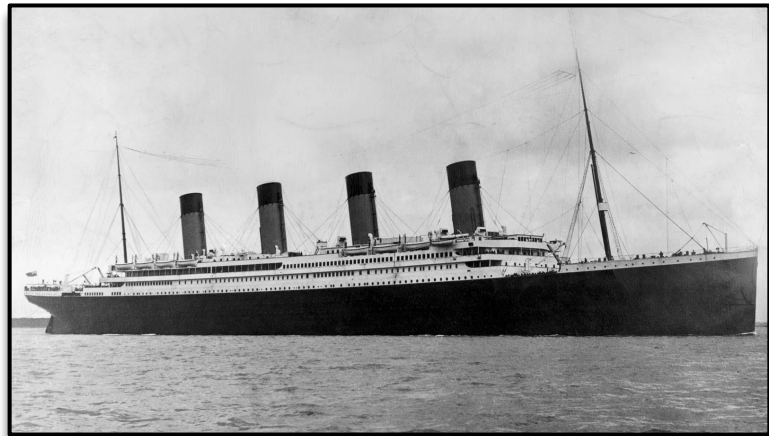# Titanic Survivors

Project Scope

An analysis of likelihood of survival for passengers aboard the Titanic

Source: Titanic Passenger Data from Data Flair*

1 **Titanic.csv**
- ○ 1 CSV file, 887 records
- ○ Passenger Information
    - ■ Survived Indicator
    - ■ Passenger Class
    - ■ Name
    - ■ Sex
    - ■ Age
    - ■ Siblings Aboard
    - ■ Parents Aboard
    - ■ Fare paid in Euros

Cleaning CSV Files
Sample code

1 Data cleaning was at multiple levels, to determine the probability of survival according to gender, class, relatives
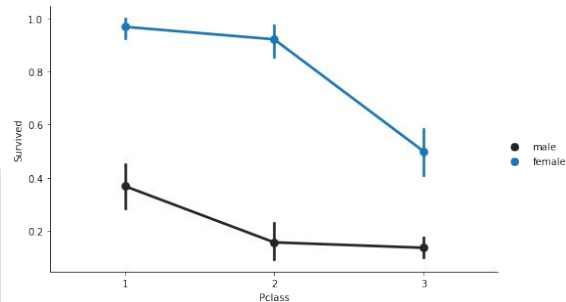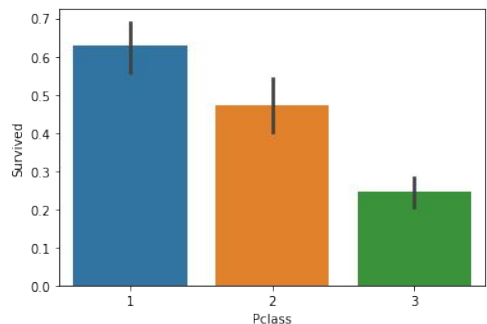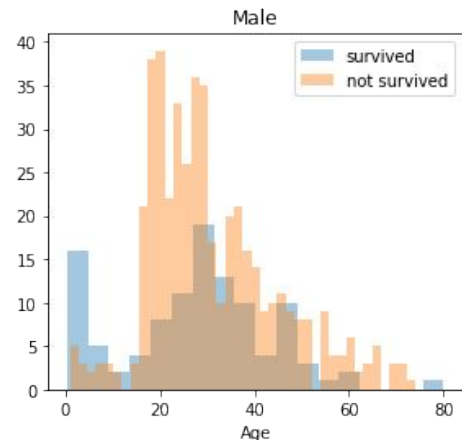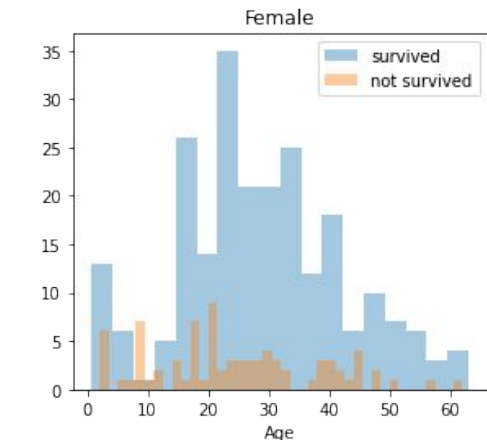
2 We stripped down the data and created several visualizations

3 The visualizations included:
- Histogram
- Facet Grid
- Bar Plot

Deploying Supervised Machine Learning

Running a random classifier model

1 We prepared for the ML model at the outset by renaming the sex column as 0, 1. We wanted to determine the ***probability of survival according to gender***

2 We ran a random classifier model on the titanic dataset

3 We created a "y" data set with just the 'Survived' column and an "X" data set by dropping Survived and Name

4 We compared the two data sets and ran a random classifier model to find the training and test score of survival according to gender

Troubleshooting: *we had to rename the sex column as a name on the female column was making it difficult to run the ML model*

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=1)
```

```
from sklearn.ensemble import RandomForestClassifier

clf = RandomForestClassifier(random_state=1, n_estimators=500).fit(X_train, y_train
)
print(f'Training Score: {clf.score(X_train, y_train)}')
print(f'Testing Score: {clf.score(X_test, y_test)}')
```

```
Training Score: 0.98796992481203
Testing Score: 0.7747747747747747
```

Creating a PostgreSQL database
Sample code

1 Initiate database and drop existing tables with desired table names

2 Create table for Titanic CSV file

3 Import relevant datasets with headers

4 Perform SELECT ALL query to ensure datasets imported correctly

Troubleshooting: *needed to rename Name column, removing slash from column names, in order for query to work.*

```
postgres/postgres@PostgreSQL 14 ⌄

Query Editor    Query History

1   -- Create tables
2   Drop table if exists titanic;
3   CREATE TABLE titanic(
4       Survived Int,
5       Pclass Int,
6       Pass_Name Text,
7       Sex Int,
8       Age Float,
9       Siblings_spouses_aboard Int,
10      Parents_child_aboard Int,
11      Fare Float
12  );
13  Select * from titanic;
14
```

Creating a PostgreSQL database
Screenshot example

1 Create query to investigate child's survival

```
18  -- Look at children info
19  Select * from titanic where Age < 12;
20  -- from the children who did not survive, all but 1 were in class 3
21  Select Pclass, Sex, Pass_name from titanic where Age < 12 and survived = 0;
22  -- look at female children who did not survive
23  Select Pclass, Sex, Pass_name from titanic where Age < 12 and survived = 0 and Sex = 1;
24  Select Pass_name, Age, Pclass from titanic where Age < 12 and survived = 0 and Sex = 1  order by age;
```

Data Output | Messages | Explain | Notifications

pclass integer	sex integer	pass_name text
10	3	0	Master. Eino Viljami Panula
11	3	0	Master. Arthur Rice
12	3	0	Master. Henry Forbes Lefebre
13	3	1	Miss. Constance Gladys Sage
14	3	0	Master. Clarence Gustaf Hugo Asplund
15	3	1	Miss. Telma Matilda Strom
16	3	0	Master. Eric Rice
17	1	1	Miss. Helen Loraine Allison
18	3	1	Miss. Stina Viola Palsson
19	3	0	Master. Sidney Leonard Goodwin

2 Query information on family survival

```
27  -- Look at family info
28  Select Pclass, Siblings_spouses_aboard, Parents_child_aboard from titanic where Parents_child_aboard > 0 and Siblings_spouses_aboard > 0;
29  Select Pclass, Siblings_spouses_aboard, Parents_child_aboard from titanic where Parents_child_aboard > 0 and Siblings_spouses_aboard > 0
30      order by Parents_child_aboard;
```

Data Output | Messages | Explain | Notifications

	pclass integer	siblings_spouses_aboard integer	parents_child_aboard integer
1	2	1	1
2	3	1	1
3	2	2	1
4	3	1	1
5	3	1	1
6	2	1	1
7	1	1	1
8	2	1	1

3 Investigate elderly survival rates

```
32  -- Look at elderly info
33  Select count(*) from titanic where Age > 60;
34  -- from the elderly that did not survive, all but 1 were Male
35  Select Pclass, Sex, Pass_name from titanic where Age > 60 and survived = 0;
36
```

Data Output | Messages | Explain | Notifications

	pclass integer	sex integer	pass_name text
12	1	0	Mr. Ramon Artagaveytia
13	3	0	Mr. James Webber
14	3	0	Mr. Samuel Beard Risien
15	1	0	Mr. Arthur Ernest Nicholson
16	1	0	Mr. George Wright
17	3	1	Miss. (Marion Ogden) Meanwell

Analytics using Tableau
Sample code

1 Upload Titanic CSV to Tableau

# titanic.csv Survived	# titanic.csv Pclass	Abc titanic.csv Name	Abc titanic.csv Sex	# titanic.csv Age	# titanic.csv Siblings/Spouses Aboard	# titanic.csv Parents/Children
0	3	Mr. Owen Harris Braund	male	22.0000	1	
1	1	Mrs. John Bradley (Florence B...	female	38.0000	1	
1	3	Miss. Laina Heikkinen	female	26.0000	0	
1	1	Mrs. Jacques Heath (Lily May ...	female	35.0000	1	
0	3	Mr. William Henry Allen	male	35.0000	0	

2 Set up binning for age and fare tables

Adjust dimension and measure tables

Edit Bins [Age]

New field name: Age (bin)

Size of bins: 5   Suggest Bin Size

Range of Values:

Min:	0.42	Diff:	79.58
Max:	80.00	CntD:	89

Cancel   OK

3 Create visualizations to view effect of:

- Passenger class on survival
- Gender on survival
- Age on survival
- Fare paid on survival

Marks

Automatic

Color   Size   Label

Detail   Tooltip

Survived
CNT(titanic.csv)
CNT(titanic... △

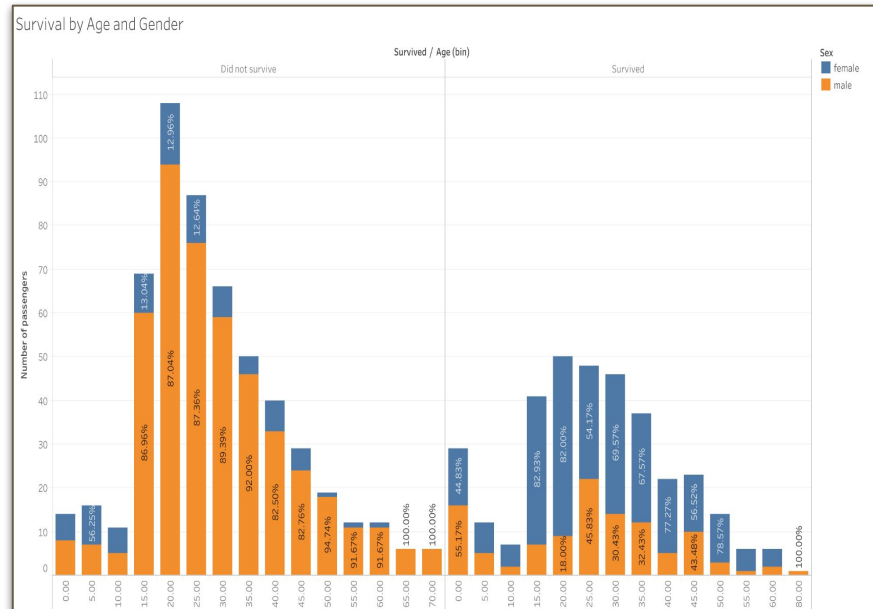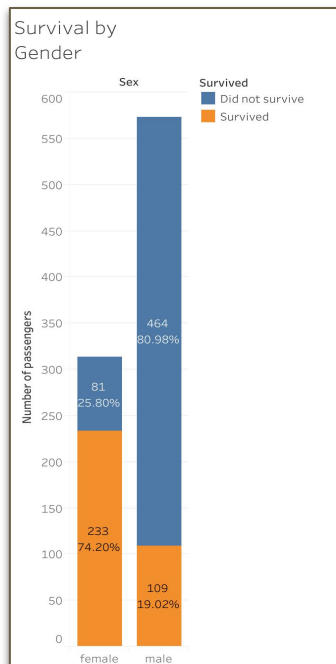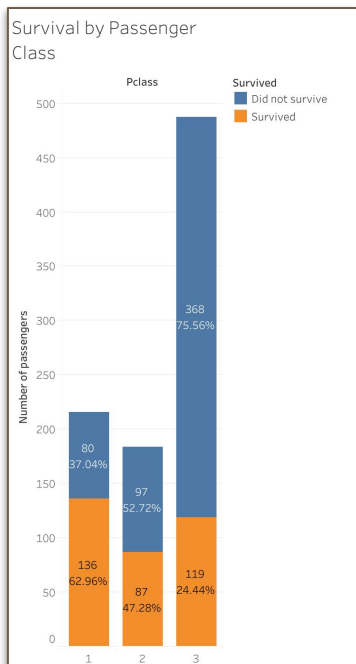| iii Columns	Sex
≡ Rows	CNT(titanic.csv)

Analytics using Tableau

Screenshot example

1 Passenger class 3 had 2x the deaths than class 1 and 2 combined

2 Where the majority of women survived, a greater majority of men did not survive

3 At all ages (except from 5-9) women were more likely to survive

Survival by Passenger Class

Survival by Gender

Survival by Age and Gender

Creating a Flask API
Sample code

1 Create Flask API with three routes to show cleaned dataset, ML train/test scores, and Tableau visualizations

2 Build accompanying HTML templates using Bootstrap components and stylesheets

```python
# main.py                                    ×
C: > Users > School > Desktop > nhakkarainen > titanic_survivors > ⬚ main.py > ...
 1    # Setup and import dependencies
 2    from flask import Flask, jsonify, render_template, redirect, render_template_string, url_for
 3    from sklearn.model_selection import train_test_split
 4    from sklearn.ensemble import RandomForestClassifier
 5    import pandas as pd
 6    import numpy as np
 7    import os
 8
 9    # Flask setup
10    app = Flask(__name__, template_folder="templates")
11
12    # Read data in CSV file
13    df = pd.read_csv("resources/titanic.csv")
14    df.to_csv("resources/titanic.csv", index = None)
15
16    # Flask routes
17    @app.route("/")
18    def Homepage():
19        """List all API routes."""
20        return render_template("index.html")
21
22    # APP ROUTE 1 - SHOW PASSENGER ROSTER
23    @app.route("/api/v1.0/dataset")
24    def dataset():
25        # Read CSV and convert to HTML table
26        data = pd.read_csv("resources/titanic.csv")
27        return render_template("table.html", tables = [data.to_html()], titles = [""])
28
29    # APP ROUTE 2 - SHOW PROBABILITY OF SURVIVAL
30    @app.route("/api/v1.0/survival")
31    def survival():
32        # Read File and store into Pandas data frame
33        titanic_df = pd.read_csv("resources/titanic.csv")
```

```html
# main.py        <> index.html ●    <> graphs.html ×
templates > ⬚ graphs.html > ...
 1    <!DOCTYPE html>
 2    <html lang="en">
 3
 4    <head>
 5        <meta charset="UTF-8">
 6        <title>Charts & Visualizations</title>
 7        <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.m
 8        <link rel="stylesheet" href="style.css">
 9    </head>
10    <body>
11        <div class="container">
12            <div class="jumbotron">
13                <h1 class="display-4">Charts and Visualizations</h1>
14                <p class="lead">A closer look at passenger features as an predictor of survival.</p>
15            </div>
16        </div>
17
18        <div class="container">
19            <div class="row">
20                <h2>Survival by Gender</h2>
21                <div class="thumbnail">
22                    <img src="{{url_for('static', filename='age_1.png')}}", class = "img-fluid"/>
23                </div>
24            </div>
25            <hr class="my-4">
26        </div>
27
28        <div class="container">
29            <hr class="my-4">
30            <div class="row">
31                <h2>Survival by Age & Gender</h2>
32                <div class="thumbnail">
```

Deploying app on Heroku
Screenshot examples

https://nu-p4-titanic.herokuapp.com/