

**Problem Set #2 : Time-Dependent Active List**

WARNING: This problem set is very long. You have one week to do it. It is due on Tuesday the 22nd of October 2019 before class (8:30am). If you start two days before you'll never make it. You should start today.

1) Download from the CCLE intranet the variable `allstocks` that represents the solution to Problem Set 1. It contains the universe of stocks we will study. Do not take the version of `allstocks` that you produced: everybody must start from the same universe. Let  $n$  denote the number of stocks in this universe.

2) Download the following data from DataStream for all the stocks in the universe and for all the days from January 1<sup>st</sup>, 1997 to December 31<sup>st</sup>, 2002. Let  $T$  denote the number of weekdays in this time range. Every variable is a large matrix of dimension  $(T \times n)$ :

- a) `price`: unadjusted share price in local currency,
- b) `tri`: total return index in Euros,
- c) `volume`: daily volume expressed in Euros,
- d) `mtbv`: Market-to-Book value of equity (even if it changes less frequently, you still need to have a value for it every day so the matrix has the same size as the other matrices),
- e) `cap`: total market capitalization in Euros (DataStream datatype: MV, but needs to be converted into Euros).

As a by-product, you will also obtain from DataStream a list of all the weekdays from January 1<sup>st</sup>, 1997 to December 31<sup>st</sup>, 2002. Transform it into the format dd-mmm-yyyy and store it in a variable called `myday`.

3) Download the database `TickSummary.mat`. It contains median monthly bid-ask spreads for a large list of stocks that has a substantial overlap with the universe in `allstocks`. Use it to build a bid-ask spread matrix for our universe. Fill in the missing data with NaN (Matlab value for Not-a-Number). Note that the third dimension of the `spread` matrix in `Ticksummary.mat` corresponds to the 7 hours of the trading day (ordered from morning to afternoon). As we simulate trading daily VWAP, you must take the average bid-ask spread across all 7 sessions. The unit for the spread number is a fraction of the stock price; the exact formula is:  $(\text{ask} - \text{bid}) / ((\text{ask} + \text{bid}) / 2)$ . The output from Step 3) is a matrix called `tcost` of dimension  $(T \times n)$  which contains the transaction cost for every stock on every day, using the formula given in class. The entries of this matrix will remain constant throughout the month, and only change at the turn of the month. Given that the bid-ask spread data I gave you only starts from November 1999, you must backfill all previous months with the first available bid-ask spreads, assuming that bid-ask spreads were constant through time up to November 1999. The bias that this induces is minimal and can be ignored.

4) Using the IBES tickers associated with each DataStream DScode, pull down the history of analyst recommendation revisions for every stock. There is a way to obtain this from our WRDS subscription. Construct a matrix of dimension  $(T \times n)$  named `rec` that contains the total number of analyst upgrades minus the total number of analyst downgrades for any given stock on any given day. Most of the entries of this matrix will be zero.

5) Construct a boolean matrix `isactivenow` of dimension  $(T \times n)$  that contains the time-dependent active list. The entry is equal to 1 if the stock is active that day, and 0 if it is inactive. The criteria to be applied are as follows:

- a) The stock must have valid price and total return index for at least 90% of the past 252 business days
- b) The stock must not be dead. A dead stock is a stock whose unadjusted share price in local currency stays the same for 10 days or more.
- c) The stock must have valid volume and total return index for at least 90% of the past 21 business days.
- d) The stock must have valid book value of equity and valid total market cap today.
- e) The stock must have valid bid-ask spread last month.
- f) The stock must have had at least one analyst recommendation revision in the past.

6) Further refine the matrix `isactivenow` by applying two liquidity filters. For a stock to be active, *both* liquidity filters must be satisfied. Do this stock by stock.

- a) At the beginning of each month, a stock passes the first liquidity threshold if:
  - it was active the previous month and its bid-ask spread is less than 1%
  - or it was inactive the previous month and its bid-ask spread is less than 0.80%
- b) At the beginning of each month, a stock passes the second liquidity threshold if:
  - it was active the previous month and its average daily volume over the past month was more than 1 million euros per day
  - or it was inactive the previous month and its average daily volume over the past month was more than 1.2 million euros per day.

**Problem Set output:** You must post on CCLE a Matlab database containing the variables:

1. `allstocks`  $(1 \times n)$
2. `myday`  $(T \times 1)$
3. `price`  $(T \times n)$
4. `tri`  $(T \times n)$
5. `volume`  $(T \times n)$
6. `mtbv`  $(T \times n)$
7. `rec`  $(T \times n)$
8. `tcost`  $(T \times n)$
9. `isactivenow`  $(T \times n)$

**Grading:** You will be judged by how complete and correct these variables are, especially `isactivenow`. To get `isactivenow` right you must get all the others right.

**VERY IMPORTANT:** Due to the limited time that the Teaching Assistant can allocate, every student is allowed at most one e-mail question to her – so use it wisely! In addition, he will hold one session which will be devoted to this problem set.