Final Project

Bharath Chennu, Michael Geist, Elmy Luka

2011-12-11

Objective : You have been tasked with the objective of forming groups. Assume that your class consists of 12 students, and you would like to form 4 groups of 3 students each. Your primary objective is to ensure that you maximize the chance that each group will do well on a class project. Here are the requirements to form groups:
1.Each group should have exactly 3 students
2.The objective is to maximize the chance of success for each group on a class project
Before you can solve this problem, there are several issues that must be addressed. Some of these being:
What factors affect the success of groups? Define three factors, e.g., GPA, gender, etc., that you feel affect the contributions that students make towards project success.
How do the above factors combine to define success? For example, is a person with high GPA the same as one with a more relevant background? Decide on how each of the factors contribute toward your definition of success.
How will you collect data for these factors? For this assignment, randomly generate sensible data for each of the above three defined factors.

What are your decision variables?
What is your objective function?
What are your constraints?
Formulate and solve the problem. Provide the corresponding R markdown file, and a narrative recorded presentation justifying your approach to the choice of factors, data collection, and formulation.

So the first thing we did is we wanted to make a table of data Within excel we used the RAND function and INT(RAND()*(a specific number)) That randomly generated data is being display below.

Student	GPA	Attendance	Current Course Grade
1	2.23	20	85
2	2.18	25	56
3	1.92	6	69
4	3.49	11	93
5	2.87	17	78
6	1.92	24	72
7	2.09	23	80
8	2.06	14	64
9	3.18	12	85
10	3.98	24	89
11	2.49	16	97
12	4.00	20	88

The three different variables we picked were GPA, Attendance, and then Course Grade.
GPA is between 0 and 4. The higher the number the the better the student grade would be.
Attendance is based on 25 classes.
Course Grade is based on 100 percent to 50 percent(We made the assumption that if they were below 50 they would have withdrew from class).
We also made sure that the number presented by the random number generator within excel would make sense. Someone with a 4.00 GPA would not have a 50 percent in the class (most often). As long as it made sense we left it alone.

Now we want to formulate the problem. We will let X represent GPA. We will let Y represent Attendance. We will let Z represent current class grade.
Objective Function

$$MaxZ = 85\sum_{i=1}^{4} x_{i1} + 56\sum_{i=1}^{4} x_{i2} + 69\sum_{i=1}^{4} x_{i3} + 93\sum_{i=1}^{4} x_{i4} + 78\sum_{i=1}^{4} x_{i5} + 72\sum_{i=1}^{4} x_{i6} + 80\sum_{i=1}^{4} x_{i7} + 64\sum_{i=1}^{4} x_{i8} + 85\sum_{i=1}^{4} x_{i9} + 89\sum_{i=1}^{4} x_{i10} + 97\sum_{i=1}^{4} x_{i11}$$

You can see that we are using the grades of each person along with the summation of i being the number of groups.

Our Objective Function comes with the following constants:
Constraint for Groups

$$\sum_{j=1}^{12} x_{ij}$$

where i= groups (1-4) and j=Students (1-12)

Constraint for GPA

$$\sum_{j=1}^{12} GPA_j x_{ij} >= 12$$

where i= groups (1-4) and j=Students (1-12)

Constraint for Attendance

$$\sum_{j=1}^{12} GPA_j x_{ij} >= 75$$

where i= groups (1-4) and j=Students (1-12)

Constraint to only assign a student to one group

$$\sum_{i=1}^{4} x_{ij} = 1$$

where i= groups (1-4) and j=Students (1-12)

```
library(lpSolve)
library(lpSolveAPI)
data <- data.frame(Person = c(1:12),
                        GPA = c(2.23, 2.18, 1.92, 3.49, 2.87, 1.92, 2.09, 2.06, 3.18, 3.98, 2.49, 4.00),
                        Attendence = c(20,25,6,11,17,24,23,14,12,24,16,10),
                        ClassGrade = c(85,56,69,93,78,72,80,64,85,89,97,88))
#make linear programming object
lp <- make.lp(0,48)
#setting as integer
set.type(lp, 48, "integer")
set.objfn(lp, rep(c(85,56,69,93,78,72,80,64,85,89,97,88,
                    85,56,69,93,78,72,80,64,85,89,97,88,
                    85,56,69,93,78,72,80,64,85,89,97,88,
                    85,56,69,93,78,72,80,64,85,89,97,88)))
#setting the LP to be a maximization
lp.control(lp, sense="max")
```

```
## $anti.degen
## [1] "fixedvars" "stalling"
##
## $basis.crash
## [1] "none"
##
## $bb.depthlimit
## [1] -50
##
## $bb.floorfirst
## [1] "automatic"
##
## $bb.rule
## [1] "pseudononint" "greedy"       "dynamic"      "rcostfixing"
##
## $break.at.first
## [1] FALSE
##
## $break.at.value
## [1] 1e+30
##
## $epsilon
##       epsb       epsd      epsel     epsint epsperturb   epspivot
##      1e-10      1e-09      1e-12      1e-07      1e-05      2e-07
##
## $improve
## [1] "dualfeas" "thetagap"
##
## $infinite
## [1] 1e+30
##
## $maxpivot
## [1] 250
##
## $mip.gap
## absolute relative
##    1e-11    1e-11
##
## $negrange
## [1] -1e+06
##
## $obj.in.basis
## [1] TRUE
##
## $pivoting
## [1] "devex"    "adaptive"
##
## $presolve
## [1] "none"
##
## $scalelimit
## [1] 5
##
## $scaling
## [1] "geometric"   "equilibrate" "integers"
##
## $sense
## [1] "maximize"
##
## $simplextype
## [1] "dual"   "primal"
##
## $timeout
## [1] 0
##
## $verbose
## [1] "neutral"
```

```
#adding the constraint for the groups to be formed (3 people per group)
add.constraint(lp, c(rep(1,12),rep(0,36)), "=", 3)
add.constraint(lp, c(rep(0,12),rep(1,12),rep(0,24)), "=", 3)
add.constraint(lp, c(rep(0,24),rep(1,12),rep(0,12)), "=", 3)
add.constraint(lp, c(rep(0,36),rep(1,12)), "=", 3)
#Adding the GPA constraint (no one can have higher than 4.00*3 people =12)
add.constraint(lp, rep(data$GPA,4), ">=", 12)
#Adding attendance constraints (no one could have come to class more than 25 times * 3 people = 75)
add.constraint(lp, rep(data$Attendence,4), ">=", 75)
#Making sure that each student is counted 1 time so that student could not be added too multiple groups
add.constraint(lp, rep(c(1,rep(0,11)),4), "=", 1)
add.constraint(lp, rep(c(rep(0,1),1,rep(0,10)),4), "=", 1)
add.constraint(lp, rep(c(rep(0,2),1,rep(0,9)),4), "=", 1)
add.constraint(lp, rep(c(rep(0,3),1,rep(0,8)),4), "=", 1)
add.constraint(lp, rep(c(rep(0,4),1,rep(0,7)),4), "=", 1)
add.constraint(lp, rep(c(rep(0,5),1,rep(0,6)),4), "=", 1)
add.constraint(lp, rep(c(rep(0,6),1,rep(0,5)),4), "=", 1)
add.constraint(lp, rep(c(rep(0,7),1,rep(0,4)),4), "=", 1)
add.constraint(lp, rep(c(rep(0,8),1,rep(0,3)),4), "=", 1)
add.constraint(lp, rep(c(rep(0,9),1,rep(0,2)),4), "=", 1)
add.constraint(lp, rep(c(rep(0,10),1,rep(0,1)),4), "=", 1)
add.constraint(lp, rep(c(rep(0,11),1),4), "=", 1)
solve(lp)
```

```
## [1] 0
```

```
get.objective(lp)
```

```
## [1] 956
```

```
get.variables(lp)
```

```
##  [1] 0 0 0 0 0 1 1 0 0 1 0 0 1 1 0 0 0 0 0 0 0 1 0 0 0 0 1 1 0 0 0 0 0 0 1 0 0
## [39] 1 0 0 0 0 1 1 0 0 0
```

```
data1 <- data.frame(Person = c(1:12),
                    GPA = c(2.23, 2.18, 1.92, 3.49, 2.87, 1.92, 2.09, 2.06, 3.18, 3.98, 2.49, 4.00),
                    Attendence = c(20,25,6,11,17,24,23,14,12,24,16,10),
                    ClassGrade = c(85,56,69,93,78,72,80,64,85,89,97,88),
                    Group = c(2,2,4,3,3,1,1,4,4,1,2,3))
data1
```

```
##    Person  GPA Attendence ClassGrade Group
## 1       1 2.23         20         85     2
## 2       2 2.18         25         56     2
## 3       3 1.92          6         69     4
## 4       4 3.49         11         93     3
## 5       5 2.87         17         78     3
## 6       6 1.92         24         72     1
## 7       7 2.09         23         80     1
## 8       8 2.06         14         64     4
## 9       9 3.18         12         85     4
## 10     10 3.98         24         89     1
## 11     11 2.49         16         97     2
## 12     12 4.00         10         88     3
```