

Assignment 1

Xiaoshu Gui

1/24/2019

```
# Load packages
library(dplyr)
library(purrr)
library(magrittr)
library(tidyr)
library(tibble)
library(stringr)
library(reshape2)

# Read data
jss <- read.csv("datjss.csv", header=T, sep=",")
sss <- read.csv("datsss.csv", header=T, sep=",")
stu <- read.csv("datstu.csv", header=T, sep=",", na.strings=c("", " ", "NA"))
# Note that there are empty cells in stu, we replace them with NAs first.
```

Exercise 1

- Number of students: 340823

```
length(unique(stu$X))
```

```
## [1] 340823
```

- Number of schools: 898

```
length(unique(sss$schoolcode))
```

```
## [1] 898
```

```
# clean sss
ss <- sss %>%
  select(-X) %>%
  mutate(schoolname = str_remove_all(schoolname, "\\d")) %>% # clean school names
  mutate(schoolname = str_remove_all(schoolname, "\\W")) %>%
  distinct(schoolcode, .keep_all = T) # Note that some school names do not match their
# school code, I will only use school codes as id variable.
length(unique(ss$schoolcode))
```

```
## [1] 898
```

- Number of programs: 32

```
pgm <- stu %>%
  select(starts_with("choice")) %>% # get a subset of stu (choicepgm1 ~ 6)
  map(., function(x) unique(x)) %>% # find the number of programs in each "choicepgm"
  unlist %>%
  unique() %>% # join all the unique programs in 6 choices and find the unique value.
  length()
pgm
```

```
## [1] 33
```

Note that there's NAs in choicepgm, so the number of programs is: $33 - 1 = 32$

- Number of choices (school, program):

```
school <- stu %>%
  select(X:schoolcode6) %>%
  gather(key = 'school', 'schoolcode', -c(X:male)) %>%
  mutate(choice = str_match(school, "[1-6]") %>% as.numeric())

pgm <- stu %>%
  select(X, choicepgm1:rankplace) %>%
  gather(key = 'program', 'choicepgm', -c(X, jssdistrict, rankplace)) %>%
  mutate(choice = str_match(program, "[1-6]") %>% as.numeric())

stu1 <- left_join(school, pgm, by = c("X", "choice")) %>%
  drop_na(., c(schoolcode, choicepgm)) %>%
  mutate(choice_sp = paste(schoolcode, choicepgm, sep = ","))
# number of all choices of school and program:
length(stu1$choice_sp)
```

```
## [1] 2006470
```

```
# number of unique choices of school and program
length(unique(stu1$choice_sp))
```

```
## [1] 2773
```

- Missing test score: 179887 (number of NAs in “score” in datstu)
- Apply to the same school (different programs)

```
stu2 <- stu1 %>%
  group_by(schoolcode) %>%
  summarise(count= n())
# The table presents the number of students apply to the same school (dif programs)
stu2
```

```
## # A tibble: 640 x 2
##   schoolcode count
##   <int> <int>
## 1    10101  5891
## 2    10102  1958
## 3    10103  8419
## 4    10104  2474
## 5    10105  1496
## 6    10106  4015
## 7    10107  4075
## 8    10108  4181
## 9    10109  8995
## 10   10110  3017
## # ... with 630 more rows
```

- Apply to less than 6 choices: 20988

```
# rule out NAs in choicepgm, leaving us students who apply 6 choices
stu3 <- drop_na(stu, starts_with("choice"))
# then students who apply less than 6 choices should be total number of students minus
```

```
# those who apply 6 choices:  
340823 - count(stu3)
```

```
##           n  
## 1 20988
```

Excercise 2

The school level dataset is presented by 'stu5'

```
df <- stu1 %>%  
  select(X, choice_sp, schoolcode, score, rankplace)  
  
# merge ss with df,  
stu4 <- left_join(df, ss, by = "schoolcode")  
  
admin <- stu4 %>%  
  filter(!is.na(rankplace)) %>%  
  filter(rankplace < 7) %>% # get a subset of admitted students  
  group_by(choice_sp) %>%  
  summarise(cutoff = min(score), quality = mean(score), size = n())  
  
stu5 <- left_join(stu4, admin, by = "choice_sp") %>%  
  select(choice_sp, sssdistrict:size) %>%  
  distinct() # the dataset
```

Excercise 3 Distance

```
distance <- stu %>%  
  select(X, schoolcode1:schoolcode6, jssdistrict)
```