CS 6140: Machine Learning
Homework 1

Ashish Dasu

Due: February 5, 2026

Problem 1

Problem: Show that the complexity of computing AB where $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{n \times p}$ is $O(mnp)$.

Solution:

Let $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{n \times p}$. The matrix product $C = AB$ results in a matrix $C \in \mathbb{R}^{m \times p}$.

Each entry c_{ij} of the matrix C is computed as:

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj} \tag{1}$$

This is the inner product of the i-th row of A with the j-th column of B. From class, we know that computing an inner product between two n-dimensional vectors requires $O(n)$ operations (specifically, n multiplications and $n - 1$ additions).

The matrix C has $m \times p$ entries, and each entry requires $O(n)$ operations to compute. Therefore, the total number of operations is:

$$\text{Total operations} = (m \times p) \times O(n) = O(mnp) \tag{2}$$

Thus, the complexity of computing AB is $O(mnp)$.

Problem 2

Problem: Show that $\|X\theta - Y\|_1 = \sum_{i=1}^{N} |\theta^\top x_i - y_i|$ where $X \in \mathbb{R}^{N \times d}$ has rows corresponding to samples x_i for $i = 1, \ldots, N$, and $Y \in \mathbb{R}^N$ has entries corresponding to responses y_i for $i = 1, \ldots, N$.

Solution:

Let $X \in \mathbb{R}^{N \times d}$ be a matrix whose rows correspond to samples $x_i^\top$ for $i = 1, \ldots, N$. Similarly, $Y \in \mathbb{R}^N$ is a vector whose entries correspond to responses y_i for $i = 1, \ldots, N$. We can write:

$$X = \begin{bmatrix} x_1^\top \\ x_2^\top \\ \vdots \\ x_N^\top \end{bmatrix}, \quad Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \tag{3}$$

The product $X\theta$ gives us:

$$X\theta = \begin{bmatrix} x_1^\top \theta \\ x_2^\top \theta \\ \vdots \\ x_N^\top \theta \end{bmatrix} \tag{4}$$

Therefore, $X\theta - Y$ is:

$$X\theta - Y = \begin{bmatrix} x_1^\top \theta - y_1 \\ x_2^\top \theta - y_2 \\ \vdots \\ x_N^\top \theta - y_N \end{bmatrix} \tag{5}$$

By definition, the ℓ_1 norm of a vector $v \in \mathbb{R}^N$ is:

$$\|v\|_1 = \sum_{i=1}^{N} |v_i| \tag{6}$$

Applying this definition to $X\theta - Y$:

$$\|X\theta - Y\|_1 = \sum_{i=1}^{N} |(X\theta - Y)_i| = \sum_{i=1}^{N} |x_i^\top \theta - y_i| = \sum_{i=1}^{N} |\theta^\top x_i - y_i| \tag{7}$$

where the last equality uses the fact that $x_i^\top \theta = \theta^\top x_i$ (inner product is commutative).

Problem 3

Problem: Let $X \in \mathbb{R}^{2\times 2}$ and $Y \in \mathbb{R}^{2\times 2}$ be two arbitrary matrices. Also let $\theta \in \mathbb{R}^2$ denote a vector. Show the following results and provide all steps of your derivations.

Part 1

Show that $\frac{\partial \operatorname{tr}(Y^\top X)}{\partial X} = Y$.

Solution:

We want to show that $\frac{\partial \operatorname{tr}(Y^\top X)}{\partial X} = Y$ where $X, Y \in \mathbb{R}^{2\times 2}$.

Let $X = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix}$ and $Y = \begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix}$.

First, compute $Y^\top X$:

$$Y^\top X = \begin{bmatrix} y_{11} & y_{21} \\ y_{12} & y_{22} \end{bmatrix} \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix} = \begin{bmatrix} y_{11}x_{11} + y_{21}x_{21} & y_{11}x_{12} + y_{21}x_{22} \\ y_{12}x_{11} + y_{22}x_{21} & y_{12}x_{12} + y_{22}x_{22} \end{bmatrix} \tag{8}$$

The trace is the sum of diagonal elements:

$$\operatorname{tr}(Y^\top X) = y_{11}x_{11} + y_{21}x_{21} + y_{12}x_{12} + y_{22}x_{22} \tag{9}$$

Now, take the derivative with respect to each element of X:

$$\frac{\partial \operatorname{tr}(Y^\top X)}{\partial x_{11}} = y_{11} \tag{10}$$

$$\frac{\partial \operatorname{tr}(Y^\top X)}{\partial x_{12}} = y_{12} \tag{11}$$

$$\frac{\partial \operatorname{tr}(Y^\top X)}{\partial x_{21}} = y_{21} \tag{12}$$

$$\frac{\partial \operatorname{tr}(Y^\top X)}{\partial x_{22}} = y_{22} \tag{13}$$

Therefore, the derivative matrix is:

$$\frac{\partial \operatorname{tr}(Y^\top X)}{\partial X} = \begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix} = Y \tag{14}$$

Part 2

Show that $\frac{\partial \theta^\top X\theta}{\partial \theta} = (X + X^\top)\theta$.

Solution:

We want to show that $\frac{\partial \theta^\top X\theta}{\partial \theta} = (X + X^\top)\theta$ where $X \in \mathbb{R}^{2\times 2}$ and $\theta \in \mathbb{R}^2$.

Let $\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$ and $X = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix}$.

First, compute $X\theta$:

$$X\theta = \begin{bmatrix} x_{11}\theta_1 + x_{12}\theta_2 \\ x_{21}\theta_1 + x_{22}\theta_2 \end{bmatrix} \tag{15}$$

Now compute $\theta^\top X\theta$:

$$\theta^\top X\theta = \theta_1(x_{11}\theta_1 + x_{12}\theta_2) + \theta_2(x_{21}\theta_1 + x_{22}\theta_2) \tag{16}$$

Expanding:

$$\theta^\top X\theta = x_{11}\theta_1^2 + x_{12}\theta_1\theta_2 + x_{21}\theta_1\theta_2 + x_{22}\theta_2^2 \tag{17}$$

Taking derivatives with respect to each component of θ:

$$\frac{\partial(\theta^\top X\theta)}{\partial\theta_1} = 2x_{11}\theta_1 + x_{12}\theta_2 + x_{21}\theta_2 \tag{18}$$

$$\frac{\partial(\theta^\top X\theta)}{\partial\theta_2} = x_{12}\theta_1 + x_{21}\theta_1 + 2x_{22}\theta_2 \tag{19}$$

Therefore:

$$\frac{\partial\theta^\top X\theta}{\partial\theta} = \begin{bmatrix} 2x_{11}\theta_1 + x_{12}\theta_2 + x_{21}\theta_2 \\ x_{12}\theta_1 + x_{21}\theta_1 + 2x_{22}\theta_2 \end{bmatrix} \tag{20}$$

Now compute $(X + X^\top)\theta$:

$$X + X^\top = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix} + \begin{bmatrix} x_{11} & x_{21} \\ x_{12} & x_{22} \end{bmatrix} = \begin{bmatrix} 2x_{11} & x_{12} + x_{21} \\ x_{12} + x_{21} & 2x_{22} \end{bmatrix} \tag{21}$$

Multiplying by θ:

$$(X + X^\top)\theta = \begin{bmatrix} 2x_{11}\theta_1 + (x_{12} + x_{21})\theta_2 \\ (x_{12} + x_{21})\theta_1 + 2x_{22}\theta_2 \end{bmatrix} \tag{22}$$

This matches our derivative, so:

$$\frac{\partial\theta^\top X\theta}{\partial\theta} = (X + X^\top)\theta \tag{23}$$

Problem 4

Problem: Consider the function of a matrix,

$$f\left(\begin{bmatrix} x_1 & x_2 \\ 0 & x_3 \end{bmatrix}\right) = x_1^2 + (x_1 - x_2)^2 + (x_3 - 1)^2.$$

Part 1

Compute the derivative of f with respect to the input matrix $X = \begin{bmatrix} x_1 & x_2 \\ 0 & x_3 \end{bmatrix}$. What is the dimension of this derivative?

Solution:

We need to compute the derivative of f with respect to the matrix $X = \begin{bmatrix} x_1 & x_2 \\ 0 & x_3 \end{bmatrix}$.

Given:

$$f(X) = x_1^2 + (x_1 - x_2)^2 + (x_3 - 1)^2$$

Taking partial derivatives with respect to each element of X:

$$\frac{\partial f}{\partial x_1} = 2x_1 + 2(x_1 - x_2) = 4x_1 - 2x_2 \tag{24}$$

$$\frac{\partial f}{\partial x_2} = -2(x_1 - x_2) = -2x_1 + 2x_2 \tag{25}$$

$$\frac{\partial f}{\partial x_3} = 2(x_3 - 1) \tag{26}$$

The element at position $(2, 1)$ is fixed at 0, so $\frac{\partial f}{\partial x_{21}} = 0$.

Therefore, the derivative (gradient) matrix is:

$$\frac{\partial f}{\partial X} = \begin{bmatrix} 4x_1 - 2x_2 & -2x_1 + 2x_2 \\ 0 & 2(x_3 - 1) \end{bmatrix}$$

This derivative is a 2×2 matrix, matching the dimensions of X.

Part 2

Starting with $x_1 = 1, x_2 = 1, x_3 = 2$, write down the first 3 iterations of the Gradient Descent (GD) algorithm using learning rate $\rho = 1.5$. For each iteration, write down the updated value of X and the value of the function at the updated X. Check for convergence at each iteration.

Has GD converged in 3 iterations? If not, will it converge if you keep running it beyond 3 iterations? Explain what is happening.

Solution:

Starting with $x_1 = 1, x_2 = 1, x_3 = 2$ and learning rate $\rho = 1.5$.

The gradient descent update rule is:

$$X^{(k+1)} = X^{(k)} - \rho \nabla f(X^{(k)})$$

Iteration 0 (Initial):

$$X^{(0)} = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix}$$

$$f(X^{(0)}) = 1^2 + (1-1)^2 + (2-1)^2 = 1 + 0 + 1 = 2$$

$$\nabla f(X^{(0)}) = \begin{bmatrix} 4(1) - 2(1) & -2(1) + 2(1) \\ 0 & 2(2-1) \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

Iteration 1:

$$X^{(1)} = X^{(0)} - 1.5 \cdot \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix} - \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix} = \begin{bmatrix} -2 & 1 \\ 0 & -1 \end{bmatrix}$$

$$f(X^{(1)}) = (-2)^2 + (-2-1)^2 + (-1-1)^2 = 4 + 9 + 4 = 17$$

$$\nabla f(X^{(1)}) = \begin{bmatrix} 4(-2) - 2(1) & -2(-2) + 2(1) \\ 0 & 2(-1-1) \end{bmatrix} = \begin{bmatrix} -10 & 6 \\ 0 & -4 \end{bmatrix}$$

Iteration 2:

$$X^{(2)} = X^{(1)} - 1.5 \cdot \begin{bmatrix} -10 & 6 \\ 0 & -4 \end{bmatrix} = \begin{bmatrix} -2 & 1 \\ 0 & -1 \end{bmatrix} - \begin{bmatrix} -15 & 9 \\ 0 & -6 \end{bmatrix} = \begin{bmatrix} 13 & -8 \\ 0 & 5 \end{bmatrix}$$

$$f(X^{(2)}) = 13^2 + (13 - (-8))^2 + (5-1)^2 = 169 + 441 + 16 = 626$$

$$\nabla f(X^{(2)}) = \begin{bmatrix} 4(13) - 2(-8) & -2(13) + 2(-8) \\ 0 & 2(5-1) \end{bmatrix} = \begin{bmatrix} 68 & -42 \\ 0 & 8 \end{bmatrix}$$

Iteration 3:

$$X^{(3)} = X^{(2)} - 1.5 \cdot \begin{bmatrix} 68 & -42 \\ 0 & 8 \end{bmatrix} = \begin{bmatrix} 13 & -8 \\ 0 & 5 \end{bmatrix} - \begin{bmatrix} 102 & -63 \\ 0 & 12 \end{bmatrix} = \begin{bmatrix} -89 & 55 \\ 0 & -7 \end{bmatrix}$$

$$f(X^{(3)}) = (-89)^2 + (-89 - 55)^2 + (-7-1)^2 = 7921 + 20736 + 64 = 28721$$

Analysis:

GD has not converged in 3 iterations. The gradient is not zero, and the function values are: $2 \to 17 \to 626 \to 28721$.

The function value is increasing exponentially, indicating that the learning rate $\rho = 1.5$ is too large. The algorithm is diverging - each step overshoots the minimum and lands further away. If we continue running GD with this learning rate, it will not converge; instead, it will continue to diverge with the function value growing without bound.

Part 3

Starting with $x_1 = 1, x_2 = 1, x_3 = 2$, write down the first 3 iterations of the Gradient Descent (GD) algorithm using learning rate $\rho = 0.5$. For each iteration, write down the updated value of X and the value of the function at the updated X. Check for convergence at each iteration.

Has GD converged in 3 iterations? If not, will it converge if you keep running it beyond 3 iterations? Explain what is happening.

Solution:

Starting with $x_1 = 1, x_2 = 1, x_3 = 2$ and learning rate $\rho = 0.5$.

The gradient at the initial point is the same as Part 2:

$$X^{(0)} = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix}, \quad f(X^{(0)}) = 2, \quad \nabla f(X^{(0)}) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

Iteration 1:

$$X^{(1)} = X^{(0)} - 0.5\nabla f(X^{(0)}) = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix} - 0.5 \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$$

$$f(X^{(1)}) = 0^2 + (0-1)^2 + (1-1)^2 = 1$$

$$\nabla f(X^{(1)}) = \begin{bmatrix} -2 & 2 \\ 0 & 0 \end{bmatrix}$$

Iteration 2:

$$X^{(2)} = X^{(1)} - 0.5 \begin{bmatrix} -2 & 2 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$f(X^{(2)}) = 1^2 + (1-0)^2 + (1-1)^2 = 2$$

$$\nabla f(X^{(2)}) = \begin{bmatrix} 4 & -2 \\ 0 & 0 \end{bmatrix}$$

Iteration 3:

$$X^{(3)} = X^{(2)} - 0.5 \begin{bmatrix} 4 & -2 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ 0 & 1 \end{bmatrix}$$

$$f(X^{(3)}) = (-1)^2 + (-1-1)^2 + (1-1)^2 = 5$$

GD has not converged in 3 iterations. The gradient is not zero at iteration 3. The function values are: $2 \to 1 \to 2 \to 5$.

Notice that x_3 reached its optimal value of 1 after iteration 1 (since $\nabla_{x_3} f = 0$ from that point onward), so the x_3 component has converged. However, x_1 and x_2 are diverging: the function value decreased initially but then started growing again.

Although $\rho = 0.5$ diverges more slowly than $\rho = 1.5$, it is still too large and GD will *not* converge if we keep running it. The function f is quadratic in (x_1, x_2) with Hessian $H = \begin{bmatrix} 4 & -2 \\ -2 & 2 \end{bmatrix}$, whose largest eigenvalue is $\lambda_{\max} = 3 + \sqrt{5} \approx 5.24$. Convergence requires $\rho < \frac{2}{\lambda_{\max}} \approx 0.382$, so $\rho = 0.5$ is too large and the algorithm will keep diverging.

Problem 5

Problem: Consider the modified linear regression problem

$$\hat{\theta} = \arg\min_{\theta} \frac{1}{2} \sum_{i=1}^{N} \left(\theta^\top \phi(x_i) - y_i \right)^2 + \lambda a^\top \theta,$$

where a is a given (known) vector whose all entries have negative values.

Part 1

Derive the closed-form solution for the above problem. Provide all steps of the derivation.

Solution:

We want to derive the closed-form solution for:

$$\hat{\theta} = \arg\min_{\theta} \frac{1}{2} \sum_{i=1}^{N} \left(\theta^\top \phi(x_i) - y_i \right)^2 + \lambda a^\top \theta$$

where a is a given vector with all negative entries.

Let $\Phi \in \mathbb{R}^{N \times d}$ be the matrix whose rows are $\phi(x_i)^\top$ for $i = 1, \ldots, N$, and let $Y \in \mathbb{R}^N$ be the vector of responses y_i.

We can rewrite the objective function in matrix form:

$$J(\theta) = \frac{1}{2} \|\Phi\theta - Y\|_2^2 + \lambda a^\top \theta \tag{27}$$

Expanding the squared norm:

$$J(\theta) = \frac{1}{2} (\Phi\theta - Y)^\top (\Phi\theta - Y) + \lambda a^\top \theta \tag{28}$$

$$= \frac{1}{2} (\theta^\top \Phi^\top \Phi\theta - 2\theta^\top \Phi^\top Y + Y^\top Y) + \lambda a^\top \theta \tag{29}$$

$$= \frac{1}{2} \theta^\top \Phi^\top \Phi\theta - \theta^\top \Phi^\top Y + \frac{1}{2} Y^\top Y + \lambda a^\top \theta \tag{30}$$

To find the minimum, take the gradient with respect to θ and set it to zero:

$$\nabla_\theta J(\theta) = \Phi^\top \Phi\theta - \Phi^\top Y + \lambda a = 0 \tag{31}$$

Solving for θ:

$$\Phi^\top \Phi\theta = \Phi^\top Y - \lambda a \tag{32}$$

Therefore, the closed-form solution is:

$$\boxed{\hat{\theta} = (\Phi^\top \Phi)^{-1} (\Phi^\top Y - \lambda a)} \tag{33}$$

Part 2

What is the solution $\hat{\theta}$ when $\lambda \to \infty$?

Solution:

We want to determine $\hat{\theta}$ as $\lambda \to \infty$.

From Part 1, we have:

$$\hat{\theta} = (\Phi^\top \Phi)^{-1}(\Phi^\top Y - \lambda a)$$

We can rewrite this as:

$$\hat{\theta} = (\Phi^\top \Phi)^{-1}\Phi^\top Y - \lambda(\Phi^\top \Phi)^{-1}a$$

As $\lambda \to \infty$, the first term is constant so the second term dominates. Since all entries of a are negative, $-a$ is a vector with all positive entries. Moreover, $(\Phi^\top \Phi)^{-1}$ is positive definite, so $(\Phi^\top \Phi)^{-1}(-a)$ is a nonzero vector. Therefore $\hat{\theta}$ grows without bound in the direction of $(\Phi^\top \Phi)^{-1}(-a)$:

$$\boxed{\|\hat{\theta}\| \to \infty \quad \text{as } \lambda \to \infty}$$

This makes sense: since a has all negative entries, the term $a^\top \theta$ decreases (becomes more negative) as θ grows, so $\lambda a^\top \theta$ is minimized by making θ arbitrarily large. Increasing λ amplifies this effect, pushing $\hat{\theta}$ away from the origin rather than toward it. This is the opposite of standard regularization.

Problem 6

Problem: In this problem we study the effect of the cost/loss function on the solution of the regression problem in the presence of outliers. Consider the dataset shown in the figure, where there are 10 input samples (inliers) whose output response is 1 and there are 4 input samples (outliers) with the output response of 0. For simplicity, we consider the regression model $h(x) = \theta_0$, which corresponds to a horizontal line.

Part 1

Consider the cost/loss function $\sum_{i=1}^{14}(h(x_i) - y_i)^2$. Derive the solution of minimizing the loss with respect to θ_0 (show all steps of the derivation). Draw the associated line. Does this solution best fit the inliers?

Solution:

We want to minimize the L2 loss function:

$$L(\theta_0) = \sum_{i=1}^{14}(h(x_i) - y_i)^2 = \sum_{i=1}^{14}(\theta_0 - y_i)^2$$

where $h(x) = \theta_0$ (a horizontal line).

From the dataset, we have 10 inliers with $y_i = 1$ and 4 outliers with $y_i = 0$. Therefore:

$$L(\theta_0) = \sum_{i=1}^{10}(\theta_0 - 1)^2 + \sum_{i=11}^{14}(\theta_0 - 0)^2$$

$$= 10(\theta_0 - 1)^2 + 4\theta_0^2$$

To find the minimum, take the derivative with respect to θ_0 and set it to zero:

$$\frac{dL}{d\theta_0} = 10 \cdot 2(\theta_0 - 1) + 4 \cdot 2\theta_0 = 0$$

$$20\theta_0 - 20 + 8\theta_0 = 0$$

$$28\theta_0 = 20$$

$$\theta_0 = \frac{20}{28} = \frac{5}{7} \approx 0.714$$

The associated horizontal line is $h(x) = \frac{5}{7}$.

Does this solution best fit the inliers?

No, this solution does not best fit the inliers. The optimal fit for the 10 inliers alone would be $\theta_0 = 1$ (their actual response value). However, the L2 loss is heavily influenced by the 4 outliers at $y = 0$, pulling the solution down to $\theta_0 = \frac{5}{7}$. The squared loss heavily penalizes large errors, so the optimization compromises between fitting the inliers and not being too far from the outliers.

Part 2

Consider the cost/loss function $\sum_{i=1}^{14} |h(x_i) - y_i|$. Given the dataset shown in the figure, we know that the optimal θ_0 must be between 0 and 1, i.e., $\theta_0 \in [0, 1]$. Using this knowledge, derive the solution of minimizing the loss with respect to θ_0 (show all steps of the derivation). Draw the associated line. Does this solution best fit the inliers?

Solution:

We want to minimize the L1 loss function:

$$L(\theta_0) = \sum_{i=1}^{14} |h(x_i) - y_i| = \sum_{i=1}^{14} |\theta_0 - y_i|$$

Given that $\theta_0 \in [0, 1]$, we have 10 samples with $y_i = 1$ and 4 samples with $y_i = 0$.

For $\theta_0 \in [0, 1]$: - When $y_i = 1$: $|\theta_0 - 1| = 1 - \theta_0$ (since $\theta_0 \leq 1$) - When $y_i = 0$: $|\theta_0 - 0| = \theta_0$ (since $\theta_0 \geq 0$)

Therefore:

$$L(\theta_0) = \sum_{i=1}^{10}(1 - \theta_0) + \sum_{i=11}^{14} \theta_0$$

$$= 10(1 - \theta_0) + 4\theta_0$$

$$= 10 - 10\theta_0 + 4\theta_0$$

$$= 10 - 6\theta_0$$

To minimize $L(\theta_0) = 10 - 6\theta_0$, we note that the loss is a decreasing linear function of θ_0. Since the coefficient of θ_0 is negative ($-6 < 0$), the loss decreases as θ_0 increases.

Given the constraint $\theta_0 \in [0, 1]$, the minimum occurs at the largest possible value:

$$\theta_0^* = 1$$

The associated horizontal line is $h(x) = 1$.

Does this solution best fit the inliers?

Yes, this solution perfectly fits the inliers. The L1 loss is less sensitive to outliers compared to L2 loss. By choosing $\theta_0 = 1$, we achieve zero error on all 10 inliers (the majority of the data), while accepting an error of 1 on each of the 4 outliers. The L1 loss effectively finds the median response, which in this case is dominated by the inliers. This demonstrates the robustness of L1 loss to outliers.

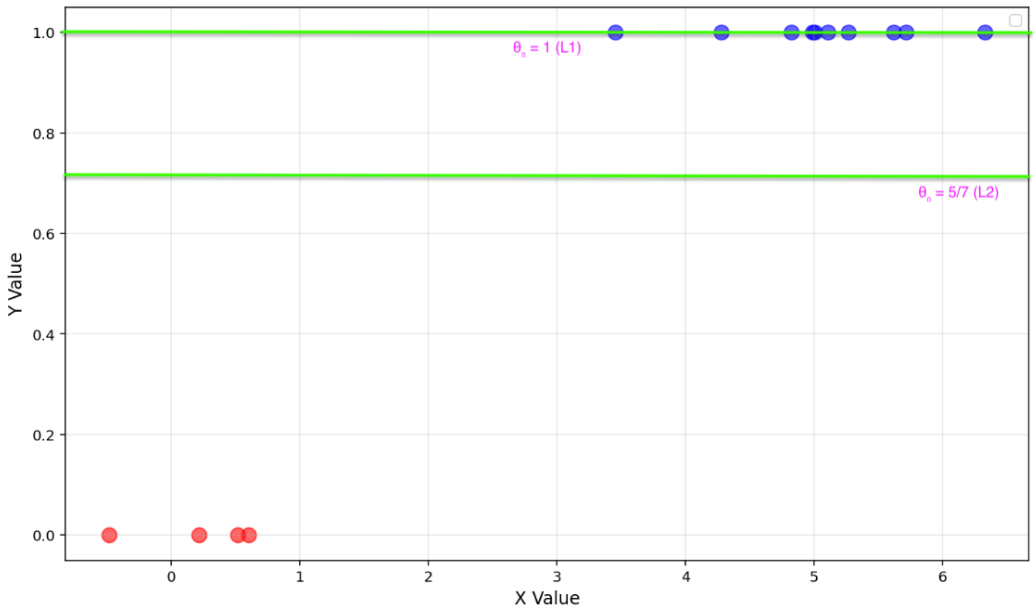

Figure 1: Comparison of L2 and L1 loss solutions. The L2 solution ($\theta_0 = \frac{5}{7}$) is pulled toward the outliers, while the L1 solution ($\theta_0 = 1$) fits the inliers exactly.

Problem 7

Part 1: Data Visualization

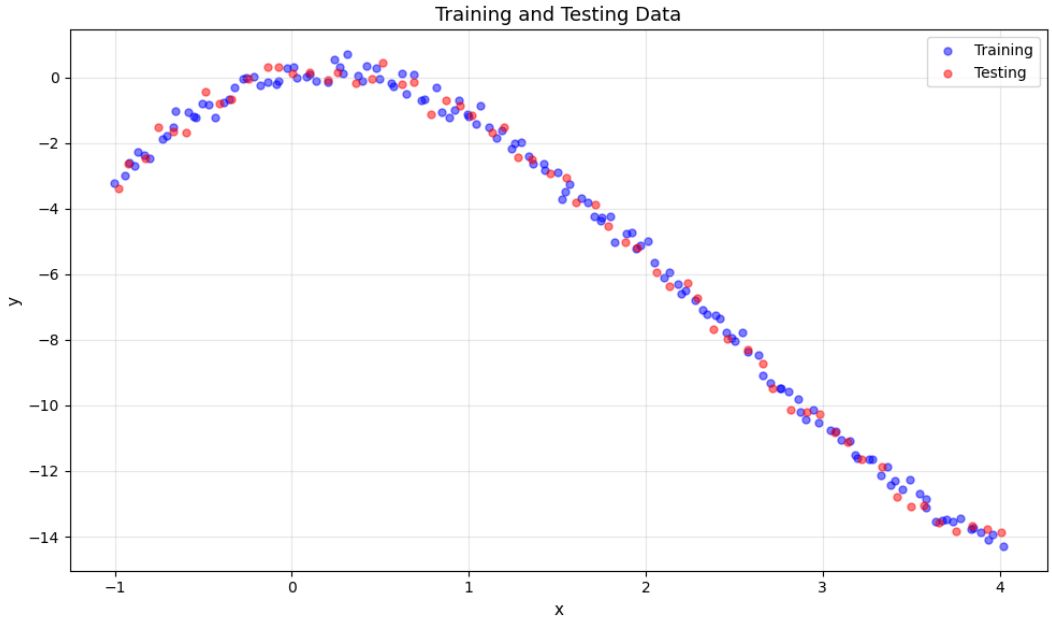

Figure 2: Degree 1 (linear) regression: closed-form and gradient descent solutions.

Part 2: Linear Model ($y = \theta_0 + \theta_1 x$)

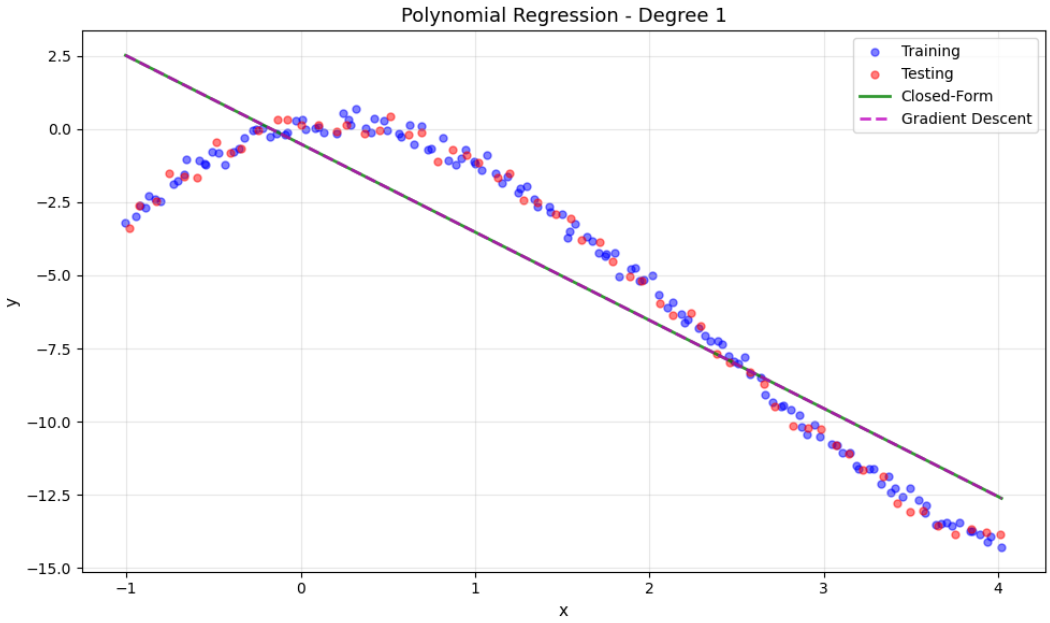

Figure 3: Degree 1 (linear) regression: closed-form and gradient descent solutions.

(a) Closed-Form: $\theta_0 = -0.5100$, $\theta_1 = -3.0136$. Train MSE: 3.7514, Test MSE: 3.9580.

(b) Gradient Descent: $\theta_0 = -0.5101$, $\theta_1 = -3.0136$. Train MSE: 3.7514, Test MSE: 3.9580.

(c) Comparison: The parameters from both methods are nearly identical (max difference $< 8 \times 10^{-5}$), and the MSE values match to six decimal places. The two fitted lines overlap completely in the plot, confirming that gradient descent converged to the same solution as the closed-form.

Part 3: Quadratic Model ($y = \theta_0 + \theta_1 x + \theta_2 x^2$)

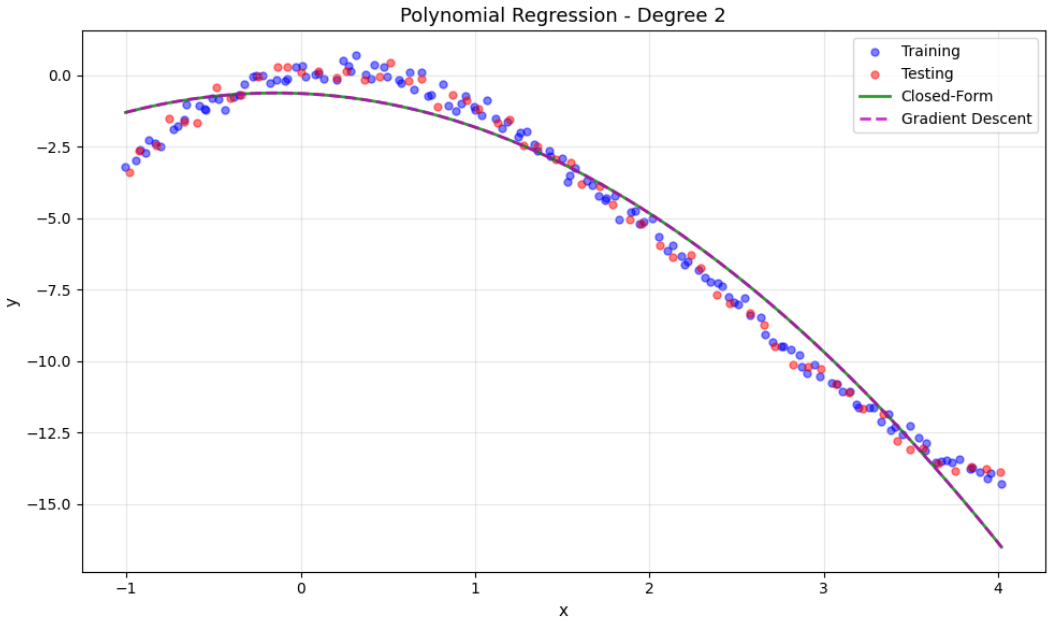

Figure 4: Degree 2 (quadratic) regression: closed-form and gradient descent solutions.

(a) Closed-Form: $\theta_0 = -0.6322$, $\theta_1 = -0.2612$, $\theta_2 = -0.9178$. Train MSE: 0.7359, Test MSE: 0.8703.

(b) Gradient Descent: $\theta_0 = -0.6317$, $\theta_1 = -0.2621$, $\theta_2 = -0.9176$. Train MSE: 0.7359, Test MSE: 0.8703.

(c) Comparison: Parameters match closely (max difference $< 9 \times 10^{-4}$). MSE values are effectively identical. The quadratic model substantially improves over the linear fit, reducing train MSE from 3.75 to 0.74.

Part 4: Cubic Model ($y = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3$)

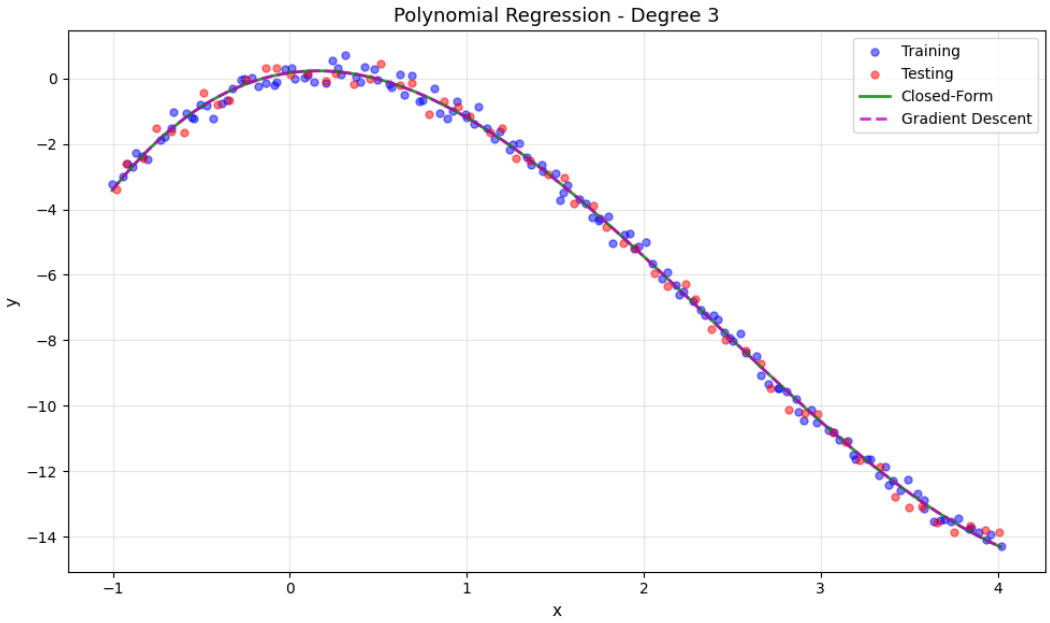

Figure 5: Degree 3 (cubic) regression: closed-form and gradient descent solutions.

(a) Closed-Form: $\theta_0 = 0.1712$, $\theta_1 = 0.7525$, $\theta_2 = -2.4655$, $\theta_3 = 0.3438$. Train MSE: 0.0474, Test MSE: 0.0660.

(b) Gradient Descent: $\theta_0 = 0.1689$, $\theta_1 = 0.7474$, $\theta_2 = -2.4605$, $\theta_3 = 0.3428$. Train MSE: 0.0474, Test MSE: 0.0661.

(c) Comparison: Parameters are close (max difference $\approx 5 \times 10^{-3}$), slightly larger than the lower-degree models due to the smaller learning rate needed for stability with higher-degree polynomial features. The cubic model provides the best fit overall, with train MSE dropping from 0.74 (quadratic) to 0.047.

Problem 8

Part 2(a): Polynomial Degree Analysis

Degree	Train MSE	Validation MSE
1	21.9953	29.2090
2	16.1150	21.8699
3	10.8825	15.1081
4	6.3568	11.8550
5	4.0695	25.8632

Both training and validation error decrease from degree 1 to 4. At degree 5, the training error continues to drop but the validation error jumps sharply, indicating overfitting.

Part 2(b): Ridge Regularization

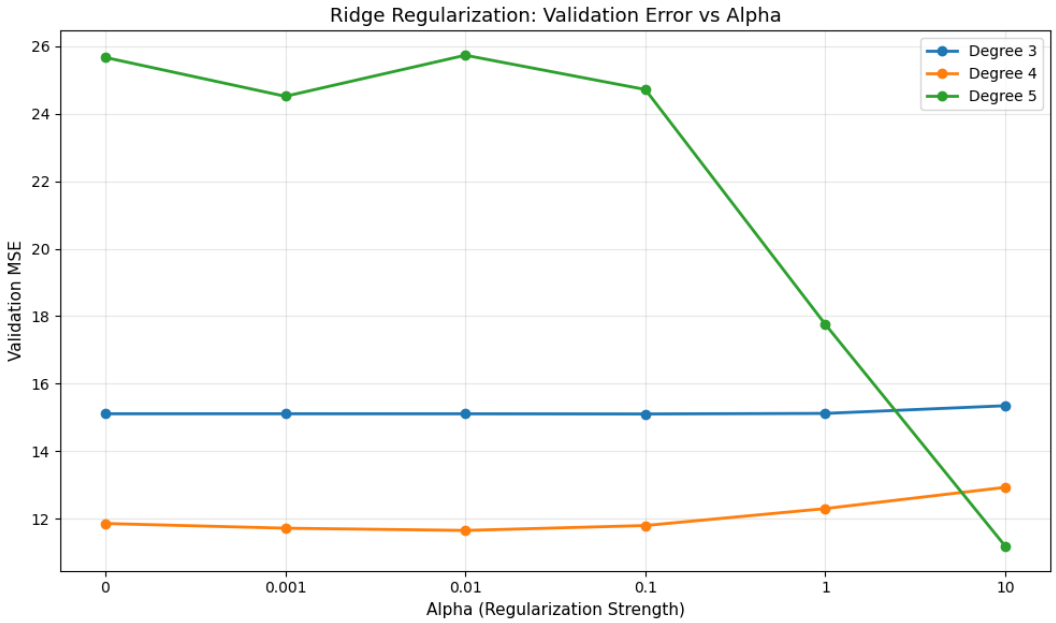

Figure 6: Validation MSE vs. regularization strength (alpha) for degrees 3, 4, and 5.

Degrees 3 and 4 are relatively stable across all alpha values, indicating they are not severely overfitting. Degree 5 is highly sensitive to regularization: without it, the validation MSE is around 25, but with alpha = 10, it drops to 11.18, which is the lowest validation MSE across all models.

Part 2(c): Best Model and Test Performance

The best model is **Ridge regression with degree 5 and alpha** = 10, selected based on the lowest validation MSE (11.18).

Test MSE: 6.9726