

Scoring Rules

Column (Variable) Ordering

$$\text{Score}_i = 1,000,000 \cdot P + 1 \cdot \sum_j (\log(1 + |\delta_{ij}|)) + 100 \cdot \log(1 + |\text{Obj}_i|) + 1 \cdot \#\text{occurrences}$$

where:

1. P = Type Priority

- **Binary Variables:** $P = 3$
- **Integer Variables:** $P = 2$
- **Continuous Variables:** $P = 1$

The multiplier of 1,000,000 ensures that **variable type** is the dominant factor. Thus, **all** binary variables will appear before **all** integer variables, which in turn appear before **all** continuous variables in a descending sort.

2. $\sum \log(1 + |\delta_{ij}|)$

This term measures the total “strength” of variable i across all constraints. For each constraint coefficient δ_{ij} (the coefficient of variable i in constraint j), we take $\log(1 + |\delta_{ij}|)$ and sum over all j . A larger sum indicates either more constraints in which this variable appears or larger coefficients, implying a stronger role in shaping feasibility. Its **multiplier is 1**, so it is important but *far smaller* than the type-priority block factor.

3. $\log(1 + |\text{Obj}_i|)$

This highlights how much each variable contributes to the objective function. We take the log of the absolute objective coefficient to avoid blowing up large values but still reflect their impact. The multiplier of **100** ensures that variables with large objective contributions get a noticeable bump in ordering—but still well below the $10^6 \cdot P$ term that separates types.

4. $\#\text{occurrences}$

This is the **number of constraints** in which variable i is *non-zero*. Variables appearing in many constraints can be crucial because they interact with more parts of the model. We multiply this by **1** for moderate influence.

Row (Constraint) Ordering

$$\text{Score}_j = 1,000,000 \cdot P + 1 \cdot \sum_i (\log(1 + |\gamma_{ji}|)) + 100 \cdot \log(1 + |\text{RHS}_j|) + 1 \cdot \log(1 + \text{Range}_j)$$

where:

1. P = Constraint-Sense Priority

- “>” Constraints: $P = 3$
- “=” Constraints: $P = 2$
- “<” Constraints: $P = 1$

Multiplying P by 1,000,000 forces constraints of sense “>” to come before those of sense “=” and those before “<” in a descending sort.

2. $\sum \log(1 + |\gamma_{ji}|)$

Similar to the variable side, this sums the logarithms of the absolute values of all coefficients in constraint j . A higher sum indicates the constraint involves more variables or larger coefficients, potentially making it more “influential” in shaping the feasible region. Its multiplier of **1** keeps it subordinate to P .

3. $\log(1 + |\text{RHS}_j|)$

Constraints with larger right-hand side values may allow bigger feasible regions or, conversely, impose tighter restrictions if negative. We multiply by **100** to ensure noticeable effect.

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

The “range” of a constraint (e.g., distance between lower and upper bounds in a ranged constraint) indicates how flexible it is. A narrow range suggests a tighter constraint. The multiplier of **1** gives a moderate effect.

Why a 10^6 Factor for Type (or Sense) Priority?

Under a **hierarchical** ordering approach, you want **all items in block A** to rank higher than **all items in block B**—regardless of their intra-block scores. Multiplying P by a large constant (e.g., 10^6) achieves exactly that. Even the maximum contributions of the other terms (which might go into the thousands or tens of thousands) cannot outweigh a difference of 10^6 in the final score.

Hence, in a descending sort:

1. **Binary** variables (with $P = 3$) end up with at least 3×10^6 points, whereas
2. **Integer** variables (with $P = 2$) have at most $2 \times 10^6 + (\text{thousands})$.

So all binary variables will come before all integers, and all integers before all continuous variables.

Revised Tables

Term	Typical Min	Typical Max	Multiplier
Type Priority (Var)	1,000,000	3,000,000	1,000,000
Sum of Coeffs (Var)	0	$\sim 276,310$	1

Term	Typical Min	Typical Max	Multiplier
Objective Coefficient (Var)	0.0	~ 2,763	100
Occurrences (Var)	0	100+	1

Table 1: Typical ranges for each column (variable) scoring component, using an updated multiplier of 1,000,000 for type priority.

Term	Typical Min	Typical Max	Multiplier
Sense Priority (Constraint)	1,000,000	3,000,000	1,000,000
Sum of Coeffs (Constraint)	0	~ 276,310	1
RHS (Constraint)	0.0	~ 2,763	100
Range (Constraint)	0	~ 27.63	1

Table 2: Typical ranges for each row (constraint) scoring component, with the sense priority multiplied by 1,000,000.

Putting It All Together

- **Block-Level Priority (Type or Sense):** Scales to 10^6 to enforce a strict hierarchy among different *blocks* (e.g., binary > integer > continuous, or “>” > “=” > “<”).
- **Intra-Block Terms:** Coefficients, objective/RHS, occurrences, and range all add finer distinctions *within* each block. They are multiplied by much smaller numbers (1 or 100), ensuring they **never** override the block-level separation.

In a **descending** sort of Score_i , the hierarchy is:

1. **Binary** variables (or “>” constraints) first, sorted by sum-of-coeffs / objective / etc.
2. **Integer** variables (or “=” constraints) next, sorted by their internal scores.
3. **Continuous** variables (or “<” constraints) last, again ordered by their intra-block contributions.

Because the largest multiplier is on **block** type, you get a **two-level ordering**:

1. **Which block are you in?** (Major level)
2. **Within that block, what are your intra-block characteristics?** (Minor level)