CP468: Artificial Intelligence

Winter 2025

Assignment #2

Due March 21, 2025

Problem #1: A retail company analyzes customer behaviour to improve product recommendations. They have data on 20 customers, represented in a 2-dimensional space with two purchasing behaviour metrics:

x_1: Average amount spent per purchase (in dollars).

x_2: Frequency of purchases per month.

Each customer belongs to one of two categories (y) regarding their likelihood of purchasing a premium membership:

- y = 0 → Not interested (blue dots).
- y = 1 → Interested (red dots).

The company wants to use the k-nearest neighbours (k-NN) algorithm to classify new customers based on this data stored in CustomerDataset_Q1.csv (columns: x_1, x_2, y).

a) Create a 2D scatter plot of the customer data using Python (e.g., Matplotlib):
   - Plot x_1 (x-axis) vs. x_2 (y-axis).
   - Use blue dots for y = 0 and red dots for y = 1.
   - Label axes and include a legend.

b) Implement a function fnKNN(dataset, new_point, k) in Python (no external ML libraries like scikit-learn) to perform k-NN classification:
   - Input: dataset (list of [x_1, x_2, y] rows), new_point (list [x_1, x_2]), k (integer).
   - Use Euclidean distance: $\sqrt{((x_{11} - x_{12})^2 + (x_{21} - x_{22})^2)}$.
   - Output: Predicted y (0 or 1) based on majority vote of k nearest neighbors.

c) Assess the k-NN classifier's performance with k = 1:
   - Split the dataset into training and test sets: 80% (16 train, 4 test), 60% (12 train, 8 test), 50% (10 train, 10 test).
   - Use a fixed random seed (e.g., 42) for reproducibility.
   - Report accuracy (fraction of correct predictions) for each split and briefly comment on trends.

d) Repeat Task C for k = 2, 3, and 4. Report accuracies in a table and explain:
   - How *k* affects performance.
   - How training set size impacts results.

e) Which combination seems best and why?

Problem #2: A marketing analytics team at a Waterloo-based e-commerce company wants to understand customer behaviour by grouping similar customers into clusters. The team has collected a dataset (CustomerProfiles_Q2.csv) containing 30 customer profiles, with two key features representing each customer:

- Feature 1 (x1): Customer's average monthly spending
- Feature 2 (x2): Customer's total number of purchases

To segment customers into different groups, the team uses K-means clustering.

A) Plot the customer data points in a 2D graph before clustering to visualize how customers are distributed.
B) Write a function k-means() in Python or Java to implement the K-Means clustering algorithm for k clusters, ensuring:
    - Step 1: Select the first k data points as initial centroids.
    - Step 2: Assign each data point to the closest centroid based on Euclidean distance.
    - Step 3: Label each data point with its respective cluster.
    - Step 4: Compute new centroids for each cluster by averaging the data points in the cluster.
    - Step 5: Repeat steps (2) to (4) until the centroids stop changing.
C) Plot the final clusters when k = 2, clearly showing how the data points are grouped.
D) Report how many data points are in each cluster after finishing the clustering process.
E) Reporting Final Centroids. Provide the final centroid coordinates for each cluster.

Problem #3: A smart farming company is developing an AI-based weather prediction model to help farmers make better decisions about irrigation and crop protection decisions. They want to use historical weather data (temperature and humidity) to predict whether it will rain.
To achieve this, they will train a Perceptron-based AI model to classify weather conditions into two categories:

- y = 0 (No Rain)
- y = 1 (Rain)

The company collects data from sensors and stores it in weather_2025.csv, which contains 20 recorded instances of weather conditions with:

- x_1 = Temperature (scaled between 0 and 1)
- x_2 = Humidity (scaled between 0 and 1)
- y = Rain (0 or 1)

The company needs to analyze this dataset and evaluate the performance of a Perceptron model in predicting rainfall. **Do not use scikit-learn or external machine-learning libraries**.

a) Plot the weather data on a 2D graph where:
    - No Rain (y = 0) is represented as blue squares.
    - Rain (y = 1) is represented as red circles.
    - Include labels, a title, and a legend for clarity.

b) Implement a Perceptron Model (Manually, Without Scikit-Learn)
- Implement a perceptron algorithm manually in Python or Java
- Train the model using weather_2025.csv as input.
- Procedure:
    - Split the dataset into a training set (first 15 instances) and a test set (last five instances)
    - Train the perceptron for a maximum of 1000 iterations using temperature and humidity as inputs to predict rainfall
    - Report the training and test accuracy after training
    - The perceptron must be implemented from scratch (i.e., manually coding the weight updates, activation function, and learning algorithm)
    - Use a learning rate of 0.1
    - Initialize weights randomly between -0.5 and 0.5
    - Include a bias term in your implementation

c) Evaluate the Perceptron's Performance:
- Does the perceptron separate the two classes (rain vs. no rain)?
- Based on the dataset's pattern, explain **why** or **why not** the perceptron works well (or fails).
- Suggest possible improvements (e.g., using a different model or feature engineering).
- Plot the decision boundary on the same graph as your data points
- Experiment with at least two different train/test splits and compare the results
- Discuss how different learning rates might affect the model's performance

---

Problem #4**:** A human resources (HR) department is scheduling four employees (X_1, X_2, X_3, X_4) for different shifts. Shifts are numbered sequentially (e.g., 1, 2, 3, etc.), where higher numbers represent later shifts. Each employee has a set of available shift options (domains):

- X_1 (Employee 1): {1, 2, 3, 4}
- X_2 (Employee 2): {3, 4, 5, 8, 9}
- X_3 (Employee 3): {2, 3, 5, 6, 7, 9}
- X_4 (Employee 4): {3, 5, 7, 8, 9}

The following constraints must be enforced:

1. $X_1 \geq X_2 \rightarrow$ Employee 1's shift must be later than or equal to Employee 2's shift.
2. $X_2 > X_3 \rightarrow$ Employee 2's shift must be later than Employee 3's shift.

3. $X_3 \neq X_4 \rightarrow$ Employee 3 and Employee 4 must not have the same shift.

a) Draw the constraint graph for this scheduling problem, showing variables as nodes and constraints as directed arcs.
b) Apply the AC-3 algorithm to achieve arc consistency:

- List the initial queue of arcs.
- Show step-by-step which domains are revised (if any).
- Provide the final domains after achieving arc consistency.

  c) Is the scheduling network arc-consistent? If not, identify which constraint(s) cause the inconsistency.

  d) If the network is arc-consistent, find one possible schedule (assignment of shifts to X_1, X_2, X_3, X_4) and verify it satisfies all constraints. If not, explain why no solution exists.

  e) Suppose the HR department adds the constraint "$X_1 \neq X_4$". Revise the domains from your answer in (b) to enforce arc consistency with this new constraint, and state whether the network remains arc-consistent.

Submission Instructions
Deadline: See the calendar in MLS for the deadline.
By the assignment deadline, one designated person from the group will need to upload the following to MLS:

- A report in PDF format. Please ensure that all group members' names and SIDs are included at the top of the report. The report file should be named **group_number_asn2.pdf**.
- Your source code is compressed to a single ZIP file. It should be well commented, and the correspondence between the code and the report should be clear. You don't need to include executables or various supporting files (e.g., utility libraries) whose content is irrelevant to the assignment. We will contact you if we encounter any minor issues running your code to evaluate your solution.
- Failure to excuse the code after submission will result in a penalty of 50% off the total question grade.
- The name of the code archive should be **group_number _asn2.zip**.
- Multiple attempts for submission will be allowed, but only your last submission before the deadline will be graded.
- We reserve the right to take off points for not following directions.
- Any updates related to the assignment will be posted on MyLS.

Good Luck!