NYC Marathon

1. Introduction

Millions of people run marathons world-wide each year. Marathons help raise money for charities, connect runners around the world, and help people to exercise and lead healthy lives while inspiring others to do the same. Our group chose to analyze the the NYC Marathon Data because this marathon is a unique event which brings together extraordinary, driven people from all over the globe, from all age groups, at different levels of physical ability to push their bodies to the limit. **We were curious to find out who these people were, where they came from but also how this variables influence their performance. Additionally we were interested in understanding how large is the participation of woman relative to men and what is the trend. Finally, we explored how to summarize the whole race in a graph**. The team consisted of Arthur Herbout, Antonia Lovjer, Andrea Navarrete and Andres Potapczynski.

Andrea took on the challenge of scraping the data from the NYC Road Runners website, and cleaning it using an R script which she shard with the rest of the group. The five data sets consisted of the NYC marathon data on race finishers for years 2015 to 2018. Andrea and Antonia continued to do a demographic analysis of the data, working towards understanding who were the people participating in the race. Arthur wanted to understand how the race was organized, and used variables such as "official_time" and "gender" to understand the meaning behind the "waves" and how they correlated with performance. Andres took a more in depth approach to the data by clustering the runners according to their performance in order to analyze patterns amongst the different groups.

As a team, we worked on brainstorming ideas for the project, organizing the analysis, completing the exploratory data analysis and visualization, and creating the presentation. Design of the interactive Shiny app was completed as a group, with the development led by Andrea, and contributions by Arthur, Antonia and Andres.

2. Description of data

To answer research questions mentioned in the introduction we generated the data set by web scrapping the official site of the race (for all the details of this process look at section 2.2 below). It is worth mentioning that our data is made from only the competitors that finished the race. In 2018, our data set contains 52,669 runners. In 2017, we had 49,238 runners. In 2016, we had 50,486 runners and finally in 2015, we had 48,742 runners. This section is divided as follows. First we mention the relevant variables that we included in the data (as well as their actual R code names). Then we dive into the source of the information. Finally, we explain (or rather summarize) the intricate detials of web scrapping the data.

2.1 Relevant variables for the analysis

Following is a list of the main variables used for the analysis (we ommit adding the details for variables that we did not use in any of our analysis).

- `gender`: the gender of the athlete encoded as "M" for male and "F" for female.
- `age`: the age of the athlete.
- `official_time`: the time it took the athlete to finish the race.
- `gun_time`: the actual time when the athlete got to the finish line (remember that some athlete started at different times).
- `name`: the name of the athlete (we used this to keep track of how people improve or worsen their performance if they appear in other years).

- `city`: the city where the athlete comes from.
- `state`: the parse state abbreviation (only for the US) where the athlete comes from.
- `stat_name`: as above but the complete name of the state.
- `country`: the country where the athlete comes from.
- `lat`: the latitude given by the country where the athlete comes from.
- `long`: similar to above but now the longitude.
- `split_<x>k` the time when the athlete got to the xth split where $x \in \{5, 10, 15, 20, 25, 30, 35, 40\}$.
- `type`: the inferred group category. "R" is for runners, "H" for handicap and "W" for wheelchair.
- `team`: the name of the team if the athlete belong to one. For example, professional athletes might be sponsored say by *NIKE*, other athletes represent charities associations like *Team for Kids* and finally some others might belong to an amateur group like *North Brooklyn Runners.* Nonetheless the majority does not report belonging to a group and are added as *NA*.

2.2 Source

Our data set comes from the *TCS New York City Marathon Results* which is hosted by the New York City Road Runners. We scrapped the data from their official website link. Accessing the previous link renders the following view:

[][./pics/webpage.png .png]

Thus, as it can be seen above the information that the website provides falls into two categories.

- **Demographics**: Age, gender and country / city
- **Performance metrics**: Official time, pace per mile and the time per 5 kilometer split

The main difficulty with this data is that it is embedded in a web page. Even though it is easily accessible it is hard to download locally! Thus we had figure out how to web scrape it.

2.3 Web Scrapping

Web scrapping was more time consuming than we thought. Even though we used the really well-develop package of `BeautifulSoup` it still required us to overcome two main obstacles (1) understanding the web scrapping process and (2) to come up with the template that our program should use in order to find the information.

To get a complete understanding of the process, we made intensive use of different resources online: either YouTube videos or other Q&A websites such as `StackOverflow`. A particular source that we found useful was link. Moreover we became acquainted with the myriad of details that were not evident when we started the process. First, we had to distribute our work in a computer engine on the cloud. The difficulty is that web scrapping is a really slow process. For example, we have to run a long `for-loop` were we have to make a different connection for each of the over 50 K participants of the marathon (this times the number of year that we downloaded). Moreover, we cannot make a constant connection to the web site since that could potentially be considered as an attack. Thus we had to replicate the pace that a person takes to access the website. Thus we had to set-up the computer engine and let it do the work (which took approximately 3 days)

In term of coming up with a template for our program to run. We had to learn (actually before starting D3) how to read all the html elements of a web page and embed that knowledge into a script for `BeautifulSoup` to perform its magic. Furthermore, this is a sensitive procedure. Altering any location of an element in the web page renders the script useless. Thus, on the one hand, we had to make several "robustness checks" for our script before letting it run (because we risk losing days of work). On the other hand, we had to develop for each year a new template! Every year the layout was altered in some particular way, thus we had to adapt for those changes. However, we were still able to download the same information every year (which made the template creating process slightly more exciting).

At the end, we were quite happy that we were able to download the data. Mostly because we were doing an analysis that excited us but also because, due to the difficulty of the process, we knew that not many people had done this analysis before.

3. Analysis of data quality

Due to the web scrapping procedure we were afraid that the data would have a bunch of errors, but actually it did not. We divided our analysis for data quality into four sections. First, we analyze the general missing values patters. Second, we delve into the one source of missing values: *messy locations*. Then, we analyze the other source of mising values: *inconsistent time formats*. Finally, we comment on the nature of outliers in this data.

3.1 Missing values and its patterns

[][Add visna plot from raw data to show what are the missing percentages and patterns - I believe Joyce expects this when treating missing values]

3.2 Messy Locations

Sometimes the geographical information displayed in the webpage related to the city of the runner, sometimes to the her state and some others times to the county she was was born (definitely our web scrapping program was not so smart as to make those types of swaps). Then we had to make a decision on what to do about this: either to spend quite some time on mapping manually all the cases or do some kind of parsing and leave aside the cases without a match.

A priori we were unsure what to do, we had mixed feelings about not using this information, which was a key variable for our analysis. Hence to make a decision first we assessed how bad our problem was. For this we ran our parsing procedure to see how many cases lack a match. Our parsing procedure consisted on two steps: first, we would try to parse the states and for the remaining we would pass our country mapping (which was tractable). Fortunately, we got quite lucky and the procedure almost mapped completely off-the-shelve. The reason is that the majority of the inputs that had a state name in the geographical location where from the USA and also because the R function `state.abb` was quite robust and it allowed us to get the state abbreviations. Therefore, we left the rest of the cases as NA and move on into adding the latitude and longitude of the place.

3.3 Inconsistent Time Formats

We have never had a good experience when working with a time series data sets and this was, unfortunately, not the exception. We made extensive use of the available tools to go from formats like “01:00:23” to an actual meaningful time stamp (on this, it appears as if rather than spreading the best practices of presenting time variables in a data set, the world is rather creating more sophisticated tools to go over every eventuality).

The previous discussed problem was multiplied by all the columns that contained time data which was quite a few: the finished time, the time per 5 k split, the official time and many more. Thus after running our parsing function on each column we filtered out any observation that had at least one missing value (that is, not a parsable time) in any of the main variables such as `gun_time`, `official_time`, and each of the splits `split_<x>k`.

3.4 Outliers

Outliers in this context come either from poor performing runners, excellent performing runners and other type of competitions categories (wheelchair and handicap). Thus, the notion of *outlier* here stems from mixing many different cohorts of people (professional athletes, standard competitors, wheelchair and handicap) in the data rather than by having some plausible error in a certain recording of a variable. Although there were some people that finished the marathon in around 7 hours. But we are uncertain if this is a mistake or maybe they burnt out and finished the race as they could.

Hence, after separating the data into different groups, now there where not evident outliers (in the sense of points going past the whiskers in the boxplots). For example the top 100 runners all stayed very tied. Whereas the latters groupd had similar runnig times but there was more variance. [][Think if it is worth adding the comparison of this two boxplots]

4. Main analysis

This section contains the main results of our exploratory data analysis. Each of the subsections below addresses one of the research questions that we posed in the introduction.

4.1 Demographic

(over time)

- Age
- Gender
- Location
- Teams (ranks)

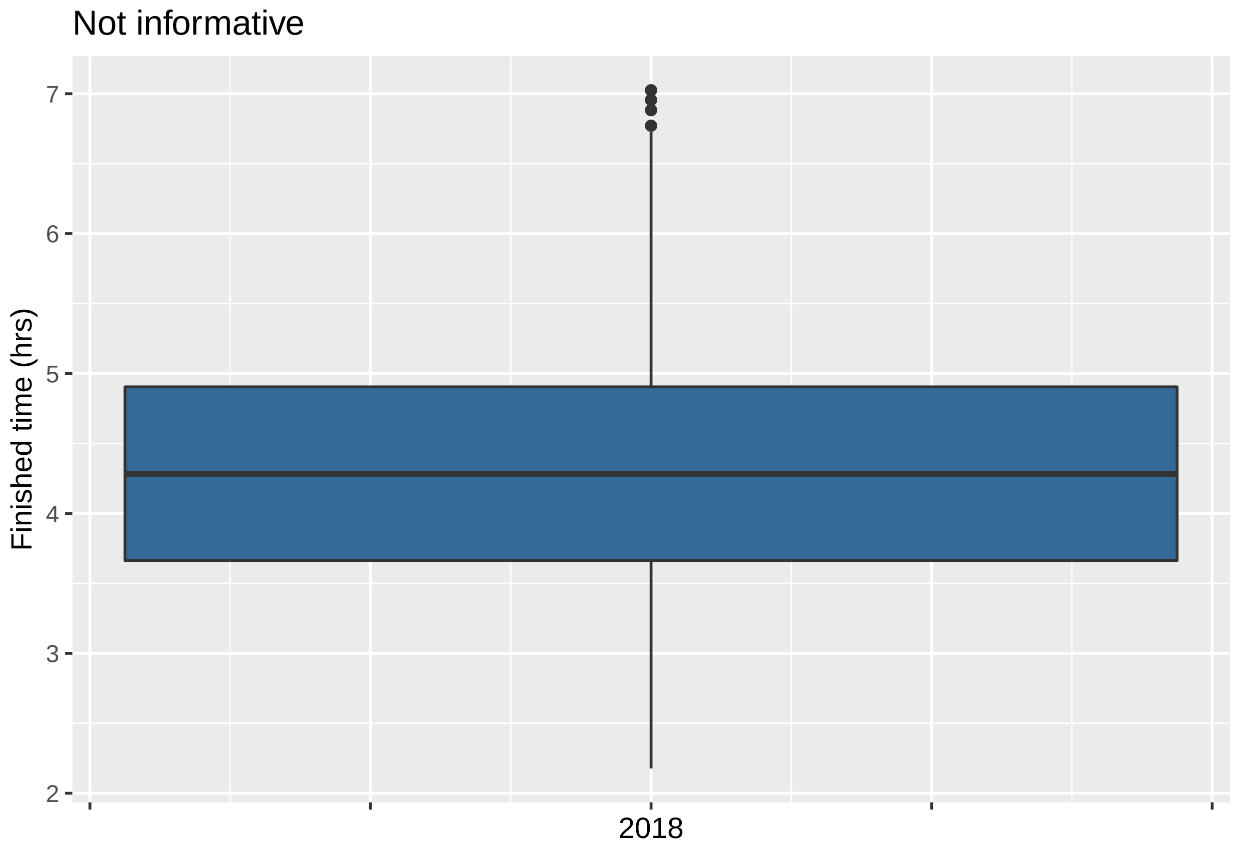
4.2 Running Variables

- Strategy (clusters)
- Waves

4.3 Summarizing the whole race in a graph

We wanted to understand how we could summarize the race in a single graph. This was a challenge because we were unsure of what to graph to choose: should it be a histogram a boxpplot. Moreover, what variable should we used. The `official_time` but what about the performance between the different splits? Maybe some athlete had a strong start but finished poorly. Our first take to this problem was to generate a boxplot of the running time.

```
gbox <- ggplot(data = df_embed) +
  geom_boxplot(aes(x = year, y = sec_8 / 3600, fill = year)) +
  ggtitle('Not informative') +
  ylab('Finished time (hrs)') +
  theme(legend.position="none",
        axis.text.x = element_blank()) +
  xlab('2018')
gbox
```



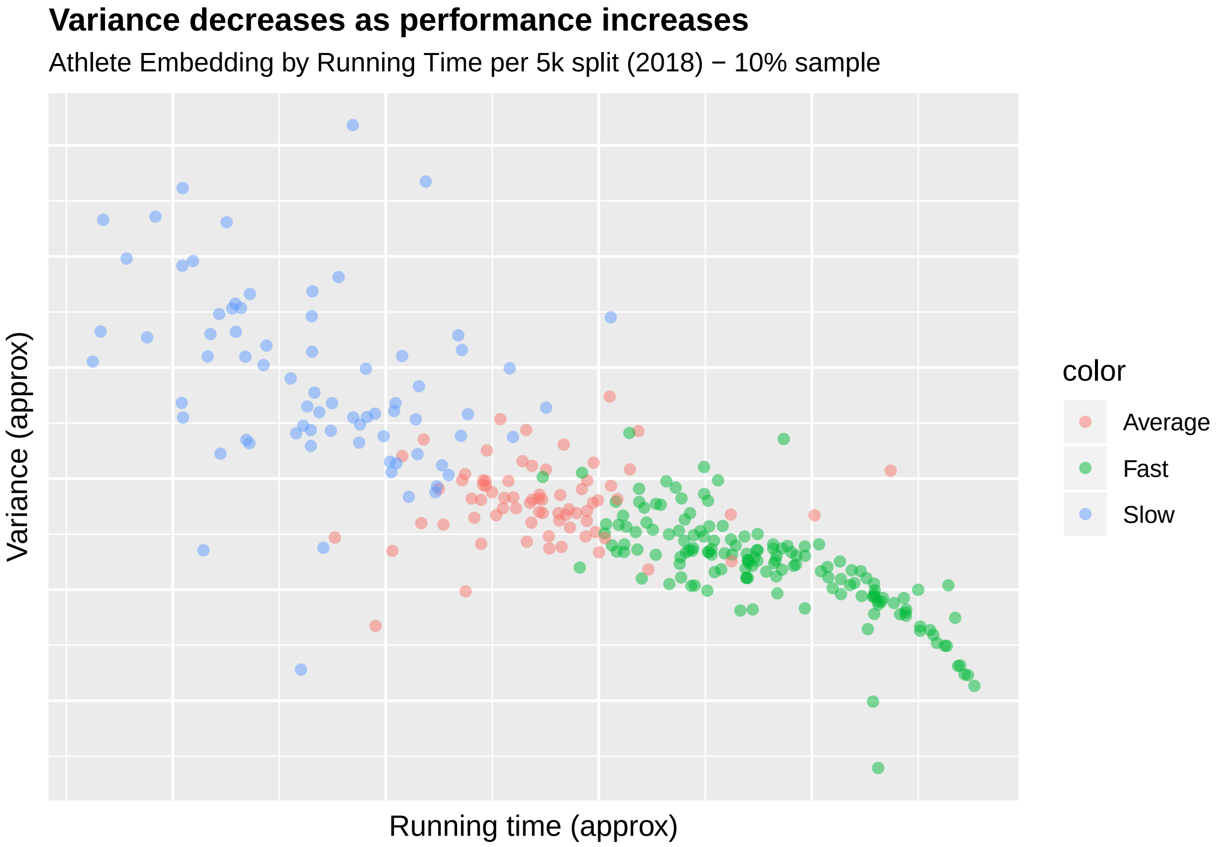
as we see above, the boxplot is not as informative as we would like. It tells us that the finishing time was over 4 hours, where the mass concentrates (from slightly below 4 hours to 5 hours) and that the max and min. But we were not satisfied. Then, based on one of our ML homeworks, we saw that there was a technique that could go from distances between the observations to a two dimensional embedding. As an illustration, this technique could take the distances from all the cities within the US and recover the whole map

[][Add the distance matrix and the map recovered… maybe]

where we see how New York and Boston are bunched together were as Seattle and Miami are thrown to two different corners. Thus, we thought that maybe this two dimensional embedding would be a useful summary of our data. The only missing ingredient was to define what the distance between different runners ment.

We found that the most useful summary resulted when we defined the distance between runners as the difference between each of its 5 k splits. The resulting embedding that we got for 2018 and only 10 % of the data is

```
df_embed$color = 'Fast'
df_embed$color[df_embed$sec_8 > quantile(df_embed$sec_8, 0.5)] = 'Average'
df_embed$color[df_embed$sec_8 > quantile(df_embed$sec_8, 0.75)] = 'Slow'
g <- ggplot(data = df_embed, mapping = aes(x = x, y = y)) +
  geom_point(aes(color=color), alpha = 0.5) +
  xlab('Running time (approx)') +
  ylab('Variance (approx)') +
  labs(title = 'Variance decreases as performance increases',
       subtitle = 'Athlete Embedding by Running Time per 5k split (2018) - 10% sample') +
  theme(plot.title = element_text(size = 12, face = "bold")) +
  theme(plot.subtitle = element_text(size = 10),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank(),
        axis.ticks.y=element_blank(),
        axis.text.y=element_blank())
g
```



This is really cool! Before going into the details of the graph it is worth mentioning a couple of points. **The x and y axis are not the actual variables displayed in the labels but rather an approximation or interpretation** (it is like in PCA, where the two principal components are not a single variable but rather a combination of many that might render a certain interpretation). Thus, as seen from the graph **the x-axis displays the running time** where the time increases as you move to the left; the first green dot to the right was the fastest runner. It is worth pointing out that I colored the data points to show that this interpretation holds, but the embedding did what it thought best. In terms of **the y-axis**, the interpretation is that this coordinate **captures the variance** in running time per 5 k split. Since points that had disimilar performance on each split get separated, then the more disimilar you are on each 5 k split to the atheletes that had a similar running time, the further away you get pulled from them. As an illustration, take the green dot that is closest to the x-axis. That athlete had a great overall running time but the reason that it got thrown away from the elite “pack” is that it did not maintain a constant pace in all the splits as the rest of the top performers. What happened was that this athlete decrease severily its speed at the last 5 k split and that is why it got separated.

Now, past the introduction to this graph, the main insights are the following. Look how **the performance differences decelerate as we move to the right**. The graph resembles a logarithm curve that has been rotated on the x-axis. Based on this curvature, we see that a change for the green dots move you further above than in the red dots. **Thus, for the faster runners small perturbations in their running times sets them more further appart than for slower runners**. This make a lot of intuitive sense since

Just for completeness, mathematically the problem that the embedding solves is

$$\min_x L = \sum_{i,j} \left(\|x_i - x_j\|_2 - D_{ij} \right)^2$$

5. Executive Summary

[][Write the conclusion – need to have all the graphs from the previous section]

6. Interactive component

[][Add the link to the interactive component] [][Explain how to use the visualization. What to move, what to look for, etc]

7. Conclusion

[][Discuss limitations, lessons learned and future directions]

Questions

1. Is gender a significant indicator of performance? (Arthur)
2. How could we summarize the whole race in a graph? (Andres)
3. How has the gender ratio changed over time? (Andrea & Antonia)
4. How do performance change by location? (Andrea & Antonia)