CSCI-599 Machine Learning Theory
Assignment - 4

Sree Priyanka Uppu
(uppu@usc.edu)

1 2.5

a. Show that $|\hat{f}(i)| \leq Inf_i(f)$ with equality if and only if f is unate in the i^{th} direction.

Given, $f : \{-1, 1\}^n \rightarrow \{-1, 1\}$ and f is unate. We know that,

$$Inf_i(f) = Pr[f(x^{i\rightarrow 1}) \neq f(x^{i\rightarrow -1})] \tag{1}$$

Consider the influence for the first bit, $Inf_1(f) = Pr[f(x^{i\rightarrow 1}) \neq f(x^{i\rightarrow -1})]$
$Inf_1(f) = Pr[f(1, x_2, x_3, ..., x_n) \neq f(-1, x_2, x_3, ..., x_n)]$
Let $f(1, x^{'}) = f(1, x_2, x_3, ..., x_n)$ and $f(-1, x^{'}) = f(-1, x_2, x_3, ..., x_n)$ and substituting in the above equation,
$Inf_1(f) = Pr[f(1, x^{'}) \neq f(-1, x^{'})]$
The probability of the number of events for the RHS is $1/2^{n-1}$ Hence,
$Inf_1(f) = Pr[f(1, x^{'}) \neq f(-1, x^{'})] = \frac{1}{2^{n-1}}.\beta$
There are 2 cases where the above probability exists,and β is determined by $f(1, x^{'}) = 1$ and $f(-1, x^{'}) = -1$ or when $f(1, x^{'}) = -1$ and $f(-1, x^{'}) = 1$.
Hence, $Inf_i(f) = Pr[f(x^{i\rightarrow 1}) \neq f(x^{i\rightarrow -1})] = \frac{1}{2^{n-1}}.\beta$
Now,
$\hat{f}(i) = E_x[f(x).x_i] = Pr[f(x) = x_i] - Pr[f(x) \neq x_i]$.
For unate functions, $\hat{f}(1) = E[f(x).x_1] = \frac{1}{2^{n-1}} \sum [f(1, x^{'}) - f(-1, x^{'})]$
There are 3 possible values for the $[f(1, x^{'}) - f(-1, x^{'})]$ for the unate functions.
a. f=1 to f=1 and f=-1 to f=-1, then $[f(1, x^{'}) - f(-1, x^{'})]$=0
b. f=-1 to f=1 monotone unate , then $[f(1, x^{'}) - f(-1, x^{'})]$=2
c. f=1 to f=-1 anti-monotone unate, then $[f(1, x^{'}) - f(-1, x^{'})]$=-2
Substituting in the equation,
$\hat{f}(i) = \frac{2}{2^n}\beta = \frac{\beta}{2^{n-1}}$
For a non-unate function,
$\hat{f}(i) = E_x[f(x).x_i] = Pr[f(x) = x_i] - Pr[f(x) \neq x_i]$

and the probability depends on the function chosen and the total sum will be less than the case of unate functions. Similarly,for all values of i we can say that,
$|\hat{f}(i)| \leq Inf_i(f)$.

b. Show that the second statement of Theorem 2.33 holds even for all unate f .

We have,

$$\begin{aligned} I(f) &= \sum_{i=1}^{n} Inf_i(f) \\ &= \sum_{i=1}^{n} \frac{1}{2^{n-1}} \\ &= n.\frac{1}{2^{n-1}} \\ &= 2^{2-n} \\ &\leq \mathcal{O}(\sqrt{n}) \end{aligned} \tag{2}$$

Hence, $I(f) \leq \mathcal{O}(\sqrt{n})$

2.6 Show that linear threshold functions are unate.

Let f(x) be a non-unate linear threshold function. There exists,
a.) $f(a_0, a_1, ..., a_{i-1}, 0, a_{i+1}, ..., a_n) = 1$ and $f(a_0, a_1, ..., a_{i-1}, 1, a_{i+1}, ..., a_n) = 0$
and,
b.) $f(b_0, b_1, ..., b_{i-1}, 0, b_{i+1}, ..., b_n) = 0$ and $f(b_0, b_1, ..., b_{i-1}, 1, b_{i+1}, ..., b_n) = 1$
The linear threshold function is of the form,$f(x) = \sum_{j=1}^{n} w_j x_j - T$
We get, a.)

$$\begin{aligned} f(a_0, a_1, ..., a_{i-1}, 0, a_{i+1}, ..., a_n) &< f(a_0, a_1, ..., a_{i-1}, 1, a_{i+1}, ..., a_n) \\ (w_1 a_1 + ... + w_i.0 + ... + w_n a_n - T) &< (w_1 a_1 + ... + w_i.1 + ... + w_n a_n - T) \\ \sum_{k=1 k\neq i}^{n} w_k a_k + w_i.0 &< \sum_{k=1 k\neq i}^{n} w_k a_k + w_i.1 \\ 0 &< w_i \end{aligned} \tag{3}$$

Similarly, for case b.

$$\begin{aligned} f(b_0, b_1, ..., b_{i-1}, 0, b_{i+1}, ..., b_n) &< f(b_0, b_1, ..., b_{i-1}, 1, b_{i+1}, ..., b_n) \\ (w_1 b_1 + ... + w_i.0 + ... + w_n b_n - T) &< (w_1 b_1 + ... + w_i.1 + ... + w_n b_n - T) \\ \sum_{k=1k\neq i}^{n} w_k b_k + w_i.0 &< \sum_{k=1k\neq i}^{n} w_k b_k + w_i.1 \\ 0 &> w_i \end{aligned} \tag{4}$$

Thus, $w_i > 0$ for case a. and $w_i < 0$ for case b. By contradiction, all LTF must be unate.

2 2.21

a. Show that $Inf_i[f] = E_x[Var_i f(x)]$

For LHS, we know that,

$$Inf_i[f] = \sum_{S \ni i} \hat{f}(S)^2 \tag{5}$$

For RHS, from definition of variance of a boolean function,

$$\begin{aligned} Var(f) &= E[f(x)^2] - E[f(x)]^2 \\ &= \sum_{S \subset [n]} \hat{f}(S)^2 - \hat{f(\phi)}^2 \\ &= \sum_{S \neq \phi} \hat{f}(S)^2 (\text{From Parseval's inequality}) \end{aligned} \tag{6}$$

Now finding the expectation of variance of the i^{th} bit,

$$E_x(Var_i(f)) = \sum_{S \ni i} \hat{f}(S)^2 \tag{7}$$

Thus, $Inf_i[f] = E_x[Var_i f(x)]$.

b. Show that $Inf_i[f] = \frac{1}{2} E_{x_i, x_i'\ [-1,1]}[||f_{|x_i} - f_{|x_i'}||_2^2]$

The influence of on the i^{th} variable is :

$$Inf_i[f] = E_{x\ [-1,1]^n}[D_i f(x)^2] = ||D_i f(x)||_2^2 \tag{8}$$

From the RHS, $\frac{1}{2}E_{x_i,x_i^{'}\ [-1,1]}[||f_{|x_i} - f_{|x_i^{'}}||_2^2]$
The possible values for both $x_i and x_i^{'}$ is :(1,1),(1,-1),(-1,1) and (-1,-1).

$$\frac{1}{2}E_{x_i,x_i^{'}\ [-1,1]}[||f_{|x_i} - f_{|x_i^{'}}||_2^2]$$

Expanding the expectation to all the 4 possible cases

$$= \frac{1}{8}[||f_{|1} - f_{|1}||_2^2 + ||f_{|1} - f_{|-1}||_2^2 + ||f_{|-1} - f_{|1}||_2^2 + ||f_{|-1} - f_{|-1}||_2^2]$$

$f_{|1} - f_{|1}$ and $f_{|1} - f_{|1}$ will be 0

$$= \frac{1}{8}[||f_{|1} - f_{|-1}||_2^2 + ||f_{|-1} - f_{|1}||_2^2] \tag{9}$$

From L_2 norm both the terms are symmetric

$$= \frac{1}{4}[||f_{|1} - f_{|-1}||_2^2]$$
$$= \frac{1}{4}[||D_i f(x)||_2^2]$$
$$= [||D_i f(x)||_2^2]$$
$$= Inf_i[f]$$

Hence proved, $Inf_i[f] = \frac{1}{2}E_{x_i,x_i^{'}\ [-1,1]}[||f_{|x_i} - f_{|x_i^{'}}||_2^2]$.

3 2.22

a. Show that $Inf_i[Maj_n] = \binom{n-1}{\frac{n-1}{2}}2^{1-n}$ for all i ∈ [n].

The n-bit majority function outputs 1 if more than half of the input bits are 1 and outputs 0 otherwise.

Here, each variable has an influence $\Theta(1/\sqrt{n})$ since the probability that the other n-1 bits are set such that x_i determines the majority value is exactly $\binom{n-1}{\frac{n-1}{2}}/2^{n-1}$.

In other words, for Maj_n, the i^{th} voter has influence if and only if the other $n-1$ voters split evenly. This event happens with probability $\binom{n-1}{\frac{n-1}{2}}/2^{n-1}$.

Hence, $Inf_i[Maj_n] = \binom{n-1}{\frac{n-1}{2}}2^{1-n}$

b. Show that $Inf_1[Maj_n]$ is a decreasing function of (odd)n.

From the above we have, $Inf_i[Maj_n] = \binom{n-1}{\frac{n-1}{2}}2^{1-n}$. Substituting, i=1 we have $Inf_1[Maj_n] = \binom{n-1}{\frac{n-1}{2}}2^{1-n}$. For any odd value of n, 2^{1-n} decreases expo-

nentially as the value of n increases. And hence, $\binom{n-1}{\frac{n-1}{2}}2^{1-n}$ is a decreasing function when n is odd.

c. Use Stirling's formula $m! = (m/\epsilon)^m(\sqrt{2\pi m} + \mathcal{O}(m^{-1/2}))$ to deduce that $Inf_i[Maj_n] = \frac{\sqrt{2/\pi}}{\sqrt{n}} + \mathcal{O}(n^{-3/2})$.

$$\begin{aligned} Inf_i[Maj_n] &= \binom{n-1}{\frac{n-1}{2}}2^{1-n} \\ &= \frac{(n-1)!}{(\frac{n-1}{2})!^2} * 2^{1-n} \\ Let, \quad n-1 &= k \\ &= \frac{k!}{(k/2)!^2} * 2^{-k} \end{aligned}$$

Using Stirling's formula for the factorials,

$$\begin{aligned} &= \frac{(k/e)^k(\sqrt{2\pi k} + \mathcal{O}(k^{-1/2}))}{[(k/2e)^{k/2}(\sqrt{2\pi k/2} + \mathcal{O}((k/2)^{-1/2}))]^2} * 2^{-k} \\ &= \frac{(\sqrt{2\pi k} + \mathcal{O}(k^{-1/2}))}{[\sqrt{\pi k} + \mathcal{O}(k^{-1/2})]^2} * 2^k * 2^{-k} \\ &= \frac{(\sqrt{2\pi k} + \mathcal{O}(k^{-1/2}))}{[\pi k + \mathcal{O}(k^{-1/2})^2 + 2\sqrt{\pi k}.\mathcal{O}(k^{-1/2})]} \end{aligned}$$

Simplifying, the above for large values of n and substituting the value of k=n-1, we get

$$Inf_i[Maj_n] = \frac{\sqrt{2/\pi}}{\sqrt{n}} + \mathcal{O}(n^{-3/2}) \tag{10}$$

d. Deduce that $2/\pi \leq W^1[Maj_n] \leq 2/\pi + \mathcal{O}(n^{-1})$

From the definition of $W_k[f]$,
$W_k[f] = \sum_{|S|=k} \hat{f}(S)^2$.

For k=1,we need to find the sum of squared co-efficients all the subsets of S,

having size |S|=1.

$$\begin{aligned} W_1[Maj_n] &= \sum_{|S|=1} \hat{f}(S)^2 \\ &= \sum_{|S|=1} Inf_i[Maj_n]^2 \\ &= n.Inf_i[Maj_n]^2 \end{aligned} \tag{11}$$

Bounding with the small and large values of n we have,
$2/\pi \leq W^1[Maj_n] \leq 2/\pi + \mathcal{O}(n^{-1})$

e. Deduce that $\sqrt{2/\pi}\sqrt{n} \leq I[Maj_n] \leq \sqrt{2/\pi}\sqrt{n} + \mathcal{O}(n^{-1/2})$.

From the definition, $I[f] = \sum Inf_i(f)$.
We get, $I[Maj_n] = \sum_{i=0}^{n} Inf_i[Maj_n] = n.Inf_i[Maj_n]$
Similar to the above, bounding for the small and large values of n we have,
$\sqrt{2n/\pi} \leq I[Maj_n] \leq \sqrt{2n/\pi} + \mathcal{O}(n^{-1/2})$

f. Suppose n is even and $f : \{-1,1\}^n \to \{-1,1\}$ is a majority function. Show that $I[f] = I[Maj_{n-1}] = \sqrt{2/\pi}\sqrt{n} + \mathcal{O}(n^{-1/2})$.

From the definition of Majority function,
$Maj_n(x) = sgn(x_1 + x_2 + ... + x_n)$
and the Maj function outputs a non-zero value.
Let m be odd and m+1=n and n is even. By introducing the $(m+1)^{th}$ value, the sign of the Maj function does not change. If the $(m+1)^{th}$ variable if of the same sign as that of the $Inf(Maj_m)$, the sign would not change, else the value of the Maj function becomes 0(and that was not defined). Thus, the introduction of new variable would not affect the function.

$$\begin{aligned} I[Maj_n] &= \sum_{i=0}^{n-1} Inf_i[Maj_n] + Inf_n[Maj_n] \\ &= \sum_{i=0}^{n-1} Inf_i[Maj_n] + 0 \\ &= I[Maj_{n-1}] \end{aligned} \tag{12}$$

Hence, $I[f] = I[Maj_{n-1}] = \sqrt{2/\pi}\sqrt{n} + \mathcal{O}(n^{-1/2})$.

4 3.40 Suppose A learns C from random examples with error $\epsilon/4$ in time T with probability at least 9/10.

a. After producing a hypothesis h on target $f : \{-1, 1\}^n \rightarrow \{-1, 1\}$, show that A can check whether h is a good hypothesis in time poly (n,T,$1/\epsilon$).log ($1/\delta$). Specifically, except with probability at most δ, A should output "YES" if $dist(f, h) \leq \epsilon/2$ and "NO" if $dist(f, h) > \epsilon$.

We have a learning algorithm A, such that for any target concept $c \in C$ and any distribution D, A outputs h such that error $\epsilon/4$ with probability at least 9/10.

We now show that, if we are willing to tolerate a slightly higher error say $\epsilon + \gamma$ we can achieve arbitrarily high confidence $1 - \delta$, which implies arbitrarily smaller δ value.

Let us generate a new algorithm, $A^{'}$, which will be simulated from A by running it a total of k times using an independent sample from EX(c,D) for each simulation. Let $h_1^{'}, h_2^{'}, ..., h_k^{'}$ be the list of hypotheses output from each run. Since, the simulations are independent the probability of all the $h_1^{'}, h_2^{'}, ..., h_k^{'}$ having error greater than $\epsilon/4$ is atmost $(1 - \delta_0)^k$. Solving, $(1 - \delta_0)^k \leq \delta/2$ yields $k \geq (1/\delta_0)ln(1/\delta)$ for the choice of k.

Choose a hypothesis $h^{'}$, which has an error atmost $\epsilon/4+\gamma$ with probability atleast $1-\delta/2$. This is accomplished by drawing a sufficiently large sample of labeled examples S from EX(c,D) and choosing $h_i^{'}$ that makes fewest mistakes on S. We have to choose a large sample size so that with confidence at least $1-\delta/2$, the empirical error of each $h_j^{'}$, is within $\gamma/2$ of its true error($h_j^{'}$) with respect to c and D. This ensures that we get a $h_i^{'}$, with confidence $1 - \delta/2$ and the error is atmost $\epsilon/4 + \gamma$.

By Chernoff bounds, the empirical error of $h_j^{'}$ on S is an estimate for error($h_j^{'}$) that is accurate to an additive factor of $\gamma/2$ with confidence atleast $1 - \delta/2k$ provided that the number of examples m in S, satisfies $m \geq (c_0/\gamma^2)log(2k/\delta)$ for some constant $c_0 > 0$.

By union bound, the probability that any of the k hypotheses has empirical error on S deviating from its true error by more than $\gamma/2$ is atmost $k(\delta/2k) = \delta/2$.

We have, $dist(h, h^{'}) = Pr[h(x) \neq h^{'}(x)]$.

If the $dist(h, h^{'}) \leq \epsilon/4 then$,

$$\begin{aligned} dist(f,h) &= Pr[f(x) \neq h(x)] \\ &= Pr[f(x) \neq h^{'}(x)] + Pr[h(x) \neq h^{'}(x)] \\ &= dist(f,h^{'}) + dist(h^{'},h) \\ &\leq \epsilon/4 + \epsilon/4 \\ &\leq \epsilon/2 \end{aligned} \tag{13}$$

If the $dist(h^{'}, h) \leq \epsilon/4$ then we output "YES".

If the $dist(h^{'}, h) \geq 3\epsilon/4$ then we output "NO", by inclusive of Chernoff bound additive error.

The time to calculate $dist(h^{'}, h)$ is $poly(n, T, 1/\epsilon).log(1/\delta)$

Hence, poly(T) is the time taken to evaluate a hypothesis, we can conclude that the running time is $poly(n, T, 1/\epsilon).log(1/\delta)$.

Hence, in the above way we can check if h is a good or a bad hypothesis.

b. Show that for every $\delta \in (0, 1/2]$ there is a learning algorithm that learns C with error ϵ in time poly(n,T,$1/\epsilon$).$log(1/\delta)$ with probability atleast 1-δ.

As shown above, by varying the value of γ we can change the value of error, ϵ to get the desired value for $\delta \in (0, 1/2]$ in the boosting of the confidence parameter. Also, the algorithm takes poly(T) time for each run we can conclude that poly(n,T,$1/\epsilon$).$log(1/\delta)$ is the time taken to learn C with error ϵ with probability atleast 1-δ.

5 Show that one can learn any target $f : \{-1, 1\}^n \rightarrow \{-1, 1\}$ with error 0 from random examples only in time $\tilde{\mathcal{O}}(2^n)$.

The intuition behind our algorithm is to maintain a list of labeled examples from the example oracle and keep collecting them till we have 2^n unique examples.Hence, the error is 0 as the label and input is present for the example. And we train our algorithm with these unique set.

Below is the algorithm to learn the C:

a.) Initially the hashtable (to store the labeled examples will be empty)

b.) Extract the labeled example from the Oracle, and check if it already

exists in the hash table. If its present ignore it else, add the example to the hash table as a new entry.
c.) Keep repeating step b. till we have 2^n unique examples.
d.) Now we train our algorithm using the extracted 2^n examples, to learn C with error 0.

To extract each example from the hash table it takes O(1) time. Consider the probability of finding a labeled example out of 2^n examples:
Probability of finding one example is $1/2^n$. If we need to determine n such examples then under Chernoff bounds it can be found in poly(n) time.

Thus, the number of examples to get all the 2^n examples, is $2^n.poly(n) = \tilde{\mathcal{O}}(2^n)$.