Adaptive Debiased SGD in High-dimensional Generalized Linear Models with Streaming Data

Luo Yuanhang Supervisors: Han Ruijian, Huang Jian

Departament of Applied Mathematics, The Hong Kong Polytechnic University.

DEPARTMENT OF APPLIED MATHEMATICS
應 用 數 學 系

Objective

We aim to develop an online statistical inference algorithm in high-dimensional generalized linear models that sequentially updates both coefficient and variance estimates upon the arrival of a new data point. This approach has broad applications in network security, quantitative finance, and recommendation systems. In contrast to offline techniques, which require storing extensive historical raw data, our online debiasing technique enjoys a space complexity of $\mathcal{O}(p)$, built on top of the regularization annealed epoch dual averaging (RADAR) algorithm.

Introduction

A sequence of n data points $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ are independently and identically sampled from a generalized linear model:

$$\mathbb{P}(y \mid \mathbf{x}; \beta^*) \propto \exp\left\{\frac{y(\mathbf{x}^\top\beta^*) - \Phi(\mathbf{x}^\top\beta^*)}{c(\sigma)}\right\}$$

- $\mathcal{D}_i = (\mathbf{x}_i, y_i), i \in [n]$.
- $\mathbf{x}_i \in \mathbb{R}^p$. High-dimensional case: $n \ll p$.
- Sparsity: $\|\beta^*\|_0 = s_0$.

Offline approach:

$$\beta^{(n)} = \underset{\beta\in\mathbb{R}^p}{\arg\min}\frac{1}{n}\sum_{i=1}^n\left\{-y_i(\mathbf{x}_i^\top\beta) + \Phi(\mathbf{x}_i^\top\beta)\right\} + \lambda_n\|\beta\|_1,$$

However, retrieving historical raw data of space complexity $\mathcal{O}(np)$ or saving summary statistics of space complexity $\mathcal{O}(p^2)$. We can solve the problem in an online manner.

Method

We propose the approximated debiased Lasso (ADL) for online estimation and inference via a variant of online stochastic gradient descent.
Let $l_n(\beta) = \frac{1}{n}\sum_{i=1}^n\left\{-y_i\left(\mathbf{x}_i^\top\beta\right) + \Phi\left(\mathbf{x}_i^\top\beta\right)\right\}$.

- Receive incoming data.
- Update estimates $\widetilde{\beta}_j^{(k)}$ with stochastic gradients (RADAR).
- De-bias via one-step Newton correction (Adaptive RADAR):

$$\widehat{\beta}_j^{(k)} = \widetilde{\beta}_j^{(k)} - \left[\left\{\nabla^2 l_k\left(\widetilde{\beta}^{(k)}\right)\right\}^{-1}\right]_j \nabla l_k\left(\widetilde{\beta}^{(k)}\right),$$

  where
  - $[\{\nabla^2 l_n(\widetilde{\beta}^{(k)})\}^{-1}]_j \Leftarrow$ the node-wise Lasso.
  - $\nabla l_n(\widetilde{\beta}^{(k)}) \Leftarrow$ Taylor expansion approximation.
  - Obtain some statistics $s^{(k)}$ and $\widetilde{\tau}_j^{(k)}$
- The $(1-\alpha)$-confidence interval for β_j^* with $0 < \alpha < 1$ at time k is given as:

$$\left(\widehat{\beta}_j^{(k)} - z_{\alpha/2}\widetilde{\tau}_j^{(k)}, \widehat{\beta}_j^{(k)} + z_{\alpha/2}\widetilde{\tau}_j^{(k)}\right), k \in [n].$$

- Recurse variables $(\widetilde{\beta}_j^{(k)}, \widetilde{\gamma}_{\cdot,j}^{(k)}, s^{(k)}, \widetilde{\tau}_j^{(k)})$ and clear all others in memory.

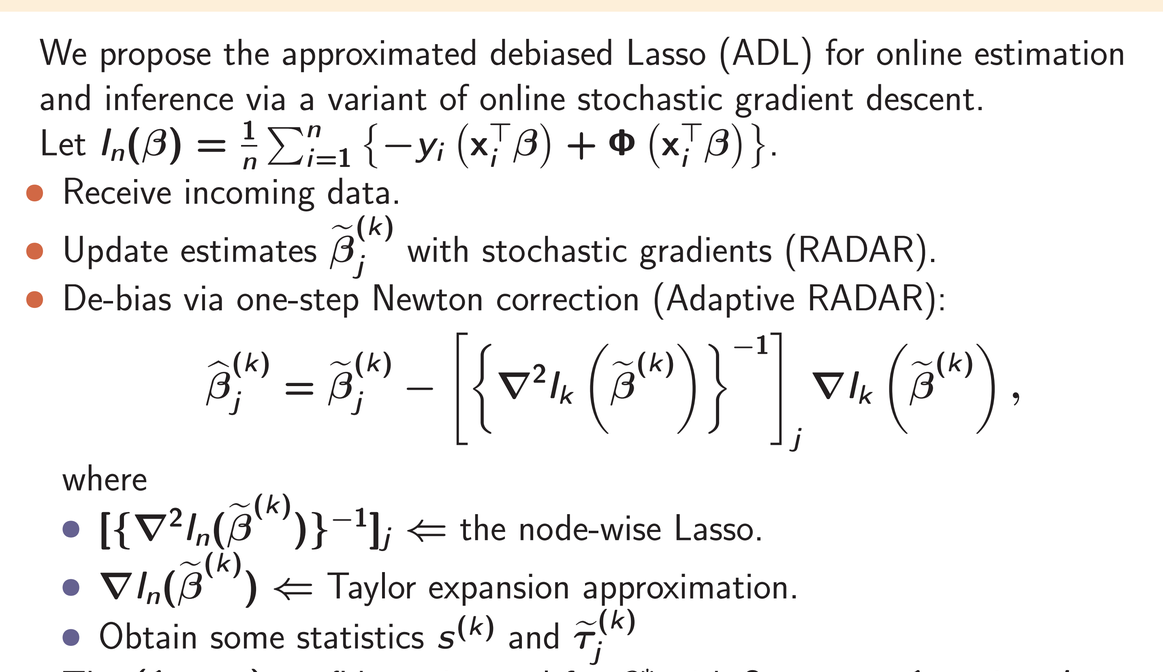

Figure 1: The whole procedure and timeline of constructing an ADL estimator. For statistical inference of the j-th element in β^*, we only need to compute and store up to some p-dimensional vectors instead of some $p \times p$ matrix.

Theorem (Convergence rate)

Under some regularity conditions, with proper choice of hyperparameters, the following events

$$\|\widetilde{\beta}^{(k)} - \beta^*\|_1 \leq \|\widetilde{\beta}^{(0)} - \beta^*\|_1 c_1 s_0\sqrt{\frac{(\log p)^3}{k}},$$
$$\|\widetilde{\beta}^{(k)} - \beta^*\|_2^2 \leq \|\widetilde{\beta}^{(0)} - \beta^*\|_2^2 c_2 s_0\frac{(\log p)^3}{k}$$

for $k \geq n_1$ hold uniformly for universal constants n_1, c_1 and c_2 with probability at least $1 - 7(\log p)^{-6}$.

Theorem (Asymptotic normality)

Under some regularity conditions, with proper choice of hyperparameters,

$$\left(\widehat{\beta}_j^{(n)} - \beta_j^*\right)/\widetilde{\tau}_j^{(n)} \to \mathcal{N}(0,1)$$

in distribution as $n \to \infty$.

Numerical Results

- Logistic regression with synthetic data:

Table 1: $n = 200$, $p = 500$, $s_0 = 6$, $\Sigma = \{0.5^{|i-j|}\}_{i,j=1,\ldots,p}$. Simulation results are summarized over 200 replications.

	β_j^*	deLasso	LSW		ODL			ADL	
Sample size n		200	200	40	120	200	40	120	200
Coverage probability	0	0.983	0.996	1.000	0.999	0.998	0.951	0.951	0.961
	1	0.598	0.936	1.000	0.880	0.705	0.891	0.955	0.943
	-1	0.610	0.946	1.000	0.861	0.670	0.881	0.961	0.945
Absolute bias	0	0.114	0.113	0.022	0.019	0.017	0.901	0.425	0.309
	1	0.325	0.260	0.657	0.590	0.539	1.008	0.423	0.328
	-1	0.326	0.259	0.660	0.598	0.548	1.034	0.426	0.324
Coverage length	0	0.753	1.207	3.423	1.691	1.287	4.419	2.053	1.532
	1	0.769	1.261	3.434	1.689	1.286	4.555	2.086	1.536
	-1	0.767	1.271	3.416	1.648	1.282	4.431	2.078	1.541
Time (s)		6.550	20.04		0.628			0.365	

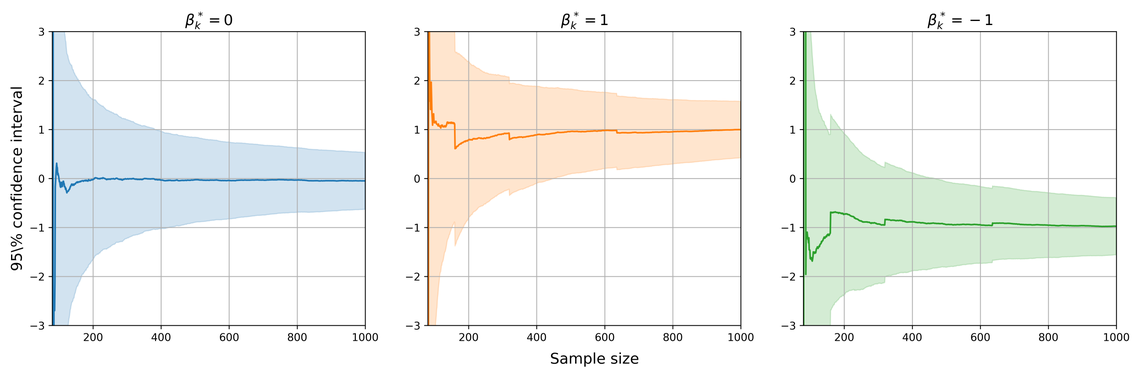

Figure 2: $n = 1000$, $p = 20000$, $s_0 = 20$, $\Sigma = \{0.5^{|i-j|}\}_{i,j=1,\ldots,p}$. Simulation results are averaged over 200 replications (around 110.4s for each).

- Large-scale URLs data application on logistic regression:
  - x: 3,231,961 features containing lexical, host-based information...
  - y: Binary, is this URL a phishing site or not?

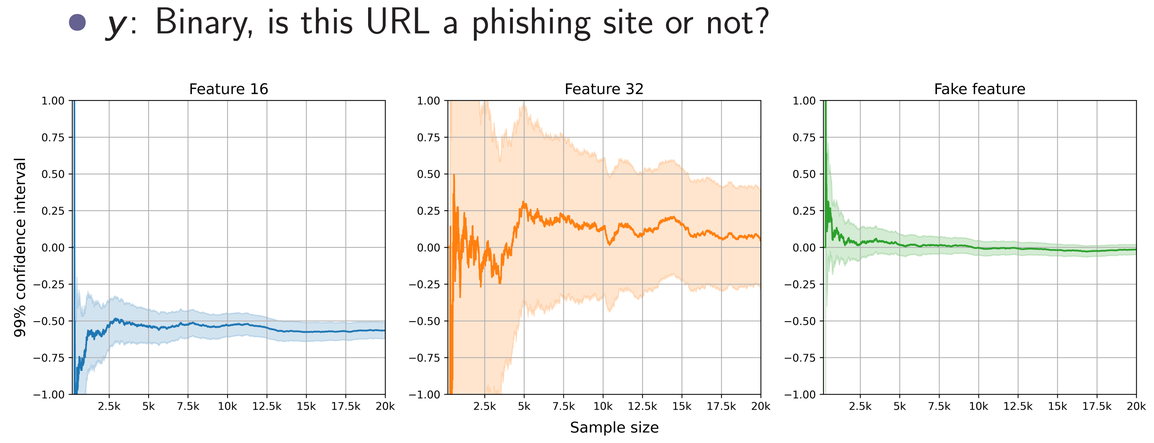

Figure 3: Trace plots of impacts of three features on the detection of malicious websites. Each data takes 2s to arrive, but ADL consumes around only 0.3s to process.

Reference

Han, R., Luo, L., Luo, Y., Lin, Y., & Huang, J. (2024). *Adaptive debiased sgd in high-dimensional glms with streaming data.* Retrieved from `https://arxiv.org/abs/2405.18284`