Advanced Machine Learning Project

Breast Cancer Classification

In this project, we have chosen working on the breast cancer Wisconsin dataset. We first present and explore the data inside this dataset. Then we will build a binary classification model. We will compare 5 classification machine learning methods. Finally, we will visualize our results and we will present in depth the selected model for the prediction of breast cancer.

# Analysis of the dataset

The dataset we will make our predictions on is the Wisconsin breast cancer dataset. This dataset presents the diagnostic of 569 female patient to diagnose whether they have a breast cancer or not. The observations come from the extraction of information from a cell taken inside the breast of the patient.

**Analysis of the domain application:**

The dataset is composed of 32 features. The two first features correspond to information about the patient:

* ID number of the patient: we won’t use this feature because it is irrelevant.
* Diagnosis of the patient: this feature is the target feature. The cell can either be benign (the breast doesn’t present a tumor) or malignant (the breast presents a tumor). Since we use categorical values, a benign cell will correspond to 1 and a malignant cell will correspond to 0.

The 30 remaining features are numerical features which were computed from a digitized image of a fine needle aspirate of a breast mass. They describe characteristics of the cell nuclei present in the image:

* The radius of the cell nuclei. Since it is not perfectly circular, it is computed as the mean of the distances from center to points on the perimeter.
* The texture of the cell nuclei. It is computed as the standard deviation of the gray scale values. The image of the cell is in gray scale.
* The perimeter of the cell nuclei. It is the length of the contours of the cell.
* The area of the cell nuclei.
* The smoothness of the cell nuclei. It is computed as the local variation in radius lengths. Since it is not a perfect circular shape, the different radiuses are not uniform. The variation of them results in the smoothness. For instance, if the cell nuclei was perfectly circular, the smoothness would be equal to zero because there won’t be any possible variation.
* The compactness is the result of a mathematical formula between the perimeter and the area of the cell nuclei (perimeter^2 / area - 1.0).
* The concavity results in the severity of concave portions of the contour.
* The concave points. It is the number of concave portions of the contour of the cell nuclei.
* The symmetry
* The fractal dimension ("coastline approximation" - 1)

For each of the ten previous numerical features, three are computed:

* The mean of the measures
* The standard error
* The “worst” or the largest of these measures. It is the mean of the three largest values.

This results in a total of 30 features computed for each image. One image corresponds to one patient.

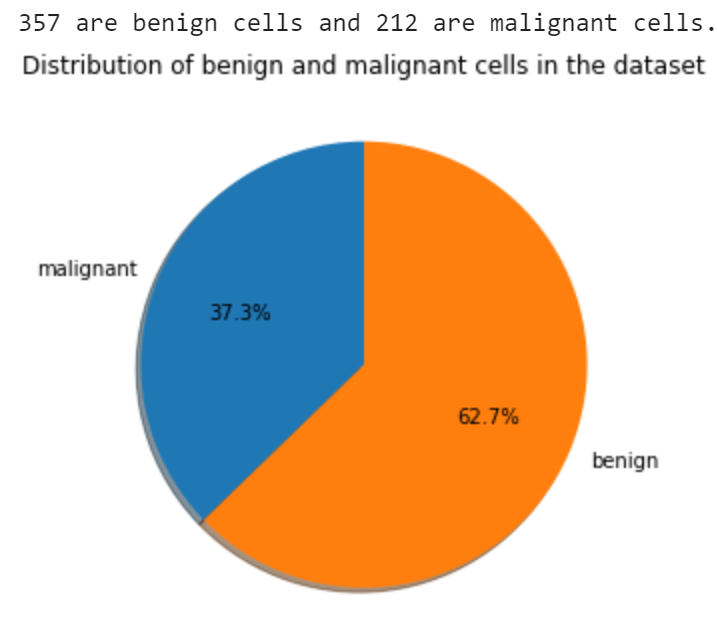
**Goal of the prediction:**

The goal of the prediction is straight forward. It is about predicting whether the patient presents a breast tumor or not. It is a binary classification problem sincewe have only two possibilities: the cell nuclei is benign or the cell nuclei is malignant.

Based on the features of the cell’s nuclei (size, shape…), we have to predict if the cell is a malignant or benign cell. This enables to make a diagnosis on whether the breast presents a tumor or not. It is a binary classification problem.

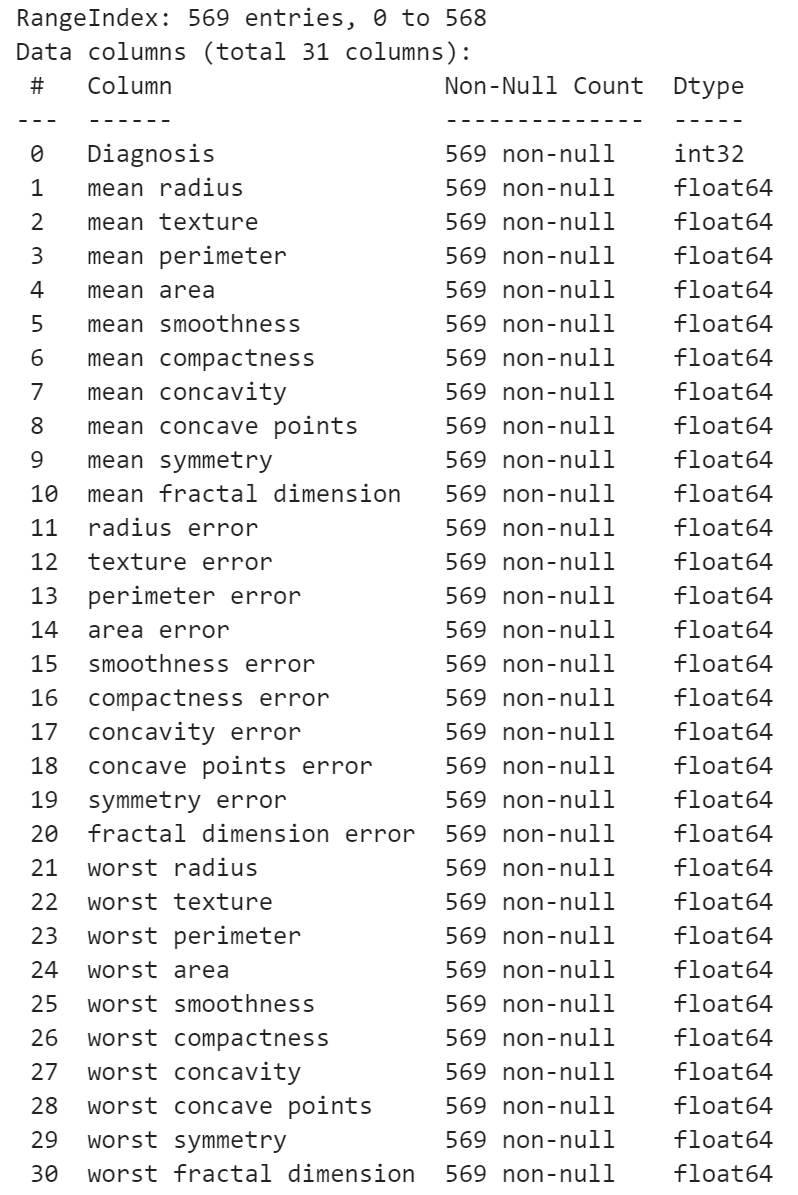
**Statistics on data:**

The data presents a total of 569 observation with 30 numerical features computed from the cell nuclei image and with a diagnosis feature. The repartition of malignant and benign observations is the following:



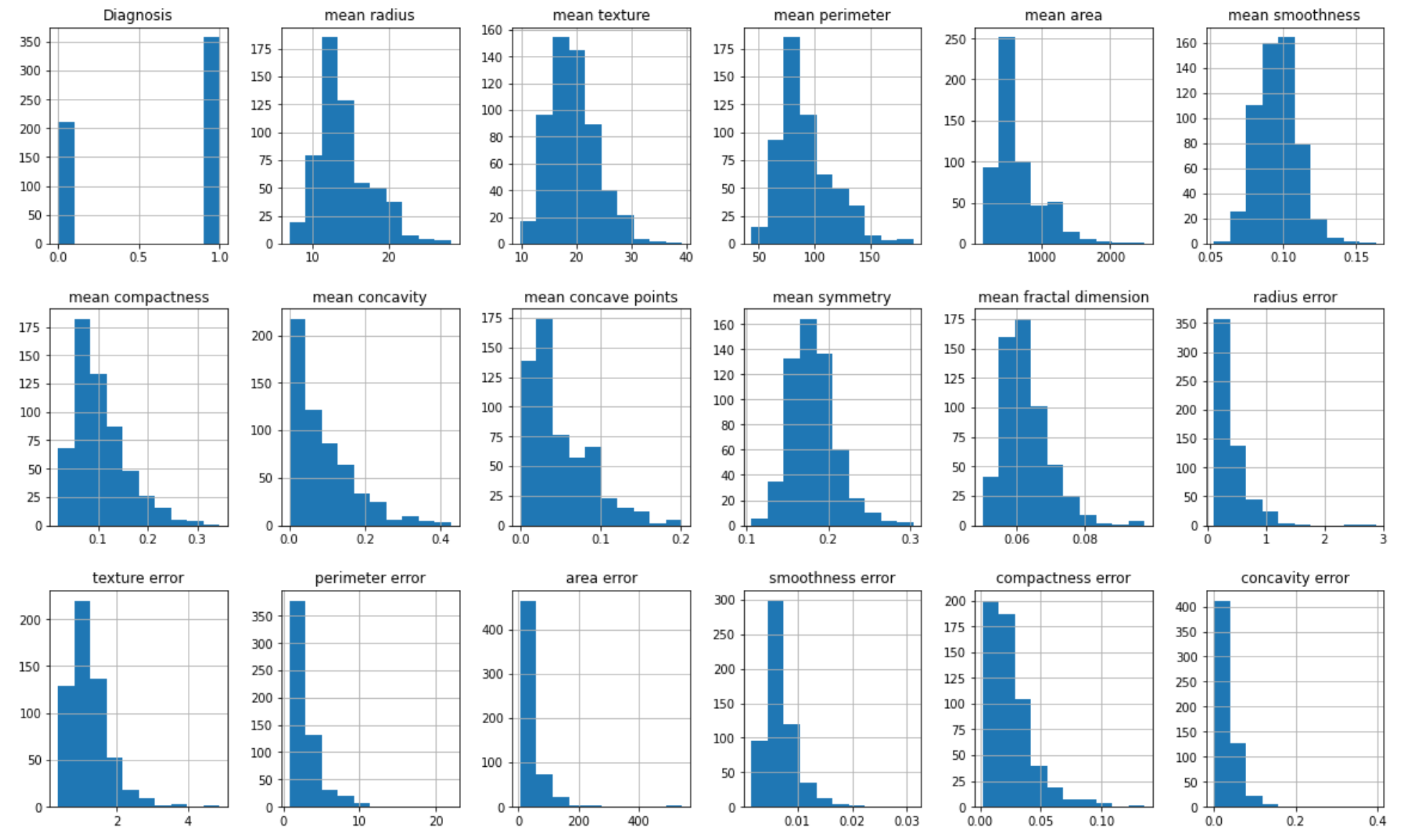
*Diagram of the repartition between malignant and benign cells in the dataset*

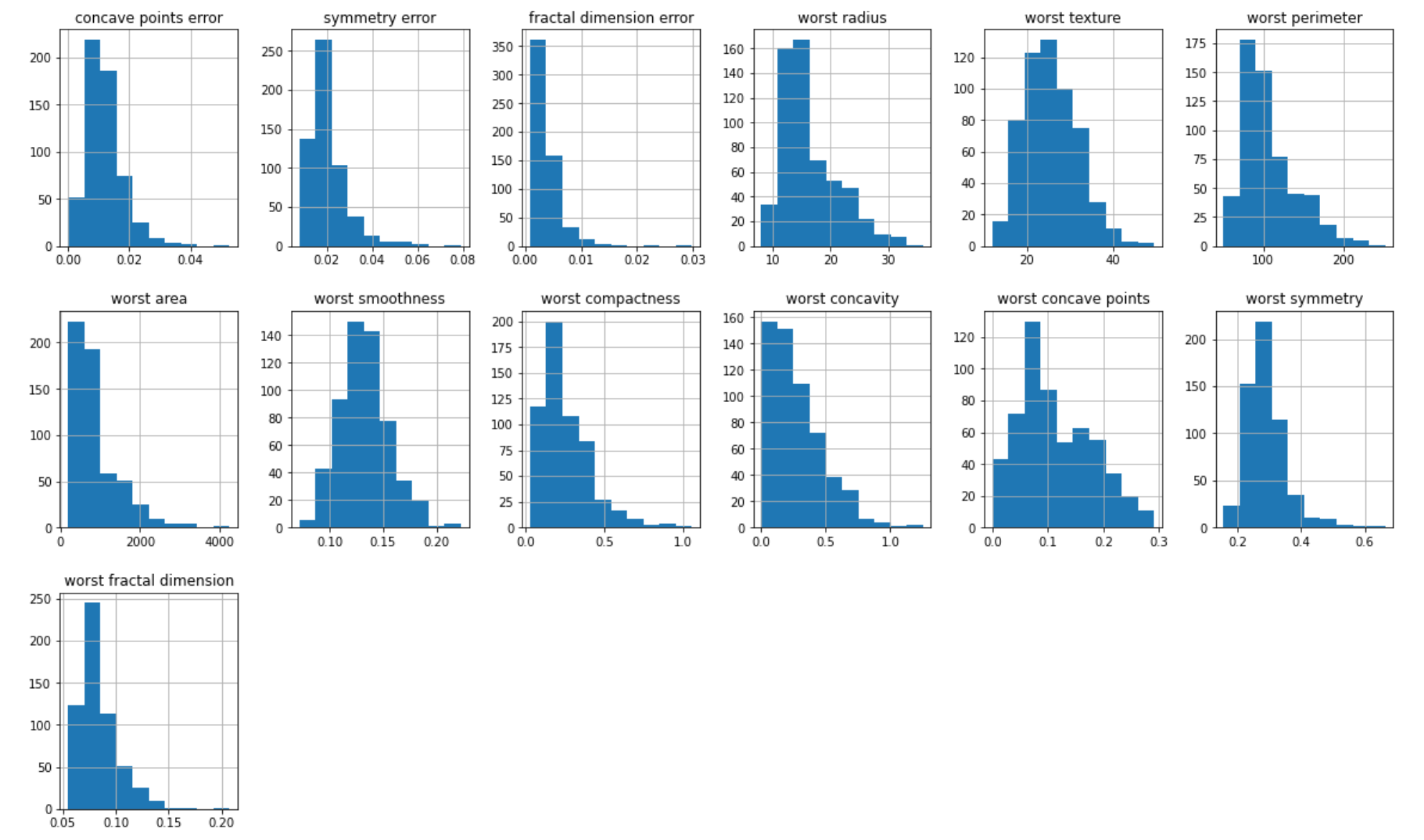
The dataset is almost balanced since we have a repartition of almost 60% of benign cells and of 40% of malignant cells. There are no missing values as we can see in the sceen shot of the result of a Python command to get information on the dataset:



*Image of the names and types of every features. There are no missing values*

In the notebook associated to this report, we also give some basic statistics about the data (mean, standard deviation, minimum values…). We also plot some histogram to visualize if the data presents some outliers or not. Based on the histograms, we can conclude that there are no particular outliers that we would like to remove. Besides, before the training, we won’t forget about normalizing the data to a range of 0 to 1 because normalization is important when we have different scales.

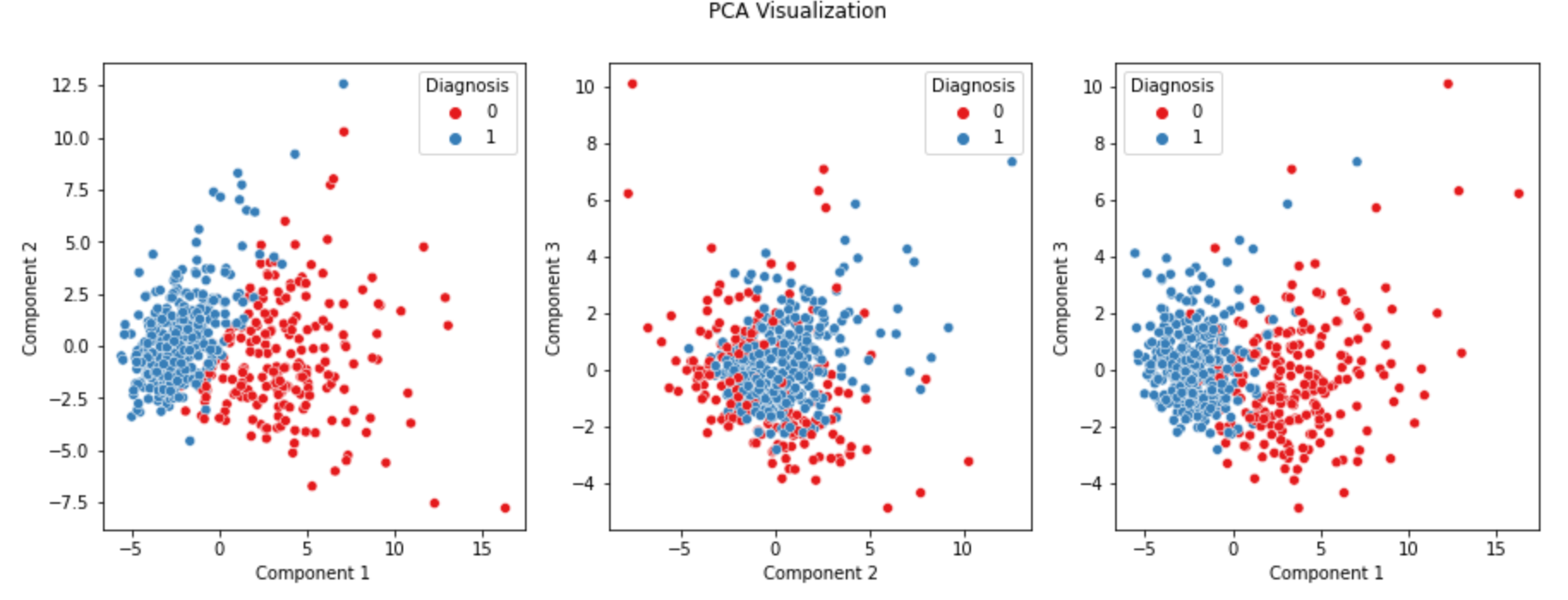




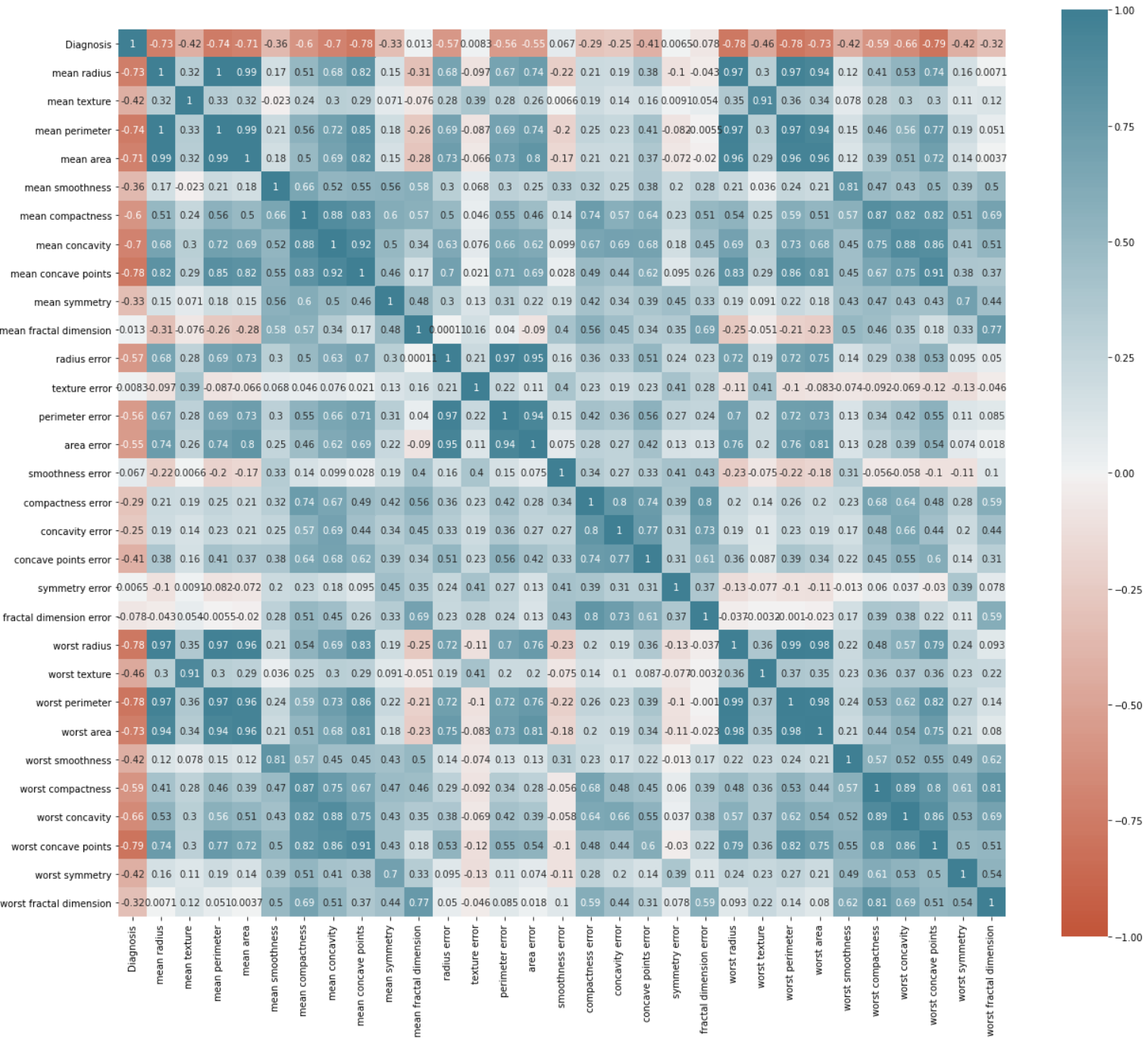
*Histograms of the 30 features and the target (called ‘Diagnosis’)*

We also compute the correlation between each feature and the correlation between the target feature and the rest of the features. We plot the correlation. We can see that some features are very correlated between them such as area, perimeter and radius which makes sense because they depend on each other. It is the case for the three features types: the mean, the standard error and the worst measures. We also have a strong correlation between the target feature (called ‘diagnosis’ in the correlation matrix) with the radius, perimeter, area, concavity and concave points features.

Now, we plot the results of the principal component analysis (PCA) on the dataset. We have chosen to have a number of 3 principal components. We did the PCA to visualize the dataset and assess if it can be linearly separable. We plot three scatter graphs of the dataset according to the three principal components. We can see that the dataset is linearly separable especially in the first and second graph.



*PCA with 3 components results on the dataset. Blue points are benign cells and red points are malignant cells.*



*Heatmap of the correlation matrix between every feature and the target.*

# Prediction

**Models**

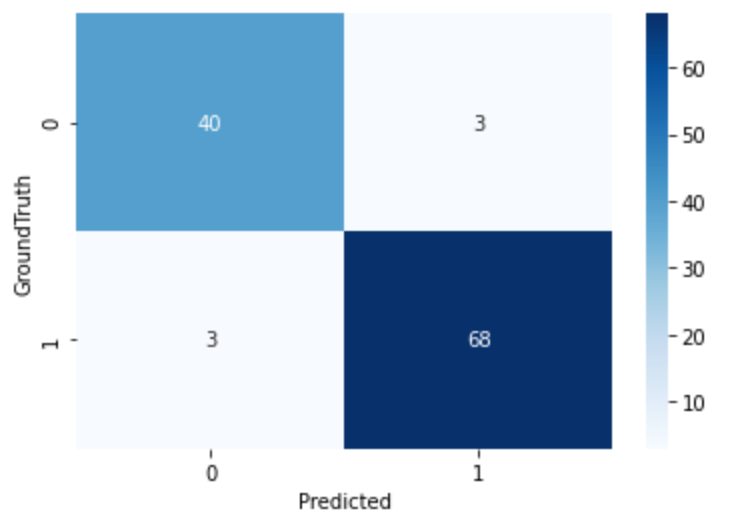
In this study, we compare 5 classifier models:

* Logistic regression
* Decision tree
* KNN
* Random forest classifier
* Support vector machine classifier

We decided to use these 5 machines learning methods to compare them and to assess which one is the best. We train the 5 models on 80% of the data and we evaluate them on 20% of the data. Every feature of the breast cancer Wisconsin dataset was normalized to have a range of 0 to 1 before fitting the data to every model. We present the metrics we used in our study.

**Metrics**

The classification is a binary classification. To assess the results of the predictions we plot a confusion matrix of the results. A confusion matrix is a table showing how many individuals are classified correctly or not, for each of the two classes. To understand well the metrics we used in our study, let’s look at a confusion matrix we got for the K-Nearest Neighbors (KNN) classifier model we trained:

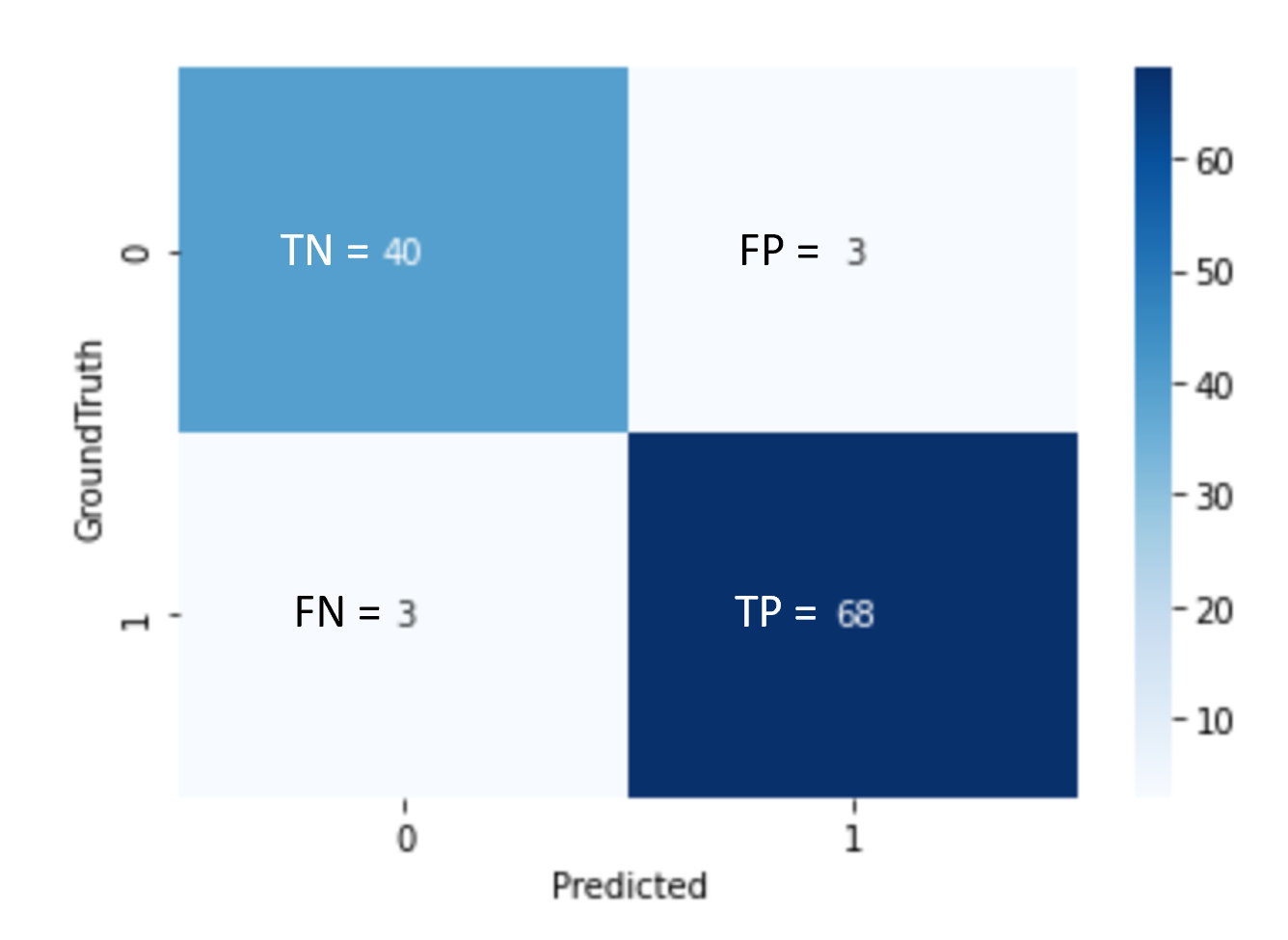


*Confusion matrix for the results of the prediction for the KNN model.*

In a binary classification there are 4 possible cases of results depending on the predicted value and depending on the actual value (ground truth value). In the case of breast cancer prediction:

* The model predicts the cell is **benign** (categorical class of the predicted value of **1**) and it is really a **benign** cell (categorical class of the ground truth value of **1**). In this case, the prediction is a **True Positive (TP).**
* The model predicts the cell is **benign** (categorical class of the predicted value of **1**) but in reality, the cell is a **malignant** cell (categorical class of the ground truth value of **0**). In this case, the prediction is a **False Positive (FP)**.
* The model predicts the cell is **malignant** (categorical class of the predicted value of **0**) but in reality, the cell is a **benign** cell (categorical class of the ground truth value of **1**). In this case, the prediction is a **False Negative (FN)**.
* The model predicts the cell is **malignant** (categorical class of the predicted value of **0**) and it is really a **malignant** cell (categorical class of the ground truth value of **0**). In this case, the prediction is a **True Negative (TN)**.

We can see these values in the confusion matrix we showed before.



*Confusion matrix for the results of the prediction for the KNN model with TP, TN, FP and FN indications*

Based on these values, we can compute several useful metrics to assess our results. We first compute the accuracy. It is the number of correct predictions divided by the total number of predictions. Accuracy quantifies the correctness of our predictions.

We then compute the precision of our model. Precision metric is defined by the following formula:

Precision metric shows how accurate the model is on the predicted positive class (in the case of our prediction the positive class is the benign cells). In the case of our study, this metric is very important. If a cell is predicted as a benign cell but it is in fact a malignant cell (it is a false positive), it can have dangerous consequences.

We also compute the recall metric. Recall metric is defined by the following formula:

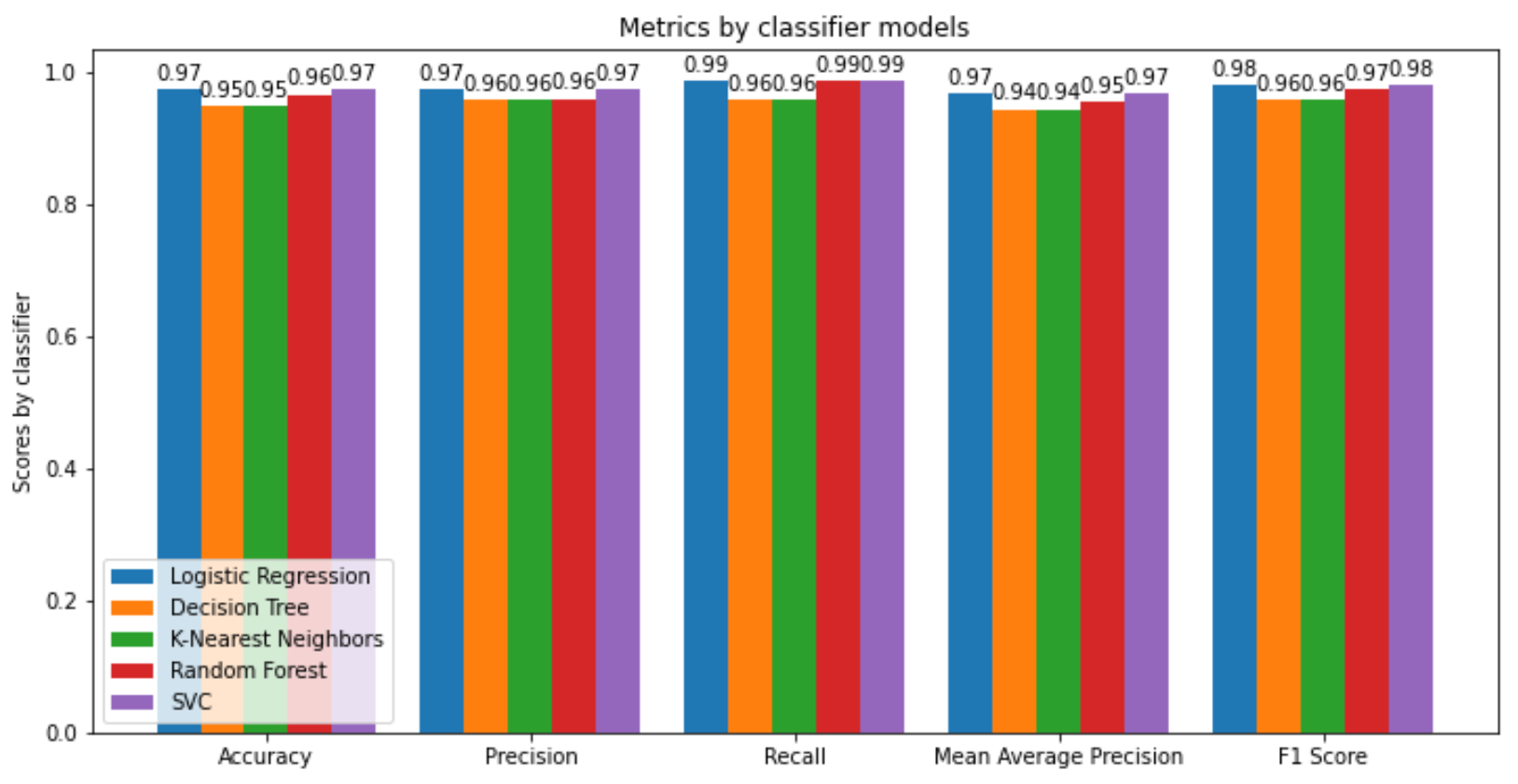
Thanks to this formula, we can understand that the recall metric calculates how many of the actual benign cells our model captures through labeling them as benign cells.

We then compute the mean average precision (mAP). This metric is computed thanks to the values of precision and recall metrics. It makes an average of these two metrics. We finally compute the F1-score of our predictions. F1 is a function of precision and recall metric. F1 Score is useful to seek a balance between precision and recall.

These 5 metrics will be used to assess which model is the best model.

**Comparison of the results**

After training every one of the 5 models, we have computed the metrics on the test dataset for every one of the 5 models. Each model was trained on the same training data and was evaluated on the same test data. The repartition of the dataset was of 445 observations for the training dataset and of 114 observations for the test dataset. Among these 114 observations, there are a total of 71 benign observations and 43 malignant observations. Here are the results we manage to get with these evaluations.



*Histograms of the obtained metrics for every model*

Based on this histogram, we can see that the two best models are the logistic regression and the support vector machine classifiers. Both have the best metrics values. These models were trained with the default parameters from sklearn package. Now, we will do hyper parameters tuning for each model and we will use cross-validation on 10 folds to validate the results of the predictions.

**Hyper parameters tuning and cross-validation**

For each of the 5 models, we present the parameters we have tuned to get the best results. We explain why we have used them for each model. The notebook illustrates the different parameters we have used.

Logistic Regression

Logistic regression is an iterative method. One parameter is the maximum number of iterations. The more it is, the longer the training will be. The model could converge in better estimations with a higher number of iterations.

We also tune the hyper parameters regulating the regularization. The regularization factor enables the model not to overfit to the training data. A big value means a bigger regularization. Regularization makes the model generalize better to new data.

Logistic regression method estimates parameters of a mathematical function with the use of a solver function. In sklearn, we can specify several solver methods as parameter to the logistic regression model.

Decision Tree

For the decision tree classifier, we specify different numbers for the maximum depth of the tree. The higher the depth of the tree is, the higher the classifier will overfit to our data. We use different values to find which maximal depth fits the best to our data. We also use other parameters to modify the structure of the tree to find the best model.

K Nearest Neighbors

For the KNN classifier, a parameter to fine tune is the number of neighbors used in the model for neighbors’ queries. Increasing the number of neighbors improves the test scores. Having a lower number of neighbors usually means in overfitting our training data.

Weights function used in the prediction can be modified. Weights can be either uniform (points in each neighbor are weighted equally) or can be dependent to the distance of each neighbors to the query point. The distance function metric can also be modified. It can be either an Euclidian distance or a Manhattan distance (L2 or L1 respectively).

Random Forest

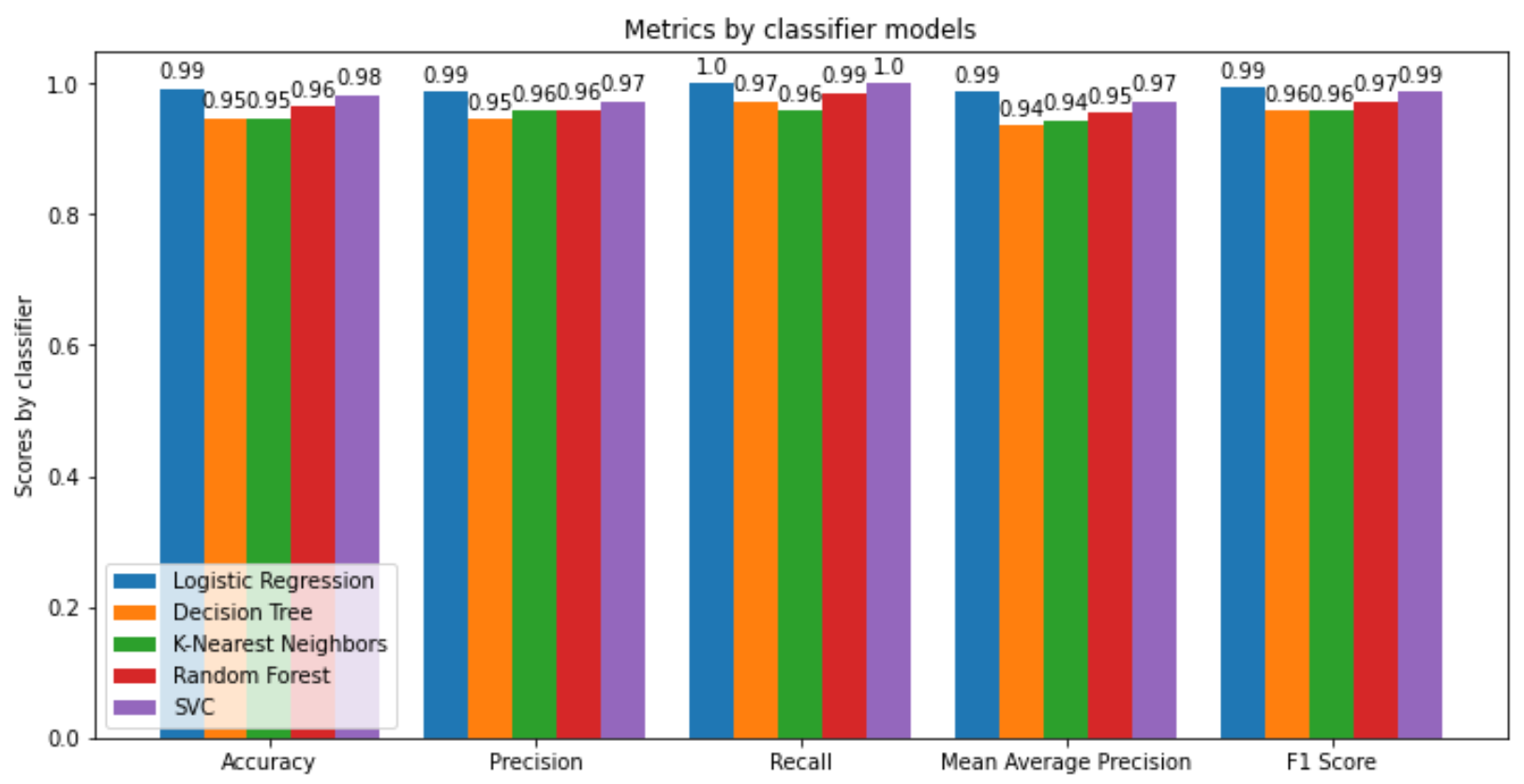
The random forest fine tuning is very similar to the decision tree fine tuning. Basically, a random forest is composed of a certain number of decision trees. Therefore, we have similar parameters such as the maximum depth of each of the trees in the forest. Similarly to the decision tree, these parameters will influence overfitting during our training phase. We also have the number of trees and the number of features to consider during the decision as parameters to fine tune.

Support Vector Machine

Support vector machine is one the famous classification model in machine learning. As with the previous methods, we avoid overfitting by fine tuning a regularization parameter. We give him various values. We also fine tune another parameter responsible for how quickly the boundary dissipates.

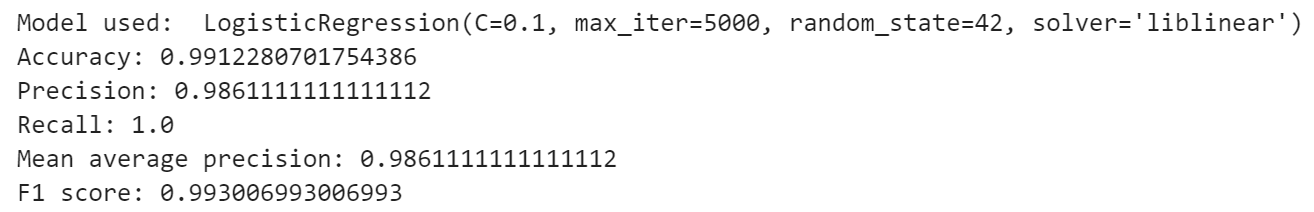
Results

The cross-validation was done on 10 folds. We present in the histogram below the metrics we got for each of the best model resulting from the cross-validation and the hyper parameter fine tuning.



*Histograms of the obtained metrics for every model after fine tuning and cross-validation*

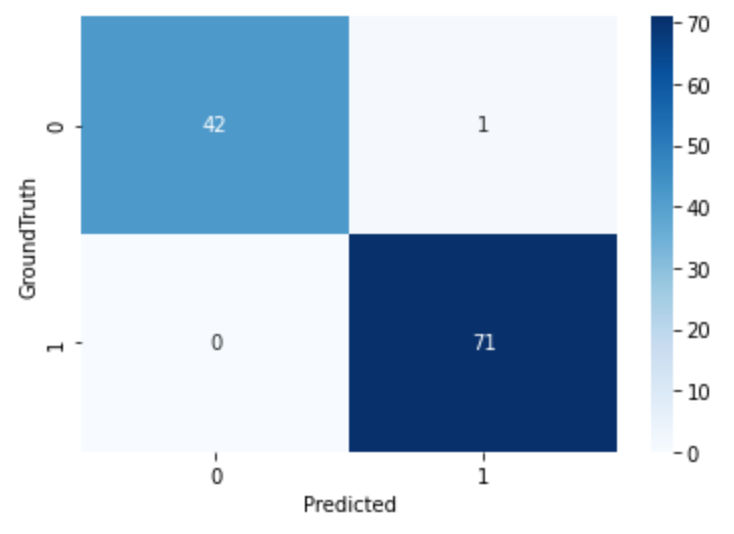
With this histogram, we can see that every model has improved its performances during the evaluation. The best model is the logistic regression model with the following parameters and with the following performances.



We manage to get 99% of accuracy on the prediction of breast cancer which is very good. There is a little room to improve the results. An approach to improve the results may be to combine the models of logistic regression and of support vector machine which are the two best models. Neural networks could be an approach to investigate also.

# Visualization

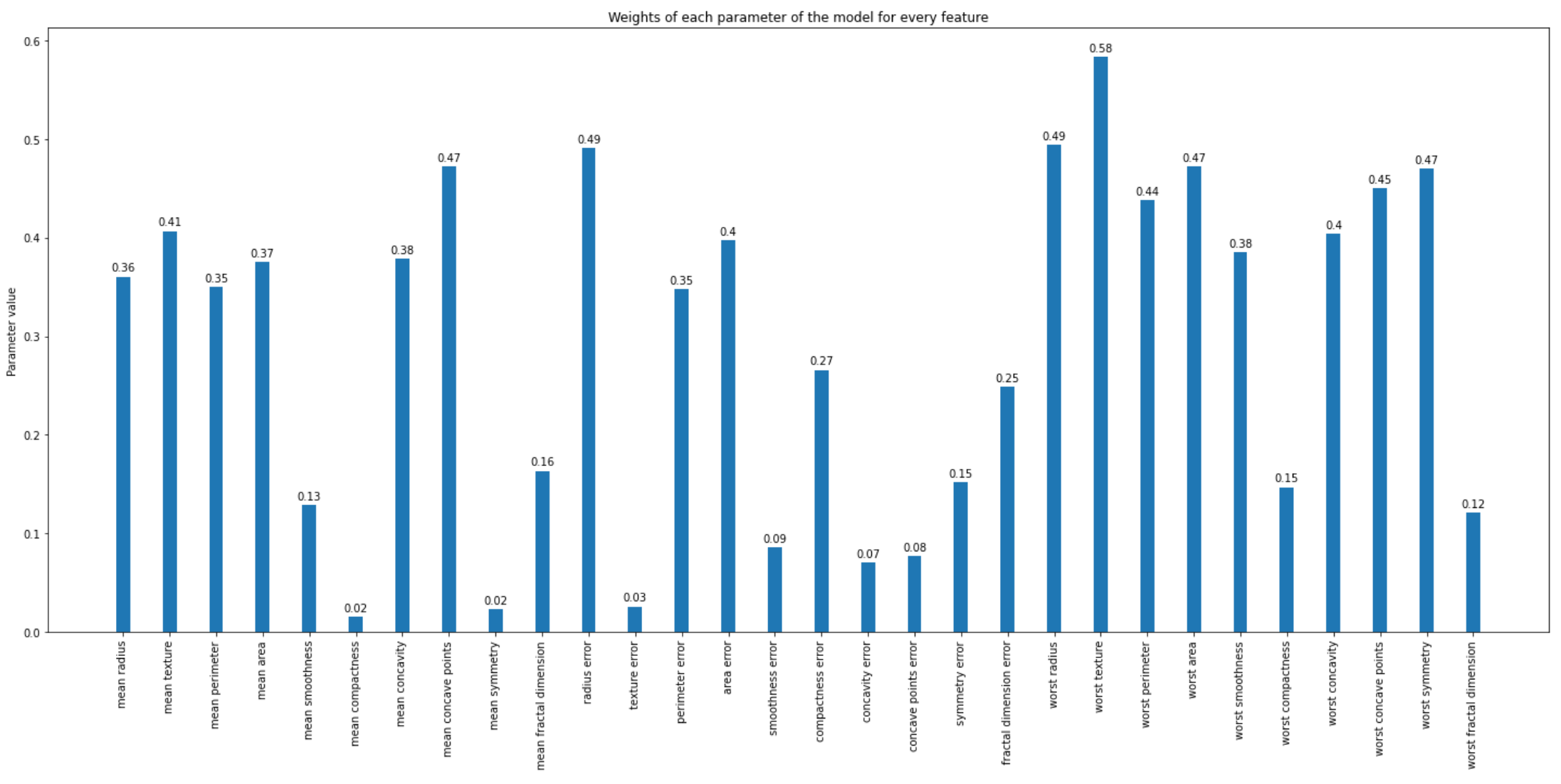
In this part, we visualize the results we obtained thanks to our logistic regression model. We also show the weights of each features in the logistic regression model. Here is the confusion matrix we got for logistic regression giving us an accuracy of 99%.



*Confusion matrix obtained for the best logistic regression model*

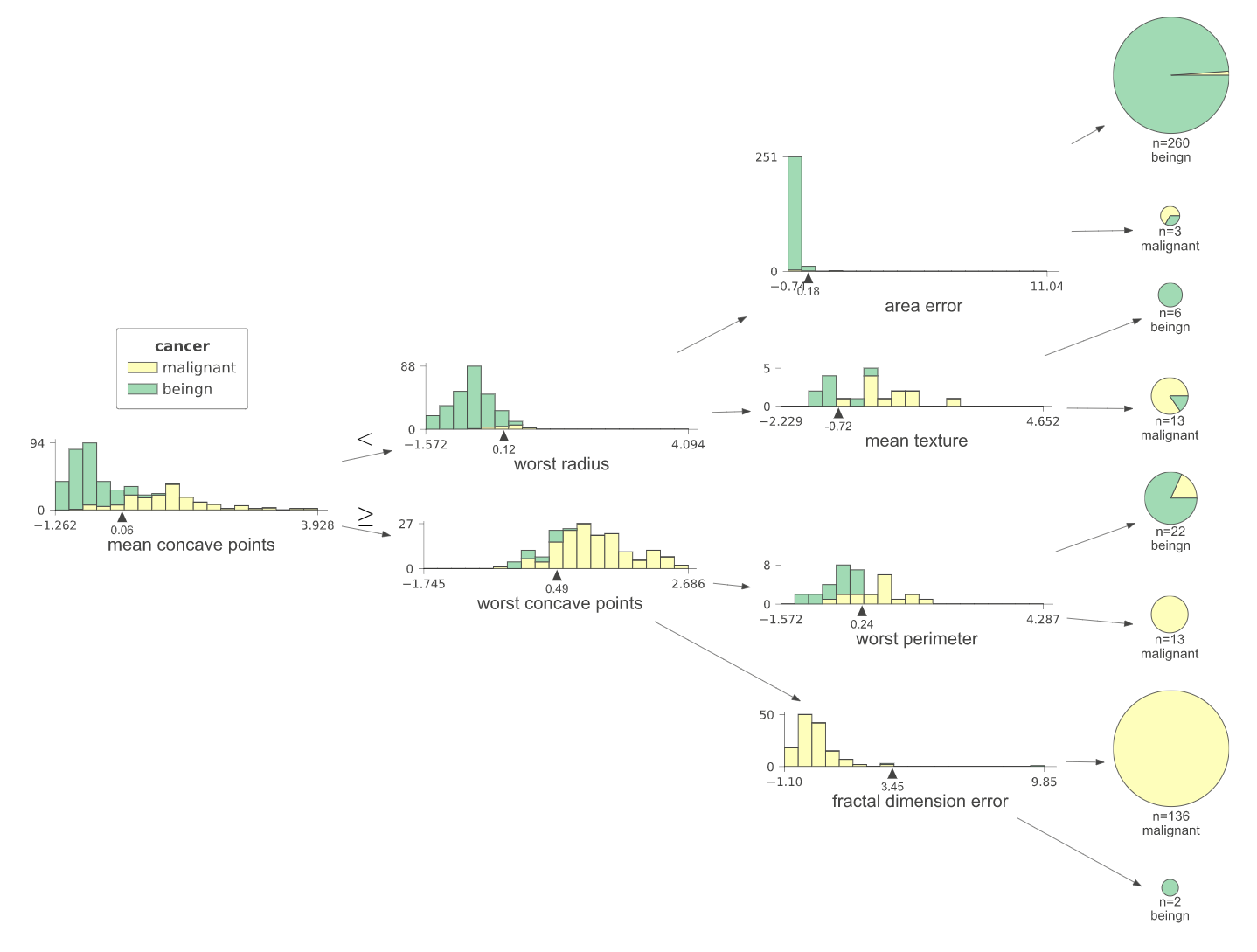
Among the total of the 114 observations test dataset, only one observation was misclassified by our model.

Then, we plot the extracted knowledge from the logistic regression model by plotting a histogram of the weights of each feature in the model. Based on this histogram, we can see that features like worst radius, worst texture, worst area, worst perimeter, radius error and mean concave points are the features with the most impact on our model. These features make sense because they were very correlated to the diagnosis target feature as we show in the analysis of the dataset part.



*Histogram of the most impactful features on the logistic regression model*

Finally, we also show the best decision tree model we had after the fine tuning and the cross-validation part. We used the *dtreeviz* to visualize the decision tree.



*Best decision tree model representation after fine tuning and cross-validation*

We can see that the features used in the decision tree corresponds to the most impactful features of the logistic regression model which is coherent.

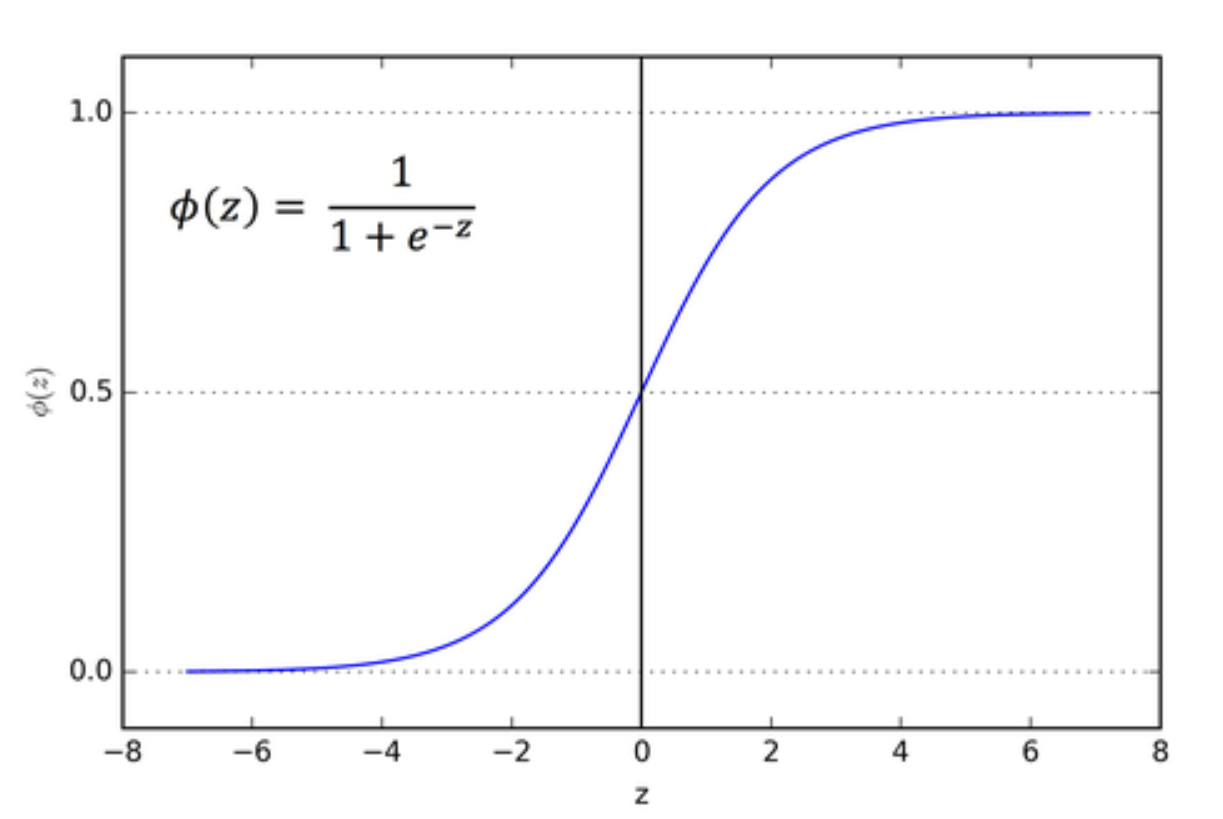
# Theoretical details

In this part, we present in an in-depth manner the logistic regression machine learning method we had selected for the prediction of breast cancer dataset.

Logistic regression is a supervised machine learning method used to predict whether something is true or false. It is used for binary classification. In the case of our dataset, it is used to predict if a cell nuclei is benign or malignant.

The term logistic refers to the ‘log odds’ probability that is modeled. The odds are the ratios that compare the number of ways the event can occur with the number of ways the event cannot occur. Mathematically, it is defined as follow:

Logistic regression returns a prediction as a probability. It is the conditional probability that the prediction y is equal to the positive class given the features X. To be able to make predictions such that X can be defined in any domain and that y is between 0 and 1, we model this behavior by using a sigmoid function. Here is the curve of the sigmoid function:

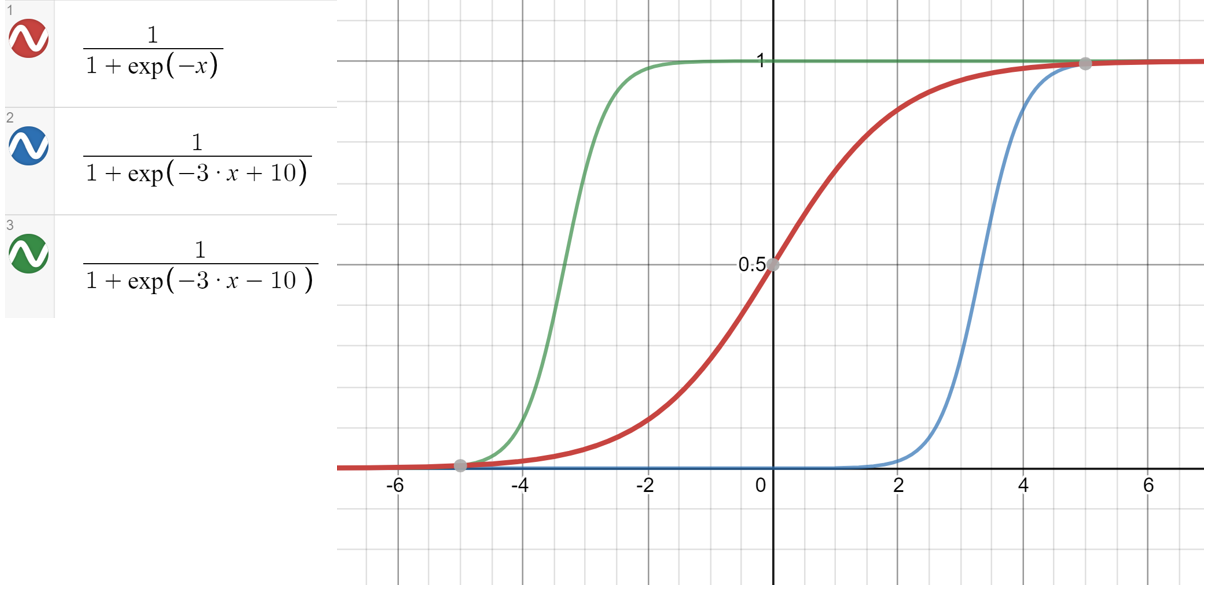


*Curve of the sigmoid function*

Based on this sigmoid function we can write:

Which is equivalent to:

With parameters of the sigmoid function to estimate. The parameter influences on the position of the curve on the horizontal axis in the graph whereas the parameter influences on the strength of the slope in the sigmoid curve. Here is an example of the influence of the parameters.



*Different curves of the sigmoid function showing the influence of the parameters*

The left term of the previous equation is the log of odds ratios. A graph of log odds against X gives us a linear graph. So, the goal is to estimate the two parameters. Similarly to the line regression method which fits a line to the data, logistic regression will fit this sigmoid function to the data. The estimated parameters in logistic regression are the and parameters. Logistic regression uses maximum likelihood to estimate these parameters. How does this work?

To accomplish maximum likelihood, training data is split in two groups based on their labels. For samples of the positive class (y = 1), the goal is to estimate and such that is as close to 1 as possible. Similarly, for samples of the negative class, the goal is to estimate and such that is as close to 1 as possible. In other words, if we take every sample for the positive class, we want the product of their probability to be as close to 1. If we take every sample for the negative class, we want the product of their probability to be as close to 1.

Mathematically, we have a function to maximize:

It is the maximum likelihood function. The goal is to find to maximize this function. This function cannot be solved mathematically, but it can be solved with numerical methods. There are several methods to do so. After a number of iterations, the function will converge to values.

To conclude, logistic regression returns a probability as a prediction. By specifying a threshold, logistic regression classifies the prediction in two categories. For example, if the probability is lower than 0.5 then the prediction will be the negative class.

With the mathematics behind logistic regression, we can understand some input parameters of the method:

* Number of maximum iterations: the maximum number of iterations before stopping the estimation of the parameters. Increasing it will cause a longer training. Having one too small will cause a non-convergence of the parameters which will cause wrong predictions.
* The solver parameter: it is the algorithm to use in the estimation of the and .
* A regularization factor so that the model does not overfit to the training data. A big value means a bigger regularization.

# Conclusion

To conclude, we have managed to have a logistic regression model working with 99% of accuracy on the breast cancer dataset after comparison to other classification machine learning models. We now understand the mathematical formalism behind the logistic regression method which is a very interesting machine learning method for binary classification. After some literature research, we have seen that our results are very similar to the results in the literature. Different methods were used to get similar results (neural networks, SVM, random forest…)