Université Gustave Eiffel
Arous Souhir – M2 SIA

Heart Failure Prediction Report

Link DataSet : https://www.kaggle.com/andrewmvd/heart-failure-clinical-data

Steps :

1) Download the DataSet and prepare the working environment with all required libraries

`pandas` `numpy` `matplotlib` `seaborn` `sklearn`

2) Data understanding

The dataset used in this study is based on historical data. It is a public dataset in a csv document with size = 12.24 kB.
Familiarization with the Dataset :

`.shape` `.columns` `.dtypes` `df.describe()` `df.isnull().sum()`

It contains 299 rows with 13 attributes.

Age	Patient's age	float
Anaemia	Decrease of red blood cells or hemoglobin	Int [0,1]
Creatinine phosphokinase	Level of the CPK enzyme in the blood (mcg/L)	Int
Diabetes	If the patient has diabetes	Int [0,1]
Ejection fraction	Percentage of blood leaving the heart at each contraction (percentage)	Int
High blood pressure	If the patient has hypertension	Int [0,1]
Patelets	Platelets in the blood (kiloplatelets/mL	Float
Serum creatinine	Level of serum creatinine in the blood (mg/dL)	Float
Serum sodium	Level of serum sodium in the blood (mEq/L)	Int
Sex	Woman or man	Int [0,1]
Smoking	If the patient smokes	Int [0,1]
Time	Recording time of the informations	Int
Death event	If the patient is dead (Our target)	Int [0,1]

3) Verify the Data quality and prepare the data to be able to work with

For a better and more optimal result we have to make strategic decisions and select the most useful variables for the development of a model. Thus, there a column we decided to delete wich is the column 'time' because it has no meaningful impact on our classification model.
The dataset does not have any missing or erroneous data values.It is ready to use.

4) Data visualization and correlation analysis

Data visualization provides an important suite of tools for identifying a qualitative understanding. This can be helpful to explore the dataset and extract some informations. In our case we noticed :
67.89% of registered patients are alive
The attributes 'smoking', 'sex', 'high_blood_pressure', 'diabetes', 'creatinine_phosphokinase', 'ejection fraction' and 'anaemia' don't have an important impact on the death probability
Death probability based on the age : from 80 we can see that the death probability increases
Death probability based on the ejection fraction : we cannot conclude at this stage
Death probability based on the platelets : between 100000 and 400000 the death probability increases
Death probability based on the serum creatinine : between 0.5 and 3 the death probability increases
Death probability based on the serum sodium : between 127 and 143 the death probability increases
Death probability based on the serum sodium : between 127 and 143 the death probability increases

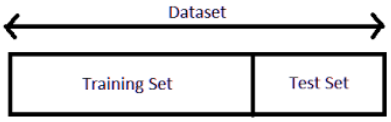
However, the visualization allows us to have an idea about the dataset but we can not conclude at this stage.

5) Encoding and dividing

This crucial step concerns transforming the raw data that was collected into a form that can be used in predictive modeling. The first part is encoding our data, machine learning models often take in numerical data as input so our goal is to transform our categorical data to numerical ones.
One approach is pandas' built in function get_dummies() which takes in raw categorical data and turns it into dummy/indicator variables.

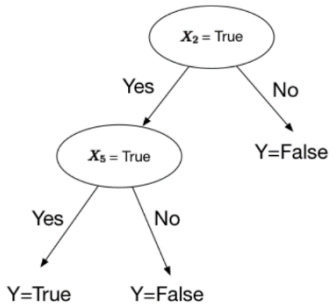
```
df=pd.get_dummies(df,drop_first=True)
```

Besides, we should never evaluate the performance of our model on the same data that will be used to train it. So when we do machine learning we always divide our dataset in two parts. In general we put 80% of the data in the training part and 20% in the test part.



6) Decision Tree Model

Decision trees are among the most popular machine learning algorithms given their intelligibility and simplicity.
It's used to visually and explicitly represent decisions and decision making. As the name goes, it uses a tree-like model of decisions.
For predicting a class label for a record we start from the root of the tree. We compare the values of the root attribute with the record's attribute. On the basis of comparison, we follow the branch corresponding to that value and jump to the next node.



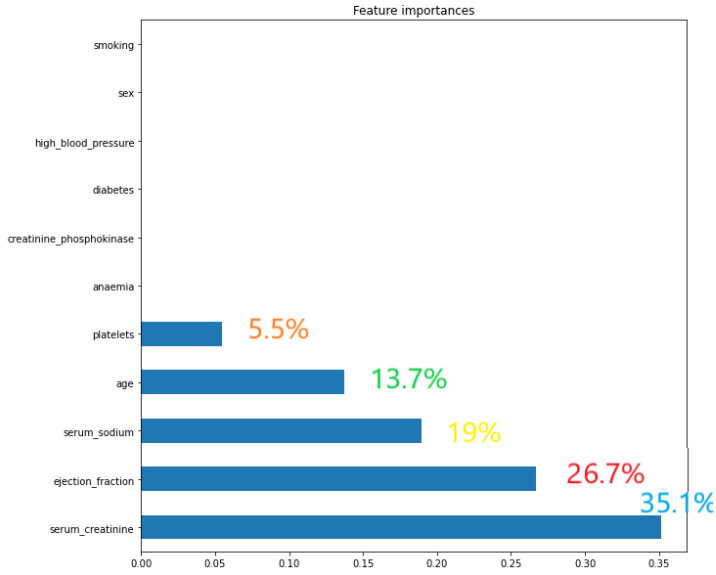
First thing first, we created a dictionary called 'param_grid' in order to configure the hyperparameters:
criterion: The function to measure the quality of a split. Supported criteria are "gini" for the Gini impurity and "entropy" for the information gain.
max_depth: The maximum depth of the tree.
max_leaf_nodes: Max number of leaf nodes - Best nodes are defined as relative reduction in impurity.
Then we created a grid search instance that will test all the combinations of the values of the hyperparameters and give us the best combination which turned out to be :

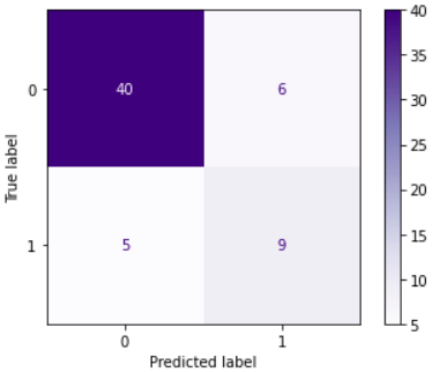
```
{'criterion': 'entropy', 'max_depth': 4, 'max_leaf_nodes': None}
```

We created an instance of the decision tree classification algorithm called DT, we trained it and got the scores below.

Model	Accuracy	Precision	Recall	F1 Score	F2 Score
Decision Tree	0.816667	0.6	0.642857	0.62069	0.633803

```
Train score : 0.8368200836820083
Test score : 0.8166666666666667
```





The impact of the attributes 'smoking', 'sex', 'high_blood_pressure', 'diabetes', 'creatinine_phosphokinase', 'anaemia' is not strong enough to take them into consideration. Thanks to GridSearchCV we know that doing the analysis on the attributes 'platelets', 'age', 'serum_sodium', 'ejection_fraction' and 'serum_creatinine' allows us to have the best accuracy possible.

The test score is very high we can say we have a good prediction model.