**Breast Cancer Prediction using Machine Learning Classification Models**

**Introduction**

Breast Cancer is the most commonly occurring cancer in women and the second most cancer overall. Identifying breast cancer is difficult and identifying them in the initial stages can help in curing it. According to the World Cancer Research Fund, about 2 million new cases were observed throughout the globe. Every year, more than 1,000 women under age 40 die from breast cancer.[1] Doctors also suggest young women frequently give their tests so that they can identify in the initial or young stage and can fight against it by using proper medication. Early detection can help in increasing the survival rate by suggesting appropriate treatment required to cure the disease.

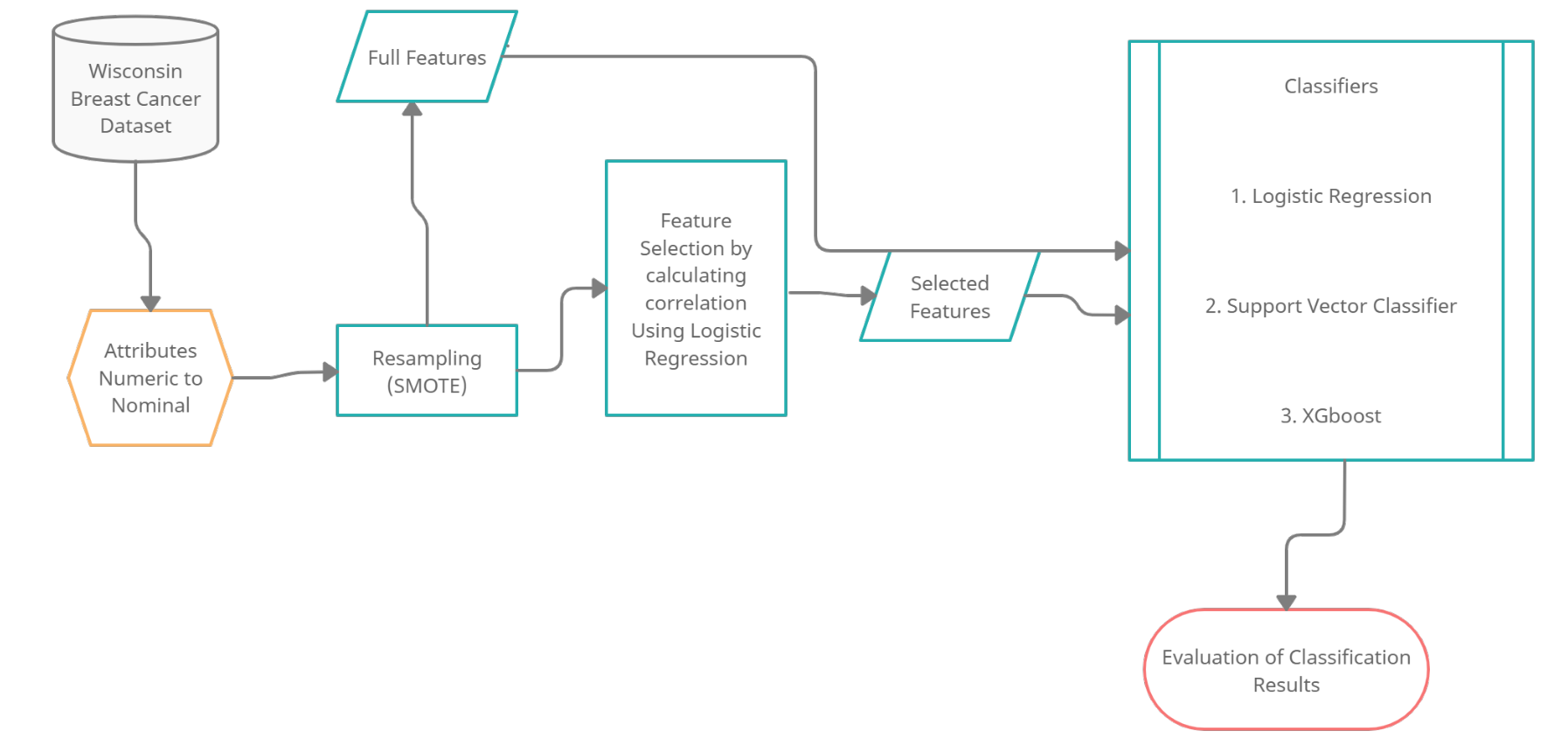
Machine learning plays an important role in the field of health care by detecting the patterns associated with the disease and health conditions by studying thousands of healthcare records. They also improve the accuracy of the treatment protocol which indirectly reduces the cost of the treatment. Not only predicting the disease using NLP techniques are used to understand and classify the clinical documents which reduce the human effort.

**Literature Survey**

Previously a lot more research work has been done using statistical methods. Some of the previous work is shown below Statistical methods are used for grouping various kinds of data such as sound, color, and words. The size of the image should be equal in the training and testing dataset to keep the features constant [2]. A special kind of Support Vector Machine (SVM) using the least square method helps to detect breast cancer. Verisimilitude has been achieved more using cross validating the matrix [3]. The grid search method is combined with an f1-score to give the predictive and classification verisimilitude [4]. The data is imbalanced and previous researchers have performed data balancing such as upsampling or downsampling which will help in increasing the accuracy of the model.

**Data Set**

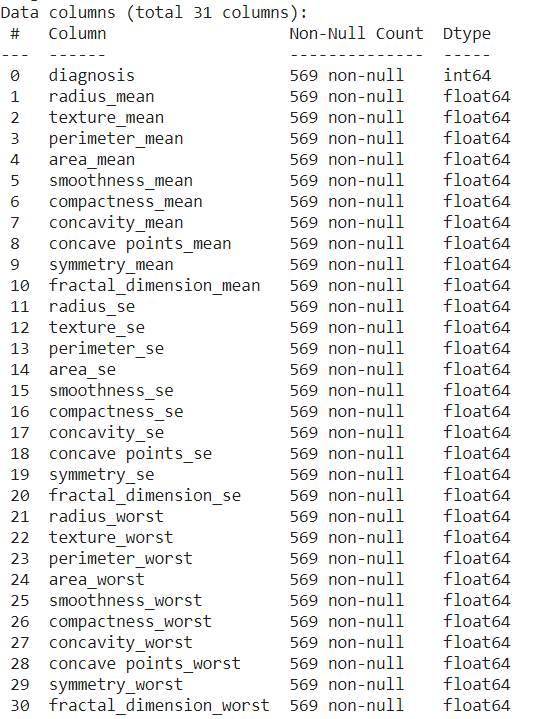
Using Machine learning techniques, we can predict the cells’ behavior and predict whether the cells are benign or malignant. In this project, I am using breast cancer datasets on available repository datasets from the University of California, Irvine. (Breast Cancer Wisconsin (Diagnostic) Data Set)The dataset consists of features computed from digital images of fine needle aspirate of a breast mass. These features represent the characteristics of cell nuclei present in the image.There are about 33 columns or features on cell images of 569 participants.Out of 33 features, 32 features are independent of 569 participants. 63% of cases are benign and 37% are malignant.It shows that this dataset consists of data imbalance due to which the accuracy of classifiers affects a lot.Using feature selection and data balancing techniques to select the most appropriate or important features in predicting the disease and performance of the model.



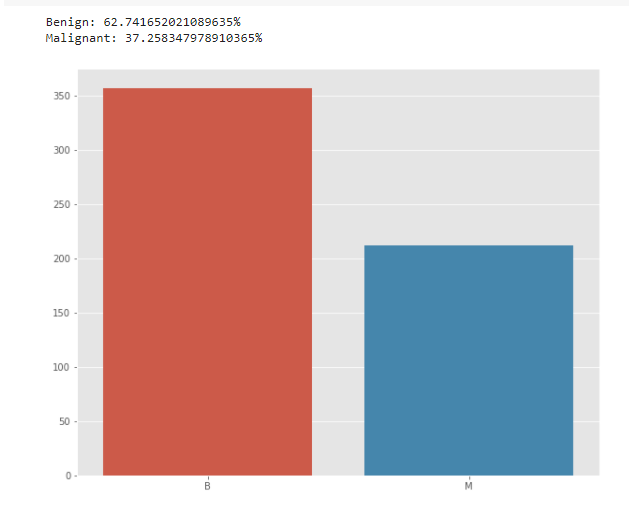
Proposed framework of the Breast Cancer Classification Model

**Data Preprocessing**

Out of 33 columns 2 columns include id and Unnamed column which had null values are not useful in prediction. So, they are dropped and the following shows the list of features in the dataset.



From EDA we found that the data is imbalanced and directly feeding this data to the model will not result in correct accuracy. So, in order to get the better results before feeding the model the data can be either upsampled or downsampled.



**Downsampling**

A mechanism that reduces the count of training samples falling under the majority class. As it helps to even up the counts of target categories. By removing the collected data. But there is a drawback using this model, which is that we might lose important data.

**Upsampling**

As we can see Malignant records are less when compared to Benign records so, synthetically generated Malignant features are injected into the dataset to balance the data. Using SMOTE (Synthetic Minority Oversampling Technique)technique upsampling is performed. It works based on the KNearestNeighbours algorithm, synthetically generating data points that fall in the proximity of the already existing outnumbered group.

**Feature Selection**

After the data is balanced. Bivariate Analysis is performed to know the Pearson correlation between the features and the values >= 0.9. We can eliminate those columns that are highly correlated based on the correlation with the target data.

The correlation between the categorical variables and continuous variables can be achieved by following methods:

1. Point biserial Correlation: It is same as Pearson correlation but it compares categorical or dichotomous variables with the continuous data.
2. Logistic Regression
3. T-Test OR ANOVA

In this paper I used simple logistic regression to find the important features and then eliminate the highly correlated variables.After performing the above step following columns are highly correlated with target variables.

area\_mean

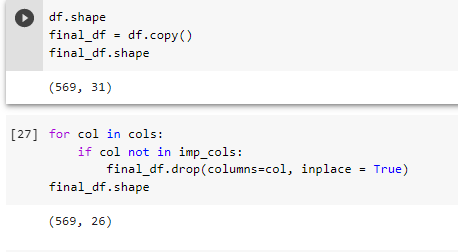
concavity\_mean

concave points\_mean

perimeter\_se

texture\_worst

concave points\_worst



Comparison of data frame shape

**Results**

**Model Comparison**

As mentioned above in the proposed framework diagram following tables shows the performance of each model.

|  |  |
| --- | --- |
| Model | F1 Score |
| Logistic Regression | 94.48 |
| Support Vector Classifier | 97.38 |
| XGboost | 99.52 |

Model Performance after Feature Selection is performed.

**Model Performance Before and After Feature Selection**

I performed feeding the data into the model with and without feature selection. There is not much difference in the accuracy of the model but there is a difference in the precision,recall and F1 values.

|  |  |  |
| --- | --- | --- |
| Property | Before Feature Selection | After Feature Selection |
| Accuracy | 98.59% | 98.41% |
| Precision | 0.98 | 0.99 |
| Recall | 0.97 | 0.96 |
| F1-Score | 0.98 | 0.97 |

Logistic Regression Performance before and after feature selection

|  |  |  |
| --- | --- | --- |
| Property | Before Feature Selection | After Feature Selection |
| Accuracy | 98.24% | 98.41% |
| Precision | 0.98 | 0.99 |
| Recall | 0.97 | 0.96 |
| F1-Score | 0.976 | 0.978 |

SVC Performance before and after feature selection

|  |  |  |
| --- | --- | --- |
| Property | Before Feature Selection | After Feature Selection |
| Accuracy | 99.472% | 99.472% |
| Precision | 0.990 | 0.995 |
| Recall | 0.995 | 0.995 |
| F1-Score | 0.99 | 0.99 |

XGBoost Performance before and after feature Selection

**Discussion**

**Logistic Regression:**

A very convenient and useful side effect of a logistic regression solution is that it doesn’t give you discrete output or outright classes as output. Instead you get probabilities associated with each observation. You can apply many standard and custom performance metrics on this probability score to get a cutoff and in turn classify output in a way which best fits your business problem. A very popular application of this property is scorecards in the financial industry,where you can adjust your threshold [cutoff ] to get different results for classification from the same model. It is pretty efficient in terms of time and memory requirements. It can be applied on distributed data and handle large data on less resources.[6] In addition to the above , logistic regression algorithm is robust to small noise in the data and is not particularly affected by mild cases of multicollinearity.In contrast to the above advantages there are few drawbacks of for Logistic regression it doesn’t perform well when feature space is too large, can’t handle large number of categorical variables, relies on non- linear transformation. It relies on entire data.

In order to overcome the problem of not handling the Non-linearity in Logistic regression. I tried using Support Vector Classification.

**Support Vector Classification (SVC):**

Support Vector Machine (SVM) is a supervised machine learning algorithm. It can be applied to classification or regression tasks. However, it is preferred for classification problems. In the SVM algorithm, each data item is plotted as a point in x-dimensional space (where x is equal to the number of features) with the value of each feature being the value of the particular coordinate. As a solution to separate the two classes of the data points, many possible hyperplanes may be applied. Here the objective is to find a plane that has the maximum distance between data points of each class. By maximizing a margin distance, it is provided with some reinforcement so that future data points can be classified with more confidence. The loss function that helps maximize the margin is hinge loss.[5]

SVM can handle large feature space, non- linear features, and doesn’t rely on entire data. In contrast to the advantages, SVM doesn’t perform well when the feature space is huge because it takes higher training time, and doesn't perform well when there is noise in the data.

SVM doesn’t perform well when there is missing data. It is always better to impute the missing values before feeding into SVM.

In order to overcome the missing data and to reduce the computation training time I tried using XGBoost.

**XGBoost (eXtreme Gradient Boosting):**

XGBoost is an ensemble technique for improving the predictive performance of classification. They can handle interactions, automatically select variables, are robust to outliers, missing data, and numerous correlated and irrelevant variables and can construct variable importance.XGBoost allows user to run a cross-validation at each iteration of the boosting process and thus it is easy to get the exact optimum number of boosting iterations in a single run.

**Comparison of models before and after Selecting Feature Variables**

Feature Selection is very important in the field of medical industry. Though there is not much change in the accuracy of the model, precision and recall values are varying in case of Logistic Regression and Support Vector Classifier. But there is not much change in the XGBoost because it can handle missing data and automatically select variables based on the correlated and irrelevant variables.

**Conclusion**

First of all, we applied the ML classifiers on the pre-processed data in which data imbalance was handled using the re-sampling and SMOTE approach. And tried with the selected and complete features. Among LR, SVC, XGBoost, XGBoost was performing well handling missing values, picking selected features which would impact the target variable.Based on my domain knowledge it is always important to classify the Malignant, Benign properly because if the classification is not done there might be chances of impacting the lives of the patients. So, when performing ML algorithms we shouldn’t completely rely on Accuracy values, should check for precision, recall values so that they don't do any misclassification. In case LR, SVC there are more misclassifications when we have considered complete features.So, it is always better to go with the features that are impacting the target variables instead of considering all the feature variables.

**References**

[1] American Cancer Society. Breast Cancer Facts & Figures 2015- 2016. Available

[2] Polat K, Güneş S 2007 Breast cancer diagnosis using least square support vector machine Digit. Signal Process., 17 694–701.

[3] Chen HL, Yang B, Wang G, Wang SJ, Liu J, Liu DY 2011 Support Vector Machine Based Diagnostic System for Breast Cancer Using Swarm Intelligence J. Med. Syst 36 2505– 2519.

[4] Othman DMF, Yau TMS 2006 Comparison of Different Classification Techniques Using WEKA for Breast Cancer in 3rd International Conference on Biomedical Engineering Kuala Lumpur (Springer) pp 511.

[5]Solanki, Y. S., Chakrabarti, P., Jasinski, M., Leonowicz, Z., Bolshev, V., Vinogradov, A., Jasinska, E., Gono, R., & Nami, M. (2021). A Hybrid Supervised Machine Learning Classifier System for Breast Cancer Prognosis Using Feature Selection and Data Imbalance Handling Approaches. *Electronics*, *10*(6), 699. https://doi.org/10.3390/electronics10060699

[6]https://edvancer.in/logistic-regression-vs-decision-trees-vs-svm-part2/