## Assignment Details

Module Code: **ICS5110**

Module Name: **Applied Machine Learning**

Assignment Weighting: 80%

Report Authors

* **Gabriel Sammut (184395M)**
* **Nicholas Frendo**

Contents

[Assignment Details 1](#_Toc496878527)

[Section 1 – Introduction 1](#_Toc496878528)

[Chosen Dataset 2](#_Toc496878529)

[Machine Learning Techniques 3](#_Toc496878530)

[Section 2 – Background 4](#_Toc496878531)

[Terminology 4](#_Toc496878532)

[Support Vector Machine Classification 5](#_Toc496878533)

[Maximal Margin Classification 5](#_Toc496878534)

[Quantitative Measures 7](#_Toc496878535)

[Section 3 – Experiments 7](#_Toc496878536)

[Section 4 – Conclusions 7](#_Toc496878537)

[References 7](#_Toc496878538)

# 

# Section 1 – Introduction

This practical assignment looks into the utilisation of four machine learning techniques on a self-chosen data set. [1] However this report will specifically focus on the application of two of the implemented techniques. The objective of this study is to compare and monitor a selected number of machine learning algorithms by applying them to the afore mentioned dataset and reporting in detail the observed results. The report will feature an apt comparison between the performance and applicability of the machine learning algorithms as applied to the chosen dataset, as well as detailing each of the methods in a manner of detail. Furthermore the implemented algorithms were not picked under any particular bias, and merely implemented to detail the capabilities, distinction and applicability when applied to different datasets.

## Chosen Dataset

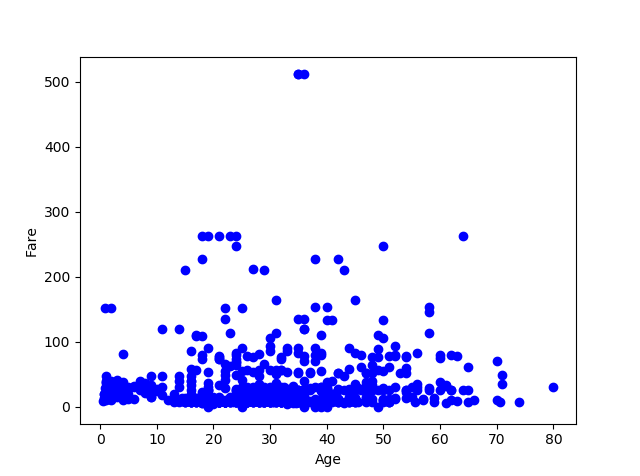
The tragedy known popularly as the sinking of the RMS Titanic is a very popular event. On April 15th of 1912, the Titanic sank after colliding with an iceberg, during the ship’s maiden voyage. Out of the 2224 aboard the vessel, a combination of 1502 crew and passengers were killed during this disastrous accident. Amongst the many reasons that resulted in the high fatality rate in the wake of the accident, was the lack of lifeboats available for the people on board. Partially considered an element of luck, some groups of people were more likely to survive than others, including but not limited to the women and children on board the ship, as well as those from the upper-class. [1]

Using this particular dataset allows us to analyse what sort of people were likely to survive the calamitous event. Composed of twelve data fields, the dataset consists of a total of 1309 candidates. Each candidate is summarised using the following descriptors:

|  |  |  |
| --- | --- | --- |
| **Variable** | **Definition** | **Key** |
| PassengerId | Passenger Identification |  |
| Survived | Survival | 0 = No, 1 = Yes |
| Pclass | Ticket class | 1 = 1st, 2 = 2nd, 3 = 3rd |
| Sex | Sex | 0 = Male, 1 = Female |
| Age | Age in years |  |
| Sibsp | # of siblings / spouses aboard the Titanic |  |
| Parch | # of parents / children aboard the Titanic |  |
| Ticket | Ticket number |  |
| Fare | Passenger fare (ticket cost) |  |
| Cabin | Cabin Number |  |
| Embarked | Port of embarkation | C = Cherbourg,  Q = Queenstown,  S = Southampton |

It is worth noting the following assumptions which were considered when applying our analysis on this particular dataset:

* **Pclass**: A proxy for socio-economic status (SES).
* **Age**: Age is a fractional value when less than one.
* **Sibsp**: The dataset defines family relationships as:
  + *Sibling –* brother, sister, stepbrother, stepsister.
  + *Spouse* – Husband, wife (Mistresses and fiancés are ignored).
* **Parch**: The dataset defines family relations as:
  + *Parent* – mother, father
  + *Child* – daughter, son, stepdaughter, stepson



***Figure 1:*** *A scatter plot representation of the Titanic Dataset, using features* ***Fare*** *and* ***Age***

## Machine Learning Techniques

A mixture of supervised and unsupervised techniques were utilised, totalling up to four machine learning algorithms being applied to the Titanic dataset:

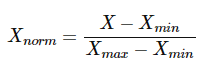
* Naïve Bayes Classification
* Support Vector Machine Classification

# Section 2 – Background

The following consists of a detailed explanation regarding the inner workings of the applied machine learning algorithms. However, a particular focus will be given to two of the implementations, leaving the other two to be used as reference and comparison methods. Before delving into the inner workings of these machine learning approaches, some terminology requires to be established:

## Terminology

* **Features**: A column of data to be utilised as the input criteria. The machine learning algorithm will be taught to identify patterns based on the features input into the system.
* **Labels**: A column of data to be utilised as the mapping criteria (or output criteria) to a particular feature set.
* **Hyperplane**: A separation (not necessarily linear) line in a (hyper) spatial domain, used to generalize different data groups.
* **Supervised Learning**: A method of identifying labels and mapping them to different inputs. This enables the ‘learning’ aspect of the algorithm as it learns to generalize better the more it is trained.
* **Normalization**: A data row operation that is applied to the chosen dataset, comprising of methods to make the data from all samples uniform, and relatable to each other. It is common to minimize the effect of feature dilution specifically aimed at minimizing outlier data points, making the term very important in the field of data science for this reason alone.
* **Rescaling**: Generally a linear normalization technique, rescaling of data points is about the linear scaling down of data. It can be considered as an alternative approach to Z-score standardization. In this approach, the data is usually scaled to a fixed range between 0 - 1. It is useful to constrain the data within a certain range (and therefore eliminating outliers), an example of which can be the Min-Max scaling equation:



Where:

* + Xmax = Maximum value of X
  + Xmin = Minimum value of X
  + Xnorm = Normalized value of X
* **Cross Validation**: It is common practice to diverge the dataset into two (sometimes even three) separate groups, one to be used for the training of the machine learning algorithm, and the remaining to utilise as a testing measure to gauge the accuracy of the classifier. It is also recommended to utilise three way data splits, essentially reserving a third category of data to be tested on classifier after the validation phase. Although cross validation is similar in many cases, it varies from one method to another:
  + **Random sub-sampling**: Randomly splits the dataset into K data segments, and repeat the algorithm classification for each of the folds. The final error estimate is calculated as an average of the individual segments.
  + **K-Fold Cross Validation**: Equal K-fold partitions in the dataset are reserved for validating the phase (similar to Random sub-sampling, but without the degree of randomness). Similarly, this method equates the average of the error estimate of each of the K-folds.
  + **Leave one out Cross Validation**: A subset case of the K-Fold approach, this method places K=1, and for each experiment uses N-1 examples for training and the remaining one for testing.
* **Feature Selection**: When applying the chosen dataset, it important to assess and know the data very well. It is worth noting that a dataset is composed of a number of features (columns, which denote the dataset). Some features will yield better correlations, others may not be so applicable. Feature selection speaks of a identifying which of the features are to be utilised for a particular classification problem.
* **Dimensionality Reduction**: Refers to the process of converting a set of data having a number of dimensions into a dataset with fewer dimensions, whilst ensuring that the dataset retains concise information. An example of this would be to project a four dimensional dataset down to three dimensions, to not only (potentially) obtain better features for classification and regression, but to also help visualize the dataset using a graph. Some methods of dimensionality reduction are seen through PCA (**Principal Component Analysis**) and/or Backward **Feature Elimination.** [7]

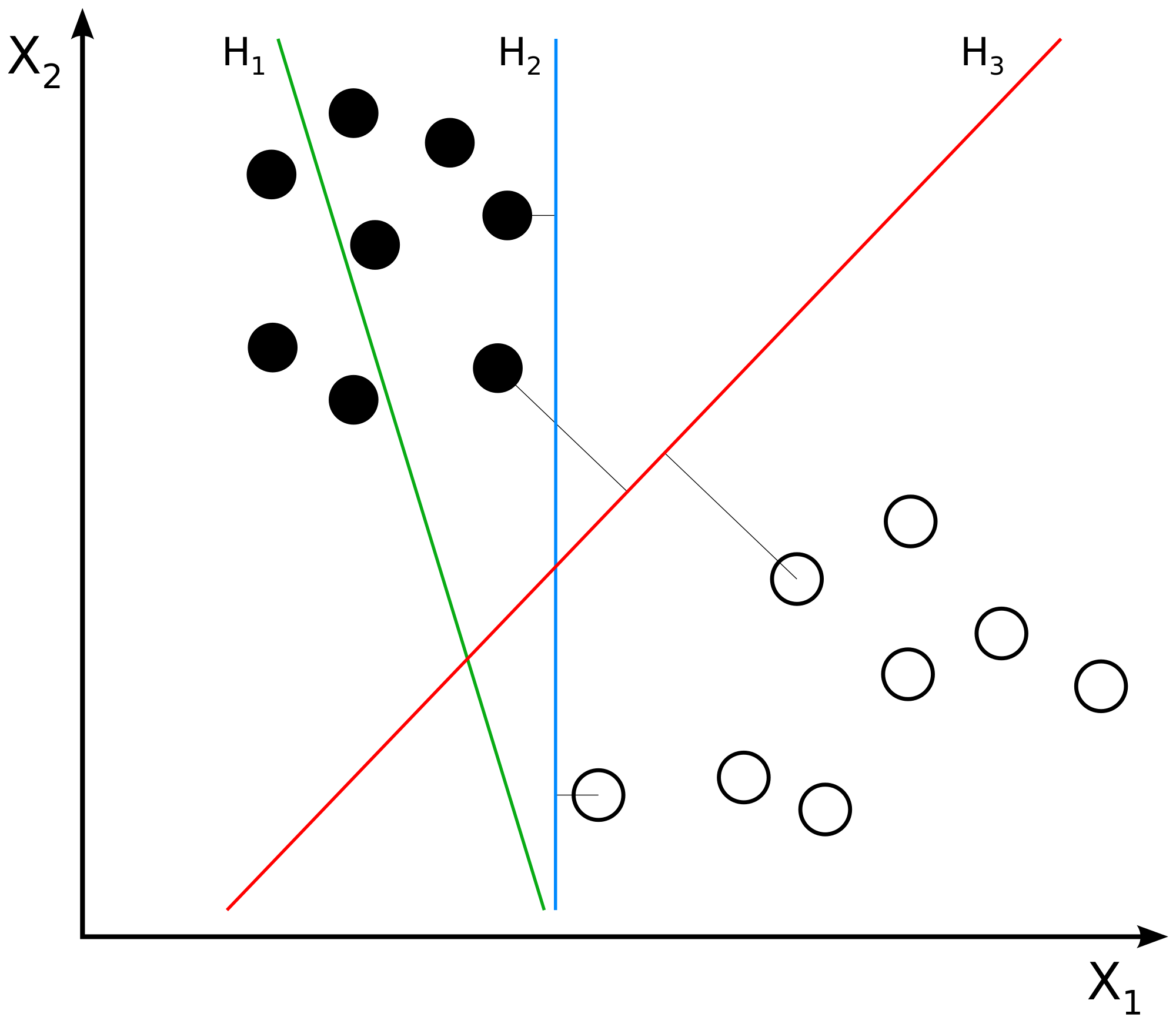
## Support Vector Machine Classification

Shortly referred to as a **SVM**, Support Vector Machine Classification is a supervised machine learning methodology used in the field of computer science and machine classification. It is a discriminative classifier formally defined by the separation of a hyperplane. [2] They were extremely important around the time they were developed in the 1990s and continue to be very useful today for particular problem domains. [3]

### Maximal Margin Classification

The hyperplane can be thought off as a line that separates and generalizes between a set of data. This is usually done using a pair of features (as carried out in this report), however it is possible to utilise SVMs across more than two dimensions. Intuitively, the further from the hyperplane our data points lie, the more confident the classifier is capable to generalize new features. [4]

With this assumption, it is therefore important to identify the most optimum position of the hyperplane. The distance between this plane and the nearest data point is formally known as the **Margin**.It is through balancing the margin values on each side of the hyperplane, and ensuring that the margin on each side of it is equal, that the most optimum position of the classifier is achieved. [4]



***Figure 2:*** *A visual example of a SVM classifier on a two dimensional space. The figure depicts different hyper planes, however it is the red marked hyper plane which is the most optimum one, due to the equidistant margins on each side of the classifier. [5]*

There are circumstances where the hyper plane cannot be fit perfectly as described above due to data disparity, and considered to be known informally as ‘unclean data’. This is where the SVMs become very capable at, being able to project onto an extra dimension; Moving from 2nd dimension to 3rd dimension, in what is known as **Kernel trick**.

A Kernel is actually a mathematical function, which effectively computes dot products in a higher dimensional space. By using kernels, we can implicitly transform datasets into a higher dimensional spatial space using no extra memory and minimal computation overhead (depending on the utilised Kernel). [6] There are many popular kernels which apply to under different criteria, the most common of Kernels being:

* Linear Kernels
* Polynomial Kernels
* Sigmoid Kernels
* RBF Kernels

Choosing and applying the correct Kernel is not a trivial task, as the best fit depends on the applicable dataset. No matter which Kernels are utilised, the Kernel parameters **gamma** and **C** still need to tuned accordingly in order to achieve optimum performance on the SVM classifier.

## Quantitative Measures

To calculate the effectiveness and compare each of the carried out algorithms between each other, a measure of how accurate each of the algorithm (with respect to the applied) data set is retrieved. Each of the algorithms has been exposed to K-Fold validation techniques, calculating the final accuracy score as an average of all the k-fold results. Each of the applied machine learning algorithms will be facing a binomial classification, required to generalize and label whether:

* The subject did not survive the Titanic catastrophe – *Represented as* ***0***
* The subject survived the Titanic catastrophe – *Represented as* ***1***

# Section 3 – Experiments

# Section 4 – Conclusions

# References

[1]"Titanic: Machine Learning from Disaster | Kaggle", *Kaggle.com*, 2017. [Online]. Available: https://www.kaggle.com/c/titanic/data. [Accessed: 26- Oct- 2017].

[2]"Introduction to Support Vector Machines — OpenCV 2.4.13.4 documentation", *Docs.opencv.org*, 2017. [Online]. Available: https://docs.opencv.org/2.4/doc/tutorials/ml/introduction\_to\_svm/introduction\_to\_svm.html. [Accessed: 27- Oct- 2017].

[3]J. Brownlee, "Support Vector Machines for Machine Learning - Machine Learning Mastery", *Machine Learning Mastery*, 2017. [Online]. Available: https://machinelearningmastery.com/support-vector-machines-for-machine-learning/. [Accessed: 27- Oct- 2017].

[4]"Support Vector Machines: A Simple Explanation", *Kdnuggets.com*, 2017. [Online]. Available: https://www.kdnuggets.com/2016/07/support-vector-machines-simple-explanation.html. [Accessed: 27- Oct- 2017].

[5]"File:Svm separating hyperplanes (SVG).svg - Wikimedia Commons", *Commons.wikimedia.org*, 2017. [Online]. Available: https://commons.wikimedia.org/wiki/File:Svm\_separating\_hyperplanes\_(SVG).svg. [Accessed: 27- Oct- 2017].

[6]"The Kernel Trick", *Eric-kim.net*, 2017. [Online]. Available: http://www.eric-kim.net/eric-kim-net/posts/1/kernel\_trick.html. [Accessed: 27- Oct- 2017].

[7]"Beginners Guide To Learn Dimension Reduction Techniques", *Analytics Vidhya*, 2017. [Online]. Available: https://www.analyticsvidhya.com/blog/2015/07/dimension-reduction-methods/. [Accessed: 27- Oct- 2017].