CS 221, Fall 2020
Quiz 2

You have 1 hour to complete the quiz. You are allowed to consult course notes and books but no communication or general internet access is allowed. Good luck!

Stanford University Honor Code

The Honor Code is the University's statement on academic integrity written by students in 1921. It articulates University expectations of students and faculty in establishing and maintaining the highest standards in academic work:

- The Honor Code is an undertaking of the students, individually and collectively:
  - that they will not give or receive aid in examinations; that they will not give or receive unpermitted aid in class work, in the preparation of reports, or in any other work that is to be used by the instructor as the basis of grading;
  - that they will do their share and take an active part in seeing to it that others as well as themselves uphold the spirit and letter of the Honor Code.
- The faculty on its part manifests its confidence in the honor of its students by refraining from proctoring examinations and from taking unusual and unreasonable precautions to prevent the forms of dishonesty mentioned above. The faculty will also avoid, as far as practicable, academic procedures that create temptations to violate the Honor Code.
- While the faculty alone has the right and obligation to set academic requirements, the students and faculty will work together to establish optimal conditions for honorable academic work.

Question	**Score**
1	/ 9
2	/ 6
Total score:	**/ 22**

Multiple Choice

1. **[0 points]** You are allowed to consult online lecture notes, personal notes, and use a basic calculator during this quiz. You may not use the internet to access any websites other than Gradescope or the course website. You may not communicate with others to give or receive aid during or after the quiz. **I agree to abide by these rules and the Stanford Honor Code.**

   (a) Agree

   (b) Disagree

   Answer: (a)

2. **[1 point]** Which of the following are examples of **regression**? Select all that apply.

   (a) Using data on house prices in an area to predict how much a newly constructed house in the same area will sell for.

   (b) Using examples of spam and non-spam emails to determine whether a new email is spam or not.

   (c) Using image data to determine which of the following a given image depicts: an apple, a pear, an orange, or a tomato.

   (d) Using past data on stock prices to predict what the price of a certain stock will be tomorrow.

   Answer: (a) and (d) are regression ((b) and (c) are classification).

3. **[2 points]** For a given training dataset, let the *minimum training loss* denote the minimum over the training losses of predictors in the hypothesis class under consideration. Which of the following will **never increase** the minimum training loss? Select all that apply.

   (a) Adding features

   (b) Removing features

   (c) Making the hypothesis class larger

   (d) Making the hypothesis class smaller

   (e) Collecting a separate validation set and tuning hyperparameters using the new validation set

   Answer: (a) and (c). More features makes it easier to fit the data. More hypotheses makes it easier to fit the data.

4. **[1 point]** Are your answers for the above question guaranteed to also decrease test error?

   (a) Yes

   (b) No

   Answer: No. We can't make any guarantees about our test error before actually computing it.

5. **[2 points]** Neha and Juan have each been tasked with developing a machine learning model for classifying Facebook posts as either human-generated or machine-generated. Neha and Juan work independently, and have access to the same data set $\mathcal{D}$ to use for training, and the same data set $\mathcal{D}_{\text{test}}$ to use for testing. However, they decide to use $\mathcal{D}$ a bit differently.

   Neha decides to split $\mathcal{D}$ into two parts, one larger set $\mathcal{D}_{\text{train}}$ consisting of 80% of the data (chosen randomly), and a smaller validation set $\mathcal{D}_{\text{val}}$ consisting of the remaining 20%. She trains her model on $\mathcal{D}_{\text{train}}$ only, but measures her performance on $\mathcal{D}_{\text{val}}$. She tunes the hyperparameters of her model to minimize the error she gets on $\mathcal{D}_{\text{val}}$. After some time, she is able to get $\textbf{ValLoss}_{\text{Neha}}$ on $\mathcal{D}_{\text{val}}$. She then runs her model on a the test set $\mathcal{D}_{\text{test}}$ and obtains $\textbf{TestLoss}_{\text{Neha}}$.

   Juan develops his model by training on all of $\mathcal{D}$. He tunes the hyperparameters of his model to minimize the error he gets on $\mathcal{D}$. After some time, he is able to get his training error on $\mathcal{D}$ down to $\textbf{TrainLoss}_{\text{Juan}}$. He then runs his model on the test set $\mathcal{D}_{\text{test}}$ and obtains $\textbf{TestLoss}_{\text{Juan}}$.

   (a) Would you expect $\textbf{ValLoss}_{\text{Neha}}$ to be smaller or greater than $\textbf{TrainLoss}_{\text{Juan}}$?

   (b) Would you expect $\textbf{TestLoss}_{\text{Neha}}$ to be smaller or greater than $\textbf{TestLoss}_{\text{Juan}}$?

   I (a) $\textbf{ValLoss}_{\text{Neha}} \leq \textbf{TrainLoss}_{\text{Juan}}$ (b) $\textbf{TestLoss}_{\text{Neha}} \leq \textbf{TestLoss}_{\text{Juan}}$

   II (a) $\textbf{ValLoss}_{\text{Neha}} \leq \textbf{TrainLoss}_{\text{Juan}}$ (b) $\textbf{TestLoss}_{\text{Neha}} \geq \textbf{TestLoss}_{\text{Juan}}$

   III (a) $\textbf{ValLoss}_{\text{Neha}} \geq \textbf{TrainLoss}_{\text{Juan}}$ (b) $\textbf{TestLoss}_{\text{Neha}} \leq \textbf{TestLoss}_{\text{Juan}}$

   IV (a) $\textbf{ValLoss}_{\text{Neha}} \geq \textbf{TrainLoss}_{\text{Juan}}$ (b) $\textbf{TestLoss}_{\text{Neha}} \geq \textbf{TestLoss}_{\text{Juan}}$

   Answer: The answer is III. We would expect $\textbf{TrainLoss}_{\text{Juan}}$ to be smaller than $\textbf{ValLoss}_{\text{Neha}}$. Since Juan is trying to minimize the error he gets on the same data he also trains on, he will be able to achieve near-perfect accuracy. We would expect $\textbf{TestLoss}_{\text{Juan}}$ to be greater than $\textbf{TestLoss}_{\text{Neha}}$. Neha uses a validation set to ensure that she does not overfit to the training data. As a result, her model will likely generalize better than Juan's, and perform better on the test set.

6. **[3 points]** Consider least-squares linear regression using a two-layer neural network with the ReLU activation function, i.e., $\text{ReLU}(z) = \max(z, 0)$, and just one hidden unit. The input is $x \in \mathbb{R}$, and the feature vector is just the input squared: $\phi(x) = x^2$. The prediction is thus $f_{v,w}(x) = w \cdot \max(v \cdot \phi(x), 0)$, where $w \in \mathbb{R}$ and $v \in \mathbb{R}$ are parameters that we wish to learn. Suppose Bob, who did not take CS221, initialized $v = -1$ and $w = 1$ and trained the network using gradient descent with step size $\eta = 0.1$ for 5 steps.

   To Bob's surprise, nothing happened: the outputs of the neural network did not change at all from their initial values after the 5 steps. Why was this the case?

   (a) The step size was too small; we should increase the step size and run for more iterations.

   (b) Gradient descent only works on linear classifiers; we should use backpropagation instead.

   (c) The gradient is 0 with this initialization; we should change the initialization.

   (d) Running gradient descent on neural networks with only one hidden unit is unstable when the input is a quadratic function; we should use at least two hidden units.

   Answer: The answer is (c). With this initialization, the ReLU activation function will always yield zero, so the gradient will always be zero as well.

Short Answer

7. **[3 points] K-means clustering:**

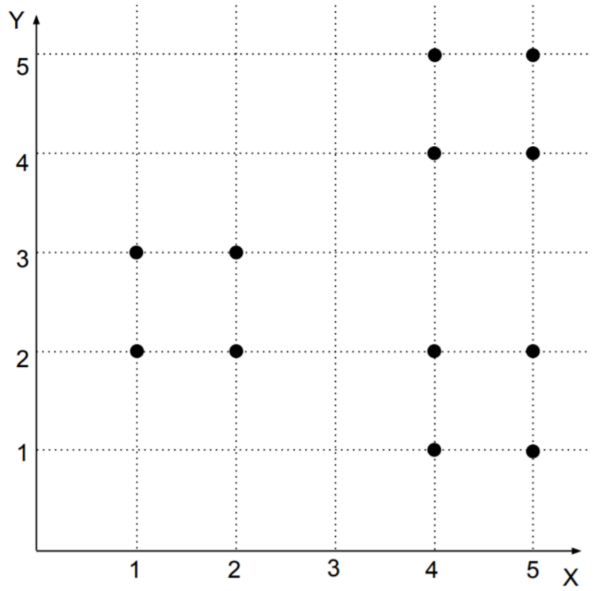


Figure 1: Points to be clustered.

Suppose we have 12 points shown in Figure 1. Recall that the k-means algorithm tries to minimize the reconstruction loss, alternating between optimizing over the cluster centroids and optimizing over the cluster assignments. When we optimize over the assignments, suppose we **break ties by assigning points to the cluster with the lower index** (e.g. we assign points to centroid μ_1 rather than centroid μ_2 if the distances to both centroids are equal).

Suppose we initialize k-means with the following cluster centroids :

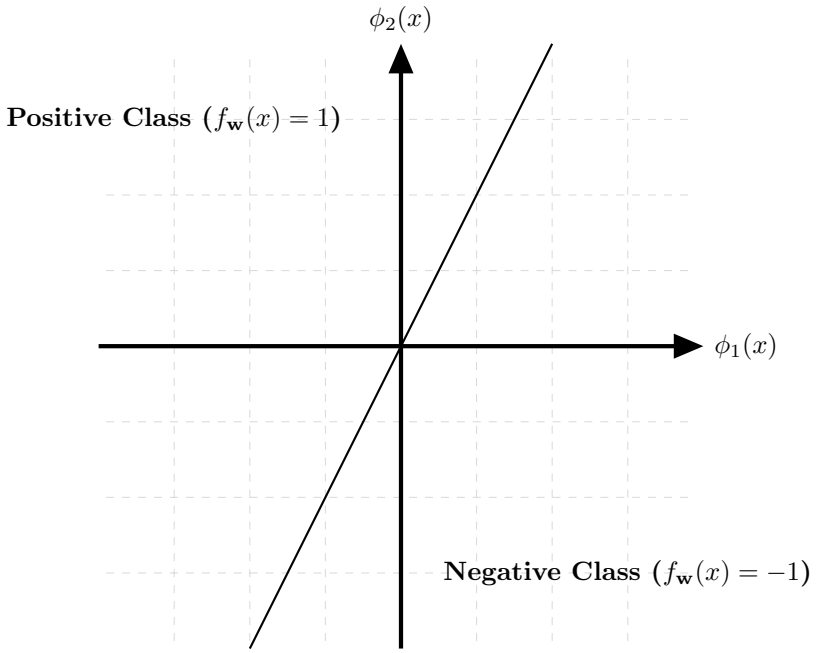
$$\mu_1 = (5, 2) \qquad \mu_2 = (5, 4) \qquad \mu_3 = (5, 5)$$

Here, $\mu_1 = (5, 2)$ means that μ_1 is at $x = 5$, $y = 2$ in Figure 1. Now, run k-means until convergence. What will the final cluster centroids be? You might find it easier to go through the k-means algorithm visually rather than grinding it out numerically. **Keep in mind the tie-breaking mechanism as defined above**.

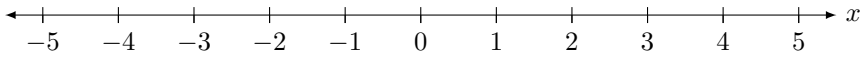
Answer: Initially, we assign points $\{(5, 4), (4, 4)\}$ to μ_2, $\{(5, 5), (4, 5)\}$ to μ_3, and the rest to μ_1. This gives us new centroids $(3, 2), (4.5, 4), (4.5, 5)$, which cause convergence.

8. **[3 points] Non-linear decision boundaries:** Consider the binary classifier $f_{\mathbf{w}}(x) = \text{sign}(\mathbf{w} \cdot \phi(x))$ that operates on 2-dimensional feature vectors $\phi(x) \in \mathbb{R}^2$. Suppose that $x \in \mathbb{R}$, $\phi : x \mapsto [x, x^2]$, and $\mathbf{w} = [-2, 1]$.

   In Quiz 1, we asked you to draw a decision boundary (for a different classifier) in **feature space**. For this classifier given above, the decision boundary in feature space is:



   This week, we'd like you to draw the decision boundary for this classifier in the **raw input space**. On the number line below, indicate the region(s) of the number line corresponding to **positive** ($f_{\mathbf{w}}(x) = 1$) predictions. **Pay attention to minus signs in this problem and double check your work.**



   Answer: