

# Machine Learning

CRITICAL EVALUATION

S5125260 - 15/01/2021

SOFIA DIANA KOHAN

# Abstract

Breast Cancer is one of the most common types of cancer worldwide and with the highest mortality, according to Ak (2020). As its causes are still unknown, early detection and accurate diagnosis are essential; however, it is hard to achieve (Osareh and Shadgar 2010). Nevertheless, breast cancer mortality has decreased within the last decades due to the improvement and the application of technologies for diagnostic purposes (Vashibh 2018). In this report, whether a machine learning classifier approach will improve the detection of breast cancer is discussed as follows:

1. The dataset, which provides the data for the problem, is described and analyzed,
2. The data processing approaches, a description of the classification algorithm, its implementation, the validation method, and the evaluative performance metrics description,
3. The investigation results are presented and critically evaluated, in addition to conclusions and recommendations for further work.

# Contents

Table of figures.....	3
Table of tables.....	4
Definition and abbreviations .....	5
1.Context & Data science.....	6
2.Methodology.....	9
3.Results and Discussions .....	11
4.Critical Evaluation .....	12
5.Conclusions & further work .....	13
6.References .....	14
Appendix 1.....	16
Appendix 2.....	17
Appendix 3.....	19
Appendix 4.....	20
Appendix 5.....	21
Appendix 6.....	22
Appendix 7.....	23
Appendix 8.....	24

## Table of figures

Figure 1. Null values per attribute .....	7
Figure 2. Class imbalance .....	7
Figure 3. Manhattan Distance Correlation Matrix (full dataset) .....	11
Figure 4. Euclidean Distance Correlation Matrix (full dataset) .....	11
Figure 5. Euclidean Distance Confusion Matrix (post feature selection) .....	12
Figure 6. Boxplot for the "Mean" attributes.....	17
Figure 7. Boxplot for the "Error" attributes.....	17
Figure 8. Boxplot for the "Worst" attributes.....	18
Figure 9. Spearman Correlation Visual Representation.....	19

## Table of tables

Table 1. Overview of dataset .....	6
Table 2. Performance results full dataset.....	11
Table 3. Performance results from feature selected dataset .....	11
Table 4. Data fact sheet for all attributes.....	16
Table 5. Attributes Description .....	20
Table 6. Recursive Feature Elimination .....	21
Table 7. Dataset attributes with dimensionality reduction.....	22
Table 8. Results table from the whole dataset.....	23
Table 9. Results from dataset feature selection.....	24

## Definition and abbreviations

- [1] KNN: K-Nearest Neighbor
- [2] DT: Decision Tree
- [3] RF: Random Forest
- [4] NB: Naïve-Bayes

## 1. Context & Data science

### 1.1 The problem (Scenario)

To improve the detection of breast tumor being either benign or malign from a given dataset, throughout the use of Machine Learning technologies to classify whether a patient may or may not have cancer.

### 1.2 The data

The data set contains values about the fine needle aspirate (FNA) of breast mass results and the patient diagnosis referring to M as malignant and B as benign. The data set consists of 569 instances and 31 features, excluding the class. For diagnosing, a series of features collected throughout an FNA examination, which measures the mean, standard error and the "worst" or largest values of the radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension are required. For more details about the features refer to Appendix 1.

Table 1. Overview of dataset

Dataset	Number of classes	Number of instances	Number of features	Number of null values	Duplicate values	Total number of instances in class 1 (B)	Total number of instances in class 2 (M)
Wisconsin Breast Cancer Dataset (Diagnostic)	2	569	32	111	0	358	211

Most data types are numeric except the categorical Diagnosis feature, representing the class in the dataset. The values contained are not normalized; therefore, the data requires scaling.

### 1.3 Outliers

Each attribute should undergo an individual outlier evaluation for an appropriate analysis. As it is a 32-dimension dataset, it will not be possible to analyze each attribute and its outliers for the scope of this report. Therefore, data went through scaling for visual outliers detection with boxplots for each type of measurement (Mean, Standard Error and Worst) of the corresponding features, as shown in Appendix

2. The attributes with the most significant number of outliers are in the "Error" type features, followed by the "Worst" and finally the "Mean".

#### 1.4 Null values

The dataset also contains several null values. No matter if the mean, error or worst, the Radius feature is the feature with most missing values with an approximate of 18/19 on each respectively (as shown in Fig.1). The total amount of missing values throughout the whole dataset is of 111.

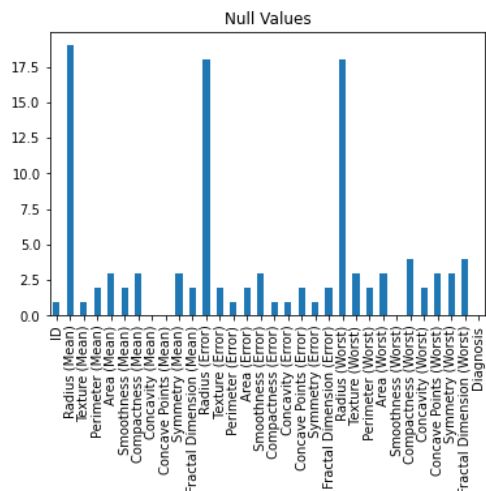


Figure 1. Null values per attribute

#### 1.5 Duplicates

The data set does not contain any duplicated values.

#### 1.6 Class Distribution

Fig.2 shows the class imbalance between diagnosis, which 358 instances are classified as benign and 211 as malignant, out of 569 instances in the dataset. The class imbalance is frequently found in real-world scenarios. Dealing with imbalance may be an issue for the validation process as it must ensure a balance between benign and malignant data instances when splitting between the train and test sets.

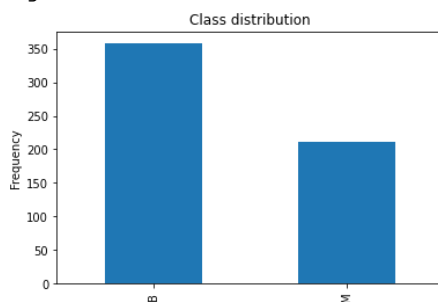


Figure 2. Class imbalance



## 1.7 Correlation Analysis (Spearman)

According to the matrix shown in Appendix 3, it is possible to identify four different levels of correlation. Firstly, it is observed perfect positive correlation between the features Radius, Perimeter and Area, for all three measurements respectively (Mean, Error and Worst), meaning that the rate of growth of these values will be equitable. Furthermore, these beforementioned attributes are also perfectly correlated between the mean and worst measurements. Secondly, it is also found a high positive correlation between the attributed Concavity, Concave Point and Compactness for all three measures respectively (Mean, Error and Worst). Furthermore, these three attributes are highly correlated between the "Worst" and "Mean" measurements and Concave Points (Mean) also highly correlates to Area, Perimeter and Radius (Mean). Thirdly, some less highly correlated values are shown between Concavity, Concave Point and Compactness between the "Worst" and "Error" values and between Perimeter, Radius and Area (Mean) with Concave Points (Worst). Finally, very few negative correlated values are shown.

## 1.8 Real World Implications & Insights

Medical sciences have been applying machine learning and artificial intelligence for diagnostic purposes to help detect diseases and process the data, leading to such diagnosis (Vollmer et al. 2019). Therefore, the application of such method in the real world must be effective and accurate. However, according to Vollmer et al. (2019), the application of such studies to real-world cases, are not as much as intended, because it seems unclear for medical practitioners the decision-making process the algorithm goes through. This gap impacts most of the research, as most of them end up useless.

For this reason, it is encouraged to understand what is guiding the algorithm to classify whether a tumor is benign or malign, and this may be achieved throughout understanding the values which represent specific attributes. Appendix 4. describes the values that are determinants for diagnosing.

Another critical aspect that will impact the use of the model for medical purposes are the implications of false-positive (F/P) and false-negatives (F/N) in the real world. These values represent which instances have been incorrectly classified as benign or vice versa. F/N and F/P are significant as, if there is a high number of false positives or false negatives, the classification will not be as accurate as it should be and will not provide any insight for the doctors.

## 2. Methodology

### 2.1 Aims and objectives

The investigation aims to test whether a machine learning approach for diagnostic breast cancer data helps classify whether cancer is benign or malign, and compare its performance against similar practices. Another aim is to determine whether the model will provide more insights for medical practitioners when making related decisions.

### 2.2 Data processing

Before proceeding with the predictive model, an essential step is the cleaning and the processing of data. Firstly, the "ID" feature in the dataset is dropped as it is not relevant for the classification purposes. Secondly, the class attribute "Diagnosis" is an object, however, for the KNN to be able to classify, the values where B = 0 (benign) and M = 1 (malign) given.

#### 2.2.1 Dealing with null values & duplicates

No action regarding duplicated instances needed as none are encountered. Nevertheless, we encounter 111 missing values throughout the whole dataset. As missing values correspond to different features (as shown in fig 1), new values were imputed with the mean, as we are dealing with numerical values. After imputing the values, once again the dataset was checked for duplicated values and none were found.

### 2.3 Classification Algorithm

The classification algorithm chosen for this problem was a K-Nearest Neighbor. It is a supervised and instance-based algorithm. It will find a predefined number of training samples and measure the distance to the new point, predicting the new label (Pedragosa et al. 2011). The number of samples is determined by a constant K, and for the sake of this report, it was user-defined. There are several distance metrics to implement for this algorithm, which are described in the configuration/tuning section. The method will output a classification label for an unknown instance from a test set. Furthermore, K-NN is suitable for numerical and small datasets. However, its performance may be affected by a high number of features. Therefore, the experiment will test both the performance in the data set as a whole and later through a featured selected dataset and will be compared.

## 2.4 Configuration/ Tuning

The model requires some specific parameters for it to perform accordingly. The parameters usually involved with this model are the K value and a distance metric. The K value was kept the default one, which is 5. The distance metric selected is the Minkowski Distance. This metric has its own parameter to tune, known as p. Depending on the value of p, the distance calculation that will be performed. In the case of  $p=1$ , the distance metric calculation is the Manhattan Distance. If  $p=2$  the calculation performed is the Euclidian Distance. Other values can be tested; yet the first two were selected.

## 2.5 Feature Selection

The correlation analysis and the implementation of a recursive feature elimination provide insight into which are the most descriptive features when predicting and determining whether the selected features will improve the model's performance throughout a dimensionality reduction. The recursive feature elimination was performed ten times. Per each time, a note of the attribute and its rank was taken. Appendix 5 shows per attribute how many times it appeared as highly descriptive. Based on these results and the correlation matrix analysis, a smaller data set with a reduced number of features has been created and will also be tested. The feature selected dataset is composed of 9 attributes shown in Appendix 6.

## 2.6 Validation Method

To test whether the KNN is working correctly, it should undergo a validation method. The one selected was the hold-out validation, splitting the dataset into test and training sets for all the features and the classifier attribute (Allibhai 2018). The training set will hold 60% of data for training and 40% for testing. Furthermore, the dataset will undergo a random shuffling before splitting the data. As mentioned in the data analysis section, the classes are not equally distributed, so the dataset has been stratified, meaning that the splits will contain a balanced number of samples from each class (Pedragosa et al. 2011).

## 2.7 Evaluation metrics

Evaluation metrics is a method to determine how effective the model is. There are various methods; the evaluation metrics implemented are accuracy, precision, recall, and a confusion matrix. As the validation method is on a loop, there will be various values for each metric. Therefore, the values presented will be an average of all the values per metric. However, the evaluation metrics used for critically

evaluating the models' performance are more detailed, providing us with an f1-score.

### 3. Results and Discussions

The table below presents the results after executing the algorithm in the full dataset. These are not the expected results in contrast to the ones presented by Vashibh (2018). The model is not classifying in such a way that will provide any help or insight for the doctors when performing a diagnosis.

Table 2. Performance results full dataset

K-NN	Accuracy	Precision	Recall
Euclidean Distances	0.54 (+/- 0.24)	0.27 (+/- 0.12)	0.50 (+/- 0.00)
Manhattan Distance	0.54 (+/- 0.25)	0.27 (+/- 0.13)	0.50 (+/- 0.00)

Furthermore, the confusion matrices represent the correctly and incorrectly classified instances by the model for the full dataset:

0	143
0	85

Figure 4. Euclidean Distance  
Correlation Matrix (full dataset)

143	0
85	0

Figure 3. Manhattan Distance  
Correlation Matrix (full dataset)

The results after executing the algorithm in the feature selected dataset is shown below. Unfortunately, the results are not as expected and even poorer than the ones from the full dataset.

Table 3. Performance results from feature selected dataset

K-NN	Accuracy	Precision	Recall
Euclidean Distances	0.54 (+/- 0.24)	0.27 (+/- 0.12)	0.50 (+/- 0.00)
Manhattan Distance	0.58 (+/- 0.20)	0.29 (+/- 0.10)	0.50 (+/- 0.00)

The following confusion matrices represent the correctly and incorrectly classified instances by the model for the feature selected dataset:

143	0
85	0

Figure 5. Euclidean Distance Confusion Matrix (post feature selection)

0	143
0	85

Figure 6. Manhattan Distance Confusion Matrix (post feature selection)

As the results turned out to be poorer from the ones expected, it would lead to think that a classification algorithm is not appropriate for such problem. Nonetheless, this may not be the case as many other studies have been carried out with such dataset and have been successful achieving accuracies of 0.90% and higher (Kinnerkar and Potdar 2016). Furthermore, we can clearly see from the confusion matrices, how the algorithm is not performing an appropriate classification. This may mean that there might be some issue that has not been encountered on time from the code developed for the classifier. Moreover, the model takes too long to predict the labels, corroborating there is likely to be an issue with the code implementation. For this reason, performing a critical evaluation is crucial.

#### 4. Critical Evaluation

An exemplary implementation of a Machine Learning algorithm might help diagnose, in this case, cancer, if adequately implemented. However, as shown in the results section, this was not achieved by implementing the KNN by scratch. Therefore, other classification methods such as Naïve Bayes(NB), Decision Tre, Random Forest and KNN were implemented for further evaluation. The classifiers are predefined by Scikit Learn, and different models of the different classifiers where compared. The validation method implemented for all the classifiers in the full data set and the selected features dataset is the repeated stratified k-fold cross-validation. This strategy was selected as it ensures a balance in the class frequencies in the train and test folds, handling the class imbalance beforementioned in the data analysis (Pedragosa et al. 2011). When performing the cross-validation, the dataset was split into six folds, and the validation was repeated twice. The performance metrics were kept the same as described in the evaluative metrics section. In addition, as a classification report was implemented, the f1-score is also used as metric and the 'macro' metrics were used to interpret the results.

The KNN implemented was kept with the same distance metric and same value for k. A grid search was carried out for the DT classifiers to determine the best possible candidate parameters for this

implementation. Five different parameters have been tuned, based on the grid search results (including a maximum number of features which will not be determined when performing the DT in the feature selected dataset). Nevertheless, the table of results shows just two of the four decision trees. Two RF were also implemented with different criterion parameter tuning for more insight into the implication it may have for this kind of problem. Finally, three types of Naïve Bayes classifiers were implemented. The first one is a GaussianNB, followed by a MultinomialNB and finally BernoulliNB.

Appendix 7 shows the results for the full dataset, while Appendix 8 contains the results for the selected features dataset. There is a clear difference between the K-Nearest Neighbor from Scikit Learn and the one implemented from scratch in both types of datasets. The evaluation metrics from the Scikit Learn KNN represent more relevant results for the problem to be solved. The results presented are more similar to those presented by Kinnerkar and Potdar (2016). According to these further results, it is proved that an appropriate implementation of a machine learning approach could help the medical diagnostic industry.

## 5. Conclusions & further work

To conclude, the KNN implemented from scratch does not provide doctors' help or insight when performing diagnosis. However, according to various research papers and further experiments using Scikit Learn predefined classifiers it is possible to affirm that a good implementation of a classifier that is suitable for problem may provide the needed insights. It also seems like the selected features do not tend to improve the model's performance, but in this case, worsen it. Further investigation on why the algorithm implemented from scratch does not classify correctly is highly recommended as well as understanding why the selected features have not been as helpful as desired. Carrying out experiments with other features is also suggested. Furthermore, a more in-depth analysis of the outliers and anomalies shown in the dataset would be highly recommended. It may provide more insight into the problem and how to apply an appropriate solution.

## 6. References

- [1] Ak M. F., (2020). A Comparative Analysis of Breast Cancer Detection and Diagnosis Using Data Visualization and Machine Learning Applications. *Healthcare (Basel, Switzerland)* [online], 8(2), 111. Available from: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7349542/> [Accessed 13 October 2020]
- [2] Allibhai, E., 2018. *Holdout vs. Cross-validation in Machine Learning* [online]. Medium. Available from: <https://medium.com/@eijaz/holdout-vs-cross-validation-in-machine-learning-7637112d3f8f> [Accessed 5 Dec 2020].
- [3] Doddipalli, L., Devi, D. and Visvavicyalayam, S., 2015. Importance of Feature Extraction for Classification of Breast Cancer - A Study. *International Journal of Scientific and Innovative Mathematical Research (IJSIMR)* [online], 3(2), PP 763-768.
- [4] Dua, D. and Graff, C., 2019. *UCI Machine Learning Repository* [online]. Irvine, CA: University of California, School of Information and Computer Science. Available from: <http://archive.ics.uci.edu/ml> [Accessed 9 October 2020].
- [5] Kinnerkar, R. and Potdar, K., 2016. A Comparative Study of Machine Learning Algorithms applied to Predictive Breast Cancer Data. *International Journal of Science and Research (IJSR)* [online], 5(9), ART20161994.
- [6] Osareh, A. and Shadgar, B., 2010. "Machine learning techniques to diagnose breast cancer," *2010 5th International Symposium on Health Informatics and Bioinformatics* [online], pp. 114-120, doi: 10.1109/HIBIT.2010.5478895.
- [7] Pedregosa, F., et al., 2011. *3.1. Cross-validation: evaluating estimator performance - scikit-learn 0.24.1 documentation* [online]. Scikit-learn.org. Available from: [https://scikit-learn.org/stable/modules/cross\\_validation.html#cross-validation](https://scikit-learn.org/stable/modules/cross_validation.html#cross-validation) [Accessed 17 Nov 2020].
- [8] Pedregosa, F., et al., 2011. *1.6. Nearest Neighbors - scikit-learn 0.24.1 documentation* [online]. Scikit-learn.org. Available from:

<https://scikit-learn.org/stable/modules/neighbors.html#classification>  
[Accessed 12 Nov 2020].

[9] Pedragosa, F., et al., 2011. *sklearn.model\_selection.train\_test\_split – scikit-learn 0.24.1 documentation* [online]. Scikit-learn.org. Available from:  
[https://scikit-learn.org/stable/modules/generated/sklearn.model\\_selection.train\\_test\\_split.html](https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html) [Accessed 26 Nov 2020].

[10] Wolberg, H.W., Street, N., and Mangasarian O. L., 1995. *UCI Machine Learning Repository*. UCI Machine Learning Repository: Breast Cancer Wisconsin (Diagnostic) Data Set. [online] Available from:  
[https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+\(Diagnostic\)](https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)) [Accessed 8 October 2020].

[11] Vashibh, G., 2018. *Building a Simple Machine Learning Model on Breast Cancer Data* [online]. Towards Data Science. Available from:  
<https://towardsdatascience.com/building-a-simple-machine-learning-model-on-breast-cancer-data-eca4b3b99fa3> [Accessed 17 Nov 2020].

[12] Vollmer, S., et al., 2019. Machine learning and artificial intelligence research for patient benefit: 20 critical questions on transparency, replicability, ethics, and effectiveness. *theBMJ* [online]. BMJ 2020;368:16927. Available from:  
<https://www.bmj.com/content/bmj/368/bmj.16927.full.pdf> [Accessed 20 October 2020]



## Appendix 1

Table 4. Data fact sheet for all attributes

(Mean)	Data Type	Min value	Max value	Mean	Standard Deviation
Radius	float64	6.981000	28.110000	14.123118	3.528840
Texture	float64	9.710000	39.280000	19.280986	4.299856
Perimeter	float64	43.790000	188.500000	92.010459	24.322495
Area	float64	143.500000	2501.000000	654.704064	352.102951
Smoothness	float64	0.052630	0.163400	0.096376	0.014074
Compactness	float64	0.019380	0.345400	0.104220	0.052820
Concavity	float64	0.000000	0.426800	0.088799	0.079720
Concave points	float64	0.000000	0.201200	0.048919	0.038803
Symmetry	float64	0.106000	0.304000	0.181032	0.027410
Fractal Dimension	float64	0.049960	0.097440	0.062799	0.007073

(Error)	Data Type	Min value	Max value	Mean	Standard Deviation
Radius	float64	0.111500	2.873000	0.403911	0.277612
Texture	float64	0.360200	4.885000	1.214336	0.550986
Perimeter	float64	0.757000	21.980000	2.863195	2.022481
Area	float64	6.802000	542.200000	40.237915	45.491767
Smoothness	float64	0.001713	0.031130	0.007034	0.003009
Compactness	float64	0.002252	0.135400	0.025468	0.017922
Concavity	float64	0.000000	0.396000	0.031895	0.030213
Concave points	float64	0.000000	0.052790	0.011783	0.006175
Symmetry	float64	0.007882	0.078950	0.020535	0.008272
Fractal Dimension	float64	0.000895	0.029840	0.003795	0.002650

(Worst)	Data Type	Min value	Max value	Mean	Standard Deviation
Radius	float64	7.930000	36.040000	16.275134	4.847944
Texture	float64	12.020000	49.540000	25.629170	6.115881
Perimeter	float64	50.410000	251.200000	107.235326	33.609513
Area	float64	185.200000	4254.000000	879.487986	569.722315
Smoothness	float64	0.071170	0.222600	0.132369	0.022832
Compactness	float64	0.027290	1.058000	0.254019	0.157306
Concavity	float64	0.000000	1.252000	0.272220	0.208649
Concave points	float64	0.000000	0.291000	0.114233	0.065489
Symmetry	float64	0.156500	0.663800	0.289828	0.061686
Fractal Dimension	float64	0.055040	0.207500	0.083974	0.018101

## Appendix 2

Figure 6. Boxplot for the "Mean" attributes

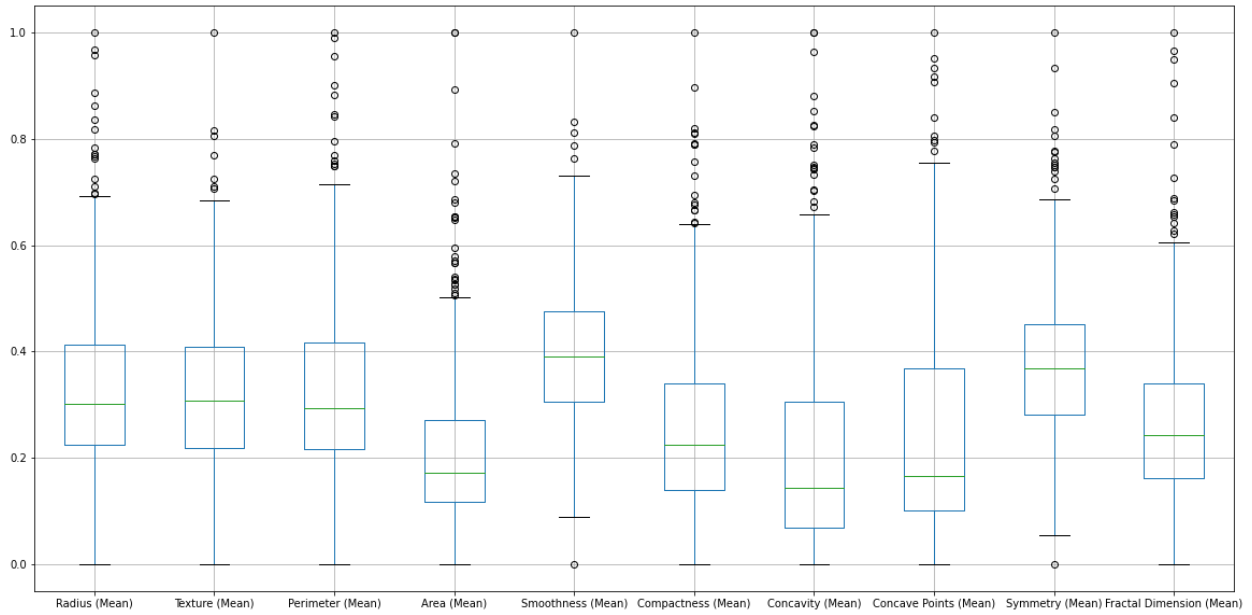


Figure 7. Boxplot for the "Error" attributes

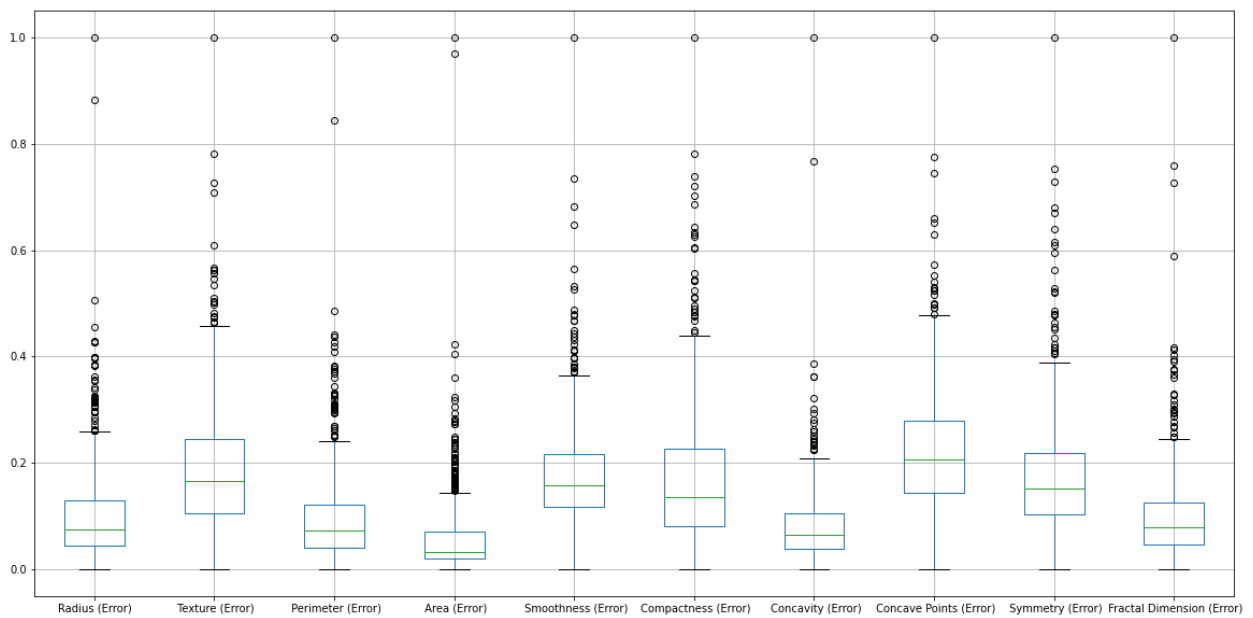
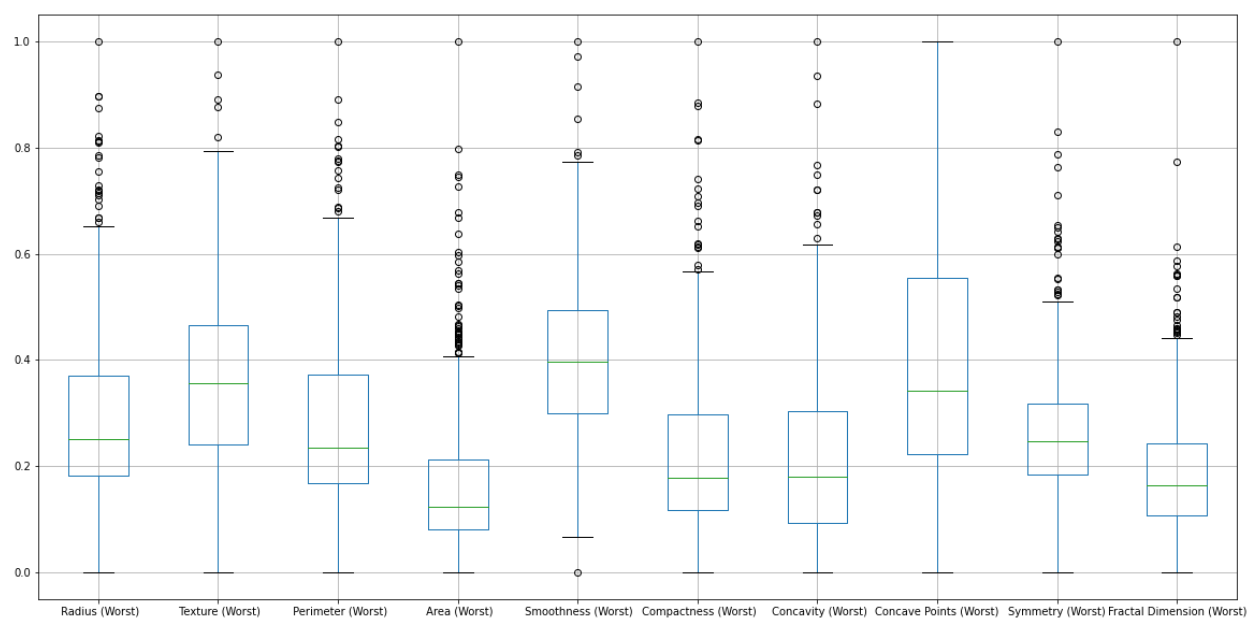
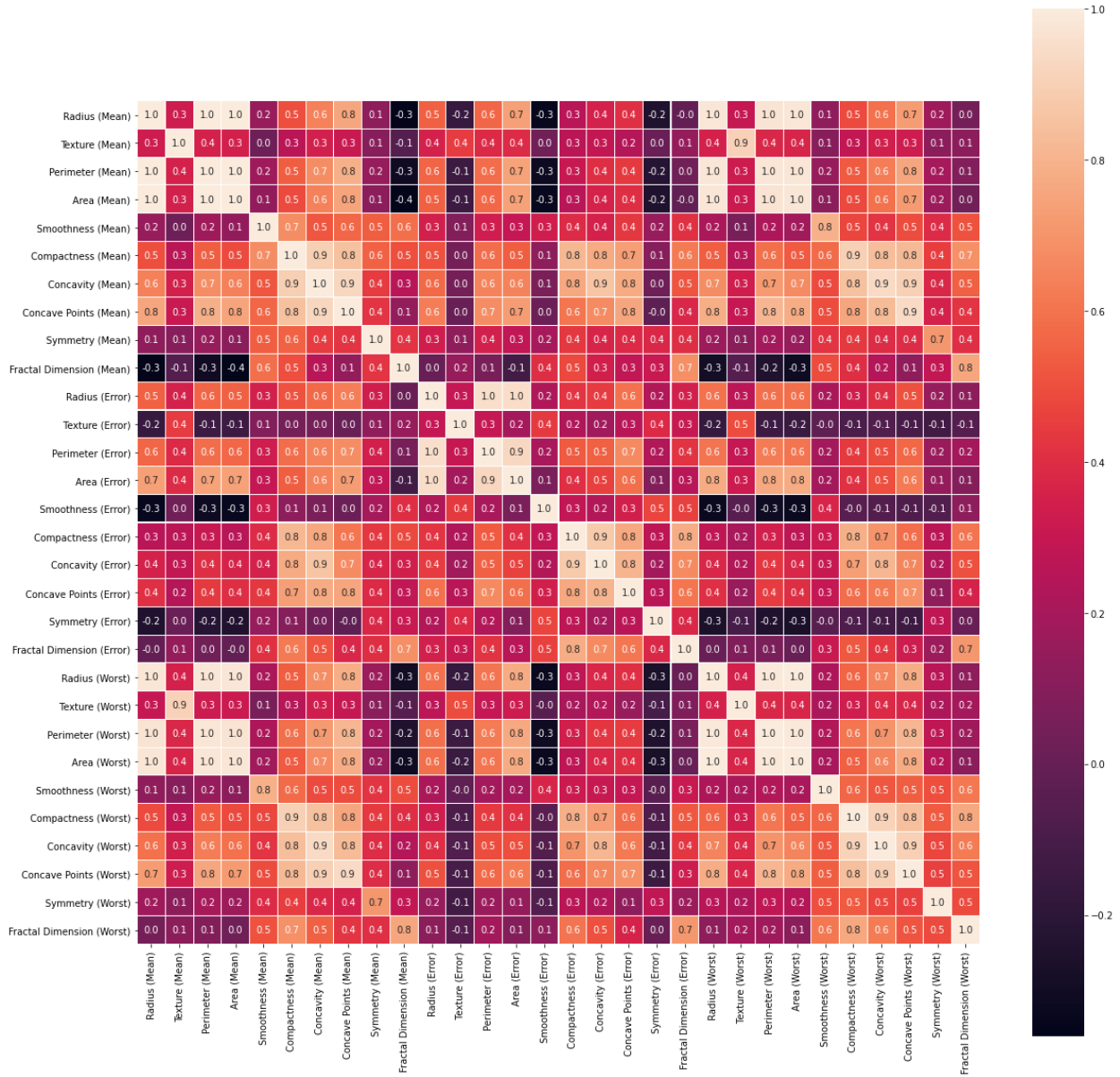


Figure 8. Boxplot for the "Worst" attributes



## Appendix 3

Figure 9. Spearman Correlation Visual Representation



## Appendix 4

Table 5. Attributes Description

Attribute	Description
Radius (Mean/Worst)	The mean value measures the range between the center and the perimeter. The Worst value is the highest value of the center of the tumor for the estimated range. This value will determine whether surgery is required or not.
Area	Values related to the mean of cancer cell areas (related to perimeter)
Texture	Represents gray-scale values which help locate the tumor
Perimeter (Mean/Worst)	Mean and worst values of the core tumor.
Smoothness	Rational variation in radius range
Compactness (Mean/Worst)	Is the mean value of estimation of the perimeter and area. The worst value contains the highest estimates value.
Concavity	Severity of concave portions of the shape
Concave Points (Mean)	Number of concave points of the coutour
Fractal Dimension (Mean/Worst)	Mean and worst values calculated for coastline approximation

## Appendix 5

Table 6. Recursive Feature Elimination

Feature	Total number of occurrences
Area (worst)	10
Texture (worst)	9
Concave Points (worst)	10
Texture (mean)	9
Concavity (mean)	6
Concavity (worst)	1
Perimeter (worst)	3
Smoothness (worst)	1

## Appendix 6

Table 7. Dataset attributes with dimensionality reduction

Feature	Type
Texture (mean)	float64
Area (mean)	float64
Compactness (mean)	float64
Concavity (mean)	float64
Perimeter (mean)	float64
Area (worst)	float64
Smoothness (worst)	float64
Concave Point (worst)	float64
Diagnosis	int (0-1)

## Appendix 7

Table 8. Results table from the whole dataset

'macro' w/o feature selection	DT (Rank 1)	DT (Rank 4)	RF (Model 1) * <sup>1</sup>	RF (Model 2) * <sup>2</sup>	NB (Model 1)	NB (Model 2)	NB (Model 3)	KNN (Model 1)	KNN (model 2)
Accuracy (Training)	1.0 (+/- 0.0)	1.0 (+/- 0.0)	1.0 (+/- 0.0)	1.0 (+/- 0.0)	0.94 (+/- 0.02)	0.84 (+/- 0.01)	0.63 (+/- 0.0)	0.98 (+/- 0.01)	1.0 (+/- 0.0)
Accuracy (Testing)	0.94 (+/- 0.04)	0.92 (+/- 0.04)	0.96 (+/- 0.05)	0.96 (+/- 0.02)	0.93 (+/- 0.04)	0.83 (+/- 0.05)	0.62 (+/- 0.0)	0.97 (+/- 0.02)	0.97 (+/- 0.02)
Precision	0.94	0.92	0.97	0.97	0.93	0.89	0.42	0.96	0.97
Recall	0.94	0.92	0.96	0.96	0.92	0.78	0.49	0.96	0.96
F1 - score	0.94	0.92	0.96	0.96	0.92	0.80	0.39	0.96	0.97
T/P	341	336	350	351	341	357	351	351	355
T/N	196	191	200	200	287	118	2	197	196
F/P	17	22	8	8	17	1	7	7	3
F/N	15	20	11	11	24	93	209	14	15

\*<sup>1</sup> Criterion = 'entropy'

\*<sup>2</sup> Criterion = 'gini'



## Appendix 8

Table 9. Results from dataset feature selection

'macro' With feature selection	DT (Rank 1)	DT (Rank 4)	RF (Model 1) * <sup>1</sup>	RF (Model 2) * <sup>2</sup>	NB (Model 1)	NB (Model 2)	NB (Model 3)	KNN (Model 1)	KNN (model 2)
Accuracy (Training)	1.0 (+/- 0.0)	1.0 (+/- 0.0)	0.99 (+/- 0.0)	1.0 (+/- 0.0)	0.92 (+/- 0.01)	0.63 (+/- 0.00)	0.63 (+/- 0.00)	0.95 (+/-0.01)	1.0 (+/-) 0.0
Accuracy (Testing)	0.91 (+/- 0.06)	0.93 (+/- 0.05)	0.94 (+/- 0.04)	0.94 (+/- 0.05)	0.92 (+/- 0.04)	0.63 (+/- 0.01)	0.62 (+/- 0.1)	0.93 (+/- 0.05)	0.94 (+/- 0.05)
Precision	0.91	0.92	0.94	0.94	0.92	0.31	0.31	0.93	0.94
Recall	0.91	0.92	0.93	0.94	0.91	0.50	0.50	0.93	0.94
F1 - score	0.91	0.92	0.94	0.94	0.91	0.39	0.38	0.93	0.94
T/P	335	338	347	344	340	358	356	339	346
T/N	188	188	190	102	184	0	0	191	191
F/P	23	20	11	14	18	0	2	19	12
F/N	23	23	21	19	27	211	211	20	20

\*<sup>1</sup> Criterion = 'entropy'

\*<sup>2</sup> Criterion = 'gini'