Sneakerhead Gear

STAT 385 FA2018 - Team XSWL

Ziwei Liu
Zepeng Xiao
Yuquan Zheng

November 19, 2018

Abstract

This project aims to design a shiny-powered app to address an underlying difficulty of many sneaker shoes lovers & collectors, which is to buy or trade sneakers at reasonable prices. The goal is to scrape real-time price information from StockX, one of the most popular stock market for sneakers, and produce visualizations as well as give predictions for the prices of popular items by incorporating statistical computing methods. The motivation behind this project is a shared experience among the group members about previous hardships involved with trading sneaker shoes, and a strong desire to work with web scrapping and to practice statistical methods with R. The expected gain is hence as stated above.

Contents

1 Introduction **2**

2 Related Work **2**

3 Methods **2**

4 Feasibility **3**

5 Conclusion **3**

6 Appendix **4**

6.1 Formatting Notes 4

7 References **5**

1 Introduction

In this project, we are dealing with predicting the market for sneaker shoes. More specifically, we are interested in investigating the price variation in a time period of any given brand of shoes, and how does that help predict future trends. In recent years, the featuring of fashionably designed sneaker shoes has been heated remarkably on a global scale. SportsOneSource (2015) presents that the international sneaker market has reached about $55 billion (as cited in Weinswig, 2016). Moreover, StockX is a marketplace where people can buy or sell sneakers in real time. According to Matt Powell, "the annual market for sneaker reselling has grown to somewhere between $ 200 million and $ 500 million" (as cited in Noskova, 2016). If we can successfully predict the reselling prices for both buyers and sellers, then it can greatly benefit the cast sneaker lovers.

One of the problems in trading on StockX is similar to that of traditional stock trades: the variation of the market prices is stochastic, therefore it is hard to guarantee that people can maximize their benefits. Our idea is to scrape data from stockX, namely the bid/ask prices and other attributes for a given type of sneaker shoes in a period of time, which is available as each of transactions will be recorded. We believe that by using R's `ggplot` and `shiny` packages, we can present the price information in a simpler and more user-friendly way. Further, by taking advantages of R's statistical computing and data manipulation abilities, we can try to predict how the price of a particular sneaker shoe will change in the near future and also transform that into graphics, hence practical for potential users, in the sense that buyers can buy shoes at lower prices and sellers can sell their sneakers at their desired price.

In completing this project, we expect to practice with programming techbiques prevailing in R, and experience collaborative development to come up with analysis and behavioural suggestions based on data, all of which adhere to the general purposes of this course as stated in the syllabus.

- Consider adding subsections in this section. For example, consider adding a **data** subsection. The data subsection would describe your data. What is it? Where did it come from? How will it be useful in answering your problem?

2 Related Work

Our group has decided to do web scrapping from the begining, so it was only the matter of choosing a webpage that is easy to scrape from and which kind of analysis should we perform. Later we discovered our common interest in sneaker shoes, so that we decided to scrape StockX. As for the detailed contents, we decided to do visualizations for historical prices, and then we discovered a related study which can be accessed at:

Scraping StockX: Adidas Yeezy Resell Analysis.

It offered a lot of ideas pertaining to scrapping and data analysis for a certain type of sneaker shoes. In ensuring originality, we decided to incorporate time-series analysis in order to give prediction about shoes prices. There have been similar analysis project focused on forecasting stock prices, and we think our originality lies in bringing this idea to the market of sneaker shoes.

3 Methods

The majority of your code should be *suppressed* from the displaying in this section. Please refer to code and figures placed in the appendix. The latter can be referenced using:

`Figure \\ref{fig:code-chunk-name-here}.`

For example, the figure of the data science workflow is accessible via Figure **??**.

To satisfy this section, provide detailed responses for the following:

- What packages will you use in your implementation?

The main body of this project can be broken into three parts: construction of data by scrapping data from StockX, design and implementation of user interface and app server with `shiny`, and barplot/line graph visualizations of price information and prediction results with `ggplot2`. Our intended action is to perform time-series analysis through fitting ARIMA models (with built-in function `arima`) that is available in base R. The packages that need to be specifically loaded are:

1. `ggplot2` for visualizations
2. `rvest` for web scrapping
3. `tidyr` for data cleaning
4. `stringr` for string manipulation

The group will write codes to first complete web scrapping, then potentially split or combine some of the variables for data cleaning. In order to fulfill the requirement for the number of attributes, there is also the need to merge data using SQL or basic R. To provide prediction of prices would need hands-on experience with `arima`, and the plotting of line graphs demonstrating price trends requires using `ggplot2`. Finally, codes are needed for building a `shiny` application.

The main source of practices on this project has been from Homework 8 and all previous homeworks as well as practices of statistical methods in other stats courses taken by the group members.

4 Feasibility

We originally wanted to enable users to select any type of sneakers from any brands and give price predictions and visualizations, but then we realized to enable searching and then scrapping the corresponding data from different webpages requires a higher level of web scrapping technique, which takes time to master. We then decided that for this project, we would allow 5 to 10 pairs of shoes to choose from, and the current web scrapping techniques would still ensure that the information is real-time. Therefore we believe this project can be finished before the end of semester, as we can roughly send one week constructing data and building & testing UI and server, and another one to two week implementing time-series analysis and compile reports and video demo. To make the project goes, our group has decided to meet at least twice a week and take advantages of the office hours.

As for tasks split, Ziwei Liu will undertake the part of fitting ARIMA models on getting time-series analysis works, while Zepeng Xiao will take charge of data processing and data cleaning after scrapping. Finally, Yuquan Zheng will be in charge of app design and report writing.

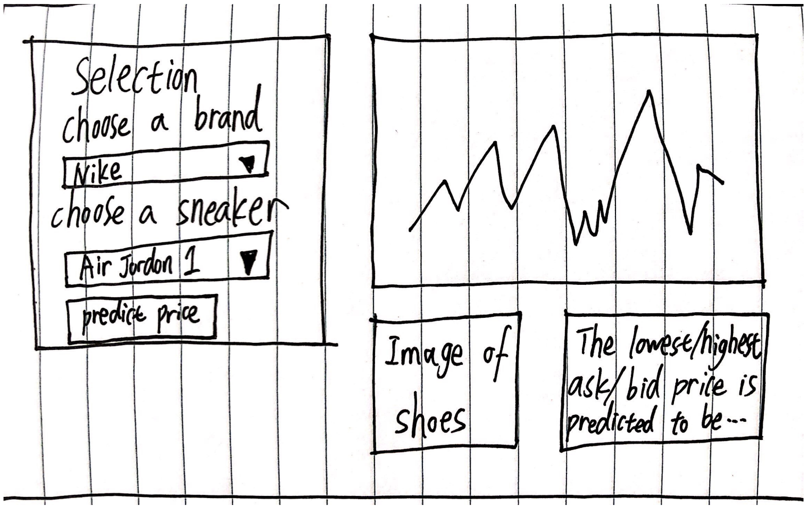
5 Conclusion

In summary, this project aims to offer a gear for sneaker lovers to buy or sell their sneakers at desired prices, eased by visualization and powered by data scrapped from StockX. It resolves to a certain degree the uncertainty in market prices faced by sneaker traders when using StockX. The novelty will arise from the attempt to apply time-series analysis to predict shoes prices.

6 Appendix

The **Appendix** section contains figures, sample data, and other miscellaneous entries. Generally, this sketch seeks to contain all of your *planning* information.

- Provide the sketches of visualisations and the shiny application.



- Provide an overview on the desired functions.
  - What is a function's input? Output? How are functions related to each other.
  - For example, `read_data("hospital_data.csv")` must be called before `tidy_hospital()`, et cetera.
- Provide a sample of the data set you intend to use (~10 observations).

If you used previous code chunks within the document, this information can be dynamically retrieved and embedded.

6.1 Formatting Notes

You **are** required to use BibTeX for references. With BibTeX, we could reference the `rmarkdown` paper (Allaire et al. 2015) or the tidy data paper. (Wickham and others 2014) Some details can be found in the `bookdown` book. Also, hint, Google Scholar makes obtaining BibTeX reference extremely easy. For more details, see the next section...

7 References

The **References** section acts as a bibliography for all papers referenced in the **Introduction**, **Related Works**, and **Method** sections. The references should be formated in Chicago author-date format, which is the default for RMarkdown.

- Provide a list (5+) of papers or items you have read to write this proposal.
- Please list all *R* packages or software referenced.

To acquire software citation information, *R* has a built-in command that creates a BibTex and in-line text citation. To generate the citation of an installed *R* package, type:

```r
# In R
citation(package="pkg_name")
```

For example, to cite `dplyr`, one would generate the BibTex entry from:

```r
citation(package="dplyr")
```

```
@Manual{dplyr:2018,
    title = {dplyr: A Grammar of Data Manipulation},
    author = {Hadley Wickham and Romain François and Lionel Henry and Kirill Müller},
    year = {2018},
    note = {R package version 0.7.7},
    url = {https://CRAN.R-project.org/package=dplyr},
}
```

Note, we added a “name” to the autogenerated citation of `dplyr:2018`. Using this name, we can reference the work within the paper via (Wickham et al. 2018) or Wickham et al. (2018).

Allaire, JJ, Joe Cheng, Yihui Xie, Jonathan McPherson, Winston Chang, Jeff Allen, Hadley Wickham, Aron Atkins, and Rob Hyndman. 2015. "Rmarkdown: Dynamic Documents for R." *R Package Version 0.5*.

Wickham, Hadley, Romain François, Lionel Henry, and Kirill Müller. 2018. *Dplyr: A Grammar of Data Manipulation*. https://CRAN.R-project.org/package=dplyr.

Wickham, Hadley, and others. 2014. "Tidy Data." *Journal of Statistical Software* 59 (10). Foundation for Open Access Statistics: 1–23.