Simulation and real data study of the paper *Accelerating Conformal Prediction via Approximate Leave-One-Out*

Jiachen Cong

1 Numerical experiments

1.1 Synthetic data

In this section, we conduct simulations to support the results in Theorem 3.3-3.4. We report the mean of coverage, mean of operation time, mean of interval length and mean of Jaccard index of original methods (labeled as “JK+” and “JK-minmax”, where “JK” stands for "Jackknife") and accelerated methods (labeled as “Fast JK+” and “Fast JK-minmax”). Recall that the Jaccard index between two sets $\mathcal{S}_1, \mathcal{S}_2$ is defined as

$$\mathcal{J}(\mathcal{S}_1, \mathcal{S}_2) = \frac{|\mathcal{S}_1 \cap \mathcal{S}_2|}{|\mathcal{S}_1 \cup \mathcal{S}_2|} \in [0, 1].$$

Values closer to 1 indicate more precise approximations. In Table 3 we report the Jaccard index between Jackknife+ and Fast Jackknife+, Jackknife-minmax and Fast Jackknife-minmax.

In our simulation, we set $\alpha = 0.1$, which means our target coverage level is $1 - \alpha = 0.9$ (in the jackknife+ case, as shown in [Barber et al., 2021], the coverage level is $1 - 2\alpha = 0.8$). We use training sample size $n = 100$, test sample size $n_{test} = 100$, and repeat the experiment at each dimension $p = 50, 100, 200$, with i.i.d. data points $(\boldsymbol{x}_i, y_i)$ generated as $\boldsymbol{x}_i \sim \mathcal{N}(0, I_p/\sqrt{p})$ and $y_i|\boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{x}_i^\top \theta, 1)$. The true coefficient vector θ is randomly generated from a standard normal distribution . For the Ridge regression model, we define the loss function as $\ell(y, \boldsymbol{x}^\top \theta) = (y - \boldsymbol{x}^\top \theta)^2/2$ and the regularization term as $r(\theta) = \frac{1}{2} r_0(\beta) + \frac{1}{2}\theta^\top \theta$, where Pseudo-Huber regularizer (this setting allows our simulation results to be compared with those reported in [Clarté and Zdeborová, 2024]) $r_0(\beta) = \sum_{j=1}^{p} 4(\sqrt{1 + \frac{\beta_j^2}{4}} - 1)$, with the ridge parameter $\lambda = 0.1, 1$, separately. To obtain stable results, we repeat the procedures above for 50 iterations and report the averaged outcomes.

Table 1: Comparison of JK+ and Fast JK+.

Parameters	Model	Coverage	Time(s)	Length
(n=100, p=50, λ=1)	JK+	0.880	0.075	3.808
(n=100, p=100, λ=1)	JK+	0.892	0.109	4.189
(n=100, p=200, λ=1)	JK+	0.897	0.185	4.409
(n=100, p=50, λ=0.1)	JK+	0.872	0.091	4.135
(n=100, p=100, λ=0.1)	JK+	0.888	0.172	4.729
(n=100, p=200, λ=0.1)	JK+	0.889	0.354	4.662
(n=100, p=50, λ=1)	Fast JK+	0.880	0.004	3.809
(n=100, p=100, λ=1)	Fast JK+	0.893	0.008	4.190
(n=100, p=200, λ=1)	Fast JK+	0.897	0.043	4.410
(n=100, p=50, λ=0.1)	Fast JK+	0.872	0.005	4.135
(n=100, p=100, λ=0.1)	Fast JK+	0.888	0.009	4.734
(n=100, p=200, λ=0.1)	Fast JK+	0.890	0.044	4.672

Table 2: Comparison of JK-minmax and Fast JK-minmax.

Parameters	**Model**	**Coverage**	**Time(s)**	**Length**
(n=100, p=50, λ=1)	JK-minmax	0.911	0.086	4.187
(n=100, p=100, λ=1)	JK-minmax	0.920	0.103	4.549
(n=100, p=200, λ=1)	JK-minmax	0.918	0.189	4.718
(n=100, p=50, λ=0.1)	JK-minmax	0.926	0.094	4.925
(n=100, p=100, λ=0.1)	JK-minmax	0.942	0.164	5.574
(n=100, p=200, λ=0.1)	JK-minmax	0.937	0.388	5.387
(n=100, p=50, λ=1)	Fast JK-minmax	0.911	0.004	4.188
(n=100, p=100, λ=1)	Fast JK-minmax	0.920	0.008	4.550
(n=100, p=200, λ=1)	Fast JK-minmax	0.918	0.042	4.719
(n=100, p=50, λ=0.1)	Fast JK-minmax	0.926	0.004	4.925
(n=100, p=100, λ=0.1)	Fast JK-minmax	0.943	0.008	5.750
(n=100, p=200, λ=0.1)	Fast JK-minmax	0.937	0.047	5.400

Table 3: Prediction-interval Overlap (Jaccard Index) of Fast JK+ & JK+ and Fast JK-minmax & JK-minmax.

Parameters	**Fast JK+ & JK+**	**Fast JK-minmax & JK-minmax**
(n=100, p=50, λ=1)	0.9997	0.9997
(n=100, p=100, λ=1)	0.9997	0.9996
(n=100, p=200, λ=1)	0.9998	0.9997
(n=100, p=50, λ=0.1)	0.9996	0.9996
(n=100, p=100, λ=0.1)	0.9983	0.9982
(n=100, p=200, λ=0.1)	0.9978	0.9976

Table 1 and Table 2 present the mean of coverage, mean of operation time and mean of interval length of Jackknife+, Fast Jackknife+, Jackknife-minmax and Fast Jackknife-minmax, respectively. Table 3 shows the similarity, measured by Jaccard Index, between the prediction intervals constructed by the accelerated methods and their corresponding original methods.

We find that our accelerated methods substantially reduce the average computational time while maintaining coverage, and without significantly altering the length (efficiency) of the prediction intervals. The prediction intervals constructed by the accelerated methods exhibit a high degree of similarity to those from the original methods. We can observe that, in most cases, a smaller value of λ tends to result in a wider interval, and the interval length also increases as p grows. Moreover, our methods exhibit more significant acceleration when the dimensionality of the covariates is higher.

Compared with the synthetic-data simulation results reported in Table 1 of Clarté and Zdeborová [2024], both of our acceleration methods achieve higher coverage than Taylor-AMP, while also providing more efficient prediction (in terms of shorter average interval length) compared with Taylor-AMP, SCP [Vovk et al., 2005] and CQP [Romano et al., 2019]. With respect to Table 2 of Clarté and Zdeborová [2024], fast Jackknife+ and fast Jackknife-minmax exhibit higher Jaccard similarity to exact LOO than Taylor-AMP and SCP. Under Gaussian settings (Table 3 of Clarté and Zdeborová [2024]), our methods deliver more efficient prediction than Bayes posterior and FCP combined with Taylor-AMP. In comparison with Table 4 of Clarté and Zdeborová [2024], both of our accelerated procedures achieve higher prediction efficiency and higher coverage than Taylor-AMP and approximate homotopy [Ndiaye and Takeuchi, 2019], while exhibiting comparable computation time. Furthermore, our framework systematically explores multiple dimensions ($p = 50, 100, 200$), corresponding to different n/p ratios, and two regularization strengths ($\lambda = 0.1, 1$), demonstrating consistent performance across regimes.

These improvements can be attributed to two main factors. First, unlike Clarté and Zdeborová [2024], which assumes i.i.d. features with a diagonal covariance matrix, our framework accommodates a general covariance structure Σ. This relaxation is more realistic in practice and ensures that the resulting ALO characterization remains accurate even when the features are correlated, thereby preventing the deterioration observed for AMP and Taylor-AMP under non-isotropic designs. Second, while AMP-based methods rely on an ALO heuristic whose approximation error is not rigorously controlled, our approach benefits from a explicit and tighter ALO error bound derived via the Newton step and Woodbury identity. This leads to more accurate leave-one-out predictions, which in turn improves the quality of the prediction intervals and enhances efficiency.

1.2 Application to Real Data

In this section, we compare the performance of the fast Jackknife+ with the original Jackknife+, and the fast Jackknife-minmax with the original Jackknife-minmax on real data. We use for this two datasets-the Concrete Compressive Strength Dataset [Yeh, 1998] and the Energy Efficiency Dataset [Tsanas and Xifara, 2012].We validate that our methods provide the correct coverage and equally efficiency with faster speed.

In particular, from Table 4 and Table 5, we observe that our proposed methods, Fast Jackknife+ and Fast Jackknife-minmax, substantially accelerate their corresponding baseline counterparts while maintaining the predictive accuracy and efficiency.

Table 4: Performance the Concrete Compressive Strength Dataset

Method	**Coverage Rate**	**Operation Time (s)**	**Average Interval Length**
Jackknife+	0.9660	1.7275	37.0917
Fast Jackknife+	0.9660	0.0651	37.0851
Jackknife-minmax	0.9709	1.5735	37.8552
Fast Jackknife-minmax	0.9709	0.0414	37.8671

Table 5: Performance on the Energy Efficiency Dataset

Method	**Coverage Rate**	**Operation Time (s)**	**Average Interval Length**
Jackknife+	0.9351	1.6567	12.0404
Fast Jackknife+	0.9351	0.0509	12.0403
Jackknife-minmax	0.9481	1.0640	12.2182
Fast Jackknife-minmax	0.9481	0.0371	12.2181

References

Rina Foygel Barber, Emmanuel J Candes, Aaditya Ramdas, and Ryan J Tibshirani. Predictive inference with the jackknife+. *The Annals of Statistics*, 49(1):486–507, 2021. https://doi.org/10.1214/20-AOS1965.

Lucas Clarté and Lenka Zdeborová. Building conformal prediction intervals with approximate message passing, 2024. URL https://arxiv.org/abs/2410.16493.

Eugene Ndiaye and Ichiro Takeuchi. Computing full conformal prediction set with approximate homotopy, 2019. URL https://arxiv.org/abs/1909.09365.

Yaniv Romano, Evan Patterson, and Emmanuel Candes. Conformalized quantile regression. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Athanasios Tsanas and Angeliki Xifara. Energy Efficiency. UCI Machine Learning Repository, 2012. DOI: https://doi.org/10.24432/C51307.

Vladimir Vovk, Alex Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. 01 2005. doi:10.1007/b106715.

I-Cheng Yeh. Concrete Compressive Strength. UCI Machine Learning Repository, 1998. DOI: https://doi.org/10.24432/C5PK67.