Question Bank

Units	Q.No	Question	Marks
INTRODUCTION TO MACHINE LEARNING Unit-1	1	Define Machine Learning.	1.5
	2	List two perspectives on Machine Learning.	1.5
	3	Explain the difference between supervised and unsupervised learning.	1.5
	4	What is reinforcement learning?	1.5
	5	Give two examples of Machine Learning applications.	1.5
	6	How does numerical data differ from graph data in terms of representation?	1.5
	7	Explain the different types of Machine Learning with examples.	5
	8	Discuss the perspectives and issues in Machine Learning.	5
	9	Describe how data can be represented for ML tasks.	5
	10	Discuss the key applications of Machine Learning across different domains.	5
SUPERVISED LEARNING	1	What is the K-Nearest Neighbors (KNN) algorithm?	1.5
	2	Define Decision Trees.	1.5
	3	Briefly explain univariate linear regression.	1.5
	4	What is the purpose of regularized regression?	1.5
	5	Explain the concept of Support Vector Machines (SVM).	1.5
	6	What are kernel methods in SVM?	1.5
	7	Explain Naïve Bayes classifier with an example.	5
	8	Differentiate between univariate and multivariate linear regression, with examples.	5
	9	Describe Support Vector Machines and how kernel methods enhance their performance.	5
	10	Compare and contrast KNN, Decision Trees, and Naïve Bayes.	5
UNSUPERVISED LEARNING	1	Define clustering in Machine Learning.	1.5
	2	What is the K-means clustering algorithm?	1.5
	3	Explain Kernel K-means.	1.5
	4	What is vector quantization?	1.5
	5	Describe the concept of a Self-Organizing Feature Map.	1.5
	6	Differentiate between clustering and classification.	1.5
	7	Describe the K-means clustering algorithm with an example.	5
	8	What is Kernel K-means? How does it differ from traditional K-means?	5
	9	Explain the Apriori algorithm and provide an example of its application.	5
	10	Compare and contrast clustering and classification techniques.	5
TECHNIQUES AND	1	What is scalable Machine Learning?	1.5
	2	Define Bayesian Learning.	1.5

APPLICATIONS	3	What is inference in Bayesian Learning?	1.5
	4	Mention two recent trends in Machine Learning techniques.	1.5
	5	Why is scalability important in ML?	1.5
	6	List two applications of Bayesian Learning.	1.5
	7	Explain scalable Machine Learning and its importance with examples.	5
	8	Discuss the fundamentals of Bayesian Learning and Inference.	5
	9	Explain classification methods with examples.	5
	10	Discuss the significance of scalability in Machine Learning and provide relevant examples.	5

Question Bank Solution

1. Define Machine Learning. (1.5 Marks)

Machine Learning (ML) is a branch of artificial intelligence that involves the development of algorithms and statistical models that enable computers to learn patterns from data and make predictions or decisions without being explicitly programmed.

2. List two perspectives on Machine Learning. (1.5 Marks)

1. **The engineering perspective**: ML is seen as a tool to build systems that can improve performance over time by learning from data.
2. **The mathematical perspective**: ML focuses on understanding and creating models that can identify patterns, make predictions, and classify data based on statistical methods.

3. Explain the difference between supervised and unsupervised learning. (1.5 Marks)

- **Supervised learning** involves training a model using labeled data, where the algorithm learns to predict outputs from known inputs. For example, predicting house prices based on features like size and location.
- **Unsupervised learning** uses unlabeled data, where the algorithm tries to find hidden patterns or groupings in the data. For example, clustering customers based on purchasing behavior.

4. What is reinforcement learning? (1.5 Marks)

1. Reinforcement learning is a type of machine learning where an agent learns to make decisions by performing actions in an environment to achieve a goal. It receives feedback in the form of rewards or penalties, which helps it learn the optimal strategy (or policy) to maximize the total reward over time.

5. Give two examples of Machine Learning applications. (1.5 Marks)

1. **Image Recognition**: ML is used to identify objects, people, or scenes in images (e.g., facial recognition systems).
2. **Fraud Detection**: ML algorithms analyze transaction patterns to detect and prevent fraudulent activities.

6. How does numerical data differ from graph data in terms of representation? (1.5 Marks)

- **Numerical data** is represented as a collection of numerical values, often organized in tabular form (e.g., spreadsheets), where each row represents a data point, and each column represents a feature.
- **Graph data** represents relationships between entities as nodes (vertices) connected by edges, often visualized as networks. Examples include social networks and transportation maps, where the focus is on connections and structure rather than individual data points.

6. How does numerical data differ from graph data in terms of representation?

- **Numerical Data**:
  - Represented in a structured, tabular form where each row corresponds to a data sample or instance, and each column represents a feature or attribute. Numerical data types include integers, floats, and sometimes categorical data encoded as numbers.
  - Examples include datasets with columns like age, income, temperature, or sales figures. Each data instance is typically independent of others, meaning relationships between rows are not inherently captured.
  - **Applications**: Predictive modeling, regression analysis, and classification tasks.
- **Graph Data**:
  - Represented as a set of nodes (vertices) and edges (connections between nodes). Nodes represent entities, and edges represent the relationships or interactions between them. This structure can represent complex interactions and is more suited for scenarios where connectivity or relationships matter.
  - Examples include social networks (where nodes are users and edges are friendships), molecular structures (nodes are atoms, edges are bonds), and transportation networks (nodes are locations, edges are routes).
  - **Applications**: Network analysis, recommendation systems, fraud detection, and biological research.

7. Explain the different types of Machine Learning with examples.

1. **Supervised Learning**:
   - **Definition**: In supervised learning, the algorithm is trained on a labeled dataset, where each training example consists of an input and a known output. The algorithm learns a mapping function to predict outputs for new inputs based on this training data.
   - **Examples**:

        - *Classification*: Email spam detection (classifying emails as "spam" or "not spam").
        - *Regression*: House price prediction (estimating the price of a house based on features like area, number of rooms, etc.).
2. **Unsupervised Learning**:
    - **Definition**: The algorithm is provided with data that does not have labeled outputs. It tries to identify underlying patterns or groupings in the data. This is useful for exploratory data analysis and understanding the structure of the data.
    - **Examples**:
        - *Clustering*: Customer segmentation (grouping customers into segments based on purchasing behavior).
        - *Dimensionality Reduction*: Principal Component Analysis (PCA) for reducing the number of features in a dataset while preserving essential information.
3. **Reinforcement Learning**:
    - **Definition**: Reinforcement learning involves an agent interacting with an environment, where it learns to perform actions to maximize a cumulative reward. The agent receives feedback in the form of rewards or penalties based on its actions and uses this to improve its strategy over time.
    - **Examples**:
        - *Game AI*: Training an AI to play chess or Go by learning strategies that maximize its chances of winning.
        - *Robotics*: Teaching a robot to navigate through a maze by rewarding it when it reaches the goal and penalizing it when it hits obstacles.

8. Discuss the perspectives and issues in Machine Learning.

- **Perspectives**:
    1. **Engineering Perspective**: ML is seen as a practical tool for building systems that can learn from data and automate tasks that would otherwise require manual effort. Engineers focus on creating scalable, efficient, and robust ML models that can handle real-world data.
    2. **Mathematical Perspective**: Emphasizes the theoretical foundation of ML, including understanding the underlying algorithms, probability, and optimization techniques. This perspective aims to improve model accuracy, generalizability, and robustness by refining mathematical models.
    3. **Data Science Perspective**: Focuses on extracting insights and knowledge from data. Here, machine learning is a tool for analyzing and interpreting data to make data-driven decisions.
- **Issues**:
    1. **Data Privacy and Security**: Machine learning models rely on vast amounts of data, often containing sensitive information. Ensuring privacy and security is crucial, as data breaches can have serious consequences.
    2. **Bias and Fairness**: Models can learn biases present in training data, leading to unfair outcomes, especially in areas like hiring, credit scoring, and criminal justice. Addressing this requires careful examination of data and model behavior.

3. **Model Interpretability**: Some machine learning models, especially deep learning, can be "black boxes," making it difficult to understand how they make decisions. This lack of transparency can be problematic, especially in critical applications like healthcare.
4. **Scalability and Efficiency**: Training complex models on large datasets requires significant computational resources. Researchers and engineers must find ways to make ML more efficient and scalable.

9. Describe how data can be represented for ML tasks.

1. **Tabular Data**:
   - Commonly used for structured datasets where each row is a data point, and each column is a feature. Examples include datasets for regression (predicting continuous values) and classification (categorizing items into classes).
   - **Example**: Housing dataset with columns for "Number of Rooms," "Location," and "Price."
2. **Text Data**:
   - Represented as sequences of words or characters, often processed through methods like tokenization and embedding. Text data is used in tasks like sentiment analysis, language translation, and information retrieval.
   - **Example**: Tweets can be analyzed to predict public sentiment on a topic.
3. **Image Data**:
   - Represented as pixel grids, where each pixel has values corresponding to color channels (e.g., RGB). Image data requires specialized processing techniques like convolutional neural networks (CNNs) for tasks like object detection and image classification.
   - **Example**: Handwritten digit recognition (MNIST dataset).
4. **Graph Data**:
   - Represents entities (nodes) and their connections (edges). Graph data is suitable for tasks where relationships between entities are essential, such as social network analysis and recommendation systems.
   - **Example**: Social network where users are nodes, and friendships are edges.
5. **Time-Series Data**:
   - Consists of sequences of data points collected over time intervals. Used for tasks where temporal dependencies are significant, like stock price prediction, weather forecasting, and sensor data analysis.
   - **Example**: Daily temperature readings, stock market prices over time.

10. Discuss the key applications of Machine Learning across different domains.

1. **Healthcare**:
   - **Applications**: Disease diagnosis, drug discovery, medical imaging, and personalized treatment. ML models can analyze medical images to detect diseases like cancer, predict patient outcomes, and identify potential drug compounds.
   - **Example**: IBM Watson for oncology, which helps doctors provide personalized cancer treatment recommendations.
2. **Finance**:

- **Applications**: Fraud detection, credit scoring, algorithmic trading, and risk management. Machine learning models can analyze transaction patterns to identify fraud, assess credit risk, and optimize trading strategies.
- **Example**: Real-time fraud detection systems that monitor and flag suspicious transactions.

3. **Retail**:
   - **Applications**: Product recommendation, customer segmentation, inventory management, and demand forecasting. Retailers use machine learning to understand customer behavior, suggest products, and optimize supply chains.
   - **Example**: Amazon's recommendation engine that suggests products based on user browsing and purchase history.
4. **Transportation**:
   - **Applications**: Self-driving cars, route optimization, traffic prediction, and logistics planning. ML models help improve route efficiency, reduce delivery times, and enhance safety in autonomous vehicles.
   - **Example**: Tesla's Autopilot system uses machine learning for lane-keeping, navigation, and object detection.
5. **Manufacturing**:
   - **Applications**: Predictive maintenance, quality control, and process optimization. ML models can predict equipment failures before they occur, ensuring smooth operations and reducing downtime.
   - **Example**: Predictive maintenance systems that analyze machine sensor data to predict when equipment needs servicing.
6. **Entertainment**:
   - **Applications**: Content recommendation, personalization, and automated content creation. Streaming services like Netflix use ML to recommend shows based on user viewing habits.
   - **Example**: Spotify's recommendation system that creates personalized playlists based on user listening patterns.

Units -2

1. What is the K-Nearest Neighbors (KNN) algorithm?

The **K-Nearest Neighbors (KNN)** algorithm is a simple, instance-based learning method used for classification and regression tasks. It works by identifying the ‘k’ closest data points (neighbors) in the training dataset to a new data point and predicting the output based on the majority class (in classification) or averaging the values (in regression) of those neighbors. The distance between data points is usually measured using metrics like Euclidean, Manhattan, or Minkowski distance. KNN is non-parametric, meaning it makes no assumptions about the underlying data distribution.

2. Define Decision Trees.

A **Decision Tree** is a machine learning model used for classification and regression tasks. It works by splitting the data into subsets based on the value of input features, creating a tree-like structure. Each internal node represents a decision based on a feature, each branch represents an outcome of that decision, and each leaf node represents a final prediction. Decision Trees are easy to interpret and understand, but they can be prone to overfitting if not properly managed.

3. Briefly explain univariate linear regression.

Univariate Linear Regression is a type of regression analysis where a single independent variable (feature) is used to predict a dependent variable (output). It assumes a linear relationship between the two variables, expressed by the equation:

$$y=mx+cy = mx + cy=mx+c$$

where:

- yyy is the predicted output,
- mmm is the slope of the line (coefficient),
- xxx is the input feature (independent variable), and
- ccc is the y-intercept.

The goal of univariate linear regression is to find the best-fitting line (using techniques like least squares) that minimizes the difference between predicted and actual values.

4. What is the purpose of regularized regression?

The purpose of **regularized regression** is to prevent overfitting by adding a penalty term to the regression model. Overfitting occurs when a model learns the noise in the training data rather than the actual underlying patterns, leading to poor performance on unseen data. Regularization discourages the model from fitting too closely to the training data by penalizing large coefficients, resulting in a simpler and more generalizable model.

- **Lasso (L1) Regularization**: Adds a penalty equal to the absolute value of the coefficients, which can lead to some coefficients becoming zero, effectively performing feature selection.
- **Ridge (L2) Regularization**: Adds a penalty equal to the square of the coefficients, which shrinks the coefficients towards zero without making them exactly zero.
- **Elastic Net**: Combines both L1 and L2 penalties.

5. Explain the concept of Support Vector Machines (SVM).

Support Vector Machines (SVM) are supervised learning models used for classification and regression tasks. The core idea of SVM is to find the hyperplane that best separates data points of different classes in a high-dimensional space. The hyperplane is chosen to maximize the margin between the nearest data points from each class (known as support vectors) and the hyperplane itself. By maximizing this margin, SVM aims to improve the generalization ability of the classifier.

- **Linear SVM**: Directly separates data that is linearly separable.
- **Non-Linear SVM**: Uses kernel functions to project data into a higher-dimensional space where it can be linearly separated.

6. What are kernel methods in SVM?

Kernel methods in SVM are techniques that enable the algorithm to handle non-linear relationships between input features by transforming data into a higher-dimensional space. Instead of explicitly calculating the coordinates of data points in this new space, a kernel function is used to calculate the inner product between pairs of data points in the original feature space. This allows SVM to find a hyperplane that can separate non-linear data.

Common kernel functions include:

- **Linear Kernel**: Suitable for linearly separable data.
- **Polynomial Kernel**: Captures non-linear relationships by raising data to a specified degree.
- **Radial Basis Function (RBF) Kernel**: Maps data to an infinite-dimensional space, useful for complex, non-linear boundaries.
- **Sigmoid Kernel**: Similar to a neural network activation function, used in some specific cases.

7. Explain Naïve Bayes classifier with an example. (5 Marks)

The **Naïve Bayes classifier** is a probabilistic machine learning model used for classification tasks. It is based on Bayes' Theorem and assumes that features are independent of each other given the class label, which is the "naïve" part of the assumption. Despite this strong assumption, Naïve Bayes often performs well in practice, especially for text classification tasks.

Bayes' Theorem:

P(A|B)=P(B|A)·P(A)P(B)P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}P(A|B)=P(B)P(B|A)·P(A)

- **P(A|B)**: Probability of event A occurring given that B is true (posterior probability)
- **P(B|A)**: Probability of event B occurring given that A is true (likelihood)
- **P(A)**: Probability of event A (prior probability)
- **P(B)**: Probability of event B (evidence)

Example: Suppose we want to classify emails as "spam" or "not spam" based on certain words (features). The Naïve Bayes classifier will calculate the probability of an email being spam given the presence of words like "discount" and "buy now." The classifier calculates the posterior probability for each class (spam, not spam) and assigns the email to the class with the highest probability.

For example, if:

- P(spam) = 0.4 (prior probability of spam)
- P(not spam) = 0.6 (prior probability of not spam)

- P("discount" | spam) = 0.7, P("buy now" | spam) = 0.8
- P("discount" | not spam) = 0.2, P("buy now" | not spam) = 0.1

The classifier will compute which class has a higher probability based on the presence of words.

8. Differentiate between univariate and multivariate linear regression, with examples. (5 Marks)

- **Univariate Linear Regression**:
  - Involves only one independent variable (predictor) to predict a dependent variable (response). It assumes a linear relationship between the single predictor and the output.
  - **Example**: Predicting house prices based solely on the size of the house. The equation would look like: y=mx+cy = mx + cy=mx+c where yyy is the house price, xxx is the size of the house, mmm is the slope (coefficient), and ccc is the intercept.
- **Multivariate Linear Regression**:
  - Involves multiple independent variables to predict a dependent variable. It can capture the influence of more than one feature, making it more powerful than univariate regression.
  - **Example**: Predicting house prices based on multiple factors like size, number of bedrooms, and location. The equation would look like: y=m1x1+m2x2+m3x3+cy = m_1x_1 + m_2x_2 + m_3x_3 + cy=m1x1+m2x2+m3x3+c where x1,x2,x3x_1, x_2, x_3x1,x2,x3 are features (size, bedrooms, location), and m1,m2,m3m_1, m_2, m_3m1,m2,m3 are their respective coefficients.

9. Describe Support Vector Machines and how kernel methods enhance their performance. (5 Marks)

Support Vector Machines (SVM) are supervised learning algorithms used for classification and regression tasks. SVM aims to find the optimal hyperplane that best separates data points of different classes in a high-dimensional space. The goal is to maximize the margin, which is the distance between the hyperplane and the closest data points from each class, known as support vectors.

- **Linear SVM**: Directly separates linearly separable data by finding a straight line (in 2D) or hyperplane (in higher dimensions) that divides the classes.
- **Non-Linear SVM**: When data is not linearly separable, SVM uses **kernel methods** to transform the data into a higher-dimensional space where a linear separation is possible.

Kernel Methods:

- **Concept**: Kernels transform the original feature space into a higher-dimensional space without explicitly computing the transformation. This allows SVM to handle non-linear relationships effectively.
- **Types of Kernels**:
  1. **Linear Kernel**: Useful for linearly separable data.

2. **Polynomial Kernel**: Allows the algorithm to learn polynomial relationships between features.
3. **Radial Basis Function (RBF) Kernel**: Maps data to an infinite-dimensional space, making it effective for complex, non-linear decision boundaries.
4. **Sigmoid Kernel**: Similar to a neural network activation function, used in certain scenarios.

By using kernel methods, SVM can find the optimal hyperplane in a higher-dimensional space, making it suitable for complex classification tasks.

10. Compare and contrast KNN, Decision Trees, and Naïve Bayes. (5 Marks)

Feature	K-Nearest Neighbors (KNN)	Decision Trees	Naïve Bayes
Type	Instance-based, non-parametric	Tree-based, non-parametric	Probabilistic, parametric
Learning Approach	Lazy learning (no training phase, predictions are made at runtime)	Eager learning (builds a model during the training phase)	Eager learning (builds a model during the training phase)
Assumptions	No assumptions about data distribution	No assumptions about feature relationships, but may overfit	Assumes features are conditionally independent
Pros	Simple to implement, good for low-dimensional data	Easy to interpret, can handle both categorical and numerical data	Fast, efficient, works well with large datasets, interpretable
Cons	Slow with large datasets, sensitive to irrelevant features	Prone to overfitting, especially with noisy data	Assumption of independence may not hold in all cases
Use Cases	Recommendation systems, classification	Decision-making tasks, classification, and regression	Spam detection, text classification, sentiment analysis

- **KNN**: Works by finding the ‘k’ closest data points to make predictions. It is simple but computationally expensive when making predictions on large datasets.
- **Decision Trees**: Create a model that makes decisions by splitting data based on features. They are interpretable and can handle various types of data but are prone to overfitting if not controlled.
- **Naïve Bayes**: Uses probability and the assumption of independence to make predictions. It is fast, efficient, and works well for text classification but relies on the assumption that all features are independent, which may not always be true.

Units-3

1. Define clustering in Machine Learning. (1.5 Marks)

Clustering is an unsupervised learning technique in Machine Learning used to group similar data points into clusters based on their features. Unlike classification, clustering does not require labeled data; instead, it identifies inherent patterns in the

dataset. The objective is to ensure that data points within the same cluster are more similar to each other than to those in other clusters.

2. What is the K-means clustering algorithm? (1.5 Marks)

K-means clustering is an unsupervised algorithm that partitions data into 'k' clusters, where 'k' is a predefined number. The algorithm works by initializing 'k' centroids, then assigning each data point to the nearest centroid, forming clusters. It iteratively updates the centroids based on the mean of all data points in each cluster and reassigns data points until convergence is achieved (when data point assignments no longer change).

3. Explain Kernel K-means. (1.5 Marks)

Kernel K-means is an extension of the traditional K-means algorithm that can handle non-linearly separable data. It uses kernel functions to transform the original feature space into a higher-dimensional space where clusters are more easily separable. By applying the kernel trick, Kernel K-means can detect complex, non-linear patterns in the data, making it more flexible than standard K-means.

4. What is vector quantization? (1.5 Marks)

Vector quantization (VQ) is a process of partitioning a large set of data points into groups (clusters), where each group is represented by a prototype or centroid (code vector). This technique is used in signal compression, data compression, and pattern recognition, where it reduces the data's dimensionality by mapping similar data points to the same representative code vector, minimizing distortion.

5. Describe the concept of a Self-Organizing Feature Map. (1.5 Marks)

A **Self-Organizing Feature Map (SOFM)**, or **Self-Organizing Map (SOM)**, is a type of artificial neural network that uses unsupervised learning to produce a low-dimensional (usually 2D) representation of data while preserving its topological structure. It maps high-dimensional input data onto a grid of neurons, where similar data points are mapped closer together, creating a feature map that highlights patterns and relationships within the data.

6. Differentiate between clustering and classification. (1.5 Marks)

- **Clustering**:
  - Unsupervised learning technique.
  - Groups data points into clusters based on similarities without prior labels.
  - Example: Grouping customers based on purchasing behavior.
- **Classification**:
  - Supervised learning technique.
  - Assigns data points to predefined categories based on labeled training data.
  - Example: Email spam detection (classifying emails as "spam" or "not spam").

7. Describe the K-means clustering algorithm with an example.

K-means clustering is a method that partitions a dataset into 'k' clusters by minimizing the variance within each cluster. The algorithm follows these steps:

1. **Initialization**: Randomly select 'k' initial centroids.
2. **Assignment**: Assign each data point to the nearest centroid, forming 'k' clusters.
3. **Update**: Recalculate the centroids of the clusters by taking the mean of all data points within each cluster.
4. **Repeat**: Repeat the assignment and update steps until the centroids do not change significantly or a maximum number of iterations is reached.

Example: Suppose a retailer wants to segment their customers based on annual spending and frequency of visits. Using K-means with k=3k = 3k=3, the algorithm might group the customers into three clusters:

- **Cluster 1**: Frequent shoppers with low spending.
- **Cluster 2**: Occasional shoppers with moderate spending.
- **Cluster 3**: Regular shoppers with high spending.

The retailer can use these clusters to develop targeted marketing strategies for each group.

8. What is Kernel K-means? How does it differ from traditional K-means?

Kernel K-means is a variation of the K-means algorithm that allows clustering of data that is not linearly separable. It uses a **kernel function** to implicitly map the data into a higher-dimensional space, making it easier to identify clusters that have complex shapes. The standard K-means algorithm only finds spherical clusters in the original feature space, but Kernel K-means can find clusters that are non-linear.

Difference:

- **Traditional K-means**:
  - Operates in the original feature space.
  - Assumes that clusters are spherical and linearly separable.
  - Cannot handle complex, non-linear relationships effectively.
- **Kernel K-means**:
  - Uses a kernel function (e.g., Gaussian, Polynomial) to project data into a higher-dimensional space.
  - Can detect clusters with non-linear boundaries.
  - More flexible but computationally more intensive than traditional K-means.

9. Explain the Apriori algorithm and provide an example of its application.

The **Apriori algorithm** is an unsupervised algorithm used for mining frequent itemsets and generating association rules in a dataset. It is commonly applied in market basket analysis to identify products that are frequently purchased together. The algorithm works by:

1. Generating frequent itemsets by iteratively finding combinations of items that meet a minimum support threshold.
2. Using these frequent itemsets to derive association rules that satisfy a minimum confidence level.

Example: A supermarket uses the Apriori algorithm to analyze customer transactions. If it finds that "bread" and "butter" are often bought together (frequent itemset) with a high confidence level, it can generate a rule like: "If a customer buys bread, they are likely to buy butter." The supermarket can use this information to place bread and butter close to each other on the shelves or run promotions to boost sales.

10. Compare and contrast clustering and classification techniques.

Aspect	Clustering	Classification
Type	Unsupervised learning	Supervised learning
Objective	Group data points into clusters based on similarity	Assign data points to predefined classes
Labeling	Does not require labeled data	Requires labeled training data
Output	Clusters with similar data points	Predicted class labels
Examples	Customer segmentation, document clustering	Email spam detection, disease diagnosis
Use Cases	Exploratory data analysis, anomaly detection	Predictive tasks, decision-making processes
Algorithms	K-means, Hierarchical Clustering, DBSCAN	Decision Trees, SVM, Naïve Bayes, Neural Networks
Strengths	Can find hidden patterns and relationships without labels	Makes accurate predictions using historical labeled data
Weaknesses	Requires defining the number of clusters, can be sensitive to noise	Requires labeled data, may not generalize well to new unseen data

- **Clustering** is primarily used for exploratory analysis, pattern discovery, and finding groups within data without prior knowledge of categories. It is useful when labels are not available.
- **Classification** relies on labeled data to train a model and can predict the category of new data points. It is suitable for applications where categories are known and predefined.

Units-4

1. What is scalable Machine Learning? (1.5 marks)

Scalable Machine Learning refers to the ability of machine learning algorithms to handle increasing volumes of data and computational tasks efficiently. It ensures that models can be trained and make predictions without a significant increase in computational cost as the data size grows.

2. Define Bayesian Learning. (1.5 marks)

Bayesian Learning is a probabilistic approach to machine learning where the goal is to update the probability of a hypothesis as more evidence or data becomes available.

It uses Bayes' Theorem to combine prior knowledge with new information to make predictions.

3. What is inference in Bayesian Learning? (1.5 marks)

Inference in Bayesian Learning is the process of updating the probability distribution of a model's parameters based on observed data. It involves calculating the posterior probability, which combines prior beliefs and the likelihood of observed data.

4. Mention two recent trends in Machine Learning techniques. (1.5 marks)

1. **Deep Reinforcement Learning**: Combining deep learning with reinforcement learning to enable more complex decision-making.
2. **Federated Learning**: Training models across multiple decentralized devices while preserving data privacy.

5. Why is scalability important in ML? (1.5 marks)

Scalability is essential in ML because it allows models to handle large and diverse datasets efficiently. As the volume of data grows, scalable systems can process and analyze the data without a drastic increase in computational resources, making the solution more feasible for real-world applications.

6. List two applications of Bayesian Learning. (1.5 marks)

1. **Spam Filtering**: Using Bayesian classifiers to predict whether an email is spam or not.
2. **Medical Diagnosis**: Estimating the probability of diseases based on symptoms and prior patient data.

7. Explain Scalable Machine Learning and Its Importance with Examples

Scalable Machine Learning refers to designing and implementing machine learning algorithms that can handle increasing amounts of data and computational workloads efficiently. As the size of datasets and the number of features grow, scalable machine learning ensures that models can still process and learn from data without significant delays or a drastic increase in computational resources. Scalability involves both the training and inference phases and often requires optimizing algorithms and hardware infrastructure.

Importance:

1. **Data Growth**: With the rise of big data, the volume of data generated by various sources like social media, IoT devices, and e-commerce platforms is growing exponentially. Scalable ML allows businesses to leverage this data for insights and decisions without hitting performance bottlenecks.
2. **Real-Time Processing**: Applications such as fraud detection, recommendation systems, and autonomous driving require real-time data processing. Scalable models can analyze data and make predictions quickly, even under heavy data loads.

3. **Cost Efficiency**: Scalable ML reduces the need for excessive computing resources, making it more cost-effective. Instead of deploying multiple servers, businesses can use optimized models that scale well.

Examples:

- **Google Search**: Google's search algorithms analyze millions of web pages to return relevant results within milliseconds. Scalable machine learning is at the core of this system, enabling it to handle billions of searches every day.
- **Netflix Recommendations**: Netflix uses scalable ML algorithms to analyze users' viewing habits and provide personalized recommendations. The system processes massive amounts of data daily to deliver content suggestions in real time.
- **Autonomous Vehicles**: Self-driving cars rely on scalable ML to process data from multiple sensors (cameras, lidar, radar) in real time, enabling them to make split-second decisions while driving.

8. Discuss the Fundamentals of Bayesian Learning and Inference

Bayesian Learning is a probabilistic approach to learning where we update our beliefs about a model or hypothesis as new data becomes available. The foundation of Bayesian learning is **Bayes' Theorem**, which states:

P(H|D)=P(D|H)·P(H)P(D)P(H|D) = \frac{P(D|H) \cdot P(H)}{P(D)}P(H|D)=P(D)P(D|H)·P(H)

where:

- P(H|D)P(H|D)P(H|D) is the posterior probability of the hypothesis HHH given data DDD,
- P(D|H)P(D|H)P(D|H) is the likelihood of observing data DDD given HHH,
- P(H)P(H)P(H) is the prior probability of HHH, and
- P(D)P(D)P(D) is the marginal likelihood of observing DDD.

Key Concepts:

1. **Prior Probability (P(H)P(H)P(H))**: Represents our belief about the hypothesis before seeing the data.
2. **Likelihood (P(D|H)P(D|H)P(D|H))**: The probability of the observed data under the hypothesis.
3. **Posterior Probability (P(H|D)P(H|D)P(H|D))**: Updated belief about the hypothesis after considering the new data.
4. **Inference**: The process of calculating the posterior distribution to make predictions or decisions. Bayesian inference provides a framework for updating beliefs and handling uncertainty, especially when dealing with limited data.

Advantages:

- **Handling Uncertainty**: Bayesian learning is particularly effective in scenarios where uncertainty plays a crucial role, allowing the integration of prior knowledge with new evidence.
- **Interpretability**: It provides a clear probabilistic interpretation of results, making it easier to understand how decisions are made.

Examples:

- **Spam Filtering**: Bayesian classifiers can predict the probability that an email is spam by combining prior information (previous emails labeled as spam) with new data (words in the current email).
- **Medical Diagnosis**: Bayesian networks can estimate the probability of a disease given symptoms and patient history, making it useful in healthcare for probabilistic diagnosis.

9. Explain Classification Methods with Examples

Classification is a supervised learning technique where the goal is to categorize data points into predefined classes based on input features. Various classification algorithms exist, each with its strengths and applications:

1. **Decision Trees**:
   - Decision trees classify data by creating a tree-like structure of decisions. Each node represents a feature, and branches represent possible values. The process continues until the model classifies data into specific categories.
   - **Example**: Classifying whether a person will buy a product based on features like age, income, and browsing behavior.
2. **Support Vector Machines (SVM)**:
   - SVMs aim to find the optimal hyperplane that best separates data points of different classes. The goal is to maximize the margin between the classes, making it robust to outliers.
   - **Example**: Image classification tasks, like distinguishing between pictures of cats and dogs.
3. **K-Nearest Neighbors (KNN)**:
   - KNN classifies data based on the majority class of its nearest neighbors. It does not have a training phase but stores all the data and finds the closest kkk points to make a decision.
   - **Example**: Recommending products based on similar user behavior.
4. **Neural Networks**:
   - Neural networks learn complex patterns by adjusting weights during training. They can handle non-linear data relationships, making them suitable for tasks that require high accuracy.
   - **Example**: Recognizing handwritten digits in postal addresses.

Real-World Applications:

- **Email Spam Detection**: Classifiers can categorize emails as "spam" or "not spam" based on words, phrases, and other features.
- **Fraud Detection**: Classifying transactions as "fraudulent" or "legitimate" based on spending patterns.

10. Discuss the Significance of Scalability in Machine Learning and Provide Relevant Examples

Significance of Scalability:

1. **Data Explosion**: The amount of data generated daily has grown tremendously, thanks to social media, IoT, and other digital platforms. Scalable ML algorithms can process this data efficiently without compromising performance.
2. **Real-Time Analysis**: In areas like financial trading or autonomous systems, decisions must be made in real time. Scalable models can handle high throughput, providing fast and accurate results.
3. **Cost Efficiency**: Scalable systems reduce the need for extensive computational resources, making ML solutions more affordable and feasible for businesses of all sizes.
4. **Adaptability**: Scalable algorithms can adapt to changes in data patterns without requiring a complete overhaul, which is crucial for dynamic environments.

Examples:

- **Amazon's Product Recommendations**