CS446: Machine Learning **Spring 2017**

Problem Set 4

Kexin Hui *Handed In: March 11, 2017*

1. Answer to problem 1 - PAC Learning

   a. Basically, for all positive labeled examples, we calculate the L2 norm of their features in each example and return the minimum and maximum distances as r_min and r_max. Therefore, r_max should be no larger than ${r_2}^*$, and r_min should be no smaller than ${r_1}^*$.

   b. i. According to the previous problem, we have ${r_1}^* \leq r\_min \leq r\_max \leq {r_2}^*$ so that $||x||$ is bounded and we can never make mistakes on negative examples. The only possible case when we make mistakes is that the calculated L2 norm falls in the range $({r_1}^*, r_1)$ or $(r_2, {r_2}^*)$, where the hypothesis we made would classify it as positive since it is inside the bound $[{r_1}^*, {r_2}^*]$ but actually it is negative.

   ii. The probability of drawing one point where it doesn't lie in the area is $(1-\epsilon)$. Given the I.I.D. assumption, the probability of drawing m points where none of them lie in the area is $(1-\epsilon)^m$

   c. The true error larger than ϵ occurs when the point doesn't lie in the specified area where $r_1 \leq ||x|| \leq r_2$. Therefore, we want the probability of drawing m points where none of them lie in this area to be less than δ. That is,

$$\begin{aligned} (1-\epsilon)^m < \delta &\Rightarrow (e^{-\epsilon})^m < \delta \\ -m \cdot \epsilon &< ln(\delta) \\ m &> \frac{1}{\epsilon} ln(\frac{1}{\delta}) \end{aligned} \tag{1}$$

   d. We could use VC dimension and Occam's Razor to find the sample complexity as well. The VC dimension of two concentric circle is 2 since it can shatter one or two points, but never three points. Therefore, given a sample D of m examples, m is given by

$$\begin{aligned} m &> \frac{1}{\epsilon}[8VC(H)log\frac{13}{\epsilon} + 4log\frac{2}{\delta}] \\ \Rightarrow m &> \frac{1}{\epsilon}[16log\frac{13}{\epsilon} + 4log\frac{2}{\delta}] \end{aligned} \tag{2}$$

2. Answer to problem 2 - VC Dimension

   VC dimension of H is 3. It is quite straight forward that any quadratic function can shatter two points. Therefore $VC(H) \geq 2$. We can easily show no subset of H can shatter 4 points. For example, given the case where four points are colinear with $+-+-$ or $-+-+$ labelled, we cannot find a quadratic function to shatter these four points

since there are three sign flips, namely three boundaries. Therefore, $VC(H) < 4$. The last step is to show there always exists a subset that can shatter 3 points. There are 8 possible cases correlated to 3 points. Say we fix the first point to be a positive example. Then the possible labeling of these three points can be: $+++$, $++-$, $+-+$, and $+--$. There are at most two sign flips, in which the quadratic function can shatter apart. By the same token, a parabola function can shatter the three points if the first point is a negative example. Therefore, $VC(H) \geq 3$. To summarize, $VC(H) = 3$.

3. Answer to problem 3 - Kernels

   a. In the dual representation, weight vector can be written as a weighted sum of training examples.

$$\begin{aligned} w &= \sum_{1}^{m} r\alpha_i y_i x^i \\ f(x) = w \cdot x &= (\sum_{1}^{m} r\alpha_i y_i x^i)x = \sum_{1}^{m} r\alpha_i y_i (x^i \cdot x) \end{aligned} \tag{3}$$

   Here, α_i is the number of mistakes Perceptron algorithm made on x^i. In other words, instead of updating weights directly when current example is misclassified in the primal Perceptron update rule, we increment α_i by 1 every time x_i is misclassified.

   b. In order to show $K(\vec{x}, \vec{z})$ is a valid Kernel function, we can prove each term of this function is a valid Kernel since adding multiple Kernel functions still gives a Kernel function.
   First, it is quite straight forward that $\vec{x}^T\vec{z}$ is a valid Kernel since it is a linear Kernel as we learnt in the class. In addition, $(\vec{x}^T\vec{z})^3$ and $(\vec{x}^T\vec{z}+4)^2$ are also Kernels since they are a polynomial function with non-negative coefficients of an existing Kernel I just proved. As we know, multiplying with a constant value still gives a Kernel function. Therefore, $(\vec{x}^T\vec{z})^3$, $49(\vec{x}^T\vec{z}+4)^2$ and $64\vec{x}^T\vec{z}$ are all valid Kernel functions.
   To conclude, $K(\vec{x}, \vec{z}) = (\vec{x}^T\vec{z})^3 + 49(\vec{x}^T\vec{z}+4)^2 + 64\vec{x}^T\vec{z}$ is a valid Kernel function.

   c. $c(x)c(z)$ is only contributing to $K(x, z)$ when they are both 1, that is $c(x) = c(z) = 1$.

$$K(x, z) = \sum_{c \in C} c(x)c(z) = \sum_{i=0}^{k} \binom{same(x, z)}{2} \tag{4}$$

   where $same(x, z)$ is the number of variables that share the same value for both x and z. As we learn from class, $same(x, z)$ can be computed in linear time. Therefore, $K(x, z)$ can be computed in linear time as well.

4. Answer to problem 4 - SVM

   a. 1. $w = (0, -1)^T$, $\theta = 0$

2. $w = (-\frac{1}{2}, -\frac{1}{2})^T$, $\theta = 0$
3. In order to maximize the margin in SVM, we need to first find the closest two points where one is positive and another is negative. Based on observation, the closest positive and negative examples are $\{(0,2), -1\}$ and $\{(-2,0), 1\}$. The hyperplane with maximum margin should be the bisector of the line segment decided by (0,2) and (-2,0). According to math, the bisector should pass through the midpoint which is (-1,1) and perpendicular to the segment. Given the slope of the line segment to be 1, the bisector should have the slope to be -1. Thus, w_1 should be equal to w_2. Plug into the inequality $y_i(w^T x_i + \theta) \geq 1$, we can find the hyperplane function, given by

$$\begin{pmatrix} -\frac{1}{2} \\ -\frac{1}{2} \end{pmatrix}^T \cdot x = 0 \tag{5}$$

where $w = (-\frac{1}{2}, -\frac{1}{2})^T$ and $\theta = 0$.

b. 1. I = {1,6}
2. $\alpha = \{\frac{1}{4}, \frac{1}{4}\}$
3. $Objective\ function\ value = \frac{1}{4}$

c. C controls the tradeoff between large margin ($||w||$) and small hinge loss. Generally, a large C means classifications are bad, resulting in smaller margins and less training errors. A small C results in more training errors, but hopefully better true error.
When $C = \infty$, objective function value goes to infinity. Namely, the margin will be 0. No examples can fall in this margin area actually.
When $C = 1$, we take both the weight vector and empirical loss into account. So we would have a reasonably large margin with a good linear separator with reasonably less errors.
When $C = 0$, we get $||w|| = 0$ in order to minimize the objective function. We have the maximum margin and really worse training errors then.
Therefore, we could choose $C = 0$ to achieve the highest margin, which will give us the hyperplane in (a)-2.

5. Answer to problem 5 - Boosting

a. Based on observation, I chose $f_1 \equiv [x > 2]$ and $f_2 \equiv [y > 6]$.

b. As we can see, f_1 made two mistakes on example 9 and 10, and f_2 made three mistakes on example 1, 5 and 10. In order to minimize the error for this distribution, we chose $h_1 = f_1$.

c. h_1 made 2 mistakes over 10 exmaples, that is $\epsilon_0 = 0.2$.
$\alpha_0 = \frac{1}{2} ln(\frac{1-\epsilon_0}{\epsilon_0}) = ln(2) = 0.693$
Since

$$\begin{aligned} z_0 &= \sum_i D_0(i) \times exp(-\alpha_0 y_i h_1(x_i)) \\ &= 8 \times 0.1 \times e^{-\alpha_0} + 2 \times 0.1 \times e^{\alpha_0} \\ &= 0.8 \end{aligned} \tag{6}$$

		Hypothesis 1				Hypothesis 2			
i	Label	D_0	$f_1 \equiv [x > 2\ ]$	$f_2 \equiv [y > 6\ ]$	$h_1 \equiv [x > 2\ ]$	D_1	$f_1 \equiv [x > 2\ ]$	$f_2 \equiv [y > 11\ ]$	$h_2 \equiv [y > 11\ ]$
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
1	$-$	0.1	$-$	$+$	$-$	0.0625	$-$	$-$	$-$
2	$-$	0.1	$-$	$-$	$-$	0.0625	$-$	$-$	$-$
3	$+$	0.1	$+$	$+$	$+$	0.0625	$+$	$-$	$-$
4	$-$	0.1	$-$	$-$	$-$	0.0625	$-$	$-$	$-$
5	$-$	0.1	$-$	$+$	$-$	0.0625	$-$	$+$	$+$
6	$+$	0.1	$+$	$+$	$+$	0.0625	$+$	$-$	$-$
7	$+$	0.1	$+$	$+$	$+$	0.0625	$+$	$-$	$-$
8	$-$	0.1	$-$	$-$	$-$	0.0625	$-$	$-$	$-$
9	$+$	0.1	$-$	$+$	$-$	0.25	$-$	$+$	$+$
10	$-$	0.1	$+$	$+$	$+$	0.25	$+$	$-$	$-$

Table 1: Table for Boosting results

then,

$$\begin{aligned} D_1(i) &= D_0(i)/z_0 \times e^{-\alpha_0} \qquad if\ y_i = h_1(x_i) \\ &\quad D_0(i)/z_0 \times e^{\alpha_0} \qquad if\ y_i \neq h_1(x_i) \\ &= D_0(i)/0.8 \times e^{-ln(2)} = \frac{5}{8} \times 0.1 = 0.0625 \qquad if\ y_i = h_1(x_i) \\ &\quad D_0(i)/0.8 \times e^{ln(2)} = \frac{5}{2} \times 0.1 = 0.25 \qquad if\ y_i \neq h_1(x_i) \end{aligned} \tag{7}$$

In terms of the new best weak learners, we chose $f_1 \equiv [x > 2]$ and $f_2 \equiv [y] > 11$. We can calculate that $\epsilon_{f_1} = 0.25 \times 2 = 0.5$, and $\epsilon_{f_2} = 0.0625 \times 4 = 0.25$ since f_2 made mistakes on example 3, 5, 6 and 7. In this case, we would choose $h_2 = f_2$. Therefore, $\epsilon_1 = 0.25$, and $\alpha_1 = \frac{1}{2} ln(\frac{1-\epsilon_1}{\epsilon_1}) = 0.549$.

d. The final hypothesis is given by

$$\begin{aligned} H_{final} &= sgn(\sum_t \alpha_t h_t(x)) \\ &= sgn(0.693 \times [x > 2] + 0.549 \times [y > 11]) \end{aligned} \tag{8}$$