Machine Learning Documentation

Sunpreet Singh Chahal

Table of Contents:

1. [Foreword](#Foreword) (Pg. 3)
2. [Introduction](#Introduction) (Pg. 4 – Pg. 5)
   1. [Model Data](#ModelTraining) (Pg.4)
   2. [Machine Learning vs. Deep Leaning vs. Neural Networks](#MLvsDL) (Pg. 4 – Pg.5)
3. Machine Learning Dictionary (Pg. X – Pg. X)
4. [Machine Learning Models](#ML_Models) (Pg. X – Pg. X)

Foreword:

After spending some time on learning a handful of machine learning models, I realised that I had made progress on learning the code for each of the models, however, I made minimal progress on understanding the models well enough to where I could converse the results the model had produces.

I will still continue playing around with various models to develop my practical knowledge within this domain.

The end-goal of the document is to gain a deeper and more well-rounded understanding of machine learning.

*This document hasn’t been reviewed to be used as a professional reference, more so, just personal journaling of what I’ve learned so-far and for personal reference.*

***GitHub Repository - ML Model Overview & Documentation:***

[*SunpreetSChahal/MachineLearning-Learning: Indapendant ML Learning - Reference (github.com)*](https://github.com/SunpreetSChahal/MachineLearning-Learning)

***GitHub Repository - Separate Exemplar ML Projects:***

[*SunpreetSChahal/ML\_Practice\_Projs (github.com)*](https://github.com/SunpreetSChahal/ML_Practice_Projs)

Introduction:

A branch of artificial intelligence; machine learning uses data and algorithms to train computers to imitate how humans think. Using patterns within the data to produce hidden insights, gradually improving the accuracy of the models as new data comes in.

**Model Training:**

As data gets fed into the model, its accuracy improves – this is called training a model. Training your model is a key step in the process, this step allows you to, 1) see if your model is providing you your desired result and 2) if the model is getting increasingly accurate – by being able to find further patterns within the data. The training set must have a desired value or *target value*, which the model will aim to produce by finding patterns within the provided dataset. Your training set is your “gold standard” – you fit your model using this set.

Once fitted you feed your model a validation/test set. This is where you test how well you model has been fitted. This could be through splitting your entire data set into training and test sets or by having dedicated sets for each function. If splitting the data; do a split of 70:30 (70% of the data is for testing, whereas, 30% is for training.)

For example, you have a model which will predict the species of various animals. You train your model to take in various inputs of the animals’ features and get a desired result of what species each animal is. Once fitted, you then provide the model with random set of features and see whether the model produces an accurate result.

As big data grows year over year, it is no surprise that data scientists are in demand to be able to get business insight using machine learning models. The amount of data isn’t project to slow down anytime soon, so, this demand is set to climb at a consistent pace.

**Machine Learning vs. Deep Learning:**

Often used interchangeably, though are quite different. Machine learning is a sub-field of AI, Deep Learning is a sub-field of ML and Neural Networks are a sub-field of DL.

Machine learning has significant human intervention. We set the features of the models. With the case usually being that structured data is required.

*Deep learning* has the ability to eliminate most of the human interaction that traditional ML requires. Though deep learning models can benefit from using labelled data, unstructured data can still be used effectively. DL models can automatically find patterns within the input data, highlighting differentiations without human interference. In areas such as NLP, facial recognition and speech recognition, deep learning excels.

In this document I will go over:

1. Various machine learning models in detail, covering; **what the model is**, **the purpose of said model** and **where it could be applied in a real-life scenario**.
2. To **understand key parameters of machine learning models**; P-value, R-Squared, MAPE, MSE, F-Statistic etc. Rather than glossing over these variables, but to understand them and be able to extract *that* much more value from the model.
3. To cover the **mathematical theory behind machine learning** – This is very important, as understanding the theory behind this topic will allow me to **\*\* INCLUDE ANSWER SENT TO SHANGHUA/BRUNO/LEO – IN SNIP OR NOTEBOOK REGARDING THIS DOCUMENT (AT THE BOTTOM) \*\***

ML Model Dictionary:

In this section I’ll cover various important parameters and keywords relevant to machine leaning.

* **Underfitting Model**: A model that is too simple for a dataset that is too complex. Many data points are missed. For example, data is all over the place, not very linear, yet a linear regression model is used – this results in poor model accuracy.
* **Overfitting Model**: A model that has zero error. The model learns the training data very well, understanding its patterns completely, but fails to be able to generalize; won’t perform well against testing data. Doesn’t give you an idea, which ML is all about.

ML Models:

In this section I will cover various machine learning models. Each model will have comprehensive notes about; what the model is, what the model does and what its use case is.

*To find reference examples of each model, refer to the* [*foreword page*](#Foreword)*, where my relevant GitHub links can be found.*

There are three types of machine learning models: Supervised, Unsupervised & Reinforcement Learning. Detailed notes on each class of model can be found in their respective sections.

Diagram

Description automatically generated

**Supervised Learning Models:**

A model that uses labelled data to train the machine. The output(s) are known, the machine just needs to class the input accordingly.

For example, our data has two columns: experience (input) and salary (output). We can use a supervised model to predict; what our salary would be given *x* years of experience.

There are two sub-categories within supervised learning – Regression & Classification models.

**Regression Models:**

Regression models are models where the response variable is continuous.

* Time
* Money

**Linear Regression:**

Formula; y = mx + b

Linear regression is a model used to find the line of best fit for your data set. A line of best fit is, as its name suggests, a line that covers as many points as possible, and minimising any data points anomalous to the line.

Chart, scatter chart

Description automatically generated

In the example above, we look at a linear regression model that looks at Area (Square meters) various lots against the Price ($) of those lots. We fit a line a best fit and are able to see; “When a lot is *x* square meters, it’s price is *$x*.”

**Decision Trees:**

As the model’s name suggest, the shape of the model is of a tree. Starting from a *root* input and branching out based on the model’s features and eventually ending at the final node or the *leaves* of the tree. Each step that the tree takes is called a *node*. A new filter so-to-speak.

Though easy to implement, the accuracy of this model is questionable.

**Random Forest:**

Using ensemble learning (multiple learning algorithms), Random forest models build upon decision trees. Instead of a single tree, this model uses multiple – hence the name, Random *Forest*.

This model uses various, smaller samples of the same dataset and finds the mode of each result. For example, there are 4 trees, 3 produce the result “*2*” and 1 produces the result “*15*”. The final output would be *2*. The usage of a *popularity* contest between each tree results in improved predictive accuracy and minimizing over-fitting your model

\*\*Bootstrapped datasets\*\*

**Classification Models:**

Classifications models are models that put the response variable into a category/class.

* Vehicle type
* Email spam or not

Mathematical Theory Behind ML

**Linear Algebra:**

xxx