MSDS 7333 Spring 2021: Case Study 02

Analysis of Runners' Performance

Sachin Chavan,Tazeb Abera,Gautam Kapila,Sandesh Ojha

2021 February 01

Introduction

This is an information age and there is lot of data generated and published every day in different forms. If we just think about websites there are hundreds of thousands of websites across the world publish different types of content like research, news, education,blogs etc everyday and most of it is freely available to read and digest. Such information can be potentially be useful in gaining insights for personal and professional interests. e.g. we can learn lot about population like education, labor, gender distribution etc with exploring data from the census data published on world bank or census.gov. This is all available for free to download and view on such websites.

In this case study we are going to explore one of such website www.cherryblossom.org. **Cherry blossom Ten Mile Run** is annual 10-mile road race in Washington, D.C. founded in 1973, almost 48 years ago and its popularity and participation in the event has grown over the years. As per their website in year 2018 around 17000 people participated in the race ranging in age 9 to 89 years. Textbook has covered Male racers in its example, we are going to explore Female racers. We will study how age affect physical performance in female racers and try to get many other insights hidden in the data.

"According to Bureau for labor Statistics, road running is third most common form of sport and exercise activity among Americans." [2]

Business Understanding

The purpose of the case study is to gather meaningful insights from women participant's results that are published on famous Cherry Blossom' website for the years 1999-2012 under 10 mile race category. The primary objective is to download the 1999 to 2012 results of 10 mile running event from the website to explore and shed light on the effect of age on the runner's physical performance over the years along with other meaningful insights if any.

The Cherry Blossom organizes 5K and 10M running event year in Washington,D.C early in April when cherry trees are said to be in bloom. They record results for every participants and publish on their website. The race has been in such demand that the runner are chosen via lottery system to enter the race. Participants have different age range and for both male and female, however, this case study focuses on Woman's result.

Objective

Download the woman's results for the duration of 1999-2012 from Cherry Blossom's website, perform data wrangling and gather meaningful insights.

Data Extraction / Preparation

At the time of writing this document www.cherryblossom.org was revamped and they have made results available in tabular format on http://www.cballtimeresults.org. Based on new web page that they have designed it can list results based on option selected in the dropdowns and all results are paginated with one page shows 20 results. So result for selected year is spanned across multiple pages for selected year of result. There is no way data can be extracting manually. This has to be automated by method called Web scraping. The program will automatically scrape through pages and extract records for our analysis.

The new Cherry blossom results pages are not dynamic content and it makes somewhat easier to pull all required records from the website.

e.g URL for First page of 1999 Woman's results for 10 Mile run:

http://www.cballtimeresults.org/performances?division=Overall+Women&page=1&section=10M&sex=W&utf8=&year=1999

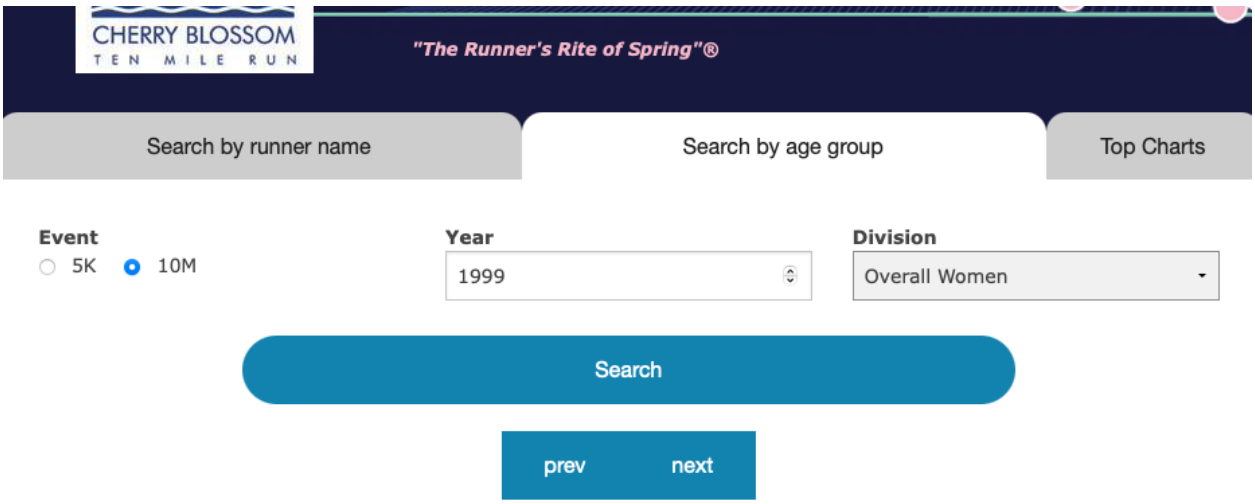

Figure 1: Results Page

URL pattern was observed same for all pages, records for all runners for selected years can be extracted just by changing values of

- Page Number - *[page=]*
- Race Year - *[year=]*

Following attributes remains same for our purpose as case study is limited to analysis of women racers.

- Division - *[division=Overall+Women]*

- Sex - *[sex=W]*
- section - *[section=10M]*

As seen in screenshot results are presented in tabular format. So web scraping is the method needs to be utilized to extract data from these results pages. R programming langauge comes with sophisticated technique to scrape through web pages to extract records from html tables. R package **rvest** provides interfaces which can accept URL and tags from which data to be extracted. Most of HTML parsing is done by rvest package but it does not format data to put in R dataframe. So additional programming is required to format data that is pulled from webpages.

So to pull data from results web pages following things are required:

- Use R rvest package to parse HTML page with required tags. read_html, html_nodes, html_text are the supporting functions that will be used to extract tabular data.
- Regular expressions are also helpful sometimes to find specic pattern in strings. e.g. time column has specific format HH:MM:SS.Where HH, MM and SS are digits. Regular expressions are useful to check if time column contains data that contains other than colon and digits.
- Format records returned by **rvest** functions
- String manipulations are required to format data
- Put data in R Data Frame and do further wrangling, once data was extracted.

Constraints

Web scraping comes with limitations. Programs developed to pull data from the websites cannot be generalized much and it heavily dependent on design of the web pages. Although, no one changes web page design that frequently, it is required for us to check if there are any changes in the web pages, structure of data etc to extract data from the website. Programs are needs to be updated if there is any change in the content on the web page. In general following things to needs to be taken into considerations when using web scraping to extract data

- URL - It may change. For cherry blossom results URL has similar format for all results. so we just have to change few parameters in the URL to retrieve required result. But that may not be the case always. URL may need to checked and program may need to be reconfigured.
- If data is in tabular format like in the case of Cherry Blossom website, they may add more columns or remove existing columns. So one has to keep checking on the structure of the data that they are scraping.
- Dynamic contents are even more challenging. In the case of dynamic content and URL to access is may not be that sophisticated as we found in Cherry Blossom website.
- Websites normally puts restrictions amount of to be extracted in single timeframe so large scale data extraction is harder.
- Robots.txt - This is very important. This file is used to manage traffic to the website. One should look at this file which is generally located at http://www.abc.com/robots.txt to see if website allows web scraping/crawling. This is not the hard rule to follow what is written in the **robots.txt** but violation could lead to legal troubles.
- The targeted website can also block web crawler's IP address permanently if guidelines are not followed.
- R provides R package to extract data from website, but it does not support everything. There are many other tools available with R and Python that developers need to keep themselves updated instead of reinventing the wheel.

Data Extraction / Execution

First step in extracting data from the website is crawl through different web pages by changing parameters in the URL as mentioned in previous section. Extract following fields for each runner, which in later steps can be restrcutred the desired format.

Table 1: Data Fields on the Website

Field Name	Field Description
Year	Year of participation
Name	Last Name and First Name of the Runner
Age	Age
Pace	Miles per hour
PiSTiS	Rank of the runner in Male/Female category
Division	Division
PiDTiD	Rank in Runner's Division
HomeTown	Runner's Home Town

Data extracted from the wesbite loaded into R dataframe. R's rvest package was used to extract data from the HTML page on the Cherry Blossom website. All R methods are written in *cs02_methods.R* file. Method *loadDF* calls rvest functions *read_html*, *html_text*, regular expression along with custom made functions getURL, getPlayerRecord to read player's record to store into R dataframe. Values of the fields were loaded as it is from the HTML tables to R dataframe. First five records are as shown in below table.

Table 2: Runner's data parsed into dataframe

year	name	age	time	Pace	PiSTiS	Division	PiDTiD	Hometown
1999	Jane Omoro (W)	26	0:53:37	5:22	1/2358	W2529	1/559	Kenya
1999	Jane Ngotho (W)	29	0:53:38	5:22	2/2358	W2529	2/559	Kenya
1999	Lidiya Grigoryeva (W)	NA	0:53:40	5:22	3/2358	NA	NA	Russia
1999	Eunice Sagero (W)	20	0:53:55	5:24	4/2358	W2024	1/196	Kenya
1999	Alla Zhilyayeva (W)	29	0:54:08	5:25	5/2358	W2529	3/559	Russia

Missing values

As we can see in above table shows there few NAs in the data.These are missing values. Fig 2 shows plot of missing values. It shows there are less than 20 records where Age,PiDTiD, Division is missing and around 70 records where hometown is missing.Missing Age is more of concern than hometown as Age is most crucial for our analysis. But missing values contributes only 0.025% in the required fields.

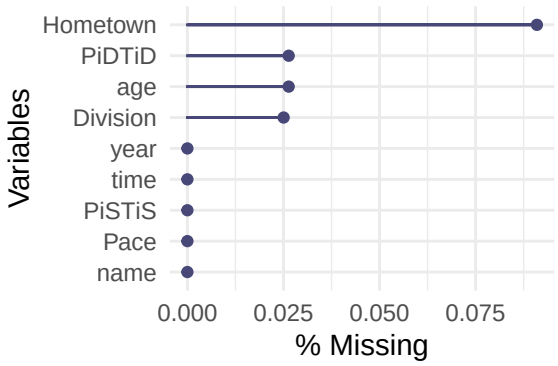

Figure 2: Missing Values

Impute or Drop Missing values

In R VIM package can be use to determine MCAR, MAR and MNAR conditions on missing data.[3][4]

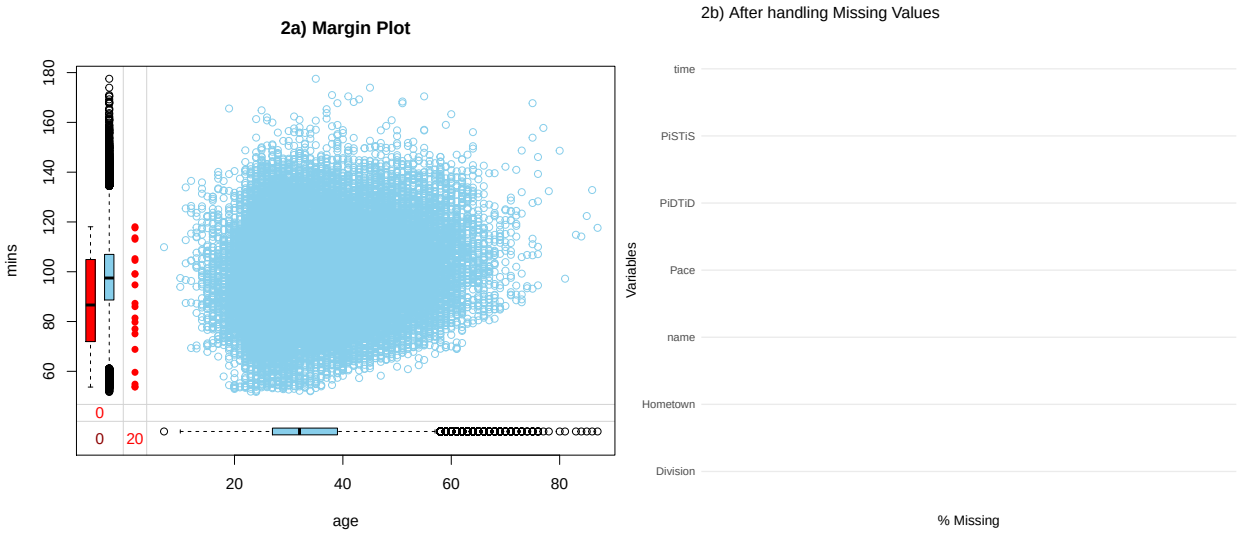

Figure 3: Missing Values Actions

Interpretation of the marginplot goes as follows:

- From the left vertical boxplots, blue is distribution of minutes when age is present
- Red boxplot distribution of minutes when age is not present.
- Age values are missing at 0.02% also from red and blue boxplot show these distributions are overlapping
- Therefore missing values doesn't seem to violate MCAR assumptions.
- Values are Missing Completely at random and hence removing these values will not create any bias.
- Imputation is not required therefore 19 observations have been removed.
- Missing Hometown values are replaced by "UNKNOWN" hometown.
- After removing rows with NA in age, only one value is NA in column PiDTiD, upon observation it was found that this player shares rank under women category and they both belong to same Division and apparently shares same rank in the divison as well. Therefore Rank in her division has hard-coded to "961/1257" where 961 is her rank and 1257 is total player in her division.
- Plot 2b) shows there is nothing missing in the data after handling all missing values.

Preparing final dataset

Final dataset now contains following fields. Few variables are derived from the original and changed their datatypes as follows.

e.g. PiDTiD is character string made up of Rank in Division and Total Runners in Division.

e.g. 9/100 indicates runner's rank is 9 in his division and there are total 100 runners in his division. These values are split into two columns in the final dataset.

Table shows which are orignal fields and which are derived and from which it is derived.

Table 3: Final structure of dataframe

Field Name	Original/Derived	Data Type	Field Description
Year	Original	Integer	Year of participation
Name	Original	String	Last Name and First Name of the Runner
Age	Original	Integer	Age
mins	Derived from time	Float	Total Minutes
PaceMins	Derived from Pace	Float	Miles per hour
rankWomens	Derived from PiSTiS	Integer	Rank in woman's category
TotalWomens	Derived from PiSTiS	Integer	Total women particiapants
Division	Original	String	Division
rankDivision	Derived from PiDTiD	Integer	Rank in Runner's Division
TotalDivisions	Derived from PiDTiD	Integer	Total Participants in Division
HomeTown	Original	String	Runner's Home Town
country	Derived from Hometown	String	Runner's Home Country

Structure of dataframe

```
'data.frame':   75846 obs. of  12 variables:
 $ year          : int  1999 1999 1999 1999 1999 1999 1999 1999 ...
 $ name          : chr  "Jane Omoro (W)" "Jane Ngotho (W)" "Eunice Sagero (W)" ...
 $ age           : int  26 29 20 29 24 38 27 30 ...
 $ mins          : num  53.6 53.6 53.9 54.1 ...
 $ PaceMins      : num  5.37 5.37 5.4 5.42 ...
 $ rankWomens    : int  1 2 4 5 6 7 9 10 ...
 $ TotalWomens   : int  2358 2358 2358 2358 2358 2358 2358 2358 ...
 $ Division      : chr  "W2529" "W2529" "W2024" ...
 $ rankDivision  : num  1 2 1 3 2 1 4 1 ...
 $ TotalDivisions: chr  "559" "559" "196" ...
 $ Hometown      : chr  "Kenya" "Kenya" "Kenya" ...
 $ country       : chr  "Kenya" "Kenya" "Kenya" ...
```

Summary of the data

```
+--------------+------------------+---------------+----------------+
|     year     |       name       |      age      |      mins      |
+==============+==================+===============+================+
| Min.  :1999  |   Length:75846   | Min.  : 7.00  | Min.  : 51.73  |
+--------------+------------------+---------------+----------------+
| 1st Qu.:2005 | Class :character | 1st Qu.:27.00 | 1st Qu.: 88.65 |
```

Median :2008	Mode :character	Median :32.00	Median : 97.48
Mean :2007	NA	Mean :33.85	Mean : 98.22
3rd Qu.:2010	NA	3rd Qu.:39.00	3rd Qu.:106.97
Max. :2012	NA	Max. :87.00	Max. :177.52

Table: Table continues below

PaceMins	rankWomens	TotalWomens	Division
Min. : 5.167	Min. : 1	Min. :2166	Length:75846
1st Qu.: 8.867	1st Qu.:1357	1st Qu.:4333	Class :character
Median : 9.750	Median :2786	Median :6395	Mode :character
Mean : 9.823	Mean :3306	Mean :6610	NA
3rd Qu.:10.700	3rd Qu.:4906	3rd Qu.:8853	NA
Max. :17.750	Max. :9729	Max. :9729	NA

Table: Table continues below

rankDivision	TotalDivisions	Hometown	country
Min. : 1.0	Length:75846	Length:75846	Length:75846
1st Qu.: 165.0	Class :character	Class :character	Class :character
Median : 404.0	Mode :character	Mode :character	Mode :character
Mean : 595.6	NA	NA	NA
3rd Qu.: 816.0	NA	NA	NA
Max. :5302.0	NA	NA	NA

Exploratory Data Analysis

Age Distributions participation over the years

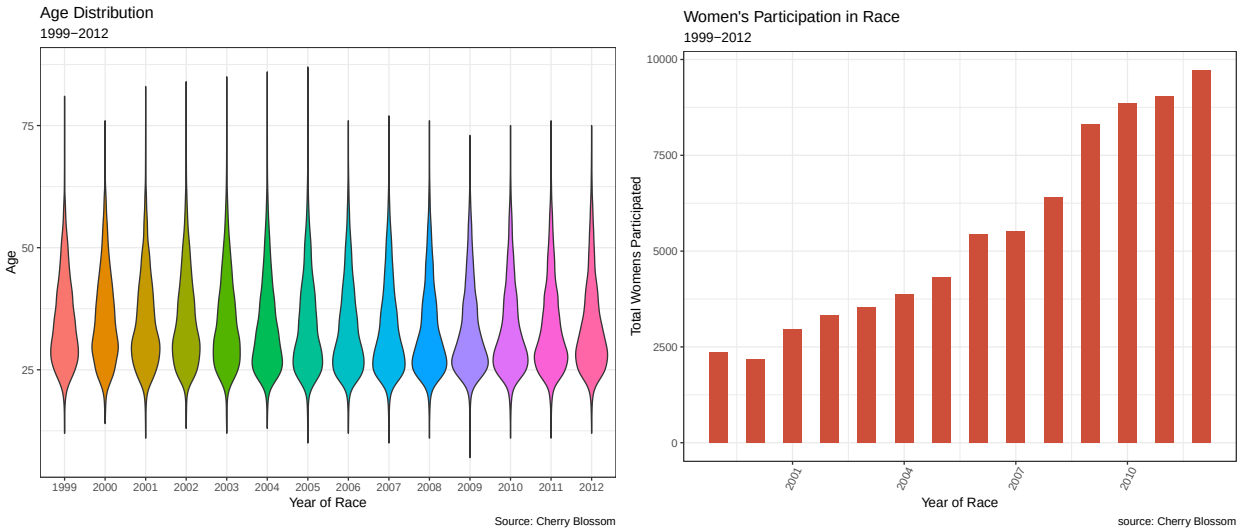

Figure 4: Year wise Distribution

Business Analysis

Conclusion

References

[1] Deborah Nolan; Duncan Temple Lang. Data Science in R.Chapman and Hall/CRC, 2015.

[2] Sports and exercise among Americans : The Economics Daily: U.S. Bureau of Labor Statistics

[3] Templ, Matthias & Alfons, Andreas & Filzmoser, Peter. (2012). Exploring incomplete data using visualization techniques. Advances in Data Analysis and Classification. 6. 29-47. 10.1007/s11634-011-0102-y.

[4] Example of interpretation of VIM::marginplot