DSGD-AC: Controlled consensus errors improve generalization in decentralized training

Zesen Wang [1] Mikael Johansson [1]

[1]KTH Royal Institute of Technology, Sweden

Background

Decentralized training avoids global synchronization and can greatly reduce communication overhead, but is often believed to generalize worse than centralized algorithms. From convex problems, the common intuition is that **consensus errors** — the differences between local models and global average — **are harmful noise that should be minimized.**

Decentralized SGD

The update of DSGD:

$$x_i^{(t+1)} = x_i^{(t)} - \alpha^{(t)} g_i^{(t)} + \sum_{j\in\mathcal{N}(i)} W_{ij}(x_j^{(t)} - x_i^{(t)})$$

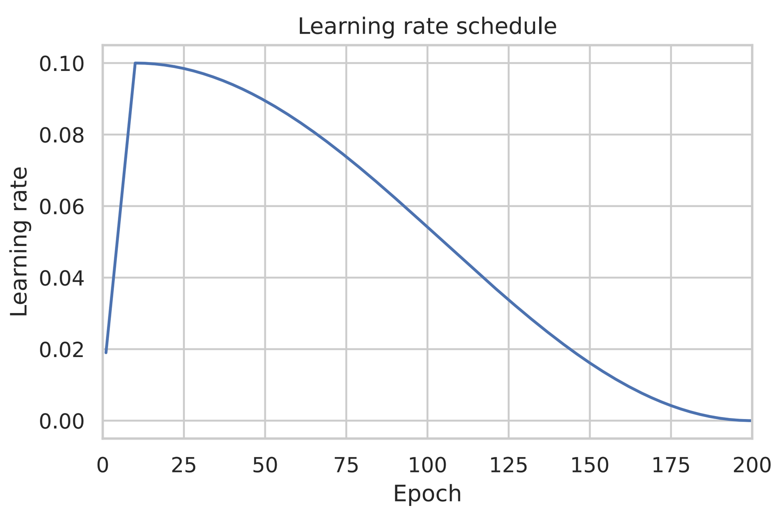
is optimizing the global objective $J^{(t)}(x_1^{(t)}, \cdots, x_n^{(t)})$:

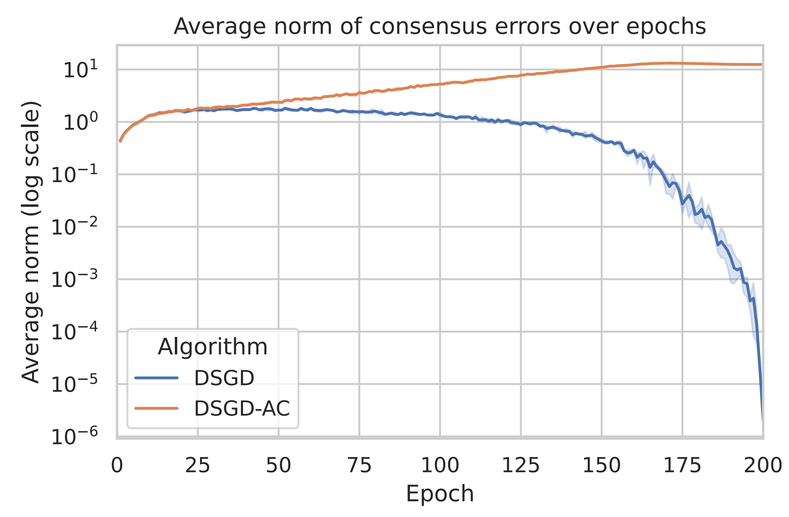
$$= \underbrace{\sum_{i=1}^{n} F_i(\bar{x}^{(t)})}_{\text{objective on deployed model}} + \underbrace{\sum_{i=1}^{n}[F_i(x_i^{(t)}) - F_i(\bar{x}^{(t)})]}_{\text{sharpness}}$$

$$+ \underbrace{\frac{1}{2\alpha^{(t)}} \sum_{i,j\in[n]} W_{ij}\|x_i^{(t)} - x_j^{(t)}\|^2}_{\text{consensus regularizer}}$$

However, the consensus errors vanish with diminishing step sizes, which voids the potential sharpness regularization.

$$e_i^{(t)} := x_i^{(t)} - \frac{1}{n}\sum_{j=1}^{n} x_j^{(t)} = x_i^{(t)} - \bar{x}^{(t)}$$





Contributions

- **Insight (alignment structure)** Consensus errors tend to align with the dominant Hessian subspace.
- **Insight (curvature regularizer)** Consensus errors can be viewed as a curvature regularizer instead of noises.
- **Algorithm (DSGD-AC)** A simple yet effective algorithm that *intentionally maintains non-vanishing consensus errors*, exploit the regularization, and improves generalization.

DSGD-AC: Decentralized SGD with adaptive consensus

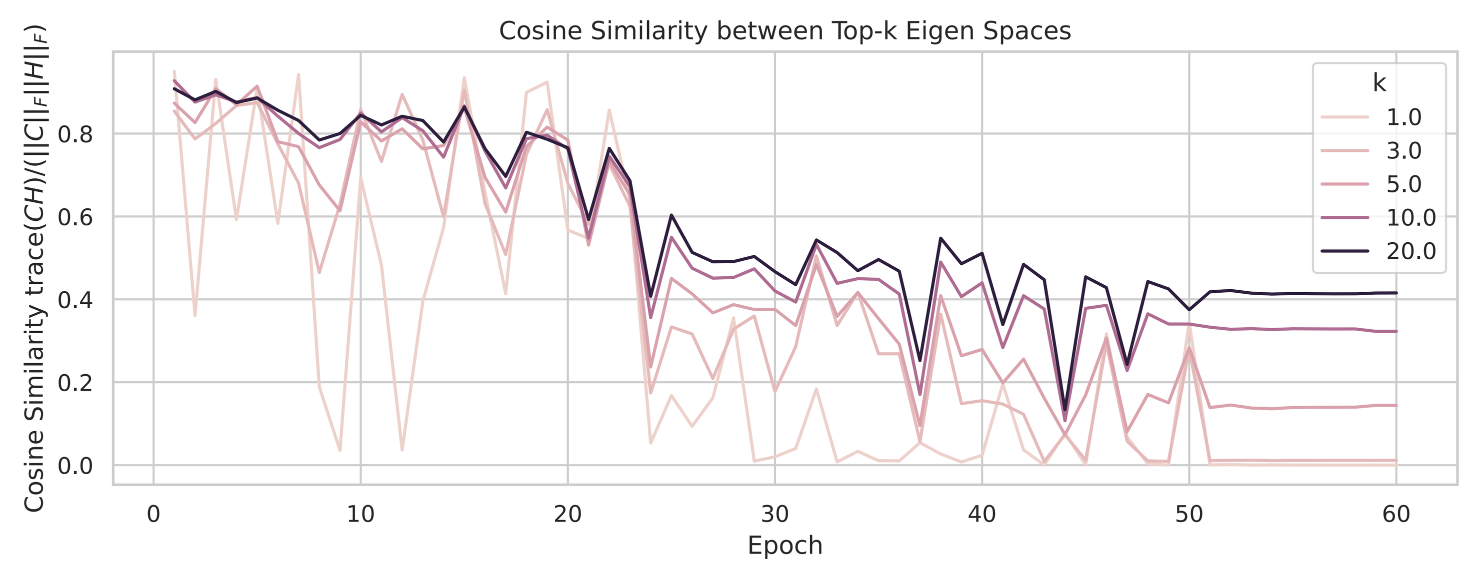
Require: Dataset (D), the number of workers (n), the number of epoch (E), the number of batches per epoch (T), intialization ($x^{(0)}$), and a hyperparameter p ($p \geq 2$).

Ensure: Deployed model $\bar{x} = \frac{1}{n}\sum_{j=1}^{n} x_j^{(TE)}$

1: $x_1^{(0)} = x_2^{(0)} = \cdots = x_n^{(0)} = x^{(0)}$
2: **for** $t = 0$ to $TE - 1$ **do**
3: $\quad g_i^{(t)} = \nabla f(x_i^{(t)}; \xi_i^{(t)})$
4: $\quad \gamma^{(t)} = [\alpha^{(t)}/\alpha_{\max}]^p$
5: $\quad x_i^{(t+1)} = x_i^{(t)} - \alpha^{(t)} g_i^{(t)} + \gamma^{(t)} \sum_{j\in\mathcal{N}(i)} W_{ij}(x_j^{(t)} - x_i^{(t)})$
6: **end for**

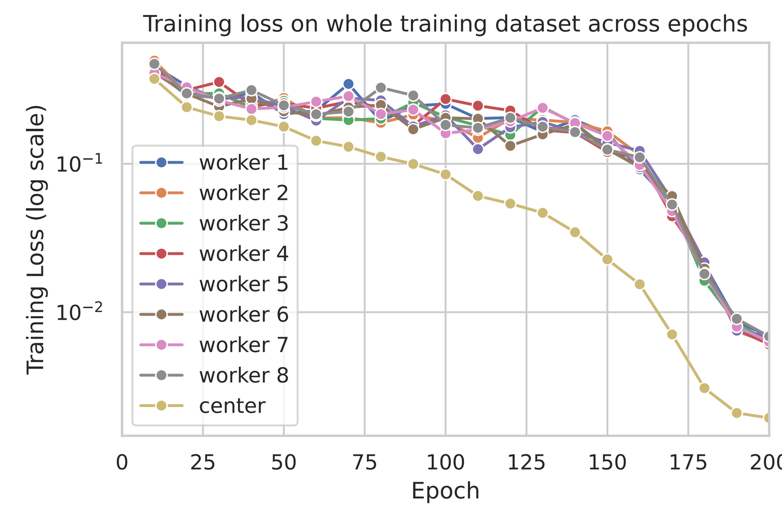
Insightful observations

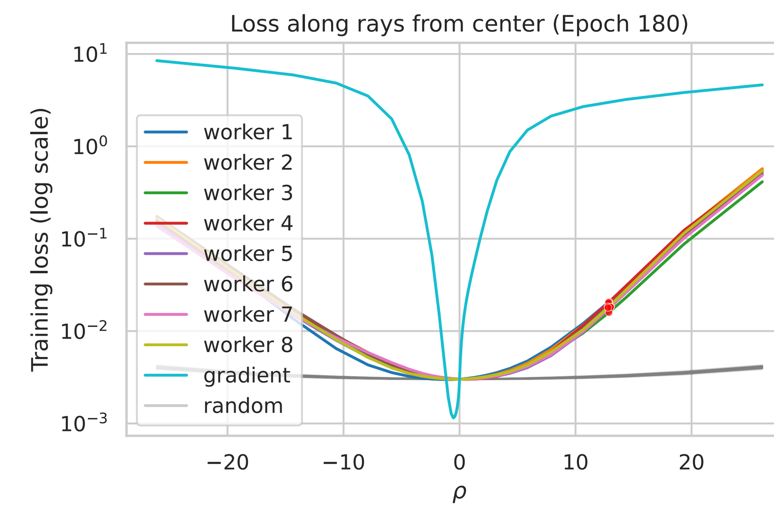
Correlation between Hessian and covariance matrix of stochastic gradients



There exists a non-trivial level of correlation between the covariance matrix C of the gradient noise and Hessian H (though decreasing).

Hessian-alignment structure in consensus errors





Left: Loss at the average center is significantly smaller than the losses at the local models.

Right: Losses along the directions of consensus errors exhibit significantly higher curvature than along random directions.

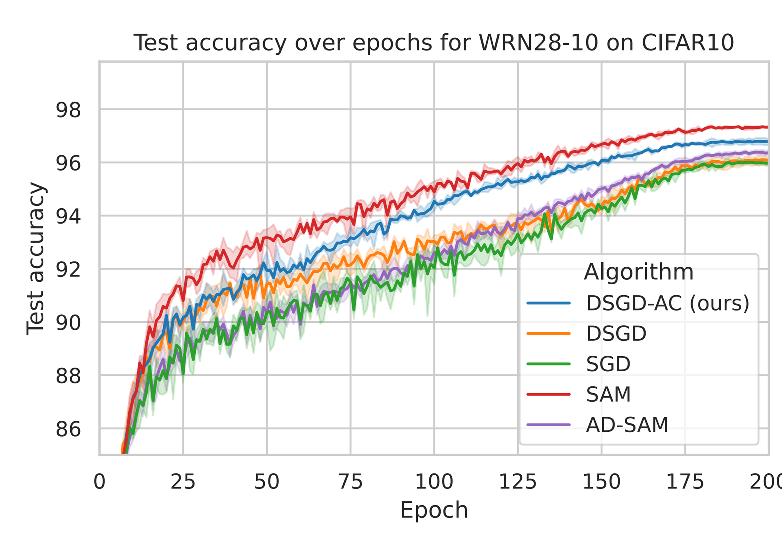
Consensus errors as curvature regularizer

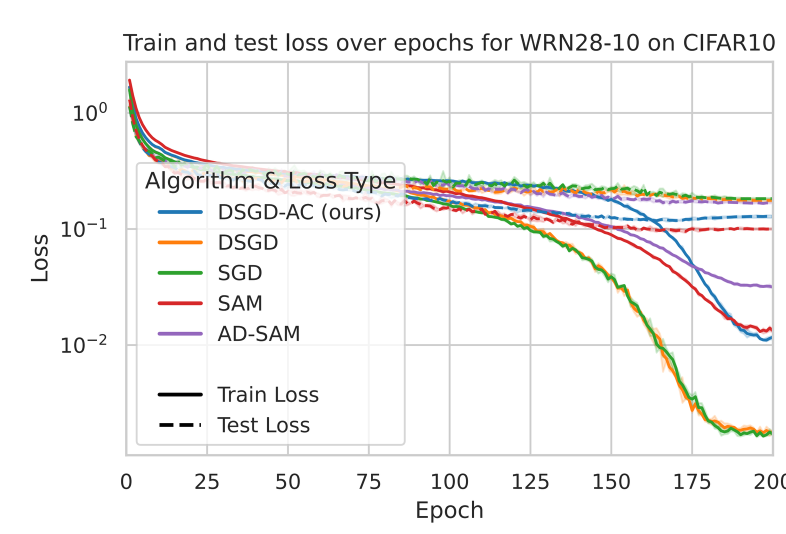
With i.i.d. data distributions and $\Sigma^{(t)}$ as the consensus error covariance, we can have the sum of the local objectives as

$$\frac{1}{n}\sum_{i=1}^{n} F_i(x_i^{(t)}) = F(\bar{x}^{(t)}) + \frac{1}{2}\operatorname{tr}(H\Sigma^{(t)}) + O\big((\operatorname{tr}\Sigma^{(t)})^{3/2}\big),$$

Thus, DSGD-AC can be interpreted as minimizing the central loss $F(x^{(t)})$ plus a Hessian-weighted disagreement penalty.

Numerical results





Algorithm	Test Acc. (%) ↑	Test Loss ↓	Mean Top-1 Eigenvalue ↓	Computation ↓
DSGD	96.07 ± 0.13	0.176 ± 0.005	22.4360 ± 3.9916	1x
SGD	95.96 ± 0.14	0.182 ± 0.004	16.8485 ± 0.3251	1x
DSGD-AC	96.77 ± 0.11	0.128 ± 0.003	8.9693 ± 0.3514	1x
AD-SAM [1]	96.37 ± 0.11	0.168 ± 0.002	24.9059 ± 1.6212	1x
SAM [2]	**97.33 ± 0.04**	**0.100 ± 0.002**	**0.3523 ± 0.0312**	2x

References

[1] Bisla, Devansh, Jing Wang, and Anna Choromanska. "Low-pass filtering sgd for recovering flat optima in the deep learning optimization landscape." *International Conference on Artificial Intelligence and Statistics*. PMLR, 2022.

[2] Foret, Pierre, et al. "Sharpness-aware minimization for efficiently improving generalization." *arXiv preprint arXiv:2010.01412* (2020).