67577 – Intro. to Machine Learning **Fall semester, 2003/4**

Lecture 7: Generalization II

Lecturer: Amnon Shashua *Scribe: Amnon Shashua* [1]

In this lecture will use the measure of VC dimension, which is a combinatorial measure of concept class complexity, to bound the sample size complexity. We first would like to obtain a bound on the growth of $|\Pi_S(C)|$ when the sample size $|S| = m$ is much larger than the VC dimension $VCdim(C) = d$ of the concept class. We will need few more definitions:

Definition 1 (Growth function)

$$\Pi_C(m) = \max\{|\Pi_S(C)| \ : \ |S| = m\}$$

The measure $\Pi_C(m)$ is the maximum number of dichotomies induced by C for samples of size m. As long as $m \leq d$ then $\Pi_C(m) = 2^m$. The question is what happens to the growth pattern of $\Pi_C(m)$ when $m > d$. We will see that the growth becomes polynomial — a fact which is crucial for the learnability of C.

Definition 2 *For any natural numbers m, d we have the following definition:*

$$\begin{aligned} \Phi_d(m) &= \Phi_d(m-1) + \Phi_{d-1}(m-1) \\ \Phi_d(0) &= \Phi_0(m) = 1 \end{aligned}$$

By induction on m, d it is possible to prove the following:

Theorem 1

$$\Phi_d(m) = \sum_{i=0}^{d} \binom{m}{i}$$

Proof: by induction on m, d. For details see Kearns & Vazirani pp. 56.□

For $m \leq d$ we have that $\Phi_d(m) = 2^m$. For $m > d$ we can derive a polynomial upper bound as follows.

$$\left(\frac{d}{m}\right)^d \sum_{i=0}^{d} \binom{m}{i} \leq \sum_{i=0}^{d} \left(\frac{d}{m}\right)^i \binom{m}{i} \leq \sum_{i=0}^{m} \left(\frac{d}{m}\right)^i \binom{m}{i} = (1 + \frac{d}{m})^m \leq e^d$$

From which we obtain:

$$\left(\frac{d}{m}\right)^d \Phi_d(m) \leq e^d.$$

Dividing both sides by $\left(\frac{d}{m}\right)^d$ yields:

$$\Phi_d(m) \leq e^d \left(\frac{m}{d}\right)^d = \left(\frac{em}{d}\right)^d = O(m^d).$$

We need one more result before we are ready to present the main result of this lecture:

[1]

Theorem 2 (Sauer's lemma) *If $VCdim(C) = d$, then for any m, $\Pi_C(m) \leq \Phi_d(m)$.*

Proof: By induction on both d, m. For details see Kearns & Vazirani pp. 55–56.□

Taken together, we have now a fairly interesting characterization on how the combinatorial measure of complexity of the concept class C scales up with the sample size m. When the VC dimension of C is infinite the growth is exponential, i.e., $\Pi_C(m) = 2^m$ for all values of m. On the other hand, when the concept class has a bounded VC dimension $VCdim(C) = d < \infty$ then the growth pattern undergoes a discontinuity from an exponential to a polynomial growth:

$$\Pi_C(m) = \left\{ \begin{array}{ll} 2^m & m \leq d \\ \leq (\frac{em}{d})^d & m > d \end{array} \right\}$$

As a direct result of this observation, when $m >> d$ is much larger than d the entropy becomes much smaller than m. Recall than from an information theoretic perspective, the entropy of a random variable Z with discrete values $z_1, ..., z_n$ with probabilities p_i, $i = 1, ..., n$ is defined as:

$$H(Z) = \sum_{i=0}^{n} p_i \log_2 \frac{1}{p_i},$$

where $I(p_i) = \log_2 \frac{1}{p_i}$ is a measure of "information", i.e., is large when p_i is small (meaning that there is much information in the occurrence of an unlikely event) and vanishes when the event is certain $p_i = 1$. The entropy is therefore the expectation of information. Entropy is maximal for a uniform distribution $H(Z) = \log_2 n$. The entropy in information theory context can be viewed as the number of bits required for coding $z_1, ..., z_n$. In coding theory it can be shown that the entropy of a distribution provides the lower bound on the average length of any possible encoding of a uniquely decodable code fro which one symbol goes into one symbol. When the distribution is uniform we will need the maximal number of bits, i.e., one cannot compress the data. In the case of concept class C with VC dimension d, we see that one when $m \leq d$ all possible dichotomies are realized and thus one will need m bits (as there are 2^m dichotomies) for representing all the outcomes of the sample. However, when $m >> d$ only a small fraction of the 2^m dichotomies can be realized, therefore the distribution of outcomes is highly non-uniform and thus one would need much less bits for coding the outcomes of the sample. The technical results which follow are therefore a formal way of expressing in a rigorous manner this simple truth — *If it is possible to compress, then it is possible to learn.* The crucial point is that learnability is a direct consequence of the "phase transition" (from exponential to polynomial) in the growth of the number of dichotomies realized by the concept class.

7.1 A Polynomial Bound on the Sample Size m for PAC Learning

In this section we will follow the material presented in Kearns & Vazirani pp. 57–61 and prove the following:

Theorem 3 (Double Sampling) *Let C be any concept class of VC dimension d. Let L be any algorithm that when given a set S of m labeled examples $\{\mathbf{x}_i, c(\mathbf{x}_i)\}_i$, sampled i.i.d according to some fixed but unknown distribution D over the instance space X, of some concept $c \in C$, produces as output a concept $h \in C$ that is consistent with S. Then L is a learning algorithm in the formal sense provided that the sample size obeys:*

$$m \geq c_0 \left(\frac{1}{\epsilon} \log \frac{1}{\delta} + \frac{d}{\epsilon} \log \frac{1}{\epsilon} \right)$$

for some constant $c_0 > 0$.

The idea behind the proof is to build an "approximate" concept space which includes a finite number of concepts arranged on a grid such that the distance between the approximate concepts h and the target concept c is at least ϵ — where distance is defined as the weight of the region in X which is in conflict with the target concept. The probability that a random sample S will not hit all the error regions between each approximate concept and the target concept is shown to be bounded by δ provided that the sample complexity m obeys the bound above. Since the learner produces a consistent concept h^* with S and each point of S hits one of the error regions (whose weight is larger than ϵ) then $err(h^*) \leq \epsilon$. To formalize this story we will need few more definitions. Unless specified otherwise, $c \in C$ denotes the target concept and $h \in C$ denotes *some* concept.

Definition 3

$$c\Delta h = h\Delta c = \{\mathbf{x} \ : \ c(\mathbf{x}) \neq h(\mathbf{x})\}$$

$c\Delta h$ is the region in instance space where both concepts do not agree — the error region. The probability that $\mathbf{x} \in c\Delta h$ is equal to (by definition) $err(h)$.

Definition 4

$$\begin{aligned} \Delta(c) &= \{h\Delta c \ : \ h \in C\} \\ \Delta_\epsilon(c) &= \{h\Delta c \ : \ h \in C \ and \ err(h) \geq \epsilon\} \end{aligned}$$

$\Delta(c)$ is a set of error regions, one per concept $h \in C$ over all concepts. The error regions are with respect to the target concept. The set $\Delta_\epsilon(c) \subset \Delta(c)$ is the set of all error regions whose weight exceeds ϵ. Recall that weight is defined as the probability that a point sampled according to D will hit the region.

It will be important for later to evaluate the VC dimension of $\Delta(c)$. Unlike C, we are not looking for the VC dimension of a class of function but the VC dimension of a set of regions in space. Recall the definition of $\Pi_C(S)$ from the previous lecture: there were two equivalent definitions one based on a set of vectors each representing a labeling of the instances of S induced by some concept. The second, yet equivalent, definition is based on a set of subsets of S each induced by some concept (where the concept divides the sample points of S into positive and negative labeled points). So far it was convenient to work with the first definition, but for evaluating the VC dimension of $\Delta(c)$ it will be useful to consider the second definition:

$$\Pi_{\Delta(c)}(S) = \{r \cap S \ : \ r \in \Delta(c)\},$$

that is, the collection of subsets of S induced by intersections with regions of $\Delta(c)$. An intersection between S and a region r is defined as the subset of points from S that fall into r. We can easily show that the VC dimensions of C and $\Delta(c)$ are equal:

Lemma 4

$$VCdim(C) = VCdim(\Delta(c)).$$

Proof: we have that the elements of $\Pi_C(S)$ and $\Pi_{\Delta(c)}(S)$ are susbsets of S, thus we need to show that for every S the cardinality of both sets is equal $|\Pi_C(S)| = |\Pi_{\Delta(c)}(S)|$. To do that it is sufficient to show that for every element $s \in \Pi_C(S)$ there is a unique corresponding element in $\Pi_{\Delta(c)}(S)$. Let $c \cap S$ be the subset of S induced by the target concept c. The set s (a subset of S) is realized by some concept h (those points in S which were labeled positive by h). Therefore, the set $s \cap (c \cap S)$ is the subset of S containing the points that hit the region $h\Delta c$ which is an element of $\Pi_{\Delta(c)}(S)$. Since this is a one-to-one mapping we have that $|\Pi_C(S)| = |\Pi_{\Delta(c)}(S)|$. □

Definition 5 (ϵ-net) *For every $\epsilon > 0$, a sample set S is an ϵ-net for $\Delta(c)$ if every region in $\Delta_\epsilon(c)$ is hit by at least one point of S:*

$$\forall r \in \Delta_\epsilon(c), \quad S \cap r \neq \emptyset.$$

In other words, if S hits all the error regions in $\Delta(c)$ whose weight exceeds ϵ, then S is an ϵ-net. Consider as an example the concept class of intervals on the line $[0, 1]$. A concept is defined by an interval $[\alpha_1, \alpha_2]$ such that all points inside the interval are positive and all those outside are negative. Given $c \in C$ is the target concept and $h \in C$ is some concept, then the error region $h\Delta c$ is the union of two intervals: I_1 consists of all points $x \in h$ which are not in c, and I_2 the interval of all points $x \in c$ but which are not in h. Assume that the distribution D is uniform (just for the sake of this example) then, $prob(x \in I) = |I|$ which is the length of the interval I. As a result, $err(h) > \epsilon$ if either $|I_1| > \epsilon/2$ or $|I_2| > \epsilon/2$. The sample set

$$S = \{x = \frac{k\epsilon}{2} \ : \ k = 0, 1, ..., 2/\epsilon\}$$

contains sample points from 0 to 1 with increments of $\epsilon/2$. Therefore, every interval larger than ϵ must be hit by at least one point from S and by definition S is an ϵ-net.

It is important to note that if S forms an ϵ-net then we are guaranteed that $err(h) \leq \epsilon$. Let $h \in C$ be the consistent hypothesis with S (returned by the learning algorithm L). Becuase h is consistent, $h\Delta c \in \Delta(c)$ has not been hit by S (recall that $h\Delta c$ is the error region with respect to the target concept c, thus if h is consistent then it agrees with c over S and therefore S does not hit $h\Delta c$). Since S forms an ϵ-net for $\Delta(c)$ we must have $h\Delta c \notin \Delta_\epsilon(c)$ (recall that by definition S hits all error regions with weight larger than ϵ). As a result, the error region $h\Delta c$ must have a weight smaller than ϵ which means that $err(h) \leq \epsilon$.

The conclusion is that if we can *bound* the probability that a random sample S *does not* form an ϵ-net for $\Delta(c)$, then we have bounded the probability that a concept h consistent with S has $err(h) > \epsilon$. This is the goal of the proof of the double-sampling theorem which we are about to prove below:

Proof (following Kearns & Vazirani pp. 59–61): Let S_1 be a random sample of size m (sampled i.i.d. according to the unknown distribution D) and let A be the event that S_1 *does not* form an ϵ-net for $\Delta(c)$. From the preceding discussion our goal is to upper bound the probability for A to occur, i.e., $prob(A) \leq \delta$.

If A occurs, i.e., S_1 is not an ϵ-net, then by definition there must be some region $r \in \Delta_\epsilon(c)$ which is not hit by S_1, that is $S_1 \cap r = \emptyset$. Note that $r = h\Delta(c)$ for some concept h which is consistent with S_1. At this point the space of possibilities is infinite, because the probability that we fail to hit $h\Delta(c)$ in m random examples is at most $(1 - \epsilon)^m$. Thus the probability that we fail to hit *some* $h\Delta c \in \Delta_\epsilon(c)$ is bounded from above by $|\Delta(c)|(1 - \epsilon)^m$ — which does not help us due to the fact that $|\Delta(c)|$ is infinite. The idea of the proof is to turn this into a finite space by using another sample, as follows.

Let S_2 be another random sample of size m. We will select m (for both S_1 and S_2) to guarantee a high probability that S_2 will hit r many times. In fact we wish that S_2 will hit r at least $\frac{\epsilon m}{2}$ with probability of at least 0.5:

$$prob(|S_2 \cap r| > \frac{\epsilon m}{2}) = 1 - prob(|S_2 \cap r| \leq \frac{\epsilon m}{2}).$$

We will use the Chernoff bound (multiplicative form) to obtain a bound on the right-hand side term. Recall that if we have m Bernoulli trials (coin tosses) $Z_1, ..., Z_m$ with expectation $E(Z_i) = p$

and we consider the random variable $Z = Z_1 + ... + Z_m$ with expectation $E(Z) = \mu$ (note that $\mu = pm$) then for all $0 < \psi < 1$ we have:

$$prob(Z < (1-\psi)\mu) \leq e^{-\frac{\mu\psi^2}{2}}.$$

Considering the sampling of m examples that form S_2 as Bernoulli trials, we have that $\mu \geq \epsilon m$ (since the probability that an example will hit r is at least ϵ) and $\psi = 0.5$. We obtain therefore:

$$prob(|S_2 \cap r| \leq (1-\frac{1}{2})\epsilon m) \leq e^{-\frac{\epsilon m}{8}} = \frac{1}{2}$$

which happens when $m = \frac{8}{\epsilon}\ln 2 = O(\frac{1}{\epsilon})$. To summarize what we have obtained so far, we have calculated the probability that S_2 will hit r many times *given* that r was fixed using the previous sampling, i.e., given that S_1 does not form an ϵ-net. To formalize this, let B denote the combined event that S_1 does not form an ϵ-event and S_2 hits r at least $\epsilon m/2$ times. Then, we have shown that for $m = O(1/\epsilon)$ we have:

$$prob(B/A) \geq \frac{1}{2}.$$

From this we can calculate $prob(B)$:

$$prob(B) = prob(B/A)prob(A) \geq \frac{1}{2}prob(A),$$

which means that our original goal of bounding $prob(A)$ is equivalent to finding a bound $prob(B) \leq \delta/2$ because $prob(A) \leq 2 \cdot prob(B) \leq \delta$. The crucial point with the new goal is that to analyze the probability of the event B, we need only to consider a finite number of possibilities, namely to consider the regions of $\Pi_{\Delta_\epsilon(c)}(S_1 \cup S_2)$. This is because the occurrence of the event B is equivalent to saying that there is some $r \in \Pi_{\Delta_\epsilon(c)}(S_1 \cup S_2)$ such that $|r| \geq \epsilon m/2$ and $S_1 \cap r = \emptyset$. This is because $\Pi_{\Delta_\epsilon(c)}(S_1 \cup S_2)$ contains all the subsets of $S_1 \cup S_2$ realized as intersections over all regions in $\Delta_\epsilon(c)$. Thus even though we have an infinite number of regions we still have a finite number of subsets. We wish therefore to analyze the following probability:

$$prob\left(r \in \Pi_{\Delta_\epsilon(c)}(S_1 \cup S_2) \ : \ |r| \geq \epsilon m/2 \ and \ S_1 \cap r = \emptyset\right).$$

Let $S = S_1 \cup S_2$ a random sample of $2m$ (note that since the sampling is i.i.d. it is equivalent to sampling S_1 and S_2 separately) and r satisfying $|r| \geq \epsilon m/2$ being *fixed*. Consider some random partitioning of S into S_1 and S_2 and consider then the problem of estimating the probability that $S_1 \cap r = \emptyset$. This problem is equivalent to the following combinatorial question: we have $2m$ balls, each colored Red or Blue, with exaclty $l \geq \epsilon m/2$ Red balls. We divide the $2m$ balls into groups of equal size S_1 and S_2 and we are interested in bounding the probability that all of the l balls fall in S_2 (that is, the probability that $S_1 \cap r = \emptyset$). This in turn is equivalent to first dividing the $2m$ uncolored balls into S_1 and S_2 groups and then randomly choose l of the balls to be colored Red and analyze the probability that all of the Red balls fall into S_2. This probability is exactly

$$\frac{\binom{m}{l}}{\binom{2m}{l}} = \prod_{i=0}^{l-1}\frac{m-i}{2m-i} \leq \prod_{i=0}^{l-1}\frac{1}{2} = \frac{1}{2^l} = 2^{-\epsilon m/2}.$$

This probability was evaluated for a *fixed* S and r. Thus, the probability that this occurs for *some* $r \in \Pi_{\Delta_\epsilon(c)}(S)$ satisfying $|r| \geq \epsilon m/2$ (which is $prob(B)$) can be calculated by summing over all

possible fixed r and applying the union bound $prob(\sum_i Z_i) \leq \sum_i prob(Z_i)$:

$$\begin{aligned} prob(B) &\leq |\Pi_{\Delta_\epsilon(c)}(S)|2^{-\epsilon m/2} \leq |\Pi_{\Delta(c)}(S)|2^{-\epsilon m/2} \\ &= |\Pi_C(S)|2^{-\epsilon m/2} \leq \left(\frac{2\epsilon m}{d}\right)^d 2^{-\epsilon m/2} \leq \frac{\delta}{2}, \end{aligned}$$

from which it follows that:

$$m = O\left(\frac{1}{\epsilon}\log\frac{1}{\delta} + \frac{d}{\epsilon}\log\frac{1}{\epsilon}\right).$$

□

Few comments are worthwhile at this point:

1. It is possible to show that the upper bound on the sample complexity m is tight by showing that the lower bound on m is $\Omega(d/\epsilon)$ (see Kearns & Vazirani pp. 62).

2. The treatment above holds also for the unrealizable case (target concept $c \notin C$) with slight modifications to the bound. In this context, the learning algorithm L must simply minimize the sample (empirical) error $\hat{err}(h)$ defined:

   $$\hat{err}(h) = \frac{1}{m}|\{i : h(\mathbf{x}_i) \neq y_i\}| \quad \mathbf{x}_i \in S.$$

   The generalization of the double-sampling theorem (Derroye'82) states that the empirical errors converge uniformly to the true errors:

   $$prob\left(\max_{h\in C}|\hat{err}(h) - err(h)| \geq \epsilon\right) \leq 4e^{(4\epsilon+4\epsilon^2)}\left(\frac{\epsilon m^2}{d}\right)^d 2^{-m\epsilon^2/2} \leq \delta,$$

   from which it follows that

   $$m = O\left(\frac{1}{\epsilon^2}\log\frac{1}{\delta} + \frac{d}{\epsilon^2}\log\frac{1}{\epsilon}\right).$$

Taken together, we have arrived to a fairly remarkable result. Despite the fact that the distribution D from which the training sample S is drawn from is *unknown* (but is known to be fixed), the learner simply needs to minimize the empirical error. If the sample size m is large enough the learner is guaranteed to have minimized the true errors for some accuracy and confidence parameters which define the sample size complexity. Equivalently,

$$|Opt(C) - \hat{err}(h)| \longrightarrow_{m\to\infty} 0.$$

Not only is the convergence is independent of D but also the rate of convergence is independent (namely, it does not matter where the optimal h^* is located). The latter is very important because without it one could arbitrarily slow down the convergence rate by maliciously choosing D. The beauty of the results above is that D does not have an effect at all — one simply needs to choose the sample size to be large enough for the accuracy, confidence and VC dimension of the concept class to be learned over.

7.2 So Why Does SVM Work?

In lectures 4 and 5 we discussed the large margin principle for finding an optimal separating hyperplane. It is natural to ask how does the theory presented in lectures 6,7 relate to the reason as to why a maximal margin hyperplane is optimal with regard to the formal sense of learning (i.e. to generalization from empirical errors to true errors)? We saw in the previous section that the sample complexity $m(\epsilon, \delta, d)$ depends also on the VC dimension of the concept class — which is $n+1$ for hyperplanes in R^n. Thus, another natural question that may certainly arise is what is the gain in employing the "kernel trick"? For a fixed m, mapping the input instance space X of dimension n to some higher (exponentially higher) feature space might simply mean that we are compromising the accuracy and confidence of the learner (since the VC dimension is equal to the instance space dimension plus 1).

Given a fixed sample size m, the best the learner can do is to minimize the empirical error *and at the same time to try to minimize the VC dimension d of the concept class*. The smaller d is, for a fixed m, the higher the accuracy and confidence of the learning algorithm. Likewise, the smaller d is, for a fixed accuracy and confidence values, the smaller sample size is required.

There are two possible ways to decrease d. First is to decrease the dimension n of the instance space X. This amounts to "feature selection", namely find a subset of coordinates that are the most "relevant" to the learning task. We will not discuss feature selection in this course. A second approach is to maximize the margin. Let the margin associated with the separating hyperplane h (i.e. consistent with the sample S) be γ. Let the input vectors $\mathbf{x} \in X$ have a bounded norm, $|\mathbf{x}| \leq R$. It can be shown that the VC dimension of the concept class C_γ of hyperplanes with margin γ is:

$$C_\gamma = \min\left\{\frac{R^2}{\gamma^2}, n\right\} + 1.$$

Thus, if the margin is very small then the VC dimension remains $n+1$. As the margin gets larger, there comes a point where $R^2/\gamma^2 < n$ and as a result the VC dimension decreases. Moreover, mapping the instance space X to some higher dimension feature space will not change the VC dimension as long as the margin remains the same. It is expected that the margin will not scale down or will not scale down as rapidly as the scaling up of dimension from image space to feature space.

To conclude, maximizing the margin (while minimizing the empirical error) is advantageous as it decreases the VC dimension of the concept class and causes the accuracy and confidence values of the learner to be largely immune to dimension scaling up while employing the kernel trick.