Supervised Machine Learning:

● Supervised learning is a type of machine learning in which the algorithm is trained on the labelled dataset. It learns to map input features to targets based on labeled training data. In supervised learning, the algorithm is provided with input features and corresponding output labels, and it learns to generalize from this data to make predictions on new, unseen data.

● There are two main types of supervised learning:

● Regression: Regression is a type of supervised learning where the algorithm learns to predict continuous values based on input features. The output labels in regression are continuous values, such as stock prices, and housing prices. The different regression algorithms in machine learning are: Linear Regression, Polynomial Regression, Ridge Regression, Decision Tree Regression, Random Forest Regression, Support Vector Regression, etc

● Classification: Classification is a type of supervised learning where the algorithm learns to assign input data to a specific category or class based on input features. The output labels in classification are discrete values. Classification algorithms can be binary, where the output is one of two possible classes, or multiclass, where the output can be one of several classes. The different Classification algorithms in machine learning are: Logistic Regression, Naive Bayes, Decision Tree, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), etc

Unsupervised Machine Learning:

● Unsupervised learning is a type of machine learning where the algorithm learns to recognize patterns in data without being explicitly trained using labeled examples. The goal of unsupervised learning is to discover the underlying structure or distribution in the data.

● There are two main types of unsupervised learning:

● Clustering: Clustering algorithms group similar data points together based on their characteristics. The goal is to identify groups, or clusters, of data points that are similar to each other, while being distinct from other groups. Some popular clustering algorithms include K-means, Hierarchical clustering, and DBSCAN.

● Dimensionality reduction: Dimensionality reduction algorithms reduce the number of input variables in a dataset while preserving as much of the original information as possible. This is useful for reducing the complexity of a dataset and making it easier to visualize and analyze. Some popular dimensionality reduction algorithms include Principal Component Analysis (PCA), t-SNE, and Autoencoders.

Reinforcement Machine Learning

● Reinforcement learning is a type of machine learning where an agent learns to interact with an environment by performing actions and receiving rewards or penalties based on its actions. The goal of reinforcement learning is to learn a policy, which is a mapping from states to actions, that maximizes the expected cumulative reward over time.

● There are two main types of reinforcement learning:

● Model-based reinforcement learning: In model-based reinforcement learning, the agent learns a model of the environment, including the transition probabilities between states and the rewards associated with each state-action pair. The agent then uses this model to plan its actions in order to maximize its expected reward. Some popular model-based reinforcement learning algorithms include Value Iteration and Policy Iteration.

● Model-free reinforcement learning: In model-free reinforcement learning, the agent learns a policy directly from experience without explicitly building a model of the environment. The agent interacts with the environment and updates its policy based on the rewards it receives. Some popular model-free reinforcement learning algorithms include Q-Learning, SARSA, and Deep Reinforcement Learning.

Applications of Machine Learning:

● Automation: Machine learning, which works entirely autonomously in any field without the need for any human intervention. For example, robots perform the essential process steps in manufacturing plants.

● Finance Industry: Machine learning is growing in popularity in the finance industry. Banks are mainly using ML to find patterns inside the data but also to prevent fraud.

● Government organization: The government makes use of ML to manage public safety and utilities. Take the example of China with its massive face recognition. The government uses Artificial intelligence to prevent jaywalking.

● Healthcare industry: Healthcare was one of the first industries to use machine learning with image detection.

● Marketing: Broad use of AI is done in marketing thanks to abundant access to data. Before the age of mass data, researchers develop advanced mathematical tools like Bayesian analysis to estimate the value of a customer. With the boom of data, the marketing department relies on AI to optimize customer relationships and marketing campaigns.

● Retail industry: Machine learning is used in the retail industry to analyze customer behavior, predict demand, and manage inventory. It also helps retailers to personalize the shopping experience for each customer by recommending products based on their past purchases and preferences.

● Transportation: Machine learning is used in the transportation industry to optimize routes, reduce fuel consumption, and improve the overall efficiency of transportation systems. It also plays a role in autonomous vehicles, where ML algorithms are used to make decisions about navigation and safety.

Polynomial Curve Fitting – Conceptual Notes

Polynomial curve fitting is a fundamental technique in machine learning and statistics used to model the relationship between an independent variable xxx and a dependent variable ttt by fitting a polynomial function. The primary goal is to find a smooth curve that best approximates the underlying pattern in the given dataset. This method is particularly useful when the data exhibits a non-linear trend that cannot be captured by a straight line (linear regression).

In polynomial curve fitting, we assume a model of the form:

$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \cdots + w_M x^M$$

where M is the degree of the polynomial, and w={w_0,w_1,...,w_M} are the parameters (weights) to be estimated. The fitting process involves minimizing an error function that measures the difference between the predicted values and the actual observed data points. The most commonly used error function is the **sum-of-squares error**, defined as:

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \left(y(x_n, \mathbf{w}) - t_n\right)^2$$

This is a classic example of supervised learning, where the training set consists of N pairs of input-output values (x_n,t_n). The optimal weights w can be found by solving this minimization problem using linear algebra techniques or numerical optimization.

One of the key challenges in polynomial curve fitting is choosing the right model complexity, determined by the degree M of the polynomial. If M is too low, the model **underfits** the data, failing to capture important trends. If M is too high, especially when the number of data points is small, the model may **overfit**, capturing noise rather than the underlying function. Overfitting leads to poor generalization on new, unseen data.

To address overfitting, one common strategy is to use **regularization**, where a penalty term is added to the error function to discourage large weight values. This leads to the **regularized least squares** formulation, also known as **ridge regression**:

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \left(y(x_n, \mathbf{w}) - t_n\right)^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2$$

Here, λ is a regularization parameter that controls the trade-off between the fitting error and the model complexity. A higher value of λ suppresses large coefficients, thereby producing a smoother curve.

Polynomial curve fitting provides a clear illustration of the **bias-variance trade-off**. Low-degree polynomials have high bias and low variance, while high-degree polynomials have low bias and high variance. Finding the right degree (or using cross-validation to select it) helps achieve a good balance for predictive performance.

In practical machine learning, polynomial regression is not often used for high-dimensional datasets due to the explosion in the number of parameters as the degree increases. However, it serves as a vital conceptual foundation for understanding more complex non-linear models, kernel methods, and basis function expansions used in regression and classification tasks.