International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Breast Cancer Diagnosis: Integrating Constructive Deep Neural Network

Janhvi A. Chauhan[1], Dhaval M. Modi[2]

[1, 2]Lecturer, Government Polytechnic, Ahmedabad

Abstract

The Oncotype DX (ODX) breast cancer assay is the most widely used Gene Expression Profiling (GEP) test globally. It plays a significant role in guiding decisions regarding Adjuvant Chemotherapy (ACT). Despite the availability of several standard approaches for this purpose, their accuracy has yet to reach optimal levels.This paper focuses on Breast Cancer Computer-Aided Diagnosis (BC-CAD) using a Deep Constructive Neural Network to predict the Recurrence Score (RS) provided by the ODX assay. The proposed ConstDeepNet algorithm was evaluated by developing two types of classifiers: the first uses a "one-against-all" architecture, building a separate Deep Neural Network for each class, while the second employs a single DNN to classify all three classes. A separate network is constructed for each class in the first architecture, while the second architecture utilizes a single deep neural network to classify all three classes. The proposed BC-CAD algorithm was evaluated on a real-world dataset and demonstrated strong performance. The dataset consists of 92 cases of luminal B mammary carcinoma with available Oncotype DX test results collected between 2012 and 2017 from the Georges Francois Leclerc Cancer Centre.

Keywords: Deep Learning, Neural Networks, Breast Cancer, Recurrence Score, Oncotype DX

1. Introduction

A review of the literature reveals that there is extensive scientific research focused on Breast Cancer Computer-Aided Diagnosis (BC-CAD). The methods used in BC-CAD vary depending on whether the input data consists of medical images or clinical information. When medical imaging techniques are employed, diagnosis relies on the analysis of digital images such as histopathology or immunohistochemical (IHC) images. Two recent publications that provide comprehensive overviews of these imaging-based methods are by Aswathy and Jagannath (2016), and Saha et al. (2016).

Alternatively, breast cancer diagnosis can also be approached by examining various clinical aspects of the tumor, including both pathological and biological factors, which offer valuable prognostic and predictive insights. These clinical features may include surgical pathology biomarkers such as tumor type, tumor size, tumor grade, and lymph node status.The BC-CAD algorithm was tested on a real-world dataset comprising 92 cases of luminal B mammary carcinoma, with available Oncotype DX results collected between 2012 and 2017 from the Georges Francois Leclerc Cancer Centre in Dijon and the North Trévenans County Hospital in Belfort, France. The results demonstrate promising performance of the proposed approach. Machine learning techniques for computer-aided diagnosis are commonly developed and evaluated using the well-known Wisconsin Breast Cancer Dataset (WBCD) (Lichman,

2013), which is available through the University of California at Irvine (UCI) machine learning repository (see Abdel-Zaher and Eldeib, 2016; Asri et al., 2016; Devi and Deepika, 2015).

In cases where breast cancer is detected, the modern management of early-stage estrogen receptor-positive (ER+) breast cancer relies on accurately determining which patients will benefit from additional adjuvant chemotherapy, as opposed to those who may only require hormonal therapy (Romo-Bucheli et al., 2017). The prognosis and treatment decisions for tumors with varying levels of aggressiveness are often guided by the Oncotype DX (ODX) gene expression assay (M.B. Flanagan, 2008). Developed by Genomic Health Inc. (GHI), the ODX assay measures the expression of a panel of 21 genes in tumor tissue (Klein et al., 2013; Ademuyiwa et al., 2011; Carlson and Roth, 2013). The results are summarized as a Recurrence Score (RS), which ranges from 0 to 100 and is categorized into three risk groups: class 1 for low risk (RS < 18), class 2 for intermediate risk (18 ≤ RS < 31), and class 3 for high risk (RS ≥ 31). Typically, more aggressive (high-risk) cancers require adjuvant chemotherapy, whereas less aggressive (low-risk) cases are effectively managed with hormonal therapy alone. However, the Oncotype DX test is costly, access to specialized laboratory equipment is limited, and the turnaround time from biopsy to prognostic prediction can be lengthy.

The objective of this study is to predict the Recurrence Score (RS) of the Oncotype DX (ODX) assay using histological and immunohistochemical features of invasive breast carcinoma. To achieve this, we employed a Constructive Deep Neural Network for RS prediction. Two classification architectures were compared (see Figure 1):

- In the first architecture (classifier No1), a separate neural network with a single binary output neuron is used for each class.
- In the second architecture (classifier No2), a single neural network with two output neurons is utilized.

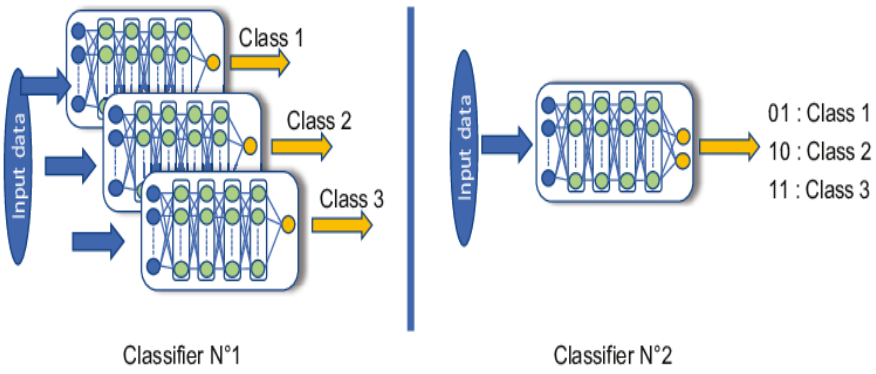

Fig. 1. The used structures of the classifiers tested for the RS prediction

The structure of this paper is as follows: Section 2 offers a concise overview of Constructive Neural Networks. Section 3 details the proposed Constructive Deep Neural Network algorithm. In Section 4, we outline the metrics and performance evaluation methods used to determine the convergence criteria for the constructive algorithm. Section 5 presents the data description and experimental results. Finally, Section 6 concludes the paper and discusses potential directions for future research.

2. Constructive Neural Network: State of The Art

Determining the optimal neural network architecture to maximize performance in function approximation or classification tasks is a challenging research problem. Key questions include how to select the ideal number of neurons per layer and the appropriate number of layers. Typically, users experiment with various network topologies to identify the best structure, which becomes especially cumbersome and resource-intensive when dealing with deep architectures (Chandra and Sharma, 2016; Hosseini-Asl et al., 2016; Schmidhuber, 2015; LeCun et al., 2015). Even after extensive testing, there is no guarantee that the chosen number of hidden units is truly optimal.Traditional learning algorithms require the network topology to be defined in advance, adjusting only the connection weights during training. However, to better match the complexity and size of the data, it is beneficial to allow the neural network structure to adapt dynamically throughout the learning process. For further reading, see Pérez-Sánchez et al. (2016), Ding et al. (2013), Franco and Jerez (2009), Parekh et al. (2000, 1997), and Yao (1993).

Adaptive neural networks generally fall into three categories:

- **Constructive or Growing Algorithms:**

  These methods progressively build the neural network by adding one neuron and its connections at each training step. Typically, training begins with a minimal structure, such as a single hidden layer containing one neuron. A well-defined convergence criterion is essential to halt network growth at the right moment; otherwise, poor criterion selection can result in overfitting (Qiao et al., 2016; Zemouri, 2017; Islam et al., 2009a; Puma-Villanueva et al., 2012; Subirats et al., 2012; Lan et al., 2010; Islam et al., 2009b; Zemouri and Zerhouni, 2012).

- **Pruning Algorithms:**

  In contrast to constructive methods, pruning algorithms start with a network containing the maximum number of layers and neurons. During training, neurons and their connections—or specific connections within the network—are systematically removed. As with constructive algorithms, a suitable convergence criterion is necessary to determine when to stop pruning. One major challenge of this approach is deciding on the initial network size (Fnaiech et al., 2004; Srivastava et al., 2014; Fnaiech et al., 2011, 2009; Miche et al., 2010; Lauret et al., 2006; Xu and Ho, 2006).

- **Hybrid Methods:**

  These techniques offer a compelling alternative by combining both constructive and pruning strategies. Typically, the network is first expanded during a constructive phase and then pruned to prevent overfitting. As with the other methods, defining an appropriate convergence criterion is crucial (Han and Qiao, 2013; Yang and Chen, 2012; Narasimha et al., 2008; Islam et al., 2009a; Wu et al., 2015).

In practice, hybrid methods are particularly attractive, often delivering better results than using growing or pruning algorithms alone.

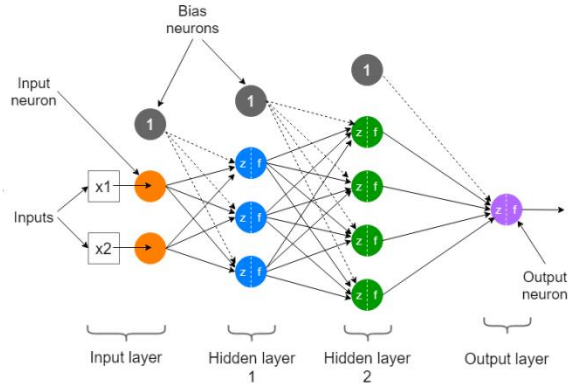

Fig. 2. Deep Neural Network with m hidden layers

3. The Constructive Deep Neural Network

A Deep Multi-Layer Perceptron (MLP) with m hidden layers (see Fig. 2) can be formally defined using two parameters: Λ and Φ. The vector Λ specifies the number of neurons n_l in each layer$l$: $\Lambda = (n_0,n_1,...,n_l,...,n_m,n_{m+1})$, where l=0 corresponds to the input layer. The vector Φ represents the weight connections vector:$\Phi=(W^1,W^2,...,W^l,...,W^{m+1})$. Each W^l is a weight matrix, and each element w^l_{ij} within W^l denotes the connection weight between the i^{th} neuron in layer l and the j^{th} neuron in layer $(l-1)$.

The proposed Constructive Deep Neural Network (ConstDeepNet) is a deep learning architecture that evolves incrementally during training (Zemouri, 2017; Zemouri et al., 2018). Let n_l^t denote the size of hidden layer l at iteration t. To prevent infinite loops during the construction process, two parameters must be defined: Max_{HL} (the maximum number of hidden layers) and Max_n (the maximum number of neurons per layer), both set by the user. The detailed procedure for ConstDeepNet is outlined in Algorithm 1.

Algorithm 1 Deep Constructive Algorithm

Data:
M : size of training data set
N_{iter} : the number of iterations for the training algorithm
t : training step index. At each stept, $(N_{iter}\times M)$ iterations of the training algorithm are computed.
L : The index of the current hidden layer (HL)
W^l : Matrix of the connections between the layers l and l-1
ConvCond : Convergence condition

Result:
Deep Neural Network

Initialization:
$t \leftarrow 0$
ConvCond=False
Initialize the CNN with one HL and one neuron (l=1)
whileConvCond = False **do**

if ($n^t_l \leq Max_n$ and ConvCond = False) **then**
add a new neuron ($n^{t+1}_l = n^t_l$+1) for *HL(l)*,
initializerandomlyall thenewweights,
end
ifn^t_l>$Max_n$and $l \leq Max_{HL}$ andConvCond= False**then**
addanewhiddenlayer(l=l+1)withone neuron,
initializerandomlyall thenewweights,
end
if$l$>Max_{HL}andConvCond=False**then**
Stoptheconstructiveprocedure(failed)
end
UpdateonlytheweightsW^lofthe*HL(l)*
t⇐t+1
end
fine tuned of the last layerW^{l+1} with (N_{iter}×M) iterations

The Deep Neural network is successfully built

4. Performance Evaluation and Convergence Condition

To evaluate the performance of different classification methods, it is essential to introduce quantitative criteria. A confusion matrix is commonly used for this purpose. This matrix provides information about both the actual and predicted classifications.

- **True Positives (TP)** refer to the number of positive instances correctly identified as positive.
- **True Negatives (TN)** are the number of negative instances correctly identified as negative.
- **False Positives (FP)** represent negative instances that are incorrectly classified as positive.
- **False Negatives (FN)** are positive instances that are incorrectly classified as negative.

Using these values, several performance metrics can be calculated (see Table 1).

Table 1. Summary of the used metrics

Accuracy (Acc)	$\frac{TP + TN}{TP + TN + FP + FN}$
Negative Predictive Value (NPV)	$\frac{TN}{TN + FN}$
Positive Predictive Value (PPV)	$\frac{TP}{TP + FP}$
True Negative Rate (TNR)	$\frac{TN}{TN + FP}$
True Positive Rate (TPR)	$\frac{TP}{TP + FN}$

The convergence condition used by the Constructive Deep Neural Network algorithm is:
If **Accuracy > θ**, **PPV > θ**, and **TPR > θ**, then

ConvCond = True

Else

ConvCond = False

where **θ** is a threshold value defined by the user.

5. Data Description and Result

The study dataset comprises 92 cases of luminal B mammary carcinoma, all with available Oncotype DX test results collected between 2012 and 2017. These cases were sourced from the Georges François Leclerc Cancer Centre in Dijon and the North Trévenans County Hospital in Belfort, France.

The **Recurrence Score (RS)** was determined using ten input features: patient age, tumor size, lymph node (ganglionic) status, four different tumor grading parameters, estrogen receptor (ER) status, progesterone receptor (PR) status, and Ki-67 expression (see Table 2).Based on the RS, the cases were divided into three risk categories:

- **Class 1**: Low risk – 40 cases
- **Class 2**: Intermediate risk – 38 cases
- **Class 3**: High risk – 12 cases

The dataset was split equally: the first half was used for training the model, and the second half for testing.

Figure 4 illustrates a 2D reduced representation of the dataset using **t-distributed Stochastic Neighbor Embedding (t-SNE)**, a technique introduced by van der Maaten and Hinton (2008). t-SNE is a dimensionality reduction method especially effective for visualizing high-dimensional data. It can be implemented using the Barnes-Hut approximation, enabling it to handle large, real-world datasets efficiently.

Table 2. Patient and tumor characteristics

Characteristic	n (%)	Characteristic	n (%)
Age		Mitosies Grade	
<40 Years	1 (1.5)	1	21 (23.9)
40-49 years	5 (5.9)	2	52 (55.3)
50-59 years	3 (2.8)	3	18 (21.3)
> 59 years	84 (91.8)	Nuclei Grade	
Tumor size (cm)		1	1 (1.5)
< 1.0	14 (14.1)	2	43 (46.8)
1.1 -2.0	41 (43.5)	3	48 (53.4)
2.1-4.0	37 (39.1)	Glande grade	
> 4.0	3 (3.2)	1	1 (1.6)
Ganglionic Status		2	26 (31.2)
0	61 (66.3)	3	64 (67.3)
1	31 (33.7)	Progesterone Receptor	

SBR Grade		< 10	11 (12)
1	9 (9.8)	10-20	12 (14)
2	43 (46.7)	> 20	70 (78)
3	40 (43.5)		

The proposed **ConstDeepNet** algorithm was tested to build two classifiers, as illustrated in Figure 1. In the first architecture, a *"one-against-all"* structure is employed, where a separate deep neural network (DNN) is trained for each class. If an input feature belongs to class *i*, the output of the *i-th* DNN is 1; otherwise, the output is 0.In the second architecture, a single DNN is used to classify all three classes. Here, each class (1, 2, and 3) is encoded using two binary output neurons as follows:

- Class 1: 01
- Class 2: 10
- Class 3: 11

Figure 5 presents the comparative performance results for each class across the metrics: **Accuracy (Acc)**, **Positive Predictive Value (PPV)**, and **True Positive Rate (TPR)**. Different convergence conditions were evaluated by varying the threshold **θ** from 0.1 to 0.9. To assess the algorithm's repeatability, the **ConstDeepNet** algorithm was executed 100 times for each value of θ. The plots in Figure 5 show the average performance values obtained for each metric.

- The **red plots (a)** represent the results from the first classifier.
- The **blue plots (b)** correspond to the second classifier.

Table 3 summarizes the performance of both classifiers in predicting the **Oncotype DX Recurrence Score (RS)** for θ = 0.9.

	Low Risk		Inter. Risk		High Risk	
Classifier	#1	#2	#1	#2	#1	#2
Accuracy	0.75	0.72	0.59	0.53	0.86	0.88
NPV	0.70	0.69	0.58	0.54	0.93	0.90
PPV	0.69	0.69	0.49	0.46	0.49	0.45
TNR	0.79	0.75	0.69	0.68	0.95	0.93
TPR	0.53	0.49	0.35	0.34	0.52	0.41
Total	3.46	3.34	2.70	2.55	3.75	3.57

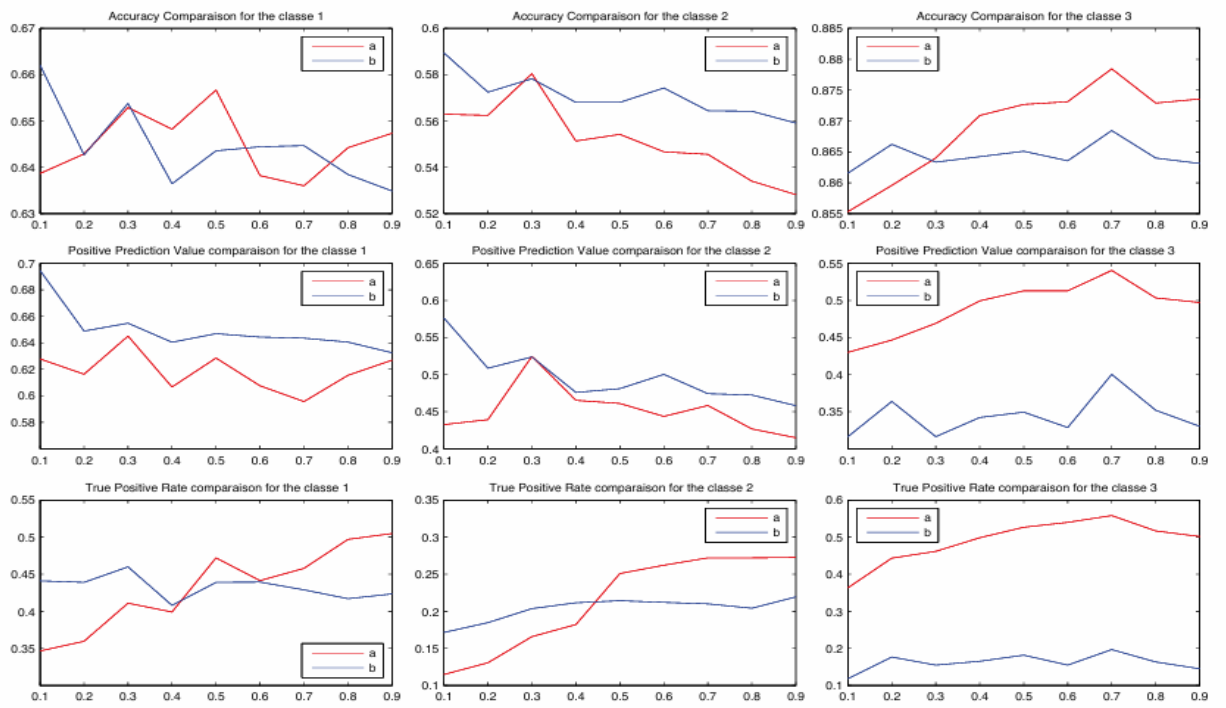

Figure 5 Based on the variation in the convergence threshold θ, the results were produced for each class and metric. The results of the first classifier are shown in plot a (red), and those of the second classifier are shown in plot b (blue).

From both Table 3 and Figure 5, it is evident that the first classifier performs better for the **Low** and **High risk** classes. However, both classifiers show poor performance for the **Intermediate risk** class. This can be attributed to the data's topography, as shown in Figure 4. Many data points in the Intermediate class overlap with those of the Low and High classes, creating a fuzzy boundary. This indistinct separation is largely due to the subjective nature of certain tumor characteristics. Apart from age and tumor size, most clinicopathological features are "human-sensitive" and rely heavily on expert interpretation. Data accuracy remains a recurring challenge in biological applications (Mahmud et al., 2018).

A recent study by **Sparano et al. (2018)** on **Adjuvant Chemotherapy Guided by a 21-Gene Expression Assay in Breast Cancer** concluded that the benefit of chemotherapy remains uncertain for most patients with an Intermediate RS score. This aligns with our findings, emphasizing the need for more accurate prediction of RS for the **Low** and **High risk** classes.

6. Conclusion

The objective of this study was to predict the **Recurrence Score (RS)** of the **Oncotype DX** test. A novel constructive algorithm was proposed to develop two deep neural network (DNN) architectures. The first architecture consists of one dedicated DNN per class, while the second employs a single DNN to classify all three classes.The results obtained are promising and demonstrate that deep neural networks can effectively predict the Oncotype DX Recurrence Score. Further research and testing with larger datasets are planned to enhance prediction accuracy.Future work will focus on improving the reliability of the input data to achieve better separation among the three RS classes. In practice, the first

architecture offers several advantages, particularly in the context of medical diagnosis. Its key benefit lies in the multidimensional output—providing an individual response for each class—which plays a crucial role in improving the overall classification accuracy.

References

1. Abdel-Zaher, A. M., &Eldeib, A. M. (2016). Breast cancer classification using deep belief networks. *Expert Systems with Applications*, 46, 139–144.
2. Ademuyiwa, F. O., Miller, A., O'Connor, T., Edge, S. B., Thorat, M. A., Sledge, G. W., Levine, E., & Badve, S. (2011). The effects of Oncotype DX recurrence scores on chemotherapy utilization in a multi-institutional breast cancer cohort. *Breast Cancer Research and Treatment*, 126(3), 797–802.
3. Asri, H., Mousannif, H., Moatassime, H. A., & Noel, T. (2016). Using machine learning algorithms for breast cancer risk prediction and diagnosis. *Procedia Computer Science*, 83, 1064–1069.
4. Aswathy, M., & Jagannath, M. (2016). Detection of breast cancer on digital histopathology images: Present status and future possibilities. *Informatics in Medicine Unlocked*.
5. Carlson, J., & Roth, J. (2013). The impact of the Oncotype DX breast cancer assay in clinical practice: A systematic review and meta-analysis. *Breast Cancer Research and Treatment*, 141(1), 13–22.
6. Chandra, B., & Sharma, R. K. (2016). Deep learning with adaptive learning rate using Laplacian score. *Expert Systems with Applications*, 63, 1–7.
7. Devi, R. D. H., & Deepika, P. (2015). Performance comparison of various clustering techniques for diagnosis of breast cancer. In *2015 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, 1–5.
8. Ding, S., Li, H., Su, C., Yu, J., & Jin, F. (2013). Evolutionary artificial neural networks: A review. *Artificial Intelligence Review*, 39(3), 251–260.
9. Fnaiech, N., Abid, S., Fnaiech, F., & Cheriet, M. (2004). A modified version of a formal pruning algorithm based on local relative variance analysis. In *First International Symposium on Control, Communications and Signal Processing*, 849–852.
10. Fnaiech, N., Fnaiech, F., & Jervis, B. W. (2011). Feedforward Neural Networks Pruning Algorithms. In *Industrial Electronics Handbook* (Vol. 5, Ch. 15, pp. 15-1 to 15-15). J.D. Irwin (Ed.), 2nd edition.
11. Fnaiech, N., Fnaiech, F., Jervis, B., & Cheriet, M. (2009). The combined statistical stepwise and iterative neural network pruning algorithm. *Intelligent Automation & Soft Computing*, 15(4), 573–589.
12. Franco, L., & Jerez, J. M. (Eds.). (2009). *Constructive Neural Networks* (Vol. 258). Springer-Verlag Berlin Heidelberg.
13. Han, H. G., & Qiao, J. F. (2013). A structure optimisation algorithm for feedforward neural network construction. *Neurocomputing*, 99, 347–357.
14. Hosseini-Asl, E., Zurada, J. M., &Nasraoui, O. (2016). Deep learning of part-based representation of data using sparse autoencoders with nonnegativity constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 27(12), 2486–2498.

15. Islam, M. M., Sattar, M. A., Amin, M. F., Yao, X., & Murase, K. (2009a). A new adaptive merging and growing algorithm for designing artificial neural networks. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(3), 705–722.

16. Islam, M. M., Sattar, M. A., Amin, M. F., Yao, X., & Murase, K. (2009b). A new constructive algorithm for architectural and functional adaptation of artificial neural networks. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(6), 1590–1605.

17. Klein, M. E., Dabbs, D. J., Shuai, Y., Brufsky, A. M., Jankowitz, R., Puhalla, S. L., & Bhargava, R. (2013). Prediction of the Oncotype DX recurrence score: Use of pathology-generated equations derived by linear regression analysis. *Modern Pathology*, 26, 658–664.

18. Lan, Y., Soh, Y. C., & Huang, G. B. (2010). Constructive hidden nodes selection of extreme learning machine for regression. *Neurocomputing*, 73(16), 3191–3199.

19. Lauret, P., Fock, E., & Mara, T. A. (2006). A node pruning algorithm based on a Fourier amplitude sensitivity test method. *IEEE Transactions on Neural Networks*, 17(2), 273–293.

20. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444.

21. Lichman, M. (2013). UCI Machine Learning Repository. Retrieved from http://archive.ics.uci.edu/ml.

22. Mahmud, M., Kaiser, M. S., Hussain, A., &Vassanelli, S. (2018). Applications of deep learning and reinforcement learning to biological data. *IEEE Transactions on Neural Networks and Learning Systems*.

23. Flanagan, M. B., Dabbs, D. J., et al. (2008). Histopathologic variables predict Oncotype DX recurrence score. *Modern Pathology*, 21(10), 1255–1261.

24. Miche, Y., Sorjamaa, A., Bas, P., Simula, O., Jutten, C., &Lendasse, A. (2010). OP-ELM: Optimally pruned extreme learning machine. *IEEE Transactions on Neural Networks*, 21(1), 158–162.

25. Narasimha, P. L., Delashmit, W. H., Manry, M. T., Li, J., & Maldonado, F. (2008). An integrated growing-pruning method for feedforward network training. *Neurocomputing*, 71(13), 2831–2847.

26. Parekh, R., Yang, J., &Honavar, V. (2000). Constructive neural-network learning algorithms for pattern classification. *IEEE Transactions on Neural Networks*, 11(2), 436–451.

27. Parekh, R. G., Yang, J., &Honavar, V. (1997). Constructive neural network learning algorithms for multi-category real-valued pattern classification. *Tech. Rep. ISU-CS-TR97-06146*, Department of Computer Science, Iowa State University.

28. Pérez-Sánchez, B., Fontenla-Romero, O., & Guijarro-Berdiñas, B. (2016). A review of adaptive online learning for artificial neural networks. *Artificial Intelligence Review*.

29. Puma-Villanueva, W. J., dos Santos, E. P., &Zuben, F. J. V. (2012). A constructive algorithm to synthesize arbitrarily connected feedforward neural networks. *Neurocomputing*, 75(1), 14–32.

30. Qiao, J., Li, F., Han, H., & Li, W. (2016). Constructive algorithm for fully connected cascade feedforward neural networks. *Neurocomputing*, 182, 154–164.

31. Romo-Bucheli, D., Janowczyk, A., Gilmore, H., Romero, E., &Madabhushi, A. (2017). A deep learning-based strategy for identifying and associating mitotic activity with gene expression-derived risk categories in estrogen receptor positive breast cancers. *Cytometry Part A*, 91(6), 566–573.

32. Saha, M., Mukherjee, R., & Chakraborty, C. (2016). Computer-aided diagnosis of breast cancer using cytological images: A systematic review. *Tissue and Cell*, 48(5), 461–474.

33. Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61, 85–117.

34. Sparano, J. A., et al. (2018). Adjuvant chemotherapy guided by a 21-gene expression assay in breast cancer. *New England Journal of Medicine*.

35. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., &Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15, 1929–1958.