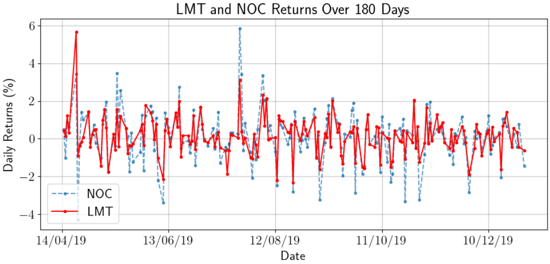
Similarity Analysis in Financial Time Series

Zella Baig
zella.baig@maths.ox.ac.uk

Supervisors :
Mohsin Javed (BlackRock)
Yuji Nakatsukasa (University of Oxford)



- Problem: We wish to extract some measure of "similarity" between stocks, but they are noisy. Can we seperate noise and signal?
- A possible solution: Singular Spectrum Analysis (SSA).

Outline of SSA: Constructing the Hankel Matrix

Consider[1] a time series of observations $Z_T = (z_1, \dots z_T)$. With fixed *window length* L and with $K := T - L + 1$:

1. Construct the (Hankel) trajectory matrix:

$$\mathbf{X} := \begin{bmatrix} z_1 & z_2 & z_3 & \cdots & z_K \\ z_2 & z_3 & z_4 & \cdots & z_{K+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ z_L & z_{L+1} & z_{L+2} & \cdots & z_T \end{bmatrix} \tag{1}$$

[1] Hassani, Mahmoudvand, et al. 2011.

2. Compute the *singular value decomposition* (SVD) of $\mathbf{X}$:

$$\mathbf{X} = \sum_{i=1}^{n} u_i v_i^T \sigma_i$$

3. Truncate the SVD to r rank-1 matrices, with *rank* r chosen s.t. $r \leq n$:

$$\mathbf{X} \approx \mathcal{X} = \sum_{i=1}^{r} u_i v_i^T \sigma_i$$

4. $\mathcal{X}$ is not necessarily Hankel, so re-diagonalise on the off-diagonals to reconstruct a de-noised series $\bar{Z}_T = (\bar{z}_1, \dots \bar{z}_T)$ from the averaged Hankel matrix

$$\bar{\mathbf{X}} := \begin{bmatrix} \bar{z}_1 & \bar{z}_2 & \bar{z}_3 & \dots & \bar{z}_K \\ \bar{z}_2 & \bar{z}_3 & \bar{z}_4 & \dots & \bar{z}_{K+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \bar{z}_L & \bar{z}_{L+1} & \bar{z}_{L+2} & \dots & \bar{z}_T \end{bmatrix} \tag{2}$$

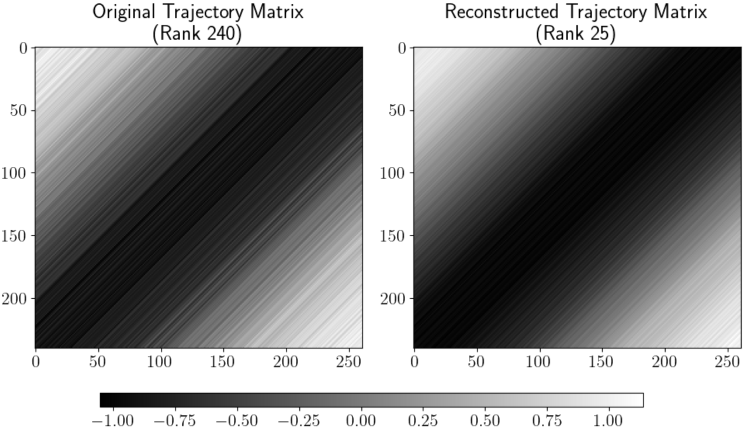


Figure 1: Noisy sinusoidal signal and denoised signals' trajectory matrices.

Choosing Parameters: Setting r

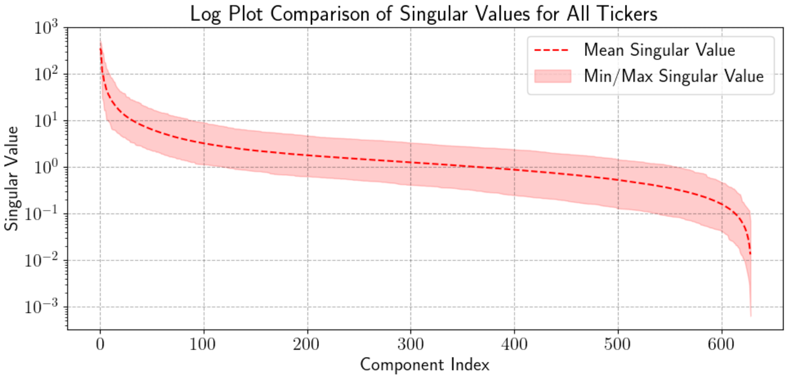


Figure 2: Choose r via examination of the scree plot, with knee at $r \approx 25$.

We measure similarity of two time (de-noised) time series using the Time Warped Edit Distance[2] (TWED). Why?

1. Cointegration v.s. correlation.
2. Elasticity
3. (Relatively) cheap

[2] Marteau 2008.

TWED: Graphical Example

	1	∞	18	17	16
Y_T	1	∞	13	12	13
	9	∞	4	13	14
		0	∞	∞	∞
			5	-3	1
				X_T	

Figure 3: Populated TWED grid, with $\nu = \lambda = 0.5$. $D_{T,T} = 16$.

SSA Reconstruction Rank Dependancy

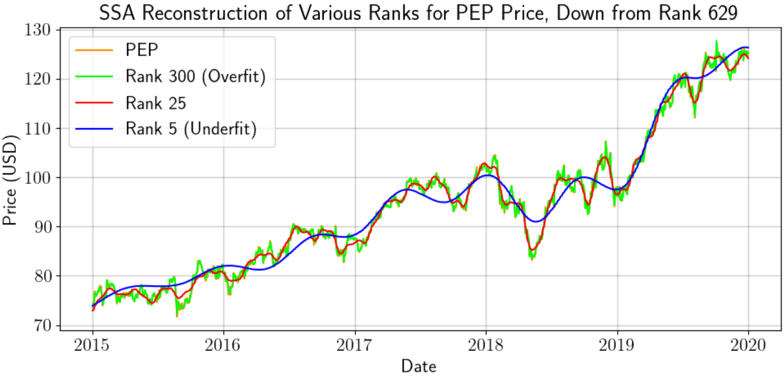


Figure 4: Different rank SSA reconstructions. Note underfitting at $r = 5$, and overfitting at $r = 300$.

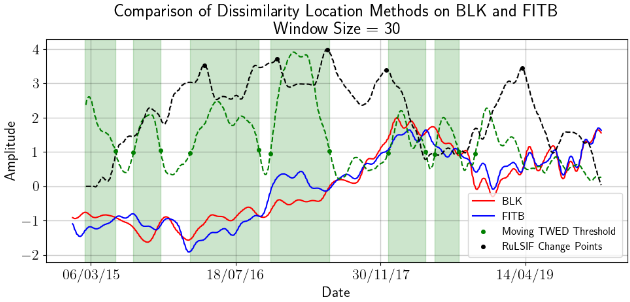
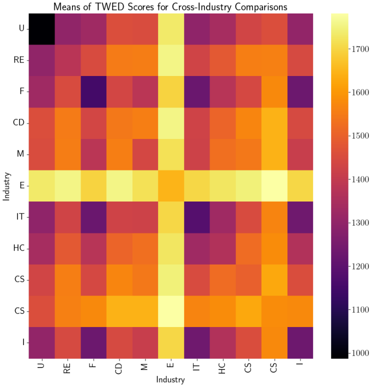


Figure 5: Implementation on our two dissimilarity finding methods on BLK and FITB over five years.

Inter-Industry Similarity (Mean)

Key takeaways:

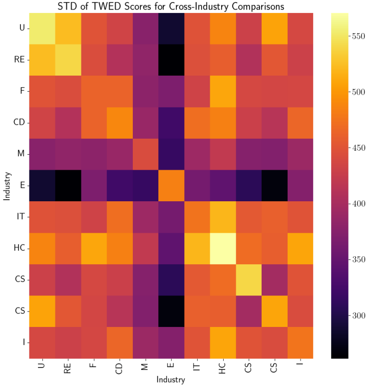
- Energy, Consumer Staples sector dissimilar to other sectors.
- Utilities, Finance, IT show strong inter-similarity.



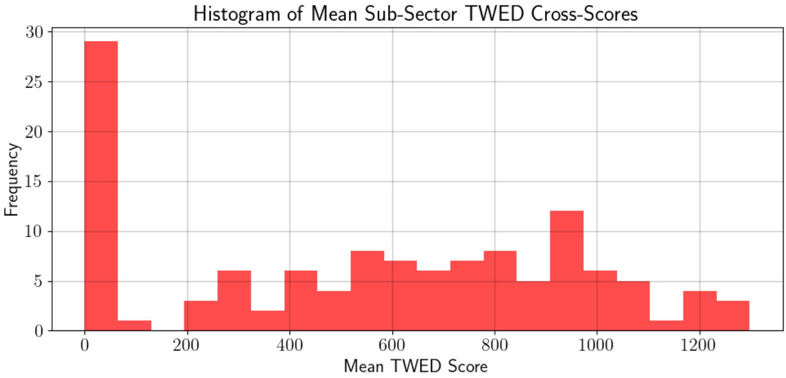
Inter-Industry Similarity (StDev)

Key takeaways:

- Energy sector conclusions strong.
- Utilities, Health Care conclusion very weak.



Sub-Sector Cross Scores

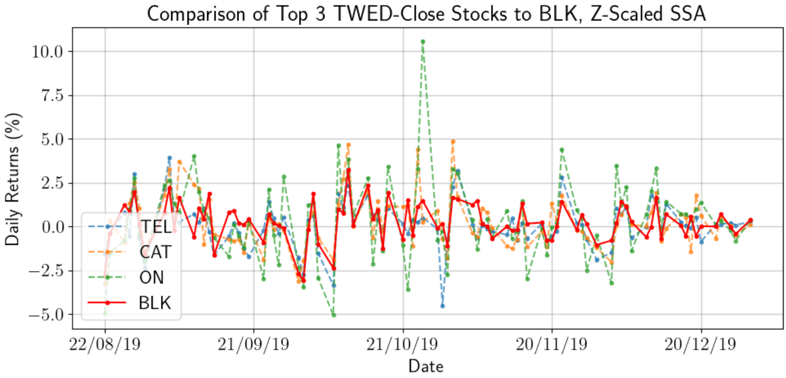


Back to Choosing a Similarity Measure

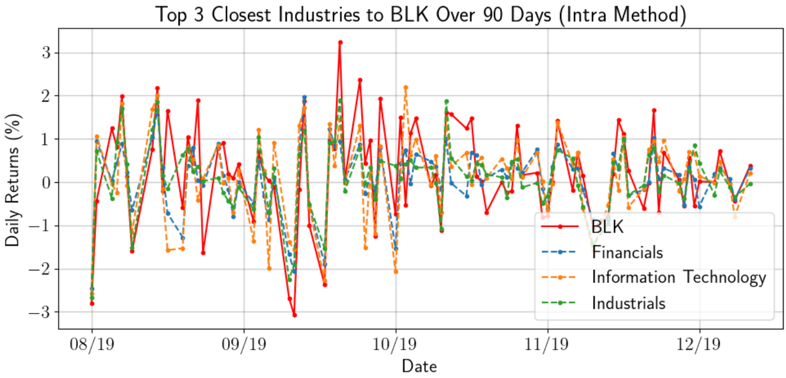
Similar Tickers	Top Five Tickers by Similarity Method			
	d_E	d_{DTW}	d_{TWED}	...
TIC_1	TIC_1	TIC_2	TIC_7	...
TIC_2	TIC_{23}	TIC_{11}	TIC_{99}	...
TIC_3	TIC_2	TIC_8	TIC_3	...
	TIC_{366}	TIC_{223}	TIC_{23}	...
	TIC_3	TIC_3	TIC_7	...

Table 1: Example scoring method used to help choose a similarity measure.

Ticker Similarity for BLK



Sector Similarity for BLK

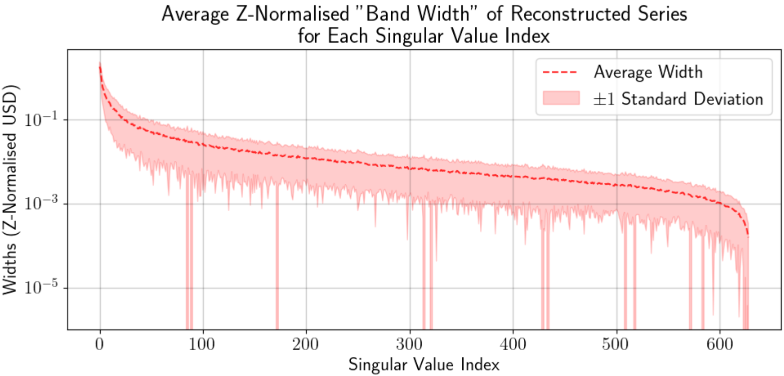


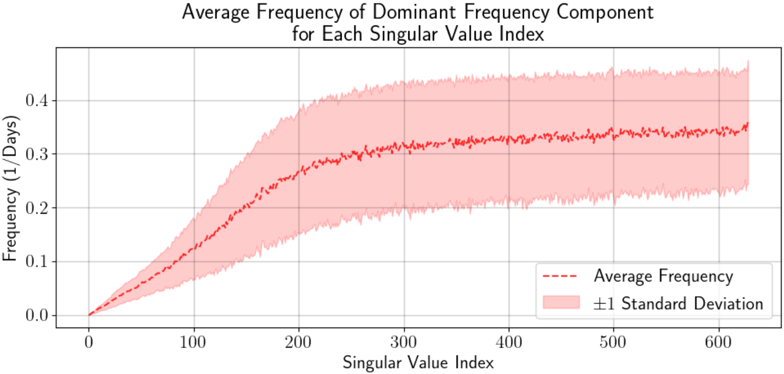
Similar Pairs

Example pairs of (potentially surprising!) similar tickers:

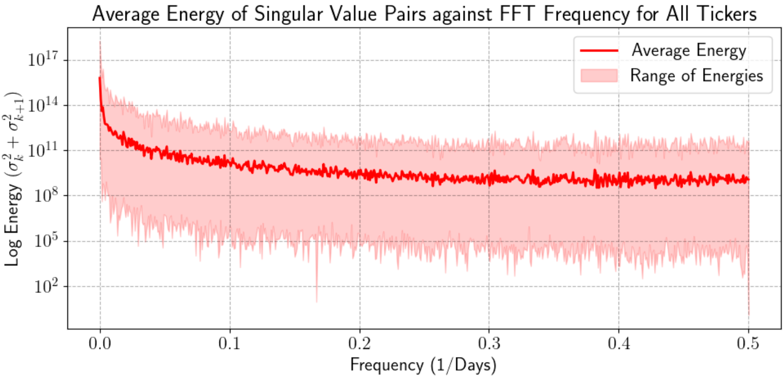
- Visa/Microsoft (Financials/IT)
- Mastercard/Intuit (Financials/IT)
- MSCI/Cintas (Financials/Industrials)
- TransDigm/FICO (Industrials/IT)
- Steris/CoStar (Health Care/Industrials)

Band Width


Average Z-Normalised "Band Width" of Reconstructed Series
for Each Singular Value Index
Average Width
±1 Standard Deviation
Widths (Z-Normalised USD)
10^{-1}
10^{-3}
10^{-5}
0
100
200
300
400
500
600
Singular Value Index


Average Frequency of Dominant Frequency Component
for Each Singular Value Index
Frequency (1/Days)
0.0
0.1
0.2
0.3
0.4
0
100
200
300
400
500
600
Singular Value Index
Average Frequency
±1 Standard Deviation

Singular Value Energy/Frequency Relationship



Other Points of Note

- Proper orthogonal decomposition
- Similarity v.s. distance
- `series_scorer`: A Python package for multiple time series scoring.
- Back testing!

Thank you!

zella.baig@maths.ox.ac.uk

Bibliography I

(1) Dey, S.; Dutta, A.; Toledo, J. I.; Ghosh, S. K.; Llados, J.; Pal, U. SigNet: Convolutional Siamese Network for Writer Independent Offline Signature Verification, Number: arXiv:1707.02131, 2017.

(2) Ghodsi, M.; Hassani, H.; Rahmani, D.; Silva, E. S. *Journal of Applied Statistics* **2018**, *45*, Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/02664763.2017.1401050, 1872–1899.

(3) Hassani, H.; Kalantari, M.; Yarmohammadi, M. *Comptes Rendus Mathematique* **2017**, *355*, 1026–1036.

(4) Hassani, H.; Mahmoudvand, R.; Zokaei, M. *Comptes Rendus Mathematique* **2011**, *349*, 987–990.

Bibliography II

(5) Hou, L.; Jin, X.; Zhao, Z. In *2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), 2019, pp 1–6.

(6) Marteau, P.-F. Time Warp Edit Distance, Number: arXiv:0802.3522, 2008.

(7) Serrà, J.; Arcos, J. L. *Knowledge-Based Systems* **2014**, *67*, 305–314.

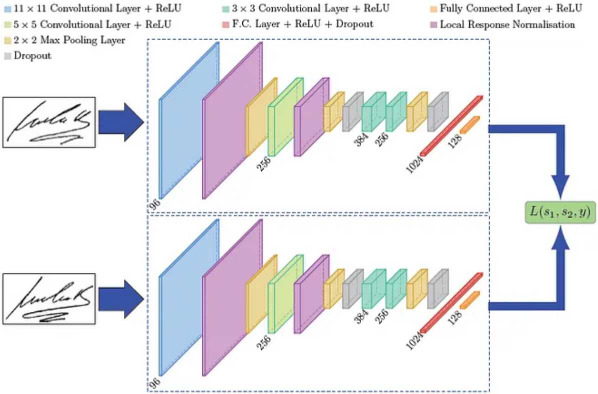


Figure 6: Overview of an SNN, as used in SigNet[3].

[3]Dey et al. 2017.

SSA Forecasting: Recurrent Forecasting

There exist two different types of SSA forecasting: recurrent, and vector. We go over them in turn:

1. Recurrent forecasting[4]: Consider the left singular vectors $u_1, u_2, \ldots u_r$. Take their L^{th} components, denoted π_i, and define
$$v^2 := \sum_{i=1}^{r} \pi_i^2. \tag{3}$$
Denote by $\hat{u}_i$ the $L-1 \times 1$ vector which is u_i with the final component removed.

[4] Ghodsi et al. 2018.

Then define

$$A = (\alpha_{L-1}, \dots \alpha_1)^T = \frac{1}{1-v^2} \sum_{i=1}^{r} \pi_i \hat{u}_i,$$

and thus construct

$$z_t = \begin{cases} \bar{z}_t & t = 1, \dots T, \\ \sum_{i=1}^{L-1} \alpha_i z_{t-i} & t = T+1, \dots T+h, \end{cases}$$

for a forecast to h steps ahead.

2. Vector forecasting[5]: First define

$$\hat{\mathbf{U}} = [\hat{u}_1, \dots \hat{u}_r],$$

and construct

$$\mathbf{\Pi} = \hat{\mathbf{U}}\hat{\mathbf{U}}^T + \left(1 - v^2\right) AA^T.$$

Finally, construct the operator Θ s.t.

$$\Theta V := \begin{bmatrix} \mathbf{\Pi}\hat{V} \\ A^T\hat{V} \end{bmatrix},$$

where $\hat{V}$ denotes the vector V with the last element removed.

[5]Ghodsi et al. 2018.

Define now

$$\Xi_i = \begin{cases} \mathcal{X}_i & i = 1, \dots K, \\ \Theta \Xi_{i-1} & i = K+1, \dots K+h+L-1, \end{cases}$$

where $\mathcal{X}_i$ are the columns of $\mathcal{X}$. Next construct

$$\boldsymbol{\Xi} = \left[\Xi_1, \dots \Xi_{K+h+L-1} \right],$$

and hankelise to get the matrix $\bar{\boldsymbol{\Xi}}$ from which we recover an "extended" time series containing forecasted values.