5. Automatic Knowledge Acquisition

The Knowledge Acquisition Bottleneck

• Initially, research in the field of KBS focused on issues of implementation:
  – Separating Knowledge and Control
  – Reasoning with Rules (backward, forward)
  – Dealing with uncertainty

• Later, those developing KBS systems found that most of the effort was put into acquiring knowledge from experts.

• Experts often could not put their knowledge into words.

• Knowledge acquisition was thus called “the bottleneck of expert system development” (Feigenbaum 1979)

• New focus: how to acquire knowledge without needing human experts to verbalise their knowledge.
5. Automatic Knowledge Acquisition

Automating Knowledge Acquisition

• One approach to automatic knowledge acquisition:
  1. Rather than using the experts themselves, use the files detailing the cases they have solved (data of each case, and the decision).
  2. Replace the Knowledge Engineer with a program that analyzes the solved cases and produces a set of rules which produce the same decisions given the data.

• The idea is that these rules can be used not only to repeat observed decisions, but also to produce decisions for new cases.
• This is a process of **induction**: reasoning from the particular to the general, from cases to rules.

Sometimes called:

• **Rule induction**: rules are derived from data.

A type of:

• **Machine learning**: machine learns how to classify new data given past data classified by an expert
• **Data Mining** when the cases form a large database, and the process is used to extract economic or social knowledge.
• **Scientific Discovery**: when used with laboratory data.

Different from:

• **Case-Based Reasoning** (CBR), because CBR only makes inferences from its cases when a new case is presented.
5. Automatic Knowledge Acquisition

A Case:

Terminology and description of the problem: Labor Negotiations

- Each line describes a finished negotiation (a ‘case’, ‘example’ or ‘instance’).
- The last item on each line is the classification of the case (provided by the expert): good or bad negotiation.
- The other data on each line are attributes of the case.
  - contract duration, salary increment for 1st year, 2nd year, and 3rd year, adjustment to IPC, work hours, retirement, ...

- Independent variables: each column (except the last one), representing an attribute defining the problem
- Dependent variable: the final column, representing the goal or class that we are trying to learn
5. Automatic Knowledge Acquisition

Decision Trees for Modeling Case Data

- Given a set of case data:
- Goal is to produce a decision tree to classify examples

<table>
<thead>
<tr>
<th>Training examples</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Age</td>
<td>Marital Stat</td>
<td>Class</td>
</tr>
<tr>
<td>18</td>
<td>S</td>
<td>-</td>
</tr>
<tr>
<td>22</td>
<td>M</td>
<td>-</td>
</tr>
<tr>
<td>25</td>
<td>S</td>
<td>+</td>
</tr>
<tr>
<td>33</td>
<td>M</td>
<td>-</td>
</tr>
<tr>
<td>45</td>
<td>S</td>
<td>+</td>
</tr>
</tbody>
</table>

Producing Decision Trees from Case Data

- Each case can be mapped onto n-dimensional space (where n is the number of dependent variables)
- We can then define regions in this N-dimensional space (the dashed lines), grouping similar classifications together
5. Automatic Knowledge Acquisition

Producing Decision Trees from Case Data

- Each line in this region-space can then be placed as a question node in a decision tree.
- If the cases contained in the region have the same class, produce a terminal node labelled with the class.
- Otherwise, introduce another question node to further divide the space.

Producing Rules from Decision Trees

- A decision tree can then be represented as a set of rules.

\[
\text{IF} \quad (\neg(Age > 75) \land (Age > 23) \land (Married = Yes) \land (Age > 45)) \\
\text{THEN} \quad (Class = +)
\]
5. Automatic Knowledge Acquisition

Producing Rules from Decision Trees

- We produce one rule for each leaf node of the tree.
  - If ((age>75)) then (Class= -))
  - If (~(age>23)) then (Class= -))
  - If ((age>23) (married=no)(age<52)) then (Class= +))
  
  ...
5. Automatic Knowledge Acquisition

Region Lines need not be flat
- Take the previous example, but change “married” for “no. of children”
- The best region lines may have a slope indicating an interdependency between the two attributes.
- But hard to put such regions into rules.

5. Automatic Knowledge Acquisition

Other Models: The Restaurant Problem
- Will people wait for a table to be available?
- Assume that an expert provided the following decision tree
- (But ... is Sociology an exact science? Or is Medicine? ...)

![Decision Tree Diagram](image)
5. Automatic Knowledge Acquisition

- Data from experts, their observations/diagnostics, guide us
- Perfection is unreachable (even for experts—see $X_4$ and previous tree)
- Goal: equal (or better) successful prediction rate on unseen instances, compared with the human expert

<table>
<thead>
<tr>
<th>Example</th>
<th>Alt</th>
<th>Bar</th>
<th>Fri</th>
<th>Han</th>
<th>Put</th>
<th>Price</th>
<th>Rain</th>
<th>Res</th>
<th>Type</th>
<th>Est</th>
<th>WillWait</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X_1$</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Some</td>
<td>$$$$</td>
<td>No</td>
<td>Yes</td>
<td>French</td>
<td>0–10</td>
<td>Yes</td>
</tr>
<tr>
<td>$X_2$</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Full</td>
<td>$</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>30–60</td>
<td>No</td>
</tr>
<tr>
<td>$X_3$</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Some</td>
<td>$</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>0–10</td>
<td>Yes</td>
</tr>
<tr>
<td>$X_4$</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Full</td>
<td>$</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>10–30</td>
<td>Yes</td>
</tr>
<tr>
<td>$X_5$</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Full</td>
<td>$$$$</td>
<td>No</td>
<td>Yes</td>
<td>French</td>
<td>&gt;60</td>
<td>No</td>
</tr>
<tr>
<td>$X_6$</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Some</td>
<td>$</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>0–10</td>
<td>Yes</td>
</tr>
<tr>
<td>$X_7$</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>None</td>
<td>$</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>0–10</td>
<td>No</td>
</tr>
<tr>
<td>$X_8$</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Some</td>
<td>$</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>0–10</td>
<td>Yes</td>
</tr>
<tr>
<td>$X_9$</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Full</td>
<td>$</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>&gt;60</td>
<td>No</td>
</tr>
<tr>
<td>$X_{10}$</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Full</td>
<td>$$$$</td>
<td>No</td>
<td>Yes</td>
<td>Italian</td>
<td>10–30</td>
<td>No</td>
</tr>
<tr>
<td>$X_{11}$</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>None</td>
<td>$</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>0–10</td>
<td>No</td>
</tr>
<tr>
<td>$X_{12}$</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Full</td>
<td>$</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>30–60</td>
<td>Yes</td>
</tr>
</tbody>
</table>