Tracking High Frequency Cointegrations in the US Stock Market

A Filtering Approach

Nodari Alessandro
Proserpio Lorenzo

November 3, 2022

The definition

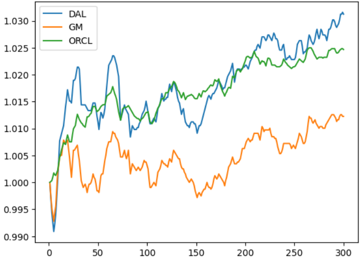
- N non-stationary stochastic processes $\{X_t^j\}_{t\geq 0}^{j=1,\dots,N}$ are said to be cointegrated if there exist N real coefficients $\{\alpha^j\}_{j=1,\dots,N}$ such that

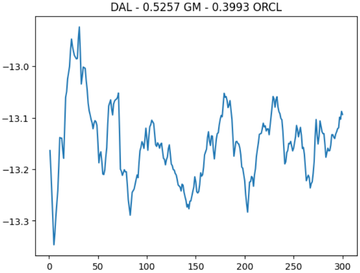
$$\alpha^1 X^1 + \alpha^2 X^2 + \dots + \alpha^N X^N$$

is a stationary stochastic process.

Example

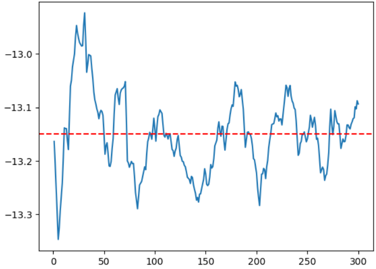
- Cointegration among prices of Oracle Corporation (ORCL), Delta Air Lines (DAL) and General Motors (GM):





The Motivation

- Cointegrations among securities' prices can be useful in quantitative finance because are often strong indicators in the markets, moreover they can be profitably traded using a strategy called "Statistical Arbitrage".
- We short the cointegration whenever it is "sufficiently" above the long-term mean and we buy the cointegration whenever it is "sufficiently" below the long-term mean. Whenever it crosses the mean we will close every open positions.



The Problem

- The coefficients for which multiple assets are cointegrated are usually not independent from time. Too rapid changes can ruin our strategy and lead to huge losses.
- Moreover, cointegrations can also die, in the sense that two assets which are cointegrated until a certain point in time are not necessarily cointegrated forever.
- We need an online tool to keep track of the coefficients (which are not directly observable from the market) for risk-management purposes.

The Dataset

- We obtained from OpenBB the prices of all the listed stocks in the US market on the day 2022-10-12 with 1-min candles.
- We selected only the tickers which have not many values missing in their time-series and took only the adjusted closing value for each candle.
- We propagated forward the last prices where there are candles with missing values and then we lagged the entire dataframe of 1 min.
- The tickers used in this work are: 'D', 'F', 'ABT', 'BA', 'BABA', 'O', 'BHP', 'CCL', 'CVS', 'CVX', 'DAL', 'DASH', 'DIS', 'DUK', 'DVN', 'EQT', 'GM', 'IBM', 'LVS', 'MDT', 'NIO', 'ORCL', 'PFE', 'RBLX', 'RCL', 'SCHW', 'SE', 'TJX', 'TSM', 'VALE', 'WMT', 'XPEV'.

The Tool

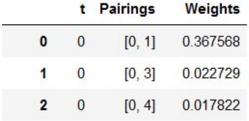
- We implemented an extension of an algorithm that is already known in the engineering field and that is traditionally used for multiple objects tracking (like for example tracking airplanes with radar) which is called Rao-Blackwellized Particle Filter (RBPF).
- RBPF incorporates the Rao–Blackwell theorem to improve the sampling done in a particle filter by marginalizing out some variables. For more information see reference [3].

The Measurements

- We used the Johansen-cointegration trace test minute by minute to obtain the coefficients for which pairs of stocks are cointegrated with confidence 95%. For more information see references [1]-[2].
- The whole process can be done in an online fashion while receiving new data, however, to ease up the implementation, we decided to prepare beforehand our dataset with all the measurements. Notice that in this procedure there is no leakage of future information.
- Due to low computational resources we decided to stick with only cointegrated pairs, however the algorithm can manage cointegrations among up to 12 terms.

- The first 40 candles of each stock are used as a training set.
- We procedeed in the following manner:
    1. we looked for all the possible pairs of stocks at time t;
    2. tested their price series from time 0 to time t with the Johansen test;
    3. if they were not cointegrated we store a zero, if they are we store only the ratio between the two coefficients for which they are cointegrated (in practice we normalized with respect to the first coefficient);
    4. repeat until the end.

- The measurements are stored in a Pandas dataframe in the following fashion (remember that $t = 0$ for us is $t = 40$ because of our training set):



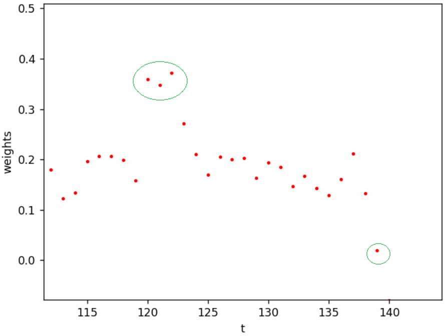
	t	Pairings	Weights
0	0	[0, 1]	0.367568
1	0	[0, 3]	0.022729
2	0	[0, 4]	0.017822

- In the end we trimmed out all the pairs that are always zero across all the timesteps. This is not necessary, but it would have been useless to store and process that.

Clutters

- Engineers usually intend *clutters* as false measurements. In our case *clutters* are not due to failure in the measurements, but are extreme movements in our coefficients due to temporary anomalies in the dynamics of prices before reverting to their previous state.
- In our opinion (but it requires further investigation), they can be attributed to really small periods with a lot of volume or to the execution of large orders.

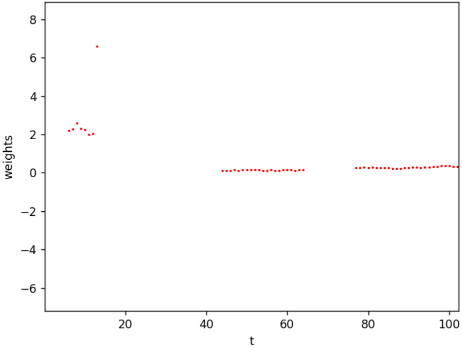
- To illustrate the point this is the time series of the measurements (cointegration coefficient) for one pair of equities, the points in the green circles are clearly *clutters*. Cointegration CVS-TSM:



- To determine if a point is a clutter we use the following rule:
  1. we use the Kalman Filter of each particle to determine if the point is between two standard deviation from the actual state mean estimation (using as standard deviation the square root of the estimated state variance);
  2. if the majority of the particles classifies it as clutter we classify it as clutter.
- In the implementation *Majority* is a parameter that can be tuned and correspond to the number of particles that should classify it as clutter in order to be considered as clutter.
- After classifying as clutter *NC* consecutive measurements (where *NC* is a tunable parameter), we kill all the particles and initialize new ones.

Births & Deaths

- We use the cointegration ABT-BA to illustrate that cointegrations can vanish and reappear. We will refer to these events as births and deaths.



- Since we tested the cointegration with 95% confidence we decide to model the births and deaths as Bernoullians in the following way:

$$\begin{cases} p(\text{birth}|\text{measurement} \neq 0, \text{dead at previous step}) \sim Be(0.95) \\ p(\text{birth}|\text{measurement} = 0, \text{dead at previous step}) = 0 \\ p(\text{death}|\text{measurement} = 0, \text{alive at previous step}) \sim Be(0.95) \\ p(\text{death}|\text{measurement} \neq 0, \text{alive at previous step}) = 0 \end{cases}$$

The Model

- Denote with $\mathbf{x}_t$ the vector of the "true" coefficients (states) at time t and $\mathbf{y}_t$ the vector of the coefficients returned by the Johansen test. We use a really simple model for our dynamics:

$$\begin{cases} \mathbf{x}_{t+1} = \mathbf{x}_t \\ \mathbf{y}_{t+1} = \mathbf{x}_t + \mathcal{N}(0, Q_t) \end{cases}$$

- The matrix Q_t is a tunable parameter and can change with time. The sense for this model is that we obtained the coefficients through the Johansen test, which is based on the observed prices and these are generally noisy. Moreover, we are interested only in "persistent" cointegrations in which the coefficient doesn't change much.

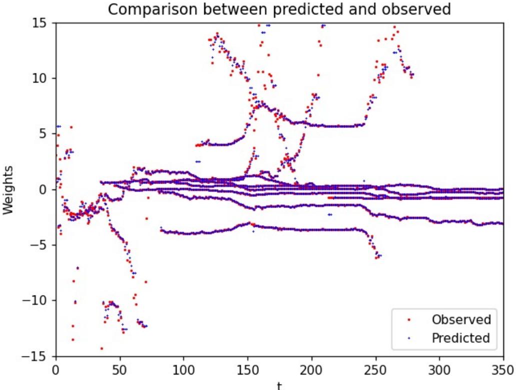
- Notice that if Q_t is diagonal this system can be split in indipendent 1-D state spaces, as we do in our implementation.
- Thanks to the nice form of the model we use a standard Kalman Filter to do the exact step (Rao-Blackwellization) in our particle filter.
- To know exactly how our particles are made please look into the code under the section *Particle Class*.

The Pseudo-Algorithms

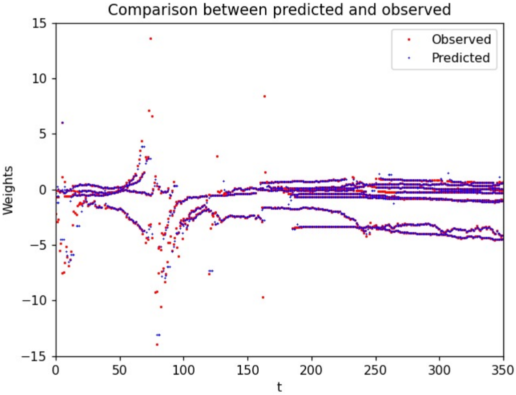
- For the Pseudo-Algorithms please refer to the code or to the *Pseudo_Algorithms.pdf*.

Results

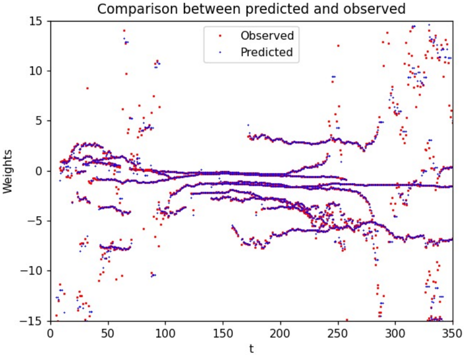
Here there is a selection of our results (if you want to see all just look into the code). There are 10 different cointegrations per plot. Here: ABT-XPEV, BA-BABA, BA-CVS, BA-TSM, BABA-LVS, BABA-NIO, BABA-TJX, BABA-TSM, BABA-VALE, BABA-XPEV.



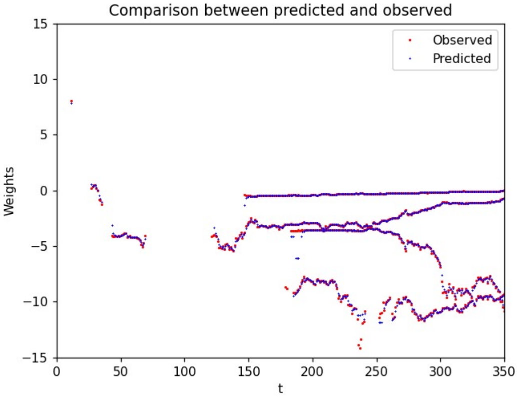
Here: O-BHP, O-DAL, O-ORCL, O-VALE, DAL-DIS, DAL-LVS, DAL-NIO, DAL-SCHW, DAL-VALE, DAL-XPEV.



Here: DIS-LVS, DIS-ORCL, DIS-SCHW, DIS-VALE, LVS-ORCL, LVS-SCHW, LVS-VALE, NIO-TJX, ORCL-TSM, ORCL-VALE.

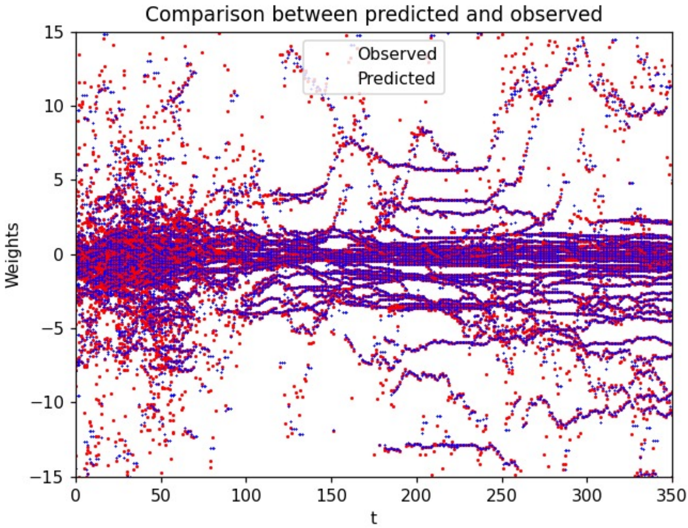


Here: SCHW-VALE, TJX-TSM, TJX-XPEV, TSM-VALE.



Conclusions

- The results are promising, moreover the computational time is acceptable (under 3 minutes to process all the day, 2 candles per second).
- The state-space model can be improved, maybe choosing in a more clever way the error in the measurements.
- It seems that the market ”gains” structure while the time is passing. The beginning of the day seems more noisy than the end of the day. This may need further investigation. To illustrate this point we put the plot of all the stocks and our predictions in the next page.


Comparison between predicted and observed
Observed
Predicted
Weights
t
15
10
5
0
−5
−10
−15
0
50
100
150
200
250
300
350

References

1. Soren Johamen, Katarina Jtiselius. *Maximum Likelihood Estimation and Inference on Cointegration - with applications to the demand for money*;
2. Lütkepohl, H. 2005. *New Introduction to Multiple Time Series Analysis*. Springer;
3. Simo Särkkä, Aki Vehtari, Jouko Lampinen. *Rao-Blackwellized Particle Filter for Multiple Target Tracking*;
4. `www.statsmodels.org/dev/generated/statsmodels.tsa.vector_ar.vecm.coint_johansen.html`;
5. `https://openbb-finance.github.io/OpenBBTerminal`;
6. `https://filterpy.readthedocs.io/en/latest/monte_carlo/resampling.html`;
7. `https://pykalman.github.io`.