Machine Learning: Foundations	**Fall Semester, 2012/13**

Homework 3: Dec 23, 2012

Lecturer: Yishay Mansour

Homework number 3.

Theory question I:

1. Compute the VC dimension of the class of convex polygons in the plane with d edges.(Show that it is $\Theta(d)$. Try to get the tightest bound.)
2. Bound the Rademacher Complexity of the class L of linear functions $h_w(x) = \sum_{i=1}^{d} x_i w_i$, where $\|x\|_2 = R$ and $\|w\|_2 = \Lambda$. (Recall that $R_S(L) = E_\sigma[\frac{1}{m} \max_w \sum_{i=1}^{m} \sigma_i h_w(x_i)]$.)

Theory question II:

1. For each d, show that if two concept classes C_1 and C_2 have both VC dimension d, their union has VC dimension at most $2d+1$. (If you are unable to prove $2d+1$, prove the best upper bound you can.)
2. For each d, show an example of two concept classes C_1 and C_2 whose VC dimension is d, and whose union has VC dimension $2d+1$. (If you are unable to prove $2d+1$, prove the best lower bound you can.)
3. *Bonus:* Show that if the concept classes C_i , $1 \le i \le d^d$ have VC dimension at most d, then their union $C = \cup_{i \in [1,d^d]} C_i$ has VC dimension at most $O(d \log d)$.

Theory question III: Let k-NN(S) be the k Nearest Neighbor algorithm on sample S, which takes the majority of the closest k points.

1. Show that if in both $1 - NN(S_1)$ and $1 - NN(S_2)$ the label of point x is positive, then in $1 - NN(S_1 \cup S_2)$ the label of x is positive.
2. Show an example such that in both $3 - NN(S_1)$ and $3 - NN(S_2)$ the label of x is positive, and in $3 - NN(S_1 \cup S_2)$ the label of x is negative.

Programming assignment:

Write a simple k Nearest Neighbor implementation. Run the implementation on the `glass` data set (from: `http://archive.ics.uci.edu/ml/datasets/Glass+Identification`)

Estimate the performance of the k-NN algorithm with and without normalization and across a range of values for k (from 1 to 25). Plot the accuracy, measured using 10 fold cross validation, as a function of k (with and without normalization of features).

10-fold cross validation means that you split the data in to 10 equal size parts. You run 10 times, each time you train on different 9 parts and test on the remaining 10th part.

The homework is due in two weeks