Thesis Report

Shubham Mathur
Matrikel Nr.: 00805119

Supervisor:
Prof. Dr. Gefei Zhang

Software Engineering For Industrial Applications Department of Computer Science

Hof University Of Applied Science

Abstract

The sinking of the Titanic is one of the most infamous shipwrecks in history. On April 15, 1912, during her maiden voyage, the widely considered "unsinkable" RMS Titanic sank after colliding with an iceberg. Unfortunately, there weren't enough lifeboats for everyone on board, resulting in the death of 1502 out of 2224 passengers and crew. While there was some element of luck involved in surviving, it seems some groups of people were more likely to survive than others. The Project is about Predicting the Survival or death of a given Test Dataset using another Train Dataset. The algorithm used in the project is K Nearest Neighbour Algorithm. Initially, the Data Analysis was done and data was understood along with the features and their relation to survival. After selecting the features responsible the Knn Algorithm was implemented.

Acknowledgements

I would like to take this opportunity to express my sincere gratitude to Prof. Dr. Gefei Zhang for his constant supervision and support in this course of Analytical Information Systems. The tutorials provided about Python by the Professor was very helpful in understanding the Project and starting my journey in the field of python to which I was a total beginner. The professor also gave a brief about how to make the project better and even gave advice over how to make the most of the course further after completion and use python as a career choice and ignited an interest in the field which will help me in the Master's thesis .

Contents

Chapter 1

Introduction

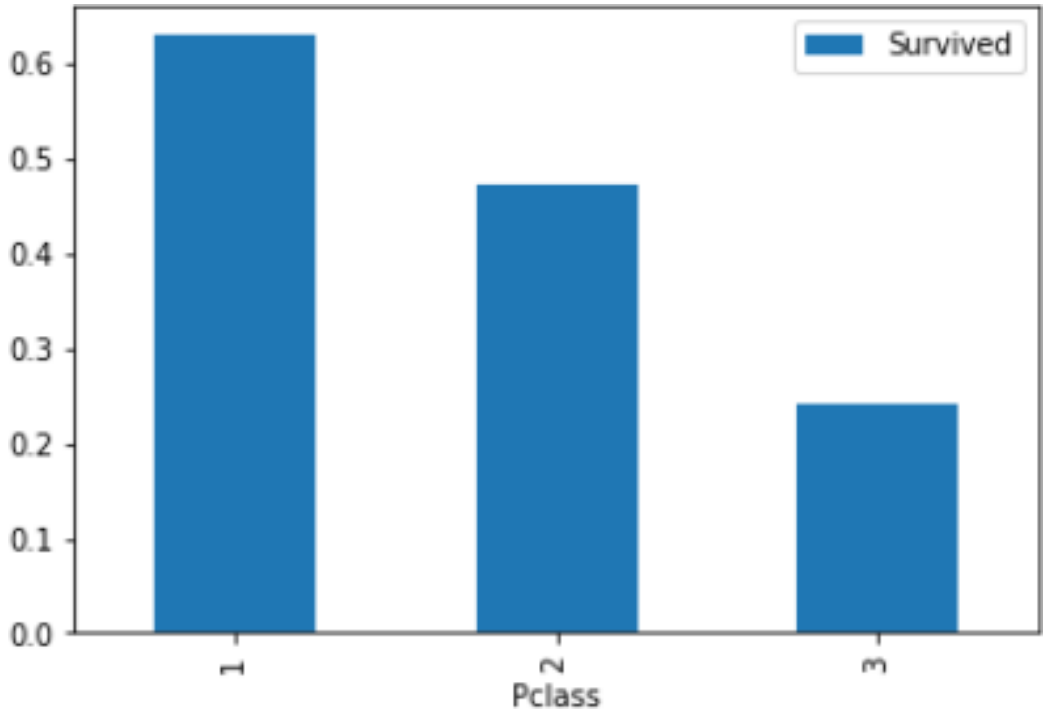
On 5 April 1912, A ship which was called "The Unsinkable " sank in the deep ocean as it collided with an Iceberg which ultimately broke it apart and it became a tragedy where a lot of people lost their lives and those who live to tell the tale were scarred mentally for all their life. The following project consist of the Dataset of Titanic passengers with their details. There are two datasets, First is the Train dataset and the other one is the Test dataset. The train dataset consists of details about the passengers with the data of their survival or death. The task is to use this data to predict the life or death of the data in the Test dataset which has only 1 feature missing which is Survived. So the goal is to make predictions about all the Passenger in Test Dataset using one of the algorithms.

Chapter 2

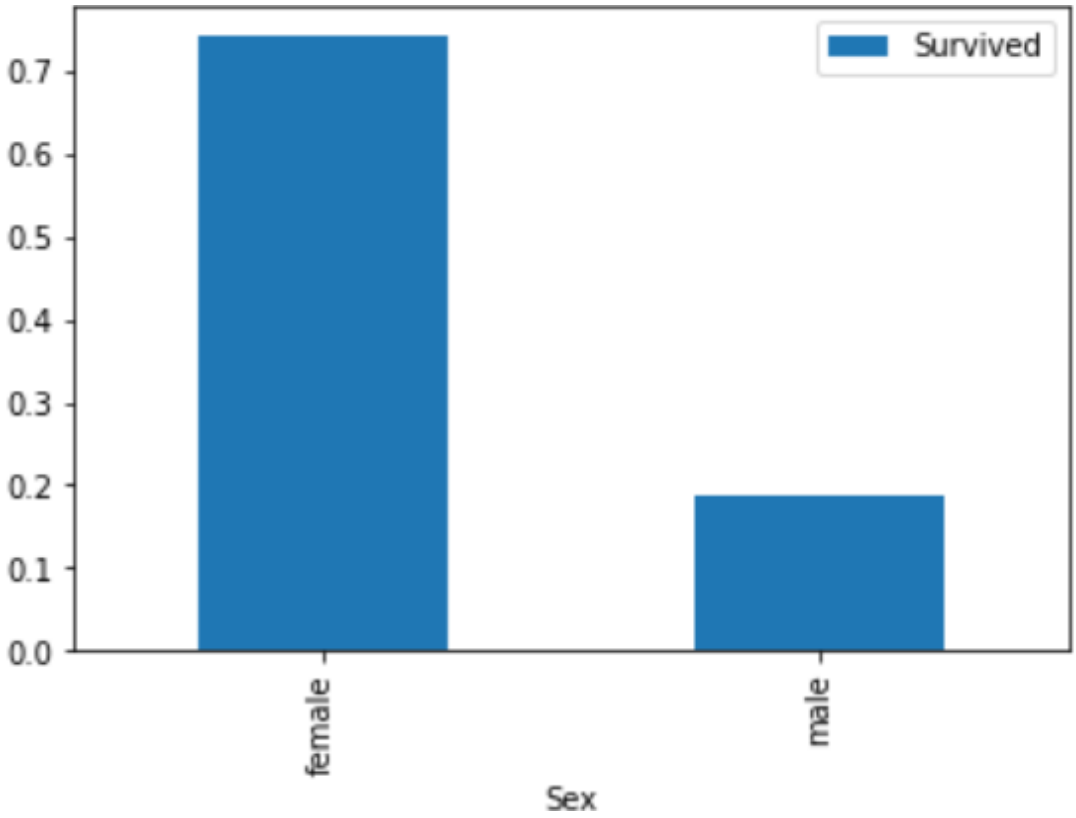
DATA ANALYSIS

After Analysing the Train dataset, it was clear that only a few features are more important than others for the predictions. It was done by eliminating less informative features using python.Numpy and Matplotlib libraries were really helpful in the task. It was clear that the ticket number isn't very important as it doesn't give any important information which can be related to survival. The ticket price can be linked to Passenger Class so both of them together are not necessary, only one of them is necessary and Passenger class looks much better in case of survival projections. So the Ticket cost feature is eliminated but not the Passenger class. The data analysis was done and the projections of passenger class made it clear that the better the class of the passenger, the more chances the passenger had, in surviving as the data showed that more percentage of first-class passengers survived more than the second and third class, and more percentage of the second class comparatively

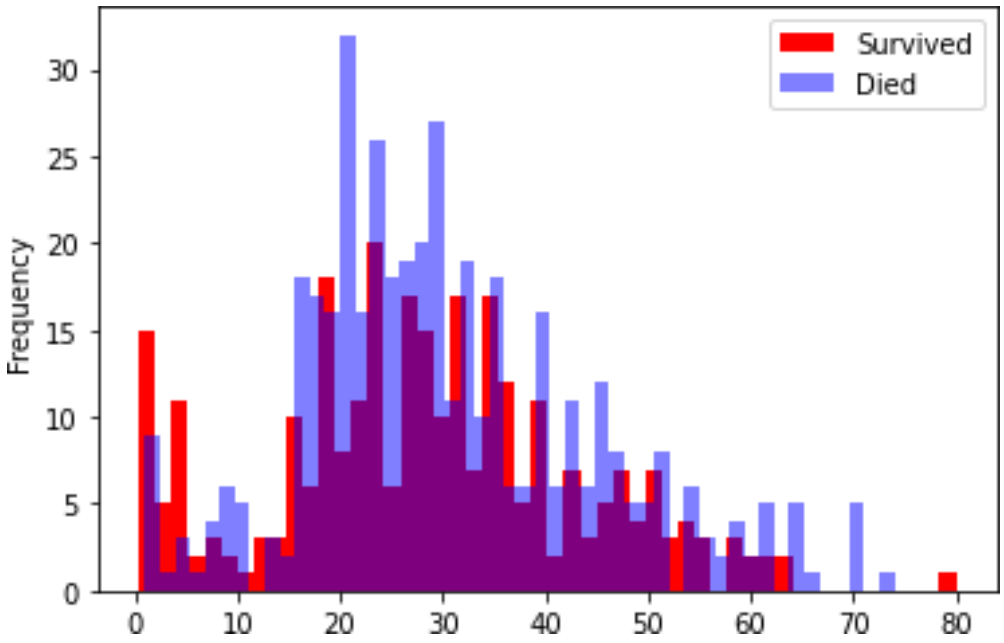
survived more than the third class as shown in the graph below



Another thing the analysis made clear was about Females,who had more chance to survive than Men as the females have a lot more percentage of survival than Men in the Train dataset as depicted in the comparison below.This was one of the important things which helped to select the correct data set feature of Gender.



For a better understanding of data for the computer, features were changed to numeric values as it is easier to sort , compare , make graphs and fill empty cells which helped a lot in handling the data well. As age can also be a major factor deciding the survival, data analysis was done on age. After the data analysis it was still not sure that Age played an important role in the survival of the passengers as many young people survived but died also as shown in the graph below.



For a better understanding of data for the computer, features were changed to numeric values as it is easier to sort, compare, make graphs and fill empty cells which helped a lot in handling the data well. As age can also be a major factor deciding survival, data analysis was done on age. After the data analysis, it was still not sure that Age played an important role in the survival of the passengers as many young people survived but died also as shown in the graph below. So it can't be said clearly.So the final features selected were Gender and Pclass .

Chapter 3

K- Nearest Neighbour Algorithm

The k-nearest neighbours (KNN) algorithm is a simple, easy-to-implement supervised machine learning algorithm that can be used to solve both classification and regression problems. The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other. k-NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until function evaluation. Since this algorithm relies on distance for classification, normalizing the training data can improve its accuracy dramatically.

Both for classification and regression, a useful technique can be, to assign weights to the contributions of the neighbours, so that the nearer neighbours contribute more to the average than the more distant ones.

The distance between neighbours can be calculating using one of the three distance

functions which are Euclidean Distance, Minkowski Distance and Manhattan Distance. Euclidean Distance is used in this project.

The neighbours are taken from a set of objects for which the class (for k-NN classification) or the object property value (for k-NN regression) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required

These are the steps to implement Knn Algorithm:

1. Load the data

2. Initialize K to your chosen number of neighbours

3. For each example in the data

3.1 Calculate the distance between the query example and the current example from the data.

3.2 Add the distance and the index of the example to an ordered collection

4. Sort the ordered collection of distances and indices from smallest to largest (in ascending order) by the distances

5. Pick the first K entries from the sorted collection

6. Get the labels of the selected K entries

7. If regression, return the mean of the K labels

8. If classification, return the mode of the K labels

In this particular project, the neighbours are the values from Train dataset and the points which will find neighbours will be the data points from Test Datasets.

3.1 Distance Function

There are 3 Distance functions which can be used in Knn Algorithm .

1. Manhattan Distance.

$$Mdist = |x2 - x1| + |y2 - y1|$$

2. Euclidean Distance.

$$d(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

3. Minkowski Distance.

$$D(\mathbf{x}_i, \mathbf{x}_j) = \left(\sum_{l=1}^{d} |x_{il} - x_{jl}|^{1/p} \right)^{p}.$$

3.2 Choosing the K value

The K value is the number of neighbours with which the distance is taken so to calculate the distance and select the neighbours, which can help determining the outcome .There are few approaches to tackle the question: .

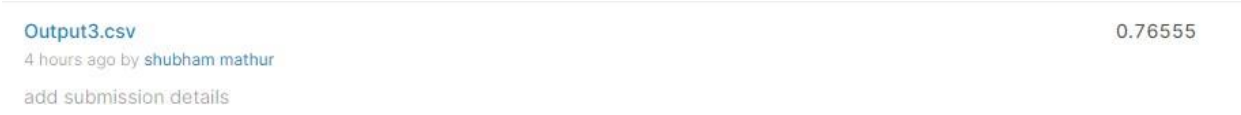
It is a better practice to take K as odd as even value can bring confusion if both results are tied and it is not possible to favour one outcome. Also if K is very small it is not possible for the algorithm to make accurate predictions as there can be certainty accounting less number of neighbours compared to each other and thus a larger K is better but it shouldn't't be very large so to avoid over fitting.Also, larger values might have values from other classes to put in some confusion. The state of the art approach suggested by experts is to take K as the square root of total number of observations which is exactly what was implemented in this project

Chapter 4

Accuracy

4.1 Accuracy of the Algorithm

The accuracy for the model was 76.5 percent when 2 of the features which are Gender and Passenger Class were used. It was 39 percent and 60 percentage when other features were used. The increased accuracy was due to the fact that Gender and Passenger class gave a clear picture for the predictions, far better when other features were used. Thus, these features were used.The following images shows the accuracy achieved selecting various features and then gradually increased accuracy and then finally selecting the two features Gender and Pclass for the accuracy of 76.5 percent .



for cross validation ,I made 2 dummy datasets and then run knn over it to verify if it is working or not and I got similar results for the dummy data set ,too.

4.2 How can a better accuracy be reached?

Another question is how can the accuracy be increased? This can be done by using doing some extra work or doing minor changes.

- Firstly, **Principle Component Analysis (PCA)** can be used to find the best features to compare between databases that can fetch the better accuracy.
- Also , cross validation can be used to determine the optimum value of K or neighbours that can be used for the Knn Algorithm .The optimum value can increase accuracy .Also to make the program faster ,Manhattan distance can be used to increase a speed .Manhattan distance formula has no root function which can make it a little faster to process eliminating the extra operation in each step.

Chapter 5

Application of K-nearest neighbour Algorithm

- The cutting edge frameworks are currently ready to utilize k-nearest neighbor for visual pattern recognition to filter and hidden bundles in the bottom bin of the shopping cart at checking out. On the off chance that an item is identified that is an accurate counterpart for an article recorded in the information base, at that point the cost of the spotted item could even consequently be put on the customer's bill. While this automated charging practice isn't utilized widely as of now, the innovation has been created and is accessible for use.
- The pattern recognition is also used for surveillance in the retail business in CCTV. modern surveillance system is smart enough to analyze and interpret video data by itself, without a need for human assistance.

5. Application of K-nearest neighbour Algorithm 13

- K-closest neighbor is likewise utilized in retail to recognize designs in Credit card use. Numerous new exchange examining programming applications use kNN calculations to break down register information and spot strange examples that show dubious action.

Chapter 6

Pros and Cons

6.1 Pros

- It stores the training dataset and learns from it just at the time of making time predictions. This makes the KNN algorithm a lot quicker than different algorithms that require training for example SVM, Linear Regression and so on
- Since the KNN calculation requires no training prior to making predictions, new data can be added flawlessly which won't affect the precision of the algorithm.
- It is very easy to implement as it requires only 2 parameters ,the value of K and the distance function.

6.2 Cons

- Does not work so well with large datasets because in large datasets the cost of calculating the distance between the new point and each existing points is huge which degrades the performance of the algorithm.
- In case of large dimensions It is not easy for the algorithm to calculate the distance so it doesn't work so well
- Standardization and normalization are needed for the data to work well with the algorithm

Chapter 7

Implementation of Knn Algorithm

Finally using the selected Algorithm and applying it using the features from the dataset, It was possible to implement the KNN Classifier algorithm on the data after filtering and filling up missing values. NUMPY,Math,CSV and Pandas were the only libraries used for the algorithm .Functions were defined for the various tasks such as the (Euclidean Distance) function which calculated the Euclidean distance . The (get neighbors) function which helped in getting the neighbours in the Train dataset for the given datapoint of the Train dataset After this a function called (predict classification) was defined which used to predict the outcomes using the neighbours provided by (get neighbor) function.An array was made to keep the tempororay data and then later transfer it to CSV file which was uploaded to the titanic data set to see the accuracy .

Bibliography

References

- https://en.wikipedia.org/wiki/K-nearest_neighbours_algorithm.
- https://www.kaggle.com/c/titanic
- https://www.codespeedy.com/compute-manhattan-distance-between-two-points-in-cpp/
- https://neo4j.com/docs/graph-algorithms/current/labs-algorithms/euclidean/
- https://datascience.stackexchange.com/questions/38078/minkowski-distance-with-missing-values
- https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761
- https://math.stackexchange.com/questions/128255/what-is-the-correct-definition-of-minkowski-distance
- https://www.dummies.com/programming/big-data/data-science/solving-real-world-problems-with-nearest-neighbor-algorithms/
- http://theprofessionalspoint.blogspot.com/2019/02/advantages-and-disadvantages-of-knn.html/
- https://sciencing.com/how-to-calculate-euclidean-distance-12751761.html
- https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.iloc.html

- https://seaborn.pydata.org/
- https://www.geeksforgeeks.org/python-pandas-dataframe-astype/