CS 189 Spring 2020 Introduction to Machine Learning — Midterm B

- The exam is closed book, closed notes except your self-made cheat sheets.
- You will submit your answers to the multiple-choice questions through Gradescope via the assignment "**Midterm B – Multiple Choice**"; please **do not** submit your multiple-choice answers on paper. By contrast, you will submit your answers to the written questions by writing them on paper by hand, scanning them, and submitting them through Gradescope via the assignment "**Midterm B – Writeup**."
- Please write your name at the top of each page of your written answers. (You may do this before the exam.)
- You have 80 minutes to complete the midterm exam (6:40–8:00 PM). (If you are in the DSP program and have an allowance of 150% or 200% time, that comes to 120 minutes or 160 minutes, respectively.)
- When the exam ends (8:00 PM), **stop writing**. You must submit your multiple-choice answers before 8:00 PM sharp. Late multiple-choice submissions will be penalized at a rate of 5 points per minute after 8:00 PM. (The multiple-choice questions are worth 40 points total.)
- From 8:00 PM, you have 15 minutes to scan the written portion of your exam and turn it into Gradescope via the assignment "Midterm B Writeup." Most of you will use your cellphone and a third-party scanning app. If you have a physical scanner, you may use that. You do not need to scan the title page or the multiple-choice page. Late written submissions will be penalized at a rate of 10 points per minute after 8:15 PM. (The written portion is worth 60 points total.)
- Mark your answers to multiple-choice questions directly into Gradescope. Write your answers to written questions on the corresponding pages of the Answer Sheet or on blank paper. If you need overflow space for a written question, use additional sheets of blank paper. **Clearly label all written questions and all subparts of each written question**.
- Following the exam, you must use Gradescope's **page selection mechanism** to mark which questions are on which pages of your exam (as you do for the homeworks).
- The total number of points is 100. There are 10 multiple choice questions worth 4 points each, and three written questions worth 20 points each.
- For multiple answer questions, fill in the bubbles for **ALL correct choices:** there may be more than one correct choice, but there is always at least one correct choice. **NO partial credit** on multiple answer questions: the set of all correct answers must be checked.

First name	
Last name	
SID	

Q1. [40 pts] Multiple Answer

Fill in the bubbles for **ALL correct choices**: there may be more than one correct choice, but there is always at least one correct choice. **NO partial credit**: the set of all correct answers must be checked.

(a) [4 pts] Recall that in subset selection, we attempt to identify poorly predictive features and ignore them. Which of the following are reasons why we may seek to drop features available to our model?

- ○ A: To reduce model bias
- ● B: To reduce model variance
- ● C: To increase speed of prediction on test points
- ● D: To improve model interpretability

Removing features likely increases bias, since the model becomes less expressive. On the other hand, all features increase variance; therefore, dropping features may reduce the variance of the model. Of course, having fewer features may also make prediction slightly faster and improve interpretability.

(b) [4 pts] Consider a random variable $X \sim \mathcal{N}(\mu, \Sigma) \in \mathbb{R}^d$, where the multivariate Gaussian probability density function (PDF) is axis-aligned, Σ is positive definite, and the standard deviation along coordinate axis i is σ_i. Select all that apply.

- ○ A: The d features of X are uncorrelated but not necessarily independent
- ○ B: $\Sigma = \text{diag}(\sqrt{\sigma_1}, \sqrt{\sigma_2}, \ldots, \sqrt{\sigma_d})$
- ● C: Σ has a symmetric square root $\Sigma^{1/2}$ with eigenvalues $\sigma_1, \sigma_2, \ldots, \sigma_d$
- ● D: $(X-\mu)^\top \Sigma^{-1}(X-\mu) \geq 0$

A is incorrect: It is true that because the multivariate Gaussian is axis-aligned, its components are uncorrelated. However, the uncorrelated multivariate Gaussian also implies independence here, since $p(X;\mu,\Sigma) = \prod_{i=1}^n p(x_i;\mu_i,\sigma_i^2)$ (see Option C).

B is incorrect: The eigenvectors are standard basis vectors, so Σ must be a diagonal matrix, but with elements $\text{diag}(\sigma_1^2, \ldots, \sigma_n^2)$.

C is correct: Because Σ is a diagonal matrix, expanding out the probability density function gives us option C.

D is correct: Σ is positive definite, so Σ^{-1} is also positive definite. Thus, $(X-\mu)^\top \Sigma^{-1}(X-\mu)$ must be nonnegative.

(c) [4 pts] Given a design matrix $X \in \mathbb{R}^{n \times d}$ representing n sample points with d features, you compute the sample covariance matrix M of your dataset and find that its determinant is $\det M = 0$. What do you know to be true?

- ● A: There is some direction in $\mathbb{R}^d$ along which the sample points have zero variance
- ○ B: The covariance matrix M is positive definite
- ● C: The columns of the centered design matrix $\dot{X}$ are linearly dependent
- ○ D: The rows of the centered design matrix $\dot{X}$ are linearly dependent

(d) [4 pts] For classification problems with two features ($d = 2$, test point $z \in \mathbb{R}^2$), which of the following methods have posterior probability distributions of the form $P(Y|X=z) = s(Az_1^2 + Bz_2^2 + Cz_1z_2 + Dz_1 + Ez_2 + F)$ where s is the logistic function $s(\gamma) = \frac{1}{1+e^{-\gamma}}$ and $A, B, C, D, E, F \in \mathbb{R}$ can all be nonzero?

- ○ A: Logistic regression with linear features
- ● B: Linear discriminant analysis (LDA) with quadratic features
- ● C: Logistic regression with quadratic features
- ● D: Quadratic discriminant analysis (QDA) with linear features

All but A, as standard logistic regression does not have the potential for A, B, C to be nonzero.

(e) [4 pts] Which of the following statements regarding Bayes decision theory are true?

○ A: If the Bayes optimal classifier $r^*(x)$ correctly classifies all points in the design matrix X with 100% accuracy, its Bayes risk must be zero

● B: With 0-1 loss, the two-class Bayes optimal classifier $r^*(x)$ classifies points in a way that minimizes the probability of misclassification

○ C: If you have a design matrix X and you are given the Bayes optimal classifier $r^*(x)$, then you sample a different design matrix X' from the same distribution(s), the original $r^*(x)$ is no longer optimal

○ D: None of the above

A: There may still be some nonzero probability for posteriors of non-selected classes. The optimal classifier aims to achieve lowest Expected Risk, there are no guarantees that this will always be zero.

B: This might be easier to view in terms of having 0-1 Loss. When we have 0-1 loss and decide between class C or D or Loss is effectively just the probability of the opposing class. Therefore to minimize our loss, we would choose to avoid the class with lower probability. This results in ultimately the lowest misclassification error in expectation.

C: The Bayes optimal classifier has nothing to do with any particular set of sample points drawn from the underlying distributions.

(f) [4 pts] We want to minimize the function

$$f(\beta) = \begin{cases} \beta^2 & \text{if } \beta \leq 0, \\ \beta & \text{otherwise.} \end{cases}$$

○ A: Starting from $\beta = 1$, Newton's method will take us to the minimum of f in one step

● B: Starting from $\beta = -1$, Newton's method will take us to the minimum of f in one step

● C: There exists a learning rate ϵ such that starting from $\beta = 1$, gradient descent will take us to the minimum of f in one step

● D: There exists a learning rate ϵ such that starting from $\beta = -1$, gradient descent will take us to the minimum of f in one step

A: The second derivative is 0 for $\beta \geq 0$, hence Newton's method will not be able to find the minimum starting at $\beta = 1$.

B: f is quadratic for $\beta \leq 0$ (the equality is not a typo), and the minimum of that quadratic is at $\beta = 0$, hence Newton's method will find the optimum starting from any $\beta < 0$ in exactly one step.

C: $\epsilon = 1$ works.

D: $\epsilon = 0.5$ works.

(g) [4 pts] In each of the following two figures, there are exactly three training points drawn from an unknown distribution, and the dashed line is a decision boundary.

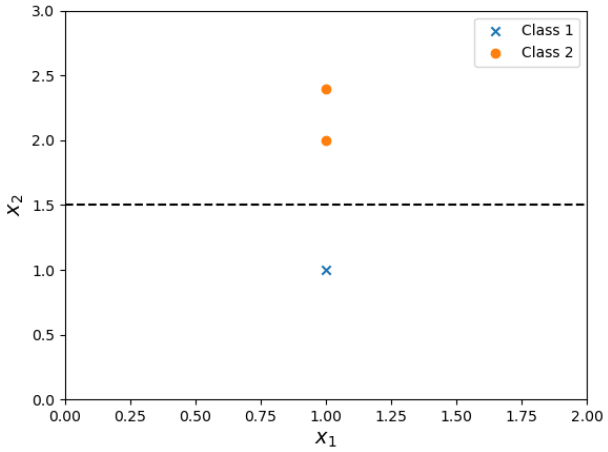


Figure 1

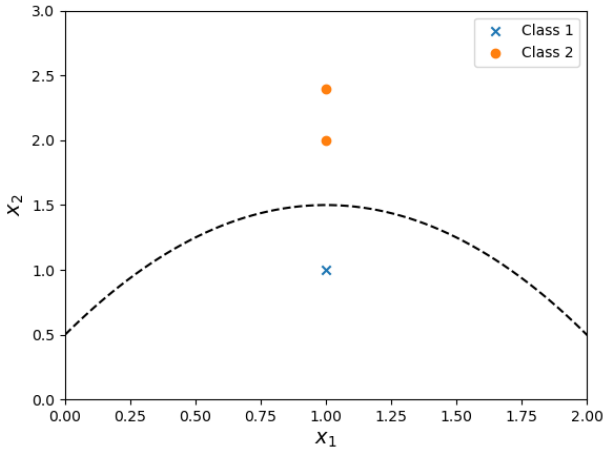


Figure 2

○ A: The Bayes optimal decision boundary always appears as drawn in Figure 1

● B: Both hard-margin and soft-margin (for some choice of C) SVMs could produce the decision boundary in Figure 1 (using only the features x_1 and x_2, plus the fictitious dimension where appropriate)

○ C: Both logistic regression and QDA could produce the decision boundary in Figure 2 (using only the features x_1 and x_2, plus the fictitious dimension where appropriate)

○ D: A hard-margin SVM augmented with the parabolic lifting map $\Phi(x) = [x_1 \quad x_2 \quad x_1^2 + x_2^2]^\top$ could produce the decision boundary in Figure 2

We cannot say anything about the Bayes optimal decision boundary from training points alone; we need to know the distributions that the training points are drawn from.

Logistic regression can only find linear decision boundaries when not given any features other than x_1 and x_2.

The parabolic lifting map cannot produce the decision boundary in Figure 2, because the boundary isn't a circle or a line. (Moreover, in this example, the parabolic lifting map would still produce a line.)

(h) [4 pts] Consider least-squares linear regression with a design matrix $X \in \mathbb{R}^{n \times d}$ and labels $y \in \mathbb{R}^n$. If the solution to the least-squares problem is unique, which of the following must be true?

- ● A: $\text{rank}(X) = d$
- ○ C: $n \leq d$
- ● B: $\text{rank}(X^\top) = d$
- ● D: $d \leq n$

The normal equations from the least-squares problem are

$$X^\top X w = X^\top y.$$

For the solution to be unique, we need to have that $X^\top X$ is invertible. This is equivalent to saying that $\text{rank}(X^\top X) = d$, and equivalently, $\text{rank}(X^\top) = d$. Since the rank of a matrix and the rank of its transpose must be that same, we must have that $\text{rank}(X) = d$ as well. Note that the rank of a matrix is always less than or equal to the number of columns, and also always less than or equal to the number of rows. In this case, regardless of the matrix X, we must have that $\text{rank}(X) \leq \min\{n, d\}$. Since we already determined that $\text{rank}(X) = d$, and $d \geq \min\{n, d\}$, we must therefore have that $d = \min\{n, d\}$ and thus $\text{rank}(X) = d = \min\{n, d\}$. Again, the rank of a matrix and that of its matrix is the same, so the same statement applies for $X^\top$.

(i) [4 pts] Suppose we are performing linear regression on a design matrix X and a label vector y. Recall that the conventional least-squares formulation finds the linear function $h(\cdot)$ that minimizes the empirical risk $R(h) = \frac{1}{n}\sum_{i=1}^{n} L(h(X_i), y_i)$, where $L(\zeta, \gamma) = (\zeta - \gamma)^2$ is the squared-error loss for a prediction $\zeta \in \mathbb{R}$ and a label $\gamma \in \mathbb{R}$. However, you are afraid that your training data may have outliers. Which of the following changes will help mitigate this issue if it exists?

● A: Change the loss function from $L(\zeta, \gamma) = (\zeta - \gamma)^2$ (squared error) to $L(\zeta, \gamma) = |\zeta - \gamma|$ (absolute error)

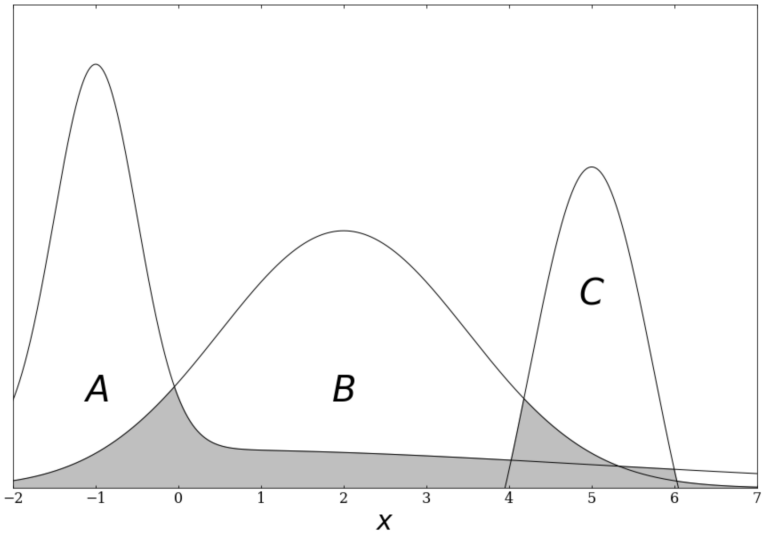
○ B: Change the loss function from $L(\zeta, \gamma) = (\zeta - \gamma)^2$ (squared error) to $L(\zeta, \gamma) = -\gamma \ln \zeta - (1 - \gamma) \ln (1 - \zeta)$ (logistic loss)

○ C: Change the cost function from $R(h) = \frac{1}{n}\sum_{i=1}^{n} L(h(X_i), \gamma_i)$ (mean loss) to $R(h) = \sum_{i=1}^{n} L(h(X_i), \gamma_i)$ (total loss)

○ D: Change the cost function from $R(h) = \frac{1}{n}\sum_{i=1}^{n} L(h(X_i), \gamma_i)$ (mean loss) to $R(h) = \max_{i \in [1,\ldots,n]} L(h(X_i), \gamma_i)$ (maximum loss)

Absolute error is linear wrt the error so it is less sensitive than squared error. Logistic loss is only valid for labels in $[0, 1]$ and is not robust to outliers anyways. Removing the $\frac{1}{n}$ in the objective does not change the optimization problem. Maximum loss is actually more sensitive to outliers than mean loss.

(j) [4 pts] The following chart depicts the class-conditional distributions $P(X|Y)$ for a classification problem with three classes, A, B, and C. Classes A and B are normally distributed over the domain $(-\infty, \infty)$; Class C is defined only over the finite domain depicted below. All three classes have prior probabilities π_A, π_B, π_C **strictly greater than zero**; the chart does **not** show the influence of these priors. We use the **0-1 loss** function.



○ A: The Bayes risk is the area of the shaded region in the chart (including the area not depicted off the sides of the chart, going to $x = \pm\infty$)

● B: Depending on the priors, it is possible that the Bayes rule $r^*(x)$ will classify all inputs as class B

○ C: Depending on the priors, it is possible that the Bayes rule $r^*(x)$ will classify all inputs as class C

○ D: Depending on the priors, it is possible that the Bayes risk is zero

A) This is wrong for several reasons. First, to get the Bayes risk, you would need to scale each curve by its prior. Second, some of the shaded area has to be counted twice; e.g., the area under A where B and C are both above A.

B and C) The only way for something like this to occur is if the prior for a class is significant enough that its joint distribution raises above the other classes' joint distribution for all x. Class C can't do that, as it does not span all of the x-space. Class B can, as it covers the whole number line and its variance is greater than Class A's.

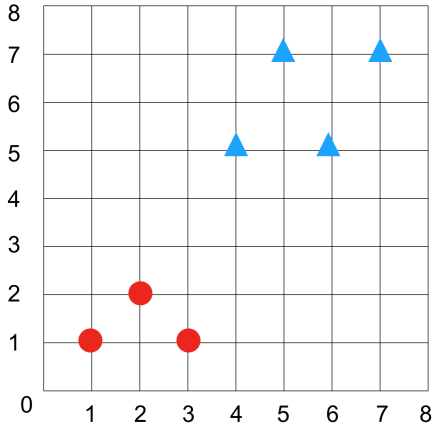
D) Haha

Q2. [20 pts] Hard-Margin Support Vector Machines

Recall that a **maximum margin classifier**, also known as a hard-margin support vector machine (SVM), takes n training points $X_1, X_2, \ldots, X_n \in \mathbb{R}^d$ with labels $y_1, y_2, \ldots, y_n \in \{+1, -1\}$, and finds parameters $w \in \mathbb{R}^d$ and $\alpha \in \mathbb{R}$ that satisfy a certain objective function subject to the constraints

$$y_i(X_i \cdot w + \alpha) \geq 1, \quad \forall i \in \{1, \ldots, n\}.$$

For parts (a) and (b), consider the following training points. Circles are classified as positive examples with label +1 and triangles are classified as negative examples with label −1.



(a) [3 pts] Which points are the support vectors? Write it as $\begin{bmatrix}\text{horizontal}\\ \text{vertical}\end{bmatrix}$. E.g., the bottom right circle is $\begin{bmatrix}3\\1\end{bmatrix}$.

The support vectors are the points $\begin{bmatrix}2\\2\end{bmatrix}$ and $\begin{bmatrix}4\\5\end{bmatrix}$.

(b) [4 pts] If we add the sample point $x = \begin{bmatrix}5\\1\end{bmatrix}$ with label −1 (triangle) to the training set, which points are the support vectors?

The support vectors are the points $\begin{bmatrix}3\\1\end{bmatrix}$, $\begin{bmatrix}4\\5\end{bmatrix}$, and $\begin{bmatrix}5\\1\end{bmatrix}$.

For parts (c)–(f), forget about the figure above, but assume that there is at least one sample point in each class and that the sample points are linearly separable.

(c) [2 pts] Describe the geometric relationship between w and the decision boundary.

The weight vector w (called the *normal vector*) is orthogonal to the decision boundary.

(d) [2 pts] Describe the relationship between w and the margin. (For the purposes of this question, the margin is just a number.)

The margin (the distance from the decision boundary to the nearest sample point) is $1/\|w\|$.

(e) [4 pts] Knowing what you know about the hard-margin SVM objective function, explain why for the optimal (w, α), there must be at least one sample point for which $X_i \cdot w + \alpha = 1$ and one sample point for which $X_i \cdot w + \alpha = -1$.

The objective is to minimize $\|w\|^2$ (or equivalently, $\|w\|$). If every sample point has $y_i(X_i \cdot w + \alpha) > 1$, we can simply scale w to make it smaller until there is a point such that $y_i(X_i \cdot w + \alpha) = 1$, thereby improving the "solution."

If we have a positive sample point for which $X_i \cdot w + \alpha = 1$ but every negative sample point has $X_i \cdot w + \alpha < -1$, we can make α a little greater so that every sample point has $y_i(X_i \cdot w + \alpha) > 1$. Then we can shrink w some more. So any such "solution" cannot be optimal. (The symmetric argument applies if a negative sample point touches the slab but not positive sample point does.)

(f) [5 pts] If we add new features to the sample points (while retaining all the original features), can the optimal $\|w_{\text{new}}\|$ in the enlarged SVM be greater than the optimal $\|w_{\text{old}}\|$ in the original SVM? Can it be smaller? Can it be the same? Explain why! (Most of the points will be for your explanation.)

It can be smaller, or it can be the same, but it cannot be greater.

If w_{old} and α are an optimal solution of the original SVM, when we add features we can create a w_{new} that has the same values as w_{old}, with zeros added for the new features. Then w_{new} and α satisfy all the constraints of the enlarged SVM. These might not be the optimal solution, but the optimal solution of the enlarged SVM cannot have $\|w_{\text{new}}\|$ greater than $\|w_{\text{old}}\|$.

$\|w_{\text{new}}\|$ can be smaller, because the new features can put an arbitrarily large amount of space between the classes, making the margin arbitrarily large.

$\|w_{\text{new}}\|$ will be the same as $\|w_{\text{old}}\|$ if the new features are all zeros in all the sample points.

Q3. [20 pts] Regression with Varying Noise

We derived a cost function for regression problems by assuming that sample points and their labels arise from the following process, and applying **maximum likelihood estimation** (MLE).

- Sample points come from an unknown distribution, $X_i \sim D$.
- Labels y_i are the sum of a deterministic function g plus random noise: $\forall i,\ y_i = g(X_i) + \epsilon_i$, where $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$.

For this problem, we will assume that $\epsilon_i \sim \mathcal{N}(0, \sigma_i^2)$—that is, the variance σ_i^2 of the noise is different for each sample point—and we will examine how our cost function changes as a result. We assume that (magically) we know the value of each σ_i^2. You are given an $n \times d$ design matrix X, an n-vector y of labels, such that the label y_i of sample point X_i is generated as described above, and a list of the noise variances σ_i^2.

(a) [8 pts] Apply MLE to derive the optimization problem that will yield the maximum likelihood estimate of the distribution parameter g. (Note: g is a function, but we can still treat it as the parameter of an optimization problem.) Express your cost function as a summation of loss functions (where you decide what the loss function is), one per sample point.

We first derive the log-likelihood $\ell(g; X, y)$. If $\epsilon_i \sim \mathcal{N}(0, \sigma_i^2)$, then $y_i \sim \mathcal{N}(g(X_i), \sigma_i^2)$. Substituting this into the Gaussian PDF gives

$$\ln f(y_i) = -\frac{(y_i - g(X_i))^2}{2\sigma_i^2} - \text{constant}.$$

Therefore,

$$\begin{aligned}\ell(g; X, y) &= \ln(f(y_1)f(y_2)\dots f(y_n)) \\ &= \ln f(y_1) + \ln f(y_2) + \cdots + \ln f(y_n) \\ &= -\frac{1}{2}\sum_{i=1}^{n} \frac{1}{\sigma_i^2}(y_i - g(X_i))^2 - \text{constant}.\end{aligned}$$

We maximize the likelihood by minimizing the negative log-likelihood, so we have

$$\min_g \sum_{i=1}^{n} \frac{1}{\sigma_i^2}(y_i - g(X_i))^2.$$

(b) [4 pts] We decide to do linear regression, so we parameterize $g(X_i)$ as $g(X_i) = w \cdot X_i$, where w is a d-vector of weights. Write an equivalent optimization problem where your optimization variable is w and the cost function is a function of X, y, w, and the variances σ_i^2. Find a way to express your cost function in matrix notation, with no summations. (This may entail defining a new matrix.)

By substitution, the cost function is

$$\min_w \sum_{i=1}^{n} \frac{1}{\sigma_i^2}(y_i - w \cdot X_i)^2,$$

which we can write as

$$\sum_{i=1}^{n} \frac{1}{\sigma_i^2}(y_i - w \cdot X_i)^2 = (y - Xw)^\top \Omega (y - Xw)$$

where Ω is a diagonal matrix such that $\Omega_{ii} = \frac{1}{\sigma_i^2}$.

(c) [4 pts] Write the solution to your optimization problem as the solution of a linear system of equations. (Again, in matrix notation, with no summations.)

We get the normal equations as usual by setting $\nabla J(w) = 0$, so w can be any solution of

$$X^\top \Omega X w = X^\top \Omega y.$$

(d) [2 pts] Does your solution resemble that of a similar method you know? What is its name?

This is an example of *weighted* least-squares regression.

(e) [2 pts] Compare your solution to the case in which we assume that every sample point has the same noise distribution. In simple terms, how does the amount of noise affect the optimization, and why does this seem like the intuitively right thing to do? Answer in 3 sentences or fewer.

We are penalized less for deviation from sample points that have high variance. If we know our measurement is noisy, we shouldn't try to overfit to it. On the other hand, we should try our best to fit points with low variance because we know what we are fitting is correct.

Q4. [20 pts] Finding Bias, Variance, and Risk

For $z \in \mathbb{R}$, you are trying to estimate a true function $g(z) = 2z^2$ with **least-squares regression**, where the regression function is a line $h(z) = wz$ that goes through the origin and $w \in \mathbb{R}$. Each sample point $x \in \mathbb{R}$ is drawn from the **uniform distribution on** $[-1, 1]$ and has a corresponding label $y = g(x) \in \mathbb{R}$. There is no noise in the labels. We train the model with **just one sample point**! Call it x, and assume $x \neq 0$. We want to apply the bias-variance decomposition to this model.

(a) [3 pts] In one sentence, why do we expect the bias to be large?

Because a line is not a good fit for a parabola.

(b) [6 pts] What is the bias of your model $h(z)$ as a function of a test point $z \in \mathbb{R}$? (*Hint: start by working out the value of the least-squares weight w.*) Your final bias should not include an x; work out the expectation.

The least-squares solution is $w = X^+ y$, where X is the 1×1 matrix $[x]$. Hence $X^+ = (X^\top X)^{-1} X^\top = \frac{x}{x^2} = \frac{1}{x}$. Then

$$\begin{aligned} w &= X^+ y = \frac{1}{x}(2x^2) = 2x, \\ h(z) &= wz = 2xz, \text{ and} \\ \text{bias}(z) &= \mathbb{E}[h(z)] - g(z) = \mathbb{E}[2xz] - 2z^2 = 2z\,\mathbb{E}[x] - 2z^2 = 2z \int_{-1}^{1} x \frac{1}{2}\,\mathrm{d}x - 2z^2 = -2z^2. \end{aligned}$$

(Note: it's pretty obvious that $\mathbb{E}[x] = 0$ for a uniformly distributed $x \in [-1, 1]$; the integral is not required.)

(c) [6 pts] What is the variance of your model $h(z)$ as a function of a test point $z \in \mathbb{R}$? Your final variance should not include an x; work out the expectation.

$$\text{Var}(h(z)) = \text{Var}(2xz) = \mathbb{E}[4x^2 z^2] - \mathbb{E}[2xz]^2 = \int_{-1}^{1} 4x^2 z^2 \frac{1}{2}\,\mathrm{d}x - 4z^2\,\mathbb{E}[x]^2 = \frac{2}{3} x^3 z^2 \Big|_{-1}^{1} - 0 = \frac{4}{3} z^2.$$

An alternative answer is

$$\text{Var}(h(z)) = \text{Var}(2xz) = 4z^2\,\text{Var}(x) = 4z^2\,\mathbb{E}[(x - \mathbb{E}[x])^2] = 4z^2 \int_{-1}^{1} x^2 \frac{1}{2}\,\mathrm{d}x = \frac{2}{3} z^2\, x^3 \Big|_{-1}^{1} = \frac{4}{3} z^2.$$

(d) [5 pts] Let $R(h, z)$ be the risk (expected loss) for a fixed, arbitrary test point $z \in \mathbb{R}$ with the noise-free label $g(z)$ (where the expectation is taken over the distribution of values of (x, y)). What is the mathematical relationship between the risk $R(h, z)$, the bias of $h(z)$ at z, and the variance of $h(z)$ at z? What are the values (as numbers) of these three quantities for $z = 1$?

From class, recall the bias-variance decomposition: $R(h, z) = \text{bias}(z)^2 + \text{Var}(h(z))$. (There is no irreducible error because there is no noise in z's label.)

For $z = 1$, we have $\text{bias}(z) = -2$, $\text{Var}(h(z)) = \frac{4}{3}$, and therefore $R(h, z) = (-2)^2 + \frac{4}{3} = \frac{16}{3}$.