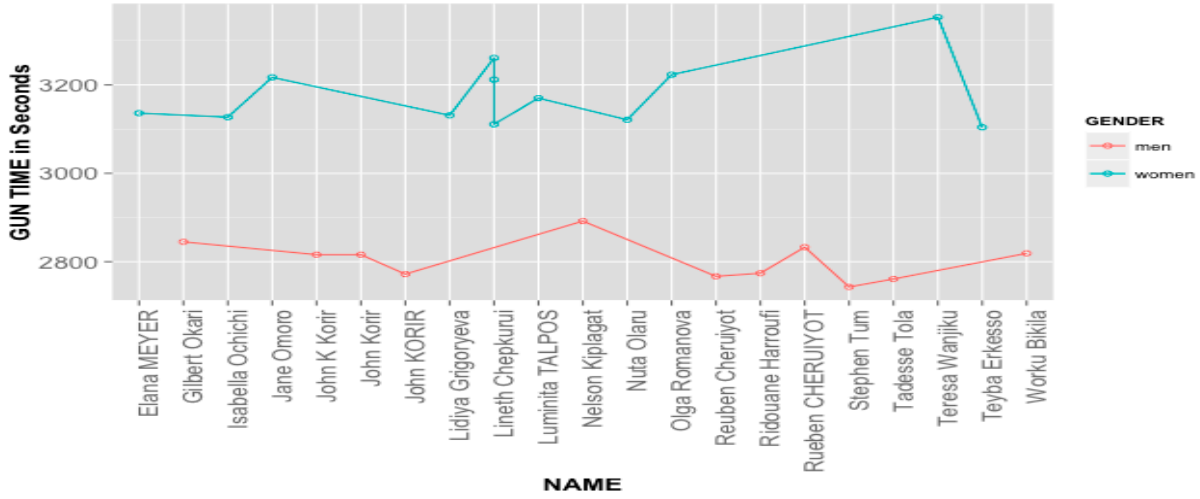
Sam Hatamian (STA242)

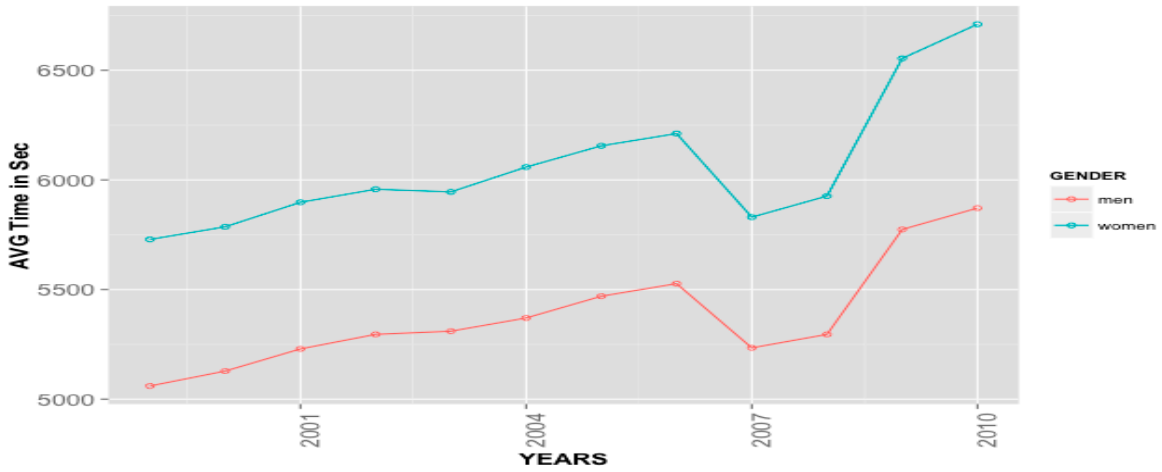
Cherry Blossom Run

For this assignment, I created a function combine() that was able to uniformly input every one of the text files containing racing information and then output a data frame. I used readLines to take in my data and from there, I used regular expressions to assign column widths and perform a multitude of other tasks necessary for processing the data. I then combined all of my information into a large data frame and picked specific columns that I wanted to analyze. On the right is the head of my original data frame.

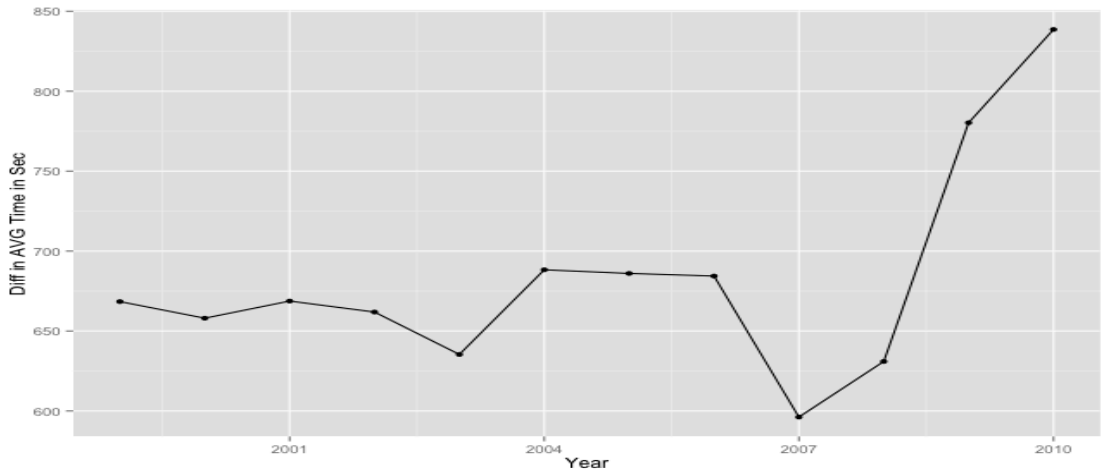
	PLACE	NAME	AG	HOMETOWN	GUN	GENDER	YEAR
men10Mile_1999.1	1	Worku Bikila	28	Ethiopia	46M 59S	men	1999
men10Mile_1999.2	2	Lazarus Nyakeraka	24	Kenya	47M 1S	men	1999
men10Mile_1999.3	3	James Kariuki	27	Kenya	47M 3S	men	1999
men10Mile_1999.4	4	William Kiptum	28	Kenya	47M 7S	men	1999
men10Mile_1999.5	5	Joseph Kimani	26	Kenya	47M 31S	men	1999
men10Mile_1999.6	6	Josphat Machuka	25	Kenya	47M 33S	men	1999

I first wanted to examine how each first place runner performed by year, separated by males and females. We can see that male times were always faster. The Y Axis GUN is denoted in seconds and for the rest of the report the analysis will look at time in seconds. We also see that the number of seconds each year for first place finishers stays generally consistent.

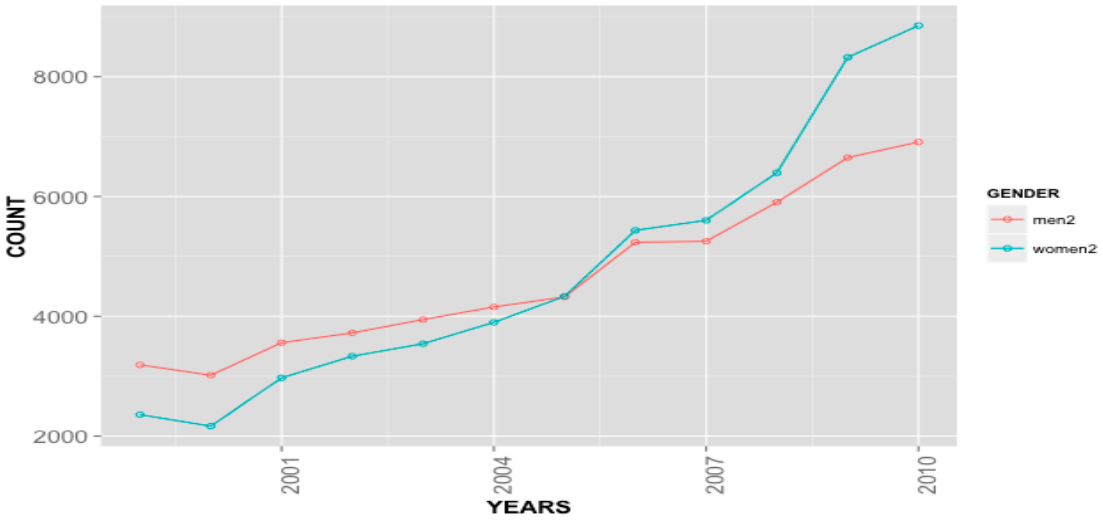




We also see that the average time in seconds for men and females seems to vary fairly closely to one another each year with a dip around 2007. This seems peculiar but generally the trend is increasing so thus I am assuming that perhaps the count of those entering each marathon is increasing each year which is worsening the average times to finish since there are more people. Below I plot the difference between the two average times.

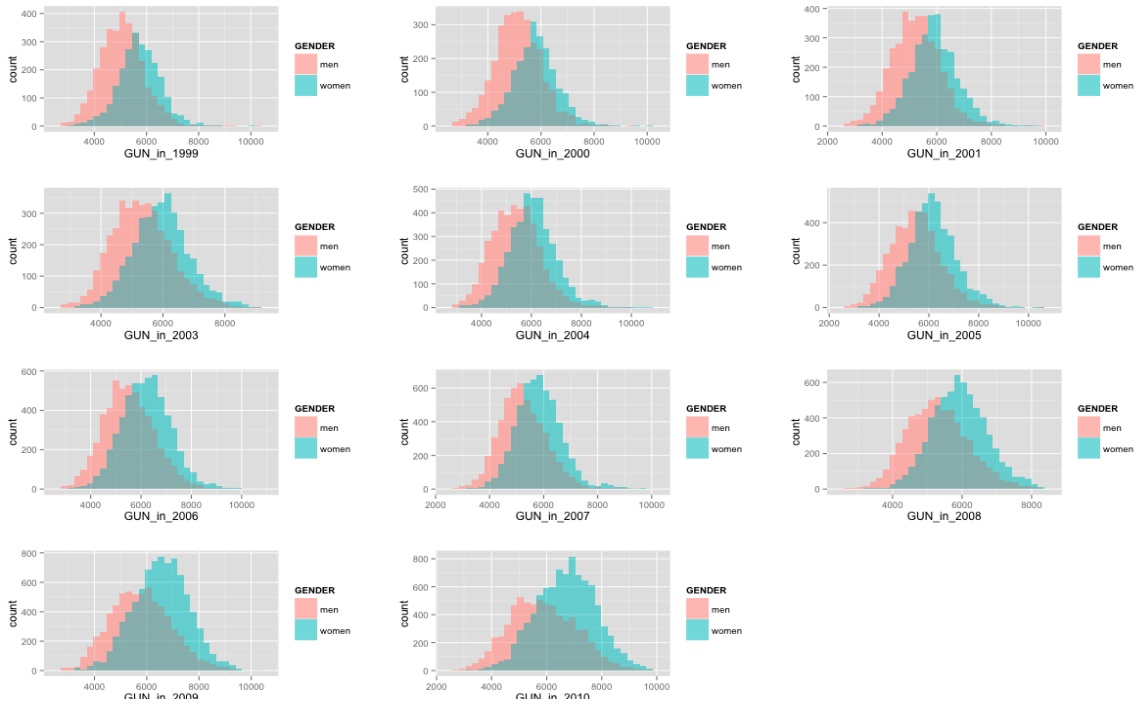


We see that the differences between men and female average time actually seems to increase after 2007 which means that women times were slower and slower. I was curious to see the count of men and women in the races over time because it seems maybe more women decided to race as the years went on. Below is the plot.



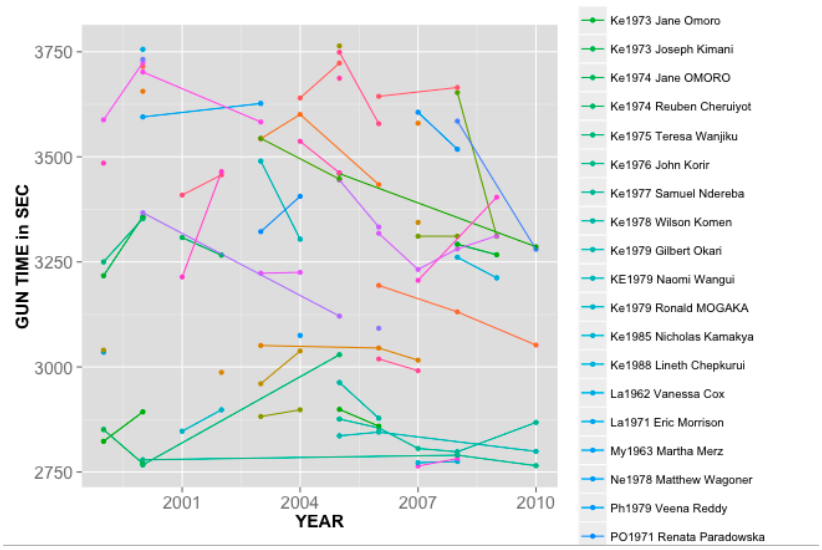
We do see that the number of women surpass men as time goes on which is perhaps why the differences in average times also seems to increase (as there seems to be a relation between slower times and an increase in the number of people).

I wanted to look at the distribution of times for men and females across years. Below are the distributions and they are on intervals measures in seconds.

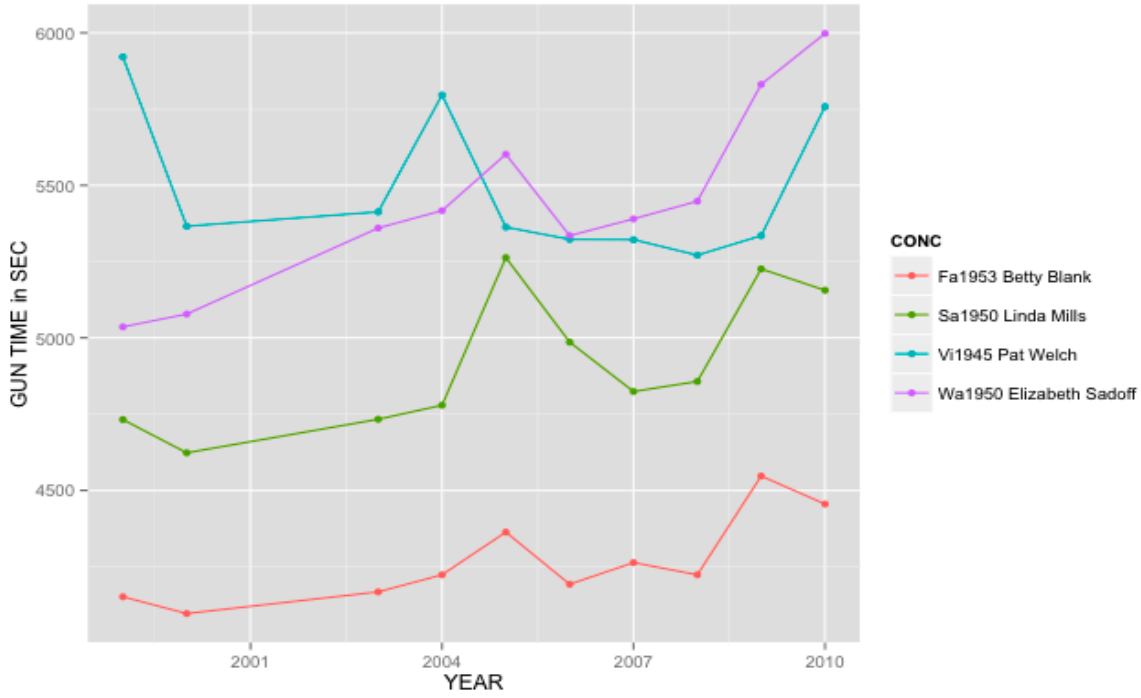


Interestingly enough we see that the combined distribution of both seems to center somewhere around 6000 seconds but the distribution of women seems to slowly split with that of men.

At this point, I wanted to examine how the same people did over time. There were many people who raced over and over again so mapping them all would be hard. I decided to look at those who placed in the top 20 and see how they did over time, if they were to have raced again. (Note for the points with no lines, this means that the person did race again but they were unable to capture a top 20 position).

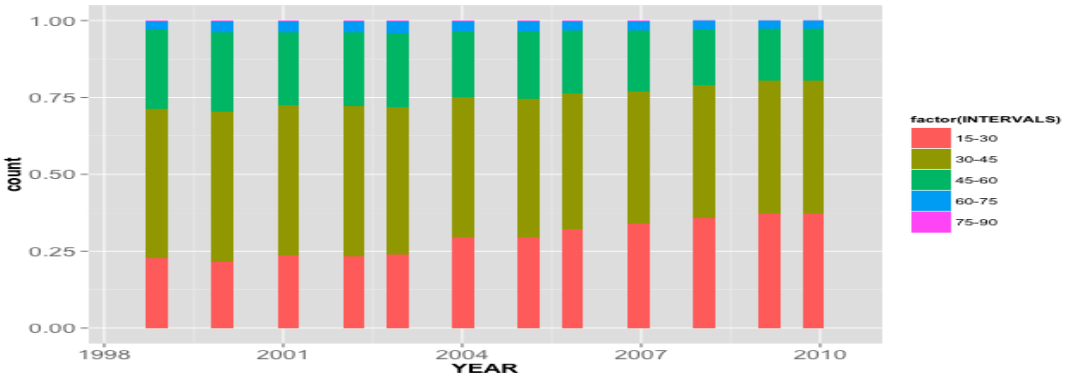


We see that generally people's times seemed to decrease meaning that they improved. Most of the lines have negative slopes but not all. I also wanted to examine how those who participated in at least 9 years of races. Below are the times over the years for these people.

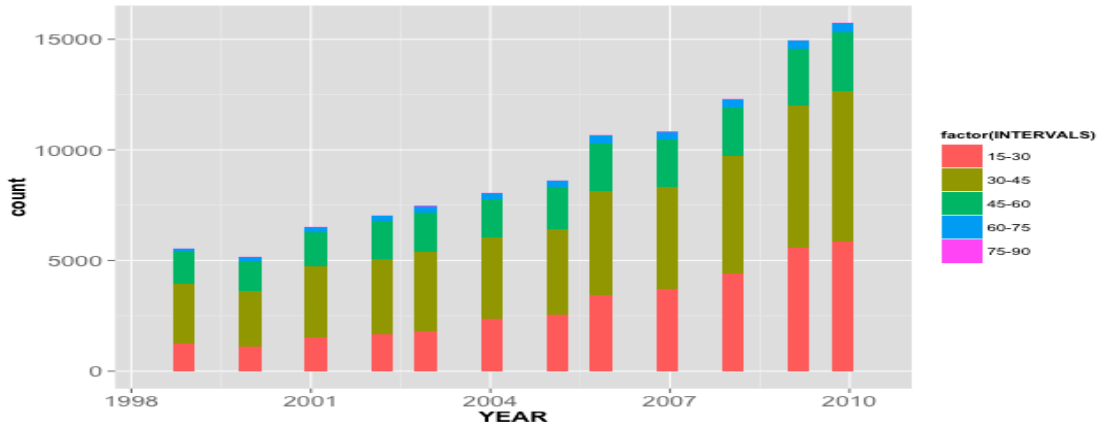


We see that over time, these four individuals had their times slightly increase and interestingly enough each individual did worse at some point and then improved once again. However they all seemed to get a bit slower towards the final year.

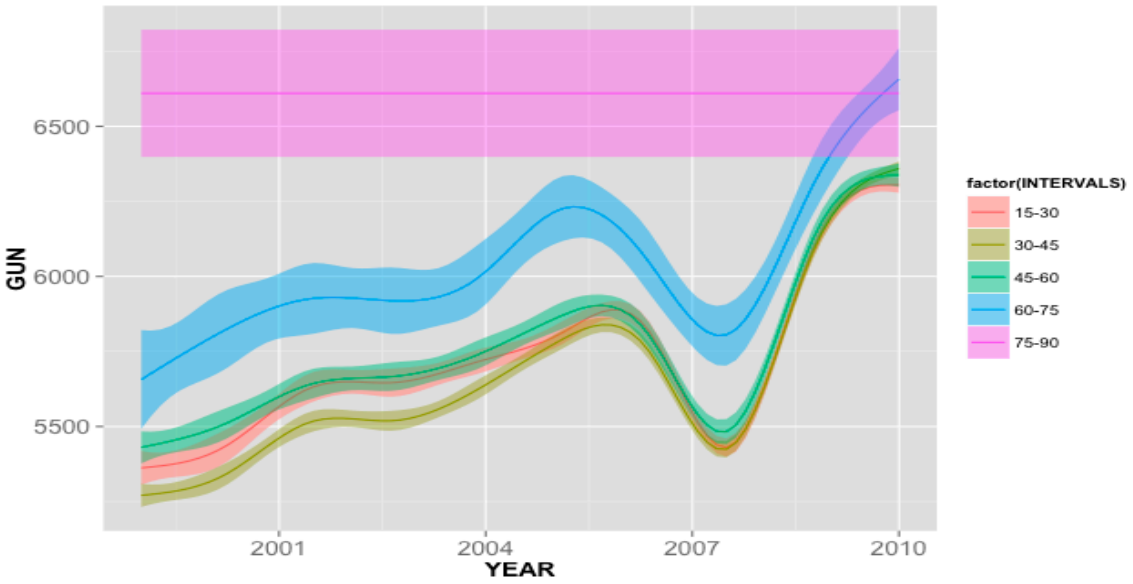
Finally I was curious in examining the proportion of those racing in different age groups. I split the age groups into five categories and mapped the proportions as well as the counts over time.



We see that the greatest proportion of racers are those between 30-45 years old. We also see that the number of races between 15-30 increases while those from 45-60 decreases.



We also see that the counts of each age group increase but they seem to increase at a constant rate.



Finally we examine how each age group did over time. The shaded region just represents the standard error of the smoothed line. We see that anyone over 75 consistently did the worse. However those between 30-45 seemed to do a bit better than those between 15-30 in the initial years. However it seems that towards the end of the years the younger generation seems to do better in terms of time.

Appendix 1:

```
setwd("~/Desktop/School/STA242/stat242_2015/Assignment1/data")
setwd("~/Desktop/data")

#below just for testing

filename <- "men10Mile_2009"
head(combine(filename))
tail(combine(filename))

#below is function to read in the data called COMBINE

combine <- function(filename) {

test <- readLines(filename, n=-1)

#account for weird space by replacing it with regular space.

test <- gsub(intToUtf8(160), " ", test)

#find where line of dashes start and determine position
dash = grep("= ", test)
pos <- unlist(gregexpr("= |=$",test[dash]))

#if there are no = set to zero

if (length(pos) == 0) {
  pos <- c(0,0)
}

#account for funky widths of a couple of data sets

pos <- pos + 1

#account for specific file here

if (filename == "men10Mile_2006" |filename == "women10Mile_2006") {
  pos <- c(pos[1:5], 64, pos[6:9])
  }

#create widths by subtracting width from previous in vector

posvec <- vector(mode="numeric", length=length(pos))
posvec[1] <- pos[1]

testlen <- length(pos)-1
for (i in 1:testlen)
{
  posvec[i+1] <- pos[i+1]-pos[i]
}

#determine how much should be skipped when reading in data
pattern <- "PLACE|Place"
skip = grep(pattern, test)-1

#case for women10mile2001 to replace with men10mile2001 for widths

if (filename == "women10Mile_2001") {

  pattern <- "Elana"
  skip = grep(pattern, test)-1

  filename1 <- "men10Mile_2001"
  test <- readLines(filename1, n=-1)
  test <- gsub(intToUtf8(160), " ", test)
  dash = grep("= ", test)
  pos <- unlist(gregexpr("= |=$",test[dash]))
```

```
  pos <- pos + 1
  posvec <- vector(mode="numeric", length=length(pos))
  posvec[1] <- pos[1]
  testlen <- length(pos)-1
  for (i in 1:testlen)
  {
    posvec[i+1] <- pos[i+1]-pos[i]
  }
}

#read in data

test1 <- read.fwf(filename, skip= skip, widths=posvec, comment.char='', stringsAsFactors=FALSE, strip.white = TRUE)

#create if else statement for column names since 2001 for women has no title
if (filename=="women10Mile_2001"){
  colnames(test1) <- c("PLACE", "NUM", "NAME", "AG", "HOMETOWN", "NET", "GUN")
} else{
names <- unlist(test1[1,])
colnames(test1) <- names
test1 <- test1[-(1:2),]
rownames(test1) = 1:nrow(test1)}

#specific case for men 2009

colnames(test1) <- gsub("[[:space:]]", "", colnames(test1))


#removing unnecessary last two lines as well as star and hash
test1 <- apply(test1, 2, function(x) {gsub("[#*]", "", x)})
test1 <- as.data.frame(test1, stringsAsFactors=FALSE)
rm <- grep("Und|Un", test1[,1])

if (!(length(rm)==0))
{test1 <- test1[-rm,]}

#setting blanks to NA
tester <- test1
tester[tester==""] <- NA


#removing rows with all NAs
foo <- apply(tester, 1, function(x) all(is.na(x)))
tester <- tester[!foo,]

#function to change names to lower case

testernames <- toupper(colnames(tester))
colnames(tester) <- testernames

#standardize all the names

place <- grep("GUN", colnames(tester))
colnames(tester)[place] <- "GUN"

place <- grep("NET", colnames(tester))
colnames(tester)[place] <- "NET"

place <- grep("TIME|TIM", colnames(tester))
colnames(tester)[place] <- "GUN"

#couple of modifications to specific files with random dates at the bottom

if (filename=="women10Mile_2008"){
  tester <- tester[-6398,]
}

if (filename=="men10Mile_2008"){
  tester <- tester[-5906,]
```

```
}

return (tester)
}

### end of initial function to combine

files <- list.files()

#combine all data frames into a list

foo <- sapply(files, combine)

sapply(foo, head)
sapply(foo, tail)

#to see if names are in all of the columns
x = sapply(foo, names)
y = sapply(x, function(z) c("PLACE", "NAME", "AG", "HOMETOWN", "GUN") %in% z )


getnames <- function(i){
  specnames <- c("PLACE", "NAME", "AG", "HOMETOWN", "GUN")
 (i[ ,specnames])
}

#I decided to only keep place name age hometown and gun time
foo1 <- lapply(foo, getnames)

lapply(foo1, head)
lapply(foo1, tail)


#combine the dataframe to get large dataframe
combinedframe <- do.call("rbind", foo1)

testing <- combinedframe

rowname <- (rownames(testing))

#add a column of year and gender

GENDER <- gsub("men.*", "men", rowname)
year1 <- gsub(".*_", "", rowname)
YEAR <- gsub("\\..*", "", year1)

combinedframe$GENDER<- GENDER
combinedframe$YEAR <- YEAR
```

Appendix 2:

```
#replicate our data frame twice for time purposes
testing <- combinedframe

#now we begin the analysis

library(lubridate)

#first we fix the time

time <- testing$GUN

#here we replace the gun time with hours minutes and seconds
mstime <- ms(time)
timeLOG <- is.na(mstime)
mstime[timeLOG] <- hms(time[timeLOG])

combinedframe$GUN <- mstime
```

```
# now we have our hours and minutes and seconds

# I wanted to see how each first place did over time seperated my males and females

first <- subset(combinedframe, combinedframe$PLACE=="1")

library(ggplot2)

first$GENDER <- as.factor(first$GENDER)
first$GUN <- period_to_seconds(first$GUN)
first$GUN <- as.numeric(first$GUN)

ggplot(data=first, aes(x=NAME, y=GUN,  group = GENDER, colour = GENDER)) +
  geom_line() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))+
  geom_point(size=2, shape=21) +
  ylab("GUN TIME in Seconds") +
  theme(axis.text=element_text(size=14),
     axis.title=element_text(size=14,face="bold"))

#now to get the average time by year for males and females

seconds <- combinedframe
seconds$GUN <- period_to_seconds(seconds$GUN)


seconds <- seconds[!is.na(seconds$GUN),]
test <- split(seconds[,c(5,7)], seconds$GENDER)
men <- test[[1]]
women <- test[[2]]
men <- tapply(men$GUN, men$YEAR, mean)
women <- tapply(women$GUN, women$YEAR, mean)
men <-as.vector(men)
women<-as.vector(women)

require(reshape2)
menwomen <- melt(data.frame(men,women))
colnames(menwomen) <- c("GENDER", "TIME")

YEARS <- rep(1999:2010,2)
menwomen$YEARS <- YEARS

ggplot(data=menwomen, aes(x=YEARS, y=TIME,  group = GENDER, colour = GENDER)) +
  geom_line() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1), axis.text=element_text(size=14),
       axis.title=element_text(size=14,face="bold"))+
  geom_point(size=2, shape=21) +
  ylab("AVG Time in Sec")

#shows us that the difference in time between men and women each year is almost the same on average and that men have
better average times then women
#actually showing the difference below

diff1 <- women-men
diff <- as.vector(diff1)
year <- rep(1999:2010,1)
frame = data.frame(Diff=diff, Year=year)

  ggplot(frame, aes(x=Year, y=Diff)) + geom_line() + geom_point() + ylab("Diff in AVG Time in Sec")
  + theme(axis.text=element_text(size=14), axis.title=element_text(size=14,face="bold"))

#we see that there are more distinct differences than the original chart leads us to believe
#but maybe the reason average time for both drops and increases at the same rate for a reason..but why? Perhaps the number
of people effect average times of the race?
#Lets find out


people <- combinedframe
```

```
people <- people[!is.na(people$GUN),]

men1 <- test[[1]]
women1 <- test[[2]]

men2 <- tapply(men1$GUN, men1$YEAR, length)
women2 <- tapply(women1$GUN, women1$YEAR, length)

men2 <-as.vector(men2)
women2<-as.vector(women2)

require(reshape2)
menwomen1 <- melt(data.frame(men2,women2))
colnames(menwomen1) <- c("GENDER", "COUNT")

YEARS <- rep(1999:2010,2)
menwomen1$YEARS <- YEARS

 ggplot(data=menwomen1, aes(x=YEARS, y=COUNT,  group = GENDER, colour = GENDER)) +
  geom_line() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1), axis.text=element_text(size=14),
axis.title=element_text(size=14,face="bold"))+
  geom_point(size=2, shape=21) +
  ylab("COUNT")


library(gridExtra)
grid.arrange(x,y)


# i want to see the distribution of times each year for men and women
yeardata <- combinedframe
yeardata$GENDER <- as.factor(yeardata$GENDER)
yeardata$GUN <- as.numeric(period_to_seconds(yeardata$GUN))
yeardata <- split(yeardata, YEAR)
year1 <- yeardata[[1]]
year1$GUN_in_1999 <- year1$GUN
year2 <- yeardata[[2]]
year2$GUN_in_2000 <- year2$GUN
year3 <- yeardata[[3]]
year3$GUN_in_2001 <- year3$GUN
year4 <- yeardata[[4]]
year4$GUN_in_2002 <- year4$GUN
year5 <- yeardata[[5]]
year5$GUN_in_2003 <- year5$GUN
year6 <- yeardata[[6]]
year6$GUN_in_2004 <- year6$GUN
year7 <- yeardata[[7]]
year7$GUN_in_2005 <- year7$GUN
year8 <- yeardata[[8]]
year8$GUN_in_2006 <- year8$GUN
year9 <- yeardata[[9]]
year9$GUN_in_2007 <- year9$GUN
year10 <- yeardata[[10]]
year10$GUN_in_2008 <- year10$GUN
year11 <- yeardata[[11]]
year11$GUN_in_2009 <- year11$GUN
year12 <- yeardata[[12]]
year12$GUN_in_2010 <- year12$GUN


a <- ggplot(year1, aes(x=GUN_in_1999, fill=GENDER)) +
  geom_histogram(position="identity", alpha=0.4)
b <-ggplot(year2, aes(x=GUN_in_2000, fill=GENDER)) +
  geom_histogram(position="identity", alpha=0.4)
c <-ggplot(year3, aes(x=GUN_in_2001, fill=GENDER)) +
  geom_histogram(position="identity", alpha=0.4)
d <-ggplot(year4, aes(x=GUN_in_2002, fill=GENDER)) +
  geom_histogram(position="identity", alpha=0.4)
```

```
e <-ggplot(year5, aes(x=GUN_in_2003, fill=GENDER)) +
  geom_histogram(position="identity", alpha=0.4)
f  <-ggplot(year6, aes(x=GUN_in_2004, fill=GENDER)) +
  geom_histogram(position="identity", alpha=0.4)
g  <-ggplot(year7, aes(x=GUN_in_2005, fill=GENDER)) +
  geom_histogram(position="identity", alpha=0.4)
h <-ggplot(year8, aes(x=GUN_in_2006, fill=GENDER)) +
  geom_histogram(position="identity", alpha=0.4)
i  <-ggplot(year9, aes(x=GUN_in_2007, fill=GENDER)) +
  geom_histogram(position="identity", alpha=0.4)
j  <-ggplot(year10, aes(x=GUN_in_2008, fill=GENDER)) +
  geom_histogram(position="identity", alpha=0.4)
k  <-ggplot(year11, aes(x=GUN_in_2009, fill=GENDER)) +
  geom_histogram(position="identity", alpha=0.4)
l  <-ggplot(year12, aes(x=GUN_in_2010, fill=GENDER)) +
  geom_histogram(position="identity", alpha=0.4)

grid.arrange(a,b,c, e, f, g, h, i ,j,k,l)

####create a unique column that represents each person then we find duplicates and only return those that have duplicates
over the years

head(combinedframe)
rep <- combinedframe
rep$AG <- as.numeric(rep$AG)
rep$YEAR <- as.numeric(rep$YEAR)
rep$CONC <- rep$YEAR-rep$AG
rep$CONC <- as.character(rep$CONC)
rep$CONC <- paste(substr(rep$HOMETOWN,1,2),rep$CONC, sep="")
rep$CONC <- paste(rep$CONC, rep$NAME)
test <- duplicated(rep$CONC)
testing <- rep[test,]


rep1 <-rep[rep$CONC %in% testing$CONC, ]
subset(rep1, rep1$CONC == "Ke1973 Joseph Kimani")

#we have sixteen thousand unique individuals who seemed to have ran the race multiple times. So lets only select those who
placed in the top

vec <- c(1:20)
rep1$PLACE <- as.numeric(rep1$PLACE)
rep2 <- rep1[rep1$PLACE %in% vec,]
x <- unique(rep2$CONC)

rep2$GUN <- period_to_seconds(rep2$GUN)

ggplot(rep2, aes(x=YEAR, y=GUN, colour=CONC)) + geom_point() + geom_line()+
  theme(axis.text=element_text(size=14),
        axis.title=element_text(size=14,face="bold")) + ylab("GUN TIME in SEC")


#the ones with only one point showed that they did not place in the top ten during the next time arond

#now i want to look at those who competed in 9 or more years

x <- table(rep1$CONC)
x <- subset(x, x>8)
namesx <- names(x)
rep3 <- rep1[rep1$CONC %in% namesx,]
rep3$GUN <- period_to_seconds(rep3$GUN)
rep3$GUN <- as.numeric(rep3$GUN)

ggplot(rep3, aes(x=YEAR, y=GUN, colour=CONC)) + geom_point() + geom_line()

# in ten or more years

x <- table(rep1$CONC)
x <- subset(x, x>9)
```

```
namesx <- names(x)
rep3 <- rep1[rep1$CONC %in% namesx,]
rep3$GUN <- period_to_seconds(rep3$GUN)
rep3$GUN <- as.numeric(rep3$GUN)

ggplot(rep3, aes(x=YEAR, y=GUN, colour=CONC)) + geom_point() + geom_line() + ylab("GUN TIME in SEC")

###now I want to look at proportion of those with different ages in each year

vecage <- c(15:100)
ages <- combinedframe
ages$AG <- as.numeric(ages$AG)
ages <- ages[!is.na(ages$AG),]
ages <- ages[ages$AG %in% vecage,]
intervals <- cut(ages$AG, 5, labels=c("15-30", "30-45", "45-60", "60-75", "75-90"))
ages$INTERVALS <- intervals
ages$GUN <- period_to_seconds(ages$GUN)
ages$GUN <- as.numeric(ages$GUN)

#these two functions below i got with the help of CARLOS from class he is in my group

ggplot(ages, aes(x=YEAR,  fill=factor(INTERVALS))) +
  (geom_bar(position = "fill" )) +  theme(axis.text=element_text(size=14),
                                axis.title=element_text(size=14,face="bold"))

ggplot(ages, aes(x=YEAR,  fill=factor(INTERVALS))) +
  geom_bar() +   theme(axis.text=element_text(size=14),
                 axis.title=element_text(size=14,face="bold"))

###Looking at ages over time in terms of perofrmance, also help from CARLOS in my group

ages <- ages[!is.na(ages$GUN),]

m + stat_smooth(aes(x = YEAR, y = GUN,
                colour = factor(INTERVALS),
                fill = factor(INTERVALS))) +   theme(axis.text=element_text(size=14),
                                        axis.title=element_text(size=14,face="bold"))
```