ANNES ’95

Workshop on

Intelligent Data Analysis

using

the WEKA Workbench

Ian Witten
Sally Jo Cunningham
Geoff Holmes

Department of Computer Science
The University of Waikato
Hamilton

email: {ihw, sallyjo, geoff} @cs.waikato.ac.nz

Contents

Ian Witten
Machine learning and data mining
The WEKA workbench

Sally Jo Cunningham
Machine learning and statistics
Data mining tools

Geoff Holmes
Data engineering
Process model for data mining
Supporting the process model

WEKA Video
Transforming data into information

• $5 \times 10^6$ databases in the world (1989)

• growing gap between data generation and data understanding

• intelligently analyzed data is a valuable resource:
  to improve existing operations
  to “sell knowledge”
  for training
Data Mining and Machine Learning

Finding rules and correlations in data
- Preprocessing, data cleaning
  - Data engineering
  - Process model
  - Iterative analysis
  - Rule interpretation
  - Database mining vs file mining
- Statistical analysis
- Emphasis on successful applications

Learning models from examples and other information
- Rule induction
  - Empirical learning
  - Supervised and unsupervised
- Probabilistic representation (e.g., Bayesian networks)
- Emphasis on general tools for learning

Fringe:
- Fuzzy sets
- Rough sets

Fringe:
- Neural nets
- Genetic algorithms
- Reinforcement learning

Explanation-based learning
- Case-based learning
- Interactive learning
- Inductive logic programming
- Theory of learning
Fielded Applications of ML

- Increasing yield in chemical process control
- Making credit decisions
- Diagnosis of mechanical devices
- Automatic classification of celestial objects
- Reducing banding in rotogravure printing
- Improving the separation of gas from oil
- Preventing breakdowns in electrical transformers
- Basket analysis for supermarket promotions
- Telephone traffic analysis to identify new services
Introduction to machine learning

• Systems that build models
• Systems that adapt
• Example-based
• Useful for:
  • knowledge discovery
  • classification
  • prediction
Empirical Learning

- examples
- model formation algorithm
- model
- user
eg sample cases  eg decision rule

cases with redundancy and noise

- case with incorrect decisions (noise)
- cases with irrelevant attributes (redundancy)

correct, non-redundant cases

- critical, non-redundant cases

rules

- adequate rules complete
give correct decision
- minimal rules complete, correct

data

many 1000's

reduction

handful

knowledge
Empirical learning

Induce a classification scheme from examples

**Input data**

<table>
<thead>
<tr>
<th>case</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>decision</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>11</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>12</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>13</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>14</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>15</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>16</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>17</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>18</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>19</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>20</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>21</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>22</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>23</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>24</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

**Decision tree (from ID3)**

![Decision tree diagram](image)

**Production rules (from ID3)**

1. \( d=1 \) \( \rightarrow \) decide 3
2. \( a=1 \) & \( b=1 \) & \( c=1 \) & \( d=2 \) \( \rightarrow \) decide 2
3. \( a=2 \) & \( b=1 \) & \( c=1 \) & \( d=2 \) \( \rightarrow \) decide 2
4. \( a=3 \) & \( b=1 \) & \( c=1 \) & \( d=2 \) \( \rightarrow \) decide 3
5. \( b=2 \) & \( c=1 \) & \( d=2 \) \( \rightarrow \) decide 2
6. \( b=1 \) & \( c=2 \) & \( d=2 \) \( \rightarrow \) decide 1
7. \( a=1 \) & \( b=2 \) & \( c=2 \) & \( d=2 \) \( \rightarrow \) decide 3
8. \( a=2 \) & \( b=2 \) & \( c=2 \) & \( d=2 \) \( \rightarrow \) decide 3
9. \( a=3 \) & \( b=2 \) & \( c=2 \) & \( d=2 \) \( \rightarrow \) decide 3

**Production rules (from PRISM)**

1. \( d=1 \) \( \rightarrow \) decide 3
2. \( a=1 \) & \( c=1 \) & \( d=2 \) \( \rightarrow \) decide 2
3. \( a=2 \) & \( c=1 \) & \( d=2 \) \( \rightarrow \) decide 2
4. \( a=3 \) & \( b=1 \) & \( c=1 \) \( \rightarrow \) decide 3
5. \( b=2 \) & \( c=1 \) & \( d=2 \) \( \rightarrow \) decide 2
6. \( b=1 \) & \( c=2 \) & \( d=2 \) \( \rightarrow \) decide 1
7. \( a=1 \) & \( b=2 \) & \( c=2 \) & \( d=2 \) \( \rightarrow \) decide 1
8. \( a=2 \) & \( b=2 \) & \( c=2 \) & \( d=2 \) \( \rightarrow \) decide 3
9. \( a=3 \) & \( b=2 \) & \( c=2 \) & \( d=2 \) \( \rightarrow \) decide 3
Supervised learning

Classes are

• predefined
• discrete (ie. no prediction of continuous variables)

Cases

• values for a fixed set of attributes (ie. not list or data structures)
• are independent of each other (ie. not seeking relations between cases)
• are sufficiently numerous (to overcome noise)
• are sufficiently representative (to illustrate every facet of the model)
Unsupervised learning

Given a set of object descriptions, find their “natural” groupings

Clustering: the algorithm determines how many classes there are
Conceptual clustering

**CO BWEB (Fisher, 1987)**

<table>
<thead>
<tr>
<th>Name</th>
<th>BodyCare</th>
<th>HeartChamber</th>
<th>BodyTemp</th>
<th>Fertilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>‘mammal’</td>
<td>hair</td>
<td>four</td>
<td>regulated</td>
<td>internal</td>
</tr>
<tr>
<td>‘bird’</td>
<td>feathers</td>
<td>four</td>
<td>regulated</td>
<td>internal</td>
</tr>
<tr>
<td>‘reptile’</td>
<td>cornified-skin</td>
<td>imperfect-four</td>
<td>unregulated</td>
<td>internal</td>
</tr>
<tr>
<td>‘amphibian’</td>
<td>moist-skin</td>
<td>three</td>
<td>unregulated</td>
<td>external</td>
</tr>
<tr>
<td>‘fish’</td>
<td>scales</td>
<td>two</td>
<td>unregulated</td>
<td>external</td>
</tr>
</tbody>
</table>

Yields a concept hierarchy:
**WEKA** machine learning workbench

**Input**
- ARFF input data format
- Attribute editor
- File viewer

**Machine learning schemes**
- Supervised
  - C4.5
  - FOIL
  - Induct
  - IB1
  - K*
  - 1R
- Unsupervised
  - Classweb
  - AutoClass

**Output**
- Output viewers: tree and text
- WORF rule format
- Evaluation – PREVAL
Machine learning in agriculture: example problems

**White Clover**  Predict the amount of white clover from the amount of other species growth over the previous 3 years

**Valleys**  Find classes of valley depending on the form of the surface; then find rules that to describe each class

**Weight and Behaviour of Bulls**  Use live weight, testosterone level, testes size, and riding behaviour to determine whether the male was entire, immunocastrated, or castrated

**Venison Bruising**  Find which factors contribute most to bruising: origin farm, distance travelled, carrier, weight of deer, fat content, other damage

**River Quality**  Find relationships among properties such as flow rate, temperature, and chemical composition; measurements taken at many sites over a long time period.

**Apple Bruising**  Find what contributes most to bruise area: bruise depth top, bruise depth bottom, contact area, apple radius, and impact energy

**Fleece and Body Weight**  Determine the relationship between age of dam, age of sire, birth weight, birth rank, rearing rank, and breed line (whether increased body weight, increased fleece weight, or control)

**Resistance to Sporidesmin**  Determine what attributes to use in classifying the line of sheep (whether resistant, susceptible, or control)

**Sheep Wool Growth**  Find a relationship between live weight, wool growth, nutritional level and lambing number for two breeds of sheep

**Cow Culling**  Determine rules for culling cows depending on milk production and other factors

**Oestrus of Cows**  Determine when a cow is in heat from factors such as milk volume, milking order and behaviour
Characteristics of the input data

Data should be as complete as possible

Completeness: percentage of all possible attribute/value combinations that actually occur in the data

Some algorithms require absolute completeness

Must include “important” areas of the data

Difficulty with extrapolating/interpolating in constructing model for areas not covered (or not “adequately” covered) in the data

Data should not be dynamic

Some algorithms adapted for the case if concepts embodied in the data “drift”

Data should contain as little noise as possible

Can cause contradictory classifications in the data
Can cause incorrect classifications in the model
Sources of noise

Redundant attributes—should be automatically pruned out

Mis-measurement
  missing values
  incorrectly measured/perceived
  faulty instrument
  random noise might be smoothed out
  otherwise, probably will be incorporated in model

Residual variation—unmeasured factors that actually influence the classification in real life
  some factors may be unknown, or unmeasurable
  may be possible to construct a simplified model
How do you know how good your model is?

Try it out on new data!

No more data? then use N-fold cross validation (generally, N=10)

Divide the data into N blocks
   number of cases and class distribution is uniform

Run the ML algorithm N times
   in each run one block is omitted from the training data
   resulting model is tested on the cases in that omitted block

Measure the error rate for each of the N models

Average error rate over the N models is the cross-validation estimate of the error rate of a rule set built from all the data.
Statistical analysis

• tends to focus on problems where attributes have continuous values

• user formulates and tests own hypotheses

• often assumes a parametric form for the data model

Machine learning

• formulates and tests hypotheses autonomously

• looking for more logically complex relationships existing in the data
What has ML learned from statistics?

• similar techniques used in initial example set construction (visualization, selection of attributes, etc.)

• many ML algorithms use statistical tests in constructing rules/trees

• borrow techniques for correcting over-fitted models

• statistical tests used to validate ML models

• statistical tests used to evaluate ML algorithms (which work best? on what data?)
What can ML/CS contribute to statistics?

• efficient implementation techniques
  K-nearest neighbor
  \[\rightarrow\] instance-based learning, case-based reasoning

• efficient search techniques

• different focus in tools
(i) Sample rules derived by machine learning techniques:

Rule 1: If 4.8” <= petal length <= 6.7” and 1.8” <= petal width <= 2.5”
Then species = Virginica

Rule 7: If 1.7” <= petal length <= 4.9” and 0.6” <= petal width <= 1.7”
Then species = Versicolor

Rule 15: If 1” <= petal length <= 1.9”
Then species = Setosa

(ii) Output of analysis using a statistical package:

SUM OF PRODUCT MATRIX M = G’A’ [A(X’X)-1]-1 AB (Hypothesis)

<table>
<thead>
<tr>
<th></th>
<th>S-LENGTH</th>
<th>S-WIDTH</th>
<th>P-LENGTH</th>
<th>P-WIDTH</th>
</tr>
</thead>
<tbody>
<tr>
<td>S-LENGTH</td>
<td>61.332</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S-WIDTH</td>
<td>-15.583</td>
<td>14.193</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P-LENGTH</td>
<td>163.141</td>
<td>-52.047</td>
<td>417330</td>
<td></td>
</tr>
<tr>
<td>P-WIDTH</td>
<td>73.197</td>
<td>23.239</td>
<td>175.126</td>
<td>84.230</td>
</tr>
</tbody>
</table>

MULTIVARIATE RESULTS

HOTELLING-LAWLEY = 35.727
FSTAT = 584.923  DF = 8286  PROB = .000

WILKS’ LAMBDA = .033
FSTAT = 196.491  DF = 8288  PROB = .000

PILLAI TRACE = 1.219
F-STAT = 56.636  DF = 8300  PROB = .000

THETA = .708  S = 3  M = .6  N = 70.1  PROB = .000

(SUM1-1) 2P2+3P-1
RHO = 1.0-(N(J)-1N-G) 6(P+1)(G-1)
Data mining tool: Explora

Willi Klosgen (kloesgen@gmd.de)
ftp from ftp.gmd.de
in directory “gmd/explora”
Pattern:
Probabilistic rule (mean),
continuous dependent variable

Population:
Employees, USA
Mean of the variable <CURRENT SALARY> in the population: 13768

The mean is larger in the groups:
MALES 16577
EDUCATIONAL LEVEL > 15 17624
WHITE 14409

Population:
AGE OF EMPLOYEE>40, CLERICAL, USA.
Mean of the variable <CURRENT SALARY> in the population: 9422

The mean is larger in the groups:
NONWHITE 9892
Internet machine learning/data mining resources

General information WWW pages
• Knowledge discovery mine: http://info.gte.com/~kdd/
• Data Mine: http://www.cs.bham.ac.uk/~anp/TheDataMine.html
• MLnet Machine Learning archive: http://www.gmd.de/ml-archive
• Vienna ML Information Resources list: http://www.ai.univie.ac.at/oefai/ml/ml-ressources.html
• Data Engineering for Inductive Learning: http://ai.iit.nrc.ca/deil

Other WWW resources
• STATLOG (comparative studies of different machine learning, neural and statistical classification algorithms)
  ftp to ftp.ncc.up.pt, cd pub/statlog or http://www.up.pt/liacc/ML/statlog/index.html
• COSMIC’s Program Catalog
  Programs developed by NASA, including AUTOCLASS II (Automatic class discovery from data), COBWEB/3 (an algorithm for data clustering and incremental concept formation), and IND (a decision tree package).
  http://www.cosmic.uga.edu/maincat.html#45
• Siftware: a guide to public-domain, research-prototype, and commercial discovery tools.
  http://info.gte.com/~kdd/siftware.html
• UC Irvine ML database repository (largest collection of data sets used in ML research).

Mailing lists
• KDD Nuggets (knowledge discovery in databases)
  (to subscribe, e-mail to kdd-request@gte.com)
• ml-list (machine learning)
  ml-request@ics.uci.edu.
  Back issues may be FTP’d from ics.uci.edu in pub/ml-list
1. Process Model (Data Engineering)

1.1 Raw data from providers

1.2 Pre-processing - tools and techniques

1.3 Research goals

1.4 Attribute analysis

1.5 Experimental phase
WEKA Process Model
Raw data from providers

Typical sources
   Spreadsheets
   Relational databases
   Text files

Notes: RDBs are typically quite old technology (e.g. COBOL fixed length records) - especially if data has been collected over a long time frame. RDBs represent the biggest challenge!

Typical providers
   Scientists - CRIs, agricultural agencies
   Commercial - supermarkets, market trends

Notes: providers are typically at the “casual” spreadsheet user level. The methods for analysis are sufficiently complex that they involve a major investment of time to learn.
Pre-processing - tools and techniques

Integrated

Clementine System
Separate phase

**WEKA** - data is typically text and so languages that support the extraction and manipulation of text are used.

Unix scripts written in AWK and PERL

Direct access to data in INGRES databases (using SQL)

**What are you trying to do in this phase?**

  Determine the types of the attributes, eg numeric codes, id numbers (unique), ordered symbols, co-dependent attributes, implied attributes, missing values.

  Clarify anomalies - outliers in data (2-3 std deviations from the mean), useful to have visual tools such as box plots and histogram charts.

  Overall, the aim at this stage is to “clean” the data so that meaningful experiments can be run. All known dependencies have been determined, all the data lies in expected bounds, all missing items are accounted for.

  This phase involves heavy involvement with the data provider in order to verify changes, etc. The phase serves the added purpose of familiarising the researcher with the data.
Research goals

Classification

Which attribute do you want to predict?

Which attributes are factors in determining others?

What if you do if you don’t know anything about the data, or enough to know which one should be used for Classification?

Clustering

Automatic class discovery

Both these topics are avenues for research.
Attribute analysis

Our experience as led us to understand that clean raw data is unlikely to produce useful results.

**WEKA dairy herd project**

- Livestock Improvement Corporation
- insight into decisions made about removing cows from a herd
- 19000 records: 10 herds over 6 years
- 705 attributes
  - production Indexes
    - protein
    - milk-fat
    - volume
  - breeding Indexes
  - likely merit of progeny
Decision tree with original attributes

1. **Transfer out date**
   - <= 900420
   - > 900420

2. **Transfer out date**
   - <= 880217
   - > 880217

3. **Unknown**
4. **Animal Date of Birth**
   - <= 860811
   - > 860811

5. **Transfer out date**

6. **Cause of fate**
   - <= 890613
   - > 890613

7. **Mating date**
   - <= 890613
   - > 890613

8. **Animal Key**
   - <= 2811510
   - > 2811510

Attributes:
- IA
- BL
- CT
- GS
- IN
- LP
- MF
- MT
- OA
- OC
- UD

Sold
Died
Unknown
Sold
Died
Sold
Sold
Sold
Sold
Sold
Sold
**Weka derived attributes**

- 40 attributes including
  - Weka Age
  - Weka X PI
  - Weka X BI
  - Weka Prev X PI
  - Weka Prev X BI
  - Weka X PI Change
  - Weka X BI Change
  - Weka AvgDiff X PI
  - Weka AvgDiff X BI

where X is fat, protein, milk volume or payment

- New Class: Weka Status Code
  - Retained, Culled, Random
Decision tree with derived attributes

```
<table>
<thead>
<tr>
<th>Age</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;= 2</td>
</tr>
<tr>
<td>Retained</td>
</tr>
<tr>
<td>&gt; 2</td>
</tr>
<tr>
<td>Payment BI relative to herd</td>
</tr>
<tr>
<td>&lt;= -10.8</td>
</tr>
<tr>
<td>Milk Volume PI relative to herd</td>
</tr>
<tr>
<td>&lt;= -33.93</td>
</tr>
<tr>
<td>Culled</td>
</tr>
<tr>
<td>&gt; -33.93</td>
</tr>
<tr>
<td>Retained</td>
</tr>
</tbody>
</table>
```


To aid this process we have developed the $W_{\text{EKA}}$ attribute editor.

Attribute filtering (delete, undelete) - PROJECTION

Tuple selection - SELECTION

Conditional statements for class formation

Substring matching (extracting years from date records)

Concatenation (for merging attributes)
Experimental phase

Once the most relevant attributes and their aggregates have been decided we are ready to use the neural networks, machine learning, statistical analysis, etc.

Given the need to trial different combinations of attributes, and to renew research goals it is important to provide an environment for large scale experiments to be run.

**WEKA experiment editor:**

- Select different data sets
- Select a number of learning techniques
- Run cross-validation studies
- Collate and present results