Applications, Beauty and Challenges of Integer Programming for Decision Making and Data Mining

African Institute for Mathematical Sciences (Senegal)

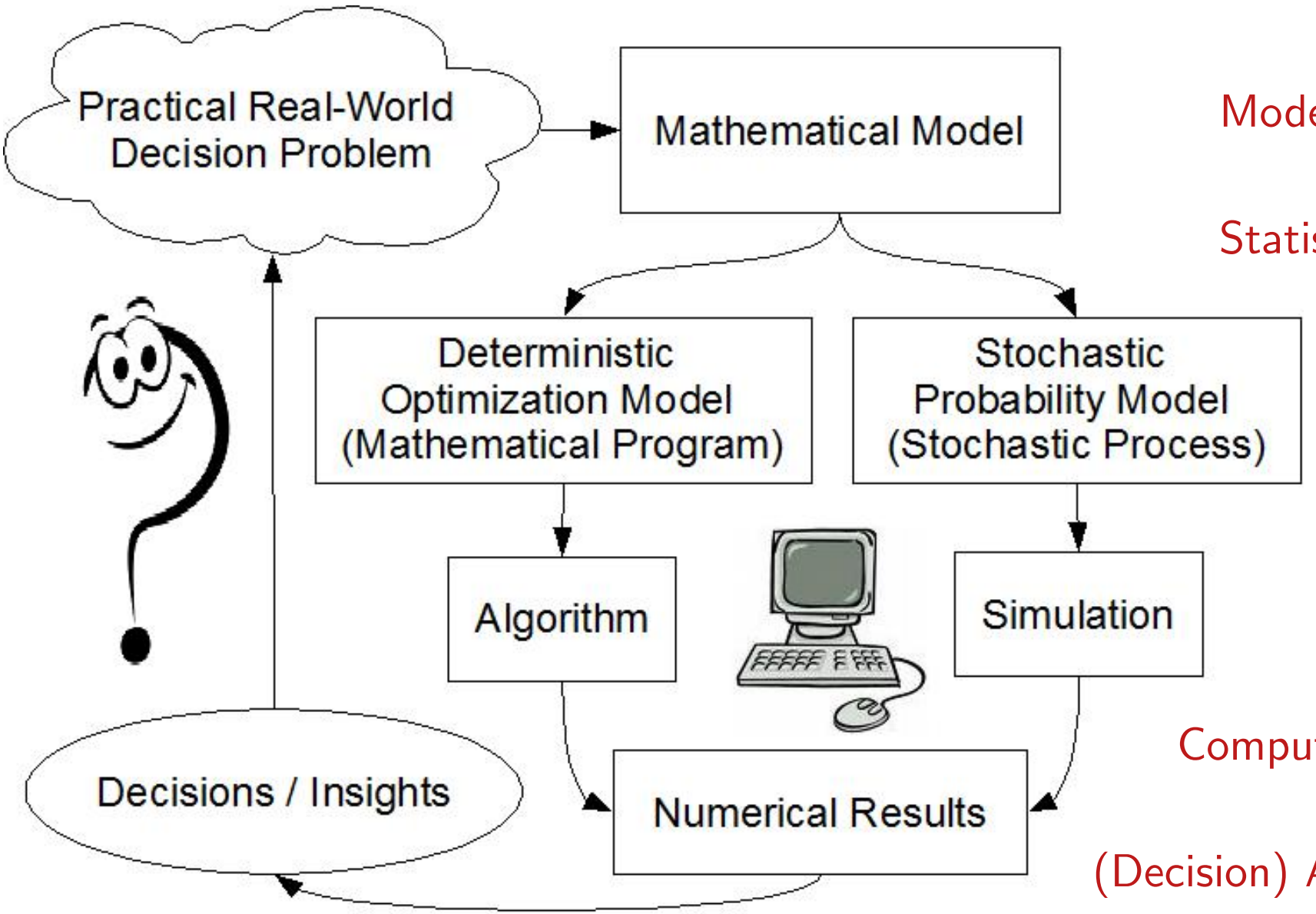
Alexander Engau, University of Colorado Denver (USA)
(on sabbatical leave at Dalhousie University, Canada)

AIMS | African Institute for Mathematical Sciences SENEGAL

Department of Mathematical & Statistical Sciences
UNIVERSITY OF COLORADO DENVER

DALHOUSIE UNIVERSITY
Inspiring Minds
Faculty of Management

Mathematical Problem Solving for Decision Making



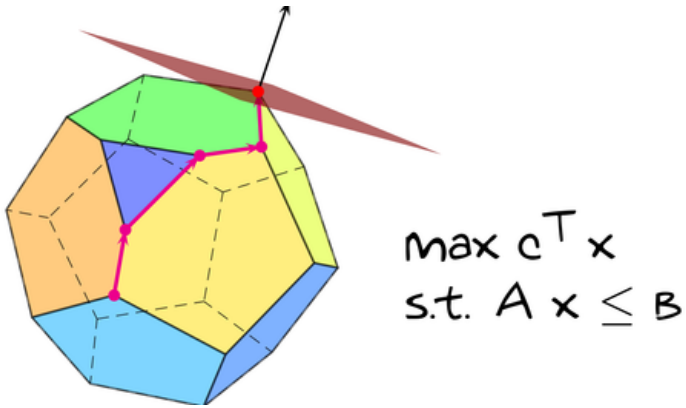
Linear and Nonlinear Programming for Optimization

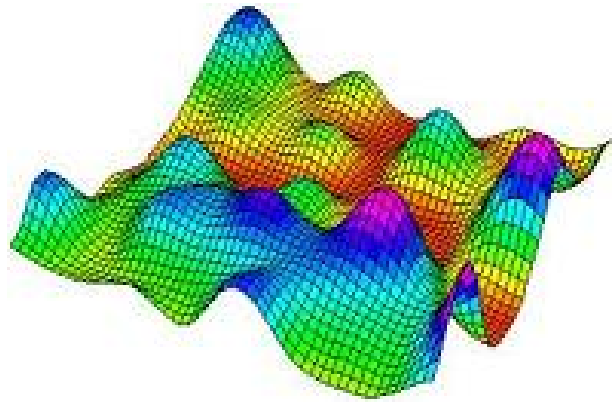
Linear Programming (LP):

$$\begin{aligned} \text{Minimize } & c^T x \\ \text{subject to } & A_{ub} x \leq b_{ub} \\ & A_{eq} x = b_{eq} \end{aligned}$$

Nonlinear Programming (NLP):

$$\begin{aligned} \text{Minimize } & f(x) \\ \text{subject to } & g_i(x) \geq 0, \quad i = 1, \ldots, m \\ & h_j(x) = 0, \quad j = 1, \ldots, p \end{aligned}$$



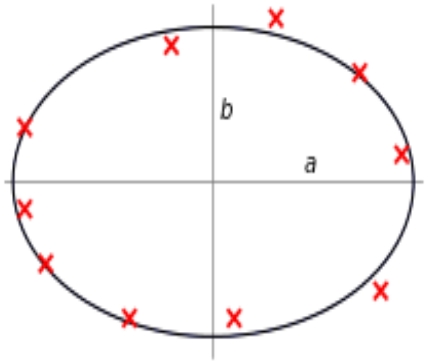


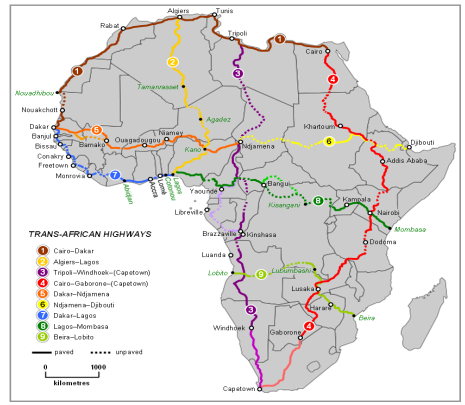
Linear and Nonlinear Optimization in Python (computer programming):

```
scipy.optimize.linprog(c, A_ub, b_ub, A_eq, b_eq, bounds)
scipy.optimize.minimize(f, x0, args, constraints, bounds)
```

Applications of Linear and Nonlinear Optimization

business, economics, engineering, finance, logistics, management, medicine
health, military, transportation, sustainability, environmental protection, ...





The Importance of Integer Solutions in Applications

Many decision quantities are integers:

- number of satellites in a satellite configuration
- number of workers or staff to employ in our own company
- number of bottles to produce, purchase or sell (or drink at a party)
- number of husbands or wives
- money (in cents)

Many decisions themselves are binary (yes or no):

- Do we repair an old satellite or replace?
- Do we invest into a stock or not?
- Do you want to marry me?

Very often, we can solve continuous problems and round. But not always!

Continuous vs. Discrete vs. Binary Knapsack Problem

$$\begin{array}{lrcrcrcl} \text{Maximize} & 3x_1 & + & 10x_2 & + & 12x_3 & & \\ \text{subject to} & x_1 & + & 2x_2 & + & 3x_3 & \leq & 5 \\ & x_1 & , & x_2 & , & x_3 & \in & \mathbb{X} \end{array}$$

- Continuous Knapsack Problem $\mathbb{X} = \mathbb{R}_+$

$$x^* = (0, 5/2, 0) \qquad f^* = c^T x^* = 25$$

- Discrete Knapsack Problem $\mathbb{X} = \mathbb{Z}_+$

$$x^* = (1, 2, 0) \qquad f^* = c^T x^* = 23$$

- Binary Knapsack Problem: $\mathbb{X} = \mathbb{B} = \{0, 1\}$

$$x^* = (0, 1, 1) \qquad f^* = c^T x^* = 22$$

Both the discrete and binary (0,1) knapsack problems are NP-complete!

The Beautiful Power of Binary Variables in Logic

One beauty of binary variables is the ability to model logical relationships:

$$x_i = \begin{cases} 1 & \text{if we select item } i \\ 0 & \text{otherwise.} \end{cases}$$

- We can select at most n items:

$$\sum_i x_i \leq n \qquad \text{(set packing)}$$

- We must select at least n items:

$$\sum_i x_i \geq n \qquad \text{(set covering)}$$

- We must select exactly n items:

$$\sum_i x_i = n \qquad \text{(partitioning)}$$

The Beautiful Power of Binary Variables in Logic (cont.)

- We must select either item i or item j, but not both.

$$x_i + x_j = 1 \qquad \text{(exclusive disjunction)}$$

- We must either select both items i and j, or none of them.

$$x_i = x_j \quad \Longleftrightarrow \quad x_i - x_j = 0 \qquad \text{(equivalence)}$$

- If we select item i, then we must also select item j.

$$x_i \leq x_j \quad \Longleftrightarrow \quad x_i - x_j \leq 0 \qquad \text{(implication)}$$

- If we select item i, then we must NOT select item j.

$$x_i + x_j \leq 1 \qquad \text{(same as set packing)}$$

- If we do not select item i, then we must select item j.

$$x_i + x_j \geq 1 \qquad \text{(same as set covering)}$$

The Beauty of Numbers and Addition Chains

Given a positive integer n, an addition chain is a set of positive integers

$$A(n) = \{a_1 = 1, a_2 = 2, \ldots, a_s = n\} \qquad \text{such that}$$
$$(\forall\ k = 1, 2, \ldots, s)\ (\exists\ 1 \leq i \leq j < k)\colon\ a_k = a_i + a_j.$$

Shortest Addition Chain (SAC) Problem: Find $A(N)$ of shortest length.

- $A_{binary}(15) = \{1, 2, 3, 4, 7, 8, 15\}$ has length 6 (cardinality 7)
- $A_{amadou}(15) = \{1, 2, 4, 5, 10, 15\}$ has length 5 (this is optimal !!)
- $A_{alex}(15) = \{a_1 = 1, a_2 = 2, a_3 = 3, a_4 = 4, a_5 = 5, \ldots, a_{15} = 15\}$??

  Let $x_k = \begin{cases} 1 & \text{if we need } a_k \\ 0 & \text{otherwise} \end{cases} \quad \Rightarrow \quad$ solve a binary integer program $\ddot{\smile}$

- $x_1 = x_2 = x_3 = x_6 = x_{12} = x_{15} = 1 \Rightarrow A_{bip}(15) = \{1, 2, 3, 6, 12, 15\}$!!

Integer Programming Model for the SAC Problem

Given an integer n, let $a_k = k$ for $k = 1, 2, \ldots, n$ and define the variables

$$x_k = \begin{cases} 1 & \text{if we use the term } a_k \text{ in the addition chain} \\ 0 & \text{otherwise.} \end{cases}$$

- Objective:

$$\text{minimize} \quad \sum_{k=1}^{n} x_k$$

- Easy Constraints:

$$x_1 = x_2 = x_n = 1$$

- Simplified Objective:

$$\text{minimize} \quad \sum_{k=1}^{n} x_k = 3 + \sum_{k=3}^{n-1} x_k$$

Difficult Constraints for the SAC Problem

- If $x_k = 1$ we need $x_i = 1$ and $x_j = 1$ such that $a_k = a_i + a_j$ so define

$$y_{ki} = \begin{cases} 1 & \text{if we use } a_i \text{ as the first term in the sum of } a_k \\ 0 & \text{otherwise;} \end{cases} \qquad (i < k)$$

$$z_{kj} = \begin{cases} 1 & \text{if we use } a_j \text{ as the second term in the sum } a_k \\ 0 & \text{otherwise.} \end{cases} \qquad (j < k)$$

- Additional Constraints:

$$x_k a_k = \sum_{i<k} y_{ki} a_i + \sum_{j<k} z_{kj} a_j \qquad (k > 2)$$

$$x_k = \sum_{i<k} y_{ki} \quad x_k = \sum_{j<k} z_{kj} \qquad (k > 2)$$

$$x_k \geq y_{ki} \quad x_k \geq z_{kj} \qquad (k > 2)$$

Final Model Formulation for the SAC Problem

$$\text{Minimize} \quad \sum_{k=3}^{n-1} x_k$$

$$\text{subject to} \quad x_k a_k = \sum_{i<k} y_{ki} a_i + \sum_{j<k} z_{kj} a_j \qquad k = 3, 4, \ldots, n$$

$$x_k = \sum_{i<k} y_{ki} \qquad k = 3, 4, \ldots, n$$

$$x_k = \sum_{j<k} z_{kj} \qquad k = 3, 4, \ldots, n$$

$$x_k \geq y_{ki} \qquad k = 3, 4, \ldots, n; \;\; i < k$$

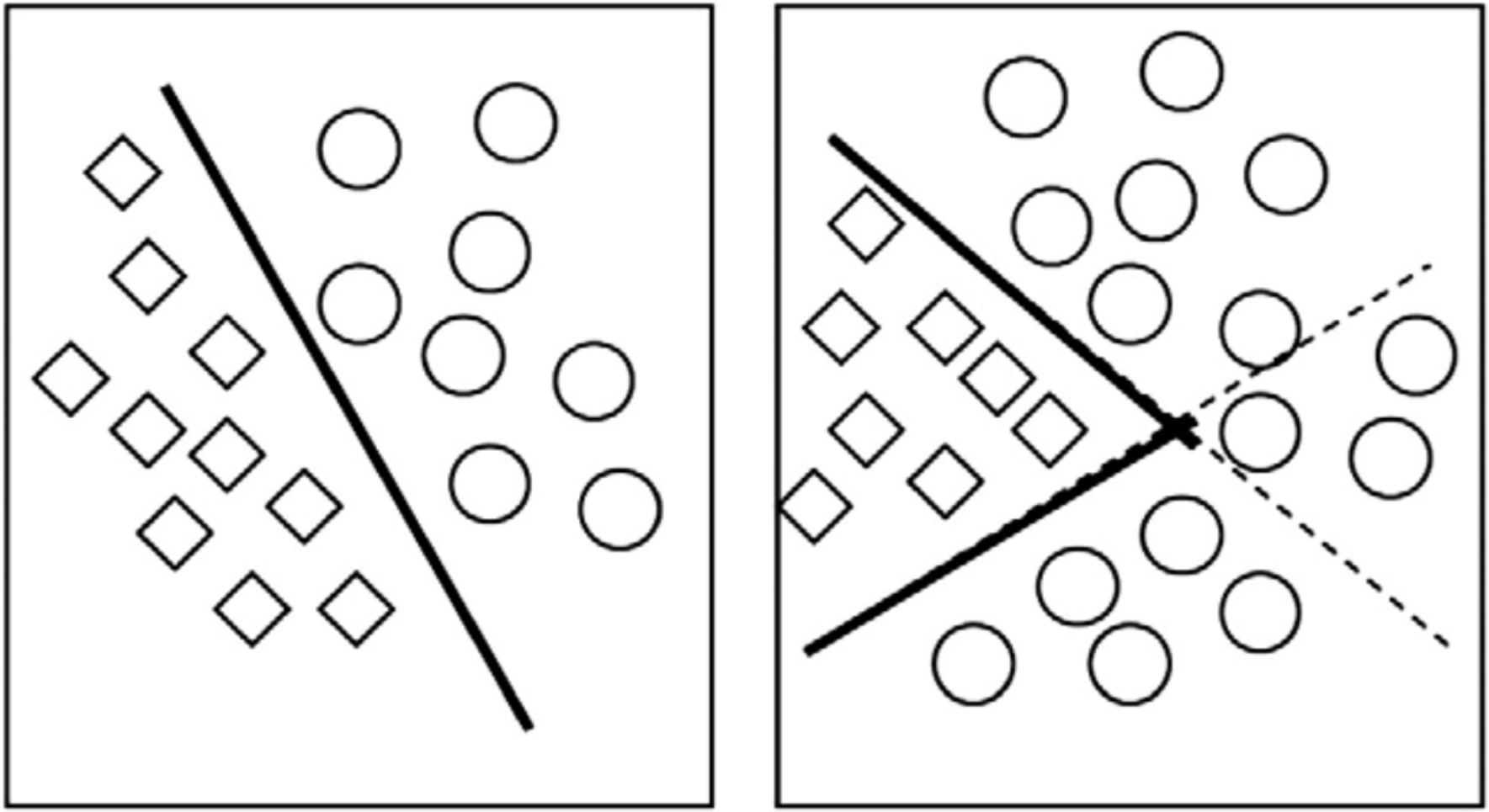
$$x_k \geq z_{kj} \qquad k = 3, 4, \ldots, n; \;\; j < k$$

$$x_1 = x_2 = x_n = 1$$

$$x_k \in \{0, 1\}, y_{ki} \in \{0, 1\}, z_{kj} \in \{0, 1\}$$

Discriminant Analysis: Separating Hyperplane Approaches

We would like to separate the round elements from the diamonds elements.



M. Better et al. / Decision Support Systems 48 (2010) 430-436, Figure 1

Data Classification: Single Hyperplane Case

Given: two-partitioned data set $G = G_1 \cup G_2$ with $m = m_1 + m_2$ elements.

- Each element $i \in G$ is characterized by n attributes indexed by j:

$$x = (x_{i1}, x_{i2}, \ldots, x_{in}).$$

- Then we can write the classification rule as two linear inequalities:

$$\sum_{j=1}^{n} a_j x_{ij} \geq b + \varepsilon \qquad i \in G_1$$

$$\sum_{j=1}^{n} a_j x_{ij} \leq b - \varepsilon \qquad i \in G_2$$

where the parameter $\varepsilon > 0$ is a small positive constant (separation zone).

Classification Accuracy Model

- Use $c_i \in \{0, 1\}$ whether observation i is classified correctly or not.
- Find a_j and b to minimize the number of classifications with error.

$$\begin{aligned}
\text{Minimize} \quad & \sum_{i=1}^{m} c_i \\
\text{subject to} \quad & \sum_{j=1}^{n} a_j x_{ij} + M c_i \geq b + \varepsilon && i \in G_1; \\
& \sum_{j=1}^{n} a_j x_{ij} - M c_i \leq b - \varepsilon && i \in G_2; \\
& c_i \in \{0, 1\} && i = 1, 2, \ldots, m; \\
& a_j, b \text{ free} && j = 1, 2, \ldots, n,
\end{aligned}$$

where $M > 0$ is a large constant that allows for errors ("big M" method).

Model Extension for Multiple Hyperplane Case

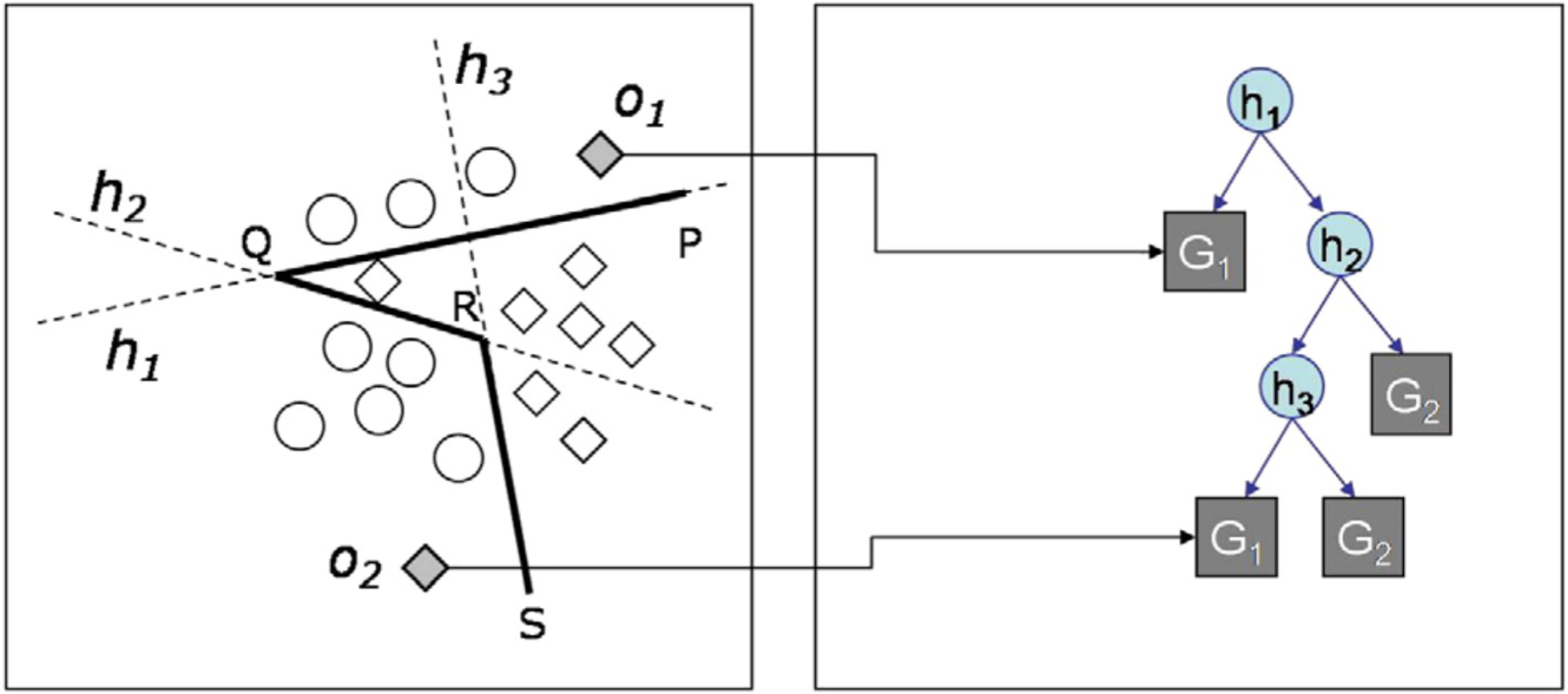
- Use an additional set of binary variables $c_{ki} \in \{0, 1\}$ only for $i \in G_2$ to indicate whether observation i is classified correctly by hyperplane k.
- A single correct classification $c_{ki} = 0$ suffices to set $c_i = 0$ for $i \in G_2$.

$$\text{Minimize} \quad \sum_{i=1}^{m} c_i$$

$$\begin{aligned}
\text{subject to} \quad & \sum_{j=1}^{n} a_{kj} x_{ij} + M c_i \geq b_k + \varepsilon && k \in H, i \in G_1; \\
& \sum_{j=1}^{n} a_{kj} x_{ij} - M c_{ki} \leq b_k - \varepsilon && k \in H, i \in G_2; \\
& \sum_{i=1}^{m} (1 - c_{ki}) \geq 1 - c_i && k \in H, i \in G_2; \\
& c_{ki} \in \{0, 1\} && k \in H, i \in G_2; \\
& c_i \in \{0, 1\} && i = 1, 2, \ldots, m; \\
& a_{kj}, b_k \text{ free} && k \in H, j = 1, 2, \ldots, n.
\end{aligned}$$

Vertical Decision Tree (VDT) with depth $d = 3$

Circles belong to the group G_1 and diamonds belong to the group to G_2.



M. Better et al. / Decision Support Systems 48 (2010) 430-436, Figure 2

VDT Full Model Formulation of depth $d = p$

$$\text{Minimize} \quad \sum_{i=1}^{m} c_{pi}$$

$$\text{subject to} \quad \sum_{j=1}^{n} a_{kj}x_{ij} + M\left(c_{ki} + \sum_{l=1}^{k-1} f_{li}\right) \geq b_k + \varepsilon \qquad k \in H, i \in G_1;$$

$$\sum_{j=1}^{n} a_{kj}x_{ij} - M\left(c_{ki} + \sum_{l=1}^{k-1} f_{li}\right) \leq b_k - \varepsilon \qquad k \in H, i \in G_2;$$

$$c_{ki} \leq g_k \qquad k \in H_0, i \in G_1;$$

$$c_{ki} \leq 1 - g_k \qquad k \in H_0, i \in G_2;$$

$$f_{ki} \leq g_k \qquad k \in H_0, i \in G_1;$$

$$f_{ki} \leq 1 - g_k \qquad k \in H_0, i \in G_2;$$

$$c_{ki} + f_{ki} \leq 1 \qquad k \in H_0, i \in G;$$

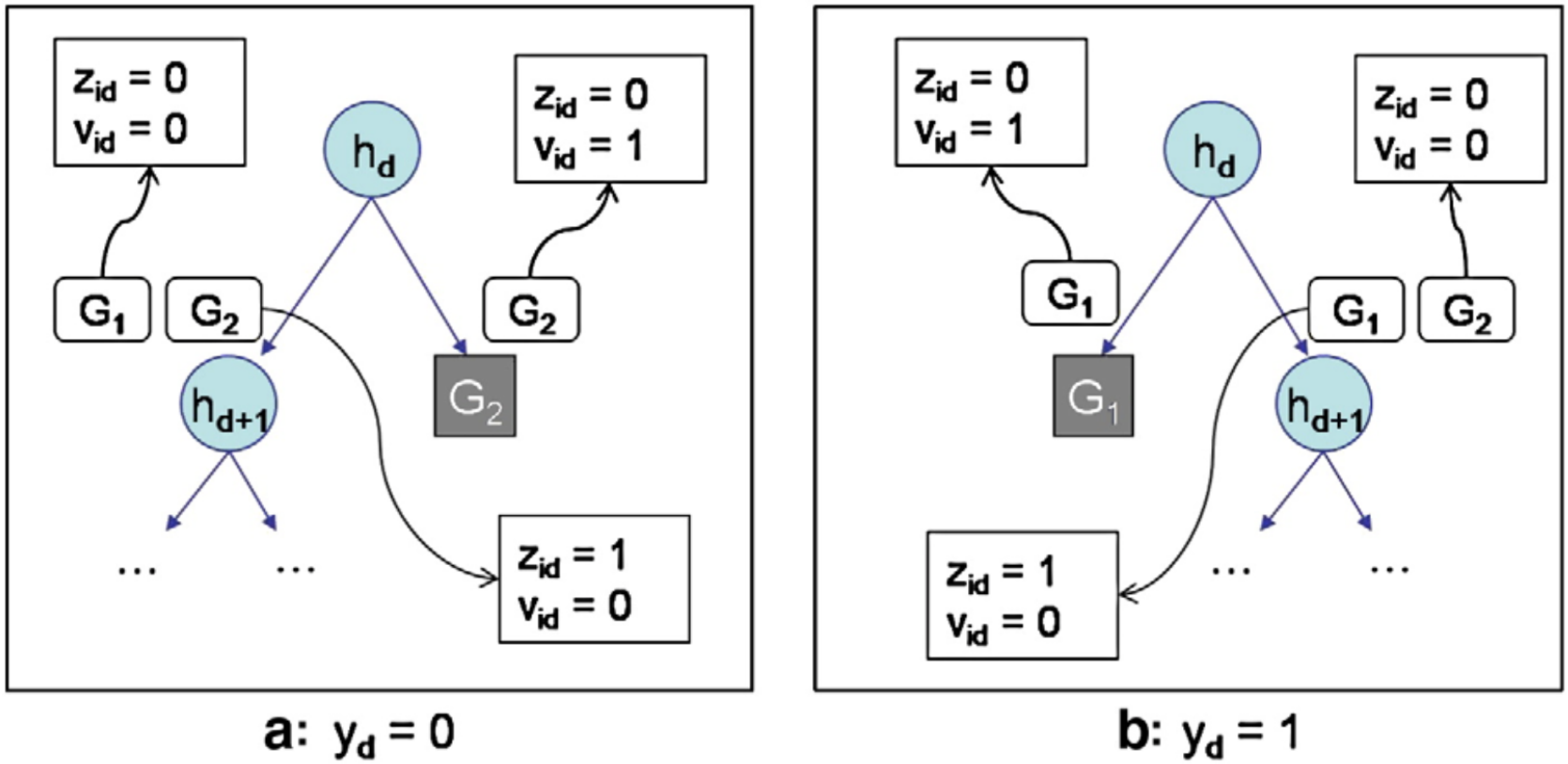
$$g_k, f_{ki} \in \{0, 1\} \qquad k \in H_0; i \in G;$$

$$c_{ki} \in \{0, 1\} \qquad k \in H, i \in G;$$

$$a_{kj}, b_k \text{ free} \qquad k \in H, j = 1, \ldots, n.$$

Relation Among Variables g_k (y_d), c_{ki} (z_{id}) and f_{ki} (v_{id})

If $y_d = 0$ (a), the tree grows to the left and the objects of G_2 that fall on the right (into a leaf associated with G_2) are correctly classified by the tree at depth d ($z_{id} = 0, v_{id} = 1$). Conversely, objects of both groups may fall on the left where those of G_1 have $z_{id} = 0$, $v_{id} = 0$, and those of G_2 have $z_{id} = 1$, $v_{id} = 0$. If $y_d = 1$ (b), the situation is symmetric.



M. Better et al. / Decision Support Systems 48 (2010) 430-436, Figure 4

学而不厌诲人不倦 孔子