Scheduling Optimization Overview

Louise & Nitya

August 2024

1 Introduction

1.1 Optimality

What makes a schedule good/bad

- **demand** - distribution of appointments (how many appointments per day / avg appointments at a given time, etc) - does the data fit into a nice distribution (eg poisson or normal?)
- do want the schedule to be as close as possible to the model of demand?
- **objective function** - to minimize the difference between tutor working hours and demand (linear or quadratic?)

1.2 Demand

- counted avg appts for each day and time, and rounded up
- looks fairly accurate - we can use these to represent demand

1.3 Optimization problem formulation

Decision Variables

- variables $x_{i,j,k} \in \{0, 1\}$ deciding if tutor i will work on day j during time k

Constants

- Demand: 2d matrix, entries $D_{j,k}$ are demand for a given day j and hour k (rounded avg appts over a term)
- n - total number of tutors
- d - total number of days in a week
- h - total number hours in a day

- $w = dh$ - total hours in a week

Constraints

- Max 24 hrs per tutor per week $\rightarrow \sum_{j=1}^{w} x_{i,j} \leq 24 \ \ \forall \ i$
- Max 7 hrs per tutor per day? $\rightarrow \sum_{k=0}^{h} x_{i,j,k} \leq 7 \ \ \forall \ i, j$
- No more than 5 hrs without break $\rightarrow \sum_{l=0}^{5} x_{i,j,k+l} \leq 5 \ \ \forall \ i, j, k$
- Bounds $\rightarrow 0 \leq x_{i,j,k} \leq 1, \forall \ i, j, k$

Objective Function

- We have our demand values (excel sheet in Teams folder)
- We can minimize the difference between sum of tutors working & sum of demand
- Linear (absolute value) or quadratic function (square)
- $$f(\{x_{it}\}) = \sum_{t=1}^{n} \left(d_t - \sum_{i} x_{it} \right)^2$$
- $$f(\{x_{i,j,k}\}) = \sum_{j=1}^{d} \sum_{k=1}^{h} \left(D_{j,k} - \sum_{i=1}^{h} x_{i,j,k} \right)^2$$

2 Questions

- Depending on term, can we consider student's schedules to optimize demand even further? (eg. accounting students have classes on Thursday in the Fall, thus an accounting tutor on Thursdays would increase appointments)
- This would mean getting students' timetables, however this is not feasible (?) just yet.

3 Next Steps

- Add another variable that correlates to subjects (English, Math, etc.)
- Make output schedule more visually appealing
- Translate the code and math into readable English
- Fix any coding errors if any
- Program does not need to be perfect, as long as it is more optimal than random scheduling
-