Code No: 156BN **R18**

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
B. Tech III Year II Semester Examinations, February - 2023
MACHINE LEARNING
(Computer Science and Engineering)

Time: 3 Hours **Max. Marks: 75**

Note: i) The Question paper consists of Part A, and Part B.
ii) Part A is compulsory, which carries 25 marks. In Part A, Answer all questions.
iii) In Part B, Answer any one question from each unit. Each question carries 10 marks and may have a, and b as sub-questions.

PART – A **(25 Marks)**

1. a) Define Machine learning. [2]
b) Write about inductive bias. [3]
c) Define hypothesis. [2]
d) Write about sampling theory. [3]
e) Define Eager learning. [2]
f) What is lazy Learning? [3]
g) Define the term Genetic. [2]
h) Discuss Dynamic Programming. [3]
i) Define explanation-based learning. [2]
j) Write about control knowledge. [3]

PART – B **(50 Marks)**

2. Which disciplines have their influence on machine learning? Explain with examples. [10]

OR

3. a) Explain the two uses of features in machine learning.
b) Discuss about decision tree representation, in detail. [5+5]

4. Explain in detail about Kernel Perceptron. [10]

OR

5. Discuss in detail about representation of Neural Networks. [10]

6. Describe briefly about k-nearest neighbor algorithm. [10]

OR

7.a) Discuss about Bayesian belief networks.
b) Explain about Bayes theorem. [5+5]

8. Discuss Briefly about Genetic algorithms in detail. [10]

OR

9. a) Discuss about Q–learning, in detail.
   b) Explain temporal difference learning in detail. [5+5]

10. Explain about PROLOG-EBG, in detail. [10]

OR

11. a) Discuss about augment search operators.
    b) Explain about search control knowledge in detail. [5+5]

---ooOoo---

Answer Key

PART - A

1. a) Define Machine Learning.

Machine learning is a field of artificial intelligence that involves the development of algorithms and statistical models that enable computers to learn and make predictions or decisions without being explicitly programmed, based on data.

1. b) Write about Inductive Bias.

Inductive bias refers to the set of assumptions a machine learning algorithm makes to generalize beyond the specific data it is trained on. It's essential for dealing with the problem of underdetermined learning, where the learning algorithm must make predictions for unseen instances. Examples include preferring simpler models or specific model structures.

1. c) Define Hypothesis.

In the context of machine learning, a hypothesis is a specific model or solution proposed by a learning algorithm to solve a given problem. It represents a potential generalization derived from the training data and is tested against unseen data.

1.d) Write about Sampling Theory.

Sampling theory is a statistical theory concerning the collection of a subset of individuals from within a statistical population to estimate characteristics of the whole population. It underpins the methodology of survey samples, random sampling, and estimation techniques, ensuring that samples represent the population accurately.

1. e) Define Eager Learning.

Eager learning is a machine learning approach where the learning process is completed before receiving new data to make predictions. It involves constructing a complete model from the training data, which is then used for future predictions, contrasting with lazy learning where generalization is deferred until data is queried.

f) What is Lazy Learning?

Lazy Learning is a machine learning approach where the generalization of the training data is delayed until a query is made. This means the learning system does not build a model until it's required to make a decision, often leading to less time spent in training but more time in predicting.

g) Define the term Genetic

The term "Genetic" in the context of computing and artificial intelligence usually refers to genetic algorithms. These are optimization techniques inspired by the process of natural selection, using mechanisms such as selection, crossover, and mutation to evolve solutions to problems.

h) Discuss Dynamic Programming

Dynamic Programming is a method for solving complex problems by breaking them down into simpler subproblems. It is applicable when the problem can be divided into overlapping sub-problems that can be solved independently. The key technique is to store the solutions of these subproblems to avoid redundant calculations.

i) Define Explanation-Based Learning

Explanation-Based Learning is a type of machine learning that focuses on understanding and interpreting the logic behind the data. It involves forming a hypothesis or explanation for observed data and using this understanding to improve future predictions or decisions.

j) Write about Control Knowledge

Control knowledge refers to the knowledge used in artificial intelligence systems to guide the decision-making process. It helps in selecting the most appropriate strategies or rules in a given situation, thereby improving the efficiency and effectiveness of problem-solving or learning processes.

PART – B

2. Which disciplines have their influence on machine learning? Explain with examples.

1. Computer Science: Provides the algorithms, theory, and technological foundations for creating machine learning models. Example: Development of neural networks.

2. Statistics: Offers methodologies for making inferences and predictions, handling uncertainty and variance. Example: Regression analysis.
3. Mathematics: Fundamental for algorithms, especially in areas like linear algebra, calculus, and optimization. Example: Support Vector Machines (SVM) rely on complex mathematical formulations.
4. Data Science: Focuses on practical applications of machine learning, including data preparation, exploration, and visualization. Example: Predictive analytics in business.
5. Psychology: Influences understanding of human learning processes, aiding in the development of models that mimic cognitive functions. Example: Reinforcement learning inspired by behavioral psychology.
6. Neuroscience: Provides insights into brain function, inspiring designs of neural network architectures. Example: Deep learning networks inspired by the structure of the brain.
7. Physics: Methods from physics, especially in dealing with large datasets and complex systems. Example: Using entropy and thermodynamics concepts in clustering algorithms.
8. Linguistics: Especially in natural language processing (NLP), for understanding and generating human language. Example: Development of language models like GPT (Generative Pre-trained Transformer).
9. Philosophy: Particularly in areas of ethics and logic, shaping the way algorithms are developed and applied. Example: Ethical AI and machine ethics.
10. Biology: Particularly in evolutionary algorithms and understanding natural processes. Example: Genetic algorithms inspired by the process of natural evolution.

3. a) Explain the two uses of features in machine learning.

1. Representation: Features act as the primary form of data representation for machine learning models. They capture the characteristics of the data that are relevant to the problem. For instance, in image recognition, pixel values serve as features.
2. Prediction/Classification: Features are used by machine learning models to make predictions or classifications. The model learns how different features correlate with different outcomes. For example, in email spam detection, features like specific words or the sender's address are used to classify emails as spam or not.

3. b) Discuss about decision tree representation, in detail.

1. Structure: A decision tree is a flowchart-like structure with internal nodes representing tests on attributes, branches representing outcomes of the test, and leaf nodes representing class labels or decision outcomes.

2. Top-Down Induction: Trees are typically built top-down, starting with a root node that represents the entire dataset, and recursively splitting it into subsets.
3. Splitting Criteria: Uses measures like Gini impurity, entropy, or information gain to decide how to split the data at each node.
4. Handling Both Types of Data: Can handle categorical and numerical data, though the method of splitting may vary.
5. Pruning: To avoid overfitting, trees are pruned either by limiting the depth as they grow or by removing branches post-growth.
6. Visual Interpretability: One of the most interpretable models due to their graphical nature, allowing easy understanding of how decisions are made.
7. Binary and Multi-way Trees: Commonly binary (each node has two children), but can be expanded to multi-way trees based on the nature of the data.
8. Handling Missing Values: Capable of handling missing values through strategies like surrogate splits or distributing samples proportionally across branches.
9. Regression Trees: Besides classification, decision trees can be used for regression tasks, predicting continuous values.
10. Ensemble Methods: Often used as a base learner in ensemble methods like Random Forests and Gradient Boosting Machines, enhancing predictive performance and robustness.

4. Explain in detail about Kernel Perceptron.

1. Basic Concept: The Kernel Perceptron is an extension of the traditional perceptron algorithm, utilizing kernel methods to enable it to classify non-linearly separable data.
2. Perceptron Review: The traditional perceptron is a linear classifier that uses a linear function to make predictions based on input features.
3. Limitation of Linear Perceptron: It struggles with non-linearly separable data, which is where Kernel Perceptron comes into play.
4. Kernel Trick: The kernel trick involves mapping input data into a higher-dimensional space where it is linearly separable.
5. Common Kernels: Popular kernel functions include Polynomial, Radial Basis Function (RBF), and Sigmoid kernels.
6. Learning Algorithm: The learning process is similar to the standard perceptron, but it operates in the feature space defined by the kernel.
7.Dual Representation: The Kernel Perceptron often uses a dual representation where the decision function is expressed in terms of a subset of training samples (support vectors).
8. Computational Efficiency: While more computationally intensive than the linear perceptron, it's efficient in handling complex, non-linear relationships.

9. Application Areas: Useful in various fields like image and speech recognition, where data is often not linearly separable.

10. Limitations and Challenges: Choosing the right kernel and parameters can be challenging, and the algorithm can be computationally expensive for large datasets.

5. Discuss in detail about representation of Neural Networks.

1. Basic Structure: Neural Networks consist of layers of interconnected nodes or neurons, including input, hidden, and output layers.
2. Neurons: Each neuron in a network processes input signals and produces an output signal using an activation function.
3. Weights and Biases: Connections between neurons have associated weights and biases, which are adjusted during learning to optimize network performance.
4. Activation Functions: Functions like Sigmoid, ReLU, and Tanh determine the output of a neuron based on the input signals.
5. Layers: The input layer receives the initial data, hidden layers perform computations, and the output layer produces the final prediction.
6. Feedforward Networks: In these networks, data moves forward from input to output layers, typically used in straightforward prediction tasks.
7. Backpropagation: A key learning process where the network adjusts its weights and biases based on the error in its predictions.
8. Deep Learning: Involves networks with many hidden layers (deep neural networks), capable of learning complex patterns.
9. Convolutional Neural Networks (CNNs): Specialized for processing data with a grid-like topology, such as images.
10. Recurrent Neural Networks (RNNs): Designed to handle sequential data, with connections between nodes forming a directed graph along a temporal sequence.

6. Describe briefly about k-nearest neighbor algorithm.

1. Basic Concept: The k-nearest neighbor (k-NN) algorithm is a simple, non-parametric method used for classification and regression. It predicts the label of a data point by looking at the 'k' closest labeled data points.
2. Distance Measure: It uses distance metrics like Euclidean, Manhattan, or Hamming distance to find the closest neighbors.
3. Choosing 'k' Value: The choice of 'k' is crucial; too small a value makes the algorithm sensitive to noise, while too large a value might include points from other classes.

4. Majority Voting: In classification, the label of a new point is determined by the majority vote of its neighbors, with the most common class among its k nearest neighbors being assigned.

5. Weighted Neighbors: Sometimes, the contribution of neighbors is weighted by the inverse of their distance, giving closer neighbors more influence.
6. No Training Phase: Unlike other algorithms, k-NN does not have a training phase. It simply stores the dataset and performs actions on the dataset at the time of classification.
7. Lazy Learning Algorithm: It is a type of lazy learning where the function is only approximated locally and all computation is deferred until classification.
8. Handling of Non-linear Data: It can capture non-linear relationships between data points without any assumptions about the underlying data distribution.
9. Scalability and Efficiency: The efficiency decreases as the dataset grows, making it less effective for large datasets.
10. Applications: Widely used in recommendation systems, image recognition, and credit scoring, where data relationships are more important than underlying patterns.

7.a) Discuss about Bayesian belief networks.

1. Definition: Bayesian Belief Networks, also known as Bayesian Networks or Bayes Nets, are graphical models that represent probabilistic relationships among variables using Directed Acyclic Graphs (DAGs).
2. Nodes and Edges: Each node represents a random variable, while edges represent probabilistic dependencies between these variables.
3. Conditional Probability: The strength of the relationships between variables is quantified using conditional probability distributions.
4. Inference: These networks are used for reasoning and making inferences about unknown variables given known ones.
5. Dealing with Uncertainty: Particularly effective in handling uncertainty and incomplete data.
6. Learning: Can be constructed both from prior knowledge and by learning from data.
7. Dynamic Bayesian Networks: Extension of Bayesian networks to model temporal processes.
8. Applications: Widely used in various fields like medical diagnosis, risk assessment, and machine learning.
9. Complexity: The major challenge is the computational complexity involved in learning and inference.
10. Flexibility: They can easily be updated with new evidence, making them adaptable for dynamic environments.

7.b) Explain about Bayes theorem.

1. Definition: Bayes' Theorem is a fundamental theorem in probability theory that describes how to update the probabilities of hypotheses when given evidence.
2. Formula: It is mathematically expressed as P(A|B) = [P(B|A) * P(A)] / P(B), where P(A|B) is the probability of hypothesis A given the data B.
3. Prior Probability: P(A) is the prior probability of the hypothesis before seeing the evidence.
4. Likelihood: P(B|A) is the likelihood, which is the probability of observing the evidence given that the hypothesis is true.
5. Marginal Probability: P(B) is the marginal probability of the evidence under all possible hypotheses.
6. Posterior Probability: The result, P(A|B), is known as the posterior probability, representing the probability of the hypothesis after taking the evidence into account.
7. Updating Beliefs: It provides a way to update our beliefs in light of new, relevant information.
8. Foundation in Bayesian Inference: It is the core principle behind Bayesian inference, a method of statistical inference.
9. Versatility: Widely used across various fields including statistics, machine learning, and data science.
10. Decision Making: Essential in decision-making processes, especially in uncertain conditions and complex systems.

8. Discuss Briefly about Genetic algorithms in detail.

1. Definition: Genetic Algorithms (GAs) are search heuristics inspired by Charles Darwin's theory of natural evolution, designed to find optimal solutions to complex problems.
2. Components: GAs consist of a population of candidate solutions, represented as chromosomes.
3. Initialization: The process begins with a randomly generated population.
4. Fitness Function: Each solution is evaluated using a fitness function to determine how close it is to the desired solution.
5. Selection: The fittest solutions are selected for reproduction. This mimics natural selection where the strongest survive.
6. Crossover: Selected solutions undergo crossover (mating) to produce offspring, sharing features of parent solutions.
7. Mutation: To maintain genetic diversity and avoid premature convergence, offspring may undergo random mutations.

8. Generation Loop: The process of selection, crossover, and mutation is repeated over several generations, gradually evolving better solutions.
9. Convergence: The algorithm terminates when either a maximum number of generations is reached or an acceptable level of fitness is achieved.
10. Applications: GAs are widely used in optimization problems, scheduling, machine learning, computational biology, and artificial intelligence.

9.a) Discuss about Q–learning, in detail.

1. Definition: Q-learning is a model-free reinforcement learning algorithm used to learn the value of an action in a particular state.
2. Learning Policy: It does not require a model of the environment and learns by trial and error.
3. Q-Value: Represents the expected future rewards for an action taken in a given state.
4. Update Rule: Q-values are updated using the Bellman Equation, a recursive formula incorporating immediate rewards and future rewards.
5. Exploration vs. Exploitation: Involves balancing exploration (trying new actions) and exploitation (using known good actions).
6. Convergence: Under certain conditions, Q-learning is proven to converge to the optimal action-value function.
7. Learning Rate: Involves a learning rate parameter that affects the speed and stability of learning.
8. Discount Factor: The algorithm includes a discount factor which balances immediate and future rewards.
9. Off-policy: Q-learning is off-policy, meaning the learned policy is independent of the agent's actions.
10. Applications: Widely used in robotics, gaming, and automated control systems.

9. b) Explain temporal difference learning in detail.

1. Definition: Temporal Difference (TD) Learning is a reinforcement learning approach that learns directly from raw experience without a model of the environment.
2. TD Error: The key component is the TD error, representing the difference between predicted and actual rewards over time.
3. Learning from Difference: Learns by adjusting predictions to match the more accurate later estimate.
4. Bootstrapping: Uses existing estimates to update new estimates, a method known as bootstrapping.
5. Update Rule: Involves a simple update rule which adjusts values towards estimated future rewards.

6. Balancing Exploration and Exploitation: Like other reinforcement learning methods, it deals with the exploration-exploitation dilemma.
7. Advantages: Can learn before knowing the final outcome and can learn online after every step.
8. Variants: Includes methods like TD(λ) and SARSA, each with unique approaches to learning from temporal differences.
9. Applications: Useful in problems where the environment is partially known or dynamic.
10. Significance: TD learning is a bridge between Monte Carlo methods (learning from complete sequences) and dynamic programming (learning from known models).

10. Explain about PROLOG-EBG, in detail.

1. Definition: PROLOG-EBG is a machine learning technique within the framework of the Prolog programming language, focused on Explanation-Based Generalization.
2. Explanation-Based Generalization: It involves learning from a single example by creating a general rule from a specific problem-solving instance.
3. Use of Prolog: PROLOG-EBG leverages Prolog's strengths in symbolic reasoning and logic programming for learning.
4. Process: The system first solves a problem, then explains the solution in terms of its domain knowledge.
5. Generalization Step: The explanation is generalized into a rule that can be applied to new, similar problems.
6. Efficiency in Learning: By learning from specific instances, PROLOG-EBG can efficiently create rules without requiring large datasets.
7. Application Areas: Useful in domains like natural language processing, expert systems, and knowledge-based systems.
8. Knowledge Representation: Utilizes Prolog’s declarative nature for effective representation of knowledge and explanations.
9. Integration with Expert Systems: Can be integrated into expert systems for enhancing their problem-solving capabilities.
10. Challenges and Limitations: The quality of generalization depends heavily on the underlying domain knowledge and the representational capabilities of Prolog.

11.a) Discuss about augment search operators.

1. Purpose: Augmented search operators are used to expand the capabilities of basic search operators in problem-solving and AI.
2. Enhancing Search Space Navigation: They help in navigating complex search spaces more effectively.
3. Incorporating Domain Knowledge: Often includes domain-specific knowledge for more informed search decisions.
4. Adaptability: Can adapt to different problem scenarios, improving the flexibility of the search process.
5. Improving Efficiency: Aimed at reducing the search time and computational resources required.
6. Handling Constraints: Better at handling constraints and special conditions within the search space.
7. Dynamic Adjustment: Capable of adjusting strategies based on the current state of the search.
8. Integration with Heuristics: Often integrated with heuristic methods for more effective problem-solving.
9. Application in Various Algorithms: Used in algorithms like genetic algorithms, A* search, and reinforcement learning.
10. Balance Between Exploration and Exploitation: Aim to balance between exploring new areas of the search space and exploiting known good areas.

11. **b) Explain about search control knowledge in detail.**

1. Definition: Search control knowledge refers to the knowledge used to guide and optimize the search process in AI systems.
2. Role in AI: Critical for enhancing the efficiency of problem-solving by focusing the search.
3. Heuristics: Often involves heuristics, which are rules of thumb that guide the search towards more promising areas.
4. Learning from Past Searches: Involves learning effective search strategies from past experiences.
5. Dynamic Adaptation: Can adapt search strategies based on real-time feedback.
6. Types: Includes domain-specific knowledge, general problem-solving strategies, and learned heuristics.
7. Influence on Search Behavior: Impacts how a search algorithm chooses the next state to explore.
8. Improvement of Search Efficiency: Aims to reduce the time and computational resources needed for search.
9. Application Examples: Used in pathfinding algorithms, optimization problems, and game playing.
10. Challenges: Balancing the completeness and optimality of the search with practical constraints like time and memory.