A picture containing text

Description automatically generated

RIGA TECHNICAL UNIVERSITY

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

INSTITUTE OF APPLIED COMPUTER SYSTEMS

Practical Assignment #2

“Fundamentals of Artificial Intelligence”

**Applying methods of machine learning**

**GitHub: https://github.com/beriafra/Practical-2**

Author: Afra Beria Baki

Course, group: 2-1

Student card no.:201ADB079

Checked: Alla Anohina-Naumeca

2021 / 2022study year

## **Introduction**

In this practical assignment I have chosen a dataset containing information about the passengers in Titanic. This dataset is taken from github. (<https://github.com/awesomedata/awesome-public-datasets>)

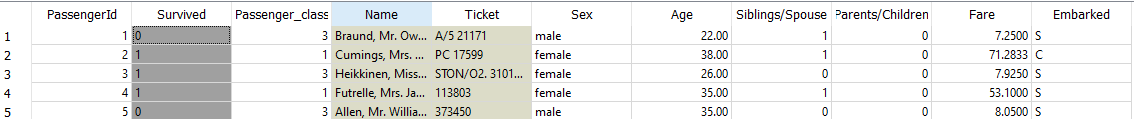
The data set consists of 9 features and 3 meta data with 891 instances. At first it had 2.2% missing values although they are filled using impute feature by considering of average/most frequent values. In this project I will be examining the factors and classifications of passengers that affected the survival from the huge sink of ship Titanic. We will use binary classification in this task.

Figure 1.1 shows the classes and its labels. PassengerId ist the ID for the passenger. Survived feature represents if the passenger survived or not, this will be our target feature to be examined.

|  |  |  |
| --- | --- | --- |
| Name | Value | Role |
| PassengerId | 1-891 | Identification of passenger |
| Survived | 0=deceased, 1=survived | Survival State |
| Passenger\_class | 1, 2, 3 | Ticket class of the passenger |
| Sex | Female, male | Gender |
| Siblings/Spouse | integer | Number of siblings/spouse on board |
| Parent/Children | integer | Number of Parent/Children on board |
| Fare | integer | Price of the ticket |
| Embarked | C= Cherbourg, Q= Queenstown, S= Southampton | Where people mounted |
| Name | String | Name of the passenger |
| Ticket | Varchar | Ticket of the passenger |

Figure 1.1


Figure 1.1



Example Table

# Exploring The Data:

## Scatter Plot:

Scatter plots allow us to examine the relationship between two continuous variables by combining the values of two quantitative variables and display them in a graph.

Figure 2.1, I created a scatter plot using fare and age to examine the survival. Here red dots indicate the survival and blue indicates perished. We can see that younger people tend to survive more than the older ones especially in low fared classes.

These features do not demonstrate good class seperability and it is an imbalanced classification.

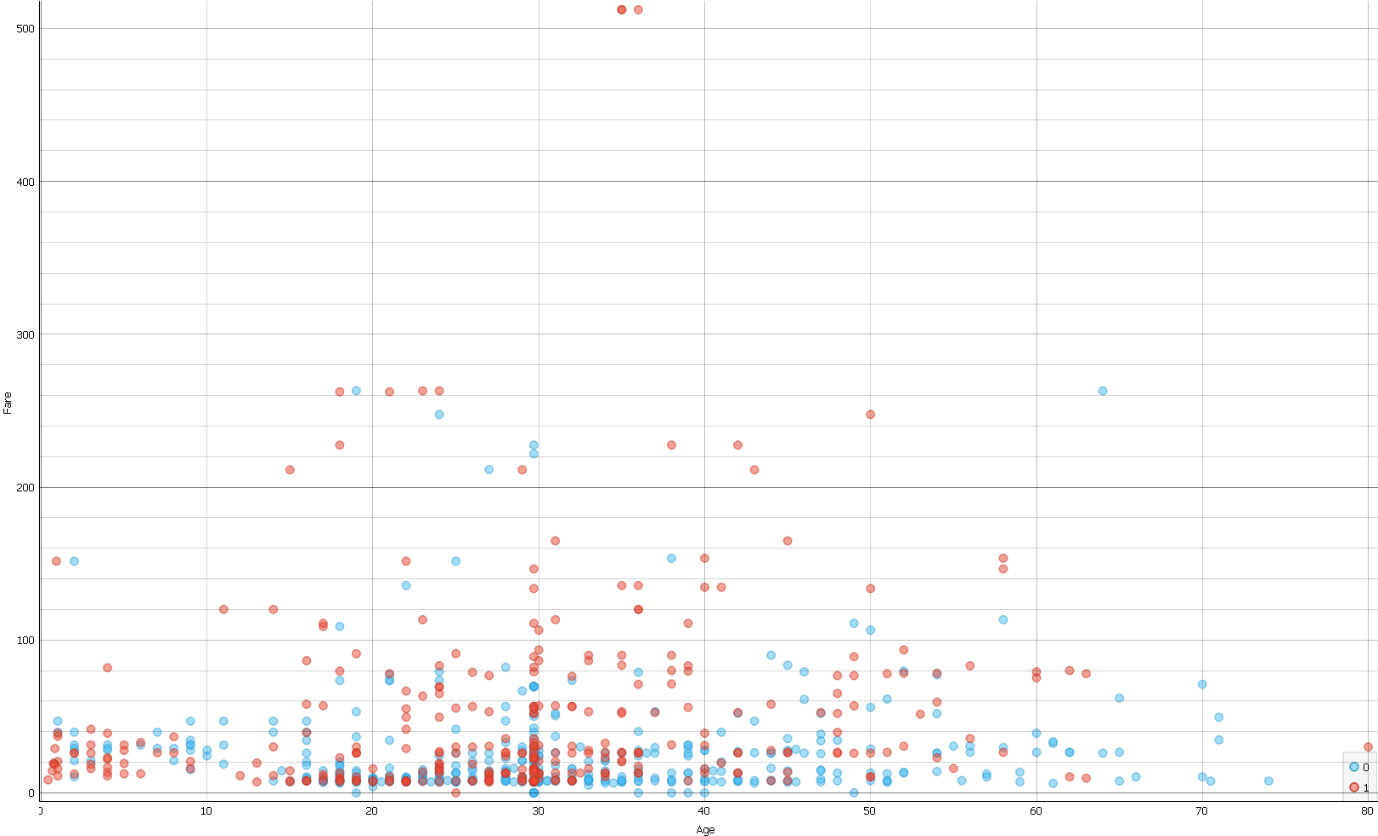


Figure 2.1

Figure 2.2 shows another scatter plot using fare and sex. This demonstrate a better class separability than figure 2.1 however we still cannot say it is well balanced. From the graph it is seen that more female survived from the sink than male at the same age boundaries.

Chart, histogram

Description automatically generated with medium confidence

Figure 2.2

Figure 2.3 visualize passenger class clustering using age and fare. This graph shows better demonstration of class separability as different classes banked up together.

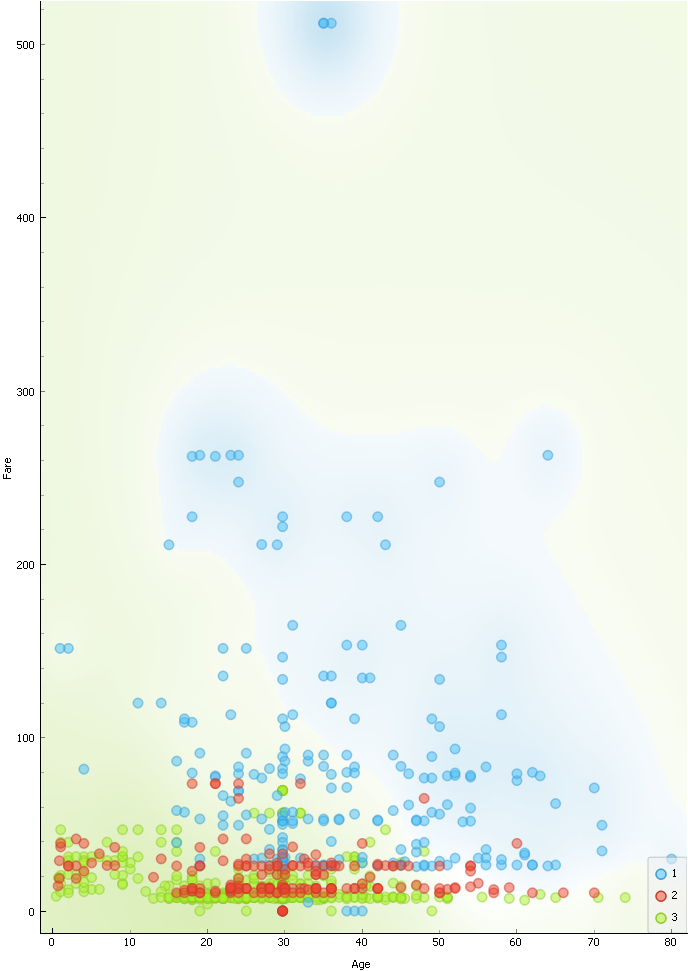


Figure 2.3

## Histograms:

Continuous variables are analyzed easily by using histogram charts. From orange tool different types of distribution tables could be used. Although it is advised to use relative frequency rather than absolute frequency when working on comparison of distribution of two data sets.

Figure 3.1 shows the survival by age and it suggests that there were vast amount of difference in the age 30 where there were 70% victims and only 30% survived the sink. Passengers aged 20-30 were more likely o die and infants were most likely live. The graph shows that only one passenger at the age of 80 lived. We even saw from this results in scatter plot.

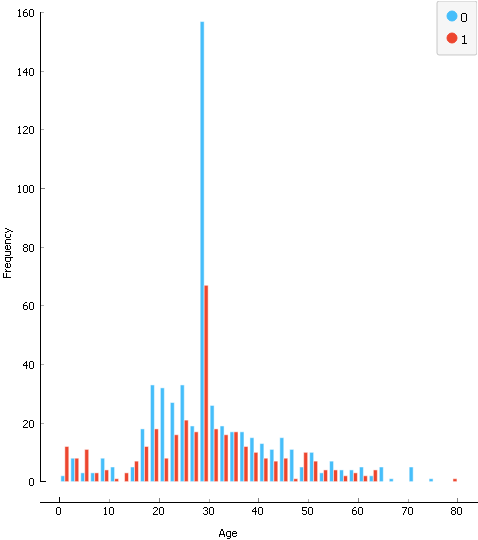


Figure 3.1

Another useful visualization would be to use passenger classes as it also contributes the chance of survival. It seen from the graph that there is a high probability that a passenger in class 3 would not survive the sink. Moreover it can be seen that almost 63% of people staying in first class had survival chance.

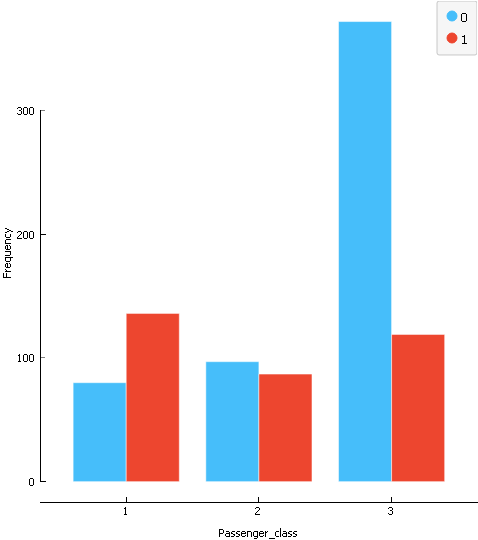


Figure 3.1

Figure 3.2 also visualize another predictor of our target, survival. Based on the below plot, female had about 72% survival chance while male had about 18%.

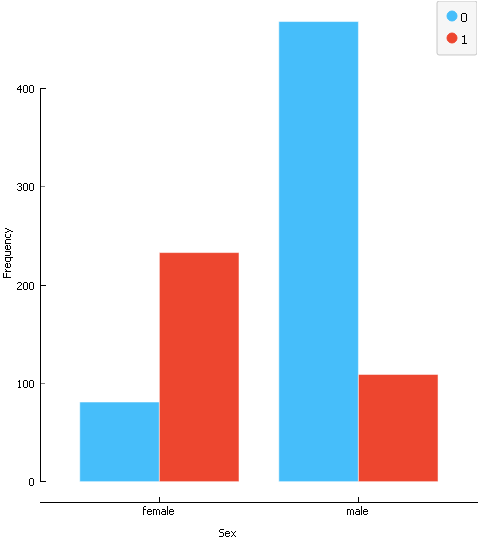


Figure 3.2

As we seen in the graphs, age, sex and passenger class were good predictors of our target, survival.

On the other hand, figure 3.3 shows distributions of different usage of features. PassengerId would not give us a proper analysis and it doesn’t provide suitable class separability. Siblings/spouse, parents/children and fare features would also not give better separability. In these distributions data are not evenly distributed and balanced. For instance, in feature “Fare” all data collapsed in first column but there are so few in the other columns.

A picture containing chart

Description automatically generated

Figure 3.3

## Distributions:

Data could be visualized using distributions just like we saw in the previous examples. Orange tool shows the central tendency of feature values as following; if it’s a categorical feature central tendency is the mode and if it’s a numerical feature its central tendency is mean value. Dispersion is the degree to which data is distributed around the central tendency (Figure 3.3).

Figure 4.1 shows normal distribution for age feature. Mean of people who survived is 28.55 and the ones that perished is 30.42 with standard deviations of 13.75 and 12.44 respectively. For a perfectly normal distribution the mean, median and mode will be the same value. So in our case it is not a perfectly normal distribution.

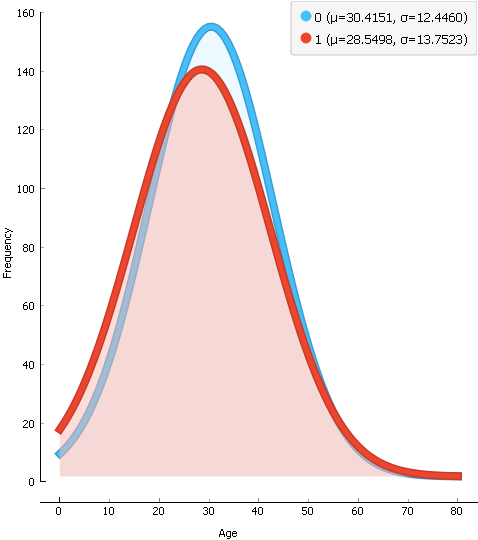


Figure 4.1

Below is another normal distribution showing the effect of passenger fare to survival. The mean value of survival is low compared to perish. From the graph it is seen that more people died in low fared classes compared to higher fared classes.

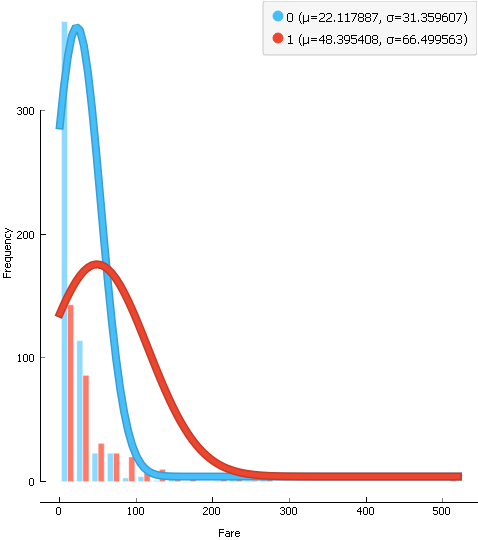


Figure 4.2

# **UNSUPERVISED LEARNING**

## Hierarchical clustering:

The optimal number of clusters will be determined via hierarchical clustering. Steps for hierarchical clustering is as follows:

Every point is placed in its own cluster, making each cluster a singleton.

Based on the distances from the distance matrix, it then combines the two points that are closest to each other. As a result, one cluster has been eliminated.

The distances between the new and old clusters are then recalculated and saved in a new distance matrix, which will be used in the following stage.

At the end, repeat steps 1 and 2 until all clusters have been combined into a single cluster with all points.

There are different types of linkage in hierarchical clustering, in this task we will demonstrate some of them. I also used Euclidian distance for demonstration.

Diagram

Description automatically generated with low confidence

Figure 5.1 shows an example for binary clustering for survival. Here, single linkage is used which is the minimum distance between two clusters. The length of the vertical lines in the dendrogram shows the distance between clusters. Here we have 2 clusters however it could be changed by cutting the line somewhere below the dendrogram. The cluster at the top C1 is the outliers we saw in scatter plot and this would not make a such a good cluster

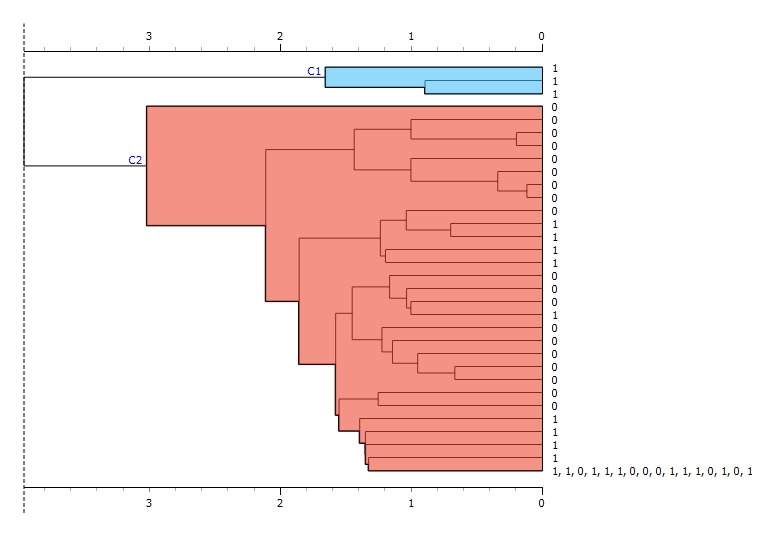


Figure 5.1

Below is another experiment with max depth set to 7 and linkage to average, which shows the distance between two pair in each cluster are added up and divided by the number of pairs. When the cut-off is placed at 4 we get 6 clusters.

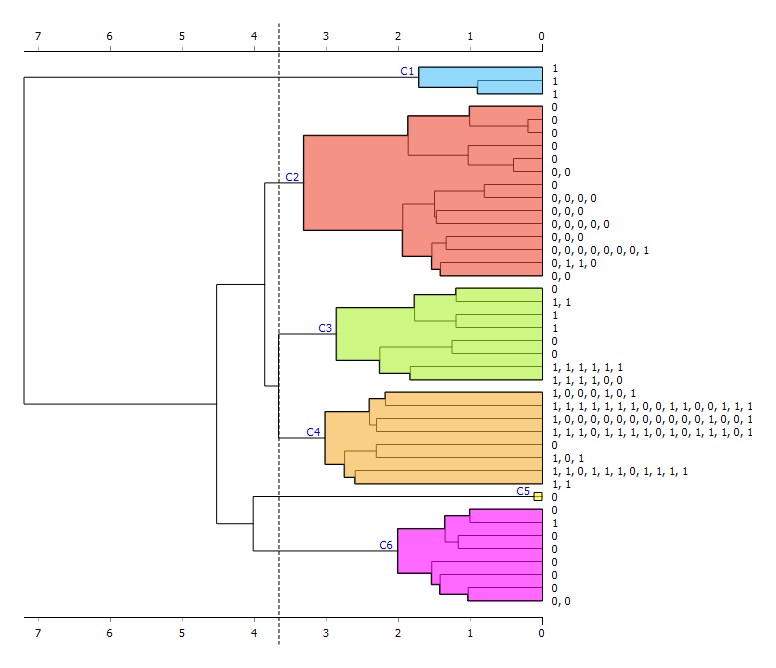


Figure 5.2

Figure 5.3 demonstrates another dendrogram with different hyperparameter, complete linkage method. This method is the furthest distance between the two points of different clusters. Using the same max depth gives more clusters in this method by 10 clusters on the same cut off. Similar to single linkage, the largest difference in heights occurs before the final combination.

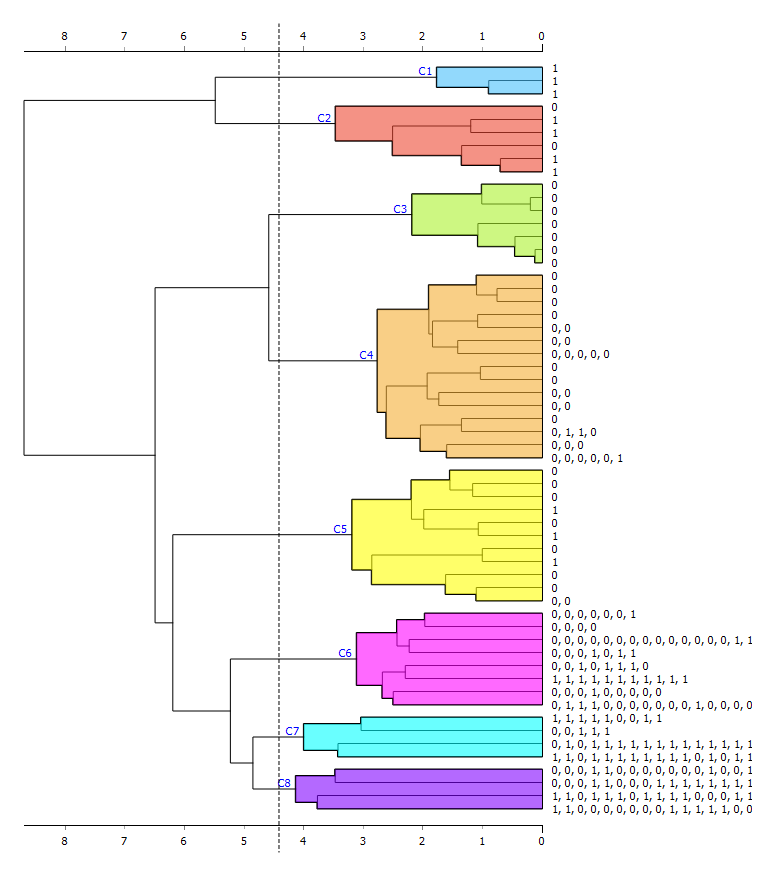


Figure 5.3

Figure 5.4 shows the MDS result for the above clustering. Even if we change the iteration number the cluster collapse in an uneven manner. We can conclude that these would not make a good clustering.

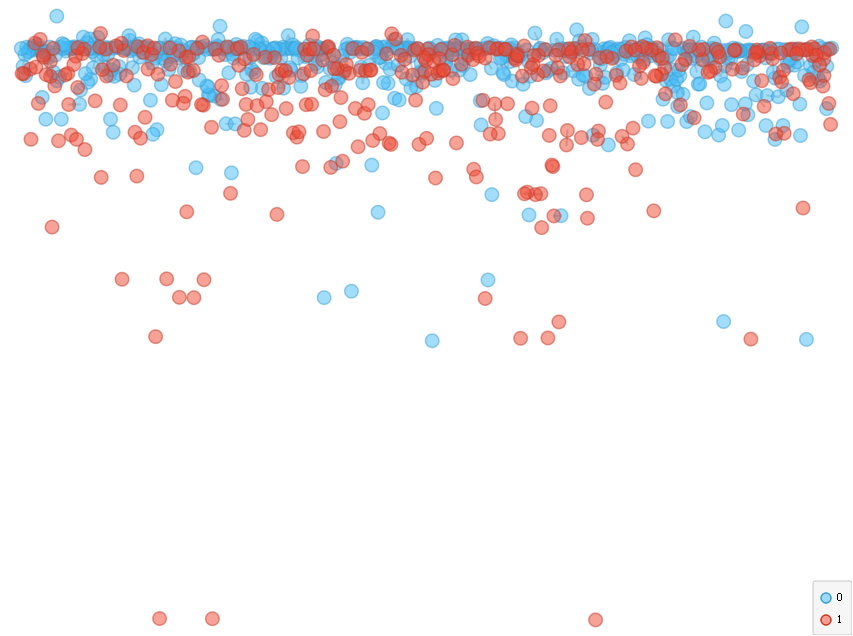


Figure 5.4

## K-means clustering:

k-means finds the (dis)similarities between different data objects. It aims to partition n observations into k clusters, and each observation will be placed to the nearest cluster. For this type of clustering, we will use silhouette score to evaluate the quality of clusters. The silhouette score falls within range -1-1. The higher the silhouette score the better the clustering will be.

In my example, we have two survival options, 1 and 2. The machine will find clusters using clustering algorithm. After those values will be assigned to them in the order it finds them, and we have only 2 options depending on a degree of randomness. Figure 6.1 demonstrates how many clusters I need to use, since second has the highest score, we will use that one. Figure 6.2 visualize the center of the cluster. If we connect this silhouette plot to a scatter plot, we would see which data object selected and the higher scores are indeed the center of the cluster. As we have seen from the previous analysis, survived feature does not really make good clustering.

Graphical user interface, application

Description automatically generated Chart

Description automatically generated

Figure 6.1

Figure 6.2

Let us know examine the results using different clusters (k). Below is the result after having 3 clusters.

Chart

Description automatically generated

Figure 6.3 – k=3

Chart

Description automatically generated

K=4

Chart

Description automatically generated with medium confidence

K=5

Background pattern

Description automatically generated

K-means is not advised when you have binary data since it requires continuous features as input. As we have seen in the previous demonstrations it does not compute the distances in a meaningful way, for example red lines have almost similar length. Let us now try with a feature that has 3 values, passenger class.

To conclude with after analyzing the plots and performance, this dataset did not give us sufficient separability. For almost all silhouette scores we got values of around 0, which indicates that the distance between neighboring clusters is not significant. Hence, data objects lie on the decision boundary. Although we did not get any negative values, therefore objects are assigned to correct clusters.

# **SUPERVISED LEARNING**

In supervised learning, we know what our dataset is and what the desired output should be from that data.

Supervised learning provides a function (a match between the input data and the result data) to be extracted from this information by feeding the data and the results from that data back to the machine. Thus, the machine learns the relationship between the data.

For this task I will divide my data into two parts, consisting of train dataset and test dataset using data sampler, as in Figure 7.1. I will split the data into 85:15 ratio so that train data will have 85% and test data will have 15%. As seen from the Figure 7.2, 758 data points used for training and 133 data points used for testing.

There are different types of classification tasks to use in supervised learning I will use 4 of them, decision tree, logistic regression, neural networks and kNN.

Graphical user interface, application

Description automatically generatedDiagram

Description automatically generated



Figure 7.2

Figure 7.1

## Decision Tree:

Tree-based methods have high accuracy, stability, and ease of interpretation. Unlike linear models, they can also map nonlinear relationships quite well. Decision tree also does not need normalization since they can both process numerical and categorical values. The decision tree helps to understand the survival status in a very user friendly way. At each level, the visualization is easier compared to reading decision tree summary.

The decision tree uses sex, passenger class, fare and age at 4 levels depth.

Diagram

Description automatically generated

Decision Tree

Maximal tree depth is one of the important hyperparameters in trees, If we do not specify it, the nodes will expand until all leaf nodes contain less than minimum value. The deeper the tree grows, the more complex the model will be as we would have more splits and it captures more information about the data. We can see from the below experiment that increasing max depth makes tree model less expressive and gives worse accuracy. It is seen that when we had higher maximum depth the diagonal of the confusion matrix also gives somehow worse predictions. The reason for this might be because it increased the flexibility and end up overfit the data and performance suffered.

Graphical user interface, text, application, email

Description automatically generated A picture containing table

Description automatically generated

Table

Description automatically generated with medium confidence

Graphical user interface, text, application, email

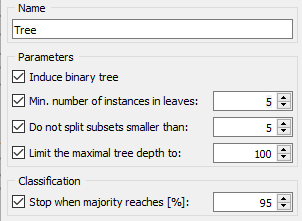
Description automatically generated 

Graphical user interface, application

Description automatically generated

Now let us change the value of minimum number of instances in leaves which allow to have leaves with a minimum number of samples and after that number will not be searched. Again we see that having a smaller minimum sample leaf gives better accuracy. Therefore we can say that this is alternative to fix the maximum depth hyperparameter.

From the confusion matrix we can say that increasing the number of minimum instances in leaves decreased the predicted values of true negatives and true positives. It is important to correctly find the survivals from the ship therefore we need to find the best hyperparameters to get the most accurate results.

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated Graphical user interface

Description automatically generated with low confidence

Graphical user interface, application

Description automatically generated

Defining the minimum number of samples required to split an internal node affect the performance of the model. For confusion matrix we are getting almost the same answer. When we examine the accuracy we see that having more splits end up having better accuracy. Although when we specify the minimum number of leaves we are guaranteed to have similar values of performance, no matter what the minimum split is.

Graphical user interface, text, application, email

Description automatically generatedA picture containing diagram

Description automatically generated

Graphical user interface, table

Description automatically generated with medium confidence

Graphical user interface, text, application, email

Description automatically generatedTable

Description automatically generated with low confidence

Graphical user interface

Description automatically generated

## Logistic Regression

Logistic regression tells us the probabilities that a record belongs to each of the two states, yes-no or yes-no, based on the training data we gave it. Logistic regression is easier to use as it does not want the assumptions that linear regression requires. The biggest difference between logistic regression and linear regression; In logistic regression, the target (independent) variable is to be categorical, mostly binary (yes-no, yes-no).

The characteristic of the logistic function is that it produces values ​​between 0 and 1 therefore it would be suitable to use it in survival analysis.

Although logistic regression does not have any important hyperparameters, we can demonstrate this type of classification for the sake of the task.

Below we applied 3 different regularization strength to evaluate and observe the model.

The weight coefficients shrink if we increase the regularization strength. It is seen that CA value decreases as we increase the regularization strength. This is mostly used to avoid overfitting of data. Observing F1 results will tell us the number of data objects classified correctly, so as we had stronger regularization we had less correctly classified data objects, giving us not adequate model.

Graphical user interface, application

Description automatically generatedTable

Description automatically generated with medium confidence

Graphical user interface, text, application

Description automatically generatedGraphical user interface

Description automatically generated with medium confidence

Graphical user interface, application

Description automatically generatedText

Description automatically generated with low confidence

## Neural Network:

Artificial neural networks emerged as a result of mathematical modeling of the learning process by taking the human brain as an example. It mimics the structure of biological neural networks in the brain and their ability to learn, remember and generalize. Learning process in artificial neural networks is carried out using examples. During learning, input and output information is given and rules are set.

Neural Networks has different hyperparameters such as number of hidden layers, activation function, learning rate, momentum etc. Below I will try different values of hyperparameters to get the maximum accuracy.

Let us start with having different hidden layers.

A picture containing table

Description automatically generated

Hidden layer :100

Table

Description automatically generated with low confidence

Hidden layer: 100,40,30

Graphical user interface, text

Description automatically generated with medium confidence

Hidden layer: 3,5

Table

Description automatically generated with medium confidence

Hidden layer 3,5,10

Text

Description automatically generated with medium confidence

Hidden layer: 5

I first started with having 1 layer and 100 neurons it gave pretty good accuracy and F1 result. Later I increased the number of layers and decreased the number of neurons. This modification did not yield better than the previous one. I continued having layers with smaller neurons and they did not turn out to be performing better as well. I realized that the performance is when we have more neurons in less hidden layers. At the end I tried having the smallest numbers in one single layer and I saw that the accuracy is not changing so I stopped.

Let us now examine using different activation functions using the best hidden layer to see if there are any changes.

Graphical user interface, application

Description automatically generated

Activation: Identity

A picture containing table

Description automatically generated

Activation: Logistic

Graphical user interface

Description automatically generated with medium confidence

Activation: Tanh

Text

Description automatically generated with low confidence

Activation: ReLu

As it seen from the observations, while having the same hidden layers, ReLu gave the best performance.

## kNN:

KNN is an algorithm that uses training datasets to find the k closest data points in some dataset. KNN is a supervised machine learning algorithm used for both regression and classification problems. It is often applied for pattern recognition.

This algorithm first stores and defines the distance between all inputs in the data, choosing the closest input to the query and outputs.

One pf the hyperparameters of kNN algorithm is number of neighbors (k) . Let us examine how different K affects the accuracy. I will start from smaller values to higher ones.

A picture containing text

Description automatically generated

K=2

A picture containing graphical user interface

Description automatically generated

K=3

Table

Description automatically generated with medium confidence

K=4

A picture containing table

Description automatically generated

K=5

Observations show that smallest number of k that gives the best accuracy and F1 result is 3, after that the accuracy of the algorithm decreases.

Now let us see if distance metric affects the accuracy of the model.

Table

Description automatically generated with low confidence

Manhattan

Graphical user interface

Description automatically generated with low confidence

Chebyshev



Euclidean

We can conclude from the tests that Manhattan distance metric yielded the best compared to other metrices.

## Discussion:

Finally we can compare all above algorithms to see which one gives the best performance and accuracy. I will also add other types of algorithms to see.

Table

Description automatically generated

From the above table we can conclude that random forest algorithm provides the best performance with 0.827 CA which is quite good number and 0.823 F1 score, followed by decision tree and neural networks.

# Workflow:

Diagram

Description automatically generated

# References:

The license for the file:

*The MIT License (MIT)*

*Copyright (c) 2014-2015 Xiaming Chen and other contributors to this list.*

*Permission is hereby granted, free of charge, to any person obtaining a copy*

*of this software and associated documentation files (the "Software"), to deal*

*in the Software without restriction, including without limitation the rights*

*to use, copy, modify, merge, publish, distribute, sublicense, and/or sell*

*copies of the Software, and to permit persons to whom the Software is*

*furnished to do so, subject to the following conditions:*

*The above copyright notice and this permission notice shall be included in all*

*copies or substantial portions of the Software.*

*THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR*

*IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,*

*FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE*

*AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER*

*LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,*

*OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE*

*SOFTWARE.*

<https://derangedphysiology.com/main/cicm-primary-exam/required-reading/research-methods-and-statistics/Chapter%203.0.2/variability-dispersion-and-central-tendency#:~:text=Central%20tendency%20is%20described%20by,standard%20deviation%20and%20standard%20error>

<https://www.javatpoint.com/hierarchical-clustering-in-machine-learning>

<https://towardsdatascience.com/neural-networks-parameters-hyperparameters-and-optimization-strategies-3f0842fac0a5>