Machine Learning, Spring 2018
Homework 5

Due on 23:59 May 16, 2018
Xiaohe He (48655395)

Understanding VC dimension (20 points)

In this part, you need to complete some mathematical proofs about VC dimension. Suppose the hypothesis set

$$\mathcal{H} = \{f(x,\alpha) = \text{sign}\ (\sin(\alpha x))|, \alpha \in \mathbb{R}\}$$

where x and f are feature and label, respectively.

- Show that $\mathcal{H}$ cannot shatter the points $x_1 = 1, x_2 = 2, x_3 = 3, x_4 = 4$. (8 points)

  (Key: Mathematically, you need to show that there exists y_1, y_2, y_3, y_4, for any $\alpha \in \mathbb{R}$, $f(x_i) \neq y_i, i = 1, 2, 3, 4$, for example, $+1, +1, -1, +1$)

- Show that the VC dimension of $\mathcal{H}$ is ∞. (Note the difference between it and the first question) (12 points)

  (Key: Mathematically, you have to prove that for any label sets $y_1, \cdots, y_m, m \in \mathbb{N}$, there exists $\alpha \in \mathbb{R}$ and $x_i, i = 1, 2, \cdots, m$ such that $f(x;\alpha)$ can generate this set of labels. Consider the points $x_i = 10^{-i}$...)

Solution:

- Assuming lables $y_1 = +1, y_2 = +1, y_3 = -1, y_4 = +1$, we can get

$$sin(\alpha) > 0$$
$$sin(2\alpha) > 0$$
$$sin(3\alpha) < 0$$
$$sin(4\alpha) > 0$$

  Since $sin(4\alpha) = 2sin(2\alpha)cos(2\alpha) > 0, sin(2\alpha) > 0$, thus $cos(2\alpha) > 0$.
  Besides $sin(3\alpha) = sin(\alpha)cos(2\alpha) + sin(2\alpha)cos(\alpha) < 0, sin(\alpha) > 0, sin(2\alpha) > 0$, thus $cos(\alpha) < 0$.
  We also have $sin(2\alpha) = 2sin(\alpha)cos(\alpha) > 0, sin(\alpha)$, thus $cos(\alpha) > 0$.
  Clearly, it's a contradiction. Hence $\mathcal{H}$ cannot shatter points $x_1 = 1, x_2 = 2, x_3 = 3, x_4 = 4$.

- Let $x_i = 10^{-i}, i = 1, \cdots, m$ and $\alpha = \pi(1 + \sum_{i=1}^{m} \frac{1-y_i}{2} 10^i)$, then we have

$$\begin{aligned}
\alpha x_j &= \pi(10^{-j} + \sum_{i=1}^{m} \frac{1-y_i}{2} 10^{i-j}) \\
&= \pi(10^{-j} + \sum_{i=1}^{j-1} \frac{1-y_i}{2} 10^{i-j} + \frac{1-y_i}{2} + \sum_{i=j+1}^{m} \frac{1-y_i}{2} 10^{i-j})
\end{aligned}$$

Where tht last term $\sum_{i=j+1}^{m} \frac{1-y_i}{2} 10^{i-j} = 2k$, k is a positive integer.
Consider $\theta = \pi(10^{-j} + \sum_{i=1}^{j-1} \frac{1-y_i}{2} 10^{i-j} + \frac{1-y_i}{2}$, we have

$$\pi(\frac{1-y_j}{2}) < \theta < \pi(1 + \frac{1-y_j}{2})$$

Thus for $y_j = +1, 0 < \theta < \pi, sign(sin(\alpha x_j)) = +1$;
for $y_j = -1, \pi < \theta < 2\pi, sign(sin(\alpha x_j)) = -1$
Hence the VC dimension of $\mathcal{H}$ is ∞.

Understanding Lasso (30 points)

Consider the following generalized Lasso problem

$$\min_{\boldsymbol{x}} \frac{1}{2} \left\| \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} \right\|_2^2 + \lambda \left\| \boldsymbol{F}\boldsymbol{x} \right\|_1, \tag{1}$$

where $\boldsymbol{A}$ is the under-determined sensing matrix, $\boldsymbol{F}$ is the transformed matrix. In particular, it can be reduced to Lasso problem if $\boldsymbol{F} = \boldsymbol{I}$. The above problem is equivalent to the following formulation

$$\begin{aligned} &\min_{\boldsymbol{x},\boldsymbol{z}} \frac{1}{2} \left\| \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} \right\|_2^2 + \lambda \left\| \boldsymbol{z} \right\|_1 \\ &\text{s.t. } \boldsymbol{F}\boldsymbol{x} = \boldsymbol{z}, \end{aligned} \tag{2}$$

and one can employ augmented Lagrangian multiplier method to solve it. Specifically, the augmented Lagrangian is

$$\mathcal{L} = \frac{1}{2} \left\| \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} \right\|_2^2 + \lambda \left\| \boldsymbol{z} \right\|_1 + < \boldsymbol{y}, \boldsymbol{F}\boldsymbol{x} - \boldsymbol{z} > + \frac{1}{2}\rho \left\| \boldsymbol{F}\boldsymbol{x} - \boldsymbol{z} \right\|_2^2,$$

which yields the ADMM algorithm, see Algorithm 1. The soft-thresholding operator $\mathbb{S}_{\frac{\lambda}{\rho}}$ is defined as

$$\mathbb{S}_{\frac{\lambda}{\rho}}(x_i) = \begin{cases} x_i - \frac{\lambda}{\rho},\ x_i \geq \frac{\lambda}{\rho} \\ 0,\ |x_i| < \frac{\lambda}{\rho} \\ x_i + \frac{\lambda}{\rho},\ x_i \leq -\frac{\lambda}{\rho}. \end{cases} \tag{3}$$

Algorithm 1 ALM for generalized Lasso problem

Input: $\boldsymbol{A}, \boldsymbol{F}, \boldsymbol{b}, \lambda, \mu$ (for augmented Lagrange multiplier)

Initialized $\rho, \boldsymbol{x}_0, \boldsymbol{z}_0$, and $\boldsymbol{y}_0$, $k = 0$
while not converged **do**
  $\boldsymbol{x}^{(k+1)} = update\ \boldsymbol{x}$?
  $\boldsymbol{z}^{(k+1)} = update\ \boldsymbol{z}$? (you may want to use $\mathbb{S}_{\frac{\lambda}{\rho}}$ element-wise)
  $\boldsymbol{y}^{(k+1)} = \boldsymbol{y}^{(k)} + \rho(\boldsymbol{F}\boldsymbol{x}^{(k+1)} - \boldsymbol{z}^{(k+1)})$
  $\rho = \rho\mu$
  $k = k + 1$
end while

Output: $\boldsymbol{x}^* = \boldsymbol{x}^{(k)}$

- Derive the steps of update $\boldsymbol{x}$ and $\boldsymbol{z}$ in Algorithm 1. (10 points)
- Complete the function `glasso.m`, and pass the test using `testglasso.m` (15 points)
- Run `demo.m` and report your MSE and PSNR. (5 points)

Solution:

- $$\boldsymbol{x}^{(k+1)} = (A^T A + \rho F^T F)^{-1}(A^T b + \rho F^T \boldsymbol{z}^k - F^T \boldsymbol{y}^k)$$
  $$\boldsymbol{z}^{(k+1)} = \mathbb{S}_{\frac{\lambda}{\rho}}(\boldsymbol{F}\boldsymbol{x}^{k+1} - \frac{1}{\rho}\boldsymbol{y}^k)$$

- $\boldsymbol{x}^{k+1}$ and $\boldsymbol{z}^{k+1}$ updated in $glasso.m$ as follows

Myfile

```
        % update x, write your formulation
        x1 = (inv(A'*A + rho * F'*F))*(A'*b + rho * F'*(z) - F'*y);
        % update z, write your formulation
        z = soft((F*x1 + y/rho), (lambda/rho));
```

And from the $testglasso.m$ we got
relative_error_x = 4.5290e-04
relative_error_z = 4.5290e-04

- Result:

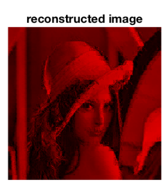


Figure 1: original image VS. restructed image

Mean square error (MSE) : 0.0049722 0.21814 0.21814
Peak signal-noise ratio (PSNR): 23.0346 6.61264 6.61264 dB

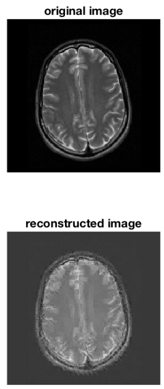


Figure 2: original image VS. restructed image

Mean square error (MSE) : 0.0024867
Peak signal-noise ratio (PSNR): 26.0438 dB

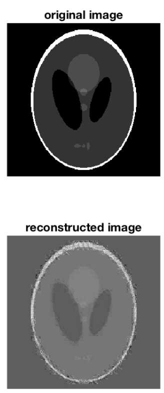


Figure 3: original image VS. restructed image

Mean square error (MSE) : 0.01004
Peak signal-noise ratio (PSNR): 19.9825 dB

Dual Formulation of the SVM (25 points)

Compared with the SVM formulation in Lecture 15 and 16, its dual problem will be much eaiser, since the original problem has so many constraints.

- Give the dual formulation of the SVM and what the KKT condition is for primal SVM. We need you show the induction of the procedure.
- There is a efficient for dual SVM problem, called Sequential Minimal Optimization. You can find many materials for this algorithm from website, it make use of the KKT conditios to solve the dual quadratic problem.
  - Give us an abstract of its principle and the pseudocode of the SMO algorithm for the dual SVM problem.

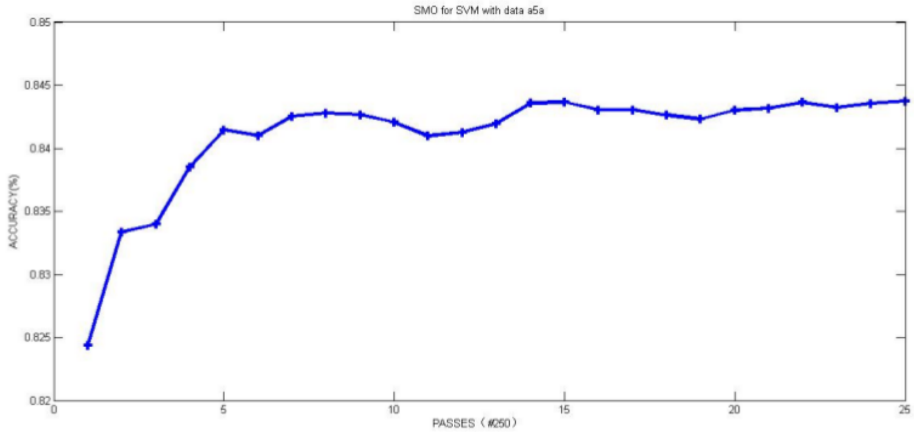


Figure 4: Accuracy demo.

– *a3a* is a data for the binary classification, show us your accuracy of the test data, like Fig 4. You need download the data *a3a* from https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/

Solution:

- Dual formulation of the SVM:

$$\begin{aligned}\underset{\alpha}{\text{minimize}}\ & \frac{1}{2}\sum_{n,m=1}^{N}\alpha_n\alpha_m\boldsymbol{y}_n y_m(\boldsymbol{x}_n^T\boldsymbol{x}_m)-\sum_{n=1}^{N}\alpha_n\\ s.t.\ \sum_{n=1}^{N} a_n y_n &= 0\\ \alpha_n &\geq 0, \qquad n=1,\dots,N\\ \boldsymbol{w}^* &= \sum_{n=1}^{N}\alpha_n^* y_n\boldsymbol{x}_n\\ b^* &= y_s-\boldsymbol{w}^T\boldsymbol{x}_s, \quad (\alpha_s^*>0)\end{aligned}.$$

Consider the Primall SVM:

$$\begin{aligned}\underset{\alpha}{\text{minimize}}\ & \frac{1}{2}||w||_2^2\\ s.t.\ & y^i(w^Tx^i-b)\geq 1.\end{aligned}$$

Lagrange function:

$$L=-\frac{1}{2}\sum_{i=1}^{l}\alpha_i y_i(x_i\cdot w+b)+\sum_{i=1}^{l}\alpha_i$$

Then the KKT condition is for primal SVM is:

$$
\begin{aligned}
\frac{\partial L}{\partial w} = w - \sum_{i=0}^{n} \alpha_i y_i x_i = 0 \\
\frac{\partial L}{\partial b} = -\sum_{i=0}^{n} \alpha_i y_i = 0 \\
y_i(\boldsymbol{w}^T \boldsymbol{x} + b) \geq 1 \\
a_i \geq 0 \\
\alpha_i \left[y_i(\boldsymbol{w}^T \boldsymbol{x} + b - 1) \right] = 0
\end{aligned}
$$

- An abstract of SMO's principles:

  Repeat 1-3 until converge:

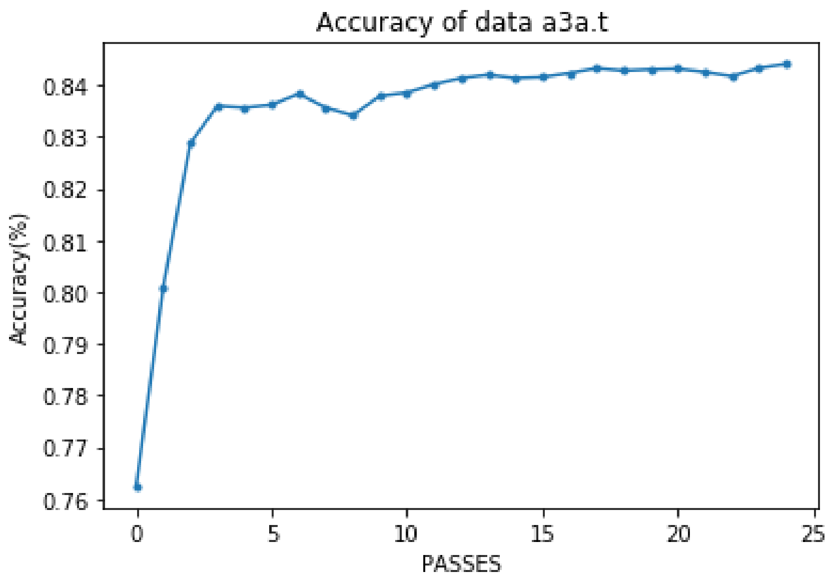
  - 1. Choose 2 Lagrange multipliers α_i and α_j;
  - 2. Optimize α_i and α_j while keep other lagrange multipliers unchanged;
  - 3. Update threshold value b according optimized α_i and α_j;

  Pesudo-Code of SMO:

Algorithm 2 Pseudo-Code for Simplified SMO

Input:
C: regularization parameters;
tol: numerical tolerance;
$maxPasses$: max iteration;
$(x^1, y^1), ..., (x^m, y^m)$: training data
Output:
$\alpha \in \mathbb{R}^m$: Lagrange multipliers for solution
$b \in \mathbb{R}$: threshold for solution
Initialized $\alpha_i = 0, \forall i$ and $b = 0$
Intiialized $count = 0$
while $count < maxPasses$ **do**
$numChangedAlpha = 0$
for $i = 1, \cdots, m$ **do**
Caculate $E_i = f(x^i) - y^i$
if $(y^i E_i < tol \&\& \alpha_i < C)||(y^i E_i > tol \&\& \alpha_i > 0)$ **then**
Select $j \neq i$ randomly
Calculate $E_j = f(x^j)y^j$
Save old α : $\alpha_i^{old} = \alpha_i, \alpha_j^{old} = \alpha_j$
Compute L and H
if $L == H$ **then**
continue to next i
end if
Compute η
if $\eta \geq 0$ **then**
continue to next i
end if
Compute and clip new value for α^j
if $||\alpha_j - \alpha_j^{old}|| < 10^{-5}$ **then**
continue to next i
end if
Determine value for α_i
Compute b_1, b_2
Compute b
$numChangedAlpha := numChangedAlpha + 1$
end if
end for
if $numChangedAlpha = 0$ **then**
$count = count + 1$
else
$count := 0$
end if
end while

- SMO accuracy pic for SVM with test data $a3a.t$ show as below, and for details please check $SMO.ipynb$



Kernel function (25 points)

Suppose we are given the following positively ("+1") labeled data points $\mathbb{R}^2$:

$$\left\{ \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \end{pmatrix} \right\},$$

and the following negatively labeled ("-1") data points in $\mathbb{R}^2$ (see Figure 5):

$$\left\{ \begin{pmatrix} 2 \\ 2 \end{pmatrix}, \begin{pmatrix} 2 \\ -2 \end{pmatrix}, \begin{pmatrix} -2 \\ 2 \end{pmatrix}, \begin{pmatrix} -2 \\ -2 \end{pmatrix} \right\}.$$

Question:

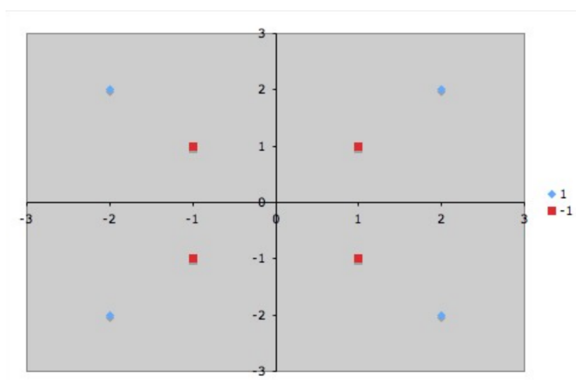


Figure 5: Blue diamonds are positive examples and red squares are negative examples.

1. Find a kernel function to map the data into a $\mathbb{R}^3$ feature space and make the new data linearly separable. (Show your kernel function please!) (5points)
2. Use SVM classifier to seperate the data. Show the SVM problem in your report. (10points) Solve this SVM problem, write down the expression of the final separating hyperplane, and plot the data and the separating hyperplane in a figure. (10points) (You can solve the SVM problem by apllying a convex problem solver.)

Solution:

- Here I choose kernel function
$$\phi([x_1, x_2]^T) = \left[x_1, x_2, x_1^2 + x_2^2\right]^T$$
which can make the new data linearly separable.

- The svm problem of kernal-svm of this problem defined as follow:
$$\underset{\alpha}{\text{minimize}} \ \frac{1}{2} \sum_{n,m=1}^{N} \alpha_n \alpha_m y_n y_m \left\langle \phi(x_m), \phi(x_n) \right\rangle - \sum_{n=1}^{N} \alpha_n$$
The form of the final hypothesis is:
$$g(x) = sign\left(\sum_{\alpha>0} (\alpha_n)^* y_n \left(x_1^n x_1 + x_2^n x_2 + \left((x_1^n)^2 + (x_2^n)^2 \right) \left((x_1)^2 + (x_2)^2 \right) \right) \right) + b^*$$
The data and the separating hyperplane shows as below:
w= [[6.24500451e-17 6.24500451e-17 -3.33284735e-01]]
b= [1.66637508]
For details please check $Kernal.ipynb$