CSCI 567, Fall 2021
Haipeng Luo

Written Assignment #1

Due: Sep 14, 2021, 11:59 pm, PT

Instructions

Submission: Assignment submission will be via `courses.uscden.net`. By the submission date, there will be a folder set up in which you can submit your files. Please be sure to follow all directions outlined here.

You can submit multiple times, but only *the last submission* counts. That means if you finish some problems and want to submit something first and update later when you finish, that's fine. In fact you are encouraged to do this: that way, if you forget to finish the homework on time or something happens, you still get credit for whatever you have turned in.

Problem sets must be typewritten or neatly handwritten when submitted. In both cases, your submission must be a single PDF. Please also follow the rules below:

- The file should be named as `firstname_lastname_USCID.pdf` (e.g., Joe_Doe_1234567890.pdf).
- Do not have any spaces in your file name when uploading it.
- Please include your name and USCID in the header of the report as well.

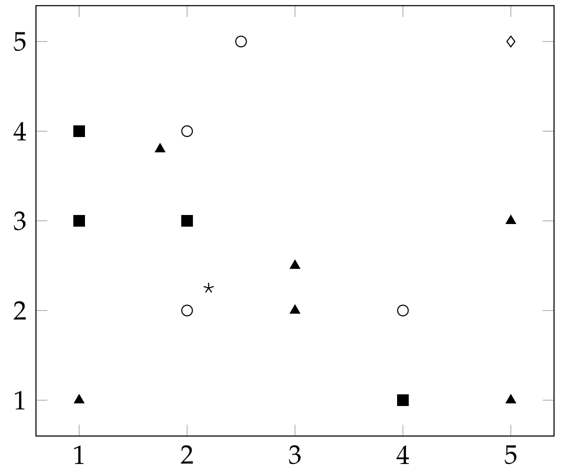
Total points: 40 points

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.
- $\|.\|$ means L2-norm unless specified otherwise i.e. $\|.\| = \|.\|_2$

Problem 1 Nearest Neighbor Classification (8 points)

For the data given below, squares, triangles, and open circles are three different classes of data in the training set and the diamond (♢) and star (*) are test points. We denote the total number of training points as N and consider K-nearest-neighbor (KNN) classifier with L2 distance.



(a) What is the prediction for the test point *star* when $K = 4$? Explain why.

(b) What is the *diamond* classified as for $K = N$? Explain why.

(c) Suppose one performs leave-one-out validation (that is, N-fold cross validation) to choose the best hyper-parameter K. List the *triangles* that are correctly classified (as a validation point) in this process for the run with $K = 1$.

Problem 2 Linear Regression (16 points)

2.1 (4 points) In the class we discussed L2 regularized least square solution defined as

$$\boldsymbol{w}_* = \operatorname*{arg\,min}_{\boldsymbol{w}\in\mathbb{R}^D} \|\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{w}\|_2^2$$

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 generalize the L2 regularization by

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

where $\boldsymbol{M} \in \mathbb{R}^{D\times D}$ is a positive definite matrix (L2 regularization is clearly a special case with $\boldsymbol{M}$ being the identity matrix scaled by λ).

Find the closed form of $\boldsymbol{w}'_*$.

2.2 (12 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 of seeing 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 probability of seeing 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 probability. 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)

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. (6 points)

Problem 3 Convergence of Perceptron Algorithm (16 points)

In this problem you need to show that when the two classes are linearly separable, the perceptron algorithm will converge. Specifically, for a binary classification dataset of N data points, where every $\mathbf{x}_i$ has a corresponding label $y_i \in \{-1, 1\}$ and is normalized: $\|\mathbf{x}_i\| = \sqrt{\mathbf{x}_i^T \mathbf{x}_i} = 1$, $\forall i \in \{1, 2, ..., N\}$, the perceptron algorithm proceeds as below:

Algorithm 1 Perceptron

```
Initialize $\mathbf{w}_1 = \mathbf{0}$
for k = 1, 2, ..., do
    Pick a data point $\mathbf{x}_i$ randomly
    Make a prediction $y = sign(\mathbf{w}_k^T \mathbf{x}_i)$ using current $\mathbf{w}_k$
    if $y \neq y_i$ then
        $\mathbf{w}_{k+1} \leftarrow \mathbf{w}_k + y_i \mathbf{x}_i$
    else
        $\mathbf{w}_{k+1} \leftarrow \mathbf{w}_k$
```

Suppose there exists an unknown hyperplane $\mathbf{w}_{opt}$ such that $\|\mathbf{w}_{opt}\| = 1$ and $y_i \mathbf{w}_{opt}^T \mathbf{x}_i \geq \gamma$, $\forall i \in [N]$ for some positive value γ. Note that this implies that the linear classifier defined by $\mathbf{w}_{opt}$ perfectly classifies all the N data points.

Following the steps below, you will show that the Perceptron algorithm makes a finite number of mistakes that is at most γ^{-2}, and therefore the algorithm must converge.

3.1 Show that if the algorithm makes a mistake, the update rule moves the weight $\mathbf{w}_k$ towards the direction of the optimal weights $\mathbf{w}_{opt}$. Specifically, suppose in iteration k we have $y_i \neq sign(\mathbf{w}_k^T \mathbf{x}_i)$. Prove

$$\mathbf{w}_{k+1}^T \mathbf{w}_{opt} \geq \mathbf{w}_k^T \mathbf{w}_{opt} + \gamma. \qquad \text{(4 points)}$$

3.2 Show that the length of the weight vector does not increase by a large amount when the algorithm makes a mistake. More specifically, if in iteration k we have $y_i \neq sign(\mathbf{w}_k^T \mathbf{x}_i)$, then

$$\|\mathbf{w}_{k+1}\|^2 \leq \|\mathbf{w}_k\|^2 + 1. \qquad \text{(4 points)}$$

3.3 Using results from Problem 3.1 and 3.2, show that for any iteration $k + 1$, with M being the total number of mistakes the algorithm has made so far for the first k iterations, we have

$$M\gamma \leq \|\mathbf{w}_{k+1}\| \leq \sqrt{M}$$

Hint: use the fact $\mathbf{a}^T \mathbf{b} \leq \|\mathbf{a}\| \|\mathbf{b}\|$ *for any two vectors* $\mathbf{a}$ *and* $\mathbf{b}$. (6 points)

3.4 Using result of Problem 3.3, conclude $M \leq \gamma^{-2}$. (2 points)