Cherry Blossom 10K Run

Ananth S, Vitaly B, Jackson Au & Rick F

September 22, 2019

**Abstract**

This case study covers the analyses of male and female runners in Cherry Blossom 10K run.

# Introduction

The Credit Union Cherry Blossom 10K run is an annual event, and historically has been used as a preparation for the Boston Marathon. Participants include amateur and professional runners, and those who perform well would continue to do well in the world stage.

The data provided for this study include the runner’s name, country of birth, age, run completion time, and position. The data was collected scraping the Cherry Blossom website (<http://www.cherryblossom.org/>) for both males and females for a total of 14 years from 1999 through 2012.

The requirements for this case study were to normalize the performance of the runners across different ages (20 years to 80 years). The data was to be fitted using various methodologies (linear and non-parametric) and plotted (quantile and density) to showcase observed and reported performance. The performance comparison was completed separately for male and female runners.

Given that the data was scraped from the web, the group exercised caution when reviewing the output monitoring for malformed data or special characters that may impact our analysis.

The implementation for this study relied on the R programming platform.

# Methods

Using R’s XML library, the group used htmlParse and xmlValue to read the url for each year of data and navigate the Domain Object Model attributes in HTML. A utility function was developed to traverse multiple urls while writeLines stored each output as a text file. Subsequently, these text files were read into variables using readLines.

To ensure that data aligned to the appropriate header (e.g., Age, Time, Name of runner, Country of birth), additional helper functions were created. As the data was read from files in the form of character vectors, a variety of conversions was performed to ready the data for analyses (e.g., Age converted to numeric, time converted to minutes, etc.). Apply functions were used for data capturing and conversion. As different times were reported in the data (e.g., Gun time, Net time, and time), all times were organized under one common name “Time” for consistency. This was achieved using split and rounding functions.

A final data frame was created to combine results from 1999 and 2012 for male and female runners. This data frame was used for fitting the model and studying the results. Box plots were used to study the age range across different years. Brewer color package was applied for scatter plots with some noise added to the age with jitter function. This provides an optimal plot to showcase Age vs Run time distribution.

Loess and LM functions are used to fit run and age data and predict the performance for a range of ages. The resultant plot helps to analyze the difference between the Loess and LM functions. The Loess function does a good job in predicting the change in performance with age. Density and quantile plots are then used to compare the distribution of age vs normalized time across runners for different ages for the years 1999 and 2012 and across male and female. These are discussed in detail in the results section.

# Results

Normalizing is done based on the fastest male and female runners in years 1999 and 2012. Below are charts to display the top male and female performers in each respective year.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Year** | **Sex** | **Name** | **Home** | **Age** | **RunTime** |
| 1999 | M | Worku Bikila | Ethiopia | 28 | 46.98 |
| 1999 | M | Lazarus Nyakeraka | Kenya | 24 | 47.02 |
| 1999 | M | James Kariuki | Kenya | 27 | 47.05 |

*Figure 1.1: Top 3 Fastest Male Runners (1999)*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Year** | **Sex** | **Name** | **Home** | **Age** | **RunTime** |
| 2012 | M | Allan Kiprono | Kenya | 22 | 45.25 |
| 2012 | M | Lani Kiplagat | Kenya | 23 | 46.46 |
| 2012 | M | John Korir | Kenya | 36 | 47.55 |

*Figure 1.2: Top 3 Fastest Male Runners (2002)*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **year** | **sex** | **name** | **home** | **age** | **runTime** |
| 1999 | F | Jane Omoro | Kenya | 26 | 53.62 |
| 1999 | F | Jane Ngotho | Kenya | 29 | 53.63 |
| 1999 | F | Alla Zhilyayeva | Russia | 29 | 54.13 |

*Figure 1.3: Top 3 Fastest Female Runners (1999)*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| year | sex | name | home | age | runTime |
| 2012 | F | Jelliah Tinega | Kenya | 26 | 54.03 |
| 2012 | F | Malika Mejdoub | Ethiopia | 29 | 54.40 |
| 2012 | F | Yihunlish Delelecha | Ethiopia | 30 | 54.55 |

*Figure 1.4: Top 3 Fastest Female Runners (2012)*

### ***Normalization***

An example of the normalization process, each runner’s time was normalized in accordance to the fastest runner of each year. For the 1999 Male group, runner time was normalized according to Worku Bikila’s time of 46:98. Another example would be Jelliah Tinega’s 54:03 time for the Female runners in 2012.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **age** | **year** | **sex** | **name** | **home** | **runTime** | **minTime** | **NormRun** | **sm\_time** | **NormRuntoSmooth** |
| 20 | 1999 | M | Gary Rovner | Washington DC | 97.01667 | 57.78333 | 39.233333 | 77.03825 | 1.259331 |
| 20 | 1999 | M | Jaret Seiberg | Silver Spring MD | 72.53333 | 57.78333 | 14.75 | 77.03825 | 0.941524 |
| 20 | 1999 | M | Ron Varnum | Westmont NJ | 61.95 | 57.78333 | 4.166667 | 77.03825 | 0.804146 |

*Figure 2.1: Normalized Male Runner Times (1999)*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **age** | **year** | **sex** | **name** | **home** | **runTime** | **minTime** | **NormRun** | **sm\_time** | **NormRuntoSmooth** |
| 66 | 2012 | F | Nancy Malan | Washington DC | 114.9833 | 107.4667 | 7.516667 | 109.4419 | 1.050634 |
| 66 | 2012 | F | Rosemary Dawson | Lynchburg VA | 107.4667 | 107.4667 | 0 | 109.4419 | 0.981952 |
| 66 | 2012 | F | Courtenay Mullen | Fairfax VA | 111.0667 | 107.4667 | 3.6 | 109.4419 | 1.014846 |

*Figure 2.2: Normalized Female Runner Times (2012)*

|  |  |
| --- | --- |
| **NormRun** | Normalized run time |
| **SM\_Time** | Predicted smooth time |
| **NormRunToSmooth** | LOESS method applied to normalized run time |

*Figure 2.3: Data Description Table*

#### **Interpretation** - **Males (1999 vs 2012)**

A close up of a map

Description automatically generated

*Figure 3.1: QQ Plot – Male Runners*

A screenshot of a cell phone

Description automatically generated  
*Figure 3.2: Density Plot – Male Runners*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Min** | **1Q** | **Median** | **Mean** | **3Q** | **Max** |
| **Male-1999** | 0.5756 | 0.8907 | 0.9974 | 1.007 | 1.1011 | 1.8168 |
| **Male-2012** | 0.5332 | 0.8785 | 0.9885 | 0.9993 | 1.1058 | 1.6766 |

*Figure 3.3: Summary Statistics – Male Runners*

Looking at the QQ-plot, there appears to be a linear relationship between age and time, where the older the runner, the slower their time. The inverse is also true: if a runner is younger, they are expected to produce a faster time. This theory seems to hold true as there are very little outlier data points.

The density curves, representing age, are quite consistent across both years. Middle-aged runners seem to be the make of the race population, while we see very little younger boys and very little senior citizens. There is a slight bump (mini-mound) in the density plot for 1999, this seems to represent a large concentration of runners that are above the average age. In 2012, most runners belong in the average age group. Comparing 1999 to 2012, 1999 seems to contain more runners, while 2012 has a younger group of runners.

In reviewing the summary statistics, majority of the runners’ times (using median) in 1999 seemed to be closer to the average, suggesting that very few runners were separating themselves in comparison to the competition. In contrast, the competition in 2012 seems to be faster than the average, as the median is deviating from the normalized times.

**Interpretation** - **Females (1999 vs 2012)**

A close up of a map

Description automatically generated

*Figure 3.4: QQ plot – Female Runners*

A close up of a mans face

Description automatically generated

*Figure 3.5: Density Chart – Female Runners*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Min** | **1Q** | **Median** | **Mean** | **3Q** | **Max** |
| **Female-1999** | 0.675 | 0.9146 | 0.9971 | 0.9998 | 1.0876 | 1.743 |
| **Female-2012** | 0.5524 | 0.9006 | 0.9907 | 0.9995 | 1.0881 | 1.6999 |

*Figure 3.6: Summary Statistics – Female Runners*

Similar to the men, there appears to be a linear relationship between age and time among female participants, where the older the runner, the slower their time. The inverse is also true: if a runner is younger, they are expected to produce a faster time. This theory seems to hold true as there are very little outlier data points.

The density curves, representing age, are quite consistent across both years. Middle-aged runners seem to be the make of the race population, while we see very little younger girls and very little senior citizens. The density of age groups do show very little variance across 1999 and 2012, meaning running is a consistent passion for females.

In reviewing the summary statistics, runner times are very consistent from 1999 to 2012 (median: 0.9971 vs 0.9907, mean: 0.9998 vs 0.9995). Female runners are very consistent relative to their competition!

### ***Additional Insight - Between Male and Female***

An interesting thing to note is the bell curves of the male and female density plots. From a volume-perspective, there are noticeably more female runners than male runners in the average age group. Male runners appear to be more staggered across age groups. Whereas female runners are heavily concentrated around the average age group.

# Conclusion

The fastest male runner in 2012 was 1.5 minutes faster than the fastest male runner in 1999. Using normalization and statistical methods, it is confirmed that male running times are getting faster as the years go by. From an age-perspective, the sport of running is appealing to younger male audiences. Running is mostly enjoyed by middle-aged men, but the Cherry Blossom competition attracts men staggered across all age groups, including old and young.

The fastest female runner in 1999 was faster than the 2012 fastest female by half a minute. This however did not represent the drop off in running ability for female runners. Run times were tight and consistent across the two years for the majority of female runners, the difference was only in the top performers. From an age-perspective, female runners are heavily concentrated in the middle-age group. Unlike the men, females do not have many runners in the younger and older age groups, but are mostly comprised in the average age group. This observation is consistent across both 1999 and 2012.

# References

1. Nolan, D. and Lang, D. T. “Data Science in R.” CRC Press, 2015 (Chapter 1).

# Appendix – Code

# library(XML)

# library(data.table)

# library(ggplot2)

# menURLs =

# c("results/1999/cb99m.html","results/1999/cb99f.html","results/2012/2012cucb10m-m.htm","results/2012/2012cucb10m-f.htm"

# )

# ubase = "http://www.cherryblossom.org/"

# urls = paste(ubase, menURLs, sep = "")

# urls[1:4]

# years = c("1999\_M.txt","1999\_F.txt","2012\_M.txt","2012\_F.txt")

extractResTable =

#

# Retrieve data from web site,

# find the preformatted text,

# and write lines or return as a character vector.

#

function(url = "/09cucb-F.htm",

year = 1999, file = NULL) #sex = "male",

{

doc = htmlParse(url)

if (year == "1999\_F.txt" || year == "1999\_M.txt" ) {#& sex == "mmale"

# Get preformatted text from <pre> elements

pres = getNodeSet(doc, "//pre")

txt = xmlValue(pres[[1]])

els = strsplit(txt, "\n")[[1]]

}

else { #if (year == 2012)

# Get preformatted text from <pre> elements

pres = getNodeSet(doc, "//pre")

txt = xmlValue(pres[[1]])

els = strsplit(txt, "\r\n")[[1]]

}

if (is.null(file)) return(els)

# Write the lines as a text file.

file=year

writeLines(els, con = file)

}

Tables = mapply(extractResTable, url = urls, year = years)

names(Tables) = years

sapply(Tables, length)

findColLocs = function(spacerRow) {

spaceLocs = gregexpr(" ", spacerRow)[[1]]

rowLength = nchar(spacerRow)

if (substring(spacerRow, rowLength, rowLength) != " ")

return( c(0, spaceLocs, rowLength + 1))

else return(c(0, spaceLocs))

}

selectCols =

function(colNames, headerRow, searchLocs)

{

sapply(colNames,

function(name, headerRow, searchLocs)

{

startPos = regexpr(name, headerRow)[[1]]

if (startPos == -1)

return( c(NA, NA) )

index = sum(startPos >= searchLocs)

c(searchLocs[index] + 1, searchLocs[index + 1] - 1)

},

headerRow = headerRow, searchLocs = searchLocs )

}

# class(Values)

# colnames(Values) = shortColNames

# head(Values)

# tail(Values)[ , 1:3]

extractVariables =

function(file, varNames =c("name", "home", "ag", "gun",

"net", "time"))

{

# Find the index of the row with =s

eqIndex = grep("^===", file)

# Extract the two key rows and the data

spacerRow = file[eqIndex]

headerRow = tolower(file[ eqIndex - 1 ])

body = file[ -(1 : eqIndex) ]

# Obtain the starting and ending positions of variables

searchLocs = findColLocs(spacerRow)

locCols = selectCols(varNames, headerRow, searchLocs)

Values = mapply(substr, list(body), start = locCols[1, ],

stop = locCols[2, ])

show(Values)

colnames(Values) = varNames

invisible(Values)

}

allfilenames = paste(years[1:4], sep = "")

allFiles = lapply(allfilenames, readLines)

names(allFiles) = years[1:4]

ResMat = lapply(allFiles, extractVariables)

length(ResMat)

sapply(ResMat, nrow)

# age = as.numeric(ResMat[['2012']][ , 'ag'])

# tail(age)

age = sapply(ResMat,

function(x)

{

as.numeric(x[ , 'ag'])

}

)

boxplot(age, ylab = "Age", xlab = "Years")

test=ResMat

ag=lapply(age, function(x)

{

d=which(is.na(x))

}

)

Ag

test[['1999\_M.txt']]=test[['1999\_M.txt']][-c(ag$`1999\_M.txt`),]

test[['1999\_F.txt']]=test[['1999\_F.txt']][-c(ag$`1999\_F.txt`),]

test[['2012\_M.txt']]=test[['2012\_M.txt']][-c(ag$`2012\_M.txt`),]

nrow(test[['1999\_M.txt']])

nrow(test[['1999\_F.txt']])

nrow(test[['2012\_M.txt']])

nrow(test[['2012\_F.txt']])

ResMat=test

charTime = ResMat[['1999\_M.txt']][, 'time']

head(charTime, 5)

convertTime = function(time) {

timePieces = strsplit(time, ":")

timePieces = sapply(timePieces, as.numeric)

sapply(timePieces, function(x) {

if (length(x) == 2) x[1] + x[2]/60

else 60\*x[1] + x[2] + x[3]/60

})

}

createDF =

function(Res,name) #function(Res, year, sex)

{

# Determine which time to use

useTime = if( !is.na(Res[1, 'net']) )

Res[ , 'net']

else if( !is.na(Res[1, 'gun']) )

Res[ , 'gun']

else

Res[ , 'time']

#

splitter = strsplit(name, "\_")[[1]]

splitter2 = strsplit(splitter, "\\.")[[2]]

runTime = convertTime(useTime)

Results = data.frame(year = rep(splitter[1], nrow(Res)),

sex = rep(splitter2[1], nrow(Res)),

name = Res[ , 'name'],

home = Res[ , 'home'],

age = as.numeric(Res[, 'ag']),

runTime = runTime,

stringsAsFactors = FALSE)

invisible(Results)

}

# menDF = mapply(createDF, ResMat, year = years,

# sex = rep("M", 14), SIMPLIFY = FALSE)

allDF = mapply(createDF, ResMat,names(ResMat), SIMPLIFY = FALSE)

#menDF$'1999m.txt'

# class(menDF$'1999m.txt')

sapply(allDF, function(x) sum(is.na(x$runTime)))

sapply(allDF, function(x) sum(is.na(x$name)))

allDF

tapp=

function (yy){

# show(class(y))

yyy = yy[ which(yy$age>=20 & yy$age <= 80), ]

tx=(tapply(yyy$runTime,list(yyy$age), min))

#data.frame(tx)

df = data.frame(age=names(tx),minTime=tx)

sss=merge(x= yyy, y= df, by= 'age', all.x= FALSE)

#

sss$NormRun = (sss$runTime-sss$minTime)

loessMod10 = loess(runTime ~ age, data=sss) # 10% smoothing span, span=0.10

age20to80 = 20:80

smoothed10 = predict(loessMod10,age20to80)

smoothed10mergeage=merge(x= smoothed10, y= age20to80, by= 0, all.x= FALSE)

dff=data.frame(smoothed10mergeage)

setnames(dff, 2, "sm\_time")

setnames(dff, 3, "age")

sss1=merge(x= sss, y= dff, by= 'age', all.x= F)

sss1$NormRuntoSmooth = (sss1$runTime/sss1$sm\_time)

plot(smoothed10 ~ age20to80,

type = "l", col = "purple", lwd = 3,

xlab = "Age (years)", ylab = "Run Time Prediction")

return (sss1)

}

#allDF1 = mapply(createDF, ResMat,names(ResMat), SIMPLIFY = FALSE)

allDF1 = mapply(tapp, allDF, SIMPLIFY = FALSE)

#allDF1

allgoups = do.call(rbind, allDF1)

allgoups

nrow(allgoups)

summary(allgoups)

write.csv(allgoups, file = "MyData.csv")

age1999m = allgoups[ (allgoups$sex == 'M') & (allgoups$year == 1999), "NormRuntoSmooth"]

age2012m = allgoups[ (allgoups$sex == 'M') & (allgoups$year == 2012), "NormRuntoSmooth"]

age1999f = allgoups[ (allgoups$sex == 'F') & (allgoups$year == 1999), "NormRuntoSmooth"]

age2012f = allgoups[ (allgoups$sex == 'F') & (allgoups$year == 2012), "NormRuntoSmooth"]

summary(age1999m)

summary(age2012m)

summary(age1999f)

summary(age2012f)

qq=

function (arg1,arg2,name){

titlee= paste("Quantile-quantile plot of", name , "runner's age", sep=" ")

qqplot(arg1, arg2, pch = 19, cex = 0.5,

ylim = c(0,2), xlim = c(0,2),

xlab = "NormRunTime in 1999 Race",

ylab = "NormRunTime in 2012 Race",

main = paste("Quantile-Quantile plot of", name , "runner's age", sep=" "))

abline(a =0, b = 1, col="red", lwd = 2)

}

qq(age1999m,age2012m,'Male')

qq(age1999f,age2012f,'Female')

age1999m = allgoups[ (allgoups$sex == 'M') & (allgoups$year == 1999), "NormRuntoSmooth"]

age2012m = allgoups[ (allgoups$sex == 'M') & (allgoups$year == 2012), "NormRuntoSmooth"]

age1999f = allgoups[ (allgoups$sex == 'F') & (allgoups$year == 1999), "NormRuntoSmooth"]

age2012f = allgoups[ (allgoups$sex == 'F') & (allgoups$year == 2012), "NormRuntoSmooth"]

qqdens=

function (arg1,arg2,name){

plot(density(arg1, na.rm = TRUE),

ylim = c(0, 3.5), col = "purple",

lwd = 3, xlab = "Age (years)",

main = paste("Density plot of", name , "runner's age", sep=" "))

lines(density(arg2, na.rm = TRUE),

lwd = 3, lty = 2, col="green")

legend("topleft", col = c("purple", "green"), lty= 1:2, lwd = 3,

legend = c("1999", "2012"), bty = "n")

}

# Density Plots

qqdens(age1999m,age2012m,'Male')

qqdens(age1999f,age2012f,'Female')