Machine Teaching

Jerry Zhu

Department of Computer Sciences
University of Wisconsin-Madison

ICML 2015 Workshop on Machine Learning for Education

Example 1: Teaching a support vector machine

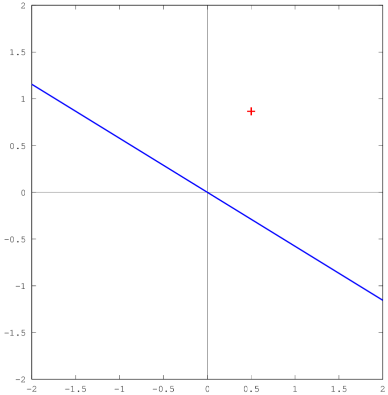
- Here is an SVM: $\min_{\theta \in \mathbb{R}^2} \sum_{i=1}^n \max(1 - y_i \mathbf{x}_i^\top \theta, 0) + \frac{1}{2}\|\theta\|^2$

Example 1: Teaching a support vector machine

- Here is an SVM: $\min_{\theta \in \mathbb{R}^2} \sum_{i=1}^n \max(1 - y_i \mathbf{x}_i^\top \theta, 0) + \frac{1}{2}\|\theta\|^2$
- What (batch) training set can teach the model: $\theta^* = (\frac{1}{2}, \frac{\sqrt{3}}{2})^\top$?

Example 1: Teaching a support vector machine

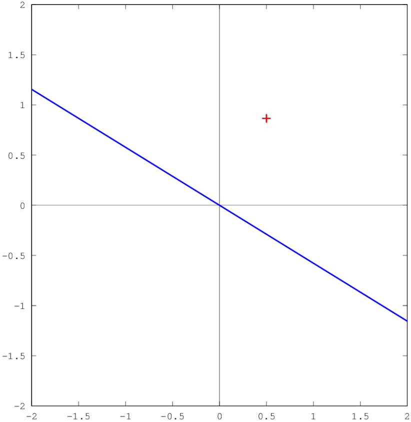
- Here is an SVM: $\min_{\theta\in\mathbb{R}^2} \sum_{i=1}^n \max(1 - y_i \mathbf{x}_i^\top \theta, 0) + \frac{1}{2}\|\theta\|^2$
- What (batch) training set can teach the model: $\theta^* = (\frac{1}{2}, \frac{\sqrt{3}}{2})^\top$?
- Teach the exact θ^*, not just the decision boundary $\mathbf{x}^\top\theta^* = 0$



One training item is necessary and sufficient!

$$x_1 = (\frac{1}{2}, \frac{\sqrt{3}}{2})^\top,\ y_1 = 1$$

You don't even need negative training items.



What's the catch?

[Liu & Zhu unpublished]

Theorem (Teaching Dimension of homogeneous SVM)

To teach any target model $\theta^* \neq 0$ *to a homogeneous SVM*

$$\min_{\theta \in \mathbb{R}^2} \sum_{i=1}^{n} \max(1 - y_i \mathbf{x}_i^\top \theta, 0) + \frac{\lambda}{2} \|\theta\|^2$$

one needs $n = \lceil \lambda \|\theta^*\|^2 \rceil$ *training items.*

What's the catch?

[Liu & Zhu unpublished]

Theorem (Teaching Dimension of homogeneous SVM)

To teach any target model $\theta^* \neq 0$ *to a homogeneous SVM*

$$\min_{\theta \in \mathbb{R}^2} \sum_{i=1}^{n} \max(1 - y_i \mathbf{x}_i^\top \theta, 0) + \frac{\lambda}{2} \|\theta\|^2$$

one needs $n = \lceil \lambda \|\theta^*\|^2 \rceil$ *training items.*

Proposition

One such training set is $n = \lceil \lambda \|\theta^*\|^2 \rceil$ *identical copies of the following item:*

$$\mathbf{x} = \frac{\lambda \theta^*}{n}, \quad y = 1.$$

Example 2: Teaching a Gaussian density estimator

- Given $\mathbf{x}_1 \ldots \mathbf{x}_n \in \mathbb{R}^d$, the student computes sample mean and sample covariance:

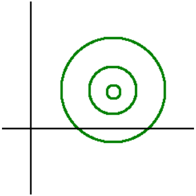
$$\hat{\mu} = \frac{1}{n}\sum \mathbf{x}_i, \quad \hat{\Sigma} = \frac{1}{n-1}\sum (\mathbf{x}_i - \hat{\mu})(\mathbf{x}_i - \hat{\mu})^\top$$

Example 2: Teaching a Gaussian density estimator

- Given $\mathbf{x}_1 \ldots \mathbf{x}_n \in \mathbb{R}^d$, the student computes sample mean and sample covariance:

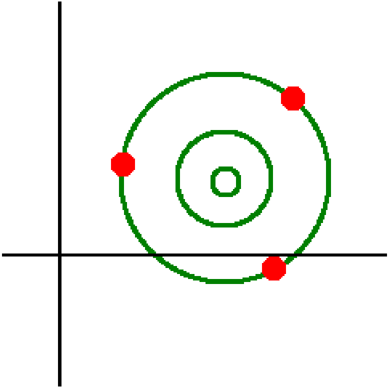
$$\hat{\mu} = \frac{1}{n} \sum \mathbf{x}_i, \quad \hat{\Sigma} = \frac{1}{n-1} \sum (\mathbf{x}_i - \hat{\mu})(\mathbf{x}_i - \hat{\mu})^\top$$

- The teacher wants to teach the model $N(\mu^*, \Sigma^*)$



$d + 1$ training items necessary and sufficient

Vertices of d-dim tetrahedron



Two sides of a coin

- Machine learning: given data D, find model θ

Two sides of a coin

- Machine learning: given data D, find model θ
- Machine teaching: given model θ, find (the smallest) data D

Two sides of a coin

- Machine learning: given data D, find model θ
- Machine teaching: given model θ, find (the smallest) data D
  - for any given learner

Two sides of a coin

- Machine learning: given data D, find model θ
- Machine teaching: given model θ, find (the smallest) data D
  - for <u>any</u> given learner
  - D will usually not be $i.i.d.$

Two sides of a coin

- Machine learning: given data D, find model θ
- Machine teaching: given model θ, find (the smallest) data D
  - for any given learner
  - D will usually not be $i.i.d.$
  - studied as optimal teaching [Goldman & Kearns 1995, many others]

Two sides of a coin

- Machine learning: given data D, find model θ
- Machine teaching: given model θ, find (the smallest) data D
  - for any given learner
  - D will usually not be $i.i.d.$
  - studied as optimal teaching [Goldman & Kearns 1995, many others]
  - traditional emphasis: version-space learners

Two sides of a coin

- Machine learning: given data D, find model θ
- Machine teaching: given model θ, find (the smallest) data D
    - for any given learner
    - D will usually not be $i.i.d.$
    - studied as optimal teaching [Goldman & Kearns 1995, many others]
    - traditional emphasis: version-space learners
    - our emphasis: modern optimization-based learners

(Optimization-based) machine learning

Given $D = \{z_1 \ldots z_n\}$, we consider any learner with regularized empirical risk minimization:

$$\hat{\theta} \leftarrow \operatorname*{argmin}_{\theta \in \Theta} \frac{1}{|D|} \sum_{z_i \in D} \ell(z_i, \theta) + \Omega(\theta)$$

- $z_i = (x_i, y_i)$ for supervised learning, $z_i = x_i$ for unsupervised learning

(Optimization-based) machine learning

Given $D = \{z_1 \ldots z_n\}$, we consider any learner with regularized empirical risk minimization:

$$\hat{\theta} \leftarrow \operatorname*{argmin}_{\theta \in \Theta} \frac{1}{|D|} \sum_{z_i \in D} \ell(z_i, \theta) + \Omega(\theta)$$

- $z_i = (x_i, y_i)$ for supervised learning, $z_i = x_i$ for unsupervised learning
- $|D| = n$

(Optimization-based) machine learning

Given $D = \{z_1 \dots z_n\}$, we consider any learner with regularized empirical risk minimization:

$$\hat{\theta} \leftarrow \operatorname*{argmin}_{\theta \in \Theta} \frac{1}{|D|} \sum_{z_i \in D} \ell(z_i, \theta) + \Omega(\theta)$$

- $z_i = (x_i, y_i)$ for supervised learning, $z_i = x_i$ for unsupervised learning
- $|D| = n$
- $\ell(\cdot, \cdot)$ a loss function

(Optimization-based) machine learning

Given $D = \{z_1 \ldots z_n\}$, we consider any learner with regularized empirical risk minimization:

$$\hat{\theta} \leftarrow \operatorname*{argmin}_{\theta \in \Theta} \frac{1}{|D|} \sum_{z_i \in D} \ell(z_i, \theta) + \Omega(\theta)$$

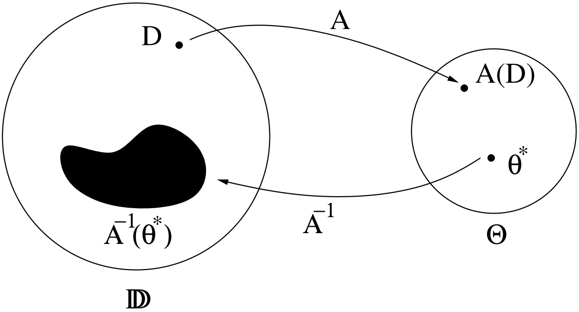
- $z_i = (x_i, y_i)$ for supervised learning, $z_i = x_i$ for unsupervised learning
- $|D| = n$
- $\ell(\cdot, \cdot)$ a loss function
- $\Omega(\cdot)$ a regularizer

(Optimization-based) machine learning

Given $D = \{z_1 \ldots z_n\}$, we consider any learner with regularized empirical risk minimization:

$$\hat{\theta} \leftarrow \operatorname*{argmin}_{\theta \in \Theta} \frac{1}{|D|} \sum_{z_i \in D} \ell(z_i, \theta) + \Omega(\theta)$$

- $z_i = (x_i, y_i)$ for supervised learning, $z_i = x_i$ for unsupervised learning
- $|D| = n$
- $\ell(\cdot, \cdot)$ a loss function
- $\Omega(\cdot)$ a regularizer
- argmin is the learning algorithm A


D
A
A(D)
θ*
A⁻¹(θ*)
A⁻¹
Θ
𝔻

Machine teaching

Given θ^* and A, find the smallest training set:

$$D^* \leftarrow \operatorname*{argmin}_{D \in \mathbb{D}} \quad |D| \qquad \text{Teacher's problem}$$

$$\text{s.t.} \qquad \theta^* = \operatorname*{argmin}_{\theta \in \Theta} \frac{1}{|D|} \sum_{z_i \in D} \ell(z_i, \theta) + \Omega(\theta) \qquad \text{learner's algorithm } A$$

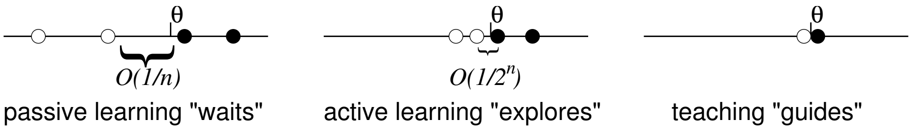
Bilevel optimization

Solution idea

Convert lower level problem to nonlinear constraints:

$$\begin{aligned} D^* \leftarrow \operatorname*{argmin}_{D \in \mathbb{D}} \quad & |D| \\ \text{s.t.} \quad & \text{Karush-Kuhn-Tucker conditions of } A \text{ at } \theta^* \end{aligned}$$

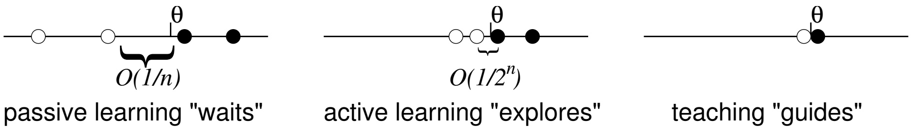
Machine teaching is stronger than active learning



Sample complexity to achieve ϵ error

- passive learning $1/\epsilon$

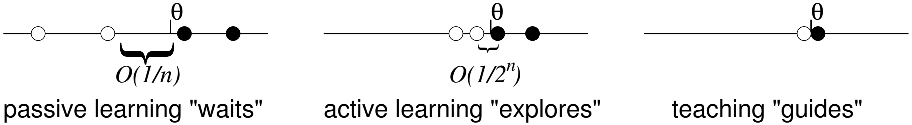
Machine teaching is stronger than active learning



Sample complexity to achieve ϵ error

- passive learning $1/\epsilon$
- active learning $\log(1/\epsilon)$

Machine teaching is stronger than active learning



Sample complexity to achieve ϵ error

- passive learning $1/\epsilon$
- active learning $\log(1/\epsilon)$
- machine teaching 2: the teacher knows θ

Education

- Assumption 1: The educational goal can be reasonably approximated by an objective function on θ^*

Education

- Assumption 1: The educational goal can be reasonably approximated by an objective function on θ^*
  - e.g. $\|\hat{\theta} - \theta^*\|^2$

Education

- Assumption 1: The educational goal can be reasonably approximated by an objective function on θ^*
  - e.g. $\|\hat{\theta} - \theta^*\|^2$
  - e.g. test set accuracy

Education

- Assumption 1: The educational goal can be reasonably approximated by an objective function on θ^*
  - e.g. $\|\hat{\theta} - \theta^*\|^2$
  - e.g. test set accuracy
- Assumption 2: The student can be reasonably approximated by a machine learning algorithm $A : \mathbb{D} \mapsto \Theta$

Education

- Assumption 1: The educational goal can be reasonably approximated by an objective function on θ^*
  - e.g. $\|\hat{\theta} - \theta^*\|^2$
  - e.g. test set accuracy
- Assumption 2: The student can be reasonably approximated by a machine learning algorithm $A : \mathbb{D} \mapsto \Theta$
  - e.g. regression, SVM, neural network

Machine teaching for education

1. $D^* \leftarrow \mathrm{MachineTeaching}(\theta^*, A)$

Machine teaching for education

1. $D^* \leftarrow \mathrm{MachineTeaching}(\theta^*, A)$
2. Train human on D^*

Machine teaching for education

1. $D^* \leftarrow \text{MachineTeaching}(\theta^*, A)$
2. Train human on D^*
3. Test human on a test set

Machine teaching for education

1. $D^* \leftarrow \text{MachineTeaching}(\theta^*, A)$
2. Train human on D^*
3. Test human on a test set

Machine teaching for education

1. $D^* \leftarrow \mathrm{MachineTeaching}(\theta^*, A)$
2. Train human on D^*
3. Test human on a test set

D^* should be better than any other lesson D by definition!

Machine teaching for education

1. $D^* \leftarrow \text{MachineTeaching}(\theta^*, A)$
2. Train human on D^*
3. Test human on a test set

D^* should be better than any other lesson D by definition!

What if it isn't?

Machine teaching for education

1. $D^* \leftarrow \mathrm{MachineTeaching}(\theta^*, A)$
2. Train human on D^*
3. Test human on a test set

D^* should be better than any other lesson D by definition!
What if it isn't?
. . . blame yourself (choice of A)

Machine teaching for education

1. $D^* \leftarrow \text{MachineTeaching}(\theta^*, A)$
2. Train human on D^*
3. Test human on a test set

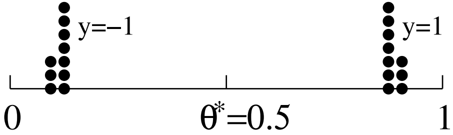
D^* should be better than any other lesson D by definition!

What if it isn't?

... blame yourself (choice of A)

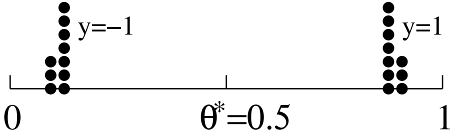
... blame us (bilevel optimization)

Evidence from cognitive psychology



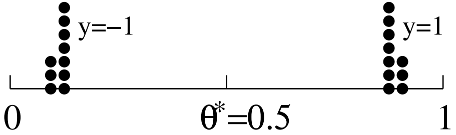
- Human categorization [Patil Z Kopeć Love 2014]

Evidence from cognitive psychology



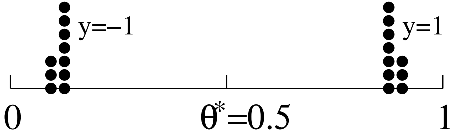
- Human categorization [Patil Z Kopeć Love 2014]
- A is a limited capacity retrieval cognitive model $\approx$ kernel density classifier

Evidence from cognitive psychology



- Human categorization [Patil Z Kopeć Love 2014]
- A is a limited capacity retrieval cognitive model ≈ kernel density classifier

Evidence from cognitive psychology



- Human categorization [Patil Z Kopeć Love 2014]
- A is a limited capacity retrieval cognitive model $\approx$ kernel density classifier

human trained on	human test accuracy
optimal lesson D^*	72.5%
iid	69.8%

(statistically significant)

Open research questions in machine teaching

- approximate teaching: $\|\hat{\theta} - \theta^*\| \leq \epsilon$

Open research questions in machine teaching

- approximate teaching: $\|\hat{\theta} - \theta^*\| \leq \epsilon$
- teaching under budget: $n \leq B$

Open research questions in machine teaching

- approximate teaching: $\|\hat{\theta} - \theta^*\| \leq \epsilon$
- teaching under budget: $n \leq B$
- uncertainty in learner A: unknown λ

Open research questions in machine teaching

- approximate teaching: $\|\hat{\theta} - \theta^*\| \leq \epsilon$
- teaching under budget: $n \leq B$
- uncertainty in learner A: unknown λ
- sequential learner

Open research questions in machine teaching

- approximate teaching: $\|\hat{\theta} - \theta^*\| \leq \epsilon$
- teaching under budget: $n \leq B$
- uncertainty in learner A: unknown λ
- sequential learner
- reinforcement learner

Open research questions in machine teaching

- approximate teaching: $\|\hat{\theta} - \theta^*\| \leq \epsilon$
- teaching under budget: $n \leq B$
- uncertainty in learner A: unknown λ
- sequential learner
- reinforcement learner
- real world applications

Open research questions in machine teaching

- approximate teaching: $\|\hat{\theta} - \theta^*\| \le \epsilon$
- teaching under budget: $n \le B$
- uncertainty in learner A: unknown λ
- sequential learner
- reinforcement learner
- real world applications
- ...

Summary

$D^* \leftarrow \mathrm{MachineTeaching}(\theta^*, A)$

- not machine learning

Summary

$D^* \leftarrow \mathrm{MachineTeaching}(\theta^*, A)$

- not machine learning
- reverse engineering

Summary

$D^* \leftarrow \mathrm{MachineTeaching}(\theta^*, A)$

- not machine learning
- reverse engineering
- potential new paradigm for education

Summary

$D^* \leftarrow \mathrm{MachineTeaching}(\theta^*, A)$

- not machine learning
- reverse engineering
- potential new paradigm for education

Summary

$D^* \leftarrow \text{MachineTeaching}(\theta^*, A)$

- not machine learning
- reverse engineering
- potential new paradigm for education

`http://pages.cs.wisc.edu/~jerryzhu/machineteaching/`

Summary

$D^* \leftarrow \mathrm{MachineTeaching}(\theta^*, A)$

- not machine learning
- reverse engineering
- potential new paradigm for education

`http://pages.cs.wisc.edu/~jerryzhu/machineteaching/`

Collaborators: Scott Alfeld, Martha Alibali, Michael Ferris, Ji Liu, Bradley Love, Percival Matthews, Shike Mei, Bilge Mutlu, Gorune Ohannessian, Martina Rau, Tim Rogers, Ayon Sen, Steve Wright.