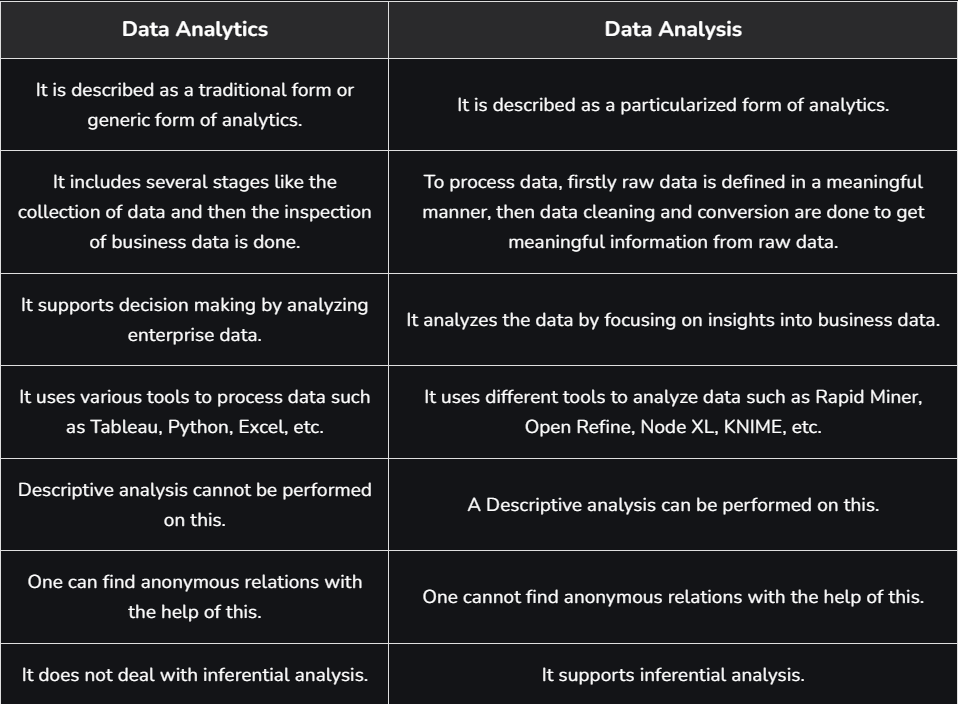
Data Science

*What is Data Science?*

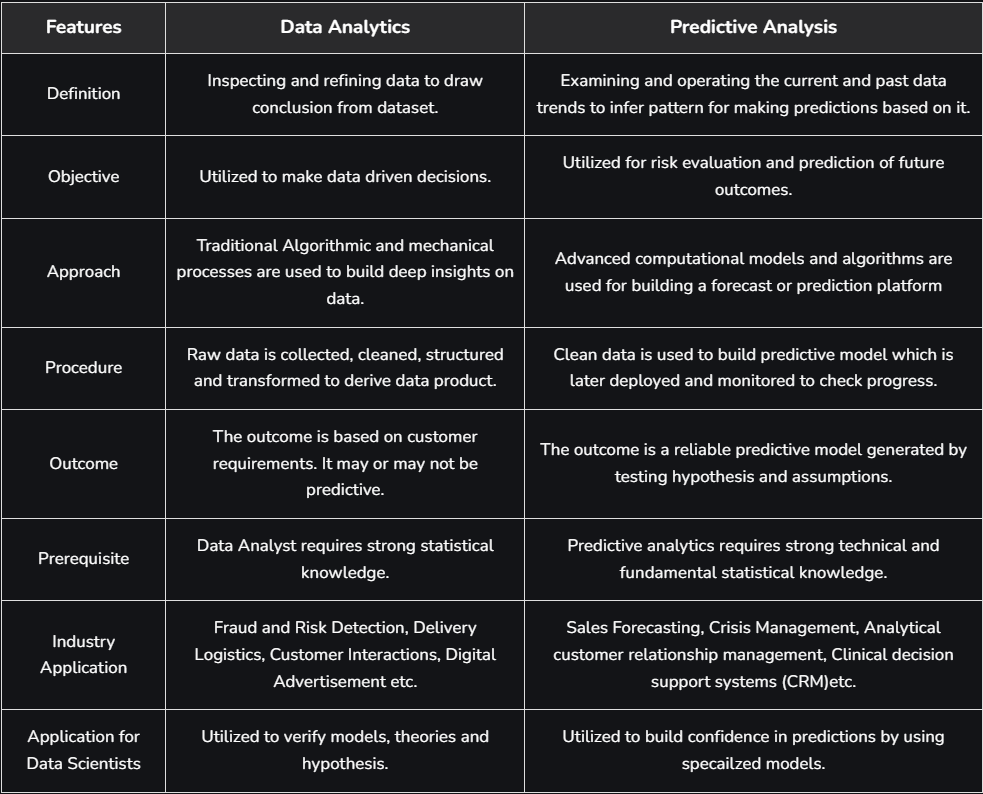
* Data science is an inter-disciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from many structural and unstructured data.[1][2] Data science is related to data mining, machine learning and big data.
* Data science is a "concept to unify statistics, data analysis and their related methods" in order to "understand and analyse actual phenomena" with data.
* It uses techniques and theories drawn from many fields within the context of mathematics, statistics, computer science, domain knowledge and information science.

*Analysis vs Analytics:*



The need for quantization arises from the growing demand to deploy deep learning models on devices with limited hardware resources. Here are the key motivations:

1. **Reduced Memory Footprint**: Lower-precision formats, such as INT8, take up less memory compared to FP32. For instance, an INT8 value requires 8 bits of memory, whereas an FP32 value requires 32 bits. This reduction in memory is crucial for deploying models on devices with limited storage capacity.
2. **Faster Inference**: Quantized models require fewer computations. Low-precision arithmetic is faster, especially on hardware that supports integer arithmetic or specialized accelerators like TPUs or NVIDIA Tensor Cores.
3. **Lower Power Consumption**: Quantization significantly reduces the energy required for inference, which is especially important for battery-operated devices like wearables, mobile phones, or IoT devices.
4. **Edge Device Deployment**: With quantized models, high-performance deep learning models can be deployed on edge devices without needing to rely on powerful cloud-based inference.



Artificial Intelligence:

Greek Mythology – Talos-

▪ Talos was a giant animated bronze warrior programmed to guard the island of Crete

1950 – Alan Turing

▪ Alan Turing published a landmark paper in which he speculated about the possibility of creating machines that think

▪ What he created is known as Turing Test which is used to determine whether or not the computer can think intelligently like human being--

1951 – Game AI

▪ Christopher Strachey wrote a checkers program and Dietrich Prinz wrote one for chess

1956 – The birth of AI

▪ John McCarthy first coined the term Artificial Intelligence at Dartmouth Conference

1959 – First AI laboratory

▪ MIT AI lab was first set up in 1959 and research on AI began

1960 – General Motors Robot

▪ First robot was introduced to General Motors assembly line

1961 – First chatbot

▪ The first AI chatbot called ELIZA was introduced in 1961

1997 - IBM Deep Blue

▪ IBM’s Deep Blue beats world champion Garry Kasparov in the game of chess

2005 - DARPA Grand Challenge-

▪ Stanford Racing Team’s autonomous robotic car, Stanley wins the 2005 DARPA Grand Challenge

2011 – IBM Watson->medical

▪ IBM’s question answering system, Watson, defeated the two grated Jeopardy champions Brad Ruther and Ken Jennings.

*What is AI?*

Artificial Intelligence (AI), sometimes called machine intelligence, is intelligence demonstrated by - machines, in contrast to the natural intelligence displayed by humans. Any device that perceives its environment and takes actions that maximize its chance of successfully achieving its goals context. The theory and development of computer system able to perform tasks normally requiring human intelligence, such as visual perception, speech recognition, decision making and translation. Often used to describe machines (or computers) that mimic "cognitive" functions that humans associate with the human mind, such as "learning" and "problem solving”.

Artificial intelligence (AI) is technology that enables computers and machines to simulate human learning, comprehension, problem solving, decision making, creativity and autonomy.

Aspects of AI (1955):

* Simulating higher functions of the human brain
* Programming a computer to use general languages translation NLP generation
* Arranging hypothetical neurons in a manner so that they can form concepts
* A way to determine and measure problem complexity
* Self-improvement
* more data => models more accuracy
* Abstraction: defined as the quality of dealing with ideas rather than events
* Randomness and creativity

AI applications

* Google’s search engine.
* JPMorgan Chase’s Contract Intelligence (COiN) platform uses AI, machine learning and image recognition software to analyse legal documents.
* IBM Watson: Healthcare organizations use IBM AI (Watson) technology for medical diagnosis.
* Google’s AI Eye Doctor can examine retina scans and identify a condition called as diabetic retinopathy which can cause blindness.
* Facebook uses ML and DL to detect facial features and tag your friends.
* Twitter uses AI to identify hate speech and terroristic language in the tweets
* Smart Assistants: Siri, Google Assistant, Alexa, Cortana
* Tesla automated cars
* Netflix uses AI for movie recommendations
* Spam filtering

Artificial intelligence (AI) is the simulation of human intelligence processes by machines, especially computer systems. It includes learning, reasoning, and self-correction. Examples of AI applications include expert systems, natural language processing (NLP), speech recognition, machine vision, and generative tools like ChatGPT and Perplexity.

As the hype around AI has accelerated, vendors have scrambled to promote how their products and services incorporate it. Often, what they refer to as "AI" is a well-established technology such as machine learning.

AI requires specialized hardware and software for writing and training machine learning algorithms. No single programming language is used exclusively in AI, but Python, R, Java, C++ and Julia are all popular languages among AI developers.

How does AI work?

In general, AI systems work by ingesting large amounts of labelled training data, analysing that data for correlations and patterns, and using these patterns to make predictions about future states.

Machine Learning

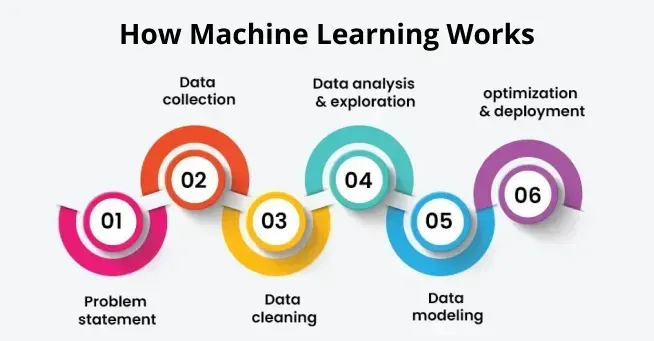
A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E

-Tom Mitchell, 1997

Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed

-Arthur Samuel, 1959

**Machine learning** is a branch of Artificial Intelligence that focuses on developing models and algorithms that let computers learn from data without being explicitly programmed for every task. In simple words, ML teaches the systems to think and understand like humans by learning from the data.



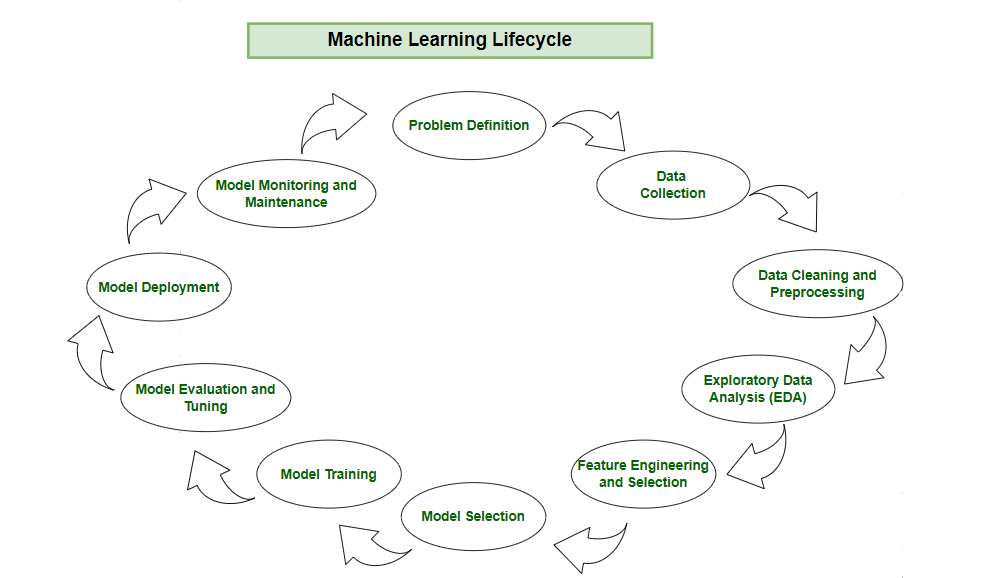
*Types:*

Machine Learning is mainly divided into three core types: Supervised, Unsupervised and Reinforcement Learning along with two additional types, Semi-Supervised and Self-Supervised Learning.

* [**Supervised Learning**](https://www.geeksforgeeks.org/supervised-machine-learning/)**:** Trains models on labelled data to predict or classify new, unseen data.
* [**Unsupervised Learning**](https://www.geeksforgeeks.org/ml-types-learning-part-2/): Finds patterns or groups in unlabelled data, like clustering or dimensionality reduction.
* [**Reinforcement Learning**](https://www.geeksforgeeks.org/what-is-reinforcement-learning/): Learns through trial and error to maximize rewards, ideal for decision-making tasks.

Machine learning lifecycle is a process that guides development and deployment of machine learning models in a structured way. It consists of various steps. Each step plays a crucial role in ensuring the success and effectiveness of the machine learning model. By following the machine learning lifecycle we can solve complex problems, can get data-driven insights and create scalable and sustainable models. The steps are:

1. **Problem Definition**
2. **Data Collection**
3. **Data Cleaning and Preprocessing**
4. **Exploratory Data Analysis (EDA)**
5. **Feature Engineering and Selection**
6. **Model Selection**
7. **Model Training**
8. **Model Evaluation and Tuning**
9. **Model Deployment**
10. **Model Monitoring and Maintenance**



**Step 1: Problem Definition**

In this initial phase we need to identify the business problem and frame it. By framing the problem in a comprehensive manner, team can establish foundation for machine learning lifecycle. Crucial elements such as **project objectives, desired outcomes and the scope of the task** are carefully designed during this stage.

Here are some steps for problem definition:

* **Collaboration:** Work together with stakeholders to understand and define the business problem.
* **Clarity:**Clearly write down the objectives, desired outcomes and scope of the task.
* **Foundation:**Establish a solid foundation for the machine learning process by framing the problem comprehensively.

**Step 2: Data Collection**

After problem definition, machine learning lifecycle progresses to [data collection](https://www.geeksforgeeks.org/methods-of-data-collection/). This phase involves systematic collection of datasets that can be used as raw data to train model. The quality and diversity of the data collected directly impact the robustness and generalization of the model.

During data collection we must consider the relevance of the data to the defined problem ensuring that the selected datasets consist all necessary features and characteristics. A well-organized approach for data collection helps in effective model training, evaluation and deployment ensuring that the resulting model is accurate and can be used for real world scenarios.

Here are some basic features of Data Collection:

* **Relevance:** Collect data should be relevant to the defined problem and include necessary features.
* **Quality:** Ensure data quality by considering factors like accuracy and ethical use.
* **Quantity:** Gather sufficient data volume to train a robust model.
* **Diversity:**Include diverse datasets to capture a broad range of scenarios and patterns.

**Step 3: Data Cleaning and Preprocessing**

With datasets in hand now we need to do [data cleaning and preprocessing](https://www.geeksforgeeks.org/data-cleansing-introduction/). Raw data is often messy and unstructured and if we use this data directly to train then it can lead to poor accuracy and capturing unnecessary relation in data, data cleaning involves addressing issues such as missing values, outliers and inconsistencies in data that could compromise the accuracy and reliability of the machine learning model.

Preprocessing is done by standardizing formats, scaling values and encoding categorical variables creating a consistent and well-organized dataset. The objective is to refine the raw data into a format that it is meaningful for analysis and training. By data cleaning and preprocessing we ensure that the model is trained on high-quality and reliable data.

Here are the basic features of Data Cleaning and Preprocessing:

* **Data Cleaning:**Address issues such as missing values, outliers and inconsistencies in the data.
* **Data Preprocessing:** Standardize formats, scale values, and encode categorical variables for consistency.
* **Data Quality:**Ensure that the data is well-organized and prepared for meaningful analysis.

**Step 4: Exploratory Data Analysis (EDA)**

To find patterns and characteristics hidden in the data [Exploratory Data Analysis (EDA)](https://www.geeksforgeeks.org/what-is-exploratory-data-analysis/) is used to uncover insights and understand the dataset's structure. During EDA patterns, trends and insights are provided which may not be visible by naked eyes. This valuable insight can be used to make informed decision.

Visualizations helps in showing statistical summary in easy and understandable way. It also helps in making choices in feature engineering, model selection and other critical aspects.

Here are the basic features of Exploratory Data Analysis:

* **Exploration:** Use statistical and visual tools to explore patterns in data.
* **Patterns and Trends:**Identify underlying patterns, trends and potential challenges within the dataset.
* **Insights:** Gain valuable insights for informed decisions making in later stages.
* **Decision Making:** Use EDA for feature engineering and model selection.

**Step 5: Feature Engineering and Selection**

[Feature engineering and selection](https://www.geeksforgeeks.org/what-is-feature-engineering/) is a transformative process that involve selecting only relevant features for model prediction. Feature selection refines pool of variables identifying the most relevant ones to enhance model efficiency and effectiveness.

Feature engineering involves selecting relevant features or creating new features by transforming existing ones for prediction. This creative process requires domain expertise and a deep understanding of the problem ensuring that the engineered features contribute meaningfully for model prediction. It helps accuracy while minimizing computational complexity.

Here are the basic features of Feature Engineering and Selection:

* **Feature Engineering:**Create new features or transform existing ones to capture better patterns and relationships.
* **Feature Selection:**Identify subset of features that most significantly impact the model's performance.
* **Domain Expertise:** Use domain knowledge to engineer features that contribute meaningfully for prediction[.](https://www.geeksforgeeks.org/power/)
* **Optimization:**Balance set of features for accuracy while minimizing computational complexity.

**Step 6: Model Selection**

For a good machine learning model, model selection is a very important part as we need to find model that aligns with our defined problem and the characteristics of the dataset. Model selection is a important decision that determines the algorithmic framework for prediction. The choice depends on the nature of the data, the complexity of the problem and the desired outcomes.

Here are the basic features of Model Selection:

* **Alignment:** Select a model that aligns with the defined problem and characteristics of the dataset.
* **Complexity:**Consider the complexity of the problem and the nature of the data when choosing a model.
* **Decision Factors:** Evaluate factors like performance, interpretability and scalability when selecting a model.
* **Experimentation:** Experiment with different models to find the best fit for the problem.

**Step 7: Model Training**

With the selected model the machine learning lifecycle moves to model training process. This process involves exposing model to historical data allowing it to learn patterns, relationships and dependencies within the dataset.

Model training is an iterative process where the algorithm adjusts its parameters to minimize errors and enhance predictive accuracy. During this phase the model fine-tunes itself for better understanding of data and optimizing its ability to make predictions. Rigorous training process ensure that the trained model works well with new unseen data for reliable predictions in real-world scenarios.

Here are the basic features of Model Training:

* **Training Data:** Expose the model to historical data to learn patterns, relationships and dependencies.
* **Iterative Process:**Train the model iteratively, adjusting parameters to minimize errors and enhance accuracy.
* **Optimization:** Fine-tune model to optimize its predictive capabilities.
* **Validation:** Rigorously train model to ensure accuracy to new unseen data.

**Step 8: Model Evaluation and Tuning**

[Model evaluation](https://www.geeksforgeeks.org/machine-learning-model-evaluation/) involves rigorous testing against validation or test datasets to test accuracy of model on new unseen data. We can use technique like accuracy, precision, recall and F1 score to check model effectiveness.

Evaluation is critical to provide insights into the model's strengths and weaknesses. If the model fails to acheive desired performance levels we may need to tune model again and adjust its hyperparameters to enhance predictive accuracy. This iterative cycle of evaluation and tuning is crucial for achieving the desired level of model robustness and reliability.

Here are the basic features of Model Evaluation and Tuning:

* **Evaluation Metrics:** Use metrics like accuracy, precision, recall and F1 score to evaluate model performance.
* **Strengths and Weaknesses:**Identify the strengths and weaknesses of the model through rigorous testing.
* **Iterative Improvement:**Initiate model tuning to adjust hyperparameters and enhance predictive accuracy.
* **Model Robustness:** Iterative tuning to achieve desired levels of model robustness and reliability.

**Step 9: Model Deployment**

Upon successful evaluation machine learning model is ready for deployment for real-world application. Model deployment involves integrating the predictive model with existing systems allowing business to use this for informed decision-making.

Here are the basic features of Model Deployment:

* **Integration:** Integrate the trained model into existing systems or processes for real-world application.
* **Decision Making:** Use the model's predictions for informed decision.
* **Practical Solutions:** Deploy the model to transform theoretical insights into practical use that address business needs.
* **Continuous Improvement:**Monitor model performance and make adjustments as necessary to maintain effectiveness over time.

The Machine Learning lifecycle is a comprehensive and recursive process that involves multiple steps from problem definition to model deployment and maintenance. Each step is essential for building a successful machine learning model that can provide valuable insights and predictions. By following the Machine learning lifecycle organizations, we can solve complex problems.

Types of machine learning

There are so many different types of Machine Learning systems that it is useful to classify them in broad categories, based on the following criteria

▪ Whether or not they are trained with human supervision

supervised, unsupervised, and Reinforcement Learning

▪ Whether or not they can learn incrementally on the fly -

online versus batch learning

▪ Whether they work by simply comparing new data points to known data points, or instead by detecting patterns in the training data and building a predictive model, much like scientists do--

instance-based versus model-based learning

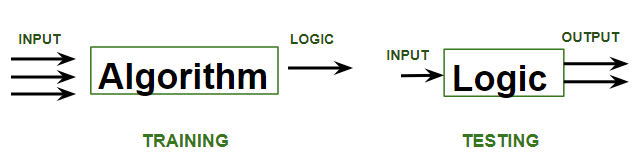
**Supervised Machine Learning**

**Supervised learning** is a type of machine learning where a model is trained on labelled data—meaning each input is paired with the correct output. the model learns by comparing its predictions with the actual answers provided in the training data. Over time, it adjusts itself to minimize errors and improve accuracy. The goal of supervised learning is to make accurate predictions when given new, unseen data. For example, if a model is trained to recognize handwritten digits, it will use what it learned to correctly identify new numbers it hasn't seen before.

Supervised learning can be applied in various forms, includingsupervised learning classification and supervised learning regression, making it a crucial technique in the field of artificial intelligence and supervised data mining.

A fundamental concept in supervised machine learning is learning a class from examples. This involves providing the model with examples where the correct label is known, such as learning to classify images of cats and dogs by being shown labelled examples of both. The model then learns the distinguishing features of each class and applies this knowledge to classify new images.

Supervised machine learning involves training a model on labelled data to learn patterns and relationships, which it then uses to make accurate predictions on new data.



* **Training**phase involves feeding the algorithm labelled data, where each data point is paired with its correct output. The algorithm learns to identify patterns and relationships between the input and output data.
* **Testing**phase involves feeding the algorithm new, unseen data and evaluating its ability to predict the correct output based on the learned patterns.

**Regression in machine learning:**

Regression in machine learning refers to a [**supervised learning**](https://www.geeksforgeeks.org/supervised-machine-learning/)technique where the goal is to predict a continuous numerical value based on one or more independent features. It finds relationships between variables so that predictions can be made. we have two types of variables present in regression:

* **Dependent Variable (Target)**: The variable we are trying to predict e.g. house price.
* **Independent Variables (Features)**: The input variables that influence the prediction e.g. locality, number of rooms.

Regression analysis problem works with if output variable is a real or continuous value such as “salary” or “weight”. Many different regression models can be used but the simplest model in them is linear regression.

**Types of Regression**

Regression can be classified into different types based on the number of predictor variables and the nature of the relationship between variables:

**1. Simple Linear Regression**

[**Linear regression**](https://www.geeksforgeeks.org/ml-linear-regression/)is one of the simplest and most widely used statistical models. This assumes that there is a linear relationship between the independent and dependent variables. This means that the change in the dependent variable is proportional to the change in the independent variables.For example, predicting the price of a house based on its size.

**2. Multiple Linear Regression**

[Multiple linear regression](https://www.geeksforgeeks.org/ml-multiple-linear-regression-using-python/) extends simple linear regression by using multiple independent variables to predict target variable.For example, predicting the price of a house based on multiple features such as size, location, number of rooms, etc.

**3. Polynomial Regression**

[Polynomial regression](https://www.geeksforgeeks.org/python-implementation-of-polynomial-regression/)is used to model with non-linear relationships between the dependent variable and the independent variables. It adds polynomial terms to the linear regression model to capture more complex relationships.For example, when we want to predict a non-linear trend like population growth over time we use polynomial regression.

**4. Ridge & Lasso Regression**

[Ridge & lasso regression](https://www.geeksforgeeks.org/ridge-regression-vs-lasso-regression/) are regularized versions of linear regression that help avoid overfitting by penalizing large coefficients.When there’s a risk of overfitting due to too many features we use these type of regression algorithms.

**5. Support Vector Regression (SVR)**

SVR is a type of regression algorithm that is based on the [Support Vector Machine (SVM)](https://www.geeksforgeeks.org/support-vector-machine-algorithm/) algorithm. SVM is a type of algorithm that is used for classification tasks but it can also be used for regression tasks. SVR works by finding a hyperplane that minimizes the sum of the squared residuals between the predicted and actual values.

**6. Decision Tree Regression**

[Decision tree](https://www.geeksforgeeks.org/decision-tree/) Uses a tree-like structure to make decisions where each branch of tree represents a decision and leaves represent outcomes. For example, predicting customer behaviour based on features like age, income, etc there we use decision tree regression.

**7. Random Forest Regression**

[Random Forest](https://www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/) is a ensemble method that builds multiple decision trees and each tree is trained on a different subset of the training data. The final prediction is made by averaging the predictions of all of the trees. For example customer churn or sales data using this.

**Regression Evaluation Metrics**

Evaluation in machine learning measures the performance of a model. Here are some popular evaluation metrics for regression:

* [**Mean Absolute Error (MAE):**](https://www.geeksforgeeks.org/how-to-calculate-mean-absolute-error-in-python/) The average absolute difference between the predicted and actual values of the target variable.
* [**Mean Squared Error (MSE):**](https://www.geeksforgeeks.org/python-mean-squared-error/)The average squared difference between the predicted and actual values of the target variable.
* [**Root Mean Squared Error (RMSE)**](https://www.geeksforgeeks.org/rmse-root-mean-square-error-in-matlab/)**:** Square root of the mean squared error.
* [**Huber Loss:**](https://www.geeksforgeeks.org/sklearn-different-loss-functions-in-sgd/) A hybrid loss function that transitions from MAE to MSE for larger errors, providing balance between robustness and MSE’s sensitivity to outliers.
* [R2 – Score](https://www.geeksforgeeks.org/python-coefficient-of-determination-r2-score/): Higher values indicate better fit ranging from 0 to 1.

**Applications of Regression**

* **Predicting prices:** Used to predict the price of a house based on its size, location and other features.
* **Forecasting trends:** Model to forecast the sales of a product based on historical sales data.
* **Identifying risk factors:** Used to identify risk factors for heart patient based on patient medical data.
* **Making decisions:** It could be used to recommend which stock to buy based on market data.

**Advantages of Regression**

* Easy to understand and interpret.
* Robust to outliers.
* Can handle both linear relationships easily.

**Disadvantages of Regression**

* Assumes linearity.
* Sensitive to situation where two or more independent variables are highly correlated with each other i.e multicollinearity.
* May not be suitable for highly complex relationships.

**Conclusion**

Regression in machine learning is a fundamental technique for predicting continuous outcomes based on input features. It is used in many real-world applications like price prediction, trend analysis and risk assessment. With its simplicity and effectiveness regression is used to understand relationships in data.

**Classification:**

**Classification teaches a machine to sort things into categories. It learns by looking at examples with labels (like emails marked "spam" or "not spam"). After learning, it can decide which category new items belong to, like identifying if a new email is spam or not**. For example, a classification model might be trained on dataset of images labelled as either**dogs** or **cats** and it can be used to predict the class of new and unseen images as dogs or cats based on their features such as colour, texture and shape.

**1. Binary Classification**

This is the simplest kind of classification. In binary classification, the goal is to sort the data into **two distinct categories**. Think of it like a simple choice between two options. Imagine a system that sorts emails into either **spam** or **not spam**. It works by looking at **different features of the email** like certain keywords or sender details, and decides whether it’s spam or not. It only chooses between these two options.

**2. Multiclass Classification**

Here, instead of just two categories, the data needs to be sorted into **more than two categories**. The model picks the one that best matches the input. Think of an image recognition system that sorts pictures of animals into categories like **cat**, **dog**, and **bird**.

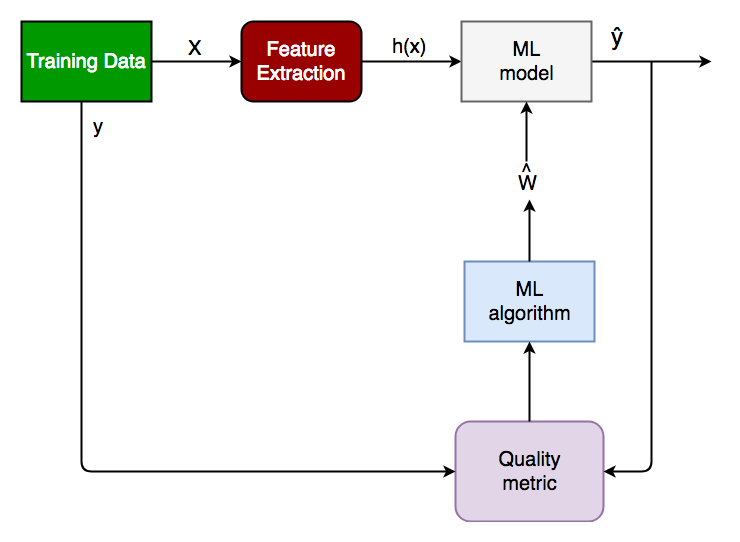
Basically, machine looks at the **features in the image (like shape, colour, or texture) and chooses which animal the picture is most likely to be based on the training it received.**

**3. Multi-Label Classification**

In [**multi-label classification**](https://www.geeksforgeeks.org/an-introduction-to-multilabel-classification/) single piece of data can belong to **multiple categories** at once. Unlike multiclass classification where each data point belongs to only one class, multi-label classification allows **datapoints to belong to multiple classes.**A movie recommendation system could tag a movie as both **action** and **comedy**. The system checks various features (like movie plot, actors, or genre tags) and assigns multiple labels to a single piece of data, rather than just one.

In machine learning, **classification** works by training a model to **learn patterns** from labelled data, so it can predict the category or class of new, unseen data. Here's how it works:

1. **Data Collection**: You start with a dataset where each item is labelled with the correct class (for example, "cat" or "dog").
2. **Feature Extraction**: The system identifies features (like colour, shape, or texture) that help distinguish one class from another. These features are what the model uses to make predictions.
3. **Model Training**: Classification - machine learning algorithm uses the labelled data to learn how to map the features to the correct class. It looks for patterns and relationships in the data.
4. **Model Evaluation**: Once the model is trained, it's tested on new, unseen data to check how accurately it can classify the items.
5. **Prediction**: After being trained and evaluated, the model can be used to predict the class of new data based on the features it has learned.
6. **Model Evaluation**: Evaluating a classification model is a key step in machine learning. It helps us check how well the model performs and how good it is at handling new, unseen data. Depending on the problem and needs we can use different metrics to measure its performance.



**Classification Modelling in Machine Learning**

**Classification modelling** refers to the process of using machine learning algorithms to categorize data into predefined classes or labels. These models are designed to handle both binary and multi-class classification tasks, depending on the nature of the problem. Let's see key characteristics of **Classification Models:**

1. **Class Separation**: Classification relies on distinguishing between distinct classes. The goal is to learn a model that can separate or categorize data points into predefined classes based on their features.
2. **Decision Boundaries**: The model draws decision boundaries in the feature space to differentiate between classes. These boundaries can be linear or non-linear.
3. **Sensitivity to Data Quality**: Classification models are sensitive to the quality and quantity of the training data. Well-labelled, representative data ensures better performance, while noisy or biased data can lead to poor predictions.
4. **Handling Imbalanced Data**: Classification problems may face challenges when one class is underrepresented. Special techniques like resampling or weighting are used to handle class imbalances.
5. **Interpretability**: Some classification algorithms, such as Decision Trees, offer higher interpretability, meaning it's easier to understand why a model made a particular prediction.

**Classification Algorithms**

Now, for implementation of any classification model it is essential to understand **Logistic Regression**, which is one of the most fundamental and widely used algorithms in machine learning for classification tasks. There are various types of **classifiers algorithms**. Some of them are:

**Linear Classifiers**: Linear classifier models create a linear decision boundary between classes. They are simple and computationally efficient. Some of the linear **classification**models are as follows:

* [Logistic Regression](https://www.geeksforgeeks.org/understanding-logistic-regression/)
* [Support Vector Machines having kernel = 'linear'](https://www.geeksforgeeks.org/support-vector-machine-algorithm/)
* [Single-layer Perceptron](https://www.geeksforgeeks.org/single-layer-perceptron-in-tensorflow/)
* [Stochastic Gradient Descent (SGD) Classifier](https://www.geeksforgeeks.org/stochastic-gradient-descent-classifier/)

**Non-linear Classifiers**: Non-linear models create a non-linear decision boundary between classes. They can capture more complex relationships between input features and target variable. Some of the non-linear **classification**models are as follows:

* [K-Nearest Neighbours](https://www.geeksforgeeks.org/k-nearest-neighbours/)
* [Kernel SVM](https://www.geeksforgeeks.org/major-kernel-functions-in-support-vector-machine-svm/)
* [Naive Bayes](https://www.geeksforgeeks.org/naive-bayes-classifiers/)
* [Decision Tree Classification](https://www.geeksforgeeks.org/decision-tree/)
* [Ensemble learning classifiers:](https://www.geeksforgeeks.org/ensemble-classifier-data-mining/)
* [Random Forests,](https://www.geeksforgeeks.org/random-forest-classifier-using-scikit-learn/)
* [AdaBoost,](https://www.geeksforgeeks.org/implementing-the-adaboost-algorithm-from-scratch/)
* [Bagging Classifier,](https://www.geeksforgeeks.org/ml-bagging-classifier/)
* [Voting Classifier,](https://www.geeksforgeeks.org/ml-voting-classifier-using-sklearn/)
* [Extra Trees Classifier](https://www.geeksforgeeks.org/ml-extra-tree-classifier-for-feature-selection/)
* [Multi-layer Artificial Neural Networks](https://www.geeksforgeeks.org/multi-layer-perceptron-learning-in-tensorflow/)

**Supervised Machine Learning Algorithms**

**Supervised learning** can be further divided into several different types, each with its own unique characteristics and applications. Here are some of the most common types of supervised learning algorithms:

* [**Linear Regression**](https://www.geeksforgeeks.org/ml-linear-regression/): Linear regression is a type of supervised learning regression algorithm that is used to predict a continuous output value. It is one of the simplest and most widely used algorithms in supervised learning.
* [**Logistic Regression**](https://www.geeksforgeeks.org/understanding-logistic-regression/): Logistic regression is a type of supervised learning classification algorithm that is used to predict a binary output variable.
* [**Decision Trees**](https://www.geeksforgeeks.org/decision-tree/): Decision tree is a tree-like structure that is used to model decisions and their possible consequences. Each internal node in the tree represents a decision, while each leaf node represents a possible outcome.
* [**Random Forests**](https://www.geeksforgeeks.org/random-forest-regression-in-python/): Random forests again are made up of multiple decision trees that work together to make predictions. Each tree in the forest is trained on a different subset of the input features and data. The final prediction is made by aggregating the predictions of all the trees in the forest.
* [**Support Vector Machine(SVM)**](https://www.geeksforgeeks.org/support-vector-machine-algorithm/): The SVM algorithm creates a hyperplane to segregate n-dimensional space into classes and identify the correct category of new data points. The extreme cases that help create the hyperplane are called support vectors, hence the name Support Vector Machine.
* [**K-Nearest Neighbours**](https://www.geeksforgeeks.org/k-nearest-neighbours/)**(KNN):**KNN works by finding k training examples closest to a given input and then predicts the class or value based on the majority class or average value of these neighbours. The performance of KNN can be influenced by the choice of k and the distance metric used to measure proximity.
* [**Gradient Boosting**](https://www.geeksforgeeks.org/ml-gradient-boosting/): Gradient Boosting combines weak learners, like [decision trees](https://www.geeksforgeeks.org/decision-tree/), to create a strong model. It iteratively builds new models that correct errors made by previous ones.
* [**Naive Bayes Algorithm**](https://www.geeksforgeeks.org/naive-bayes-classifiers/): The Naive Bayes algorithm is a supervised machine learning algorithm based on applying [Bayes' Theorem](https://www.geeksforgeeks.org/bayes-theorem/) with the “naive” assumption that features are independent of each other given the class label.

**Unsupervised learning:**

Unsupervised learning works by analysing unlabelled data to identify patterns and relationships. The data is not labelled with any predefined categories or outcomes, so the algorithm must find these patterns and relationships on its own. This can be a challenging task, but it can also be very rewarding, as it can reveal insights into the data that would not be apparent from a labelled dataset.

* **Unstructured data**: May contain noisy(meaningless) data, missing values, or unknown data
* **Unlabelled data**: Data only contains a value for input parameters, there is no targeted value(output). It is easy to collect as compared to the labelled one in the Supervised approach.

There are mainly 3 types of Algorithms which are used for Unsupervised dataset.

* **Clustering**
* **Association Rule Learning**
* **Dimensionality Reduction**

**1. Clustering Algorithms**

[Clustering](https://www.geeksforgeeks.org/clustering-in-machine-learning/) in unsupervised machine learning is the process of grouping unlabelled data into clusters based on their similarities. The goal of clustering is to identify patterns and relationships in the data without any prior knowledge of the data's meaning.

Broadly this technique is applied to group data based on different patterns, such as similarities or differences, our machine model finds. These algorithms are used to process raw, unclassified data objects into groups. For example, in the above figure, we have not given output parameter values, so this technique will be used to group clients based on the input parameters provided by our data.

***Some common clustering algorithms:***

* [***K-means Clustering***](https://www.geeksforgeeks.org/k-means-clustering-introduction/)***:*** *Groups data into K clusters based on how close the points are to each other.*
* [***Hierarchical Clustering***](https://www.geeksforgeeks.org/ml-hierarchical-clustering-agglomerative-and-divisive-clustering/)***:*** *Creates clusters by building a tree step-by-step, either merging or splitting groups.*
* [***Density-Based Clustering (DBSCAN)***](https://www.geeksforgeeks.org/dbscan-clustering-in-ml-density-based-clustering/)***:*** *Finds clusters in dense areas and treats scattered points as noise.*
* [***Mean-Shift Clustering***](https://www.geeksforgeeks.org/ml-mean-shift-clustering/)***:*** *Discovers clusters by moving points toward the most crowded areas.*
* [***Spectral Clustering***](https://www.geeksforgeeks.org/ml-spectral-clustering/)***:*** *Groups data by analysing connections between points using graphs.*

**2. Association Rule Learning**

[Association rule learning](https://www.geeksforgeeks.org/association-rule/) is also known as association rule mining is a common technique used to discover associations in unsupervised machine learning. This technique is a rule-based ML technique that finds out some very useful relations between parameters of a large data set. This technique is basically used for market basket analysis that helps to better understand the relationship between different products.

For e.g. shopping stores use algorithms based on this technique to find out the relationship between the sale of one product w.r.t to another's sales based on customer behaviour. **Like if a customer buys milk, then he may also buy bread, eggs, or butter**. Once trained well, such models can be used to increase their sales by planning different offers.

***Some common Association Rule Learning algorithms:***

* [***Apriori Algorithm***](https://www.geeksforgeeks.org/apriori-algorithm/)***:****Finds patterns by exploring frequent item combinations step-by-step.*
* [***FP-Growth Algorithm***](https://www.geeksforgeeks.org/frequent-pattern-growth-algorithm/)***:****An Efficient Alternative to Apriori. It quickly identifies frequent patterns without generating candidate sets.*
* [***Eclat Algorithm***](https://www.geeksforgeeks.org/ml-eclat-algorithm/)***:****Uses intersections of itemsets to efficiently find frequent patterns.*
* [***Efficient Tree-based Algorithms***](https://www.geeksforgeeks.org/introduction-to-tree-data-structure-and-algorithm-tutorials/)***:****Scales to handle large datasets by organizing data in tree structures.*

**3. Dimensionality Reduction**

Dimensionality reduction is the process of reducing the number of features in a dataset while preserving as much information as possible. This technique is useful for improving the performance of machine learning algorithms and for data visualization.

Imagine a dataset of 100 features about students (height, weight, grades, etc.). To focus on key traits, you reduce it to just 2 features: height and grades, making it easier to visualize or analyse the data.

*Here are some popular* ***Dimensionality Reduction algorithms****:*

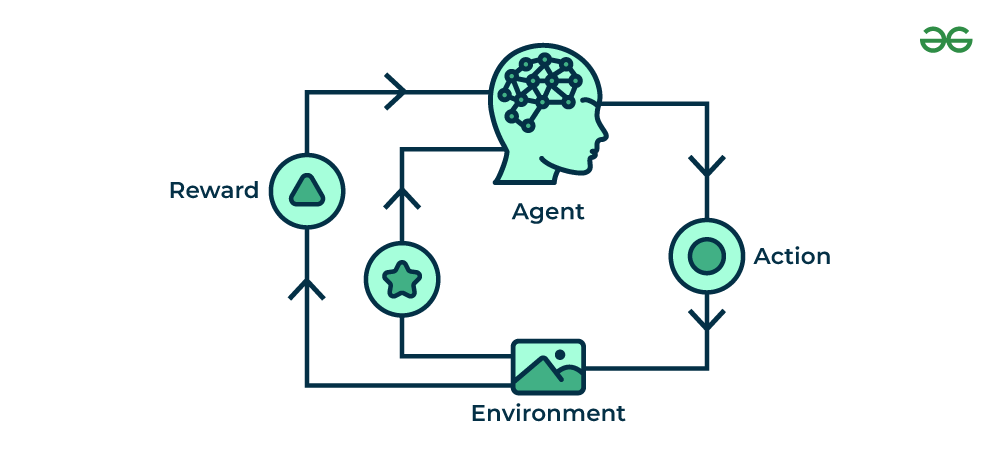
* [***Principal Component Analysis (PCA)***](https://www.geeksforgeeks.org/principal-component-analysis-pca/)***:****Reduces dimensions by transforming data into uncorrelated principal components.*
* [***Linear Discriminant Analysis (LDA)***](https://www.geeksforgeeks.org/ml-linear-discriminant-analysis/)***:****Reduces dimensions while maximizing class separability for classification tasks.*
* [***Non-negative Matrix Factorization (NMF***](https://www.geeksforgeeks.org/non-negative-matrix-factorization/)***):****Breaks data into non-negative parts to simplify representation.*
* [***Locally Linear Embedding (LLE)***](https://www.geeksforgeeks.org/locally-linear-embedding-in-machine-learning/)***:****Reduces dimensions while preserving the relationships between nearby points.*
* [***Isomap***](https://www.geeksforgeeks.org/isomap-a-non-linear-dimensionality-reduction-technique/)***:****Captures global data structure by preserving distances along a manifold.*

**Applications of Unsupervised learning**

**Unsupervised learning has diverse applications across industries and domains. Key applications include:**

* **Customer Segmentation:** Algorithms cluster customers based on purchasing behaviour or demographics, enabling targeted marketing strategies.
* **Anomaly Detection:** Identifies unusual patterns in data, aiding fraud detection, cybersecurity, and equipment failure prevention.
* **Recommendation Systems**: Suggests products, movies, or music by analysing user behaviour and preferences.
* **Image and Text Clustering**: Groups similar images or documents for tasks like organization, classification, or content recommendation.
* **Social Network Analysis**: Detects communities or trends in user interactions on social media platforms.
* **Astronomy and Climate Science:** Classifies galaxies or groups weather patterns to support scientific research

**Reinforcement Learning:**



**Reinforcement Learning (RL)**is a branch of machine learning that focuses on how agents can learn to make decisions through trial and error to maximize cumulative rewards. RL allows machines to learn by interacting with an environment and receiving feedback based on their actions. This feedback comes in the form of**rewards or penalties**.

Reinforcement Learning revolves around the idea that an agent (the learner or decision-maker) interacts with an environment to achieve a goal. The agent performs actions and receives feedback to optimize its decision-making over time.

* **Agent**: The decision-maker that performs actions.
* **Environment**: The world or system in which the agent operates.
* **State**: The situation or condition the agent is currently in.
* **Action**: The possible moves or decisions the agent can make.
* **Reward**: The feedback or result from the environment based on the agent’s action.

Here’s a breakdown of RL components:

* **Policy**: A strategy that the agent uses to determine the next action based on the current state.
* **Reward Function**: A function that provides feedback on the actions taken, guiding the agent towards its goal.
* **Value Function**: Estimates the future cumulative rewards the agent will receive from a given state.
* **Model of the Environment**: A representation of the environment that predicts future states and rewards, aiding in planning.

**Types of Reinforcements in RL**

**1. Positive Reinforcement**

Positive Reinforcement is defined as when an event, occurs due to a particular behaviour, increases the strength and the frequency of the behaviour. In other words, it has a positive effect on behaviour.

* **Advantages**: Maximizes performance, helps sustain change over time.
* **Disadvantages**: Overuse can lead to excess states that may reduce effectiveness.

**2. Negative Reinforcement**

Negative Reinforcement is defined as strengthening of behaviour because a negative condition is stopped or avoided.

* **Advantages**: Increases behaviour frequency, ensures a minimum performance standard.
* **Disadvantages**: It may only encourage just enough action to avoid penalties.

**Application of Reinforcement Learning**

1. **Robotics:**RL is used to automate tasks in structured environments such as manufacturing, where robots learn to optimize movements and improve efficiency.
2. **Game Playing:** Advanced RL algorithms have been used to develop strategies for complex games like chess, Go, and video games, outperforming human players in many instances.
3. **Industrial Control:**RL helps in real-time adjustments and optimization of industrial operations, such as refining processes in the oil and gas industry.
4. **Personalized Training Systems:**RL enables the customization of instructional content based on an individual's learning patterns, improving engagement and effectiveness.

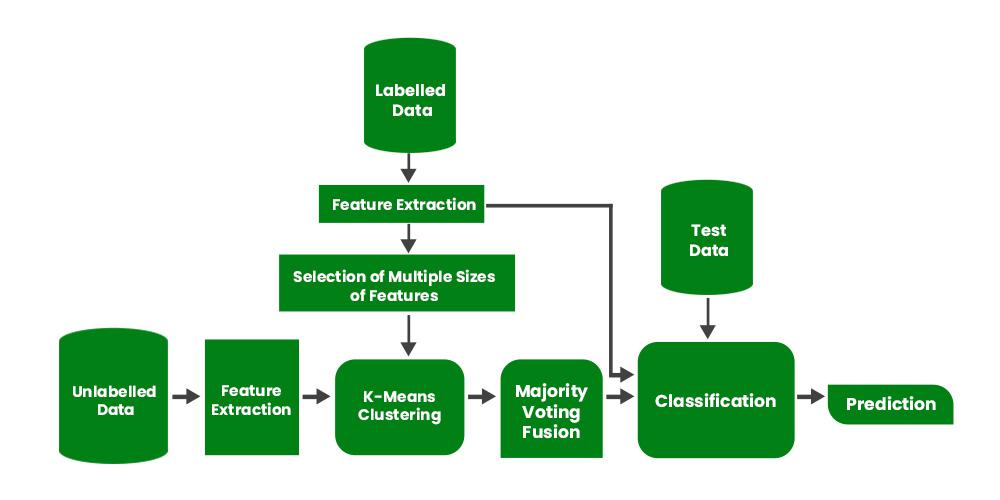
**Semi-Supervised Learning in ML:**

Semi-supervised learning is a type of [machine learning](https://www.geeksforgeeks.org/machine-learning/) that falls in between supervised and unsupervised learning. It is a method that uses a small amount of labelled data and a large amount of unlabelled data to train a model. The goal of semi-supervised learning is to learn a function that can accurately predict the output variable based on the input variables, similar to supervised learning. However, unlike supervised learning, the algorithm is trained on a dataset that contains both labelled and unlabelled data.

Semi-supervised learning is particularly useful when there is a large amount of unlabelled data available, but it's too expensive or difficult to label all of it.

**Examples of Semi-Supervised Learning**

* [**Text classification**](https://www.geeksforgeeks.org/semi-supervised-learning-examples/): In text classification, the goal is to classify a given text into one or more predefined categories. Semi-supervised learning can be used to train a text classification model using a small amount of labeled data and a large amount of unlabeled text data.
* [**Image classification**](https://www.geeksforgeeks.org/python-image-classification-using-keras/): In image classification, the goal is to classify a given image into one or more predefined categories. Semi-supervised learning can be used to train an image classification model using a small amount of labeled data and a large amount of unlabeled image data.
* [**Anomaly** **detection**](https://www.geeksforgeeks.org/machine-learning-for-anomaly-detection/): In anomaly detection, the goal is to detect patterns or observations that are unusual or different from the norm



Batch Learning

* In batch learning, the system is incapable of learning incrementally.
* It must be trained using all the available data.
* This will generally take a lot of time and computing resources, so it is typically done offline.
* First the system is trained, and then it is launched into production and runs without learning anymore, it just applies what it has learned.
* This is also called as offline learning.

Batch Learning - cons

▪ If you want a batch learning system to know about new data, you need to train a new version of the system from scratch on the full dataset, then stop the old system and replace it with the new one

▪ The whole process of training, evaluating, and launching a Machine Learning system can be automated easily

▪ Training using the full set of data can take many hours

▪ Typically train a new system only every 24 hours or even just weekly

▪ Training on the full set of data requires a lot of computing resources (CPU, memory space, disk space, disk I/O, network I/O)

Online Learning

In online learning, you train the system incrementally by feeding it data instances sequentially, either individually or in small groups called mini-batches.

▪ Each learning step is fast and cheap, so the system can learn about new data on the fly, as it arrives.

▪ Online learning is great for systems that receive data as a continuous flow (e.g., stock prices) and need to adapt to change rapidly or autonomously.

▪ It is also a good option if you have limited computing resources.

▪ once an online learning system has learned about new data instances, it does not need them anymore, so you can discard them.

▪ This can save a huge amount of space.

▪ Online learning algorithms can also be used to train systems on huge datasets that cannot fit in one.machine’s main memory (this is called out-of-core learning).

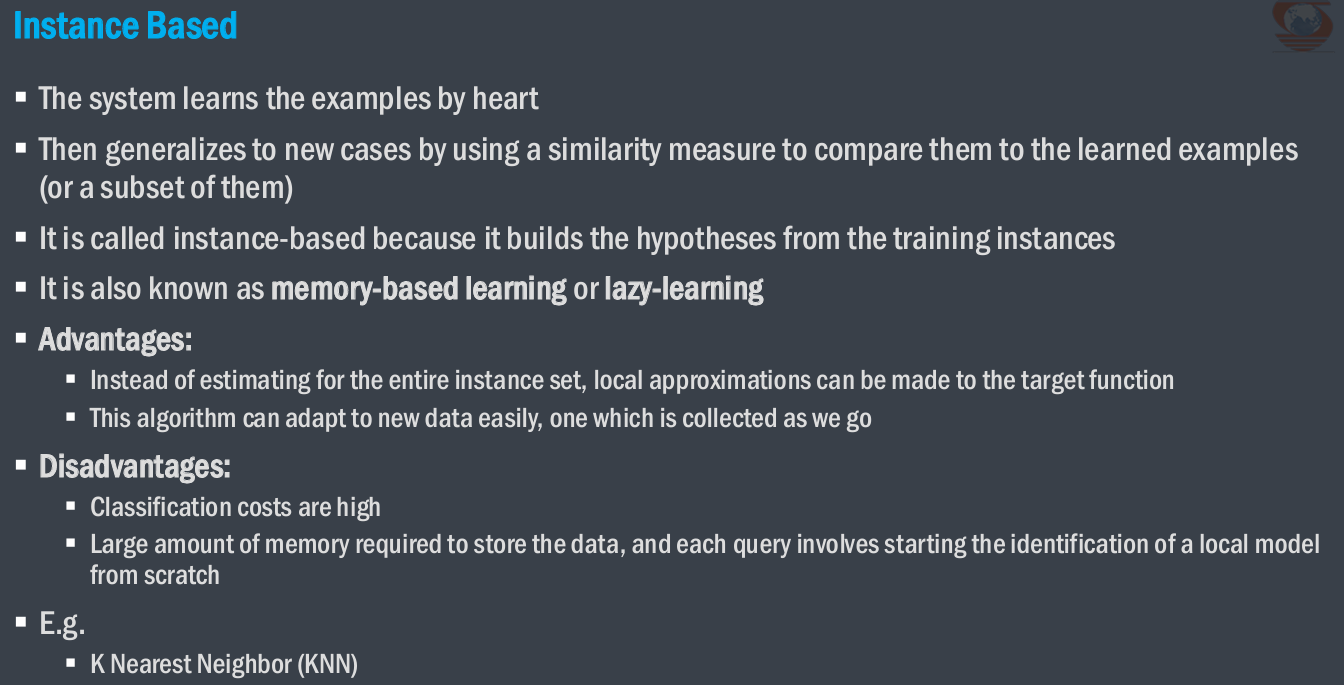
▪ One important parameter of online learning systems is how fast they should adapt to changing data: this is called the learning rate

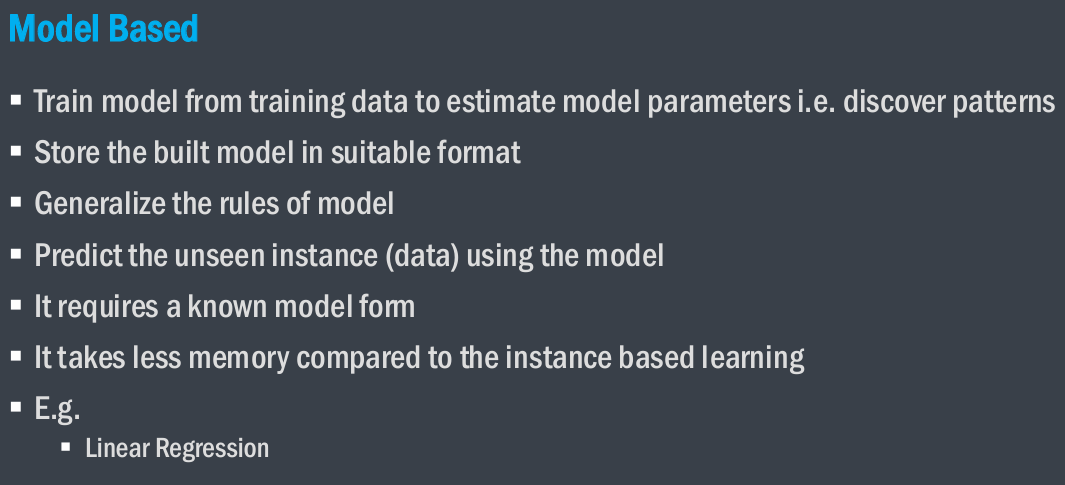
▪ If you set a high learning rate, then your system will rapidly adapt to new data, but it will also tend to quickly forget the old data

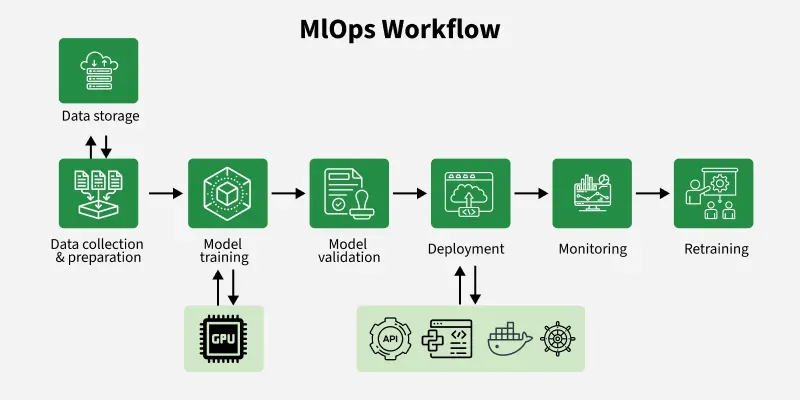
▪ you don’t want a spam filter to flag only the latest kinds of spam it was shown

▪ if you set a low learning rate, the system will have more inertia

▪ that is, it will learn more slowly, but it will also be less sensitive to noise in the new data or to sequences of nonrepresentative data points (outliers).







* **Collect and Clean the Data**: The first step is gathering data from different sources. This data often needs to be cleaned like fixing missing values or removing errors so that the model can learn from it properly.
* **Train and Test the Model**: Next the cleaned data is used to train the machine learning model. After training the model is tested to see how well it performs.
* **Package and Deploy the Model**: If the model gives good results it’s packaged (prepared for production) and deployed. This means the model is now ready to make real-world predictions.
* **Monitor the Model After Deployment**: Once the model is live it’s important to monitor it. Monitoring tools check how well it’s working like checking accuracy, speed or if it’s facing any problems.
* **Detect Performance Drops and Restart the Workflow**: If the model’s performance goes down. For example, if the incoming data is different from the data it was trained on known as data drift, then the system can automatically restart the workflow. This means retraining the model with fresh data.

**Benefits of MLOps**

Machine Learning Operations’ target is to bridge the gap between collaborating teams which can save time and help everyone align with the business objectives. Let’s discuss the benefits of MLOps:

* **Faster Deployment:**The process from building the model to using it in real life is done automatically, saving time and effort.
* **Better Teamwork**: It helps people from different teams like data science, engineering and operations work together more easily.
* **Easy Monitoring**: Once the model is live, its performance and errors are tracked automatically so there's less manual checking.
* **Grows with Your Needs**: It’s easy to use this process for small or large projects even when using the cloud or containers.
* **More Reliable Results**: By keeping track of versions and using automation the process gives stable and repeatable results every time.