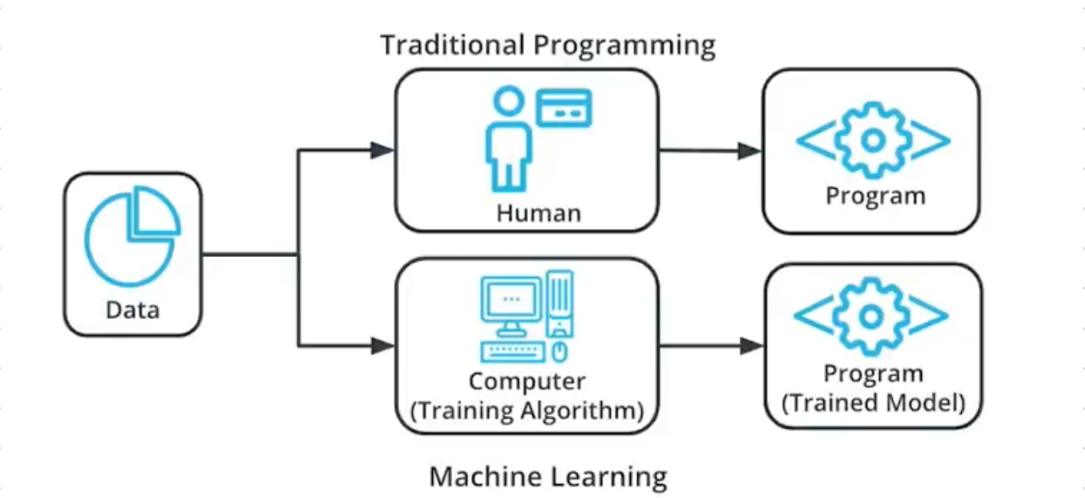
INTRODUCTION TO MACHINE LEARNING

In this lesson you will learn the fundamentals of the machine learning (ML) process and explore case studies using ML. When finished, you will understand when and how machine learning can be used to solve a problem.

Shared by The Ravit Show

WHAT IS MACHINE LEARNING ?

Machine learning is a modern software development technique, and a type of artificial intelligence (AI), that enables computers to solve problems by using examples of real-world data. It allows computers to automatically learn and improve from experience without being explicitly programmed to do so.



"Machine Learning is said to learn from experience with respect to some class of tasks, and a **performance measure P**, if learners performance at tasks in a class, as a measured by P, improves with experience "

-Tom Mitchell

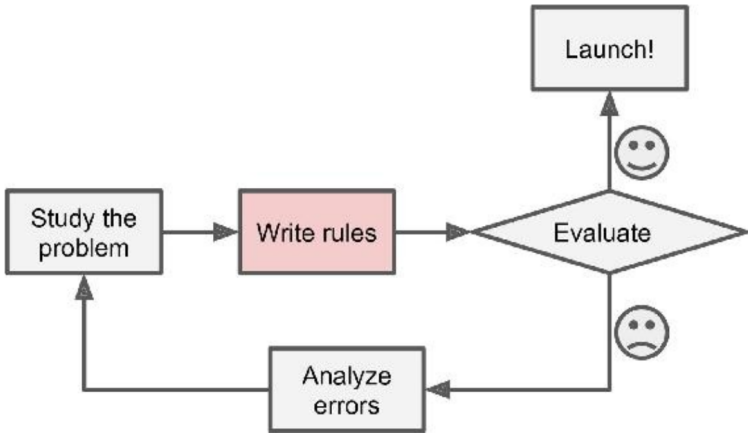
"Former Chair of the Machine Learning Department"

WHY USE MACHINE LEARNING ?

Consider how you would write a spam filter using traditional programming methods:

1. First you would look at what spam typically looks like. You might notice that some words or phrases (such as “4U,” “credit card,” “free,” and “amazing”) tend to come up a lot in the subject. Perhaps you would also notice a few other patterns in the sender’s name, the email’s body, and so on.
2. You would write a detection algorithm for each of the patterns that you noticed, and your program would flag emails as spam if a number of these patterns are detected.
3. You would test your program, and repeat steps 1 and 2 until it is good enough.

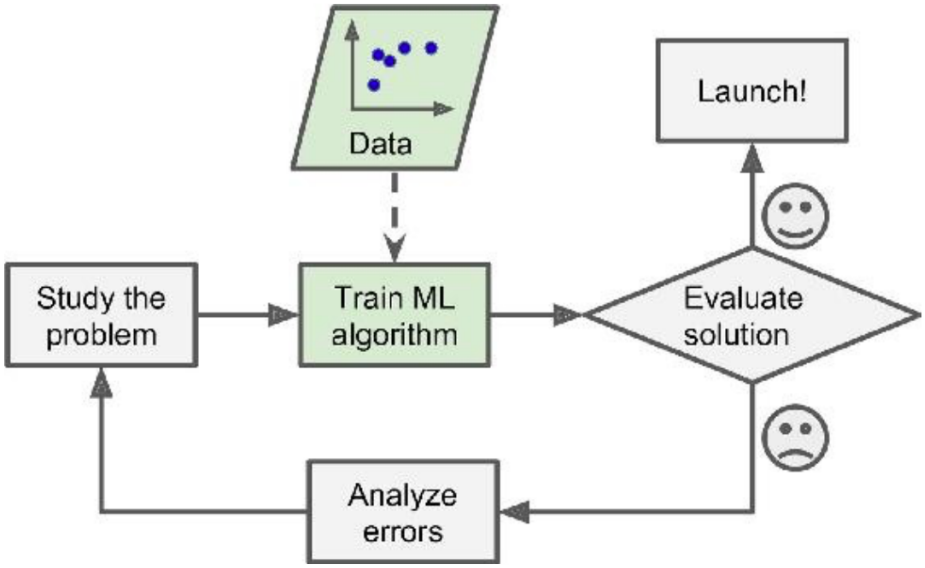
In contrast, a spam filter based on **Machine Learning techniques** automatically learns which words and phrases are good predictors of spam by detecting unusually frequent patterns of words in the spam examples compared to the ham examples . The program is much shorter, easier to maintain, and most likely more accurate.



Traditional Approach for writing a program on **SpamFilter**

WHY USE MACHINE LEARNING ?

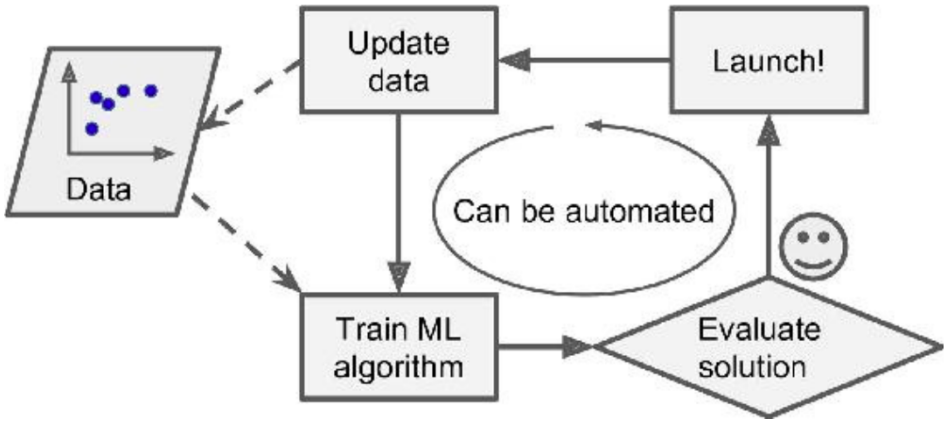
Moreover, if spammers notice that all their emails containing “4U” are blocked, they might start writing “For U” instead. A spam filter using traditional programming techniques would need to be updated to flag “For U” emails. If spammers keep working around your spam filter, you will need to keep writing new rules forever.



ML Approach for writing a program on **SpamFilter**

In contrast, a spam filter based on Machine Learning techniques automatically notices that “For U” has become unusually frequent in spam flagged by users, and it starts flagging them without your intervention.

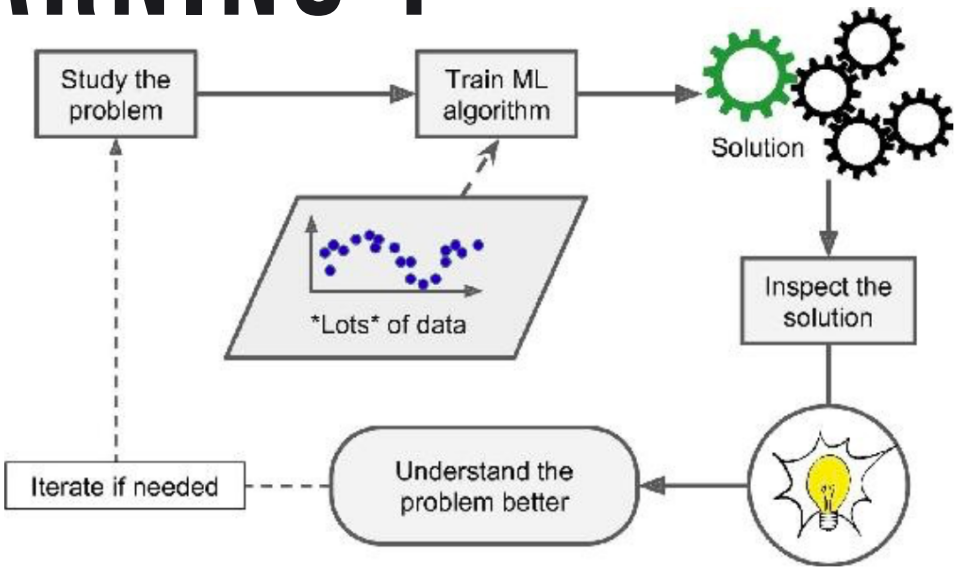
WHY USE MACHINE LEARNING ?



Automatically adapting to change

Another area where Machine Learning shines is for problems that either are too complex for traditional approaches or have no known algorithm. **For example**, consider speech recognition: say you want to start simple and write a program capable of distinguishing the words “one” and “two.” You might notice that the word “two” starts with a high-pitch sound (“T”), so you could hardcode an algorithm that measures high-pitch sound intensity and use that to distinguish ones and twos. Obviously this technique will not scale to thousands of words spoken by millions of very different people in noisy environments and in dozens of languages. The best solution (at least today) is to **write an algorithm that learns by itself, given many example recordings for each word.**

WHY USE MACHINE LEARNING ?



Machine Learning can help humans learn

Finally, Machine Learning can help humans learn: ML algorithms can be inspected to see what they have learned (although for some algorithms this can be tricky). For instance, once the spam filter has been trained on enough spam, it can easily be inspected to reveal the list of words and combinations of words that it believes are the best predictors of spam. Sometimes this will reveal unsuspected correlations or new trends, and thereby lead to a better understanding of the problem.

Applying ML techniques to dig into large amounts of data can help discover patterns that were not immediately apparent. **This is called data mining.**

BASIC ALGORITHMS

Naive Bayes

Naive Bayes Classifiers are based on applying Bayes Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other. Here we have put a naive assumption to the Bayes' theorem, which is, independence among the features. So now, we split evidence into the independent parts.This algorithm is mostly used in text classification and with problems having multiple classes.

Algorithm 1.1 Naive Bayes

Train($\mathbf{X}, \mathbf{Y}$) {reads documents $\mathbf{X}$ and labels $\mathbf{Y}$}
  Compute dictionary D of $\mathbf{X}$ with n words.
  Compute m, m_{ham} and m_{spam}.
  Initialize $b := \log c + \log m_{\text{ham}} - \log m_{\text{spam}}$ to offset the rejection threshold
  Initialize $p \in \mathbb{R}^{2\times n}$ with $p_{ij} = 1$, $w_{\text{spam}} = n$, $w_{\text{ham}} = n$.
  {Count occurrence of each word}
  {Here x_i^j denotes the number of times word j occurs in document x_i}
  for $i = 1$ to m **do**
    if $y_i =$ spam **then**
      for $j = 1$ to n **do**
        $p_{0,j} \leftarrow p_{0,j} + x_i^j$
        $w_{\text{spam}} \leftarrow w_{\text{spam}} + x_i^j$
      end for
    else
      for $j = 1$ to n **do**
        $p_{1,j} \leftarrow p_{1,j} + x_i^j$
        $w_{\text{ham}} \leftarrow w_{\text{ham}} + x_i^j$
      end for
    end if
  end for
  {Normalize counts to yield word probabilities}
  for $j = 1$ to n **do**
    $p_{0,j} \leftarrow p_{0,j}/w_{\text{spam}}$
    $p_{1,j} \leftarrow p_{1,j}/w_{\text{ham}}$
  end for
Classify(x) {classifies document x}
  Initialize score threshold $t = -b$
  for $j = 1$ to n **do**
    $t \leftarrow t + x^j(\log p_{0,j} - \log p_{1,j})$
  end for
  if $t > 0$ **return** spam **else return** ham

BASIC ALGORITHMS

Nearest Neighbor Estimators

The principle behind nearest neighbor methods is to find a predefined number of training samples closest in distance to the new point, and predict the label from these. The number of samples can be a user-defined constant (k-nearest neighbor learning), or vary based on the local density of points (radius-based neighbor learning). The distance can, in general, be any metric measure: standard Euclidean distance is the most common choice. However, the problem with nearest neighbour classification is that the estimates can be very noisy whenever the data itself is very noisy. For instance, if a spam email is erroneously labeled as nonspam then all emails which are similar to this email will share the same fate.

Algorithm 1.2 k-Nearest Neighbor Classification

Classify$(\mathbf{X}, \mathbf{Y}, x)$ {reads documents $\mathbf{X}$, labels $\mathbf{Y}$ and query x}
  for $i = 1$ **to** m **do**
    Compute distance $d(x_i, x)$
  end for
  Compute set I containing indices for the k smallest distances $d(x_i, x)$.
  return majority label of $\{y_i$ where $i \in I\}$.

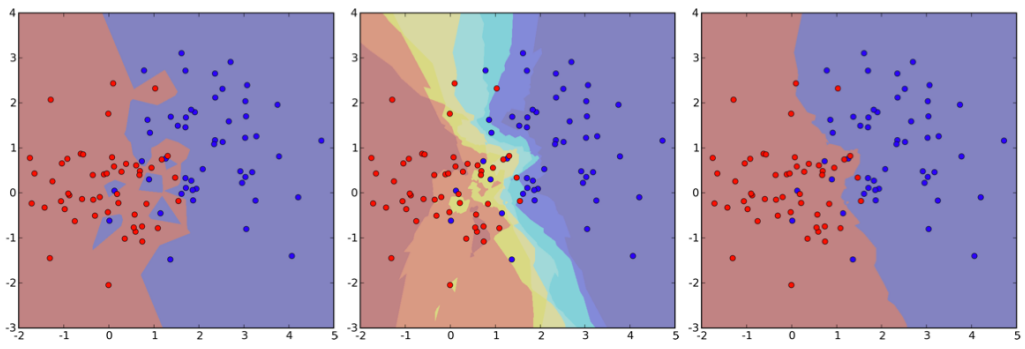


Fig. 1.18. k-Nearest neighbor classifiers using Euclidean distances. Left: decision boundaries obtained from a 1-nearest neighbor classifier. Middle: color-coded sets of where the number of red / blue points ranges between 7 and 0. Right: decision boundary determining where the blue or red dots are in the majority.

BASIC ALGORITHMS

A Simple Classifier : Extension of Nearest Neighbor Algorithm

It uses geometry to design another simple classification algorithm. We define the means μ+ and μ− to correspond to the classes y ∈ {±1} via

$$\mu_- := \frac{1}{m_-} \sum_{y_i=-1} x_i \text{ and } \mu_+ := \frac{1}{m_+} \sum_{y_i=1} x_i.$$

Here we used m− and m+ to denote the number of observations with label yi = −1 and yi = +1 respectively. An even simpler approach than using the nearest neighbor classifier would be to use the class label which corresponds to the mean closest to a new query x.

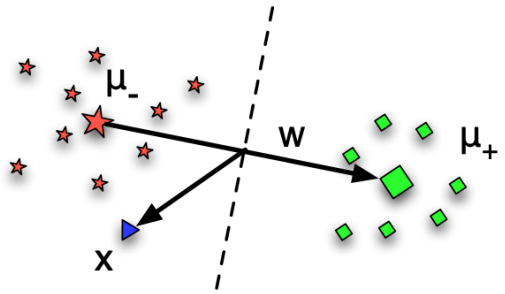


Fig. 1.20. A trivial classifier. Classification is carried out in accordance to which of the two means μ_- or μ_+ is closer to the test point x. Note that the sets of positive and negative labels respectively form a half space.

Classifier Rule:

$$f(x) = \sum_{i=1}^{m} \alpha_i \langle x_i, x \rangle + b$$

BASIC ALGORITHMS

Perceptron: Biological Inspiration of Neural Networks

Rosenblatt's perceptron is basically a binary classifier. The perceptron consists of 3 main parts:

- Input nodes or input layer: The input layer takes the initial data into the system for further processing. Each input node is associated with a numerical value. It can take any real value.
- Weights and bias: Weight parameters represent the strength of the connection between units. Higher is the weight, stronger is the influence of the associated input neuron to decide the output. Bias plays the same as the intercept in a linear equation.
- Activation function: The activation function determines whether the neuron will fire or not. At its simplest, the activation function is a step function, but based on the scenario, different activation functions can be used.

In the first step, all the input values are multiplied with their respective weights and added together. The result obtained is called weighted sum ∑wi*xi, or stated differently, x1*w1 + x2*w2 +… wn*xn. This sum gives an appropriate representation of the inputs based on their importance. Additionally, a bias term bis added to this sum ∑wi*xi + b. Bias serves as another model parameter (in addition to weights) that can be tuned to improve the model's performance.
In the second step, an activation function f is applied over the above sum ∑wi*xi + b to obtain output Y = f(∑wi*xi + b). Depending upon the scenario and the activation function used, the Output is either binary {1, 0} or a continuous value.

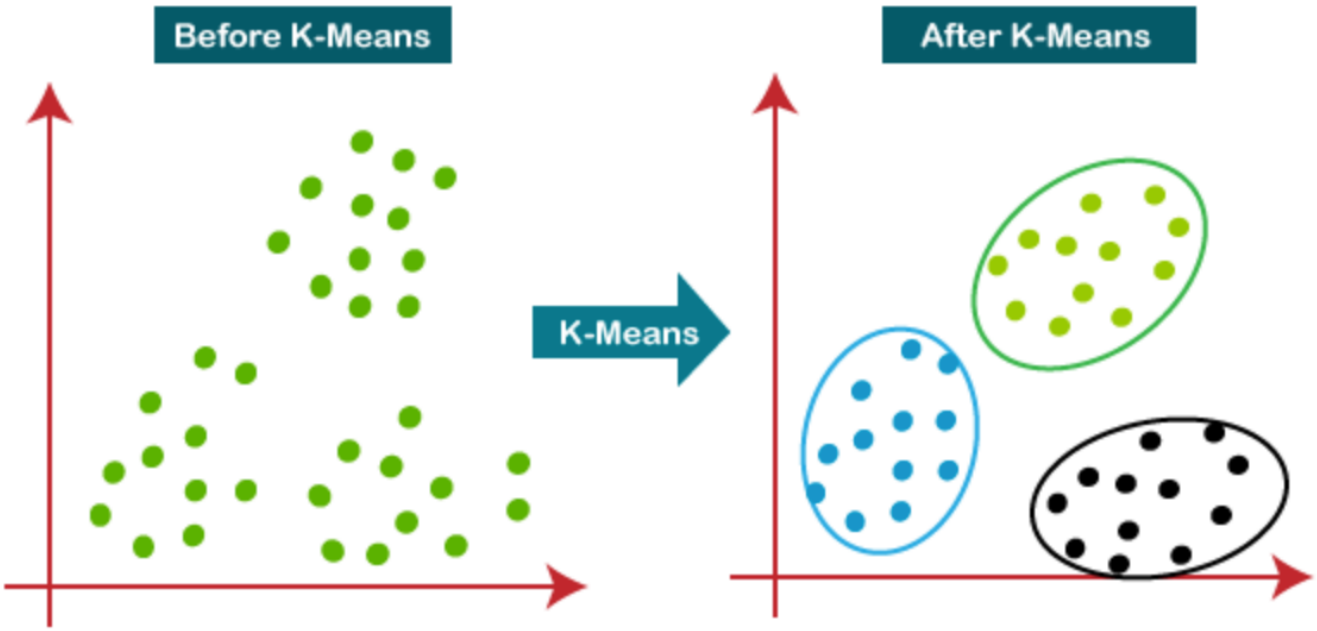
BASIC ALGORITHMS

k-means: A prototypical unsupervised learning algorithm

Given X = {x1, . . . , xm} the goal of K-means is to partition it into k clusters such that each point in a cluster is similar to points from its own cluster than with points from some other cluster.
Towards this end, define prototype vectors μ1, . . . , μk and an indicator vector rij which is 1 if, and only if, xi is assigned to cluster j.
To cluster our dataset we will minimize the following distortion measure, which minimizes the distance of each point from the prototype vector:

$$J(r,\mu) := \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{k} r_{ij}\|x_i - \mu_j\|^2, \qquad (1.29)$$

where $r = \{r_{ij}\}$, $\mu = \{\mu_j\}$, and $\|\cdot\|^2$ denotes the usual Euclidean square norm.

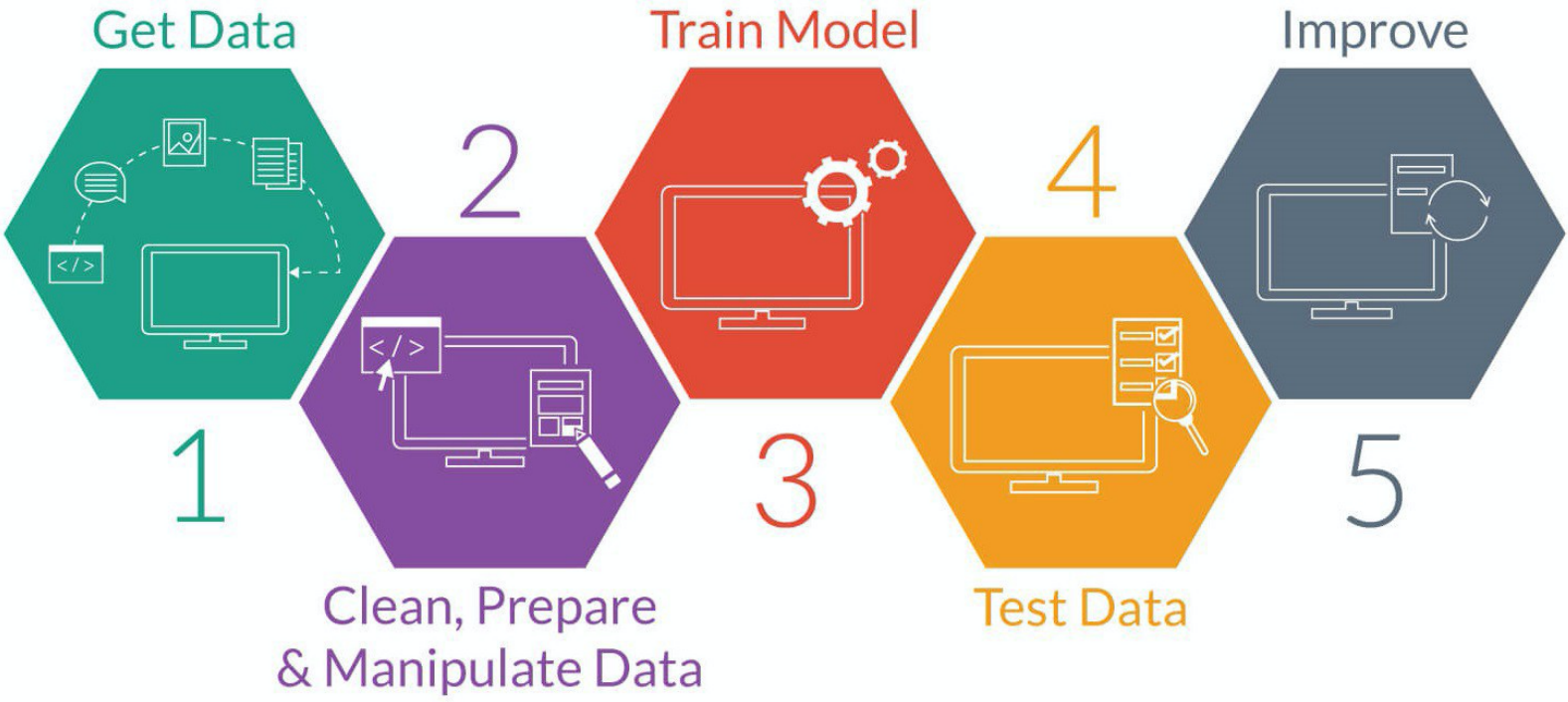


MODELING FLOW

In Machine Learning

Following protocol is used in ML:

1. **Business Problem:** Firstly define the problem statement and the goal to be achieved along with the assumptions we have in the data.
2. **Data Analysis:** Analyse data like whether it is regression or a classification problem.
3. **Data Preparation:** Check for null values and outliers and clear.
4. **Modeling:** Build a model using train data.
5. **Evaluation:** Tweak the model using the test data.
6. **Deployment:** Deploy the model based on accuracy on final version



TYPES OF ML

Supervised Learning:

1.**Regression:** Linear Regression and Decision Tree (RPART)

2.**Classification:** Logistic Regression, Support Vector Machines, Naive Bayes and Decision Tree (CART)

UnSupervised Learning:

1.**Clustering:** K-Means Clustering

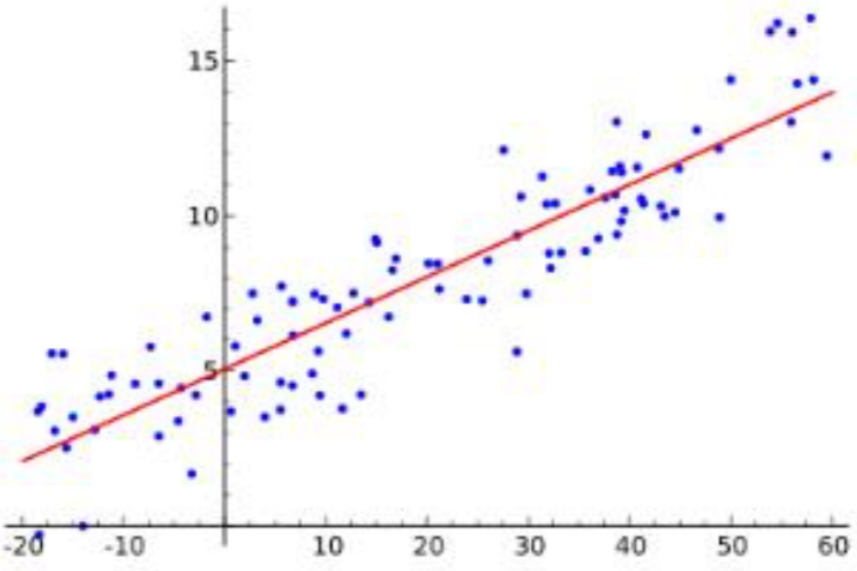
2.**Association:** Apriori Rules

SUPERVISED LEARNING

Here, all data is labelled and the algorithms learn to predict the output from the input data.

Regression Models

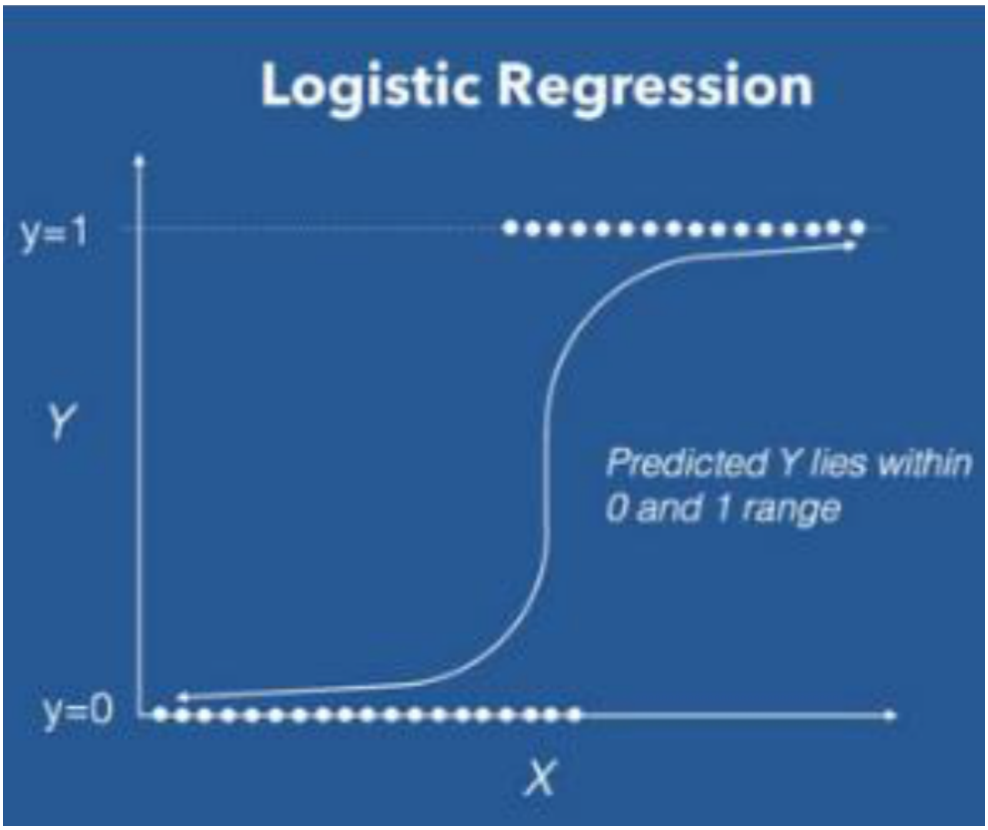
Linear Regression models are often fitted using the *least squares approach.* It will predict only the continuous variable. When there is a single input variable (x), the method is referred to as simple linear regression. When there are multiple input variables, then refers to the method as multiple linear regression.



SUPERVISED LEARNING

Classification Models

Logistic Regression is used to predict the probability of an instance belonging to the default class. Logistic Regression is a statistical method for analysing a dataset in which there are one or more independent variables that determine an outcome.



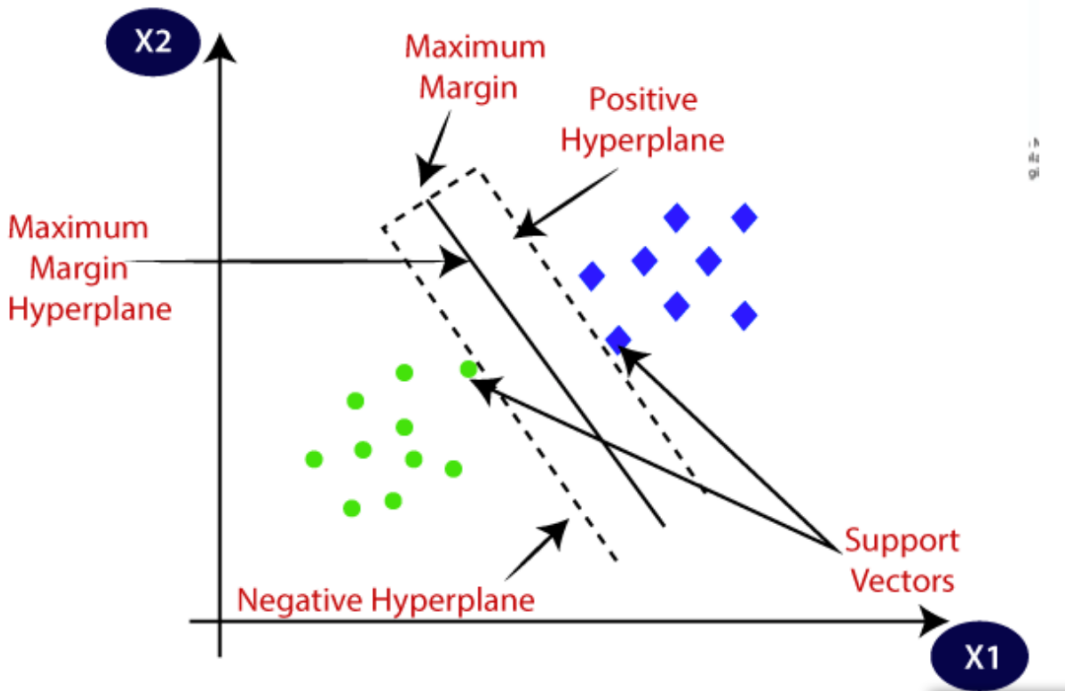
SUPERVISED LEARNING

Classification Models

Support Vector Machines

It can be used both for regression and classification problems. The goal of SVM is to find the hyperplane that separates these two classes. Multiple hyperplanes can be used to classify data.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane. SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.



SUPERVISED LEARNING

Classification Models

Artificial Neural Network

ANN are a class of pattern matching method. These methods are used for regression, classifications, image recognition, sequential data. **Two types of method** are involved in ANN: Learning Vector Quantisation and Self-Organising Maps.

UNSUPERVISED LEARNING

Clustering

K-Means Clustering

How does the K-Means Algorithm Work?

The working of the K-Means algorithm is explained in the below steps:

Step-1: Select the number K to decide the number of clusters.

Step-2: Select random K points or centroids. (It can be other from the input dataset).

Step-3: Assign each data point to their closest centroid, which will form the predefined K clusters.

Step-4: Calculate the variance and place a new centroid of each cluster.

Step-5: Repeat the third steps, which means reassign each datapoint to the new closest centroid of each cluster.

Step-6: If any reassignment occurs, then go to step-4 else go to FINISH.

Step-7: The model is ready.

UNSUPERVISED LEARNING

Association Rule Analysis

An association rule problem is where you want to discover rules that describe large portions of your data, such as people that buy X also tend to buy Y. Eg.- Market based analysis.
Association rule learning is a type of unsupervised learning technique that checks for the dependency of one data item on another data item and maps accordingly so that it can be more profitable. It tries to find some interesting relations or associations among the variables of dataset. It is based on different rules to discover the interesting relations between variables in the database.

Association rule learning can be divided into three types of algorithms:

1. Apriori
2. Eclat
3. F-P Growth Algorithm

Apriori uses a "bottom up" approach, where frequent subsets are extended one item at a time. It is a classical algorithm used in data mining for mining frequent item sets and relevant association rules.