AI Starter Task: Semester 2

Introduction to AI Modules - Kaggle

AI Task 0 - Kaggle Courses

Objective: Complete the introductory and intermediate Machine Learning modules on Kaggle and obtain certificates for each course.

1. **Create a Kaggle Account:** If you haven't already, create an account on Kaggle (https://www.kaggle.com/).
2. **Complete Courses:**
   - Introduction to Machine Learning
   - Intermediate Machine Learning
3. **Certificate Submission:** Upon completion of each course, download the certificate of completion.

Submission: Upload both certificates (.pdf or .jpg format) as proof of completion.

AI Task 1 - Classification Algorithms

Objective: Explore classification algorithms using a dataset from the UCI Machine Learning Repository.

1. **Dataset Selection:** Choose a classification dataset from the UCI Machine Learning Repository (https://archive.ics.uci.edu/ml/index.php). Select a dataset that interests you and has clear classification labels.
2. **Data Preprocessing:** Perform necessary data preprocessing steps such as handling missing values, encoding categorical variables, and scaling features.
3. **Classification Algorithms:** Apply various classification algorithms such as Logistic Regression, Decision Trees, Random Forest, Support Vector Machines, k-Nearest Neighbours, etc.
4. **Model Evaluation:** Compare the performance of different classifiers using appropriate evaluation metrics such as accuracy, precision, recall, F1-score, and ROC curves.
5. **Submission:** Prepare a Jupyter notebook (.ipynb) documenting your analysis, including code, visualizations, and interpretations.

AI Task 2 - Implementing Classifiers

Objective: Implement your own classifiers - Linear Regression and Naive Bayes - in Python or MATLAB.

1. **Linear Regression Classifier:**
   - Implement a simple linear regression classifier from scratch using Python or MATLAB.

    - Utilize gradient descent or any other optimization technique for parameter estimation.
    - Test the classifier on a synthetic dataset or a real-world dataset.

2. **Naive Bayes Classifier:**
    - Implement a Naive Bayes classifier from scratch using Python or MATLAB.
    - Handle discrete and continuous features appropriately using the naive Bayes assumption.
    - Test the classifier on a suitable dataset and evaluate its performance.
3. **Submission:** Provide the implementation code for both classifiers along with a brief explanation of the algorithms and results obtained.

Bonus Task - Introduction to Deep Learning

Objective: Gain an understanding of the basics of Deep Learning, focusing on feedforward neural networks.

1. **Explore Deep Learning Concepts:**
    - Learn about the fundamentals of Deep Learning, including neural networks, activation functions, loss functions, and optimization algorithms.
    - Understand the architecture of a feedforward neural network, including input layer, hidden layers, and output layer.
2. **Implement a Simple Feedforward Network:**
    - Build a simple feedforward neural network using a deep learning framework like TensorFlow or PyTorch.
    - Train the network on a toy dataset or a simple classification problem.
3. **Review and Analysis:**
    - Write a review summarizing your understanding of the training process in a feedforward neural network.
    - Explain key terms such as forward pass, backward pass, gradient descent, epochs, and batch size.
    - Reflect on challenges faced and insights gained during the implementation.
4. **Submission:** Provide a written review detailing your understanding of the training process in a feedforward neural network, along with any code or visualizations produced during the implementation