Communication-efficient Conformal Prediction for Distributed Datasets

Nery Riquelme-Granada nery.riquelmegranada@rhul.ac.uk
Zhiyuan Luo Zhiyuan.Luo@rhul.ac.uk
Royal Holloway University of London, Surrey, TW20 0EX, United Kingdom

Khuong An Nguyen k.a.nguyen@brighton.ac.uk
University of Brighton, East Sussex BN2 4GJ, United Kingdom

Editor: Ulf Johansson, Henrik Boström, Khuong An Nguyen, Zhiyuan Luo and Lars Carlsson

1. Extended Abstract

Coresets have been proven useful in accelerating the computation of inductive conformal predictors (ICP) when the training data becomes large in size. This work shows that coreset-based conformal predictors are not only computationally efficient in the centralised setting, but may also naturally be used in scenarios where the dataset of interest is inherently distributed over at least two machines.

This work follows the line of research started in Riquelme-Granada et al. (2019) and Riquelme-Granada et al. (2020b), where ICP is accelerated by using the idea of data compression. That is, instead of manipulating the classic conformal algorithm (Vovk et al. (2005)) to gain some computational acceleration, the idea is to use the ICP method *as is*, on a small dataset that acts as a *proxy* to the original dataset of interest. This proxy dataset is called a coreset (or core-set) (Feldman et al. (2013)), and it provably correctly approximates the original much-larger dataset, in a well-defined sense, with respect to a machine learning problem.

The combination of the coreset technique with that of ICP is known as *Coreset-based Inductive Conformal Prediction* (C-ICP), and it was shown that it may save a large amount of computing time while retaining the validity and efficiency of ICP (see experiments results in Riquelme-Granada et al. (2019)).

2. Learning over Distributed Data

This piece of research exploits an attractive aspect of coresets that was left unexplored in the previous works: its *aggregation* properties. Specifically, the coreset paradigm defines the *addition property* (Braverman et al. (2016)), which states that two coresets can be safely merged together, provided that they *represent* two non-overlapping sets of data. Formally, let $\mathcal{C}_1$ be a Δ-coreset for P_1 and $\mathcal{C}_2$ be a Δ-coreset for P_2, with $P_1 \cap P_2 := \emptyset$. Then, by the addition property, we have that $\mathcal{C}_1 \cup \mathcal{C}_2$ is a Δ-coreset for $P_1 \cup P_2$.

By the above addition property, it becomes natural to extend C-ICP to applications where the data are inherently distributed and one wants to apply conformal prediction *without* creating high communication overhead. Table 1 shows information regarding the validity and efficiency of C-ICP compared to ICP for the problem of Logistic Regression

Table 1: Validity (errors count) and efficiency (singleton count) measures for Covertype and Webspam dataset for ICP and C-ICP when data are stored in 4 machines. The overhead is the number of d-dimensional vectors sent over the network in order to learn the data. The coreset size for C-ICP is 1% of the proper training set.

# Nodes: 4			$\epsilon = 0.3$
Measure	**Dataset**	**ICP**	**C-ICP (1%)**
Errors (validity)	Covertype	34,886(0.3002)	34,929 (0.3005)
	Webspam	21,025 (0.3004)	20,991 (0.2999)
Singleton (efficiency)	Covertype	104,274 (0.8973)	104,365 (0.8981)
	Webspam	0 (0)	1 (0)
Overhead	Covertype	581,012 vectors	5,815 vectors (1% of data)
	Webspam	350,000 vectors	3,504 vectors (1% of data)

(Riquelme-Granada et al. (2020a)) using the well-known Covertype (581,012 data points) and Webspam (350,000 data points) datasets. Furthermore, the table shows the overhead generated by both methods when the data is distributed over 4 machines and they need to send it to a centralised aggregator to perform conformal prediction. C-ICP only requires to move 1% of the dataset over the network, while standalone ICP needs 100% of the data to be shuffled from the nodes to the centralised server. Also, as already established in Riquelme-Granada et al. (2019), C-ICP approximates closely the calibration and efficiency of ICP.

In summary, C-ICP does not only provide ICP with large acceleration in the centralised scenario, but also allows ICP to be communication-efficient when the data are partitioned across machines and the practitioner needs a quick method for applying conformal prediction.

References

Vladimir Braverman, Dan Feldman, and Harry Lang. New frameworks for offline and streaming coreset constructions. *CoRR*, abs/1612.00889, 2016. URL http://arxiv.org/abs/1612.00889.

Dan Feldman, Melanie Schmidt, and Christian Sohler. Turning big data into tiny data: Constant-size coresets for k-means, pca and projective clustering. In *Proceedings of the twenty-fourth annual ACM-SIAM symposium on Discrete algorithms*, pages 1434–1453. SIAM, 2013.

Nery Riquelme-Granada, Khuong Nguyen, and Zhiyuan Luo. Coreset-based conformal prediction for large-scale learning. In *Conformal and Probabilistic Prediction and Applications*, pages 142–162, 2019.

Nery Riquelme-Granada, Khuong An Nguyen, and Zhiyuan Luo. Coreset-based data compression for logistic regression. In *International Conference on Data Management Technologies and Applications*, pages 195–222. Springer, 2020a.

Nery Riquelme-Granada, Khuong An Nguyen, and Zhiyuan Luo. Fast probabilistic prediction for kernel svm via enclosing balls. In *Conformal and Probabilistic Prediction and Applications*, pages 189–208. PMLR, 2020b.

Vladimir Vovk, Alex Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*. Springer Science & Business Media, 2005.