Minimizing Commute Distance for Small Groups: A Linear Programming Approach

Kevin Payne
Portland State University

Kritika Kumari
Portland State University

Levi Huddleston
Portland State University

Rabi Hassan
Portland State University

Minimizing Commute Distance for Small Groups: A Linear Programming Approach

Course Title: Operations Research
Course Number: ETM540
Instructor: Timothy Anderson
Term: Winter
Year: 2018
Author(s): Kevin Payne, Kritika Kumari, Levi Huddleston, Rabi Hassan

ETM OFFICE USE ONLY

Report No.:
Type: Student Project
Note:

Abstract: This paper aims to minimize total drive time between members and their respective group leader. Given a limit on group size and days available, how can a formulation of a group occur such that the sum of the total drive time is minimized. To accomplish this task a Linear Program (LP) is implemented that includes three sets of binary decisions variables summing to 4100 variables and a variety of constraints summing between 4200 and 4341 depending on the constraints enforced. For 200 members and 15 leaders the minimized average commuting time was found to be between 4.99 and 5.36 minutes depending on the distribution of availability assumed.

Keywords: Linear Programming

Table of Contents

Introduction

One of the key functions of a church is to provide a member with a venue to build a sense of community with other church members. One of the primary reasons why congregants will select a small church to attend rather than a large church is the close-knit community that many small churches lend themselves to. Because regular church-goers of large churches rarely know – let alone recognize – all the other congregants, large churches will often offer spiritually-related special interest groups or connect congregants into small groups where members are encouraged to engage with their faith. While churches of moderate to large size can often accommodate a manual sorting process of connecting interested member to available small groups, the process can become cumbersome for churches with weekly worshippers above two thousand. The process is often complicated by the fact that members wish to attend a small group located close to their residence. Small groups are typically hosted by a member of the small group at their private residence, however it is not unusual that the group location alternates between members' residences.

Church leaders are tasked with connecting new congregants with existing groups, forming new groups, and/or encouraging mature congregants to form a group of their own. Many churches take a direct hand in the management of the small groups that form between the members of their church. Because the well-being of church-goers – especially regular church-goers who are involved beyond the Sunday service – is vital to the success of a church, it is important that the leaders of the church ensure that members are satisfied within their small groups and that the fundamental principles of faith held by the church are not compromised. There are many alternative purposes behind encouraging small groups. These purposes include but are not limited to constructing an avenue to build up more leaders for the church, providing an avenue for church-members to serve their local community or a social need that they feel passionate about, and increasing the health of a member's spiritual life. While the selection of members for a group does not have a deterministic effect on the fulfillment of the church's goals, factors such as the distribution of age and gender within a group, the geographic proximity of group members to each other, and the spiritual maturity of group members do play an important role in the degree of success in the above goals that the group achieves.

The analysis done in this paper is to cluster members of a church into multiple groups such that the characteristics of the group are optimal for success. An interview was conducted with Collin

Mayjack (Pastor of Communities) and Gavin Bennett (Director of Communities) from the Bridgetown Church in which the manual sorting process for connecting congregants with small groups was described and the formulation of this project began. All church-goers who wish to join a small group (known as a Bridgetown Community) must fill out a questionnaire. These questions will ultimately serve as our data for our model. For information privacy reasons, data on addresses of congregants could not be obtained. However, a dataset of addresses in the Portland area was obtained and transformed to the same format as the spreadsheet template provided by Bridgetown Church.

In addition to home address, the spreadsheet template also included fields for age, gender, schedule availability and desire to lead a small group. The distribution for age, gender, geographic proximity, and availability of members' schedules was discussed Bridgetown leaders. The demographics information could be used for future work but was not included in our initial analysis. The most recent Basics cohort consisted of approximately 200 members, with 20 members expressing interest in leading a group. An ideal group size consists of 10-15 people, but because members may opt out of their group, the optimization problem allows formation of groups sized 12-18 people. There is also a non-binding constraint that limits the maximum number of leaders to 15.

Literature Review

The problem of grouping members by leader is, at base, a clustering problem. However, because the problem involves assigning members to pre-specified leaders, typical data mining algorithms such as K-means clustering could not be used. Rather, nonlinear optimization is the tool used to select a specified number of leaders from a larger group of potential leaders such that the distance from members to leaders is minimized and availability constraints of members are satisfied.

In general, optimization problems have the following components:

- An objective – expressed as a function to maximize or minimize.
- Decision variables – elements of the problem at hand that can be changed.
- Parameters – characteristics of the problem that are fixed.
- Constraints – limits on the objective or decision variables.

Alfred Weber (1970) was the pioneer in research in location analysis, a field concerned with locating facilities such that cost and distance are minimized. Many facility location problems are multi-objective in nature, the top two objectives being cost/distance minimization and demand coverage. Profit maximization and environmental concerns are also occasionally factored into location analysis objective functions, but traditionally minimizing cost/time has been the primary objective of such models.[1]

Facility location problems can also be classified according to number of levels. Bi-level problems are significantly more difficult to construct, as one must formulate the model such that there is a leader and a follower in decision making and where information availability is different at different stages of decision making. [2] Our model involves only one level of decision-making: which members to assign to which leaders on a given day.

Our optimization problem could be categorized further as a scheduling problem, as the availability of each leader and member must match in order to form groups. Examining the literature in this field, we distinguished three main groups of scheduling problems: shift scheduling, days off scheduling and tour scheduling, which combines the first two types. Our project falls under the second category (days off or “day-of-week” scheduling), as the availability constraint in our model forces the optimization of member and leader schedules such that all members of a group can meet with the leader on at least one day of the week.[3]

The scheduling problem within our model can be classified more specifically as a special form of covering problem where one follower should be grouped with one leader. Schilling et al. (1993)[4] classify such models that use the concept of covering in two categories: (1) Set Covering Problems (SCP) where coverage is required and (2) Maximal Covering Location Problems (MCLP) where coverage is optimized. Our project takes the Set Covering Problem (SCP) approach.

---

[1] Current, J., Min, H., & Schilling, D. (1990). Multiobjective analysis of facility location decisions. European Journal of Operational Research, 49(3), 295-307.
[2] Caunhye, A. M., Nie, X., & Pokharel, S. (2012). Optimization models in emergency logistics: A literature review. Socio-economic planning sciences, 46(1), 4-13.
[3] Baker, K. R. (1976). Workforce allocation in cyclical scheduling problems: A survey. Journal of the Operational Research Society, 27(1), 155-167.
[4] Schilling, D. A. (1993). A review of covering problems in facility location. Location Science, 1, 25-55.

Revelle and Hogan (1989) introduce randomness in availability and considers travel time as deterministic. [5] Our analysis likewise considers member availability to be random and uses one non-rush-hour estimate of drive time between every member with each possible leader.

In their nurse scheduling problem, Brucker et al. [56] construct shift sequences for each nurse, with the differing skill levels of nurses being coded as hard constraints. [6] While the availability of members is constructed as a hard constraint in our model, the most prudent method of incorporating gender and age targets would be to formulate these goals as secondary objectives. Setting appropriate weights to these secondary objectives would be equivalent to forcing a soft constraint on the objective of minimizing distance. Thus, similar to the effect of availability on the objective value of our model, incorporating age and gender into the objective function would result in a solution in which members are matched with leaders that are potentially further away than otherwise. Formulating member characteristics as hard constraints (as in Brucker et al. [56]) would greatly increase the average distance between member and leader. Because the main concern of Bridgetown Church is to minimize distance between members and leaders, it was determined that factoring in gender and age would unnecessarily complicate the model.

Project Objective

This optimization problem selects leaders, and members to be in a group with those leaders, such that the overall drive time traveled is minimized. The first task after attaining the dataset which included those variables listed above, is to find the drive time traveled between members and leaders. This is the crux of the optimization problem, since the main objective is to minimize drive time subject to a variety of constraints. To find these times traveled it may be tempting to use some form of Euclidian distance as a measure and divide by average speed, but often times this can be misleading since actual drive times may be much different. To counteract this, this research pulls drive time data from Google Map's API in seconds, as well as distance. The total number of seconds driven for all members to their

[5] ReVelle, C., & Hogan, K. (1989). The maximum availability location problem. Transportation Science, 23(3), 192-200.
[6] Van den Bergh, J., Beliën, J., De Bruecker, P., Demeulemeester, E., & De Boeck, L. (2013). Personnel scheduling: A literature review. European Journal of Operational Research, 226(3), 367-385.

selected leaders will be the objective value for our minimization problem. Algebraically the formulation is as follows:

$$Minimize \, \{\sum_{l=1}^{20} \sum_{m=1}^{200} Time[m,l] * x[m,l]\}$$

$$Where\ Time[m,l]\ defines\ the\ drive\ time\ between\ member\ m\ and\ leader\ l, and$$

$$x[m,l]\ is\ equal\ to\ 1\ if\ member\ m\ is\ assigned\ to\ leader\ l, 0\ otherwise$$

There are many constraints needed in this model so that the assignments of individuals to leaders is both realistic and optimal. The first most obvious constraint is that every member needs to be assigned a leader. This is shown below.

$$\sum_{l=1}^{20} x[m,l] = 1 \qquad \forall\ Members\ m \qquad Constraint\ I$$

Next, it must be imposed that if a member is assigned to a leader, then it must be true that the leader has been has been chosen out of the pool of potential leaders. It is important to recall that not all individuals who specify that they wish to lead a group will end up leading. For this reason, it is necessary to impose the following condition.

$$x[m,l] \le y[l] \qquad \forall \quad Members\ m\ and\ Leaders\ l \qquad Constraint\ II$$

$$Where\ y[l]\ is\ equal\ to\ 1\ if\ potential\ leader\ l\ is\ assigned\ to\ be\ a\ realized\ leader$$

The above condition restricts the value of $x[m,l]$ such that no member m will be assigned to a group leader l if the group leader does not exist. Next, a restraint must be in place to limit the total number of leaders. Otherwise all leaders will likely be selected since it could reduce total drive time. The following constraint restricts the total number of leaders allowable.

$$\sum_{l=1}^{20} y[l] \le 20 - k \qquad Constraint\ III$$

$$Where\ k\ is\ a\ predefined\ parameter\ by\ the\ user$$

The last constraint that will make the optimization problem realistic, is that the total size of every group needs to be greater or equal to twelve but less than or equal to eighteen. This constraint can be implemented as follows.

$$\sum_{m=1}^{200} x[m,l] \leq 18 \ \ and \ \ \sum_{m=1}^{200} x[m,l] \geq 12 * y[l] \quad \forall \quad Leaders\ l \qquad Constraint\ IV$$

These constraints insure that for all leaders the maximum number of members must be less than or equal to 18. In addition, if the leaders are selected, then their group members should sum greater than or equal to 12. These constraints would ensure that realistic groupings would be made.

More constraints are needed in order to take into account the added dimension of availability. If two individuals do not have matching schedules then these two individuals should not be placed in the same group even if the two individuals are close in proximity. We can extend this so that if a member is assigned a group then all members in that group must have a common day availability. This ensures that there are no scheduling conflicts between members in the same group. This constraint can be implemented as follows.

$$\sum_{m=1}^{200} x[m,l] * Availability[m,l,D] \geq \left(\sum_{m=1}^{200} x[m,l] - K\right) - 18 * (r[l,D] - y[l] + 1) \qquad Constraint\ V$$

$$\forall \ \ Leaders\ l\ and\ Days\ D$$

$$And$$

$$\sum_{1}^{4} r[D,l] \leq 3 \quad \forall\ Leaders\ l \qquad Constraint\ VI$$

$$Where\ Availability\ is\ equal\ to\ 1\ if\ membe?\ m\ and\ leader\ l\ can\ meet\ on\ Day\ D, 0\ otherwise$$

$$r[l,D]\ is\ equal\ to\ 1\ if\ Leader\ l\ on\ Day\ D\ is\ not\ selected$$

$$K\ is\ a\ predefined\ parameter\ that\ softens\ the\ contsraint$$

The above statements in Constraints V and VI ensure that if a member is assigned to a group then the total availability must be greater than or equal to the total number of individuals in that group for at least one day D. Since $r[l, D]$ needs to be less than three in Constraint VI, this ensures only one of the three days will be a binding constraint. In addition, the included $y[l]$ in Constraint VI provides a measure such that if the member is not created then the group availability constraint will easily be met. *K* is equal to 0 if the constraint is hard or is chosen as equaling 2 if the constraint is soft. The total population's availability including both members and leaders in terms of percentages follows two distributions. The first is often referred in this paper as the 'More Availability' matrix and follows the following distribution with 1 percent of individuals are available for only 1 day, 1 percent of individuals are available for only 2 days, 30 percent of individuals are available for 3 days, and 68 percent of individuals are available for all four days. The second distribution often referred to as the 'Fewer Availability' matrix has a distribution such that 5% are available for only one day, 25% for two, 40% for three and 30% for four. These constraints together with an availability matrix will provide for a minimized drive time traveled that will place members into groups and select respective leader for those groups such that there is at least one day where everyone can meet.

Results

As noted in the abstract, the optimal average commute time was between 4.99 and 5.36 minutes for the 200 members depending on availability. In R studio the LP solver "symphony" and "GLPK" were used to solve these problems. The "symphony" solver solution was ultimately used. Figure 1 in the appendix displays the initial problem set with members coded in the color blue and leaders coded in with the color red. Figure 2 displays the same addresses, yet color codes each member's address based on who their respective leader is for their group. This optimization uses all constraints and the 'More Availability' matrix with *K* set to zero to find an objective value of 59870 seconds or 4.99 minutes on average per person. An accessible version, with interactive tools for both maps has been made accessible through the following links.

Figure 1: https://drive.google.com/open?id=1r8m2RUSJWpFgb6FZSjkWKiVrWt7e4-Fy&usp=sharing

Figure 2: https://drive.google.com/open?id=1t7aPwC66Bbo38NjuCMkVC7_Xb7skBDmv&usp=sharing

In addition to the initial problem, a variety of other problems were also solved. One of these problems was to assess the impacts of the availability, group size, and leader size constraints on the optimal solution. It can be observed in the description of the 'More Availability' matrix that 98 percent of all individuals in the population are available for more than three days a week. As a result, it would be expected that the removal of such an easily satisfied constraint may not drastically affect the optimal solution. Removing the other constraints would decrease commute time, but the size of the effect may be vaguer. After removing all constraints dealing with group size, number of leader and availability, time traveled decreased by about 18 seconds per person. Although this amount may seem small on an individual level, the total increase in time traveled due to constraints is 3,642 seconds – about an hour of extra commute time aggregated across all members each week.

While it may be interesting to analyze the impact of the group size and leader size constraints on commute time, these constraints are essential components of the solution desired by the church. Using an availability matrix that is more aligned with actual reported availability of church members (the 'Less Availability' matrix), the 'symphony' solver reports an infeasible solution. To find a solution, K is set to two so that at least two members in a group are allowed to be unavailable. Under this formulation, the LP problem is feasible and the optimal solution for the 'Less Availability' matrix is 60449 seconds (on average 5.36 minutes per member). Comparing this solution to the previous result where the 'More Availability' matrix was used, we find that reducing the availability of members and leaders adds a total of 60449 – 59870 = 579 seconds or 9.65 minutes to total commute time per week. This is particularly interesting, since availability decreases quite dramatically, while the solution only increases slightly in terms of time. The laxing of the availability constraint with K equal to two, allows for this modest increase. The clustering of this LP can be seen in Figure 3, or in the link provided below.

Figure 3: https://drive.google.com/open?id=1LDiGu6W57vkdD0wkQ9IKojFlFmpvEYjd&usp=sharing

Figures 4 and 5 give a summary of the findings with different constraints. Figure 4 presents the objective value for a variety of different constraints. As is expected and will always be the result of imposing more stringent constraints, the objective value increases with each addition of a new constraint. Special attention should be given to the impact of having K equals to two on the objective function. Not only does it allow the LP to find a solution, but it only marginally increases the objective value. Figure 5 displays how many members each leader is assigned for differing levels of constraints. With no constraints imposed, number of members per leader is volatile, with some leaders having 31 members while others only have 1 member. For formulations involving limits on group size, however, the differences in number of members assigned to each leader vary little with changes in the availability matrix. One notable exception is the model using the 'Less Availability' matrix, which assigns no members to leader 19. It should also be noted that Figure 5 only compares the number of members assigned to each leader under different formulations of the model but does not inform us as to whether individual members are consistently assigned to the same group. The consistency of individual member-group assignments across models can be verified by crosschecking the interactive maps of Figure 1, 2 and 3.

Although not discussed in detail here, the same optimizations were conducted on walking time. The maps for these LP problems are in the same order in terms of constraints as the previous problems with drive time and can be viewed in the following links.

https://drive.google.com/open?id=1gLjbDIXJA98N1PH0wtssUmbJVjMExBt-&usp=sharing

https://drive.google.com/open?id=1e4BNlbWVow32TxFuMnnltxZkfn-CjqQm&usp=sharing

https://drive.google.com/open?id=1LviR-TMPC2i0czUoTkBSvsrkUe2tuSFA&usp=sharing

Conclusion

The analysis done in this paper lays the groundwork for automating the assigning of members to leaders at Bridgetown Church. While the data were generated to match a template provided by the church and the solution is merely a proof of concept, the potential usefulness of the model framework is significant. Our model generates the most efficient groups possible subject to the availability of every member and leader.

Though our project is solving a problem of effective group formation for a church, the model we proposed to find the optimum solution that reduces the travel time for the participants in the group can easily be replicable to solve a myriad of problems. For example, consider a retail business model, where firms are constantly trying to reach out to as many customers as possible without incurring large costs related to the construction of facilities. Our model would minimize the distance between customers and stores, making the decision of where to build very simple, given a list of potential construction sites. Similarly, when setting up a new hospitals or bank, decision makers are chiefly concerned with how they could make the service provided more accessible to all customers. Our model again would make the decision clear as to what the optimal allocation would be. Additional constraints or weights could be added, so that different areas might be associated with a higher priority relative to other areas.

Our original data set has demographic information of all the members. It is natural and compelling that an extension of our model would include such factors when creating groups. This type of improvement would make the model more inclusive and not only solve the distance minimization problem but would add yet another layer on top of the availability constraint. Such a model would undoubtedly prove of interest to many businesses, since optimal allocation, may need to include some reference as to what type of people live around the area.

Other extensions of our research work could be testing the effectiveness of the created groups. For Bridgetown Church this could be illustrated by the turnover rate in each group under the leadership of a particular leader. In doing this one could create a set of efficiency indices that could measure the performance of those groups in comparison to other groups. This could potentially use a data envelopment analysis method as has been suggested by some papers in group leadership theory.

While there are many more extensions that come naturally, this model solves the issue of creating groups, while accounting for not only distance but for availability as well. This method of clustering will provide value added to the Bridgetown Church, as it will now have the option of creating groups with compatible schedules, as opposed to the previous technique where it was unaccounted for in their method. This value is embodied in the time saved in selecting groups that the church leadership goes through, as well as the time saved for each member in terms of drive time each week.

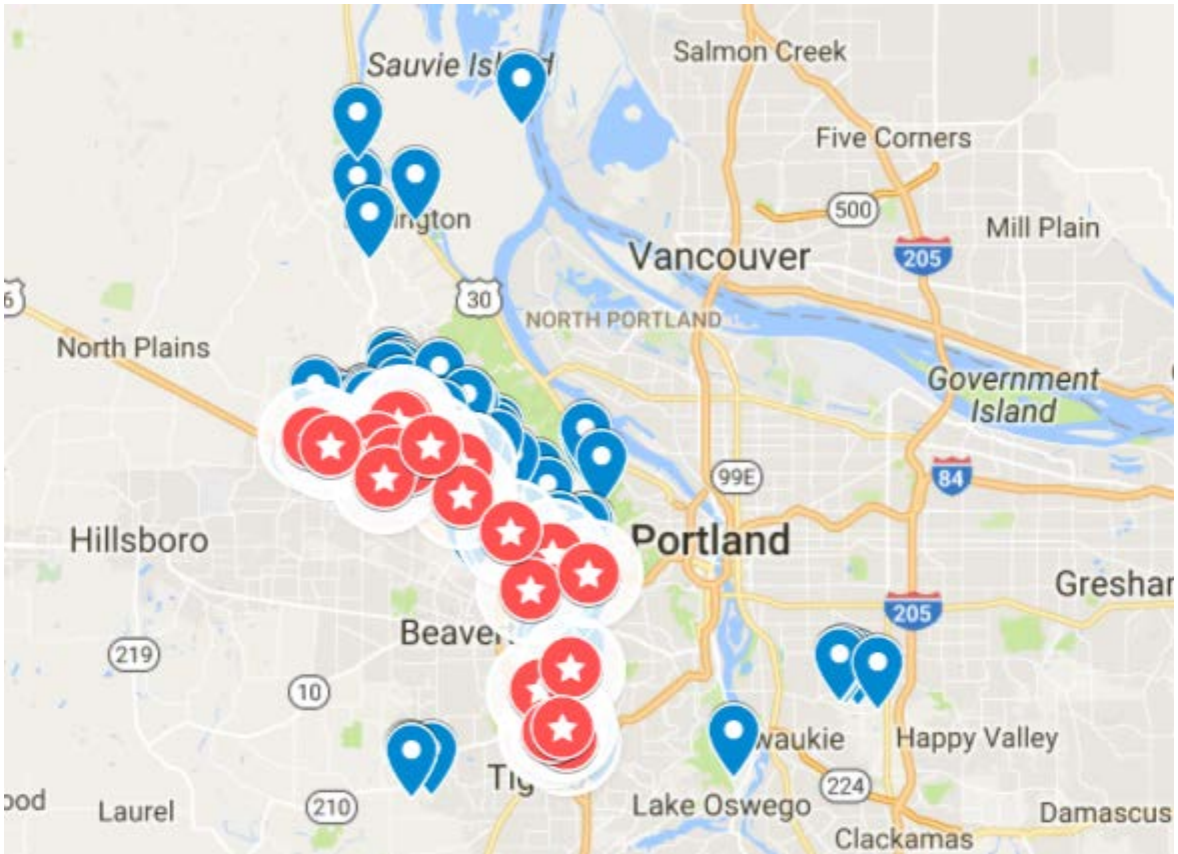


Figure 1: Initial Minimization Problem

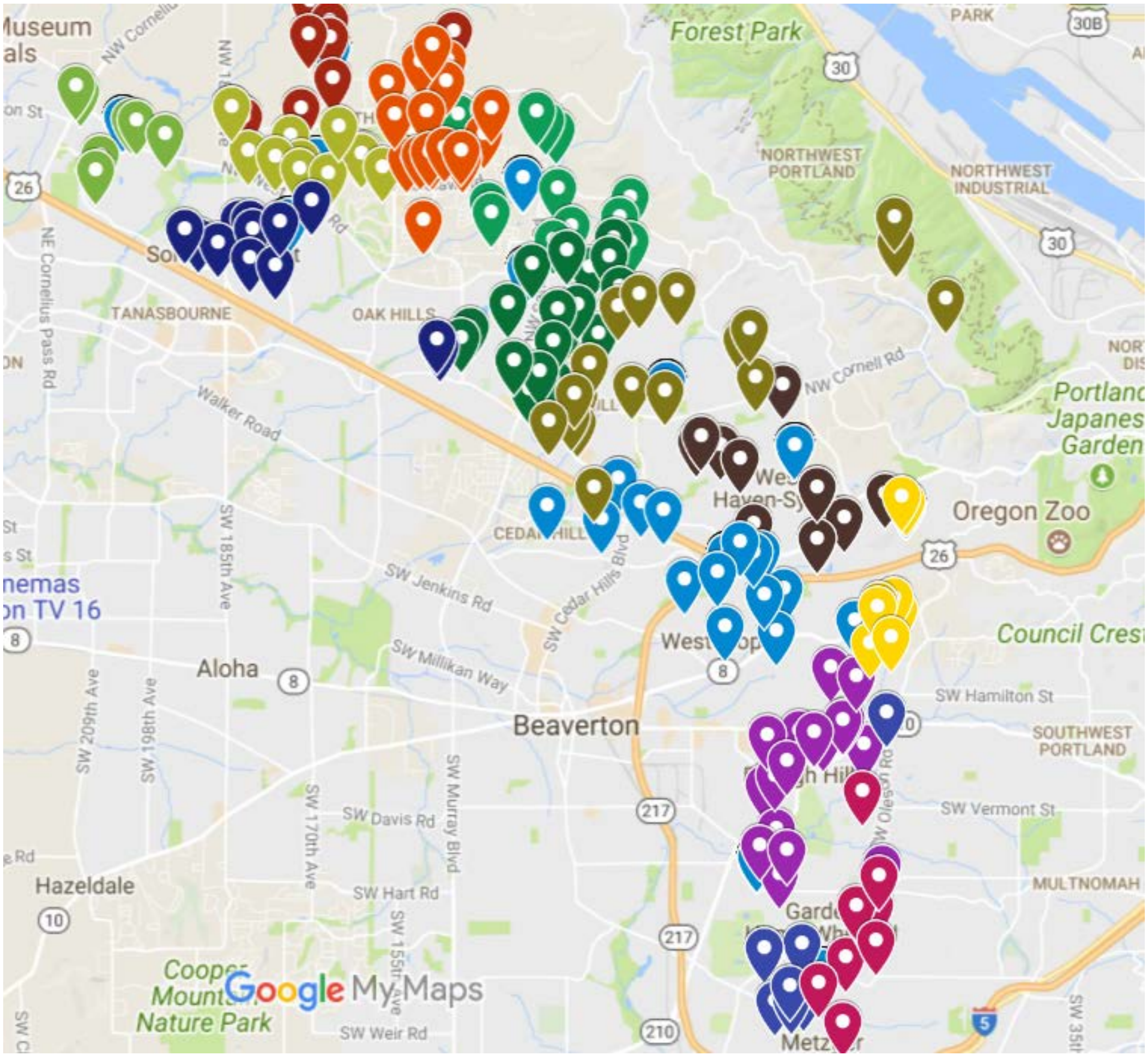


Figure 2: Group Assignments for Drive Time 'More Availability'

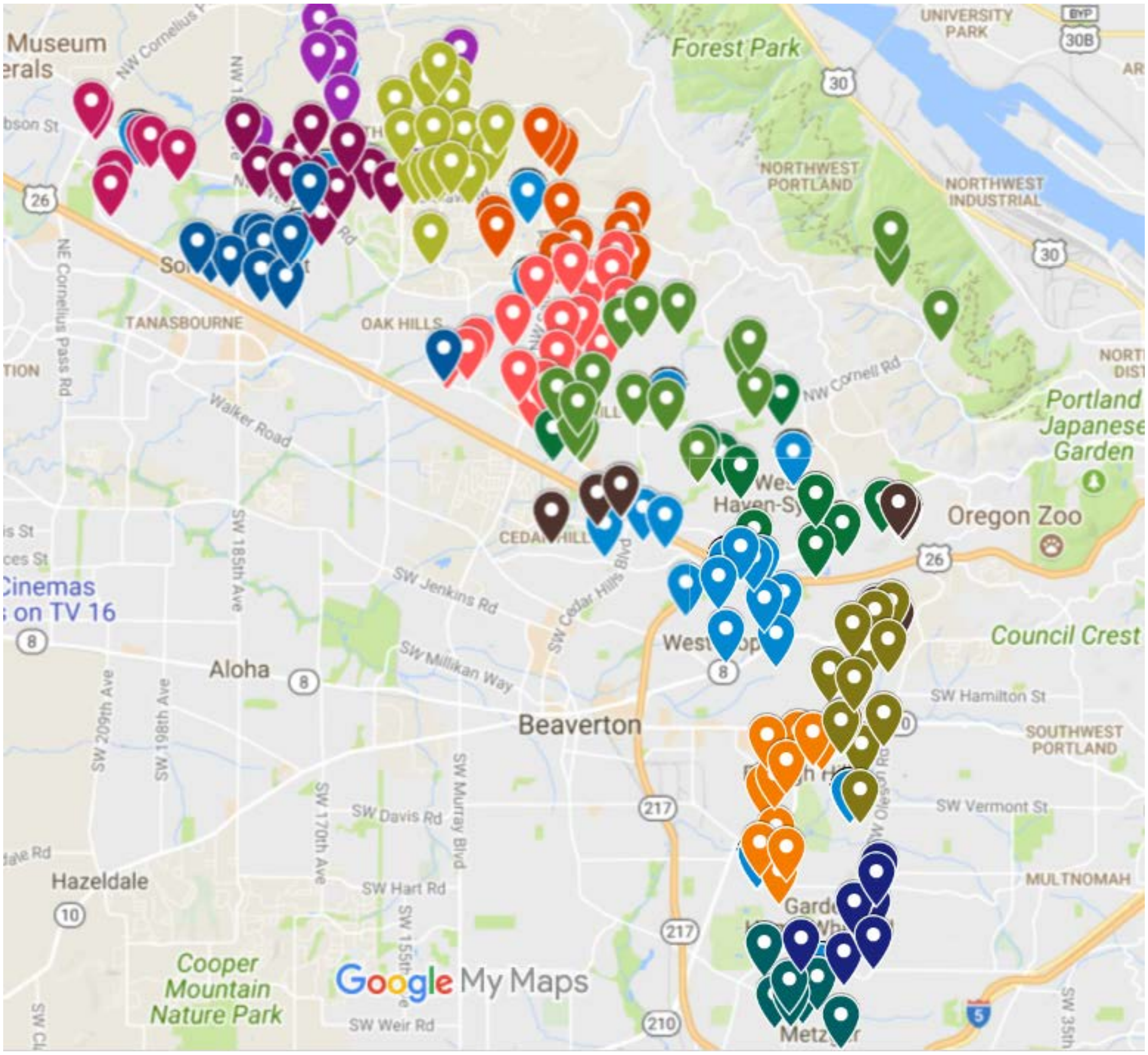


Figure 3: Group Assignments for Drive Time 'Less Availability'

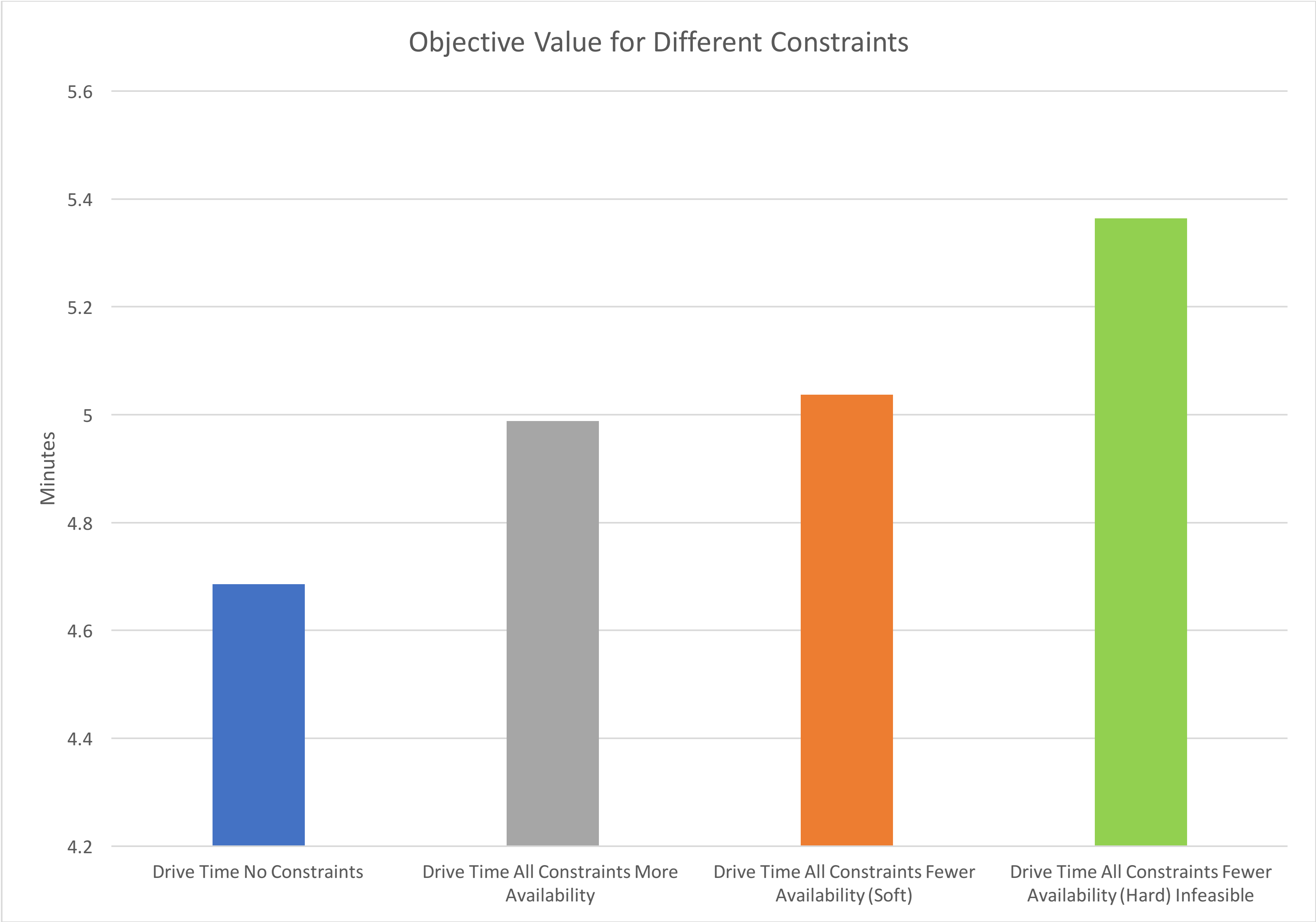
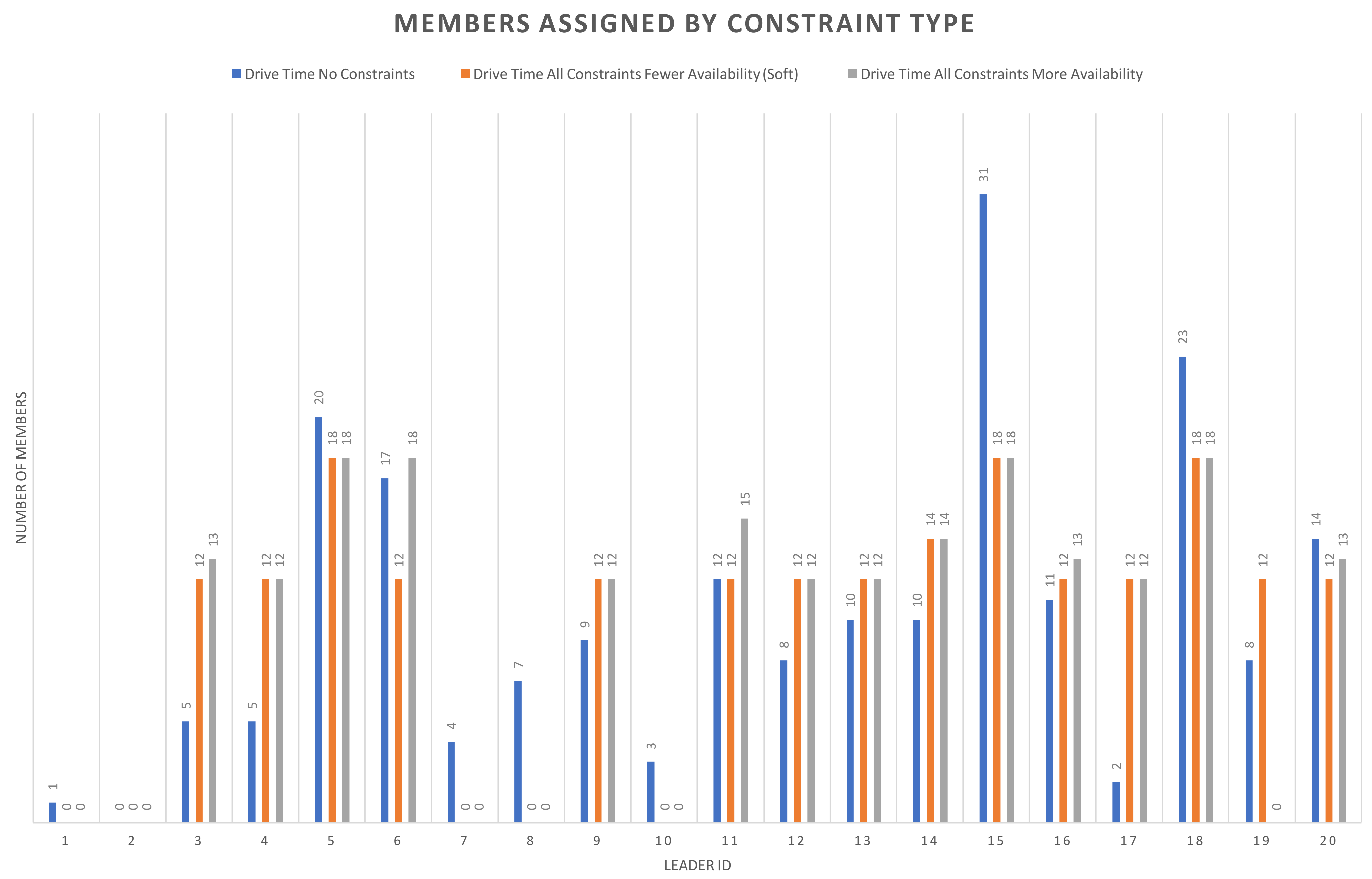


Figure 4: Objective Value for Different Constraint



MEMBERS ASSIGNED BY CONSTRAINT TYPE
Drive Time No Constraints
Drive Time All Constraints Fewer Availability (Soft)
Drive Time All Constraints More Availability
NUMBER OF MEMBERS
LEADER ID
1 0 0
0 0 0
5 12 13
5 12 12
20 18 18
17 12 18
4 0 0
7 0 0
9 12 12
3 0 0
12 12 15
8 12 12
10 12 12
10 14 14
31 18 18
11 12 13
2 12 12
23 18 18
8 12 0
14 12 13
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

R Code

Kritika Kumari, Rabi Hassan, Levi Huddleston and Kevin Payne

3/9/2018

```
#Model

install.packages("ompr.roi")
install.packages("ROI.plugin.glpk")
install.packages("ROI.plugin.symphony")
install.packages("slam")
install.packages("devtools")
install.packages("data.table")
install.packages("tidyr")
install.packages("ompr")
install.packages("magrittr")
install.packages("dplyr")
install.packages("dplyr")
install_url("https://cran.r-project.org/src/contrib/Archive/slam/slam_0.1-37.tar.gz")

require(slam)
require(ompr.roi)
require(ROI.plugin.glpk)
require(ROI.plugin.symphony)
require(devtools)
library(data.table)
library(tidyr)
library(ompr)
library(magrittr)
library(dplyr)
library(stringr)
library(readr)

setwd("C:/Users/Levi/Documents/Portland State/Operations Research/Project")
#save(results_distance,file="distance")
#save(availability_leaders_matrix,file="availability_leaders")
#save(availability_members_matrix,file="availability_members")
#save(availability_leaders_matrix,file="availability_leaders_sparse") #5,25,40,30
#save(availability_members_matrix,file="availability_members__sparse") #5,25,40,30

#load(file= "distance") #Drive time
results_distance <- read_csv("~/Portland State/Operations Research/Project/distance_walk.csv")
load(file="availability_leaders")
availability_leader=availability_leaders_matrix
load(file="availability_members")
```

```
availability_member=availability_members_matrix
distance <- as.data.table(results_distance[,1:3])
rm(results_distance,availability_leaders_matrix,availability_members_matrix)
#distance=distance[order(Time.de)]
#distance <- transform(distance, id=match(Time.de, unique(Time.de)))
a=200
b=20
k=5 # b-k=actual number of groups
uplimit=18
lowlimit=12
availabilty_soft=2



distance <- spread(distance, Time.de, Time.Time)
setnames(distance,"Time.or","Member Locations")
distance=as.matrix(distance)
distance=na.omit(distance)

distance=distance[1:a,1:(b+1)]
availability_leader=availability_leader[1:b,]
availability_member=availability_member[1:a,]


availability=matrix(0,nrow=a*b,ncol=4)

for(m in 1:a)
{
for(day in 1:4)
{
 for(l in 1:b)
 {
  availability[b*(m-1)+l,day]=availability_leader[l,day]*availability_member[m,day]
 }
}
}
colnames(availability)=colnames(availability_leader)
availability=as.data.table( availability)
availability$L=rep(1:b,a)


member_stuff=NULL
for (i in 1:a)
{
 temp=rep(i,b)
 dim(temp)=c(b,1)
```

```
 member_stuff=rbind(member_stuff,temp)
}
availability$M=member_stuff
rm(temp,member_stuff,availability_leader,availability_member)

#FORMULATING THE MODEL
model <- MIPModel() %>%
  # 1 iff member m gets assigned to leader l
  add_variable(x[m, l], m = 1:a, l = 1:b, type = "binary") %>%

  # 1 iff leader l is choosen
  add_variable(y[l], l = 1:b, type = "binary") %>%

  # 1 if on Day D for leader L the day is not chosen as the meeting
  add_variable(r[D,l],D=1:4,l=1:b, type = "binary") %>%

  # Minimize the distance
  set_objective(sum_expr(as.numeric(distance[m,l+1] )* x[m, l], m = 1:a, l = 1:b)
              , "min") %>%

  # every member needs to be assigned to a leader
  add_constraint(sum_expr(x[m, l], l = 1:b) == 1, m = 1:a) %>%

  # if a member is assigned to a leader,
  # then this leader must be the actual leader of the group
  # not just the potential leader
  add_constraint(x[m,l] <= y[l], m = 1:a, l = 1:b) %>%



  # Less leaders than the total
  add_constraint(sum_expr(y[l],l=1:b)<=(b-k)) %>%

  #Number of members in each group needs to be between 12 and 18
  add_constraint(sum_expr(x[m,l],m=1:a)<=(uplimit),l=1:b) %>%
  add_constraint(sum_expr(x[m,l],m=1:a)>=(lowlimit)*y[l],l=1:b)%>%

  #Availability,atleast one day in the week, they all can meet on the same day
  add_constraint(
   sum_expr(x[m,l]*as.numeric(availability[M==m & L==l,1]),m=1:a)
   >=
    (sum_expr(x[m,l],m=1:a)-availabilty_soft)-uplimit*(r[1,l]-y[l]+1),l=1:b) %>%

  add_constraint(
   sum_expr(x[m,l]*as.numeric(availability[M==m & L==l,2]),m=1:a)
   >=
    (sum_expr(x[m,l],m=1:a)-availabilty_soft)-uplimit*(r[2,l]-y[l]+1),l=1:b) %>%
```

```
  add_constraint(
   sum_expr(x[m,l]*as.numeric(availability[M==m & L==l,3]),m=1:a)
   >=
    (sum_expr(x[m,l],m=1:a)-availabilty_soft)-uplimit*(r[3,l]-y[l]+1),l=1:b) %>%

  add_constraint(
   sum_expr(x[m,l]*as.numeric(availability[M==m & L==l,4]),m=1:a)
   >=
    (sum_expr(x[m,l],m=1:a)-availabilty_soft)-uplimit*(r[4,l]-y[l]+1),l=1:b) %>%

  add_constraint(sum_expr(r[D,l],D=1:4)<=3,l=1:b)

model


result <- solve_model(model, with_ROI(solver = "symphony",time_limit=1200,gap_limit=.1))
result

#result <- solve_model(model, with_ROI(solver = "glpk", verbose = TRUE))
#result

matching <- result %>%
  get_solution(x[i,j]) %>%
  filter(value > .9) %>%
  select(i, j) #%>%
  #arrange(i)

matching=as.data.table(matching)
setnames(matching,"i","Member ID")
setnames(matching,"j","Leader ID")

groups=data.table(NULL)
for(p in 1:b)
{
  if(p %in% unique(matching$`Leader ID`))
  {
groups=rbind(groups,cbind(distance[matching[`Leader ID`==p]$`Member ID`,1],colnames(distance)[p+1
]))
}
}
setnames(groups,"V1","Member Locations")
setnames(groups,"V2","Leader Location")
tabulate(matching$`Leader ID`)

###Levi's Code
```

```
List=cbind(matching[1:200,1],groups[1:200,1],matching[1:200,2],groups[1:200,2])
write.csv(List,"List.csv")
```