Universidad
Carlos III de Madrid

PROGRAMMING IN R

Mortality outcomes for females suffering myocardial infarction

Students: Ilán Francisco Carretero Juchnowicz
Sofía Sorbet Santiago

Teacher: Juan Miguel Marin Diazaraque

Academic course: 2020/2021

Contents

List of Tables

List of Figures

Chapter 1

Descriptive study of the dataset

In this chapter, a descriptive analysis will be carried out from a certain dataset in which multiple information will be extracted and analyzed from various tables and graphs. In this way, it is intended to achieve two main objectives: firstly, to become familiar with R to do any initial data analysis and secondly to show the power of this free software for this task.

Note: the code created with which all the results present in the following work have been obtained can be found in the following github repository with the name of "part_1_code.R"

`https://github.com/ilancarretero/MSDS/tree/main/programming_in_R_project`

1.1 Introduction to the database

To carry out this work, the R Data Analysis and Graphics Data and Functions (DAAG) software package was used. This package is made up of several data sets of different kinds used by Maindonald & Braun in their book "Data Analysis and Graphics Using R"[1]. In turn, due to our biomedical interest, the database corresponding to Mortality Outcomes For Females Suffering Myocardial Infarction (`mifem`) has been selected. The main information regarding this dataset can be found in R documentation [2] and it is worth mentioning that this is a female subset extrated from the Monica Project [3].

In this way, this dataset consists of a total of 1295 rows and 10 columns, where such columns, that is, the variables are the following:

Name	Description	Type	Levels
outcome	mortality outcome	factor	live, dead
age	age at onset	numeric	
yronset	year of onset	numeric	
premi	previous myocardial infarction event	factor	y (yes), n (no), nk (not known)
smstat	smoking status	factor	c (current), x (ex-smoker), n (non-smoker), nk (not known)
diabetes	diabetes disease	factor	y (yes), n (no), nk (not known)
highbp	high blood pressure	factor	y (yes), n (no), nk (not known)
highcol	high cholesterol	factor	y (yes), n (no), nk (not known)
angina	chest pain	factor	y (yes), n (no), nk (not known)
stroke	cell death caused by poor brain blood flow	factor	y (yes), n (no), nk (not known)

Table 1.1: Name, description and type of all the variables involved in the mifem database

1.2 Descriptive analysis

1.2.1 initial observations

The first thing to deal with when studying a database is making a descriptive analysis. To do this, first of all we are going to visualize what our data set looks like, as observed in Table 1.2:

	outcome	age	yronset	premi	smstat	diabetes	highbp	hichol	angina	stroke
1	live	63.00	85.00	n	x	n	y	y	n	n
6	live	55.00	85.00	n	c	n	y	y	n	n
8	live	68.00	85.00	y	nk	nk	y	nk	y	n

Table 1.2: First view of the mifem database

As we can see in Table 1.1 only have two continuous variables, the age of the woman suffering the myocardial attack, and the year of the acquisition of the information (`age`, `yronset`). In this way, to initially analyze these variables, frequency tables with their corresponding histograms have been made.

The following results have been obtained for the `age` variable:

	Class limits	f	rf(%)	cf(%)
1	[35, 40)	16.00	1.24	1.24
2	[40, 45)	32.00	2.47	3.71
3	[45, 50)	60.00	4.63	8.34
4	[50, 55)	107.00	8.26	16.60
5	[55, 60)	214.00	16.53	33.13
6	[60, 65)	371.00	28.65	61.78
7	[65, 70)	495.00	38.22	100.00

Table 1.3: Frequency data (absolute value), relative frequency data (%) and cumulative frequency data (%) for the `age`

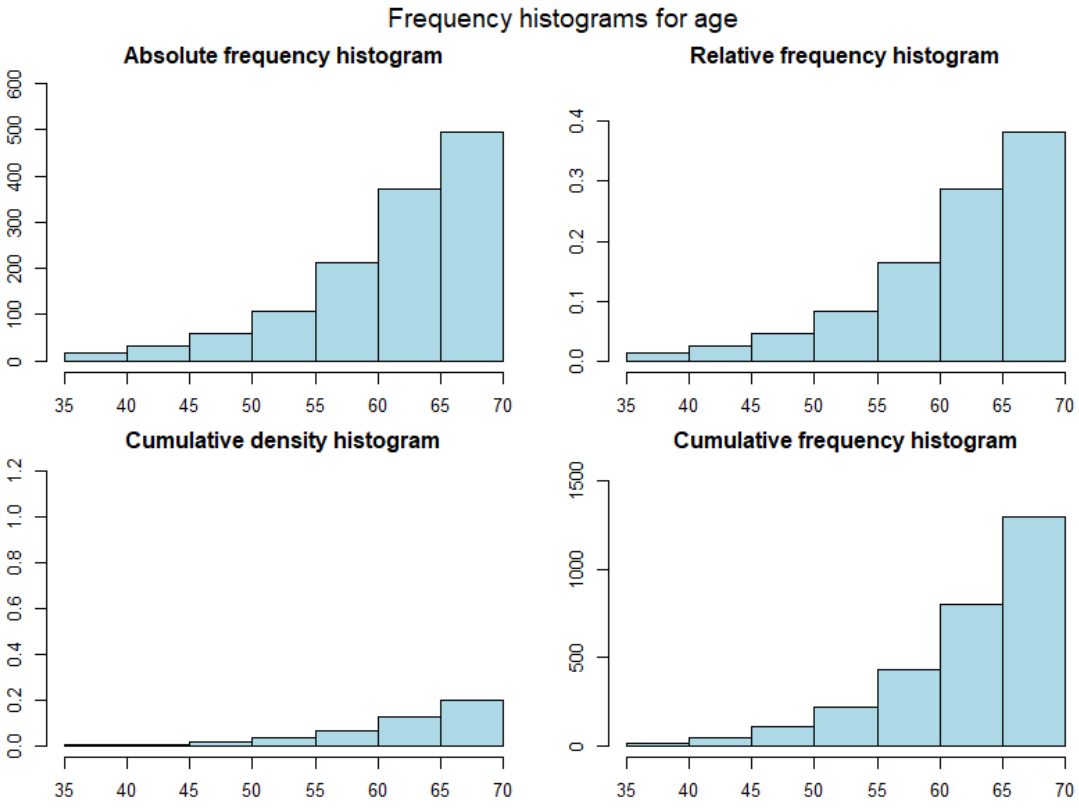


Figure 1.1: Frequency histograms for `age` variable.

These results show that the most part of the women that has suffered from a myocardial attack belongs to the interval 65-70. There is obviously a relation between the incidence of the myocardial attack and the age. Less than a 1.3% of the cases happen in the range of age between 35 and 40 years. This percentage gets higher as the interval of age gets greater.

We repeat the same procedure for the variable `yronset`, to see if the number of cases shown is the same in each year, obtaining the following results:

	Class limits	f	rf(%)	cf(%)
1	[84, 86)	153.00	11.81	11.81
2	[86, 88)	326.00	25.17	36.99
3	[88, 90)	300.00	23.17	60.15
4	[90, 92)	268.00	20.69	80.85
5	[92, 94)	248.00	19.15	100.00

Table 1.4: Frequency data (absolute value), relative frequency data (%) and cumulative frequency data (%) for the year of the event registration.

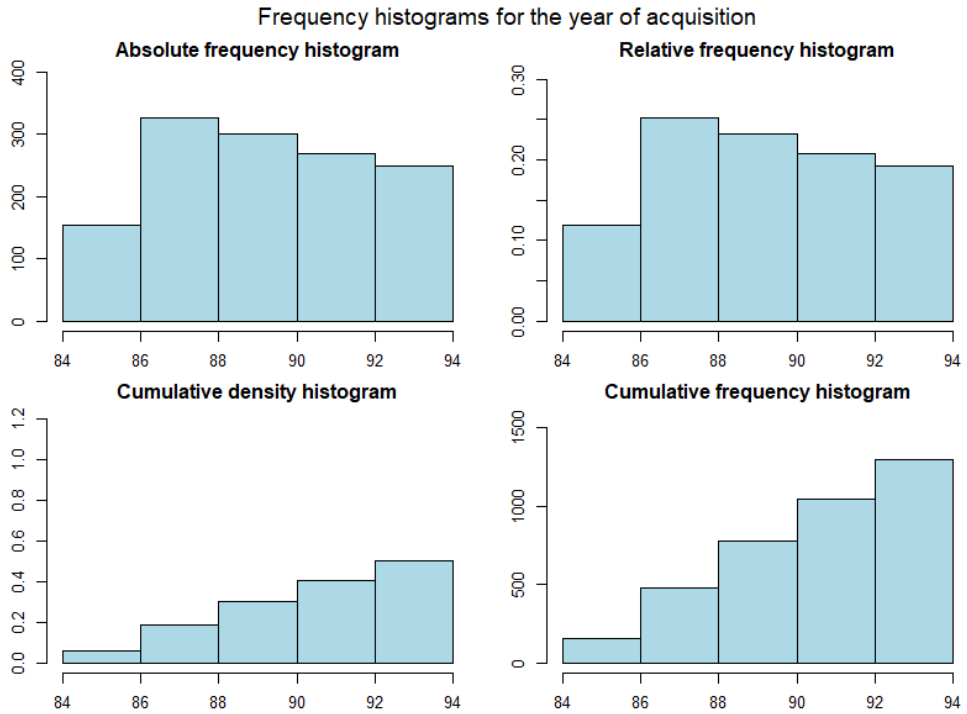


Figure 1.2: Frequency histograms for `year` variable.

Here we can see that the number of cases picked up belongs stable over the years. The differences between the lower number of cases and the higher one is less than a 14%.

We have also done a proportion table for the categorical variables, to know the interest proportion of the different categories. For the representation of the Tables 1.5 and 1.6 we have used bar plots as shown in Figures 1.3 and 1.4.

	Category	f	rf(%)	cf(%)
1	live	974.00	75.21	75.21
2	dead	321.00	24.79	100.00

(a) Outcome frequencies

	Category	f	rf(%)	cf(%)
1	n	928.00	71.66	71.66
2	y	311.00	24.02	95.68
3	nk	56.00	4.32	100.00

(b) Premi frequencies

	Category	f	rf(%)	cf(%)
1	n	522.00	40.31	40.31
2	c	390.00	30.12	70.42
3	x	280.00	21.62	92.05
4	nk	103.00	7.95	100.00

(c) SMSTAT frequencies

	Category	f	rf(%)	cf(%)
1	n	978.00	75.52	75.52
2	y	248.00	19.15	94.67
3	nk	69.00	5.33	100.00

(d) Diabetes frequencies

Table 1.5: Absolute and relative frequencies of categorical variables (1/2).

	Category	f	rf(%)	cf(%)
1	y	813.00	62.78	62.78
2	n	406.00	31.35	94.13
3	nk	76.00	5.87	100.00

(a) High blood pressure frequencies

	Category	f	rf(%)	cf(%)
1	n	655.00	50.58	50.58
2	y	452.00	34.90	85.48
3	nk	188.00	14.52	100.00

(b) High cholesterol frequencies

	Category	f	rf(%)	cf(%)
1	n	724.00	55.91	55.91
2	y	472.00	36.45	92.36
3	nk	99.00	7.64	100.00

(c) Angina frequencies

	Category	f	rf(%)	cf(%)
1	y	813.00	62.78	62.78
2	n	406.00	31.35	94.13
3	nk	76.00	5.87	100.00

(d) Stroke frequencies

Table 1.6: Absolute and relative frequencies of categorical variables (2/2).

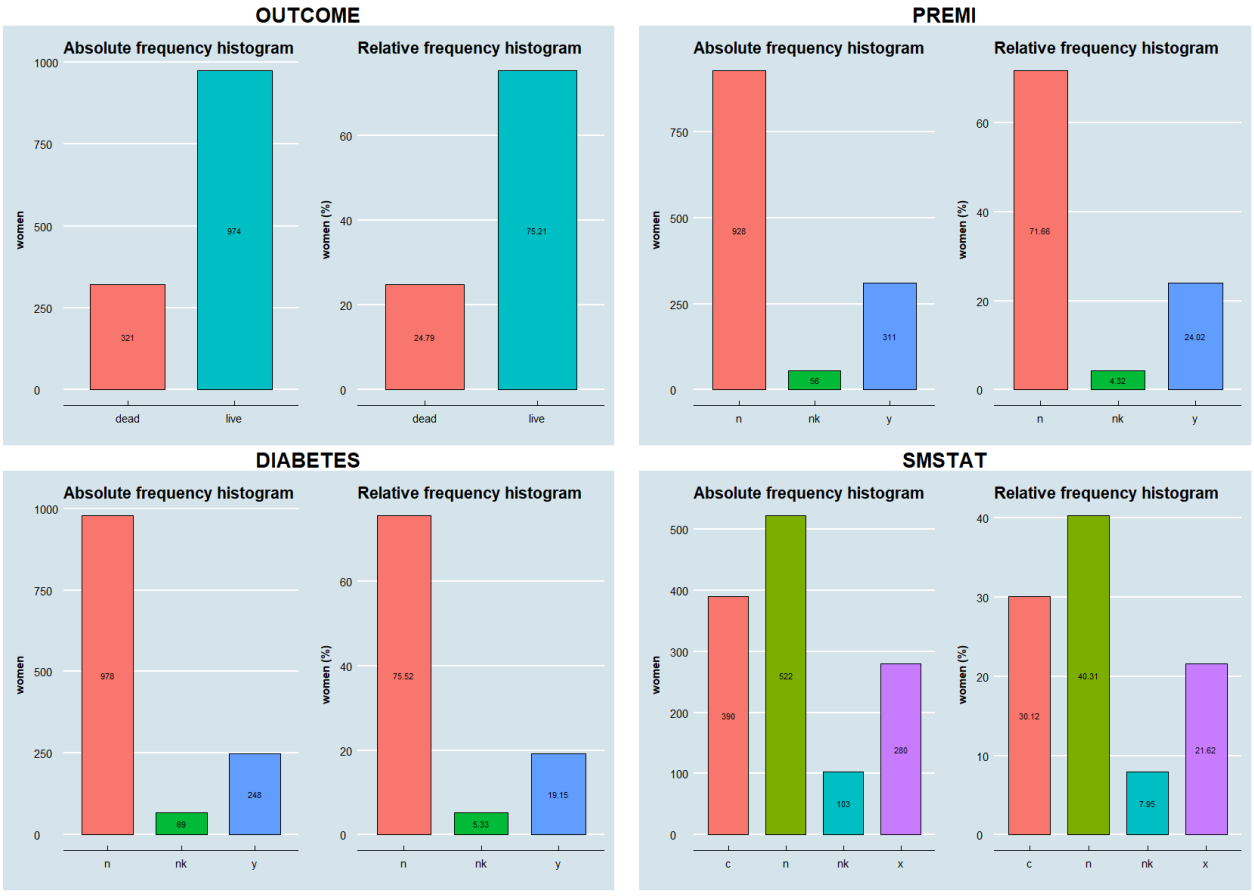


Figure 1.3: Bar Graphs for the absolute frequency of each category of each categorical variable (1/2).

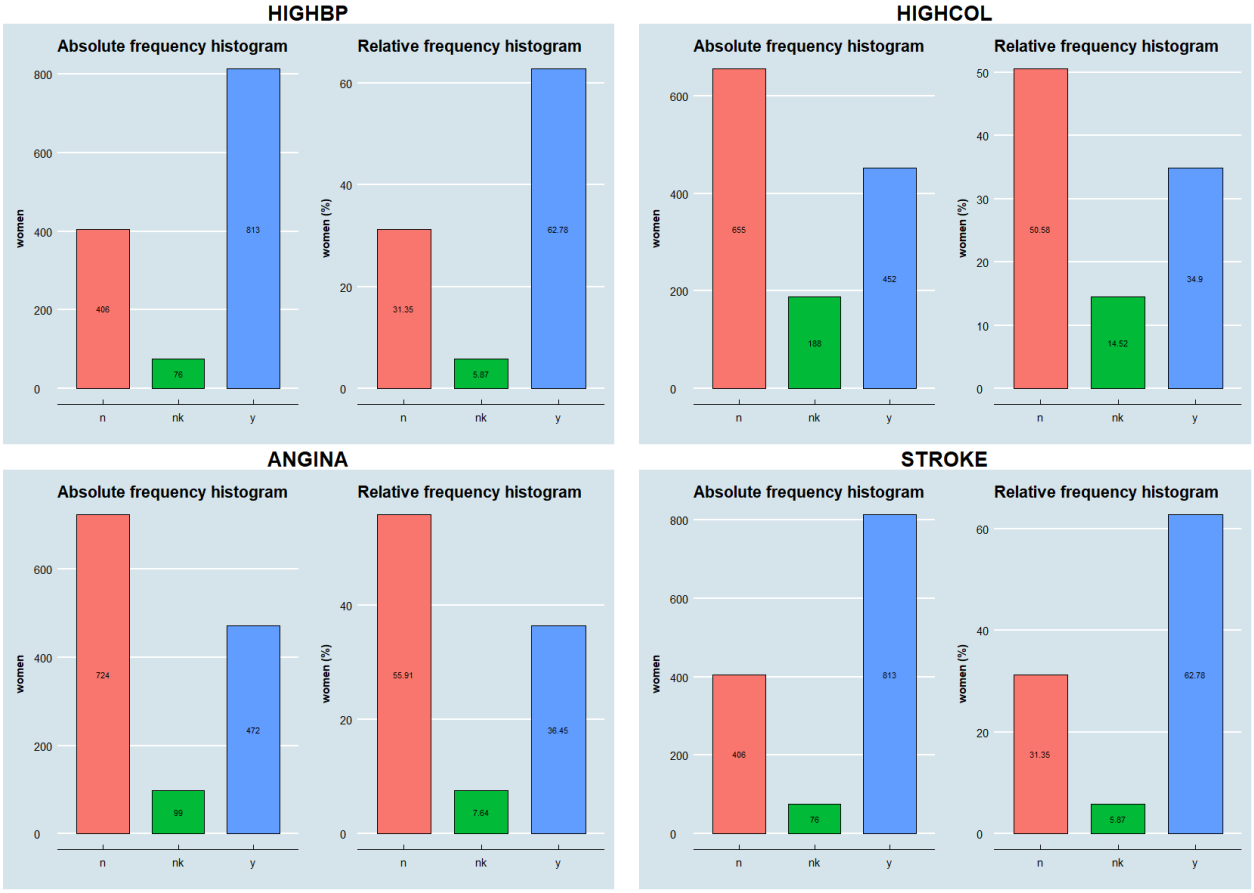
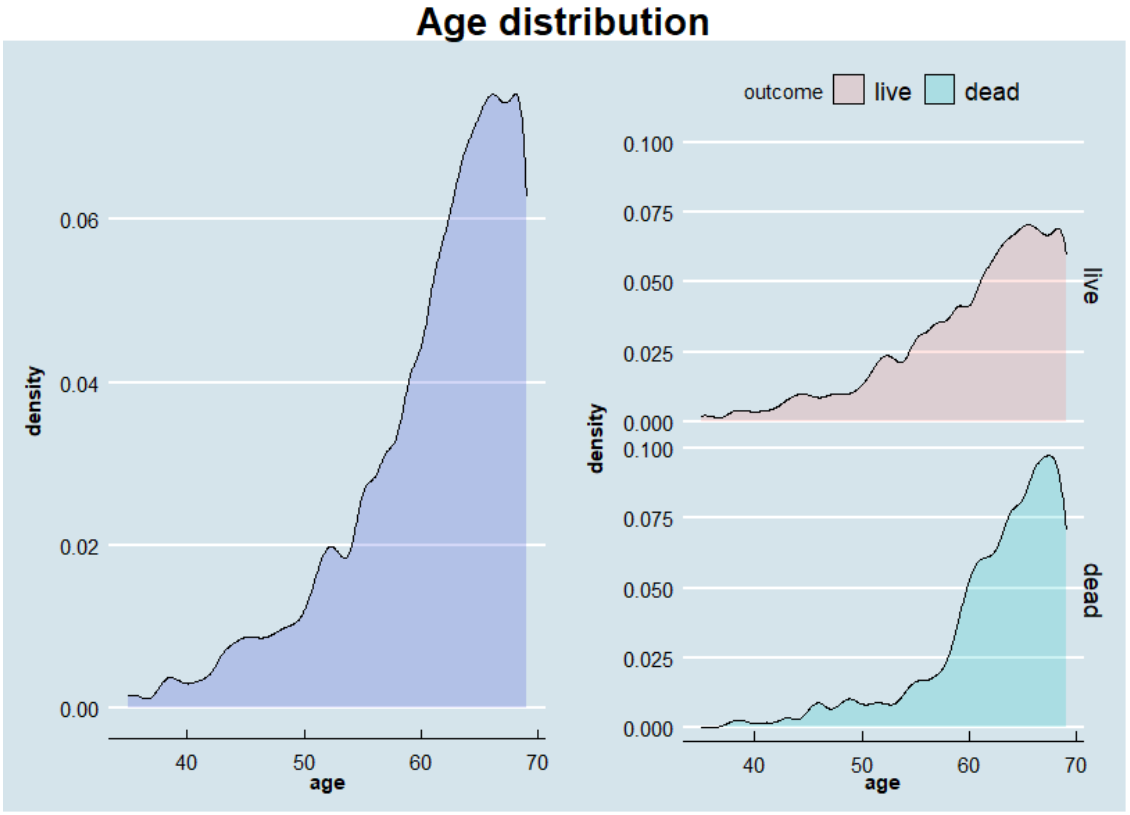


Figure 1.4: Bar Graphs for the absolute frequency of each category of each categorical variable (2/2).

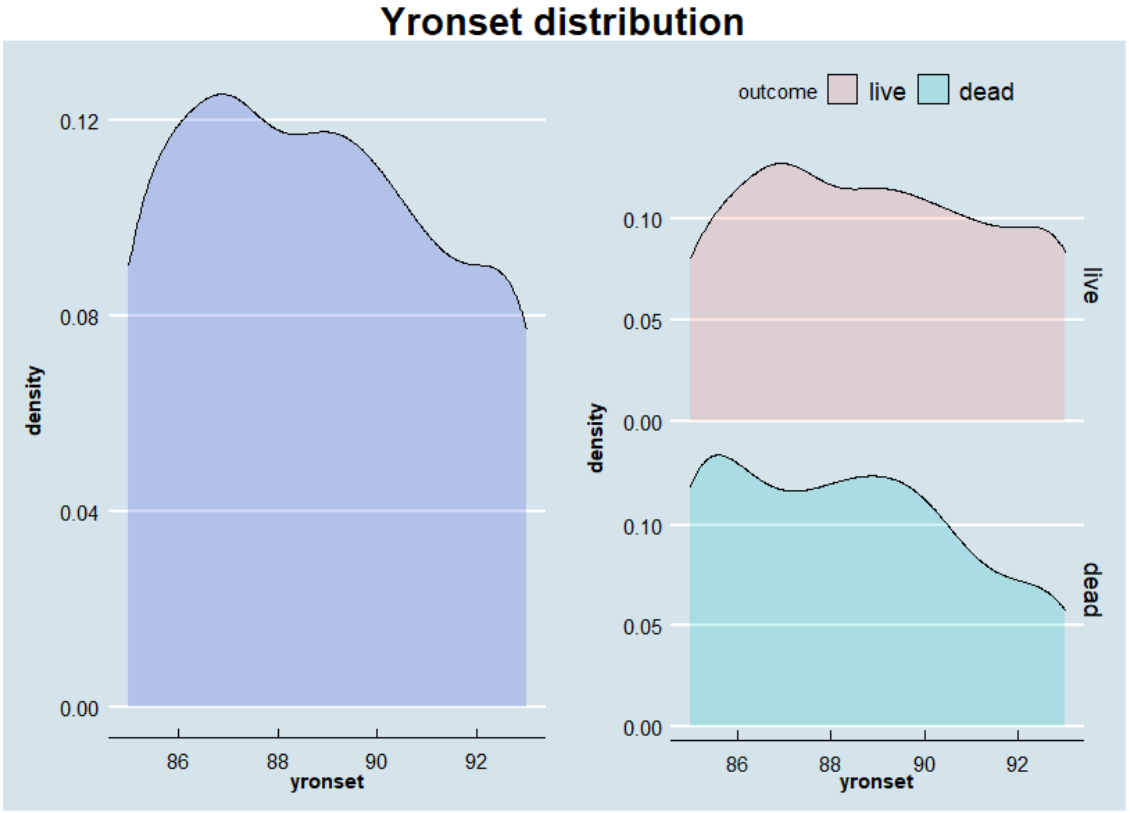
In this way, we can see that the most part of the women that suffered from a myocardial attack, had had a stroke previously. The most part of the women survived the myocardial attack. Moreover, the most part of them had high blood pressure.

1.2.2 Measures of centrality, variability, skewness and kurtosis

To analyze the measures of centrality, variability and shape of the continuous variables, a density plot of the sample as a whole was first performed and then separated according to the outcome variable. Thus, Figure 1.5 (a) shows what distribution the `age` variable follows, while Figure 1.5 (b) shows the distribution of the `year` variable.



(a) Distribution of the `age` variable



(b) Distribution of the `year` variable

Figure 1.5: Distribution of continuous variables.

Thus, seeing the distributions we can see how the age variable is skewed to the left to a great extent, and also when grouping this variable into dead or living women, both follow similar distributions. In addition, it can be seen how the distribution is not symmetric at all, which is quite logical since, as previously commented in the analysis of the frequency table, the majority of women in the sample are between 60 and 70 years.

Regarding the distribution of the registered years, it is much more uniform than the previous one, but it is slightly skewed to the left. Again, no large visual differences are perceived between the different levels of the outcome variable.

Once a first visual approximation has been made, we calculate the measures of mean, skewness, kurtosis and variance, obtaining the result that can be seen in Table 1.7.

	age	yronset
mean	60.92	88.79
variance	49.59	6.52
sd	7.04	2.55
skewness	-1.17	0.14
kurtosis	4.02	1.85

(a) Measures of age and year variables

LIVE	age	yronset
mean	60.41	88.91
variance	52.80	6.52
sd	7.27	2.55
skewness	-1.07	0.11
kurtosis	3.72	1.82

(b) Measures of age and year of live women

DEAD	age	yronset
mean	62.46	88.40
variance	36.82	6.35
sd	6.07	2.52
skewness	-1.50	0.25
kurtosis	5.26	1.94

(c) Measures of age and year of dead women

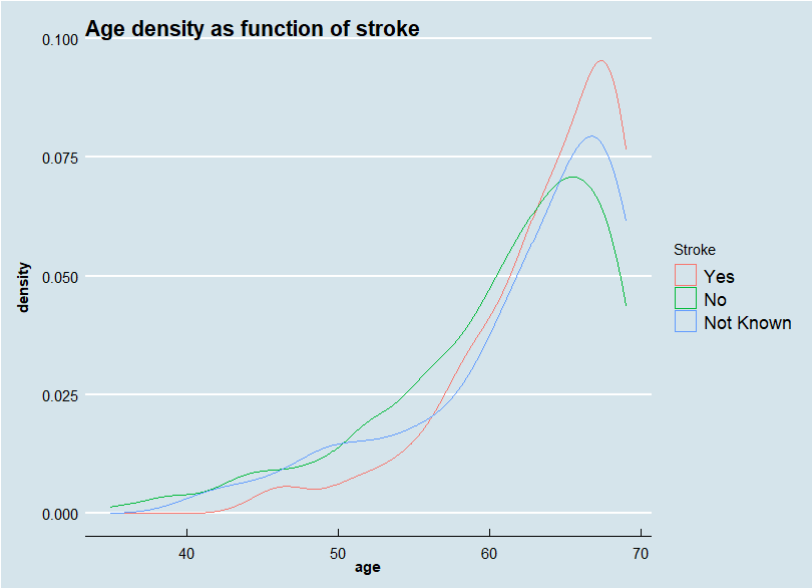
Table 1.7: Measures of mean, skewness, kurtosis and variance.

The first result is that the mean of age of the women with myocardial infarction in this study is 60.92. We can see that the variance in the age is not so great. It has a positive distribution, as we can see from the negative skewness, and it has a bit more kurtosis than the normal value 3, which means that it is a leptokurtic distribution, since its kurtosis is over 3, which means that it has fatter tails than the normal one. Likewise, in relation to the analysis of the age of women for living and dead women, it is appreciated that living women have a lower mean and greater variability than dead women. It should be noted that the kurotsis of the distribution of living women is closer to the normal value, while that of dead women is further away.

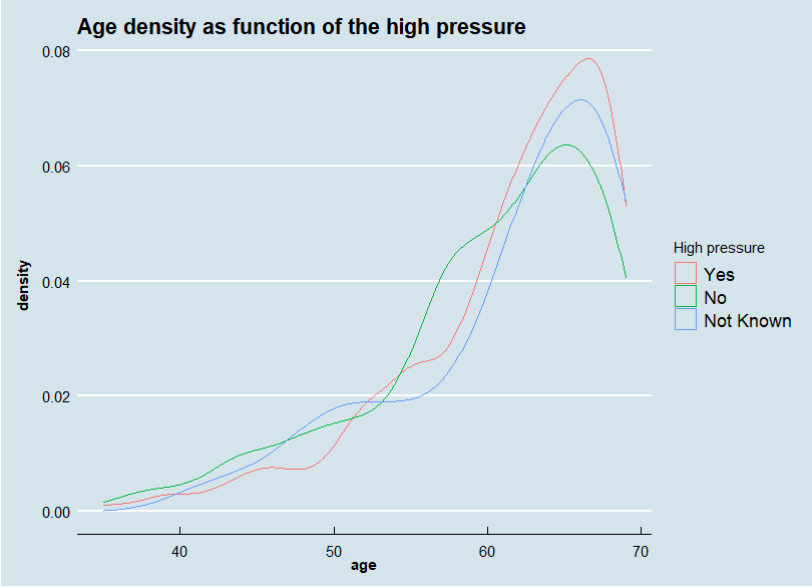
On the other hand, in relation to the variable years, it should be mentioned that the measures having been taken in each year approximately a similar number of measures, the distribution of this variable is quite uniform and with a much lower variability than that of age, being able to approximate because of the skewness and kurtosis values (close to 0) to a normal distribution.

1.2.3 Density plots, box-plots, bar plots, and histograms

As previously mentioned in the analysis of the different frequency tables, the majority of women who have suffered a myocardial infarction have high blood pressure and had suffered a stroke before. Therefore, below we are going to analyze the distribution of these variables as a function of age, as we can see in the Figure 1.6.



(a) Age density as function of stroke



(b) Age density as function of high blood pressure

Figure 1.6: Density distributions for stroke and high blood pressure variables

As we can see, the age distributions according to the different levels of stroke and high blood pressure are similar. Although it is true that we could have found another distribution, this was to be expected, since the majority of women who have suffered a myocardial infarction are between 60 and 70 years old.

To visualize whether there are statistically significant differences in age according to whether the women have died or not and whether or not they have suffered from angina, the following box plots have been represented.

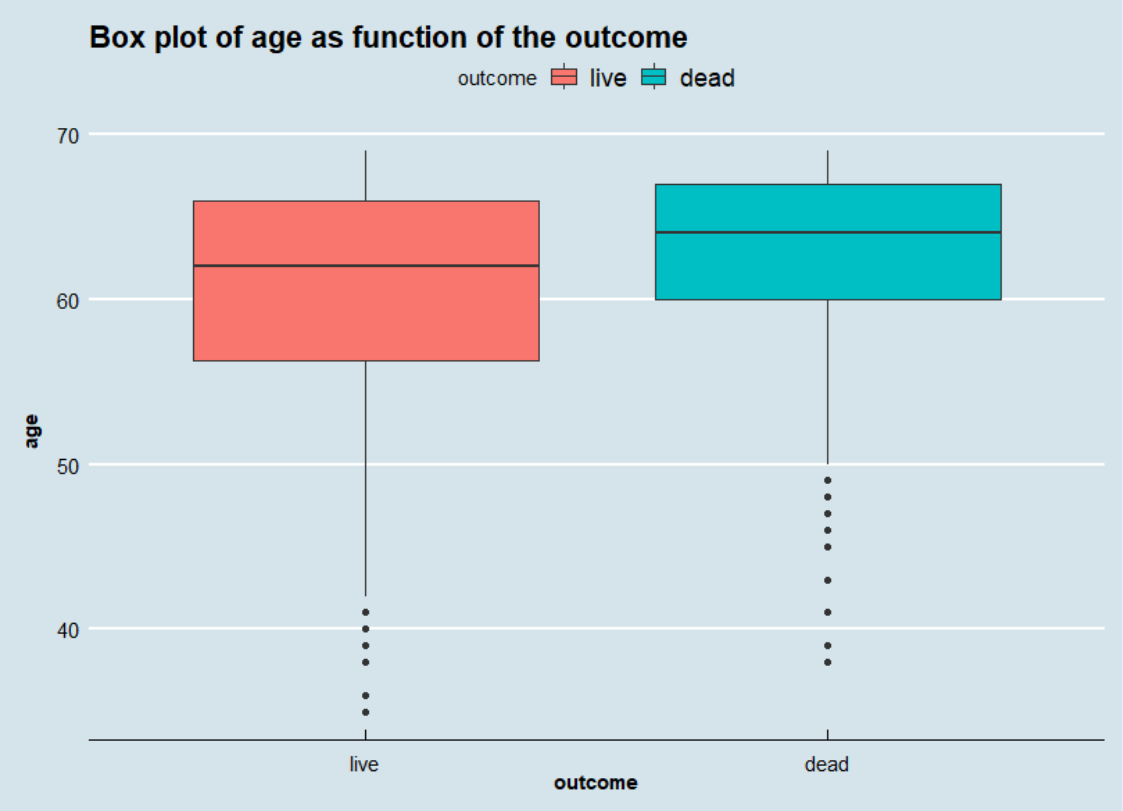


Figure 1.7: Box plot of age as function of the outcome.

As we can see in Figure 1.7, the differences between the ages of living and dead women are not statistically significant. Although it is true that the median of the female deaths is slightly higher than those of the living women, in relation to the variability of the respective samples, it is appreciated that the age of the living women is older, covering a greater range of ages. in the first quartile. It should be noted that the interquartile range of the age of dead women is lower, with a higher concentration around an age interval.

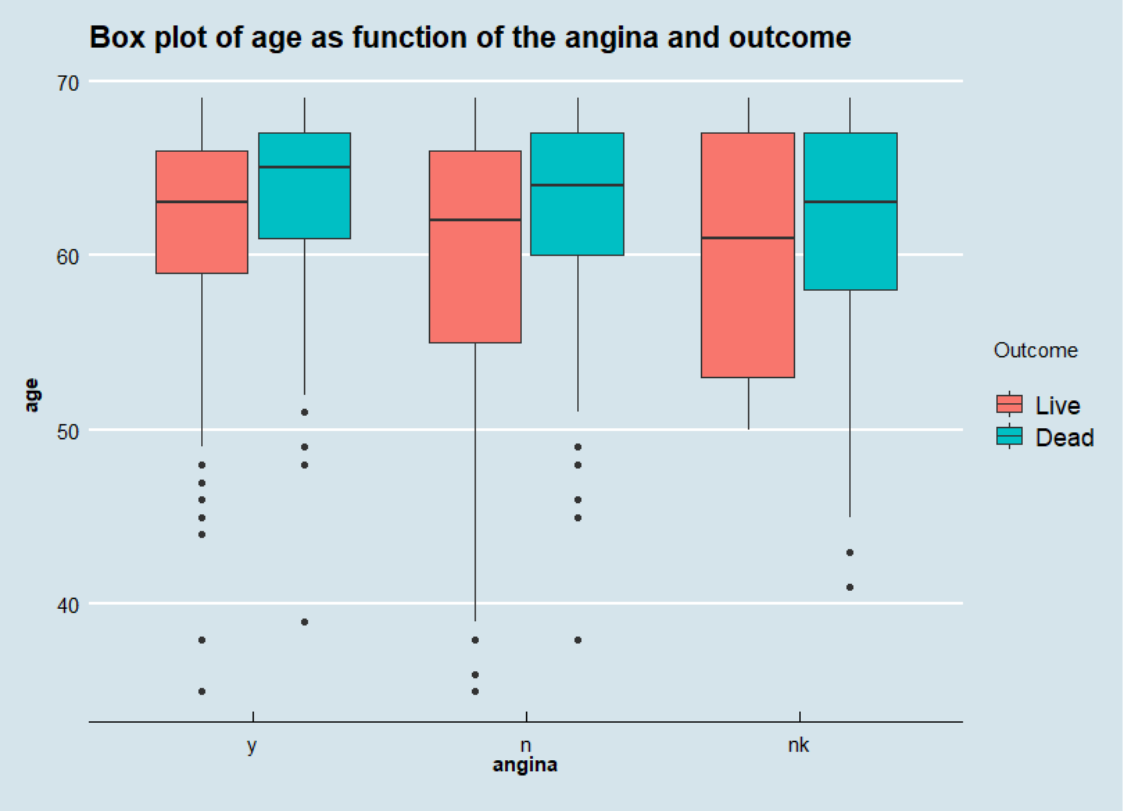


Figure 1.8: Box plot of age as function of the angina and outcome variables.

Regarding Figure 1.8, we again observe that there are no statistical differences between the ages according to the angina and outcome variables. However, it is interesting to note that the variability of those women who have suffered from angina is the smallest, while the greatest variability, as expected, corresponds to those women who do not know whether or not they have suffered from angina. As in the previous case, the median age of the dead women is greater than that of the living women at all levels of the angina variable.

In order to also illustrate a bar diagram, the number of living and dead diabetic women has been represented as seen in Figure 1.9.
In this way, we could obtain the following information from the represented bar plot:

- In the sample there is a greater number of women who are not diabetic.
- Looking at the chromatic separation, in the sample there is a greater number of living women than dead women.
- It should be noted that the proportion of dead women who do not know whether or not they are diabetic is much higher than in living women.

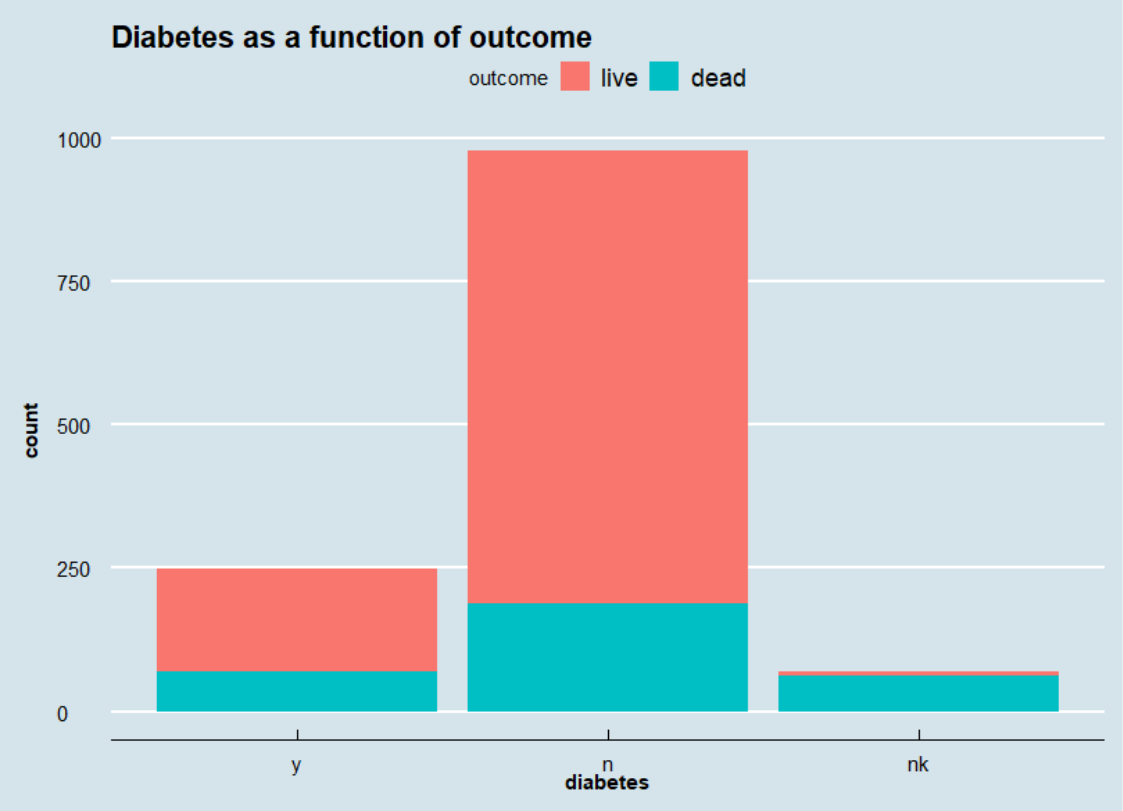


Figure 1.9: Diabetes variable as a function of outcome.

In the same way that a bar plot can be represented for a categorical variable (Figure 1.9), we can make a histogram for a continuous variable where in fact we can group according to another categorical variable. Thus, we obtain the Figure 1.10

In relation to this figure, we could extract the following information:

- Between 60 and 70 years a large part of the sample is found for the three levels of high blood pressure.

- Among women who do have high blood pressure, the age with the most recurrence by far is approximately 65 years.

- In the sample, there is a greater number of women with high blood pressure than women who do not. This is completely logical since the sample is from women who suffer a myocardial infarction

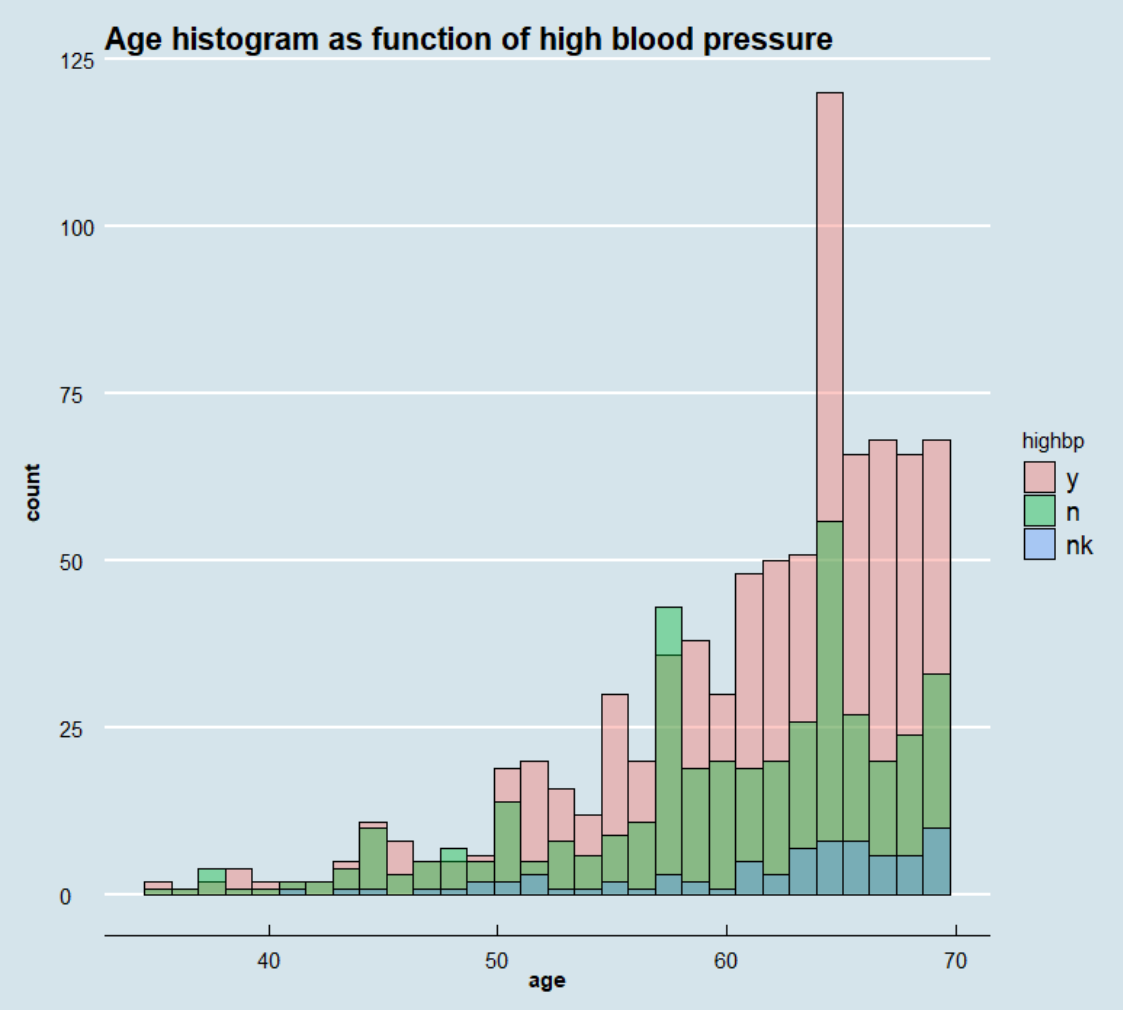


Figure 1.10: Age histogram as function of high blood pressure.

A plot that has been really interesting is the comparison of mean age of women depending on the year the myocardial infarction was registered. Thus, when separating the results according to living and dead women, the Figure 1.11 is obtained.

If we look at the image, we can see that the mean age of the living women is slightly lower than the mean age of the dead women. Likewise, it can also be deduced that from the registered cases, the variability of the mean age of the living women is lower than that of the dead women. While it is true that to make other types of conclusions it would be necessary to carry out more elaborate analyzes, it is evident that the probability of surviving a myocardial infarction is probably higher if you are younger, as can be seen Figure 1.11.

Figure 1.11: Mean age as function of year of onset and outcome.

1.2.4 Descriptive contingency table and mosaic plot

Finally, to end this first section of the project, a contingency table with its corresponding mosaic plot has been made and analyzed between the categorical variables outcome and diabetes.

A contingency table is used in statistics to summarize the relationship between two or more categorical variables [4]. In this way, through the example carried out, it is intended to elucidate if there is a significant relationship between diabetes and the outcome. First we are going to represent the contingency table together with the tables of proportions of each variable, as shown in the Table 1.8.

	y	n	nk
live	178	789	7
dead	70	189	62

(a) Contingency table of diabetes and outcome

	y	n	nk
live	18.28	81.01	0.72
dead	21.81	58.88	19.31

(b) Proportion table for diabetes variable

	y	n	nk
live	71.77	80.67	10.14
dead	28.23	19.33	89.86

(c) Proportion table for outcome variable

Table 1.8: Contingency table and proportion table for diabetes and the outcome.

Once we have the contingency table and the table of proportions of each variable, we can represent the mosaic plot, which is nothing more than the summary of the three tables in an image, in this case the Figure 1.12.

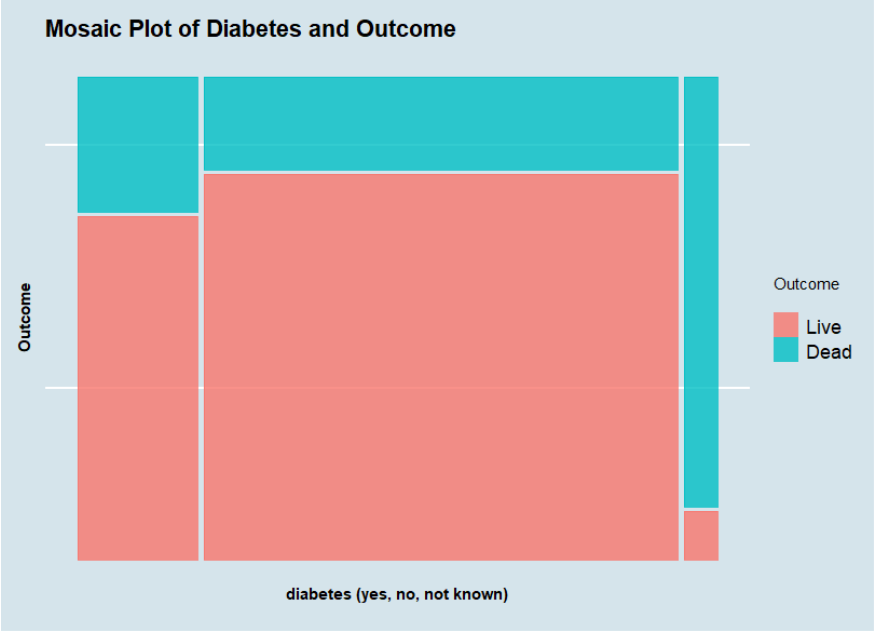


Figure 1.12: Mosaic plot of outcome and diabetes.

A mosaic plot gives us a lot of information if we know how to interpret this type of diagram.

In this way, in Figure 1.12 we can see how on the X axis we have the variable diabetes. The width of each bar is determined by the number of observations in each level (in this case three: yes, no, unknown) of this variable. Therefore, the width of the bars indicates the proportion of each level of the diabetes variable with respect to the total sample.

On the other hand, on the Y axis we have the outcome variable. In this way, the existing horizontal separation in each of the three vertical bars indicates the proportion of women alive and dead in each of the levels of the variable diabetes. Moreover, the levels of the living and dead outcome variables have been indicated with colors to distinguish them and make the plot easier to interpret.

To finalize the related analysis between the relationship between the diabetes and outcome variables, a χ^2 test was performed, in which a practically null p-value (> 0.01) was obtained (2.2×10^{-16}). Therefore, this means that we cannot accept the null hypothesis that there is a significant relationship between the categorical variables outcome and diabetes.

Chapter 2

Machine Learning analysis

In this section, it will be applied on the treated database in section 1 machine learning (ML) techniques in order to analyze the results obtained and the relevant variants for each of these in each model. Likewise, a model ensemble will be used that encompasses all classifiers and we will analyze the results for the various metrics evaluated.

In order to be able to analyze data and be able to extract information, classifications and predictions from them, it is necessary to follow a series of steps in order that allow us to obtain feasible results that can be interpreted. To do this, in this chapter the following process has been followed:

- Obtain descriptive statistics
- Clean the database
- Perform database partition in training (and validation, explained later) and test
- Data preprocessing and visualization
- Train algorithms
- Analyze the metrics of the trained models used for the test set

Note: the code created with which all the results present in the following work have been obtained can be found in the following github repository with the name of "part_2_code.R"

`https://github.com/ilancarretero/MSDS/tree/main/programming_in_R_project`

2.1 Descriptive statistics

Although it is true that this section of the analysis using machine learning has been carried out in the previous chapter, in this type of procedure it is ideal to obtain few tables that include as much information as possible. In this way, tables such as Table 2.1 are really nice when applying this step.

	type	variable	n_missing	complete_rate	f.ord	f.uniq	f.c
1	factor	outcome	0	1.00	FALSE	2	liv: 974, dea: 321
2	factor	premi	0	1.00	FALSE	3	n: 928, y: 311, nk: 56
3	factor	smstat	0	1.00	FALSE	4	n: 522, c: 390, x: 280, nk: 103
4	factor	diabetes	0	1.00	FALSE	3	n: 978, y: 248, nk: 69
5	factor	highbp	0	1.00	FALSE	3	y: 813, n: 406, nk: 76
6	factor	hichol	0	1.00	FALSE	3	n: 655, y: 452, nk: 188
7	factor	angina	0	1.00	FALSE	3	n: 724, y: 472, nk: 99
8	factor	stroke	0	1.00	FALSE	3	n: 1063, y: 153, nk: 79

(a) Descriptive information on factor type variables

	9 age	10 yronset
type	numeric	numeric
n_missing	0	0
complete_rate	1	1
n.mean	60.92	88.79
n.sd	7.04	2.55
n.p0	35	85
n.p25	57	87
n.p75	66	91
n.p100	69	93

(b) Descriptive information on numeric type variables

Table 2.1: Descriptive information on mifem variables.

2.2 Clean database

In this section, it is intended to remove or replace all values from the database that are not available in it. Although it is true that there are several procedures to substitute these values, in this case the k-nearest neighbors (kNN) algorithm has been used. By means of this algorithm, the unknown values are basically replaced by those values of other observations that are the most similar to those with incomplete data. In our case, five neighbors have been selected to determine the missing values.

On the other hand, in order to be able to apply all machine learning models correctly, one-hot encoding has been applied to categorical variables with more than two levels, while those categorical variables with two levels have been binarized (0 one level, 1 the other).

In this way, in Table 2.2 (a) we can see those positions where there were unavailable values (in our case nk, not known), while in Table 2.2 (b) we can see which values have been substituted (note that we no longer have values with value `nk`).

	premi_imp	smstat_imp	diabetes_imp	highbp_imp	hichol_imp	angina_imp	stroke_imp
1	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
2	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
3	FALSE	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE
4	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
5	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
6	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE

(a) Position of not known values (marked as TRUE)

	outcome	age	yronset	premi	smstat	diabetes	highbp	hichol	angina	stroke
1	live	63.00	85.00	n	x	n	y	y	n	n
2	live	55.00	85.00	n	c	n	y	y	n	n
3	live	68.00	85.00	y	x	n	y	n	y	n
4	live	64.00	85.00	n	x	n	y	n	y	n
5	dead	67.00	85.00	n	n	n	y	y	y	y
6	live	66.00	85.00	n	x	n	y	n	y	n

(b) Clean database of unknown values

Table 2.2: Mifem database without now known values.

2.3 Database partition

In this section, a part of the initial observations is reserved, while the rest of the samples are trained in machine learning (ML) models. The reason for this partition is due to the fact that it is necessary to have a minimum amount of data with which the model has never worked, so that, based on its training and modifying hyperparameters, it makes predictions of data that it does not know. Usually the partition that is made is:

- 80% training
- 20% test

Consequently, the final metrics will be those obtained from applying the test set to the model. It should be noted that within the training set there are various validation procedures with which to train the models. In other words, a part of the data with which the models are trained is used as validation so that the model can learn. In our case, a repeated cross-validation has been used as a validation procedure.

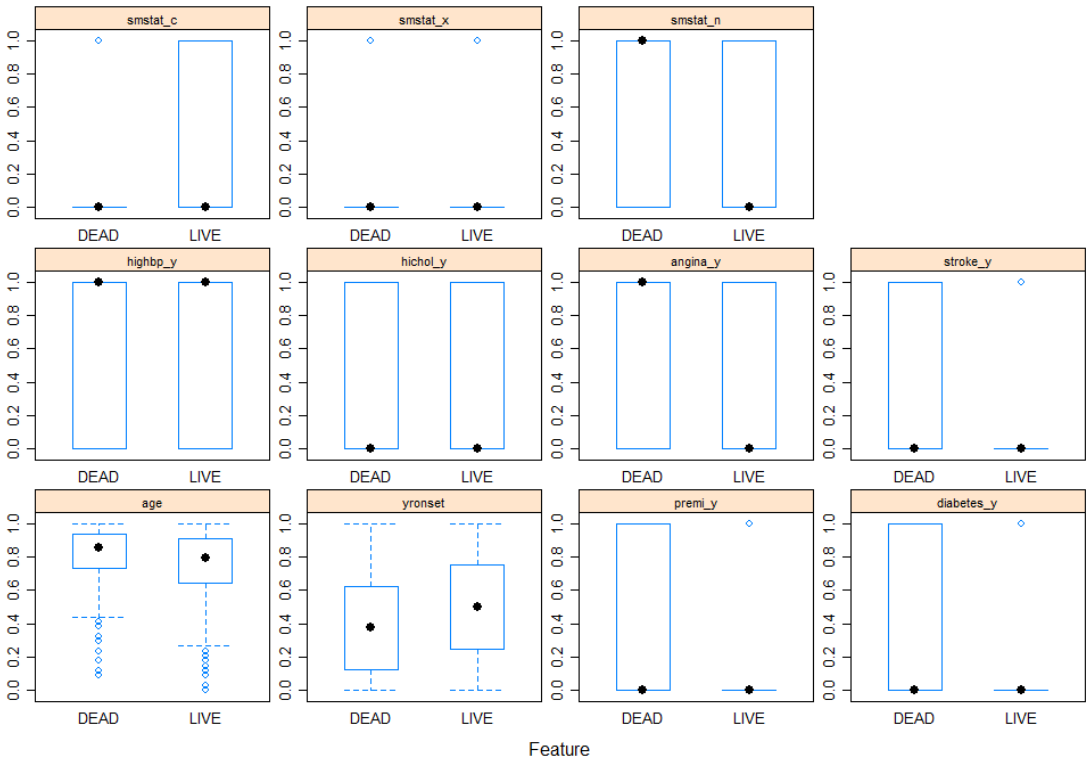
Cross validation, also known as k-fold validation, consists of a procedure in which the training data set is divided into i equal parts. All but one of these parts are taken to train the model, using the sample that has been left untrained as validation. This process is repeated i times, each time with a part and as a final result of that process all the results are averaged. Likewise, the concept of cross-validation is identical, but is added the condition that the cross-validation process is repeated n times, obtaining as a final result, the average of the final results of all cross-validations.

2.4 Data preprocessing and visualization

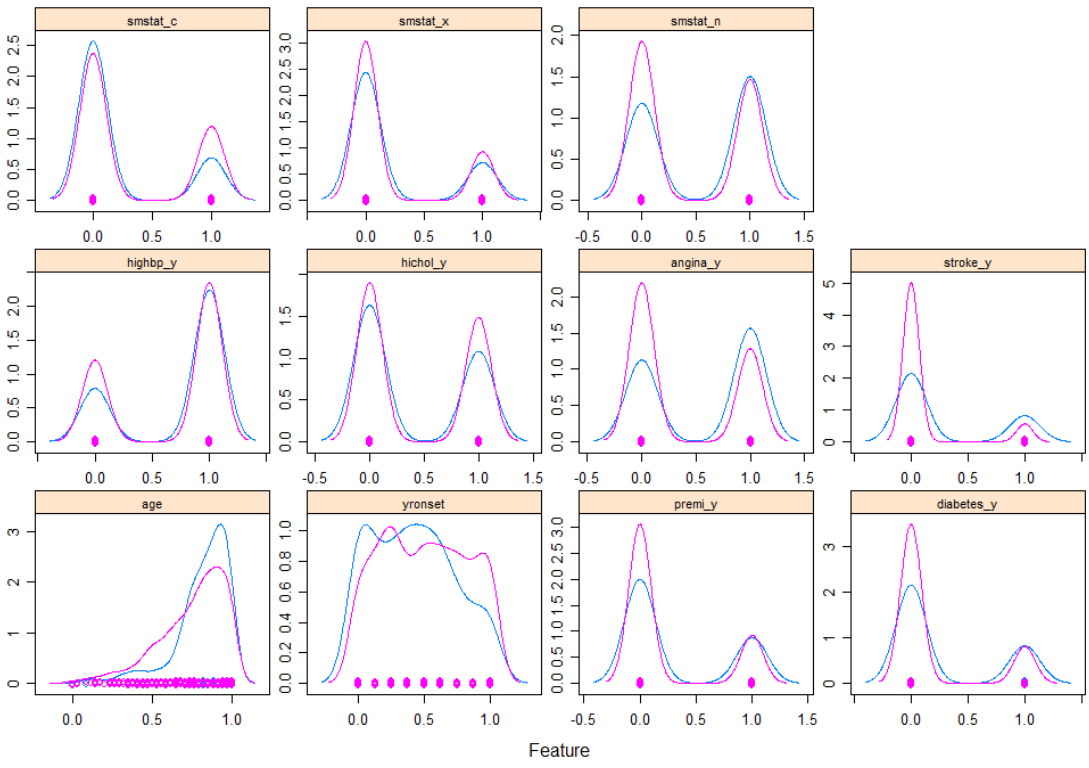
This section is one of the most important, since the future results of our classifiers depend on it.Currently, it is practically impossible to find data analysts who do not first preprocess the data to be treated, since thanks to these procedures the dimensionality of the databases can be reduced and at the same time we can obtain better results using those variables that are really significant.

Among the most used procedures we can ranged our variables between zero and one and then using features plot we can analyze them. Through the features plots, the values of one of the predictor variables (for example, by box plots or density plots) are represented as a function of the variable to be predicted (in this case $1 = live$, $0 = dead$). Thus, if the data distribution for that variable is very different for the values of the variable to be predicted, that variable is probably important.

In our case, a feature box plot and a feature density plot have been made, as shown in Figure 2.1.



(a) Features box plot



(b) Features density plot

Figure 2.1: Features plots

In the plots represented, it can be seen that in general lines there is no distribution within each plot that is very different according to the value of the outcome variable (variable to predict).

If we look at Figure 2.1 (a), we observe that, since the majority of predictor variables are binary, the box of the box plot occupies the entire interval between 0 and 1. Also, in relation to continuous variables, the differences between the box plot according to the values of the variable to predict are not significant. It is worth mentioning that, in relation to categorical variables, the only ones that have a different median according to the value of the outcome variable are `smstat_n` and `angina_y`, and therefore we will use the feature density plot below to see if these differences are significant.

If we analyze the feature density plot, the results are similar to those of the feature box plot. That is, there are no visually significant differences between the distributions of each variable according to the value of the variable to be predicted. Likewise, if we analyze the variables `smstat_n` and `angina_y`, which had different median values according to the value of the outcome variable, we observe that the differences in the distributions are minimal, so it is by no means conclusive.

Likewise, it should be mentioned that although it is true that ranged variables have been used as preprocessing, one of the most common preprocesses with which it is possible, according to the variables in the database, to greatly reduce the dimensionality without losing a large amount of information is principal component analysis (PCA).

2.5 Machine Learning algorithms and training

The models used to predict and classify our outcome variable have been the following:

- Random Forest (RF): It is a set of decision trees each trained with different training subsets. The result of this method is the combination of the different results of the decision trees.
- Multivariate Adaptative Regression Splines (MARS): It is a non-parametric regression technique in which the interactions and non-linearities of the different variables are modeled.
- k-Nearest Neighbors (kNN): It is an algorithm that predicts or classifies data based on "similarity" (by proximity) with other data learned during the training phase.
- Adaboost: Adaptive boosting in which weak classifiers are iteratively trained, with each new classifier focusing on data that was misclassified by its predecessor.
- XGBoost: Extra Gradient Boosting is based on weak classifiers from which its results are enhanced by a loss or cost function that minimizes the error in each iteration.

- Generalized Linear Model (GLM): it is a flexible generalization of linear regression in which the response variables are allowed to have error distribution models different from a normal distribution.

- Naive Bayes: It is a probabilistic classifier based on Bayes' theorem widely used for its simplicity and speed.

- Boosted Logistic Regression: Combination of regression trees (models that relate the response and the predictors by mean of recursive divisions) and boosting in order to obtain better performance.

- Neural Networks with feature extraction: Type of neural network where the relevant variables are extracted from the data sample and then passed to the layers that form the neural network to perform the classification.

- Ensemble model: In this case, it is an assembler of all the previous models in order to predict or classify the samples of the test set as well as possible.

It is worth mentioning that the repeated cross-validation method has been used for all the models in order to obtain the most robust possible results. Likewise, the models have been optimized based on three metrics: ROC, sensitivity and specificity, showing in the next section only the best results obtained based on one of the three metrics.

2.5.1 Recursive Feature Elimination

Recursive feature elimination is a feature selection algorithm in which the different predictor variables are combined in different numbers to obtain results of a classification or prediction from a specific machine learning method. In this way, as output from the application of this algorithm, the different accuracies are obtained according to the number of variables used and the variables of greatest relevance according to the algorithm.

In our case, the RFE has been carried out with random forest and naive bayes obtaining the following results:

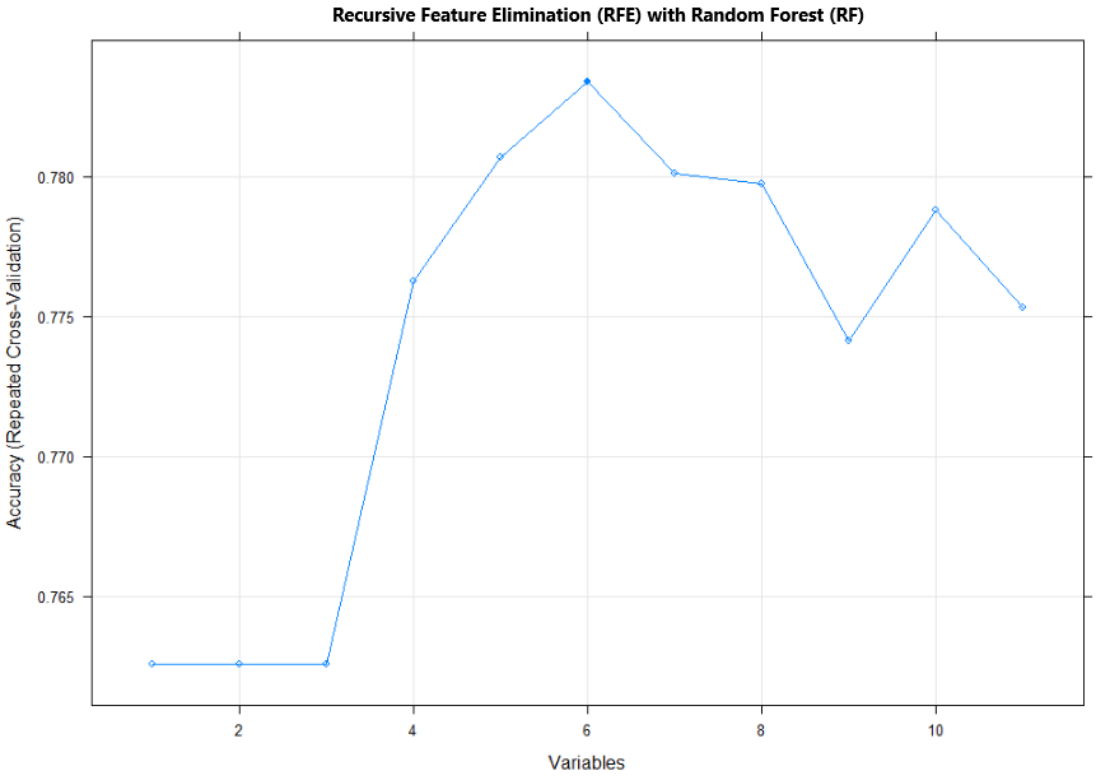


Figure 2.2: Recursive Feature Elimination with Random Forest.

	Overall
stroke_y	21.11
angina_y	16.62
hichol_y	9.57
premi_y	9.50
age	8.14
yronset	8.04
diabetes_y	7.90
smstat_n	7.60
smstat_c	6.97
highbp_y	5.58

Table 2.3: Importance of RFE variables with random forest

As we can see in Figure 2.2, the best accuracy using RFE with random forest is achieved using the six variables with the highest score in Table 2.3. In this way, from the application of this algorithm it is observed how this model with these six variables would achieve an optimal result. Now, if we change the model, as shown in Figure 2.3 and Table 2.4, the results are as follows:

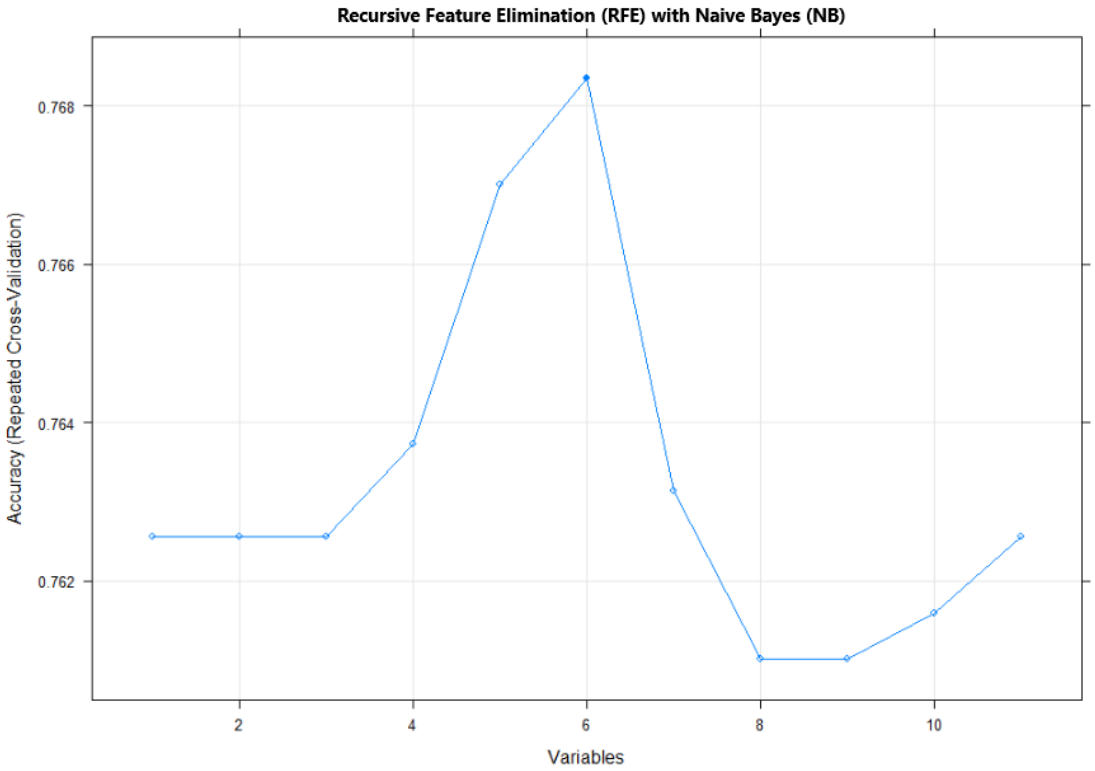


Figure 2.3: Recursive Feature Elimination with Naive Bayes.

	Overall
angina_y	0.61
age	0.59
stroke_y	0.59
yronset	0.57
smstat_n	0.56
smstat_c	0.56

Table 2.4: Importance of RFE variables with naive bayes.

In this way, we observe how the accuracy changes and the importance of the variables as well. Likewise, when comparing both methods, it should be noted that better accuracy is obtained by applying random forest. Also, the value of the importance of the variables in rl rfe with naive bayes (very small) is striking. However, despite not considering the variables with relevance, the accuracy oscillates around 0.76, which implicitly informs us of a possible imbalance of the database because it considers that there are no relevant variables but the accuracy results are not bad.

2.6 Analysis of model metrics

2.6.1 Analysis of training results

In the first place, boxplot of the results of each trained model have been obtained from the sensitivity, specificity and ROC metrics (a balance between both). Figure 2.4 shows the box plots of the models according to sensitivity.

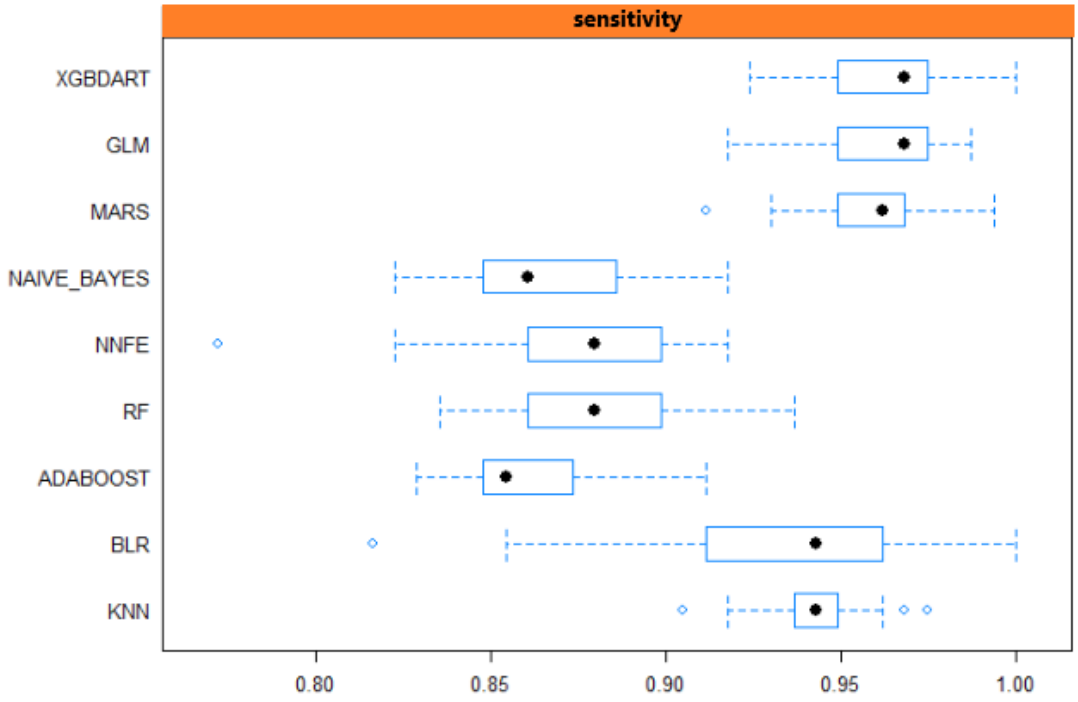


Figure 2.4: Box plots of the models according to the sensitivity.

As we can see, the sensitivity of all the models is very good, which tells us that the models detect the true positives of the sample very well, in this case, living women.

The box plots of the results of the models according to specificity are shown in the Figure 2.5.

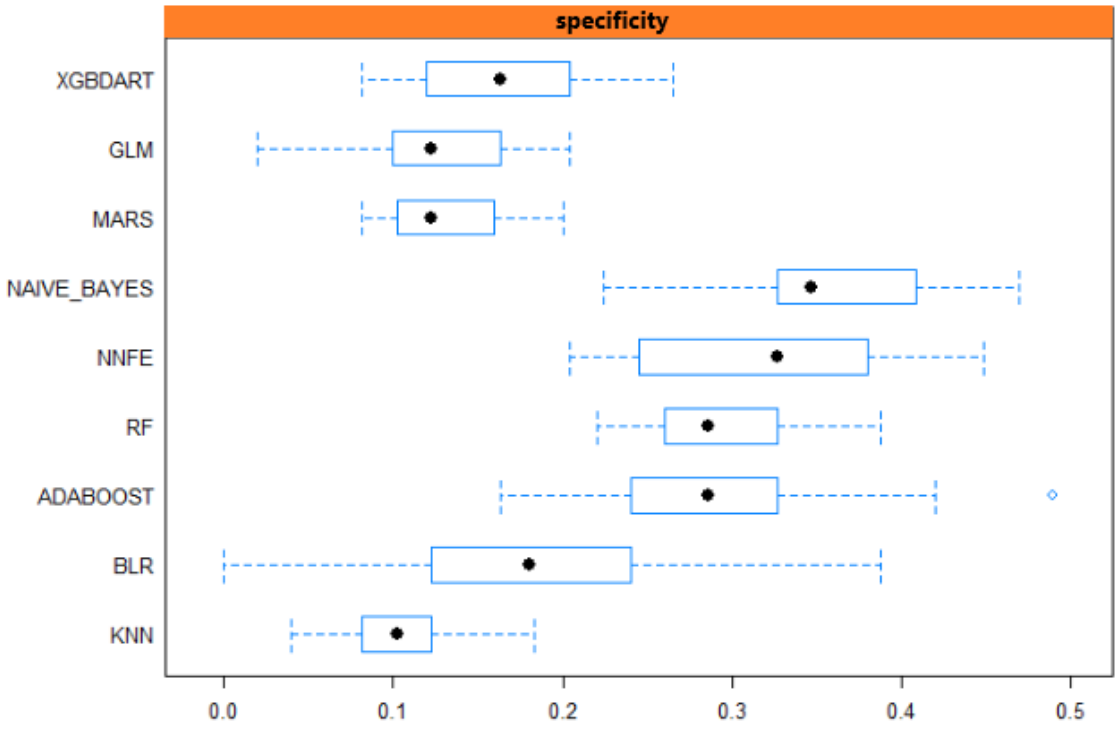


Figure 2.5: Box plots of the models according to the specificity.

As we can see, the results are really bad. Thus, the reason why all the models have a specificity lower than 0.5 is because the database we are dealing with is unbalanced. That is, there is a much higher number of living women than dead women and, consequently, the models classify most of the women as alive, obtaining a great error when it comes to finding true negatives (dead women), as demonstrated by the low specificity.

Finally, Figure 2.6 shows us the box plots of the models based on the ROC. metric. If we analyze this figure, we can see the results of the ROC metric in the models, being an average of the previous metrics, not being excessively good. However, these results camouflage the difference between sensitivity and specificity and that is why it is always convenient to analyze more than one metric in order to detect if there are differences that may be relevant to the study in question.

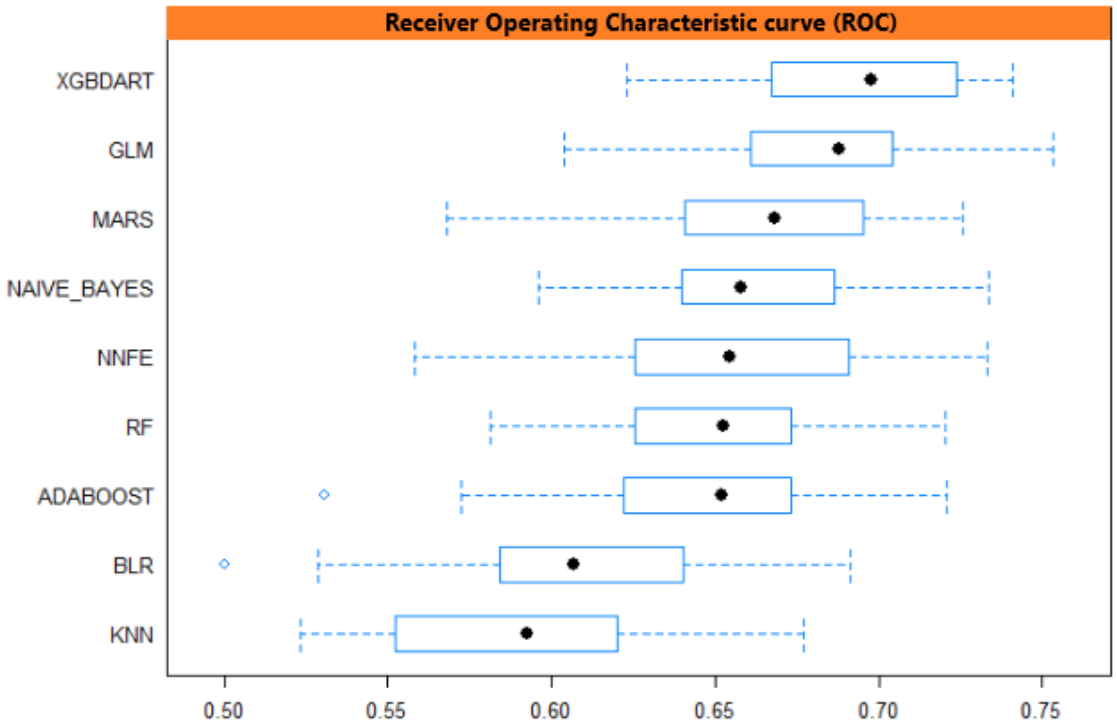


Figure 2.6: Box plots of the models according to the ROC.

2.6.2 Analysis of the confusion matrices with the test set

In this section we are going to analyze the results obtained for each model using various metrics and the confusion matrix of each of these obtained for the test set.

- Random Forest (RF): As we can see, the results obtained by this algorithm are really good. It is noteworthy that specificity has greatly improved (in training it was much lower as we can see in Figure 2.5) and sensitivity to a lesser extent. In this way, the classification capacity of the test set of this model is excellent.

  Likewise, it should be mentioned that the variables that have been most important to it can be obtained from all the models. In this way, Figure 2.8 is shown. Likewise, in this figure it is observed that the continuous variables `year` and `age` are of special importance for random forest.

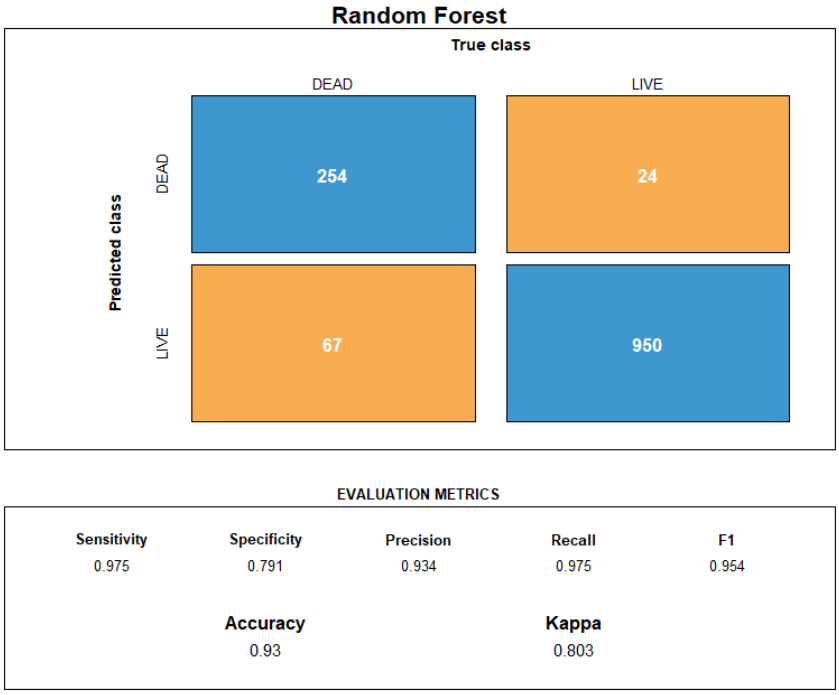


Figure 2.7: Random forest confusion matrix.

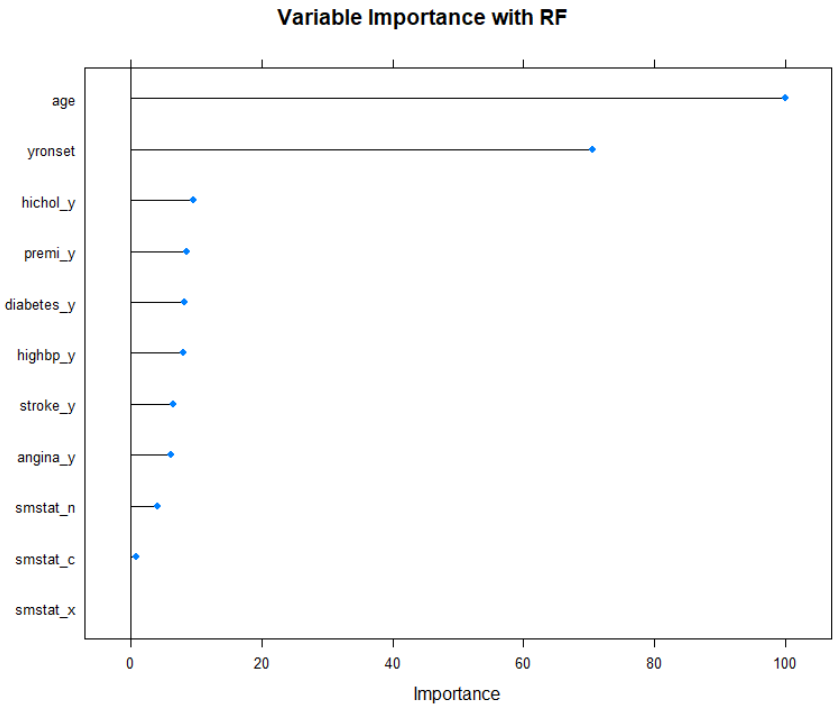


Figure 2.8: Random forest important variables.

- Multivariate Adaptative Regression Splines (MARS):

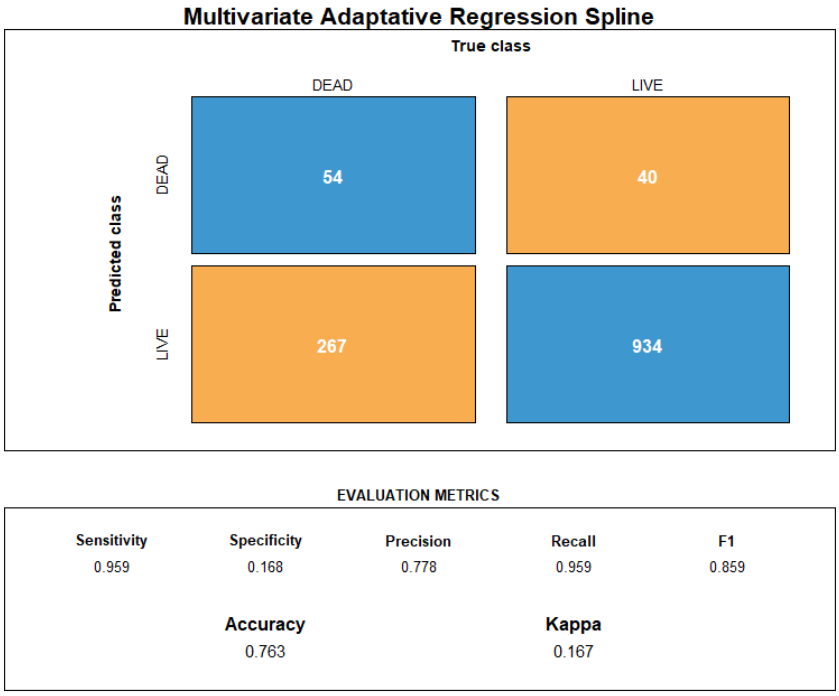


Figure 2.9: Multivariate Adaptative Regression Splines confusion matrix.

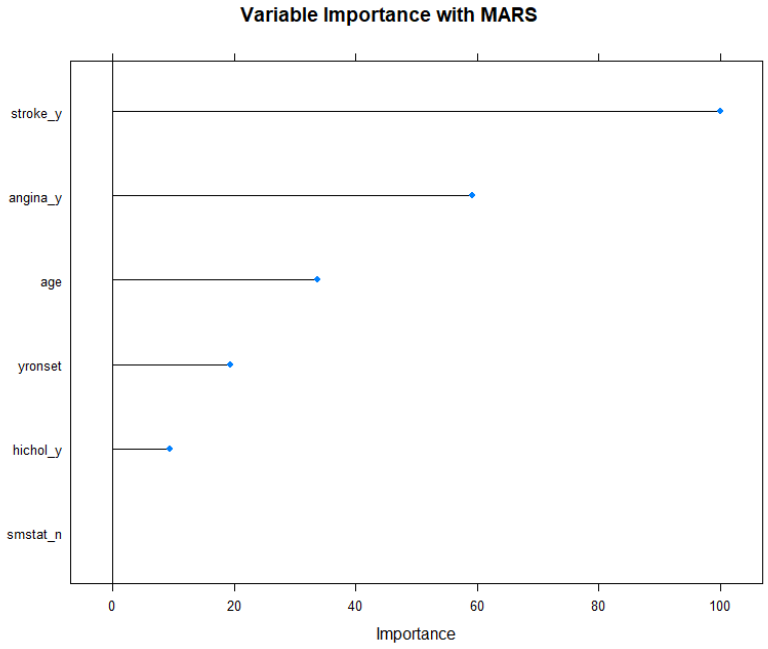


Figure 2.10: Multivariate Adaptative Regression Splines important variables.

As we can see in the figure 2.9 this model, although it has an accuracy that is not bad, is not good. This is because it has a specificity of 0.163 which means that it practically does not distinguish true negatives. Consequently, this model is practically classifying all women as alive. Likewise, if sensitivity is being taken as positive and specificity as negative, the F1-score, widely used as an evaluation metric in unbalanced databases, could not be selected. To use this metric, which is nothing more than the harmonic mean of sensitivity and precision, it would be necessary to take it dead as positive and live as negative.

Regarding the importance of the variables, `stroke` and `angina` stand out above the rest, and there are also several variables that this model does not consider significant.

- k-Nearest Neighbors (kNN)

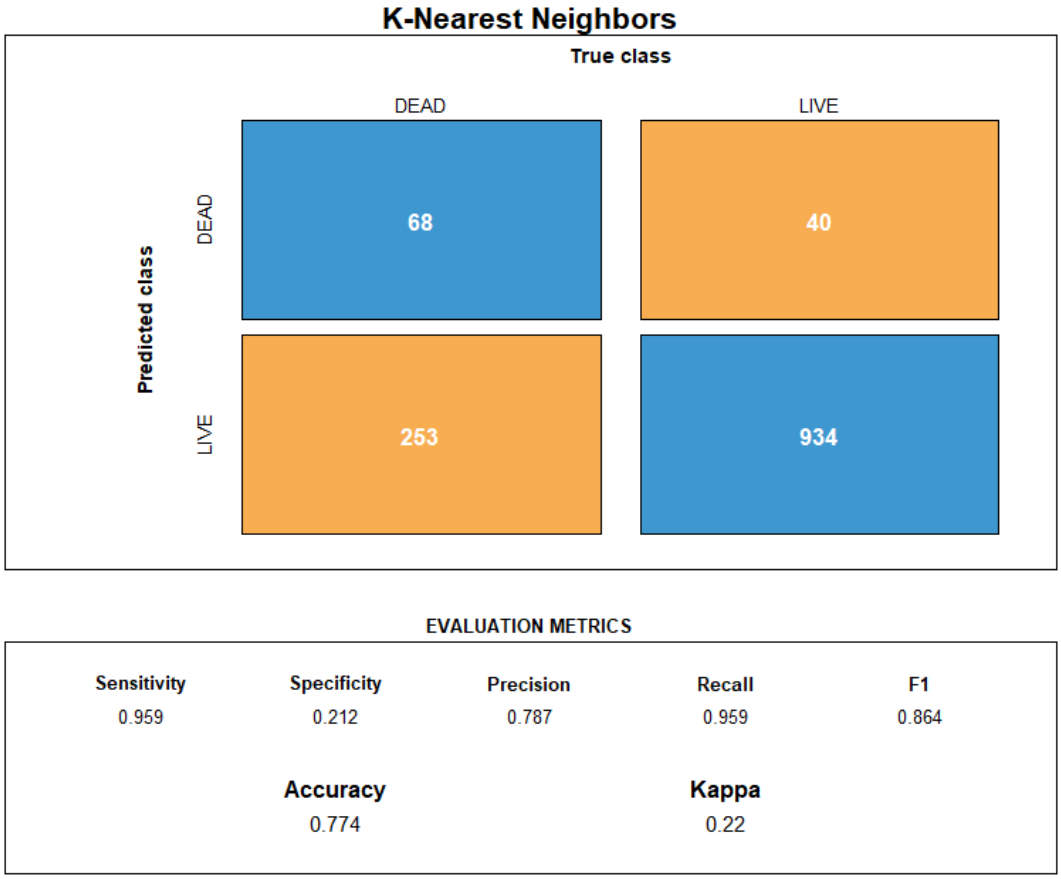


Figure 2.11: k-Nearest Neighbors confusion matrix.

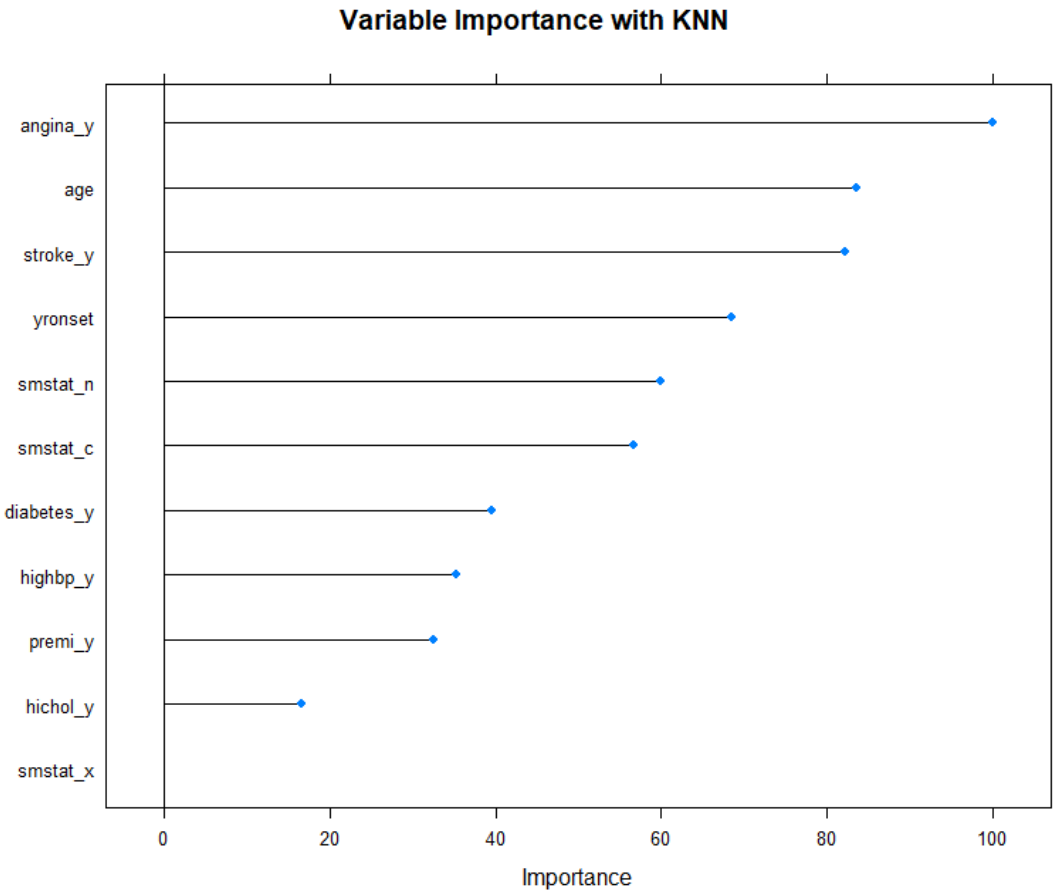


Figure 2.12: k-Nearest Neighbors important variables.

Like MARS, Figure 2.13 allows us to affirm that this model does not obtain good results either. While it is true that it improves specificity slightly, 0.212 is still an extremely poor value. Regarding the importance of the variables of this model, it should be noted that it uses all the variables except one, `smstat_x`.

- Adaboost

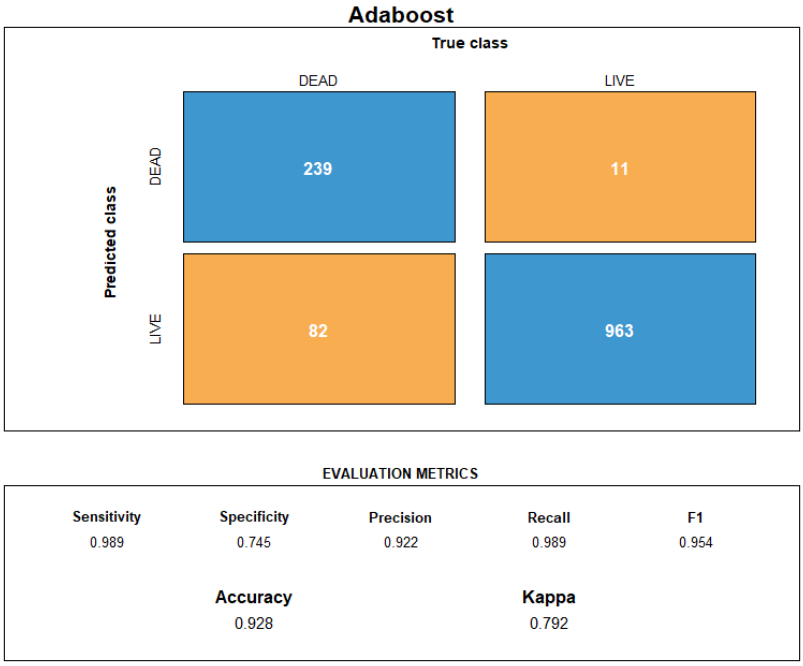


Figure 2.13: Adaboost confusion matrix.

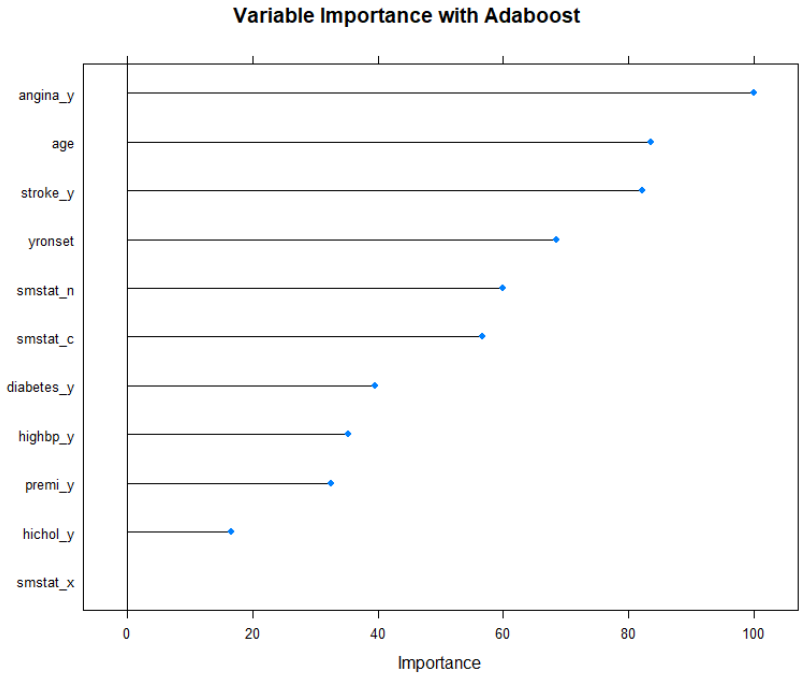


Figure 2.14: Adaboost important variables.

Adaboost presents one of the best results, similar to those of random forest, obtaining a very high sensitivity value (0.989) and increasing the specificity to 0.745, a considerable increase compared to its results in training. Also, in relation to the importance of the variables, `angina` is the main one and then `age`. In this way, it is observed how `age` is important in the models that make the best predictions at the moment.

- XGBoost: As can be seen in Figure 2.15, this model obtains very poor results in specificity. Although the sensitivity is very good, as in most models, it has little capacity to detect true negatives or observations from the test set identified as "dead".

  On the other hand, as seen in the Figure 2.16 `stroke_y` and `age` are the variables of greatest importance.

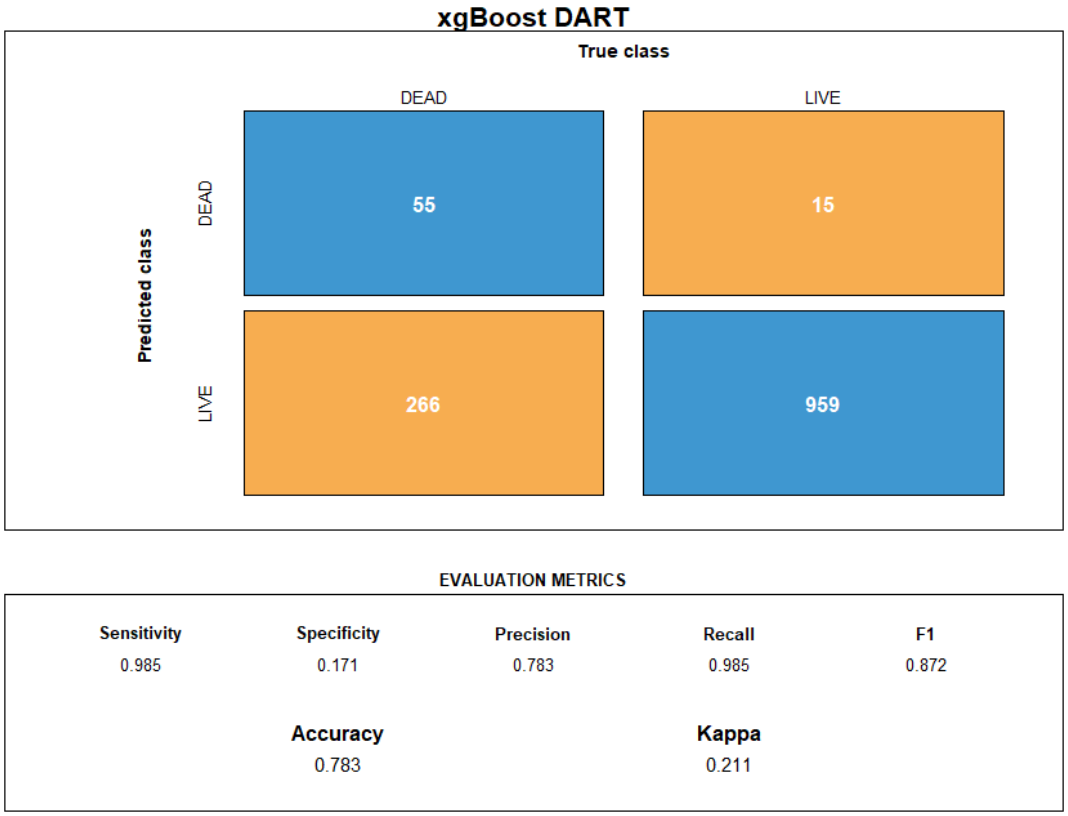


Figure 2.15: XGBoost confusion matrix.

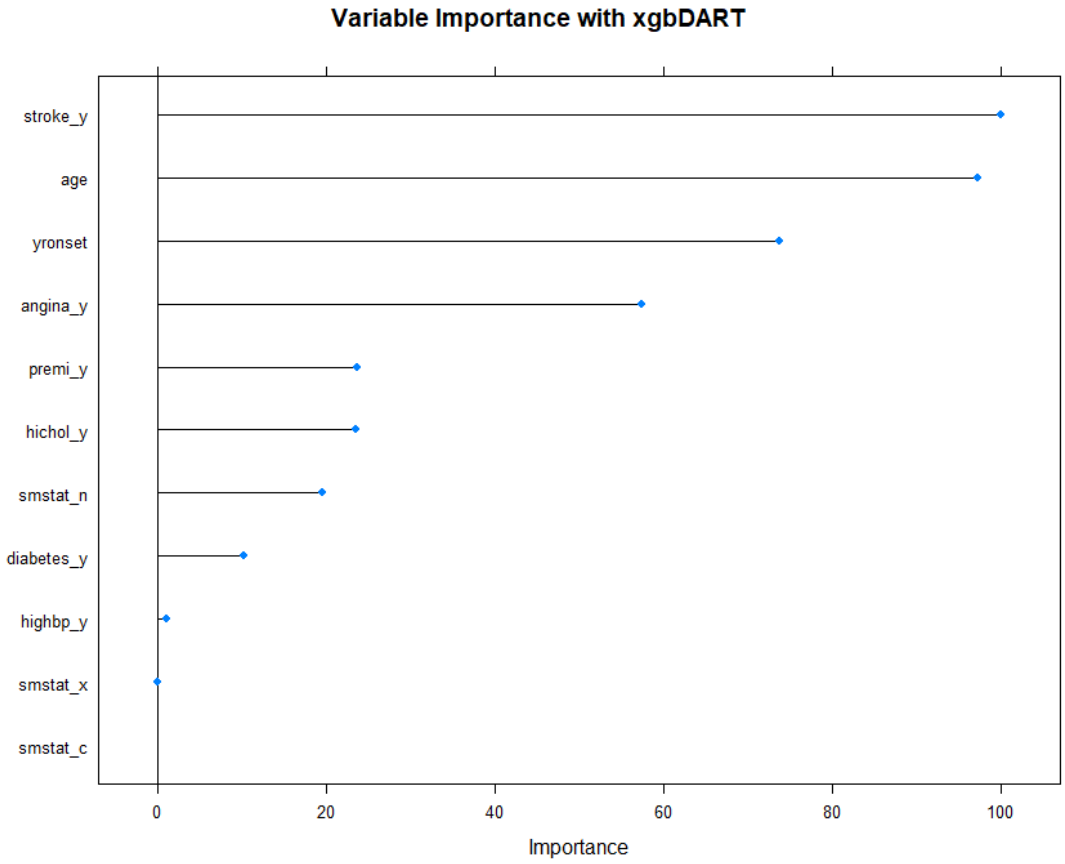


Figure 2.16: XGBoost important variables.

- Generalized Linear Model (GLM): Like the MARS model, the results of this model are extremely poor. By classifying the majority of individuals in the test set as "live" (1) and being the majority class, metrics such as accuracy or precision do not come out so bad when in reality the model is understanding that almost the entire set of tets is positive (live) without caring for the minority class.

Regarding the importance of the variables, it is also similar to the MARS model, being `stroke_y` and `angina_y` the most relevant variables.

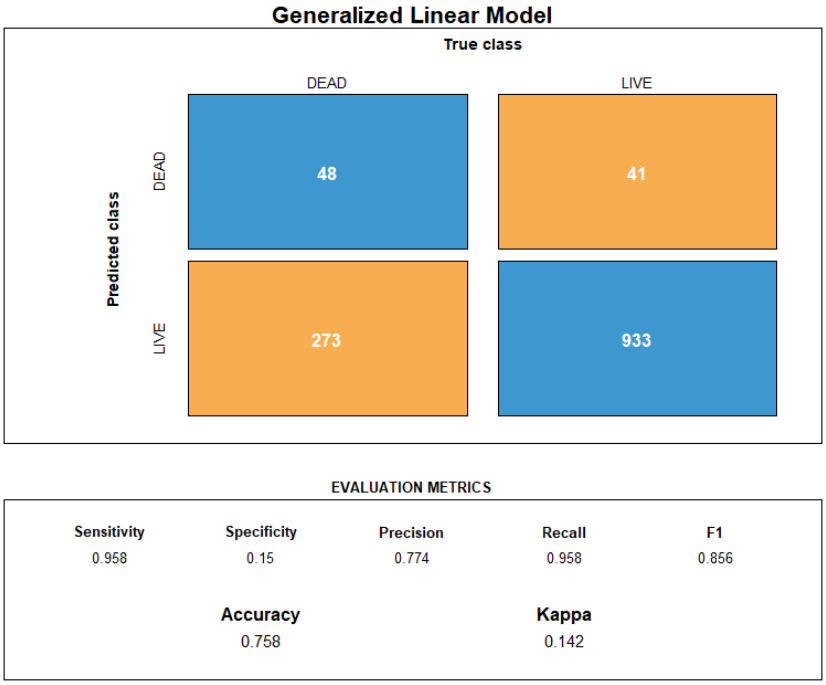


Figure 2.17: Generalized Linear Model confusion matrix.

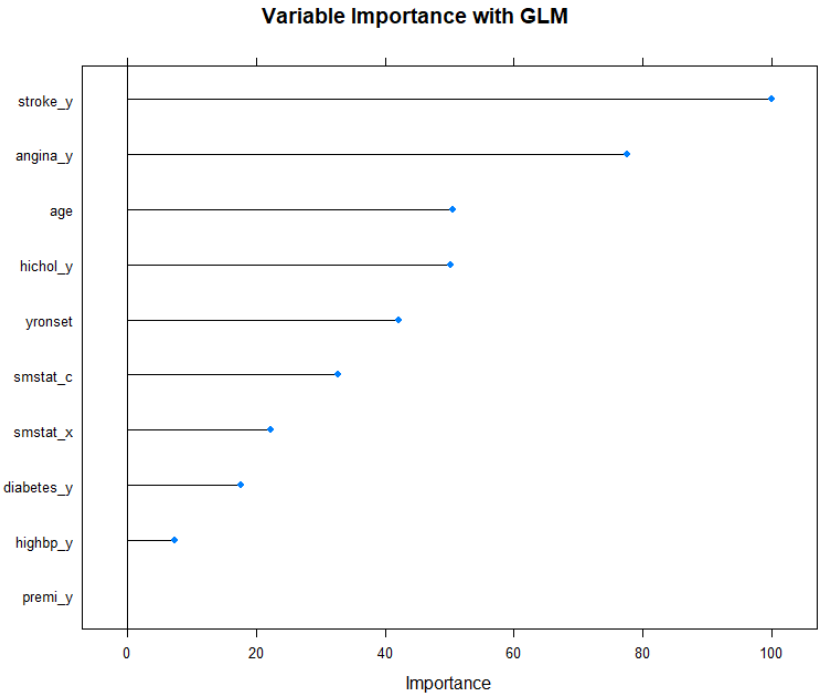


Figure 2.18: Generalized Linear Model important variables.

- Naive Bayes: As we can see in Figure 2.19, this classifier is not entirely valid for its function. Although it is true that it improves the specificity compared to other models such as MARS or GLM without excessively reducing its sensitivity, a value of 0.371 is the specificity is still too low. Regarding the importance of the variables, in Figure 2.20 we can see how `angina_y`, `age` and `stroke_y` are considered the most relevant.

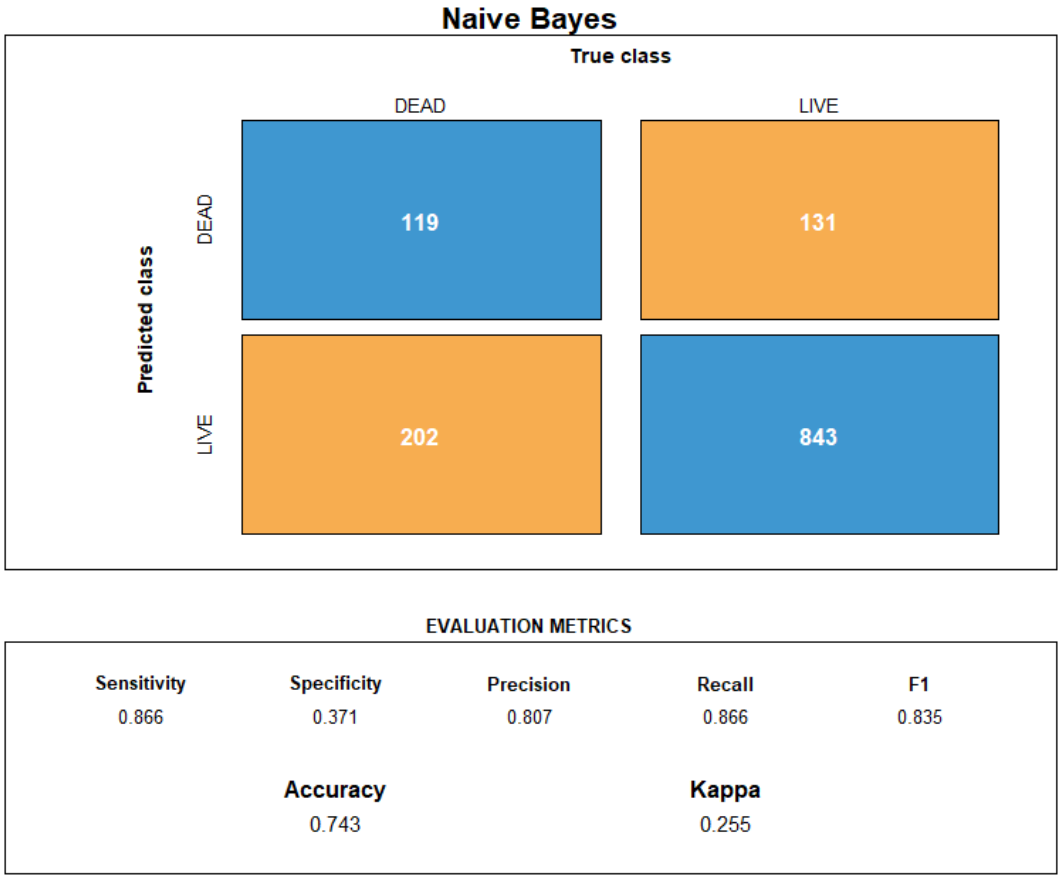


Figure 2.19: Naive Bayes confusion matrix.

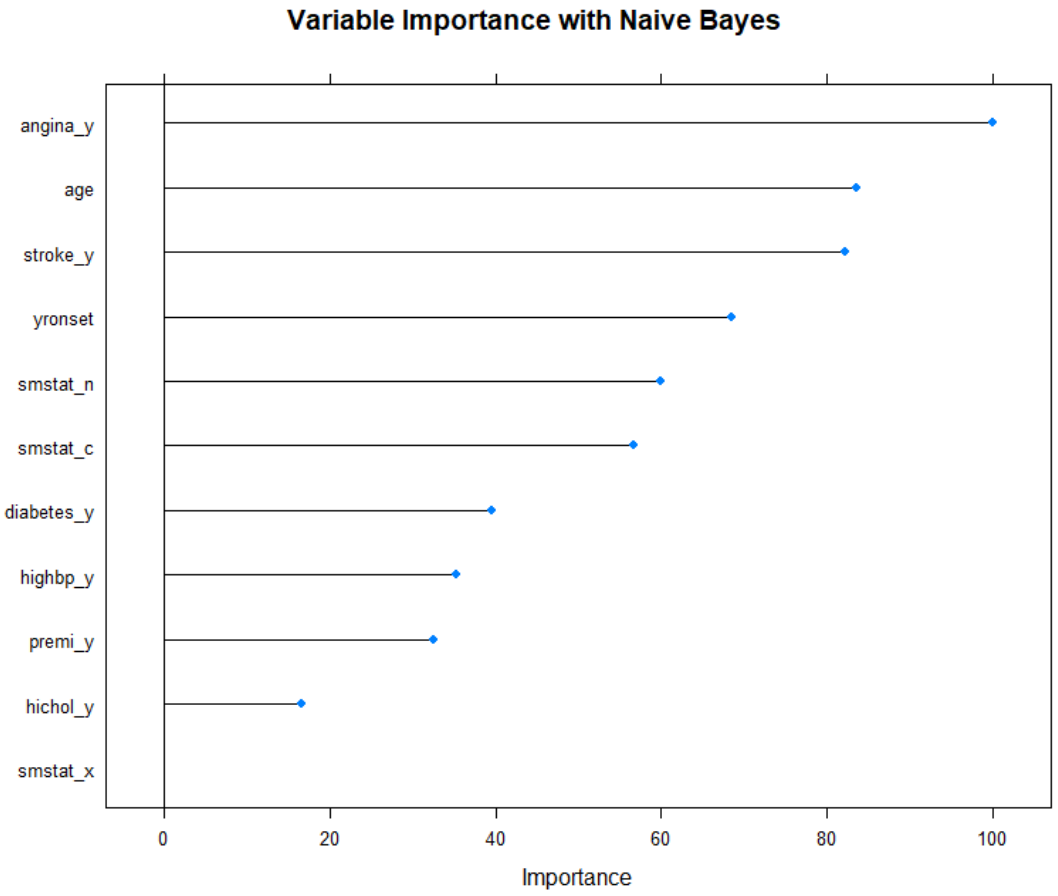


Figure 2.20: Naive Bayes important variables.

- Boosted Logistic Regression: As can be seen in Figure 2.21, this model has one of the lowest values of specificity, which can be understood as a negative aspect. 0.109 in specificity is very close to 0. A model whose specificity is 0 and sensitivity is 1 means that it is classifying the entire sample as positive regardless of the type of observation. Therefore, it would not be at all advisable to use this model for the dataset used.

  On the other hand, in relation to the important variables selected by this model, we observe in Figure 2.22 that they are the same as that of the naive bayes model.

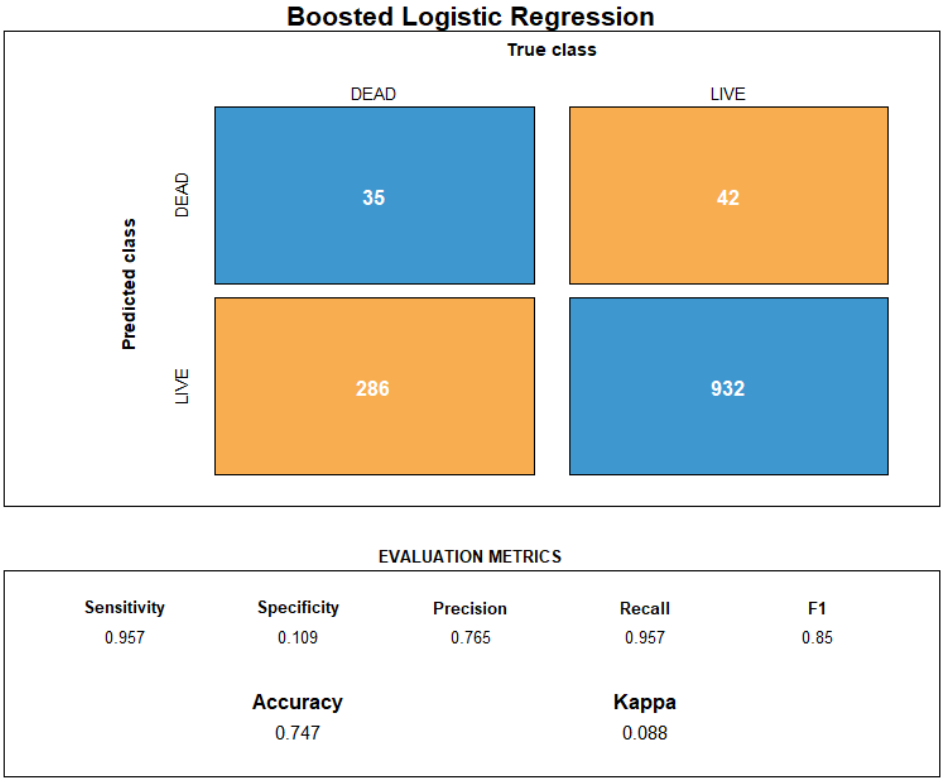


Figure 2.21: Boosted Logistic Regression confusion matrix.

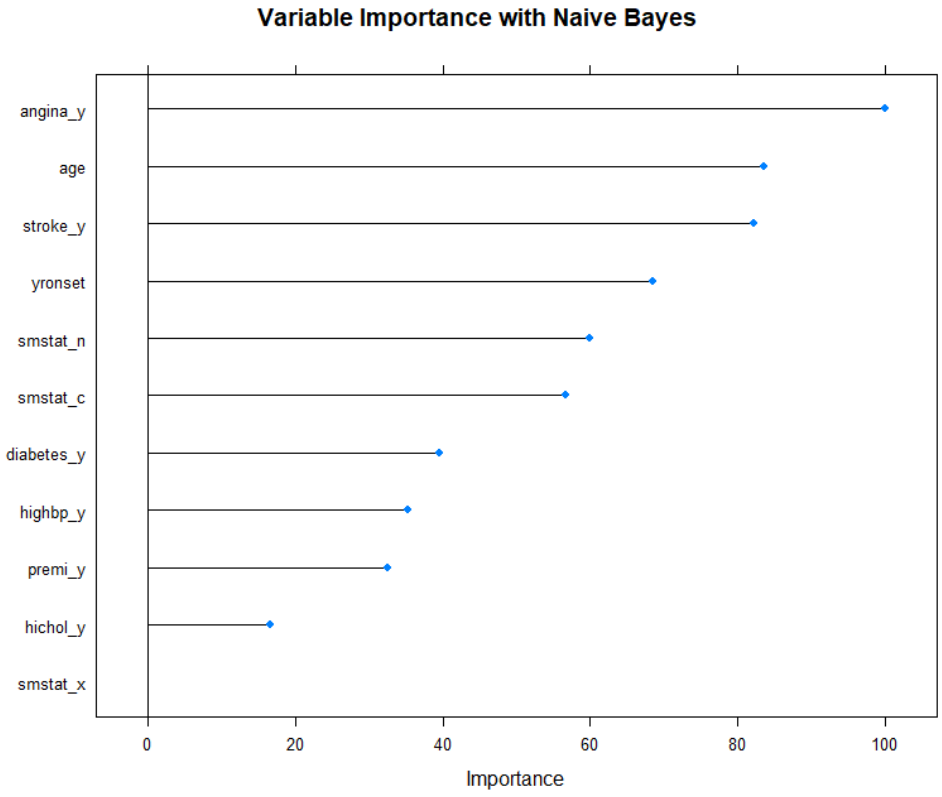


Figure 2.22: Boosted Logistic Regression important variables.

- Neural Network with feature extraction

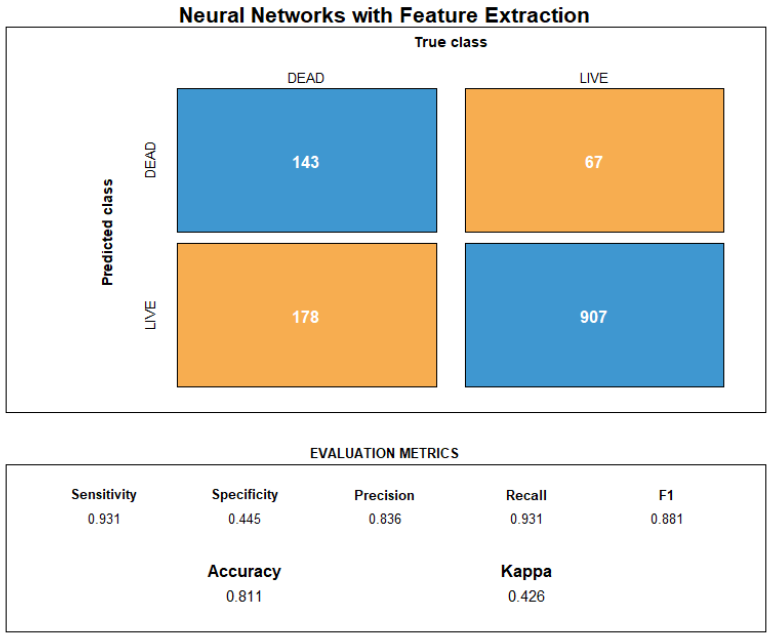


Figure 2.23: Neural Network with feature extraction confusion matrix.

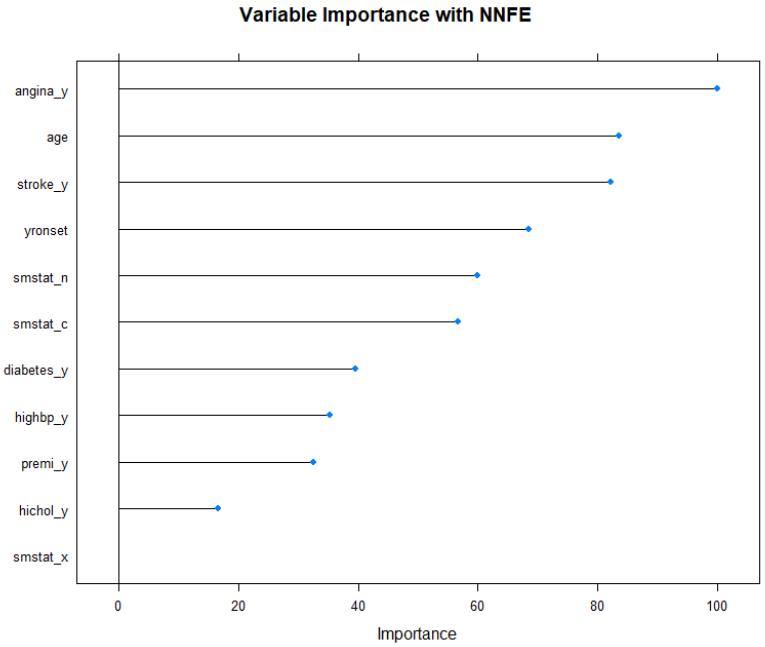


Figure 2.24: Neural Network with feature extraction important variables.

As we can see in Figure 2.23, this model classifies better than most, since it has a very good sensitivity value and improves specificity compared to most. However, the value is still less than 0.5, so it could continue to be considered an invalid model or with poor results for the test set.

On the other hand, as we can see in Figure 2.24 `angina_y`, `age` and `stroke_y` are the most relevant according to the model.

- Ensemble model: As we can see in Figure 2.25, the ensemble model, like the vast majority of models, has a low specificity value. Possibly it is due to the fact that it will be optimizing based on the accuracy and not on the specificity as for example has been indicated in random forest. Normally, these types of models usually obtain quite good results since they combine multiple models. However, they improve the evaluation metrics in exchange for losing understanding of the model.

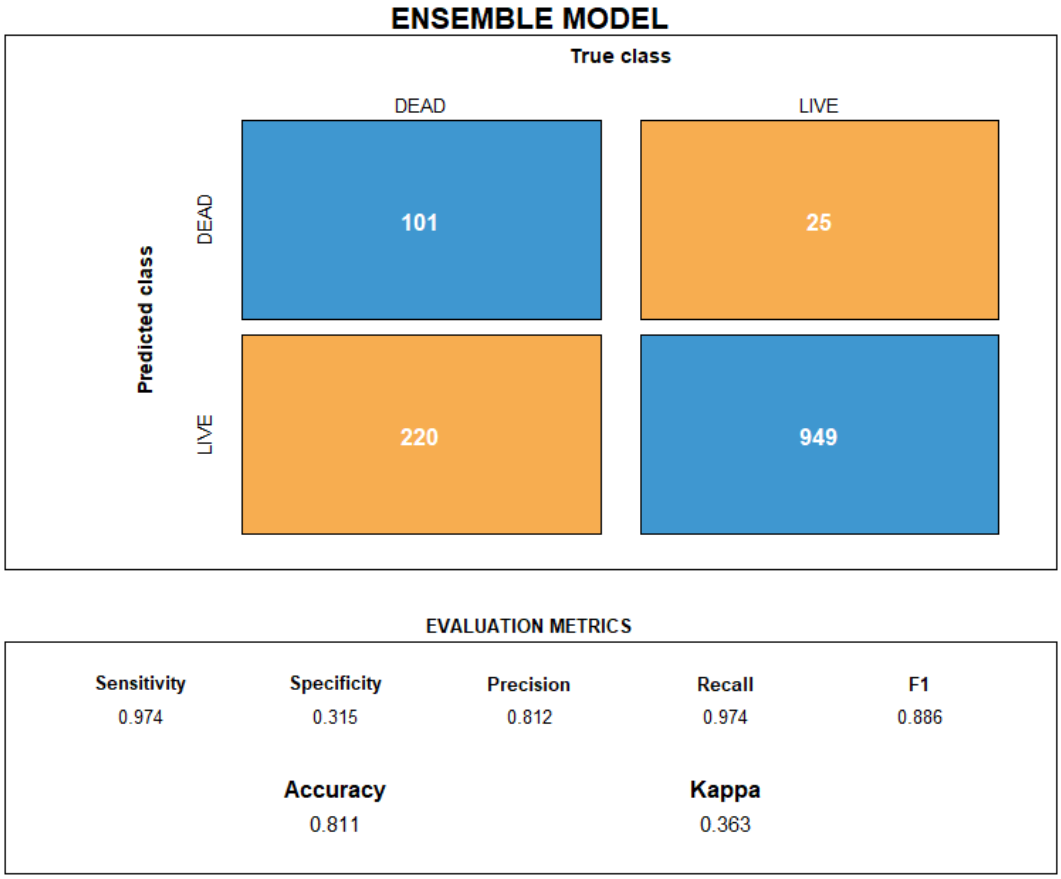


Figure 2.25: Ensemble model confusion matrix.

Finally, once we have analyzed each and every one of the confusion matrices of the and tested ML models, we can reach the conclusions set out in Table 2.5.

Good models for the classification and prediction of the outcome variable according to the evaluation metrics of the test set	**Bad models** for the classification and prediction of the outcome variable according to the evaluation metrics of the test set
Random Forest (RF)	Multivariate Adaptative Regression Splines (MARS)
Adaboost	k-Nearest Neighbors (kNN)
	XGBoost
	Generalized Linear Model (GLM)
	Naive Bayes
	Boosted Logistic Regression
	Neural Networks with feature extraction
	Ensemble

Table 2.5: Comparison of models according to evaluation metrics

Chapter 3

Conclusions

By carrying out this project, theoretical content seen during the classes has been put into practice. In the first place, at the end of this project the usefulness of R as a software for applying data analysis can be revealed. Despite not being so fast in certain sections of machine learning such as python, thanks to the multiple packages it is possible to carry out very exhaustive and relevant studies for the scientific community. On the other hand, it should be noted that R is one of the most recommended software for statistical analysis.

On the other hand, during the code carried out to obtain each and every one of the results, the good practices discussed in class have been used, such as commenting on the code, indenting, being ordered or referencing the packages from which the functions come. In this sense, applying all these good practices has two uses: for oneself and for those who read your code and have to work on it.

Finally, when carrying out this work, not only has there been a first approach to how to program in R, but also to how to perform a data analysis. In this way, a descriptive analysis has been carried out, the data and their distributions have been visualized and ML has been applied to the database to be able to use models that are capable of classifying new data similar to those of the database in question. Likewise, all the results obtained have been analyzed, and consequently conclusions can be reached on the data processed.

Bibliography

1. Maindonald, J. & Braun, J. *Data analysis and graphics using R: an example-based approach* (Cambridge University Press, 2006) (cit. on p. 9).
2. R-Documentation. *mifem: Mortality Outcomes For Females Suffering Myocardial Infarction* [Accessed Oct. 20, 2020]. 2020. `rdocumentation.org/packages/DAAG/versions/1.22/topics/mifem` (cit. on p. 9).
3. Böthig, S. WHO MONICA Project: objectives and design. *International Journal of Epidemiology* **18,** S29–37 (1989) (cit. on p. 9).
4. Agresti, A. *An introduction to categorical data analysis* (John Wiley & Sons, 2018) (cit. on p. 23).