Active Regression for Cyberphysical Systems

PIs: Baosen Zhang, Ramesh Johari

University of Washington, Stanford University

zhangbao@uw.edu, ramesh.johari@stanford.edu

November 3, 2015

Humans and CPS

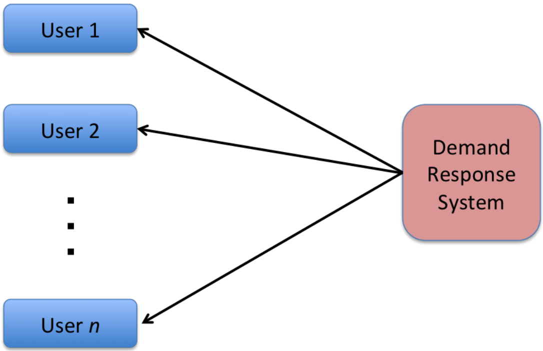
Emphasis of our project:

How do we design cyberphysical systems that effectively learn about their users, and optimize system behavior accordingly?

This poster: *active regression* as a vehicle to learn about users.

Motivating example: Demand response

In a demand response system, we learn *users' preferences* by experimentation; e.g., current demand response programs will often run randomized experiments to learn preferences.



Question: How can we be efficient in choice of which users to include in an experiment?

Motivating example: Demand response

We abstract the problem as follows:

1. A utility decides to run a demand response program.
2. Successive users arrive, and a choice must be made about whether to include them in the trial.
3. The goal is to learn a model that maps user *features* to the expected *outcome* (e.g., energy savings).

The formal problem is to choose users to include in an *online* fashion, based on their features.

The Problem – Formal Definition I

Data Generating Source

We assume the response of a user is given by a linear model $Y = X\beta + \epsilon$, where $X \in \mathbf{R}^d$, $Y \in \mathbf{R}$, $\beta \in \mathbf{R}^d$ and $\epsilon \sim \mathcal{N}(0, \sigma^2) \in \mathbf{R}$.

Also, $X_1, \ldots, X_n \sim D = \mathcal{N}(0, \Sigma)$; these are the *features*. We assume β and σ^2 are unknown.

We consider both the case where Σ is known and where it is unknown.

The Problem – Formal Definition II

Online Setting

We see $X_1, \ldots, X_n$ sequentially, and we have to choose k out of them in an online fashion. After selecting X_i, we get to see Y_i.

Let $S = \{X_{(1)}, \ldots, X_{(k)}\}$ be the set of selected observations. Finally, we compute our estimate β_S by using those observations.

The Problem – Formal Definition III

Goal

Our goal is to estimate β.

More concretely, we want to find β_S to minimize

$$\mathbf{MSE}(\beta_S) = \mathbf{E}[\|\beta_S - \beta\|^2] = \sigma^2 \ \mathbf{E}[\mathrm{Tr}((X_S^T X_S)^{-1})],$$

where the expectation is taken wrt the training sequence of n observations, and the algorithm / selection rule for S.

For passive learning, $\mathbf{MSE}(\beta_S) = \sigma^2 \ \frac{d}{k-d-1} \geq \sigma^2 \ \frac{d}{k}$.

The Problem – A few remarks

Minimizing the MSE for β_S is equivalent to minimizing the expected trace of the inverse Fisher information matrix. But (because we assumed a linear model) Fisher Information does not depend on β! So no need to look at the y's.

Want to minimize $\mathbf{E}[\mathrm{Tr}((X_S^T X_S)^{-1})]$.

The Solution – Intuition

We want large feature vectors leading to orthogonal columns.

But columns of $\mathbf{X}$ live in $\mathbf{R}^k$, and there are d of them, with $d \ll k$.

So they will be close to orthogonal. Hence, we focus on **large** norms.

Idea: set a threshold Γ, and choose user i iff $\|x_i\| \geq \Gamma$.

Try to capture largest feature vectors: $\mathbf{P}(\|x_i\| \geq \Gamma) = k/n$.

The Solution – Threshold Algorithm

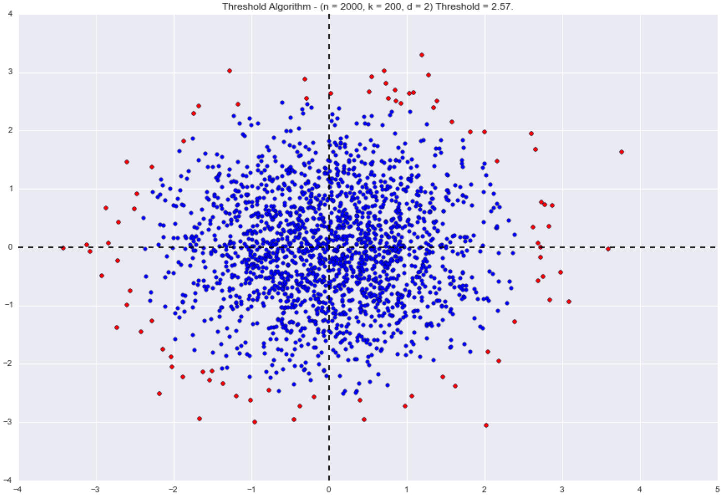
After some analysis, we think a very simple algorithm works well:

Algorithm 1 Norm-based online active linear regression.

Set $\Gamma = C\sqrt{d + 2\log(n/k)}$ and $S = \emptyset$.
for time $1 \leq t \leq n$ **do**
  Observe X_t, estimate $\widehat{\Sigma}_t$, compute $\overline{X_t} = \widehat{\Sigma}_t^{-1/2} X_t$.
  if $\|\overline{X_t}\| > \Gamma$ **then**
    Choose X_t: $S = S \cup X_t$.
    if $|S| = k$ **then**
      Break.
    end if
  end if
end for

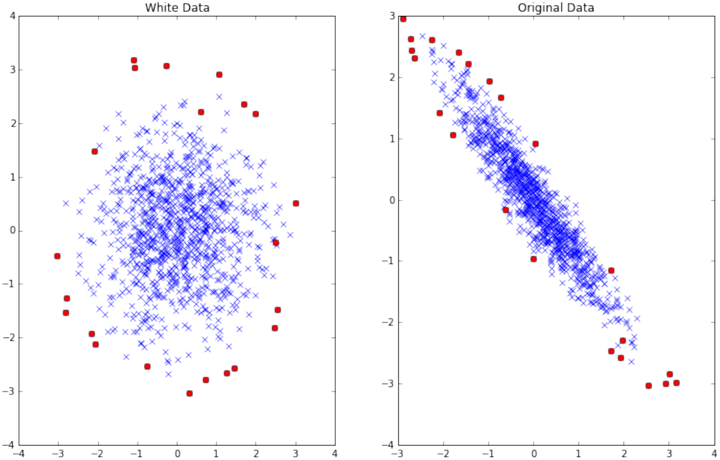
The Solution – Threshold Algorithm

The *selected* observations (in red) usually look like:



The Solution – Threshold Algorithm

And if Σ is not the identity, after whitening:



Conjecture - Active Learning Gain

We think an algorithm like the one described in the previous slide yields

$$\text{active learning error} \approx \left(\frac{1}{1 + \frac{2}{d} \log \frac{n}{k}} \right) \text{passive learning error.}$$

How good is this?

Theorem - Active Learning Gain

For the case where Σ is known:

Theorem

Let X be a $k \times d$ matrix with k observations in $\mathbf{R}^d$ chosen by the thresholded-algorithm with $T = \sqrt{d + 2\log n/k}$. Let $\phi > 0$, then there exist $C_1, C_2 > 0$, positive constants (that may depend on d, k, n), such that $-C_1 \geq \log(1 - 1/d)$, and such that with probability at least $1 - d\ e^{-C_1 k\phi - C_2 k}$

$$Tr\left((X^T X)^{-1}\right) \leq \frac{d}{k\left(1 + 2\ \frac{\log n/k}{d}\right)(1-\phi)}. \tag{1}$$

Open Questions and Next Steps

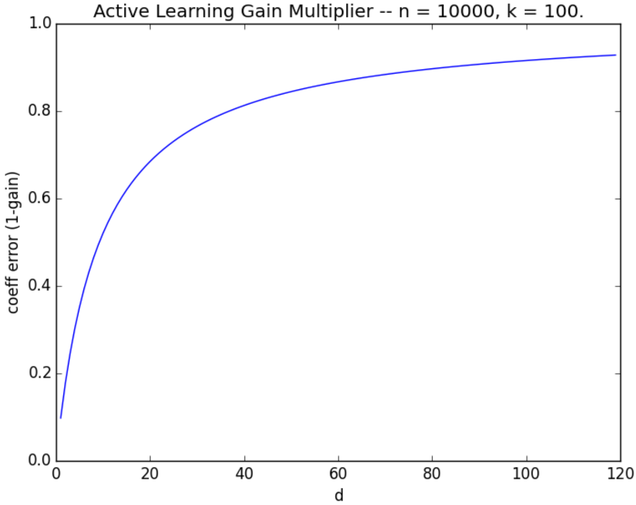
1. Extend analysis to settings where Σ is unknown, we need to compute an initial estimate $\hat{\Sigma}$ (Secretary Problem type of algorithms).
2. Extend analysis to other families of distributions for $\mathbf{X}$ (subgaussian, subexponential distributions...).
3. Extend analysis to settings where $d \geq k$, and **regularization** is needed. In these cases, estimating Σ could be difficult.

Open Questions and Next Steps II

1. If the underlying data source is *not* linear, what's the gain with respect to the best linear approximation with *passive* learning?
2. Apply same analysis to **Logistic** Regression; quite a different setting. Fisher Info depends on β. Need to use β_t to choose observation $t+1$.
3. Latent feature subspaces in high-dimensions.
4. More simulations, and real experiments.

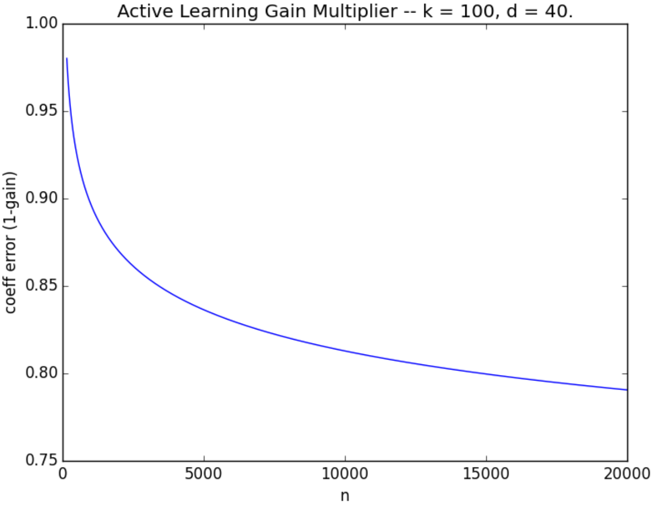
Active Learning Gain

As a function of d, for fixed $n = 10000$ and $k = 100$:



Active Learning Gain

As a function of n, for fixed $k = 100$ and $d = 40$:



Active Learning Gain

As a function of k, for fixed $n = 10000$ and $d = 50$: