MIDTERM EXAM SOLUTIONS

CS 178, SPRING 2023

May 5th 2023

YOUR STUDENT ID:

YOUR NAME:

YOUR SEAT NUMBER:

Instructions

- Please write your student ID, full name, and seat number in the spaces provided above.
- DO NOT TURN OVER THE PAGE TO START UNTIL INSTRUCTED TO DO SO.
- Put away everything (including any electronic devices, written materials, etc) except for a pen or pencil.
- If you think there is a typo in a question, please read the question carefully at least twice. If you still think there is a typo please raise your hand and we will respond.
- If you need to use a rest-room please raise your hand and let one of the TAs know.
- The exam will last for 50 minutes. When instructed that the exam is over, put down your pen/pencil and stop writing. Remain in your seat and we will collect exams row by row.
- If you finish within 40 minutes of the exam starting you can leave your seat and drop your exam off at the front of the lecture hall. Otherwise please wait in your seat until we collect the exams.

Feel free to use this page for your rough work: it will be ignored during grading

1 True or False Questions

Answer true or false for each of the following statements:

1. The learning rate λ in the gradient descent algorithm can be any number:
   False: λ must be positive

2. An advantage of the kNN classifier over the nearest-centroid (NC) classifier is that kNN is much faster than NC at prediction time:
   False: NC is $O(Cd)$ whereas kNN is $O(nd + nk)$, and typically $C \ll n$.

3. For a logistic classifier, the time complexity of making a prediction on a single new datapoint is $O(d)$, where d is the number of features in the data:
   True

4. One way of preventing overfitting when using the nearest centroid algorithm is to regularize the model:
   False: The nearest centroid algorithm isn't fit by minimizing a loss, so we can't regularize.

5. Suppose for a classification problem that the number of training datapoints n is greater than the number of classes C. At prediction time, the nearest centroids algorithm has a greater space complexity than the kNN algorithm on this problem:
   False: Nearest centroids is $O(Cd)$ whereas kNN is $O(nd)$.

6. Using additional features in a classification problem will always result in better performance of the learned model:
   False: distance based classifiers can suffer from worse performance if we increase the number of features.

7. The gradient descent algorithm is guaranteed to always find the global minimum of any loss function:
   False: gradient descent can get stuck in local minima for non-convex loss functions.

8. Smaller values of k in the kNN algorithm will cause the model to be more likely to overfit the data:
   True: extreme case is $k = 1$ where the model achieves perfect accuracy on the training data.

9. In general, we should use the accuracy of the model on the data it was trained on in order to select the best hyperparameters for our model:
   False: you should look at the performance on some held-out data.

10. When using the $L2$ regularization function in linear regression, we are more likely to learn a model where many of the weights are zero:
    False: this is true for the $L1$ regularization function.

2 Regression

1. Consider the one-dimensional regression training dataset given in Table 1. We would like to evaluate the performance of the linear regression model on this dataset, where we will measure performance using the mean-squared-error (MSE).

   As we saw in class, one way of assessing model performance on a small dataset is by using cross validation. What is the leave-one-out cross validation MSE of a linear regression model on this data? That is, perform cross-validation for the linear regression model on this dataset with three folds using MSE as the evaluation metric. You do not need to simplify any numbers in your answer.

x	y
1	1
3	3
5	10

   Table 1:

   To solve this problem, you should fit three different linear models on each of the three folds in this dataset. The resulting linear model then should be used to make a prediction for the held-out datapoint, and this prediction compared with the true y-value for the held-out datapoint using the MSE.

   (a) First fold: train on $x = 1$, $y = 1$ and $x = 3$, $y = 3$. The line of best fit is $y = x$. For the held-out datapoint, the MSE is $(5 - 10)^2 = 25$.

   (b) Second fold: train on $x = 3, y = 3$ and $x = 5, y = 10$. Line of best fit is $y = \frac{7}{2}x - \frac{15}{2}$. MSE on held-out point is $(-4 - 1)^2 = 25$.

   (c) Third fold: train on $x = 1, y = 1$ and $x = 5, y = 10$. Line of best fit is $y = \frac{9}{4}x - \frac{5}{4}$. MSE on held-out point is $(\frac{9}{4}(3) - \frac{5}{4} - 3)^2 = 25/4$.

   The cross-validation MSE is then the average of these three numbers.

   Solution: $\frac{1}{3}(25 + 25 + \frac{25}{4})$

Problem 2 (Continued)

2. Suppose we are doing regression on a two dimensional dataset with feature vectors $\mathbf{x} = (x_1, x_2)$. Write an expression for the degree three polynomial feature expansion of $\mathbf{x}$, which we denote by $\mathbf{z}$. In addition, write an expression for the corresponding degree three polynomial regression model $f(\mathbf{x} \mid \theta)$. Use θ's to denote the weights of this model.

$$\mathbf{z} = (1, x_1, x_2, x_1x_2, x_1^2, x_2^2, x_1x_2^2, x_2^2x_1, x_1^3, x_2^3)$$

$$f(\mathbf{x} \mid \theta) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1 x_2 + \theta_4 x_1^2 + \theta_5 x_2^2 + \theta_6 x_1^2 x_2 + \theta_7 x_1 x_2^2 + \theta_8 x_1^3 + \theta_9 x_2^3$$

3 kNN Classifier

Consider a binary classification problem with two class labels y taking values 0 or 1, where $\mathbf{x} = (x_1, \ldots, x_d)$ is a d-dimensional feature vector. Assume in this problem that, unless otherwise stated, that we will use absolute distance in our kNN algorithm, defined between two vectors $\mathbf{x}$ and $\mathbf{z}$ as:

$$dist(\mathbf{x}, \mathbf{z}) = \sum_{j=1}^{d} |x_j - z_j|$$

1. Suppose we have a 2-dimensional classification problem with the following training data:

$\mathbf{x}$	y
(1, -1)	0
(2, 3)	0
(5, -1)	1
(3, 4)	0
(5, 2)	1
(1, 0)	0
(6, 6)	1
(4, 2)	1

Consider test data point $\mathbf{x} = (3, 3)$. For each of (a) $k = 3$, (b) $k = 5$: provide the values of the nearest neighbors and provide the class label prediction from kNN:

(a) $k = 3$:
NEIGHBORS = $(2, 3), (3, 4), (4, 2)$

CLASS LABEL PREDICTION = 0

(b) $k = 5$:
NEIGHBORS = $(2, 3), (3, 4), (4, 2), (5, 2), (1, 0)$

CLASS LABEL PREDICTION = 0

Problem 3 Continued

2. Of all the possible values for k, which value minimizes the training error on this dataset? What is the resulting training error?

   $k = 1$ results in a training error of zero.

3. Suppose that we fit two different kNN models: (A) one model to the dataset above, and (B) one model to the dataset above, but where every feature vector is scaled to be 10 times larger. For example, the first datapoint would become $(10, -10)$. Both models use the same value of k. Would the accuracy of model (A) be higher, the same as, or lower, than that of model (B)? Assume the data each model is being tested on is scaled in the same way as the data used to train the model. Justify your answer in one or two sentences.

   Model (A) and model (B) will have the same accuracies – in fact, they will make exactly the same predictions. This is because, in model (B), all of the distances computed will be scaled by a factor of ten. However, this won't affect which datapoints are closest to a particular test point.

4 Classification Accuracy

Consider the confusion matrix in Table 2 below. Assume that these numbers are the results from using a particular classifier to make predictions on a particular test data set. The rows represent the true labels y and the columns represent the predicted labels $\hat{y}$ from the classifier on the test data.

	$\hat{y} = 1$	$\hat{y} = 2$	$\hat{y} = 3$
$y = 1$	30	10	2
$y = 2$	3	25	5
$y = 3$	6	4	15

Table 2:

1. What is the classification accuracy of this classifier? Write your answer as a percentage.

   ACCURACY = 70$\%$

2. What is the most likely class in the test data (according to the true labels)?

   MOST LIKELY CLASS = 1

3. Provide a reason (in one sentence) why classification accuracy is not used to define loss functions for gradient descent.

   Classification accuracy does not have useful gradient information for gradient descent: it is a piecewise constant function as a function of the parameters $\boldsymbol{\theta}$, with the gradient being zero in most of the parameter space. **Additional explanation:** For example, consider a binary classifier for a 1-dimensional feature space x with a single threshold parameter θ (some value on the real-line), such that we predict one of the classes if x is above the threshold θ and the other if x is less than θ. As we move in parameter space θ, along the real-line, the classification accuracy on any training data set (as a loss function) will remain constant between data points and "jump" when we cross-over a datapoint. Thus, the gradients (here just a single partial derivative with respect to the single parameter θ) does not provide useful directional information (on whether to increase or decrease θ) since the gradient is always either 0 (in the "flat parts") or infinite (at the jumps).

Problem 4 Continued

Consider now a different confusion matrix given in Table 3, where y can take values $0, 1$, and the rows correspond to true labels y and the columns correspond to predicted labels $\hat{y}$. Suppose that this binary classification problem corresponds to classifying emails as spam $(y = 1)$ or not spam $(y = 0)$.

	$\hat{y} = 0$	$\hat{y} = 1$
$y = 0$	6	3
$y = 1$	10	20

Table 3:

4. Compute the precision and recall for the confusion matrix in Table 3.

   PRECISION = $\frac{20}{20+3}$

   RECALL = $\frac{20}{20+10}$

5. For the purposes of spam detection, it is important that our classifier does not mistakenly label emails which are not spam $(y = 0)$ as spam $(y = 1)$. In this case, would precision or recall be a better evaluation metric for our classifier?

   You should use the precision. The precision measures how accurate the model is when the model predicts $\hat{y} = 1$. We want our spam detector to have a high precision: it should be highly accurate whenever it labels an email as spam.

5 Nearest Centroid

Consider a binary classification problem with two class labels y taking values 0 or 1, where $\mathbf{x} = (x_1, \ldots, x_d)$ is a d-dimensional feature vector. In this problem, we will use the Euclidean distance, where the Euclidean distance between two d-dimensional vectors $\mathbf{x}$ and $\mathbf{z}$ is defined as:

$$\text{dist}(\mathbf{x}, \mathbf{z}) = \sqrt{\sum_{j=1}^{d} (x_j - z_j)^2}$$

Answer the following questions:

1. Say we have been provided with the following training data:

x	y
(1, 4)	0
(4, 1)	1
(2, 3)	0
(8, 7)	1
(3, 2)	0
(6, 10)	1

What are the centroids for each class? Your answers should be in the form of two vectors μ_0 and μ_1 for class 0 and 1 respectively.

$\mu_0 \quad = (2, 3)$

$\mu_1 \quad = (6, 6)$

2. What is the Euclidean distance of test datapoint $\mathbf{x} = (4, 5)$ to each centroid in the dataset above?

DISTANCE $0 = \sqrt{8}$ DISTANCE $1 = \sqrt{5}$

3. What is the class label predicted by the NC classifier for $\mathbf{x} = (4, 5)$ in the dataset above?

LABEL = 1

Problem 4 Continued

4. Your friend has trained a NC model and a logistic classifier on a dataset, and gets an accuracy of 73% with the NC model and an accuracy of 92% with the logistic classifier. Both of these accuracies are on the training set. Your friend says that this means they should use the logistic classifier for their task. In one sentence, say whether you agree or disagree with your friend, and why.

   You should disagree with your friend. They have only evaluated the performance of their models on the *training* set. You should only make decisions about which model is likely to outperform the other on unseen data using a validation set.

5. Suppose we have a binary classification problem, with two-dimensional feature vectors $\mathbf{x} = (x_1, x_2)$, and that we have fit a nearest centroid classifier with centroid $\boldsymbol{\mu}_0 = (\mu_{01}, \mu_{02})$ for the class $y = 0$ and centroid $\boldsymbol{\mu}_1 = (\mu_{11}, \mu_{12})$ for the class $y = 1$. Derive an expression for the decision boundary of this classifier. You should simplify your answer as much as possible.

   See Lecture 7, slide 20

6 Gradient Descent

Assume in the problem below that we are are working with a d-dimensional feature vector $\mathbf{x}$. Assume we are minimizing some loss function $L(\boldsymbol{\theta})$ (e.g. MSE or cross-entropy) with some model (e.g. linear regression or a logistic model). For the purposes of this question, it does not matter which loss function and which classifier we are using.

1. Define an equation for the gradient update (in the gradient descent algorithm) in terms of the current parameter vector $\boldsymbol{\theta}^{current}$, the new parameter vector $\boldsymbol{\theta}^{new}$, the learning rate λ, and the gradient vector $\nabla L(\boldsymbol{\theta}^{current})$.

   $\boldsymbol{\theta}^{new} \quad = \boldsymbol{\theta}^{current} - \lambda \nabla L(\boldsymbol{\theta}^{current})$

2. Say we are doing gradient descent and $\boldsymbol{\theta}^{current} = (2, -3, 8)$, and $\nabla L(\boldsymbol{\theta}^{current}) = (10, 30, 20)$. For $\lambda = 0.1$, provide the values for the new parameter vector $\boldsymbol{\theta}^{new}$.

   $\boldsymbol{\theta}^{new} \quad = (1, -6, 6)$

3. State briefly (a) one advantage, and (b) one disadvantage of making the learning rate λ very small when running a gradient descent algorithm.

   ADVANTAGE: We are guaranteed theoretically to converge to a local minimum AND/OR we can very close to an actual local minimum on our final steps

   DISADVANTAGE: A very small learning rate will be computationally slow, i.e. it may take many steps for gradient descent to converge.