

Low-Cost Learning via Active Data Procurement

October 2015

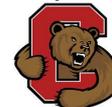
Jacob Abernethy



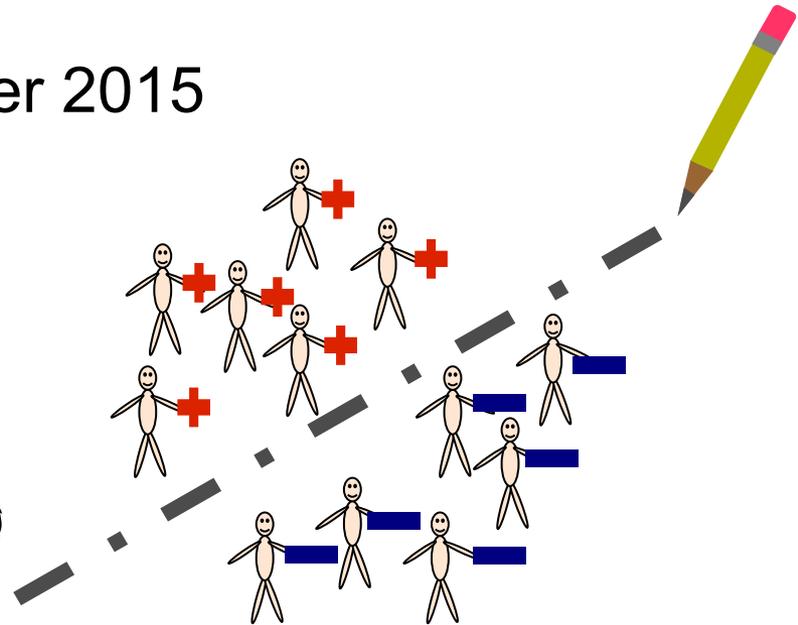
Yiling Chen



Chien-Ju Ho

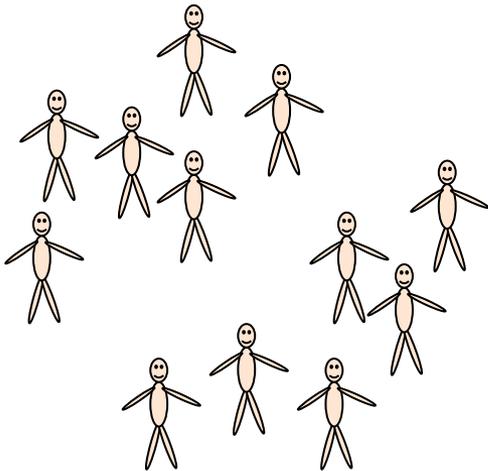


Bo Waggoner



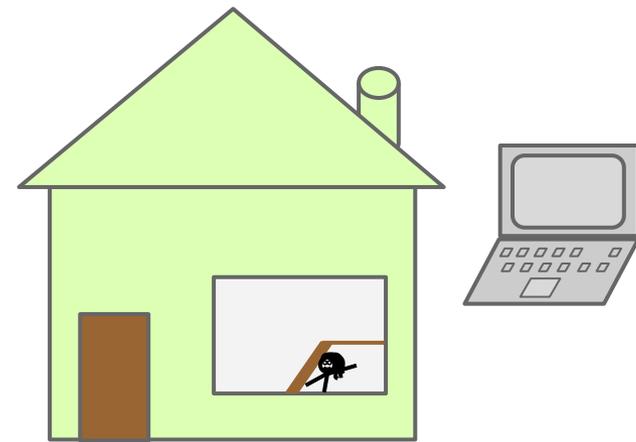
Coming soon to a society near you

data-holders



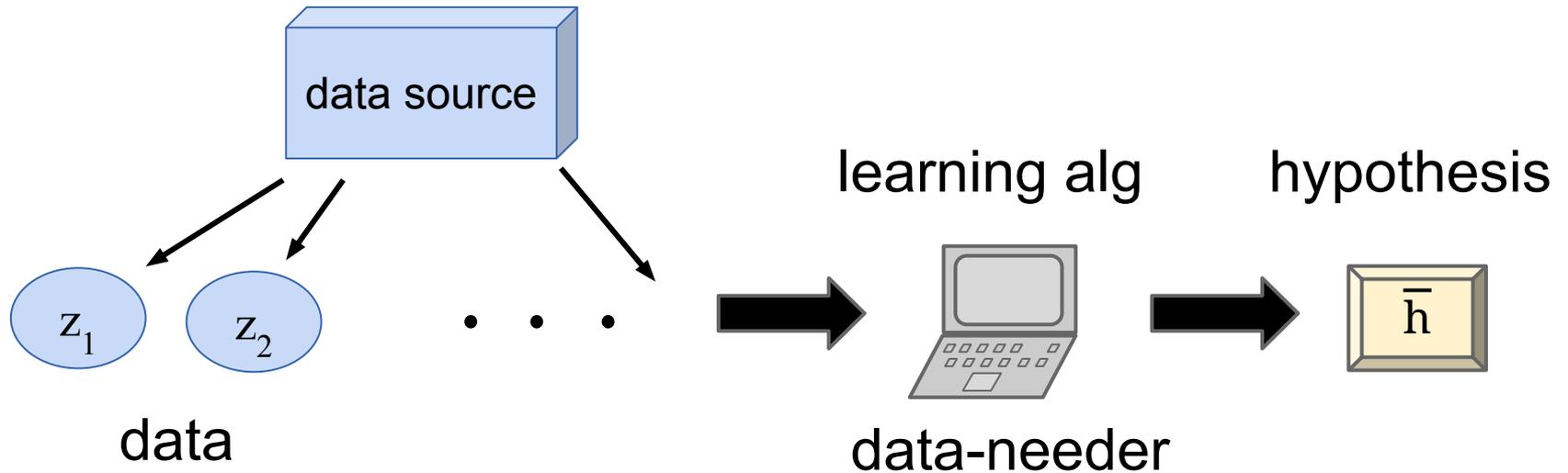
ex: medical data

data-needers



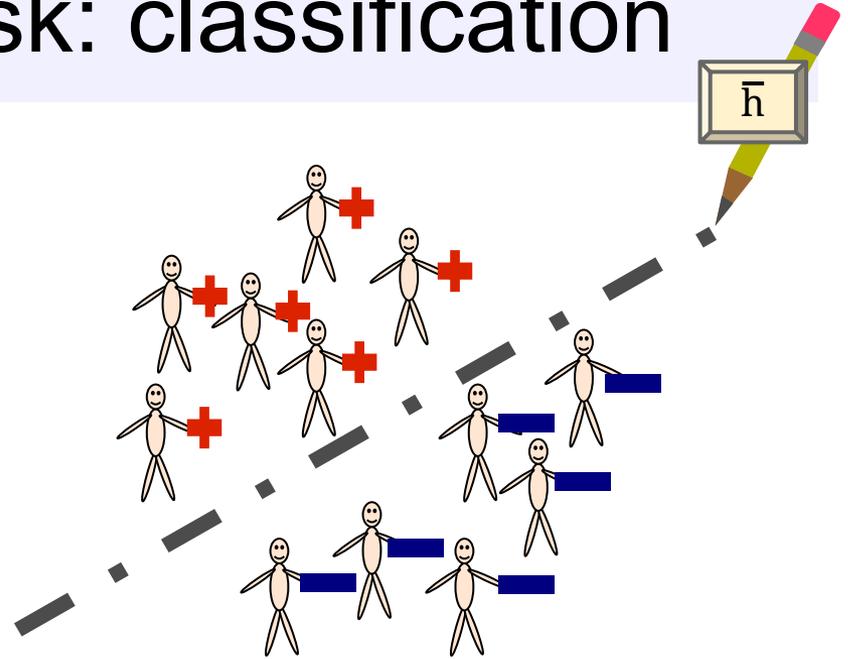
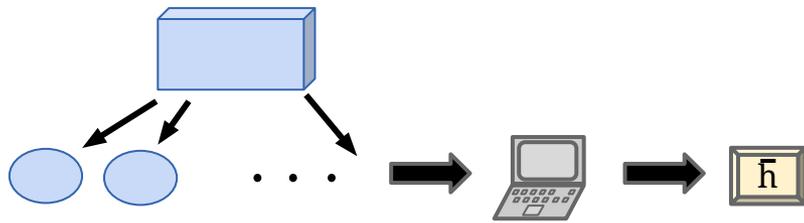
ex: pharmaceutical co.

Classic ML problem



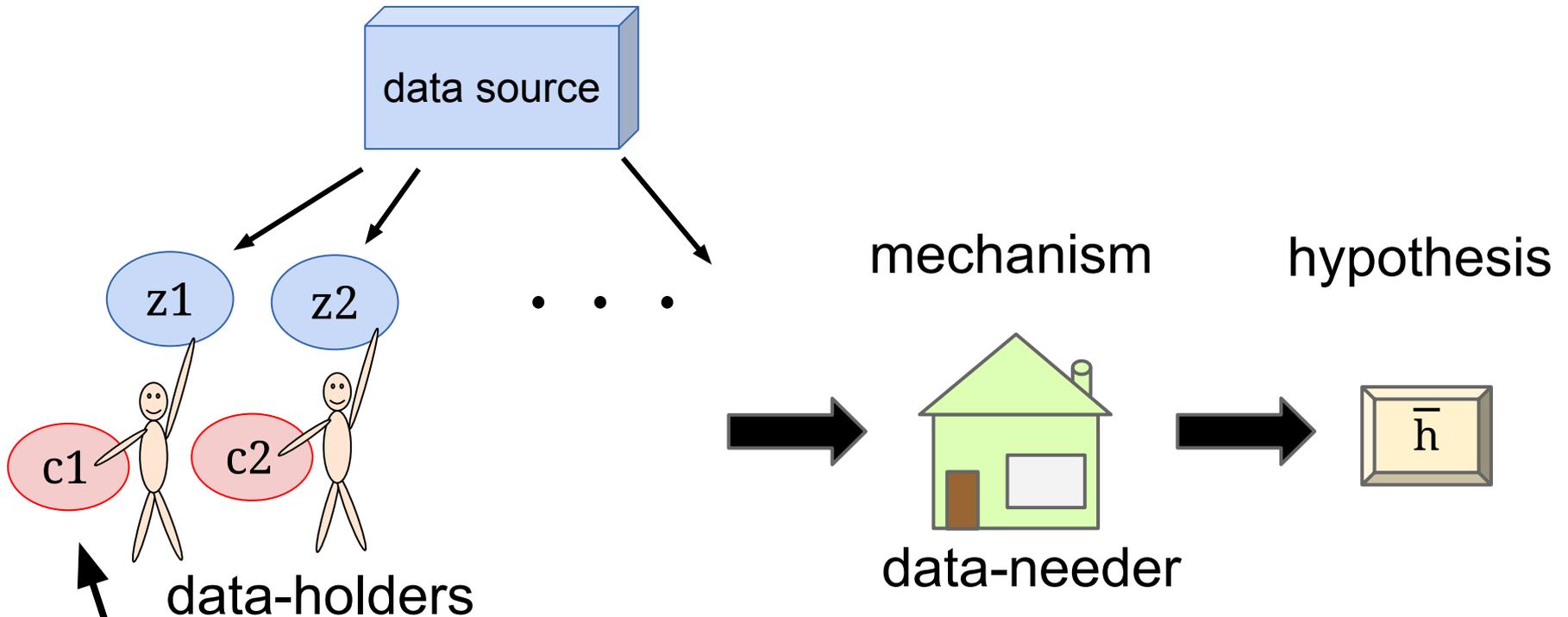
Goal: use small amount of data, output “good” h .

Example learning task: classification



- **Data:** (point, label) where label is $+$ or $-$
- **Hypothesis:** hyperplane separating the two types

Twist: data is now held by individuals

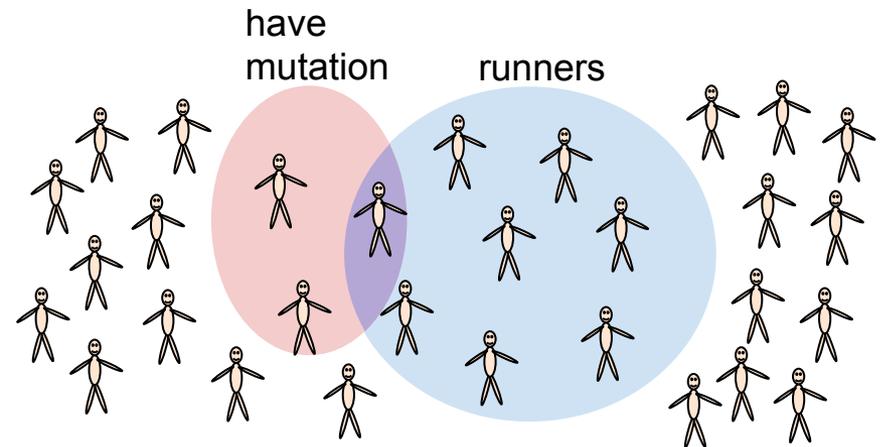
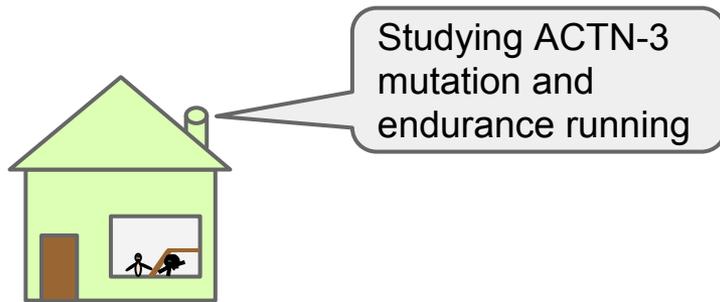


“Cost of revealing data” (formal model later...)

Goal: spend small budget, output “good” h .

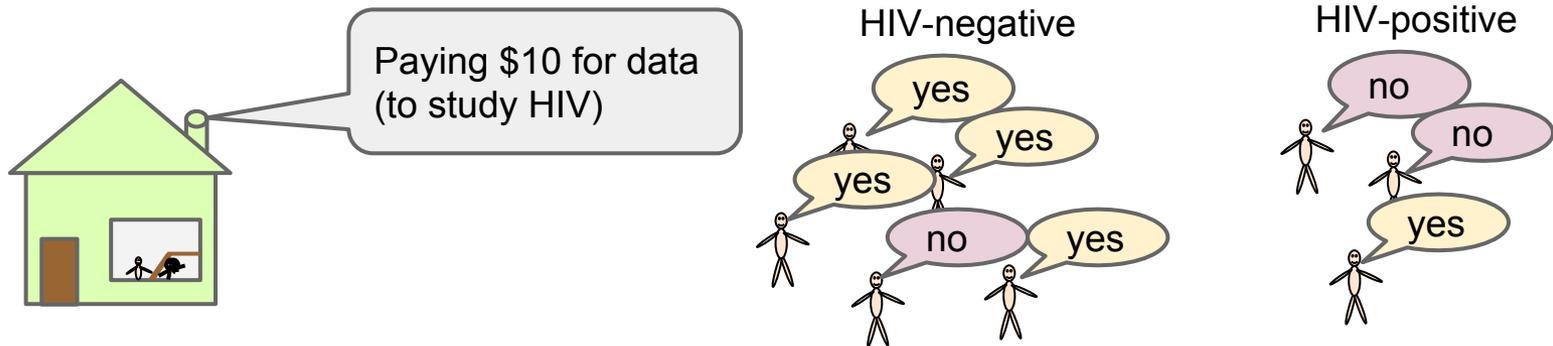
Why is this difficult?

1. (Relatively) few data are **useful**



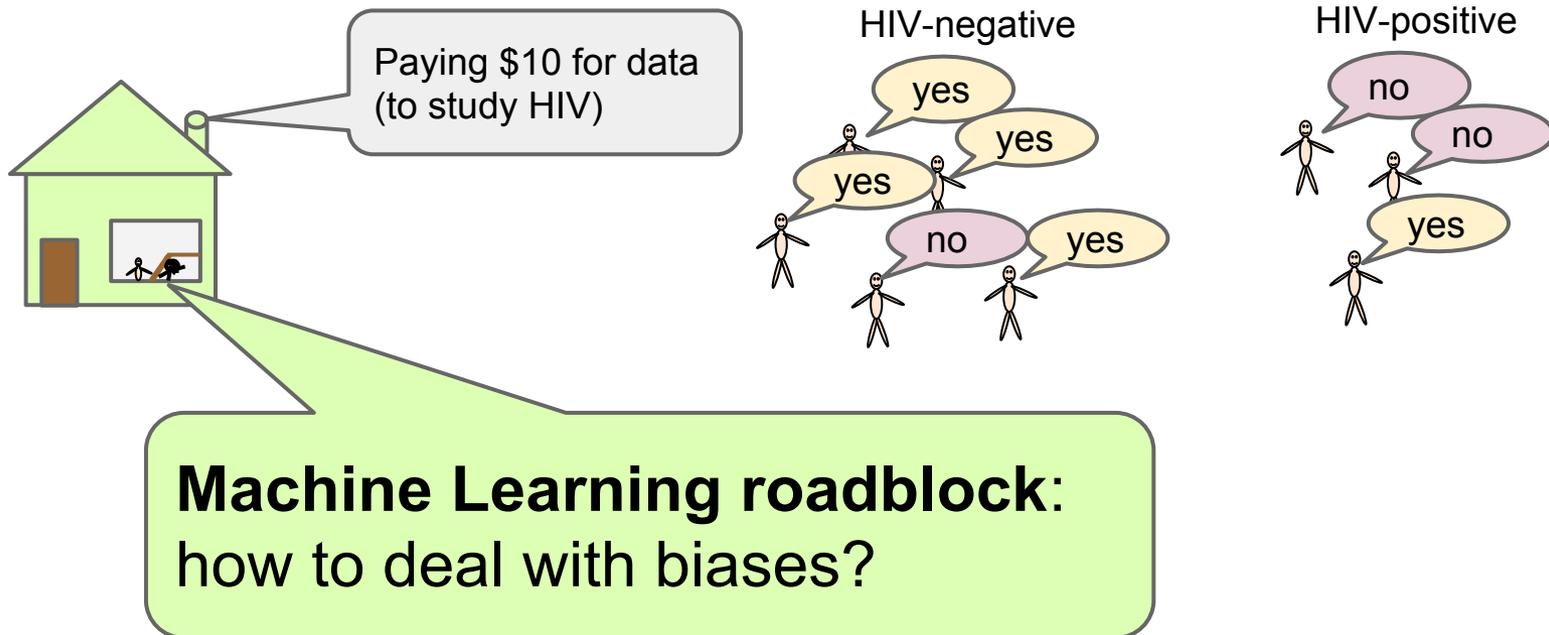
Why is this difficult?

2. Utility of data may be **correlated** with cost (causing bias)



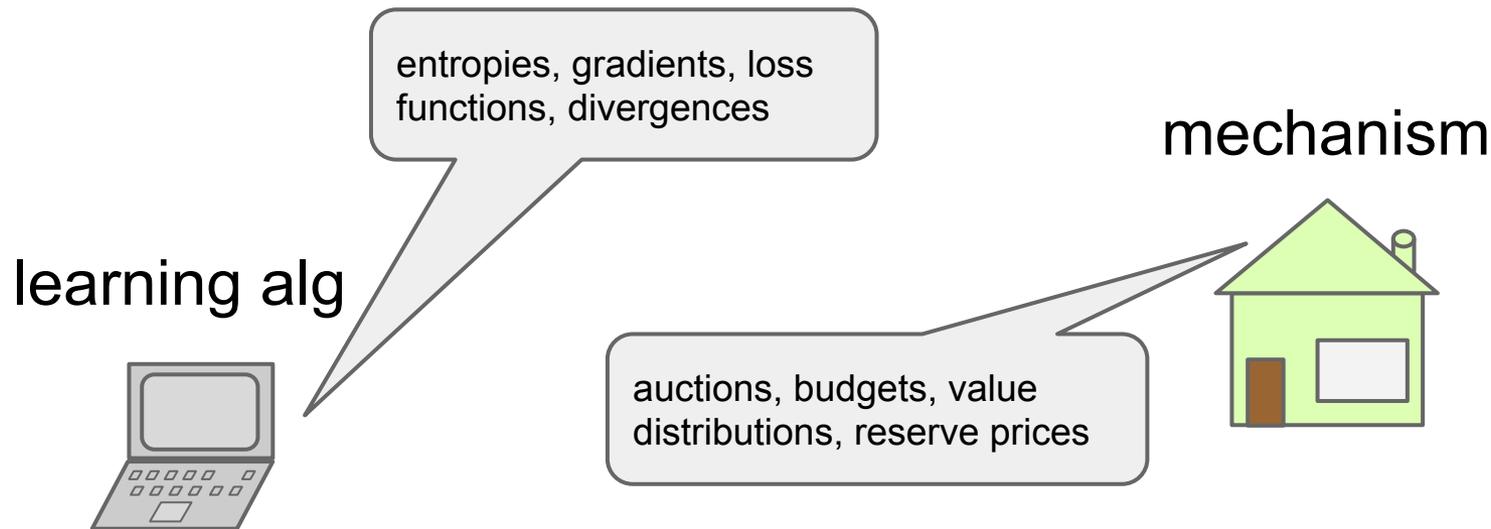
Why is this difficult?

2. Utility of data may be **correlated** with cost (causing bias)



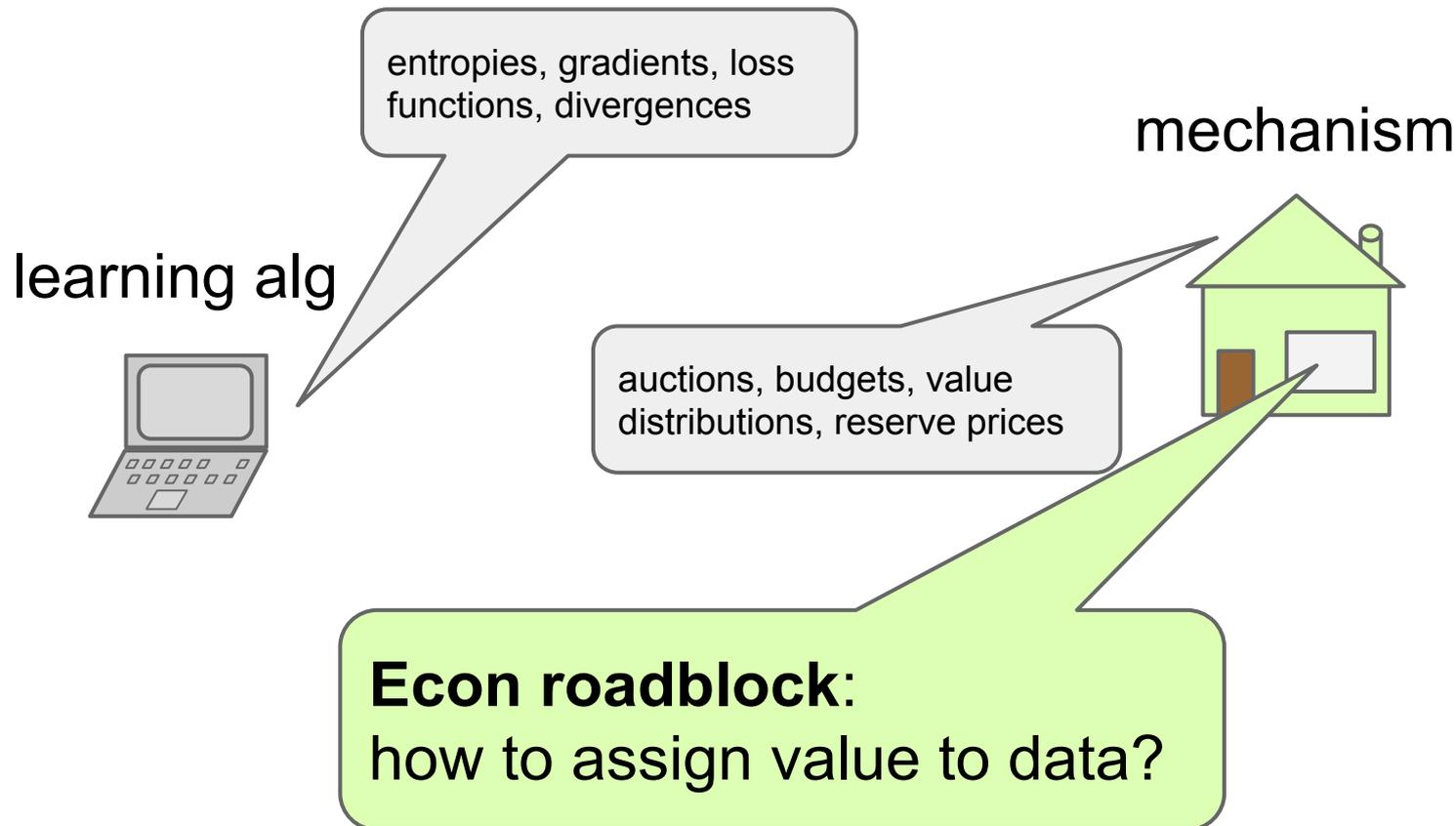
Why is this difficult?

3. Utility (ML) and cost (econ) live in **different worlds**



Why is this difficult?

3. Utility (ML) and cost (econ) live in **different worlds**



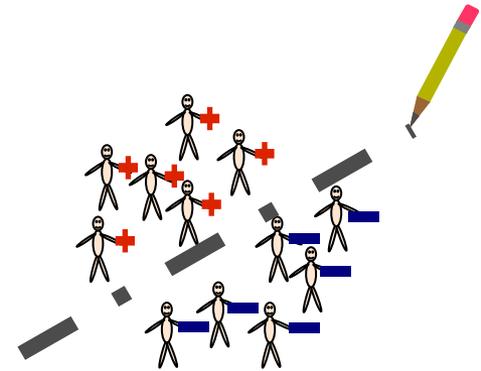
Broad research challenge:

1. How to assign **value** (prices) to pieces of data?
2. How to design **mechanisms** for procuring and learning from data?
3. Develop a **theory** of budget-constrained learning: what is (im)possible to learn given budget B and parameters of the problem?

Outline



1. Overview of literature, our contributions
2. Online learning model/results
3. “Statistical learning” result, conclusion



Related work

How are agents
strategic?



agents cannot
fabricate data,
have costs

this work

principal-agent
style, data
depends on effort

Roth, Schoenebeck 2012

Ligett, Roth 2012

Horel, Ionnadis, Muthukrishnan 2014

Cummings, Ligett, Roth, Wu, Ziani 2015

Cai, Daskalakis, Papadimitriou 2015

Related work

Type of goal  risk/regret bounds

agents cannot fabricate data, have costs

this work

principal-agent style, data depends on effort

minimize variance or related goal

Roth, Schoenebeck 2012

Ligett, Roth 2012

Horel, Ionnadis, Muthukrishnan 2014

Cummings, Ligett, Roth, Wu, Ziani 2015

Cai, Daskalakis, Papadimitriou 2015

Related work

**risk/regret
bounds**

**minimize variance
or related goal**

**agents cannot
fabricate data,
have costs**

this work

Roth, Schoenebeck 2012

Ligett, Roth 2012

Horel, Ionnadis, Muthukrishnan 2014

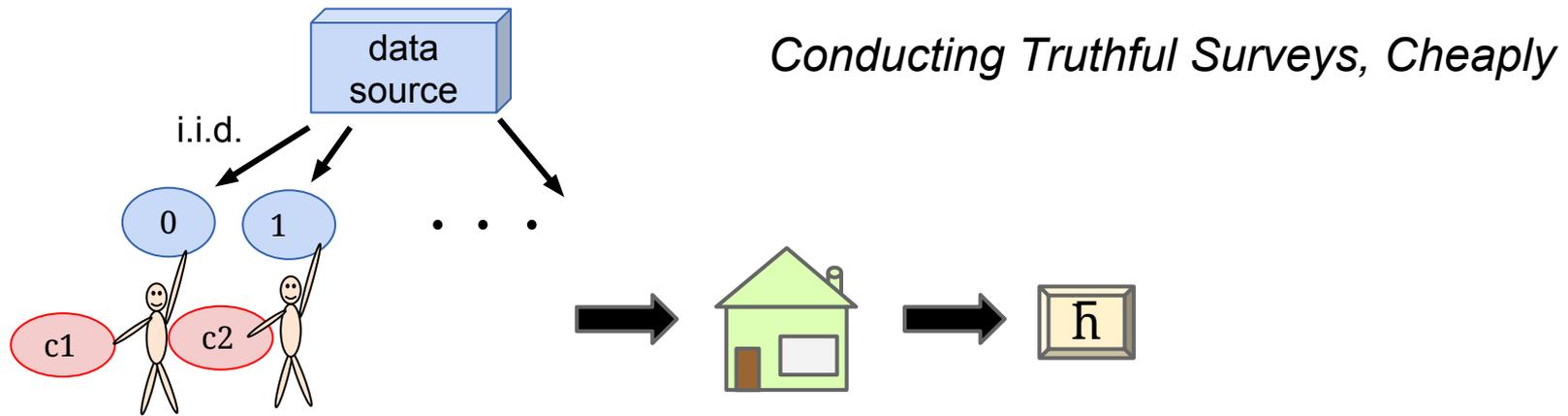
**principal-agent
style, data
depends on effort**

Cummings, Ligett, Roth, Wu, Ziani 2015

Cai, Daskalakis, Papadimitriou 2015

**Waggoner, Frongillo, Abernethy NIPS 2015:
prediction-market style mechanism**

e.g. Roth-Schoenebeck, EC 2012



- Each datapoint is a number. Task is to **estimate the mean**
- **Approach:** offer each agent a price drawn i.i.d.
- **Goal:** minimize the estimate's variance

What we wanted to do differently

1. Prove ML-style risk or regret bounds

Why: ML-style approach: understand error rate as function of budget and problem characteristics.

2. Interface with existing ML algorithms.

Why: understand how value derives from learning alg.
Toward black-box use of learners in mechanisms.

3. Online data arrival

Why: active-learning approach, simpler model

Overview of our contributions

Propose model of online learning with purchased data: T arriving data points and budget B .

Convert any “FTRL” algorithm into a mechanism.

Show regret on order of T / \sqrt{B}
and lower bounds of same order.

Overview of our contributions

Extend model to case where data is drawn i.i.d.
("statistical learning")

Propose model of online learning with purchased data: T arriving data points and budget B .

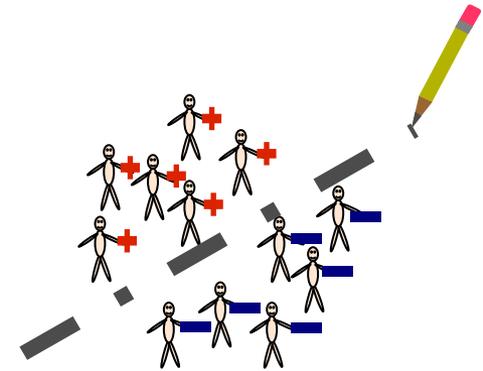
Convert any "FTRL" algorithm into a mechanism.

Show regret on order of T / \sqrt{B}
and lower bounds of same order.

Extend result to "risk" bound on order of $1 / \sqrt{B}$.

Outline

1. Overview of literature,
our contributions



 2. Online learning model/results

3. “Statistical learning” result,
conclusion

Online learning with purchased data

-  a. Review of online learning
- b. Our model: adding \$\$
- c. Deriving our mechanism and results

Standard online learning model

For $t = 1, \dots, T$:

- algorithm posts a hypothesis h_t
- data point z_t arrives
- algorithm sees z_t and updates to h_{t+1}



$$\mathbf{Loss} = \sum_t \ell(h_t, z_t)$$

$$\mathbf{Regret} = \mathbf{Loss} - \sum_t \ell(h^*, z_t) \quad \text{where } h^* \text{ minimizes sum}$$

Follow-the-Regularized-Leader (FTRL)

Assume: loss function is convex and Lipschitz, hypothesis space is Hilbert, etc

Algorithm: $h_t = \operatorname{argmin} \sum_{s < t} \ell(h, z_s) + R(h)/\eta$



Follow-the-Regularized-Leader (FTRL)

Assume: loss function is convex and Lipschitz, hypothesis space is Hilbert, etc

Algorithm: $h_t = \operatorname{argmin} \sum_{s < t} \ell(h, z_s) + R(h)/\eta$



Example 1 (Euclidean norm): $R(h) = \|h\|_2^2$

$\Rightarrow h_t = h_{t-1} - \eta \nabla \ell(h, z_t)$

online gradient descent

Follow-the-Regularized-Leader (FTRL)

Assume: loss function is convex and Lipschitz, hypothesis space is Hilbert, etc

Algorithm: $h_t = \operatorname{argmin} \sum_{s < t} \ell(h, z_s) + R(h)/\eta$



Example 1 (Euclidean norm): $R(h) = \|h\|_2^2$

$\Rightarrow h_t = h_{t-1} - \eta \nabla \ell(h, z_t)$

online gradient descent

Example 2 (negative entropy): $R(h) = \sum_j h^{(j)} \ln(h^{(j)})$.

$\Rightarrow h_t^{(j)} \propto h_{t-1}^{(j)} \exp[\eta \nabla \ell(h_{t-1}, z_t)]$

multiplicative weights

Regret Bound for FTRL

Fact: the regret of FTRL is bounded by O of $1/\eta + \eta \sum_t \Delta_t^2$ where $\Delta_t = \|\nabla \ell(\mathbf{h}_t, \mathbf{z}_t)\|$.



Regret Bound for FTRL

Fact: the regret of FTRL is bounded by O of $1/\eta + \eta \sum_t \Delta_t^2$ where $\Delta_t = \|\nabla \ell(\mathbf{h}_t, \mathbf{z}_t)\|$.



We know $\Delta_t \leq 1$ by assumption, so we can choose $\eta=1/\sqrt{T}$ and get $\text{Regret} \leq O(\sqrt{T})$.

“No regret”: average regret $\rightarrow 0$.

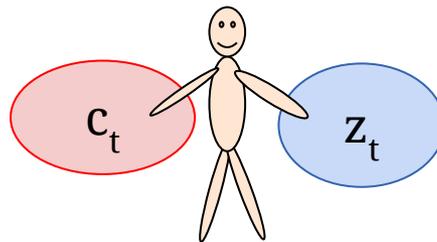
Online learning with purchased data

- a. Review of online learning
-  b. Our model: adding \$\$
- c. Deriving our mechanism and results

First: model of strategic data-holder

Model of agent:

- holds data z_t and cost c_t
- cost is **threshold price**
 - agent agrees to sell data iff price $\geq c_t$
 - interpretations: privacy, transaction cost,

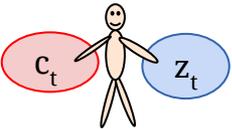


- Assume: all costs ≤ 1

Model of agent-mechanism interaction

- Mechanism posts **menu** of prices offered:

data:	(32,12) 	(20,18) 	(32,12) 
price:	\$0.22	\$0.41	\$0.88

- agent t arrives 
- If $c_t \leq \text{price}(z_t)$, agent **accepts**:
 - agent reveals (z_t, c_t)
 - mechanism pays agent $\text{price}(z_t)$
- Otherwise, agent **rejects**:
 - mechanism learns that agent rejected, pays nothing

Recall: standard online learning model

For $t = 1, \dots, T$:

- algorithm posts a hypothesis h_t
- data point z_t arrives
- algorithm sees z_t and updates to h_{t+1}



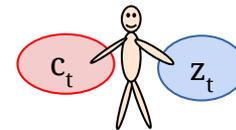
Our model: online learning with \$\$

For $t = 1, \dots, T$:

- mechanism posts a hypothesis h_t *and* a menu of prices



- data point z_t arrives with cost c_t



- If $c_t \leq \text{menu price of } z_t$: mech pays price, learns z_t
- else: mech pays nothing

$$\mathbf{Loss} = \sum_t \ell(h_t, z_t)$$

$$\mathbf{Regret} = \mathbf{Loss} - \sum_t \ell(h^*, z_t)$$

where h^* minimizes sum

Online learning with purchased data

a. Review of online learning

b. Our model: adding \$\$

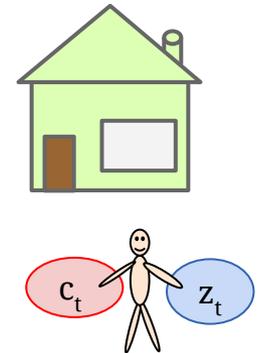
 c. Deriving our mechanism
and results

Start easy

Suppose all costs are 1.

⇒ Determine which data points to sample.

data:	(32,12) ■	(20,18) +	(32,12) +
price:	\$1	\$0	\$0

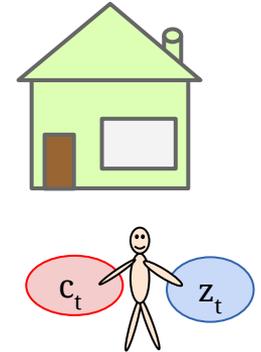


Start easy

Suppose all costs are 1.

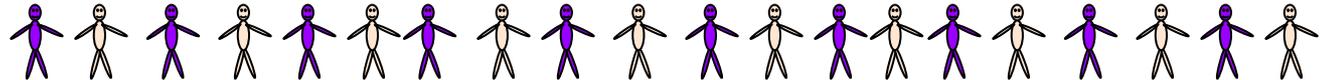
⇒ Determine which data points to sample.

data:	(32,12) 	(20,18) 	(32,12) 
price:	\$1	\$0	\$0

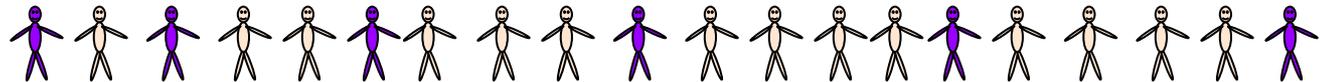


Examples:

- $B = T/2$



- $B = \sqrt{T}$



- $B = \log(T)$



Key idea #1: randomly sample

Can purchase each data point z_t with probability $q_t(z_t)$.

Menu is now **randomly chosen**:

data:	(32,12) 	(20,18) 	(32,12) 
Pr[price=1]:	0.3	0.06	0.41

Key idea #1: randomly sample

Can purchase each data point z_t with probability $q_t(z_t)$.

Menu is now **randomly chosen**:

data:	(32,12) 	(20,18) 	(32,12) 
Pr[price=1]:	0.3	0.06	0.41

Lemma (importance-weighted regret bound):

For any q_t s, the regret of (modified) FTRL is O of

$$1/\eta + \eta \mathbb{E} \left[\sum_t (\Delta_t^2 / q_t) \right]$$

See also: *Importance-Weighted Active Learning*, Beygelzimer et al, ICML 2009.

Result for easy case

Lemma (importance-weighted regret bound):

For any q_t s, the regret of (modified) FTRL is O of

$$1/\eta + \eta \mathbb{E} \left[\sum_t (\Delta_t^2 / q_t) \right]$$

Corollary:

Setting all $q_t = B/T$ and choosing $\eta = \sqrt{B} / T$ yields
regret $\leq T / \sqrt{B}$.

“No data, no regret”:

average amount of data $\rightarrow 0$ and average regret $\rightarrow 0$.

Result for easy case

Lemma (importance-weighted regret bound):

For any q_t s, the regret of (modified) FTRL is O of

$$1/\eta + \eta \mathbb{E} \left[\sum_t (\Delta_t^2 / q_t) \right]$$

Corollary:

Setting all $q_t = B/T$ and choosing $\eta = \sqrt{B} / T$ yields
regret $\leq T / \sqrt{B}$.

Theorem:

This is tight.

(Predict a repeated coin toss whose bias is either $1+1/\sqrt{B}$ or $1-1/\sqrt{B}$)

Now a bit harder....

Costs can be arbitrary, but agents are **nonstrategic**: they will accept payment exactly c_t .

At each time step, randomly choose which (data, cost) pairs to purchase.

data,cost:	(32,12)  , $c=0.3$	(20,18)  , $c=0.8$
Pr[purchase]:	0.12	0.08

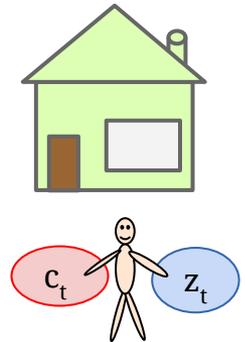
Question: how to set probabilities of purchase q_t ?

Key idea #2: sample proportional to...

Imagine we knew the arrivals in advance.

Optimization problem:

$$\begin{array}{ll} \text{minimize} & \sum_t (\Delta_t^2 / q_t) \\ \text{s.t.} & \sum_t q_t c_t \leq B \\ & q_t \leq 1. \end{array}$$



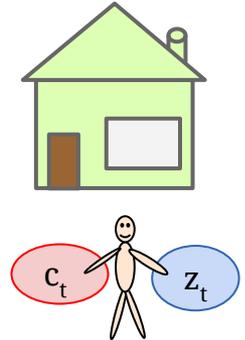
Solution: $q_t = \Delta_t / K \sqrt{c_t}$ (K a normalizing constant).

Key idea #2: sample proportional to...

Imagine we knew the arrivals in advance.

Optimization problem:

$$\begin{array}{ll} \text{minimize} & \sum_t (\Delta_t^2 / q_t) \\ \text{s.t.} & \sum_t q_t c_t \leq B \\ & q_t \leq 1. \end{array}$$



Solution: $q_t = \Delta_t / K \sqrt{c_t}$ (K a normalizing constant).

The point: only need advance knowledge of K to implement the “optimal” sampling strategy!

Turns out: $K = \gamma T / B$, where $\gamma \in [0,1]$ (discuss later)

Result for this “at-cost” setting

Theorem:

Given rough advance estimate of γ , can achieve
$$\text{regret} \leq \gamma T / \sqrt{B}$$

Theorem:

This is tight (in a reasonable sense).

(Same bad instance, but with “useless” free data points sprinkled in.)

Implication: γ is capturing the “difficulty of the problem”.

Discussion

$$\begin{aligned}\gamma &= (1/T) \sum_t \Delta_t \sqrt{c_t} \\ &= \text{average } \text{sqrt}(\text{difficulty} * \text{cost}).\end{aligned}$$

Discussion

$$\begin{aligned}\gamma &= (1/T) \sum_t \Delta_t \sqrt{c_t} \\ &= \text{average } \text{sqrt}(\text{difficulty} * \text{cost}).\end{aligned}$$

Example simplified corollary:

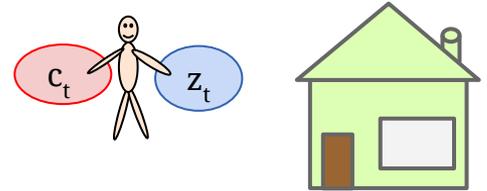
Given rough advance estimate of avg cost μ ,

$$\text{regret} \leq \sqrt{\mu} T / \sqrt{B}$$

- Low avg cost \Rightarrow low regret
- Low avg difficulty \Rightarrow low regret
- **good correlations** \Rightarrow low regret

Finally, the “full” problem.

Now agents are **strategic**
and we must **post prices**.

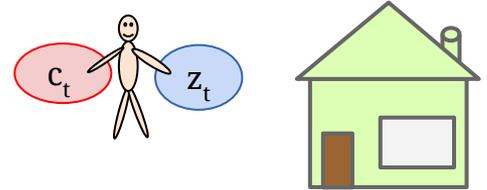


Recall: had sampling probability $q_t = \Delta_t / K \sqrt{c_t}$.

But: we don't know c_t .

Finally, the “full” problem.

Now agents are **strategic**
and we must **post prices**.



Recall: had sampling probability $q_t = \Delta_t / K \sqrt{c_t}$.

But: we don't know c_t .

Key idea #3: randomly draw price from the distribution s.t.
 $\Pr[\text{price} \geq c_t] = \Delta_t / K \sqrt{c_t}$.

\Rightarrow achieve the “right” probability for *every* c_t simultaneously!

Description of final mechanism

Input: estimate of γ

At each time t :

- post hypothesis $h_t \leftarrow \text{FTRL}$
- for each data point z_t , compute $\Delta_t = \|\nabla \ell(h_t, z_t)\|$ and post random price from distribution
- If arriving agent accepts, send “re-weighted” $z_t \rightarrow \text{FTRL}$

Main result for online learning setting

Theorem:

Given rough advance estimate of γ , can achieve
$$\text{regret} \leq \sqrt{\gamma} T / \sqrt{B}$$

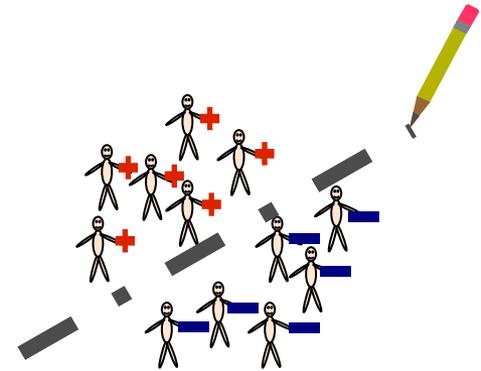
Theorem (recall):

No mechanism for the easier, “at-cost” setting can beat
$$\text{regret} \leq \gamma T / \sqrt{B}$$

Note: lost a $\sqrt{\gamma}$ factor compared to easier setting,
due to paying our posted price rather than the agent’s cost.
 (“cost of strategic behavior”)

Outline

1. Overview of literature, our contributions
2. Online learning model/results
-  3. “Statistical learning” result, conclusion



Recalling contributions

Extend model to case where data is drawn i.i.d.
("statistical learning")

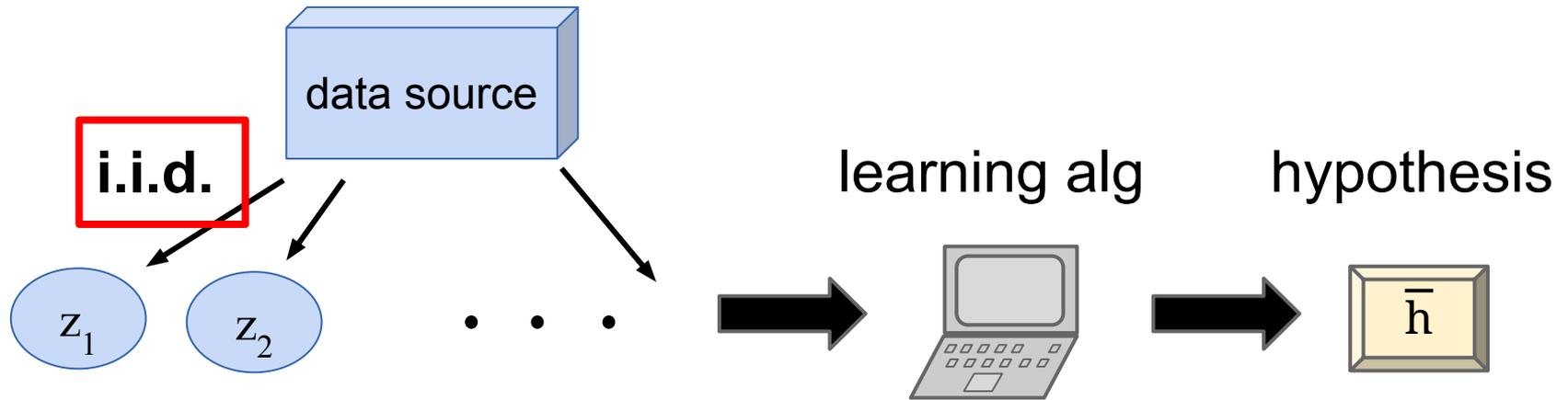
Propose model of online learning with purchased data: T arriving data points and budget B .

Convert any "FTRL" algorithm into a mechanism.

Show regret on order of T / \sqrt{B}
and lower bounds of same order.

Extend result to "risk" bound on order of $1 / \sqrt{B}$.

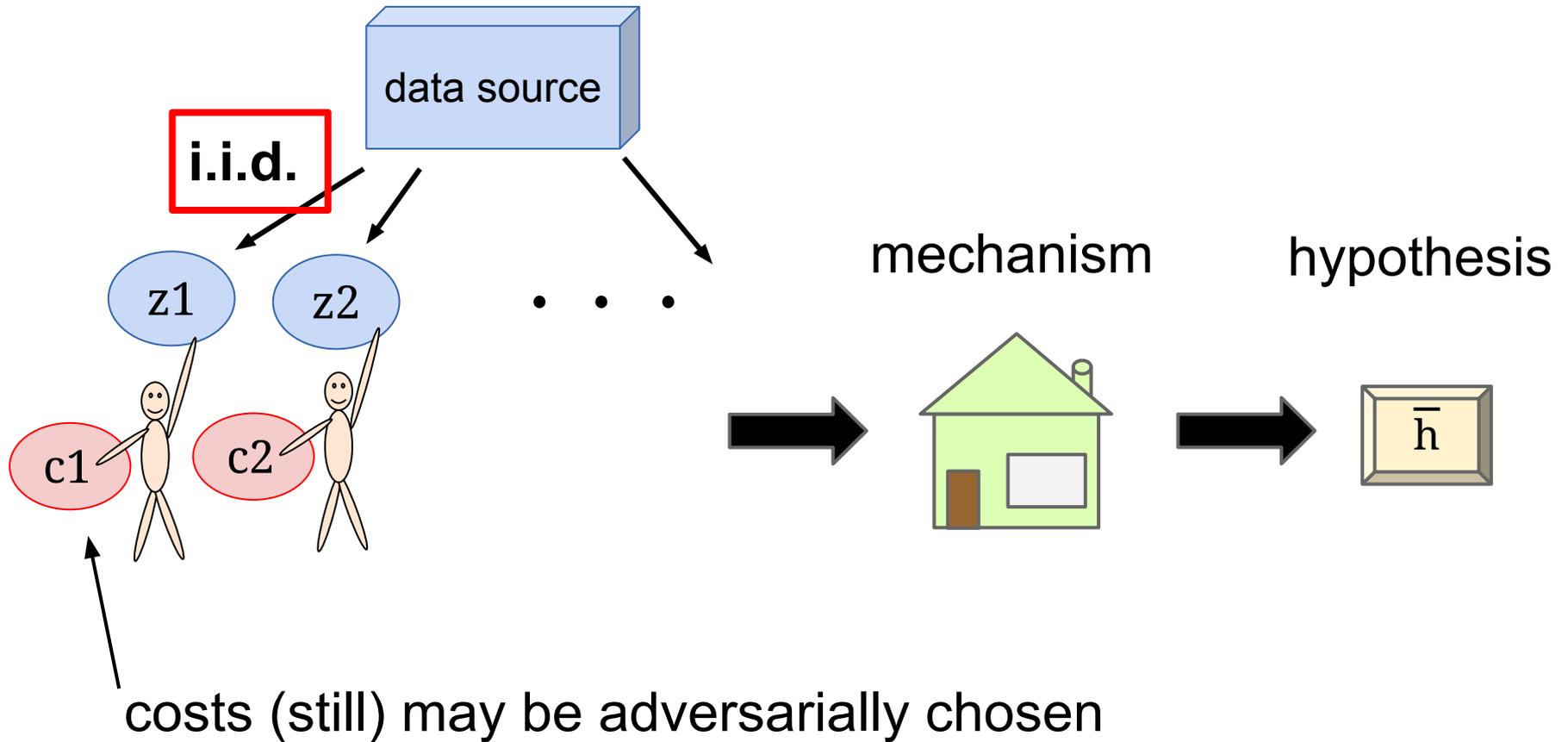
Classic statistical learning model



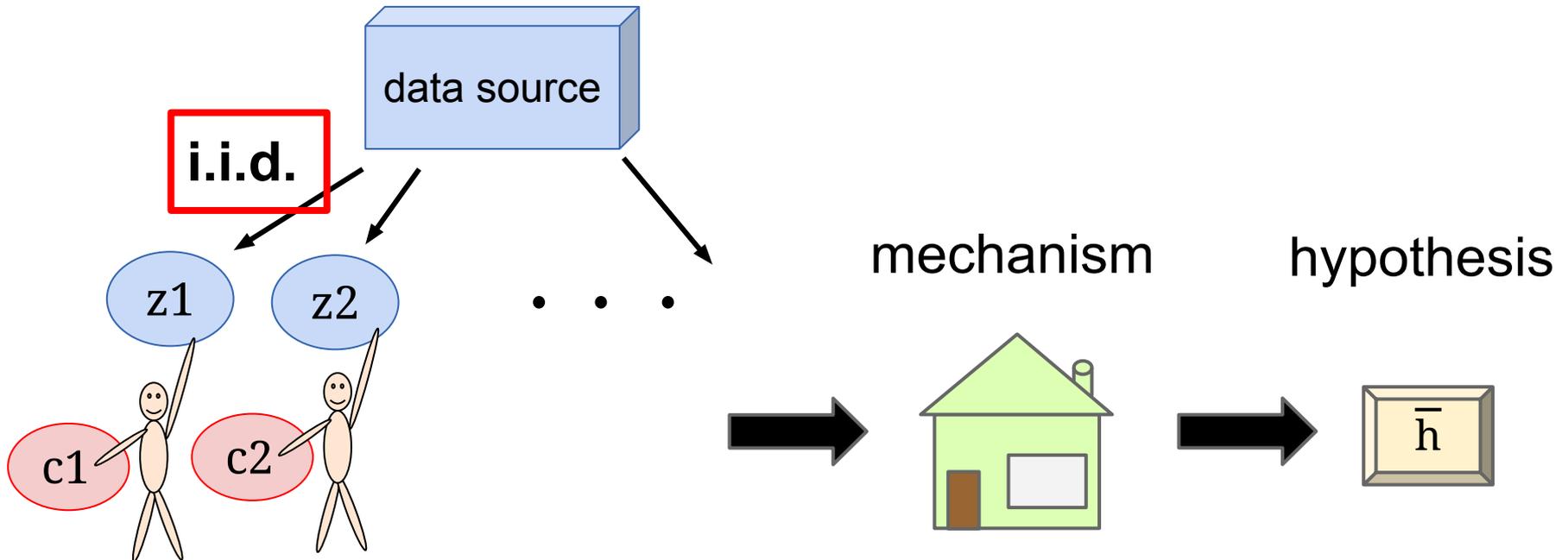
For classification:

$$\mathbf{E} \text{ loss}(\bar{h}) \leq \mathbf{E} \text{ loss}(h^*) + O\left(\sqrt{\frac{\text{VC-dim}}{T}}\right)$$

Our statistical learning model



Our statistical learning model

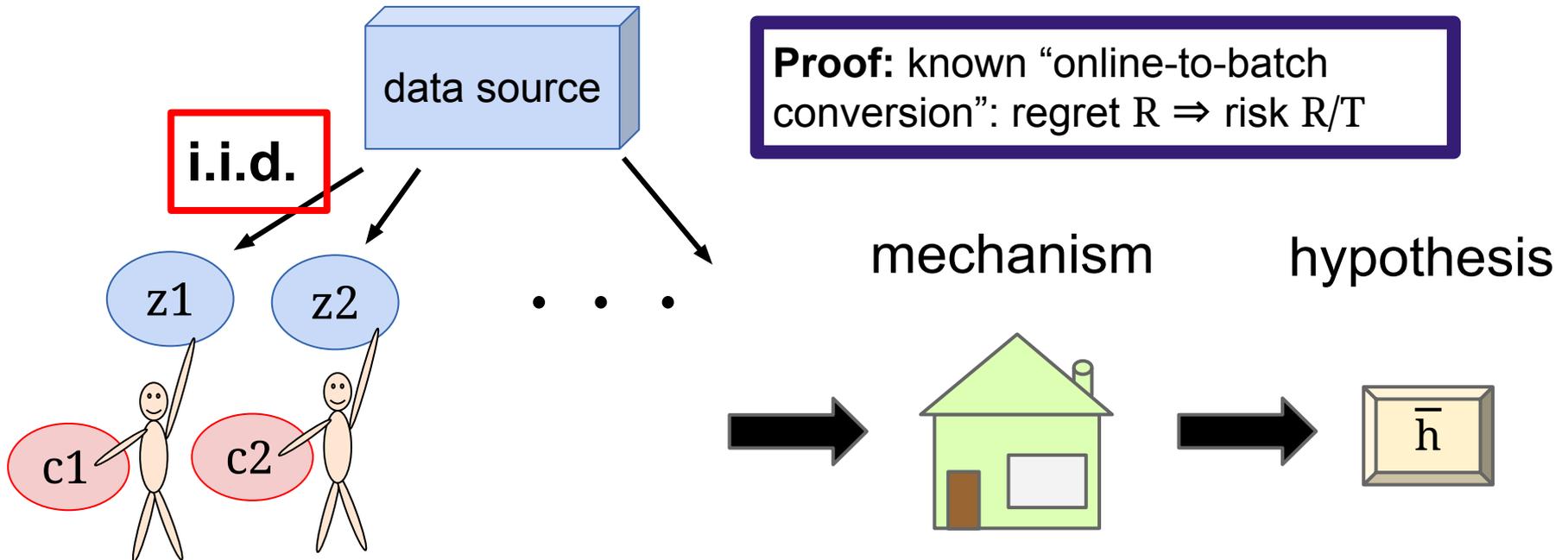


Theorem:

Given rough advance estimate of γ , can achieve

$$\mathbf{E} \text{ loss}(h) \leq \mathbf{E} \text{ loss}(h^*) + O\left(\sqrt{\frac{\gamma}{B}}\right)$$

Our statistical learning model

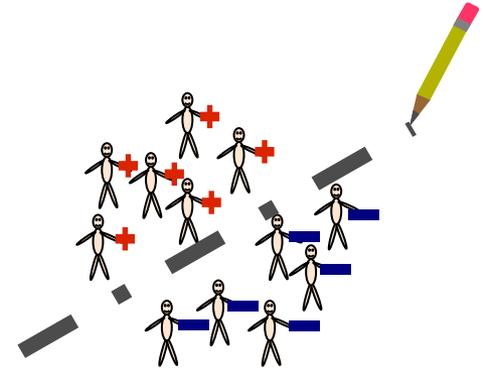


Theorem:

Given rough advance estimate of γ , can achieve

$$\mathbf{E} \text{ loss}(h) \leq \mathbf{E} \text{ loss}(h^*) + O\left(\sqrt{\frac{\gamma}{B}}\right)$$

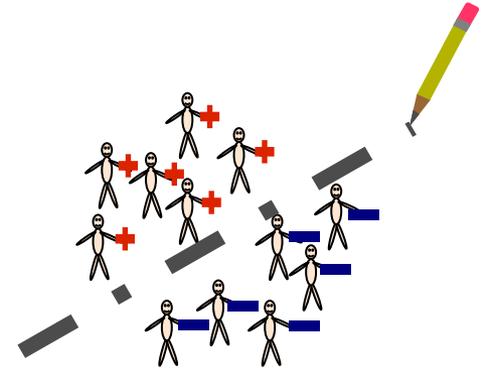
Summary



Model:

- online arrival of agents
- post prices to procure data
- adversarial costs and data
(online learning setting)
- adversarial costs, i.i.d. data
(statistical learning setting)

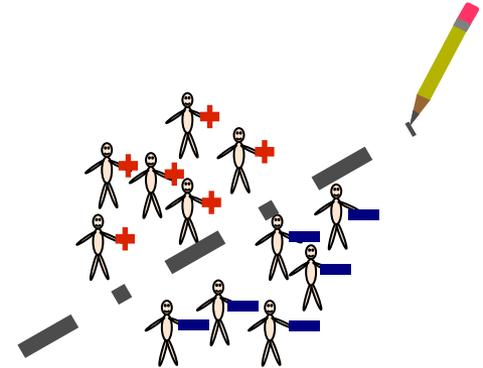
Summary



Results:

- upper/lower bounds on regret
(online learning setting)
- upper bound on risk
(statistical learning setting)

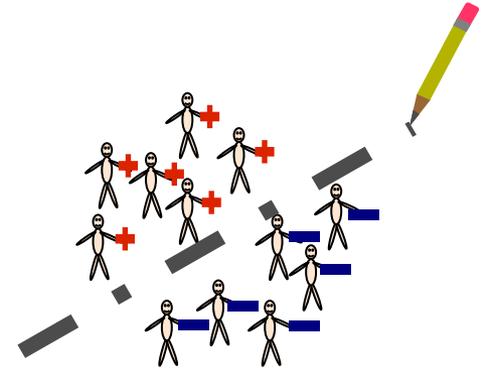
Summary



Big picture:

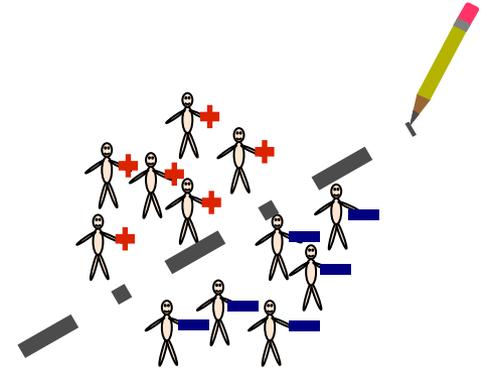
- design mechanisms to interface with existing learning algs
- prove ML-style bounds: risk and regret
- toward a “theory of the learnable...on a budget”

Future work



- Improve bounds (!)
- Propose “universal quantity” to replace γ in bounds (analogue of VC-dimension?)
- Explore models for purchasing data

Future work



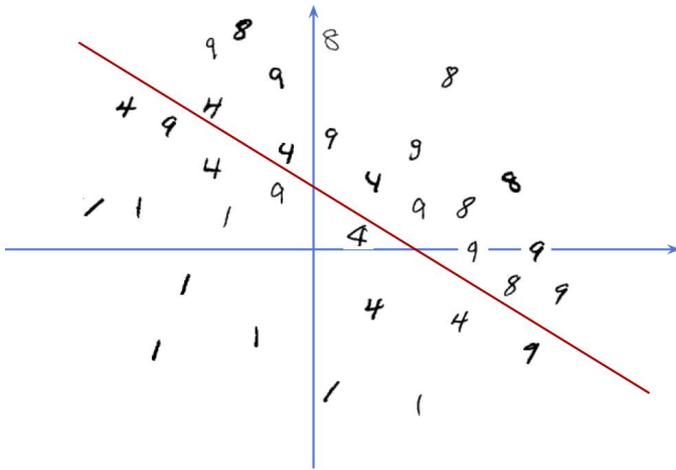
- Improve bounds (!)
- Propose “universal quantity” to replace γ in bounds (analogue of VC-dimension?)
- Explore models for purchasing data

Thanks!

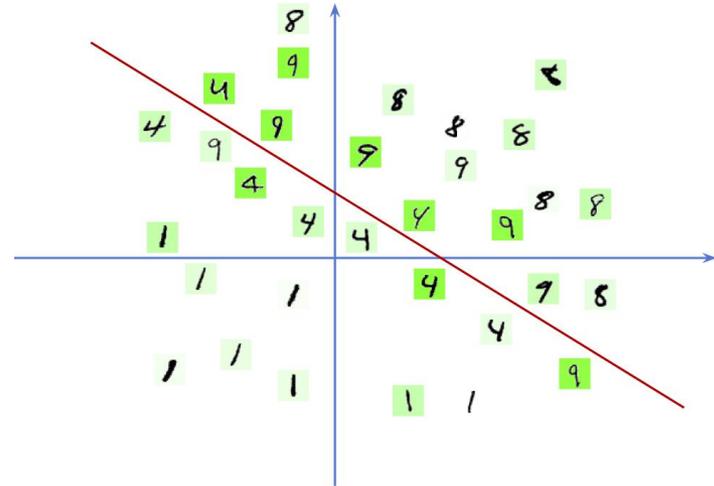
Additional slides

Simulation results

MNIST dataset -- handwritten digit classification



Toy problem:
classify (1 or 4)
vs (9 or 8)



Brighter green =
higher cost

Simulation results

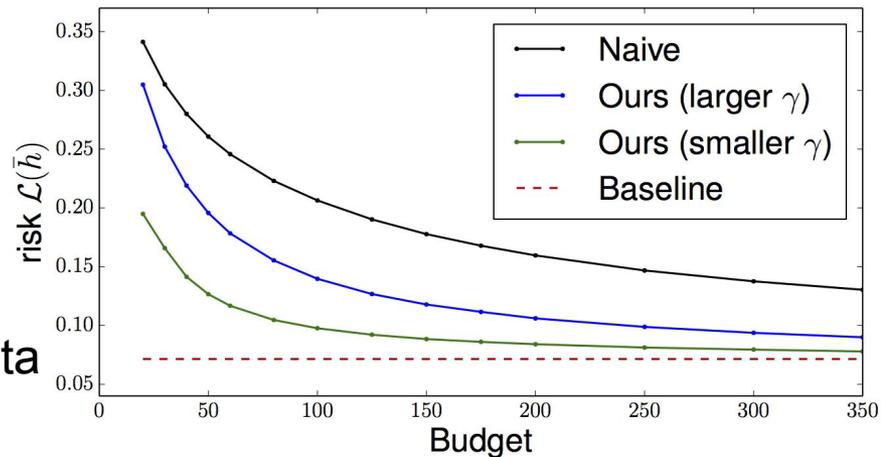
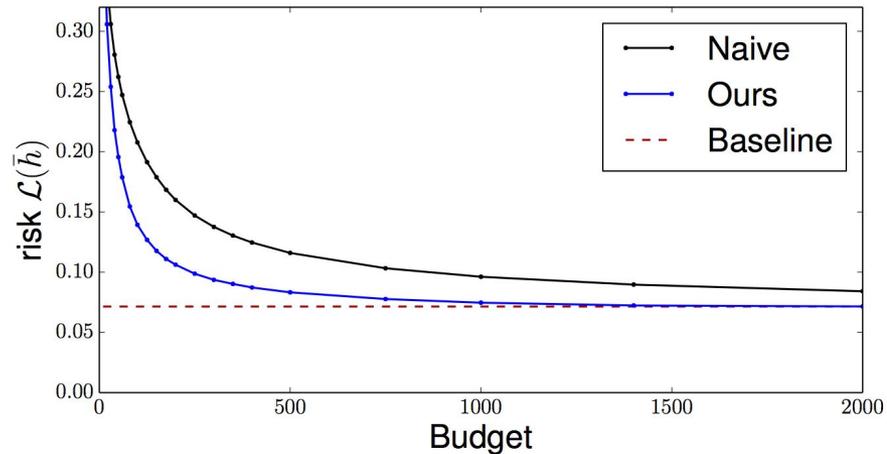
- $T = 8503$
- train on half, test on half
- Alg: Online Gradient Descent

Naive: pay 1 until budget is exhausted, then run alg

Baseline: run alg on all data points (no budget)

Large γ : bad correlations

Small γ : independent cost/data



Pricing distribution

