Order Execution Analysis Progress Report

Greg Stepaniouk

October 27, 2017

1 Introduction

The goal of this project is the predict the transaction costs an investor can expect when an order for securities is routed to an electronic market center. Various security specific factors are being used, along with SEC Rule 605 data, which mandates the disclosure of order execution metrics by market centers. The Rule 605 data is merged with a dataset from the Center for Research in Security Prices, which provides additional security specific factors.

This progress report details the feature transformations performed on the data as well as the qualities of specific features. The process to obtain an initial linear model is also explained and the results of the fit are provided. The initial performance is discussed and the next steps for the project are determined.

2 Rule 605 Data Processing

2.1 Broker Selection

The SEC Rule 605 data is available for every eligible market center and broker-dealer across a wide range of time. It is impractical to obtain and parse every single file, so I have devised a methodology to obtain the greatest amount of significant data in a reasonable time frame. The RANK function on the Bloomberg Terminal allows for the ranking of broker-dealers by dollar volume for a given security.

I have chosen 4 securities to obtain ranking metrics for: AAPL (Apple), MSFT (Microsoft), JPM (JP Morgan), and TSLA (Tesla). The first three of these represent top members of the S&P 500 index by market capitalization, while TSLA has a slightly smaller market capitalization, but is quite popular with retail investors. To select broker-dealers from which to pull the Rule 605 data, I perform an intersection operation on the top 20 broker-dealers for each of the 4 securities, which leaves me with 15 broker-dealers.

2.2 Illustrative Transformations

The 605 data is presented in an inconvenient format and filled with missing values. Each line item represents a group of executed orders, categorized by the security, market center code, order size code, order type code, and the month when the orders were executed. I initially filter for only the order type codes representing market orders and marketable limit orders. These two order types behave essentially the same way at execution.[1] The other order types have the field for average effective spread missing, as they represent situations where the investor has specified a price at or outside of the National Best Bid or Offer (NBBO). The average effective spread is a key component of my output space, so I choose to disregard non-marketable orders initially.

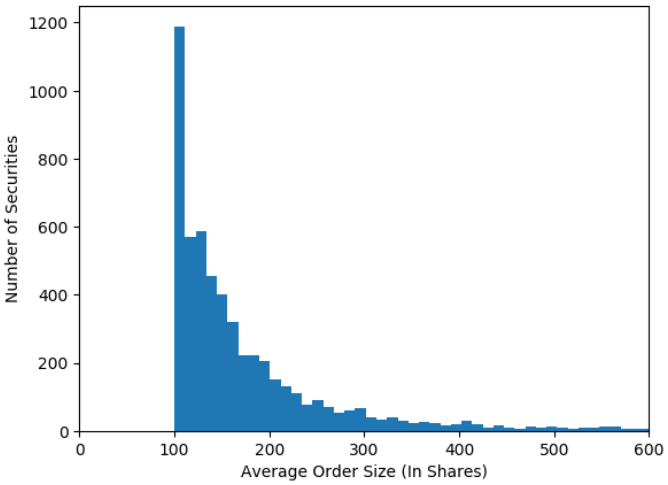
Next, I consolidate the data so each security corresponds to a single line item for a given month. I separate the features into two types, features which can be summed and features which require a weighted average. For example, if I am combining the metric of total covered orders (TCOR), I perform the following transformation to have one value for one security in a given month.

$$\sum_{MC}\sum_{Size}\sum_{Type} TCOR = TCOR\ for\ Security \tag{1}$$

Weighted average transformations are performed with either total covered shares (TCSH) or executed shares (EXSH) as the weighting factor. EXSH is calculated by subtracting canceled shares (CNSH) from TCSH. For example, the weighted average effective spread (AESP) for a security is calculated using the following transformation:

$$\frac{1}{\sum EXSH}\sum_{MC}\sum_{Size}\sum_{Type} AESP_{ijk} * EXSH_{ijk} \tag{2}$$

Although I have removed the granularity provided by order size codes, I preserve order size information by calculating the average shares per order (SHPO) for each security. This is done by dividing total covered shares (TCSH) by total covered orders (TCOR). To prevent underfitting, I avoid removing a useful feature for the sake of consolidation. The following is a histogram of average order sizes for Virtu Financial (a major electronic market maker) during December 2016.



The floor of 100 shares appears to be a result of only "round lots" (i.e. orders of 100 shares or more) being routed to Virtu's market center.

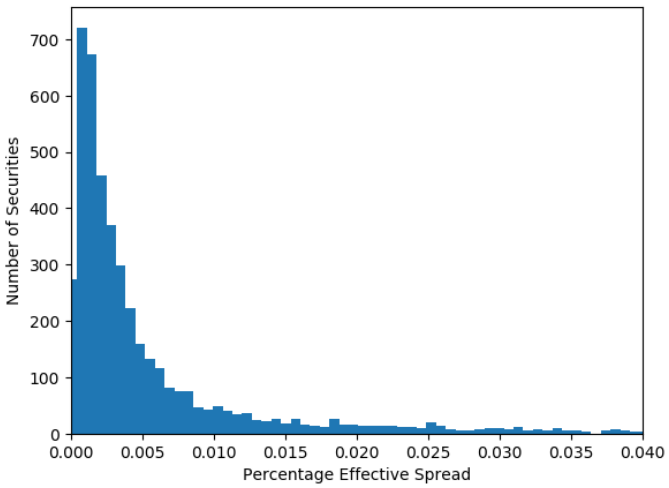
3 Merging with CRSP Data

3.1 Key Fields

Since the original project proposal, I came to the understanding that an additional data source will be required for meaningful conclusions. The Center for Research in Security Prices (CRSP) dataset contains security specific information such as daily pricing, trading volume, and shares outstanding.

A basic issue that this dataset solves is that the effective spread is quoted in dollars in the Rule 605 data. This carries little meaning when not expressed as a percentage of the security's price. For example, if an investor is buying a $1000 share of stock, she might not mind paying $1 of transaction costs per share. If she is buying at $5 per share, the $1 becomes a problem.

The following is a histogram of Virtu Financial's average percentage effective spread during the month of December 2016. I calculate that 70% of average effective spreads were under 0.5% of the share price.



3.2 Monthly Averages

The Rule 605 data only comes in monthly intervals, while the CRSP data is available at a daily resolution. To address this, I take an average of the CRSP features for every month. For example, monthly average percentage effective spread (PESP) is calculated for a security as follows, where AESP is average effective spread.

$$\overline{P} = \frac{1}{N_{days}} \sum_{i=1}^{N_{days}} P_i \tag{3}$$

$$PESP = AESP/\overline{P} \tag{4}$$

3.3 Missing Values

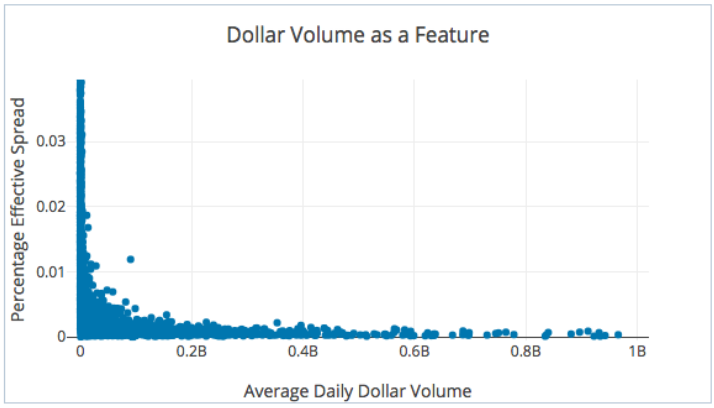
Like most financial data, the CRSP dataset has a substantial amount of missing data. Complete cases make up less than 25% of my chosen feature space. As a result, when taking monthly averages of features, I implement the following logic. This ensures I can have as many usable line items as possible to prevent overfitting by more complex models.

- If there are at least 3 days in a month when the feature is not NA: Take the monthly average over the number of days available.
- Else: Set the monthly average to NA.

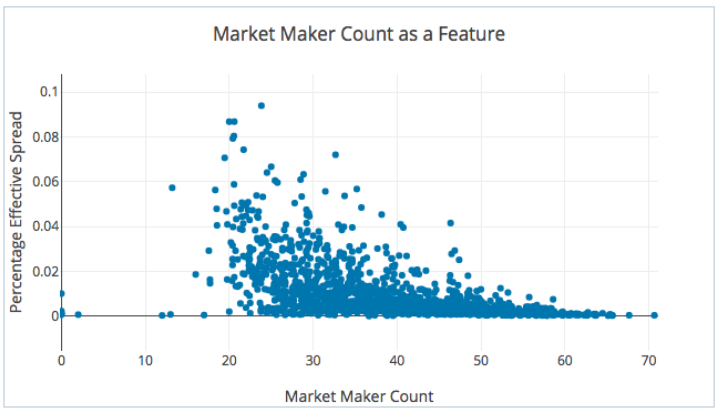
4 Initial Fitting

4.1 Feature Selection

I plotted a variety of features against the percentage effective spread in order to better determine their suitability for an initial model. One feature which I was expecting strong correlation with is dollar volume, a simple transformation of average daily volume multiplied by share price. Intuitively, more liquid stocks might have lower transaction costs. However, a plot of this feature against PESP shows mainly noise and no sign of any linear relationship



One feature which seemed promising is MMCNT from the CRSP dataset. This is the count of designated market makers (essentially liquidity providers) for a given security. The following graph shows a reasonable correlation with percentage effective spread. Unfortunately, this feature is only available for NASDAQ listed securities,[2] so using it in the model would cut my sample space of securities by more than 65%.



For the purpose of an initial linear model, I chose the following features. This narrows the data to roughly 1800 points where all of the features are available. This is not preferable, but it should be enough data for a trial fit.

- DVOL - Average daily dollar volume
- MCAP - Average market capitalization
- SHPO - Average shares per order
- CNSHP - Canceled shares as a percentage of total shares
- MMCNT - Count of designated market makers
- NUMTRD - Average daily number of trades
- Y OUTPUT - Average percentage effective spread

4.2 Linear Model

I fit the trial linear model using Virtu Financial's disclosed trades in December 2016. This does not represent the entire final dataset, but does contain 28,000 line items and information about 1.46 million market and marketable limit orders. I conduct all the previously described transformations on this dataset and extract the chosen features.

I start by adding a standard offset as a column of ones to my feature matrix. I then proceed to winsorize the data at a 1%, 99% level to remove any extreme outliers. The scale of the input measures varies by several orders of magnitude. To make the coefficients interpretable, I standardize all the features as well as my Y output. I fit the following model on a random training set representing 80% of the data.

FEATURE	COEFFICIENT
OFFSET	0.000180009
DOLLAR VOLUME	-0.0383022
MARKET CAP	0.0564476
SHARES/ORDER	0.158772
CANCELLED SHARES %	-0.0394251
# MARKET MAKERS	-0.556392
# TRADES	0.0407796

I am surprised to see a positive coefficient on market capitalization, as I would expect this metric to be correlated with liquidity and thus lower the effective transaction costs. The coefficient on average shares per order is particularly interesting as it appears to indicate that securities that transact in larger order sizes on average have larger effective spreads per share. As expected, the market maker count is the strongest coefficient.

4.3 Performance

The training set mean squared error of the model is 0.572, measured in standard deviations squared due to the standardization process. This corresponds to a mean error of 0.0098, or about one percentage point. These metrics for the test set are 1.154 and 0.0138 respectively. Given that the mean of the percentage average spread is 0.0086, when scale is considered these errors are rather high. My initial model and choice of features do not appear to work very well for this data, but serve as a starting point for new ideas.

5 Conclusion and Next Steps

Thus far, I have made significant progress. The methodology and code for cleaning and consolidating the data have been completed. Adjustments can be made easily if I decide to cross section the data differently for future versions of the predictive model. Although the initial linear fit has been somewhat disappointing, I have several ideas to try out moving forward.

For the next version of the model, I will combine all the data from all 15 of the broker-dealers I have selected and separate it by month. For each month, I will include various security-specific volatility metrics in my feature space. Ideally, I want to use some form of a linear model, but I am also very curious to see how a decision tree based model would perform in the random forest configuration. To prevent overfitting, I will continue to use as much of the data as possible and cross-validate extensively. I expect the data set to grow far larger once I include 12 months of available data.

6 Bibliography

1. http://financialencyclopedia.net/exchanges/questions/what-is-the-difference-between-market-order-and-marketable-limit-order.html

2. http://www.crsp.com/products/documentation/data-definitions-n#nasdaq-market-makers-most-recent