Data Mining - Evaluation of Classifiers

Lecturer: JERZY STEFANOWSKI
Institute of Computing Sciences
Poznan University of Technology
Poznan, Poland

Lecture 4
SE Master Course
2008/2009 revised for 2010

Outline

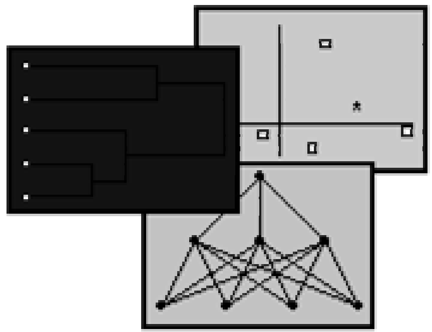
1. Evaluation criteria – preliminaries.
2. Empirical evaluation of classifiers
    - Hold-out
    - Cross-validation
    - Leaving one out and other techniques
3. Other schemes for classifiers.

Classification problem – another way …

- General task: assigning a decision class label to a set of unclassified objects described by a fixed set of attributes (features).
- Given a set of pre-classified examples, discover the classification knowledge representation,
  - to be used either as a *classifier* to classify new cases (**<u>a predictive perspective</u>**)
    or
    to *describe* classification situations in data
    (<u>a descriptive perspective</u>).
  - *Supervised* learning: classes are known for the examples used to build the classifier.

Approaches to learn classifiers

- Decision Trees
- Rule Approaches
- Logical statements (ILP)
- Bayesian Classifiers
- Neural Networks
- Discriminant Analysis
- Support Vector Machines
- k-nearest neighbor classifiers
- Logistic regression
- Artificial Neural Networks
- Genetic Classifiers
- …



Discovering and evaluating classification knowledge

Creating classifiers is a multi-step approach:

- Generating a classifier from the given learning data set,
- Evaluation on the test examples,
- Using for new examples.

Train and test paradigm!

Evaluation criteria (1)

- *Predictive* (*Classification*) *accuracy*: this refers to the ability of the model to correctly predict the class label of new or previously unseen data:
  - accuracy = % of testing set examples correctly classified by the classifier
- *Speed*: this refers to the computation costs involved in generating and using the model
- *Robustness*: this is the ability of the model to make correct predictions given noisy data or data with missing values

Evaluation criteria (2)

- *Scalability*: this refers to the ability to construct the model efficiently given large amount of data
- *Interpretability*: this refers to the level of understanding and insight that is provided by the model
- *Simplicity*:
  - decision tree size
  - rule compactness
- Domain-dependent quality indicators

Predictive accuracy / error

- General view (statistical learning point of view):
- Lack of generalization – prediction risk:

$$R(f) = E_{xy} L(y, f(x))$$

- where $L(y, \hat{y})$ is a loss or cost of predicting value $\hat{y}$ when the actual value is y and *E* is expected value over the joint distribution of all (*x,y*) for data to be predicted.
- Simple classification $\rightarrow$ zero-one loss function

$$L(y, \hat{y}) = \begin{cases} 0 & if\, y = f(y) \\ 1 & if\, y \neq f(y) \end{cases}$$

Evaluating classifiers – more practical …

Predictive (classification) accuracy (0-1 loss function)

- Use testing examples, which do not belong to the learning set
  - N_t – number of testing examples
  - N_c – number of correctly classified testing examples
- Classification accuracy: $$\eta = \frac{N_c}{N_t}$$
- (Misclassification) Error: $$\varepsilon = \frac{N_t - N_c}{N_t}$$
- Other options:
  - analysis of confusion matrix

A confusion matrix

	Predicted		
Original classes	K_1	K_2	K_3
K_1	50	0	0
K_2	0	48	2
K_3	0	4	46

- Various measures could be defined basing on values in a confusion matrix.

Confusion matrix and cost sensitive analysis

	Predicted		
Original classes	K_1	K_2	K_3
K_1	50	0	0
K_2	0	48	2
K_3	0	4	46

$$C(\varepsilon) = \sum_{i=1}^{r} \sum_{j=1}^{r} n_{ij} \cdot c_{ij}$$

- Costs assigned to different types of errors.
- Costs are unequal
- Many applications:
  loans, medical diagnosis, fault detections,
  spam …
- Cost estimates may be difficult to be acquired from real experts.

Other measures for performance evaluation

- Classifiers:
  - Misclassification cost
  - Lift
  - Brier score, information score, margin class probabilities
  - Sensitivity and specificity measures (binary problems), ROC curve $\rightarrow$ AUC analysis.
  - Precision and recall, F-measure.
- Regression algorithms
  - Mean squared error
  - Mean absolute error and other coefficient
- More will be presented during next lectures
  - Do not hesitate to ask any questions or read books!

Theoretical approaches to evaluate classifiers

- So called COLT
  - COmputational Learning Theory – subfield of Machine Learning
  - PAC model (Valiant) and statistical learning (Vapnik Chervonenkis Dimension $\rightarrow$ VC)
- Asking questions about general laws that may govern learning concepts from examples
  - Sample complexity
  - Computational complexity
  - Mistake bound

COLT typical research questions

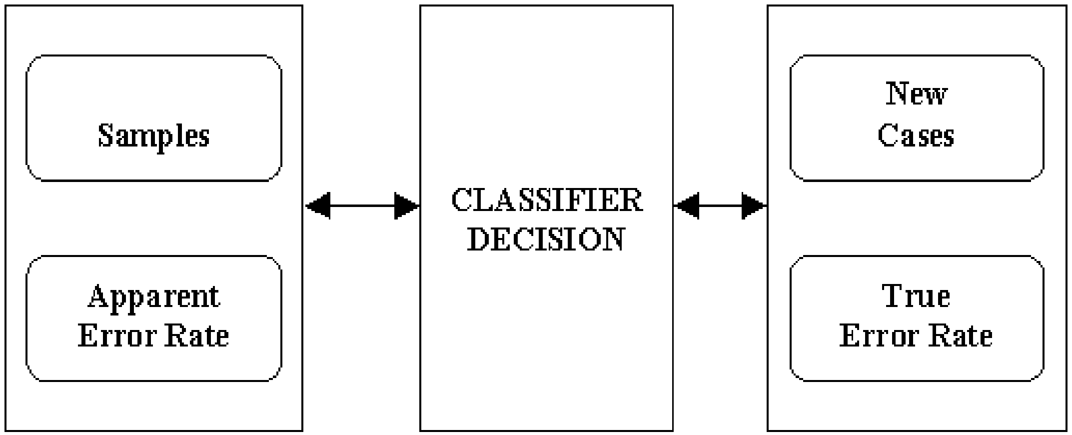
- Is it possible to identify problems that are inherently difficult of easy, independently of the learning algorithms?
- What is the number of examples necessary or sufficient to assure successful learning?
- Can one characterize the number of mistakes that an algorithm will make during learning?
  - The probability that the algorithm will output a successful hypothesis.
- All examples available or incremental / active approaches?
- Read more in T.Mitchell’s book – chapter 7.
  or P.Cichosz (Polish) coursebook – Systemy uczące się.

Experimental evaluation of classifiers

- How predictive is the model we learned?
- Error on the training data is *not* a good indicator of performance on future data
  - ***Q: Why?***
  - A: Because new data will probably not be **exactly** the same as the training data!
- Overfitting – fitting the training data too precisely - usually leads to poor results on new data.
  - Do not learn too much peculiarities in training data;
    think about generality abilities!
  - We will come back to it latter during the lecture on *pruning* structures of classifiers.

Empirical evaluation

- The general paradigm $\rightarrow$ „Train and test”
- Closed vs. open world assumption.
- The rule of a supervisor?
- Is it always probably approximate correct?
- How could we estimate with the smallest error?



Experimental estimation of classification accuracy

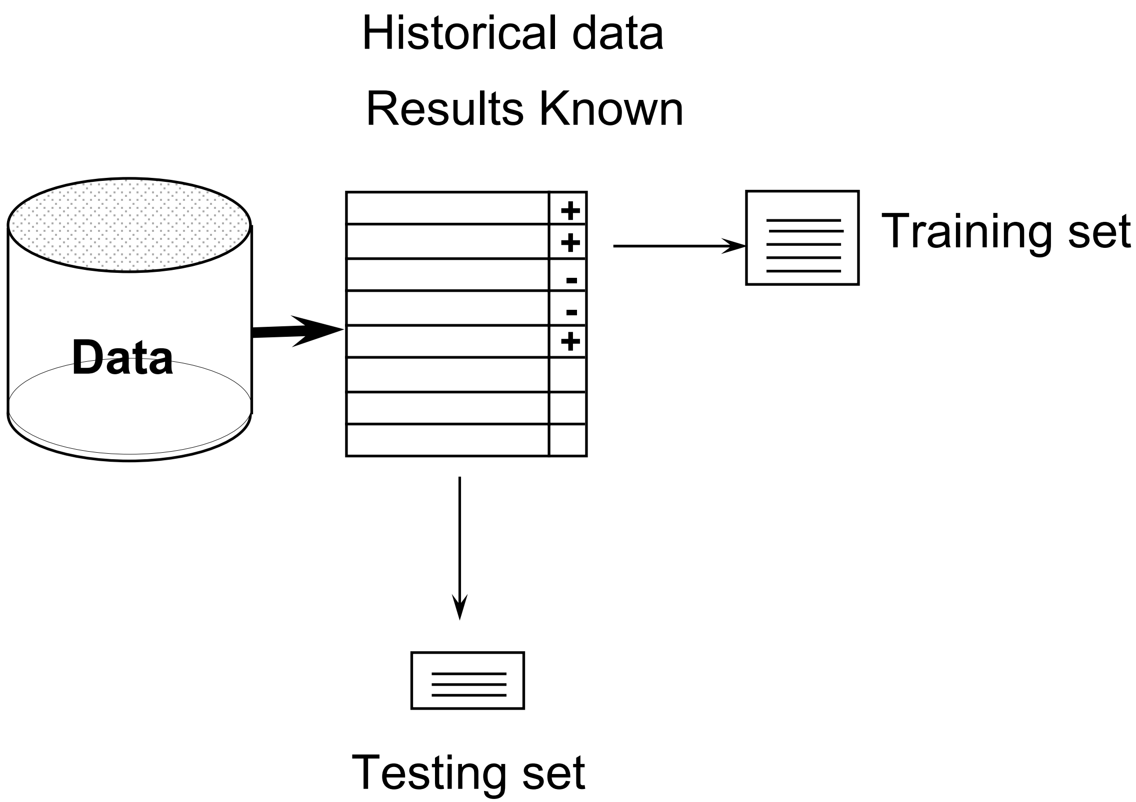
Random partition into **train and test** parts:

- Hold-out
  - use two independent data sets, e.g., training set (2/3), test set(1/3); random sampling
  - repeated hold-out
- *k*-fold cross-validation
  - randomly divide the data set into *k* subsamples
  - use *k-1* subsamples as training data and one sub-sample as test data --- repeat *k* times
- Leave-one-out for small size data

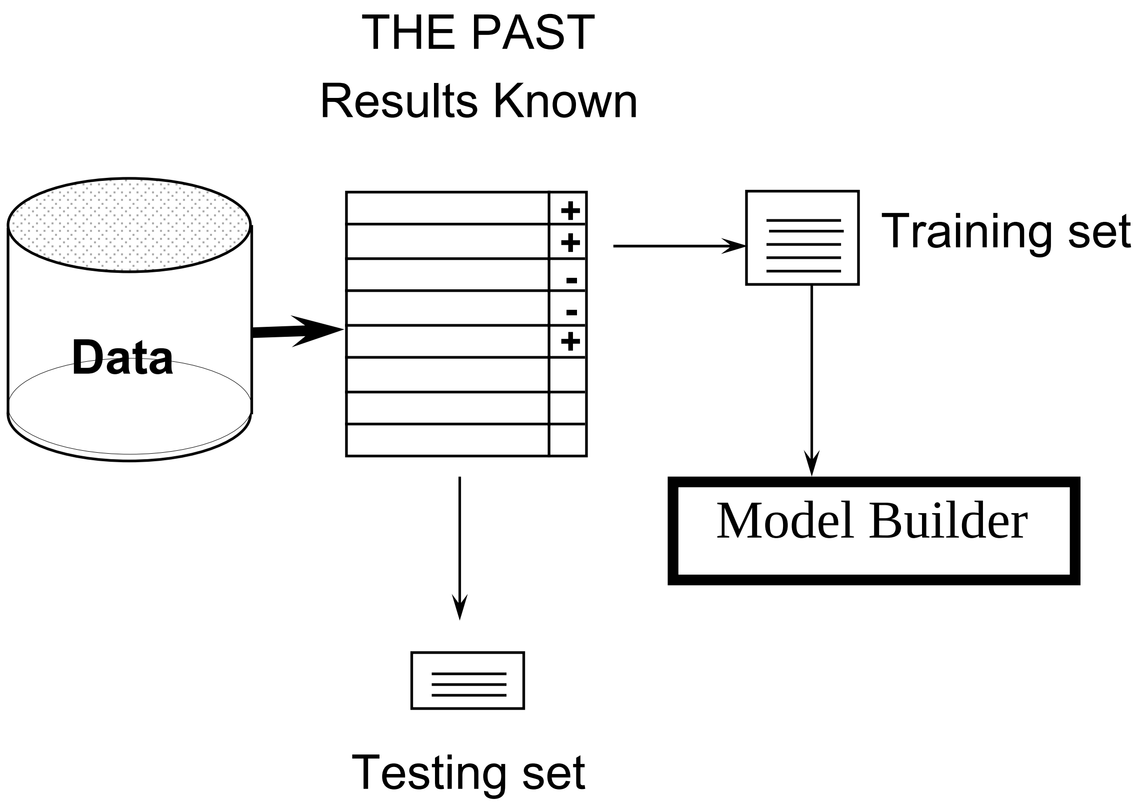
Evaluation on “LARGE” data, hold-out

- A simple evaluation is sufficient
  - Randomly split data into training and test sets (usually 2/3 for train, 1/3 for test)
- Build a classifier using the *train* set and evaluate it using the *test* set.

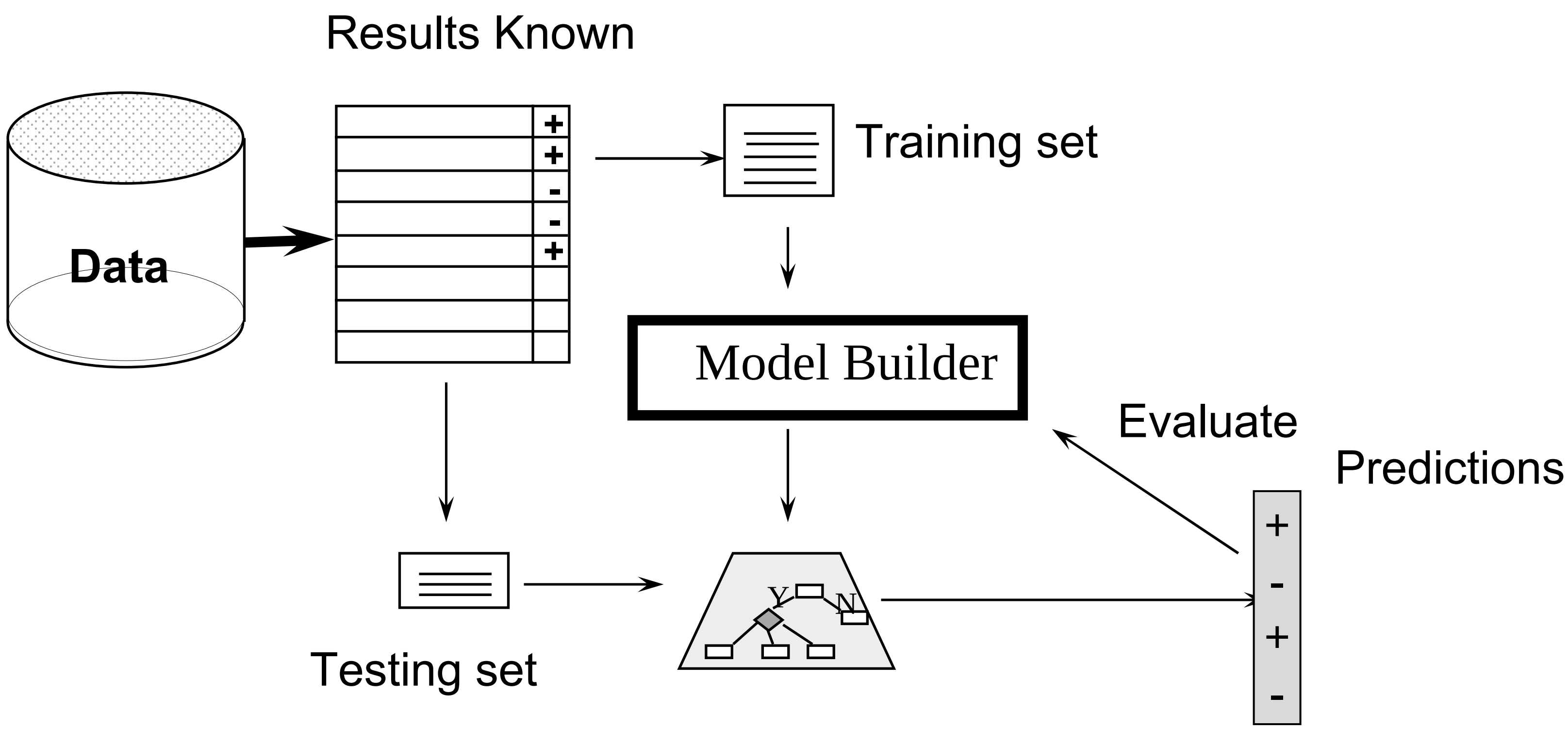
Step 1: Split data into train and test sets



Step 2: Build a model on a training set



Step 3: Evaluate on test set



Remarks on hold-out

- It is important that the test data is not used *in any way* to create the classifier!
- One random split is used for really large data
- For medium sized $\rightarrow$ **repeated hold-out**
- Holdout estimate can be made more reliable by repeating the process with different subsamples
  - In each iteration, a certain proportion is randomly selected for training (possibly with stratification)
  - The error rates (classification accuracies) on the different iterations are averaged to yield an overall error rate
  - Calculate also a standard deviation!

Repeated holdout method, 2

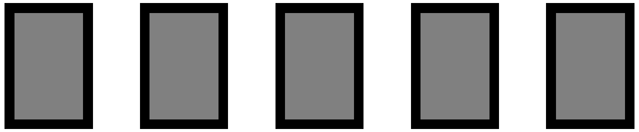
- Still not optimum: the different test sets usually overlap (difficulties from statistical point of view).
- Can we prevent overlapping?

Cross-validation

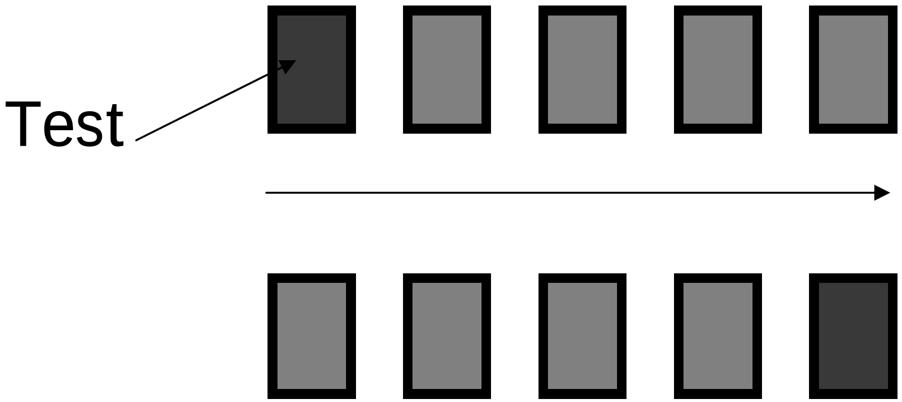
- *Cross-validation* avoids overlapping test sets
  - First step: data is split into *k* subsets of equal size
  - Second step: each subset in turn is used for testing and the remainder for training
- This is called *k-fold cross-validation*
- Often the subsets are stratified before the cross-validation is performed
- The error estimates are averaged to yield an overall error estimate

Cross-validation example:

- Break up data into groups of the same size



- Hold aside one group for testing and use the rest to build model



- Repeat

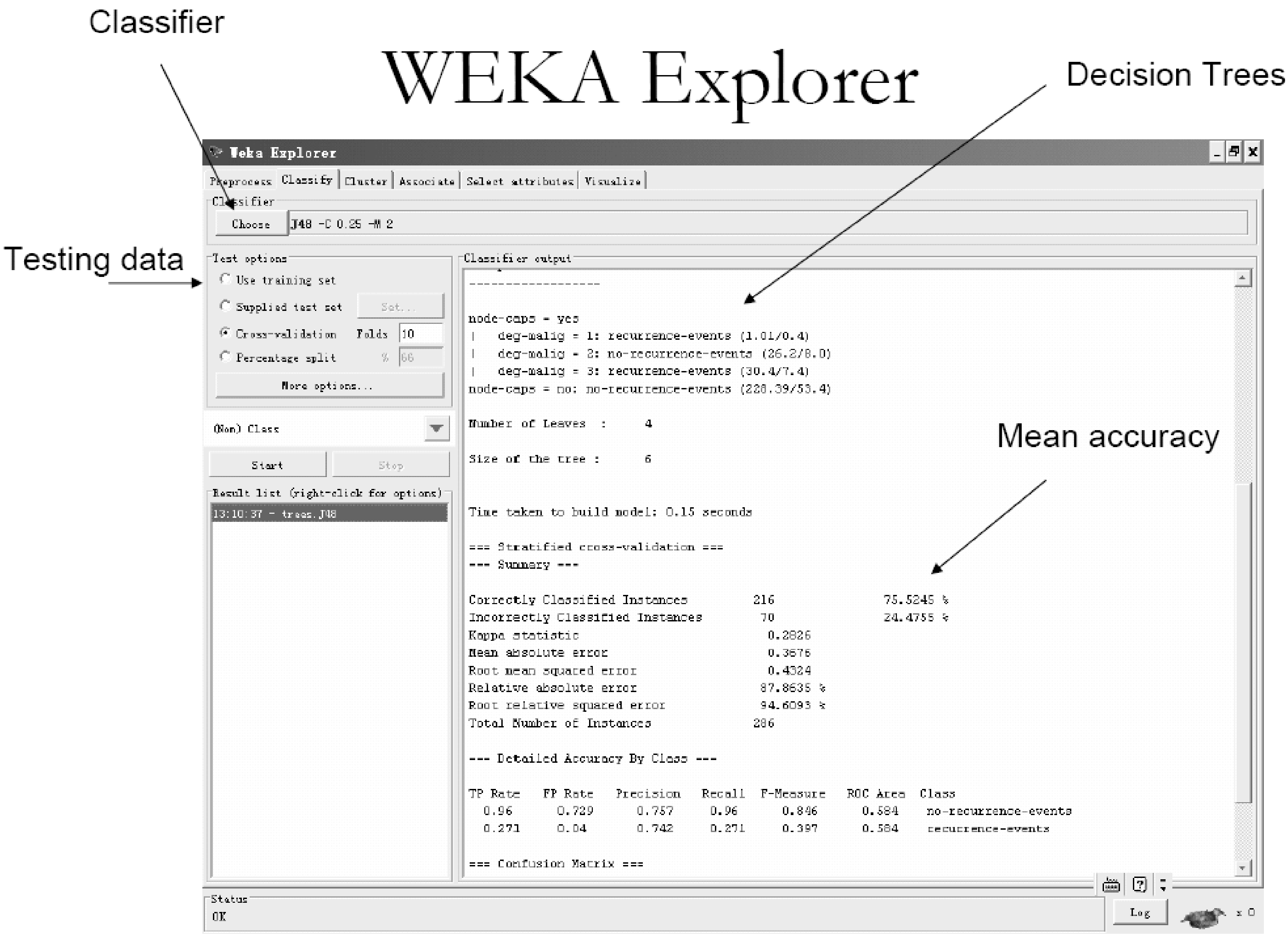
More on 10 fold cross-validation

- Standard method for evaluation: stratified ten-fold cross-validation
- Why ten? Extensive experiments have shown that this is the best choice to get an accurate estimate
  (since CART book by Breiman, Friedman, Stone, Olsen 1994)
  However, other splits – e.g. 5 cv – are also popular.
- Also the standard deviation is essential for comparing learning algorithms.
- Stratification reduces the estimate's variance!
- Even better: repeated stratified cross-validation
  - E.g. ten-fold cross-validation is repeated more times and results are averaged (reduces the variance)!

Leave-One-Out cross-validation

- Leave-One-Out:
  a particular form of cross-validation:
  - Set number of folds to number of training instances
  - i.e., for *n* training instances, build classifier *n* times but from *n* -1 training examples …
- Makes best use of the data.
- Involves no random sub-sampling.
- Quite computationally expensive!

WEKA Explorer



Comparing data mining algorithms

- Frequent situation: we want to know which one of two learning schemes performs better.
- Note: this is domain dependent!
- Obvious way: compare 10-fold CV estimates.
- Problem: variance in estimate.
- Variance can be reduced using repeated CV.
- However, we still don’t know whether the results are reliable.
  - There will be a long explanation on this topic in future lectures

Comparing two classifiers on the same data

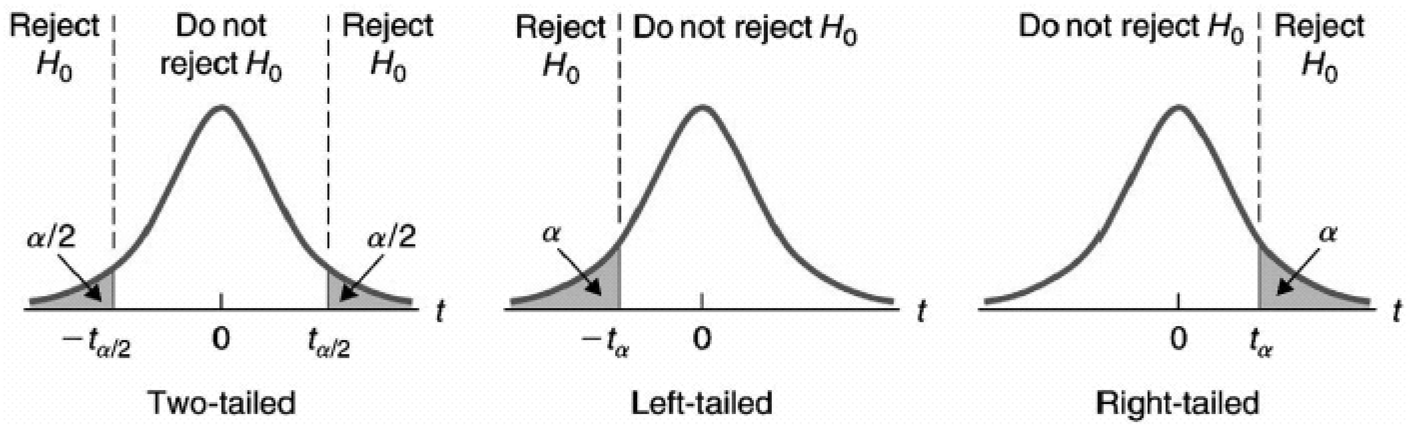
- Summary of results in separate folds

Podział	**Kl_1**	**Kl_2**
1	87,45	88,4
2	86,5	88,1
3	86,4	87,2
4	86,8	86
5	87,8	87,6
6	86,6	86,4
7	87,3	87
8	87,2	87,4
9	88	89
10	85,8	87,2
Srednia	**86,98**	**87,43**
Odchylenie	**0,65**	**0,85**

The general question: given two classifiers K1 and K2 produced by feeding a training dataset D to two algorithms A1 and A2,
which classifier will be more accurate in classifying new examples?

Paired t-test

- The null hypothesis H0: the average performance of classifiers on the data D is =
- H1: usually $\neq$
- Test statistics and the decision based on α



- Remark: assumption $\rightarrow$ the paired difference variable should be normally distributed!

An example of „paired t-test” $\alpha = 0{,}05$

Table 1. Comparison of classification accuracies [%] obtained by the single MODLEM based classifier and the bagging approach

Name of dataset	Single MODLEM	Bagging - with different T 3	5	7	10
bank	93.81 ± 0.94	95.05 ± 0.91	94.95 ± 0.84	95.22 ± 1.02	93.95* ± 0.94
buses	97.20 ± 0.94	98.05* ± 0.97	99.54 ± 1.09	97.02* ± 1.15	97.45* ± 1.13
zoo	94.64 ± 0.67	93.82* ± 0.68	93.89* ± 0.71	93.47 ± 0.73	93.68 ± 0.70
hepatitis	78.62 ± 0.93	82.00 ± 1.14	84.05 ± 1.1	81.05 ± 0.97	84.0 ± 0.49
iris	94.93 ± 0.5	95.13* ± 0.46	94.86* ± 0.54	95.06* ± 0.53	94.33* ± 0.59
automobile	85.23 ± 1.1	82. 98 ± 0.86	83.0 ±0.99	82.74 ±0.9	81.39 ± 0.84
segmentation	85.71 ± 0.71	86.19* ± 0.82	87.62 ± 0.55	87.61 ± 0.46	87.14 ± 0.9
glass	72.41 ± 1.23	68.5 ± 1.15	74.81 ± 0.94	74.25 ± 0.89	76.09 ± 0.68
bricks	90.32* ± 0.82	90.3 * ± 0.54	89.84* ± 0.65	91.21* ± 0.48	90.77* ± 0.72
vote	92.67 ± 0.38	93.33* ± 0.5	94.34 ± 0.34	95.01 ± 0.44	96.01 ± 0.29
bupa	65.77 ± 0.6	64.98* ± 0.76	76.28 ± 0.44	70.74 ± 0.96	75.69 ± 0.7
election	88.96 ± 0.54	90.3 ± 0.36	91.2 ± 0.47	91.66 ± 0.34	90.75 ± 0.55
urology	63.80 ± 0.73	64.8 ± 0.83	65.0 ± 0.43	67.40 ± 0.46	67.0 ± 0.67
german	72.16 ± 0.27	73.07* ± 0.39	76.2 ± 0.34	75.62 ± 0.34	75.75 ± 0.35
crx	84.64 ± 0.35	84.74* ± 0.38	86.24 ± 0.39	87.1 ± 0.46	89.42 ± 0.44
pima	73.57 ± 0.67	75.78* ± 0.6	74.35* ± 0.64	74.88 ± 0.44	77.87 ± 0.39

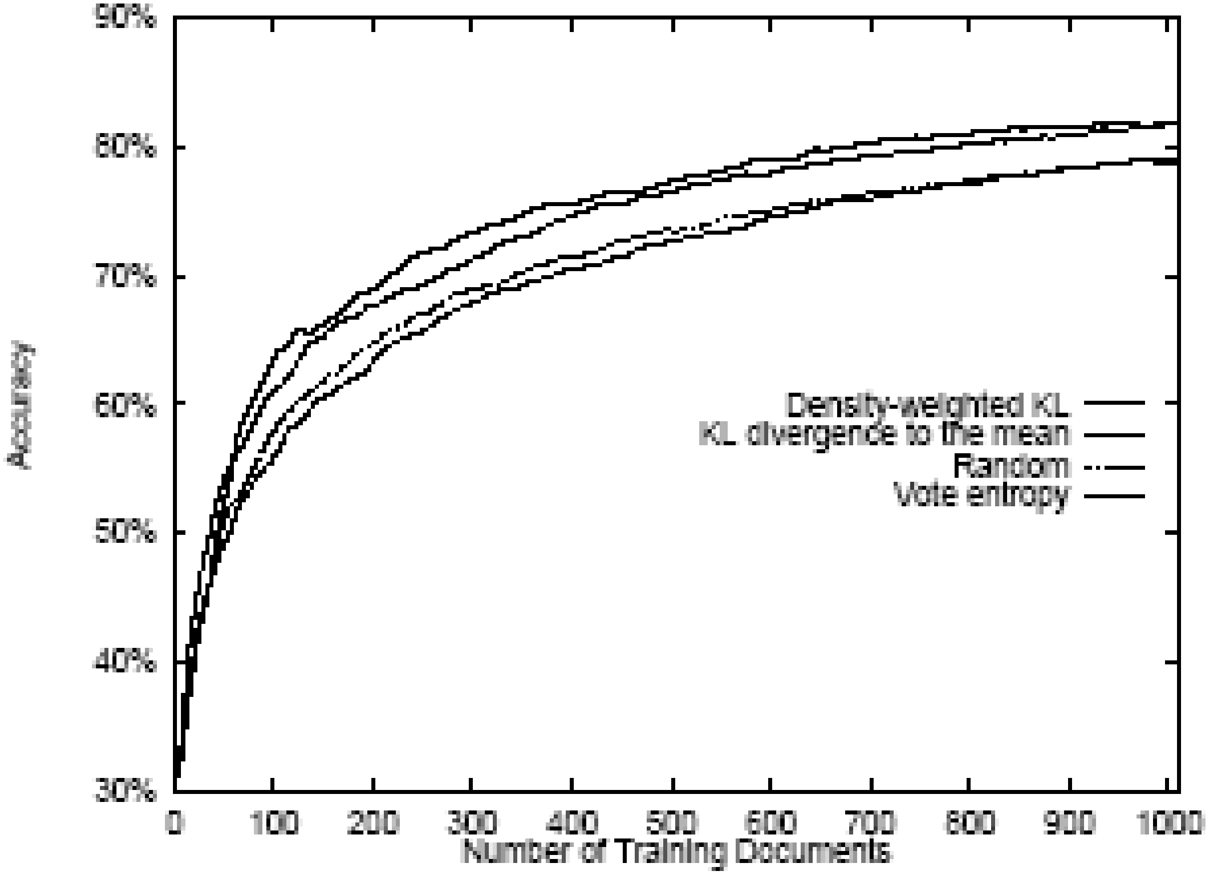
One classifier (Single MODLEM) versus other bagging schema - J.Stefanowski

Other sampling techniques for classifiers

- There are other approaches to learn classifiers:
  - Incremental learning
  - Batch learning
  - Windowing
  - Active learning
- Some of them evaluate classification abilities in stepwise way:
  - Various forms of learning curves

An example of a learning curve

- Used naïve Bayes model for text classification in a Bayesian learning setting (20 Newsgroups dataset) -



[McCallum & Nigam, 1998]

Summary

- What is the classification task?
- Discovering classifiers is a muti-step approach.
  - Train and test paradigm.
- How could you evaluate the classification knowledge:
  - Evaluation measures – predictive ability.
- Empirical approaches – use independent test examples.
  - Hold-out vs. cross validation.
  - Repeated 10 fold stratified cross validation.
- More advances issues (e.g. more about comparing many algorithms and ROC analysis will be presented during future lectures)

Any questions, remarks?

?