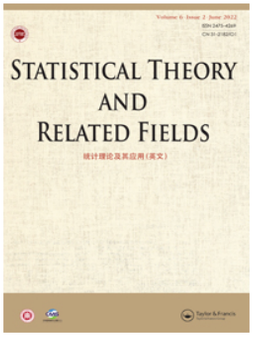

Statistical Theory and Related Fields

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tstf20

Discussion of: 'A review of distributed statistical inference'

Shaogao Lv & Xingcai Zhou

To cite this article: Shaogao Lv & Xingcai Zhou (2022) Discussion of: 'A review of distributed statistical inference', Statistical Theory and Related Fields, 6:2, 105-107, DOI: 10.1080/24754269.2021.2015868

To link to this article: https://doi.org/10.1080/24754269.2021.2015868

Published online: 28 Dec 2021.

STATISTICAL THEORY AND RELATED FIELDS
2022, VOL. 6, NO. 2, 105–107
https://doi.org/10.1080/24754269.2021.2015868

Discussion of: 'A review of distributed statistical inference'

Shaogao Lv and Xingcai Zhou

School of Statistics and Data Science, Nanjing Audit University, Nanjing, People's Republic of China

ARTICLE HISTORY Received 11 November 2021; Accepted 20 November 2021

First of all, we would like to congratulate Dr Gao et al. for their excellent paper, which provides a comprehensive overview of amounts of existing work on distributed estimation (learning). Different from related work Gu et al. (2019); Liu et al. (2021); Verbraeken et al. (2020) that focus on computing, storage and communication architecture, the current paper leverages how to guarantee statistical efficiency of a given distributed method from a statistical viewpoint.

In the following, we divide our discussion into three parts:

(1) The principle of variance-bias trade off.
(2) The specific difficulty of multivariate-nonparametric distributed learning.
(3) Robustness problems for various adversarial models.

1. The principle of variance-bias trade off

In the sequel, we start by general parametric models which can illustrate the interplay between the variance and the bias of an estimator.

Let $\beta_0 \in \mathbb{R}^d$ be the true parameter that we want to learn from all the observations. We denote by $\hat{\beta}$ an estimator generated by a given learning rule. According to the classical variance-bias decomposition, we informally have

$$\|\hat{\beta} - \beta_0\| \leq \text{Variance}(\hat{\beta}) + \text{Bias}(\hat{\beta}). \tag{1}$$

Variance describes how much a random variable differs from its expected value, while bias is the amount that a model's prediction differs from the target value. The correct balance of bias and variance is crucial to building statistical methodologies that create accurate results from their models.

Under the classical master/workers distributed framework, all the works with the number m produce local estimators based on their individual data, and then the master machine merges all the local estimators into a global estimator. In the case of linear Lasso estimation, the j-th worker runs the Lasso estimation to generate a local linear estimator ($\hat{\beta}_j$), and the master takes a simple average ($\bar{\beta} = \frac{1}{m}\sum_{j=1}^{m}\hat{\beta}_j$) to form a global linear estimator.

In such an estimation procedure, each $\hat{\beta}_j$ has a greater variance than the centralized Lasso estimator, mainly because of using a smaller sample size. It is known that the simple average can generate a global estimator with a small variance. In contrast to the variance term, the bias of the global estimator cannot be reduced significantly, since the simple average is an unbiased estimation. On one hand, this insight tells us that, the simple average strategy may be feasible for any less biased distributed estimation. On the other hand, when the bias term dominates the variance, the global estimator is no longer applicable under the distributed framework.

This paper follows the principle of variance-bias trade off and provides a full overview for classical parametric models, high-dimensional models and nonparametric models. In particular, it was shown in Zhang et al. (2013, Corollary 2) that, under appropriate regularity conditions for fixed dimensional parametric models,

$$\|\bar{\beta} - \beta_0\| = O_p\left(\frac{1}{N}\right),$$

provided that the local sample size satisfies $n \gg \sqrt{N}$, where $N = nm$ refers to the total number of sample size. This means that data splitting has no negative effect in a minimax senses and also just one-round communication among machines is enough in sense that the above divide and conquer strategy is quite communication efficient. However, these one-shot distributed learning suffer from the following three issues: (1) the size constraint $n \gg \sqrt{N}$ is often so restrictive that local machines cannot run such an amount data under some big data setting; (2) some interesting structures (e.g. sparsity) cannot be reserved well by the simple average; and (3) the achievable minimax rate is just one angle of measuring statistics, and some additional indexes such as asymptotic normality need to be considered as well in some situations.

To this end, this paper spends much spaces in reviewing a class of iterative distributed estimation,

proposed originally by Shamir et al. (2014). This iterative distrusted learning can be viewed as one Newton-type approximate optimization, which makes full use of local high-order information over the master machine and the first-order information over the workers. Instead of the simple average as mentioned above, the global estimator is formulated by an objective over the master machine. Despite of some little loss on communication rounds, this iterative distributed algorithm can yield satisfactory solutions under suitable conditions, e.g., sparsity reservation and the local sample size can be taken a far smaller value than $\sqrt{N}$. Subsequently, this idea motivates the rapid development of many modern distributed algorithms.

2. The specific difficulty of multivariate-nonparametric distributed learning

For genetic multivariate-nonparametric models, developing good distributed learning algorithm has to face with some particular challenges, since those classical spline-base tool or local constant/linear smooth methods may not be a good choice. Instead, reproducing kernel methods and neural network ones are two class of popular tools for high dimensional nonparametric models.

This paper also reviews some distributed learning based on kernel methods, such as two representative work in Lin et al. (2017) and Zhang et al. (2013). Let $K(\cdot,\cdot)$ be a given reproducing kernel and $\hat{f}_j$ be an nonparametric estimator at the j-th worker. Under kernel-based framework, $\hat{f}_j$ based on the local data $\{x_{ij}, y_{ij}\}_{i=1}^n$ has the form

$$\hat{f}_j(x) = \sum_{i=1}^{n} \alpha_{ij} K(x_{ij}, x), \quad \forall\, x \in \mathcal{X},$$

where $\alpha = (\alpha_{1j}, \ldots, \alpha_{nj}) \in \mathbb{R}^n$ is an estimated vector parameter. The global nonparametric estimator by the simple average is expressed as follows:

$$\bar{f}(x) = \frac{1}{m} \sum_{j=1}^{m} \hat{f}_j(x).$$

Although kernel-based distributed algorithm does not result in any computational issue, the communication issue in the prediction phase becomes a major barrier observed from $\bar{f}$, where N parameters and all samples need to be passed to the host. Alternatively, every input point x is sent to all local workers, and then send back the result of $\hat{f}_j(x)$. However, this requires that all the workers always run normally and specially are communication efficient at any time. In other words, developing kernel-based distributed learning still has a lot of space to be explored in the future.

We discuss several related distributed design using neural networks, which has been regarded as the state of art in modern artificial intelligence. However, we observe that neural network function is highly nonconvex, and so the simple average cannot make sure stationary consistency of the global estimator. On the other hand, it seems that the iterative approximate methods mentioned as above still work. Yet, the communication issues become a major bottleneck since typical neural networks contain too many weight parameters, even its size is far larger than N. Therefore, how to develop competitive distributed algorithms for neural network models is an interesting topic.

3. Robustness problems for various adversarial models

The paper gives some related important works about distributed learning in Section 4: principal component analysis, feature screening and Bootstrap. It presents a comprehensive overview on the three issues. In addition, robustness is also an important topic in the distributed inference, especially Byzantine-robust distributed learning. In distributed learning, data and computation generally come from individual computer/smartphones or worker machines of units. These devices can be easily reprogrammed and controlled by external attackers and thus behave adversarially. It may also due to crashes, faulty hardware, stalled computation or unreliable communication channels. Such worker machines are often unpredictable, which can incur major degradation in learning performance. Therefore, it is necessary to develop Byzantine-robust distributed learning algorithm to cope with adversarial attacks. For statisticians, there are some questions to consider: what kind of robust learning algorithm or the adversarial models can confront Byzantine failures, what is the best achievable statistical performance while being Byzantine-robust, how to design a learning algorithm that can achieve this performance and communication efficiency.

For adversarial models, various Byzantine-robust distributed algorithms have been considered recently, for example, Blanchard et al. (2017), Chen et al. (2017), Ghosh et al. (2020), Su and Vaidya (2016), Tu et al. (2021), Yin et al. (2018), and Zhou et al. (2021). Specially, Yin et al. (2018) proposed two robust distributed gradient descent algorithms based on coordinate-wise median and coordinate-wise trimmed mean and established order-optimal statistical error rates $\tilde{\mathcal{O}}\left(\frac{1}{\sqrt{n}}\alpha + \frac{1}{\sqrt{m}}\right)$ under mild conditions, where α is the fraction of Byzantine machines. It is a new statistical result under the robustness framework. Developing robust distributed learning algorithms cannot sacrifice the quality of learning, meanwhile pursue the best possible statistical accuracy in the presence of adversarial attacks. Robust distributed learning deserves attention and research in the era of big data.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

Shaogao Lv's work is partially supported by NSFC [Grant Number 11871277]; Xingcai Zhou's work is partially supported by NSFC [Grant Number 12171242] and National Social Science Foundation of China [Grant Number 19BTJ034].

References

Blanchard, P., El Mhamdi, E. M., Guerraoui, R., & Stainer, J. (2017). Machine learning with adversaries: Byzantine tolerant gradient descent. *Proceedings of the 31st international conference on neural information processing systems, Long Beach, CA, USA* (pp. 118–128).

Chen, Y., Su, L., & Xu, J. (2017). Distributed statistical machine learning in adversarial settings: Byzantine gradient descent. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, *46*(1), 1–25. https://doi.org/10.1145/3308809.3308857

Ghosh, A., Maity, R. K., Kadhe, S., Mazumdar, A., & Ramchandran, K. (2020). Communication-efficient and byzantine-robust distributed learning with error feedback. arXiv:1911.09721v3.

Gu, R., Yang, S., & Wu, F. (2019). Distributed machine learning on mobile devices: a survey. arXiv:1909.08329v1, 1–28.

Lin, S.-B., Guo, X., & Zhou, D.-X. (2017). Distributed learning with regularized least squares. *The Journal of Machine Learning Research*, *18*(49), 3202–3232.

Liu, J., Huang, J., Zhou, Y., Li, X., Ji, S., Xiong, H., & Dou, D. (2021). From distributed machine learning to federated learning: a survey. arXiv:2104.14362.

Shamir, O., Srebro, N., & Zhang, T. (2014). Communication-efficient distributed optimization using an approximate newton-type method. In *International Conference on Machine Learning* (pp. 1000–1008).

Su, L., & Vaidya, N. H. (2016). Fault-tolerant multi-agent optimization: optimal iterative distributed algorithms. In *Proceedings of the 2016 ACM Symposium on Principles of Distributed Computing* (pp. 425–434). Association for Computing Machinery.

Tu, J. Y., Liu, W. D., & Mao, X. J. (2021). Byzantine-robust distributed sparse learning for M-estimation. *Machine Learning*. https://doi.org/10.1007/s10994-021-06001-x.

Verbraeken, J., Wolting, M., Katzy, J., Kloppenburg, J., Verbelen, T., & J. S. Rellermeyer (2020). A survey on distributed machine learning. *ACM Computing Surveys*, *53*(2), 1–33. https://doi.org/10.1145/3377454

Yin, D., Chen, Y., Ramchandran, K., & Bartlett, P. (2018). Byzantine-robust distributed learning: Towards optimal statistical rates. *Proceedings of the 35th International Conference on Machine Learning, Stockholm, Sweden, PMLR 80* (pp. 5650–5659).

Zhang, Y., Duchi, J. C., & Wainwright, M. J. (2013). Communication-efficient algorithms for statistical optimization. *The Journal of Machine Learning Research*, *14*, 3321–3363.

Zhou, X. C., Chang, L., Xu, P. F., & Lv, S. G. (2021). Communication-efficient Byzantine-robust distributed learning with statistical guarantee. arXiv:2103.00373v1.