Dynamic Programming for Design and Analysis of Decision Trees

Mikhail Moshkov
King Abdullah University of Science and Technology
Saudi Arabia

School for Advanced Sciences of Luchon

July 10, 2015

Research Group

Fawaz Alsolami

Ph.D. Student, Computer Science

Research Interest: Machine learning, Inhibitory Rules, Dynamic programming, Greedy algorithms

fawaz.alsolami@kaust.edu.sa

Hassan AbouEisha

Ph.D. Student, Computer Science

Research Interest: Dynamic programming, Greedy algorithms, Finite element mesh solvers

hassan.aboueisha@kaust.edu.sa

Mohammad Mohiuddin Azad

Ph.D. Student, Computer Science

Research Interest: Dynamic programming, Decision rules, Decision trees, Machine learning, Combinatorial machine learning, Greedy algorithms, Sequential optimization

mohammad.azad@kaust.edu.sa

Shahid Hussain

Ph.D. Student, Computer Science

Research Interest: Discrete optimization, Dynamic programming, Decision trees, Machine learning, Greedy algorithms

shahid.hussain@kaust.edu.sa

Talha Amin

Ph.D. Student, Computer Science

Research Interest: Dynamic programming, Decision rules, Greedy algorithms, Machine learning

talha.amin@kaust.edu.sa

Research Group

Dr. Igor Chikalov
Consultant

Monther Busbait

Alumni

- Dr. Beata Zielosko, SRS
- Abdulaziz Alkhalid, PhD student
- Chandra Prasetyo Utomo, MS student with thesis
- Enas Mohammad, MS student with thesis
- Malek A. Mahayni, MS student with thesis
- Maram Alnafie, Dir. Res.
- Jewahir AbuBekr, Dir. Res
- Majed Alzahrani, Dir. Res.
- Saad Alrawaf, Dir. Res.
- Mohammed Al Farhan, Dir. Res.
- Liam Mencel, Dir. Res.

“Greatest Problem of Science Today”

- Tomaso Poggio and Steve Smale, The mathematics of learning: dealing with data, Notices of The AMS, Vol. 50, Nr. 5, 2003, 537-544

- The problem of understanding intelligence is said to be the greatest problem in science today and “the” problem for this century—as deciphering the genetic code was for the second half of the last one

Remark from KDnuggets

- http://www.kdnuggets.com/2013/11/top-conferences-data-mining-data-science.html
- While there is now a glut of industry and business oriented conferences on Big Data and Data Science, the technology which powers the current boom in Big Data comes from research … (after that – a list of top research conferences in Data Mining, Data Science)

Dynamic Programming

- The idea of dynamic programming is the following. For a given problem, we define the notion of a sub-problem and an ordering of sub-problems from “smallest” to “largest”
- If (i) the number of sub-problems is polynomial, and (ii) the solution of a sub-problem can be easily (in polynomial time) computed from the solution of smaller sub-problems then we can design a polynomial algorithm for the initial problem

Dynamic Programming

- The aim of usual Dynamic Programming (DP) is to find an optimal object from a finite set of objects

Extensions of DP

We consider extensions of dynamic programming which allow us

- To describe the set of optimal objects
- To count the number of these objects
- To make sequential optimization relative to different criteria
- To find the set of Pareto optimal points for two criteria
- To describe relationships between two criteria

Extensions of DP

The areas of applications include

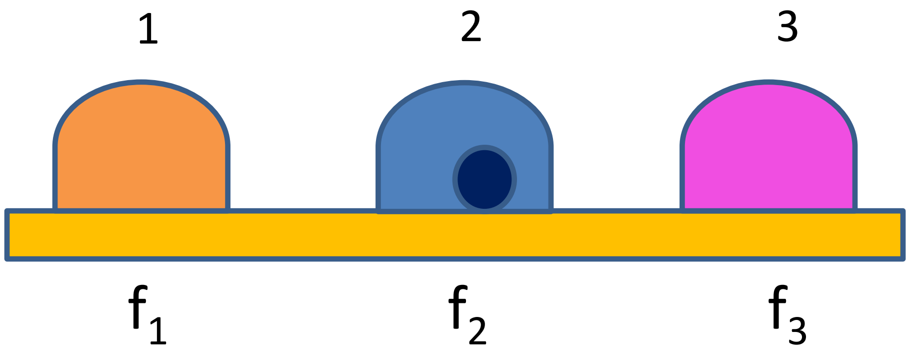
- Combinatorial optimization
- Finite element method
- Fault diagnosis
- Complexity of algorithms
- Machine learning
- Knowledge representation

Applications for Decision Trees

In the presentation, we consider applications of this new approach to the study of decision trees

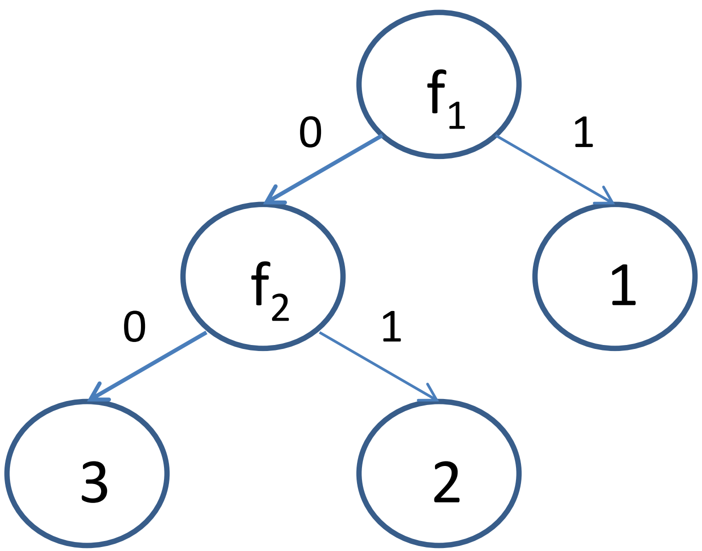
- As algorithms for problem solving
- As a way for knowledge extraction and representation
- As predictors which, for a new object given by values of conditional attributes, define a value of the decision attribute

Decision Trees



f_1	f_2	f_3	d
1	0	0	1
0	1	0	2
0	0	1	3

Decision table

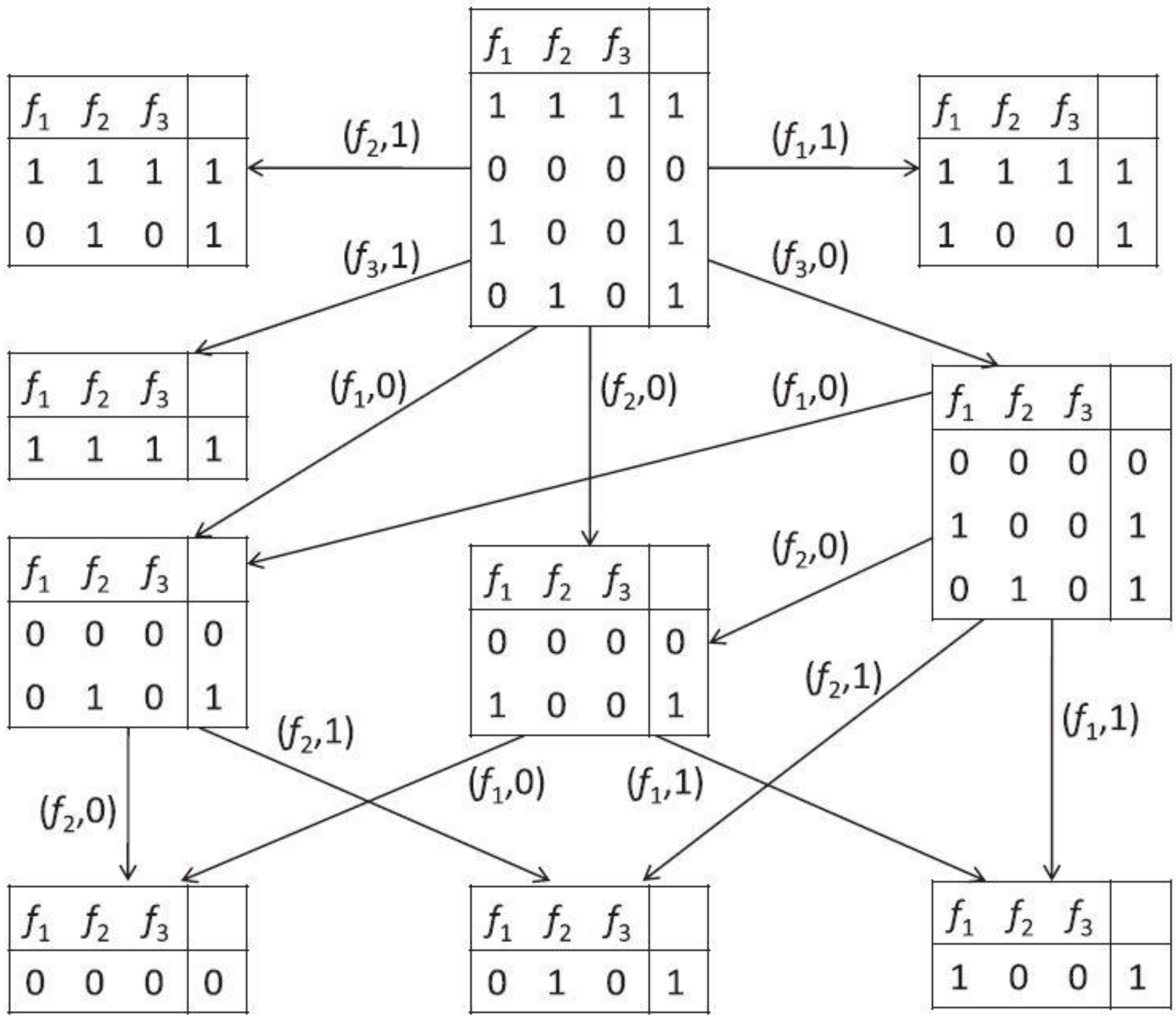


Decision tree

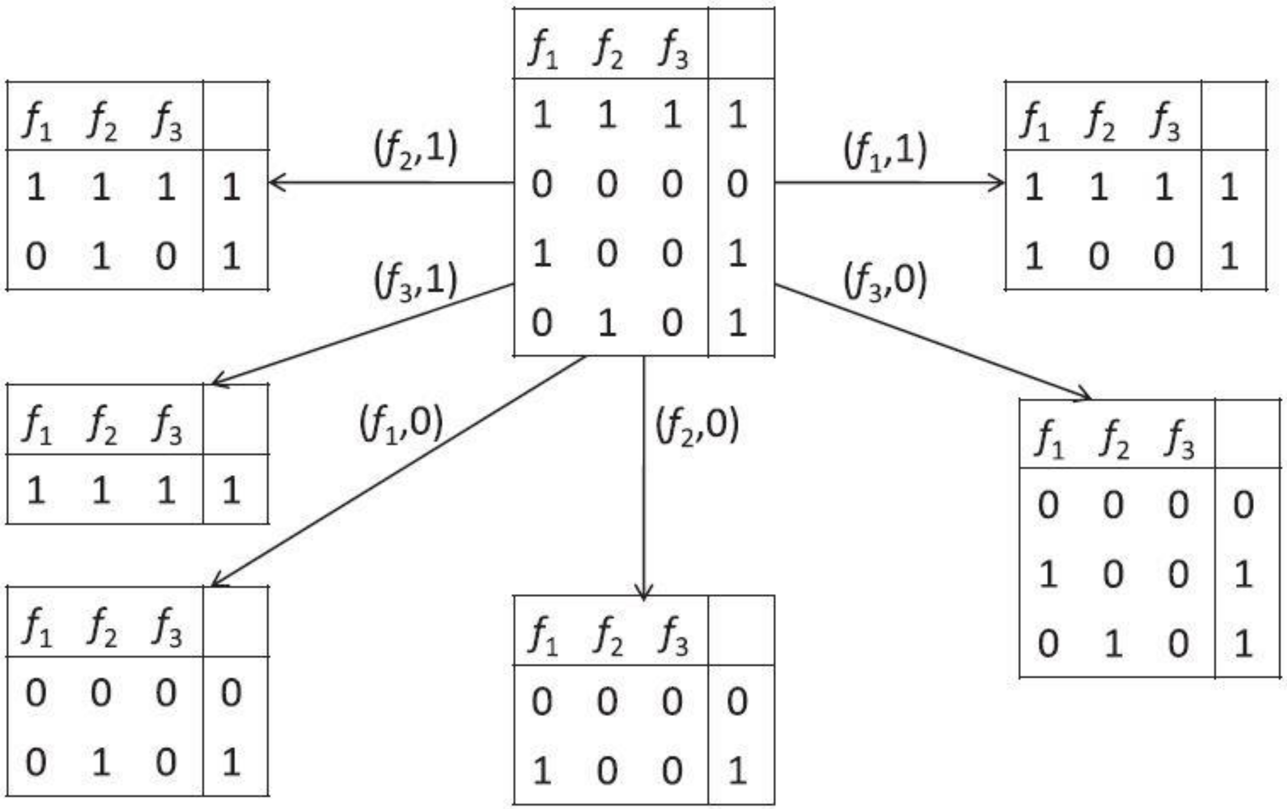
Depth
Number of nodes
Total path length (average depth)
Number of terminal nodes

Cost functions

Directed Acyclic Graph $\Delta_0(T)$



Directed Acyclic Graph $\Delta_{\alpha}(T)$



About Scalability

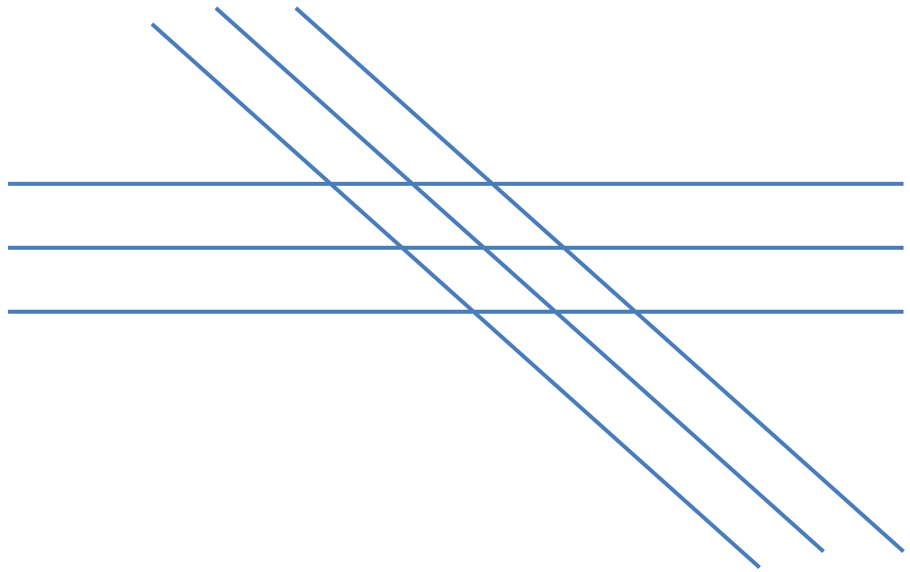
Table 1. Exeperimental results for Poker Hand data set

sf	nodes	time	optimal			greedy		
			depth	avg depth	# nodes	depth	avg depth	# nodes
0	1426236	177	5	4.08	18831	5	4.15	22989
10^{-8}	1112633	124	5	3.99	15766	5	4.03	20071
10^{-7}	293952	27	4	3.73	6658	4	3.82	15966
10^{-6}	79279	7	3	3	2269	3	3	2381
10^{-5}	15395	2	3	3	733	3	3	2381
10^{-4}	4926	< 1	2	2	183	2	2	183
10^{-3}	246	< 1	2	2	57	2	2	183
10^{-2}	21	< 1	1	1	14	1	1	14
10^{-1}	1	< 1	1	1	5	1	1	14

Training part of Poker Hand data set contains 25010 objects and 10 conditional attributes

Restricted Information Systems

- We described classes of decision tables for which the considered algorithms have polynomial time complexity depending on the number of conditional attributes

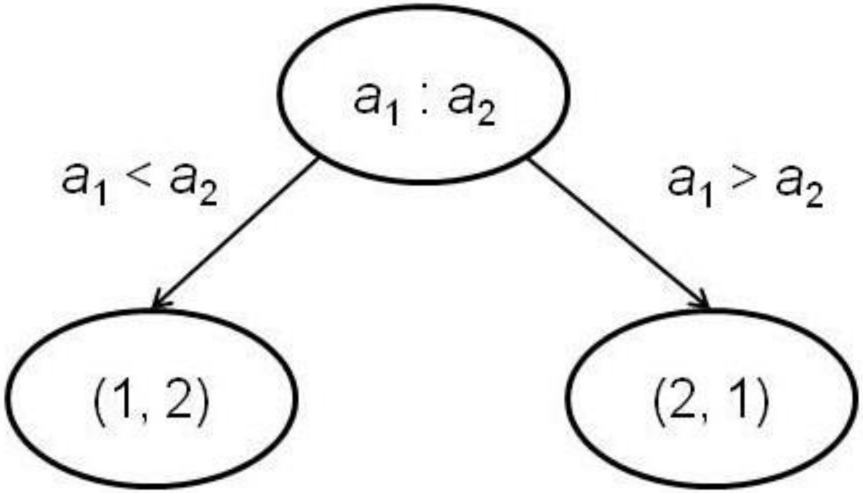


Extensions of DP for Decision Trees

- Sequential optimization
- Evaluation of the number of optimal trees
- Relationships between cost and accuracy
- Relationships between two cost functions
- Construction of the set of Pareto optimal points

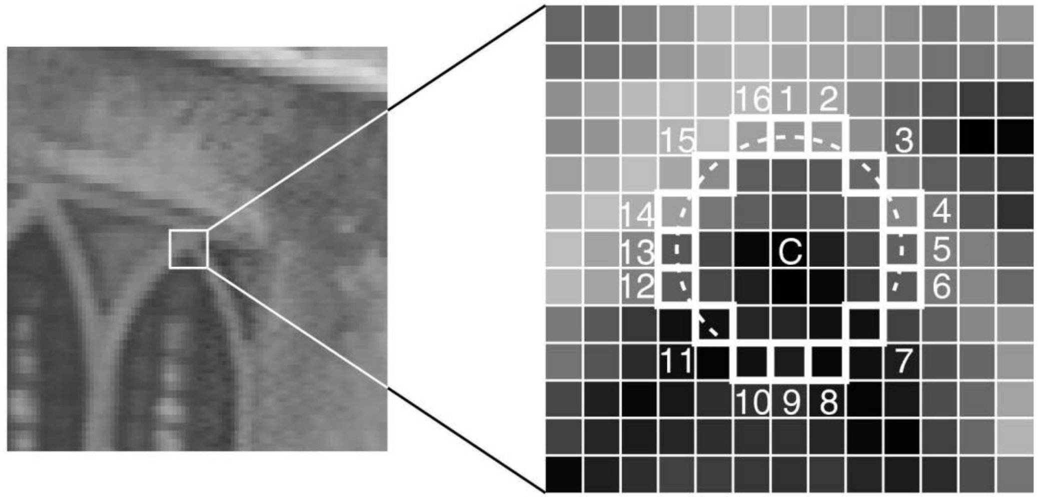
Sorting of 8 Elements

- We proved that the minimum average depth of a decision tree for sorting 8 elements is equal to 620160/40320



- This solved a long-standing problem (since 1968) considered by D. Knuth in his famous book The Art of Computer Programming, Volume 3, Sorting and Searching
- We proved also that each decision tree for sorting 8 elements with minimum average depth has minimum depth. The number of such trees is equal to 8.548×10^{326365}

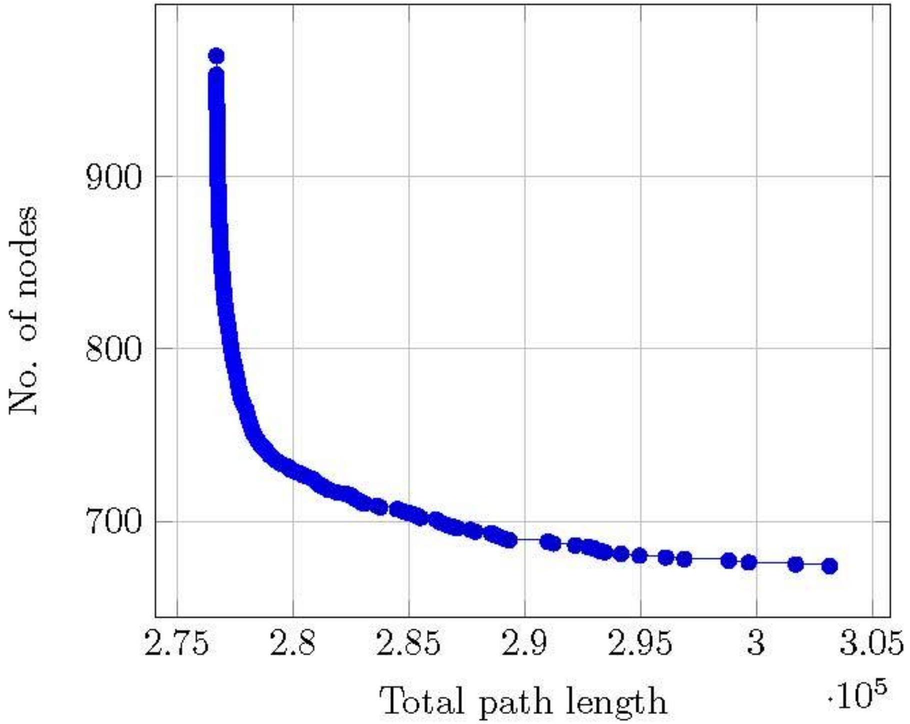
Corner Point Detection



Corner points are used in computer vision for object tracking (FAST algorithm devised by Rosten and Drummond)

A pixel is assumed to be a *corner point* if at least 12 contiguous pixels on the circle are all either brighter or darker than the central point by a given threshold

Corner Point Detection

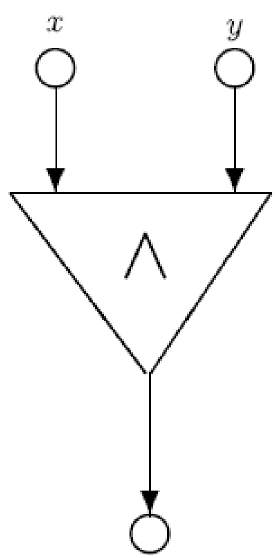


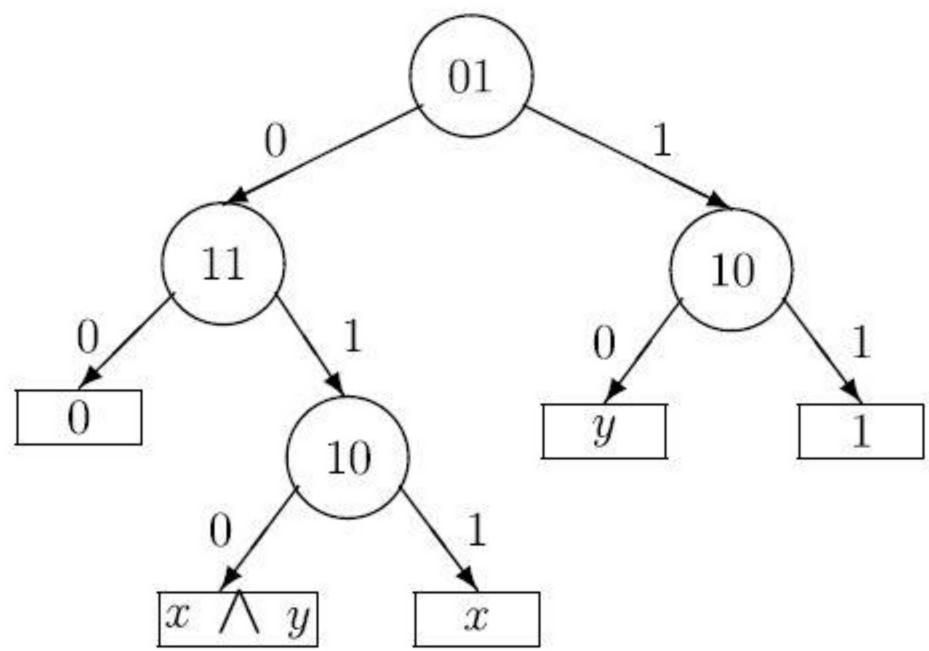
Dynamic programming approach allows us to construct decision trees for corner point detection with average time complexity 7% less than for known ones, and analyze time-memory tradeoff for such trees

Diagnosis of 0-1 Faults

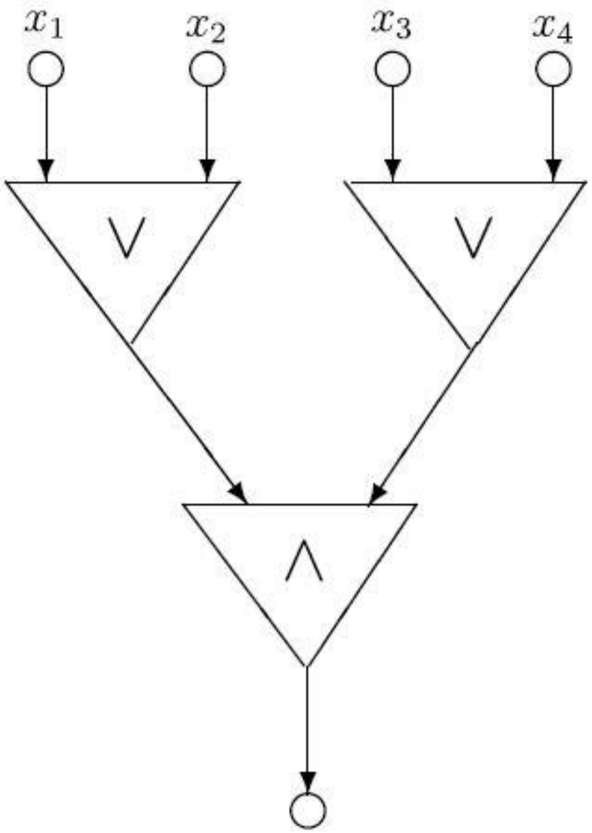
Number $M(n)$ of monotone Boolean functions with n, $1 \leq n \leq 5$, variables.

n	1	2	3	4	5
$M(n)$	3	6	20	168	7581





Diagnosis of 0-1 Faults



$$h(S) \le \begin{cases} (n+1)L(S), & 1 \le n \le 4, \\ (n+2)L(S), & n = 5. \end{cases}$$

Values of $H(n)$ and $\varphi(n)$ for $n = 1, \ldots, 5$.

n	1	2	3	4	5
$H(n)$	2	3	4	5	7
$\varphi(n)$	2	3	6	10	20

Totally Optimal Decision Trees for Boolean Functions

Table 1: The number of monotone boolean functions, $M(n)$, and the number of boolean functions, $B(n)$, with $n = 0, \ldots, 7$ variables.

n	0	1	2	3	4	5	6	7
$M(n)$	2	3	6	20	168	7581	7828354	2414682040998
$B(n)$	2	4	16	256	65536	4.2×10^9	1.8×10^{19}	3.4×10^{38}

Totally Optimal Decision Trees for Boolean Functions

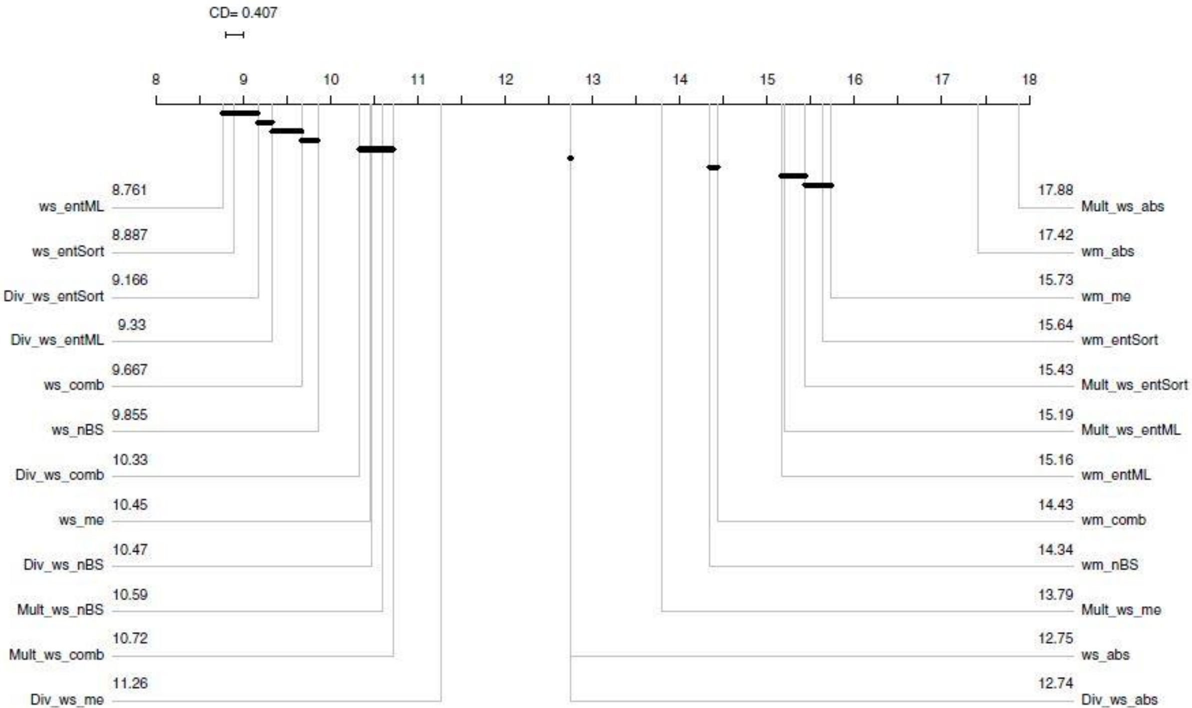
Table 2: The existence (example f_i) or nonexistence (—) of a monotone boolean function (MON) or a boolean function (ALL) with n variables which does not have totally optimal decision trees relative to a subset of the set of parameters $\{D =$ depth, $T =$ total path length, $N =$ number of nodes $\}$.

n	$\{D,N\}$		$\{D,T\}$		$\{T,N\}$		$\{D,T,N\}$	
	MON	ALL	MON	ALL	MON	ALL	MON	ALL
0	—	—	—	—	—	—	—	—
1	—	—	—	—	—	—	—	—
2	—	—	—	—	—	—	—	—
3	—	—	—	—	—	—	—	—
4	—	—	—	f_3	—	—	—	f_3
5	—	f_1	—	f_3	—	f_4	—	f_3
6	—	f_1	f_2	f_3	f_2	f_4	f_2	f_3
7	f_5	f_1	f_2	f_3	f_2	f_4	f_2	f_3
> 7	f_5	f_1	f_2	f_3	f_2	f_4	f_2	f_3

Totally Optimal Decision Trees for Boolean Functions

$$
\begin{aligned}
f_1 &= x_1\bar{x}_2\bar{x}_3\bar{x}_4 \vee \bar{x}_1\bar{x}_2x_3 \vee \bar{x}_1x_3x_5 \vee \bar{x}_1x_4 \vee x_2x_4 \vee x_3x_4x_5 \\
f_2 &= x_1x_2x_4 \vee x_1x_4x_5 \vee x_5x_6 \vee x_3x_4 \vee x_3x_6 \\
f_3 &= \bar{x}_1x_2\bar{x}_4 \vee \bar{x}_1x_3x_4 \vee \bar{x}_2\bar{x}_3 \\
f_4 &= x_1\bar{x}_2\bar{x}_3x_5 \vee x_1x_3\bar{x}_4\bar{x}_5 \vee x_1x_4x_5 \vee \bar{x}_1x_2x_3x_5 \vee \bar{x}_1\bar{x}_2x_3\bar{x}_5 \\
&\vee \bar{x}_1\bar{x}_2\bar{x}_3x_4 \vee \bar{x}_1x_4\bar{x}_5 \vee x_2\bar{x}_3x_4\bar{x}_5 \\
f_5 &= x_1x_2x_5x_7 \vee x_1x_2x_6x_7 \vee x_1x_3x_6x_7 \vee x_1x_4x_6x_7 \vee x_2x_3x_6x_7 \\
&\vee x_2x_5x_6x_7 \vee x_1x_4x_5 \vee x_2x_4x_5 \vee x_3x_4x_5
\end{aligned}
$$

Heuristics for Decision Tree Construction



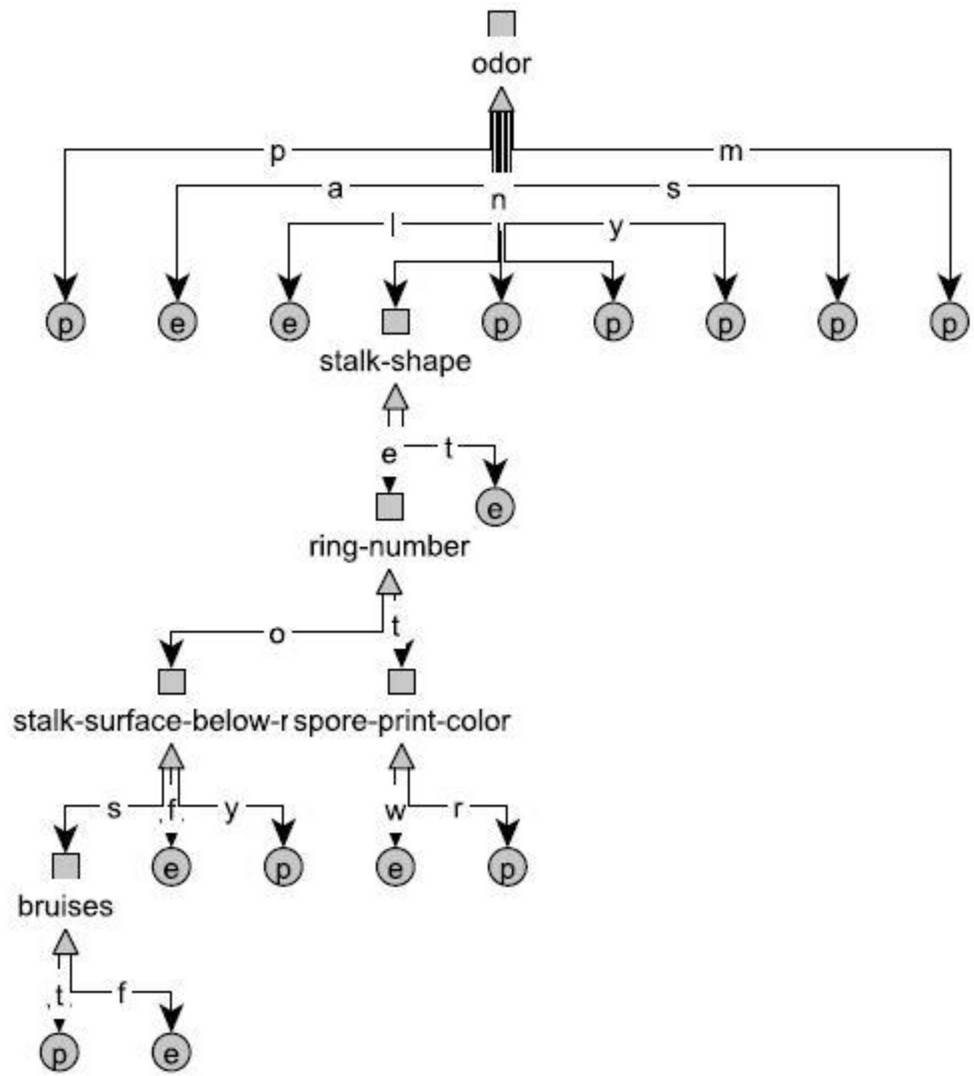
Minimization of decision tree average depth for decision tables with many-valued decisions

Algorithm	*ARD*
ws_entML	3.26%
ws_entSort	3.49%
Div_ws_entSort	4.53%

Minimization of Number of Nodes

Decision table *Mushroom* contains 22 conditional attributes and 8124 rows

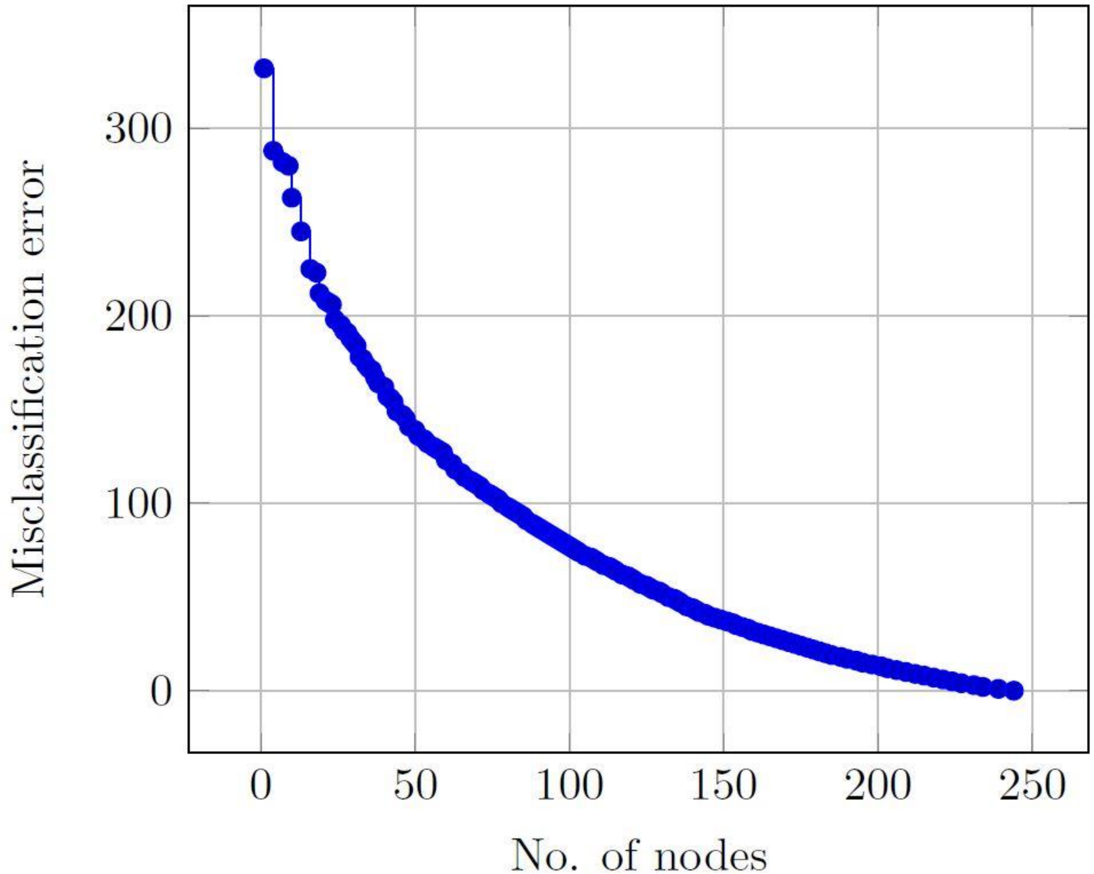
The minimum number of nodes in a decision tree for *Mushroom* is equal to 21



Relationships Number of Nodes vs. Misclassification Error

When the number of misclassifications is increasing, the number of nodes in decision trees can decrease

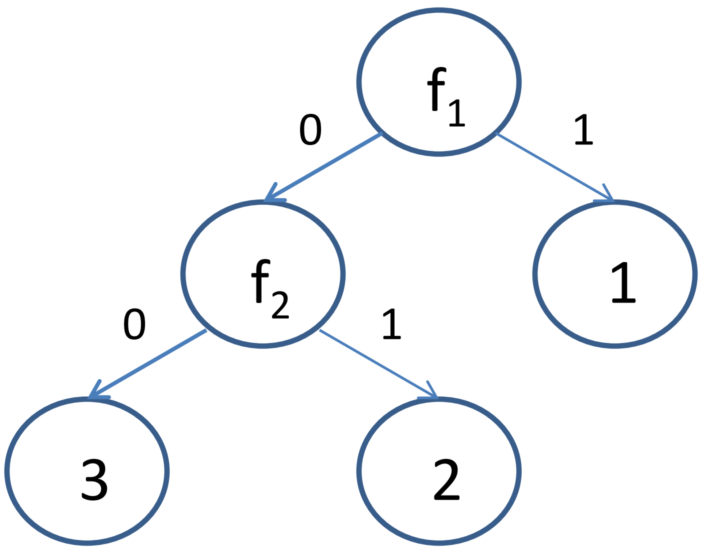
One can be interested in less accurate but more understandable decision trees



Tic Tac Toe, 9 attributes, 959 rows

Decision Trees and Rules

- Decision rules are widely used in machine learning and for knowledge representation
- One of the ways to obtain decision rules is to construct a decision tree and derive rules from this tree

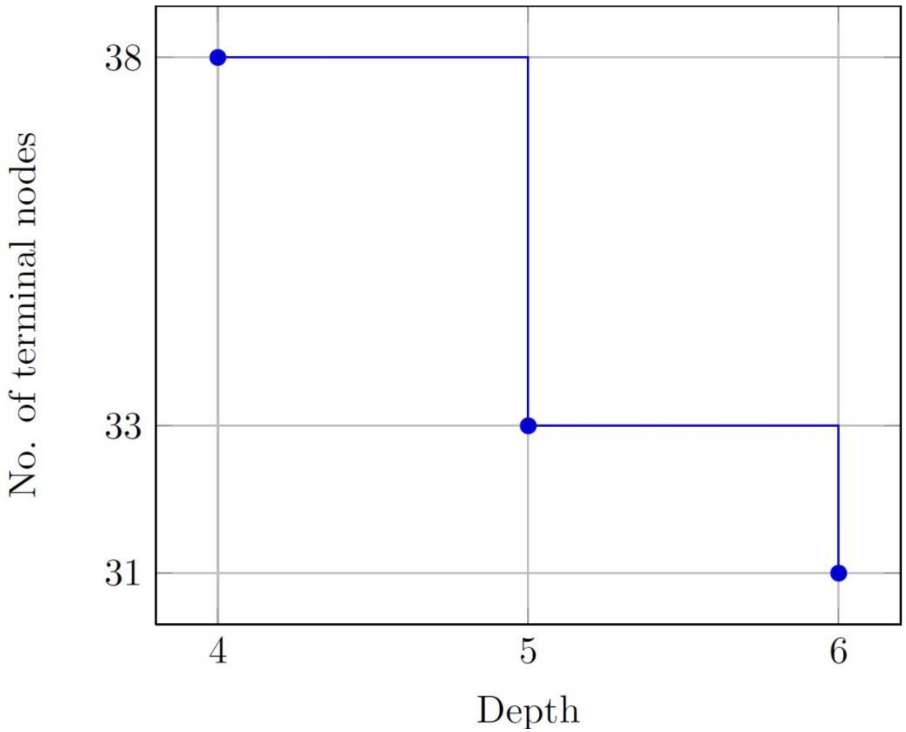


Decision tree

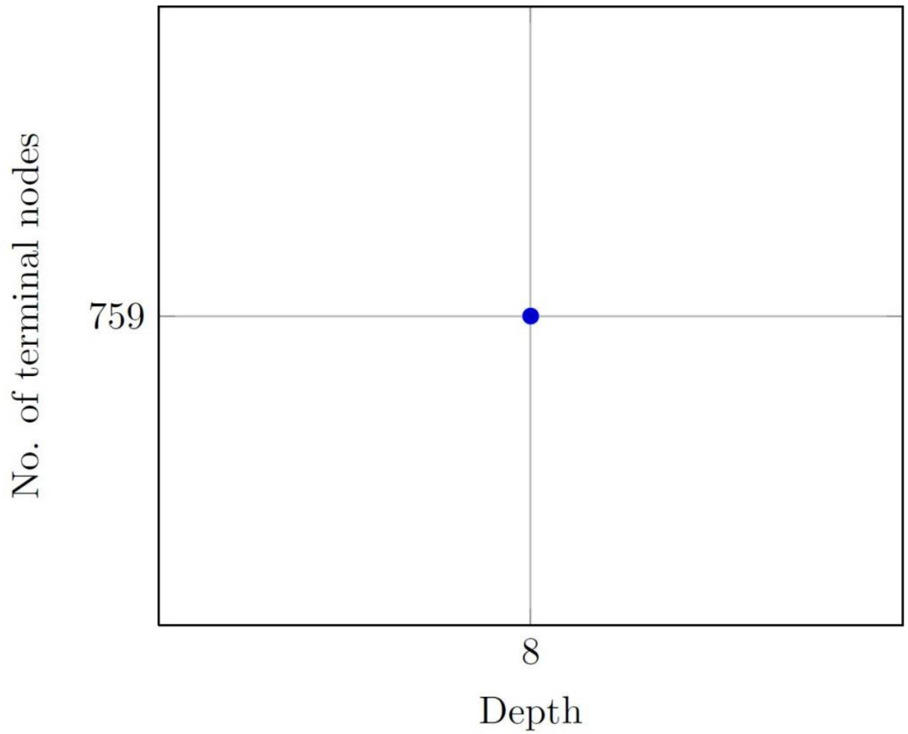
$$f_1 = 0 \wedge f_2 = 0 \rightarrow d = 3$$
$$f_1 = 0 \wedge f_2 = 1 \rightarrow d = 2$$
$$f_1 = 1 \rightarrow d = 1$$

Set of decision rules

Relationships Depth vs. Number of Terminal Nodes



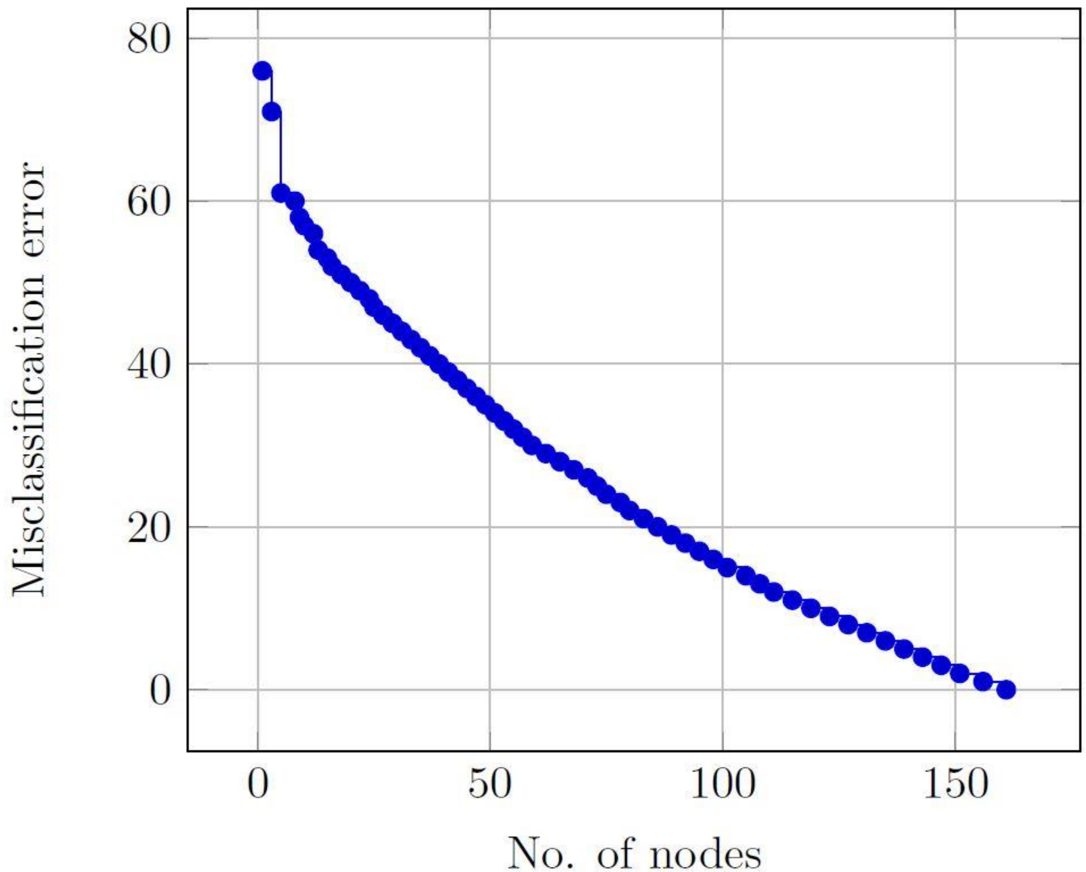
Lymphography, 18 attributes, 148 rows



Nursery, 8 attributes, 12960 rows

Relationships Number of Nodes vs. Misclassification Error

Relationships between the number of nodes and the number of misclassifications can be used in a special procedure of pruning

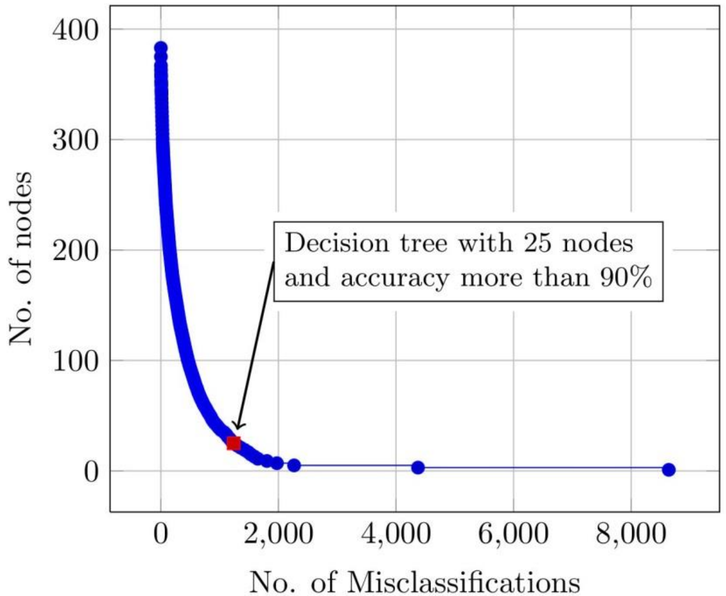


Breast cancer, 9 attributes, 266 rows

Pareto-Optimal Points (POPs) for Bi-Criteria Optimization of Decision Trees

We consider the number of nodes and number of misclassifications as two criteria for decision trees. Construction of the set of POPs allows us:

- To find relatively small and accurate decision trees which represent the knowledge contained in the dataset

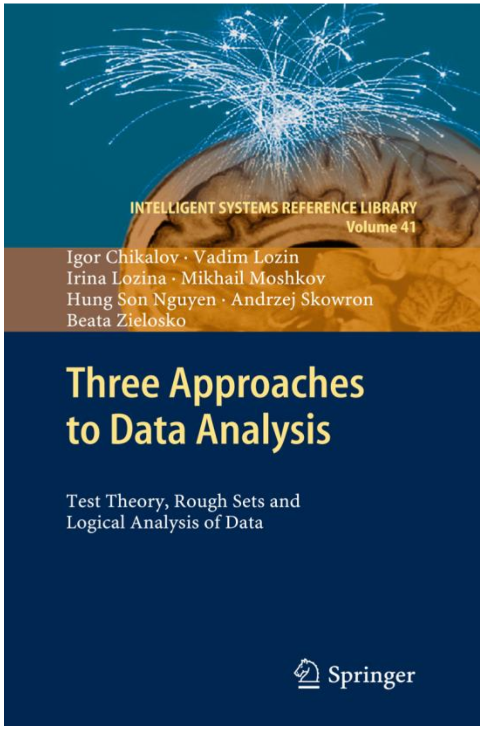


Dataset NURSERY with 9 attributes and 12960 objects

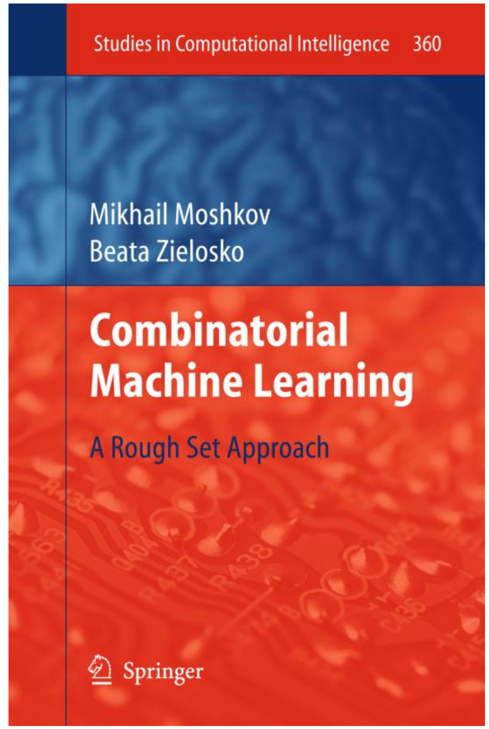
- To build classifiers using new multi-pruning procedure (MP) which outperform classifiers constructed by well known CART method

Dataset	Objects	Attr.	MP	CART
BALANCE-SCALE	625	5	23.26	23.75
BREAST-CANCER	266	10	29.02	29.77
CARS	1728	7	4.69	5.23
HAYES-ROTH-DATA	69	5	24.33	34.44
HOUSE-VOTES-84	279	17	6.88	6.60
LENSES	10	5	16.00	28.00
LYMPHOGRAPHY	148	19	25.14	27.70
NURSERY	12960	9	1.44	1.38
SHUTTLE-LANDING	15	7	54.29	46.25
SOYBEAN-SMALL	47	36	6.74	18.75
SPECT-TEST	169	23	4.74	5.21
TIC-TAC-TOE	958	10	7.85	10.73
ZOO-DATA	59	17	21.36	22.01
BANKNOTE	1372	5	2.05	3.38
IRIS	150	5	5.43	5.71
GLASS	214	10	38.31	39.82
WINE	178	13	8.99	11.80
		Average error:	*16.50*	*18.85*

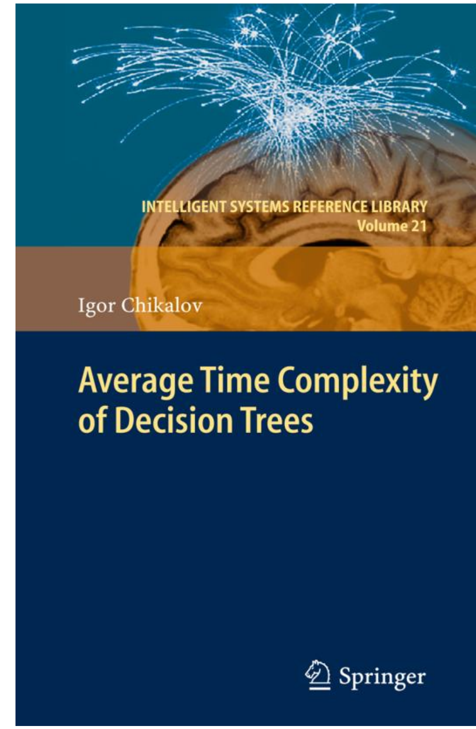
Three Books Published by Springer



“Bridge" among three approaches in Data Analysis which previously were not connected



Textbook for the course CS361 in KAUST



Research monograph

New Book and New Course

Extensions of Dynamic Programming for Combinatorial Optimization and Data Mining

KAUST

KAUST

- KAUST is an international graduate-level research university located on the shores of the Red Sea in Saudi Arabia
- The University’s new facilities, excellent faculty, state-of-art library and Shaheen II Supercomputer offer an ideal environment and resources for graduate level study and research

KAUST

KAUST

Students receive a KAUST fellowship that includes:

- full tuition
- competitive monthly living allowance
- private medical and dental coverage
- housing
- relocation support

KAUST