# Optimal design of experiments and model-based survey sampling in Big Data

Laura Deldossi[1] Chiara Tommasi[2]

[1]Department of Statistical Sciences,
Università Cattolica del Sacro Cuore, Milan, Italy

[2]Department of Economics, Management and Quantitative Methods
University of Milan, Italy

Enbis 2019
Budapest, September 2-4, 2019

# Motivation of the work

Advances in technology have brought us the ability to collect, transfer and store large data set.
The availability of huge quantity of data is a great challenge nowadays BUT:

- Enormous computational effort is required to analyze an entire Big Dataset.
- Very often Big Dataset contains redundant data.
- From a practical point of view, some inferential conclusions may be ineffective with large samples

# Subsampling from Big Datasets

## A possible solution

To select and analyse only a sample from the Big Dataset

Some papers on this topic:

- Ma and Sun (2015), *Leveraging for big data regression*, Wiley Interdisciplinary Reviews: Computational Statistics.
- Wang, Yang, Stufken (2018), *Information-Based Optimal Subdata selection for Big Data Linear Regression*, Journal of the American Statistical Association.
- Wang, Zhu, Ma (2018), *Optimal Subsampling for Large Sample Logistic Regression*, Journal of the American Statistical Association.
- Drovandi, Holmes, McGree, Mengersen, Richardson, Ryan (2018), *Principles of experimental design for big data analysis*, Statistical Sciences.
- Campbell and Broderick (2019), *Automated scalable Bayesian inference via Hilbert coresets*, Journal of Machine Learning Research.

# OUR PROPOSAL

## CONTEXT

- A Big Dataset is conceived as a finite population.
- We are interested in making inference about the parameters of the model (the so called *super-population model*) the Big Dataset has been generated from.

# OUR PROPOSAL

## CONTEXT

- A Big Dataset is conceived as a finite population.
- We are interested in making inference about the parameters of the model (the so called *super-population model*) the Big Dataset has been generated from.

## GOAL

To select a sample which contains the majority of information about the unknown parameters of the super-population model **using the optimal design theory**.

# OUR PROPOSAL

## CONTEXT

- A Big Dataset is conceived as a finite population.
- We are interested in making inference about the parameters of the model (the so called *super-population model*) the Big Dataset has been generated from.

## GOAL

To select a sample which contains the majority of information about the unknown parameters of the super-population model **using the optimal design theory**.

**N.B.** We consider the **model-based survey** approach in place of the **design based survey** approach adopted by other authors like Ma and Sun (2015).

# Overview

1. DOE: notation and framework
2. The super-population model and Big Dataset
3. The Optimal Design Based (ODB) sample
4. Performance of our approach in comparison with other sampling technics (SRS, PPS, IBOSS)
5. Conclusion and future work

# DOE: framework and notation

- $\boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^p$: **multivariate experimental condition** chosen by the experimenter.
- $Y = Y(\boldsymbol{x})$: **response** variable observed at the experimental point $\boldsymbol{x}$.

- Responses and experimental conditions are related (at least approximatively) through a **regression model**:

$$Y_i \quad = \quad \eta(\boldsymbol{x}_i, \boldsymbol{\theta}) + \varepsilon_i \cong \boldsymbol{f}(\boldsymbol{x}_i)^T \boldsymbol{\theta} + \varepsilon_i = \theta_0 + \sum_{j=1}^{m-1} f_j(\boldsymbol{x}_i)\, \theta_j + \varepsilon_i,$$

$$\mathrm{E}(\varepsilon_i) = 0, \quad \mathrm{Var}(\varepsilon_i) = \sigma^2, \quad \mathrm{Cov}(\varepsilon_i, \varepsilon_l) = 0.$$

- **Inferential goal:** to provide a precise estimation of $\boldsymbol{\theta}$.

# Continuous designs

- **A continuous design** $\xi$ is a discrete probability measure on $\mathcal{X}$ with a finite number of support points:

$$\xi = \left\{ \begin{array}{ccc} \boldsymbol{x}_1 & \cdots & \boldsymbol{x}_k \\ \omega_1 & \cdots & \omega_k \end{array} \right\}, \quad 0 \leq \omega_j \leq 1, \ \sum_{j=1}^{k} \omega_j = 1.$$

- **Information matrix** for an approximate design $\xi$:

$$M(\xi) = \sum_{j=1}^{k} \boldsymbol{f}(\boldsymbol{x}_j) \boldsymbol{f}(\boldsymbol{x}_j)^T \, \omega_j.$$

- An **optimality criterion function** $\Phi(M)$ is a **concave** function of the information matrix which summarizes the inferential goal: precise estimation of the parameters. A design $\xi_\Phi^*$ is called $\Phi$–optimal iff

$$\xi_\Phi^* = \arg\max_{\xi} \Phi[M(\xi)]$$

# Optimality criterion function and efficiency

- Common criteria for a precise estimation of $\theta$ are:
  1. **D-optimality:** $\Phi_D[M(\xi)] = |M(\xi)|$
     A $\mathrm{D}$-optimum design minimizes the **generalized variance** of $\theta$
  2. **A-optimality:** $\Phi_A[M(\xi)] = -\mathrm{Tr}\left[M(\xi)^{-1}\right]$
     An $\mathrm{A}$-optimum design minimizes the **total variation** of $\theta$

- The **Efficiency of a design** $\xi$ is a measure of the goodness of a design $\xi$ with respect to the $\Phi$-optimum design $\xi^*_\Phi$:

$$0 \le \mathrm{Eff}(\xi) = \frac{\Phi[M(\xi)]}{\Phi[M(\xi^*_\Phi)]} \le 1$$

**Interpretation:** If $\mathrm{Eff}(\xi) = 60\%$ then using $\xi^*_\Phi$ we save 40% of the observations to get the same precision in parameter estimation.

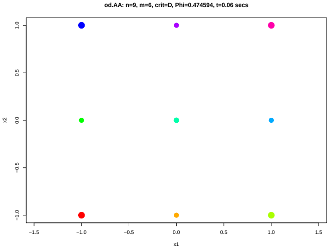
# Example: D-optimal design

Consider two explanatory variables, $x_1$ and $x_2$, and the following quadratic model with the interaction term:

$$\begin{aligned} Y_i &= \theta_0 + \theta_1\, x_{1i} + \theta_2\, x_{2i} + \theta_3\, x_{1i}^2 + \theta_4\, x_{2i}^2 + \theta_5\, x_{1i}\cdot x_{2i} + \varepsilon_i, \\ &= \boldsymbol{f}(\boldsymbol{x}_i)^T\boldsymbol{\theta} + \varepsilon_i, \quad \boldsymbol{\theta} = (\theta_0, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5)^T, \;\; \boldsymbol{x} = (x_1, x_2)^T \\ &\qquad\qquad\qquad \boldsymbol{f}(\boldsymbol{x}) = (1,\, x_1,\, x_2,\, x_1^2,\, x_2^2,\, x_1\cdot x_2)^T \end{aligned}$$

The D-optimal design is:

$$\xi_D^* = \left\{ \begin{array}{ccccccccc} (-1,-1) & (0,-1) & (1,-1) & (-1,0) & (0,0) & (1,0) & (-1,1) & (0,1) & (1,1) \\ .146 & .080 & .146 & .080 & .096 & .080 & .146 & .080 & .146 \end{array} \right\}$$



# The super-population model and the Big Dataset

- The Big Dataset is assumed to be generated by a super-population model:

$$\mathbf{y}_U \cong \mathbf{F}_U \theta + \varepsilon_U,$$

- $\theta = (\theta_0, \ldots, \theta_m)$ is a $(m+1) \times 1$ vector of unknown parameters of interest,
- $\varepsilon_U = (\varepsilon_1, \ldots, \varepsilon_N)$ is a vector of homoschedastic independent errors such that $\mathrm{E}(\varepsilon_i) = 0$ and $\mathrm{Var}(\varepsilon_i) = \sigma^2$

$$\mathbf{F}_U = \begin{bmatrix} \mathbf{f}(\mathbf{x}_1)^T \\ \vdots \\ \mathbf{f}(\mathbf{x}_N)^T \end{bmatrix}$$

is the design matrix where $\mathbf{f}(\mathbf{x})^{\mathbf{T}} = (f_1(\mathbf{x}), \ldots, f_m(\mathbf{x}))^T$ contains $m$ linear independent known functions.

# Big Data

- Let the Big Dataset be a $N \times (1+p)$ matrix with $p << N$:
  1. First column: $N$ observations for a response variable $Y$;
  2. Remaining $N \times p$ matrix (denoted by $\mathcal{X}$): $N$ **observed** values for an auxiliary variable $X \in \mathbb{R}^{\mathrm{p}}$.

$$\left[\begin{array}{c|ccc} y_1 & x_{11} & \cdots & x_{1p} \\ \vdots & \vdots & \ddots & \vdots \\ y_N & x_{N1} & \cdots & x_{Np} \end{array}\right] = \left[\begin{array}{cc} y_1 & \boldsymbol{x}_1^T \\ \vdots & \vdots \\ y_N & \boldsymbol{x}_N^T \end{array}\right]$$

- “Experimental conditions” as rows of the Big Data set

### “Nature” as an experimenter

If Nature had been a “wise” experimenter then it would have chosen the $N$ values for the explanatory vector variable $\boldsymbol{X}$ according to an optimality criterion.

# A measure for the quality of the Big Data

- Given a super-population model it is always possible to compute the (continuous) Φ-optimum design:

$$\xi^*_\Phi = \begin{Bmatrix} x^*_1 & \cdots & x^*_j & \cdots & x^*_k \\ \omega^*_1 & \cdots & \omega^*_j & \cdots & \omega^*_k \end{Bmatrix}.$$

- A wise Nature would have generated $N\omega^*_j$ responses at $x^*_j$ (with $j = 1, \ldots, k$) providing the **“ideal” Big Data**.
- Let $\mathbf{F}^*_\mathbf{U}$ denote the related “ideal” design matrix.
- $\mathbf{M}(\mathbf{x_U}) = \mathbf{F}^\mathbf{T}_\mathbf{U}\mathbf{F}_\mathbf{U}/N$ measures the per-unit information contained in the Big Dataset

# A measure for the quality of the Big Data

- Given a super-population model it is always possible to compute the (continuous) Φ-optimum design:

$$\xi^*_\Phi = \left\{ \begin{array}{ccccc} x^*_1 & \cdots & x^*_j & \cdots & x^*_k \\ \omega^*_1 & \cdots & \omega^*_j & \cdots & \omega^*_k \end{array} \right\}.$$

- A wise Nature would have generated $N\omega^*_j$ responses at $x^*_j$ (with $j = 1, \ldots, k$) providing the **"ideal" Big Data**.
- Let $\mathbf{F}^*_\mathbf{U}$ denote the related "ideal" design matrix.
- $\mathbf{M}(\mathbf{x_U}) = \mathbf{F}^\mathbf{T}_\mathbf{U}\mathbf{F}_\mathbf{U}/N$ measures the per-unit information contained in the Big Dataset

## Quality of the Big Data

Φ-efficiency is a measure of the quality of the Big Data:

$$0 \leq \mathrm{Eff}_\Phi \mathbf{M}(\mathbf{x_U}) = \frac{\Phi[\mathbf{F}^\mathbf{T}_\mathbf{U}\mathbf{F}_\mathbf{U}]}{\Phi[\mathbf{F}^{*\mathbf{T}}_\mathbf{U}\mathbf{F}^*_\mathbf{U}]} \leq 1$$

# To extract the most informative observations

To analyse an entire Big Data could be computationally difficult. It makes sense to use only the most informative observations.

- $\mathbf{F_s}$ denotes the design matrix of a sample $s$ of $n$ observations selected from the Big Data; $\boldsymbol{M}_s = \mathbf{F_s^T F_s}/n$ is the **sample information matrix** which measures the per-unit information contained in $s$.
- If we select all the $\binom{N}{n}$ samples of size $n$ from the Big Data, we can compute all the corresponding

$$\mathrm{Eff}_{\Phi}[\mathbf{M_s}] = \frac{n \cdot \Phi[\mathbf{F_s^T F_s}]}{N \cdot \Phi[\mathbf{F_U^{*T} F_U^{*}}]}$$

.

- Let $s^*$ be sample of size $n$ with the **largest** value of $\mathrm{Eff}_{\Phi}[\mathbf{M_s}]$.

# To extract the most informative observations

To analyse an entire Big Data could be computationally difficult. It makes sense to use only the most informative observations.

- $\mathbf{F_s}$ denotes the design matrix of a sample $s$ of $n$ observations selected from the Big Data; $\boldsymbol{M}_s = \mathbf{F_s^T F_s}/n$ is the **sample information matrix** which measures the per-unit information contained in $s$.
- If we select all the $\binom{N}{n}$ samples of size $n$ from the Big Data, we can compute all the corresponding

$$\mathrm{Eff}_{\Phi}[\mathbf{M_s}] = \frac{n \cdot \Phi[\mathbf{F_s^T F_s}]}{N \cdot \Phi[\mathbf{F_U^{*T} F_U^*}]}$$

.

- Let $s^*$ be sample of size $n$ with the **largest** value of $\mathrm{Eff}_{\Phi}[\mathbf{M_s}]$.

- **Drawback:** $s^*$ cannot be computed if $N$ and $n$ are large.

# To extract the most informative observations

To analyse an entire Big Data could be computationally difficult. It makes sense to use only the most informative observations.

- $\mathbf{F_s}$ denotes the design matrix of a sample $s$ of $n$ observations selected from the Big Data; $\boldsymbol{M}_s = \mathbf{F_s^T F_s}/n$ is the **sample information matrix** which measures the per-unit information contained in $s$.
- If we select all the $\binom{N}{n}$ samples of size $n$ from the Big Data, we can compute all the corresponding

$$\mathrm{Eff}_\Phi[\mathbf{M_s}] = \frac{n \cdot \Phi[\mathbf{F_s^T F_s}]}{N \cdot \Phi[\mathbf{F_U^{*T} F_U^*}]}$$

.

- Let $s^*$ be sample of size $n$ with the **largest** value of $\mathrm{Eff}_\Phi[\mathbf{M_s}]$.

- **Drawback:** $s^*$ cannot be computed if $N$ and $n$ are large.
- We propose **an optimal design strategy** to approximate $s^*$.

# ODB: Optimal design based sample

The (continuous) Φ-optimal design

$$\xi_\Phi^* = \left\{ \begin{array}{ccccc} x_1^* & \cdots & x_j^* & \cdots & x_k^* \\ \omega_1^* & \cdots & \omega_j^* & \cdots & \omega_k^* \end{array} \right\}.$$

suggests the following sampling rule:

### Optimal design based (ODB) sample

To select from the Big Data the $n\omega_j^*$ rows of $\mathbf{F_U}$ which are closest to $\mathbf{f}(\mathbf{x_j^*})^T$ for $j = 1, \ldots, k$.

- Given a super-population model the Φ-optimal design is easily found (we have applied the R package *Optimal design* by Harman and Filova (2016)).
- As a measure of closeness we have applied Euclidean distance but any other distance can be used.
- If $n\omega_j^*$ is not integer than a suitable rounding-off rule can be applied (see Pulkesheim and Rieder, 1992).

# Another possibility: Exchange Algorithm

## Exchange alghoritm sample

An approximation of $s^*$ is provided by the **exchange algorithm**; see for instance Mitchell and Miller (1970) and Wynn (1972).

- An initial sample $s_0 = \{x_1, \ldots, x_n\}$ is chosen at random from $\mathcal{X}$, i.e. the $N$ rows of the Big Data.
- $s_0$ is improved by adding that point $x_{n+1} \in \mathcal{X}$ which **most improves** the $\Phi$-criterion, followed by removing that point from $\{x_1, \ldots, x_n, x_{n+1}\}$, which gives the **smallest reduction** in the $\Phi$-criterion.
- This add/remove procedure is continued until it converges, with the same point being added and then removed.

# Comparison with three other sampling strategies

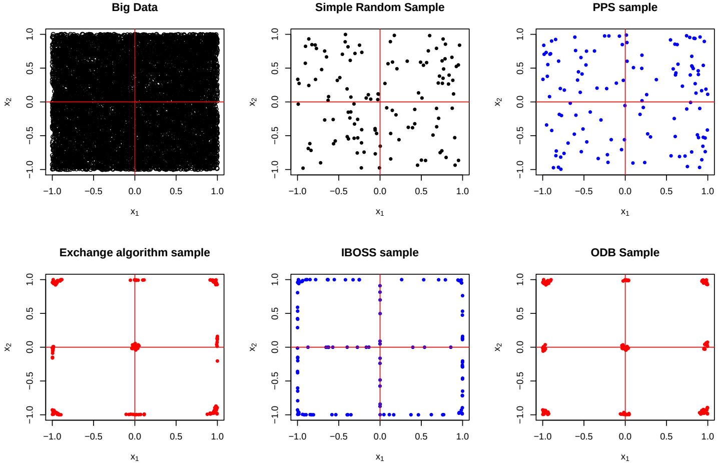
- The simple random sampling without replacement (SRS);
- The probability proportional to size (PPS) sampling with selection probabilities given by

$$p_i = \frac{f(x_i)^T(\mathbf{F}_\mathbf{U}^\mathbf{T}\mathbf{F}_\mathbf{U})^{-1}f(x_i)}{m+1}, \quad i = 1, \cdots, N$$

  to select more frequently the rows (the units) with largest prediction variance (see also Ma and Sun, 2015).
- The Information Based Optimal Subdata Selection (IBOSS) algorithm: a novel approach to data selection recently proposed by Wang, Yang and Stufken (2018).
  It is an algorithm motivated by D-optimality according to which large variation in covariates is more informative and results in better parameter estimation.

# N=10000, n=120; Linear model $f(x)^T = (1, x_1, x_2, x_1 \cdot x_2, x_1^2, x_2^2)$

$$\xi_D^* = \left\{ \begin{array}{ccccccccc} (-1,-1) & (0,-1) & (1,-1) & (-1,0) & (0,0) & (1,0) & (-1,1) & (0,1) & (1,1) \\ .146 & .080 & .146 & .080 & .096 & .080 & .146 & .080 & .146 \end{array} \right\};\ x \sim U_2(-1,1)$$



Big Data

Simple Random Sample

PPS sample

Exchange algorithm sample

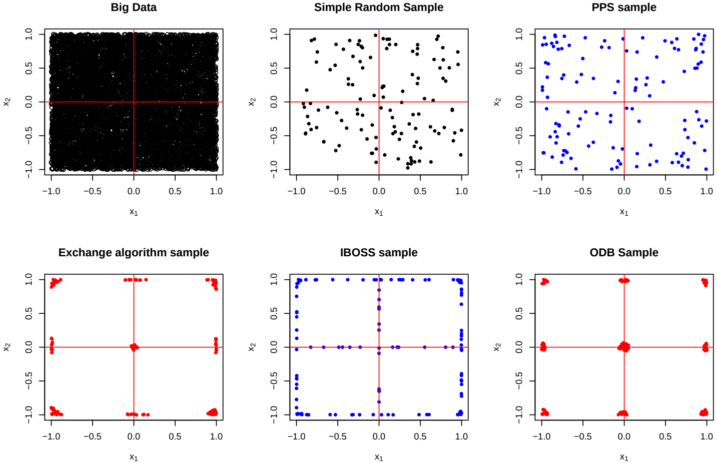
IBOSS sample

ODB Sample

$\text{Eff}_D(\text{Big Data}) = .456$, $\text{Eff}_D(\text{SRS}) = 461$, $\text{Eff}_D(\text{PPS}) = .556$,
$\text{Eff}_D(\text{Exchange}) = .920$, $\text{Eff}_D(\text{IBOSS}) = .833$, $\text{Eff}_D(\text{ODB}) = .914$

# N=10000, n=120; Linear model $f(x)^T=(1, x_1, x_2, x_1 \cdot x_2, x_1^2, x_2^2)$

$$\xi_A^* = \left\{ \begin{array}{ccccccccc} (-1,-1) & (0,-1) & (1,-1) & (-1,0) & (0,0) & (1,0) & (-1,1) & (0,1) & (1,1) \\ .094 & .098 & .094 & .098 & .233 & .098 & .094 & .098 & .094 \end{array} \right\}; x \sim U_2(-1,1)$$



Big Data

Simple Random Sample

PPS sample

Exchange algorithm sample

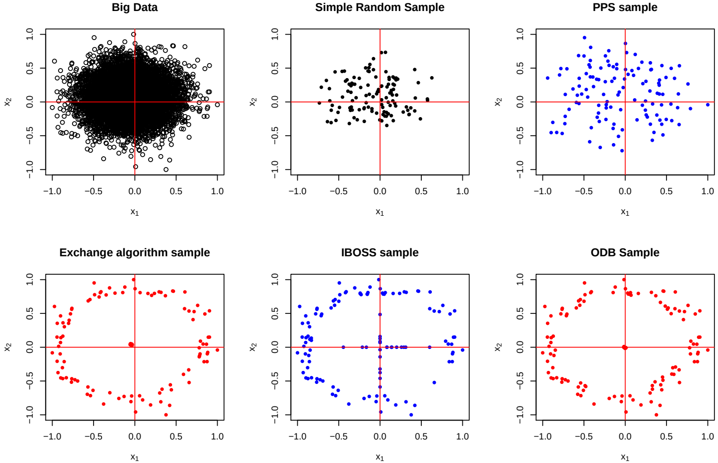
IBOSS sample

ODB Sample

$\text{Eff}_A(\text{Big Data}) = .441$, $\text{Eff}_A(\text{SRS}) = .438$, $\text{Eff}_A(\text{PPS}) = .514$,
$\text{Eff}_A(\text{Exchange}) = .805$, $\text{Eff}_A(\text{IBOSS}) = .741$, $\text{Eff}_A(\text{ODB}) = .921$

# N=10000, n=120; Linear model $f(x)^T=(1, x_1, x_2, x_1 \cdot x_2, x_1^2, x_2^2)$

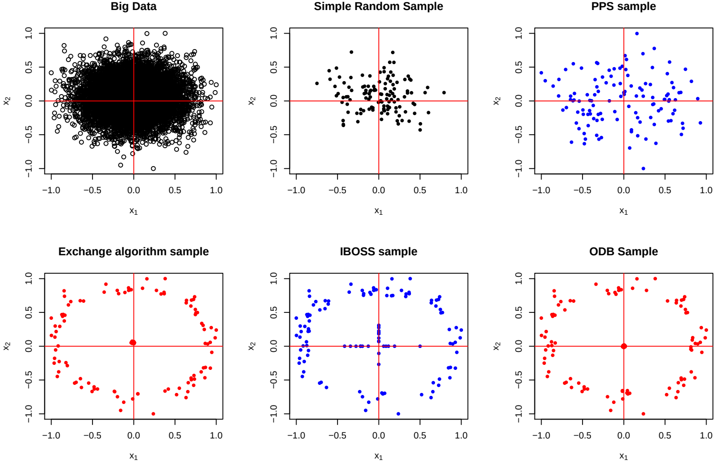
$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \sim N_2\left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma = \begin{bmatrix} 16 & 0 \\ 0 & 16 \end{bmatrix} \right), \quad x_j = \frac{z_j - \left(z_{j(1)} + z_{j(N)}\right)/2}{\left(z_{j(N)} - z_{j(1)}\right)/2}$$



$\text{Eff}_D(\text{Big Data}) = .071$, $\text{Eff}_D(\text{SRS}) = .072$, $\text{Eff}_D(\text{PPS}) = .193$,
$\text{Eff}_D(\text{Exchange}) = .394$, $\text{Eff}_D(\text{IBOSS}) = .376$, $\text{Eff}_D(\text{ODB}) = .378$

# N=10000, n=120; Linear model $f(x)^T = (1, x_1, x_2, x_1 \cdot x_2, x_1^2, x_2^2)$

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \sim N_2\left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \Sigma = \begin{bmatrix} 16 & 0 \\ 0 & 16 \end{bmatrix} \right), \; x_j = \frac{z_j - \left(z_{j(1)} + z_{j(N)}\right)/2}{\left(z_{j(N)} - z_{j(1)}\right)/2}$$



Big Data

Simple Random Sample

PPS sample

Exchange algorithm sample

IBOSS sample

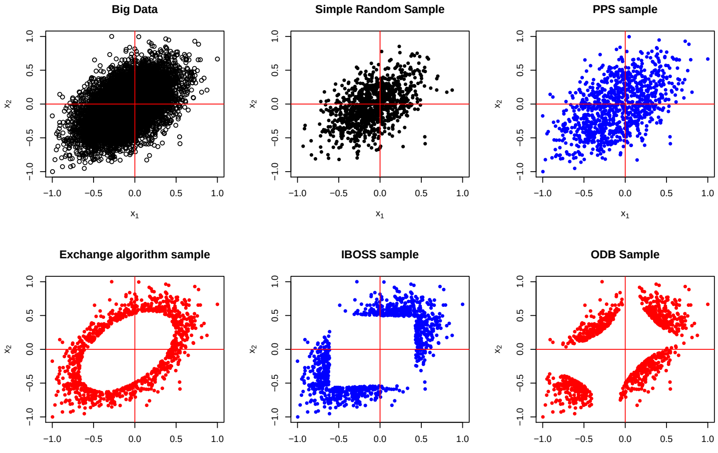
ODB Sample

$\mathrm{Eff}_A(\text{Big Data}) = .042$, $\mathrm{Eff}_A(\text{SRS}) = .034$, $\mathrm{Eff}_A(\text{PPS}) = 146$,

$\mathrm{Eff}_A(\text{Exchange}) = .356$, $\mathrm{Eff}_A(\text{IBOSS}) = .324$, $\mathrm{Eff}_A(\text{ODB}) = .355$

# N=10000, n=1000; Logistic model; $E(Y)=\frac{1}{1+e^{-(\theta_0+\theta_1 x_1+\theta_2 x_2)}}$

$$\xi_D^*(\theta)=\left\{\begin{array}{cccc}(-1,-1) & (-1,1) & (1,-1) & (1,1)\\ .22 & .24 & .27 & .27\end{array}\right\},\quad \theta^T=(-1,.3,.1);\quad \begin{bmatrix}z_1\\ z_2\end{bmatrix}\sim N_2\left(\begin{bmatrix}0\\ 0\end{bmatrix},\begin{bmatrix}3 & 1.5\\ 1.5 & 3\end{bmatrix}\right)$$


Big Data

Simple Random Sample

PPS sample

Exchange algorithm sample

IBOSS sample

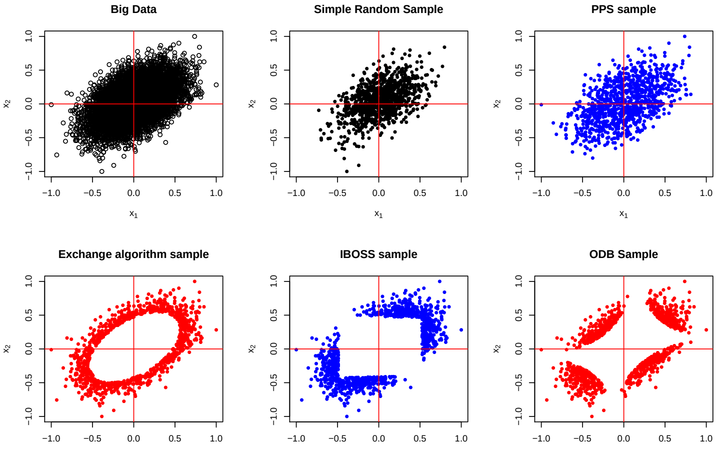
ODB Sample

$\text{Eff}_D(\text{Big Data})=.162$, $\text{Eff}_D(\text{SRS})=166$, $\text{Eff}_D(\text{PPS})=.230$,

$\text{Eff}_D(\text{Exchange})=.358$, $\text{Eff}_D(\text{IBOSS})=.337$, $\text{Eff}_D(\text{ODB})=.336$

# N=10000, n=1000; Logistic model; $E(Y)=\frac{1}{1+e^{-(\theta_0+\theta_1 x_1+\theta_2 x_2)}}$

$$\xi_A^*(\theta)=\left\{\begin{matrix}(-1,-1) & (-1,1) & (1,-1) & (1,1)\\ .22 & .24 & .27 & .27\end{matrix}\right\},\quad \theta^T=(-1,.3,.1);\quad \begin{bmatrix}z_1\\ z_2\end{bmatrix}\sim N_2\left(\begin{bmatrix}0\\ 0\end{bmatrix},\begin{bmatrix}3 & 1.5\\ 1.5 & 3\end{bmatrix}\right)$$



$\text{Eff}_A(\text{Big Data})=.069$, $\text{Eff}_A(\text{SRS})=.071$, $\text{Eff}_A(\text{PPS})=.105$,
$\text{Eff}_A(\text{Exchange})=.216$, $\text{Eff}_A(\text{IBOSS})=.169$, $\text{Eff}_A(\text{ODB})=.198$

# Simulation 1: Regression model

- We generate a $N \times (p+1)$ design matrix $\mathbf{X}_U$, where:
$$N = 10^6,\ p = 10,\ X_i \sim U_{10}(0,1)$$
- A response vector $\mathbf{y}_U$ $(N x 1)$ is simulated $S = 1000$ times from
$$\mathbf{y}_U \cong \mathbf{F}_U \theta + \varepsilon_U,$$
with $f(\mathbf{x})^T = (1, x_1, x_2, \cdots, x_{10})$, $\boldsymbol{\theta} = (2, 0.5, 1, 1, 1, 2, 2, 2, 4, 4, 4)^T$, $\text{Var}(\epsilon_i) = 9$
- At each step $s$, with $s = 1, 2, \cdots, S = 1000$:
a sample of size $n = 200$ is drown from the Big Dataset according to the following sampling scheme:
    1. ODB
    2. IBOSS
    3. SRS
    4. PPS

**N.B.** Differently from ODB and IBOSS, that are deterministic sampling scheme, SRS and PPS are random selection methods and therefore we obtain 100 different SRS and PPS independent samples at each step $s$.

# Simulation results

For each subsample we compute the D- and A-efficiencies and in addition the OLS estimates of the coefficients in the linear model.

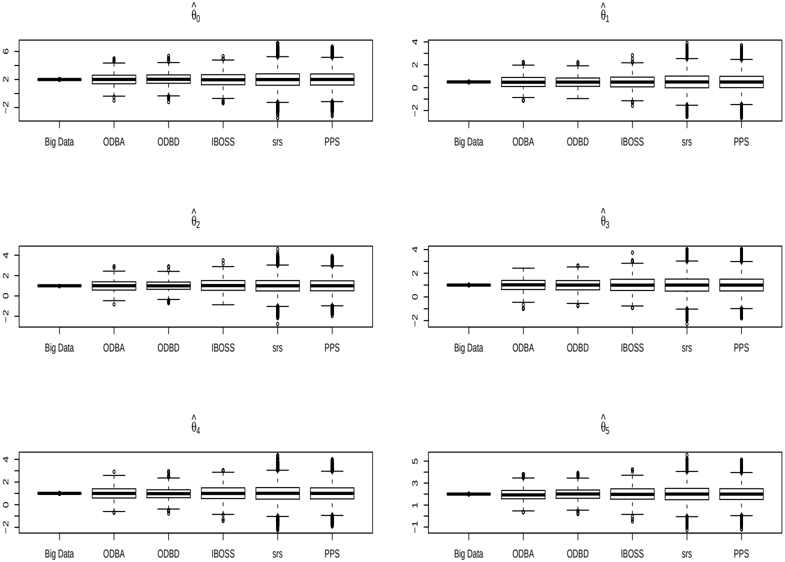
| **Φ-Efficiency** | Big Data | **ODB** | IBOSS | SRS | PPS |
|---|---|---|---|---|---|
| *D*-Efficiency | 0.3684 | **0.6170** | 0.4246 | 0.3584 | 0.3821 |
| *A*-Efficiency | 0.3549 | **0.5998** | 0.4025 | 0.3357 | 0.3594 |

Table: Monte Carlo averages of D- and A-efficiencies of the Big Data set and the subsamples obtained from ODB, IBOSS, SRS and PPS.

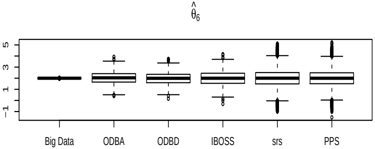
| **Criterion** | Big Data | **ODB** | IBOSS | SRS | PPS |
|---|---|---|---|---|---|
| Determinant | 1.4e-45 | **2.4e-07** | 1.7e-05 | 1.8e-04 | 8.6e-05 |
| Trace | 0.0013 | **3.9213** | 5.8610 | 7.2002 | 6.6926 |

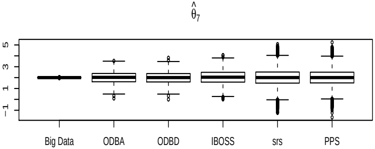
Table: Determinant and trace of the Monte Carlo covariance matrix of the estimates.

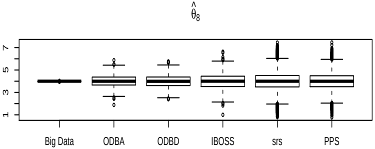
# BOXPLOT of parameter estimates

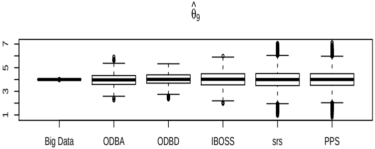


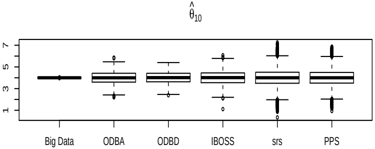
# BOXPLOT of parameter estimates











# Simulation 2: Logistic model

- We generate a $N \times (p+1)$ design matrix $\mathbf{X}_U$, where:

$$N = 10^6,\ p = 10,\ X_i \sim U_{10}(0,1)$$

- A response vector $\mathbf{y}_U$ $(Nx1)$ is simulated $S = 1000$ times from a logistic model

$$E(\mathbf{y}_U) = \frac{1}{1 + e^{-\mathbf{F}_U\,\theta}}$$

  with $f(\mathbf{x})^T = (1, x_1, x_2, \cdots, x_{10})$,
  $\boldsymbol{\theta} = (-1, 1, -0.5, -1, -0.5, 0.25, 2, -0.5, 0.5, 0.5)^T$

- At each step $s$, with $s = 1, 2, \cdots, S = 1000$:
  a sample of size $n = 200$ is drown from the Big Dataset according to the following sampling scheme:

  1. ODB
  2. IBOSS
  3. SRS
  4. PPS

**N.B.** For SRS and PPS we obtain 100 different independent samples at each step $s$.

# Simulation results

For each subsample we compute the ML estimates of the coefficients in the logistic model and in addition the D- and A-efficiencies.

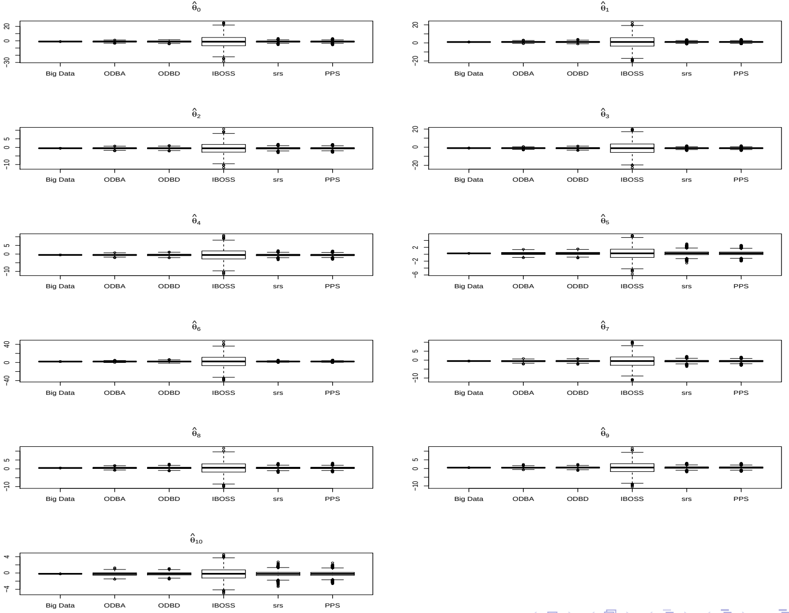
| $\Phi$-Efficiency | Big Data | **ODB** | IBOSS | SRS | PPS |
|---|---|---|---|---|---|
| *D*-Efficiency | 0.3000 | **0.6234** | 0.3085 | 0.2903 | 0.3616 |
| *A*-Efficiency | 0.4028 | **0.5468** | 0.0055 | 0.3778 | 0.4475 |

Table: Monte Carlo averages of D- and A-efficiencies of the Big Dataset and the subsamples obtained from ODB, IBOSS, SRS and PPS.

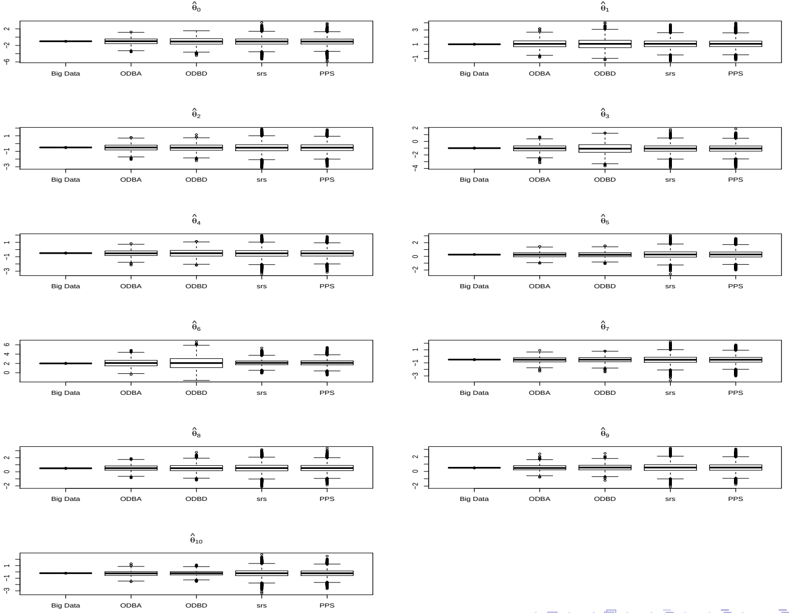
| Criterion | Big Data | **ODB** | IBOSS | SRS | PPS |
|---|---|---|---|---|---|
| Determinant | 1e-48 | **3e-09** | 1e-05 | 6e-07 | 2e-07 |
| Trace | 7e-04 | **3.59** | 400 | 4.27 | 4.00 |

Table: Determinant and trace of the Monte Carlo covariance matrix of the estimates.

# BOXPLOT of parameter estimates



# BOXPLOT of parameter estimates (without IBOSS)



# Discussion and conclusion

### Major features of our approach

- it allows to measure the quality of the Big Dataset
- when $N >> p$, it guarantees the selection of the most informative sample to estimate the parameter of the super-population model the Big Data has been generated from
- through the local linearization approach it can be applied also to GLM models
- it can be implemented for different optimality criterion.

# Discussion and conclusion

### Major features of our approach

- it allows to measure the quality of the Big Dataset
- when $N >> p$, it guarantees the selection of the most informative sample to estimate the parameter of the super-population model the Big Data has been generated from
- through the local linearization approach it can be applied also to GLM models
- it can be implemented for different optimality criterion.

### Future works

- Explore the question of sampling from Big Datasets arising from an irregular design space
- Optimize the computation time of our algorithm
- Perform a sensitivity analysis to understand how change the results according to different nominal values in the logistic model
- Examine the impact of a misspecified super-population model on the performance of our approach.

# References

- Campbell and Broderick (2019), *Automated scalable Bayesian inference via Hilbert coresets*, Journal of Machine Learning Research
- Drovandi C., Holmes C. McGree J.M., Mengersen K., Richardson S. and Ryan E. (2017). Principles of experimental design for Big Data analysis. Statistical Science.
- Harman R. and Filova L. (2016). Optimal Design: Algorithms for D-, A-, and IV- Optimal Designs., R package version 0.2, URL https://CRAN.R-project.org/package= OptimalDesign.
- Ma, P. and Sun, X. (2015). Leveraging for big data regression. Wiley Interdisciplinary Review: Computational Statistics.
- Wang, H., Yang, M. and Stufken, J. (2018). Information-Based Optimal Subdata Selection for Big Data Linear Regression, JASA.
- Wang, Zhu, Ma (2018), *Optimal Subsampling for Large Sample Logistic Regression*, Journal of the American Statistical Association.

# References

- Campbell and Broderick (2019), *Automated scalable Bayesian inference via Hilbert coresets*, Journal of Machine Learning Research
- Drovandi C., Holmes C. McGree J.M., Mengersen K., Richardson S. and Ryan E. (2017). Principles of experimental design for Big Data analysis. Statistical Science.
- Harman R. and Filova L. (2016). Optimal Design: Algorithms for D-, A-, and IV- Optimal Designs., R package version 0.2, URL https://CRAN.R-project.org/package= OptimalDesign.
- Ma, P. and Sun, X. (2015). Leveraging for big data regression. Wiley Interdisciplinary Review: Computational Statistics.
- Wang, H., Yang, M. and Stufken, J. (2018). Information-Based Optimal Subdata Selection for Big Data Linear Regression, JASA.
- Wang, Zhu, Ma (2018), *Optimal Subsampling for Large Sample Logistic Regression*, Journal of the American Statistical Association.

**Thank you for your attention**

# IBOSS sampling algorithm

Suppose that $r = \frac{n}{2\,m}$ is an integer, where $n$ is the sample size and $m$ is the number of slope parameters in the model: $\theta_1, \ldots, \theta_m$ (without the intercept).

1. for $f_1(\boldsymbol{x}_i)$ with $1 \leq i \leq N$, include $r$ data points with the r smallest $f_1(\boldsymbol{x}_i)$ values and $r$ data points with the largest $f_1(\boldsymbol{x}_i)$ values;
2. for $j = 2, \cdots, m$, exclude data points that were previously selected and from the remainder ones, select $r$ data points with the smallest $f_j(\boldsymbol{x}_i)$ values and $r$ data points with the largest $f_j(\boldsymbol{x}_i)$ values;
3. compute the estimation of $\boldsymbol{\theta}$ using the sub-data selected in the previous steps.

# Model-based vs Design-based survey sampling

Given a finite population $U = \{1, \cdots, N\}$

- Under **model-based** approach:
  1. $\mathbf{y_U} = (y_1, \cdots, y_N)$ is the realization of a $N$x1 random vector $\mathbf{Y_U}$ whose probabilistic law is the super-population model
  2. the model which generates $\mathbf{y_U}$ is the only source of variation to be taken into account if the sampling scheme is non-informative (the sampling design does not depend on $\mathbf{y_U}$)
  3. the interest is in estimating the unknown parameters of the super-population model
  4. inference may be obtained by maximum likelihood.

- Under **design-based** approach:
  1. $\{y_1, \cdots, y_N\}$ is the population value of a variable of interest $Y$
  2. the only source of random variation is that induced by the sampling mechanism
  3. the interest is in estimating finite population parameters which are specific function of $y_1, \cdots, y_N$
  4. descriptive inference is the traditional setting