UNIVERSITY OF ALBERTA
CMPUT 267 Winter 2025

Midterm Exam 2

Do Not Distribute

Duration: 60 minutes

Last Name: _____

First Name: _____

Carefully read all of the instructions and questions. Good luck!

---

1. **Do not turn this page** until instructed to begin.
2. This is exam version **0**. Please mark **0 in the special code section in coloum J** of your scantron.
3. Verify that your exam package includes 17 pages (last two are blanck), along with a formula sheet at the end.
4. **Only the scantron will be marked**. All of your answers must be clearly marked on the scantron.
5. Use **pencil only** to fill out the scantron (preferably an HB or #2 pencil).
6. **Erase mistakes completely** on the scantron to avoid misreading by the scanner.
7. **Mark answers firmly and darkly**, filling in the bubbles completely.
8. This exam consists of **15 questions**. Each question is worth **1 mark**. The exam is worth a total of **15 marks**.
9. Some questions may have **multiple correct answers**. To receive **full marks**, you must select **all correct answers**. If you select only **some** of the correct answers, you will receive **partial marks**. Selecting an incorrect option will cancel out a correct one. For example, if you select two answers—one correct and one incorrect—you will receive zero points for that question. If the number of incorrect answers exceeds the correct ones, your score for that question will be zero. **No negative marks** will be given.

Question 1. [1 MARK]

Consider an optimization problem where the goal is to maximize a function $f(w)$ with respect to $w \in \mathbb{R}^d$:

$$\max_{w \in \mathbb{R}^d} \ f(w).$$

Which of the following statements is true?

A. $\max_{w \in \mathbb{R}^d} \ f(w)$ results in the same w^* as $\min_{w \in \mathbb{R}^d} \ [-f(w)]$.

B. $\max_{w \in \mathbb{R}^d} \ f(w)$ results in the same w^* as $\min_{w \in \mathbb{R}^d} \ f(w)$.

C. $\min_{w \in \mathbb{R}^d} \ f(w)$ results in the same optimal value as $\max_{w \in \mathbb{R}^d} \ [-f(w)]$.

D. $\min_{w \in \mathbb{R}^d} \ f(w)$ results in the same optimal value as $-\max_{w \in \mathbb{R}^d} \ [-f(w)]$.

Solution 1. Correct answer(s): A, D

Explanation:

A. **True.** Maximizing $f(w)$ is equivalent to minimizing $-f(w)$ since

$$\arg\max_{w \in \mathbb{R}^d} f(w) = \arg\min_{w \in \mathbb{R}^d} \big[-f(w)\big].$$

Therefore, both formulations yield the same optimal solution w^*.

B. **False.** Minimizing $f(w)$ finds the minimizer of $f(w)$, which generally does not correspond to the maximizer of $f(w)$.

C. **False.** The optimal value of $\max_{w \in \mathbb{R}^d} \ [-f(w)]$ is $-\min_{w \in \mathbb{R}^d} f(w)$, which is the negative of the optimal value from $\min_{w \in \mathbb{R}^d} \ f(w)$, so they are not the same.

D. **True.** Since

$$\max_{w \in \mathbb{R}^d} \ [-f(w)] = -\min_{w \in \mathbb{R}^d} f(w),$$

it follows that

$$-\max_{w \in \mathbb{R}^d} \ [-f(w)] = \min_{w \in \mathbb{R}^d} f(w).$$

Thus, the optimal value from minimizing $f(w)$ is the same as that obtained from $-\max_{w \in \mathbb{R}^d}[-f(w)]$.

Question 2. [1 MARK]

Consider a polynomial regression model that uses a feature map ϕ_p of degree p. The predictor is given by $f(x) = \phi_p(x)^T w$. Suppose you are given a dataset $\mathcal{D} = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ and the empirical loss is defined as

$$\hat{L}(w) = \frac{1}{n} \sum_{i=1}^{n} \Big(\phi_p(x_i)^T w - y_i \Big)^2.$$

In this question, we are interested in finding $\hat{w} = \arg\min_{w \in \mathbb{R}^{\bar{p}}} \hat{L}(w)$ using first-order gradient descent. Which of the following statements is true?

A. The first-order gradient update rule with a fixed step size is

$$w^{(t+1)} = w^{(t)} - \frac{2}{n} \eta^{(t)} \left[\sum_{i=1}^{n} \Big(\phi_p(x_i)^T w - y_i \Big) \phi_p(x_i) \right].$$

B. If you run first-order gradient descent for infinitly many epochs T, you are guaranteed to converge to the minimizer $w^{(T)} = \hat{w}$, where $\hat{w} = \arg\min_{w \in \mathbb{R}^{\bar{p}}} \hat{L}(w)$.

C. $\hat{L}(w)$ is convex.

D. $\hat{L}(w)$ is not convex.

Solution 2. **Correct answer(s): A, B, C**

Explanation:

A. **True.** The loss function is

$$\hat{L}(w) = \frac{1}{n} \sum_{i=1}^{n} \Big(\phi_p(x_i)^T w - y_i \Big)^2.$$

Its gradient with respect to w is

$$\nabla \hat{L}(w) = \frac{2}{n} \sum_{i=1}^{n} \Big(\phi_p(x_i)^T w - y_i \Big) \phi_p(x_i),$$

so the gradient descent update becomes

$$w^{(t+1)} = w^{(t)} - \frac{2}{n} \eta^{(t)} \sum_{i=1}^{n} \Big(\phi_p(x_i)^T w - y_i \Big) \phi_p(x_i).$$

B. **True.** Since $\hat{L}(w)$ is a quadratic function in w (and thus convex) and under appropriate step size choices, gradient descent converges asymptotically to the unique minimizer $\hat{w}$. Therefore, running the algorithm for infinitely many epochs guarantees that $w^{(T)} \to \hat{w}$.

C. **True.** The empirical loss $\hat{L}(w)$ is a sum of squared errors, making it a quadratic function in w. Quadratic functions are convex, so $\hat{L}(w)$ is convex.

D. **False.** This statement contradicts statement C. Since $\hat{L}(w)$ is convex, it is not true that it is non-convex.

Question 3. [1 MARK]

Let everything be defined as in the previous question. Now we consider regularized polynomial regression. The regularized empirical loss is defined as

$$\hat{L}_\lambda(w) = \frac{1}{n}\sum_{i=1}^{n}\left(\phi_p(x_i)^T w - y_i\right)^2 + \frac{\lambda}{n}\|w\|^2,$$

where $\lambda > 0$ is the regularization parameter. Which of the following statements is true?

A. The optimization problem $\min_{w\in\mathbb{R}^p} \hat{L}_\lambda(w)$ does not have a closed form solution, since $\hat{L}_\lambda$ is not convex.

B. The regularization term $\frac{\lambda}{n}\|w\|^2$ penalizes large weights, which helps to prevent overfitting.

C. The gradient descent update rule for the regularized loss is

$$w^{(t+1)} = w^{(t)} - \eta^{(t)}\left[\frac{2}{n}\sum_{i=1}^{n}\left(\phi_p(x_i)^T\phi_p(w^{(t)}) - y_i\right)\phi_p(x_i) + \frac{2\lambda}{n}\phi_p(w^{(t)})\right].$$

D. The gradient descent update rule for the regularized loss is

$$w^{(t+1)} = w^{(t)} - \eta^{(t)}\left[\frac{2}{n}\sum_{i=1}^{n}\left(\phi_p(x_i)^T w^{(t)} - y_i\right)\phi_p(x_i) + \frac{2\lambda}{n}w^{(t)}\right].$$

Solution 3. **Correct answer(s): B, D**

Explanation:

A. **False.** The regularized optimization problem is quadratic in w and therefore has a closed form solution (the ridge regression solution).

B. **True.** The regularization term $\frac{\lambda}{n}\|w\|^2$ penalizes large weights, which helps to prevent overfitting and improves the conditioning of the problem.

C. **False.** The update rule in this statement mistakenly applies the feature map ϕ_p to $w^{(t)}$ (i.e., $\phi_p(w^{(t)})$). The gradient descent update should involve the weight vector $w^{(t)}$ directly.

D. **True.** The correct gradient descent update rule for the regularized loss is

$$w^{(t+1)} = w^{(t)} - \eta^{(t)}\left[\frac{2}{n}\sum_{i=1}^{n}\left(\phi_p(x_i)^T w^{(t)} - y_i\right)\phi_p(x_i) + \frac{2\lambda}{n}w^{(t)}\right].$$

Question 4. [1 MARK]

Consider a convex optimization problem solved by gradient descent $\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta^{(t)} \nabla g\big(\mathbf{w}^{(t)}\big)$. Which of the following statements is true?

A. Exponential decaying step sizes, defined by $\eta^{(t)} = \eta_0 \exp(-\lambda t)$, reduce the learning rate more rapidly than inverse decaying step sizes, defined by $\eta^{(t)} = \frac{\eta_0}{1+t}$.

B. The normalized gradient step size, that is defined by $\eta^{(t)} = \frac{\eta}{\epsilon + ||\nabla g(\mathbf{w}^{(t)})||}$, accelerates convergence by removing the influence of the gradient's magnitude from the update direction.

C. The normalized gradient step size, that is defined by $\eta^{(t)} = \frac{\eta}{\epsilon + ||\nabla g(\mathbf{w}^{(t)})||}$, avoids overshooting the minimum by inversly proportionally adapting the step size with respect to the gradient magnitude (gradient magnitude large $\rightarrow$ small step size and vice versa).

D. A constant step size, $\eta^{(t)} = \eta_0$, is generally preferred over any decaying schedule because it maintains consistent update magnitudes throughout the iterations.

Solution 4. Correct answer(s): A, C

Explanation:

A. **True.** Exponential decaying step sizes, given by

$$\eta^{(t)} = \eta_0 \exp(-\lambda t),$$

reduce the learning rate more rapidly than inverse decaying step sizes,

$$\eta^{(t)} = \frac{\eta_0}{1+t},$$

so they lead to a faster decrease in update magnitude as iterations increase.

B. **False.** While the normalized gradient step size,

$$\eta^{(t)} = \frac{\eta}{\epsilon + \|\nabla g(\mathbf{w}^{(t)})\|},$$

removes the influence of the gradient's magnitude, it does not inherently accelerate convergence; its primary benefit is to stabilize updates and prevent overshooting.

C. **True.** The normalized gradient step size adapts inversely with the gradient magnitude (large gradient $\rightarrow$ smaller step, small gradient $\rightarrow$ larger step), effectively avoiding overshooting the minimum.

D. **False.** A constant step size, $\eta^{(t)} = \eta_0$, does maintain consistent update magnitudes, but it does not adapt to the local curvature of the loss surface and may lead to oscillations or divergence, making decaying schedules generally more effective in convex optimization.

Question 5. [1 MARK]

The Poisson distribution is a discrete probability distribution that models the number of events occurring in a fixed interval. Its probability mass function (pmf) is given by

$$p(k;\lambda) = \frac{\lambda^k e^{-\lambda}}{k!},$$

for $k = 0, 1, 2, \ldots$, where $a! = a \times (a-1) \times \cdots \times 1$ is the factorial function.
Now suppose we have data $D = (X_1, X_2, X_3) = (3, 2, 4)$, where each X_i is independently drawn from a Poisson distribution with parameter λ. We want to estimate λ using maximum likelihood estimation (MLE).
Which of the following statements is true?

A. The likelihood function is
$$\frac{\lambda^9 e^{-3\lambda}}{(3!)^2 \cdot 4!}.$$

B. The likelihood function is
$$\frac{\lambda^9 e^{-3\lambda}}{3! \cdot 2! \cdot 4!}.$$

C. The likelihood function is
$$\frac{\lambda^8 e^{-3\lambda}}{3! \cdot 2! \cdot 4!}.$$

D. The likelihood function is
$$\frac{\lambda^9 e^{-2\lambda}}{3! \cdot 2! \cdot 4!}.$$

Solution 5. Correct answer(s): B

Explanation:

A. **False.** For data $D = (3, 2, 4)$, the likelihood is the product of the individual Poisson pmfs:

$$\frac{\lambda^3 e^{-\lambda}}{3!} \cdot \frac{\lambda^2 e^{-\lambda}}{2!} \cdot \frac{\lambda^4 e^{-\lambda}}{4!} = \frac{\lambda^{3+2+4} e^{-3\lambda}}{3! \cdot 2! \cdot 4!} = \frac{\lambda^9 e^{-3\lambda}}{3! \cdot 2! \cdot 4!}.$$

Option A incorrectly uses $(3!)^2$ instead of $3! \cdot 2!$.

B. **True.** The correct likelihood function is obtained by multiplying the pmfs:

$$\frac{\lambda^3 e^{-\lambda}}{3!} \cdot \frac{\lambda^2 e^{-\lambda}}{2!} \cdot \frac{\lambda^4 e^{-\lambda}}{4!} = \frac{\lambda^9 e^{-3\lambda}}{3! \cdot 2! \cdot 4!}.$$

C. **False.** Option C uses an exponent of 8 on λ instead of 9, which does not reflect the total count ($3 + 2 + 4 = 9$).

D. **False.** Option D has the correct exponent of 9 on λ but the exponential term $e^{-2\lambda}$ is incorrect; it should be $e^{-3\lambda}$ to account for three independent observations.

Question 6. [1 MARK]

Let everything be defined as in the previous question regarding the Poisson distribution. Recall the logarithm property that $\log(x^a) = a\log(x)$ and $\log(\frac{a}{b}) = \log(a) - \log(b)$. Using maximum likelihood estimation (MLE), which of the following statements is true regarding the MLE of λ?

A. The maximum likelihood estimate of λ is $\lambda_{\text{MLE}} = 3$.

B. The maximum likelihood estimate of λ is $\lambda_{\text{MLE}} = \frac{9}{2}$.

C. The maximum likelihood estimate of λ is $\lambda_{\text{MLE}} = \frac{3}{4}$.

D. The maximum likelihood estimate of λ is $\lambda_{\text{MLE}} = 4$.

Solution 6. Correct answer(s): A

Explanation:

A. **True.** For independent Poisson observations X_1, X_2, X_3 with parameter λ, the likelihood is

$$\prod_{i=1}^{3} \frac{\lambda^{X_i} e^{-\lambda}}{X_i!}.$$

Taking the logarithm and applying the logarithm properties, we obtain:

$$\log L(\lambda) = \big(X_1 + X_2 + X_3\big) \log \lambda - 3\lambda - \log\big(X_1! X_2! X_3!\big).$$

Differentiating with respect to λ yields:

$$\frac{d}{d\lambda} \log L(\lambda) = \frac{X_1 + X_2 + X_3}{\lambda} - 3.$$

Setting this derivative to zero gives:

$$\frac{X_1 + X_2 + X_3}{\lambda} - 3 = 0 \quad \Longrightarrow \quad \lambda = \frac{X_1 + X_2 + X_3}{3}.$$

With data $D = (3, 2, 4)$, we have $X_1 + X_2 + X_3 = 3 + 2 + 4 = 9$, so

$$\lambda_{\text{MLE}} = \frac{9}{3} = 3.$$

Therefore, option A is correct.

B. **False.** Option B states $\lambda_{\text{MLE}} = \frac{9}{2}$, which does not match the calculated value.

C. **False.** Option C suggests $\lambda_{\text{MLE}} = \frac{3}{4}$, which is not consistent with the sample average of the counts.

D. **False.** Option D claims $\lambda_{\text{MLE}} = 4$, which is also inconsistent with the derived value of 3.

Question 7. [1 MARK]

Suppose we want to model the number of lightnings appearing in Edmonton each year using a Poisson distribution that is governed by the parameter λ. We place a Gamma prior on λ such that $\lambda \sim \text{Gamma}(a, b)$, with probability density function given by

$$p(\lambda) \propto \lambda^{a-1} e^{-b\lambda},$$

where $\propto$ means proportional to, that is, excluding the constants that do not depend on λ.
We now observe f_1, f_2 and f_3 lightnings in year one, two and three resulting in a likelihood function that is proportional to $\lambda^{f_1+f_2+f_3} e^{-3\lambda}$. What is the posterior distribution of λ given this data?

A. $p(\lambda|\mathcal{D}) \propto \lambda^{f_1+f_2+f_3-a+1} e^{-(b-3)\lambda}$.

B. $p(\lambda|\mathcal{D}) \propto \lambda^{a-1-f_1-f_2-f_3} e^{-(3-b)\lambda}$.

C. $p(\lambda|\mathcal{D}) \propto \lambda^{f_1+f_2+f_3+a-1} e^{-b\lambda}$.

D. $p(\lambda|\mathcal{D}) \propto \lambda^{a-1+f_1+f_2+f_3} e^{-(b+3)\lambda}$.

Solution 7. Correct answer(s): D

Explanation:

A. **False.** Option B rearranges the exponent on λ incorrectly (writing $f_1+f_2+f_3-a+1$ instead of $a-1+f_1+f_2+f_3$) and uses $e^{-(b-3)\lambda}$, which does not account properly for the likelihood's $e^{-3\lambda}$ factor.

B. **False.** Option C incorrectly subtracts the sum $f_1 + f_2 + f_3$ from $a - 1$ and reverses the sign in the exponential term.

C. **False.** Option D omits the likelihood's contribution to the exponential term (i.e. the -3λ), leaving the exponential factor as $e^{-b\lambda}$ rather than $e^{-(b+3)\lambda}$.

D. **True.** The Gamma prior is given by

$$p(\lambda) \propto \lambda^{a-1} e^{-b\lambda},$$

and each Poisson observation with parameter λ has a likelihood proportional to

$$\frac{\lambda^{f_i} e^{-\lambda}}{f_i!}.$$

Since the data $D = (f_1, f_2, f_3)$ are independent, the likelihood function is proportional to

$$\lambda^{f_1+f_2+f_3} e^{-3\lambda}.$$

Multiplying the prior and the likelihood gives

$$p(\lambda \mid D) \propto \lambda^{a-1} e^{-b\lambda} \cdot \lambda^{f_1+f_2+f_3} e^{-3\lambda} = \lambda^{a-1+f_1+f_2+f_3} e^{-(b+3)\lambda}.$$

Question 8. [1 MARK]

Suppose we want to perform ridge regression, where the data is generated from a Gaussian distribution with mean $\mathbf{x}^T\mathbf{w}$ and variance 1. For the bias term w_0, we assume a Gaussian prior with zero mean and a very large variance a: $p(w_0) = \sqrt{\frac{1}{2\pi a}}\,\exp\left(-\frac{w_0^2}{2a}\right)$. For each regression weight w_j for $j = 1, \ldots, d$, we assume a Gaussian prior with zero mean and variance $1/\lambda$: $p(w_j) = \sqrt{\frac{\lambda}{2\pi}}\,\exp\left(-\frac{\lambda}{2}w_j^2\right)$, where $\lambda \geq 0$ is the regularization parameter. All weights $w_0, w_1, \ldots, w_d$ are independent. The MAP estimate of the weights is given by

$$\mathbf{w}_{\text{MAP}} = \arg\min_{\mathbf{w}} \left\{ \frac{1}{2}\sum_{i=1}^{n}(y_i - \mathbf{x}_i^T\mathbf{w})^2 - \log\big(p(\mathbf{w})\big) \right\}.$$

What is the expression for $-\log(p(\mathbf{w}))$?

A. $-\frac{1}{2}\log(\frac{1}{2\pi a}) - \frac{d}{2}\log\left(\frac{\lambda}{2\pi}\right) + \frac{w_0^2}{2a} + \frac{\lambda}{2}\sum_{j=1}^{d} w_j^2$.

B. $-\frac{1}{2}\log(\frac{1}{2\pi a}) - \frac{d}{2}\log\left(\frac{\lambda}{2\pi}\right) + \frac{1}{2a} + \frac{\lambda}{2}\sum_{j=1}^{d} w_j^2$.

C. $\frac{1}{2}\log(\frac{1}{2\pi a}) + \frac{d}{2}\log\left(\frac{\lambda}{2\pi}\right) + \frac{w_0^2}{2a} + \frac{\lambda}{2}\sum_{j=1}^{d} w_j^2$.

D. $-\frac{1}{2}\log(\frac{1}{2\pi a}) + \frac{d}{2}\log\left(\frac{\lambda}{2\pi}\right) - \frac{1}{2a} - \frac{\lambda}{2}\sum_{j=1}^{d} w_j^2$.

Solution 8. Correct answer(s): A

Explanation:
We assume the following priors:

$$p(w_0) = \sqrt{\frac{1}{2\pi a}}\,\exp\left(-\frac{w_0^2}{2a}\right)$$

and for $j = 1, \ldots, d$,

$$p(w_j) = \sqrt{\frac{\lambda}{2\pi}}\,\exp\left(-\frac{\lambda}{2}w_j^2\right),$$

with $\lambda \geq 0$ and all weights independent. Also note that both priors assume zero mean.
For the bias term w_0, we have:

$$\log p(w_0) = \frac{1}{2}\log\left(\frac{1}{2\pi a}\right) - \frac{w_0^2}{2a}.$$

Taking the negative logarithm yields:

$$-\log p(w_0) = -\frac{1}{2}\log\left(\frac{1}{2\pi a}\right) + \frac{w_0^2}{2a}.$$

For each regression weight w_j, we have:

$$\log p(w_j) = \frac{1}{2}\log\left(\frac{\lambda}{2\pi}\right) - \frac{\lambda}{2}w_j^2.$$

Thus,

$$-\log p(w_j) = -\frac{1}{2}\log\left(\frac{\lambda}{2\pi}\right) + \frac{\lambda}{2}w_j^2.$$

Since all weights are independent, the joint negative log-prior is the sum:

$$-\log(p(\mathbf{w})) = -\log\big(p(w_0)\big) + \sum_{j=1}^{d}\left[-\log\big(p(w_j)\big)\right].$$

Substituting the above expressions:

$$-\log(p(\mathbf{w})) = \left[-\frac{1}{2}\log\left(\frac{1}{2\pi a}\right) + \frac{w_0^2}{2a}\right] + \sum_{j=1}^{d}\left[-\frac{1}{2}\log\left(\frac{\lambda}{2\pi}\right) + \frac{\lambda}{2}w_j^2\right].$$

This simplifies to:

$$-\log(p(\mathbf{w})) = -\frac{1}{2}\log\left(\frac{1}{2\pi a}\right) - \frac{d}{2}\log\left(\frac{\lambda}{2\pi}\right) + \frac{w_0^2}{2a} + \frac{\lambda}{2}\sum_{j=1}^{d}w_j^2.$$

Now we evaluate each option:

A. **True.** This option exactly matches the derived expression:

$$-\frac{1}{2}\log\left(\frac{1}{2\pi a}\right) - \frac{d}{2}\log\left(\frac{\lambda}{2\pi}\right) + \frac{w_0^2}{2a} + \frac{\lambda}{2}\sum_{j=1}^{d}w_j^2.$$

B. **False.** This option incorrectly replaces $\frac{w_0^2}{2a}$ with $\frac{1}{2a}$, removing the dependence on w_0.

C. **False.** This option has the opposite signs for the logarithmic (constant) terms, which would correspond to $\log(p(\mathbf{w}))$ rather than $-\log(p(\mathbf{w}))$.

D. **False.** This option negates the quadratic terms (i.e. $-\frac{w_0^2}{2a}$ and $-\frac{\lambda}{2}\sum_{j=1}^{d}w_j^2$).

Question 9. [1 MARK]

Let the dataset be $\mathcal{D} = ((x_1, y_1), \ldots, (x_n, y_n))$, the mini-batch size $b \in \mathbb{N}$, and $M = \text{floor}(n/b)$ is the number of full batches. In class we learned about mini-batch gradient descent (MBGD). In this question we are interested in the computational efficiency of MBGD compared to batch gradient descent. Which of the following statements are true?

A. MBGD is more efficient than batch gradient descent because it requires less computation for the same number of epochs.

B. MBGD is more efficient than batch gradient descent because the gradient estimate is more precise.

C. MBGD usually finds a better solution in the same number of epochs compared to batch gradient descent.

D. MBGD is often more efficient than batch gradient descent because it updates the weights more frequently.

Solution 9. Correct answer(s): C, D

Explanation:

A. **False.** Mini-batch gradient descent (MBGD) is computationally identical for the same number of epochs T.

B. **False.** MBGD uses a mini-batch of data, so its gradient estimates are inherently noisier and less precise than the gradient computed using the full dataset in batch gradient descent.

C. **True.** This statement is true.

D. **True.** MBGD updates the weights after processing each mini-batch, leading to more frequent weight updates. This increased frequency can help accelerate convergence.

Question 10. [1 MARK]

Let everything be defined as in the previous question. Which of the following statements are true?

A. If n is divisible by b then there are M mini-batches.

B. If n is not divisible by b then the size of the last mini-batch is $n - Mb$.

C. If n is divisible by b then the estimated loss based on the last mini-batch is

$$\frac{1}{b} \sum_{i=(M-1)b+1}^{Mb} \ell(f(\mathbf{x}_i), y_i).$$

D. The variance of the estimated loss for each mini-batch (not considering the last batch) increases if b decreases.

Solution 10. Correct answer(s): A, B, C, D

Explanation:

A. **True.** When n is divisible by b, we have $n = Mb$ exactly. Since $M = \text{floor}(n/b)$, this means there are exactly M mini-batches.

B. **True.** If n is not divisible by b, then the number of full mini-batches is $M = \text{floor}(n/b)$, and the remaining samples form the last mini-batch, which has size $n - Mb$.

C. **True.** When n is divisible by b, every mini-batch (including the last one) has exactly b examples. Therefore, the estimated loss for the last mini-batch is the average over its b samples:

$$\frac{1}{b} \sum_{i=(M-1)b+1}^{Mb} \ell(f(\mathbf{x}_i), y_i),$$

which is exactly as stated.

D. **True.** The variance of the estimated loss (i.e., the sample mean of the loss over a mini-batch) is inversely proportional to the mini-batch size b. Hence, as b decreases, the variance of the estimate increases.

Question 11. [1 MARK]

Let everything be defined as in the previous two questions. Your friend is trying to implement the version of mini-batch gradient descent discussed in the previous two questions with a constant step size. They have written the following pseudocode and asked you to review it. Please select all possible mistakes in the pseuodocode below. Multiple statements may be correct.

Algorithm 1: MBGD Linear Regression Learner (with a constant step size and last mini-batch)

1: **input:** $\mathcal{D} = ((\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n))$, step size η, number of epochs T, mini-batch size b
2: $\mathbf{w} \leftarrow$ random vector in $\mathbb{R}^{d+1}$
3: $M \leftarrow \text{floor}\left(\frac{n}{b}\right)$
4: **for** $m = 1, \ldots, M$ **do**
5: randomly shuffle $\mathcal{D}$
6: **for** $t = 1, \ldots, T$ **do**
7: $\nabla \hat{L}(\mathbf{w}) \leftarrow \frac{2}{b} \sum_{i=(m-1)b+1}^{mb} (\mathbf{x}_i^\top \mathbf{w} - y_i)\mathbf{x}_i$
8: $\mathbf{w} \leftarrow \mathbf{w} - \eta \nabla \hat{L}(\mathbf{w})$
9: **if** $n > Mb$ **then**
10: $\nabla \hat{L}(\mathbf{w}) \leftarrow \frac{2}{n-Mb} \sum_{i=Mb+1}^{n} (\mathbf{x}_i^\top \mathbf{w} - y_i)\mathbf{x}_i$
11: $\mathbf{w} \leftarrow \mathbf{w} - \eta \nabla \hat{L}(\mathbf{w})$
12: **return** $\hat{f}(\mathbf{x}) = \mathbf{x}^\top \hat{\mathbf{w}}$

A. There are no mistakes. The pseudocode is correct.

B. The pseudocode is incorrect because the outer loop (line 4) should iterate over t and the inner loop (line 6) over m.

C. The pseudocode is incorrect because the gradient calculation for the last mini-batch is incorrect.

D. The pseudocode is incorrect because we update the parameter twice: in lines 8 and 11.

Solution 11. Correct answer(s): B

Explanation:

A. **False.** The statement that "there are no mistakes" is incorrect because the pseudocode contains an error in the loop structure.

B. **True.** In standard mini-batch gradient descent, the outer loop should iterate over the number of epochs (indexed by t) and the inner loop should iterate over the mini-batches (indexed by m). In the given pseudocode, the outer loop iterates over mini-batches and the inner loop over epochs, which is not the conventional implementation and can lead to inefficient training.

C. **False.** The gradient calculation for the last mini-batch is computed as

$$\frac{2}{n - Mb} \sum_{i=Mb+1}^{n} (\mathbf{x}_i^\top \mathbf{w} - y_i)\mathbf{x}_i,$$

which correctly averages the gradients over the remaining samples. This is not a mistake.

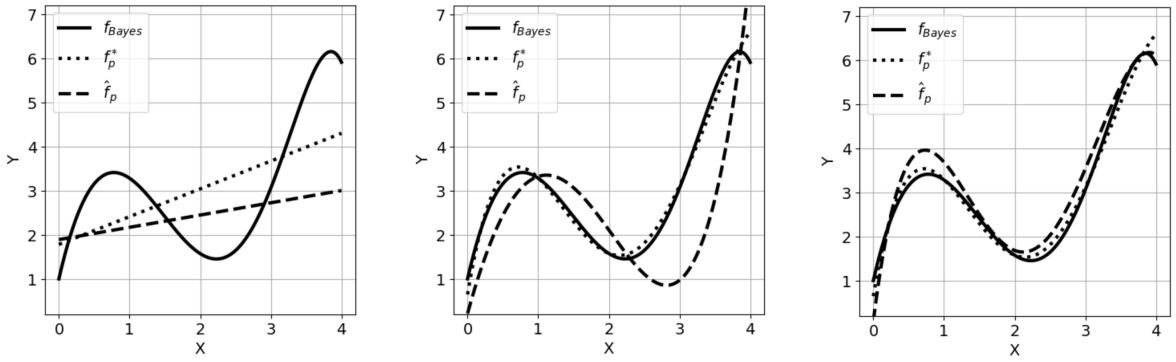
D. **False.** The pseudocode updates the parameter vector in two different contexts: within the inner loop for each full mini-batch, and then once more for the final (incomplete) mini-batch if n is not divisible by b. These updates are intentional and necessary, not a mistake.

Question 12. [1 MARK]

Let ϕ_p be the polynomial feature map of degree p, and $\mathcal{F}_p$ the function class containing all polynomials of degree p or less.
Recall that

$$f_{\text{Bayes}} = \underset{f \in \{f \mid f:\mathbb{R}^{d+1} \to \mathbb{R}\}}{\arg\min} L(f), \quad f_p^* = \underset{f \in \mathcal{F}_p}{\arg\min}\, L(f), \quad \hat{f}_p = \underset{f \in \mathcal{F}_p}{\arg\min}\, \hat{L}(f).$$

Below are plots for different values of p and dataset size n. Which of the following statements are true?



(a) Fig 1 (b) Fig 2 (c) Fig 3

A. The data set sizes n does not affect f_p^*.

B. The data set sizes n does not affect f_{Bayes}.

C. For very large data set sizes n, $\hat{f}_p$ in Fig 1 will never be closer to f_{Bayes} than $\hat{f}_p$ in Fig 2, assuming that the polynomial degree p in Fig 1 is smaller than in Fig 2.

D. That $\hat{f}_p$ is much closer to f_p^* in Fig 3 than in Fig 2 might stem from a larger dataset size n.

Solution 12. Correct answer(s): A, B, D

Explanation:

A. **True.** The dataset size n does not affect f_p^*, which is defined as

$$f_p^* = \arg\min_{f \in \mathcal{F}_p} L(f).$$

This is the best possible predictor in the class of polynomials of degree p (i.e., $\mathcal{F}_p$), with respect to the true loss L. The definition of f_p^* does not involve the dataset size n, since it is purely an idealized concept (i.e., we assume knowledge of the true loss).

B. **True.** Similarly,

$$f_{\text{Bayes}} = \arg\min_{f | f: \mathbb{R}^{d+1} \to \mathbb{R}} L(f)$$

is the absolute best function in the space of all possible functions with respect to the true loss L. It depends only on the underlying data-generating mechanism (the "true" function), not on any particular dataset size n.

C. **False.** Although having a higher polynomial degree typically allows $\hat{f}_p$ to capture more complex patterns, it is not strictly impossible for a smaller-degree polynomial (with a very large dataset) to approximate f_{Bayes} better than a higher-degree polynomial (with potentially suboptimal fitting). In practice, we can see overfitting or poor fitting with higher-degree polynomials if the dataset is not used effectively or if other factors (like regularization) come into play. Thus, it is not guaranteed that a small-degree polynomial with a huge dataset can *never* surpass a higher-degree polynomial's fit.

D. **True.** In Fig 3, $\hat{f}_p$ is much closer to f_p^* than in Fig 2. One likely explanation is that the dataset size n is larger, giving a more accurate empirical estimate of the best polynomial parameters. With more data, $\hat{f}_p$ can more closely approach f_p^*.

Question 13. [1 MARK]

Let everything be defined as in the previous question. Which of the following statements are true?

A. The remaining mismatch between f_p^* and f_{Bayes} in Fig 3 is due to the irreducable error.

B. Increasing the polynomial degree p will reduce the approximation and the estimation error.

C. The underlying function that generated the data can be less well represented by quadratic functions than by the function class chosen in Fig 2.

D. $\hat{f}_p$ would match f_{Bayes} perfectly if the approximation and the estimation error are both zero.

Solution 13. Correct answer(s): C, D

Explanation:

A. **False.** The mismatch between f_p^* and f_{Bayes} arises from *approximation error*, not the irreducible error. The irreducible error corresponds to randomness or noise in the data-generation process, whereas the approximation error stems from the limited expressiveness of polynomials of degree p.

B. **False.** Increasing the polynomial degree p typically reduces the *approximation error* (i.e., the gap between f_p^* and f_{Bayes}), but it often *increases* the *estimation error* (i.e., the gap between $\hat{f}_p$ and f_p^*), especially when the dataset size is not large enough to reliably estimate more parameters.

C. **True.** A quadratic function (degree 2) may be insufficient to capture the complex shape of the true data-generating function, whereas a higher-degree polynomial (as in Fig 2) is better suited to represent such a wavy structure.

D. **True.** If both approximation error (the difference between f_p^* and f_{Bayes}) and estimation error (the difference between $\hat{f}_p$ and f_p^*) are zero, then $\hat{f}_p$ coincides perfectly with f_{Bayes}. This would mean we have a function class that exactly matches the true function and enough data (and perfect estimation) to recover it exactly.

Question 14. [1 MARK]

You have access to the true feature-label distribution $\mathbb{P}_{\mathbf{X},Y}$ that generated the data. You are interested in training a model and impatiently start plotting the following figure and start trying to make sense of it. Which of the following statements are true?

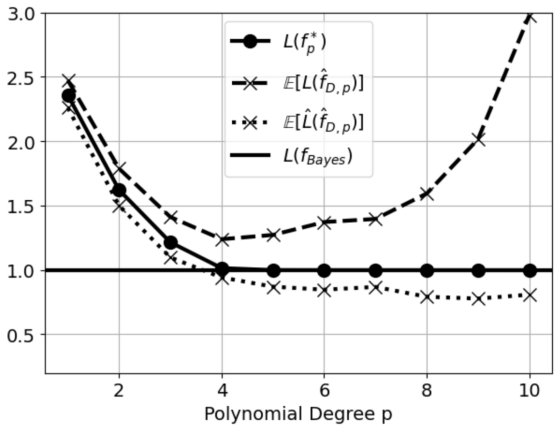


Fig 1

A. $E[\hat{L}(\hat{f}_{D,p})]$ decreases to a lower value than $L(f_{\text{Bayes}})$ for larger p because $\hat{f}_{D,p}$ fits the dataset it was trained on better than f_{Bayes}.

B. f_{Bayes} is a better predictor than $\hat{f}_{D,p}$ for the true feature-label distribution.

C. The expected value of $E[\hat{L}(\hat{f}_{D,p})]$ is calulated with respect to true feature-label distribution $\mathbb{P}_{\mathbf{X},Y}$.

D. The expected value of $E[\hat{L}(\hat{f}_{D,p})]$ is calulated with respect to distribution over the polynomial degree $\mathbb{P}_p$.

Solution 14. Correct answer(s): A, B, C

Explanation:

A. **True.** In the figure, for sufficiently large p, the plotted value of $E[\hat{L}(\hat{f}_{D,p})]$ does indeed drop below $L(f_{\text{Bayes}})$. This can happen because the plotted loss $\hat{L}$ is the *empirical* loss on the training data, rather than the true expected loss over the data-generating distribution. A sufficiently flexible model $\hat{f}_{D,p}$ can overfit its training set. Hence, $\hat{f}_{D,p}$ can "fit the dataset it was trained on better" than f_{Bayes} even though f_{Bayes} is optimal with respect to the true distribution.

B. **True.** By definition,

$$f_{\text{Bayes}} = \arg\min_f L(f)$$

is the predictor that achieves the smallest possible *true* loss over the actual data-generating process. No finite-sample model $\hat{f}_{D,p}$ can consistently beat f_{Bayes} in terms of *true* expected loss, even though it may achieve lower in-sample loss (as per statement A).

C. **True.** The expectation $E[\hat{L}(\hat{f}_{D,p})]$ is taken with respect to new samples drawn from the true feature-label distribution $\mathbb{P}_{\mathbf{X},Y}$. Formally,

$$E[\hat{L}(\hat{f}_{D,p})] \;=\; \int \hat{L}(\hat{f}_{D,p})\, d\mathbb{P}_{\mathbf{X},Y}.$$

This means we consider how $\hat{f}_{D,p}$ performs on data drawn from the underlying process that generated the dataset.

D. **False.** There is no distribution $\mathbb{P}_p$ over the polynomial degree p. The index p is a fixed hyperparameter (the model complexity), not a random variable. Hence, we do not take an expectation over $\mathbb{P}_p$.

Question 15. [1 MARK]

Let ϕ_p be the polynomial feature map of degree p. The function class containing all polynomials of degree p or less is

$$\mathcal{F}_p = \left\{ f \;\middle|\; f : \mathbb{R}^{d+1} \to \mathbb{R}, \text{ and } f(\mathbf{x}) = \phi_p(\mathbf{x})^\top \mathbf{w} \text{ for some } \mathbf{w} \in \mathbb{R}^{\bar{p}} \right\}.$$

Which of the following statements is true?

A. The dimension of $\phi_p(\mathbf{x})$ is $\bar{p} = \binom{p+d}{d}$.

B. $\mathcal{F}_p \subseteq \mathcal{F}_{p+1}$.

C. The dimension of $\mathbf{w}$ is $d+1$.

D. $\mathcal{F}_{25}$ contains exponential functions.

Solution 15. Correct answer(s): A, B

Explanation:

A. **True.** By definition, the polynomial feature map ϕ_p of degree p maps $\mathbf{x}$ into a vector whose dimension is

$$\bar{p} = \binom{p+d}{d},$$

which counts the number of monomials (including the constant term) of degree at most p in d variables.

B. **True.** Every polynomial of degree p is also a polynomial of degree $p+1$ (by setting the coefficients of the extra terms to zero). Hence, the function class $\mathcal{F}_p$ is a subset of $\mathcal{F}_{p+1}$.

C. **False.** The weight vector $\mathbf{w}$ in the representation $f(\mathbf{x}) = \phi_p(\mathbf{x})^\top \mathbf{w}$ belongs to $\mathbb{R}^{\bar{p}}$, not $\mathbb{R}^{d+1}$. In general, $\bar{p} = \binom{p+d}{d}$ is larger than $d+1$ when $p > 1$.

D. **False.** The function class $\mathcal{F}_{25}$ consists only of polynomials of degree at most 25. Exponential functions are not polynomials, so they are not contained in $\mathcal{F}_{25}$.