Untitled

Alexis Solis

9/25/2020

1. Introduction & Importing Data

We'll work with intraday data for the *S&P/BMV IPC Equity Index*. The data consists of `n = 2,133,890` observations and `k = 23` variables. The time-series is composed of prices and trades per minute, spanning from the beginning of 1996 through the first half of 2018.

```r
# Read the data
IPC <- arrow::read_parquet(file = here("01-Data", "parquet", "raw_MEXICO_IPC.parquet"))
```

First thing we do is take a look at the columns and data types that we have:

```
## Rows: 2,133,890
## Columns: 23
## $ `#RIC`                  <chr> ".MXX", ".MXX", ".MXX", ".MXX", ".MXX", ".M...
## $ `Date[L]`               <dbl> 19960102, 19960102, 19960102, 19960102, 199...
## $ `Time[L]`               <time> 08:36:00, 08:38:00, 08:39:00, 08:40:00, 08...
## $ Type                    <chr> "Intraday 1Min", "Intraday 1Min", "Intraday...
## $ Open                    <dbl> 2777.47, 2777.47, 2777.47, 2777.47, 2777.47...
## $ High                    <dbl> 2777.47, 2777.47, 2777.47, 2777.47, 2777.47...
## $ Low                     <dbl> 2777.47, 2777.47, 2777.47, 2777.47, 2777.14...
## $ Last                    <dbl> 2777.47, 2777.47, 2777.47, 2777.47, 2777.14...
## $ Volume                  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ `Ave. Price`            <dbl> 2777.470, 2777.470, 2777.470, 2777.470, 277...
## $ VWAP                    <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ `No. Trades`            <dbl> 1, 2, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 2, 3...
## $ `Correction Qualifiers` <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ `Open Bid`              <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ `High Bid`              <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ `Low Bid`               <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ `Close Bid`             <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ `No. Bids`              <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ `Open Ask`              <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ `High Ask`              <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ `Low Ask`               <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ `Close Ask`             <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...
## $ `No. Asks`              <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
```

We can count how many `NA`s are present in our data. We do this per column:

```
## Rows: 1
## Columns: 23
## $ ticker                <int> 0
## $ raw_date              <int> 0
## $ raw_time              <int> 0
## $ type                  <int> 0
## $ open                  <int> 0
## $ high                  <int> 0
## $ low                   <int> 0
## $ last                  <int> 0
## $ volume                <int> 0
## $ average_price         <int> 0
## $ vwap                  <int> 2133890
## $ no_trades             <int> 0
## $ correction_qualifiers <int> 2133890
## $ open_bid              <int> 2133890
## $ high_bid              <int> 2133890
## $ low_bid               <int> 2133890
## $ close_bid             <int> 2133890
## $ no_bids               <int> 0
## $ open_ask              <int> 2133890
## $ high_ask              <int> 2133890
## $ low_ask               <int> 2133890
## $ close_ask             <int> 2133890
## $ no_ask                <int> 0
```

We see that there are 10 columns (variables) that have all values as `NA`. We assign these variables to the `columns_to_remove` object and remove them from the data.

```
##  [1] "vwap"                  "correction_qualifiers" "open_bid"
##  [4] "high_bid"              "low_bid"               "close_bid"
##  [7] "open_ask"              "high_ask"              "low_ask"
## [10] "close_ask"
```

We name the *clean* dataset as `IPC_ip` (IPC intraday prices) and again see the column names and each data type.

```
## Rows: 2,133,890
## Columns: 13
## $ ticker        <chr> ".MXX", ".MXX", ".MXX", ".MXX", ".MXX", ".MXX", ".MXX...
## $ raw_date      <dbl> 19960102, 19960102, 19960102, 19960102, 19960102, 199...
## $ raw_time      <time> 08:36:00, 08:38:00, 08:39:00, 08:40:00, 08:41:00, 08...
## $ type          <chr> "Intraday 1Min", "Intraday 1Min", "Intraday 1Min", "I...
## $ open          <dbl> 2777.47, 2777.47, 2777.47, 2777.47, 2777.47, 2777.14,...
## $ high          <dbl> 2777.47, 2777.47, 2777.47, 2777.47, 2777.47, 2777.14,...
## $ low           <dbl> 2777.47, 2777.47, 2777.47, 2777.47, 2777.14, 2777.14,...
## $ last          <dbl> 2777.47, 2777.47, 2777.47, 2777.47, 2777.14, 2777.14,...
## $ volume        <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ average_price <dbl> 2777.470, 2777.470, 2777.470, 2777.470, 2777.360, 277...
## $ no_trades     <dbl> 1, 2, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 2, 3, 3, 3, 3,...
## $ no_bids       <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ no_ask        <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
```

2. Feature Engineering & Data Wrangling

We then carry on with the analysis by creating some new variables (a.k.a. *Feature Engineering*) and manipulating the data.

First, we create a `tidy_date` variable where we store the date according to the *ISO 8601* standard that states that dates should be expressed in the `YYYY-MM-DD` format. In consequence, the `raw_date` column is dropped and we keep the newly created `tidy_date` variable instead.

Also, we drop the `ticker`, `type`, `open`, `high`, and `low` columns because we think they are of no use for the analysis.

We store the modified data into the `IPC_tbl` (IPC tibble) object.

Time-Series Data Print

Next, we create some time-related variables, such as:

`tidy_year`: a `dbl` that stores the year (from 1996 - 2018).

`tidy_month`: a `dbl` that stores the month as a number (from 1 through 12).

`tidy_mday`: a `dbl` that stores the day number within each month (from 1 through 31).

`tidy_wday`: a categorical variable (`fctr`) that includes: `Mon Tue Wed Thu Fri`.

`tidy_hour`: a `dbl` that stores the hour (we have data from 5 through 20 hours).

`tidy_minute`: a `dbl` that stores the minute of the trade (from 0 through 59).

`tidy_time`: an `hms` (hour-minute-second) object that stores the time of the trade.

Table 1: Data summary

Name	Piped data
Number of rows	2133890
Number of columns	15
Column type frequency:	
Date	1
difftime	1
factor	1
numeric	12
Group variables	None

Variable type: Date

skim_variable	n_missing	complete_rate	min	max	median	n_unique
tidy_date	0	1	1996-01-02	2018-06-05	2007-07-25	5613

Variable type: difftime

skim_variable	n_missing	complete_rate	min	max	median	n_unique
tidy_time	0	1	19920 secs	73320 secs	42300 secs	647

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
tidy_wday	0	1	TRUE	5	Wed: 435537, Tue: 434397, Thu: 426174, Fri: 425693

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
trade_id	0	1	1066945.50	616001.13	1.00	533473.25	1066945.50	1600417.75	2.133890e+06	
last	0	1	23945.92	16340.42	2731.21	6502.76	24198.56	40435.79	7.813872e+04	
volume	0	1	55341105.47	147693040.82	0.00	11697950.00	42651727.00	83705131.75	1.759968e+09	
average_price	0	1	23945.91	16340.43	2731.36	6502.69	24199.01	40435.50	7.813872e+04	
no_trades	0	1	34.98	39.95	1.00	8.00	21.00	53.00	3.693000e+03	
no_bids	0	1	0.00	0.00	0.00	0.00	0.00	0.00	0.000000e+00	
no_ask	0	1	0.00	0.00	0.00	0.00	0.00	0.00	0.000000e+00	
tidy_year	0	1	2007.00	6.39	1996.00	2002.00	2007.00	2013.00	2.018000e+03	
tidy_month	0	1	6.43	3.43	1.00	3.00	6.00	9.00	1.200000e+01	
tidy_mday	0	1	15.84	8.75	1.00	8.00	16.00	23.00	3.100000e+01	
tidy_hour	0	1	11.24	1.92	5.00	10.00	11.00	13.00	2.000000e+01	
tidy_minute	0	1	30.52	17.33	0.00	16.00	31.00	46.00	5.900000e+01	

2.1 Computing Intraday Log-Returns

Next, we compute the intraday returns and assign them to the `log_ret` variable. We also convert our data into a time-friendly type of object called `tibble time` (we do this via the `as_tbl_time()` function).

```
## # A time tibble: 2,133,890 x 16
## # Index: tidy_date
##    trade_id tidy_date   log_ret tidy_time  last volume average_price no_trades
##       <int> <date>        <dbl> <time>    <dbl>  <dbl>         <dbl>     <dbl>
##  1        1 1996-01-02 NA       08:36     2777.      0         2777.         1
##  2        2 1996-01-02  0.      08:38     2777.      0         2777.         2
##  3        3 1996-01-02  0.      08:39     2777.      0         2777.         3
##  4        4 1996-01-02  0.      08:40     2777.      0         2777.         3
##  5        5 1996-01-02 -1.19e-4 08:41     2777.      0         2777.         3
##  6        6 1996-01-02  0.      08:42     2777.      0         2777.         3
##  7        7 1996-01-02  0.      08:43     2777.      0         2777.         3
##  8        8 1996-01-02  0.      08:44     2777.      0         2777.         3
##  9        9 1996-01-02  0.      08:45     2777.      0         2777.         3
## 10       10 1996-01-02  0.      08:46     2777.      0         2777.         3
## # ... with 2,133,880 more rows, and 8 more variables: no_bids <dbl>,
## #   no_ask <dbl>, tidy_year <dbl>, tidy_month <dbl>, tidy_mday <int>,
## #   tidy_wday <ord>, tidy_hour <int>, tidy_minute <int>
```