Learning Theory

COMP 135 Intro to Machine Learning
Liping Liu

Questions

• Do we have guarantee that we can learning good classifiers?
  - Yes with a few assumptions

• What factors affect classification error?

• More insightful understanding of bias/variance?

Setup

• Hypothesis space (model space): all possible classifiers

• Training set/training error
• Expected classification error
• If we minimize training error, can we say anything about expected classification error?

Training process is a process of excluding ‘bad’ classifiers

A game: classification problem

• Predict whether a person likes “yellow” color or not

Example: training set

• Training set: randomly sampled from the population

A game: classifiers

\[ H = \epsilon_0 + \epsilon_1 + \epsilon_2 \]

Error rate
A game: classifiers

- I'm one classifier, and my rule is:
  - If the first letter of one's name is before "O" predict "YES"
  - Otherwise, predict "NO"

Example: expected training error

- Training: keep a classifier that have ZERO training error
  (a simple theoretical analysis here, don't worry overfitting)

- Is the trained classifier good (error rate < \( \epsilon \), say 0.05)?

A game: probability of a good result

- What's the possibility that AT LEAST ONE bad classifier (say Peter or I) can cheat all training examples?
  - The probability at least one bad classifier of making all instances correct is \( \leq |H| \cdot (1 - \epsilon)^N \), assuming we have \( N \) training instances (by union bound)

The first formal result

With probability at least \( 1 - |H| \cdot (1 - \epsilon)^N \), the training process (of picking a classifier with zero training error) will return a classifier with an error rate less than \( \epsilon \) given the training set with \( N \) training instances.
**PAC Learning**

• Probably Approximately-Correct (PAC) learning

**DEFINITION:** An algorithm $A$ is an $(\varepsilon, \delta)$-PAC learning algorithm if, given samples from a distribution, the probability that it returns a “bad function” is at most $\delta$, where a “bad” function is one with test error rate more than $\varepsilon$ on the distribution.

Our result: With probability at least $1 - \delta = 1 - H \cdot \frac{1}{\varepsilon}$, the training process (of picking a classifier with zero training error) will return a classifier with an error rate less than $\varepsilon$, given the training set with $N$ training instances.

**Infinite hypothesis space**

• What is the hypothesis space is infinite?
  - SVM, logistic regression, trees, ...

• VC-dimension to characterize the complexity of the classifier
  - Higher VC-dim $\Rightarrow$ larger model space
  - Lower VC-dim $\Rightarrow$ smaller model space

**Second formal result**

With probability $1 - \delta$, the learning algorithm can return a classifier with error at most $\varepsilon$, given a training set with $N \geq \frac{1}{\varepsilon} \log_2 \frac{1}{\delta} + 8V(C)\log \frac{1}{\delta}$ instances

**VC-dimension**

• Definition:
  For data drawn from some space $X$, the VC dimension of a hypothesis space $H$ over $X$ is the maximal $K$ such that: there exists a set $S \subseteq X$ of size $|S| = K$, such that for any binary labeling of $S$, there exists a function $f \in H$ that matches this labeling.

• Example:
  - linear classifier for $R^2$, VC-dim = 3
  - linear classifier for $R^d$, VC-dim = $d+1$

**Theory on agnostic learning**

• Does not assume zero training error
• Bounding the difference between training error and expected classification error
• More realistic

(out of the scope of this class)

**A summary of learning theory**

• Relation among $N$, $H$, and $(\varepsilon, \delta)$
  - Assuming $H$ always containing the true hypothesis
  $N \gtrsim (\varepsilon, \delta)

The theoretical result we just talked is about the relation among the four values: $VC(H)$, $N$, $\delta$, $\varepsilon$. We bound $\delta$ or $\varepsilon$ with other three fixed.