Lower-Bound Estimate for Cost-sensitive Decision Trees

Mikhail Goubko, ICS RAS, Moscow

Институт Проблем Управления РАН

Summary

1. New lower-bound estimate is suggested for the decision tree with case-dependent test costs
2. Unlike known estimates it performs well when the number of classes is small
3. Estimate calculation average performance is n^2m operations for n examples and m tests
4. Use the estimate to evaluate absolute losses of heuristic decision tree algorithms
5. Use the estimate in split criteria of greedy top-down algorithms of decision tree construction
6. Experiments on real data show our algorithms to give comparable results with popular heuristics
7. Algorithms suggested perform better on small data sets with lack of tests

Decision tree – a popular classification tool for:

- machine learning
- pattern recognition
- fault detection
- medical diagnostics
- situational control

- *Decision* is made from a series of *tests* of *attributes*
- The next attribute tested depends on the results of the previous tests
- Decision trees are learned from data
- Compact trees are good
- Expected length of the path in a tree is the most popular measure of tree size

Motivation

Growing an optimal decision tree is a discrete optimization problem. It is known to be NP-hard. Moreover, the size of an optimal tree is hard to approximate up to any constant factor.

For this reason numerous heuristic algorithms of finding near-optimal decision trees were suggested during several recent decades. Most of them employ greedy top-down tree induction.

Numerous experiments show good performance of these heuristics, but in any real situation the question remains open

how much extra cost is due to imperfectness of an algorithm?

Is it worth improving the adopted approach by looking for more sophisticated search techniques, or losses are already acceptable to stop?

The estimate of that sort is most interesting for the problems where test costs are measured in money units and are high enough.

As long as an exact optimal tree cost is hard to compute, it should be approximated from below to assure that "no more than X dollars can be economized by further improvement of a currently calculated decision tree".

In this paper a new lower-bound estimate for the expected classification cost of an optimal tree is suggested.

Estimates known from the literature have common limitations:

- too optimistic when the number of classes is small
- attributes' cardinality variations are not accounted

The model

Set of decisions (or classes) $D = \{1, \dots, d\}$

Set of attributes $M = \{1, \dots, m\}$

Only categorical attributes are considered

$k(q)$ is the cardinality of attribute q.

The learning set of examples (*cases*) N

Example $w \in N$ is a unique vector $(a_{wq})_{q \in M}$ of attribute values, and a class label $f(w) \in D$.

$\mu(w)$ – the probability, or frequency, of the example.

Every attribute q gives rise to the test or the *question* of the form "what is the value of attribute q?". Different answers partition the whole set of examples N into the sets $S_1(q), \dots, S_{k(q)}(q)$ (some sets may be empty).

In many applications tests differ in cost of measuring the value of the attribute. The expected cost of classification seems to be a natural optimization criterion in this framework given the decision tree correctly classifies examples available.

Test costs may depend on:

- an individual case;
- the true class of a case;
- side-effects;
- prior tests performed;
- prior test results,
- the correct answer of a current question.

We study, the first type, the case-dependent test costs of the form t_{qw} ($q \in M$, $w \in N$). They immediately cover the second and the third categories. The last two categories are reduced to case-dependent test costs by adding virtual tests that combine related questions.

Lower bound estimate definition

Definition 1. A subset of tests $Q \subseteq M$ *isolates* case w in subset of cases $S \subseteq N$ ($w \in S$) iff sequence of tests Q assures proper decision $f(w)$ given initial uncertainty S and w is the real state of the world.

Definition 2. *Optimal set of questions* $Q(w, S) \subseteq M$ is the cheapest of the sets of questions that isolate case w in S;

Define also *minimum cost* $t(w,S) := \sum_{w \in Q(w,S)} t_{wq}$

The lower-bound estimate

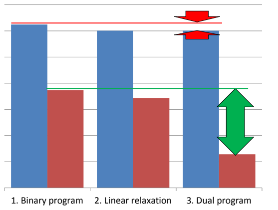
$$T_l(N) := \sum_{w \in N} \mu(w) \cdot t(w,N) = \sum_{w \in N} \mu(w) \sum_{q \in Q(w,N)} t_{qw}$$

The considered estimate is based on substituting the solution of the initial problem with the solution of a simpler problem. Imagine you know the true case w, but your colleague does not. You prove the true case is really w by suggesting him available tests from M. To achieve the goal at minimum cost you should choose the tests from $Q(w, N)$. Expected cost of proof then equals exactly $T_l(N)$.

Unlike known estimates the proposed estimate performs well when:

1) **the number of classes is small compared to that of examples**
2) **there is a small number of examples**

Calculation of the estimate is reduced to a number of set-covering problems and is NP-hard in the worst case.



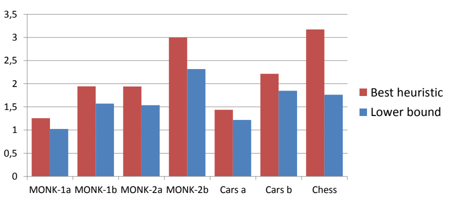
But experiments show nice performance for the estimate and its linear programming relaxation.

Average time ~ n^2m, (number of cases n, number of tests m)

Applications

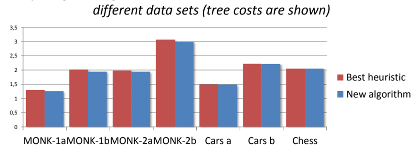
1. Use the lower-bound estimate to evaluate extra costs due to imperfectness of heuristic tree growing algorithms

The quality of the lower-bound estimate on real data sets



2. Use the estimate to build new tree growing algorithms

Comparing new algorithms with known heuristics (IDX, CS-ID3, EG2) on different data sets (tree costs are shown)



- **new algorithms perform better on small data sets**
- **new algorithms work worse in the presence of "dummy" tests**
- **adjacency of results shows that heuristic trees are nearly optimal**

A wider view: universal methods for hierarchy optimization

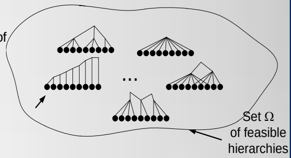
The decision tree problem considered above belongs to a wide range of problems of hierarchy optimization. Problems of this sort are met in very different areas – from computer science to management.

We suggest a general mathematical framework giving a common language to put the applied problems of hierarchy optimization, and providing the body of universal analytical and algorithmic methods for optimal hierarchy search.

In general, the problem is to find a hierarchy that minimizes a cost function defined on a set of feasible hierarchies.

Allowed
arbitrary number of layers
asymmetric hierarchies
multiple subordination
several top nodes

Not allowed
cycles
unconnected parts
subordinating to the "worker" nodes (black)



Cost function $c: \Omega \to \Re^+$

The problem is to find $H^* \in \operatorname{Arg\,min}_{H \in \Omega} c(H)$

The core of the framework is the concept of sectional cost functions. They are general enough to cover a wide range of applications and, at the same time, concise enough to allow for comprehensive deductions about the shape of an optimal hierarchy – when a hierarchy is tall or flat, is tree-shaped or looks as a conveyor belt. Also a number of general algorithms were developed for sectional cost functions. They help to seek an optimal hierarchy, an optimal tree, or an optimal conveyor-like hierarchy.

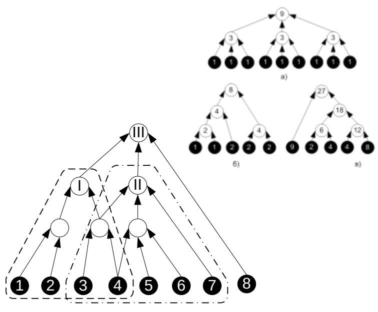
At the same time, a hierarchy optimization problem for a sectional cost function is usually hard to solve. Homogenous cost functions provide an example of an interesting subclass, which allows for a complete solution of an optimal hierarchy problem.

The optimal hierarchy is proved to be uniform, the closed-form solution is derived for an optimal hierarchy cost and its shape (a span of control and a skewness profile), and efficient algorithms were developed to construct nearly-optimal hierarchies.

Sectional cost function

1. Covers most of applied problems of hierarchy optimization
2. Analytical methods
   - when the tallest or the flattest hierarchy is optimal
   - when an optimal tree exists
   - when an optimal hierarchy has the shape of a conveyor
3. Algorithms
   - optimal hierarchy search
   - optimal tree search

   Build an optimal conveyor-belt hierarchy
4. The general problem of hierarchy optimization is complex



Homogenous cost function

1. Also has numerous applications
2. Closed-form solution for the optimal hierarchy problem
3. Efficient algorithms of nearly-optimal tree construction

The models of sectional cost functions and their subclasses were used to solve hierarchy optimization problems in many areas.

- **Manufacturing planning (assembly line balancing)**
- **Networks design**
  - communication and computing networks
  - data collection networks
  - structure of hierarchical cellular networks
- **Computational mathematics**
  - optimal coding
  - structure of algorithms
  - real-time computation and aggregation
  - hierarchical parallel computing
- **User interfaces design**
  - optimizing hierarchical menus
  - building compact and informative taxonomies
- **Data mining**
  - decision trees growing
  - structuring database indices
- **Organization design**
  - org. chart re-engineering
  - theoretical models of a hierarchical firm