Problem Set 2

Problem 1 (Gradient of the MSE)

Part A

Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ be a matrix, let $\mathbf{w} \in \mathbb{R}^d$ be a vector, and let $\mathbf{y} \in \mathbb{R}^n$ be a vector. Determine the dimensions of the following quantities:

- $\mathbf{X}\mathbf{w}$
- $\mathbf{X}\mathbf{w} - \mathbf{y}$
- $\mathbf{X}^\top \mathbf{X}$
- $\mathbf{X}^\top(\mathbf{X}\mathbf{w} - \mathbf{y})$

Part B

The mean-squared error for multivariate regression with data matrix $\mathbf{X}$, weight vector $\mathbf{w}$, and target vector $\mathbf{y}$ can be expressed with the formula

$$\text{MSE}(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^{n} \left(\mathbf{w}^\top \mathbf{x}_i - y_i\right)^2,$$

where $\mathbf{w}^\top \mathbf{x}_i$ is the inner product (dot product) with formula $\sum_{j=1}^{d} w_j x_{ij}$.

Compute $\frac{\partial \text{MSE}}{\partial w_j}$, the partial derivative of MSE with respect to the jth weight parameter w_j. It's fine for your solution to be expressed in terms of the *entries* of $\mathbf{X}$, $\mathbf{w}$, and $\mathbf{y}$ (i.e. in terms of sums and products of scalar quantities, not matrix and vector operations).

Part C

Matrix and vector operations have entrywise formulas for individual entries. We just saw one in the previous part: the value (only entry) of $\mathbf{w}^\top \mathbf{x}_i$ is $\sum_{j=1}^{d} w_j x_{ij}$. Similarly, for example, the jth entry of $\mathbf{X}\mathbf{w}$ is $\sum_{k=1}^{d} x_{jk} w_k$, using standard matrix-vector multiplication.

Use these and other formulas to give an expression for the jth entry of the vector $\mathbf{X}^\top(\mathbf{X}\mathbf{w} - \mathbf{y})$, which we computed the dimensions of in Part A.

Hint: *I found it helpful to first define the vector $\hat{\mathbf{y}} = \mathbf{X}\mathbf{w}$, which is the vector of predictions for each data point. Then, the ith entry of $\hat{\mathbf{y}}$ is $\hat{y}_i = \sum_{k=1}^{d} w_k x_{ik}$. I then wrote out the entries of $\hat{\mathbf{y}} - \mathbf{y}$ and then the entries of $\mathbf{X}^\top(\hat{\mathbf{y}} - \mathbf{y})$, and then plugged in the formulas for the entries of $\hat{\mathbf{y}}$ back in.*

Part D

Give a formula for the gradient $\nabla \text{MSE}(\mathbf{w})$ in terms of $\mathbf{X}$, $\mathbf{y}$, and $\mathbf{w}$.

Problem 2 (One Source of Overfitting in Linear Regression)

Let $R(\mathbf{w})$ be the mean-squared error for multivariate linear regression with data matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$, weight vector $\mathbf{w} \in \mathbb{R}^d$, and target vector $\mathbf{y} \in \mathbb{R}^n$:

$$R(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^{n} \left(\mathbf{w}^\top \mathbf{x}_i - y_i \right)^2 = \frac{1}{n} (\| \mathbf{X}\mathbf{w} - \mathbf{y} \|)^2.$$

The equation $\nabla R(\mathbf{w}) = \mathbf{0}$ is a necessary condition for $\mathbf{w}$ to be a minimizer of R, and is equivalent to the system of linear equations often called the *normal equations*:

$$\mathbf{X}^\top \mathbf{X}\mathbf{w} = \mathbf{X}^\top \mathbf{y}.$$

Part A

An extreme case in which overfitting often occurs in which we have *fewer* data points than features, i.e. $n < d$. Do the normal equations have a unique solution?

Part B

Still assuming that $n < d$, suppose that we have found a solution $\hat{\mathbf{w}}$ to the normal equations. Give an algorithm (recipe, approach) for finding *another* solution $\hat{\mathbf{w}}'$ to the normal equations.

Part C

Still assuming that $n < d$, give an algorithm (recipe, approach) for finding a solution $\hat{\mathbf{w}}$ to the normal equations such that at least one entry of $\hat{\mathbf{w}}$ is arbitrarily large in magnitude.

Problem 3 (Closed-Form Solution for Least-Squares Linear Regression)

Recall that to perform maximum-likelihood estimation in the multivariate linear-Gaussian model, it's sufficient to minimize the mean-squared error (MSE):

$$R(\mathbf{x}, \mathbf{y}; \mathbf{w}) = \frac{1}{n}(\| \mathbf{X}\mathbf{w} - \mathbf{y} \|)^2,$$

Part A

Solve the equation $\nabla R(\mathbf{x}, \mathbf{y}; \hat{\mathbf{w}}) = 0$ for $\hat{\mathbf{w}}$ to derive a closed-form expression for the maximum-likelihood estimate $\hat{\mathbf{w}}$ in terms of $\mathbf{X}$ and $\mathbf{y}$. You may assume in this part that the matrix $\mathbf{X}^T\mathbf{X}$ is invertible.

Part B

Describe a necessary and sufficient condition for the matrix $\mathbf{X}^T\mathbf{X}$ to be invertible. (Your condition may be stated in terms of the rank of $\mathbf{X}$ or in terms of linear independence of the columns of $\mathbf{X}$.) Relate your answer to the idea of a "redundant" feature in the dataset.

Part C

(*There's no work for you to do here, just a note.*)

Question

Why don't we use the closed-form solution from Part A when using linear regression in the lecture notes?

Answer

There are two main reasons.

First, computing the matrix inverse $\left(\mathbf{X}^T\mathbf{X}\right)^{-1}$ is computationally expensive in problems with a large number of features. The time complexity of matrix inversion is typically $O(d^3)$ for a $d \times d$ matrix, which can be prohibitive for large d.

Second, we're using linear regression to set the stage for more complex models, especially including deep neural networks. There's no closed-form approach for neural networks – we need to use iterative optimization procedures like gradient methods, and we're using linear regression as a simple example to illustrate these methods.

Problem 4 (Predictions with Noise)

Let Y be a (discrete) random variable with known mean $\mathbb{E}[Y] = \mu$ and variance $\text{var}(Y) = \sigma^2$. We want to predict Y with a single number $\hat{y}$.

We're going to measure the success of our prediction $\hat{y}$ with the mean-squared error, which in this case is just $(\hat{y} - Y)^2$. Since Y is a random variable, the mean-squared error of our prediction $\hat{y}$ is also a random variable. To get a single number that measures the quality of our prediction, we take the expectation of the mean-squared error. This is called the *risk* of our prediction $\hat{y}$, and is given by

$$R(\hat{y}) = \mathbb{E}[(\hat{y} - Y)^2] = \sum_y (\hat{y} - y)^2 p(y).$$

Here, the sum ranges over all possible values of y.

Part A

Show that the risk of our prediction $\hat{y}$ can be written as

$$R(\hat{y}) = (\hat{y} - \mu)^2 + \sigma^2.$$

We can think of the first term as a measure of our prediction's *bias* (deviation from the mean μ of Y) and the second term as a measure of the *noise level* in our target (the inherent variability of Y).

Part B

Prove that the risk $R(\hat{y})$ is minimized when $\hat{y} = \mu$. In other words, in the absence of other information about Y, the best prediction we can possibly make (as measured by the expected MSE R) is to predict the mean μ of Y. What is the lowest possible value of the risk that we can achieve?

Problem 5 (Justifying Gradient Descent)

Part A

Let $f : \mathbb{R} \to \mathbb{R}$ be a differentiable function for which we'd like to solve

$$\hat{x} = \arg\min_{x \in \mathbb{R}} f(x).$$

Suppose we have a current guess $x_0 \neq \hat{x}$, which has value $f(x_0) > f(\hat{x})$ and derivative $f'(x_0) \neq 0$. We'd like to form a new guess x_1 with the property that $f(x_1) < f(x_0)$, i.e. that x_1 is a better guess than x_0. In the gradient descent update for this function (with fixed learning rate α), we would set $x_1 = x_0 - \alpha f'(x_0)$ for some $\alpha > 0$.

Consider the function $g(x) = f(x_0) + f'(x_0)(x - x_0)$, which is the tangent line approximation to f at x_0. Prove that, for any $\alpha > 0$, we have $g(x_1) < g(x_0)$.

Note: In a course like real analysis, you would prove that for α sufficiently small, $g(x)$ is *close* to $f(x)$ near x_0, and so the fact that $g(x_1) < g(x_0)$ implies that $f(x_1) < f(x_0)$ if α is small enough.

Part B

Let $f : \mathbb{R}^d \to \mathbb{R}$ be a differentiable function for which we'd like to solve

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x} \in \mathbb{R}^d} f(\mathbf{x}).$$

Suppose we again have a current guess $\mathbf{x}_0 \neq \hat{\mathbf{x}}$, which has value $f(\mathbf{x}_0) > f(\hat{\mathbf{x}})$ and gradient $\nabla f(\mathbf{x}_0) \neq \mathbf{0}$. We'd like to form a new guess $\mathbf{x}_1$ with the property that $f(\mathbf{x}_1) < f(\mathbf{x}_0)$, i.e. that $\mathbf{x}_1$ is a better guess than $\mathbf{x}_0$. In the gradient descent update for this function (with fixed learning rate α), we would set $\mathbf{x}_1 = \mathbf{x}_0 - \alpha \nabla f(\mathbf{x}_0)$ for some $\alpha > 0$.

Consider the function $g(\mathbf{x}) = f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0)^\top (\mathbf{x} - \mathbf{x}_0)$, which is the tangent hyperplane approximation to f at $\mathbf{x}_0$. Prove that, for any $\alpha > 0$, we have $g(\mathbf{x}_1) < g(\mathbf{x}_0)$.

Note: This is the multivariate analogue of the previous part, and the proof can be very similar.