Introduction to Machine Learning

# What is Machine Learning?

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

Machine learning is part of the broader field of artificial intelligence. This field is concerned with the capability of machines to perform activities using human-like intelligence. Within machine learning there are several different kinds of tasks or techniques:

* In **supervised learning**, every training sample from the dataset has a corresponding label or output value associated with it. As a result, the algorithm learns to predict labels or output values.
* In **unsupervised learning**, there are no labels for the training data. A machine learning algorithm tries to learn the underlying patterns or distributions that govern the data.
* In **reinforcement learning**, the algorithm figures out which actions to take in a situation to maximize a reward (in the form of a number) on the way to reaching a specific goal. This is a completely different approach than supervised and unsupervised learning.

# Components of Machine Learning

Nearly all tasks solved with machine learning involve three primary components:

* A machine learning model
* A model training algorithm
* A model inference algorithm

## Machine Learning Model

A machine learning model is a block of code or framework that can be modified to solve different but related problems based on the data provided.

DEFINITION: A model is an extremely generic program (or block of code), made specific by the data used to train it. It is used to solve different problems.

## Model Training

Model training algorithms work through an interactive process:

* Uses the model to process data and then compares the results against some end goal
* Gently nudges specific parts of the model in a direction that brings the model close to achieving the goal
* Iterating over these steps, get closer to what you want until you determine that you’re close enough that you can stop.

## Model Inference

Using your trained model to generate predictions. This process is referred to as model inference.

# Steps in Machine Learning Process



## STEP 1 - Define the Problem

### How do you define a machine learning task?

* Define a very specific task
* Identify the machine learning task we might use to solve this problem.

### What is a machine learning task?

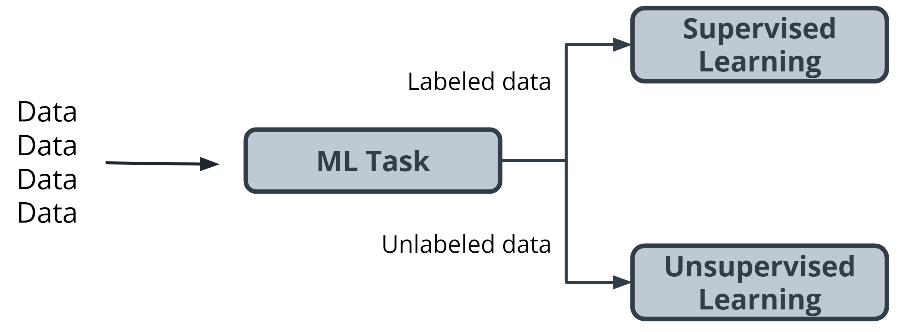
All model training algorithms, and the models themselves, take data as their input. Their outputs can be very different and are classified into a few different groups based on the *task* they are designed to solve. Often, we use the kind of data required to train a model as part of defining a machine learning task.

In this lesson, we will focus on two common machine learning tasks:

* Supervised Learning
* Unsupervised Learning

### Supervised and Unsupervised Learning

The presence or absence of labelling in your data is often used to identify a machine learning task.



#### Supervised Tasks

A task is *supervised* if you are using *labeled data* (data that already contains the solutions called *labels*). We are providing the model with labeled data and therefore, we are performing a *supervised machine learning task*.

#### Unsupervised Tasks

A task is considered to be *unsupervised* if you are using *unlabeled data*. This means you don’t need to provide the model with any kind of label or solution while the model is being trained.

### How do we classify tasks when we don’t have a label?

Unsupervised learning involves using data that doesn’t have a label. One common task is called **clustering**. Clustering helps to determine if there are nay naturally occurring groupings in the data.

#### Example: Identifying book micro-genres with unsupervised learning

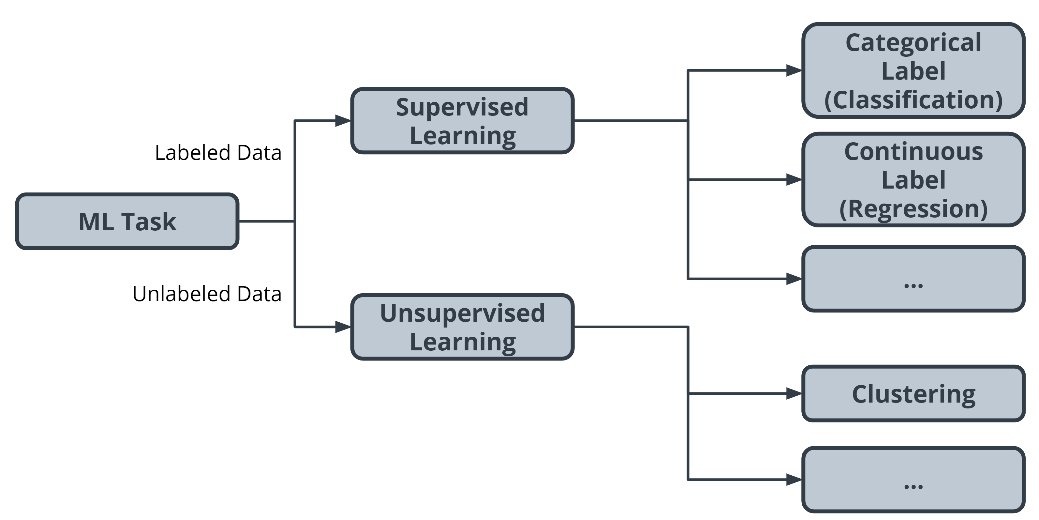
Imagine that you work for a company that recommends books to readers.

*The assumption* - You are fairly confident that micro-genres exist, and that there is one called *Teen Vampire Romance*. Because you don’t know which micro-genres exist, you can’t use **supervised learning** techniques.

This is where the **unsupervised learning** clustering technique might be able to detect some groupings in the data. The words and phrases used in the book description might provide some guidance on a book’s micro-genre.

### Further Classifying by using Label Types

Initially, we divided tasks based on the presence or absence of labeled data while training our model. Often, tasks are further defined by the types of label which is present.



In **supervised** learning, there are two main identifiers you will see in machine learning:

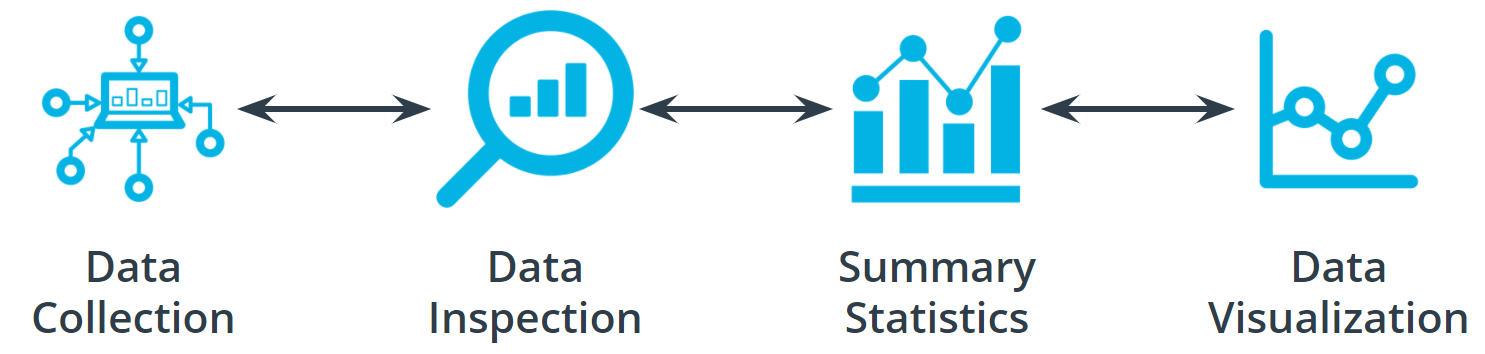
* A **categorical** label has a discrete set of possible values. Furthermore, when you work with categorical labels, you often carry out classification tasks, which are a part of the supervised learning family.
* A **continuous** (regression) label does not have a discrete set of possible values which often means you are working with numerical data.

In unsupervised learning, **clustering** is just one example. There are many other options, such as deep learning.

## STEP 2 - Build a Dataset

The next step in the machine learning process is to build a dataset that can be used to solve your machine learning-based problem. Understanding the data needed helps you select better models and algorithms so you can build more effective solutions.

### The Four Aspects of Working with Data



#### Data Collection

Data collection can be as straightforward as running the appropriate SQL queries or as complicated as building custom web scraper applications to collect data for your project. You might even have to run a model over your data to generate needed labels.

#### Data Inspection

The quality of your data will ultimately be the largest factor that affects how well you can expect your model to perform. As you inspect your data, look for:

* Outliers
* Missing or incomplete values
* Data that needs to be transformed or preprocessed so it's in the correct format to be used by your model

#### Summary Statistics

Models can assume how your data is structured.

Now that you have some data in hand it is a good best practice to check that your data is in line with the underlying assumptions of your chosen machine learning model.

With many statistical tools, you can calculate things like the mean, inner-quartile range (IQR), and standard deviation. These tools can give insight into the *scope*, *scale*, and *shape* of the dataset.

#### Data Visualization

You can use data visualization to see outliers and trends in your data and to help stakeholders understand your data.

## STEP 3 - Model Training

### Splitting your Dataset

The first step in model training is to randomly split the dataset. This allows you to keep some data hidden during training, so that data can be used to evaluate your model before you put it into production. Specifically, you do this to test against the bias-variance trade-off.

Splitting your dataset gives you two sets of data:

* Training dataset: The data on which the model will be trained. Most of your data will be here. Many developers estimate about 80%.
* Test dataset: The data withheld from the model during training, which is used to test how well your model will generalize to new data.

MODEL TRAINING - The model training algorithm iteratively updates a *model's parameters* to minimize some *loss function*.

Let's define those two terms:

* *Model parameters*: Model parameters are settings or configurations the training algorithm can update to change how the model behaves. Depending on the context, you’ll also hear other more specific terms used to describe model parameters such as *weights* and *biases*. Weights, which are values that change as the model learns, are more specific to neural networks.
* *Loss function:* A loss function is used to codify the model’s distance from this goal. For example, if you were trying to predict a number of snow cone sales based on the day’s weather, you would care about making predictions that are as accurate as possible. So, you might define a loss function to be “the average distance between your model’s predicted number of snow cone sales and the correct number.”

### Putting it All Together

The end-to-end training process is

* Feed the training data into the model.
* Compute the loss function on the results.
* Update the model parameters in a direction that reduces loss.

You continue to cycle through these steps until you reach a predefined stop condition. This might be based on a training time, the number of training cycles, or an even more intelligent or application-aware mechanism.

### Advice From the Experts

Remember the following advice when training your model:

* Practitioners often use machine learning frameworks that already have working implementations of models and model training algorithms. You could implement these from scratch, but you probably won't need to do so unless you’re developing new models or algorithms.
* Practitioners use a process called model selection to determine which model or models to use. The list of established models is constantly growing, and even seasoned machine learning practitioners may try many different types of models while solving a problem with machine learning.
* Hyperparameters are settings on the model which are not changed during training but can affect how quickly or how reliably the model trains, such as the number of clusters the model should identify.
* Be prepared to iterate.

Pragmatic problem solving with machine learning is rarely an exact science, and you might have assumptions about your data or problem which turn out to be false. Don’t get discouraged. Instead, foster a habit of trying new things, measuring success, and comparing results across iterations.

## Extended Learning

This information hasn't been covered in the above video but is provided for the advanced reader.

### Linear models

One of the most common models covered in introductory coursework, linear models simply describe the relationship between a set of input numbers and a set of output numbers through a linear function (think of y = mx + b or a line on a x vs ychart).

Classification tasks often use a strongly related logistic model, which adds an additional transformation mapping the output of the linear function to the range [0, 1], interpreted as “probability of being in the target class.” Linear models are fast to train and give you a great baseline against which to compare more complex models. A lot of media buzz is given to more complex models, but for most new problems, consider starting with a simple model.

### Tree-based models

Tree-based models are probably the second most common model type covered in introductory coursework. They learn to categorize or regress by building an extremely large structure of nested if/else blocks, splitting the world into different regions at each if/else block. Training determines exactly where these splits happen and what value is assigned at each leaf region.

For example, if you’re trying to determine if a light sensor is in sunlight or shadow, you might train tree of depth 1 with the final learned configuration being something like if (sensor\_value > 0.698), then return 1; else return 0. The tree-based model XGBoost is commonly used as an off-the-shelf implementation for this kind of model and includes enhancements beyond what is discussed here. Try tree-based models to quickly get a baseline before moving on to more complex models.

### Deep learning models

Extremely popular and powerful, deep learning is a modern approach based around a conceptual model of how the human brain functions. The model (also called a neural network) is composed of collections of neurons (very simple computational units) connected together by weights (mathematical representations of how much information to allow to flow from one neuron to the next). The process of training involves finding values for each weight.

Various neural network structures have been determined for modeling different kinds of problems or processing different kinds of data.

A short (but not complete!) list of noteworthy examples includes:

* **FFNN**: The most straightforward way of structuring a neural network, the Feed Forward Neural Network (FFNN) structures neurons in a series of layers, with each neuron in a layer containing weights to all neurons in the previous layer.
* **CNN**: Convolutional Neural Networks (CNN) represent nested filters over grid-organized data. They are by far the most commonly used type of model when processing images.
* **RNN**/**LSTM**: Recurrent Neural Networks (RNN) and the related Long Short-Term Memory (LSTM) model types are structured to effectively represent for loops in traditional computing, collecting state while iterating over some object. They can be used for processing sequences of data.
* **Transformer**: A more modern replacement for RNN/LSTMs, the transformer architecture enables training over larger datasets involving sequences of data.

### Machine Learning Using Python Libraries

* For more classical models (linear, tree-based) as well as a set of common ML-related tools, take a look at scikit-learn. The web documentation for this library is also organized for those getting familiar with space and can be a great place to get familiar with some extremely useful tools and techniques.
* For deep learning, mxnet, tensorflow, and pytorch are the three most common libraries. For the purposes of the majority of machine learning needs, each of these is feature-paired and equivalent.

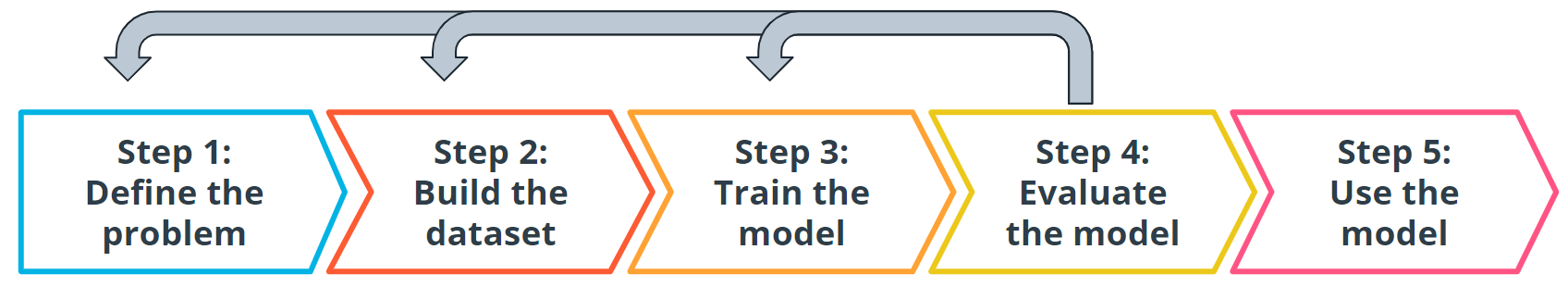
## STEP 4 - Model Evaluation

After you have collected your data and trained a model, you can start to evaluate how well your model is performing. The metrics used for evaluation are likely to be very specific to the problem you have defined. As you grow in your understanding of machine learning, you will be able to explore a wide variety of metrics that can enable you to evaluate effectively.

### Using Model Accuracy

Model accuracy is a fairly common evaluation metric. Accuracy is the fraction of predictions a model gets right.

### Remember: This Process is Iterative



Every step we have gone through is highly iterative and can be changed or re-scoped during the course of a project. At each step, you might find that you need to go back and reevaluate some assumptions you had in previous steps. Don't worry! This ambiguity is normal.

### Extended Learning

This information hasn't been covered in the above video but is provided for the advanced reader.

USING LOG LOSS

Log loss seeks to calculate how uncertain your model is about the predictions it is generating. In this context, uncertainty refers to how likely a model thinks the predictions being generated are to be correct.



For example, let's say you're trying to predict how likely a customer is to buy either a jacket or t-shirt.

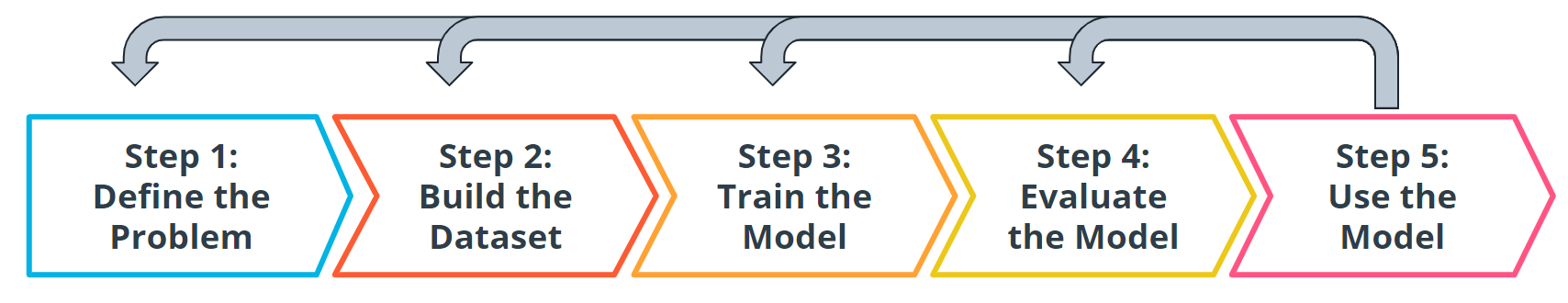
Log loss could be used to understand your model's uncertainty about a given prediction. In a single instance, your model could predict with 5% certainty that a customer is going to buy a t-shirt. In another instance, your model could predict with 80% certainty that a customer is going to buy a t-shirt. Log loss enables you to measure how strongly the model believes that its prediction is accurate.

In both cases, the model predicts that a customer will buy a t-shirt, but the model's certainty about that prediction can change.

## STEP 5 - Model Inference

Once you have trained your model, have evaluated its effectiveness, and are satisfied with the results, you're ready to generate predictions on real-world problems using unseen data in the field. In machine learning, this process is often called **inference**.

### Iterative Process



Even after you deploy your model, you're always monitoring to make sure your model is producing the kinds of results that you expect. There may be times where you reinvestigate the data, modify some of the parameters in your model training algorithm, or even change the model type used for training.

## Case Studies

* Supervised learning - Using machine learning to predict housing prices in a neighborhood based on lot size and number of bedrooms
* Unsupervised learning - Using machine learning to isolate micro-genres of books by analyzing the wording on the back cover description.
* Deep neural network - While this type of task is beyond the scope of this lesson, we wanted to show you the power and versatility of modern machine learning. You will see how it can be used to analyze raw images from lab video footage from security cameras, trying to detect chemical spills.