WEKA: Practical Machine Learning Tools and Techniques in Java

Seminar A.I. Tools

WS 2006/07

Rossen Dimov

Overview

- Basic introduction to Machine Learning
- Weka Tool
- Conclusion
- Document classification Demo

What is Machine Learning

- Definition: A computer program is said to *learn* from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.

What is Machine Learning

- **T** – playing chess
- **P** – percentage of wins
- **E** – 1000 recorded whole games

Basic definitions

	Outlook	Temperature	Humidity	Windy	Surfing
1	Sunny	Mild	Normal	True	Yes
2	Sunny	Hot	High	False	No
3	Rainy	Mild	High	False	No
4	Overcast	Cool	Normal	True	Yes

Basic definitions

Attributes

	Outlook	Temperature	Humidity	Windy	Surfing
1	Sunny	Mild	Normal	True	Yes
2	Sunny	Hot	High	False	No
3	Rainy	Mild	High	False	No
4	Overcast	Cool	Normal	True	Yes

Basic definitions

Special Attribute – Class Attribute

	Outlook	Temperature	Humidity	Windy	Surfing
1	Sunny	Mild	Normal	True	Yes
2	Sunny	Hot	High	False	No
3	Rainy	Mild	High	False	No
4	Overcast	Cool	Normal	True	Yes

Basic definitions

Instance

	Outlook	Temperature	Humidity	Windy	Surfing
1	Sunny	Mild	Normal	True	Yes
2	Sunny	Hot	High	False	No
3	Rainy	Mild	High	False	No
4	Overcast	Cool	Normal	True	Yes

Basic definitions

Dataset

	Outlook	Temperature	Humidity	Windy	Surfing
1	Sunny	Mild	Normal	True	Yes
2	Sunny	Hot	High	False	No
3	Rainy	Mild	High	False	No
4	Overcast	Cool	Normal	True	Yes

Basic definitions

T – test set: the class attribute of every instance has no value, and it should be predicted

atr1	attr2	attr3	cl_attr
a1_v1	a2_v1	a3_v1	?
a1_v2	a2_v2	a3_v2	?
a1_v3	a2_v3	a3_v3	?

E – training set: the class attribute of every instance has a value, inserted by expert or with experiment

atr1	attr2	attr3	cl_attr
a1_v1	a2_v3	a3_v2	cl_1
a1_v2	a2_v2	a3_v1	cl_2
a1_v3	a2_v5	a3_v2	cl_1

Basic definitions

- Hypothesis – consist of conjunction of constraints on the instance attributes
- < Outlook, Temperature, Humidity, Windy >
- < ? , Cold , Ø , Strong >

When to apply Machine Learning

- Dependencies and correlations can not be obvious - the instances in training and test set usually have huge number of attributes
- The algorithms need to evolute in the changing environment
- Some problems are better defined with examples - OCR

Disciplines with influence on ML

- AI – ML in general is search problem using prior knowledge
- Bayesian methods – Bayes’ theorem as the basis for calculating probabilities of hypothesis
- Statistics – characterization of errors that occur when estimating the accuracy of a hypothesis based on a limited sample of data

Disciplines with influence on ML

- Psychology – simulation of the ‘law of practice’
- Neurobiology – neurobiological studies motivate creating a simple models of biological neurons.
- Control theory – procedures for optimizing predefined objectives

Categorization based on the desired outcome of the algorithm

- Supervised learning - technique for creating a function from training data
- Unsupervised learning - method where a model is fit to observations
- Semi-supervised learning - combines both labeled and unlabeled examples to generate an appropriate function

Categorization based on the desired outcome of the algorithm

- Reinforcement learning – an agent exploring an environment in which perceives its current state and takes actions.
- Learning to learn - where the algorithm learns its own inductive bias based on previous experience.

Some ML algorithm types

- Concept learning
- Decision tree learning
- Neural networks
- Genetic algorithms
- Instance based learning
- Bayesian learning
- Clustering

WEKA

- The Weka is an endemic bird of New Zealand or ..

- W(aikato) E(nvironment) for K(nowlegde) A(nalysis)

Project Weka

- Developed by the University of Waikato in New Zealand
- http://www.cs.waikato.ac.nz/~ml/index.html

What is WEKA?

- Comprehensive suite of Java class libraries
- Implement many state-of-the-art machine learning and data mining algorithms

WEKA consists of

- Explorer
- Experimenter
- Knowledge flow
- Simple Command Line Interface
- Java interface

Explorer

- WEKA’s main graphical user interface
- Each of the major weka packages Filters, Classifiers, Clusterers, Associations, and Attribute Selection is represented along with a Visualization tool

Explorer – Data pre-processing

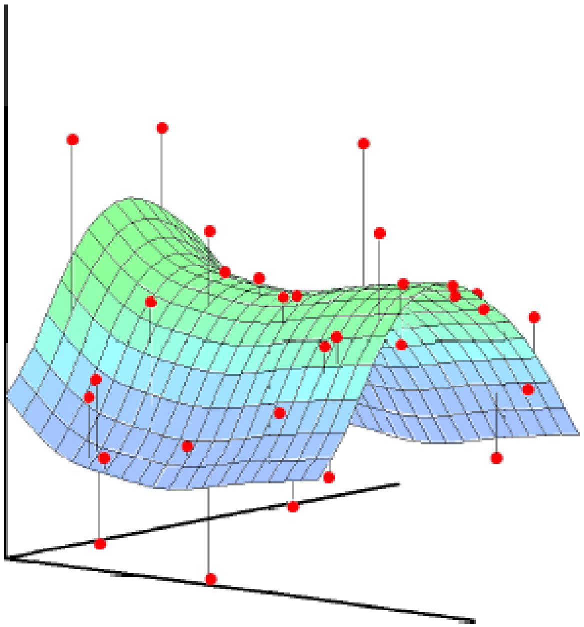
- ARFF, CSV, C4.5 or binary data
- Data loaded from URL or DB
- Preprocessing routines in WEKA are called ‘filters’ – *MergeAttributeValuesFilter, NominalToBinaryFilter*, *DiscretiseFilter*, *ReplaceMissingValuesFilter* …

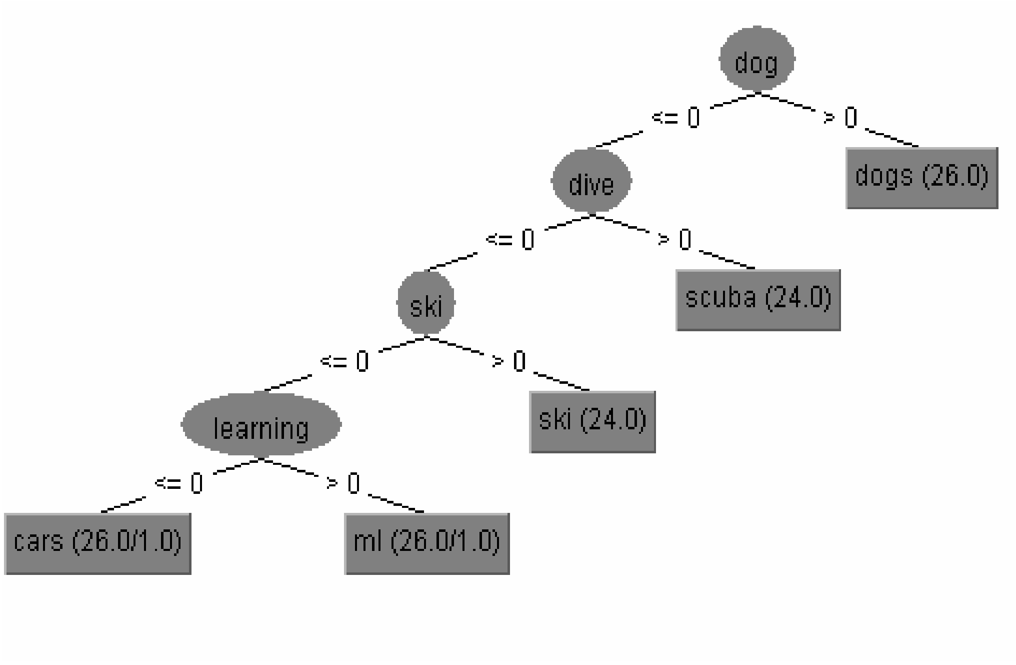
Explorer – train Classifier

- The process of creating a function or data structure, that will be used for classifying of new instances
- A set of user defined options is used to refine the result of training

Explorer – train Classifier

- How a trained classifier looks like?



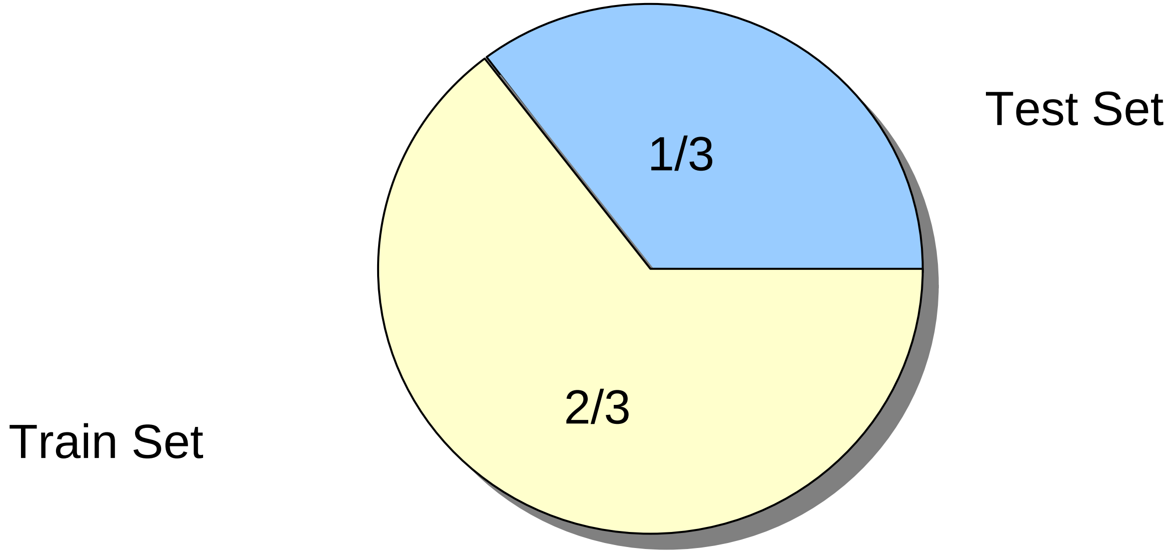


Explorer – evaluate Classifiers

- Train set
- Test set

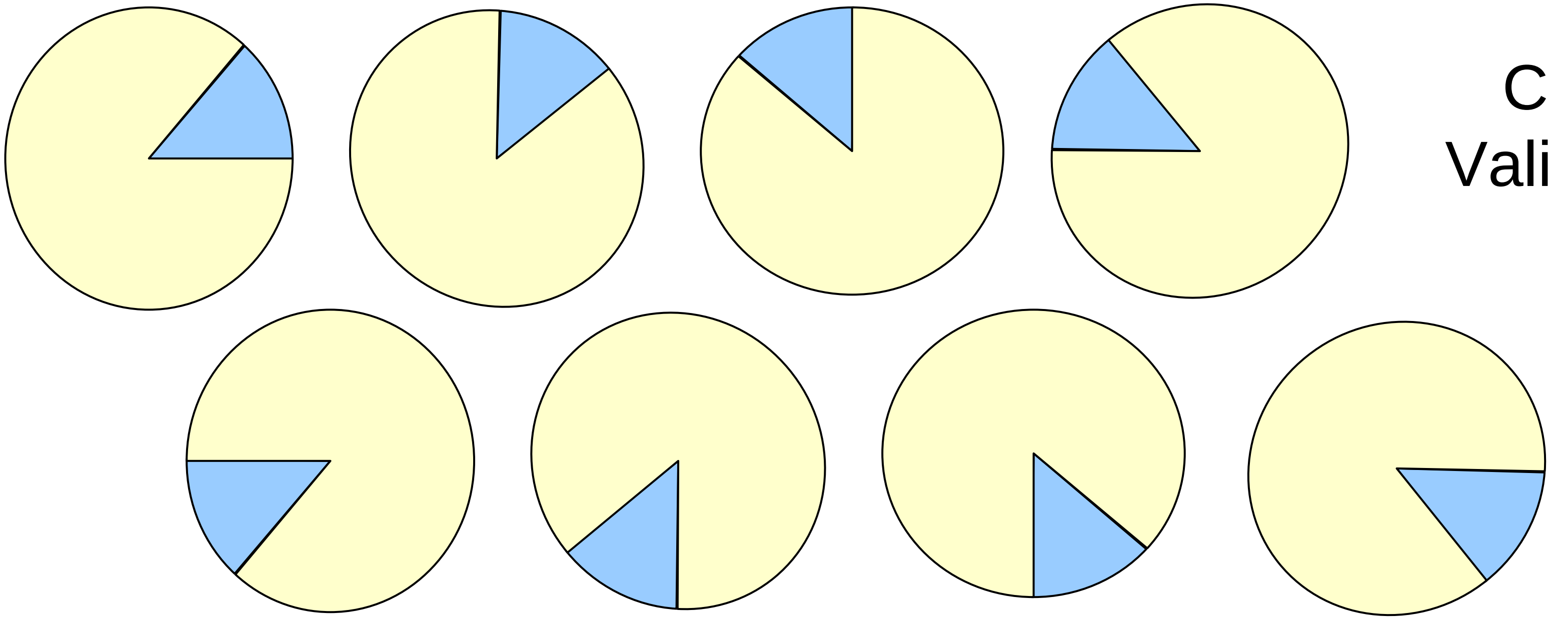
Explorer – evaluate Classifiers

- Train set
- Test set
- The amount of the data is ‘enough’



Explorer – evaluate Classifiers

- Train set
- Test set
- The amount of the data is limited



Cross Validation

Explorer – Classification results

- Confusion matrix

dogs	ski	scuba	ml	cars	
26	0	0	0	0	dogs
0	24	0	0	1	ski
0	0	24	0	1	scuba
0	0	0	25	0	ml
0	0	0	0	25	cars

- TPR matrix

dogs	ski	scuba	ml	cars	
100	0	0	0	0	dogs
0	96	0	0	4	ski
0	0	96	0	4	scuba
0	0	0	100	0	ml
0	0	0	0	100	cars

Explorer – Meta Classifiers

- Methods that enhance the performance or extend the capabilities of the basic classifiers
- The Meta Classifiers will be discussed in more details in the talk next week

Explorer – Association Rules

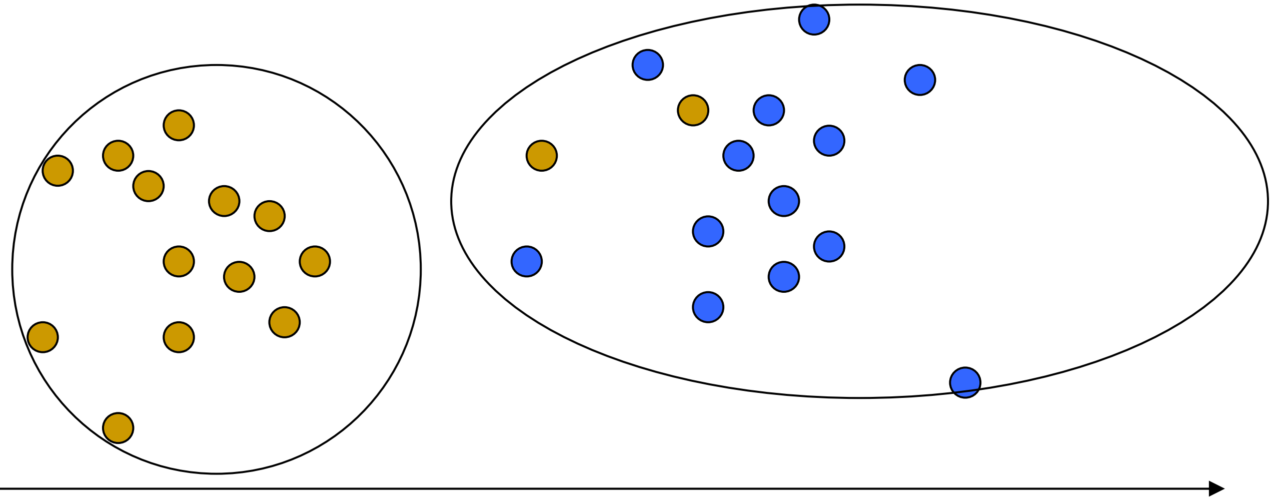
- Weka contains an implementation of the *Apriori* learner for generating association rules
- outlook=sunny humidity=high 3 → surfing=no 3

Explorer – Clustering

- Unsupervised learning

Explorer – Clustering

- Unsupervised learning
- Implies metric to calculate the 'similarity' between the instances.



Explorer - Attributes selection

- Relevant attributes for classification

Explorer - Attributes selection

- Relevant attributes for classification
- Finding which subset of attributes works best for prediction

attr1	….	attr4	…	attr13	class
a1v1	…	a4v1	…	a13v1	cl1
a1v2	…	a4v2	…	a13v2	cl2
a1v3	…	a4v3	…	a13v3	cl1

Explorer - Visualize

- Visualization of the dataset
- A matrix for every pair of attributes

Experimenter

- Comparing different learning algorithms
- … on different datasets
- … with various parameter settings
- … and analyzing the performance statistics

Knowledge flow

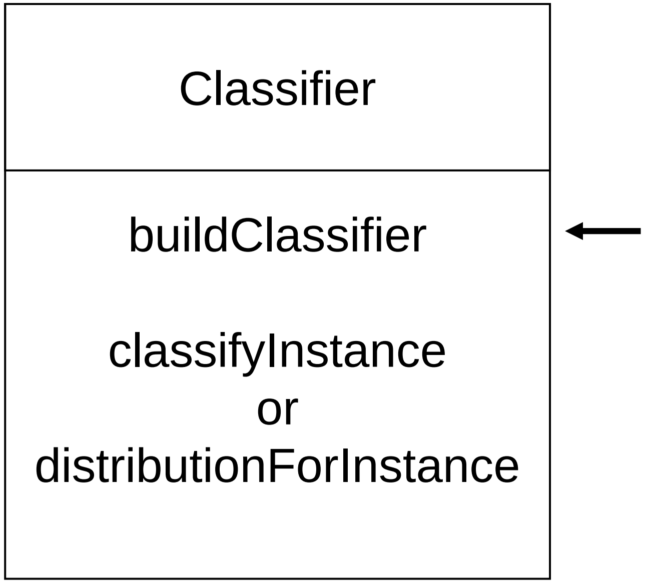
- The KnowledgeFlow provides an alternative to the Explorer as a graphical front end to Weka's core algorithms.
- The KnowledgeFlow is a work in progress so some of the functionality from the Explorer is not yet available.

Simple command line interface

- All implementations of the algorithms have a uniform command-line interface.
- java weka.classifiers.trees.J48 -t weather.arff

Java Interface – Classifier class

- public abstract class **Classifier**



Classifier

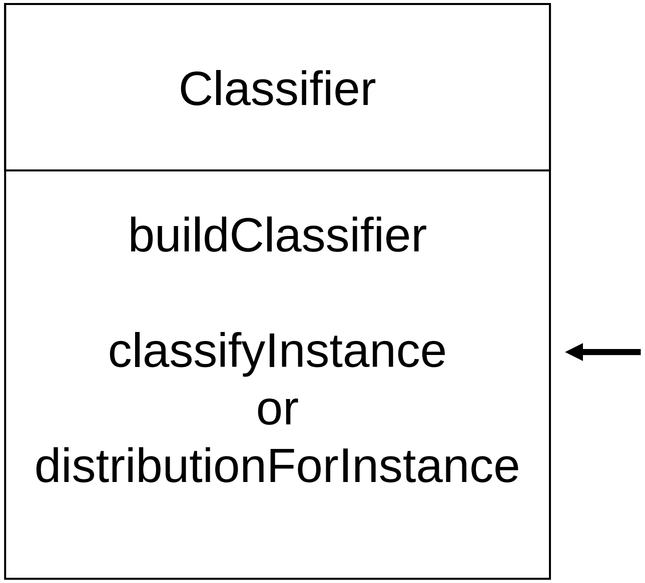
buildClassifier

classifyInstance
or
distributionForInstance

a routine which generates a classifier model from a training dataset

Java Interface – Classifier class

- public abstract class **Classifier**



Classifier

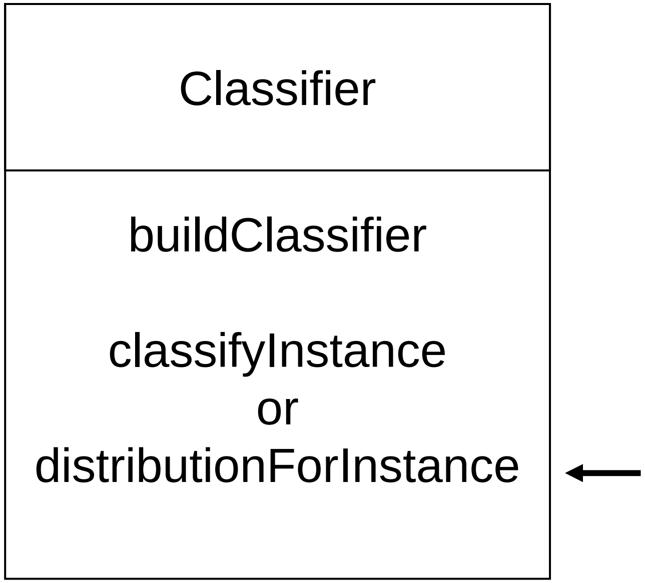
buildClassifier

classifyInstance
or
distributionForInstance

routine which evaluates the generated model on an unseen test dataset

Java Interface – Classifier class

- public abstract class **Classifier**



a routine which generates a probability distribution for all classes

Java Interface

```
Instances data = new Instances( "data.arff");  // loading data
data.setClassIndex(position); // setting class attribute

Remove remove = new Remove();                    // new instance of filter
remove.setOptions("-R");                               // set options
remove.setInputFormat(data);                // to inform filter about dataset
Instances newData = Filter.useFilter(data, remove); // apply filter

J48 tree = new J48();        // new instance of tree
tree.setOptions("-U");       // set the options
tree.buildClassifier(data);   // build classifier
```

Java Interface

```
// using 10 times 10-fold cross-validation.
Evaluation eval = new Evaluation(newData);
eval.crossValidateModel( tree, newData, 10,
newData.getRandomNumberGenerator(1));

Instances unlabeled = new Instances( “unlabeled.arff" ); // unlabeled data
unlabeled.setClassIndex( position); // set class attribute
Instances labeled = new Instances(unlabeled);          // create copy
// label instances
for (int i = 0; i < unlabeled.numInstances(); i++)
{
  clsLabel = tree.classifyInstance(unlabeled.instance(i));
  labeled.instance(i).setClassValue(clsLabel);
}
```

Conclusion

- Weka is a collection of machine learning algorithms for solving real-world data mining problems
- It is written in Java and runs on almost any platform
- The algorithms can either be applied directly to a dataset or called from your own Java code.

Conclusion

- License - GNU General Public License (GPL)
- So possible to study how the algorithms works and to modify them.

Demo

- Document classification – five different categories
  - Car maintaining
  - Machine learning
  - Dogs breeding
  - Scuba diving
  - Skiing

Demo

- Every category has 25 documents and every document has ca. 200 words
- Before pre-processing every document is represented by two attributes – class attribute and the next attribute contains the whole document

Demo

- Used filters
  - StringToWordVector
  - NumericToBinary
  - StringToWordVector with IDFTransform option
- Attribute Selection method
  - ChiSquaredAttributeEval

Demo

- Used classifiers
  - J48(C4.5)
  - Naive Bayes
  - IBk (kNN)

Demo

- Results

	J48	NB	1NN	3NN
StringToWordVector	96.80%	97.60%	35.20%	-
StringToWordVector with IDFTransform	96.80%	100%	-	75.20%
NumericToBinary	96.80%	99.20%	-	75.20%
with smaller set of attributes				
StringToWordVector	98.41%	100%	96.83%	-
StringToWordVector with IDFTransform	97.60%	100%	99.20%	-
NumericToBinary	97.60%	100%	99.20%	-

References

- Mitchell, T. Machine Learning, 1997 McGraw Hill.
- Ian H. Witten, Eibe Frank, Len Trigg, Mark Hall, Geoffrey Holmes, and Sally Jo Cunningham (1999). Weka: Practical machine learning tools and techniques with Java implementations.
- Ian H. Witten, Eibe Frank (2005). Data Mining: Practical Machine Learning Tools and Techniques (Second Edition, 2005). San Francisco: Morgan Kaufmann