Lesson 9. Logical Constraints

1 Today...

- We introduce logical (either/or) constraints using binary decision variables. I.e., we either enforce or relax a constraint based on the value of a $\{0, 1\}$ variable.

2 Resource Allocation Revisited

Quality Cabinets used an IP to determine how many of each kind of cabinet to make each week in order to maximize profit. They used decision variables x_a, x_d, and x_e to represent the number of standard, deluxe, and enhanced cabinets, respectively, to produce each week. Each cabinet requires a certain number of hours of painting time, and there is a limit on the number of painting hours available. A small part of the model is...

Maximize	$25x_a + 45x_d + 60x_e$	
subject to	$2x_a + 4x_d + 5x_e \leq 700$	(painting time)

(1) Explain the objective function. What does the "25" represent?

It's the total profit from making cabinets.
QC makes \$25 of profit for every standard cabinet made and sold.

(2) Explain the painting time constraint. What does the "2" represent? The "700"?

Cannot use more than 700 total hours to paint all the cabinets. | It takes 2 hours to paint a standard cabinet.

3 Model Update Requested

Now Quality Cabinets is considering renting better painting equipment and has asked us to update our model to help with this decision. The equipment will cost \$300 per week, but will reduce the time required to to do the painting by 15 minutes for a standard cabinet, by 30 minutes for a deluxe cabinet, and by 1 hour for an enhanced cabinet.

We decide to add a binary variable z. In the solution, if $z = 1$, then Quality Cabinets rents the equipment. If $z = 0$, they do not.

(3) Should the objective function change? If not, explain. If so, write the updated objective function.

Yes, you will lose \$300 if QC rents the equipment.

$25x_a + 45x_d + 60x_e - 300z$

(4) If the equipment is not rented, how does the painting constraint change? Label it (A).

STAYS THE SAME

(5) If the equipment is rented, how does the painting constraint change? Label it (B).

$1.75x_a + 3.5x_d + 4x_e \leq 700$

4 Logical (Either/Or) Constraints

There is no place for "if – then" statements in a math programming model, so we have to enforce this logic indirectly using linear constraints and binary variables.

A constraint is *relaxed* if it has no impact. *You can think of this as "practically" removing a constraint without actually doing so.* We can relax a $\leq$-constraint by making the value on the right so large that it will never restrict the values of the variables on the left.

(6) Rewrite constraint (A) above *using* z so that it is enforced if $z = 0$ (the equipment is not rented), but *relaxed* if $z = 1$ (the equipment is rented).

$2x_a + 4x_d + 5x_e \leq 700 + Mz$

(7) Rewrite constraint (B) above *using* z so that it is enforced if $z = 1$ (the equipment is rented), but *relaxed* if $z = 0$ (the equipment is not rented).

$2x_a + 4x_d + 5x_e \leq 700 + M(1-z)$

(8) Write the updated version of the partial Quality Cabinets model here.

$$\text{Max } 25x_a + 45x_d + 60x_e - 300z$$

$$2x_a + 4x_d + 5x_e \leq 700 + Mz$$

$$x_a, x_d, x_e \in \mathbb{Z}^+ \quad z \in \{0,1\}$$

5 Logical Constraints Summary

Suppose $ax \leq b$ is a linear constraint, M is a number that is bigger than ax could ever be (but not TOO big!–choose M wisely), and z is a binary variable.

A constraint that is enforces $ax \leq b$ if $z = 0$ and relaxes $ax \leq b$ if $z = 1$:

$$ax \leq b + Mz$$

A constraint that is enforces $ax \leq b$ if $z = 1$ and relaxes $ax \leq b$ if $z = 0$:

$$ax \leq b + M(1-z)$$

6 Many more possibilities

There are many ways to creatively use linear constraints to enforce modeling requirements; this is just one such example. Often this process takes some trial and error. **Always be sure to test the logic with various values of the decision variables to make sure the constraint is doing what you want it to do by comparing the value of the function on the left-hand-side with the value of the function on the right-hand-side.**