Shiraz University

Course Title: Artificial Intelligence (2025)

Assignment 4: Discriminative vs. Generative Models

Deadline: 31 Tir 1404

---

## Theoretical Questions

### Q1:

Why don't we use the MSE error function in logistic regression?

### Q2:

Plot the sigmoid function 1/(1 + e^(-wX)) vs. X ∈ ℝ for increasing weight w ∈ {1, 5, 100}. A qualitative sketch is enough. Use these plots to argue why a solution with large weights can cause logistic regression to overfit.

### Q3:

Suppose the data come from a Bernoulli distribution with parameter $\theta$ θ, and we have $n$ n samples with $k$ k of them equal to 1. Write down the likelihood function and find the optimal value of $\theta$ θ using the Maximum Likelihood Estimation (MLE) method.

### Q4:

a) The table below is a training dataset consisting of 8 samples. The columns **color**, **legs**, **height**, and **smelly** represent the features of each sample. The column **species** is the target column with two classes: H and M. Using a Naive Bayes classifier, determine to which of the two classes, H or M, the following sample belongs:
Sample data:
Color = green, legs = 2, height = tall, smelly = no

| ID | Color | Legs | Height | Smelly | Species |
|---|---|---|---|---|---|
| 1 | White | 3 | Short | Yes | M |
| 2 | Green | 2 | Tall | No | M |
| 3 | Green | 3 | Short | Yes | M |
| 4 | White | 3 | Short | Yes | M |
| 5 | Green | 2 | Short | No | H |
| 6 | White | 2 | Tall | No | H |
| 7 | White | 2 | Tall | No | H |
| 8 | White | 2 | Short | Yes | H |

b) Write a Python code to solve the problem in part (a) using logistic regression. Obtain the regression weights using the Newton's method. Implement all steps from scratch, and initialize the weights to zero. Plot the error (loss) versus iteration. Using the numpy and matplotlib libraries is allowed. Clearly indicate the steps to compute the gradient, Hessian, etc., with comments in the code.

## Q5:

a) What are the differences between generative and discriminative methods?

b) In each of the following situations, which method is better to use? Why?

1)When we need a complex and flexible model

2)When we have missing values in the data

c) Among logistic regression and Naive Bayes (NB), which one is generative and which one is discriminative?

d) What assumptions must hold true to properly apply Naive Bayes?

e) We know that logistic regression estimates the probability that a sample belongs to the positive class. How does the logistic regression formula align with this interpretation?

f) We want to classify emails into spam and non-spam using the Naive Bayes method. Suppose the vocabulary set is as follows: ["buy", "now", "cheap", "meeting", "project", "schedule"] If the word "project" appears only in non-spam emails and never in spam emails, what problem does the Naive Bayes algorithm encounter? What is the solution to this problem?

## Programming Questions

### Title:

Discriminative vs. Generative Modeling for Cardiovascular Risk Prediction

### Scenario

A research team at Tehran University of Medical Sciences is developing an intelligent system to predict the risk of heart disease in patients visiting cardiology clinics. As a senior medical data analyst, you are responsible for implementing, analyzing, and comparing generative and discriminative models using real clinical data. Your analysis should focus on the **geometric and statistical properties** of their decision boundaries and assess how changes in data (e.g., noise, feature removal) affect performance and decision-making.

### Dataset

Heart Disease UCI Dataset
Source: https://www.kaggle.com/datasets/redwankarimsony/heart-disease-data

### Target Feature:

- target – Binary class label:
  - 0 → No heart disease
  - 1 → Heart disease

### Tasks

### Task 1: Data Exploration & Gaussian Fitting

- Handle missing or invalid values via deletion or imputation.
- Plot distributions (histogram or KDE) of all features per class.
- Select two informative features (e.g., chol and thalach).
- Fit 2D Gaussian distributions (one per class) and plot contour maps.
- Evaluate Gaussian assumption via plots and normality tests.

## Task 2: Generative Modeling (Gaussian Classifier

- Estimate mean and covariance matrix for each class.
- Derive and plot decision boundary (log-likelihood ratio).
- Compare:
    - LDA (shared covariance)
    - QDA (distinct covariance)
- Discuss impact of covariance assumption.

## Task 3: Gaussian Naive Bayes

- Implement from scratch assuming conditional independence.
- Estimate per-feature mean/variance per class.
- Compute posterior probabilities.
- Compare decision boundaries with Task 2.
- Discuss impact of correlation.

## Task 4: Discriminative Modeling (Logistic Regression)

- Implement from scratch using:
    - Log-likelihood
    - Gradient & Hessian
    - Newton's method
- Plot:
    - Convergence of log-likelihood
    - Decision boundary

## Task 5: Model Evaluation & Robustness

- Fixed train-test split (80/20, seed=42).
- Report: Accuracy, Precision, Recall, F1-score.
- Plot decision boundaries.
- Robustness Tests:

1. Add Gaussian noise to inputs.
2. Remove key feature (e.g., chol).
3. Add synthetic outliers (e.g., ±3σ to age, chol).

- Analyze boundary/performance changes.

## Task 6: Generative vs. Discriminative Analysis

- Compare models visually and numerically:
  - Decision boundary shapes
  - Generalization to noise, limited data, correlation
  - Interpretability, overfitting, data efficiency
- When and why is each model preferred?

## Task 7: Clinical Decision Scenario

- Given:
- age = 58, chol = 245, thalach = 140, trestbps = 130
- (other features = column mean)
- Predict probability of heart disease using each model.
- Provide numerical results and interpret model behavior with plots.

## Bonus

## A: Feature Engineering & Nonlinear Decision Boundary

- Create a nonlinear feature (e.g., chol_squared = $chol^2$)
- Retrain logistic regression with this feature.
- Analyze:
  - Decision boundary deformation
  - Linearity in transformed vs. original space

## B: Softmax Regression (Two-Class Case)

- Prove mathematically that softmax reduces to logistic regression for two classes.
- Implement softmax regression using your logistic code.
- Plot and compare its decision boundary with logistic regression.

## You are allowed to use only the following:

- All code must be implemented from scratch.
- Use only: numpy, pandas, and matplotlib.
- You may also use seaborn for data visualization.
- No machine learning libraries allowed (e.g., scikit-learn, statsmodels, tensorflow, xgboost, etc.).

## Important Notes

- Academic Integrity Any form of academic dishonesty will result in a zero grade for this assignment.
- You are allowed to use tools like GPT to assist you in writing code or understanding the concepts, but it is crucial that you fully understand the code you write and how it works. You may be required to explain your implementation during an assessment.
- Code Quality: Your code should be well-documented and follow good programming practices.
- You need to implement the algorithms from scratch. Using the built-in functions and algorithms is not allowed.
- Provide a report in PDF format and explain your code and results.
- The name of the uploading file should be "Firstname_Lastname_StdNumber.zip" of all group members.
- Collaboration Policy: While discussions with peers are allowed, your submitted work must be original.
- Late Submissions: Late submissions may incur penalties as per course policy.
- If you have any questions about anything, feel free to ask in the course's group chat.

## Good Luck!