Project 3 writeup

The bulk of this assignment was designing a linear programming model.

Decision variables:
We had a single 3D matrix of size (num facilities x max vehicles per facility x num customers) that was used for all of our constraints. Each entry in the matrix was between 0 and 1 (or exactly either 0 or 1 for the integer programming problem). A 1 at indices [i,j,k] meant that customer k was being fulfilled by the jth vehicle of facility i.

We also created two additional "helper" variable matrices with entries between 0 and 1:

1. *Is_facility_open*: 1D array of size (num_facilities). 1 at entry i means that the ith facility is open.
2. *Is_facility_vehicle_used*: 2D array of size (num_facilities x max vehicles per facility). 1 at entry [i,j] means that the ith facility is using their jth vehicle. Intuitively, this can be thought of as a 1D array tracking whether or not each vehicle is being used - we just store it as a 2D matrix so we can index vehicles by which facility they belong to.

Constraints:
We had two types of constraints, problem constraints and consistency constraints. Our problem constraints were:

1. Make sure each customer is served by one facility: We implemented this as "make sure each customer is served by *at least* one facility" because that's the standard form for LP constraints. This was fine because the objective function penalizes having more facilities open.
2. Make sure each facility has a capacity large enough to fulfill the demand of the customers they're serving
3. Make sure each vehicle doesn't drive too far

Consistency constraints ensured that the helper variables *is_facility_open* and *is_facility_vehicle_used* were consistent with the 3D matrix. Specifically, we required for all facility, vehicle, and customer:

1. *Is_facility_open*[facility] >= 3D matrix[facility][vehicle][customer]
2. *Is_facility_vehicle_used*[i][j] >= 3D matrix[facility][vehicle][customer]

Functionally, this ensures that *is_facility_open*[i] = 1 if any goods are being delivered from that facility and that *is_facility_vehicle_used*[i][j] = 1 if that vehicle is being used to deliver any goods.

Objective function
The objective function we were minimizing was a sum of three different values:

1. Cost of opening facilities: here we used *is_facility_open*
2. Fixed cost of using vehicles: here we used *is_facility_vehicle_used* is helpful.
3. Allocation cost of delivering goods: we used the 3D matrix as a facility -> customer mapping

Team members: Kshitij Sachan (ksachan), Aaron Wang (awang167)
Leaderboard names: newport_narwhals, i
Hours spent on project: 10