CITY UNIVERSITY OF HONG KONG

Course code & title	:	CS5489 Machine Learning: Algorithms & Applications
Session	:	Midterm, Semester A 2023
Time allowed	:	Two hours (Oct 24th, 1:00pm-2:50pm)

This question paper has 10 pages (including this cover page).

---

1. This paper consists of 13 questions.
2. Answer ALL questions.
3. Write your answers in this question paper.

---

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

Candidates are allowed to use the following materials/aids:

One A4 page (single-sided only) of handwritten notes with physical pen or pencil. Digital version or print of digital version is not allowed.

Materials/aids other than those stated above are not permitted. Candidates will be subject to disciplinary action if any unauthorized materials or aids are found on them.

Student EID: _____SOLUTIONS_____

Student ID: _____

Seat Number: _____

Multiple Choice/Selection Questions (30 points)
5 marks each. *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 accurately describe the relationship between the Bayesian decision rule and Naïve Bayes (NB) classifiers? (select all that apply)
A) Naïve Bayes classifiers are a specific implementation of the Bayesian decision rule.
B) For a binary classification task with feature $\mathbf{x}$ and $y \in \{0,1\}$ as the class labels, according to Bayesian decision rule, if $p(\mathbf{x}|y=0)>p(\mathbf{x}|y=1)$, then choose Class 0.
C) Naïve Bayes classifiers assume each feature dimension is modeled independently.
D) The Bayesian decision rule is more computationally efficient than Naïve Bayes classifiers.
E) The Bayesian decision rule is only applicable to binary classification tasks.

Answer: <A,C>

Q2. Which statements are correct about the logistic regression? (select all that apply)
A) It estimates the probability of an input belonging to a particular class by dividing the likelihood of the class by the prior probability of the input.
B) The sigmoid function is commonly used in logistic regression to squeeze the output between 0 and 1.
C) In a "One-vs-Rest" (OVR) strategy for multiclass classification with three classes, only two decision boundaries are required to classify all three classes.
D) The output of logistic regression can be interpreted as a direct probability.
E) It is guaranteed that logistic regression will always converge to a unique solution for any dataset.

Answer: <B, D>

Q3. Which of the following are characteristics of Support Vector Machines (SVMs)? (select all that apply)
A) SVMs can work on non-separable data by adding slack variables.
B) SVMs find the hyperplane that maximizes the margin between classes.
C) SVMs are sensitive to outliers in the training data.
D) SVMs are unsuitable for high-dimensional data.
E) SVMs use k-means clustering to separate data.

Answer: <A,B>

Q4. Which statements about kernel support vector machines (SVMs) are correct? (select all that apply)

A) Support Vector Machines (SVM) can directly handle non-linearly separable data.
B) Radial basis functions are a commonly used type of kernel function.
C) Kernel methods can transform non-linear problem into a more efficient problem in the feature space.
D) The dual problem for SVM mainly involves selecting support vectors.
E) Using kernels can reduce the memory and computation requirements via explicit feature transformation and inner product calculations.

Answer: <B,D>

Q5. Which of the following is/are true about Random Forest and Boosting ensemble methods? (select all that apply)

A) An individual tree in the random forest is built on a subset of the features.
B) Random forests use learning rate as of one of its hyperparameters.
C) Both methods can be used for the regression task.
D) Random Forest is only used for regression whereas gradient boosting is only used for classification.
E) None of the above.

Answer: <A, C>

Q6. Which of the following statements is correct about linear regression and feature selection? (select all that apply)?

A) Feature selection can be achieved by encouraging some linear weights to go to zero.
B) Ridge regression is effective for feature selection because it uses L2 norm to regularize the weights.
C) Orthogonal matching pursuit (OMP) can find the global optimal set of features for a given sparsity level.
D) The same set of features will always give the same set of linear weights, regardless of the features selection method used with linear regression.
E) L1 regularization is good at encouraging sparse weights because of the “corners” in its contours.

Answer: <A, E>

Discussion Questions (70 marks)
10 marks each question.

Q7. Consider the Naïve Bayes Gaussian classifier with feature vector $\mathbf{x} \in \mathbb{R}^d$ and binary class label $y \in \{0,1\}$.

(a) What are the assumptions of this classification model?
(b) What are the probability distributions in this model, and what form do they have?
(c) Given a feature vector **x**, what is the rule for performing classification with this model?

a) [3 marks] assumptions (any 3):
- there are 2 classes, and each class has a prior probability of occurring.
- for each class, a class conditional density of the features x is assumed to be Gaussian.
- the features are statistically independent, so that the Gaussian has diagonal covariance matrix.
- the samples are independent and identically distributed.

b) [4 marks] probability distributions:
- prior probability p(y) is a Bernoulli distribution, where p(y=1) = pi, p(y=0) = 1-pi.
- class conditional density: p(x|y) = Normal(x| mu, diag(s)), where diag(s) is a diagonal matrix. Or equivalently p(x|y) is a product of univariate Gaussians, each with its own mean and variance parameters.

c) [3 marks] classification:
- calculate the posterior probability p(y|x)=p(x|y)p(y)/p(x) using Bayes rule. Select the class with larger posterior. I.e., choose class 0 if p(y=0|x) > p(y=1|x) and vice versa. Equivalently, log p(y=0|x) > log p(y=1|x)
log p(x|y=0)+log p(y=0) > log p(x|y=1) + log p(y=1)

Q8. For binary classification, the logistic regression model is trained using the following objective function:

$$(\mathbf{w}^*, b^*) = \underset{\mathbf{w},b}{\operatorname{argmin}}\ \alpha \mathbf{w}^T \mathbf{w} + \sum_{i=1}^{N} \log\left(1 + \exp\left(-y_i f(\mathbf{x}_i)\right)\right)$$

(a) Explain the purpose of each term in the objective function.
(b) Discuss the meaning of the hyperparameter α in logistic regression. How does varying α influence the values of weights $\mathbf{w}$ and the performance of the model.

a)

[2 marks] The first term is the Regularization Term. It penalizes large weights to prevent overfitting. It helps in ensuring that the model generalizes well on unseen data.

[3 marks] The second term is the data-fit term. The Log-Loss (or Cross-Entropy) Term, quantifies how well the predicted probability $f(\mathbf{x}_i)$ aligns with the true label y_i. Specifically, if f(x) and y are the same sign (correct classification), then the loss is close to 0. When f(x) and y are different signs (misclassification), then the loss is large.

b)

[1 mark] Because there is an assumption that $\mathbf{w}$ follows a Gaussian distribution, the α can be seen an inversed scaled variance corresponding to the Gaussian distribution.

[2 marks] Larger α applies more penalty on large $\mathbf{w}$, pushing it towards smaller values. The model becomes simpler and might underfit the data if α is too high.

[2 marks] Smaller α applies less penalty on large $\mathbf{w}$, allows the weights $\mathbf{w}$ to take larger values. The model becomes more complex and might overfit the data if α is too low.

Q9. Explain the concept of the "kernel trick" in Support Vector Machines (SVMs) and provide an intuitive example (e.g., using a figure).

[3 marks] SVM can be applied to non-linearly separable data by transforming the feature vector $\mathbf{x}$ into a high-dimensional space using phi($\mathbf{x}$), and then applying SVM in the high-dim space.

[3 marks] In the SVM algorithm, it can be written to use only inner products of feature vectors, i.e., $\mathbf{x}_i^T \mathbf{x}_j$ or phi($\mathbf{x}_i$)^T phi($\mathbf{x}_j$). Thus, the kernel trick is to replace the inner product with a kernel function $k(\mathbf{x}_i,\mathbf{x}_j)$ = phi($\mathbf{x}_i$)^T phi($\mathbf{x}_j$), turning the linear SVM into a non-linear SVM.

[4 marks] showing an intuitive example or figure.

Q10. You are working for a hospital to build a classifier as a cost-effective screening test for a lung disease based on the patient's demographic data and saliva sample. If the classifier predicts positive, then the patient will do a follow-up CT scan that is more costly and very accurate. You have collected a dataset of 1000 patients, of which 50 have the lung disease. What are the key issues to consider when training and testing your classifier? How do you address these key issues during training and testing?

- [4 marks; 2 marks each] There are 2 issues:
  - 1) the data unbalance problem of the classifier (more negative examples vs positive examples)
  - 2) the classifier imbalance problem – as a screening test, we don't want to miss any positive samples. That is errors on the positive class (false negatives) are undesirable.
- [6 marks; 3 marks each] How to address:
  - 1) increase the weight on the positive class during training.
  - 2) increase the weight on the positive class during training AND adjust the classification threshold during test time.

Q11. You are working on an online self-driving system that employs the Adaboost model for real-time decision making. The Adaboost model will be updated with newly collected data in the real-world while the system is running, which requires the model to converge fast.
(a) How would you adjust the model or hyperparameters to make this real-time system?
(b) Is using Adaboost a good choice in this situation? Why or why not?

(a) [5 marks] use a relatively larger "learning rate" and fewer weak learners (i.e., fewer boosting iterations) for faster convergence. Fewer weak learners also means it will run faster.

(b) [5 marks] No, there are several disadvantages:
- Adaboost is sensitive to outliers (due to the exponential loss), so it could be easily overfit to outliers that might appear in the real world (e.g., rare or unexpected occurences).
- The model will grow continuously as more and more weak learners are added when new data is collected, and thus the whole model becomes slower and slower.

Q12. You are working for a bike-sharing company, and your job is to predict the number of bicycles that will be borrowed from a popular bike-sharing station each day. This prediction will be used to inform the logistics team about how many bikes should be sent to the station. The CEO tells you that it is okay to have too many bikes sent to the station, but it is not good if the station runs out of bikes, since the customers will use other bike-sharing companies.

You have a training dataset $(\mathbf{x}_i, y_i)$, i=1…N, of historical data that contains feature vectors $\mathbf{x}_i$, and number of bikes borrowed y_i. Consider a linear regression function $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x}+b$, which is trained with following optimization problem:

$$(\hat{\mathbf{w}}, \hat{b}) = \underset{\mathbf{w},b}{\operatorname{argmin}} \sum_{i=1}^{N} L(f(\boldsymbol{x}_i), y_i)$$

Design a loss function that can help you to do the regression according to the requirements, and draw a plot. Explain how your loss achieves the desired requirements.

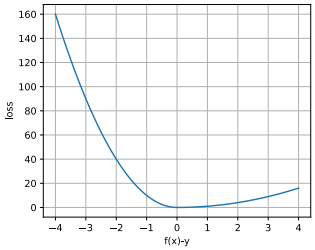
[4 marks] The loss should be asymmetrical – for the same error $|f(x_i)-y_i|$ predicting $f(x_i)>y_i$ (overestimation) should have less loss than predicting $f(x_i)<y_i$ (underestimation). This can encourage overestimation of f(x), while discouraging underestimation.

[4 marks] The loss can be achieved by breaking the loss into two parts, $f(x_i)-y_i > 0$ (overestimation) and $f(x_i)-y_i < 0$ (underestimation), and applying different loss functions. For example:

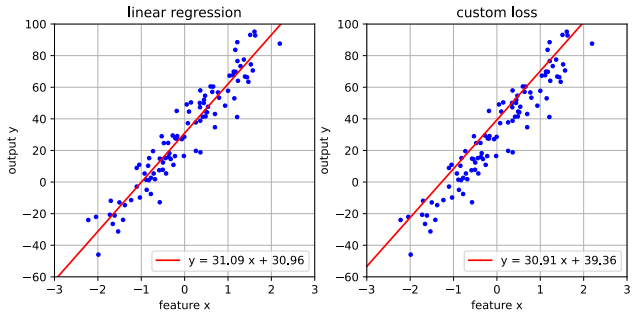
$$L(f(x_i), y_i) = \begin{cases} a * (f(x_i) - y_i)^2, f(x_i) > y_i \\ b * (f(x_i) - y_i)^2, f(x_i) < y_i \end{cases}$$

where a < b will ensure less loss for overestimation, and more loss for underestimation.

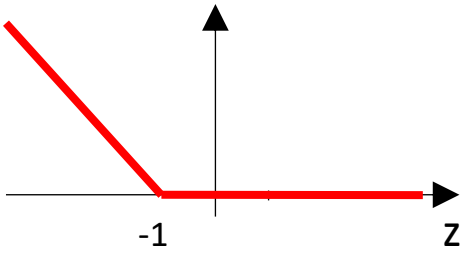
[2 marks] Here is a plot of the loss:



[0 marks, students don't need to show this plot] and an example of using it for regression:



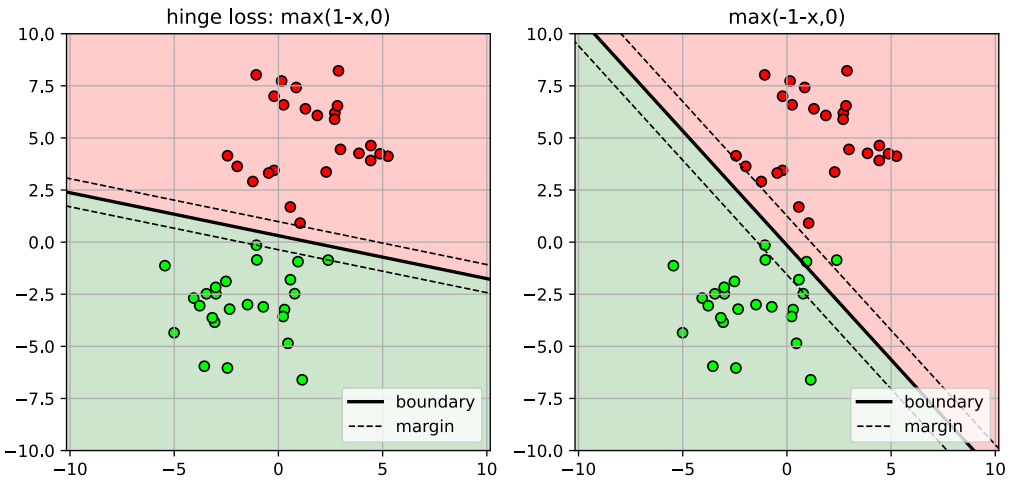
Q13. Consider the following classification loss function. Describe the properties of the classifier learned with this loss function. Will this be a good classifier? Why or why not? Draw a figure of an example of a linear classifier trained with this loss. Note: z = y*f(x).



[2 marks] It is not a good classifier.

[4 marks] The loss is 0 when for z>-1, which means that some samples may be misclassified (when -1<z<0) but have zero loss. After the sample is brought to z=-1, then it will not be considered anymore by the loss. Thus, the decision boundary will probably have points that are misclassified, and at least one on the opposite margin. Therefore it is not a good classifier.

[4 marks] example figure. There should be some points misclassified on the opposite margin. In the figure below, the left plot shows the classifier using the normal SVM hinge loss, and right plot shows the classifier using the above loss. In the right plot, there is one green point nearly on the opposite margin, which is allowed by the loss function.



--- END ---