DATA ANALYSIS OF FACEBOOK

CODE

```
#Analyzing pseudo_facebook dataset
#load required libraries
library(ggplot2)
library(gridExtra)
library(reshape2)
library(GGally)
df_fb <- read.csv("D:/ITU/R-Language/MyProject/pseudo_facebook.txt", sep = "\t")
df_fb$dob_month <- as.factor(df_fb$dob_month)

qplot(x = df_fb$dob_day , data=df_fb) +
  scale_x_discrete(breaks = 1:31) +
  facet_wrap(~dob_month, ncol = 3)

# histogram of friend count
qplot(x = df_fb$friend_count, data = df_fb, xlim = c(0,1000))

#friend count histogram, faceted by gender
qplot(x = df_fb$friend_count, data=df_fb, binwidth = 25)+
  scale_x_continuous(limits = c(0,1000), breaks = seq(0,1000,50)) +
  facet_wrap(~gender)

# applying limits to x
qplot(x = df_fb$friend_count, data = df_fb,xlim = c(0,1000),
      xlab = seq(0,1000,50), binwidth = 25)+
  facet_wrap(~gender)

#histogram of tenure on facebook; tenure denotes number of days; tenure/365 --> number of years
qplot(x = df_fb$tenure/365, data = df_fb, binwidth = 0.25,
      color = I("black"), fill = I("#0000FF"))+
  scale_x_continuous(breaks = seq(0,7,1), limits= c(0,7))

#histogram of age of all facebook users
qplot(x = df_fb$age, data = df_fb, binwidth = 1,
      xlab = "Age of facebook users",
      ylab = "Number of users",
      color = I("black"), fill = I("#0000FF"))+
  scale_x_continuous(breaks = seq(10,50,1), limits= c(10,50))

library(gridExtra)
summary(log(df_fb$friend_count))

# using grid.arrange
# histograms of friend counts using different scales
g1 = qplot(x = df_fb$friend_count, data = df_fb, binwidth = 10, xlim = c(0,1000), xlab = seq(0,1000,50))
```

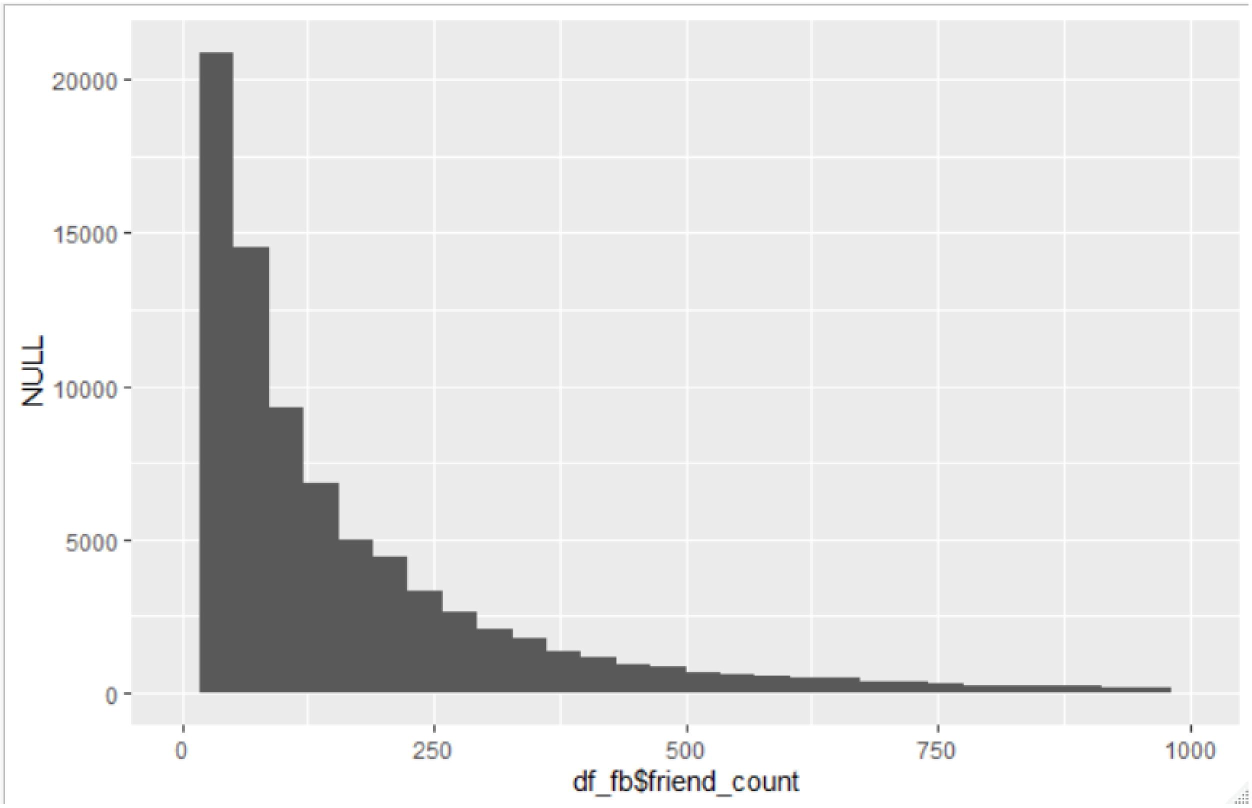
```
g2 = qplot(x = log10(df_fb$friend_count+1), data = df_fb, binwidth = 1, xlim = c(0,10), xlab = seq(0,10,1))

g3 = qplot(x = sqrt(df_fb$friend_count), data = df_fb, binwidth = 1, xlim = c(0,1000), xlab =
seq(0,1000,50))

grid.arrange(g1,g2,g3)

# histogram of friend count vs relative count, colored by gender
qplot(x = df_fb$friend_count, y = ..count../sum(..count..), data=df_fb,
      binwidth = 5, geom = "freqpoly", color = gender) +
  scale_x_continuous(limits = c(0,1000), breaks = seq(0,1000,10))

# relative count of www_likes, colored by gender
qplot(x = df_fb$www_likes, y = ..count../sum(..count..), data=df_fb,
      binwidth = 20, geom = "freqpoly", color = gender) +
  scale_x_continuous(limits = c(0,300), breaks = seq(0,300,5))

# count of www_likes, with a log10 as x scale
qplot(x = df_fb$www_likes, data=df_fb,
      geom = "freqpoly", color = gender) +
  scale_x_continuous()+
  scale_x_log10()


# subset of facebook dataframe, with only data about males
df_males <- df_fb[df_fb$gender == "male",]
str(df_males)

# subset of facebook dataframe, with data about females
df_females <- df_fb[df_fb$gender == "female",]

# male vs female- www likes (likes received when signed in from a computer)
p1 = qplot(x = df_males$www_likes, data = df_males)

p2 = qplot(x = df_females$www_likes, data = df_females)

grid.arrange(p1,p2)


male_count <- 24.42 * (nrow(df_males)-176)

female_count <- 87.14 * (nrow(df_females)-176)

by(df_fb$www_likes, df_fb$gender, sum)

# boxplot of friend count vs gender
qplot(x = df_fb$gender, y = df_fb$friend_count, data = df_fb, geom = "boxplot")+
  scale_y_continuous(limits = c(0,1000), breaks = seq(0,1000,50))

# boxplot of friend count vs gender, subsetting dataframe to exclude NAs
qplot(x = gender, y = friend_count, data = subset(df_fb, !is.na(gender)), geom = "boxplot")+
  scale_y_continuous(limits = c(0,1000), breaks = seq(0,1000,50))

# boxplot of friend count vs gender, adjusting Y values
qplot(x = gender, y = friend_count, data = subset(df_fb, !is.na(gender)), geom = "boxplot") +
  coord_cartesian(ylim = c(0,1000))

# boxplot of friendships initiated by males and females
qplot(x = gender, y = friendships_initiated, data = subset(df_fb, !is.na(gender)), geom = "boxplot") +
  coord_cartesian(ylim = c(0,250))


by(df_fb$friendships_initiated, df_fb$gender,summary)

# creating column for mobile check ins based on mobile likes
df_fb$mobile_checkin <- 0
df_fb$mobile_checkin[df_fb$mobile_likes > 0] <- 1
df_fb$mobile_checkin <- as.numeric(df_fb$mobile_checkin)

summary(df_fb$mobile_checkin)

library(plyr)
checkins <- count(df_fb$mobile_checkin[df_fb$mobile_checkin == 1])
total <- nrow(df_fb)
percentage <- checkins*100/total


library(ggplot2)
#aes wrapper
# histogram of age vs friend count
ggplot(aes(x = age, y = friend_count),data = df_fb)+
  geom_point(alpha = 1/20)+
  xlim(13,90)

# using geom_jitter to reduce overplotting
ggplot(aes(x = age, y = friend_count),data = df_fb)+
  geom_jitter(alpha = 1/20)+
  xlim(13,90)

# scatterplot of age vs friend count
ggplot(aes(x = age, y = friend_count),data = df_fb)+
  geom_point(alpha = 1/20)+
  xlim(13,90)+
  coord_trans(y = "sqrt")
```

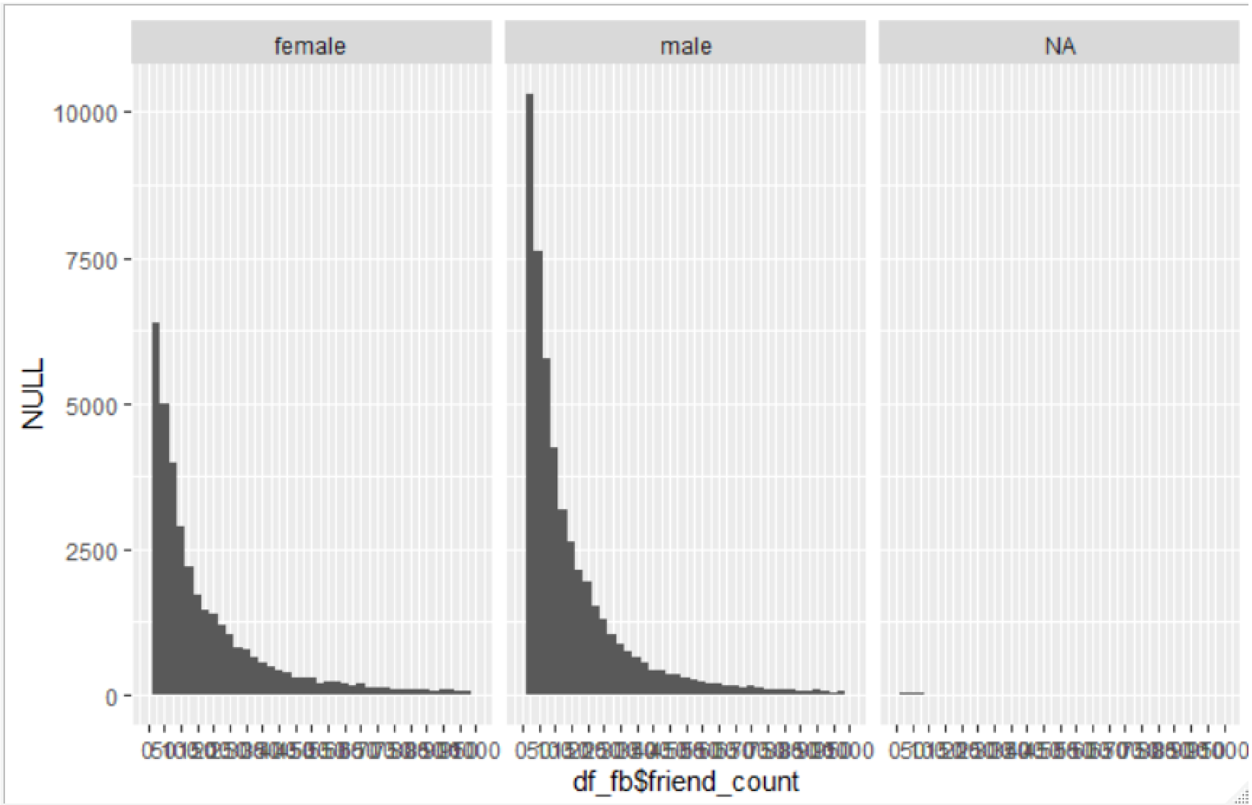
```
ggplot(aes(x = age, y = friend_count), data = df_fb)+
  geom_point(alpha = 1/20,
           position = position_jitter(h= 0),
           color = "red")+
  xlim(13,90)+
  coord_trans(y='sqrt')

qplot(age, friendships_initiated, data = df_fb)

# scatterplot of age vs friendships initiated
ggplot(aes(x = age, y = friendships_initiated),data = df_fb)+
  geom_point(alpha = 1/20)+
  xlim(13,90)

# scatterplot of age vs friendships initiated, using geom jitter to reduce overplotting
ggplot(aes(x = age, y = friendships_initiated),data = df_fb)+
  geom_jitter(alpha = 1/20)+
  xlim(13,90)

ggplot(aes(x = age, y = friendships_initiated),data = df_fb)+
  geom_point(alpha = 1/20,
           position = position_jitter(h = 0))+
  xlim(13,90)+
  coord_trans(y = 'sqrt')


library(dplyr)

# bin ages, create new dataframe with mean and median friend counts for each age group
age_groups <- group_by(df_fb,age)
df_fc_by_age <- summarise(age_groups,
                      mean_friend_count = mean(friend_count),
                      median_friend_count = median(friend_count),
                      n = n())

df_fc_by_age <- arrange(df_fc_by_age,age)

#scatterplot of mean friend counts for each age group
ggplot(aes(x = age, y = mean_friend_count), data = df_fc_by_age)+
  geom_line()+
  xlim(13,90)

# scatterplot of age vs friend count, using a jitter to reduce overplotting
# three lines denote mean, first quantile and third quantiles
ggplot(aes(x = age, y = friend_count), data = df_fb)+
  geom_point(alpha = 1/20,
           position = position_jitter(h= 0),
           color = "red")+
  xlim(13,90)+
  coord_trans(y='sqrt')+
  geom_line(stat = 'summary', fun.y = mean)+
  geom_line(stat = 'summary', fun.y = quantile, fun.args=list(probs=0.2), color = 'blue')+
  geom_line(stat = 'summary', fun.y = quantile, fun.args=list(probs=0.8), color = 'green')

# correlation co-efficient between age and friend count using pearson correlation coefficient
cor.test(df_fb$age, df_fb$friend_count, method="pearson")
#OR
with(df_fb, cor.test(age, friend_count, method = "pearson"))

# subsetting dataframe to include only rows with ages below 70, for better correlation coefficient
with(subset(df_fb, subset = df_fb$age<70), cor.test(age, friend_count, method = "pearson"))

#
ggplot(aes(x = www_likes_received, y = likes_received), data = df_fb)+
  geom_point()+
  xlim(0, quantile(df_fb$www_likes_received, 0.9))+
  ylim(0, quantile(df_fb$likes_received, 0.9))+
  geom_smooth("lm", color = 'red')

ggplot(aes(x = www_likes_received, y = likes_received), data = df_fb)+
  geom_point()+
  xlim(0, quantile(df_fb$www_likes_received, 0.9))+
  ylim(0, quantile(df_fb$likes_received, 0.9))

# scatterplot of www_likes received vs total likes received
qplot(data = df_fb, x = www_likes_received, y = likes_received)+
  scale_x_continuous(limits = c(0,30000), breaks = seq(0,30000,2000))+
  scale_y_continuous(limits = c(0,80000), breaks = seq(0,80000,10000))

# correlation between www likes and total likes
cor.test(df_fb$www_likes_received, df_fb$likes_received)

# new column to show age in months
df_fb$age_by_month <- NULL
df_fb$age_with_months <- df_fb$age + (12-df_fb$dob_month)/12

# bin ages in months; create new dataframe that has mean and median friend counts for each age in
months
age_months_groups <- group_by(df_fb, age_with_months)
df_fc_by_age_months <- summarise(age_months_groups,
                                 mean_friend_count = mean(friend_count),
                                 median_friend_count = median(friend_count),
                                 n = n())

df_fc_by_age_months <- arrange(df_fc_by_age_months, age_with_months)
```

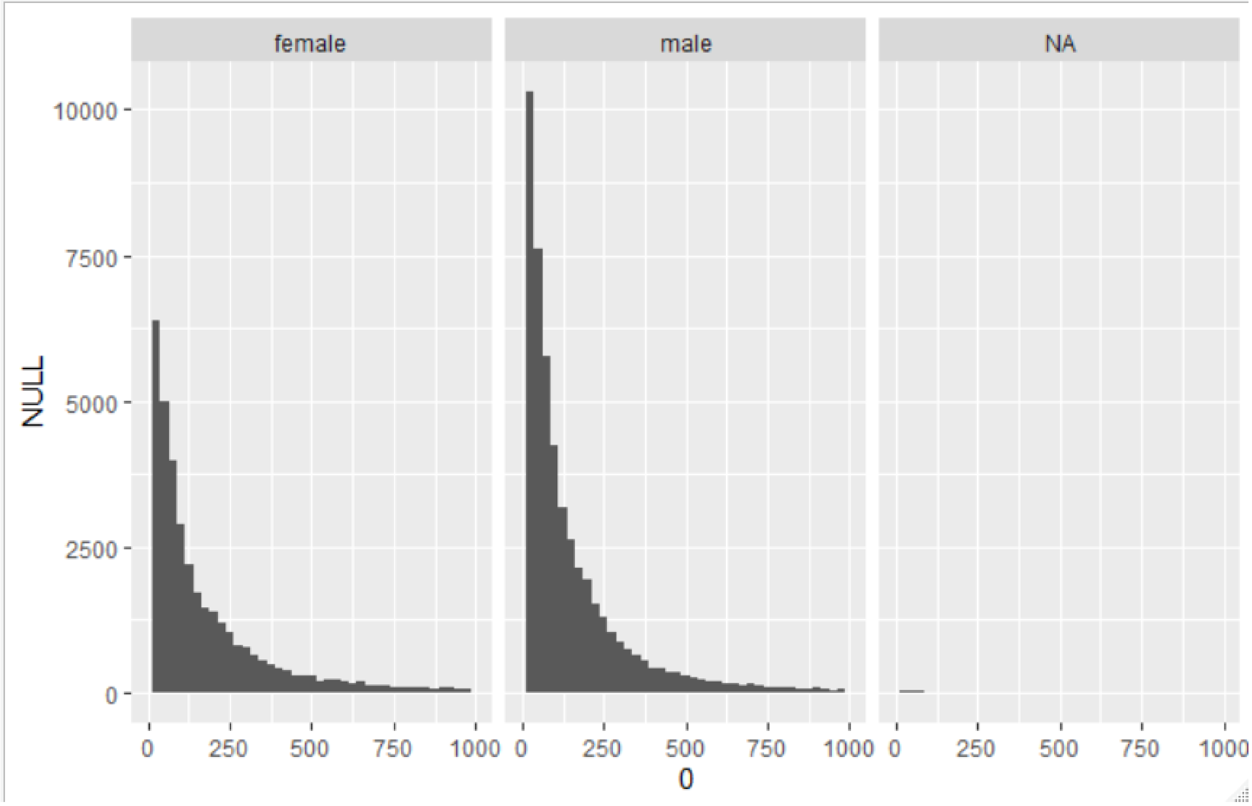
```
df_fc_by_age_months$age_with_months <- as.numeric(df_fc_by_age_months$age_with_months)

p1 <- ggplot(aes(x = age_with_months, y = mean_friend_count),
             data = subset(df_fc_by_age_months, subset = age_with_months < 71))+
  geom_line()+
  geom_smooth()

p2 <- ggplot(aes(x = age, y = mean_friend_count),
             data = subset(df_fc_by_age, subset = age < 71))+
  geom_line()+
  geom_smooth()

p3 <- ggplot(aes(x = round(age/5)*5, y = friend_count),
             data = subset(df_fb, subset = age < 71))+
  geom_line(stat = "summary", fun.y = mean)

grid.arrange(p1,p2, p3, ncol = 1)



ggplot(aes(x = gender, y = friend_count),
       data = subset(df_fb, !is.na(gender)),
       geom = "boxplot")+
  scale_y_continuous(limits = c(0,1000), breaks = seq(0,1000,50))+
  stat_summary(fun.y = mean, geom = "point", shape = 4)


ggplot(aes(x = age, y = friend_count),
       data = subset(df_fb, !is.na(gender)))+
  geom_line(aes(color = gender), stat = "summary", fun.y = median)


age_groups <- group_by(df_fb, age, gender)
df_fc_by_age_gender <- summarise(age_groups,
                                 mean_friend_count = mean(friend_count),
                                 median_friend_count = median(friend_count),
                                 n = n())

df_fc_by_age_gender <- arrange(pf.fc_by_age_gender, age)

#using chaining
pf.fc_by_age_gender <- df_fb %>%
  filter(!is.na(gender)) %>%
  group_by(age, gender) %>%
  summarise(mean_friend_count = mean(friend_count),
            median_friend_count = median(friend_count),
            n = n()) %>%
  ungroup() %>%
  arrange(age)


ggplot(aes(x = age, y = median_friend_count), data = pf.fc_by_age_gender)+
  geom_line(aes(color = gender), stat = "summary", fun.y = median)


#convert to wide format
#package required: reshape2
df_fb_wide <- dcast(pf.fc_by_age_gender,
                  age ~ gender,
                  value.var = 'median_friend_count')

#ratio of friend_counts of both genders
ggplot(aes(x = age, y = female/male), data = df_fb_wide)+
  geom_line()+
  geom_hline(yintercept = 1, alpha = 0.3, linetype = 2)


# consider tenure while analyzing friend_count
# tenure variable shows number of days since member joined facebook
# create new quantitative variable called year_joined

str(df_fb)
df_fb$year_joined <- as.integer(2014 - df_fb$tenure/365)

#alternative to as.integer --> floor()
df_fb$year_joined <- floor(2014 - df_fb$tenure/365)

# buckets for year_joined
df_fb$year_joined.buckets <- cut(df_fb$year_joined,
                                c(2004,2009, 2011, 2012, 2014))



# friend count vs age for year joined buckets
ggplot(aes(x = age, y = friend_count), data = df_fb[!is.na(df_fb$year_joined.buckets),])+
  geom_line(aes(color = year_joined.buckets), stat = "summary", fun.y = median)+
  geom_line(stat = "summary", fun.y = mean, linetype = 3)


# median friend count vs age for year joined buckets
pf.fc_by_age_yearjoined <- df_fb %>%
  filter(!is.na(year_joined.buckets)) %>%
  group_by(age, year_joined.buckets) %>%
  summarise(mean_friend_count = mean(friend_count),
          median_friend_count = median(friend_count),
          n = n()) %>%
  ungroup() %>%
  arrange(age)

ggplot(aes(x = age, y = median_friend_count),
      data = pf.fc_by_age_yearjoined)+
  geom_line(aes(color = year_joined.buckets),
          stat = "summary", fun.y = median)



# rate of friendship
df_fb$friend_rate <- df_fb$friend_count / df_fb$tenure

summary(subset(df_fb$friend_rate, df_fb$friend_rate != Inf))

#alternative to new column:

with(subset(df_fb, tenure >= 1), summary(friend_count/tenure))

#bias variance trade-off

# friendships initiated vs tenure
ggplot(aes(x = tenure, y = friendships_initiated/tenure),
      data = subset(df_fb, tenure >= 1))+
  geom_line(aes(color = year_joined.buckets),
          stat = "summary", fun.y = mean)


# making the plot smooth, remove noise
ggplot(aes(x = 50* round(tenure/50), y = friendships_initiated/tenure),
      data = subset(df_fb, tenure >= 1))+
  geom_smooth(aes(color = year_joined.buckets),
           stat = "summary", fun.y = mean)


# using ggally
# scatterplots between all combinations of important variables
# seed ensures reproducible results
set.seed(7777)
df_subset <- df_fb[,c(2:15)]

ggpairs(df_subset[sample.int(nrow(df_subset),1000), ])

str(df_fb)
#creating a heat map
ggplot(aes(x = age, y = tenure), data = df_fb)+
  geom_tile()+
  scale_fill_gradient(colors = colorRampPalette(c('blue','red'))(100))
```

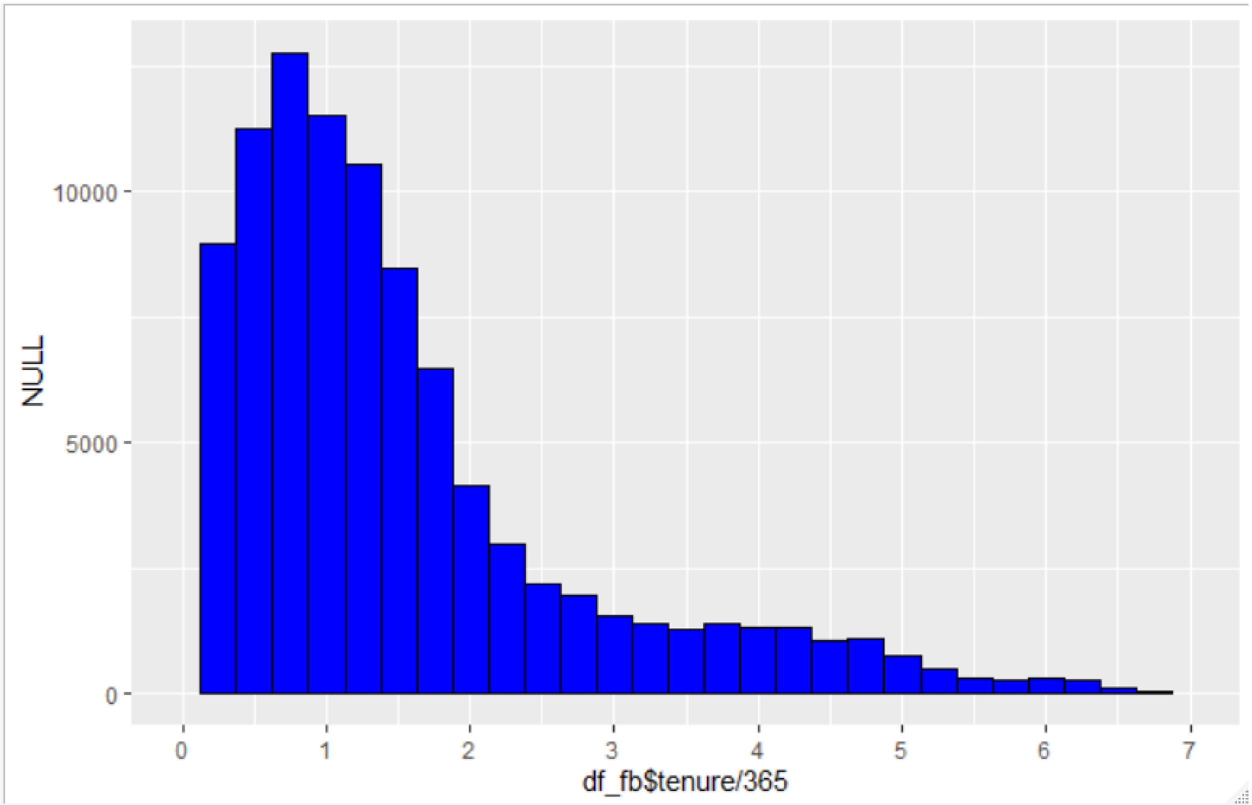
```
# ratio of friendships initiated and total friend count
df_fb$prop_initiated <- df_fb$friendships_initiated / df_fb$friend_count

# frindship proportion vs tenure on facebook
ggplot(aes(x = tenure, y = prop_initiated), data = df_fb)+
  geom_line(aes(color = year_joined.buckets),
          stat = 'summary', fun.y = median)
```

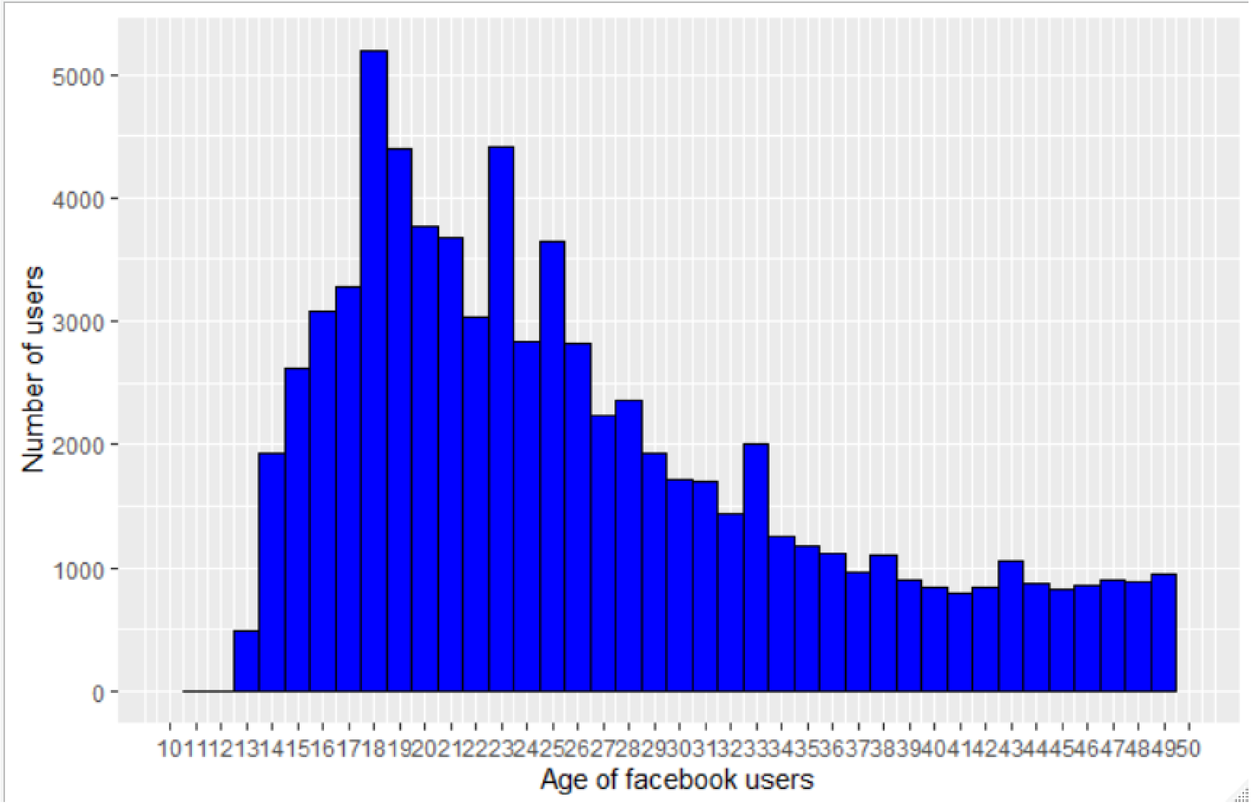
Output



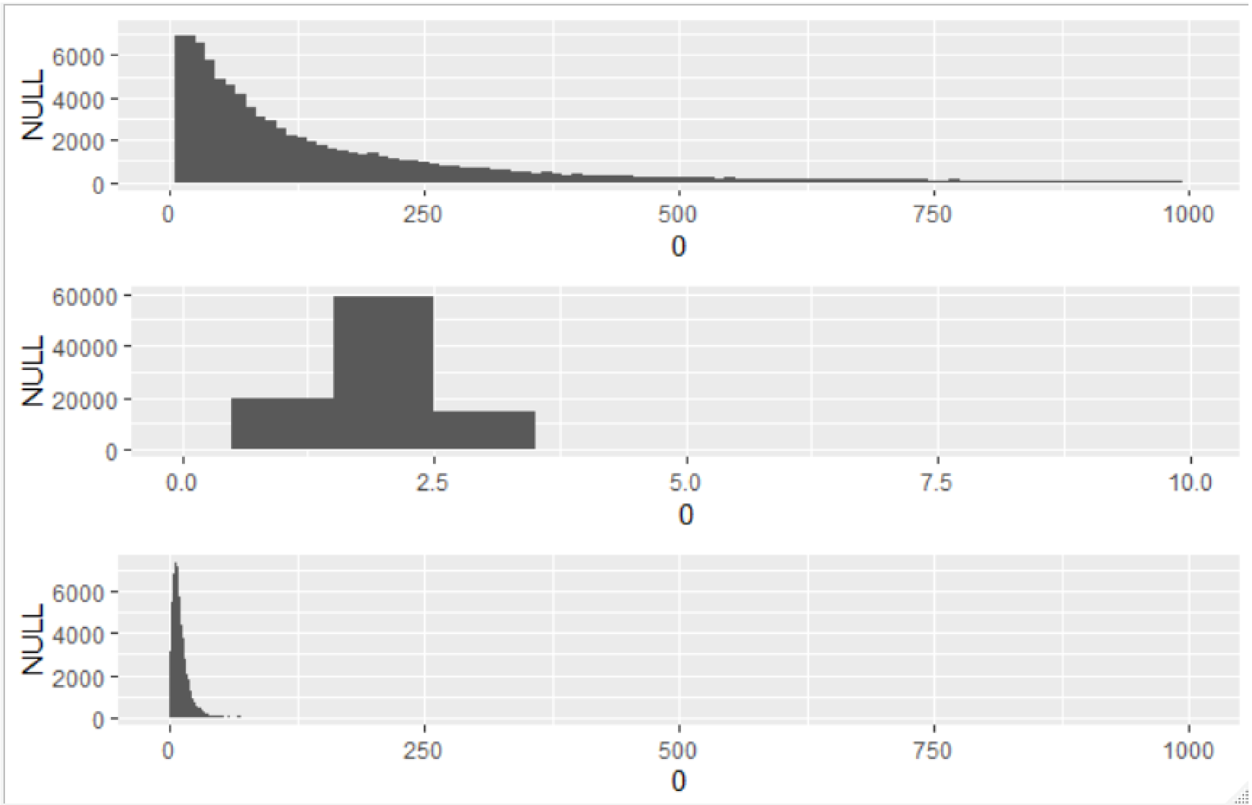

female
male
NA
NULL
10000
7500
5000
2500
0
df_fb$friend_count

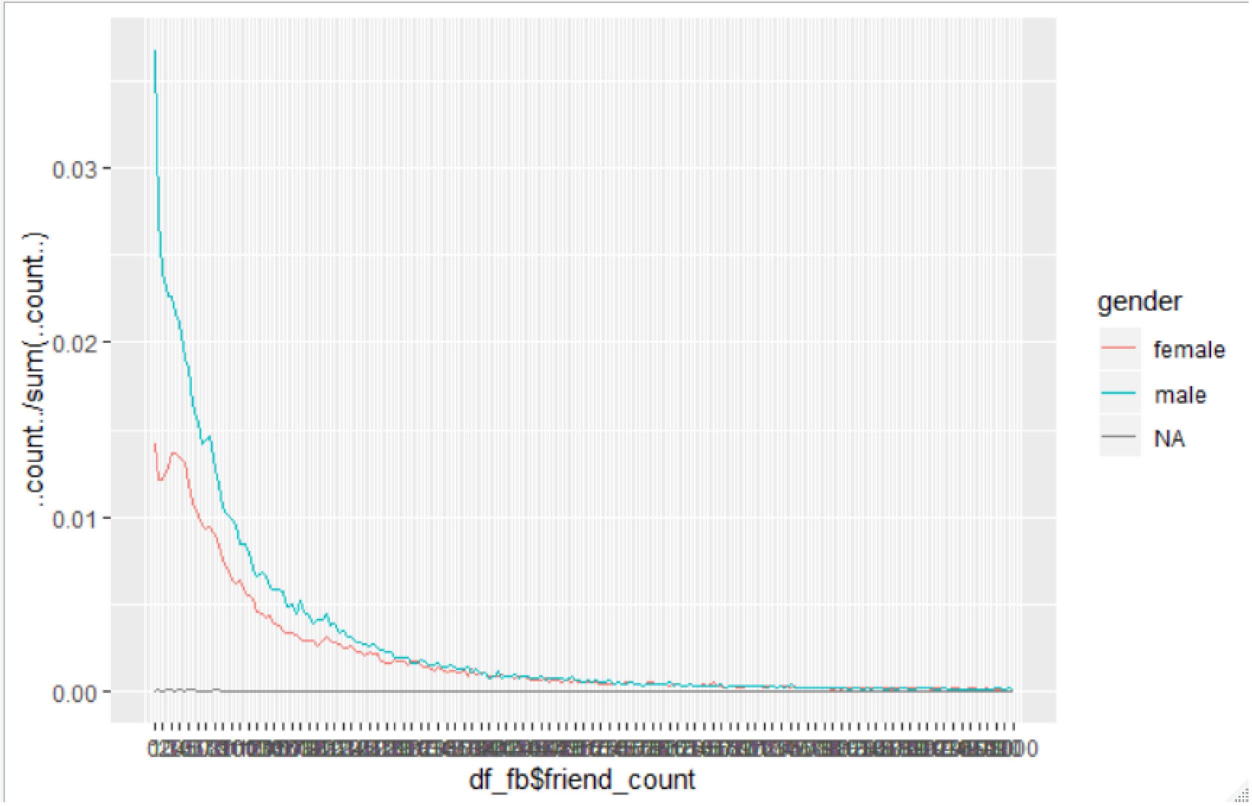

female
male
NA
NULL
10000
7500
5000
2500
0
0
250
500
750
1000
0

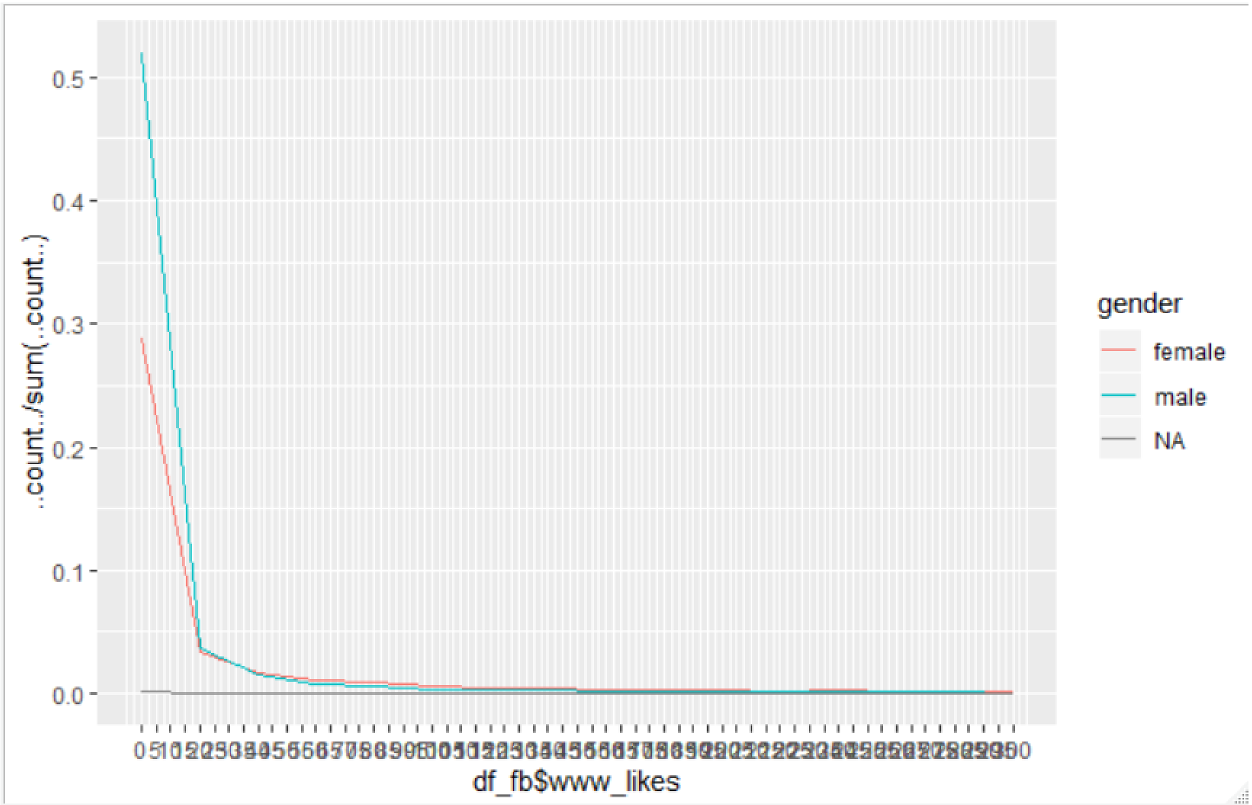


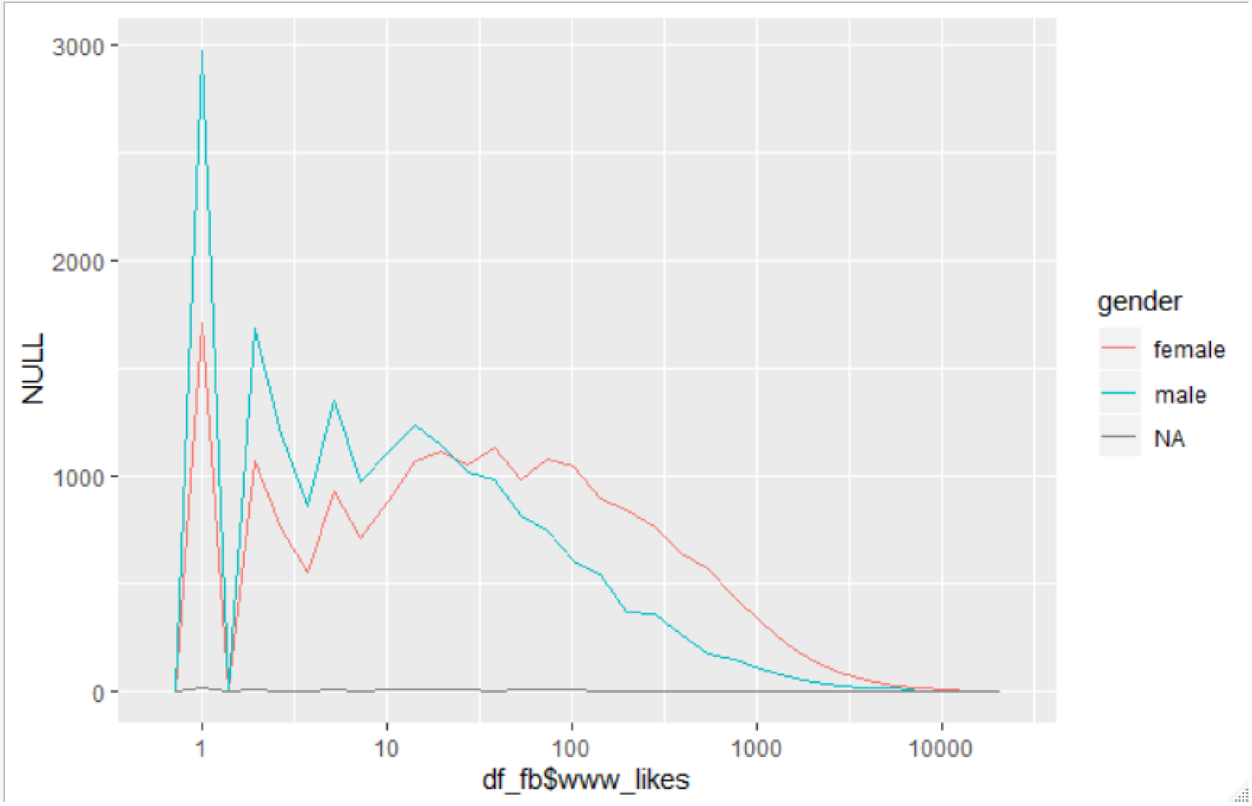


```
summary(log(df_fb$friend_count))
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
   -Inf   3.434   4.407    -Inf   5.328   8.502 
```


NULL
6000
4000
2000
0
0
250
500
750
1000
0
NULL
60000
40000
20000
0
0.0
2.5
5.0
7.5
10.0
0
NULL
6000
4000
2000
0
0
250
500
750
1000
0

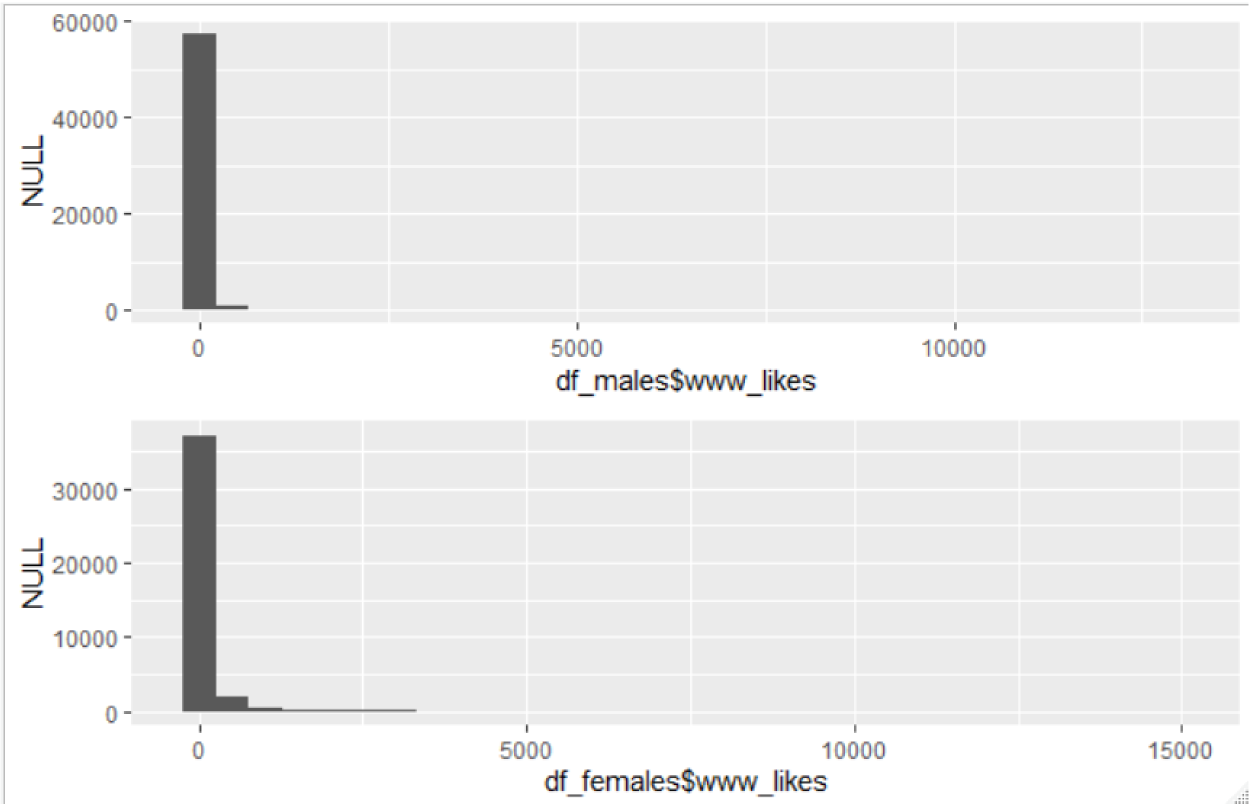

..count../sum(..count..)
0.03
0.02
0.01
0.00
df_fb$friend_count
gender
female
male
NA

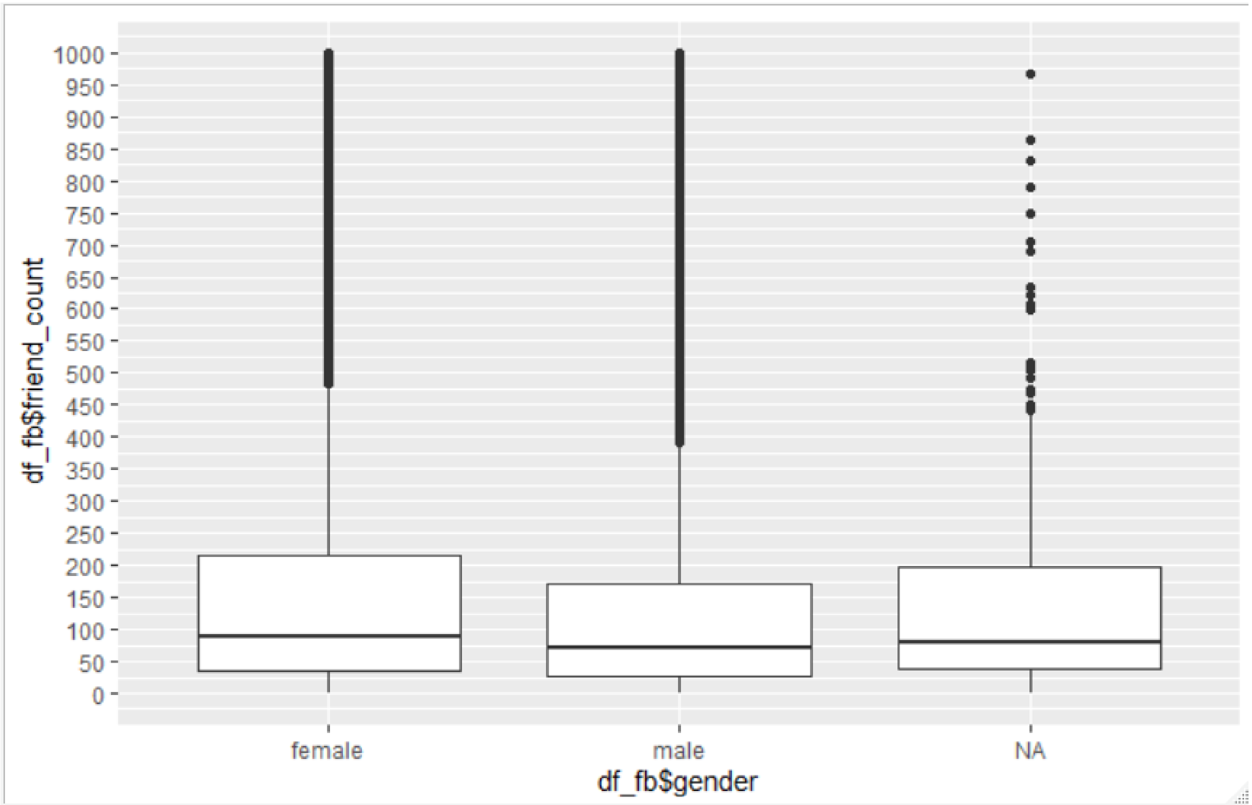


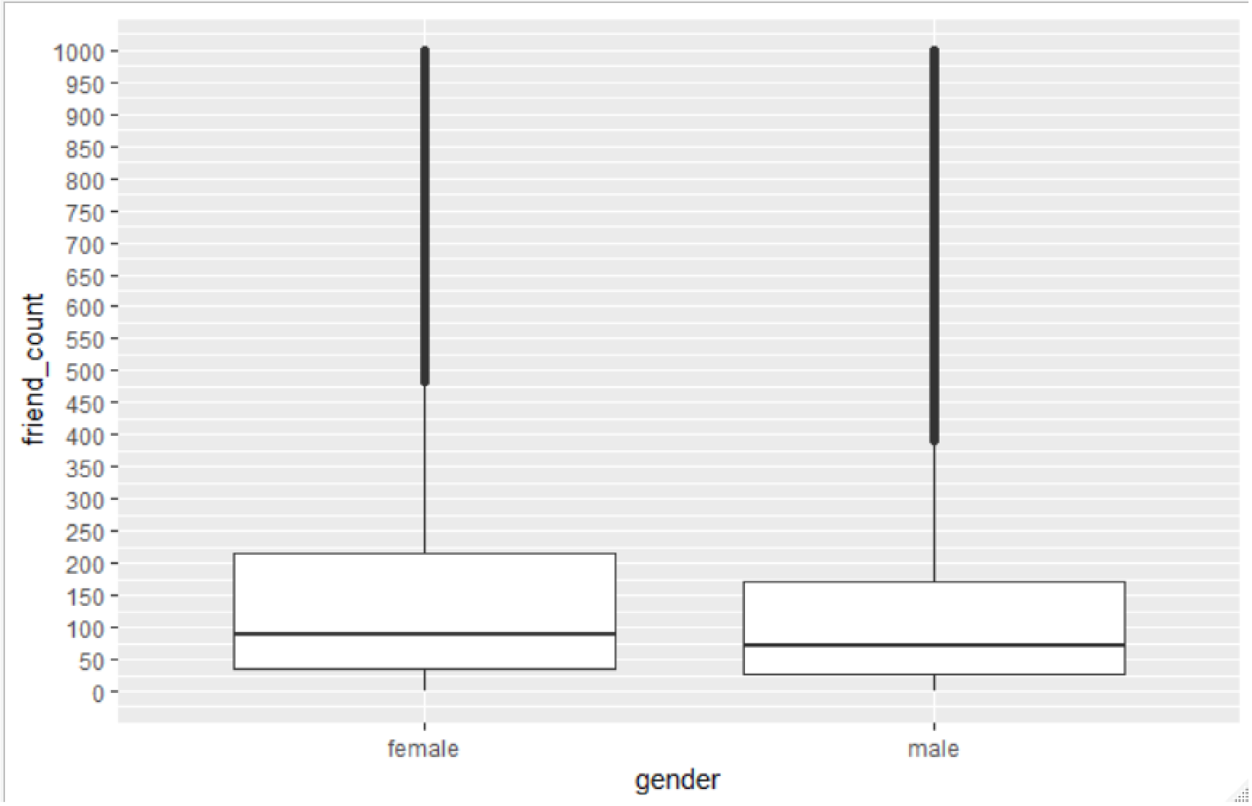


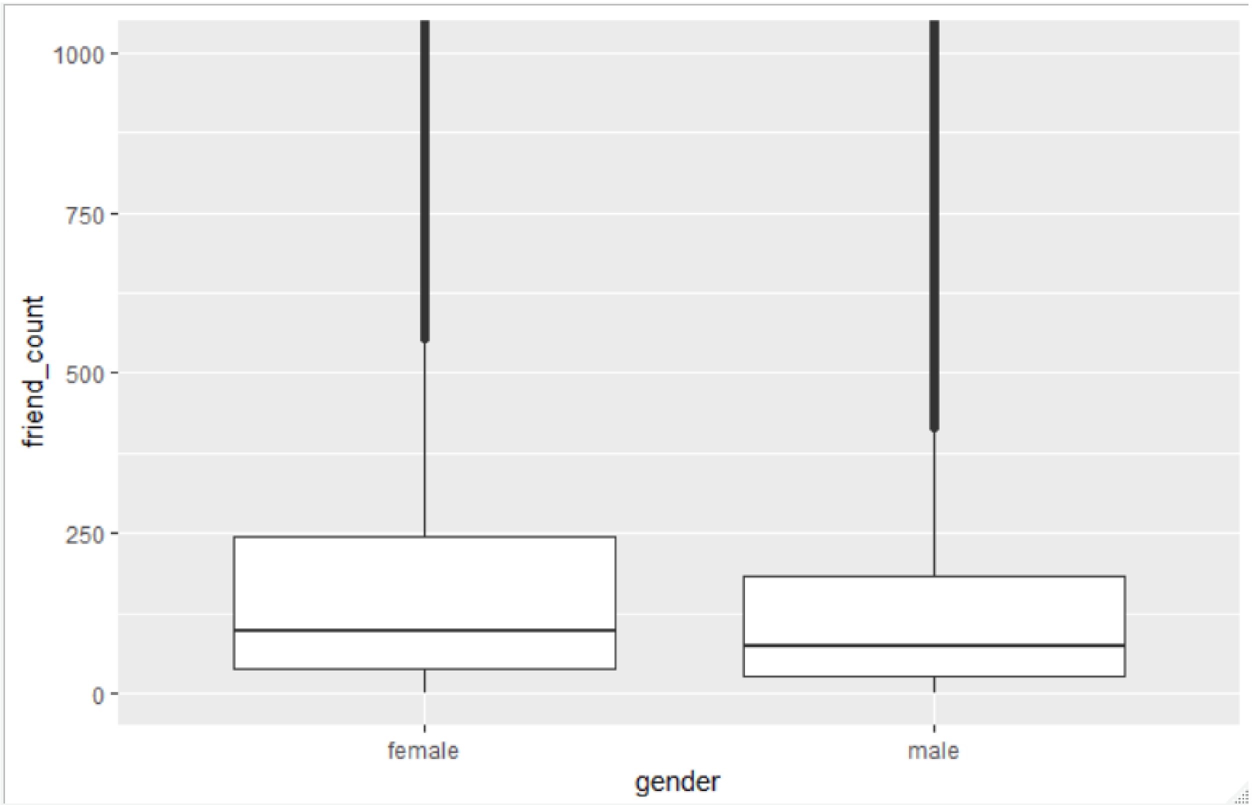
```
> # subset of facebook dataframe, with only data about males
> df_males <- df_fb[df_fb$gender == "male",]
> str(df_males)
```

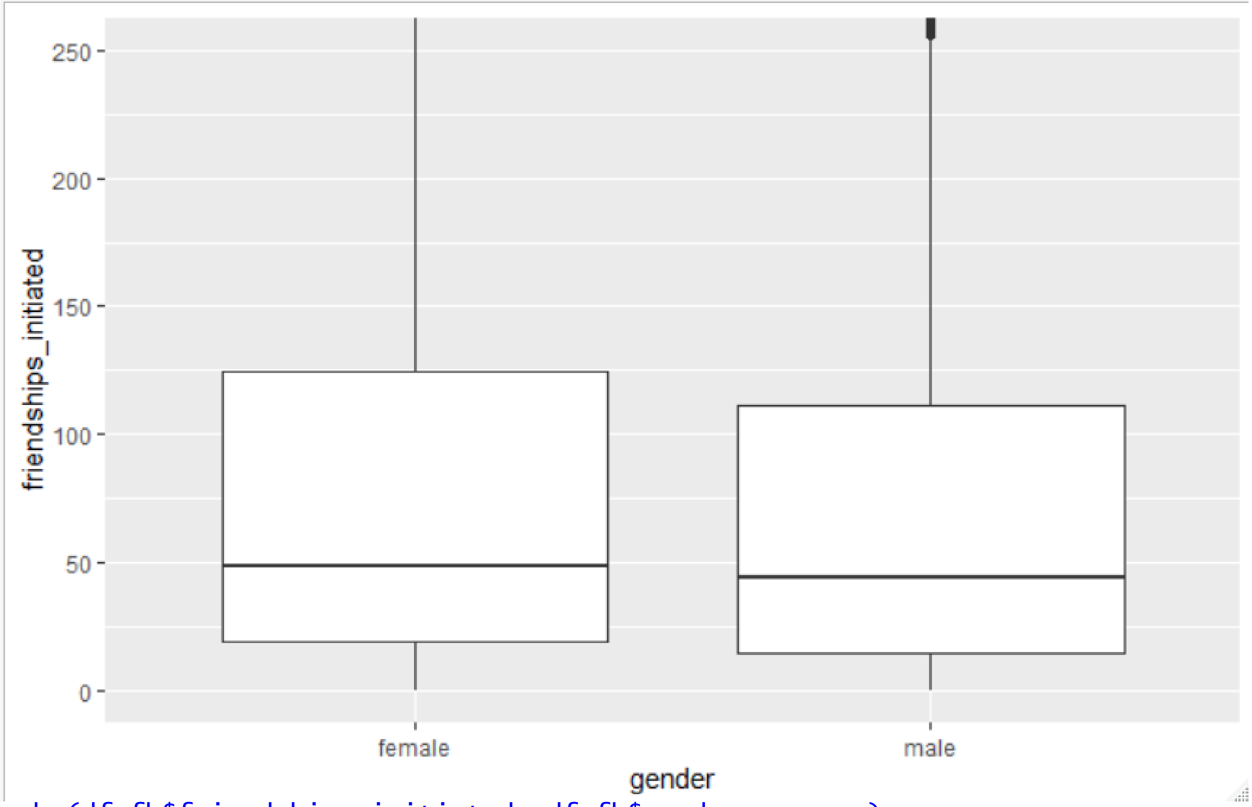
```
'data.frame':	58749 obs. of  15 variables:
 $ userid               : int  2094382 2083884 1733186 1524765 1136133 136517
4 1712567 1612453 2104073 1918584 ...
 $ age                  : int  14 14 14 14 13 13 13 13 13 13 ...
 $ dob_day              : int  19 16 4 1 14 1 2 22 1 5 ...
 $ dob_year             : int  1999 1999 1999 1999 2000 2000 2000 2000 2000 2
000 ...
 $ dob_month            : Factor w/ 12 levels "1","2","3","4",..: 11 11 12 12
1 1 2 2 2 3 ...
 $ gender               : Factor w/ 2 levels "female","male": 2 2 2 2 2 2 2 2
2 2 ...
 $ tenure               : int  266 13 82 15 12 81 171 98 55 106 ...
 $ friend_count         : int  0 0 0 0 0 0 0 0 0 0 ...
```




df_fb$friend_count
1000
950
900
850
800
750
700
650
600
550
500
450
400
350
300
250
200
150
100
50
0
female
male
NA
df_fb$gender


friend_count
1000
950
900
850
800
750
700
650
600
550
500
450
400
350
300
250
200
150
100
50
0
female
male
gender

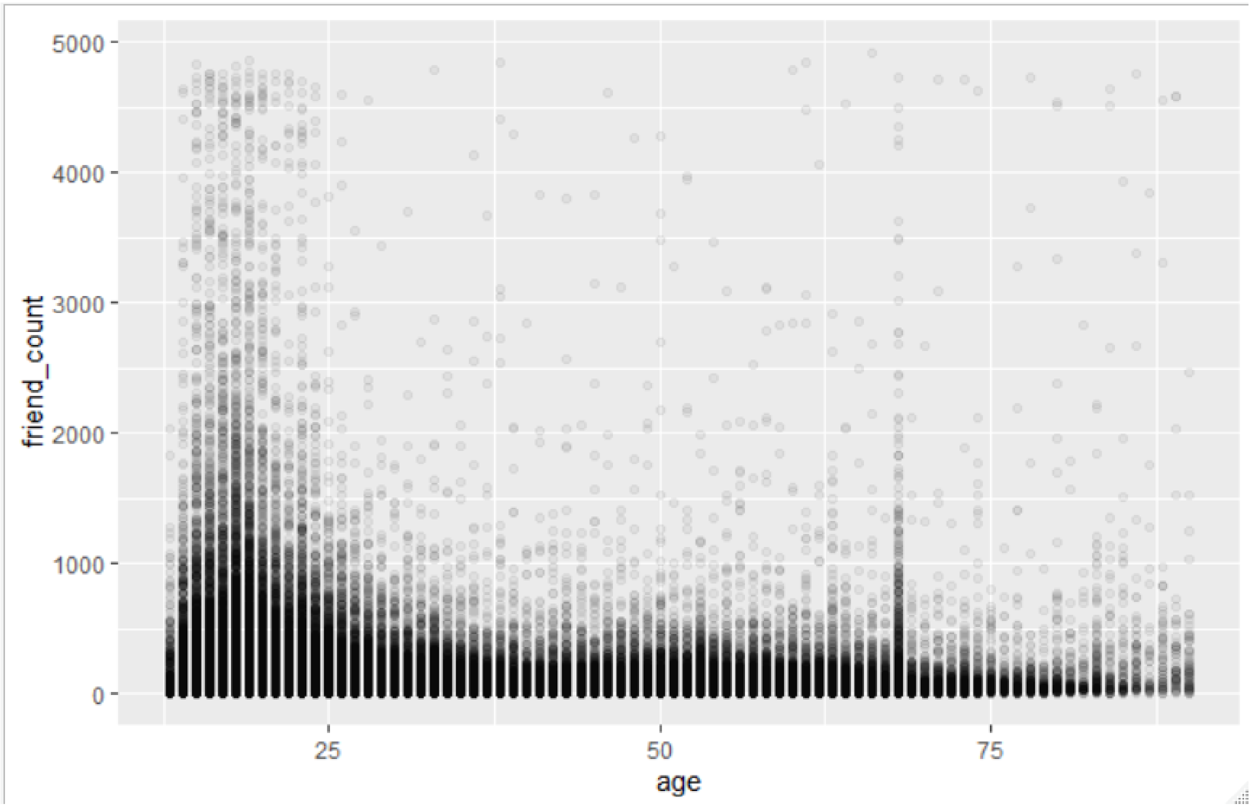


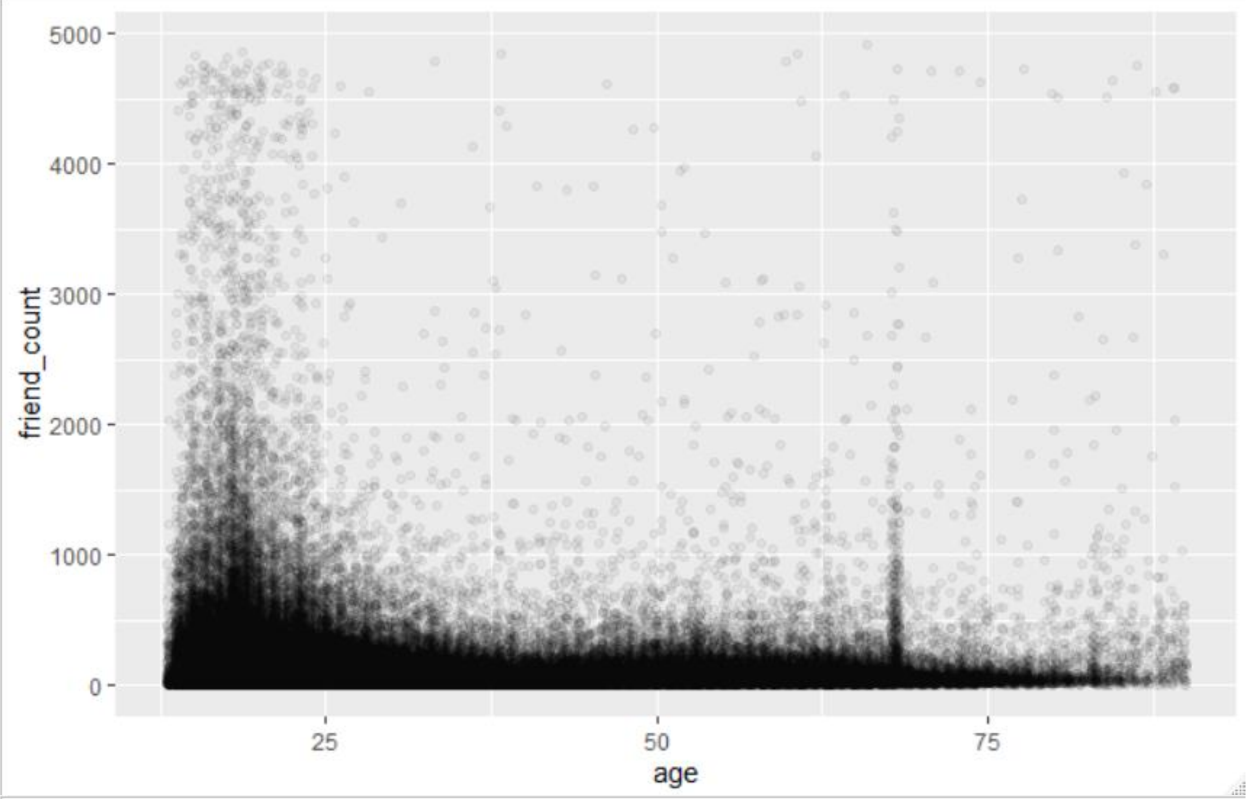


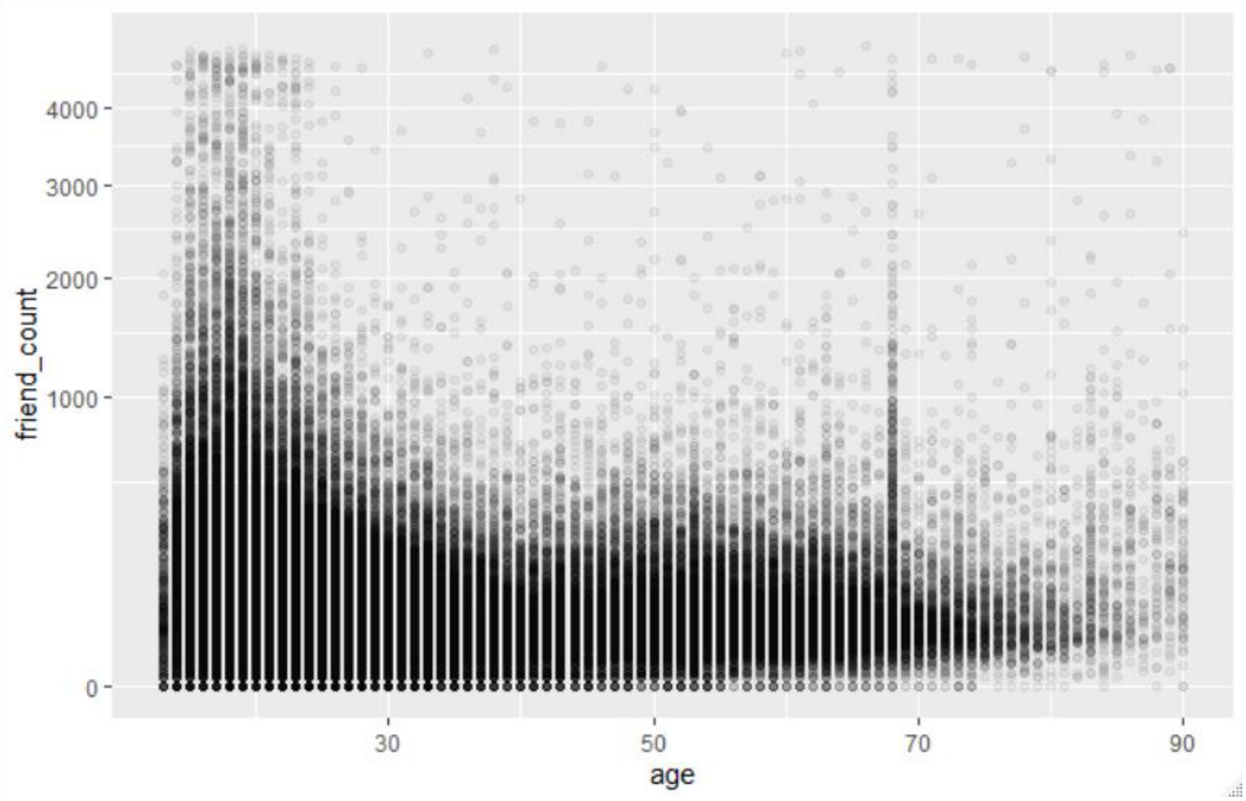
```
> by(df_fb$friendships_initiated, df_fb$gender,summary)
df_fb$gender: female
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
    0.0    19.0    49.0   113.9   124.8  3654.0 
```

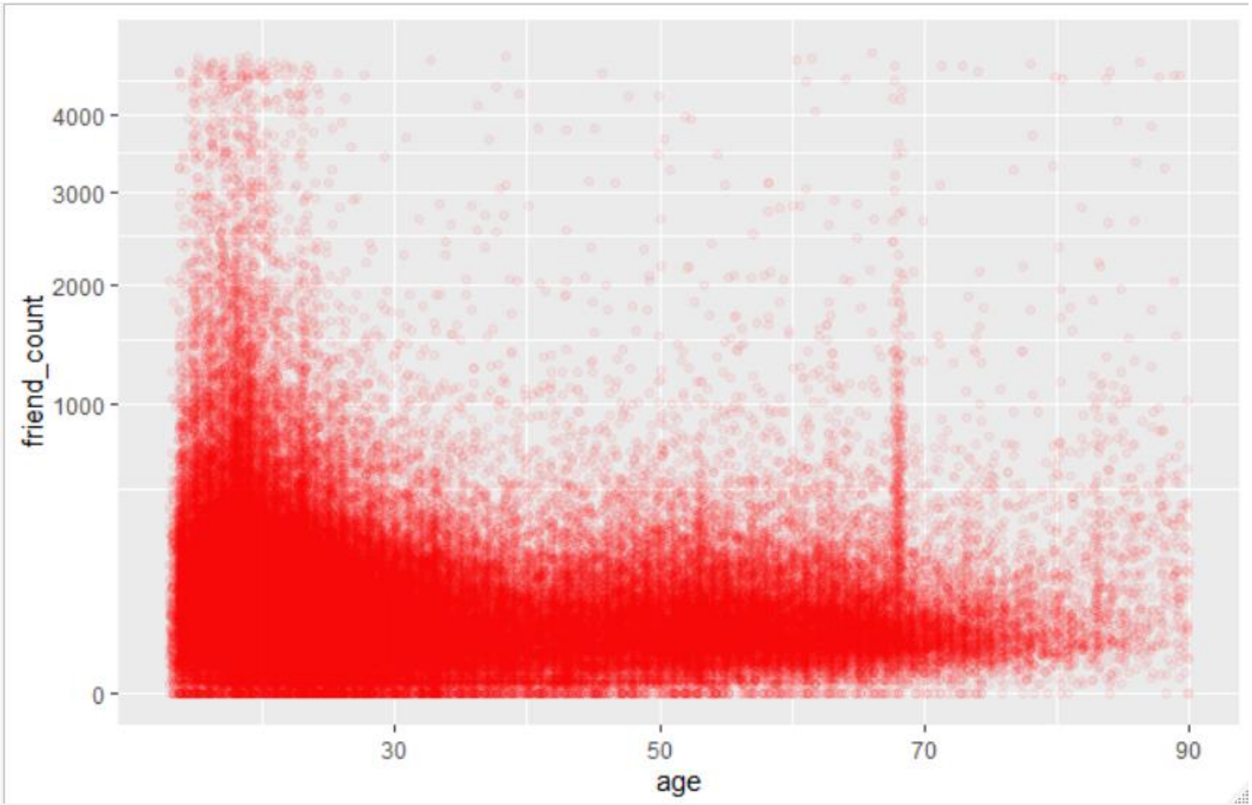
```
----------------------------------------------------------------------
df_fb$gender: male
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
    0.0    15.0    44.0   103.1   111.0  4144.0 
```

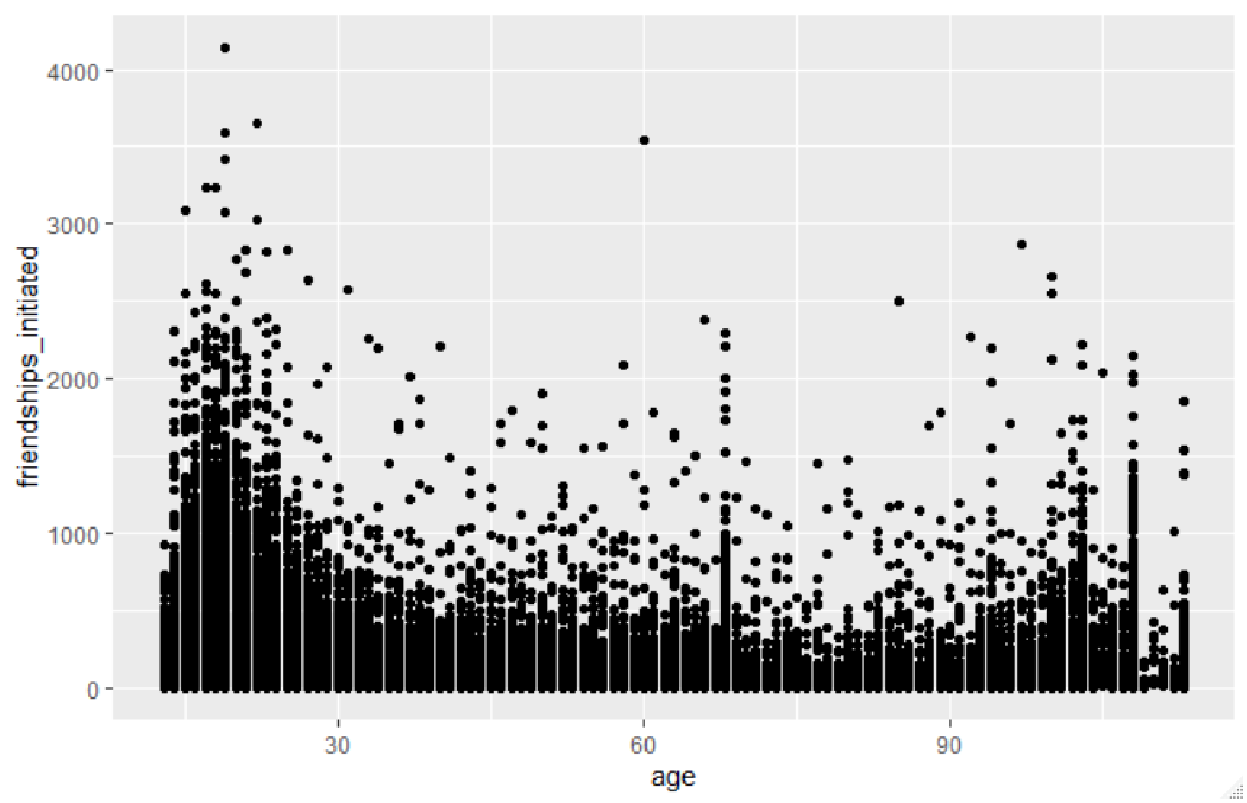
```
summary(df_fb$mobile_checkin)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
 0.0000  0.0000  1.0000  0.6459  1.0000  1.0000 
```

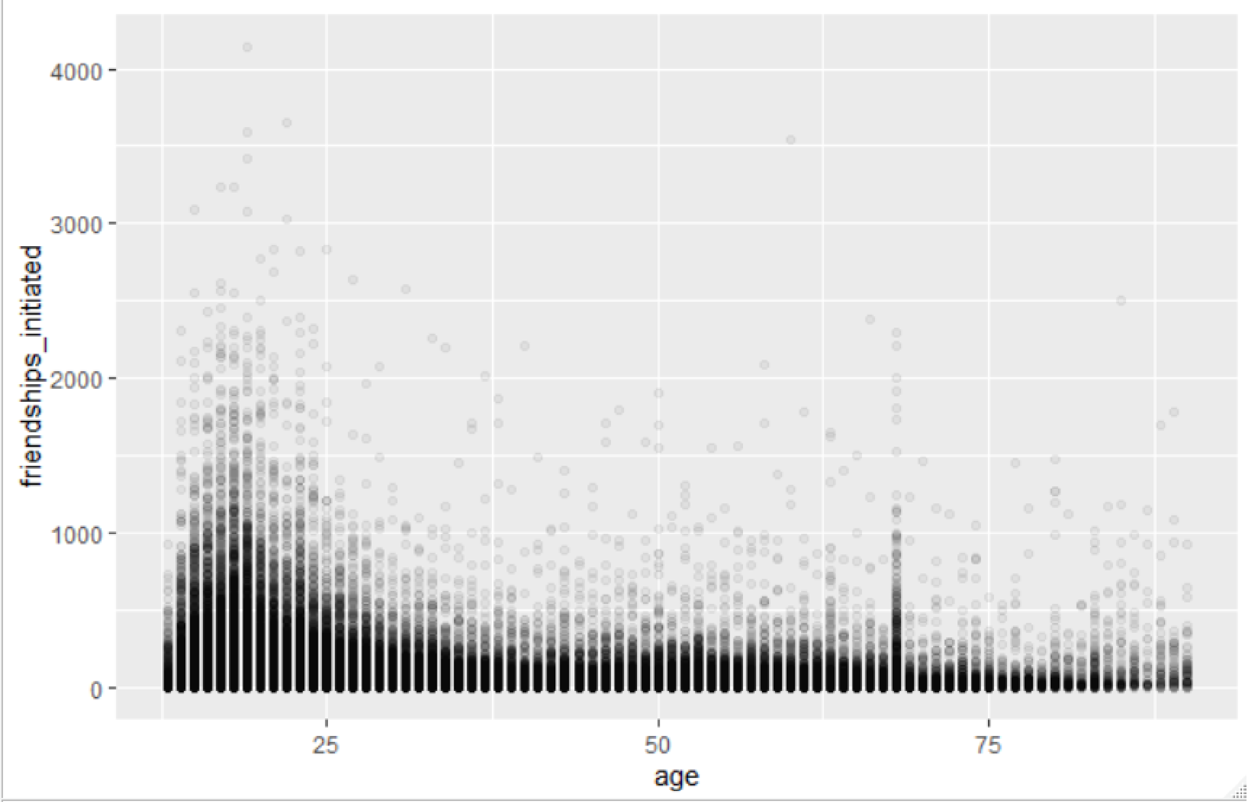


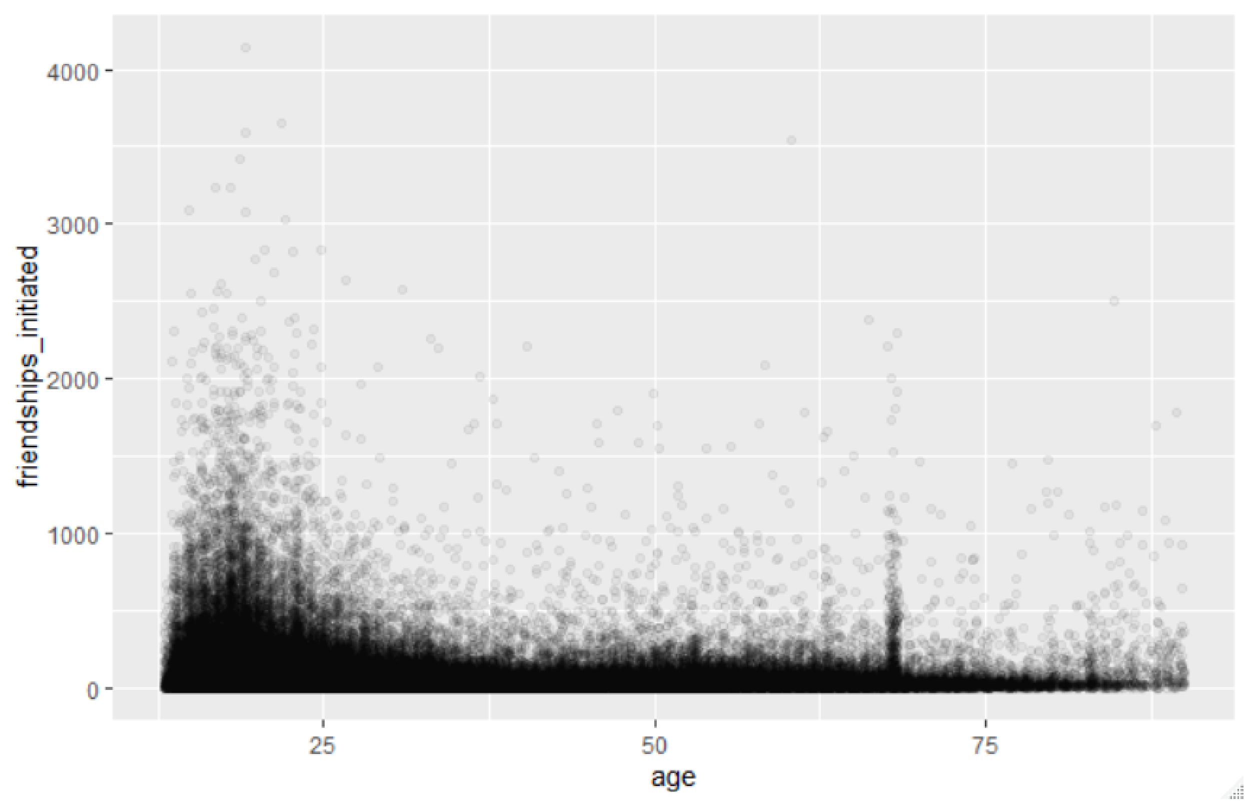

5000
4000
3000
2000
1000
0
friend_count
25
50
75
age

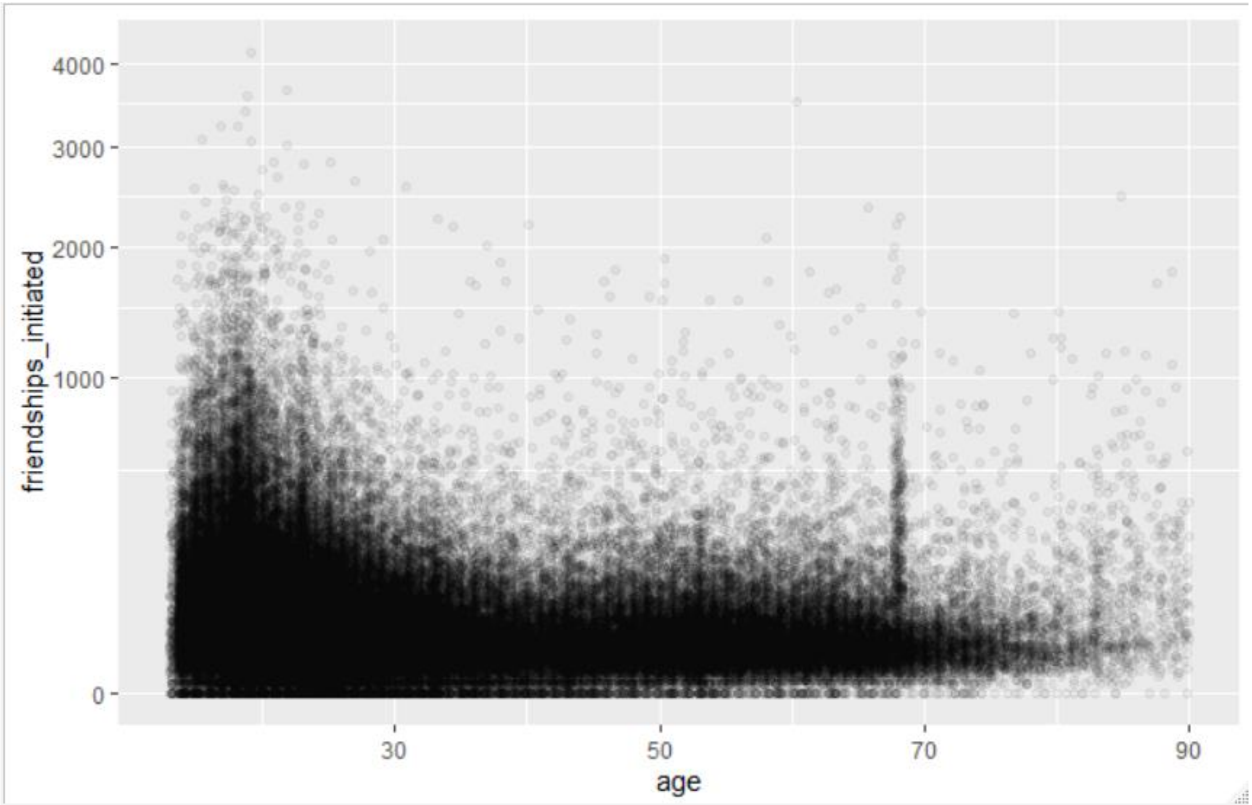

4000
3000
2000
1000
0
friend_count
30
50
70
90
age

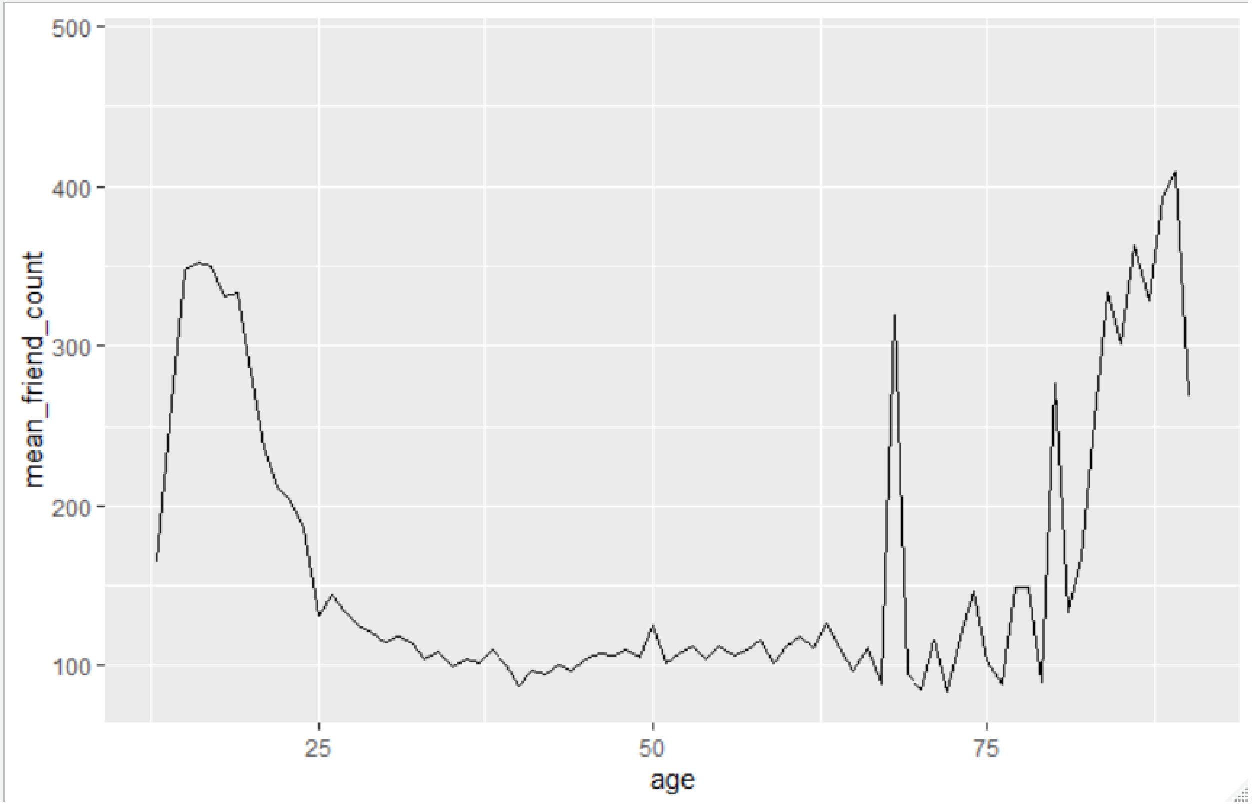

4000
3000
2000
1000
0
friend_count
30
50
70
90
age

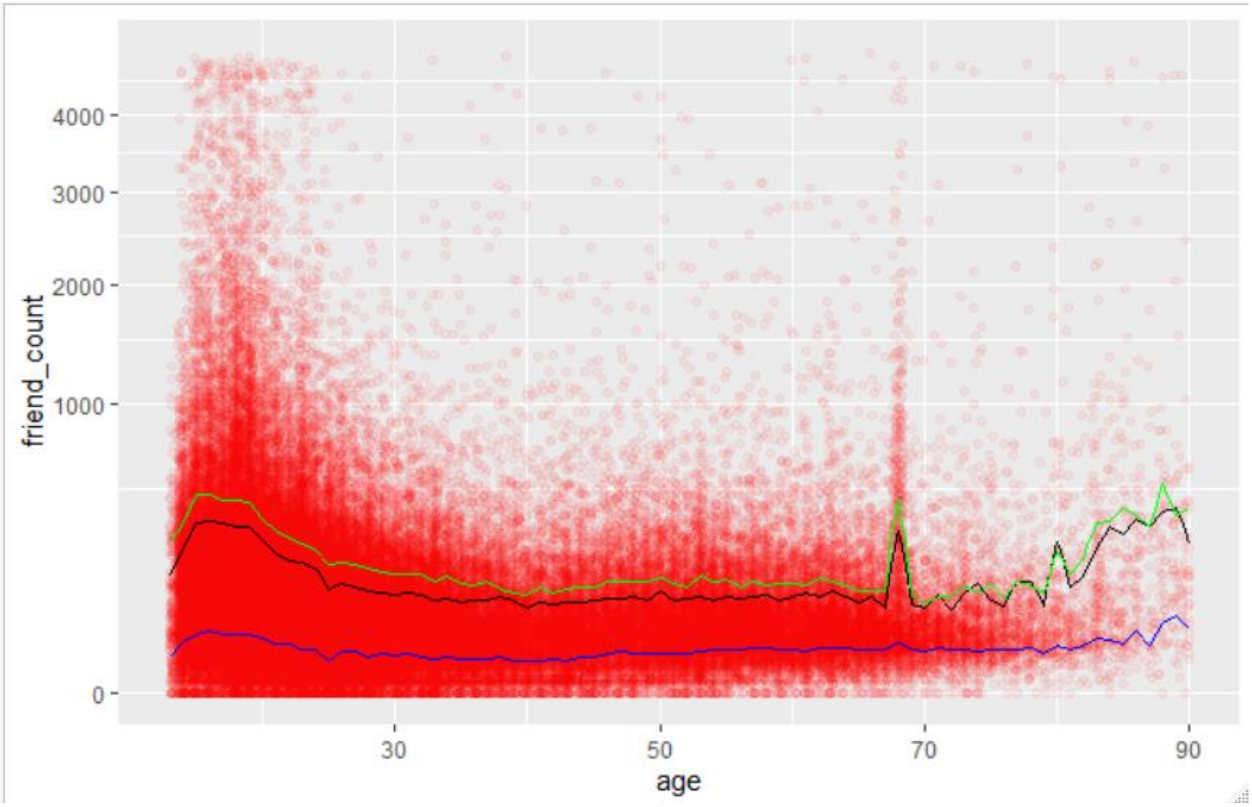

4000
3000
2000
1000
0
friendships_initiated
30
60
90
age


4000
3000
2000
1000
0
friendships_initiated
25
50
75
age

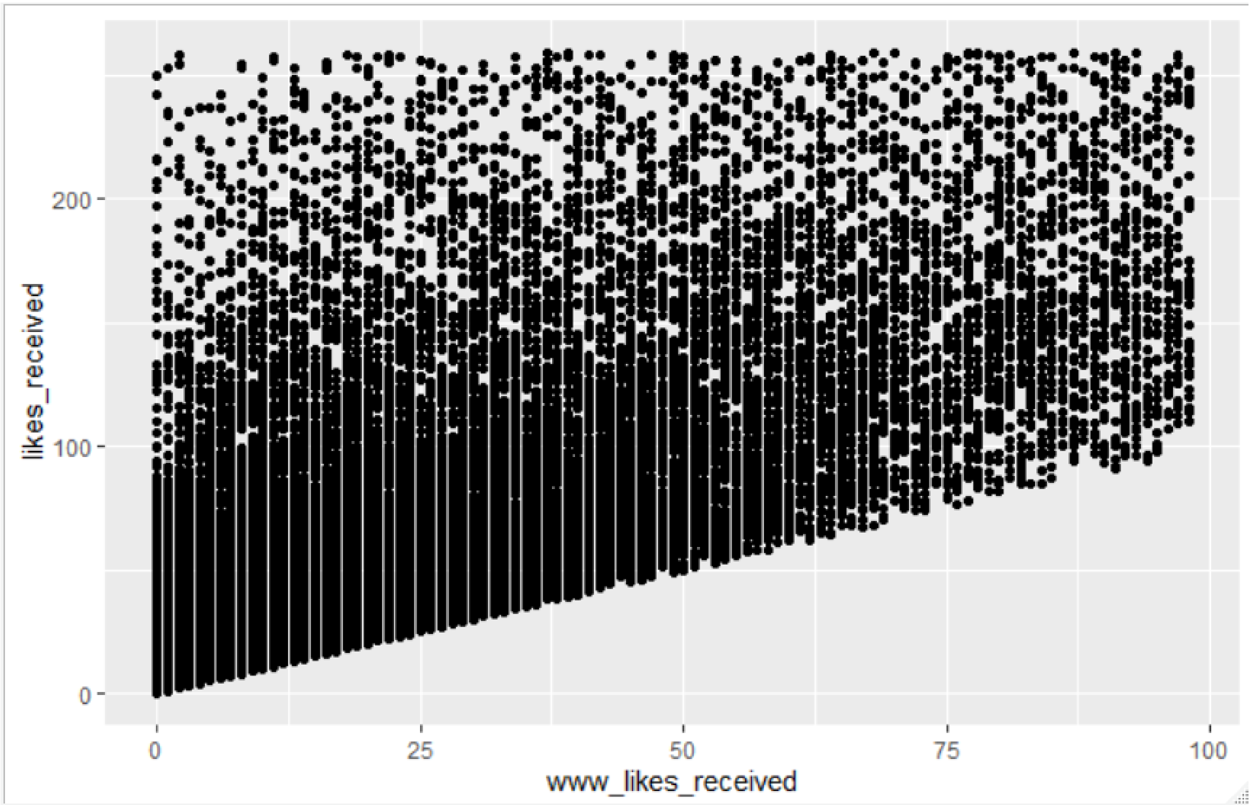

4000
3000
2000
1000
0
friendships_initiated
25
50
75
age

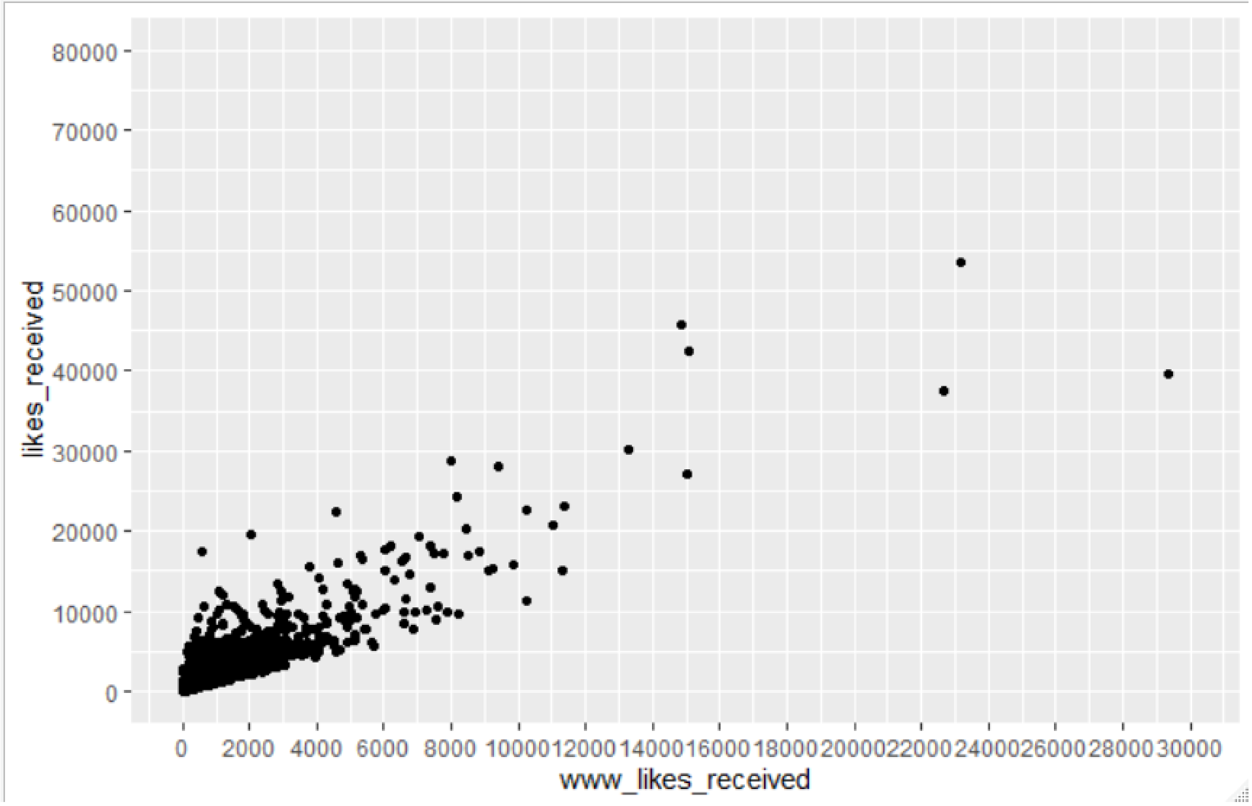

friendships_initiated
4000
3000
2000
1000
0
30
50
70
90
age


mean_friend_count
500
400
300
200
100
25
50
75
age



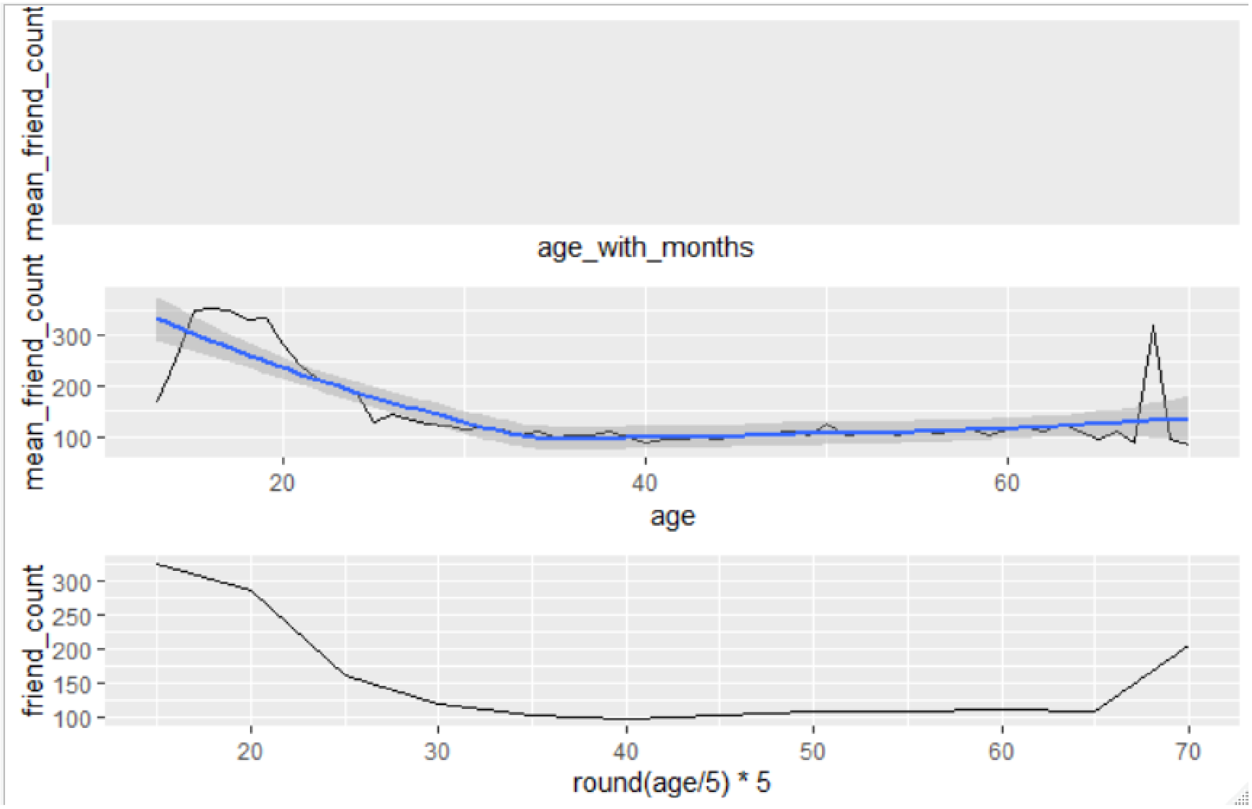
```
> with(subset(df_fb, subset = df_fb$age<70), cor.test(age, friend_count, meth
od = "pearson"))

        Pearson's product-moment correlation

data:  age and friend_count
t = -52.326, df = 90664, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.1775257 -0.1648889
sample estimates:
       cor
-0.1712144
```

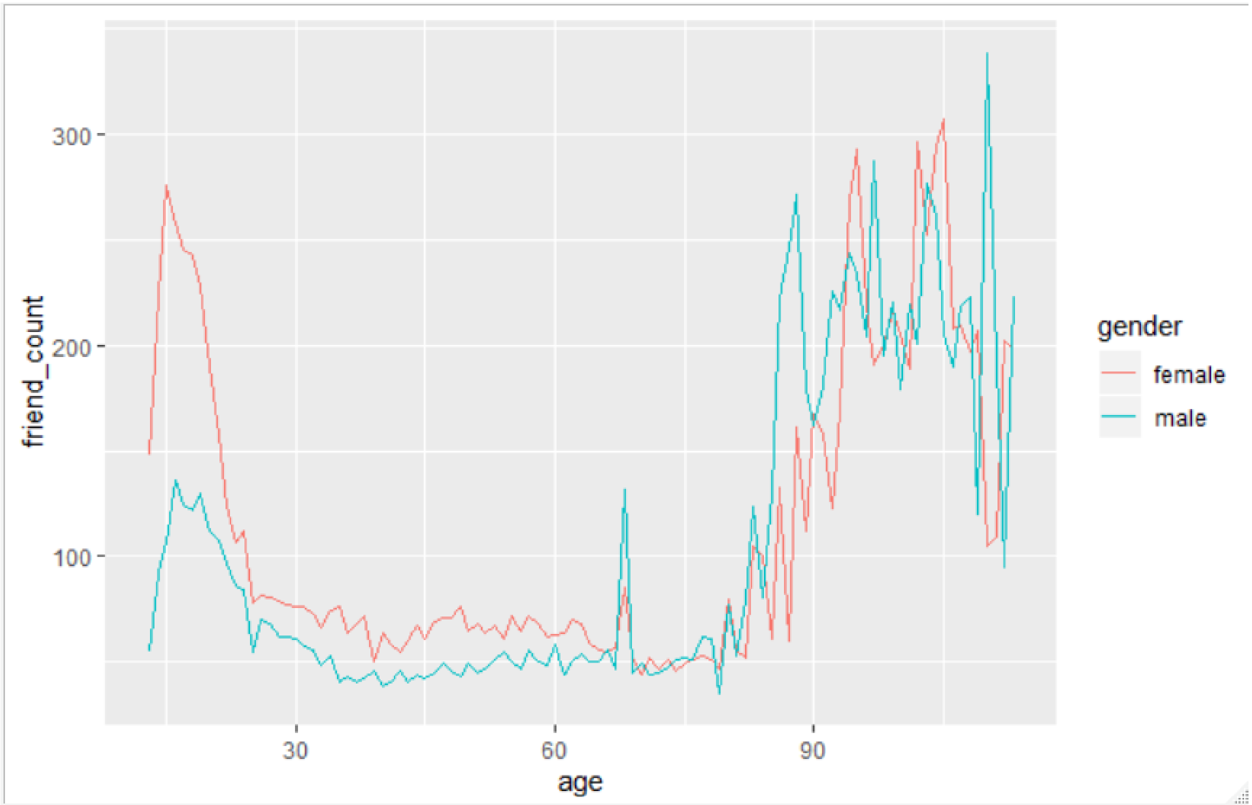


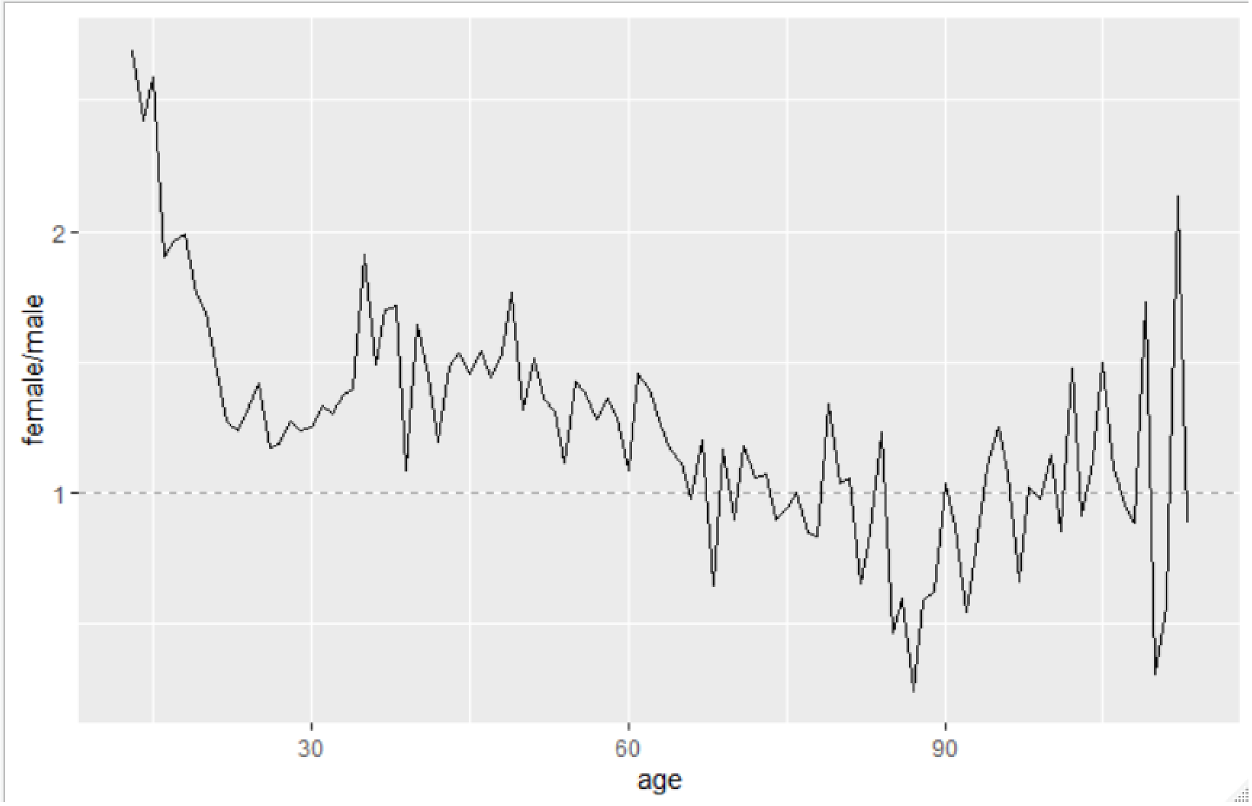


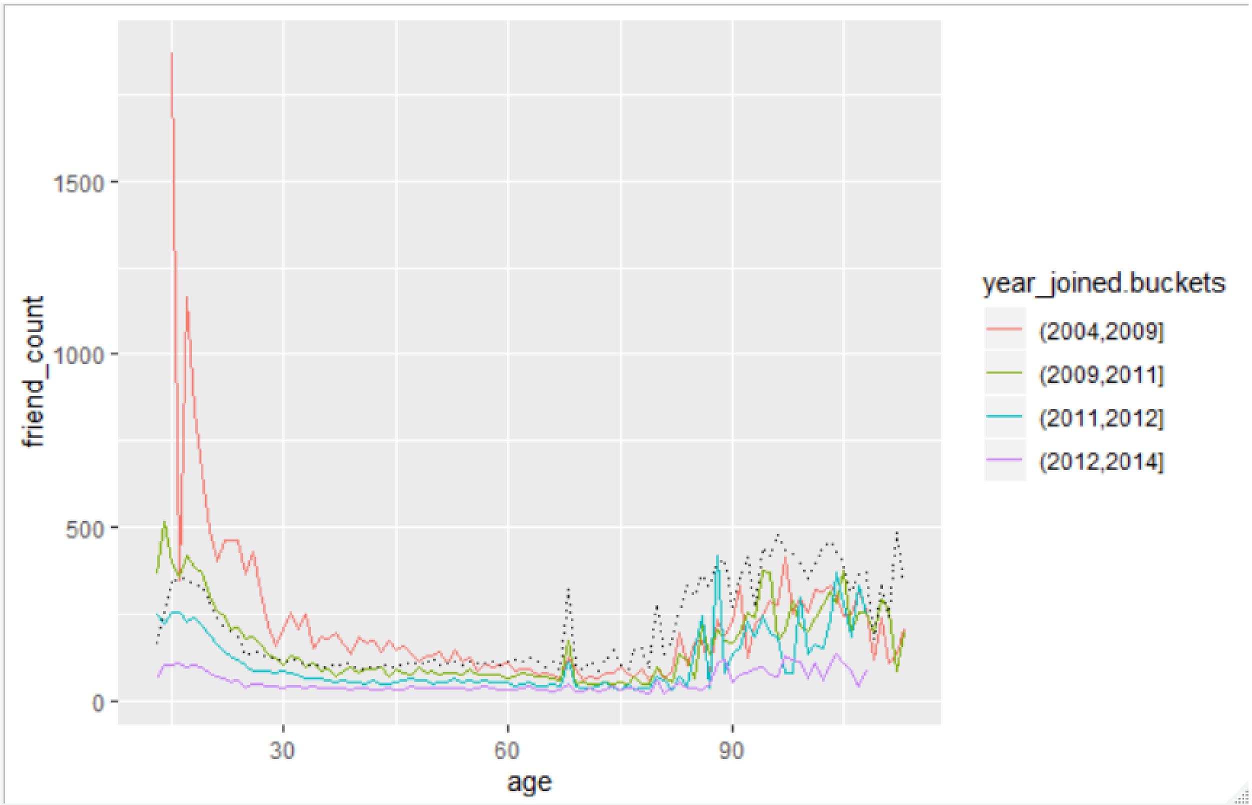
```
> # correlation between www likes and total likes
> cor.test(df_fb$www_likes_received, df_fb$likes_received)
```

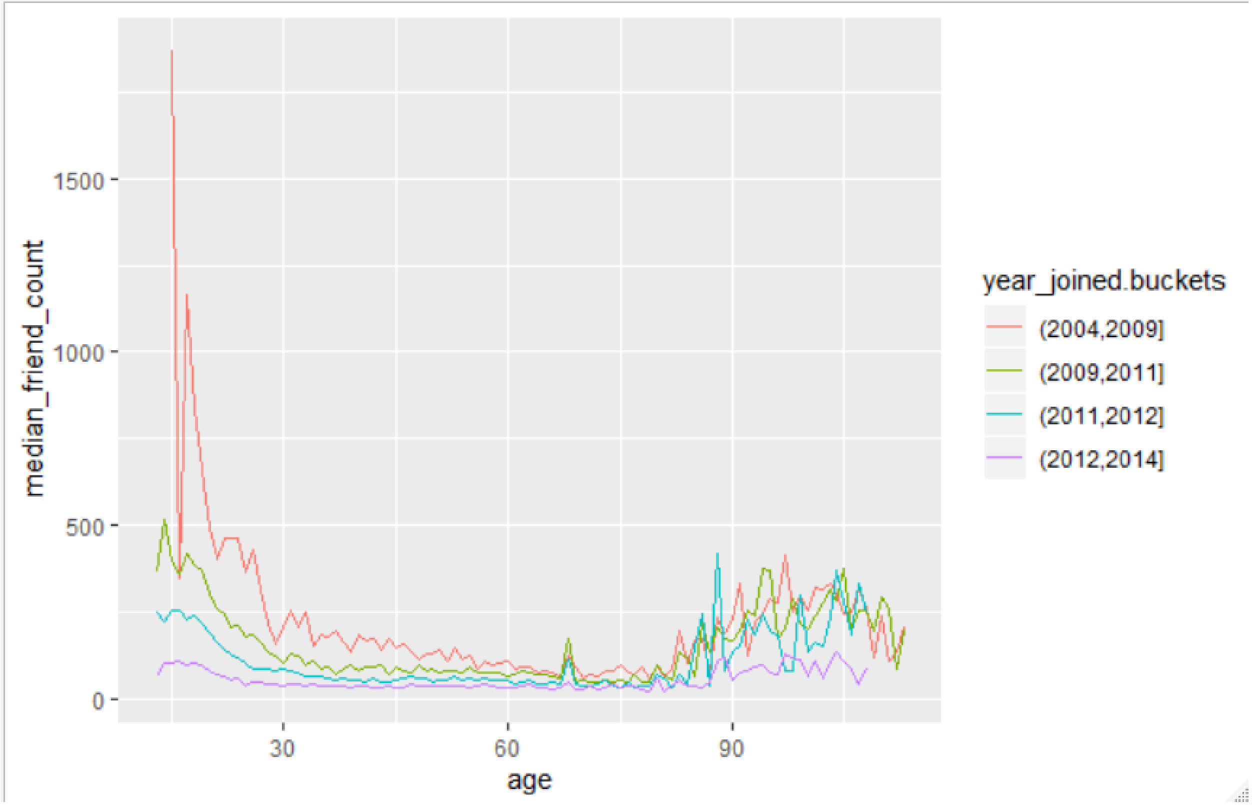
```

        Pearson's product-moment correlation

data:  df_fb$www_likes_received and df_fb$likes_received
t = 937.1, df = 99001, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.9473553 0.9486176
sample estimates:
      cor 
0.9479902 
```

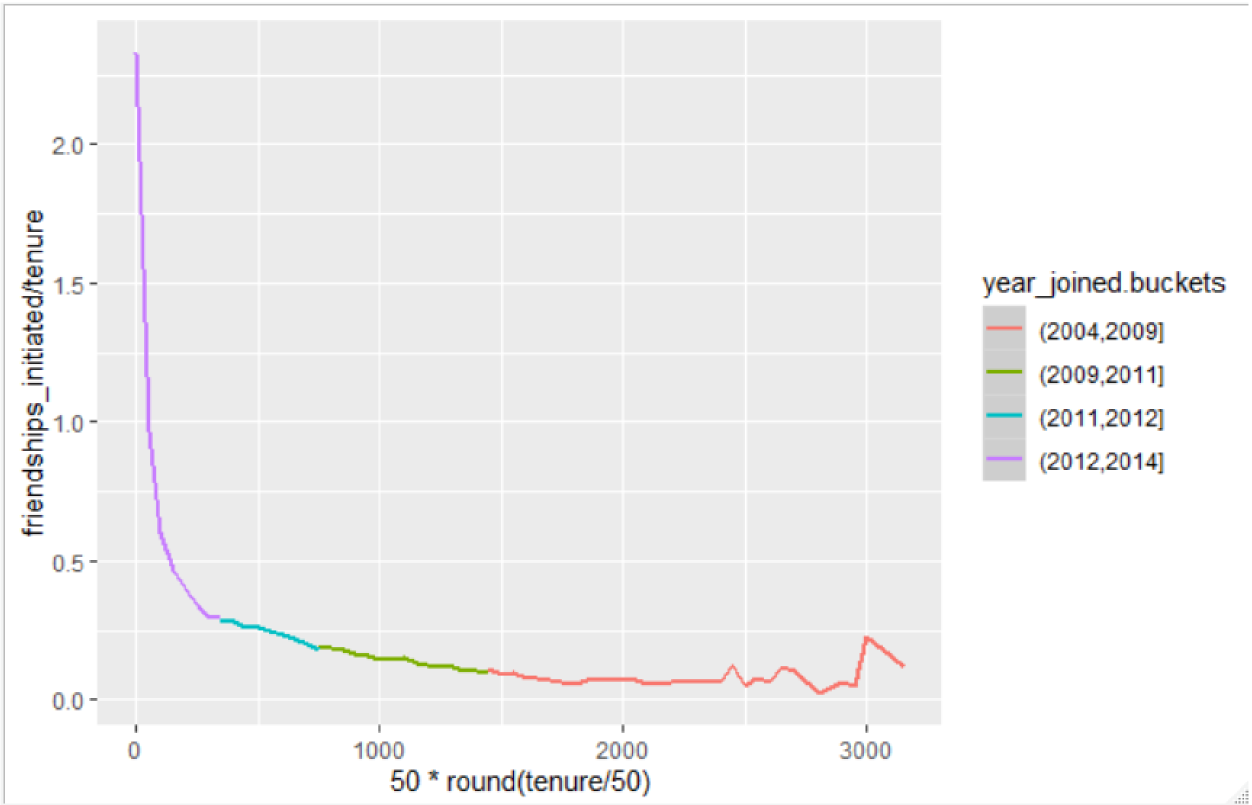


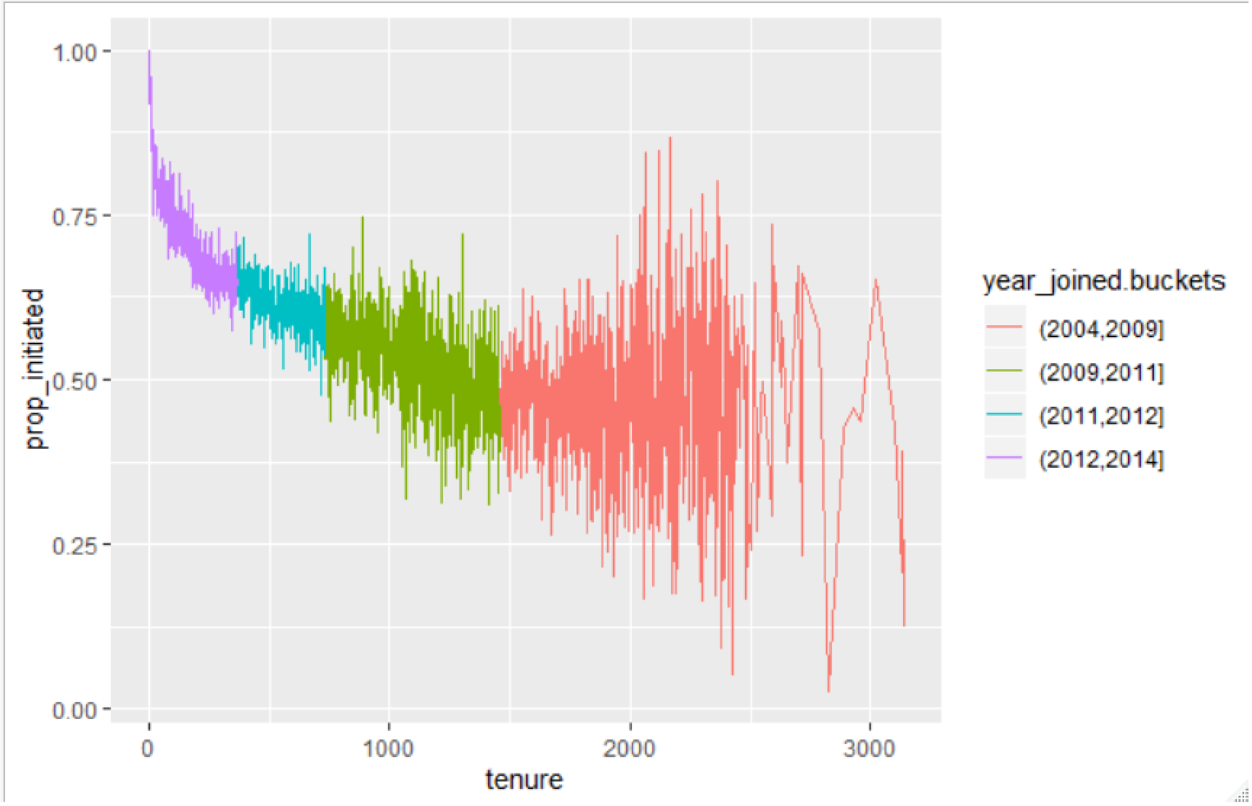

friend_count
300
200
100
30
60
90
age
gender
female
male


female/male
2
1
30
60
90
age


friend_count
1500
1000
500
0
30
60
90
age
year_joined.buckets
(2004,2009]
(2009,2011]
(2011,2012]
(2012,2014]


median_friend_count
1500
1000
500
0
30
60
90
age
year_joined.buckets
(2004,2009]
(2009,2011]
(2011,2012]
(2012,2014]


friendships_initiated/tenure
2.0
1.5
1.0
0.5
0.0
0
1000
2000
3000
50 * round(tenure/50)
year_joined.buckets
(2004,2009]
(2009,2011]
(2011,2012]
(2012,2014]


prop_initiated
1.00
0.75
0.50
0.25
0.00
0
1000
2000
3000
tenure
year_joined.buckets
(2004,2009]
(2009,2011]
(2011,2012]
(2012,2014]