

Scoring Rules

Cols Ordering

$$\text{Score}_i = 100000 \cdot P + 1 \cdot \sum (\log(1 + |\delta_j|)) + 100 \cdot \log(1 + |\text{Obj}_i|) + 1 \cdot \#occurrences$$

Components

1. Type Priority (P):

Assign priorities to variable types to reflect their importance in the model:

- **Binary Variables** ($VType = \text{Binary}$): $P = 3$
- **Integer Variables** ($VType = \text{Integer}$): $P = 2$
- **Continuous Variables** ($VType = \text{Continuous}$): $P = 1$

The multiplier of **100,000** ensures that the variable type dominates the scoring, making binary variables the highest priority, followed by integer and continuous variables.

2. Sum of Coefficients ($\sum \log(1 + |\delta_j|)$):

This term accounts for the influence of variable i across all constraints. The sum of logarithms of the absolute constraint coefficients (δ_j) reflects how actively the variable participates in the model. A larger sum indicates that the variable appears in more constraints or with stronger coefficients, increasing its overall importance. The multiplier of **1** ensures this influence is considered without overpowering type priority or objective contributions.

3. Objective Coefficient ($|\text{Obj}_i|$):

Variables with larger absolute contributions to the objective function receive higher scores. This reflects their impact on the optimization goal, with the logarithm ensuring numerical stability over a wide range of values. The multiplier of **100** amplifies the influence of significant objective coefficients while maintaining balance with other components.

4. Occurrences ($\#occurrences$):

This term counts the number of constraints in which the variable appears (i.e., the number of non-zero entries in its column in the constraint matrix). Variables that appear in more constraints are likely to play a more central role in the model. The multiplier of **1** gives a moderate influence to this term, complementing the sum of coefficients.

Rows Ordering

$$\text{Score}_j = 100000 \cdot P + 1 \cdot \sum (\log(1 + |\gamma_j|)) + 100 \cdot \log(1 + |\text{RHS}_j|) + 1 \cdot \log(1 + |\text{Range}_j|)$$

Components

1. **Type Priority (P):**

Assign priorities based on the constraint type to reflect their structural importance:

- **Greater-than Constraints ($>$):** $P = 3$
- **Equality Constraints ($=$):** $P = 2$
- **Less-than Constraints ($<$):** $P = 1$

The large multiplier of **100,000** ensures that constraint type strongly influences the ordering, giving higher priority to constraints with stricter forms like $>$ or $=$.

2. **Sum of Coefficients ($\sum \log(1 + |\gamma_j|)$):**

This term measures the cumulative impact of all variables involved in constraint j . By summing the logarithms of the absolute values of the coefficients (γ_j), the score reflects both the strength and the number of variables participating in the constraint. A higher sum indicates a more "influential" constraint in shaping the feasible region. The multiplier of **1** ensures a balanced contribution relative to other terms.

3. **RHS ($|\text{RHS}_j|$):**

Constraints with larger absolute right-hand side values receive higher scores. This reflects their potential to impose stronger restrictions on feasible solutions. The logarithm helps handle large ranges of RHS values, and the multiplier of **100** ensures these constraints are given appropriate weight in the scoring.

4. **Range ($\log(1 + |\text{Range}_j|)$):**

This term accounts for the numerical range of the constraint, defined as the difference between its upper and lower bounds (or, in the case of equality constraints, effectively zero). Constraints with a larger range are less restrictive, while those with a narrow range are tighter and potentially more critical. The logarithmic transformation stabilizes the influence of large ranges, and the multiplier of **1** provides moderate weight in the overall score.

This expanded explanation provides a clear rationale for each term, highlighting how they contribute to the scoring while maintaining numerical stability and meaningful differentiation.

Term	Min Value	Max Value	Multiplier
Type Priority (Col)	100,000	300,000	100,000
Sum of Coefficients (Col)	0.0	27,6310.21	1
Objective Coefficient (Col)	0.0001	2,763.10	100

Term	Min Value	Max Value	Multiplier
Occurrences (Col)	0.0	100.0	1

Table 1: Minimum and maximum values for each term in the column scoring rules, along with their corresponding multipliers.

Term	Min Value	Max Value	Multiplier
Type Priority (Row)	100,000	300,000	100,000
Sum of Coefficients (Row)	0.0	27,6310.21	1
RHS (Row)	0.0001	2,763.10	100
Range (Row)	0.0	27.63	1

Table 2: Minimum and maximum values for each term in the row scoring rules, along with their corresponding multipliers.

The scoring rules for both variables (columns) and constraints (rows) are designed to balance contributions from different structural features of the optimization problem. The **Type Priority** term remains dominant with values ranging from 10,000 to 30,000, ensuring that variable and constraint types play a primary role in ordering decisions. The **Objective Coefficient** and **RHS** terms are scaled logarithmically to handle wide ranges of values, with maximum contributions around 2,763. The **Sum of Coefficients** term can reach up to 276,310 when aggregated across many constraints, emphasizing variables heavily involved in the model. The **Occurrences** term has been scaled down (divided by 10) to prevent it from overshadowing more critical factors, while the **Range** term for constraints adds minor adjustments, capped at approximately 27.63. This configuration maintains numerical stability while allowing meaningful differentiation in ordering.

Use Cases

Because the approach focuses on relative differences and the structure of constraints rather than on absolute coefficient sizes, it tends to suit problems where the decision variable roles are defined by their interaction with many constraints and where relative cost or benefit differences are crucial. Examples include:

- **Assignment and Matching Problems:**
In these problems, the decision to assign a task (or match an element) is driven by cost or profit

differences. Here, a variable's importance comes more from how its objective coefficient compares to others rather than its absolute value.

- **Knapsack and Set Covering/Packing Problems:**

For knapsack problems, it's the profit-to-weight ratio (a relative measure) that drives the optimal selection, not the raw profit numbers. Similarly, in set covering/packing, the "importance" of a column (or variable) is determined by how frequently it appears and how its cost compares with others.

- **Network Design and Facility Location:**

In these models, although the absolute sizes of capacities and costs can vary widely, it's the relative differences (e.g., the ratio of capacity to demand, or the cost difference between opening and not opening a facility) that typically determine which variables are most critical.

- **Production Planning/Scheduling Problems:**

Such problems often have constraints with widely varying RHS values. Your row score, which includes $\log(1+|\text{RHS}|)$ and additional measures for coefficient variation (the γ and Range terms), would capture the "tightness" or variability of these constraints—a factor that is often more important than the absolute magnitudes.