



SCHOOL OF SCIENCE AND TECHNOLOGY

APT 3025 APPLIED MACHINE LEARNING ASSIGNMENT 3

Instructions: This is a group work. However, each individual is required to submit his or her work.

Machine Learning in Real-World Applications

Objective: Design and implement a machine learning-powered solution to address a real-world problem in a chosen domain (e.g., inventory systems or precision agriculture). Students will apply machine learning algorithms in Python and deploy the resulting model in a software solution using tools like Node.js, Next.js/React, or other web development frameworks. Creativity and originality are encouraged in exploring diverse use cases.

Project Requirements:

1. Define the Problem:

- i). Identify a practical problem in the chosen domain:
 - **Inventory Systems:** For example, optimize stock levels using demand prediction or automate stock categorization using image recognition.
 - **Precision Agriculture:** For example, predict crop yield based on weather and soil data or detect plant diseases using computer vision.
- ii). Explain the importance and impact of solving this problem.

2. Data Acquisition and Preprocessing:

- i). Collect relevant data (e.g., customer demand data for inventory systems or weather and soil data for precision agriculture).
- ii). Clean, preprocess, and explore the data. Provide visualization and statistical summaries.

3. Model Selection and Training:

- i). Choose an appropriate machine learning algorithm (e.g., regression, classification, clustering, etc.) based on the problem.
- ii). Train the model using Python, employing libraries like Scikit-learn, TensorFlow or PyTorch.

4. Evaluate and Fine-Tune:

- i). Test the model on validation data and assess its performance using metrics like accuracy, RMSE, or F1-score.
- ii). Fine-tune hyperparameters to improve results.

5. Software Integration:

- i). Deploy the trained model into a software application:

- Use tools like Node.js to build back-end services that host and query the machine learning model.
 - Develop a user-friendly interface using frameworks like Next.js or React for web applications.
- ii). Ensure the software allows users to interact with the model (e.g., input predictions, view results, or visualize insights).

6. **Final Deliverables:**

- i). A technical report detailing:
- Problem definition, data preparation, model implementation, and deployment process.
- ii). A fully functional software solution with:
- Real-time model predictions.
 - A deployed and testable application.
- iii). A presentation explaining the project workflow, challenges, and results.