Orthogonal Array Based Big Data Subsampling

Hongquan Xu

University of California, Los Angeles

Joint work with

Lin Wang, Jake Kramer and Weng Kee Wong

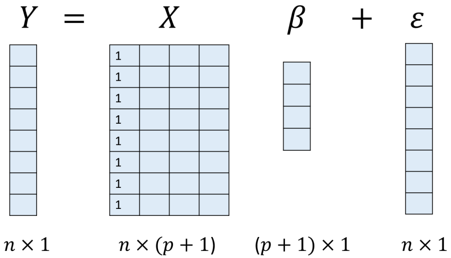
Motivation

- Data are big (sometimes redundant)
- Analyzing the full data may be computationally unfeasible
- Storing all of the data may not be possible
- There are a lot of circumstances under which X is big while the label (or response) Y is expensive to obtain

Big X	Small Y
Patients' records	Performance of a new medicine
Images as visual stimuli	Brain response

Linear Regression Setup

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \varepsilon$$



Question:

1. If the budget only allows k responses (labels), $k \ll n$, choose which k to label?

2. When all responses are available, to accelerate the computation, how to choose a subsample of size $k \ll n$?

Leverage Subsampling

A subsampling method consists of

- sampling probabilities π_i, $i = 1, \ldots, n$, $\sum_i^n \pi_i = 1$
- a weighted estimator $\hat{\beta}_s = (X_s^T W X_s)^{-1} X_s^T W Y_s$, where X_s is the subsample taken from X and $W = diag(w_1, \ldots, w_k)$ is a weight matrix

1. Uniform sampling: $\pi_i = 1/n$, $w_i = 1$
2. Leveraging: $\pi_i = h_{ii}/(p+1)$, $w_i = 1/\pi_i$, where

$$h_{ii} = (X(X^T X)^{-1} X^T)_{ii}$$

(Drineas et al., 2006; Ma et al., 2015)

Information-based Subsampling

The OLS for a subsample X_s is $\hat{\beta}_s = (X_s^T X_s)^{-1} X_s^T Y_s$

$$\mathrm{E}(\hat{\beta}_s) = \beta, \quad \mathrm{Var}(\hat{\beta}_s) = \sigma^2 (X_s^T X_s)^{-1}$$

- The Fisher information matrix for β with subdata is

$$M_s = X_s^T X_s$$

- D-optimality: to find X_s with k points that maximizes $\det(M_s)$

Available approach: IBOSS (Wang H., Yang, and Stufken, 2018 JASA)
Include data points with extreme (largest and smallest) covariate values

Orthogonal Array-based Subsampling

Lemma

Suppose each covariate is scaled to $[-1, 1]$. For a subsample X_s of size k,

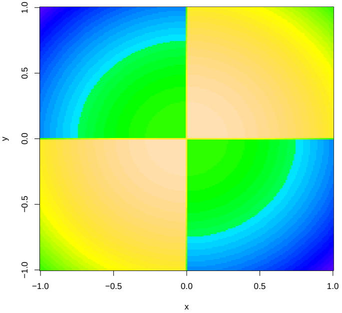
$$\det(M_s) \leq k^{p+1},$$

and the equality holds (D-optimal) if and only if X_s forms a two-level OA with levels from $\{-1, 1\}$ and strength $t \geq 2$.

A Measure of "Similarity"

For $x, y \in [-1, 1]$, define

$$\delta(x,y) = \begin{cases} 2 - (x^2 + y^2)/2, & \text{if } sign(x) = sign(y); \\ 1 - (x^2 + y^2)/2, & \text{otherwise.} \end{cases}$$



$$\begin{cases} 1 \le \delta(x,y) \le 2, & \text{if } sign(x) = sign(y); \\ 0 \le \delta(x,y) \le 1, & \text{otherwise.} \end{cases}$$

$$\delta(-1,1) = \delta(1,-1) = 0$$
$$\delta(-1,-1) = \delta(1,1) = 1$$

J-optimality

For two data points $x_i = (x_{i1}, \ldots, x_{ip})$ and $x_j = (x_{j1}, \ldots, x_{jp})$, let

$$\delta(x_i, x_j) = \sum_{l=1}^{p} \delta(x_{il}, x_{jl})$$

and define the J-optimality criterion (Xu 2002, *Technometrics*) as

$$J(X_s) = \sum_{1 \le i < j \le k} [\delta(x_i, x_j)]^2 .$$

Theorem

For any k-point subsample X_s over $[-1, 1]^p$,

$$J(X_s) \ge [k^2 p(p+1) - 4kp^2]/8,$$

with equality if and only if X_s forms an OA with strength 2.

Question: How to find X_s that minimizes $J(X_s)$?

Algorithm: Select and eliminate points simultaneously

Step 0. Given $n \times p$ matrix X, scale each variable to [-1,1].

Step 1. Find a point that is farthest to the origin. Include it as X_s and let $i = 1$.

Step 2. For each $x \in X$, compute the J-score

$$J(x, X_s) = \sum_{x_s \in X_s} \delta(x, x_s)^2.$$

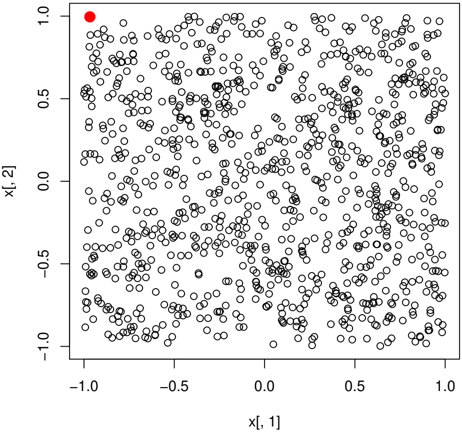
Step 3. Find $x^* \in X$ that minimizes $J(x, X_s)$ and add x^* to X_s.

Step 4. Keep $t = \lfloor n/i \rfloor$ points in X with t smallest $J(x, X_s)$ values. Remove x^* and other points from X.

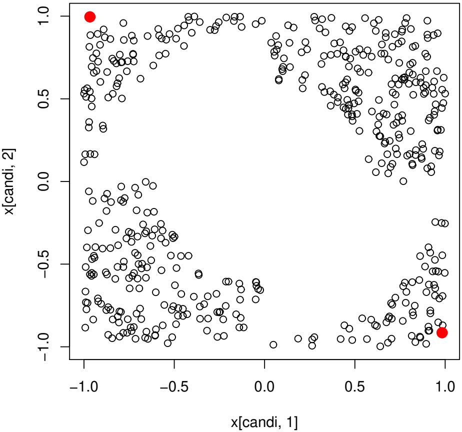
Step 5. Increase i by 1 and repeat Steps 2–4 until X_s contains k points.

Note: The complexity is $O(np \log(k))$ as $\sum_{i=1}^{k}(1/i) = O(\log(k))$.

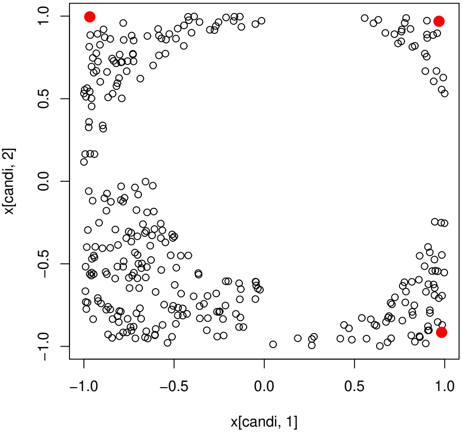
Algorithm



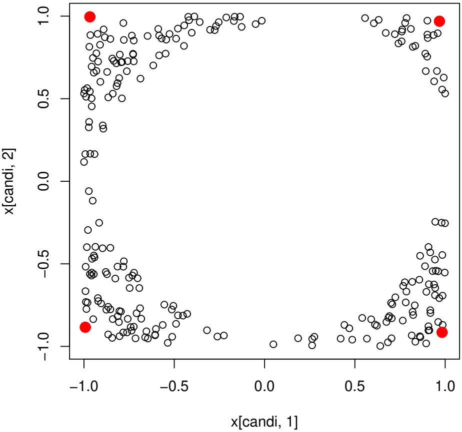
Algorithm



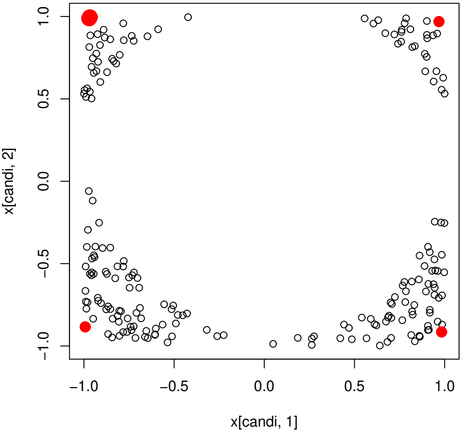
Algorithm



Algorithm



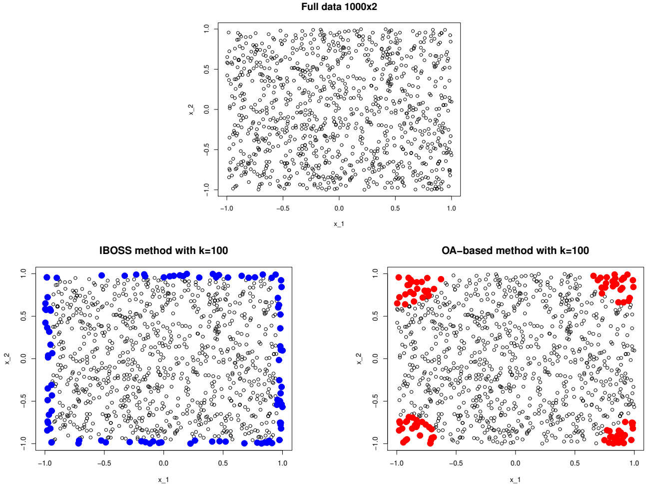
Algorithm



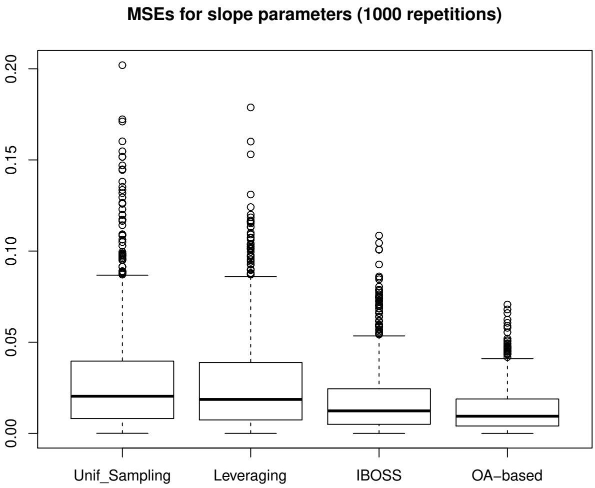
A Toy Simulation

Setup: $x_1, x_2 \overset{\text{iid}}{\sim} \text{Unif}[-1,1]$, $n = 1000$, $p = 2$, $k = 100$, $\varepsilon \sim N(0,1)$

$$y = 1 + 2x_1 + 2x_2 + \varepsilon$$



Comparison


MSEs for slope parameters (1000 repetitions)
0.00
0.05
0.10
0.15
0.20
Unif_Sampling
Leveraging
IBOSS
OA–based

Theoretical Properties

Assume that the covariates $x_1, \ldots, x_p$ are i.i.d. with a uniform distribution in $[a, b]^p$, and we are considering the model

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \varepsilon.$$

Theorem

The OA-based subsampling minimizes MSE, i.e., the sum of the variances of coefficient estimations, almost surely as $n \to \infty$.

Model with Interactions

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \beta_{12} x_1 x_2 + \cdots + \beta_{(p-1)p} x_{p-1} x_p + \varepsilon$$

Define the J-optimality criterion as

$$J_4(X_s) = \sum_{1 \le i < j \le k} [\delta(x_i, x_j)]^4 .$$

Assume that the covariates $x_1, \ldots, x_p$ are i.i.d. with a uniform distribution in $[a, b]^p$.

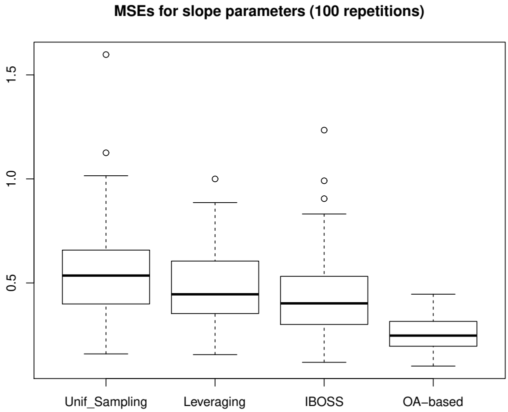
Theorem

The OA-based subsampling minimizes MSE, i.e., the sum of variances of coefficient estimations, almost surely as $n \to \infty$.

Simulation

Setup: $x_1, \ldots, x_p \overset{\text{iid}}{\sim} \text{Unif}[-1, 1]$, $n = 10^4$, $p = 10$, $k = 128$, $\varepsilon \sim N(0, 3^2)$

$$y = 1 + x_1 + \cdots + x_{10} + x_1 x_2 + \cdots + x_9 x_{10} + \varepsilon$$



Summary

- Propose a new framework for subsampling: OA-based subsampling
- Computational complexity: $O(np\log(k))$
- Minimize $J(X_s)$ for a linear model without interactions, attains optimality for coefficient estimation for large n
- Minimize $J_4(X_s)$ for a linear model with interactions, attains optimality for coefficient estimation for large n
- More efficient than leverage sampling and IBOSS