Hauptseminar Machine Learning

Boosting – Bagging – Stacking

Referent: Maximilian Schwinger
Betreuer: Prof. Kramer

Introduction – Ensemble Methods

Ensemble Methodes

- combine different classifiers
- try to get the best results

Introduction – Ensemble Methods

Idea: - scale down the mean square error

Strategies: - scale down bias
- scale down variance

Introduction – Ensemble Methods

Decomposition of mean square error

The mean squared error (MSE) of $\hat{\theta}$ is

$$E\left[(\hat{\theta} - \theta)^2\right] = E\left[\left(\hat{\theta} - E\left[\hat{\theta}\right] + E\left[\hat{\theta}\right] - \theta\right)^2\right]$$

$$= \underbrace{\left(E\left[\hat{\theta}\right] - \theta\right)^2}_{\text{Bias}^2} + \underbrace{E\left[\left(\hat{\theta} - E\left[\hat{\theta}\right]\right)^2\right]}_{\text{Variance}}$$

Introduction – Ensemble Methods

Some interesting ensemble methods are

- **boosting**
- **bagging**

and

- **stacking**

Introduction – Boosting

Origin of Boosting: PAC-learning (probably approximately correct)
L.G.Valiant A theory of the learnable (1984)

Idea: Boost “weak” learners to a “strong” learner

Practical Application: OCR, speech recognition

Theoretical Specification – Boosting(1)

PAC Model

Let C be a concept class over X. C is PAC-learnable, if there exists an algoritm L, with the properties

- for every $0 \leq \varepsilon \leq \frac{1}{2} and\ 0 \leq \delta \leq \frac{1}{2}$
- for every distributin D on X
- for every $c \in C$
- with probability at least $1-\delta$

L outputs a hypothesis $h \in C$ satisfying error $(h) \leq \varepsilon$

Theoretical Specification – Boosting(2)

Weak PAC learner
can achieve the PAC criterion only for

- fixed values ε_0
- and δ_0

Theoretical Specification – Boosting(3)

- It is relatively easy to get an efficient weak learner, even trivial. E.g. In a sample set composed of more then 60% positive samples a learner could always predict positive.

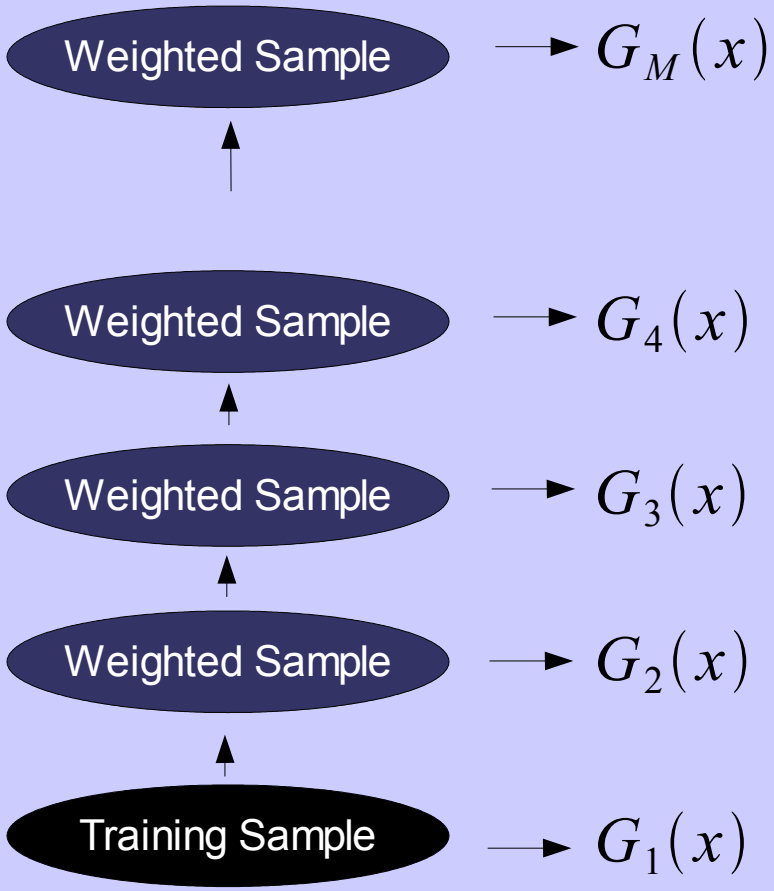
- We use a weak learner repeatedly on modified versions of the data and combine the resulting hypotheses to get better results.

- We combine the weighted hypotheses

Example: AdaBoost.M1 – Boosting(1)

- AdaBoost.M1 was invented by Freund and Schapire in 1997
- Considers a two-class-problem
- Output can be coded as $Y \in \{-1\,;1\}$
- G() produces a prediction from a vector of prediction variables X

Example: AdaBoost.M1 – Boosting(2)



final prediction obtained by using a weighted 'majority vote'

$$G(x) = sign\left(\sum_{m=1}^{M} \alpha_m G_m(x)\right)$$

Example: AdaBoost.M1 – Boosting(3)

The Modification:

- weights are assigned to training observations
- initial weight is 1/N
- misclassified observations get their weights increased

Example: AdaBoost.M1 – Boosting(4)

The Code

1. Initialize the observation weights
2. For m=1 to M do

2.1. Fit a classifier $G_m(x)$ to the training data using weights w_i.

2.2. Compute

$$err_m = \frac{(\sum_{i=i}^{N} W_i I(Y_i \neq G_m(x_i)))}{\sum_{i=1}^{N} w_i}$$

2.3. Compute

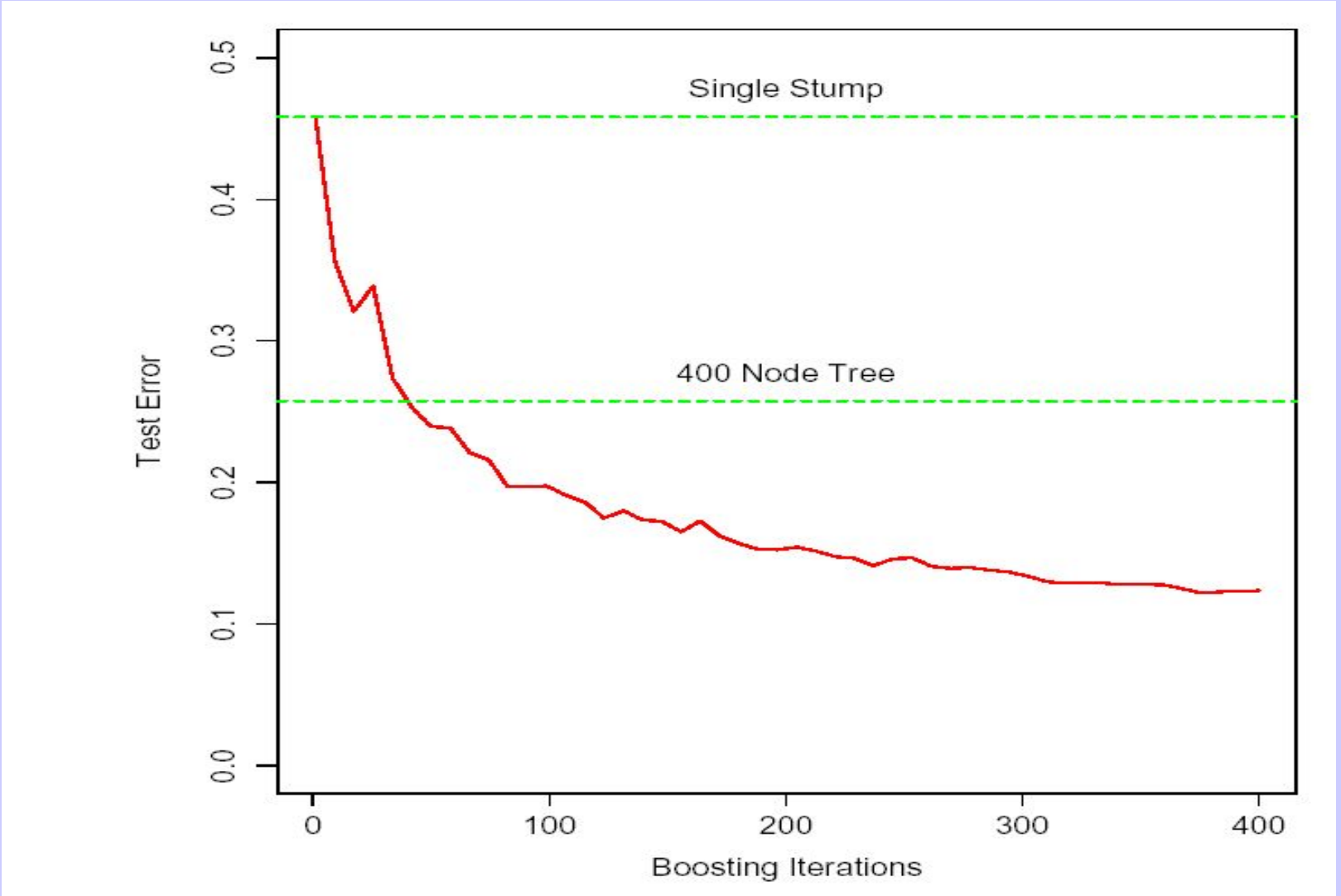
$$\alpha_m = \log(\frac{(1-err_m)}{err_m})$$

2.4. Set $w \leftarrow w_i e[\alpha_m \times I(y_i \neq G_m(x_i))], \quad i = 1,2.... N$

3. Output $G(x) = sign[\sum_{m=1}^{M} \alpha_m G_m(x)]$

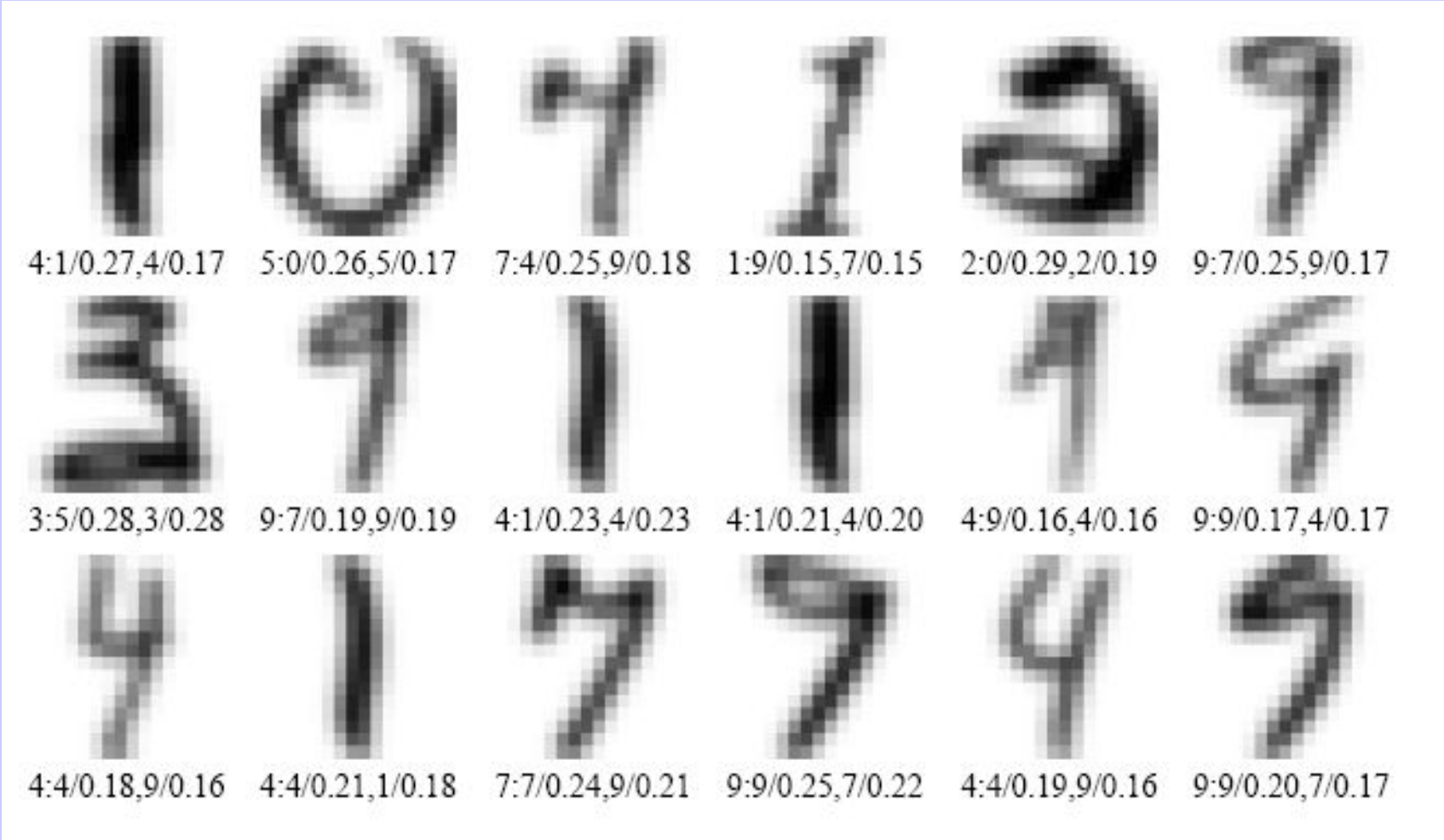
Example: AdaBoost.M1 – Boosting(5)

Example for Performance



Example: AdaBoost.M1 – Boosting(6)

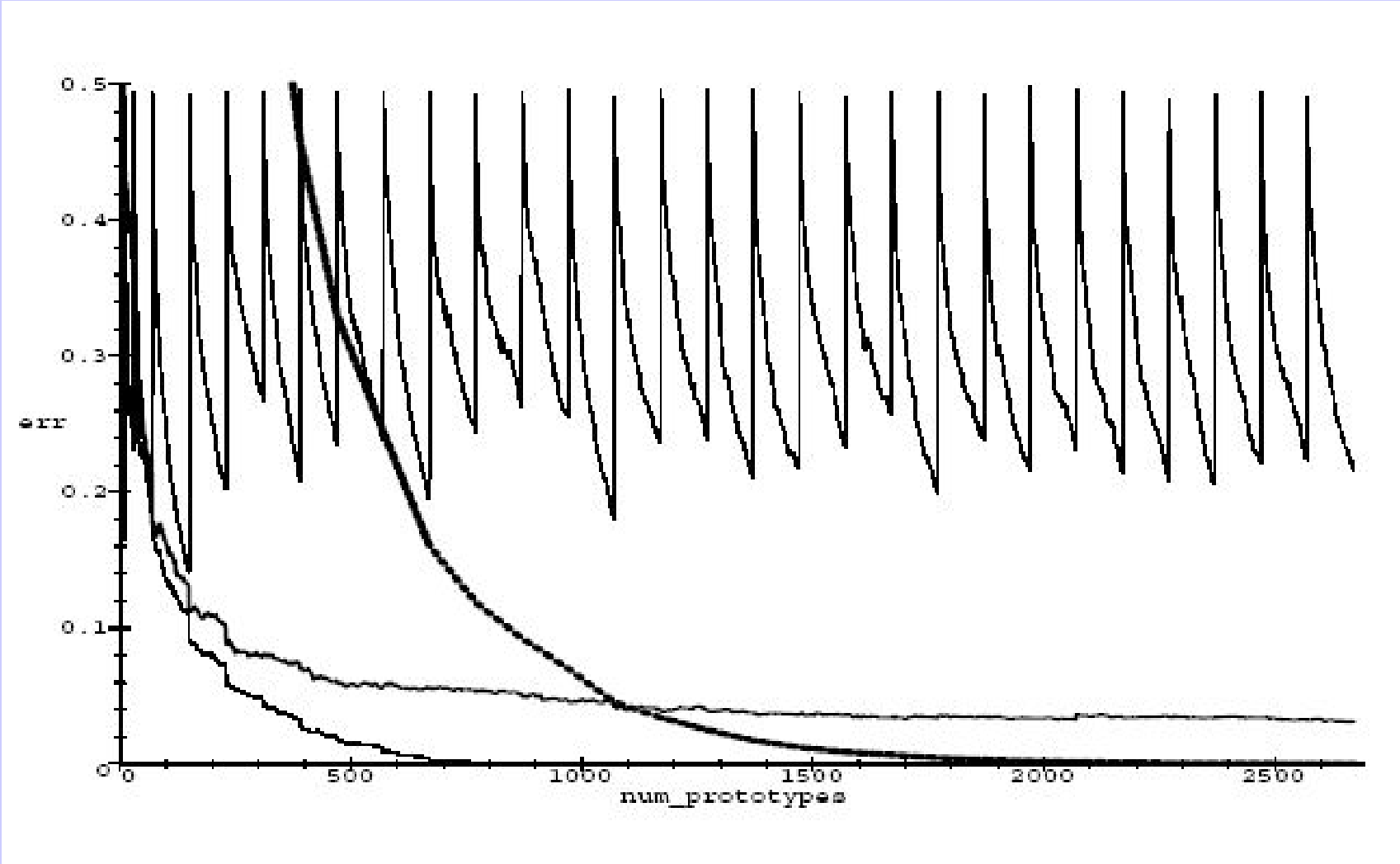
Example for practical application



"Experiments with a New Boosting Algorithm", Yoav Freund and Robert E. Schapire, AT&T Laboratories 1996

Example: AdaBoost.M1 – Boosting(7)

Example for practical application



"Experiments with a New Boosting Algorithm", Yoav Freund and Robert E. Schapire, AT&T Laboratories 1996

Résumé - Boosting

Pros: - It's easy to find weak efficient learners
- good results with little work

Cons: - Problem with 'noisy' data, because misclassification will have a higher weight

Introduction – Bagging

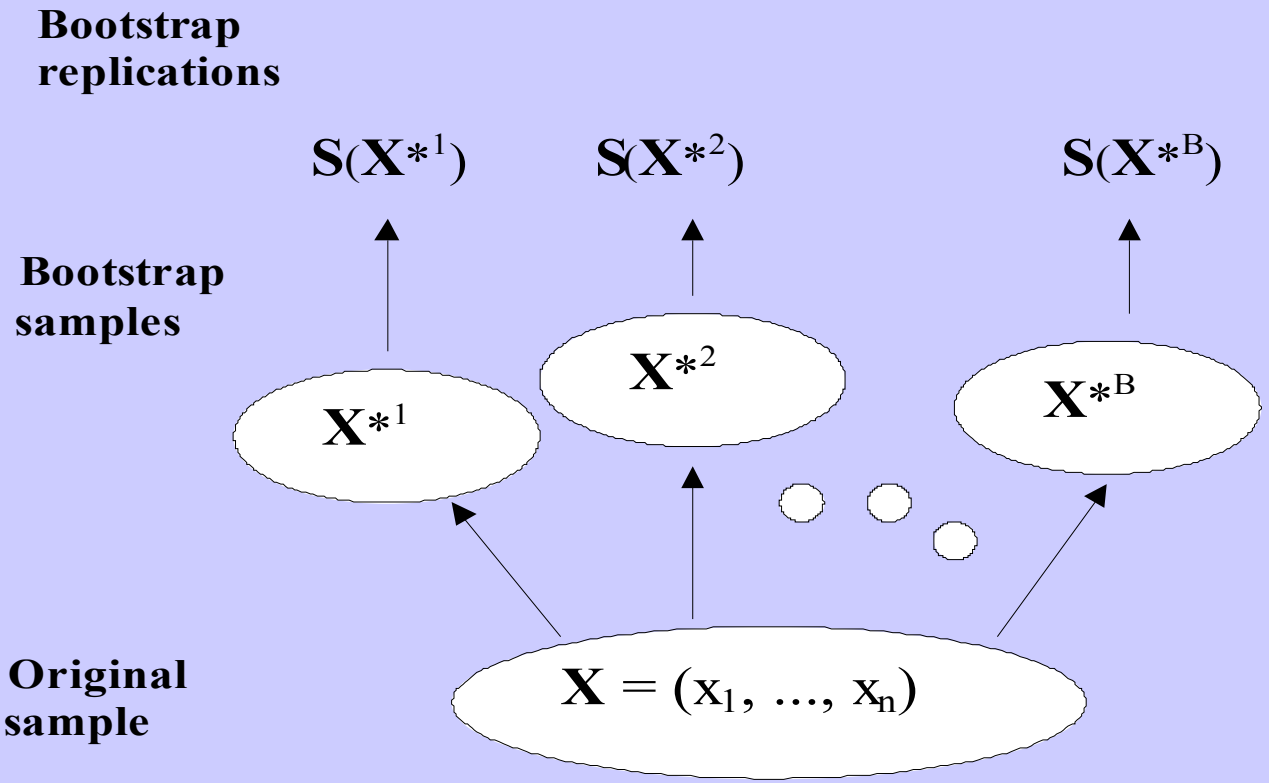
Origin of Bagging: Bagging Predictors, Leo Breiman, 1994
Name comes from **B**ootstrap **Agg**regat**ing**

Idea: Generate more learning samples and create a predictor using the predictions that are based on all the learning samples

Practical Application: clinical applications

Assumption: Bootstrapping – Bagging

Bootstrapping is the creation of new samples from the original sample



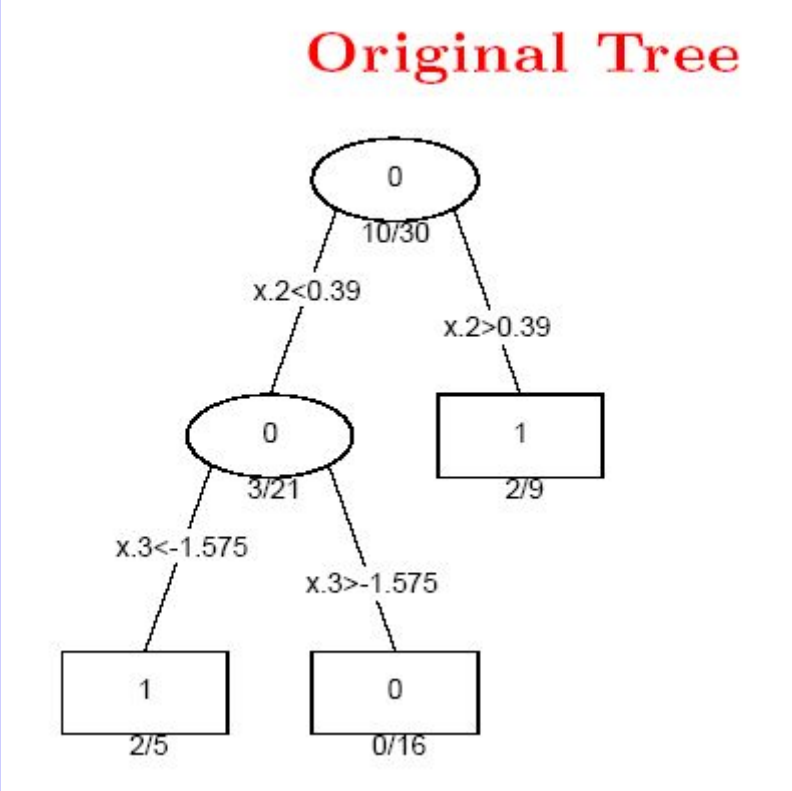
Yaochu Jin Future,Technology Research, Honda R&D Europe (Germany), 2000

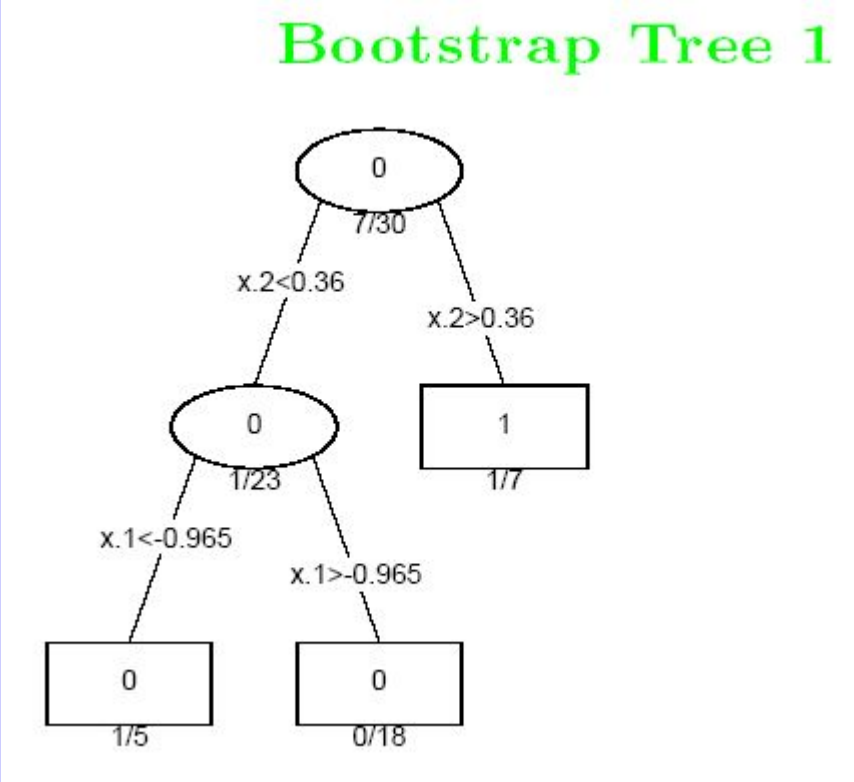
Theoretical Specification – Bagging(1)

- divide data set into a test set and a learning set
- create a classifier from the learning set
- create bootstrap replications from learning sample
- create classifiers from the replicants
- assemble the individual classifiers to a single one by taking the average (majority vote)

Example – Bagging(1)

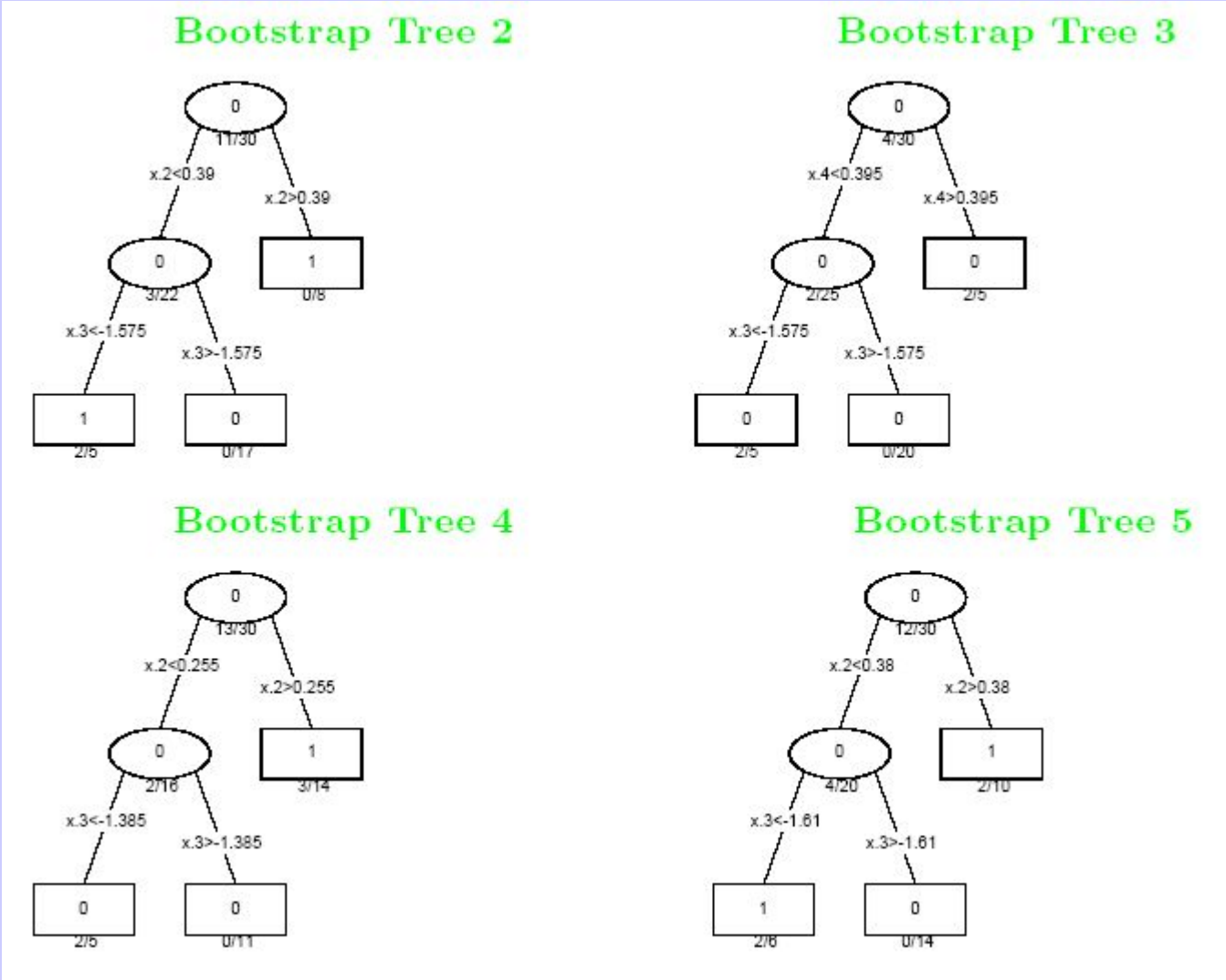
tree with simulated data





Elements of Statistical Learning (c) Hastie, Tibshirani & Friedman 2001

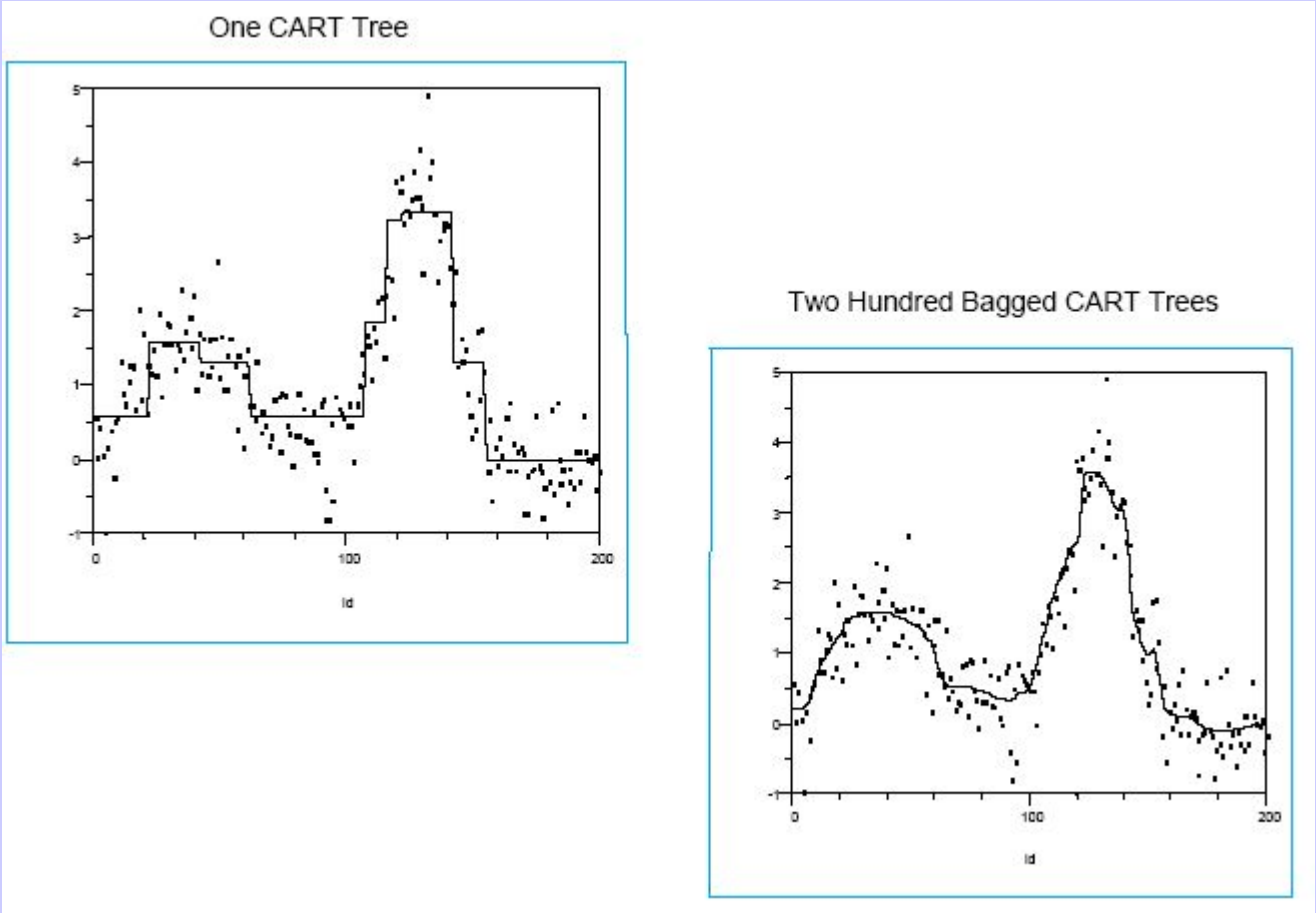
Example – Bagging(2)



Elements of Statistical Learning (c) Hastie, Tibshirani & Friedman 2001

Example – Bagging(3)

CART(Classification And Regression Trees)

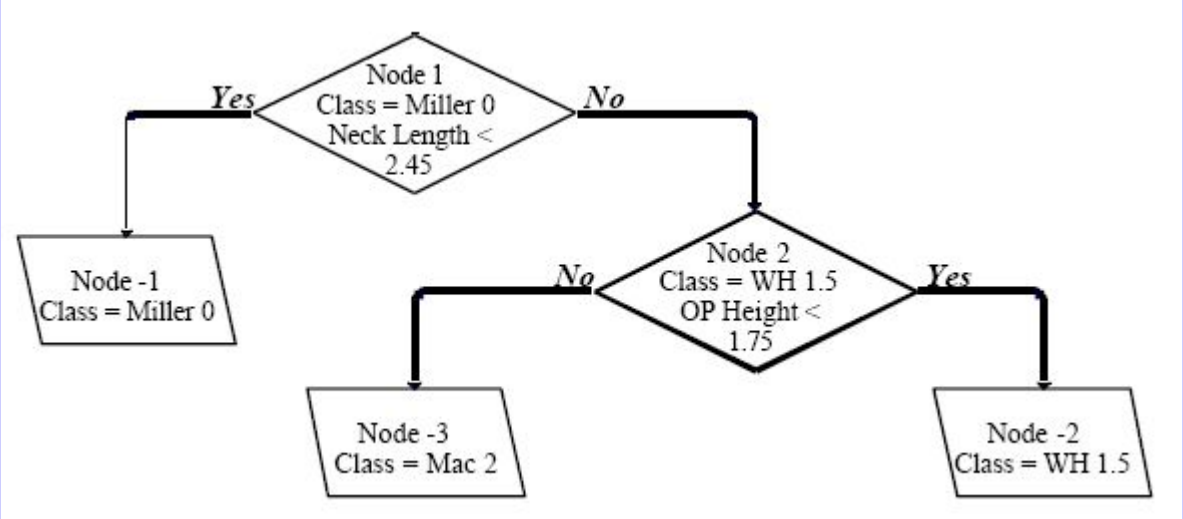


Rick Higgs & Dave Cummins, Statistical & Information Sciences, Lilly Research Laboratories, 2003

Example – Bagging(4)

CART(Classification And Regression Trees) – short description

- especially used in clinical issues



Roger J. Lewis, M.D., Ph.D., An Introduction to Classification and Regression Tree (CART) Analysis, 2000

- implements binary recursiv partitioning procedure

Résumé – Bagging

Pros: - Good to improve unstable methodes by scaling down the variance

Cons: - large number of bootstrap replicants
- performance

Introduction - Stacking

Origin of Stacking: Stacking was introduced in D.Wolpert's "Stacked generalization", Neural Networks, 5(2):241-260, 1992.

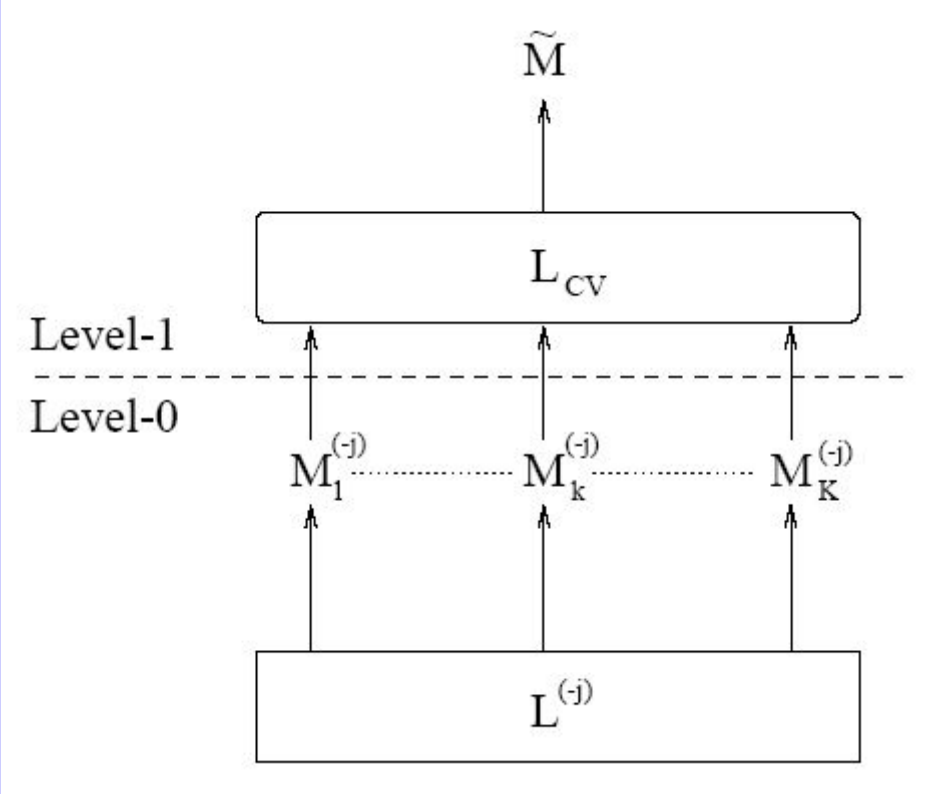
Idea: Use the output of more than one classificator to get optimal results. Kind of meta - classificator

Theoretical Specification - Stacking(1)

- We use N different learning algorithms L on a data set S with examples $s_i = (x_i, y_i)$, with feature vectors y_i and x_i classifications
- Classifiers C_i are created with $C_i = L_i(S)$ (base classifiers)
- A 'meta-classifiere' has to be created, that combines the output of the C_i

Theoretical Specification - Stacking(2)

Principle of Stacking



(Issues in Stacked Generalisation, Kai Ming Ting, Ian H. Witten, 1998)

Example - Stacking(1)

Stacking C4.5, NB and IB1 in Level 0, MLR in Level 1
(Issues in Stacked Generalization, Kai Ming Ting, Ian H. Witten, 1998)

C4.5
: induces classification rules structured as decision trees

NB
: Naive Bayesian algorithm

IB1
: instance-based learning algorithm. For every class one example instance is saved. In the classification-process the tested instance's class is simply the class of the 'closest' example.

MLR
: variant of a least-square linear regression

Example - Stacking(2)

MLR

The input Data for the 1st-level may have attributes, such as

- probabilities (Model $\tilde{M}'$)
- classes (Model $\tilde{M}$)

in the first case the linear regression for class l is

$$LR_l(x) = \sum_k^K \alpha_{kl} P_{kl}(x)$$

In the second case the attributes are unordered, nominals. We map them to 1 and 0. We set P 1, if x is class l, 0 otherwise

Example - Stacking(3)

MLR

$$LR_l(x) = \sum_k^K \alpha_{kl} P_{kl}(x)$$

Choose α_{kl} to minimize

$$\sum_j \sum_{(y_n, x_n) \in L_j} (y_n - \sum_k \alpha_{kl} P_{kl}^{(-j)}(x_n))$$

Now we compute a LR for all classes and assign that instance to class l, where LR has the greatest value

$$LR_l(x) > LR_l'(x) \; for \; all \; l' \neq l$$

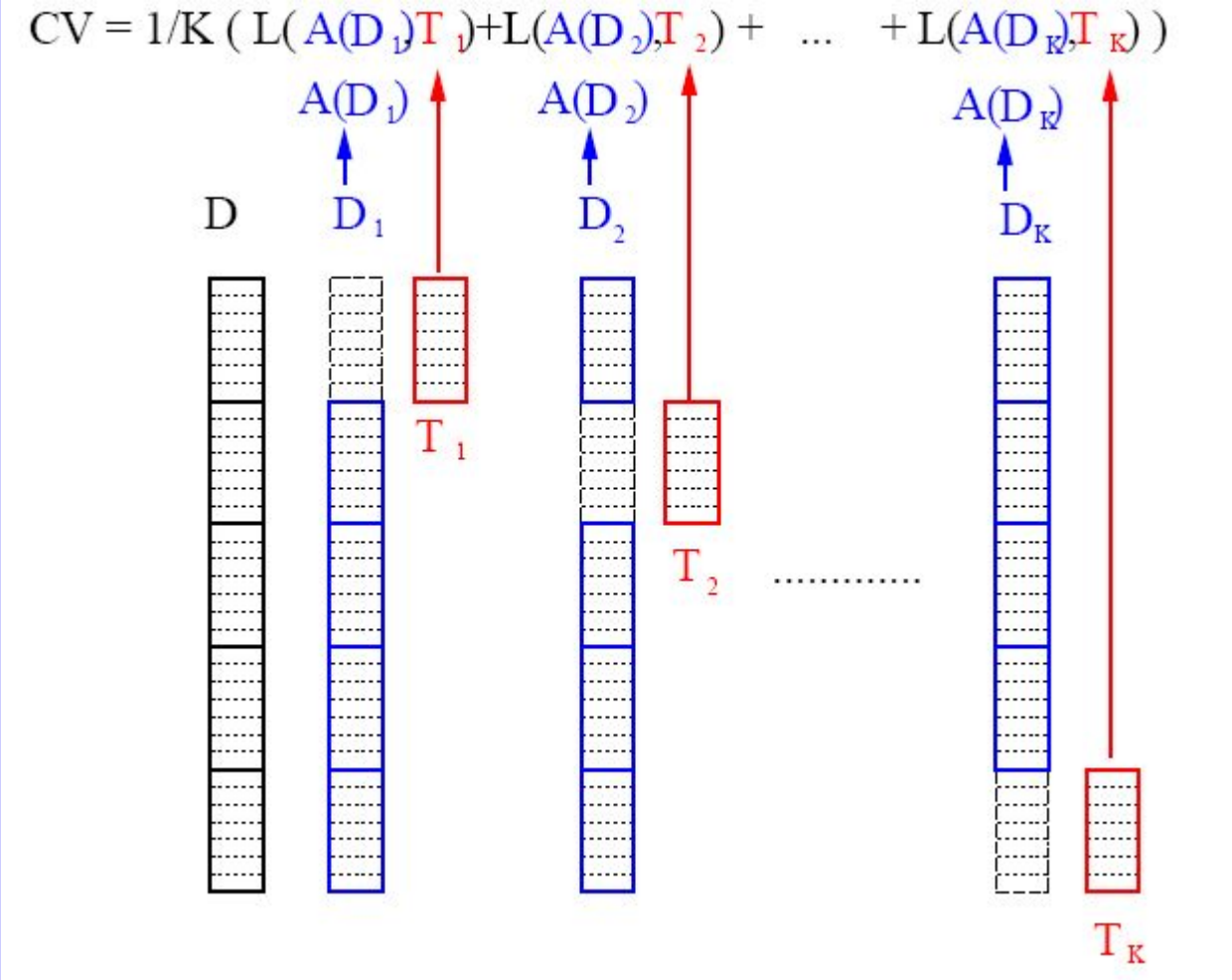
Example - Stacking(4)

J-fold cross validation

- divide sample set in j partitions
- use 1 partition as test data, the others as learning data
- repeat last step, use every partition as test data
- generate 'majority vote'

Example - Stacking(5)

J-fold cross validation



Example - Stacking(6)

Datasets	# Samples	# Classes	# Attr & Type
Led24	200/5000	10	10N
Waveform	300/5000	3	40C
Horse	368	2	3B+12N+7C
Credit	690	2	4B+5N+6C
Vowel	990	11	10C
Euthyroid	3163	2	18B+7C
Splice	3177	3	60N
Abalone	4177	3	1N+7C
Nettalk(s)	5438	5	7N
Coding	20000	2	15N

N-nominal; B-binary; C-Continuous.

(Issues in Stacked Generalization, Kai Ming Ting, Ian H. Witten, 1998)

Example - Stacking(7)

Datasets	Level-0 Generalizers			
	C4.5	NB	IB1	BestCV
Led24	35.4	35.4	32.2	32.8 ±0.6
Waveform	31.8	17.1	26.2	17.1 ±0.3
Horse	15.8	17.9	15.8	17.1 ±1.6
Credit	17.4	17.3	28.1	17.4 ±1.2
Vowel	22.7	51.0	2.6	2.6 ±0.2
Euthyroid	1.9	9.8	8.6	1.9 ±0.3
Splice	5.5	4.5	4.7	4.5 ±0.4
Abalone	41.4	42.1	40.5	40.1 ±0.6
Nettalk(s)	17.0	15.9	12.7	12.7 ±0.4
Coding	27.6	28.8	25.0	25.0 ±0.3

(Issues in Stacked Generalization, Kai Ming Ting, Ian H. Witten, 1998)

Average error rates of C4.5, NB, IB1, and BestCV – the best among them selected using J-fold cross-validation. The standard errors are shown in the last column.

Example - Stacking(8)

Datasets		Level-1 model, $\tilde{\mathcal{M}}$				Level-1 model, $\tilde{\mathcal{M}}'$			
	BestCV	C4.5	NB	IB1	MLR	C4.5	NB	IB1	MLR
Led24	32.8	34.0	32.4	35.0	33.3	41.7	35.7	32.1	**31.3**
Waveform	17.1	17.7	19.2	18.7	17.2	20.6	17.6	17.8	**16.8**
Horse	17.1	16.9	**14.9**	17.6	16.3	18.0	18.5	17.7	15.2
Credit	17.4	18.4	16.1	16.9	17.4	15.4	15.9	**14.3**	16.2
Vowel	2.6	2.6	3.8	3.6	2.6	2.7	7.2	3.3	**2.5**
Euthyroid	**1.9**	**1.9**	**1.9**	**1.9**	**1.9**	2.2	4.3	2.0	**1.9**
Splice	4.5	3.9	3.9	**3.8**	**3.8**	4.0	3.9	**3.8**	**3.8**
Abalone	40.1	38.5	38.5	38.2	38.1	43.3	**37.1**	39.2	38.3
Nettalk(s)	12.7	12.4	11.9	12.4	12.6	14.0	14.6	12.0	**11.5**
Coding	25.0	23.2	23.1	23.2	23.2	22.3	21.2	21.2	**20.7**

(Issues in Stacked Generalization, Kai Ming Ting, Ian H. Witten, 1998)

Average error rate for stacking C4.5, NB and IB1

Résumé – Stacking

Pros: - Takes the best from everything

Cons: - You have to care about the handling of multiple methodes

Bibliographie

Elements of Statistical Learning,
Trevor Hastie, Robert Tibshirani, Jerome Friedman, Springer, 2001.

Data Mining: Practical Machine Learning Tools and Techniques with JAVA Implementations,
Ian H. Witten, Eibe Frank, Morgan Kaufmann, 2000.

Eric Bauer and Ron Kohavi,
An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants.
Machine Learning, 36(1/2), 105-139

Kai Ming Ting, Ian H. Witten
Issues in Stacked Generalization
1998

Roger J. Lewis, M.D., Ph.D.,
An Introduction to Classification and Regression Tree (CART) Analysis
2000

Rick Higgs & Dave Cummins
Technical Report,
Lilly Research Laboratories, 2003

Bibliographie

Tibshirani & Friedman
Elements of Statistical Learning (c) Hastie
2001

Yaochu Jin Future
Technology Research
Honda R&D Europe (Germany), 2000

Yoav Freund and Robert E. Schapire,
“Experiments with a New Boosting Algorithm”,
AT&T Laboratories 1996

Bernhard Pfahringer
Winning the KDD99 Classification Cup: Bagged Boosting
Austrian Research Institute for AI