SEMARAK ILMU PUBLISHING

Journal of Advanced Research in Applied Sciences and Engineering Technology

Journal homepage:
https://semarakilmu.com.my/journals/index.php/applied_sciences_eng_tech/index
ISSN: 2462-1943

Novel Deep Learning Neural Networks for Breast Cancer Malignancy Estimation

G. Nivedhitha[1,*], P. Kalpana[1], Rajagopal R[2], Anusha Rani V[3], Ajith B. Singh[1], A. Sheik Sidthik[4]

[1] Department of Computer Science & Engineering, Sri Krishna College of Technology Coimbatore, Tamil Nadu, India
[2] Department of Electrical and Electronics Engineering, Francis Xavier Engineering College, Tirunelveli, India
[3] Department of Electronics and Instrumentation Engineering, Saveetha Engineering College, Chennai, Tamil Nadu, India
[4] Department of Electrical and Electronics Engineering, VSB College of Engineering & Technical Campus, Coimbatore, Tamil Nadu, India

ARTICLE INFO

Article history:
Received 26 August 2023
Received in revised form 6 March 2024
Accepted 30 April 2024
Available online 20 June 2024

Keywords:
Artificial Neural Network (ANN); Deep Neural Network (DNN); Adaptive Moment Estimation (ADAM); Lasso Regression (L1); Categorical Cross-Entropy (CCE)

ABSTRACT

Medicine has become one of the huge fields which has its own growth each and every day. Few diseases are still a threat to medical field. Because of its destructive nature, cancer is a disease that causes fear in many people throughout the world. However, most cancers are curable if detected early and treated with the right medical care and computer-assisted diagnosis. These days, diagnosis is more popular because it works well as a primary screening test for many diseases, particularly cancer. Deep learning is a method of artificial intelligence where the computer is given intelligence that attempts to emulate the way a human brain thinks. This research focuses on creating a deep neural network that can accurately predict the malignancy of breast cancer up to 98%, allowing medical personnel to diagnose patients more quickly and administer therapy more effectively.

1. Introduction

In the disease known as cancer, aberrant cells multiply and obliterate healthy body tissues. Cancer can affect any region of the human body, but breast cancer is the most prevalent variety [1]. In 2012 the world saw around 1.7 million new cases of breast cancer in women. Deep learning is a method of artificial intelligence where the machine acquires intelligence by attempting to replicate a human brain's reasoning process. This research looks into the aspect of how a deep learning model can be used to detect the malignancy of a breast cancer case. Malignant means that the tumour is made of cancer cells, and it can invade nearby tissues. A benign tumour isn't the same as a cancerous tumour (Malignant). It doesn't travel to other parts of the body or infiltrate surrounding tissue [2]. Due to its high frequency and status as one of the leading causes of death among women, breast cancer has been demonstrating to be a significant clinical challenge. It is currently the most frequently diagnosed malignant tumour in women and one of the most prevalent neoplasms worldwide. A correct diagnosis of breast cancer in its early stages can help with decision-making,

* *Corresponding author.*
E-mail address: nivedhithag789975@gmail.com

https://doi.org/10.37934/araset.47.1.140151

action planning, and treatment effectiveness. The diagnosis of breast cancer depends on the doctor's interpretation of data from the patient's complete examinations. However, due to the numerous variables involved in the diagnostic process, correctly identifying cancer is a complex and challenging process. Medical expertise is essential for the analysis of diagnostic tests, especially when determining the stage of cancer. Deep learning (DL) is a sort of machine learning (ML) that is part of the larger family of ML and is based on artificial neural networks (ANN) [3].

The idea of connections between neurons, which are the components involved in information processing, and how they communicate with one another through connections, which stand in for synapses, is related to the representation of knowledge in neural networks. Neural networks primarily take their cues from how the human brain is organised, structured, and functions. The structure of the network, with a focus on the connections, or synapses, between the network's constituent neurons, serves as a physical representation of the knowledge of artificial neural networks (ANNs). In the computer-simulated artificial neural networks, each connection is given a numerical number called a synaptic weight that describes the strength of the connection between two neurons. An ANN learns by adapting its synaptic weights, utilizing the appropriate activation functions and neural network training techniques. It is highly challenging for humans to analyse and evaluate the knowledge that an ANN has learned because it is encoded in the network's connectivity structure and the weights attached to these connections. The understanding of neural networks can be reduced to a collection of numerical values that represent connections and, in turn, the behaviour of the network.

1.1 Artificial Neural Network (ANN)

A type of artificial intelligence known as an artificial neural network (ANN) employs a variety of optimization tools to "learn" from past experiences and then applies that knowledge to categorize new data, find novel patterns, or forecast future events. Neural network models were used in this study's analysis to identify and forecast breast cancer [4]. Reports from clinical and microscopic tests for breast cancer are compiled. Based on the patient's test findings, our diagnostic system will decide if they are benign or cancerous. After a number of tests and parameter adjustments based on theoretical knowledge and industry standards, the best classifier was found. A table for each model contains the experimental results of a breast cancer diagnosis system using neural network models, with the best parameter values selected by experiment [5]. The most effective network parameters were found to be the learning rate, number of hidden layers, and hidden layer neurons. Accuracy (AC), false-negative rate (FNR), false-positive rate (FPR), true-positive rate (TPR), and true-negative rate (TNR) are all metrics used to evaluate the effectiveness of tests. Each model's value for each of the aforementioned terms is determined, and the best classier is selected using performance comparison [6].

1.2 Database for Breast Cancer

The Wisconsin breast cancer diagnostic (WBCD) database grew out of the University of Wisconsin Hospital's efforts to reliably diagnose breast tumours using only an FNA test. The data set's goal is to categorize tumours as benign or malignant based on cell descriptions obtained from the microscopic examination. The data set includes ten features such as clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, clump thickness, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, and mitoses, as well as the classification benign and malignant [7]. There are 699 patient datasets, 458 of which are benign and 241 of which are malignant. We removed

16 attribute values from the database because they were missing, and 444 are used for training, with 260 benign and 184 malignant instances. Similarly, 239 were employed for testing, with 184 Benign and 55 Malignant instances. List of features used for classification is shown in Table 1.

Table 1
List of features used for classification

No.	Feature	No.	Feature
1	Mean radius	16	Compactness error
2	Mean texture	17	Concavity error
3	Mean perimeter	18	Concave points error
4	Mean area	19	Symmetry error
5	Mean smoothness	20	Fractal dimension error
6	Mean compactness	21	Worst radius
7	Mean concavity	22	Worst texture
8	Mean concave points	23	Worst perimeter
9	Mean symmetry	24	Worst area
10	Mean fractal dimension	25	Worst smoothness
11	Radius error	26	Worst compactness
12	Texture error	27	Worst concavity
13	Perimeter error	28	Worst concave points
14	Area error	29	Worst symmetry
15	Smoothness error	30	Worst fractal dimension

1.3 Label Encoding

Label encoding is the process of converting labels into a machine-readable numeric form. The use of those labels can then be better decided by machine learning algorithms. The encoding of the dataset is an essential pre-processing step in supervised learning [8]. Using this method, the prediction classes can be changed for the obtained dataset from MALIGNANT and BENIGN to 1,0 and 0, respectively.

1.4 Scaling

Scaling is a normalization strategy that is particularly beneficial when working with a dataset that comprises continuous features at multiple scales and you're using a model that runs in some kind of linear space (like linear regression or K-nearest neighbours). Feature scaling changes the mean and variance of the features in your dataset to zero and one [9]. This will make it easier to compare features in a linear fashion. Additionally, many models in scikit-learn require this. By using this approach, we scale our dataset.

1. 5 Confusion Matrix

Information about the actual and anticipated classifications made by a classification system is contained in a confusion matrix. The matrix's data is frequently used to evaluate the effectiveness of these systems. An example of confusion matrix for a two-class classifier is shown in Figure 1. The following common terminology has been defined for the 2-class matrix: Accuracy (AC) is the proportion of accurate forecasts among all forecasts. The true positive rate (TPR) by Rani *et al.,* [10] is the percentage of positive events that were correctly identified. The false positive rate (FPR) is the percentage of cases that should have been classified as negative but were instead considered positive. The true negative rate (TNR) is the percentage of accurately diagnosed negative cases. The

percentage of positive instances that were incorrectly labelled as negative is known as the false-negative rate (FNR) [11].

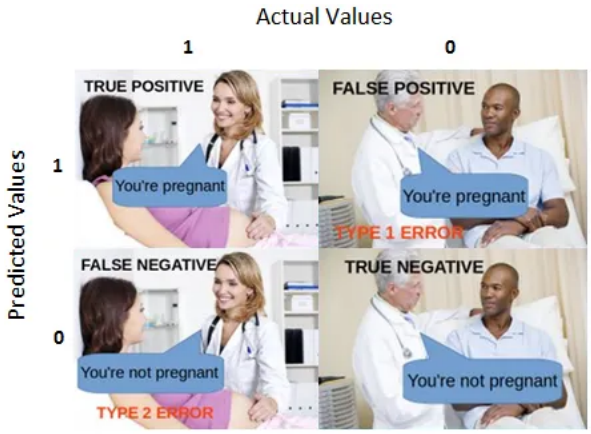

Fig. 1. Confusion matrix used for classification

2. Estimation of Adaptive Moment

Adaptive Moment Estimation (Adam) is another technique for calculating adaptive learning rates for each parameter. Adam retains both an exponentially decaying average of previous squared gradients s and an exponentially decaying average of previous gradients v, just like Adadelta and RMSprop. Adam behaves more like a heavy ball with friction, as opposed to momentum, which can be visualized as a ball rolling down a hill and prefers flat minima on the error surface [12].

$$v_t = \beta_1 v_{t-1} + (1-\beta_1) g_t \tag{1}$$

$$s_t = \beta_2 s_{t-1} + (1-\beta_2) {g_t}^2 \tag{2}$$

In the case when *v* is the first moment, it is analogous to momentum, which records the previous normalized gradient. And *s* is the second instance, as defined in adaptive gradient descent and RMSprop. It is a term that aids in determining distinct learning rates for various characteristics [13].

$${v'}_t = \frac{v_t}{1-\beta_1^t} \tag{3}$$

$${s'}_t = \frac{s_t}{1-\beta_2^t} \tag{4}$$

So, the process will be,

Step 1: while w_t do not converges
do {
Step 2: Calculate gradient $g_t = \frac{\partial f(x,w)}{\partial w}$
Step 3: Calculate $p_t = m_1 \cdot p_{t-1} + (1-m_1) \cdot g_t$
Step 4: Calculate $q_t = m_2 \cdot q_{t-1} + (1-m_2) \cdot g_t^2$
Step 5: Calculate $\widehat{p_t} = p_t/(1-m_1^t)$
Step 6: Calculate $\widehat{q_t} = q_t/(1-m_2^t)$

Step 7: Update the parameter $w_t = w_{t-1} - \alpha \cdot \widehat{p_t}/(\sqrt{\widehat{q_t}} + \epsilon$

}

Step 8: return w_t

3. Activation Functions

In a neural network, the activation function is responsible for transforming the node's summed weighted input into node activation or output for that input [14].

3.1 ReLU (Rectified Linear Unit)

The rectified linear activation function, abbreviated as ReLU, is a piecewise linear function that outputs the input directly if it is positive; otherwise, it outputs zero. This contributes to the reduction of Vanishing Gradients issues which is graphically represented in Figure 2.

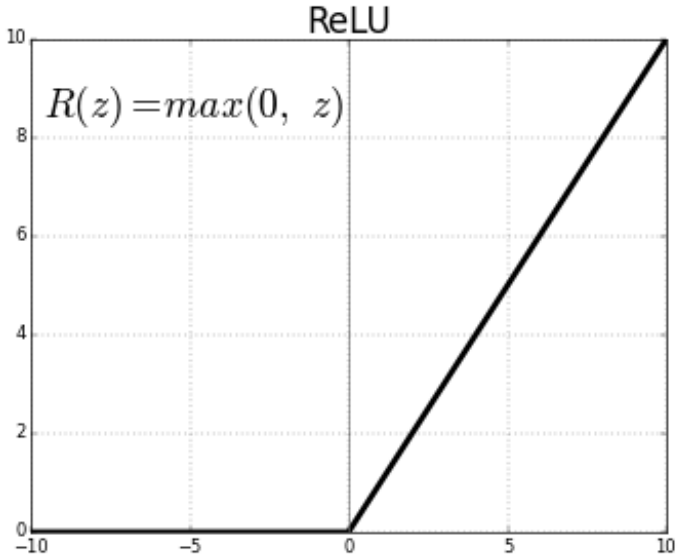

Fig. 2. Vanishing gradient issues

3.2 SoftMax

SoftMax is a mathematical function that turns a vector of integers into a vector of probabilities, where the probability of each value in the vector is proportional to their relative scale. This gives multi-class classification output in our output layer [15].

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{5}$$

4. Working of Deep Neural Network

Utilising adaptive moment estimation (ADAM), deep neural network (DNN) calculates the updated weights and biases. For optimum performance, it has the capacity to instantaneously change network weights. The graph or space that represents the error of the system for each set of weights and biases is known as the error space. The global optimum is what any training algorithm aims to

find in this search area. BPA commonly ends up in neighbourhood minima. This is because there hasn't been any attempt to cover the entire mistake space, which is incredibly complex and dimensional, or any kind of global guiding strategy. One hidden layer, input, and output were all included in the feed-forward neural network architecture used in this experiment. ReLU and SoftMax, respectively, are the activation mechanisms in neurons of the hidden layer and the output layer. Categorical Cross Entropy was the chosen performance measure. Here i, j, and k are indices referring to neurons in the input, hidden, and output layers, respectively, and p, m, k are the total number of neurons in the input, hidden, and output layers [16].

x: p X_1 input vector
h: Weighted sum of input stimuli
v: m X1 output vector of hidden layer
g: Weighted sum of v_j
Y: n X1 output vector of output layer
w_{ij}: Weight connecting i^{th} unit of output layer and j^{th} unit of hidden layer
w_{jk}: Weight connecting j^{th} unit of hidden layer to k^{th} unit of input layer
y: Actual output
y^d: Desired output

The network has computed all of its actual outputs. The following calculations have been performed

$$h_j = \sum_{k=1}^{p} W_{jk} X_k \tag{6}$$

$$v_j = \frac{1}{1 + e^{-h_j}} \tag{7}$$

$$g_i = \sum_{j=1}^{m} W_{jk} V_j \tag{8}$$

Backward pass

1. Compute the error e = y^d -y
2. Compute

$$\delta_i = y.(1-y).(y^d - y) \tag{9}$$

The error at the output unit is distributed back to the preceding layers using δ_i the following updating rule should be used to update the weights connecting the hidden layer to the output layer [12].

$$w_{ij}(t+1) = W_{ij}(t) + \delta_i h_j \tag{10}$$

$$\delta_j = V_j(1-V_j) w_{jk} \delta_j X_k \tag{11}$$

It is not essential to transmit the error back to the input layer. δ_j is used to modify the weights that connect the input and hidden layers. Update the weights between the input and hidden

layers. $W_{ij}(t+1) = W_{ij}(t) + \delta_i \delta_j x_k$ After all the δ factors have been determined, the weights for all layers adjusted simultaneously and are mentioned in Figure 3.

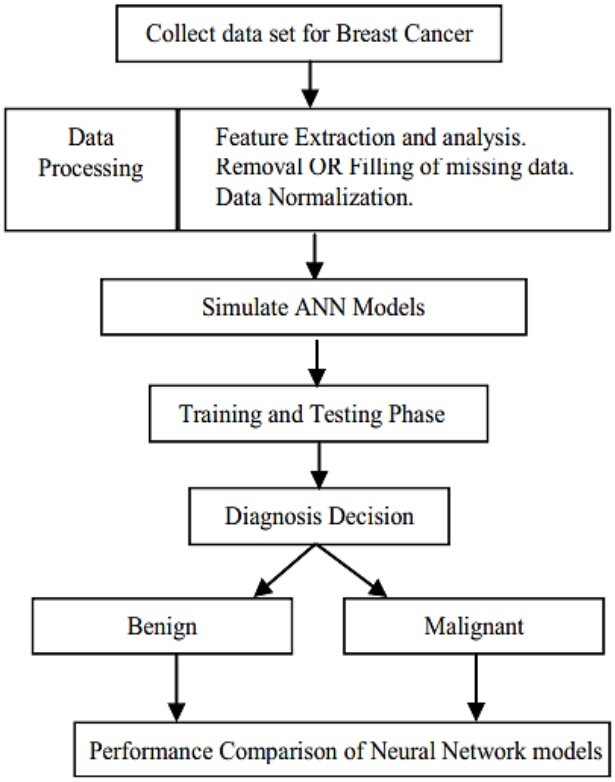

Fig. 3. Flowchart of neural network process

4.1 L1 Regularization

Regularization is a strategy for reducing mistakes and avoiding overfitting by fitting the function suitably on the supplied training set. We employ L1 and Dropout Regularization. The "absolute value of magnitude" of a coefficient is included as a penalty term to the loss function in Lasso regression (least absolute shrinkage and selection operator) [17].

$$\sum_{i=1}^{n}(Y_i - \sum_{j=1}^{p} X_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \quad (12)$$

If λ is zero, then we will get back OLS whereas very large value will make coefficients zero hence it will under-fit.

4.2 Dropout Regularization

Dropout is a training technique where neurons are rejected at random. They are arbitrarily "dropped-out" of school. This indicates that any weight updates are not applied to the cell on the backward trip and that on the forward pass, their influence on the activation of downstream neurons is abolished temporally. As a neural network learns, its neuron weights adapt to the environment. In order to provide some specialization, neuronal weights are set for specific features. This specialization makes nearby neurons dependent on it, which, if it goes too far, could lead to a brittle model that is highly specialized to the training input. The results of a neuron's reliance on the context

during training include complex co-adaptations. There may be a misconception that if neurons are arbitrarily removed from the network while training, other neurons will have to fill in and manage the representation needed to make predictions for the missing neurons. In turn, it is claimed that the network has learned a variety of unique internal representations. The network gets less sensitive to the precise weights of neurons as a result, and it also becomes more generalizable and less prone to overfitting the training set.

The DNN model with 4 hidden layers and 40 neurons in each, default decay rates for ADAM, and 100 epoch provides the best accuracy of 98% for testing for breast cancer diagnosis. There were 184 benign and 55 malignant instances in the testing collection. The trained neural network detected 138 benign instances properly (true negative) but incorrectly identified the other 46 benign cases (false positive). Similarly, 44 of the 55 malignant instances in the training set were incorrectly detected (false negative), with the remaining 11 correctly identified (true positive).

5. Results

5.1. Base Model

It has been discovered that machine learning models perform classification and detection tasks with greater accuracy. The machine learning model based on neural networks was created using TensorFlow 2.0 and its built-in library Keras. Although the model has many layers, not every layer contains the same number of neurons. Except for the final output layer, which is provided with a SoftMax activation function that provides multi-class classification accuracy, all activation functions within each layer are provided as rectified linear units (RELUs). The neural network model is free from vanishing and exploding gradient issues thanks to the rectified linear unit activation function. Figure 4 shows how to create neural network models.

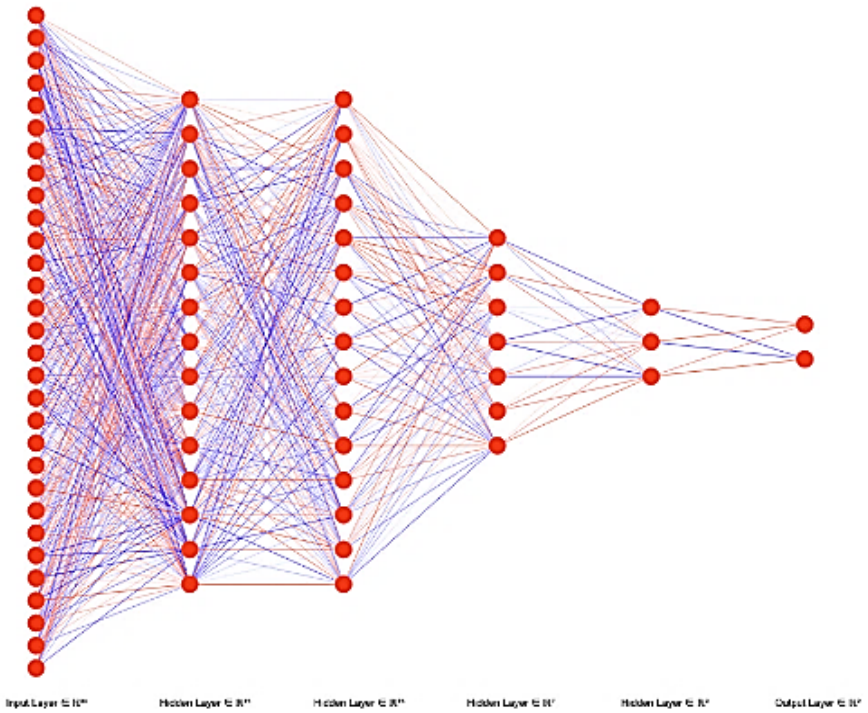

Fig. 4. Neural network progression

The first input layer's first input layer has 30 neurons since there are 30 input features. Following the first hidden layer, which has 15 neurons, is the second hidden layer. There are seven neurons in

the third hidden layer. Three neurons make form the fourth Hidden Layer, and two neurons make up the output layer. The model is trained using the training set, and it is validated using the test set over the course of 500 iterations. The relationship between this model's training loss and validation loss is illustrated in Figure 5.

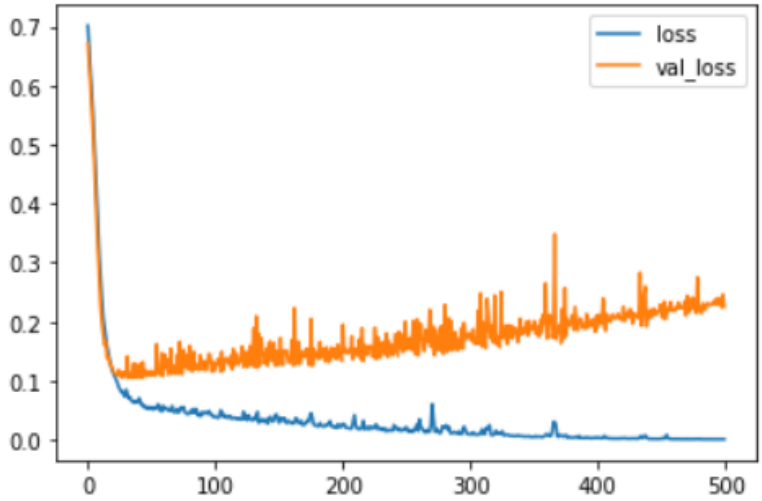

Fig. 5. Training and validation loss

The training loss is decreasing but the validation loss is increasing. Overfitting of neural networks is the term used to describe this process. This shows that the model fits the training data more closely than the validation data, which is unacceptable. The removal of the neural network's overfitting will be covered in the section that follows.

5.2 With Early Stopping, Dropouts and Lasso Regression

A dropout layer is added to each dense layer of the neural network model in Figure 4 to make it better, with the exception of the final output layer. The contingency table could be used to assess the performance of the categorization system. A Python programme created for the proposed and Keras library was used to run five simulations, and the number of false positives (FP), false positives (FN), true positives (TP), and true negatives (TN) were taken into account to make a statistical analysis. On the other hand, since NN training is a stochastic process and every time the training process was performed a different weighted network could be achieved. Table 2 and Table 3 show the conditions and the outcomes of FP, FN, TP, and TN for each simulation.

Table 2
Simulation conditions

Test		Predicted	
		Negative	Positive
Actual	Negative	a	B
	Positive	c	d
Accuracy (AC)		(a+d) / (a+b+c+d)	
True positive rate (TPR)		d / (c+d)	
False negative rate (FNR)		c / (c+d)	
False positive rate (FPR)		b / (a+b)	
True negative rate (TNR)		a / (a+b)	

Table 3
Simulation results

No	Network Architecture(I-H-O)	Learning rate	Epochs	% of TPR	% of TNR	AC (%)	Training Time (seconds)
1	30-15-7-3-2	0.01	100	94.17	94.12	94.14	68.1
2	30-15-10-7-3-2	0.01	100	94.17	94.12	94.14	67.95
3	30-15-15-7-3-2	0.01	100	95.83	95.8	98.02	70.76
4	30-15-15-5-3-2	0.05	100	93.33	93.28	93.31	70.82
5	30-15-15-7-2	0.001	100	95.00	94.96	94.98	74.62

According to the Table 2, there is a sizable discrepancy between the number of TP and TN and the number of FP and FN. This indicates that the system's classification rate is high and that it has good sensitivity and specificity. Since FP and FN amount are closely related variables, it is also possible to evaluate the specificity and sensitivity levels using these quantities. Indeed, the sensitivity decreases with increasing FN quantity, while the FN specificity decreases with increasing FP amount. By examining the numerous simulations carried out, it can be shown that the classification system is more specific than sensitive because the FN amount is somewhat higher than the FP amount. From Table 2 it is clear that simulation number 3 gives the best performance out of 5 simulations performed. The success Model uses ANN in the configuration of 30, 15, 15, 7, 3, 2 neurons respectively as shown in Figure 4.

A dropout layer is added to each dense layer of the neural network model in Figure 4 to make it better, with the exception of the final output layer. The most effective neural network design. The percentage of neurons that are turned off during the training phase is indicated by the value entered in the dropout parameter. For each epoch, a layer's neurons that will be disabled are chosen at random. This model's early stopping feature is engaged in order to keep track of the validation loss measure.

Figure 6 shows that the training was terminated after the 146th training epoch. In contrast to the previous instances, the training loss and validation loss are converging more favourably, which shows that the model is not overfitting the training data. This provides us good validation accuracy when used with lasso regression over the learning parameters.

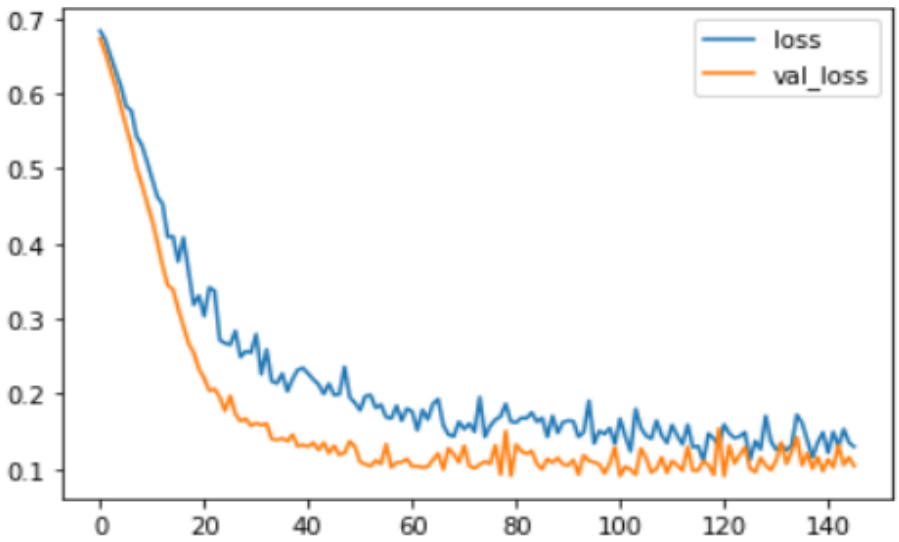

Fig. 6. Improved training and validation loss

6. Conclusion

The investigation focused on using ANN to diagnose breast cancer. In this way, Keras just repaired an MLP neural network. The outcomes demonstrate that the network provides good training accuracy as well as good validation and test accuracy. High validation and test accuracy was accomplished using the Dropout, L1, and Early Stopping Regularization methods, while the High Training Accuracy was achieved using the ADAM Optimizer and ReLU Activation Function. The optimal iteration of our network model, one that performs better and is more algorithmically resilient due to its increased sensitivity, specificity, and accuracy, was created by tuning the hyper parameters in each network architecture. It can categorize situations accurately in 98% of cases, demonstrating an excellent rating system. This enables to draw the conclusion that the proposed ANN model can be used to address the issue of breast cancer diagnosis. Future simulations of various structures and activation mechanisms will aim to improve categorization outcomes.

Acknowledgement
There is no acknowledgement involved in this work.

References
[1] Prakash, Sidharth S., Binsu C. Kovoor, and K. Visakha. "Convolutional neural network based malaria parasite infection detection using thin microscopic blood smear samples." In *2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA)*, pp. 308-313. IEEE, 2020. https://doi.org/10.1109/ICIRCA48905.2020.9182944

[2] Ara, Sharmin, Annesha Das, and Ashim Dey. "Malignant and benign breast cancer classification using machine learning algorithms." In *2021 International Conference on Artificial Intelligence (ICAI)*, pp. 97-101. IEEE, 2021. https://doi.org/10.1109/ICAI52203.2021.9445249

[3] Janghel, Rekh Ram, Anupam Shukla, Ritu Tiwari, and Rahul Kala. "Breast cancer diagnosis using artificial neural network models." In *The 3rd International Conference on Information Sciences and Interaction Sciences*, pp. 89-94. IEEE, 2010. https://doi.org/10.1109/ICICIS.2010.5534716

[4] Arjaria, Siddhartha Kumar, and Devshri Roy. "Learning Content Management Using Machine Learning." *International Journal of Information and Education Technology* 2, no. 5 (2012): 472. https://doi.org/10.7763/IJIET.2012.V2.182

[5] Wu, Yuyao. "Diagnosis of Breast Cancer Based on Support Vector Machine and Random Forest Methods." In *2020 International Conference on Computing and Data Science (CDS)*, pp. 147-151. IEEE, 2020. https://doi.org/10.1109/CDS49703.2020.00036

[6] Sathiyanarayanan, P., S. Pavithra, M. Sai Saranya, and M. Makeswari. "Identification of breast cancer using the decision tree algorithm." In *2019 IEEE International conference on system, computation, automation and networking (ICSCAN)*, pp. 1-6. IEEE, 2019. https://doi.org/10.1109/ICSCAN.2019.8878757

[7] Amrane, Meriem, Saliha Oukid, Ikram Gagaoua, and Tolga Ensari. "Breast cancer classification using machine learning." In *2018 electric electronics, computer science, biomedical engineerings' meeting (EBBT)*, pp. 1-4. IEEE, 2018. https://doi.org/10.1109/EBBT.2018.8391453

[8] Guna, G., D. Prabhakaran, and M. Thirumarimurugan. "Design, implementation, control and optimization of single stage pilot scale reverse osmosis process." *Water Science and Technology* 84, no. 10-11 (2021): 2923-2942. https://doi.org/10.2166/wst.2021.302

[9] Guna, G., D. Prabhakaran, and M. Thirumarimurugan. "Active disturbance rejection control (ADRC) for a pilot scale reverse osmosis system." *Journal of Environmental Protection and Ecology* 20, no. 4 (2019): 1923-1934.

[10] Rani, V. A., G. Guna, D. Prabhakaran, and M. Thirumarimurugan. "Temperature control in an exothermic continuous stirred tank reactor." (2023). https://doi.org/10.4038/jnsfsr.v51i1.10844

[11] Karim, Abdul Razif Abdul, and Roslina Mohammad. "Meta-study of sensitivity analysis in solar renewable energy application." *Progress in Energy and Environment* (2023): 14-25. https://doi.org/10.37934/progee.23.1.1425

[12] Chiranjeevi, Phaneendra, and A. Rajaram. "A lightweight deep learning model based recommender system by sentiment analysis." *Journal of Intelligent & Fuzzy Systems* Preprint (2023): 1-14. https://doi.org/10.3233/JIFS-223871

[13] Sathiyaraj, K., and A. Rajaram. "An optimized design modelling of neural network based green house management system using solar and rectenna." *DYNA–Ingen. Ind* 97, no. 1 (2022): 85-91. https://doi.org/10.6036/10089

[14] Yagoub, Sami Abdelrahman Musa, Gregorius Eldwin Pradipta, and Ebrahim Mohammed Yahya. "Prediction of bubble point pressure for Sudan crude oil using Artificial Neural Network (ANN) technique." *Progress in Energy and Environment* (2021): 31-39.

[15] Zulkifli, Mohamad Dzulhilmi, Mostafa Yusefi, Kamyar Shameli, and Sin Yeang Teow. "Curcumin Extract Loaded with Chitosan Nanocomposite for Cancer Treatment." *Journal of Research in Nanoscience and Nanotechnology* 6, no. 1 (2022): 1-13. https://doi.org/10.37934/jrnn.6.1.113

[16] Kek, Hong Yee, Huiyi Tan, Desmond Daniel Chin Vui Sheng, Yi Lee, Nur Dayana Ismail, Muhd Suhaimi Deris, Haslinda Mohamed Kamar, and Keng Yinn Wong. "A CFD assessment on ventilation strategies in mitigating healthcare-associated infection in single patient ward." *Progress in Energy and Environment* (2023): 35-45. https://doi.org/10.37934/progee.24.1.3545

[17] Indira, D. N. V. S. L. S., Rajendra Kumar Ganiya, P. Ashok Babu, A. Xavier, L. Kavisankar, S. Hemalatha, V. Senthilkumar et al. "Improved artificial neural network with state order dataset estimation for brain cancer cell diagnosis." *BioMed Research International* 2022 (2022). https://doi.org/10.1155/2022/7799812