Introduction to Machine Learning (in Natural Language Processing) PFL054

Barbora Hladká, Martin Holub

Charles University in Prague,
Faculty of Mathematics and Physics,
Institute of Formal and Applied Linguistics

December 14, 2011

Lecture 10 :: Probably Approximately Correct learning model

Computational learning theory (CLT)

- is a part of theoretical computer science that formally studies how to design computer programs that are capable of learning, and identifies the computational limits of learning machines (Credits: Kononenko I., Kukar Matja: Machine learning and Data mining, 2007)
- Using statistics, we compare learning algorithms empirically (we measure performance on sample data).
- CLT provides a formal framework to precisely formulate and address questions regarding the performance of different learning algorithms. Are there any general laws that govern machine learners?

Computational learning theory (2)

- Probably Approximately Correct (PAC) learning framework is a part of CLT.
- *Sample complexity* (i.e. data requirements) How many training examples are needed for a learner to converge with high probability to a successful hypothesis?
- *Computational complexity* How much computational effort is needed for a learner to converge with high probability to a successful hypothesis?

The problem setting

- Input data X.
- Output values $Y = \{0, 1\}$.
- Training data $Data = \{\langle \mathbf{x}_i, c(\mathbf{x}_i) = y_i \rangle, \mathbf{x}_i \in X, y_i \in Y\}_{i=1}^{m}$.
- C set of target concepts $c \in C : c : X \rightarrow \{-1, +1\}$
- Instances are generated at random from X according to some probability distribution $\mathcal{D}$. In general, $\mathcal{D}$ may be any distribution and it will be unknown to the learner. $\mathcal{D}$ must be stationary, i.e. it does not change over time.
- A set H of possible hypotheses.
- A learner L outputs some hypothesis h from H as a model of c.
- What are the capabilities of learning algorithms? We will not concentrate on individual learning algorithms, but rather on broad classes of them.

Error of a hypothesis

How closely the learner's output hypothesis h approximates the target concept c?

Definition

True error $error_{\mathcal{D}}(h)$ of the hypothesis h with respect to the target function c and the probabilistic distribution $\mathcal{D}$ is the probability that the hypothesis h wrongly classifies a randomly selected instance according to $\mathcal{D}$ ($error_{\mathcal{D}}(h) \equiv \Pr_{x \in \mathcal{D}}[c(x) \neq h(x)]$)

(*You are already familiar with this definition.*)

PAC learnability

to characterize classes of target concepts that can be reliably learned from a reasonable number of randomly drawn training examples and a reasonable amount of computation.

Definition Consider a concept class C defined over a set of instances X of length n (n is the size of instances, i.e. the size of their representation) and a learner L using hypothesis space H. C is **PAC-learnable** by L using H if for all $c \in C$, distributions $\mathcal{D}$ over X, ϵ such that $0 < \epsilon < \frac{1}{2}$, δ such that $0 < \delta < \frac{1}{2}$, learner L will with probability at least $1 - \delta$ (confidence) output a hypothesis $h \in H$ such that $error_{\mathcal{D}}(h) \leq \epsilon$, in time that is polynomial in $\frac{1}{\epsilon}$, $\frac{1}{\delta}$, n, and $size(c)$ ($size(c)$ is the encoding length of $c \in C$, assuming some representation for C).

PAC learnability (2)

I.e., two things are required from L:

1. L must output, with arbitrarily high probability $1 - \delta$, a hypothesis having arbitrarily low error ϵ.
2. It must do efficiently in time that grows at most polynomially with $\frac{1}{\epsilon}, \frac{1}{\delta}$, with n and $size(c)$ (that define inherent complexity of the underlying instance space X and concept class C).

I.e. to show that some class C of target concepts is PAC learnable, we have to

1. show that each $c \in C$ can be learned from a polynomial numebr of trainng examples.
2. show that the processing time per example is polynomially bounded.

Sample complexity

How many training examples are needed for a learner to converge (with high probability) to a successful hypothesis? We will express it in terms of size of the hypothesis space H and so-called **Vapnik-Chervonenkis dimension**.

Sample complexity for FINITE hypothesis spaces

Can we derive a bound on the number of training examples required by *any consistent* learner? Answer is *yes*. Why?

Recall the definition of *Version Space*: **Version Space** ($VS_{H,Data}$) with respect to H and training data $Data$ is the subset of H consistent with the training examples in $Data$, i.e. $VS_{H,Data} \equiv \{h \in H | Consistent(h, Data)\}$.

Sample complexity for FINITE hypothesis spaces (2)

To bound the number of examples needed by any consistent learner, we need only bound the number of examples needed to assure that the Version Space contains no unacceptable hypotheses. The following definition states this condition precisely:

Definition Consider a hypothesis space H, target concept c, instance distribution $\mathcal{D}$, and set of training examples *Data* of c. The version space $VS_{H,Data}$ is said to be ϵ-**exhausted** with respect to c and $\mathcal{D}$, if every hypothesis h in $VS_{H,Data}$ has true error less than ϵ with respect to c and $\mathcal{D}$: $(\forall h \in VS_{H,Data}) error_{\mathcal{D}}(h) < \epsilon$.

Sample complexity for FINITE hypothesis spaces (3)

So we bound the number of training examples needed to be sure that the version space contains no hypotheses that does not match the training examples. The following theorem provides such a bound:

Theorem ϵ-exhausting version space

If the hypothesis space H is finite, and *Data* is a sequence of $m \geq 1$ independent randomly drawn examples of some target concept c, than for any $0 \leq \epsilon \leq 1$, the probability that the version space $VS_{H,Data}$ is not ϵ-exhausted (with respect to c) is less than or equal to $|H|e^{-m\epsilon}$.

Theorem ϵ-exhausting version space :: Proof

Let $h_1, h_2, ..., h_k$ be all the hypotheses in H that have true error greater than ϵ with respect to c. We fail to ϵ-exhaust the Version Space if and only if at least one of these k hypotheses happens to be consistent with all m independent random training examples. The probability that any single hypothesis having true error greater than ϵ would be consistent with one randomly drawn examples is at most $(1-\epsilon)$. Therefore the probability that this hypothesis will be consistent with m independently drawn examples is at most $(1-\epsilon)^m$. Given that we have k hypotheses with error greater than ϵ, the probability that at least one of these will be consistent with all m training examples is at most $k(1-\epsilon)^m$. Since $k \leq |H|$, this is at most $|H|(1-\epsilon)^m$. Finally, we use a general inequality stating that if $0 \leq \epsilon \leq 1$ then $(1-\epsilon) \leq e^{-\epsilon}$. Thus, $k(1-\epsilon)^m \leq |H|(1-\epsilon)^m \leq |H|e^{-m\epsilon}$ which proves the theorem.

In other words, this bounds the probability that m training examples will fail to eliminate all ”bad” hypotheses for any consistent learner using hypothesis space H.

We use this result to determine the number of training examples required to reduce this probability of failure below some desired level δ:

$$|H|e^{-\epsilon m} \leq \delta \rightarrow m \geq \frac{1}{\epsilon}(\ln|H| + \ln(\frac{1}{\delta})) \tag{1}$$

The given inequality provides a general bound on the number m of training examples sufficient to assure that any consistent hypothesis will be probably (with probability $(1-\delta)$) approximately (within error ϵ) correct. m grows linearly in $\frac{1}{\epsilon}$ and logarithmically in $\frac{1}{\delta}$.

Agnostic learning and inconsistent hypotheses

If H does not contain the target concept c, then a zero-training-error hypothesis cannot always be found. We ask to output hypothesis with the minimum error over the training examples.

Agnostic learner

makes no prior commitment about whether or not $C \subset H$. The equation $m \geq \frac{1}{\epsilon}(\ln |H| + \ln(\frac{1}{\delta}))$ is based on the assumption of zero-training-error hypothesis. Let's generalize it for nonzero training error hypothesis: $error_{Data}(h)$, let $h_{best} = argmin_{h \in H} error_{Data}(h)$.

Agnostic learner (2)

How many training examples suffice to ensure (with high probability) that its true error $error_{\mathcal{D}}(h)$ will be no more than $\epsilon + error_{Data}(h_{best})$? (in the previous case $error_{Data}(h_{best}) = 0$).

The Hoeffding bounds state if $error_{Data}(h)$ is measured over the set *Data* containing m randomly drawn examples, then

$$\Pr[error_{\mathcal{D}}(h) > error_{Data}(h) + \epsilon] \leq e^{-2m\epsilon^2}. \tag{2}$$

It gives us a bound on the probability that an arbitrary chosen single hypothesis has a misleading training error.

To assure that the **best** hypothesis found by L has an error bounded in this way, we must consider that any $h \in H$ could have a large error

$$\Pr[(\exists h \in H)(error_{\mathcal{D}}(h) > error_{Data}(h) + \epsilon)] \leq |H|e^{-2m\epsilon^2}. \tag{3}$$

If we call $\delta = \Pr[(\exists \in H)(error_{\mathcal{D}}(h) > error_{Data}(h) + \epsilon)]$, then

$$m \geq \frac{1}{2\epsilon^2}(ln|H| + ln(\frac{1}{\delta})). \tag{4}$$

In this less restrictive case m grows as the square of $\frac{1}{\epsilon}$, rather than linearly with $\frac{1}{\epsilon}$.

Conjunctions of Boolean literals are PAC learnable

Consider the class C of target concepts described by conjunction of up to n literals. A literal is either a Boolean variable or its negation, i.e. either $l_i = a_i$ or $l_i = \neg(a_i)$ and $Values(a_i) \in \{+1, -1\}$. For example, $c = l_1 \& l_2 \& l_4 \& ... \& l_n$ (l_3 is missing). Is C PAC-learnable?

To answer *yes*,

- we have to show that any consistent learner will require only a polynomial number of training examples to learn any c in C.
- Then suggest a specific algorithm that uses polynomial time per training example.

Consider any consistent learner L using a hypothesis space H identical to C. We need only determine the size $|H|$.

Consider H defined by conjunctions of literals based on n boolean variables. Then $|H| = 3^n$ (include the variable as a literal in the hypothesis, include its negation as a literal, or ignore it).

Example

$n = 2$

$h_1 = a_1$
$h_2 = \neg a_1$
$h_3 = a_2$
$h_4 = \neg a_2$
$h_5 = a_1 \wedge a_2$
$h_6 = a_1 \wedge \neg a_2$
$h_7 = \neg a_1 \wedge a_2$
$h_8 = \neg a_1 \wedge \neg a_2$
$h_9 = a_1 \wedge \neg a_1 \wedge a_2 \wedge \neg a_2$

So

$$m \geq \frac{1}{\epsilon}(n \ln 3 + \ln \frac{1}{\delta}). \tag{5}$$

For example, if a consistent learner attempts to learn a target concept described by conjunctions of up to 10 literals, and we desire 95% probability that it will learn a hypothesis with error less than 0.1, then it suffices to present m randomly drawn training examples, where $m = \frac{1}{0.1}(10 \ln 3 + ln(\frac{1}{0.05}) = 140$.

Recall FIND-S algorithm.

What is the FIND-S algorithm doing? For each new positive example, the algorithm computes the intersection of the literals shared by the current hypothesis and the new training example, i.e

For a positive example $\mathbf{x} = \langle x_1, x_2, ..., x_n \rangle$, removes literals from h to make it consistent with $\mathbf{x}$.

The most specific hypothesis: $a_1 \wedge \neg a_1 \wedge a_2 \wedge \neg a_2 \wedge ... \wedge a_n \wedge \neg a_n$.

Theorem on PAC-learnability of boolean conjunctions

The class C of conjunctions of boolean literals is PAC-learnable by the FIND-S algorithm using $H = C$.

k-term DNF is not PAC learnable (2)

$|H| \leq 3^{nk}$ (k terms, each of which may take on 3^n possible values). However, 3^{nk} is an overestimate of H, because it is double-counting the cases where $T_i = T_j$ and where T_i is more general than T_j. We can write

$$m \geq \frac{1}{\epsilon}(nk \ln 3 + \ln(\frac{1}{\delta})). \tag{6}$$

It indicates that the sample complexity of k-term DNF is polynomial in $\frac{1}{\epsilon}, \frac{1}{\delta}, n, k$. BUT ... can be shown that the computational complexity is not polynomial since this problem is equivalent to other problems that are known to be unsolvable in polynomial time.

Sample complexity for INFINITE hypothesis space

We can state bounds on sample complexity that use Vapnik-Chervonenkis dimension of H rather than $|H|$. Even more, this bounds allow us to charachterize the sample complexity of many infinite hypothesis spaces.

Shattering a set of instances

Definition: A **dichotomy** of a set S is a partition of S into two disjoint subsets.

Let's assume a sample set $S \subset X$. Each hypothesis $h \in H$ imposes some dichotomy on S, i.e. h partitions S into two subsets $\{x \in S; h(x) = 1\}$ and $\{x \in S; h(x) = 0\}$.

Shattering a set of instances (2)

Definition: A set of instances S is **shattered** by hypothesis space H if and only if for every dichotomy of S there exists some hypothesis in H with this dichotomy.

What if H cannot shatter X, but can shatter some large subset S of X? Intuitively, it is reasonable to say that the larger the subset of X that can be shattered, the more expressive H. The Vapnik-Chervonenkis Dimension of H is precisely the measure of expressivity:

Definition:The Vapnik-Chervonenkis dimension, $VC(H)$, of hypothesis space H defined over instance space X is the size of the largest finite subset of X shattered by H. If arbitrarily large finite sets of X can be shattered by H, then $VC(H) \equiv \infty$.

Shattering a set of instances (2) :: Note

For any finite $|H|, VC(H) \leq \log_2 |H|$.

To see this, suppose $VC(H) = d$. Then for any finite H will require For any finite 2^d distinct hypotheses to shatter For any finite d instances. For any finite $|H| \leq 2^d$.

Shattering a set of instances :: Examples

1. Consider $X = \mathcal{R}$ and H the set of real intervals $a < x < b$. What is $VC(H)$?
   We must find the largest subset of X that can be shattered by H. Consider $S = \{3.1, 5.7\}$. Can S be shattered by H? For example four hypotheses will do
   $1 < x < 2, 1 < x < 4, 4 < x < 7, 1 < x < 7$. So we know that $VC(H) \geq 2$. $VC(H) \geq 3$???
   Consider $S = \{x_1, x_2, x_3\}$, without loss of generality assume $x_1 < x_2 < x_3$. Clearly, this set cannot be shattered, because the dichotomy that includes x_1 and x_3 and not x_2 cannot be represented by a single closed interval. So $VC(H) = 2$.

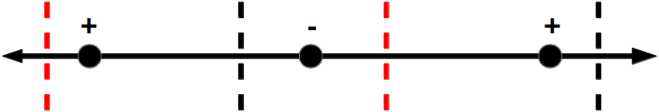

Figure 1: Shattering instances (2)

2. Each instance in X is described by the conjunction of exactly three boolean literals and each hypothesis in H is described by the conjunction of up to three boolean literals. What is $VC(H)$?
   Represent each instance by a 3-bit string of values of the literals l_1, l_2, l_3. Consider three instances: $i_1 : 100, i_2 : 010, i_3 : 001$. This set can be shattered by H, because a hypothesis can be constructed for any desired dichotomy as follows: if dichotomy is to exclude the instance i_j, add the literal $\neg l_j$ to the hypothesis. For example, include i_2 and exclude $i_1, i_3 \rightarrow$ use the hypothesis $\neg l_1 \wedge \neg l_3$. This can be extended from three features to n. Thus, the VC dimension for conjunctions of n boolean variables is at least n.
3. What is the VC-dimension of axis parallel rectangles in the plane $X = \mathcal{R}^2$? The target function is specified by a rectangle, and labels any example positive iff it lies inside that rectangle.

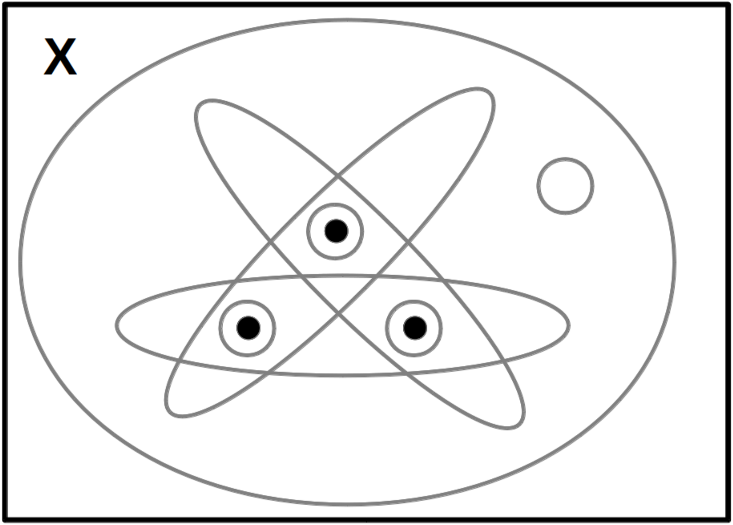

Figure 2: Shattering instances (1)

Sample complexity and the VC dimension

Recall the question How many randomly drawn training examples suffice to probably approximately correct learn any target concept in C?

Let's derive the analogous answer to the earlier bound of m (recall $VC(H) \leq \log_2 |H|$):

$$m \geq \frac{1}{\epsilon}(4\log_2(\frac{2}{\delta} + 8VC(H)\log_2(\frac{13}{\epsilon}). \tag{7}$$

Theorem: Low bound on sample complexity

Consider any concept class C such that $VC(C) \geq 2$, any learner L, and any $0 < \epsilon < \frac{1}{8}$, and $0 < \delta < \frac{1}{100}$. Then there exists a distribution $\mathcal{R}$ and target concept in C such that if L observes fewer examples than

$$max[\frac{1}{\epsilon} \log(\frac{1}{\delta}), \frac{(VC(C) - 1)}{32\epsilon}] \tag{8}$$

then with probability at least δ, L outputs a hypothesis h having error $error_{\mathcal{D}}(h) > \epsilon$.