Data Modelling and Presentation: Heart Failure Clinical Records Dataset

Classification: k-nearest neighbours and decision tree algorithm

Date of Report: 21/05/2022

Abstract

Predicting the death event using the heart failure clinical records dataset which entails the medical records of 299 patients who had heart failure, collected during their follow up period, where each patient profile has 13 clinical features. Classification algorithms such as k-nearest neighbors and decision tree were implemented to achieve the predictions.

Affiliations

RMIT University

I certify that this is all my own original work. If I took any parts from elsewhere, then they were non-essential parts of the assignment, and they are clearly attributed in my submission. I will show I agree to this honor code by typing "Yes": *Yes*.

Pranav Karnth Mannur | s3828461
S3828461@student.rmit.edu.au

Table of Contents

1. Executive Summary:

1.1 Goal of the Project:

The dataset used in this project is the Heart Failure Clinical Records Dataset, it entails medical records of 299 patients who had a heart failure, collected during their follow-up period, where each patient profile has 13 clinical features.

The goal of our project is to use the features in the dataset and predict the death event using Classification and also find out which factors influence the chances of leading to a death_event or heart attack. Classification is the technique of predicting the class of given data points. Targets, labels, and categories are all terms used to describe classes. The task of approximating a mapping function (f) from discrete input variables (X) to classification predictive modelling (y). *In our case target: death_event, train: rest of the features/columns.*

The classification algorithms we are using in this project to predict the death event of the patients are: k-nearest neighbours and decision tree algorithm.

2. Introduction:

As the goal of the project suggests we aim to predict the death event of the patients using the heart failure clinical records dataset by implementing classification algorithms such as k-nearest neighbor and decision tree algorithm.

In order to implement these algorithms on the dataset, we first need to perform data cleaning and data exploration.

2.1 Data Cleaning:

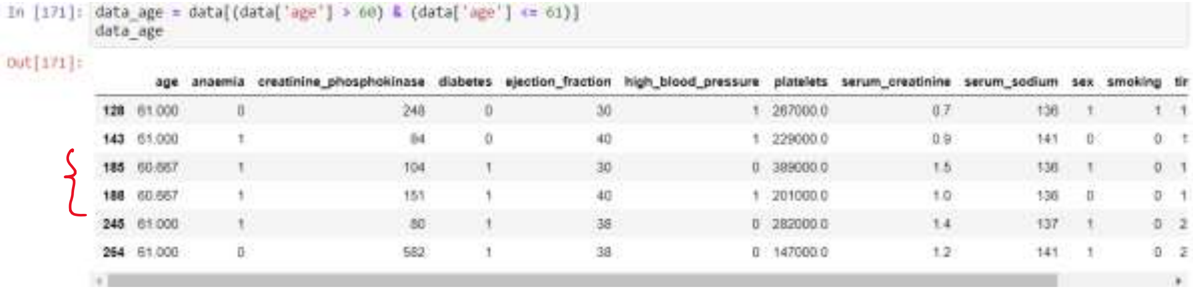
Step 1:

- Checking for duplicate values if any in the Dataframe using duplicated and dropping them using drop_duplicates().
- Checking for NaNs values if any in the Dataframe using isna().sum

This is a Heart Failure Clinical Records Dataset and we do not require Duplicated records as it would result in our distribution and give us wrong statistics. The NaNs values too are not required as they do not provide us clear data.

Step 2:

- Checking the data types of each of the columns in the dataset.
- We find that the data type of the age column is in Float
- The age has been entered in the for of decimals in 2 records as seen below.



```
In [171]: data_age = data[(data['age'] > 60) & (data['age'] <= 61)]
          data_age
```

Out[171]:

	age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressure	platelets	serum_creatinine	serum_sodium	sex	smoking	tir
128	61.000	0	248	0	30	1	267000.0	0.7	136	1	1	1
143	61.000	1	84	0	40	1	229000.0	0.9	141	0	0	1
185	60.667	1	104	1	30	0	389000.0	1.5	136	1	0	1
188	60.667	1	151	1	40	1	201000.0	1.0	136	0	0	1
245	61.000	1	80	1	38	0	282000.0	1.4	137	1	0	2
264	61.000	0	582	1	38	0	147000.0	1.2	141	1	0	2

- Rounding off the decimal values using round() function on the age columns and convert the data type to int using astype(int).

We round off and convert the datatype of the age column from float to int because we do not require age values in the form of decimals and it would provide us incorrect statistics.

2.2 Data Exploration:

Step 1:

- Using describe() function on the dataset in order to view basic statistical details of each column in the Dataframe like: count, mean, std .etc.

As per our goal, we need to make predictions for the death event, so we can explore the correlation of each column in the dataset with the death event column.

Step 2:

- Plotting a heatmap using seaborn to explore the correlation between each of the column's values.
    1. If the correlation value is positive and closer to 1, it means that as one column increases so does the other.
    2. If the correlation value is negative and closer to -1, it means that as one column increases, the other decreases.

Columns positively correlated with death event:

- age, anaemia, creatinine_phosphokinase, high_blood_pressure, serum_creatinine

Columns negatively correlated with death event:

- diabetes, ejection_fraction, platelets, serum_sodium, sex, smoking, time

Step 3:

- Plotting a pairplot using seaborn to explore the relationship between each of the column's values.

2.2.1 Exploring each column in the Dataset:

Age Column:

1. Exploring the age column in the dataset using the Boxplot
2. Exploring the age column will give us the Maximum, Minimum, Median, Quartiles of the age group of people in the dataset. By this we can get the statistical distribution of the column.

Anaemia Column:

1. Exploring the anaemia column in the dataset using the Bar plot
2. Exploring the anaemia column will give us the total number of cases where there is decrease of red blood cells or hemoglobin 0 = False, 1 = True. By this we can get the count of the values.

Creatinine phosphokinase Column:

1. Exploring the creatinine phosphokinase column in the dataset using the Boxplot

2. Exploring the creatinine phosphokinase column will give us the Maximum, Minimum, Median, Quartiles of the level of the CPK enzyme in the blood (mcg/L) in the dataset. By this we can get the statistical distribution of the column.

As shown in the Boxplot there are outliers present in the column before and after cleaning. These are not to be removed from the dataset as it is not acceptable to remove values just because it is an outlier. So there is no error in this perspective.

Diabetes Column:

1. Exploring the diabetes column in the dataset using the Bar plot
2. Exploring the diabetes column will give us the number of cases if the patient has diabetes (boolean) 0 = False, 1 = True. By this we can get the count of the values.

Ejection fraction Column:

1. Exploring the Ejection fraction column in the dataset using the Boxplot
2. Exploring the Ejection fraction column will give us the Maximum, Minimum, Median, Quartiles of the level of the percentage of blood leaving the heart at each contraction (percentage) in the dataset. By this we can get the statistical distribution of the column.

As shown in the Boxplot there are outliers present in the column before and after cleaning. These are not to be removed from the dataset as it is not acceptable to remove values just because it is an outlier. So there is no error in this perspective.

Serum creatinine Column:

1. Exploring the serum creatinine column in the dataset using the Boxplot
2. Exploring the serum creatinine column will give us the Maximum, Minimum, Median, Quartiles of the level of the level of serum creatinine in the blood (mg/dL) in the dataset. By this we can get the statistical distribution of the column.

As shown in the Boxplot there are outliers present in the column before and after cleaning. These are not to be removed from the dataset as it is not acceptable to remove values just because it is an outlier. So there is no error in this perspective.

Serum sodium Column:

1. Exploring the serum sodium column in the dataset using the Boxplot
2. Exploring the serum sodium column will give us the Maximum, Minimum, Median, Quartiles of the level exploring the serum creatinine column will give us the Maximum, Minumum, Median, Quartiles of the level of serum sodium in the blood (mEq/L) in the dataset. By this we can get the statistical distribution of the column.

As shown in the Boxplot there are outliers present in the column before and after cleaning. These are not to be removed from the dataset as it is not acceptable to remove values just because it is an outlier. So there is no error in this perspective.

Sex Column:

1. Exploring the sex column in the dataset using the Bar plot
2. Exploring the sex column will give us the number of male and female in the dataset 0 = Female, 1 = Male. By this we can get the count of the values.

Smoking Column:

1. Exploring the smoking column in the dataset using the Bar plot
2. Exploring the smoking column will give us the total number of patients in the dataset who smoke or not. 0 = No, 1 = Yes. By this we can get the count of the values.

Time Column:

1. Exploring the time column in the dataset using the Boxplot
2. Exploring the time column will give us the Maximum, Minimum, Median, Quartiles of the follow-up period (days) in the dataset. By this we can get the statistical distribution of the column.

Death event Column:

1. Exploring the DEATH_EVENT column in the dataset using the Bar plot
2. Exploring the DEATH_EVENT column exploring the death event column will give us the total number patient deceased during the follow-up period (boolean). 0 = No, 1 = Yes. By this we can get the count of the values.

2.2.2 Exploring the relationship between all pairs of attributes:

1. **Percentage of Male and Female patients deceased during the follow-up period**

A pie chart can be used on a categorical variable, each slice can be used to represent a specific category. A categorical variable represents types of data which are most likely to be divided into groups. Example: Sex (Male, Female) , Fuel type (Petrol, diesel), Age group. Therefore, plotting a pie chart to project the percentage of total number of Male and Female patients deceased during the follow-up period.

We can see that:

65% of the female patients were deceased during the follow-up period and 35% of the male patients were deceased during the follow-up period.

2. **Percentage of Male/ Female patients who smoke or dont smoke, and deceased during the follow-up period**

A pie chart can be used on a categorical variable, each slice can be used to represent a specific category. A categorical variable represents types of data which are most likely to be divided into groups. Example: Sex (Male, Female) , Fuel type (Petrol, diesel), Age group. Therefore, plotting a pie chart to project the percentage of total number of Male/ Female patients who smoke or not and deceased during the follow-up period.

We can see that:

69% of the male/ female patients who do not smoke were deceased during the follow-up period and 31% who smoke weren’t.

3. **Percentage of Male/ Female patients who have high blood pressure or not, and deceased during the follow-up period**

A pie chart can be used on a categorical variable, each slice can be used to represent a specific category. A categorical variable represents types of data which are most likely to be divided into groups. Example: Sex (Male, Female) , Fuel type (Petrol, diesel), Age group. Therefore, plotting a pie chart to project the percentage of Male/ Female patients who have high blood pressure or not, and deceased during the follow-up period .

We can see that:

59% of the male/ female patients who do not have high blood pressure were deceased during the follow-up period and 41% who have high blood pressure weren't.

4. **Percentage of patients who have high anaemia or not, and deceased during the follow-up period**

A pie chart can be used on a categorical variable, each slice can be used to represent a specific category. A categorical variable represents types of data which are most likely to be divided into groups. Example: Sex (Male, Female) , Fuel type (Petrol, diesel), Age group. Therefore, plotting a pie chart to project the percentage of Male/ Female patients who have diabetes or not, and deceased during the follow-up period.

We can see that:

52% of the male/ female patients who do not have diabetes were deceased during the follow-up period and 48% who have diabetes weren't.

5. **Percentage of patients who have diabetes or not, and deceased during the follow-up period**

A pie chart can be used on a categorical variable, each slice can be used to represent a specific category. A categorical variable represents types of data which are most likely to be divided into groups. Example: Sex (Male, Female) , Fuel type (Petrol, diesel), Age group. Therefore, plotting a pie chart to project the percentage of Male/ Female patients who have diabetes or not, and deceased during the follow-up period.

We can see that:

58% of the male/ female patients who do not have diabetes were deceased during the follow-up period and 42% who have diabetes weren't.

6. **Total count of patients across different age groups who were and were not deceased during the follow-up period**

A countplot is used to show the counts of observations in each categorical bin using bars. Therefore, using seaborn and plotting a countplot to project the total number of Male and Female patients across different age groups who have and haven't been deceased during the follow-up period.

We can see that:

- The maximum number of patients deceased during the follow-up period at the age of 60 is the highest.
- The maximum number of patients not deceased during the follow-up period at the age of 60 is the highest.

7. **Percentage of patients who have diabetes or not, and deceased during the follow-up period**

A Catplot shows frequencies or percents of the categories of one, two or three categorical variables. Therefore, using seaborn and plotting a Catplot to visualize the percentage of females and males in different age groups having heart disease or not.

We can see that:

- Over the age of 90 there are a very few patients (both male and female) who were deceased during the follow-up period.
- The percentage of male who were deceased during the follow-up period between the age 60 and 80 are high.
- The percentage of both male and female between the age of 50 and 70 who were not deceased during the follow-up period is high.
- The percentage of men who were deceased during the follow-up period are higher than that of women

8. **The ejection fraction in male and female who were and were not deceased during the follow-up period**

A bar graph is used when we have to plot a categorical variable. A categorical variable represents types of data which are most likely to be divided into groups. Example: Sex (Male, Female) , Fuel type (Petrol, diesel), Age group. Therefore, using seaborn and plotting a bar plot to project the ejection fraction of Male/ Female patients who have and haven't been deceased during the follow-up period.

We can see that:

- The ejection fraction is lower in both male and female who have been deceased during the follow-up period.
- Overall, the ejection fraction is higher in females than in males, irrespective of whether they have been deceased or not.

9. **The creatinine phosphokinase in male and female who were and were not deceased during the follow-up period**

A bar graph is used when we have to plot a categorical variable. A categorical variable represents types of data which are most likely to be divided into groups. Example: Sex (Male, Female) , Fuel type (Petrol, diesel), Age group. Therefore, using seaborn and plotting a bar plot to project the creatinine phosphokinase of Male/ Female patients who have and haven't been deceased during the follow-up period.

We can see that:

- The creatinine phosphokinase is higher in both male and female who have been deceased during the follow-up period.
- Overall, the creatinine phosphokinase is higher in males than in females, irrespective of whether they have been deceased or not.

10. **The platelets in male and female who were and were not deceased during the follow-up period**

A bar graph is used when we have to plot a categorical variable. A categorical variable represents types of data which are most likely to be divided into groups. Example: Sex (Male, Female) , Fuel type (Petrol, diesel), Age group. Therefore, using seaborn and plotting a bar plot to project the platelets of Male/ Female patients who have and haven't been deceased during the follow-up period.

We can see that:

- The platelet count have been more or less the same in males irrespective of whether they were deceased during the follow-up period or not.
- The platelet count is higher in females when they were not deceased during the follow-up period.

11. The serum creatinine in male and female who were and were not deceased during the follow-up period

A bar graph is used when we have to plot a categorical variable. A categorical variable represents types of data which are most likely to be divided into groups. Example: Sex (Male, Female) , Fuel type (Petrol, diesel), Age group. Therefore, using seaborn and plotting a bar plot to project the serum creatinine of Male/ Female patients who have and haven't been deceased during the follow-up period.

We can see that:

- The serum creatinine count has been high in both male and female when they were deceased during the follow-up period.
- The serum creatinine count has been low in both male and female when they were not deceased during the follow-up period.
- The serum creatinine count has been high in female than male during the time when they were deceased during the follow-up period.

12. The count of male and female who were and were not deceased during the follow-up period

A countplot is used to show the counts of observations in each categorical bin using bars. Therefore, using seaborn and plotting a countplot to project the total number of Male and Female patients who have and haven't been deceased during the follow-up period.

We can see that:

- The total number of male who have been deceased during the follow-up period is higher than that of the females.
- The total number of male who have not been deceased during the follow-up period is higher than that of the females.

13. The time (follow-up period - days) in male and female who were and were not deceased

A bar graph is used when we have to plot a categorical variable. A categorical variable represents types of data which are most likely to be divided into groups. Example: Sex (Male, Female) , Fuel type (Petrol, diesel), Age group. Therefore, time (follow-up period - days) in male and female who were and were not deceased.

We can see that:

- The follow-up period in female is higher than male when the patient has been deceased.
- The follow-up period in female is higher than male when the patient has not been deceased.

3. Methodology:

Using classification algorithms: k-nearest neighbor and decision tree.

Train test split: As per our dataset, the target label is the death_event column and the rest of the columns are used for training the machine learning model. The death event gives us the data whether the patient was deceased during the follow-up period or not.

Going by the good train test split, allocating 80% of the data for training and the rest 20% for testing and choosing the value of random state = 0 which controls the shuffling applied to the data before applying the split

The confusion matrix is a matrix that is used to evaluate the classification models' performance for a given set of test data. Only if the true values for test data are known can it be determined. This is computed for evaluation in each scenario of our algorithms.

A classification report is a machine learning performance evaluation statistic. It's used to demonstrate your trained classification model's precision, recall, F1 Score, and support. This is computed for evaluation in each scenario of our algorithms.

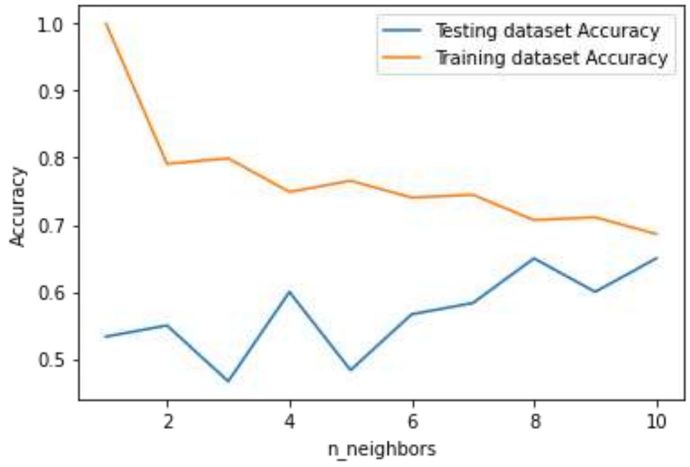
a. k-nearest neighbor algorithm:

k-nearest neighbour algorithm is a supervised machine learning classifier that makes classifications or predictions about the grouping of individual data points based on closeness.

- a basic but effective algorithm for classifying observations
- From textual visuals, you can recognise digits.

To determine the k value for the k-nearest neighbor algorithm we keep in mind:

- Not choose a very high k-value, because it becomes computationally expensive
- Not choose a very low value, since noise will influence the result



Therefore, plotting a graph for the testing dataset and the training dataset for different k values from the range of 1 to 11 and locating the point where both the line graphs are the closest/ meet can be chosen as the optimal k value. As per our graph, we choose 8 as the k value for the classification algorithm. This is basically a trail and error methodology to run all possible values of k in the algorithm and choose the best value by visualization.

Weights: Setting this value to be uniform, which means all point in each neighborhood are weighed equally.

Metric: The default 'minkowski' metric is set as the distance metric.

P: When p=1 it refers to the Manhattan_distance and Euclidean distance refers to p = 2. For selecting the p value it can be seen that the small p value works better for high dimensional vectors. So reducing the value makes other features play a bigger role in distance calculation. Therefore based on accuracy, p=1 is a better choice.

b. Decision tree algorithm:

Decision tree algorithm can be used to visually and explicitly describe decisions and decision making in decision analysis. It employs a decision-tree model, as the name implies.

It can be used with both categorical and continuous input and output variables. Based on the most significant splitter / differentiator in input variables, it divides the population or sample into two or more homogeneous sets (or sub-populations).

For the decision tree algorithm:

Criterion: Using both 'gini' and 'entropy' and tabulating the accuracy, this function is used to measure the quality of a split.

Max_features: Setting this value as none, which directly means that the number of features to be considered will be n_features when looking for the best split.

Max_depth: This is set as None, which means nodes will be expanded till all the leaves are pure.

Min_samples_split: Set as the default value = 2, which would be the min number of samples to split an internal node.

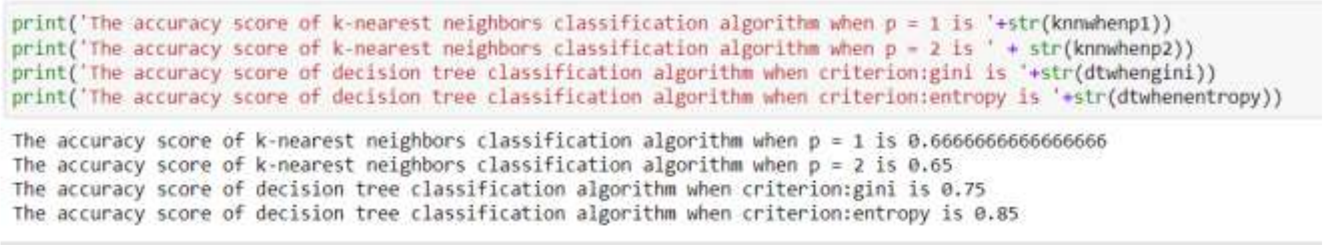
Min_samples_leaf: Set as the default value = 1, which would be the minimum number of samples to be a leaf node.

Tree Visualisation:

The decision tree algorithm can be visualized after computing using graphviz, and users can use it to relate to their hypothesis. The decision tree for both the criterion – gini and entropy are visualized and stored in a .dot file.

4. Results:

From our project result:



```
print('The accuracy score of k-nearest neighbors classification algorithm when p = 1 is '+str(knnwhenp1))
print('The accuracy score of k-nearest neighbors classification algorithm when p - 2 is ' + str(knnwhenp2))
print('The accuracy score of decision tree classification algorithm when criterion:gini is '+str(dtwhengini))
print('The accuracy score of decision tree classification algorithm when criterion:entropy is '+str(dtwhenentropy))
```

```
The accuracy score of k-nearest neighbors classification algorithm when p = 1 is 0.6666666666666666
The accuracy score of k-nearest neighbors classification algorithm when p = 2 is 0.65
The accuracy score of decision tree classification algorithm when criterion:gini is 0.75
The accuracy score of decision tree classification algorithm when criterion:entropy is 0.85
```

5. Discussion

As per our data science project, we can see that:

- The highest accuracy is computed for the decision tree algorithm is 0.85 with criterion = entropy.
- The lowest accuracy is computed for the k-nearest neighbors classification algorithm when p = 2.
- The highest accuracy obtained from the k-nearest neighbors is 0.6666666666666 when p = 1.
- The lowest accuracy obtained from the decision tree classification algorithm is 0.75 when criterion = gini.
- The accuracy of the decision tree algorithm is higher than the k-nearest neighbors algorithm.

6. Conclusion:

The decision tree algorithm for classification has a much higher accuracy than k-nearest neighbors classification. For both gini and entropy criterion, the accuracy value is much higher. Therefore, for our project, to achieve our goal in an efficient manner decision tree classification is the best algorithm in this case.

Apart from accuracy, the decision tree algorithm is also better in a lot of ways such as:

- Easy to understand
- Tree can be visualized
- Less data cleaning required
- They don't have assumptions on space distribution and classifier structure
- Useful in data exploration

7. References:

1. GeeksforGeeks. 2022. *Matplotlib Tutorial - GeeksforGeeks*. [online] Available at: < https://www.geeksforgeeks.org/matplotlib-tutorial/ > [Accessed 7 May 2022].
2. GeeksforGeeks. 2022. *Plotting graph using Seaborn | Python - GeeksforGeeks*. [online] Available at: <https://www.geeksforgeeks.org/plotting-graph-using-seaborn-python/ > [Accessed 7 May 2022].
3. Medium. 2022. *Machine Learning Basics with the K-Nearest Neighbors Algorithm*. [online] Available at: < https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761 > [Accessed 7 May 2022].
4. GeeksforGeeks. 2022. *Decision Tree Introduction with example - GeeksforGeeks*. [online] Available at: < https://www.geeksforgeeks.org/decision-tree-introduction-example/> [Accessed 7 May 2022].
5. GeeksforGeeks. 2022. *k-nearest neighbor algorithm in Python - GeeksforGeeks*. [online] Available at: < https://www.geeksforgeeks.org/k-nearest-neighbor-algorithm-in-python/ > [Accessed 14 May 2022].
6. GeeksforGeeks. 2022. *Python | Decision tree implementation - GeeksforGeeks*. [online] Available at: < https://www.geeksforgeeks.org/decision-tree-implementation-python/ > [Accessed 14 May 2022].