Introduction to Machine Learning, Fall 2023
Homework 4
(Due Tuesday Dec.19 at 11:59pm (CST))

1. [15 points] [Maximum Margin Classifier] Consider a data set of n d-dimensional sample points, $\{X_1, \ldots, X_n\}$. Each sample point, $X_i \in \mathbb{R}^d$, has a corresponding label, y_i, indicating to which class that point belongs. For now, we will assume that there are only two classes and that every point is either in the given class ($y_i = 1$) or not in the class ($y_i = -1$). Consider the linear decision boundary defined by the hyperplane

$$\mathcal{H} = \left\{x \in \mathbb{R}^d : x \cdot w + \alpha = 0\right\}.$$

The maximum margin classifier maximizes the distance from the linear decision boundary to the closest training point on either side of the boundary, while correctly classifying all training points.

(a) An in-class sample point is correctly classified if it is on the positive side of the decision boundary, and an out-of-class sample is correctly classified if it is on the negative side. Write a set of n constraints to ensure that all n points are correctly classified. [3 points]

(b) The maximum margin classifier aims to maximize the distance from the training points to the decision boundary. Derive the distance from a point X_i to the hyperplane $\mathcal{H}$. [3 points]

(c) Assuming all the points are correctly classified, write an inequality that relates the distance of sample point X_i to the hyperplane $\mathcal{H}$ in terms of only the normal vector w. [3 points]

(d) For the maximum margin classifier, the training points closest to the decision boundary on either side of the boundary are referred to as support vectors. What is the distance from any support vector to the decision boundary? [3 points]

(e) Using the previous parts, write an optimization problem for the maximum margin classifier. [3 points]

Solution:

(a) Since all sample points are correctly classified, so for the in-class sample points, the label $y_i = 1$ and it is on the positive side of the decision boundary, so we have $x_i \cdot w + \alpha > 0$. So $y_i(x_i \cdot w + \alpha) > 0$.

For the out-of-class sample points, the label $y_i = -1$ and it is on the negative side of the decision boundary, so we have $x_i \cdot w + \alpha < 0$. So $y_i(x_i \cdot w + \alpha) > 0$.

So above all, we have the constraints as follows:

$$y_i(x_i \cdot w + \alpha) > 0, \forall i \in \{1, 2, \cdots, n\}$$

(b) For any point X_i, suppose that the projection of X_i on the hyperplane $\mathcal{H}$ is x, then we have

$$x \cdot w + \alpha = 0$$

And since x is the projection of X_i on the hyperplane $\mathcal{H}$, so we have $(X_i - x) \perp \mathcal{H}$, which means $(X_i - x) \parallel w$. So we can suppose that $X_i - x = d\frac{w}{\|w\|}$, then we have

$$\begin{aligned}
d\frac{w}{\|w\|} &= X_i - x \\
d\frac{w^T w}{\|w\|} &= w^T(X_i - x) \text{ (multiply } w^T \text{ on both sides)} \\
d\frac{\|w\|^2}{\|w\|} &= w^T X_i - w^T x = w^T X_i + \alpha \ (w^T x + \alpha = 0) \\
d &= \frac{w^T X_i + \alpha}{\|w\|}
\end{aligned}$$

And since X_i could be in the positive side or negative side of the hyperplane $\mathcal{H}$, so d may be positive or negetive. So the distance from a point X_i to the hyperplane $\mathcal{H}$ is

$$r = |d| = \frac{|w^T X_i + \alpha|}{\|w\|}$$

So above all, the distance from a point X_i to the hyperplane $\mathcal{H}$ is

$$r = \frac{|w^T X_i + \alpha|}{\|w\|}$$

(c) Since the sample point X_i is correctly classified, so we have $y_i(x_i \cdot w + \alpha) > 0$.
But as the inequality should only relate to the normal vector w, so we could get that $x_i \cdot w + \alpha \neq 0$.

So above all, the inequality that relates the distance of sample point X_i to the hyperplane $\mathcal{H}$ in terms of only the normal vector w is

$$X_i \cdot w + \alpha \neq 0, \ \forall i \in \{1, 2, \cdots, n\}$$

(d) Suppose that the margin of the maximum margin classifier is γ.
Then for any support vector X_i, we have $\gamma = \frac{|w^T X_i + \alpha|}{\|w\|}$.
So above all, the distance from any support vector to the decision boundary is:

$$\gamma = \frac{|w^T X_i + \alpha|}{\|w\|}$$

Where X_i is any support vector.

(e) The original problem for the maximum margin classifier is

$$\begin{aligned} \max_{w,\alpha} \quad & \gamma \\ \text{subject to} \quad & y_i(X_i \cdot w + \alpha) \geq \gamma, \ \forall i \in \{1, 2, \cdots, n\} \\ & \|w\| = 1 \end{aligned} \tag{1}$$

But $\|w\| = 1$ is not a convex constraint, so we can divide both sides of the equation by γ simultaneously. And let $w' = \frac{w}{\gamma}, \alpha' = \frac{\alpha}{\gamma}$.

Since $\|w\| = 1$, so maximize γ is equivalent to minimize $\|w'\| = \frac{\|w\|}{\gamma} = \frac{1}{\gamma}$, which has the same effect as minimzing $\|w'\|^2$.
So the original problem is equivalent to

$$\begin{aligned} \min_{w',\alpha'} \quad & \|w'\|^2 \\ \text{subject to} \quad & y_i(x_i \cdot w' + \alpha') \geq 1, \ \forall i \in \{1, 2, \cdots, n\} \end{aligned} \tag{2}$$

And since $\|w'\|^2$ is a convex objective funcion, and the constrains are linear constrains. So it is an optimization problem with convex objective function and linear constraints that also maximize the margin.
So above all, the optimization problem for the maximum margin classifier could be:

$$\begin{aligned} \min_{w',\alpha'} \quad & \|w'\|^2 \\ \text{subject to} \quad & y_i(x_i \cdot w' + \alpha') \geq 1, \ \forall i \in \{1, 2, \cdots, n\} \end{aligned} \tag{3}$$

2. [15 points] Consider a dataset of n observations $\mathbf{X} \in \mathbb{R}^{n \times d}$, and our goal is to project the data onto a subspace having dimensionality p, $p < d$. Prove that PCA based on projected variance maximization is equivalent to PCA based on projected error (Euclidean error) minimization.

Solution:
Suppose that all sampled points are centered, so the sample mean is $\mu = 0$.
And suppose that $\mathbf{v}$ is the direction of the projection. Where $\mathbf{v} \in \mathbb{R}^d$ and let $\|\mathbf{v}\| = 1$.
So for each sampled point X_i, the projection of X_i on the direction $\mathbf{v}$ is $X_i \cdot \mathbf{v} = X_i^\top \mathbf{v}$.
And for the PCA problem, our goal is to find the most suitable p directions $\mathbf{v}$. We could consider them seperately, and with the method to take the most p suitable directions $\mathbf{v}$.

1. The method based on projected variance maximization:
The objective function is to maximize the projected variance, which is

$$\max_{\mathbf{v}} \frac{1}{n} \sum_{i=1}^{n} (X_i^\top \mathbf{v})^2$$

2. As for the method based on projected error minimization:
The objective function is to minimize the projected error, which is

$$\min_{\mathbf{v}} \sum_{i=1}^{n} \|X_i - (X_i^\top \mathbf{v})\mathbf{v}\|^2$$

From the vector's addition operation, we can get that $X_i - (X_i^\top \mathbf{v})\mathbf{v}$ is perpendicular to $\mathbf{v}$.
So we have $(X_i - (X_i^\top \mathbf{v})\mathbf{v}) \cdot \mathbf{v} = 0$.
So

$$\|X_i\|^2 = \|(X_i - (X_i^\top \mathbf{v})\mathbf{v}) + ((X_i^\top \mathbf{v})\mathbf{v})\|^2 = \|X_i - (X_i^\top \mathbf{v})\mathbf{v}\|^2 + \|(X_i^\top \mathbf{v})\mathbf{v}\|^2$$

Since $\|\mathbf{v}\| = 1$, so

$$\|(X_i^\top \mathbf{v})\mathbf{v}\|^2 = (X_i^\top \mathbf{v})^2 \|\mathbf{v}\|^2 = (X_i^\top \mathbf{v})^2$$

So

$$\|X_i - (X_i^\top \mathbf{v})\mathbf{v}\|^2 = \|X_i\|^2 - (X_i^\top \mathbf{v})^2$$

So the objective function is equivalent to

$$\min_{\mathbf{v}} \sum_{i=1}^{n} \|X_i - (X_i^\top \mathbf{v})\mathbf{v}\|^2 = \min_{\mathbf{v}} \sum_{i=1}^{n} \|X_i\|^2 - \sum_{i=1}^{n} (X_i^\top \mathbf{v})^2$$

Since our goal is to find the suitable $\mathbf{v}$, so the sample points X_i is fixed.
So $\sum_{i=1}^{n} \|X_i\|^2$ is a constant. And n is also a constant.
So the objective function is equivalent to

$$\min_{\mathbf{v}} \sum_{i=1}^{n} \|X_i\|^2 - \sum_{i=1}^{n} (X_i^\top \mathbf{v})^2 \Leftrightarrow \min_{\mathbf{v}} - \sum_{i=1}^{n} (X_i^\top \mathbf{v})^2 \Leftrightarrow \max_{\mathbf{v}} \sum_{i=1}^{n} (X_i^\top \mathbf{v})^2 \Leftrightarrow \max_{\mathbf{v}} \frac{1}{n} \sum_{i=1}^{n} (X_i^\top \mathbf{v})^2$$

So above all, the objective function of the method based on projected error minimization is the same as the objective function of the method based on projected variance maximization.
And they also have the same constrain that is $\|\mathbf{v}\| = 1$.
So the two method is actually the same optimization problem.
So PCA based on projected variance maximization is equivalent to PCA based on projected error minimization.

3. [15 points] [Performing PCA by Hand] Let's do principal components analysis (PCA)! Consider this sample of six points $X_i \in \mathbb{R}^2$.

$$\left\{ \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \begin{bmatrix} 2 \\ 2 \end{bmatrix} \right\}.$$

(a) [4 pts] Compute the mean of the sample points and write the centered design matrix $\dot{X}$.

Hint: The sample mean is

Hint: By subtracting the mean from each sample, we form the centered design matrix

$$\dot{X} =$$

(b) [5 pts] Find all the principal components of this sample. Write them as unit vectors.

Hint: The principal components of our dataset are the eigenvectors of the matrix

$$\dot{X}^\top \dot{X} =$$

The characteristic polynomial of this symmetric matrix is

$$\det\left(sI - X^\top X\right)$$

(c) [6 pts]
Which of those two principal components would be preferred if you use only one? [2 pts]

What information does the PCA algorithm use to decide that one principal components is better than another? [2 pts]

From an optimization point of view, why do we prefer that one? [2 pts]

Solution:
(a) Original sample matrix $X \in \mathbb{R}^{n \times d} = \mathbb{R}^{6 \times 2}$.

The sample mean is that $\mu = \frac{1}{6} \sum_{i=1}^{6} X_i = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

After subtracting the mean from each sample, we form the centered design matrix

$$\dot{X} = X - \mu = \begin{bmatrix} -1 & -1 \\ -1 & 0 \\ 0 & -1 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}$$

(b) We can calculate that

$$\dot{X}^\top \dot{X} = \begin{bmatrix} 4 & 2 \\ 2 & 4 \end{bmatrix}$$

The characteristic polynomial of this symmetric matrix is

$$\det(\lambda I - \dot{X}^\top \dot{X}) = (\lambda - 2)(\lambda - 6)$$

So the eigenvalues of $\dot{X}^\top \dot{X}$ are $\lambda_1 = 6, \lambda_2 = 2$.

For $\lambda_1 = 6$, we have the corresponding eigenvector is that $\mathbf{v}_1 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix}$.

And for $\lambda_2 = 2$, we have the corresponding eigenvector is that $\mathbf{v}_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$.

So above all, the principal components of this sample are $\frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix}$ and $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$.

(c) 1. Since $\lambda_1 = 6 > \lambda_2 = 2$, so we prefer $\mathbf{v}_1$ if we use only one principal component.

2. The PCA algorithm use the variance of the data projected onto the corresponding eigenvector $\mathbf{v}$ or the minimum projected error to decide that one principal components is better than another.
Or we can say that the PCA algorithm use the eigenvalue of the matrix $\dot{X}^\top \dot{X}$ to decide that one principal

components is better than another.

3. From an optimization point of view, we prefer $\mathbf{v}_1$ because the variance of the data projected onto $\mathbf{v}_1$ is larger than the variance of the data projected onto $\mathbf{v}_2$. And since λ is the eigenvalue of $\dot{X}^\top \dot{X}$, so

$$\begin{aligned} \dot{X}^\top \dot{X}\mathbf{v} &= \lambda \mathbf{v} \\ \mathbf{v}^\top \dot{X}^\top \dot{X}\mathbf{v} &= \mathbf{v}^\top \lambda \mathbf{v} \quad \text{(multiply} \mathbf{v}^\top \text{to the left on both sides)} \\ \mathbf{v}^\top \dot{X}^\top \dot{X}\mathbf{v} &= \lambda \quad (\mathbf{v}^\top \mathbf{v} = \|\mathbf{v}\|^2 = 1) \end{aligned} \tag{4}$$

Also, the variance of the data projected onto $\mathbf{v}$ is

$$\begin{aligned} \dot{\sigma}^2 &= \frac{1}{n}\sum_{i=1}^{n} (\dot{X_i}^\top \mathbf{v})^2 \quad \text{(the centered designed } \dot{X}_i \text{is with mean 0)} \\ &= \frac{1}{n}\sum_{i=1}^{n} \mathbf{v}^\top \dot{X}_i \dot{X_i}^\top \mathbf{v} \\ &= \mathbf{v}^\top (\frac{1}{n}\sum_{i=1}^{n} \dot{X}_i \dot{X}_i)^\top \mathbf{v} \\ &= \mathbf{v}^\top (\frac{1}{n}\dot{X}^\top \dot{X})\mathbf{v} \quad \text{(the covirance matrix of the centered designed } \dot{X} \text{is } \frac{1}{n}\dot{X}^\top \dot{X}) \\ &= \frac{1}{n}\mathbf{v}^\top \dot{X}^\top \dot{X}\mathbf{v} \end{aligned} \tag{5}$$

So $\lambda = \frac{1}{n}\dot{\sigma}^2$.
Since the sample points' number n is a constant, so we can use the eigenvalue to represent the variance of the data projected onto the corresponding eigenvector.

4. [15 points] [Backpropagation on an Arithmetic Expression] Consider an arithmetic network with the inputs a, b, and c, which computes the following sequence of operations, where $s(\gamma) = \frac{1}{1+e^{-\gamma}}$ is the logistic (sigmoid) function and $r(\gamma) = \max\{0, \gamma\}$ is the hinge function used by ReLUs.

$$d = ab \quad e = s(d) \quad f = r(a) \quad g = 3a \quad h = 2e + f + g \quad i = ch \quad j = f + i^2$$

We want to find the partial derivatives of j with respect to every other variable a through i, in backpropagation style. This means that for each variable z, we want you to write $\partial j/\partial z$ in two forms: (1) in terms of derivatives involving each variable that directly uses the value of z, and (2) in terms of the inputs and intermediate values $a \ldots i$, as simply as possible but with no derivative symbols. For example, we write

$$\frac{\partial j}{\partial i} = \frac{\mathrm{d}j}{\mathrm{d}i} = 2i \quad \text{(no chain rule needed for this one only)}$$
$$\frac{\partial j}{\partial h} = \frac{\partial j}{\partial i}\frac{\partial i}{\partial h} = 2ic \quad \text{(chain rule, then backprop the derivative expressions)}$$

(a)Now, please write expressions for $\partial j/\partial g, \partial j/\partial f, \partial j/\partial e, \partial j/\partial d, \partial j/\partial c, \partial j/\partial b$, and $\partial j/\partial a$ as we have written $\partial j/\partial h$ above. If they are needed, express the derivative $s'(\gamma)$ in terms of $s(\gamma)$ and express the derivative $r'(\gamma)$ as the indicator function $1(\gamma \geq 0)$. (Hint: f is used in two places and a is used in three, so they will need a multivariate chain rule. It might help you to draw the network as a directed graph, but it's not required.)

Solution: