Bloomberg

# Big Data's Dirty Secret

Harvey J. Stein

Head, Quantitative Risk Analytics, Bloomberg L.P.

Yan Zhang

Quantitative Risk Analytics, Bloomberg L.P.

2018-6-29

**Abstract**

Amidst the avalanche of articles on big data and machine learning, the phrase "after cleaning the data" is often found. Here we focus on the work hidden behind this phrase. We analyze the types of dirty data found in financial time series, the problems caused by dirty data, and the performance of data cleaning algorithms. And we extend the MSSA hole filling algorithm of Kondrashov and Ghil [KG06] to improve its performance on CDS spread data, and combine it with clustering techniques from data science [KKZ10; BG05] to detect bad data.

**Keywords.** Data cleaning, big data, machine learning, SSA, MSSA, PCA, Data science, outlier detection, anomaly detection.

Bloomberg

## Contents



## List of Figures

Electronic copy available at: https://ssrn.com/abstract=3205524

# 1 Introduction

It is well known that outliers and bad data can corrupt the conclusions of statistical analyses. As a result, data cleaning is an integral part of statistical modeling, with many articles and books devoted to the subject such as those by Barnett and Lewis [BL94], Hawkins [Haw80], and Rousseeuw and Leroy [RL03]

Big data and machine learning exacerbate the issues. While machine learning techniques can often filter out noise, bad data can cause machine learning techniques to fail as well. So data cleaning is even more important in the big data/machine learning space than it is in general statistical analysis. Such issues are discussed in general by Aggarwal [Agg13], Ben-Gal [BG05], Hodge and Austin [HA04], and Kriegel, Kröger, and Zimek [KKZ10]

Accurately identifying bad data is especially difficult when working with big data due to the sheer volume of data. Having more data and high dimensional data makes visualization difficult, requiring modelers to blindly apply cleaning algorithms. If cleaning algorithms are not carefully tuned, cleaning can corrupt the data more than it corrects it.

When cleaning data, it is important to understand both the types of bad data and the usage of the data. While phenomenologically there are all sorts of bad data (fat finger trades, stuck sensors, incorrect units, misplaced decimal points, bad copies, ...), there are fundamentally two types of bad data, namely missing values and incorrect values. The goal of data cleaning is to address these two types.

The usage of the data must also be considered. In the financial space, data can be used for a variety of purposes, including relative valuation, mark to market, trading strategy development, and risk analysis. Each places different demands on the data and as a result require different data cleaning approaches.

For example, marking to market requires a current market snapshot that is close to the market to feed pricing models, and is largely unconcerned with historical behavior. On the other hand, both trading strategy development and risk analysis require historical data. But each places very different demands on the historical data set. Whereas both make use of daily returns, trading strategy development can fail spectacularly if missing data is filled using information from the future. This often results in strategies that yield phenomenal performance on the historical data set, yet lose a tremendous amount of money when actually put into practice. Risk analysis, being largely concerned with the distribution of historical returns, is largely insensitive to such considerations.

On the other hand, risk analysis is very sensitive to outliers. Outliers that are in fact correct need to stay in the data set, lest risk estimates become overly optimistic. On the other hand, not correcting bad values makes risk estimates overly pessimistic.

Here we address these issues for a particular financial data set that we use in risk analysis. We apply the Multivariate Singular Spectrum Analysis (MSSA) hole filling algorithm of Kondrashov and Ghil [KG06] to credit default swap (CDS) spread data, analyze its performance and introduce several enhancements of the algorithm to improve its performance and robustness. We then combine the algorithm with cluster analysis [BG05; KN98; KKZ10] to yield a general cleaning algorithm that both fills missing data and detects and corrects anomalies.


Electronic copy available at: https://ssrn.com/abstract=3205524

## 2 Desiderata

Before cleaning data, it is critical to understand the data as well as its intended usage. In our case, we were concerned with historical credit default swap (CDS) spreads which will be used for risk analysis.

CDS spreads are the cost (in basis points) of insuring against the default of a given company for a given time period. The spreads are quoted for insurance for 6 months, 1 year, 2 years, 3 years, 4 years, 5 years, 7 years and 10 years. If a default occurs within the insured time frame, the insured party is paid the loss that was incurred by owners of a reference instrument. CDS contracts can pay in different currencies, and pay the loss on either senior or subordinated debt. Each combination of currency and seniority can potentially have a CDS spread curve and thus separate quotes, although most commonly only USD and EUR spreads are quoted. This yields 16 separate time series for each company if only one currency is quoted, or 8 time series when only senior or subordinated debt CDS spreads are quoted in one currency.

CDS spreads are quoted for thousands of individual companies, yielding on the order of 35,000 time series. Risk analytics will make use of the daily changes to estimate a distribution of market moves. Due to regulatory requirements [Bas16], the data is needed as far back as January of 2007. This precludes cleaning each historical time series by hand, thus requiring an algorithmic approach.

The development of an effective algorithmic approach is impeded by the lack of a well defined metric for measuring algorithm performance. As a result, algorithms must be tried and retried and their performance must be visually inspected. In our case, we selected 40 curves for testing purposes. Given the large range in values, we needed to look at the data one year at a time. This yielded 400 graphs to inspect for each test, and dozens of test runs.

Given the volume of data involved, effort must be made to develop effective data visualization methodologies. The visualization approach we settled on for this work is illustrated in figure 1, which shows the CDS spread time series time for Avon's senior debt in 2009. The colors chosen range from blue to green as the tenor increases so as to make it clear when the curve is inverting or spreads are crossing. This is important because CDS spreads tend to rise as a function of maturity. Dots of the same color as the curve are used to indicate where data points are filled, with the dots being connected by a shifted color. Because there are often many missing points, the dots have to be small. Because it can be hard to see individual missing points, we add an indicator function for missing data, namely the black line at the bottom. Its value is nonzero on each date for which a data value is missing.

In addition to these general considerations, for CDS spread data, we also have to contend with a number of idiosyncratic features, as illustrated in figure 2. First of all, the 6 month point was not quoted until mid 2009. This means that data filling algorithms need to contend with long gaps in time series. Also, note that volatility can be extremely high and change over time. This makes detecting bad values difficult. The fact that spreads can span orders of magnitude over time and often exhibit regime changes further exacerbates this problem.

In general, for risk management, the concern is for the accuracy of the joint distribution of the returns of the risk factors being used. Return distributions differ by tenor (e.g. weekly return distributions differ from those gotten by compounding independent daily returns), so it's important


Electronic copy available at: https://ssrn.com/abstract=3205524

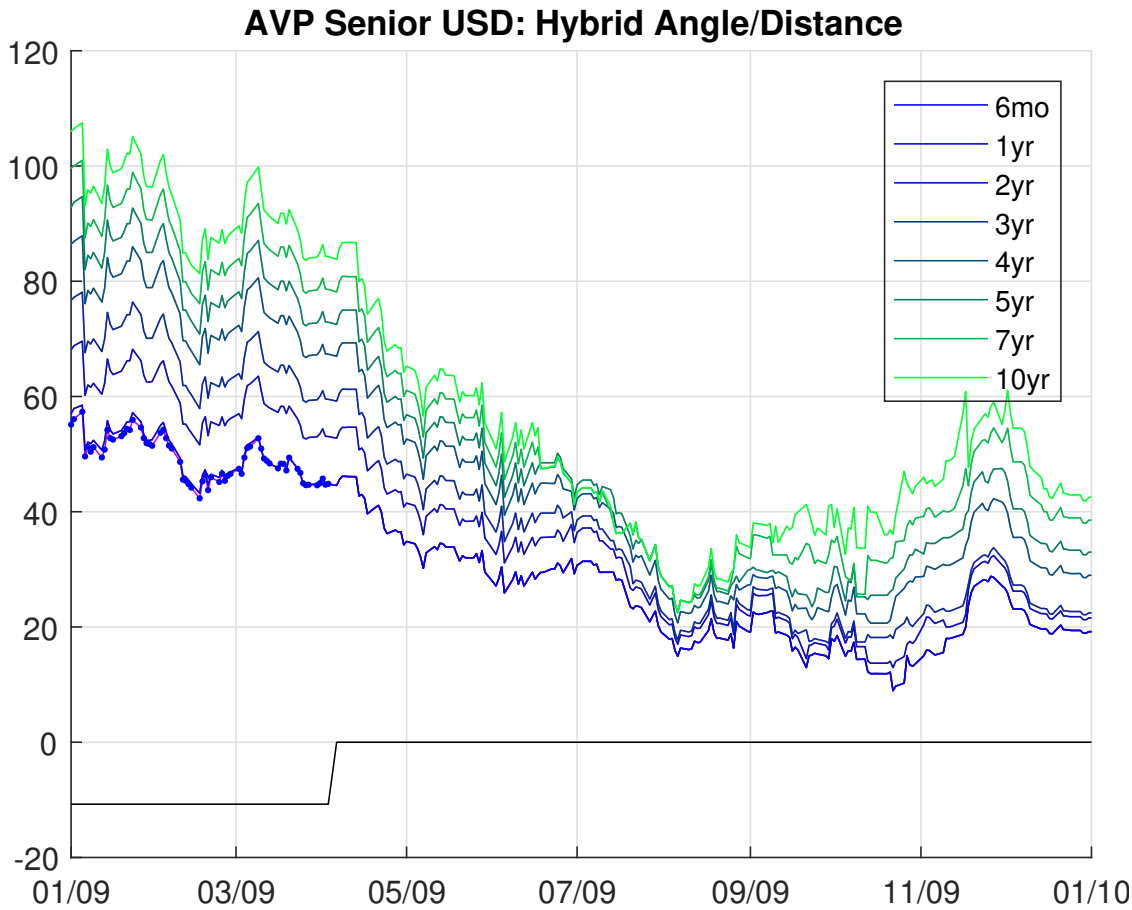


Figure 1: CDS spreads for Avon senior USD debt in 2009.

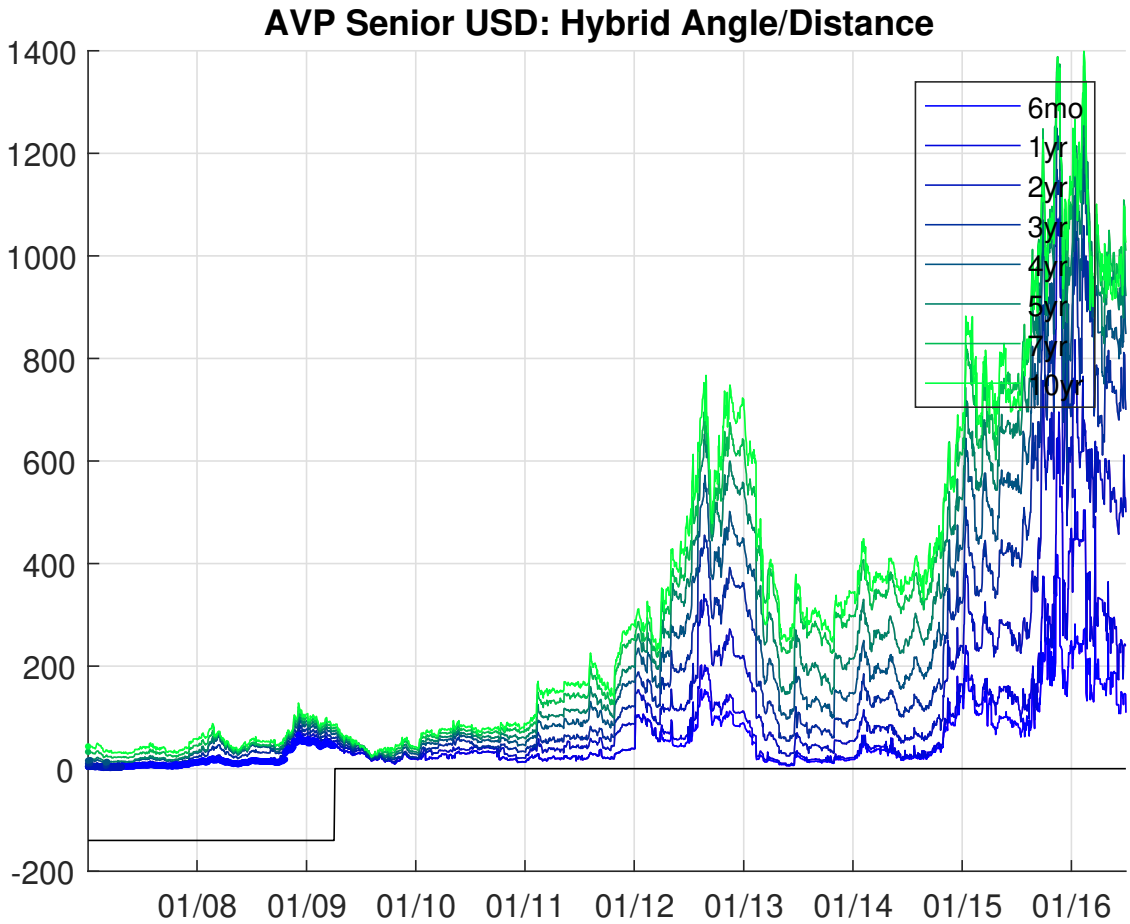


Figure 2: CDS spreads for Avon senior USD debt from 2007 to 2016. Note the large range of values and the varying volatility.


Electronic copy available at: https://ssrn.com/abstract=3205524

to have data that is uniformly of the same periodicity.

Missing data can corrupt distribution estimates, as can improper filling of missing data points. Filling data points by linear interpolation will tend to reduce the variance of the returns, whereas flat filling will increase the variance. It's better to use regression approaches which attempt to preserve the covariance. But regression approaches don't preserve autocorrelation, and cannot be implied if all the data is missing on a given date. EM algorithms [Sch01; DLR77] can improve on this, but still fail to preserve autocorrelation.

As for anomaly detection, there are a variety of approaches, from filtering out moves that are large when compared to the trailing standard deviation, to applying clustering algorithms to using neural networks [Agg13; BG05; HA04; KKZ10]. Trailing volatility methods tend to flag regime changes, making them problematic for CDS data. Neural network approaches suffer from making inexplicable decisions.

## 3 Hole filling

Expectation and Maximization Algorithm

For hole filling, we started with the Multivariate Singular Spectrum Analysis (MSSA) hole filling algorithm [KG06]. MSSA itself extends Singular Spectrum Analysis (SSA) to a set of time series. It makes use of both space relationships (covariance) and time relationships (autocovariance and cross-autocovariance).

The SSA algorithm is a nonparametric spectral estimation method for analysis of a time series $X =< x_i >$. The algorithm yields a decomposition of the time series into a sum of series based on the singular value decomposition of the Hankel matrix (trajectory matrix) $[T_{ij}] = [x_{i-j+1}]$. Different components tend to capture trend vs periodicity vs noise.

Usage of the SSA algorithm includes inspecting eigenvectors and components to extract specific features of the data, smoothing data by throwing away small eigenvalues, and forecasting [CG; GNZ01; HT10]. We've also found it useful for stabilizing correlation calculations by smoothing the data before computing correlations [Das+16a; Das+16b].

The MSSA algorithm applies the SSA algorithm to a set of time series simultaneously so as to take into account cross covariances and autocovariances [Ghi+02]. It reduces to the SSA algorithm when applied to just one time series. One use of MSSA is for forecasting [Pat+11; HM13].

MSSA analysis starts with $n$ time series $X^i =< x^i_j >$, $1 \leq i \leq n$, where each time series has $m$ elements. It starts by forming the Hankel (or trajectory) matrix $Y^i$ of each time series, where the rows consist of shifts of the time series. Thus, with $l$ shifts used (zero through $l-1$), the trajectory matrix $Y^i$ for the time series $X^i$ is the $l \times (m-l+1)$ matrix

$$Y^i = \begin{bmatrix} x^i_1 & x^i_2 & \dots & x^i_{m-l+1} \\ x^i_2 & x^i_3 & \dots & x^i_{m-l+2} \\ \vdots & \vdots & \vdots & \vdots \\ x^i_{l-1} & x^i_l & \dots & x^i_{m-1} \\ x^i_l & x^i_{l+1} & \dots & x^i_m \end{bmatrix} \tag{3.1}$$

The total transition matrix $Y$ for the calculation is given by stacking up the $Y^i$ matrices to yield


Electronic copy available at: https://ssrn.com/abstract=3205524

an $nl \times (m - l + 1)$ matrix:

$$Y = \begin{bmatrix} Y^1 \\ Y^2 \\ \vdots \\ Y^n \end{bmatrix} \tag{3.2}$$

Next, the singular value decomposition of $Y$ is computed, yielding

$$Y = U\Sigma V^t, \tag{3.3}$$

where $U$ and $V$ are unitary and $\Sigma$ is an $nl \times (m - l + 1)$ diagonal matrix. The nonzero elements of $\Sigma$ are the singular values of the SVD, which we order from largest to smallest.

The MSSA components of $Y$ are gotten by expressing $\Sigma$ as a sum. Let $\Sigma_k$ be zero except for containing the $k$th singular value in the $(k, k)$th position. Then

$$\Sigma = \sum_k \Sigma_k \tag{3.4}$$

and the $k$th component of $Y$ is

$$Z_k = U\Sigma_k V^t \tag{3.5}$$

and

$$Y = \sum_k Z_k \tag{3.6}$$

The MSSA matrix components of the original $Y^i$ matrices are the corresponding submatrices $Z_k^i$, where

$$Z^i = \begin{bmatrix} Z_1^i \\ Z_2^i \\ \vdots \\ Z_n^i \end{bmatrix} \tag{3.7}$$

and each matrix $Z_k^i$ has $l$ rows.

The MSSA components of each of the original time series are gotten by reverse diagonal averaging. Given a matrix $A = (a_{ij})$, the reverse diagonal average is the vector $R(A)$ given by

$$R(A) =< a_{11}, \frac{a_{12} + a_{21}}{2}, \frac{a_{13} + a_{22} + a_{31}}{3}, \dots > \tag{3.8}$$

Given the MSSA matrix component $Z_k^i$ of the $Y^i$ trajectory matrix for the time series $X^i$ corresponding to the $k$th singular value, the $k$th MSSA component of the $X^i$ is the reverse diagonal average of the elements of $Z_k^i$, namely $R(Z_k^i)$. Since $Y^i = \sum_k Z_k^i$, and $R(Y^i)$ is $X^i$, we have that

$$X^i = \sum_k R(Z_k^i) \tag{3.9}$$


Electronic copy available at: https://ssrn.com/abstract=3205524

The expression of each time series as the sum of the $R(Z_k^i)$ vectors is the MSSA decomposition of the original time series. Smoothing is achieved by doing a partial reconstruction which leaves out the smallest singular values.

The MSSA hole filling algorithm [KG06] attempts to reconstruct the missing data by iteratively using the MSSA components to fill the holes. The goal is to find the values for the holes such that filling them doesn't change the decomposition. The algorithm takes the following steps:

1. Select an initial number of singular values $k_0$ to use.
2. Nominally fill holes (e.g. via interpolation).
3. Use the level $k$ hole filling algorithm with $k = k_0$:
   - Run the MSSA algorithm.
   - Replace holes with the partial MSSA reconstruction using the $k$ largest singular values.
   - Repeat until convergence.
4. Increment $k$ by one and repeat the level $k$ algorithm until adding singular values have negligible impact.

What remains to be specified is what constitutes convergence of the level $k$ algorithm and what constitutes convergence of the full algorithm. In practice, one would use an $L^1$ or $L^2$ norm and would choose a distance that trades off speed versus precision while noting that requiring excessive precision can cause the algorithm to not converge. In addition, for robustness, convergence of the outer loop has to be adjusted by limiting the number of singular values that are incorporated.

This algorithm was originally tested in interest rate data [DZ16]. The results were fairly good when random observations of individual values were removed. When we applied the algorithm to CDS data, the results were mixed.

For example, on International Lease Finance Corp senior CDS spreads (figure 3), results are fairly good. Despite having no 6 month quotes and missing much 1 year and 2 year data as well as some 3 year and 4 year data, the algorithm fills the data at reasonable levels and follows the other tenors.

On the other hand, we observed that sometimes the MSSA hole filling algorithm yields results that follow the other data, but at the wrong overall level. This is illustrated in figures 4 and 5. Figure 4 shows the MSSA hole filling algorithm reflects the behavior of other data but at the wrong level, as can be seen by the jump between the reconstruction of the 6 month tenor and the actual data that occurs around 4/2009. And figure 5 shows the algorithm filling a substantial amount of data at the wrong level. We also observed bottoming out of the algorithm – the algorithm would yield negative spreads even though all of the data were positive. There was also a lack of jitter – the reconstructions are smoother than the actual data.

The most egregious error was that the reconstructions didn't match the overall levels in the data. We attributed this to the fact that the correct level is not in the initial components. As a result, reconstructing with just those components fills the holes at the wrong level. This incorrect level then continues to persist as the algorithm proceeds. We corrected for this by shifting the reconstruction so that the data matches the observed levels. In other words, we computed the difference between the partial MSSA reconstruction and the actual data bracketing the hole and added that linearly interpolated difference to the MSSA reconstruction. We call this anchoring, as it amounts to anchoring the reconstruction at the bracketing points.


Electronic copy available at: https://ssrn.com/abstract=3205524

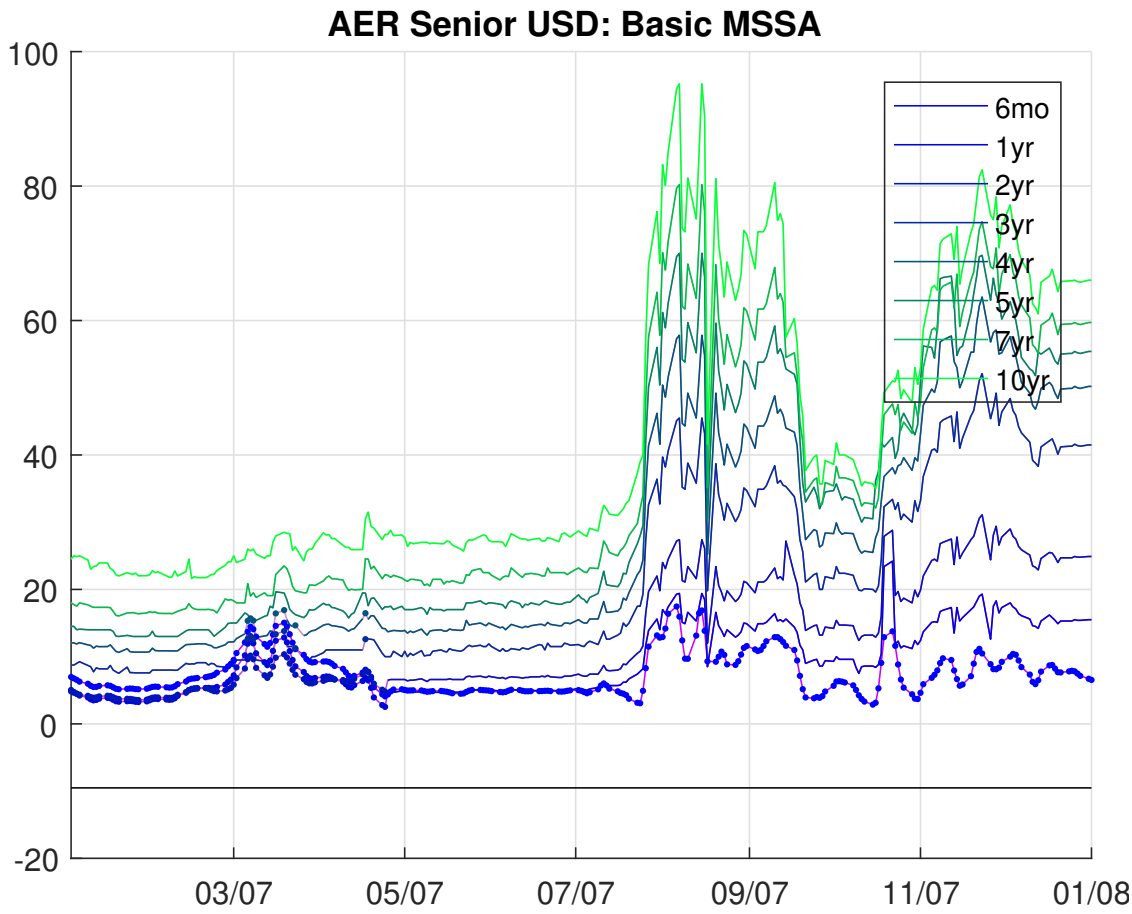


Figure 3: Original MSSA hole filling algorithm applied to International Lease Financial Corp senior USD CDS spreads. The algorithm does a good job of filling in the data.

Once anchoring was introduced, the question comes up of whether to adjust the partial reconstruction additively or multiplicatively. When using the MSSA algorithm directly on the spreads, we found that anchoring multiplicatively gave superior performance.

The next problem to address was that of negative spreads. One can view the MSSA hole filling algorithm as a fixed point algorithm. It tries to find values for the holes which match the reconstruction of the same points. This means the algorithm is performing an optimization and the fact that negative values are produced is a constraint violation. As such, we needed to treat this like a constrained optimization problem.

One common approach to constrained optimization problems is to apply a transformation that eliminates the constraint, or in other words, reparameterizing the problem. We applied that approach here, applying the MSSA algorithm in log space (i.e., on the log of the spreads) instead of in spread space. This solved the problem of negative spreads and also helped to make the algorithm more robust in the face of the large changes in magnitudes observed in the data. Working in log space is further justified by the fact that we observed the singular values dropping off faster when


Electronic copy available at: https://ssrn.com/abstract=3205524

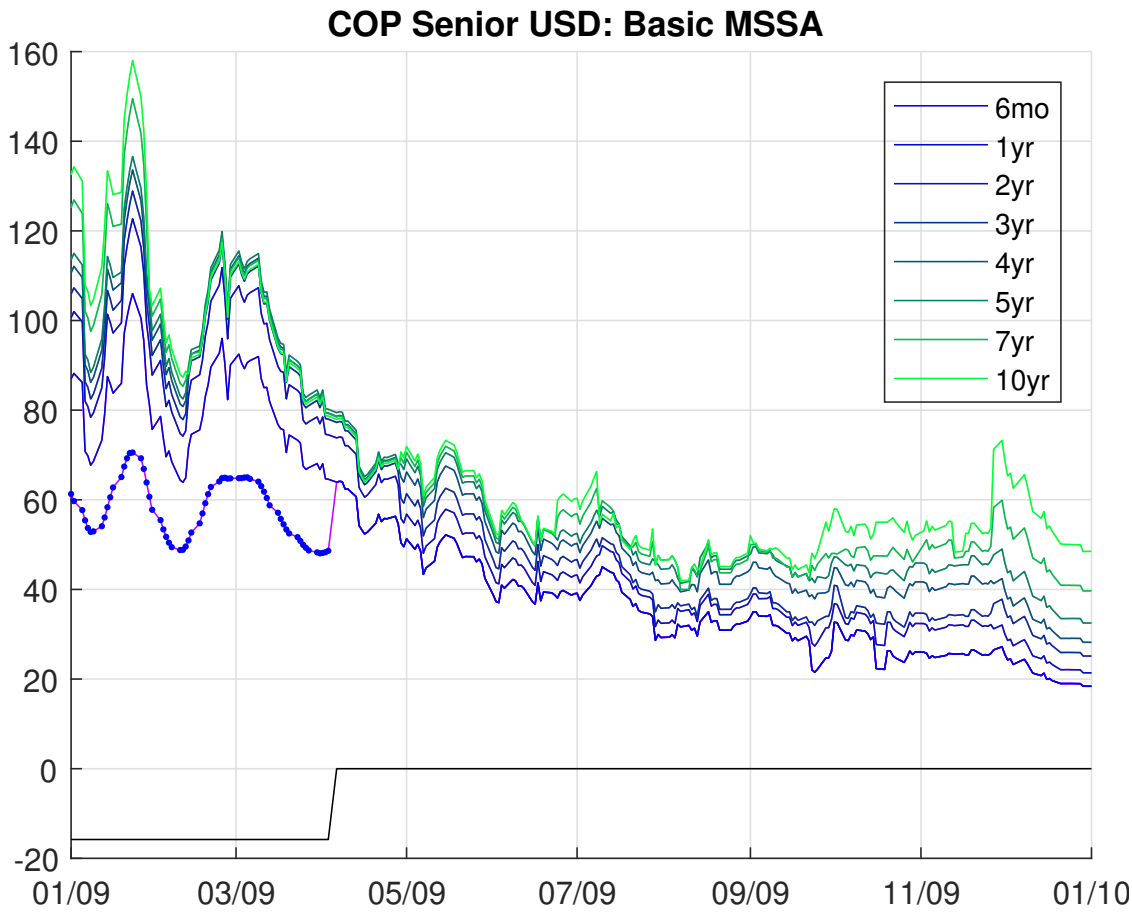


Figure 4: Original MSSA hole filling algorithm applied to ConocoPhillips senior USD CDS spreads. Note that the holes in the 6 month tenor are filled at too low a level, as illustrated by the jump that occurs when the 6 month tenor began being quoted.

working in log space than when working in spread space, as illustrated in figure 6.

As for bringing back the jitter, we increased the number of singular values used in the reconstruction. While this mostly worked, it also caused two artifacts. Firstly the algorithm sometimes failed to converge. Secondly, the algorithm sometimes filled the holes with extreme values. We theorize that the low singular values are capturing both the jitter as well as some of the idiosyncratic components, and thus sometimes will contain extreme values that are subsequently dampened by other components. This leads to instabilities in the algorithm. As a result, while we did adjust the convergence criteria and the number of singular values used, due to the reduced robustness, we refrained from reintroducing the jitter.

The end result of these modifications are illustrated in figures 7 and 8. The problems with incorrect levels and negative spreads were eliminated.


Electronic copy available at: https://ssrn.com/abstract=3205524

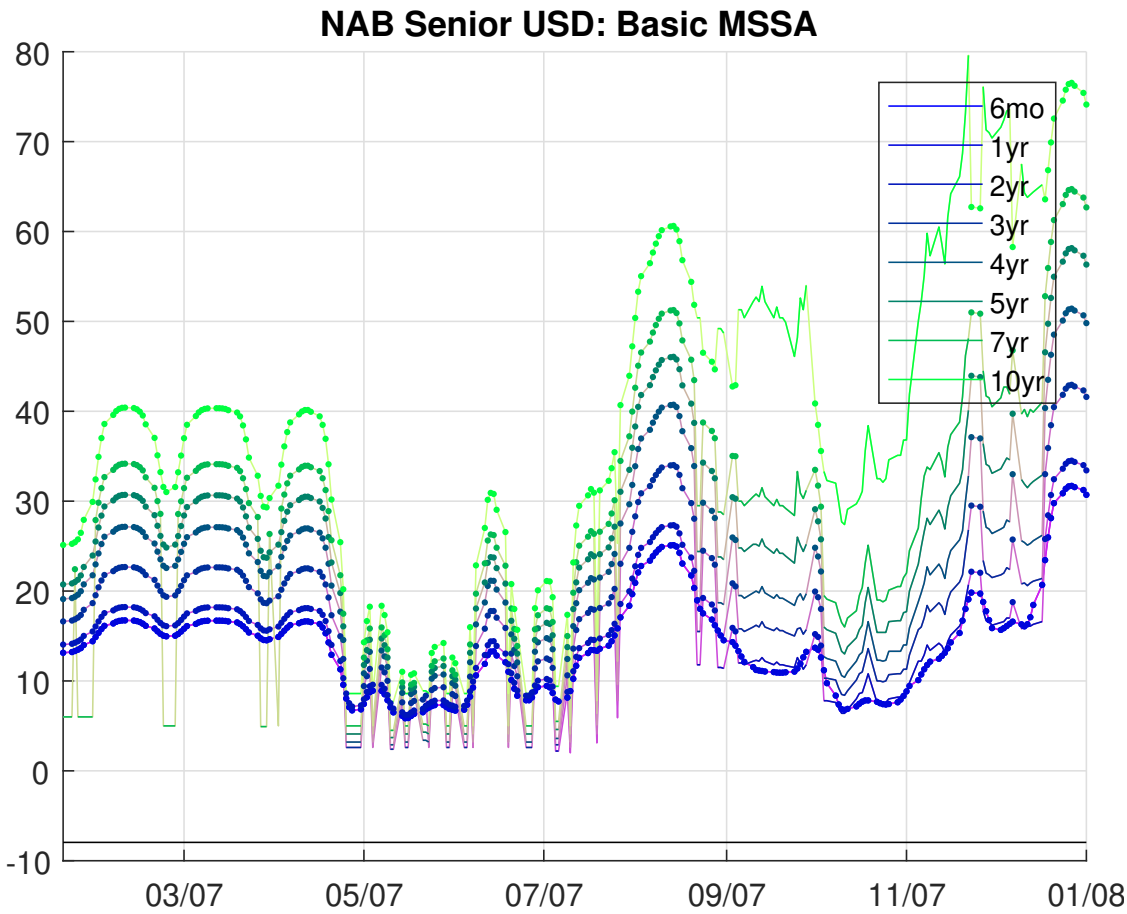


Figure 5: Original MSSA hole filling algorithm applied to National Australia Bank Ltd senior USD CDS spreads. We observe that the reconstruction deviates substantially from nearby observations.

## 4 Bad data handling

Bad data handling amounts to at least detecting it and removing it. Depending on the subsequent usage of the data, bad data might also be replaced by corrections.

There are a number of approaches to detecting bad data, including statistical, applying data science clustering techniques and using neural networks. General references include Aggarwal [Agg13], Hodge and Austin [HA04], Kriegel, Kröger, and Zimek [KKZ10], and Verhoevena and McAleer [VM]. Addressing time series data in particular is addressed by Verhoevena and McAleer [VM], and detecting spikes in real time data is discussed by Franke et al. [Fra+10].

In the case of CDS data, regime changes, changing volatility, and large (i.e., orders of magnitude) changes in values over time present substantial hurdles to accurately detecting bad data. Figure 9 illustrates several regime changes (around 11/14 and 2/15, for example), as well as the sorts of extreme volatility and changes in magnitude of the data that is common in CDS data.


Electronic copy available at: https://ssrn.com/abstract=3205524

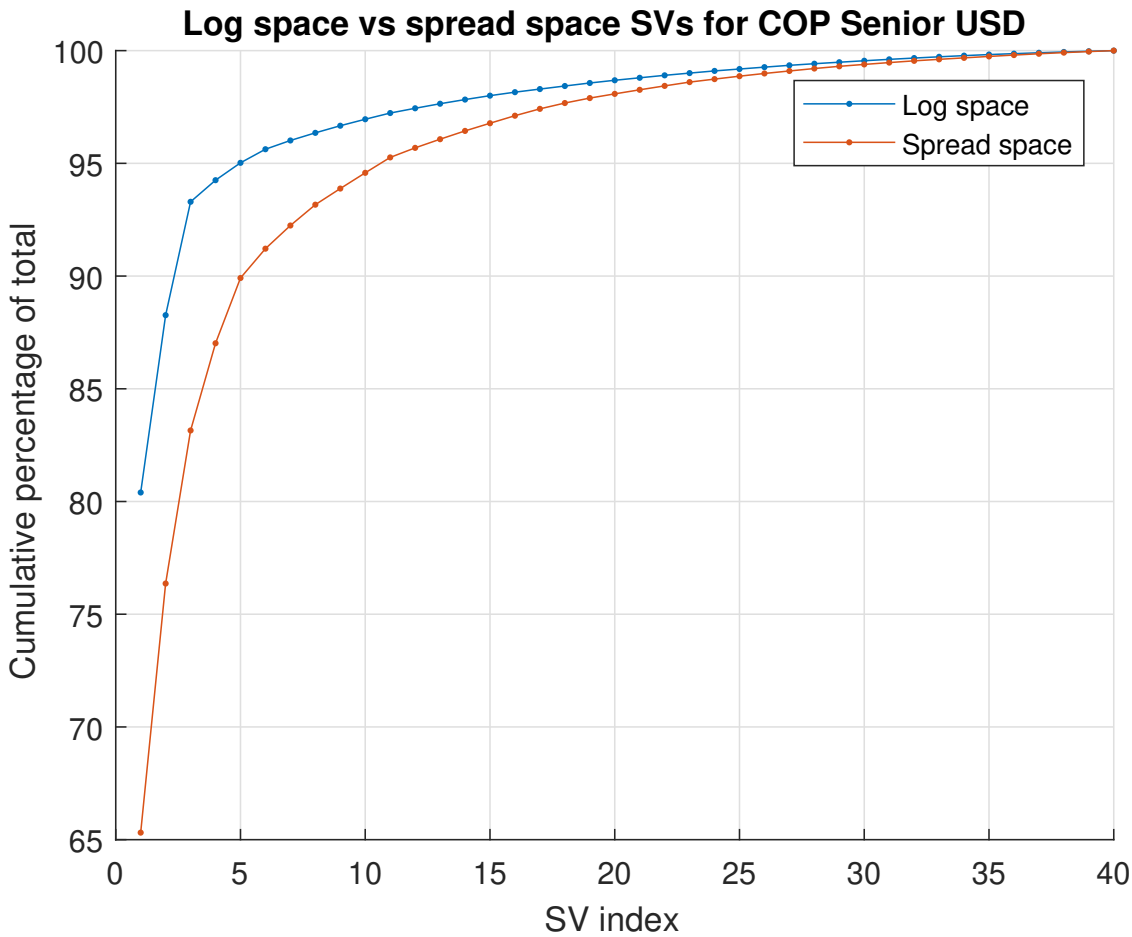


Figure 6: Comparison of singular values computed in spread space versus log space. We plot the cumulative percentage of the sum of the singular values. Note that the variance is more quickly captured in log space.

One common approach to detecting bad data is essentially statistical in nature. It amounts to computing the trailing standard deviation of the changes and flagging points which exceed some number of standard deviations. This has a number of shortcomings. First of all, it removes all of the large moves, thus reducing risk measures. Secondly, it tends to flag regime changes and jumps in volatility as bad data.

The data science approach is based on cluster analysis. All of the values observed on a given day are treated as an observation of a vector in a vector space. In our case, we used a vector on each day whose entries are all of the observed spreads for a given company, currency and seniority. One could consider adding additional dimensions so as to treat the senior and subordinate spreads together, or to include a relevant reference index. To account for the fact that observations that are far from each other in time should be less relevant than those that are close together, we also added the date as an additional dimension.


Electronic copy available at: https://ssrn.com/abstract=3205524

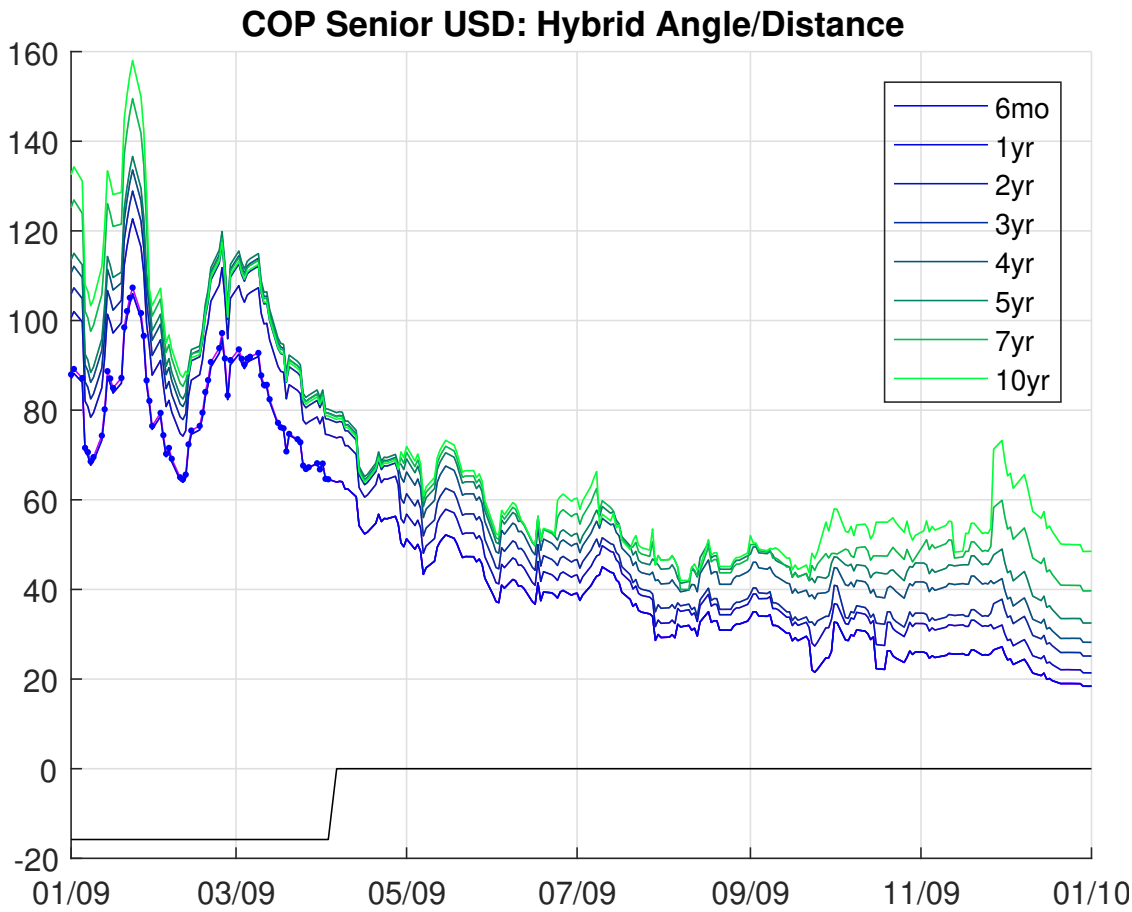


Figure 7: New MSSA hole filling algorithm applied to ConocoPhillips senior USD CDS spreads. Note that the new algorithm lines up with the correct levels.

There are two types of cluster analyses that can be done, namely distance based and angle based. Distance based clustering is based on the distance between observations. Angle based clustering is based on the range of the angles between observations.

In distance based clustering, all of the distances between observations are computed. An average inter-point distance is computed and a point which has too few points within a prespecified multiple of the inter-point distance is then considered a bad data point. Figure 10 illustrates this approach.

Angle based clustering works with the differences between points. Given a set of observations $x_i$, we fix a $k$, calculate the difference vectors $v_i = x_i - x_k$, and for each pair of vectors $v_i$ and $v_j$, compute the angle $\alpha_{ij}$ between them. Then observation $k$ is considered an outlier if all of these angles is small. Figures 11 and 12 illustrate this approach.

In practice, we found the angle based approach required modification. Rather than requiring all of the angles to be less than some cutoff, we use a number of percentages and cutoffs. If, for each cutoff, the percentage of angles less than the cutoff exceeds the corresponding percentage, then the


Electronic copy available at: https://ssrn.com/abstract=3205524

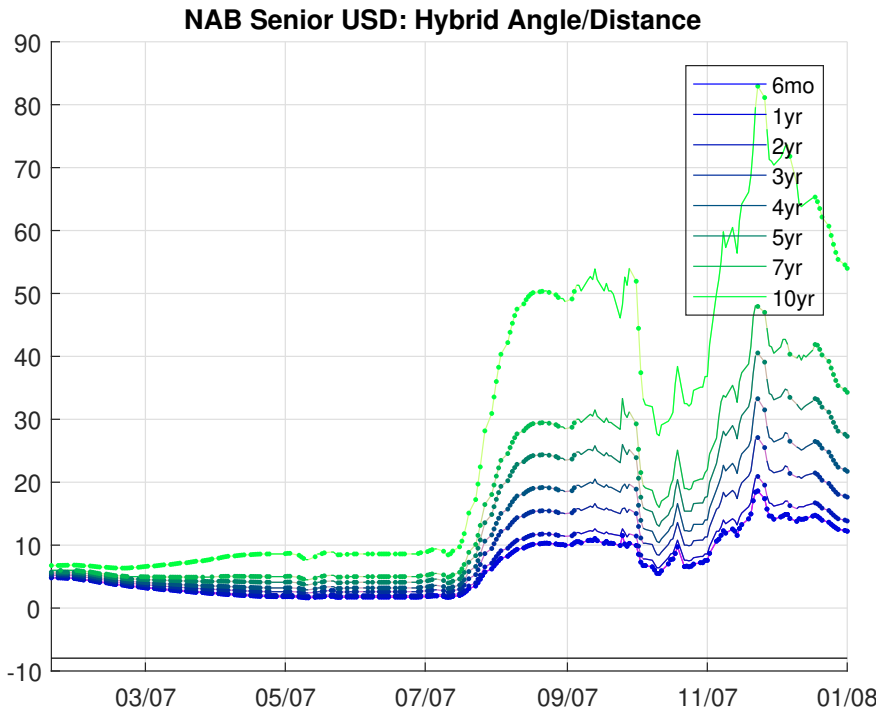


Figure 8: New MSSA hole filling algorithm applied to National Australia Bank Ltd senior USD CDS spreads. Anchoring solves the problem of incorrect levels.

point is considered an outlier.

Additionally, because of the extreme range of values, the clustering algorithms cannot be run on the entire data set. Instead, to determine if a point is an anomaly, the algorithm is run in a neighborhood of the point. In our case, we used a window of 40 days.

Even with these modifications, while the angle and distance based clustering algorithms performed better than the statistical approach, they still tended to choose too many extreme values.

To rectify this, we used both approaches and combined them with the MSSA hole filling algorithm. For each clustering algorithm, we deleted all of the points it flags as anomalies and then filled them using the MSSA algorithm. If the results were sufficiently close to the original values, we removed those points from the list of anomalies for that clustering algorithm. We only considered points to be anomalies when both modified clustering algorithms agreed that the points were anomalies.

Figures 13 through 15 illustrate the results of the hole filling along with the anomaly detection and correction. In figure 13, the algorithm fills the missing 6 month spreads in an appropriate fashion. Figure 14 illustrates that the algorithm avoids flagging jumps in levels and regime changes as anomalies. Figure 15 illustrates some anomalies that the algorithm flags and the values it replaces them with.

One additional result is given in figure 16. We had received a request to apply the algorithm to the historical CMO OAS spreads. In this case, the algorithm flagged the jump in the LC290OAS


Electronic copy available at: https://ssrn.com/abstract=3205524

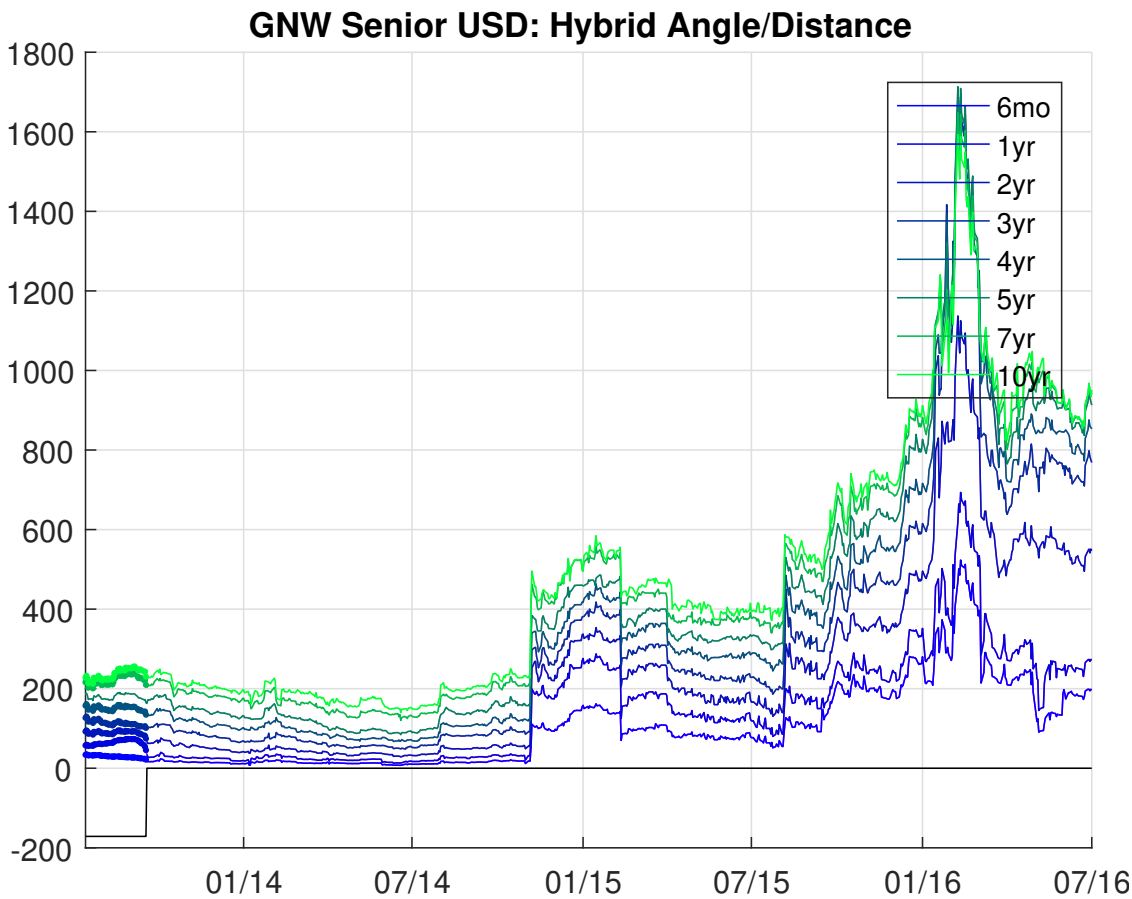


Figure 9: New MSSA hole filling algorithm applied to Genworth Holdings Inc senior USD CDS spreads. The large changes in magnitude and the regime changes make detecting bad data difficult.


Electronic copy available at: https://ssrn.com/abstract=3205524

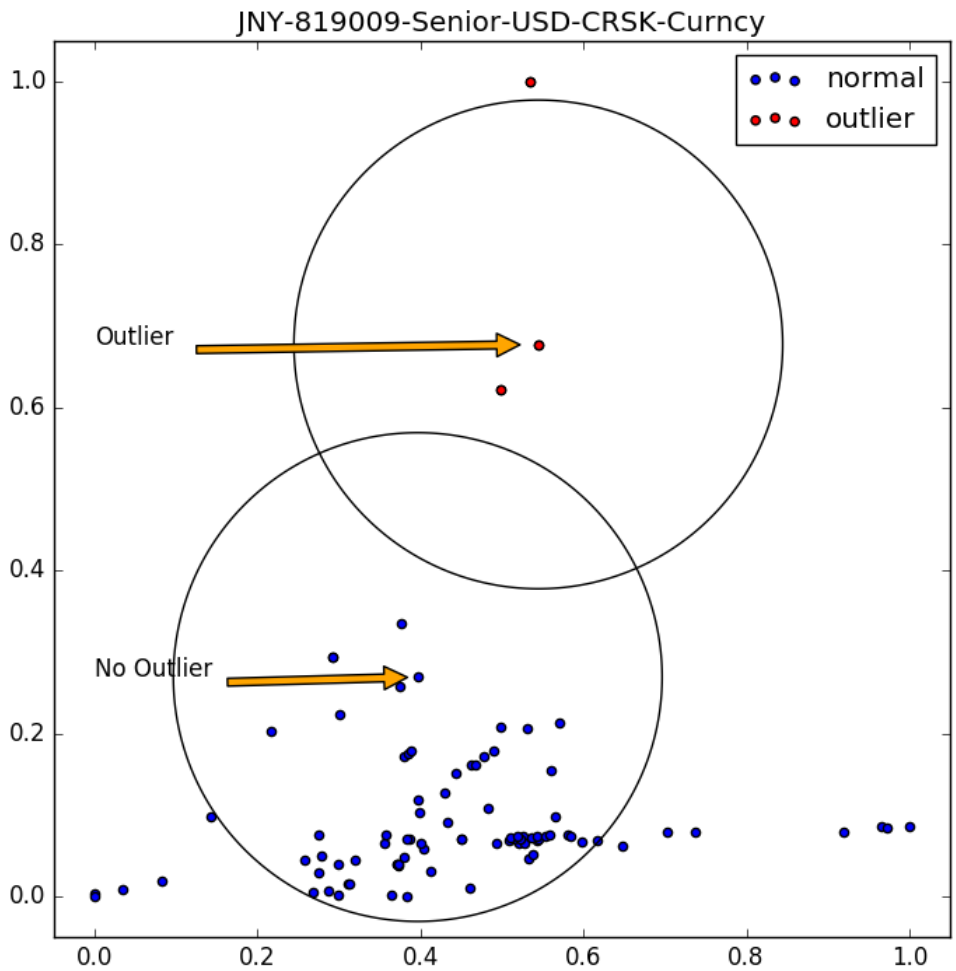


Figure 10: Distance based anomaly detection applied to Nine West Holdings senior USD CDS spreads. To illustrate, we just use two dimensions (the 6 month and 10 year tenors). Values are normalized to lie within the interval $[0, 1]$. Points with too few neighbors are flagged as anomalies.


Electronic copy available at: https://ssrn.com/abstract=3205524

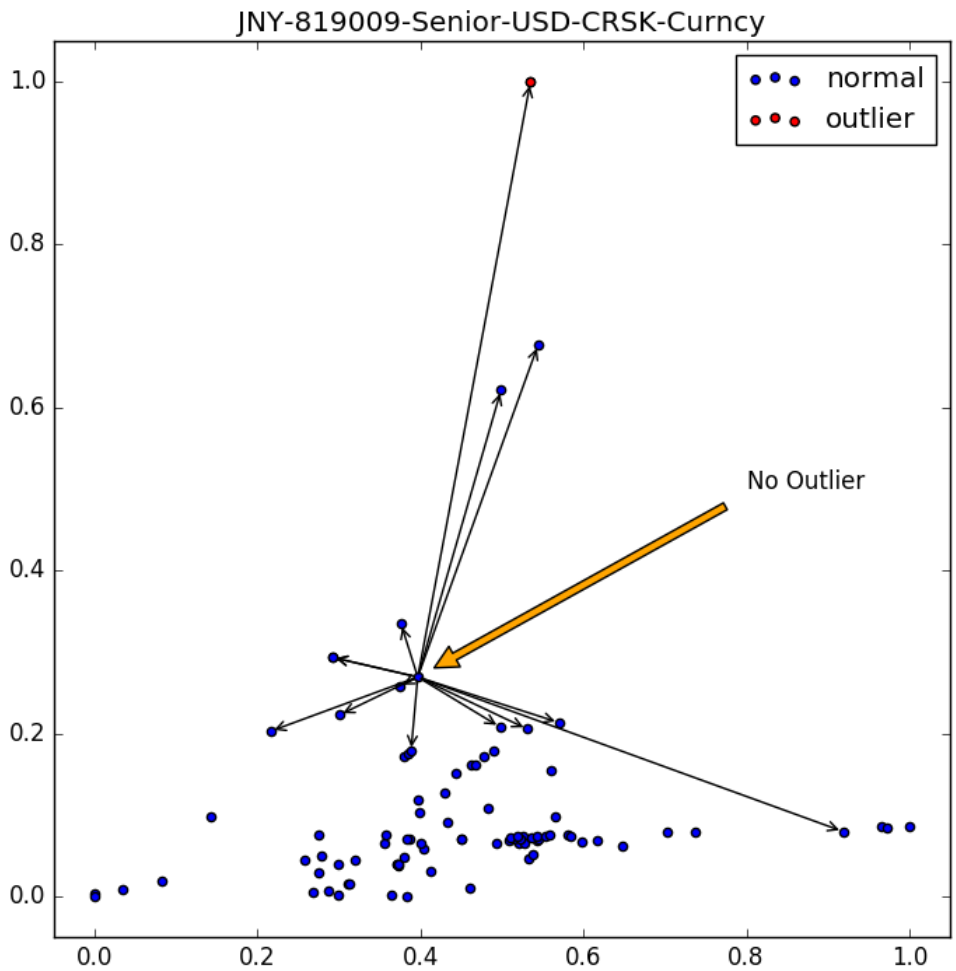


Figure 11: Angle based anomaly detection on the same data as in figure 10. A point is considered OK when there is a large range of angles.


Electronic copy available at: https://ssrn.com/abstract=3205524

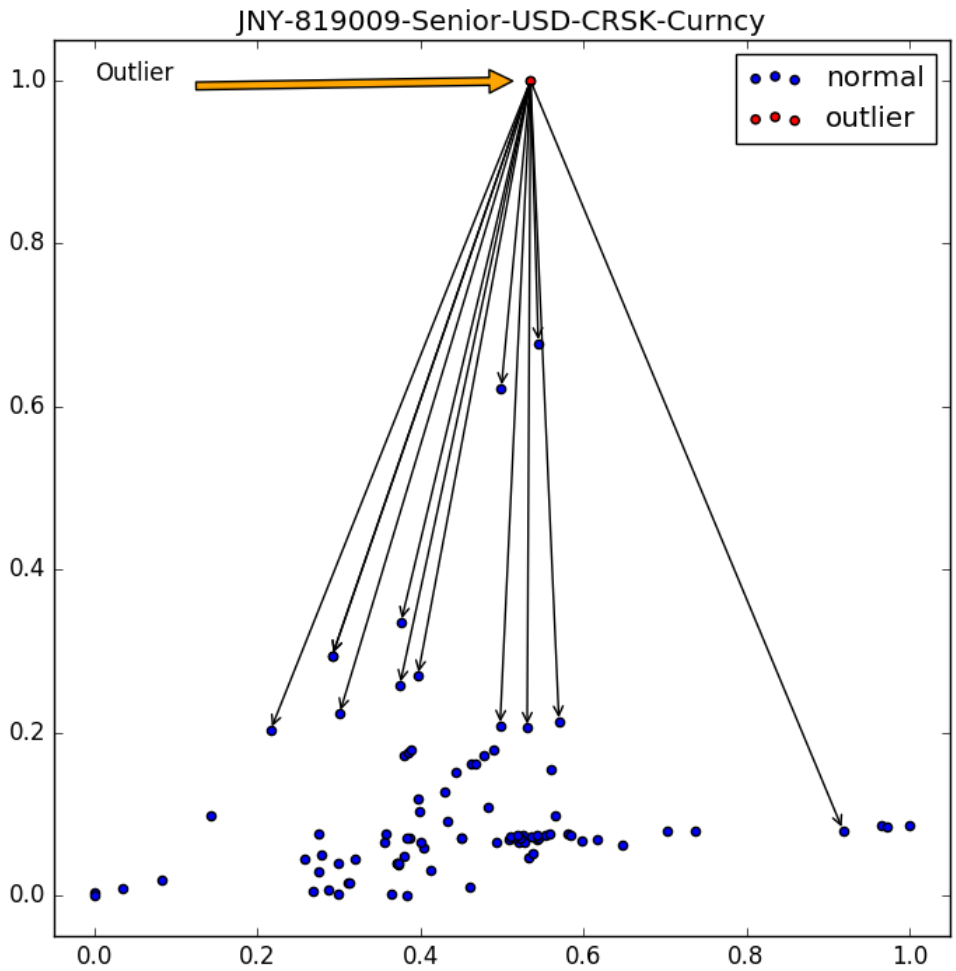


Figure 12: Angle based anomaly detection where an outlier is detected (using the same data as in figure 10). A point is considered an anomaly if there is a small range of angles.


Electronic copy available at: https://ssrn.com/abstract=3205524

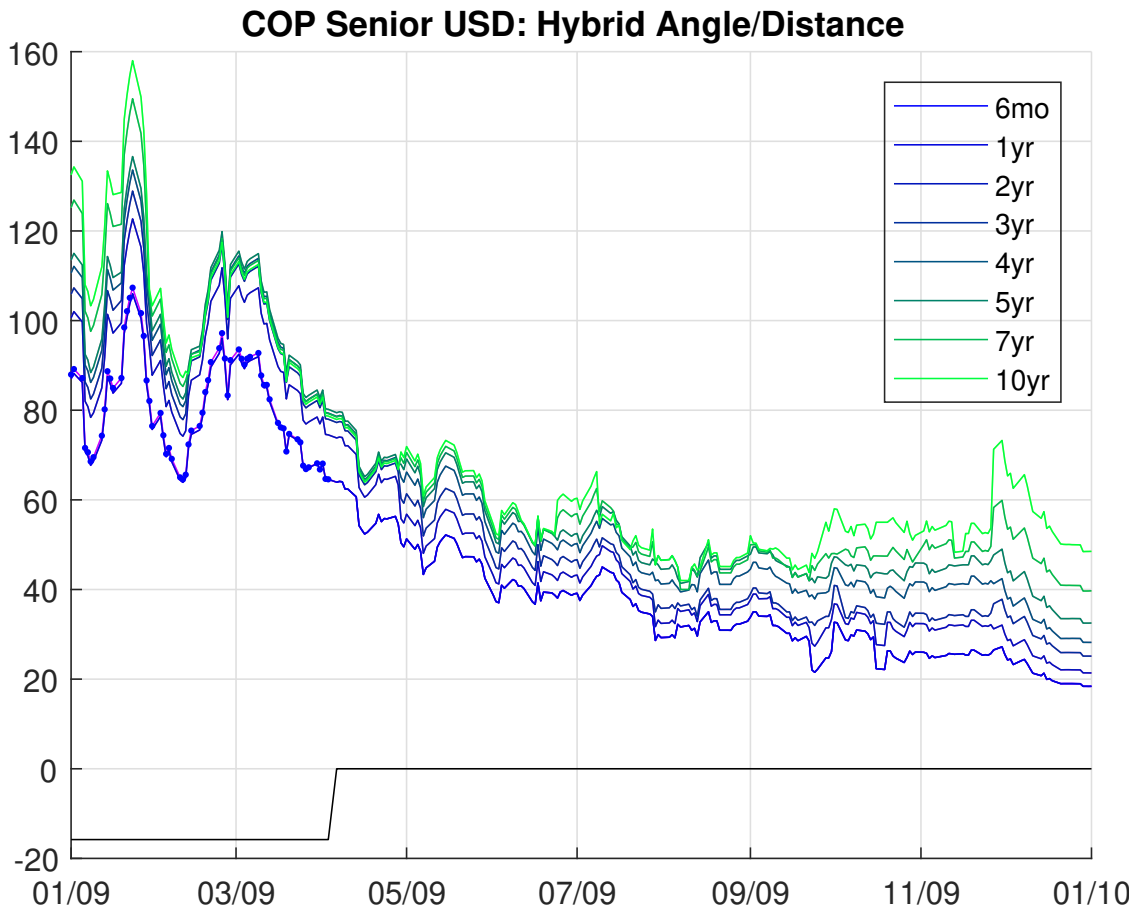


Figure 13: New tuned MSSA hole filling algorithm with anomaly detection applied to ConocoPhillips senior USD CDS spreads.


Electronic copy available at: https://ssrn.com/abstract=3205524

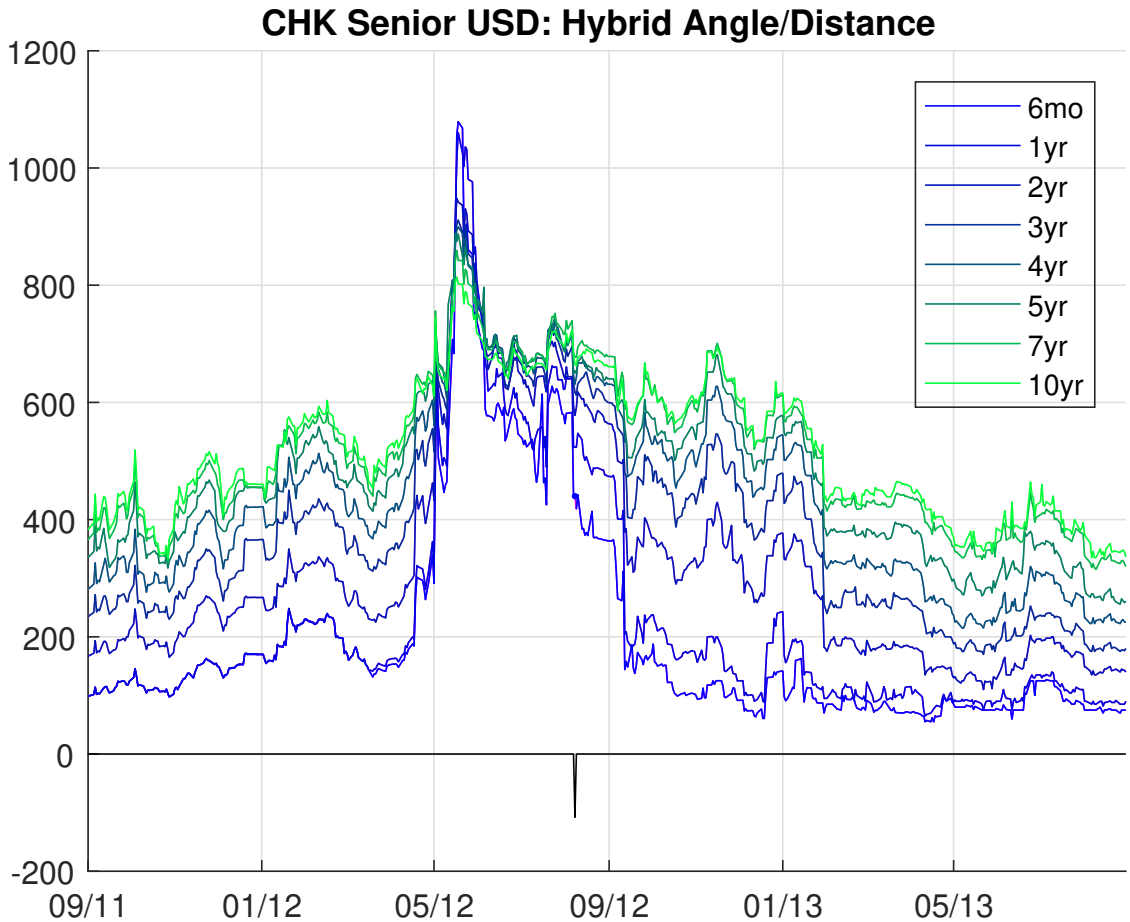


Figure 14: New tuned MSSA hole filling algorithm with anomaly detection applied to Chesapeake Energy Corp senior USD CDS spreads.


Electronic copy available at: https://ssrn.com/abstract=3205524

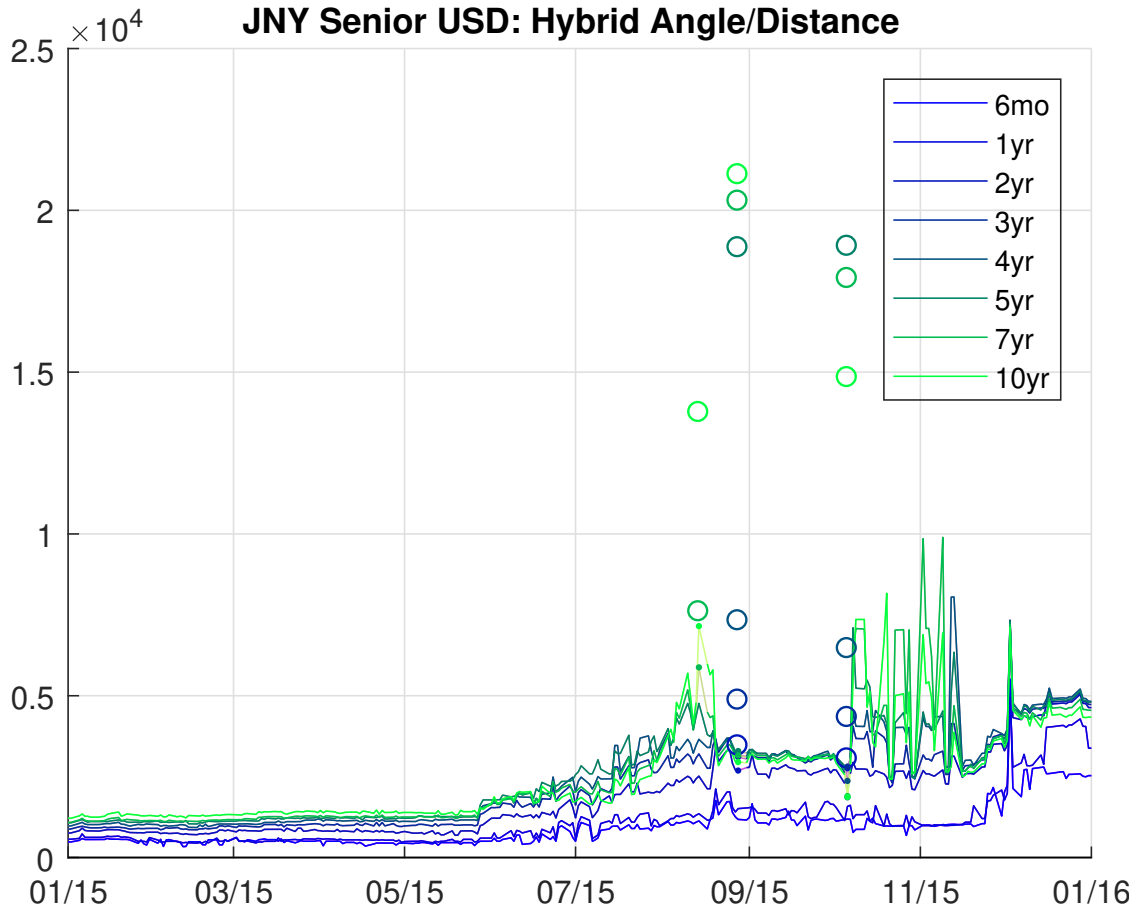


Figure 15: New tuned MSSA hole filling algorithm with anomaly detection applied to Nine West Holdings Inc senior USD CDS spreads.


Electronic copy available at: https://ssrn.com/abstract=3205524

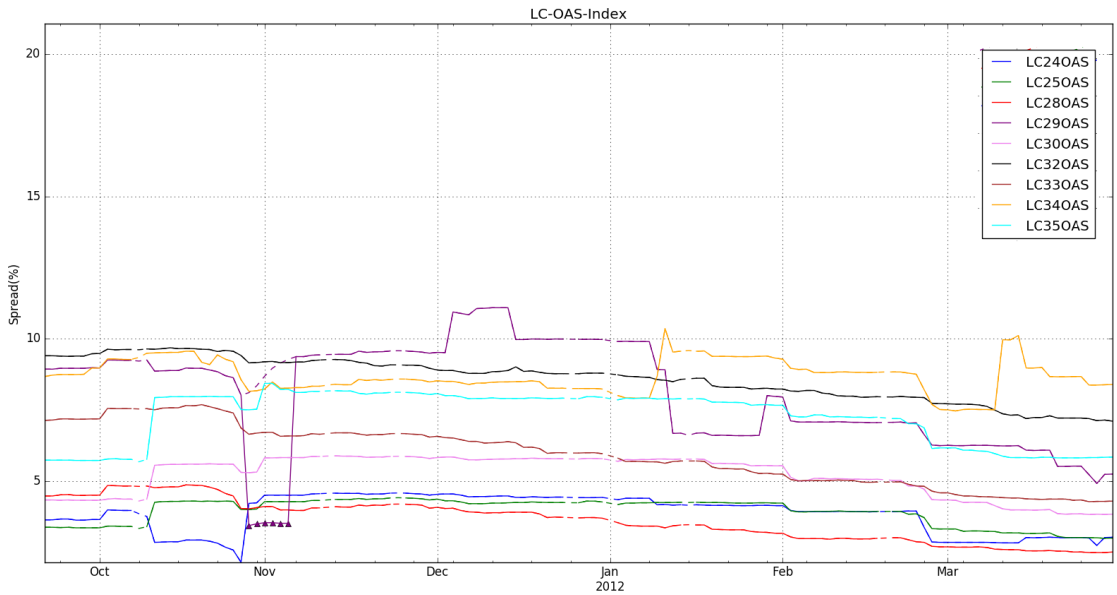


Figure 16: New tuned MSSA hole filling algorithm with anomaly detection applied CMO spreads.

index as an anomaly while avoiding flagging as anomalies long stretches where the spreads migrated between levels.

## 5 Summary

To summarize, cleaning data requires knowing the data and knowing its usage. The more data being used, the more important this becomes and the more important it is to apply robust algorithms to the problem.

There is much literature devoted to the subject, but it's not uncommon for algorithms which work well in one context to fail or be inappropriate in other contexts.

In our case, cleaning CDS data required developing, expanding, combining and tuning a variety of algorithms, combining data science approaches with a new version of the MSSA hole filling algorithm.

## References

[Agg13] Charu C. Aggarwal. *Outlier analysis*. Springer, 2013.

[BG05] Irad Ben-Gal. "Outlier detection". In: *Data mining and knowledge discovery handbook*. Ed. by Oded Maimon and Rokach Lior. Springer, 2005, pp. 131–146.

[BL94] V. Barnett and T. Lewis. *Outliers in Statistical Data*. Wiley, 1994.


Electronic copy available at: https://ssrn.com/abstract=3205524

[CG] David Claessen and Andreas Groth. *A beginner's guide to SSA*. CERES-ERTI, Ecole Normale Supérieure. URL: http://environnement.ens.fr/IMG/file/DavidPDF/SSA_beginners_guide_v9.pdf.

[Das+16a] Jan W. Dash, Xipei Yang, Mario Bondioli, and Harvey J. Stein. *SSA, Random Matrix Theory, and Noise-Reduced Correlations*. Tech. rep. Bloomberg LP, Sept. 2016. URL: https://ssrn.com/abstract=2808027.

[Das+16b] Jan W. Dash, Xipei Yang, Harvey J. Stein, and Mario Bondioli. *Stable Reduced-Noise 'Macro' SSA–Based Correlations for Long-Term Counterparty Risk Management*. Tech. rep. Bloomberg LP, May 2016. URL: https://ssrn.com/abstract=2808015.

[DLR77] Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. "Maximum likelihood from incomplete data via the EM algorithm". In: *Journal of the royal statistical society. Series B (Methodological)* 39.1 (1977), pp. 1–38.

[DZ16] Jan W. Dash and Yan Zhang. *Cleaning Financial Data Using SSA and MSSA*. Tech. rep. Bloomberg LP, Sept. 2016. URL: https://ssrn.com/abstract=2808156.

[Fra+10] Felix Franke, Michal Natora, Clemens Boucsein, Matthias HJ Munk, and Klaus Obermayer. "An online spike detection and spike classification algorithm capable of instantaneous resolution of overlapping spikes". In: *Journal of computational neuroscience* 29.1-2 (2010), pp. 127–148.

[Ghi+02] M. Ghil, M. R. Allen, M. D. Dettinger, K. Ide, D. Kondrashov, M. E. Mann, A. W. Robertson, A. Saunders, Y. Tian, F. Varadi, and P. Yiou. "Advanced spectral methods for climatic time series". In: *Reviews of Geophysics* 40.1 (2002).

[GNZ01] Nina Golyandina, Vladimir Nekrutkin, and Anatoly A. Zhigljavsky. *Analysis of Time Series Structure: SSA and Related Techniques*. CRC Monographs on Statistics & Applied Probability. Chapman & Hall, 2001.

[HA04] Victoria Hodge and Jim Austin. "A survey of outlier detection methodologies". In: *Artificial intelligence review* 22.2 (2004), pp. 85–126.

[Haw80] D. Hawkins. *Identification of Outliers*. Chapman and Hall, 1980.

[HM13] Hossein Hassani and Rahim Mahmoudvand. "Multivariate singular spectrum analysis: A general view and new vector forecasting approach". In: *International Journal of Energy and Statistics* 1.01 (2013), pp. 55–83.

[HT10] Hossein Hassani and Dimitrios Thomakos. "A review on singular spectrum analysis for economic and financial time series". In: *Statistics and Its Interface* 3.3 (2010), pp. 377–397. ISSN: 1938-7989.

[KG06] Dmitri Kondrashov and Michael Ghil. "Spatio-temporal filling of missing points in geophysical data sets". In: *Nonlinear Processes in Geophysics* 13.2 (2006), pp. 151–159.

[KKZ10] Hans-Peter Kriegel, Peer Kröger, and Arthur Zimek. "Outlier Detection Techniques". In: The 2010 SIAM International Conference on Data Mining, 2010. URL: http://www.imada.sdu.dk/~zimek/publications/KDD2010/kdd10-outlier-tutorial.pdf.


Electronic copy available at: https://ssrn.com/abstract=3205524

[KN98] Edwin M. Knorr and Raymond T. Ng. "Algorithms for Mining Distance-Based Outliers in Large Datasets". In: Proceedings of the 24th VLDB Conference New York, 1998.

[Pat+11] Kerry Patterson, Hossein Hassani, Saeed Heravi, and Anatoly Zhigljavsky. "Multivariate singular spectrum analysis for forecasting revisions to real-time data". In: *Journal of Applied Statistics* 38.10 (2011), pp. 2183–2211.

[RL03] P. Rousseeuw and A. Leroy. *Robust regression and outlier detection*. Wiley, 2003.

[Sch01] Tapio Schneider. "Analysis of incomplete climate data: Estimation of mean values and covariance matrices and imputation of missing values". In: *Journal of climate* 14.5 (2001), pp. 853–871.

[VM] Peter Verhoevena and Michael McAleer. *Detecting Local Outliers in Financial Time Series*. Department of Economics, University of Western Australia.

[Bas16] Basel Committee on Banking Supervision. *Minimum capital requirements for Market Risk*. Tech. rep. Bank for International Settlements, Jan. 2016. URL: https://www.bis.org/bcbs/publ/d352.pdf.


Electronic copy available at: https://ssrn.com/abstract=3203524