Mining latent relationships in stocks daily movements

Data Science for Networks: project report

Giuseppe Attanasio
S265910

February 24, 2019

1 Introduction

Stock trading is an interesting applicative field where a significant number of machine learning and pattern recognition solutions are being designed to support traders decisions. Financial markets have become a breeding ground both for the availability of large and heterogeneous data sets and for the presence of several underlying economical rules that drive the market that can be mined.

This work aimed to analyze relationships between price movements of different stocks in order to find similar *behaviors*. The study focused on unsupervised modeling of stock prices variations by means of a multi-way representation. A tensor decomposition algorithm was applied to spot relevant information.

1.1 Modeling price variations

The strategy for data modeling is a crucial and discriminant factor in this type of analysis. Since a tensor decomposition algorithm is used, the number and type of dimensions as well as the content of the cells has to be chosen. The solution proposed is based on a graph-based representation built with the following steps.

1. Since price data comes with a daily frequency, it is possible to compute the Rate Of Change (ROC) of closing prices with respect to the previous day:

$$ROC_t = 100 \cdot \frac{C_t - C_{t-1}}{C_{t-1}} \tag{1}$$

2. Then, each day is labeled using its ROC value and a threshold tr:

$$L_t = \begin{cases} UP, & \text{if } ROC_t > tr \\ MID, & \text{if } -tr \leq ROC_t \leq tr \\ DOWN, & \text{if } ROC_t < -tr \end{cases}$$

3. Once days are labeled for each stock, build a graph $G = (V, E)$ where each vertex represents a stock and edges are created, for a given day t, if both the stocks had a positive variation:

$$\forall i, j : L_{t,i} = UP \wedge L_{t,j} = UP \Rightarrow \exists e(i, j) \tag{2}$$

Given this representation, each day, price relationships that were raising up can be depicted as a undirected graph, synthesized by a symmetric adjacency matrix with binary values in cells. Furthermore, the graph is not static. It changes day-by-day accordingly to updated information. Consequently, all the information may be encoded in a third-order tensor having the stocks on two axis and the time along the third:

$$X \in \mathbb{N}^{SxSxT}$$

Any slice of the tensor at time t represents the adjacency matrix for the day t. Additionally, the tensor is partially symmetric because all its slices along t axis are symmetric.

1.2 Tensor decomposition

In this work the CANDECOMP/PARAFAC (CP) [1] [2] tensor decomposition was used to spot latent factors that generated the tensor - the reader here should read: *the factors characterized by certain stocks that have grown together across the time.* There exist many matrix decomposition techniques used in collaborative filtering and recommendation tasks. CP factorization can be seen as a higher-order decomposition with respect to, for example, Singular Value Decomposition. The factorization generates an approximation of the original third-order tensor, expressed as a sum of R rank-one tensors.

$$\hat{X} = \sum_{r=1}^{R} a_r \circ b_r \circ c_r$$

while the single cell:

$$\hat{x}_{ijk} = \sum_{r=1}^{R} a_{ir} b_{jr} c_{kr}$$

The latent factors can be spotted on components generated by the decomposition. As pointed out by Papalexakis [3], each component highlights a soft co-clustering of the tensor, with the high values of

$$a_r, b_r, c_r$$

as the membership values to co-clusters.

2 Case study: S&P500 stocks in 2017

Even though there exist many stock market indices, the analysis is narrowed to Standard and Poor 500 index. It gathers the top 500 influential companies in United States of America. The temporal window analyzed is the solar year 2017. The shape of the third-order tensor would be in principle $500x500x365$. Considering only those stocks whose Global Industry Classification Standard (GICS) sector is known and the business days when the market was open, the input tensor become:

$$X \in \mathbb{N}^{496x496x250}$$

In order to encode only strong relationships, a threshold value of 2% was used. In financial market context this value is way higher than the common variation range [-1%, 1%] that stock prices unlikely cross.

2.1 Sparsity and connected components

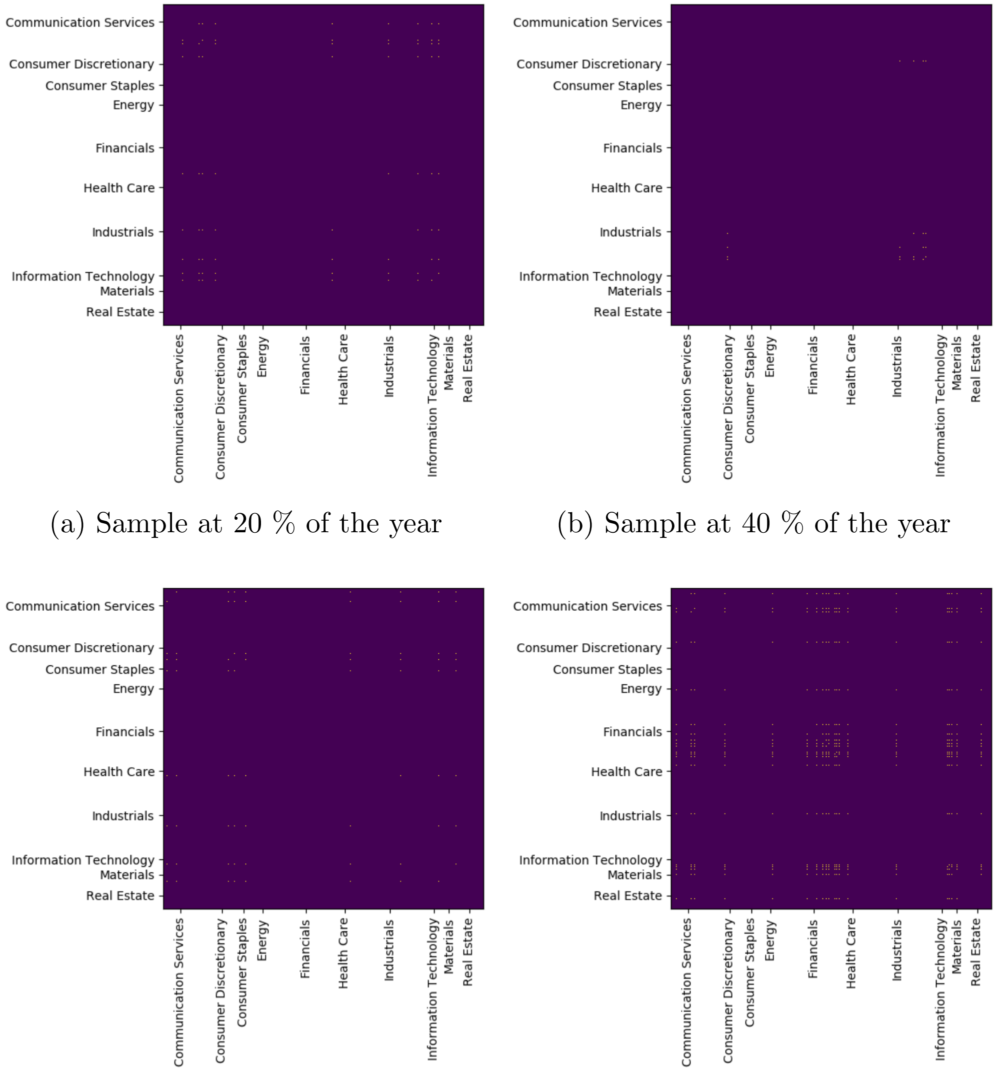
The choices for building the graph corresponding to one time step lead to a very sparse distribution as well as an high number of connected components. As an example, several adjacency matrices are reported in Figure 1.

Every stock-node whose growth was higher than the threshold is connected with all other stocks that behaved equally. As a consequence, every *daily* graph is characterized by many unconnected nodes and one single fully connected component. Even though it is difficult to display a visual representation of a graph with such a large number of edges and nodes, those peculiarities become clear analyzing connected components. The latter are reported in Figure 2 and 3.

2.2 Latent factor analysis

The analysis of latent factor is possible thanks to CP decomposition. The main parameter for the algorithm is R, the number of components the sum of which approximates the original third-order tensor. The choice of the right R is known to be a complex task, especially for very sparse tensor and for binary cell values - e.g. in [3], $AUTOTEN$ includes an automatic quality assessment for the decomposition.

In this work, R was chosen analyzing the error defined as the squared Frobe-



(a) Sample at 20 % of the year

(b) Sample at 40 % of the year

(c) Sample at 60 % of the year

(d) Sample at 80 % of the year

Figure 1: Adjacency matrix for sample days

nius norm of the difference between the real tensor and its reconstruction:

$$E = ||X - \hat{X}||_F^2$$

Figure 4 shows that the resultant chart is not *elbow-shaped* as expected. Since it was not possible to assess that after a given R the error decay becomes

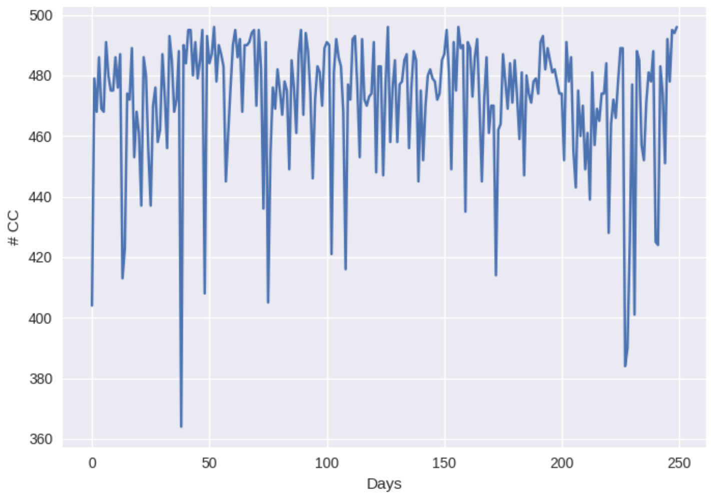


Figure 2: Number of connected components each day of the year. The larger the value on y-axis the larger the number of unconnected nodes - i.e. the lower the number of stocks that, on that day, have grown together.

less significant, another strategy was used.

The value $R = 52$ was chosen by evaluating an approximation of the discrete second derivative with explicit Euler's formula:

$$d(E)_t = E_{t-1} + E_{t+1} - 2E_t$$

and picking the point with maximum value. That value corresponds to point of the curve with maximum concavity.

Given this choice of R, 52 latent factors were produced. Each component is characterized by three column vectors:

- $S_1^{(500x1)}$: *weights* of stocks relative to that factor;
- $S_2^{(500x1)}$: *weights* of stocks relative to that factor;
- $T^{(250x1)}$: *weights* of time steps relative to that factor.

For sake of simplicity, this report includes only a subset of the 52 factors - in Appendix section it is reported a link to the Jupyter Notebook of the project

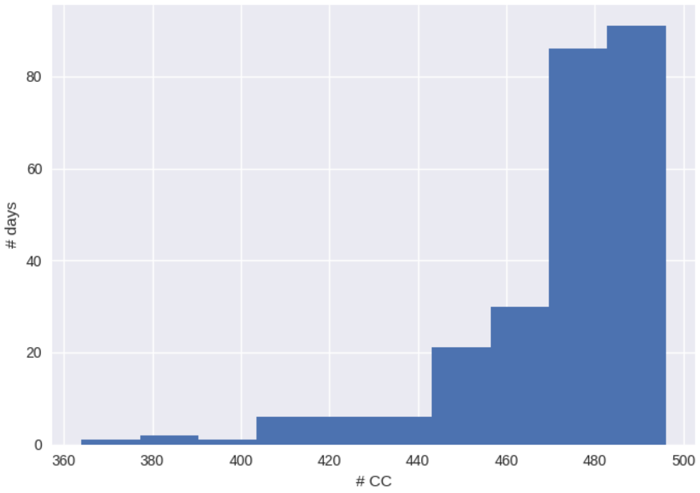


Figure 3: The distribution of connected components considering their number each day of the year.

containing the other charts. Furthermore, since there are almost 500 stocks, for synthesis purposes, factor plots were created by:

1. ranking by value in descending order the stock weights;
2. retaining only the top 5%;
3. grouping by GICS sector the weights.

All factors have a temporal component characterized by spikes in short periods of one or two days across the year. It could be a consequence of the binary encoding: the main membership stocks of the factor have a simultaneous growth in a restricted period while they are not related the rest of the time.

Co-clusters are not always clear since S_1 and S_2 contains similar rankings of the stocks (with a different scale): it is clear in Figure 6.

Figure 5 instead shows two clear dominant sectors: *Energy* and *Financials*. Such result might suggest that stocks belonging to those sectors have had similar periods of growth.

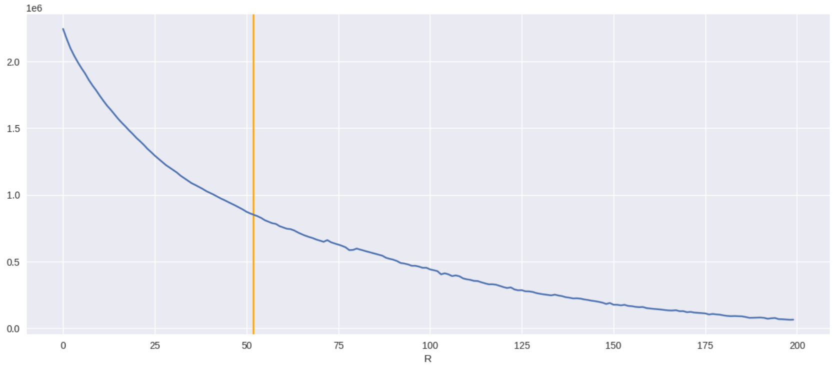


Figure 4: The reconstruction error versus the number of components. The orange line highlights the choice $R = 52$.

Figure 7 reports two quite different distributions in S_1 and S_2. The most influential sectors are not overlapped: *Industrials* and *Information Technology* lead in S_1 and *Communication Services*, *Consumer Discretionary*, *Consumer Staples* and *Materials* lead in S_2.

3 Conclusion and future works

This work aimed to highlight underlying relationships in stock price movements. What is known and consolidated is that stocks of the same sectors are likely going to move together across the time: they are likely influenced by the same external factors. The results highlighted in most of the cases that this rule is true: many latent factors have shown the highest values for stocks belonging to the same sector.

Nevertheless, the most interesting results regard those latent factors where membership values are not stocks within the same sector. They potentially give valuable information on positively correlated stocks.

Surely this work requires deeper exploration of several aspects. S_1 and S_2 appear to have similar distribution in most cases. It might be a bias introduced by the partially symmetric representation of input tensor - it recalls *SVD* decomposition of a positive symmetric matrix that generates two identical orthonormal factors. Given that, it could be useful explore a different encoding for tensor cells - e.g. using continuous values as the difference be-

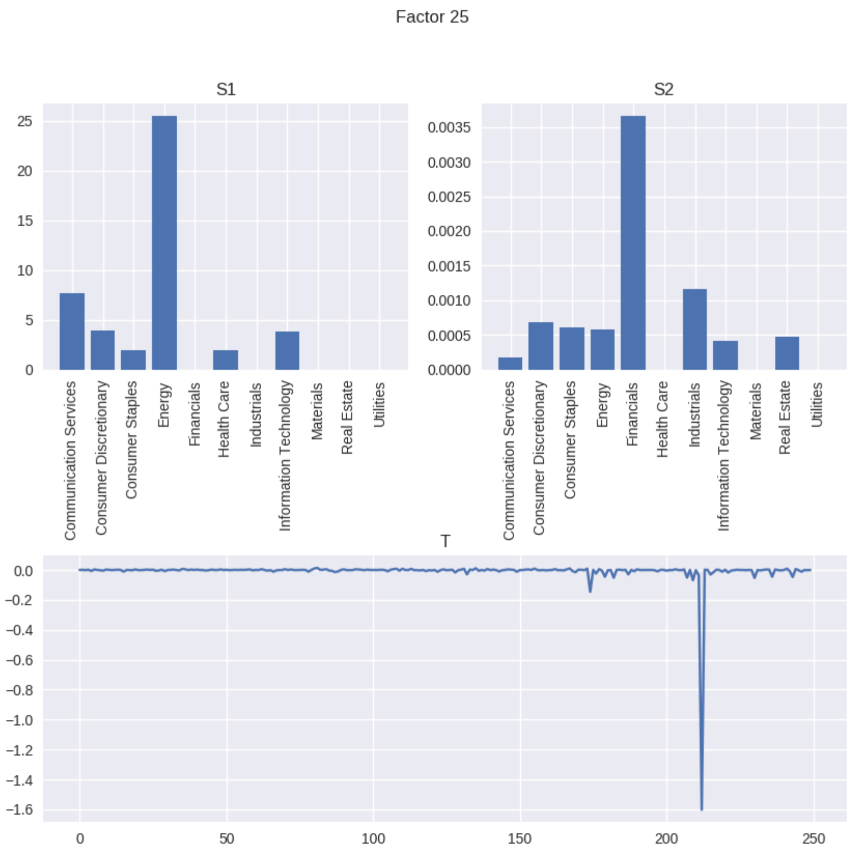


Figure 5: Factor 25.

tween the growth of the stocks. Another possibility would be a temporal aggregation: it would be equivalent to sum up two or more adjacency matrices and, consequently, eventually generate tensor cells larger than 1.
Finally, since the CP decomposition could not fit well with sparse binary tensors, this work could have an interesting extension with Boolean Tensors Decomposition. Unfortunately, such technique has not a ready-to-use implementation yet.

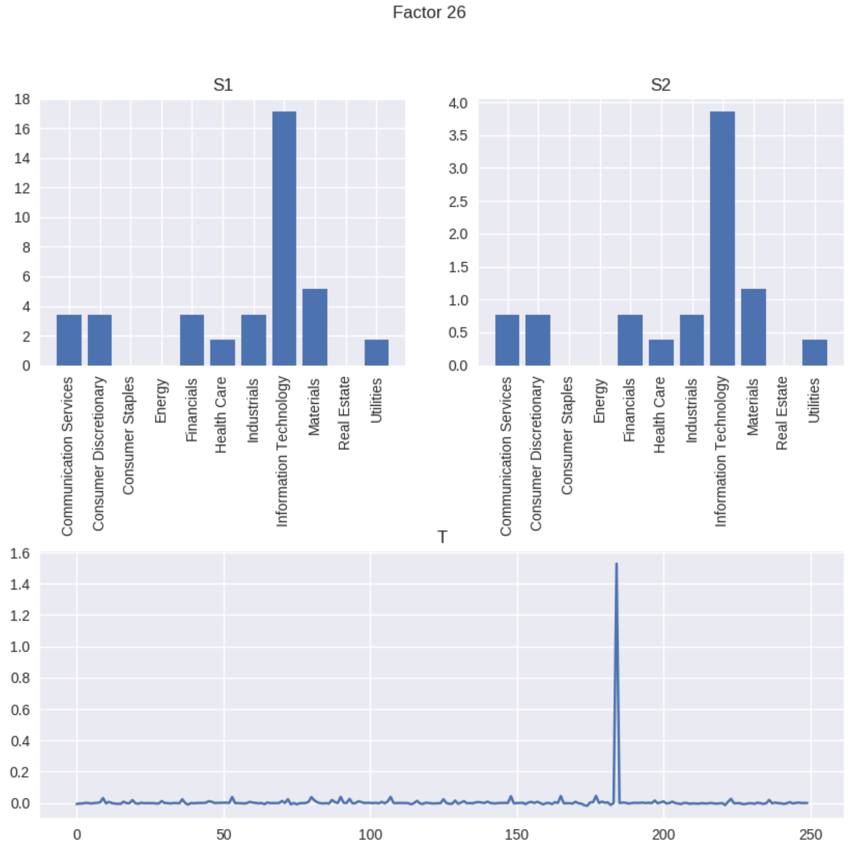


Figure 6: Factor 26.

4 Appendix

The code and the complete list of results are available at:
https://github.com/g8a9/dsnet
The Jupyter notebook used for the project is available at:
https://github.com/g8a9/dsnet/blob/master/final_project.ipynb

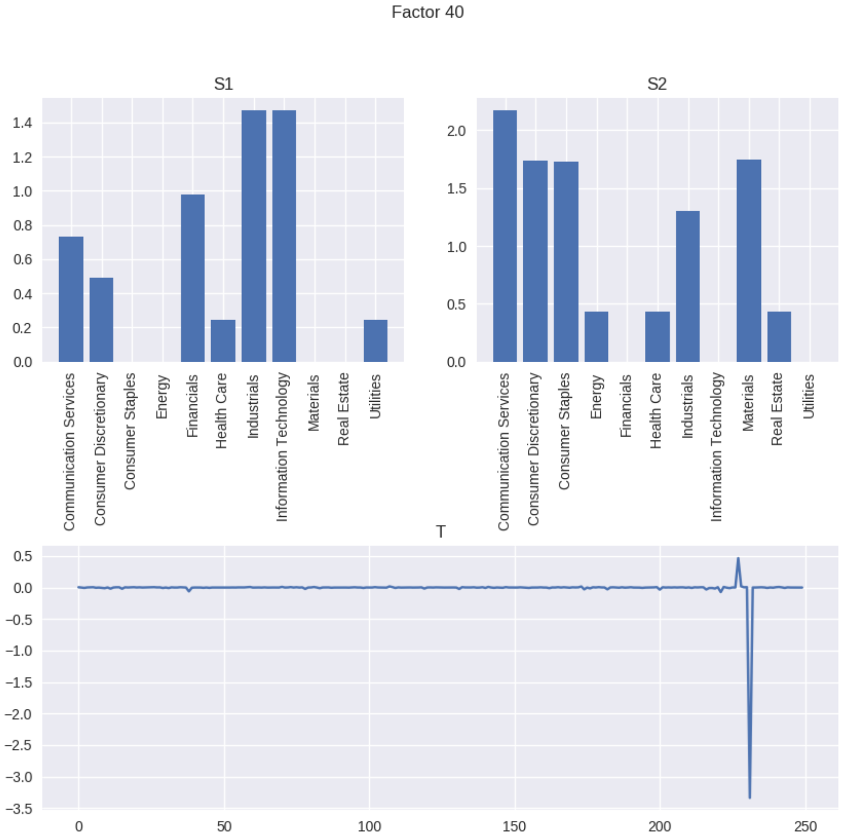


Figure 7: Factor 40.

References

[1] J Douglas Carroll and Jih-Jie Chang. Analysis of individual differences in multidimensional scaling via an n-way generalization of "eckart-young" decomposition. *Psychometrika*, 35(3):283–319, 1970.

[2] Richard A Harshman et al. Foundations of the parafac procedure: Models and conditions for an" explanatory" multimodal factor analysis. 1970.

[3] Evangelos E Papalexakis. Automatic unsupervised tensor mining with quality assessment. In *Proceedings of the 2016 SIAM International Conference on Data Mining*, pages 711–719. SIAM, 2016.