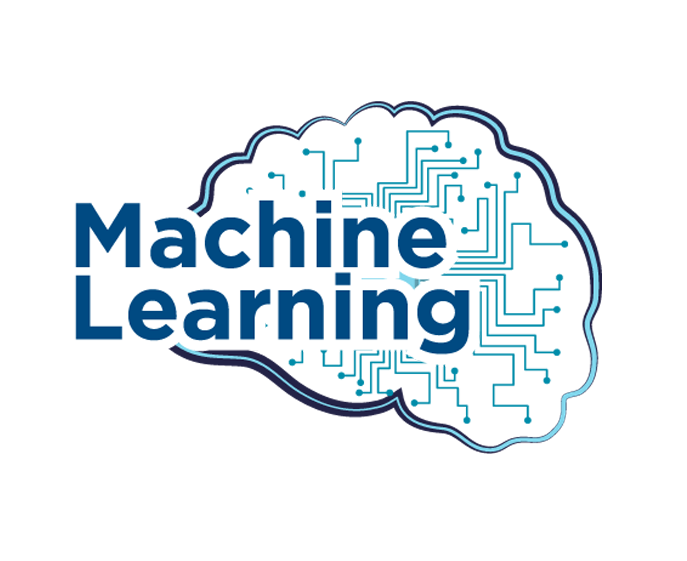
**Machine Learning Projects Documentation**



Rahul Sharma

[rahultechip@gmail.com](mailto:rahultechip@gmail.com)

<https://github.com/mr-rahul-sharma>

<https://www.linkedin.com/in/rahul-2/>

**Index**

1. Introduction
   1. Overview
   2. Purpose
2. Literature Survey
   1. Existing Problem
   2. Proposed Solution
3. Theoretical Analysis
   1. Block Diagram
   2. Algorithms
4. Result
5. Applications
6. Conclusion
7. Bibliography
8. **Introduction**
   1. **Overview**

Machine learning is a subfield of artificial intelligence (AI). The goal of machine learning generally is to understand the structure of data and fit that data into models that can be understood and utilized by people.

Because of this, machine learning facilitates computers in building models from sample data in order to automate decision-making processes based on data inputs.

Any technology user today has benefitted from machine learning. Facial recognition technology allows social media platforms to help users tag and share photos of friends. Optical character recognition (OCR) technology converts images of text into movable type. Recommendation engines, powered by machine learning, suggest what movies or television shows to watch next based on user preferences. Self-driving cars that rely on machine learning to navigate may soon be available to consumers.

Some of the Machine Learning projects are House Price Prediction, Churn Prediction, Loan Prediction, Default of Credit Card Clients Predictive Model, Clustering of Population Data etc. These projects predict specific column or groups the data into clusters. This can be very helpful because these implementations can tell us, How the data will turn into future or say, what will be this column value when we have some input values. These are implemented by different Machine Learning algorithms.

* 1. **Purpose**

Machine learning is important because it gives enterprises a view of trends in customer behavior and business operational patterns, as well as supports the development of new products. Many of today's leading companies, such as Facebook, Google and Uber, make machine learning a central part of their operations. Machine learning has become a significant competitive differentiator for many companies.

The machine learning field is continuously evolving. And along with evolution comes a rise in demand and importance. There is one crucial reason why data scientists need machine learning, and that is: ‘High-value predictions that can guide better decisions and smart actions in real-time without human intervention.’

Machine learning as technology helps analyze large chunks of data, easing the tasks of data scientists in an automated process and is gaining a lot of prominence and recognition. Machine learning has changed the way data extraction and interpretation works by involving automatic sets of generic methods that have replaced traditional statistical techniques.

1. **Literature Survey**
   1. **Existing Problem**

Machine learning algorithms are typically used in areas where the solution requires continuous improvement post-deployment. Adaptable machine learning solutions are incredibly dynamic and are adopted by companies across verticals.

In case of House Price Prediction Model, if the customers don’t know what will be house price with this area, height, location etc., they may suffer with loss. In Loan Prediction Model, if bank don’t know whether this customer will be able to return the loan or not, bank may be in loss.

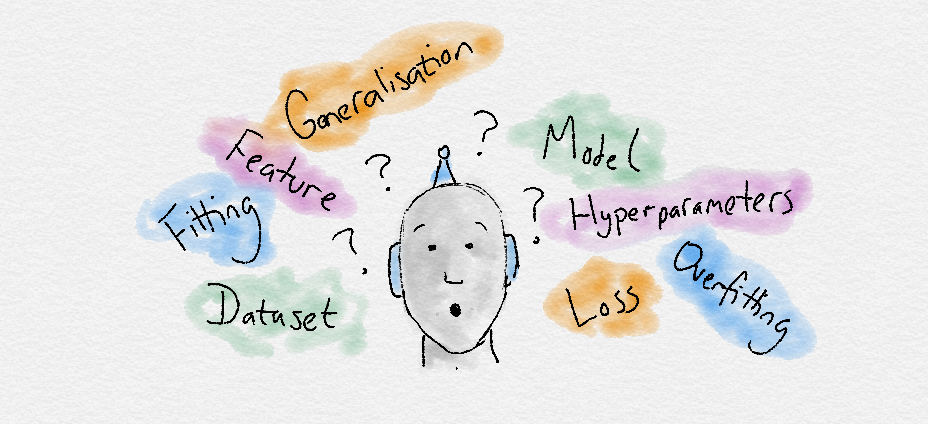
There are many fields whose problems can be solved by Machine Learning like education, research, share-market, agriculture etc. We never thought that before commencing from a place to reach the desired destination, we could check the exact status of traffic on that route. Or for that matter, ten years ago it was tough to believe that we can order food with just a few clicks! In fact, did you ever think about saying ‘Ok Google or Hey Siri’ and in return, somebody will speak to you and do as you want them to do!

We can’t deny the fact that our personal and professional life relies on the internet! Today we are all dependent upon technology. Almost a decade ago, we used to rely on all manual ways to fulfill our objectives and never imagined that in this era, we even think of machine learning applications.

* 1. **Proposed Solution**

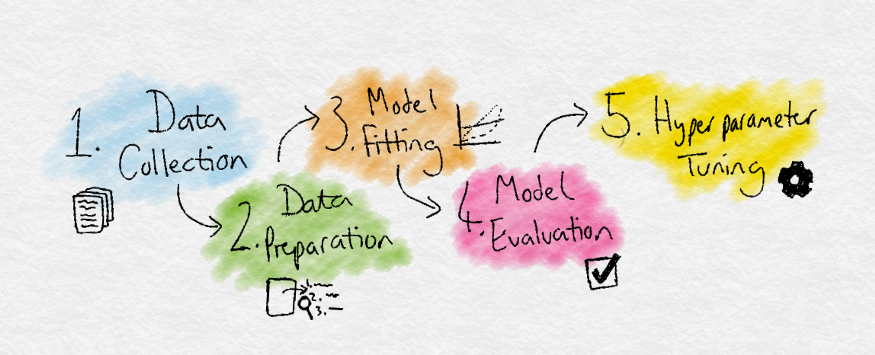
Machine learning is a tool for turning information into knowledge. In the past 50 years, there has been an explosion of data. This mass of data is useless unless we analyze it and find the patterns hidden within. Machine learning techniques are used to automatically find the valuable underlying patterns within complex data that we would otherwise struggle to discover. The hidden patterns and knowledge about a problem can be used to predict future events and perform all kinds of complex decision making.

**Terminology**



* Dataset: A set of data examples, that contain features important to solving the problem.
* Features: Important pieces of data that help us understand a problem. These are fed in to a Machine Learning algorithm to help it learn.
* Model: The representation (internal model) of a phenomenon that a Machine Learning algorithm has learnt. It learns this from the data it is shown during training. The model is the output you get after training an algorithm. For example, a decision tree algorithm would be trained and produce a decision tree model.

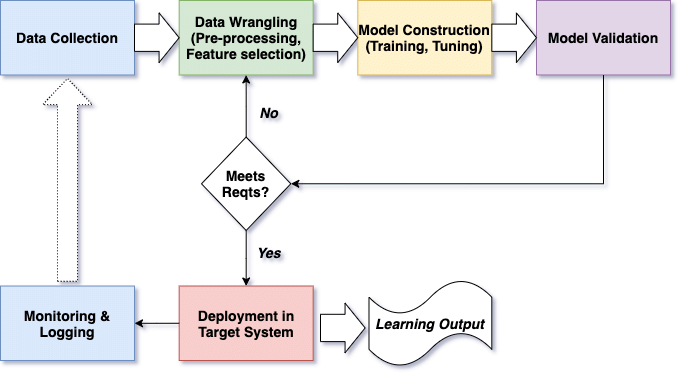
**Process**



* Data Collection: Collect the data that the algorithm will learn from.
* Data Preparation: Format and engineer the data into the optimal format, extracting important features and performing dimensionality reduction.
* Training: Also known as the fitting stage, this is where the Machine Learning algorithm actually learns by showing it the data that has been collected and prepared.
* Evaluation: Test the model to see how well it performs.
* Tuning: Fine tune the model to maximize its performance.

1. **Theoretical Analysis**
   1. **Block Diagram**

The Machine Learning works with different steps.



The components of a machine learning solution:

* Data Generation: Every machine learning application lives off data. That data has to come from somewhere. Usually, it’s generated by one of your core business functions.
* Data Collection: Data is only useful if it’s accessible, so it needs to be stored – ideally in a consistent structure and conveniently in one place.
* Feature Engineering Pipeline: Algorithms can't make sense of raw data. We have to select, transform, combine, and otherwise prepare our data so the algorithm can find useful patterns.
* Training: This is where the magic happens. We apply algorithms, and they learn patterns from the data. Then they use these patterns to perform particular tasks.
* Evaluation: We need to carefully test how well our algorithm performs on data it hasn’t seen before (during training). This ensures we don’t use prediction models that work well on “seen” data, but not in real-world settings.
* Task Orchestration: Feature engineering, training, and prediction all need to be scheduled on our compute infrastructure (such as AWS or Azure) – usually with non-trivial interdependence. So, we need to reliably orchestrate our tasks.
* Prediction: This is the moneymaker. We use the model we’ve trained to perform new tasks and solve new problems – which usually means making a prediction.
* Infrastructure: Even in the age of the cloud, the solution has to live and be served somewhere. This will require setup and maintenance.
* Authentication: This keeps our models secure and makes sure only those who have permission can use them.
* Interaction: We need some way to interact with our model and give it problems to solve. Usually this takes the form of an API, a user interface, or a command-line interface.‍
* Monitoring: We need to regularly check our model’s performance. This usually involves periodically generating a report or showing performance history in a dashboard.

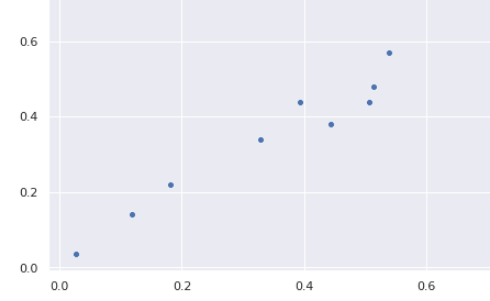
Machine learning solutions are used to solve a wide variety of problems, but in nearly all cases the core components are the same. Whether you simply want to understand the skeleton of machine learning solutions better or are embarking on building your own, understanding these components - and how they interact - can help.

* 1. **Algorithms**

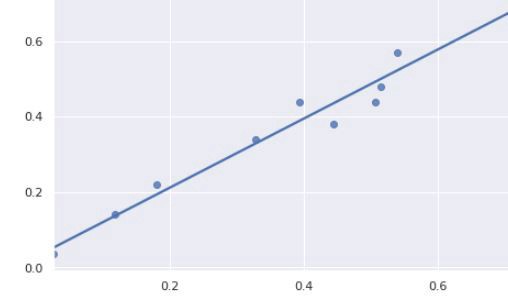
**Linear Regression**

Linear regression is a supervised learning algorithm and tries to model the relationship between a continuous target variable and one or more independent variables by fitting a linear equation to the data.

For a linear regression to be a good choice, there needs to be a linear relation between independent variable(s) and target variable. There are many tools to explore the relationship among variables such as scatter plots and correlation matrix. For example, the scatter plot below shows a positive correlation between an independent variable (x-axis) and dependent variable (y-axis). As one increases, the other one also increases.



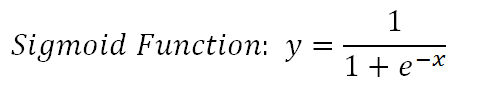
A linear regression model tries to fit a regression line to the data points that best represents the relations or correlations. The most common technique to use is ordinary-least squares (OLE). With this method, best regression line is found by minimizing the sum of squares of the distance between data points and the regression line. For the data points above, the regression line obtained using OLE seems like:



**Logistic Regression**

Logistic regression is a supervised learning algorithm which is mostly used for binary classification problems. Although “regression” contradicts with “classification”, the focus here is on the word “logistic” referring to logistic function which does the classification task in this algorithm. Logistic regression is a simple yet very effective classification algorithm so it is commonly used for many binary classification tasks. Customer churn, spam email, website or ad click predictions are some examples of the areas where logistic regression offers a powerful solution.

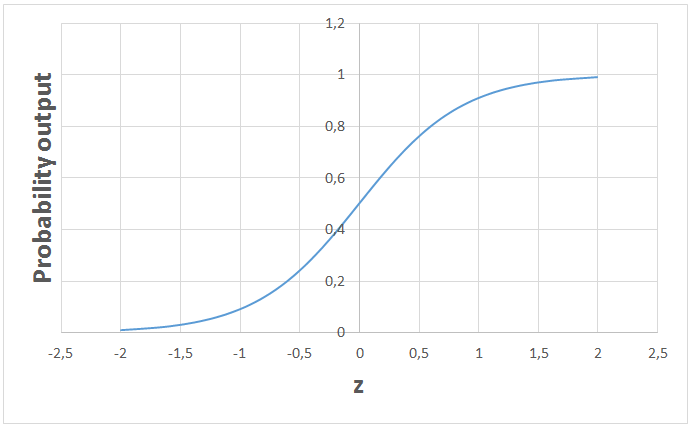
The basis of logistic regression is the logistic function, also called the sigmoid function, which takes in any real valued number and maps it to a value between 0 and 1.



Consider we have the following linear equation to solve:



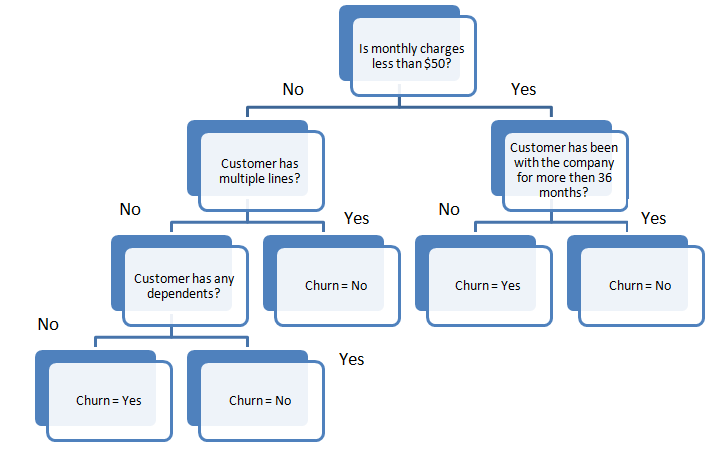
Logistic regression model takes a linear equation as input and uses logistic function and log odds to perform a binary classification task. Then, we will get the famous s shaped graph of logistic regression:



We can use the calculated probability as is. For example, the output can be “the probability that this email is spam is 95%” or “the probability that customer will click on this ad is 70%”. However, in most cases, probabilities are used to classify data points. For instance, if the probability is greater than 50%, the prediction is positive class (1). Otherwise, the prediction is negative class (0).

**Decision Trees**

A decision tree builds upon iteratively asking questions to partition data. It is easier to conceptualize the partitioning data with a visual representation of a decision tree:



This represents a decision tree to predict customer churn. First split is based on monthly charges amount. Then the algorithm keeps asking questions to separate class labels. The questions get more specific as the tree gets deeper.

The aim of the decision tree algorithm is to increase the predictiveness as much as possible at each partitioning so that the model keeps gaining information about the dataset. Randomly splitting the features does not usually give us valuable insight into the dataset. Splits that increase purity of nodes are more informative. The purity of a node is inversely proportional to the distribution of different classes in that node. The questions to ask are chosen in a way that increases purity or decrease impurity.

How many questions do we ask? When do we stop? When is our tree sufficient to solve our classification problem? The answer to all these questions leads us to one of most important concepts in machine learning: overfitting. The model can keep asking questions until all the nodes are pure. However, this would be a too specific model and would not generalize well. It achieves high accuracy with training set but performs poorly on new, previously unseen data points which indicates overfitting. The depth of a tree is controlled by “max\_depth” parameter for decision tree algorithm in scikit-learn.

Decision tree algorithm usually does not require to normalize or scale features. It is also suitable to work on a mixture of feature data types (continuous, categorical, binary). On the negative side, it is prone to overfitting and needs to be ensembled in order to generalize well.

**Random Forest**

Random forest is an ensemble of many decision trees. Random forests are built using a method called bagging in which decision trees are used as parallel estimators. If used for a classification problem, the result is based on majority vote of the results received from each decision tree. For regression, the prediction of a leaf node is the mean value of the target values in that leaf. Random forest regression takes mean value of the results from decision trees.

Random forests reduce the risk of overfitting and accuracy is much higher than a single decision tree. Furthermore, decision trees in a random forest run in parallel so that the time does not become a bottleneck.

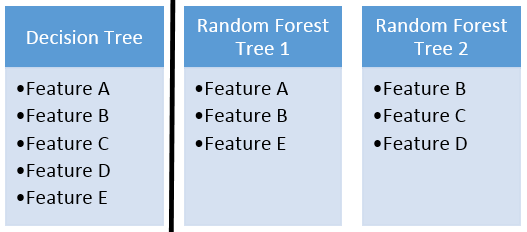
The success of a random forest highly depends on using uncorrelated decision trees. If we use same or very similar trees, the overall result will not be much different than the result of a single decision tree. Random forests achieve to have uncorrelated decision trees by bootstrapping and feature randomness.

Bootstrapping is randomly selecting samples from training data with replacement. They are called bootstrap samples.



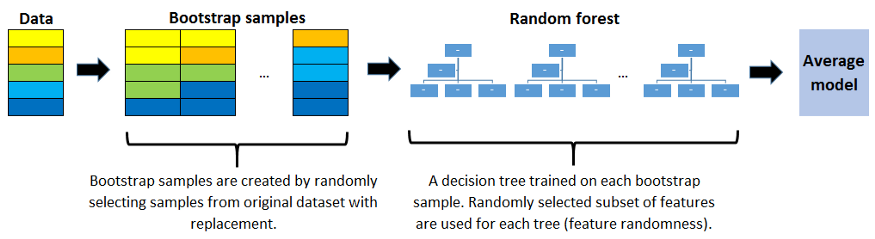
Bootstrap samples

Feature randomness is achieved by selecting features randomly for each decision tree in a random forest. The number of features used for each tree in a random forest can be controlled with “max\_features” parameter.



Feature randomness

Random forest is a highly accurate model on many different problems and does not require normalization or scaling. However, it is not a good choice for high-dimensional data sets (i.e., text classification) compared to fast linear models (i.e., Naive Bayes).



**K-Means Clustering**

Clustering is a way to group a set of data points in a way that similar data points are grouped together. Therefore, clustering algorithms look for similarities or dissimilarities among data points. Clustering is an unsupervised learning method so there is no label associated with data points. Clustering algorithms try to find the underlying structure of the data.

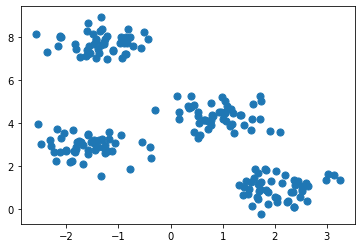
Clustering is not classification.

Observations (or data points) in a classification task have labels. Each observation is classified according to some measurements. Classification algorithms try to model the relationship between measurements (features) on observations and their assigned class. Then the model predicts the class of new observations.

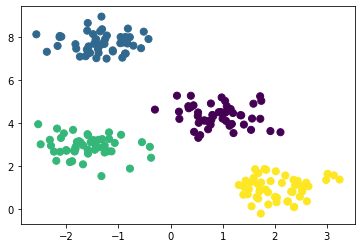
K-means clustering aims to partition data into k clusters in a way that data points in the same cluster are similar and data points in the different clusters are farther apart. Thus, it is a partition-based clustering technique. Similarity of two points is determined by the distance between them.

K-means clustering tries to minimize distances within a cluster and maximize the distance between different clusters. K-means algorithm is not capable of determining the number of clusters. We need to define it when creating the K-Means object which may be a challenging task.

Consider the following 2D visualization of a dataset:



It can be partitioned into 4 different clusters as below:



Real life datasets are much more complex in which clusters are not clearly separated. However, the algorithm works in the same way. K-means is an iterative process. It is built on expectation-maximization algorithm. After number of clusters are determined, it works by executing the following steps:

* Randomly select centroids (center of cluster) for each cluster.
* Calculate the distance of all data points to the centroids.
* Assign data points to the closest cluster.
* Find the new centroids of each cluster by taking the mean of all data points in the cluster.
* Repeat steps 2,3 and 4 until all points converge and cluster centers stop moving.

K-Means clustering is relatively fast and easy to interpret. It is also able to choose the positions of initial centroids in a smart way that speeds up the convergence.

One challenge with k-means is that number of clusters must be pre-determined. K-means algorithm is not able to guess how many clusters exist in the data. If there is a non-linear structure separating groups in the data, k-means will not be a good choice.

1. **Result**

After preprocessing data and applying machine learning algorithms, these models can predict required targets with great accuracy.

The projects are:

* House Price Prediction
* Churn Prediction
* Credit Card Clients Prediction
* Loan Prediction
* Population Data Clustering

To get the project files, go to the below GitHub repository link.

<https://github.com/mr-rahul-sharma/ML-Internshala-Trainings>

1. **Applications**

There are many applications of machine learning which are very helpful for our society.

1. Social Media Features

Social media platforms use machine learning algorithms and approaches to create some attractive and excellent features. For instance, Facebook notices and records your activities, your chats, likes, and comments, and the time you spend on specific kinds of posts. Machine learning learns from your own experience and makes friends and page suggestions for your profile.

2. Product Recommendations

One of the most popular and known applications of machine learning is Product Recommendation. Product recommendation is one of the stark features of almost every e-commerce website today, which is an advanced application of machine learning techniques. Using machine learning and AI, websites track your behavior based on your previous purchase, your searching pattern, your cart history, and make product recommendations.

3. Image Recognition

Image Recognition is one of the most significant and notable Machine Learning and AI techniques: an approach for cataloging and detecting a feature or an object in the digital image. This technique is being adopted for further analysis, such as pattern recognition, face detection, or face recognition.

4. Sentiment Analysis

Sentiment analysis is one of the most necessary Applications of Machine Learning. Sentiment analysis is one of the most necessary Applications of Machine Learning. Sentiment analysis is a real-time machine learning application that determines the emotion or opinion of the speaker or the writer. For instance, if someone has written a review or email (or any form of a document), a sentiment analyzer will instantly find out the actual thought and tone of the text. This sentiment analysis application can are used to analyze a review-based website, decision-making applications, etc.

5. Automating Employee Access Control

Organizations are actively implementing machine learning algorithms to determine the level of access employees would need in various areas, depending on their job profiles. This is one of the coolest applications of Machine Learning.

6. Marine Wildlife Preservation

Machine learning algorithms are used to develop behavior models for endangered cetaceans and other marine species, helping scientists regulate and monitor their populations.

7. Regulating Healthcare Efficiency and Medical Services

Significant healthcare sectors are actively looking at using Machine Learning algorithms to manage better. They predict the waiting times of patients in the emergency waiting rooms across various departments of hospitals. The models use vital factors that help define the algorithm, details of staff at various times of day, records of patients, and complete logs of department chats and the layout of emergency rooms. Machine learning algorithms also come to play when detecting a disease, therapy planning, and prediction of the disease situation. This is one of the most necessary Applications of Machine Learning.

8. Predict Potential Heart Failure

An algorithm designed to scan a doctor’s free-form e-notes and identify patterns in a patient’s cardiovascular history is making waves in medicine. Instead of a physician digging through multiple health records to arrive at a sound diagnosis, redundancy is now reduced with computers making an analysis based on available information.

9. Banking Domain

Banks are now using the latest advanced technology machine learning has to offer to help prevent fraud & protect accounts from hackers. The algorithms determine what factors to consider to create a filter to keep harm at bay. Various sites that are unauthentic will be automatically filtered out and restricted from initiating transactions.

10. Language Translation

One of the most common applications of Machine Learning is Language Translation. Machine learning plays a significant role in the translation of one language to another. We are amazed at how the websites can translate from one language to another effortlessly and gives contextual meaning as well. The technology behind the translation tool is called ‘machine translation.’ It has enabled the world to interact with people from all corners of the world; without it, life would not be as easy as it is now. It has provided a sort of confidence to travelers and business associates to safely venture into foreign lands with the conviction that language will no longer be a barrier.

1. **Conclusion**

Machine Learning can be a Supervised or Unsupervised. If you have lesser amount of data and clearly labelled data for training, opt for Supervised Learning. Unsupervised Learning would generally give better performance and results for large data sets. If you have a huge data set easily available, go for deep learning techniques. You also have learned Reinforcement Learning and Deep Reinforcement Learning. You now know what Neural Networks are, their applications and limitations.

The machine learning based model could provide the very efficient and fast proxy for complex and slow full physical based simulators which in application like optimization, could be very helpful. However, two areas need further attention:

* When coming from a complex model to the more simplification of the ‘truth model’, the part of accuracy will be sacrificed. The research in ML application must have analysis about the trade-off between speed and accuracy.
* In the ML model developed in this work, it was assumed the training data set here is “Historical Data” that was fed to the ML to make a prediction about the future field. In fact, that training data set is the ‘sample’ while we are considering to make a prediction about the population. Here, the challenging statistical question comes that how much training data set is representative of the population of the data set?

1. **Bibliography**

<https://trainings.internshala.com/machine-learning-training/>

<https://en.wikipedia.org/wiki/Machine_learning>

<https://www.geeksforgeeks.org/machine-learning/>

<https://www.youtube.com/watch?v=GwIo3gDZCVQ>