Simultaneous Inference for Massive Data: Distributed Bootstrap

Yang Yu[1], Shih-Kang Chao[2], Guang Cheng[1]

[1]Purdue University

[2]University of Missouri

ICML 2020

We have N i.i.d. data points: $Z_1, \ldots, Z_N$

Estimation:
Fit a model that has an unknown parameter $\theta \in \mathbb{R}^d$ by minimizing the empirical risk

$$\widehat{\theta} := \arg\min_{\theta \in \mathbb{R}^d} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(\theta; Z_i)$$

We have N i.i.d. data points: $Z_1, \ldots, Z_N$

Estimation:
Fit a model that has an unknown parameter $\theta \in \mathbb{R}^d$ by minimizing the empirical risk

$$\widehat{\theta} := \arg\min_{\theta \in \mathbb{R}^d} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(\theta; Z_i)$$

Ideally, we want $\widehat{\theta}$ to be close to the expected risk minimizer

$$\theta^* := \arg\min_{\theta \in \mathbb{R}^d} \mathbb{E}_Z[\mathcal{L}(\theta; Z)]$$

We have N i.i.d. data points: $Z_1, \dots, Z_N$

Estimation:
Fit a model that has an unknown parameter $\theta \in \mathbb{R}^d$ by minimizing the empirical risk

$$\widehat{\theta} := \arg\min_{\theta \in \mathbb{R}^d} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(\theta; Z_i)$$

Ideally, we want $\widehat{\theta}$ to be close to the expected risk minimizer

$$\theta^* := \arg\min_{\theta \in \mathbb{R}^d} \mathbb{E}_Z[\mathcal{L}(\theta; Z)]$$

Examples:

- Linear regression: $Z = (X, Y)$, $\mathcal{L}(\theta; Z) = (Y - X^\top \theta)^2/2$
- Logistic regression: $Z = (X, Y)$, $\mathcal{L}(\theta; Z) = -YX^\top \theta + \log(1 + \exp[X^\top \theta])$

Inference: Find an interval $[L, U]$ s.t.

$$P(\theta_1^* \in [L, U]) \approx 95\%$$

Inference: Find an interval $[L, U]$ s.t.

$$P(\theta_1^* \in [L, U]) \approx 95\%$$

Simultaneous Inference: Find intervals $[L_1, U_1], \ldots, [L_d, U_d]$ s.t.

$$P(\theta_1^* \in [L_1, U_1], \ldots, \theta_d^* \in [L_d, U_d]) \approx 95\%$$

Inference: Find an interval $[L, U]$ s.t.

$$P(\theta_1^* \in [L, U]) \approx 95\%$$

Simultaneous Inference: Find intervals $[L_1, U_1], \ldots, [L_d, U_d]$ s.t.

$$P(\theta_1^* \in [L_1, U_1], \ldots, \theta_d^* \in [L_d, U_d]) \approx 95\%$$

How to perform **Simultaneous** Inference:

Inference: Find an interval $[L, U]$ s.t.

$$P(\theta_1^* \in [L, U]) \approx 95\%$$

Simultaneous Inference: Find intervals $[L_1, U_1], \ldots, [L_d, U_d]$ s.t.

$$P(\theta_1^* \in [L_1, U_1], \ldots, \theta_d^* \in [L_d, U_d]) \approx 95\%$$

How to perform **Simultaneous** Inference:

Step 1: Compute point estimator $\widehat{\theta}$

Inference: Find an interval $[L, U]$ s.t.

$$P(\theta_1^* \in [L, U]) \approx 95\%$$

Simultaneous Inference: Find intervals $[L_1, U_1], \ldots, [L_d, U_d]$ s.t.

$$P(\theta_1^* \in [L_1, U_1], \ldots, \theta_d^* \in [L_d, U_d]) \approx 95\%$$

How to perform **Simultaneous** Inference:

Step 1: Compute point estimator $\widehat{\theta}$

Step 2: Estimate the 0.95-quantile $c(0.95)$ of $\left\|\sqrt{N}\left(\widehat{\theta} - \theta^*\right)\right\|_\infty$ (by bootstrap)

Inference: Find an interval $[L, U]$ s.t.

$$P(\theta_1^* \in [L, U]) \approx 95\%$$

Simultaneous Inference: Find intervals $[L_1, U_1], \ldots, [L_d, U_d]$ s.t.

$$P(\theta_1^* \in [L_1, U_1], \ldots, \theta_d^* \in [L_d, U_d]) \approx 95\%$$

How to perform **Simultaneous** Inference:

Step 1: Compute point estimator $\widehat{\theta}$

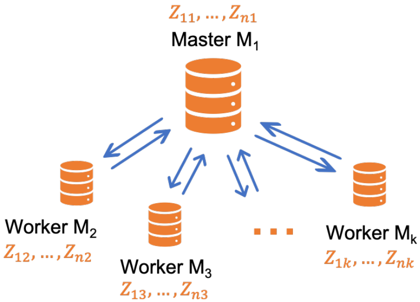
Step 2: Estimate the 0.95-quantile $c(0.95)$ of $\left\|\sqrt{N}\left(\widehat{\theta} - \theta^*\right)\right\|_\infty$ (by bootstrap)

Step 3: For $l = 1, \ldots, d$,

$$L_l = \widehat{\theta}_l - \frac{\widehat{c}(0.95)}{\sqrt{N}}, \quad U_l = \widehat{\theta}_l + \frac{\widehat{c}(0.95)}{\sqrt{N}}$$

Distributed framework:
Distribute N data points evenly across k machines s.t. each machine stores $n = N/k$ data points

- 1 master node $\mathcal{M}_1$
- $k-1$ worker nodes $\mathcal{M}_2, \mathcal{M}_3, \ldots, \mathcal{M}_k$
- Z_{ij}: the i-th data point at machine $\mathcal{M}_j$



Distributed Simultaneous Inference

[1]Kleiner, et al. "A scalable bootstrap for massive data." JRSS-B (2014)
[2]Sengupta, et al. "A subsampled double bootstrap for massive data." JASA (2016)

Distributed Simultaneous Inference

Step 1: Compute $\widehat{\theta}$

- Can be approximated by existing efficient distributed estimation methods

[1] Kleiner, et al. "A scalable bootstrap for massive data." JRSS-B (2014)
[2] Sengupta, et al. "A subsampled double bootstrap for massive data." JASA (2016)

Distributed Simultaneous Inference

Step 1: Compute $\widehat{\theta}$

- Can be approximated by existing efficient distributed estimation methods

Step 2: Bootstrap $c(0.95)$

- Traditional bootstrap cannot be efficiently applied in the distributed framework
- BLB[1] and SDB[2] are computationally expensive due to repeated resampling and not suitable for large k

[1] Kleiner, et al. "A scalable bootstrap for massive data." JRSS-B (2014)
[2] Sengupta, et al. "A subsampled double bootstrap for massive data." JASA (2016)

Question: How can we efficiently do Step 2 in a distributed manner?

Question: How can we efficiently do Step 2 in a distributed manner?

Our contributions:

- We propose communication-efficient and computation-efficient distributed bootstrap methods: `k-grad` and `n+k-1-grad`
- We prove a sufficient number of communication rounds that guarantees statistical accuracy and efficiency

Approximate by sample average:

$$\|\sqrt{N}(\widehat{\theta} - \theta^*)\|_\infty \approx \left\| \mathbb{E}[\nabla^2 \mathcal{L}(\theta^*; Z)]^{-1} \frac{1}{\sqrt{N}} \sum_{i=1}^{n} \sum_{j=1}^{k} \nabla \mathcal{L}(\theta^*; Z_{ij}) \right\|_\infty$$

Approximate by sample average:

$$\|\sqrt{N}(\widehat{\theta} - \theta^*)\|_\infty \approx \left\| \mathbb{E}[\nabla^2 \mathcal{L}(\theta^*; Z)]^{-1} \frac{1}{\sqrt{N}} \sum_{i=1}^{n} \sum_{j=1}^{k} \nabla \mathcal{L}(\theta^*; Z_{ij}) \right\|_\infty$$

Multiplier bootstrap: $\epsilon_{ij} \overset{\text{iid}}{\sim} \mathcal{N}(0,1)$ for $i = 1, \ldots, n$ and $j = 1, \ldots, k$

$$\|\sqrt{N}(\widehat{\theta} - \theta^*)\|_\infty \overset{\mathcal{D}}{\approx} \left\| \mathbb{E}[\nabla^2 \mathcal{L}(\theta^*; Z)]^{-1} \frac{1}{\sqrt{N}} \sum_{i=1}^{n} \sum_{j=1}^{k} \epsilon_{ij} \nabla \mathcal{L}(\theta^*; Z_{ij}) \right\|_\infty \Big| \{Z_{ij}\}_{i,j}$$

Approximate by sample average:

$$\|\sqrt{N}(\widehat{\theta}-\theta^*)\|_\infty \approx \left\|\mathbb{E}[\nabla^2\mathcal{L}(\theta^*;Z)]^{-1}\frac{1}{\sqrt{N}}\sum_{i=1}^{n}\sum_{j=1}^{k}\nabla\mathcal{L}(\theta^*;Z_{ij})\right\|_\infty$$

Multiplier bootstrap: $\epsilon_{ij} \overset{\text{iid}}{\sim} \mathcal{N}(0,1)$ for $i=1,\ldots,n$ and $j=1,\ldots,k$

$$\|\sqrt{N}(\widehat{\theta}-\theta^*)\|_\infty \overset{\mathcal{D}}{\approx} \left\|\mathbb{E}[\nabla^2\mathcal{L}(\theta^*;Z)]^{-1}\frac{1}{\sqrt{N}}\sum_{i=1}^{n}\sum_{j=1}^{k}\epsilon_{ij}\nabla\mathcal{L}(\theta^*;Z_{ij})\right\|_\infty \Big| \{Z_{ij}\}_{i,j}$$

`k-grad` (computed at $\mathcal{M}_1$): $\epsilon_j \overset{\text{iid}}{\sim} \mathcal{N}(0,1)$ for $j=1,\ldots,k$

$$\|\sqrt{N}(\widehat{\theta}-\theta^*)\|_\infty \overset{\mathcal{D}}{\approx} \overline{W} := \left\|\widetilde{\Theta}\frac{1}{\sqrt{k}}\sum_{j=1}^{k}\epsilon_j\sqrt{n}(\mathbf{g}_j-\bar{\mathbf{g}})\right\|_\infty \Big| \{Z_{ij}\}_{i,j}$$

where $\mathbf{g}_j = \frac{1}{n}\sum_{i=1}^{n}\nabla\mathcal{L}(\bar{\theta};Z_{ij})$ computed at $\mathcal{M}_j$, transmitted to $\mathcal{M}_1$

$\bar{\mathbf{g}} = \frac{1}{k}\sum_{j=1}^{k}\mathbf{g}_j$ averaged at $\mathcal{M}_1$, $\quad \widetilde{\Theta} = \left(\frac{1}{n}\sum_{i=1}^{n}\nabla^2\mathcal{L}(\bar{\theta};Z_{i1})\right)^{-1}$ computed at $\mathcal{M}_1$

`k-grad` fails for small k!

`k-grad` fails for small k!

Solution: `n+k-1-grad` (computed at $\mathcal{M}_1$):
$\epsilon_{i1}, \epsilon_j \overset{\text{iid}}{\sim} \mathcal{N}(0,1)$ for $i = 1, \ldots, n$ and $j = 2, \ldots, k$

$$\widetilde{W} := \left\| \widetilde{\Theta} \frac{1}{\sqrt{n+k-1}} \left(\sum_{i=1}^{n} \epsilon_{i1} (\mathbf{g}_{i1} - \bar{\mathbf{g}}) + \sum_{j=2}^{k} \epsilon_j \sqrt{n} (\mathbf{g}_j - \bar{\mathbf{g}}) \right) \right\|_\infty \Big| \{Z_{ij}\}_{i,j}$$

where $\quad \mathbf{g}_{i1} = \nabla \mathcal{L}(\bar{\theta}; Z_{i1})$ computed at $\mathcal{M}_1$

An example algorithm: apply `k-grad`/`n+k-1-grad` with CSL estimator[3]

[1] Jordan, et al. "Communication-efficient distributed statistical inference." JASA (2019)

An example algorithm: apply `k-grad/n+k-1-grad` with CSL estimator[3]

Step 1: compute point estimator $\widetilde{\theta}$ (τ rounds of communication)

$\widetilde{\theta}^{(0)} \leftarrow \arg\min_\theta \mathcal{L}_1(\theta)$ at $\mathcal{M}_1$
for $t = 1, \ldots, \tau$ **do**
    Transmit $\widetilde{\theta}^{(t-1)}$ to $\{\mathcal{M}_j\}_{j=2}^k$
    Compute $\nabla\mathcal{L}_1(\widetilde{\theta}^{(t-1)})$ and $\nabla^2\mathcal{L}_1(\widetilde{\theta}^{(t-1)})^{-1}$ at $\mathcal{M}_1$
    for $j = 2, \ldots, k$ **do**
        Compute $\nabla\mathcal{L}_j(\widetilde{\theta}^{(t-1)})$ at $\mathcal{M}_j$
        Transmit $\nabla\mathcal{L}_j(\widetilde{\theta}^{(t-1)})$ to $\mathcal{M}_1$
    $\nabla\mathcal{L}_N(\widetilde{\theta}^{(t-1)}) \leftarrow k^{-1}\sum_{j=1}^k \nabla\mathcal{L}_j(\widetilde{\theta}^{(t-1)})$ at $\mathcal{M}_1$
    $\widetilde{\theta}^{(t)} \leftarrow \widetilde{\theta}^{(t-1)} - \nabla^2\mathcal{L}_1(\widetilde{\theta}^{(t-1)})^{-1}\nabla\mathcal{L}_N(\widetilde{\theta}^{(t-1)})$ at $\mathcal{M}_1$

---

[1] Jordan, et al. "Communication-efficient distributed statistical inference." JASA (2019)

An example algorithm: apply `k-grad/n+k-1-grad` with CSL estimator[3]

Step 1: compute point estimator $\widetilde{\theta}$ (τ rounds of communication)

$\widetilde{\theta}^{(0)} \leftarrow \arg\min_\theta \mathcal{L}_1(\theta)$ at $\mathcal{M}_1$

for $t = 1, \ldots, \tau$ **do**

  Transmit $\widetilde{\theta}^{(t-1)}$ to $\{\mathcal{M}_j\}_{j=2}^k$

  Compute $\nabla \mathcal{L}_1(\widetilde{\theta}^{(t-1)})$ and $\nabla^2 \mathcal{L}_1(\widetilde{\theta}^{(t-1)})^{-1}$ at $\mathcal{M}_1$

  for $j = 2, \ldots, k$ **do**

    Compute $\nabla \mathcal{L}_j(\widetilde{\theta}^{(t-1)})$ at $\mathcal{M}_j$

    Transmit $\nabla \mathcal{L}_j(\widetilde{\theta}^{(t-1)})$ to $\mathcal{M}_1$

  $\nabla \mathcal{L}_N(\widetilde{\theta}^{(t-1)}) \leftarrow k^{-1} \sum_{j=1}^k \nabla \mathcal{L}_j(\widetilde{\theta}^{(t-1)})$ at $\mathcal{M}_1$

  $\widetilde{\theta}^{(t)} \leftarrow \widetilde{\theta}^{(t-1)} - \nabla^2 \mathcal{L}_1(\widetilde{\theta}^{(t-1)})^{-1} \nabla \mathcal{L}_N(\widetilde{\theta}^{(t-1)})$ at $\mathcal{M}_1$

Step 2: bootstrap quantile $c(0.95)$ (0 round of communication)

Run `k-grad/n+k-1-grad` with $\bar{\theta} = \widetilde{\theta}^{(t-1)}$ at $\mathcal{M}_1$

---

[1] Jordan, et al. "Communication-efficient distributed statistical inference." JASA (2019)

An example algorithm: apply `k-grad`/`n+k-1-grad` with CSL estimator[3]

Step 1: compute point estimator $\widetilde{\theta}$ (τ rounds of communication)

$\widetilde{\theta}^{(0)} \leftarrow \arg\min_\theta \mathcal{L}_1(\theta)$ at $\mathcal{M}_1$
for $t = 1, \ldots, \tau$ **do**
  Transmit $\widetilde{\theta}^{(t-1)}$ to $\{\mathcal{M}_j\}_{j=2}^k$
  Compute $\nabla \mathcal{L}_1(\widetilde{\theta}^{(t-1)})$ and $\nabla^2 \mathcal{L}_1(\widetilde{\theta}^{(t-1)})^{-1}$ at $\mathcal{M}_1$
  for $j = 2, \ldots, k$ **do**
    Compute $\nabla \mathcal{L}_j(\widetilde{\theta}^{(t-1)})$ at $\mathcal{M}_j$
    Transmit $\nabla \mathcal{L}_j(\widetilde{\theta}^{(t-1)})$ to $\mathcal{M}_1$
  $\nabla \mathcal{L}_N(\widetilde{\theta}^{(t-1)}) \leftarrow k^{-1} \sum_{j=1}^k \nabla \mathcal{L}_j(\widetilde{\theta}^{(t-1)})$ at $\mathcal{M}_1$
  $\widetilde{\theta}^{(t)} \leftarrow \widetilde{\theta}^{(t-1)} - \nabla^2 \mathcal{L}_1(\widetilde{\theta}^{(t-1)})^{-1} \nabla \mathcal{L}_N(\widetilde{\theta}^{(t-1)})$ at $\mathcal{M}_1$

Step 2: bootstrap quantile $c(0.95)$ (0 round of communication)

Run `k-grad`/`n+k-1-grad` with $\bar{\theta} = \widetilde{\theta}^{(t-1)}$ at $\mathcal{M}_1$

Step 3: 95% simultaneous confidence interval

$\widetilde{\theta}_l^{(t)} \pm N^{-1/2} \widetilde{c}(0.95)$ for $l = 1, \ldots, d$

In total, τ rounds of communication

[1] Jordan, et al. "Communication-efficient distributed statistical inference." JASA (2019)

An example algorithm: apply `k-grad/n+k-1-grad` with CSL estimator[3]

Step 1: compute point estimator $\widetilde{\theta}$ (τ rounds of communication)

$\widetilde{\theta}^{(0)} \leftarrow \arg\min_\theta \mathcal{L}_1(\theta)$ at $\mathcal{M}_1$

for $t = 1, \dots, \tau$ **do**

  Transmit $\widetilde{\theta}^{(t-1)}$ to $\{\mathcal{M}_j\}_{j=2}^k$

  Compute $\nabla \mathcal{L}_1(\widetilde{\theta}^{(t-1)})$ and $\nabla^2 \mathcal{L}_1(\widetilde{\theta}^{(t-1)})^{-1}$ at $\mathcal{M}_1$

  for $j = 2, \dots, k$ **do**

    Compute $\nabla \mathcal{L}_j(\widetilde{\theta}^{(t-1)})$ at $\mathcal{M}_j$

    Transmit $\nabla \mathcal{L}_j(\widetilde{\theta}^{(t-1)})$ to $\mathcal{M}_1$

  $\nabla \mathcal{L}_N(\widetilde{\theta}^{(t-1)}) \leftarrow k^{-1} \sum_{j=1}^k \nabla \mathcal{L}_j(\widetilde{\theta}^{(t-1)})$ at $\mathcal{M}_1$

  $\widetilde{\theta}^{(t)} \leftarrow \widetilde{\theta}^{(t-1)} - \nabla^2 \mathcal{L}_1(\widetilde{\theta}^{(t-1)})^{-1} \nabla \mathcal{L}_N(\widetilde{\theta}^{(t-1)})$ at $\mathcal{M}_1$

Step 2: bootstrap quantile $c(0.95)$ (0 round of communication)

Run `k-grad/n+k-1-grad` with $\bar{\theta} = \widetilde{\theta}^{(t-1)}$ at $\mathcal{M}_1$

Step 3: 95% simultaneous confidence interval

$\widetilde{\theta}_l^{(t)} \pm N^{-1/2} \widetilde{c}(0.95)$ for $l = 1, \dots, d$

In total, τ rounds of communication

Question: How many rounds of communication are sufficient?

[1] Jordan, et al. "Communication-efficient distributed statistical inference." JASA (2019)

Assume $n = d^{\gamma_n}$ and $k = d^{\gamma_k}$ for constants $\gamma_n, \gamma_k > 0$

Assume $n = d^{\gamma_n}$ and $k = d^{\gamma_k}$ for constants $\gamma_n, \gamma_k > 0$

Statistical accuracy: $\sup_{\alpha \in (0,1)} |P(\|\sqrt{N}(\widetilde{\theta} - \theta^*)\|_\infty \leq c_W(\alpha)) - \alpha| = o(1)$

Assume $n = d^{\gamma_n}$ and $k = d^{\gamma_k}$ for constants $\gamma_n, \gamma_k > 0$

Statistical accuracy: $\sup_{\alpha \in (0,1)} |P(\|\sqrt{N}(\widetilde{\theta} - \theta^*)\|_\infty \leq c_W(\alpha)) - \alpha| = o(1)$
Statistical efficiency: $\sup_{\alpha \in (0,1)} |P(\|\sqrt{N}(\widehat{\theta} - \theta^*)\|_\infty \leq c_W(\alpha)) - \alpha| = o(1)$

Illustration of main results for **linear models**:

Left: `k-grad` Right: `n+k-1-grad`

Blue areas: accuracy and efficiency are guaranteed if $\tau \geq \tau_{\min}$

Gray areas: accuracy and efficiency are not guaranteed

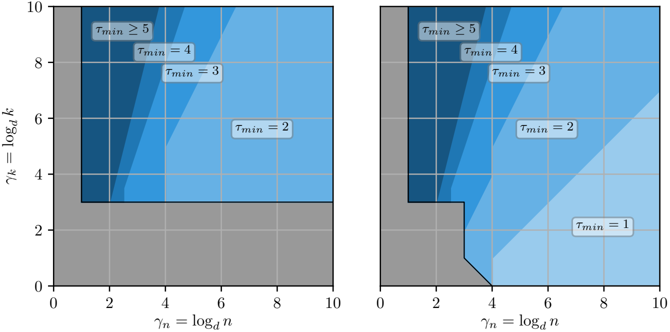
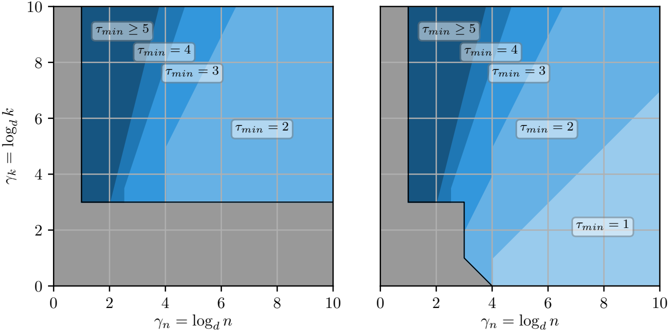


Illustration of main results for **linear models**:

Left: `k-grad` Right: `n+k-1-grad`

Blue areas: accuracy and efficiency are guaranteed if $\tau \geq \tau_{\min}$

Gray areas: accuracy and efficiency are not guaranteed



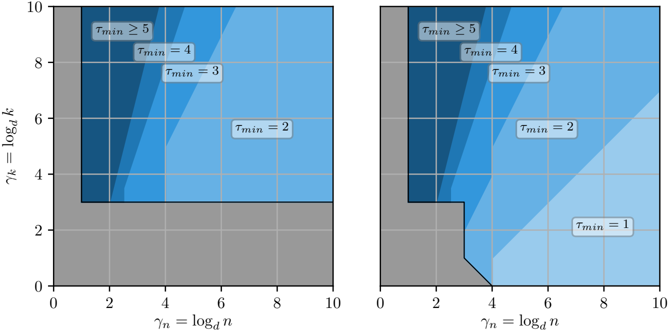
- $\tau_{\min} \nearrow$ logarithmically as $k \nearrow$, $n \searrow$ and $d \nearrow$

Illustration of main results for **linear models**:

Left: `k-grad` Right: `n+k-1-grad`

Blue areas: accuracy and efficiency are guaranteed if $\tau \geq \tau_{\min}$

Gray areas: accuracy and efficiency are not guaranteed



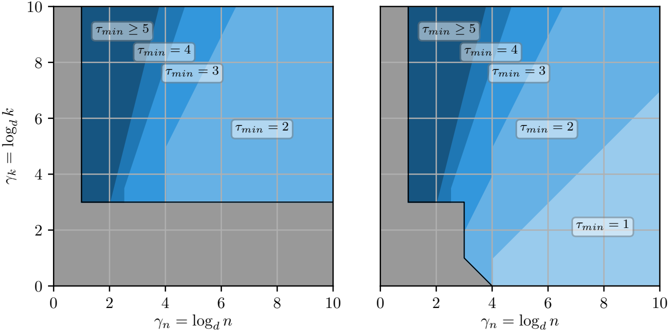
- $\tau_{\min} \nearrow$ logarithmically as $k \nearrow$, $n \searrow$ and $d \nearrow$
- $\tau_{\min,\texttt{n+k-1-grad}} \leq \tau_{\min,\texttt{k-grad}}$

Illustration of main results for **linear models**:

Left: `k-grad` Right: `n+k-1-grad`

Blue areas: accuracy and efficiency are guaranteed if $\tau \geq \tau_{\min}$

Gray areas: accuracy and efficiency are not guaranteed



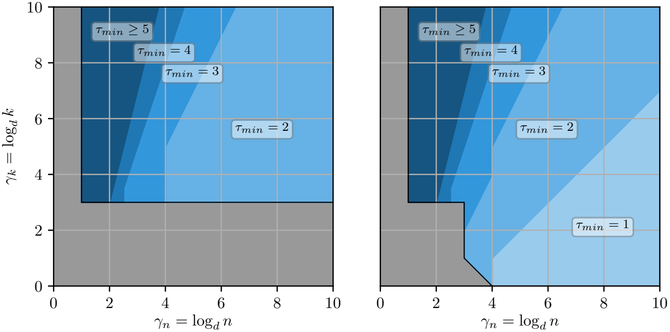
- $\tau_{\min} \nearrow$ logarithmically as $k \nearrow$, $n \searrow$ and $d \nearrow$
- $\tau_{\min,\texttt{n+k-1-grad}} \leq \tau_{\min,\texttt{k-grad}}$
- $\tau_{\min,\texttt{n+k-1-grad}} \geq 1, \quad \tau_{\min,\texttt{k-grad}} \geq 2$

Illustration of main results for **linear models**:

Left: `k-grad` Right: `n+k-1-grad`

Blue areas: accuracy and efficiency are guaranteed if $\tau \geq \tau_{\min}$

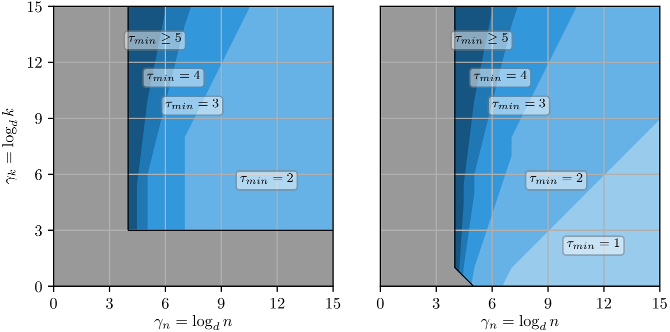
Gray areas: accuracy and efficiency are not guaranteed



- $\tau_{\min} \nearrow$ logarithmically as $k \nearrow$, $n \searrow$ and $d \nearrow$
- $\tau_{\min,\texttt{n+k-1-grad}} \leq \tau_{\min,\texttt{k-grad}}$
- $\tau_{\min,\texttt{n+k-1-grad}} \geq 1, \quad \tau_{\min,\texttt{k-grad}} \geq 2$
- γ_k has to be large for `k-grad`, but not for `n+k-1-grad`

Illustration of main results for **generalized linear models**

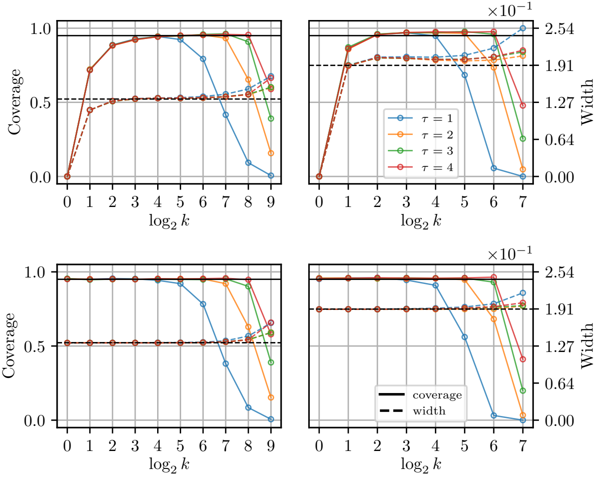
Left: `k-grad`   Right: `n+k-1-grad`



Simulations: logistic regression, $N = 2^{16}$

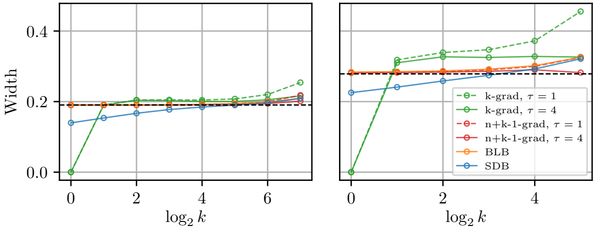
Top left: `k-grad`, $d = 2^3$ Top right: `k-grad`, $d = 2^5$

Bottom left: `n+k-1-grad`, $d = 2^3$ Bottom right: `n+k-1-grad`, $d = 2^5$



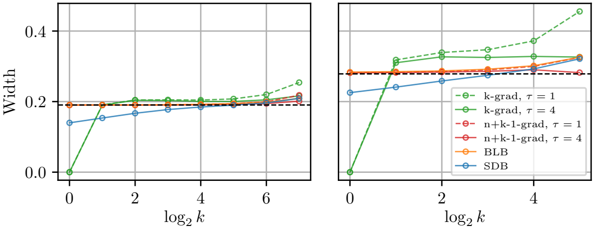
Comparisons to BLB and SDB:

- Width (logistic regression, left: $d = 2^5$, right: $d = 2^7$)



Comparisons to BLB and SDB:

- Width (logistic regression, left: $d = 2^5$, right: $d = 2^7$)



- Run time in seconds (linear regression, $d = 2^7$)

Methods	$k = 2^2$	$k = 2^6$	$k = 2^9$
`k-grad`	0.82	0.51	0.50
`n+k-1-grad`	1.49	0.67	0.64
SDB	3.44	3.83	12.66
BLB	981.17	842.50	1950.91

Extensions:

- To other models, e.g., graphical models
- To high-dimensional sparse models (in progress)

Thank you!