615 Midterm

Jennifer Mo (w/ the BU Healthy Minds team)

December 12, 2016

For this project, I am working with the BU Healthy Minds survey data to extract a specific subgroup of people. More specifically, people who are not getting professional mental health support but should be. This group of people is called the help gap and they will be selected based on their self-reported perceieved need as well as their belief of the efficacy of therapy.

```
require(Base)
```

```
## Loading required package: Base
```

```
## Warning in library(package, lib.loc = lib.loc, character.only = TRUE,
## logical.return = TRUE, : there is no package called 'Base'
```

```
require(ggplot2)
```

```
## Loading required package: ggplot2
```

```
require(ggvis)
```

```
## Loading required package: ggvis
```

```
##
## Attaching package: 'ggvis'
```

```
## The following object is masked from 'package:ggplot2':
##
##     resolution
```

```
require(dplyr)
```

```
## Loading required package: 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
```

```
require(stringr)
```

```
## Loading required package: stringr
```

```
require(tidyr)
```

```
## Loading required package: tidyr
```

```
require(data.table)
```

```
## Loading required package: data.table
```

```
## -------------------------------------------------------------------------
```

```
## data.table + dplyr code now lives in dtplyr.
## Please library(dtplyr)!
```

```
## -------------------------------------------------------------------------
```

```
## 
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:dplyr':
## 
##     between, last
```

```
require(ggthemes)
```

```
## Loading required package: ggthemes
```

```
require(gridExtra)
```

```
## Loading required package: gridExtra
```

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

```
library(formattable)
require(plyr)
```

```
## Loading required package: plyr
```

```
## ------------------------------------------------------------------------
```

```
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
```

```
## ------------------------------------------------------------------------
```

```
## 
## Attaching package: 'plyr'
```

```
## The following objects are masked from 'package:dplyr':
## 
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

Here please enter the name of the csv file that you wish to clean:

```
load("hms.rda")
```

The csv file has now been loaded as a rda file into the working directory

```
#1. race

race <- hms[15:23]
text <- unique(race$race_other_text)
text_asian_indicator <- c(3,9,14,15,16,18,21,29,32,39,42,45,50,58,68,73,75,76,79,80)
text_biracial_indicator <- c(24, 30,40,41,44,55,74)
text_black_indicator <- c(6)
text_asian <- text[text_asian_indicator]
text_biracial <- text[text_biracial_indicator]
text_black <- text[text_black_indicator]

race$race_cleaned[race$race_white == 1] <- "Caucasian"
race$race_cleaned[race$race_asian == 1 | race$race_other_text %in% text_asian] <- "Asian"
race$race_cleaned[race$race_black == 1 | race$race_other_text %in% text_black] <- "African Ameri
can"
race$race_cleaned[rowSums(race[1:7], na.rm = TRUE) >=2 | race$race_other_text %in%
text_biracial] <- "Biracial"
race$race_cleaned[race$race_ainaan == 1] <- "Native American"
race$race_cleaned[race$race_mides == 1] <- "Middle Eastern"
race$race_cleaned[race$race_ainaan == 1 | race$race_mides == 1] <- "Other"
race$race_cleaned[is.na(race$race_cleaned) & race$race_other == 1] <- "Other"

# 2. academic status + year of school
educ <- hms[41:50]
educ$aca_status[educ$degree_bach == 1] <- "undergraduate"
educ$aca_status[rowSums(educ[2:5], na.rm = TRUE) >= 1] <- "graduate"

# dealing with text column
other <- educ %>% filter(degree_other == 1 & !is.na(degree_other_text))
other_text <- unique(other$degree_other_text)
other_text_indicator <- c(T, T, T, F, F, F, T, T, T, T, T, T, T, T, T, T, F, F, F, T, T, T, F,
F, T, F, F, F, T, T, F)
grad <- other_text[other_text_indicator]
educ$aca_status[educ$degree_other_text %in% grad] <- "graduate"
educ$aca_status[is.na(educ$aca_status) & educ$degree_other == 1 &
!is.na(educ$degree_other_text)] <- "other"

# year of school
educ$yr_sch[educ$aca_status == "graduate"] <- NA
educ$yr_sch[educ$yr_sch == 1] <- "freshman"
educ$yr_sch[educ$yr_sch == 2] <- "sophomore"
educ$yr_sch[educ$yr_sch == 3] <- "junior"
educ$yr_sch[educ$yr_sch == 4 | educ$yr_sch == 5] <- "senior"




# 3. gender
gender <- hms$gender
#didn't include gender_text because the column is all NAs
gender[gender == 1] <- "male"
gender[gender == 2] <- "female"
gender[gender == 3 | gender == 4] <- "trans gender"
gender[gender == 5 | gender == 6] <- "other"
```

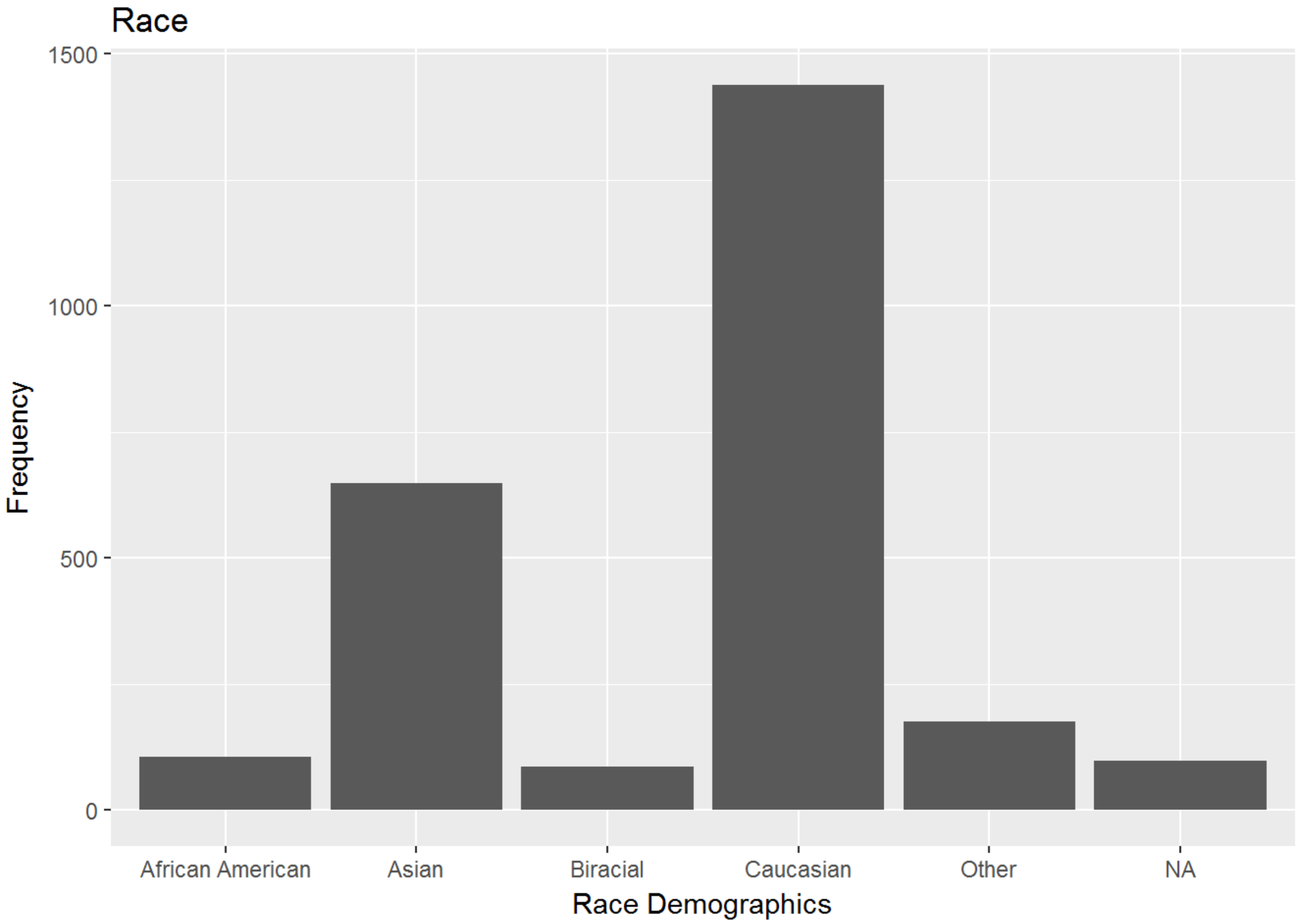
```
# 4. citizenship
citizenship <- hms$citizen
citizenship[citizenship == 1] <- "domestic"
citizenship[citizenship == 0] <- "international"

# 5.field of study
df_field <- hms[53:73]

#df_field$rowSum <- rowSums(df_field[, 1:20], na.rm = T)
#table(df_field$rowSum)

fieldls <- c(
  'Humanities',
  'Natural sciences/math',
  'Social sciences',
  'Architecture/urban planning',
  'Art & design',
  'Business',
  'Dentistry',
  'Education',
  'Engineering',
  'Law',
  'Medicine',
  'Music/theatre/dance',
  'Nursing',
  'Pharmacy',
  'Pre-professional',
  'Public health',
  'Public policy',
  'Social Work',
  'Undecided',
  'Other'
)
tmp1 <- apply(df_field[,1:20], 1, function(x) which(!is.na(x)))
tmp2 <- lapply(tmp1, function(x) x[1])
pos <- unlist(tmp2)
df_field$field <- fieldls[pos]


demographics <- data.frame(citizenship, gender, educ$aca_status, educ$yr_sch, race$race_cleaned,
 df_field$field)
names(demographics) <- c("citizenship", "gender", "academic_status", "year_of_school", "race",
"field")

save(demographics, file = "demographics.rda")
```

Here the code is seperating the non-demographics data into different data frames. That way, we can merge sections that we are interested in into different data frames to make working with this dataset easier.

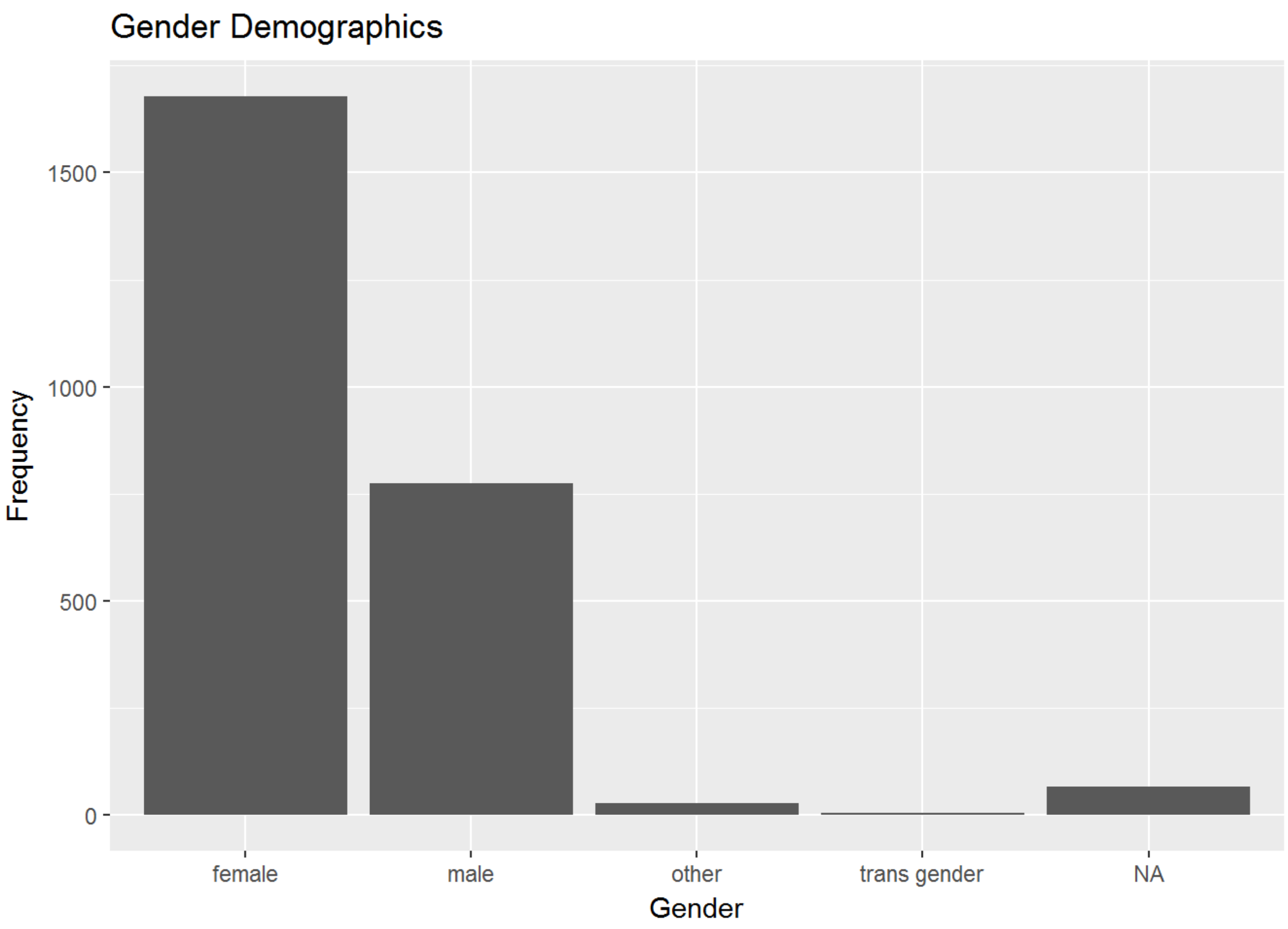
```
row.names(hms) <- hms$responseid

key <- paste("df",unlist(lapply(strsplit(names(hms), split="_"),function(x){x[1]})),sep = "_")
# function to create data.frame
pull <- function(x){
  df = as.data.frame(hms[,grep(x,key)])
  colnames(df) = names(hms)[grep(x,key)]
  rownames(df) = rownames(hms)  # if not defined, single column section will lose row names.
  assign(x, df, envir=.GlobalEnv)
}

invisible(lapply(unique(key),pull))


save.image("split_data.RData")
```

Here’s some EDA to take a look at the overall demographics of the survey responders.

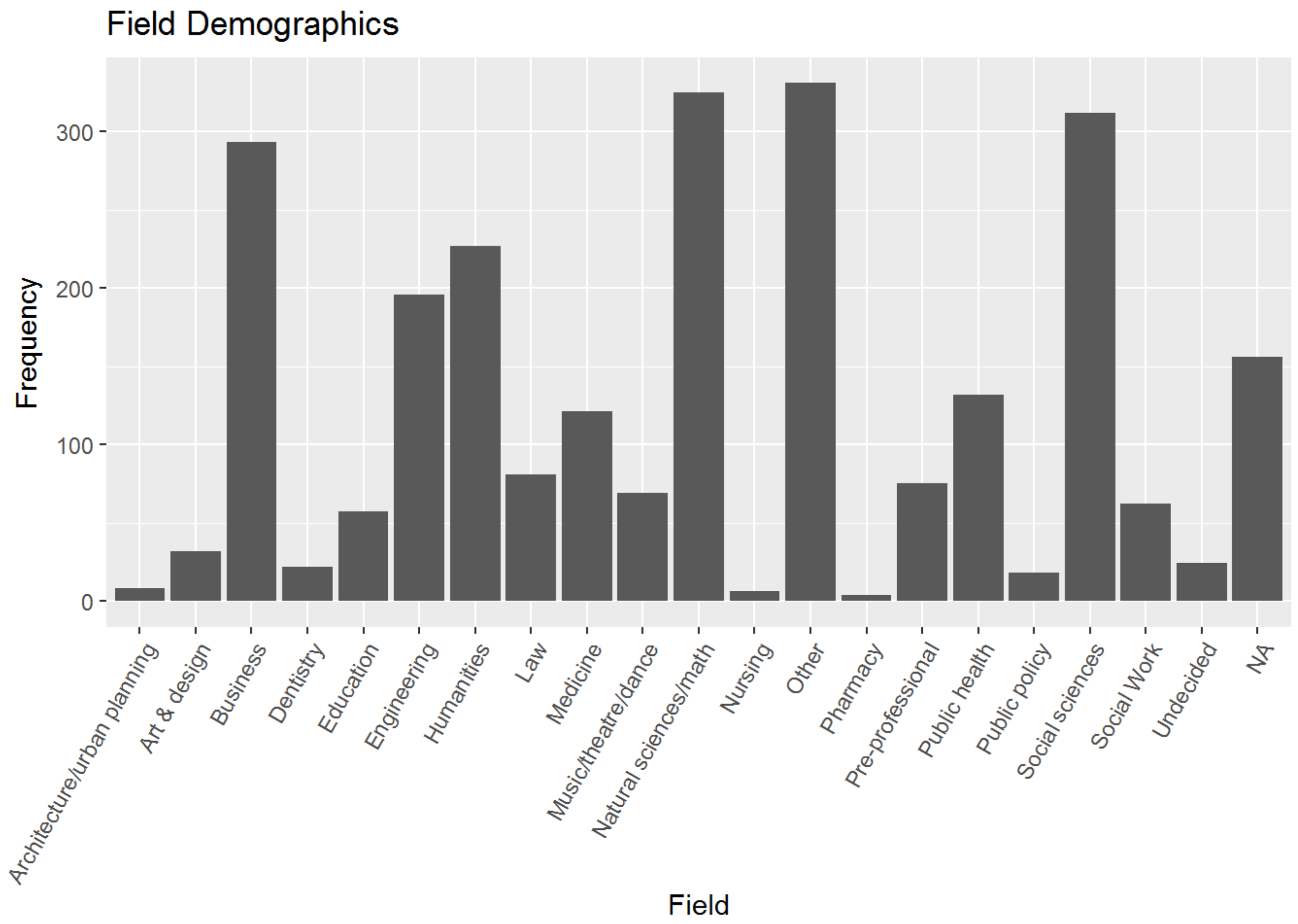
```
#race demographics

demo_race <- data.frame(count(demographics,'race'))
ggplot(demo_race,aes(x=race,y=freq))+geom_bar(stat="identity")+ggtitle("Race")+ylab("Frequency")+x
ab("Race Demographics")
```



```
#gender demographics
demo_gender <- data.frame(count(demographics, 'gender'))
ggplot(demo_gender, aes(x=gender, y=freq)) + geom_bar(stat="identity") + ggtitle("Gender Demographics")+ ylab("Frequency")+ xlab("Gender")
```



```
#field demographics

demo_field <- data.frame(count(demographics,'field'))
ggplot(demo_field, aes(x=field, y=freq)) + geom_bar(stat="identity") + ggtitle("Field Demographics")+ ylab("Frequency")+ xlab("Field") + theme(axis.text.x = element_text(angle=60,hjust=1))
```

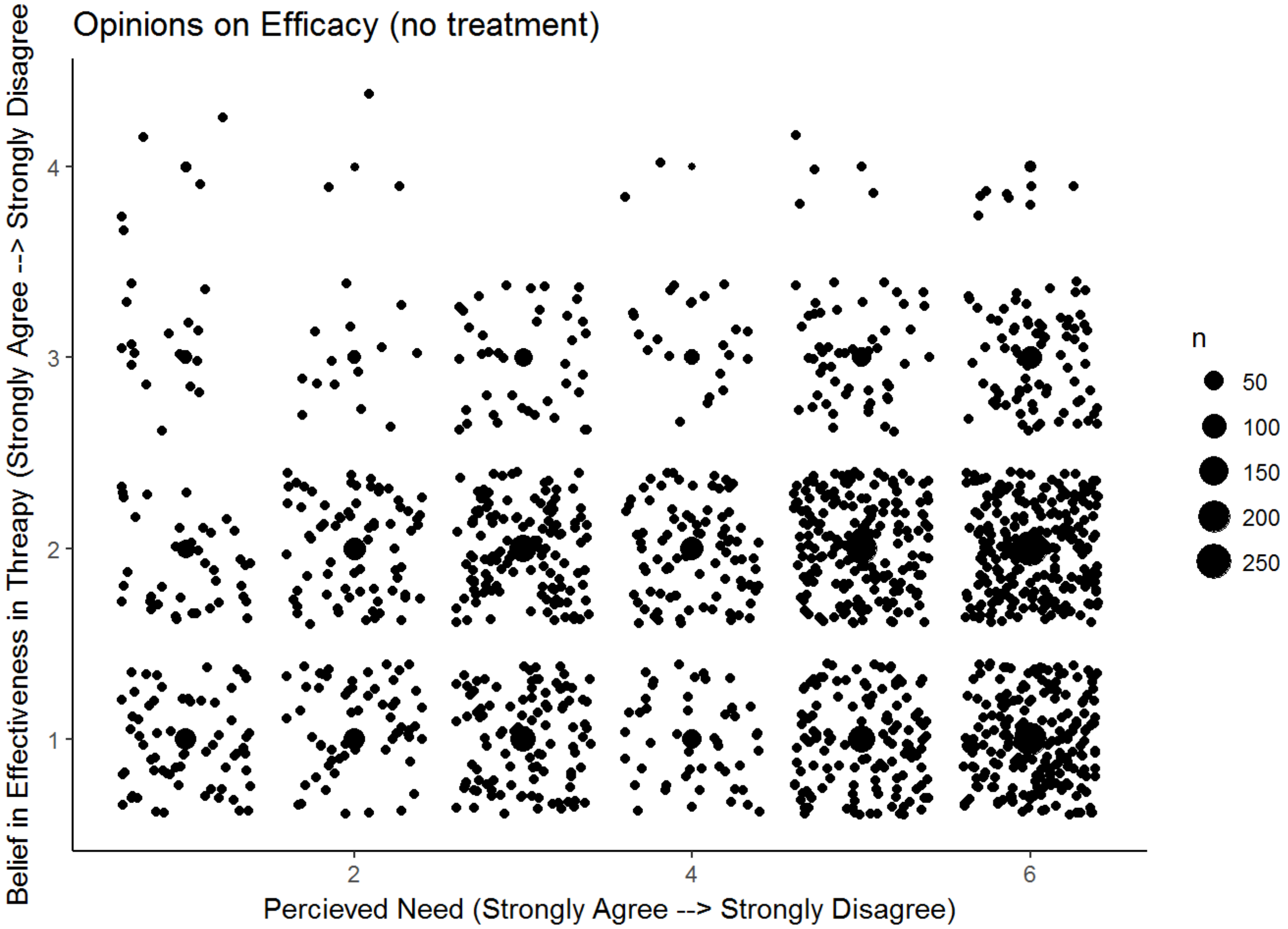


As an exmaple, if we are interested in looking at people who has never been to therapy, believes in the effectiveness of therapy and also has a self- reported perceived need for such treatment, we first need to create a data frame with the relevent variables.

```
efficacy <- data.frame(df_responseid,demographics,df_ther$ther_any,df_ther$ther_help,df_med$med_help,df_percneed, df_bar$bar_ns_1, df_bar$bar_ns_2, df_bar$bar_ns_3, df_bar$bar_ns_4, df_bar$bar_ns_5,df_bar$bar_ns_6, df_bar$bar_ns_7,df_bar$bar_ns_8,df_stig)

efficacy1<-(filter(efficacy,df_ther$ther_any == 0)) #1688 responders

efficacy1$count<-rep(1)

ggplot(efficacy1, aes(x=efficacy1$percneed,y=efficacy1$df_ther.ther_help))+geom_count()+geom_jitter()+xlab("Percieved Need (Strongly Agree --> Strongly Disagree)")+ylab("Belief in Effectiveness in Threapy (Strongly Agree --> Strongly Disagree")+ggtitle("Opinions on Efficacy (no treatment)")+theme_classic()
```

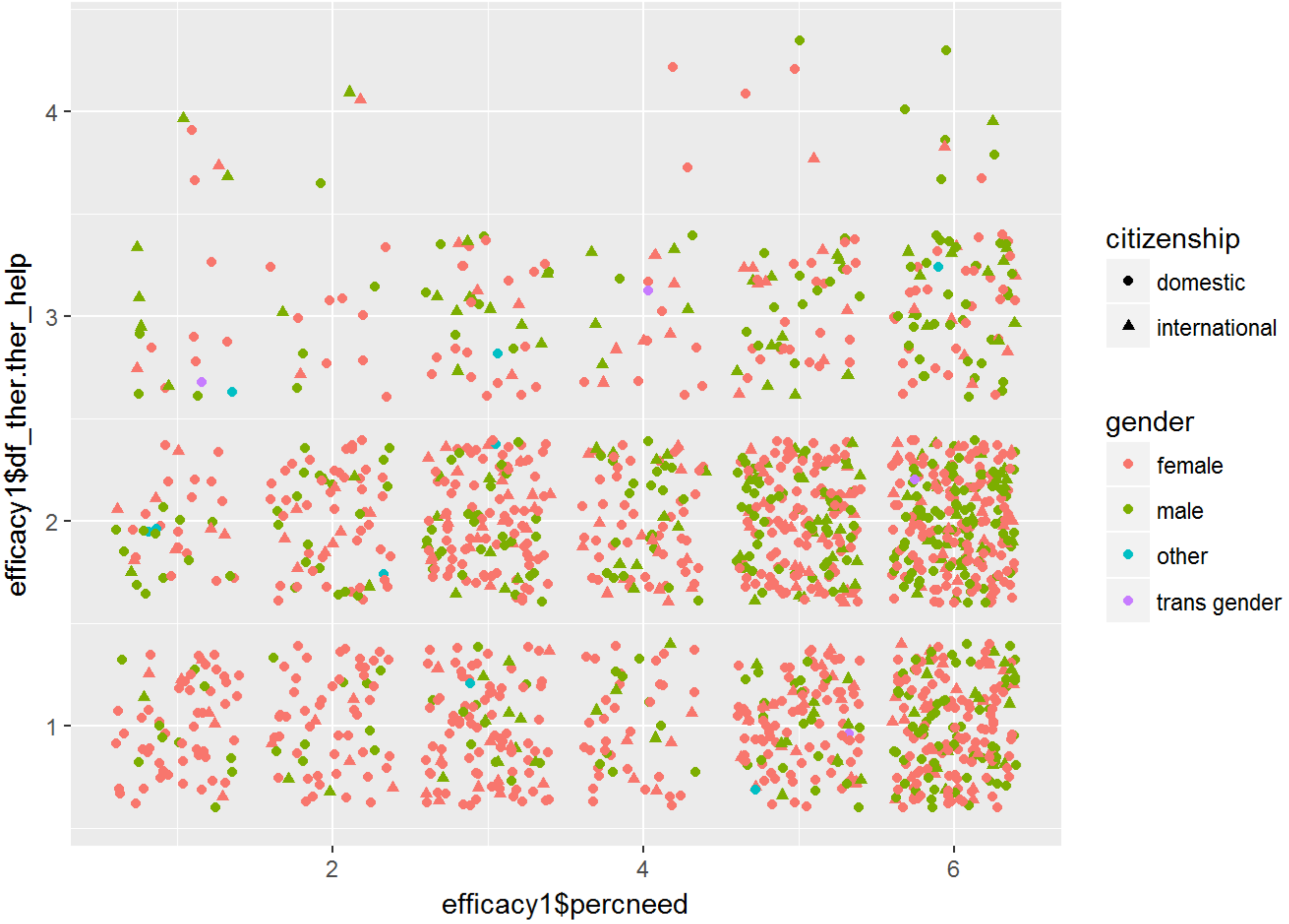
```
## Warning: Removed 36 rows containing non-finite values (stat_sum).
```

```
## Warning: Removed 36 rows containing missing values (geom_point).
```



```
ggplot(efficacy1, aes(x=efficacy1$percneed,y=efficacy1$df_ther.ther_help,color=gender,shape=citi
zenship))+geom_jitter()
```

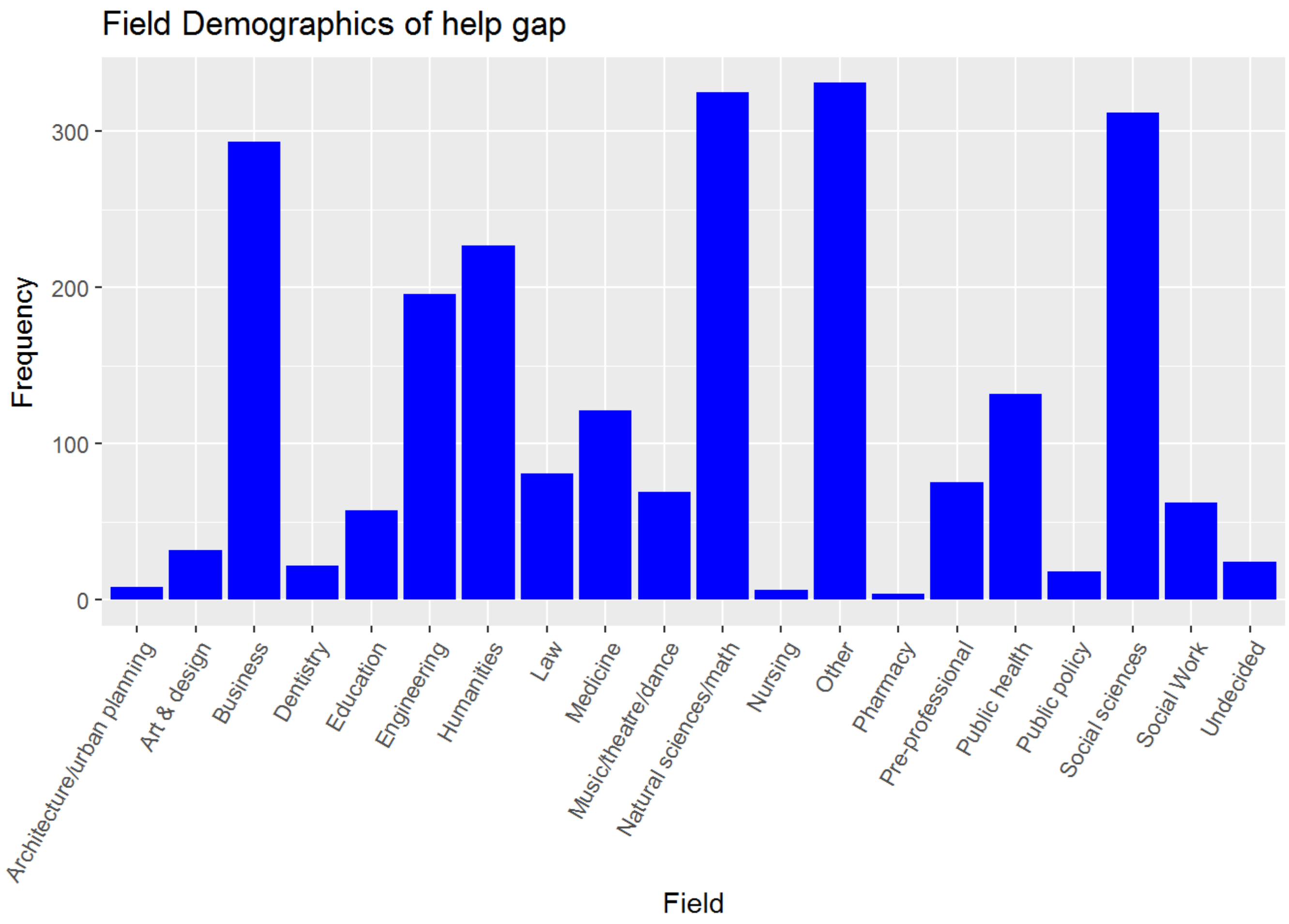
```
## Warning: Removed 36 rows containing missing values (geom_point).
```



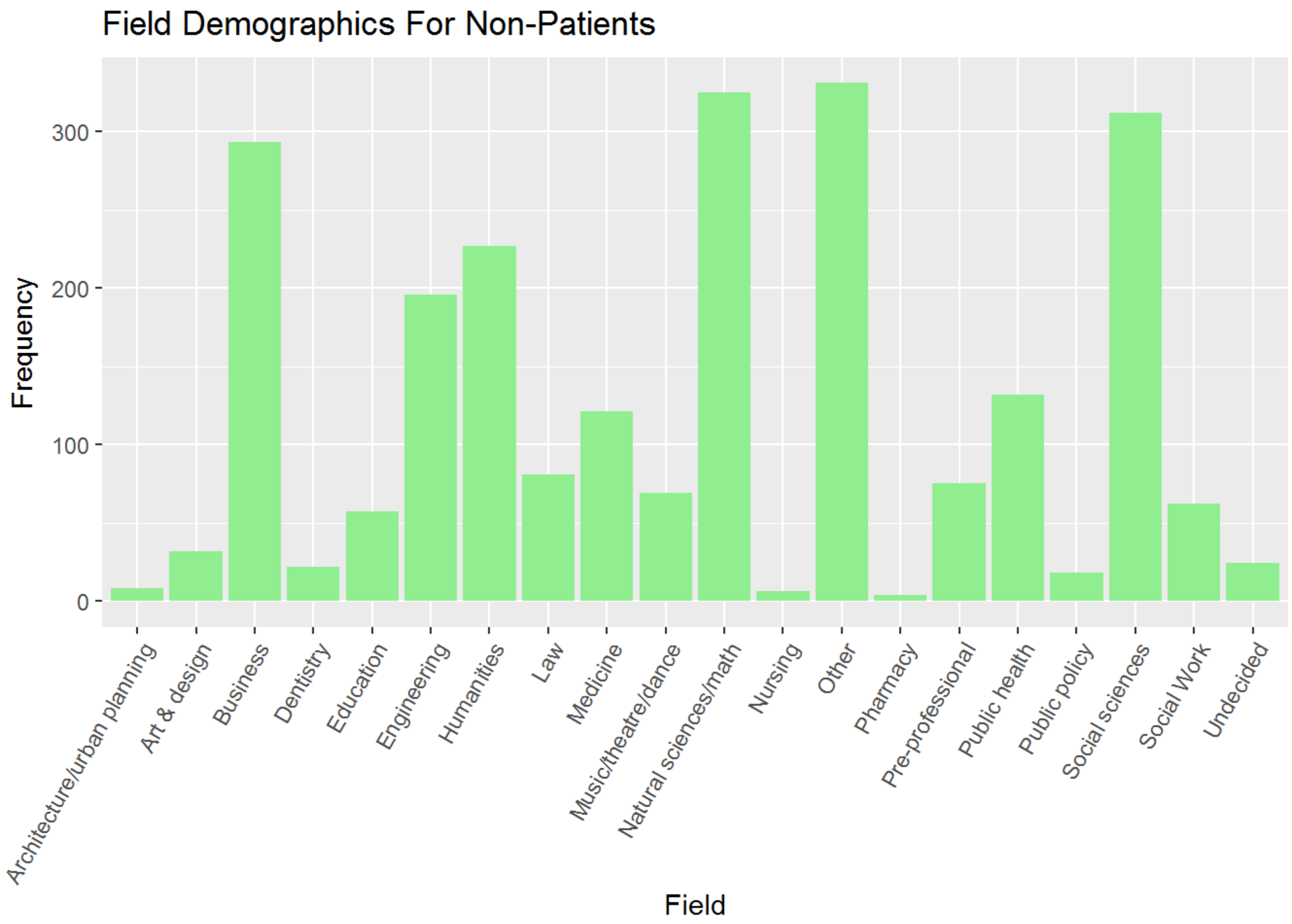
The first filter will be for people who have never been to therapy in the data table, “efficacy1”. From the first jittered plot, we can see where the concentration of people are. Logically those who has a low perceived need and those who do not believe in the efficacy of therapy are groups of people are not within our target group. We will be focusing on the people who are on the upper left corner who believe in the efficacy of therapy and also have a high perceved need. Additionally, the 2nd group has the gender color coded and the shape of the point on the graph indicates citizenship. Sometimes such techniques are used to see and obvious patterns and in this case, it looks like there are none.

Our next step, we will be filtering for people with high perceived need and high belief in efficacy from the group of people that has already been filter for never having gone to therapy. In total, there are 471 responders that fits into this group.

```
help.per<-filter(efficacy1,efficacy1$percneed <=3 & efficacy1$df_ther.ther_help < 3)

#471 total number of responders

help_field <- data.frame(count(help.per,'field'))
help_field <- na.omit(demo_field)
ggplot(help_field, aes(x=help_field$field,y=help_field$freq))+geom_bar(stat="identity",fill='blu
e') + ggtitle("Field Demographics of help gap")+ ylab("Frequency")+ xlab("Field") + theme(axis.t
ext.x = element_text(angle=60,hjust=1))
```



```
eff_field <- data.frame(count(efficacy1,'field'))
eff_field<- na.omit(demo_field)

ggplot(eff_field, aes(x=eff_field$field,y=eff_field$freq))+geom_bar(stat="identity",fill='lightgreen') + ggtitle("Field Demographics For Non-Patients")+ ylab("Frequency")+ xlab("Field") + theme(axis.text.x = element_text(angle=60,hjust=1))
```



```
save(help.per, file = "helpGap.rda")
```

Help.per is the resulting dataset. Now anyone with the Rda file can upload the cleaned file and start coding for more EDA such as the previous 2 graphs displaying responders by their respective fields.