CS 189/289A, Midterm

Spring 2026

Name: _____

Email: _____@berkeley.edu

Student ID: _____

Room where you're taking the exam: _____

Name and SID of the person on your right: _____

Name and SID of the person on your left: _____

Instructions:

This exam consists of **56 points** spread out over **6 questions** and the Honor Code certification. The exam must be completed in **110 minutes** unless you have accommodations supported by a DSP letter.

Note that you should **select one choice** for questions with **circular bubbles**. There is always at least one correct answer. Please **fully** shade in the circle to mark your answer. For all math questions, **please simplify your answer**. Please also **show your work** if a large box is provided. For all coding questions, you may use commas and/or one or more function calls in each blank.

You MUST write your Student ID number at the top of each page.

Honor Code [1 Pt]:

As a member of the UC Berkeley community, I act with honesty, integrity, and respect for others. I am the person whose name is on the exam, and I completed this exam in accordance with the Honor Code.

Signature: _____

This page has been intentionally left blank.

A Linear Affair **[5 Pts]**

Suppose we are given a dataset of n data points $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ where each $\mathbf{x}_i \in \mathbb{R}^d$ is a feature vector and $y_i \in \mathbb{R}$ is a continuous target. We predict the targets using a linear model:

$$f_{\mathbf{w}}(\mathbf{x}_i) = \mathbf{w}^\top \mathbf{x}_i$$

parameterized by the weight vector $\mathbf{w} \in \mathbb{R}^d$.

For matrix-vector form, we define the design matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ as having n rows $\mathbf{x}_i^\top$, and vector $\mathbf{y} \in \mathbb{R}^n$ as having entries y_i.

(a) **[5 Pts]** Select one answer for each question.

(i) Linear regression is always linear with respect to _____.

- ◯ Bias^2
- ◯ Parameters
- ◯ Variance
- ◯ Loss

(ii) If $n \gg d$ (many more data points than feature dimensions), what is the most **direct** consequence?

- ◯ Zero training error
- ◯ High model variance
- ◯ Low test error
- ◯ $\mathbf{X}^\top\mathbf{X}$ is more likely to be invertible

(iii) If $n \ll d$ (far fewer data points than feature dimensions), what is the most likely consequence?

- ◯ Zero training error
- ◯ High model bias
- ◯ Low model variance
- ◯ $\mathbf{X}^\top\mathbf{X}$ is more likely to be invertible

(iv) Consider a regularized objective with regularizer $R(\mathbf{w})$ and $\lambda > 0$:

$$\min_{\mathbf{w}} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2 + \lambda\, R(\mathbf{w})$$

Compared to $\lambda = 0$, adding regularization generally:

- ◯ Reduces model variance at the cost of introducing some bias
- ◯ Reduces both bias and variance simultaneously
- ◯ Reduces training error at the cost of higher test error
- ◯ Increases model complexity to fit the data more closely

(v) Which is always true for ridge regression with $\lambda > 0$?

- ◯ Ridge regression always achieves zero training error.
- ◯ Ridge regression guarantees a unique solution — $\mathbf{w}^{\star}_{\text{Ridge}}$.
- ◯ Ridge regression produces a solution with many coefficients exactly to zero.
- ◯ Ridge regression makes the model invariant to the scale of input features.

Solution:

(i) (i) [1 Pt] Correct answer selected

Parameters.

(ii) (ii) [1 Pt] Correct answer selected

$\mathbf{X}^\top\mathbf{X}$ is more likely to be invertible.

(iii) (iii) [1 Pt] Correct answer selected

Zero training error.

(iv) (iv) [1 Pt] Correct answer selected

Reduces model variance at the cost of introducing some bias.

(v) (v) [1 Pt] Correct answer selected

Ridge regression guarantees a unique solution.

Smooth Operator **[7 Pts]**

Suppose we are given a dataset of n data points $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ where each $\mathbf{x}_i \in \mathbb{R}^d$ is a feature vector and $y_i \in \mathbb{R}$ is a continuous target.

We predict the targets using a linear model parameterized by the weight vector — $\mathbf{w} \in \mathbb{R}^d$.

$$f_{\mathbf{w}}(\mathbf{x}_i) = \mathbf{w}^\top \mathbf{x}_i$$

For matrix-vector form, we define the design matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ and vector $\mathbf{y} \in \mathbb{R}^n$ as:

$$\mathbf{X} = \begin{bmatrix} - & \mathbf{x}_1^\top & - \\ - & \mathbf{x}_2^\top & - \\ & \vdots & \\ - & \mathbf{x}_n^\top & - \end{bmatrix} \qquad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

To encourage adjacent weights to be similar, we apply a smoothness regularizer, which penalizes the difference between consecutive weights, using a regularization hyperparameter $\lambda > 0$:

$$J(\mathbf{w}) = \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2 + \lambda\, R(\mathbf{w}) \qquad R(\mathbf{w}) = \sum_{j=1}^{d-1} (w_{j+1} - w_j)^2.$$

(a) **[2 Pts]** Define a matrix $\mathbf{D}$ (state its entries and dimensions) such that $R(\mathbf{w}) = \|\mathbf{D}\mathbf{w}\|_2^2$.

Solution:

(i) [1 Pt] Correct dimensions: $(d-1) \times d$

(ii) [1 Pt] Correct entry structure: $-1, +1$ on consecutive columns per row

Define $\mathbf{D}$ as a $(d-1) \times d$ first-order difference matrix:

$$\mathbf{D} = \begin{bmatrix} -1 & 1 & 0 & \dots & 0 & 0 \\ 0 & -1 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & -1 & 1 \end{bmatrix}$$ *— awards (i)*

Then $\mathbf{D}\mathbf{w}$ has j-th entry $(w_{j+1} - w_j)$, so:

$$R(\mathbf{w}) = \sum_{j=1}^{d-1} (w_{j+1} - w_j)^2 = \|\mathbf{D}\mathbf{w}\|_2^2 = \mathbf{w}^\top \mathbf{D}^\top \mathbf{D}\mathbf{w}$$ *— awards (ii)*

(b) **[3 Pts]** Suppose we would like to find the optimal weights — $\hat{\mathbf{w}}$ — that minimize our smoothness-regularized objective function:

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2 + \lambda\|\mathbf{D}\mathbf{w}\|_2^2$$

Determine a closed-form expression for the minimizer $\hat{\mathbf{w}}$. You may assume all necessary matrices are invertible.

Hint: If you're stuck, reconsider the ridge regression objective and its closed-form minimizer $\hat{\mathbf{w}}_{Ridge} = (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{y}$. *How might its structure help you?*

Solution:

(i) [2 Pt] Rigorous derivation OR identification of the structural analogy ($\mathbf{D}^\top\mathbf{D}$ replaces $\mathbf{I}$)

(ii) [1 Pt] Correct final answer: $(\mathbf{X}^\top\mathbf{X} + \lambda\,\mathbf{D}^\top\mathbf{D})^{-1}\mathbf{X}^\top\mathbf{y}$

From part (a), our penalty is $\lambda\|\mathbf{D}\mathbf{w}\|_2^2 = \lambda\,\mathbf{w}^\top\mathbf{D}^\top\mathbf{D}\,\mathbf{w}$. — *awards (i)*

In ridge regression, the penalty is $\lambda\|\mathbf{w}\|_2^2 = \lambda\,\mathbf{w}^\top\mathbf{I}\,\mathbf{w}$.

Our objective has the same form with $\mathbf{D}^\top\mathbf{D}$ replacing $\mathbf{I}$, so:

$$\hat{\mathbf{w}} = (\mathbf{X}^\top\mathbf{X} + \lambda\,\mathbf{D}^\top\mathbf{D})^{-1}\mathbf{X}^\top\mathbf{y}$$ — *awards (ii)*

(c) **[2 Pts]** Let $\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2 + \lambda R(\mathbf{w})$. Consider the limit $\lambda \to \infty$.

(i) **[1 Pt]** As $\lambda \to \infty$, $R(\hat{\mathbf{w}}) \to$ _____0_____

(iii) [1 Pt] $R(\hat{\mathbf{w}}) \to 0$: penalty dominates, forcing all consecutive differences to zero

(ii) **[1 Pt]** As $\lambda \to \infty$, $\hat{\mathbf{w}} \to c \cdot$ __**1 (all-ones vector)**__ for some constant c.

(iv) [1 Pt] $\hat{\mathbf{w}} \to c\,\mathbf{1}$: $\mathbf{D}\mathbf{w} = \mathbf{0}$ forces all weights equal

Chill Gaussian Question [11 Pts]

Let $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n \in \mathbb{R}^d$ be i.i.d. data points from a multivariate Gaussian

$$\mathbf{x}_i \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \qquad \text{where } \boldsymbol{\Sigma} \text{ is known, invertible, and symmetric.}$$

We place a Gaussian prior on the mean:

$$\boldsymbol{\mu} \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0) \qquad \text{where } \boldsymbol{\Sigma}_0 \text{ is known, invertible, and symmetric.}$$

The multivariate Gaussian PDF is

$$p(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right).$$

(a) **[3 Pts]** Show that the **log-likelihood** — $\ell(\boldsymbol{\mu}) = \log p(\{\mathbf{x}_i\}_{i=1}^n \mid \boldsymbol{\mu})$ — can be written as:

$$\ell(\boldsymbol{\mu}) = \text{const} - \frac{1}{2}\sum_{i=1}^n (\mathbf{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}_i - \boldsymbol{\mu})$$

Solution:

(i) [1 Pt] Writes joint likelihood as product, then log as sum: $\sum \log p(\mathbf{x}_i)$

(ii) [1 Pt] Substitutes Gaussian PDF to extract the quadratic exponent

(iii) [1 Pt] Drops $\boldsymbol{\mu}$-independent terms into const to reach target form

$$\begin{aligned}
\ell(\boldsymbol{\mu}) &= \log\left(\prod_{i=1}^n p(\mathbf{x}_i \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})\right) = \sum_{i=1}^n \log p(\mathbf{x}_i \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}) && \text{— awards (i)} \\
&= \sum_{i=1}^n \left[\log\left(\frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}}\right) - \frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}_i - \boldsymbol{\mu})\right] && \text{— awards (ii)} \\
&= \text{const} - \frac{1}{2}\sum_{i=1}^n (\mathbf{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}_i - \boldsymbol{\mu}) && \text{— awards (iii)}
\end{aligned}$$

(b) **[2 Pts]** Derive the **maximum likelihood estimator** — $\hat{\boldsymbol{\mu}}_{\text{MLE}} = \arg\max_{\boldsymbol{\mu}} \log p(\{\mathbf{x}_i\}_{i=1}^n \mid \boldsymbol{\mu})$.
Hint: $\nabla_{\mathbf{x}}(\mathbf{x}^\top \mathbf{A}\mathbf{x}) = 2\mathbf{A}\mathbf{x}$ *for symmetric* $\mathbf{A}$.

Solution: (Carry-forward: full credit possible using student's own prior answer.)

(i) [1 Pt] Takes gradient of log-likelihood w.r.t. $\boldsymbol{\mu}$ and sets to $\mathbf{0}$

(ii) [1 Pt] Solves to get $\hat{\boldsymbol{\mu}}_{\text{MLE}} = \frac{1}{n}\sum \mathbf{x}_i$

At a maximum of a differentiable function, the gradient vanishes. So we differentiate w.r.t $\boldsymbol{\mu}$ and set the gradient to zero:

$$\nabla_{\boldsymbol{\mu}} \left[-\frac{1}{2} \sum_{i=1}^{n} (\mathbf{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}) \right] = \sum_{i=1}^{n} \boldsymbol{\Sigma}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}) = \mathbf{0}. \qquad \textit{— awards (i)}$$

Factor out $\boldsymbol{\Sigma}^{-1}$:

$$\boldsymbol{\Sigma}^{-1} \left[\left(\sum_{i=1}^{n} \mathbf{x}_i \right) - n\boldsymbol{\mu} \right] = \mathbf{0}.$$

Multiply both sides on the left by $\boldsymbol{\Sigma}$ (which is invertible) to obtain $\left(\sum_{i=1}^{n} \mathbf{x}_i\right) - n\boldsymbol{\mu} = \mathbf{0}$. Solve for $\boldsymbol{\mu}$:

$$n\boldsymbol{\mu} = \sum_{i=1}^{n} \mathbf{x}_i \quad \Longrightarrow \quad \hat{\boldsymbol{\mu}}_{\text{MLE}} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i. \qquad \textit{— awards (ii)}$$

(c) **[3 Pts]** Write the **log-posterior** — $\log p(\boldsymbol{\mu} \mid \{\mathbf{x}_i\}_{i=1}^n)$ — as a function of $\boldsymbol{\mu}$.

You may omit any terms that do not depend on $\boldsymbol{\mu}$.

Solution: (Carry-forward: full credit possible using student's own prior answer.)

(i) [1 Pt] States Bayes' rule: log-posterior = log-likelihood + log-prior + const

(ii) [1 Pt] Writes log-prior as Gaussian quadratic form in $\boldsymbol{\mu}$

(iii) [1 Pt] Combines both terms correctly with right signs/coefficients

By Bayes' rule,

$$\log p(\boldsymbol{\mu} \mid \{\mathbf{x}_i\}_{i=1}^n) = \log p(\{\mathbf{x}_i\}_{i=1}^n \mid \boldsymbol{\mu}) + \log p(\boldsymbol{\mu}) + \text{const.} \qquad \textit{— awards (i)}$$

The first term is the log-likelihood from part (a); the second is the Gaussian prior log-density:

$$\log p(\boldsymbol{\mu}) = -\frac{1}{2} (\boldsymbol{\mu} - \boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}_0^{-1} (\boldsymbol{\mu} - \boldsymbol{\mu}_0) + \text{const.} \qquad \textit{— awards (ii)}$$

Substituting both:

$$\log p(\boldsymbol{\mu} \mid \{\mathbf{x}_i\}_{i=1}^n) = -\frac{1}{2}\left[\sum_{i=1}^n (\mathbf{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}_i - \boldsymbol{\mu})\right] - \frac{1}{2}(\boldsymbol{\mu} - \boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}_0^{-1}(\boldsymbol{\mu} - \boldsymbol{\mu}_0) + \text{const}$$ — *awards (iii)*

(d) **[1 Pt]** The **MAP estimator** — $\hat{\boldsymbol{\mu}}_{\text{MAP}}$ — is derived by finding the solution that solves which of the following optimization problems?

- ◯ $\hat{\boldsymbol{\mu}}_{\text{MAP}} = \arg\max_{\boldsymbol{\mu}} \sum_{i=1}^n p(\mathbf{x}_i \mid \boldsymbol{\mu}) p(\boldsymbol{\mu})$
- ◯ $\hat{\boldsymbol{\mu}}_{\text{MAP}} = \arg\max_{\boldsymbol{\mu}} p(\{\mathbf{x}_i\}_{i=1}^n \mid \boldsymbol{\mu}) p(\boldsymbol{\mu})$
- ◯ $\hat{\boldsymbol{\mu}}_{\text{MAP}} = \arg\max_{\boldsymbol{\mu}} \dfrac{p(\{\mathbf{x}_i\}_{i=1}^n \mid \boldsymbol{\mu})}{p(\boldsymbol{\mu})}$
- ◯ $\hat{\boldsymbol{\mu}}_{\text{MAP}} = \arg\max_{\boldsymbol{\mu}} p(\{\mathbf{x}_i\}_{i=1}^n \mid \boldsymbol{\mu})$

Solution:

(i) [1 Pt] Correct answer selected: $\arg\max_{\boldsymbol{\mu}} p(\{\mathbf{x}_i\}_{i=1}^n \mid \boldsymbol{\mu}) p(\boldsymbol{\mu})$

$\arg\max_{\boldsymbol{\mu}} p(\{\mathbf{x}_i\}_{i=1}^n \mid \boldsymbol{\mu}) p(\boldsymbol{\mu})$**.** By Bayes' rule, the posterior $p(\boldsymbol{\mu} \mid \{\mathbf{x}_i\}_{i=1}^n)$ is proportional to the likelihood $p(\{\mathbf{x}_i\}_{i=1}^n \mid \boldsymbol{\mu})$ times the prior $p(\boldsymbol{\mu})$. Since the denominator $p(\{\mathbf{x}_i\}_{i=1}^n)$ does not depend on $\boldsymbol{\mu}$, maximizing the posterior is strictly equivalent to maximizing the joint probability (the numerator).

(e) **[2 Pts]** Fill in the blanks to conceptually explain the behavior of the MAP estimator — $\hat{\boldsymbol{\mu}}_{\text{MAP}}$.

(i) **[1 Pt]** Suppose we collect an infinite amount of data ($n \to \infty$).
Conceptually, $\hat{\boldsymbol{\mu}}_{\text{MAP}}$ converges to ___$\hat{\boldsymbol{\mu}}_{\text{MLE}}$ (or $\bar{\mathbf{x}}$)___ because with infinite data, the ___**data / evidence**___ completely overwhelms any prior beliefs.
(ii) [0.5 Pt] First blank: $\hat{\boldsymbol{\mu}}_{\text{MLE}}$ (or $\bar{\mathbf{x}}$)
(iii) [0.5 Pt] Second blank: data / evidence / likelihood

(ii) **[1 Pt]** Suppose our prior becomes a "point mass," meaning we are absolutely certain that $\boldsymbol{\mu} = \boldsymbol{\mu}_0$ before seeing any data (equivalent to $\boldsymbol{\Sigma}_0 \to \mathbf{0}$).
Conceptually, $\hat{\boldsymbol{\mu}}_{\text{MAP}}$ converges to ___$\boldsymbol{\mu}_0$___ because an absolutely certain prior will ignore all observed ___**data**___.
(iv) [0.5 Pt] First blank: $\boldsymbol{\mu}_0$
(v) [0.5 Pt] Second blank: data / observations

Ozan Risks It All for MoG

[11 Pts]

Suppose we observe n i.i.d. data points $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$ from a **Mixture of Gaussians** — a model where each data point was generated by one of K Gaussian clusters, but we don't observe which cluster k generated which data point $\mathbf{x}_i$.

The model has parameters $\boldsymbol{\theta} = (\pi_1, \ldots, \pi_K,\ \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K)$ and one latent variable z_i per data point $\mathbf{x}_i$:

Quantity	Notation	Interpretation
Mixing weights	$\pi_k \in [0, 1]$	How likely a data point is to come from cluster k (must sum to 1 across clusters).
Cluster means	$\boldsymbol{\mu}_k \in \mathbb{R}^d$	Center of cluster k.
Latent assignments	$z_i \in \{1, \ldots, K\}$	Which cluster generated $\mathbf{x}_i$ (unknown to us).

To generate each data point $\mathbf{x}_i$:

1. Draw a cluster — $z_i \sim p(z_i)$ — where $p(z_i = k) = \pi_k$.
2. Draw a data point — $\mathbf{x}_i \sim p(\mathbf{x}_i \mid z_i)$ — where $p(\mathbf{x}_i \mid z_i) = \mathcal{N}(\mathbf{x}_i \mid \boldsymbol{\mu}_{z_i},\ \sigma^2\mathbf{I})$.

We assume all K components share a **known, fixed covariance** $\sigma^2\mathbf{I}$, so the Gaussian PDF simplifies to

$$\mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_k, \sigma^2\mathbf{I}) = \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{\|\mathbf{x} - \boldsymbol{\mu}_k\|^2}{2\sigma^2}\right).$$

(a) **[2 Pts]** Write the **log-likelihood** — $\ell(\boldsymbol{\theta}) = \sum_{i=1}^n \log p(\mathbf{x}_i \mid \boldsymbol{\theta})$ — marginalizing over z_i.

Recall: to marginalize over a discrete latent variable, sum over all its possible values:

$$p(a) = \sum_b p(a \mid b)\, p(b).$$

Solution:

(i) [1 Pt] Marginal density: $p(\mathbf{x}_i \mid \boldsymbol{\theta}) = \sum_k \pi_k \mathcal{N}(\mathbf{x}_i \mid \boldsymbol{\mu}_k, \sigma^2\mathbf{I})$

(ii) [1 Pt] Log-likelihood: $\ell(\boldsymbol{\theta}) = \sum_i \log(\sum_k \ldots)$

Marginalizing over z_i:

$$p(\mathbf{x}_i \mid \boldsymbol{\theta}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_i \mid \boldsymbol{\mu}_k, \sigma^2 \mathbf{I}). \qquad \textit{— awards (i)}$$

The marginal log-likelihood is:

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log \left(\sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_i \mid \boldsymbol{\mu}_k, \sigma^2 \mathbf{I}) \right). \qquad \textit{— awards (ii)}$$

(b) **[4 Pts]** We define the posterior responsibility — r_{ik} — as the probability that data point $\mathbf{x}_i$ was generated by cluster k:

$$r_{ik} = p(z_i = k \mid \mathbf{x}_i, \boldsymbol{\theta}).$$

(i) **[3 Pts]** Derive an expression for r_{ik} using Bayes' rule.

$$p(A \mid B) = \frac{p(B \mid A)\, p(A)}{p(B)}$$

Solution:

(i) [1 Pt] Applies Bayes' rule correctly: numerator $= \pi_k \mathcal{N}(\dots)$, denominator $= \sum_j$

(ii) [1 Pt] Correctly substitutes Gaussian density

(iii) [1 Pt] Simplifies: normalizing constants cancel, shows exponential/distance form

By Bayes' rule,

$$r_{ik} = \frac{\pi_k \mathcal{N}(\mathbf{x}_i \mid \boldsymbol{\mu}_k, \sigma^2 \mathbf{I})}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}_i \mid \boldsymbol{\mu}_j, \sigma^2 \mathbf{I})}. \qquad \textit{— awards (i)}$$

Substituting the Gaussian density,

$$\mathcal{N}(\mathbf{x}_i \mid \boldsymbol{\mu}_k, \sigma^2 \mathbf{I}) = \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2 \right), \qquad \textit{— awards (ii)}$$

the normalizing constants $(2\pi\sigma^2)^{d/2}$ cancel:

$$r_{ik} = \frac{\pi_k \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2\right)}{\sum_{j=1}^{K} \pi_j \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{x}_i - \boldsymbol{\mu}_j\|^2\right)}. \qquad \textit{— awards (iii)}$$

(ii) **[1 Pt]** Based on your derivation in part (i), which of the following best describes the role of the squared distance $\|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2$ in determining the responsibility r_{ik}?

- ◯ r_{ik} is directly proportional to the squared distance to $\boldsymbol{\mu}_k$ (i.e., further points have higher responsibility).
- ◯ r_{ik} decreases exponentially as the squared distance to $\boldsymbol{\mu}_k$ increases, heavily favoring the closest cluster.
- ◯ The squared distance only scales the responsibility linearly without affecting which cluster is most favored.
- ◯ The squared distance dictates the overall scale of σ^2 but does not impact the soft assignments.

Solution:

(i) [1 Pt] Correct answer selected: r_{ik} decreases exponentially as the squared distance to $\boldsymbol{\mu}_k$ increases...

Option (b). The exponential decay $\exp(-\frac{1}{2\sigma^2}\|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2)$ means that as distance grows, the responsibility shrinks exponentially, forming a softmax-like distribution over distances.

(c) **[1 Pt]** If we try to maximize $\ell(\boldsymbol{\theta})$ directly by differentiating and setting to zero, we obtain:

$$\frac{\partial \ell}{\partial \boldsymbol{\mu}_k} = \frac{1}{\sigma^2} \sum_{i=1}^{n} r_{ik} \left(\mathbf{x}_i - \boldsymbol{\mu}_k\right) = \mathbf{0},$$

Why doesn't this yield a closed-form solution for $\boldsymbol{\mu}_k$?

- ◯ The system is underdetermined (more unknowns than equations).
- ◯ The weights r_{ik} themselves depend on all the $\boldsymbol{\mu}_j$, creating a circular dependency.
- ◯ The covariance $\sigma^2\mathbf{I}$ is unknown.
- ◯ The log-likelihood has multiple global optima.

Solution:

(i) [1 Pt] Correct answer selected: The weights r_{ik} depend on all the $\boldsymbol{\mu}_j$, creating a circular dependency.

The weights r_{ik} depend on all the $\boldsymbol{\mu}_j$, creating a circular dependency. Solving for $\boldsymbol{\mu}_k$ requires knowing r_{ik}, but r_{ik} depends on all the means through the Gaussian densities. This motivates an iterative approach: alternate between computing r_{ik} given fixed parameters, and optimizing parameters given fixed r_{ik}.

(d) **[2 Pts]** To break this circularity, we define the **Looks** function.

$$\mathit{Looks}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \sum_{k=1}^{K} r_{ik} \left[\log \pi_k - \frac{1}{2\sigma^2}\|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2\right] + \text{const.}$$

We update our parameters by maximizing $\mathcal{L}ooks$ — a process we call *looksmaxxing*.

(i) **[1 Pt]** We obtain our updated cluster centers — $\hat{\boldsymbol{\mu}}_k$ — by looksmaxxing over $\boldsymbol{\mu}_k$.

$$\hat{\boldsymbol{\mu}}_k = \underset{\boldsymbol{\mu}_k}{\operatorname{argmax}}\, \mathcal{L}ooks(\boldsymbol{\theta}) = \frac{\sum_{i=1}^n r_{ik}\, \mathbf{x}_i}{\sum_{i=1}^n r_{ik}}.$$

Which of the following correctly explains $\hat{\boldsymbol{\mu}}_k$?

- ◯ It is the unweighted average of all data points, since every point contributes equally to every cluster.
- ◯ It is a weighted average of the data, where each point's contribution to cluster k is weighted by its responsibility r_{ik}.
- ◯ It is the data point closest to the previous mean $\boldsymbol{\mu}_k$, since the algorithm selects a representative point.
- ◯ It is the median of the points assigned to cluster k, which is more robust to outliers than the mean.

Solution:
(i) [1 Pt] Correct answer selected: It is a weighted average of the data, where each point's contribution to cluster k is weighted by its responsibility r_{ik}.

Option (b). $\hat{\boldsymbol{\mu}}_k$ is a responsibility-weighted average: each $\mathbf{x}_i$ is weighted by $r_{ik} = p(z_i = k \mid \mathbf{x}_i, \boldsymbol{\theta})$, the soft probability that point i belongs to cluster k. Option (a) is wrong because points contribute unequally via r_{ik}. Option (c) is wrong because the update computes a continuous weighted average, not a discrete selection. Option (d) is wrong because the update is a weighted mean, not a median.

(ii) **[1 Pt]** We obtain our updated mixing weights — $\hat{\pi}_k$ — by looksmaxxing over π_k.

$$\hat{\pi}_k = \underset{\text{s.t.}\sum_{k=1}^K \pi_k = 1}{\underset{\pi_k}{\operatorname{argmax}}}\, \mathcal{L}ooks(\boldsymbol{\theta}) = \frac{\sum_{i=1}^n r_{ik}}{n}.$$

Which of the following correctly explains $\hat{\pi}_k$?

- ◯ r_{ik} indicates whether point i is closest to $\boldsymbol{\mu}_k$, so $\sum_i r_{ik}$ counts the points nearest to cluster k, and $\hat{\pi}_k$ is their fraction.
- ◯ r_{ik} is the probability that point i belongs to cluster k, so $\sum_i r_{ik}$ is the effective number of points in cluster k, and $\hat{\pi}_k$ is the expected fraction.
- ◯ r_{ik} sums to 1 over both i and k, so $\sum_i r_{ik} = 1$ for every k, and $\hat{\pi}_k = 1/n$ always.
- ◯ r_{ik} measures the distance from $\mathbf{x}_i$ to $\boldsymbol{\mu}_k$, so $\sum_i r_{ik}$ is the total distance to cluster k, and $\hat{\pi}_k$ is the average distance.

Solution:

(i) [1 Pt] Correct answer selected: r_{ik} is the probability that point i belongs to cluster k, so $\sum_i r_{ik}$ is the effective number of points...

Option (b). r_{ik} is a soft assignment; $\sum_i r_{ik}$ is the effective count N_k; N_k/n is the expected fraction. Option (a) is wrong because r_{ik} is not thresholded. Option (c) is wrong because responsibilities normalize over k (for fixed i), not over i (for fixed k). Option (d) is wrong because r_{ik} is a probability, not a distance.

(e) **[2 Pts]** Consider the behavior of this model in the limit $\sigma^2 \to 0$. In this limit, $r_{ik} \to 1$ for exactly one cluster k and 0 for all other clusters.

(i) **[1 Pt]** Which of the following is the correct interpretation of this result?

- ◯ The model is overfitting because the variance shrinks to zero.
- ◯ The responsibilities become hard cluster assignments, committing each point to its closest center.
- ◯ The responsibilities become a uniform distribution across all clusters since σ^2 is negligible.
- ◯ The $\mathcal{L}ooks$ function diverges to negative infinity.

Solution:
(i) [1 Pt] Correct answer selected: The responsibilities become hard cluster assignments...

Option (b). As $\sigma^2 \to 0$, only the smallest distance $\|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2$ dominates the normalization equation. This effectively yields a hard one-hot assignment where a data point's responsibility shifts entirely to the single nearest cluster mean.

(ii) **[1 Pt]** In this limit, what well-known algorithm does this iterative process reduce to?

Solution:
(i) [1 Pt] K-means (or Lloyd's Algorithm)

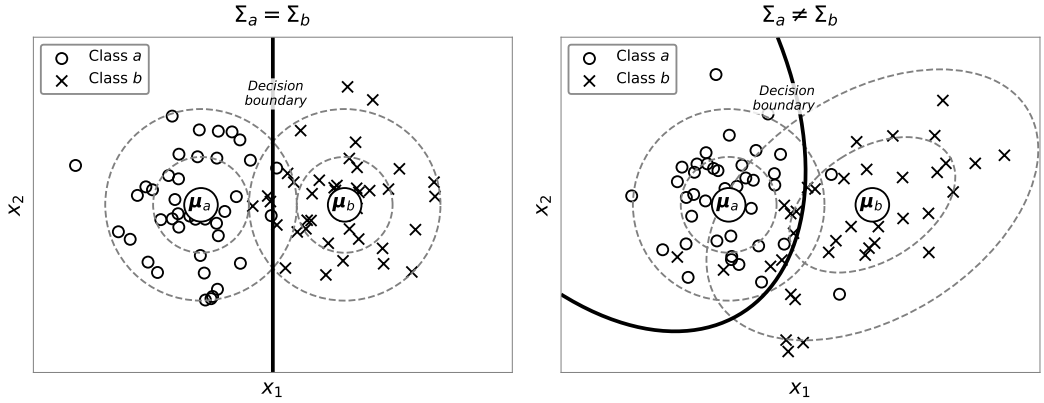
K-means.

We Adopt a Sigma Grindset **[10 Pts]**

Consider binary classification with feature vectors $\mathbf{x} \in \mathbb{R}^d$, labels $y \in \{a, b\}$, and equal priors $p(y = a) = p(y = b) = \frac{1}{2}$. We consider three classifiers trained on a dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$:

Classifier	Description
LDA	Models the data using Gaussian distributions with a **shared** covariance matrix $\boldsymbol{\Sigma}$ across both classes.
QDA	Models the data using Gaussian distributions with **individual** covariance matrices $\boldsymbol{\Sigma}_a, \boldsymbol{\Sigma}_b$ for each class.
Logistic Regression	Directly estimates the probability of the label $p(y \mid \mathbf{x})$ using the logistic (sigmoid) function. It makes no assumptions about the data being Gaussian.

The figure below illustrates shared covariances (left) and differing covariances (right).



(a) **[3 Pts] Generative vs. Discriminative Models.**

Which of the following correctly categorizes each classifier as a **generative** or **discriminative** model?

(i) Linear Discriminant Analysis (LDA) is:

◯ Generative ◯ Discriminative

Solution:

(i) [1 Pt] Correct answer selected: Generative

Generative. LDA models the joint distribution $p(\mathbf{x}, y)$ by specifying class priors and class-conditional covariance matrices.

(ii) Quadratic Discriminant Analysis (QDA) is:

◯ Generative ◯ Discriminative

Solution:
(i) [1 Pt] Correct answer selected: Generative

Generative. Like LDA, QDA explicitly models the distributions of the features $p(\mathbf{x} \mid y)$.

(iii) Logistic Regression is:

◯ Generative ◯ Discriminative

Solution:
(i) [1 Pt] Correct answer selected: Discriminative

Discriminative. Logistic Regression models the posterior $p(y \mid \mathbf{x})$ directly without attempting to model the distribution of the features $\mathbf{x}$.

(b) **[4 Pts]** Suppose the class-conditional distributions are Gaussian with a **shared** covariance (left panel above):

$$p(\mathbf{x} \mid y = a) = \mathcal{N}(\boldsymbol{\mu}_a, \boldsymbol{\Sigma}), \qquad p(\mathbf{x} \mid y = b) = \mathcal{N}(\boldsymbol{\mu}_b, \boldsymbol{\Sigma}).$$

(i) **[2 Pts]** Starting from Bayes' rule and these Gaussian class-conditionals, show that the log-odds

$$\log \frac{p(y = b \mid \mathbf{x})}{p(y = a \mid \mathbf{x})}$$

simplifies to $\mathbf{w}^\top \mathbf{x} + b$ for some weight vector $\mathbf{w}$ and bias b.
You may drop any terms that do not depend on $\mathbf{x}$.

Solution:
(i) [1 Pt] Applies Bayes' rule and substitutes Gaussian PDFs
(ii) [1 Pt] Cancels $\mathbf{x}^\top \boldsymbol{\Sigma}^{-1} \mathbf{x}$ (shared $\boldsymbol{\Sigma}$) and identifies $\mathbf{w}^\top \mathbf{x} + b$

By Bayes' rule with equal priors,

$$\log \frac{p(y=b \mid \mathbf{x})}{p(y=a \mid \mathbf{x})} = \log \frac{p(\mathbf{x} \mid y=b)}{p(\mathbf{x} \mid y=a)}. \qquad \textit{— awards (i)}$$

Substituting the Gaussian PDFs,

$$= -\tfrac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_b)^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu}_b) + \tfrac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_a)^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu}_a).$$

The $\mathbf{x}^\top \boldsymbol{\Sigma}^{-1}\mathbf{x}$ terms cancel (shared $\boldsymbol{\Sigma}$), leaving

$= (\boldsymbol{\mu}_b - \boldsymbol{\mu}_a)^\top \boldsymbol{\Sigma}^{-1}\mathbf{x} - \frac{1}{2}(\boldsymbol{\mu}_b^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_b - \boldsymbol{\mu}_a^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_a) = \mathbf{w}^\top \mathbf{x} + b,$ *— awards (ii)*

which is linear in $\mathbf{x}$.

(ii) **[1 Pt]** Using your result from (i), which statement best describes the relationship between LDA and Logistic Regression?

- ◯ **LDA is a special case of Logistic Regression** — LDA makes stronger assumptions (LDA assumes Gaussian data whereas Logistic Regression doesn't).
- ◯ **Logistic Regression is a special case of LDA** — Logistic Regression requires Gaussian data to work.
- ◯ **They are equivalent** — they always produce the same decision boundary.
- ◯ **They are unrelated** — the matching functional form is a coincidence.

Solution:
(i) [1 Pt] Correct answer selected: LDA is a special case of Logistic Regression

LDA is a special case of Logistic Regression. LDA assumes Gaussian class-conditionals and derives the sigmoid form; Logistic Regression assumes the sigmoid form directly without requiring Gaussian data. LDA's model is a strict subset.

(iii) **[1 Pt]** Suppose n is small relative to d. Which classifier has lower variance — LDA or QDA?

◯ LDA ◯ QDA ◯ Equal variance

Solution:
(i) [1 Pt] Correct answer selected: LDA

LDA. It estimates one shared covariance matrix; QDA estimates two. With small n, QDA's extra parameters increase variance.

(c) **[3 Pts]** Now suppose the true covariances differ: $\boldsymbol{\Sigma}_a \neq \boldsymbol{\Sigma}_b$.

(i) **[1 Pt]** What happens to the log-odds from part (a)(i)?

- ◯ **Still linear in $\mathbf{x}$** — the $\mathbf{x}^\top \boldsymbol{\Sigma}^{-1}\mathbf{x}$ terms still cancel.
- ◯ **Quadratic in $\mathbf{x}$** — the $\mathbf{x}^\top \boldsymbol{\Sigma}_k^{-1}\mathbf{x}$ terms no longer cancel.
- ◯ **Exponential in $\mathbf{x}$** — the Gaussian PDF introduces exponentials that don't simplify.

Solution:
(i) [1 Pt] Correct answer selected: Quadratic in $\mathbf{x}$

Quadratic. When $\boldsymbol{\Sigma}_a \neq \boldsymbol{\Sigma}_b$, the quadratic terms $\mathbf{x}^\top \boldsymbol{\Sigma}_k^{-1} \mathbf{x}$ differ between classes and don't cancel.

(ii) **[1 Pt]** Compared to when $\boldsymbol{\Sigma}_a = \boldsymbol{\Sigma}_b$, what happens to **LDA**'s bias when $\boldsymbol{\Sigma}_a \neq \boldsymbol{\Sigma}_b$?

◯ bias increases ◯ bias is unchanged ◯ bias decreases

Solution:
(i) [1 Pt] Correct answer selected: Increases

Increases. LDA can only produce linear boundaries; the true boundary is now quadratic.

(iii) **[1 Pt]** Compared to when $\boldsymbol{\Sigma}_a = \boldsymbol{\Sigma}_b$, what happens to **QDA**'s bias when $\boldsymbol{\Sigma}_a \neq \boldsymbol{\Sigma}_b$?

◯ Increases ◯ Unchanged ◯ Decreases

Solution:
(i) [1 Pt] Correct answer selected: Unchanged

Unchanged (low). QDA can represent the quadratic boundary.

On a Downward Spiral **[11 Pts]**

Consider the following dataset:

Example	x_1	x_2	y
1	1	0	1
2	0	1	2
3	1	1	4

We model the data with linear regression (no regularization, no bias):

$$f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}, \qquad \mathbf{w} \in \mathbb{R}^2.$$

The squared loss is

$$J(\mathbf{w}) = \sum_{i=1}^{3} \left(\mathbf{w}^\top \mathbf{x}^{(i)} - y^{(i)}\right)^2.$$

(a) **[2 Pts]** Derive the gradient of $J(\mathbf{w})$ with respect to $\mathbf{w}$ in summation form.

Solution:

(i) [1 Pt] Correctly applies chain rule: $2(\mathbf{w}^\top \mathbf{x}^{(i)} - y^{(i)})\mathbf{x}^{(i)}$

(ii) [1 Pt] Writes complete summation form

Since

$$J(\mathbf{w}) = \sum_{i=1}^{3} \left(\mathbf{w}^\top \mathbf{x}^{(i)} - y^{(i)}\right)^2,$$

we differentiate term-by-term:

$$\nabla_{\mathbf{w}} J(\mathbf{w}) = \sum_{i=1}^{3} 2\left(\mathbf{w}^\top \mathbf{x}^{(i)} - y^{(i)}\right)\mathbf{x}^{(i)}. \qquad \textit{— awards (i,ii)}$$

(b) **[3 Pts]** Assume that at iteration t, $\mathbf{w}^{(t)} = \begin{bmatrix}1\\1\end{bmatrix}$ and $\eta = 0.1$.

(i) **[1 Pt]** Compute the gradient and the updated parameters.

$\nabla_{\mathbf{w}} J(\mathbf{w}^{(t)}) =$ _____ $\mathbf{w}^{(t+1)} =$ _____

Solution: (Carry-forward: full credit possible using student's own prior answer.)
(i) [0.5 Pt] Gradient $= [-4, -6]^\top$
(ii) [0.5 Pt] $\mathbf{w}^{(t+1)} = [1.4, 1.6]^\top$

$$\nabla_{\mathbf{w}} J(\mathbf{w}^{(t)}) = \begin{bmatrix} -4 \\ -6 \end{bmatrix} \quad \textit{— awards (i)}$$

$$\mathbf{w}^{(t+1)} = \begin{bmatrix} 1.4 \\ 1.6 \end{bmatrix} \quad \textit{— awards (ii)}$$

(ii) **[1 Pt]** Compute both losses.
$J(\mathbf{w}^{(t)}) =$ _____ $J(\mathbf{w}^{(t+1)}) =$ _____

Solution:
(i) [0.5 Pt] $J(\mathbf{w}^{(t)}) = 5$
(ii) [0.5 Pt] $J(\mathbf{w}^{(t+1)}) = 1.32$

$J(\mathbf{w}^{(t)}) = 5$ *— awards (i)*, $J(\mathbf{w}^{(t+1)}) = 1.32$ *— awards (ii)*.

(iii) **[1 Pt]** Did the loss decrease?

◯ Yes ◯ No

Solution:
(i) [1 Pt] Correct answer selected: Yes

Yes. $1.32 < 5$.

(c) **[3 Pts]** In **stochastic gradient descent (SGD)**, instead of summing over all data points, we approximate the gradient using a single randomly chosen point $(\mathbf{x}^{(i)}, y^{(i)})$:

$$\hat{g}_i(\mathbf{w}) = 2\left(\mathbf{w}^\top \mathbf{x}^{(i)} - y^{(i)}\right)\mathbf{x}^{(i)}.$$

(i) **[1 Pt]** Starting from $\mathbf{w}^{(t)} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$, compute $\hat{g}_1(\mathbf{w}^{(t)})$ (using data point 1).

$\hat{g}_1(\mathbf{w}^{(t)}) =$ _____

Solution: (Carry-forward: full credit possible using student's own prior answer.)
(i) [1 Pt] Correct: $\hat{g}_1 = \mathbf{0}$ (zero vector)

$\mathbf{w}^\top \mathbf{x}^{(1)} = 1$, $y^{(1)} = 1$, residual $= 0$. So $\hat{g}_1 = \mathbf{0}$. The gradient estimate is zero because data point 1 is perfectly predicted at this $\mathbf{w}$. — *awards (i)*

(ii) **[1 Pt]** Compute $\hat{g}_3(\mathbf{w}^{(t)})$ (using data point 3).

$\hat{g}_3(\mathbf{w}^{(t)}) =$ _____

Solution: (Carry-forward: full credit possible using student's own prior answer.)
(i) [1 Pt] Correct: $\hat{g}_3 = [-4, -4]^\top$

$\mathbf{w}^\top \mathbf{x}^{(3)} = 2$, $y^{(3)} = 4$, residual $= -2$:

$$\hat{g}_3 = 2(-2)\begin{bmatrix}1\\1\end{bmatrix} = \begin{bmatrix}-4\\-4\end{bmatrix} \qquad \text{— awards (i)}$$

(iii) **[1 Pt]** Let $\nabla_{\mathbf{w}} J(\mathbf{w}^{(t)})$ denote the full-batch gradient you computed in part (b). Which statement is true about SGD gradients?

- ◯ They always point in the same direction as the full-batch gradient.
- ◯ They are always smaller in magnitude than the full-batch gradient.
- ◯ They can be zero even when the full-batch gradient is nonzero.
- ◯ They are always equal to $\frac{1}{n}\nabla_{\mathbf{w}} J(\mathbf{w})$.

Solution:
(i) [1 Pt] Correct answer selected: They can be zero even when the full-batch gradient is nonzero.

(c). As (i) shows, $\hat{g}_1 = \mathbf{0}$ while $\nabla J \neq \mathbf{0}$: a single-point gradient can vanish at a non-stationary point. (a) is false: $\hat{g}_3 = [-4, -4]^\top$ vs $\nabla J = [-4, -6]^\top$ have different directions. (b) is false: neither bound always holds. (d) is false: $\frac{1}{3}[-4, -6]^\top = [-\frac{4}{3}, -2]^\top$, which does not equal any $\hat{g}_i$.

(d) **[2 Pts]** Suppose we replace x_1 with $100 \cdot x_1$ for every data point (x_2 and y unchanged).

(i) **[1 Pt]** How does $\nabla_{\mathbf{w}} J(\mathbf{w})$ change?

- ◯ Both $\dfrac{\partial J}{\partial w_1}$ and $\dfrac{\partial J}{\partial w_2}$ scale by a factor of 100.
- ◯ $\dfrac{\partial J}{\partial w_1}$ increases dramatically; $\dfrac{\partial J}{\partial w_2}$ is completely unchanged.
- ◯ $\dfrac{\partial J}{\partial w_1}$ increases dramatically and dominates the gradient; $\dfrac{\partial J}{\partial w_2}$ also changes but less severely.

◯ The gradient direction is unchanged; only its overall magnitude scales.

Solution:

(i) [1 Pt] Correct answer selected: $\partial J/\partial w_1$ increases dramatically and dominates the gradient...

$\frac{\partial J}{\partial w_1}$ increases dramatically and dominates the gradient; $\frac{\partial J}{\partial w_2}$ also changes but less severely. The partial $\partial J/\partial w_1$ scales with x_1^2 ($\sim 10{,}000\times$). The partial $\partial J/\partial w_2$ also changes due to cross-terms involving x_1 in the residuals, but less dramatically. The gradient becomes dominated by the w_1 direction.

(ii) **[1 Pt]** Which of the following **most directly** addresses the **root cause** of this problem?

◯ Use a much smaller learning rate η.
◯ Standardize each feature to have zero mean and unit variance before training.
◯ Add L2 regularization.
◯ Use more training data.

Solution:

(i) [1 Pt] Correct answer selected: Standardize each feature to have zero mean and unit variance...

Standardize each feature... The root cause is that features have vastly different scales, making the loss landscape ill-conditioned. Standardization eliminates the scale mismatch directly. A smaller η mitigates divergence but does not fix the ill-conditioning (progress in the w_2 direction becomes unnecessarily slow). L2 regularization shrinks weights but does not address the conditioning. More data is irrelevant.

(e) **[1 Pt]** Which of the following models from this exam does **not** have a closed-form optimal solution, and therefore requires an iterative method such as gradient descent?

◯ Ordinary least squares linear regression
◯ Ridge regression
◯ MAP estimation for a Gaussian mean with a Gaussian prior
◯ Logistic regression

Solution:

(i) [1 Pt] Correct answer selected: Logistic regression

Logistic regression. OLS, ridge regression, and MAP estimation for a Gaussian mean all yield closed-form solutions (derived earlier on this exam). Logistic regression involves the sigmoid function, making $\nabla L = 0$ a nonlinear system with no closed-form solution.

You are done with the midterm! Congratulations!

Use this page to draw something hard 💯