Preprocessing Experimentation for Breast Cancer Classification

Timothy Stockton, Brandon Peddle, Angelica Gaulin, Emma Wiechert
Keene State College, timothy.stockton, brandon.peddle, angelica.gaulin, emma.wiechert@keene.edu

Abstract **- According to the American Cancer Society, breast cancer is the most diagnosed cancer today. It will affect about 13% of women at some point in their lives [1]. The incidence rate is on the rise, and according to research, the survival rate 10 years after diagnosis is 84% [2]. When a cancerous mass forms, it is important to know whether it is malignant or benign as the former can spread throughout the body and cause further harm. Early detection saves lives, as it allows for more treatment options. However, 41% of breast cancers were not detected when those screened had very dense breast tissue [3]. It is also difficult to observe small lesions [4]. Computer aided diagnosis can improve the prediction accuracy and has been adopted for the classification of breast cancer. There are several different datasets, machine learning (ML) models, pre-processing techniques, and more available today, with each combination yielding different results. Our research is focused on the use of different feature selection and extraction techniques. The methods we have chosen to use are correlation-based feature selection, recursive feature elimination, linear discriminant analysis, principal component analysis, and combination approaches of these algorithms. We hypothesize that at least one of these combination algorithms will outperform other feature selection/extraction methods and/or combinations.**

Index Terms - Correlation based Feature Selection (CFS), linear Discriminant Analysis (LDA), Principal Component Analysis (PCA), Recursive Feature Elimination (RFE)

INTRODUCTION

Breast cancer is a common disease in the present day. Women are more likely to be diagnosed, and it affects millions every day. It is the most common type of cancer diagnosis, making up 30% (or approximately 1 out of 3) cancer diagnoses among women. It has also caused more deaths than almost any other kind of cancer; it is second only to lung cancer. About 13% of women (1 out of every 8) will be diagnosed with breast cancer within their lifetimes. Its incidence is on the rise, with diagnosis rates increasing by 0.5% every year [1].

There are many potential risk factors for breast cancer. Those which an individual could change include excess weight, exercise, alcohol consumption, smoking, breastfeeding, and use of menopausal hormones. Other risk factors include personal and family history of cancer, tall height, high bone mineral density, naturally high levels of certain endogenous sex hormones, use of oral contraceptives, and diagnosis of ductal carcinoma in situ (DCIS), lobular carcinoma in situ (LCIS), or benign breast disease [2].

Early diagnosis and treatment of breast cancer goes along with a better prognosis. The longer a cancer is present, the more time it has to grow and spread in the body. There are also more treatment options available for smaller cancers. As a result, early diagnosis and treatment is critical.

Breast cancer can be difficult to diagnose, even for trained specialists. Although a variety of diagnosis methods exist, breast cancer is typically screened for via mammogram. It is recommended that all women from age 45 to 54 receive annual mammograms for breast cancer screening, then biennially afterwards for as long as they are in good health [2]. However, mammograms have their limitations. Nearly half of women aged 40 or older have dense breast tissue, which makes mammogram results more difficult to interpret [5]. According to one study, 41% of breast cancers were not found when mammograms were performed on very dense breast tissue [3]. In addition, mammograms are weak at detecting small lesions [4]. This can cause breast cancers to go unnoticed, giving them the opportunity to spread and become more likely to be fatal.

Machine learning (ML) is a kind of artificial intelligence. It is focused on determining which statistical algorithms can make accurate predictions with a given set of data. If predictions are accurate enough, the ML model is generally able to perform tasks without human instruction.

There are some ML models that are commonly used, such as linear regression (LR), decision tree (DT), random forest (RF), support vector machine (SVM), neural network (NN), and XGBoost tree (XGB). Their performance varies depending on their application, and sometimes even the specific datasets used with them. Information within a dataset is split up for training and testing ML models. The standard train/test split is 30% for training and 70% for testing, although some researchers prefer using alternative percentages.

In recent years, the medical industry has begun using ML to aid with diagnosis for a plethora of diseases and conditions. There are now computer systems designed to help health professionals better detect signs of disease. This is called computer aided diagnosis (CAD). ML algorithms are able to inspect information thoroughly and much more

quickly than a human. This allows diagnoses to be made more quickly and accurately.

Breast cancer research and diagnosis are benefitting from CAD systems. Investigation of test results by both ML and a human specialist can cut down on undiagnosed breast cancer cases. This clears the way to better prognoses and fewer deaths.

The purpose of this study is to expand on previous research done on the application of machine learning in breast cancer diagnosis. The research done in this study explores the idea of alternative approaches to data preprocessing and how they affect various models. This will help advance our ability to diagnose breast cancer effectively and quickly.

In this study 8 different approaches to data preprocessing were done, including a control set, this will demonstrate that models' accuracy results can be further optimized with a preprocessing method that tailors to the model.

RELATED RESEARCH

Data preprocessing is a common practice in ML research. Preprocessing helps control for missing, redundant, or bad information within a dataset, allowing the ML models to learn more efficiently. This can be done a variety of ways, including feature selection, feature extraction, sampling, scaling, and manual data manipulation.

Elsadig et al. calculated Chi-square, ReliefF, ANOVA, Gini Index, and Gain ratio for feature selection on their dataset, resulting in 17 integral features [6]. Meanwhile, Hassan et al. utilized the Least Absolute Shrinkage and Selection Operator (LASSO) to isolate the 14 most important features. Hassan et al. also preprocessed their data with Standard Scaler for normalization [7].

Fulorunso et al. preprocessed their dataset via sampling. They experimented with the use of Edited Nearest Neighbor (ENN), Synthetic Minority Oversampling Technique (SMOTE), and a combination of both approaches (SMOTEENN), finding that SMOTEENN sampling performed best [8].

Some researchers choose to manually correct erroneous data before feeding it to ML algorithms. This is what Poornajaf and Yousefi did with their dataset before giving it to their ML models [9].

Abhishek S. Powar et al. [10] handled missing or inconsistent/erroneous values and removed duplicates to reduce redundancy and potential bias as well as improve the dataset quality. They selected features based on relevancy and prediction impact using correlation analysis. The features were scaled and normalized so that large value features do not dominate the learning process. They used a 60-40 train-test split.

Sahar Arooj et al. [11] believe that the complexity of ML procedures such as preprocessing, and feature extraction factor into a decrease in efficiency and accuracy. They used a train-test split of 80-20.

Reza Rabiei et al. [12] used Synthetic Minority Oversampling Technique (SMOTE) to "balance the training data due to the difference in the number of study class records".

Esraa A. Mohamed et al. [13] state that Gomaz et al. studied the impact of data preprocessing.

David A. Omondiagbe et al. [14] centered, scaled, and cleaned their data. They used feature selection techniques (CFS and RFE) as well as feature extraction methods (PCA and LDA). Their results show that SVM with RBF kernel and LDA preprocessing can produce promising results and that an ANN model built with LDA preprocessed data performs better than the other ANN models. They have a table of the model and technique combinations they used.

Ahmed Elazab et al. [15] mention feature selection techniques such as PCA and LDA, as well as filtering techniques such as chi-square test used to select the most discriminative features to avoid overfitting and reduce the redundancy in feature space.

Dahri et al. utilized the scikit-learn Standard Scaler module to transform their dataset, ensuring all features were close in numerical scale [16]. The researchers then proceeded to implement feature selection to retain only the features which exerted the greatest variance necessary to properly predict on records.

By contrast, Amethiya et al. catalogued a pipeline in which the dataset is loaded, and then feature extraction techniques were executed directly upon the raw data [17]. They include a figure which notes preprocessing as an interstitial process, but do not elaborate on the exact methods used in the associated description.

Khater et al. explain in detail their methodology of cleaning the data for erroneous and misleading datapoints which taint the fidelity of the dataset, and its ability to later train models [18]. The researchers also utilize what they describe as a gamut of feature selection techniques before moving on to model training.

Silva-Aravena et al. utilized label encoding to prepare the label in their unique dataset for each record, transforming the malignant/benign designation into a numerical format which is digestible by machine learning algorithms [19].

Harinishree et al. catalogued the findings of numerous similar studies in a comparative study. The authors also describe numerous datasets utilized in previous studies to train models to detect breast cancer. These datasets include the MIAS dataset and the ubiquitous Wisconsin Breast Cancer Dataset [20].

METHOD

I. Data Set

The data set we used is the Wisconsin Diagnostic Breast Cancer data set. It contains 32 features, including a benign/malignant classification, and information about clump radius, texture, perimeter, area, smoothness, compactness, concavity, symmetry, and fractal dimensions. These features are derived from mammograms which allow

us to visualize the internal structures of the breast tissue using x-rays [15]. More recently, data sets containing data derived from thermographic imaging are being used because thermography is non-invasive, low-cost, and radiation-free [14].

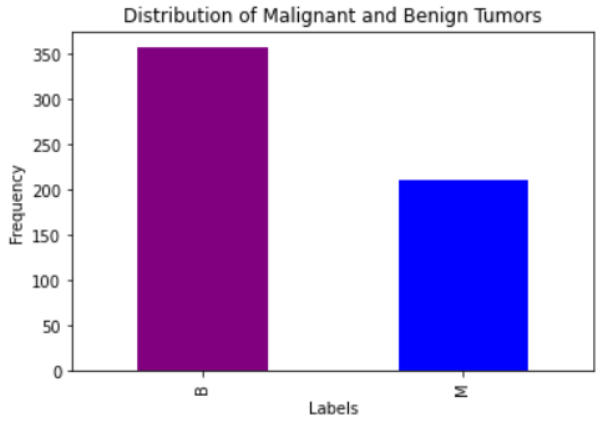
II. Cleaning

The data set must be "cleaned" in order to account for things such as: missing data, incomplete data, erroneous values, and duplicates. The way we did this was by checking for null values within the data set, which we did not find, using isna().sum(). Next, we used LabelEncoder to convert the diagnosis label strings, M for malignant and B for benign, into integers of 1 and 0 respectively so that the machine learning models could utilize this feature. Then we used RobustScaler to scale the data, ensuring that extreme values do not skew the results. It puts all features on the same scale between 0 and 1 so that although a number may be large or small, it is processed relative to its position within that feature only.

This cleaning is performed on the original data set and is performed prior to making duplicates so that each data set receives the same cleaning.

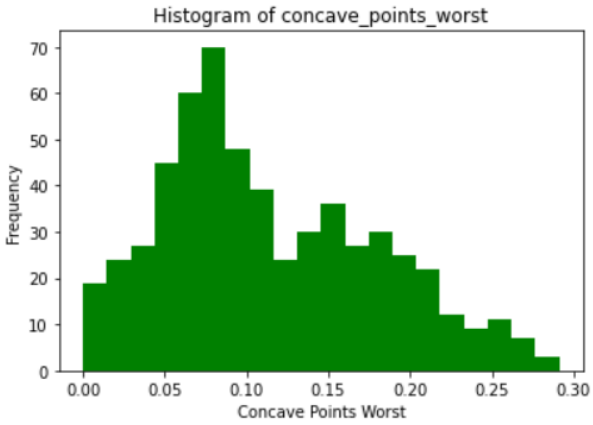
II. Univariate Analysis

Figure I.



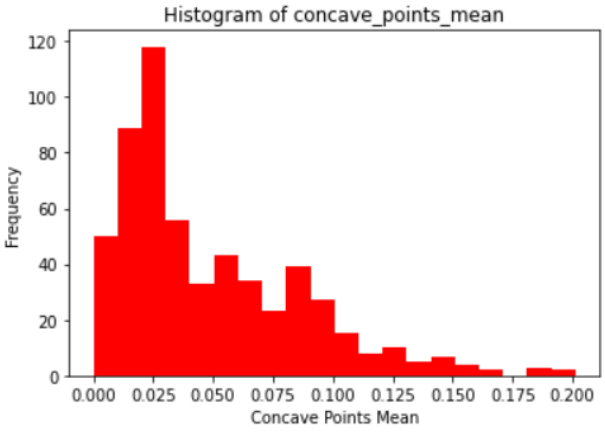
This distribution shows the number of malignant and benign tumors in the Wisconsin Breast Cancer dataset. The number of benign tumors in this dataset is 350. The number of malignant tumors in this dataset is roughly 200.

Figure I.



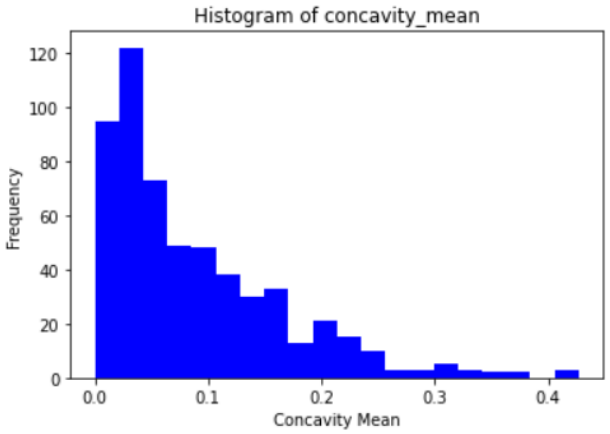
This histogram shows the spread of feature concave_points_worst in the Wisconsin Breast Cancer dataset. The highest value of this feature is 0.29, and the value with the highest number of occurrences is 0.8.

Figure III.



This histogram shows the spread of values in the feature concave_points_mean in the Wisconsin Breast Cancer dataset. The highest value of this feature is 0.195, and the lowest value is 0.005. The highest number of occurrences is 0.03 with roughly 118 occurrences.

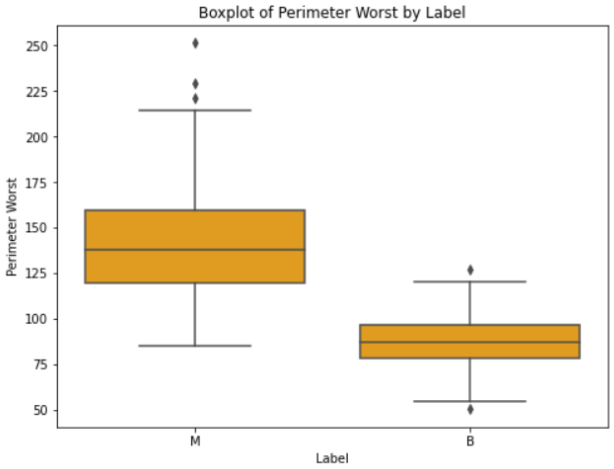
Figure IV.



This histogram shows the values stored in the Wisconsin Breast Cancer dataset for the feature concavity_mean. The highest value in this feature is 0.425, and the lowest value is 0.025. The highest number of occurrences in this feature is 0.05, with 120 occurrences.

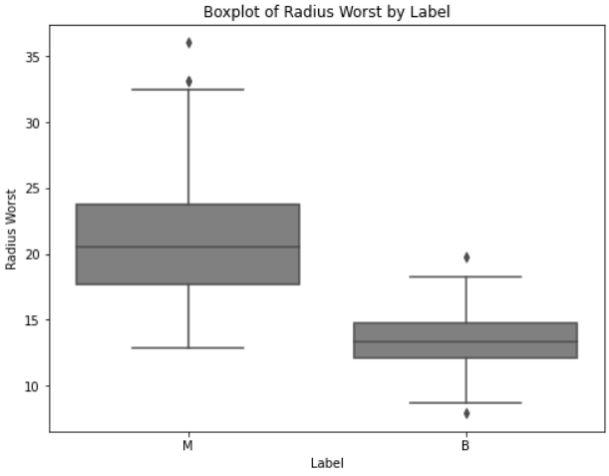
III. Bivariate Analysis

Figure V.



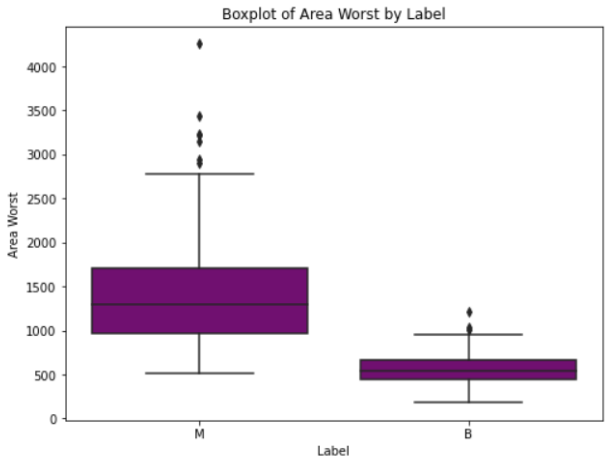
This boxplot compares the features Perimeter Worst with Label, this shows the correlation between the data collected on Perimeter Worst and the diagnosis assigned to the mammogram sample. The mean of the malignant tumors perimeter worse value is around 132. The mean of the benign tumors is around 87.5. The minimum perimeter worst value for malignant tumors is 87.5 and for benign tumors is around 60. The maximum perimeter worst value for malignant tumors is 210, and the maximum perimeter for benign tumors is 110. The outliers for malignant tumors are 215, 230, and 250. The outliers for benign tumors are 65, and 130. This shows that malignant tumors tend to score higher in the perimeter worst feature but, there is some indication that other features are needed to successfully determine if a tumor is malignant or not. This indicates that perimeter worse, in conjunction with other features, is an excellent feature to use in the feature selection for the models tested.

Figure VI.



This boxplot compares the Label feature with the Radius Worst feature, this shows the correlation between Radius Worst and the tumors diagnosis label. The mean of the malignant tumors is 22. The mean of the benign tumors is 13. The minimum value of Radius Worst for malignant tumors is 13, and the minimum value of Radius Worst for benign tumors is 8. The maximum value of Radius Worst for malignant tumors is 33, and the maximum value of Radius Worst for benign tumors is 18. The outliers for malignant tumors are 33.5, and 36. The outliers for benign tumors are 7 and 21. This shows that Radius Worst is a feature that has higher values in malignant tumors, and lower in benign tumors. In conjunction with other features, this feature is excellent for testing the models in this study.

Figure VII.



This boxplot compares the label feature with the Area Worst feature, showing the correlations between the Area Worst and breast cancer diagnosis. The minimum value for Area

Worst in malignant tumors is 500 and the minimum value for Area Worst in benign tumors is 200. The maximum value for Area Worst in malignant tumors is 2800 and the maximum value for Area Worst in benign tumors is 1000. The mean value of Area Worst in malignant tumors is 1400, and the mean value of Area Worst in benign tumors is 500. The outliers of Area Worst in malignant tumors are 2850, 2900, 3100, 3200, and 3400. The outliers of Area Worst in benign tumors are 1100, and 1300. This shows that higher values in Area Worst are related to malignant tumors and the lower values are related to benign tumors. In correlation with other features the feature Area Worst is an excellent feature to apply to machine learning models to diagnose breast cancer.

IV. Feature Selection

In feature selection, only features with information useful for classification are kept while the others are dropped. This is determined by the technique used. Our research utilizes two different feature selection techniques which are correlation-based feature selection (CFS) and recursive feature elimination (RFE). We created nine duplicates of the original cleaned data set. Three of these duplicates then have CFS applied, while another three of the duplicates have RFE applied.

Specifically, what CFS does is look at the features intrinsic properties and removes features that correlate highly with other features because they provide redundant information [14]. CFS does not require the use of a ML model and derives its own weights. We retained 6 features including: perimeter worst, radius worst, area worst, concave points worst, concave points mean, and concavity mean.

What RFE does is build a ML model with all of the features and then rank them based on how important they are to reducing the amount of error in the model [14]. We used a random forest classifier model for this and retained the six highest ranked features.

V. Feature Extraction

Feature extraction involves the combination of features and reduction of dimension. The techniques we use are principle component analysis (PCA) and linear discriminant analysis (LDA). First, we apply PCA to one of the data set duplicates that did not have a feature selection technique applied, and then we do the same for LDA leaving a single control data set duplicate with no feature selection or feature extraction techniques applied. Next, we apply PCA to a duplicate that had CFS applied and then do the same for LDA on another CFS duplicate. Finally, we use PCA on a duplicate that has RFE applied and then use LDA on another RFE duplicate.

Specifically, what PCA does is transform the data set by combining features in a way that they do not correlate [14]. These new features are called principle components. We specified an output of six components which has an explained variance of roughly 89%.

What LDA does is compute feature transformations such that the scatter between classes is maximized, and the scatter within each class is minimized [14].

VI. Train-Test Split

At this stage, we have nine different data sets derived from the cleaned original. Specifically, we have one with only CFS applied, one with only RFE applied, one with only LDA applied, and one with only PCA applied. We also have combination data sets including: one with CFS and LDA applied, one with CFS and PCA applied, one with RFE and LDA applied, and one with RFE and PCA applied. Finally, we have a dataset that had no feature selection or feature extraction techniques used which acts as a control data set.

Each of our nine data sets are now split into an X variable which contains all features other than the labels and a Y variable which contains only the labels. These variables are used to build the training and testing sets. We used a 70-30 split which means the training set contains 70% of the records and the testing set contains 30% of the records.

VII. Logistic Regression

Our first ML model used is linear regression which applies a non-linear transformation to the input variables (features) such that the output is restricted to either 0 or 1. In our case, the model's output is a determination of either being benign or malignant. This is achieved with a cutoff value that can be adjusted and acts as a decision boundary. The logistic function (sigmoid function) at the heart of logistic regression will model the probability of a record belonging to a certain class. This probability is then compared to the cutoff value and, if it is higher than the cutoff, will be classified as 1 or, if it is lower than the cutoff, will be classified as 0.

Our study uses the default cutoff of 0.5 because we are not fine tuning our models. The point of this study is to compare the outcome of different pre-processing techniques which can be seen regardless of the model tuning provided all data sets use the same model settings. We created nine separate linear regression models, each using a different one of our data sets to train and test it.

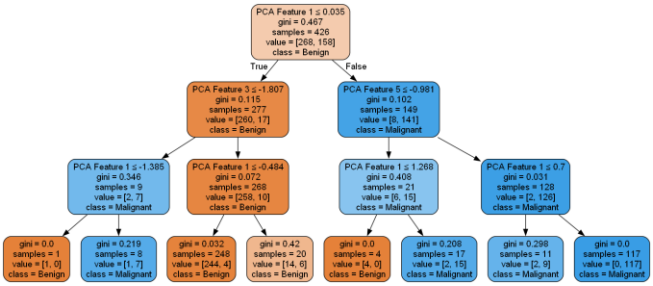
VIII. Decision Tree

Our second ML model used is decision tree which is a series of decision rules similar to if-else statements. It is a tree because each decision branch, starting with the first, is followed by either another decision branch or an end point (leaf). These endpoints state a specific class, and any inputs that reach that endpoint via the decisions are determined to be a part of that class. A decision tree can be incredibly complex or very simple. When not using combination features (meaning each feature is explainable, not the case with PCA components), the tree itself can be easily explained and it is quite clear why the classification is made.

In our study, we used the default max_depth of none, and the default min_samples_leaf of 1 because the focus of the study is on pre-processing techniques. We created nine separate decision tree models, each using a

different one of our data sets to train it and test it. Figure VIII shows a decision tree similar in accuracy to the models we used, but it is smaller. The bottom right leaf has the most effective decision rule, whether the unexplainable PCA feature 1 is greater than 0.7. It has a gini index value of zero meaning it is pure and has 117 labels of only a single class.

FIGURE VIII.
METHODOLOGY FLOWCHART



IX. Random Forest

Our third ML model used is random forest which is comprised of multiple decision trees as an ensemble [12]. Each decision tree within the forest has unique splitting criteria and uses a different subset of the data. They will each produce an individual classification based on the input. These classifications are combined as a majority vote which gives us a probabilistic classification.

In this study, we used the default model settings and created nine separate random forest models, each using a different one of our data sets to train it and test it.

X. SVM

Our fourth ML model used is a support vector machine (SVM) which creates a hyperplane in an optimal position that linearly separates the data points into two classes based on their features [10]. The points closest to the hyperplane are the support vectors, and the distance between the support vectors and hyperplane is called the margin. This optimization is possible because the optimal hyperplane will maximize the margin. If the data is not linearly separable in its current dimension, it will be mapped to a higher-dimensional space by a kernel function such as polynomial kernel or radial basis function (RBF). There is a cost parameter, called c, which defines how hard the margin should be. Smaller c values have a higher tolerance for misclassification and a wider margin, while lower c values have a lower tolerance for misclassification and a smaller margin.

In this study, we used a linear kernel and a c of 10. We created nine separate SVM models, each using a different one of our data sets to train it and test it.

XI. Neural Network

Our fifth ML model used is a neural network (NN) which is a set of inter-connected nodes, also called neurons, organized into different layers. We specifically used a multi-layer perceptron (MLP) NN where data flows in one direction starting with an input layer, then the computational hidden layers, and finally an output layer [12]. NN's are based on how a brain works, and each node will either be active or inactive. The state of each node will impact the nodes in the next layer. This state of being either active or inactive is controlled by means of an activation function such as hyperbolic tangent (tanh), rectified linear (relu), and logistic function (sigmoid).

In our study, we used a NN with two 5x5 hidden layers, a logistic activation function, a stochastic gradient descent (SGD) solver, batch size of 25, and a learning rate of 0.1. We created nine separate neural network models, each using a different one of our data sets to train it and test it.

XII. XGBoost Tree

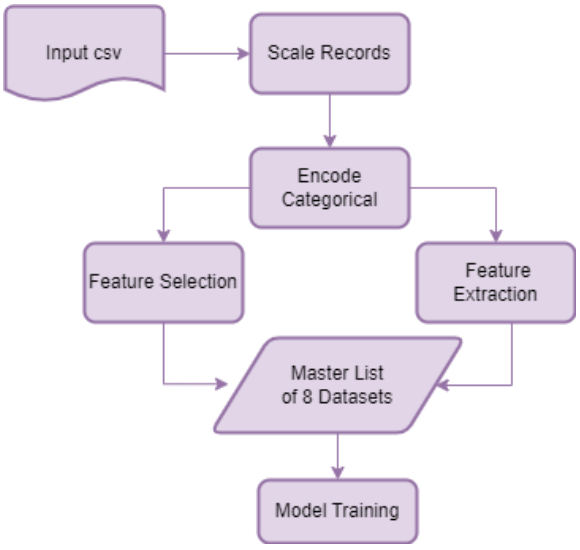
Our sixth and final ML model used is an extreme gradient boosting (XGBoost) tree, meaning it has improved regularization compared to a standard gradient boosting tree (GBT), which consists of many small decision trees called weak learners. Each tree will additively correct the errors from previous trees such that the accuracy of the prediction will increase with each tree [12]. The final prediction will be a summation of each tree's individual prediction. Small trees are used to avoid overfitting. There is a learning rate parameter which scales a tree's output. A high learning rate makes the model less robust but requires less trees, while a lower learning rate makes the model more robust and requires more trees.

In this study, we used the default learning rate of 0.1, the default max_depth of 3, and an objective of "multi:softmax". We created nine separate XGBoost tree models, each using a different one of our data sets to train it and test it.

XIII. Method Summary

First, we obtained the Wisconsin Breast Diagnostic Cancer data set. Then we performed some general cleaning on the data set including handling missing entries, scaling, and encoding. We then used different combinations of pre-processing techniques to make nine unique data sets. These data sets had a 70-30 train-test split applied. Next, we created two lists, one with all the training data sets and one with all the testing data sets. For each model, we instantiated it and then made a loop in which the model was fit with each data set and then its accuracy was calculated. This resulted in 9 variations of each model with 54 different models in total. Refer to our flowchart. We used the default settings for each model because this study's focus is on pre-processing techniques.

FIGURE IX.
METHODOLOGY FLOWCHART



RESULTS/SUMMARY

Table I.
Experimental Results Tabulated

Algorithm	Preprocessing Combination	Accuracy	Precision	Recall	F1-Score
LR	Control	0.96479	0.98	0.92453	0.95146
	CFS	0.92254	1	0.79245	0.88421
	RFE	0.92254	1	0.79245	0.88421
	LDA	0.97183	0.96226	0.96226	0.96226
	PCA	0.97183	1	0.92453	0.96078
	CFS -> LDA	0.14789	0.01429	0.01887	0.01626
	CFS -> PCA	0.92254	1	0.79245	0.88421
	RFE -> LDA	0.92254	0.9375	0.84906	0.89109
	RFE -> PCA	0.92254	1	0.79245	0.88421
DT	Control	0.89437	0.83929	0.88679	0.86239
	CFS	0.91549	0.90196	0.86792	0.88462
	RFE	0.91549	0.90196	0.86792	0.88462
	LDA	0.94366	0.92453	0.92453	0.92453
	PCA	0.9507	0.92593	0.9434	0.93458
	CFS -> LDA	0.1338	0.03947	0.0566	0.04651
	CFS -> PCA	0.88028	0.84615	0.83019	0.8381
	RFE -> LDA	0.90845	0.87037	0.88679	0.8785
	RFE -> PCA	0.88028	0.84615	0.83019	0.8381
RF	Control	0.92958	0.95745	0.84906	0.9
	CFS	0.91549	0.93617	0.83019	0.88
	RFE	0.90141	0.89796	0.83019	0.86275
	LDA	0.94366	0.92453	0.92453	0.92453
	PCA	0.92254	0.90385	0.88679	0.89524
	CFS -> LDA	0.1338	0.03947	0.0566	0.04651
	CFS -> PCA	0.92254	0.97727	0.81132	0.8866
	RFE -> LDA	0.90845	0.87037	0.88679	0.8785
	RFE -> PCA	0.92254	0.97727	0.81132	0.8866
SVM	Control	0.95775	0.96078	0.92453	0.94231
	CFS	0.92958	0.97778	0.83019	0.89796
	RFE	0.92958	0.97778	0.83019	0.89796
	LDA	0.97183	0.96226	0.96226	0.96226
	PCA	0.96479	0.98	0.92453	0.95146
	CFS -> LDA	0.14789	0.01429	0.01887	0.01626
	CFS -> PCA	0.92958	0.97778	0.83019	0.89796
	RFE -> LDA	0.92254	0.9375	0.84906	0.89109
	RFE -> PCA	0.92958	0.97778	0.83019	0.89796
NN	Control	0.96479	0.98	0.92453	0.95146
	CFS	0.90845	0.95455	0.79245	0.86598
	RFE	0.92254	1	0.79245	0.88421
	LDA	0.97183	0.96226	0.96226	0.96226
	PCA	0.96479	0.98	0.92453	0.95146
	CFS -> LDA	0.1338	0.01389	0.01887	0.016
	CFS -> PCA	0.91549	0.97674	0.79245	0.875
	RFE -> LDA	0.92958	0.92157	0.88679	0.90385
	RFE -> PCA	0.91549	0.97674	0.79245	0.875
XGBoost	Control	0.93662	0.95833	0.86792	0.91089
	CFS	0.90845	0.9	0.84906	0.87379
	RFE	0.90141	0.89796	0.83019	0.86275
	LDA	0.94366	0.92453	0.92453	0.92453
	PCA	0.94366	0.92453	0.92453	0.92453
	CFS -> LDA	0.1338	0.03947	0.0566	0.04651
	CFS -> PCA	0.90845	0.95455	0.79245	0.86598
	RFE -> LDA	0.91549	0.88679	0.88679	0.88679
	RFE -> PCA	0.90845	0.95455	0.79245	0.86598

The researchers of this study created 8 modified training sets, each transformed using unique preprocessing techniques, and an unmodified control (excepting scaling) from the original Wisconsin Breast Cancer Dataset. Necessarily, 8 corresponding preprocessed testing sets, and a control, were also created to test the models. 6 Machine Learning algorithms were then trained on each of the 9 possible training sets, and the resulting models were evaluated using 4 popular statistical methods: Accuracy, Precision, Recall, and the F1-Score [].

In general, the results of this study seem to suggest overall that depending on the model, the PCA and LDA preprocessing transformations seem to provide the best overall preparation of the dataset for algorithmic classification. Despite the simplicity of use and prevalence in the previous literature of Principle Component Analysis, the authors of this study found the LDA method to be a highly defensible and visibly effective preprocessing and

feature reduction strategy for the stated goal. The LDA preprocessing technique provided exemplary scores when used in conjunction with the Logistic Regression, Support Vector Classifier, and Multi-Layer Perceptron ML algorithms. In cases of comparable performance between two training sets from the same ML algorithm, the F1-Score was generally considered the tiebreaker. It is worth noting that the PCA preprocessing technique did extremely well in many areas and could easily overtake LDA as the preferred choice, depending on the model choice and use-case of an implementation.

Table II.
Summary of Best Performing Preprocessing Algorithms by ML Model

Algorithm	Preprocessing Combination(s)	Accuracy(s)	Precision	Recall	F1-Score
LR	LDA	0.97183	0.96226	0.96226	0.96226
DT	PCA	0.9507	0.92593	0.9434	0.93458
RFC	LDA	0.94366	0.92453	0.92453	0.92453
SVM	LDA	0.97183	0.96226	0.96226	0.96226
NN	LDA	0.97183	0.96226	0.96226	0.96226
XGBoost	LDA	0.94366	0.92453	0.92453	0.92453
	PCA	0.94366	0.92453	0.92453	0.92453

For the purposes of this study, the Logistic Regression algorithm provided the most impressive testing scores when exposed to new data, while remaining the simplest algorithm; LR was therefore declared the "best" performing model. Nevertheless, it is worth restating that the models in this study intentionally received little to no hyperparameter tuning. This decision to use default model training settings was made to demonstrate the unaltered effects of preprocessing on the results of the training and testing process. Therefore, while the Logistic Regression Model operated most impressively here as LR has relatively few hyperparameters to tune, other models would likely surpass that performance with further optimization. Of special interest to the researchers is the Support Vector Classifier, whose linear functions synergize well with the underlying linear mathematics of the LDA and PCA modules, and whose processing time easily outpaced the more complex Multi-Layer Perceptron. Finally, even without tuning, all the best-performing test models performed better than those trained on the unaltered control sample. This result, while providing a final interesting finding, also underscores the potential for preprocessing to enhance the accuracy of breast cancer detection to almost 100% with the proper combination of preprocessing algorithms and model hyperparameter tuning. Of course, it must be noted that the streamlining of feature extraction comes at the cost of feature transparency in the final model's decisions, reducing the explainability of the machine learning model.

ACKNOWLEDGMENT

We would like to thank Dr. Wei Lu for introducing us to these machine learning concepts, the University of Wisconsin for the dataset, and sklearn for the python machine learning libraries.

REFERENCES

[1] American Cancer Society, "About Breast Cancer," 2023. Available: https://www.cancer.org/content/dam/CRC/PDF/Public/8577.00.pdf

[2] American Cancer Society, "Breast Cancer Facts & Figures 2022-2024," American Cancer Society, 2022. Accessed: Dec. 11, 2023. [Online]. Available: https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/breast-cancer-facts-and-figures/2022-2024-breast-cancer-fact-figures-acs.pdf

[3] P. B. Gordon, "The Impact of Dense Breasts on the Stage of Breast Cancer at Diagnosis: A Review and Options for Supplemental Screening," *Current Oncology*, vol. 29, no. 5, pp. 3595–3636, May 2022, doi: https://doi.org/10.3390/curroncol29050291.

[4] H.-L. Chen, J.-Q. Zhou, Q. Chen, and Y.-C. Deng, "Comparison of the sensitivity of mammography, ultrasound, magnetic resonance imaging and combinations of these imaging modalities for the detection of small (≤2 cm) breast cancer," *Medicine*, vol. 100, no. 26, p. e26531, Jul. 2021, doi: https://doi.org/10.1097/MD.0000000000026531.

[5] National Cancer Institute, "Dense Breasts: Answers to Commonly Asked Questions - National Cancer Institute," *www.cancer.gov*, Feb. 16, 2018. https://www.cancer.gov/types/breast/breast-changes/dense-breasts (accessed Dec. 11, 2023).

[6] M. A. Elsadig, A. Altigani, and H. T. Elshoush, "Breast cancer detection using machine learning approaches: a comparative study," *International Journal of Power Electronics and Drive Systems*, vol. 13, no. 1, p. 736, Feb. 2023, doi: 10.11591/ijece.v13i1.pp736-745.

[7] Md. M. Hassan *et al.*, "A comparative assessment of machine learning algorithms with the Least Absolute Shrinkage and Selection Operator for breast cancer detection and prediction," *Decision Analytics Journal*, vol. 7, p. 100245, Jun. 2023, doi: 10.1016/j.dajour.2023.100245.

[8] S. O. Fulorunso, J. B. Awotunde, A. A. Adigun, et al. "A hybrid model for post-treatment mortality rate classification of patients with breast cancer," *Healthcare Analytics*, vol. 4, p. 100254, Dec. 2023, doi: 10.1016/j.health.2023.100254.

[9] M. Poornajaf and S. Yousefi, "Improvement of the performance of machine learning algorithms in predicting breast cancer," *Frontiers in Health Informatics*, vol. 12, p. 132, Mar. 2023, doi: 10.30699/fhi.v12i0.400.

[10] Powar, Abhishek S. and Batwal, Aditit P. et al. May 2023. "Research Paper On Enhancing Breast Cancer Prediction Through Machine Learning And Cross-Validation Techniques: A Comparative Analysis."

[11] Arooj, Sahar and Zubair, Muhammad et al. July 2022. "Breast Cancer Detection And Classification Empowered With Transfer Learning."

[12] Rabiei, Reza and Sohrabei, Solmaz et al. June 2007. "Prediction Of Breast Cancer Using Machine Learning Approaches." *J Biomed Phys Eng 2002* 12(3), pp. 297 – 308.

[13] Mohamed, Esraa A. and Rashed, Essam A. et al. January 2022. "Deep Learning Model For Fully Automated Breast Cancer Detection System From Thermograms."

[14] David A. Omondiagbe et al. 2019 "Machine Learning Classification Techniques For Breast Cancer Diagnosis." IOP Conf. Ser.: Mater. Sci. Eng. 495 012033

[15] Elazab, Ahmed and Lei, Baiying et al. 2019. "Breast Cancer Detection And Diagnosis Using Mammographic Data: Systematic Review." *J Med Internet Res 2019* 21(7):e14464.

[16] H. Dhahri, E. Al Maghayreh, and A. Mahmood et al. "Automated Breast Cancer Diagnosis Based on Machine Learning Algorithms," *Journal of Healthcare Engineering*, vol. 2019, pp. 1–11, Nov. 2019, doi: https://doi.org/10.1155/2019/4253641. Available: https://www.hindawi.com/journals/jhe/2019/4253641/

[17] Y. Amethiya, P. Pipariya, and S. Patel, "Comparative Analysis of Breast Cancer detection using Machine Learning and Biosensors," *Intelligent Medicine*, vol. 2, no. 2, Oct. 2021. doi: https://doi.org/10.1016/j.imed.2021.08.004

[18] T. Khater, A. Hussain, and R. Bendardaf et al. "An Explainable Artificial Intelligence Model for the Classification of Breast Cancer," *IEEE Access*, vol. 4, pp. 1, Jan. 2023. doi: https://doi.org/10.1109/access.2023.3308446. Available: https://ieeexplore.ieee.org/document/10229149

[19] F. Silva-Aravena, H. Núñez Delafuente, and J. H. Gutiérrez-Bahamondes, et al. “A Hybrid Algorithm of ML and XAI to Prevent Breast Cancer: A Strategy to Support Decision Making,” *Cancers*, vol. 15, no. 9, pp. 2443–2443, Apr. 2023. doi: https://doi.org/10.3390/cancers15092443. Available: https://www.mdpi.com/2072-6694/15/9/2443

[20] M. S. Harinishree, C. R. Aditya and D. N. Sachin. "Detection of Breast Cancer using Machine Learning Algorithms – A Survey," *Proceedings of the* 5th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2021, pp. 1598-1601. doi: 10.1109/ICCMC51019.2021.9418488.

[21] M. Tahmooresi, D. Remondo, and J. Á. A. Segura, “Breast Cancer Detection Using Machine Learning with Thermograms in an Edge Computing Scenario,” *HealthDL’21: Proceedings of the 2nd Workshop on Deep Learning for Wellbeing Applications Leveraging Mobile Devices and Edge Computing*, pp. 13–16, Jun. 2021, doi: 10.1145/3469258.3469850.

[22] “Today in Science History: Engineering Quotes.” 2012. todayinsci.com/QuotationsCategories/E_Cat/Engineering-Quotations.htm. Web. Accessed: April 9, 2012.

[23] Donohue, Susan K. and Richards, Larry G. October 2011. “P-12 Engineering Education: Using Engineering Teaching Kits to Address Student Misconceptions in Science.” *Proceedings* of the 41st Frontiers in Education Conference, Rapid City, SD, pp. F2A-1 – F2A-3.

[24] Dweck, Carol S. 2006. *Mindset: The New Psychology of Success*, New York: Random House, Inc.

[25] Kaplan, Avi and Maehr, Martin L. June 2007. “The Contributions and Prospects of Goal Orientation Theory.” *Educational Psychology Review* 19(2), pp. 141 – 184.

[26] Dweck, Carol S. “Messages That Motivate: How Praise Molds Students’ Beliefs, Motivation, and Performance (In Surprising Ways).” In Aronson, Joshua (ed.), 2006, *Improving Academic Achievement: Impact of Psychological Factors on Education*. New York: Elsevier Science, pp. 37 – 60.