Machine Learning Project:
Heart Failure Analysis using ML in R

Pietro Ferrazzi
Margarita Hernández Casas

INTRODUCTION

The following project is based on data coming from a medical study about hearth attacks in Pakistan. The observations were taken over 299 patients with age above 40. The data set has been downloaded at this link. It contains 13 variables. Each statistic unit is one patient. What follows is a brief description of the variables: *TIME*: time under observation for the patients, i.e., follow-up time during which the patient was constantly monitored. If the event occurred after that time, there is not track of that in the data; *Event*: it is set to 1 is the heart attack occurred, 0 otherwise; *Gender*, Smoking, Diabetes, BP (blood pressure) and Anaemia are dichotomous variables; *Age*, Ejection.Fraction (percentage of blood leaving the heart at each contraction), Sodium, Creatinine, Pletelets and CPK (level of the CPK enzyme in the blood) are continuous variables.

PREVIOUS AVAILABLE WORK

There are previous studies conducted with this dataset, two research papers with different approaches, as well as different results. Both tried to find significant predictors, one implementing Cox regression for survival analysis, the other using rather modern techniques, biostatistics and machine learning. The first study [1] determined that age, serum creatinine, blood pressure, anaemia and ejection fraction were contributors to the risk of mortality. The second study [2] concluded that only serum creatinine and ejection fraction were relevant features, and that those two alone led to more accurate predictions. Furthermore, the more recent study, showed that random forest was the top performing classifier among 10 different prediction models.

After conducting our own analysis, we would like to see which of these studies have results most similar to ours and analyse why this might be the case.

1. Chicco, D., Jurman, G. Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone. BMC Med Inform Decis Mak 20, 16 (2020). DOI

2. Ahmad T, Munir A, Bhatti SH, Aftab M, Raza MA (2017) Survival analysis of heart failure patients: A case study. PLoS ONE 12(7): e0181001. DOI

SCOPE OF THE PROJECT

We wanted to see if the combination of the two approaches -survival models and ordinary classification- could perform better than the latter one on itself. To do that, we divided the work in three main sections: Survival Analysis, Classification and Comparison. Before getting into this analysis, we performed an explorative analysis to identify important variables and correlations, in order to better understand the

data and identify which variables could be useful to provide forecasts about the variable Event treated as a class.

MOTIVATION

Integrate classical statistical analysis with ML techniques to see if they can perform well in an adverse scenario: few variables, few observations, and the fact that the data are of a survival type and then should be treated following strict procedures. In addition, we are really interested in data coming from the sanitary field.

Gender, *Smoking*, *Diabetes*, *BP* and *Anaemia* are factors that have been saved as numerical values. To start, we decided to transform them into categorical variables.

INITIAL SPLIT

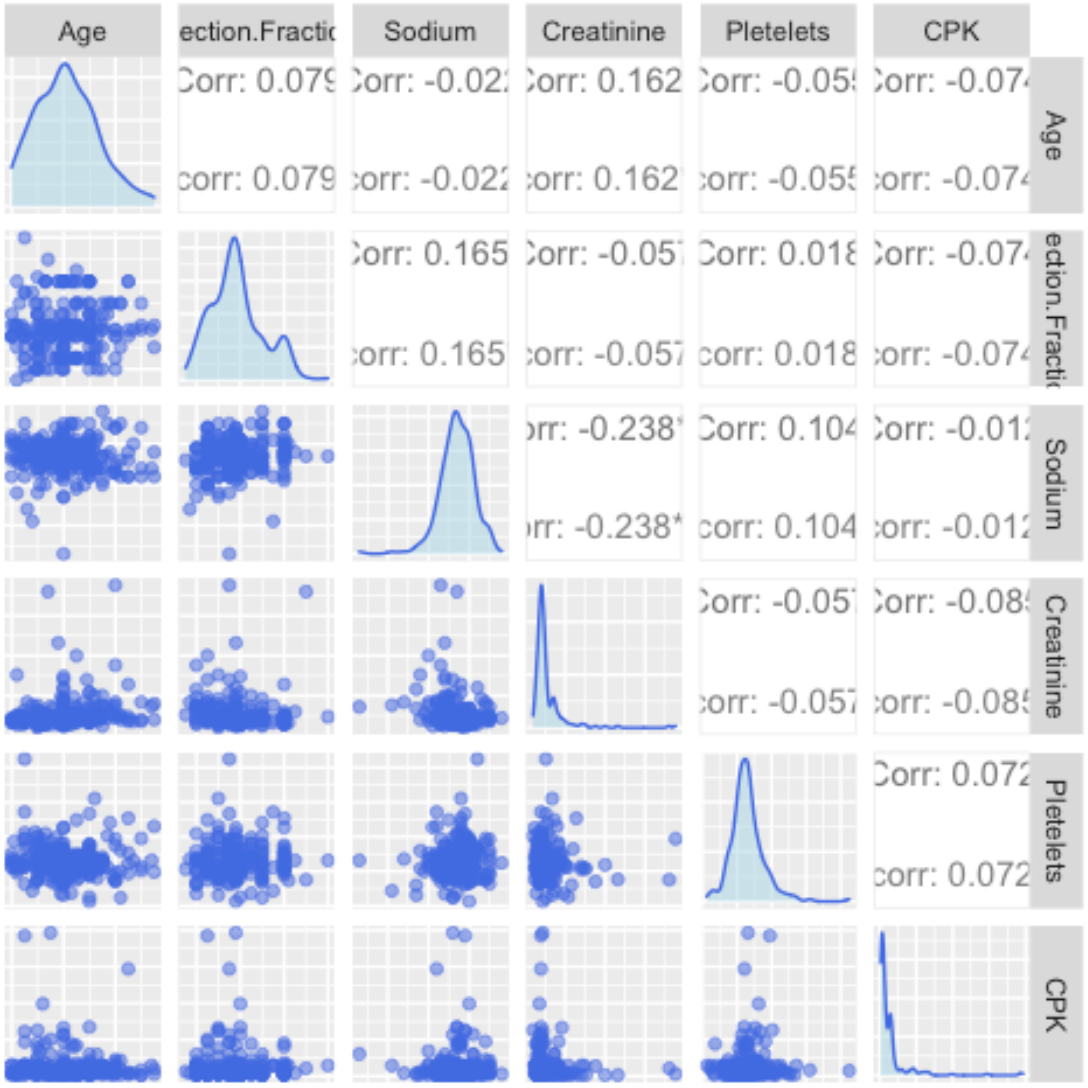
Before starting looking at the data we randomly split it in one set for training and one for testing. We are splitting this two sets of data with an 75-25 ratio. We want to use as much data as possible for training since we have few instances, while being able to get a good estimation of the best model at the end of this analysis. We also make this split stratified, which means keeping the proportion of positive cases in the subsamples as it appears in the full data set. We want our models to learn the data as it is in reality, quite unbalanced. Splitting data in train and test will assure that the observations we will do during the explorative analysis and the decisions that will be made on them will not be biased by the knowledge acquired from the test set. What follows is based on the training set.

EXPLORATIVE ANALYSIS

We started with the explorative analysis of the data. The aim of this first section is to reach a general understanding of the data we have, their distribution, the correlation between variables and in general all the aspects that concern the descriptive analysis of the observations. Only after a rigorous analysis we can start to model the data, since before that we will not have a complete comprehension of variables and their relationships.

Pairs of continuous variables

First, we are interested to see the joint and disjoint distribution of all the pairs of continuous variables.



The correlation is significantly different from zero only for the couples *Creatinine-Age*, *Creatinine-Sodium* and *Sodium-Ejection.Fraction*, and anyways very low. *Creatinine* and *CPK* looks to have long right tails.

The correlation between the event and the explicative variables is:

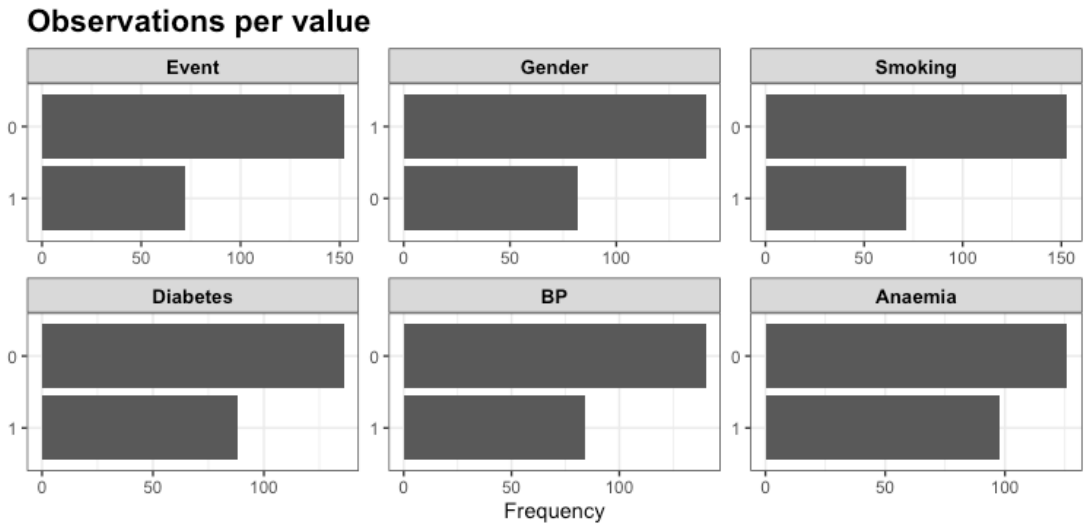
Biserial Correlation between Ejection.Fraction and Event: 0.22 Biserial Correlation between Creatinine and Event:-0.29 Biserial Correlation between Age and Event: -0.24 Biserial Correlation between Sodium and Event: 0.19 Biserial Correlation between Pletelets and Event: 0.01 Biserial Correlation between CPK and Event: -0.11 Correlation between BP and Event: -0.08 Correlation between Anaemia and Event: -0.05 Correlation between Smoking and Event: 0.02 Correlation between Diabetes and Event: -0.01

The correlation between numerical variables and the response is evaluated using the Point-Biserial Correlation.

The observed values are quite low. This might be a limitation in our project, because not only we have few instances available but also low correlations.

Factors

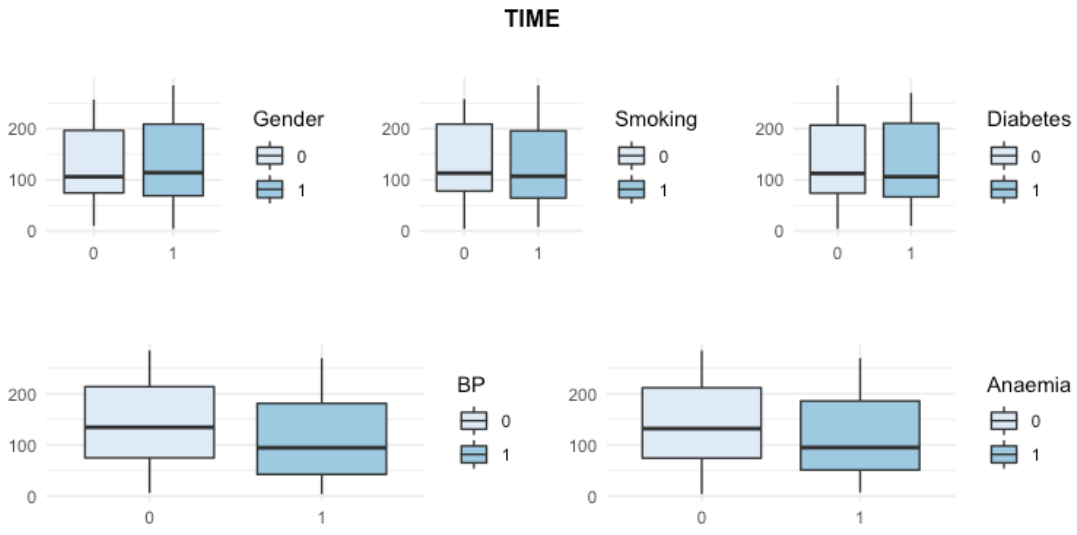
Then, we plotted the binomial variables in order to see if their distribution is balanced.



These distribution plots show that the proportion of statistical units with *Event*=1 is only 0.24. Even *Gender*, *Smoking* and *BP* have unbalanced distributions, but the number of elements in the two levels for all these variables are enough to say that we are not in presence of rare classes.

Time and factors

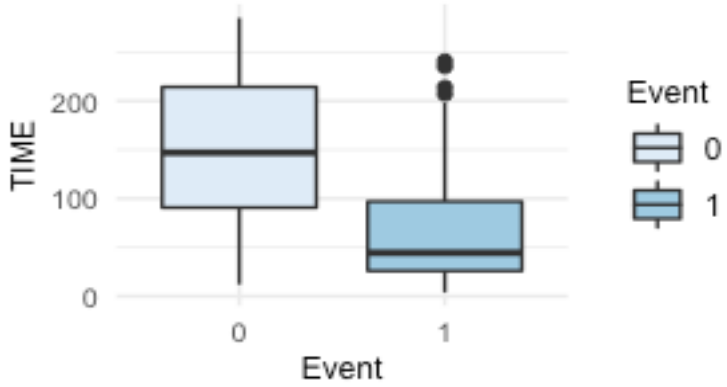
An important aspect to consider is how the time under observation is related to the other variables. The following boxplots show that there is not relation between the time and the binomial variables.



This result is interesting, since it tell us that the experiment from which the data has been collected is unbiased in respect of differences between the persons under observation. In other words, the selected

people were kept under observation independently from the collected variables that described themselves. It is important because it means that we can rely the way data has been collected and use it to provide results that will not be biased.

We have to consider aside the box plot about *TIME* in respect to the *Event*. It is clear that the time under observation for people that have presented an hart attack is lower than the the one of the ones that did not. As expected, this indicates that the heart attack looks to reduce the time under observation. The opposite would have been unexpected. It can seems obvious, but not having this result would have lead us to conclude that the data were not able to explain the relation between the explicative variables and the answer variable.



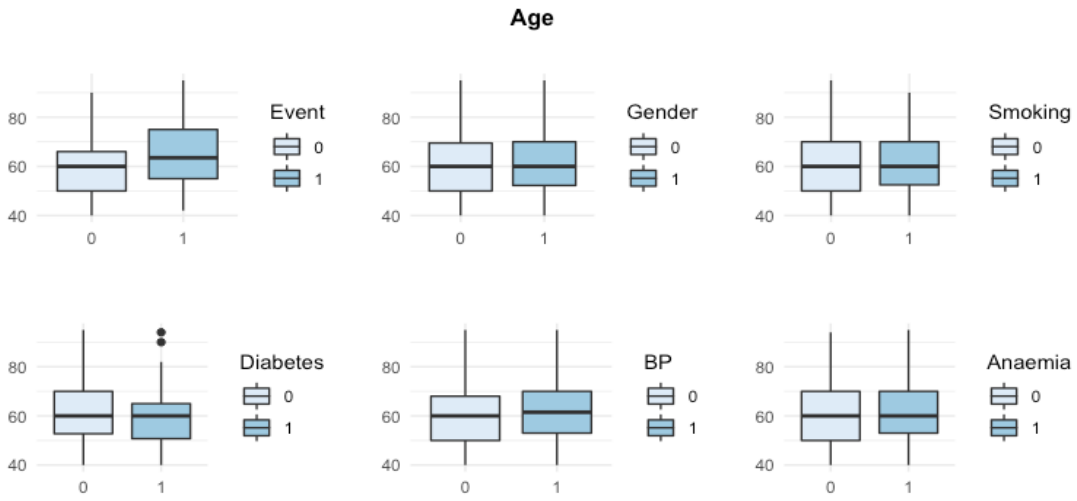
Continuous variables vs Event or categorical variables

Then, we looked at the relation between the pairs of categorical and continuous variables, included the *Event.* The scope of the following analysis is double:

a) to understand if the event of having an heart attack is related to different distributions of the continuous variables, i.e. if the population of people having *Event*=1 is the same of the one having *Event*=0;

b) to understand if and how the categorical variables are related to the continuous ones.

Age vs factors

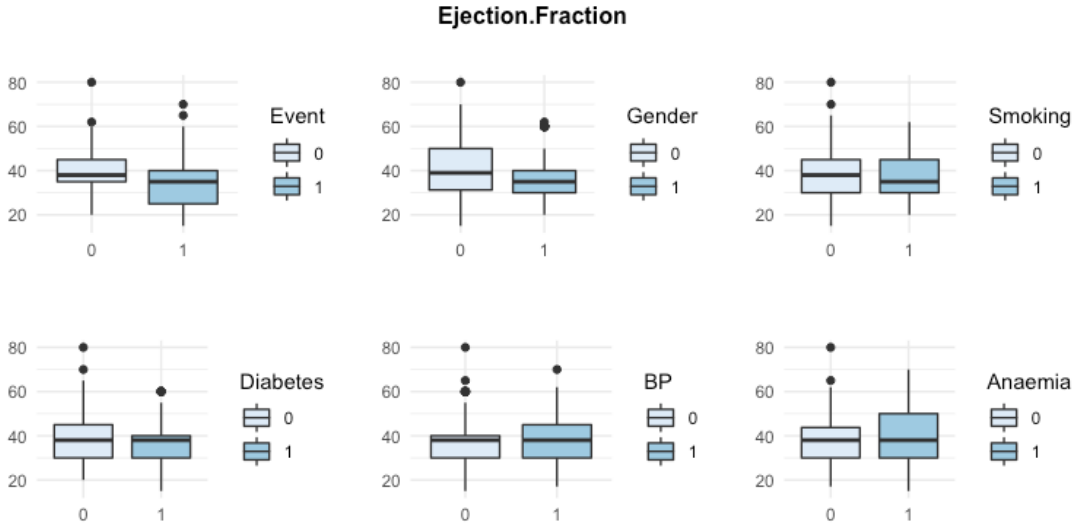


It seems that the population on which the event has occurred is older than the one that did not observed it. It is reasonable, since heart attack are generally more frequent in older people. For all the other variables, there is basically no difference between the groups.

Some points are more than 1.5 times the interquartile distance far from the median, but that are few and there is not enough evidence to consider them as outliers. Anyways, it is something that we should keep in mind.

As expected, the youngest observed age is forty.

Ejection Fraction vs factors

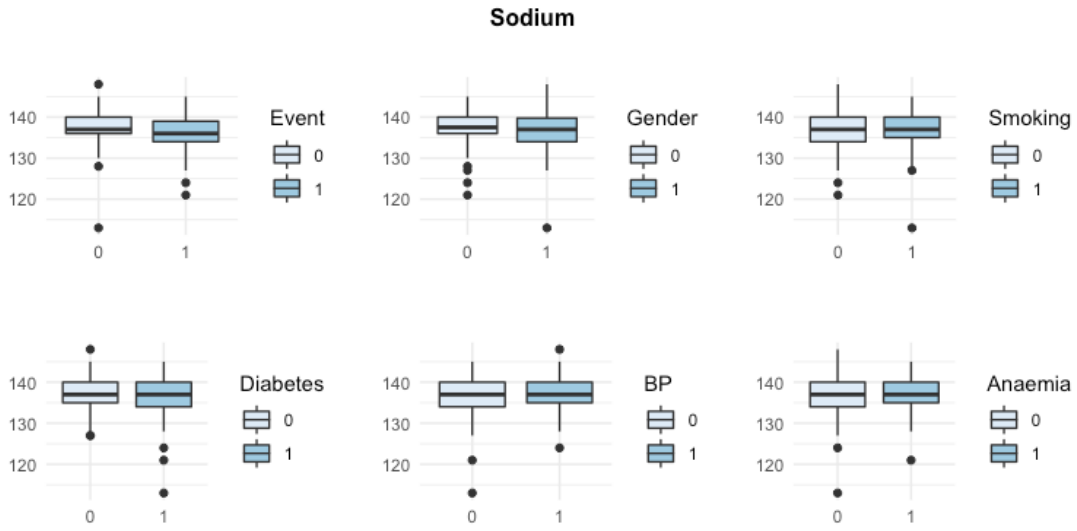


It seems that the population on which the event has occurred has lower values of *Ejection fraction*. That means that this variable can be useful to describe the distribution of the vent over the population.

About the relation between this variable and the categorical ones, we can say something similar to what we said for the *Age*.

For men and people with *Diabetes* the *Ejection fraction* looks to be lower. The remaining ones are not really related to this continuous variable.

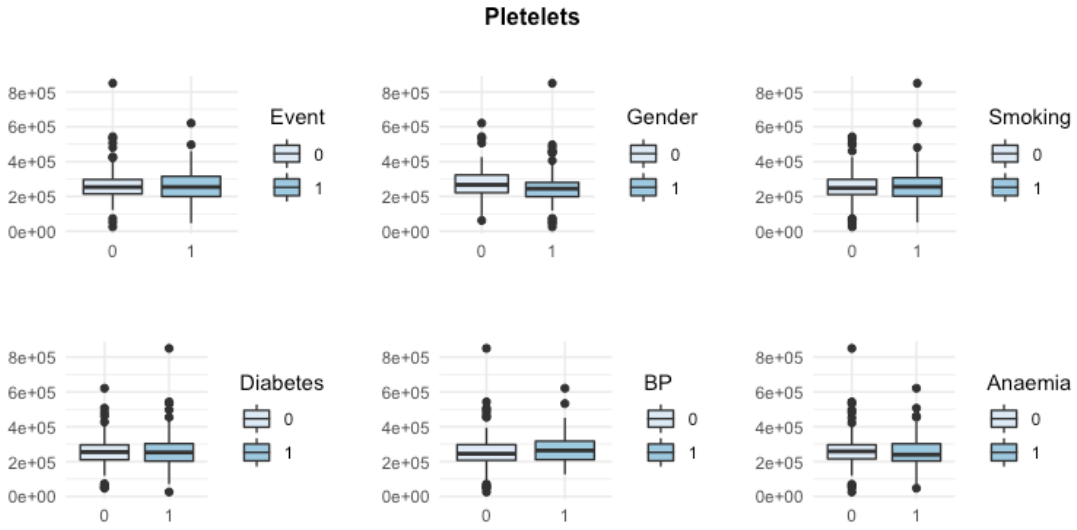
Sodium vs factors



These graphs show less regularity then the previous ones. In particular, some data present an unusual low value of *Sodium*.

Talking about differences between the distribution of the amount of *Sodium* in the two groups, there is not difference to be noticed. On the other hand, this variable does not look very useful in term of explaining the distribution of the *Event*, since the level of *Sodium* is almost the same for both the groups of people having had an heart attack while being under observation and people who didn't.

Pletelets vs factors

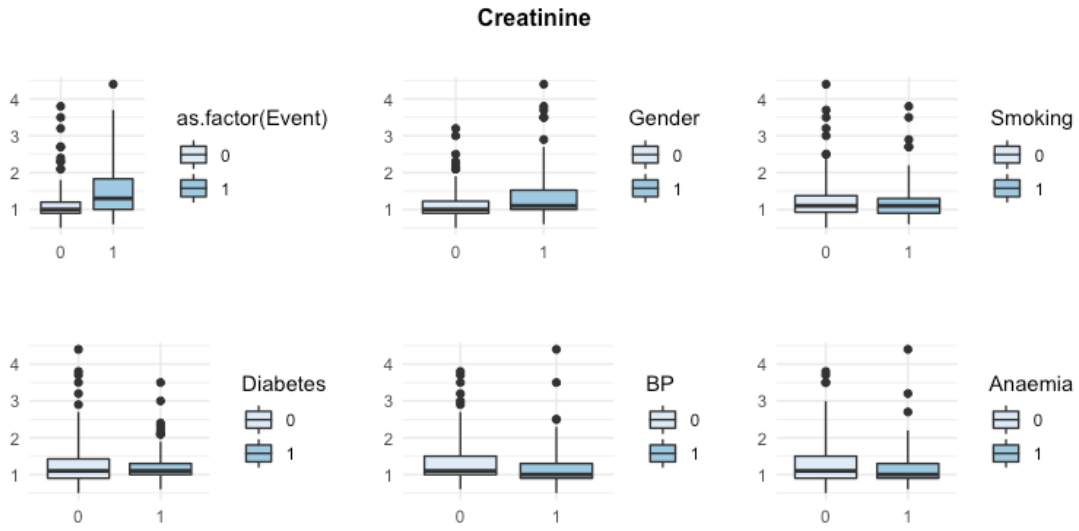


The *Pletelets'* value has not distribution changes between the people having had the heart attack and the others. This suggest that this variable is not really useful for our scopes.

Furthermore, we can notice that there is not difference in distribution between this continuous variable and all the others.

Creatininine vs factors

At the beginning of our analysis we noticed that the distribution of *Creatinine* is caraterized by a long right tail. In order to provide a meaningfull visualization, we removed those very high value. More specifically, we removed from the visualization observation with a level of Creatinine higher than 5.

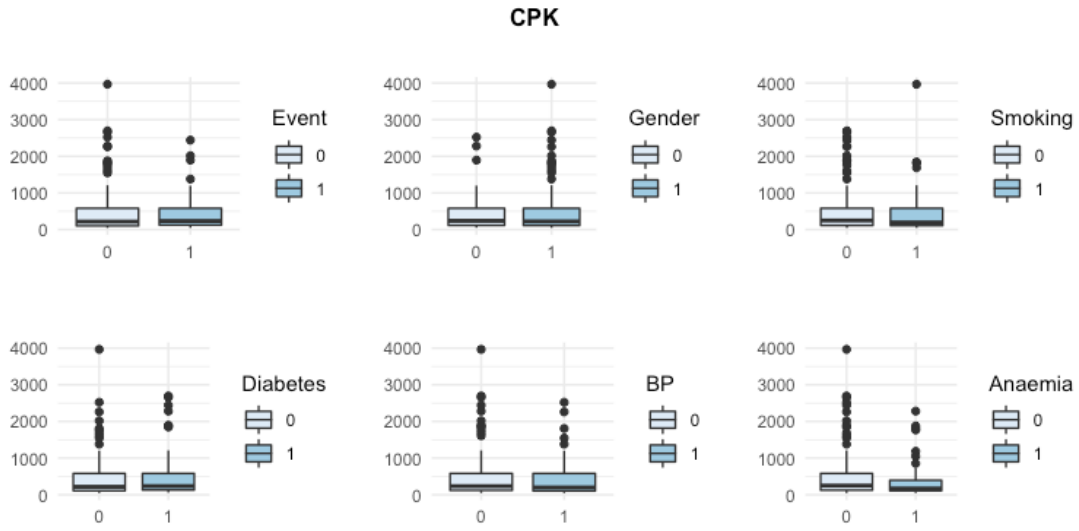


As saw before, the distribution of the *Creatinine* variable is uneven. Anyways, it seems that this variable can be considered between the ones useful to forecast the event, since people having observed the event during the observation time seems to have higher value of *Creatinine*.

On the other hand, the other categorical variables do not present differences in distribution in the two groups identify by the two levels of each categorical variable.

CPK vs factors

Before analyzing the level of enzyme *CPK*, we had to apply the same process we used with *Creatinine*. In fact, its distribution is unbalanced. The relation between the distribution of two of them can also been seen through the following graphs:



Even the distribution of CPK is unbalanced to the rigth, with a lot of big values. Unfortunately, the *CPK*/*Event* boxplot shows that the former variable is probably not really useful to explain the latter one. The relation between the level of CPK and the other factors is the same of the *Creatinine* variable.

From this latter graphs, we can conclude that the relation between these two variables (*Creatinine* and *CPK*) is very strong. The difference between the two of them is only in the effect that they have on the distribution of the *Event*. While the *Creatinine* level looks to have an effect on the heart attacks, the *CPK* does not.

We should take into account this observation while performing variable selection.

Pairs of continuous variables and Event

The next step is to look for patterns between pairs of continuous variables and the fact that the event was observed (or not).

There are not natural clusters defined by these composition of variables.

Conclusions of the explorative analysis

To summarize, we have seen that:

- there is a strong relation between variables *CPK* and *Creatinine*;
- the level of *Creatinine* looks to have an high impact in the *Event*;

- the level of *Sodium*, the *Age* and the *Ejection Fraction* look to have a medium impact on the Event variable;
- the number of *Pletelets* in the blood, the level of *CPK* and all the categorical variables seem to have no impact on the *Event*;
- there are not natural clusters defined by pairs of continuous variables.

All these aspect are fundamental to have a first overview of the information contained in the data. It does not mean that *Creatinine*, *Sodium*, *Age* and *Ejection Fraction* will certainly be the variables that will be used in the classification models, but it will help us in better understanding the nature of the case-study.

SURVIVAL ANALYSIS

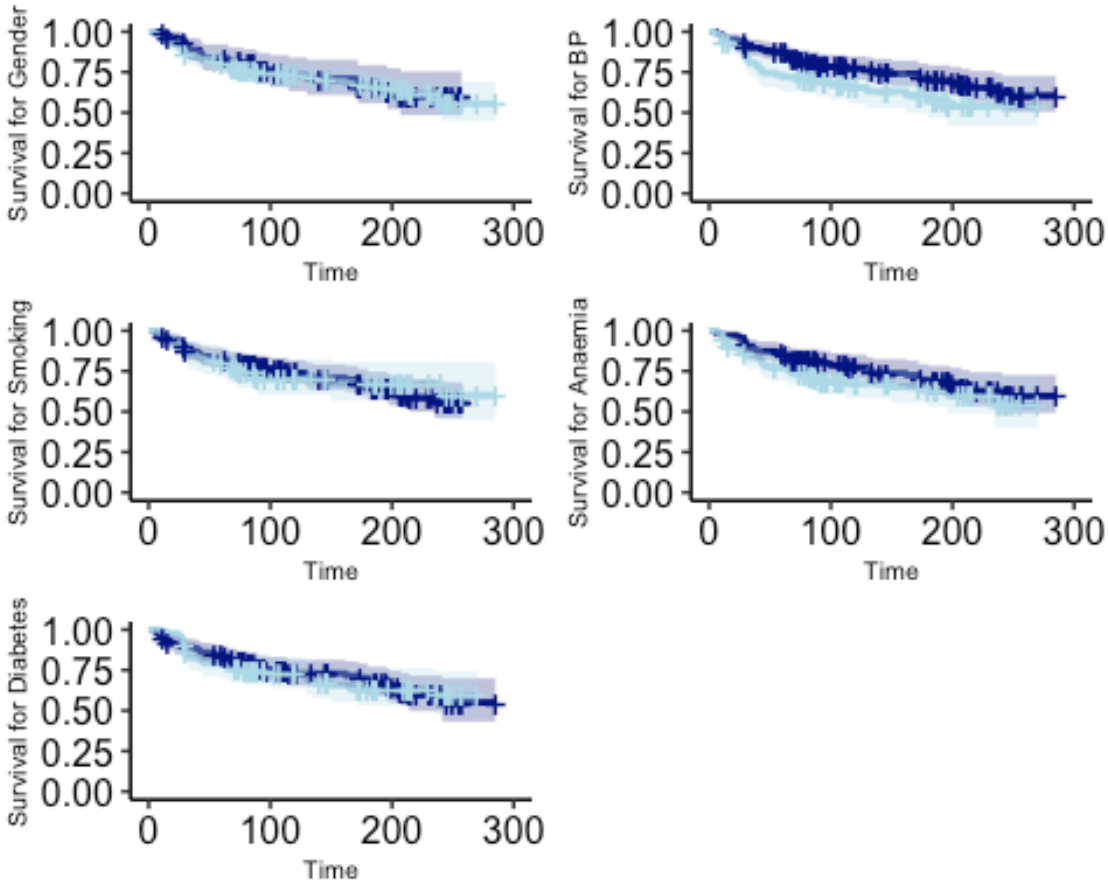
As said in the introduction to the data, this dataset has a specificity regarding the way we the variable *Event* has been collected. More precisely, each patient had been under observation for a certain number of days (variable *TIME*). This means that the *Event* could be observed only under the period of observation of each statistical unit. This considerations led us to use the traditional approach for data of this type, i.e. survival analysis. Our main idea is that this kind of approach will better explain the behavior of the data, leading us to a deeper comprehension of the relation between the variables and the *Event* and, most of all, to a better choice of the features to include in our classification models.

Non-parametric analysis

In the survival context, the first thing to do is to look at the survival curves using the Kaplain-Meyer method (more details at *Goel, Manish Kumar et al. "Understanding survival analysis: Kaplan-Meier estimate." International journal of Ayurveda research vol. 1,4 (2010): 274-8. DOI: 10.4103/0974-7788.76794*).

Individual effect of the factors

First, we evaluated the dichotomous variables.



Each pair of survival curves defined by different levels of the categorical variables have confidence intervals that overlaps, as it can be seen throw the fact that the dark blue area and the light blue one always overlap. This means that, from a descriptive point of view, these variables do not have effects on the answer. This is in line with what we saw in the previous explorative analysis.

Joint effect of the factors

We can perform a general test to see if the combination of all these factors, taken together, has significant effect on the survival curve.

Previously, we saw the comparison between survival curves of groups of people defined by each of the factors individually taken. From those graphs, we could notice that the lines generally do not intersect. On this base, we can reasonably use a *logarithmic rank test* for the hypothesis that all the survival curves defined by all the combination of the possible values of the binomial variables are equal. The overall p-value of this test is 0.11. It means that, even if individually the variables does not look to be very useful, **their combination can be used to explain the behavior of the heart attacks among the population of interest** .

To conclude, we can say that each of the categorical variable does not seem to have impact on the event if individually considered, the combination of all the categorical variables seems to have impact on the event.

These results imply that, while doing variables selection and, in general, building our models, we will have to pay attention in finding the set of variables which the joints effect is explicative, even if the individual ones are not.

Survival models

We can now apply survival regression models to better understand the usefulness of the different variables in explaining the survival ratio. In other words, we will make some assumption over the data, apply a parametric model that can be build on the top of those assumptions, check the goodness of this model and, if the measures are indicating that it well fit the data, extract conclusion from its output.

It should be noticed that we are still working in a descriptive paradigm. Our scope is not to provide previsions on future observations, but only to understand, using the training set, which are the useful variables. We applied a **Weibull model**.

Assumptions:

- the residuals follow a Weibull distribution wikipedia reference;
- the risks of observing the event for different groups (defined by different levels of the factors) are proportional;
- the hazard ratio is constant (i.e., the risk for a person with certain levels of the explicative variables divided by the risk for a person with different levels of those variables is constant)

Model: $S(time) = e^{-\lambda time^{\alpha}}$

First, we trained the model with all the variables. The optimal starting model would be the one with all the possible interactions since we noticed in the previous non parametric analysis that some of them are ineffective if individually considered but could be important if jointly taken into account. Unfortunately, the small amount of data we have prevent us to apply the complete model. We will then manually adapt some forward selection to individuate the useful features and discard the others. The selection is semiautomatic based on the Akaike Information Criteria.

```
##                   Value Std. Error     z    p
## (Intercept)        8.54       0.93  9.19 0.00
## Gender            -0.07       0.31 -0.22 0.82
## Smoking           -0.12       0.30 -0.42 0.68
## Diabetes          -0.29       0.27 -1.09 0.28
## BP                -0.55       0.26 -2.11 0.03
## Anaemia           -0.34       0.25 -1.35 0.18
## Age               -0.05       0.01 -4.05 0.00
## Ejection.Fraction  0.04       0.01  3.53 0.00
## Creatinine        -0.32       0.08 -3.97 0.00
## Pletelets          0.00       0.00 -0.42 0.68
## Log(scale)         0.05       0.10  0.46 0.65
```

The variables *Gender*, *Smoking*, *Diabetes* and *Pletelets* are not significant for this model. Before discard them we tried, for each of them, to see if their interactions with the other variables are to be kept in the model. The only significant result to be *Smoking:Gender* and *Smoking:Age*.

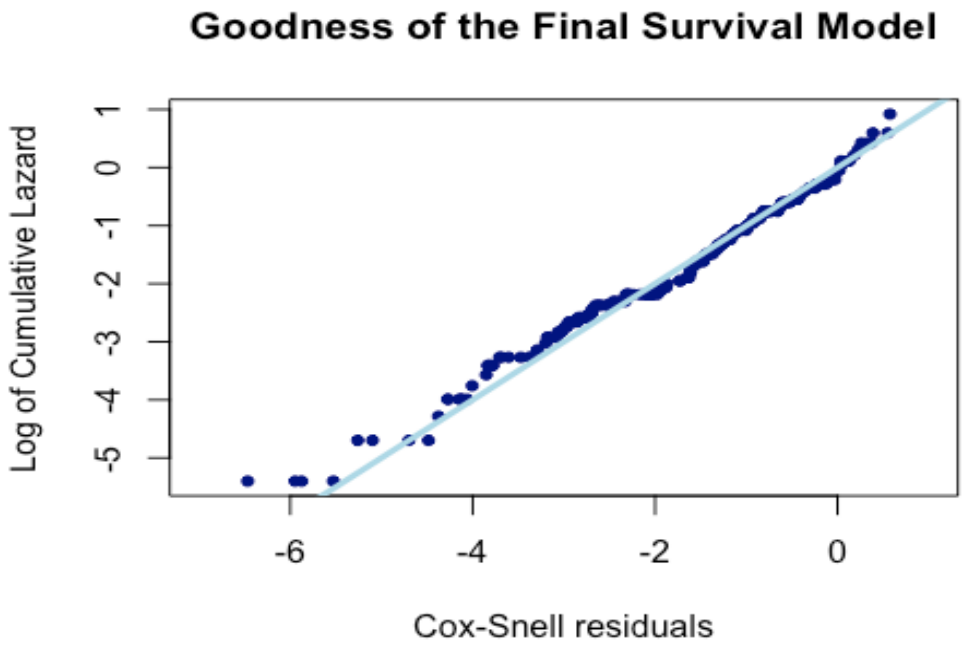
We obtained a scale parameter not significantly different from 0. It means that the (easier) exponential model would be a better choice for this data. Based on this observation, we then decided to use it: $S(time) = e^{-\lambda time}$. The results in terms of variables to be kept are the same.

We then have the following model, with these p-values:

```
##                 (Intercept)                    Pletelets
##                        0.26                         0.03
```

```
##                             Age                        Diabetes
##                            0.02                            0.00
##                              BP                Ejection.Fraction
##                            0.05                            0.00
##                          Gender                   Smoking:Gender
##                            0.02                            0.00
##                      Smoking:Age                Gender:Pletelets
##                            0.00                            0.00
##          Age:Ejection.Fraction   Ejection.Fraction:Diabetes
##                            0.00                            0.00
## Ejection.Fraction:Creatinine
##                            0.00
```

Usual methodologies to asses models' goodness are not appropriate for these family of algorithms. In fact, usually there is no the concept of time under observation to be taken into account. This exponential model, on the other hand, deal with that time. For this reason, we performed the assessment of the fit of the final model using the **Cox-Snell residuals**. The approach allowed us to verify about correctness of the distribution assumptions. The idea is that the relation between these residuals and the logarithm of the cumulative hazard ratio estimated using the Fleming-Harrington method is linear. We then plotted these two variables obtaining a confirm about this linear relation. It means the the assumption of exponential distribution is quite fine.



The process through which we selected these variables was, as already mentioned, purely based on the AIC. It leaded to a model were there are iterations of terms that are not individually present. In order to increment the interpretability of the results, we decided to add those variables anyways.

Meaning of these interactions

- *Smoking* and *Gender*: the data is collected in Pakistan. It could be reasonable to think that there are not collected variables that could distinguish women that smoke from the ones that does not. For example, it could be that women that smoke are from specific social subgroups where other risky

behaviors are more frequent than in the average men population (e.g., the women that smoke are western women with different habits from Pakistani people).

- *Gender* and *Pletelets*: see Gender-based differences in platelet function and platelet reactivity to P2Y12 inhibitors Ranucci M, Aloisio T, Di Dedda U, Menicanti L, de Vincentiis C, et al. (2019) Gender-based differences in platelet function and platelet reactivity to P2Y12 inhibitors. PLOS ONE 14(11): e0225771. DOI
- *Age* and *Ejection Fraction*: the increment of the ejection fraction leads to a decrease of the survival ratio (of -0.0035857) for each more year of age. This result in completely aligned with experimental results (Chuang, Michael L et al. "Association of age with left ventricular volumes, ejection fraction and concentricity: the Framingham heart study." Journal of Cardiovascular Magnetic Resonance vol. 15,Suppl 1 P264. 30 Jan. 2013, DOI.
- *Ejection Fraction* and *Diabetes*: the increment of the ejection fraction leads to a decrease of the survival ratio (of -0.08294) for people with diabetes. This is out of the range of our knoweldge and after some research we were able to say that this is still an open issue (Ehl NF, K??hne M, Brinkert M, M??ller-Brand J, Zellweger MJ. Diabetes reduces left ventricular ejection fraction–irrespective of presence and extent of coronary artery disease. Eur J Endocrinol. 2011 Dec;165(6):945-51. DOI Epub 2011 Sep 8. PMID: 21903896.)
- *Ejection Fraction* and *Creatinine*: the same as before.

CPK and Creatinine

1) In the previous analysis we saw that *CPK* and *Creatinine* have similar distributions. On the other hand, the correlation between the two is very low: $Corr_{CPK,Creat}$ =-0.085.
2) The explorative analysis showed that *CPK* doesn't seem useful to classify the *Event*.
3) In addition, we just saw that *CPK* is not even founded to be useful neither using the Kaplain-Meyer method nor the Exponential models.

Based on these three observations, *CPK* variable looks to be not relevant for the analysis. In addition, we applied a generalized linear model with only *CPK* and *Creatinine* to explain the *Event*:

The scope was to answer the question if both variables were useful if jointly used to (linearly) classify the Event. To look at this kind of linear dependence we have generated two models:

- $logit(Event) = \beta_0 + \beta_1\ CPK + \beta_2\ Creatinine + \varepsilon$
- $logit(Event) = \beta_0 + \beta_1\ CPK + \varepsilon$

The significances of the parameters of the first model are, respectively: 0.05, 0.

Before definitely dropping *CPK* as a variable for the classification of the level of Event, we performed two other tests to see even over the linear dependency. In order to do that, we built the following model: $logit(Event) = \beta_0 + \beta_1\ CPK + \beta_2\ CPK^2 + \beta_3\ CPK^3 + \varepsilon$

All these coefficients where not significantly different from 0. In fact, their p-values are respectively: 0.31, 0.2, 0.18.

On the base of this, we decided to drop *CPK* from our dataset. In our context, it means that *CPK* is not considered useful in inferring the *Event* of interest if combined with *Creatinine.*

Conclusions

In conclusion, it can be said that the variables that significantly directly impact the mortality curves are *Pletelets*, *Age*, *Diabetes*, *BP*, *Ejection.Fraction* and *Gender*. *Smoking1* and *Creatinine* have impact only if combined with some of the others. Relying of these results, we can create some new features on the top of the original ones. More precisely, we are generating new features representing the iterations that were founded as significant by the Exponential model: *Smoking:Gender, Smoking:Age, Gender:Pletelets, Age:Ejection.Fraction, Ejection.Fraction:Diabetes, Ejection.Fraction:Creatinine*.

CLASSIFICATION

After having performed some statistical analysis, we want to follow two different paths using Machine Learning to discover which of the two achieves better results in our test data set. These paths are: classification using the original variables, and classification using the new variables obtained from the statistical analysis. The same algorithms and decisions regarding the models will be used in order to make the results as comparable as possible.

We used R package mlr as a framework, which contains all the needed functions for our project. We then start by creating the task and the learners. We will be using repeated Cross Validation in our 75% of training data set.

If the sample size is small, which is our case, it is recommended to use repeated Cross Validation, as it achieves a good bias-variance balance and, given that there are not many observations, the computational cost is not excessive. In this case, we will use 2 folds with 5 iterations, which is not equivalent to 10-fold-cross-validation. The issue with partitioning data with few instances is that results may depend on luck. That's why we are considering 5 different partitions with 2 folds, so results will not depend on chance, but the average will be closer to reality.

As for preprocessing models, we used the package mlrCPO which creates a pipeline of operations. This way, the operations are applied to the learned without information leakage. In our case, we scaled all variables in a range from 0 to 1. Some of our variables are dichotomous, so they already have that range, and some of the learners do not support this type of variable. Therefore, when converting them to numeric, we wanted all our variables to be in the same scale. This way we can perform the training with all the chosen models at the same time. And as a result, parallelization can be executed for this task.

Classification Analysis

We will not consider the variable *TIME* from now on because it would not be information that we would obtain from a new patient, it is only a control variable.

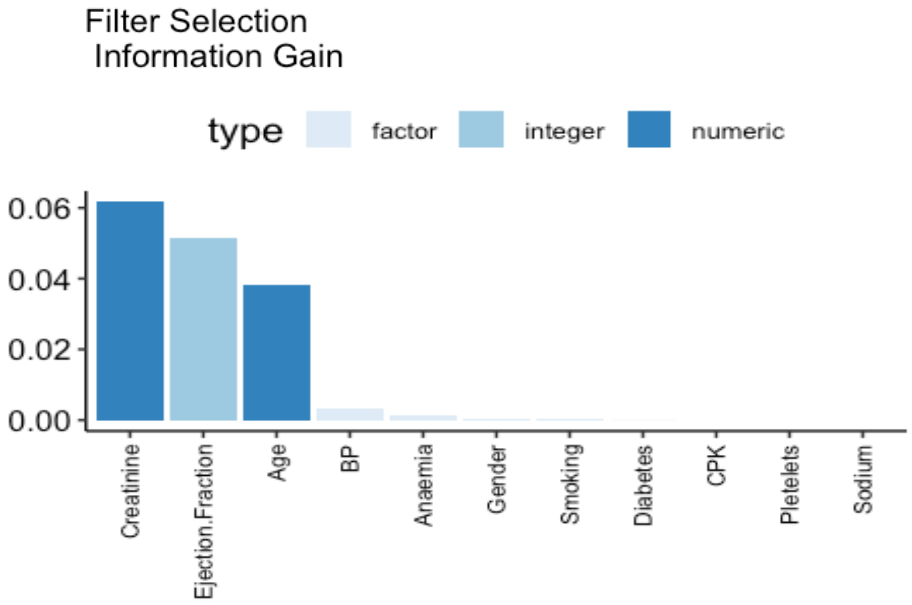
We started by training the following models: *generalised linear model*, *Naive Bayes*, *K-nearest neighbours*, *decision tree*, *random forest*, *ranger* (a fast implementation of random forest), *extreme gradient boosting* and *neural network*.

Filter Selection

In order to decide which features would be selected to see if they help the models get better predictions, we will perform filter selection. Unlike feature selection, this method does not require of a learner to reach a conclusion. We have checked that depending on the learner selected, results change drastically. Filter selection was found to be a more impartial methodology.

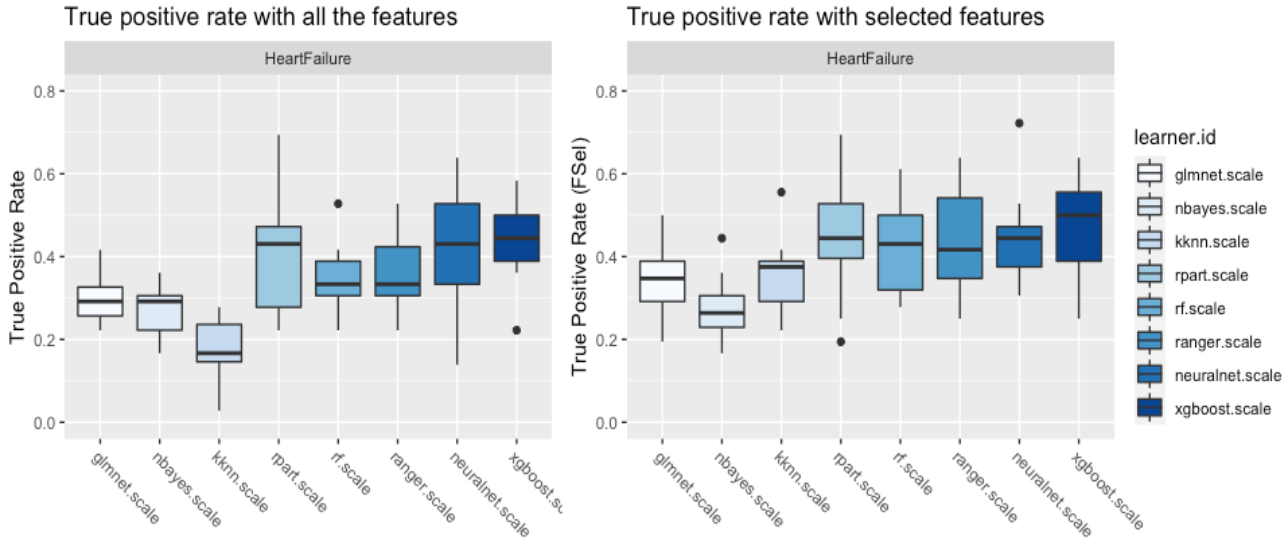
The metric used for this filter selection is *Information Gain*. It measures the reduction in entropy (or surprise) by splitting a dataset according to a given value of a random variable. A larger information gain suggests a lower entropy group or groups of samples, and hence less surprise. Reference

Entropy quantifies how much information there is in a random variable, or more specifically its probability distribution. A skewed distribution has a low entropy, whereas a distribution where events have equal probability has a larger entropy.



We saw in the BMC article that one of their conclusions was: "Our results of these two-feature models show not only that *serum creatinine* and *ejection fraction* are sufficient to predict survival of heart failure patients from medical records, but also that using these two features alone can lead to more accurate predictions than using the original data set features in its entirety." This is consistent with what we are showing here, with the difference that we added the *age* as well.

We then compared the results of the benchmarking performed using all the variables or only the selected ones.



Here, our aim is to maximize *TPR*, because in the medical area it is better to have false alarms (overestimating positives) than not predicting real positives. Some models may have a good accuracy (even better than other models), but that is not the priority in healthcare analysis (such as predicting cancer and other diseases). For instance, a trivial model (e.g. predicting all patients as class 0), would give very high accuracies, not taking into account any of the data provided. That's why we will decide which model is better based on *TPR* (true positive rate). On the other hand, a model that always forcast TRUE will always have a *TPR*=1. To avoid that, we also took into account the accuracy just as a way to balance the results. This means that the models we will find will not be the absolute best in terms of *TPR*, but will avoid situations of huge overestimation of the positives.

In this stage, we would like to decide whether we use the whole dataset or the subset obtained selecting some features. From the results we can comment on four main points:

- Weak learners perform visibly worse, some of them even reaching a tpr of around 0.2.
- K-nearest neighbor performs better when selecting few features. This can be explained because the algorithm does not work well with high dimensions. If fact, if a dimension is not explicative, it will influence the forecast in a direction that might not be the correct one. Performance changed from 0.18 to 0.36, with a default of 7 neighbors.
- We saw that overall models perform slightly better with the feature selection. However, results vary much more depending on the partition. For this reason, we decided to perform tuning for both datasets using a subset of the best performing learners. We will be tuning mainly "strong learners" (i.e. excluding *glm*, *naive bayes*), because we believe they will be able to discard useless features and reach better performances while being consistent, not fluctuating much.
- Random Forest (and Ranger) increase its performances as expected. In fact, choosing for each tree the variables that are to be used only between a set of clearly more explicative ones will produce, on average, better results.

Tuning models

To continue, we tuned different hyperparameters for the following models: *kknn*, *rpart*, *random forest*, *ranger*, *extreme gradient boosting*, and *neural network* for models with all the variables and the ones built with only the selected ones. Below we will explain the reasoning behind selecting each parameter. The same validation method as for the previous analysis will be used, i.e. Repeated Cross Validation with 2 folds and 5 iterations. For the hyperparameter optimization, random search will be executed. It is shown to be more efficient in a paper by Bergstra, J. and Bengio, Y. (2012) Reference. A total of 500 iterations will be splitted among the 6 learners chosen for tuning.

The best model is a Random Forest (Ranger) with ntree = 625, mtry = 1 and nodesize = 35 built on the filtered features (see below for more details about the parameters). For more details about the models check the code. Thanks to this result, we selected this model as representative for this section.

Explanation of parameters

- KNN:
  - *k*: the number of neighbors used for the prediction.
- RPART For the model *rpart* we are tuning two of the "stopping" parameters in the algorithm, that tells the tree when to stop growing (a way of pruning):

  - *minsplit*: minimum number of observations that must exist in a node in order for a split to be attempted. If it is too small, the tree will keep growing and probably lead to overfitting, if it is too big the accuracy may decrease.
  - *maxdepth*: Set the maximum depth of any node of the final tree, with the root node counted as depth 0. Again, if we let the tree make too many splits, it will lead to overfitting. A tree that is too small may generalize too much.
- RANDOM FOREST:
  - *ntree*: this should not be set to too small a number, to ensure that every input row gets predicted at least a few times.
  - *mtry*: number of variables to possibly split at in each node. (Cannot be bigger than the number of variables).
  - *nodesize*: Setting this number larger causes smaller trees to be grown (and thus preventing from overfitting and takes less time)
- RANGER: Same parameters as in random forest, since it's the same algorithm, just optimized.
- XGBOOST:
  - *eta*: control the learning rate. Used to prevent overfitting by making the boosting process more conservative.
  - *nrounds*: max number of boosting iterations.
  - *max_depth*: maximum depth of a tree
- NEURALNET:
  - *hidden*: hidden neurons, for each layer (only one layer in our iterations). To help the algorithm converge when enlarging *hidden*, we need to make the *threshold* and *stepmax* bigger as well.
  - *threshold*: threshold for the partial derivatives of the error function as stopping criteria.
  - *stepmax*: maximum steps for the training of the neural network, if we make it bigger, we let more time for the algorithm to converge.

Test result

The **true positive rate on the test** set obtained with the Random Forest having 625 trees and minimum node size of 35 using the filtered variables is **0.67**.

Survival Analysis

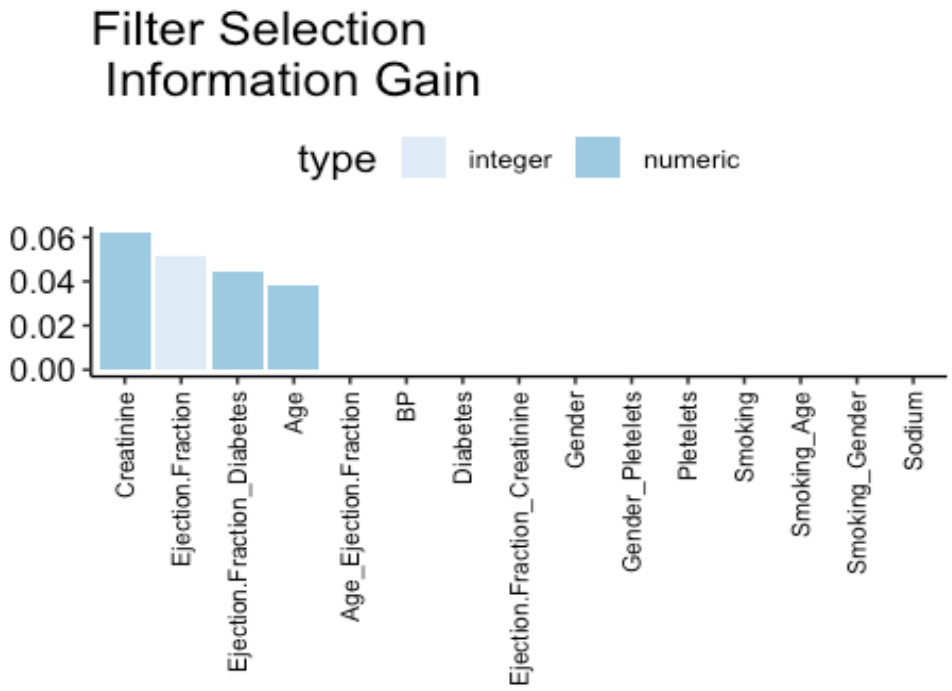
In this section we want to do a comparison of the two different approaches, statistic-based and generic classification. To do so, we have independently chosen the most important features given by each approach, and we will be performing the same machine learning analysis to see which gets a higher true positive rate. Every parameter (e.g. validation methodology, search space) will be the same as the previous analysis.

Tuning models

We tuned the same models as before, with the same ranges for their parameters in order to obtain the best performing one.

The result showed us that the best model using the data obtained from the survival analysis is a Random Forest with 345 trees, nodesize = 20 and mtry = 2. The **test tpr** on this model is **0.71**.

As done in the previous section, it is possible to perform feature selection based on the entropy of each variable.



The best model built on the top of these variable is a Random Forest with 344 trees, nodesize = 34 and mtry = 2, that is pretty similar to the previous one. The **test tpr** on this model is **0.75**. This suggested us to choose this model as representative for the survival approach.

Test result

We were then able to test the best model selected in the previous paragraph.

The true positive rate on the test set obtained with the best model selected for this section is 0.75.

CONCLUSIONS

First, we performed an explorative analysis that helped us in understanding the data through the correlation between the variables, their individual and combined distributions and the similarities between them. Thanks to that, we were able to exclude one useless variable from our dataset.

We then performed a non parametric survival analysis, realizing that the interactions between variables were in some cases useful in explain the *Event* even if the variables taken individually were not.

Based on those results, we were able to build an exponential survival model that we used to select some interactions and drop some variables. On this basis, we generated some new columns in our dataset by combining the existing features.

In parallel, we applied some classification models on the starting data and on a subset of them (in terms of features) obtained through an entropy measure. We observed that the best model in terms of true positive rate was a Random Forest built only on some selected columns. The best TPR on the test set was of 0.67

We then applied the same algorithms to the data modified on the output of the survival model, obtaining as best result a Random Forest that gave us a TPR on the test set of 0.75.

We concluded that the usage of a survival model to do feature engineering leaded to better results.

POSSIBLE EXTENSIONS AND KNOWN LIMITATIONS

As we have mentioned previously, we are aware that our study has some limitations.

One of the main restrictions is the fact that only 299 instances are available to perform the analysis. Having a small data set can lead to results being sensitive to different partitions. This means that changing the seed may sometimes change the accuracy on a study. A clear extension in this sense would be to merge this data set with another related one, making sure that the same procedures were followed in the selection and collection of patients' data.

Another limitation that has already been mentioned throughout this project is the low correlations between the predictors and the response variable. Although this is one of the reasons for researchers to be interested in the combined use of traditional techniques with other alternatives, such as machine learning, it is still a limitation for the latter one. This restriction could be addressed by the already mentioned in the previous paragraph, with the aim of having a broader image of the relation between heart failure and the other variables; and the possibility of adding other features that could make predictions more accurate.

All in all, we believe that, given the limitations that this data set presents, we have conducted a thorough analysis of a rather complex field, performing two different approaches and obtaining good results, that may be boosted with the aforementioned extensions.