

Titanic - Machine Learning from Disaster

Copyright @ Nabajeet Barman (nabajeetbarman4@gmail.com)

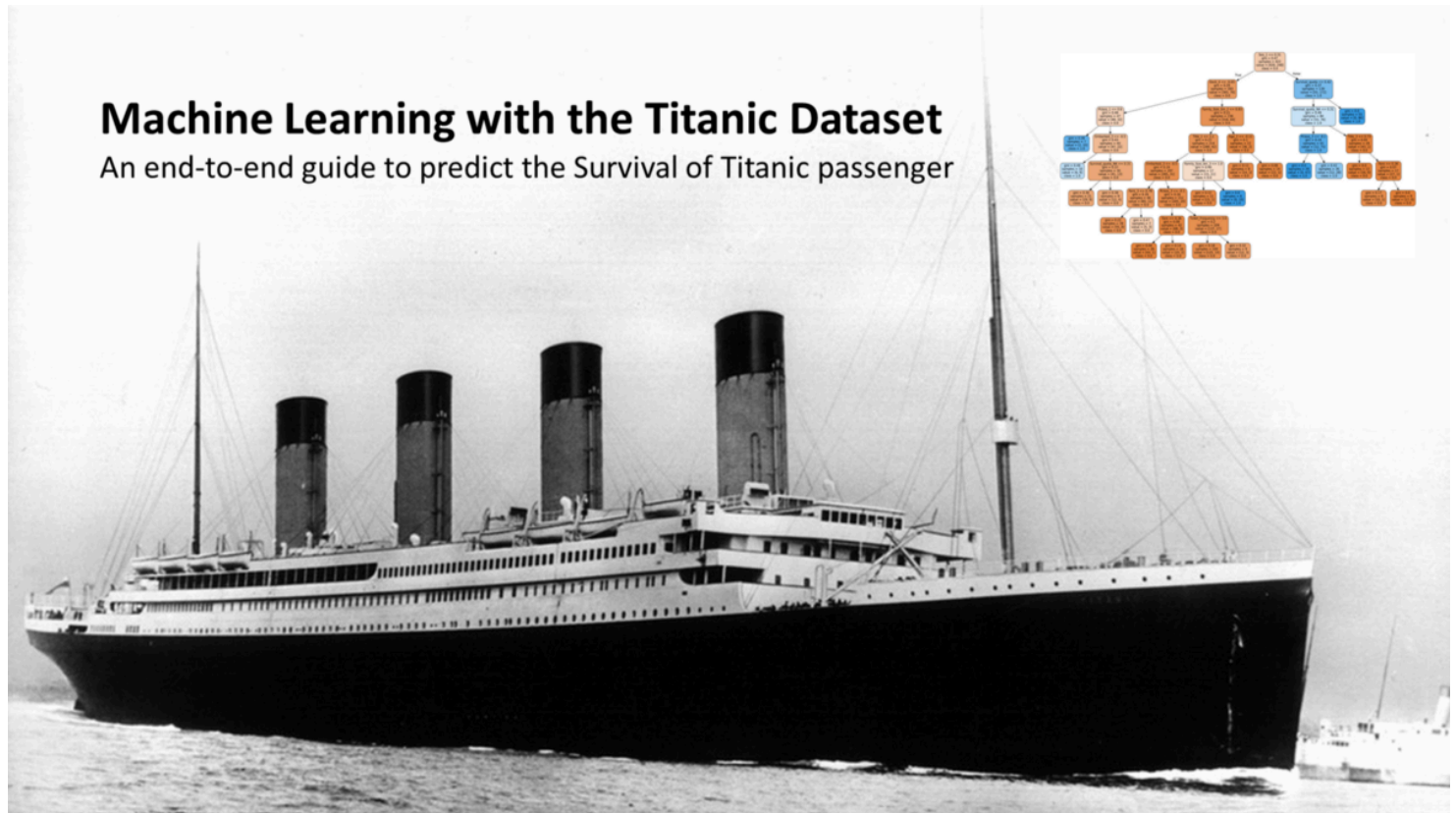


Image Courtesy

#####

Titanic dataset, originally posted on Kaggle is the first dataset that is used by any Data Scientist. Over the years, it has been the de-facto dataset for DS over the past several years due to its simplicity.

You can read more about the dataset [here](#)

Table of Contents

Titanic - Machine Learning from Disaster.....	1
Importing the Dataset.....	2
Setup the Import Options and import the data.....	2
Clear temporary variables.....	3
Understanding the Titanic Dataset.....	3
Save the excel file for sharing or off-Matlab use.....	4
Preliminary Data Analysis and Preprocessing.....	9
Printing Summary Stats.....	9
Handling Missing Data.....	11
Using "Task" control functionality.....	11
Doing programmatically by writing code.....	13
Removing Columns.....	18
Handling Outliers.....	20
Using "Task" control functionality.....	20
Doing programmatically by writing code.....	21

Visualization of Titanic Dataset.....	23
Chart 01: Survival count by passenger count.....	23
Chart 02: Passenger count grouped by Passenger Class.....	24
Chart 03: Survival Count by Passenger class.....	25
Chart 04: Plotting scatter plot of Age grouped by Survival	26
Chart 05: Plotting scatter plot of Age grouped by Pclass	27
Chart 06: Stacked Plot of all column variables.....	28
Machine Learning with Titanic Dataset - Predicting survival on the Titanic	29
Classification.....	30
Evaluating a classification model.....	30
Classification Evaluation Criteria.....	31
Confusion Matrix:	31
Supervised Classification Using the Classification App.....	32
Logistic Regression.....	32
(Fine) Decision Tree.....	34
Solution??.....	36
What is the Problem with Categorical Data?.....	36
Integer Encoding.....	37
One-hot encoding!!!!.....	37
When to use a Label Encoding vs. One Hot Encoding?.....	37
Supervised Classification Using the Classification Algorithms (by coding).....	39
Model 01: Logistic Regression.....	39
Model 02: Decision Trees.....	40
How to choose the right algorithm?.....	41

Importing the Dataset

Download the dataset file, titanic.csv file from [Box](#)

Setup the Import Options and import the data

```
clear;
close all;
clc;

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", "Paro
opts.VariableTypes = ["double", "categorical", "categorical", "string", "categorical", "double'

% 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
```

```
titanic = readtable("C:\Users\Nabajeet Barman\Dropbox\MA6600\Workshop02-25Feb\titanic.csv", opt
```

```
titanic = 891x12 table
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp
1	1	0	3	"Braund, ...	male	22	1
2	2	1	1	"Cumings,...	female	38	1
3	3	1	3	"Heikkine...	female	26	0
4	4	1	1	"Futrelle...	female	35	1
5	5	0	3	"Allen, M...	male	35	0
6	6	0	3	"Moran, M...	male	NaN	0
7	7	0	1	"McCarthy...	male	54	0
8	8	0	3	"Palsson,...	male	2	3
9	9	1	3	"Johnson,...	female	27	0
10	10	1	2	"Nasser, ...	female	14	1
11	11	1	3	"Sandstro...	female	4	1
12	12	1	1	"Bonnell,...	female	58	0
13	13	0	3	"Saunders...	male	20	0
14	14	0	3	"Andersso...	male	39	1

⋮

Clear temporary variables

```
% to use them later, i have commented this out
% clear opts
```

To check the datatype of the table column!!

```
class(titanic.Pclass)
```

```
ans =
'categorical'
```

Understanding the Titanic Dataset

The Titanic data set consists of the following data columns:

- **PassengerId:** Id of every passenger.
- **Survived:** This feature have value 0 and 1. **0 for not survived** and **1 for survived**.
- **Pclass:** There are 3 classes: Class 1, Class 2 and Class 3.
- **Name:** Name of passenger.
- **Sex:** Gender of passenger.
- **Age:** Age of passenger.

- **SibSp**: Indication that passenger have siblings and spouse.
- **Parch**: Whether a passenger is alone or have family.
- **Ticket**: Ticket number of passenger.
- **Fare**: Indicating the fare.
- **Cabin**: The cabin of passenger.
- **Embarked**: The embarked category.

Notes:

pclass: A proxy for socio-economic status (SES)

- 1st = Upper
- 2nd = Middle
- 3rd = Lower

sibsp: The dataset defines family relations in this way...

- Sibling = brother, sister, stepbrother, stepsister
- Spouse = husband, wife (mistresses and fiancés were ignored)

parch: The dataset defines family relations in this way...

- Parent = mother, father
- Child = daughter, son, stepdaughter, stepson
- Some children travelled only with a nanny, therefore parch=0 for them.

The 'Survived' column is our binary *target* variable which we need to predict; where 0- Not Survived, 1- Survived.

Save the excel file for sharing or off-Matlab use

```
writetable(titanic, 'titanic_modified.csv')
% save the file after done with the basic pre-processing steps, so that you
% do not need to run that part of the code everytime.
```

Previewing the file at any stage

```
titanic(1:5,:)
```

ans = 5x12 table

...

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp
1	1	0	3	"Braund, ...	male	22	1
2	2	1	1	"Cumings,...	female	38	1
3	3	1	3	"Heikkine...	female	26	0
4	4	1	1	"Futrelle...	female	35	1

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp
5	5	0	3	"Allen, M...	male	35	0

or alternatively

```
disp(titanic)
```

PassengerId	Survived	Pclass	Name
1	0	3	"Braund, Mr. Owen Harris"
2	1	1	"Cumings, Mrs. John Bradley (Florence Briggs Thayer)"
3	1	3	"Heikkinen, Miss. Laina"
4	1	1	"Futrelle, Mrs. Jacques Heath (Lily May Peel)"
5	0	3	"Allen, Mr. William Henry"
6	0	3	"Moran, Mr. James"
7	0	1	"McCarthy, Mr. Timothy J"
8	0	3	"Palsson, Master. Gosta Leonard"
9	1	3	"Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)"
10	1	2	"Nasser, Mrs. Nicholas (Adele Achem)"
11	1	3	"Sandstrom, Miss. Marguerite Rut"
12	1	1	"Bonnell, Miss. Elizabeth"
13	0	3	"Saunderscock, Mr. William Henry"
14	0	3	"Andersson, Mr. Anders Johan"
15	0	3	"Vestrom, Miss. Hulda Amanda Adolfina"
16	1	2	"Hewlett, Mrs. (Mary D Kingcome) "
17	0	3	"Rice, Master. Eugene"
18	1	2	"Williams, Mr. Charles Eugene"
19	0	3	"Vander Planke, Mrs. Julius (Emelia Maria Vandemoortele)"
20	1	3	"Masselmani, Mrs. Fatima"
21	0	2	"Fynney, Mr. Joseph J"
22	1	2	"Beesley, Mr. Lawrence"
23	1	3	"McGowan, Miss. Anna "Annie" "
24	1	1	"Sloper, Mr. William Thompson"
25	0	3	"Palsson, Miss. Torborg Danira"
26	1	3	"Asplund, Mrs. Carl Oscar (Selma Augusta Emilia Johansson)"
27	0	3	"Emir, Mr. Farred Chehab"
28	0	1	"Fortune, Mr. Charles Alexander"
29	1	3	"O'Dwyer, Miss. Ellen "Nellie" "
30	0	3	"Todoroff, Mr. Lalio"
31	0	1	"Uruchurtu, Don. Manuel E"
32	1	1	"Spencer, Mrs. William Augustus (Marie Eugenie)"
33	1	3	"Glynn, Miss. Mary Agatha"
34	0	2	"Wheadon, Mr. Edward H"
35	0	1	"Meyer, Mr. Edgar Joseph"
36	0	1	"Holverson, Mr. Alexander Oskar"
37	1	3	"Mamee, Mr. Hanna"
38	0	3	"Cann, Mr. Ernest Charles"
39	0	3	"Vander Planke, Miss. Augusta Maria"
40	1	3	"Nicola-Yarred, Miss. Jamila"
41	0	3	"Ahlin, Mrs. Johan (Johanna Persdotter Larsson)"
42	0	2	"Turpin, Mrs. William John Robert (Dorothy Ann Wonnacott)"
43	0	3	"Kraeff, Mr. Theodor"
44	1	2	"Laroche, Miss. Simonne Marie Anne Andree"
45	1	3	"Devaney, Miss. Margaret Delia"
46	0	3	"Rogers, Mr. William John"
47	0	3	"Lennon, Mr. Denis"
48	1	3	"O'Driscoll, Miss. Bridget"
49	0	3	"Samaan, Mr. Youssef"
50	0	3	"Arnold-Franchi, Mrs. Josef (Josefine Franchi)"
51	0	3	"Panula, Master. Juha Niilo"
52	0	3	"Nosworthy, Mr. Richard Cater"
53	1	1	"Harper, Mrs. Henry Sleeper (Myna Haxtun)"

54	1	2	"Faunthorpe, Mrs. Lizzie (Elizabeth Anne Wilkinson)"
55	0	1	"Ostby, Mr. Engelhart Cornelius"
56	1	1	"Woolner, Mr. Hugh"
57	1	2	"Rugg, Miss. Emily"
58	0	3	"Novel, Mr. Mansouer"
59	1	2	"West, Miss. Constance Mirium"
60	0	3	"Goodwin, Master. William Frederick"
61	0	3	"Sirayanian, Mr. Orsen"
62	1	1	"Icard, Miss. Amelie"
63	0	1	"Harris, Mr. Henry Birkhardt"
64	0	3	"Skoog, Master. Harald"
65	0	1	"Stewart, Mr. Albert A"
66	1	3	"Moubarek, Master. Gerios"
67	1	2	"Nye, Mrs. (Elizabeth Ramell)"
68	0	3	"Crease, Mr. Ernest James"
69	1	3	"Andersson, Miss. Erna Alexandra"
70	0	3	"Kink, Mr. Vincenz"
71	0	2	"Jenkin, Mr. Stephen Curnow"
72	0	3	"Goodwin, Miss. Lillian Amy"
73	0	2	"Hood, Mr. Ambrose Jr"
74	0	3	"Chronopoulos, Mr. Apostolos"
75	1	3	"Bing, Mr. Lee"
76	0	3	"Moen, Mr. Sigurd Hansen"
77	0	3	"Staneff, Mr. Ivan"
78	0	3	"Moutal, Mr. Rahamin Haim"
79	1	2	"Caldwell, Master. Alden Gates"
80	1	3	"Dowdell, Miss. Elizabeth"
81	0	3	"Waelens, Mr. Achille"
82	1	3	"Sheerlinck, Mr. Jan Baptist"
83	1	3	"McDermott, Miss. Brigdet Delia"
84	0	1	"Carrau, Mr. Francisco M"
85	1	2	"Ilett, Miss. Bertha"
86	1	3	"Backstrom, Mrs. Karl Alfred (Maria Mathilda Gustafsson)"
87	0	3	"Ford, Mr. William Neal"
88	0	3	"Slocovski, Mr. Selman Francis"
89	1	1	"Fortune, Miss. Mabel Helen"
90	0	3	"Celotti, Mr. Francesco"
91	0	3	"Christmann, Mr. Emil"
92	0	3	"Andreasson, Mr. Paul Edvin"
93	0	1	"Chaffee, Mr. Herbert Fuller"
94	0	3	"Dean, Mr. Bertram Frank"
95	0	3	"Coxon, Mr. Daniel"
96	0	3	"Shorney, Mr. Charles Joseph"
97	0	1	"Goldschmidt, Mr. George B"
98	1	1	"Greenfield, Mr. William Bertram"
99	1	2	"Doling, Mrs. John T (Ada Julia Bone)"
100	0	2	"Kantor, Mr. Sinai"
101	0	3	"Petraneck, Miss. Matilda"
102	0	3	"Petroff, Mr. Pastcho ("Pentcho")"
103	0	1	"White, Mr. Richard Frasar"
104	0	3	"Johansson, Mr. Gustaf Joel"
105	0	3	"Gustafsson, Mr. Anders Vilhelm"
106	0	3	"Mionoff, Mr. Stoytcho"
107	1	3	"Salkjelsvik, Miss. Anna Kristine"
108	1	3	"Moss, Mr. Albert Johan"
109	0	3	"Rekic, Mr. Tido"
110	1	3	"Moran, Miss. Bertha"
111	0	1	"Porter, Mr. Walter Chamberlain"
112	0	3	"Zabour, Miss. Hileni"
113	0	3	"Barton, Mr. David John"
114	0	3	"Jussila, Miss. Katriina"
115	0	3	"Attalah, Miss. Malake"
116	0	3	"Pekoniemi, Mr. Edvard"
117	0	3	"Connors, Mr. Patrick"
118	0	2	"Turpin, Mr. William John Robert"

119	0	1	"Baxter, Mr. Quigg Edmond"
120	0	3	"Andersson, Miss. Ellis Anna Maria"
121	0	2	"Hickman, Mr. Stanley George"
122	0	3	"Moore, Mr. Leonard Charles"
123	0	2	"Nasser, Mr. Nicholas"
124	1	2	"Webber, Miss. Susan"
125	0	1	"White, Mr. Percival Wayland"
126	1	3	"Nicola-Yarred, Master. Elias"
127	0	3	"McMahon, Mr. Martin"
128	1	3	"Madsen, Mr. Fridtjof Arne"
129	1	3	"Peter, Miss. Anna"
130	0	3	"Ekstrom, Mr. Johan"
131	0	3	"Drazenoic, Mr. Jozef"
132	0	3	"Coelho, Mr. Domingos Fernando"
133	0	3	"Robins, Mrs. Alexander A (Grace Charity Laury)"
134	1	2	"Weisz, Mrs. Leopold (Mathilde Francoise Pede)"
135	0	2	"Sobey, Mr. Samuel James Hayden"
136	0	2	"Richard, Mr. Emile"
137	1	1	"Newsom, Miss. Helen Monypeny"
138	0	1	"Futrelle, Mr. Jacques Heath"
139	0	3	"Osen, Mr. Olaf Elon"
140	0	1	"Giglio, Mr. Victor"
141	0	3	"Boulos, Mrs. Joseph (Sultana)"
142	1	3	"Nysten, Miss. Anna Sofia"
143	1	3	"Hakkarainen, Mrs. Pekka Pietari (Elin Matilda Dolck)"
144	0	3	"Burke, Mr. Jeremiah"
145	0	2	"Andrew, Mr. Edgardo Samuel"
146	0	2	"Nicholls, Mr. Joseph Charles"
147	1	3	"Andersson, Mr. August Edvard ("Wennerstrom")"
148	0	3	"Ford, Miss. Robina Maggie "Ruby""
149	0	2	"Navratil, Mr. Michel ("Louis M Hoffman")"
150	0	2	"Byles, Rev. Thomas Roussel Davids"
151	0	2	"Bateman, Rev. Robert James"
152	1	1	"Pears, Mrs. Thomas (Edith Wearne)"
153	0	3	"Meo, Mr. Alfonzo"
154	0	3	"van Billiard, Mr. Austin Blyler"
155	0	3	"Olsen, Mr. Ole Martin"
156	0	1	"Williams, Mr. Charles Duane"
157	1	3	"Gilnagh, Miss. Katherine "Katie""
158	0	3	"Corn, Mr. Harry"
159	0	3	"Smiljanic, Mr. Mile"
160	0	3	"Sage, Master. Thomas Henry"
161	0	3	"Cribb, Mr. John Hatfield"
162	1	2	"Watt, Mrs. James (Elizabeth "Bessie" Inglis Milne)"
163	0	3	"Bengtsson, Mr. John Viktor"
164	0	3	"Calic, Mr. Jovo"
165	0	3	"Panula, Master. Eino Viljami"
166	1	3	"Goldsmith, Master. Frank John William "Frankie""
167	1	1	"Chibnall, Mrs. (Edith Martha Bowerman)"
168	0	3	"Skoog, Mrs. William (Anna Bernhardina Karlsson)"
169	0	1	"Baumann, Mr. John D"
170	0	3	"Ling, Mr. Lee"
171	0	1	"Van der hoef, Mr. Wyckoff"
172	0	3	"Rice, Master. Arthur"
173	1	3	"Johnson, Miss. Eleanor Ileen"
174	0	3	"Sivola, Mr. Antti Wilhelm"
175	0	1	"Smith, Mr. James Clinch"
176	0	3	"Klasen, Mr. Klas Albin"
177	0	3	"Lefebvre, Master. Henry Forbes"
178	0	1	"Isham, Miss. Ann Elizabeth"
179	0	2	"Hale, Mr. Reginald"
180	0	3	"Leonard, Mr. Lionel"
181	0	3	"Sage, Miss. Constance Gladys"
182	0	2	"Pernot, Mr. Rene"
183	0	3	"Asplund, Master. Clarence Gustaf Hugo"

184	1	2	"Becker, Master. Richard F"
185	1	3	"Kink-Heilmann, Miss. Luise Gretchen"
186	0	1	"Rood, Mr. Hugh Roscoe"
187	1	3	"O'Brien, Mrs. Thomas (Johanna "Hannah" Godfrey)"
188	1	1	"Romaine, Mr. Charles Hallace ("Mr C RoImane")"
189	0	3	"Bourke, Mr. John"
190	0	3	"Turcin, Mr. Stjepan"
191	1	2	"Pinsky, Mrs. (Rosa)"
192	0	2	"Carbines, Mr. William"
193	1	3	"Andersen-Jensen, Miss. Carla Christine Nielsine"
194	1	2	"Navratil, Master. Michel M"
195	1	1	"Brown, Mrs. James Joseph (Margaret Tobin)"
196	1	1	"Lurette, Miss. Elise"
197	0	3	"Mernagh, Mr. Robert"
198	0	3	"Olsen, Mr. Karl Siegwart Andreas"
199	1	3	"Madigan, Miss. Margaret "Maggie" "
200	0	2	"Yrois, Miss. Henriette ("Mrs Harbeck")"
201	0	3	"Vande Walle, Mr. Nestor Cyriel"
202	0	3	"Sage, Mr. Frederick"
203	0	3	"Johanson, Mr. Jakob Alfred"
204	0	3	"Youseff, Mr. Gerious"
205	1	3	"Cohen, Mr. Gurshon "Gus" "
206	0	3	"Strom, Miss. Telma Matilda"
207	0	3	"Backstrom, Mr. Karl Alfred"
208	1	3	"Albimona, Mr. Nassef Cassem"
209	1	3	"Carr, Miss. Helen "Ellen" "
210	1	1	"Blank, Mr. Henry"
211	0	3	"Ali, Mr. Ahmed"
212	1	2	"Cameron, Miss. Clear Annie"
213	0	3	"Perkin, Mr. John Henry"
214	0	2	"Givard, Mr. Hans Kristensen"
215	0	3	"Kiernan, Mr. Philip"
216	1	1	"Newell, Miss. Madeleine"
217	1	3	"Honkanen, Miss. Eliina"
218	0	2	"Jacobsohn, Mr. Sidney Samuel"
219	1	1	"Bazzani, Miss. Albina"
220	0	2	"Harris, Mr. Walter"
221	1	3	"Sunderland, Mr. Victor Francis"
222	0	2	"Bracken, Mr. James H"
223	0	3	"Green, Mr. George Henry"
224	0	3	"Nenkoff, Mr. Christo"
225	1	1	"Hoyt, Mr. Frederick Maxfield"
226	0	3	"Berglund, Mr. Karl Ivar Sven"
227	1	2	"Mellors, Mr. William John"
228	0	3	"Lovell, Mr. John Hall ("Henry")"
229	0	2	"Fahlstrom, Mr. Arne Jonas"
230	0	3	"Lefebvre, Miss. Mathilde"
231	1	1	"Harris, Mrs. Henry Birkhardt (Irene Wallach)"
232	0	3	"Larsson, Mr. Bengt Edvin"
233	0	2	"Sjostedt, Mr. Ernst Adolf"
234	1	3	"Asplund, Miss. Lillian Gertrud"
235	0	2	"Leyson, Mr. Robert William Norman"
236	0	3	"Harknett, Miss. Alice Phoebe"
237	0	2	"Hold, Mr. Stephen"
238	1	2	"Collyer, Miss. Marjorie "Lottie" "
239	0	2	"Pengelly, Mr. Frederick William"
240	0	2	"Hunt, Mr. George Henry"
241	0	3	"Zabour, Miss. Thamine"
242	1	3	"Murphy, Miss. Katherine "Kate" "
243	0	2	"Coleridge, Mr. Reginald Charles"
244	0	3	"Maenpaa, Mr. Matti Alexanteri"
245	0	3	"Attalah, Mr. Sleiman"
246	0	1	"Minahan, Dr. William Edward"
247	0	3	"Lindahl, Miss. Agda Thorilda Viktoria"
248	1	2	"Hamalainen, Mrs. William (Anna)"

249	1	1	"Beckwith, Mr. Richard Leonard"
250	0	2	"Carter, Rev. Ernest Courtenay"
251	0	3	"Reed, Mr. James George"
252	0	3	"Strom, Mrs. Wilhelm (Elna Matilda Persson)"
253	0	1	"Stead, Mr. William Thomas"
254	0	3	"Lobb, Mr. William Arthur"
255	0	3	"Rosblom, Mrs. Viktor (Helena Wilhelmina)"

or alternatively, like Python Pandas

```
head(titanic)
```

```
ans = 8x12 table
```

...

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp
1	1	0	3	"Braund, ...	male	22	1
2	2	1	1	"Cumings,...	female	38	1
3	3	1	3	"Heikkine...	female	26	0
4	4	1	1	"Futrelle...	female	35	1
5	5	0	3	"Allen, M...	male	35	0
6	6	0	3	"Moran, M...	male	NaN	0
7	7	0	1	"McCarthy...	male	54	0
8	8	0	3	"Palsson,...	male	2	3

Get the dataset dimension

```
size(titanic) % rows columns
```

```
ans = 1x2
      891      12
```

Preliminary Data Analysis and Preprocessing

In the previous step, we have used the MATLAB's import function to import the text file, add meaningful column names,

and started this live script

Printing Summary Stats

```
summary(titanic) % remember the lecture slides - summary stats might be misleading!!!
```

Variables:

PassengerId: 891x1 double

Values:

Min	1
Median	446
Max	891

Survived: 891x1 categorical

Values:

0	549
1	342

Pclass: 891×1 categorical

Values:

1	216
2	184
3	491

Name: 891×1 string

Sex: 891×1 categorical

Values:

female	314
male	577

Age: 891×1 double

Values:

Min	0.42
Median	28
Max	80
NumMissing	177

SibSp: 891×1 categorical

Values:

0	608
1	209
2	28
3	16
4	18
5	5
8	7

Parch: 891×1 double

Values:

Min	0
Median	0
Max	6

Ticket: 891×1 string

Fare: 891×1 double

Values:

Min	0
Median	14.454
Max	512.33

Cabin: 891×1 string

Embarked: 891×1 categorical

Values:

C	168
Q	77
S	644
NumMissing	2

Getting summary stats values for analysis later

```
summary_vals = summary(titanic);  
sprintf("The minimum age values is %s",summary_vals.Age.Min)
```

```
ans =  
"The minimum age values is 4.200000e-01"
```

Grouping variables and calculating stats

```
varfun(@mean,titanic,'InputVariables','Age',...  
       'GroupingVariables','Sex')
```

```
ans = 2x3 table
```

	Sex	GroupCount	mean_Age
1	female	314	NaN
2	male	577	NaN

Handling Missing Data

Standard missing data is defined as:

- NaN - for double and single floating-point arrays
- NaN - for duration and calendarDuration arrays
- NaT - for datetime arrays
- <missing> - for string arrays
- <undefined> - for categorical arrays
- blank character [' '] - for character arrays
- empty character {''} - for cell arrays of character vectors

ismissing handles leading and trailing white space differently for indicators that are cell arrays of character vectors, character arrays, or categorical arrays.

- For cell arrays of character vectors, ismissing does not ignore indicator white space. All character vectors must match exactly.
- For character arrays in table variables, ismissing ignores trailing white space in the indicator.
- For categorical arrays, ismissing ignores leading and trailing white space in the indicator.

Using "Task" control functionality

```
% Fill missing data  
[cleanedData,missingIndices] = fillmissing(titanic.Age,'constant',28);  
  
% Visualize results
```

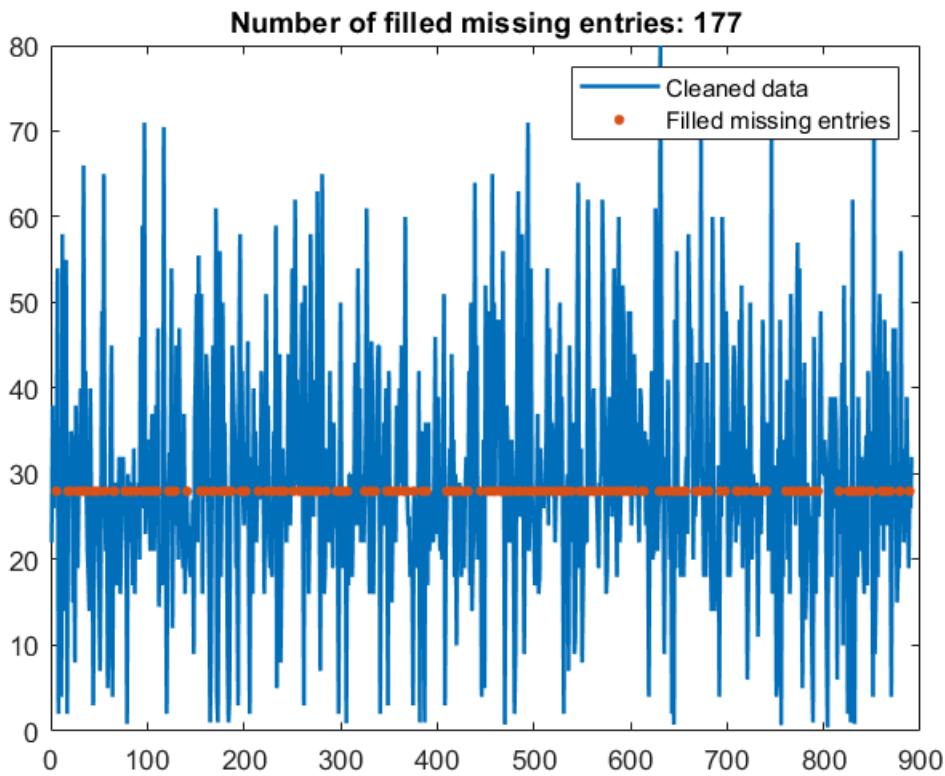
```

clf
plot(cleanedData,'Color',[0 114 189]/255,'LineWidth',1.5,...
      'DisplayName','Cleaned data')
hold on

% Plot filled missing entries
plot(find(missingIndices),cleanedData(missingIndices),'.','MarkerSize',12,...
      'Color',[217 83 25]/255,'DisplayName','Filled missing entries')
title(['Number of filled missing entries: ' num2str(nnz(missingIndices))])

hold off
legend

```



```
clear missingIndices
```

```
titanic % the Age column still has missing values - remember to reassign the values to the table
```

```
titanic = 891x12 table
```

...

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp
1	1	0	3	"Braund, ...	male	22	1
2	2	1	1	"Cumings,...	female	38	1
3	3	1	3	"Heikkine...	female	26	0
4	4	1	1	"Futrelle...	female	35	1
5	5	0	3	"Allen, M...	male	35	0

36	0	1	"Holverson, Mr. Alexander Oskar"
38	0	3	"Cann, Mr. Ernest Charles"
39	0	3	"Vander Planke, Miss. Augusta Maria"
40	1	3	"Nicola-Yarred, Miss. Jamila"
41	0	3	"Ahlin, Mrs. Johan (Johanna Persdotter Larsson)"
42	0	2	"Turpin, Mrs. William John Robert (Dorothy Ann Wonnacott)"
44	1	2	"Laroche, Miss. Simonne Marie Anne Andree"
45	1	3	"Devaney, Miss. Margaret Delia"
50	0	3	"Arnold-Franchi, Mrs. Josef (Josefine Franchi)"
51	0	3	"Panula, Master. Juha Niilo"
52	0	3	"Nosworthy, Mr. Richard Cater"
54	1	2	"Faunthorpe, Mrs. Lizzie (Elizabeth Anne Wilkinson)"
57	1	2	"Rugg, Miss. Emily"
58	0	3	"Novel, Mr. Mansouer"
59	1	2	"West, Miss. Constance Mirium"
60	0	3	"Goodwin, Master. William Frederick"
61	0	3	"Sirayanian, Mr. Orsen"
64	0	3	"Skoog, Master. Harald"
68	0	3	"Crease, Mr. Ernest James"
69	1	3	"Andersson, Miss. Erna Alexandra"
70	0	3	"Kink, Mr. Vincenz"
71	0	2	"Jenkin, Mr. Stephen Curnow"
72	0	3	"Goodwin, Miss. Lillian Amy"
73	0	2	"Hood, Mr. Ambrose Jr"
74	0	3	"Chronopoulos, Mr. Apostolos"
75	1	3	"Bing, Mr. Lee"
79	1	2	"Caldwell, Master. Alden Gates"
80	1	3	"Dowdell, Miss. Elizabeth"
81	0	3	"Waelens, Mr. Achille"
82	1	3	"Sheerlinck, Mr. Jan Baptist"
84	0	1	"Carrau, Mr. Francisco M"
85	1	2	"Ilett, Miss. Bertha"
86	1	3	"Backstrom, Mrs. Karl Alfred (Maria Mathilda Gustafsson)"
87	0	3	"Ford, Mr. William Neal"
90	0	3	"Celotti, Mr. Francesco"
91	0	3	"Christmann, Mr. Emil"
92	0	3	"Andreasson, Mr. Paul Edvin"
94	0	3	"Dean, Mr. Bertram Frank"
95	0	3	"Coxon, Mr. Daniel"
99	1	2	"Doling, Mrs. John T (Ada Julia Bone)"
100	0	2	"Kantor, Mr. Sinai"
101	0	3	"Petranec, Miss. Matilda"
104	0	3	"Johansson, Mr. Gustaf Joel"
105	0	3	"Gustafsson, Mr. Anders Vilhelm"
106	0	3	"Mionoff, Mr. Stoytcho"
107	1	3	"Salkjelsvik, Miss. Anna Kristine"
109	0	3	"Rekic, Mr. Tido"
112	0	3	"Zabour, Miss. Hileni"
113	0	3	"Barton, Mr. David John"
114	0	3	"Jussila, Miss. Katriina"
115	0	3	"Attalah, Miss. Malake"
116	0	3	"Pekoniemi, Mr. Edvard"
117	0	3	"Connors, Mr. Patrick"
118	0	2	"Turpin, Mr. William John Robert"
120	0	3	"Andersson, Miss. Ellis Anna Maria"
121	0	2	"Hickman, Mr. Stanley George"
123	0	2	"Nasser, Mr. Nicholas"
126	1	3	"Nicola-Yarred, Master. Elias"
128	1	3	"Madsen, Mr. Fridtjof Arne"
130	0	3	"Ekstrom, Mr. Johan"
131	0	3	"Drazenoic, Mr. Jozef"
132	0	3	"Coelho, Mr. Domingos Fernando"
133	0	3	"Robins, Mrs. Alexander A (Grace Charity Laury)"
134	1	2	"Weisz, Mrs. Leopold (Mathilde Francoise Pede)"
135	0	2	"Sobey, Mr. Samuel James Hayden"

136	0	2	"Richard, Mr. Emile"
139	0	3	"Osen, Mr. Olaf Elon"
142	1	3	"Nysten, Miss. Anna Sofia"
143	1	3	"Hakkarainen, Mrs. Pekka Pietari (Elin Matilda Dolck)"
144	0	3	"Burke, Mr. Jeremiah"
145	0	2	"Andrew, Mr. Edgardo Samuel"
146	0	2	"Nicholls, Mr. Joseph Charles"
147	1	3	"Andersson, Mr. August Edvard ("Wennerstrom")"
148	0	3	"Ford, Miss. Robina Maggie "Ruby""
150	0	2	"Byles, Rev. Thomas Roussel Davids"
151	0	2	"Bateman, Rev. Robert James"
153	0	3	"Meo, Mr. Alfonzo"
154	0	3	"van Billiard, Mr. Austin Blyler"
156	0	1	"Williams, Mr. Charles Duane"
157	1	3	"Gilnagh, Miss. Katherine "Katie""
158	0	3	"Corn, Mr. Harry"
161	0	3	"Cribb, Mr. John Hatfield"
162	1	2	"Watt, Mrs. James (Elizabeth "Bessie" Inglis Milne)"
163	0	3	"Bengtsson, Mr. John Viktor"
164	0	3	"Calic, Mr. Jovo"
165	0	3	"Panula, Master. Eino Viljami"
166	1	3	"Goldsmith, Master. Frank John William "Frankie""
168	0	3	"Skoog, Mrs. William (Anna Bernhardina Karlsson)"
170	0	3	"Ling, Mr. Lee"
172	0	3	"Rice, Master. Arthur"
173	1	3	"Johnson, Miss. Eleanor Ileen"
174	0	3	"Sivola, Mr. Antti Wilhelm"
176	0	3	"Klasen, Mr. Klas Albin"
179	0	2	"Hale, Mr. Reginald"
180	0	3	"Leonard, Mr. Lionel"
183	0	3	"Asplund, Master. Clarence Gustaf Hugo"
185	1	3	"Kink-Heilmann, Miss. Luise Gretchen"
188	1	1	"Romaine, Mr. Charles Hallace ("Mr C Rolmane")"
189	0	3	"Bourke, Mr. John"
190	0	3	"Turcin, Mr. Stjepan"
191	1	2	"Pinsky, Mrs. (Rosa)"
192	0	2	"Carbines, Mr. William"
193	1	3	"Andersen-Jensen, Miss. Carla Christine Nielsine"
198	0	3	"Olsen, Mr. Karl Siegwart Andreas"
200	0	2	"Yrois, Miss. Henriette ("Mrs Harbeck")"
201	0	3	"Vande Walle, Mr. Nestor Cyriel"
203	0	3	"Johanson, Mr. Jakob Alfred"
204	0	3	"Youseff, Mr. Gerious"
205	1	3	"Cohen, Mr. Gurshon "Gus""
207	0	3	"Backstrom, Mr. Karl Alfred"
208	1	3	"Albimona, Mr. Nassef Cassem"
209	1	3	"Carr, Miss. Helen "Ellen""
211	0	3	"Ali, Mr. Ahmed"
212	1	2	"Cameron, Miss. Clear Annie"
213	0	3	"Perkin, Mr. John Henry"
214	0	2	"Givard, Mr. Hans Kristensen"
217	1	3	"Honkanen, Miss. Eliina"
218	0	2	"Jacobsohn, Mr. Sidney Samuel"
220	0	2	"Harris, Mr. Walter"
221	1	3	"Sunderland, Mr. Victor Francis"
222	0	2	"Bracken, Mr. James H"
223	0	3	"Green, Mr. George Henry"
226	0	3	"Berglund, Mr. Karl Ivar Sven"
227	1	2	"Mellors, Mr. William John"
228	0	3	"Lovell, Mr. John Hall ("Henry")"
229	0	2	"Fahlstrom, Mr. Arne Jonas"
232	0	3	"Larsson, Mr. Bengt Edvin"
233	0	2	"Sjostedt, Mr. Ernst Adolf"
234	1	3	"Asplund, Miss. Lillian Gertrud"
235	0	2	"Leyson, Mr. Robert William Norman"

237	0	2	"Hold, Mr. Stephen"
238	1	2	"Collyer, Miss. Marjorie "Lottie""
239	0	2	"Pengelly, Mr. Frederick William"
240	0	2	"Hunt, Mr. George Henry"
243	0	2	"Coleridge, Mr. Reginald Charles"
244	0	3	"Maenpaa, Mr. Matti Alexanteri"
245	0	3	"Attalah, Mr. Sleiman"
247	0	3	"Lindahl, Miss. Agda Thorilda Viktoria"
248	1	2	"Hamalainen, Mrs. William (Anna)"
250	0	2	"Carter, Rev. Ernest Courtenay"
254	0	3	"Lobb, Mr. William Arthur"
255	0	3	"Rosblom, Mrs. Viktor (Helena Wilhelmina)"
256	1	3	"Touma, Mrs. Darwis (Hanne Youssef Razi)"
259	1	1	"Ward, Miss. Anna"
260	1	2	"Parrish, Mrs. (Lutie Davis)"
262	1	3	"Asplund, Master. Edvin Rojj Felix"
266	0	2	"Reeves, Mr. David"
267	0	3	"Panula, Mr. Ernesti Arvid"
268	1	3	"Persson, Mr. Ernst Ulrik"
272	1	3	"Tornquist, Mr. William Henry"
273	1	2	"Mellinger, Mrs. (Elizabeth Anne Maidment)"
277	0	3	"Lindblom, Miss. Augusta Charlotta"
279	0	3	"Rice, Master. Eric"
280	1	3	"Abbott, Mrs. Stanton (Rosa Hunt)"
281	0	3	"Duane, Mr. Frank"
282	0	3	"Olsson, Mr. Nils Johan Goransson"
283	0	3	"de Pelsmaeker, Mr. Alfons"
284	1	3	"Dorking, Mr. Edward Arthur"
286	0	3	"Stankovic, Mr. Ivan"
287	1	3	"de Mulder, Mr. Theodore"
288	0	3	"Naidenoff, Mr. Penko"
289	1	2	"Hosono, Mr. Masabumi"
290	1	3	"Connolly, Miss. Kate"
291	1	1	"Barber, Miss. Ellen "Nellie""
294	0	3	"Haas, Miss. Aloisia"
295	0	3	"Mineff, Mr. Ivan"
297	0	3	"Hanna, Mr. Mansour"
303	0	3	"Johnson, Mr. William Cahoon Jr"
309	0	2	"Abelson, Mr. Samuel"
313	0	2	"Lahtinen, Mrs. William (Anna Sylfven)"
314	0	3	"Hendekovic, Mr. Ignjac"
315	0	2	"Hart, Mr. Benjamin"
316	1	3	"Nilsson, Miss. Helmina Josefina"
317	1	2	"Kantor, Mrs. Sinai (Miriam Sternin)"
318	0	2	"Moraweck, Dr. Ernest"
321	0	3	"Dennis, Mr. Samuel"
322	0	3	"Danoff, Mr. Yoto"
323	1	2	"Slayter, Miss. Hilda Mary"
324	1	2	"Caldwell, Mrs. Albert Francis (Sylvia Mae Harbaugh)"
327	0	3	"Nysveen, Mr. Johan Hansen"
329	1	3	"Goldsmith, Mrs. Frank John (Emily Alice Brown)"
334	0	3	"Vander Planke, Mr. Leo Edmondus"
339	1	3	"Dahl, Mr. Karl Edwart"
343	0	2	"Collander, Mr. Erik Gustaf"
344	0	2	"Sedgwick, Mr. Charles Frederick Waddington"
345	0	2	"Fox, Mr. Stanley Hubert"
347	1	2	"Smith, Miss. Marion Elsie"
349	1	3	"Coutts, Master. William Loch "William""
350	0	3	"Dimic, Mr. Jovan"
351	0	3	"Odahl, Mr. Nils Martin"
353	0	3	"Elias, Mr. Tannous"
354	0	3	"Arnold-Franchi, Mr. Josef"
356	0	3	"Vanden Steen, Mr. Leo Peter"
358	0	2	"Funk, Miss. Annie Clemmer"
361	0	3	"Skoog, Mr. Wilhelm"

362	0	2	"del Carlo, Mr. Sebastiano"	ma
363	0	3	"Barbara, Mrs. (Catherine David)"	fe
364	0	3	"Asim, Mr. Adola"	ma
366	0	3	"Adahl, Mr. Mauritz Nils Martin"	ma
372	0	3	"Wiklund, Mr. Jakob Alfred"	ma
373	0	3	"Beavan, Mr. William Thomas"	ma
374	0	1	"Ringhini, Mr. Sante"	ma
375	0	3	"Palsson, Miss. Stina Viola"	fe
377	1	3	"Landergren, Miss. Aurora Adelia"	fe
379	0	3	"Betros, Mr. Tannous"	ma
380	0	3	"Gustafsson, Mr. Karl Gideon"	ma
381	1	1	"Bidois, Miss. Rosalie"	fe
382	1	3	"Nakid, Miss. Maria ("Mary")"	fe
383	0	3	"Tikkanen, Mr. Juho"	ma
384	1	1	"Holverson, Mrs. Alexander Oskar (Mary Aline Towner)"	fe
386	0	2	"Davies, Mr. Charles Henry"	ma
387	0	3	"Goodwin, Master. Sidney Leonard"	ma
388	1	2	"Buss, Miss. Kate"	fe
390	1	2	"Lehmann, Miss. Bertha"	fe
392	1	3	"Jansson, Mr. Carl Olof"	ma
393	0	3	"Gustafsson, Mr. Johan Birger"	ma
396	0	3	"Johansson, Mr. Erik"	ma
397	0	3	"Olsson, Miss. Elina"	fe
398	0	2	"McKane, Mr. Peter David"	ma
399	0	2	"Pain, Dr. Alfred"	ma
400	1	2	"Trout, Mrs. William H (Jessie L)"	fe
401	1	3	"Niskanen, Mr. Juha"	ma
402	0	3	"Adams, Mr. John"	ma
403	0	3	"Jussila, Miss. Mari Aina"	fe
404	0	3	"Hakkarainen, Mr. Pekka Pietari"	ma
405	0	3	"Oreskovic, Miss. Marija"	fe
406	0	2	"Gale, Mr. Shadrach"	ma
407	0	3	"Widegren, Mr. Carl/Charles Peter"	ma
408	1	2	"Richards, Master. William Rowe"	ma
409	0	3	"Birkeland, Mr. Hans Martin Monsen"	ma
415	1	3	"Sundman, Mr. Johan Julian"	ma
417	1	2	"Drew, Mrs. James Vivian (Lulu Thorne Christian)"	fe
418	1	2	"Silven, Miss. Lyyli Karoliina"	fe
419	0	2	"Matthews, Mr. William John"	ma
420	0	3	"Van Impe, Miss. Catharina"	fe
422	0	3	"Charters, Mr. David"	ma
423	0	3	"Zimmerman, Mr. Leo"	ma
424	0	3	"Danbom, Mrs. Ernst Gilbert (Anna Sigrid Maria Brogren)"	fe
425	0	3	"Rosblom, Mr. Viktor Richard"	ma
427	1	2	"Clarke, Mrs. Charles V (Ada Maria Winfield)"	fe
428	1	2	"Phillips, Miss. Kate Florence ("Mrs Kate Louise Phillips Marshall")"	fe
433	1	2	"Louch, Mrs. Charles Alexander (Alice Adelaide Slow)"	fe
434	0	3	"Kallio, Mr. Nikolai Erland"	ma
437	0	3	"Ford, Miss. Doolina Margaret "Daisy""	fe
438	1	2	"Richards, Mrs. Sidney (Emily Hocking)"	fe
440	0	2	"Kvillner, Mr. Johan Henrik Johannesson"	ma
441	1	2	"Hart, Mrs. Benjamin (Esther Ada Bloomfield)"	fe
442	0	3	"Hampe, Mr. Leon"	ma
443	0	3	"Pettersson, Mr. Johan Emil"	ma
444	1	2	"Reynaldo, Ms. Encarnacion"	fe
447	1	2	"Mellinger, Miss. Madeleine Violet"	fe
448	1	1	"Seward, Mr. Frederic Kimber"	ma
449	1	3	"Baclini, Miss. Marie Catherine"	fe
451	0	2	"West, Mr. Edwy Arthur"	ma
456	1	3	"Jalsevac, Mr. Ivan"	ma
459	1	2	"Toomey, Miss. Ellen"	fe
462	0	3	"Morley, Mr. William"	ma
464	0	2	"Milling, Mr. Jacob Christian"	ma
466	0	3	"Goncalves, Mr. Manuel Estanslas"	ma
468	0	1	"Smart, Mr. John Montgomery"	ma

```

470      1      3      "Baclini, Miss. Helene Barbara"
472      0      3      "Cacic, Mr. Luka"
473      1      2      "West, Mrs. Edwy Arthur (Ada Mary Worth)"
475      0      3      "Strandberg, Miss. Ida Sofia"
477      0      2      "Renouf, Mr. Peter Henry"
478      0      3      "Braund, Mr. Lewis Richard"
479      0      3      "Karlsson, Mr. Nils August"
480      1      3      "Hirvonen, Miss. Hildur E"
481      0      3      "Goodwin, Master. Harold Victor"
483      0      3      "Rouse, Mr. Richard Henry"
484      1      3      "Turkula, Mrs. (Hedwig)"
489      0      3      "Somerton, Mr. Francis William"
490      1      3      "Coutts, Master. Eden Leslie "Neville""
492      0      3      "Windelov, Mr. Einar"

```

...

This is NOT a good approach as it might result in loss of lots of important data (we are left with only 362 observations)

```
size(titanic_rows)
```

```
ans = 1x2
      529      12
```

Removing Columns

```
sum(ismissing(titanic,{" " ' ' '.' '<undefined>' 'NA' NaN -99}))
```

```
ans = 1x12
      0      0      0      0      0      177      0      0      0      0      687      2
```

We see from above that, quite a lot - 687 (approx 70%) of values of Cabin are missing. Similarly, 177 data entries for column Age is missing. Dealing with missing values is tricky, as discussed earlier. We simply cannot remove 687 rows for Cabin or 177 rows for Ages. But we can (and we will) drop the Cabin column (as most of its values are missing and it does not add any value to the dataset). Other than Age and Cabin, Embarked column has 2 missing values.

```
titanic = removevars(titanic,{'Cabin'})
```

```
titanic = 891x11 table
```

...

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp
1	1	0	3	"Braund, ...	male	22	1
2	2	1	1	"Cumings,...	female	38	1
3	3	1	3	"Heikkine...	female	26	0
4	4	1	1	"Futrelle...	female	35	1
5	5	0	3	"Allen, M...	male	35	0
6	6	0	3	"Moran, M...	male	NaN	0
7	7	0	1	"McCarthy...	male	54	0

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp
8	8	0	3	"Palsson,...	male	2	3
9	9	1	3	"Johnson,...	female	27	0
10	10	1	2	"Nasser, ...	female	14	1
11	11	1	3	"Sandstro...	female	4	1
12	12	1	1	"Bonnell,...	female	58	0
13	13	0	3	"Saunders...	male	20	0
14	14	0	3	"Andersso...	male	39	1

⋮

```
size(titanic)
```

```
ans = 1x2
      891      11
```

```
% Just for Information
% T.Properties.VariableNames{end} = 'Final' ----> rename the last column
% T.EleventhColumn = rand(8,1); - to add a new column towards the end
```

Replacing missing Age values using the Median

```
titanic.Age = fillmissing(titanic.Age, 'constant', summary_vals.Age.Median); % size(titanic,2) is
                                                                    % of the number of rows
%disp(titanic)
```

Check for missing values in another column - Embarked column (categorical variable)

First lets check the number of occurrence of values of Embarked column

```
tabulate(titanic.Embarked) % tabulate(x) displays a frequency table of the data in the vector x
```

Value	Count	Percent
C	168	18.90%
Q	77	8.66%
S	644	72.44%

```
% For each unique value in x, the tabulate function shows the number of instances and percentage
```

We see that Embarked class 'S' occurs the highest number of times - hence we replace the missing values with Class 'S'

```
titanic.Embarked = fillmissing(titanic.Embarked, 'constant', 'S')
```

```
titanic = 891x11 table
```

...

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp
1	1	0	3	"Braund, ...	male	22	1
2	2	1	1	"Cumings,...	female	38	1

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp
3	3	1	3	"Heikkine...	female	26	0
4	4	1	1	"Futrelle...	female	35	1
5	5	0	3	"Allen, M...	male	35	0
6	6	0	3	"Moran, M...	male	28	0
7	7	0	1	"McCarthy...	male	54	0
8	8	0	3	"Palsson,...	male	2	3
9	9	1	3	"Johnson,...	female	27	0
10	10	1	2	"Nasser, ...	female	14	1
11	11	1	3	"Sandstro...	female	4	1
12	12	1	1	"Bonnell,...	female	58	0
13	13	0	3	"Saunderc...	male	20	0
14	14	0	3	"Andersso...	male	39	1

⋮

```
%titanic = fillmissing(titanic,'constant','S','DataVariables',{'Embarked'})
```

Check if the Embarked column is not without any missing value

```
tabulate(ismissing(titanic.Embarked))
```

```
Value    Count    Percent
      0         891    100.00%
```

A final check to see if all missing values are now handled

```
%tabulate(ismissing(titanic))
TF = ismissing(titanic);
rowsWithMissing = titanic(any(TF,2),:)
```

```
rowsWithMissing =
```

```
0x11 empty table
```

```
%disp(rowsWithMissing)
```

Handling Outliers

Note: There is also the Data Preprocessing toolbox functionality you can access graphically using the "Task" functionality (see below for a demo)

Using "Task" control functionality

```
% Remove outliers
[cleanedData2,outlierIndices] = rmoutliers(titanic.Fare,'quantiles',...
    'ThresholdFactor',2.75);
```

```

% Visualize results
clf
plot(titanic.Fare,'Color',[109 185 226]/255,'DisplayName','Input data')
hold on
plot(find(~outlierIndices),cleanedData2,'Color',[0 114 189]/255,'LineWidth',1.5,...
     'DisplayName','Cleaned data')

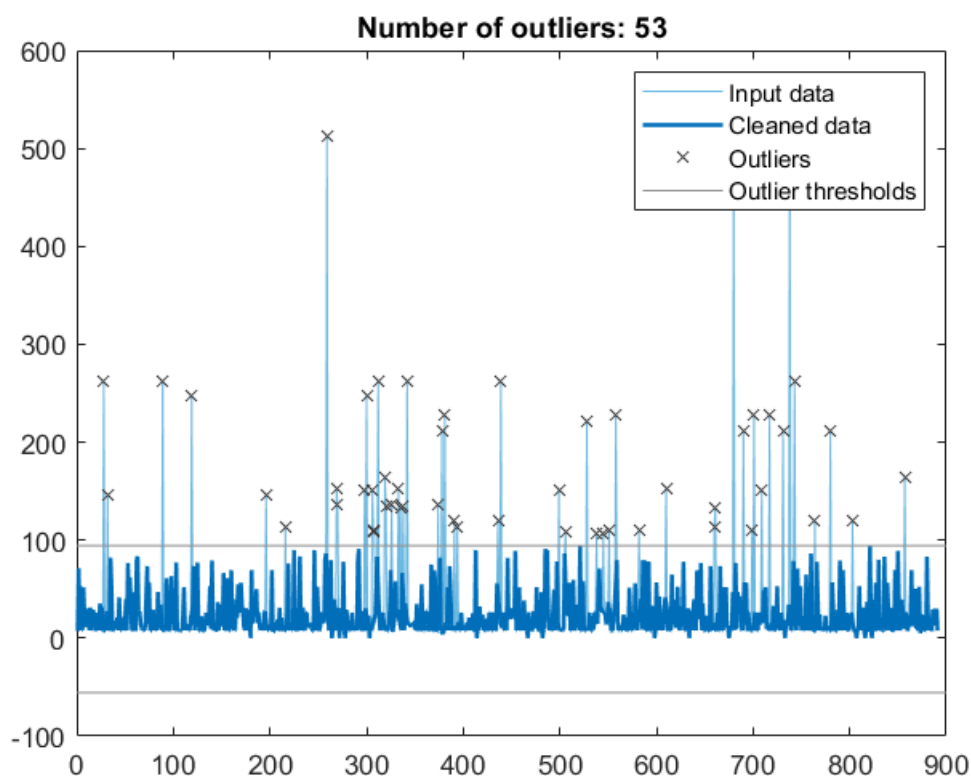
% Plot outliers
plot(find(outlierIndices),titanic.Fare(outlierIndices),'x',...
     'Color',[64 64 64]/255,'DisplayName','Outliers')
title(['Number of outliers: ' num2str(nnz(outlierIndices))])

% Compute thresholds and center
[~,thresholdLow,thresholdHigh] = isoutlier(titanic.Fare,'quartiles',...
     'ThresholdFactor',2.75);

% Plot outlier thresholds
plot([xlim missing xlim],[thresholdLow*[1 1] NaN thresholdHigh*[1 1]],...
     'Color',[145 145 145]/255,'DisplayName','Outlier thresholds')

hold off
legend

```

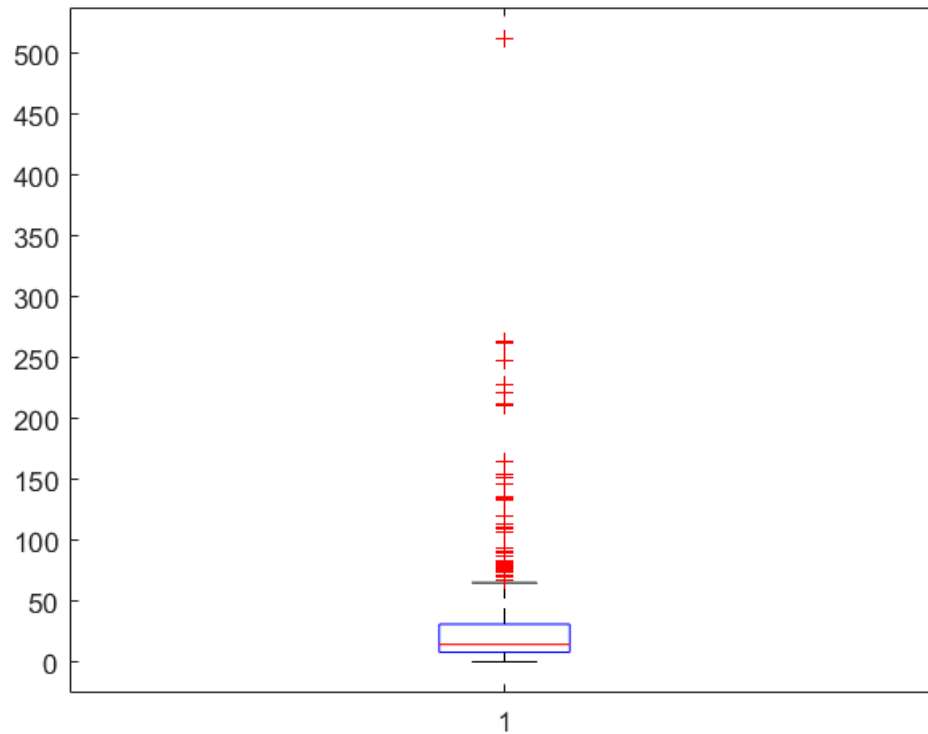


```
clear thresholdLow thresholdHigh
```

Doing programmatically by writing code

```
% to check for outliers, one can use boxplot
```

```
boxplot(titanic.Fare) % we do not have any!!!!
```



```
% to check the percentile distribution  
prctile(titanic.Fare,[0,99.90])
```

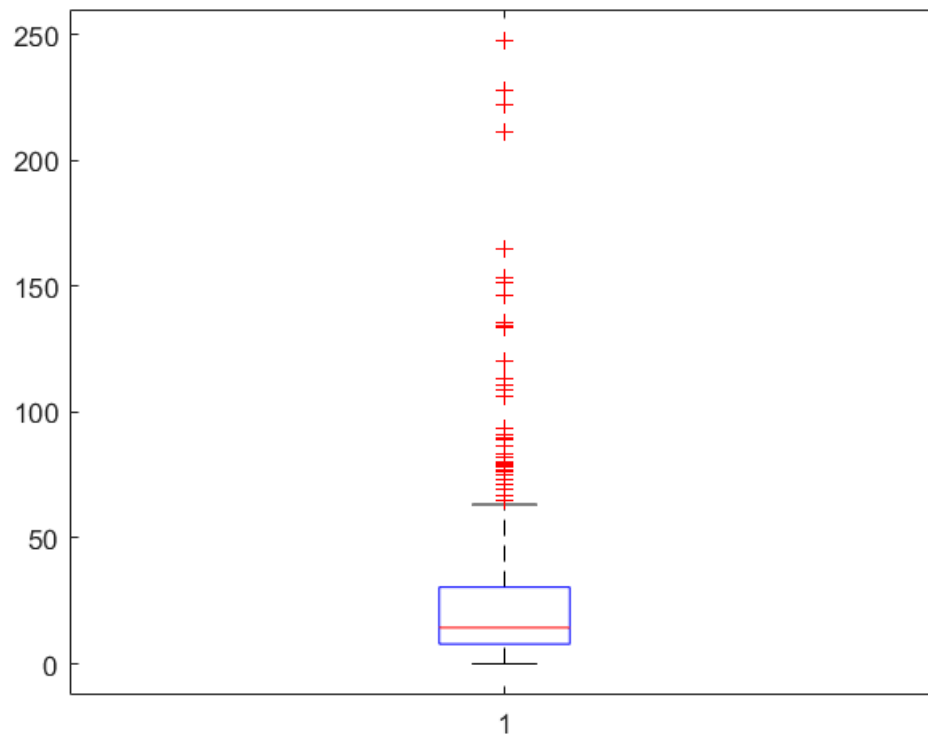
```
ans = 1x2  
      0  512.3292
```

To remove the outliers (in case we have any)

```
new_fare = rmoutliers(titanic.Fare,'percentiles',[0,99])
```

```
new_fare = 882x1  
7.2500  
71.2833  
7.9250  
53.1000  
8.0500  
8.4583  
51.8625  
21.0750  
11.1333  
30.0708  
⋮  
⋮
```

```
boxplot(new_fare)
```



Save the pre-processed file

```
writetable(titanic, 'titanic_pre_processed.csv')
```

Visualization of Titanic Dataset

Quick plotting can help indicate outliers, skewed data, etc. Lets plot some charts

Chart 01: Survival count by passenger count

```
histogram(titanic.Survived)
xlabel("Survival")
ylabel('Number of passengers')
title('Number of passengers per survival class')
xt = xticklabels;
xt = {'Not Survived', 'Survived'}; % %xt{1} = 'Not Survived', %xt{2} = 'Survived'
xticklabels(xt)
```

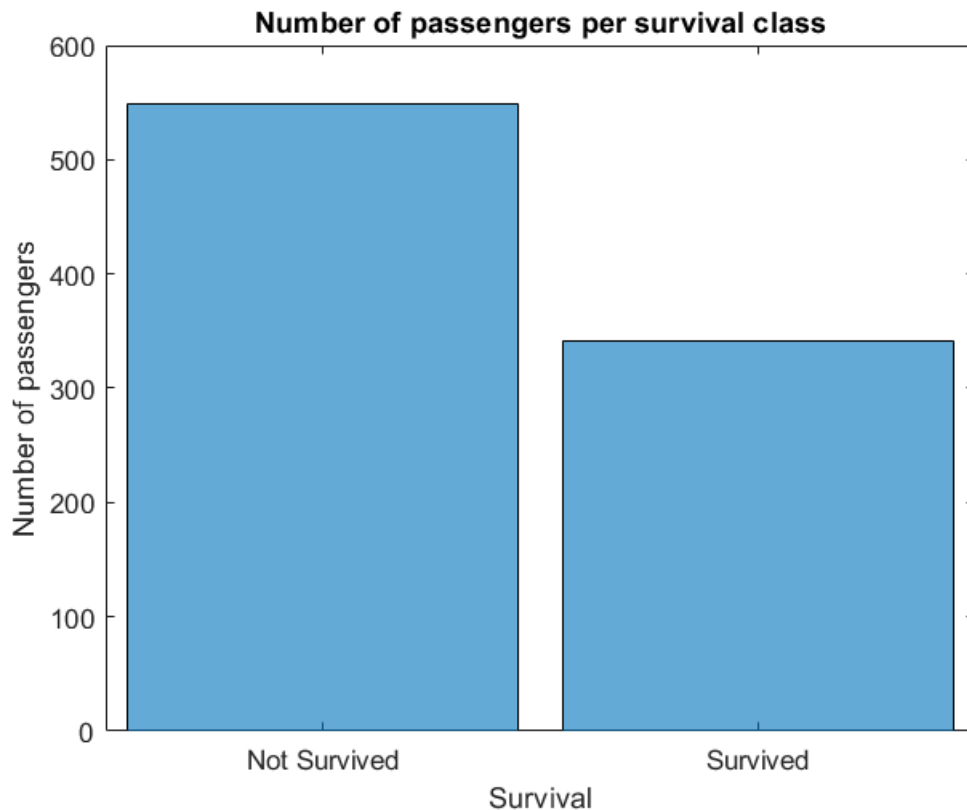


Chart 02: Passenger count grouped by Passenger Class

```
% Number of counts for Passenger class 1
rows = titanic.Pclass == '1';
vals = size(titanic.PassengerId(rows,:));
total_count_pclasss1 = vals(1);
% Number of counts for Passenger class 2
rows = titanic.Pclass == '2';
vals = size(titanic.PassengerId(rows,:));
total_count_pclasss2 = vals(1);
% Number of counts for Passenger class 3
rows = titanic.Pclass == '3';
vals = size(titanic.PassengerId(rows,:));
total_count_pclasss3 = vals(1);

x = categorical(["PClass1" "PClass2" "PClass3"]);
x = reordercats(x,{'PClass1' 'PClass2' 'PClass3'}); % Categoryicals can have order associated with them
y = [total_count_pclasss1,total_count_pclasss2,total_count_pclasss3];
bar(x,y);
```

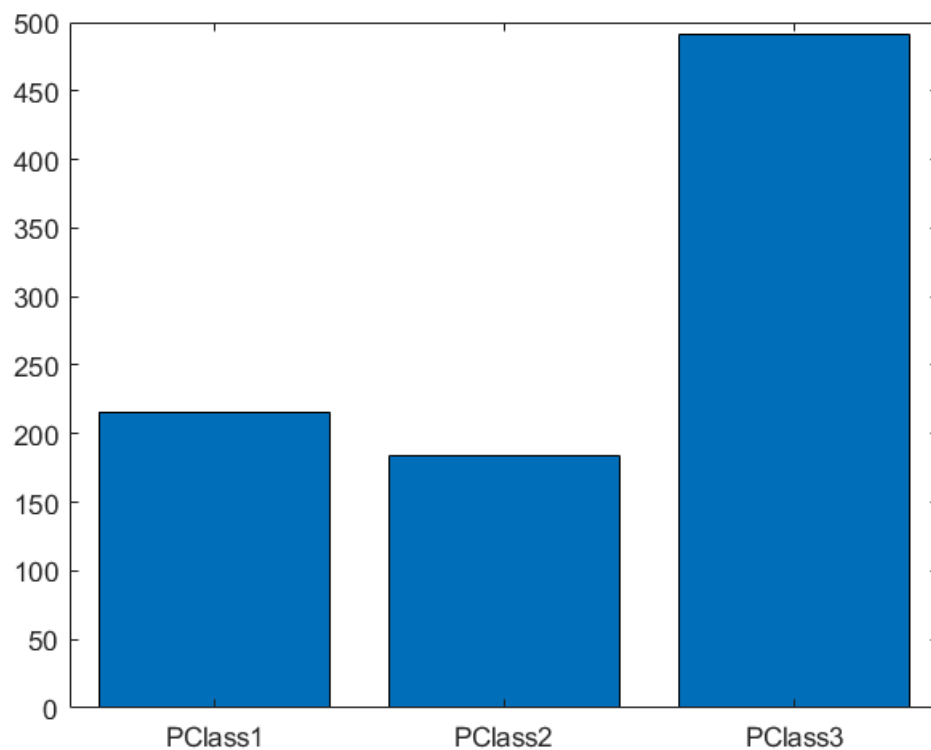
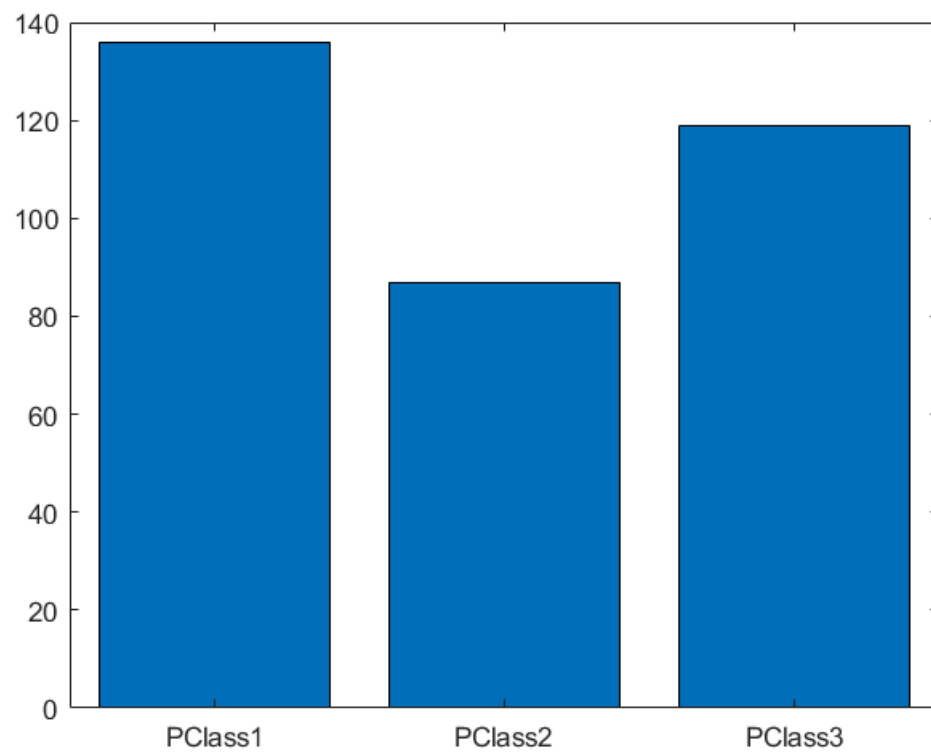



Chart 03: Survival Count by Passenger class

```
% Number of counts for Passenger class 1
rows = titanic.Pclass == '1';
vals= titanic.Survived(rows,:);
countvals = sum(vals(:) == '1');
total_count_pclasss1 = countvals(1);
% Number of counts for Passenger class 2
rows = titanic.Pclass == '2';
vals= titanic.Survived(rows,:);
countvals = sum(vals(:) == '1');
total_count_pclasss2 = countvals(1);
% Number of counts for Passenger class 3
rows = titanic.Pclass == '3';
vals= titanic.Survived(rows,:);
countvals = sum(vals(:) == '1');
total_count_pclasss3 = countvals(1);
x = categorical(["PClass1" "PClass2" "PClass3"]);
x = reordercats(x,{'PClass1' 'PClass2' 'PClass3'});
y = [total_count_pclasss1,total_count_pclasss2,total_count_pclasss3];
bar(x,y);
```



Observations

- More first class passengers survived (about 2/3rd)
- Third class passengers had very little chance of survival (about 7/9 died)
- Second class passengers had a near 50% chance of survival

A better idea would be to plot the survival percentage by class

Chart 04: Plotting scatter plot of Age grouped by Survival

```
gscatter(titanic.Age,titanic.Fare,titanic.Survived)
```

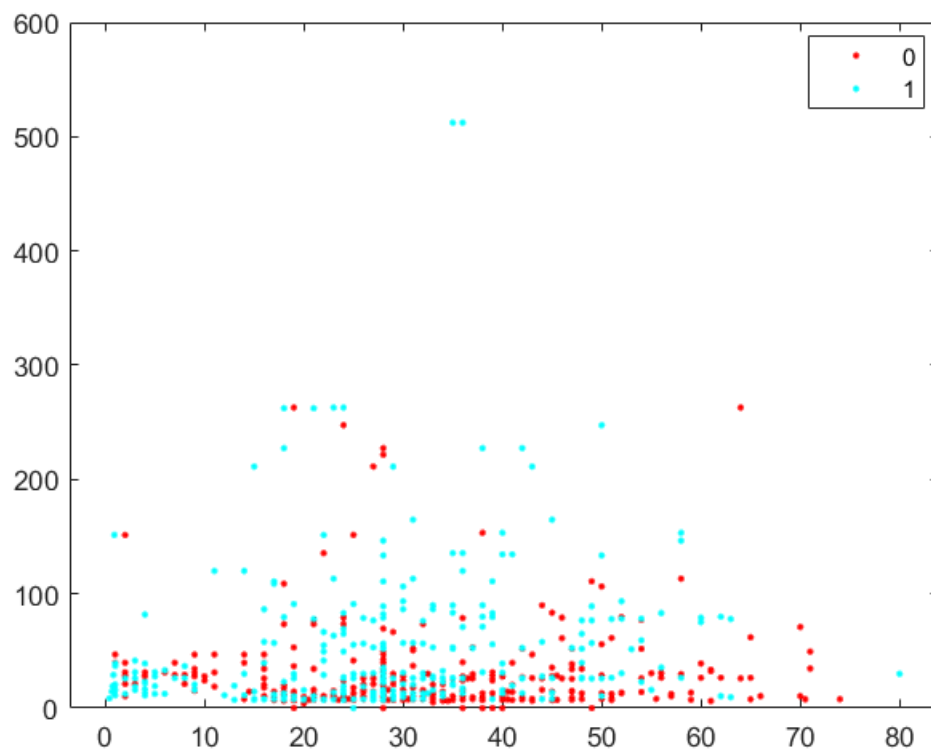


Chart 05: Plotting scatter plot of Age grouped by Pclass

```
gplotmatrix(titanic.Age,titanic.Fare,titanic.Pclass)
```

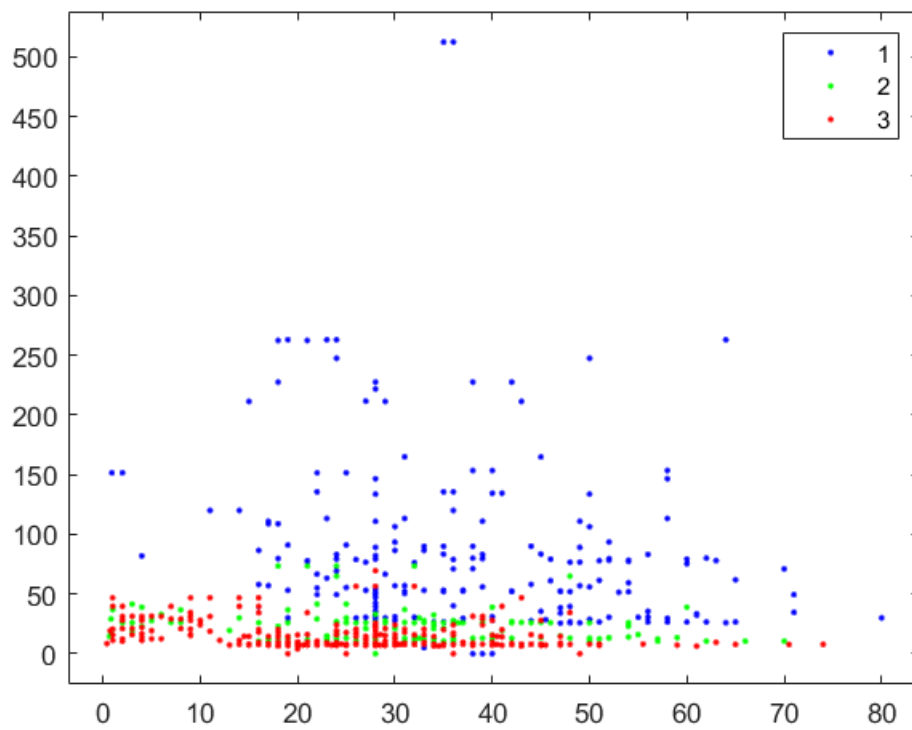
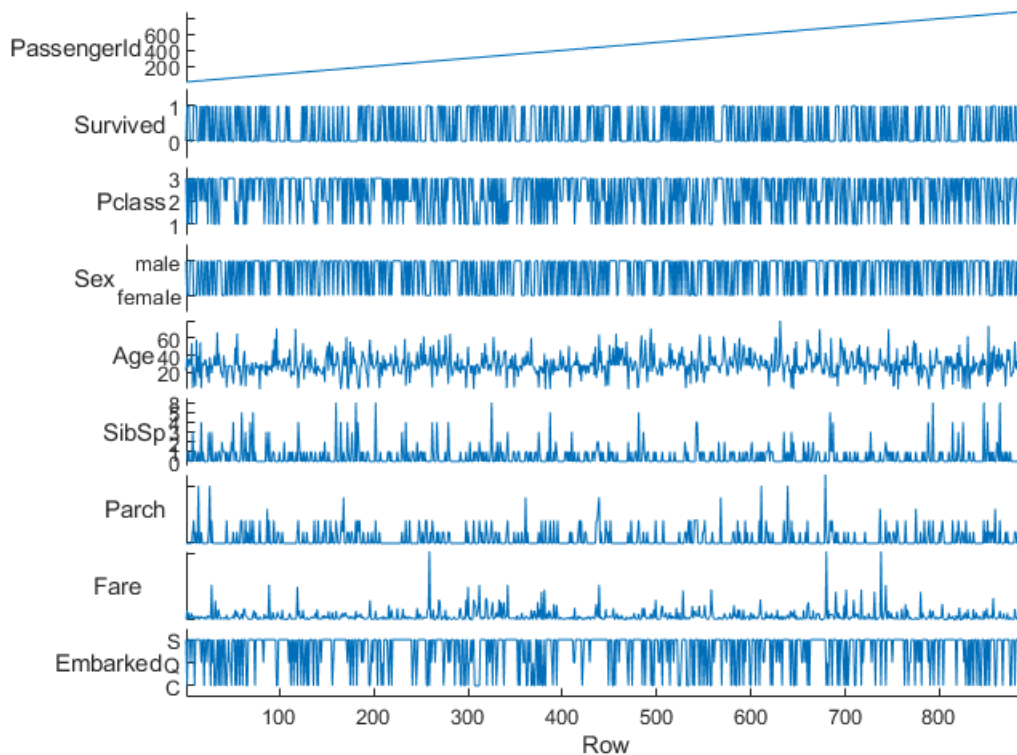


Chart 06: Stacked Plot of all column variables

```
stackedplot(titanic)
```



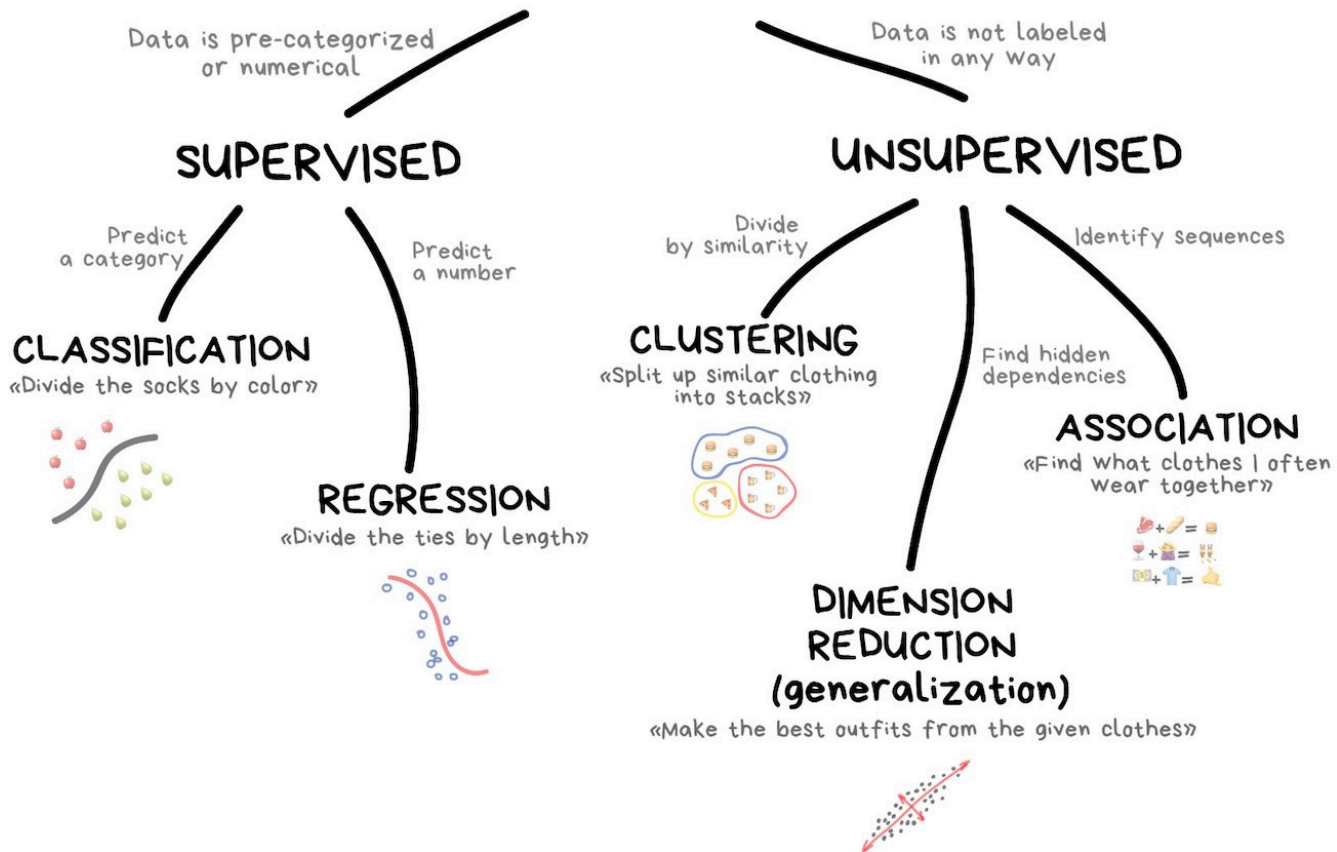
```
ans =
StackedLineChart with properties:

    SourceTable: [891x11 table]
    DisplayVariables: {'PassengerId' 'Survived' 'Pclass' 'Sex' 'Age' 'SibSp' 'Parch' 'Fare' 'Embarked'}
    XVariable: []
    Color: [0 0.4470 0.7410]
    LineStyle: '-'
    LineWidth: 0.5000
    Marker: 'none'
    MarkerSize: 6

Show all properties
```

Machine Learning with Titanic Dataset - Predicting survival on the Titanic

CLASSICAL MACHINE LEARNING



Classification

In classification problems we split input examples by certain characteristic.

Usage examples:

- Spam filters (e.g., Naive Bayes algorithm)
- Fraud detection (e.g., Decision Trees)
- A search of similar documents
- Sentiment analysis
- Handwritten characters and numbers recognition

Popular algorithms: [Naive Bayes](#), [Decision Tree](#), [Logistic Regression](#), [K-Nearest Neighbours](#), [Support Vector Machine](#)

Evaluating a classification model

There are multiple ways to evaluate a classification model.

- Construct a **Confusion Matrix**: A confusion matrix is a tabular way of visualizing the performance of your prediction model. Each entry in a confusion matrix denotes the number of predictions made by the model where it classified the classes correctly or incorrectly. Target class is the Actual/True class - what really is!!! Output Class is the class predicted by the designed ML algorithm
- Receiver Operating Curve(ROC) Curve

- Area Under Curve(AUC) metrics.

Classification Evaluation Criteria

- **Accuracy:** It gives you the overall accuracy of the model, meaning the fraction of the total samples that were correctly classified by the classifier. To calculate accuracy, use the following formula: $(TP+TN)/(TP+TN+FP+FN)$.
- **Precision:** It tells you what fraction of predictions as a positive class were actually positive. To calculate precision, use the following formula: $TP/(TP+FP)$.
- **Recall:** It tells you what fraction of all positive samples were correctly predicted as positive by the classifier. It is also known as **True Positive Rate (TPR)**, *Sensitivity*, *Probability of Detection*. To calculate Recall, use the following formula: $TP/(TP+FN)$.
- **Sensitivity/Specificity:** It tells you what fraction of all negative samples are correctly predicted as negative by the classifier. It is also known as **True Negative Rate (TNR)**. To calculate specificity, use the following formula: $TN/(TN+FP)$.
- **F1-Score:** It combines precision and recall into a single measure. Mathematically it's the harmonic mean of precision and recall. It can be calculated as follows:

$$F_1\text{-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2TP}{2TP + FP + FN}$$

- **ROC/AUC:** A common approach to evaluate a model and the trade-off between *Recall (TPR)* and *Fallout(FPR)* as functions of the threshold value is to use the Receiver Operating Characteristic(ROC) curve and Area Under the Curve(AUC).

Confusion Matrix:

A confusion matrix is a tabular way of visualizing the performance of your prediction model. Each entry in a confusion matrix denotes the number of predictions made by the model where it classified the classes correctly or incorrectly. Target class is the Actual/True class - what really is!!! Output Class is the class predicted by the designed ML algorithm

- **True Positive (TP):** Number of predictions where the classifier correctly predicts the positive class as positive.
- **True Negative (TN):** Number of predictions where the classifier correctly predicts the negative class as negative.
- **False Positive (FP):** Number of predictions where the classifier incorrectly predicts the negative class as positive.
- **False Negative (FN):** Number of predictions where the classifier incorrectly predicts the positive class as negative.

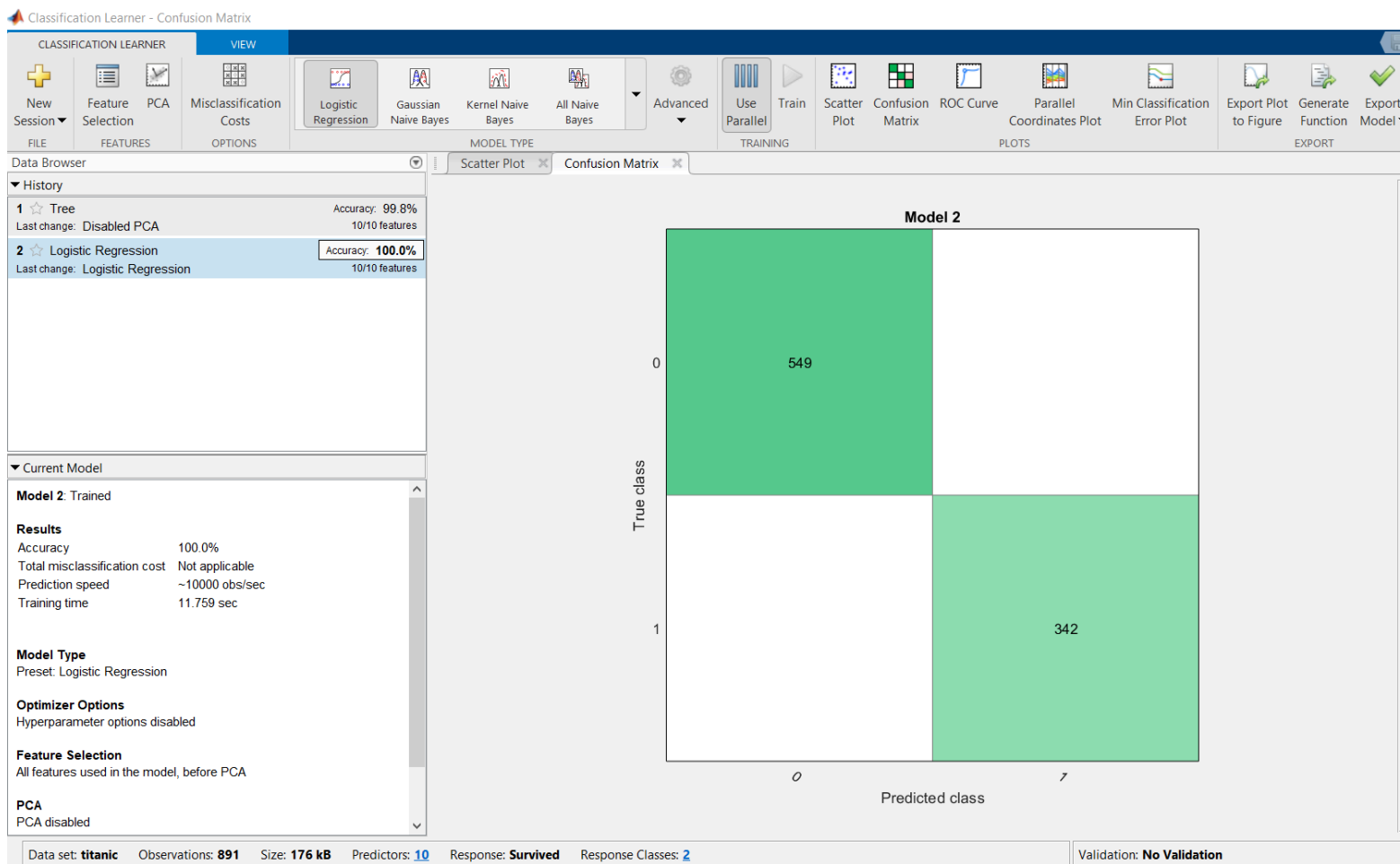
		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

Supervised Classification Using the Classification App

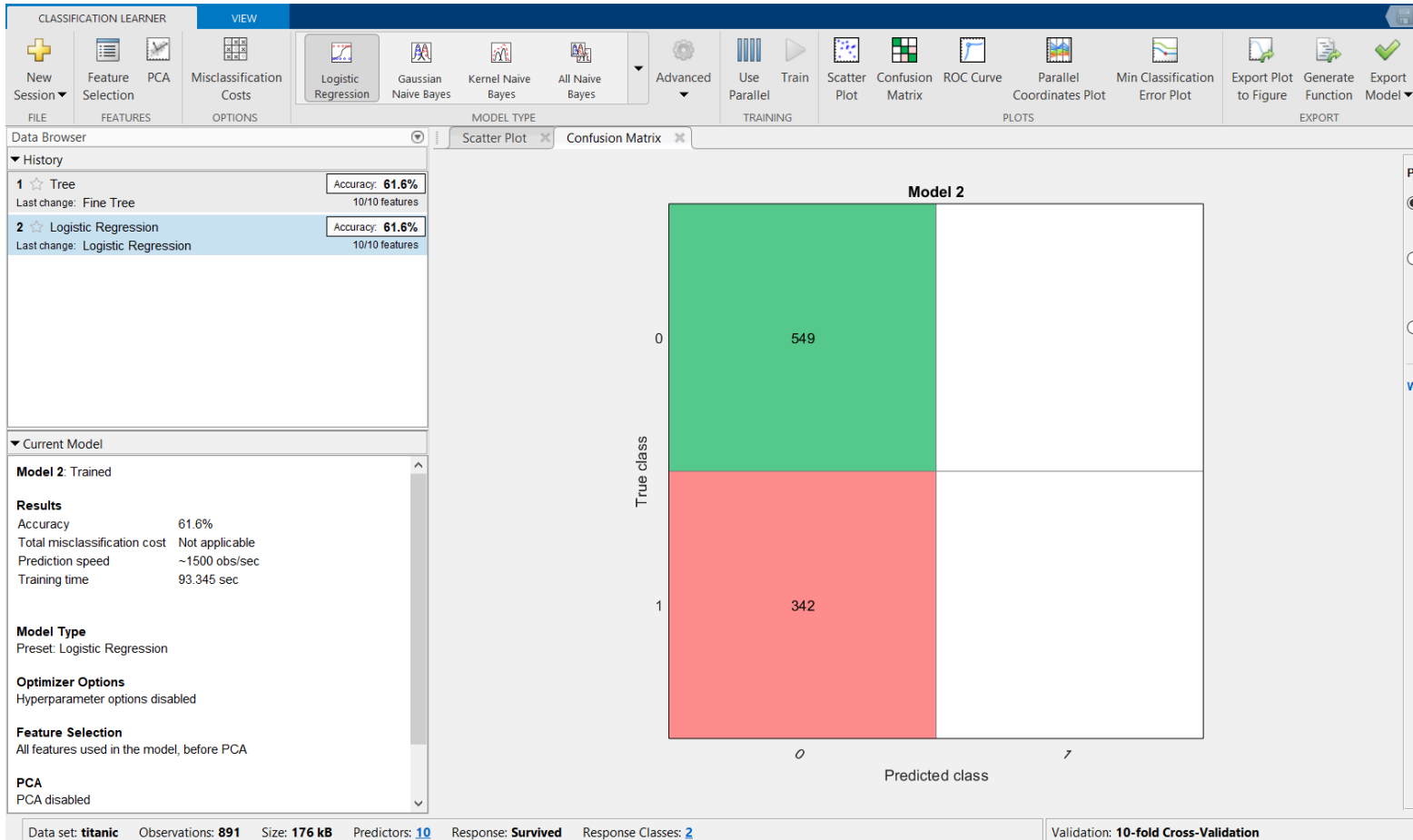
Demo using the Classification App

Logistic Regression

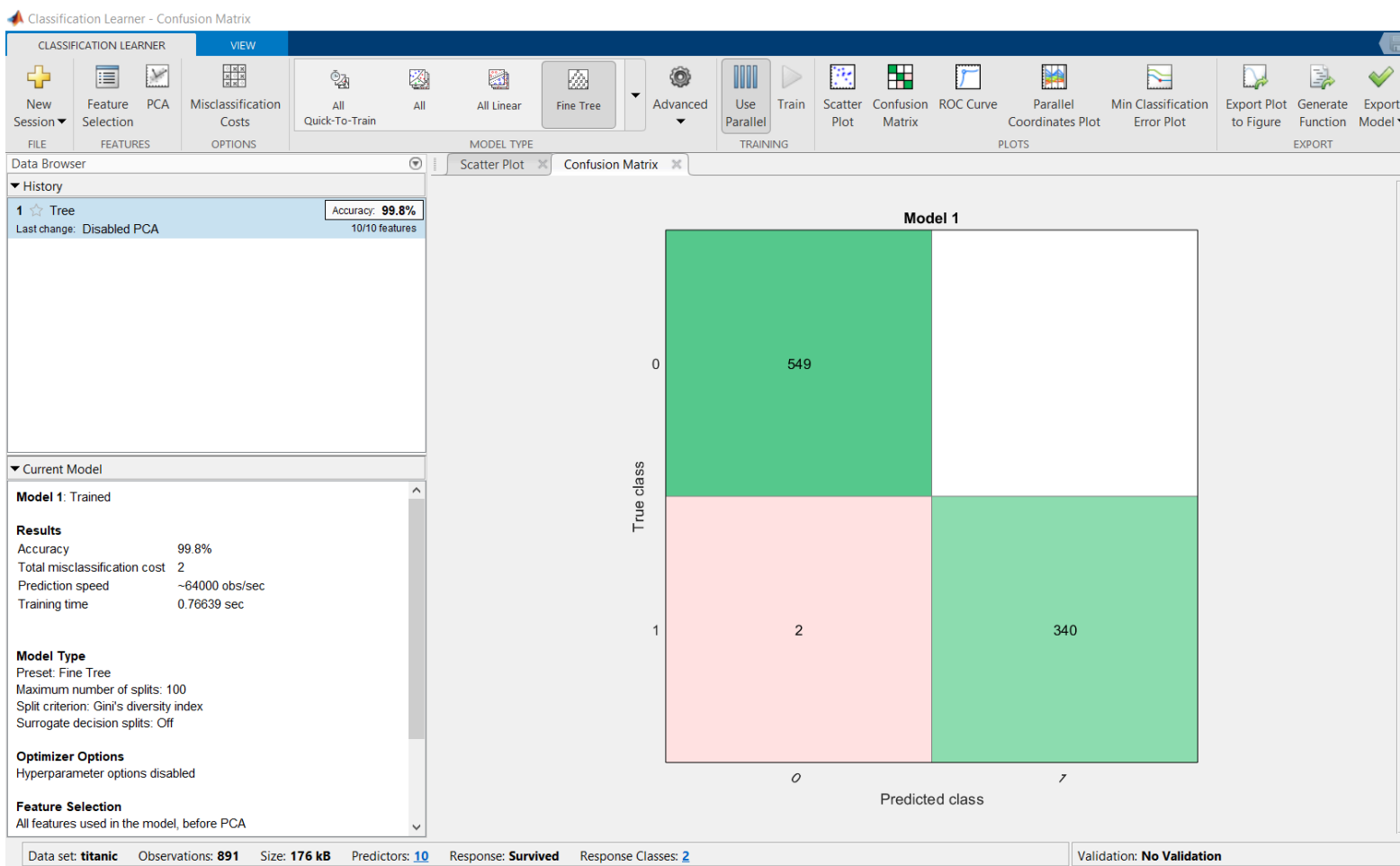
A logistic regression model is an equation where each predictor is multiplied by a coefficient and summed together. This sum becomes the argument to the logistic function to predict which of the two classes the observation belongs to. Response values above the specified threshold are in the positive class. Those below the threshold are assigned to the negative class. **Classification Threshold:** The threshold for classification can be set between 0 and 1. The Classification Learner App uses 0.5 as the threshold value. Use the `fitglm` function to specify a different value.



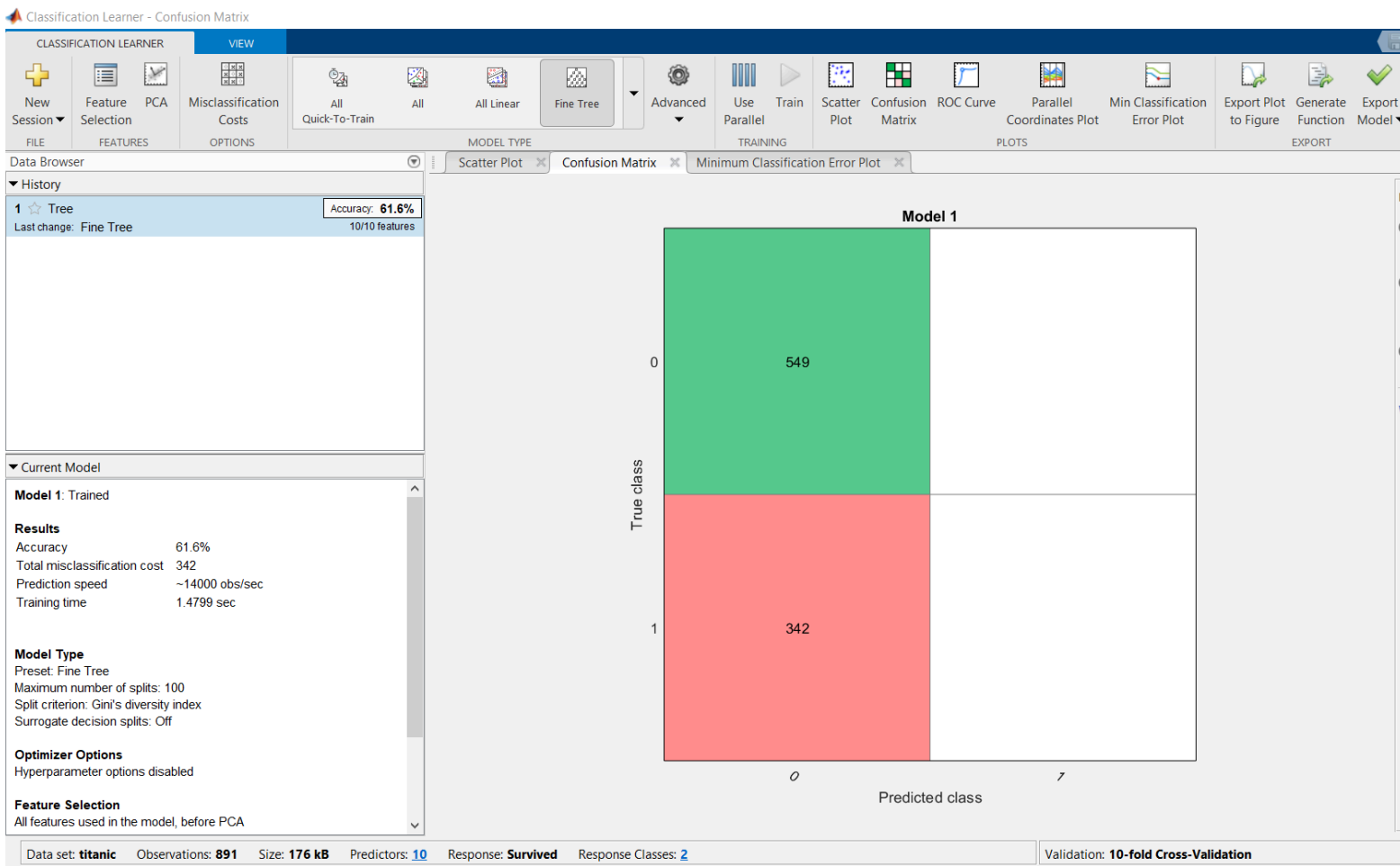
Linear Regression Model (with 10-fold cross validation)



(Fine) Decision Tree



DT with 10-fold CV



Solution??

Normalize/Standardize the data!!!!

```
% % Import the data
% titanic = readtable("C:\Users\Nabajeet Barman\Dropbox\MA6600\Workshop02-25Feb\titanic_preproc.mat")
% % To standardize data
% titanic.Age = (titanic.Age - mean(titanic.Age))/std(titanic.Age)
% % To normalize data
% titanic.Age = (titanic.Age - min(titanic.Age)) / (max(titanic.Age) - min(titanic.Age))
```

What is the Problem with Categorical Data?

Some algorithms can work with categorical data directly. For example, a **decision tree** can be learned directly from categorical data with no data transform required (this depends on the specific implementation).

Many machine learning algorithms cannot operate on label data directly. They require all input variables and output variables to be numeric. In general, this is mostly a constraint of the efficient implementation of machine learning algorithms rather than hard limitations on the algorithms themselves.

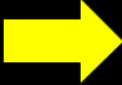
This means that *categorical data must be converted to a numerical form*. If the categorical variable is an output variable, you may also want to convert predictions by the model back into a categorical form in order to present them or use them in some application.

Integer Encoding

For example, “apple” is 1, “orange” is 2, and “banana” is 3. The integer values have a natural ordered relationship between each other and machine learning algorithms may be able to understand and harness this relationship.

One-hot encoding!!!!

For categorical variables where no such ordinal relationship exists, the integer encoding is not enough.



Color	Red	Yellow	Green
Red	1	0	0
Red	1	0	0
Yellow	0	1	0
Green	0	0	1
Yellow	0	0	1

When to use a Label Encoding vs. One Hot Encoding?

We apply One-Hot Encoding when:

- The categorical feature is not ordinal (Note: An **Ordinal** Number is a number that tells the position of something in a list, such as 1st, 2nd, 3rd, 4th, 5th etc.)
- The number of categorical features is less so one-hot encoding can be effectively applied (Note: if the number of categories are too many, one-hot encoding might not be suitable. In such cases, you should probably limit your algorithms to the ones which can work with either "labelled" categorical data and/or with integer encoding!!!)

We apply Label Encoding when:

- The categorical feature is ordinal (like Level 4, Level 5, Level 6 and Level 7 students)
- The number of categories is quite large as one-hot encoding can lead to high memory consumption

```
% Get unique categories and create indices:  
[categories, ~, index] = unique(titanic.Survived)
```

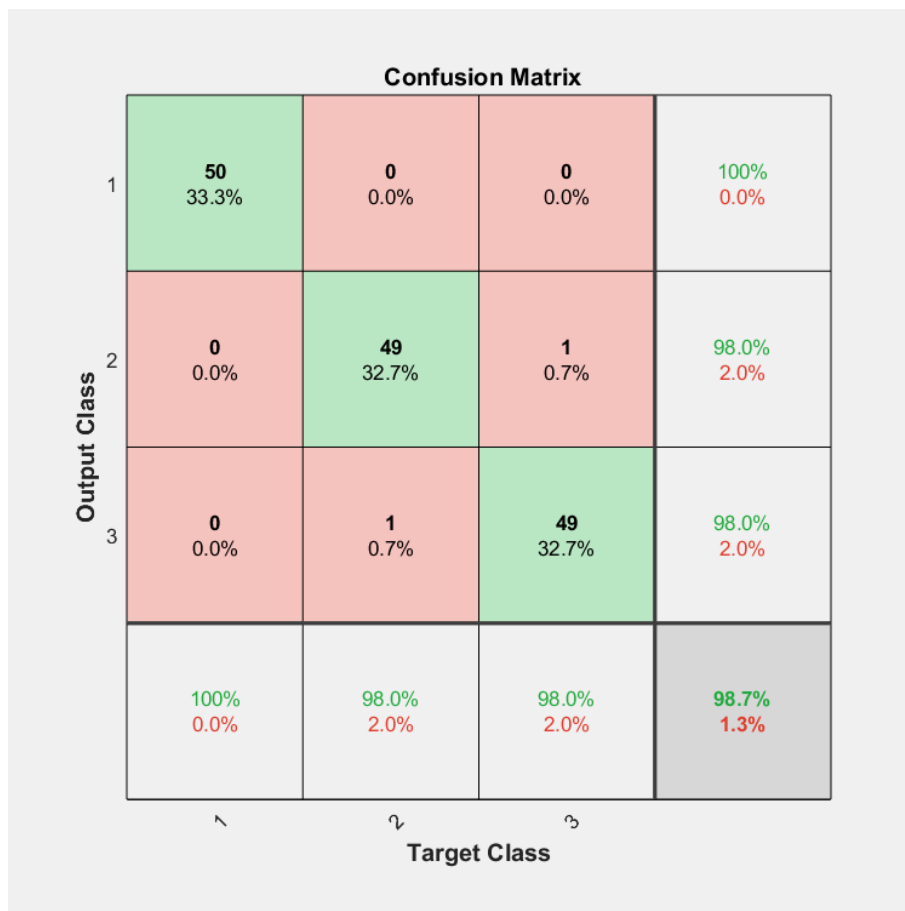
```
categories = 2x1 categorical array  
0  
1  
index = 891x1  
1  
2  
2  
2  
1  
1  
1  
1  
2  
2
```

⋮

```
% Create logical matrix:  
titanic_survived_one_hot = logical(accumarray([(1:numel(index)).' index], 1))
```

```
titanic_survived_one_hot = 891x2 logical array  
1 0  
0 1  
0 1  
0 1  
1 0  
1 0  
1 0  
1 0  
1 0  
0 1  
0 1  
⋮  
⋮
```

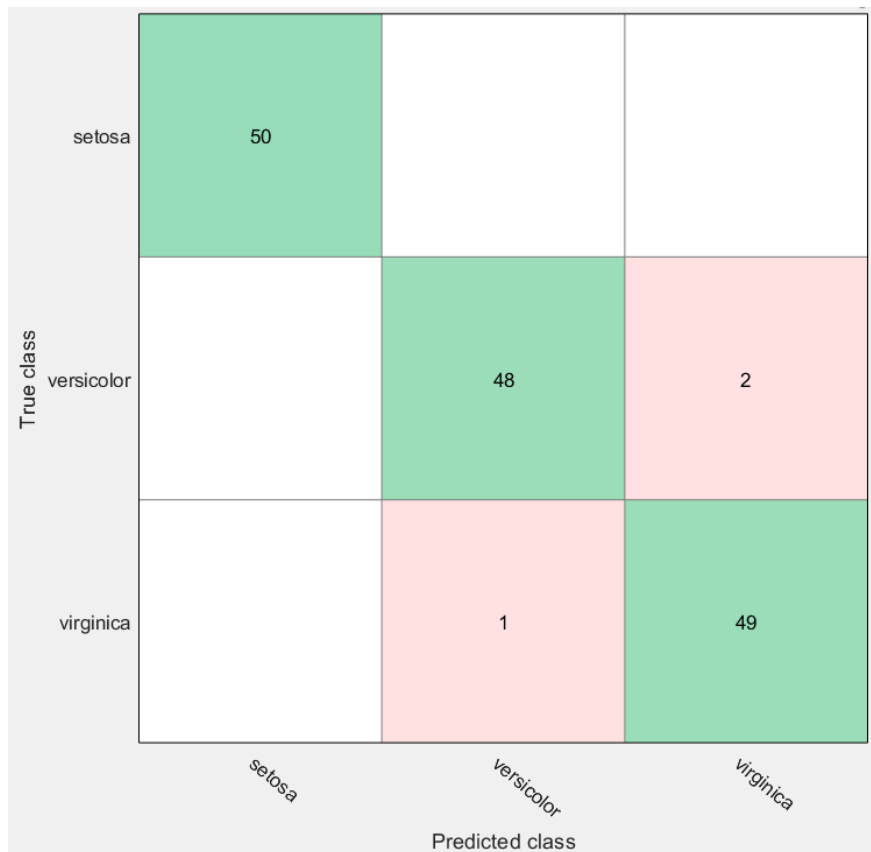
Unlike binary classification, there are no positive or negative classes here. At first, it might be a little difficult to find TP, TN, FP and FN since there are no positive or negative classes.



The **rows** correspond to the **predicted class (Output Class)** and the **columns** correspond to the **true class (Target Class)**. The diagonal cells correspond to observations that are correctly classified. The off-diagonal cells correspond to incorrectly classified observations. Both the number of observations and the percentage of the total number of observations are shown in each cell. The column on the far right of the plot shows the percentages of all the examples predicted to belong to each class that are correctly and incorrectly classified.

These metrics are often called the precision (or positive predictive value) and false discovery rate, respectively. The row at the bottom of the plot shows the percentages of all the examples belonging to each class that are correctly and incorrectly classified. These metrics are often called the recall (or true positive rate) and false negative rate, respectively. The cell in the bottom right of the plot shows the overall accuracy.

Important!!! The axis are transposed in the confusion matrix in the Classification Learner App



Supervised Classification Using the Classification Algorithms (by coding)

Model 01: Logistic Regression

A logistic regression model is an equation where each predictor is multiplied by a coefficient and summed together. This sum becomes the argument to the logistic function to predict which of the two classes the observation belongs to.

```
% Fit a Logistic Regression Classifier, predict and find accuracy using the confusion matrix
% we start with moving the predictor column, here Survived towards the end as the algorithm bel
titanic = titanic(:, [1 3:end 2]);
% fitglm(tbl) returns a generalized linear model fit to variables in the table or dataset array
% By default, fitglm takes the last variable as the response variable.
log_model = fitglm(titanic, 'Distribution', 'binomial');
```

Warning: Iteration limit reached.

Warning: The estimated coefficients perfectly separate failures from successes. This means the theoretical best estimates are not finite. For the fitted linear combination XB of the predictors, the sample proportions P of $Y=N$ in the data satisfy:

$XB < 3.60956e-12$: $P=0$

$XB > 3.60956e-12$: $P=1$

Warning: Regression design matrix is rank deficient to within machine precision.

```
ypred = predict(log_model,titanic(:,1:end-1));
ypred = double(round(ypred)); % to round the probabilities to 0 and 1 i.e. 0: not survived, 1:
confusion_matrix_LR = confusionmat(titanic.Survived,categorical(ypred));
accuracy_LR = trace(confusion_matrix_LR)/sum(confusion_matrix_LR, 'all')
```

```
accuracy_LR = 1
```

Recall: Using the app and No Validation setting, we achieved a similar performance measure (accuracy = 1)

Model 02: Decision Trees

```
% fitctree : Fit binary decision tree for multiclass classification
decision_tree = fitctree(titanic(:,1:end-1),titanic(:,end));
% Finding the accuracy of the decision tree model
label = predict(decision_tree,titanic(:,1:end-1));
confusion_matrix_tree = confusionmat(titanic.Survived,label);
accuracy_tree = trace(confusion_matrix_tree)/sum(confusion_matrix_tree, 'all')
```

```
accuracy_tree = 0.9978
```

Improving the Decision Tree's performance by pruning

When you grow a decision tree, consider its simplicity and predictive power. A deep tree with many leaves is usually highly accurate on the training data. However, the tree is not guaranteed to show a comparable accuracy on an independent test set. A leafy tree tends to overtrain (or overfit), and its test accuracy is often far less than its training (resubstitution) accuracy. In contrast, a shallow tree does not attain high training accuracy. But a shallow tree can be more robust — its training accuracy could be close to that of a representative test set. Also, a shallow tree is easy to interpret. If you do not have enough data for training and test, estimate tree accuracy by cross validation.

Pruning

Pruning optimizes tree depth (leafiness) by merging leaves on the same tree branch. [Control Depth or "Leafiness"](#) describes one method for selecting the optimal depth for a tree. Unlike in that section, you do not need to grow a new tree for every node size. Instead, grow a deep tree, and prune it to the level you choose.

```
% Improving the Decision Tree's performance by pruning
% We start again with fitting a decision tree and then viewing it
mytree = fitctree(titanic(:,1:end-1),titanic(:,end));

% Finding Best level for pruning using Misclassification error and cross-validation error. Prun
[~,~,~,BestLevel] = cvloss(mytree,'subtrees','all','treesize','min');
prunedtree = prune(mytree,'Level',BestLevel);
%view(prunedtree,'mode','graph')

% Finding the accuracy of the pruned decision tree model
```



```
label = predict(prunedtree,titanic(:,1:end-1));
confusion_matrix_tree_pruned = confusionmat(titanic.Survived,label);
accuracy_tree_pruned = trace(confusion_matrix_tree_pruned)/sum(confusion_matrix_tree_pruned, 'a')

accuracy_tree_pruned = 0.6162
```

How to choose the right algorithm?

(Kind of a hack) **The rule of thumb is the more complex the data, the more complex the algorithm.**

For tasks such as these, simple models such as LR, k-NN, etc. should work fine. For more complex data such as image and videos, consider using NN/CNN models.

Characteristics of Supervised Learning Algorithms

Algorithm	Predictive Accuracy	Fitting Speed	Prediction Speed	Memory Usage	Easy to Interpret	Handles Categorical Predictors
Trees	Low	Fast	Fast	Low	Yes	Yes
Boosted Trees	High	Medium	Medium	Medium	No	Yes
Bagged Trees	High	Slow	Slow	High	No	Yes
SVM	High	Medium	*	*	*	No
Naive Bayes	Low	**	**	**	Yes	Yes
Nearest Neighbor	***	Fast***	Medium	High	No	Yes***

* — SVM prediction speed and memory usage are good if there are few support vectors, but can be poor if there are many support vectors. When you use a kernel function, it can be difficult to interpret how SVM classifies data, though the default linear scheme is easy to interpret.

** — Naive Bayes speed and memory usage are good for simple distributions, but can be poor for kernel distributions and large data sets.

*** — Nearest Neighbor usually has good predictions in low dimensions, but can have poor predictions in high dimensions. For linear search, Nearest Neighbor does not perform any fitting. For *kd*-trees, Nearest Neighbor does perform fitting. Nearest Neighbor can have either continuous or categorical predictors, but not both.