EDA Lesson 3

Tom Gocken

Saturday, January 23, 2016

### What to Do First

# https://s3.amazonaws.com/udacity-hosted-downloads/ud651/EDA\_Course\_Materials.zip  
# Set directory, load data  
  
setwd("C:/Projects/UD651")  
list.files()

## [1] "EDA\_lesson3.docx" "EDA\_lesson3.html" "EDA\_lesson3.Rmd"   
## [4] "EDA\_lesson4.html" "EDA\_lesson4.Rmd" "pseudo\_facebook.tsv"  
## [7] "UD651.Rproj"

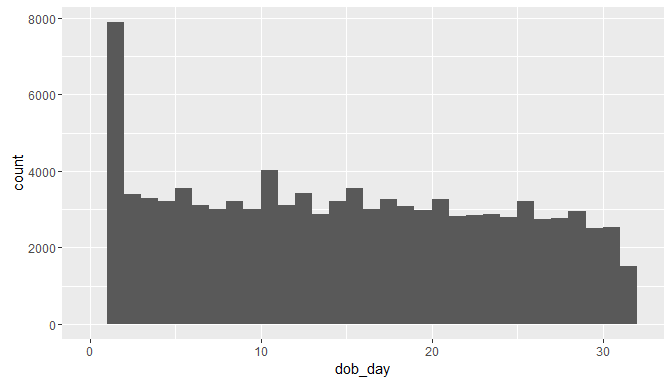
pf <- read.csv('pseudo\_facebook.tsv', sep = '\t')  
names(pf)

## [1] "userid" "age"   
## [3] "dob\_day" "dob\_year"   
## [5] "dob\_month" "gender"   
## [7] "tenure" "friend\_count"   
## [9] "friendships\_initiated" "likes"   
## [11] "likes\_received" "mobile\_likes"   
## [13] "mobile\_likes\_received" "www\_likes"   
## [15] "www\_likes\_received"

### Histogram of Users Birthdays

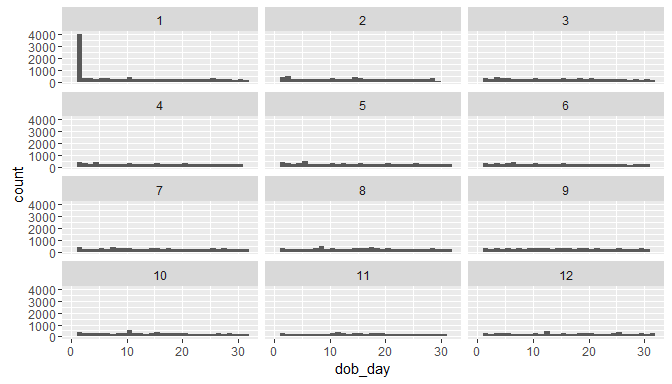
# install.packages('ggplot2')  
library(ggplot2)  
qplot(x = dob\_day, data = pf) +  
 #scale\_x\_discrete(breaks = 1:31) + # RETURNS ERROR  
 geom\_histogram(binwidth = 1, bins = 31)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



### Faceting  
# Facet by month  
qplot(x = dob\_day, data = pf) +  
 #scale\_x\_discrete(breaks = 1:31) +  
 facet\_wrap(~dob\_month, ncol = 3)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



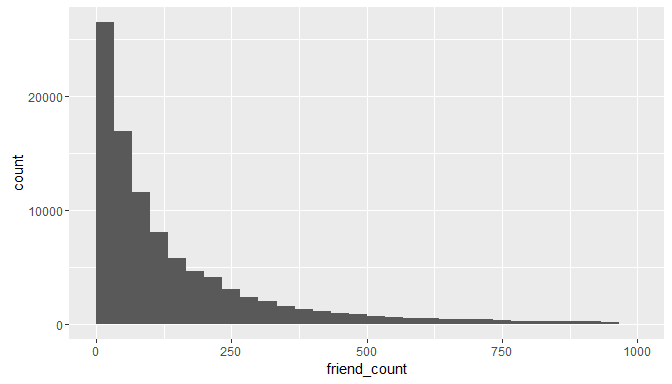
### Perceived Audience Size

# Friend count histograms  
# Method 1:  
# qplot(x = friend\_count, data = pf, xlim = c(0, 1000))  
  
# Method 2:  
qplot(x = friend\_count, data = pf) +  
 scale\_x\_continuous(limits = c(0, 1000))

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 2951 rows containing non-finite values (stat\_bin).

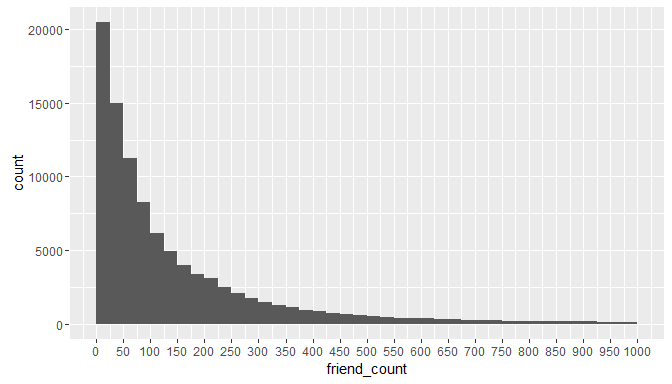
## Warning: Removed 2 rows containing missing values (geom\_bar).



# Add binwidth, adjust breaks  
qplot(x = friend\_count, data = pf, binwidth = 25) +  
 scale\_x\_continuous(limits = c(0, 1000), breaks = seq(0, 1000, 50))

## Warning: Removed 2951 rows containing non-finite values (stat\_bin).

## Warning: Removed 2 rows containing missing values (geom\_bar).

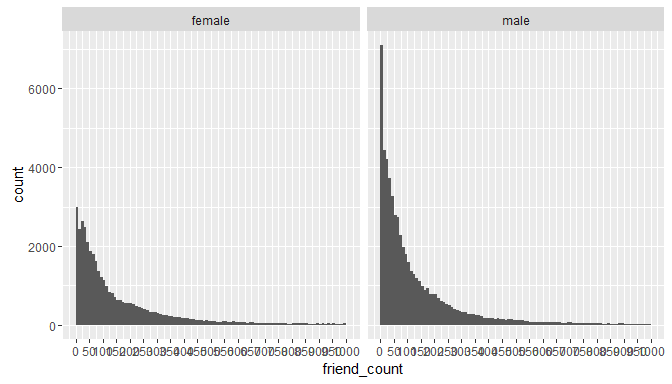


### Friend Count

# Facet by gender  
qplot(x = friend\_count, data = na.omit(pf), binwidth = 10) +  
 scale\_x\_continuous(limits = c(0, 1000), breaks = seq(0, 1000, 50)) +  
 facet\_wrap(~gender)

## Warning: Removed 2949 rows containing non-finite values (stat\_bin).

## Warning: Removed 4 rows containing missing values (geom\_bar).



table(pf$gender)

##   
## female male   
## 40254 58574

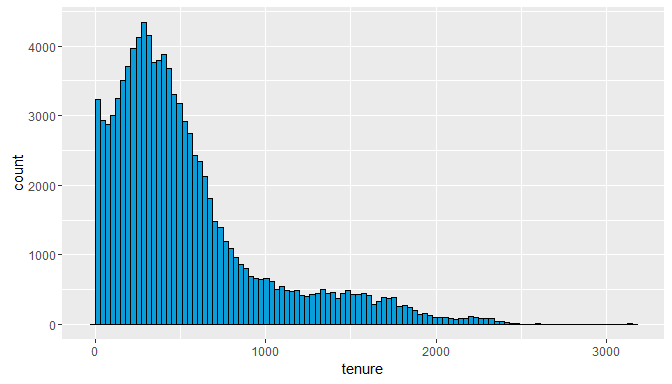
by(pf$friend\_count, pf$gender, summary)

## pf$gender: female  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0 37 96 242 244 4923   
## --------------------------------------------------------   
## pf$gender: male  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0 27 74 165 182 4917

### Tenure

qplot(x = tenure, data = pf, binwidth = 30,  
 color = I('black'), fill = I('#099DD9'))

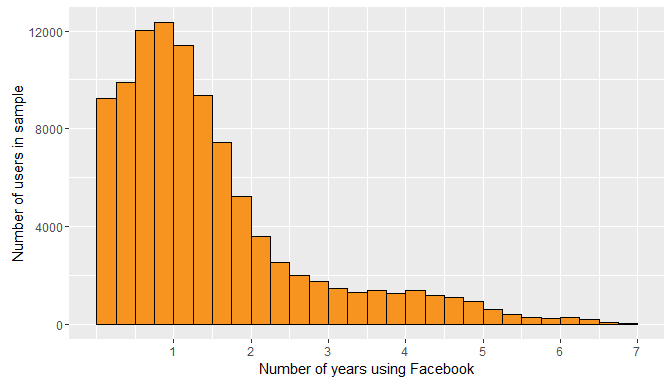
## Warning: Removed 2 rows containing non-finite values (stat\_bin).



# by year  
qplot(x = tenure/365, data = pf, binwidth = .25,  
 xlab = 'Number of years using Facebook',  
 ylab = 'Number of users in sample',  
 color = I('black'), fill = I('#F79420')) +  
 scale\_x\_continuous(breaks = seq(1, 7, 1), lim = c(0, 7))

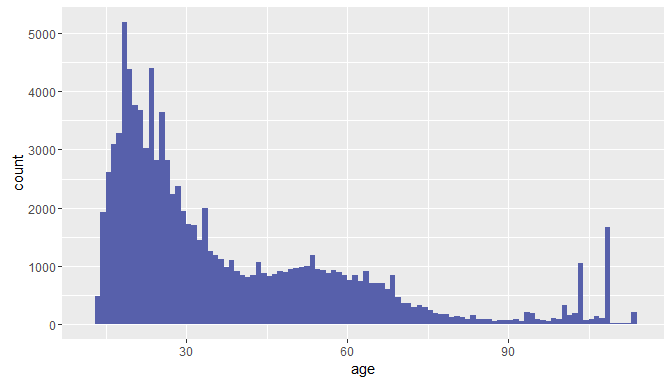
## Warning: Removed 26 rows containing non-finite values (stat\_bin).

## Warning: Removed 2 rows containing missing values (geom\_bar).



### User Ages

qplot(x = age, data = pf, binwidth = 1,  
 fill = I('#5760AB')) #+



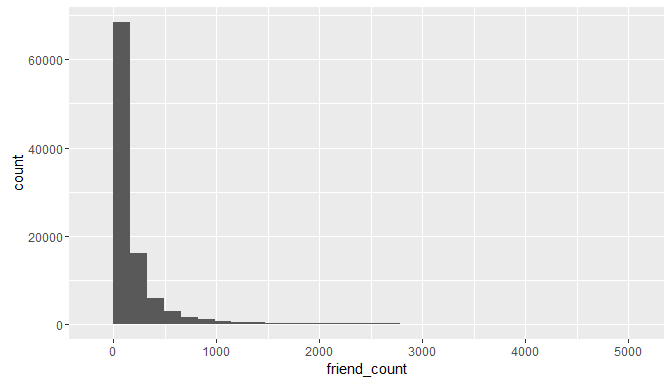
# scale\_x\_discrete(breaks = seq(0, 113, 5)) # RETURNS ERROR  
  
summary(pf$age)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 13.00 20.00 28.00 37.28 50.00 113.00

### Transforming Data

qplot(x = friend\_count, data = pf)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



summary(pf$friend\_count)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.0 31.0 82.0 196.4 206.0 4923.0

summary(log10(pf$friend\_count + 1))

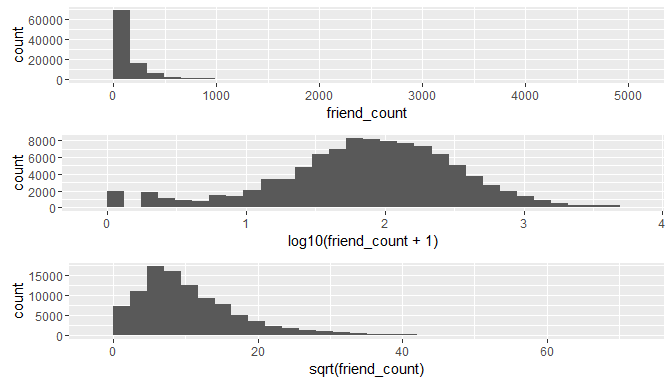
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.000 1.505 1.919 1.868 2.316 3.692

summary(sqrt(pf$friend\_count))

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.000 5.568 9.055 11.090 14.350 70.160

# gridExtra used to create multiple graphs per page  
# install.packages('gridExtra')  
  
library(gridExtra)  
  
p1 <- qplot(x = friend\_count, data = pf)  
p2 <- qplot(x = log10(friend\_count + 1), data = pf)  
p3 <- qplot(x = sqrt(friend\_count), data = pf)  
  
grid.arrange(p1, p2, p3, ncol = 1)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

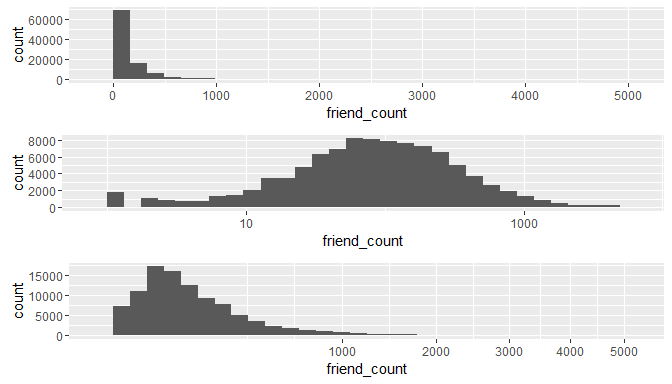


# Alternative method  
p1 <- ggplot(aes(x = friend\_count), data = pf) + geom\_histogram()  
p2 <- p1 + scale\_x\_log10()  
p3 <- p1 + scale\_x\_sqrt()  
  
grid.arrange(p1, p2, p3, ncol = 1)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 1962 rows containing non-finite values (stat\_bin).

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



# Transformation using a wrapper versus using a scaling layer  
logScale <- qplot(x = log10(friend\_count), data = pf)  
countScale <- ggplot(aes(x = friend\_count), data = pf) + geom\_histogram() +  
 geom\_histogram() +  
 scale\_x\_log10()  
  
grid.arrange(logScale,countScale, ncol = 2)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

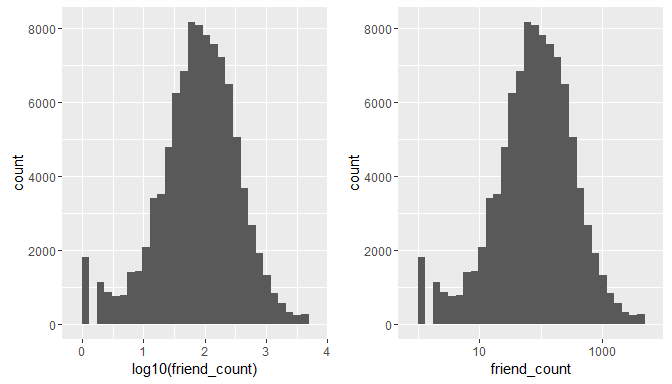
## Warning: Removed 1962 rows containing non-finite values (stat\_bin).

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 1962 rows containing non-finite values (stat\_bin).

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 1962 rows containing non-finite values (stat\_bin).

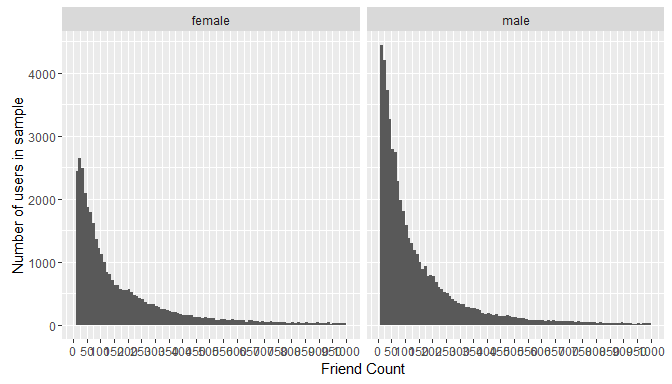


### Frequency Polygons

# Comparison of histogram to frequency polygon  
  
qplot(x = friend\_count , data = subset(pf,!is.na(gender)),binwidth = 10,  
 xlab = 'Friend Count',  
 ylab = 'Number of users in sample'  
 ) +  
 scale\_x\_continuous(breaks = seq(0,1000,50), limits = c(10,1000)) +  
 facet\_wrap(~gender)

## Warning: Removed 13040 rows containing non-finite values (stat\_bin).

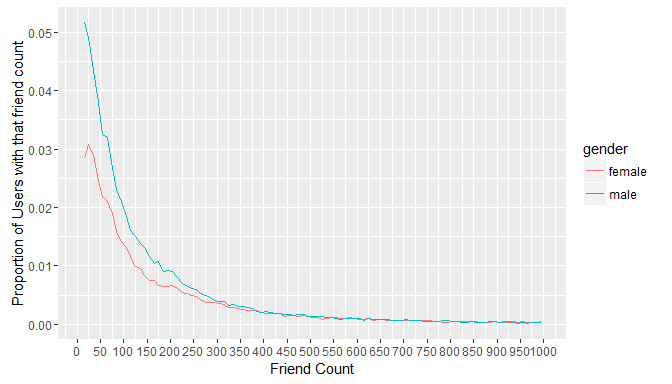
## Warning: Removed 4 rows containing missing values (geom\_bar).



qplot(x = friend\_count , y = ..count../sum(..count..),  
 data = subset(pf,!is.na(gender)),binwidth = 10,  
 geom = 'freqpoly', color = gender,  
 xlab = 'Friend Count',  
 ylab = 'Proportion of Users with that friend count'  
 ) +  
 scale\_x\_continuous(breaks = seq(0,1000,50), limits = c(10,1000))

## Warning: Removed 13040 rows containing non-finite values (stat\_bin).

## Warning: Removed 4 rows containing missing values (geom\_path).

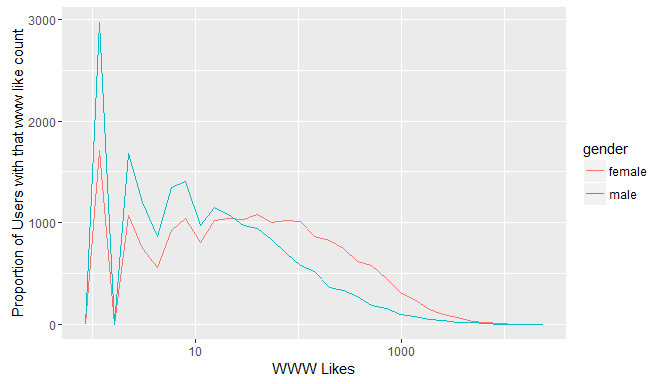


### Likes on the Web

qplot(x = www\_likes,  
 data = subset(pf,!is.na(gender)),  
 geom = 'freqpoly', color = gender,  
 xlab = 'WWW Likes',  
 ylab = 'Proportion of Users with that www like count'  
 ) +  
 scale\_x\_continuous() +  
 scale\_x\_log10()

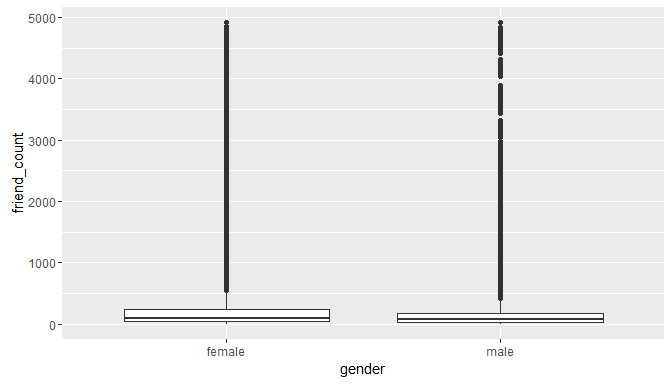
## Scale for 'x' is already present. Adding another scale for 'x', which  
## will replace the existing scale.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 60935 rows containing non-finite values (stat\_bin).

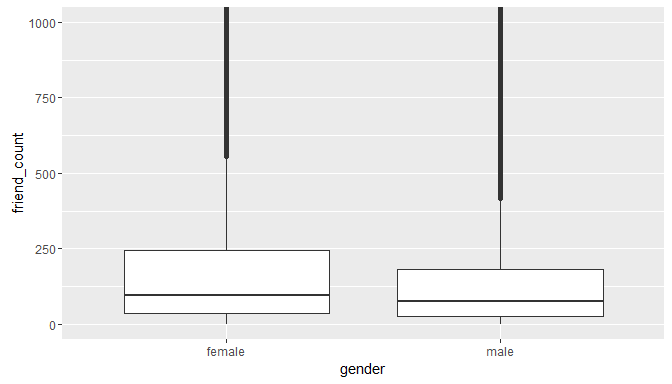


### Boxplots

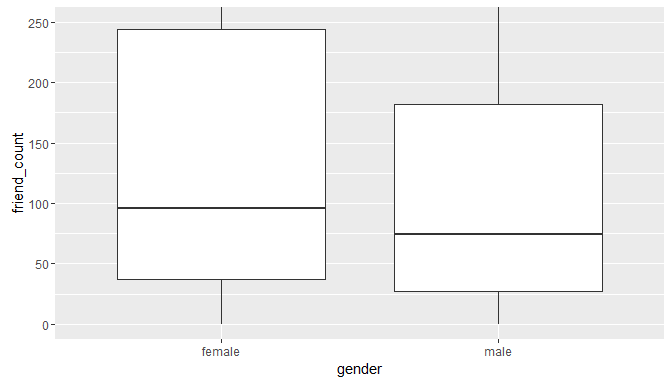
qplot(x = gender, y = friend\_count,   
 data = subset(pf, !is.na(gender)),   
 geom = 'boxplot')



# Adjustments to y axis  
# coor\_cartesian to prevent removal of values  
qplot(x = gender, y = friend\_count,   
 data = subset(pf, !is.na(gender)),   
 geom = 'boxplot') +  
 coord\_cartesian(ylim = c(0,1000))



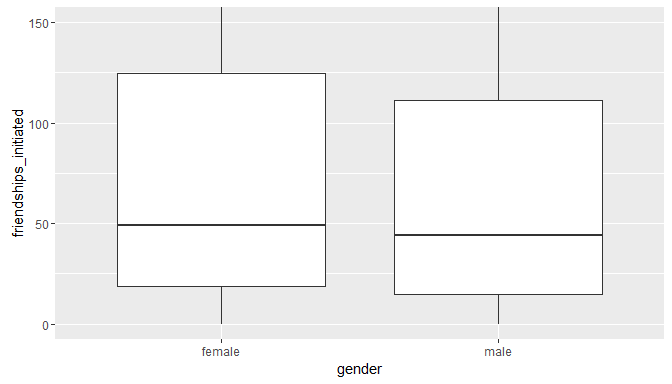
# Zoom in to view user counts <= 250  
qplot(x = gender, y = friend\_count,   
 data = subset(pf, !is.na(gender)),   
 geom = 'boxplot') +  
 coord\_cartesian(ylim = c(0,250))



by(pf$friend\_count, pf$gender, summary)

## pf$gender: female  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0 37 96 242 244 4923   
## --------------------------------------------------------   
## pf$gender: male  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0 27 74 165 182 4917

# Determine who initiates more friend counts  
qplot(x = gender, y = friendships\_initiated,  
 data = subset(pf, !is.na(gender)), geom = 'boxplot') +  
 coord\_cartesian(ylim = c(0, 150))



by(pf$friendships\_initiated, pf$gender, summary)

## pf$gender: female  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.0 19.0 49.0 113.9 124.8 3654.0   
## --------------------------------------------------------   
## pf$gender: male  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.0 15.0 44.0 103.1 111.0 4144.0

### Getting Logical

summary(pf$mobile\_likes > 0)

## Mode FALSE TRUE NA's   
## logical 35056 63947 0

# Create variable, populate with NA's  
pf$mobile\_check\_in <- NA  
  
# Update based on mobile likes  
pf$mobile\_check\_in <- ifelse(pf$mobile\_likes > 0,1,0)  
  
# Convert to factor  
pf$mobile\_check\_in <- factor(pf$mobile\_check\_in)  
summary(pf$mobile\_check\_in)

## 0 1   
## 35056 63947

# Determine percent mobile check-in  
sum(pf$mobile\_check\_in == 1) / length(pf$mobile\_check\_in)

## [1] 0.6459097