EED363 Applied Machine Learning Project Report
Date of submission: 30th april' 18

Group Members:
Pavan Sai Gali (1510110137)
Pranav Reddy M (1510110209)
Kavya Kanuri (1510110189)

Breast Cancer Dataset

Objective

Cancer refers to the uncontrolled multiplication of a group of cells in particular location of the body. A group of rapidly dividing cells may form a lump, micro calcifications or architectural distortions which are referred to as tumors. Breast cancer is any form of malignant tumor which develops from breast cells. Breast cancer is one of most hazardous types of cancer among women in the world. Breast cancer is the most common cancer among women and one of the major causes of death among women worldwide.

Every year approximately 124 out of 100,000 women are diagnosed with breast cancer, and the estimation is that 23 out of the 124 women will die of this disease. So, today there is an urgent need in breast cancer control and it is achieved primarily by knowing different risk factors. Secondly, there is need to detect this disease in early stage by knowing different symptoms of this disease, so that it can be cured. So our major objective of our project is efficient prediction of the presence of breast cancer cells in the body by improving performance and accuracy as more inaccurate prediction would lead to severe results.

Currently, the most used techniques to detect breast cancer in early stages are: mammography (63% to 97% correctness), FNA (Fine Needle Aspiration) with visual interpretation (65% to 98% correctness) and surgical biopsy (approximately 100% correctness). Therefore, mammography and FNA with visual interpretation correctness varies widely, and the surgical biopsy, although reliable, is invasive and costly.

Our dataset is taken from the database of Wisconcin Breast cancer (WBC). There are many other Wisconsin Breast cancer datasets. Wisconsin Diagnosis Breast Cancer (WDBC) which has total of 569 instances with 32 attributes and Wisconsin Prognosis Breast Cancer (WPBC) with 198 instances with 34 attributes and all are of 2 classes. References for these datasets are provided at the end of the document.

Problem Statement

The Machine learning problem for Breast cancer analysis is Classification as it can be classified into two parts either malignant tumor or the benign breast mass which we should predict based on the given features:

1. Clump Thickness
2. Uniformity of Cell Size
3. Uniformity of Cell Shape
4. Marginal Adhesion
5. Single Epithelial Cell Size
6. Bare Nuclei
7. Bland Chromatin
8. Normal Nucleoli
9. Mitoses

The features in this dataset characterize cell nucleus properties and were generated from image analysis of Fine Needle Aspirates (FNA) of breast masses. They describe characteristics of the cell nuclei present in the image.

In the Clump thickness benign cells tend to be grouped in monolayers, while cancerous cells are often grouped in multilayer. While in the Uniformity of cell size/shape the cancer cells tend to vary in size and shape. That is why these parameters are valuable in determining whether the cells are cancerous or not. In the case of Marginal adhesion the normal cells tend to stick together, where cancer cells tend to lose this ability. So loss of adhesion is a sign of malignancy. In the Single epithelial cell size the size is related to the uniformity mentioned above. Epithelial cells that are significantly enlarged may be a malignant cell.

The Bland Chromatin describes a uniform "texture" of the nucleus seen in benign cells. In cancer cells the chromatin tends to be coarser. The Normal nucleoli are small structures seen in the nucleus. In normal cells the nucleolus is usually very small if visible. In cancer cells the nucleoli become more prominent, and sometimes there are more of them. The Bare nuclei is a term used for nuclei that is not surrounded by cytoplasm (the rest of the cell). Those are typically seen in benign tumors.

Mitosis is nuclear division and produce two identical daughter cells during prophase. It is the process in which the cell divides and replicates. Pathologists can determine the grade of cancer by counting the number of mitoses

All features are evaluated on a scale from 1 to 10, with 1 being the closest to benign and 10 the closest to malignant. Before being publically available the dataset had 701 points, but on January of 1989, after being revised, 2 instances from group 1 were considered inconsistent and were removed from the dataset. Two more revisions occurred before the actual state of the dataset, both of them aimed to substitute values from zero to one, so the value range of the features is 1-10.

The Class 2 determines benign and class 4 determines malignant. There are 458 samples that come under benign class and 241 samples under malignant class among 699 samples. We will be training the model based on these features.

Data Processing

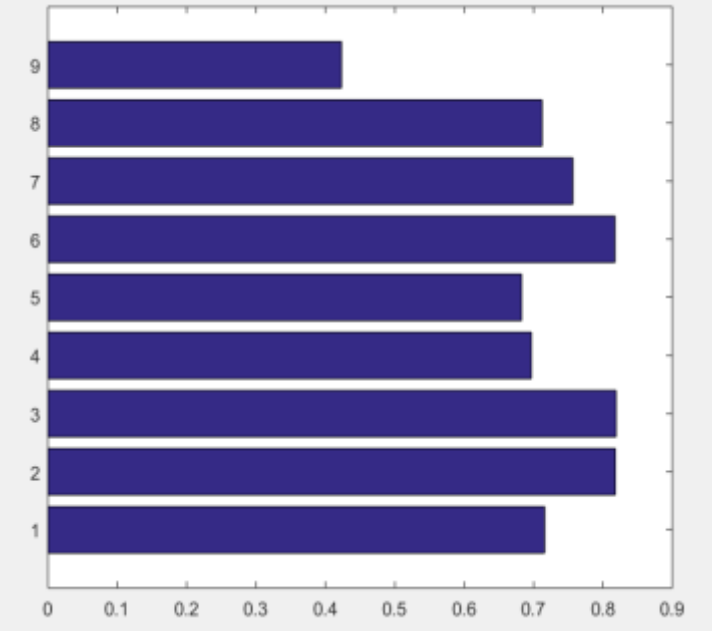
Missing Data Substitution

In the column of the feature Bare Nuclei we find that at some points '-1' is present and 16 of them in total and all the missing data values are only from this attribute.. This data has to be replaced with mean or median or mode of the remaining values of that particular feature. The mode, median and mean of the data are 1,1 and 3.54 respectively.

But rather than substituting this value of 3 for those missing values we found it to be more accurate if we substituted the mean of the particular class for that feature. On these lines we found out that the mean of Class 2 of Bare Nuclei was 1.34 and of Class 4 was 7.54 so we substituted 1 and 7 respectively for 14 missing values of class 2 and 2 missing values of class 4.

Importance of features

To understand how each feature accounts for classification of the data, we can build a plot of all the features which shows us the correlation with output.



This method helps just to figure out the importance of features which account the most for the classification in our model.
We can observe from the plot that 2,3,6 are most important while 7,1 and 8 come next in the order.

We have seen correlation and covariance matrices to see how much each feature is related to others and how much each feature is varying with others.

Correlation

By the definition of Correlation it means that more the value of correlation more similar is both the features which would give us redundant information. Also we need to know that just because a feature is leading us to a solution it may not necessarily be the causing factor.

We use correlation matrix to find the correlation value between all the features.
One of the two features with correlation value more than 0.9 which implies they are very much correlated. We found only one highly correlated feature with .907 value.

	1	2	3	4	5	6	7	8	9
1	1	0.6449	0.6546	0.4864	0.5218	0.5884	0.5584	0.5358	0.3500
2	0.6449	1	0.9069	0.7056	0.7518	0.6874	0.7557	0.7229	0.4587
3	0.6546	0.9069	1	0.6831	0.7197	0.7098	0.7359	0.7194	0.4389
4	0.4864	0.7056	0.6831	1	0.5996	0.6661	0.6667	0.6034	0.4176
5	0.5218	0.7518	0.7197	0.5996	1	0.5823	0.6161	0.6289	0.4791
6	0.5884	0.6874	0.7098	0.6661	0.5823	1	0.6762	0.5773	0.3398
7	0.5584	0.7557	0.7359	0.6667	0.6161	0.6762	1	0.6659	0.3442
8	0.5358	0.7229	0.7194	0.6034	0.6289	0.5773	0.6659	1	0.4283
9	0.3500	0.4587	0.4389	0.4176	0.4791	0.3398	0.3442	0.4283	1

So we found that 2nd and 3rd features are highly correlated. They are ‘Uniformity of cell size’ and ‘Uniformity of cell shape’. But we cannot eliminate a feature by correlation factor

Covariance matrix

Covariance matrix calculates for every combination of variables which change similarly (on a scale of 0 to any number). Then we can choose the redundant variables from there to make our model more effective and accurate.

	1	2	3	4	5	6	7	8	9
1	7.9284	5.5412	5.4777	3.9103	3.2535	5.9693	3.8341	4.6072	1.6904
2	5.5412	9.3114	8.2242	6.1478	5.0798	7.5573	5.6230	6.7357	2.4006
3	5.4777	8.2242	8.8323	5.7966	4.7359	7.6000	5.3331	6.5291	2.2372
4	3.9103	6.1478	5.7966	8.1532	3.7911	6.8522	4.6420	5.2608	2.0452
5	3.2535	5.0798	4.7359	3.7911	4.9031	4.6456	3.3265	4.2523	1.8195
6	5.9693	7.5573	7.6000	6.8522	4.6456	12.9800	5.9400	6.3517	2.0999
7	3.8341	5.6230	5.3331	4.6420	3.2265	5.9400	5.9456	4.9580	1.4393
8	4.6072	6.7357	6.5291	5.2608	4.2523	6.3517	4.9580	9.3247	2.2433
9	1.6904	2.4006	2.2372	2.0452	1.8195	2.0999	1.4393	2.2433	2.9415

One main applications that we can think of where covariance matrix is used is in Principle component analysis.

Feature Omission

In our dataset we have nine features. So for precision and accuracy, we need to use suitable combinations of features and not all of them because too many features would result in problem of overfitting model. So we would want to restrict our features to those that are most relevant to our final response which we want to predict. Also less features will take less time for computation and will have less complexity and lower computational power.

So we tried to eliminate the least important or the one which is not providing much variance to the to the total variance through PCA.

Principal Component Analysis (PCA)

PCA finds directions of maximal variance of data. The main idea of principal component analysis (PCA) is to reduce the dimensionality of a data set consisting of many variables

correlated with each other, either heavily or lightly, while retaining the variation present in the dataset, up to the maximum extent. It is a mathematical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. The number of principal components is less than or equal to the number of original variables.
We need to select number of features such that they give 99% variance of data.

Eigen Values

	1
1	48.4611
2	5.1100
3	4.2877
4	3.1134
5	2.7416
6	2.4328
7	1.7775
8	1.5944
9	0.8017

This array gives the Eigen Values with respect to Eigen vectors but not with respect to each feature, each element in this array shows us how much variance each Eigen vector has. These eigen vectors do not represent the importance of a particular feature but are linear combinations of the features. We deduce that for 98% variance we can omit one feature as only 8 principal components are needed for this variance but for 99% variance we have to include that one feature in our data set. We chose to have 99% variance so we are not omitting any of the nine features and also we don't exactly know which feature is least important through this as this tells us only about Eigen Vectors.

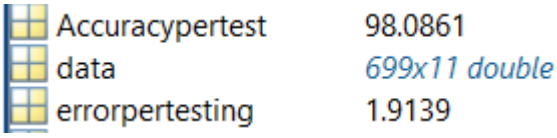
Also,
We have checked for the presence of any outliers and we found that there are no Outliers in our data set.

Classifier

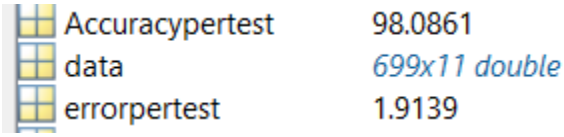
We have divided the whole data into two parts. 70% (490 samples) for training and 30% (209) for testing data among 699 samples. We trained our model to check how accurate every algorithm is. We implemented three algorithms that are suitable to our dataset. They are Naive-Bayes, K-NN and SVM algorithms. SVM algorithm provided best accuracy results compared to Naive-Bayes and KNN. KNN tends to perform very well with a lot of data points but it needs to be tuned carefully as choosing K and the distance are somewhat critical and when coming to Naive Bayes, it gives the best result when we have multi-class classification whereas SVM tends

to be good for binary classification and also when we have a limited set of points in many dimensions. As our dataset has less samples with high dimensions (attributes), we chose SVM. And amongst Linear and Non-Linear SVM we choose Linear SVM as we are able to get better accuracy with lesser error rate with repetitive 10 - fold cross validations.
We found out the accuracy for each of the above mentioned algorithms by writing an algorithm for each of the case and they came out to be:

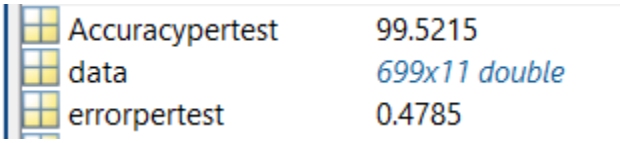
For Naive-Bayes:



For KNN:



For SVM:



And after finding out the accuracy of these algorithms, we found out the accuracy results of linear and quadratic SVM using MATLAB toolbox.

Above figure shows that the Linear SVM is providing us the best accuracy among others. So we chose Linear SVM and tune to get 100% accuracy. Here, in the figure, it is showing accuracy for Linear and Quadratic SVM, however Linear SVM also works on Quadratic Programming, the figure shows the difference between Linear and Non-Linear SVM as Quadratic SVM is a subset of Non-Linear SVM.

Support Vector Machine

Support Vector Machine is a method/algorithm we use for pattern classifications which have been successfully applied to a wide range of pattern recognition problems example being our data set. It is also a training algorithm for learning classification rules from data. SVM is most suitable for working accurately and efficiently with high dimensionality feature spaces. The standard SVM takes a set of input data and predicts, for each given input, which of two possible classes forms the input, making the SVM a non-probabilistic binary linear classifier. A simple way to build a binary classifier is to construct a hyperplane separating class members from non-members in the input space. SVM also finds a nonlinear decision function in the input space by mapping the data into a higher dimensional feature space and separating by means of a maximum margin hyperplane. The system automatically identifies a subset of informative points called support vectors and uses them to represent the separating hyperplane which is sparsely a linear combination of these points. Finally SVM solves a simple convex optimization problem.

Efficiency of our Classifier

To evaluate our classifier we have chosen to apply the 10-fold cross validation test which is a technique used in evaluating predictive models that split the original set into a training sample to train the model, and a test set to evaluate it. After applying the pre-processing and preparation methods, we try to analyze the data. To calculate the efficiency we need to know some of the terms getting us to it.

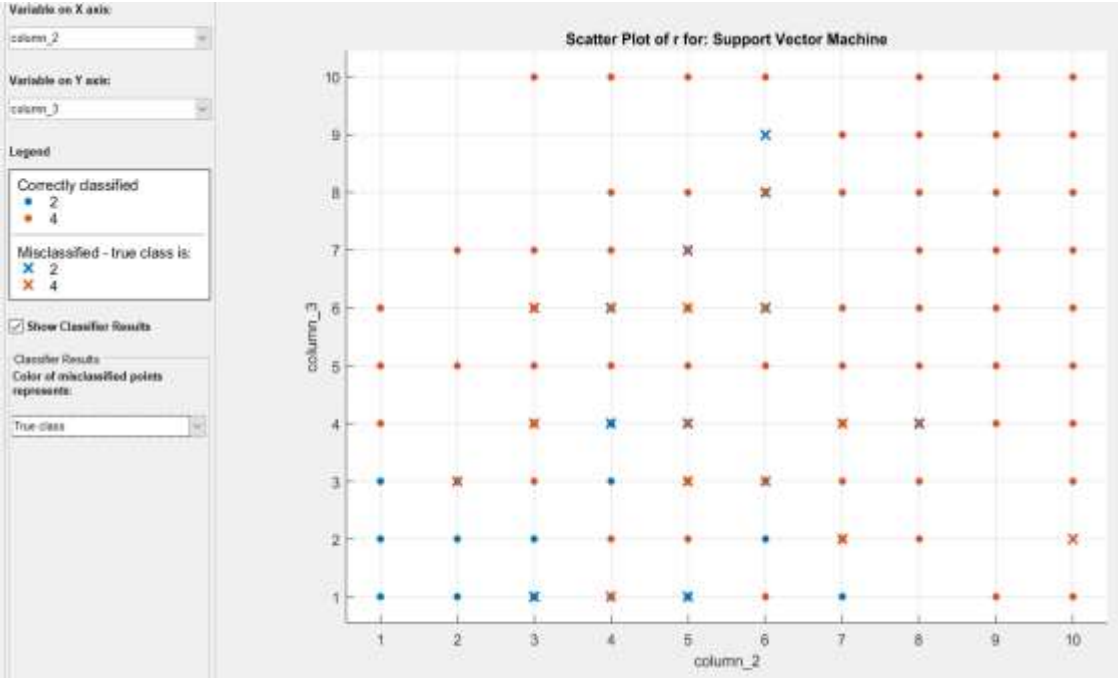
- Accuracy gives us the number of correct predictions done among the total predictions in our model. In case of our data classification it is 97% accurately classifying data.
- Misclassification error is the percentage of error that can incur in our classifier in the process of classification of the data points. In our case it is 3%.
- Sensitivity and specificity are statistical measures of the performance of a binary classification problem where Sensitivity (true positive rate) also called as recall measures the proportion of positives that are correctly identified and the Specificity (true negative rate) measures the proportion of negatives that are correctly identified in our data set.

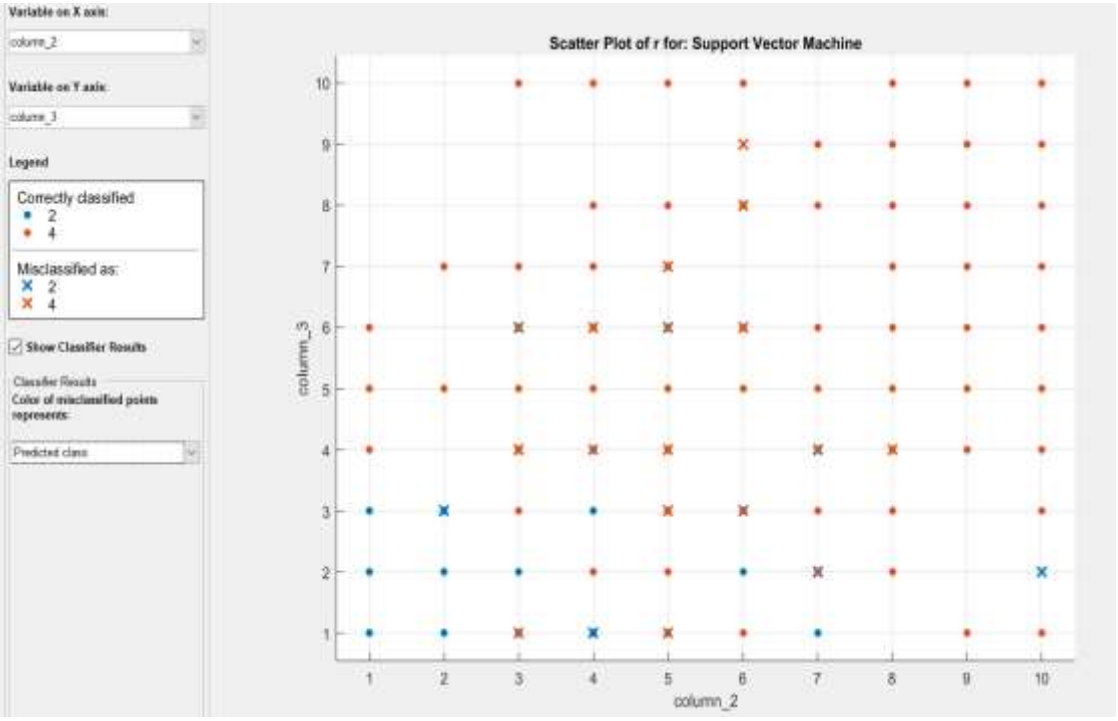
All of these terms can be calculated using the data from the confusion matrix that is drawn for SVM below the scatter plots.

Scatter Plots

One effective way to visualize the data that we are working with and find misclassifications is through scatter plot diagrams. If we were to create a scatterplot graph of 9 dimensions (one for each attribute), then we could represent all of these features.

Below are the scatter plots between ‘Uniformity of Cell Size’ and ‘Uniformity of cell shape’ features first plot representing the true class and the second plot representing the predicted class.





Confusion Matrix

It is a representative table that gives us the information about the performance of the classification model we used on the set of test data we created.

As we can see from the picture,

True Positive - 445
True Negative - 232
False Positive - 9
False Negative - 13

Calculating the terms from the confusion matrix by keeping the benign class as negative and malignant as positive, we got the values of the same to be:

Accuracy = 0.97
Precision = 0.98
Misclassification Error = 0.03
True Positive Rate (Sensitivity) = 0.97
False Negative Rate = 0.03
True Negative Rate (Specificity) = 0.96
False Positive Rate/False Alarm Rate = 0.04

we can get all these values with the help of confusion matrix but we may not be able to visualize about the cost through confusion matrix. For that we need ROC curve in which every point on that curve tells us about the classifier.

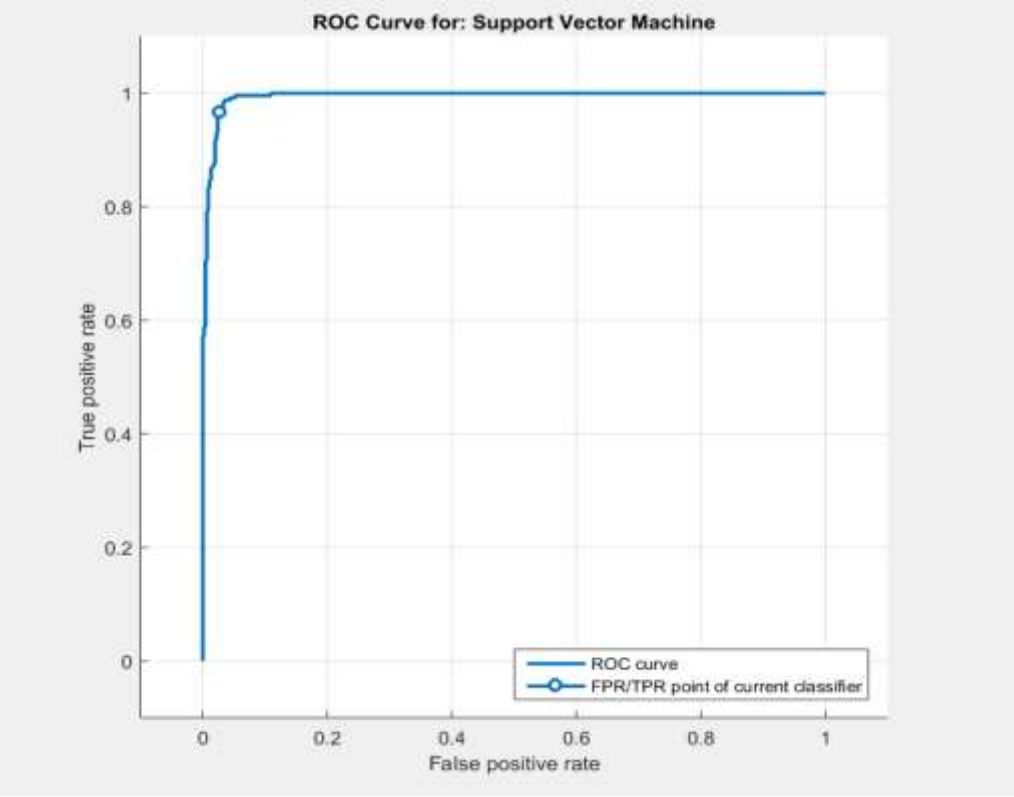
ROC Curve

To better understand efficiency, the below ROC curve of our classifier better illustrates the precision of each classifier. The ROC curve illustrates the performance of different classifiers.

Every point in the ROC curve tells us about the classifier and can gives us the cost, which is every important in our dataset as any misclassification may lead to a death of a women. So, in our problem, if false positive rate is high, that won't be a much problem because women with no cancer are predicted as positive (malignant) and can check it up by consulting a doctor but if the false negative rate is high, it will take lives of women who all are predicted negative (benign) although they have cancer.

From the plot we can easily select optimal models and discard others for best classification. It is plotted between Sensitivity and the complement of specificity. Area under the ROC curve will tell us the performance of the classifier. More the area under the curve better the classifier performance.

We have illustrated ROC curves for each of the classifiers i.e., for Naïve – Bayes, for KNN and for SVM and we got to know that SVM has more efficiency than others. And we have illustrated only for SVM as it got the better Area Under Curve (AUC).



The ROC Curve for our classifier Linear SVM is shown here.

Area under the curve is 0.99446. This is more compared to other classifiers used by us for the same data set.

Conclusion

By above observations, we can conclude that in Machine Learning, there is no specific model or an algorithm which can give 100% result or accuracy to every single dataset. We need to understand the data before we apply any algorithm and build our model depending on the desired result. The correlation method operates regardless of feature importance. The features with the highest importance were also flagged as highly correlated. So, we can say that Correlation model is not useful much in this dataset as our dataset was small with only 9 features, removing highly correlated features was not followed. By using SVM algorithm, we can reduce the chances of overfitting and the variance in the data which thus leading to better accuracy with less error rate.

Research paper LINKS:

http://ieeexplore.ieee.org/document/7818560/ ,
https://www.sciencedirect.com/science/article/pii/S2001037014000464 ,
https://arxiv.org/pdf/1711.07831
https://pdfs.semanticscholar.org/8cc7/a7c5fa82fe078cef66272184211313e7520e.pdf
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5217832/
https://link.springer.com/content/pdf/10.1007%2F0-387-34224-9_58.pdf
http://members.cbio.mines-paristech.fr/~jvert/svn/bibli/local/Liu2003Diagnosing.pdf
https://www.researchgate.net/publication/311950799_Analysis_of_the_Wisconsin_Breast_Cancer_Dataset_and_Machine_Learning_for_Breast_Cancer_Detection
http://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+%28diagnostic%29
https://www.kaggle.com/uciml/breast-cancer-wisconsin-data
https://arxiv.org/pdf/1711.07831.pdf
https://shiring.github.io/machine_learning/2017/01/15/rfe_ga_post
https://pdfs.semanticscholar.org/ab6c/4f08484db95f8950d26376dbd22c03b19b21.pdf
(for the reference of other datasets based on same author.)