COS 324: Lecture 14

Boosting

Elad Hazan Yoram Singer

PRINCETON UNIVERSITY

Admin

- Application exercise – due today
- Next theory exercise (decision trees and entropy), next Tue
- Per student request – list of common mistakes in exercises to be posted online

Agenda

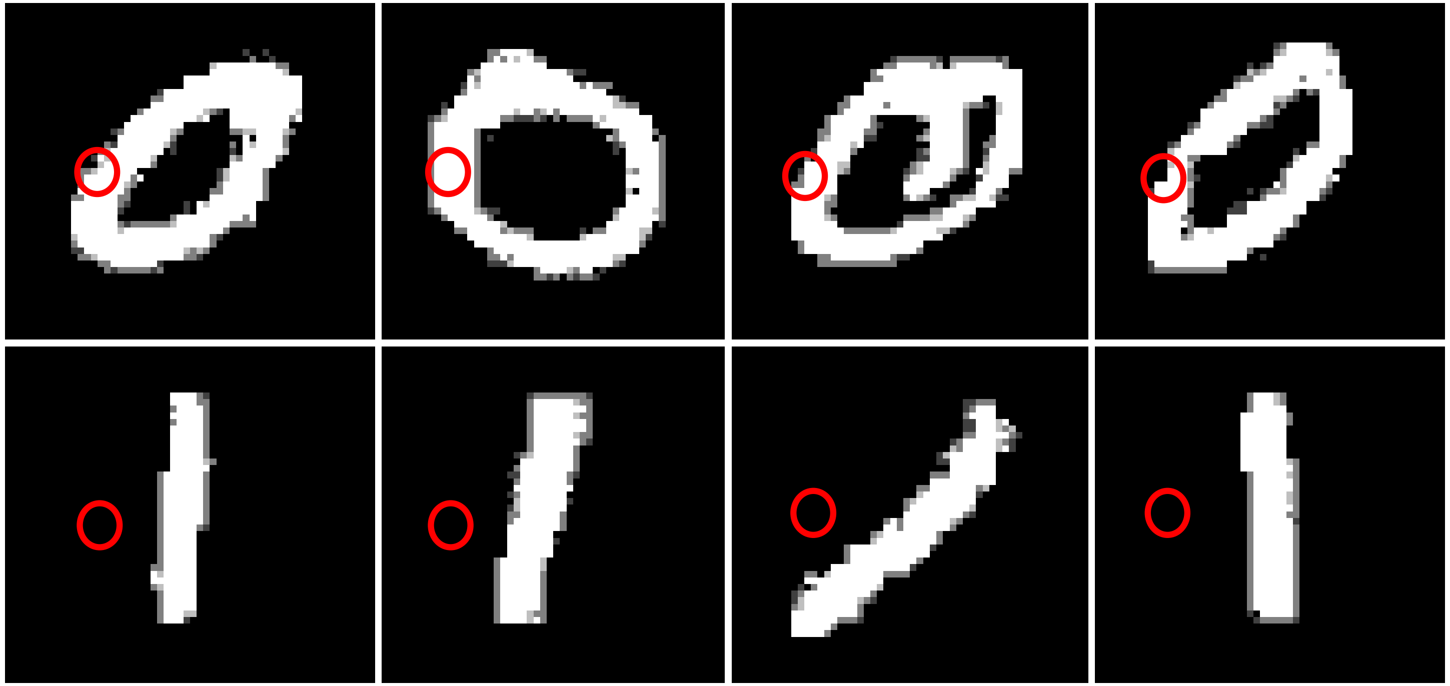
Last lecture:

- Stepped beyond linear classifiers: decision trees
  - Intuitive, easy to interpret, expressive
  - Sample complexity is reasonable (for bounded size)
  - Computationally ill-behaved
  - Thus we looked at efficient heuristics

Today:

- Theoretically sound technique to take a rule of thumb, and turn it into an accurate classifier

Rules of thumb, easy to come by?



Rules of thumb

- One-word classifier for text
- One-pixel classifier for images
- Small decision tree created by CART
- …

Can we turn rules of thumb into accurate classifiers?

Boosting [Schapire]: taking a generic weak-learner (rule of thumb), and using it to PAC learn

Formalizing the boosting question

Learning problem $L = (X, Y, H)$ is PAC-learnable if there exists a learning algorithm s.t. for every $\delta, \epsilon > 0$, there exists $\mathrm{m} = f(\epsilon, \delta, H) < \infty$, s.t. after observing S examples from any distribution, for $|S| = m$, returns a hypothesis $h \in H$, such that with probability at least

$$1 - \delta$$

it holds that

$$\Pr[h(x) \neq y] = err(h) \leq \epsilon$$

Formalizing the boosting question

Learning problem $L = (X, Y, H)$ is weakly PAC-learnable if there exists a learning algorithm (called **weak learner**) s.t. for some $\gamma > 0$, there exists $\mathrm{m} = f(\epsilon, \delta, H) < \infty$, s.t. after observing S examples from any distribution, for $|S| = m$, returns a hypothesis $h \in H$, such that with probability at least

$$\frac{2}{3}$$

it holds that

$$\Pr[h(x) \neq y] = err(h) \leq \frac{1}{2} - \gamma$$

The boosting question

- Is weak PAC learnability equivalent to (strong) PAC learnability?
- i.e., does there exist an efficient algorithm that takes as an input a weak learner, and converts it to a strong learner?

- Answer: Yes! [Shapire ‘90], culminating in the AdaBoost algorithm [Freund and Schapire ‘93]

Boosting the error probability

For any randomized algorithm that succeeds with probability 2/3:
Repeat $\mathrm{O}(\log(\frac{1}{\delta}))$ times , and with probability at least $1-\delta$, it will succeed at least once!

Proof sketch: prob of failure is upper bounded by:

$$\prod_{i=1\ldots O(\log\frac{1}{\delta})}\left(1-\frac{2}{3}\right)=\left(1-\frac{2}{3}\right)^{\mathrm{O}(\log\frac{1}{\delta})} \leq 1-\delta$$

Formalizing the boosting question

Learning problem $L = (X, Y, H)$ is weakly PAC-learnable if there exists a learning algorithm (called **weak learner**) s.t. for some $\gamma > 0$, and every $\delta > 0$, there exists $\mathrm{m} = f(\epsilon, \delta, H) < \infty$, s.t. after observing S examples from any distribution, for $|S| = m$, returns a hypothesis $h \in H$, such that with probability at least

$$1 - \delta$$

it holds that

$$\Pr[h(x) \neq y] = err(h) \leq \frac{1}{2} - \gamma$$

General idea

change the distribution of the examples to focus on the hard instance, and every time find a weak learner for the "harder" distribution.

Finally, combine all weak learners into one rule.

Multiplicative updates!

1. How to change the distribution over examples?
2. How to combine all weak learners?

Use majority vote

Simple boosting algorithm

- Input: learning problem $L = (X, Y, H)$, weak learner for L
- Output: strong learner for L, i.e. hypothesis such that

$$err(h) \leq \epsilon$$

Simple boosting algorithm

1. Take $\mathrm{m} = \frac{\dim(H) + \log\frac{1}{\delta}}{\epsilon}$ samples from distribution of L, call it S
2. Let $p_1 = unif(m)$ be the uniform distribution over S

What happens if we find h that has zero error on S??

Simple boosting algorithm

1. Take m samples from distribution of L, call it S (think of $\mathrm{m} = \frac{\dim(H)+\log\frac{1}{\delta}}{\epsilon}$)
2. Let $p_1 = unif(m)$ be the uniform distribution over S
3. For t =1,2,…,T do:
   1. Let h_t be the output of the weak learner on current distribution p_t
   2. Update distribution by multiplicative update rule:

$$p_{t+1}(i) = \frac{p_t(i)(1-\epsilon)^{r_t(i)}}{\sum_i p_t(i)(1-\epsilon)^{r_t(i)}}$$

$$r_t(i) = 1_{h_t(x_i)=y_i}$$

4. Return the majority of all hypothesis:

$$\bar{h}(x) = majority\big(h_1(x), h_2(x), \dots, h_T(x)\big)$$
$$= sign(\sum_t h_t(x) \; - \frac{T}{2})$$

Simple boosting algorithm

Theorem:

$$err_S(\bar{h}) = 0$$

(and hence the generalization error of $\bar{h}$ is at most ϵ, though there's a slight subtlety here we'll not go into)

Proof of simple boosting alg guarantee

Observation 1:

by the definition of r_t, p_t, ($r_t(i) = 1_{h_t(x_i)=y_i}$) and the weak learning guarantee, we have that

$$r_t^{\top} p_t \geq \frac{1}{2} + \gamma$$

And thus

$$\frac{1}{T}\sum_t r_t^{\top} p_t \geq \frac{1}{2} + \gamma$$

Proof of simple boosting alg guarantee

Observation 2:

by the online-learning multiplicative weights guarantee: (lecture 2+3)

$$\sum_t r_t^\top p_t \le (1+\epsilon) \sum_t r_t(i^*) + \frac{\log m}{\epsilon}$$

Take $\epsilon = \gamma$

$$\sum_t r_t^\top p_t \le (1+\gamma) \sum_t r_t(i^*) + \frac{\log m}{\gamma}$$

Proof of simple boosting alg guarantee

From both observations:
Suppose that some example i^* has more than ½ errors, then:

$$\frac{1}{2}+\gamma \leq \frac{1}{T}\sum_{t} r_t^{\mathsf{T}} p_t \leq \frac{(1+\gamma)}{T}\sum_{t} r_t(i^*) + \frac{\log m}{T\gamma} \leq \frac{1+\gamma}{2} + \frac{\log m}{T\gamma}$$

Take $T = \frac{4 \log m}{\gamma^2}$, we get:

$$\frac{1}{2}+\gamma \leq \frac{1}{2}+\frac{3\gamma}{4}$$

By contradiction!
Thus, after T iterations all examples are correctly classified by majority!

Simple boosting algorithm

We concluded with the theorem:

$$err_S(\bar{h}) = 0$$

(and hence the generalization error of $\bar{h}$ is at most ϵ, though there's a slight subtlety here we'll not go into)

AdaBoost

1. Take m samples from distribution of L, call it S (think of $m = \frac{\dim(H)+\log\frac{1}{\delta}}{\epsilon}$)
2. Let $p_1 = unif(m)$ be the uniform distribution over S

3. For t =1,2,…,T do:
   1. Let h_t be the output of the weak learner on current distribution p_t
   2. Update distribution by multiplicative update rule:

$$p_{t+1}(i) = \frac{p_t(i)(1-\epsilon_t)^{r_t(i)}}{\sum_i p_t(i)(1-\epsilon_t)^{r_t(i)}}$$

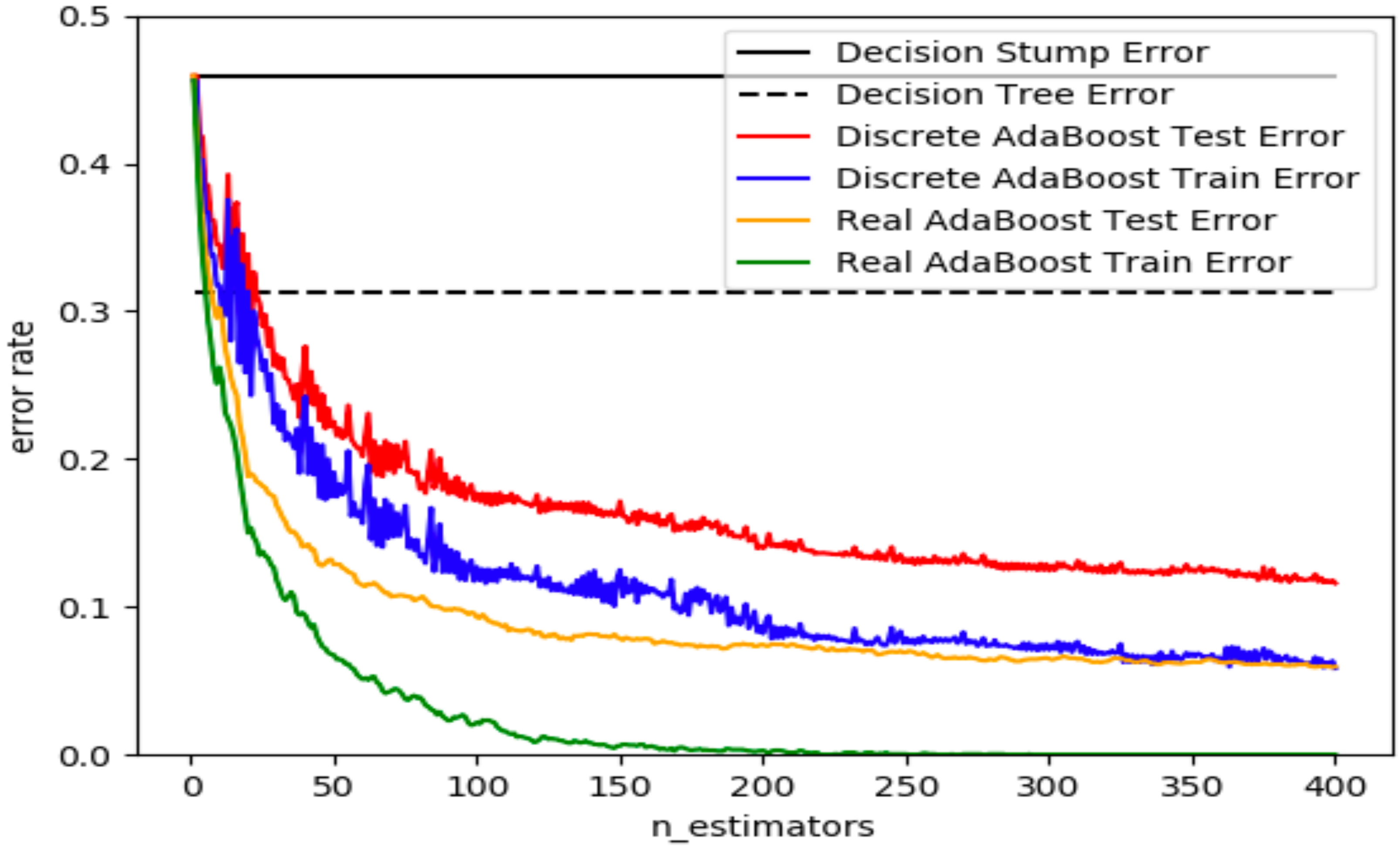
$$r_t(i) = 1_{h_t(x_i)=y_i}$$

For $\epsilon_t = r_t^{\top} p_t - \frac{1}{2}$

4. Return the weighted majority of all hypothesis:

$$\bar{h}(x) = sign(\sum_i \epsilon_t h_t(x) \quad {}_1 - \frac{T}{2})$$

AdaBoost in practice



AdaBoost in practice