Carnegie Mellon University

Boosted CVaR Classification

Runtian Zhai Chen Dan Arun Sai Suggala Zico Kolter Pradeep Ravikumar

TL;DR

We train ensemble models with high tail performance using a novel Boosting framework that boosts an unfair learner to a fair learner and demonstrate its efficiency.

CVaR: Conditional Value at Risk

Fair Models with High Tail Performance

A fair model should have high **tail performance**, *i.e.* high performance on samples where the performance is the lowest. It can be measured by the α**-CVaR loss**:

$$\text{CVaR}_\alpha^\ell(F) = \max_{\boldsymbol{w}\in\Delta_n, \boldsymbol{w}\preccurlyeq(\alpha n)^{-1}} \sum_{i\in[n]} w_i \ell(F(\boldsymbol{x}_i), y_i) \tag{1}$$

where $\{(\boldsymbol{x}_i, y_i)\}_{i\in[n]}$ is the training set, $\alpha \in (0,1)$, $\ell(\hat{y}, y)$ is a loss function and Δ_n is the unit simplex in $\mathbb{R}^n$.

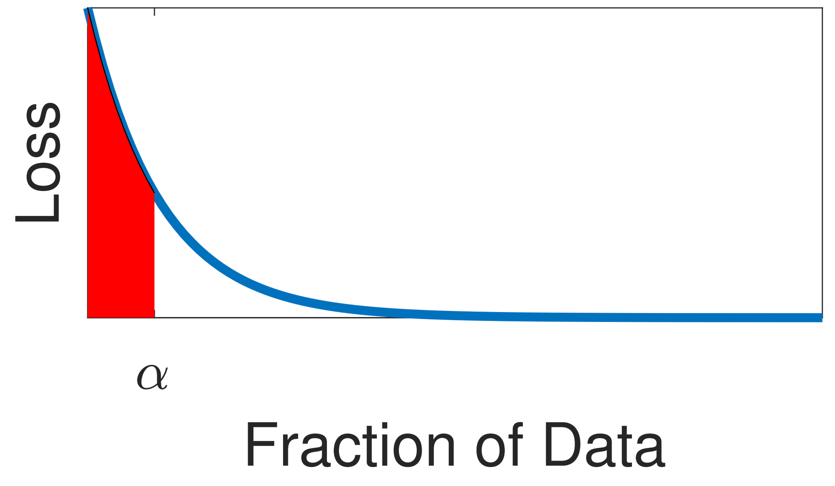


Figure 1: α-CVaR loss (red region) is the average loss over the worst α fraction of the data.

ERM Achieves Lowest CVaR in Classification

In classification tasks, we evaluate the models with the zero-one loss $\ell_{0/1}(\hat{y}, y) = \mathbf{1}_{\{\hat{y}\neq y\}}$. We can prove that

$$\text{CVaR}_\alpha^{\ell_{0/1}}(F) = \min\{1, \alpha^{-1}\hat{\mathcal{R}}^{\ell_{0/1}}(F)\} \tag{2}$$

where $\hat{\mathcal{R}}^{\ell_{0/1}}$ is the empirical zero-one loss which ERM minimizes. Thus, ERM also minimizes the CVaR loss, meaning that using CVaR has no gain compared to ERM.

Training Ensemble Models via Boosting

Eqn. (2) only holds for deterministic models. For randomized models whose outputs are distributions over the output space, using CVaR can improve the tail performance.

α-CVaR is Equivalent to α-LPBoost

α-LPBoost refers to the following primal/dual LP:

Dual:

$$\min_{\boldsymbol{w},\gamma} \gamma \quad \text{s.t. } \langle \boldsymbol{w}, \ell^s\rangle \geq 1-\gamma;\ \forall s\in[t] \quad \boldsymbol{w}\in\Delta_n, \boldsymbol{w}\preccurlyeq \frac{1}{\alpha n}$$

Primal:

$$\max_{\lambda,\rho} \rho - \frac{1}{\alpha n}\sum_{i=1}^n (\rho - 1 + \sum_{s=1}^t \lambda_s \ell_i^s)_+ \quad \text{s.t. } \lambda\in\Delta_t$$

where ℓ_i^s is the loss of f^s on sample i. We can show that:

Theorem. The optimal objective value of this LP (same for primal/dual) is equal to $1 - \min_{\lambda\in\Delta_t} \text{CVaR}_\alpha^{\ell_{0/1}}(F)$, where F is the ensemble model consisting of $f^1, \cdots, f^t$ and λ.

Thus, minimizing the CVaR loss is equivalent to maximizing the optimal objective of α-LPBoost, which is equivalent to the problem of **boosting an unfair learner**.

α-AdaLPBoost: Improving Efficiency with AdaBoost

In α-LPBoost, for each different α we need a different set of base models $\{f^t\}_{t\in[T]}$. However, in many real tasks we need to tune α frequently, which would be very inefficient.

To improve efficiency, we pick $\boldsymbol{w}^t$ with AdaBoost:

$$w_i^{t+1} \propto \exp(\eta \sum_{s=1}^t \ell_i^s) \tag{3}$$

Then, for all α we have the same set of base models, which makes tuning α much more efficient.

Boosting an Unfair Learner: Framework

We have an **unfair learner** $\mathcal{L}$ that outputs models with high average performance but low tail performance.

For $t = 1, \cdots, T$ do

1. Pick a sample weight vector $\boldsymbol{w}^t \in \Delta_n$ and feed it to $\mathcal{L}$.
2. Receive a base model f^t from $\mathcal{L}$ whose weighted average 0/1 loss *w.r.t.* $\boldsymbol{w}^t \leq g$ for constant $g\in(0,1)$.

Finally, pick a model weight vector $\lambda \in \Delta_T$.

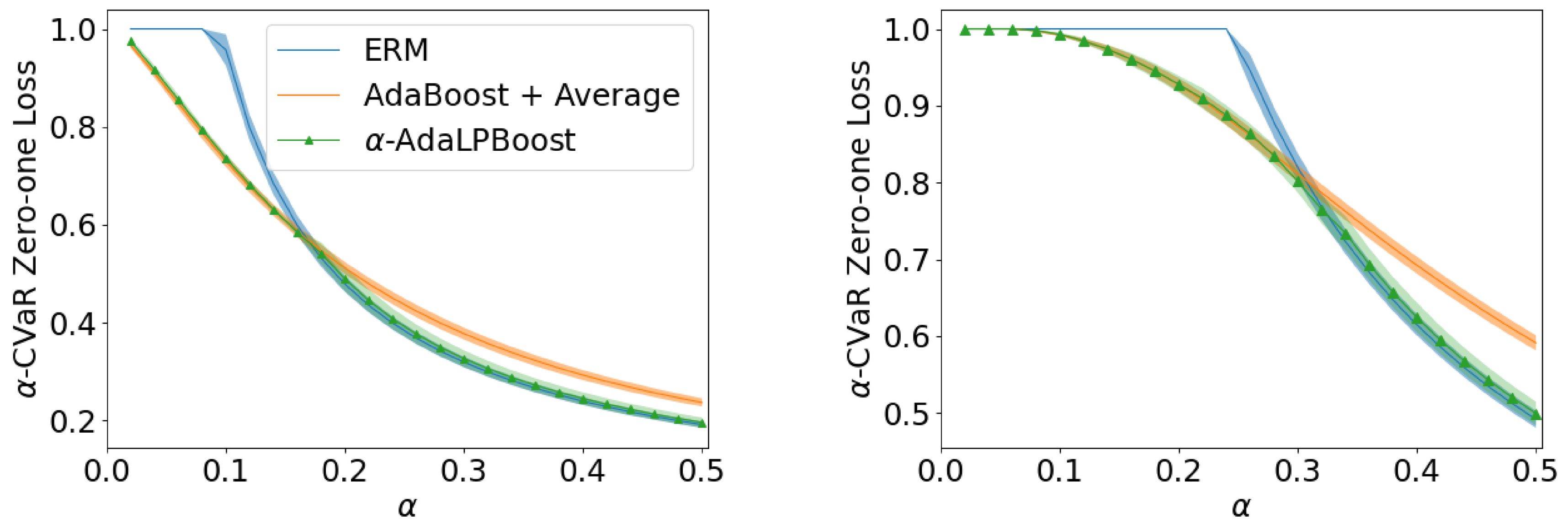
At inference time, first randomly sample a base model f^t according to λ, and then predict with f^t.

Theoretical and Empirical Results

Theorem. For any $\delta > 0$, and for $T = O(\frac{\log n}{\delta^2})$, the training α-CVaR zero-one loss of the ensemble model given by α-AdaLPBoost is at most $g + \delta$ if we set $\eta = \sqrt{8\log n / T}$.

In the worst case, the zero-one loss of the ensemble model is at least g. This theorem shows that α-AdaLPBoost can get as close to this lower bound as possible.

We also empirically show the efficiency of the framework:



(a) CIFAR-10 (b) CIFAR-100

NeurIPS 2021: Thirty-fifth Conference on Neural Information Processing Systems

rzhai@cmu.edu