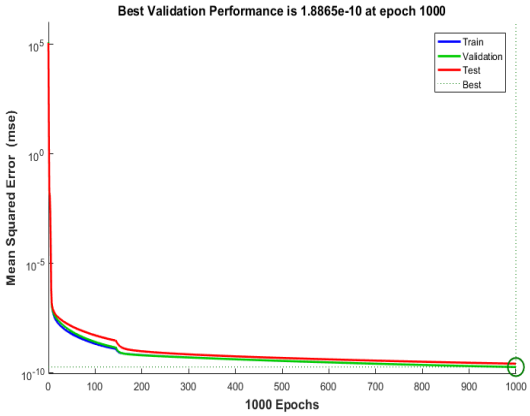
(a) Performance 1



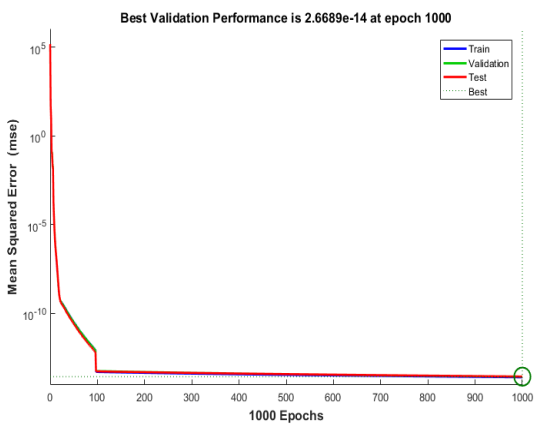
(b) Performance 2



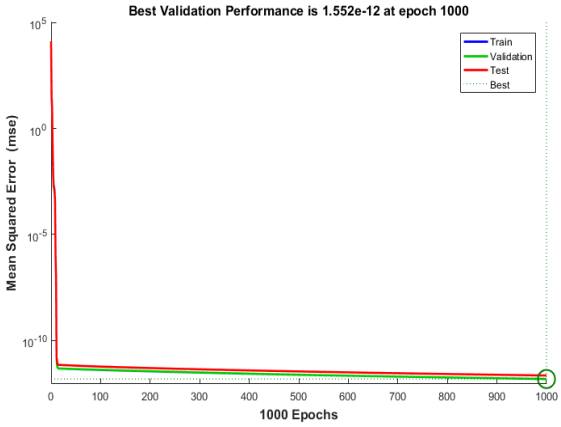
(c) Performance 3



(d) Performance 4

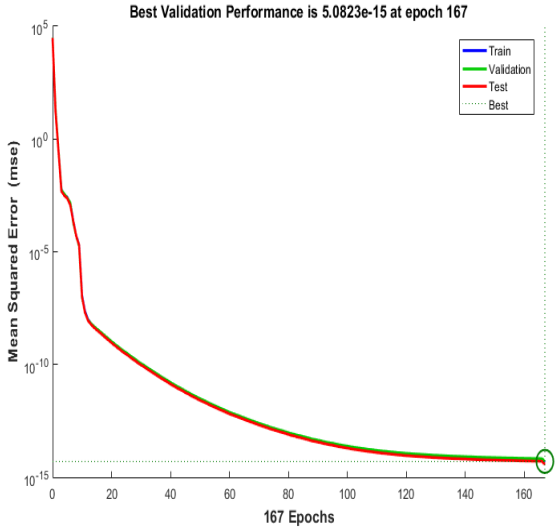


(e) Performance 5

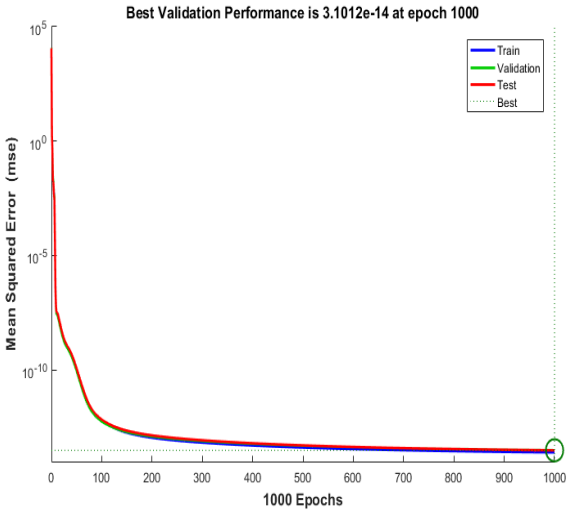


(f) Performance 6

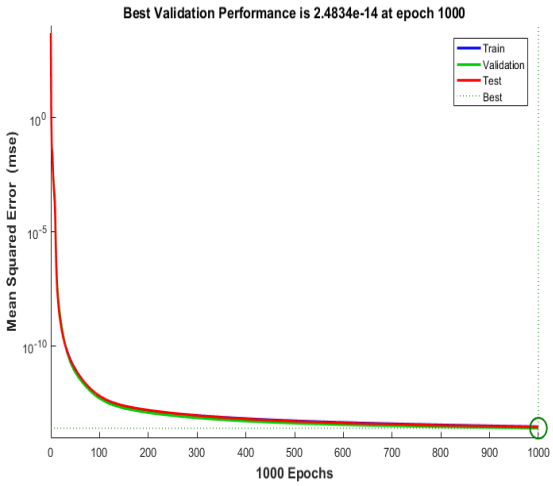
Figure 7. LMB-NN performance graphs of CD8+T lymphocytes utilizing various scenarios' Parameters, as shown in Table 01, and parameter values in Table 02.



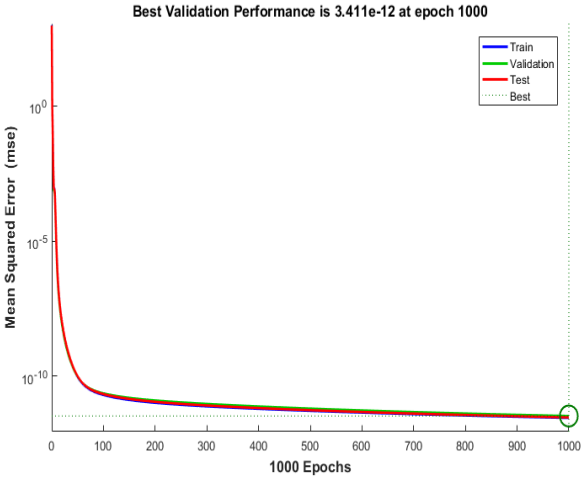
(**a**) Performance 1



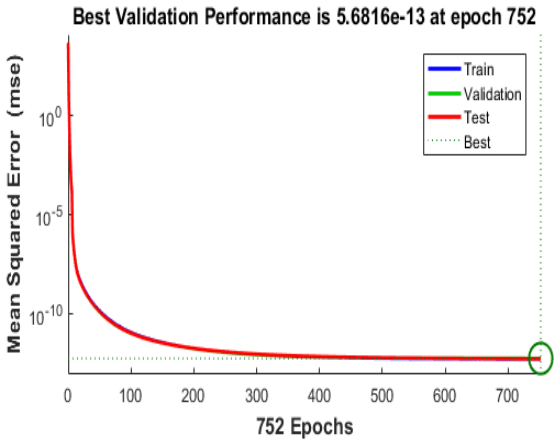
(b) Performance 2



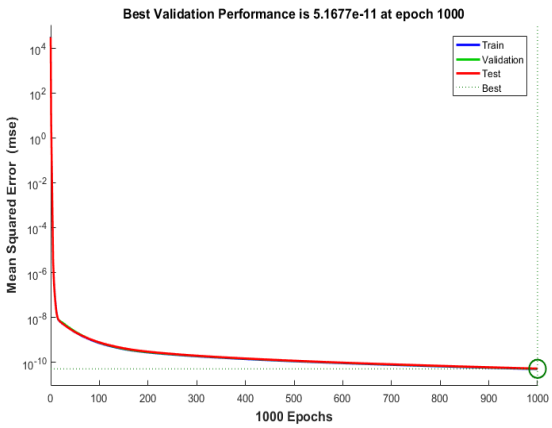
(c) Performance 3



(d) Performance 4



(f) Performance 5



(e) Performance 6

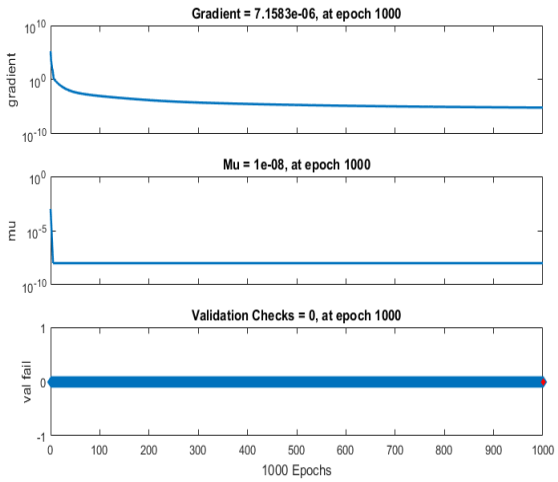
Figure 8. LMB-NN performance graphs of CD8+T lymphocytes utilizing varying Parameters of table 01 and values of table 02.

Table 5. Comparative analysis of LMB-NN and RK4 method for scenario 01 case 03.

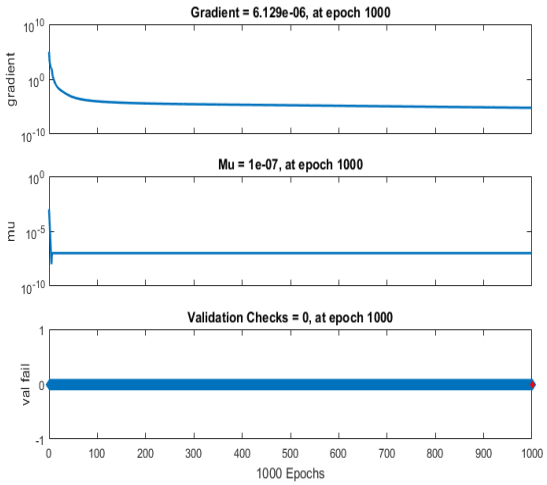
	T[t]				I[t]		Z[t]		
t	**RK4**	**LMB-NN**	**Error**	**RK4**	**LMB-NN**	**Error**	**RK4**	**LMB-NN**	**Error**
0	200	199.2016	3.02E-05	200	0.998123	9.55E-08	200	199.8461	4.66E-07
20	163.746148	163.0925	7.04E-06	0.90681416	0.904974	6.82E-05	192.446341	192.2982	7.67E-08
40	134.064002	133.5288	2.22E-05	0.82242263	0.8209	8.09E-05	185.177971	185.0354	1.67E-07
60	109.762324	109.3241	2.11E-05	0.74606203	0.744526	8.55E-05	178.184114	178.047	1.69E-07
80	89.865791	89.50702	2.27E-05	0.67696809	0.675728	7.24E-05	171.454404	171.3224	1.26E-07
100	73.5758857	73.28216	1.37E-05	0.61444931	0.613254	7.34E-06	164.978864	164.8519	4.49E-08
120	60.2388422	59.99837	5.00E-06	0.55787998	0.55672	8.50E-05	158.747893	158.6257	2.78E-07
140	49.3193952	49.12253	1.70E-05	0.50669393	0.505793	7.13E-05	152.752256	152.6347	2.60E-07
160	40.3793056	40.21813	1.84E-05	0.46037888	0.459582	8.31E-05	146.983064	146.8699	2.15E-07
180	33.0597805	32.92782	1.44E-05	0.41847129	0.417646	2.91E-05	141.431764	141.3229	7.94E-08
200	27.0670601	27.01045	1.17E-06	0.38055173	0.381035	0.000039	136.090127	136.0101	4.50E-06

Table 6. Comparative analysis of the LMB-NN and RK4 methods for scenario 02, case 01.

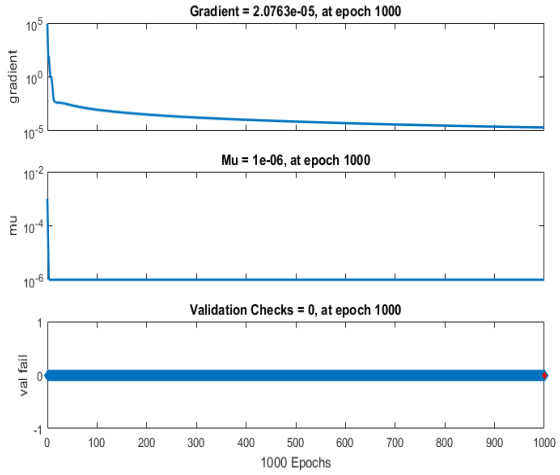
	T[t]			I[t]			Z[t]		
t	**RK4**	**LMB-NN**	**Error**	**RK4**	**LMB-NN**	**Error**	**RK4**	**LMB-NN**	**Error**
0	200	199.2015	6.54E-05	200	0.998123	2.95E-09	200	199.8461	8.64E-06
20	163.746148	163.0925	1.97E-05	0.90681451	0.905045	2.21E-06	192.446341	192.2982	1.94E-06
40	134.064002	133.5288	1.35E-05	0.82242295	0.820824	3.86E-06	185.177971	185.0354	3.65E-07
60	109.762324	109.3242	6.55E-06	0.74606231	0.74461	1.90E-06	178.184114	178.047	2.41E-06
80	89.8657915	89.50705	6.75E-06	0.67696835	0.675644	1.20E-05	171.454404	171.3224	2.47E-06
100	73.5758866	73.28216	9.27E-06	0.61444954	0.613256	5.58E-06	164.978864	164.8519	1.87E-06
120	60.2388417	59.99835	1.88E-05	0.55788019	0.55681	4.01E-06	158.747893	158.6257	1.01E-06
140	49.319395	49.12253	1.35E-05	0.50669412	0.50573	8.21E-06	152.752256	152.6347	8.59E-07
160	40.3793064	40.21813	1.46E-05	0.46037905	0.459505	6.31E-06	146.983064	146.8699	1.94E-06
180	33.0597802	32.92778	2.06E-05	0.41847145	0.417665	9.87E-06	141.431764	141.3229	1.72E-06
200	27.0670603	27.06701	6.32E-05	0.38055187	0.381066	3.20E-04	136.090127	136.0103	4.30E-06



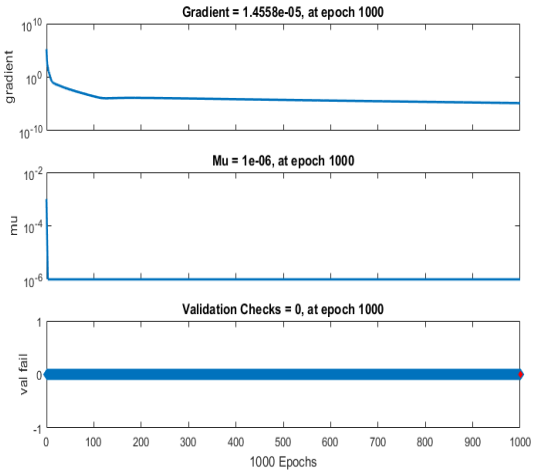
(a) Training state1



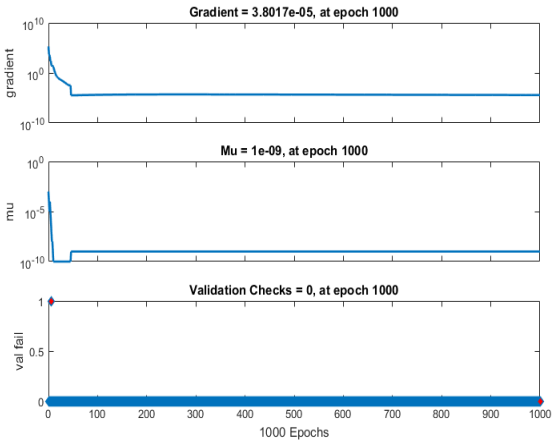
(b) Training state 2



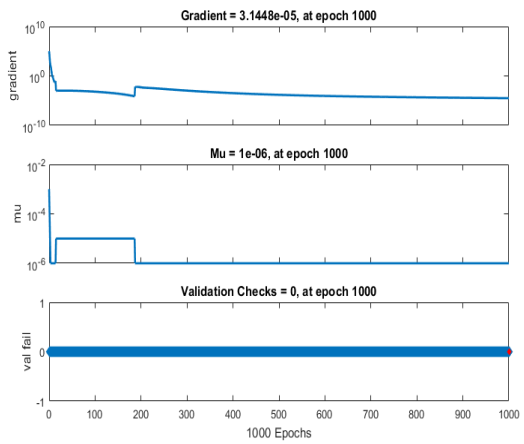
(**c**) Training state 3



(d) Training state 4

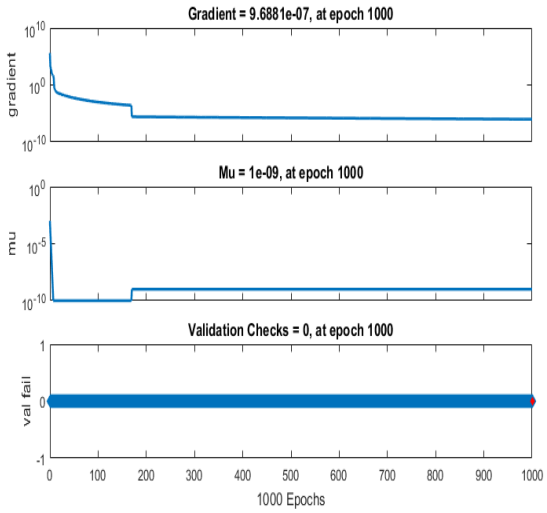


(e) Training state 5

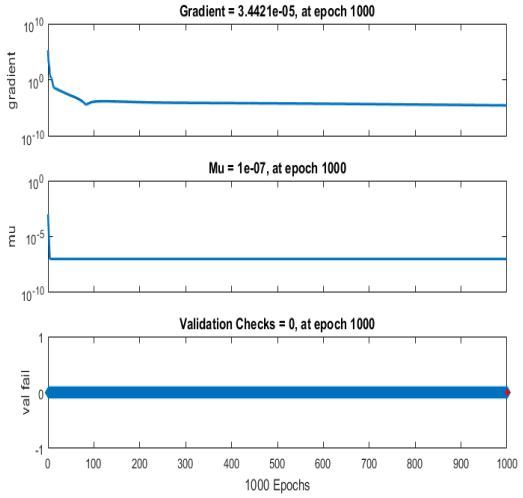


(**f**) Training state 6

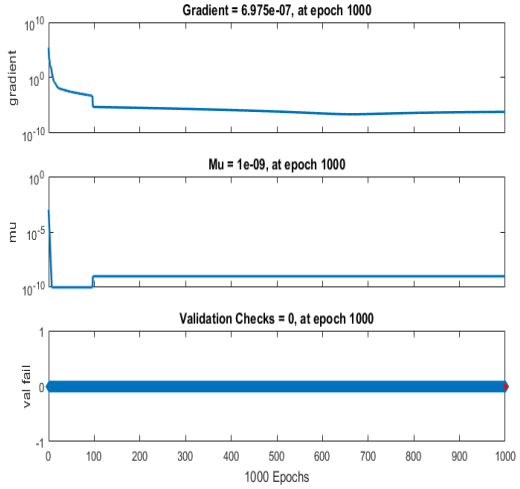
Figure 9. As shown in Tables 01 and 02, the plots of the IL-10 cytokine's training state (I10) were created using LMB-NN with the various parameters of the provided scenarios.



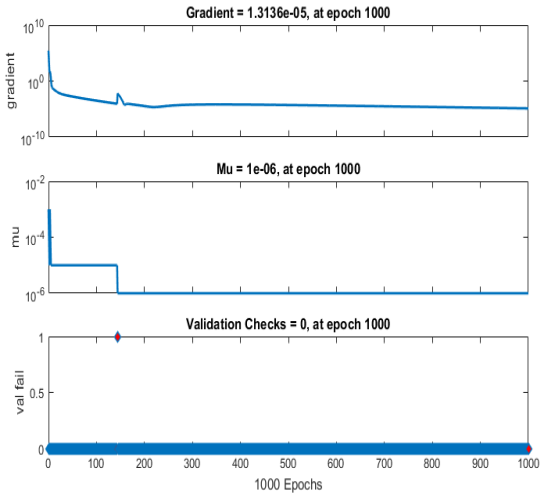
(a) Training state 1



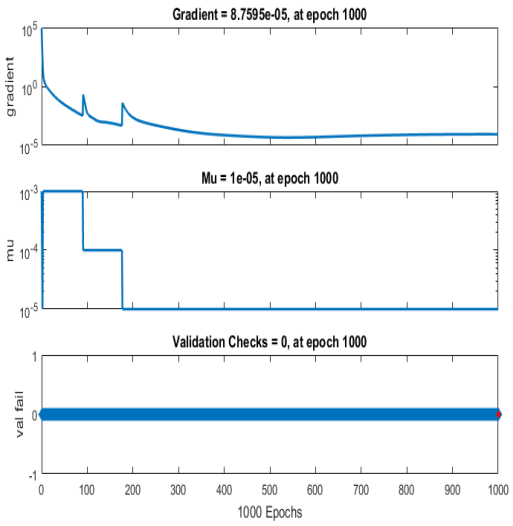
(b) Training state 2



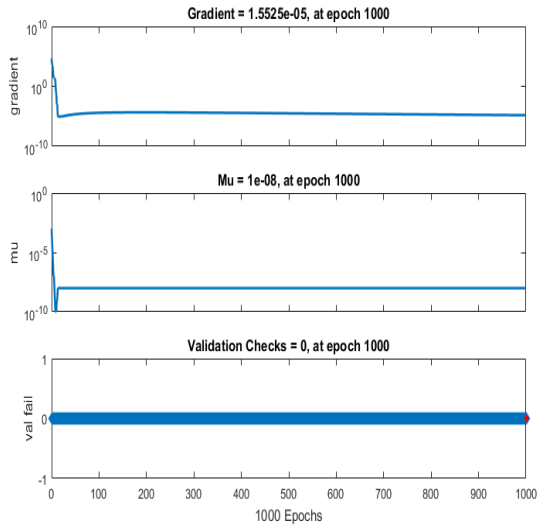
(c) Training state 3



(d) Training state 4



(e) Training state 5



(f) Training state 6

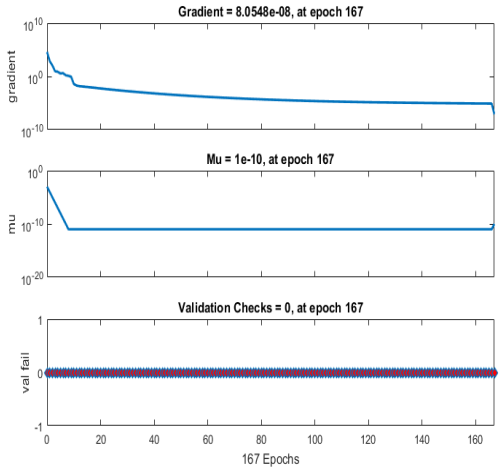
Figure 10. LMB-NN training state graphs of CD8+T lymphocyte cells are obtained according to the parameter values of table02 and varying parameters of table 01.

Table 7. Comparative analysis of LMB-NN and RK4 method for scenario 02 case 02.

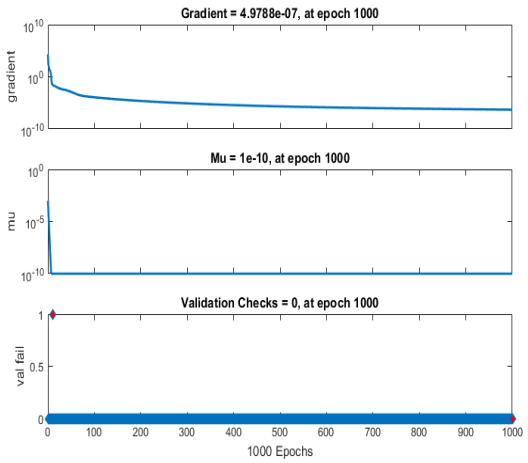
	T[t]			I[t]			Z[t]		
t	RK4	LMB-NN	Error	RK4	LMB-NN	Error	RK4	LMB-NN	Error
0	200	199.201596	1.66E-06	200	0.998123	9.30E-10	200	199.8461	1.08E-06
20	163.7461	163.092472	1.43E-07	0.90681459	0.905043	2.21E-07	192.446341	192.2982	3.27E-08
40	134.064	133.528816	9.36E-07	0.82242303	0.82082	4.86E-11	185.177971	185.0354	2.75E-07
60	109.7623	109.324152	8.90E-07	0.74606238	0.744612	1.54E-07	178.184114	178.047	4.51E-07
80	89.86579	89.5070462	5.59E-08	0.67696841	0.675656	2.53E-07	171.454404	171.3224	1.11E-06
100	73.57589	73.2821712	1.57E-06	0.6144496	0.613262	2.17E-07	164.978864	164.8519	1.90E-07
120	60.23884	59.9983686	6.46E-07	0.55788024	0.556806	1.49E-07	158.747893	158.6257	7.51E-07
140	49.3194	49.1225106	8.91E-07	0.50669417	0.505722	1.48E-08	152.752256	152.6347	1.07E-06
160	40.37931	40.2181121	1.31E-06	0.46037909	0.459499	1.10E-07	146.983064	146.8699	1.16E-06
180	33.05978	32.9278055	1.11E-07	0.41847148	0.417675	1.10E-07	141.431764	141.3229	6.09E-07
200	27.06706	27.06701	5.40E-07	0.38055191	0.380143	1.23E-07	136.090127	136.091	2.60E-06

Table 8. Comparative analysis of the LMB-NN and RK4 methods for scenario 02, case 03.

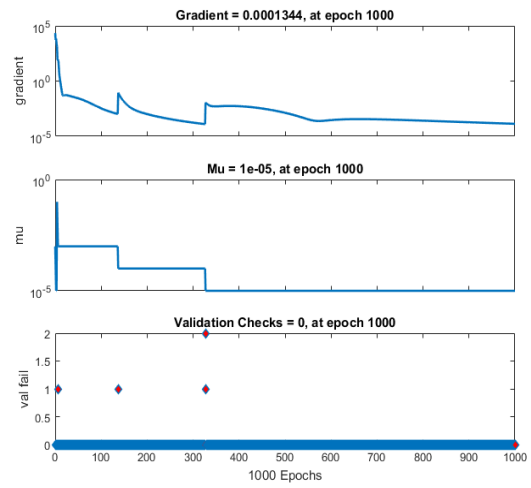
	T[t]			**I[t]**			**Z[t]**		
t	**RK4**	**LMB-NN**	**Error**	**RK4**	**LMB-NN**	**Error**	**RK4**	**LMB-NN**	**Error**
0	200	199.201628	3.05E-05	200	0.998124	5.60E-08	200	199.846	3.17E-05
20	163.746148	163.092464	7.15E-06	0.90681469	0.905045	2.30E-06	192.446341	192.2982	3.87E-06
40	134.064002	133.528839	2.23E-05	0.82242311	0.820821	7.84E-07	185.177971	185.0354	5.00E-06
60	109.762324	109.32413	2.12E-05	0.74606246	0.744611	4.35E-07	178.184114	178.047	1.05E-05
80	89.8657913	89.5070241	2.20E-05	0.67696848	0.675655	3.62E-07	171.454404	171.3224	9.39E-06
100	73.5758863	73.2821579	1.23E-05	0.61444966	0.613263	5.99E-07	164.978864	164.8519	3.92E-06
120	60.2388416	59.9983721	4.33E-06	0.5578803	0.556807	1.29E-06	158.747893	158.6257	3.50E-06
140	49.319395	49.1225281	1.64E-05	0.50669422	0.505722	7.37E-08	152.752256	152.6347	5.46E-06
160	40.3793063	40.218131	1.96E-05	0.46037914	0.459498	1.36E-06	146.983064	146.8699	7.11E-06
180	33.0597802	32.9278193	1.41E-05	0.41847153	0.417674	1.15E-06	141.431764	141.3229	7.79E-06
200	27.0670603	27.0681	1.76E-06	0.38055195	0.38014	2.78E-06	136.090127	136.1089	3.76E-05



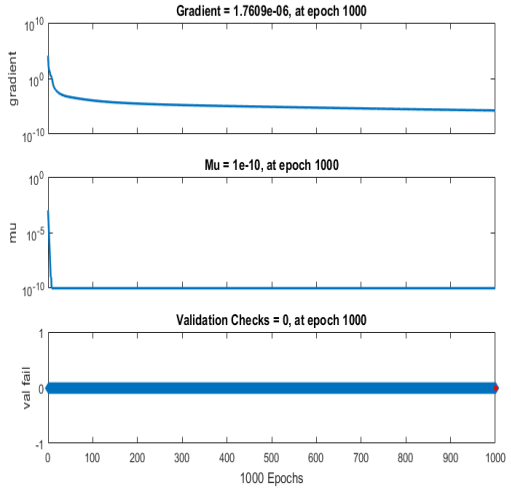
(a) Training state 1



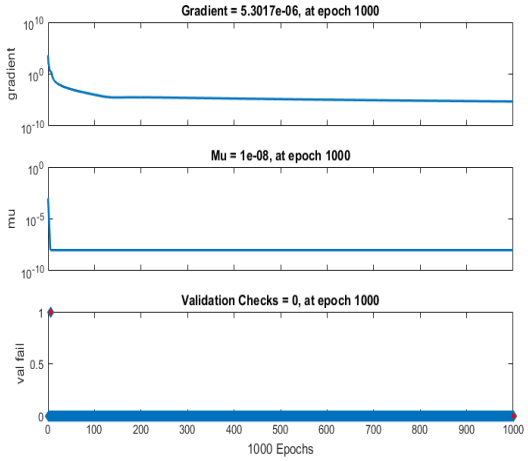
(b) Training state 2



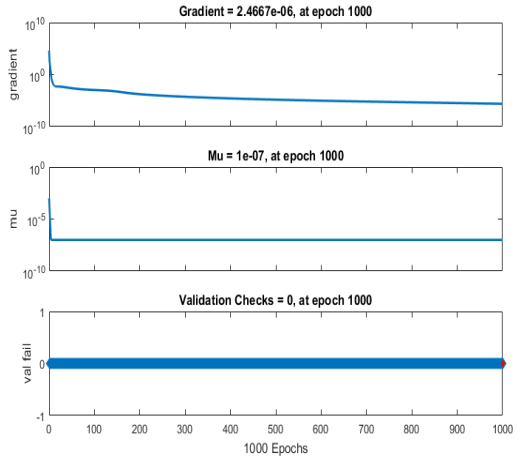
(c) Training state 3



(d) Training state 4

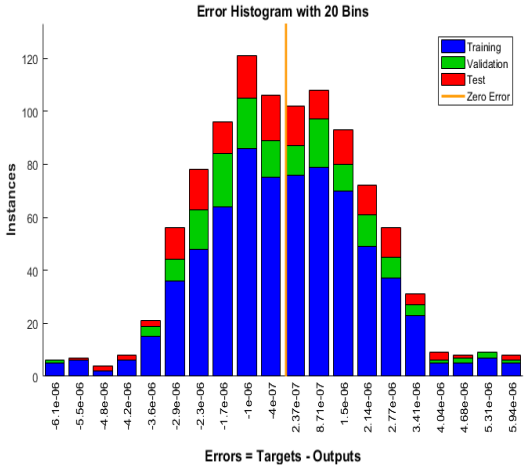


(e) Training state 5

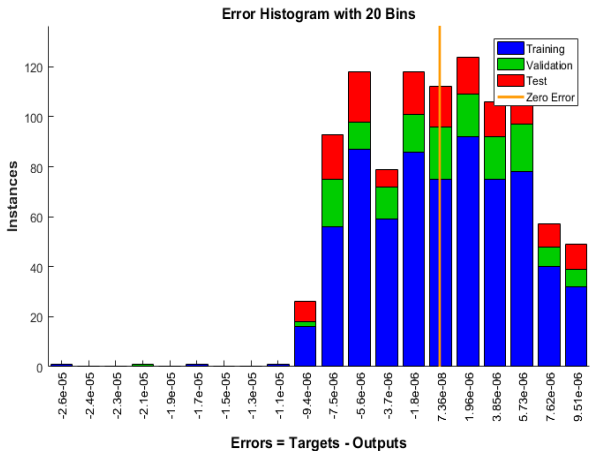


(f) Training state 6

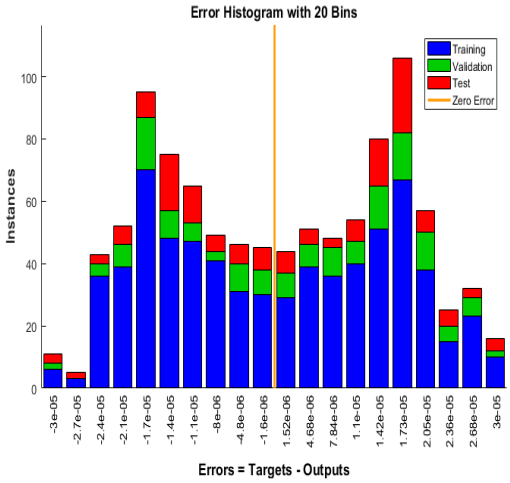
Figure 11. According to the scenarios shown in Table 1 and the values in Table 02, anti-PD-L1 cell state graphs are created using LMB-NN training.



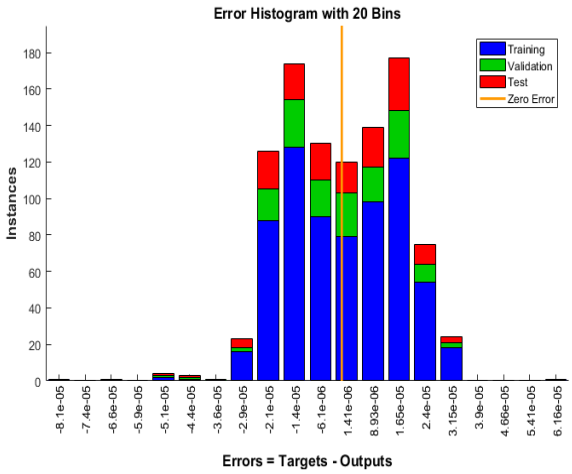
(a) Error histogram1



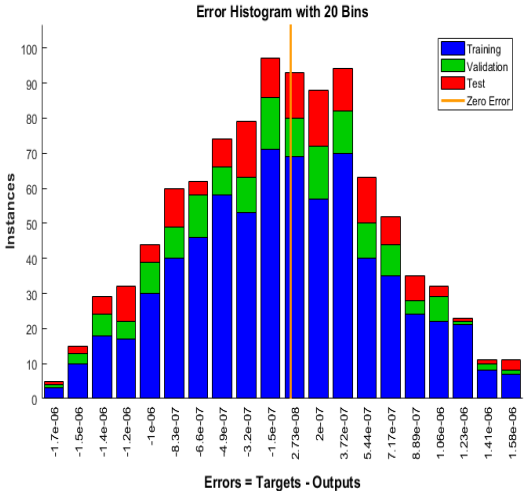
(b) Error histogram2



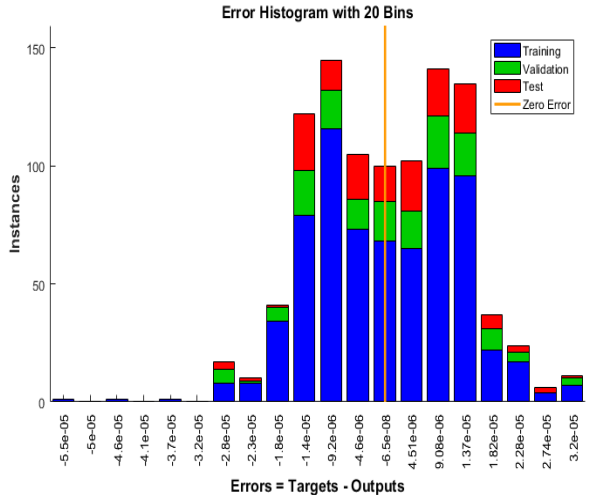
(c) Error histogram 3



(d) Error histogram 4

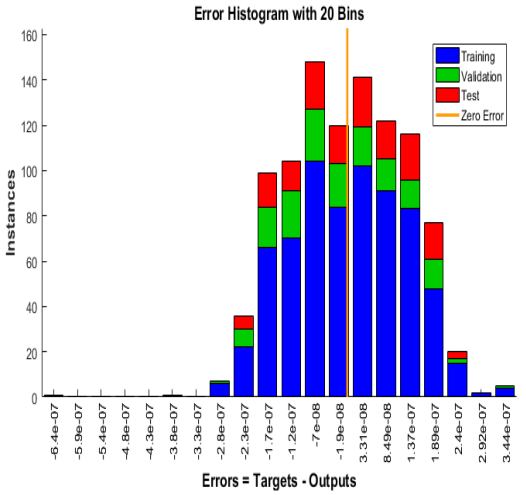


(e) Error histogram 5

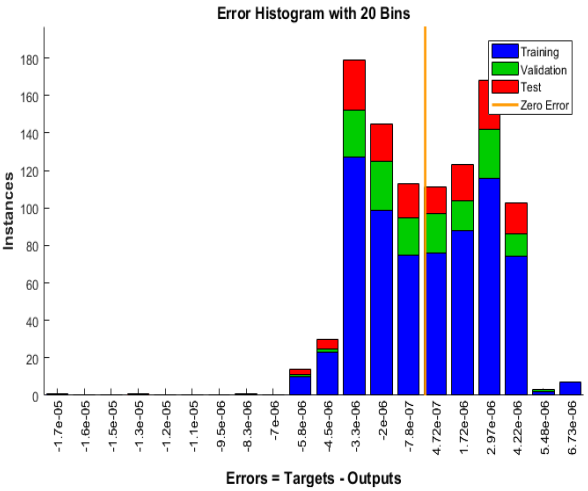


(f) Error histogram 6

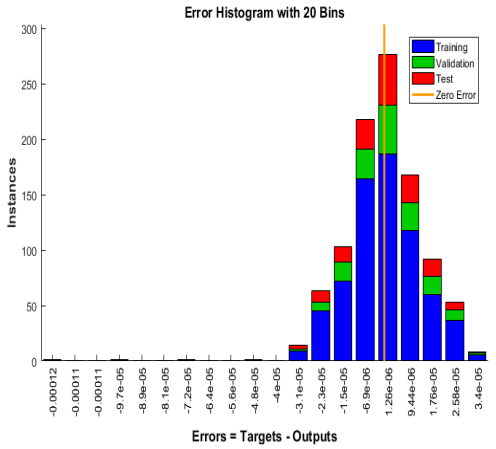
Figure 12. According to the parameter values in Table 2 and Table 1, the LMB-NN algorithm is used to generate the error plots of the IL-10 cytokine (I10).



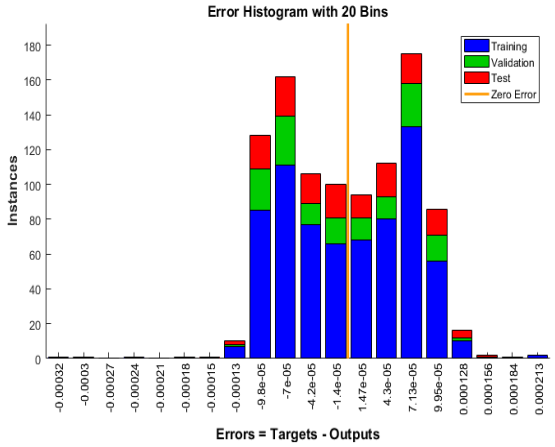
(a) Error histogram 1



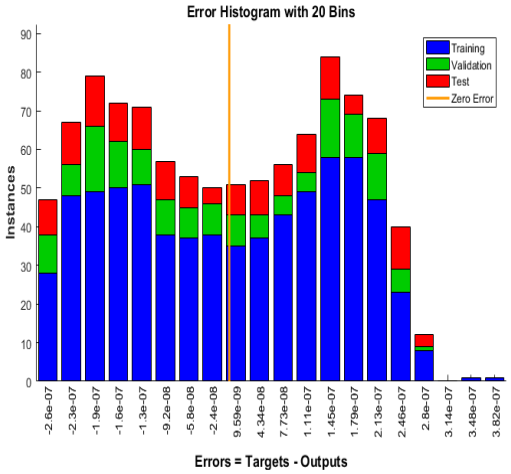
(b) Error histogram 2



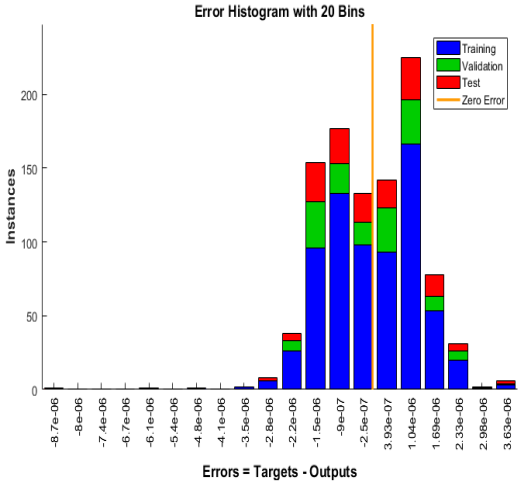
(c) Error histogram 3



(d) Error histogram 4

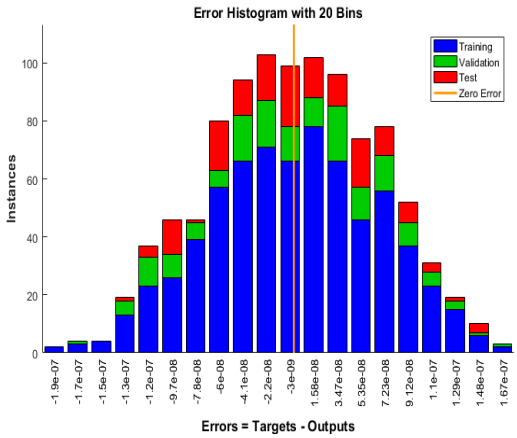


(e) Error histogram 5

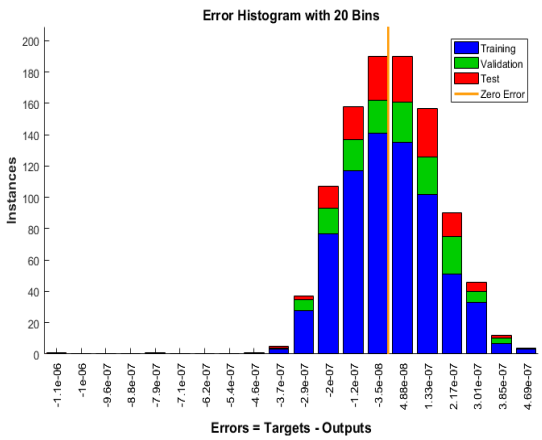


(b) Error histogram 6

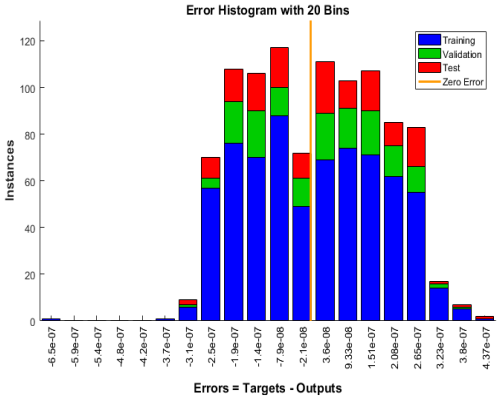
Figure 13. Using the parameters presented in Table 01 scenarios and Table 02 values creates an LMB-NN graph of the Error histogram of CD8+T lymphocytes.



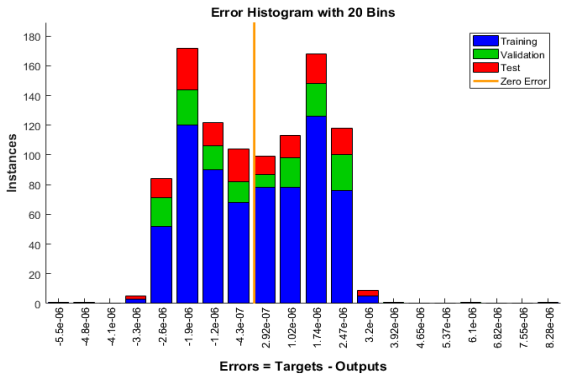
(a) Error histogram 1



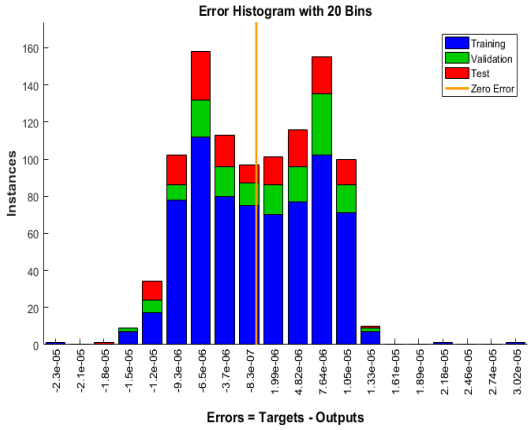
(b) Error histogram 2



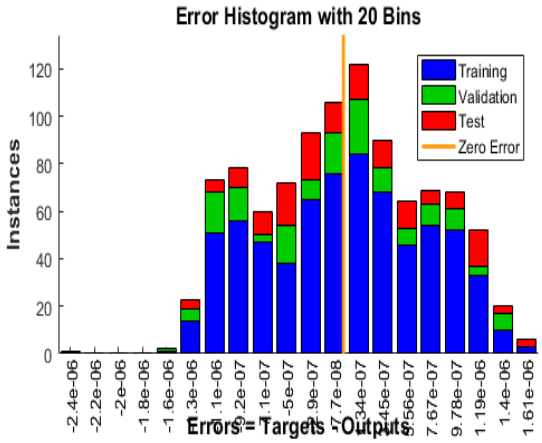
(c) Error histogram 3



(d) Error histogram 4

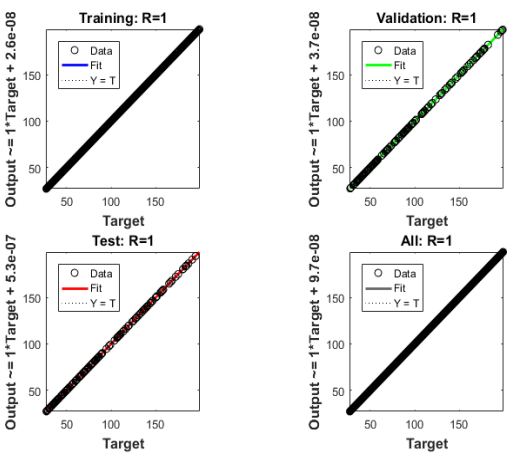


(c) Error histogram 5

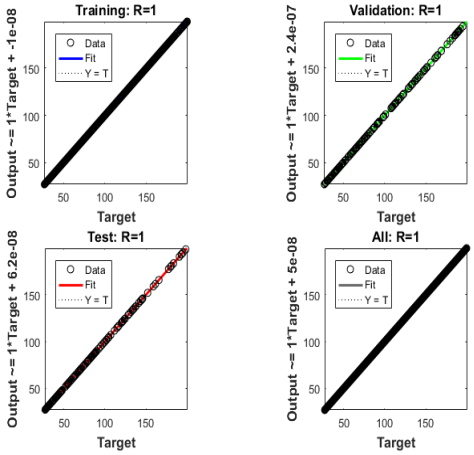


(d) Error histogram 6

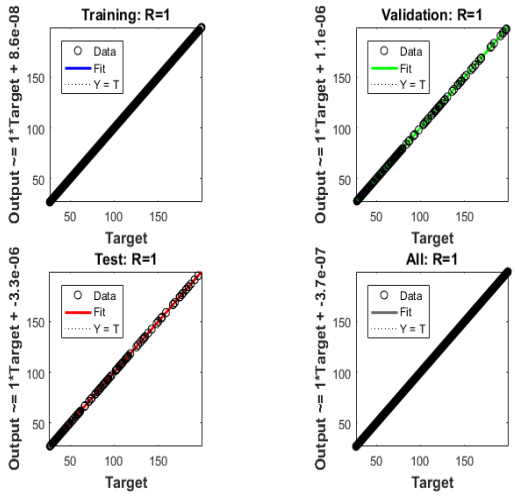
Figure 14. The data from Table 1 scenarios are used to create anti-PDL1 cell LMB-NN error Histogram graphs.



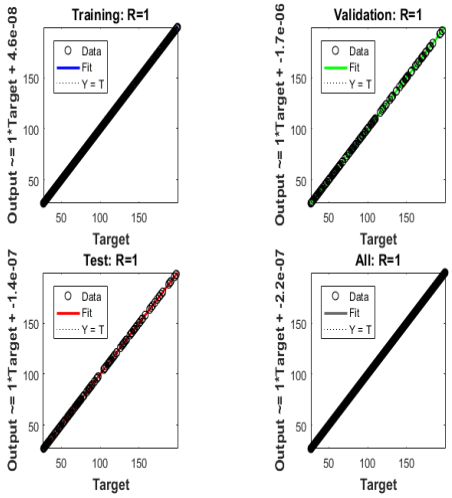
(a) Regression 1



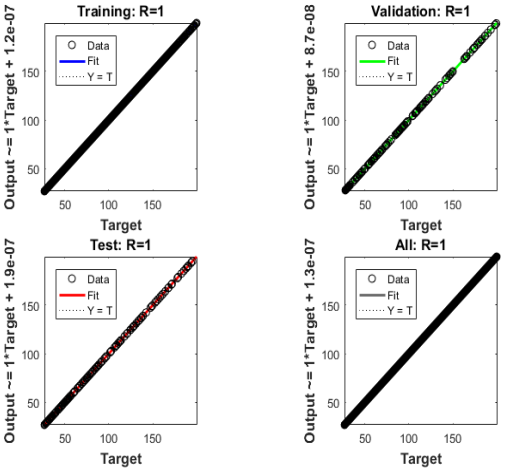
(b) Regression 2



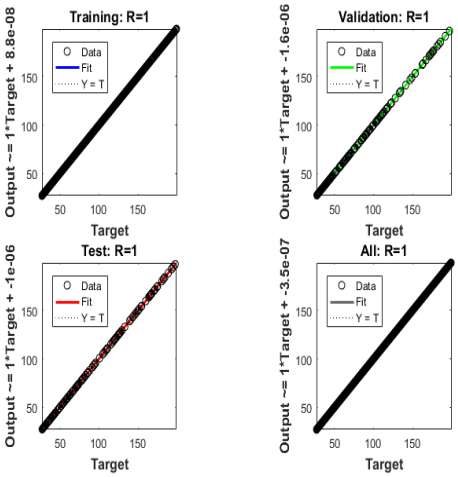
(c) Regression 3



(d) Regression 4

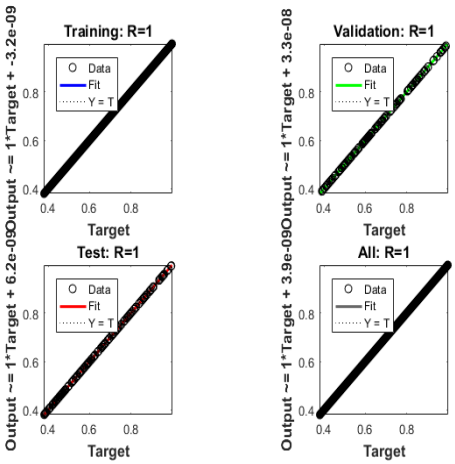


(e) Regression 5

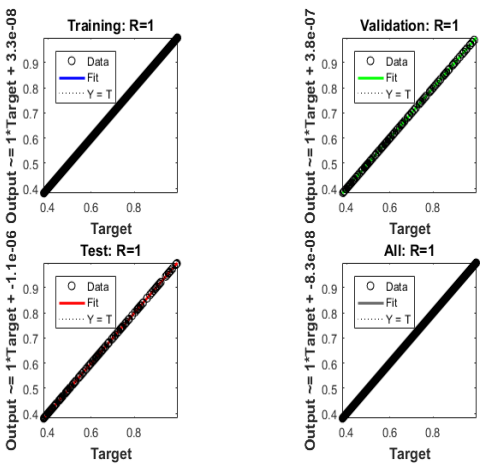


(f) Regression 6

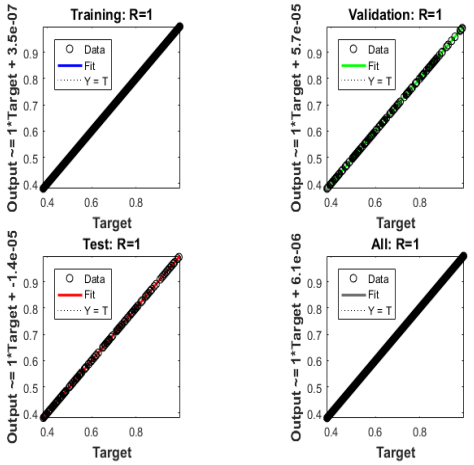
Figure 15. Using the parameter values from the scenario in Table 01 and values in Table 02, the Regression graphs of the IL-10 cytokine (I10) appear according to LMB-NN.



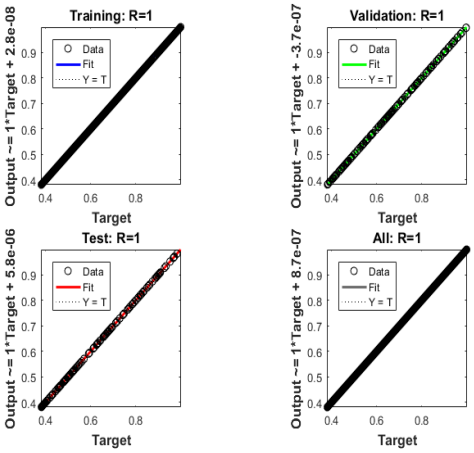
(a) Regression 1



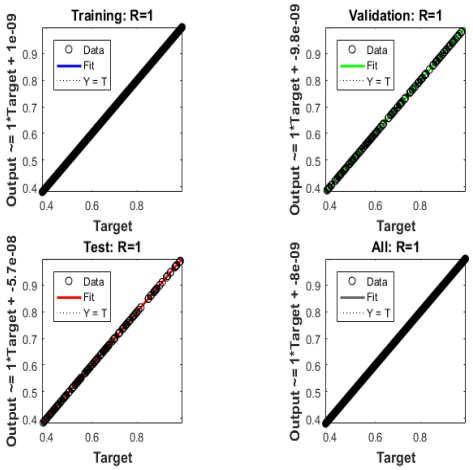
(b) Regression 2



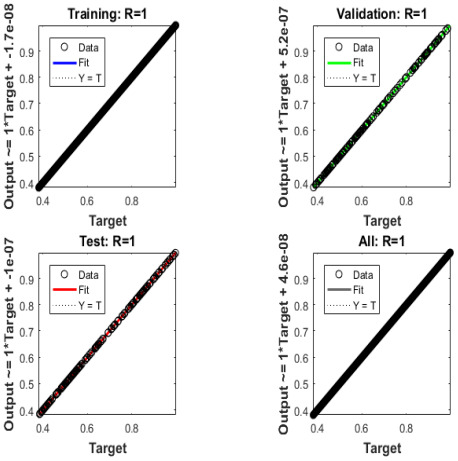
(c) Regression 3



(d) Regression 4

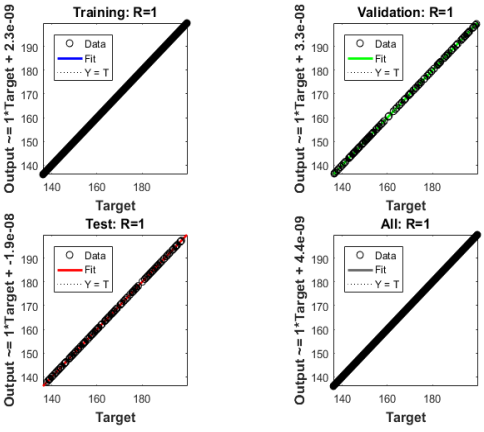


(e) Regression 5

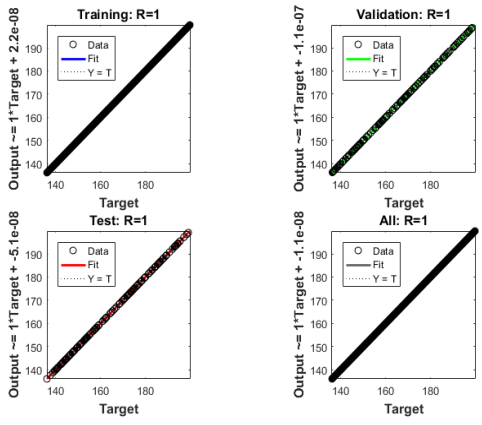


(f) Regression 6

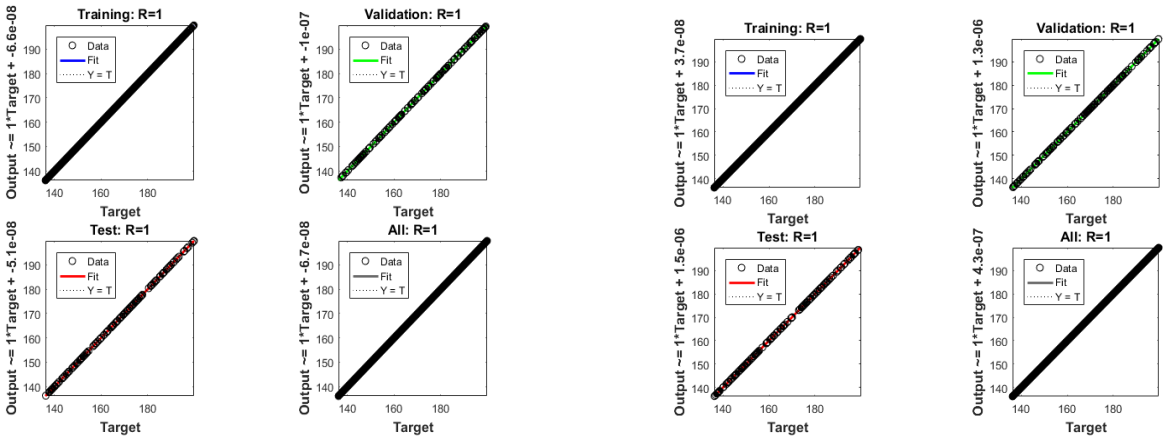
Figure 16. The inputs from Table 1 and Table 2 parameters are used to create CD8+ T lymphocyte LMB-NN regression graphs.



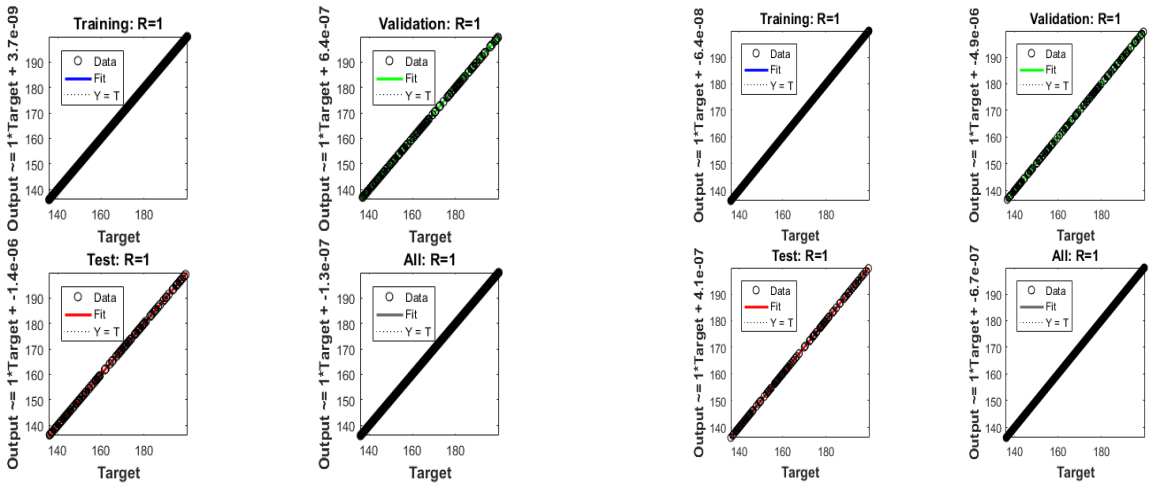
(a) Regression 1



(b) Regression 2



(c) Regression 3 (d) Regression 4



(d) Regression 5 (e) Regression 6

Figure 17. Using the inputs from the scenario listed in Table 1 and Table 2, anti-PD-L1 cell LMB-NN Regression Graphs are produced.

Table 9. Results for each case of the stated model using LMB-NN (1) - (4).

			MSE				
Case	**Training**	**Validation**	**Testing**	**Performance**	**Gradient**	**mu**	**Epochs**
1	4.44E-12	4.41E-12	4.90E-12	2.70E-10	8.69E-12	2.27E-01	1000
	1.57E-14	1.69E-14	1.52E-14	5.32E-13	4.98E-09	3.75E-15	1000
	4.52E-15	5.08E-15	3.75E-15	2.66E-10	2.68E-10	3.08E-14	1000
2	2.67E-11	2.85E-11	3.13E-11	7.78E+06	5.77E+06	7.89E+05	75
	8.54E-12	6.59E-12	8.69E-12	1.00E+11	3.43E+06	9.87E+10	985
	2.49E-14	3.10E-14	3.08E-14	7.48E+07	5.56E+06	1.23E+08	44
3	2.49E-10	2.53E-10	2.66E-10	8.44E+08	2.22E+05	7.67E+09	45
	4.92E-09	5.70E-09	4.98E-09	9.30E+06	3.33E+05	7.58E+08	22
	2.93E-14	2.48E-14	2.86E-14	9.89E+05	1.11E+06	8.58E+07	245

7. | DISCUSSION

The trained LMB-NN provides an approximation with a small error of the map between the moment t and the four state variables of the particular ODE system, as per the obtained results. Good regression properties and small joint MSE values indicate the fact that the learnt map approximates the numerically calculated trajectories with the smallest possible error under the given conditions. Unlike numerical integration of the governing equations independently, the meaning of computation is the representation of the numerically generated solution. Unlike LMB-NN, which learns the input-output map based on the reference samples, RK4 uses the governing equations to calculate the next step of the ODE solution instead of relying on numerical methods. Therefore, RK4 remains the generator of the reference solutions in the current experiments, but LMB-NN may be useful in cases when there is a need for repeated evaluation of the learnt map in the corresponding domain. The current set of experiments does not prove LMB-NN to be either more accurate, stable, or computationally cheaper than the RK4; such statements require a dedicated time comparison and independent generation of reference solutions.

Moreover, the findings are not as comprehensive as those provided by the newer cancer-immune models that use experimental verification or calibration. Other pathways like T-cell depletion, radiotherapy, neoantigen induction, or differential responsiveness to therapy have been represented in more recent mechanistic models that have been compared to animal/patient results [43–46]. The current paper, however, deals with a numerical estimation of the trajectories of a four-dimensional ODE model. Hence, biological interpretation of the generated trajectories depends on the assumptions made in the corresponding mathematical model. Some limitations should be pointed out, as well. Firstly, the biological model is a reduced deterministic ODE model that does not take into account inter-patient variability, spatial heterogeneity, stochasticity, pharmacokinetics, pharmacodynamics, and the entire tumor microenvironment. Secondly, the choice of the two families of scenarios and the parameter set used is a computation-driven one without any calibration to a clinical cohort. Thirdly, the simulation time domain is $t \in [0, 200]$.

Moreover, there was no architectural optimization performed for the 25 hidden neurons present in the LMB-NN architecture. Fifthly, since the RK4/NDSolve method produces neural network targets via numerical simulation, test errors reflect the discretization properties and assumptions of the reference method. Finally, there is no external dataset used to verify the results obtained through the experiments – no evidence was given of the generalizability of the presented cases. There is no guarantee that the considered cases are generalizable. The 70% training, 15% testing, and 15% validation split used here does not demonstrate any kind of generalization because of the

testing of the held-out data from the generated trajectories. To obtain a more general result, one needs to use independent validation based on experimental/clinical data along with the other steps.

7.1. Interpretation of Numerical Validation

The MSE values, error histograms, regression plots, and comparative tables can only be considered as internal numerical validation of the LMB-NN approximation relative to the selected RK4 reference trajectories. They do not contribute to the independent validation of the cancer dynamics model. Instead of measurements obtained from other biological experiments or patient cases, the testing data constitute withheld samples from the very same simulated trajectory distribution. The representative testing MSE values shown in Table 09 are 4.90×10^{-12}, 1.52×10^{-14}, 3.75×10^{-15}, 3.13×10^{-11}, 8.69×10^{-12}, and 3.08×10^{-14}. Neither clinical validity nor superiority over classical numerical integration can be derived from these figures.

8. | CONCLUSION

Applying a numerical reference framework based on RK4 and a supervised learning of LMB-NN to a four-dimensional nonlinear cancer–immune ODE model, which contains CD8+ T lymphocytes, cancer cells, IL-10, and anti-PD-L1 antibodies, the present study reveals that MSE values of training, validation, and testing achieved in the six cases are approaching 10^-12 - 10^-15 in many trials, with testing MSE values of 4.90 × 10^-12, 1.52 × 10^-14, and 3.75 × 10^-15. The outputs of LMB-NN and the corresponding numerical reference trajectories are well aligned according to the results from regression and error-histogram analysis. The primary contribution is the computational one, in the sense that LMB-NN can be regarded as a data-driven surrogate for computing the trajectories of the given ODE system of cancer dynamics. It is important to note that the findings should not be interpreted as biological/clinical validation or LMB-NN being superior to RK4/NDSolve. The limitations of the study are the reduced model formulation, selected parameter scenarios, limited simulation time, constant neural network structure, and dependence on numerical reference data. Future research will have to cover equilibrium stability analysis, solver refinement study, independent experiments or clinical observations, parameter calibration, parameter uncertainty study, and much broader parameter investigation.

Acknowledgment: The authors thank the anonymous reviewers for their valuable comments and constructive suggestions, which helped improve the quality of this manuscript.

Author Contributions: Conceptualization, Hussain, Z.; Methodology, Hussain, Z. and Sulaiman, M.; Formal Analysis, Hussain, Z. and Sulaiman, M.; Investigation, Hussain, Z. and Sulaiman, M.; Validation, Sulaiman, M. and Ahmad, F.; Visualization, Hussain, Z.; Supervision, Ahmad, F.; Project Administration, Ahmad, F.; Writing – Original Draft, Hussain, Z.; Writing – Review & Editing, Sulaiman, M. and Ahmad, F. All authors have read and agreed to the published version of the manuscript.

Declaration of Conflicting Interest: The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Licenses and Copyright: © The Author(s) 2026. This article is an open-access publication distributed under the terms of the Creative Commons Attribution 4.0 International (CC BY 4.0) License, which permits unrestricted use, distribution, adaptation, and reproduction in any medium or format, provided appropriate credit is given to the original author(s) and the source, a link to the license is provided, and any changes made are indicated.

Funding: The author(s) received no financial support for the research, authorship, and/or publication of this article.

Data Availability Statement: The data supporting the findings of this study are not available.

Plagiarism Statement: This manuscript was screened for textual similarity using Turnitin. The similarity results were found to be within the plagiarism thresholds prescribed by the Higher Education Commission (HEC) of Pakistan and the journal's editorial policy.

Artificial Intelligence (AI) Statement: The authors declare that no generative artificial intelligence tools or AI-assisted technologies were used in the preparation of this manuscript.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of the publisher "AGASR" and/or the editor(s). The Publisher AGASR and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

REFERENCES

[1] V. Kumar, A. K. Abbas, and J. C. Aster, *Robbins basic pathology E-Book: Robbins basic pathology E-Book*. Elsevier Health Sciences, 2017.

[2] B. T. Fife *et al.*, "Interactions between PD-1 and PD-L1 promote tolerance by blocking the TCR–induced stop signal," *Nat Immunol*, vol. 10, no. 11, pp. 1185–1192, Nov. 2009, doi: 10.1038/ni.1790.

[3] L. M. Francisco *et al.*, "PD-L1 regulates the development, maintenance, and function of induced regulatory T cells," *Journal of Experimental Medicine*, vol. 206, no. 13, pp. 3015–3029, Dec. 2009, doi: 10.1084/jem.20090847.

[4] Y. Latchman *et al.*, "PD-L2 is a second ligand for PD-1 and inhibits T cell activation," *Nat Immunol*, vol. 2, no. 3, pp. 261–268, Mar. 2001, doi: 10.1038/85330.

[5] F. Bertucci, P. Finetti, D. Birnbaum, and E. Mamessier, "The PD1/PDL1 axis, a promising therapeutic target in aggressive breast cancers," *OncoImmunology*, vol. 5, no. 3, p. e1085148, Mar. 2016, doi: 10.1080/2162402X.2015.1085148.

[6] S. Muenst *et al.*, "Expression of programmed death ligand 1 (PD-L1) is associated with poor prognosis in human breast cancer," *Breast Cancer Res Treat*, vol. 146, no. 1, pp. 15–24, Jul. 2014, doi: 10.1007/s10549-014-2988-5.

[7] I. H. Chan *et al.*, "The potentiation of IFN-γ and induction of cytotoxic proteins by pegylated IL-10 in human CD8 T cells," *Journal of Interferon & Cytokine Research*, vol. 35, no. 12, pp. 948–955, 2015.

[8] H. Sun, P. Gutierrez, M. J. Jackson, N. Kundu, and A. M. Fulton, "Essential role of nitric oxide and interferon-γ for tumor immunotherapy with interleukin-10," *Journal of immunotherapy*, vol. 23, no. 2, pp. 208–214, 2000.

[9] H. Sun, M. J. Jackson, N. Kundu, and A. M. Fulton, "Interleukin-10 gene transfer activates interferon-? and the interferon-?-inducible genesGbp-1/Mag-1 andMig-1 in mammary tumors," *Int. J. Cancer*, vol. 80, no. 4, pp. 624–629, Feb. 1999, doi: 10.1002/(SICI)1097-0215(19990209)80:4<624::AID-IJC23>3.0.CO;2-9.

[10] A. Naing *et al.*, "Safety, Antitumor Activity, and Immune Activation of Pegylated Recombinant Human Interleukin-10 (AM0010) in Patients With Advanced Solid Tumors," *JCO*, vol. 34, no. 29, pp. 3562–3569, Oct. 2016, doi: 10.1200/JCO.2016.68.1106.

[11] A. R. M. Carvalho and C. M. A. Pinto, "New developments on AIDS-related cancers: The role of the delay and treatment options," *Math Methods in App Sciences*, vol. 41, no. 18, pp. 8915–8928, Dec. 2018, doi: 10.1002/mma.4657.

[12] J. Danane, Z. Hammouch, K. Allali, S. Rashid, and J. Singh, "A fractional-order model of coronavirus disease 2019 (COVID-19) with governmental action and individual reaction," *Math Methods in App Sciences*, vol. 46, no. 7, pp. 8275–8288, May 2023, doi: 10.1002/mma.7759.

[13] M. T. Şenel, R. Habbireeh, and F. Özköse, "Fractional-order mathematical modelling of cancer cells-cancer stem cells-immune system interaction with chemotherapy," *mmnsa*, vol. 1, no. 2, pp. 67–83, Dec. 2021, doi: 10.53391/mmnsa.2021.01.007.

[14] F. Evirgen, "Transmission of Nipah virus dynamics under Caputo fractional derivative," *Journal of Computational and Applied Mathematics*, vol. 418, p. 114654, Jan. 2023, doi: 10.1016/j.cam.2022.114654.

[15] F. Evirgen, S. Uçar, and N. Özdemir, "System Analysis of HIV Infection Model with $CD\,4^{+}\,T$ under Non-Singular Kernel Derivative," *Applied Mathematics and Nonlinear Sciences*, vol. 5, no. 1, pp. 139–146, Jan. 2020, doi: 10.2478/amns.2020.1.00013.

[16] M. F. Farayola, S. Shafie, F. M. Siam, and I. Khan, "Mathematical modeling of radiotherapy cancer treatment using Caputo fractional derivative," *Computer Methods and Programs in Biomedicine*, vol. 188, p. 105306, May 2020, doi: 10.1016/j.cmpb.2019.105306.

[17] A. A. Hamou, R. R. Q. Rasul, Z. Hammouch, and N. Özdemir, "Analysis and dynamics of a mathematical model to predict unreported cases of COVID-19 epidemic in Morocco," *Comp. Appl. Math.*, vol. 41, no. 6, p. 289, Sep. 2022, doi: 10.1007/s40314-022-01990-4.

[18] A. Khan, K. Shah, T. Abdeljawad, and M. A. Alqudah, "Existence of results and computational analysis of a fractional order two strain epidemic model," *Results in Physics*, vol. 39, p. 105649, Aug. 2022, doi: 10.1016/j.rinp.2022.105649.

[19] X. Lai and A. Friedman, "Combination therapy of cancer with cancer vaccine and immune checkpoint inhibitors: A mathematical model," *PLoS ONE*, vol. 12, no. 5, p. e0178479, May 2017, doi: 10.1371/journal.pone.0178479.

[20] X. Lai and A. Friedman, "Mathematical modeling of cancer treatment with radiation and PD-L1 inhibitor," *Sci. China Math.*, vol. 63, no. 3, pp. 465–484, Mar. 2020, doi: 10.1007/s11425-019-1648-6.

[21] N. Özdemir and E. Uçar, "Investigating of an immune system-cancer mathematical model with Mittag-Leffler kernel," *AIMS Mathematics*, vol. 5, no. 2, pp. 1519–1531, 2020, doi: 10.3934/math.2020104.

[22] N. Ozdemir, E. Ucar, and D. Avcı, "DYNAMIC ANALYSIS OF A FRACTIONAL SVIR SYSTEM MODELING AN INFECTIOUS DISEASE," *FU Math Inform*, p. 605, Sep. 2022, doi: 10.22190/FUMI211020042O.

[23] F. Özköse *et al.*, "A Fractional Modeling of Tumor–Immune System Interaction Related to Lung Cancer with Real Data," *Eur. Phys. J. Plus*, vol. 137, no. 1, p. 40, Jan. 2022, doi: 10.1140/epjp/s13360-021-02254-6.

[24] E. Uçar, N. Özdemir, and E. Altun, "Qualitative analysis and numerical simulations of new model describing cancer," *Journal of Computational and Applied Mathematics*, vol. 422, p. 114899, Apr. 2023, doi: 10.1016/j.cam.2022.114899.

[25] E. Uçar and N. Özdemir, "A fractional model of cancer-immune system with Caputo and Caputo–Fabrizio derivatives," *Eur. Phys. J. Plus*, vol. 136, no. 1, p. 43, Jan. 2021, doi: 10.1140/epjp/s13360-020-00966-9.

[26] E. Ucar, N. Özdemir, and E. Altun, "Fractional order model of immune cells influenced by cancer cells," *Math. Model. Nat. Phenom.*, vol. 14, no. 3, p. 308, 2019, doi: 10.1051/mmnp/2019002.

[27] A. Yousef, F. Bozkurt, and T. Abdeljawad, "Mathematical modeling of the immune-chemotherapeutic treatment of breast cancer under some control parameters," *Adv Differ Equ*, vol. 2020, no. 1, p. 696, Dec. 2020, doi: 10.1186/s13662-020-03151-5.

[28] H. Zhang, Y. Wang, E. S. Hwang, and Y.-W. He, "Interleukin-10: An Immune-Activating Cytokine in Cancer Immunotherapy," *JCO*, vol. 34, no. 29, pp. 3576–3578, Oct. 2016, doi: 10.1200/JCO.2016.69.6435.

[29] Y. Guerrero Sánchez, Z. Sabir, H. Günerhan, and H. M. Baskonus, "Analytical and Approximate Solutions of a Novel Nervous Stomach Mathematical Model," *Discrete Dynamics in Nature and Society*, vol. 2020, pp. 1–9, Jul. 2020, doi: 10.1155/2020/5063271.

[30] M. Umar, Z. Sabir, M. A. Z. Raja, M. Shoaib, M. Gupta, and Y. G. Sánchez, "A Stochastic Intelligent Computing with Neuro-Evolution Heuristics for Nonlinear SITR System of Novel COVID-19 Dynamics," *Symmetry*, vol. 12, no. 10, p. 1628, Oct. 2020, doi: 10.3390/sym12101628.

[31] Y. G. Sánchez, Z. Sabir, and J. L. G. Guirao, "DESIGN OF A NONLINEAR SITR FRACTAL MODEL BASED ON THE DYNAMICS OF A NOVEL CORONAVIRUS (COVID-19)," *Fractals*, vol. 28, no. 08, p. 2040026, Dec. 2020, doi: 10.1142/S0218348X20400265.

[32] M. Onal and A. Esen, "A Crank-Nicolson approximation for the time fractional Burgers equation," *Applied Mathematics and Nonlinear Sciences*, vol. 5, no. 2, pp. 177–184, 2020.

[33] M. Topsakal and F. Tascan, "Exact travelling wave solutions for space-time fractional Klein-Gordon equation and (2+ 1)-Dimensional time-fractional Zoomeron equation via auxiliary equation method," *Applied Mathematics and Nonlinear Sciences*, vol. 5, no. 1, pp. 437–446, 2020.

[34] H. Durur, O. Tasbozan, and A. Kurt, "New Analytical Solutions of Conformable Time Fractional Bad and Good Modified Boussinesq Equations," *Applied Mathematics and Nonlinear Sciences*, vol. 5, no. 1, pp. 447–454, Jan. 2020, doi: 10.2478/amns.2020.1.00042.

[35] S. Ghosh, S. Mukhopadhyay, and K. Vajravelu, "MHD 3-dimensional nanofluid flow induced by a power-law stretching sheet with thermal radiation, heat and mass fluxes," *Applied Mathematics and Nonlinear Sciences*, vol. 6, no. 2, pp. 361–380, Jul. 2021, doi: 10.2478/amns.2020.2.00036.

[36] M. Dewasurendra and K. Vajravelu, "On the Method of Inverse Mapping for Solutions of Coupled Systems of Nonlinear Differential Equations Arising in Nanofluid Flow, Heat and Mass Transfer.," *Applied Mathematics & Nonlinear Sciences*, vol. 3, no. 1, 2018.

[37] P. Lakshminarayana, K. Vajravelu, G. Sucharitha, and S. Sreenadh, "Peristaltic slip flow of a Bingham fluid in an inclined porous conduit with Joule heating.," *Applied Mathematics & Nonlinear Sciences*, vol. 3, no. 1, p. 41, 2018.

[38] Z. Sabir, M. G. Sakar, M. Yeskindirova, and O. Saldir, "Numerical investigations to design a novel model based on the fifth order system of Emden–Fowler equations," *Theoretical and Applied Mechanics Letters*, vol. 10, no. 5, pp. 333–342, Jul. 2020, doi: 10.1016/j.taml.2020.01.049.

[39] M. E. Iglesias Martínez, J. A. Antonino-Daviu, P. F. De Córdoba, and J. A. Conejero, "Higher-Order Spectral Analysis of Stray Flux Signals for Faults Detection in Induction Motors," *Applied Mathematics and Nonlinear Sciences*, vol. 5, no. 2, pp. 1–14, Jul. 2020, doi: 10.2478/amns.2020.1.00032.

[40] G. Chen, C. Chen, Y. Yuan, and L. Zhu, "Modelling and Simulation Analysis of High-Pressure Common Rail and Electronic Controlled Injection System for Diesel Engine," *Applied Mathematics and Nonlinear Sciences*, vol. 5, no. 2, pp. 345–356, Jul. 2020, doi: 10.2478/amns.2020.2.00043.

[41] "Persistence and extinction of a stochastic delay competitive system under regime switching," *J. Nonlinear Funct. Anal.*, vol. 2019, no. 1, 2019, doi: 10.23952/jnfa.2019.36.

[42] Z. Sabir, M. A. Z. Raja, J. L. G. Guirao, and M. Shoaib, "Integrated intelligent computing with neuro-swarming solver for multi-singular fourth-order nonlinear Emden–Fowler equation," *Comp. Appl. Math.*, vol. 39, no. 4, p. 307, Dec. 2020, doi: 10.1007/s40314-020-01330-4.

[43] R. Eftimie, J. L. Bramson, and D. J. D. Earn, "Interactions Between the Immune System and Cancer: A Brief Review of Non-spatial Mathematical Models," *Bull. Math. Biol.*, vol. 73, no. 1, pp. 2–32, Jan. 2011, doi: 10.1007/s11538-010-9526-3.

[44] G. E. Mahlbacher, K. C. Reihmer, and H. B. Frieboes, "Mathematical modeling of tumor-immune cell interactions," *Journal of Theoretical Biology*, vol. 469, pp. 47–60, May 2019, doi: 10.1016/j.jtbi.2019.03.002.

[45] L. G. De Pillis, W. Gu, and A. E. Radunskaya, "Mixed immunotherapy and chemotherapy of tumors: modeling, applications and biological interpretations," *Journal of Theoretical Biology*, vol. 238, no. 4, pp. 841–862, Feb. 2006, doi: 10.1016/j.jtbi.2005.06.037.

[46] J. D. Butner *et al.*, "Mathematical modeling of cancer immunotherapy for personalized clinical translation," *Nat Comput Sci*, vol. 2, no. 12, pp. 785–796, Dec. 2022, doi: 10.1038/s43588-022-00377-z.

[47] M. L. Shahab, I. Mukhlash, and H. Susanto, "Do physics-informed neural networks (PINNs) need to be deep? Shallow PINNs using the Levenberg-Marquardt algorithm," *arXiv preprint arXiv:2602.08515*, 2026.