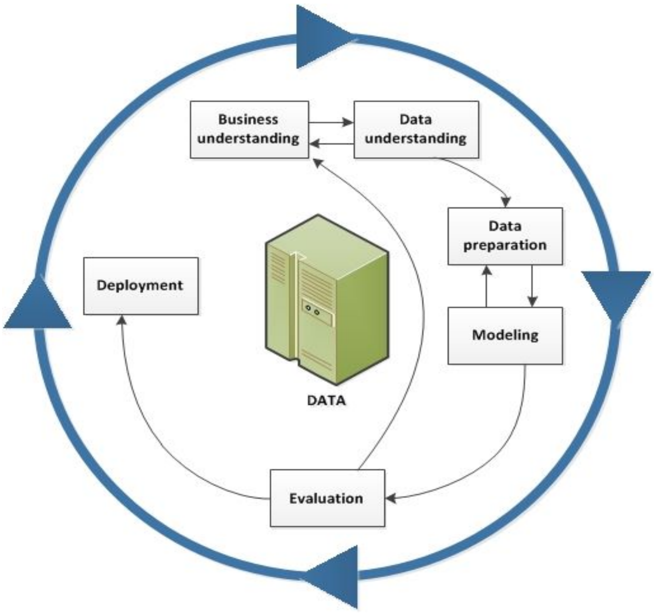
Kidney Failure Project

using the CRISP-DM Methodology



Date: 13 Enero 2018

Submitted by:
Arnau Alabort
Jose Manuel Paredes
Bernard Mickael
Alvaro Gallego

Business Understanding

Determining Business Objectives

Create an effective system to help the physicians of the nephrology department to identify patients that are more likely to suffer from kidney complications after a new treatment in the hospital.

1. Predict if the patient will develop or not kidney complications after the treatment.
2. Determine the most important variables that influence in having such complications.

Business Success Criteria

How we will assess if we have accomplished our goal or not.

1. Provide valuable information to the physicians on predicting if a patient will suffer or not from kidney failure complications.
2. Find the most important 5-10 variables of our prediction.

Assessing the Situation

1. **Resource inventory:**
    1. Data:
        1. drugs.csv
        2. kiney_fail_dataset_v2.csv
    2. Tools:
        1. R
        2. SPSS Modeler
        3. Any other free tool software for Data Analysis.
2. **Requirements:**
    1. Done by January 13th at 22:00. (Report, Deployment Manual, Files used)
3. **Risks:**
    1. Scheduling: Due the load of assignments from different subjects could be possible to do not achieve the deadline.
    2. Data: Could be impossible to achieve the results of accuracies wanted.

Determining Data Mining Goals

1. **Data Mining Goals:**
    1. Use both datasets to generate a **classification model** available to predict kidney failure.
    2. Analysis of the relation between prediction variables and the target variables to provide most important ones.

2. **Determining Success Criteria:**
    1. Accuracy >75% with the classification model on a Validation set.
    2. Find between 5-10 variables which achieve +-5% of the accuracy than using the whole dataset.

Data Understanding

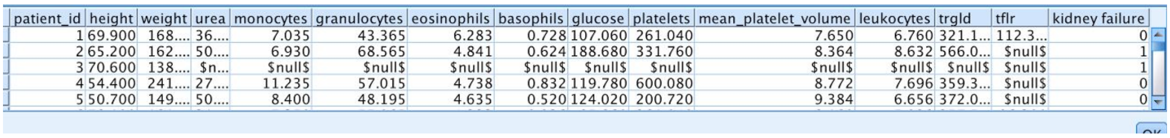
Collecting Initial Data

As we previously said, we will work with two datasets provided by the hospital.

1. **kiney_fail_dataset_v2.csv**: Dataset with different information of around 1000 patients who have finished the treatment indicating if they suffered from kidney complications or not.
2. **drugs.csv**: Dataset with the drugs took by each patient during the treatment.

Describing Data

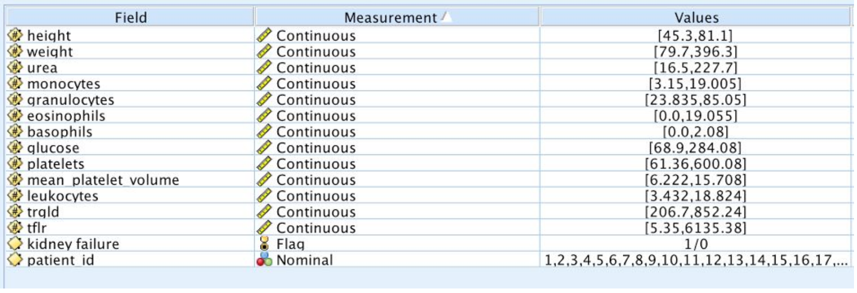
- **Amount of data:**
    1. **kiney_fail_dataset_v2.csv**:
        - 957 (row) patients with 20 attributes each one. One attribute is for the patient ID (key = drugs.csv) , another is the boolean target indicating if the patient suffered from kidney failure and the others are characteristics of the patient.
    2. **drugs.csv**:
        - 957 (row patients) with 10 columns where one is the patient ID and the others have the name of the drugs taken by the patient during the treatment. It is worth to remark that each column does not indicate the same thing for each patient that is why this dataset will need a further cleaning to have it in a profitable form.



patient_id	height	weight	urea	monocytes	granulocytes	eosinophils	basophils	glucose	platelets	mean_platelet_volume	leukocytes	trgld	tflr	kidney failure
1	69.900	168....	36....	7.035	43.365	6.283	0.728	107.060	261.040	7.650	6.760	321.1...	112.3...	0
2	65.200	162....	50....	6.930	68.565	4.841	0.624	188.680	331.760	8.364	8.632	566.0...	$null$	1
3	70.600	138....	$n...	$null$	$null$	$null$	$null$	$null$	$null$	$null$	$null$	$null$	$null$	1
4	54.400	241....	27....	11.235	57.015	4.738	0.832	119.780	600.080	8.772	7.696	359.3...	$null$	0
5	50.700	149....	50....	8.400	48.195	4.635	0.520	124.020	200.720	9.384	6.656	372.0...	$null$	0

- **Type of data:**
    1. **kiney_fail_dataset_v2.csv**: Mostly all the patient attributes are numeric except target kidney failure that is boolean. Seems to be that the patient id it will not provide any information but for the moment we will keep it as an input at least

until we can give some kind of more accurate proof of that.



Field	Measurement	Values
height	Continuous	[45.3,81.1]
weight	Continuous	[79.7,396.3]
urea	Continuous	[16.5,227.7]
monocytes	Continuous	[3.15,19.005]
granulocytes	Continuous	[23.835,85.05]
eosinophils	Continuous	[0.0,19.055]
basophils	Continuous	[0.0,2.08]
glucose	Continuous	[68.9,284.08]
platelets	Continuous	[61.36,600.08]
mean_platelet_volume	Continuous	[6.222,15.708]
leukocytes	Continuous	[3.432,18.824]
trgld	Continuous	[206.7,852.24]
tflr	Continuous	[5.35,6135.38]
kidney failure	Flag	1/0
patient_id	Nominal	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,...

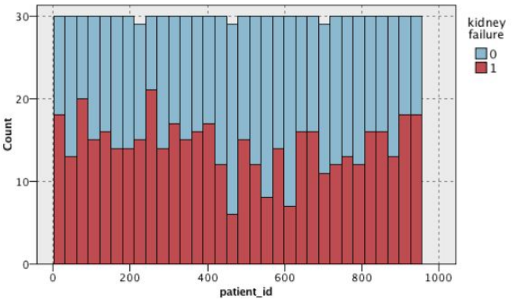
2. **drugs.csv**: Strings (categorical) describing drugs taken by each patient introduced by hospital staff. This data does not have very good quality due to typo errors same drugs have different names. This, plus that the format of the data is not the best, forces us to take a deep look into this dataset to clean it properly.

Exploring Data

Lets plot some graphs and run statistics on the available data to try to form some hypotheses about how the data can answer the technical and business goals.

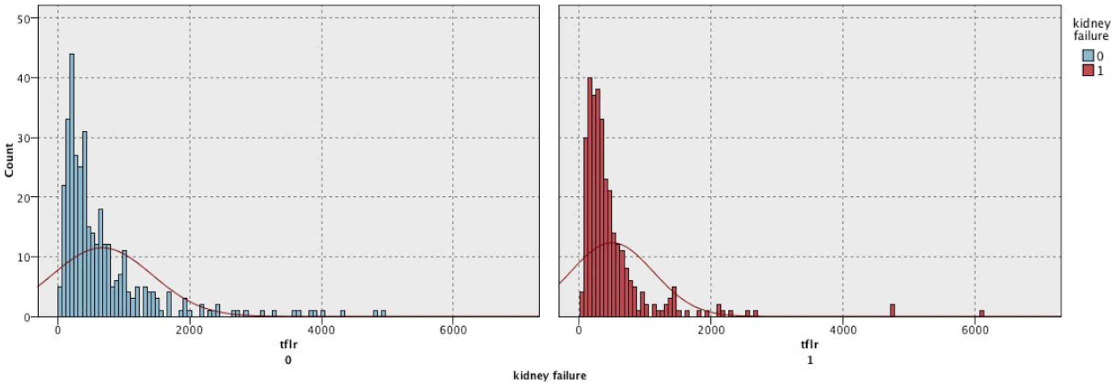
We perform this step using Graphs and Output tools of SPSS Modeler.

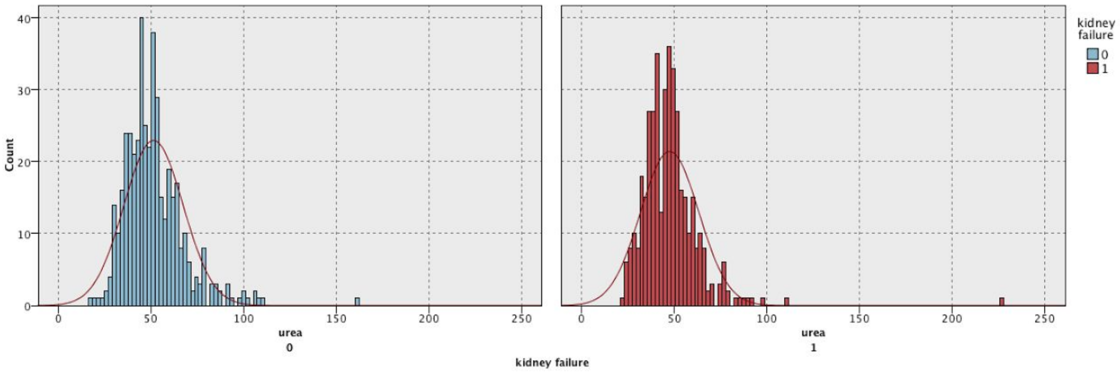
1. Plotting the distribution of the patient id we can consider that this number does not provide any valuable information so probably we will discard it as a predictor input.



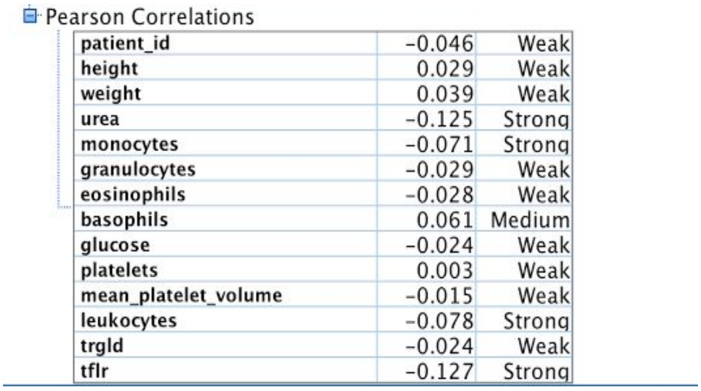
2. Seems to be that lower number of basophils (below the mode) could lead to be more likely to do not develop kidney failure.

3. The contrary with tflr attribute and urea





4. Taking a look at the correlations of the predictor variables vs the target we can see which variables seem to be more important for our prediction. We could use also the Feature selection model of SPSS Modeler.



Pearson Correlations

patient_id	−0.046	Weak
height	0.029	Weak
weight	0.039	Weak
urea	−0.125	Strong
monocytes	−0.071	Strong
granulocytes	−0.029	Weak
eosinophils	−0.028	Weak
basophils	0.061	Medium
glucose	−0.024	Weak
platelets	0.003	Weak
mean_platelet_volume	−0.015	Weak
leukocytes	−0.078	Strong
trgld	−0.024	Weak
tflr	−0.127	Strong

5. Taking a look to the correlation between variables we found that trgld and glucose are totally correlated. So we will discard trgld for example.

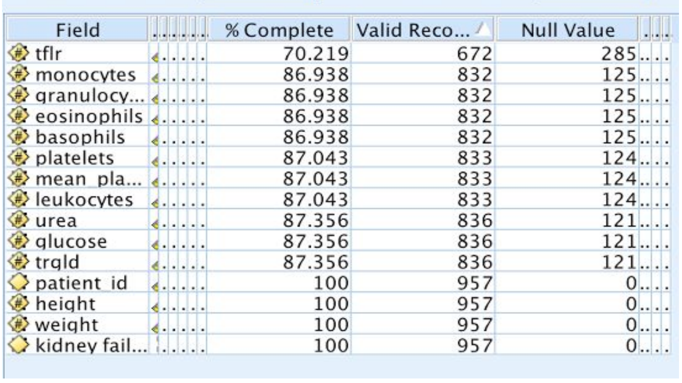
6. We are not experts of the field so a priori we do not know what represent any of the fields in the kiney_fail_dataset_v2.csv file but the “height” and the “weight”. So first thing we will do later will be to create the BMI variable:

$$BMI = \frac{weight(kg)}{height(m)^2}$$

which gives information about if the person is underweight or overweight. So for this reason we will have to transform the scales of the weight and the height to kg and m.

Verifying Data Quality:

- **Missing data:** We have some fields with missing values. It seems that the information was gathered in different way since we can detect 5 different groups according the way the data was collected regarding the null fields (285, 125, 124, 121, 0).



Field	% Complete	Valid Reco...	Null Value
tflr	70.219	672	285
monocytes	86.938	832	125
granulocy...	86.938	832	125
eosinophils	86.938	832	125
basophils	86.938	832	125
platelets	87.043	833	124
mean_pla...	87.043	833	124
leukocytes	87.043	833	124
urea	87.356	836	121
glucose	87.356	836	121
trgld	87.356	836	121
patient_id	100	957	0
height	100	957	0
weight	100	957	0
kidney fail...	100	957	0

The only information we have from all patients is “height” and “weight” and the target field.

- **Data Error:** On the “drugs.csv” file as we said before there are a lot of typo errors and also inconsistencies with the data.

Data Preparation:

Selecting Data

- **Selecting items (rows):**At first we consider all patients of equal importance so we will keep all the rows.

- **Selecting attributes or characteristics (columns):**First of all we will rid off those variables that clearly do not give us any information such as the patient id.

  We can advise that some variables are more important than others in different ways, for example as we saw when we explore the correlation between the variables or with SPSS Modeler we can use the “Feature Selection Option”which ranks the importance of the variables in different manners depending its origin. Even though some variables are more important than others for the moment we will keep all of them.

Cleaning Data

- **Missing data:** We decided to remove those patients of which we just have information of height and weight, since logically thinking, we do not reckon that is possible to give any insight about kidney failure with just these two measures.

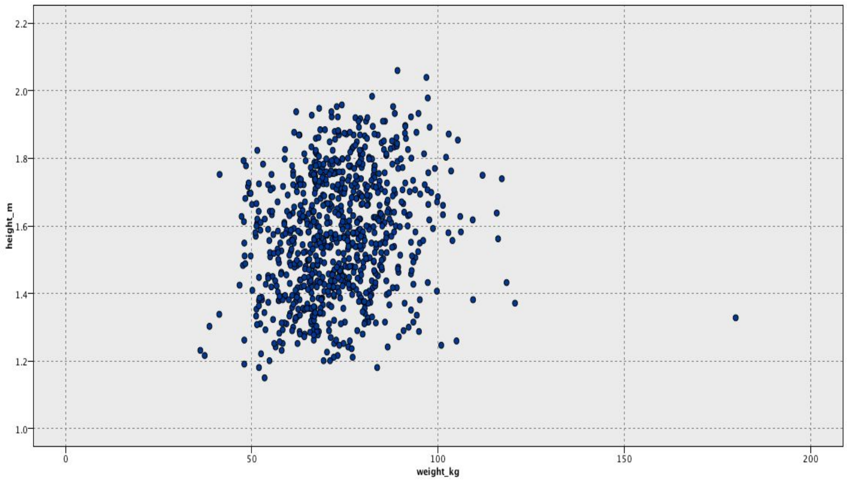
  After discarding those patients we have 836 patients remaining. Some of these patients, 164 concretely, do not have the feature “tflr” which we consider an important feature to predict kidney failure due its correlation with the target.

  For this reason, we consider that a good practice that can increase the performance of our model could be to fill up the null “tlfr” values for some likely realistic values. We have considered different approaches for replacing this null values but taking the mean is the one that worked the best for us.

  A part from “tlfr” we still have some other features with around 30 values missed so we can also replace these missing values for the mean.

- **Data errors and measurement errors.** Here it’s kind of difficult to spot errors since we do not have knowledge of the field and the majority of the fields we do not know what they mean.

  But this is not the case for the height and weight, creating a scatterplot of both variables we can spot if there is any outlier.



As we can see above there is one clear outlier, because it is almost impossible that someone less than 1,4 m had a weight of more than 179 kg.

To spot more outliers we had to take a look at the distribution of the data separately and look for values that were far from the rest, and also, if a priori we know any relationship between some of the attributes we could spot multidimensional outliers like the one we found before for the weight and height.

As we said before, as we do not know the field and the attributes in question, at the beginning we are not going to consider any more value as outlier, maybe at the end we will do it in case it increases the model’s accuracy.

In SPSS Modeler there are functions that can find extreme values and outliers automatically, we will try them at the modeling part to see if they improve accuracies.

Regarding the drugs.csv file we had to make a huge cleaning for 2 main reasons. The first is due the inconsistency with names of same drugs and second because the structure is not the best to apply ML algorithms. We would like to have the data onehot encoded. All the steps regarding this clesaning can be found on the Drugs_DataCleaning.r script delivered.

Constructing New Data

As we said previously, our major knowledge of the variables are the weight and height, so we can create a new categorical variable that takes into account the relationship between those two. Also first we create the variable weight in kg and the height in m.

We create the variable BMI which is highly known. The calculation is the following:

$$BMI = \frac{weight(kg)}{height(m)^2}$$

Regarding the findings we talked in the Exploration Data we will create three new categorical variables for urea, basophils and tflr.

Regarding the findings we talked in the Exploration Data we will create three new categorical variables for urea, basophils and tflr and tflr*urea.

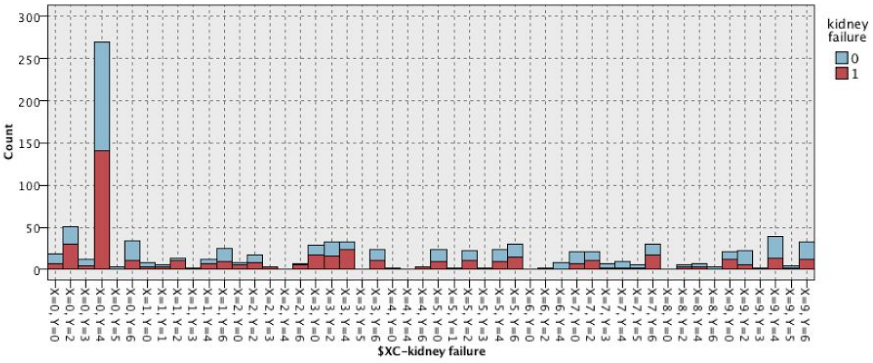
We also used the AutoDataPrep module of SPSS Modeler in order to transform and create new valuable variables. Basically what this model have done with our data is to scale the continuous variables to a mean=0 and sd=1 as well as some box-cox transformations to reduce skewness.

Integrating Data

After the cleaning of the drugs.csv file we realize that we have there are to many drugs to introduce them to our mobile without a preprocessment. For this we have tried to find some insights from the drugs that could be useful in order to improve our model and also to give some information regarding how each drug can affect to develop kidney complications.

For this reason we have tried different things. We started applying PCA to reduce the number of drugs to less variables representing a linear combination of drugs each one and keeping more or less the same information.

Also we have tried to perform clustering expecting groups of drugs that lead to curative outcomes. In the graph above we can see that some. Bad point it is that we do not know exactly what represents each cluster. In the graph below we can see that.



Also we tried to correlate each drug with the target variable and keep just the ones more correlated.

At the end we ended up choosing the PCA option. Is the one that makes more sense and also the one that improves more the accuracy even though just a little bit.

Modeling

Generating a Test Design

Our business goal was to predict the kidney failure target so we focused on improving the accuracy. Sometimes, usually in health problems, false positives or true negatives are more important than the others but in our case we just want to improve the accuracy since no more information was provided.

To find which are the most important features we will focus on these that are most important in our models.

To generate our models we will split our data into a Training and Testing set. 80% of the data will go for the Training and 20% for the Testing. We do that because we do not want to experiment overfitting of our model. Overfitting happens when our model memorize too much our data and does not generalize well for new data. Also we perform a k-fold cross validation model in SPSS Modeler to assess more accurately the performances of the different models.

Building Models:

Our final approach here after proving almost all the classificators separately was the following.

First we partitioned our dataset (80/20) and we modeled our final dataset (after all cleaning part) with the autoclassifier module of SPSS Modeler selecting the option to focus on improving the accuracy. After applying it several time we keep the classifiers that experienced more accuracy during all the runnings and the ones that were more time ranked as number one. The thing is that we are finding which is the best classifier for different subsets of 80% of the dataset since is not always the same the best one. Once we have the ones that perform better we will perform an ensemble of this models.

We finally select the following models to perform the ensemble:

1. Random Tree
2. XGBoost Tree
3. CHAID
4. C&R Tree
5. Neural Networks

In the Manual of SPSS there are the requirements of each model regarding the type of the predictors and targets that can apply.

Assessing the Model:

Comprehensive Model Assessment

- Interpret the models according to domain knowledge, DM success criteria and desired test design.
- Judge the success of the application of modelling and discovery techniques technically.
- Rank the models and assess them according to the evaluation criteria (taking the business objectives and business success criteria into account as far as we can here.)

(In most data mining projects a single technique is applied more than once and data mining results are generated with several different techniques.)

- **Model assessment** - Summarise the results of this task, list the qualities of your generated models (e.g.in terms of accuracy) and rank their quality in relation to each other.

Running our models through our partitions and the ensemble, we obtained an overall accuracy of 86.65% on the training partition and an overall accuracy of 54.55% on the testing partition. Despite the performances of our models on the training partition, there seems to be a significant loss of model accuracy on the testing partition.

In order to better understand the distribution of the overall accuracy on each model we ranked each model based on their respective training and testing accuracy as follows:

1. XGBboost Tree (95.85% and 54.55%)
2. Random Tree (87.39% and 53.41%)
3. CHAID (71.36% and 52.87%)
4. C&R Tree (68.1% and 53.41%)
5. Neural Networks (62.61% and 53.98%)

As we can see three out of five models surpassed a 75% accuracy in the training partition, however the maximum accuracy attained in the testing partition (54.55%) remains below our success criteria.

- **Revised parameter settings:**
  - **Models:** We have changed a bit the default settings because after a testing of the different settings we did not appreciate a big improvement of the accuracy.
  - **PCA:** We keep 10 components instead of 5 default components.
  - **Ensemble:** it is set to “Highest confidence wins”, using that setting we are sure we got the best confidence in predictions

Evaluation:

Evaluate our results

Previous evaluation steps dealt with factors such as the accuracy and generality of the model. During this step you'll assesses the degree to which the model meets your business objectives and seek to determine if there is some business reason why this model is deficient. Another option is to test the model(s) on test applications in the real application, if time and budget constraints permit. The evaluation phase also involves assessing any other data mining results you've generated. Data mining results involve models that are necessarily related to the original business objectives and all other findings that are not necessarily related to the original business objectives, but might also unveil additional challenges, information, or hints for future directions.

- **Assessment of data mining results - Summarise assessment results in terms of business success criteria, including a final statement regarding whether the project already meets the initial business objectives.**

- **Approved models - After assessing models with respect to business success criteria, the generated models that meet the selected criteria become the approved models.**

The medical nature of the application domain requires from our models a high accuracy. However,from what we can see from our previous model results, none of the models can be approved under our defined success criteria.

Given that our data is balanced, we expected a success rate way higher than 50% which is the one we can achieve predicting all 1 or 0. Even though all of our models are above this threshold we do not consider that we have succeed. Mainly we think that this happens because we do not have enough data and that is the reason why accuracies in training and testing are so different for all the models trained.

Review process

At this point, the resulting models appear to be satisfactory and to satisfy business needs. It is now appropriate for you to do a more thorough review of the data mining engagement in order to determine if there is any important factor or task that has somehow been overlooked. This review also covers quality assurance issues—for example: did we correctly build the model? Did we use only the attributes that we are allowed to use and that are available for future analyses?

- **Review of process - Summarise the process review and highlight activities that have been missed and those that should be repeated.**

In order to satisfy the business objectives of our project a great effort has been made to improve the quality of our models but here are a few aspects that could have been handled better in the model construction process of our project:

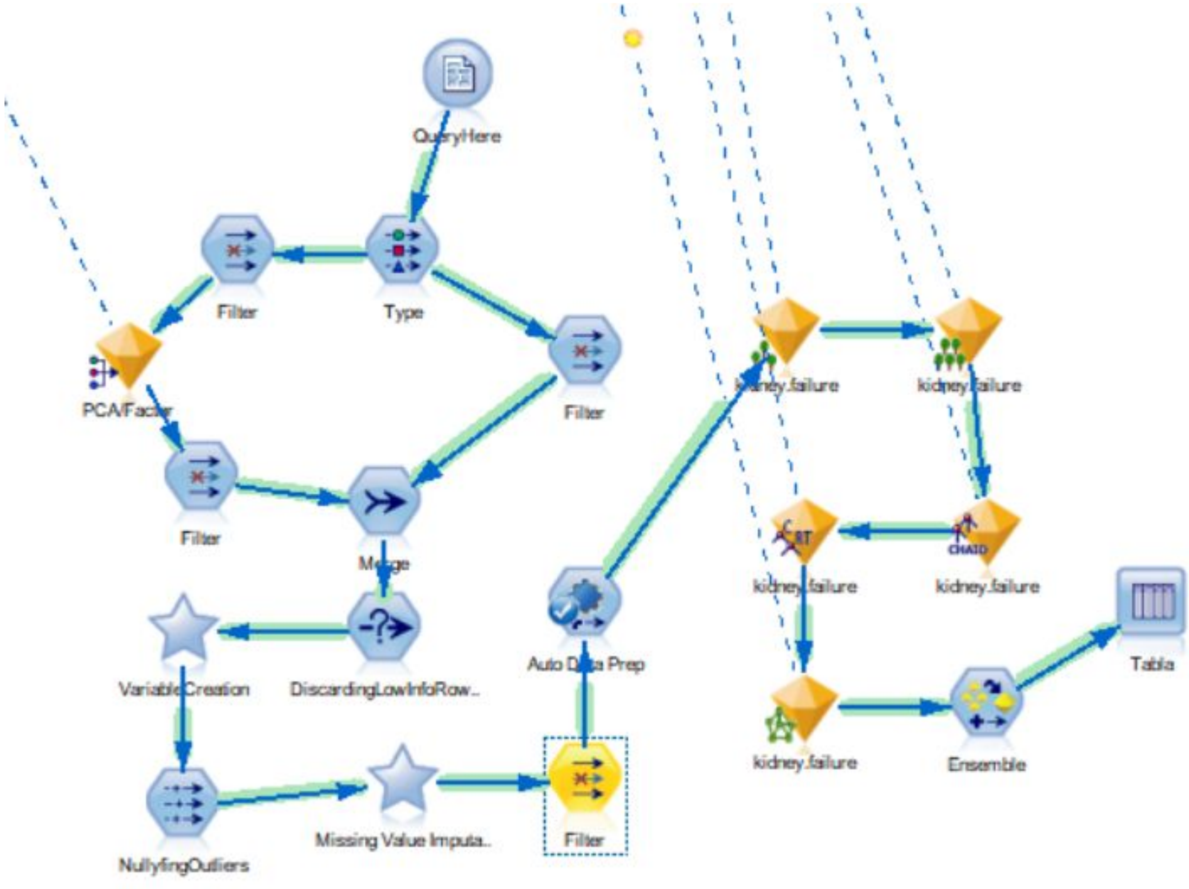
- In the cleaning stage of our data preparation different techniques could have been able to raise the quality of our data. This also applies to the transformation of our attributes which leads to potential loss of information.

- The models with the highest accuracy in the training partitions may show symptoms of model overfitting. A better control of the cross-validation could have helped us to obtain better results in the test partition.

Deployment:

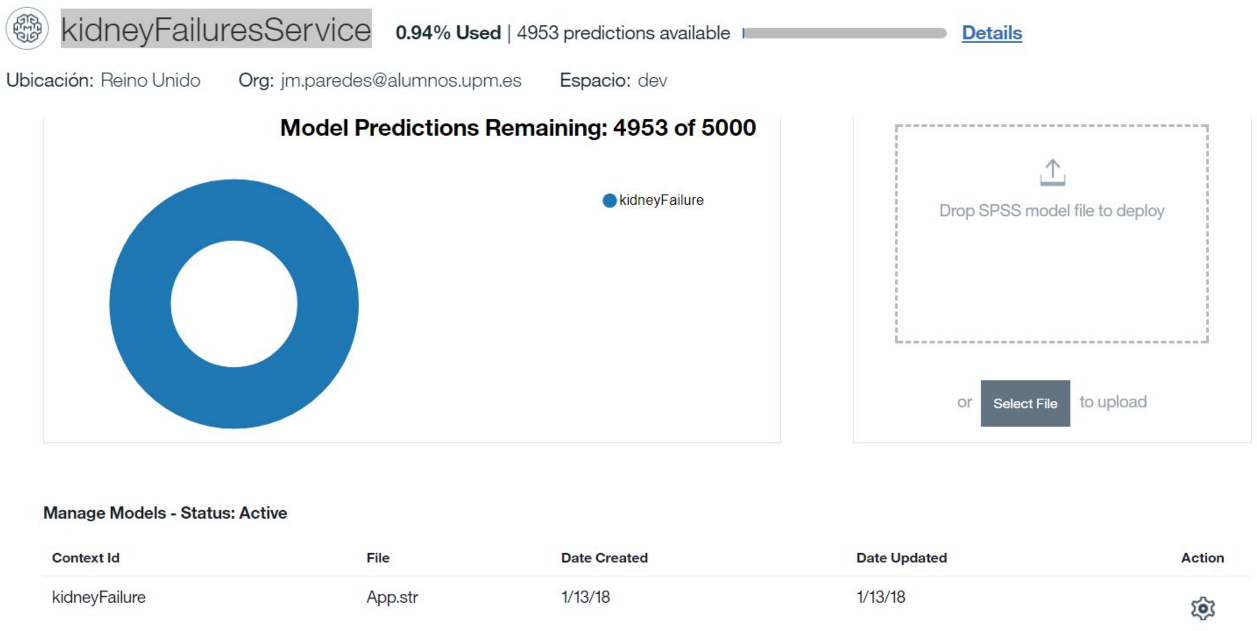
How to deploy our model:

Onces there is a final model, it is time to deploy it on Bluemix. The first step is to create the score branch:



After that we are ready to deploy our model and create the service.

Following the documentation provided by the teacher we have managed to create a service named: kidneyFailuresService where we deployed our App.str, we used as context ID: KidneyFailure.



kidneyFailuresService 0.94% Used | 4953 predictions available Details

Ubicación: Reino Unido Org: jm.paredes@alumnos.upm.es Espacio: dev

Model Predictions Remaining: 4953 of 5000

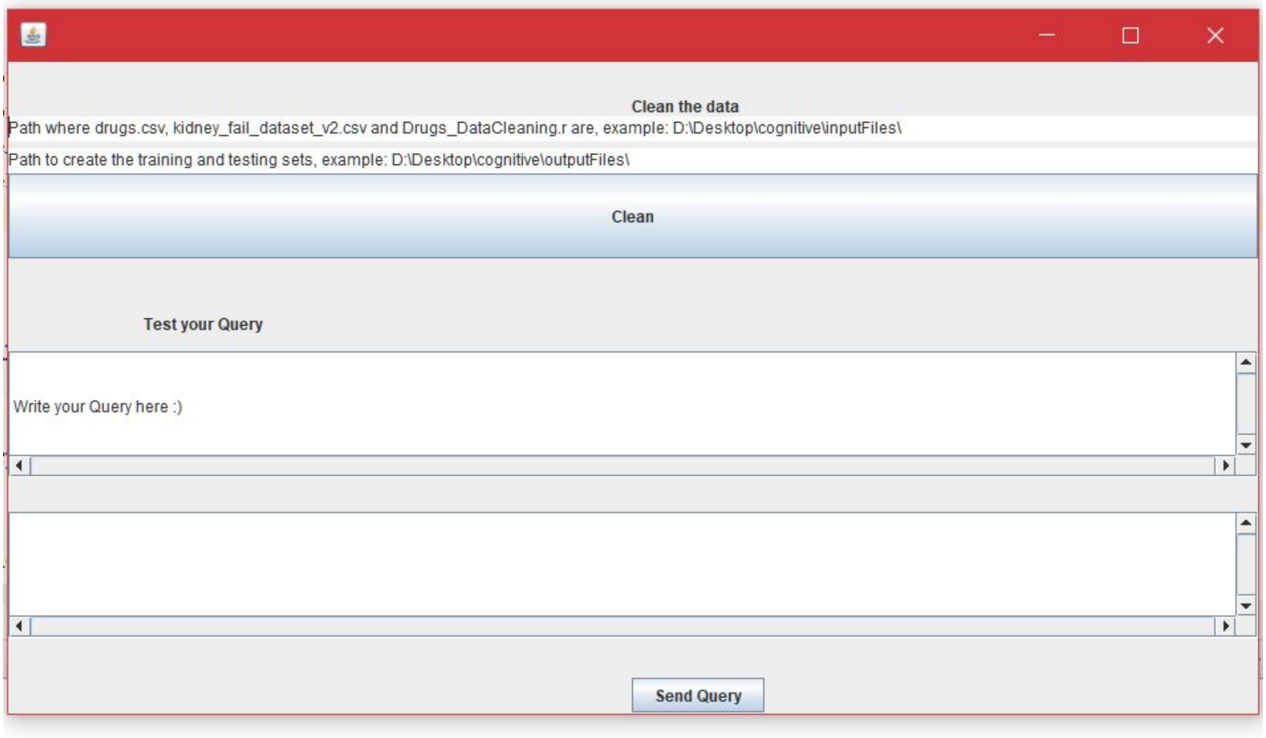
Manage Models - Status: Active

Context Id	File	Date Created	Date Updated	Action
kidneyFailure	App.str	1/13/18	1/13/18	⚙

Finally we made this link using the service credentials:

https://ibm-watson-ml.eu-gb.bluemix.net/pm/v1/score/kidneyFailure?accesskey=kFYv/HXMBb41KTuLSxf/UMluS/ddOPgsQISHVlv7d5aa6+NjYY/ay1aub5VV7A7VpvelDBj2EWArRQzCnErs5G6xF7OPG2R5H0oB0w5syog=

We have created an interface to send queries to bluemix easier. We have also included a function to run our R code in the interface.



Clean the data

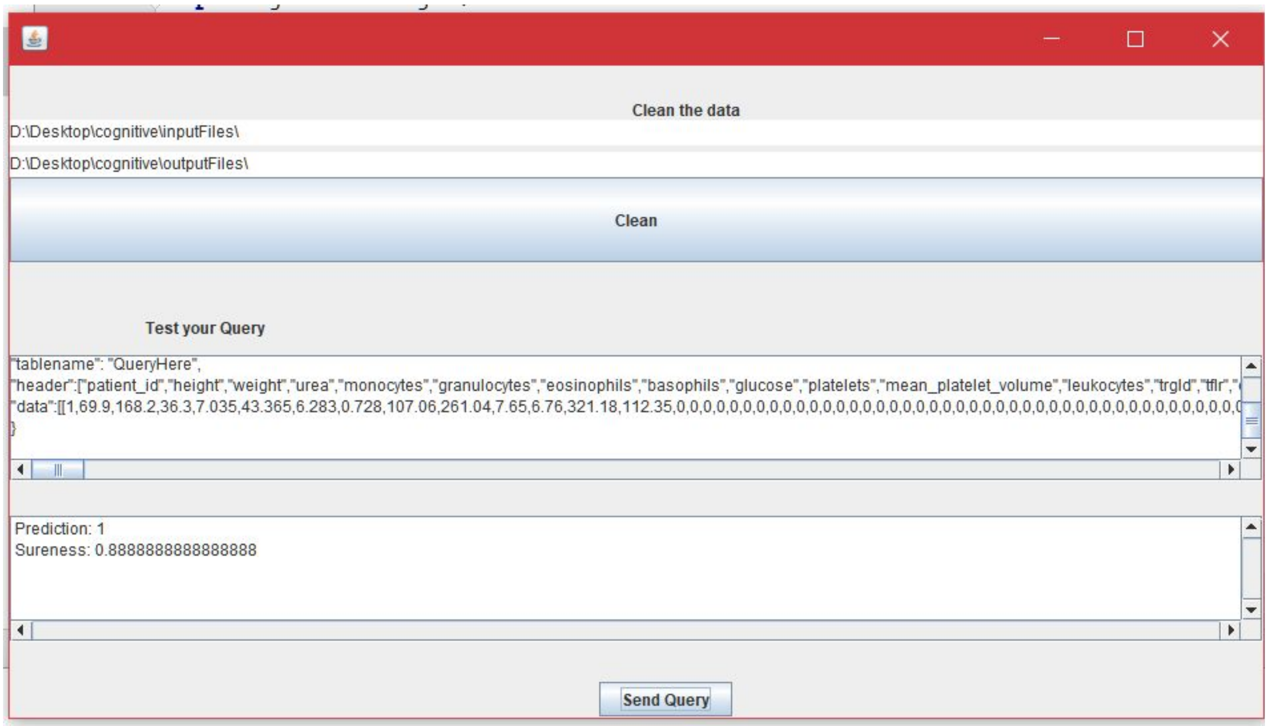
Path where drugs.csv, kidney_fail_dataset_v2.csv and Drugs_DataCleaning.r are, example: D:\Desktop\cognitive\inputFiles\

Path to create the training and testing sets, example: D:\Desktop\cognitive\outputFiles\

Test your Query

Write your Query here :)

Reply without errors:



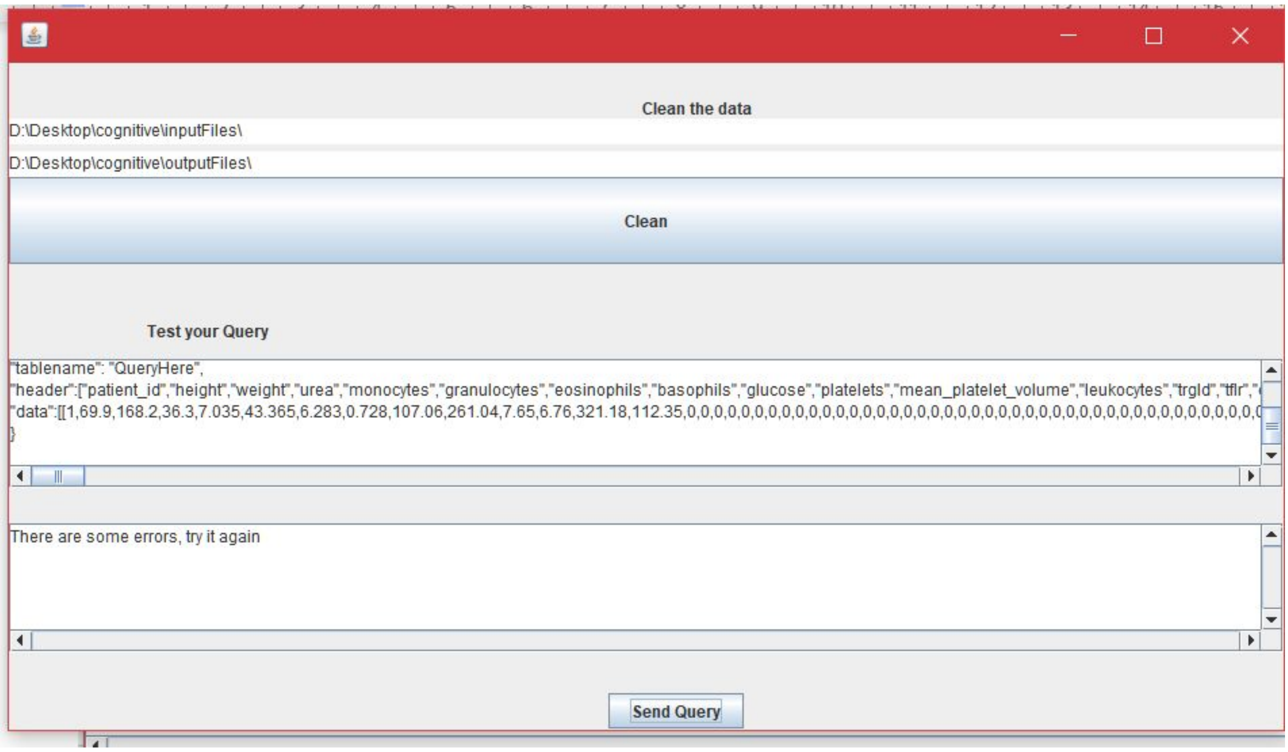
D:\Desktop\cognitive\inputFiles\

D:\Desktop\cognitive\outputFiles\

```
"tablename": "QueryHere",
"header":["patient_id","height","weight","urea","monocytes","granulocytes","eosinophils","basophils","glucose","platelets","mean_platelet_volume","leukocytes","trgld","tflr","
"data":[[1,69.9,168.2,36.3,7.035,43.365,6.283,0.728,107.06,261.04,7.65,6.76,321.18,112.35,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
}
```

```
Prediction: 1
Sureness: 0.8888888888888888
```

Reply with errors:



D:\Desktop\cognitive\inputFiles\

D:\Desktop\cognitive\outputFiles\

```
"tablename": "QueryHere",
"header":["patient_id","height","weight","urea","monocytes","granulocytes","eosinophils","basophils","glucose","platelets","mean_platelet_volume","leukocytes","trgld","tflr","
"data":[[1,69.9,168.2,36.3,7.035,43.365,6.283,0.728,107.06,261.04,7.65,6.76,321.18,112.35,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
}
```

```
There are some errors, try it again
```

Software used to create our model

SPSS Modeler is a data mining and text analytics software application from IBM. It is used to build predictive models and conduct other analytic tasks. It has a visual interface which allows users to leverage statistical and data mining algorithms without programming. One of its main aims from the outset was to get rid of unnecessary complexity in data transformations, and to make complex predictive models very easy to use.

Software used in the different steps of methodology CRISP-DM

We have used a variety tools in the process:

- SPSS: Used to create the final model and also to do part of the data preparation.
- R: We developed a R script to clean our data.
- Java: There are two java classes, their purpose is to create the interface showed previously and send the query to the model on bluemix.

Review project

Assess what went right and what went wrong, what was done well and what needs to be improved.

- **Experience documentation** - Summarise important experience gained during the project. For example, any pitfalls you encountered, misleading approaches, or hints for selecting the best suited data mining techniques in similar situations could be part of this documentation. In ideal projects, experience documentation also covers any reports that have been written by individual project members during previous phases of the project.

Overall, our project has been the successful in:

- Selecting the most suited models for our tasks.
- Identifying the most important variables in a dataset.
- Creating new variables through PCA and existing attributes to reduce the number of available variables.

The potential problems in our project were:

- Possible overfitting in our models.
- Insufficient amount of data to create high accuracy models.
- Our understanding of the SPSS tool.
- Lack of medicine background, we could have done better transformations of the variables.

By focusing on a small number of the most important attributes we gained better insights on the factors of kidney failure but the created models couldn't reach the target accuracy while ensuring the restrictions on the data of the second success criteria were met.