INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

Dundigal, Hyderabad - 500 043

Course Code	ACAC03
Course Name	Foundations of Machine Learning
Class / Semester	3rd Year/ VI Semester
Section	A
Name of the Department	CSE(Data Science)
Employee ID	IARE10805
Employee Name	Dr.G.Sucharitha
Topic Covered	Introduction to Machine Learning
Course Outcome/s	Able to understand the fundamentals of machine learning
Handout Number	1
Date	

1.Introduction to Machine Learning

1.1 What Is Machine Learning?

Machine learning is programming computers to optimize a performance criterion using example data or past experience. We have a model defined up to some parameters, and learning is the execution of a computer program to optimize the parameters of the model using the training data or past experience. The model may be predictive to make predictions in the future, or descriptive to gain knowledge from data, or both.

Arthur Samuel, an early American leader in the field of computer gaming and artificial intelligence, coined the term “Machine Learning” in 1959 while at IBM. He defined machine learning as “the field of study that gives computers the ability to learn without being explicitly programmed.” However, there is no universally accepted definition for machine learning. Different authors define the term differently.

Definition of learning

Definition

A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks T, as measured by P, improves with experience E.

Examples

i) Handwriting recognition learning problem

- Task T: Recognizing and classifying handwritten words within images
- Performance P: Percent of words correctly classified

- Training experience E: A dataset of handwritten words with given classifications

ii) A robot driving learning problem

- Task T: Driving on highways using vision sensors
- Performance measure P: Average distance traveled before an error
- training experience: A sequence of images and steering commands recorded while observing a human driver

iii) A chess learning problem

- Task T: Playing chess
- Performance measure P: Percent of games won against opponents
- Training experience E: Playing practice games against itself.

1.2 What kind of problems can be tackled using machine learning?

Predicting the label of a document, also known as document classification, is by no means the only learning task. Machine learning admits a very broad set of practical applications, which include the following:

- Text or document classification. This includes problems such as assigning a topic to a text or a document, or determining automatically if the content of a web page is inappropriate or too explicit; it also includes spam detection.

- Natural language processing (NLP). Most tasks in this field, including part-of-speech tagging, named-entity recognition, context-free parsing, or dependency parsing, are cast as learning problems. In these problems, predictions admit some structure. For example, in part-of-speech tagging, the prediction for a sentence is a sequence of part-of-speech tags labeling each word. In context-free parsing the prediction is a tree. These are instances of richer learning problems known as structured prediction problems.

- Speech processing applications. This includes speech recognition, speech synthesis, speaker verification, speaker identification, as well as sub-problems such as language modeling and acoustic modeling.

- Computer vision applications. This includes object recognition, object identification, face detection, Optical character recognition (OCR), content-based image retrieval, or pose estimation.

- Computational biology applications. This includes protein function prediction, identification of key sites, or the analysis of gene and protein networks.

- Many other problems such as fraud detection for credit card, telephone or insurance companies, network intrusion, learning to play games such as chess, backgammon, or Go, unassisted control of vehicles such as robots or cars, medical diagnosis, the design of recommendation systems, search engines, or information extraction systems, are tackled using machine learning techniques.

This list is by no means comprehensive. Most prediction problems found in practice can be cast as learning problems and the practical application area of machine learning keeps expanding.

The algorithms and techniques discussed in this book can be used to derive solutions for all of these problems, though we will not discuss in detail these applications.

1.3 Components of Learning

Basic components of learning process

The learning process, whether by a human or a machine, can be divided into four components, namely, data storage, abstraction, generalization and evaluation. Figure 1.1 illustrates the various components and the steps involved in the learning process.

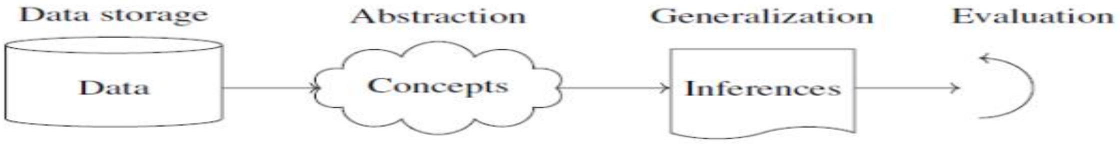


Figure 1.1: Components of learning process

1. Data storage

Facilities for storing and retrieving huge amounts of data are an important component of the learning process. Humans and computers alike utilize data storage as a foundation for advanced reasoning.

- In a human being, the data is stored in the brain and data is retrieved using electrochemical signals.
- Computers use hard disk drives, flash memory, random access memory and similar devices to store data and use cables and other technology to retrieve data.

2. Abstraction

The second component of the learning process is known as abstraction.

Abstraction is the process of extracting knowledge about stored data. This involves creating general concepts about the data as a whole. The creation of knowledge involves application of known models and creation of new models. The process of fitting a model to a dataset is known as training. When the model has been trained, the data is transformed into an abstract form that summarizes the original information.

3. Generalization

The third component of the learning process is known as generalization.

The term generalization describes the process of turning the knowledge about stored data into a form that can be utilized for future action. These actions are to be carried out on tasks that are similar, but not identical, to those what have been seen before. In generalization, the goal is to discover those properties of the data that will be most relevant to future tasks.

4. Evaluation

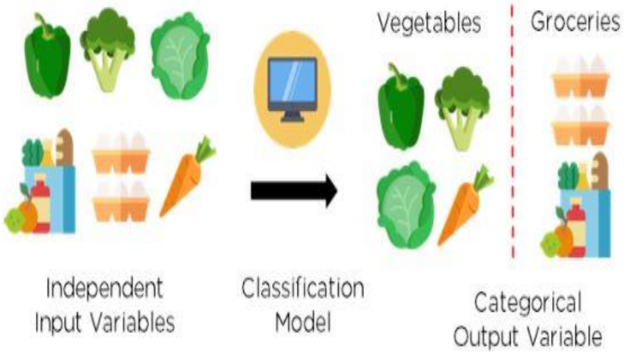
Evaluation is the last component of the learning process.

It is the process of giving feedback to the user to measure the utility of the learned knowledge. This feedback is then utilized to effect improvements in the whole learning process.

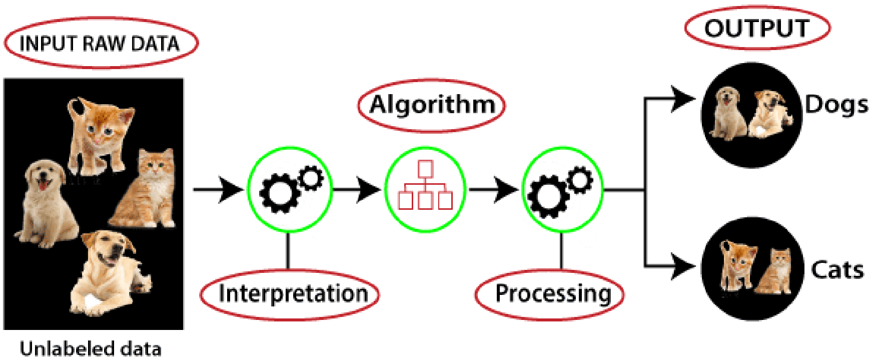
Course Code	ACAC03
Course Name	Foundations of Machine Learning
Class / Semester	3rd Year/ VI Semester
Section	A
Name of the Department	CSE(Data Science)
Employee ID	IARE10805
Employee Name	Dr.G.Sucharitha
Topic Covered	Learning Problems and Scenarios
Course Outcome/s	Able to understand the learning problems involved in machine learning.
Handout Number	2
Date	

Some common machine learning scenarios are given below. These scenarios differ in the types of training data available to the learner, the order and method by which training data is received and the test data used to evaluate the learning algorithm.

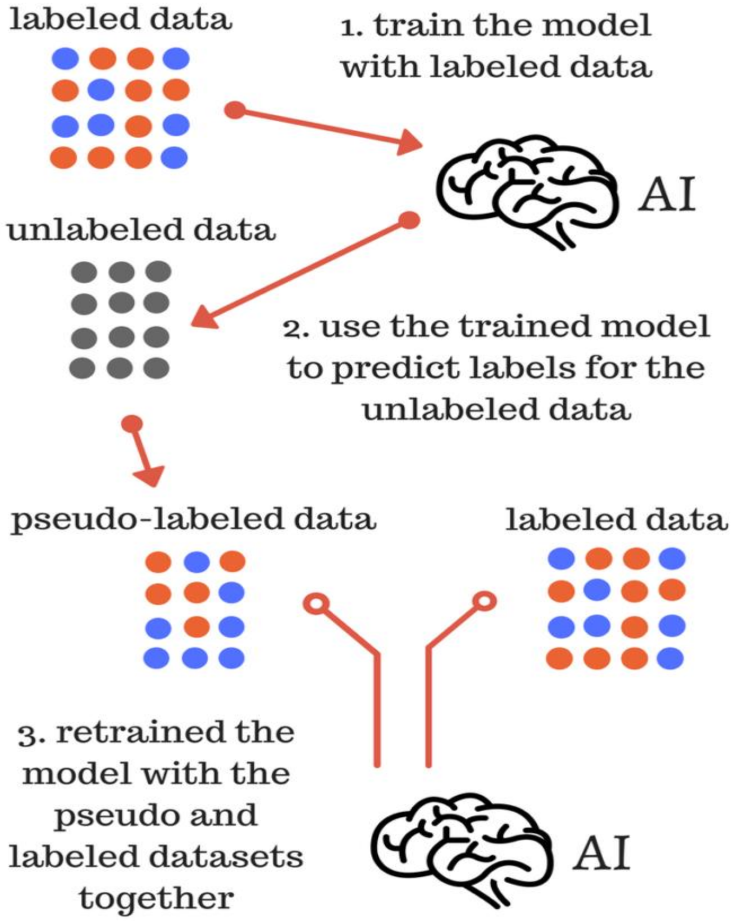
•Supervised learning: The learner receives a set of labeled examples as training data and makes predictions for all unseen points. This is the most common scenario associated with classification, regression, and ranking problems. The spam detection problem discussed in the previous section is an instance of supervised learning.



• Unsupervised learning: The learner exclusively receives unlabeled training data, and makes predictions for all unseen points. Since in general no labeled example is available in that setting, it can be difficult to quantitatively evaluate the performance of a learner. Clustering and dimensionality reduction are example of unsupervised learning problems.

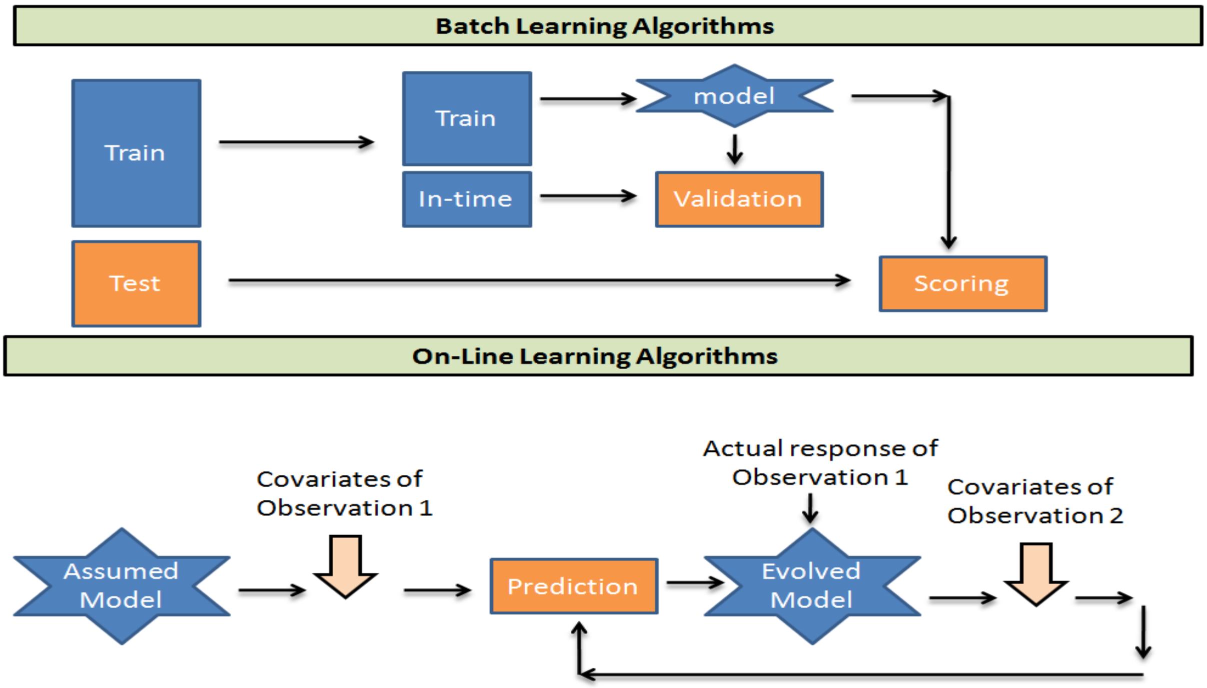


• Semi-supervised learning: The learner receives a training sample consisting of both labeled and unlabeled data, and makes predictions for all unseen points. Semi-supervised learning is common in settings where unlabeled data is easily accessible but labels are expensive to obtain. Various types of problems arising in applications, including classification, regression, or ranking tasks, can be framed as instances of semi-supervised learning. The hope is that the distribution of unlabeled data accessible to the learner can help him achieve a better performance than in the supervised setting. The analysis of the conditions under which this can indeed be realized is the topic of much modern theoretical and applied machine learning research.

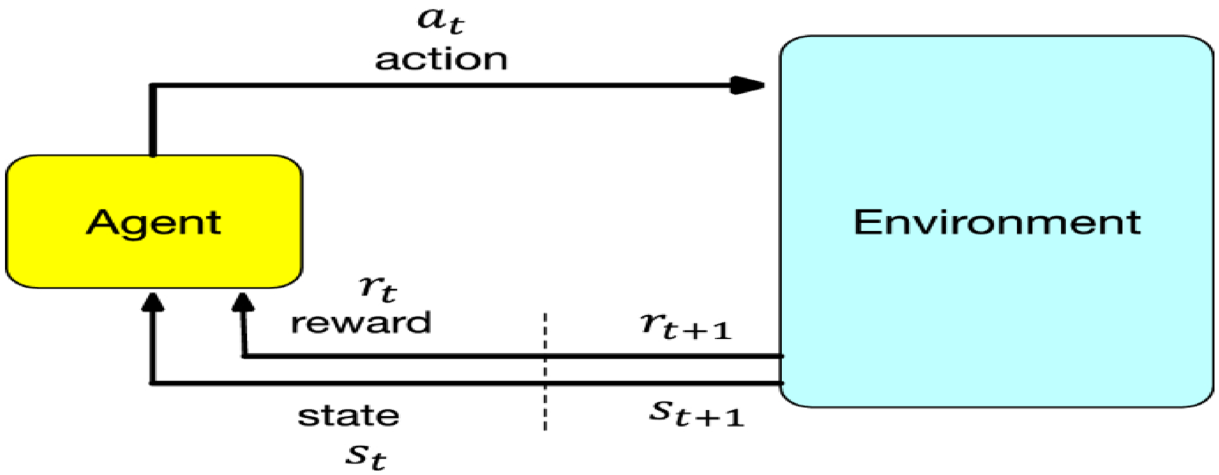


•Transductive inference: As in the semi-supervised scenario, the learner receives a labeled training sample along with a set of unlabeled test points. However, the objective of transductive inference is to predict labels only for these particular test points. Transductive inference appears to be an easier task and matches the scenario encountered in a variety of modern applications. However, as in the semi-supervised setting, the assumptions under which a better performance can be achieved in this setting are research questions that have not been fully resolved.

• On-line learning: In contrast with the previous scenarios, the online scenario involves multiple rounds where training and testing phases are intermixed. At each round, the learner receives an unlabeled training point, makes a prediction, receives the true label, and incurs a loss. The objective in the on-line setting is to minimize the cumulative loss over all rounds or to minimize the regret, that is the difference of the cumulative loss incurred and that of the best expert in hindsight. Unlike the previous settings just discussed, no distributional assumption is made in on-line learning. In fact, instances and their labels may be chosen adversarial within this scenario.


Batch Learning Algorithms
Train
Train
model
In-time
Validation
Test
Scoring
On-Line Learning Algorithms
Covariates of Observation 1
Actual response of Observation 1
Covariates of Observation 2
Assumed Model
Prediction
Evolved Model

•Reinforcement learning: The training and testing phases are also intermixed in reinforcement learning. To collect information, the learner actively interacts with the environment and in some cases affects the environment, and receives an immediate reward for each action. The object of the learner is to maximize his reward over a course of actions and iterations with the environment. However, no long-term reward feedback is provided by the environment, and the learner is faced with the exploration versus exploitation dilemma, since he must choose between exploring unknown actions to gain more information versus exploiting the information already collected.



•Active learning: The learner adaptively or interactively collects training examples, typically by querying an oracle to request labels for new points. The goal in active learning is to achieve a performance comparable to the standard supervised learning scenario (or passive learning scenario), but with fewer labeled examples. Active learning is often used in applications where labels are expensive to obtain, for example computational biology applications.

Course Code	**ACAC03**
Course Name	**Foundations of Machine Learning**
Class / Semester	**3rd Year/ VI Semester**
Section	**A**
Name of the Department	**CSE(Data Science)**
Employee ID	**IARE10805**
Employee Name	**Dr.G.Sucharitha**
Topic Covered	Need For Machine Learning
Course Outcome/s	Able to understand the necessity of machine learning.
Handout Number	**3**
Date	

Need For Machine Learning ?

Machine Learning allows the user to feed a computer algorithm an immense amount of data and have the computer analyze and make data-driven recommendations and decisions based on only the input data.

What is most important for Machine Learning?

Training is the most important part of Machine Learning. Choose your features and hyper parameters carefully. Machines don't take decisions, people do. Data cleaning is the most important part of Machine Learning.

Why is Machine Learning important for the future?

It could use to automate tasks or improve processes, predict outcomes and make decisions based on past experiences. Machine learning can also be used to create powerful algorithms that help you make sense of large amounts of data.

Course Code	ACAC03
Course Name	Foundations of Machine Learning
Class / Semester	3rd Year/ VI Semester
Section	A
Name of the Department	CSE(Data Science)
Employee ID	IARE10805
Employee Name	Dr.G.Sucharitha
Topic Covered	Types of learning
Course Outcome/s	Able to understand different types of machine learning models
Handout Number	4
Date	

Machine Learning is a core form of Artificial Intelligence that enable machine to learn from past data and make predictions It involves data exploration and pattern matching with minimal human intervention. There are mainly four technologies that machine learning used to work.

1. Supervised Learning:

Supervised Learning is a machine learning method that needs supervision similar to the student-teacher relationship. In supervised Learning, a machine is trained with well-labelled data, which means some data is already tagged with correct outputs. So, whenever new data is introduced into the system, supervised learning algorithms analyse this sample data and predict correct outputs with the help of that labelled data.

It is classified into two different categories of algorithms. These are as follows:

- **Classification:** It deals when output is in the form of a category such as Yellow, blue, right or wrong, etc.
- **Regression:** It deals when output variables are real values like age, height, etc.

This technology allows us to collect or produce data output from experience. It works the same way as humans learn using some labelled data points of the training set. It helps in optimizing the performance of models using experience and solving various complex computation problems.

2. Unsupervised Learning:

Unlike supervised learning, unsupervised Learning does not require classified or well-labelled data to train a machine. It aims to make groups of unsorted information based on some patterns and differences even without any labelled training data. In unsupervised Learning, no supervision is provided, so no sample data is given to the machines. Hence, machines are restricted to finding hidden structures in unlabelled data by their own.

It is classified into two different categories of algorithms. These are as follows:

- **Clustering:** It deals when there is a requirement of inherent grouping in training data, e.g., grouping students by their area of interest.
- **Association:** It deals with the rules that help to identify a large portion of data, such as students who are interested in ML and also interested in AI.

3.Semi-supervised learning:

Semi-supervised Learning is defined as the combination of both supervised and unsupervised learning methods. It is used to overcome the drawbacks of both supervised and unsupervised learning methods.

In the semi-supervised learning method, a machine is trained with labelled as well as unlabelled data. Although, it involves a few labelled examples and a large number of unlabelled examples.

Speech analysis, web content classification, protein sequence classification, and text documents classifiers are some most popular real-world applications of semi-supervised Learning.

4. Reinforcement learning:

Reinforcement learning is defined as a feedback-based machine learning method that does not require labeled data. In this learning method, an agent learns to behave in an environment by performing the actions and seeing the results of actions. Agents can provide positive feedback for each good action and negative feedback for bad actions. Since, in reinforcement learning, there is no training data, hence agents are restricted to learn with their experience only.

Course Code	ACAC03
Course Name	Foundations of Machine Learning
Class / Semester	3rd Year/ VI Semester
Section	A
Name of the Department	CSE(Data Science)
Employee ID	IARE10805
Employee Name	Dr.G.Sucharitha
Topic Covered	Standard Learning Tasks
Course Outcome/s	Able to understand the learning process involved in machine learning
Handout Number	5
Date	

Standard Learning Tasks

Here, we will use the canonical problem of spam detection as a running example to illustrate some basic definitions and describe the use and evaluation of machine learning algorithms in practice, including their different stages. Spam detection is the problem of learning to automatically classify email messages as either spam or non-spam. The following is a list of definitions and terminology commonly used in machine learning:

• Examples: Items or instances of data used for learning or evaluation. In our spam problem, these examples correspond to the collection of email messages we will use for learning and testing.

• Features: The set of attributes, often represented as a vector, associated to an example. In the case of email messages, some relevant features may include the length of the message, the name of the sender, various characteristics of the header, the presence of certain keywords in the body of the message, and so on.

• Labels: Values or categories assigned to examples. In classification problems, examples are assigned specific categories, for instance, the spam and non-spam categories in our binary classification problem. In regression, items are assigned real-valued labels.

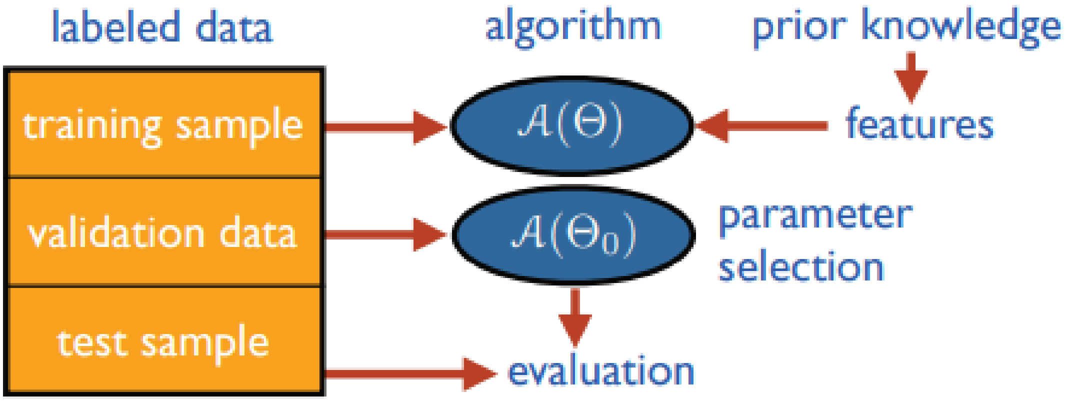
• Hyperparameters: Free parameters that are not determined by the learning algorithm, but rather specified as inputs to the learning algorithm.

• Training sample: Examples used to train a learning algorithm. In our spam problem, the training sample consists of a set of email examples along with their associated labels. The training sample varies for different learning scenarios.

• Validation sample: Examples used to tune the parameters of a learning algorithm when working with labeled data. The validation sample is used to select appropriate values for the learning algorithm's free parameters (hyperparameters).

• Test sample: Examples used to evaluate the performance of a learning algorithm. The test sample is separate from the training and validation data and is not made available in the learning stage. In the spam problem, the test sample consists of a collection of email examples for which the learning algorithm must predict labels based on features. These predictions are then compared with the labels of the test sample to measure the performance of the algorithm.

• Loss function: A function that measures the difference, or loss, between a predicted label and a true label. Denoting the set of all labels as y and the set of possible predictions as y', a loss function L is a mapping L: y × y' → R+. In most cases, y'=y and the loss function are bounded, but these conditions do not always hold.



Hypothesis set: A set of functions mapping features (feature vectors) to the set of labels Y. In our example, these may be a set of functions mapping email features to Y = {spam, non-spam}. More generally, hypotheses may be functions mapping features to a different set y'. They could be linear functions mapping email feature vectors to real numbers interpreted as scores (y' = R), with higher score values more indicative of spam than lower ones.

Course Code	ACAC03
Course Name	Foundations of Machine Learning
Class / Semester	3rd Year/ VI Semester
Section	A
Name of the Department	CSE(Data Science)
Employee ID	IARE10805
Employee Name	Dr.G.Sucharitha
Topic Covered	Statistical Learning Framework
Course Outcome/s	Able to understand the learning process involved in machine learning
Handout Number	6
Date	

Statistical learning theory is a framework for machine learning that draws from statistics and functional analysis. It deals with finding a predictive function based on the data presented. The main idea in statistical learning theory is to build a model that can draw conclusions from data and make predictions.

Types of Data in Statistical Learning:

With statistical learning theory, there are two main types of data:

➢ Dependent Variable — a variable (y) whose values depend on the values of other variables (a dependent variable is sometimes also referred to as a target variable).

➢ Independent Variables — a variable (x) whose value does not depend on the values of other variables (independent variables are sometimes also referred to as predictor variables, input variables, explanatory variables, or features).

➢ In statistical learning, the independent variable(s) are the variable that will affect the dependent variable.

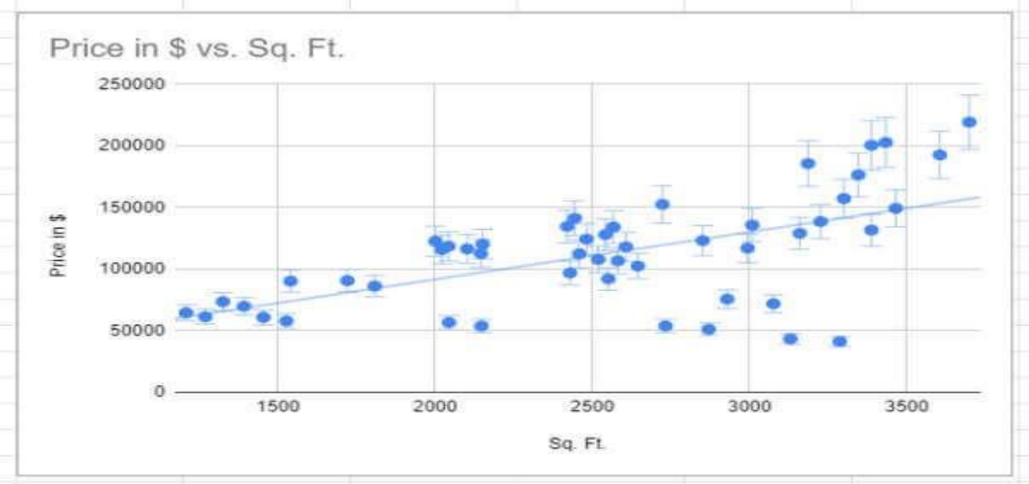
A common example of an Independent Variable is Age. There is nothing that one can do to increase or decrease age. This variable is independent.

Some common examples of Dependent Variables are:

➢ Weight — a person's weight is dependent on his or her age, diet, and activity levels (as well as other factors).

➢ Temperature — temperature is impacted by altitude, distance from equator (latitude) and distance from the sea.

The price of a home is affected by the size of the home, sq. ft is the independent variable while price of the home is the dependent variable.



Statistical Model

A statistical model defines the relationships between a dependent and independent variable. In the above graph, the relationships between the size of the home and the price of the home is illustrated by the straight line. We can define this relationship by using y = mx + c where m represents the gradient and c is the intercept. Another way that this equation can be expressed is with roman numerals which would look something like.

$$Price\ of\ Home = \beta_0 + \beta_1 * Sq.Ft$$

If we suppose that the size of the home is not the only independent variable when determining the price and that the number of bedrooms is also an independent variable, the equation would look like

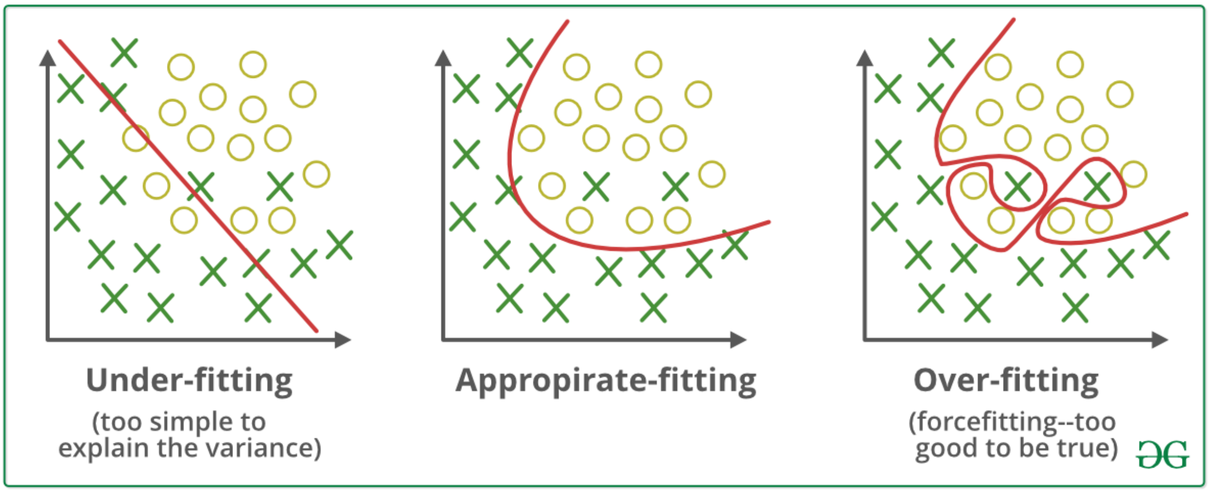
$$Price\ of\ Home = \beta_0 + \beta_1 * Sq.Ft + \beta_2 * No.of\ Bedrooms$$

Model Generalization

In order to build an effective model, the available data needs to be used in a way that would make the model generalizable for unseen situations. Some problems that occur when building models is that the model under-fits or over-fits to the data.

- Under-fitting — when a statistical model does not adequately capture the underlying structure of the data and, therefore, does not include some parameters that would appear in a correctly specified model.

- Over-fitting — when a statistical model contains more parameters that can be justified by the data and includes the residual variation ("noise") as if the variation represents underlying model structure.



Under-fitting
(too simple to explain the variance)

Appropirate-fitting

Over-fitting
(forcefitting--too good to be true)

Course Code	**ACAC03**
Course Name	**Foundations of Machine Learning**
Class / Semester	**3rd Year/ VI Semester**
Section	**A**
Name of the Department	**CSE(Data Science)**
Employee ID	**IARE10805**
Employee Name	**Dr.G.Sucharitha**
Topic Covered	Probably Approximately Correct (PAC) learning
Course Outcome/s	Able to understand the learning process involved in machine learning
Handout Number	**7&8**
Date	

PAC (Probably Approximately Correct) learning is **a framework used for mathematical analysis**. A PAC Learner tries to learn a concept (approximately correct) by selecting a hypothesis from a set of hypotheses that has a low generalization error.

The PAC Learning Model

We denote by X the set of all possible examples or instances. X is also sometimes referred to as the input space. The set of all possible labels or target values is denoted by Y. For the purpose of this introductory chapter, we will limit ourselves to the case where Y is reduced to two labels, Y = {0, 1}, which corresponds to the so-called binary classification. Later chapters will extend these results to more general settings.

A concept c : X → Y is a mapping from X to Y. Since Y = {0, 1}, we can identify c with the subset of X over which it takes the value 1. Thus, in the following, we equivalently refer to a concept to learn as a mapping from X to {0, 1}, or as a subset of X. As an example, a concept may be the set of points inside a triangle or the indicator function of these points. In such cases, we will say in short that the concept to learn is a triangle. A concept class is a set of concepts we may wish to learn and is denoted by C. This could, for example, be the set of all triangles in the plane.

We assume that examples are independently and identically distributed (i.i.d.) according to some fixed but unknown distribution D. The learning problem is then formulated as follows. The learner considers a fixed set of possible concepts H, called a hypothesis set, which might not necessarily coincide with C. It receives a sample $S = (x_1, x_2 \ldots . x_m)$ drawn i.i.d. according to D as well as the labels $c = c(x_1), c(x_2), \ldots c(x_m))$, which are based on a specific target concept c ∈ C to learn. The task is then to use the labelled sample S to select a hypothesis h_S ∈ H that has a small generalization error with respect to the concept c. The generalization error of a hypothesis h ∈ H, also referred to as the risk or true error (or simply error) of h is denoted by R(h) and defined as follows.

A concept class C is said to be PAC-learnable if there exists an algorithm A and a polynomial function poly(·, ·, ·, ·) such that for any $\epsilon > 0$ and $\delta > 0$, for all distributions D on X and for any target concept c ∈ C, the following holds for any sample size $m \geq \text{poly}(1/\epsilon, 1/\delta, n, \text{size}(c))$:

$$\underset{S\sim\mathcal{D}^m}{\mathbb{P}}[R(h_S) \leq \epsilon] \geq 1 - \delta.$$

If A further runs in *poly*(1/ϵ, 1/δ, n, size(c)), then C is said to be efficiently PAC-learnable. When such an algorithm A exists, it is called a PAC-learning algorithm for C.

A concept class C is thus PAC-learnable if the hypothesis returned by the algorithm after observing a number of points polynomial in 1/ ϵ and 1/δ is approximately correct (error at most ϵ) with high probability (at least 1 − δ), which justifies the PAC terminology. The parameter δ > 0 is used to define the confidence 1 – δ and ϵ > 0 the accuracy 1 − ϵ. Note that if the running time of the algorithm is polynomial in 1/ ϵ and 1/δ, then the sample size m must also be polynomial if the full sample is received by the algorithm.

Several key points of the PAC definition are worth emphasizing. First, the PAC framework is a distribution-free model: no particular assumption is made about the distribution D from which examples are drawn. Second, the training sample and the test examples used to define the error are drawn according to the same distribution D. This is a natural and necessary assumption for generalization to be possible in general. It can be relaxed to include favourable domain adaptation problems. Finally, the PAC framework deals with the question of learnability for a concept class C and not a particular concept. Note that the concept class C is known to the algorithm, but of course the target concept c ∈ C is unknown.

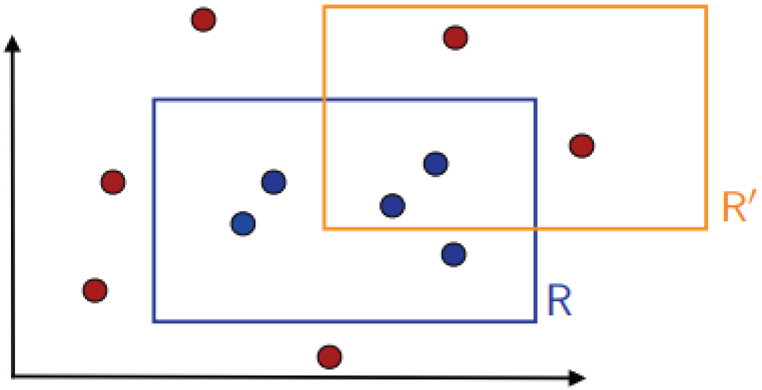


Figure1: Target concept R and possible hypothesis R 0 . Circles represent training instances. A blue circle is a point labelled with 1, since it falls within the rectangle R. Others are red and labelled with 0.

In many cases, in particular when the computational representation of the concepts is not explicitly discussed or is straightforward, we may omit the polynomial dependency on n and size(c) in the PAC definition and focus only on the sample complexity.

Module-II

SUPERVISED LEARNING ALGORITHMS

Course Code	ACAC03
Course Name	Foundations of Machine Learning
Class / Semester	3rd Year/ VI Semester
Section	A
Name of the Department	CSE(Data Science)
Employee ID	IARE10805
Employee Name	Dr.G.Sucharitha
Topic Covered	Learning a Class from Examples
Course Outcome/s	Able to understand the learning process involved in machine learning
Handout Number	9
Date	

Let us assume that, we want to learn the class, C, of a "family car." We have a set of examples of cars, and we have a group of people that we survey to whom we show these cars. The people look at the cars and label them; the cars that they believe are family cars are positive examples, and the other cars are negative examples. Class learning is finding a description that is shared by all the positive examples and none of the negative examples. Doing this, we can make a prediction: Given a car that we have not seen before, by checking with the description learned, we will be able to say whether it is a family car or not. Or we can do knowledge extraction: This study may be sponsored by a car company, and the aim may be to understand what people expect from a family car.

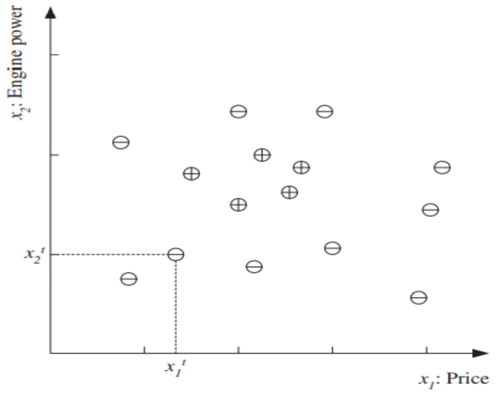


Figure1: Training set for the class of a "family car." Each data point corresponds to one example car, and the coordinates of the point indicate the price and engine power of that car. '+' denotes a positive example of the class (a family car), and '−' denotes a negative example (not a family car); it is another type of car.

Let us denote price as the first input attribute x1 (e.g., in U.S. dollars) and engine power as the second attribute x2 (e.g., engine volume in cubic centimetres). Thus, we represent each car using two numeric values.

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{1}$$

and its label denotes its type

$$r = \begin{cases} 1 & if\ x\ is\ a\ positive\ Example \\ 0 & if\ x\ is\ a\ negative\ Example \end{cases} \tag{2}$$

Each car is represented by such an ordered pair (x,r) and the training set contains N such examples

$$X = \{x^t, r^t\}_{t=1}^{N} \tag{3}$$

where t indexes different examples in the set; it does not represent time or any such order.

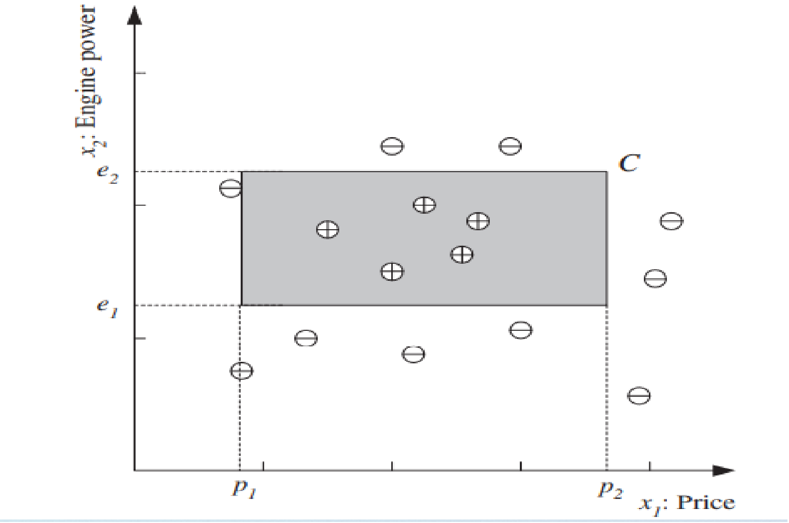


Figure 2: Example of a hypothesis class. The class of family car is a rectangle in the price-engine power space.

Course Code	ACAC03
Course Name	Foundations of Machine Learning
Class / Semester	3rd Year/ VI Semester
Section	A
Name of the Department	CSE(Data Science)
Employee ID	IARE10805
Employee Name	Dr.G.Sucharitha
Topic Covered	Linear and Non-linear Classification
Course Outcome/s	Able to apply the linear and non-linear classifications on the supervised data.
Handout Number	10
Date	

Definition: Linear & Non-Linear Classification:

Linear Classification refers to categorizing a set of data points to a discrete class based on a linear combination of its explanatory variables. On the other hand, Non-Linear Classification refers to separating those instances that are not linearly separable.

Linear Classification

Linear Classification refers to categorizing a set of data points into a discrete class based on a linear combination of its explanatory variables. Some of the classifiers that use linear functions to separate classes are Linear Discriminant Classifier, Naive Bayes, Logistic Regression, Perceptron, SVM (linear kernel).

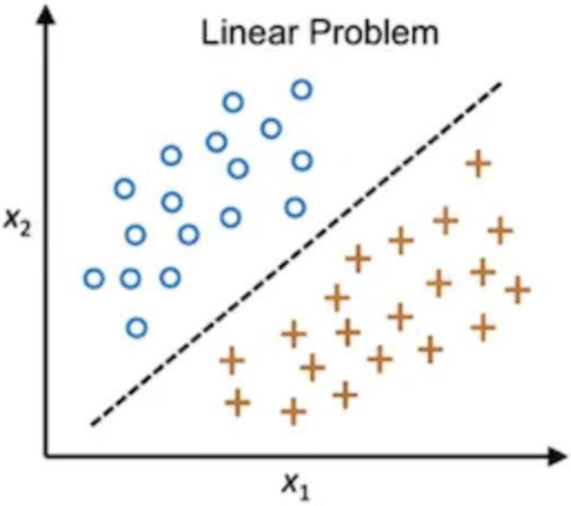


Figure 1: Linear Classification

In the figure above, the two classes, namely 'O' and '+.' To differentiate between the two classes, an arbitrary line is drawn, ensuring that both the classes are on distinct sides.

- Since we can tell one class apart from the other, these classes are called ‘linearly-separable.’

- However, an infinite number of lines can be drawn to distinguish the two classes.

- The exact location of this plane/hyperplane depends on the type of the linear classifier.

Non-Linear Classification

Non-Linear Classification refers to categorizing those instances that are not linearly separable. It is possible to classify data with a straight line. It is not easy to classify data with a straight line. Data is classified with the help of a hyperplane.

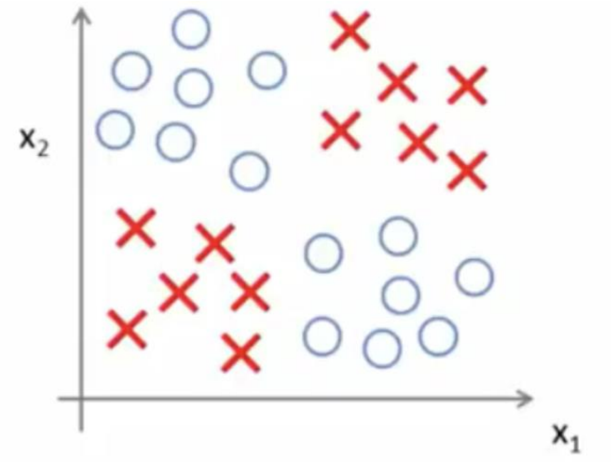


Figure1: Non-Linear Classification

In the figure above, we have two classes, namely 'O' and 'X.' To differentiate between the two classes, it is impossible to draw an arbitrary straight line to ensure that both the classes are on distinct sides.

- We notice that even if we draw a straight line, there would be points of the first-class present between the data points of the second class.

- In such cases, piece-wise linear or non-linear classification boundaries are required to distinguish the two classes.

Course Code	ACAC03
Course Name	Foundations of Machine Learning
Class / Semester	3rd Year/ VI Semester
Section	A
Name of the Department	CSE(Data Science)
Employee ID	IARE10805
Employee Name	Dr.G.Sucharitha
Topic Covered	Multi-Class and Multi-Label Classification
Course Outcome/s	Able to understand the Multi-class and Multi-Label classification on the supervised data.
Handout Number	11
Date	

Multi-Class and Multi-Label Classification

Multiclass classification means a classification problem where the task is to classify between more than two classes. Multilabel classification means a classification problem where we get multiple labels as output.

The differences between the types of classifications

- **Binary classification:**
  It is used when there are only two distinct classes and the data, we want to classify belongs exclusively to one of those classes, e.g. to classify if a post about a given product as positive or negative.
- **Multiclass classification:** It is used when there are three or more classes and the data, we want to classify belongs exclusively to one of those classes, e.g., to classify if a semaphore on an image is red, yellow or green.
- **Multilabel classification:**
  It is used when there are two or more classes and the data, we want to classify may belong to none of the classes or all of them at the same time, e.g., to classify which traffic signs are contained on an image.

Multi-label classification: It allows us to classify data sets with more than one target variable. In multi-label classification, we have several labels that are the outputs for a given prediction. When making predictions, a given input may belong to more than one label.

For example, when predicting a given movie category, it may belong to horror, romance, adventure, action, or all simultaneously. In this example, we have multi-labels that can be assigned to a given movie.

Multi-class classification: In multi-class classification, an input belongs to only a single label. For example, when predicting if a given image belongs to a cat or a dog, the output can be either a cat or dog but not both at the same time.

In this tutorial, we will be dealing with multi-label text classification, and we will build a model that classifies a given text input into different categories. Our text input can belong to multiple categories or labels at the same time.

Course Code	ACAC03
Course Name	Foundations of Machine Learning
Class / Semester	3rd Year/ VI Semester
Section	A
Name of the Department	CSE(Data Science)
Employee ID	IARE10805
Employee Name	Dr.G.Sucharitha
Topic Covered	Decision Trees: ID3
Course Outcome/s	Able to understand the ID3 Decision Tree algorithm to classifying the supervised data.
Handout Number	12
Date	

What is Decision Tree?

A decision tree is a structure that contains nodes and edges and is built from a dataset. Each node is either used to make a decision (known as decision node) or represent an outcome known as leaf node.

Decision Tree Example:

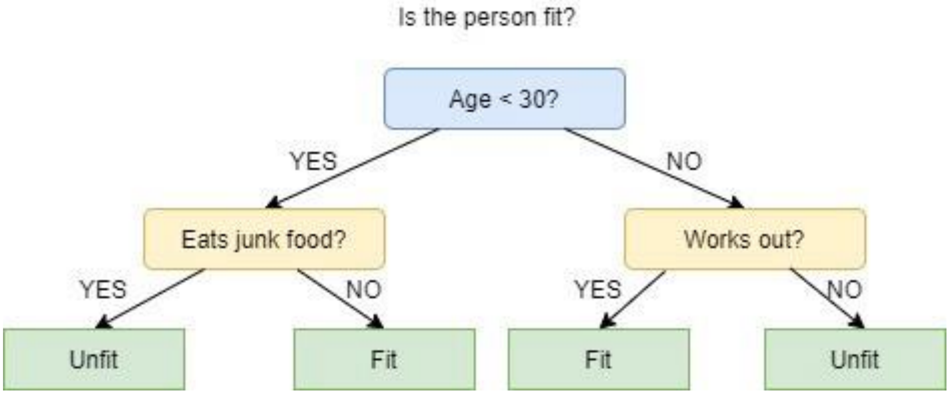


Figure 1: A Decision tree used to classify whether a person is Fit or Unfit.

The decision nodes here are questions like *'Is the person less than 30 years of age?'*, *'Does the person eat junk?'*, etc. and the leaves are one of the two possible outcomes viz. **Fit** and **Unfit**. Looking at the Decision Tree we can say make the following decisions: if a person is less than 30 years of age and doesn't eat junk food then he is Fit, if a person is less than 30 years of age and eats junk food then he is Unfit and so on.

The initial node is called the root node (colored in blue), the final nodes are called the leaf nodes (colored in green) and the rest of the nodes are called intermediate **or** internal nodes. The root and intermediate nodes represent the decisions while the leaf nodes represent the outcomes.

ID3 Algorithm

ID3 stands for Iterative Dichotomiser 3 and is named such because the algorithm iteratively (repeatedly) dichotomizes(divides) features into two or more groups at each step.
Invented by Ross Quinlan, ID3 uses a top-down greedy approach to build a decision tree. In simple words, the top-down approach means that we start building the tree from the top and the **greedy** approach means that at each iteration we select the best feature at the present moment to create a node.
Most generally ID3 is only used for classification problems with nominal features only.

Metrics in ID3

As mentioned previously, the ID3 algorithm selects the best feature at each step while building a Decision tree. Before you ask, the answer to the question: 'How does ID3 select the best feature?' is that ID3 uses Information Gain or just Gain to find the best feature.

Information Gain calculates the reduction in the entropy and measures how well a given feature separates or classifies the target classes. The feature with the highest Information Gain is selected as the **best** one. In simple words, Entropy is the measure of disorder and the Entropy of a dataset is the measure of disorder in the target feature of the dataset.
In the case of binary classification (where the target column has only two types of classes) entropy is **0** if all values in the target column are homogenous(similar) and will be **1** if the target column has equal number values for both the classes.
We denote our dataset as **S,** entropy is calculated as:

$$Entropy(S) = -\sum P_i * log_2(P_i); i = 1\ to\ n$$

where, **n** is the total number of classes in the target column (in our case n = 2 i.e YES and NO), **p$_i$** is the probability of class **'i'** or the ratio of "*number of rows with class i in the target column"* to the "*total number of rows"* in the dataset.

Information Gain for a feature column **A** is calculated as:

$$I_G(S, A) = Entropy(S) - \sum\left(\left(\frac{|S_V|}{|S|}\right) * Entropy(S_V)\right)$$

Where **S$_v$** is the set of rows in **S** for which the feature column **A** has value **v**, **|S$_v$|** is the number of rows in **S$_v$** and likewise **|S|** is the number of rows in **S.**

Course Code	ACAC03
Course Name	Foundations of Machine Learning
Class / Semester	3rd Year/ VI Semester
Section	A
Name of the Department	CSE(Data Science)
Employee ID	IARE10805
Employee Name	Dr.G.Sucharitha
Topic Covered	Classification and Regression Trees (CART)
Course Outcome/s	Able to understand the regression models to predict output from labelled data.
Handout Number	13
Date	

CART (Classification and Regression Tree) is a variation of the decision tree algorithm. It can handle both classification and regression tasks.

CART Algorithm

CART is a predictive algorithm used in Machine learning and it explains how the target variable's values can be predicted based on other matters. It is a decision tree where each fork is split into a predictor variable and each node has a prediction for the target variable at the end.

In the decision tree, nodes are split into sub-nodes on the basis of a threshold value of an attribute. The root node is taken as the training set and is split into two by considering the best attribute and threshold value. Further, the subsets are also split using the same logic. This continues till the last pure sub-set is found in the tree or the maximum number of leaves possible in that growing tree.

The CART algorithm works via the following process:

- The best split point of each input is obtained.
- Based on the best split points of each input in Step 1, the new "best" split point is identified.
- Split the chosen input according to the "best" split point.
- Continue splitting until a stopping rule is satisfied or no further desirable splitting is available.

CART algorithm uses Gini Impurity to split the dataset into a decision tree. It does that by searching for the best homogeneity for the sub nodes, with the help of the Gini index criterion.

Gini index/Gini impurity

The Gini index is a metric for the classification tasks in CART. It stores the sum of squared probabilities of each class. It computes the degree of probability of a specific variable that is wrongly being classified when chosen randomly and a variation of the Gini coefficient. It works on categorical variables, provides outcomes either “successful” or “failure” and hence conducts binary splitting only.

The degree of the Gini index varies from 0 to 1,

- Where 0 depicts that all the elements are allied to a certain class, or only one class exists there.
- The Gini index of value 1 signifies that all the elements are randomly distributed across various classes.
- A value of 0.5 denotes the elements are uniformly distributed into some classes.

Mathematically, we can write Gini Impurity as follows:

$$Gini = 1 - \sum_{i=1}^{n} (P_i)^2$$

where P_i is the probability of an object being classified to a particular class.

CART Model Representation

CART models are formed by picking input variables and evaluating split points on those variables until an appropriate tree is produced.

Steps to create a Decision Tree using the CART algorithm:

- **Greedy algorithm**: In this the input space is divided using the Greedy method which is known as a recursive binary spitting. This is a numerical method within which all of the values are aligned and several other split points are tried and assessed using a cost function.

- **Stopping Criterion:** As it works its way down the tree with the training data, the recursive binary splitting method described above must know when to stop splitting. The most frequent halting method is to utilize a minimum amount of training data allocated to every leaf node. If the count is smaller than the specified threshold, the split is rejected and also the node is considered the last leaf node.

- **Tree pruning:** Decision tree’s complexity is defined as the number of splits in the tree. Trees with fewer branches are recommended as they are simple to grasp and less prone to cluster the data. Working through each leaf node in the tree and evaluating the effect of deleting it using a hold-out test set is the quickest and simplest pruning approach.

- **Data preparation for the CART:** No special data preparation is required for the CART algorithm.

Advantages of CART

- Results are simplistic.
- Classification and regression trees are Nonparametric and Nonlinear.
- Classification and regression trees implicitly perform feature selection.
- Outliers have no meaningful effect on CART.
- It requires minimal supervision and produces easy-to-understand models.

Limitations of CART

- Overfitting.
- High Variance.
- low bias.
- the tree structure may be unstable.

Applications of the CART algorithm

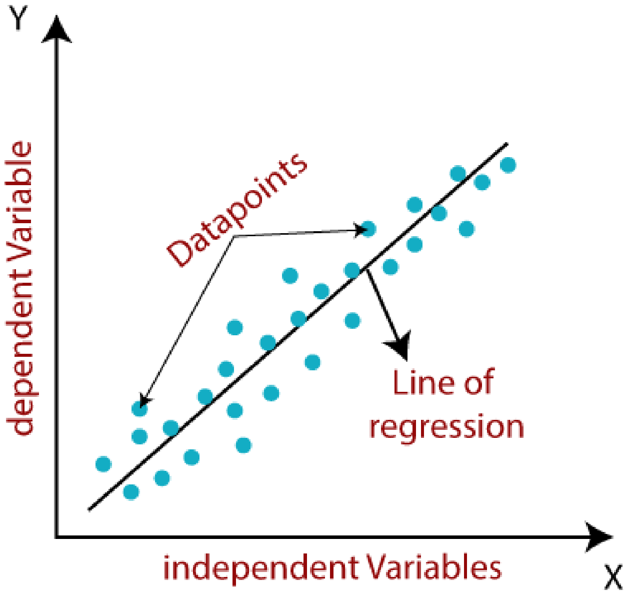
- For quick Data insights.
- In Blood Donors Classification.
- For environmental and ecological data.
- In the financial sectors.

Course Code	ACAC03
Course Name	Foundations of Machine Learning
Class / Semester	3rd Year/ VI Semester
Section	A
Name of the Department	CSE(Data Science)
Employee ID	IARE10805
Employee Name	Dr.G.Sucharitha
Topic Covered	Regression: Linear Regression
Course Outcome/s	Able to understand the linear regression models to predict output from labelled data.
Handout Number	14
Date	

Linear regression is one of the easiest and most popular Machine Learning algorithms. It is a statistical method that is used for predictive analysis. Linear regression makes predictions for continuous/real or numeric variables such as sales, salary, age, product price, etc.

Linear regression algorithm shows a linear relationship between a dependent (y) and one or more independent (y) variables, hence called as linear regression. Since linear regression shows the linear relationship, which means it finds how the value of the dependent variable is changing according to the value of the independent variable.

The linear regression model provides a sloped straight line representing the relationship between the variables. Consider the below image:



Mathematically, we can represent a linear regression as:

$$Y= a_0+a_1X+ \varepsilon$$

Here,

Y= Dependent Variable (Target Variable)
X= Independent Variable (predictor Variable)
a0= intercept of the line (Gives an additional degree of freedom)
a1 = Linear regression coefficient (scale factor to each input value).
ε = random error

The values for x and y variables are training datasets for Linear Regression model representation.

Types of Linear Regression

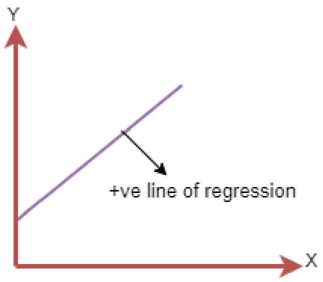
Linear regression can be further divided into two types of the algorithm:

- **Simple Linear Regression:**
  If a single independent variable is used to predict the value of a numerical dependent variable, then such a Linear Regression algorithm is called Simple Linear Regression.
- **Multiple Linear regression:**
  If more than one independent variable is used to predict the value of a numerical dependent variable, then such a Linear Regression algorithm is called Multiple Linear Regression.

Linear Regression Line

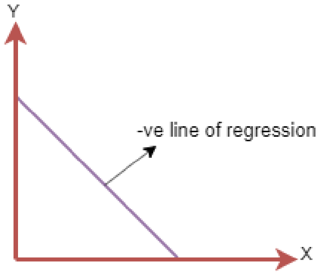
A linear line showing the relationship between the dependent and independent variables is called a **regression line**. A regression line can show two types of relationship:

- **Positive Linear Relationship:**
  If the dependent variable increases on the Y-axis and independent variable increases on X-axis, then such a relationship is termed as a Positive linear relationship.



The line equation will be: $Y= a_0+a_1x$

Negative Linear Relationship:
If the dependent variable decreases on the Y-axis and independent variable increases on the X-axis, then such a relationship is called a negative linear relationship.



The line of equation will be: **$Y= -a_0+a_1x$**

Finding the Best Fit Line:

When working with linear regression, our main goal is to find the best fit line that means the error between predicted values and actual values should be minimized. The best fit line will have the least error.

The different values for weights or the coefficient of lines (a_0, a_1) gives a different line of regression, so we need to calculate the best values for a_0 and a_1 to find the best fit line, so to calculate this we use cost function.

Cost Function

- The different values for weights or coefficient of lines (a_0, a_1) gives the different line of regression, and the cost function is used to estimate the values of the coefficient for the best fit line.
- Cost function optimizes the regression coefficients or weights. It measures how a linear regression model is performing.

- We can use the cost function to find the accuracy of the mapping function, which maps the input variable to the output variable. This mapping function is also known as Hypothesis function**.**

For Linear Regression, we use the **Mean Squared Error (MSE)** cost function, which is the average of squared error occurred between the predicted values and actual values. It can be written as,

For the above linear equation, MSE can be calculated as:

$$MSE = \frac{1}{N}\sum_{i=1}^{n}(y_i - (a_i x_i + a_0)^2$$

N=Total number of observations
Yi = Actual value
($a1x_i+a_0$) = Predicted value.

Course Code	ACAC03
Course Name	Foundations of Machine Learning
Class / Semester	3rd Year/ VI Semester
Section	A
Name of the Department	CSE(Data Science)
Employee ID	IARE10805
Employee Name	Dr.G.Sucharitha
Topic Covered	Multiple Linear Regression
Course Outcome/s	Able to understand the concepts of multiple linear regression models to predict output from labelled data.
Handout Number	15
Date	

In order to model the link between two or more characteristics and a response, multiple linear regression fits a linear equation to the observed data. The procedures for performing multiple linear regression are almost identical to those for performing simple linear regression. The assessment is where the difference lies. It can help us determine which variable has the most influence on the expected result and how several factors interact.

Regression model presumption:

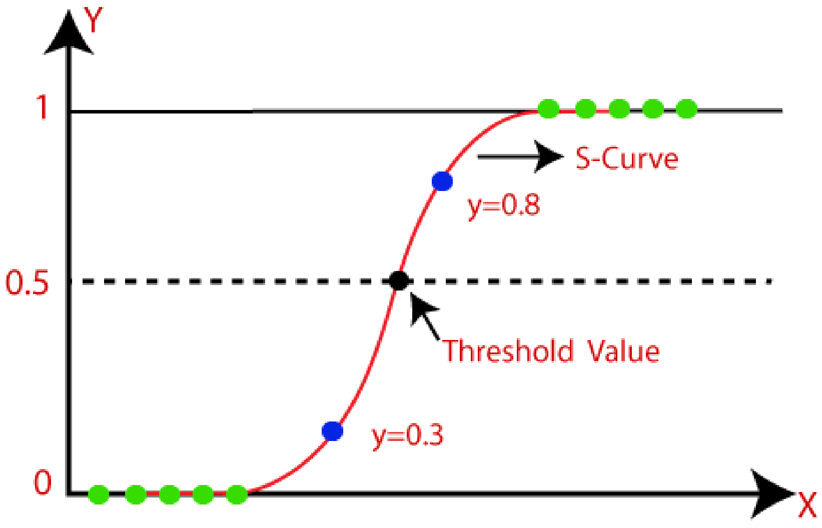
- Linearity: Dependent and independent variables should have a straight-line connection.
- Homoscedasticity: The errors' variance should be constant.
- Multivariate normality: The residuals are thought to be normally distributed when using multiple regression.
- Lack of Multicollinearity: It is believed that there is little or no multicollinearity in the data.

Course Code	ACAC03
Course Name	Foundations of Machine Learning
Class / Semester	3rd Year/ VI Semester
Section	A
Name of the Department	CSE(Data Science)
Employee ID	IARE10805
Employee Name	Dr.G.Sucharitha
Topic Covered	Logistic Regression.
Course Outcome/s	Able to understand the concepts of logistic regression models to predict output from labelled data.
Handout Number	16
Date	

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables. Logistic regression predicts the output of a categorical dependent variable. Therefore, the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.

Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas Logistic regression is used for solving the classification problems. In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1). The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc. Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.

Logistic Regression can be used to classify the observations using different types of data and can easily determine the most effective variables used for the classification. The below image is showing the logistic function:



Logistic Function (Sigmoid Function):

- The sigmoid function is a mathematical function used to map the predicted values to probabilities.
- It maps any real value into another value within a range of 0 and 1.
- The value of the logistic regression must be between 0 and 1, which cannot go beyond this limit, so it forms a curve like the "S" form. The S-form curve is called the Sigmoid function or the logistic function.
- In logistic regression, we use the concept of the threshold value, which defines the probability of either 0 or 1. Such as values above the threshold value tends to 1, and a value below the threshold values tends to 0.

Assumptions for Logistic Regression:

- The dependent variable must be categorical in nature.
- The independent variable should not have multi-collinearity.