Generalization, Overfitting, Sample Complexity.

Maria-Florina (Nina) Balcan
February 26th, 2018

• Recommended reading: Mitchell: Ch. 7
  • Suggested exercises: 7.1, 7.2, 7.7
Supervised Classification

Decide which emails are spam and which are important.

Goal: use emails seen so far to produce good prediction rule for future data.
### Example: Supervised Classification

Represent each message by features. (e.g., keywords, spelling, etc.)

<table>
<thead>
<tr>
<th>“money”</th>
<th>“pills”</th>
<th>“Mr.”</th>
<th>bad spelling</th>
<th>known-sender</th>
<th>spam?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>

*Example*

Predict SPAM if unknown AND (money OR pills)

Predict SPAM if $2 \times \text{money} + 3 \times \text{pills} - 5 \times \text{known} > 0$

Reasonable RULES:

Linearly separable
Two Core Aspects of Machine Learning

Algorithm Design. How to optimize?  Computation

Automatically generate rules that do well on observed data.

- E.g.: logistic regression, SVM, Adaboost, etc.

Confidence Bounds, Generalization  (Labeled) Data

Confidence for rule effectiveness on future data.

- Very well understood: Occam’s bound, VC theory, etc.
- Note: to talk about these we need a precise model.
PAC/SLT models for Supervised Learning

Learning Algorithm

Data Source

Distribution \( D \) on \( X \)

Labeled Examples

Expert / Oracle

\( (x_1, \ldots, x_m) \)

\( (x_1, c^*(x_1)), \ldots, (x_m, c^*(x_m)) \)

\( h : X \rightarrow Y \)

\( c^* : X \rightarrow Y \)

\( x_1 > 5 \)

\( x_2 > 2 \)

\( +1 \)

\( -1 \)

\( + \)

\( + \)

\( + \)

\( + \)

\( - \)

\( - \)
PAC/SLT models for Supervised Learning

- **Learning Algorithm**
- **Expert/Oracle**
- **Data Source**
- **Distribution D on X**
- **Labeled Examples**

- Algorithm sees training sample $S: (x_1, c^*(x_1)), ..., (x_m, c^*(x_m))$, $x_i$ independently and identically distributed (i.i.d.) from $D$; labeled by $c^*$

- Does optimization over $S$, finds hypothesis $h$ (e.g., a decision tree).

- Goal: $h$ has small error over $D$. 

Today: $Y = \{-1, 1\}$
PAC/SLT models for Supervised Learning

- **X** - feature or instance space; distribution $D$ over $X$
  
  e.g., $X = \mathbb{R}^d$ or $X = \{0,1\}^d$

- Algo sees training sample $S$: $(x_1,c^*(x_1)), \ldots, (x_m,c^*(x_m))$, $x_i$ i.i.d. from $D$
  
  - labeled examples - assumed to be drawn i.i.d. from some distr. $D$ over $X$ and labeled by some target concept $c^*$
  - labels $\in \{-1,1\}$ - binary classification

- Algo does optimization over $S$, find hypothesis $h$.

- **Goal:** $h$ has small error over $D$.

  \[
  err_D(h) = \Pr_{x \sim D} (h(x) \neq c^*(x))
  \]

**Need a bias: no free lunch.**
PAC/SLT models for Supervised Learning

- **X** - feature or instance space; distribution **D** over **X**
  
e.g., \( X = \mathbb{R}^d \) or \( X = \{0,1\}^d \)

- Algo sees training sample **S**: \((x_1,c^*(x_1)), \ldots, (x_m,c^*(x_m))\), \(x_i\) i.i.d. from **D**
  
  - labeled examples - assumed to be drawn i.i.d. from some distrib. **D** over **X** and labeled by some target concept **c**
  
  - labels \( \in \{-1,1\} \) - binary classification

- Algo does optimization over **S**, find hypothesis **h**.

- **Goal**: **h** has small error over **D**.
  
  \[
  err_D(h) = \Pr_{x \sim D}(h(x) \neq c^*(x))
  \]

  - **Bias**: Fix hypotheses space **H**.
    (whose complexity is not too large).

  - **Realizable**: \( c^* \in H \).
  
  - **Agnostic**: \( c^* \) “close to” **H**.
PAC/SLT models for Supervised Learning

- Algo sees training sample $S: (x_1, c^*(x_1)), \ldots, (x_m, c^*(x_m))$, $x_i$ i.i.d. from $D$
- Does optimization over $S$, find hypothesis $h \in H$.
- **Goal:** $h$ has small error over $D$.
  
  **True error:** $err_D(h) = \Pr_{x \sim D} (h(x) \neq c^*(x))$
  
  How often $h(x) \neq c^*(x)$ over future instances drawn at random from $D$

- **But, can only measure:**
  
  **Training error:** $err_S(h) = \frac{1}{m} \sum_i I(h(x_i) \neq c^*(x_i))$
  
  How often $h(x) \neq c^*(x)$ over training instances

**Sample complexity:** bound $err_D(h)$ in terms of $err_S(h)$
Sample Complexity for Supervised Learning

**Consistent Learner**

- **Input:** $S: (x_1, c^*(x_1)), \ldots, (x_m, c^*(x_m))$
- **Output:** Find $h$ in $H$ consistent with the sample (if one exits).

**Theorem**

$$m \geq \frac{1}{\varepsilon} \left[ \ln(|H|) + \ln \left( \frac{1}{\delta} \right) \right]$$

labeled examples are sufficient so that with prob. $1 - \delta$, all $h \in H$ with $err_D(h) \geq \varepsilon$ have $err_S(h) > 0$.

**Contrapositive:** if the target is in $H$, and we have an algo that can find consistent fns, then we only need this many examples to get generalization error $\leq \varepsilon$ with prob. $\geq 1 - \delta$
Sample Complexity for Supervised Learning

Consistent Learner

• Input: \( S: (x_1,c^*(x_1)),..., (x_m,c^*(x_m)) \)

• Output: Find \( h \) in \( H \) consistent with the sample (if one exits).

Theorem

\[
m \geq \frac{1}{\varepsilon} \left[ \ln(|H|) + \ln \left( \frac{1}{\delta} \right) \right]
\]

labeled examples are sufficient so that with prob. \( 1 - \delta \), all \( h \in H \) with \( \text{err}_D(h) \geq \varepsilon \) have \( \text{err}_S(h) > 0 \).

• \( \varepsilon \) is called error parameter
  • \( D \) might place low weight on certain parts of the space

• \( \delta \) is called confidence parameter
  • there is a small chance the examples we get are not representative of the distribution

Bound inversely linear in \( \varepsilon \)

Bound only logarithmic in \( |H| \)
Sample Complexity for Supervised Learning

Consistent Learner

- **Input:** $S: (x_1, c^*(x_1)), \ldots, (x_m, c^*(x_m))$
- **Output:** Find $h$ in $\mathcal{H}$ consistent with the sample (if one exists).

**Theorem**

$$m \geq \frac{1}{\varepsilon} \left[ \ln(|\mathcal{H}|) + \ln \left( \frac{1}{\delta} \right) \right]$$

labeled examples are sufficient so that with prob. $1 - \delta$, all $h \in \mathcal{H}$ with $err_D(h) \geq \varepsilon$ have $err_S(h) > 0$.

**Example:** $\mathcal{H}$ is the class of conjunctions over $X = \{0,1\}^n$. $|\mathcal{H}| = 3^n$

E.g., $h = x_1 \overline{x}_3 x_5$ or $h = x_1 \overline{x}_2 x_4 x_9$

Then $m \geq \frac{1}{\varepsilon} \left[ n \ln 3 + \ln \left( \frac{1}{\delta} \right) \right]$ suffice

$n = 10, \varepsilon = 0.1, \delta = 0.01$ then $m \geq 156$ suffice
Sample Complexity for Supervised Learning

Consistent Learner

• Input: $S: (x_1, c^*(x_1)), \ldots, (x_m, c^*(x_m))$

• Output: Find $h$ in $H$ consistent with the sample (if one exits).

Theorem

$$m \geq \frac{1}{\varepsilon} \left[ \ln(|H|) + \ln\left(\frac{1}{\delta}\right) \right]$$

labeled examples are sufficient so that with prob. $1 - \delta$, all $h \in H$ with $err_D(h) \geq \varepsilon$ have $err_S(h) > 0$.

Example: $H$ is the class of conjunctions over $X = \{0,1\}^n$.

Side HWK question: show that any conjunctions can be represented by a small decision tree; also by a linear separator.
Sample Complexity for Supervised Learning

Theorem

$$m \geq \frac{1}{\epsilon} \left[ \ln(|H|) + \ln\left(\frac{1}{\delta}\right) \right]$$

labeled examples are sufficient so that with prob. $1 - \delta$, all $h \in H$ with $err_D(h) \geq \epsilon$ have $err_S(h) > 0$.

Proof

Assume $k$ bad hypotheses $h_1, h_2, \ldots, h_k$ with $err_D(h_i) \geq \epsilon$

1) Fix $h_i$. Prob. $h_i$ consistent with first training example is $\leq 1 - \epsilon$.

   Prob. $h_i$ consistent with first $m$ training examples is $\leq (1 - \epsilon)^m$.

2) Prob. that at least one $h_i$ consistent with first $m$ training examples is $\leq k (1 - \epsilon)^m \leq |H|(1 - \epsilon)^m$.

3) Calculate value of $m$ so that $|H|(1 - \epsilon)^m \leq \delta$

3) Use the fact that $1 - x \leq e^{-x}$, sufficient to set $|H|(1 - \epsilon)^m \leq |H| e^{-em} \leq \delta$
Sample Complexity: Finite Hypothesis Spaces

Realizable Case

Theorem

\[ m \geq \frac{1}{\varepsilon} \left[ \ln(|H|) + \ln\left(\frac{1}{\delta}\right) \right] \]

labeled examples are sufficient so that with prob. \( 1 - \delta \), all \( h \in H \) with \( err_D(h) \geq \varepsilon \) have \( err_S(h) > 0 \).

Probability over different samples of m training examples
Sample Complexity: Finite Hypothesis Spaces Realizable Case

1) PAC: How many examples suffice to guarantee small error whp.

Theorem

\[ m \geq \frac{1}{\varepsilon} \left[ \ln(|H|) + \ln \left( \frac{1}{\delta} \right) \right] \]

labeled examples are sufficient so that with prob. \( 1 - \delta \), all \( h \in H \) with \( \text{err}_D(h) \geq \varepsilon \) have \( \text{err}_S(h) > 0 \).

2) Statistical Learning Way:

With probability at least \( 1 - \delta \), for all \( h \in H \) s.t. \( \text{err}_S(h) = 0 \) we have

\[ \text{err}_D(h) \leq \frac{1}{m} \left( \ln |H| + \ln \left( \frac{1}{\delta} \right) \right). \]
Supervised Learning: PAC model (Valiant)

- $X$ - instance space, e.g., $X = \{0,1\}^n$ or $X = \mathbb{R}^n$
- $S = \{(x_i, y_i)\}$ - labeled examples drawn i.i.d. from some distr. $D$ over $X$ and labeled by some target concept $c^*$
  - labels $\in \{-1,1\}$ - binary classification

- Algorithm $A$ PAC-learns concept class $H$ if for any target $c^*$ in $H$, any distr. $D$ over $X$, any $\varepsilon, \delta > 0$:
  - $A$ uses at most $\text{poly}(n,1/\varepsilon,1/\delta,\text{size}(c^*))$ examples and running time.
  - With probab. $1-\delta$, $A$ produces $h$ in $H$ of error at most $\varepsilon$. 
Uniform Convergence

Theorem

\[ m \geq \frac{1}{\varepsilon} \left[ \ln(|H|) + \ln \left( \frac{1}{\delta} \right) \right] \]

labeled examples are sufficient so that with prob. \( 1 - \delta \), all \( h \in H \) with \( \text{err}_D(h) \geq \varepsilon \) have \( \text{err}_S(h) > 0 \).

- This basic result only bounds the chance that a bad hypothesis looks perfect on the data. What if there is no perfect \( h \in H \) (agnostic case)?

- What can we say if \( c^* \notin H \)?
- Can we say that whp all \( h \in H \) satisfy \( |\text{err}_D(h) - \text{err}_S(h)| \leq \varepsilon \)?
  - Called “uniform convergence“.
  - Motivates optimizing over \( S \), even if we can’t find a perfect function.
Sample Complexity: Finite Hypothesis Spaces

Realizable Case

Theorem

\[ m \geq \frac{1}{\varepsilon} \left[ \ln(|H|) + \ln\left(\frac{1}{\delta}\right) \right] \]

labeled examples are sufficient so that with prob. \(1 - \delta\), all \(h \in H\) with \(err_D(h) \geq \varepsilon\) have \(err_S(h) > 0\).

Agnostic Case

What if there is no perfect \(h\)?

Theorem After \(m\) examples, with probab. \(\geq 1 - \delta\), all \(h \in H\) have \(|err_D(h) - err_S(h)| < \varepsilon\), for

\[ m \geq \frac{1}{2\varepsilon^2} \left[ \ln(|H|) + \ln\left(\frac{2}{\delta}\right) \right] \]

To prove bounds like this, need some good tail inequalities.
Consider coin of bias $p$ flipped $m$ times. Let $N$ be the observed # heads. Let $\varepsilon \in [0,1]$.

**Hoeffding bounds:**

- $\Pr[N/m > p + \varepsilon] \leq e^{-2m\varepsilon^2}$, and
- $\Pr[N/m < p - \varepsilon] \leq e^{-2m\varepsilon^2}$.

**Exponentially decreasing tails**

- **Tail inequality:** bound probability mass in tail of distribution (how concentrated is a random variable around its expectation).
Sample Complexity: Finite Hypothesis Spaces

Agnostic Case

**Theorem** After $m$ examples, with probab. $\geq 1 - \delta$, all $h \in H$ have $|err_D(h) - err_S(h)| < \varepsilon$, for

$$m \geq \frac{1}{2\varepsilon^2} \left[ \ln(|H|) + \ln \left( \frac{2}{\delta} \right) \right]$$

- **Proof:** Just apply Hoeffding.
  - Chance of failure at most $2|H|e^{-2|S|\varepsilon^2}$.
  - Set to $\delta$. Solve.

- So, whp, best on sample is $\varepsilon$-best over $D$.
  - Note: this is worse than previous bound ($1/\varepsilon$ has become $1/\varepsilon^2$), because we are asking for something stronger.
  - Can also get bounds “between” these two.
What you should know

• Notion of sample complexity.

• Understand reasoning behind the simple sample complexity bound for finite $H$. 