Final Project -QMM

Chathurani Ekanayake

2023-12-10

Data set up

What are the three factors that influence group success?

The GPA- range on scale of 0.0-4.0 Experience - consider experience 0-4 years Interest - the factor whether students are interested in the group project or not impact the successfullness of the project-(0-not interested, 1-interested)

Due to the dataset is not provided we have randomly generated student information under above mentioned three factors.

Loading required packages

```
library(lpSolve)
```

```
## Warning: package 'lpSolve' was built under R version 4.2.3
```

```
library(lpSolveAPI)
```

```
## Warning: package 'lpSolveAPI' was built under R version 4.2.3
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2
## --
```

```
## v tibble  3.1.8     v purrr   1.0.1
## v tidyr   1.3.0     v stringr 1.5.0
## v readr   2.1.3     v forcats 0.5.2
## -- Conflicts --------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
##
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(gridExtra)
```

GPA

```
 round(runif(15, min = 0, max = 4.0), 2)  # Simulating GPAs between 0 and 4.0
```

```
##  [1] 2.07 1.46 1.62 1.32 1.14 2.87 2.85 1.65 3.10 2.14 2.86 2.60 1.89 0.55 2.63
```

Each time the code is executed, the values generated for the random GPAs will differ.Therefore the result of my execution is 3.57 3.18 2.02 1.69 2.62 3.55 3.57 0.71 1.03 0.45 3.19 0.44 3.32 3.08 3.77

Experience

```
round(runif(15,min = 0,max = 4)) # Assume that students do not have experience more than 4 years
```

```
##  [1] 3 3 2 3 2 3 3 0 0 2 2 4 3 1 4
```

Each time the code is executed, the values generated for the random Experience will differ.Therefore the result of my execution is 4 4 1 4 2 2 2 3 0 1 3 1 3 1 2.

Interest

```
round(runif(15,min = 0,max=1)) # Assume 0 as not interested and 1 as interested
```

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

Each time the code is executed, the values generated for the random Interest will differ.Therefore the result of my execution is 1 1 0 0 0 1 1 1 1 1 1 1 1 1 1

All the data of above three factors are in the same scale, so we do not need to re scale the above data.

For better understanding we can summarize as follow the results of above three factors for each 15 students. *Y1: GPA= 3.57, Experience= 4, Interest = Yes(1)* Y2: GPA= 3.18, Experience= 4, Interest = Yes(1) *Y3: GPA= 2.02, Experience= 1, Interest = No(0)* Y4: GPA= 1.69, Experience= 4, Interest = No(0) *Y5: GPA= 2.62, Experience= 2, Interest = No(0)* Y6: GPA= 3.55, Experience= 2, Interest = Yes(1) *Y7: GPA= 3.57, Experience= 2 , Interest = Yes(1)* Y8: GPA= 0.71, Experience= 3 , Interest = Yes(1) *Y9: GPA= 1.03,*

Experience= 0, Interest = Yes(1) Y10: GPA= 0.45, Experience= 1 , Interest = Yes(1) *Y11: GPA= 3.19, Experience= 3 , Interest = Yes(1)* Y12: GPA= 0.44, Experience= 1 , Interest = Yes(1) *Y13: GPA= 3.32, Experience= 3 , Interest = Yes(1)* Y14: GPA= 3.08, Experience= 1 , Interest = Yes(1) *Y15: GPA= 3.77, Experience= 2 , Interest = Yes(1)

We assume that all variables are equally important to make success the group project. So we assign equal weight for each factor and calculate the weighted average of the data = 5.412667

Define the GPAs, Experiences, and Interests for each Y

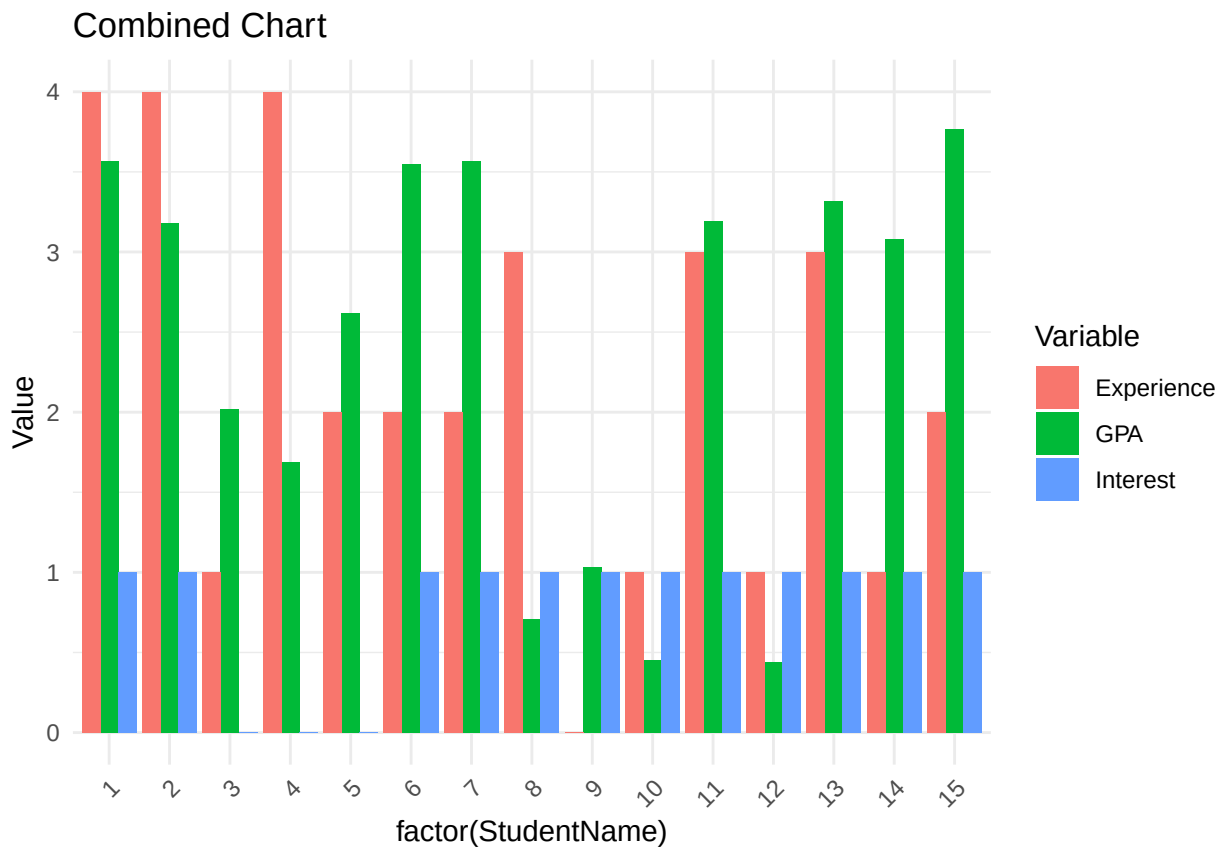
```
GPAs <- c(3.57, 3.18, 2.02, 1.69, 2.62, 3.55, 3.57, 0.71, 1.03, 0.45, 3.19, 0.44, 3.32, 3.08, 3.77)
Experience <- c(4, 4, 1, 4, 2, 2, 2, 3, 0, 1, 3, 1, 3, 1, 2)
Interest <- c(1, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1)
```

Create a data frame

```
Students_data <- data.frame(
  StudentName = 1:length(GPAs),  # Generating a sequence of student numbers
  GPA = GPAs,
  Experience= Experience,  # Using Experience as Classes_attended for example purposes
  Interest = Interest     # Using Interest as Prior_Experience for example purposes
)
Students_data
```

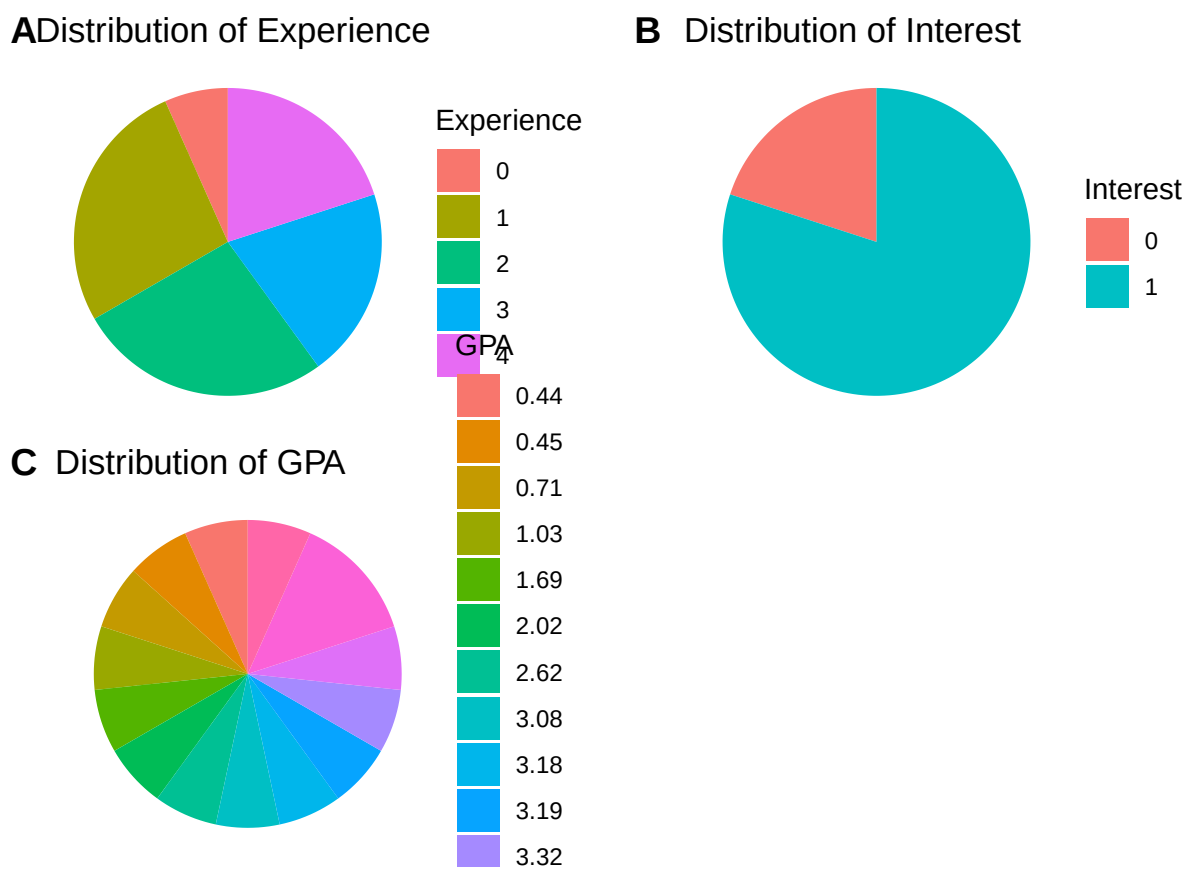
```
##    StudentName  GPA Experience Interest
## 1            1 3.57          4        1
## 2            2 3.18          4        1
## 3            3 2.02          1        0
## 4            4 1.69          4        0
## 5            5 2.62          2        0
## 6            6 3.55          2        1
## 7            7 3.57          2        1
## 8            8 0.71          3        1
## 9            9 1.03          0        1
## 10          10 0.45          1        1
## 11          11 3.19          3        1
## 12          12 0.44          1        1
## 13          13 3.32          3        1
## 14          14 3.08          1        1
## 15          15 3.77          2        1
```

```
Students_data %>%
  gather(key = "Variable", value = "Value", -StudentName) %>%
  ggplot(aes(x = factor(StudentName), y = Value, fill = Variable)) +
  geom_bar(stat = "identity", position = "dodge") +
  ggtitle("Combined Chart") +
  labs(fill = "Variable") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, vjust = 1))
```




```
library(ggplot2)
library(dplyr)
library(tidyr)
library(cowplot)

# Assuming your data is stored in a dataframe named 'Students_data'
Students_data <- data.frame(
  StudentName = 1:length(GPAs),  # Generating a sequence of student numbers
  GPA = GPAs,
  Experience = Experience,  # Using Experience as Classes_attended for example purposes
  Interest = Interest      # Using Interest as Prior_Experience for example purposes
)

# Function to generate pie charts
generate_pie_chart <- function(data, variable, title) {
  counts <- table(data[[variable]])
  df <- as.data.frame(counts)
  colnames(df) <- c(variable, "Count")

  pie <- ggplot(df, aes(x = "", y = Count, fill = factor(!!sym(variable)))) +
    geom_bar(stat = "identity", width = 1) +
    coord_polar("y", start = 0) +
    labs(title = title) +
    theme_void() +
    scale_fill_discrete(name = variable)
```

```
  return(pie)
}

# Create separate pie charts for Experience, Interest, and GPA
experience_pie <- generate_pie_chart(Students_data, "Experience", "Distribution of Experience")
interest_pie <- generate_pie_chart(Students_data, "Interest", "Distribution of Interest")
gpa_pie <- generate_pie_chart(Students_data, "GPA", "Distribution of GPA")

# Arrange pie charts together
combined_pie <- plot_grid(experience_pie, interest_pie, gpa_pie, labels = "AUTO")

# Display the combined pie chart
print(combined_pie)
```




```
# Define weights
weights <- c(weight_GPA = 0.5, Weight_Experience = 0.3, Weight_Interest = 0.2)
```

```
# calculate weighted average
weighted_average_data <- Students_data %>%
  mutate(weighted_avg = round((GPA/4) * weights["weight_GPA"] +
            (Experience/4) * weights["Weight_Experience"] +
            (Interest/1) * weights["Weight_Interest"], 2))

weighted_average_data
```

```
##    StudentName  GPA Experience Interest weighted_avg
## 1            1 3.57          4        1         0.95
## 2            2 3.18          4        1         0.90
## 3            3 2.02          1        0         0.33
## 4            4 1.69          4        0         0.51
## 5            5 2.62          2        0         0.48
## 6            6 3.55          2        1         0.79
## 7            7 3.57          2        1         0.80
## 8            8 0.71          3        1         0.51
## 9            9 1.03          0        1         0.33
## 10          10 0.45          1        1         0.33
## 11          11 3.19          3        1         0.82
## 12          12 0.44          1        1         0.33
## 13          13 3.32          3        1         0.84
## 14          14 3.08          1        1         0.66
## 15          15 3.77          2        1         0.82
```

According to the above results, we can identify some of the students get minus values and they are away from the average. Hence that our goal should be minimizing the variance of each student to see all of them around the weighted average value.

The objective function we can build up as follow. S refer for Student and G refer for Group.

Objective (Maximize) = 0.95S1G1 + 0.95S1G2 + 0.95S1G3 + 0.95S1G4 + 0.95S1G5 + 0.90S2G1 + 0.90S2G2 + 0.90S2G3 + 0.90S2G4 + 0.90S2G5 + 0.33S3G1 + 0.33S3G2 + 0.33S3G3 + 0.33S3G4 + 0.33S3G5 + 0.51S4G1 + 0.51S4G2 + 0.51S4G3 + 0.51S4G4 + 0.51S4G5 + 0.48S5G1 + 0.48S5D2 + 0.48S5G3 + 0.48S5G4 + 0.48S5G5 + 0.79S6G1 + 0.79S6G2 + 0.79S6G3 + 0.79S6G4 + 0.79S6G5 + 0.80S7G1 + 0.80S7G2 + 0.80S7G3 + 0.80S7G4 + 0.80S7G5 + 0.51S8G1 + 0.51S8G2 + 0.51S8G3 + 0.51S8G4 + 0.51S8G5 + 0.33S9G1 + 0.33S9G2 + 0.33S9G3 + 0.33S9G4 + 0.33S9G5 + 0.33S10G1 + 0.336S10G2 + 0.33S10G3 + 0.33S10G4 + 0.33S10G5 + 0.82S11G1 + 0.82S11G2 + 0.82S11G3 + 0.82S11G4 + 0.82S11G5 + 0.33S12G1 + 0.33S12G2 + 0.33S12G3 + 0.33S12G4 + 0.33S12G5 + 0.84S13G1 + 0.84S13G2 + 0.84S13G3 + 0.84S13G4 + 0.84S13G5 + 0.66S14G1 + 0.66S14G2 + 0.66S14G3 + 0.66S14G4 + 0.66S14G5 + 0.82S15G1 + 0.82S15G1 + 0.82S15G3 + 0.82S15G4 + 0.82S15G5

There are few constraints as follow,

1. Group constraints:The total number of students in each group must be 3. *Group 1: S1G1 + S2G1 + S3G1 + S4G1 + S5G1 + S6G1 + S7G1 + S8G1 + S9G1 + S10G1 + S11G1 + S12G1 + S13G1 + S14G1 + S15G1 = 3* Group 2: S1G2 + S2G2 + S3G2 + S4G2 + S5G2 + S6G2 + S7G2 + S8G2 + S9G2 + S10G2 + S11G2 + S12G2 + S13G2 + S14G2 + S15G2 = 3 *Group 3: S1G3 + S2G3 + S3G3 + S4G3 + S5G3 + S6G3 + S7G3 + S8G3 + S9G3 + S10G3 + S11G3 + S12G3 + S13G3 + S14G3 + S15G3 = 3* Group 4: S1G4 + S2G4 + S3G4 + S4G4 + S5G4 + S6G4 + S7G4 + S8G4 + S9G4 + S10G4 + S11G4 + S12G4 + S13G4 + S14G4 + S15G4 = 3 *Group 5: S1G5 + S2G5 + S3G5 + S4G5 + S5G5 + S6G5 + S7G5 + S8G5 + S9G5 + S10G5 + S11G5 + S12G5 + S13G5 + S14G5 + S15G5 = 3

2. Student constraints: Each student should be assigned to only one group. *S1G1 + S1G2 + S1G3 + S1G4 + S1G5 = 1* S2G1 + S2G2 + S2G3 + S2G4 + S2G5 = 1 *S3G1 + S3G2 + S3G3 + S3G4 + S3G5 = 1* S4G1 + S4S2 + S4G3 + S4G4 + S4G5 = 1 *S5G1 + S5G2 + S5G3 + S5G4 + S5G5 = 1* S6G1 + S6G2 + S6G3 + S6G4 + S6G5 = 1 *S7G1 + S7G2 + S7G3 + S7G4 + S7G5 = 1* S8G1 + S8G2 + S8G3 + S8G4 + S8G5 = 1 *S9G1 + S9G2 + S9G3 + S9G4 + S10G5 = 1* S10G1 + S10G2 + S10G3 + S10G4 + S10G5 = 1 *S11G1 + S11G2 + S11G3 + S11G4 + S11G5 = 1* S12G1 + S12G2 + S12G3 + S12G4 + S12G5 = 1 *S13G1 + S13G2 + S13G3 + S13G4 + S13G5 = 1* S14G1 + S14G2 + S14G3 + S14G4 + S14G5 = 1 *S15G1 + S15G2 + S15G3 + S15G4 + S15G5 = 1

3. Factor 1 constraint: GPA *Group 1: 3.57S1G1 + 3.18S2G1 + 2.02S3G1 + 1.69S4G1 + 2.62S5G1 + 3.55S6G1 + 3.57S7G1 + 0.71S8G1 + 1.03S9G1 + 0.45S10G1 + 3.19S11G1 + 0.44S12G1 + 3.32S13G1 + 3.08S14G1 + 3.77S15G1 => 2.41 (Average of Total GPA)* Group 2: 3.57S1G2 + 3.18S2G2 + 2.02S3G2 + 1.69S4G2 + 2.62S5G2 + 3.55S6G2 + 3.57S7G2 + 0.71S8G2 + 1.03S9G2 + 0.45S10G2 + 3.19S11G2 + 0.44S12G2 + 3.32S13G2 + 3.08S14G2 + 3.77S15G2 => 2.41 *Group 3: 3.57S1G3 + 3.18S2G3 + 2.02S3G3 + 1.69S4G3 + 2.62S5G3 + 3.55S6G3 + 3.57S7G3 + 0.71S8G3 + 1.03S9G3 + 0.45S10G3 + 3.19S11G3 + 0.44S12G3 + 3.32S13G3 + 3.08S14G3 + 3.77S15G3 => 2.41* Group 4: 3.57S1G4 + 3.18S2G4 + 2.02S3G4 + 1.69S4G4 + 2.62S5G4 + 3.55S6G4 + 3.57S7G4 + 0.71S8G4 + 1.03S9G4 + 0.45S10G4 + 3.19S11G4 + 0.44S12G4 + 3.32S13G4 + 3.08S14G4 + 3.77S15G4 => 2.41 *Group 5: 3.57S1G5 + 3.18S2G5 + 2.02S3G5 + 1.69S4G5 + 2.62S5G5 + 3.55S6G5 + 3.57S7G5 + 0.71S8G5 + 1.03S9G5 + 0.45S10G5 + 3.19S11G5 + 0.44S12G5 + 3.32S13G5 + 3.08S14G5 + 3.77S15G5 => 2.41

4.Factor 2 constraint: Experience *Group 1: 4S1G1 + 4S2G1 + 1S3G1 + 4S4G1 + 2S5G1 + 2S6G1 + 2S7G1 + 3S8G1 + 0S9G1 + 1S10G1 + 3S11G1 + 1S12G1 + 3S13G1 + 1S14G1 + 2S15G1 => 2.2 (Average of experience)* Group 2: 4S1G2 + 4S2G2 + 1S3G2 + 4S4G2 + 2S5G2 + 2S6G2 + 2S7G2 + 3S8G2 + 0S9G2 + 1S10G2 + 3S11G2 + 1S12G2 + 3S13G2 + 1S14G2 + 2S15G2 => 2.2 *Group 3: 4S1G3 + 4S2G3 + 1S3G3 + 4S4G3 + 2S5G3 + 2S6G3 + 2S7G3 + 3S8G3 + 0S9G3 + 1S10G3 + 3S11G3 + 1S12G3 + 3S13G3 + 1S14G3 + 2S15G3 => 2.2* Group 4: 4S1G4 + 4S2G4 + 1S3G4 + 4S4G4 + 2S5G4 + 2S6G4 + 2S7G4 + 3S8G4 + 0S9G4 + 1S10G4 + 3S11G4 + 1S12G4 + 3S13G4 + 1S14G4 + 2S15G4 => 2.2 *Group 5: 4S1G5 + 4S2G5 + 1S3G5 + 4S4G5 + 2S5G5 + 2S6G5 + 2S7G5 + 3S8G5 + 0S9G5 + 1S10G5 + 3S11G5 + 1S12G5 + 3S13G5 + 1S14G5 + 2S15G5 => 2.2

5.Factor 3 constraint: Interest *Group 1: 1S1G1 + 1S2G1 + 0S3G1 + 0S4G1 + 0S5G1 + 1S6G1 + 1S7G1 + 1S8G1 + 1S9G1 + 1S10G1 + 1S11G1 + 1S12G1 + 1S13G1 + 1S14G1 + 1S15G1 => 0.8 (Average of Interest)* Group 2: 1S1G2 + 1S2G2 + 0S3G2 + 0S4G2 + 0S5G2 + 1S6G2 + 1S7G2 + 1S8G2 + 1S9G2 + 1S10G2 + 1S11G2 + 1S12G2 + 1S13G2 + 1S14G2 + 1S15G2 => 0.8 *Group 3: 1S1G3 + 1S2G3 + 0S3G3 + 0S4G3 + 0S5G3 + 1S6G3 + 1S7G3 + 1S8G3 + 1S9G3 + 1S10G3 + 1S11G3 + 1S12G3 + 1S13G3 + 1S14G3 + 1S15G3 => 0.8* Group 4: 1S1G4 + 1S2G4 + 0S3G4 + 0S4G4 + 0S5G4 + 1S6G4 + 1S7G4 + 1S8G4 + 1S9G4 + 1S10G4 + 1S11G4 + 1S12G4 + 1S13G4 + 1S14G4 + 1S15G4 => 0.8 *Group 5: 1S1G5 + 1S2G5 + 0S3G5 + 0S4G5 + 0S5G5 + 1S6G5 + 1S7G5 + 1S8G5 + 1S9G5 + 1S10G5 + 1S11G5 + 1S12G5 + 1S13G5 + 1S14G5 + 1S15G5 => 0.8

Non negativity constraint

S,G>0 where S=1,2,3,4,5,6,7,8,9,10,11,12,13,14,15 and G=1,2,3,4,5

calculate average GPA

```
average_GPA<- mean(Students_data$GPA)
average_GPA
```

```
## [1] 2.412667
```

Caculate average experience

```r
average_Experience<- mean(Students_data$Experience)
average_Experience
```

```
## [1] 2.2
```

caculate average interest

```r
average_Interest<- mean(Students_data$Interest)
average_Interest
```

```
## [1] 0.8
```

Setting the lp model

```r
lp_model <- make.lp(0,75)
set.objfn(lp_model,c(0.95,0.95,0.95,0.95,0.95,
                     0.90,0.90,0.90,0.90,0.90,
                     0.33,0.33,0.33,0.33,0.33,
                     0.51,0.51,0.51,0.51,0.51,
                     0.48,0.48,0.48,0.48,0.48,
                     0.79,0.79,0.79,0.79,0.79,
                     0.80,0.80,0.80,0.80,0.80,
                     0.51,0.51,0.51,0.51,0.51,
                     0.33,0.33,0.33,0.33,0.33,
                     0.33,0.33,0.33,0.33,0.33,
                     0.82,0.82,0.82,0.82,0.82,
                     0.33,0.33,0.33,0.33,0.33,
                     0.84,0.84,0.84,0.84,0.84,
                     0.66,0.66,0.66,0.66,0.66,
                     0.82,0.82,0.82,0.82,0.82))
lp.control(lp_model, sense = 'max', all.bin= TRUE)
```

```
## $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"
```

Setting constrains in lp model

```
add.constraint(lp_model, c(1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,
add.constraint(lp_model, c(0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,
add.constraint(lp_model, c(0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,
add.constraint(lp_model, c(0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,
add.constraint(lp_model, c(0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,
add.constraint(lp_model, c(1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
add.constraint(lp_model, c(0,0,0,0,0,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
add.constraint(lp_model, c(0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
add.constraint(lp_model, c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
add.constraint(lp_model, c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,
add.constraint(lp_model, c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,0,0,0,0,0,0,0,0,
add.constraint(lp_model, c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,0,0,0,
add.constraint(lp_model, c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,
add.constraint(lp_model, c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
add.constraint(lp_model, c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
add.constraint(lp_model, c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
add.constraint(lp_model, c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
add.constraint(lp_model, c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
add.constraint(lp_model, c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
add.constraint(lp_model, c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
add.constraint(lp_model, c(1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,
add.constraint(lp_model, c(0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,
add.constraint(lp_model, c(0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,
add.constraint(lp_model, c(0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,
add.constraint(lp_model, c(0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,
add.constraint(lp_model, c(1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,
add.constraint(lp_model, c(0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,
add.constraint(lp_model, c(0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,
add.constraint(lp_model, c(0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,
add.constraint(lp_model, c(0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,
add.constraint(lp_model, c(1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,
add.constraint(lp_model, c(0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,
add.constraint(lp_model, c(0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,
add.constraint(lp_model, c(0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,
add.constraint(lp_model, c(0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,
```

Solving lp model

```
solve(lp_model)
```

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

```
get.objective(lp_model)
```

```
## [1] 9.4
```

```
get.variables(lp_model)
```

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

identify the students in groups

```
#Group-1
get.variables(lp_model)[1:15]
```

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

```
#Group-2
get.variables(lp_model)[16:30]
```

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

```
#Group-3
get.variables(lp_model)[31:45]
```

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

```
#Group-4
get.variables(lp_model)[46:60]
```

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

```
#Group-5
get.variables(lp_model)[61:75]
```

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

Conclusion

Group 1: Student 6, Student 11, Student 15 Group 2: Student 7, Student 8, Student 14 *Group 3: Student 1, Student 2, Student 13* Group 4: Student 4, Student 5, Student 12 *Group 5: Student 3, Student 9, Student 10