

# Information Loss During Size Reduction Depending On Structure Scale

Laurens Bogaardt<sup>1</sup>, Romulo Goncalves<sup>1</sup>, Raul Zurita-Milla<sup>2</sup>, and Emma Izquierdo-Verdiguier<sup>2,3</sup>

<sup>1</sup>NLeSC Amsterdam, The Netherlands `{l.bogaardt, r.goncalves}@esciencecenter.nl}`

<sup>2</sup>Faculty of Geo-Information Science and Earth Observation (ITC), University of Twente, the Netherlands  
`{{r.zurita-milla,e.izquierdoverdiguier}@utwente.nl}}`

<sup>3</sup>Image Processing Laboratory (IPL), Universitat de Valencia, Spain

## ABSTRACT

The analysis of large datasets can be time consuming and costly. Often, techniques exist to arrive at the same output, or at a close approximation, which require far less effort. This article looks at several such techniques and at the inherent scale of the structure within the data. When the values of a dataset vary slowly, e.g. in a spatial field of temperature over a country, there is a high level of auto correlation and the structure of the field has a large scale. Datasets need not have a high resolution to describe such fields faithfully. Using generated *Gaussian Random Fields* with various levels of spatial auto correlation, we examine several exact and approximate analysis techniques. Our aim is to find a relation between performance and the scale of the structure described by the input datasets.

## 1. INTRODUCTION

### 1.1 Gaussian Random Spatial Field

The singular values of a zero-mean matrix represent the amount of variance in the original dataset explained by the associated singular vector. The `explainedVariance` function takes the list of singular values and returns a list of explained variance. This can e.g. be used in a plot. [5] [6]

The  $\alpha$  is a measure of the spatial autocorrelation of the field. In particular, fields with large scale structure have a more negative  $\alpha$ . Additionally, the size of the singular values can be plotted. This is related to the power spectrum and will also provide information about the scale of the structure in the field. Finally, there are other measures of spatial autocorrelation. These include the Gamma Index and Moran's I. The Python module `pysal` contains functions to calculate these values. We will include them for each plot.

When  $\alpha$  is less negative, the Gaussian Random Field has smaller scale structure. It is closer to randomness and has less autocorrelation. As expected, this can be seen in the two measures of autocorrelation, which are now smaller.

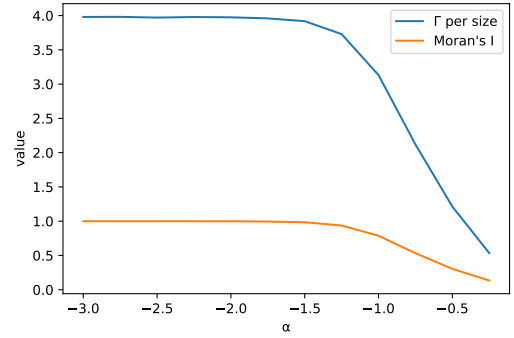


Figure 1: Caption

### 1.2 Gaussian Random Spatial-Temporal Field

In many real-world applications, the analysis of a field does not only involve a single time snapshot. In fact, it often includes data over multiple weeks, months or years, where the field over the entire time period does not change drastically. Just as there is spatial autocorrelation, there is temporal autocorrelation. In principle, there can be a different level of correlation over time than over space. However, for simplicity, we are using the same  $\alpha$  to determine the level of autocorrelation in all dimensions.

### 1.3 CCA and MCA Analyses

In an MCA, modes are found where the left- and the right-field covary maximally [2]. One technique to find these modes is to perform an SVD on the product of the datasets. That is why the term SVD is often used synonymously with MCA. In an CCA, modes are found where the left- and the right-field correlate maximally. In practice, the datasets are first standardized before multiplied and decomposed. MCA and CCA are very much related, but the resulting modes and vectors need not be the same [2]. In a PCA, the first and the second dataset are identical. Whether to maximize the covariance or the correlation depends on whether the units of each vector is the same.

## 2. TECHNIQUES

This section will discuss four techniques to analyse large datasets efficiently using SVD.

### 2.1 Exact Norm Difference via SVD

In real-world application, one often wants to find the norm of the difference between two fields. This can be done by subtracting one matrix from the other and calculating the norm. However, for large matrices, this can be inefficient. Performing an SVD of both matrices can reduce the internal calculations. Figure 2 shows a procedure which takes such input SVDs and determine the norm of their difference in an efficient manner, provided the number of singular values is small. The result is mathematically identical to the full calculation, which means that any error will be of the order of machine-precision.

$$\sum_{i,j} \left\| \begin{matrix} A \\ B^T \end{matrix} \right\|$$

Figure 2: Caption

For very large datasets, the SVDs may be obtained using a random algorithm reviewed extensively in an article by Halko et al. [3]. In section 2.4, we will also apply this technique to our generated fields. In this case, the resulting norm will no longer be exact, but the error can be made very small.

## 2.2 Exact SVD via QR Decomposition

In real-world applications, one often wants to find the relation between two fields. Analyses such as the MCA and CCA discussed in section 1.3 rely on performing an SVD of the cross-covariance or cross-correlation matrix of the two fields. In particular, the two input datasets often have the various gridpoints as rows and will have the sample of recorded values over time as columns. Multiplying these gives the cross-correlation matrix. However, for highly rectangular matrices, when there are many spatial gridpoint but few temporal samples, the resulting cross-correlation matrix is inefficiently large. The `qrProductSVD` function can take such input data and perform an SVD in an efficient manner. The result is mathematically identical to the full SVD, which means that the difference will be at machine-precision.

$$\begin{bmatrix} A \\ B^T \end{bmatrix} = \begin{bmatrix} R_A \\ R_B \end{bmatrix} \begin{bmatrix} Q_A^T \\ Q_B^T \end{bmatrix} = \begin{bmatrix} C \\ Q_A^T \end{bmatrix} \begin{bmatrix} Q_B^T \\ Q_A^T \end{bmatrix} = \begin{bmatrix} U_C \\ Q_A \end{bmatrix} \begin{bmatrix} S_C \\ V_C^T \end{bmatrix} = \begin{bmatrix} S_C \\ V_C^T \end{bmatrix}$$

Figure 3: Caption

## 2.3 Approximate SVD via Spatial Coarsening

Although the `qrProductSVD` function works well for two rectangular matrices, sometimes the input data is large and square. Performing an SVD on such a large dataset will be time consuming and, perhaps, inefficient given the desired level of precision. When a field have large scale structure, the values of neighbouring cells do not change drastically. This is what autocorrelation means. As such, perhaps neighbouring cells can be averaged together to produce a smaller dataset which still faithfully describes the original field. The `matrixToGrid` function can cut a matrix into multiple smaller sections.

The `testProductSpatialTemporalFieldsViaCoarsening` function determines the percentage of the information lost in a coarsening process for matrices of various sizes and  $\alpha$ 's, and at various levels of coarsening. The two input matrices used here are similar, as they

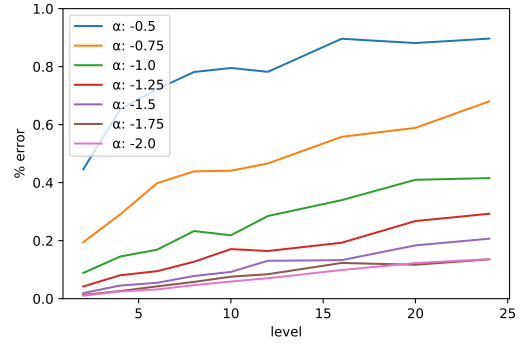


Figure 4: Caption

are generated by the same Gaussian Random Process. Therefore, they will correlate highly and the bases in which they are best described will be similar. In principle, any two datasets can be analysed and the amount of information lost during the coarsening process will likely depend on the similarity between the two datasets. This is one aspect which we do not cover here and leave for further research.

Due to the multiplication step in this analysis, the typical error due to coarsening is larger than before. As before,  $\alpha$  plays an important part, with more negative  $\alpha$ 's leading to a less dramatic loss in information due to coarsening.

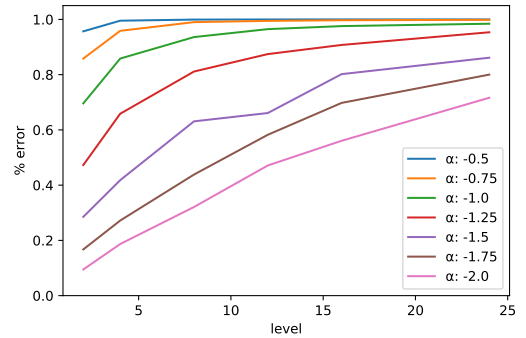


Figure 5: Caption

The additional benefit of coarsening is that data collection can be optimised.

## 2.4 Approximate SVD via Dimensionality Reduction

The spatial coarsening process is intuitive and easy to implement. It is not, however, the most efficient way to reduce the size of a dataset. Dimensionality reduction refers to discarding modes which contribute little to the variance in a dataset. An SVD is precisely the procedure used to find modes which explain as much variance as possible. Discarding the smallest singular values/vectors is, therefore, the most efficient form of dimensionality reduction. Performing an SVD on a large dataset, however, is compu-

tationally costly. The Randomised Dimensionality Reduction process is far more efficient.

The `reduceSizeRandomisedSquare` function reduces the input matrix to a smaller square matrix of  $l$  by  $l$ . It also gives two projection matrices which can bring the rows and columns of this smaller matrix back to the bases of the original input. To make the result more precise, the procedure can be repeated multiple times. The parameter  $i$  indicates how many loops are performed.

$$A = H \begin{bmatrix} L & W^T \end{bmatrix} = H \begin{bmatrix} \tilde{U} & S & \tilde{V}^T \end{bmatrix} W^T = \begin{bmatrix} U & S & V^T \end{bmatrix}$$

Figure 6: Caption

As seen, it is possible for some fields to be represented by matrices of much smaller sizes without losing any substantial amount of information. This is obvious when one realises the singular modes which are removed during the reduction are the smallest ones, described by the tail-end of the power law. In the review article by Halko, Martinsson and Tropp on randomised dimensionality reduction, it is suggested to oversample the reduction. This is because the error introduced in the process is of the same order as the size of the last sampled singular value. If one is interested in the  $k$  dominant modes, reducing to a  $k + l$ , for some small  $l$ , rank approximation will ensure the first  $k$  modes are approximated quite well. Indeed, as seen below, the more modes one is interested in, the larger the difference compared with the original matrix.

The Randomised Dimensionality Reduction process can also be applied to the CCA or MCA analysis of two spatial-temporal fields. Similar to the QR Product SVD, it has the advantage that the SVD is applied to a small  $l \times l$  matrix. The result will be an approximation, but, as we will see, can be close to the real solution.

$$A \begin{bmatrix} B^T \end{bmatrix} = \begin{bmatrix} U_A & W_A \end{bmatrix} \begin{bmatrix} L_A & W_A^T \end{bmatrix} = \begin{bmatrix} U_A & S_A & V_A^T \end{bmatrix} \begin{bmatrix} W_A^T \end{bmatrix} = \begin{bmatrix} U_A & S_A & V_A^T \end{bmatrix} \begin{bmatrix} W_A^T \end{bmatrix} = \begin{bmatrix} U_A & S_A & V_A^T \end{bmatrix} \begin{bmatrix} W_A^T \end{bmatrix}$$

Figure 7: Caption

To see the effect of dimensionality reduction on such a matrix product, let's generate two Gaussian Random Fields and plot their cross-correlation matrix together with a reduced version. To determine precisely how much information is lost during the reduction, we should look at the variance of the datasets. The norm of the difference between the reduced matrix and the original is the amount of information lost in the reduction process.

The `testRandomisedSizeReducedMatrixProduct` function determines the percentage of the information lost in a reduction process for matrices of various sizes and  $\alpha$ 's, and at various levels of reduction. The two input matrices used here are similar, as they are generated by the same Gaussian Random Process. Therefore, they will correlate highly and the bases in which they are best described will be similar. In principle, any two datasets can be analysed and the amount of information lost during the coarsening process will likely depend on the similarity between the two datasets. This is one aspect which we do not cover here and leave for further research.

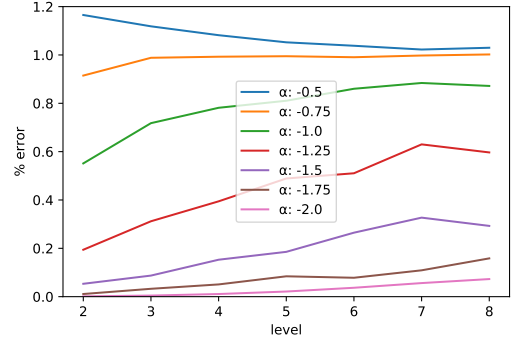


Figure 8: Caption

Due to the multiplication step in this analysis, the typical error due to reduction is larger than before. There is an affect of temporal size on the information loss. The `randomisedSquareProductSVD` function is especially useful for square input matrices. To be able to do the comparisons with the full SVD quickly, we use rectangular matrices here. The larger the temporal size, the more square the input matrices. As before, the scale of the structure of the field influences the information loss. The effect in this case can be quite dramatic. Especially for more negative  $\alpha$ , this procedure performs much better than the coarsening procedure.

Note that, unlike the coarsening procedure, the reduction is not applied on each time-slice of the spatial field, but rather on the spatially-flattened time-series. Therefore, the level of spatial auto correlation may not be as important as the level of temporal auto correlation. The proper analysis of this is left for further research.

The reduction of the number of dimensions of each input dataset is actually advised by some researcher. It has been suggested that filtering data for noise is important before the statistical analysis [1]. This is because any statistical analysis will choose its regression-coefficients so as to optimize the fit. It may occur that two noise-vectors in the two fields coincidentally covary. In a CCA, where the fields are standardized, this resulting mode may appear important even though it stems from noise. In an MCA, the variance of the noise will be low, so the chance that it will appear as an important mode is less [2]. One method of finding the right level of filtering is by bootstrapping/cross-validating the results [4].

### 3. APPLICATIONS

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

#### 3.1 Approximate SVD via Spatial Coarsening

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley

of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

### 3.2 Approximate SVD via Dimensionality Reduction

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

## 4. FURTHER QUESTIONS

It may be interesting to extend this research to fields other than the Gaussian Random Field. This type was chosen because its structure scale can be captured in a single parameter  $\alpha$ . In many applications, however, the dataset does not resemble such a Gaussian Random Field.

Additionally, it would be an improvement to relax the assumption that the auto-correlation in the time direction is similar to that in the spatial directions. In fact, it may even be more realistic to have different levels of auto correlation in the  $x$  and in the  $y$  direction.

## 5. REFERENCES

- [1] T. P. Barnett and R. Preisendorfer. Origins and levels of monthly and seasonal forecast skill for united states surface air temperatures determined by canonical correlation analysis. *Monthly Weather Review*, 115(9):1825–1850, 1987.
- [2] C. S. Bretherton, C. Smith, and J. M. Wallace. An intercomparison of methods for finding coupled patterns in climate data. *Journal of Climate*, 5(6):541–560, 1992.
- [3] N. Halko, P. G. Martinsson, and J. A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2):217–288, 2011.
- [4] R. E. Livezey and T. M. Smith. Covariability of aspects of north american climate with global sea surface temperatures on interannual to interdecadal timescales. *Journal of Climate*, 12(1):289–302, 1999.
- [5] P. A. P. Moran. Notes on continuous stochastic phenomena. *Biometrika*, 37(1/2):17–23, 1950.
- [6] S. Rey. Pysal. <http://pysal.readthedocs.io>, 2009–2013.