CS578/STAT590: Introduction to Machine Learning	**Fall 2014**
Problem Set 3	
Gen Nishida	*Handed In: November 5, 2014*

1 Questions

1. Fitting an SVM classifier by hand (source: Machine Learning, a probabilistic perspective. K Murphy)

   (a) Write down a vector that is parallel to the optimal vector w.
   The decision boundary is perpendicular to the vector $\phi(x_2) - \phi(x_1)$.

   $$\phi(x_2) - \phi(x_1) = [1, 2, 2]^T - [1, 0, 0]^T = [0, 2, 2]^T$$

   Thus, the vector that is parallel to the optimal vector w is $[0, 1, 1]^T$.

   (b) What is the value of the margin that is achieved by this w?
   The maximum margin is the half of the distance between two points in the 3d feature space. Thus,

   $$\frac{\|\phi(x_2) - \phi(x_1)\|}{2} = \frac{\|[0, 2, 2]^T\|}{2} = \sqrt{2}$$

   (c) Solve for w, using the fact the margin is equal to $1/\|w\|$.
   Let $w = k[0, 1, 1]^T$. Then,

   $$\|w\| = k\|[0, 1, 1]^T\| = \sqrt{2}k$$

   Since the maring is $\sqrt{2}$,

   $$\sqrt{2} = \frac{1}{\|w\|} = \frac{1}{\sqrt{2}k}$$

   By solving this, we obtain $k = 1/2$. Thus, $w = [0, 1/2, 1/2]^T$.

   (d) Solve for w_0 using your value for w and Equations 1 to 3.
   By substituting w of Equations 2 and 3, we get

   $$\begin{cases} y_1(w^T\phi(x_1) + w_0) = -([0, 1/2, 1/2]^T \cdot [1, 0, 0]^T + w_0) = -w_0 \geq 1 \\ y_2(w^T\phi(x_2) + w_0) = ([0, 1/2, 1/2]^T \cdot [1, 2, 2]^T + w_0) = 2 + w_0 \geq 1 \end{cases}$$

   By solving this, we obtain

   $$-1 \leq w_0 \leq -1$$

   Thus, $w_0 = -1$.

(e) Write down the form of the discriminant function $f(x) = w_0 + w^T \phi(x)$ as an explicit function of x.

$$f(x) = w_0 + w^T \phi(x) = -1 + [0, 1/2, 1/2]^T \cdot [1, \sqrt{2}x, x^2]^T = \frac{1}{2}x^2 + \frac{1}{\sqrt{2}}x - 1$$

2. We define a concept space C that consists of the union of k disjoint intervals in a real line. A concept in C is represented therefore using $2k$ parameters: $a_1 \le b_1 \le a_2 \le b_2 \le \cdots \le a_k \le b_k$. An example (a real number) is classified as positive by such concept iff it lies in one of the intervals. Give the VC dimension of H (and prove its correctness).

   The answer is $2k$.

   Proof:

   Let $VC(k)$ be the VC dimension for k disjoint intervals in a real line. I prove by indiction that $VC(k) = 2k$ in the following. When $k = 1$, two examples have four patterns in total, and all the cases can be correctly classified (Figure.1). Thus, $VC(1) = 2$, which satisifies my hypothesis.

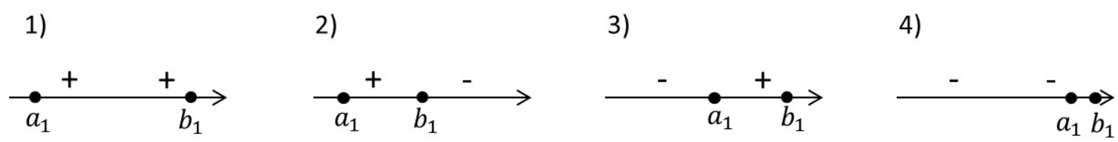

Figure 1: For the case of $k = 1$, two examples are correctly classified in all the four cases. Thus, $VC(1) = 2$.

   Given that $VC(k-1) = 2(k-1)$, we want to show that two additional examples can be correctly separated by an additional interval. Here, we do not lose generalization even if we assume that two additional examples are greater than the existing $2(k-1)$ examples that are already correctly classified. Then, there are only four cases in terms of the labels of two additional examples, and the two additional examples are correctly classified by the additional interval in all the cases (Figure. 2). Thus,

$$VC(k) = VC(k-1) + 2 = 2(k-1) + 2 = 2k$$

   Therefore, $VC(k) = 2k$ by induction.

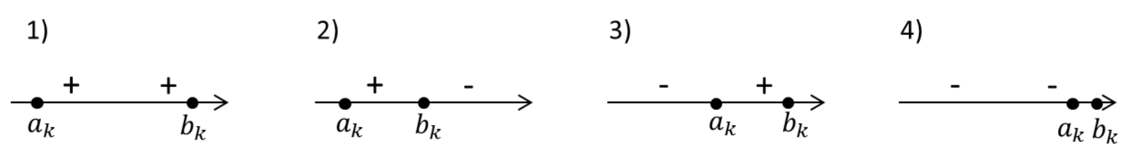

Figure 2: When adding k-th interval, two additional examples are correctly classified in all the four cases. Thus, $VC(k) = VC(k-1) + 2$.

3. The Gradient Descent (GD) algorithm

(a) Write in one sentence: what are the hyper parameters of the GD algorithm.

The hyper parameters are the initial weight vector which you can randomly initialize if you want, learning rate that is the step size of updating the weight vector, a parameter λ for a regularization term, and the convergence criteria when to stop the iteration such as the maximum number of iterations.

(b) Write in one sentence: What is the difference between $l1$ and $l2$ regularization.

$l1$ regularization uses $l1$ norm as the regularization term, which encourages the sparsity, while $l2$ regularization uses $l2$ norm.

(c) Write down the gradient descent algorithm applied to hinge loss with $l2$ regularization.

Algorithm 1 Gradient descent algorithm applied to hige loss with l2 regularization

procedure GradientDescent()
    Initialize $w^{(0)}$ randomly
    for $i = 0$ to T **do**
        $\Delta w = (0, \cdots, 0)$
        for all training data x_d for $d = 1, \cdots, D$ **do**
            for all component w_j for $j = 1, \cdots, N$ **do**
                if $y_d w^{(i)} \cdot x_d > 1$ **then**
                    $\Delta w_j += 0$
                else
                    $\Delta w_j += \eta y_d x_{dj}$
        for all component w_j for $j = 1, \cdots, N$ **do**
            $\Delta w_j -= \eta \lambda w_j$
        $w^{(i+1)} = w^{(i)} + \Delta w$
        return $w^{(i+1)}$ when it has converted

2 Programming Assignment