Online Active Regression

Cheng Chen, Yi Li, **Yiming Sun**

Division of Mathematical Science, Nanyang Technological University

July 15, 2022

Background

Linear regression: a method to model the relationship between the data points in Euclidean space and their scalar labels.

Background

Linear regression: a method to model the relationship between the data points in Euclidean space and their scalar labels.

$\min_{x \in \mathbb{R}^d} \|Ax - b\|_p$ where $A \in \mathbb{R}^{n \times d}$ and $b \in \mathbb{R}^n$

Background

Active linear regression: we can only observe a small number of labels.

Linear regression: a method to model the relationship between the data points in Euclidean space and their scalar labels.

Background

Active linear regression: we can only observe a small number of labels.

Linear regression: a method to model the relationship between the data points in Euclidean space and their scalar labels.

Online extension of regression: the learner receives data points one by one and immediately decides whether it should collect the data point.

Online Active Regression

In the online learning, for given $A \in \mathbb{R}^{n\times d}$, $b \in \mathbb{R}^n$ and $0 < \epsilon \leq 1$, find $\tilde{x}$ such that

$$\|A\tilde{x} - b\|_p \leq (1+\epsilon) \min_{x\in\mathbb{R}^d} \|Ax - b\|_p,$$

while querying as few labels as possible.

Contributions

	Queries*	Offline	Space	Runtime
ℓ_p $p \in [1,2]$	$\tilde{\mathcal{O}}(\frac{d}{\epsilon^2}\log(n\kappa^{\mathsf{OL}}))$ †	$\tilde{\mathcal{O}}(\frac{d}{\epsilon^2})$	$\tilde{\mathcal{O}}(\frac{d^2}{\epsilon^2}\log(n\kappa^{\mathsf{OL}}))$	IPM$+n\cdot$ $\mathcal{O}(d^3\,\mathrm{poly}(\log n))$
ℓ_2	$\tilde{\mathcal{O}}(\frac{d}{\epsilon^2}\log(n\frac{\\|A\\|_2}{\sigma}))$ ‡	$\tilde{\mathcal{O}}(\frac{d}{\epsilon})$	$\tilde{\mathcal{O}}(\frac{d^2}{\epsilon^2}\log(n\frac{\\|A\\|_2}{\sigma}))$	$\mathcal{O}(\mathrm{nnz}(A)\log n)+$ $\tilde{\mathcal{O}}(\frac{d^3}{\epsilon^4}\log\frac{\\|A\\|_2}{\sigma}\cdot$ $\log\frac{1}{\epsilon}(\log n+d))$

* See the arxiv version for our latest results: `https://arxiv.org/abs/2207.05945`.

† $\kappa^{\mathsf{OL}} = \|A\|_2 \max_i \|(A^{(i)})^\dagger\|_2$ where $A^{(i)}$ is the submatrix consisting of the first i rows of A.

‡ σ is the smallest singular value of the first d rows of A.

Background

Definition 1 (Subspace Embedding)

Let $A \in \mathbb{R}^{n \times d}$ and $S \in \mathbb{R}^{m \times n}$, if $(1-\epsilon)\|Ax\|_p \leq \|SAx\|_p \leq (1+\epsilon)\|Ax\|_p$ holds for any $x \in \mathbb{R}^d$, then S is an ϵ-subspace embedding (ϵ-SE) matrix for A.

Background

Definition 1 (Subspace Embedding)

Let $A \in \mathbb{R}^{n\times d}$ and $S \in \mathbb{R}^{m\times n}$, if $(1-\epsilon)\|Ax\|_p \leq \|SAx\|_p \leq (1+\epsilon)\|Ax\|_p$ holds for any $x \in \mathbb{R}^d$, then S is an ϵ-subspace embedding (ϵ-SE) matrix for A.

General idea to solve linear regression:

$$\min_{x\in\mathbb{R}^d} \|Ax-b\|_p \xrightarrow[\text{matrix } S \text{ for } [A\ b]]{\text{generating } \epsilon\text{-SE}} \min_{x\in\mathbb{R}^d} \|SAx-Sb\|_p$$

$$\xrightarrow[\mathrm{Reg}(A,b,p)=\arg\min\limits_{x\in\mathbb{R}^d}\|Ax-b\|_p]{\tilde{x}=\mathrm{Reg}(SA,Sb,p)} \|A\tilde{x}-b\|_p \leq (1+\epsilon)\min_{x\in\mathbb{R}^d}\|Ax-b\|_p$$

Background

Cohen and Peng'15 proved that ℓ_p Lewis weights sampling can generate subspace embedding matrix with $\tilde{\mathcal{O}}(\epsilon^{-2}d)$ rows.

Background

Cohen and Peng'15 proved that ℓ_p Lewis weights sampling can generate subspace embedding matrix with $\tilde{\mathcal{O}}(\epsilon^{-2}d)$ rows.

Definition 2 (Lewis Weights)

The ℓ_p Lewis weights of A are defined to be $w_i = (a_i^\top (A^\top W^{1-\frac{2}{p}} A)^{-1} a_i)^{\frac{p}{2}}$, where W is a diagonal matrix with diagonal entries $w_1, \ldots, w_n$ and a_i is the i-th row of A.

Sampling matrix $S_{i,i} = p_i^{-\frac{1}{p}}$ with probability p_i, where $p_i = \min\{\beta w_i, 1\}$.

Background

- In the active learning, it is impossible to query every b_i.
- In the online learning, it is impossible to get the whole A.

Background

- In the active learning, it is impossible to query every b_i.
  - Musco et al.'21 solved the active regression in the offline setting by proposing a new algorithm based on ℓ_p lewis weights sampling.
- In the online learning, it is impossible to get the whole A.
  - Braverman et al.'20 solved online ℓ_1 Lewis weights sampling.

Methods

$$A \in \mathbb{R}^{n\times d},\ b \in \mathbb{R}^{n} \xrightarrow[\text{Musco et al. 21}]{\text{Lewis weights sampling}} \tilde{x} \in \mathbb{R}^{d}$$

$$[A\ b] \xrightarrow{S \in \mathbb{R}^{\tilde{O}(d)\times n}} x_c = \mathrm{Reg}(SA, Sb, p)$$

Methods

$$A \in \mathbb{R}^{n\times d},\ b \in \mathbb{R}^n \xrightarrow[\text{Musco et al. 21}]{\text{Lewis weights sampling}} \tilde{x} \in \mathbb{R}^d$$

Sampling scheme:

$$[A\ b] \xrightarrow{S \in \mathbb{R}^{\tilde{\mathcal{O}}(d)\times n}} x_c = \mathrm{Reg}(SA, Sb, p) \xrightarrow[z = b - Ax_c]{S_1 \in \mathbb{R}^{\frac{\tilde{\mathcal{O}}(d)\log n}{\epsilon^{\mathbf{2p+5}}}\times n}} \hat{x} = \mathrm{Reg}(S_1A, S_1z, p)$$

Methods

$$A \in \mathbb{R}^{n\times d},\ b \in \mathbb{R}^n \xrightarrow[\text{Musco et al. 21}]{\text{Lewis weights sampling}} \tilde{x} \in \mathbb{R}^d$$

Sampling scheme:

$$[A\ b] \xrightarrow{S \in \mathbb{R}^{\tilde{\mathcal{O}}(d)\times n}} x_c = \mathrm{Reg}(SA, Sb, p) \xrightarrow[z = b - Ax_c]{S_1 \in \mathbb{R}^{\frac{\tilde{\mathcal{O}}(d)\log n}{\epsilon^{\mathbf{2}p+\mathbf{5}}}\times n}} \hat{x} = \mathrm{Reg}(S_1 A, S_1 z, p)$$

$$\xrightarrow{\text{output}} \tilde{x} = x_c + \hat{x}$$

Methods

Definition 3 (Online Lewis Weights)

For $A \in \mathbb{R}^{n \times d}$, the ℓ_p online Lewis weights of A are defined to be $w_i^{\mathsf{OL}}(A) = w_i(A_i)$, where A_i is the submatrix consisting of the first i rows of A.

Methods

Definition 3 (Online Lewis Weights)

For $A \in \mathbb{R}^{n \times d}$, the ℓ_p online Lewis weights of A are defined to be $w_i^{\mathrm{OL}}(A) = w_i(A_i)$, where A_i is the submatrix consisting of the first i rows of A.

- $w_i^{\mathrm{OL}}(A) \geq w_i(A)$

Methods

Definition 3 (Online Lewis Weights)

For $A \in \mathbb{R}^{n \times d}$, the ℓ_p online Lewis weights of A are defined to be $w_i^{\mathrm{OL}}(A) = w_i(A_i)$, where A_i is the submatrix consisting of the first i rows of A.

- $w_i^{\mathrm{OL}}(A) \geq w_i(A)$
- $\sum_i w_i^{\mathrm{OL}}(A) = \mathcal{O}(d \log n \log \kappa^{\mathrm{OL}}(A))$

Methods

Definition 3 (Online Lewis Weights)

For $A \in \mathbb{R}^{n\times d}$, the ℓ_p online Lewis weights of A are defined to be $w_i^{\mathrm{OL}}(A) = w_i(A_i)$, where A_i is the submatrix consisting of the first i rows of A.

- $w_i^{\mathrm{OL}}(A) \geq w_i(A)$
- $\sum_i w_i^{\mathrm{OL}}(A) = \mathcal{O}(d \log n \log \kappa^{\mathrm{OL}}(A))$
- Runtime to calculate w_i^{OL} is $\mathcal{O}(id^2 + d^3)$; words of space is $\mathcal{O}(id)$.

Methods

- Approximate online Lewis weights: $\tilde{w}_i^{\mathrm{OL}}(A) = (a_i^\top (\tilde{A}_i^\top W^{1-\frac{2}{p}} \tilde{A}_i)^{-1} a_i)^{\frac{p}{2}}$.
- $$\tilde{A}_i \xleftarrow[\text{Braverman et al.'20}]{\text{compression algorithm for } \ell_1} A_i$$

Methods

- Approximate online Lewis weights: $\tilde{w}_i^{\mathrm{OL}}(A) = (a_i^\top (\tilde{A}_i^\top W^{1-\frac{2}{p}} \tilde{A}_i)^{-1} a_i)^{\frac{p}{2}}$.
- $$\tilde{A}_i \xleftarrow[\text{Braverman et al.'20}]{\text{compression algorithm for } \ell_1} A_i$$
- Generalize the compression algorithm to ℓ_p.

Technical Contribution

$$M = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_r \\ B \end{bmatrix} \xrightarrow{\text{Compress}} M' = \begin{bmatrix} S_1 A_1 \\ S_2 A_2 \\ \vdots \\ S_r A_r \\ B \end{bmatrix}$$

Lemma 4

Let $A_i \in \mathbb{R}^{n_i \times d}$ and $B \in \mathbb{R}^{k \times d}$. Let $S_i \in \mathbb{R}^{m_i \times n_i}$ be the sampling matrix for A_i. Then we have with probability at least $1 - \delta$,

$$(1-\eta) w_{i+\sum_j n_j}(M) \leq w_{i+\sum_j m_j}(M') \leq (1+\eta) w_{i+\sum_j n_j}(M)$$

for any $i \in [k]$ and $m_i = \eta^{-2} d \log(d/\delta)$.

Methods

- $\mathcal{O}(\eta^{-2} d \,\mathrm{poly}(\log n))$ rows in M'.

Methods

- $\mathcal{O}(\eta^{-2} d \,\mathrm{poly}(\log n))$ rows in M'.
- $(1-\eta) w_i^{\mathsf{OL}}(A) \leq \tilde{w}_i^{\mathsf{OL}}(A) \leq (1+\eta) w_i^{\mathsf{OL}}(A)$

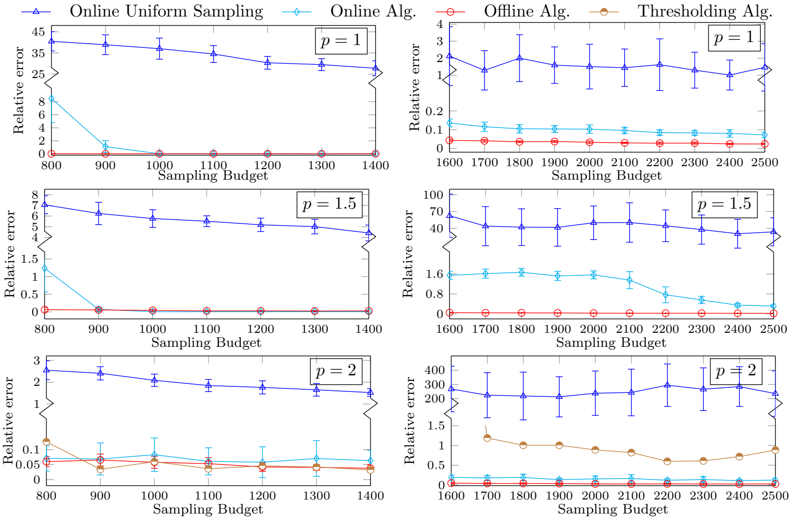
Methods

- $\mathcal{O}(\eta^{-2} d \operatorname{poly}(\log n))$ rows in M'.
- $(1-\eta) w_i^{\mathrm{OL}}(A) \leq \tilde{w}_i^{\mathrm{OL}}(A) \leq (1+\eta) w_i^{\mathrm{OL}}(A)$
- $\tilde{w}_i^{\mathrm{OL}}(A)$ can be applied to online regression.

Methods

- $\mathcal{O}(\eta^{-2} d \,\mathrm{poly}(\log n))$ rows in M'.
- $(1-\eta) w_i^{\mathrm{OL}}(A) \leq \tilde{w}_i^{\mathrm{OL}}(A) \leq (1+\eta) w_i^{\mathrm{OL}}(A)$
- $\tilde{w}_i^{\mathrm{OL}}(A)$ can be applied to online regression.
- Time to calculate $\tilde{w}_i^{\mathrm{OL}}(A)$ is $\mathcal{O}(\eta^{-2} d^3 \,\mathrm{poly}(\log \frac{n}{\eta}))$; words of space is $\mathcal{O}(\eta^{-2} d \,\mathrm{poly}(\log n))$.

Experiments



Synthetic data

Real-world data: Gas Sensor dataset

Thank you!