# Introduction to Artificial Intelligence, Fall 2024 Midter Exam (100 pts)

Your name: _____ Student ID: _____ Date: _____

**Problem 1. Gradient Descent. [10 pts]** We are given a single training item $x = [x_1, x_2] = [3, 8]$, $y = 1$ and asked to train a linear perceptron, i.e., $y = f(x) = w_0 + w_1 x_1 + w_2 x_2$, initialized with $w = [w_0, w_1, w_2] = [0, 1, 1]$ and with a learning rate $\alpha = 0.3$.
**(a) [3 pts]** Calculate the initial squared error.
**(b) [7 pts]** Perform a single gradient descent step and give the new values for $w$.

**Problem 2. k-means. [15 pts]** Consider the following dataset of 5 points in a two-dimensional space:

$$A = (1, 2), \quad B = (1, 4), \quad C = (5, 2), \quad D = (5, 4), \quad E = (3, 3)$$

**(a) [6 pts] (Calculation)**: Perform one iteration of the k-means algorithm to cluster these points into $k = 2$ clusters. Use the following steps:

1) Assume that points $A$ and $D$ are the initial centroids of the clusters.
2) Assign each point to the nearest centroid based on Euclidean distance.
3) Update the centroids by calculating the mean of the points in each cluster.

Show your calculations for each step.

**(b) [4 pts] (Proof)**: Explain why the k-means algorithm is guaranteed to converge.

[Hints] Prove that the algorithm will reach a point where the cluster assignments no longer change by discussing the concept of within-cluster sum of squares (WCSS) and how k-means minimizes this value iteratively.

**(c) [5 pts] (Proof)**: How about k-medoids algorithm? Is it guaranteed to converge? Prove or disprove it.

**Problem 3. Naïve Bayes Network [22 pts]** Using a small subset of data from an experiment of a game, construct a Naïve Bayes Network to predict a player's first action (`MoveFirst`) in close-range engagements under various circumstances.

- **Urban**: Indicates the environment, where `Urban = True` means the subject was in a city, and `Urban = False` means they were in a forest.
- **Day**: Indicates the time of the encounter, where `Day = True` means it happened in daylight, and `Day = False` means it happened at night with night-vision equipment.
- **Within50m**: Indicates the range, where `Within50m = True` means the target was first observed at a range less than 50 meters, and `Within50m = False` means it was observed at a range up to 100 meters.

- **MoveFirst**: Indicates the subject's first action upon detecting the threat, where `MoveFirst = True` means the subject chose to move to cover first, and `MoveFirst = False` means they chose to fire immediately.

**HINT**: For this Naïve Bayes Network, you may use the following formula to calculate the probability of `MoveFirst` given the features:

$$P(\text{MoveFirst} \mid \text{Urban}, \text{Day}, \text{Within50m})$$

$$\propto P(\text{MoveFirst}) \cdot P(\text{Urban} \mid \text{MoveFirst}) \cdot P(\text{Day} \mid \text{MoveFirst}) \cdot P(\text{Within50m} \mid \text{MoveFirst})$$

| Urban | Day | Within50m | MoveFirst |
|---|---|---|---|
| true | true | true | false |
| true | true | true | false |
| true | true | true | true |
| false | true | false | false |
| true | false | false | false |
| true | false | false | true |
| true | false | false | false |
| false | false | true | true |
| false | false | true | false |
| false | true | false | false |
| false | false | false | true |
| false | false | false | false |

**(a) [1 pt]** Write the exact fraction for P(MoveFirst = true).
**(b) [1 pt]** Write the exact fraction for P(MoveFirst = false).
**(c) [1 pt]** Write the exact fraction for P( Urban = false | MoveFirst = false ).
**(d) [1 pt]** Write the exact fraction for P( Urban = false | MoveFirst = true ).
**(e) [1 pt]** Write the exact fraction for P( Urban = true | MoveFirst = false ).
**(f) [1 pt]** Write the exact fraction for P( Urban = true | MoveFirst = true ).
**(g) [1 pt]** Write the exact fraction for P( Day = false | MoveFirst = false ).
**(h) [1 pt]** Write the exact fraction for P( Day = false | MoveFirst = true ).
**(i) [1 pt]** Write the exact fraction for P( Day = true | MoveFirst = false ).
**(j) [1 pt]** Write the exact fraction for P( Day = true | MoveFirst = true ).
**(k) [1 pt]** Write the exact fraction for P( Within50m = false | MoveFirst = false ).
**(l) [1 pt]** Write the exact fraction for P( Within50m = false | MoveFirst = true ).
**(m) [1 pt]** Write the exact fraction for P( Within50m = true | MoveFirst = false ).
**(n) [1 pt]** Write the exact fraction for P( Within50m = true | MoveFirst = true ).
**(o) [8 pts]** Write the equation to predict the probability that a subject's first action is to shoot when he encounters a threat within 50 meters in a city at night. In order words, write the equation that could predict the probability in terms of only the probabilities above.

**Problem 4. Training and Testing. [8 pts]**
**(a) [4 pts]** Explain in one sentence why, given a that we'd like to train and evaluate our learner on a dataset, we split the data into a Training Set and a held aside Test Set instead of training and evaluating on the full dataset?
**(b) [4 pts]** If the testing performance is much worse than the training performance, what's happened? [2 pts] Any solutions? [2 pts]

**Problem 5. Decision Tree. [15 pts]**
Consider a dataset with the following attributes for predicting whether a person will buy a ticket to a concert:

- **Age**: Young, Middle-aged, Senior
- **Income**: Low, Medium, High
- **Student**: Yes, No
- **Credit Rating**: Fair, Excellent
- **Buy Ticket**: Yes, No (target variable)

The dataset is as follows:

| Age | Income | Student | Credit Rating | Buy Ticket |
|---|---|---|---|---|
| Young | High | No | Fair | No |
| Young | High | No | Excellent | No |
| Middle-aged | High | No | Fair | Yes |
| Senior | Medium | No | Fair | Yes |
| Senior | Low | Yes | Fair | Yes |
| Senior | Low | Yes | Excellent | No |
| Middle-aged | Low | Yes | Excellent | Yes |
| Young | Medium | No | Fair | No |
| Young | Low | Yes | Fair | Yes |
| Senior | Medium | Yes | Fair | Yes |
| Young | Medium | Yes | Excellent | Yes |
| Middle-aged | Medium | No | Excellent | Yes |
| Middle-aged | High | Yes | Fair | Yes |
| Senior | Medium | No | Excellent | No |

Compute the information gain to decide the best attribute for the first split for a decision tree.
[Hints] The formula for information gain is shown below.
**Entropy**: The entropy $H(S)$ for a dataset $S$ with two classes (e.g., Yes and No) is calculated as:

$$H(S) = -p_+ \log_2(p_+) - p_- \log_2(p_-)$$

where $p_+$ is the proportion of positive examples and $p_-$ is the proportion of negative examples in $S$.
**Information Gain**: The information gain $IG(S, A)$ of an attribute $A$ with respect to the dataset $S$ is calculated as:

$$IG(S, A) = H(S) - \sum_{v \in \text{Values}(A)} \frac{|S_v|}{|S|} H(S_v)$$

where Values$(A)$ is the set of all possible values of attribute $A$, $S_v$ is the subset of $S$ for which attribute $A$ has value $v$, and $H(S_v)$ is the entropy of $S_v$.

| Value | Approximate $\log_2$(Value) |
|---|---|
| 2 | 1.00 |
| 3 | 1.58 |
| 5 | 2.32 |
| 7 | 2.81 |
| 11 | 3.46 |
| 13 | 3.70 |

TABLE I: Approximate values of $\log_2$ for selected integers

**Problem 6. Convolutional Neural Nets. [10 pts]**
**(a) [3 pts]** Consider a small 6x6 grayscale image with pixel values represented as follows:

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

You have a 3x3 filter (kernel) with the following values:

$$\begin{bmatrix} 1 & -1 & -1 \\ -1 & 0 & -1 \\ -1 & -1 & 1 \end{bmatrix}$$

Perform a convolution operation with this filter on the image using a stride of 1 without padding. Calculate the resulting feature map.

**(b) [7 pts]** Consider a Convolutional Neural Network (CNN) with the following structure:

- The **first convolutional layer** takes an input image of size $32 \times 32 \times 3$ (height, width, channels) and applies 16 filters, each of size $3 \times 3$, with a stride of 1 and no padding.
- The **second convolutional layer** takes the output from the first layer and applies 32 filters, each of size $5 \times 5$, with a stride of 1 and no padding.

Calculate the exact number of parameters of **a filter** in the **second convolutional layer** [4 pts]. Moreover, what is the output size of the feature map after the second convolutional layer? [3 pts]

**Problem 7. Vision Transformer. [10 pts]**
**(a) [2 pts]** Is positional encoding in the original Transformer important for vision transformer? State the reason for scoring instead of ”Yes” or ”No”.
**(b) [4 pts]** What are the learnable parameters in the Vision Transformer? State at least four kinds of parameters.
**(c) [4 pts]** What is the main advantage/disadvantage of the vision transformer? [2 pts each]

**Problem 8. Smart Choices. [10 pts, 2 pts for each]**

8.1 You are designing a model to classify customer reviews as positive, negative, or neutral. You have a large dataset of labeled reviews available.
**What is the most appropriate learning method?**
A) Supervised Learning
B) Unsupervised Learning
C) Reinforcement Learning
D) Self-Supervised Learning
**Answer:**

8.2 A model needs to detect tumors from MRI scans, but only a few labeled tumor images are available due to the high cost of annotation. However, a vast amount of unlabeled medical images is accessible.
**What learning approach should be used to leverage both labeled and unlabeled data?**
A) Transfer Learning
B) Semi-Supervised Learning
C) Multi-Task Learning
D) Active Learning
**Answer:**

8.3 You are building an autonomous drone navigation system. The drone needs to continuously learn from streaming data while flying, as the environment may change dynamically.
**Which learning method best suits this scenario?**
A) Online Learning
B) Multi-Instance Learning
C) Supervised Learning
D) Ensemble Learning
**Answer:**

8.4 A company wants to implement a spam filter that will gradually improve by querying a human operator only for uncertain cases, reducing the need for continuous human labeling.
**Which learning method is most suitable for this task?**
A) Active Learning
B) Reinforcement Learning
C) Self-Supervised Learning
D) Multi-Source Learning
**Answer:**

8.5 You are developing a personal assistant AI that needs to handle various tasks such as scheduling, answering questions, and setting reminders. Each of these tasks requires its own model, but the system can improve by sharing knowledge between these tasks.
**Which learning method is best suited for this scenario?**
A) Multi-Task Learning
B) Transfer Learning
C) Semi-Supervised Learning
D) Reinforcement Learning
**Answer:**