Decision tree

Hung Son Nguyen

Institute of Mathematics,
Warsaw University

February 15, 2006

Outline

1. Conflict measure
2. MD-heuristics
3. Searching for binary partition of symbolic values
4. Searching for cuts on numeric attributes
5. Searching for best cuts
   - Divide and Conquer Technique
   - Example
   - Discernibility measure:
6. Soft cuts and soft DT
   - Soft Decision Tree

Test functions

1. **Attribute-based tests:** $t_a(u) = a(u)$;
2. **Value-based tests:**

$$t_{a=v}(u) = \begin{cases} 1 & \text{if } a(u) = v \\ 0 & \text{otherwise;} \end{cases}$$

3. **Cut-based tests:**

$$t_{a>c}(u) = \begin{cases} 1 & \text{if } a(u) > c \\ 0 & \text{otherwise;} \end{cases}$$

4. **Value set based tests:**

$$t_{a\in S}(u) = \begin{cases} 1 & \text{if } a(u) \in S \\ 0 & \text{otherwise;} \end{cases}$$

5. **Hyperplane-based tests:**

$$t_{w_1a_1+\ldots+w_ka_k>w_0}(u) = \begin{cases} 1 & \text{if } w_1a_1(u) + \ldots + w_ka_k(u) > w_0 \\ 0 & \text{otherwise;} \end{cases}$$

Issues of decision tree induction methods

- Determine a collection of test functions;

$$\mathcal{T} = \{t_1, t_2, ..., t_m\}$$

- Estimation measure for tests;

$$\mathcal{F} : \mathcal{T} \times \mathcal{P}(U) \to \mathbb{R}$$

- Search algorithm: e.g., top-down
- Pruning techniques;

Outline

Conflict and discernibility measure

- A conflict measure can be defined by

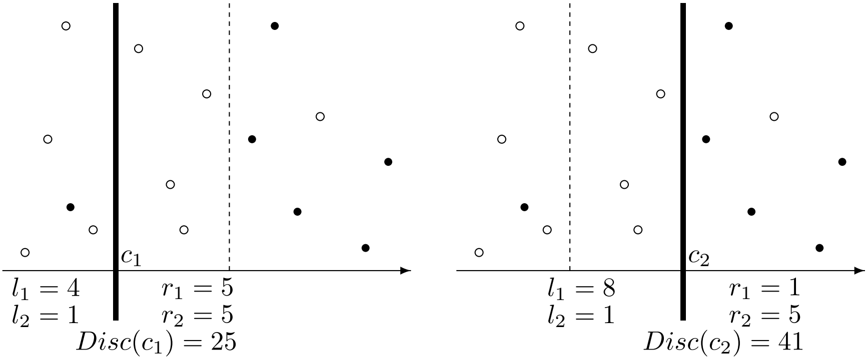
$$conflict(X) = \sum_{i<j} n_i n_j$$

  where $(n_1, ..., n_d)$ is the counting table of X, i.e., $n_i = |\{x \in X : dec(x) = i\}|$

- If a test t determines a partition of a set of objects X into $X_1, X_2, ..., X_{n_t}$, then discernibility measure for t is defined by

$$Disc(t, X) = conflict(X) - \sum_{i=1}^{n_t} conflict(X_i)$$

Example



Test functions in MD-heuristics

MD algorithm is using two kinds of tests depending on attribute types.

- For symbolic attributes $a_j \in A$, *test functions defined by sets of values*, i.e.,

$$t_{a_j \in V}(u) = 1 \iff [a_j(u) \in V]$$

  where $V \subset V_{a_j}$, are considered.

- For numeric attributes $a_i \in A$, only *test functions defined by cuts*:

$$t_{a_i > c}(u) = True \iff [a_i(u) \le c] \iff [a_i(u) \in (-\infty; c\rangle)]$$

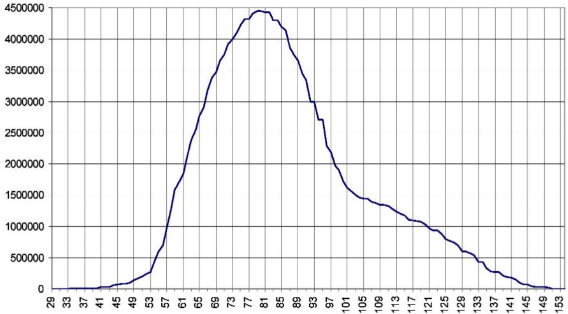
  where c is a *cut* in V_{a_i}, are considered.

MD algorithm

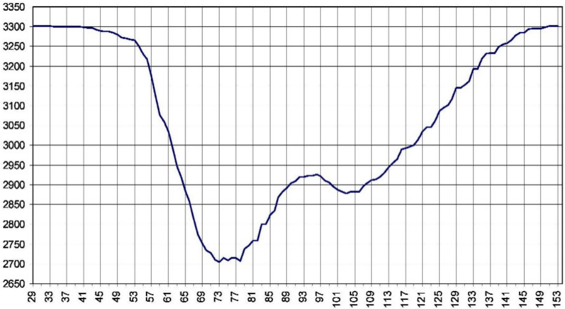
Initialize a decision tree $\mathbf{T}$ with one node labeled by the set of all objects U;
$\mathbf{Q} := [\mathbf{T}]$; {*Initialize a FIFO queue* $\mathbf{Q}$ *containing* $\mathbf{T}$}
while $\mathbf{Q}$ is not empty **do**
    $N := \mathbf{Q}.head()$; {*Get the first element of the queue*}
    $X := N.Label$;
    if the major class of X is large enough **then**
        $N.Label := major\_class(X)$;
    else
        $t := ChooseBestTest(X)$;
        {*Search for best test of form* $t_{a \in V}$ *for* $V \subset V_a$ *with respect to* $Disc(., X)$}
        $N.Label := t$;
        Create *two successors of the current node* N_L *and* N_R *and label them by* X_L *and* X_R*, where*

$$X_L = \{u \in X : t(u) = 0\} \quad X_R = \{u \in X : t(u) = 1\}$$

        $\mathbf{Q}.insert(N_L, N_R)$;{*Insert* N_L *and* N_R *into* $\mathbf{Q}$}
    end if
end while

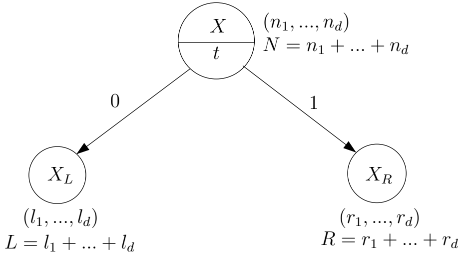


Discernibility



Entropy

Properties of MD-heuristics



$$Disc(t, X) = LR - \sum_{i=1}^{d} l_i r_i$$

$$Disc(t, X) = \sum_{i=1}^{d} l_i \sum_{i=1}^{d} r_i - \sum_{i=1}^{d} l_i r_i$$

$$Disc(t, X) = \sum_{i \neq j} l_i r_j$$

$$
\begin{aligned}
Disc(t, X) &= conflict(X) - conflict(X_1) - conflict(X_2) \\
&= \frac{1}{2}\sum_{i \neq j} n_i n_j - \frac{1}{2}\sum_{i \neq j} l_i l_j - \frac{1}{2}\sum_{i \neq j} r_i r_j \\
&= \frac{1}{2}\left(N^2 - \sum_{i=1}^{d} n_i^2\right) - \frac{1}{2}\left(L^2 - \sum_{i=1}^{d} l_i^2\right) - \frac{1}{2}\left(R^2 - \sum_{i=1}^{d} r_i^2\right) \\
&= \frac{1}{2}\left(N^2 - L^2 - R^2\right) - \frac{1}{2}\sum_{i=1}^{d}(n_i^2 - l_i^2 - r_i^2) \\
&= \frac{1}{2}\left[(L+R)^2 - L^2 - R^2\right] - \frac{1}{2}\sum_{i=1}^{d}[(l_i + r_i)^2 - l_i^2 - r_i^2] \\
&= LR - \sum_{i=1}^{d} l_i r_i
\end{aligned}
$$

Outline

Problem

For a fixed attribute a and an object set $X \subset U$, we define the *discernibility degree* of a partition $P = (V_1, V_2)$ as follows

$$\begin{aligned} Disc_a(P|X) &= Disc(t_{a \in V_1}, X) \\ &= \left| \{(x, y) \in X^2 : x, y \text{ are discerned by } P\} \right| \end{aligned}$$

MD-PARTITION:

input: A set of objects X and an symbolic attribute a.

output: A binary partition P of V_a such that $Disc_a(P|X)$ is maximal.

Let $\mathbf{s}(v_i) = (n_1(v_i), n_2(v_i), ..., n_d(v_i))$ denote the counting table of the set $X_{v_i} = \{x \in X : a(x) = v_i\}$. The distance between two symbolic values $v, w \in V_a$ is determined as follows:

$$\delta_{disc}(v, w) = Disc(v, w) = \sum_{i \neq j} n_i(v) \cdot n_j(w)$$

One can generalize the definition of distance function by

$$\delta_{disc}(V_1, V_2) = \sum_{v \in V_1, w \in V_2} \delta_{disc}(v, w)$$

For arbitrary sets of values V_1, V_2, V_3

$$\delta_{disc}(V_1 \cup V_2, V_3) = \delta_{disc}(V_1, V_3) + \delta_{disc}(V_2, V_3) \quad (1)$$

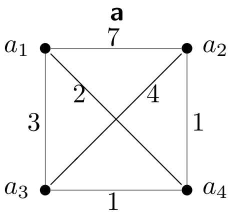
$$\delta_{disc}(V_1, V_2) = \delta_{disc}(V_2, V_1) \quad (2)$$

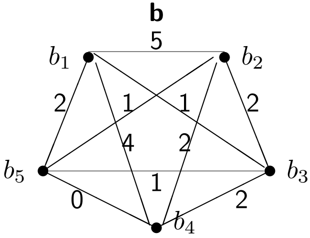
Example

A	a	b	dec
u_1	a_1	b_1	1
u_2	a_1	b_2	1
u_3	a_2	b_3	1
u_4	a_3	b_1	1
u_5	a_1	b_4	2
u_6	a_2	b_2	2
u_7	a_2	b_1	2
u_8	a_4	b_2	2
u_9	a_3	b_4	2
u_{10}	a_2	b_5	2

	$dec = 1$	$dec = 2$
a_1	2	1
a_2	1	3
a_3	1	1
a_4	0	1

	$dec = 1$	$dec = 2$
b_1	2	1
b_2	1	2
b_3	1	0
b_4	0	2
b_5	0	1



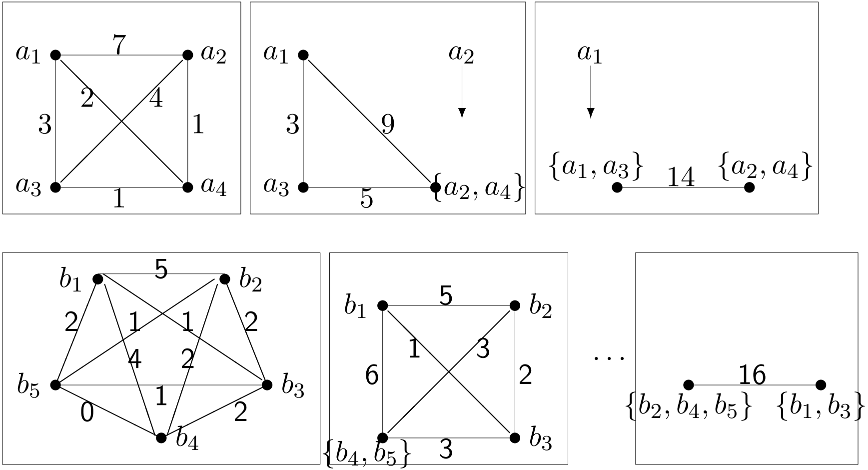


Heuristics

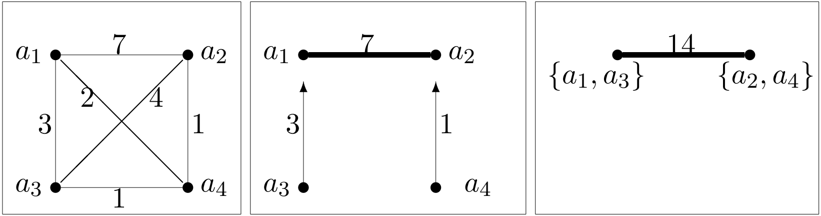
We have proposed the following heuristics for MD-PARTITION problem:

1. *grouping by minimizing conflict*: a kind of agglomerative hierarchical clustering algorithm
2. *grouping by maximizing discernibility.*

grouping by minimizing conflict



grouping by maximizing discernibility



Outline

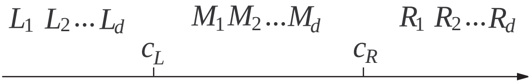
Let us consider two cuts $c_L < c_R$ on attribute a.

Lemma

The following equation holds:

$$Disc(c_R) - Disc(c_L) = \sum_{i=1}^{d} \left[(R_i - L_i) \sum_{j \neq i} M_j \right] \tag{3}$$

where $(L_1, ..., L_d)$, $(M_1, ..., M_d)$ *and* $(R_1, ..., R_d)$ *are the counting tables of intervals* $(-\infty; c_L)$, $[c_L; c_R)$ *and* $[c_R; \infty)$, *respectively (see Figure* **??***).*



Boundary cuts

Definition

The cut $c_i \in \mathbf{C}_a$, where $1 < i < N$, is called the *boundary cut* if there exist at least two such objects $u_1, u_2 \in U$ that $a(u_1) \in [c_{i-1}, c_i)$, $a(u_2) \in [c_i, c_{i+1})$ and $dec(u_1) \neq dec(u_2)$.

Theorem

The cut c_{Best} maximizing the function $Disc(a, c)$ can be found among boundary cuts.

Tail cuts

Definition

By a median of the k^{th} decision class we mean a cut $c \in \mathbf{C}_a$ which minimizing the value $|L_k - R_k|$. The median of the k^{th} decision class will be denoted by $Median(k)$.

Let $c_1 < c_2 ... < c_N$ be the set of consecutive candidate cuts, and let

$$c_{min} = \min_i \{Median(i)\} \text{ and } c_{max} = \max_i \{Median(i)\}$$

Then we have the following theorem:

Theorem

The quality function $Disc : \{c_1, ..., c_N\} \rightarrow \mathbb{N}$ *defined over the set of cuts is increasing in* $\{c_1, ..., c_{min}\}$ *and decreasing in* $\{c_{max}, ..., c_N\}$. *Hence*

$$c_{Best} \in \{c_{min}, ..., c_{max}\}$$

Properties of MD-heuristics

Theorem

In case of decision tables with two decision classes, any single cut c_i*, which is a local maximum of the function* $Disc$*, resolves more than half of conflicts in the decision table, i.e.*

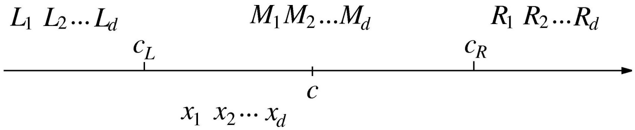
$$Disc\left(c_i\right) \geq \frac{1}{2} \cdot \mathit{conflict}\left(\mathbb{S}\right)$$

Theorem

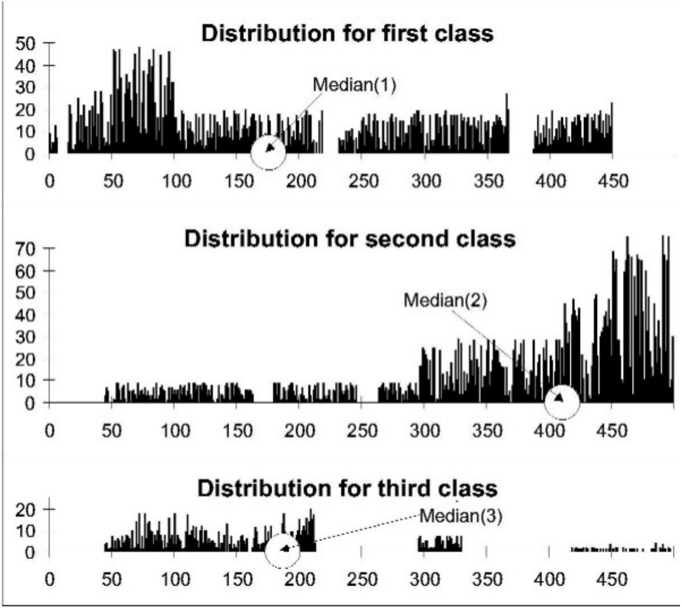
In case of decision table with two decision classes and n *objects, the height of the MD decision tree using hyperplanes is not larger than* $2\log n - 1$*.*

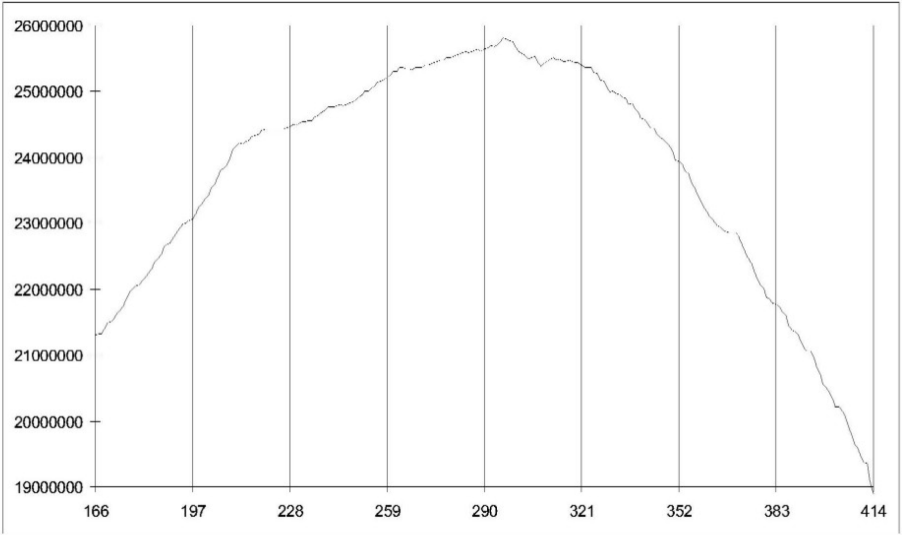
Outline

- 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 $[c_L; c_R]$.


Distribution for first class
Median(1)
50
40
30
20
10
0
0 50 100 150 200 250 300 350 400 450
Distribution for second class
Median(2)
70
60
50
40
30
20
10
0
0 50 100 150 200 250 300 350 400 450
Distribution for third class
Median(3)
20
10
0
0 50 100 150 200 250 300 350 400 450


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

We construct estimation measures for intervals in four cases:

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

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

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

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. $x_1, ..., x_d$ are independent random variables with uniform distribution over sets $\{0, ..., M_1\}$, ..., $\{0, ..., M_d\}$, 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

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

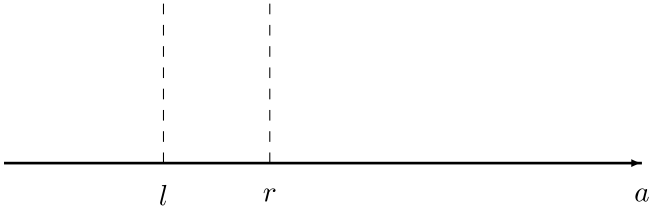
- One can construct the measure estimating quality of the best cut in $[c_L; c_R]$ by

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

Outline

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.*

- 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.

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

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

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

- The minimal number of simple SQL queries of form

  SELECT COUNT
  FROM datatable
  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