INTERNATIONAL JOURNAL OF COMPUTERS
DOI: 10.46300/9108.2020.14.13 Volume 14, 2020

Classifications of Breast Cancer Diagnosis using Machine Learning

Hajra Naveed Iqbal
"Department of Computer Engineering
University of Sharjah
Sharjah, United Arab Emirates
U16107036@sharjah.ac.ae"

Ali Bou Nassif
"Department of Computer Engineering
University of Sharjah
Sharjah, United Arab Emirates
anassif@sharjah.ac.ae"

Ismail Shahin
"Department of Electrical Engineering
University of Sharjah
Sharjah, United Arab Emirates
ismail@sharjah.ac.ae"

***Abstract*—Breast Cancer (BC) is amongst the most common and leading causes of deaths in women throughout the world. Recently, classification and data analysis tools are being widely used in the medical field for diagnosis, prognosis and decision making to help lower down the risks of people dying or suffering from diseases. Advanced machine learning methods have proven to give hope for patients as this has helped the doctors in early detection of diseases like Breast Cancer that can be fatal, in support with providing accurate outcomes. However, the results highly depend on the techniques used for feature selection and classification which will produce a strong machine learning model. In this paper, a performance comparison is conducted using four classifiers which are Multilayer Perceptron (MLP), Support Vector Machine (SVM), K-Nearest Neighbors (KNN) and Random Forest on the Wisconsin Breast Cancer dataset to spot the most effective predictors. The main goal is to apply best machine learning classification methods to predict the Breast Cancer as benign or malignant using terms such as accuracy, f-measure, precision and recall. Experimental results show that Random forest is proven to achieve the highest accuracy of 99.26% on this dataset and features, while SVM and KNN show 97.78% and 97.04% accuracy respectively. MLP shows the least accuracy of 94.07%. All the experiments are conducted using RStudio as the data mining tool platform.**

***Keywords*—breast cancer, machine learning, classification, feature selection**

I. Introduction and Motivation

Breast Cancer (BC) is a very common cancer type in women and it is the second leading cause of cancer death after lung cancer [1]. It starts off when malignant cancerous cells start to grow from the breast cells [18]. Sometimes, the doctors might diagnose the patient to having a benign tumour (not cancerous) instead of malignant and hence advanced systems supporting machine learning should be used to help with the early detection [2].

Machine learning (ML) has become a widely used approach in the medical field due to the high performance in predicting results, reducing costs of drugs, resulting in patient's good health, valuing the quality of medical care being provided and in making concrete choices to save lives. It is a type of artificial intelligence that works or deals with the development of computer programs with the aid of computer models and information from different sets of data, to help in the process of classification, prediction and detection process [2]. This paper mainly focuses on the different machine learning algorithms used to classify the breast cancer as benign or malignant based upon many other factors or terms. Early diagnosis of BC can improve the prediction and chance of survival significantly and machine learning in support of evident results with good accuracies have helped achieve this [3].

Moreover, proper approach on classification helps doctors to identify benign tumours at an early stage and prevents patients from undergoing unnecessary treatments. The whole experiment is implemented using RStudio. The main objective is to distinguish between benign and malignant cancer by first, cleaning the dataset and then building a regression model for feature selection, in the end providing the best classifier that generates a model with highest accuracy. The selection of classifiers is based upon the most commonly used in data mining algorithms and research, which included, KNN, MLP, Random Forest and SVM. The "Wisconsin Breast Cancer dataset is found from the UCI Machine Learning Repository and this dataset is created by Dr. William H. Wolberg, physician at the University of Wisconsin Hospital in Madison, Wisconsin, USA".

The rest of the paper is divided as follows. Section II describes the technical background. Section III deals with the pre-processing and feature selection process. Section IV is about related work, where Section V describes the model design in detail. Section VI discusses the results, where Section VII concludes the paper with future work.

II. Technical Background

A. Support Vector Machine

A very useful machine learning algorithm is Support Vector Machine (SVM). Used for both regression and classification, it is a supervised algorithm that helps identify the hyperplane that helps differentiate between classification

data points. It is used mainly for objectives related to classification. [4]

Data points have been split using hyperplanes, as shown in Fig. 1. If a data point can be said to be a feature vector with n dimensions, then the hyperplane can be said to be the geometric shape that is seen in the occupied n-1 dimensions. The hyperplane enables us to specify the side on which the selected data point is and is classified according to that.

The objective is to determine the maximum distance between hyperplanes that define the least error of classification. Parallel hyperplanes are used to distinguish between classes when data is linearly separable. Also, the hyperplane situated in the middle is called the decision boundary while the points lying on the hyperplanes are known as support vectors.

If the support vectors are deleted, the position of the hyperplane changes. Additionally, SVM is built using these points. However, if the data cannot be separated, kernels can then be used to create non-linear classifiers. These can be produced by transforming the features [5] into a dimensional space that is higher, allowing them to be separated linearly by a hyperplane.

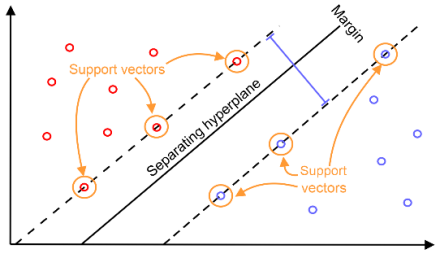

Fig. 1. Support vector machine (SVM) [6]

B. K-Nearest Neighbors

K-nearest neighbors (KNN) is a simple algorithm that keeps all available cases and classifies new ones using a similarity measure [17] [19]. It is shown in Fig. 2. A case is defined by a majority vote by its neighbors and is then assigned to the class that is most common amongst its nearest neighbors. A distance function is used to measure this. For instant, if K=1, the case gets assigned to the class of its nearest neighbor.

$$\text{Euclidean: } D(x,y)=\sqrt{\sum_{i=1}^{k}(x_i-y_i)^2} \quad (1)$$

$$\text{Manhattan: } D(x,y)=\sum_{i=1}^{k}|x_i-y_i| \quad (2)$$

$$\text{Minkowsi: } D(x,y)=\left(\sum_{i=1}^{k}(|x_i-y_i|)^q\right)^{1/q} \quad (3)$$

Distance functions (1), (2), and (3) are only valid for variables that are continuous. For the case in which variables are categorical, the Hamming distance must be used as illustrated in (4). The standardization issue is also brought into focus for variables between 0 and 1; when both numerical and categorical variables exist in the dataset.

$$D_H=\sum_{i=1}^{k}|x_i-y_i| \quad (4)$$

Another notable method of determining a suitable K value is cross-validation. It uses an independent dataset to validate the specific K value. Going by past statistics, 3-10 has been the optimal K value for most datasets. Better results are produced than 1NN [7].

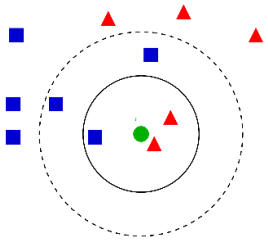

Fig. 2. K-nearest neighbor (KNN) [8]

C. Multilayer Perceptron

A good example of quintessential deep learning models is deep feedforward networks. They are also often referred to as feedforward neural networks, or multilayer perceptrons (MLPs) [9]. There can be more than one linear layer (combinations of neutrons) in the MLP. The three-layer network shown in fig. 3 is an example. The first layer in the *input layer* and the last one is the *output later*. The one in the middle is called the *hidden layer*. Data is fed into the input layer and taken from the output layer.

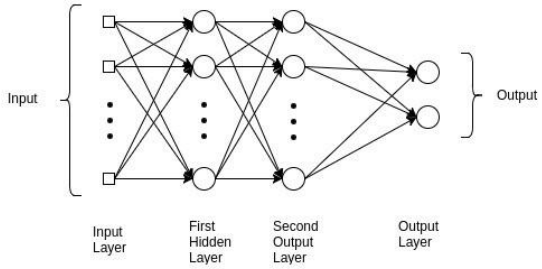

Fig. 3. 3-layer multilayer perceptron model [10]

The numbers of hidden layers can be increased in order to make the model more complex, catering to the task at hand. Every input vector has a label or ground truth that links it to its class. For each input, the output of the network returns a class score or a prediction [10].

D. Random Forest

A random forest consists of a large group of individual decision trees [20], [21], [22]. These individual decision trees operate as an ensemble and each tree outputs a class prediction. The class with the greatest number of votes becomes the prediction of the model [11]. A prediction of a tally of six 1's and three 0's is shown in Fig. 4.

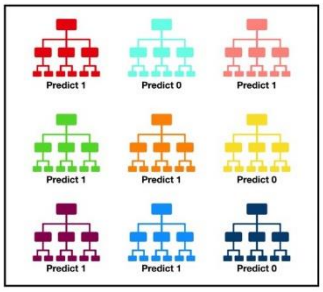

Fig. 4. Tally: six 1s and three 0s – Prediction: 1 [11]

Low correlation between models in vital. For random forest to perform well, the following points must be taken care of: a) actual signals must exist in our features in order for the models that are built to perform better than random guesses and b) The predictions and the errors that are made by individual trees must have less correlation with one another [11].

III. Dataset Prepocessing and Feature Selection

The dataset used was the "Wisconsin Breast Cancer dataset found in the UCI Machine Learning Repository created by Dr. William H. Wolberg, physician at the University of Wisconsin Hospital in Madison, Wisconsin, USA." The dataset was extracted from RStudio found in the mlbench library and the problem domain for this dataset was healthcare.

This database reflects the chronological grouping of the data as received by Dr. William H. Bolberg. The coloumns 1-10 were all independent variables with attribute type integer and values ranging 0-10. The dependent variable 'Class' was categiorical with two levels, benign and malignant. There are 16 missing attribute values and the dataset to be used had 699 instances in total. Table. 1 shows the explanation of each variable in detail.

TABLE 1. DATASET DETAILS

"Column	Variable	Description
[1]	Id	Sample code number
[2]	Cl.thickness	Clump Thickness
[3]	Cell.size	Uniformity of Cell Size
[4]	Cell.shape	Uniformity of Cell Shape
[5]	Marg.adhesion	Marginal Adhesion
[6]	Epith.c.size	Single Epithelial Cell Size
[7]	Bare.nuclei	Bare Nuclei
[8]	Bl.cromatin	Bland Chromatin
[9]	Normal.nucleoli	Normal Nucleoli
[10]	Mitoses	Mitoses
[11]	Class	Class"

A. Dataset Preprocessing

The initial step is to clean the dataset, that is to remove any missing values and outliers and thus, the first column 'ID' was removed without applying feature selection, as it didn't seem to correlate to the dependent variable 'Class'. There were 16 missing values found using the is.na( ) command in RStudio and thus the best way to deal with these were to simply remove them. Fig. 5 shows the output generated after executing the is.na( ) command indicating TRUE and FALSE values, where TRUE defined a missing or NA value. The rows containing these missing value were removed using the na.omit( ) command followed by updating the current dataset. However, an alternative approach could be getting the mean and replacing the NA values with this mean value.

	Bare.nuclei	Bl.cromatin	Normal.nucleoli	Mitoses	Class
[1,]	FALSE	FALSE	FALSE	FALSE	FALSE
[2,]	FALSE	FALSE	FALSE	FALSE	FALSE
[3,]	FALSE	FALSE	FALSE	FALSE	FALSE
[4,]	FALSE	FALSE	FALSE	FALSE	FALSE
[5,]	FALSE	FALSE	FALSE	FALSE	FALSE
[6,]	FALSE	FALSE	FALSE	FALSE	FALSE
[7,]	FALSE	FALSE	FALSE	FALSE	FALSE
[8,]	FALSE	FALSE	FALSE	FALSE	FALSE
[9,]	FALSE	FALSE	FALSE	FALSE	FALSE
[10,]	FALSE	FALSE	FALSE	FALSE	FALSE
[11,]	FALSE	FALSE	FALSE	FALSE	FALSE
[12,]	FALSE	FALSE	FALSE	FALSE	FALSE
[13,]	FALSE	FALSE	FALSE	FALSE	FALSE
[14,]	FALSE	FALSE	FALSE	FALSE	FALSE
[15,]	FALSE	FALSE	FALSE	FALSE	FALSE
[16,]	FALSE	FALSE	FALSE	FALSE	FALSE
[17,]	FALSE	FALSE	FALSE	FALSE	FALSE
[18,]	FALSE	FALSE	FALSE	FALSE	FALSE
[19,]	FALSE	FALSE	FALSE	FALSE	FALSE
[20,]	FALSE	FALSE	FALSE	FALSE	FALSE
[21,]	FALSE	FALSE	FALSE	FALSE	FALSE
[22,]	FALSE	FALSE	FALSE	FALSE	FALSE
[23,]	FALSE	FALSE	FALSE	FALSE	FALSE
[24,]	TRUE	FALSE	FALSE	FALSE	FALSE
[25,]	FALSE	FALSE	FALSE	FALSE	FALSE
[26,]	FALSE	FALSE	FALSE	FALSE	FALSE
[27,]	FALSE	FALSE	FALSE	FALSE	FALSE
[28,]	FALSE	FALSE	FALSE	FALSE	FALSE
[29,]	FALSE	FALSE	FALSE	FALSE	FALSE
[30,]	FALSE	FALSE	FALSE	FALSE	FALSE
[31,]	FALSE	FALSE	FALSE	FALSE	FALSE
[32,]	FALSE	FALSE	FALSE	FALSE	FALSE

Fig. 5. Missing values in the dataset

Identification of outliers is also an important step in the preprocessing stage. However, the dependent variable 'Class' is of type categorical and this is a binary classification problem and thus no outliers could be detected if the method was to be carried out to detect any of them. But if it was to be carried out, plotting a boxplot could be a good approach to detect any
outliers and removing them would be the best way to cope with them. A cleaned dataset was thus obtained which consisted of ten columns and reducing the total number of instances from 699 to 683 as 16 values were proven to be missing or NA values. This cleaned dataset was then divided into training and testing datasets before applying features selection.

B. Feature Selection

As discussed in the previous part, the cleaned dataset consisted of 10 columns because the input column 'ID' was removed. This dataset was divided into 80% training and 20% testing datasets using data partition. All the variables were treated properly before applying regression that is all were converted to numerical form.

A multilinear regression (MLR) model was built, and this was used to detect the most affective attributes using the olsrr command in RStudio.

On correct observation, model number 8 (according to the summary found), was proven to show the least mean square error and highest R-square which showed that all the attributes except 'Mitoses' effect the model performance and that this could be removed. Later, to confirm this observation, the feature selection step was repeated but this time using all 11 attributes (including 'Id'), and this resulted with the best model which excluded the 'Id' and 'Mitoses' attributes and included the rest inputs. Thus, it was proven that the attributes 'Id' and 'Mitosis' do not have any effect on the output or model performance. Based on this the final attributes were:

1. Cl.thickness	4. Marg.adhesion	7. Bl.cromatin
2. Cell.size	5. Epith.c.size	8. Normal.nucleoli
3. Cell.Shape	6. Bare.nuclei	9. Class

IV. Related Work

Machine learning and data mining play a very significant role in medical research and classification is the one of the most essential task in these processes. Many researches have been conducted on the classification of breast cancer and many have actually shown good results with excellent accuracy.

Prediction of benign and malignant breast cancer using data mining techniques: This study, conducted by members of the VBS Purvanchal Univeristy in Jaipur, India, compared the peformance of different classifiers that focussed on the survivability of breast cancer using data mining algorithms such as Naïve Bayes, J448 and RBF network [12].. The results showed highest accuarcy recorded by Naïve Bayes predictor with 97.36%. Comparing this study to my study, I got more accurate results using different algorithms as my highest predictor accuarcy was 99.26% generated by random forest.

Machine Learning Classification Techniques for Breast Cancer Diagnosis: This study was conducted by members of Curtine University. The main approach of this study was similar to mine which was to integrate machine learning techniques with feature selection methods and then comparing these to identify the best approach. The best approach was then used to reduce the not needed features and then subjected to support vector machine (SVM) which recorded an accuracy of 98.82% [13]. However, my study focuses more on the classification than the feature selection technique and I used support vector machine too in addition to other classifiers, but on contrary, my model generated more accurate results with additional classifiers with the highest beimg 99.26%.

Data mining Techniques: To Predict and Resolve Breast Cancer Survivability: In this paper Vikas Chaurasia and Saurarabh Pal compared the different supervised learning classifier performances using Naïve Bayes, SVM-RBP kernel, RBF neural ntworks and to find the best classifier in datasets they used Decision trees (J48) and simple CART. The results showed that SVM-RBF proved the best performance of accuracy 96.84% in Wisconsin Breast Cancer (original) datasets [14]. This accuracy compared to my results was low as mine showed 99.26%.

V. Model Design

A. Regression Analysis

Multilinear regression (MLR) is used to predict a numeric outcome from a set of numeric independent variables. MLR model was built using the 'lm' function and partitioning using the 'createDataPartition' method, from the caret library, which contained 80% training and 20% testing dataset attributes. The variable 'training' consisted of the training dataset and 'testing' consisted the testing dataset. The 'createDataPartition' function takes in parameters such as the output variable, the training dataset percentage, and if it the variable is a list of values or matrix. The 'lm' function was used to perform regression, it takes in the training dataset variables in a form of an equation and outputs the regression formula.

Fig. 6. VIF

```
> car::vif(modelLR)
training[, 1] training[, 2] training[, 3] training[, 4] training[, 5] training[, 6]
     1.931167      7.460393      6.804209      2.596452      2.726955      2.483671
training[, 7] training[, 8] training[, 9]
     2.851650      2.675164      1.434058
```

Next, the VIF (Variance Inflation Factor) values were checked to detect the multicollinearity problem. Multicollinearity problem exists if there is a correlation among the independent variables [15]. Fig. 6 shows that the VIF value of column 2 or 'Cell.size' was the highest, and anything greater than 10 can cause a problem, however, in order to get more accurate results, variables with high VIF can be removed from the model. Hence, 'Cell.size' was removed to prove this and it showed that this does not have any effect as the relative standard error (RSE) remains almost the same in both cases.

Fig. 7 shows the statistical analysis on the regression model that illustrates the relative standard error (RSE) as 0.1971, which is low to define the regression model as "good". It also shows the minimum, maximum, 1st quartile, 3rd quartile and median values. The p-values indicated that column 9 or 'Mitoses' could be removed to improve the accuracy of the results as this had the highest p-value, just as proved in the features selection step. Therefore this should be removed for better regression analysis

Fig. 7. Summary of modelLR

Next, the model was tested using the testing dataset and the results were plotted on a graph. Fig. 8 shows series of 1s and 2s that indicates the number of benign and malignant cancer types. The mean absolute error (MAE) was recorded which was 0.1145, this low error proved that the model performance was "good".

```
> summary(modelLR)

Call:
lm(formula = training[, 10] ~ training[, 1] + training[, 2] +
    training[, 3] + training[, 4] + training[, 5] + training[,
    6] + training[, 7] + training[, 8] + training[, 9])

Residuals:
     Min       1Q   Median       3Q      Max
-0.82884 -0.08377 -0.01786  0.05413  0.77452

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.762575   0.019419  39.269  < 2e-16 ***
training[, 1]  0.031823   0.004170   7.631 1.07e-13 ***
training[, 2]  0.026481   0.007367   3.595 0.000355 ***
training[, 3]  0.015378   0.007150   2.151 0.031948 *
training[, 4]  0.009060   0.004544   1.994 0.046676 *
training[, 5]  0.010933   0.006160   1.775 0.076459 .
training[, 6]  0.042153   0.003638  11.586  < 2e-16 ***
training[, 7]  0.013580   0.005797   2.342 0.019521 *
training[, 8]  0.020122   0.004440   4.532 7.21e-06 ***
training[, 9] -0.002406   0.005386  -0.447 0.655333
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1971 on 537 degrees of freedom
Multiple R-squared:  0.8341,    Adjusted R-squared:  0.8313
F-statistic:   300 on 9 and 537 DF,  p-value: < 2.2e-16
```

Fig. 8. Testing plot

B. Classification

This section shows the different classifiers used and the functions used to predict the output. The classifiers used, as discussed before, are multilayer perceptron (MLP), support vector machine (SVM), random forest, and k-nearest neighbor (KNN).

a) Multilayer Perceptron (MLP): Before any work on the MLP classification, we must convert our categorical independent variables to dummy variables. First the data was divided into training and testing datasets. Then to perform MLP classification, we must convert our variables from named lists to matrices. The model was tested, and then numerical results were converted to a result of 2 levels. Level '1.0' indicated 'Benign' and level '2.0' indicated 'Malignant'. At the end the function confusionMatrix was used from the caret library to test the results.

b) Support Vector machine (SVM), random forest, k-nearest neighbor (KNN): On contrary to MLP, the variables do not need to be converted to dummy variables and hence, classification was performed on the original input data. Later the results were predicted using the testing dataset and confusionMatrix displayed the results.

VI. Results and Discussion

This section discusses the outcomes of the four different classifier models using certain statistical evaluation parameters, which include accuracy, recall, precision and f-measure. Confusion matrices were found to calculate these parameters. Equations (4), (5), (6) and (7) show how these parameters were calculated using the confusion matrix.

$$\text{"Accuracy}= \frac{TP+TN}{TP+FP+TN+FN}\text{"} \quad (4)$$

$$\text{" Recall}= \frac{TP}{TP+FN}\text{"} \quad (5)$$

$$\text{"Precision}= \frac{TP}{TP+FP}\text{"} \quad (6)$$

$$\text{"F-measure}= \frac{2*recall*precision}{recall+precision}\text{"} \quad (7)$$

The positive class for all the classification models was found to be 'benign'. TP is defined as the number of positive or benign breast cancer cases while FP are negative or malignant cases. TN represents correctly classified as not benign (malignant) and FN represents benign cases but wrongly classified as malignant.

Accuracy stands for the total number of correct classifications, recall represents the total number of positive or benign cases that were correctly classified, and precision signifies how accurate the positive classification was, irrespective of the wrongly classified situation and f-measure is the harmonic mean of precision and recall [16].

Classification of the breast cancer cases as benign or malignant is vital especially during the early stages when the cancer type can be identified that has caused minimal damage and the survival rate is high. Wrongly classifying a cancer as malignant (cancerous) can simply waste the doctors and the patients time and money along with the treatment procedures and medications being supplied, while wrongly classifying it as benign (not cancerous) can be a matter of life and death.

Table. 2 summarizes the accuracy, recall, precision and f-measure of the four different classifiers. This proves that Random forest has the highest recall, precision and f-measure, thus the highest accuracy of 99.26%. This explains that random forest can be the best classifier to diagnose a breast cancer as benign or malignant.

VII. Conclusion and Future Work

This paper analysed the Wisconsin Breast Cancer Dataset using four different algorithms to classify if a tumour is malignant or benign. The experimental work proves that Random forest is best classifier for this dataset features and attributes, as it showed excellent results. It obtained an accuracy of 99.26%, precision of 98.99%, recall of 100% and f-measure was 99.44%. This paper also proves that the features selection method also helped improve the diagnosis of benign and malignant tumours. Although the results of most classifiers were quite close to each other but considering the most accurate classifier is extremely important in the field of diagnosis of diseases, in order to get excellent results.

Future work can be focussed upon implementing the chosen approach into clinical trials or any form of practical testing methods which can be used by doctors to study the details of diagnosing breast cancer. Moreover, additional approaches on building a better regression model and using other classifiers and features selection techniques can also be

considered to compare the results and choose the best approach. This can be tested on other similar diseases.

REFERENCES

[1] A. Felman, "What to know about breast cancer", *Medicalnewstoday.com*, 2019.[Online].Available:https://www.medicalnewstoday.com/articles/37136. [Accessed: 20- Apr- 2020].

[2] D. A. Omondiagbe, S. Veeramani and A. S. Sidhu, "Machine Learning Classification Techniques for Breast Cancer Diagnosis", in *11th Curtin University Technology, Science and Engineering (CUTSE) International Conference*, Sarawak, Malaysia, 2019, pp. 1-2.

[3] H. You and G. Rumbe, "Comparative Study of Classification Techniques on Breast Cancer FNA Biopsy Data", *International Journal of Artificial Intelligence and Interactive Multimedia*, vol. 1, no. 3, p. 6, 2010. [Accessed 21 April 2020].

[4] R. Gandhi, "Support Vector Machine — Introduction to Machine Learning Algorithms", *Medium*. [Online]. Available: https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47. [Accessed: 22- Apr- 2020].

[5] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the fifth annual workshop on Computational learning theory - COLT '92*, 1992, pp. 144–152.

[6] Packt. *support vector machine* [Image]. Retrieved 4 May 2020, from https://static.packt-cdn.com/products/9781789345070/graphics/6a831600-9a0d-429f-9d34-d957c45b9517.png.

[7] *KNN Classification*. Saedsayad.com. Retrieved 5 May 2020, from https://www.saedsayad.com/k_nearest_neighbors.htm.

[8] Srivastava, T. (2018). *K-nearest neighbor* [Image]. Retrieved 5 May 2020, from https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/.

[9] Karim, M. (2016). *Deep Learning via Multilayer Perceptron Classifier - DZone Big Data*. dzone.com. Retrieved 5 May 2020, from https://dzone.com/articles/deep-learning-via-multilayer-perceptron-classifier.

[10] Kumar Kain, N. (2018). *Understanding of Multilayer perceptron (MLP)*. Medium. Retrieved 5 May 2020, from https://medium.com/@AI_with_Kain/understanding-of-multilayer-perceptron-mlp-8f179c4a135f.

[11] Yiu, T. (2019). *Understanding Random Forest*. Towards data science. Retrieved 5 May 2020, from https://towardsdatascience.com/understanding-random-forest-58381e0602d2.

[12] Chaurasia, V., Pal, S., & Tiwari, B. (2018). Prediction of benign and malignant breast cancer using data mining techniques. *Journal Of Algorithms & Computational Technology*, *12*(2), 119-126. https://doi.org/10.1177/1748301818756225

[13] Chaurasia, V., Pal, S., & Tiwari, B. (2018). Prediction of benign and malignant breast cancer using data mining techniques. *Journal Of Algorithms & Computational Technology*, *12*(2), 119-126. https://doi.org/10.1177/1748301818756225

[14] V. Chaurasia and S. Pal, "Data Mining Techniques : To Predict and Resolve Breast Cancer Survivability," vol. 3, no. 1, pp. 10–22, 2014.

[15] *VIF – Lean Manufacturing and Six Sigma Definitions*. Leansixsigmadefinition.com. Retrieved 5 May 2020, from http://www.leansixsigmadefinition.com/glossary/vif/.

[16] Nighania, K. (2018). *Various ways to evaluate a machine learning models performance*. Medium. Retrieved 6 May 2020, from https://towardsdatascience.com/various-ways-to-evaluate-a-machine-learning-models-performance-230449055f15.

[17] C. López-Martín, Y. Villuendas-Rey, M. Azzeh, A. Bou Nassif, and S. Banitaan, "Transformed k-nearest neighborhood output distance minimization for predicting the defect density of software projects," J. Syst. Softw., vol. 167, p. 110592, Sep. 2020.

[18] R. Hamoudi, M. Bettayeb, A. Alsaafin, M. Hachim, Q. Nassir, and A. B. Nassif, "Identifying Patterns of Breast Cancer Genetic Signatures using Unsupervised Machine Learning," in 2019 IEEE International Conference on Imaging Systems and Techniques (IST), 2019, pp. 1–6.

[19] C. López-Martín, Y. Villuendas-Rey, M. Azzeh, A. Bou Nassif, and S. Banitaan, "Transformed k-nearest neighborhood output distance minimization for predicting the defect density of software projects," J. Syst. Softw., vol. 167, p. 110592, Sep. 2020.

[20] A. B. Nassif, "Short term power demand prediction using stochastic gradient boosting," in International Conference on Electronic Devices, Systems, and Applications, 2017.

[21] A. B. Nassif, M. Azzeh, L. F. Capretz, and D. Ho, "A comparison between decision trees and decision tree forest models for software development effort estimation," in 2013 3rd International Conference on Communications and Information Technology, ICCIT 2013, 2013, pp. 220–224.

[22] A. B. Nassif, "Software Size and Effort Estimation from Use Case Diagrams Using Regression and Soft Computing Models," University of Western Ontario, 2012.

TABLE. 2. SUMMARY OF CLASSIFIER RESULTS

Classifier Models	Accuracy (%)	Precision (%)	Recall (%)	F-measure (%)
Muiltilayer perceptron (MLP)	94.07 %	96.51 %	94.32%	95.40%
Support vector machine (SVM)	97.78%	98.85%	97.73%	98.29%
K-nearest neigbors (KNN)	97.04%	98.83%	96.59%	97.70%
Random forest	99.26%	98.88%	100%	99.44%

Authors Contributions: Hajra Iqbal wrote the paper. Ali Bou Nassif and Ismail Shahin revised the model design, as well as the whole paper