EECS 553 Homework 4 Solution (FA24)

1. Subgradient methods for the optimal soft margin hyperplane

(a) Grading rubrics

- 1 pt for correct gradient/subgradient in each scenario, e.g., $y_i(\mathbf{w}^T\mathbf{x}_i+b)<1$, >1, or $=1$. Dock 0.5 pt for each minor error, e.g, missing/incorrect sign, missing constant term, etc.
- 0 pt if no effort or completely wrong.

The equation

$$J_i(\mathbf{w},b)=\frac{1}{n}(L(y_i,\mathbf{w}^T\mathbf{x}_i+b)+\frac{\lambda}{2}\|\mathbf{w}\|^2)$$

satisfies

$$\sum_{i=1}^{n}J_i(\mathbf{w},b)=\frac{1}{n}\sum_{i=1}^{n}L(y_i,\mathbf{w}^T\mathbf{x}_i+b)+\frac{\lambda}{2}\|\mathbf{w}\|^2.$$

Since the non-differentiability of the hinge loss $L(y_i,\mathbf{w}^T\mathbf{x}_i+b)=\max(0,1-y_i(\mathbf{w}^T\mathbf{x}_i+b))$ occurs when $y_i(\mathbf{w}^T\mathbf{x}_i+b)=1$, we will consider the following three regions individually:

$$\text{(I) } y_i(\mathbf{w}^T\mathbf{x}_i+b)<1, \qquad \text{(II) } y_i(\mathbf{w}^T\mathbf{x}_i+b)>1, \qquad \text{(III) } y_i(\mathbf{w}^T\mathbf{x}_i+b)=1.$$

In region I, we have $J_i(\mathbf{w},b)=\frac{1}{n}\Big(1-y_i(\mathbf{w}^T\mathbf{x}_i+b)+\frac{\lambda}{2}\|\mathbf{w}\|^2\Big)$. Since this is differentiable, the subgradient in this region is given by its gradient:

$$\mathbf{u}_i^{(\mathrm{I})}=\nabla_{\boldsymbol{\theta}}J_i(\mathbf{w},b)=\begin{bmatrix}\frac{\partial}{\partial b}J_i(\mathbf{w},b)\\ \nabla_{\mathbf{w}}J_i(\mathbf{w},b)\end{bmatrix}=\frac{1}{n}\begin{bmatrix}-y_i\\ -y_i\mathbf{x}_i+\lambda\mathbf{w}\end{bmatrix}. \tag{1}$$

In region II, since $L(y_i,\mathbf{w}^T\mathbf{x}_i+b)=0$, we simply have $J_i(\mathbf{w},b)=\frac{\lambda}{2n}\|\mathbf{w}\|^2$. Once again this is differentiable, so the subgradient in this region is also given by its gradient:

$$\mathbf{u}_i^{(\mathrm{II})}=\nabla_{\boldsymbol{\theta}}J_i(\mathbf{w},b)=\begin{bmatrix}\frac{\partial}{\partial b}J_i(\mathbf{w},b)\\ \nabla_{\mathbf{w}}J_i(\mathbf{w},b)\end{bmatrix}=\frac{1}{n}\begin{bmatrix}0\\ \lambda\mathbf{w}\end{bmatrix}. \tag{2}$$

Finally, region III involves a non-differentiable point $y_i(\mathbf{w}^T\mathbf{x}_i+b)=1$. Here we can take either (1), (2), or any convex combination of the two $\tau\mathbf{u}_i^{(\mathrm{I})}+(1-\tau)\mathbf{u}_i^{(\mathrm{II})}$, $\tau\in[0,1]$, which can be written:

$$\mathbf{u}_i^{(\mathrm{III})}=\frac{1}{n}\begin{bmatrix}-\tau y_i\\ -\tau y_i\mathbf{x}_i+\lambda\mathbf{w}\end{bmatrix}\ \text{for any } \tau\in[0,1]. \tag{3}$$

We select $\tau=0$ for our code in part **f.**, which gives us $\mathbf{u}_i^{(\mathrm{III})}=\frac{1}{n}[0,\lambda\mathbf{w}^T]^T$.

To summarize, a subgradient $\mathbf{u}_i$ for J_i with respect to $\boldsymbol{\theta}$ can be written as:

$$\mathbf{u}_i = \begin{cases} \frac{1}{n}[-y_i,\ -y_i\mathbf{x}_i^T + \lambda\mathbf{w}^T\,]^T & \text{if } y_i(\mathbf{w}^T\mathbf{x}_i + b) < 1 \\ \frac{1}{n}[-\tau y_i,\ -\tau y_i\mathbf{x}_i^T + \lambda\mathbf{w}^T\,]^T, \tau \in [0,1] & \text{if } y_i(\mathbf{w}^T\mathbf{x}_i + b) = 1 \\ \frac{1}{n}[0,\ \lambda\mathbf{w}^T\,]^T & \text{if } y_i(\mathbf{w}^T\mathbf{x}_i + b) > 1. \end{cases}$$

To receive full credit, students may select any subgradient from above.

(b) Grading rubrics

- 1.5 pts for reporting correct hyperplane, e.g., w, b within ± 0.5, and margin within ± 0.01, and data plot with the separating hyperplane.
- 1.5 pts for correct pattern (no need to check exact values) in the objective vs. iteration plot and correct objective value within ± 0.1 in the end.
- 0 pt if no effort or completely wrong.

Figure 1 shows the results for the subgradient method. The estimated hyperplane parameters are $w = [-17.8163 - 9.1171]^T$ and $b = 12.0680$, margin $\rho = 0.04996$, and the final objective function value is 0.4498.

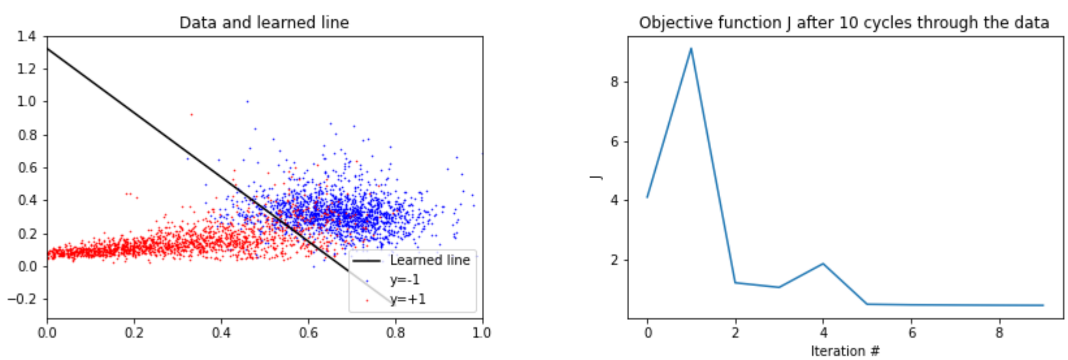


Figure 1: Results from the subgradient method.

(c) Grading rubrics

- 1.5 pts for reporting correct hyperplane, e.g., w, b within ± 0.5, and margin within ± 0.01, and data plot with the separating hyperplane.
- 1.5 pts for correct pattern (no need to check exact values) in the objective vs. iteration plot and correct objective value within ± 0.1 in the end.
- 0 pt if no effort or completely wrong.

Figure 2 shows the results for the stochastic subgradient method. The estimated hyperplane parameters are $w = [-5.8037\ \ -4.3894]^T$ and $b = 4.0535$, margin $\rho = 0.1374$ and the final objective function value is 0.2583.

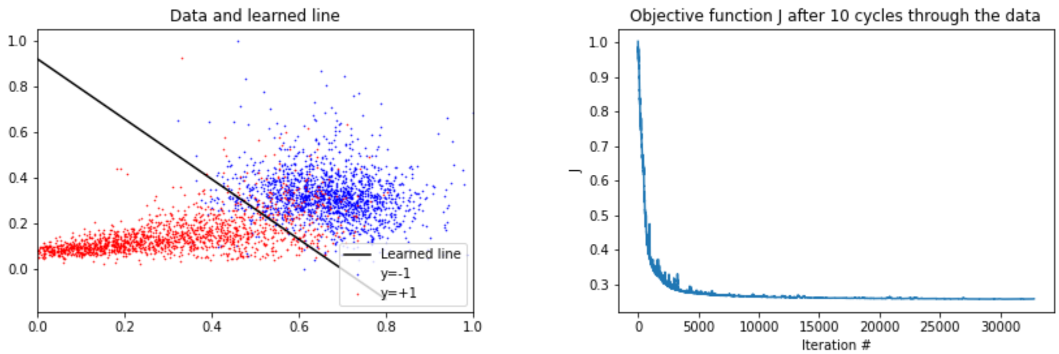


Figure 2: Results from the stochastic subgradient method.

The stochastic subgradient method converges faster than subgradient. It looks like subgradient takes until iteration 5 or 6 to converge, whereas stochastic subgradient converges after 2 or 3 cycles through the data (roughly 6000-9000 iterations).

2. Soft Thresholding Derivation

(a) Grading rubrics

- 3 pts for fully correct answer.
- 1.5 pt for correct subgradient of the MSE.
- 0.5 pt for correct subgradient of the absolute value.
- 0 pt if no effort or completely wrong.

$$
\begin{aligned}
g(\boldsymbol{w}^{(t)}, b) &= \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{w}^{(t)T}\boldsymbol{x}_i + b - y_i)^2 + \lambda||\boldsymbol{w}^{(t)}||_1 \\
&= \frac{1}{n}\sum_{i=1}^{n}(w_j x_{i,j} + \boldsymbol{w}_{-j}^{(t)T}\boldsymbol{x}_{i,-j} + b - y_i)^2 + \lambda||\boldsymbol{w}^{(t)}||_1
\end{aligned}
$$

Leverage the chain rule,

$$
\begin{aligned}
\partial_{\boldsymbol{w}_j} g(\boldsymbol{w}^{(t)}, b) &= \frac{2}{n}\sum_{i=1}^{n}(w_j x_{i,j} + \boldsymbol{w}_{-j}^{(t)T}\boldsymbol{x}_{i,-j} + b - y_i)x_{i,j} + \lambda\partial|w_j| \\
&= \frac{2}{n}\sum_{i=1}^{n} x_{i,j}^2 w_j - \frac{2}{n}\sum_{i=1}^{n} x_{i,j}(y_i - \boldsymbol{w}_{-j}^T\boldsymbol{x}_{-j} - b) + \lambda\partial|w_j|.
\end{aligned}
$$

The last step is to observe $\partial|w_j| = \text{sign}(w_j)$ if $w_j \neq 0$ and $\partial|w_j| = [-1, 1]$ if $w_j = 0$.

(b) Grading rubrics

- 1 pt for each case that is correct.
- 0 pt if no effort or completely wrong.

Case 1: $c_j > \lambda$. If $c_j > \lambda$, there is only one case where $0 \in \partial g(w_j)$, which is the case $w_j > 0$. Then $w_j = \frac{c_j - \lambda}{a_j} > 0$ and therefore $\partial_{w_j}|w_j| = 1$. Plugging this into the subdifferential yields

$$\begin{aligned}\partial_{w_j} J(\boldsymbol{w}, b) &= a_j \frac{c_j - \lambda}{a_j} - c_j + \lambda \\ &= c_j - \lambda - c_j + \lambda \\ &= 0,\end{aligned}$$

which satisfies the optimality condition.

Case 2: $c_j \in [-\lambda, \lambda]$. If $c_j \in [-\lambda, \lambda]$, there is only one case where $0 \in \partial g(w_j)$, which is the case $w_j = 0$. Therefore $\partial_{w_j}|w_j| = [-1, 1]$. Plugging this into the subdifferential yields

$$\begin{aligned}\partial_{w_j} J(\boldsymbol{w}, b) &= a_j(0) - c_j + \lambda t \\ &= -c_j + \lambda t\end{aligned}$$

for all $t \in \partial_{w_j}|w_j|$. Let $t = \frac{c_j}{\lambda}$, which is in the interval $[-1, 1]$ since $c_j \in [-\lambda, \lambda]$. Therefore, $-c_j + \lambda t \in \partial_{w_j} J(\boldsymbol{w}, b)$, and

$$\begin{aligned}-c_j + \lambda t &= -c_j + \lambda \frac{c_j}{\lambda} \\ &= 0\end{aligned}$$

which satisfies the optimality condition.

Case 3: $c_j < -\lambda$. If $c_j < -\lambda$, there is only one case where $0 \in \partial g(w_j)$, which is the case $w_j < 0$. Then $w_j = \frac{c_j + \lambda}{a_j} < 0$ and therefore $\partial_{w_j}|w_j| = -1$. Plugging this into the subdifferential yields

$$\begin{aligned}\partial_{w_j} J(\boldsymbol{w}, b) &= a_j \frac{c_j + \lambda}{a_j} - c_j - \lambda \\ &= c_j + \lambda - c_j - \lambda \\ &= 0,\end{aligned}$$

which satisfies the optimality condition.

3. Sparse Linear Regression

(a) Grading rubrics

- 1.5 pts for each plot that is approximately correct, e.g., look for similar patterns instead of exact numbers.
- 0 pt if no effort or completely wrong.

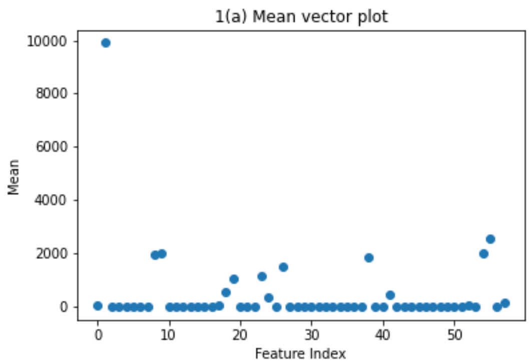
Mean, standard deviation plots:

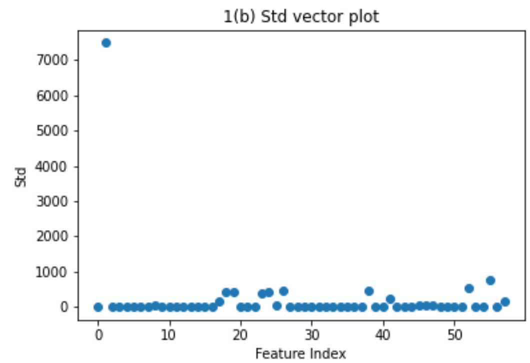
(b) Grading rubrics

- 1.5 pts for correct test MSE within ± 10 and 1.5 pts for correct number of zero weights $d - \|\boldsymbol{w}\|_0 = 3$ or 4 (3 or 4 w_i's are zeros).
- 0 pt if no effort or completely wrong.

The test MSE with $\lambda = \frac{100}{n}$ is 755.2913. $d - \|\boldsymbol{w}\|_0 = 3$ or 4 (3 or 4 w_i's are zeros)

4. Kernels (5 points each)





(a) Grading rubrics

- 3 points for a fully correct final answer. It is ok to avoid the multinomial notations.
- 2 pts for a mostly correct final answer, e.g., sign error, 1-2 missing monomials.
- 0 pt if no effort or completely wrong.

Let $\boldsymbol{u}$ and $\boldsymbol{v} \in \mathbb{R}^d$ for some $d \in \mathbb{N}$.

$$\begin{aligned} k(\boldsymbol{u}, \boldsymbol{v}) &= (\sum_{i=1}^{d} u_i v_i)^3 \\ &= \sum_{(j_1 \cdots j_d)} \binom{3}{j_1 \cdots j_d} u_1^{j_1} \cdots u_d^{j_d} v_1^{j_1} \cdots v_d^{j_d} \\ &= \langle \Phi(\boldsymbol{u}), \Phi(\boldsymbol{v}) \rangle \end{aligned}$$

where $\Phi(\boldsymbol{u}) = [\cdots, \sqrt{\binom{3}{j_1 \cdots j_d}} u_1^{j_1} \cdots u_d^{j_d}, \cdots]^T$.

(b) **Grading Rubrics**

(1) 1 point for identify the kernel matrix entries as $k(\mathbf{x}_i, \mathbf{x}_j$

(2) 1 point for invoking PSD definition (i.e compute $\mathbf{z}^T \mathbf{K} \mathbf{z}$

(3) 1 point for invoking linearity of inner product to rewrite $\mathbf{z}^T \mathbf{K} \mathbf{z}$ as inner product of the same vector hence nonnegative.

Let k be an inner product kernel, then we can express $k(\mathbf{u}, \mathbf{v}) = \langle \Phi(\mathbf{u}), \Phi(\mathbf{v}) \rangle_V$, for some inner product space V and feature map $\Phi : \mathbb{R}^d \to V$. Therefore, if K indicates the kernel matrix of k, we have for any n, any $\{\mathbf{x}_i\}_{i=1}^n \subset \mathbb{R}^d$ and any $\mathbf{z} \in \mathbb{R}^n$:

$$\begin{aligned} \mathbf{z}^T K \mathbf{z} &= \sum_{ij} z_i z_j k(\mathbf{x}_i, \mathbf{x_j}) \\ &= \sum_{ij} z_i z_j \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x_j}) \rangle \\ &= \sum_i z_i \left\langle \Phi(\mathbf{x}_i), \sum_j z_j \Phi(\mathbf{x_j}) \right\rangle \\ &= \left\langle \sum_i z_i \Phi(\mathbf{x}_i), \sum_j z_j \Phi(\mathbf{x_j}) \right\rangle \\ &\geq 0 \end{aligned}$$

where the third equality follows from the linearity property and the inequality follows from the nonnegativity property of inner products.

(c) **Grading Rubrics**

(1) 1 point for identify $\widehat{b} = \bar{y} - \widehat{\mathbf{w}}^T\bar{\mathbf{x}}$

(2) 1 points for $\widehat{b} = \bar{y} - \widetilde{\mathbf{y}}^T(\widetilde{K} + n\lambda I)^{-1}\mathbf{k}_0$

(3) 1 points for correct $\mathbf{k}_0$

From the lecture notes $\widehat{b} = \bar{y} - \widehat{\mathbf{w}}^T\bar{\mathbf{x}}$, so plugging in $\widehat{\mathbf{w}} = \frac{1}{n\lambda}\left(\mathbf{X}^T - \mathbf{X}^T(\widetilde{K} + n\lambda I)^{-1}\widetilde{K}\right)\widetilde{\mathbf{y}}$ and manipulating as in the lecture notes, we obtain $\widehat{b} = \bar{y} - \widetilde{\mathbf{y}}^T(\widetilde{K} + n\lambda I)^{-1}\mathbf{k}_0$, where $\mathbf{k}_0 \in \mathbb{R}^n$ is

$$\mathbf{k}_0 = \begin{bmatrix} \frac{1}{n}\sum_{r=1}^{n} k(\mathbf{x_1}, \mathbf{x_r}) - \frac{1}{n^2}\sum_{r=1}^{n}\sum_{s=1}^{n} k(\mathbf{x_r}, \mathbf{x_s}) \\ \vdots \\ \frac{1}{n}\sum_{r=1}^{n} k(\mathbf{x_n}, \mathbf{x_r}) - \frac{1}{n^2}\sum_{r=1}^{n}\sum_{s=1}^{n} k(\mathbf{x_r}, \mathbf{x_s}) \end{bmatrix}$$