Feature-Cost Sensitive Learning
with Submodular Trees of Classifiers

Washington University in St. Louis

Matt J. Kusner, Wenlin Chen, Quan Zhou, Zhixiang (Eddie) Xu, Kilian Q. Weinberger, Yixin Chen
Department of Computer Science & Engineering, Washington University in St. Louis, USA
Department of Automation, Tsinghua University, Beijing, China

Budgeted Learning
[Chen et al., 2012; Xu et al., 2013]

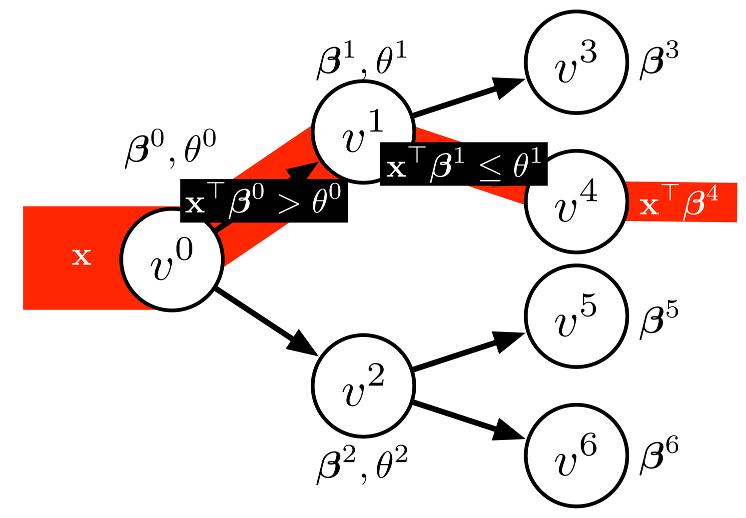
dataset $\mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{n} \subset \mathcal{R}^d \times \mathcal{Y}$

feature costs $[c_1, \cdots, c_d]^\top$

cost budget B

$$\min_{\boldsymbol{\beta}} \ell(\boldsymbol{\beta}, \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{n}) \quad \text{s.t.} \quad c(\boldsymbol{\beta}) \leq B$$

instead of a single classifier...



gradient-based optimization

set function optimization

Cost-Sensitive Tree of Classifiers
[Xu et al, 2014]

Expected Tree Loss

$$\mathbb{E}[\ell(T)] = \frac{1}{n}\sum_{v^k \in V}\sum_{i=1}^{n} p_i^k (y^{(i)} - \boldsymbol{\beta}^{k\top}\mathbf{x}^{(i)})^2$$

Expected Tree Cost

$$\mathbb{E}[c(T)] = \sum_{v^l \in L} p^l \left[\sum_a c_a \left\| \sum_{v^j \in \pi^l} |\beta_a^j| \right\|_0 \right]$$

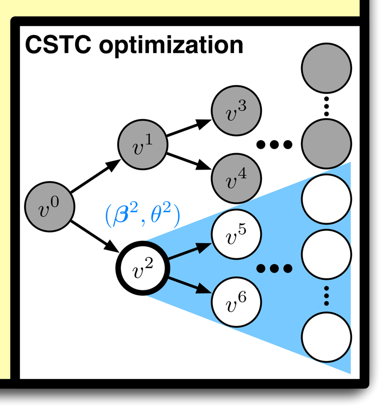
Objective

$$\min_{\boldsymbol{\beta}^1,\dots,\boldsymbol{\beta}^{|V|}} \mathbb{E}[\ell(T)] + \lambda \mathbb{E}[c(T)]$$

Results

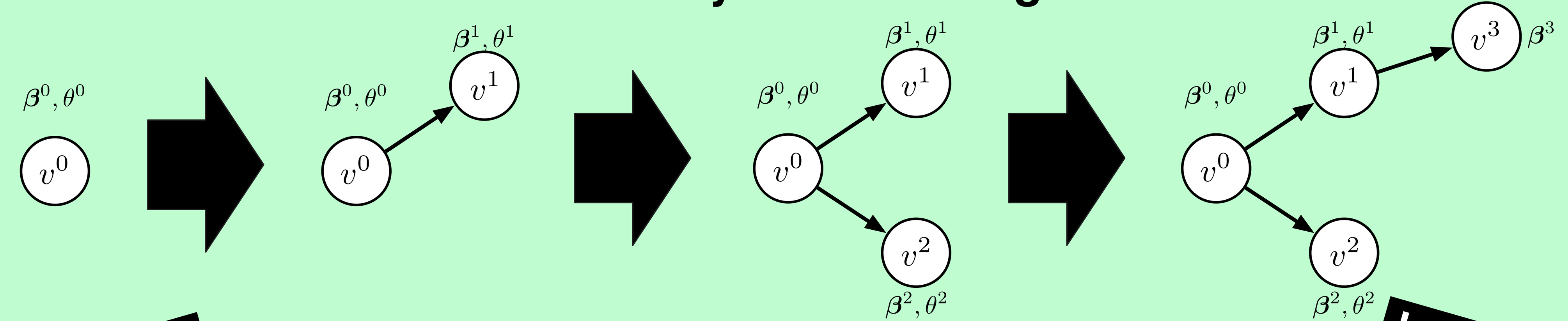
- \+ state-of-the-art performance
- \- expensive global optimization
- \- requires continuous relaxation
- \- difficult to implement

goal: a ready practical tool



Approximately Submodular Tree of Classifiers

Greedy Tree Building



best loss for features $\mathcal{A}$

Greedy Feature Selection (node k)

loss reduction per cost

$$\ell_k(\mathcal{A}) \triangleq \min_{\boldsymbol{\beta}^k} \frac{1}{n}\sum_{i=1}^{n} p_i^k (y^{(i)} - \boldsymbol{\beta}^{k\top}\mathbf{x}_{\mathcal{A}}^{(i)})^2$$

$$\arg\max_{a \in \{1,\dots,d\}} \left[\frac{\ell_k(\mathcal{A}) - \ell_k(\mathcal{A} \cup a)}{c_a}\right]$$

New Objective

$$\max_{\mathcal{A}} \ell_k(\mathcal{A}) \quad \text{s.t.} \quad c(\mathcal{A}) \leq B$$

greedy is near-optimal! $(1 - e^{-\gamma})$-approximation [Das & Kempe, 2011] [Grubb & Bagnell, 2012]

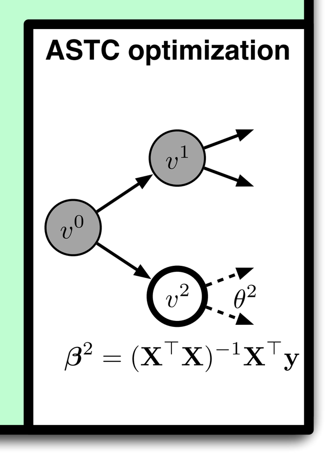
Fast Selection via QR-Decomposition

$$\frac{\ell_k(\mathcal{A}) - \ell_k(\mathcal{A} \cup a)}{c_a} = \frac{(\mathbf{q}^\top \mathbf{y})^2}{c_a}$$

where

$$\mathbf{q} = \frac{\mathbf{X}_a - \mathbf{Q}\mathbf{Q}^\top\mathbf{X}_a}{\|\mathbf{X}_a - \mathbf{Q}\mathbf{Q}^\top\mathbf{X}_a\|_2}$$

$$\mathbf{X}_{\mathcal{A}} = \mathbf{Q}\mathbf{R}$$



Results

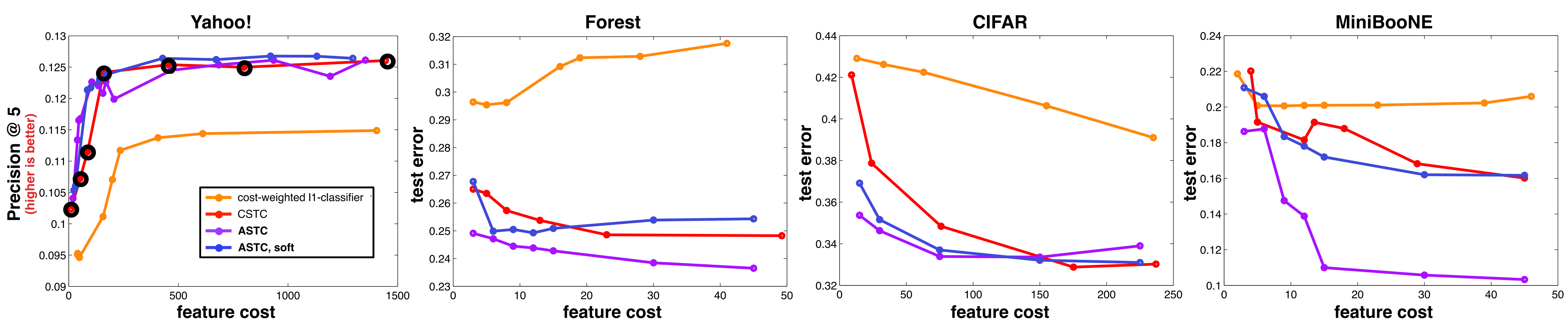


Figure 3. Plot of ASTC, CSTC, and a cost-sensitive baseline on a real-world feature-cost sensitive dataset (Yahoo!) and three non-cost sensitive datasets (Forest, CIFAR, MiniBooNE). For Yahoo! circles mark the CSTC points that are used for training time comparison, otherwise, all points are compared.

Training Speed-up

	Yahoo!							Forest						CIFAR						MiniBooNE					
Cost Budgets	10	52	86	169	468	800	1495	3	5	8	13	23	50	9	24	76	180	239	4	5	12	14	18	33	47
ASTC	119x	52x	41x	21x	15x	9.2x	6.6x	8.4x	7.0x	6.3x	4.9x	3.1x	1.4x	5.6x	2.3x	0.68x	0.25x	0.14x	7.4x	7.9x	5.5x	5.2x	4.1x	3.1x	2.0x
ASTC, soft	121x	48x	46x	18x	15x	8.2x	6.4x	8.0x	6.4x	5.7x	4.5x	2.8x	1.5x	5.3x	2.3x	0.62x	0.27x	0.13x	7.2x	6.2x	5.9x	4.2x	4.3x	2.5x	1.7x

Table 1. Training speed-up of ASTC over CSTC for different cost budgets on all datasets.

References

Chen, M., Weinberger, K. Q., Chapelle, O., Kedem, D., Xu, Z. Classifier cascade for minimizing feature evaluation cost. AISTATS, 2012

Xu, Z., Kusner, M. J., Weinberger, K. Q., Chen, M. Cost-sensitive tree of classifiers. ICML, 2013

Xu, Z., Kusner, M. J., Weinberger, K. Q., Chen, M., Chapelle, O. Classifier Cascades and Trees for Minimizing Feature Evaluation Cost. JMLR, 2014