CSCI 567, Fall 2018
Haipeng Luo

Homework #1 Written Assignments

Due: 11:59 pm, Sept 16, 2018

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 (remember Murphy's Law), 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. It is strongly recommended that you typeset with LaTeX. There are many free integrated LaTeX editors that are convenient to use (e.g Overleaf, ShareLaTeX). Choose the one(s) you like the most. This tutorial Getting to Grips with LaTeX is a good start if you do not know how to use LaTeX yet.

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.

Collaboration: You may discuss with your classmates. However, you need to write your own solutions and submit separately. Also in your report, you need to list with whom you have discussed for each problem. Please consult the syllabus for what is and is not acceptable collaboration. Review the rules on academic conduct in the syllabus: a single instance of plagiarism can adversely affect you significantly more than you could stand to gain.

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 Linear Regression (12 points)

Review In the lectures, we have described the least mean square solution for linear regression as

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

where $\tilde{\boldsymbol{X}}$ is the design matrix (N rows, $D+1$ columns) and $\boldsymbol{y}$ is the N-dimensional column vector of the true values in the training data $\mathcal{D} = \{(\boldsymbol{x}_n, y_n)\}_{n=1}^N$.

1.1 We mentioned a practical challenge for linear regression: when $\tilde{\boldsymbol{X}}^{\mathrm{T}}\tilde{\boldsymbol{X}}$ is not invertible. Please use a concise mathematical statement (*in one sentence*) to summarize the relationship between the training data $\tilde{\boldsymbol{X}}$ and the dimensionality of $\boldsymbol{w}$ when this bad scenario happens. Then use this statement to explain why this scenario must happen when $N < D+1$. **(4 points)**

$$r(\tilde{\boldsymbol{X}}^{\mathrm{T}}\tilde{\boldsymbol{X}}) < D+1$$

where $r(\boldsymbol{M})$ is the rank of matrix $\boldsymbol{M}$.
Since $r(\tilde{\boldsymbol{X}}^{\mathrm{T}}\tilde{\boldsymbol{X}}) \leq \min\{N, D+1\}$, it must be smaller than $D+1$ when $N < D+1$.

1.2 In this problem we use the notation $w_0 + \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}$ for the linear model, that is, we do not append the constant feature 1 to x.

Under certain assumption, the bias w_0 has a solution being the mean of the samples

$$w_0^* = \frac{1}{N}\mathbf{1}_N^{\mathrm{T}}\boldsymbol{y} = \frac{1}{N}\sum_n y_n, \tag{2}$$

where $\mathbf{1}_N = [1, 1, \ldots, 1]^{\mathrm{T}}$ is an N-dimensional column vector whose entries are all ones.

We proved that it is true when $D = 0$ (i.e. ignore the features such that the design matrix is a column of 1's), by the following procedure:

$$w_0^* = \arg\min_{w_0} \|\boldsymbol{y} - w_0\mathbf{1}_N\|^2 \qquad \text{Residual sum of squares} \tag{3}$$

$$\mathbf{1}_N^{\mathrm{T}}(\boldsymbol{y} - w_0^*\mathbf{1}_N) = 0 \qquad \text{Taking derivatives w.r.t } w_0 \tag{4}$$

$$w_0^* = \frac{1}{N}\mathbf{1}_N^{\mathrm{T}}\boldsymbol{y} \tag{5}$$

In this Problem, we would like you to generalize the proof above to arbitrary D and arrive at a more general condition where Eqn. 2 holds.

Please follow the three-step recipe: 1) write out the residual sum of squares objective function; 2) take derivatives w.r.t. the variable you are interested in and set the gradient to 0; 3) solve w_0^* and conclude that Eqn. 2 holds if

$$\frac{1}{N}\sum_n x_{nd} = 0, \quad \forall d = 1, 2, \ldots, D, \tag{6}$$

that is, each feature has zero mean. (In fact, centering the input data to be zero mean is a common pre-processing technique used in practice.)

(8 points)

$$w_0^* = \arg\min_{w_0} \|\boldsymbol{y} - w_0 \mathbf{1}_N - \boldsymbol{X}\boldsymbol{w}\|^2 \qquad \text{Residual sum of squares} \tag{7}$$

$$\mathbf{1}_N^{\mathrm{T}}(\boldsymbol{y} - w_0^* \mathbf{1}_N - \boldsymbol{X}\boldsymbol{w}) = 0 \qquad \text{Taking derivatives w.r.t } w_0 \tag{8}$$

$$w_0^* = \frac{1}{N}(\mathbf{1}_N^{\mathrm{T}}\boldsymbol{y} - \mathbf{1}_N^{\mathrm{T}}\boldsymbol{X}\boldsymbol{w}) \qquad \text{solve for } w_0^* \tag{9}$$

$$= \frac{1}{N}\mathbf{1}_N^{\mathrm{T}}\boldsymbol{y} \qquad \frac{1}{N}\sum_n x_{nd} = 0 \Leftrightarrow \mathbf{1}_N^{\mathrm{T}}\boldsymbol{X} = \mathbf{0} \tag{10}$$

If the feature values are zero on average, the bias w_0^* is the average response of training samples.

Problem 2 Convergence of Perceptron Algorithm (12 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

while no converged **do**
  Pick a data point $\mathbf{x}_i$ randomly
  Make a prediction $y = sign(\mathbf{w}^T \mathbf{x}_i)$ using current $\mathbf{w}$
  if $y \neq y_i$ **then**
    $\mathbf{w} \leftarrow \mathbf{w} + y_i \mathbf{x}_i$

In other words, weights are updated right after the perceptron makes a mistake (weights remain unchanged if the perceptron makes no mistakes). Let the (classification) margin for a hyperplane $\mathbf{w}$ be $\gamma(\mathbf{w}) = \min_{i \in [N]} \frac{|\mathbf{w}^T \mathbf{x}_i|}{\|\mathbf{w}\|}$ (convince yourself that $\gamma(\mathbf{w})$ is the smallest distance of any data point from the hyperplane). Let $\mathbf{w}_{opt}$ be the optimal hyperplane, i.e. it linearly separates the classes with maximum margin. Note that since data is linearly separable there will always exist some $\mathbf{w}_{opt}$. Let $\gamma = \gamma(\mathbf{w}_{opt})$.

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.

2.1 Show that if the algorithm makes a mistake, the update rule moves it towards the direction of the optimal weights $\mathbf{w}_{opt}$. Specifically, denoting explicitly the updating iteration index by k, the current weight vector by $\mathbf{w}_k$, and the updated weight vector by $\mathbf{w}_{k+1}$, show that, if $y_i \mathbf{w}_k^T \mathbf{x}_i < 0$, we have

$$\mathbf{w}_{k+1}^T \mathbf{w}_{opt} \geq \mathbf{w}_k^T \mathbf{w}_{opt} + \gamma \|\mathbf{w}_{opt}\|. \tag{11}$$

Hint: Consider $(\mathbf{w}_{k+1} - \mathbf{w}_k)^T \mathbf{w}_{opt}$ *and consider the property of* $\mathbf{w}_{opt}$. **(4 points)**

$$\mathbf{w}_{k+1} = \mathbf{w}_k + y_i \mathbf{x}_i \tag{12}$$

$$\mathbf{w}_{k+1}^T \mathbf{w}_{opt} = \mathbf{w}_k^T \mathbf{w}_{opt} + y_i \mathbf{x}_i^T \mathbf{w}_{opt} \tag{13}$$

Since, $y_i \mathbf{x}_i^T \mathbf{w}_{opt} = |\mathbf{x}_i^T \mathbf{w}_{opt}|$ ($\mathbf{w}_{opt}$ perfectly separates the data), by the definition of γ

$$\mathbf{w}_{k+1}^T \mathbf{w}_{opt} \geq \mathbf{w}_k^T \mathbf{w}_{opt} + \gamma \|\mathbf{w}_{opt}\| \tag{14}$$

$$\tag{15}$$

Note: In this case, we cannot say that the update moves $\mathbf{w}$ to exactly the optimal direction, but the magnitude of projection of $\mathbf{w}$ on $\mathbf{w}_{opt}$ has increased, which means apart from other things, the component of weights in optimal direction increases.

2.2 Show that the length of updated weights does not increase by a large amount. Mathematically show that, if $y_i \mathbf{w}_k^T \mathbf{x}_i < 0$

$$\|\mathbf{w}_{k+1}\|^2 \leq \|\mathbf{w}_k\|^2 + 1. \tag{16}$$

Hint: Consider $\|\mathbf{w}_{k+1}\|^2$ *and substitute* $\mathbf{w}_{k+1}$. **(3 points)**

$$\|\mathbf{w}_{k+1}\|^2 = \mathbf{w}_{k+1}^T \mathbf{w}_{k+1} = (\mathbf{w}_k + y_i \mathbf{x}_i)^T (\mathbf{w}_k + y_i \mathbf{x}_i) \tag{17}$$

$$= \|\mathbf{w}_k\|^2 + 2 y_i \mathbf{w}_k^T \mathbf{x}_i + y_i^2 \mathbf{x}_i^T \mathbf{x}_i \tag{18}$$

Input $\mathbf{x}_i$ has norm 1 and the algorithm has made a mistake so $y_i \mathbf{w}_k^T \mathbf{x}_i \leq 0$

$$\|\mathbf{w}_{k+1}\|^2 = \|\mathbf{w}_k\|^2 + 2 y_i \mathbf{w}_k^T \mathbf{x}_i + y_i^2 \mathbf{x}_i^T \mathbf{x}_i \leq \|\mathbf{w}_k\|^2 + 1 \tag{19}$$

2.3 Assume that the initial weight vector $\mathbf{w}_0 = \mathbf{0}$ (an all-zero vector). Using results from Problem 2.1 and 2.2, show that for any iteration $k+1$, with M being the total number of mistakes the algorithm has made for the first k iterations, we have

$$\gamma M \leq \|\mathbf{w}_{k+1}\| \leq \sqrt{M} \tag{20}$$

Hint: use Cauchy-Schwartz inequality $\mathbf{a}^T \mathbf{b} \leq \|\mathbf{a}\| \|\mathbf{b}\|$ *and telescopic sum.* **(4 points)**

By repeatedly applying results from Problem 2.1 for 1 to M mistakes and summing them up.

$$\mathbf{w}_{k+1}^T \mathbf{w}_{opt} \geq \mathbf{w}_0^T \mathbf{w}_{opt} + M\gamma \|\mathbf{w}_{opt}\|$$

Since $w_0 = 0$

$$\mathbf{w}_{k+1}^T w_{opt} \geq M\gamma \|\mathbf{w}_{opt}\|$$

Due to Cauchy-Schwartz inequality,

$$M\gamma \|\mathbf{w}_o pt\| \leq \mathbf{w}_{k+1}^T \mathbf{w}_{opt} \leq \|\mathbf{w}_{k+1}\| \|\mathbf{w}_{opt}\|$$

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

Similarly, use results of Problem 2.2 repeatedly and sum to them all to conclude

$$\|\mathbf{w}_{k+1}\|^2 \leq \|\mathbf{w}_0\|^2 + M$$

Since, initial weights are 0, conclude that

$$\|\mathbf{w}_{k+1}\|^2 \leq M$$

2.4 Using result of Problem 2.3, conclude $M \leq \gamma^{-2}$. **(1 points)**

Solve $\gamma M \leq \sqrt{M}$ for M.

Problem 3 Direction of Linear Discriminant Hyperplane (8 points)

Consider linear discriminant analysis for a two-class classification problem on a dataset of N inputs $\{\mathbf{x}_1 \dots \mathbf{x}_N\}$ and corresponding labels $\{y_1 \dots y_N\}$, $y_i \in \{-1, 1\}$ $\forall i \in \{1 \dots N\}$. We say input $\mathbf{x}_i$ belongs to class $\mathcal{C}_1$ if its label y_i is 1 and it belongs to class $\mathcal{C}_{-1}$ if its label is -1. Mathematically, $\mathcal{C}_1 = \{(\mathbf{x}_i, y_i) : i \in [N], y_i = 1\}$ and $\mathcal{C}_{-1} = \{(\mathbf{x}_i, y_i) : i \in [N], y_i = -1\}$

We aim to find a separating hyperplane $\mathbf{w}$ such that if input $\mathbf{x}_i$ belongs to $\mathcal{C}_1$ then $\mathbf{w}^T\mathbf{x}_i \geq 0$ and if it belongs to $\mathcal{C}_{-1}$ then $\mathbf{w}^T\mathbf{x}_i \leq 0$. However, this might not be always possible. Instead, one way to relax the goal is to find a hyperplane $\mathbf{w}^*$ that maximizes $f(\mathbf{w}) = \sum_{i=1}^N y_i \mathbf{w}^T\mathbf{x}_i$ under the constraint $\|\mathbf{w}\| = 1$. Note that $f(\mathbf{w})$ can be arbitrarily maximized by increasing the magnitude of $\mathbf{w}$ and thus the constraint $\|\mathbf{w}\| = 1$ (or equivalently, $\|\mathbf{w}\|^2 = 1$) is important. We also assume that $\sum_{i=1}^N y_i \mathbf{x}_i \neq \mathbf{0}$ otherwise the objective $f(\mathbf{w})$ is always 0.

This can be written as a well-defined optimization problem using Lagrange multipliers (you do not have to know what this is to solve this problem). More concretely, there exists $\lambda \neq 0$ such that the hyperplane $\mathbf{w}^*$ we are looking for satisfies:

$$\mathbf{w}^* = \arg\max_{\mathbf{w} \in \mathbb{R}^D} \sum_{i=1}^N y_i \mathbf{w}^T\mathbf{x}_i - \lambda(\mathbf{w}^T\mathbf{w} - 1) \tag{21}$$

3.1 Prove the following **(3 points)**

$$\mathbf{w}^* = \frac{1}{2\lambda}\left(\sum_{i:x_i \in \mathcal{C}_1} \mathbf{x}_i - \sum_{j:x_j \in \mathcal{C}_{-1}} \mathbf{x}_j\right).$$

To find the maximum we set the gradient of $f(w) = \sum_{i=1}^N y_i \mathbf{w}^T\mathbf{x}_i + \lambda(\mathbf{w}^T\mathbf{w} - 1)$ to $\mathbf{0}$.

$$\nabla f(\mathbf{w}) = \sum_{i=1}^N y_i \mathbf{x}_i - 2\lambda\mathbf{w} = \mathbf{0}$$

$$\implies \mathbf{w}^* = \frac{1}{2\lambda}\left(\sum_{i=1}^N y_i \mathbf{x}_i\right) = \frac{1}{2\lambda}\left(\sum_{i:x_i \in \mathcal{C}_1} \mathbf{x}_i - \sum_{j:x_j \in \mathcal{C}_{-1}} \mathbf{x}_j\right)$$

3.2 Find the value of λ. **(2 points)**

Since $\|\mathbf{w}^*\| = 1$ we know $\lambda = \frac{1}{2}\left\|\sum_{i:x_i \in \mathcal{C}_1} \mathbf{x}_i - \sum_{j:x_j \in \mathcal{C}_{-1}} \mathbf{x}_j\right\|$.

3.3 In terms of minimizing the training error, can you think of one issue of our objective, i.e. maximizing $f(\mathbf{w})$? **(2 points)**

Maximizing this objective might lead to a solution that prefers having a large margin on some data points with the price of misclassifying others.