|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Register Number |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |



**SRM Institute of Science and Technology**

**Set - B**

**College of Engineering and Technology**

**School of Computing**

**Department of Networking and Communication**

SRM Nagar, Kattankulathur – 603203, Chengalpattu District, Tamil Nadu

**Academic Year: 2023-24 (EVEN)**

**Test: CLA-T3** **Date: 07.05.2024**

**Course Code & Title: 18CSC305J – Artificial Intelligence**  **Duration:** 50 minutes

**Year & Sem: III Year / VI Sem** **Max. Marks:** 25

**Course Articulation Matrix:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S. No | Course Outcome | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| 1 | CO1 | 3 | 2 | 3 | - | - | - | - | - | - | - | - | - |
| 2 | CO2 | 3 | 2 | 3 | - | - | - | - | - | - | - | - | - |
| 3 | CO3 | 2 | 3 | 3 | - | - | - | - | - | - | - | - | - |
| 4 | CO4 | 2 | 3 | 2 | - | - | - | - | - | - | - | - | - |
| 5 | CO5 | 2 | 3 | 3 | 2 | - | - | - | - | - | - | - | - |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Answer all Questions**  **(1 x 10 = 10 Marks)** | | | | | | |
| Q. No | Question | Marks | BL | CO | PO | PI Code |
| 1. | The learner is trying to predict housing prices based on the size of each house. What type of regression is this?  Ans: Linear Regression | 1 | 1 | 4 | 2 | 2.1.3 |
| 2. | In the ensemble learning method, the whole collection or ensemble of hypothesis is selected from the hypothesis space and their predictions are combined. Whether the given statement is True or False  Ans: True | 1 | 1 | 4 | 3 | 2.1.3 |
| 3. | The hypothesis is given by h(x) = t0 + t1x. What are t0 and t1? a) Value of h(x) when x is 0, intercept along y-axis b) Value of h(x) when x is 0, the rate at which h(x) changes with respect to x c) The rate at which h(x) changes with respect to x, intercept along the y-axis d) Intercept along the y-axis, the rate at which h(x) changes with respect to x  Ans: D | 1 | 1 | 4 | 3 | 2.1.3 |
| 4-6. | Across:   1. Popular Activation Function 2. Signal that defines a goal in reinforcement learning   Down:   1. A model that generates a numerical prediction   Optimizer  Ans: ReLU, Reward, Regression | 3 | 1 | 4 | 3 | 2.1.3 |
| 7. | Reinforcement learning is a type of unsupervised learning where the model learns from rewards or punishments.  Ans: TRUE | 1 | 1 | 4 | 3 | 2.1.3 |
| 8. | In rule-based systems, the \_\_\_\_\_\_\_\_ determines which rule antecedents are satisfied by the facts.  Ans: Inference Engine | 1 | 1 | 5 | 2 | 2.2.3 |
| 9. | Which of the component stores all relevant information, data, rules, cases, and relationships used by the expert system?  Ans: Knowledge base | 1 | 1 | 5 | 2 | 2.2.3 |
| 10. | \_\_\_\_\_\_\_\_\_\_\_\_ is a branch of AI that enables computers to understand, interpret, and process human language in a way that is meaningful and contextually relevant.  Ans: NLP | 1 | 1 | 5 | 2 | 2.2.3 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Answer any three Questions**  **(3 x 5 = 15 Marks)** | | | | | | |
| Q. No | Question | Marks | BL | CO | PO | PI Code |
| 11. | Describe the process of building an Agent-Based Model. What are the steps involved in defining agents, their behaviors, and the environment they interact within?   1. **Problem Identification and Conceptualization**:    * Clearly define the problem or phenomenon that you want to model using an ABM.    * Identify the entities (agents) that will participate in the model and the interactions among them. 2. **Agent Definition**:    * Define the characteristics and attributes of the agents in the model. This includes identifying the relevant variables that describe the agents' state and behavior.    * Determine the types of agents that will be present in the model and any variation within those types.    * Specify how agents are initialized, including their initial state and starting conditions. 3. **Environment Specification**:    * Define the environment within which the agents will interact. This includes determining the spatial layout, resources, constraints, and other features of the environment.    * Decide how the environment will be represented (e.g., grid-based, network-based, continuous space).    * Establish the rules governing agent-environment interactions, including how agents perceive and respond to changes in the environment. 4. **Behavior Specification**:    * Define the behaviors of the agents based on their objectives, constraints, and interactions with other agents and the environment.    * Specify the decision-making processes of the agents, including any rules, heuristics, or algorithms they use to make choices.    * Determine how agents communicate and interact with each other, including the types of messages they can exchange and the protocols for communication. 5. **Model Implementation**:    * Implement the model using a suitable programming language or simulation platform.    * Translate the conceptual model into executable code, including defining data structures, algorithms, and simulation procedures.    * Verify the correctness of the implementation through testing and validation against known results or empirical data. 6. **Parameterization and Calibration**:    * Assign initial values to the parameters of the model, including agent attributes, environmental factors, and behavioral rules.    * Calibrate the model by adjusting parameters to match observed data or desired outcomes.    * Sensitivity analysis may be conducted to assess the impact of parameter variations on model behavior. 7. **Model Validation**:    * Validate the model by comparing its output to real-world observations or data from empirical studies.    * Assess the model's ability to reproduce known patterns, behaviors, and dynamics observed in the target system.    * Iterate on model refinement and validation as needed to improve its accuracy and reliability. 8. **Analysis and Interpretation**:    * Analyze the results of the simulation to gain insights into the behavior and dynamics of the modeled system.    * Interpret the model outputs in the context of the research question or problem being addressed.    * Explore the effects of different scenarios, interventions, or policy changes on the system's behavior. | 5 | 3 | 4 | 2 | 2.6.3 |
| 12. | Compare Support Vector Machines (SVM) and K-Nearest Neighbors (KNN) algorithms in machine learning. Discuss their advantages, disadvantages, and provide examples of scenarios where each algorithm would be appropriate.  Support Vector Machines (SVM) and K-Nearest Neighbors (KNN) are both popular algorithms in machine learning, but they differ in their approach to classification and regression tasks. Here's a comparison of the two algorithms along with their advantages, disadvantages, and examples of appropriate scenarios:  **Support Vector Machines (SVM)**:   * **Approach**: SVM is a supervised learning algorithm that finds the optimal hyperplane to separate data points into different classes. It aims to maximize the margin between classes, thereby improving generalization. * **Advantages**:   + Effective in high-dimensional spaces.   + Versatile: Can handle linear and non-linear classification, as well as regression tasks through kernel trick.   + Robust to overfitting, especially in high-dimensional spaces. * **Disadvantages**:   + Computationally expensive, especially for large datasets.   + Can be sensitive to the choice of kernel and regularization parameters.   + Not well-suited for very large datasets or datasets with noise. * **Appropriate Scenarios**:   + Text categorization and document classification.   + Image classification.   + Bioinformatics applications such as protein classification.   + Financial forecasting and stock market prediction.   **K-Nearest Neighbors (KNN)**:   * **Approach**: KNN is a simple and intuitive algorithm that classifies data points based on the majority class of their nearest neighbors. It makes predictions by calculating the distance between data points in the feature space. * **Advantages**:   + Easy to understand and implement.   + No training phase; the algorithm is instance-based.   + Works well with small datasets and non-linear decision boundaries. * **Disadvantages**:   + Computationally expensive during prediction, especially for large datasets.   + Sensitive to irrelevant features and the choice of distance metric.   + Requires careful normalization of features. * **Appropriate Scenarios**:   + Recommender systems, such as movie or product recommendations.   + Anomaly detection, where outliers can be identified based on their proximity to neighboring points.   + Pattern recognition tasks, such as handwritten digit recognition.   + Medical diagnosis, where similar patient cases are used to predict disease outcomes. | 5 | 4 | 4 | 3 | 2.6.4 |
| 13. | Compare and contrast Classification and Regression in the context of Supervised Learning. Provide examples of each type of problem and discuss how algorithms like Logistic Regression and Linear Regression are used  Classification and regression are two fundamental types of supervised learning tasks that differ in their output variables and objectives.  **Classification**:   * **Output Variable**: In classification, the output variable is categorical or discrete, representing different classes or categories. * **Objective**: The objective of classification is to predict the class label or category of new data points based on their features. * **Examples**:   + Spam detection: Classifying emails as spam or not spam.   + Disease diagnosis: Predicting whether a patient has a certain disease based on medical test results.   + Image classification: Identifying objects or patterns in images, such as classifying images of animals. * **Algorithms**:   + Logistic Regression: Despite its name, logistic regression is commonly used for binary classification problems. It models the probability of the input belonging to one of two classes using a logistic function. It's widely used for its simplicity and interpretability.   + Support Vector Machines (SVM): SVM is a versatile algorithm that can be used for both binary and multiclass classification tasks. It finds the hyperplane that best separates the classes in the feature space, maximizing the margin between them.   + Decision Trees: Decision trees partition the feature space into regions and make predictions based on majority voting within each region. They are intuitive and can handle both categorical and numerical data.   **Regression**:   * **Output Variable**: In regression, the output variable is continuous, representing a quantity or a real number. * **Objective**: The objective of regression is to predict a continuous value or outcome based on input features. * **Examples**:   + House price prediction: Predicting the price of a house based on its features such as size, location, and number of bedrooms.   + Stock price forecasting: Predicting the future price of a stock based on historical price data and other market variables.   + Temperature prediction: Forecasting the temperature for the next day based on historical weather data and environmental factors. * **Algorithms**:   + Linear Regression: Linear regression models the relationship between the input features and the output variable as a linear equation. It finds the best-fitting line through the data points to make predictions.   + Polynomial Regression: Polynomial regression extends linear regression by fitting a polynomial function to the data. It can capture non-linear relationships between variables.   + Ridge Regression and Lasso Regression: These are regularization techniques applied to linear regression to prevent overfitting and improve generalization performance.   **Usage of Logistic Regression and Linear Regression**:   * **Logistic Regression**: Logistic regression is commonly used for binary classification tasks, such as predicting whether an email is spam or not spam, whether a patient has a disease or not, or whether a customer will churn or not churn. * **Linear Regression**: Linear regression is used for regression tasks where the output variable is continuous. It's employed in various domains such as predicting house prices, forecasting sales revenue, estimating demand for products, and analyzing the relationship between variables in scientific research. | 5 | 3 | 4 | 3 | 3.6.1 |
| 14. | Provide an example scenario illustrating the development, functioning, and limitations of an expert system  **Scenario: Development of a Medical Diagnosis Expert System**  **1. Problem Identification:**   * **Problem**: Patients often visit doctors with symptoms that could indicate various medical conditions. Diagnosing these conditions accurately and efficiently is crucial for effective treatment. * **Objective**: Develop an expert system that can assist doctors in diagnosing medical conditions based on patient symptoms and medical history.   **2. Knowledge Acquisition:**   * **Expertise**: Gather knowledge from experienced doctors specializing in different medical fields. This includes diagnostic rules, medical guidelines, treatment protocols, and case studies. * **Data**: Collect anonymized patient data, including symptoms, test results, diagnoses, and treatment outcomes, to train and validate the system.   **3. Knowledge Representation:**   * **Knowledge Base**: Organize acquired knowledge into a structured format suitable for computational processing. This may involve representing diagnostic rules as production rules or decision trees, encoding medical knowledge into a knowledge base management system, or using ontologies to define medical concepts and relationships.   **4. System Development:**   * **Inference Engine**: Develop the inference engine responsible for reasoning and making diagnostic decisions based on the knowledge stored in the knowledge base. Implement reasoning algorithms such as forward chaining or backward chaining to derive diagnoses from patient symptoms. * **User Interface**: Design a user-friendly interface for doctors to input patient data, view diagnostic results, and interact with the expert system. The interface should provide explanations for diagnostic decisions and allow doctors to ask questions or seek clarification.   **5. Testing and Validation:**   * **Validation**: Test the expert system using a diverse set of patient cases, including common and rare medical conditions. Compare the system's diagnoses with those made by human experts to assess accuracy and reliability. * **Iterative Improvement**: Iterate on the system based on feedback from medical professionals and refine the knowledge base and inference engine to improve diagnostic performance.   **6. Deployment and Functioning:**   * **Deployment**: Deploy the expert system in clinical settings, such as hospitals or clinics, to assist doctors in diagnosing patients. * **Functioning**: Doctors input patient symptoms and medical history into the system, which analyzes the data and provides diagnostic suggestions along with explanations for its decisions. Doctors can use this information to confirm diagnoses, order additional tests, or recommend treatments.   **7. Limitations and Challenges:**   * **Limited Scope**: Expert systems are often domain-specific and may not cover all medical conditions or account for rare or complex cases. * **Knowledge Acquisition**: Acquiring and updating the knowledge base requires ongoing effort and may be subject to biases or inaccuracies. * **Interpretability**: Complex diagnostic reasoning may be challenging to interpret or explain to users, leading to concerns about trust and accountability. * **Uncertainty**: Expert systems may struggle to handle uncertainty in diagnosis, such as ambiguous symptoms or conflicting evidence, leading to potential errors or misdiagnoses. | 5 | 3 | 5 | 3 | 2.7.1 |