A Complete Fuzzy Decision Tree Technique
Cristina Olaru, Luis Wehenkel

Jiří Iša

December 9, 2005

Table of contents

Why do we want to extend standard decision trees?

Decision trees

- Easy to build
- Easy to understand
- Problem-independent
- Highly unstable - high variance

The difference between the fuzzy and the crisp set

- Crisp set
- Fuzzy set - membership function $\mu_S(o) : V \rightarrow [0, 1]$

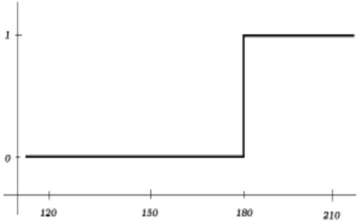


Figure: Crisp set membership function

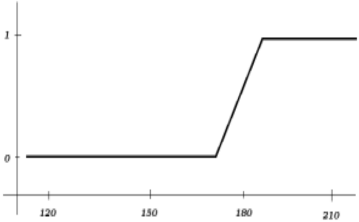
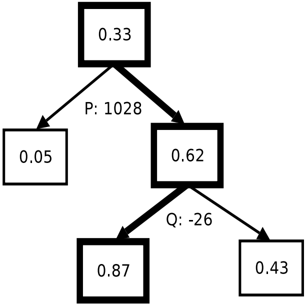


Figure: Fuzzy set membership function

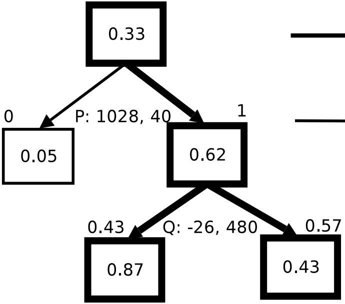
Example of the regression tree

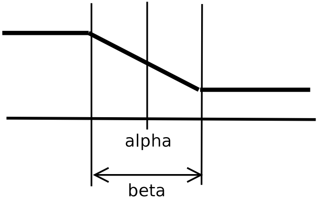
- Power plant: Variables P, Q
- Critical clearing time $\Rightarrow$ secure $\times$ insecure (threshold 155ms)
- Example: $P = 1100MW$, $Q = -40MVar$



Example of the soft decision tree

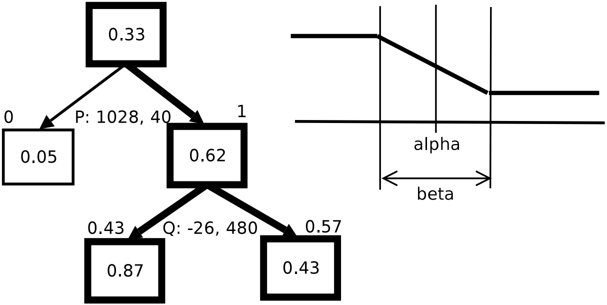
- Discriminator function: α, β
- Example: $P = 1100MW$, $Q = -40MVar$





Example of the soft decision tree

- Discriminator function: α, β
- Example: $P = 1100MW$, $Q = -40MVar$



$$\mu_{insecure}(example) = 1 * 0.43 * 0.87 + 1 * (1 - 0.43) * 0.43 \approx 0.62$$

- Membership of objects to the class (or concepts): $\mu_C(o)$, $\mu'_C(o)$
- Numerical and normalized attributes
- Fuzzy subsets $S_L, S_R \subseteq S$ and the corresponding $\mu'_S, \mu'_{S_L}, \mu'_{S_R}$
- Discriminator function $\nu(a(o), \alpha, \beta) \rightarrow [0, 1]$
- Values attached in leaves L_j

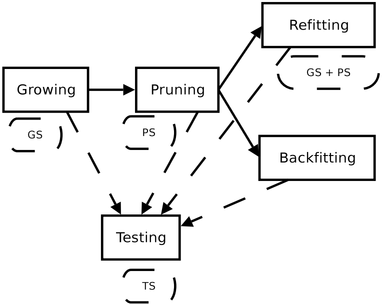
$$\mu_{S_L}(o) = \mu_S(o)\nu(a(o), \alpha, \beta)$$

$$\mu_{S_R}(o) = \mu_S(o)(1 - \nu(a(o), \alpha, \beta))$$

$$\mu'_C(o) = \frac{\sum_{j \in leaves} \mu'_{S_{L_j}}(o) L_j}{\sum_{j \in leaves} \mu'_{S_{L_j}}(o)}$$

Building a soft decision tree - Overview

- Test set (TS)
- Growing set (GS)
- Pruning set (PS)



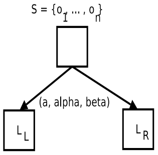
Soft decision tree growing

- Input: Fuzzy set S
- Output: Attribute, α, β, L_L, L_R
- Error function:

$$E_S = \sum_{o \in S} \mu_S(o)(\mu_c(o) - \mu'_C(o))^2$$

where

$$\mu'_C(o) = \nu(a(o), \alpha, \beta)L_L + (1 - \nu(a(o), \alpha, \beta))L_R$$



Soft decision tree pruning

- Test nodes sorted by the increasing order of their relevance $E_S = \sum_{o \in S} \mu_S(o)(\mu_c(o) - L_S)^2$
- Subtrees sequence generation
- The best subtree selection

Soft decision tree tuning

- Refitting - only terminal nodes parameters (inverse matrix, Gaussian elimination)
- Backfitting - all parameters (Levenberg-Marquardt)

Model size comparison with standard methods

Average number of nodes

GS size	C4.5	ULG	CART	SDT
250	12.4	7.7	22.2	52.1
500	20.1	12.2	45.9	52.5
1000	33.8	24.1	93.0	54.9
1500	46.0	37.5	164.0	60.2
2000	56.6	40.4	223.8	59.2

Error rate comparison with standard methods

Average error rate (%)

GS size	C4.5	ULG	CART	SDT(R)	SDT(B)
250	11.2	11.9	11.0	4.9	4.7
500	10.1	10.5	9.2	4.6	4.2
1000	8.3	8.1	7.6	4.2	4.0
1500	7.4	7.9	7.1	4.3	3.9
2000	7.0	7.0	6.5	4.4	4.1

Global variance comparison with standard methods

Global variance

GS size	Var_{CART}	$Var_{SDT(R)}$	$Var_{SDT(B)}$
250	0.023058	0.005354	0.004777
500	0.017687	0.002559	0.002264
1000	0.013400	0.001253	0.001000
1500	0.011501	0.000971	0.000835
2000	0.009871	0.000825	0.000540

- Different discriminator function (sigmoidal, Gaussian, triangular)
- Qualitative parameters instead of only qualitative

- Fuzzy classes and numerical attributes
- Regression and classification problems
- Usable for high dimensional real world problems
- Automatically sets fuzziness
- Global optimization procedure
- Fast, yet still accurate refitting
- Better results of *refitting* compared to *backfitting* on "small" data sets
- Much slower than crisp tree construction