**CITY UNIVERSITY OF HONG KONG**

Course code & title : CS5489 Machine Learning: Algorithms & Applications

Session : Midterm, Semester B 2024-25

Time allowed : Two hours (Mar 11th, 7:00pm-9:00pm)

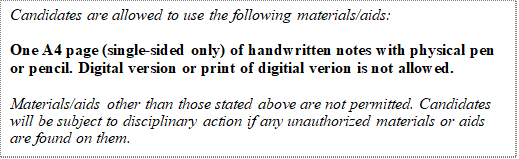
This question paper has TBD pages (including this cover page).

1. This paper consists of TBD questions.

2. Answer ALL questions.

3. Write your answers in this question paper.

*This is a* ***closed-book*** *examination.*



Student EID: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student ID: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Seat Number: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Multiple Choice/Selection Questions** (30 points)

5 marks each, select all that apply. *For a multiple selection question, an incorrect answer will be penalized 5/K marks, where K is the number of correct answers.* *If more incorrect answers are given than correct answers, the marks will be 0.*

**Q1** Which of the following statements regarding MLE are correct?

A) To obtain MLE estimates, optimizing for log likelihoods is equivalent to optimizing for normal likelihood.

B) MLE maximizes the likelihood of observing the data by adjusting the mean and variance of a Gaussian distribution, assuming that each class has a Gaussian distribution.

C) MLE assumes that the data distribution is uniform and uses it to estimate the class-conditional probabilities for each class.

D) In Gaussian MLE, the class-conditional probabilities are computed by the sample mean of the dataset alone, without considering the variance.

- Answer: AB

**Q2** Which of the following statements about Support Vector Machines (SVM) are correct?

A) The objective of SVM is to minimize the margin between the separating hyperplane and the nearest data points.  
B) In soft-margin SVM, slack variables are introduced to allow some misclassification.  
C) The margin on one side of the boundary could be larger than that of the other side.  
D) The kernel trick allows SVM to classify non-linearly separable data.

- Correct Answers: **BD**

**Q3**  Which of the following statements about the Naïve Bayes classifier is **not** correct?

A) It assumes features are conditionally independent given the class label.  
B) It directly models the posterior probabilityp(y|x) during training.  
C) It only works with Gaussian distributions.  
D) Adding Laplace smoothing helps prevent zero probabilities for unseen feature-class combinations.

- Correct Answers: BC

**Q4** Which statements about regression models are correct?

A) Ridge Regression shrinks coefficients toward zero but never sets them to exactly zero.

B) LASSO regression will always give us a more sparse solution than ridge regression.

C) Adding more training data always increases the risk of overfitting in linear regression.

D) Linear regression assumes a linear relationship between the independent variables and the dependent variable.

E) Linear regression has a closed form solution when using a L-2 loss.

- Correct Answers: **DE**

**Q5**  Which statements about Gaussian Process regression (GPR) are correct?

A) GPR is defined as the Bayesian linear regression whose linear kernel is replaced by the Gaussian/RBF kernel.

B) GPR has a closed form solution when the observation noise is Gaussian.

C) One assumption in the Gaussian process prior is that two function values are more correlated when the corresponding inputs are close together.

D) The reason why GPR is not good for large datasets is that it only has a few parameters and thus limited model complexity.

E) The inference result of GPR is a distribution.

- Correct Answers: BCE

**Q6** In the AdaBoost algorithm, which of the following is **false** regarding how it adjusts the weights of the training samples during each iteration?

A) AdaBoost decreases the weights of the misclassified examples after each iteration to focus more on correctly classified examples in future rounds.

B) AdaBoost increases the weights of misclassified examples to force the next weak learner to focus on those examples, while decreasing the weights of correctly classified examples.

C) AdaBoost is a sequential algorithm that does not take into account of the results of previous weak learners in a given iteration.

D) AdaBoost uses a fixed weight distribution throughout all rounds of learning, ensuring each training sample is treated equally.

- Correct Answers: ACD

**Discussion Questions (70 marks)**

**Q7 (10 Marks)** Gaussian Classifier

1. Explain the purpose of the Bayesian Decision Rule in a generative classifier. What is the role of the class-conditional distribution and prior distribution in this rule? [4 marks]

2. Compare the Naive Bayes Gaussian classifier and the Multivariate Gaussian Bayes classifier in terms of: Model assumptions, Parameter estimation, Decision boundaries and Suitability for high-dimensional data. [6 marks]

Answer:

1. The Bayesian Decision Rule selects the class y with the highest posterior probability p(y|x). It uses Bayes' theorem:

p(y|x) p(x|y)p(y) (1 marks)

Class-conditional distribution p(x|y): Models how features are distributed for each class (e.g., Gaussian for continuous features). (2 marks)

Prior distribution p(y): Represents the frequency of each class in the training data. The rule combines these to minimize classification error by maximizing the posterior. (1 mark)

2. Differences between Naive Bayes (NB) and Multivariate Gaussian

Model Assumptions

NB: Assumes features are conditionally independent given the class. Each feature is modeled with a univariate Gaussian. (1 mark)

Multivariate Gaussian: Models feature correlations using a covariance matrix. No independence assumption. (1 mark)

Parameter Estimation

NB: Estimates mean and variance per feature j for class c. (0.5 mark)

Multivariate Gaussian: Estimates a mean vector and full covariance matrix **Σ** for each class. (0.5 mark)

Decision Boundaries

NB: Linear or axis-aligned quadratic boundaries due to independence assumption. (0.5 mark)

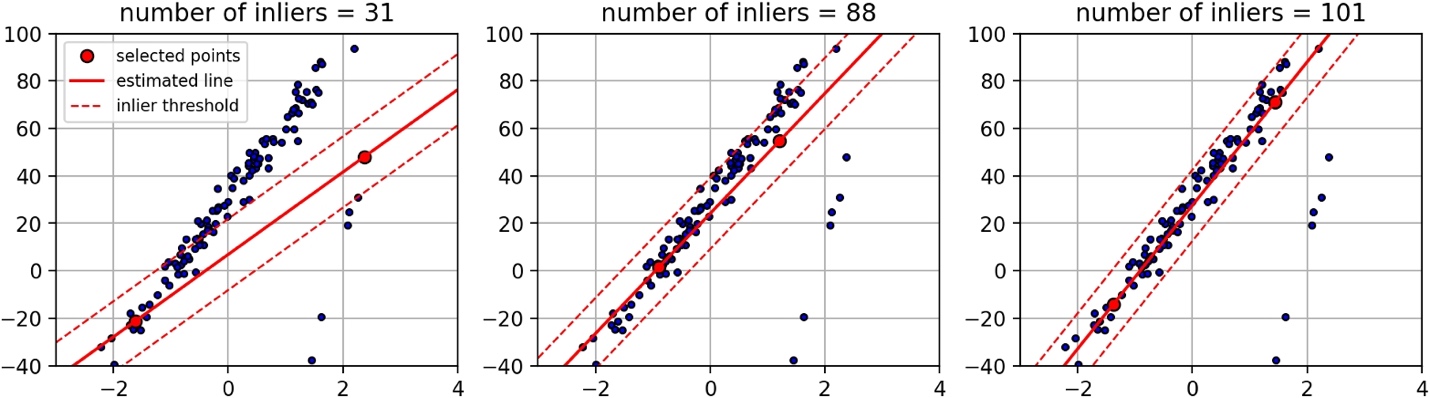
Multivariate Gaussian: Flexible quadratic boundaries that account for feature correlations.

High-Dimensional Data (0.5 mark)

NB: More efficient (fewer parameters) and less prone to overfitting. Suitable for text or datasets with many features. (1 mark)

Multivariate Gaussian: Requires estimating O(d^2) parameters (covariance matrix), which is computationally expensive for large dd. (1 mark)

**Q8 (10 Marks)**



1. In linear regression tasks, to mitigate the impact of outliers, the RANSAC algorithm is often employed. Briefly explain the principle of this method [4 marks]. The above figure shows the results of fitting different subsets of points to remove outliers. In your opinion, which subset seems to be better? [1 mark]

2. Briefly describe the core idea of the kernel trick? [4 marks] Please provide an example of a commonly used kernel function along with its mathematical formula. [1 mark]

**Reference Answer:**

1. RANSAC is a robust regression algorithm designed to handle datasets with a significant proportion of outliers. The algorithm iteratively selects a small random subset (1’) of data points, fits a model to this subset, and then classifies all data points as inliers or outliers (1’) based on a predefined residual threshold (1’). The subset yielding the highest number of inliers (1’) is considered the best model. This approach ensures that outliers do not heavily influence the final regression result. In the given figure, the subset with the highest number of inliers (101) is likely the best (1’), as it captures the underlying pattern of the data while minimizing the influence of outliers. (totally 5’)

2. The kernel trick is a technique used in machine learning to efficiently transform data into a higher-dimensional space (2’) without explicitly computing the transformation. Instead of mapping the data points, the kernel function computes the inner product (2’) of their transformed representations directly in the higher-dimensional space. This allows linear models to learn non-linear relationships. A commonly used kernel function is the Radial Basis Function (RBF) kernel, defined as (1’): (total 5’)

**Q9 (10 Marks)** Logistic Regression and Support Vector Machine (SVM), and Adaboost are all commonly used classifiers.

1. What is the key difference in the optimization objective of Logistic Regression and SVM? [6 marks]
2. Given this picture, please describe the three algorithms’ sensitivity to outliers in descending order, and explain why. [4 marks]

A graph with red blue and green lines

AI-generated content may be incorrect.

**Reference Answer:**

1. Optimization objectives

* 1. **Logistic Regression** aims to **maximize the probability (likelihood) of the data**, using the **logistic loss function** (cross-entropy loss). It models the probability of class membership using the sigmoid function. (3’)
  2. **SVM** aims to **maximize the margin between classes**, using a **hinge loss function**. It focuses on finding the decision boundary that maximizes the separation between the closest points (support vectors) of different classes. (3’)

2. The descending order is Adaboost, LR, and SVM. (2’) The reason being the increase in loss value is highest for exponential loss, second highest for logistic loss and lowest for hinge loss given the same raw (incorrect) classification. (2’)

**Q10 (10 Marks)** You are working for a transportation company in Hong Kong, and your task is to predict the number of passengers using the MTR (Mass Transit Railway) at a specific station each hour. The prediction will be used to allocate staff and resources efficiently. The management team emphasizes that it is acceptable to have slightly more staff than needed, but it is critical to avoid understaffing, as this would lead to poor customer experience and potential safety issues.

You are given a training dataset *(xi, yi)*, i=*1,…,N*, where *xi* represents features such as time of day, day of the week, weather conditions, and nearby events, and *yi* represents the number of passengers. You decide to use a linear regression model *f(x)=wTx+b* to make predictions.

**Part 1:** Design a custom loss function for this regression problem that aligns with the management’s requirements. Explain how your loss function works and why it is suitable for this task. [5 marks]

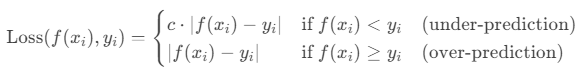
**Part 2:** Suppose your friend is also working on a similar problem but finds that their linear regression model performs no better than a "dummy" regressor that always predicts the mean number of passengers. What two pieces of advice would you give to your friend to improve their model? [3 marks]

**Part 3:** If the absolute prediction error ∣*f(xi) − yi*∣ is used as the loss function during training, what effect will this have on the model (when compared to a MSE model)? [2 marks]

Answer:

1. (5 points) **Loss function:**

One such loss function is an asymmetric loss function:



Here, *c* > 1 is a constant that controls how much more heavily under-predictions are penalized compared to over-predictions.

**Why this works:**

1. The loss function ensures that the model avoids under-predictions, which are critical to avoid understaffing.
2. Over-predictions are still penalized but less severely, as having slightly more staff than needed is acceptable.
3. (3 points) **Feature Engineering:** To create more informative features, such as: (a) Time-based features (b) Weather conditions. Linear regression relies on meaningful features to capture patterns in the data. Adding relevant features can improve the model’s predictive power.

**Regularization:** Suggest using Ridge (L2) or LASSO (L1) regularization to prevent overfitting and improve generalization. If the model is too simple or overfits noisy data, it may perform poorly on the validation set. Regularization helps balance complexity and generalization.

1. (2 points) The absolute error loss is less sensitive to outliers compared to the squared error loss (MSE).

As a result, the model will be more robust to extreme values in the data but may not predict the average number of passengers as accurately.

**Q11 (20 Marks)** Consider a dataset with samples , where each . Let be a known feature transformation, represented as:

The label vector, representing the model’s outputs, is given by:

The parameter vector to be estimated is:

Initially, we aim to minimize the squared loss:

**1. Regularization and Feature Selection [6 marks]**

(1) If we want to perform **linear regression in transformed space**, write down the loss functions incorporating **L1 regularization** and **L2 regularization**, respectively. Assume that the regularization term is weighted by . [2 marks]

**L2 Regularization (Ridge Regression)**: (1 marks)

**L1 Regularization (Lasso Regression)**: (1 marks)

(2) Please describe how the closed-form optimal value of could be derived for ridge regression and write down the solution (you’re not responsible for derivation process, just write down the results). Judging from the solution, what problem could we encounter if no regularization is present ()? [4 marks]

To find that minimizes the loss function with **L2 regularization**, we solve:

(1 mark)

Expanding the squared loss term:

Applying the derivative rule:

Solving for :

(2 marks)

If , then the inverse could be ill-posed (1 mark).

**2. Code Completion [10 marks]**

In this section, you will complete Python code snippets related to **feature selection techniques**. Some key parts of the code have been replaced with blanks (▢).

For each blank, select the most appropriate option (**A, B, C, or D**) from the multiple-choice answers provided below each code snippet.

* Questions 1-10 are worth 1 points each.

The following code implements **Ridge Regression** and **Orthogonal Matching Pursuit (OMP):**

|  |
| --- |
| import numpy as np  import matplotlib.pyplot as plt  from sklearn.linear\_model import Ridge  from sklearn.model\_selection import train\_test\_split  # Generate synthetic data  np.random.seed(114514)  X = np.random.randn(100, 10)  true\_theta = np.array([3, 0, 2, 0, 0, -1, 0, 0, 1, 0])  y = X ▢ true\_theta + np.random.randn(100) \* 0.5 **# this line has one ▢ corresponding to question (1)**  # Split into training and test sets  X\_1, X\_2, y\_1, y\_2 = train\_test\_split(▢, ▢, test\_size=0.2, random\_state=114514) **# this line has two ▢ corresponding to question (2)**  # Ridge Regression  def ridge\_regression(X\_train, y\_train, alpha=10):  ridge\_model = Ridge(alpha=▢) **# this line has one ▢ corresponding to question (3)**  ridge\_model.▢(X\_train, y\_train) **# this line has one ▢ corresponding to question (4)**  return ridge\_model.▢ **# this line has one ▢ corresponding to question (5)**  ridge\_result = ridge\_regression(X\_1, y\_1, alpha=10)  plt.figure()  plt.plot(▢, ridge\_result, marker='o') **# this line has one ▢ corresponding to question (6)**  plt.xlabel("Feature Index")  plt.ylabel("Coefficient Value")  plt.title("Ridge Regression Coefficients")  plt.show()  # Orthogonal Matching Pursuit (OMP)  def omp\_regression(X\_train, y\_train, n\_nonzero\_coefs=4):  n\_samples, n\_features = X\_train.shape  theta = np.zeros(n\_features) # Initialize coefficients  residual = ▢ **# this line has one ▢ corresponding to question (7)**    for \_ in range(n\_nonzero\_coefs):  j = np.argmax(np.abs(▢ @ residual)) **# this line has one ▢ corresponding to question (8)**  w\_j = np.dot(▢, residual) / np.dot(X\_train[:, j], X\_train[:, j]) **# this line has one ▢ corresponding to question (9)**  theta[j] = w\_j  residual -= ▢ \* w\_j **# this line has one ▢ corresponding to question (10)**    return theta  theta\_estimated = omp\_regression(X\_1, y\_1, n\_nonzero\_coefs=4)  print("Estimated Coefficients:", theta\_estimated) |

1. A. %

B. +

C. \*

D. @

D.

1. A. y; X

B. X; y

C. y\_1; X\_1

D. X\_1; y\_1

B.

1. A. alpha

B. lambda

C. 10

D. np.random.seed(114514)

A.

1. A. train

B. optimize

C. fit

D. minimize

C.

1. A. weights\_

B. theta\_

C. coef\_

D. params\_

C.

1. A. range(len(ridge\_result))

B. len(ridge\_result)

C. range(len(X))

D. len(X)

A.

1. Residual initialization:

A. np.ones\_like(y\_train)

B. np.ones\_like(y\_train)

C. X\_train[:, 0]

D. y\_train.copy()

D.

1. Select the most correlated feature:

A. X\_train

B. X\_train.T

C. theta

D. residual

B.

1. Compute the coefficient:

A. X\_train[j]

B. theta[j]

C. X\_train[:, j]

D. residual[j]

C.

1. Update the residual:

A. y\_train

B. X\_train[:, j]

C. theta[j]

D. residual

B.

**3. Explanation of Explicit Sparsity Constraints [4 marks]**

In previous formulations, **Lasso and Ridge Regression** encourage sparsity **indirectly** by adding a regularization term. However, we can also define a regression problem using an **explicit sparsity constraint**:

where  **represents the number of features we aim to select** in the model.

(1). What does mean? [2 marks]

Hint: The **"norm"** does not measure magnitude but rather a specific property of .

The **"norm"** actually **counts the number of nonzero elements** in (2 mark).

(2). Why do we still need Orthogonal Matching Pursuit (OMP) if we already have an explicit sparsity constraint? [2 marks]

Optimization problems with **"norm"** constraints are NP-hard because they require evaluating all possible feature combinations:

* **The main issue**: Optimization problems with  **"norm" constraints** are **impossible to exhaust/non-convex/NP-Hard**. (2 mark)
* **Why?** Finding the optimal solution requires checking **all possible feature combinations**, making it computationally **infeasible** for large datasets.
* **OMP provides an efficient alternative** by iteratively selecting the most relevant features in a greedy manner.(2 mark)

Maximum mark is 2 for (2), answering either of these criteria would be fine.

**Q12 (10 Marks)**

* + - 1. **Please draw the RBF kernel function** with the x-axis as Euclidean distance between two points D(x, x’) and y-axis as the kernel value for the three values of [0.1,1,10]. The plot need not be drawn to scale, and you should only highlight the differences. [2.5 marks]
      2. **Please draw the Laplacian kernel function** with the x-axis as Euclidean distance between two points D(x, x’) and y-axis as the kernel value for the three values of [0.1,1,10]. The plot need not be drawn to scale, and you should only highlight the differences. [2.5 marks]
      3. Please list the differences of RBF kernel and Laplacian kernel (with regards to both classification and regression) when it comes to sensitivity to outliers, smoothness of decision boundaries (regressed functions), and bandwidth parameters etc. [5 marks]

Answer：

1&2: check the figures below. The draft needs not to be accurate. (2.5 marks each)

3.

A) RBF Kernel (RK) uses L2 distance and penalizes larger difference more heavily, thus is more sensitive to outliers. Laplacian Kernel (LK) uses L1 distance and gives lower penalty to larger difference, thus is less sensitive to outliers. (2 marks)

B) RBF usually produce smooth, continuous decision boundaries (regression functions), while LK produce less smooth boundaries (functions). (1 mark)

C) Both RBF and Laplacian’s bandwidth parameters control the “bump” in the functions. A small parameter leads to wider bump, thus more global behavior, while a large parameter leads to more fine-grained structures. In Laplacian kernel, this scaling phenomenon is less pronounced. (2 marks)

A graph of a line graph

AI-generated content may be incorrect.A graph of a line graph

AI-generated content may be incorrect.