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

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

which offered a lot of ideas pertaining to scrapping and data analysis for certain types 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 prediction method 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. `regex` for string manipulation

- What code will the group need to write for the project?
- Provide low-fidelity prototypes (e.g. *sketches* on paper) in the **Appendix** of:
  - Visualisations
    - What kinds of graphs will you use?
    - Label axes, provide a title, and mention any interactivity.
  - Interface
    - All projects need a Shiny Application.
    - Sketch how a user will work with the shiny application.
- What have you done or learned so far for the project?

4 Feasibility

The **Feasibility** section is meant to act as a way to reflect upon the proposal. Generally speaking, there will be three weeks of heavy development time afforded to the group. Building a detailed ecosystem or heavily scripting in a different language will likely not lead your team to success. Hence, please provide a project management overview of *who* on your team will be doing *what* and *when* by answering:

- Is this project able to be completed before the end of the semester?
- What steps must occur to complete the project before the end of the semester?
- What is the work plan to accomplish the necessary tasks before the end of the semester?
  - Specify who is doing what and when.
  - Consider making a Gantt chart to highlight each stage of the project.

5 Conclusion

The **Conclusion** section provides a summary of the entire proposal. This acts as the final paragraph that can be used to justify the work being proposed. In general, this means you should make one last push to identify the problem, potential solution, and its novelty.

If a group's project is well written, uses thoughtful and creative approaches, and is sufficiently interesting you may be asked to have your work "published" as an example for future students. **All group members will have to agree to publication.** You may also be asked to make edits before publication, but you should be sure to **proofread** and **spellcheck** your work before your initial submission.

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.

```
# Sets default chunk options
knitr::opts_chunk$set(
  # Figures/Images will be centered
  fig.align = "center",
  # Code will not be displayed unless `echo = TRUE` is set for a chunk
  echo = FALSE,
  # Messages are suppressed
  message = FALSE,
  # Warnings are suppressed
  warning = FALSE
)
# All packages needed should be loaded in this chunk
pkg_list = c('knitr', 'kableExtra', 'magrittr')

# Determine what packages are NOT installed already.
to_install_pkgs = pkg_list[!(pkg_list %in% installed.packages()[,"Package"])]

# Install the missing packages
if(length(to_install_pkgs)) {
  install.packages(to_install_pkgs, repos = "https://cloud.r-project.org")
}

# Load all packages
sapply(pkg_list, require, character.only = TRUE)

kable(
  head(mtcars, 20),
  format = "latex",
  caption = "This is an example of a table in the Appendix. Notice that it is way too big, and has way t
  booktabs = TRUE
) %>%
  kable_styling(latex_options = c("striped", "scale_down"))
kable(
  head(mtcars, 20),
  format = "latex",
  caption = "This is another example of a ridiculous table. Notice that it is automatically numbered.",
  booktabs = TRUE
) %>%
  kable_styling(latex_options = c("striped", "scale_down"))
```

Table 1: This is an example of a table in the Appendix. Notice that it is way too big, and has way too much information. We use the `kableExtra` package to shrink it down, but even then, no one would actually read this table.

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4
Merc 280C	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4
Merc 450SE	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3
Merc 450SL	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3
Merc 450SLC	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3
Cadillac Fleetwood	10.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4
Lincoln Continental	10.4	8	460.0	215	3.00	5.424	17.82	0	0	3	4
Chrysler Imperial	14.7	8	440.0	230	3.23	5.345	17.42	0	0	3	4
Fiat 128	32.4	4	78.7	66	4.08	2.200	19.47	1	1	4	1
Honda Civic	30.4	4	75.7	52	4.93	1.615	18.52	1	1	4	2
Toyota Corolla	33.9	4	71.1	65	4.22	1.835	19.90	1	1	4	1

Table 2: This is another example of a ridiculous table. Notice that it is automatically numbered.

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4
Merc 280C	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4
Merc 450SE	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3
Merc 450SL	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3
Merc 450SLC	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3
Cadillac Fleetwood	10.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4
Lincoln Continental	10.4	8	460.0	215	3.00	5.424	17.82	0	0	3	4
Chrysler Imperial	14.7	8	440.0	230	3.23	5.345	17.42	0	0	3	4
Fiat 128	32.4	4	78.7	66	4.08	2.200	19.47	1	1	4	1
Honda Civic	30.4	4	75.7	52	4.93	1.615	18.52	1	1	4	2
Toyota Corolla	33.9	4	71.1	65	4.22	1.835	19.90	1	1	4	1

6.1 Formatting Notes

6.1.1 `R` Code and `rmarkdown`

An important part of the report is communicating results in a well-formatted manner. This template document should help a lot with that task. Some thoughts on using `R` and `rmarkdown`:

- Chunks are set to not echo by default in this document.
- Consider naming your chunks. This will be necessary for referencing chunks that create tables or figures.
- One chunk per table or figure!
- Tables should be created using `knitr::kable()`.
- Consider using `kableExtra()` for better presentation of tables. (Examples in this document.)
- Caption all figures and tables. (Examples in this document.)
- Use the `img/` sub-directory for any external images.
- Use the `data/` sub-directory for any external data.

6.1.2 LaTeX

While you will not directly work with LaTeX, you may wish to have some details on working with TeX can be found in this guide by UIUC Mathematics Professor A.J. Hildebrand.

With `rmarkdown`, LaTeX can be used inline, like this, $a^2 + b^2 = c^2$, or using display mode,

$$\mathbb{E}_{X,Y}\left[(Y - f(X))^2\right] = \mathbb{E}_X \mathbb{E}_{Y|X}\left[(Y - f(X))^2 \mid X = x\right]$$

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:

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

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

```
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.