Data Mining and Decision Systems  
600092  
Assigned Coursework Report

Student ID: 201707824  
Date: 15th November 2019

## Due Date: 12 December 2019

**Report must be within 8 page maximum. Strict page limits will be enforced. Any extra pages will be ignored and no marks awarded for any work on these. Exclusions to this limit are the front page, the references section, and any appendices. Please keep to the given section headings and format; subsections are permitted.**

# **Methodology**

The methodology followed for this report will be a slightly modified version of the CRISP-DM methodology. In this instance, we do not have the first and last stages of the methodology, which include the **business understanding phase** and the **deployment phase**, so these will be included in a hypothetical sense rather than a more general sense.

The task is to create a classification model with the dataset supplied. The data is provided via the domain of Cardio-Vascular medicine. The aim of this is to identify if a patient is at risk or not at risk. The model will be of a binary nature as we are working with an if or else classification, also known as a 0 or 1 classification mode (binary classifier).

### **Data Understanding Phase**

The first step is to load the data into a pandas data frame and then use the describe function to look at the columns, which can be seen at output lines [2] to [6] of the notebook code.

The process of Data Understanding is to Describe and Explore the data. This is done by observing the differences between what is in the Data Description, and then what is present within the data. These observed differences are then compared and adjustments are made. The adjustments must be justified however, as it is important to maintain data integrity throughout this process.

Any observed differences be that missing values or values not being unique where expected are explored and removed in the next phase of the methodology. Early exploration shows that the total values for the dataset are 1520, with mostly categorical attributes. Each patient has an ID which indicates their visit, but as they can have multiple visits, this number is not necessarily unique, but is it expected that Random be unique in nature. Missing values can be seen to be present, evidenced by differing column value maximums and will be the focus of the data cleaning from the offset.

In the data given, most values are noticeably similar, most of the attributes with value type nominal are classed as objects within the data frame. Random is a float and ID is an integer, which correlate correctly. IPSI is a float while expected to be a float, which is fine, but Contra is an object, this attribute should be an int, as it contains numeric values. This can be changed through code in the next phase to instead be a float through the pandas **to numeric** function.

### **Data Preparation Phase**

Code blocks [7] to [23] of the notebook code show the data cleaning steps.

Referring to the numerical values Random and ID, they are expected to have unique values, and whilst their numerical nature is great for the business to identify patients, that is not in the scope of the task here.

For the task, all features will be used besides Random and ID. This is because, while useful for indicating individual patients and their scores, the attributes themselves don’t have suitability towards the end goal and therefore are the features for exclusion. As the model will be of a binary nature, these columns indicate high numerical values and can thus cause incorrect classification later.

The data will be cleaned in steps, looking at the patterns inspected from the earlier phase and putting these into practice. This can be seen in more detail through the code found within the accompanying notebook. The data will be changed in places where it is deemed necessary, starting with the label category.

This attribute was found through initial exploration to contain 3 unique values. The same can be said of Indication which has 5 unique values when it should only have 4, but this is explored after duplicates are removed. The third unique value of label is Unknown, this will be changed into a numpy value of NaN, which stands for not a number. The reason why this must be changed to a numpy value of NaN, is that simply replacing the value will mean that when the function drop.na is called, the renamed NaN is not dropped but all the others will be.

Once this change has been made, Random and ID are dropped and Contra, discovered earlier to be in the wrong format, is changed next.Contra is changed to be of a numerical type float64 using pandas at code block [13]. Simply changing Contra to be numeric caused an error however, which was resolved by applying errors=” coerce” to the code. This meant that there was an empty value that wasn’t being picked up and the conversion was failing due to this.

There is always the argument of imputing missing data or simply removing it. This depends on the amount of data loss, however. In code blocks [13] and [14] the null data is explored and is found to equal 20 values. Considering we have 1520 expected values and only 20 contain a null value, I decided to drop these values. There is an argument for imputation here, but due to this data being legacy, and the percentage of values missing being low (code block [14]), the data was removed instead.

The last needed change to make is the indication 5th unique value. This is simply a naming issue and is corrected quickly, requiring no further intervention beyond this.

#### **Visualisation**

Although visualisation is not an integral step of the methodology, it does fall in some respect to data understanding and is therefore useful to identify links in the data, expressing them in a more visualise fashion and then, using that to decide on the most important features of the dataset. It is useful to identify any trends and look at understanding the correlation of features within the dataset.

Visualisation is a key aspect of an approach known as EDA, or Exploratory Data Analysis. This is an approach to analysing datasets to summarise their main characteristics, often through visual methods. That is the reason behind the visualisation that takes place directly after the data is cleaned within the code.

#### **Data Formatting**

Data Formatting is necessary to ensure the data is of the correct format that models can use later. The two techniques I chose to use for data formatting were techniques called Normalisation and One-hot Encoding.

Normalisation is also known as feature scaling, where all numerical values are changed to equal between 0 and 1. This is done as to change the values to a more common scale, without distorting their values. This is done in the notebook at code block [35] using the normalisation formula. This is done to make sure all models I plan to use would be able to correctly classify risk vs no risk, without the need to remove nominal values, as these nominal values are important in reaching a result.

After this, the categorical values of the dataset need to be pre-processed, which is done by way of a sklearn library function called label encoder. This is done as to make sure all categorical values equate to a numeric value. A lot of the data within the dataset is categorical values of yes and no, which will become 0 and 1, equalling a more common scale that can be assessed by machine learning models.

Indication has 4 unique values, and thus pre-processing this attribute will lead to a number for each value assigned between 0 and 3. Code blocks [36] and [37] show the process of this. Once this has been done, the data is almost ready for the modelling phase. The last thing to do, is to split the columns from the column that indicates the result, which is the attribute label. The result must be in a different place otherwise the model will know the result and will therefore always achieve 100% accuracy.

### **Modelling Phase**

Modelling is the most important phase. The first step for this stage is separation of key data. In the notebook, once the data is pre-processed and normalised, it is split from the label into its own dataset. This is because the aim is to use the label as the target of the models that will be created and thus it needs to be elsewhere and not connected.

The data is split using a method known as the train/test split, which is a common approach where data is separated into X and Y, and then split 70/30. The X all data in the dataset excluding the target, and Y is the target. The data is trained on X, and then evaluated on Y.

70% of the data is used to train the model, and then the remaining 30% is used to evaluate the model, where metrics will be used to discover how accurate the model is. This split can be randomised if necessary, but this is not always a requirement.

For the notebook, the train/test split is done at blocks [39] and [40]. To test the data without creating multiple training/testing splits, I opted to run cross validation at 10 folds to analyse the accuracy metric and make sure the accuracy achieved is sufficient and then compare the accuracy of a technique that uses all the dataset rather than simply using only the testing portion, to the train/test split method, comparing where necessary.

There are many model’s worth using for this phase. I chose to go with a variety of different models. The reason behind the models I chose was either due to their popularity in the field, or their obvious suitability of the task. I ran tests and analysis on 6 models, including Neural Networks and a Decision Tree, due to their differing nature and usefulness in the field of Machine Learning.

TO BE CONTINUED – REVISE ABOVE MODELLING SECTION AND ADD MORE BEFORE MOVING FORWARD

Provide details on the methodology applied towards the data mining analysis undertaken, providing rationale for these steps.

This should detail how you went from the raw data provided to the chosen model(s), choice of model, and how this methodology helps address the problem domain.

Evidence to support the following of this methodology should be presented, especially any cases which required moving backwards in the process to readdress issues.

# **Results**

Results should include tables showing model performance with appropriately selected metrics. No rationale should be provided for this section - simply results of evaluative processes.

If using modified variants of the dataset, these should be clearly identified in the tables with appropriate naming. The justification and description of modification is not for this section.

Additional figures may be used as appropriate, in support of discussion points in the Evaluation & Discussion section, or as evidence for methodology following above.

# **Evaluation & Discussion**

Evaluation methodology used for generating the results provided in the previous section. How were these evaluated? Why was this selected? What metrics were used and why?

Discussion of the results should be presented with appropriate evidence and rationale. E.g Which is the best model, and why?

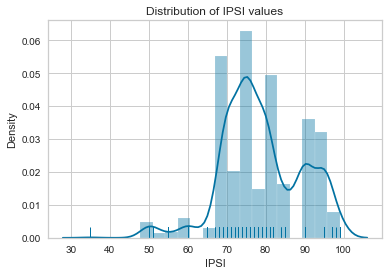
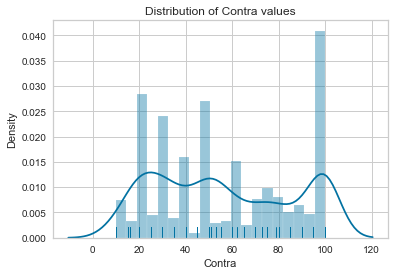
Consider each stage in the methodology, and reflect on any improvements which could have been made. Could any techniques have been used which may have improved performance? Why?

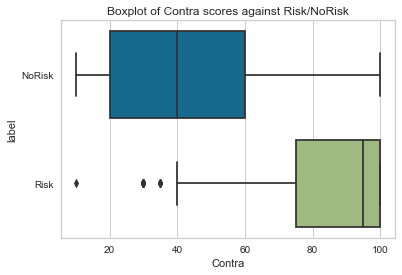
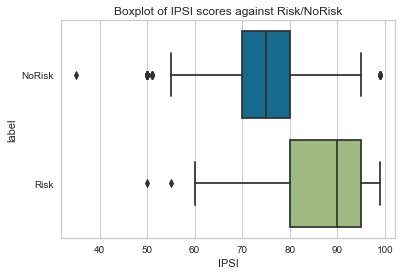
# **References**

Any references used throughout the report should be included here in Hull Harvard Style. If no references used, remove this section

# **Appendix**

##### Section A – Graphs & Boxplots





#### Section B – Report sections to be adjusted

From Data Understanding section

***There are observed differences, the most notable being the attribute ‘Contra’ not being numeric, instead being an object containing numeric values. The total values of the dataset are 1520, although some attributes have less records than this, which indicates missing or null values. There is also the attribute Indication, which has 5 unique values, this will need to be checked, as according to the data description there is only expected to be 4 unique values for this attribute. The attribute label has 3 unique values, rather than the 2 that are expected.***

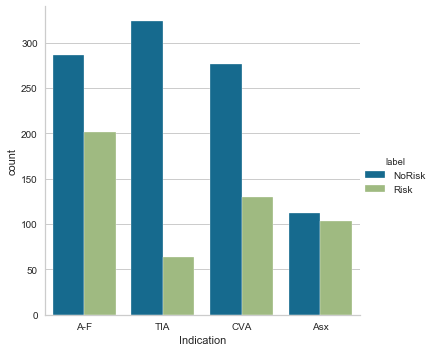
***The target attribute of the classification model will be label. This is due to all other attributes leading to a conclusion. The data description is also helpful, as the description for the other attributes indicate steps taken to discover the risk of mortality of the patient, therefore a suitable target for a binary classification model. The data appears to have slight differences from what is expected through the data description, but the differences are small and will not require much to be changed.***

***The first and most important step in the early stages of CRISP-DM is cleaning the data. The first task is to check the unique values that were found before that don’t add up with the data description starting with label. As shown in code blocks [7] to [9], there is a third categorical type in the label attribute of Unknown, which will be changed into a numpy value of NaN, which stands for not a number. The reason why this must be changed to a numpy value of NaN, is that simply replacing the value will mean that when the function drop.na is called, the renamed NaN is not dropped but all the others will be.***

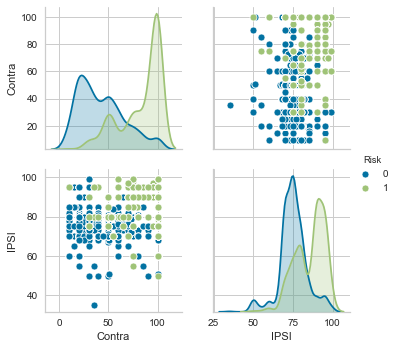
***Once this is achieved, the feature columns of Random and ID are dropped from the overall dataset (code block [11]), and then the feature***

***The Indication is corrected later at code block [17], which is discovered to be a duplicate name, one which has an uppercase and the other a lowercase spelling causing a split in attributes of the column. This is corrected by simply renaming one to be equal to the other.***

#### **Visualizing the data**

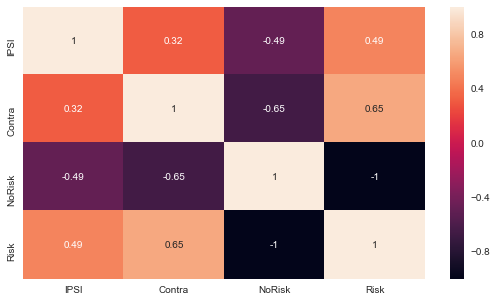
Data visualization is important to visualize the data you are working with, to discover any patterns or trends or lack thereof. The graph to the right shows the indication value count against the classification target.

This count is useful, indication is a nominal value that describes what type of event the patient had prior to hospitalization. This is important towards the end goal, because the indication has a bearing on the outcome, as it defines the seriousness of the incident.

The plots to the left show the two predominant numerical values of IPSI and Contra from the dataset.

Contra values have a higher range across the dataset, a higher Contra value is indicative of risk more than the lower values. However, IPSI values are generally right skewed as a whole, with the distribution of risk and no risk patients generally falling between 70 and 100. What is important to notice is not all high value IPSI patients have a risk outcome.

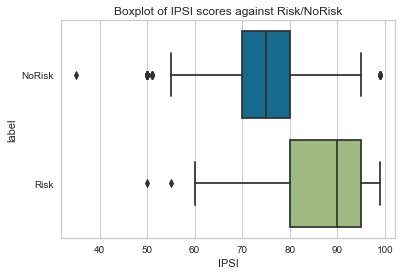
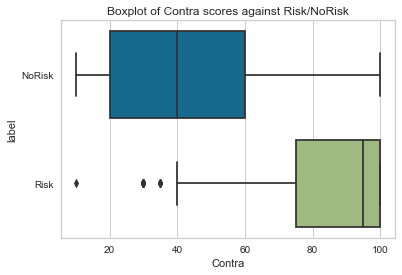
The plots above, visualize IPSI vs Contra in the top right and bottom left. They show that there is no true correlation between the two, this is explored more in a heatmap.

Correlation is a measure of the linear relationship or connection between two or more things. Continuing with this trend towards IPSI and Contra, it is important to consider the measure of Correlation.

The correlation heatmap shows a visual look at this. The important things to take from this, are the value of 0.65 for Contra against Risk, and 0.49 for IPSI against Risk. What this means is that, Contra has a 0.65 correlation with a risk classification, meaning that there is a stronger correlation with a risk classification than IPSI, which has a moderate correlation of 0.49. No Risk against these two valuations has the inverse effect, causing a decrease in the same numerical measure independent of the other.

The score for IPSI against Contra is 0.32, indicating a weak correlation between the two. The Distribution plots for IPSI and Contra can be found either in the code at blocks [28] and [29] or within the appendix. These plots aim to better distinguish the spread of the data based on these values.

We can better visualize the Distribution of values for IPSI and Contra within seaborn using a distribution density plot, which can be found in the Appendix at section A.



Boxplots are useful to see what the interquartile range of the data is like and if there are many outliers that should be investigated. In this case It is important to continue the exploration of IPSI and Contra against the Risk and NoRisk results and see if there are many outliers present. Boxplots are used here to further enhance the understanding of how their numerical values influence the end classification results.

Boxplots were also used to explore the spread of classifications against Indication type, found within the Appendix at section A. The boxplots section of the notebook goes through these in further detail.