CSCI 567, Fall 2025
Haipeng Luo

Written Assignment #1

Due: Sep 17, 2025, 11:59 pm, PT

Instructions

Total points: 50

Submission: Solutions must be typewritten or neatly handwritten and submitted through gradescope. You can submit multiple times, but only the last submission counts. It is your responsibility to make sure that you submit the right things, and we will *not* consider any regrading requests regarding mistakes in making submissions.

Recall that you have a total of three "late days" for the entire semester, and you can use at most one late day for each written assignment.

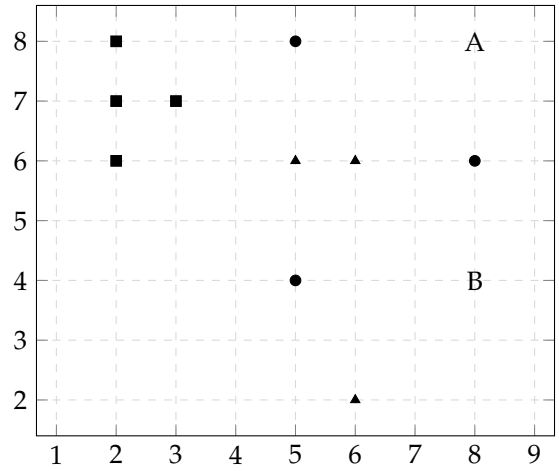
Notes on notation:

- Unless stated otherwise, scalars are denoted by small letter in normal font, vectors are denoted by small letters in bold font, and matrices are denoted by capital letters in bold font.
- $\|\cdot\|$ means L2-norm unless specified otherwise, i.e., $\|\cdot\| = \|\cdot\|_2$.

Academic integrity: Our goal is to maintain an optimal learning environment. You can discuss the written assignments at a high level with others, but you should not look at any other's solutions. Trying to find solutions online or from any other sources (including ChatGPT and other similar tools) is prohibited, will result in zero grade and will be reported. To prevent any future plagiarism, uploading any materials from this course to the Internet is also prohibited, and any violations will be reported. Please be considerate and help us help everyone get the best out of this course.

Problem 1 Nearest Neighbor Classification (10 points)

For the data given below, squares, triangles, and circles are three different classes in the training set, and A and B are two test points with an unknown class. We denote the total number of training points as N (which equals 10) and consider K-nearest-neighbor (KNN) classifier with **L1** distance.



1. What is the test point A classified as for $K = 1$? Explain briefly. (2 points)

   Circle. (1 point)

   The closest point to A is clearly the circle located at $(8, 6)$ (with L1 distance 2). (1 point)

2. What is the smallest odd value of K for KNN to predict triangle for the test point B? Explain briefly. (4 points)

   The smallest value is $K = 5$. (1 point)

   The closest point is the circle at $(8, 6)$, so $K = 1$ does not work; The next two closest points are the circle at $(5, 4)$ and the triangle either at $(6, 2)$ or $(6, 6)$, so $K = 3$ also does not work. Finally, when $K = 5$, two more triangles will be included (one at either $(6, 2)$ or $(6, 6)$ and another at $(5, 6)$), so the final prediction will be triangle. (3 points)

   Rubrics: If one uses L2 distance, the answer will instead be $K = 3$. Give 1 point (in total) to this case.

3. Suppose one performs leave-one-out validation (that is, N-fold cross validation) to choose the best hyper-parameter K. List all the points that are misclassified during the N runs when testing the hyper-parameter value $K = 1$, and report the averaged error rate for this hyper-parameter. (4 points)

   The points that are misclassified during the N runs are the triangle at $(6, 2)$ and all the three circles, and thus the error rate is $4/10 = 0.4$.

   Rubrics:

   - List the four points correctly. (2 points)
   - No listing of other points. (1 point)
   - Report the error rate correctly as the number of listed points (which might not be the correct answer 4) divided by 10. (1 point)

Problem 2 Linear Regression (24 points)

2.1 (10 points) In the class, we discussed L2 regularized least square solution defined as

$$\boldsymbol{w}_* = \arg\min_{\boldsymbol{w}\in\mathbb{R}^D} \|\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{w}\|_2^2 \tag{1}$$

where $\boldsymbol{X} \in \mathbb{R}^{N\times D}$ is the data matrix with each row corresponding to the feature of an example, $\boldsymbol{y} \in \mathbb{R}^N$ is a vector of all the outcomes, $\|\cdot\|_2$ stands for the L2 norm, and λ is the regularization coefficient. In this problem, we consider a different regularization method:

$$\boldsymbol{w}'_* = \arg\min_{\boldsymbol{w}\in\mathbb{R}^D} \|\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}\|_2^2 + \boldsymbol{w}^T\boldsymbol{M}\boldsymbol{w} \tag{2}$$

where $\boldsymbol{M} \in \mathbb{R}^{D\times D}$ is a sysmetric positive definite matrix.

1. Show that the new method is a generalization of the standard L2 regularization by picking a matrix $\boldsymbol{M}$ such that $\boldsymbol{w}'_*$ in Eq. (2) equals $\boldsymbol{w}_*$ in Eq. (1). (2 points)

   $\boldsymbol{M} = \lambda\boldsymbol{I}$ where $\boldsymbol{I}$ is the D by D identify matrix clearly makes $\boldsymbol{w}^T\boldsymbol{M}\boldsymbol{w}$ equal to $\lambda\|\boldsymbol{w}\|_2^2$. (2 points)

2. Find the closed form of $\boldsymbol{w}'_*$ by writing down the gradient of $F(\boldsymbol{w}) = \|\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}\|_2^2 + \boldsymbol{w}^T\boldsymbol{M}\boldsymbol{w}$ and setting it to $\mathbf{0}$. (4 points)

   The gradient is $2\boldsymbol{X}^T(\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}) + 2\boldsymbol{M}\boldsymbol{w}$. (2 points)

   Setting it to $\mathbf{0}$ and using the fact that $\boldsymbol{M}$ is invertible gives (2 points)

   $$\boldsymbol{w}'_* = \left(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X} + \boldsymbol{M}\right)^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{y}.$$

3. Recall the Newton method: $\boldsymbol{w}^{(t+1)} \leftarrow \boldsymbol{w}^{(t)} - \boldsymbol{H}_t^{-1}\nabla F(\boldsymbol{w}^{(t)})$ where $H_t = \nabla^2 F(\boldsymbol{w}^{(t)})$. Show that no matter what the initialization $\boldsymbol{w}^{(0)}$ is, Newton method always takes one step only to find the minimizer $\boldsymbol{w}'_*$ of $F(\boldsymbol{w}) = \|\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}\|_2^2 + \boldsymbol{w}^T\boldsymbol{M}\boldsymbol{w}$. (4 points)

   The Hessian of F at any point is always $2(\boldsymbol{X}^T\boldsymbol{X} + \boldsymbol{M})$. (2 points)

   Therefore, applying Newton method to any initial point $\boldsymbol{w}^{(0)}$ gives (2 points)

$$\begin{aligned}
\boldsymbol{w}^{(1)} &= \boldsymbol{w}^{(0)} - (\boldsymbol{X}^T\boldsymbol{X} + \boldsymbol{M})^{-1}(\boldsymbol{X}^T(\boldsymbol{X}\boldsymbol{w}^{(0)} - \boldsymbol{y}) + \boldsymbol{M}\boldsymbol{w}^{(0)}) \\
&= \boldsymbol{w}^{(0)} - (\boldsymbol{X}^T\boldsymbol{X} + \boldsymbol{M})^{-1}\left((\boldsymbol{X}^T\boldsymbol{X} + \boldsymbol{M})\boldsymbol{w}^{(0)} - \boldsymbol{X}^T\boldsymbol{y}\right) \\
&= \left(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X} + \boldsymbol{M}\right)^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{y} = \boldsymbol{w}'_*.
\end{aligned}$$

2.2 (14 points) Assume we have a training set $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N) \in \mathbb{R}^D \times \mathbb{R}$, where each outcome y_n is generated by a probabilistic model $\boldsymbol{w}_*^{\mathrm{T}} \boldsymbol{x}_n + \epsilon_n$ with ϵ_n being an independent Gaussian noise with zero-mean and variance σ^2 for some $\sigma > 0$. In other words, the probability density of any outcome $y \in \mathbb{R}$ given $\boldsymbol{x}_n$ is

$$\Pr(y \mid \boldsymbol{x}_n; \boldsymbol{w}_*, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(y - \boldsymbol{w}_*^{\mathrm{T}} \boldsymbol{x}_n)^2}{2\sigma^2}\right).$$

1. Assume σ is fixed and given, find the maximum likelihood estimation for $\boldsymbol{w}_*$. In other words, first write down the joint density of the outcomes $y_1, \ldots, y_N$ given $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ as a function of the value of $\boldsymbol{w}_*$; then find the value of $\boldsymbol{w}_*$ that maximizes this density. You can assume $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}$ is invertible where $\boldsymbol{X}$ is the data matrix as used in Problem 2.1. (6 points)

   The joint density for a linear model $\boldsymbol{w}$ is

   $$\mathcal{P}(\boldsymbol{w}) = \prod_{n=1}^{N} \Pr(y_n \mid \boldsymbol{x}_n; \boldsymbol{w}, \sigma) = \prod_{n=1}^{N} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(y_n - \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_n)^2}{2\sigma^2}\right).$$

   Taking the negative log, this becomes

   $$F(\boldsymbol{w}) = N \ln \sqrt{2\pi} + N \ln \sigma + \frac{1}{2\sigma^2} \sum_{n=1}^{N} (y_n - \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_n)^2 = N \ln \sqrt{2\pi} + N \ln \sigma + \frac{1}{2\sigma^2} \|\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}\|_2^2.$$

   Maximizing $\mathcal{P}$ is the same as minimizing F, which is clearly the same as just minimizing $\sum_{n=1}^{N}(y_n - \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_n)^2$, the same objective as for least square regression. Therefore the MLE for $\boldsymbol{w}_*$ is exactly the same as the least square solution:

   $$\boldsymbol{w}_* = (\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{y}.$$

   Rubrics:

   - Write down the correct likelihood function. (2 points)
   - Any derivation revealing that this is the same as least square regression. (2 points)
   - Arrive at the correct final answer. (2 points)

2. Now consider σ as a parameter of the probabilistic model too, that is, the model is specified by both $\boldsymbol{w}_*$ and σ. Find the maximum likelihood estimation for $\boldsymbol{w}_*$ and σ. (8 points)

From previous calculation, the MLE for $\boldsymbol{w}_*$ and σ is the minimizer of the function

$$F(\boldsymbol{w}, \sigma) = N \ln \sqrt{2\pi} + N \ln \sigma + \frac{1}{2\sigma^2} \|\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}\|_2^2.$$

We first fix σ and minimize over $\boldsymbol{w}$, which leads to the same MLE for $\boldsymbol{w}*$:

$$\boldsymbol{w}_* = (\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{y}.$$

Next we minimize $F(\boldsymbol{w}_*, \sigma)$ as function of σ. This can be done by setting the derivative w.r.t. σ to be 0:

$$\frac{\partial F(\boldsymbol{w}_*, \sigma)}{\partial \sigma} = \frac{N}{\sigma} - \frac{1}{\sigma^3} \|\boldsymbol{X}\boldsymbol{w}_* - \boldsymbol{y}\|_2^2 = 0.$$

Solving for σ gives:

$$\sigma = \frac{1}{\sqrt{N}} \|\boldsymbol{X}\boldsymbol{w}_* - \boldsymbol{y}\|_2 = \frac{1}{\sqrt{N}} \|\boldsymbol{X}(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{y} - \boldsymbol{y}\|_2.$$

This stationary point is indeed the minimizer since one can further verify that $\frac{\partial^2 F(\boldsymbol{w}_*, \sigma)}{\partial \sigma^2}$ is non-negative at this point (i.e., the function is convex around this point).

Rubrics:

- Write down the correct likelihood as a function of both $\boldsymbol{w}$ and σ. (2 points)
- Arrive at the correct solution for $\boldsymbol{w}_*$. (2 points)
- Correct derivative with respect to σ. (2 points)
- Arrive at the correct solution for σ. (2 points)
- For simplicity, it is okay to skip the last reasoning about convexity.

Problem 3 Linear Classifiers (16 points)

In Lecture 3 we have seen the hinge loss $\ell(z) = \max\{0, 1-z\}$, which is non-differentiable at $z = 1$. To avoid this issue, we can consider the square of hinge loss $\ell(z)^2$, which is differentiable everywhere. More specifically, given a binary dataset $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N) \in \mathbb{R}^D \times \{-1, 1\}$, we define the following new loss function for a linear model $\boldsymbol{w} \in \mathbb{R}^D$:

$$F(\boldsymbol{w}) = \frac{1}{N} \sum_{n=1}^{N} F_n(\boldsymbol{w}), \quad \text{where } F_n(\boldsymbol{w}) = \left(\max\left\{0, 1 - y_n \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_n\right\}\right)^2. \tag{3}$$

1. For a fixed n, write down the gradient $\nabla F_n(\boldsymbol{w})$ (show your derivation), then fill in the missing details in the repeat-loop of the algorithm below which applies SGD to minimize F. (6 points)

Algorithm 1: SGD for minimizing Eq. (3)

Input: A training set $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N) \in \mathbb{R}^D \times \{-1, 1\}$, learning rate $\eta > 0$
Initialization: $\boldsymbol{w} = \boldsymbol{0}$
Repeat:
  randomly pick an example $(\boldsymbol{x}_n, y_n)$
  update $\boldsymbol{w} \leftarrow \boldsymbol{w} + 2\eta y_n \max\{0, 1 - y_n \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_n\} \boldsymbol{x}_n$

When $1 - y_n \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_n < 0$, the function is a constant and thus the gradient is $\boldsymbol{0}$. On the other hand, when $1 - y_n \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_n > 0$, the function is simply $F_n(\boldsymbol{w}) = (1 - y_n \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_n)^2$, and thus by chain rule we have $\nabla F_n(\boldsymbol{w}) = -2y_n(1 - y_n \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_n)\boldsymbol{x}_n$, which is also $\boldsymbol{0}$ when $y_n \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_n$ approaches 1. Therefore, we have

$$\nabla F_n(\boldsymbol{w}) = -2y_n \max\{0, 1 - y_n \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_n\} \boldsymbol{x}_n.$$

Rubrics:

- 4 points for the derivation of the gradient. Technically we need some discussion for the case $y_n \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_n = 1$ as the solution above shows, but for simplicity it is okay if such discussion is missing as long as the final gradient is correct.
- 2 points for the SGD implementation. Do not deduct more points for wrong gradient from the earlier wrong derivation.

2. Next, consider modifying $F_n(\boldsymbol{w})$ as

$$F_n(\boldsymbol{w}) = \begin{cases} \left(\max\left\{0, 1 - \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_n\right\}\right)^2, & \text{if } y_n = 1, \\ 0.1\left(\max\left\{0, 1 + \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_n\right\}\right)^2, & \text{else.} \end{cases} \tag{4}$$

   (a) Consider a binary classification dataset of points in two dimensions as shown in Figure 1, where the red, plus signs denote samples with label $+1$, and the green, minus signs denote samples with label -1. When training a linear classifier with the modified loss in Eq. (4), which of $\boldsymbol{w}_1$ or $\boldsymbol{w}_2$ in Figure 1 do you think is more likely the resulting decision boundary? Explain briefly. (2 points)

   $\boldsymbol{w}_1$ is more likely. First note that the dataset is not linearly separable so any linear classifier will make mistakes. Next, due to the asymmetry in weighting, the new loss function does not penalize misclassifying negative points as heavily as it penalizes misclassifying positive points. Therefore, $\boldsymbol{w}_1$ is preferred over $\boldsymbol{w}_2$.

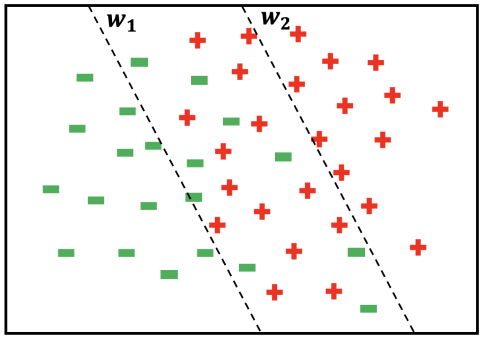


Figure 1: A binary classification task

(b) Based on your answer from the last question, give an example where one would want to modify the loss function in such a way. (2 points)

For example, in fraud detection (+1 represents a fraud), it is far more important to make sure that an actual fraud is not missed, and thus the loss function should give more weights to positive labels.

Rubrics: Any reasonable example where the one label represents a critical or high-stakes outcome and is far more important than the other label works.

(c) Similarly to Question 3.1, write down the gradient of this modified loss F_n, then fill in the missing details in the repeat-loop of the algorithm below which applies SGD to minimize F. (6 points)

Algorithm 2: SGD for minimizing modified loss Eq. (4)

Input: A training set $(x_1, y_1), \ldots, (x_N, y_N) \in \mathbb{R}^D \times \{-1, 1\}$, learning rate $\eta > 0$
Initialization: $w = \mathbf{0}$
Repeat:
  randomly pick an example (x_n, y_n)
  update $w \leftarrow w - \begin{cases} -2\eta \max\{0, 1 - w^\mathrm{T} x_n\} x_n, & \text{if } y_n = 1, \\ 0.2\eta \max\{0, 1 + w^\mathrm{T} x_n\} x_n & \text{else.} \end{cases}$

Based on previous calculation, the gradient of the modified loss is

$$\nabla F_n(w) = \begin{cases} -2 \max\{0, 1 - w^\mathrm{T} x_n\} x_n, & \text{if } y_n = 1, \\ 0.2 \max\{0, 1 + w^\mathrm{T} x_n\} x_n & \text{else.} \end{cases}$$

Rubrics:

- 4 points for the correct gradient.
- 2 points for the SGD implementation. Do not deduct more points for wrong gradient from the earlier wrong derivation.

