
Introduction and Basics of Machine Learning(ML)

HIGHLIGHTS

What is machine learning?

Why Machine Learning?

Machine Learning Problem Solving Approach

Supervised Learning

Unsupervised Learning

Difference between Supervised and Unsupervised Learning

Overview of Machine Learning Models

- Regression

 - Simple and Multiple Regression

 - Other types of Regression Models

- Classification

Difference between Classification and Regression

Different types of Classification tasks in Machine Learning

- Binary Classification

- Multi-class classification Commonly used Classification Algorithms

 - Logistic Regression

 - Naïve Bayes'

 - Support Vector Machines (SVM)

 - Types of SVMs

 - K-nearest Neighbor

 - Random Forest

 - Neural Networks and Deep Learning

Unsupervised Learning

Common Unsupervised Learning Approaches

Clustering

Types of Clustering Methods

K-means Clustering

Divisive Clustering

Mean-shift Clustering Algorithm

Gaussian Mixture Model

Association Rules

Dimensionality Reduction

Semi-supervised Machine Learning

Reinforcement Learning

Evaluating the Accuracy of Classification Methods

Evaluation Metrics for Regression Models

Some Fundamental Concepts Related to Machine Learning

Summary

What is Machine Learning (ML)?

Machine learning is a method of designing systems that can learn, adjust, and improve based on the data fed to them without being explicitly programmed. Machine learning applications analyze and create models from very large samples or huge amounts of data, commonly referred to as **big data**. Machine learning is closely related to artificial intelligence (AI). In fact, it is an application of artificial intelligence (AI).

data. The abundance of high-quality data now available can be used efficiently in ML models.

Why Machine Learning?

The machine learning approach is used because conventional programming, in many cases, cannot handle the huge quantity of data and problems at hand. For simple tasks, it is not difficult to

develop a program and instruct the computer to execute the steps and tasks that can solve the problem. In these cases, no learning is needed. For more advanced and complex tasks, it may be challenging to write programs to execute the required tasks because of huge quantities of data and the complexity of the task. In such cases, it seems more logical and effective to program the machine to develop and learn from its own algorithms. Examples include facial recognition and stock market predictions, where machines can develop their own algorithms rather than relying on programmers to execute every step. [17] -. Ethem Alpaydin (2020). *Introduction to Machine Learning (Fourth ed.)*. [MIT](#). pp. xix, 1–3, 13–18. [ISBN 978-0262043793](#).]

In machine learning applications, huge amounts of data are needed, and it is challenging to develop programs to accomplish the task due to many potential answers. One approach used in such cases is to label some correct answers as valid answers and use this as **training data**. Using the training data, the computer improves the algorithms to find the correct answer.

Machine Learning Problem Solving Approach

At the fundamental level, Machine learning tasks are typically classified into following broad categories depending on the nature of the problem and learning "signal" or "feedback" available to a learning system. These are [\[20\]](#)

(1) Supervised learning

(2) Unsupervised learning

(3) Semi-supervised learning, and

(4) Reinforced Learning.

The broad categories of machine learning are explained in Figure 2.1

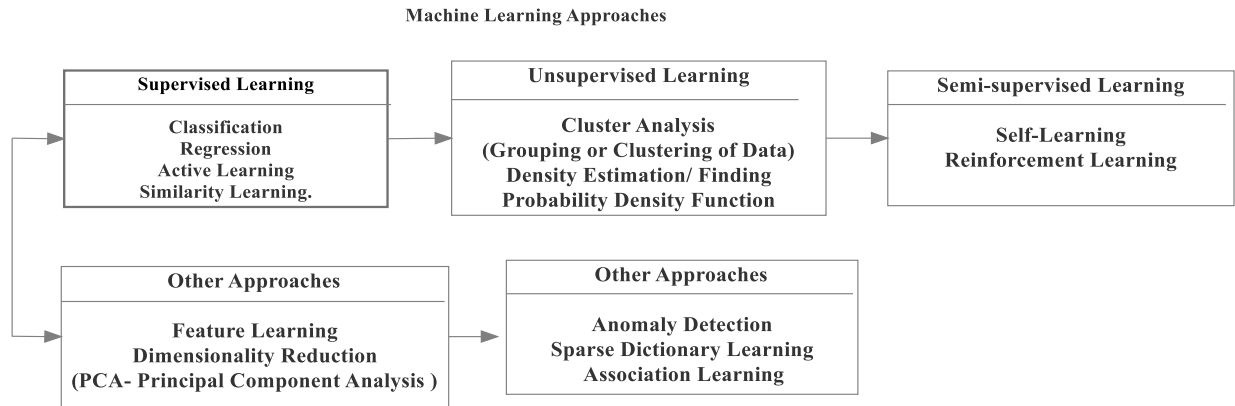


Figure 2.1: Types of Machine Learning Problems and Approaches

Supervised learning

In supervised learning, the computer is presented with example inputs and their desired outputs by the analyst, and the goal is to create a model to learn a general rule that maps inputs to outputs.

The learning involves creating a model by establishing the relationship between inputs or the *features* and the desired output or some *label*. Once this model is established, it can be used to apply labels to new, unknown data to predict future events. The idea is to learn from the past data and apply it to new data to make predictions for future events using the created learning algorithm.

In supervised learning, there is a set of measurements (x_i, y_i) where x_i are the input or predictor measurements and y_i are the corresponding response values. Thus, for each observation of the predictor measurement(s) $x_i, i = 1, 2, \dots, n$ there is a corresponding associated measurement of the response variable, $y_i, i = 1, 2, \dots, n$. We fit a model that relates the response to predictors. The fitted model is used to predict the response for future values. Commonly used models in this category include linear regression, logistic regression and several others. These models are discussed in Chapter 6.

Unsupervised Learning:

In unsupervised learning, no labels are given to the program. Unsupervised learning is where we have only input data (X) and no corresponding output variables. The learning algorithm is expected to find the structure in its input. The goal of unsupervised learning is to find hidden patterns in the large data. For example, suppose we have n measurements, x_i $i = 1, 2, 3, \dots, n$ but no corresponding response measurements, y_i . In this case, it is not possible to fit a regression model, since there is no response variable. This is the case of unsupervised learning where there is no response to predict. In such cases, prediction is not an objective, rather we try to understand the relationships between the variables or between the measurements. Instead of making predictions we may be interested to determine whether the measurements fall into distinct groups based on the analysis of $i = 1, 2, 3, \dots, n$ observations. This is commonly known as **Cluster analysis** which is a commonly used unsupervised learning model. Additional unsupervised learning models are discussed in Chapter 6.

Difference between Supervised and Unsupervised Learning

The difference between supervised and unsupervised learning is further explained in Table 2.1.

Table 2.1: Difference between Supervised and Unsupervised Learning

Supervised Learning	Unsupervised Learning
Supervised learning uses a set of input variables $(x_1, x_2, \dots, x_n)$ and an output variable, $y(x)$. An algorithm of the form $y = f(x)$ is used to learn the mapping function relating the input to output. Note that inputs are also known as features and the output is referred to as the target.	Unsupervised learning uses a set of input variables but not output variable. No labels are given to the learning algorithm. The algorithm is expected to find the structure in its input. The goals of unsupervised learning may be finding hidden patterns in the large data or feature learning. Thus, unsupervised learning can be a goal in itself or a means towards an end that is not based on general rules of teaching and training the algorithms.
This mapping function or the model relating the input and output variable is used to predict the	

output variable. The goal is to obtain the mapping function that is so accurate that it can use even the new set of data that is, the model can be used to predict the output variable as the new data becomes available. The name supervised learning means that in this process the algorithm is trained to learn from the training dataset where the learning process is supervised. In a supervised learning process, the expected output or the answer is known. The algorithm is designed to make predictions iteratively from the training data and is corrected by the analyst as needed. The learning process stops when the algorithm provides the desired level of performance and accuracy. The most used supervised problems are regression and classification problems. We briefly mentioned regression problems earlier. Time series prediction problems, random forest for classification and regression problems, and support vector machines for classification problems also fall in this category.	Unlike supervised learning, unsupervised algorithms are designed to devise and find the interesting structure in the data. The most used unsupervised learning problems are clustering and association problems. In clustering, a set of inputs is to be divided into groups. Unlike classification, the groups are not known beforehand, making this typically an unsupervised task. Association: Association problems are used to discover rules that describe association such as people that buy X also tend to buy Y.
---	---

Overview of Machine Learning Models

Machine Learning uses a vast set of tools for understanding data and solving problems. Here we will briefly discuss the supervised learning models followed by the unsupervised algorithms.

Supervised learning problems can be divided into two main categories: (1) ***Regression*** and (2) ***Classification problems***. The details of the **supervised learning** models are outlined in Figure 2.2.

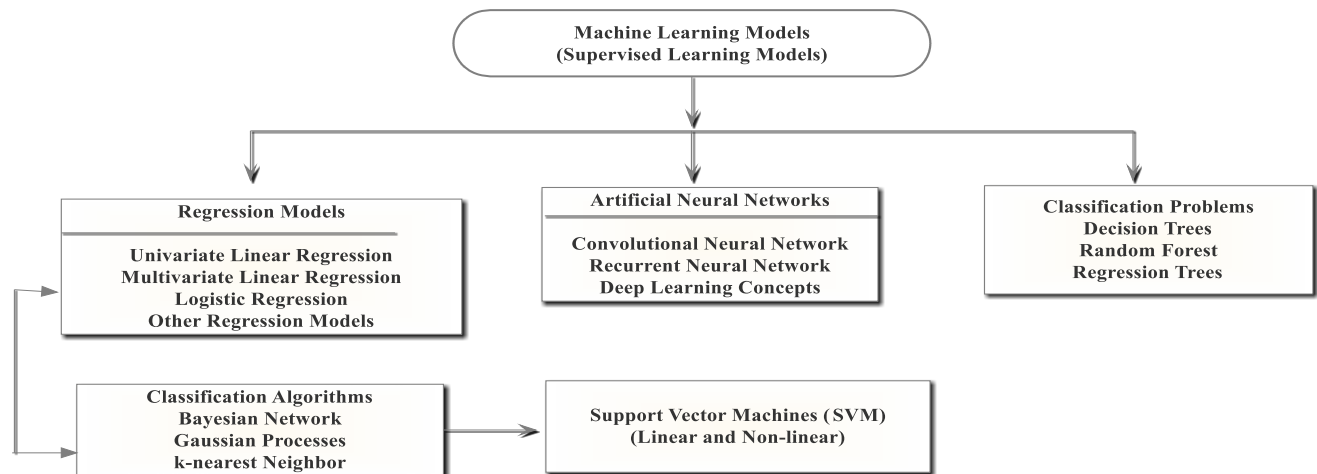


Figure 2.2: Supervised Machine Learning Models

Supervised learning refers to building models used for predicting an output using one or more input variables. This type of problem appears in several fields ranging from business, engineering, medicine, physics, astronomy, and others. Supervised learning problems fall into two broad categories – **regression** and **classification**.

Regression

Regression analysis is one of the most widely used supervised learning models. The method is used to identify the relationship between a dependent variable and one or more independent variables and is used to make predictions about future outcomes. When there is only one independent variable and one dependent variable, it is known as a simple linear regression. As the number of independent variables increases, it is referred to as multiple linear regression. For a simple linear regression, it tries to fit a line of best fit, which is calculated using the method of least squares. There are many other types of regression models. Regression is one of the widely used supervised learning models. Regression is a supervised problem where the outputs are continuous rather than discrete. Some examples of regression problems are predicting sales using advertisement expenditure, predicting the mileage of a car, predicting, demand, and so on.

Regression analysis is used to study and explain the mathematical relationship between two or more variables and predict one variable using the other variable or variables. By mathematical relationship we mean whether the relationship between the variables is linear or non-linear. Sometimes we may be interested in only two variables. In regression, the variable to be predicted is known as the dependent variable and the other variable is independent variable. The input variable is usually denoted by X . There may be more than one input variable. In this case they are denoted using a subscript written as $X_1, X_2, \dots, X_n$. The input variables go by different names, such as *predictors*, *independent variables*, or *features*. These terms are used interchangeably. The output variable is called the *dependent variable*, *response variable*, or *target* and is often denoted using the symbol, Y .

Suppose we are interested in the relationship between sales and advertising. Companies spend millions of dollars on advertising and expect that an increase in advertising expenditure will significantly improve sales. Thus, these two variables might be related. Other examples where two variables might be related are production cost and the volume of production, increase in summer temperature and the home heating cost, or the size of house in square feet and its price. Once the relationship between two variables is established, we can predict one variable using the other variable or variables. For example, if we can establish a relationship between the sales and advertising, we can predict the sales (Y) using advertising expenditure (x). This can be done using a mathematical relationship between the sales and advertising as

$$Y = f(x) + \varepsilon$$

In many cases there may be n independent variables or predictors, $X_1, X_2, \dots, X_n$. In this case, the relationship between the response Y and the predictors ($X_1, X_2, \dots, X_n$) can be written as

$$Y = f(X) + \varepsilon$$

This is an example of a ***supervised learning model*** as the problem can be modeled to predict an output variable, for example, sales based on advertising cost – an input variable.

Simple and Multiple Regression Models

When there are only two variables involved – one response and a single predictor variable, the relationship is explained and studied using the technique of ***simple regression analysis***. Many problems involve situations where many variables are involved. In such cases, we are interested in predicting one variable using several variables. The problem involving many variables is studied using the technique of ***multiple regression analysis*** where we are interested in predicting one variable using several input variables.

The objective in simple regression is to predict one variable using the other variable. The variable to be predicted is known as the ***dependent*** or ***response*** variable or the ***target*** and the other variable is known as the ***independent*** variable, ***feature***, or ***predictor***. Thus, the problem of simple regression involves one dependent and one independent variable. An example would be to predict the sales (the dependent variable) using the advertising expenditure (the independent variable). In multiple regression problems, where the relationship between multiple variables is of interest, the objective is to predict one variable – the dependent variable using the other variables known as independent variables. An example of multiple regression would be to predict the sales for a grocery chain using the food-item sales, non-food item sales, size of the store, and the operating hours (12 hours or 24 hours). *The multiple regression problem involves one dependent and two or more independent variables.* In regression the target variable is continuous. The other commonly used regression algorithms are Logistic regression, and Polynomial regression.

Other Regression Models

There are several regression models used in machine learning. The commonly used regression algorithms are:

- (1) Ordinary Least Squares Regression (OLSR) (2) Linear Regression/Multiple Regression
- (3) Logistic Regression (4) Ridge Regression (5) Non-linear regression (6) Lasso Regression and others.

Classification

Classification is another type of supervised learning problem. Classification algorithms assign test data into specific categories. The prediction task is a ***classification*** when the target variable is discrete. In classification, the labels are discrete categories where the inputs are divided into two or more classes, and the learner must produce a model that assigns unseen inputs to one or more (multi-label classification) of these classes. For example, a financial institution may study the potential of borrowers to predict whether a group of new borrowers may be classified as having a high degree of risk. Spam filtering is another example of classification, where the inputs are email messages that are classified into classes as “spam” and “no spam”.

Classification algorithms categorize the new data according to the observations of the training set.

Classification is a supervised learning technique where a training set is used to find similarities in classes. This means that the input data are divided into two or more classes or categories and the learner creates a model that assigns inputs to one or more of these classes. The objects are classified based on the training set of data. In a classification problem, the output variable is a category, such as e-mails can be classified as “spam” or “no spam” or a test can be classified as “negative” and “positive”. The algorithm that implements classification is known as the classifier.

Some of the most commonly used classification algorithms are **K-Nearest Neighbor** algorithm, Decision **T**ree algorithms, and others discussed in subsequent sections.

Classification can also be seen as a predictive modeling problem where a class label is predicted for a given set of input data for example, the algorithm may be designed to classify whether given handwritten characters is one of the known characters.

While both classification and regression are prediction problems, there are differences between them. Table 2.2 outlines the differences.

Table 2.2: Difference between Classification and Regression models

	Classification	Regression
1	Used to predict discrete target variables. The classification algorithm is used to classify a new instance into one of the two categories, 'yes' or 'no.'	Regression problems predict continuous variables. The target variable is continuous
2	The algorithm tries to find the best possible decision boundary that separates the two classes with maximum possible separation.	In the simple linear regression algorithm, we try to find the best-fit line or plane that represents a mathematical expression relating the target variable and the features.
3.	The model evaluation is done using: • Classification Accuracy (which is the ratio of correct predictions divided by the total number of input samples. The percentage gives the model accuracy. • Logarithmic loss or Log Loss. It works by penalizing false (False Positive) classifications. • Area under the curve: It is one of the widely used metrics used for binary classification. • F1 score: F1 scores are used to evaluate the performance of classification algorithms. This metric usually tells us about precision. The important metrics are: ○ Precision ○ Recall • Confusion Matrix	Model performance is evaluated using metrics like Mean Squared Error (MSE), coefficient of determination, R^2 , and MAPE. These are used to evaluate the performance of the regression models.
4.	The classification problems are binary Classification or Multi-Class Classification problems	Regression problems are Simple Regression, Multiple Regression, Ridge Regression, Non-linear

		regression models and their variations.
5	Objective is to predict categorical/class labels	Objective is to predict continuous numerical target.
6	Examples in this case are Spam detection, image recognition, sentiment analysis, disease characterization.	Examples are predicting profit, sales, or demand for a company, Stock price prediction, predicting home price, stock value prediction, forecasting demand.

Different Types of Classification Tasks in Machine Learning

Some common classification tasks in Machine learning are binary, and multi-class classification.

Binary Classification

In a binary classification, the objective is to classify the input data into two mutually exclusive categories. The training data is labeled in a binary format: true and false; positive and negative; 0 and 1; spam and not spam, etc. depending on the type of problem. For instance, we might want to detect whether a given image is a truck or a boat.

Multi-class classification

The multi-class classification algorithms has at least two mutually exclusive class labels, and the objective here is to predict to which class a given input example belongs to. For example, from the given images of boats, planes, and cars, the model is required to classify the correct image.

Commonly used binary Classification algorithms:

Most of the binary classification algorithms can also be used for multi-class classification. The commonly used classification algorithms are:

- Logistic Regression.

- Naive Bayes
- Support Vector Machine (SVM)
- K-Nearest Neighbors
- Neural Networks
- Deep Learning

Logistic regression: Linear regression is used when dependent variables are continuous, whereas, logistic regression is the appropriate model when the dependent variable is categorical; that is, they have binary outputs, such as "true" and "false" or "yes" and "no." While both regression models are applied to understand the relationships between inputs and output, logistic regression is mainly used to solve binary classification problems, such as spam identification.

Naive Bayes': Naïve Bayes or a Naive Bayes classifier is a probabilistic machine learning model that is used for classification task. The classifier is based on the Bayes' theorem. These classifiers are a collection of classification algorithms where all of them share a common principle, that is, every pair of features being classified is independent of each other.

Types of Naive Bayes Classifier:

Multinomial Naive Bayes: One of the common applications of this classifier is in document classification problem, i.e., whether a document belongs to the category of technology, finance, sports, politics, etc. The features or the predictors used by this classifier are the frequency of words present in the document.

Bernoulli Naive Bayes: In this classifier, the predictors are binary meaning the parameters used to predict the class variable take up only values 'yes' or 'no', for example if a word occurs in the text or not.

Gaussian Naive Bayes: in this case, the predictors are continuous with an assumption that the values have a gaussian distribution.

Some of the applications of Naive Bayes algorithms are sentiment analysis, spam filtering, recommendation systems etc. These algorithms are simple and easy to implement. The limitation of this algorithm is the assumption that the features or predictors are independent, which may not be the case in real world situations. This technique is primarily used in text classification, spam identification, and recommendation systems.

Support vector machines (SVM):

A support vector machine (SVM) is a machine learning algorithm that uses supervised learning algorithms to solve classification, regression, and outlier detection problems. The method is widely used for classification problems. The method works by performing optimal data transformation creating a hyperplane separating the two classes of data where the distance between two classes of data points is at its maximum. The main objective of the SVM algorithm is to identify a hyperplane that distinguishes and segregates the data points of different classes. SVMs have wide applications in healthcare, natural language processing, signal processing applications, and speech and image recognition fields.

Types of Support Vector Machines

Support vector machines can be classified as simple or linear SVM and kernel or non-linear SVM.

1. Simple or linear SVM

Linear SVMs are used to classify linearly separable data. When a dataset can be segregated into categories or classes using a single straight line, it is known as a linear SVM classifier. In this case, data is linearly distinct or separable. A simple SVM is typically used to address classification and regression analysis problems.

2. Kernel or non-linear SVM

When the data cannot be segregated into distinct categories using a straight line, then the non-linear classification using a kernel or non-linear SVM can be used. This type of classifier is referred to as a non-linear classifier. In this case, the classification can be performed with a non-linear data type by adding features into higher dimensions rather than a 2D space. The added features fit a hyperplane that helps easily separate classes or categories. Kernel SVMs is used to handle optimization problems with multiple variables. SVMs methods classify unknown data into known categories. They have applications in diverse fields. Some of the application areas include facial detection and expression classification, text categorization and handwriting recognition, and speech recognition. Some other areas where SVMs find applications are text categorization handwriting recognition, and Speech recognition.

The k-nearest-neighbor

The k-nearest-neighbor algorithm, or K-NN is a non-parametric supervised learning algorithm. It is a popular machine learning technique used for classification and regression tasks but mainly for classification. The "K" in KNN is a parameter that determines the number of nearest neighbors to include in the process. It is considered a non-parametric method because it doesn't make any assumptions about the distribution of underlying data. The KNN algorithm tries to determine to what group a data point belongs to by looking at the data points around it.

It uses proximity to make classifications or predictions about the grouping of an individual data point. The algorithm assumes that similar data points can be found near each other and based on this assumption it calculates the distance between data points using Euclidean distance (most preferred distance calculation method). and then it assigns a category based on the most frequent category or average.

Its ease of use and low processing time makes it a preferred algorithm, but as the test dataset grows, the processing time increases making it less attractive for classification tasks. KNN is typically used for recommendation engines and image recognition.

Random Forest Algorithm

Random forest is a machine learning algorithm that combines the output of multiple decision trees to reach a single result. It is used for both classification and regression problems. A decision tree is an algorithm used to classify data. It is like a probability tree that draws a clear pathway to a decision or outcome. It is a single starting point that branches off into two or more directions, with each branch of the tree showing different possible outcomes.

Random Forest grows multiple decision trees which are merged for a more accurate prediction.

The logic behind the Random Forest model is that it uses multiple uncorrelated decision trees that perform much better as a group than they do alone. When using Random Forest for classification, each tree provides a classification or a “vote.” The algorithm chooses the classification with the majority votes. In the case of Random Forest for regression, the forest picks the average of the outputs of all trees.

The regression and classification algorithms are discussed in detail in Chapter 6.

Neural Networks: Neural networks are also known as artificial neural networks (ANNs) and are comprised of node layers with an input layer, weights, a bias (or threshold), one or more hidden layers, and an output layer.

Each node or neuron is connected to another and has an associated weight and threshold. If the output of an individual node is above the specified threshold value, that node is activated and sends data to the next layer of the network. Otherwise, no data is passed along to the next layer of the network.

Neural networks rely on training data to learn and improve their accuracy over time. Modern neural networks are non-linear statistical data modeling tools. They are usually used to model complex relationships between inputs and outputs, and to find patterns in data. Some applications of neural networks are found in speech recognition or image recognition. One of the most well-known neural networks is Google's search algorithm.

In machine learning, neural networks have wide applications in *deep learning*. Deep learning is a subset of machine learning, which is essentially a neural network with three or more layers. A simple neural network is shown in Figure 2.3.

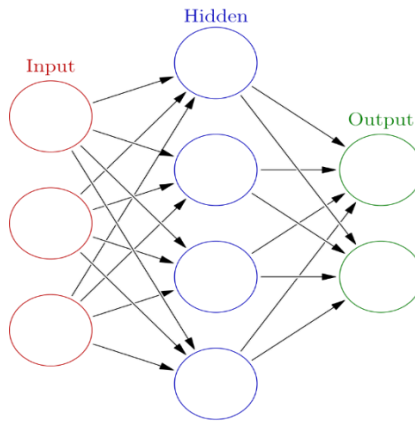


Figure 2.3: A simple Neural network

Deep learning ^[8]

Deep learning is a class of machine learning algorithms based on artificial neural networks (ANNs). Here, the learning can be supervised, semi-supervised or unsupervised.^[2]

Falling hardware prices and the development of GPUs for personal use in the last few years have contributed to the development of the concept of deep learning, which consists of multiple hidden layers in an artificial neural network. This approach tries to model the way the human brain processes light and sound into vision and hearing. Some successful applications of deep learning are computer vision and speech recognition.

Deep Learning uses multiple layers to progressively extract higher-level features from the input.

Deep-learning architectures include deep neural networks, deep belief networks, deep reinforcement learning, recurrent neural networks, convolutional neural networks, and [transformers](#). These have been applied to fields including computer vision, speech recognition, natural language processing, machine translation, bioinformatics, drug design, medical image analysis, climate science, material inspection and board game programs, where they have produced

results comparable to and in some cases surpassing human expert performance. [\[3\]\[4\]\[5\]](#) all are 27 in

[Reference](#)

Unsupervised learning

Unsupervised learning is a class of problems where there are inputs but no outputs or labels. Here are some examples to better understand the supervised and unsupervised learning models.

Unsupervised learning machine learning algorithms are used to analyze and cluster *unlabeled* datasets. These algorithms find hidden patterns or groupings in the data without the need for human intervention.

The unsupervised algorithms have applications in exploratory data analysis, customer segmentation, image processing, knowledge discovery in data and many others.

Common unsupervised learning approaches

Unsupervised learning models are used for three main tasks—**clustering**, **association**, and **dimensionality reduction**. These are discussed below with associated algorithms and approaches.

Clustering

Clustering is an unsupervised learning algorithm. Unlike supervised learning, the input is not labeled, and problem-solving is based on the experience that the algorithm gains by solving similar problems during the training schedule.

Clustering groups unlabeled data based on their similarities or differences. Clustering algorithms are used to process raw, unclassified data objects into groups represented by structures or patterns in the information.

Clustering organizes similar objects into groups in many **machine learning** applications. Clustering has applications across various fields, such as marketing, where clustering can be used to segment customers based on their buying patterns. This information can be used for promotion and advertising.

Types of Clustering Methods

In clustering, the formation of clusters uses several criteria. They use different parameters like shortest distance, graphs, and density of the data points. Grouping into clusters may be based on similarity measurements. There are different types of clustering approaches. They use different methods to group the data from the datasets. This section describes the clustering approaches. Some clustering methods and their applications are discussed here.

Clustering algorithms can be categorized into a few types, specifically exclusive, overlapping, hierarchical, and probabilistic.

Exclusive and Overlapping Clustering

Exclusive clustering is a form of grouping that stipulates a data point can exist only in one cluster. This can also be referred to as “hard” clustering. The K-means clustering algorithm is an example of exclusive clustering.

K-means clustering

k-means is the most widely used centroid-based clustering algorithm. Centroid-based algorithms are efficient but sensitive to initial conditions and outliers. This is an efficient, effective, and simple clustering algorithm.

A common example of an exclusive clustering method where data points are assigned into K groups where K is the number of clusters based on the distance from each group's centroid. The data points closest to a given centroid will be clustered under the same category. A larger K value will be indicative of smaller groupings, whereas a smaller K value will have larger grouping. K-means clustering is commonly used in market segmentation, document clustering, image segmentation, and image compression.

K-Means is a partition-based clustering technique that uses Euclidean distances between the points as a criterion for the formation of clusters. K-Means groups 'n' number of data points into 'a' predetermined 'k' number of clusters. A center is allocated to each cluster and each cluster is placed at farthest distance. The incoming data point is placed in a cluster with the closest cluster center. This process of placing the data points is repeated until all the data points get assigned to a cluster. After all the data points are allocated into clusters the cluster centers or centroids are recalculated. Once the 'k' new centroids are found, a new grouping is done between the nearest new centroid and the same data set points. The process is an iterative approach that may lead to a different k centroid value and their location. The iteration continues until the cluster centers do not change or the centroids do not move anymore.

Hierarchical clustering

Hierarchical clustering, also known as connectivity-based clustering. This unsupervised clustering algorithm is based on the principle that every object is connected to its neighbors depending on their proximity distance (degree of relationship). The clusters are represented in extensive hierarchical structures separated by a maximum distance required to connect the cluster parts. The clusters are represented as **Dendrograms**, where X-axis represents the objects that do not merge while Y-axis is the distance at which clusters merge. The similar data objects

have minimal distance falling in the same cluster, and the dissimilar data objects are placed farther in the hierarchy. Mapped data objects correspond to a Cluster amid discrete qualities concerning the multidimensional scaling, quantitative relationships among data variables, or cross-tabulation in some respects. The hierarchical clustering may vary in the data flow chosen in the following categories.

Hierarchical clustering can be categorized in two ways: agglomerative or divisive.

Agglomerative clustering is considered a “bottoms-up approach.” Its data points are isolated as separate groupings initially then they are merged iteratively on the basis of similarity until one cluster has been achieved. Four different methods are commonly used to measure similarity.

Ward’s linkage: In this method, the distance between two clusters is defined by the increase in the sum of squares after the clusters are merged.

Average linkage: This method is defined by the mean distance between two points in each cluster.

Complete (or maximum) linkage: This method is defined by the maximum distance between two points in each cluster.

Single (or minimum) linkage: This method is defined by the minimum distance between two points in each cluster.

Euclidean distance is the most common metric used to calculate these distances; however, other metrics, such as Manhattan distance, are also cited in clustering literature.

Divisive clustering can be defined as the opposite of agglomerative clustering. It takes a “top-down” approach where all the data points belong to one large cluster and try to divide the data into smaller groups based on a termination logic or a point beyond which there will be no further division of data points.

2. Mean Shift Clustering algorithm

Mean shift clustering is a nonparametric, simple, and flexible clustering technique. It is based on the estimated distribution of a given dataset, known as kernel density estimation. The basic principle of the algorithm is to assign the data points to the specified clusters recursively by shifting points towards the peak or highest density of data points. It is used in the image segmentation process.

3. Gaussian Mixture Model

The Gaussian mixture model (GMM) is a distribution-based clustering technique. It assumes that the data follows a Gaussian distribution. In this method of clustering, the probability of a point being a part of a cluster is inversely dependent on distance. This means that as the distance from distribution increases, the probability of a point belonging to the cluster decreases.

GMM determines probabilities and allocates them to data points in the ‘K’ number of clusters. Each of which has three parameters: Mean, Covariance and mixing probability. To compute these parameters GMM uses the Expectation Maximization technique.

Some of the applications of clustering are in market segmentation, retail marketing and sales, wireless network analysis, network traffic classification, regulating streaming services, health care and others.

In summary, clustering is an unsupervised method that takes a huge amount of unlabeled input data and clusters them as a group based on some similarity metrics.

There are several clustering methods. Some are explained above. The best method is selected based on the available data set and application.

The main cluster analysis techniques are Centroid Based/ Partition Clustering, Hierarchical Based Clustering, Distribution-Based Clustering, and Density-Based Clustering.

Association

Association is another type of unsupervised learning method that uses different rules to find relationships between variables in a given dataset. It is **a technique used to uncover hidden relationships between variables in large datasets**. This is a commonly used machine learning algorithm that has a wide range of applications in various fields. Some of the common applications of association rules are in market basket analysis, customer segmentation, and fraud detection. Association rules are useful for analyzing and predicting customer behavior. For example, “Customers who bought a particular item (e.g., baby formula), also bought diapers” or other recommendations. They also play an important role in customer analytics, product clustering, catalog design and store layout.

Dimensionality reduction

This technique is used when the number of features or independent variables in a dataset is too high. It reduces the number of data variables to a manageable size while preserving the data integrity. Dimensionality reduction is used in the preprocessing stage of data, when dealing high dimensional data or data with a large number of features, it is often useful to reduce the

dimensionality by projecting the data to a lower-dimensional subspace. The method should be used in a way so that it maintains the “essence” of the data. Many machine learning models use a huge quantity of data with a number of features that may affect the accuracy of the models. This technique makes the models manageable. **Semi-supervised Machine Learning Algorithms**

These algorithms fall in between supervised and unsupervised learning. They are designed with both labeled and unlabeled data for training and may typically contain a small amount of labeled data and a large amount of unlabeled data. These systems may, in some cases, be more efficient and are able to considerably improve learning accuracy. Semi-supervised learning methods are usually used when the labeled data requires skilled resources to train or learn from it. The acquired unlabeled data generally doesn't require additional resources.

- Semi-supervised learning **falls in between supervised and unsupervised learning**. This method uses a small amount of labeled data and a large amount of unlabeled data to train a model. SSL works for a variety of problems from classification and regression to clustering and association.
- Since the method uses small amounts of labeled data and large amounts of unlabeled data, it reduces effort, time and expense on manual annotation and reduces data preparation time.

Unsupervised and semi-supervised learning can be more appealing alternatives as it can be time-consuming and costly to label data appropriately for supervised learning.

Reinforcement learning (RL)

In this type of learning, the designed computer program interacts with a dynamic environment in which it has a specific goal to perform. This differs from standard supervised learning as no

input/output pairs are provided, which involves finding a balance between exploration (of uncharted territory) and exploitation (of current knowledge).^[6] Examples of reinforcement learning are playing a game against an opponent. In this type of learning, the computer program provides feedback in terms of rewards and punishments as it navigates its problem space.

Reinforcement learning is a machine learning **technique that trains the program to make decisions to achieve the most optimal results**. This method is based on rewarding desired behaviors and punishing undesired ones. Reinforcement learning is enforced through trial and error. It has applications in several disciplines including game theory, control theory, operations research, information theory, simulation-based optimization, multi-agent systems, swarm intelligence, and statistics.

Some Fundamental Concepts Related to Machine Learning

Here, we discuss important terms and concepts used in building machine learning models.

HoldOut

In the **holdout** method, the dataset is randomly divided into three subsets:

- **A training set** is a subset of the dataset that is used to build predictive models.
- **The validation set** is a subset of the dataset used to assess the performance of the model built in the training phase. This set allows for fine-tuning the parameters of the model and selecting the best-performing model. Validation may not be used for all types of models.
- **Test sets** are the subset of the data used to assess the future performance of the model. If a model fits well to the training set but not as good to the test set, it is an indication of over fitting.

Cross-Validation

The K-fold cross-validation is used in case of the availability of limited amount of data. The purpose is to obtain an unbiased estimation of the model performance.

Here, the data is divided into K subsets of equal sizes and the model fits K times leaving out one of the subsets from the training each time and using it as the test set.

If K equals the sample size, then this is called a “Leave-One-Out.”

Bootstrapping

- Bootstrapping is the technique used to make the estimations from the data by taking an average of the estimates from smaller data samples.
- The method involves iterative resampling of a dataset with replacement.
- Resampling instead of only estimating the statistics once on complete data set, provides better results.
- Repeating this multiple times helps to obtain a vector of estimates.
- Bootstrapping calculates the variance, expected value, and other relevant statistics of these estimates.

Signal and noise?

- In machine learning, “signal” is the true underlying pattern that we wish to learn from the data.
- “Noise” on the other hand, refers to irrelevant information or randomness in a dataset.

The higher the noise, the lower the quality of the signal—and the signal-to-noise ratio—is. In machine learning, noise similarly refers to unwanted behaviors within the data that provide a low signal-to-noise ratio. Essentially, $\text{data} = \text{signal} + \text{noise}$.

One key difference between noise and signal is that a signal is intentional and carries meaningful information, while noise is unintentional and does not carry any information. In other words, a signal is created to transmit a specific message, while noise is an unwanted addition that can interfere with the message.

A signal, mathematically a function, is a mechanism for conveying information. Audio, image, electrocardiograph (ECG) signals, radar signals, stock price movements, electrical current/voltages etc. are some of the examples.

A signal-to-noise ratio over 0 dB indicates that the signal level is greater than the noise level. The higher the ratio, the better the signal quality. For example, a Wi-Fi signal with S/N of 40 dB will deliver better network services than a signal with S/N of 20 dB.

Generally, a signal with an SNR value of 20 dB or more is recommended for data networks whereas an SNR value of 25 dB or more is recommended for networks that use voice applications.

For example, an SNR of 95 dB, means that the level of the audio signal is 95 dB higher than the level of the noise. Which, in turn, means that an SNR of 95 dB is better than one that is 80 dB.

Signal is the desired information being transmitted, while noise is any unwanted or random interference that can affect the quality of the signal. A strong signal is essential to overcome any interference that may be present, and various techniques are used to minimize the impact of noise on the signal.

Goodness-of-fit.

The goodness of fit refers to how closely a model's prediction matches the observed (true) values.

Overfitting, underfitting and bias-variance trade-off

Underfitting refers to the case when a model is too simple or has a small number of features or regularized too much. These make the model inflexible in learning from the dataset. *Underfitting* occurs when the model fails to capture important patterns in the training data set. Here the model is too simple or has too few explanatory variables.

This situation leads to high bias. The results show systematic lack of fit in certain reasons.

Overfitting is the condition when the model performs well on the training set but poorly on the test set. The model is too flexible with too many features for training data. Flexibility allows to ‘memorize’ the data including the noise. On the other hand, when the data contains too many features, trying to fit the predictors beyond a certain level might cause overfitting.

Bias and Variance

Bias and variance are two forms of prediction error in machine learning. Reducing errors from one might increase error from the other. This trade-off between too simple (high bias) vs. too complex (high variance) is a key concept in statistics and machine learning, and one that affects all supervised learning algorithms.

High variance means that the small changes in training data lead to big changes in the result. Variance is introduced with high sensitivity to variations in training data.

Bias is the type of error that occurs due to wrong assumptions about data. For example, assuming that the data is linear when, data follows some other complex function.

Bias and variance are inversely related. It is impossible that an ML model has low bias and a low variance. When the model is modified for a better fit it leads to low bias but high variance.

How to Compute Bias and Variance?

Bias is the distance from the average prediction and true value or the true value minus the mean(predictions).

Variance is the average deviation from the average prediction or the mean (prediction minus mean(predictions)).

Bias and Variance help to optimize the error in our model and keep it as low as possible. An optimized model will be sensitive to the patterns in our data and will be able to generalize to new data.

In general, both the bias and variance should be low so as to prevent overfitting and underfitting.

Lazy Learners and Eager Learners

The lazy learner simply stores the training data and waits until testing data appears. Classification is done based on the most related data stored training data. The k-nearest neighbor algorithm is considered a Lazy learner.

The other type of learner is known as **Eager learners** that builds a classification model based on the given training data before receiving data for classification. Because of the way the model is constructed, Eager Learners take a long time to train and less time to predict. Examples of this type of learner include Decision Trees, Naive Bayes, and Artificial Neural Networks. Compared to eager learners, lazy learners take less training time and more prediction time. One example of a lazy learner is k-nearest neighbors.

Summary:

This chapter explores the field of Machine Learning (ML), its importance, and the reasons behind the use of this field. Machine learning is a method of designing systems that can learn, adjust, and improve based on the data fed to them without being explicitly programmed. Machine learning applications are a way of analyzing and creating models from a very large sample or huge amounts of data commonly referred to as **big data**. Machine learning can perform tasks without being explicitly programmed to do so. This chapter introduced Machine Learning (ML) as a subset of Artificial Intelligence (AI) and explored this area, including the definition, needs, and types of problems machine learning can solve.

An overview of machine learning, along with the approach machine learning takes in solving problems, is discussed. The broad categories of machine learning tasks are: (1) Supervised learning (2) Unsupervised learning (3) Semi-supervised learning, and (4) Reinforcement learning, along with the models used in supervised and unsupervised learning. The difference between supervised and unsupervised learning along with the models and algorithms used under both are explained briefly. The two major categories of supervised learning problems are regression and classification. Various types of regression models used are: (1) Ordinary Least Squares Regression (OLSR) (2) Linear Regression/Multiple Regression, (3) Logistic Regression (4) Ridge Regression (5) Non-linear regression (6) Lasso Regression and others were briefly discussed.

Classification is the other class of supervised learning. Different classification approaches and the models under classification are outlined in this chapter. The common classification tasks in Machine learning are binary and multi-class classification. These along with the commonly used Classification algorithms are discussed briefly. The following classification algorithms are discussed:

Logistic Regression, Naive Bayes, Support Vector Machine (SVM), K-Nearest Neighbors.

The chapter also introduced Neural Networks and Deep Learning. Neural Networks and Deep Learning are the major application areas of machine learning. These were briefly introduced.

The other approaches to machine learning – semi-supervised learning and reinforcement learning were also discussed.

Finally, the chapter discussed the metrics to evaluate the accuracy of Classification and Regression Models and some important terms and concepts related to machine Learning.