CS446: Machine Learning	**Spring 2017**

Problem Set 4

Hsiang-Wei Hwang *Handed In: March 11, 2017*

1. **[PAC Learning]**

   (a) The radii of the examples are the learning targets. If the example is positive, r_2 should be adjusted larger or equal to the radius of the example. On the other hand, if the example is negative, r_1 should be adjusted larger or equal to the radius of the example and r_2 should be larger or equal to r_1.
   An algorithm for learning:
   Initialize: first positive example x_1*, set* $r_1 = |x_1|, r_2 = |x_1|$
   for all positive examples
   if $|x| > r_2$
   set $r_2 = |x|$
   if $|x| < r_1$
   set $r_1 = |x|$

   (b) i. Only the examples satisfying $r_1 \leq |x| \leq r_2$ are considered as positive by the classifier. Thus, the region out of learned function, h_{r_1,r_2} but in target function, $h^*_{r_1^*,r_2^*}$, is the place misclassification happens which is $r_1^* \leq |x| \leq r_1$ or $r_2 < |x|_2 \leq r_2^*$.

   ii. As mentioned, the fail rate of classification for one example is ϵ. Thus, the probability that consistent with m examples is :

   $$(1-\epsilon)^m$$

   (c) Assume the set H_e is a subset of H that any function g in H_e satisfies $Error(g) > \epsilon$. The probability that g is consistent with m examples is bounded by $(1-\epsilon)^m$ which means that $P(g \in H_e \text{ consistent with m examples}) \leq (1-\epsilon)^m$. Thus, the probability we get h in H_e:

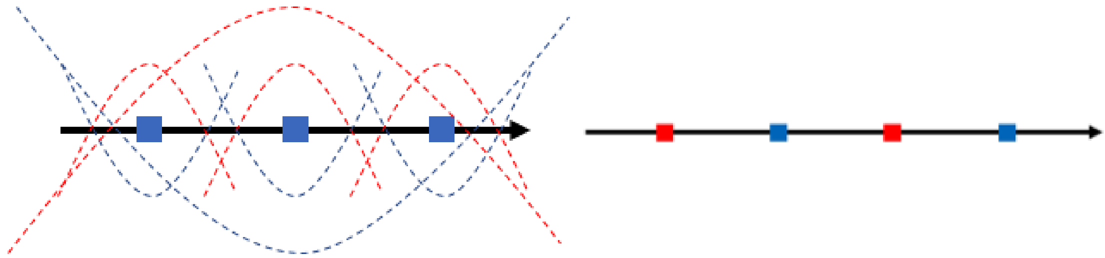
   $$P(|H_e|/|H|) * P(g \in H_e \text{ consistent with m examples}) \leq 1 * (1-\epsilon)^m < \delta$$

   $$\Rightarrow e^{-\epsilon m} < \delta \Rightarrow -\epsilon m < \ln \delta \Rightarrow m > \frac{1}{\epsilon} \ln \frac{1}{\delta}$$

   (d) VC = 2 (Similar to VC(intervals)). According to the formula mentioned in the slide,

   $$m > \frac{1}{\epsilon}\{8VC(H) \log \frac{13}{\epsilon} + 4 \log \frac{2}{\delta}\} \Rightarrow m > \frac{1}{\epsilon}\{16 \log \frac{13}{\epsilon} + 4 \log \frac{2}{\delta}\}$$

2. **[VC Dimension]**
We can adjust coefficients, a, b and c, to achieve any parabolic function required. Thus, 3-point case is achievable.



According to the figure above, the 4-point case cannot be separated by parabolic functions. Thus, VC = 3.

3. **[Kernels]**

(a) According to the slide, $\mathbf{w} = \sum_{(\mathbf{x_i},\mathbf{y_i})\in \mathbf{S}} \mathbf{r}\alpha_\mathbf{i}\mathbf{x_i}\mathbf{y_i}$, $\mathbf{y} = \mathbf{sgn}(\mathbf{w^T x})$ where r is the learning rate, α_i is the number of mistakes on the example, (x_i, y_i).

(b) Polynomial kernels functions, $K(\mathbf{x}, \mathbf{x}')$, are defined as following description:
1. Linear kernel: $K(\mathbf{x}, \mathbf{x}') = \mathbf{x}\mathbf{x}'$. 2. Polynomial kernel of degree d: $K(\mathbf{x}, \mathbf{x}') = (\mathbf{x}\mathbf{x}')^d$ (only dth-order interactions). 3. Polynomial kernel up to degree d: $K(\mathbf{x}, \mathbf{x}') = (\mathbf{x}\mathbf{x}' + c)^d$ $(c > 0)$ (all interactions of order d or lower).
Besides, kernel functions, $K(\mathbf{x}, \mathbf{x}')$, can be constructed by few methods: 1. Multiply by a constant. 2. Multiply by a function f applied to **x** and **x'**. 3. Applying a polynomial (with non-negative coefficients) to $K(\mathbf{x}, \mathbf{x}')$. 4. Exponentiating k(**x**, **x'**). 5. Add it and the other kernel function together. 6. Multiply by the other kernel function.
According to the description of polynomial kernels, we get:

$$K_1 = \vec{\mathbf{x}}^T\vec{\mathbf{z}}, K_2 = (\vec{\mathbf{x}}^T\vec{\mathbf{z}} + 4)^2, K_3 = (\vec{\mathbf{x}}^T\vec{\mathbf{z}})^3$$

According to the methods of constructing a kernel function, we can construct a kernel:

$$K' = K_3 + 49K_2 + 64K_1 = (\vec{\mathbf{x}}^T\vec{\mathbf{z}})^3 + 49(\vec{\mathbf{x}}^T\vec{\mathbf{z}} + 4)^2 + 64\vec{\mathbf{x}}^T\vec{\mathbf{z}} = K'(\vec{\mathbf{x}}, \vec{\mathbf{z}})$$

Thus, $K'(\vec{\mathbf{x}}, \vec{\mathbf{z}})$ is a kernel.

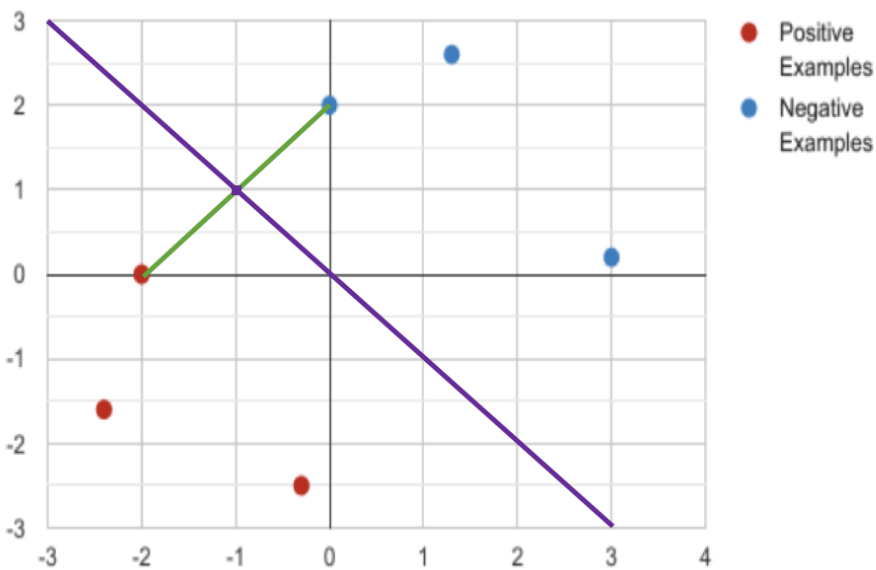
(c)
$$K(\vec{x}, \vec{z}) = \begin{cases} \binom{\vec{x}^T\vec{z}}{k} & \text{if } \vec{x}^T\vec{z} \geq k \\ 0 & \text{otherwise} \end{cases}$$

It represents the inner product of monotone conjunctions containing exactly k different variables. The inner product $\vec{x}^T\vec{z}$ means the number both variables are 1 in $\vec{x}$ and $\vec{z}$, and the monotone conjunctions containing exactly k different variables can be calculated by $\vec{x}^T\vec{z}$ choosing k that picking out k matches from the total matches to form a conjunction. For example, let k = 2, and $\vec{x}$ and $\vec{z}$

have 4 matches of 1, $\{x_2, x_5, x_6, x_8\}$ and $\{z_2, z_5, z_6, z_8\}$. We get a set that the monotone conjunctions are 1 in both vectors, $\{x_2x_5, x_2x_6, x_2x_8, x_5x_6, x_5x_8, x_6x_8\}$ and $\{z_2z_5, z_2z_6, z_2z_8, z_5z_6, z_5z_8, z_6z_8\}$, by combining all possible pairs. Thus, the number of the monotone conjunctions which inner product is 1 is 4 choose 2. This kernel can be computed in O(dm) that m means the number of the examples and d means the dimension of the vectors. Thus, this can be calculated in linear time.

4. **[SVM]**

   (a) 1. Define $\mathbf{w} = (-1, -1)$, $\theta = 0$
   2. $\mathbf{w} = (-\frac{1}{2}, -\frac{1}{2})$, $\theta = 0$
   3. We are finding a line that the distance from the point to the line represents the absolute value of $\vec{w}^T\vec{x}$ and the different side of the line represents the sign of $\vec{w}^T\vec{x}$. To maximize the margin, I consider the perpendicular bisectors of the closest pair of positive point and negative point, (-2,0) and (0,2). If I find the perpendicular bisector with distance to the points which is also the minimal distance of all points to this line, the line is the separator we are looking for.



   (b) 1. As mentioned above, the support vectors are point 1 and point 6 which means I = {1,6}.
   2. $\mathbf{w}^* = \sum \alpha_i y_i x_i \Rightarrow (-\frac{1}{2}, -\frac{1}{2}) = \alpha_1 \times 1 \times (-2, 0) + \alpha_2 \times -1 \times (0, 2)$.
   Thus, $\{\alpha_1, \alpha_2\} = \{0.25, 0.25\}$.
   3. Objective function: $\frac{1}{2}||w||^2$ ($||w||$ calculated by L2 norm).
   Objective function value $= \frac{1}{2}||w||^2 = \frac{1}{2}[(-0.5)^2 + (-0.5)^2)] = 0.25$.

   (c) **For the case** $C = 0$, it shows that the algorithm will find the w with the minimal absolute value without considering the loss. Thus, ξ_i can be really big that the result can not separate the dataset anymore. More specifically, the algorithm will get $\mathbf{w} = \mathbf{0}$ and $\xi_i \geq 1$ which minimizing the objective function being zero, but representing nothing for separating the points. Thus, the smaller C we assign the better generalization we make, but C = 0 is too general that any dataset will satisfy the answer.

For the case $C = \infty$, the algorithm will try to make the term $\sum_{j=1}^{m} \xi_i$ zero because if there is anything nonzero in that term, the objective function value will blow up to ∞. Thus, all ξ_i should be zero, and that makes the algorithm find w with the most strict margin as **Hard SVM**. **The result would be the same as I have found in (a)-2.**

For the case $C = 1$, it is a balanced option between $C = 0$ and $C = \infty$. The algorithm will get us a w between w with the most strict margin and w allowing any amount of loss. Thus, we will get a general and good solution for the dataset.

5. [**Boosting**]

(a)(b)

i	Label	Hypothesis 1				Hypothesis 2			
		D_0	$f_1 \equiv [x>2]$	$f_2 \equiv [y>6]$	$h_1 \equiv [x>2]$	D_1	$f_1 \equiv [x>9]$	$f_2 \equiv [y>11]$	$h_2 \equiv [y>11]$
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
1	−	0.1	−	+	−	$\frac{1}{16}$	−	−	−
2	−	0.1	−	−	−	$\frac{1}{16}$	−	−	−
3	+	0.1	+	+	+	$\frac{1}{16}$	−	−	−
4	−	0.1	−	−	−	$\frac{1}{16}$	−	−	−
5	−	0.1	−	+	−	$\frac{1}{16}$	−	+	+
6	−	0.1	+	+	+	$\frac{1}{4}$	−	−	−
7	+	0.1	+	+	+	$\frac{1}{16}$	+	−	−
8	−	0.1	−	−	−	$\frac{1}{16}$	−	−	−
9	+	0.1	−	+	−	$\frac{1}{4}$	−	+	+
10	+	0.1	+	+	+	$\frac{1}{16}$	−	−	−

(c)

$$\epsilon_1 = \frac{2}{10}, \alpha_1 = \frac{1}{2}\ln{(1-\epsilon_1)/\epsilon_1} = \ln 2, D_0 = \frac{1}{10},$$

$$z_0 = \sum D_0(i) \times exp(-\alpha_1 y_i h_0(xi)) = 0.1 \times (2 \times exp(\ln 2) + 8 \times exp(-\ln 2)) = 0.8$$

$$D_1(i) = D_0(i)/z_0 \times exp(-\alpha_1 y_i h_0(xi)) = \begin{cases} 0.1/0.8 \times 0.5 = \frac{1}{16} & \text{if } y_i h_0(xi) = 1 \\ 0.1/0.8 \times 2 = \frac{1}{4} & \text{otherwise} \end{cases}$$

(d)

$$\epsilon_2 = Pr_{D_1}[h_2(x_i) \neg = y_i] = \frac{1}{16} \times 4 = \frac{1}{4}, \alpha_2 = \frac{1}{2}\ln{(1-\epsilon_2)/\epsilon_2} = \frac{1}{2}\ln 3$$

Then, we combine h_1 and h_2 by α_i:

$$h_{final} = \alpha_1 h_1 + \alpha_2 h_2 = \ln 2[x > 2] + \frac{1}{2}\ln 3[y > 11]$$