**Motivation**

Physicians are regularly assigned to many inpatient units on the same day. These units are geographically far apart (and far from the physician offices), adding a lot of travel time. Plus, it makes it harder for physicians to attend daily huddles and otherwise communicate with nurses and other care staff. This impacts patient quality of care as well as length of stay.

**Idea**

What if we were able to reduce the number of units each physician service was assigned to? This would drastically improve physician productivity and improve care team communication. It would be reasonable to expect a positive impact on patient care and length of stay / occupancy.

In fact, we can do this! By using a Mixed Integer Program, we can create an algorithm that optimally assigns physician services to inpatient units.

**Considerations / Constraints**

Some units are not suitable for some services. Other units are acceptable but not ideal for those same services. This needs to be accounted for. We need to still assign services to appropriate units.

We cannot assign more patients to a unit than the unit has available staffed beds.

We cannot assign less patients from a service than the service needs to care for.

Assigning a service to less units is very desirable. Physicians would rather be assigned to fewer, slightly less desirable units than being assigned to more, more desirable units. I have confirmed this with a handful of physicians, but it would be worth confirming this further.

**The Model, in English**

Each service-unit combination gets a desirability score (with low scores meaning more desirable). We will call it DS(s,u), with service s and unit u. Let A(s,u) be the number of patient-days of service s that are assigned to unit u. We want to minimize DS(s,u) \* A(s,u) across all services and units.

Additionally, there is a penalty for each service-unit combination that is used. Let U(s,u) designate whether a service-unit combination is used. U(s,u) = 1 if the combination of s & u is used, and = 0 if not. Let P be a constant number that sets the size of the penalty. The penalty should a much larger than the scale of DS. For instance, if DS ranges from 0-10, P might be selected as 1,000 or even higher.

So, we want to minimize the sum of DS(s,u) \* A(s,u) + P \* U(s,u) over all services s and units u.

However, we cannot assign more patients to a unit than the unit has staffed beds. Let C(u) be the capacity of unit u. The sum of A(s,u) over all services s cannot be larger than C(u) for any unit u.

Also, we must meet all demand for each service. Let D(s) be the demand, in patient-days, for service s. The sum of A(s,u) over all units u must be at least as large as D(s) for all services s.

**The Model**

**More Details**

How do you define capacity? How do you define demand? Whatever you do, they had better be defined in compatible ways. For the initial pass, I used average weekly census (based on hourly snapshots) for demand and used staffed beds for capacity. But will this adequately meet peak time demands? I tried using the max hourly census for each day, and averaging that, but that resulted in an unsolvable problem, as the demand was too high for the capacity. But perhaps using the average noon census would be a good middle ground?

How do you define desirability? There is talk of surveying physicians and nurses on what they think of various units for their service, and services for their unit, respectively. This would be useful information. For the first pass, I used historical patient-days as a proxy. I.e., service-unit combinations with high patient-days are likely the most desirable. Combinations with low patient-days are like okay, but not great. Combinations with zero patient-days are likely inappropriate. Specifically, for each combination, I used the highest patient-day number for any service-unit combination on that service as the denominator and scaled the patient-day numbers for all combinations on that service by it. So, this number ranges from 0 to 1. If / when we get the survey data, I would advocate for using a combination, as it is likely that not all service-unit combinations will have sufficient responses. One idea would be to weight the survey scores as N \* x, where N is the number of responses for that combination, and x is the average score, and to weight the historical patient-days data as 2 \* pd, where pd is the average patient days for that combination. The 2 means that for 0, 1, or 2 responses, the historical data is weighted heavily, but for combinations with more responses, the historical data is less influential on the eventual score. You’ll want to make sure the x and pd are on appropriately similar scales.