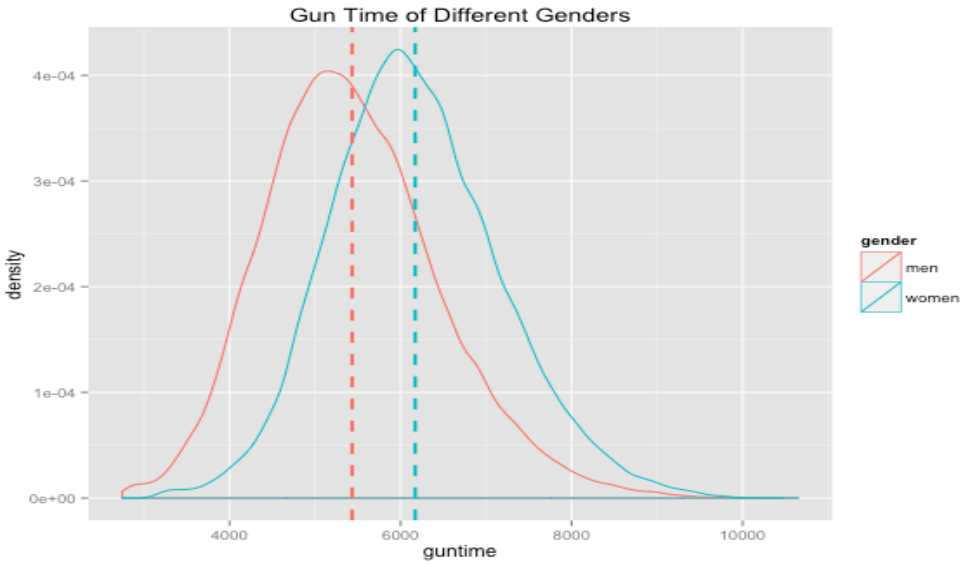
Cherry Blossom Data Analysis Report

The assignment provides us with 24 files containing Cherry Blossom Run data from 1999 to 2010 of both men and women runners. I first read these files into R, then created four functions: readFile, getBody; splitBody; and buildData. These four functions have the same input: file name of the file. Then I wrote several small functions like getYear, getGender, to.secs and getState to transform the variable into form we need or to add new variables that can be useful to my Analysis. After applying all the files with these functions, I merged these 24 data frames into one large data frame called: cherryrun, for analyze part.

Aspects I am interested in are as follows:

1. Difference between Men and Women

From figure 1: the distributions of Gun Time by different gender group, men are in need of less time than women to finish the run. It is a fact that men are faster than women, in general. So the data from 12-year's Cherry Blossom Run can also be a demonstration of the fact. The reason may lies in that women carry more body fat than men and men tend to have larger organs, which may help them build a better oxygenated blood delivery system than women. The better the oxygen delivery system is, the faster the muscles can respond [1]



[1] http://www.netplaces.com/running/girls-women-just-want-to-have-fun/comparing-men-and-women-runners.htm

Figure 1 Gun Time of Different Genders

2. Performance of runners across years

Some people may hold that based on the developments of health care and the better nutrition supply, runners' performance may enhance over years. Is it true?

From Figure 2: Gun Time by Genders of Different Years, we can interpret the following aspects:

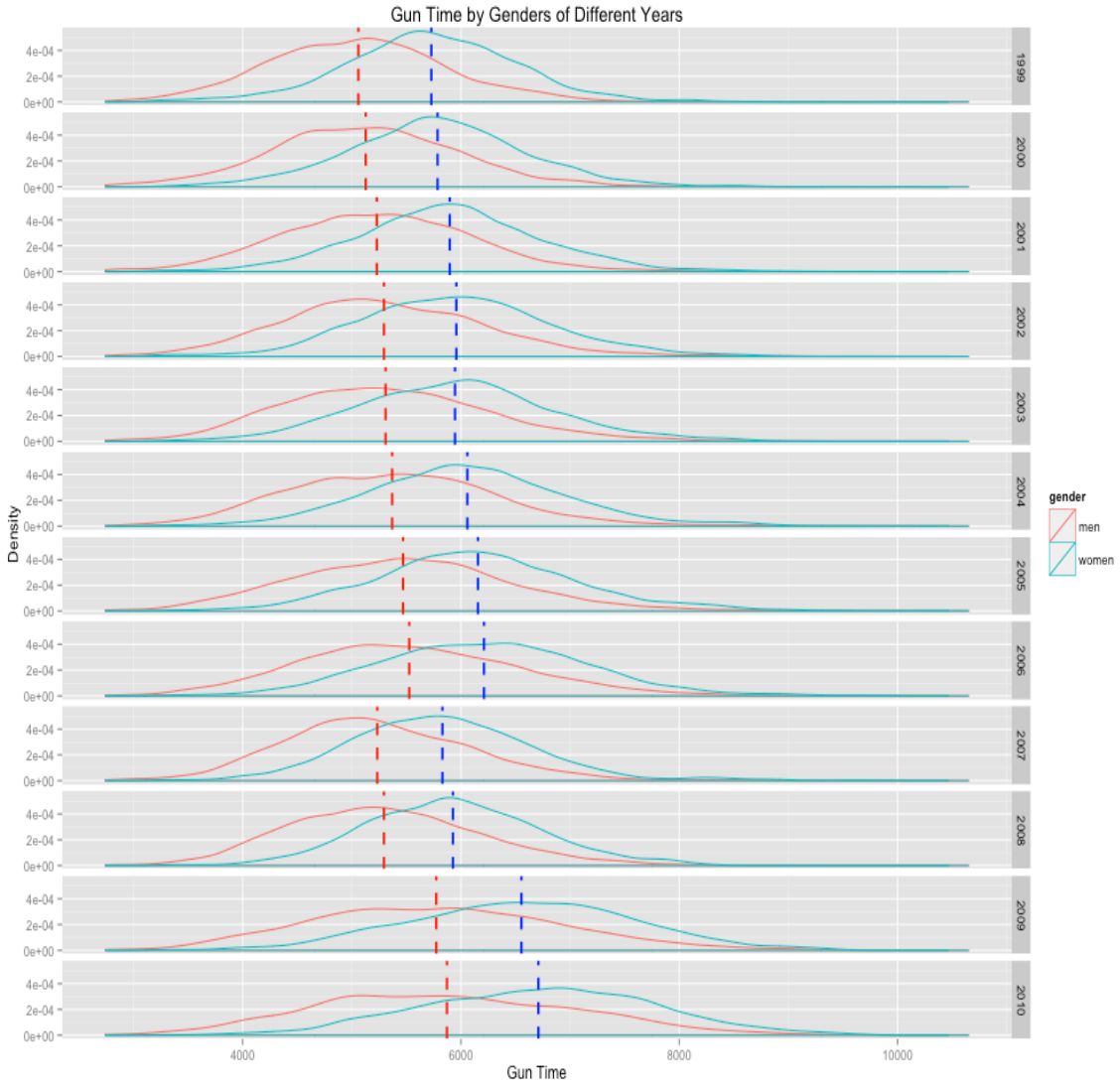


Figure 2: Gun Time by Genders of Different Years

a. In each year, the performance of men runner is better than that of women runner. The conclusion is the same as what we constructed from the data of all the years.

b. From 1999 to 2006, the performances of both women and men are regressed.

After a two-year improvement (from 2007-2008), the performances in 2009 and 2010 are worse than before. It's hard to get a reasonable explanation for this. So, we want to consider the top 100 participants to see if there are something different.

Then, we got the distribution plot of TOP 500 runners each year in each gender (see Figure 3). The regression of running time in year of 2007 and 2008 is obvious.

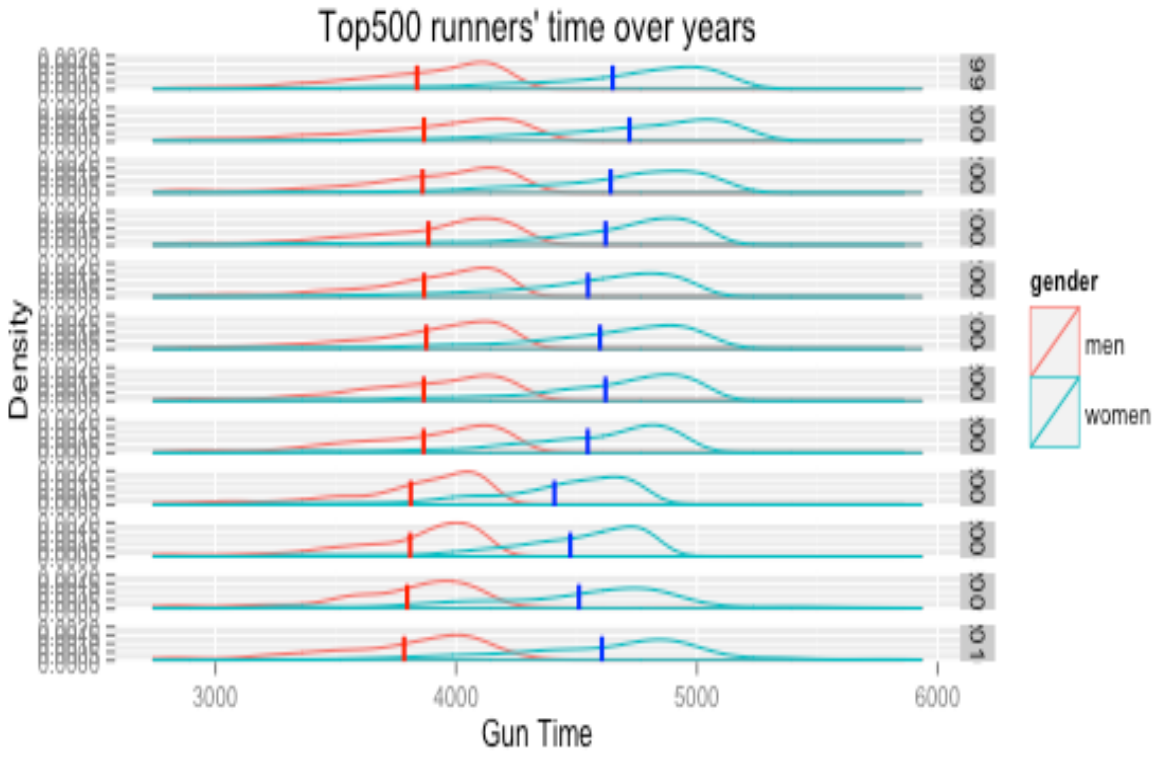


Figure 3: Top 500 Runners' time by years

3. Participants over years

Running is a kind of exercises that help people stay health through improving our immune system. With the promotion of different running events and the popular of running marathon, people got more passion in participating running events.

Table 1: Runner's number

Year	Men Runner	Women Runner	Total
1999	3190	2358	5548
2000	3016	2166	5182
2001	3561	2972	6533
2002	3723	3334	7057
2003	3946	3542	7488

2004	4156	3899	8055
2005	4324	4333	8657
2006	5235	5435	10670
2007	5274	5690	10964
2008	5905	6397	12302
2009	6649	8323	14972
2010	6909	8853	15762

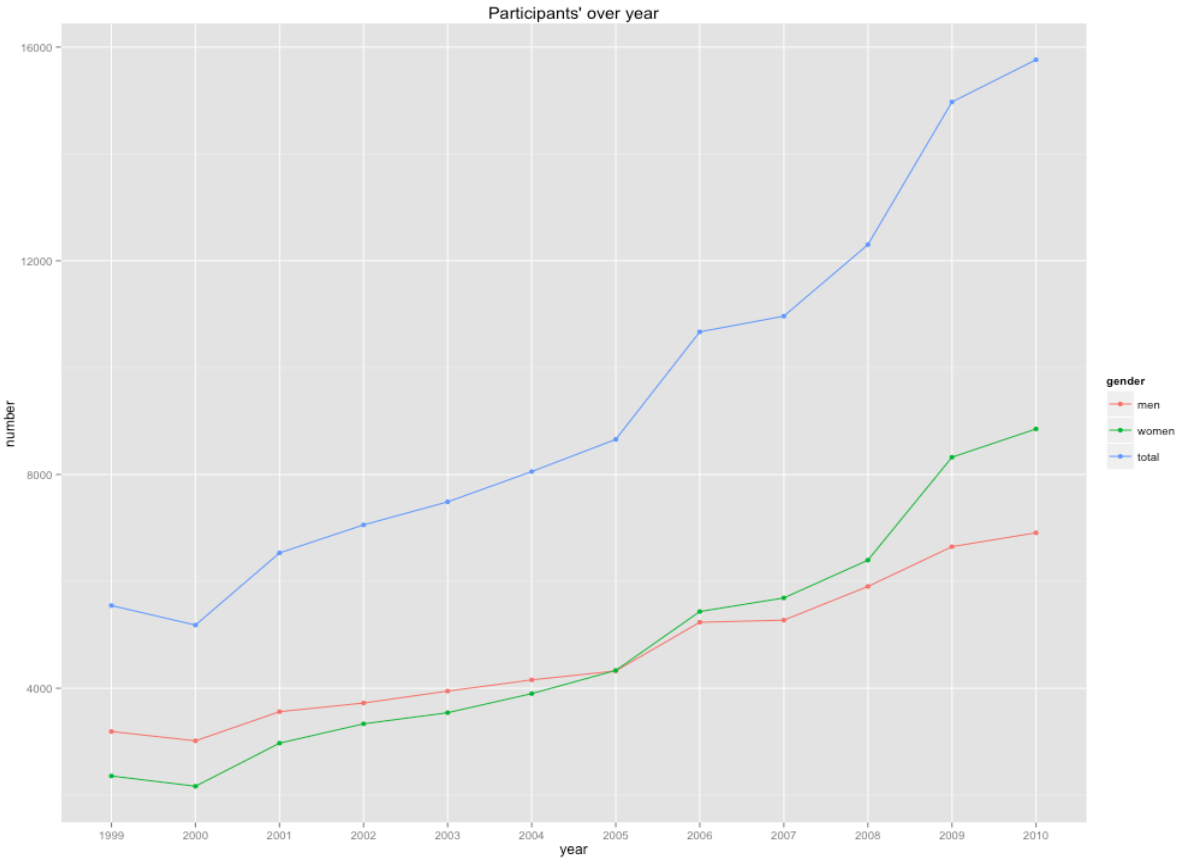


Figure 4: Participants' number over year

As Shown in Table 1 and Figure 4, Participants of Cherry Blossom Run increased every year. And both women runner and men runner are also increased. From year of 2005, number of women runner overweighs number of men runner.

4. Runner from USA

In most files, if the runner is from USA, then the hometown variable contains the information of the city and state he comes from. Then, we can create a new variable to get which state the runner is from. Then, we can find some interesting things related to their hometown.

Table 2 Top 10 home states of runners

Rank	1	2	3	4	5	6	7	8	9	10
State	VA	MD	DC	PA	NY	NJ	NC	MA	OH	CT
Number	40227	27028	20230	3320	3086	1396	914	767	439	429

Since Cherry Blossom Run was held at Washington D.C., it is reasonable that most of the runners are from states nearby.

Table 3 Top 10 in speed of runners' home state

Rank	1	2	3	4	5	6	7	8	9	10
State	ID	UT	VT	CO	NV	WA	WV	DE	MT	WI
Time	5368.4	5391.652	5421.5	5482.029	5542.889	5587.403	5630.383	5637.687	5646.75	5656.388

From Table 3, we can find out which states' people run the fastest. From these data, the answer comes out to be ID, which means Idaho. And it is interesting to find out that the states have most people are not those states who has the faster runner.

5. Age effect

Which age period's runner share the most of the runners of the run? From Figure 6, most runners are 20 – 30 year's old. And the age range of female runners' is wider than that of males.

Figure 6: age distribution over year

How people's performance changed over years? After created a unique ID for each runner, and get those who joined the run 12 times. Then, we want to test whether age is a influential variable.

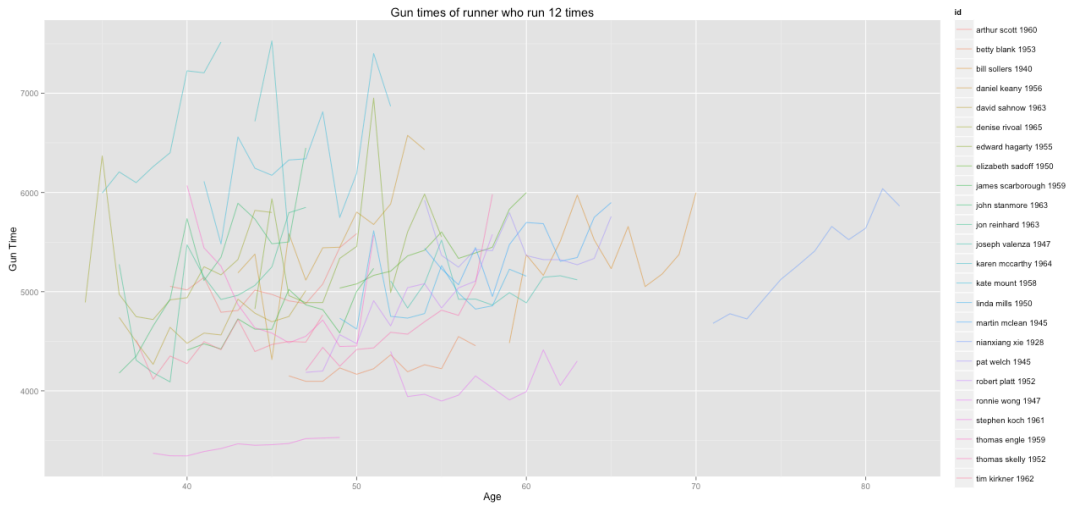


Figure 7: Gun Times of runners who run 12 times

From Figure 6, these people who joined every year did not improve with increase in age. I think if I want to construct a further study of age effect, I can apply ANOVA method to get more specific results.

6. Top 10 Runner's hometown

As we all know, African people run faster than other people. How's their performance in Cherry Blossom Run?

We first get each year and each gender's top 10 into a data frame. And then analyze their hometown.

We find out that most foreign top runners are from Kenya and Ethiopia. Since 103 of top 10 runners are from Kenya and 25 of top 10 runners are from Ethiopia during 1999 to 2010.

7. What's the proportion of foreign runners?

The proportion of domestic runner is larger than 90%, which means most of the runners are not from overseas. As an indicator of promotion level of an event, the promotion of Cherry Blossom Run can put emphasis on global market.

Due to the limits of time, some aspects can get some further study and more analyze in depth but didn't. I want to further study the top 10 runners pace in the first 5mile and 10 km. And find out the difference between Male and Female runners'. Also try to get a strategy through analyze the data of when to speed up to help runner run faster and save energy.

Appendix 1

```
setwd("/Users/jianshizhang/stat242_2015/Assignment1/data/")
files = list.files()

#functions to get year
getYear = function(filename){
  strsplit(filename,'10Mile_')[[1]][2]
}


#functions to get gender
getGender = function(filename){
  strsplit(filename, '10Mile_')[[1]][1]
}

#Read the file
readFile = function(filename){
  con = file(filename, open = "rt")
  text = readLines(con)
  text = gsub("[\u00A0]", " ", text)
  close(con)
  text
}


#remove the table name and other information before the data frame st
arts
getBody = function(filename){
  con = readFile(filename)
  if( length(which(con=="")) != 0){
    con = con[-which(con=="")]
  }
  location1 = grep('===', con, useBytes=TRUE)
  location2 = grep('^#|^ #', con, useBytes = TRUE)

  #deal with file without comments
  if(length(location2)==0){
    location2 = length(con)+1
  }


  #deal with file without head
  if(length(location1)==0){
    if(grepl("^w", filename)){
```

```
      filename1 = sub("women","men",filename)
    }
    else{
      filename1 = sub("men","women",filename)
    }
    con1=readFile(filename1)
    location.1 = grep('===', con1, useBytes=TRUE)
    add.text = con1[(location.1-1):location.1]
    location.2 = grep('1 ',con, useBytes = T)[1]
    Body = add.text
    Body[3:(location2+2-location.2)] = con[location.2:(location2-1)]
  }

  #normal situation
  if(length(location1)!=0){
    Body = con[(location1-1):(location2-1)]
  }

  #to deal with men10Mile_2008
  Body = Body[!grepl("/./",Body)]

  #################deal with the Body we already get to easier our "gr
ab"

  #uniform the signal line and variable name
  Body[1] = tolower(Body[1])

  #deal with "  "
  Body[2] = gsub("  ","= ",Body[2])

  #deal with"=========="
  position = gregexpr("net", Body[1])[[1]]
  signal.c = substr(Body[2], position-4, position-1)
  if(signal.c == "===="){
    replace.1 = substr(Body[2], 1, position-2)
    replace.2 = " "
    replace.3 = substr(Body[2],position,nchar(Body[2]))
    replace = paste0(replace.1,replace.2,replace.3)
    Body[2] = replace
  }
  #ending " " to add a = and replace the last " "to =
  Body[2] = paste0(substr(Body[2], 1, nchar(Body[2])-1),"==")

  #remove the space and vacant lines
```

```
  #grep("^\\s+$", Body)
  Body = Body[!grepl("^\\s+$|''", Body)]
  Body
}

#split the Body and build the data frame
splitBody = function(filename){
  temp = getBody(filename)
  signal = temp[2]
  context = temp[1]
  index = gregexpr(' ', signal)[[1]]
  index.start = c(1, index+1)
  index.end = c(index, 1000000L)
  z = substring(context, first = index.start, last = index.end)
  var.name = gsub("\\s+","",z)
  var.name[var.name %in% c("guntim", "gun","time")] = "guntime"
  var.name[var.name %in% c("nettim", "net")] = "nettime"
  var.name[var.name %in% c("ag")] = "age"
  var.name[var.name %in% c("5mi","5mile")] = "5mi"
  Body = matrix(rep(NA, (length(index)+1)*(length(temp)-2)),
                nrow = length(temp)-2)
  colnames(Body) = var.name

  #deal with pace name
  colnames(Body)[which(colnames(Body)=="pace")][-sum(colnames(Body)=
="pace")] =
    paste0(rep("pace",n = sum(colnames(Body)=="pace")-1),seq(1:c(sum
(colnames(Body)=="pace")-1)))

  for(i in 3:length(temp)){
  Body[i-2, ]= substring(temp[i], first = index.start, last = index.e
nd)
  }

  #get rid of spaces
  Body = gsub("^\\s+|\\s+$","",Body)

  #divided the div/tot
  temptext = Body[,which(colnames(Body)=='div/tot')]
  Body[,which(colnames(Body)=='div/tot')][temptext==""] = "/"
  text = read.table(text = Body[,which(colnames(Body)=='div/tot')],
                    sep = "/", col.names = c('div','total'),fill = T)
```

```
  #creat a new variable named identity
  if(sum(colnames(Body)=='nettime')==0){
    te = Body[,which(colnames(Body)=='guntime')]
    ind = grepl("\\#|\\*", te)
    identity = vector(length = length(ind))
    identity[ind==T] = substr(te, nchar(te),nchar(te))[ind==T]
    identity[ind!=T] = ""
    Body[,which(colnames(Body)=='guntime')] =
      gsub("\\#|\\*", "",  Body[,which(colnames(Body)=='guntime')])
  }
  if(sum(colnames(Body)=='nettime')!=0){
    te = Body[,which(colnames(Body)=='guntime')]
    tex = Body[,which(colnames(Body)=='nettime')]
    ind = grepl("\\#|\\*", te)
    ind.1 = grepl("\\#|\\*", tex)
    identity = vector(length = length(ind))
    identity[ind==T] = substr(te, nchar(te),nchar(te))[ind==T]
    identity[ind.1==T] = substr(tex, nchar(tex),nchar(tex))[ind.1==T]
    identity[ind!=T & ind!=T] = ""
    Body[,which(colnames(Body)=='guntime')] = gsub("\\#|\\*", "",  Bo
dy[,which(colnames(Body)=='guntime')])
    Body[,which(colnames(Body)=='nettime')] = gsub("\\#|\\*", "",  Bo
dy[,which(colnames(Body)=='nettime')])
  }

  Body = as.data.frame(Body)
  if(dim(text)[1]!=0){
    dat = cbind(Body[,which(colnames(Body)!='div/tot')], text, ident
ity)
  }
  if(dim(text)[1]==0){
    dat = cbind(Body[,which(colnames(Body)!='div/tot')], identity)
  }

  dat[dat==""]  = NA
  dat
}

#methond from http://www.macalester.edu/~kaplan/startingwithr/panel-
data.pdf
to.secs = function(time){
  secs = rep(0,length(time))
  if(length(time)!=0){
    time = as.character(time)
```

```
    for (k in 1:length(time)) {
      s = strsplit(time[k], ":")[[1]]
      s = as.numeric(s)
      if(length(s)==3){
        secs[k] = s[1]*3600 + s[2]*60 + s[3]
      }
      else{
        secs[k] = s[1]*60 + s[2]
      }
  }
  time = secs
}
  time
}

#if a person is from USA, get the state where he from
getState = function(hometown){
   temp = strsplit(as.character(hometown),' ')
   state = rep(NA, length(hometown))
   tmp = c()
   for(i in 1:length(temp)){
     tmp[i] = temp[[i]][length(temp[[i]])]
     states = c(state.abb,"DC")
     index = match(toupper(tmp[i]),states)
     if(!is.na(index)){
       state[i] = states[index]
     }
   }
   state
}
```

Appendix 2

```
#part 1 data cleaning

#Build the final analyze data by adding several variables
buildData = function(filename){
  dat = splitBody(filename)
  ###variables we care about
  #add year and gender
  year = rep(getYear(filename),dim(dat)[1])
  gender = rep(getGender(filename), dim(dat)[1])
  dat$year = as.integer(year)
  dat$gender = as.factor(gender)
  dat$state = getState(dat$hometown)
  #transform the time (only includes what we care about for analysis)
  dat[,which(colnames(dat)=='guntime')] = to.secs(dat[,which(colname
s(dat)=='guntime')])
  dat[,which(colnames(dat)=='nettime')] = to.secs(dat[,which(colname
s(dat)=='nettime')])
  dat[,which(colnames(dat)=='pace')] = to.secs(dat[,which(colnames(d
at)=='pace')])
  dat
}

#set the class of data and merge it into one, drop variables that we
are not interested in.
ss = sapply(files,buildData)
cherryrun = Reduce(function(x, y) merge(x, y, all=TRUE), ss)
tttt = Reduce(function(x, y) merge(x, y, all=TRUE), ss)
cherryrun = tttt
variable.care = c("year","gender","place","num","name","age","hometo
wn","guntime","nettime","identity","div","total","pace","state")
cherryrun = cherryrun[,variable.care]
cherryrun$place = as.factor(as.character(cherryrun$place))
cherryrun$year = as.factor(as.character(cherryrun$year))
cherryrun$age = as.factor(as.character(cherryrun$age))
cherryrun$num = as.factor(as.character(cherryrun$num))
cherryrun$name = as.character(cherryrun$name)
cherryrun$hometown = as.character(cherryrun$hometown)
cherryrun$state = as.factor(cherryrun$state)
cherryrun = cherryrun[order(cherryrun$year,cherryrun$gender,cherryru
n$place),]

#save the data into txt file
setwd("..")
write.table(cherryrun, "cherryrun.txt",sep="\t")
```

```r
#Part 2 data analyze

setwd("/Users/jianshizhang/stat242_2015/Assignment1")
cherryrun = read.table("cherryrun.txt",header = T, sep = "\t")
cherryrun$total = as.integer(cherryrun$total)
library(plyr)
library(ggplot2)
#code from R cookbook

#to get a plot of guntime by different gender
guntime.mean.by.gender = ddply(cherryrun,"gender",summarise, guntime.
mean = mean(guntime,na.rm = T))
ggplot(cherryrun, aes(x=guntime, colour=gender)) +
    geom_density() +
    geom_vline(data=guntime.mean.by.gender, aes(xintercept=guntime.m
ean,  colour=gender),
               linetype="dashed", size=1)+
ggtitle("Gun Time of Different Genders")



# Gun Time by Genders of Different Years

mean.men.by.year = ddply(cherryrun[cherryrun$gender == "men", ], "ye
ar", summarise, guntime.men.mean = mean(guntime, na.rm = T))
mean.women.by.year = ddply(cherryrun[cherryrun$gender == "women", ],
 "year", summarise, guntime.women.mean = mean(guntime, na.rm = T))
ggplot(cherryrun, aes(x = guntime, color = gender)) +
  geom_density(ma.rm = T)+
   facet_grid(year ~ .) +
geom_vline(data = mean.men.by.year, aes(xintercept = guntime.men.mea
n),
linetype = 'dashed', size = 1, color = 'red') +
geom_vline(data = mean.women.by.year, aes(xintercept = guntime.women.
mean),
linetype = 'dashed', size = 1, color = 'blue') +
xlab("Gun Time") + ylab("Density") +
ggtitle("Gun Time by Genders of Different Years")

# Top 500 runner's performance by years

top500 = cherryrun[cherryrun$place %in% c(1:500),]
mean.men.by.year = ddply(top500[top500$gender == "men", ], "year", s
ummarise, guntime.men.mean = mean(guntime, na.rm = T))
mean.women.by.year = ddply(top500[top500$gender == "women", ], "year
", summarise, guntime.women.mean = mean(guntime, na.rm = T))
ggplot(top500, aes(x = guntime, color = gender)) +
```

```
  geom_density(ma.rm = T)+
   facet_grid(year ~ .) +
geom_vline(data = mean.men.by.year, aes(xintercept = guntime.men.mea
n),
linetype = 'dashed', size = 1, color = 'red') +
geom_vline(data = mean.women.by.year, aes(xintercept = guntime.women.
mean),
linetype = 'dashed', size = 1, color = 'blue') +
xlab("Gun Time") + ylab("Density") +
ggtitle("Top500 runners' time over years")

# number of participants

men.by.year = ddply(cherryrun[cherryrun$gender == "men", ], "year",
summarise, men.number = length(year))
women.by.year = ddply(cherryrun[cherryrun$gender == "women", ], "yea
r", summarise, women.number = length(year))
total.by.year = cbind(year = c(1999:2010),number = as.integer(table
(cherryrun$year)),gender = rep("total", 12))
total.by.year = as.data.frame(total.by.year)
person.number = merge(men.number, women.number, all = T)
person.number = merge(person.number, total.by.year, all = T)
person.number$number = as.integer(as.character(person.number$number))
ggplot(data=person.number, aes(x=year, y=number, group=gender, colou
r=gender)) +
  geom_line() +
  geom_point()+
  ggtitle("Participants' over year")

# Performance of US runner

data.usa = cherryrun[!is.na(cherryrun$state),]
state.mean = aggregate(guntime ~ state, data.usa, mean)
state.mean[order(state.mean$guntime),]
state.participants = aggregate(year~state, data.usa, length)
state.participants = state.participants[order(state.participants[,2],
decreasing = T),][1:10,]

# Performance's analysis related to age effect

ggplot(cherryrun)+
geom_histogram(aes(x=age, fill=gender),position="dodge")+
facet_wrap(~year, nrow = 4, scales ="free") +
 ggtitle("Age distribution over year")
data.runner = cherryrun
#create ID to indicate unique player
data.runner$yob = data.runner$year-data.runner$age
```

```
data.runner$name = toupper(data.runner$name)
data.runner$id= paste(tolower(data.runner$name), data.runner$yob)
total.runs = aggregate(year~id,data = data.runner, length)
total.runs  = total.runs[total.runs$year==12,]
brave.id = total.runs$id

#index are names of people who showed up every year
data.age = data.runner[data.runner$id %in% brave.id,]
ggplot(data.age, aes(age, guntime, group = id,colour = id )) +
geom_path(alpha = 0.5) +
xlab("Age") + ylab("Gun Time") +
ggtitle("Gun times of runner who run 12 times")
# hometown of top 10 runners each year&gender

top10 = cherryrun[cherryrun$place %in% c(1:10),]
sort(table(as.character(top10$hometown)),decreasing = T)[1:10]

# proportion of foreign runners(in general)

1-sum(is.na(cherryrun$state))/length(cherryrun$state)
```