Titanic Machine Learning from Disaster

Jesús Moral Aranda

East China University of Science and Technology

2023— 2024 Term 1

《Pattern Recognition》Course Paper 2023.11

Class: Pattern recognition Student ID: 23050069

Name:Jesús Moral Aranda

School of Information Science and Engineering

Professor 赵海涛 Score_____

Requirements：

1. The paper shall be completed independently and shall not be copied;
2. The paper includes title, abstract, introduction, main body (method and experiment), conclusion, thanks and references. The format meets the requirements of Elsevier Templates;
3. In the method part, the methods used should be described in detail;
4. The experimental part should give the description of the experimental data. The experimental results should have relevant tables， figures and analysis;
5. Give the conclusions and summarize the advantages and disadvantages of the use methods;
6. List relevant references.

Comments：

Contents

1 Abstract

The purpose of this work is to become familiar with classification problems and the different algorithms that are used. To do this, we work with the Titanic classification problem. Since my knowledge of how to make an algorithm is very limited and I do not want to do a purely bibliographic or compilation work ('Forever reading, never to be read'), a statistical analysis is performed to at least try to see what could come out a priori from the results of the algorithms.

2 Introduction

On April 15, 1912, RMS Titanic sank after colliding with an iceberg. Unfortunately, there weren't enough lifeboats for everyone onboard, resulting in the death of $\frac{1502}{2224} = 67.5\%$ 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.

In this challenge, is usually asked you to build a predictive model that answers the question: "what sorts of people were more likely to survive?" using passenger data (ie name, age, gender, socio-economic class, etc).For that we use two files, one titled "train.csv" and other titled "test.csv".

Train.csv will contain the details of a subset of the passengers on board (891 to be exact) and importantly, **they reveal whether they survived or not**.

The test.csv dataset contains similar information but **does not if they die or not for each passenger**. It's the job of this article to predict these outcomes.

Sumarizing, using the patterns we find in the train.csv data from the passenger 1-891, we predict whether the other 418 passengers from 892-1309 on board (found in test.csv) survived or not.

An statistical analysis of the data will be made before go to the algorithms in order to have a better understanding of the result that we could obtain by the algorithms. We notice that kaggle give us the proportion of survivers of the population in the introduction of the the competition[1]. In sample I we also have the sample proportion of survivers P_I. We would like to estimate the parameter: propotion of survivers of sample II , P_{II}. So we define $P_2 : \#$ number of survivers/418. Where in principle it takes values from [0,1]. Since 418 is the total number of people to be clasifier as dead or not in sample II.

First, we have to notice that we know the proportion of the population P, that is $P = 1 - \frac{1502}{2224}$ and the proportion of survivers in the sample I is $P_1 = 342/891$. This is a first restriction in our clasification problem, the number of survivers can't be greater than $722 - 342 = 380$. In other words, in sample 2 , $P_2 \leq \frac{380}{418} = 0.9$. This a maximum limit for P_2, we would like a point estimation and a confindence interval. This proportion P_2 must be obeid by our classifiaction algorithms.

We also will do a descriptive statistical analysis of sample I, and how it is related whit sample II. In other words we would like to know if they share the same proportions, if so, we could use sample I to estimate over sample II. We will see the diferent proportions by gender,age, location and social status of the not survivers and other measures of the sample I and II.

[1] `https://www.kaggle.com/competitions/titanic/data`

3 Statistics analysis of data

3.1 Descriptive analysis of sample I

In figure [5] we can find the proportion of deaths for population and sample I. We can see that both proportions are very closed. **This is a good indicative that sample I have been choosen ramdomly from the population**.

We can see also that the percentage of men over Titanic was much bigger than women, how ever the percentage of women that died was really low in comparison with men. Unfortunately we can also see that the percentage of children and teenagers that survived were not so big as one could expect and even worse for old people. We can also see clearly that people from 1st class got to survived much more than those from 2nd and 3thr class. Finally we can see that many people embarked from Southampton and those from Queenstown were really unlikely.

The matlab code for this section can be checked in Apendix 6.1.

3.2 Descriptive analysis of sample I vs sample II

In this section we have made the comparison between both samples to see if both share the same proportions.The answer is absolutelly yes as we can see in figure [6], in all the aspects, age,gender,social scale and location of embark.

Like sample I seems to be ramdomly sampled from the population and sample II is similar to sample I, **then sample I is chosen randomly from population and we can use sample I and population to make estimations over sample I**[2]. For example, we can see that sample I and II have the same proportion of women, so the proportion of death remain almost the same, because women are very unlikely to die.So if there were much more women in sample I, sample II must have high proportion of deaths. In this case if sample II follow a binomial distribution (as we are going to supose) for the number of deaths, sample I follow a poisson distribution because p$\rightarrow$ 0 and n is really large.

3.3 Estimation on the proportion of survivers on sample II.

First imagine we do the experiment of doing Titanics, this is a ramdom process. You can imagine this as in other paralel universes no one died in Titanic, only one, two,... There are more universes (like modal logic) where the exact number of victims is 722 than those who only one person die. We define X : # number of surviers in Titanic population.In figure[2] we can show the distribution of probabilities of $X \sim B(N, P)$ whit N the total number of passenger in Titanic and P the proportion of survivers. Check that $X = 722$ is the most probaple value and the mean $E[X] = NP = 722$ in our model.

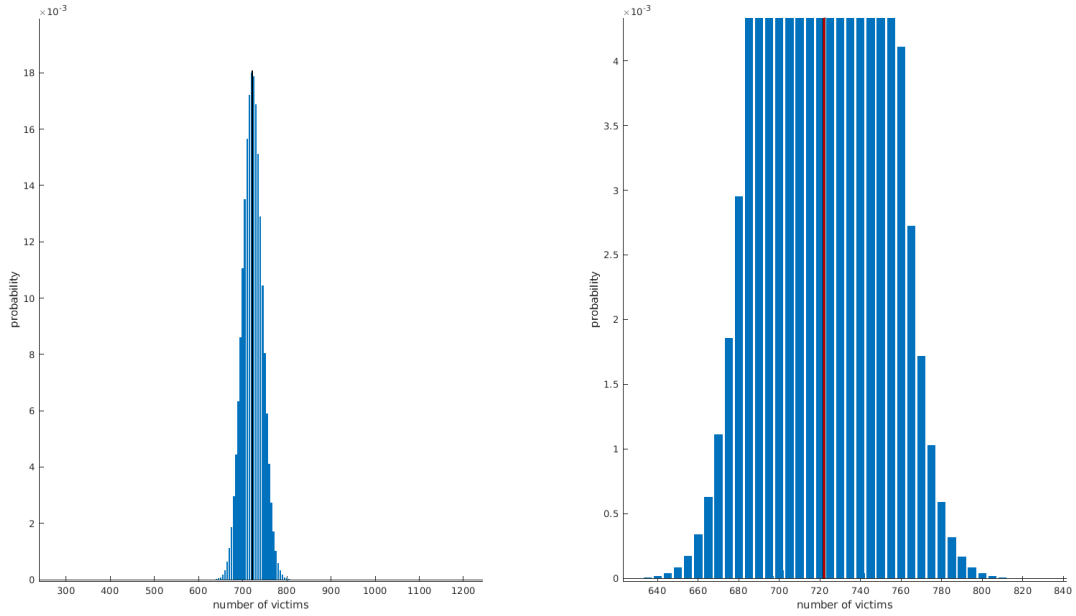


Figure 1: B(2224,0.325) of the number of victims in population

[2]Remember that if this happen, random variables sampled on this way from population follow similar distribution function as population

With sample I hapen something similar. We define X_2 : # number of survivers in sample I. With $X_2 \sim$ B(891,0.38) as we can see in figure [2]. In this case the mean value of survivers is 342. Again, notice that we select conveniently, in our model , this value to be the mean value and most probable because is the one we got from our data set 'train.csv'.

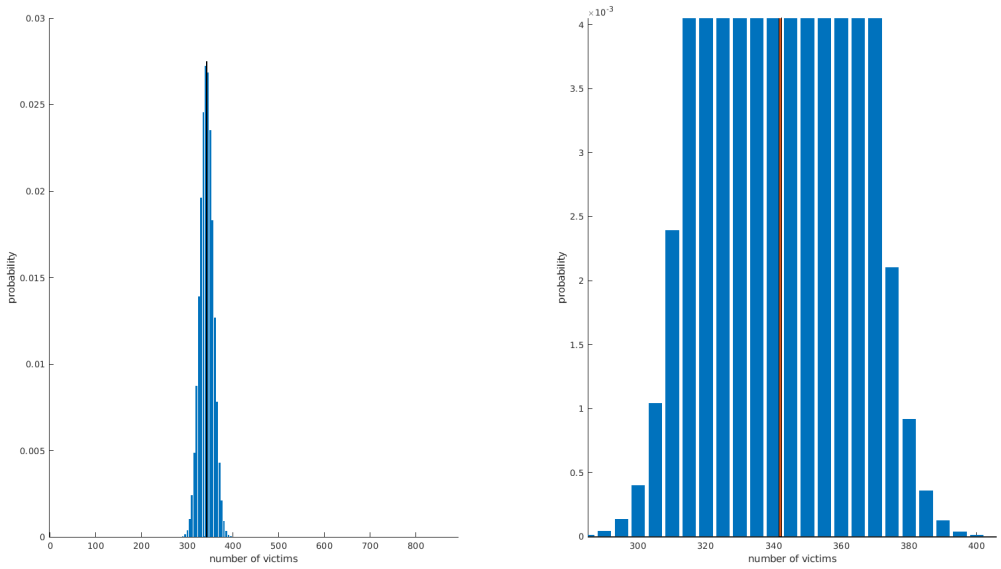


Figure 2: B(891,0.38) of the number of victims in sample I

Finally we define X_2 : # number of survivers in sample II, but in this case we don't know the proportion of survivers. We only know the sample size $N_{II} = 418$. How ever we know:

$$X = X_1 + X_2 + \epsilon \quad \rightarrow \quad X_2 = X - X_1 - \epsilon$$

Where $X \sim B(2224, 0.325)$, $X_I \sim B(891, 0.38)$ and ϵ : # the number of survivers that are not in sample I and sample II. From ϵ we also only know that $N_\epsilon = 915$. In our discusion on section 3.2 we saw that both samples had equivalent proportions of gender, age,location and status. So we could suposse that they were sampled from population randomly. As consecuence neither of them had an overconcentration of survivers. So ϵ epsilon is also sampled randomly because is the remain of survivers that are not in sample I and II. As ϵ has almost the same parametral space (915 $\sim$ 891) as X_I. We are going to suposse $\epsilon \sim B(915, P_\epsilon)$ and $\frac{N_I}{P_I} = \frac{N_\epsilon}{P_\epsilon}$. So we get $P_\epsilon = 0.39$ and $E[\epsilon]$=361, as we can see in figure 4. In this case we have:

$$X_2 = X - X_1 - \epsilon \quad X \sim B(2224, 0.325),\ X_I \sim B(891, 0.38) \quad \textit{and} \quad \epsilon \sim B(915, 0.39)$$

Notice that i got the uncertainty interval as the absolute error. Where i use the standar desviation of the population as estimation of the variance. Notice also that the distributions are symetrical and so the confidence interval.

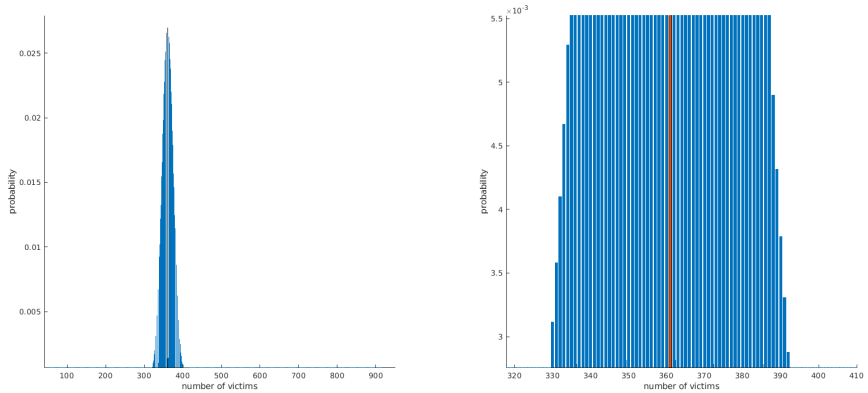


Figure 3: B(915,0.39) of the number of victims in sample I

So to get the distribution for X_2, we have made the **convolution** because the probability density of the sum of random variables is the convolution of their distributions, we get:

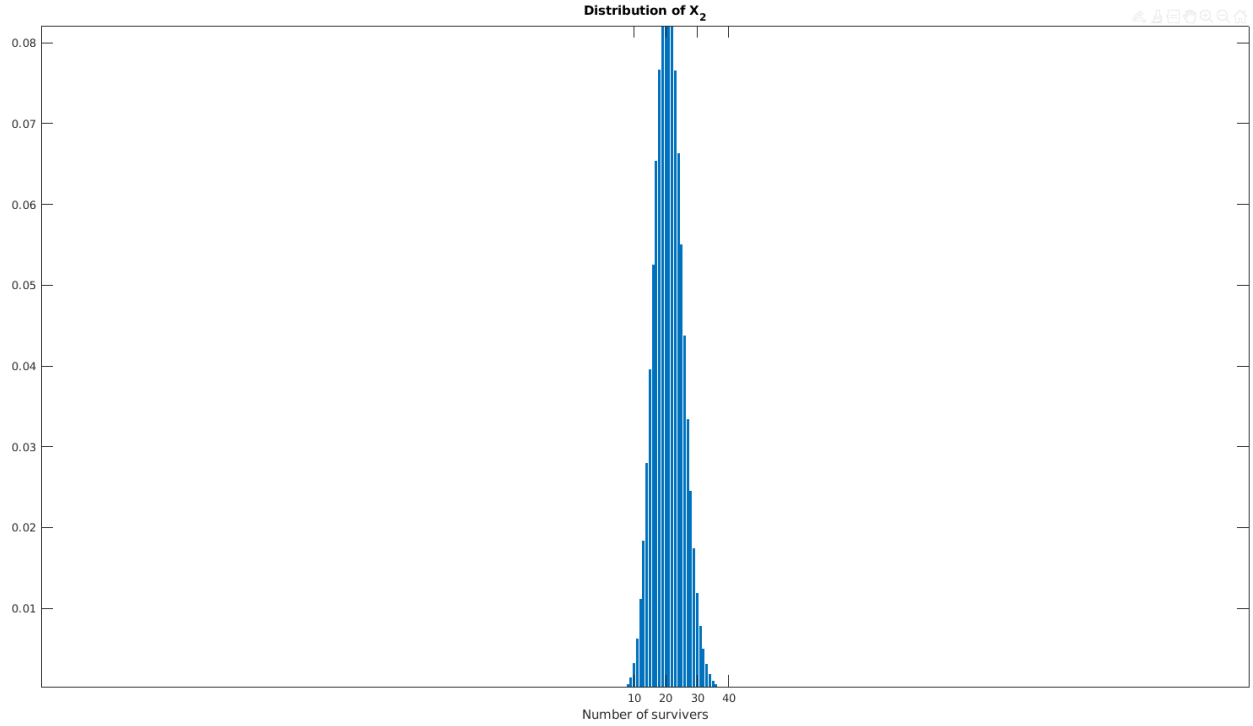


Figure 4: Probability density of sample II

We get from $P_{II} = E[X_{II}]/N_{II}$ with $E[X_{II}] = 20$:

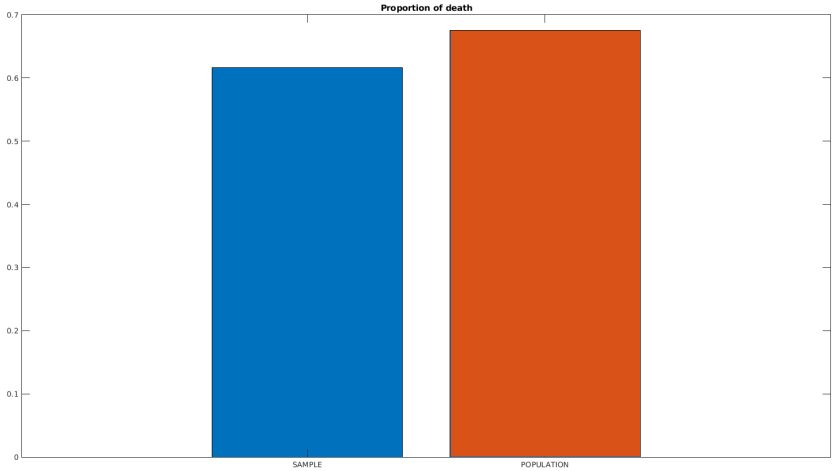
$$\boxed{P_{II} = 0.05}$$

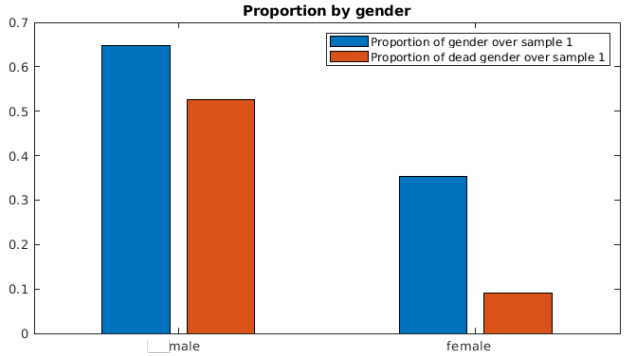
And $\sigma_{X_{II}} = 4.45$, so $\frac{\overline{X}\pm\sigma}{N_{II}} = P_{II} \pm \Delta P_{II}$:

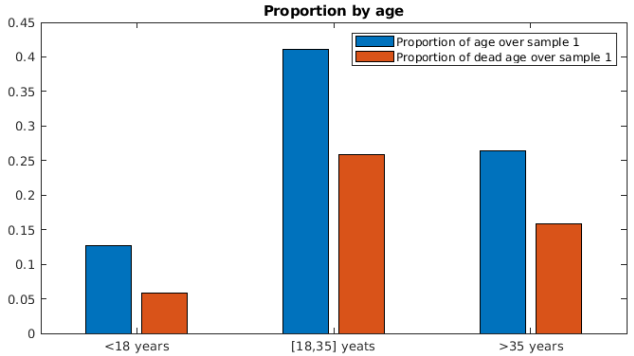
$$\boxed{P_{II} \pm \Delta P_{II} = 0.05 \pm 0.01}$$

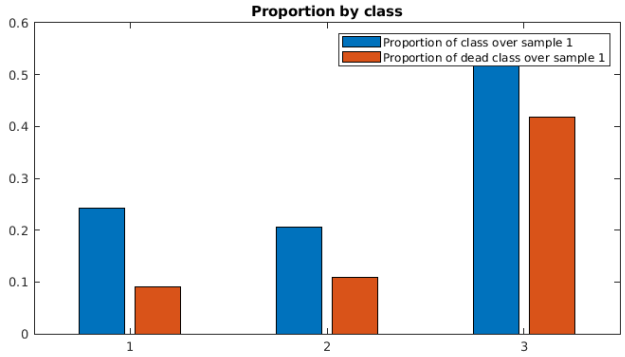
Notice that I got the uncertainty interval as the absolute error. Where I use the standar desviation of the population as estimation of the variance. Notice also that the distributions are symetrical and so the confidence interval.

←3.1









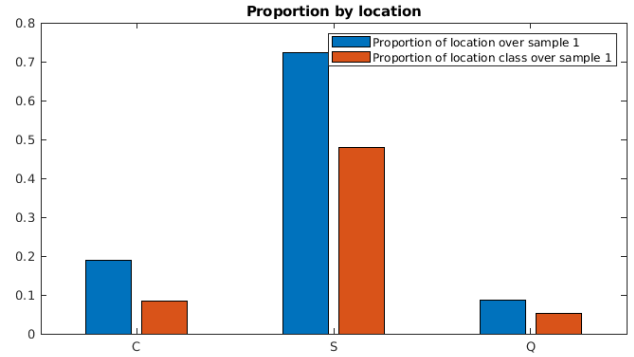


Figure 5: Graph analysis of proportions of sample I

←3.2

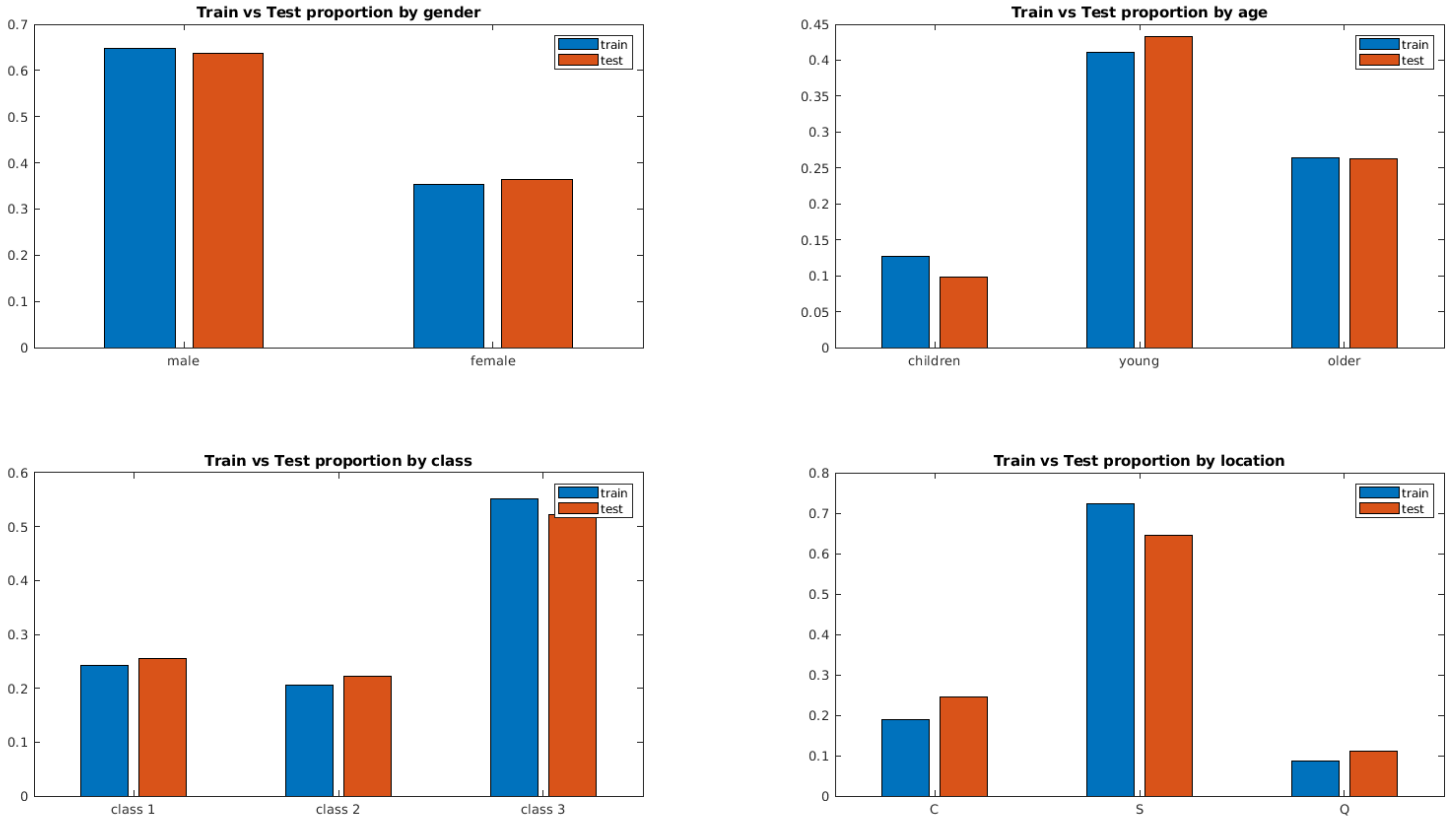


Figure 6: Graph analysis of proportions of sample I vs sample II

4 Matlab classification learner

In this section we are going to use Matlab clasification learner to train a model with 'train.cvs' and use diferent classification algorithms on 'test.csv'.

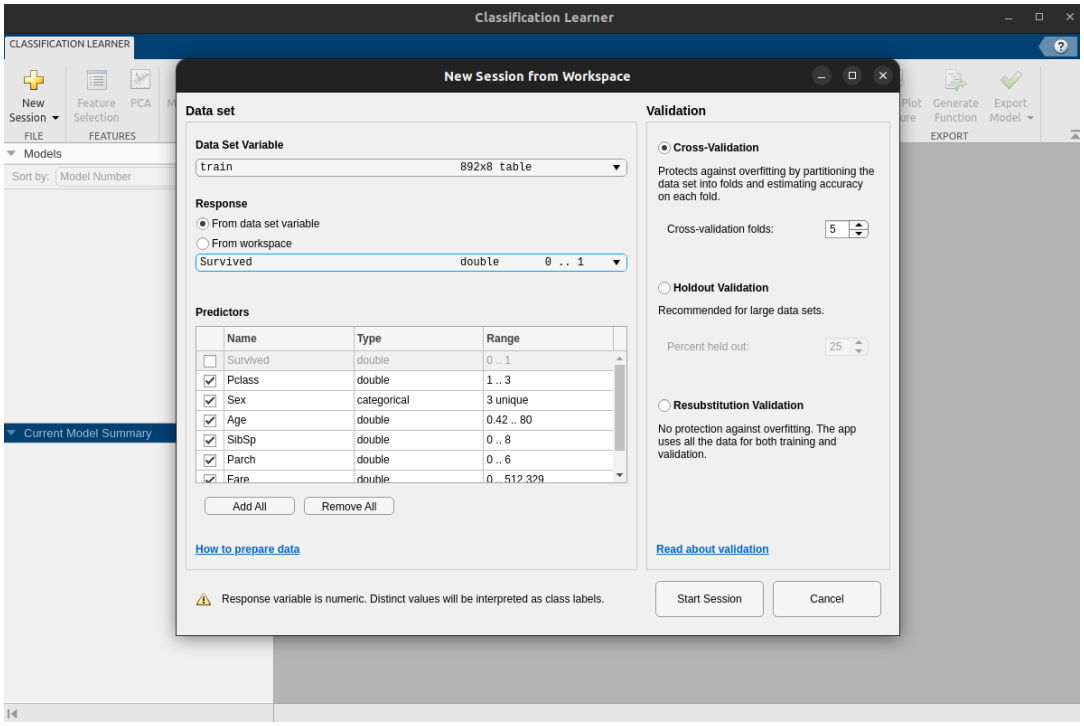
4.1 Input process

The predictors we are going to use are the following. Those as name, ticket number and cabin are not going to be taken in consideration because make more problems than usefull date to make the classification.

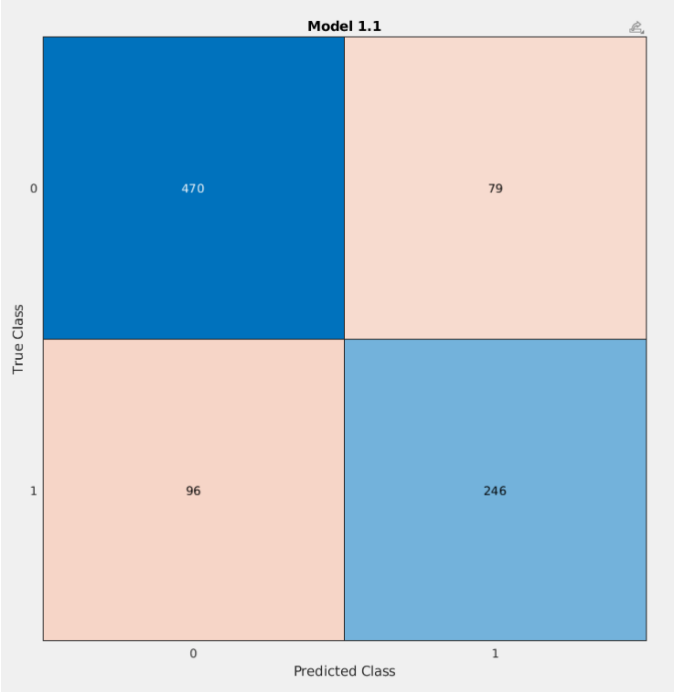
Class	Gender	Age	SibSp	Parch	Fare	Embarked
3	'male'	34.5	0	0	7.8292	'Q'
3	'female'	47	1	0	7	'S'
2	'male'	62	0	0	9.6875	'Q'
3	'male'	27	0	0	8.6625	'S'
3	'female'	22	1	1	12.2875	'S'
3	'male'	14	0	0	9.225	'S'
3	'female'	30	0	0	7.6292	'Q'
2	'male'	26	1	1	29	'S'
3	'female'	18	0	0	7.2292	'C'
↓	↓	↓	↓	↓	↓	↓

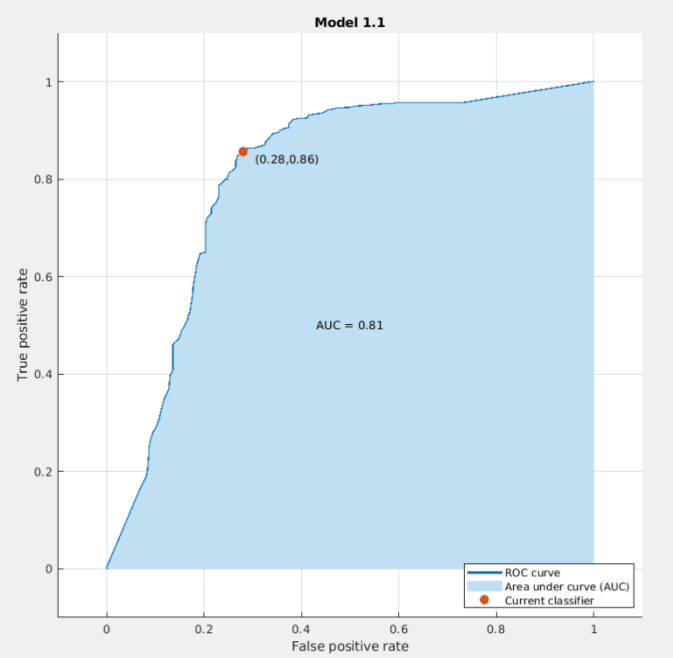
Table 1: Input data

We chose as response if survived or not and as predictors the ones we said. We also set Cross-Validation to 5. This means that we use 4/5 of data to train and 1/5 to validation, and then it makes permutations, to test the accuracy of the algorithms.



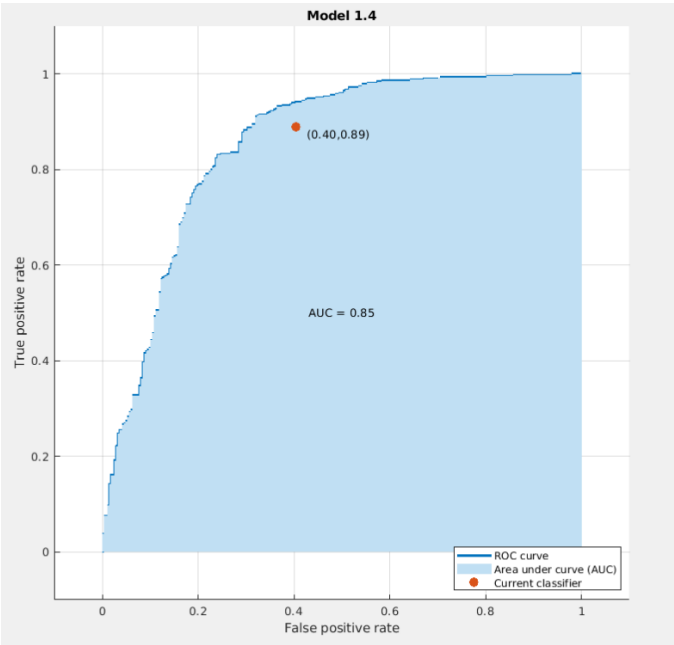
4.1.1 Tree Classification

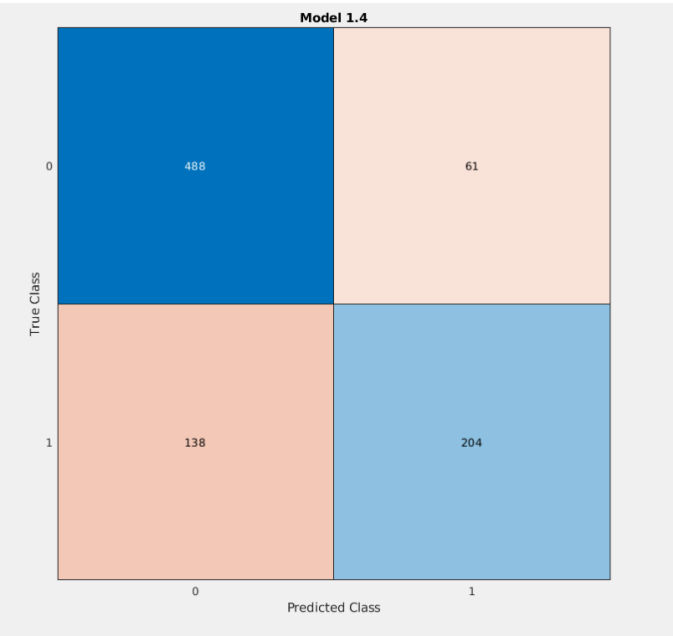




Model 1.1: Trained
Training Results Accuracy (Validation) 80.4
Model Type Preset: Fine Tree Maximum number of splits: 100 Split criterion: Gini's diversity index Surrogate decision splits: Off
Optimizer Options Hyperparameter options disabled
Feature Selection All features used in the model, before PCA
PCA PCA disabled
Misclassification Costs Cost matrix: default

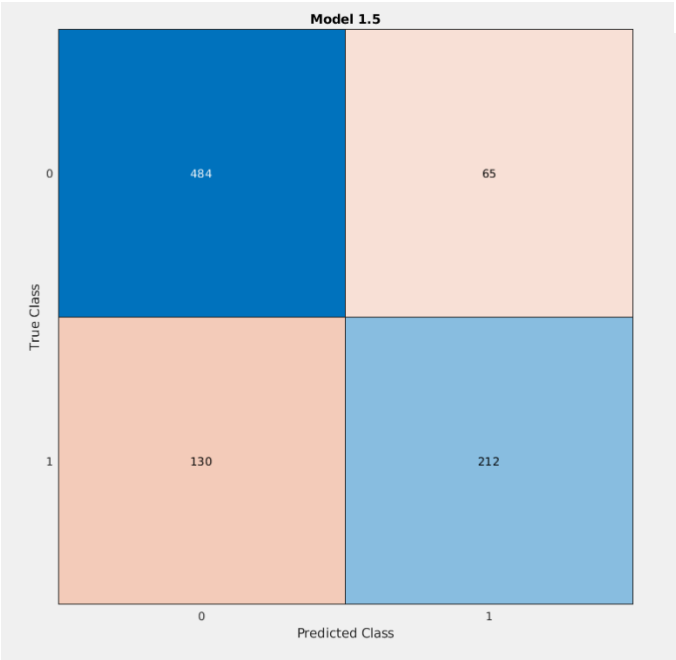
4.1.2 Logistic Regresion Classification

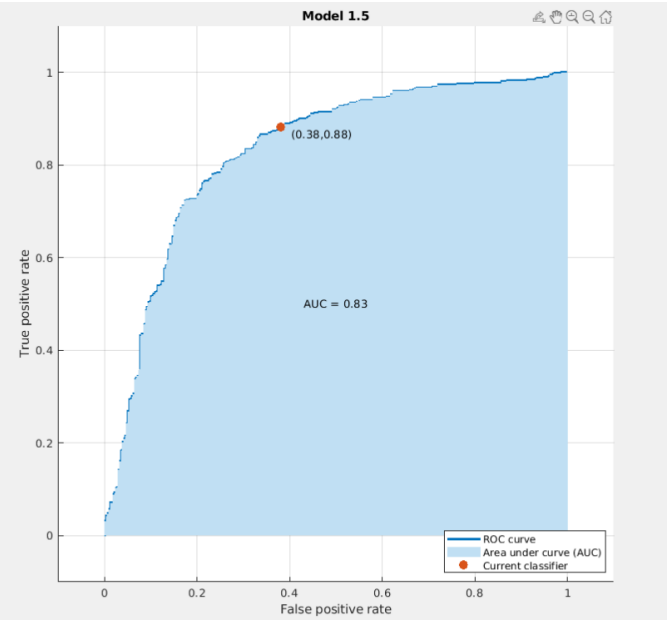




Model 1.4: Trained
Training Results Accuracy (Validation) 77.7 Total cost (Validation) Not applicable Prediction speed 5200 obs/sec Training time 3.5889 sec
Model Type Preset: Logistic Regression
Optimizer Options Hyperparameter options disabled
Feature Selection All features used in the model, before PCA
PCA PCA disabled
Misclassification Costs Not supported

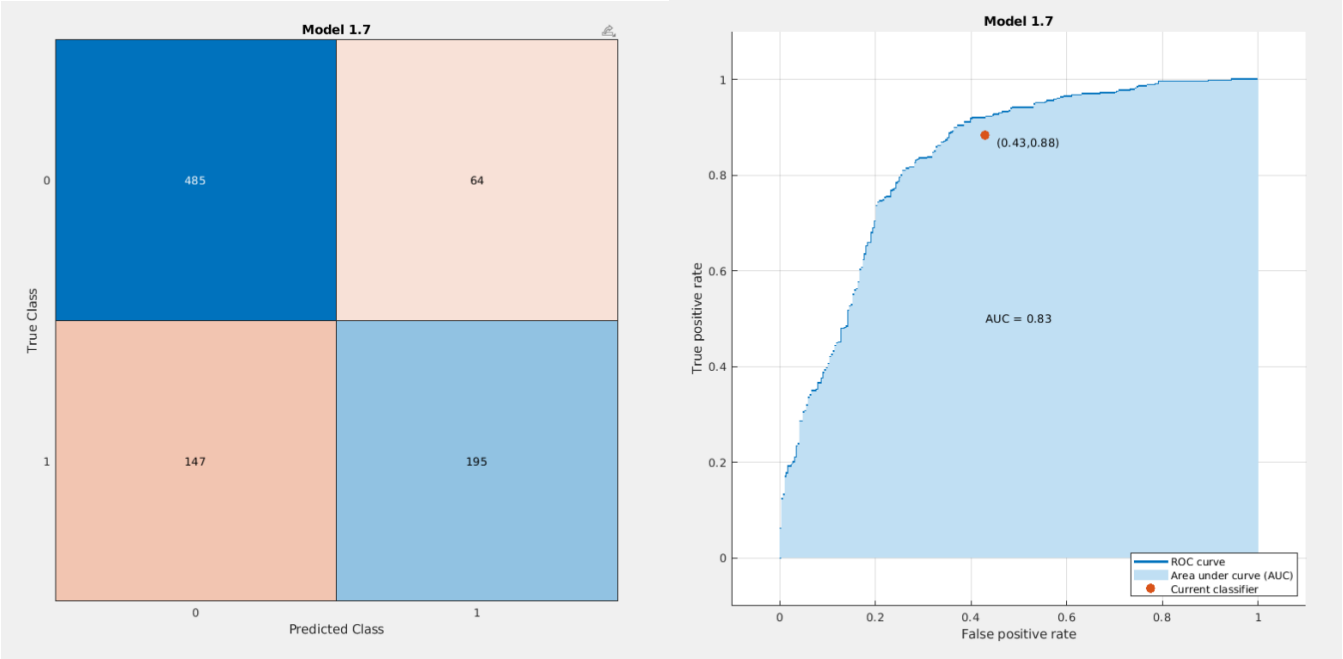
4.1.3 Naive Bayes Classification





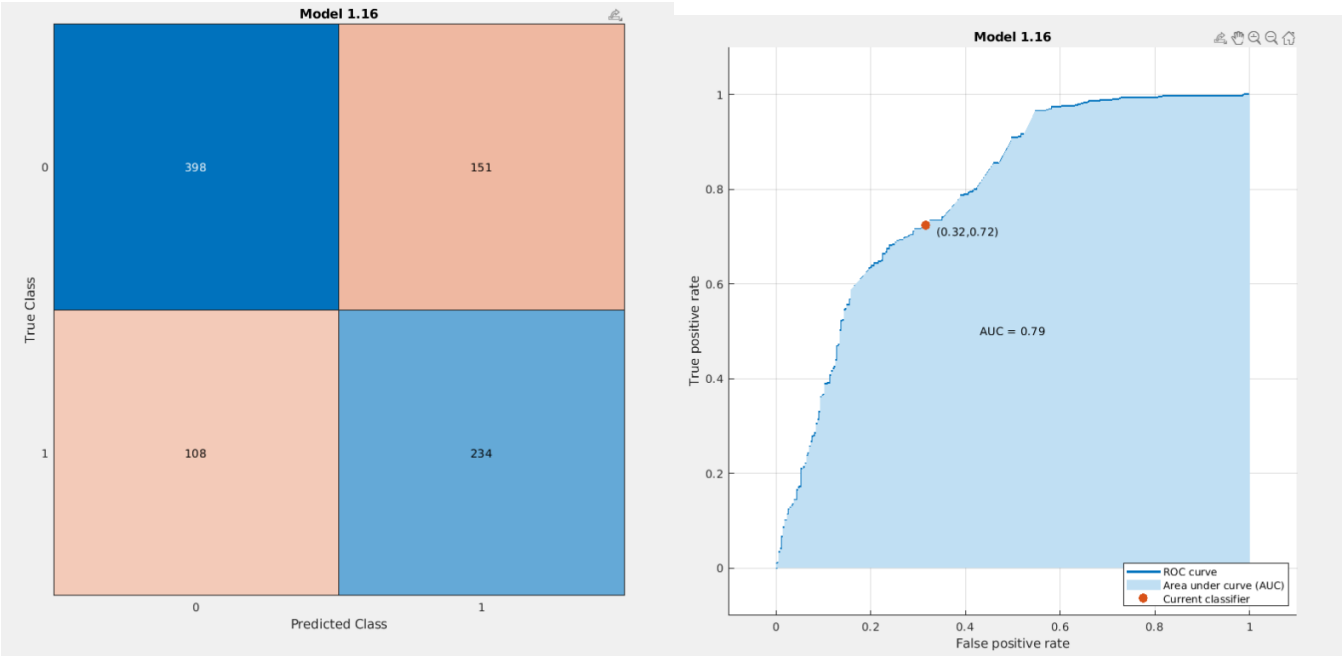
Model 1.5: Trained
Training Results Accuracy (Validation) 78.1
Total cost (Validation) 195 Prediction speed 19000 obs/sec Training time 1.2179 sec
Model Type Preset: Gaussian Naive Bayes Distribution name for numeric predictors: Gaussian Distribution name for categorical predictors: MVMN

4.1.4 Suported Vector Machine Classification



Model 1.7: Trained
Training Results Accuracy (Validation) 76.3
Total cost (Validation) 211 Prediction speed 23000 obs/sec Training time 1.7677 sec
Model Type Preset: Linear SVM Kernel function: Linear Kernel scale: Automatic

4.1.5 Neural Network Classification



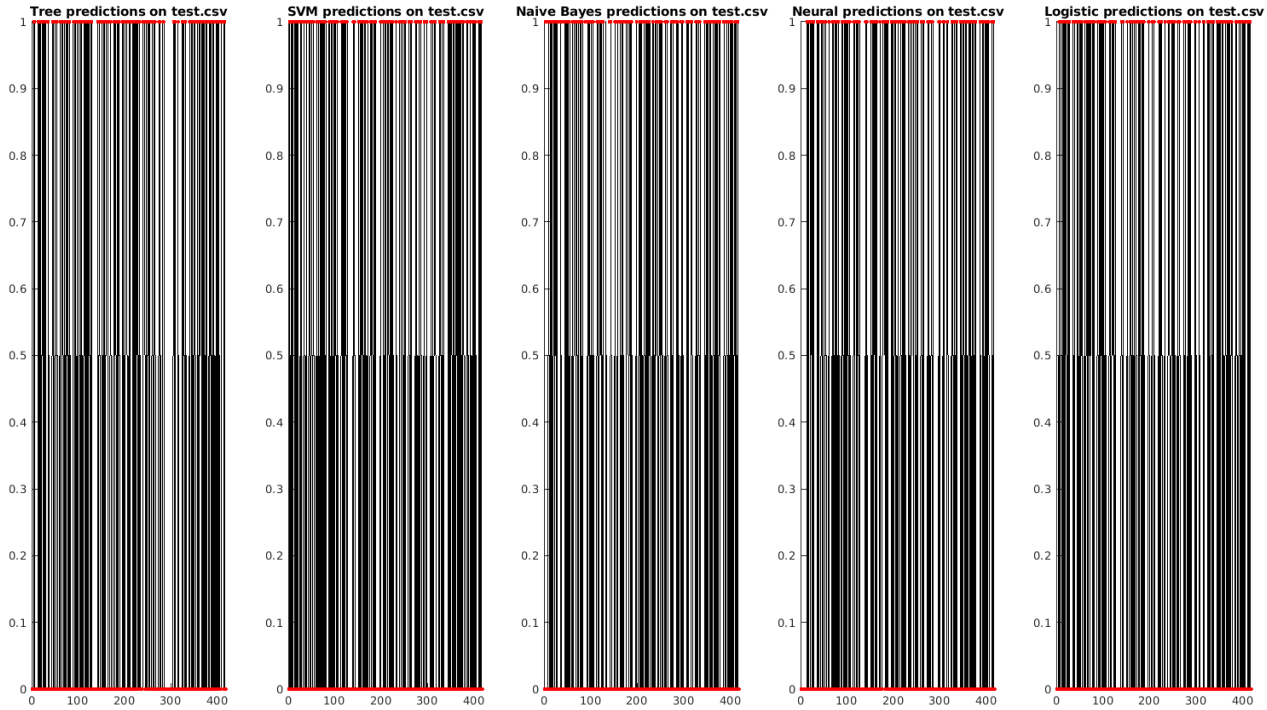
Model 1.7: Trained
Training Results Accuracy (Validation) 76.3
Total cost (Validation) 211 Prediction speed 23000 obs/sec Training time 1.7677 sec
Model Type Preset: Linear SVM Kernel function: Linear Kernel scale: Automatic

We get the following results for the proportions of survivers for 'test.csv'. **So we were wrong with our statistical model.**

	Tree	Logistic Regression	Naive Bayes	Linear Suported Machine	Neural Network
P	0,334	0,313	0,317	0,303	0,334

As we can see the algorithm of Tree Classification is the one who get best results in terms of Acuracy, if we don't want to classify someone as alive when he wasn't (False negative). we should choose the logistic regresion model.

In this figure we see the predictions on 'test.cvs' as bar code with points on one or cero over 418 points. Very visual way to see P as how much black we see.



5 Conclusion

In conclusion, it can be said that I was unable to obtain the proportion of victims a priori without even using the classification algorithms first. This was to be expected, as I have not found any estimators in the literature for the proportion in a random event that has only occurred once. However, I hope that the analysis will still be interesting. The part about using the MATLAB algorithms does not deserve much comment. They work well and are very fast.

6 Apendix

6.1 Matlab code for Descriptive analysis of sample I

```
clc; clear;
%titanic
%% Set up the Import Options and import the data
opts = delimitedTextImportOptions("NumVariables", 12);
% Specify range and delimiter
opts.DataLines = [2, Inf];
opts.Delimiter = ",";
% Specify column names and types
opts.VariableNames = ["PassengerId", "Survived", "Pclass", "Name", "Sex", "Age", "SibSp", "Parch", "Ticket", "Fare", "Cabin", "Embarked"];
opts.VariableTypes = ["double", "double", "double", "string", "categorical", "double", "double", "double", "double", "double", "string", "categorical"];
% Specify file level properties
opts.ExtraColumnsRule = "ignore";
opts.EmptyLineRule = "read";
% Specify variable properties
opts = setvaropts(opts, ["Name", "Cabin"], "WhitespaceRule", "preserve");
opts = setvaropts(opts, ["Name", "Sex", "Cabin", "Embarked"], "EmptyFieldRule", "auto");
% Import the data
train = readtable("/home/jesus/Desktop/titanic pattern recognition/titanic/train.csv", opts);
%% Data analysis
close all;
%NUMBER OF SURVIVAL
I=transpose(table2array(train(:,2)));
%% SEX FEMALE=1 MALE=0
S=transpose(table2array(train(:,5)));
S=double(S);
F=find(S==1);%MUJERERS
M=find(S==2);%HOMBRES
PF=length(F)/length(S);%Proportion of women
PM=length(M)/length(S);%Proportion of male
FD=find(S==1 & I==0);
MD=find(S==2 & I==0);
PFD=length(FD)/length(S);%Proportion of female dead
PMD=length(MD)/length(S);%proportion of male dead
%% AGE
A=transpose(table2array(train(:,6)));
AC=find(A<18);
AY=find(A>17.5 & A<35);
AO=find(A>34.5);
PC=length(AC)/length(A); %Proportion of children and teenagers
PY=length(AY)/length(A);%Proportion of young
PO=length(AO)/length(A);%Proportion of old people
ACD=find(A<18 & I==0);
AYD=find(A>17.5 & A<35 & I==0);
AOD=find(A>34.5 & I==0 );
PCD=length(ACD)/length(A); %Proportion of children and teenagers
PYD=length(AYD)/length(A);%Proportion of young
POD=length(AOD)/length(A);%Proportion of old people
%% Class
C=transpose(table2array(train(:,3)));
C1cnt=length(find(C==1));
C2cnt=length(find(C==2));
C3cnt=length(find(C==3));
Ccnt=length(C)-(C1cnt+C2cnt+C3cnt);
P1=C1cnt/length(C); %Proportion of people in class 1
P2=C2cnt/length(C); %Proportion of people in class 2
P3=C3cnt/length(C); %Proportion of people in class 3
C1cntD=length(find(C==1 & I==0));
C2cntD=length(find(C==2 & I==0));
C3cntD=length(find(C==3 & I==0));
P1D=C1cntD/length(C); %Proportion of people in class 1
P2D=C2cntD/length(C); %Proportion of people in class 2
P3D=C3cntD/length(C); %Proportion of people in class 3
%% Location
L=double(transpose(table2array(train(:,12))));
PL1=length(find(L==1))/length(L);%proportion of people in C
PL2=length(find(L==3))/length(L);%proportion of people in S
PL3=length(find(L==2))/length(L);%proportion of people in Q
PL1D=length(find(L==1 & I==0))/length(L);%proportion of people in C
PL2D=length(find(L==3 & I==0))/length(L);%proportion of people in S
PL3D=length(find(L==2 & I==0))/length(L);%proportion of people in Q
%% plots
f0=figure;
f0.Color='white';
Icnt=sum(I)/length(I);
bar([1-Icnt;0.675])
title('Proportion of death')
f=figure;
f.Color='white';
subplot(2,2,1)
bar([1,2],[PM,PMD;PF,PFD])
title('Proportion by gender')
legend('Proportion of gender over sample 1','Proportion of dead gender over sample 1');
subplot(2,2,2)
bar([0,1,2],[PC,PCD;PY,PYD;PO,POD])
title('Proportion by age')
legend('Proportion of age over sample 1','Proportion of dead age over sample 1');
subplot(2,2,3)
bar([1,2,3],[P1,P1D;P2,P2D;P3,P3D])
title('Proportion by class');
legend('Proportion of class over sample 1','Proportion of dead class over sample 1');
subplot(2,2,4)
bar([0,1,2],[PL1,PL1D;PL2,PL2D;PL3,PL3D])
title('Proportion by location');
legend('Proportion of location over sample 1','Proportion of location class over sample 1');
```

6.2 Matlab code for Descriptive analysis of sample I vs sample II

```
clc; clear;
%% read data from train.csv

opts = delimitedTextImportOptions("NumVariables", 12);

% Specify range and delimiter
opts.DataLines = [2, Inf];
opts.Delimiter = ",";

% Specify column names and types
opts.VariableNames = ["PassengerId", "Survived", "Pclass", "Name", "Sex", "Age", "SibSp", "Parch", "Ticket", "Fare", "Cabin", "Embarked"];
opts.VariableTypes = ["double", "double", "double", "string", "categorical", "double", "double", "double", "double", "double", "string", "categorical"];

% Specify file level properties
opts.ExtraColumnsRule = "ignore";
opts.EmptyLineRule = "read";

% Specify variable properties
opts = setvaropts(opts, ["Name", "Cabin"], "WhitespaceRule", "preserve");
opts = setvaropts(opts, ["Name", "Sex", "Cabin", "Embarked"], "EmptyFieldRule", "auto");

% Import the data
train = readtable("/home/jesus/Desktop/titanic pattern recognition/titanic/train.csv", opts);

clear opts
```

```
%% read table from test.csv

opts = delimitedTextImportOptions("NumVariables", 11);

% Specify range and delimiter
opts.DataLines = [2, Inf];
opts.Delimiter = ",";

% Specify column names and types
opts.VariableNames = ["PassengerId", "Pclass", "Name", "Sex", "Age", "SibSp", "Parch", "Ticket", "Fare", "Cabin", "Embarked"];
opts.VariableTypes = ["double", "double", "string", "categorical", "double", "double", "double", "double", "double", "string", "categorical"];

% Specify file level properties
opts.ExtraColumnsRule = "ignore";
opts.EmptyLineRule = "read";

% Specify variable properties
opts = setvaropts(opts, ["Name", "Cabin"], "WhitespaceRule", "preserve");
opts = setvaropts(opts, ["Name", "Sex", "Cabin", "Embarked"], "EmptyFieldRule", "auto");

% Import the data
test = readtable("/home/jesus/Desktop/titanic pattern recognition/titanic/test.csv", opts);

clear opts
```

```
%% Proportion by gender
Strain=transpose(table2array(train(:,5)));
Stest=transpose(table2array(test(:,4)));
Strain=double(Strain);
Stest=double(Stest);
Ftrain=find(Strain==1);%WOMEN
Mtrain=find(Strain==2);%MAN
Ftest=find(Stest==1);%WOMEN
Mtest=find(Stest==2);%MAN
PFtrain=length(Ftrain)/length(Strain);%Proportion of women
PMtrain=length(Mtrain)/length(Strain);%Proportion of male
PFtest=length(Ftest)/length(Stest);%Proportion of women
PMtest=length(Mtest)/length(Stest);%Proportion of male
```

```
%% Proportion by age
Atrain=transpose(table2array(train(:,6)));
Atest=transpose(table2array(test(:,5)));
ACtrain=find(Atrain<18);
AYtrain=find(Atrain>17.5 & Atrain<35);
AOtrain=find(Atrain>34.5);

ACtest=find(Atest<18);
AYtest=find(Atest>17.5 & Atest<35);
AOtest=find(Atest>34.5);

PCtrain=length(ACtrain)/length(Atrain); %Proportion of children and teenagers
PYtrain=length(AYtrain)/length(Atrain);%Proportion of young
POtrain=length(AOtrain)/length(Atrain);%Proportion of old people

PCtest=length(ACtest)/length(Atest); %Proportion of children and teenagers
PYtest=length(AYtest)/length(Atest);%Proportion of young
POtest=length(AOtest)/length(Atest);%Proportion of old people
```

```
%% Location
Ltrain=double(transpose(table2array(train(:,12))));
Ltest=double(transpose(table2array(test(:,11))));

PL1train=length(find(Ltrain==1))/length(Ltrain);%proportion of people in C
PL2train=length(find(Ltrain==3))/length(Ltrain);%proportion of people in S
PL3train=length(find(Ltrain==2))/length(Ltrain);%proportion of people in Q

PL1test=length(find(Ltest==1))/length(Ltest);%proportion of people in C
PL2test=length(find(Ltest==3))/length(Ltest);%proportion of people in S
PL3test=length(find(Ltest==2))/length(Ltest);%proportion of people in Q
```

```
%% plot
f=figure;
f.Color='white';
subplot(2,2,1)
bar([1,2],[PMtrain,PMtest;PFtrain,PFtest;]);
name = {'male';'female'};
title('Train vs Test proportion by gender')
set(gca,'xticklabel',name)
legend('train','test');
subplot(2,2,2)
bar([1,2,3],[PCtrain,PCtest;PYtrain,PYtest;POtrain,POtest;]);
title('Train vs Test proportion by age')
name = {'children';'young';'older'};
set(gca,'xticklabel',name)
legend('train','test');
subplot(2,2,3)
bar([1,2,3],[P1train,P1test;P2train,P2test;P3train,P3test;]);
title('Train vs Test proportion by class')
name = {'class 1';'class 2';'class 3'};
set(gca,'xticklabel',name)
legend('train','test');
subplot(2,2,4)
bar([1,2,3],[PL1train,PL1test;PL2train,PL2test;PL3train,PL3test;]);
title('Train vs Test proportion by location');
legend('train','test');
name = {'C';'S';'Q'};
set(gca,'xticklabel',name)
```

References

[1] Teoría de las probabilidades y estadística matemática. V.E.GMURMAN. MIR MOSCU

[2] https://towardsdatascience.com/
kaggle-predict-survival-on-the-titanic-challenge-in-matlab-56f6ad3bab78

[3] https://ww2.mathworks.cn/help/stats/classificationlearner-app.html?requestedDomain=cn