A Soft Decision Tree

Nguyen Hung Son

son@mimuw.edu.pl

June 4, 2002

Abstract

We present the novel "soft discretization" methods using "soft cuts" instead of traditional "crisp" (or sharp) cuts. This new concept allows to generate more compact and stable decision trees with high classification accuracy. We also present an efficient method for soft cut generation from large data bases.

Talk layout

1. Motivations
2. Basic notions
3. Soft cuts and soft DT
4. Searching for soft cuts
5. Conclusions

1. Motivations

- The most important **advantage** of decision tree methods are:
  - compactness and clearness of presented knowledge
  - high accuracy of classification
- The **disadvantage** of standard decision tree methods:
  - inefficiency for very large data tables.
  - instability, i.e., small deviation of data can considerably change a model.
- **Our proposition**: use "soft cuts" instead of "crisp cuts" in internal nodes. This concept allows to
  - generate more compact and stable decision trees.
  - assure high classification quality.
  - speed up induction algorithms in case of large data stored in databases.

2. Basic notions

Decision table consists of

- a set of objects U.
- a set of attributes (columns)

$$A = \{a : U \to V_a\}$$

- a decision attribute $dec \notin A$.

Assume that $V_{dec} = \{1, \ldots, d\}$,

$$DEC_k = \{x \in U : dec(x) = k\}$$

will be called the k^{th} *decision class*

Rosetta - Rosetta1

File Edit View Window Help

No name

	SepalLength	SepalWidth	PetalLength	PetalWidth	Class
1	5.0	2.0	3.5	1.0	Iris-versicol
2	6.0	2.2	5.0	1.5	Iris-virginica
3	6.0	2.2	4.0	1.0	Iris-versicol
4	6.2	2.2	4.5	1.5	Iris-versicol
5	4.5	2.3	1.3	0.3	Iris-setosa
6	5.0	2.3	3.3	1.0	Iris-versicol
7	5.5	2.3	4.0	1.3	Iris-versicol
8	6.3	2.3	4.4	1.3	Iris-versicol
9	4.9	2.4	3.3	1.0	Iris-versicol
10	5.5	2.4	3.8	1.1	Iris-versicol
11	5.5	2.4	3.7	1.0	Iris-versicol
12	5.7	2.5	5.0	2.0	Iris-virginica
13	5.5	2.5	4.0	1.3	Iris-versicol
14	5.1	2.5	3.0	1.1	Iris-versicol
15	4.9	2.5	4.5	1.7	Iris-virginica
16	5.6	2.5	3.9	1.1	Iris-versicol
17	6.3	2.5	4.9	1.5	Iris-versicol
18	6.3	2.5	5.0	1.9	Iris-virginica
19	6.7	2.5	5.8	1.8	Iris-virginica
20	5.7	2.6	3.5	1.0	Iris-versicol
21	5.5	2.6	4.4	1.2	Iris-versicol
22	6.1	2.6	5.6	1.4	Iris-virginica
23	5.8	2.6	4.0	1.2	Iris-versicol
24	7.7	2.6	6.9	2.3	Iris-virginica
25	5.8	2.7	5.1	1.9	Iris-virginica

- Any pair (a, c), where a is an attribute and c is a real value, is called *a cut*.
- We say that *"the cut (a, c) discerns a pair of objects x, y"* if either $a(x) < c \leq a(y)$ or $a(y) < c \leq a(x)$.

Home Page
Title Page
Page 5 of 16
Go Back
Full Screen
Close
Quit

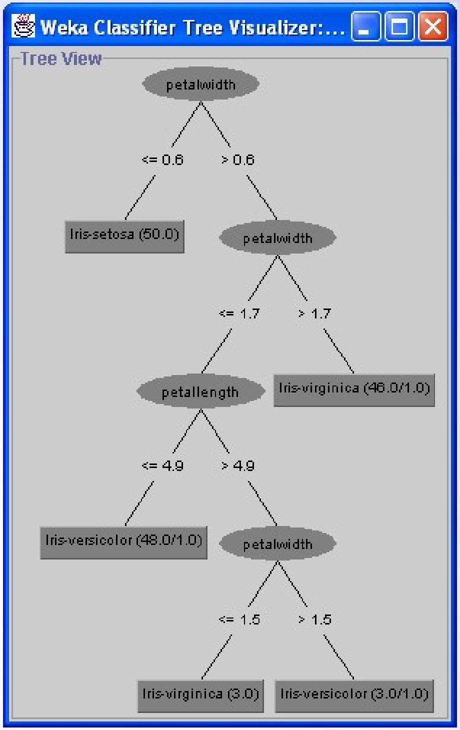
2.1. Standard decision tree

Decision tree (DT) consists of

- "test functions" in internal nodes
- "decision class" in leaves.

Decision tree tasks:

- using DT to classify new objects;
- construction of DT from data;
- choosing parameters for DT: "test function" types, "test function" evaluation, pruning ...



Optimal decision tree?

- DT is *consistent* with the decision table $\mathbb{A}$ if it classifies properly all objects from $\mathbb{A}$.
- DT is *optimal* for $\mathbb{A}$ if it has a smallest height among decision trees consistent with $\mathbb{A}$.

2.2. Decision tree construction

- The cut (a, c) is *optimal* if it labels one of internal nodes of optimal decision trees.
- The typical algorithm for DT induction:
  1. For a given set of objects U, select a cut (a, c_{Best}) of high quality among all possible cuts and all attributes;
  2. Induce a partition U_1, U_2 of U by (a, c_{Best}) ;
  3. Recursively apply Step 1 to both sets U_1, U_2 of objects until some stopping condition is satisfied.
- decision tree induction problem:

  "*For a given set of candidate cuts $\{c_1, ..., c_N\}$ on an attribute a, find a cut c_i belonging to the set of optimal cuts with highest probability*".
- Usually, we use some *measure* $F : \{c_1, ..., c_N\} \rightarrow \mathbb{R}$ to estimate the quality of cuts.
- *straightforward algorithm*:

$$c_{Best} = \arg\max_{c_i} F(c_i)$$

Motivations
Basic notions
Soft cuts and soft DT
Searching for soft cuts
Conclusions
Home Page
Title Page
Go Back
Full Screen
Close
Quit

Entropy measure

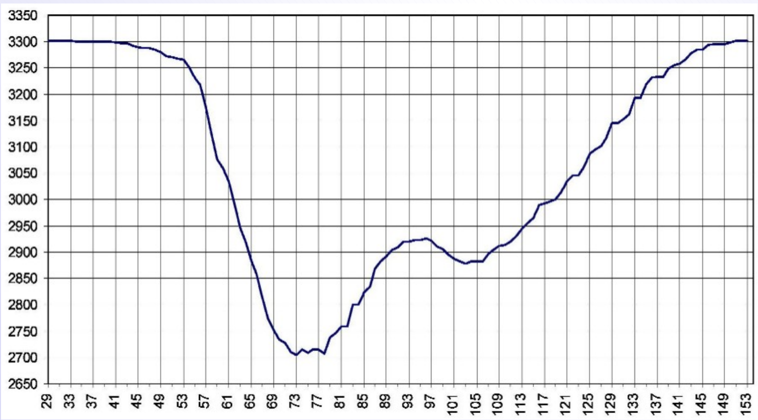
- The class information entropy of object set X with class distribution $\langle N_1, ..., N_d \rangle$, where $N_1 + ... + N_d = N$:

$$Ent(X) = -\sum_{j=1}^{d} \frac{N_j}{N} \log \frac{N_j}{N}$$

- the entropy of the partition induced by a cut (a, c):

$$E\left(a, c; U\right) = \frac{|U_L|}{n} Ent\left(U_L\right) + \frac{|U_R|}{n} Ent\left(U_R\right)$$

where $\{U_L, U_R\}$ is a partition of U defined by c.



Discernibility measure

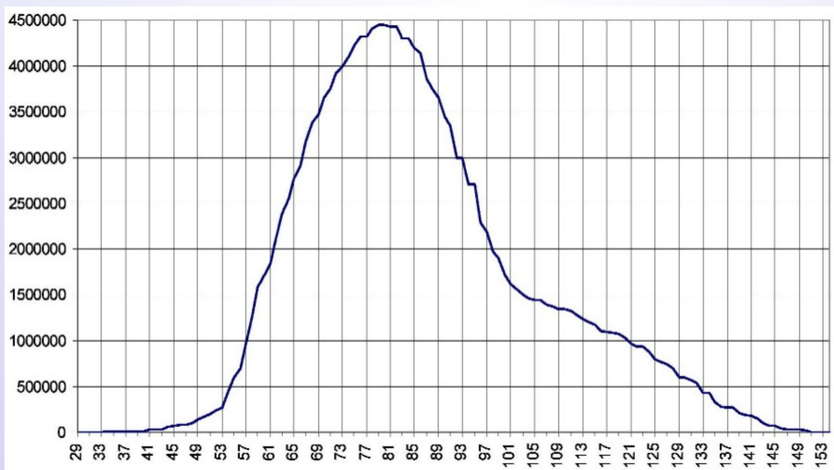
- energy of the set of objects $X \subset U$ can be defined by the number of pairs of objects from X to be discerned

$$conflict(X) = \sum_{i<j} N_i N_j$$

where $\langle N_1, ..., N_d \rangle$ is a class distribution of X

- The cut c which divides the set of objects U into U_1, and U_2 is evaluated by

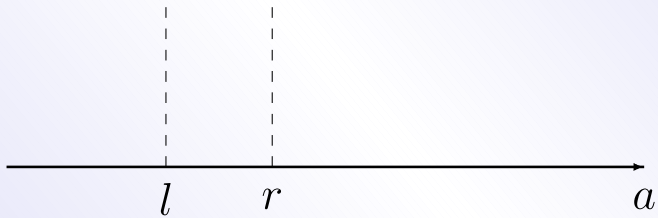
$$W(c) = conflict(U) - conflict(U_1) - conflict(U_2)$$



3. Soft cuts and soft DT

A soft cut is any triple $p = \langle a, l, r \rangle$, where

- $a \in A$ is an attribute,
- $l, r \in \Re$ are called the left and right bounds of p ;
- the value $\varepsilon = \frac{r-l}{2}$ is called the uncertain radius of p.
- We say that a soft cut p discerns a pair of objects x_1, x_2 if $a(x_1) < l$ and $a(x_2) > r$.



- The intuitive meaning of $p = \langle a, l, r \rangle$:
  - *there is a real cut somewhere between l and r.*
  - *for any value $v \in [l, r]$ we are not able to check if v is either on the left side or on the right side of the real cut.*
  - *$[l, r]$ is an uncertain interval of the soft cut p.*
  - *normal cut can be treated as soft cut of radius 0.*

3.1. Soft Decision Tree

- The test functions can be defined by soft cuts
- Here we propose two strategies using described above soft cuts:
  - *fuzzy decision tree*: any new object u can be classified as follows:
    - For every internal node, compute the probability that u turns left and u turns right;
    - For every leave L compute the probability that u is reaching L;
    - The decision for u is equal to decision labeling the leaf with largest probability.
  - *rough decision tree*: in case of uncertainty
    - Use both left and right subtrees to classify the new object;
    - Put together their answer and return the answer vector;
    - Vote for the best decision class.

4. Searching for soft cuts

STANDARD ALGORITHM FOR BEST CUT

- For a given attribute a and a set of candidate cuts $\{c_1, ..., c_N\}$, the best cut (a, c_i) with respect to given heuristic measure

$$F : \{c_1, ..., c_N\} \rightarrow \mathbb{R}^+$$

can be founded in time $\Omega(N)$.

- The minimal number of simple SQL queries of form

```
SELECT COUNT
FROM data_table
WHERE (a BETWEEN c_L AND c_R) GROUPED BY d.
```

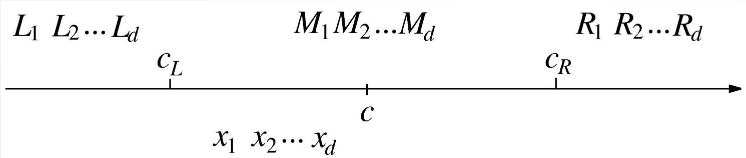
necessary to find out the best cut is $\Omega(dN)$

OUR PROPOSITIONS FOR SOFT CUTS

- Tail cuts can be eliminated
- Divide and Conquer Technique

4.1. Divide and Conquer Technique

- The algorithm outline:
  1. *Divide the set of possible cuts into k intervals*
  2. *Chose the interval to which the best cut may belong with the highest probability.*
  3. *If the considered interval is not STABLE enough then Go to Step 1*
  4. *Return the current interval as a result.*
- The number of SQL queries is $O(d \cdot k \log_k n)$ and is minimum for $k = 3$;
- How to define the measure evaluating the quality of the interval $[c_L; c_R]$?



- This measure should estimate the quality of the best cut from $[c_L; c_R]$.

Motivations
Basic notions
Soft cuts and soft DT
Searching for soft cuts
Conclusions

Home Page
Title Page
Page 13 of 16
Go Back
Full Screen
Close
Quit

We construct estimation measures for intervals in four cases:

	Discernibility measure	Entropy Measure
Independency assumption	?	?
Dependency assumption	?	?

4.2. Discernibility measure:

Under **dependency assumption**, i.e.

$$\frac{x_1}{M_1} \simeq \frac{x_2}{M_2} \simeq \ldots \simeq \frac{x_d}{M_d} \simeq \frac{x_1 + \ldots + x_d}{M_1 + \ldots + M_d} = \frac{x}{M} =: t \in [0,1]$$

discernibility measure for $[c_L; c_R]$ can be estimated by:

$$\frac{W(c_L) + W(c_R) + conflict(c_L; c_R)}{2} + \frac{[W(c_R) - W(c_L)]^2}{conflict(c_L; x_R)}$$

Under **dependency assumption**, i.e. $x_1, \ldots, x_d$ are independent random variables with uniform distribution over sets $\{0, \ldots, M_1\}, \ldots, \{0, \ldots, M_d\}$, respectively.

- The mean $E(W(c))$ for any cut $c \in [c_L; c_R]$ satisfies

$$E(W(c)) = \frac{W(c_L) + W(c_R) + conflict(c_L; c_R)}{2}$$

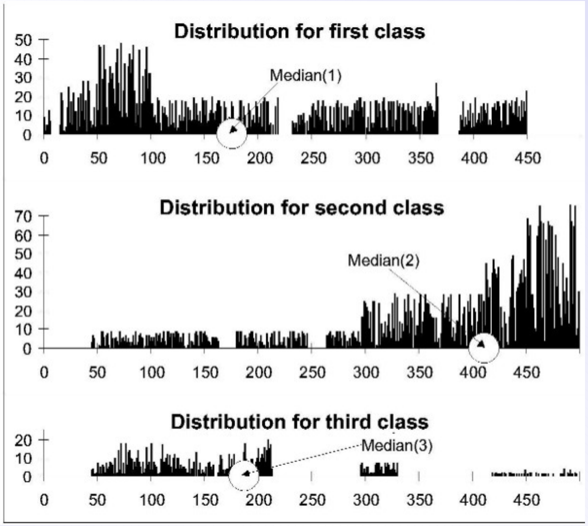
- and for the standard deviation of $W(c)$ we have

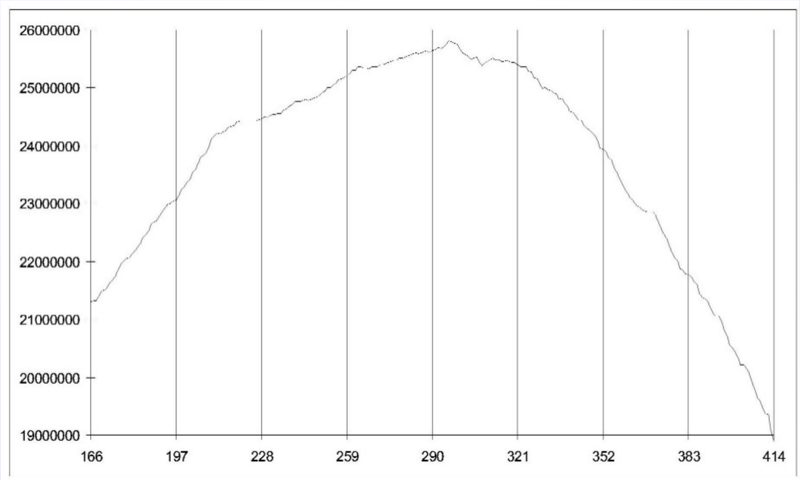
$$D^2(W(c)) = \sum_{i=1}^{n} \left[\frac{M_i(M_i+2)}{12} \left(\sum_{j \neq i} (R_j - L_j) \right)^2 \right]$$

- One can construct the measure estimating quality of the best cut in $[c_L; c_R]$ by

$$\boxed{Eval\left([c_L; c_R], \alpha\right) = E(W(c)) + \alpha \sqrt{D^2(W(c))}}$$

4.3. Example


Distribution for first class
Median(1)
Distribution for second class
Median(2)
Distribution for third class
Median(3)


26000000
25000000
24000000
23000000
22000000
21000000
20000000
19000000
166
197
228
259
290
321
352
383
414

Motivations
Basic notions
Soft cuts and soft DT
Searching for soft cuts
Conclusions

Home Page

Title Page

Page 16 of 16

Go Back

Full Screen

Close

Quit

5. Conclusions

- Soft cuts as a novel discretization concept;
- Soft decision tree;
- Efficient method for construction of soft cuts from large data (one can reduce the number of simple queries from $O(N)$ to $O(\log N)$ to construct the partition very close to the optimal one).