Machine Learning Technique Homework 2

1

$$F(A,B) = \frac{1}{N}\sum_{n=1}^{N} \ln(1+\exp(-y_n(Az_n+B)))$$

$$\begin{aligned}
\nabla F(A,B) =& \frac{1}{N}\sum_{n=1}^{N} \frac{1}{1+\exp(-y_n(Az_n+B))}\nabla \exp(-y_n(Az_n+B)) \\
=& \frac{1}{N}\sum_{n=1}^{N} \frac{\exp(-y_n(Az_n+B))}{1+\exp(-y_n(Az_n+B))}\nabla(-y_n(Az_n+B)) \\
=& \frac{1}{N}\sum_{n=1}^{N} p_n \begin{bmatrix} -y_n z_n \\ -y_n \end{bmatrix}
\end{aligned}$$

2

Let $s = -y_n(Az_n+B)$

$$\begin{aligned}
& \frac{\partial}{\partial A}\nabla F(A,B) \\
=& \frac{1}{N}\sum_{n=1}^{N} \frac{\partial\theta(s)}{\partial s}\frac{\partial s}{\partial A} \begin{bmatrix} -y_n z_n \\ -y_n \end{bmatrix} \\
=& \frac{1}{N}\sum_{n=1}^{N} \frac{\exp(s)}{(1+\exp(s))^2}(-y_n z_n) \begin{bmatrix} -y_n z_n \\ -y_n \end{bmatrix} \\
=& \frac{1}{N}\sum_{n=1}^{N} \frac{1}{2}(\frac{1+2\exp(s)+\exp(s)^2}{(1+\exp(s))^2} - \frac{1+\exp(s)^2}{(1+\exp(s))^2})(-y_n z_n) \begin{bmatrix} -y_n z_n \\ -y_n \end{bmatrix} \\
=& \frac{1}{N}\sum_{n=1}^{N} \frac{1}{2}(1-(1-p_n)^2-p_n^2)(-y_n z_n) \begin{bmatrix} -y_n z_n \\ -y_n \end{bmatrix}
\end{aligned}$$

Similarily

$$
\begin{aligned}
&\frac{\partial}{\partial B}\nabla F(A,B)\\
=&\frac{1}{N}\sum_{n=1}^{N}\frac{\partial\theta(s)}{\partial s}\frac{\partial s}{\partial B}\begin{bmatrix}-y_nz_n\\-y_n\end{bmatrix}\\
=&\frac{1}{N}\sum_{n=1}^{N}\frac{\exp(s)}{(1+\exp(s))^2}(-y_n)\begin{bmatrix}-y_nz_n\\-y_n\end{bmatrix}\\
=&\frac{1}{N}\sum_{n=1}^{N}\frac{1}{2}(1-(1-p_n)^2-p_n^2)(-y_n)\begin{bmatrix}-y_nz_n\\-y_n\end{bmatrix}
\end{aligned}
$$

Combine the results above:

$$
H(F)=\frac{1}{N}\sum_{n=1}^{N}\frac{1}{2}(1-(1-p_n)^2-p_n^2)y_n^2\begin{bmatrix}z_n^2 & z_n\\ z_n & 1\end{bmatrix}
$$

3

$$
\lim_{\gamma\to\infty}\exp(-\gamma\|x-x'\|^2)=0
$$

Thus the kernel matrix is an $N\times N$ all 0 matrix, where N is the number of data.

Then the optimal β

$$
\begin{aligned}
\beta&=(\lambda I+K)^{-1}y\\
&=\frac{1}{\lambda}Iy
\end{aligned}
$$

4

$$
\lim_{\gamma\to 0}\exp(-\gamma\|x-x'\|^2)=1
$$

Thus the kernel matrix is an $N\times N$ all 1 matrxi, where N is the number of data.

Then the optimal β

$$
\beta=(\lambda I+K)^{-1}y
$$

Let $A=(\lambda I+K)^{-1}$ (A must exist since K is s.p.d.), then

$$
A_{i,j}=\begin{cases}-\frac{1}{\lambda(\lambda+N)}+\frac{1}{\lambda} & i=j\\ -\frac{1}{\lambda(\lambda+N)} & i\neq j\end{cases}
$$

Bellow is the proof. Let $B = A(\lambda I + K) = \lambda A + AK$ and A_i, K_j denote the i-th row of matrix A, K respectively.

$$
\begin{aligned}
B_{ij} =& \lambda A_{ij} + A_i K_j^T \\
=& \lambda A_{ij} + \sum_{j=1}^{N} A_{ij} && \text{since } K \text{ is an all } 1 \text{ matrix.} \\
=& \lambda A_{ij} - N \frac{1}{\lambda(\lambda+N)} + \frac{1}{\lambda} \\
=& \lambda A_{ij} + \frac{-N + \lambda + N}{\lambda(\lambda+N)} \\
=& \lambda A_{ij} + \frac{\lambda}{\lambda(\lambda+N)}
\end{aligned}
$$

If $i = j$,

$$
\begin{aligned}
& \lambda A_{ij} + \frac{\lambda N + \lambda}{\lambda(\lambda+N)} \\
&= -\frac{\lambda}{\lambda(\lambda+N)} + 1 + \frac{\lambda}{\lambda(\lambda+N)} \\
&= 1
\end{aligned}
$$

If $i \neq j$,

$$
\begin{aligned}
& \lambda A_{ij} + \frac{\lambda N + \lambda}{\lambda(\lambda+N)} \\
&= -\frac{\lambda}{\lambda(\lambda+N)} + \frac{\lambda}{\lambda(\lambda+N)} \\
&= 0
\end{aligned}
$$

Thus $B = I$, and therefore $A = (\lambda I + K)^{-1}$, where

$$
A_{i,j} = \begin{cases} -\frac{1}{\lambda(\lambda+N)} + \frac{1}{\lambda} & i = j \\ -\frac{1}{\lambda(\lambda+N)} & i \neq j \end{cases}
$$

So optimal β

$$
\begin{aligned}
\beta =& (\lambda I + K)^{-1} y \\
=& -\frac{1}{\lambda(\lambda+N)} \sum_{i}^{N} y_i \cdot e + \frac{1}{\lambda} y
\end{aligned}
$$

where e is an all 1 vector.

5

Consider the optimal $b, w, \xi^{\vee}, \xi^{\wedge}$ of original P_2:

- When $\max(0, |y_n - w^T\phi(x_n) - b| - \epsilon) = 0$

  $$\begin{aligned} &\implies |y_n - w^T\phi(x_n) - b| \leq \epsilon \\ &\implies -\epsilon - \xi_n^{\vee} \leq y_n - w^T\phi(x_n) - b \leq \epsilon + \xi_n^{\wedge} \end{aligned}$$

  for any $\xi^{\vee}, \xi^{\wedge} \geq 0$.
  Thus to minimize $(\xi_n^{\wedge})^2 + (\xi_n^{\vee})^2$, the optimal $\xi_n^{\vee}, \xi_n^{\wedge} = 0$, so

  $$(\xi_n^{\wedge})^2 + (\xi_n^{\vee})^2 = 0 = (\max(0, |y_n - w^T\phi(x_n) - b| - \epsilon))^2$$

- When $\max(0, |y_n - w^T\phi(x_n) - b| - \epsilon) = |y_n - w^T\phi(x_n) - b| - \epsilon$,
  If $y_n - w^T\phi(x_n) - b \leq 0$,

  $$-\epsilon - \xi_n^{\vee} \leq y_n - w^T\phi(x_n) - b \leq \epsilon + \xi_n^{\wedge}$$

  for all $\xi_n^{\vee} \geq |y_n - w^T\phi(x_n) - b| - \epsilon, \xi_n^{\wedge} \geq 0$.
  Thus to minimize $(\xi_n^{\wedge})^2 + (\xi_n^{\vee})^2$, the optimal $\xi_n^{\vee} = |y_n - w^T\phi(x_n) - b| - \epsilon$, $\xi_n^{\wedge} = 0$, so

  $$(\xi_n^{\wedge})^2 + (\xi_n^{\vee})^2 = (|y_n - w^T\phi(x_n) - b| - \epsilon)^2 = (\max(0, |y_n - w^T\phi(x_n) - b| - \epsilon))^2$$

  Similarly, if $y_n - w^T\phi(x_n) - b \geq 0$,
  the optimal $\xi_n^{\vee} = 0$, $\xi_n^{\wedge} = |y_n - w^T\phi(x_n) - b| - \epsilon$, so

  $$(\xi_n^{\wedge})^2 + (\xi_n^{\vee})^2 = (|y_n - w^T\phi(x_n) - b| - \epsilon)^2 = (\max(0, |y_n - w^T\phi(x_n) - b| - \epsilon))^2$$

Therefore, P_2 is equivalent to

$$\min_{b,w} \frac{1}{2}w^Tw + C\sum_{i=1}^{N}(\max(0, |y_n - w^T\phi(x_n) - b| - \epsilon))^2$$

6

Substitute w with $\sum_{n=1}^{N} \beta_n z_n$, then

$$\begin{aligned}
&\min_{b,w} \frac{1}{2} w^T w + C \sum_{n=1}^{N} (\max(0, |y_n - w^T \phi(x_n) - b| - \epsilon))^2 \\
=&\min_{b,w} \frac{1}{2} \sum_{m=1}^{N} \sum_{n=1}^{N} \beta_n \beta_m K(x_n, x_m) + C \sum_{n=1}^{N} (\max(0, |y_n - (\sum_{m=1}^{N} \beta_m)^T \phi(x_n) - b| - \epsilon))^2
\end{aligned}$$

Then

$$\begin{aligned}
&\frac{\partial F(b, \beta)}{\partial \beta_m} \\
=&\frac{1}{2} (\sum_{i=1, i \neq m}^{N} \beta_i K(x_i, x_m) + \sum_{j=1, j \neq m}^{N} \beta_j K(x_m, x_j) + 2 \beta_m K(x_m, x_m)) + \\
&C \sum_{i=1}^{N} \begin{cases} 0 & |y_n - (\sum_{m=1}^{N} \beta_m)^T \phi(x_n) - b| - \epsilon < 0 \\ 2(|y_n - (\sum_{m=1}^{N} \beta_m)^T \phi(x_n) - b| - \epsilon) g_n \beta_m \phi(x_n) & \text{otherwise} \end{cases} \\
=&\sum_{i=1}^{N} \beta_i K(x_i, x_m) + \\
&C \sum_{i=1}^{N} \begin{cases} 0 & |y_n - (\sum_{m=1}^{N} \beta_m)^T \phi(x_n) - b| - \epsilon < 0 \\ 2(|y_n - (\sum_{m=1}^{N} \beta_m)^T \phi(x_n) - b| - \epsilon) g_n \beta_m \phi(x_n) & \text{otherwise} \end{cases}
\end{aligned}$$

where $g_n = \mathrm{sgn}(y_n - (\sum_{m=1}^{N} \beta_m)^T phi(x_n) - b)$.

7

Let the two example generated at time t be $(p_t, p_t^2), (q_t, q_t^2)$. Then the learned model will be

$$\begin{aligned}
g_t(x) &= \frac{q_t^2 - p_t^2}{q_t - p_t} x - \frac{q_t^2 - p_t^2}{q_t - p_t} q_t + q_t^2 \\
&= (q_t + p_t) x - p_t q_t
\end{aligned}$$

$$\begin{aligned}
&\bar{g}(x) \\
=&\lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} g_t(x) \\
=&E(q + p) x - E(pq)
\end{aligned}$$

where

$$
\begin{aligned}
E(p) = E(q) = & \int_0^1 p \mathrm{d}p \\
= & \frac{1}{2}
\end{aligned}
$$

$$
\begin{aligned}
E(pq) = & \int_0^1 \int_0^1 pq \mathrm{d}p \mathrm{d}q \\
= & \frac{1}{4}
\end{aligned}
$$

Thus,

$$
\begin{aligned}
& \bar{g}(x) \\
= & E(q+p)x - E(pq) \\
= & x - \frac{1}{4}
\end{aligned}
$$

8

We can get those $\tilde{y}_n$ one by one.

First, query RMSE with a randomly generated hypothesis h_0.

Then generate another hypothesis h_1 such that $h_0(\tilde{x}_1) \neq h_1(\tilde{x}_1)$ and $h_0(\tilde{x}_i) = h_1(\tilde{x}_i)$ for $i = 2, 3, \cdots, \tilde{N}$. So by querying twice RMSE, we can solve $\tilde{y}_1$ from

$$
\begin{cases}
\mathrm{RMSE}(h_1)^2 = \frac{1}{\tilde{N}}(\tilde{y}_1 - h_0(\tilde{x_1}))^2 + \sum_{n=1}^{\tilde{N}}(\tilde{y}_n - h_0(\tilde{x_n}))^2 \\
\mathrm{RMSE}(h_2)^2 = \frac{1}{\tilde{N}}(\tilde{y}_2 - h_1(\tilde{x_1}))^2 + \sum_{n=1}^{\tilde{N}}(\tilde{y}_n - h_0(\tilde{x_n}))^2
\end{cases}
$$

Similarly, we can generate $h_k(x_k) \neq h_0(x_k)$ and $h_0(\tilde{x}_i) = h_1(\tilde{x}_i)$ for $i = 1, \cdots k-1, k+1, \cdots \tilde{N}$ and solve $\tilde{y}_k$ iteratively for $k = 2$ to $k = \tilde{N} - 1$ with one query to RMSE from

$$
\begin{cases}
\mathrm{RMSE}(h_0)^2 = \sum_{n=1}^{k-1} \frac{1}{\tilde{N}}(\tilde{y}_n - h_0(\tilde{x_n}))^2 + \frac{1}{\tilde{N}}(\tilde{y}_k - h_0(\tilde{x_k}))^2 + \sum_{n=k+1}^{\tilde{N}}(\tilde{y}_n - h_1(\tilde{x_n}))^2 \\
\mathrm{RMSE}(h_k)^2 = \sum_{n=1}^{k-1} \frac{1}{\tilde{N}}(\tilde{y}_n - h_0(\tilde{x_1}))^2 + \frac{1}{\tilde{N}}(\tilde{y}_k - h_k(\tilde{x_k}))^2 + \sum_{n=k+1}^{\tilde{N}}(\tilde{y}_n - h_1(\tilde{x_n}))^2
\end{cases}
$$

To solve $\tilde{y}_{\tilde{N}}$, since we have known $\tilde{y}_k$ for $k = 1, 2, 3, \cdots, k-1$ and known $\mathrm{RMSE}(h_0)$, thus no additional query is required. Therefore, totally $\tilde{N}$ queries are required.

9

$$\begin{aligned}\text{RMSE}(h) =&\sqrt{\frac{1}{N}\sum_{n=1}^{\tilde{n}}(\tilde{y}_n - h(\tilde{x}_n))^2}\\ =&\sqrt{\frac{1}{N}\sum_{n=1}^{\tilde{n}}\tilde{y}_n^2 - 2\tilde{y}_n h(\tilde{x}_n) + (h(\tilde{x}_n))^2}\\ =&\sqrt{\frac{1}{N}\sum_{n=1}^{\tilde{n}}\tilde{y}_n^2 + \frac{1}{N}\sum_{n=1}^{\tilde{n}}(h(\tilde{x}_n))^2 - \frac{2}{N}\sum_{n=1}^{\tilde{n}}\tilde{y}_n h(\tilde{x}_n)}\\ =&\sqrt{\frac{1}{N}\sum_{n=1}^{\tilde{n}}\tilde{y}_n^2 + \frac{1}{N}\sum_{n=1}^{\tilde{n}}(h(\tilde{x}_n))^2 - \frac{2}{N}g^T\tilde{y}}\end{aligned}$$

Since $\frac{1}{N}\sum_{n=1}^{\tilde{n}}(h(\tilde{x}_n))^2$ can be compute without knowing $\tilde{y}$, we can solve $g^T\tilde{y}$ from

$$\begin{cases}\text{RMSE}(h)^2 = \frac{1}{N}\sum_{n=1}^{\tilde{n}}\tilde{y}_n^2 + \frac{1}{N}\sum_{n=1}^{\tilde{n}}(h(\tilde{x}_n))^2 - \frac{2}{N}g^T\tilde{y}\\ \text{RMSE}(-h)^2 = \frac{1}{N}\sum_{n=1}^{\tilde{n}}\tilde{y}_n^2 + \frac{1}{N}\sum_{n=1}^{\tilde{n}}(h(\tilde{x}_n))^2 + \frac{2}{N}g^T\tilde{y}\end{cases}$$

Therefore, only 2 queries are required.

10

$$\begin{aligned}&\frac{\partial}{\partial\alpha}\tilde{N}\text{RMSE}^2(\sum_{k=1}^{K}\alpha_k g_k)\\ =&\frac{\partial}{\partial\alpha}\sum_{n=1}^{\tilde{N}}(\tilde{y}_n - \sum_{k=1}^{K}\alpha_k g_k(\tilde{x}_n))^2\\ =&\frac{\partial}{\partial\alpha}(\tilde{y} - g(\tilde{x})\alpha)^T(\tilde{y} - g(\tilde{x})\alpha)\\ =&2g(\tilde{x})^T(g(\tilde{x})\alpha - \tilde{y}) = 0\end{aligned}$$

where $g(\tilde{x})$ is a $\tilde{N}\times K$ matrix and $g(\tilde{x})_{i,j} = g_j(\tilde{x}_i)$. Then the optimal α is

$$(g(\tilde{x})^T g(\tilde{x}))^{-1}g(\tilde{x})^T\tilde{y}$$

Since we know all $\tilde{x}$, thus $(g(\tilde{x})^T g(\tilde{x}))^{-1}g(\tilde{x})^T$ can be calculated locally. And

$$g(\tilde{x})^T\tilde{y} = \begin{bmatrix} g_1^T\tilde{y}\\ g_2^T\tilde{y}\\ \vdots\\ g_{\tilde{K}}^T\tilde{y}\end{bmatrix}$$

Each component can be calculated with two queries. Therefore, $2\tilde{K}$ queries are required.

11

Following combinations result in minimal $E_{in} = 0$, $(\gamma = 32, \lambda = 0.001)$: $(\gamma = 32, \lambda = 1)$, $(\gamma = 32, \lambda = 1000)$, $(\gamma = 2, \lambda = 0.001)$, $(\gamma = 2, \lambda = 1)$, $(\gamma = 2, \lambda = 1000)$, $(\gamma = 0.125, \lambda = 0.001)$.

12

The minimum $E_{out}(g) = 0.39$ and the combination is $(\gamma = 0.125, \lambda = 1000)$,

13

The minumum $E_{in} = 0$, and the combinations are $(\gamma = 32, C = 1)$, $(\gamma = 32, C = 1000)$, $(\gamma = 2, C = 1)$, $(\gamma = 0.125, C = 1000)$.

14

The minumum $E_{out} = 0.42$, and the combinations is $(\gamma = 0.125, C = 1)$.

15

The minumum $E_{in} = 0.3125$, and the combinations is $\lambda = 100$. ($x_0 = 1$ is added.)

16

The minumum $E_{out} = 0.36$, and the combinations is $\lambda = 0.01, \lambda = 0.1$. ($x_0 = 1$ is added.)

17

Let

$$z_n = \begin{bmatrix} g_1(x_n) \\ g_2(x_n) \\ \vdots \\ g_T(x_n) \end{bmatrix}$$

Then

$$\min_{\alpha_t} \frac{1}{N} \sum_{n=1}^{N} \max(1 - y_n \sum_{t=1}^{T} \alpha_t g_t(x_n), 0)$$

can be rewritten as

$$\min_{\alpha_t} \frac{1}{N} \sum_{n=1}^{N} \max(1 - y_n w^T z_n, 0)$$

where $w_n = \alpha_n$, for $n = 1, 2, \cdots, N$.
Also,

$$\min_w \sum_{n=1}^{N} \max(1 - y_n \sum_{t=1}^{T} \alpha_t g_t(x_n), 0)$$

is equivalent to

$$\min_{\xi, w} \sum_{n=1}^{N} \xi_n$$

subject to $y_n(w^T x_n) \geq 1 - \xi_n$

Since when $1 - y_n \sum_{t=1}^{T} \alpha_t g_t(x_n) \geq 0$,

$$\begin{aligned}
&\max(1 - y_n \sum_{t=1}^{T} \alpha_t g_t(x_n), 0) \\
=&1 - y_n \sum_{t=1}^{T} \alpha_t g_t(x_n) \\
=&\min_{\xi_n} \xi_n && \text{subject to } y_n(w^T x_n) \geq 1 - \xi_n, \xi_n > 0
\end{aligned}$$

and when $1 - y_n \sum_{t=1}^{T} \alpha_t g_t(x_n) > 0$,

$$\begin{aligned}
&\max(1 - y_n \sum_{t=1}^{T} \alpha_t g_t(x_n), 0) \\
=&0 \\
=&\min_{\xi_n} \xi_n && \text{subject to } y_n(w^T x_n) \geq 1 - \xi_n, \xi_n > 0
\end{aligned}$$

As C is much greater than $\frac{1}{2}$,

$$\min_{\xi, w} \frac{1}{2} w^T w + C \sum_{n=1}^{N} \xi_n \approx \min_{\xi, w} C \sum_{n=1}^{N} \xi_n$$

Therefore, we can approximate those optimal α by solving optimal w with respect to those z with LIBSVM.

18

Following the previous question, before we feed data into LIBSVM, we can append some artificial $z_{N+1}, z_{N+2}, \cdots, z_{N+T}$ after the original $z_1, z_2, \cdots, z_N$, where the k-th component of z_{N+i} is

$$z_{N+i,k} = \begin{cases} 0 & i \neq k \\ 1 & i = k \end{cases}$$

and $y_{N+1} = y_{N+2} = \cdots = y_{N+T} = 1$. So we have

$$y_{N+i}(w^T z_{N+i}) = w_i \geq 1 - \xi_{N+i}$$

for $i = 1, 2, \cdots, T$, which gives penalty when $w_i < 1$.