Alon Lerner

Professor Wang

CS488

05/10/2023

Independent Study Final Report

## Introduction

This independent study is about data analytics of stocks in the stock market. Throughout this study, I collected data about stocks, analyzed it, and visualized it. Through this study, I learned how to think like a researcher and try to discover new things as I progress. In this study, I tried to find different features that affect the stock prices. The information achieved in this study can be beneficial for selecting features for deep learning models or for portfolio construction.

## Methods

This project includes two main scripts. The first one, find_gaps.ipynb, is focused on gap analysis and the second one, earnings_impact.ipynb, is focused on the immediate impact of the earnings releases on the stock prices. In addition, there are other helper scripts (get_earnings.js, get_earnings2.js, and get_earnings3.js) that scrape earnings data from Yahoo Finance website using the JavaScript puppeteer library.

### find_gaps.ipynb

This script is mainly focused on gap analysis. In the first part, data about gaps is collected and displayed in a pandas' dataframe. In the second part, an analysis of the earnings impact is printed. This script uses Python's yfinance library to get data about historical stock prices and get_earnings.js to get data about earrings releases. This script is currently displaying the data

about the Apple stock (AAPL), but can display the data about any stock by simply changing the stock_abbr variable in the first code box. The timeframe could also be easily modified by changing the start and end variables in the first box.

In the beginning, it gets data about a stock through Python's yfinance library and displays its chart. Then, data about gaps is collected and is added to the the dataframe:

- is_gap (boolean) - true if there was a gap on that date, otherwise false.
- is_gap_up (boolean) - true if there was a gap up on that date, otherwise false.
- gap_fill_time (integer) - the time it took for the gap to get filled in days.
- extremum (float) - if it’s a gap up, it displays the max point of a stock at a time frame between the time there was a gap up and when it got filled. if it’s a gap down, it displays the min point of a stock at a time frame between the time there was a gap down and when it got filled.
- Max_return (float) - the maximal return in percentage from a stock from a time a gap occurred to when it got filled.
- gap_size (float) - the size of a gap in $.
- gap_size_ratio (float) - the size of a gap in percentage.

The columns that are gap related are only applicable for rows that present a gap. If the row is not a gap, this value will be None. At the end of this part, the min, max, and average of each column is printed.

In the second part, earnings releases data is scraped and the following columns are added to the dataframe:

- eps (float) - earnings per share.

- estimated_eps (float) - estimated earnings per share according to the Yahoo Finance analysts.
- is_earn_day (boolean) - true if it's an earnings release day, otherwise false.
- earn_timing - the timing of the earnings release. -1 if it's not an earnings release date, 0 if before market hours, 1 if during, and 2 if after. The timing of the earnings releases turned out to be wrong so this column is incorrect.

The columns that are earnings related are only applicable for rows that present an earnings day. If the row is not an earnings date, this value will be None. At the end of this part, the following data about earnings during non market hours is printed: the number of times the stock increased after earnings, the average return after earnings, the correlation coefficient of the return and the eps to its estimation eps ratio, and the correlation coefficient of the return and the eps to its previous eps ratio. However, because the data about earnings time scraped from Yahoo Finance turned out to be wrong, the analysis might not be accurate.

earnings_impact.ipynb

This script is focused on the immediate impacts of earnings releases of the stock prices. The immediate response is calculated as the return of the stock price from the close of earnings day to the close of the following day. This script assumes all earnings are released after market hours.

This script creates a dataframe that contains all the stocks on spx2020.csv and calculates their number of earnings releases, percentage of time the stock went up after a release, and the average return after a release. Additional columns display the same attributes but dividing the earnings into earnings that were higher than estimated (positive surprise) and earnings that were lower than expected (negative surprise). Afterwards histograms of the percentage of time the stock

went up after a release are displayed and two csv files are created: one is not sorted and one is sorted by best average return.

get_earnings

The get_earnings files use the strong JavaScript scraping library puppeteer and get information about earnings. get_earnings.js retrieves a list of the dates of earnings releases, their posted earnings per share, their estimated earnings for share, and their timing (0 for before market hours, 1 for during, and 2 for after). This file retrieves only the earnings that are presented on the first page of that website (approximately back until 1998). get_earnings2.js is able to scrape all pages but is slower. get_earnings3.js uses the Yahoo Finance calendar to retrieve more accurate timing. However, these timings turned out to be incorrect as well.

## Results

The results of this study are dataframes with data about gaps and data about earnings impact. We can see that although most stocks did not have a sufficient pattern of price change after earnings release, some did have a small pattern. For example, NOW stock increased 77.5% the day after an earnings release and ULTA stock's average return the day after earnings releases is 3.96%.

## Summary

This study was insightful and enjoyable. I learned how to think like a researcher and to overcome different obstacles. I especially enjoyed working on the impact of earnings releases as there are not many papers posted about it and it wasn't researched enough before. However, data about it is often limited. There is no source that shows an accurate earnings release timings and retrieving accurate data requires checking it one by one online which is very tedious and often not even

possible. Overall, the results of the study can be beneficial and help investors construct a portfolio for the earnings seasons and build deep learning models.