**Machine learning**

Let's start with defining what Machine learning is. There are many technical and functional definitions for Machine learning, and some of them are as follows:

*"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 in T, as measured by P, improves with experience E." – Tom M. Mitchell*

*"Machine learning is the training of a model from data that generalizes a decision against a performance measure." – Jason Brownlee*

*"A branch of artificial intelligence in which a computer generates rules underlying or based on raw data that has been fed into it." – Dictionary.com*

*"Machine learning is a scientific discipline that is concerned with the design and development of algorithms that allow computers to evolve behaviors based on empirical data, such as from sensor data or databases." – Wikipedia*

The preceding definitions are fascinating and relevant. They either have an algorithmic, statistical, or mathematical perspective. Beyond these definitions, a single term or definition for Machine learning is the key to facilitating the definition of a problem-solving platform. It is *a mechanism for pattern search* and building intelligence into a machine to be able to learn, implying that it will be able to do better in the future from its own experience.

Drilling down a little more into what a pattern typically is, pattern search or pattern recognition is essentially the study of how machines perceive the environment, learn to discriminate behavior of interest from the rest, and be able to take reasonable decisions about categorizing the behavior. Humans more often perform this. The goal is to foster accuracy, speed, and avoid the possibility of inappropriate use of the system. Machine learning algorithms that are constructed this way handle building intelligence. Essentially, machines make sense of data in much the same way that humans do.

The primary goal of a Machine learning implementation is to develop a general purpose algorithm that solves a practical and focused problem. Some of the aspects that are important and need to be considered in this process include data, time, and space requirements. Most importantly, with the ability to be applied to a broad class of learning problems, the goal of a learning algorithm is to produce a result that is a rule and is as accurate as possible.

**Core Concepts and Terminology**

At the heart of machine-learning is knowing and using the data appropriately. This includes collecting the *right* data, cleansing the data, and processing the data using learning algorithms iteratively to build models using certain key features of data, and based on the hypotheses from these models, making predictions.

In this section, we will cover the standard nomenclature or terminology used in machine learning, starting from how to describe data, learning, modeling, algorithms, and specific machine learning tasks.

In general, for performing Machine learning, there are primarily two types of datasets required. The first dataset is usually manually prepared, where the input data and the expected output data are available and prepared. It is important that every piece of input data has an expected output data point available as this will be used in a supervised manner to build the rule. The second dataset is where we have the input data, and we are interested in predicting the expected output.

As a first step, the given data is segregated into three datasets: training, validation, and testing. There is no one hard rule on what percentage of data should be training, validation, and testing datasets. It can be 70-10-20, 60-30-10, 50-25-25, or any other values.

The training dataset refers to the data examples that are used to learn or build a classifier, for example. The validation dataset refers to the data examples that are verified against the built classifier and can help tune the accuracy of the output. The testing dataset refers to the data examples that help assess the performance of the classifier.

There are typically three phases for performing Machine learning:

**Phase 1—Training Phase**: This is the phase where training data is used to train the model by pairing the given input with the expected output. The output of this phase is the learning model itself.

**Phase 2—Validation and Test Phase**: This phase is to measure how good the learning model that has been trained is and estimate the model properties, such as error measures, recall, precision, and others. This phase uses a validation dataset, and the output is a sophisticated learning model.

**Phase 3—Application Phase**: In this phase, the model is subject to the real-world data for which the results need to be derived.



**Data**

Data forms the main source of learning in Machine learning. The data that is being referenced here can be in any format, can be received at any frequency, and can be of any size. When it comes to handling large datasets in the Machine learning context, there are some new techniques that have evolved and are being experimented with. There are also more big data aspects, including parallel processing, distributed storage, and execution. More on the large-scale aspects of data will be covered in the next chapter, including some unique differentiators.

When we think of data, dimensions come to mind. To start with, we have rows and columns when it comes to structured and unstructured data. This book will cover handling both structured and unstructured data in the machine learning context. In this section, we will cover the terminology related to data within the Machine learning context.

**Labeled and unlabeled data**

Data in the Machine learning context can either be labeled or unlabeled. Before we go deeper into the Machine learning basics, you need to understand this categorization, and what data is used when, as this terminology will be used throughout this book.

Unlabeled data is usually the raw form of the data. It consists of samples of natural or human-created artifacts. This category of data is easily available in abundance. For example, video streams, audio, photos, and tweets among others. This form of data usually has no explanation of the meaning attached. The unlabeled data becomes labeled data the moment a meaning is attached. Here, we are talking about attaching a "tag" or "label" that is required, and is mandatory, to interpret and define the relevance. For example, labels for a photo can be the details of what it contains, such as animal, tree, college, and so on, or, in the context of an audio file, a political meeting, a farewell party, and so on. More often, the labels are mapped or defined by humans and are significantly more expensive to obtain than the unlabeled raw data.

The learning models can be applied to both labeled and unlabeled data. We can derive more accurate models using a combination of labeled and unlabeled datasets. The following diagram represents labeled and unlabeled data. Both triangles and bigger circles represent labeled data and small circles represent unlabeled data. The application of labeled and unlabeled data is discussed in more detail in the following sections. You will see that supervised learning adopts labeled data and unsupervised learning adopts unlabeled data. Semi-supervised learning and deep learning techniques apply a combination of labeled and unlabeled data in a variety of ways to build accurate models.



**Tasks**

A task is a problem that the Machine learning algorithm is built to solve. It is important that we measure the performance on a task. The term "performance" in this context is nothing but the extent or confidence with which the problem is solved. Different algorithms when run on different datasets produce a different model. It is important that the models thus generated are not compared, and instead, the consistency of the results with different datasets and different models is measured.

**Algorithms**

After getting a clear understanding of the Machine learning problem at hand, the focus is on what data and algorithms are relevant or applicable. There are several algorithms available. These algorithms are either grouped by the learning subfields (such as supervised, unsupervised, reinforcement, semi-supervised, or deep) or the problem categories (such as Classification, Regression, Clustering or Optimization). These algorithms are applied iteratively on different datasets, and output models that evolve with new data are captured.

**Models**

Models are central to any Machine learning implementation. A model describes data that is observed in a system. Models are the output of algorithms applied to a dataset. In many cases, these models are applied to new datasets that help the models learn new behavior and also predict them. There is a vast range of machine learning algorithms that can be applied to a given problem. At a very high level, models are categorized as the following:

* Logical models
* Geometric models
* Probabilistic models

**Data and inconsistencies in Machine learning**

This section details all the possible data inconsistencies that may be encountered while implementing Machine learning projects, such as: under-fitting, over-fitting, data instability, unpredictable future. Fortunately, there are some established processes in place today to address these inconsistencies. The following sections cover these inconsistencies.

**Under-fitting**

A model is said to be under-fitting when it doesn't take into consideration enough information to accurately model the actual data. For example, if only two points on an exponential curve are mapped, this possibly becomes a linear representation, but there could be a case where a pattern does not exist. In cases like these, we will see increasing errors and subsequently an inaccurate model. Also, in cases where the classifier is too rigid or is not complex enough, under-fitting is caused not just due to a lack of data, but can also be a result of incorrect modeling. For example, if the two classes form concentric circles and we try to fit a linear model, assuming they were linearly separable, this could potentially result in under-fitting.

The accuracy of the model is determined by a measure called "power" in the statistical world. If the dataset size is too small, we can never target an optimal solution.

**Over-fitting**

This case is just the opposite of the under-fitting case explained before. While too small a sample is not appropriate to define an optimal solution, a large dataset also runs the risk of having the model over-fit the data. Over-fitting usually occurs when the statistical model describes noise instead of describing the relationships. Elaborating on the preceding example in this context, let's say we have 500,000 data points. If the model ends up catering to accommodate all 500,000 data points, this becomes over-fitting. This will in effect mean that the model is memorizing the data. This model works well as long as the dataset does not have points outside the curve. A model that is over-fit demonstrates poor performance as minor fluctuations in data tend to be exaggerated. The primary reason for over-fitting also could be that the criterion used to train the model is different from the criterion used to judge the efficacy of the model. In simple terms, if the model memorizes the training data rather than learning, this situation is seen to occur more often.

**Types of learning problems**

This section focuses on elaborating different learning problem categories. Machine learning algorithms are also classified under these learning problems. The following figure depicts various types of learning problems:



**Classification**

Classification is a way to identify a grouping technique for a given dataset in such a way that depending on a value of the target or output attribute, the entire dataset can be qualified to belong to a class. This technique helps in identifying the data behavior patterns. This is, in short, a discrimination mechanism. For example, a sales manager needs help in identifying a prospective customer and wants to determine whether it is worth spending the effort and time the customer demands. The key input for the manager is the customer's data, and this case is commonly referred to as **Total Lifetime Value** (**TLV**).

We take the data and start plotting blindly on a graph (as shown in the following graph) with the *x* axis representing the total items purchased and the *y* axis representing the total money spent (in multiples of hundreds of dollars). Now we define the criteria to determine, for example, whether a customer is good or bad. In the following graph, all the customers who spend more than 800 dollars in a single purchase are categorized as good customers (note that this is a hypothetical example or analysis).



**Clustering**

In many cases, the data analyst is just given some data and is expected to unearth interesting patterns that may help derive intelligence. The main difference between this task and that of a classification is that in the classification problem, the business user specifies what he/she is looking for (a good customer or a bad customer, a success or a failure, and so on).

**Forecasting, prediction or regression**

Similar to classification, forecasting or prediction is also about identifying the way things would happen in the future. This information is derived from past experience or knowledge. In some cases, there is not enough data, and there is a need to define the future through regression. Forecasting and prediction results are always presented along with the degree of uncertainty or probability. This classification of the problem type is also called **rule extraction**.

Let's take an example here, an agricultural scientist working on a new crop that she developed. As a trial, this seed was planted at various altitudes and the yield was computed. The requirement here is to predict the yield of the crop given the altitude details (and some more related data points). The relationship between yield gained and the altitude is determined by plotting a graph between the parameters. An equation is noted that fits most of the data points, and in cases where data does not fit the curve, we can get rid of the data. This technique is called regression.



**Simulation**

In addition to all the techniques we defined until now, there might be situations where the data in context itself has many uncertainty. For example, an outsourcing manager is given a task and can estimate with experience that the task can be done by an identified team with certain skills in 2-4 hours. Let's say the cost of input material may vary between $100-120 and the number of employees who come to work on any given day may be between 6 and 9. An analyst then estimates how much time the project might take. Solving such problems requires the simulation of a vast amount of alternatives. Typically in forecasting, classification, and unsupervised learning, we are given data and we really do not know how the data is interconnected. There is no equation to describe one variable as a function of others.

Essentially, data scientists combine one or more of the preceding techniques to solve challenging problems, which are: web search and information extraction, drug design, predicting capital market behavior, understanding customer behavior, designing robots.

**Optimization**

Optimization, in simple terms, is a mechanism to make something better or define a context for a solution that makes it the best.

Considering a production scenario, let's assume there are two machines that produce the desired product but one machine requires more energy for high speed in production and lower raw materials while the other requires higher raw materials and less energy to produce the same output in the same time. It is important to understand the patterns in the output based on the variation in inputs; a combination that gives the highest profits would probably be the one the production manager would want to know. You, as an analyst, need to identify the best possible way to distribute the production between the machines that gives him the highest profit.

The following image shows the point of highest profit when a graph was plotted for various distribution options between the two machines. Identifying this point is the goal of this technique.



Unlike the case of simulations where there is uncertainty associated with the input data, in optimization we not only have access to data, but also have the information on the dependencies and relationships between data attributes.

One of the key concepts in Machine learning is a process called **induction**. The following learning subfields use the induction process to build models. Inductive learning is a reasoning process that uses the results of one experiment to run the next set of experiments and iteratively evolve a model from specific information. The following figure depicts various subfields of Machine learning. These subfields are one of the ways the machine learning algorithms are classified.

**Supervised learning**

Supervised learning is all about operating to a known expectation and in this case, what needs to be analyzed from the data being defined. The input datasets in this context are also referred to as "labeled" datasets. Algorithms classified under this category focus on establishing a relationship between the input and output attributes, and use this relationship speculatively to generate an output for new input data points. In the preceding section, the example defined for the classification problem is also an example of supervised learning. Labeled data helps build reliable models but is usually expensive and limited. When the input and output attributes of the data are known, the key in supervised learning is the mapping between the inputs to outputs. There are quite a few examples of these mappings, but the complicated function that links up the input and output attributes is not known. A supervised learning algorithm takes care of this linking, and given a large dataset of input/output pairs, these functions help predict the output for any new input value.

**Unsupervised learning**

In some of the learning problems, we do not have any specific target in mind to solve. In the earlier section, we discussed clustering, which is a classification analyses where we do not start with a specific target in mind (good/bad, will buy/will not buy) and is hence referred to as unsupervised analyses or learning. The goal in this case is to decipher the structure in the data against the build mapping between input and output attributes of data and, in fact, the output attributes are not defined. These learning algorithms operate on an "unlabeled" dataset for this reason.

**Semi-supervised learning**

Semi-supervised learning is about using both labeled and unlabeled data to learn models better. It is important that there are appropriate assumptions for the unlabeled data and any inappropriate assumptions can invalidate the model. Semi-supervised learning gets its motivation from the human way of learning.

**Reinforcement learning**

Reinforcement learning is learning that focuses on maximizing the rewards from the result. For example, while teaching toddlers new habits, rewarding them every time they follow instructions works very well. In fact, they figure out what behavior helps them earn rewards. This is reinforcement learning, and it is also called credit assessment learning.

The most important thing is that in reinforcement learning the model is additionally responsible for making decisions for which a periodic reward is received. The results in this case, unlike supervised learning, are not immediate and may require a sequence of steps to be executed before the final result is seen. Ideally, the algorithm will generate a sequence of decisions that helps achieve the highest reward or utility.

The goal in this learning technique is to measure the trade-offs effectively by exploring and exploiting the data. For example, when a person has to travel from a point A to point B, there will be many ways that include travelling by air, water, road or by walking, and there is significant value in considering this data by measuring the trade-offs for each of these options. Another important aspect is the significance of a delay in the rewards. How would this affect learning? For example, in games like chess, any delay in reward identification may change the result.