Breast Cancer diagnosis using machine learning classification methods (using Hadoop)

Article · August 2019

CITATIONS
0

READS
1,210

2 authors:

Nestor Pereira
Technological University Dublin - City Campus
3 PUBLICATIONS **1** CITATION
SEE PROFILE

Symeon Charalabides
National College of Ireland
4 PUBLICATIONS **1** CITATION
SEE PROFILE

Breast Cancer diagnosis using machine learning classification methods

PEREIRA LINARES, Nestor
HDSDA in the National College of Ireland
x01525041@student.ncirl.ie

Tutor: Symeon Charalabides
Associate Faculty Lecturer
National College of Ireland
Symeon.Charalabides@ncirl.ie

Abstract

Breast cancer is one of the most common types of cancer in Ireland and worldwide. Any effort that helps to obtain an early diagnosis or preventing any cancer cell growth is helpful. This idea inspires this project. Using data from the Breast Cancer Wisconsin's Data Set (UCI Machine Learning), we use machine learning techniques to predict the existence of any cancer cells. New technologies such as data storage using the Hadoop system on AWS (Amazon), clustering and several linear and non-linear prediction methods are used to diagnose the condition of the cell (and the patient). The different results are compared based on accuracy performance, confusion matrix and area under the Receiver Operating Characteristics (ROC) curve. A classification error means sending a patient home who could potentially have cancer. Therefore, minimizing classification errors is vital in this approach.

The goal of this project is to find one or more methods to solve the problem. The best models are chosen using performance metrics such as the area under the ROC curve and the area under the Precision-Recall curve and prediction accuracy.

Keywords *— Classification, Machine Learning, Logistic regression, Naïve Bayes, Random Forests, k-NN, Support Vector Machine, Hadoop, AWS Cluster, Accuracy, Area under the ROC curve, Area under Precision-Recall curve.*

I. INTRODUCTION

Breast cancer is the most common cancer in women in Ireland, after skin cancer, according to the Irish Cancer Society. In fact, 1 in 10 women in Ireland will get breast cancer at some stage in their lives. Therefore, any study or effort to prevent breast cancer does no matter how little is, could be the difference.[6]

It will be worked with a dataset from the repository database of the UCI machine learning repository to the University of California based on the work of Dr W. Wolberg, Dr N. Street, Dr O. Mangasarian. [1] [3][4]

In the paper "Computerized Breast Cancer Diagnosis and Prognosis from fine needle aspirates" describe how the benign and malignant cell samples were obtained from a biopsy procedure called FNA over 569 patients. [3][9]

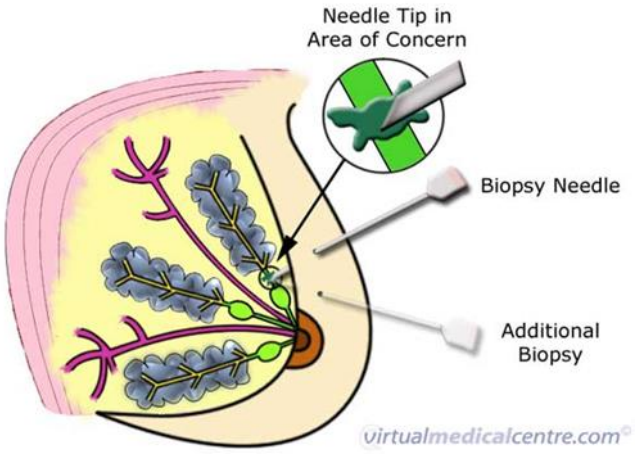

Fig. 1. Biopsy procedure called FN (Image courtesy of www.myvmc.com)

The present approach uses the data produced for this previous research to apply different machine learning techniques to predict benign and malignant cells. It will be analysing the dataset as a binary classification problem.[12][13]

Using coherent comparative methods, means that apply the same pre-processing techniques for the same dataset, it compares different algorithms for classification.

It applies re-sampling for the training dataset to fix the model and look for the best accuracy. Later, it predicts from a validation dataset and compares the results with the area under the ROC curve and the area under the precision-recall relation.

II. PROBLEM DEFINITION AND PROJECT SCOPE

The principal objective in this project is simple, classify a patient in the groups with diagnosis benign or malign (Binary classification problem) according to values of 32 features provided by the dataset created by the University of Wisconsin which has 569 instances (rows-samples).

Wrong diagnose, it means to classify some patient in a group of benign tumours when it is not (has cancer) indicate that this diagnosis is misclassified, false negative, would have a considerable impact on the patient and therefore, the method has not trustworthy for prediction.

Machine Learning in the healthcare sector is a useful help, but it has a significant impact on the healthcare subject understudy for this reason.

In this project, it applies additional metrics of performance like the area of the curve under Receiver Operating Characteristic ROC (sensitivity/specificity) and area of the curve under Precision-Recall relation in order to compare prediction models and chose the best performing method, or more than one. [7] [15] [16]

Additionally, the healthcare sector produces hundreds of data in any subjects related to disease studies or diagnosis analysis. Fortunately, nowadays, the improvement in technology allow tackling the problem of storage and processing a high volume of data. This project is using technology like Hadoop under cluster machine and Python to guarantee deal with any vast datasets.

III. OBJECTIVES

That was mention before, the principal aim in this project is simple, classify a patient in the groups with diagnosis benign or malign (Binary classification problem.

Due the data of cancer diagnosis is high sensibility, sending home a sick patient means the patient was misclassified with a severe consequence, the classification must be evaluated with a very high accuracy, very high precision, very high recall, and be assessed use different algorithms to find the best solution (or solutions) to solve the problem.

Consequently, it can be clear several additional objectives which have to be achieved to obtain a "simple" classification.

Those objectives are:

- Identify the best techniques which fix better for the problem between linear and non-linear classification algorithms.
- Evaluate all these techniques properly in order to guarantee high accuracy with low variance and low bias of the error. The challenge in the method is keeping the bias-variance trade-off in balance.
- Apply a coherent comparative technique to evaluate the performance of the algorithms applied to the problem.
- Apply the top big data technologies used to store and analyses data: Python over Hadoop Cluster, applying in different Machine Learning techniques.

IV. STATE OF THE ART - BACKGROUND

The project follows four (4) principle issues to achieve the prediction objective describe in the previous chapter.

A. *Datamining profitability in healthcare*

Nowadays, it is common to find many data mining techniques using in different industries to obtain the best customer profitability. However, looking for the profitability in term of healthcare also could provide profitability in different dimensions: help to fast diagnosis, avoid unnecessary treatment, help to reduce the diagnosis error, help to reduce the cost for the patient and to the health system. [5][12][30][31]

B. *Optimisation and performance*

Optimise the model using resampling techniques like k-folds cross-validation to fix the model from the training dataset and reduce, or measure, the variance. This technique allows estimating how well the algorithms will perform on new data. The predictions making from a validation dataset can be measured in term of performance looking at the typical performance metrics: accuracy, confusion matrix, precision, recall and the area under the ROC curve. Furthermore, measure the area under the curve for the precision-recall relation provide additional valuable information. [12] [15] [16] [28]

C. *Trustworthy model for prediction*

Sensible information and hugely negative impact on the wrong diagnosis means that need to apply and compare several classification algorithms, and complementary techniques to support the prediction and reduce the variance and the bias (bias-variance trade-off), and ideally minimize both. Therefore, find one or more than one model, complementaries between then, is useful to find a complete and trustworthy solution. [5] [12] [14] [29] [32]

D. *Big data in the healthcare sector*

Nowadays, healthcare sector produces a massive amount of data relevant for the diagnosis propose. This data about a patient, medical records, disease, images, prescriptions, results, etc. need to be store and manage appropriately. This data does not always structure that why need technology to support structure and non-structure data, for example, Hadoop system. [2][5] [11] [30] [31]

V. TECHNICAL APPROACH

In this section, it is boarded the different algorithms used, assumptions, and the benefits of them. In order to make a coherent comparison among them, it applies the same method structured in three (3) steps:

- Fix the model from the training dataset and measure the accuracy.
- Apply k-fold cross-validation to train the model to reduce, or measure, the variance in the future predictions.
- Apply classification performance metrics to compare a test dataset vs prediction, an area under the ROC curve, confusion matrix, and calculate precision, recall and F1-score.

Always using the same splitter dataset come from Hadoop Dataset File System HDFS. This dataset is splitter in 66% for training and 33% for test using the same random seed (77).

Later, it applies k-fold cross-validation with the training dataset to compare all fitted models, using accuracy and chose the two best options.

This technique, cross-validation, allow reducing the variance when the fitted model applies to validation data. [29] [32]

Finally, those two best options are compared with two metrics: area under the ROC curve and area under precision-recall to compare, visually and numerical, the performance into the predictions.

A. *Logistic Regression*

It is a probabilistic method with high performance in binary problem classification under the assumption that there is a linear relationship between the predictor's features and the dependent variable. The limitation in this method assumes that there is no correlation between the independent variables (no multicollinearity).

B. *Naïve Bayes*

It is one of the most popular techniques using in binary classification despite the fact the assumptions of independence between the predictors. In reality, this assumption is at least challenging to prove. It is easy to understand, easy to use and faster.

C. *k-Nearest Neighbours*

The advantage of this method is non-assumption about the distribution of the variable. This aspect is an important point to compare with the two previous methods.

This method needs to find the best value of k, numbers of neighbour, to optimise the classification and approach the bias-variance trade-off.

The optimum values of k allow keeping the bias-variance trade-off in balance and ideally minimize both.

D. *Random Forests*

Decision tree classification method is another popular classification method because usually perform exceptionally well. Additionally, it is easy to understand because the method is close to human decision making. However, this method has a high risk of over-fitting, and usually, other methods are better in term of accuracy.

In this point, the Random Forests algorithm as an extension of bagged decision trees is a powerful alternative, produce multiple trees to improve prediction accuracy and reduce the risk of over-fitting. [29]

E. *Support Vector Machine (SVM)*

Support Vector Machine can use a linear and non-linear function to define the boundaries among the class.

It is useful in binary problem classification because, in case, the observations are not linearity separable, it is possible to use polynomial or radial basis function (non-linear) to find the boundaries to separate the observations.

This project uses SVM linear function and SVM non-linear function, called polynomial kernel and radial basis function kernel, to classify observations.

F. *Algorithms comparison*

After all previous model are fitted to compare and find the trustworthy model, or combination of models, to binary prediction, it is vital to evaluate how well the solution is.

Firstly, use the re-sampling techniques like k-fold cross-validation, all the models are retraining with k=10 and compare the accuracy results.

Choosing the two best options, or more depending on the accuracy results, it applies the models to make a prediction and compare with the test dataset.

The principal metrics to allow this comparison are:

- Accuracy: it is a proportion of correctly classified observations.
- Confusion matrix: it shows a table with values of true positive (TP), true negative (TN), false-negative (FN) and false positive (FP) values.

		True condition	
	Total population	Condition positive	Condition negative
Predicted condition	Predicted condition positive	**True positive**	**False positive**, Type I error
	Predicted condition negative	**False negative**, Type II error	**True negative**

Fig. 2. Confusion matrix [33] Wikipedia

- Precision: it is TP / (TP+FP) how often the positive prediction is right.
- Sensitivity (Recall): it is TP / (TP+FN) how often the positive real value is predicted correctly.
- Specificity: it is TN / (TN+FP) how often the negative real value is predicted correctly.
- The area under the ROC curve (AUC): it visualises of true positive rate (Sensitivity) and false-positive rate or false alarm (1 - Specificity).
- The area under the precision-recall curve: it less frequently used and visualise the curve under relation precision-recall.

VI. TECHNICAL ARCHITECTURE AND DATASET

The dataset under study came from the repository database of the UCI machine learning repository to the University of California based on the work of Dr W. Wolberg, Dr N. Street, Dr O. Mangasarian. [1] [3][4]

In this paper [3], using image analysis methods based on a graphical computer program called Xcyt, obtained several features about the cell from the samples. The computer calculates several nuclear features from each cell that can be described as Nuclear size, expressed by Radius, Perimeter and Areas features, Nuclear shape, expressed by Smoothness, Concavity, Compactness, Concave Points, Symmetry and Fractal Dimension, and finally Nuclear texture, measured by the standard deviation of the greyscale intensities in the component pixels. This method allowed quantifying nuclear cell features. [3][4]

The dataset is composed of 569 observations with 32 features or attributes. It is studied the features **"diagnosis"** which show the results of the medical test: cell benign (B) or malignant (M).

The rest of the attribute's information is:

- Radius (mean of distances from the centre to points on the perimeter).
- Texture (standard deviation of grey-scale values)
- Perimeter
- Area
- Smoothness (local variation in radius lengths)
- Compactness (perimeter^2 / area - 1.0)
- Concavity (severity of concave portions of the contour)
- Concave points (number of concave portions of the contour)
- Symmetry
- Fractal dimension

There are in total 30 features resulting from the mean, standard error and "worst" or largest (mean of the three largest values) calculated for each cell and additional features ID number.

A. Technical architecture

This dataset is stored in format Hadoop Dataset File System (HDFS) into the Hadoop environment: Hortonworks Data Platform (HDP) [33], running on the AWS Cluster with three Linux's machines CentOS 7 [24] [25]:

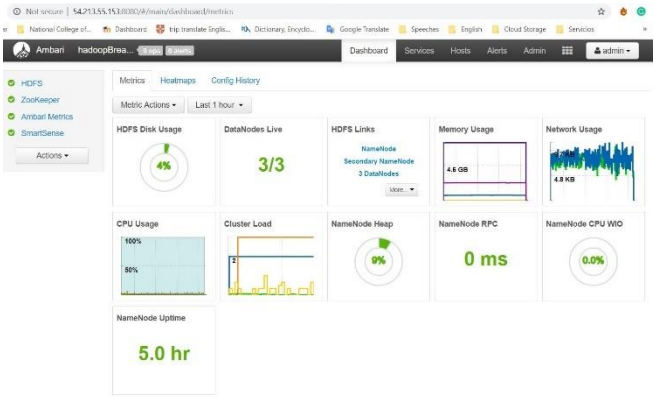

Fig. 3. HDP Hadoop environment on cluster AWS in cloud

- Hadoop server (master node)
- Hadoop data-node
- Hadoop data-node

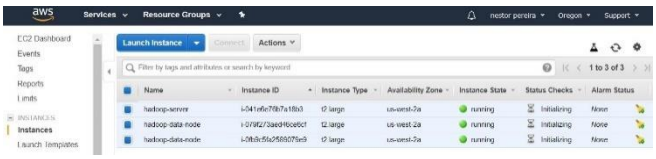

Fig. 4. Hadoop master and slaves nodes on AWS cluster in the cloud

The dataset is read from Algorithms programming in Python using REST API (WebHDFS) under development environment Anaconda-Spider. [17] [18] [19] [20] [21] [22] [23] [26]

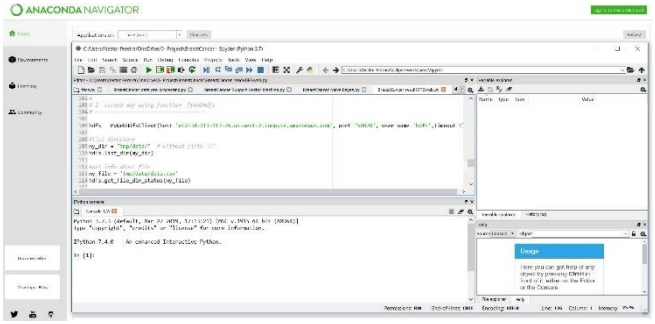

Fig. 5. Development environment Spider on Anaconda distribution for machine learning (Windows 10 – Mac OS)

Additionally, to the Hadoop Cloud Cluster technical environment showed before, it was implemented an alternative Hadoop virtual environment in local using a Linux machine CentOS 7 on VirtualBox.

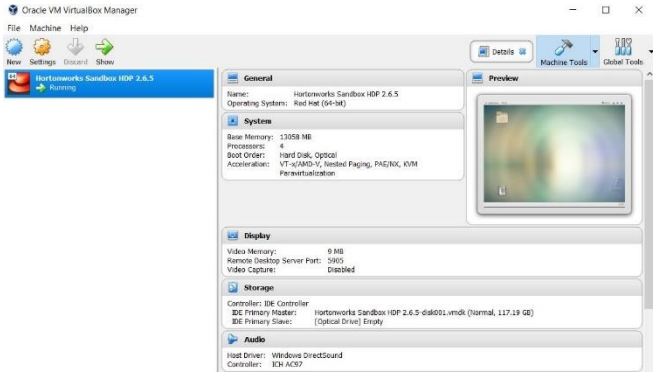

Fig. 6. Virtual Linux CentOS 7 on VirtualBox

Running the Hadoop distribution HDP Hortonworks Data Platform SandBox in one node.

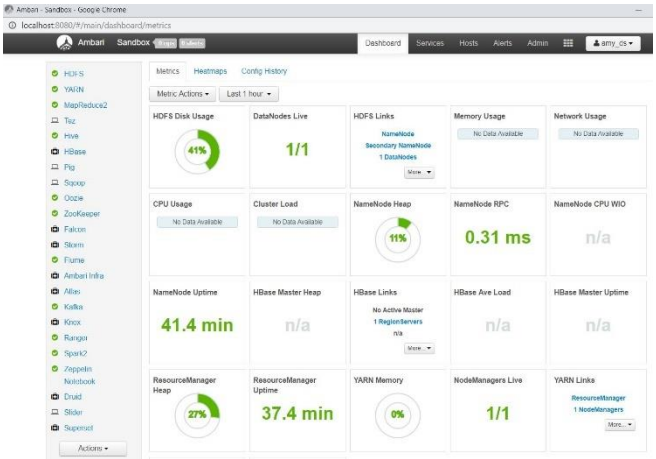

Fig. 7. Hadoop HDP SandBox one node environment on VirtualBox

B. *Diagram process*

The simple diagram below shows the flow of the processes in this project. There are two processes common for all algorithms: read the HDFS dataset from Hadoop and pre-processing and data wrangling to created training data and test data.

Later, all models are trained, testing and evaluated.

VII. DESCRIPTIVE STATISTICS OF THE ATTRIBUTES

The variable “diagnosis” is the dependent variable and indicate the cell is B (benign) or M (malignant). The class distribution is not imbalance with 357 benign, 212 malignant with ratio: 1.7:1 respectively.

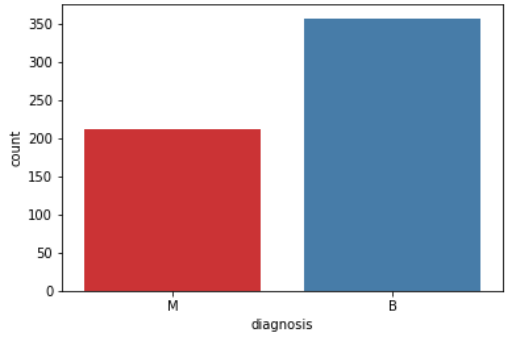

Fig. 8. Class distribution diagnosis

	diagnosis	radius_mean	...	symmetry_worst	fractal_dimension_worst
0	M	17.99	...	0.4601	0.11890
1	M	20.57	...	0.2750	0.08902
2	M	19.69	...	0.3613	0.08758
3	M	11.42	...	0.6638	0.17300
4	M	20.29	...	0.2364	0.07678

Fig. 9. A small sample of the dataset

The independent variable radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension measure in mean, standard error and "worst" value show this statistical metrics,

	count	mean	std	min	25%	50%	75%	max
radius_mean	569.0	1.41e+01	3.52e+00	6.98e+00	1.17e+01	1.34e+01	1.58e+01	28.11
texture_mean	569.0	1.93e+01	4.30e+00	9.71e+00	1.62e+01	1.88e+01	2.18e+01	39.28
perimeter_mean	569.0	9.20e+01	2.43e+01	4.38e+01	7.52e+01	8.62e+01	1.04e+02	188.50
area_mean	569.0	6.55e+02	3.52e+02	1.44e+02	4.20e+02	5.51e+02	7.83e+02	2501.00
smoothness_mean	569.0	9.64e-02	1.41e-02	5.26e-02	8.64e-02	9.59e-02	1.05e-01	0.16
compactness_mean	569.0	1.04e-01	5.28e-02	1.94e-02	6.49e-02	9.26e-02	1.30e-01	0.35
concavity_mean	569.0	8.88e-02	7.97e-02	0.00e+00	2.96e-02	6.15e-02	1.31e-01	0.43
concave points_mean	569.0	4.89e-02	3.88e-02	0.00e+00	2.03e-02	3.35e-02	7.40e-02	0.20
symmetry_mean	569.0	1.81e-01	2.74e-02	1.06e-01	1.62e-01	1.79e-01	1.96e-01	0.30
fractal_dimension_mean	569.0	6.28e-02	7.06e-03	5.00e-02	5.77e-02	6.15e-02	6.61e-02	0.10
radius_se	569.0	4.05e-01	2.77e-01	1.12e-01	2.32e-01	3.24e-01	4.79e-01	2.87
texture_se	569.0	1.22e+00	5.52e-01	3.60e-01	8.34e-01	1.11e+00	1.47e+00	4.88
perimeter_se	569.0	2.87e+00	2.02e+00	7.57e-01	1.61e+00	2.29e+00	3.36e+00	21.98
area_se	569.0	4.03e+01	4.55e+01	6.80e+00	1.79e+01	2.45e+01	4.52e+01	542.20
smoothness_se	569.0	7.04e-03	3.00e-03	1.71e-03	5.17e-03	6.38e-03	8.15e-03	0.03
compactness_se	569.0	2.55e-02	1.79e-02	2.25e-03	1.31e-02	2.04e-02	3.24e-02	0.14
concavity_se	569.0	3.19e-02	3.02e-02	0.00e+00	1.51e-02	2.59e-02	4.20e-02	0.40
concave points_se	569.0	1.18e-02	6.17e-03	0.00e+00	7.64e-03	1.09e-02	1.47e-02	0.05
symmetry_se	569.0	2.05e-02	8.27e-03	7.88e-03	1.52e-02	1.87e-02	2.35e-02	0.08
fractal_dimension_se	569.0	3.79e-03	2.65e-03	8.95e-04	2.25e-03	3.19e-03	4.56e-03	0.03
radius_worst	569.0	1.63e+01	4.83e+00	7.93e+00	1.30e+01	1.50e+01	1.88e+01	36.04
texture_worst	569.0	2.57e+01	6.15e+00	1.20e+01	2.11e+01	2.54e+01	2.97e+01	49.54
perimeter_worst	569.0	1.07e+02	3.36e+01	5.04e+01	8.41e+01	9.77e+01	1.25e+02	251.20
area_worst	569.0	8.81e+02	5.69e+02	1.85e+02	5.15e+02	6.86e+02	1.08e+03	4254.00
smoothness_worst	569.0	1.32e-01	2.28e-02	7.12e-02	1.17e-01	1.31e-01	1.46e-01	0.22
compactness_worst	569.0	2.54e-01	1.57e-01	2.73e-02	1.47e-01	2.12e-01	3.39e-01	1.06
concavity_worst	569.0	2.72e-01	2.09e-01	0.00e+00	1.15e-01	2.27e-01	3.83e-01	1.25
concave points_worst	569.0	1.15e-01	6.57e-02	0.00e+00	6.49e-02	9.99e-02	1.61e-01	0.29
symmetry_worst	569.0	2.90e-01	6.19e-02	1.57e-01	2.50e-01	2.82e-01	3.18e-01	0.66
fractal_dimension_worst	569.0	8.39e-02	1.81e-02	5.50e-02	7.15e-02	8.00e-02	9.21e-02	0.21

Fig. 10. Statistics metrics of the independent variables

The diagram of density shows the distribution of the independent variables:

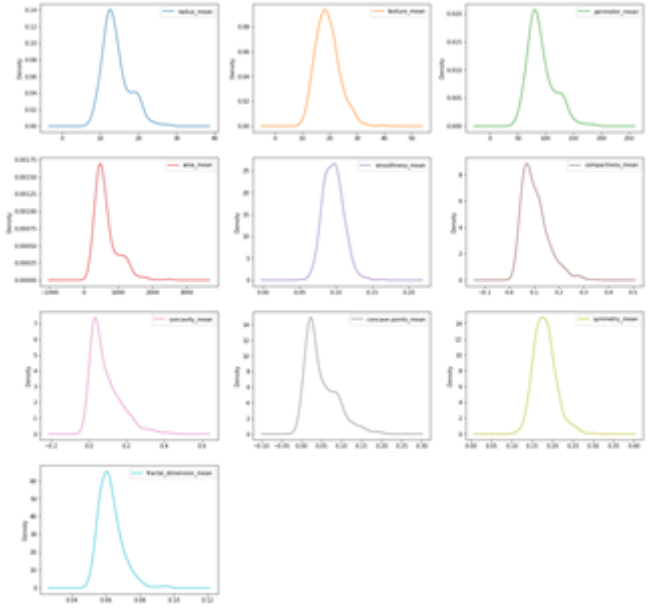

Fig. 11. Density diagram of the independent variables measured in mean

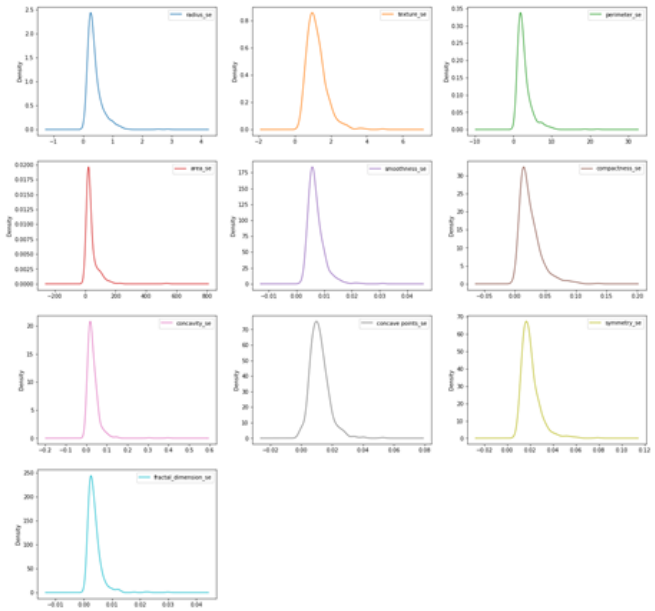

Fig. 12. Density diagram of the independent variables measured in standard error

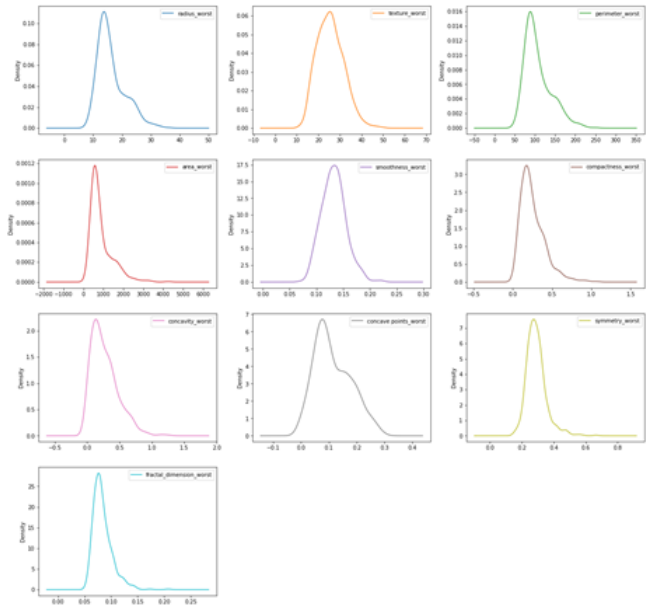

Fig. 13. The simple diagram below shows the flow of the processes in this project.

In the Boxplot diagrams below shows the quartiles of the independent variables and some outliers observations in many of them.

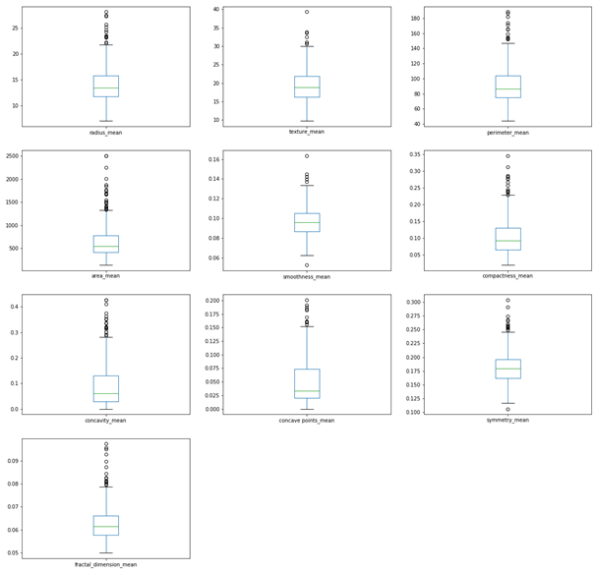

Fig. 14. Boxplot diagram of the independent variables measured in mean

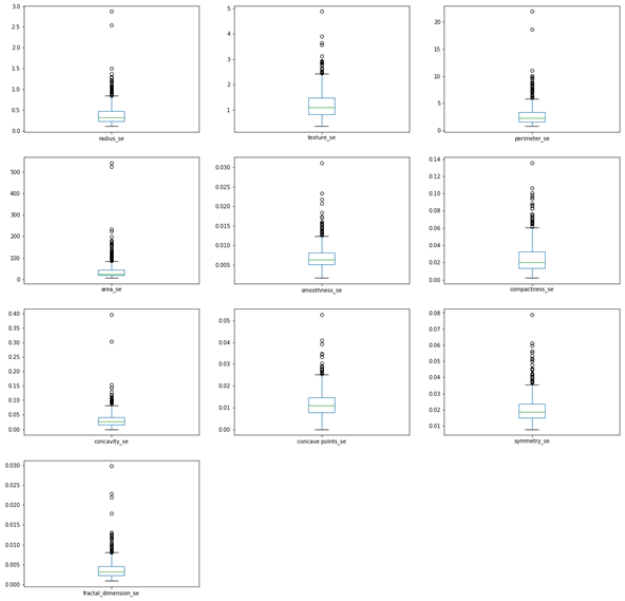

Fig. 15. Boxplot diagram of the independent variables measured in standard error

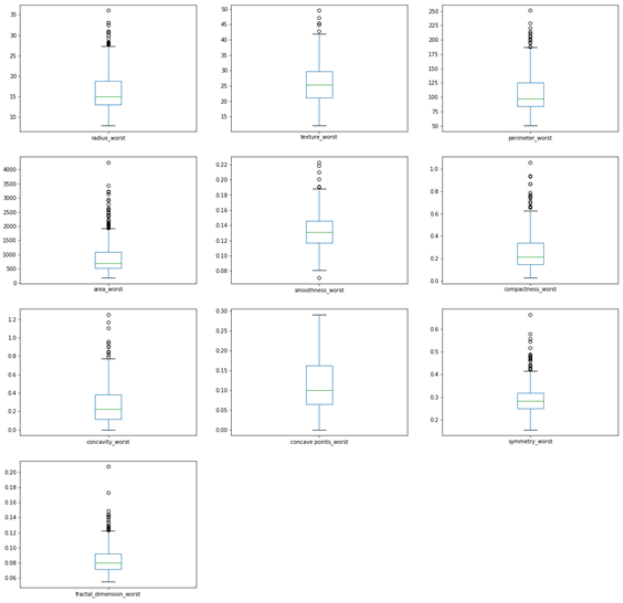

Fig. 16. Boxplot diagram of the independent variables measured in "worst" value

In the density's diagram shows an evident positive skewness in all the independent variables. Its skewness can be verified in numerical values in which the skewness is higher than zero (0).

radius_mean	0.942
texture_mean	0.650
perimeter_mean	0.991
area_mean	1.646
smoothness_mean	0.456
compactness_mean	1.190
concavity_mean	1.401
concave points_mean	1.171
symmetry_mean	0.726
fractal_dimension_mean	1.304
radius_se	3.089
texture_se	1.646
perimeter_se	3.444
area_se	5.447
smoothness_se	2.314
compactness_se	1.902
concavity_se	5.110
concave points_se	1.445
symmetry_se	2.195
fractal_dimension_se	3.924
radius_worst	1.103
texture_worst	0.498
perimeter_worst	1.128
area_worst	1.859
smoothness_worst	0.415
compactness_worst	1.474
concavity_worst	1.150
concave points_worst	0.493
symmetry_worst	1.434
fractal_dimension_worst	1.663

Fig. 17. Skewness values for the independent variables

Its show the histogram according to the values of the dependent variable diagnosis, B (benign) in green or M (malignant) in red.

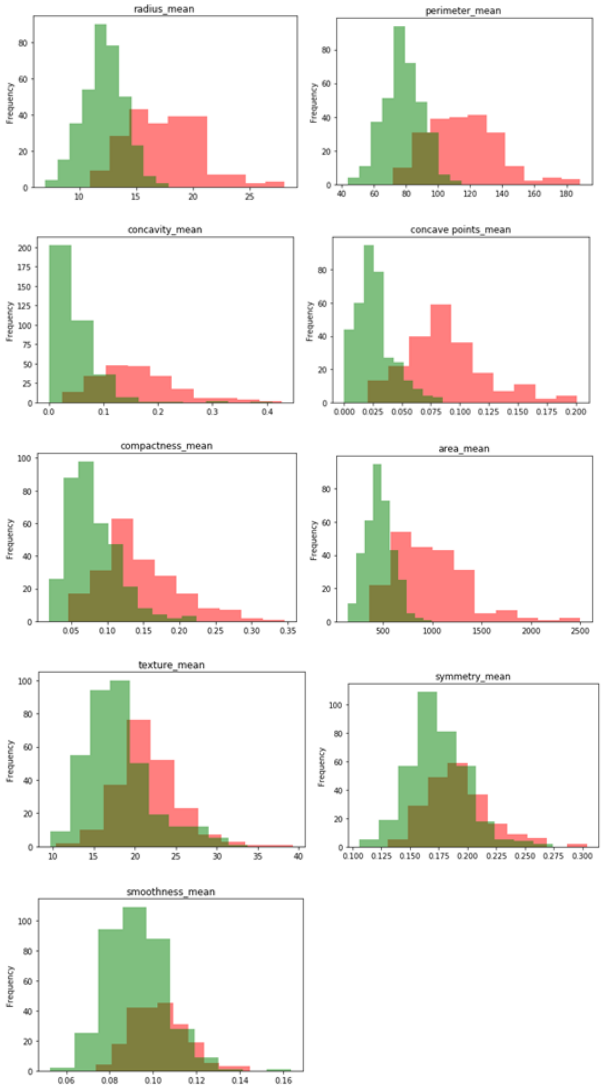

Fig. 18. Histogram according to the values of the dependent variable diagnosis, B (benign) in green or M (malignant) in red, measures in the mean.

It is interesting to note that the distribution of the features with diagnosis B (benign) is more skewness comparative to the distribution of features with diagnosis M (malignant).

Some of the methods used in this project assume that there is no correlation between the independent variables, so it is essential to show the values, numerical and visual, of the correlation among the attributes.

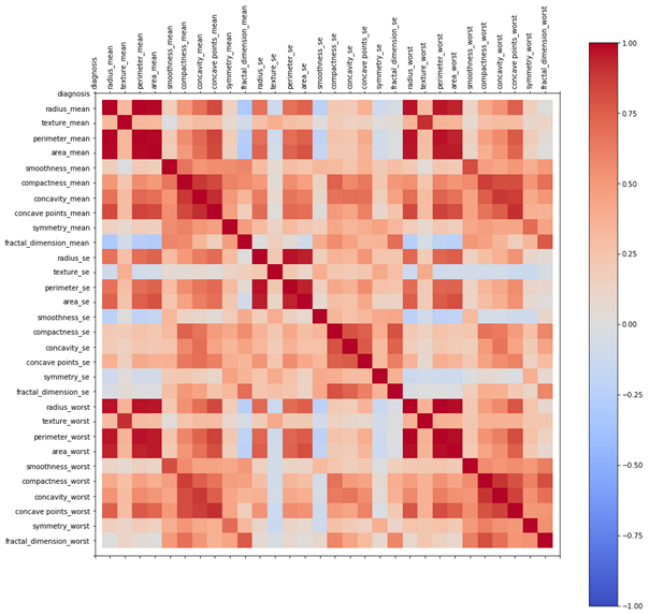

Fig. 19. Correlation among independent variables.

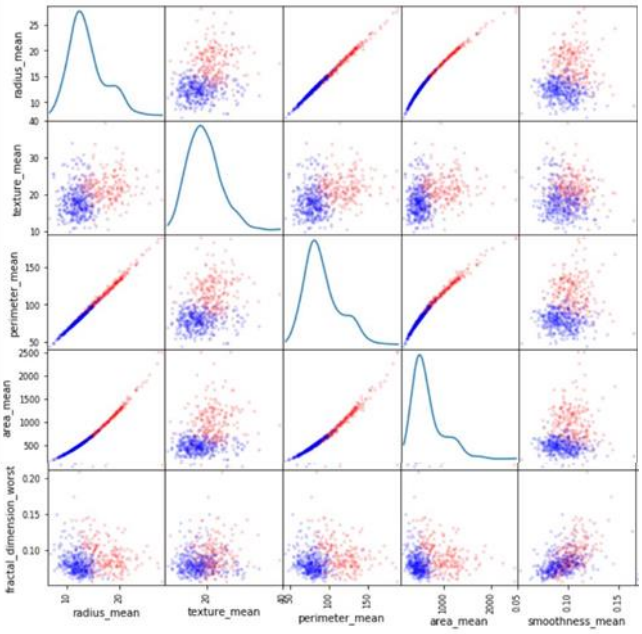

Fig. 20. Zoom of correlation among independent variables: radius, area, perimeter.

There are some areas which show a strong correlation identifies with the variables: radius, area, perimeter, concavity and concave points; it is called multicollinearity. This correlation could be explained because, for example, area and perimeter can be calculated from the radius.

	radius_mean	texture_mean	perimeter_mean	area_mean
radius_mean	1.000	0.324	0.998	0.987
texture_mean	0.324	1.000	0.330	0.321
perimeter_mean	0.998	0.330	1.000	0.987
area_mean	0.987	0.321	0.987	1.000
smoothness_mean	0.171	-0.023	0.207	0.177
compactness_mean	0.506	0.237	0.557	0.499
concavity_mean	0.677	0.302	0.716	0.686
concave points_mean	0.823	0.293	0.851	0.823
symmetry_mean	0.148	0.071	0.183	0.151
fractal_dimension_mean	-0.312	-0.076	-0.261	-0.283

Fig. 21. Zoom of correlation among independent variables: radius, area, perimeter.

VIII. PRE-PROCESSING AND DATA WRANGLING

Before it applies any algorithms to the dataset, it necessary to do some task of pre-processing in which it applies some techniques of data wrangling and the attributes are transformed with a Gaussian distribution using the standardisation technique.

Firstly, it is extracted and delete the first column of the dataset, ID number, because it is only an identification of the row in the dataset and not provide any other information.

Secondly, all the attributes, independent variables, are transformed in attributes standardised. The original dataset is splitter in one dataset Y only with the dependent variable, diagnosis, and another dataset X with the rest of the independent variables.

Each attribute, in the dataset X, is rescaled using the mean and standard deviation.

Finally, it is created a training dataset and test dataset to be used for all the algorithms.

The dataset is split 66% to data training and 33% to data test generated using the same random number, 77 in this project, to guarantee that the results are reproducible.

X_train, independent variables, training data

X_test, independent variables, test data

Y_train, dependent variable, training data

Y_test, dependent variable, test data

It is essential to create the same dataset, training data and test data, using the same proportion and the same random number to ensure that the results of each method are comparable.

IX. TRAINING, TESTING AND EVALUATION

After the process of split and standardise the original dataset, the data is ready to train the model and do the testing to compare with predictions.

The process of training, testing and evaluation are common for all methods to obtain comparable results.

A. Logistic Regression

It is one of the most useful methods to evaluate the binary problem with dichotomous variable, in this case, the variable called diagnosis, using a logistic function to predict “benign (B)” or “malignant (M)” (binary regression). It uses the

features, independent variables, relate to the dependent variable.

The program "BreastCancer logistics regression.py", in Python, implement this algorithm using the training dataset X_train (data standardized) to train the model and evaluate the accuracy.

Logistic regression. **Accuracy Training set score: 98.950%**

Later, apply re-sampling techniques, k-fold (k=10) cross-validation and measure the accuracy.

Logistic regression. **Accuracy after Cross-Validation**:

- Means: **97.895%**
- Standard deviation: **1.579%**

The next step, it calculates the area under the ROC Curve (AUC). AUC close to 1 suggesting the prediction is excellent. On the other hand, values close to 0.5 means bad prediction.

Logistic regression. **Area under ROC Curve (AUC)**:

- Means: 0.995
- Standard deviation: 0.011

Finally, it shows the performance metrics for accuracy – confusion matrix, the precision, recall, F1-score and support.

Logistic regression. **Confusion Matrix**:

[[**119** 2]

[4 **63**]]

True Positive (TP): **119**

True Negative (TN): **63**

False Negative (FN): 2

False Positive (FP): 4

And the rest of the performance values:

	precision	recall	f1-score	support
B	0.97	0.98	0.98	121
M	0.97	0.94	0.95	67
micro avg	0.97	0.97	0.97	188
macro avg	0.97	0.96	0.96	188
weighted avg	0.97	0.97	0.97	188

Even though there is a high correlation among some independent variables, multicollinearity, the perform of the model is excellent with high accuracy, and a high value of AUC. Therefore, this is a good model candidate to fix the problem.

B. Naive Bayes

The model predicts "benign (B)" or "malignant (M)" using the Bayes theorem based on the conditional probability of each event. The model assumes that the variables are independent and with normalizing distribution, or Gaussian distribution.

In this case, there is multicollinearity among some variables, and almost all of them have a skewness distribution.

The program "BreastCancer naive bayes.py" in Python implement this algorithm follows the same structure that previous model.

The accuracy after training the Naive Bayes model with the dataset X_train is:

Naive Bayes. **Accuracy Training set score: 94.751%**

Later, apply re-sampling techniques, k-fold (k=10 and same seed) cross-validation and measure the accuracy.

Naive Bayes. **Accuracy after Cross-Validation**:

- Means: **93.435%**
- Standard deviation: **5.031%**

And calculated the area under the ROC curve (AUC), confusion matrix and precision, recall, F1-score and support.

Naive Bayes. **Area under ROC Curve (AUC)** :

- Means: 0.987
- Standard deviation: 0.014

Naive Bayes. **Confusion Matrix**:

[[**118** 3]

[9 **58**]]

True Positive (TP): **118**

True Negative (TN): **58**

False Negative (FN): 3

False Positive (FP): 9

	precision	recall	f1-score	support
B	0.93	0.98	0.95	121
M	0.95	0.87	0.91	67
micro avg	0.94	0.94	0.94	188
macro avg	0.94	0.92	0.93	188
weighted avg	0.94	0.94	0.94	188

This model affected the multicollinearity is not good enough that the previous Logistic Regression model.

C. k-Nearest Neighbours

k-Nearest Neighbours uses a distance metric to find the k most similar observations, in this case, use the model from the Scikit Learning Libraries that use the Minkowski distance. It is a generalization of the Euclidean distance and the Manhattan distance.

The program "BreastCancer k-NN neighbours.py" in Python implement this algorithm follows the same methodological structure that previous models.

The advantage of using this method in this project is there is no assumption about the distribution of the variables. Therefore, the skewness distribution of the variables would not affect the results of the models. However, it is affected by the outlier's observation in the dataset that was shown in the previous chapters.

The challenge in this model is found the right values of k, a number of neighbours, that would use to train the model.

The graphs below show the accuracy to several values of k.

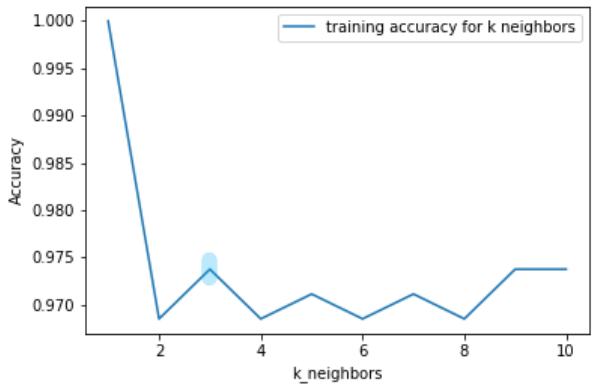

Fig. 22. The optimum value of k by accuracy.

Training the model for k=3 neighbours (accuracy: **0.974%)**, apply re-sampling techniques, k-fold (k=10 and same seed) cross-validation and measure the accuracy.

k-Nearest Neighbours. **Accuracy after Cross-Validation**:

- Means: **96.053%**
- Standard deviation: **2.697%**

And calculated the area under the ROC curve (AUC), confusion matrix and precision, recall, F1-score and support.

k-Nearest Neighbours. **Area under ROC Curve (AUC)**:

- Means: **0.981**
- Standard deviation: **0.016**

k-Nearest Neighbours. **Confusion Matrix**:

[[121 0]

[8 59]]

True Positive (TP): **121**

True Negative (TN): **59**

False Negative (FN): 0

False Positive (FP): 8

	precision	recall	f1-score	support
B	0.94	1.00	0.97	121
M	1.00	0.88	0.94	67
micro avg	0.96	0.96	0.96	188
macro avg	0.97	0.94	0.95	188
weighted avg	0.96	0.96	0.96	188

The results of the test process show a promising model to solve the problem, mainly because the false-negative value is zero (0), and the Recall is 100%. However, it is important to note that accuracy has a high variance, and the method is affected by outliers. Therefore, this result could be specific for this dataset and difficult to take as a general model for all cases.

D. Random Forests

The random forests algorithm is an alternative to the standard decision tree because it is most stable, more accuracy, and less affected by the change in the training data. In general, the method tends to fit the problem of over-fitting in the decision tree.

The program "BreastCancer Random Forests.py" in Python implement this algorithm follows the same methodological structure that previous models.

The accuracy after training the Random Forests model with the dataset X_train is:

Decision tree. Accuracy Training set score: 97.900%

Random Forests. **Accuracy Training set score**: **99.475%**

One important point in the analysis working with decision trees classification to determine the importance of the features into the decision algorithm.

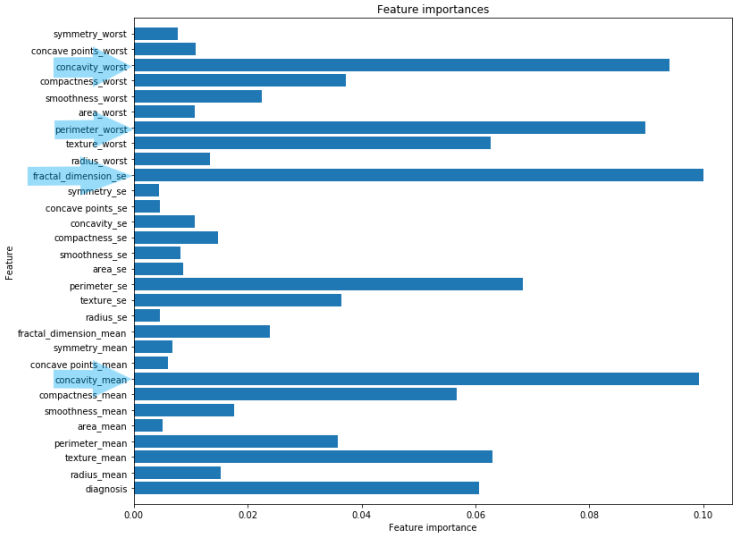

Fig. 23. Features importances on Random Forests Classification.

This result shows the variables concavity-mean, concavity-worst, perimeter-worst, and fractal-dimension-se as the most important variables into the model.

The algorithm use from the Scikit Learning Libraries calculates the Gini impurity index to provide the prediction.

Later, apply re-sampling techniques, k-fold (k=10 and same seed) cross-validation and measure the accuracy.

Random Forests. **Accuracy after Cross-Validation**:

- Means: **96.059%**
- Standard deviation: **3.172%**

And calculated the area under the ROC curve (AUC), confusion matrix and precision, recall, F1-score and support.

Random Forests. **Area under ROC Curve (AUC)**:

- Means: **0.992**
- Standard deviation: **0.010**

Random Forests. **Confusion Matrix**:

[[119 2]

[9 58]]

True Positive (TP): **119**

True Negative (TN): **58**

False Negative (FN): 2

False Positive (FP): 9

	precision	recall	f1-score	support
B	0.93	0.98	0.96	121
M	0.97	0.87	0.91	67
micro avg	0.94	0.94	0.94	188
macro avg	0.95	0.92	0.93	188
weighted avg	0.94	0.94	0.94	188

The result is not better than the previous models and shows high variance in the accuracy.

E. Support Vector Machine

The algorithms use from the Scikit Learning Libraries have a linear kernel, "Linear", and non-linear kernel, "polynomial" and "RBF" (radial basis function) to classify the observations.

The accuracy results after applying those tree functions in the X_train dataset is:

- Accuracy with SVM linear function: **99.213%**
- Accuracy with SVM polynomial non-linear function: **90.551**%
- Accuracy with SVM radial basic non-linear function: **98.688%**

However, the results show differences when applying re-sampling techniques, k-fold (k=10 and same seed) cross-validation and measure the accuracy.

Accuracy for **linear kernel after Cross-Validation**:

- Means: **96.316%**
- Standard deviation: 2.412%

Accuracy for **polynomial kernel after Cross-Validation**:

- Means: **89.764%**
- Standard deviation: 4.315%

Accuracy for **radial basis function kernel (RBF) after Cross-Validation**:

- Means: **97.105%**
- Standard deviation: 2.989%

In which, it is clearly better the accuracy using the RBF kernel.

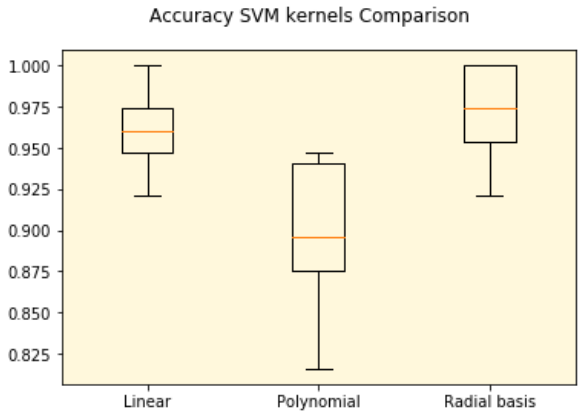

Fig. 24. Accuracy SVM kernel comparison.

Considered the RBF kernel as a better option to the model, it calculated the area under the ROC curve (AUC), confusion matrix and precision, recall, F1-score and support.

Radial basis function kernel. **Area under ROC Curve** (AUC):

- Means: 0.994
- Standard deviation: 0.013

Radial basis function kernel. **Confusion Matrix**:

[[119 2]

[3 64]]

True Positive (TP): **119**

True Negative (TN): **64**

False Negative (FN): 2

False Positive (FP): 3

	precision	recall	f1-score	support
B	0.93	0.98	0.96	121
M	0.97	0.87	0.91	67
micro avg	0.94	0.94	0.94	188
macro avg	0.95	0.92	0.93	188
weighted avg	0.94	0.94	0.94	188

As expected, the results from applying the SVM radial basis function kernel shows excellent results with high accuracy, high precision and also a high recall into this dataset. It means that it is clearly a model to be considered in the final solution.

X. EVALUATION

According to the results, there are two methods which fix very well to solve the problem with high accuracy and low values of observations in False-Negative: SVM RBF kernel and Logistic Regression. The program "BreastCancer Cross-validation Comparatives.py" in Python implement this algorithm.

Comparatives of the methods based on k-folds Cross-Validation:

- Accuracy SVM Linear Kernel: 96.316% (2.41%)
- Accuracy SVM RBF Kernel: **97.105**% (2.99%)
- Accuracy Logistic Regression: **97.895**% (1.58%)
- Accuracy k-N Neighbour: 96.053% (2.70%)
- Accuracy Decision Tree: 93.704% (4.27%)
- Accuracy Random Forests: 96.059% (3.17%)
- Accuracy Naive Bayes: 93.435% (5.03%)

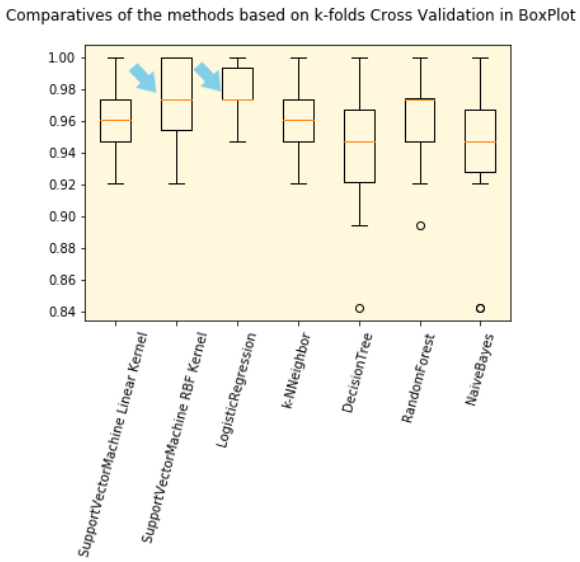

Fig. 25. Accuracy methods comparison.

In order to compare those two methods, it is a convenience to use two different performance metrics:

- The area under the ROC curve (AUC)
- The area under the precision-recall curve

Generally, the use of ROC curve should be used when the numbers of observation for each class is a balance. On the other hand, Precision-Recall curve should be used when there is a class imbalance.[15][16]

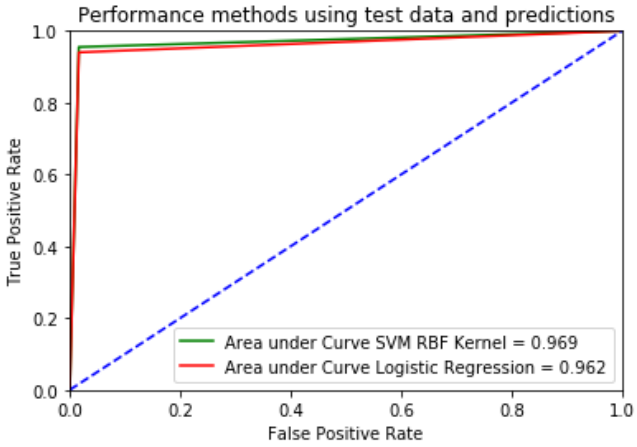

Fig. 26. Area under the ROC curve.

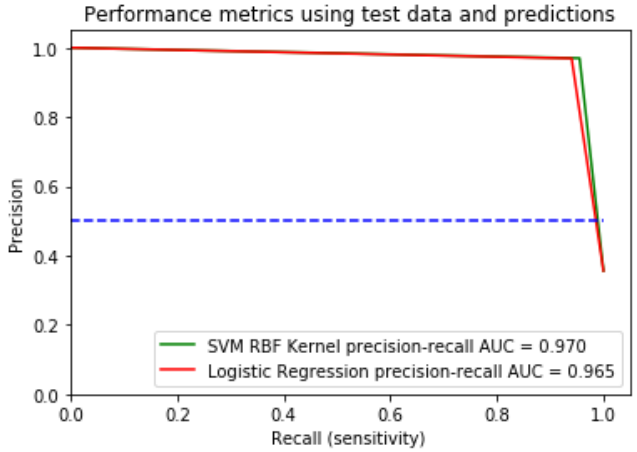

Fig. 27. Area under the Precision-Recall curve.

The observations in the class "diagnosis" are moderate imbalance:

B 357 observations

M 212 observations

Both performance metrics show high values and satisfied the accuracy-precision requirements defined in the objectives and support both methods as a trustworthy model.

XI. Conclusion and Future Work

Using coherent comparative methods, in this project was compared with different algorithms for classification to find the best trustily model to fix the problem of diagnosis Breast Cancer.

The main focus was on these classification algorithms and covered the demand for high accuracy and precision of the problem.

In this approach, it was found that perhaps the best solution is to use more than one method applied in a coherent way to support a trusty solution.

It is important to mention that innovate technologies like Hadoop, Python Scikit-learn libraries and clustering was used in this project.

Finally, the potential to use this approach in other diagnosis types of cancer should be investigated and using the Pipeline utilities in Python Scikit-learn libraries to find new solutions that involve two or more methods working together in a complementary way.

References

[1] UCI Machine Learning, 1995. https://www.kaggle.com/uciml/breast-cancer-wisconsin-data Access: June 2019

[2] M. Grover, T. Malaska, J. Seidman, G. Shapira. "Hadoop Application Architectures", 2015, O'Reilly. (pp 29, 47)

[3] Wolberg WH1, Street WN, Heisey DM, Mangasarian OL. "Computerized breast cancer diagnosis and prognosis from fine-needle aspirates", 1995. https://www.ncbi.nlm.nih.gov/pubmed/7748089

[4] O.L. Mangasarian, W.N. Street and W.H. Wolberg, "Breast cancer diagnosis and prognosis via linear programming", 1995, Operations Research, 43(4).

[5] B. Eliason, MIS, D. Crockett, "What Is Data Mining in Healthcare", 2014, https://www.healthcatalyst.com/data-mining-in-healthcare Access: June 2019

[6] Irish Cancer Society, 2019, https://www.cancer.ie/reduce-your-risk/health-education/cancer-awareness-campaigns/breast-cancer-awareness/breast-cancer-key-facts#sthash.SspCY7pU.6gcw4P08.dpbs Access: June 2019

[7] K. Ping Shung (2018) "Accuracy, Precision, Recall or F1", 2018, https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9 Access: July 2019

[8] S. Collet, "Python - Read & Write files from HDFS", 2016, https://creativedata.atlassian.net/wiki/spaces/SAP/pages/61177860/Python+-+Read+Write+files+from+HDFS Access: July 2019

[9] Virtual Medical Centre, 2018, https://www.myvmc.com/investigations/fine-needle-aspiration-biopsy-fna/ Access: July 2019

[10] S. Li, "Machine Learning for Diabetes". Towards Data Science, 2017, https://towardsdatascience.com/machine-learning-for-diabetes-562dd7df4d42 Access: July 2019

[11] Hortonworks Inc., 2018, "Hortonworks Data Platform: HDP 3.1", https://hortonworks.com/products/data-platforms/hdp/ Access: June 2019

[12] Frank, E., Hall, M. A., Pal, C.J. and Witten, I.H., 2017, "Data mining: Practical machine learning tools and techniques". 4th edition. Cambridge, Massachusetts: Elsevier/Morgan Kaufmann. (pp 147).

[13] MarinStatsLectures, 2018, https://statslectures.com/r-stats-datasets Access: June 2019

[14] Jason Brownlee, 2019, "Machine Learning Mastery with Python".

[15] J. Brownlee, "roc-curves-and-precision-recall-curves-for-classification-in-python", 2018, https://machinelearningmastery.com/roc-curves-and-precision-recall-curves-for-classification-in-python/ Access: July 2019.

[16] T. Saito and M. Rehmsmeier, "The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets", 2015, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4349800/ Access: July 2019.

[17] The Apache Software Foundation, "WebHDFS REST API ", 2008 , https://hadoop.apache.org/docs/r1.0.4/webhdfs.html#OPEN Access: July 2019

[18] F. Souto, "webhdfspy", 2015, , https://webhdfspy.readthedocs.io/en/latest/ Access: July 2019

[19] S. Gonzales, "pywebhdfs 0.2.2 documentation", 2013, https://pythonhosted.org/pywebhdfs/index.html?highlight=webhdfs Access: July 2019

[20] M. Monsch, GitHub, Inc., "API and command line interface for HDFS", 2019, https://github.com/mtth/hdfs Access: July 2019

[21] GitHub, Inc, "ProjectMeniscus/pywebhdfs", 2019 , https://github.com/ProjectMeniscus/pywebhdfs/blob/master/pywebhdfs/webhdfs.py, Access: July 2019

[22] Hortonworks Inc. , "Hortonworks Community Connection", 2019, https://community.hortonworks.com/questions/, Access: July 2019

[23] Python Software Foundation, "webhdfspy 0.3.5", 2019, https://pypi.org/project/webhdfspy/ Access: July 2019

[24] A. Karwal, "HDP 2.6 on AWS", 2018, https://cloudxlab.com/blog/install-hdp-on-aws/ Access: July 2019

[25] Laniakea Consulting, "HDP 2.6 install on AWS EC2 instances", 2018, https://medium.com/@laniakeagroupllc/hdp-install-on-ec2-using-ambari-991cc72e8742

[26] S. Collet, "Read & Write files from HDFS", 2016, https://creativedata.atlassian.net/wiki/spaces/SAP/pages/61177860/Python+-+Read+Write+files+from+HDFS. Access: July 2019

[27] A. Albon, "Machine Learning with Python Cookbook: Practical Solutions from Preprocessing to Deep Learning (p. 91)", 2018, O'Reilly, First edition.Kindle Edition.

[28] Z. Prekopcsák, T. Henk and C. Gáspár-Papanek, 2010, "Cross-validation: the illusion of reliable performance estimation". RCOMM 2010: RapidMiner Community Meeting and Conference. Dortmund, Germany. http://prekopcsak.hu/papers/preko-2010-rcomm.pdf (pp 2-5)

[29] James, Witten, Hastie, Tibshirani, 2013, "An Introduction to Statistical Learning with Applications in R", Springer (pp 25,176)

[30] k. Staci, 2018, "The Powerful Role of Big Data In The Healthcare Industry", SmartData Collective, https://www.smartdatacollective.com/powerful-role-big-data-healthcare-industry/ Access: July 2019

[31] B. Marr, 2015, "How Big Data Is Changing Healthcare", Forbes, https://www.forbes.com/sites/bernardmarr/2015/04/21/how-big-data-is-changing-healthcare/#745e9f0e2873 Access: July 2019

[32] J. Brownlee, 2018, "How to Reduce Variance in a Final Machine Learning Model", https://machinelearningmastery.com/how-to-reduce-model-variance/ Access: July 2019

[33] Wikipedia, 2019, "Sensitivity and specificity, "https://en.wikipedia.org/wiki/Sensitivity_and_specificity#Specificity Access: July 2019.

[34] Hortonworks, Inc., "Apache Ambari Installation", 2019, https://docs.hortonworks.com/HDPDocuments/Ambari-2.7.3.0/bk_ambari-installation/content/set_up_the_ambari_server.html, Access: July 2019.