CS3244 Machine Learning Midterm Sample Answers

Please note that due to both question and option randomization done automatically by LumiNUS, your personal version of the exam will differ in ordering than that listed here.

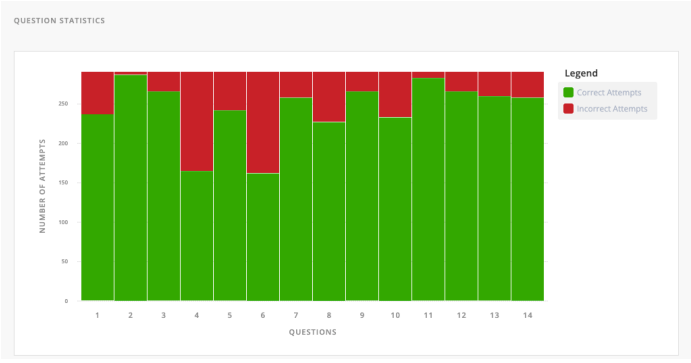
The midterm was intentionally set a bit easier than the past year midterms due to the mental fatigue and overhead we have observed with students coping with the changes in learning due to COVID-19.

We set questions calibrated to 1 mark per minute, intentionally setting sufficient time for reading and sense-making of the questions.

The grades for your midterm will be released this week. We have concluded our plagiarism checks and our review of the proctoring recordings and are finalising our review of your justification responses in the cases where those may alter the automated markings of the questions.

The histogram of current preliminary marks are as follows:

We also give the histogram of which questions were considered difficult. We see that questions 1, 4, 5, 6, 8 and 10 were more difficult for learners.



For those that performed below your expectation on the midterms, we encourage you to reach out to your tutorial group leaders and the lecturers for help. In particular, please note that our Ask Me Anything sessions are open during normal lecture slots and that are teaching assistants are self-organizing basic programming-oriented workshops to help you with the practical aspects of utilizing machine learning for your project work.

Our staff are very willing to engage you and help, but we need to know that you need help and are willing to receive it.

Section 1: Machine Learning Theory (34 marks)

Q1) [3 Marks] T/F: The goal of supervised learning is to learn a hypothesis function that best fits the training data.

Answer: False.
Explanation: The goal of supervised learning is for the hypothesis function to best approximate the target function. Fitting the training data does not necessarily mean approximating the target function. Often, fitting the data too much would lead to overfitting, leading to bad approximation of the target function (the hypothetical ground truth function).

Q2) [4 Marks] MRQ: Which of the following problem(s) is/are sensible to use machine learning to solve?

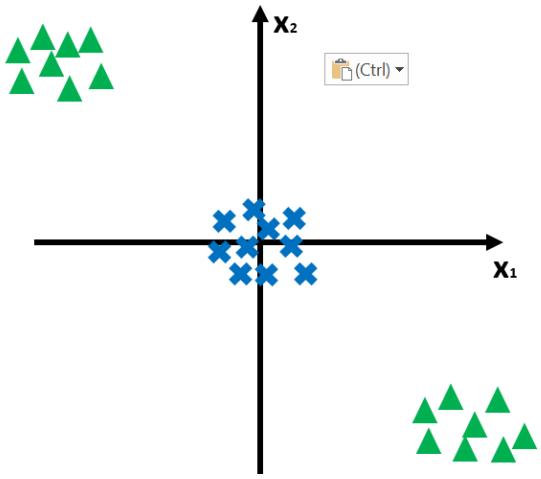
a) Modelling the relationship between force and acceleration (F=ma)
b) Create a random number generator
c) Converting speech to text
d) Facial recognition

Answer: c, d.
Explanation:
a) False. The formula is well-known, so machine learning is not needed to estimate it again.
b) False. Machine learning cannot model randomness (no inductive bias).
c) True. Converting speech to text is a sophisticated mapping function that machine learning is good to solve.
d) True. Facial recognition also requires fitting a sophisticated function that machine learning is well-suited to estimate.

Q3) [4 Marks] T/F: Is it true that after one step of Stochastic Gradient Descent, the overall training loss function will always decrease?

Answer: False.
Explanation: In SGD, we train the model one example at a time. One step of SGD would decrease the loss for that one training example, which would not necessarily decrease the loss of the whole training dataset.

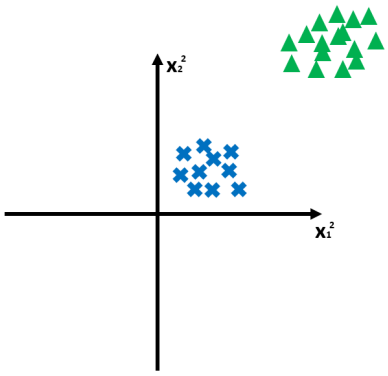
Q4) [8 Marks] MRQ: Which set of features (feature vector) can result in zero training loss on the following training examples when using a linear model? (N.B.: a bias feature is not implicitly included)



a) (x_1^2, x_2^2)
b) $(x_1^2, x_2, 1)$
c) $(x_1 * x_2, 1)$
d) $(x_1^2 * x_2, 1)$

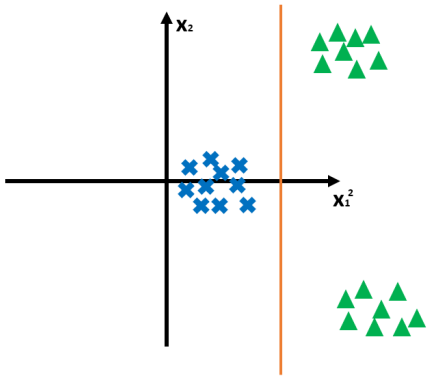
Answer: b, c.
Explanation:

a) After we project the new features, we obtain this:



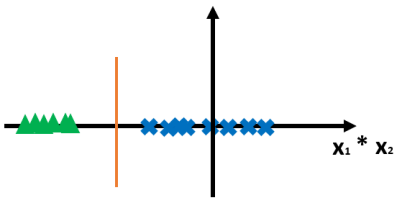
All the x_1^2 and x_2^2 values are positive
Since we do not have a bias term, we will not be able to separate the data.

b) After we project the new features, we obtain this:



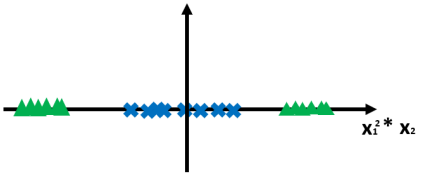
All the x_1^2 values are positive.
The decision boundary would just be a vertical line.

c) After we project the new features, we obtain this:



The green points on the top left have large positive x_2 values and large negative x_1 values, giving a large negative x_1*x_2 value.
The green points on the bottom right have large negative x_2 values and large positive x_1 values, giving a large negative x_1*x_2 value.
The green points will have highly negative x_1*x_2 values, allowing us to separate the green and the blue class.

d) After we project the new features, we obtain this:



The green points on the top left would have positive x_1^2 values and positive x_2 values, leading to a large positive $x_1^2 * x_2$ value.
The green points on the bottom right would have positive x_1^2 values and negative x_2 values, leading to a large negative $x_1^2 * x_2$ value.
We need two lines to describe the decision boundary, but a linear model can only have one straight line. Therefore, this is impossible.

Q5) [3 Marks] T/F Question: An intern trains a linear regression model (with two learnable parameters) on historical sales data to forecast future sales for her company. This will likely cause a high bias error.

Answer: True.

Explanation: In competitive business, sales are typically determined by many factors, e.g., the business condition, policy, management, sale momentum. A model with just two parameters will only consider 1 or 2 features (1 feature, if there is a bias term), hence it is very simple. Indeed, it will likely be too simple to model the real world complexities and will suffer from high bias error, i.e., underfit the target function.

Q6) [4 Marks] T/F Question: We want to estimate a target polynomial function of degree 3. Model A is a polynomial function of degree 3. Model B is polynomial function of degree 10.

Is it true that Model B has a lower bias error than model A?

Answer: False.
Explanation: Bias = stochastic noise + deterministic noise.
Both models have the same stochastic noise, as this is a property of the dataset.
Both models have the deterministic noise of 0. This is because both models are complex enough (degree $\not\geq$ 3) to model the target function.

Q7) [3 Marks] T/F Question: We want to model a target polynomial function of degree 10 using 20 data points. We fit a polynomial function of degree 2.

This model has low variance and high bias error.

Answer: True.
Explanation: This model has low variance because we are using 20 datapoints to fit a simple polynomial function of degree 2. Random samples of 20 datapoints would not lead to high variance in the polynomial function obtained. This model has a high bias error because a polynomial function of degree 2 is far too simple and not complex enough to model a polynomial function of degree 10.

Q8) [5 Marks] MCQ Question: How much longer would it take to run the training for k-fold cross validation using 5-folds compared to 10-folds?

(Do not include time taken for validation; just account for training time. Assume all training is done for the same number of epochs.)

a) 5-fold validation will take 2 times longer
b) 10-fold validation will take 2 times longer
c) 5-fold validation will take 2.25 times longer
d) 10-fold validation will take 2.25 times longer

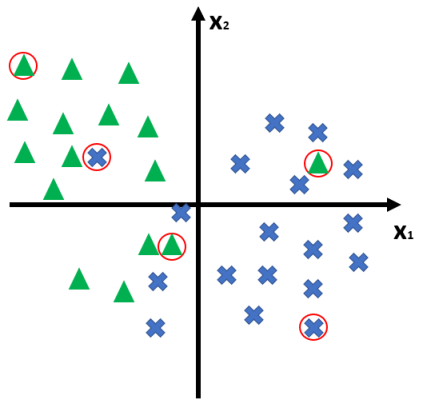
Answer: d.
Explanation: For 5-fold validation, training is done on 80% of the data 5 times.
For 10-fold validation, training is done on 90% of the data 10 times.
The ratio of training time is:

$$\frac{\text{10-fold}}{\text{5-fold}} = \frac{90\% \times 10 \text{ folds}}{80\% \times 5 \text{ folds}} = \frac{9}{4} = 2.25$$

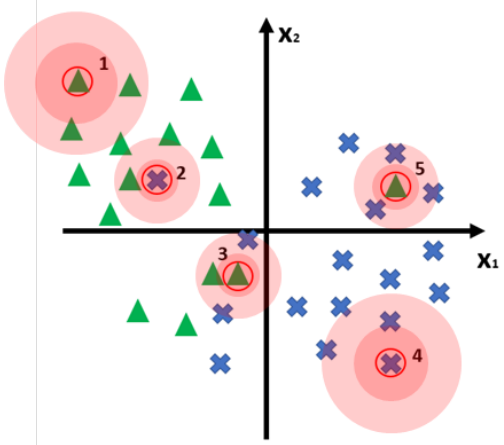
Section 2: Machine Learning Application (26 Marks)

Q9) [5 Marks] T/F: We want to determine the optimal value of *k* for k Nearest Neighbors by choosing the value that minimizes the error on the validation set. The points circled in red are the validation datapoints in this 2-dimensional set.

Is it true that for this dataset, *k*=1 has a lower validation error than *k*=3?



Answer: True.
Explanation:



We have annotated two shaded circles for each validation instance. The inner shaded circle shows the k=1 selections and outer circle shows k=3 selections. The Table below shows the prediction and correctness for each validation instance, for k=1 and k=3.

Instance	Prediction Label k=1	Prediction Label k=3	Actual Label	Correctness k=1	Correctness k=3
1	Green	Green	Green	1	1
2	Green	Green	Blue	0	0
3	Green	Blue	Green	1	0
4	Blue	Blue	Blue	1	1
5	Blue	Blue	Green	0	0
All				**3**	**2**

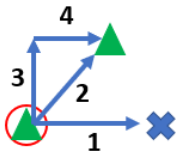
k=1 results in more correct prediction labels than k=3, so k=1 has lower validation error (**true**).

Q10) [5 Marks] MCQ: Which of the following statements is true regarding kNN?

a) The distance metric that we choose does not affect the performance of kNN
b) Setting k=1 gives us a test accuracy of 100%
c) If we set k as an odd number, it is still possible to obtain a tie in votes
d) kNN cannot be used for regression

Answer: c.
Explanation:

a) **False.** Note that performance = model performance (e.g., accuracy).



Proof by counter example. Consider the classification correctness for the point circled in red (in the diagram above). Both Euclidean and Manhattan distances lead to different kNN prediction results.

- Euclidean distance: the point would be correctly labelled as green, since Distance 2 < Distance 1.
- Manhattan distance: the point would be wrongly labelled as blue, since Distance 1 < (Distance 3 + Distance 4).

Alternative interpretation: performance = time performance.
Euclidean distance ($d = \sqrt{x^2 + y^2}$) requires calculating squares and square root, so it has more steps than Manhattan distance ($d = x + y$), and has lower performance.

b) **False.**

In this example, the test datapoint circled in green is misclassified when k=1.

c) **True.**
Proof by counter example. Consider k=3, if we also have 3 labels and each of 3 neighbours are different labels, then we have equal voting and thus a tie among 3.
In general, k = multiples of the number of classes can lead to ties.

d) **False.**
kNN just requires aggregating from neighbours, and can be used for regression by taking the average labels of neighbours.

Q11) [3 Marks] T/F: Is it true that pruning decision trees helps to decrease overfitting?

Answer: True.

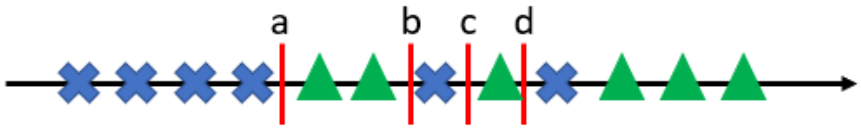
Explanation: Decision tree pruning reduces the complexity of the model. The model makes fewer assumptions about the data, is less likely to learn rare patterns in the training dataset that do not exist in the test dataset, and thus less prone to overfitting.

Q12) [3 Marks] T/F: Is it true that pruning decision trees generally increases training loss?

Answer: True.
Explanation: When we prune decision trees, we are reducing the complexity of the model and decrease its ability to fit the training dataset. This will lead to an increase in training loss.

Q13) [7 Marks] MCQ: For the 1-dimensional dataset below, which threshold should we pick to split the dataset for the decision tree node to maximize information gain?

a) a
b) b
c) c
d) d



Answer: a.
Explanation:
Let S be the set of the datapoints before the splitting.

Reference equations for entropy $H(S)$, remainder $H(S|x_i)$ and Information Gain IG:

$$H(S) = -\sum_{c=0}^{C} P_c \log_2 P_c$$

$$H(S|x_i) = \sum_{j=0}^{C_i} \frac{|S_j|}{|S|} H(S_j)$$

$$IG = H(S) - H(S|a)$$

For the example above,

$$P_{\times} = \frac{6}{12} = 0.5, \qquad P_{\blacktriangle} = \frac{6}{12} = 0.5$$

$$H(S) = H(P_{\times}, P_{\blacktriangle}) = -(P_{\times} \log_2 P_{\times} + P_{\blacktriangle} \log_2 P_{\blacktriangle}) = -\log_2 0.5 = \log_2 2 = 1$$

For each split position, the information gain is:

a)

$$\begin{aligned} IG &= H(S) - H(S|a) \\ &= 1 - \left[\frac{4}{12}H\left(\frac{4}{4},\frac{0}{4}\right) + \frac{8}{12}H\left(\frac{2}{8},\frac{6}{8}\right)\right] \\ &= 1 - \left[\frac{4}{12}\left(-\frac{4}{4}\log_2\frac{4}{4} - \frac{0}{4}\log_2\frac{0}{4}\right) + \frac{8}{12}\left(-\frac{2}{8}\log_2\frac{2}{8} - \frac{6}{8}\log_2\frac{6}{8}\right)\right] \\ &= 0.459 \end{aligned}$$

b)

$$\begin{aligned} IG &= 1 - \left[\frac{6}{12}H\left(\frac{4}{6},\frac{2}{6}\right) + \frac{6}{12}H\left(\frac{2}{6},\frac{4}{6}\right)\right] \\ &= 1 - \left[\frac{6}{12}\left(-\frac{4}{6}\log_2\frac{4}{6} - \frac{2}{6}\log_2\frac{2}{6}\right) + \frac{6}{12}\left(-\frac{2}{6}\log_2\frac{2}{6} - \frac{4}{6}\log_2\frac{4}{6}\right)\right] \\ &= 0.02 \end{aligned}$$

c)

$$\begin{aligned} IG &= 1 - \left[\frac{7}{12}H\left(\frac{5}{7},\frac{2}{7}\right) + \frac{5}{12}H\left(\frac{1}{5},\frac{4}{5}\right)\right] \\ &= 1 - \left[\frac{7}{12}\left(-\frac{5}{7}\log_2\frac{5}{7} - \frac{2}{7}\log_2\frac{2}{7}\right) + \frac{5}{12}\left(-\frac{1}{5}\log_2\frac{1}{5} - \frac{4}{5}\log_2\frac{4}{5}\right)\right] \\ &= 0.196 \end{aligned}$$

d)

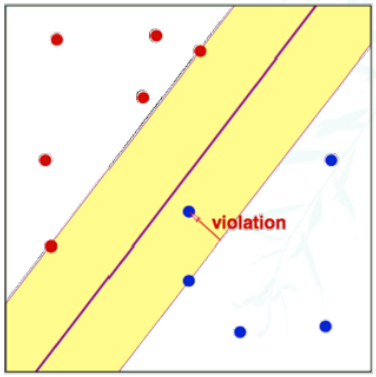
$$\begin{aligned} IG &= 1 - \left[\frac{8}{12}H\left(\frac{5}{8},\frac{3}{8}\right) + \frac{4}{12}H\left(\frac{1}{4},\frac{3}{4}\right)\right] \\ &= 1 - \left[\frac{8}{12}\left(-\frac{5}{8}\log_2\frac{5}{8} - \frac{3}{8}\log_2\frac{3}{8}\right) + \frac{4}{12}\left(-\frac{1}{4}\log_2\frac{1}{4} - \frac{3}{4}\log_2\frac{3}{4}\right)\right] \\ &= 0.094 \end{aligned}$$

IG for (a) is highest, so this is the best split position.

Q14) [3 Marks] T/F Question: Is it true that for soft SVM, the margin is the distance from the decision boundary to the closest correctly classified point?

Answer: False.

Explanation: For soft SVM, the closest correctly classified point might not define the margin because soft SVM allows for violations of the margins.



--- End ---