# Machine Learning Engineer Nanodegree
## Capstone Project

### Haitham Alhad Hyder

### March 31, 2020

# 1 Definition

Note:*(approx. 1-2 pages)*

## Project Overview

The project involves the infamous Titanic sinkage of 1912. The data set and inspiration to solve the problem are from the Titanic challenge Kaggle competition (Kaggle n.d.).

Only 1502 of the 2224 on board the ship survived and was a big tragedy since it was labelled the "unsinkable" ship. We will be building a model that will identify the chance of survival a person has.

## 1.1 Problem Statement

We will be building a model that inputs the characteristics of a person and outputs a value between 0 and 1 giving us the chance of someone surviving.

The model that we come up with should be capable of identifying patterns that affect the chance of survival and therefore, by rounding the probability it outputs we can tell if that person's prediction is survival or not.

The steps to solving this problem are as follows:

1. Loading up the train data and split it into a train, validation and test data sets.

2. Build an XGBoost binary linear classifier as well as a PyTorch neural network.

3. At the same time we build an SKLearn decision tree which will be our Statistical base model that will help us compare it to our ML(Machine Learning) models.

4. After evaluating all the models and picking the best one, we will deploy to an API endpoint.

5. Finally we will create a web page that communicates with our model; sending in a user's chosen passenger characteristics and outputting the probability of survival.

## 1.2 Metrics

The main criteria that would help us evaluate the models we create is accuracy.

$$\text{accuracy} = \frac{\text{true positive} + \text{true negative}}{\text{dataset size}}$$

Since we are not extremely concerned with neither false negatives nor false positives, using accuracy as the chosen metric is much better than recall or precision. We only want to know out of the total inputs what percentage of them did we label correctly.

# 2 Analysis

Note: *(approx. 2-4 pages)*

## 2.1 Data Exploration

The train csv file which is the only one we will use since the test csv file, doesn't contain labels making it useless in evaluating our models. It contains data on 891 passengers.

| Data columns | | |
|---|---|---|
| **Variable** | **Definition** | **Key** |
| **Survived** | *Label* if a person survived or not | 0 = No; 1 = Yes |
| **Pclass** | Ticket class<br>*Proxy for socioeconomic status (SES)* | 1 = 1st (upper),<br>2 = 2nd (middle),<br>3 = 3rd (lower) |
| **Sex** | The gender of the person | |
| **Age** | The age in years | |
| **SibSp** | # of siblings and spouses aboard<br><br>*Note:* Sibling = (brother, sister, stepbrother, stepsister)<br>Spouse = (husband, wife) | |
| **Parch** | # Parents and children aboard<br><br>*Note:* Parent = (mother, father)<br>Child = (daughter, son, stepdaughter, stepson) | |
| **Fare** | Passenger fare | |
| Embarked | Port of embarkation | C = Cherbourg,<br>Q = Queenstown,<br>S = Southampton |
| PassengerId | A unique number for each person | |
| Name | The Name of the passenger | |
| Ticket | The unique ticket number | |
| Cabin | Cabin number | |

Table 1: The variables within the data set

The columns that are not bolded have various problems such as having many null values (e.g. Cabin) or not being relevant in solving the problem (e.g. PassengerId) and will not be used to build the features of the models.

|        | Survived   | Pclass     | Age        | SibSp      | Parch      | Fare       |
|--------|------------|------------|------------|------------|------------|------------|
| count  | 891.000000 | 891.000000 | 714.000000 | 891.000000 | 891.000000 | 891.000000 |
| mean   | 0.383838   | 2.308642   | 29.699118  | 0.523008   | 0.381594   | 32.204208  |
| std    | 0.486592   | 0.836071   | 14.526497  | 1.102743   | 0.806057   | 49.693429  |
| min    | 0.000000   | 1.000000   | 0.420000   | 0.000000   | 0.000000   | 0.000000   |
| 25%    | 0.000000   | 2.000000   | 20.125000  | 0.000000   | 0.000000   | 7.910400   |
| 50%    | 0.000000   | 3.000000   | 28.000000  | 0.000000   | 0.000000   | 14.454200  |
| 75%    | 1.000000   | 3.000000   | 38.000000  | 1.000000   | 0.000000   | 31.000000  |
| max    | 1.000000   | 3.000000   | 80.000000  | 8.000000   | 6.000000   | 512.329200 |

Table 2: Statistics of the data before cleaning and processing

Table 2 gives us interesting insights into the data. The mean of survived tells us that most people among the 891 didn't survive. Since the mean is much closer to 0 while the median is at 0.

The average age of the passengers we have is around 30 years old.

To process the data we first map the gender values to the following male = 0, female = 1. Then, only pick the columns we require and finally drop all the rows with null values.

## 2.2 Exploratory Visualization

In this section, you will need to provide some form of visualization that summarizes or extracts a relevant characteristic or feature about the data. The visualization should adequately support the data being used. Discuss why this visualization was chosen and how it is relevant. Questions to ask yourself when writing this section:

- *Have you visualized a relevant characteristic or feature about the data set or input data?*

- *Is the visualization thoroughly analyzed and discussed?*

- *If a plot is provided, are the axes, title, and datum clearly defined?*

## 2.3 Algorithms and Techniques

In this section, you will need to discuss the algorithms and techniques you intend to use for solving the problem. You should justify the use of each one based on the characteristics of the problem and the problem domain. Questions to ask yourself when writing this section:

- *Are the algorithms you will use, including any default variables/parameters in the project clearly defined?*

- *Are the techniques to be used thoroughly discussed and justified?*

- *Is it made clear how the input data or data sets will be handled by the algorithms and techniques chosen?*

## 2.4 Benchmark

In this section, you will need to provide a clearly defined benchmark result or threshold for comparing across performances obtained by your solution. The reasoning behind the benchmark (in the case where it is not an established result) should be discussed. Questions to ask yourself when writing this section:

- *Has some result or value been provided that acts as a benchmark for measuring performance?*

- *Is it clear how this result or value was obtained (whether by data or by hypothesis)?*

# 3 Methodology

Note: *(approx. 3-5 pages)*

## 3.1 Data Preprocessing

In this section, all of your preprocessing steps will need to be clearly documented, if any were necessary. From the previous section, any of the abnormalities or characteristics that you identified about the data set will be addressed and corrected here. Questions to ask yourself when writing this section:

- *If the algorithms chosen require preprocessing steps like feature selection or feature transformations, have they been properly documented?*

- *Based on the **Data Exploration** section, if there were abnormalities or characteristics that needed to be addressed, have they been properly corrected?*

- *If no preprocessing is needed, has it been made clear why?*

## 3.2 Implementation

In this section, the process for which metrics, algorithms, and techniques that you implemented for the given data will need to be clearly documented. It should be abundantly clear how the implementation was carried out, and discussion should be made regarding any complications that occurred during this process. Questions to ask yourself when writing this section:

- *Is it made clear how the algorithms and techniques were implemented with the given data sets or input data?*

- *Were there any complications with the original metrics or techniques that required changing prior to acquiring a solution?*

- *Was there any part of the coding process (e.g., writing complicated functions) that should be documented?*

### 3.3 Refinement

In this section, you will need to discuss the process of improvement you made upon the algorithms and techniques you used in your implementation. For example, adjusting parameters for certain models to acquire improved solutions would fall under the refinement category. Your initial and final solutions should be reported, as well as any significant intermediate results as necessary. Questions to ask yourself when writing this section:

- *Has an initial solution been found and clearly reported?*

- *Is the process of improvement clearly documented, such as what techniques were used?*

- *Are intermediate and final solutions clearly reported as the process is improved?*

### 3.4 Results

Note: *(approx. 2-3 pages)*

### 3.5 Model Evaluation and Validation

In this section, the final model and any supporting qualities should be evaluated in detail. It should be clear how the final model was derived and why this model was chosen. In addition, some type of analysis should be used to validate the robustness of this model and its solution, such as manipulating the input data or environment to see how the model's solution is affected (this is called sensitivity analysis). Questions to ask yourself when writing this section:

- *Is the final model reasonable and aligning with solution expectations? Are the final parameters of the model appropriate?*

- *Has the final model been tested with various inputs to evaluate whether the model generalizes well to unseen data?*

- *Is the model robust enough for the problem? Do small perturbations (changes) in training data or the input space greatly affect the results?*

- *Can results found from the model be trusted?*

### 3.6 Justification

In this section, your model's final solution and its results should be compared to the benchmark you established earlier in the project using some type of statistical analysis. You should also justify whether these results and the solution are significant enough to have solved the problem posed in the project. Questions to ask yourself when writing this section:

- *Are the final results found stronger than the benchmark result reported earlier?*

- *Have you thoroughly analyzed and discussed the final solution?*

- *Is the final solution significant enough to have solved the problem?*

# 4 Conclusion

Note: *(approx. 1-2 pages)*

## 4.1 Free-Form Visualization

In this section, you will need to provide some form of visualization that emphasizes an important quality about the project. It is much more free-form, but should reasonably support a significant result or characteristic about the problem that you want to discuss. Questions to ask yourself when writing this section:

- *Have you visualized a relevant or important quality about the problem, data set, input data, or results?*

- *Is the visualization thoroughly analyzed and discussed?*

- *If a plot is provided, are the axes, title, and datum clearly defined?*

## 4.2 Reflection

In this section, you will summarize the entire end-to-end problem solution and discuss one or two particular aspects of the project you found interesting or difficult. You are expected to reflect on the project as a whole to show that you have a firm understanding of the entire process employed in your work. Questions to ask yourself when writing this section:

- *Have you thoroughly summarized the entire process you used for this project?*

- *Were there any interesting aspects of the project?*

- *Were there any difficult aspects of the project?*

- *Does the final model and solution fit your expectations for the problem, and should it be used in a general setting to solve these types of problems?*

## 4.3  Improvement

In this section, you will need to provide discussion as to how one aspect of the implementation you designed could be improved. As an example, consider ways your implementation can be made more general, and what would need to be modified. You do not need to make this improvement, but the potential solutions resulting from these changes are considered and compared/contrasted to your current solution. Questions to ask yourself when writing this section:

- *Are there further improvements that could be made on the algorithms or techniques you used in this project?*

- *Were there algorithms or techniques you researched that you did not know how to implement, but would consider using if you knew how?*

- *If you used your final solution as the new benchmark, do you think an even better solution exists?*

---

**Before submitting, ask yourself. . .**

- Does the project report you've written follow a well-organized structure similar to that of the project template?

- Is each section (particularly **Analysis** and **Methodology**) written in a clear, concise and specific fashion? Are there any ambiguous terms or phrases that need clarification?

- Would the intended audience of your project be able to understand your analysis, methods, and results?

- Have you properly proof-read your project report to assure there are minimal grammatical and spelling mistakes?

- Are all the resources used for this project correctly cited and referenced?

- Is the code that implements your solution easily readable and properly commented?

- Does the code execute without error and produce results similar to those reported?

# References

Kaggle (n.d.). *Titanic: Machine Learning from Disaster.* URL: https://www.
    kaggle.com/c/titanic/overview.