Performing trade costs analysis with the `tradeCosts` package

Aaron Schwartz and Luyi Zhao

October 2, 2007

Abstract

For those who frequently engage in trading securities in financial markets, trade costs cannot be overlooked. Without an understanding of the costs incurred during trading returns can never reach their full potential. The `tradeCosts` package provides an easy-use to use set of tools for analyzing trade costs by generating automated summaries and PDF reports of trade costs from raw trading data. The summaries and reports generated allow a user to quickly understand how far their trades executed were from a user-specified benchmark price.

1 Introduction

For institutions and individuals that frequently trade securities in financial markets trade costs are an important expense to be considered. A money manager ignoring trade costs can hinder the performance of a portfolio by not understanding the costs of trades being made. We define trading costs in our package based on the work of Kissell and Glantz's *Optimal Trading Strategies* [1]. As described in their work, trade costs can be thought of to be composed of several distinct elements: commission, fees, spreads, price appreciation, market impact, timing risk, and opportunity cost. Currently, however, the package focuses solely on slippage, which we refer to in this release of the package as cost, as an ex-post measurement of the quality of executions relative to a benchmark price.

We define trade cost in its most basic form as:

$$\text{cost} = \text{side adjustment} * \text{execution quantity} * (\text{execution price} - \text{benchmark price}).$$

1

Since we wish for the market value of executed shares to be unsigned, we assume that execution quantity is always positive regardless of the trade's side. The side adjustment variable is either or 1 or -1 and serves to adjust the sign of cost depending on the side of the trade. In the `tradeCosts` package, positive cost is considered bad while negative cost is good (Example: a negative percent cost on a buy order indicates that the security was purchased for less than the benchmark price). Thus for buy and cover orders side adjustment is 1, and for sell and short orders side adjustment is just -1. Also, total percent costs over a certain time period or security are calculated as the market value weighted average of individual trades. In the following sections these formulas will be expanded to take corporate actions into account.

Having the ability to quickly and automatically generate reports that provide information about how far trades were being executed from a benchmark price for a certain group of trades can yield valuable information for those trying to improve their returns. These reports and summaries can help users gain a big picture view of the trade costs for a group of trades and discover specific trades, time periods, or specific securities with unusually high or low trade costs. The package operates through a user interface function, `trade.costs` which displays summaries of the input trade data and/or generates PDF summaries.

[1] In certain instances we may be concerned about the percent trade cost which is defined as the trade cost over a value such as the market value of executed shares

2 Calculating Cost: Examples

We now go through simple examples of calculating slippage, or as we refer to it in this release, cost, for individual trades, a period of time, and a security. Next, we examine example calculations for trades where corporate actions have occured.

2.1 Simple Trade

2.1.1 individual trade

First, we examine the process of calculating cost and percent cost for individual trades. Refer to the above table as we calculate the trade cost for June 22. Suppose for June, 22 2007 security IBM had a volume weighted average price (VWAP) of \$105.65 on that day, 100 shares of IBM were bought at the price of \$106.00. The price that we actually bought the securities at, \$106.00, is referred to as the execution price, and the VWAP will be the benchmark price for this case. The number of shares we bought, 100, is referred to as the execution quantity. Also, as noted in the previous section, the side adjustment for this order is 1 since this is a buy order. According to our formula introduced in the introduction, we then calculate cost as:

$$\begin{aligned}
\text{cost} &= \text{side adjustment} * \text{execution quantity} * (\text{execution price} - \text{benchmark price}) \\
tc &= i * x * (p - p_b) \\
tc &= 1 * 100 * (106.00 - 105.65) \\
tc &= \$35.00
\end{aligned}$$

Percent cost can then be easily calculated as:

$$\begin{aligned}
\%\ \text{cost} &= 100 * \frac{\text{cost}}{\text{market value of executed shares}} \\
tc_p &= 100 * \frac{tc}{m} \\
tc_p &= 100 * \frac{tc}{x * p} \\
tc_p &= 100 * \frac{35.00}{100 * 106.00} \\
tc_p &= 0.35\%
\end{aligned}$$

Notice that both percent cost and cost are positive which indicate that relative to the benchmark price, the trade described yielded a loss for whoever executed the trade. If IBM had been instead sold at \$106.00, then our calculations above would have yielded a cost of $-\$35.00$ and a percent cost of -0.35%. The negative percent cost and cost would have indicated that the transaction, relative to the VWAP benchmark price, was a good trade for whoever bought the 100 shares of IBM.

2.1.2 Security

In addition to calculating the cost of an individual trade we can also examine the total trading cost of a security. For example, suppose we have 100 shares of IBM stock, the details of which are summarized in the table below:

In order to calculate the total cost of the security IBM we simply sum up the costs of each individual trade relative to VWAP:

$$TC = \sum tc_j$$
$$TC = \sum (i_j * x_j * (p_j - p_{bj}))$$
$$TC = 100 * (104.44 - 105.65) + 500 * (106.00 - 105.11) + 100 * (103.00 - 104.44)$$
$$TC = \$180$$

To obtain the total percent cost we divide the total cost by the sum of all the total market values of the executions as shown below:

$$TC(\%) = 100 * \frac{TC}{M}$$
$$TC(\%) = 100 * \frac{\sum tc_j}{\sum m_j}$$
$$TC(\%) = 100 * \frac{180}{100 * 104.44 + 500 * 106.00 + 100 * 103.00}$$
$$TC(\%) = 100 * \frac{180}{73744}$$
$$TC(\%) = 0.14\%$$

2.1.3 Period

Similarly, the total cost and percent cost can be found for a certain period of time. Consider the table of trades below:

Period	Security Name	Side	Execution Price	Execution Quantity	VWAP
3/14/2007	IBM	buy	\$33.15	200	\$33.13
3/14/2007	CAKE	cover	\$31.71	500	\$32.00
3/14/2007	NST	sell	\$44.51	400	\$44.28

Now instead of summing over the trades for a security we group the trades by period, in this case 3/14/2007, and find the total cost of the trades for 3/14/2007:

$$TC = \sum (i_j * x_j * (p_j - p_{bj}))$$
$$TC = 1 * 200 * (33.15 - 33.13) + 1 * 500 * (31.71 - 32.00) + -1 * 400 * (44.51 - 44.28)$$
$$TC = -233$$

As before, dividing by the sum of the market values of the executed trades yields the percent cost:

$$TC(\%) = 100 * \frac{\sum tc_j}{\sum m_j}$$
$$TC(\%) = 100 * \frac{-233}{200 * 33.15 + 500 * 31.71 + 400 * 44.51}$$
$$TC(\%) = -0.006\%$$

2.2 Trading and Corporate Actions

Often, the calculation of costs becomes more involved when corporate actions occur. Corporate actions include splits, cash dividends, and stock dividends. Here we have several trading periods, with one trade "decision". Corporate actions occur inter-period.

Period	6/25/07		6/26/07		6/27/07
Decision Price	\$104		\$104		\$104
Close	\$104.44		\$52.55		\$42.67
VWAP	\$105.65		\$52.56		\$42.22
Execution Price	\$105.00		\$52.00		\$42.50
Order Qty.	100		100		200
Execution Qty.	100		100		100
Corporate Action		2 : 1 split		\$10.00 div.	

2.2.1 A trade with splits

In the above trading sequence, the example security split 2 for 1 sometime between 6/25 and 6/26. When calculating cost and percent cost relative to a benchmark price that is on the same *basis* as the execution price we use the same formulas as introduced in section 1. However, when a split occurs the benchmark price may not be on the same *basis* as the execution price. In that case, the formula for cost is adjusted. For example, for the above trade we would calculate the percent cost relative to the VWAP, which is on the same basis as the execution price:

$$
\begin{aligned}
tc(\%) &= 100 * \frac{tc}{m} \\
tc(\%) &= 100 * \frac{i * x * (p - p_b)}{x * p} \\
tc(\%) &= 100 * \frac{1 * 100 * (52.00 - 52.56)}{100 * 52.00} \\
tc(\%) &= -1.08\%
\end{aligned}
$$

However, the prior day's closing price is *not* on the same basis as the execution price because of the split. Therefore we calculate the cost of the same trade relative to the prior day's closing price while adjusting for the split with the following formula:

$$
\begin{aligned}
\text{cost} &= \text{side adjustment} * \text{execution quantity} * (\text{execution price} - \frac{\text{benchmark price}}{\text{split adjustment}}) \\
tc &= i * x * (p - \frac{p_b}{s})
\end{aligned}
$$

Thus, the percent cost of the above trade with the prior day's closing price as the benchmark price would be calculated as:

$$
\begin{aligned}
tc(\%) &= 100 * \frac{tc}{m} \\
tc(\%) &= 100 * \frac{i * x * (p - \frac{p_b}{s})}{x * p} \\
tc(\%) &= 100 * \frac{1 * 100 * (52.00 - \frac{104.44}{2})}{100 * 104.44} \\
tc(\%) &= .21\%
\end{aligned}
$$

2.2.2 A trade with a cash dividend

Cash dividends occur where cash is paid out to shareholders (on a per share basis) while the market capitalisation does not change. In this example, a $10 cash dividend is paid out between 6/26 and 6/27. The price per share instantly goes down by the amount of the dividend, and trade cost calculations must take this into account. Here the VWAP is on the same basis as the execution price, the percent trade cost for this trade when the benchmark price is the VWAP is calculated as:

$$
\begin{aligned}
tc(\%) &= 100 * \frac{tc}{m} \\
tc(\%) &= 100 * \frac{i * x * (p - p_b)}{x * p} \\
tc(\%) &= 100 * \frac{1 * 100 * (42.50 - 42.22)}{100 * 42.50} \\
tc(\%) &= 0.66\%
\end{aligned}
$$

However, the prior day's closing price is *not* on the same basis as the execution price because of the dividend. Therefore we calculate the cost of the trade relative to the prior day's closing price while adjusting for the dividend with the following formula:

$$
\begin{aligned}
\text{cost} &= \text{side adjustment} * \text{execution quantity} * (\text{execution price} - (\text{benchmark price} - \text{dividend})) \\
c &= i * x * (p - (p_b - d))
\end{aligned}
$$

Thus, the percent cost of the above trade with the prior day's closing price as the benchmark price would be calculated as:

$$
\begin{aligned}
tc(\%) &= 100 * \frac{tc}{m} \\
tc(\%) &= 100 * \frac{i * x * (p - (p_b - d))}{x * p} \\
tc(\%) &= 100 * \frac{1 * 100 * (42.50 - (52.55 - 10.00))}{100 * 42.50} \\
tc(\%) &= -.12\%
\end{aligned}
$$

2.2.3 A trade with a stock dividend

2.3 Opportunity Costs

Opportunity costs occur when a number of shares are not executed because of market prices or low market liquidity. Opportunity cost can be maodeled by the following equation:

$$
O.C. = (X - \sum x_j)(p_n - p_b)
$$

3 Trade Costs Analysis: An Example

Here we step through a trade costs analysis using the top-level function `trade.costs`. After understanding the required data, we explain the options available in `trade.costs`, call the function, and examine the summary output.

First, we will introduce the raw data sets included in this package: `trade`, a `data.frame` of trading data; `dynamic`, a `data.frame` of dynamic securities data; and `static`, a `data.frame` of static securities data. Function `trade.costs` takes in the trade, dynamic descriptive, and static descriptive data in separate `data.frames`. `trade` includes the IDs of the securities, the time period, side, execution quantity, the execution price:

```
> data(trade.mar.2007)
> head(trade.mar.2007, n = 1)

      period       id side exec.qty exec.price
1 2007-03-01 03818830    X    60600        1.6
```

`dynamic` is a `data.frame` that represents the dynamic descriptive data and includes the IDs of the securities, the time period, and a benchmark price:

```
> data(dynamic.mar.2007)
> head(trade.mar.2007, n = 1)

      period       id side exec.qty exec.price
1 2007-03-01 03818830    X    60600        1.6
```

Finally, we have the `static`, the static descriptive data, which has the IDs and symbols (an alternative identification from ID) of the securities:

```
> data(static.mar.2007)
> head(static.mar.2007, n = 1)

           id symbol     name sector
1301 00036020   AAON Aaon Inc    IND
```

Once the data is loaded with the correct columns, it is time to call `trade.costs` This user-level function returns either a text summary generated in the `R` environment or a PDF report. A `tradeCostsResults` object is also returned and is explained in detial in the package documentation files. Below we show the command to use `trade.costs` to generate a text summary in `R` of the trading, static descriptive, and dynamic descriptive data included in this package:

```
> trade.costs(trade.mar.2007, dynamic.mar.2007, static.mar.2007)
```

Notice that in addition to inputting the three data frames of raw data, the parameter `out.fmt = "text"` specifies that we want a text summary to be generated. The text output above generates five distinct tables in the summary output. As you can see above, the first table includes summary information of the trade costs analysis for the entire data set. The second, third, and fourth tables show the best and worst trades grouped by `id`, time period, and individual trades. The final table is a report on the `Ca's` found in the data that might be of interest to the user.

In addition, we can run the same function without inputting static descriptive data. This is useful in instances where alternative identifications such as ticker symbols are not needed or not readily available. We simply run the command without `static.mar.2007`:

```
> trade.costs(trade.mar.2007, dynamic.mar.2007)
```

As you can see above the same tables are generated with the exception that there is no field for symbol in the tables.

PDF reports are also easily generated using the `trade.costs` function. The reports contain the same tables and information as the text summaries in a convenient PDF format. The reports are available in both normal and verbose versions and can be generated by simply changing the `out.fmt` parameter. Also, reports can simply be displayed or displayed and saved to a file path specified by the parameter `pdf.file`. For example, the command:

```
trade.costs(trade.mar.2007, dynamic.mar.2007, static.mar.2007)
```

would generate and display a normal PDF report of the raw data included in the package while the following line of code:

```
trade.costs(trade.mar.2007, dynamic.mar.2007, static.mar.2007)
```

would generate and display a verbose version of the report *and* save the PDF report to the file path specified by `pdf.file`. Again, as with the text summaries, static descriptive data is still an optional parameter. The package documentation files will explain in more detail the usage of `trade.costs` and the requirements for the raw data input.

4 Conclusion

For those who engage in frequent transactions in financial markets, trade costs can play a pivotal role in portfolio performance. Institutions and individuals who wish to improve their returns can do so by understanding and carefully managing the costs incurred during trading. This package provides a set of tools for generating summaries and reports on a component of trade costs, slippage, from raw trading data.

Jeff Enos, David Kane, Aaron Schwartz, and Luyi Zhao
Kane Capital Management
Cambridge, Massachusetts, USA
`jeff@kanecap.com`, `david@kanecap.com`,
`aaron.j.schwartz@williams.edu` and `luyizhao@fas.harvard.edu`

References

[1] Robert Kissell and Morton Glantz. *Optimal Trading Strategies.* American Management Association, 2003.