AdaBoost

Slides modified from: MLSS'03: Gunnar Rätsch,
Introduction to Boosting

http://www.boosting.org

AdaBoost: Introduction

- Classifiers
  - Supervised Classifiers
    - Linear Classifiers
      - Perceptron, Least Squares Methods
      - Linear SVM
    - Nonlinear Classifiers
      - Part I: Multi Layer Neural Networks
      - Part II: Pol. Class., RBF, Nonlinear SVM
    - Nonmetric Methods - Decision Trees
    - **AdaBoost**
  - Unsupervised Classifiers

3

AdaBoost: Agenda

- Idea AdaBoost
  (**Ada**ptive **Boost**ing, R. Scharpire, Y. Freund, ICML, 1996):
  - Combine many low-accuracy classifiers (weak learners) to create a high-accuracy classifier (strong learners)

4

AdaBoost: Introduction

- Example: 2 classes of apples

The World:

Data:	$\{ (\underline{x}_n, y_n) \}_{n=1}^{N}, \quad \underline{x}_n \in \mathbb{R}^d, \quad y_n \in \{\pm 1\}$
Unknown target function:	$y = f(\underline{x}) \ (or\ y \sim P(y \mid \underline{x}))$
Unknown distribution:	$\underline{x} \sim p(\underline{x})$
Objective:	Given new $\underline{x}$, predict y

Problem: $P(\underline{x}, y)$ **is unknown!**

AdaBoost: Introduction

The Model:

- Hypothesis class: $\mathcal{H} = \left\{ \ h \mid h : \mathbb{R}^d \to \{\pm 1\} \ \right\}$
- Loss: $l(y,\ h(\underline{x}))$ (e.g. $I[y \neq h(\underline{x})]$)

- Objective: Minimize the true (expected) loss – (“generalization error”)

$$h^* = \arg\min_{h \in \mathcal{H}} L(h) \text{ with } \boxed{L(h) := \mathbf{E}_{\mathbb{X}\times\mathbb{Y}}\left[\ l(\mathbb{Y},\ h(\mathbb{X}))\ \right]}$$

- Problem: Only a data sample is available, $P(\underline{x}, y)$ is unknown!

- Solution: Find empirical minimizer $\hat{h}_N = \min_{h \in \mathcal{H}} \frac{1}{N} \sum_{n=1}^{N} l(y_n,\ h(\underline{x}_n))$

How can we efficiently construct complex hypotheses with small generalization errors?

AdaBoost: Frame work

Algorithm

Idea:

- Simple Hypotheses are not perfect!
- Hypotheses combination → increased accuracy

Problems:

- How to generate different hypotheses?
- How to combine them?

Method:

- Compute distribution $d_1, \ldots, d_N$ on examples
- Find hypothesis on the weighted training sample $(\underline{x}_1, y_1, d_1), \ldots, (\underline{x}_N, y_N, d_N)$
- Combine hypotheses $h_1, h_2, \ldots$ linearly:

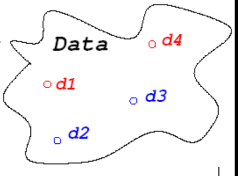
$$f = \sum_{t=1}^{T} \alpha_t h_t$$

AdaBoost: Frame work

7

Input: N examples $\{(\underline{x}_1, y_1), \ldots, (\underline{x}_N, y_N)\}$,
L a learning algorithm generating hypothesis $h_t(\underline{x})$ (classifiers)
T maxNumber of hypotheses in the ensemble

Initialize: d_n weight of example n ($\underline{d}$ is a distribution with $1 = \sum_{n=1}^{N} d_n^{(t)}$)
$d_n^{(1)} = 1/N$ for all $n = 1, \cdots, N$



Do for $t = 1, \ldots, T$,

1. Train base learner according to example distribution $\underline{d}^{(t)}$ and obtain hypothesis $h_t : \underline{x} \mapsto \{\pm 1\}$.
2. compute weighted error $\varepsilon_t = \sum_{n=1}^{N} d_n^{(t)} \, \mathbf{I}(y_n \neq h_t(x_n))$
3. compute hypothesis weight $\alpha_t = \frac{1}{2} \ln \frac{1-\varepsilon_t}{\varepsilon_t}$
4. update example distribution $d_n^{(t+1)} = d_n^{(t)} \exp\left(-\alpha_t y_n h_t(\underline{x}_n)\right) / Z_t$

Z_t is a normalization factor

Output: final hypothesis $f_{Ens}(\underline{x}) = \sum_{t=1}^{T} \alpha_t h_t(\underline{x})$

AdaBoost: Decision Stumps

8

- A family of weak learners,

e.g. Decision stump:

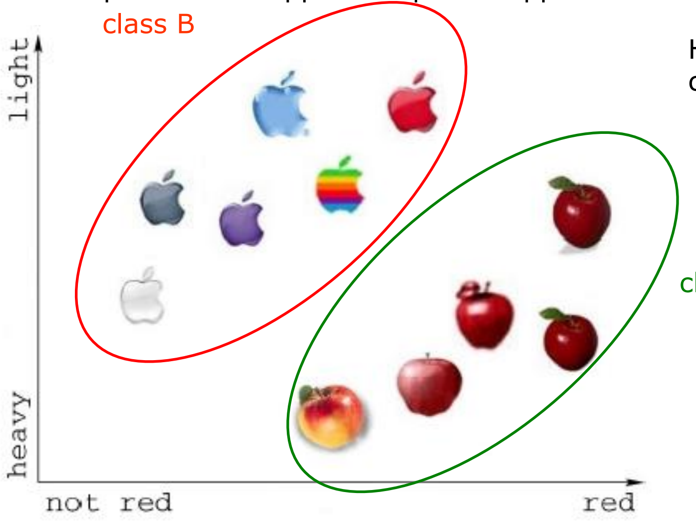
– can perform a single test on a single attribute with threshold Θ .

– parameterize all decision stumps as follows:

$$f^j(\underline{x};\theta) = \begin{cases} 1 & \text{if } x_j > \theta \\ -1 & \text{else} \end{cases} \quad , \quad j = 1, \ldots, d$$

AdaBoost: Example

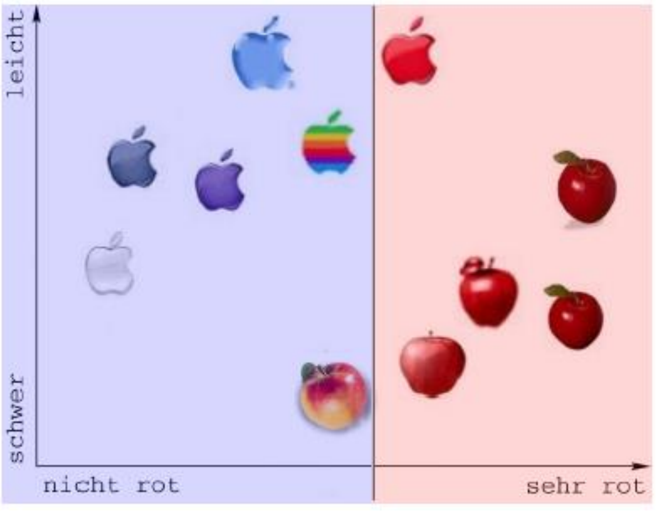
- Example: natural apples vs. plastic apples



How to classify?

AdaBoost

- Example: natural apples vs. plastic apples

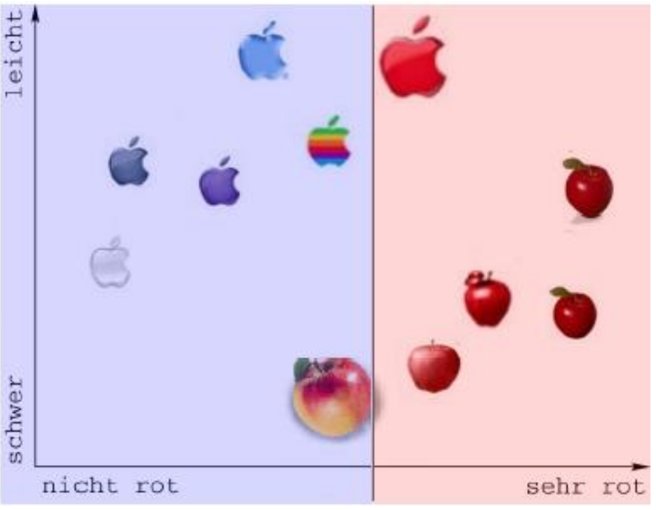


1st hypothesis

Weak classifier (cuts on coordinate axes)

AdaBoost

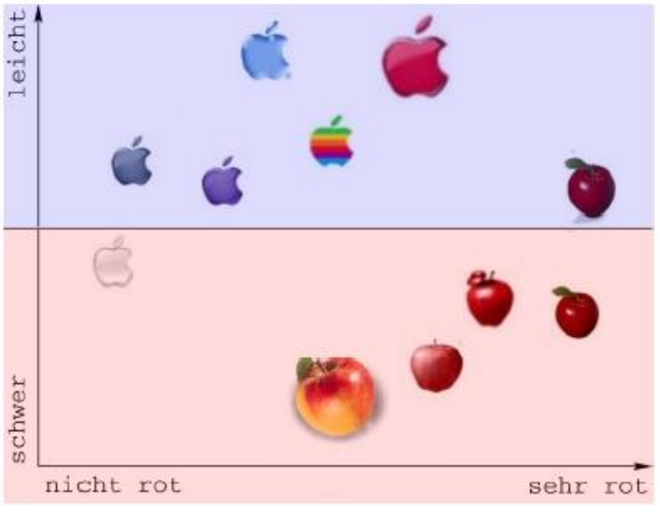
- Example:



Recomputing weightings of the training patterns

AdaBoost

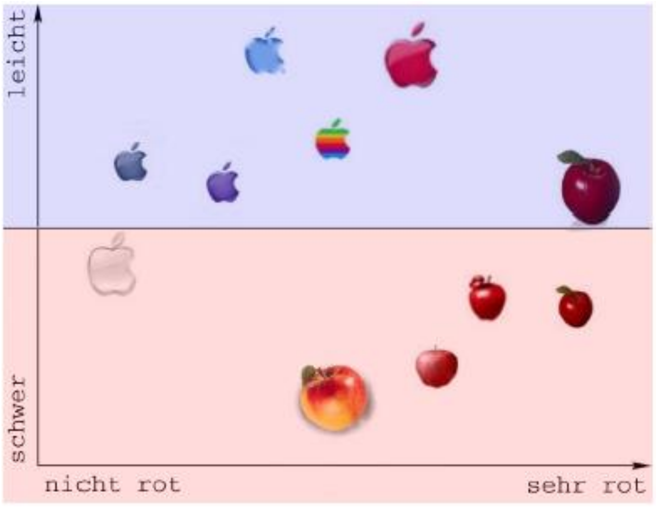
- Example:



2^{nd} hypothesis

AdaBoost

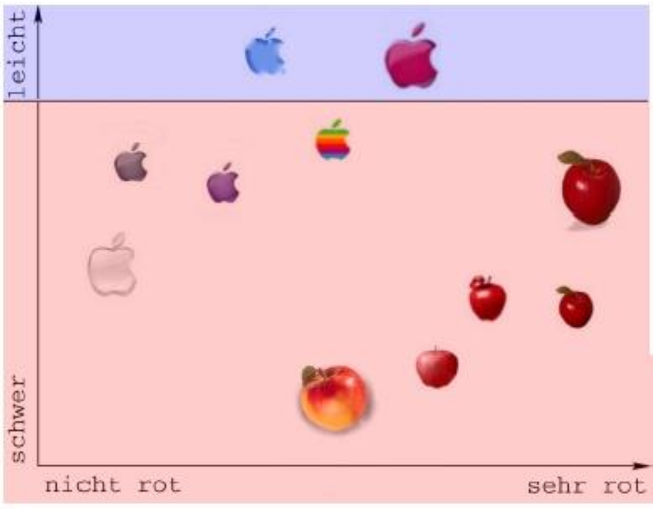
- Example:



Recompute weighting

AdaBoost

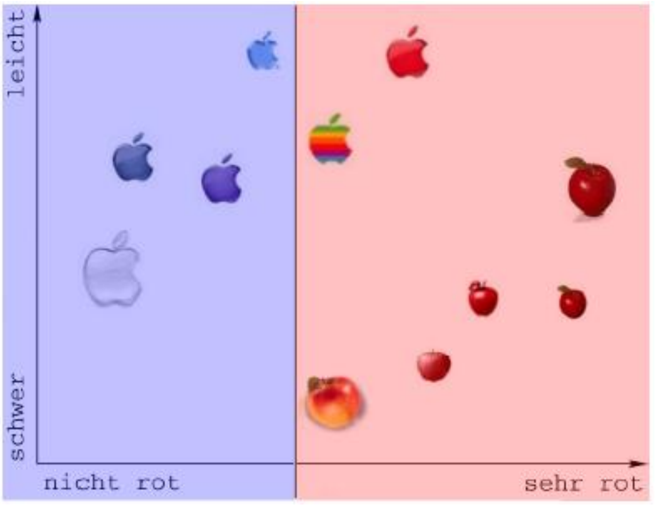
- Example:



3rd hypothesis

AdaBoost

- Example:

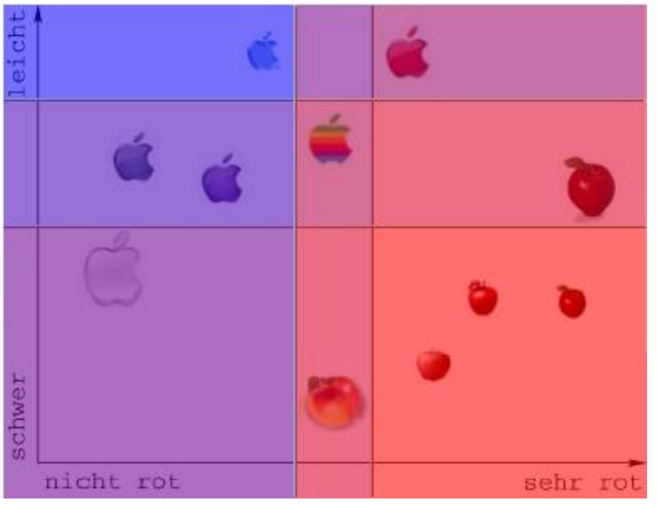


Recompute weighting

4^{th} hypothesis

AdaBoost

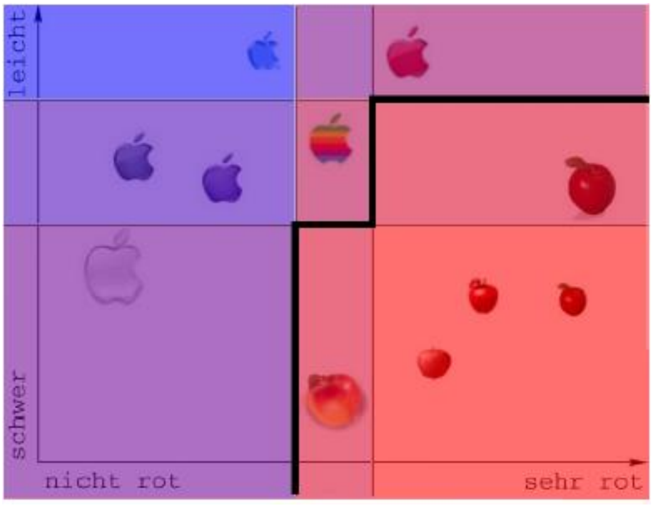
- Example:



Combination of hypotheses

AdaBoost

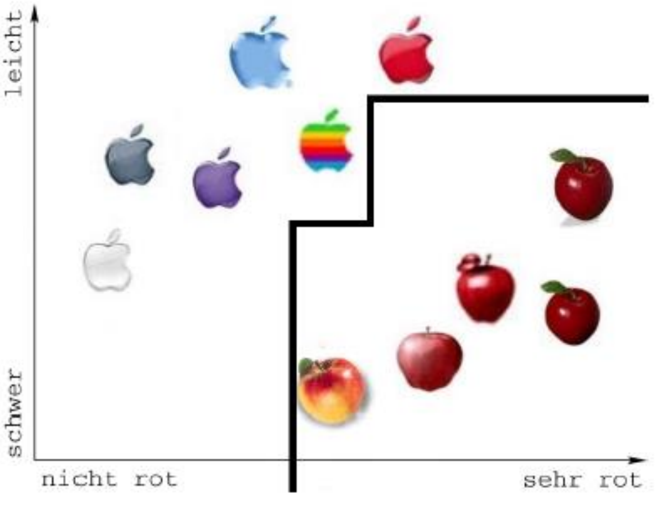
- Example:



Decision surface

AdaBoost

- Example

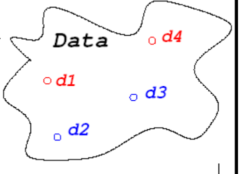


Final decision function

AdaBoost: Frame work

Input: N examples $\{(\underline{x}_1, y_1), \ldots, (\underline{x}_N, y_N)\}$,
L a learning algorithm generating hypothesis $h_t(\underline{x})$ (classifiers)
T maxNumber of hypotheses in the ensemble

Initialize: d_n weight of example n ($\underline{d}$ is a distribution with $1 = \sum_{n=1}^{N} d_n^{(t)}$)
$d_n^{(1)} = 1/N$ for all $n = 1, \cdots, N$



Do for $t = 1, \ldots, T$,

1. Train base learner according to example distribution $\underline{d}^{(t)}$ and obtain hypothesis $h_t : \underline{x} \mapsto \{\pm 1\}$.
2. compute weighted error $\varepsilon_t = \sum_{n=1}^{N} d_n^{(t)} \, \mathbf{I}(y_n \neq h_t(x_n))$
3. compute hypothesis weight $\alpha_t = \frac{1}{2} \ln \frac{1 - \varepsilon_t}{\varepsilon_t}$
4. update example distribution $d_n^{(t+1)} = d_n^{(t)} \exp\left(-\alpha_t y_n h_t(\underline{x}_n)\right) / Z_t$

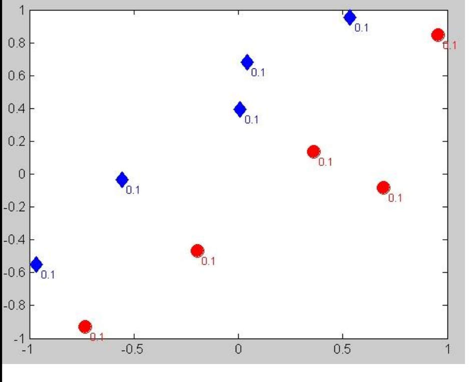
Z_t is a normalization factor

Output: final hypothesis $\boxed{f_{Ens}(\underline{x}) = \sum_{t=1}^{T} \alpha_t h_t(\underline{x})}$

AdaBoost

$t = 1$

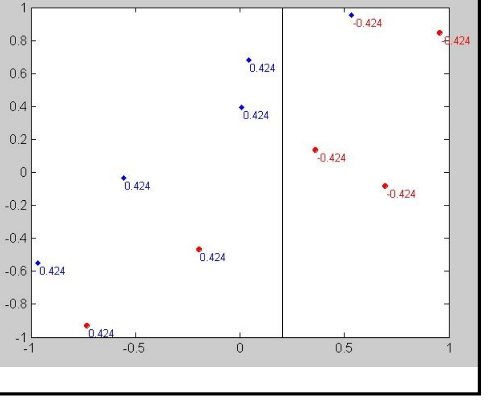
$d_n^{(1)} = 1/10 \quad N = 10$



$$\varepsilon_1 = \sum_{n=1}^{N} d_n^{(1)} \, \mathbf{I}(y_n \neq h_1(x_n)) = 0.3$$

$$\alpha_1 = \frac{1}{2} \ln \frac{1 - \varepsilon_1}{\varepsilon_1} = 0.424$$

$$\boxed{f_{Ens}(\underline{x}) = \alpha_1 h_1(x)}$$

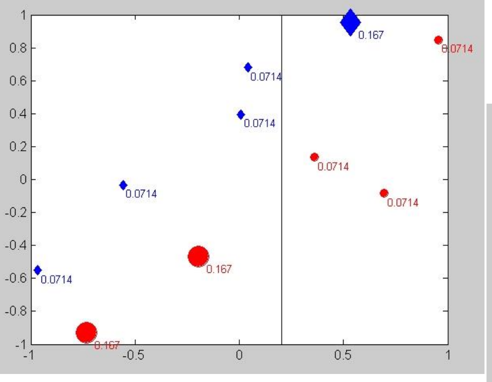


AdaBoost

t =2

$$d_n^{(2)} = d_n^{(1)} \exp\left(-\alpha_1 y_n h_1(\underline{x}_n)\right) \big/ Z_1$$

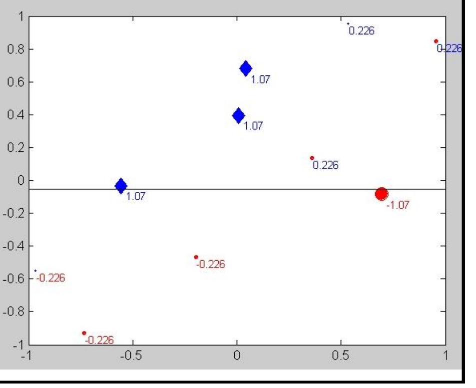
Z_t is a normalization factor



$$\varepsilon_2 = \sum_{n=1}^{N} d_n^{(2)} \mathbf{I}(y_n \neq h_2(x_n))$$

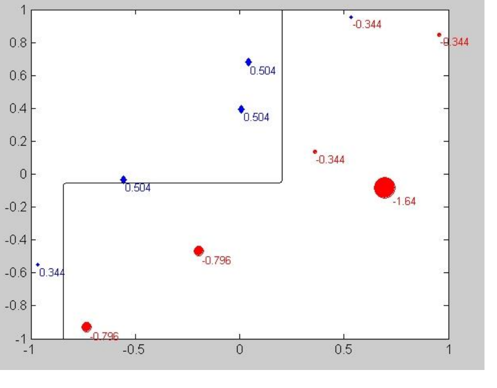
$$\alpha_2 = \frac{1}{2} \ln \frac{1 - \varepsilon_2}{\varepsilon_2}$$

$$\boxed{f_{Ens}(\underline{x}) = \alpha_1 h_1(x) + \alpha_2 h_2(x)}$$



AdaBoost

t =3



………………..

AdaBoost: Frame work

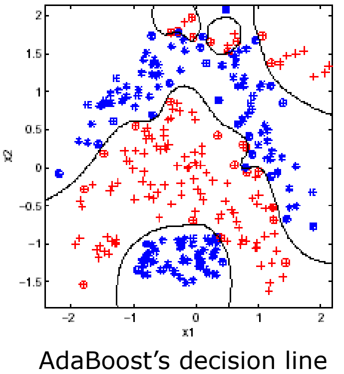
- Weak Learners used with Boosting
  - Decision stumps (axis parallel splits)
  - Decision trees (e.g. C4.5 by Quinlan 1996)
  - Multi-layer Neural networks (e.g. for OCR)
  - Radial basis function networks (e.g. UCI benchmarks, etc)

Decision trees:

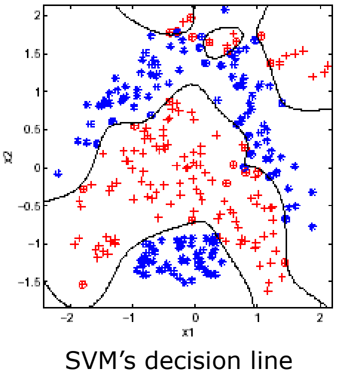
  - Hierarchical and recursive partitioning on the input space
  - Many approaches, usually axis parallel splits

AdaBoost: AdaBoost vs. SVM

- Comparison AdaBoost vs. SVM



AdaBoost's decision line



SVM's decision line

These decision lines are for a low noise case with similar generalization errors. In AdaBoost, RBF networks with 13 centers were used.

AdaBoost: Application

25

- **Application**
  - DT C4.5 as weak classifier
  - Spam, Zip Code OCR
  - Text classification: Schapire and Singer - Used stumps with normalized term frequency and multi-class encoding
  - OCR: Schwenk and Bengio (neural networks)
  - Natural language Processing: Collins; Haruno, Shirai and Ooyama
  - Image retrieval: Thieu and Viola
  - Medical diagnosis: Merle et al.
  - Fraud Detection: Rätsch & Müller 2001
  - Drug Discovery: Rätsch, Demiriz, Bennett 2002
  - Elect. Power Monitoring: Onoda, Rätsch & Müller 2000

AdaBoost: Information

26

Introduction	http://informatik.unibas.ch/lehre/ws06/cs232/_Downloads/Schapire_A_Short_Introduction_to_Boosting.pdf
Internet	http://www.boosting.org http://www.cs.princeton.edu/~schapire/boost.html
Conferences	Computational Learning Theory (COLT), Neural Information Processing Systems (NIPS), Int. Conference on Machine Learning (ICML), . . .
Journals	Machine Learning, Journal of Machine Learning Research, Information and Computation, Annals of Statistics
People	List available at http://www.boosting.org
Software	Only few implementations (algorithms 'too simple') (cf. http://www.boosting.org)