Predictive Modelling of Benign and Malignant Tumors Using Binary Logistic, Support Vector Machine and Extreme Gradient Boosting Models

Article · November 2019

DOI: 10.12691/ajams-7-6-2

3 authors, including:

Moses Muraya
Leibniz Institute of Plant Genetics and Crop Plant Research
62 PUBLICATIONS **633** CITATIONS

SEE PROFILE

American Journal of Applied Mathematics and Statistics, 2019, Vol. 7, No. 6, 196-204
Available online at http://pubs.sciepub.com/ajams/7/6/2
Published by Science and Education Publishing
DOI:10.12691/ajams-7-6-2

Predictive Modelling of Benign and Malignant Tumors Using Binary Logistic, Support Vector Machine and Extreme Gradient Boosting Models

Peter Gachoki[1,*], Moses Mburu[2], Moses Muraya[3]

[1]Department of Physical Sciences, Chuka University, P.O Box 109-60400, Chuka, Kenya
[2]KEMRI-Wellcome Trust Kilifi, P.O Box 230-80108, Kilifi, Kenya
[3]Department of Plant Sciences, Chuka University, P.O Box 109-60400, Chuka, Kenya
*Corresponding author: Peter Gachoki, pkgachoki@gmail

Received October 14, 2019; Revised November 18, 2019; Accepted November 26, 2019

Abstract Breast cancer is the leading type of cancer among women worldwide, with about 2 million new cases and 627,000 deaths every year. The breast tumors can be malignant or benign. Medical screening can be used to detect the type of a diagnosed tumor. Alternatively, predictive modelling can also be used to predict whether a tumor is malignant or benign. However, the accuracy of the prediction algorithms is important since any incidence of false negatives may have dire consequence since a person cannot be put under medication, which can lead to death. Moreover, cases of false positives may subject an individual to unnecessary stress and medication. Therefore, this study sought to develop and validate a new predictive model based on binary logistic, support vector machine and extreme gradient boosting models in order to improve the prediction accuracy of the cancer tumors. This study used the Breast Cancer Wilcosin data set available on Kaggle. The dependent variable was whether a tumor is malignant or benign. The regressors were the tumor features such as radius, texture, area, perimeter, smoothness, compactness, concavity, concave points, symmetry and fractional dimension of the tumor. Data analysis was done using the R-statistical software and it involved, generation of descriptive statistics, data reduction, feature selection and model fitting. Before model fitting was done, the reduced data was split into the train set and the validation set. The results showed that the binary logistic, support vector machine and extreme gradient boosting models had predictive accuracies of 96.97%, 98.01% and 97.73%. This showed an improvement compared to already existing models. The results of this study showed that support vector machine and extreme gradient boosting have better prediction power for cancer tumors compared to binary logistic. This study recommends the use of support vector machine and extreme gradient boosting in cancer tumor prediction and also recommends further investigations for other algorithms that can improve prediction.

Keywords: *benign, malignant, binary logistic, support vector machine, extreme gradient boosting*

Cite This Article: Peter Gachoki, Moses Mburu, and Moses Muraya, "Predictive Modelling of Benign and Malignant Tumors Using Binary Logistic, Support Vector Machine and Extreme Gradient Boosting Models." *American Journal of Applied Mathematics and Statistics*, vol. 7, no. 6 (2019): 196-204. doi: 10.12691/ajams-7-6-2.

1. Introduction

Cases of breast cancer diagnosis and mortality have grown over the years across the world. This has been attributed to changes in lifestyles as well as hormonal changes [1]. Mortalities from breast cancer have been attributed to late diagnosis as well as challenges in access to treatment. Studies have proposed increased campaign on self-examination that will facilitate early diagnosis which willing turn help in control and treatment of breast cancer [8]. Breast cancer is caused by the buildup of extra cells of the on the breast causing a mass tissue that is usually referred to as a lump or a tumor. A tumor on the breast can either be malignant or begnin [3]. A benign tumor is not cancerous while a malignant tumor is cancerous. Benign tumors are harmless and they do not cause an invasion to tissues next to them and neither do they spread to other body parts. When a benign tumor is removed, it does not grow back again. On the other hand malignant tumors are dangerous and they can affect the tissues next to them. These tumors can also spread to other body parts and even when they are removed, there is always a possibility that they will grow back [5].

Researchers have also tried to highlight some breast cancer risk factors. These include; gender, age, history, genes, radiations, ethnicity, overweight, breast feeding, alcoholism, nature of breasts, smoking, low levels of vitamin D, chemical exposure among others [2]. Prediction of whether a tumor is benign or malignant can be an important step in breast cancer control. This is because if a tumor is predicted to be malignant, early medication can be sought and thus the cancer can be

controlled before it gets to an advanced stage. However, the prediction accuracies and are very important since cases of false negatives may have dire consequences since somebody cannot be put under medication and this can lead to deaths. Cases of false positives can subject to unnecessary stress and medication. therefore, it is important to develop an algorithm that can predict if a tumor is begnin or malignant with the best accuracy as possible.

Several studies have come up with prediction algorithms for breast cancer, and have attained different prediction accuracies. For example a study that investigated whether breast cancer was caused by modifiable or non-modifiable factors using the Rep Tree, RBF Network and using simple logistic attained a classification accuracy rate of 74.5% [4]. The non-modifiable factors considered were age, menustrial history, gender, age at menopause, age at menarche and number of first degree relatives who have suffered from breast cancer. The modifiable factors were BMI, number of children, alcoholism, diet, age at first birth and number of abortions. Prediction of whether a tumor was benign or malignant using Naïve Bayes, SVM-RBF kernel, decision tree, neural networks and regression tree produced the SVM-RBF kernel with an accuracy of 96.84% [5], when prediction of benign and malignant breast cancer was done using data mining techniques, Naïve Bayes attained the best prediction accuracy of 97.73%. Application of Decision Trees, Naive- Bayesian methods, Sequential Minimal Optimization to detect breast cancer tumors, Sequential Minimal Optimization showed high level performance compare with other classifiers [4].

It is clear that prediction accuracies of the prediction algorithms vary according to the algorithm used. Further, these accuracies could also vary depending on the pre analysis performed on the data. Such pre analysis includes the data imputations and the data dimension reductions if the variables are correlated. This study therefore sought to investigate if the binary logistic, support vector machine and extreme gradient boosting improved the prediction accuracies of the begnin and malignant tumors. The algorithms were developed after performing data imputation and data dimension reduction using the principal component analysis.

2. Methodology

This study used the breast cancer data available on Kaggle. The dependent variable in the data is whether a tumor is benign or malignant. The predictor variables are features of tumor that includes; radius, texture, area, perimeter, smoothness, compactness, concavity, concave points, symmetry and fractional dimension of the tumor. The analysis of the data involved generation of descriptive statistics, performing data reduction and feature selection using the principal component analysis and then fitting the binary logistic, support vector machine and extreme gradient boosting models. The data was split into two where 70% formed the training data set and 30% formed the testing set.

2.1. Principal Component Analysis

This is a procedure that makes use of the orthogonal transformation to convert correlated variables into linearly uncorrelated variables that are referred to as principal components [7]. The first principal component has the largest possible variance and every subsequent component has the largest possible variance under the constraint that is orthogonal to the subsequent components. The end result is a vector of uncorrelated orthogonal basis set.

Mathematically, the transformation;

$$t_{k(i)} = x_{(i)}, for\ all\ i = 1,2,3,\ldots.,n$$

is a set of vectors of coefficients $w_{(k)} = (w_1, \ldots.., w_p)_{(k)}$ that map every row vector $x_{(i)}$ of x to a new vector scores of principal components $t_{(i)} = (t_1, t_2, \ldots, t_l)_{(i)}$.

This is done in such a way that the $t_1, t_2, \ldots, t_l$ of t considered over the data set inherit maximum possible variance from x. w is constrained to be a unit vector and the choice of l is selected in such a way that it is less than p so as to reduce dimensionality.

For maximum variance, the weight $w_{(1)}$ must satisfy;

$$w_{(1)} = \arg\max_{w=1} \left\{ \sum_i (t_1)_i^2 \right\} = \arg\max_{w=1} \left\{ \sum_i (x_{(i)}.w)^2 \right\}.$$

Once $w_{(1)}$ is obtained, the first principal component is given as;

$$t_{1(i)} = x_{(i)} \cdot w.$$

The k^{th} principal component can be obtained through subtracting the $k-1$ principal components from x.

$$\hat{x}_k = X - \sum_{s=1}^{k-1} x w_{(s)} w_{(s)}^T.$$

The principal component analysis also helps in data dimension reduction while still retaining much of the variance in the data sets.

2.2. Logistic Regression

It is a generalized linear model that fits data that has a binary outcome. If the data has multiple classes, the logistic regression generalizes into a multinomial regression model (Sperandei, 2014). The logistic regression equation is;

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 X_1 + \ldots + \beta_n X_n.$$

Using the maximum likelihood estimation, the cost function can be derived as;

$$\tau(\theta) = \frac{-1}{m} \sum_{i=1}^{m} \left\{ y^i \log(h_\theta\left(x^i\right) + (1-y^i) \log(1-h_\theta\left(x^i\right) \right\}.$$

The $\beta^{'}s$ are obtained by minimizing the cost function.

To solve the problem of over fitting when using a logistic regression, a regularized logistic regression is used. This is achieved by adding a regularization term to the cost function. The $L1$ regularization is achieved by adding a penalty that is equivalent to the sum of absolute values of the coefficients. That is;

$$\tau(\theta) = \frac{-1}{m}\sum_{i=1}^{m}\left\{y^i \log(h_\theta(x^i)) + (1-y^i)\log(1-h_\theta(x^i))\right\} + \frac{\lambda}{2m}\sum_{j=1}^{n}\left|\theta^i\right|.$$

To optimize λ cross-validation is used and the λ that yields the best cross-validation accuracy is chosen.

2.3. Support Vector Machine

This is an algorithm under supervised machine learning that is used for classification and regression. However, most of the times, it is used for classification [9]. To understand the working of support vector machine model, an example of a data is considered that has two classes that can be separated using a straight line which can also be referred to as the decision boundary or hyperplane (Figure 1).

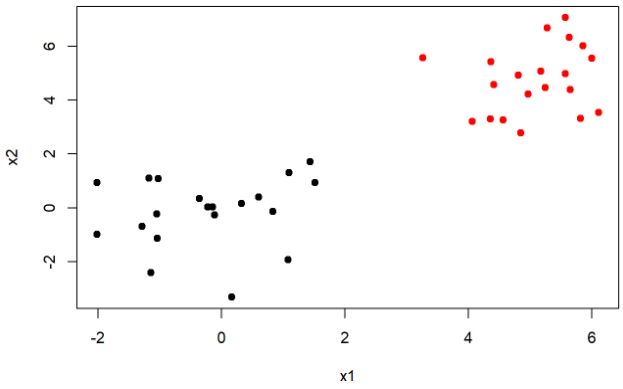

Figure 1. Data set with two classes that are separable using a straight line

The point under consideration is which is the best line that can separate the two classes since there are multiple lines that can do the separation. This consideration leads to the concept of maximum margin classification. This means that the support vector machine finds the hyperplane that yields the largest margin between the two classes.

Choosing the solid line as the hyperplane and margins as the dotted lines, the points (circled) that lie on the margin are referred to as support vectors (Figure 2).

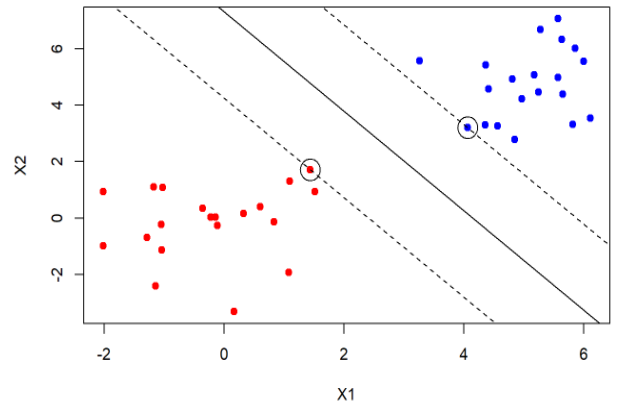

Figure 2. Diagrammatic representation of the support vectors

The support vectors are the ones that are used by the support vector machine to obtain a decision boundary. The others points are not used. Since for this case the space is two dimensional, the equation for the separating line is;

$$\beta_0 + \beta_1 + \beta_2 X_2$$

When the equation evaluates to more than 0, then 1 is predicted. That is;

$$\beta_0 + \beta_1 + \beta_2 X_2 > 0, y = 1$$

When the equation evaluates to less than 0, then -1 class is predicted. That is;

$$\beta_0 + \beta_1 + \beta_2 X_2 < 0, y = -1.$$

This yields a maximization problem;

$$width\ of\ the\ margin = M$$

$$\sum_{j=1}^{n}\beta_j = 1$$

$$y_i\ (\beta_0 + \beta_1 + \beta_2 X_2)\ \geq M.$$

In most cases, the classes are noisy. Considering a case where no matter the line chosen, some points will always be on the wrong side of the decision boundary, the maximum margin classification would not work (Figure 3).

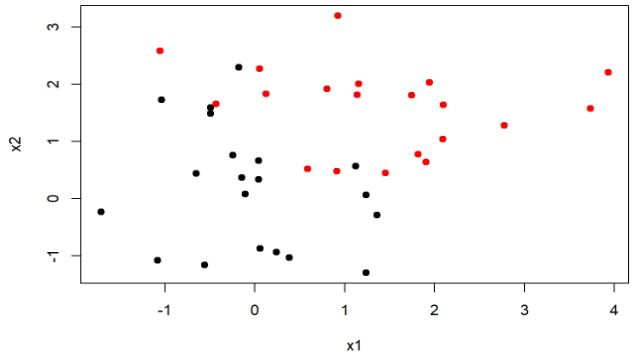

Figure 3. Diagrammatic representation of data with noisy classes

In such cases, support vector machine introduces a soft margin that allows some points to be on the wrong side. By introduction of the error term, some slack is allowed. An example of two case maximization yields;

$$y_i\ (\beta_0 + \beta_1 + \beta_2 X_2)\ \geq M\ (1\text{- }\varepsilon)$$

$$\sum_{i=0}^{n}\varepsilon_i\ \leq C$$

Where C is a tuning parameter that determines the width of the margin while $\varepsilon^{'}s$ are the slack variables that allow observations to fall on the wrong side of the margin (Figure 4).

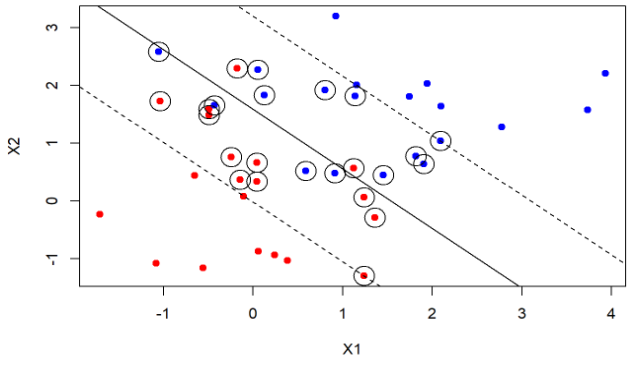

Figure 4. Support vector machine model classification for data with noisy classes.

If the decision boundary is non-linear, support vector machine introduces kernels.

2.4. XGBoost

XGBoost is an ensemble learner where the multiple machine learning algorithms are used at the same time for prediction [6]. An example of an ensemble learner is the random forest that uses many decision trees for prediction. Ensemble learners are classified into Bagging and Boosting. The random forest is a bagging learner where decision trees are developed from the subsets of the training data set and the final prediction is a weighted sum of all the decision tree functions. In boosting learners, samples are selected sequentially. For instance, the first sample is selected and a decision tree is fitted. The model then picks the examples that were hard to learn and using them and a few others selected at random from the training data set, a second model is fitted. Prediction is then made using the first and the second models. The model is then evaluated and hard examples are picked together with other randomly selected examples from training set and another model is fitted. The process of boosting algorithms continues up to a number n.

In gradient boosting, the first model is fitted using the original training set. For example, a simple regression model, $y = f(x) + \varepsilon$. If the error, say, it is too large, one might try to, say, add more features, use another algorithm, tune the algorithm, look for more training set etc. However, if the error is not white noise and it has a relationship with the output, then a second model can be fitted $\epsilon = f(x) + \varepsilon_1$. The process continues n – times and the final model will be;

$$\epsilon_n = f_n(x) + \varepsilon_{n-1}.$$

The final step involves adding these models together with some weighting criteria;

Weights = $\alpha's$ which yields the final function that is used for prediction.

2.5. Model Comparison Criterions

Below is a presentation of the criterions that were used for model comparison;

$$Accuracy = \frac{Truepositives + Truenegatives}{N}.$$

Precision: this is a measure of the proportion of patients who were predicted to have a malignant tumor and actually had it.

$$precision = \frac{True\ positives}{Predicted\ positives}.$$

Recall (sensitivity): this is a measure of the proportion of the patients that had malignant tumor and were detected by the predicting algorithm. This is referred to as the true positive rate.

$$sensitivity = \frac{True\ positives}{Actual\ positives}.$$

Specificity is the true negative rate. This is the proportion the patients who had benign tumors and were detected by the predicting algorithm

$$Specificity = \frac{True\ negatives}{Actual\ negatives}.$$

3. Results and Discussion

3.1. Descriptive Statistics

The percentage of women with malignant tumors was 37.26% while the rest 62.74% had benign tumors. These percentages presented 212 out of 569 for malignant tumors and 357 out of 569 for begnin tumors (Figure 5).

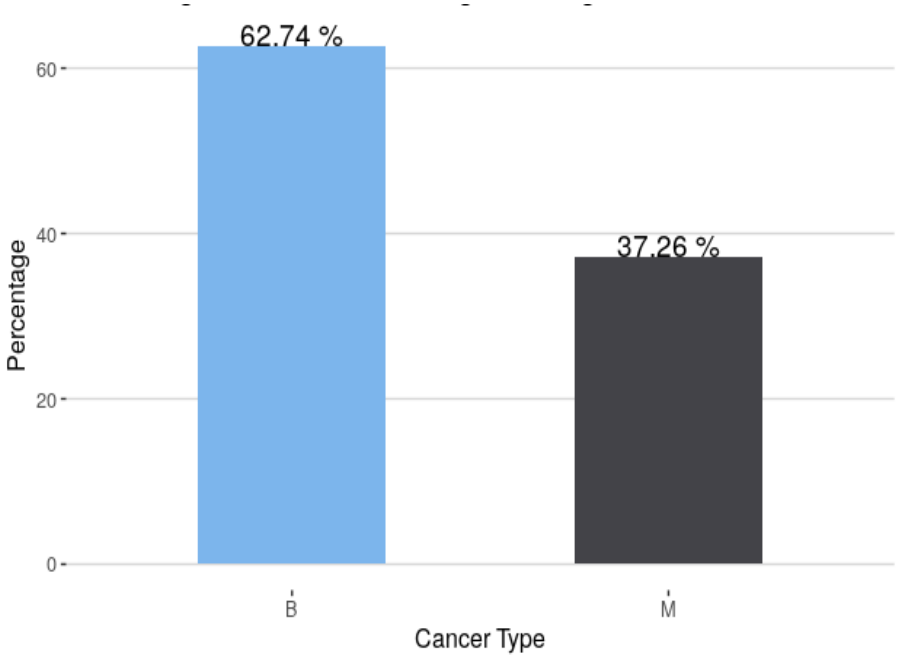

Figure 5. Percentages of women with benign and malignant tumors (B – Begnin tumor, M – Malignant tumor)

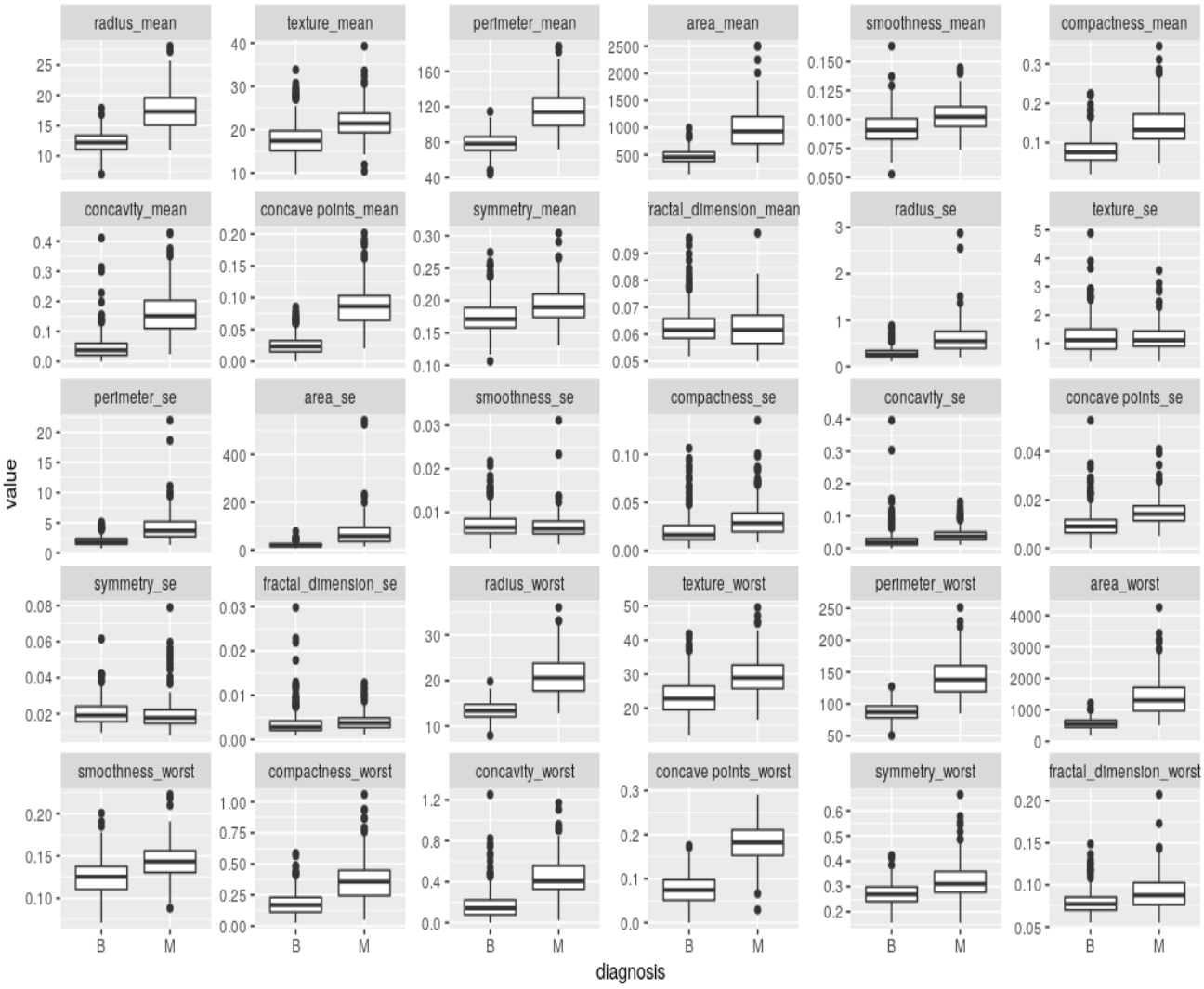

Figure 6. Box plots for variables where there is a significant difference between the tumor types

Table 1. Extract of a matrix showing correlation between variables

texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	symmetry_mean
Texture_Mean	1.00000	0.329	0.3210857	-0.0233885	0.236702	0.3024	0.293464
Perimeter_Mean	0.32953	1.000	0.9865068	0.2072782	0.556936	0.7161	0.850977
Area_mean	0.32108	0.986	1.0000000	0.1770284	0.498501	0.6859	0.823268
Smoothness_mean	-0.02338	0.207	0.1770284	1.0000000	0.659123	0.5219	0.553695
Compactness_mean	0.23670	0.556	0.4985017	0.6591232	1.000000	0.8831	0.8311350
Concavity_Mean	0.30241	0.716	0.6859828	0.5219838	0.883120	1.0000	0.921391
Concave points_mean	0.29346	0.850	0.8232689	0.5536952	0.831135	0.9213	1.000000
Symmetry_mean	0.07140	0.183	0.1512931	0.5577748	0.602641	0.5006	0.462497

From the boxplots, variables where there is a significant difference between the two groups of cancer tumors can be identified. When using a boxplot, if two distributions do not overlap or more than 75% of two boxplot do not overlap then a significant difference in the mean/median between the two groups is expected. Some of the variables where the distributions of two cancer tumors are significantly different are radius mean, texture mean among others. The visible differences between malignant tumors and benign tumors can be seen in means of all cells and worst means where worst means is the average of all the worst cells. The distributions of malignant tumors have higher scores than the benign tumors in these cases (Figure 6).

Some of the variables were highly correlated. Principal component analysis was used for data dimension reduction. Since variables were correlated it was evident that smaller set of features could be used in building of the models. The correlated variables were shown using a correlation matrix. An extract of the correlation matrix is presented in Table 1.

Using the elbow rule, the first 15 principle components were used. Using 15 principle components, almost 100% of the variance from the original data set was achieved. Since principal component analysis formed new characteristics, the variance explained plot was used to show the amount of variation of the original features captured by each principle component. The new features

were simply linear combinations of the old features. This plot is referred to as a scree plot. The Scree plot showed the variance explained by each principle component which reduced as the number of principle components increased (Figure 7).

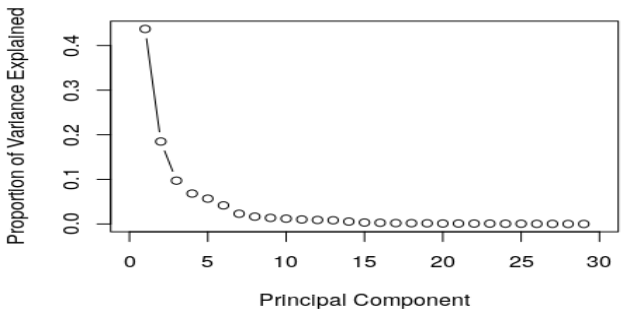

Figure 7. Scree plot for the variance explained by each principal component

The cumulative of variance plot helped to choose the number of features based on the amount of variation from original data set that the researcher wanted captured. In this case, the researcher wanted to use number of principle components that would capture almost 100% of the variation. After trying with different number of principle components, it was found out that the accuracy of the models did not increase after the 15th principle components (Figure 8).

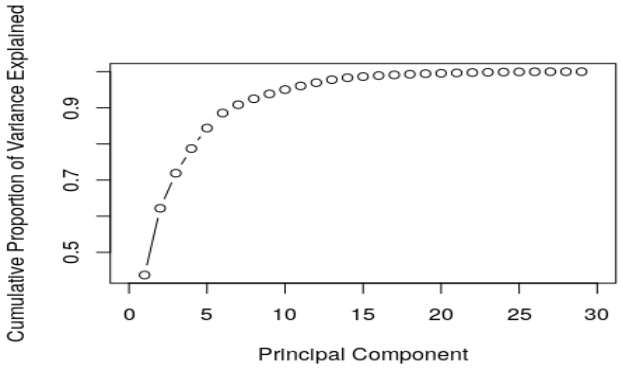

Figure 8. Cumulative variance explained by the principal components

Using the first 15 principle components as the new predictors, the data was randomly split data into training and test set in proportions of 70% and 30% respectively. The training data set was used to generate a regularized logistic regression model. The optimal values of λ were chosen using cross validation. The chosen value was the one with the highest cross-validation accuracy (Figure 9).

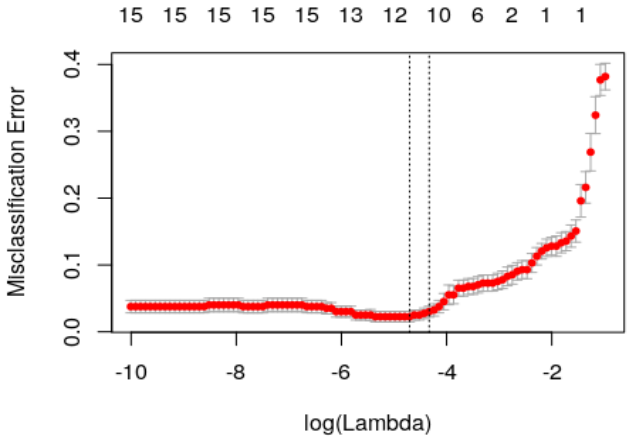

Figure 9. Misclassification errors for the log λ

The performance of the logistic model fitted was summarized using the confusion matrix and the classification tables (Table 2 & Table 3). The fitted logistic model had a classification accuracy of 96.97%. The model had a recall value of 100% implying that all the persons who had a malignant tumor were detected by the predicting algorithm. The specificity value of the algorithm was 96.33% which implied that 96.33% of the patients who had a benign tumor were detected by the algorithm. The precision of the model was 93.93%. This implied that 93.93% of the individuals who were predicted to have malignant tumor actually had it.

Table 2. Confusion matrix for the logistic regression model

		Actual	
		Benign	Malignant
Predicted	Benign	104	0
	Malignant	4	62

Table 3. Classification table for the logistic regression model

Term	Class	estimate	conf.low	conf.high	p.value
Accuracy	NA	0.9766082	0.9411932	0.9935906	0.0000000
Kappa	NA	0.9500876	NA	NA	0.1336144
Sensitivity	1	1.0000000	NA	NA	NA
Specificity	1	0.9633028	NA	NA	NA
Pos_pred_value	1	0.9393939	NA	NA	NA
Neg_pred_value	1	1.0000000	NA	NA	NA
Precision	1	0.9393939	NA	NA	NA
Recall	1	1.0000000	NA	NA	NA
f1	1	0.9687500	NA	NA	NA
Prevalence	1	0.3625731	NA	NA	NA
Detection_rate	1	0.3625731	NA	NA	NA
Detection_prevalence	1	0.3859649	NA	NA	NA
Balanced_accuracy	1	0.9816514	NA	NA	NA

Table 4. Confusion matrix for the Support Vector Machine model

		Actual	
		Benign	Malignant
Predicted	Benign	104	1
	Malignant	2	64

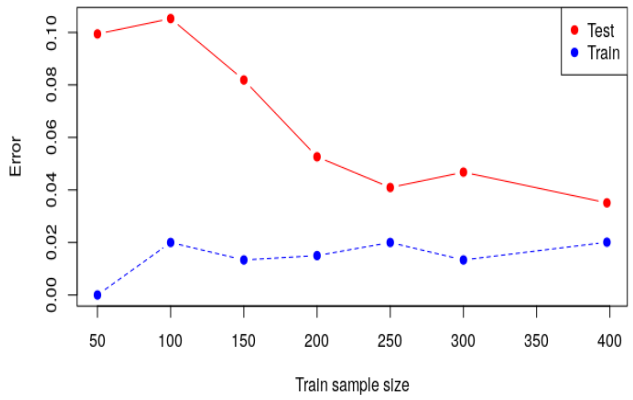

Figure 10. Training and test errors for the support vector machine model

3.2. Support Vector Machine Model

The performance of the fitted support vector machine model was summarized using a confusion matrix (Table 4). The fitted model had a classification accuracy of 98.01%. The best SVM model was attained after a sample of 400 data values (Figure 10). The model classified two persons having malignant tumors as having benign tumors and one person having a begnin tumor as having a malignant tumor.

3.3. Extreme Gradient Boosting Algorithm

When fitting the XGBoost model, increasing cut of increases the precision (Table 5). A greater fraction of those who will be predicted that they have cancer will turn out that they have, but the algorithm is likely to have lower recall. There is therefore need to avoid too many cases of people with cancer being predicted that they do not have cancer. It will be very bad to tell someone that they do not have cancer but they have. Lowering the probability to, say, to 0.3 then this it is make sure that even if there is a 30% chance that someone has cancer then they should be flagged.

The performance of the fitted XGBoost model was summarized using the confusion matrix and a classification table (Table 6 & Table 7). The fitted XGBoost model had an overall classification accuracy of 97.73%. The model had a recall value of 100% implying that all the persons who had a malignant tumor were detected by the predicting algorithm. The specificity value of the algorithm was 97.22% which implied that 97.22% of the patients who had a benign tumor were detected by the algorithm. The precision of the model was 95.45%. This implied that 95.45% of the individuals who were predicted to have malignant tumor actually had it.

Table 5. Precision changes with increasing cut off for the XGBoost Model

Iteration	Train_error_mean	Train_error_std	Test_error_mean	Test_error_std
1	0.0854217	0.0246005	0.1005200	0.0341107
2	0.0603117	0.0032692	0.0930573	0.0364313
3	0.0552993	0.0079092	0.0930573	0.0364313
4	0.0376987	0.0031983	0.0779060	0.0279312
5	0.0339107	0.0029770	0.0754183	0.0284401
6	0.0288883	0.0063645	0.0704053	0.0318404
7	0.0226133	0.0061394	0.0704053	0.0200412
8	0.0226133	0.0061394	0.0704053	0.0200412
9	0.0213317	0.0076876	0.0603800	0.0270910
10	0.0175727	0.0063618	0.0654303	0.0236148
11	0.0138037	0.0046634	0.0629053	0.0251602
12	0.0100443	0.0035311	0.0653930	0.0293481
13	0.0087910	0.0017587	0.0629053	0.0251602
14	0.0075377	0.0030697	0.0629053	0.0251602
15	0.0075377	0.0030697	0.0653930	0.0235629
16	0.0075377	0.0030697	0.0628677	0.0258586
17	0.0075377	0.0030697	0.0603423	0.0284188
18	0.0075377	0.0030697	0.0628677	0.0318281
19	0.0062750	0.0035377	0.0629053	0.0306453
20	0.0050220	0.0017657	0.0653930	0.0293481
21	0.0025157	0.0017789	0.0653930	0.0293481
22	0.0025157	0.0017789	0.0653930	0.0293481
23	0.0025157	0.0017789	0.0653930	0.0293481
24	0.0012627	0.0017857	0.0628677	0.0318281
25	0.0012627	0.0017857	0.0628677	0.0318281
26	0.0000000	0.0000000	0.0629053	0.0306453
27	0.0000000	0.0000000	0.0603800	0.0328373
28	0.0000000	0.0000000	0.0603800	0.0328373
29	0.0000000	0.0000000	0.0603800	0.0328373
30	0.0000000	0.0000000	0.0603800	0.0328373

Table 6. Classification table for the XGBoost model

Term	class	estimate	conf.low	conf.high	p.value
Accuracy	NA	0.9824561	0.9495877	0.9963673	0.0000000
Kappa	NA	0.9626719	NA	NA	0.2482131
Sensitivity	1	1.0000000	NA	NA	NA
Specificity	1	0.9722222	NA	NA	NA
pos_pred_value	1	0.9545455	NA	NA	NA
neg_pred_value	1	1.0000000	NA	NA	NA
Precision	1	0.9545455	NA	NA	NA
Recall	1	1.0000000	NA	NA	NA
f1	1	0.9767442	NA	NA	NA
Prevalence	1	0.3684211	NA	NA	NA
detection_rate	1	0.3684211	NA	NA	NA
detection_prevalence	1	0.3859649	NA	NA	NA
balanced_accuracy	1	0.9861111	NA	NA	NA

Table 7. Confusion matrix for the XGBoost model

		Actual	
		Benign	Malignant
Predicted	Benign	105	0
	Malignant	3	63

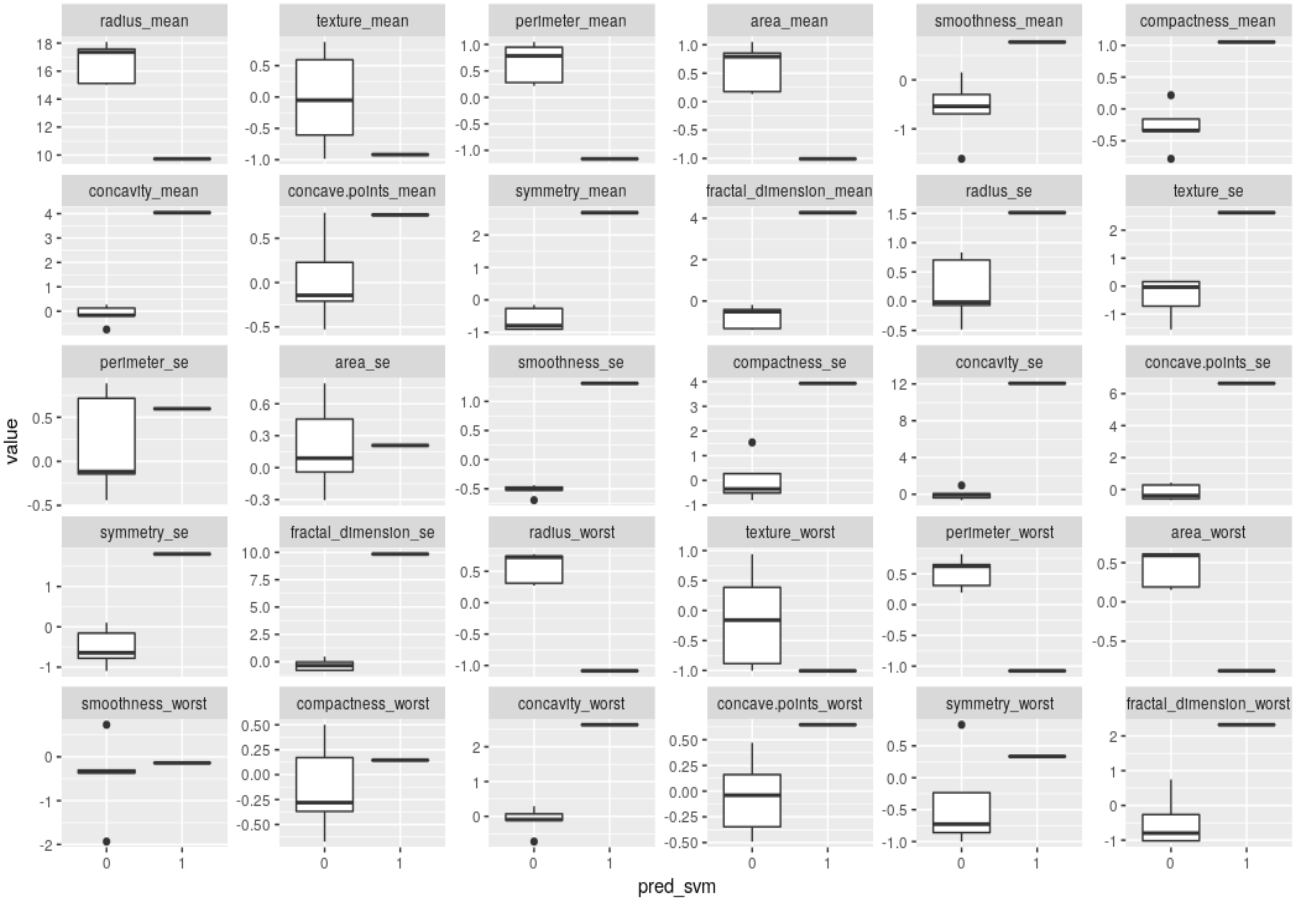

Figure 11. Error analysis for the fitted support vector machine model

3.4. Analysis of Errors of an Algorithm

Error analysis involves evaluating the examples that the algorithm misclassified to find out if there is a trend. In general terms, this is trying to find out the weak points of a predicting algorithm and also finding out why the algorithm was making those errors. For instance, from the boxplots below the malignant tumors that were misclassified had lower radius mean compared to mislassified benign tumors. This contrary to what we saw in the first boxplots graph (Figure 11).

4. Conclusion

In conclusion, the support vector machine and extreme gradient boosting models perform better in classification and prediction of breast cancer tumors as compared to the

binary logistic model. However, support vector machine shows better prediction power when compared with the extreme gradient boosting model. This performance is better compared to the performance of the already existing models. The precision of extreme gradient boosting model also increases with increased cut off point. From this study, it can therefore be recommended that support vector machine and extreme gradient boosting model can be used in predicting the breast cancer tumor types. In addition, there should be continued effort of evaluating if there are other algorithms that can yield better classification accuracy than the ones considered for this study.

References

[1] Akram, M., Iqbal, M., Daniyal, M., & Khan, A. U. (2017). Awareness and current knowledge of breast cancer. *Biological research*, *50*(1), 33.

[2] American Cancer Society (2018). Breast Cancer Facts and Figures 2017-2018. https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/breast-cancer-facts-and-figures/breast-cancer-facts-and-figures-2017-2018.pdf.

[3] Chaurasia, V., & Pal, S. (2014). Performance analysis of data mining algorithms for diagnosis and prediction of heart and breast cancer disease. *Review Of Research*, *3*(8).

[4] Chaurasia, V., & Pal, S. (2017). A novel approach for breast cancer detection using data mining techniques. *International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization) Vol*, *2*.

[5] Chaurasia, V., Pal, S., & Tiwari, B. B. (2018). Prediction of benign and malignant breast cancer using data mining techniques. *Journal of Algorithms & Computational Technology*, *12*(2), 119-126.

[6] Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785-794). ACM.

[7] Paul, L. C., Suman, A. A., & Sultan, N. (2013). Methodological analysis of principal component analysis (PCA) method. *International Journal of Computational Engineering & Management*, *16*(2), 32-38.

[8] Rivera-Franco, M. M., & Leon-Rodriguez, E. (2018). Delays in breast cancer detection and treatment in developing countries. *Breast cancer: basic and clinical research*, *12*, 1178223417752677.

[9] Shawe-Taylor, J., & Sun, S. (2011). A review of optimization methodologies in support vector machines. *Neurocomputing*, *74*(17), 3609-3618.

[10] Sperandei, S. (2014). Understanding logistic regression analysis. *Biochemia medica: Biochemia medica*, *24*(1), 12-18.