OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG

INF FAKULTÄT FÜR INFORMATIK

Fuzzy Systems

Heuristic Fuzzy Rule Learning Approaches

Prof. Dr. Rudolf Kruse **Christian Moewes**

{kruse,cmoewes}@iws.cs.uni-magdeburg.de

Otto-von-Guericke University of Magdeburg

Faculty of Computer Science

Department of Knowledge Processing and Language Engineering

Learning Fuzzy Rules Differently

There are many different methods to learn fuzzy rules from data:

Cluster-oriented approaches find clusters in data where each cluster corresponds to one rule (already discussed).

Hyperbox-oriented approaches find clusters in form of hyperboxes.

Structure-oriented approaches use predefined fuzzy sets to structure the data space and pick rules from grid cells.

Neuro-fuzzy systems (NFS) combine artificial neural networks with fuzzy rule.

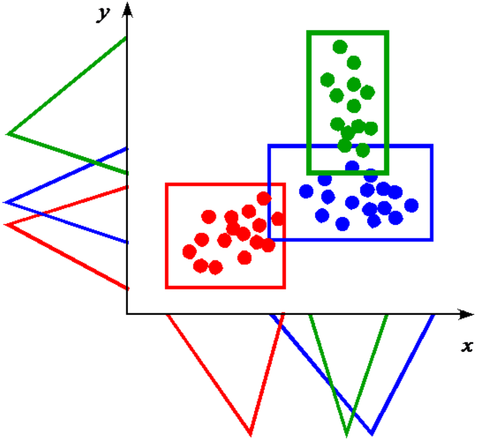
The last three topics will be discussed in the following.

Outline

1. **Hyperbox-Oriented Rule Learning**

2. Structure-Oriented Rule Learning

Hyperbox-Oriented Rule Learning



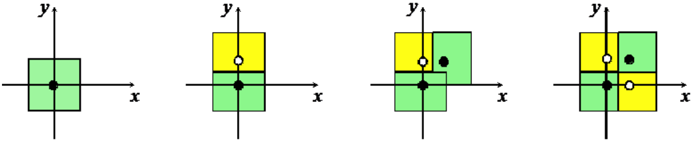
Search for hyperboxes in the data space.

Create fuzzy rules by projecting hyperboxes.

Fuzzy rules and fuzzy sets are created at the same time.

These algorithms are usually very fast.

Example: Hyperboxes in XOR Data



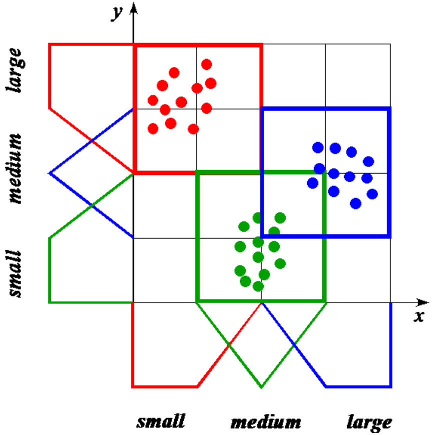
Advantage over fuzzy cluster analysis:

- There is no loss of information when hyperboxes are represented as fuzzy rules.
- Not all variables need to be used, don't care variables can be discovered.

Disadvantage: Each fuzzy rules uses individual fuzzy sets, *i.e.* the rule base is complex.

Outline

Structure-Oriented Rule Learning



We must provide the initial fuzzy sets for all variables.

This partitions the data space by a fuzzy grid.

Then we detect all grid cells that contain data [Wang and Mendel, 1992].

Finally we compute the best consequents and select the best rules, *e.g.* using NFS [Nauck and Kruse, 1997] (to be discussed later).

Structure-Oriented Rule Learning

Simple: The rule base is available after 2 cycles through the training data.

1. Discover all antecendents.
2. Determine the best consequents.

Missing values can be handled.

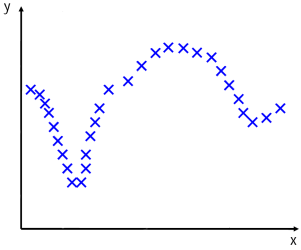
Numeric and symbolic attributes can be processed at the same time (mixed fuzzy rules).

Advantage: all rules share the same fuzzy sets.

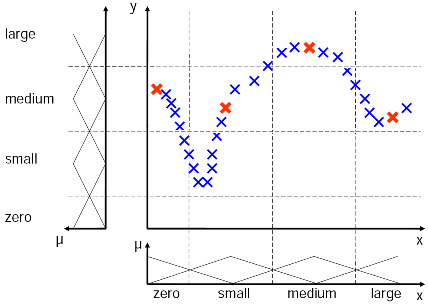
Disadvantage: fuzzy sets must be given in advance.

Example: Wang & Mendel Algorithm

given data points



step 1: granulate data space

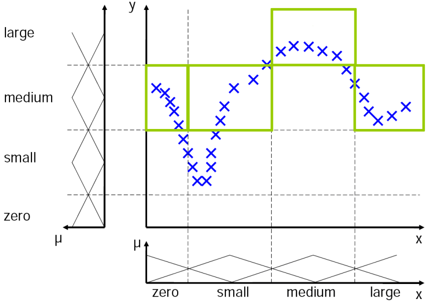


Example data set with one input and one output.

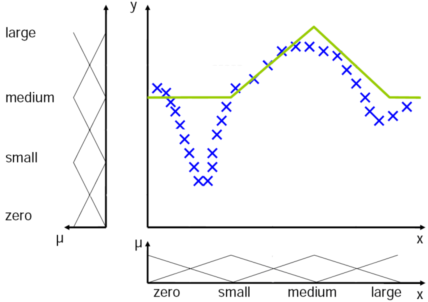
Note that the closest points to the corresponding rules are red.

Example: Wang & Mendel Algorithm (cont.)

step 2: generate rules



resulting crisp approximation



Fuzzy rules are shown by their $\alpha = 0.5$-cuts.

The learned model misses extrema far away from the rule centers.

Example: Wang & Mendel Algorithm (cont.)

Generated rule base:

$$
\begin{array}{llll}
R_1: & \textbf{if } x \textbf{ is } \mathrm{zero}_x & \textbf{then } y \textbf{ is } \mathrm{medium}_y \\
R_2: & \textbf{if } x \textbf{ is } \mathrm{small}_x & \textbf{then } y \textbf{ is } \mathrm{medium}_y \\
R_3: & \textbf{if } x \textbf{ is } \mathrm{medium}_x & \textbf{then } y \textbf{ is } \mathrm{large}_y \\
R_4: & \textbf{if } x \textbf{ is } \mathrm{large}_x & \textbf{then } y \textbf{ is } \mathrm{medium}_y
\end{array}
$$

Intuitively, rule R_2 should probably be used to describe the minimum instead:

$$R_2': \quad \textbf{if } x \textbf{ is } \mathrm{small}_x \quad \textbf{then } y \textbf{ is } \mathrm{small}_y$$

Higgins & Goodman Algorithm

[Higgins and Goodman, 1993]

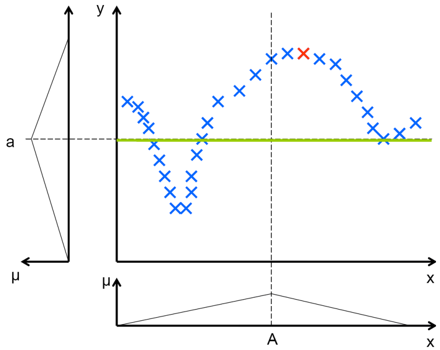
This algorithm is an extension of [Wang and Mendel, 1992].

1. Only one membership function is used for each X_j and Y. So, one large rule coveres the entire feature space initially.
2. Any new membership function is placed at the points of maximum error.

Both steps are repeated until

- a maximum number of divisions is reached or
- the approximation error remains below a certain threshold.

1. Initialization



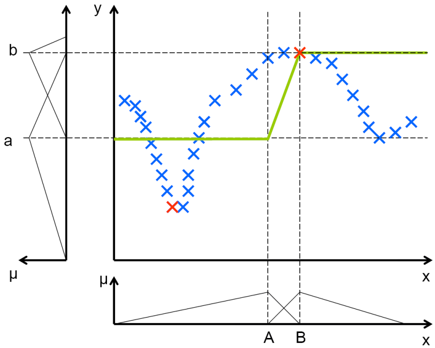
Create a membership function for each input covering the entire domain range.

Create a membership function for the output at the corner points of the input.

At the corner point, each input is maximal or minimal of its domain range.

For each corner point, the closest example from the data is used to add a membership function at its output value.

2. Adding new Membership Functions



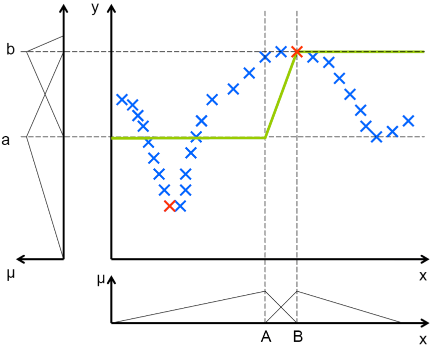
Find the point within the data with maximum error.

The defuzzification equals [Wang and Mendel, 1992]

For each X_j, add a new membership function at the corresponding value of "maximal error point".

So, this point is perfectly described by the model.

3. Create new Cell-based Rule Set



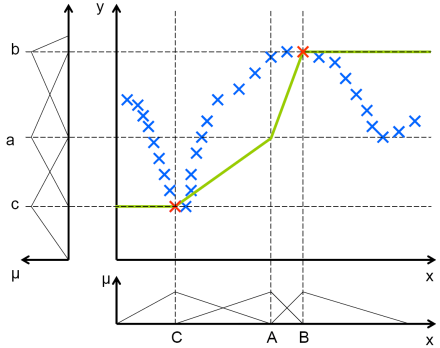
New rules: Associate the output membership functions with the newly created cells.

So, take the closest point to all membership functions of the input (equals to [Wang and Mendel, 1992])

The associated output membership function is the closest one to the output value of the “closest point”.

If the output value of the “closest point” is far away, then a new output function is created.

4. Termination Detection



If the error is below a certain threshold (or if a certain number of iterations has been performed), then the algorithm stops.

Otherwise it continue at step 2.

Summary

Heuristic fuzzy rule learning methods are usually very fast.

This is due to their greedy strategies to select rules.

For some applications, however, these strategies are too simple in terms of accuracy.

In such situations more sophisticated rule learning methods should be used, *e.g.* neuro-fuzzy systems.

References

Higgins, C. M. and Goodman, R. M. (1993).
Learning fuzzy rule-based neural networks for control.
In *Advances in Neural Information Processing Systems 5, NIPS Conference*, pages 350–357, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Nauck, D. and Kruse, R. (1997).
A neuro-fuzzy method to learn fuzzy classification rules from data.
Fuzzy Sets and Systems, 89(3):277–288.

Wang, L. and Mendel, J. M. (1992).
Generating fuzzy rules by learning from examples.
IEEE Transactions on Systems, Man, and Cybernetics, 22(6):1414–1427.