Efficient Active Learning of Halfspaces: an Aggressive Approach

Alon Gonen\textsuperscript{1}, Sivan Sabato\textsuperscript{2}, and Shai Shalev-Shwartz\textsuperscript{1}

\textsuperscript{1}School of Computer Science and Engineering, The Hebrew University, Jerusalem, Israel

\textsuperscript{2}Microsoft Research, New-England

June 2013
Active Learning

Active learning:
- Receive random unlabelled data
- Choose labels to query
- Return hypothesis with low error
- Cost: label complexity = number of queries

This talk:
Efficient active learning of halfspaces

For the talk: focus on realizable case
Logarithmic Saving in $\mathbb{R}$

- Halfspaces in $\mathbb{R}$ (thresholds):
  - The passive sample complexity is $1/\epsilon$
Logarithmic Saving in $\mathbb{R}$

- Halfspaces in $\mathbb{R}$ (thresholds):
  - The passive sample complexity is $1/\epsilon$

Mellow approach

Ask everything that you don’t know
Logarithmic Saving in $\mathbb{R}$

- Halfspaces in $\mathbb{R}$ (thresholds):
  - The passive sample complexity is $1/\epsilon$

Mellow approach

Ask everything that you don’t know

Aggressive approach

- Query the most informative examples
- For thresholds - binary search
Logarithmic Saving in $\mathbb{R}$

- Halfspaces in $\mathbb{R}$ (thresholds):
  - The passive sample complexity is $1/\epsilon$

Mellow approach

Ask everything that you don’t know

Aggressive approach

- Query the **most informative** examples
- For thresholds - binary search

- For thresholds: label complexity of both approaches: $\log(1/\epsilon)$
Halfspaces in $\mathbb{R}^d$

CAL [Cohn, Atlas, Ladner (1989)] — Mellow approach

Ask everything that you don’t know
The label of $x$ is unknown if for some $h_1, h_2$ which are consistent with the examples observed so far, $h_1(x) \neq h_2(x)$

- CAL can be implemented efficiently for halfspaces using LP

Analysis [Hanneke (2007), Friedman (2009)]
For any “smooth” distribution, the label complexity of CAL is $\log(1/\epsilon)$
Halfspaces in $\mathbb{R}^d$

**CAL** [Cohn, Atlas, Ladner (1989)] — Mellow approach

Ask everything that you don’t know
The label of $x$ is unknown if for some $h_1, h_2$ which are consistent with the examples observed so far, $h_1(x) \neq h_2(x)$

- CAL can be implemented efficiently for halfspaces using LP

**Analysis** [Hanneke (2007), Friedman (2009)]
For any “smooth” distribution, the label complexity of CAL is $\log(1/\epsilon)$

Can we do better using an aggressive approach?
Aggressive can be much better than Mellow

Uniform distribution in $\mathbb{R}^{100}$
Aggressive can be much better than Mellow

Uniform distribution in $\mathbb{R}^{100}$

- Tong and Koller 2003:
  - **Aggressive approach**: Query “most informative” example
  - “most informative” $\approx$ closest to the decision boundary
  - No formal guarantees

- Why doesn’t CAL improve over the ERM?
What is hiding under the hood

- Hidden “constants” in the $\log(1/\epsilon)$ bound
- Dependence on the *disagreement coefficient*
  - The disagreement coefficient can be large
  - We show: label complexity can be $\ll$ disagreement coefficient
- The $\log(1/\epsilon)$ rate can be non-interesting
## Intermediate summary

<table>
<thead>
<tr>
<th></th>
<th>Mellow</th>
<th>Aggressive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Theory</td>
<td>✓</td>
<td>?</td>
</tr>
<tr>
<td>Practice</td>
<td>✗</td>
<td>✓</td>
</tr>
</tbody>
</table>

Our Work: ALuMA: new aggressive learner which works well in practice and enjoys formal guarantees
## Intermediate summary

<table>
<thead>
<tr>
<th></th>
<th>Mellow</th>
<th>Aggressive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Theory</td>
<td>✓</td>
<td>?</td>
</tr>
<tr>
<td>Practice</td>
<td>✕</td>
<td>✓</td>
</tr>
</tbody>
</table>

### Our Work:

ALuMA: new aggressive learner which works well in practice and enjoys formal guarantees
Relative guarantees

- Given a specific pool of instances $X = \{x_1, \ldots, x_m\}$
- Goal: Find labels of $X$ assuming they are separated by a halfspace
- $\text{OPT}(X) :=$ label complexity of the best active learning strategy for $X$
- Implementing the optimal algorithm is intractable
- Can we design an **efficient** active learning algorithm whose label complexity is not much larger than $\text{OPT}(X)$?
A known greedy algorithm for a **small finite** $\mathcal{H}$

- $V = \mathcal{H}$
- For $t = 1, 2, \ldots$
  - For every $x \in X$ let
    
    $$V^+(x) = \{ h \in V : h(x) = +1 \}$$
    
    $$V^-(x) = \{ h \in V : h(x) = -1 \}$$
  - Query the label of $\arg\max_{x \in X} \min\{|V^+(x)|, |V^-(x)|\}$
  - Update $V = \{ h \in V : h(x) = y \}$
- Stop if $V$ is “pure”
A known greedy algorithm for a small finite $\mathcal{H}$

- $V = \mathcal{H}$
- For $t = 1, 2, \ldots$
  - For every $x \in X$ let
    
    $$V^+(x) = \{ h \in V : h(x) = +1 \}$$
    $$V^-(x) = \{ h \in V : h(x) = -1 \}$$
  - Query the label of $\arg\max_{x \in X} \min\{|V^+(x)|, |V^-(x)|\}$
  - Update $V = \{ h \in V : h(x) = y \}$
- Stop if $V$ is “pure”

Golovin and Krause 2010:
The label complexity of the greedy algorithm is $OPT \cdot \log(|\mathcal{H}|)$
Challenge: the class of halfspaces is infinite

Natural idea: consider only equivalence classes of $H$ (w.r.t. $X$)

- By Sauer’s lemma: $|\{h(x_1), \ldots, h(x_m)\}| \leq m^d$
- \[ \Rightarrow \text{Label complexity} \leq OPT(X) \cdot d \log(m) \]
- Leads to an inefficient algorithm
Back to halfspaces

- Challenge: the class of halfspaces is infinite
- Natural idea: consider only equivalence classes of $\mathcal{H}$ (w.r.t. $X$)
  - By Sauer's lemma: $|\{h(x_1), \ldots, h(x_m)\}| \leq m^d$
  - $\Rightarrow$ Label complexity $\leq \text{OPT}(X) \cdot d \log(m)$
  - Leads to an inefficient algorithm
- Instead, consider the full infinite $\mathcal{H} = \{w : \|w\|_2 \leq 1\}$
- This raises the following challenges:
  - Implementing the greedy choice
  - Replacing the factor of $\log(|\mathcal{H}|)$ with finite factor
Implementing the greedy choice

- Replace $|V^+(x)|$ with $\text{Vol}(V^+(x))$

- $V^+(x)$ is a convex polytope:
  - #P-hard to calculate the volume accurately
  - Approximation using a randomized algorithm [Kannan et al. 97]

- We implement an approximate randomized greedy choice
Margin Assumption

- Challenge: \( \log(|\mathcal{H}|) = \infty \)

The margin assumption

The correct halfspace separates the sample with margin \( \gamma \)

The margin assumption

The correct halfspace separates the sample with margin \( \gamma \)
Margin Assumption

- Challenge: \( \log(|\mathcal{H}|) = \infty \)

**The margin assumption**

The correct halfspace separates the sample with margin \( \gamma \)

![Diagram of margin assumption]

**Early termination**

- Terminate when the version space is “pure enough”
- Return a majority vote classifier

\[ \Rightarrow \text{logarithmic dependence on } 1/\gamma \]
Main Theorem

**Theorem**

- If the pool $X$ is separated with margin $\gamma$, then ALuMA’s label complexity $\leq OPT(X) \cdot d \log(1/\gamma)$

**Extensions:**

- Can work with kernels
- Can accommodate unrealizable data
Algorithms:
- ERM (passive)
- CAL (active, mellow)
- QBC (active, slight aggressive)
- TK (active, aggressive)
- ALuMA (active, aggressive)
MNIST (Non-Realizable)

Algorithms:

- soft-SVM (passive)
- IWAL (active, mellow)
- ALuMA (active, aggressive)
Summary

- Being aggressive is good (at least, if you’re active)
- ALuMA is a new aggressive active learning algorithm:
  1. A greedy approach
  2. Efficient implementation
  3. Competitive analysis based on margin assumption
  4. State of the art performance

- Many future directions, e.g.,
  - Multi-class
  - Better understanding of the agnostic case
  - Other interesting classes
Absolute guarantees are impossible

To distinguish between the two cases, all points must be queried.

Label complexity = (passive) sample complexity = $1/\epsilon$

[Dasgupta04]