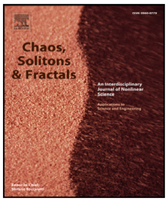
Contents lists available at ScienceDirect

Chaos, Solitons and Fractals

journal homepage: www.elsevier.com/locate/chaos



Prediction of fluctuations in a chaotic cancer model using machine learning

Elaheh Sayari [a,1], Sidney T. da Silva [a,b,1], Kelly C. Iarosz [c,d,1], Ricardo L. Viana [e,f,1], José D. Szezech Jr [a,g,1], Antonio M. Batista [a,g,*,1]

[a] Graduate Program in Science, State University of Ponta Grossa, Ponta Grossa, PR, Brazil
[b] Federal University of Paraná, Curitiba, PR, Brazil
[c] University Center UNIFATEB, Telêmaco Borba, Paraná, Brazil
[d] Graduate Program in Chemical Engineering Federal Technological University of Paraná, Ponta Grossa, PR, Brazil
[e] Department of Physics, Federal University of Paraná, Curitiba, PR, Brazil
[f] Institute of Physics, University of São Paulo, São Paulo, SP, Brazil
[g] Mathematics and Statistics Department, State University of Ponta Grossa, Ponta Grossa, PR, Brazil

ARTICLE INFO

Keywords:
Cancer
Mathematical methods
Chaotic attractor
Machine learning
Fluctuations

ABSTRACT

Cancer is a group of diseases and the second leading cause of death according to World Health Organization. Mathematical and computational methods have been used to explore the cancer cells spread and the mechanism of their growth. We study a cancer model that exhibits both periodic and chaotic attractors. It describes the interactions among host, effector immune, and cancer cells. It is observed fluctuations in the population of cells. The fluctuation range can be associated with the appearance of tumour cells. In this work, we use machine learning algorithms for the prediction of fluctuations. We show that our machine learning classification is able to identify fluctuations that are associated with the growth rate of cancer cells.

1. Introduction

Cancer is among the leading causes of death in the world. It is an abnormal growth of normal tissues and can spread throughout the body. Understanding its mechanism of proliferation and prevention can be effective to control cancer [1]. When the immune system recognises tumour cells, it plays an important role in protecting the body from malignancy [2,3]. Due to this fact, studies have been considered interactions between effector immune and tumour cells [4].

Mathematical models have been used to improve the comprehension of the relation among cancerous cells and ways to stop their growth or spreading [5]. Some cancer models based on Lotka–Volterra competition and chaotic equations were considered to investigate situations in which tumour cells may be suppressed [6]. Models with negative competitive effects of cancer cells on host cells and vice versa [7] can identify the parameters which should be targeted by treatments, as well as determine the most effective strategy for medicine therapy regime [8]. Borges et al.,[9] investigated a delay differential equations model in which the cancer cells are attacked by the immune system and chemotherapeutic agents [10,11]. Iarosz et al. [12] and Trobia et al. [13] proposed mathematical models of brain tumour growth with glia–neuron interactions considering chemotherapy treatment and drug resistance, respectively. There have been a lot of theoretical and experimental investigations about the tumour–immune cell interactions [14]. Ghosh and Banerjee [15] demonstrated a mathematical modelling of cancer-immune system based on clinical evidences that antibodies can kill cancer cells.

In recent years, it has increased the use of machine learning models in nonlinear systems. Recurrent neural networks (RNN), as one type of machine learning model, have been explored to create a mixture of experts for modelling nonlinear systems [16]. A class of RNN, namely reservoir computing (RC) network, has been successfully applied to many computational problems, such as prediction [17]. RC is suitable for model-free prediction of nonlinear and chaotic systems [18,19]. Also, this framework has been used to reconstruct bifurcation diagrams of nonlinear systems [20].

Itik and Banks [21] computed the Lyapunov exponents and the Lyapunov dimension to confirm chaotic dynamics in a cancer model composed of healthy, tumour, and effector immune cell populations. They showed that the dynamics of the tumour–immune system interaction can lead to the emergence of chaotic attractors in their phase space. It was reported a tumour escape (immune system suppression) and an uncontrolled growth of cancer cell populations. Izquierdo-Kulich et al. [22] used the entropy production rate as a robust characteristic of

[*] Corresponding author at: Graduate Program in Science, State University of Ponta Grossa, Ponta Grossa, PR, Brazil.
E-mail address: antoniomarcosbatista@gmail.com (A.M. Batista).

[1] All authors discussed the results and contributed to the final version of the manuscript.

https://doi.org/10.1016/j.chaos.2022.112616
Received 20 June 2022; Received in revised form 20 August 2022; Accepted 23 August 2022
Available online 5 September 2022
0960-0779/© 2022 Elsevier Ltd. All rights reserved.

the tumour complexity and a measurable value from the contour-based fractal dimensions of avascular tumour growth. They demonstrated that the increase of the entropy production rate is proportional to the growth rate of tumours. Toker et al. [23] applied a chaotic decision tree algorithm with very high accuracy to a wide variety of complex systems in nature, such as a model of the transcription of the NF-kB protein complex related to several genes involved in immune responses and cancer research [24]. This algorithm uses the permutation entropy of the inputted signal to approximate the degree of chaos. Obcemea [25] studied the behaviours related to tumour cells and other cells through a chaos framework by considering the bifurcation structure of nonlinear equations for tumour growth.

In this work, we study a cancer model that is based on the Lotka–Volterra equations and can exhibit chaotic behaviour. It describes the interactions among host, effector immune, and tumour cells [26]. Depending on the parameters, this model has numerous similarities with clinical evidences [21]. An extensive analysis can provide insights into cellular interactions that correspond to some empirical and clinical observations [27,28]. We use machine learning algorithms for the prediction of fluctuations. We show that our machine learning classification is able to identify fluctuations that are associated with the growth rate of cancer cells.

The paper is organised as follows. Section 2 shows the cancer model that describes the interactions among three cell populations. An analysis in the form of bifurcation diagrams is performed. In Section 3, we show the neural network used to analyse the cancer model. Section 4 presents our results. Concluding remarks are in the final section.

2. The cancer model

A model composed of ordinary differential equations containing effector immune and tumour cells was introduced by Kuznetsov et al. [29]. The model with only two cell populations was able to show some important aspects of the stages of cancer growth. De Pillis and Radunskaya [26] analysed the phase space for the model proposed by Kuznetsov et al. [29] and added a normal (host) cell population. They also studied the effect of chemotherapy treatment using optimal control theory. Furthermore, Kirschner and Panetta [30] investigated the tumour cell growth in the presence of two cell populations, namely the effector immune cells and the cytokine IL-2. To diagnosis the tumour cells by the effector immune system, they found that the antigenicity of the tumour cell populations has an essential role in it. All these models containing various cell populations have basic common characteristics [31]. According to the studies done by [32–35], it has been reported that the dynamics of interactions of tumour cells with other cells may exhibit chaos. In this regard, Itik and Banks [21] used the model presented by [26] to show chaotic dynamics in three cell populations. We study the model proposed by [26] that exhibits chaotic dynamics [21] and has numerous analogies to clinical evidences [27,28]. The parameter values, that we consider for the model, are selected according to the findings obtained from some biological evidences [26].

We consider a mathematical model that describes the interactions among host (normal), effector immune (natural killer, macrophages and CD8+T cells), and cancer cells in a single tumour site [21,26]. The normalised cancer model [36] is given by

$$\frac{dx}{dt} = p_1 x(1-x) - \alpha_{13} xz, \tag{1}$$

$$\frac{dy}{dt} = \frac{p_2 yz}{1+z} - \alpha_{23} yz - \delta_2 y, \tag{2}$$

$$\frac{dz}{dt} = z(1-z) - xz - \alpha_{32} yz, \tag{3}$$

where x, y, and z correspond to the normalised rate of change of host, effector immune, and tumour cells, respectively. The descriptions and values of the parameters are shown in Table 1 [21]. The parameters p_1 and p_2 are the growth rate of host cells and effector immune cells, respectively. The parameter p_2 is a positive constant, which can be varied to determine the bifurcations. For p_2, we consider 4.5 to simulate a patient with a compromised immune system. The parameter α_{13} corresponds to the rate of inhibiting host cells by tumour cells, α_{23} is associated with the inactivation of effector immune cells by tumour cells, and α_{32} is the rate of killing tumour cells by effector immune cells. The competition terms are positive in our simulations [37,38]. The parameters α_{13}, α_{23}, and α_{32} are considered as 1.5, 0.2, and 2.5, respectively. The parameter δ_2 equal to 0.5 denotes the per capita death rate of the immune cells.

Table 1
Parameter descriptions and values used in our simulations.

Name	Description	Value
p_1	Growth rate of host cells	[0.3,1.4]
α_{13}	Host cell killing rate by tumour cells	1.5
p_2	Growth rate of Effector immune cells	4.5
α_{23}	Effector immune cell Inhibiting rate by tumour cells	0.2
δ_2	Effector immune cell mortality	0.5
α_{32}	Effector immune cell killing rate by tumour cells	2.5

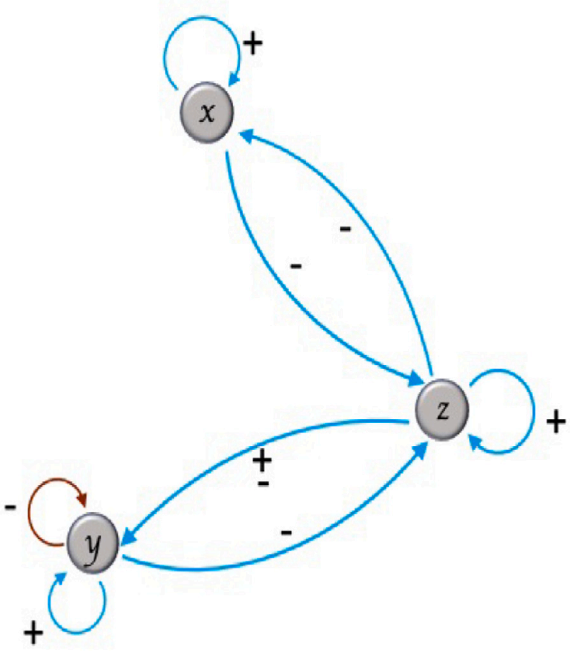


Fig. 1. Schematic representation of the cancer model, where the red and blue lines correspond to the linear and nonlinear interactions, respectively. The signs are related to the growth (+) or suppression (-) of the cells.

In Eq. (1), the first term is a logistic function and the second one is the inhibition of host cells due to the cancer. The first term in Eq. (2) is associated with the stimulation of the immune system by cancer cells. The second and third terms are responsible for the suppression of effector immune cells by tumour cells and due to the naturally die, respectively. In Eq. (3), the first term is a logistic function and the second term represents the competition between cancer and host cells. The third term corresponds the death of tumour cells by effector immune cells. Fig. 1 displays a schematic representation of the cancer model, where the red and blue lines represent the linear and nonlinear interactions, respectively.

Depending on the parameters, the cancer model can exhibit attractors. Fig. 2(a) displays a periodic attractor for $p_1 = 0.3750$ and parameters according to Table 1. For $p_1 = 0.6$, we observe a chaotic

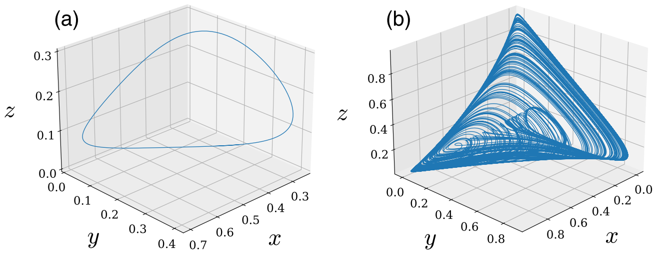


Fig. 2. (a) Periodic attractor for $p_1 = 0.3750$ and (b) chaotic attractor for $p_1 = 0.6$. We consider the parameters shown in Table 1.

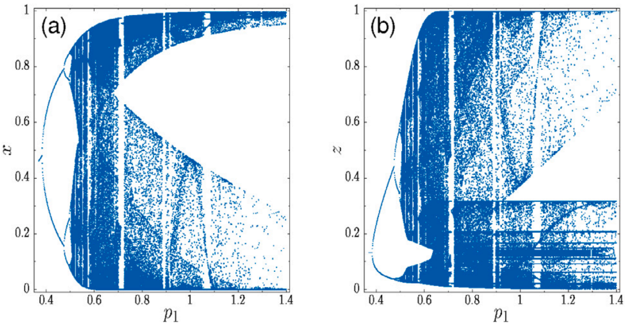


Fig. 3. Bifurcation diagrams for (a) host and (b) tumour cells.

attractor, as shown in Fig. 2(b). Then, periodic and chaotic trajectories can be generated by the cancer model.

We plot the bifurcation diagrams for the host and cancer cells as a function of p_1, as shown in Figs. 3(a) and 3(b), respectively. To do that, we compute the maxima and minima values of the time series. The diagrams exhibit period-doubling cascades, periodic windows, and chaotic attractors [39]. The population of host cells can reach value close to 1, while the minimum becomes very low. As the growth rate of host cells increases, the cycle-to-cycle diversity decreases [36]. This means that the host cells population achieves its maximum for a long time, however it can drop to zero very quickly. Also, the population of tumour cells remains near its minimum value during a time period and can increase very quickly to one.

3. Neural network

Our deep neural network model, that we use in the analysis of the cancer model, basically consists of two coupled neural networks. An echo state network (ESN) as a recurrent neural network [40,41] coupled to a multilayer perceptron (MLP) network (Fig. 4), using a principal component analysis (PCA) [42] as dimension reducer.

3.1. BDESN networks

3.1.1. Reservoir computing

The Reservoir Computing (RC) has been used for modelling nonlinear time series. In the learning context, echo state networks (ESNs) are more common in RC models, where the input sequence is projected into a larger space through the use of the non-linear reservoir. Learning is accomplished through the application of simple linear techniques in the space of the reservoir. The architecture is adapted from [43], called Bidirectional Deep-readout Echo State Networks (BDESN), which combines the speed of RC with the trainable precision of RNNs. The model is equipped with a bidirectional reservoir. Bidirectional architectures have successfully been applied in RNNs to extract temporal resources from the time series that have a very long time dependency.

The BDESN is used for the classification of time series $\mathbf{u} = \{\mathbf{u}_t\}_{t=0}^{T}$ through the following procedure. We first project the time series with smaller dimension $\mathbf{u}(t)$ to a larger space, through the reservoir. Then a dimension reduction algorithm projects the reservoir outlet into a smaller space. Finally, a multilayer perceptron (MLP) classifies the vector representative of $\mathbf{u}$, as shown in Fig. 4.

3.1.2. Reservoir

The reservoir acts as an encoder that generates the input representation in a larger space. The state produced by the reservoir brings all the dynamic information from the original input. The encoding is performed using the weights $\theta_{enc} = \{W^i, W^h\}$. The dynamics of this process is given by

$$\begin{aligned}\mathbf{h}(t) = {} & (1-\alpha)\mathbf{u}(t) + \alpha f(\mathbf{W}^h\mathbf{h}(t-1) \\ & + \mathbf{W}^i\mathbf{u}(t) + \eta(t)),\end{aligned} \tag{4}$$

where $\mathbf{h}(t)$ is the time-dependent internal state, which combines the current input $\mathbf{u}(t)$ with the previous state $\mathbf{h}(t-1)$, f is a non-linear activation function (tanh), $\mathbf{W}^h$ is the sparse matrix that defines the recurrent self-connects in the reservoir, and $\mathbf{W}^i$ defines incoming connections. Both matrices are randomly generated and are not trained. The behaviour of the reservoir is mainly controlled by five hyper parameters, they are: the size of the states N, the spectral radius ρ of W^h, the dimensioning of the inputs ω, the hyper leakage parameter α, and the noise η which is used for regularisation in the reservoir. Through an optimal fit of these hyper parameters, the reservoir produces rich dynamics and its internal states can be used to solve many prediction and classification tasks.

The states generated by the reservoir $\mathbf{h}(t)$, after all inputs are processed, are a high-dimensional representation that incorporates the temporal dependencies of $\mathbf{u}$. Since the reservoir exchanges its internal stability with a memory [44], at the time T, the state tends to lose information from the initial times. To get around this problem, we feed the same reservoir with the inverse order of the time series $\mathbf{u}' = \{\mathbf{u}_{T-t}\}_{t=0}^{T}$ and generate a new state $\mathbf{h}(t)'$ that is more influenced with the first inputs. The final resultant state is obtained by concatenating the two states, $\bar{\mathbf{h}}_T = [\mathbf{h}(t); \mathbf{h}(t)']^T$. The bidirectional reservoir has recently been used for time series prediction [45]. From the sequence of the RNN states generated over time,

$$\mathbf{H} = [\bar{h}(1), \ldots, \bar{\mathbf{h}}(T)]^T, \tag{5}$$

it is possible to extract a representation $\mathbf{r}_u = r(\mathbf{H})$ of the input $\mathbf{u}$.

The $\mathbf{r_u}$ vector brings all the information about the characteristics of the $\mathbf{u}$ input, in this case the $\mathbf{r_u}$ vector is formed by the weights and bias learned when a later state is generated by the previous state as shown in

$$\mathbf{h}(t+1) = \mathbf{W}_r\mathbf{h}(t) + \mathbf{b}_r, \tag{6}$$

where $\mathbf{r}_u = \{\mathbf{W}_r, \mathbf{b}_r\} \in \mathbb{R}^{R(R+1)}$ and R is the number of neurons that form the reservoir.

After constructing $\mathbf{r}_u$, we can decode in the output space, which are the C classes for the classification case (Fig. 4). This decoding can be performed by

$$C = g(\mathbf{r}_u, \theta_{dec}), \tag{7}$$

where g is a multilayer perceptron network and the θ_{dec} weights and biases to be learned. The $\mathbf{r}_u$ state vector can also be used as input to an unsupervised learning algorithm.

As the reservoir has a large dimension due to the number of neurons, this takes an overfit and computational resources. The Principal Component Analysis (PCA) [42] dimensionally reduces the states, by transforming large states into a smaller one that still contains most of the information in the large states, while preserving as much variability as possible, showing a better performance. The PCA aims to reduce the feature space, choosing a space that better separates these features, facilitating the decision surface. In BDESN, dimensionality reduction not only provides a strong regularisation, but also prevents overfitting in the classifier operating on the reservoir states.

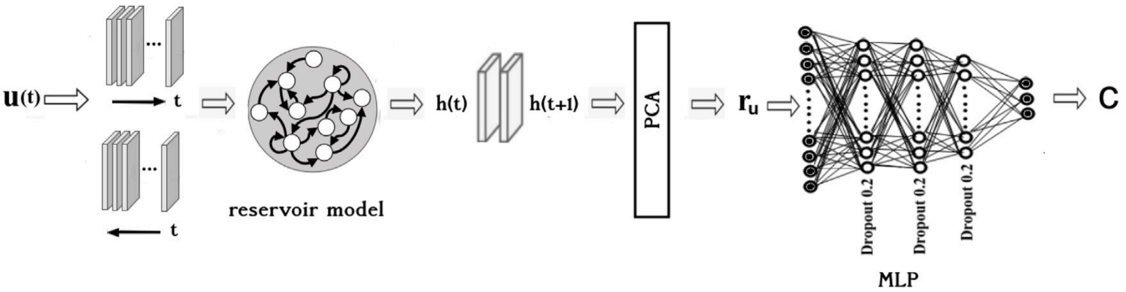


Fig. 4. Schematic representation of a deep network composed of a computational reservoir followed by a dimensional reduction method (PCA) and a reading layer formed of a multilayer perceptron network trained for classification.

3.1.3. Multilayer perceptron (MLP)

The state vectors with their reduced dimensions become the input vectors of the network (MLP), where the classification or regression takes place. MLP following a feedforward algorithm is a neural network in which the mapping between inputs and output is non-linear. In the network, supervised learning is made by a mechanism called back propagation that allows MLP to iteratively adjust the weights during training and to minimise the cost function. At this point, a normal training of a MLP network is made.

The deep MLPs are known for their generalisability and adaptability, that are important characteristics for the problem at hand. Nowadays, deep layer networks can be efficiently trained using sophisticated regularisation techniques and pre-training techniques that help to avoid overfit and null or explosive gradient problem.

In the architecture formed by the multilayer perceptron network, we use 500 neurons in the input layer and three hidden layers with 400 neurons each. These numbers of neurons in the layers give the best performance in the prediction, avoiding overfitting and resulting in a better generalisation. On the hidden layers, we use a dropout = 0.2 (Fig. 4) and "Greedy Layer-Wise" [46] as pre-training of the network. The optimiser and error function are "Adam" and "MSE", respectively.

4. Analysis of time series via machine learning

The objective of this work is to find the parameters values of the cancer model proposed in Eqs. (1), (2), and (3), that keep the dynamics of the host cells as long as possible close to 1. For this task, we train our deep neural network with several time series, which we label as classes 0 ($x = 1$), 1 (Fig. 5(a)), 2 (Fig. 5(b)), 3 (Fig. 5(c)), and 4 (Fig. 5(d)). These classes are grouped according to their dynamic characteristics.

The training is performed with 4000 samples, divided equally into 5 classes, where we use 70% for training, 5% for validation and 25% for testing. We calculate the performance measured through the confusion matrix. Performance is evaluated based on three main measurement performances in RNN models, which are accuracy, precision and recall, given by

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \tag{8}$$

$$Precision = \frac{TP}{TP + FP}, \tag{9}$$

$$Recall = \frac{TP}{TP + FN}, \tag{10}$$

$$F_1 score = \frac{2 * Precision * Recall}{Precision + Recall}. \tag{11}$$

These measurements are described using the confusion matrix that considers a two-class classification problem, as illustrated in Table 2. The main diagonal values are the correctly predicted values while the off-diagonal values are the wrongly predicted ones.

In neural networks, an important rule is the choice of hyper parameters for a better performance in the classification model without suffering overfit. After choosing these hyper parameters, using Bayesian optimisation, we can apply our network for the classification problem.

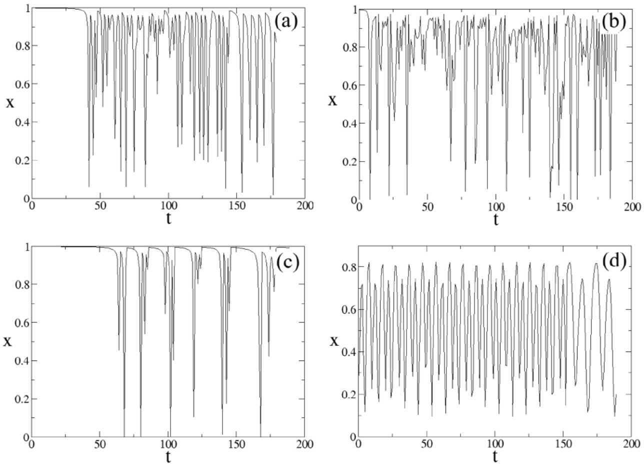


Fig. 5. $x \times t$ for (a) $\rho_1 = 1.1$, $\alpha_{13} = 1.5$, $\alpha_{23} = 0.2$ (class 1), (b) $\rho_1 = 0.7$, $\alpha_{13} = 1.5$, $\alpha_{23} = 0.2$ (class 2), (c) $\rho_1 = 0.7$, $\alpha_{13} = 0.8$, $\alpha_{23} = 0.2$ (class 3), and (e) $\rho_1 = 0.5$, $\alpha_{13} = 1.8$, $\alpha_{23} = 0.2$ (class 4). When the oscillation of the number of host cells is close to 1, the number of cancer cells remains close to zero.

Table 2
Confusion matrix.

	Predicted Positive	Predicted Negative
Current Positive	True Positive (TP)	False Negative (FN)
Current Negative	False Positive (FP)	True Negative (TN)

Table 3
Hyper parameters of the reservoir.

N	α	ρ	ω	η	PCA
500	0.8976	0.9984	0.5988	0.0008	100

The best hyper parameters of the reservoir are shown in Table 3. For the classification layer, the architecture of the MLP network (Fig. 4) is made by means of hidden layers. For the regularisation, we consider a dropout of 20 (0.2) with an adaptive learning rate of 0.01 and with a minibatch of 32. In the output layer, we apply the softmax function, and for the gradient, we use "Adam". After the training phase, the best accuracy is 98.36% in the test data with 99.19% f1 score. In Table 4, we show the summary of the algorithm's efficiency.

Fig. 6 displays the confusion matrix of the test data. The diagonal represents the hits and the values outside the diagonal correspond to the erroneously classified data.

After the training step, we apply our trained network to the entire parameter space. The phase is done as follows: For each parameter value, we generate time series, repeating this procedure throughout the parameter space, we obtain a data set relating the parameter values

Table 4
Algorithm's efficiency.

accuracy	0.9836
F1 score	0.9919
Precision	0.9839
Recall	1.0

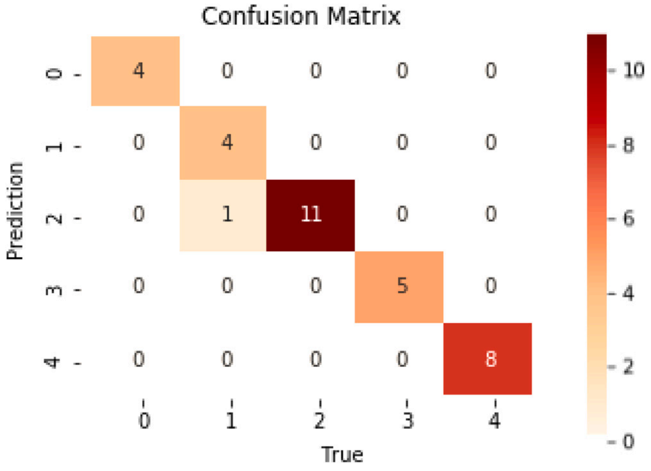


Fig. 6. Confusion matrix of the test data.

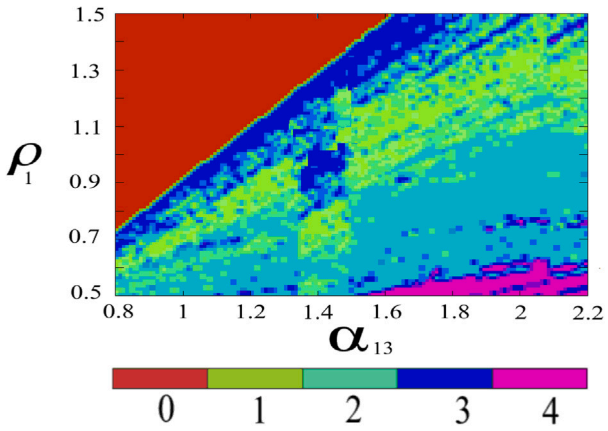


Fig. 7. $\rho_1 \times \alpha_{13}$ for $\alpha_{23} = 0.2$ and $\alpha_{32} = 2.5$.

with their respective series. Then, we apply our algorithm to the entire dataset generated in the previous step. For each parameter values, our network is ranked between one of the above classes. For these predictions, we consider $\delta_2 = 0.5$ and $\rho_2 = 4.5$. The classes are identified by means of colours.

In Fig. 7, we vary ρ_1 and α_{13}, fixing the other parameters. The first parameter is the growth rate of healthy cells, while the second parameter is the effect of tumour cells on the host cells. In the red region, the number of host cells are close to 1, that is the ideal parameter values. Fig. 8 displays our result for the parameter space ρ_1 versus α_{23}. The parameter α_{23} is directly related to the effect of the tumour cells on the immune system. We verify that the variation of α_{23} does not affect the number of healthy cells.

Increasing α_{32} (Fig. 9), we observe the effect of the immune system on the tumour cells. The undesired dynamic regime goes to an ideal condition. The boundary between the class 0 with the classes 1, 2, 3, 4 is a linear boundary throughout the parameter space, moreover, it always separates the class 0 from the class 3. The class 4 occurs for the lowest values of ρ_1 with its boundaries with the class 3.

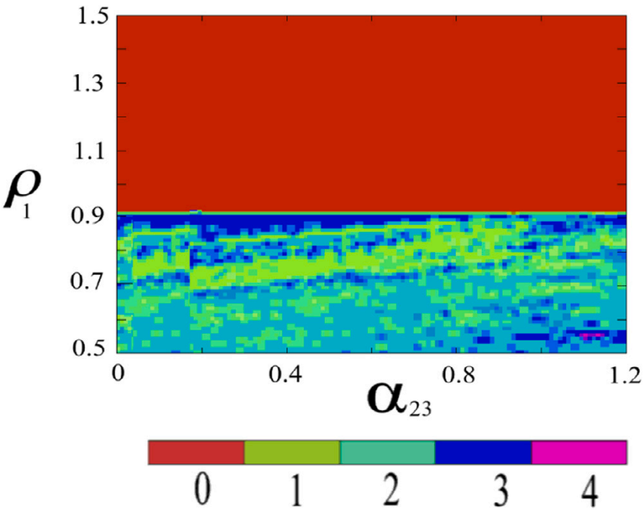


Fig. 8. $\rho_1 \times \alpha_{23}$ for $\alpha_{13} = 1$ and $\alpha_{32} = 2.5$.

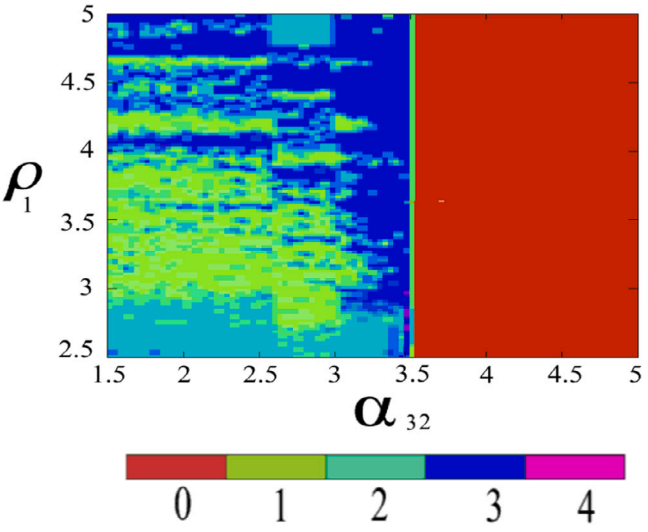


Fig. 9. $\rho_1 \times \alpha_{32}$ for $\alpha_{23} = 0.2$ and $\alpha_{13} = 1.2$.

To improve the condition of a patient, it is important to work with ρ_1 and α_{32}. The variation of these parameters changes a regime from the number of host cells oscillating close to zero to a regime in which the number of host cells are constant in time and close to the maximum value. From the therapeutic point of view, the non-linear term $\alpha_{32}yz$ is interesting to be considered.

5. Conclusion

In this work, we analysed the dynamics of a cancer model that describes the interactions among host, effector immune, and tumour cells. The model focuses on the generic interaction and how the host cells (healthy tissue cells) evolve, survive, and die in the encounter with the effector immune and tumour cells.

We use a Bidirectional Deep-readout Echo State Network (BDESN) to investigate the values of the parameters that the cancer model keeps the dynamics of the host cells oscillating close to 1. In this network, the reservoir model is optimised by Bayesian method and the output dimensions of the reservoir decrease using PCA. To classify the various time series, resulting from changing the parameters of the cancer model, into 5 classes, a multilayer perceptron network is applied. Among several parameters of the cancer model, growth rate of host cells and the effect of the immune system on tumour cells are the two most important parameters to consider for improving a patient's condition. Furthermore, our results reveal when the tumour–immune interactive dynamics are greater than 3.5, the host cells will remain at their maximum value. An effective cancer treatment starts by increasing the rate that is associated with the death of cancer cells caused by cells of the immune system and the rate of the growth of the host cells. This aspect occurs due to the fact that α_{32} is directly related to the death of the tumour cells, while p_1 acts only on the growth of the host cells. It means that the tumour cells are well controlled by the immune system.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

This work was possible by partial financial support from the following Brazilian government agencies: Fundação Araucária, FAPESP (2022/04251-7), CNPq (301019/ 2019-3, 311168/2020-5, 403120/ 2021-7), and CAPES (888 81.143103/2017-01).

References

[1] Adam J, Bellomo N. A survey of models for tumor immune dynamics. Boston; 1997.
[2] Blows WT. The biological basis of nursing: cancer. London: Routledge; 2005.
[3] Moore JS. The immunological basis of cancer, in nursing patients with cancer: principles and practice. Edinburgh: Elsevier; 2006, p. 115.
[4] Sun Y, Campisi J, Higano C, Beer TM, Porter P, Coleman I, et al. Nat Med 2012;18(9):1359.
[5] Khajanchi S. Int J Nonlinear Sci Numer Simul 2019;20:269.
[6] Forys U. Math Methods Appl Sci 2009;33:2287.
[7] Gatenby RA, Vincent TL. Mol Cancer Ther 2003;2:919.
[8] Gatenby RA, Vincent TL. Cancer Res 2003;63:6212.
[9] Borges FS, Iarosz KC, Ren HP, Batista AM, Baptista MS, Viana RL, et al. BioSystems 2014;116:43.
[10] López AG, Iarosz KC, Batista AM, Seoane JM, Viana RL, Sanjuán MAF. Commun Nonlinear Sci Numer Simul 2019;70:307.
[11] López AG, Iarosz KC, Batista AM, Seoane JM, Viana RL, Sanjuán MAF. Commun Nonlinear Sci Numer Simul 2019;79:104918.
[12] Iarosz KC, Borges FS, Batista AM, Baptista MS, Siqueira RAN, Viana RL, et al. J Theoret Biol 2015;368:113.
[13] Trobia J, Tian K, Batista AM, Grebogi C, Ren HP, Santos MS, et al. Commun Nonlinear Sci Numer Simul 2021;103:106013.
[14] Mahlbacher GE, Reihmer KC, Frieboes HB. J Theoret Biol 2019;469:47.
[15] Ghosh S, Banerjee S. Theory Biosci 2018;137:67.
[16] Butcher JB, Verstraeten D, Schrauwen B, Day CR, Haycock PW. Neural Netw 2013;38:76.
[17] Manjunath G, Jaeger H. Neural Comput 2013;25:671.
[18] Haynes ND, Soriano MC, Rosin DP, Fischer I, Gauthier DJ. Phys Rev E 2015;91:020801.
[19] Zhang C, Jiang J, Qu SX, Lai YC. Chaos 2020;30.
[20] Itoh Y, Uenohara S, Adachi M, Morie T, Aihara K. Chaos 2020;30:013128.
[21] Itik M, Banks SP. Int J Bifurcation Chaos 2010;20:71.
[22] Izquierdo-Kulich E, Allonso-Becerra E, Nieto-Villar JM. J Mod Phys 2011;2:615.
[23] Toker D, Sommer FT, D'Esposito M. Commun Biol 2020;3:11.
[24] Heltberg ML, Krishna S, Jensen MH. Nature Commun 2019;10:71.
[25] Obcemea C. Chaotic dynamics of tumor growth and regeneration. In: Unifying themes in complex systems. Berlin: Springer-Verlag; 2006, p. 349, Chapter 34.
[26] De Pillis LG, Radunskaya A. Math Comput Model 2003;37:1221.
[27] Denis F, Letellier C. Cancer Radiother 2012;16:230.
[28] Denis F, Letellier C. Cancer Radiother 2012;16:404.
[29] Kuznetsov V, Makalkin I, Taylor M, Perelson AS. Bull Math Bio 1994;56:295.
[30] Kirschner D, Panetta JC. J Math Biol 1998;37:235.
[31] d'Onofrio A. Phys D: Nonlin Phenom 2005;208:220.
[32] Ahmed E. Int J Theoret Phys 1993;32:353.
[33] Mayer H, Zaenker KS, Heiden UAD. Chaos 1995;5:155.
[34] Dalgleish A. QJM-An Int J Med 1999;92:347.
[35] Nani F, Freedman HI. Math Biosci 200:163, 159.
[36] Letellier C, Denis F, Aguirre LA. J Theoret Biol 2013;322:7.
[37] Michelson S, Leith J. Invasion Metastasis 1996;16:235.
[38] Panetta J. Bull Math Biol 1996;58:425.
[39] Khajanchi S. Int J Biomath 2020;13(2):2050009.
[40] Lukosevicius M, Jaeger H. Comput Sci Rev 2009;3(3):127.
[41] Bianchi FM, Scardapana S, Lokse S, Jenssen R. IEEE Trans Neural Netw Learn Syst 2021;32:2169.
[42] Tharwat A. Int J Appl Pattern Recognit 2016;3(3):197.
[43] Bianchi FM, Scardapane S, Lokse S, Jenssen R. Bidirectional deep-readout echo state networks. In: European Symposium on Artificial Neural Networks. 2018.
[44] Bianchi FM, Livi L, Alippi C. Investigating echo state networks dynamics by means of recurrence analysis. IEEE Trans Neural Netw Learn Syst 2018;29:427–39.
[45] Rodan A, Sheta AF, Faris H. Bidirectional reservoir networks trained using SVM+ privileged information for manufacturing process modeling. Soft Comput 2017;21:6811–24.
[46] Bengio Y, Lamblin P, Popovici D, Larochelle H. Advances in neural information processing systems. 2007, p. 153.