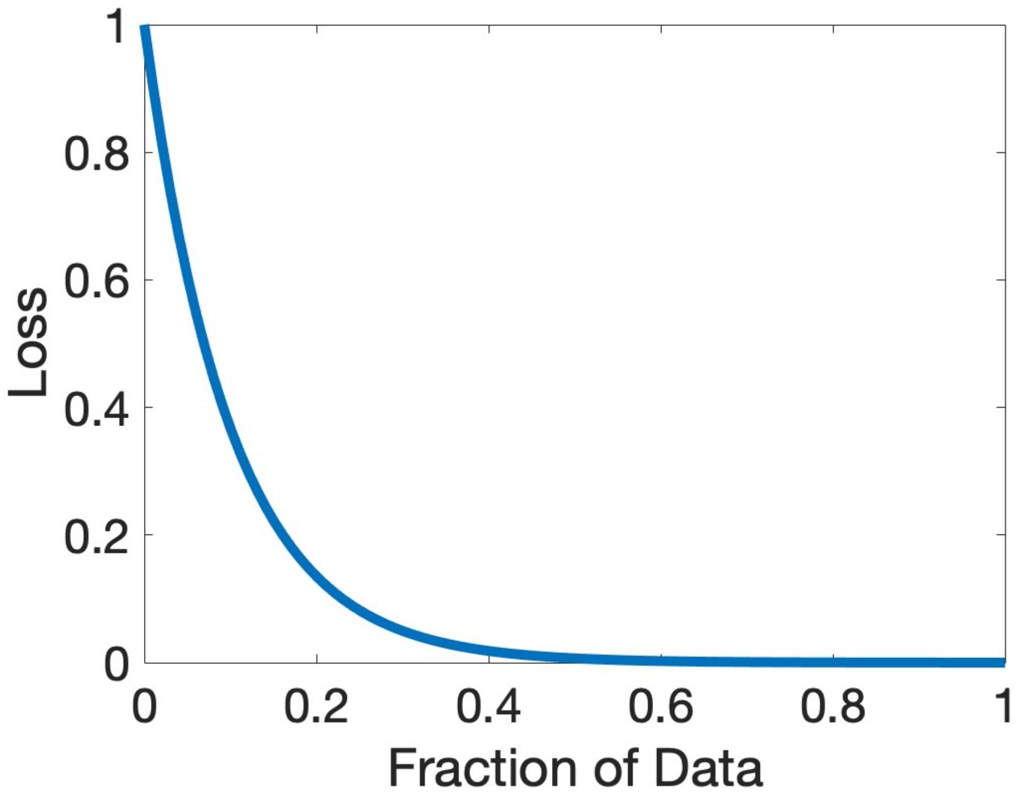
Carnegie Mellon University

Boosted CVaR Classification

NEURIPS 2021

Runtian Zhai, Chen Dan, Arun Sai Suggala,
Zico Kolter, Pradeep Ravikumar

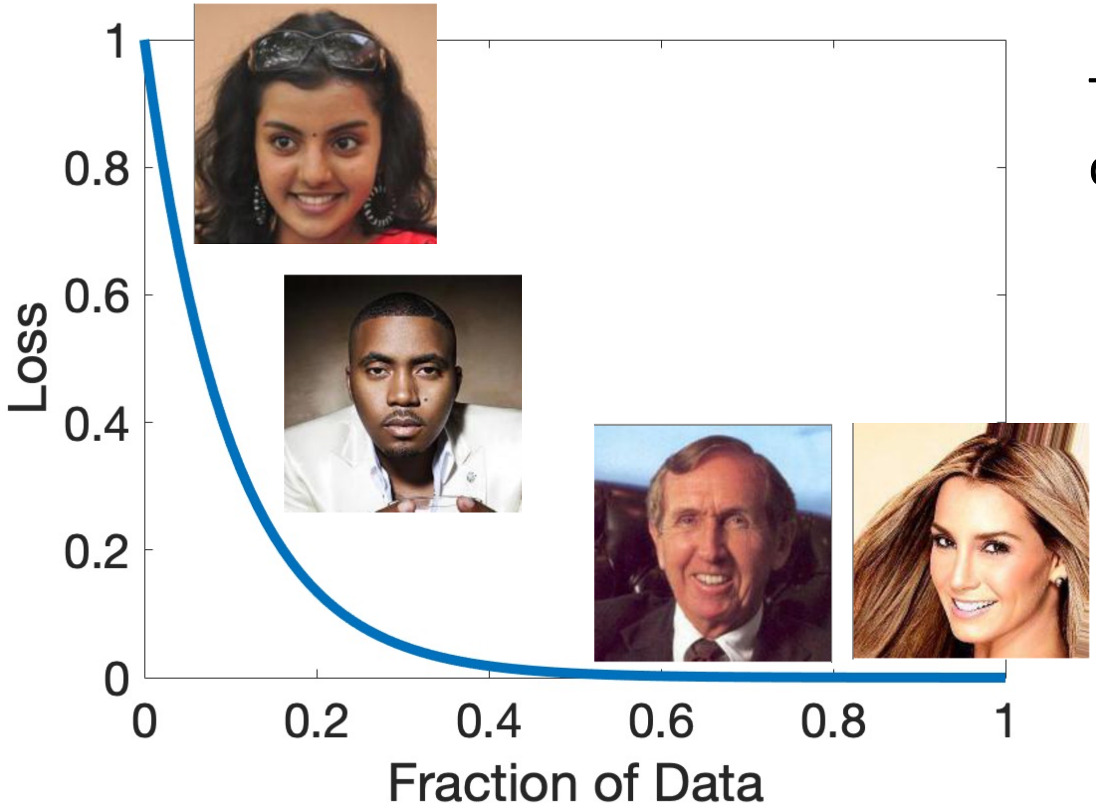
Improving Tail Performance for Fairness



A normally trained model usually has very high loss on a fraction of the data.

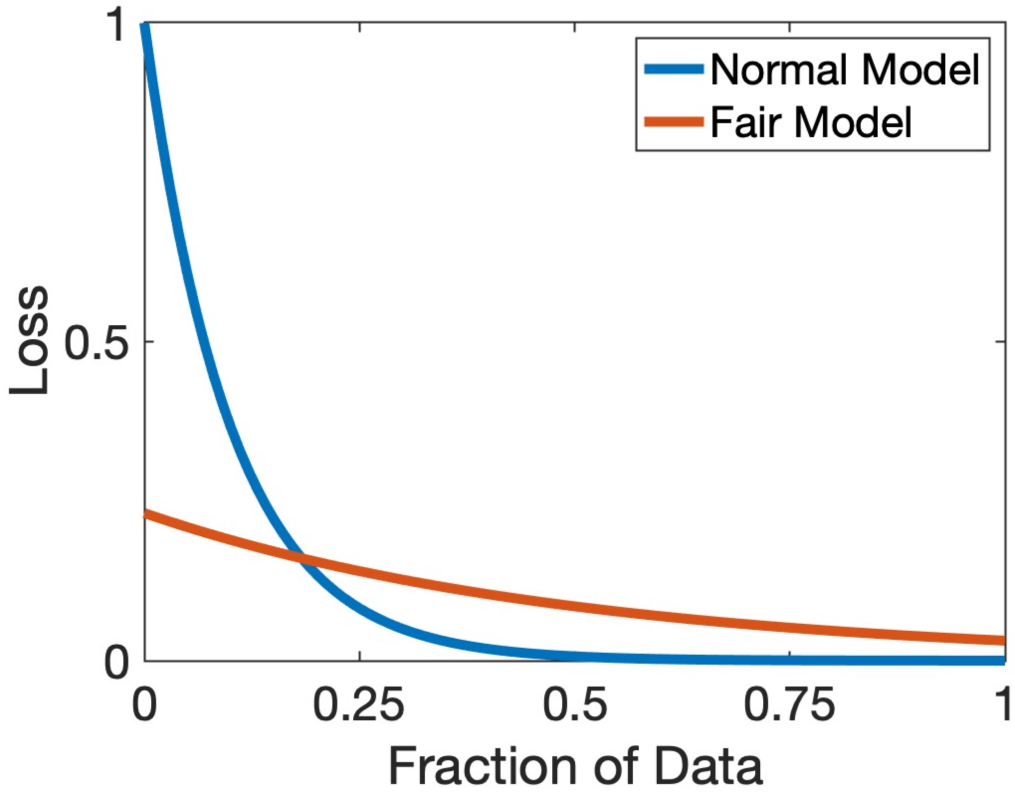
Low Tail Performance

Improving Tail Performance for Fairness

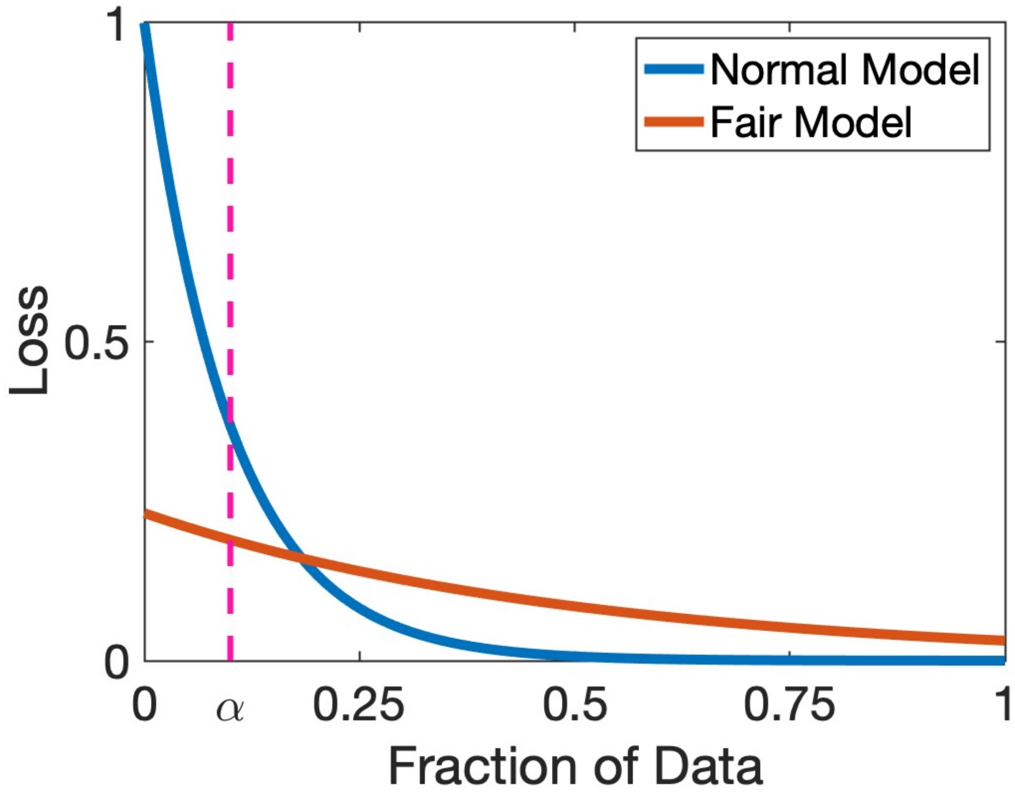


The tail usually corresponds to certain minority groups.

Improving Tail Performance for Fairness



CVaR Loss



α-CVaR Loss: Average loss over the worst α fraction of the data.

Can we minimize the α-CVaR loss to train a fair model?

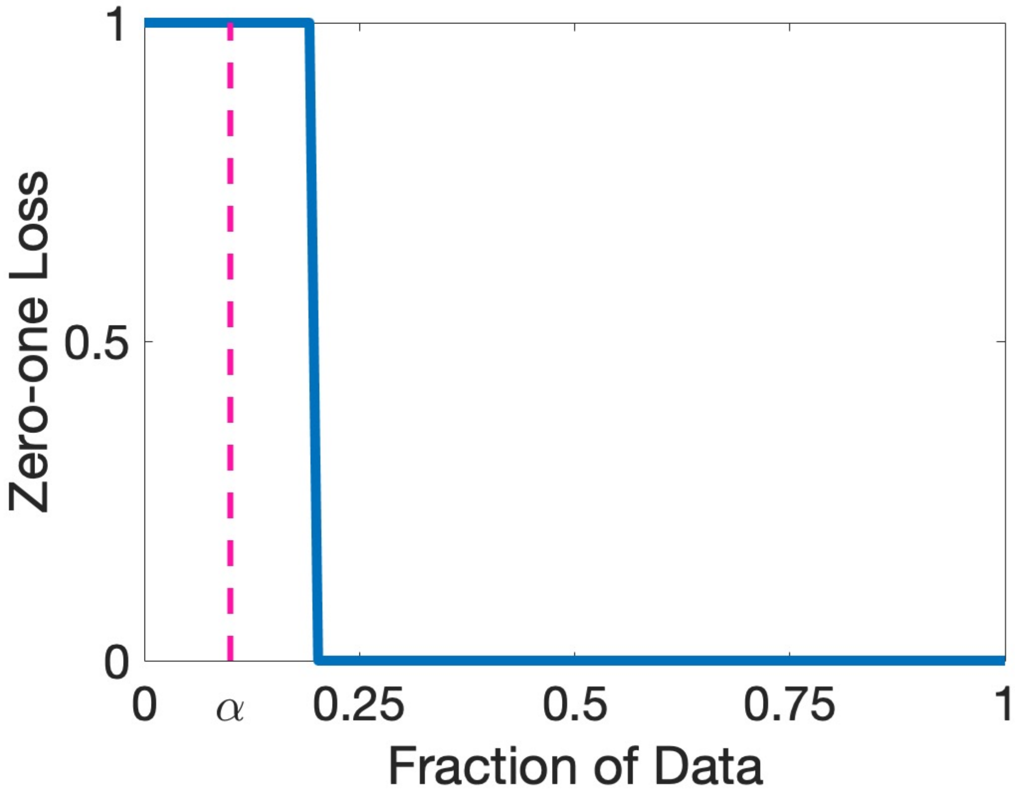
Overview

- For classification tasks, if we use deterministic models, then CVaR is almost equivalent to ERM.
- We propose to circumvent this problem by using ensemble models, and specifically we train ensemble models with Boosting.
- We find that for ensemble models, CVaR is equivalent to LPBoost, a variant of Boosting. So we design a framework based on this.

Contents

1. CVaR is Equivalent to ERM in Classification
2. Boosting
3. Connection Between Boosting and CVaR
4. The Boosted CVaR Classification Framework

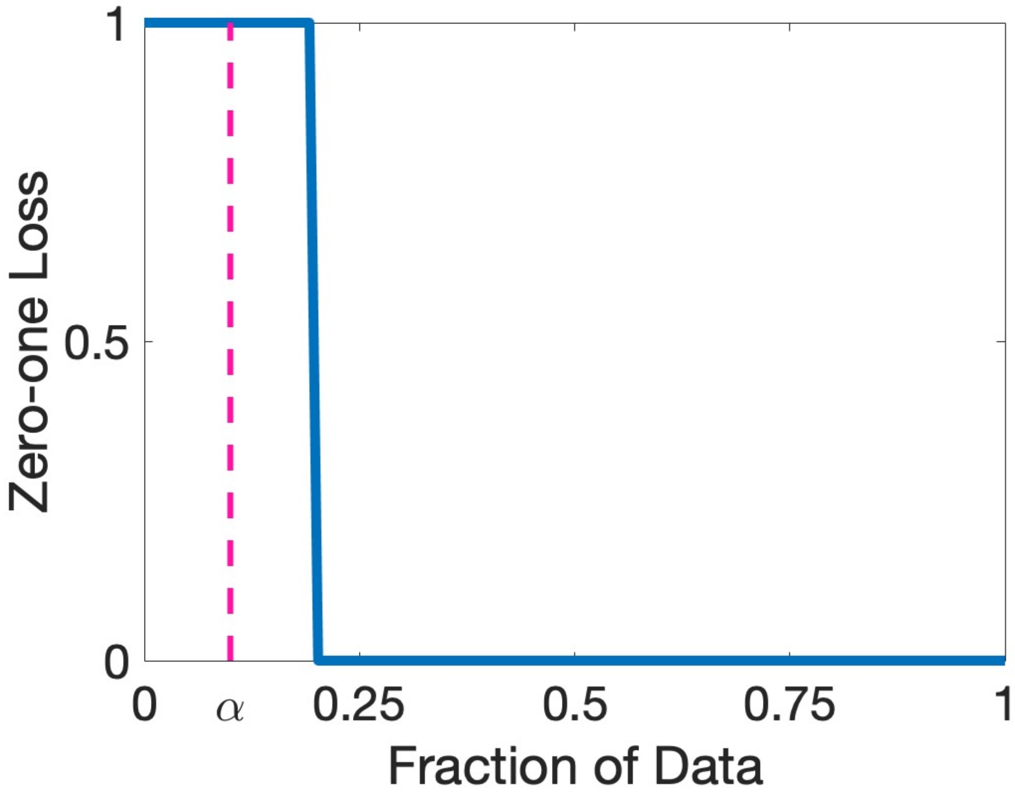
Classification: Zero-one Loss



Proposition: Let $L^{0/1}(f)$ be the average zero-one loss, and $\mathrm{CVaR}_{\alpha}^{0/1}(f)$ be the α-CVaR zero-one loss, then

$$\mathrm{CVaR}_{\alpha}^{0/1}(f) = \min\{1, L^{0/1}(f)/\alpha\}$$

Classification: Zero-one Loss



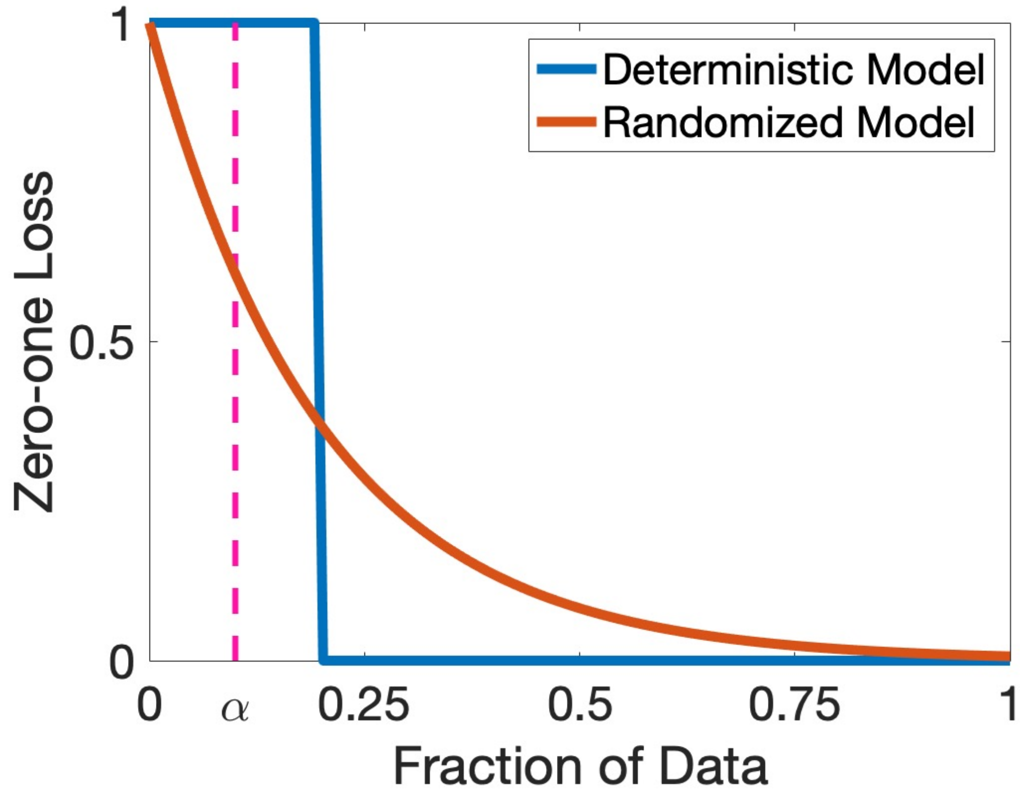
$$\mathrm{CVaR}_{\alpha}^{0/1}(f) = \min\{1, L^{0/1}(f)/\alpha\}$$

CVaR loss is non-decreasing with average loss.

Minimizing the CVaR loss is equivalent to minimizing the average loss (ERM).

Randomized Model

$f(x)$ is a distribution over $\mathcal{Y}$ instead of a single value y



The zero-one loss of a randomized model is a real value in $[0,1]$ instead of binary.

Thus, it breaks the previous connection between the CVaR loss and the average loss.

Contents

1. The Problem of CVaR in Classification
2. Boosting
3. Connection Between Boosting and CVaR
4. The Boosted CVaR Classification Framework

The General Boosting Framework

- Training set: $\{(x_1, y_1), \dots, (x_n, y_n)\}$
- Weak Learner $\mathcal{L}$ (e.g. ERM)
- For each round t:

Δ_n is the unit simplex in $\mathbb{R}^n$

1. Pick a sample weight vector $w^t = (w_1^t, \dots, w_n^t) \in \Delta_n$
2. Feed the sample weights and the training set to $\mathcal{L}$ and get a base classifier f^t

- After T rounds, pick a model weight vector $\lambda = (\lambda_1, \dots, \lambda_T) \in \Delta_T$ and output the ensemble model $F = (f^1, \dots, f^T, \lambda)$

Inference with the Ensemble Model

- Given an ensemble model $F = (f^1, \dots, f^T, \lambda)$ and an input x:

  1. Randomly sample an f^t according to the distribution λ
  2. Return $\hat{y} = f^t(x)$

- Expected loss of F on sample (x, y):

Loss function
$\ell(\hat{y}, y)$

$$\ell(F(x), y) = \sum_{t=1}^{T} \lambda_t \ell(f^t(x), y)$$

Extend Boosting to Train Fair Models

Boosting for Accuracy (Original)

We have a weak learner $\mathcal{L}$ that outputs models with accuracy at least $50\% + \delta$ for some $\delta > 0$

Weak Learner → Strong Learner

Boosting for Fairness

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

The learner is strong but unfair

Unfair Learner → Fair Learner

Contents

1. The Problem of CVaR in Classification
2. Boosting
3. Connection Between Boosting and CVaR
4. The Boosted CVaR Classification Framework

α-LPBoost[1]

Let $\ell_i^t = \ell(f^t(x_i), y_i)$. At round $t+1$, solve the following linear program to pick the sample weight vector $w = (w_1 \ldots, w_n)$:

Primal

$$\rho_*^t = \max_{\lambda,\rho} \rho - \frac{1}{\alpha n} \sum_{i=1}^n \psi_i$$

$$\text{s.t.} \quad \lambda \in \Delta_t$$

$$\psi_i \geq 0, \psi_i \geq \rho - 1 + \sum_{s=1}^t \lambda_s \ell_i^s$$

Dual

$$\gamma_*^t = \min_{w,\gamma} \gamma$$

$$\text{s.t.} \quad \sum_{i=1}^n w_i \ell_i^s \geq 1 - \gamma, \forall s \in [t]$$

$$w \in \Delta_n, w_i \leq \frac{1}{\alpha n}$$

[1] Demiriz et al. Linear Programming Boosting via Column generation. *Machine Learning*, 46(1):225-254, 2002.

α-LPBoost

Strong duality: $\rho_*^t = \gamma_*^t$

Let $\ell_i^t = \ell(f^t(x_i), y_i)$. At round $t+1$, solve the following linear pr … he sample weight vector $w = (w_1 \dots, w_n)$:

Find w such that the weighted loss of every previous model is large

Primal

$$\rho_*^t = \max_{\lambda,\rho} \rho - \frac{1}{\alpha n} \sum_{i=1}^n \psi_i$$

s.t. $\lambda \in \Delta_t$

$$\psi_i \geq 0, \psi_i \geq \rho - 1 + \sum_{s=1}^t \lambda_s \ell_i^s$$

Model weight vector

Dual

$$\gamma_*^t = \min_{w,\gamma} \gamma$$

s.t. $\sum_{i=1}^n w_i \ell_i^s \geq 1 - \gamma, \forall s \in [t]$

$$w \in \Delta_n, w_i \leq \frac{1}{\alpha n}$$

Weighted loss of f^s w.r.t. w

Sample weights

α-LPBoost is Equivalent to α-CVaR

- The primal is computing the λ that minimizes the α-CVaR zero-one loss of the ensemble model that consists of $f^1, \ldots, f^t$!

- Theorem: For any $f^1, \ldots, f^t$, we have

$$\rho_*^t = \gamma_*^t = 1 - \min_{\lambda \in \Delta_t} CVaR_\alpha^{0/1}(F)$$

where $F = (f^1, \ldots, f^t, \lambda)$ is the ensemble model.

We can minimize the α-CVaR zero-one loss by maximizing γ_*^t!

Carnegie Mellon University

Using LPBoost to minimize CVaR Loss

Goal: Maximize γ_*^t

$$\gamma_*^t = \min_{w,\gamma} \gamma$$

$$\text{s.t. } \gamma \geq 1 - \sum_{i=1}^{n} w_i \ell_i^s , \forall s \in [t]$$

$$w \in \Delta_n, w_i \leq \frac{1}{\alpha n}$$

γ is the maximum accuracy of $f^1, \dots, f^t$ w.r.t. sample weight w

How to increase γ?

By training a new model f^{t+1} such that its accuracy w.r.t. sample weight w is high.

Repeat this process until there is no w such that γ is small.

Using LPBoost to minimize CVaR Loss

- Initially, $w^1 = \left(\frac{1}{n}, \ldots, \frac{1}{n}\right)$
- For each round t:
  - Feed the w^t to the unfair learner $\mathcal{L}$ to get f^t
  - Solve the dual problem of α-LPBoost to get w^{t+1}
  - Stop if $\gamma_*^{t+1} > \gamma_0$ for some stopping criteria $\gamma_0 \in (0,1)$
- Solve the primal problem of α-LPBoost to get λ

Solve the optimization problems with tools such as MOSEK.

Contents

1. The Problem of CVaR in Classification
2. Boosting
3. Connection Between Boosting and CVaR
4. The Boosted CVaR Classification Framework

Assumption on the Unfair Learner

- We have access to an unfair learner $\mathcal{L}$ such that:
  - Given any sample weight vector $w = (w_1, \dots, w_n)$, the learner can output a model f such that its average loss w.r.t. w is at most g, i.e.

$$\sum_{i=1}^{n} w_i \ell(f(x_i), y_i) \leq g$$

  - $g \in (0,1)$ is called the guarantee of the learner

The Framework

- For each round t:

  1. Pick a sample weight vector $w^t = (w_1^t, \dots, w_n^t) \in \Delta_n$
  2. Feed the sample weights and the training set to the unfair learner $\mathcal{L}$ and get a base classifier f^t whose weighted average zero-one loss w.r.t. w^t is at most g

- After T rounds, pick a model weight vector $\lambda = (\lambda_1, \dots, \lambda_T) \in \Delta_T$ and output the ensemble model $F = (f^1, \dots, f^T, \lambda)$

Convergence Rate of Regularized LPBoost[2]

- If every sample weight vector w^{t+1} is picked by solving the regularized α-LPBoost dual problem:

$$\min_{w} \ \gamma - \frac{1}{\beta} H(w)$$

$$\text{s.t.} \quad \sum_{i=1}^{n} w_i \ell_i^s \geq 1 - \gamma, \forall s \in [t]; \quad w \in \Delta_n, w_i \leq \frac{1}{\alpha n}$$

where $H(w) = -\sum_{i=1}^{n} w_i \log w_i$ is the entropy function and $\beta = \max\left(\frac{2}{\delta}\log\frac{1}{\alpha}, \frac{1}{2}\right)$, then $CVaR_{\alpha}^{0/1}(F) \leq g + \delta$ if

$$T = \max\left\{\frac{32}{\delta^2}\log\frac{1}{\alpha}, \frac{8}{\delta}\right\} = O\left(\frac{1}{\delta^2}\log\frac{1}{\alpha}\right)$$

There exists a counterexample where unregularized LPBoost takes $T = \Omega\left(\frac{1}{\alpha}\right)$ to converge

[2] Warmuth et al. Entropy Regularized LPBoost. In *International Conference on Algorithmic Learning Theory*, pages 256-271, Springer, 2008.

α-AdaLPBoost

- Pick the sample weight vector w^t with AdaBoost and the final model weight vector λ by solving the α-LPBoost primal problem.

- AdaBoost: $w_i^{t+1} \propto \exp(\eta \sum_{s=1}^{t} \ell_i^s)$

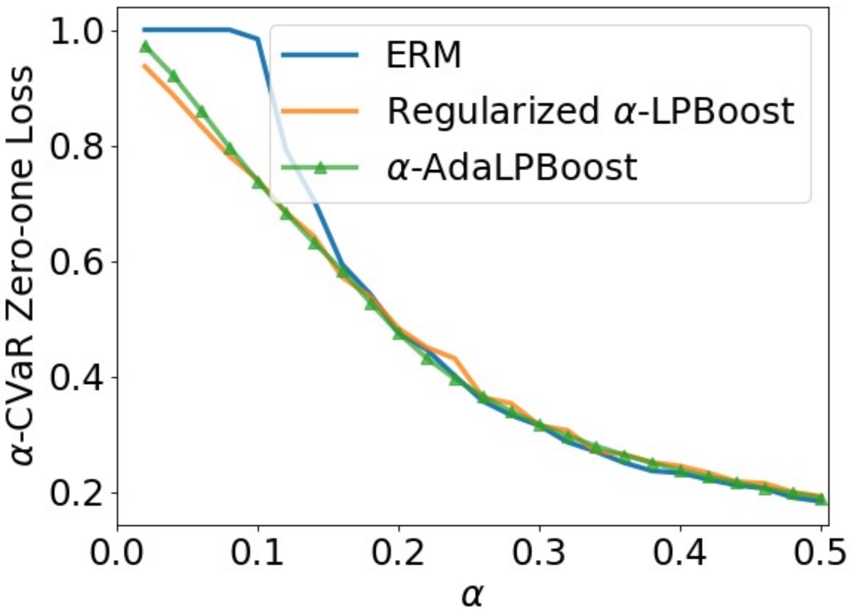
- Advantages:

  - Easier to compute w^t: No need to solve a linear program
  - Easier to adjust α: Only λ depends on α

Convergence Rate of α-AdaLPBoost

- For α-AdaLPBoost, if we set $\eta = \sqrt{\frac{8 \log n}{T}}$,then $CVaR_{\alpha}^{0/1}(F) \leq g + \delta$ with $T = O\left(\frac{\log n}{\delta^2}\right)$
- Regularization is not required

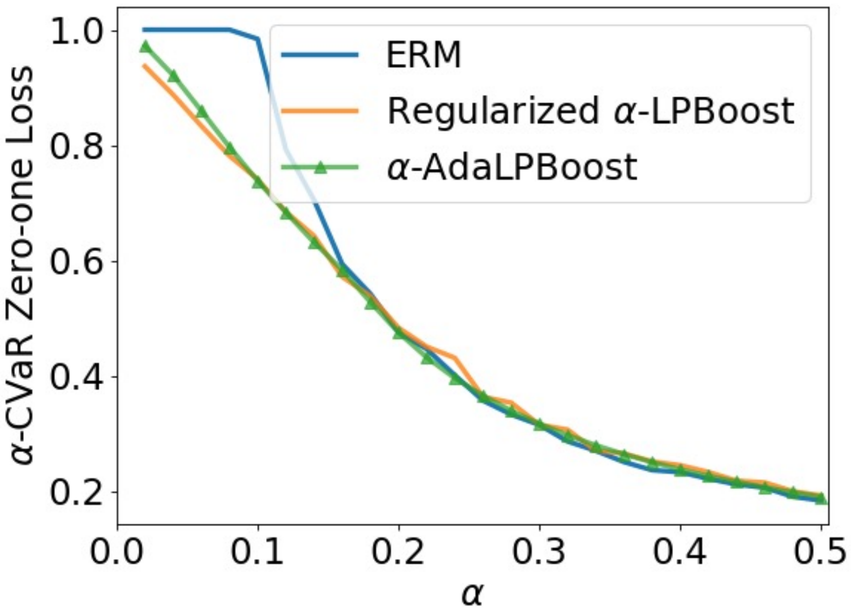
Experiments



Results on CIFAR-10

- Conducted on 4 datasets
- Run α-AdaLPBoost with different α and compare with ERM and regularized LPBoost

Experiments



Results on CIFAR-10

- When α is small, α-AdaLPBoost achieves lower α-CVaR zero-one loss than ERM
- The performance of α-AdaLPBoost is close to regularized LPBoost

Thanks.