9.1 Relations between Lectures

Here is the Big Question: When does learning work? When can we expect to generalized from i.i.d. data samples? (non-i.i.d. is the second half of the course)

- **Start**
  - **Is the problem binary classification?**
    - No: **Next few Lectures**
    - Yes: **PAC Learning**
      - **Is the problem “noise-free” (i.e. ∃ consistent hypothesis)?**
        - No: **Next Lecture**
        - Yes: **Deviation bounds + Union bound**
          - $R(h_s) \leq \log|C|$ with high prob.
          - **Is your class $C$ finite?**
            - No: **Deviation bounds + Union bound + Two simple tricks ↓**
              - $R(h_s) > \epsilon$ with prob. $\pi_C(2m)e^{-o(m\epsilon)}$
              - **Is $VCD(C)$ finite?**
                - No: **Screwed (Next Lecture)**
                - Yes: **Deviation bounds + Union bound**
                  - $R(h_s) \leq \frac{d \log m}{m} + \log$ terms with high prob.
                  - **Plus Sauer’s Lemma (Today)**
9.2 Sauer’s Lemma

Recall $C$ is the set of functions $h : X \rightarrow \{0, 1\}$. Given any $S = \{x_1, \ldots, x_m\} \subseteq X$, we may define the following.

**Definition 9.1.** The concept class $C$ restricted to set $S$ is

$$C|_S := \{(h(x_1), \ldots, h(x_m)) : h \in C\}.$$  

**Definition 9.2.** The growth function for $C$ is

$$\Pi_C(m) = \sup \{|C| : S \subseteq X, |S| = m\}.$$  

**Definition 9.3.** The VC-dimension of $C$ is the size of largest set that can be fully shattered by $C$

$$VCD(C) = \sup \{d : \pi_C(d) = 2^d\}.$$  

Sauer’s Lemma will tell us how fast $\Pi_C(m)$ grows. Consider $X = \mathbb{R}^d$, let $C$ be the class of linear threshold functions (LTFs)

$$C = \{h_{w, b} : w \in \mathbb{R}^d, b \in \mathbb{R}\} \text{ where } h_{w, b}(x) = \text{sign}(w^\top x + b).$$

**Claim 9.4.** $\Pi_C(m) \leq \binom{m}{d+1}2^{d+1}$ when $C$ is the class of LTF in dimension $d$.

**Proof:** (a non-rigorous one) Every labelling of $m$ points in $\mathbb{R}^d$ can be compressed to a labelling of $d + 1$ points in $\mathbb{R}^d$. That is, given labels on certain $d + 1$ points, we can infer the labels of the rest. Therefore, the number of possible labellings is less than or equal to the number of $d + 1$ sized labelled subsets which is equal to $\binom{m}{d+1}2^{d+1}$.  

**Fact 9.5.** $\binom{n}{k} \leq \left(\frac{ne}{k}\right)^k = O(n^k)$.  

From the above claim and fact, it follows that for the class of LTF we have $\Pi_C(m) \leq \binom{m}{d+1}2^{d+1} = \theta(m^{d+1}).$

**Theorem 9.6** (Sauer’s Lemma). Assume $VCD(C) = d$, then

$$\Pi_C(m) \leq \sum_{i=0}^{d} \binom{m}{i} = O(m^d),$$

which says that

(i). $\Pi_C(m) = 2^m$ for $m \leq d$, and

(ii). $\Pi_C(m)$ is polynomial on $m$ for $m > d$.

**Proof:** Take $S \subseteq X, |S| = m, S = \{x_1, \ldots, x_m\}$.

**Algorithm 9.7** (Shifting Algorithm).

```plaintext
1: for $i = 1, \ldots, m$ do
2:  for $j = 1, \ldots, |C|_S$ do
3:    if The entry $(x_i, h_j)$ is 1 and replacing it with 0 doesn’t duplicate a row then
4:      Replace the entry with 0
5:    end if
6:  end for
7: end for
```
Example  Consider the following table:

\[
T = \begin{array}{cccccc}
& x_1 & x_2 & x_3 & x_4 & x_5 \\
h_1 & 0 & 1 & 0 & 1 & 1 \\
h_2 & 1 & 0 & 0 & 1 & 1 \\
h_3 & 1 & 1 & 1 & 0 & 1 \\
h_4 & 0 & 1 & 1 & 0 & 0 \\
h_5 & 0 & 0 & 0 & 1 & 0
\end{array}
\]

After performing the Shifting Algorithm, it will become:

\[
S = \begin{array}{cccccc}
& x_1 & x_2 & x_3 & x_4 & x_5 \\
h_1 & 0 & 1 & 0 & 0 & 0 \\
h_2 & 0 & 0 & 0 & 1 & 1 \\
h_3 & 0 & 0 & 0 & 0 & 1 \\
h_4 & 0 & 0 & 0 & 0 & 0 \\
h_5 & 0 & 0 & 0 & 1 & 0
\end{array}
\]

We now claim the following:

1. All rows are unique.
2. If exist row with 1’s in columns \( Q \), then \( R \) shatters \( Q \).
3. \( VCD(T) \leq VCD(R) \). The subclaim is that after every shift of a column, I haven’t shattered any new subsets (of columns).

To see this, take a subset of columns \( U \rightarrow i \), consider shifting column \( i \) and suddenly a new row \( j \) restricted to \( U \) appears. I claim that all rows equivalent to row \( j \) restricted to \( U \) will be shifted as well. Hence, the number of unique rows on \( U \) can’t increase.

Statement 2 says the number of 1’s in any row is less than or equal to \( d \). Rows are unique (by 1). Hence total number of rows is less than or equal to number of subsets of size \( d \) or fewer elements of \( S \). }