GetHFData: A R package for downloading and aggregating high frequency trading data from Bovespa

(GetHFData: A R package for downloading and aggregating high frequency trading data from Bovespa)

Marcelo S. Perlin*
Henrique P. Ramos**

Resumo

This paper introduces GetHFData, a R package for downloading, importing and aggregating high frequency trading data from the Brazilian financial market. Based on a set of user choices, the package GetHFData will download the required files directly from Bovespa's ftp site and aggregate the financial data. The main objective of the publication of this software is to facilitate the computational effort related to research based on this large financial dataset and also to increase the reproducibility of studies by setting a replicable standard for data acquisition and processing. In this paper we present the available functions of the software, a brief description of the Brazilian market and several reproducible examples of usage..

***Palavras-chave*:** high frequency data, Bovespa, market microstructure, R, reproducible research
Códigos JEL: G18, G28, G32, G38.

Submetido em 25 de outobro de 2016. Reformulado em 21 de novembro de 2016. Aceito em 19 de dezembro de 2016. Publicado on-line em 27 de junho de 2016. O artigo foi avaliado segundo o processo de duplo anonimato além de ser avaliado pelo editor. Editor responsável: Márcio Poletti Laurini.

*Universidade Federal do Rio Grande do Sul, Porto Alegre/RS, Brasil. E-mail: marcelo.perlin@ufrgs.br

**Universidade Federal do Rio Grande do Sul, Porto Alegre/RS, Brasil. E-mail: hpramos4@gmail.com

Abstract

This paper introduces GetHFData, a R package for downloading, importing and aggregating high frequency trading data from the Brazilian financial market. Based on a set of user choices, the package GetHFData will download the required files directly from Bovespa's site and aggregate the financial data. The main objective of the publication of this software is to facilitate the computational effort related to research based on this large financial dataset and also to increase the reproducibility of studies by setting a replicable standard for data acquisition and processing. In this paper we present the available functions of the software, a brief description of the Brazilian market and several reproducible examples of usage.
***Keywords*:** high frequency data, Bovespa, market microstructure, R, reproducible research.

1. Introduction

Financial research has shift its profile over the years. Such a change is due to the increase of the frequency of financial data from which the researchers make their analysis. The recent regulatory and technological changes seen in the financial markets, along with the availability of trade and quote data and the increase of computer power to domestic users motivated scholars to study financial market dynamics in a finer scale, based on trading data in the high, tick by tick, frequency.

These studies are related to the topic of **market microstructure**, an upcoming field of expertise first established by the works of Kyle (1985), Glosten and Milgrom (1985) and Easley and O'hara (1987). These papers were novel at the time by showing that the price of an asset is related to existing frictions of the underlying market. The field of market microstructure specializes in studying the way that market frictions originating from the underlying structure can affect the price formation process of the traded assets. The main objective is to investigate and promote a well functioning market with high liquidity, low asymmetry of information and a better well being of market participants (De Jong and Rindi, 2009, Hasbrouck, 2007, Madhavan, 2000).

As mentioned before, empirical studies in the topic of market microstructure usually requires the analysis of high frequency trading data from the financial exchanges. These are related to large databases that are hard to store and process. While we can find pre-

vious work that discuss issues with this dataset such as Goodhart and O'Hara (1997) and Brownlees and Gallo (2006), they are mostly related to problems with the data itself such as effects of the market structure on the availability and interpretation of the data than the actual computational problems related to dealing with this large dataset in domestic computers.

The size of the dataset for a significant time window is a burden for an unexperienced researcher. As an example, the trading records for the date 2015-11-03 in Bovespa's equity market, a small market in international terms, is stored as a single compacted zip file with approximately 30 MB. When unpacked, the result is a text file with 310 MB of content. When more days are included, the size of the resulting data becomes troublesome, even for high end computers. For researchers with no background in programming, using this dataset requires a significant cost of time in order to learn how to store and process the related files in an efficient way. The complexity of this operation clearly creates a barrier that holds back the development of this research area in Brazil.

The objective of this paper is to introduce the functionalities of a software created to facilitate the importation and analysis of high frequency trading data from Bovespa. The program developed in the R platform is publicly available as a CRAN package. It allows the direct access to trade data for equity and derivative contracts from the exchange. The package contributes to the literature by facilitating the access to this particular set of data, decreasing the computational burden of users. The proposed package has the potential of setting a standard for accessing and manipulating this rich dataset by consolidating an accessible framework. The popularization of this software can increase the number and reproducibility of studies in this important area of finance.

The paper is organized as follows, first we make a brief review on the topic by discussing the main studies that have used high frequency data from Bovespa. We continue by presenting a brief introduction to the structure of the Brazilian financial market. Next, the format of the package and its main functionalities are presented. The work is followed by two empirical applications using the package. The paper finishes with the usual concluding remarks.

2. Literature Review

Previous studies using high frequency data from Bovespa approached different issues in the financial literature. The majority of these studies analyzed the volatility of the Brazilian stock market. Santos and Ziegelmann (2014) reproduced three different measures (realized variance, realized power variation and realized bipower variation) in a 15-minute interval sample ranging from 2004 to 2009. These measures were included as regressors in MIDAS (mixed data sampling) and HAR (heterogeneous autoregressive) models with the purpose of forecasting realized variance. The authors find that measures which are robust to jumps in asset prices such as realized power variation and realized bipower variation provided better forecasts for future volatility in terms of mean squared error. However, these predictions are not reported to be statistically different from those forecasts based on realized variance.

In a similar approach Araújo and Ávila (2015) estimated MIDAS and HAR models using high frequency data within a 5-minute interval. The total sample period starts in 2000 and ends in 2014. According to the authors, the HAR model resulted in better in-sample forecasts while a simple combination between the two models produced smaller values for both out-of-sample errors. Nonetheless, Junior and Pereira (2011) did not find statistical difference between the out-of-sample forecasts of the MIDAS and the HAR models.

Apart from these studies regarding the comparison of prediction performance of realized volatility estimators, one can find other studies using intraday trading data. Applications ranges from perceived volatility analysis during different intraday periods (Vicente et al., 2014), GARCH-family modeling and forecasting (Moreira and Lemgruber, 2004, Cappa and Pereira, 2009, Carvalho et al., 2006, Val et al., 2014, Garcia et al., 2014, Ceretta et al., 2011) to the classical problem of minimum variance portfolio selection (Borges et al., 2015).

Another strand of this literature is related to the analysis of trading strategies based on high frequency data. Fonseca et al. (2012) assessed abnormal returns resulting from different strategies implemented in the time period between 2006 and 2009. Considering the lead-lag effect between Ibovespa spot and future prices, the authors tested strategies built on time series forecasts estimated by ARIMA,

ARFIMA, VAR and VEC models. However, the results showed that a buy-and-hold strategy resulted in better performance than the other two market timing trading rules. The authors further split the sample in two subperiods, one before and one after the 2008 crisis and the results remained robust.

Pontuschka and Perlin (2015) tested the efficiency of a pairs trading strategy in a multi-frequency approach. Using data sampled from different frequencies (1,5,15,30, 60 minutes and daily data) in the 2008-2011 period, the authors provided evidence in favor of their hypothesis: higher sample frequency results in higher performance of the strategy and, therefore, higher evidence of market inefficiency. In a different approach, Jabbur et al. (2014) explored trading strategies based on technical analysis algorithms using five stocks during the month of October 2013. Other applications related to strategies includes the use of volatility estimators in a timing approach (Garcia et al., 2014) and macroeconomic variables (Garcia et al., 2016).

Although most studies using intraday data addressed risk and return topics, this type of data has also been used in the analysis of other subjects such as market liquidity and its commonalities (Victor et al., 2013, Silveira et al., 2014, Casarin, 2011, Marquezin and De Mattos, 2014, Perlin, 2013), bid-ask spreads/order book analysis (Cajueiro and Tabak, 2007, Maluf and Otiniano, 2014, Araújo et al., 2014), asymmetric information and corporate governance (Barbedo et al., 2007, Neto et al., 2012, Martins and Paulo, 2014), computation and algorithm programming (Silva et al., 2014, Araújo et al., 2015), high frequency data distribution (Horta and Ziegelmann, 2011, Cortines and Riera, 2007, Block et al., 2015) and other research topics in Finance (Taufemback and Da Silva, 2011, Caetano and Yoneyama, 2007, Biage et al., 2010, Perlin et al., 2014).

Table 1 summarizes recent high frequency data studies in Brazil. An arbitrary classification was made based on the main subject of the articles. According to the previous description, most of the work focused on volatility, returns and trading strategies. Another remark from Table 1 can be made about the sample period of these studies: some papers analyzed only one trading day (Caetano and Yoneyama, 2007, Maluf and Otiniano, 2014) while studies such as Araújo and Ávila (2015) used fourteen years of intraday data. This variability in sample size is expected, since different research objectives can demand

Table 1
List of published articles that have used high frequency trade data from Bovespa

Citation	Year	Time Period	Subject
Martins and Paulo (2014)	2014	2010 to 2011	Asymmetric info./Corp. Gov.
Neto et al. (2012)	2012	2009 to 2010	Asymmetric info./Corp. Gov.
Barbedo et al. (2007)	2007	2001 to 2006	Asymmetric info./Corp. Gov.
Araújo and Montini (2013)	2013	2007 to 2008	Bid-ask spreads/Order book
Cajueiro and Tabak (2007)	2007	1998 to 2003	Bid-ask spreads/Order book
Maluf and Otiniano (2014)	2014	Oct 10 2013	Bid-ask spreads/Order book
Silva et al. (2014)	2014	2013	Liquidity/Communality
Victor et al. (2013)	2013	2010 to 2012	Liquidity/Communality
Casarin (2011)	2011	2010	Liquidity/Communality
Perlin (2013)	2013	2005 to 2012	Liquidity/Communality
Martins and Paulo (2014)	2014	2010 to 2013	Liquidity/Communality
Araújo et al. (2015)	2015	Jan 1st 2013	Forecasting
Silva et al. (2014)	2014	Dec 2013	Forecasting
Cortines and Riera (2007)	2007	2002 to 2004	Data distribution
Horta and Ziegelmann (2011)	2011	2009 to 2010	Data distribution
Fonseca et al. (2012)	2012	2006 to 2009	Trading strategies
Jabbur et al. (2014)	2014	Oct 2008	Trading strategies
Garcia et al. (2016)	2016	2008 to 2011	Trading strategies
Pontuschka and Perlin (2015)	2015	2008 to 2011	Trading strategies
Garcia et al. (2014)	2014	2006 to 2011	Trading strategies
Val et al. (2014)	2014	2009 to 2012	Trading strategies
Vicente et al. (2014)	2014	2006 to 2009	Volatility
Carvalho et al. (2006)	2006	2001 to 2003	Volatility
Moreira and Lemgruber (2004)	2004	1998 to 2001	Volatility
Junior and Pereira (2011)	2011	2007 to 2010	Volatility
Borges et al. (2015)	2015	2009 to 2011	Volatility
Cappa and Pereira (2009)	2009	2005	Volatility
Ceretta et al. (2011)	2011	2009	Volatility
Araújo et al. (2015)	2015	2000 to 2014	Volatility
Santos and Ziegelmann (2014)	2014	2004 to 2009	Volatility
Biage et al. (2010)	2010	2007	Other
Perlin et al. (2014)	2014	2005 to 2011	Other
Taufemback and Da Silva (2011)	2011	2007 to 2008	Other
Caetano and Yoneyama (2007)	2007	Aug 23 2003	Other

more or less data. As for the case of the use of a small trading period, a possible explanation is the computational expertise needed in handling this type of database for larger periods. We also point out that most of the previous studies using high frequency data are related to the equity market. We have found a very low number of studies for derivative contracts such as options and futures. We hope that the use of GetHFData increases the number of studies for other markets.

3. The Brazilian financial market

Until 2008, the Brazilian financial market was concentrated in two main exchanges: Bovespa (SÃ£o Paulo Stock Exchange), which traded mainly equity contracts, and BM&F (Brazilian Mercantile and Futures Exchange), which negotiated commodities, futures and other derivatives. In 8 May 2008, Bovespa and BM&F merged on BM&FBovespa[1], creating one of the largest exchanges in Latin America in terms of market capitalization. According to BM&FBovespa's website, at the end of June/2016 there were 353 companies listed in the stock market. BMF&Bovespa's 2015 Annual Report shows an average daily trading value of R$ 6.7 bn on equities and equities derivatives. Most of the daily traded volume in these markets come from two types of investors: foreign (53%) and local institutional (27%). A small share of the market is held by retail investors (13%) and the remaining percentage refers to private companies and financial institutions.

The Brazilian market counts on 89 listed brokerage firms who are authorized to trade securities. These companies have direct access to the financial system as they are allowed to trade in behalf of their clients or on their own. Furthermore, autonomous agents may acquire customers in order to sell their financial products and advise about investment decisions. These agents use the brokerage firms' system to operate in the market. This environment and the whole market are regulated mostly by the CVM (ComissÃ£o de Valores MobiliÃ¡rios - Securities and Exchange Commission of Brazil) and the Central Bank of Brazil (BCB).

Both stock and stock options market functions in a hybrid protocol: companies may hire a market maker to increase trading activity

[1]Website: `www.bmfbovespa.com.br/en_us`, access in 2016-10-25.

and reduce price distortions due to the lack of liquidity. There are 232 assets traded using market makers in the equities market, including BDRs (Brazilian Depositary Receipts), ETFs, investment funds and stocks. The fixed income market has 7 instruments that use market makers, most of them are private bonds (debentures). These market makers compete in a continuous electronic auction in which investors may send limit and market orders. As usual, the limit orders that are not executed build the order book, defined firstly by price priority (the best deal is placed up front) and secondly by time (most recent orders are placed up front given a price level). Market makers are also allowed to trade in different stocks and a single stock can have several market makers up to a limit defined for each stock. This is a framework designed to increase competition and differs from the specialist structure found in the NYSE exchange, for example.

Since the merger of BM&F and Bovespa, there have been efforts to unify the trading systems in the Brazilian market. In 2013, BM&FBovespa started operations on the stock market using the PUMA system, aiming to transform other parallel trading platforms into a single one, aside from a reduced latency and increased processing time. Despite the unification, trading hours in BM&FBovespa varies among different markets. Table 2 exhibit times in the market for order cancellations, pre-opening, trading, closing calls and after-markets period[2]. Apart from different opening and closing times on the forward market, most of the markets works in similar periods. Both cash and odd lots markets allow for trading after market hours.

[2]All periods are shown in UTC-03:00 time zone.

Table 2
Equities market trading hours (UTC-03:00)

Market	Order cancel		Pre-opening		Trading		Closing Call		After-Market trading			
									Order cancel		Negociation	
	Start	End	Start	End	Start	End	Start	End	Start	End	Start	End
Cash Market	09:30	09:45	09:45	10:00	10:00	16:55	16:55	17:00	17:25	17:30	17:30	18:00
Odd lots market	09:30	09:45	09:45	10:00	10:00	16:55	16:55	17:00	17:25	17:30	17:30	18:00
Forward market	-	-	-	-	10:00	17:20	-	-	-	-	-	-
Options market	09:30	09:45	09:45	10:00	10:00	16:55	16:55	17:15	-	-	-	-
ETFs	09:30	09:45	09:45	10:00	10:00	16:55	16:55	17:15	-	-	-	-
OTC Market	09:30	09:45	09:45	10:00	10:00	16:55	16:55	17:00	-	-	-	-
Equity index options	09:30	09:45	09:45	10:00	10:00	16:50	16:50	17:15	-	-	-	-

3.1 Notation for tickers

Given the objective of the paper in proposing a software for financial data acquisition, it is advisable to better understand the format of tickers in the different markets traded on BM&FBovespa. In the equities market, every company has an unique four-letter ticker plus a number digit indicating whether the share is preferred (digit "4") or ordinary (digit "3"). As an example, the preferred PetrobrÃ¡s share will be identified by "PETR4"[3]. If the asset analyzed is also negotiated in the odd lots market, an additional "F" must be included to the four-letter ticker (e.g. PETR4F). The odd lots market allow for trades below the standard lot size (round lot) defined by the exchange.

Tickers for the stock options market are similar to the equities market. The four-letter ticker remains the same, in addition to the respective expiration month and strike price. A call option for a Vale do Rio Doce stock within a strike price of R$ 32.00 and expiration on October will be displayed as VALEJ32. Table 3 presents the expiration month codes for both call and put options. [4] It is noteworthy that the option ticker can also inform the options' style (American or European). If the option is European, there is an extra "E" in the end of the ticker (e.g. PETRI19E). Otherwise, it follows the American format. A list of available option contracts is published in Bovespa's website[5].

The futures market contracts are identified by a six-character ticker. The first three letters indicates the asset which is being traded, the fourth digit refers to the expiration month of the contract and the last two numbers points to the year of expiration. A *DOLX16* ticker accounts for a commercial dollar contract expiring on November 2016. A list of months and their respective codes are shown in Table 4.

[3]There are other number-digit identifiers such as digits five to eight. These numbers indicates special classes of preferred shares. For the sake of brevity, these types are not detailed here.

[4]Options expiration occurs in the third Monday of each month.

[5]`http://www.bmfbovespa.com.br/en_us/services/market-data/reports/equities/options/`, access in 2016-11-18.

Table 3
Expiration codes for options

Expiration month	Call	Put
January	A	M
February	B	N
March	C	O
April	D	P
May	E	Q
June	F	R
July	G	S
August	H	T
September	I	U
October	J	V
November	K	W
December	L	X

Table 4
Expiration month codes for futures

Expiration month	Code
January	F
February	G
March	H
April	J
May	K
June	M
July	N
August	Q
September	U
October	V
November	X
December	Z

4. The package GetHFData

The software **GetHFData** was written in R given the popularity of the programming platform for empirical research in finance and also the easiness of the public distribution of the package within the structure of CRAN (*The Comprehensive R Archive Network*). In this section we will describe the package and its functions. Actual R code is incorporated in the text, which facilitates the replication of the results described in the article. The scripts with the code for each empirical application are available in the personal webpage of the corresponding author[6]. Be advised that understanding these scripts will require some knowledge of R.

Before describing the package, we would like to express our gratitude towards Bovespa for allowing public access to its ftp site. This important gesture that will certainly motivate the growth in size and quality of studies related to the Brazilian financial market. It is important to point out that it would be impossible to create **GetHFData** without direct access to the high frequency database of Bovespa.

4.1 Installing and loading the package

Assuming an existing installation of R[7], a connection to the Internet and writing permission to the R folders in the operating system, the package **GetHFData** can be installed with the use of the function `install.packages` in the prompt:

```
> install.packages('GetHFData')
```

Once this procedure is complete, we can load the package with the command `library` and check the names of the enclosed functions with the command `ls`:

```
> library(GetHFData)
> ls("package:GetHFData")

[1] "ghfd_download_file"
[2] "ghfd_get_available_tickers_from_file"
[3] "ghfd_get_available_tickers_from_ftp"
```

[6] `https://sites.google.com/site/marceloperlin/`, access in 2016-11-18.
[7] `https://www.r-project.org/`, access in 2016-10-25.

```
[4] "ghfd_get_ftp_contents"
[5] "ghfd_get_HF_data"
[6] "ghfd_read_file"
```

As one can see, the package is compose of 6 functions with names that indicate their purpose. Based on this package the user can:

- Access the contents of the Bovespa ftp using function function *ghfd_get_ftp_contents*
- Get the list of available ticker in the trading data using *ghfd_get_available_tickers_from_ftp*
- Download individual files using *ghfd_download_file*
- Download and process a batch of dates and assets codes with *ghfd_get_HF_data*

5. Available functions in GetHFData

In this section we will describe the available R functions in detail.

ghfd_download_file Downloads a single file from Bovespa ftp (`ftp://ftp.bmf.com.br/MarketData`, access in 2016-10-25). This function will take as input a ftp address and the name of the downloaded file in the local disk.

Inputs	Default value	Description
my.ftp	-	A complete, including file name, ftp address to download the file from
out.file	-	Name of downloaded file with HFT data from Bovespa
dl.dir	"DL_Files"	The folder to download the zip files
max.dl.tries	10	Maximum number of attempts to download the files from ftp

Output: TRUE if successful, FALSE if not

Example of usage:

```
> fist.part <- 'ftp://ftp.bmf.com.br/'
> second.part <- 'MarketData/Bovespa-Opcoes/'
> ftp.address <- paste0(first.part,second.part)
```

```
> f.name <- 'NEG_OPCOES_20151229.zip'
> my.ftp <- paste0(ftp.address, f.name)
> out.file <- 'temp.zip'
> ghfd_download_file(my.ftp = my.ftp, out.file=out.file)
```

ghfd_get_available_tickers_from_file Function to get available tickers from downloaded zip file. This function will read the zip file downloaded from Bovespa and output a dataframe with all tickers found in the file, along with the number of trades for each.

Inputs	Default value	Description
out.file	-	Name of downloaded file with HFT data from Bovespa

Output: A dataframe with the ticker and number of trades for each asset found in file.

Example of usage:

```
> out.file <- system.file("extdata",
 'NEG_OPCOES_20151126.zip',
 package = "GetHFData")         ## local file from package
> tickers <- ghfd_get_available_tickers_from_file(out.file)
```

ghfd_get_available_tickers_from_ftp Function to get available tickers from ftp. This function will read the Bovespa ftp for a given market/date and output a dataframe with the tickers and number of trades for all assets found in the downloaded file.

Inputs	Default value	Description
my.date	"2015-11-03"	A single date to check tickers in ftp (e.g. "2015-11-03")
type.market	"equity"	The type of market to download data from ("equity","equity-odds", "options", "BMF")
dl.dir	"DL_Files"	The folder to download the zip files
max.dl.tries	10	Maximum attempts to download the files from ftp

Output: A dataframe with the tickers and number of trades found in file.

Example of usage:

```
> my.date <- '2015-11-03'
> my.tickers <-ghfd_get_available_tickers_from_ftp(my.date)
```

ghfd_get_ftp_contents Function to return the contents of Bovespa's ftp. This function will access the Bovespa ftp and return a vector with all files related to trades. All others files are ignored.

Inputs	Default value	Description
type.market	"equity"	The type of market to download data from ("equity" ,"equity-odds", "options", "BMF")
max.dl.tries	10	Maximum attempts to download the files from ftp

Output: A dataframe with the names of the files found in the ftp site.

Example of usage:

```
> ftp.files <- ghfd_get_ftp_contents(type.market = 'equity')
```

ghfd_get_HF_data Downloads and aggregates high frequency trading data directly from the Bovespa ftp. This function downloads zip files containing trades from Bovespa's ftp and imports it into R. If the file already exists in the user's computer, the function skips the download.

Inputs	Default value	Description
my.assets	NULL (downloads all available tickers)	The tickers (symbols) of the desired assets to import data (e.g. c("PETR4", "VALE5"))
type.market	"equity"	The name of the market to download the data from ("equity","equity-odds", "options", "BMF")
first.date	-	The first date of the imported data (Date class)
last.date	-	The last date of the imported data (Date class)
first.time	-	The first intraday period to import the data. All trades before this time of day are ignored. As a character object, e.g. "10:00:00".
last.time	-	The last intraday period to import the data. All trades after this time of day are ignored. As a character object, e.g. "18:00:00".
type.output	"agg"	Defines the type of output of the data. The choice "agg" outputs aggregated data for time intervals defined in agg.diff. The choice "raw" outputs the original, tick by tick, data from the zip files.
agg.diff	"15 mins"	The time interval used in the aggregation of data. Only used for type.output="agg". It should contain an integer followed by a time unit ("sec" or "secs", "min" or "mins", "hour" or "hours", "day" or "days"). Examples: agg.diff = "15 mins", agg.diff = "1 hour".
dl.dir	"DL_Files"	The folder to download the zip files
max.dl.tries	10	Maximum attempts to download the files from the ftp
clean.files	FALSE	Should the files be removed (deleted) after reading it? (TRUE or FALSE)

Output: A dataframe with the financial data with possible formats as raw (tick by tick) or aggregated with a time interval defined in argument `agg.diff`.

Example of usage:

```
> my.assets <- c('PETR4', 'VALE5')
> type.market <- 'equity'
> first.date <- as.Date('2015-12-29')
> last.date <- as.Date('2015-12-29')
> df.out <- ghfd_get_HF_data(my.assets,
```

```
type.market, first.date, last.date)
```

ghfd_read_file Reads a single zip file downloaded from the ftp site of Bovespa.

Inputs	Default value	Description
out.file	-	Name of zip/txt file to read data from
my.assets	NULL (imports all available tickers)	The tickers (symbols) of the desired assets to import data (e.g. c("PETR4", "VALE5"))
first.time	-	The first intraday period to import the data. All trades before this time of the day are ignored. As a character object, e.g. "10:00:00".
last.time	-	The last intraday period to import the data. All trades after this time of day are ignored. As character, e.g. "18:00:00".
type.output	"agg"	Defines the type of output of the data. The choice 'agg' outputs aggregated data for time intervals defined in agg.diff. The choice "raw" outputs the raw, tick by tick, data from the zip files.
agg.diff	"15 mins"	The time interval used in the aggregation of data. Only used for type.output='agg'. It should contain an integer followed by a time unit ("sec" or "secs", "min" or "mins", "hour" or "hours", "day" or "days"). Example: agg.diff = "15 mins", agg.diff = "1 hour".

Output: A dataframe with the financial data in the raw (tick by tick) or aggregated format.

Example of usage:

```
> my.assets <- c('ABEVA20', 'PETRL78')
> out.file <- system.file("extdata",
 'NEG_OPCOES_20151126.zip', package = "GetHFData")
> df.out <- ghfd_read_file(out.file, my.assets)
```

6. The resulting dataframe

The main function of the package is `ghfd_get_HF_data`, which allows the user to download high frequency trade data for a given time period and a set of assets defined by its tickers and type of market. Given the importance of this function, in this section we will

give more details regarding its output, which is possibly the most interesting object for the user.

The output of `ghfd_get_HF_data` is a dataframe, a R object that stores data in a tabular format. The contents of this output will change according to the type of data selected by the user, with the aggregation or not of the financial data. If the user decides to aggregate the data (`type.ouput = "agg"`), the raw (tick by tick) dataset is processed based in intervals defined by the input argument `agg.diff`. When choosing raw data, the function returns the raw dataset from the zip files.

6.1 Aggregated data

When `type.output="agg"`, the function `ghfd_get_HF_data` will output a dataframe with the columns described next. Notice that the name of the columns closely match the ones provided in the file `NEG_LAYOUT_english.txt`, which describes the dataset. This file is available in the ftp site. We keep this naming convention for consistency with the raw format. The columns are:

InstrumentSymbol The symbol of instruments (e.g. "PETR4")

SessionDate The day of the trading session (e.g. "2015-01-01")

TradeDateTime The date and time of the trading session (e.g. "2015-01-01 11:00:05")

n.trades The number of trades in each time interval (e.g. 100)

last.price The last price in the time interval (e.g. 7.8)

weighted.price The volume-weighted price for each time interval (e.g. 10)

period.ret The start-to-close arithmetic return in the time interval (e.g. 0.0001), as defined in next formula.

$$r_i = \frac{P_L}{P_F} - 1 \tag{1}$$

where:

P_F First price of time interval

P_L Last price of time interval

period.ret.volat The return volatility in the time interval, defined as:

$$R_{i,k} = \frac{P_{i,k}}{P_{i,k-1}} - 1 \quad (2)$$

$$\sigma_i = \sigma\left(R_{i,k}\right) \quad (3)$$

where:

$R_{i,k}$ — Vector of K returns ($k = 1..K$) within the time interval i

$P_{i,k}$ — Trade price at index k ($k = 1..K$), in the time interval i

sum.qtd Sum of traded contracts in each interval

sum.vol Sum of cash volume in each interval

n.buys Number of buyer initiated trades in the interval

n.sells Number of seller initiated trades in the interval

Tradetime Starting time of the period. As a character, e.g. "10:50:00"

6.2 Raw data

When using input `type.output="raw"`, `ghfd_get_HF_data` will output a dataframe with raw tick-by-tick transaction data. The columns for this choice are different than the columns for the choice of aggregate output. Notice, again, that the names of the columns match the ones provided in the file `NEG_LAYOUT_english.txt`. It is also important to point out that not all columns from the zip files are imported as some of them are redundant for research.

SessionDate	The day of the trading session (e.g. "2015-03-18")
InstrumentSymbol	The symbol/ticker of instr. (e.g."VALE5")
TradePrice	The price of the trade
TradedQuantity	The quantity of the trade
Tradetime	The time of the trade, including millisecond (e.g. "15:20:17.299")
TradeDateTime	The date and time of the trading session (e.g. "2015-03-18 15:20:17")
CrossTradeIndicator	Defines if the cross trade was intentional or not (1 - Intentional / 0 - Not Intentional)
BuyMember	Unique identifier for entering firm on the buy side of the trade
SellMember	Unique identifier for entering firm on the sell side of the trade
TradeSign	Identifies if a trades is a buyer initiated trade (+1) or a seller initiated trade (-1)

7. Empirical examples of usage

In this section we will provide two empirical examples of the usage of the package along with the R code required to replicate it. We separate the two examples by the format of the high frequency data used in the analysis, whether it is aggregated or raw. For the first, we selected a simple empirical problem from the literature, the impact of the time of the day in the liquidity. For the second we construct and analyze estimates of realized volatility for different assets. As mentioned before, both R scripts that implement this analysis are available in the corresponding author's webpage.

7.1 Liquidity and the time of the day

In order to illustrate the usage of the software with aggregated data, the chosen problem is the analysis of the intraday U shaped

pattern of liquidity in the equity market. This particular issue has been found and discussed in several papers from the literature such as Admati and Pfleiderer (1988), Back and Pedersen (1998) , Engle and Russell (1998), Groß-Klußmann and Hautsch (2011), among many others.

The data used in this empirical study is related to the six most traded assets in the period of fifteen trading days from 2016-09-12 until 2016-09-30. The use of a small time period is not accidental. We chose to keep fifteen days as it facilitates the replication of the example by decreasing the time needed to download the dataset by the user.

The first step is to select the liquid assets to run the empirical research. To do that, we select the six most traded assets in the last date of the study (2016-09-30) by checking the available tickers from the ftp site in this date. The following code executes this procedure.

```
> library(GetHFData)
> n.assets <- 6
> my.date <- as.Date('2016-09-30')
> type.market <- 'equity'
> df.tickers <- ghfd_get_available_tickers_from_ftp(my.date =
                    my.date, type.market = type.market)
```

As before, `ghfd_get_available_tickers_from_ftp` will output a vector with the number of trades for each ticker found in the dataset. As a robustness check, we can use package `ggplot2` (Wickham, 2009) to create a figure to illustrate the number of trades for each of the 25 most traded stocks in the date of 2016-09-30, as shown in Figure 1.

```
> library(ggplot2)
> temp.df <- df.tickers[1:25, ]
> p <- ggplot(temp.df, aes(x = reorder(tickers, -n.trades),
                           y = n.trades))
> p <- p + geom_bar(stat = "identity")
> p <- p + theme(axis.text.x=element_text(angle=90,
                         hjust=1,vjust=0.5))
> p <- p + labs(x = 'Tickers', y = 'Number of trades')
> print(p)
```

From Figure 1 we can see that the six most traded assets in 2016-09-30 are ITSA4, PETR4, ITUB4, BBDC4, ABEV3, BBSE3. A particular feature of the high frequency data from Brazil is that the

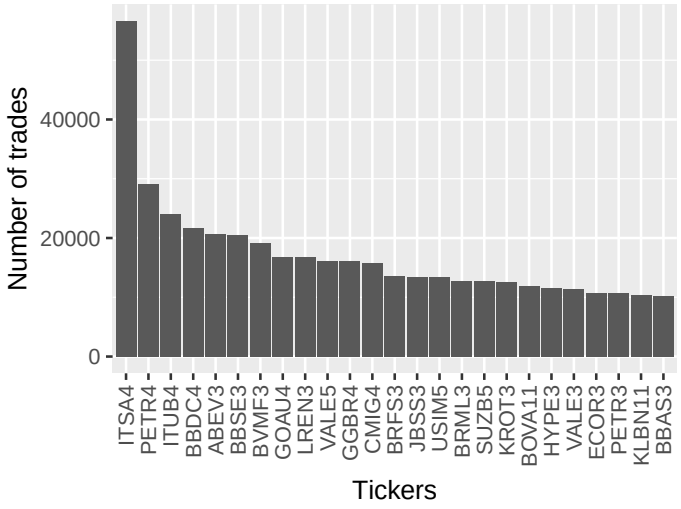

Figure 1
Most traded assets in 2016-09-30

liquidity is disperse and decreases rapidly across the assets, as we can see from Figure 1. Even though we are only looking at trading data for one day, we can expect that the number of trades will also drop quickly in other time periods.

From the programming side, the dataframe `df.tickers` is already sorted by the number of trades so, in order to select the six most traded assets, we select the first six elements of `df.tickers$tickers`.

```
> my.assets <- df.tickers$tickers[1:n.assets]
```

And now we can print it to check its content:

```
> print(my.assets)
```

```
[1] "ITSA4" "PETR4" "ITUB4" "BBDC4" "ABEV3" "BBSE3"
```

We continue the empirical example by using the package **GetHFData** to download and aggregate the desired information for later analysis. The first step in this stage is to set the options for downloading the dataset. Notice that it is good policy to set the object `my.folder` as the name of a folder in the computer's hard disk where

the user has writing permission in order to download the files. We set an example path as *"PATH TO YOUR FOLDER HERE"*. We make it clear that the user has to modify this object in order for the code to run without error.[8]

As for the intraday time periods, we use a first time of *10:30:00* and last as *16:30:00* in order to avoid the trading noise from the opening and closing of the market, which could bias our results. The options used with **GetHFData** are set as follows.

```
> my.folder<-'PATH TO YOUR FOLDER HERE'
> setwd(my.folder)
> first.time <- '10:30:00'
> last.time <- '16:30:00'
> first.date <- as.Date('2016-09-12')
> last.date <- as.Date('2016-09-30')
> type.output <- 'agg'
> agg.diff <- '15 min'
> my.assets <- c("ITSA4", "PETR4", "ITUB4", "BBDC4",
                 "ABEV3", "BBSE3")
> type.market <- 'equity'
```

After setting the inputs, we now use function `ghfd_get_HF_data` to download and aggregate the financial data.

```
> df.out <- ghfd_get_HF_data(my.assets = my.assets,
                             type.market = type.market,
                             first.date = first.date,
                             last.date = last.date,
                             first.time = first.time,
                             last.time = last.time,
                             type.output = type.output,
                             agg.diff = agg.diff)
```

We point out that the previous code will take some time to finish as it has to download and read several large files from Bovespa ftp site. Once it is finished, we can check the output of `ghfd_get_HF_data` by calling function `str` for the object `df.out`, which will show the textual representation of the object in the R environment.

```
> str(df.out)
```

[8]Users in the Windows platform should be aware that the folder path has to set using forward slashes (/) and not backslashes, which is the default.

```
'data.frame':        2160 obs. of  13 variables:
 $ InstrumentSymbol: chr  "ABEV3" "ABEV3" "ABEV3" "ABEV3" ...
 $ SessionDate     : Date, format: "2016-09-12" ...
 $ TradeDateTime   : POSIXct, format: "2016-09-12 10:30:00" ...
 $ n.trades        : int  531 1143 441 1168 603 618 617 512 492 ...
 $ last.price      : num  19.8 19.7 19.8 19.7 19.8 ...
 $ weighted.price  : num  19.7 19.7 19.7 19.7 19.8 ...
 $ period.ret      : num  -0.000506 -0.001013 0.001014 -0.002532 ...
 $ period.ret.volat: num  0.000388 0.000175 0.000261 0.000319 ...
 $ sum.qtd         : num  314000 584500 1210500 281800 189100 ...
 $ sum.vol         : num  6200565 11522865 23873576 5553286  ...
 $ n.buys          : int  304 170 199 405 230 283 181 273 348 170 ...
 $ n.sells         : int  227 973 242 763 373 335 436 239 144 209 ...
 $ Tradetime       : chr  "10:30:00" "10:45:00" "11:00:00" ...
```

As described earlier, the object returned from `ghfd_get_HF_data` is a dataframe with several columns calculated from the raw data. Notice that the columns already have the correct class, which facilitates the future manipulation of the data.

Once the data is available, we proceed to the analysis of the intraday pattern of liquidity. To do so, we use the number of trades as a proxy for liquidity. The analysis will be based on the visual examination of a figure that relates the distribution of number of trades to the time of the day. Since the number of trades are not comparable across assets, we plot the same figure for different stocks. Next, we show the R code that creates the figure based on the `ggplot2` package.

```
> p <- ggplot(df.out, aes(x =  Tradetime, y = n.trades))
> p <- p + geom_boxplot() + coord_cartesian(ylim = c(0, 3000))
> p <- p  + theme(axis.text.x=element_text(angle=90,hjust=1,
                                vjust=0.5))
> p <- p + facet_wrap(~InstrumentSymbol)#, scales = 'free')
> p <- p + labs(y='Number of Trades', x = 'Time of Day')
> print(p)
```

In Figure 2 we show the number of trades as a function of the time of the day. As expected, we find that the intraday shape of liquidity follows a U pattern, that is, the number of trades rises in the beginning and ending of the day, with the smallest value around 13:15:00. Such a pattern is found for the great majority of the assets.

This result is supported by previous findings in the literature (Engle and Russell, 1998, Groß-Klußmann and Hautsch, 2011). In the

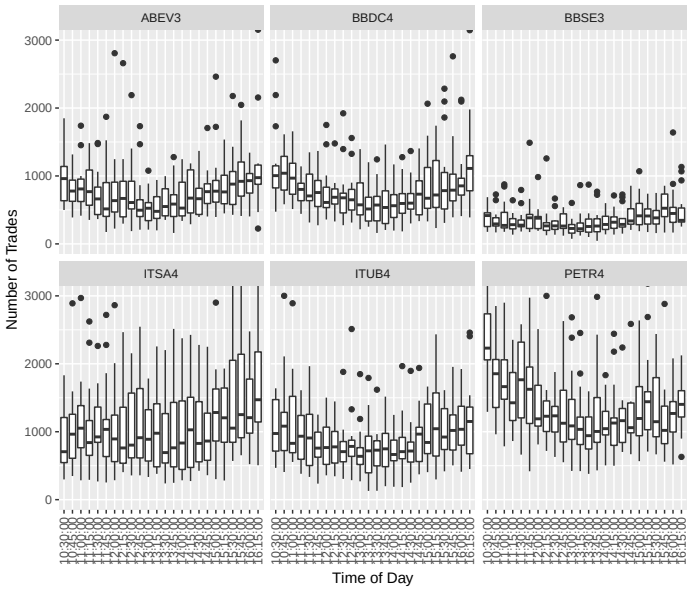

Figure 2
Number of trades and the time of the day

beginning of the trading day, a significant volume of overnight information is priced at the market, which justifies the increase of the number of trades. As for the end of the day, the higher volume of trades can be explained as a inventory strategy by the investors or market makers, which aims to finish the day with null portfolio positions in order to avoid the overnight risk. Since the decrease of portfolio size is achieved with more trades, we see a significant increase of negotiations at the end of the day. Interestingly, this pattern for liquidity is correlated to a pattern of intraday volatility (Andersen and Bollerslev, 1997).

7.2 Calculating realized volatility from tick by tick data

One of the main innovations in the research field of pricing uncertainty is the possibility of estimating ex-post measures of volatility based on high frequency data, the so called realized volatility (Andersen et al., 2003, Barndorff-Nielsen and Shephard, 2002). Empirical studies have showed that these new estimators are more accurate than traditional measures calculated from datasets of lower frequencies such as daily returns (Andersen et al., 2003, Fleming et al., 2003, Barndorff-Nielsen and Shephard, 2002).

In this section we will present a simple example of calculating daily realized volatility from tick by tick data imported using **GetHFData**. In order to do so, we will use a popular R package designed to perform common operations in high frequency financial data, package *highfrequency* (Boudt et al., 2014). This is a very useful package for a researcher in market microstructure, providing functions for the organization and manipulation of trade and quote data. It also includes several functions for the calculation of realized volatility measures, among other features. Further details about this package can be found in its main website[9].

The first step in this empirical section is to import the raw dataset. We will use a block of code similar to the previous example, however, the time period will be increased to approximately five months and we set the option `type.output` as `raw`. Next, we present the actual R code that downloads the dataset:

```
> my.folder<-'PATH TO YOUR FOLDER HERE'
```

[9] `http://highfrequency.herokuapp.com/`, access in 2016-11-18.

```
> setwd(my.folder)
> first.time <- '10:30:00'
> last.time <- '16:30:00'
> first.date <- as.Date("2016-05-25")
> last.date <- as.Date("2016-09-30")
> type.output <- 'raw'
> my.assets <- c("ITSA4", "PETR4", "ITUB4", "BBDC4", "ABEV3", "BBSE3")
> type.market <- 'equity'
> df.out <- ghfd_get_HF_data(my.assets = my.assets,
                             type.market = type.market,
                             first.date = first.date,
                             last.date = last.date,
                             first.time = first.time,
                             last.time = last.time,
                             type.output = type.output)
```

The previous code should run in any computer with internet access by only changing the value of object `my.folder`. Notice, however, that since it downloads 5 months of high frequency data, it can take a significant amount of processing time and memory from the computer.

In this example of calculating realized volatility, we will use the simplest case available in package *highfrequency*, function `medRV`. From it own help manual:

> *"The medRV belongs to the class of realized volatility measures in this package that use the series of high-frequency returns $r_{t,i}$ of a day t to produce an ex post estimate of the realized volatility of that day t. medRV is designed to be robust to price jumps. The difference between RV and medRV is an estimate of the realized jump variability. Disentangling the continuous and jump components in RV can lead to more precise volatility forecasts, as shown in Andersen et al. (2007) and Corsi et al. (2010)."*

Further inspection in the usage of `medRV` shows that it was designed to work with `xts` objects (Ryan and Ulrich, 2014). These are a special type of dataframe for time series data. Also, the function `medRV` only works in a stock-by-stock case. These elements in the

usage of `medRV` requires some adaptation since the dataframe output from **GetHFData** is not a `xts` object and it will include several stocks. Gladly, a simple solution to the difference of format is to build a wrapper function around `medRV` and use the capabilities of package *dplyr* (Wickham and Francois, 2016) to calculate the realized volatility measure for all stocks, in all days.

The wrapper function works with the following steps: it takes as input a trade price vector and its related date-time, change the input data to a `xts` object and, finally, use the new format with function `medRV` in order to calculate the realized volatility for the given inputs. Next, we present the actual R code that registers the wrapper function.

```
> my.RV.fct <- function(TradePrice, TradeDateTime){
         require(highfrequency)

         temp.x <- xts(TradePrice, order.by = TradeDateTime)
         RV <- medRV(temp.x, makeReturns = T)

         return(as.numeric(RV))
 }
```

Once the function is available, we use it together with the manipulation capabilities of package *dplyr* in order to calculate the daily realized volatility for all assets in our dataset.

```
> library(dplyr)
> RV.tab <- df.out %>%
     group_by(InstrumentSymbol, SessionDate) %>%
     summarise(RV = my.RV.fct(TradePrice, TradeDateTime))
```

The result of the previous code is a dataframe with three columns, the asset code (ticker), the date and the realized volatility:

```
> print(head(RV.tab))

Source: local data frame [6 x 3]
Groups: InstrumentSymbol [1]

  InstrumentSymbol SessionDate           RV
```

```
             <chr>      <date>        <dbl>
1            ABEV3  2016-05-25 0.0024318127
2            ABEV3  2016-05-27 0.0013886317
3            ABEV3  2016-05-30 0.0002977746
4            ABEV3  2016-05-31 0.0029552299
5            ABEV3  2016-06-01 0.0015516901
6            ABEV3  2016-06-02 0.0011200550
```

Once the processed data is ready, we illustrate the dynamics of the realized volatility for each asset in Figure 3, which was generated with the following R code:

```
> p <- ggplot(RV.tab, aes(x=SessionDate, y=RV))
> p <- p + geom_line(size=1)
> p <- p + facet_wrap(~InstrumentSymbol, scales = 'free')
> p <- p + labs(x='Date', y='Realized Volatility')
> print(p)
```

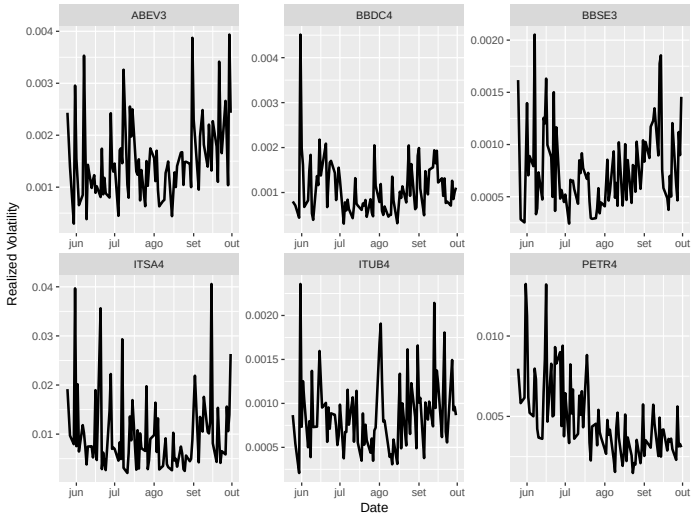

Figure 3
Realized volatility for six stocks

From Figure 3 we see that, as expected, the realized volatility presents the clustering effect, where high/low values of volatility are

followed by another high/low value. The previous example shows how to import raw trade data using **GetHFData** and how easy it is to analyze the resulting dataset with package *highfrequency*. By combining both packages, a researcher has a significant amount of computational tools at its disposal.

8. Conclusions

In this paper we present **GetHFData**, a R package publicly distributed in CRAN that facilitates access to high frequency trading data from Bovespa, the Brazilian financial exchange. The use of this program makes it easy for users to download and aggregate high frequency trade data for two different financial markets in Brazil, equity and derivatives. All data is obtained from the public available ftp site of Bovespa. In the document we present the available functions of the package along with a description of their inputs and examples of usage. We also present two empirical examples regarding the usage of the package, the first using aggregated data to show the intraday pattern of liquidity, and the second uses tick-by-tick (raw) trade data to calculate measures of realized volatility for several stocks.

The provided software decreases the computational cost of researchers interested in the area of market microstructure in Brazil, which may boost the number of studies for this topic in the future. By removing the computation burden related to dealing with this dataset, it will be easier for experienced researchers and students to access and analyze high frequency data from Bovespa. Another contribution of the package is setting a standard for manipulating high frequency data from Brazil, contributing to the reproducibility of research in the field.

References

Admati, A. R. and Pfleiderer, P. (1988). A theory of intraday patterns: Volume and price variability. *Review of Financial studies*, 1(1):3–40.

Andersen, T. G. and Bollerslev, T. (1997). Intraday periodicity and volatility persistence in financial markets. *Journal of empirical finance*, 4(2):115–158.

Andersen, T. G., Bollerslev, T., and Diebold, F. X. (2007). Roughing it up: Including jump components in the measurement, modeling, and forecasting of return volatility. *The review of economics and statistics*, 89(4):701–720.

Andersen, T. G., Bollerslev, T., Diebold, F. X., and Labys, P. (2003). Modeling and forecasting realized volatility. *Econometrica*, 71(2):579–625.

Araújo, A. and Montini, A. (2013). High frequency trading - abordagem clássica para análise de preço-volume em uma nova microestrutura de mercado. *Anais dos Seminários em Administração-SEMEAD*.

Araújo, A. C. d. and Ávila, A. M. (2015). Estimação da volatilidade percebida futura por meio de combinação de projeções. *XV Encontro Brasileiro de Finanças*.

Araújo, G. S., Barbedo, C. H. d. S., and Vicente, J. V. M. (2014). The adverse selection cost component of the spread of brazilian stocks. *Emerging Markets Review*, 21:21–41.

Araújo, R. d. A., Oliveira, A. L., and Meira, S. (2015). A hybrid model for high-frequency stock market forecasting. *Expert Systems with Applications*, 42(8):4081–4096.

Back, K. and Pedersen, H. (1998). Long-lived information and intraday patterns. *Journal of financial markets*, 1(3):385–402.

Barbedo, C. H., Camilo-Da-Silva, E., and Leal, R. P. (2007). Probability of information-based trading, intraday liquidity and corporate governance in the brazilian stock market. *Working paper*.

Barndorff-Nielsen, O. E. and Shephard (2002). Econometric analysis of realized volatility and its use in estimating stochastic volatility models. *Journal of the Royal Statistical Society Series B*, 64(2):253–280.

Biage, M., Costa, N., and Goulart, M. (2010). O efeito dia de vencimento no mercado de opções da bovespa revisitado. *Revista Economia*, 11(1):53–96.

Block, A. S., Ceretta, P. S., and Costa, A. (2015). Linear and non-linear association measures with intraday/high frequency data for all ibovespa stocks. *Revista Eletrônica de Estratégia & Negócios*, 8(3):171–186.

Borges, B., Caldeira, J., and Ziegelmann, F. (2015). Selection of minimum variance portfolio using intraday data: An empirical comparison among different realized measures for bm&fbovespa data. *Brazilian Review of Econometrics*, 35(1):23–46.

Boudt, K., Cornelissen, J., Payseur, S., Nguyen, G., and Schermer, M. (2014). *highfrequency: Tools For Highfrequency Data Analysis*. R package version 0.4.

Brownlees, C. T. and Gallo, G. M. (2006). Financial econometric analysis at ultra-high frequency: Data handling concerns. *Computational Statistics & Data Analysis*, 51(4):2232–2245.

Caetano, M. A. L. and Yoneyama, T. (2007). Characterizing abrupt changes in the stock prices using a wavelet decomposition method. *Physica A: Statistical Mechanics and its Applications*, 383(2):519–526.

Cajueiro, D. O. and Tabak, B. M. (2007). Characterizing bid–ask prices in the brazilian equity market. *Physica A: Statistical Mechanics and its Applications*, 373:627–633.

Cappa, L. and Pereira, P. (2009). Modeling the volatility of petrobras returns using high frequency data. *31 Meeting of the Brazilian Econometric Society*.

Carvalho, M., Freire, M. A., Medeiros, M. C., and Souza, L. R. (2006). Modeling and forecasting the volatility of brazilian asset returns - a realized variance approach. *Revista Brasileira de Finanças*, 4(1):321–343.

Casarin, F. (2011). Comunalidade na liquidez : evidências no mercado de capitais brasileiro. *Master's thesis. Postgraduate Program in Administration - Federal University of Santa Maria*.

Ceretta, P. S., de Barba, F. G., Vieira, K. M., and Casarin, F. (2011). Intraday volatility forecasting: Analysis of alternative distributions. *Revista Brasileira de Finanças*, 9(2):209–227.

Corsi, F., Pirino, D., and Reno, R. (2010). Threshold bipower variation and the impact of jumps on volatility forecasting. *Journal of Econometrics*, 159(2):276–288.

Cortines, A. and Riera, R. (2007). Non-extensive behavior of a stock market index at microscopic time scales. *Physica A: Statistical Mechanics and its Applications*, 377(1):181–192.

De Jong, F. and Rindi, B. (2009). *The microstructure of financial markets*. Cambridge University Press.

Easley, D. and O'hara, M. (1987). Price, trade size, and information in securities markets. *Journal of Financial economics*, 19(1):69–90.

Engle, R. F. and Russell, J. R. (1998). Autoregressive conditional duration: a new model for irregularly spaced transaction data. *Econometrica*, pages 1127–1162.

Fleming, J., Kirby, C., and Ostdiek, B. (2003). The economic value of volatility timing using realized volatility. *Journal of Financial Economics*, 67(3):473–509.

Fonseca, N. F., Lamounier, W. M., and Bressan, A. A. (2012). Abnormal returns in the ibovespa using models for high-frequency data. *Revista Brasileira de Finanças*, 10(2):243–265.

Garcia, M., Santos, F., and Medeiros, M. (2016). The high frequency impact of macroeconomic announcements in the brazilian futures markets. *Brazilian Review of Econometrics, Forthcoming*.

Garcia, M. G. P., Medeiros, M. C., de Luna, F. E., and Santos, A. (2014). Economic gains of realized volatility in the brazilian stock market. *Revista Brasileira de Finanças*, 12(3):319.

Glosten, L. R. and Milgrom, P. R. (1985). Bid, ask and transaction prices in a specialist market with heterogeneously informed traders. *Journal of financial economics*, 14(1):71–100.

Goodhart, C. A. and O'Hara, M. (1997). High frequency data in financial markets: Issues and applications. *Journal of Empirical Finance*, 4(2):73–114.

Groß-Klußmann, A. and Hautsch, N. (2011). When machines read the news: Using automated text analytics to quantify high frequency news-implied market reactions. *Journal of Empirical Finance*, 18(2):321–340.

Hasbrouck, J. (2007). *Empirical market microstructure*. Oxford University Press New York.

Horta, E. d. O. and Ziegelmann, F. A. (2011). Dynamics of financial returns densities: A functional approach applied to the bovespa intraday index. *XI Encontro Brasileiro de Finanças*.

Jabbur, E., Silva, E., Castilho, D., Pereira, A., and Brandão, H. (2014). Design and evaluation of automatic agents for stock market intraday trading. *Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)-Volume 03*, pages 396–403.

Junior, M. W. and Pereira, P. V. (2011). Modeling and forecasting of realized volatility - evidence from brazil. *Brazilian Review of Econometrics*, 31(2):315–337.

Kyle, A. S. (1985). Continuous auctions and insider trading. *Econometrica: Journal of the Econometric Society*, pages 1315–1335.

Madhavan, A. (2000). Market microstructure: A survey. *Journal of financial markets*, 3(3):205–258.

Maluf, Y. S. and Otiniano, C. E. G. (2014). Modelagem das chegadas de ordens de oferta via processo de hawkes. *XIV Encontro Brasileiro de Finanças*.

Marquezin, C. L. and De Mattos, L. B. (2014). Liquidity cost of future contract to bm&fbovespa fat cattle. *Revista de Administração Mackenzie*, 15(4).

Martins, O. S. and Paulo, E. (2014). Information asymmetry in stock trading, economic and financial characteristics and corporate governance in the brazilian stock market. *Revista Contabilidade & Finanças*, 25(64):33–45.

Moreira, J. M. d. S. and Lemgruber, E. F. (2004). O uso de dados de alta freqüência na estimação da volatilidade e do valor em risco para o ibovespa. *Revista Brasileira de Economia*, 58(1):100–120.

Neto, J. C. d. C. O., de Medeiros, O. R., and de Queiroz, T. B. (2012). Corporate governance and information incorporation speed: Lead-lag between the igc and ibrx. *Revista Brasileira de Finanças*, 10(1):149.

Perlin, M. (2013). Os efeitos da introdução de agentes de liquidez no mercado acionário brasileiro (the effects of the introduction of market makers in the brazilian equity market). *Revista Brasileira de Finanças*, 11(2):281.

Perlin, M., Brooks, C., and Dufour, A. (2014). On the performance of the tick test. *The Quarterly Review of Economics and Finance*, 54(1):42–50.

Pontuschka, M. and Perlin, M. (2015). Pairs trading in the brazilian stock market: the impact of data frequency. *Revista de Administração Mackenzie*, 16(2):188.

Ryan, J. A. and Ulrich, J. M. (2014). *xts: eXtensible Time Series*. R package version 0.9-7.

Santos, D. G. and Ziegelmann, F. A. (2014). Volatility forecasting via midas, har and their combination - an empirical comparative study for ibovespa. *Journal of Forecasting*, 33(4):284–299.

Silva, E., Castilho, D., Pereira, A., and Brandao, H. (2014). A neural network based approach to support the market making strategies in high-frequency trading. *2014 International Joint Conference on Neural Networks (IJCNN)*, pages 845–852.

Silveira, V. G. d., Vieira, K. M., and da, A. S. (2014). Comunalidade na liquidez: Um estudo intraday para as ações do índice bovespa. *Estudos do CEPE*, 1(39):139–156.

Taufemback, C. and Da Silva, S. (2011). Spectral analysis informs the proper frequency in the sampling of financial time series data. *Physica A: Statistical Mechanics and its Applications*, 390(11):2067–2073.

Val, F. d. F., Pinto, A. C. F., and Klotzle, M. C. (2014). Volatility and return forecasting with high-frequency and garch models- evidence for the brazilian market. *Revista Contabilidade & Finanças*, 25(65):189–201.

Vicente, J. V. M., Araújo, G. S., Castro, P. B. F. d., and Tavares, F. N. (2014). Assessing day-to-day volatility: Does the trading time matter? *Revista Brasileira de Finanças*, 12(1):41–66.

Victor, F. G., Perlin, M. S., and Mastella, M. (2013). Comunalidades na liquidez: evidências e comportamento intradiário para o mercado brasileiro. *Revista brasileira de finanças. Rio de Janeiro, RJ. Vol. 11, n. 3 (set. 2013), p. 375-398.*

Wickham, H. (2009). *ggplot2: Elegant Graphics for Data Analysis.* Springer-Verlag New York.

Wickham, H. and Francois, R. (2016). *dplyr: A Grammar of Data Manipulation.* R package version 0.5.0.