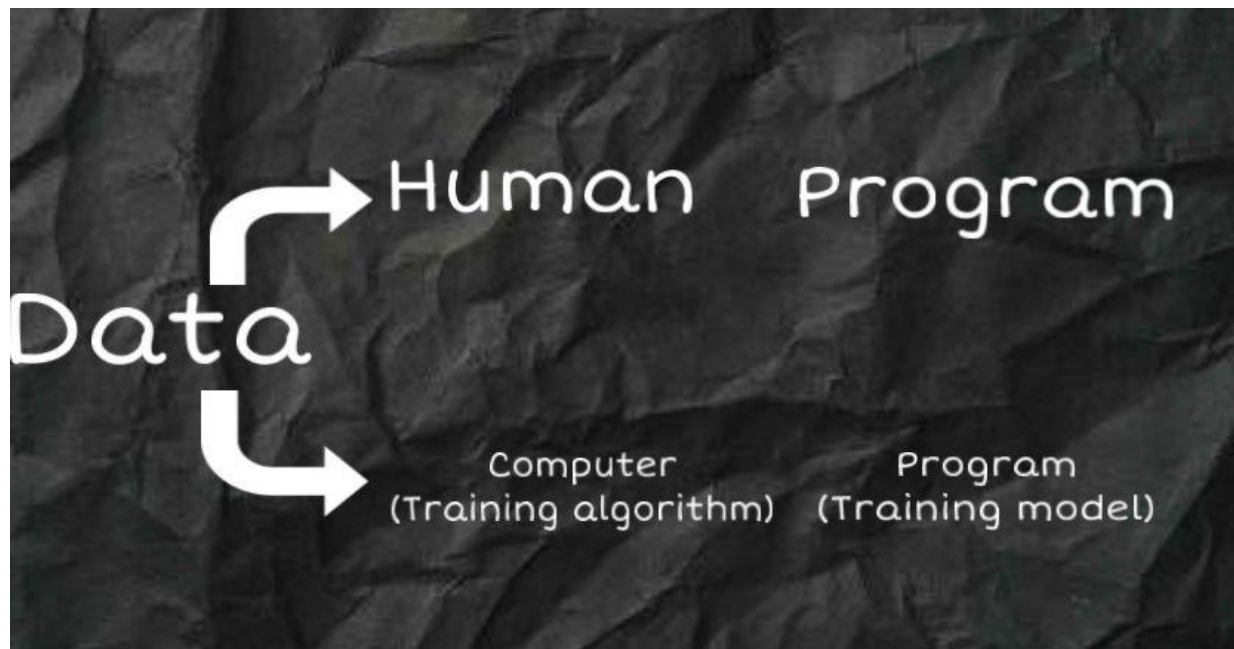


## What is machine learning?

A type of artificial intelligence known as machine learning (ML) is a contemporary method of software development that enables computers to solve issues using examples from the real world. Without being specifically programmed to do so, it enables computers to automatically learn from their experiences and get better at things.

## How is machine learning different?

In traditional software problem solving, a person analyses a problem and engineers a solution in code to solve that problem. However, for many real-world problems, this process can be time-consuming or even impossible because a correct solution would require taking a sizable number of edge cases into consideration.



**In machine learning, there are various types of tasks or techniques:**

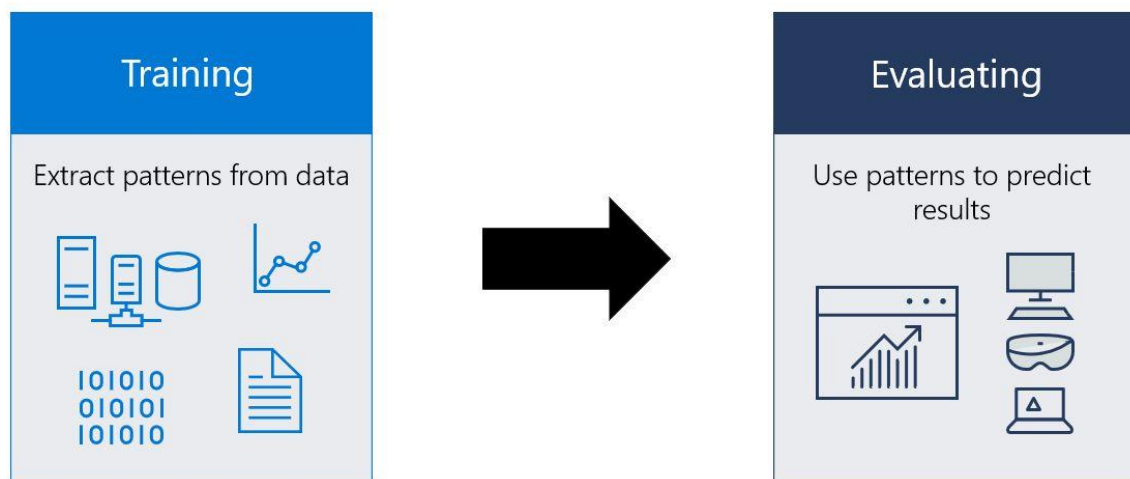


1. supervised: Each training sample in the dataset has a corresponding label or output value. The algorithm learns to predict labels or output values.

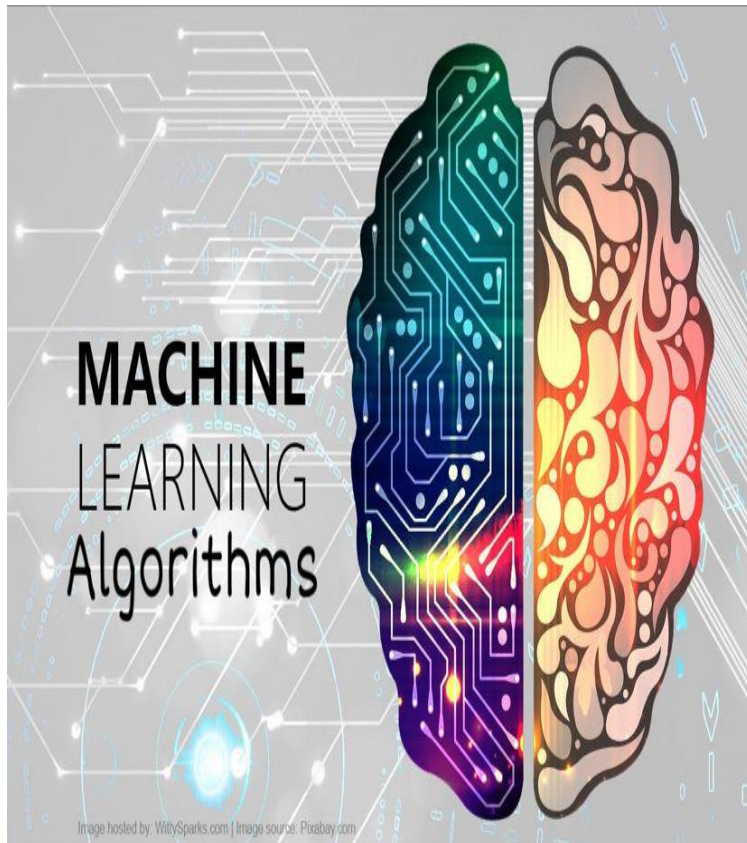
2. unsupervised: there are no labels for the training data. The algorithms attempt to learn the underlying patterns or distributions that govern the data.

3. Reinforcement: the algorithm determines which action to take in each situation in order to maximize a reward (in the form of a number) on the way to achieving a specific goal. This is a completely different approach than supervised and unsupervised learning.

In machine learning, the process involves creating a flexible component called a "model," which is trained using a specialized program called a "model training algorithm" to adjust it to real-world data. The resulting "trained model" can then be used to make predictions on outcomes not included in the training data set. This automates some of the statistical reasoning and pattern-matching that a problem solver would traditionally do, with the aim of generating predictions or identifying patterns in data to solve a specific problem. Essentially, the goal of machine learning is to utilize algorithms to create models that can learn from data and make predictions or classifications based on new, unseen data.



**All machine learning tasks involve three primary components. Here is a more straightforward explanation of each of these models in the real world:**



**Machine learning model:** A computer program that can recognize patterns in data, used in various applications such as predicting customer behavior, medical diagnosis, and stock market analysis.

**Model training algorithm:** A set of instructions that teaches a machine learning model how to recognize patterns in data, used to create tailored machine learning models for specific applications such as language processing and computer vision.

**Model interface algorithm:** A set of instructions that makes it easy for users to interact with machine learning models, used to create user-friendly applications such as medical diagnosis tools, trading platforms, and personalized shopping experiences.

**After discussing what machine learning is, the different types of machine learning, and the components of machine learning, it's time to dive deeper. We will go over the machine learning steps, and there are five of them:**



**Step 1: Define the problem.**

In this step, you need to clearly understand what you want to achieve with machine learning. Identify the task you want the model to perform, such as categorizing things or predicting values. Define the criteria for success, so you know when the model is doing a good job.

**Step 2: Build the dataset.**

Once you know the problem, gather the relevant data you'll use to train and test your model. Collect data from various sources and make sure it represents the problem you're solving. Clean and organize the data, so it's ready for the model to learn from.

**Step 3: Train the model.**

Now, it's time to build the machine learning model. Choose an algorithm or model architecture that suits your problem and data. Train the model by feeding it the prepared data. The model learns patterns and relationships in the data and adjusts its parameters to make accurate predictions.

**Step 4: Evaluate the model.**

After training, you need to check how well the model performs. Use a separate set of data (validation or test set) to assess its performance. Calculate metrics like accuracy or error to measure how close the model's predictions are to the actual values. This helps you understand if the model is working well or if improvements are needed.

**Step 5: Use the model.**

In this step, you deploy the model and use it to make predictions or inform decisions in real-life situations. It's important to monitor and update the model as needed for ongoing success.

## **Overfitting in Machine Learning: Balancing Act of Model Complexity**

Overfitting is a problem that many machine learning practitioners face while trying to create accurate predictive models. When a model learns the training data too well, it captures noise or random oscillations rather than underlying patterns. This is known as overfitting. This can result in poor generalisation performance on new, previously unknown data.

### **Key Aspects of Overfitting:**

#### **1. Model Complexity:**

- **Balancing Act:** Finding the right level of model complexity is crucial. A model that is too simple may underfit and fail to capture patterns, while a too complex model might overfit the noise in the training data.

#### **2. Training Data Fit:**

- **Overemphasis on Details:** Overfit models memorize the training data, including its quirks and outliers, rather than learning the true underlying patterns.

#### **3. Validation Performance:**

- **Drop in Generalization:** Overfit models may perform exceptionally well on the training data but poorly on new, unseen data (validation set), indicating a lack of generalization.

#### **4. Impact on Outliers:**

- **Sensitive to Noise:** Overfitting models can be highly sensitive to outliers and noisy data, leading to inaccurate predictions in real-world scenarios.

### **5. Addressing Overfitting:**

1. **Regularization Techniques:** Use methods like L1 or L2 regularization to prevent models from becoming too complex and fitting noise.

2. **Cross-Validation:** Test the model's performance on different subsets of data to see how well it generalizes, helping to spot overfitting.

3. Feature Selection: Be selective with features, choosing only the most relevant ones and avoiding noisy ones that could lead to overfitting.

4. Ensemble Methods: Combine multiple models together (like random forests or gradient boosting) to reduce the risk of overfitting present in individual models.

5. Data Augmentation: Increase the variety of training data using techniques like data augmentation, making the model more robust against overfitting.

The Art of Generalization:

Avoiding overfitting is like finding a balance between capturing important patterns and keeping the model simple. It's a skill that requires careful tweaking of models to ensure they work well with new, unseen data.

In the world of machine learning, dealing with overfitting highlights the need for thorough testing, smart choice of features, and a deep understanding of how complex the model should be while still being able to generalize effectively.