Soft-max boosting

LabelNoise'2017

Matthieu Geist
LIEC (Université de Lorraine & CNRS)
matthieu.geist@univ-lorraine.fr

November 30, 2017

- Let $\mathcal{D}_N = \{(x_i, y_i)_{1 \leq i \leq N}\}$ be a data set, with $x_i \in \mathcal{X}$, a compact subset of $\mathbb{R}^d$, and $y_i \in \mathcal{Y}$, a finite set of labels.
- Let $c \in \mathbb{R}_+^{\mathcal{X} \times \mathcal{Y}^2}$ be a (bounded) cost function: $c(x_i, y_i, y)$ is the cost (or loss) for choosing the label y instead of the oracle response y_i for the input x_i. For example,

$$c(x_i, y_i, y) = \mathbb{I}_{\{y \neq y_i\}} = \begin{cases} 1 \text{ if } y \neq y_i \\ 0 \text{ else} \end{cases} .$$

- Let $\mathcal{G}$ be a subset of (deterministic) functions mapping inputs to labels (that is, deterministic decision rules), $\mathcal{G} \subset \mathcal{Y}^{\mathcal{X}}$.
- The "ideal" empirical risk for cost-sensitive multiclass classification is

$$R_N(g) = \frac{1}{N} \sum_{i=1}^{N} c(x_i, y_i, g(x_i)).$$

- Lack of convexity and smoothness...

- Let Ψ be a subset of $\mathbb{R}^{\mathcal{X}\times\mathcal{Y}}$. A function $\psi \in \Psi$ can be understood as a score function, ranking different labels for a given input.
- Define the decision rule as being greedy resp. to ψ: $g_\psi(x) = \text{argmax}_{y\in\mathcal{Y}}\, \psi(x, y)$.
- Let $\varphi : \mathbb{R} \to \mathbb{R}$ be some convex function (*e.g.*, $\varphi(t) = e^t$).
- Define (for example) the following (convex when ψ is linearly parameterized) surrogate:

$$R_N(\psi) = \frac{1}{N}\sum_{i=1}^{N}\sum_{y\in\mathcal{Y}} c(x_i, y_i, y)\varphi(\psi(x_i, y)).$$

- The proxy should be well calibrated (Pires *et al.*, 2013), guess-averse (Beijbom *et al.*, 2014), tractable... Choosing the right proxy is difficult.

- Define a stochastic decision rule from the score function:

$$g_\psi(y|x) = \frac{e^{\psi(x,y)}}{\sum_{z\in\mathcal{Y}} e^{\psi(x,z)}}.$$

- Minimize directly the "ideal" risk, with a stochastic decision rule replacing the deterministic one:

$$R_N(\psi) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{y\sim g_\psi(.|x_i)}[c(x_i, y_i, y)]$$

$$= \frac{1}{N}\sum_{i=1}^{N}\sum_{y\in\mathcal{Y}} g_\psi(y|x_i)c(x_i, y_i, y).$$

- (Recall the "ideal" risk:)

$$R_N(g) = \frac{1}{N}\sum_{i=1}^{N} c(x_i, y_i, g(x_i)).$$

- Convexity is lost, but no calibration problem.

Basic boosting algorithm (Grubb & Bagnell, 2011)

Inputs;
Initial function ψ_0 ;
Learning rates $(\alpha_t)_{t>0}$;
Hypothesis space $\mathcal{H}$;

for $t = 1, 2, \ldots T$ **do**

Compute the gradient $\nabla_t = \nabla R_N(\psi_{t-1})$;

Project ∇_t onto $\mathcal{H}$, finding nearest directions h_t^*:

$$h^* = \operatorname*{argmax}_{h \in \mathcal{H}} \frac{\langle \nabla R_N(\psi), h \rangle_N}{\|h\|_N} \ ;$$

Update ψ: $\psi_t = \psi_{t-1} - \alpha_t \frac{\langle h_t^*, \nabla_t \rangle_N}{\|h_t^*\|_N^2} h_t^*$;

end

- We focus on hypothesis spaces generated by binary classifiers as weak learners, that is:

$$\mathcal{H} \subset \{-1, +1\}^{\mathcal{X} \times \mathcal{Y}}.$$

- It can be shown that

$$h^* \in \operatorname*{argmax}_{h \in \mathcal{H}} \frac{1}{N} \sum_{i=1}^{N} \sum_{y \in \mathcal{Y}} g_\psi(y|x_i) \Delta_\psi c_i(y) h(x_i, y)$$

$$\text{with } \Delta_\psi c_i(y) = c(x_i, y_i, y) - \sum_{z \in \mathcal{Y}} g_\psi(z|x_i) c(x_i, y_i, y).$$

- Weighted binary classification problem with
  - inputs $(x_i, y)_{1 \leq i \leq N, y \sim g_\psi(.|x_i)}$
  - weights $(|\Delta_\psi c_i(y)|)$
  - outputs $(\text{sgn}(\Delta_\psi c_i(y)))$
- `sm-boost` reduces cost-sensitive multiclass classification to a sequence of weighted binary classification problems.

Lemma

Without loss of generality, assume that $c \in [0,1]^{\mathcal{X}\times\mathcal{Y}^2}$. *The functional* $R_N(\psi)$ *defined as*

$$R_N(\psi) = \frac{1}{N}\sum_{i=1}^{N}\sum_{y\in\mathcal{Y}} g_\psi(y|x_i)c(x_i,y_i,y)$$

is 1*-Lipschitz-smooth:*

$$\forall\psi,\phi \in L^2(\mathcal{X},\mathbb{R}^{\mathcal{Y}},\hat{\rho}), \quad \|\nabla R_N(\psi) - \nabla R_N(\phi)\|_N \leq \|\psi-\phi\|_N.$$

Definition ((Grubb & Bagnell, 2011))

The hypothesis space $\mathcal{H}$ has an edge $\gamma \in]0,1[$ if for every gradient $\nabla R_N(\psi)$, there exists a function $h \in \mathcal{H}$ such that:

$$\langle \nabla R_N(\psi), h \rangle_N \geq \gamma \|\nabla R_N(\psi)\|_N \|h\|_N.$$

Theorem

Assume that $\mathcal{H}$ has an edge $\gamma > 0$. Without loss of generality, assume also that $c \in [0,1]^{\mathcal{X} \times \mathcal{Y}^2}$. Let $(\psi_t)_{t \geq 0}$ be the sequence of functions computed by the `sm-boost` algorithm for the learning rate $\alpha_t = 1$ (for all $t > 0$). We have that:

$$\lim_{t \to \infty} \|\nabla R_N(\psi_t)\|_N = 0 \text{ and } \min_{1 \leq t \leq T+1} \|\nabla R_N(\psi_t)\|_N \leq \frac{1}{\gamma} \sqrt{\frac{2}{T}}.$$

Some remarks:

- This does tell nothing about the generalization error (not addressed).
- The risk is smooth, but not convex. Zeroing the gradient is not enough to get a global minimum.
- However, a perturbation analysis in a simplified case (binary loss) shows that local minima are unlikely (see the paper).

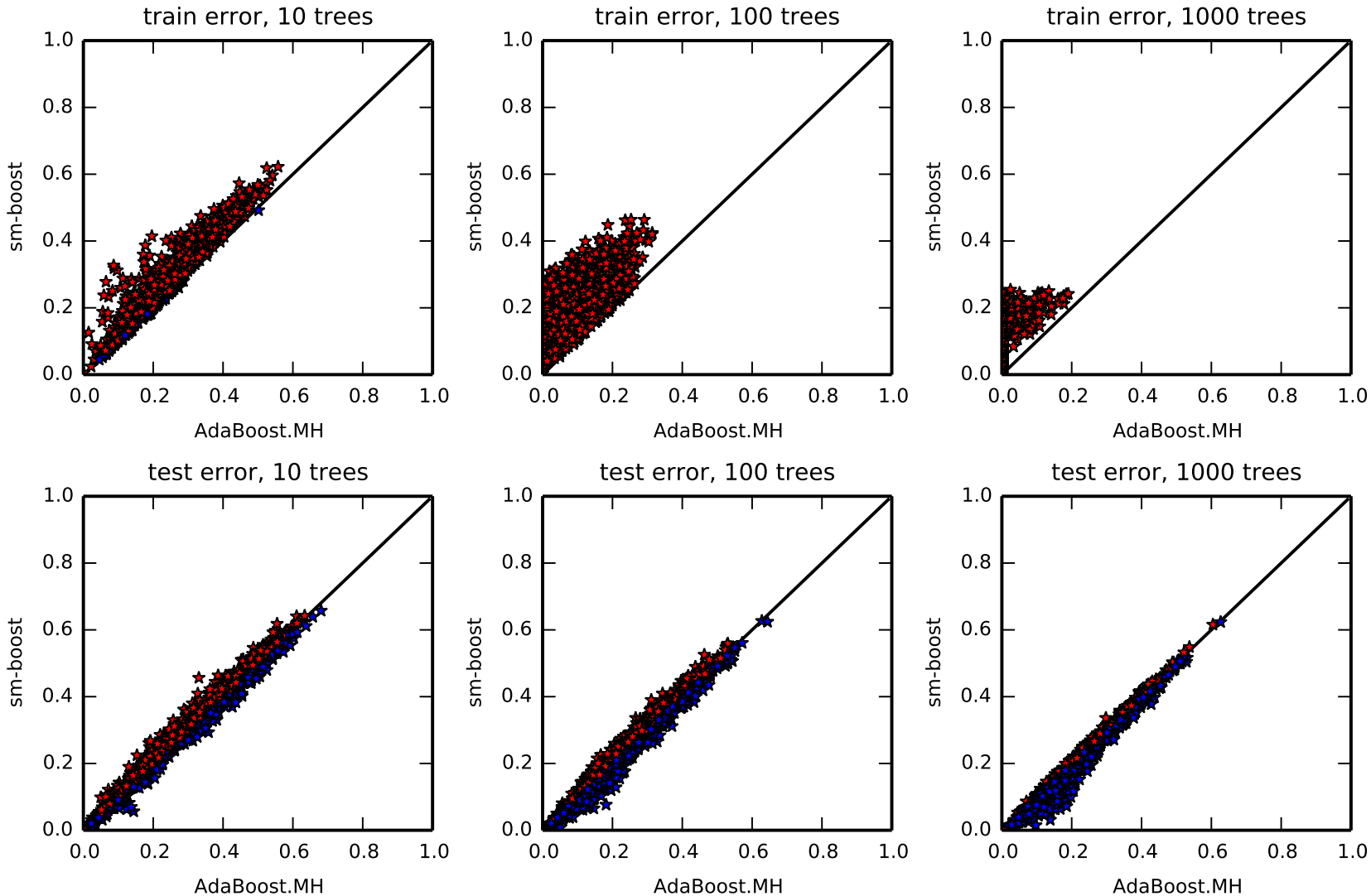
Considered competitors:

- AdaBoost.MH (Schapire & Singer, 1999), a multiclass generalization of AdaBoost based on the Hamming loss
- SAMME (Zhu *et al.*, 2009), a forward stagewise additive modeling approach minimizing an exponential-based surrogate for the binary multiclass case.
- MartiBoost (Long & Servedio, 2005), a non-convex booster.

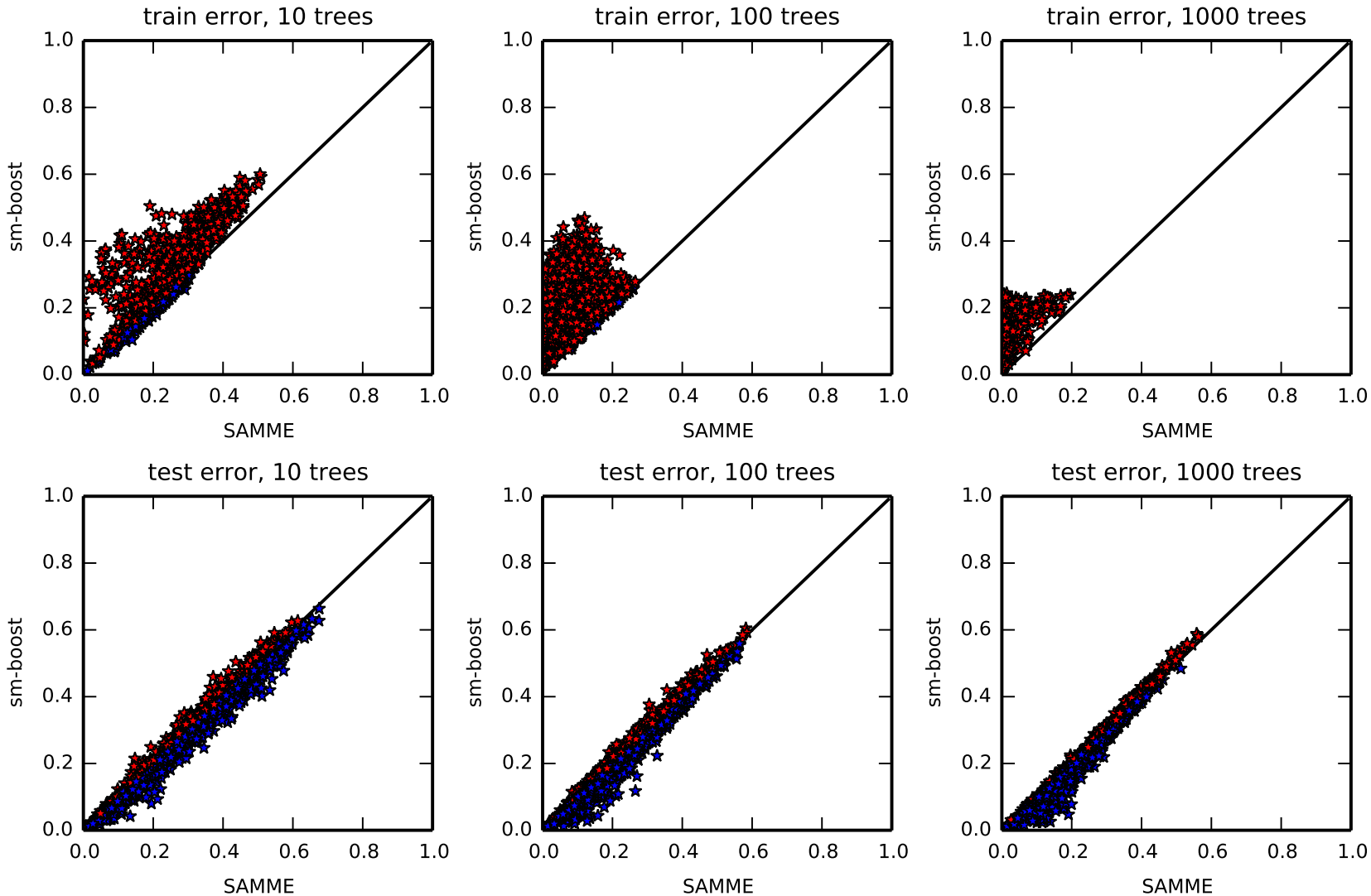
Randomly generated mutliclass classification problems inspired from the Madelon data set (Guyon, 2003). For each problem, the following quantities are chosen randomly:

- number of informative features
- number of redondant features (random linear combination of linear ones)
- number of repeated features
- number of useless features
- number of classes
- number of clusters per classe
- fraction of samples whose classes are randomly exchanged
- number of training samples

Comparison to AdaBoost.MH



Comparison to SAMME



Data set	# train	# test	# features	# classes
dna	2000	1186	180	3
letter	15000	5000	16	26
mnist	60000	10000	784	10
pendigits	7494	3498	16	10
satimage	4435	2000	36	6
segment	210	2100	19	10
splice	1000	2175	60	2
vowel	528	462	10	11

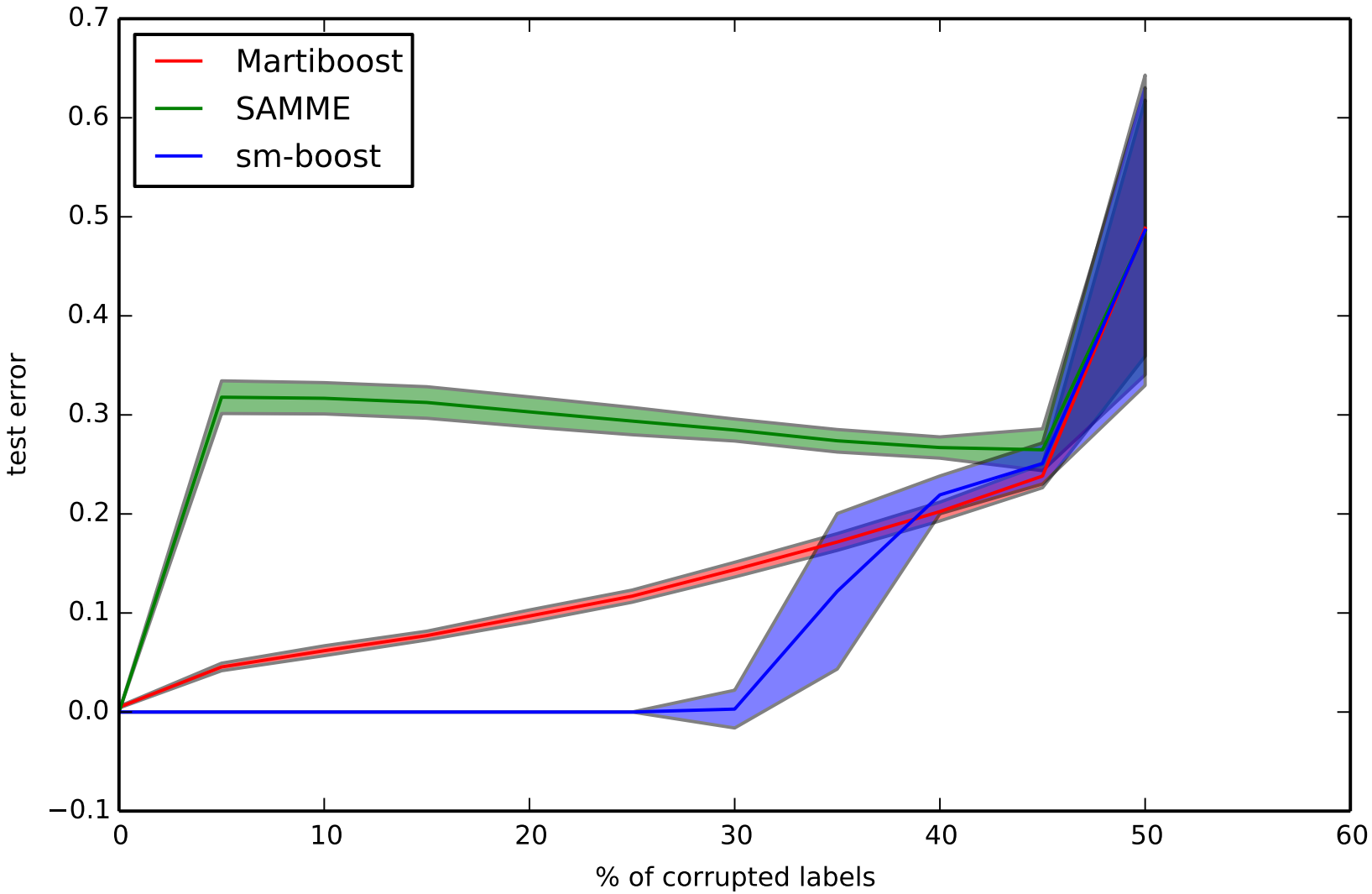
Test error rates on considered data sets

		$T = 10$	$T = 100$	$T = 1000$
dna	AdaBoost.MH	5.14	3.96	4.05
	SAMME	6.15	4.64	4.46
	sm-boost	6.49	4.89	4.13
letter	AdaBoost.MH	15.0	3.72	2.34
	SAMME	13.0	3.68	2.84
	sm-boost	26.9	10.7	5.40
mnist	AdaBoost.MH	12.7	4.14	1.76
	SAMME	10.8	3.72	2.52
	sm-boost	14.0	7.12	3.56
pendigits	AdaBoost.MH	4.86	2.43	2.17
	SAMME	3.03	2.49	2.29
	sm-boost	8.23	4.57	2.83
satimage	AdaBoost.MH	13.0	10.0	9.20
	SAMME	13.7	9.85	8.8
	sm-boost	12.9	10.8	8.95
segment	AdaBoost.MH	5.90	6.00	6.38
	SAMME	10.86	10.86	10.86
	sm-boost	9.19	6.95	6.19
splice	AdaBoost.MH	2.94	2.71	2.53
	SAMME	5.75	3.17	2.71
	sm-boost	4.64	3.63	3.08
vowel	AdaBoost.MH	49.4	44.6	42.9
	SAMME	47.8	39.4	38.3
	sm-boost	56.9	45.7	46.3

Test error rates on noisy data sets (20% of noise)

		$T = 10$	$T = 100$	$T = 1000$
dna	AdaBoost.MH	8.26 (5.14)	7.67 (3.96)	7.25 (4.05)
	SAMME	13.8 (6.15)	10.5 (4.64)	8.85 (4.46)
	sm-boost	6.15 (6.49)	4.64 (4.89)	6.24 (4.13)
letter	AdaBoost.MH	26.2 (15.0)	15.1 (3.72)	8.58 (2.34)
	SAMME	25.8 (13.0)	11.6 (3.68)	6.84 (2.84)
	sm-boost	29.2 (26.9)	13.2 (10.7)	6.98 (5.40)
mnist	AdaBoost.MH	19.8 (12.7)	11.5 (4.14)	7.01 (1.76)
	SAMME	17.4 (10.8)	12.1 (3.72)	5.48 (2.52)
	sm-boost	14.1 (14.0)	6.96 (7.12)	3.78 (3.56)
pendigits	AdaBoost.MH	10.9 (4.86)	7.63 (2.43)	5.77 (2.17)
	SAMME	11.7 (3.03)	6.66 (2.49)	4.49 (2.29)
	sm-boost	9.06 (8.23)	3.97 (4.57)	3.06 (2.83)
satimage	AdaBoost.MH	16.7 (13.0)	12.8 (10.0)	10.9 (9.20)
	SAMME	17.3 (13.7)	12.9 (9.85)	10.7 (8.8)
	sm-boost	13.8 (12.9)	11.8 (10.8)	10.4 (8.95)
segment	AdaBoost.MH	15.6 (5.90)	13.8 (6.00)	13.9 (6.38)
	SAMME	17.0 (10.86)	14.0 (10.86)	13.0 (10.86)
	sm-boost	15.9 (9.19)	12.4 (6.95)	12.0 (6.19)
splice	AdaBoost.MH	16.9 (2.94)	16.7 (2.71)	14.9 (2.53)
	SAMME	18.3 (5.75)	17.2 (3.17)	14.3 (2.71)
	sm-boost	9.15 (4.64)	10.7 (3.63)	13.1 (3.08)
vowel	AdaBoost.MH	55.8 (49.4)	47.2 (44.6)	47.6 (42.9)
	SAMME	54.1 (47.8)	44.6 (39.4)	41.3 (38.3)
	sm-boost	52.8 (56.9)	48.9 (45.7)	43.7 (46.3)

- Convex boosters are defeated by random classification noise (Long & Servedio, 2010).
- There exists a toy problem ($\mathcal{X} = \{0,1\}^{21}$, $\mathcal{Y} = \{0,1\}$) designed to defeat convex boosters (Long & Servedio, 2010).
- There exist non-convex boosters (yet, for cost-insensitive binary classification), notably MartiBoost (Long & Servedio, 2005).


Martiboost
SAMME
sm-boost
test error
% of corrupted labels
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−0.1
0
10
20
30
40
50
60

Summary and future works

Summary:

- Cost-sensitive multiclass classification.
- Instead of introducing a convex surrogate, minimize directly the risk of interest, considering a stochastic decision rule.
- Partial convergence analysis.
- Experimentally, less sensitive to noise (but not that competitive in the noiseless case).

Future works:

- Study the quality of the solution (is zeroing the gradient enough to get close to the global optimum).
- Study the generalization error.
- More algorithms (beyond naive gradient descent).

Thanks!

Questions?

References I

Beijbom, Oscar, Saberian, Mohammad, Kriegman, David, & Vasconcelos, Nuno. 2014.
Guess-Averse Loss Functions For Cost-Sensitive Multiclass Boosting.
Pages 586–594 of: International conference on machine learning (icml).

Grubb, Alexander, & Bagnell, J. Andrew. 2011.
Generalized boosting algorithms for convex optimization.
In: International conference on machine learning (icml).

Guyon, Isabelle. 2003.
Design of experiments of the nips 2003 variable selection benchmark.
In: Nips 2003 workshop on feature extraction and feature selection.

Long, Philip M., & Servedio, Rocco A. 2005.
Martingale Boosting.
Pages 79–94 of: Conference on learning theory (colt).
Springer-Verlag.

References II

Long, Philip M., & Servedio, Rocco A. 2010.
Random Classification Noise Defeats All Convex Potential Boosters.
Machine learning, **78**(3), 287–304.

Pires, Bernardo Ávila, Ghavamzadeh, Mohammad, & Szepesvári, Csaba. 2013.
Cost-sensitive Multiclass Classification Risk Bounds.
In: International conference on machine learning (icml).

Schapire, Robert E., & Singer, Yoram. 1999.
Improved boosting algorithms using confidence-rated predictions.
Machine learning, **37**(3), 297–336.

Zhu, Ji, Zou, Hui, Rosset, Saharon, & Hastie, Trevor. 2009.
Multi-class AdaBoost.
Statistics and its interface, **2**, 249–360.