Learning Theory

Nicholas Ruozzi
University of Texas at Dallas

Based on the slides of Vibhav Gogate and David Sontag
Learning Theory

• So far, we’ve been focused only on algorithms for finding the best hypothesis in the hypothesis space
  • How do we know that the learned hypothesis will perform well on the test set?
  • How many samples do we need to make sure that we learn a good hypothesis?
  • In what situations is learning possible?
If the training data is linearly separable, we saw that perceptron/SVMs will always perfectly classify the training data.

- This does not mean that it will perfectly classify the test data.
- Intuitively, if the true distribution of samples is linearly separable, then seeing more data should help us do better.
Problem Complexity

- Complexity of a learning problem depends on
  - Size/expressiveness of the hypothesis space
  - Accuracy to which a target concept must be approximated
  - Probability with which the learner must produce a successful hypothesis
  - Manner in which training examples are presented, e.g. randomly or by query to an oracle
Problem Complexity

• Measures of complexity
  • Sample complexity
    • How much data you need in order to (with high probability) learn a good hypothesis
  • Computational complexity
    • Amount of time and space required to accurately solve (with high probability) the learning problem
    • Higher sample complexity means higher computational complexity
PAC Learning

• Probably approximately correct (PAC)

• Developed by Leslie Valiant

• The only reasonable expectation of a learner is that with high probability it learns a close approximation to the target concept

• Specify two small parameters, $\epsilon$ and $\delta$, and require that with probability at least $(1 - \delta)$ a system learn a concept with error at most $\epsilon$
Consistent Learners

• Imagine a simple setting
  • The hypothesis space is finite (i.e., $|H| = c$)
  • The true distribution of the data is $p(\tilde{x})$, no noisy labels
  • We learned a perfect classifier on the training set, let’s call it $h \in H$
    • A learner is said to be consistent if it always outputs a perfect classifier (assuming that one exists)
  • Want to compute the (expected) error of the classifier
Notions of Error

• Training error of \( h \in H \)
  • The error on the training data
  • Number of samples incorrectly classified divided by the total number of samples

• True error of \( h \in H \)
  • The error over all possible future random samples
  • Probability that \( h \) misclassifies a random data point
    \[ p(h(x) \neq y) \]
• Let \((x^{(1)}, y^{(1)}), \ldots, (x^{(M)}, y^{(M)})\) be \(M\) labelled data points sampled independently according to \(p\)

• Let \(C_m^h\) be a random variable that indicates whether or not the \(m^{th}\) data point is correctly classified

• The probability that \(h\) misclassifies the \(m^{th}\) data point is

\[
p(C_m^h = 0) = \sum_{(x, y)} p(x, y) 1_{h(x) \neq y} = \epsilon_h
\]
Learning Theory

- Let \((x^{(1)}, y^{(1)}), \ldots, (x^{(M)}, y^{(M)})\) be \(M\) labelled data points sampled independently according to \(p\)

- Let \(C^h_m\) be a random variable that indicates whether or not the \(m^{th}\) data point is correctly classified

- The probability that \(h\) misclassifies the \(m^{th}\) data point is

\[
p(C^h_m = 0) = \sum_{(x,y)} p(x, y) \, 1_{h(x) \neq y} = \epsilon_h
\]

- Probability that sampled pair \((x,y)\) is incorrectly classified by \(h\)
Learning Theory

- Let \((x^{(1)}, y^{(1)}), \ldots, (x^{(M)}, y^{(M)})\) be \(M\) labelled data points sampled independently according to \(p\).

- Let \(C^h_m\) be a random variable that indicates whether or not the \(m^{th}\) data point is correctly classified.

- The probability that \(h\) misclassifies the \(m^{th}\) data point is

\[
p(C^h_m = 0) = \sum_{(x,y)} p(x, y) 1_{h(x) \neq y} = \epsilon_h
\]

This is the true error of \(h\).
• Probability that all data points classified correctly?

• Probability that a hypothesis $h \in H$ whose true error is at least $\epsilon$ correctly classifies the $m$ data points is then
Learning Theory

• Probability that all data points classified correctly?

\[ p(C_1^h = 1, \ldots, C_M^h = 1) = \prod_{m=1}^{M} p(C_m^h = 1) = (1 - \epsilon_h)^M \]

• Probability that a hypothesis \( h \in H \) whose true error is at least \( \epsilon \) correctly classifies the \( m \) data points is then
• Probability that all data points classified correctly?

\[ p(C_1^h = 1, \ldots, C_M^h = 1) = \prod_{m=1}^{M} p(C_m^h = 1) = (1 - \epsilon_h)^M \]

• Probability that a hypothesis \( h \in H \) whose true error is at least \( \epsilon \) correctly classifies the \( m \) data points is then

\[ p(C_1^h = 1, \ldots, C_M^h = 1) \leq (1 - \epsilon)^M \leq e^{-\epsilon^M} \]

for \( \epsilon \leq 1 \)
Learning Theory

• The version space (set of consistent hypotheses) is said to be $\epsilon$-exhausted if and only if every consistent hypothesis has true error less than $\epsilon$

  • Want enough samples to guarantee that every consistent hypothesis has error at most $\epsilon$

• We’ll show that, given enough samples, w.h.p. every hypothesis with true error at least $\epsilon$ is not consistent with the data
The Union Bound

• Let $H_{BAD} \subseteq H$ be the set of all hypotheses that have true error at least $\epsilon$

• From before for each $h \in H_{BAD}$,

\[ p(h \text{ correctly classifies all } M \text{ data points}) \leq e^{-\epsilon M} \]

• So, the probability that some $h \in H_{BAD}$ correctly classifies all of the data points is

\[
p\left( \bigvee_{h \in H_{BAD}} (C^h_{1} = 1, \ldots, C^h_{M} = 1) \right) \leq \sum_{h \in H_{BAD}} p(C^h_{1} = 1, \ldots, C^h_{M} = 1) \\
\leq |H_{BAD}| e^{-\epsilon M} \\
\leq |H| e^{-\epsilon M}
\]
• What we just proved:

• **Theorem:** For a finite hypothesis space, $H$, with $M$ i.i.d. samples, and $0 < \epsilon < 1$, the probability that the version space is not $\epsilon$-exhausted is at most $|H|e^{-\epsilon M}$

• We can turn this into a sample complexity bound
Sample Complexity

• Let $\delta$ be an upper bound on the desired probability of not $\epsilon$-exhausting the sample space

• That is, the probability that the version space is not $\epsilon$-exhausted is at most $|H|e^{-\epsilon M} \leq \delta$

• Solving for $M$ yields

$$M \geq -\frac{1}{\epsilon} \ln \frac{\delta}{|H|}$$

$$= \left( \ln |H| + \ln \frac{1}{\delta} \right) / \epsilon$$
Sample Complexity

• Let $\delta$ be an upper bound on the desired probability of not $\epsilon$-exhausting the sample space

  • That is, the probability that the version space is not $\epsilon$-exhausted is at most $|H|e^{-\epsilon M} \leq \delta$

• Solving for $M$ yields

  $$M \geq -\frac{1}{\epsilon \ln |H|} \ln \frac{\delta}{\epsilon} = \left( \ln |H| + \ln \frac{1}{\delta} \right)/\epsilon$$

This is sufficient, but not necessary (union bound is quite loose)
Decision Trees

• Suppose that we want to learn an arbitrary Boolean function given $n$ Boolean features

• Hypothesis space consists of all decision trees
  • Size of this space = ?

• How many samples are sufficient?
Decision Trees

- Suppose that we want to learn an arbitrary Boolean function given \( n \) Boolean features

- Hypothesis space consists of all decision trees
  - Size of this space = \( 2^{2^n} \) = number of Boolean functions on \( n \) inputs

- How many samples are sufficient?

\[
M \geq \left( \ln 2^{2^n} + \ln \frac{1}{\delta} \right) / \epsilon
\]
Generalizations

• How do we handle the case that there is no perfect classifier?
  • Pick the hypothesis with the lowest error on the training set
• What do we do if the hypothesis space isn’t finite?
  • Infinite sample complexity?
• Next time...
Chernoff Bounds

- Chernoff bound: Suppose \( Y_1, \ldots, Y_M \) are i.i.d. random variables taking values in \( \{0, 1\} \) such that \( E_p[Y_i] = y \). For \( \epsilon > 0 \),

\[
p \left( \left| y - \frac{1}{M} \sum_m Y_m \right| \geq \epsilon \right) \leq 2e^{-2M\epsilon^2}
\]
Chernoff Bounds

- Chernoff bound: Suppose $Y_1, ..., Y_M$ are i.i.d. random variables taking values in $\{0, 1\}$ such that $E_p[Y_i] = y$. For $\epsilon > 0$,

$$p \left( \left| y - \frac{1}{M} \sum_{m} Y_m \right| \geq \epsilon \right) \leq 2e^{-2M\epsilon^2}$$

- Applying this to $1 - C_1^h, ..., 1 - C_M^h$ gives

$$p \left( \left| \epsilon_h - \frac{1}{M} \sum_{m} (1 - C_m^h) \right| \geq \epsilon \right) \leq 2e^{-2M\epsilon^2}$$
Chernoff Bounds

- Chernoff bound: Suppose $Y_1, \ldots, Y_M$ are i.i.d. random variables taking values in $\{0, 1\}$ such that $E_p [Y_i] = y$. For $\epsilon > 0$,

$$p \left( \left| y - \frac{1}{M} \sum_{m} Y_m \right| \geq \epsilon \right) \leq 2e^{-2M\epsilon^2}$$

- Applying this to $1 - C^h_1, \ldots, 1 - C^h_M$ gives

$$p \left( \epsilon_h - \frac{1}{M} \sum_{m} (1 - C^h_m) \geq \epsilon \right) \leq e^{-2M\epsilon^2}$$

This is the training error
PAC Bounds

- **Theorem:** For a finite hypothesis space $H$ finite, $M$ i.i.d. samples, and $0 < \epsilon < 1$, the probability that true error of any of the best classifiers (i.e., lowest training error) is larger than its training error plus $\epsilon$ is at most $|H|e^{-2M\epsilon^2}$

- Sample complexity (for desired $\delta \geq 2|H|e^{-2M\epsilon^2}$)

$$M \geq \left( \ln|H| + \ln\frac{1}{\delta} \right)/2\epsilon^2$$
PAC Bounds

• If we require that the previous error is bounded above by $\delta$, then with probability $(1 - \delta)$, for all $h \in H$

$$\varepsilon_h \leq \varepsilon_h^{train} + \sqrt{\frac{1}{2M} \left( \ln |H| + \ln \frac{1}{\delta} \right)}$$

“bias”

“variance”

• For small $|H|$:
  • High bias (may not be enough hypotheses to choose from)
  • Low variance
PAC Bounds

• If we require that the previous error is bounded above by $\delta$, then with probability $(1 - \delta)$, for all $h \in H$

\[
\epsilon_h \leq \epsilon_h^{\text{train}} + \sqrt{\frac{1}{2M} \left( \ln |H| + \ln \frac{1}{\delta} \right)}
\]

“bias”

“variance”

• For large $|H|$
  • Low bias (lots of good hypotheses)
  • High variance