

CSE 446/546 Winter 2026 Final Exam

March 18, 2026

Name _____

UW NetID (not the numbers) _____

Assigned seat _____

- Please **wait** to open the exam until you are instructed to begin.
- Please put away phones and any other electronic devices.
- Please take out your Husky Card and have it accessible when you turn in your exam.
- If you complete the exam before the last 5 minutes, you may turn in your exam to course staff. Otherwise, please remain in your seats for the last 5 minutes of the exam.
- If you need to use the restroom, please first leave your exam with an instructor at the front of the room. You can then collect and resume your exam when you return.

This exam consists of a set of short questions (True/False, multiple choice, short answer).

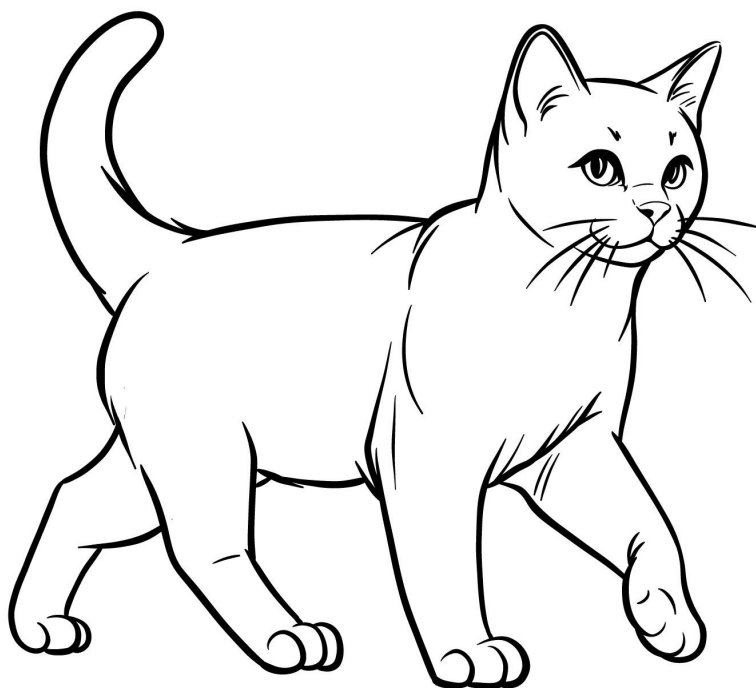
- For multiple choice questions, fill in the circle(s) to indicate your answers. Do not write your answer to the side. Example:

☒ a Selected answer.

☐ b Unselected answer.

- Multiple choice questions marked with should only be marked with one answer. All other multiple choice questions are , in which case any number of answers may be selected (**including none, one, or more**).
- For questions, we will give partial credit.
- For each short answer question, please write your answer in the provided space.
- If you need to change an answer or run out of space, clearly indicate what you would like graded as your final answer. If we cannot determine your final answer, we cannot give credit.

This image is included only to cover the back of this page. It has no relation to the exam.



1. 1.5 points Select All That Apply

You are designing a machine learning pipeline to predict housing prices. You are considering using a polynomial regression model and need to select the optimal polynomial degree p and evaluate the model's performance. Which of the following statements about model selection and evaluation are TRUE?

- (a) The validation set is used to evaluate different candidate models and select the best polynomial degree p , while the test set is used strictly to assess the performance of the final chosen model.
- (b) 10-fold cross-validation is generally faster to compute than Leave-one-out (LOO) cross-validation, but it provides a more pessimistically biased estimate of the model's error.
- (c) If you want to select a subset of the most relevant features (e.g., top 50 out of 10,000) to speed up training, it is acceptable to calculate the correlation between the features and the target prices using the entire dataset before splitting it into train and test sets.
- (d) When performing k -fold cross-validation, the standard procedure is to divide your entire dataset into k parts, using $k - 1$ parts to train and the remaining 1 part as your final test set to report out-of-sample accuracy.
- (e) To ensure your model generalizes well, you should always select the polynomial degree p that achieves the absolute minimum Mean Squared Error (MSE) on the training dataset.

Solution: Correct choices: A, B. (A) True: The validation set is used to evaluate model candidates and pick hyperparameters (like degree p), while the test set is held out completely to evaluate the final model. (B) True: k -fold is computationally faster than LOO (which requires training n times). However, because k -fold uses a smaller fraction of the data for training in each fold compared to LOO, its error estimate is more pessimistically biased. (C) False: This is known as the "Feature Selection Trap". Selecting features based on correlations computed over the entire dataset before splitting causes information from the test set to leak into the training process. (D) False: Cross-validation should only be performed on the training split. You must split the data into Train and Test first, run k -fold CV strictly within the Train set for model selection, and evaluate the final model on the untouched Test set. (E) False: Training error monotonically decreases as model complexity (degree p) increases, which eventually leads to overfitting. You should select the degree that minimizes validation/test error, not training error.

2. 1 points

You want to design a linear regression pipeline without data leakage. The pipeline includes feature selection based on correlation with the target y . Consider the following steps:

- A Select the 50 features with the largest absolute correlation with y using the training set

only.

- B** Train a linear regression model on the training set using the selected features.
- C** Split the 500 samples into an 80% training set and a 20% test set.
- D** Compute the correlation between each feature and y on the training set only.
- E** Evaluate the model's mean squared error (MSE) on the test set.

Write the letters A–E in the correct order to produce a valid evaluation with no data leakage.

$\rightarrow$ $\rightarrow$ $\rightarrow$ $\rightarrow$

Solution: The correct order is $C \rightarrow D \rightarrow A \rightarrow B \rightarrow E$. To prevent data leakage, the test set must be isolated before any processing.

3. 1 points

Suppose you have a dataset of size $n = 1000$. You are evaluating a specific model architecture using k -fold cross-validation. You know that for this model and dataset, providing more training data consistently improves the model's true performance.

(a) 0.5 points One Answer As you increase the number of folds k from $k = 2$ up to $k = 1000$ (Leave-One-Out Cross-Validation), which of the following trends do you expect to observe?

- ☐ (a) The expected CV error estimate will increase.
- ☐ (b) The expected CV error estimate will decrease.
- ☐ (c) The expected CV error estimate will remain the same.

(b) 0.5 points Justify your choice. Specifically, explain the mathematical relationship between k , the size of the training set in each fold, and the resulting bias of the cross-validation error estimate.

Solution:

- (a) The expected CV error estimate will decrease.
- (b) In k -fold cross-validation, the model in each fold is trained on $n \times \frac{k-1}{k}$ samples. As k increases, the training set used in each fold gets larger and closer to the full dataset size n . Since the model performs better with more data, the models in each fold will be more accurate, decreasing the average CV error. Smaller values of k yield a more pessimistically biased estimate.

4. 1 points Select All That Apply You train two models on the same supervised learning task and data:

- **Model A:** a linear model with few parameters
- **Model B:** a highly flexible neural network

Model B achieves significantly lower train and test error than Model A, but its test error is much higher than its own training error. Which statements are true?

- ☐ (a) Model B likely has lower bias than Model A.
- ☐ (b) Model B likely has higher variance than Model A.

- Ⓒ Model A must have higher irreducible error.
- Ⓓ The gap between training and test error in Model B suggests overfitting.

Solution: Correct choices: A, B, D. (A) True: a more flexible model can better fit complex functions, reducing bias. (B) True: higher model flexibility typically increases sensitivity to the training data, leading to higher variance. (C) False: irreducible error depends only on the data-generating process, not on the model choice. (D) True: a large train–test gap is a classic indicator of overfitting.

5. 2 points

For each of the following situations, is the bias or variance of the model's predictions expected to decrease? Select just one option. Circle your answer and give a brief explanation.

(a) 0.5 points Decreasing L2 regularization strength (smaller λ) in linear regression. Decreased Bias Decreased Variance Explanation:

(b) 0.5 points Switching from a single deep decision tree to bagging many deep trees. Decreased Bias Decreased Variance Explanation:

(c) 0.5 points Increasing k in a k -NN classifier. Decreased Bias Decreased Variance Explanation:

(d) 0.5 points Increasing the max depth of a decision tree. Decreased Bias Decreased Variance Explanation:

Solution:

- (a) Bias. Decreasing λ means decreasing regularization, which will increase variance and reduce bias.
- (b) Variance. Bagging many trees will average out many high-variance predictors, reducing variance.
- (c) Variance. Increasing k will cause larger neighborhoods, reducing sensitivity to noise (variance) and increasing bias.
- (d) Bias. Increasing the max depth will allow the model to overfit to the training dataset, increasing variance and decreasing bias.

6.

3 points Consider $X \in \mathbb{R}^{n \times d}$, and non-zero $w \in \mathbb{R}^d$. Suppose $y = Xw + \epsilon$ with $\epsilon \sim \mathcal{N}(0, I)$. For the following questions you may use the following generalized version of linearity of expectation: For any fixed matrices A, B and random vectors u, v of appropriate dimension:

$$\mathbb{E}[Au + Bv] = A\mathbb{E}[u] + B\mathbb{E}[v]$$

(a) 1.5 points Let

$$\hat{w} = \arg \min_{w \in \mathbb{R}^d} \|Xw - y\|_2^2,$$

and assume $\hat{w}$ is unique. Show that $\hat{w}$ is an unbiased estimate of w .

(b) 1.5 points Let $\lambda > 0$ and

$$\hat{w} = \arg \min_{w \in \mathbb{R}^d} \|Xw - y\|_2^2 + \lambda \|w\|_2^2.$$

Show that $\hat{w}$ is a biased estimate of w .

Solution: Part (a): We have

$$\begin{aligned}
 \mathbb{E}[\hat{w}] &= \mathbb{E} \left[(X^\top X)^{-1} X^\top y \right] \\
 &= \mathbb{E} \left[(X^\top X)^{-1} X^\top (Xw + \epsilon) \right] \\
 &= (X^\top X)^{-1} X^\top Xw + (X^\top X)^{-1} X^\top \mathbb{E}[\epsilon] && \text{Linearity of expectation} \\
 &= (X^\top X)^{-1} X^\top Xw && \mathbb{E}[\epsilon] = 0 \\
 &= w
 \end{aligned}$$

Part (b): We have

$$\begin{aligned}
 \mathbb{E}[\hat{w}] &= \mathbb{E} \left[(X^\top X + \lambda I)^{-1} X^\top y \right] \\
 &= \mathbb{E} \left[(X^\top X + \lambda I)^{-1} X^\top (Xw + \epsilon) \right] \\
 &= (X^\top X + \lambda I)^{-1} X^\top Xw + (X^\top X + \lambda I)^{-1} X^\top \mathbb{E}[\epsilon] && \text{Linearity of expectation} \\
 &= (X^\top X + \lambda I)^{-1} X^\top Xw && \mathbb{E}[\epsilon] = 0 \\
 &\neq w && \lambda > 0 \implies X^\top X + \lambda I \neq X^\top X
 \end{aligned}$$

7. 1 points Select All That Apply Suppose you are training a linear regression model. In which of the following scenarios would applying ℓ_2 (Ridge) regularization potentially be helpful?

- ☒ **a** The number of features exceeds the number of training examples ($d > n$).
- ☐ **b** You want the final model to contain many coefficients that are exactly zero.
- ☒ **c** The labels in your training set have high measurement noise.
- ☐ **d** You want the model to perfectly fit the training data.

Solution: Correct answers:

- The matrix $X^T X$ is non-invertible due to $d > n$.
- The labels in your training set contain measurement noise or random errors, and you wish to prevent the weights from fitting these fluctuations.

Explanation of Choices:

- **Correct ($d > n$):** When $d > n$, $X^T X$ becomes rank-deficient and singular. ℓ_2 regularization adds λI , making $(X^T X + \lambda I)$ positive definite and invertible, ensuring a unique solution.
- **Incorrect (Exact Zeros):** Producing coefficients that are exactly zero is a property of ℓ_1 (Lasso) regularization. Ridge shrinks coefficients but typically does not set them exactly to zero.
- **Correct (Noisy Labels):** When labels contain measurement error or random fluctuations, OLS may overfit these irregularities. ℓ_2 regularization penalizes large weights, reducing variance and improving generalization.
- **Incorrect (Perfect Fit):** Ridge regularization discourages overly complex models by penalizing large weights, which typically prevents the model from perfectly fitting the training data.

8. 1 points Select All That Apply

Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be differentiable and let ∇f denote its gradient. Which of the following conditions are sufficient to guarantee that f is convex? In other words, if a differentiable function f satisfies one of these conditions, must f be convex? Treat each of (a)-(d) as independent questions given this prompt.

- ☒ **a** $\nabla f(x) = 0$ for all x .
- ☒ **b** $f(y) \geq f(x) + \nabla f(x)^\top (y - x)$ for all x, y .
- ☐ **c** $\nabla^2 f(x) \preceq 0$ for all x .

(d) $\|\nabla f(x)\|_2 \leq f(x)$ for all x .

Solution: Correct answers: (a) and (b).

- (a) If $\nabla f(x) = 0$ for all x , then f is constant. Constant functions are convex, so this condition is sufficient for convexity.
- (b) For a differentiable function, convexity is equivalent to the first-order (supporting hyperplane) condition

$$f(y) \geq f(x) + \nabla f(x)^\top (y - x) \quad \forall x, y \in \mathbb{R}^n.$$

- (c) $\nabla^2 f(x) \preceq 0$ corresponds to concavity (when f is twice differentiable). Convexity would require $\nabla^2 f(x) \succeq 0$.
- (d) The bound $\|\nabla f(x)\|_2 \leq f(x)$ is not a standard condition related to convexity and does not imply convexity in general.

9. 2 points

Minibatch gradient descent forms one update by averaging gradients from a minibatch of size B (i.e., B training examples). List two advantages of minibatch gradient descent compared to single-sample SGD.

Solution: Two advantages are:

- The gradients for the B examples can be computed in parallel.
- The variance of the gradient estimate is reduced by a factor of $1/B$ due to averaging.

10. 3 points

Let $f(w) = \|Xw - y\|_2^2$, where $X \in \mathbb{R}^{n \times d}$, and $y \in \mathbb{R}^n$.

(a) 1 points Derive the gradient $\nabla_w f(w)$.

(b) 1 points Use your answer from part (a) to derive the Hessian $\nabla_w^2 f(w)$.

(c) 1 points Use your answer from part (b) to show that f is convex.

Solution:

(a) $\nabla f(w) = 2X^\top Xw - 2X^\top y$.

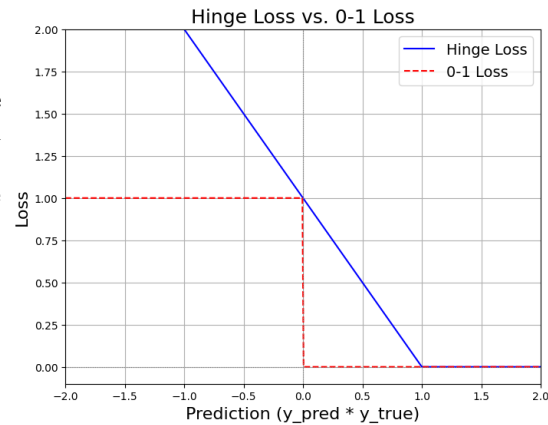
(b) $\nabla^2 f(w) = 2X^\top X$.

(c) We aim to show $2X^\top X \succeq 0$. Let $v \in \mathbb{R}^n$. We have

$$\begin{aligned} v^\top 2X^\top X v &= 2(Xv)^\top Xv \\ &= 2\|Xv\|_2^2 \\ &\geq 0. \end{aligned}$$

11. 3 points

In classification with labels $y_i \in \{-1, +1\}$, we often want to minimize the “**0-1 Loss**”, defined as $\sum_i \mathbb{I}[y_i \cdot f(x_i) \leq 0]$, where the indicator function $\mathbb{I}[\cdot]$ counts how many predictions have the wrong sign.



(a) 1 points Give two reasons why the 0-1 loss is difficult to optimize.

(b) 1 points The “**2**” is defined as: $\ell(y, f(x)) = \max(0, 1 - y \cdot f(x))$. It penalizes not only wrong predictions but also correct predictions that aren’t “confident” enough (where the margin $y \cdot f(x)$ is less than 1). The hinge loss is sometimes used as an alternative to the 0-1 loss. Give two reasons why the hinge loss may be easier to optimize than the 0-1 loss.

(c) 1 points To simplify for this problem, let’s consider the hinge function in terms of a single scalar variable z (representing the *margin*: $y \cdot f(x)$): $\ell(z) = \max(0, 1 - z)$. Give the interval of all subgradients of ℓ at $z = 1$.

Solution:

(a) The 0-1 loss is non-convex; discontinuous; no subgradients exist at the discontinuity (unlike the L_1 loss); the loss function is flat for $z < 0$ and $z > 0$, which means gradient descent cannot know the direction that will improve the loss.

(b) 3 possible reasons: Hinge loss is convex & hinge loss is sub-differentiable & non-zero gradients give better direction for learning

Common misconceptions:

- “confidence bounds” / “less over fitting” does not answer “easier to optimize”.
- “Mostly/more differentiable” was not accepted.
- “Correct predictions are also penalized” was not accepted

(c) The Hinge Loss is defined as $\ell(z) = \max(0, 1 - z)$. At $z = 1$, the function has a corner where two linear segments meet. To find the interval of all subgradients, we identify the slopes of these two segments:

- (a) To the left of $z = 1$, the slope is -1 .
- (b) To the right of $z = 1$, the slope is 0 .

Any value v that falls between these two slopes is a valid subgradient. **Interval of subgradients:** $[-1, 0]$

12. 3 points

(a) 1 points Show that $\mathbb{R}$ is a convex set.

(b) 1 points Let $c \in \mathbb{R}$. Show that $\{t \in \mathbb{R} : c \leq t\}$ is a convex set.

(c) 1 points Let $c \in \mathbb{R}$. Use part (a), part (b), and the epigraph definition of convexity to show that $f(x) = c$ is convex on $\mathbb{R}$. *Hint:* You may use that $A \times B$ is a convex set, for any convex sets $A, B \subseteq \mathbb{R}$.

Solution: Part (a)

Proof. Let $x, y \in \mathbb{R}$ and $\lambda \in [0, 1]$. Observe that $\lambda x + (1 - \lambda)y \in \mathbb{R}$. Hence $\mathbb{R}$ is a convex set. $\square$

Part (b):

Proof. Let $x, y \in \{t \in \mathbb{R} : c \leq t\}$, and $\lambda \in [0, 1]$. We have

$$\begin{aligned}\lambda x + (1 - \lambda)y &\geq \lambda c + (1 - \lambda)c \\ &= c.\end{aligned}$$

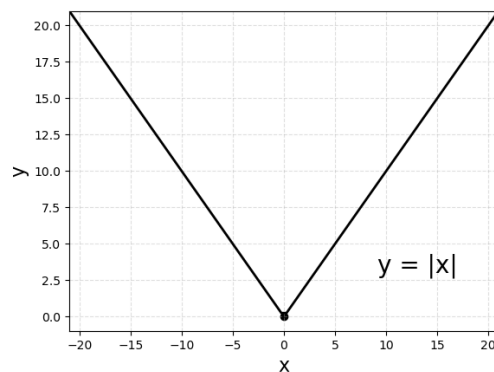
Hence $\{t \in \mathbb{R} : c \leq t\}$ is a convex set. $\square$

Part (c)

Proof. By definition of epigraph, we have the epigraph of f is $E = \{(x, t) \in \mathbb{R}^2 : f(x) \leq t\}$. By definition of f we have $E = \{(x, t) \in \mathbb{R}^2 : c \leq t\}$, which can be rewritten as $E = \mathbb{R} \times \{t \in \mathbb{R} : c \leq t\}$. $\mathbb{R}$ is convex by part (a), $\{t \in \mathbb{R} : c \leq t\}$ is convex by part (b), thus by the hint, E is convex. By the epigraph definition of convexity, f is convex. $\square$

13. 2 points

Consider the following function:



(a) 0.75 points State whether the function is convex, and explain why or why not.

(b) 0.75 points Write a piecewise function that specifies the gradient of $y = |x|$ for all x at which the gradient exists.

(c) 0.5 points What algorithm can optimize a non-differentiable function like this one?

Solution:

(a) Convex; english explanation of line test (i.e. pick any two points on y and connect them; this line will always lie on or above the function) or formal proof suffices.

(b) Gradient exists everywhere but $x = 0$:

$$\begin{cases} +1 & x > 0 \\ -1 & x < 0 \\ \text{undefined} & x = 0 \end{cases}$$

(c) Sub-gradient descent or ISTA. Note that normal GD and SGD do not work, since they are not suitable for optimizing non-differentiable functions.

14. 2 points

Suppose you have a dataset with n examples and d features, where $d \gg n$ (high-dimensional). You want to train a model $\hat{y} = f(x)$ that predicts labels y_i from features x_i , but you are worried about overfitting. So you decide to use the following procedure:

Step 1: Run a feature selection algorithm (like Lasso) on your entire dataset to find the 10 most “important” features to obtain a dataset only containing those features.

Step 2: Use bootstrapping to create 100 resampled datasets, each containing only the features identified in Step 1.

Step 3: Estimate the variance of $\hat{f}(x)$ by training a model on each bootstrapped dataset from step 2.

(a) 0.5 points

Keeping in mind that your full modeling pipeline $\hat{y} = f(x)$ includes both feature selection and predicting labels from the selected features, is the variance estimate produced in Step 3 likely to be an underestimate, overestimate, or accurate reflection of the true variance of your modeling pipeline?

(a) underestimate

(b) accurate estimate

(c) overestimate

(b) 1 points

Justify your answer above.

(c) 0.5 points

How could you change the steps above to improve your estimate?

Solution: a) It is an **underestimate**. b) By selecting the 10 "best" features using the *entire* dataset before bootstrapping, you have performed a form of Data Leakage. The 10 features were chosen because they work well for the specific points in the original dataset. When you bootstrap the dataset, you are only seeing subsets of those same points. Because you didn't include the "feature selection" step inside the bootstrap loop, your variance estimate fails to account for the uncertainty/instability of the feature selection itself. c) To improve the estimate, you should first create bootstrap-resampled datasets that contain all of the original d features, then perform feature selection separately on each of those resampled datasets, then train a model $\hat{f}_j$ on each of the feature-selected bootstrap-resampled datasets, and finally report the variability across those models. Those models now each reflect the entire modeling pipeline.

15. 1 points

Suppose you want to use the standard nonparametric bootstrap to estimate the uncertainty of a statistic computed from a dataset of size n . In each bootstrap replicate, you create a new dataset by sampling n examples *with replacement* from the observed dataset, then recompute the statistic.

(a) 0.5 points One Answer Which of the following statistics is the bootstrap generally *least reliable* for estimating a confidence interval:

- ☐ (a) Sample mean
- ☐ (b) Sample median
- ☐ (c) Sample minimum
- ☐ (d) Ridge regression weights $\hat{w}$

(b) 0.5 points Justify your answer

Solution: The sample minimum. The bootstrap is usually less reliable for **extreme order statistics** such as the minimum or maximum. Because, it resamples only reuse values already observed in the dataset, so they can never produce a value smaller than the observed minimum. As a result, bootstrap does not capture the true variability of the sample minimum well. In contrast, the sample mean, sample median, and ridge coefficient depend on many observations and are usually estimated more reliably by bootstrap.

16. 1 points Select All That Apply

Select all statements that are true about the bootstrap.

- ☐ a **Sampling without replacement may not produce valid bootstrap samples because the resulting datasets would not be i.i.d. draws from the empirical distribution.**
- ☐ b If a bootstrap dataset contains duplicate observations, this indicates an error in our bootstrap procedure since each observation should appear at most once in the bootstrapped dataset.
- ☐ c **Bootstrapping is used when constructing random forests. Specifically, each tree in the random forests is trained on a different bootstrapped sample of the training data.**
- ☐ d **Bootstrapping always assumes that the empirical distribution of the observed sample is a reasonable approximation of the true distribution.**

Solution: A is true, because bootstrap's guarantees depend on the samples being i.i.d., and if we were to sample without replacement, the samples would no longer be i.i.d. B is false, because since bootstrap samples with replacement, duplicate observations could happen. C is true, because random forests use bootstrapping to create a diverse set of decision trees. D is also true, since bootstrap works because of this reason; we assume that the empirical distribution is a good estimate of the true distribution.

17. 1.5 points Select All That Apply

You are using a k -Nearest Neighbors (k -NN) algorithm to predict customer behavior. Your training dataset contains two features: “Age” (ranging from 18 to 80 years) and “Annual Income” (ranging from \$20,000 to \$200,000). Which of the following statements about the k -NN algorithm and its application to this dataset are TRUE?

- (a) If you do not perform feature scaling (e.g., standardization), the “Annual Income” feature will likely dominate the Euclidean distance calculations, rendering the “Age” feature almost irrelevant.
- (b) A 1-NN classifier will always achieve exactly zero training error, provided there are no identical data points with conflicting labels in the training set.
- (c) Increasing the value of the hyperparameter k from 1 up to the total number of samples N will generally increase the model’s variance and decrease its bias.
- (d) Standard k -NN is a “lazy learning” algorithm, meaning that, for a fixed k , its computational costs are predominantly incurred when evaluating predictions rather than during training.
- (e) k -NN typically performs better in extremely high-dimensional feature spaces (e.g., $d = 10,000$) than in low-dimensional spaces because points become more easily separable.

Solution: Correct choices: A, B, D.

(A) True: Without feature scaling, the differences in income (in the tens of thousands) will dwarf the differences in age (in the tens) when computing the squared differences for Euclidean distance. (B) True: When evaluated on the training set, a 1-NN model simply finds the point itself (distance = 0) and returns its label, resulting in perfect accuracy unless there is contradictory noise (identical points with different labels). (C) False: Increasing k means the model averages over more neighbors, which smooths the decision boundary. This reduces variance but increases bias. At $k = N$, the model just predicts the majority class (highest bias, zero variance). (D) True: k -NN simply stores the training data ($O(1)$ or $O(Nd)$ effort). However, predicting a new point requires calculating the distance to all N training points, which is computationally expensive ($O(Nd)$ time). (E) False: This contradicts the “curse of dimensionality.” In high-dimensional spaces, the distance to the nearest neighbor approaches the distance to the farthest neighbor, making the concept of proximity less meaningful and typically degrading k -NN performance.

18. 1 points Select All That Apply

You bag multiple decision trees on a dataset where one feature, x_1 , is extremely predictive. This causes many trees to split on x_1 near the top of the tree. Which of the following are true statements?

- (a) Individual decision trees will have low correlation with each other.

- (b) Randomly selecting m features at each node will increase correlation between trees.
- (c) In this situation, random forest will likely generalize better than bagged decision trees.
- (d) Averaging predictors that are highly correlated reduces variance more effectively than averaging predictors that are less correlated.

Solution:

- (a) False. Since x_1 is very predictive, many trees will select x_1 for their initial split, resulting in trees that look very similar to each other.
- (b) False. This will diversify the trees, which would decrease correlation.
- (c) True. Random forest selects a subset of features at each node, which should help decrease correlation and reduce variance.
- (d) False. Highly correlated predictors make the same errors, which reduces the effectiveness of aggregation.

19. 2 points

You train a Kernel Ridge Regression model using the Radial Basis Function (RBF) kernel:

$$k(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right)$$

Assume you did not include an unregularized bias/intercept term. After training, you make a prediction $\hat{y}$ on a new test point x_{test} that is located extremely far away from all training points (i.e., $\|x_{test} - x_i\| \rightarrow \infty$ for all x_i in the training set).

(a) 0.5 points

What numeric value will your model's prediction $\hat{y}$ converge to?

Answer: _____

(b) 0.5 points

Justify your answer to Part A using the predictive equation for Kernel Ridge Regression.

(c) 1 points

The behavior of the RBF kernel depends on the bandwidth σ . If you decrease σ to an extremely small value (approaching 0), which describes the resulting model?

☐ (a) Underfitting

☐ (b) Overfitting

Explain your answer

Solution:

(a) 0

(b) The prediction rule is $\hat{y} = \sum_{i=1}^n \alpha_i k(x_{test}, x_i)$. As $\|x_{test} - x_i\|^2 \rightarrow \infty$, the kernel $\exp(-\infty) \rightarrow 0$. Therefore, the entire sum collapses to 0.

(c) (b) Overfitting. A very small σ means the "radius of influence" of each point is narrow. The model fits training points perfectly (low bias) but fails to generalize (high variance).

20. 1 points

Consider a Neural Network used for binary classification. The model has one hidden layer and is defined by the following function:

$$f(x) = \text{sign}(W_2\sigma(W_1x + b_1) + b_2)$$

where σ is the activation function, and $\text{sign}(x) = 1$ if $x > 0$, -1 if $x < 0$, and 0 if $x = 0$. If we choose a linear activation function such that $\sigma(z) = az + c$, explain why the model can only represent linear decision boundaries.

Solution: If σ is linear, $f(x)$ becomes a composition of affine transformations: $W_2(W_1x + b_1) + b_2$. This can be simplified to a single linear transformation $W_{\text{new}}x + b_{\text{new}}$. Thus, the hidden layer adds no expressive power; it's just a more expensive way to compute a linear regression/classification.

21. 0.5 points One Answer

When training a neural network with the minibatch stochastic gradient descent, which change most directly reduces the variance of the gradient estimate used in each update, without changing the model architecture or the dataset?

- (a) Increase the minibatch size B (for example, from 32 to 512).
- (b) Decrease the learning rate η (for example, from 10^{-2} to 10^{-3}).
- (c) Add ℓ_2 regularization (weight decay).
- (d) Randomly shuffle which data points appear in each minibatch at the start of each epoch.

Solution: Correct answer: Increase the minibatch size B . **Explanation:** In minibatch SGD, the update direction is computed by averaging per-example gradients over a minibatch of size B . Averaging more samples reduces the variance of this gradient estimate (typically scaling on the order of $1/B$ under weak dependence). Decreasing η scales the step size but does not change the variance of the gradient estimate itself. Adding ℓ_2 regularization modifies the objective by adding a deterministic term to the gradient (it does not reduce the sampling variance from using a minibatch). Training for more epochs changes how long optimization runs, not the per-update gradient variance.

22. 1.5 points

Fill in the blanks for the following training loop implemented using **PyTorch**. For simplicity, this is a “bare bones” implementation of a training loop (e.g., it does not calculate validation accuracy).

```
def train(
    train_loader: DataLoader,
    model: nn.Module,
    criterion: nn.Module,
    optimizer: optim.Optimizer,
    epochs: int,
) -> List[float]:
    results = []
    for epoch in range(epochs):
        total_loss = 0

        for x, label in train_loader:
            optimizer.zero_grad()

            prediction = _____ # FILL IN HERE

            loss = criterion(prediction, label)

            _____ # FILL IN HERE

            _____ # FILL IN HERE

            total_loss += loss.item() * len(x)

        results.append(total_loss / len(train_loader.dataset))

    return results
```

Solution: First blank: `model(x)`
Second blank: `loss.backward()`
Third blank: `optimizer.step()`

23. 1 points Select All That Apply

Consider a 2-layer neural network. Select all statements that are true.

- ☐ a Increasing the number of hidden units in each layer helps prevent overfitting on smaller datasets.
- ☐ b If our weights are initialized to 0, the gradient of the first-layer weights will always be 0, regardless of the error at the output.

- Ⓒ Replacing ReLU with a Sigmoid activation may increase the risk of vanishing gradients.
- Ⓓ Gradient descent is guaranteed to find the global minimum of the loss function.

Solution: A is false, because increasing the number of hidden units could cause overfitting, especially on smaller datasets. B is true, because as discussed in section 8, initializing weights to 0 would result in a lack of learning due to the weights not being updated (since the gradients will always be 0). C is also correct, because as discussed in lecture, Sigmoid has a vanishing gradient issue since the function saturates at 0 and 1, which is not an issue for ReLU. D is false, because neural network loss functions are non-convex.

24. 2 points

Suppose a new engineer is implementing the scaled dot-product attention mechanism for a standard autoregressive language model from scratch.

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^T}{\sqrt{d_k}} + M \right) V \quad (1)$$

By mistake, they implement the mask matrix M such that $M_{ij} = 0$ for all i and j , rather than setting the upper triangular elements to $-\infty$. They proceed to pre-train the model on a massive corpus of text using the standard next-token prediction objective.

- (a) 1 points What will happen to the training loss during this pre-training phase compared to a correctly implemented model? Explain why.

- (b) 1 points If you attempt to use this trained model for autoregressive generation (inference) by giving it a prompt and asking it to generate the next word, what behavior would you expect? Why?

Solution:

- (a) The training loss will drop rapidly to near zero. Because the mask is 0 everywhere, the model is allowed to "look into the future." When tasked with predicting token x_t , it will simply attend to x_t in the input sequence (which is passed in during training) and copy it perfectly, bypassing the need to actually learn language patterns.
- (b) The model will likely output complete garbage or fail to generate coherent text. During inference, future tokens do not actually exist yet. Because the model learned to rely entirely on peeking at future tokens to make its predictions during training, it will not know how to generate a prediction based solely on past context.

25. 1 points Select All That Apply

A satellite stores high-resolution Earth images, each containing N pixels. To reduce storage, PCA is applied to the mean-centered image data, and only the top k principal components are retained ($k \ll N$). Which statements are true?

- (a) PCA minimizes squared reconstruction error among all possible k -dimensional linear projections of the data.
- (b) The retained k components correspond to directions of highest variance in the data.
- (c) PCA with k principal components guarantees that reconstructed images are visually indistinguishable from the original images.
- (d) If every image is rotated by 90° and PCA is recomputed, the principal component directions in pixel coordinates will generally change.

Solution: Correct choices: A, B, D.

(A) True: PCA gives the optimal rank- k linear approximation in terms of minimizing squared reconstruction error. (B) True: The principal components are the eigenvectors of the covariance matrix, ordered by decreasing eigenvalue, so they capture the directions of greatest variance. (C) False: Small squared reconstruction error does not guarantee perceptual indistinguishability. (D) True: Rotating every image by 90° applies an orthogonal transformation to the dataset. When PCA is recomputed, the principal component directions rotate accordingly, so their coordinates relative to the original pixel basis generally change.

26. 1 points Select All That Apply

You use SVD to apply PCA, but you forget to center the data. Which of the following might occur?

- ☐ (a) The leading principal component may primarily reflect the overall mean offset of the data rather than directions of variation about the mean.
- ☐ (b) The low-dimensional reconstruction may have larger squared error than the reconstruction obtained from properly centered PCA.
- ☐ (c) Translating all data points by the same constant vector would not affect the principal components.
- ☐ (d) The principal components need not remain mutually orthogonal.

Solution: Correct choices: A, B.

(A) True: Without centering, the first component can be dominated by the mean offset rather than variation around the mean. (B) True: Standard PCA is optimal for minimizing squared reconstruction error after centering. If centering is omitted, the resulting subspace can be worse for reconstruction. (C) False: Without centering, PCA becomes sensitive to translations of the data, so shifting all points by a constant vector can change the principal components. (D) False: The principal components are still orthogonal because they come from an eigendecomposition (or SVD), regardless of whether centering is performed.

27. 1.5 points Select All That Apply

Consider Kernel PCA with n data points $x_i \in \mathbb{R}^d$, transformed data points $\phi(x_i)$, and kernel function $k(x_i, x_j) = \phi(x_i)^\top \phi(x_j)$. We define the de-meaned, transformed data matrix $\tilde{X}_\phi$ according to:

$$\mu_\phi = \frac{1}{n} \sum_{i=1}^n \phi(x_i) \quad \tilde{X}_\phi = \begin{bmatrix} (\phi(x_1) - \mu_\phi)^\top \\ (\phi(x_2) - \mu_\phi)^\top \\ \vdots \\ (\phi(x_n) - \mu_\phi)^\top \end{bmatrix}$$

Which of the following statements about Kernel PCA are TRUE?

- ☐ (a) Kernel PCA performs linear dimensionality reduction in the transformed feature space of $\phi(x)$.
- ☐ (b) Before applying Kernel PCA, the kernel matrix must be centered to account for subtracting μ_ϕ in feature space.
- ☐ (c) The Kernel PCA algorithm computes the SVD of $\tilde{X}_\phi$.

- (d) The Kernel PCA algorithm computes the eigenvectors of $\tilde{X}_\phi^\top \tilde{X}_\phi$.
- (e) The Kernel PCA algorithm computes the eigenvectors of $\tilde{X}_\phi \tilde{X}_\phi^\top$.
- (f) If the kernel is linear, $k(x_i, x_j) = x_i^\top x_j$, then Kernel PCA reduces to standard PCA.

Solution: Correct choices: A, B, E, F.

Following the notation in Lecture 14, let

$$\bar{X} = X - 1\bar{x}^\top, \quad J = I - \frac{11^\top}{n}, \quad \tilde{K} = JKJ^\top.$$

After replacing each x_i with $\phi(x_i)$, we analogously have

$$\tilde{X}_\phi = \begin{bmatrix} (\phi(x_1) - \mu_\phi)^\top \\ \vdots \\ (\phi(x_n) - \mu_\phi)^\top \end{bmatrix}, \quad \tilde{K} = \tilde{X}_\phi \tilde{X}_\phi^\top.$$

(A) TRUE. Kernel PCA is PCA applied to the transformed data $\phi(x_i)$, so it performs linear dimensionality reduction in the feature space.

(B) TRUE. Standard PCA operates on centered data. Similarly, Kernel PCA relies on centering in the transformed feature space. This is achieved by “centering” the kernel matrix:

$$\tilde{K} = JKJ^\top.$$

This is mathematically equivalent to centering the transformed features, then computing the matrix of inner products.

(C) FALSE. Kernel PCA avoids explicitly forming the transformed vectors $\phi(x_i)$. In particular, $\tilde{X}_\phi$ may have infinitely many columns if the feature space is infinite-dimensional.

(D) FALSE. Similar to option C, $\tilde{X}_\phi^\top \tilde{X}_\phi$ is the covariance matrix in feature space and may be infinite-dimensional.

(E) TRUE. Kernel PCA instead works with

$$\tilde{K} = \tilde{X}_\phi \tilde{X}_\phi^\top = U_\phi S_\phi^2 U_\phi^\top,$$

which is an $n \times n$ matrix and avoids working directly in the potentially infinite-dimensional feature space by computing the eigenvectors of $\tilde{X}_\phi \tilde{X}_\phi^\top$.

(F) TRUE. If

$$k(x_i, x_j) = x_i^\top x_j,$$

then $\phi(x) = x$, so

$$K = XX^\top \quad \text{and} \quad \tilde{K} = JXX^\top J^\top = \bar{X}\bar{X}^\top.$$

Thus Kernel PCA reduces to ordinary PCA.

28. 1 points One Answer

You've computed the covariance matrix $\Sigma \in \mathbb{R}^{5 \times 5}$ from your dataset of observed feature vectors $x_i \in \mathbb{R}^5$. Then, you computed the eigendecomposition of Σ and found eigenvalues $[50, 25, 15, 5, 5]$. When applying PCA to your dataset, what minimum number of components k explains at least 90% of the total variance?

(a) 1

(b) 2

(c) 3

(d) 4

(e) 5

Solution:

$$50 + 25 + 15 + 5 + 5 = 100$$

$$k = 1: \frac{50}{100} = 50\%$$

$$k = 2: \frac{50+25}{100} = \frac{75}{100} = 75\%$$

$$k = 3: \frac{50+25+15}{100} = \frac{90}{100} \geq 90\%$$

Thus, $k = 3$.

29. 1.5 points Select All That Apply

You are analyzing a dataset of astronomical observations, attempting to discover distinct categories of stars based on two continuous features: surface temperature and luminosity. Because you do not have ground-truth labels, you decide to cluster the data using both k -means and Gaussian Mixture Models (GMMs). Which of the following statements regarding the behavior and properties of these clustering algorithms are TRUE?

(a) If a specific category of stars forms a highly elongated, elliptical distribution in your 2D feature space, k -means may split this single category into multiple clusters, whereas a GMM with full covariance matrices can model the elliptical shape accurately.

(b) While k -means relies on an iterative two-step optimization process (assigning points, then updating centroids), a GMM optimizes its parameters in a single step using a closed-form analytical solution.

- (c) The k -means algorithm can be viewed as a specific, limiting case of a GMM where all mixture components are constrained to have the same spherical covariance matrix ($\Sigma = \sigma^2 I$), and the variance σ^2 approaches zero.
- (d) Both k -means and GMMs are guaranteed to converge to the global optimum of their respective objective functions, provided you let them run for enough iterations.
- (e) In a GMM, a data point located exactly halfway between the means of two different components will always have a 50/50 probability of belonging to either component (i.e., a responsibility of 0.5).

Solution: Correct choices: A, C.

(A) True: k -means implicitly assumes that clusters are spherical and have roughly equal variance (because it minimizes unweighted Euclidean distance). A GMM with unconstrained covariance matrices can model elliptical data distributions of varying sizes and orientations. (B) False: GMMs do not have a closed-form solution for their parameters. They are optimized using the Expectation-Maximization (EM) algorithm, which is also an iterative two-step process (E-step for responsibilities, M-step for parameter updates), highly analogous to k -means. (C) True: If you restrict a GMM so that all components have an identical, isotropic covariance matrix $\sigma^2 I$, and take the limit as $\sigma^2 \rightarrow 0$, the soft probabilistic assignments (responsibilities) collapse into hard, binary assignments. The point is assigned entirely to the closest mean, which is mathematically equivalent to the k -means objective. (D) False: Both algorithms optimize non-convex objective functions and are highly sensitive to their initializations. They are only guaranteed to converge to a local optimum, which is why practitioners often run them multiple times with different random seeds (e.g., k -means++). (E) False: The probability (responsibility) in a GMM depends on the probability density, not just the Euclidean distance to the mean. If one component has a much larger variance (spread) or a higher mixture weight (prior probability) than the other, the exact midpoint will have a higher probability of belonging to the broader or more heavily weighted component.

30. 2 points

Consider running a single iteration of the k -means algorithm (using Euclidean distance) on a dataset composed of the following 6 points in $\mathbb{R}^3$:

$$\begin{aligned}x_1 &= (0, 1, 2) \\x_2 &= (1, 2, 3) \\x_3 &= (2, 3, 4) \\x_4 &= (6, 7, 8) \\x_5 &= (7, 8, 9) \\x_6 &= (8, 9, 10)\end{aligned}$$

Suppose you choose $k = 2$ and initialize the centroids at: $c_1^{(0)} = (0, 0, 0)$; $c_2^{(0)} = (10, 10, 10)$.

(a) Assign each point x_i to the nearest centroid.

Point	Centroid
x_1	
x_2	
x_3	
x_4	
x_5	
x_6	

(b) Compute the updated centroids $c_1^{(1)}$ and $c_2^{(1)}$ after one iteration of k -means.

$c_1^{(1)}$: _____ $c_2^{(1)}$: _____

Solution:

(a)

Point	Centroid
x_1	c_1
x_2	c_1
x_3	c_1
x_4	c_2
x_5	c_2
x_6	c_2

(b)

$$c_1^{(1)} = \frac{1}{3}((0, 1, 2) + (1, 2, 3) + (2, 3, 4)) = (1, 2, 3)$$

$$c_2^{(1)} = \frac{1}{3}((6, 7, 8) + (7, 8, 9) + (8, 9, 10)) = (7, 8, 9)$$

31. 1.5 points Select All That Apply

A Gaussian Mixture Model (GMM) is trained on GPS location data from mobile phones. Which interpretations are reasonable?

- ☐ (a) Each Gaussian may correspond to a frequently visited location.
- ☐ (b) Component covariance reflects spatial spread of visits.
- ☐ (c) Mixture weights reflect probability of a user being at each location.
- ☐ (d) GMM assumes all clusters have identical variance.
- ☐ (e) GMM assigns each data point to exactly one cluster.

Solution: Correct choices: A, B, C.

(A) True: each mixture component can model a common geographic location pattern. (B) True: the covariance captures how dispersed the visits are around that location. (C) True: mixture weights represent the probability mass assigned to each location component. (D) False: GMMs allow each component to have its own covariance. (E) False: GMMs naturally provide soft (probabilistic) assignments, which are often preferred.

32. 1.5 points Select All That Apply

You collect unlabeled accelerometer data from smartwatches and cluster the data to discover activities such as walking, running, and sitting. Which statements are TRUE?

- ☐ (a) k -means may struggle if activities have different variances or elongated shapes.
- ☐ (b) A GMM can model different activity clusters with different covariance structures.
- ☐ (c) Soft assignments in GMM allow uncertain activity transitions to be represented.
- ☐ (d) k -means and GMM will always assign the same cluster labels if the number of clusters used is the same in both k -means and GMM.
- ☐ (e) Unsupervised clustering guarantees that all discovered clusters correspond to true activities.

Solution: Correct choices: A, B, C.

(A) True: k -means implicitly assumes spherical clusters with similar variance. (B) True: GMM components can have distinct covariance matrices, capturing different shapes and spreads. (C) True: GMM posterior probabilities represent uncertainty during transitions between activities. (D) False: k -means and GMM optimize different objectives and can

yield different clusterings. (E) False: unsupervised clustering does not guarantee semantic alignment with real-world activities.