Temperature Forecasting with RPCA and ADM

Moajjem Hossain Chowdhury & MD Nazmul Islam Shuzan

Contents

Introduction

In this report, we will be working on applying Dynamic Mode Decomposition (DMD) on a dataset that contains the Average Annual Temperature from 1960 to 2019. We will also apply DMD to a synthetic dataset that we have generated in MATLAB. Both of these datasets have spacio-temporal characteristics, the types of data on which DMD excels on. Since real life data is always noisy, we will use Robust Principal Component Analysis (RPCA) with Alternating Directions Method (ADM) to filter the data.

RPCA with ADM

$$X = L + S$$

This is the equation on which RPCA is based on. The idea is that a data, X, can be decomposed into two matrices:

- L, a low-rank matrix
- S, a sparse matrix

And in S we will have all the oscillations and noise in the data and in L we will have the main information. The idea of calculating L and S hence becomes a minimization problem. There are many ways of doing this. One of the ways that we have already tested is RPCA with Inexact Augmented Lagrange Multiplier. Another way also uses RPCA but with ADM. This is the method that we will be using in this report. To apply this algorithm, we need two helper functions : shrink and SVT.

Shrink is a simple function. It takes two parameters as input : X and tau. X can be any matrix and tau is the threshold. It first checks each of the elements in X and compares its absolute value with tau. If the value is greater than the element is replaced with the difference between its absolute value and tau. Otherwise it is replaced with zero. The resulting matrix is then multiplied by the sign of the original matrix X. The final product is then returned as an output.

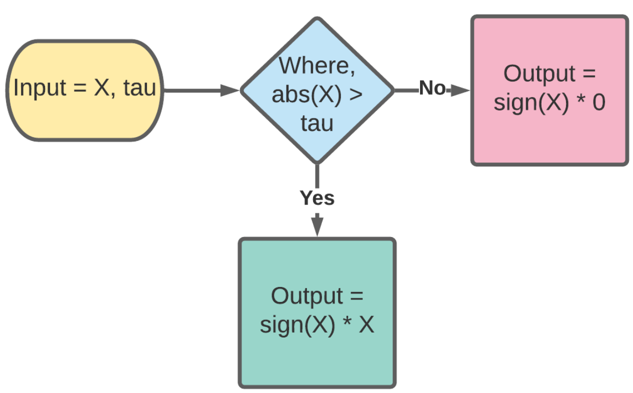

Figure 1: Process of Shrink of Algorithm

SVT means Singular Value Threshold. In SVT, we take a matrix X and tau as input. The matrix X is first decomposed into its component using SVD. We will then use shrink operator on the Singular matrix. This will cause many small values in the Singular Matrix to become zero. The product of the U V and the modified S matrix is returned. The implication of this is that the X matrix will become more low-ranked.

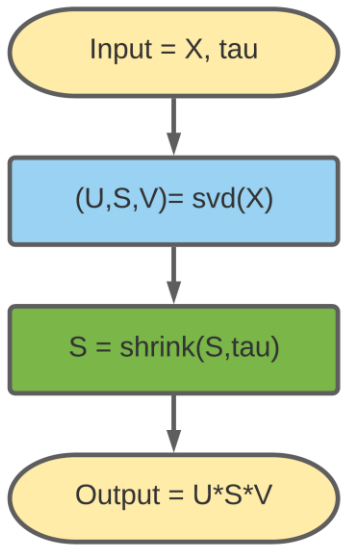

Figure 2: Process of SVT Algorithm

In the RPCA with ADM algorithm, first we need to initialize some values such as mu, lambda and threshold. We will also initialize L S and Y matrix as zero matrix. Then we will create a loop which will only end if the norm of (X - L - S) is less than the threshold. In the loop we will calculate L, S and Y. L is calculated by using SVT on (X - S + Y/mu). This causes L to have a matrix that will be low ranked as SVT does that. We then calculate S by using shrink on (X - L + Y*lambda/mu). This will cause many of the elements in S to become zero if they are below the threshold value. As a result, the matrix will be sparser. Y is simply updated with a simple expression where a large (X - L - S) will make a larger Y. The process is repeated iteratively until we get an L and S that satisfies X = L +S.

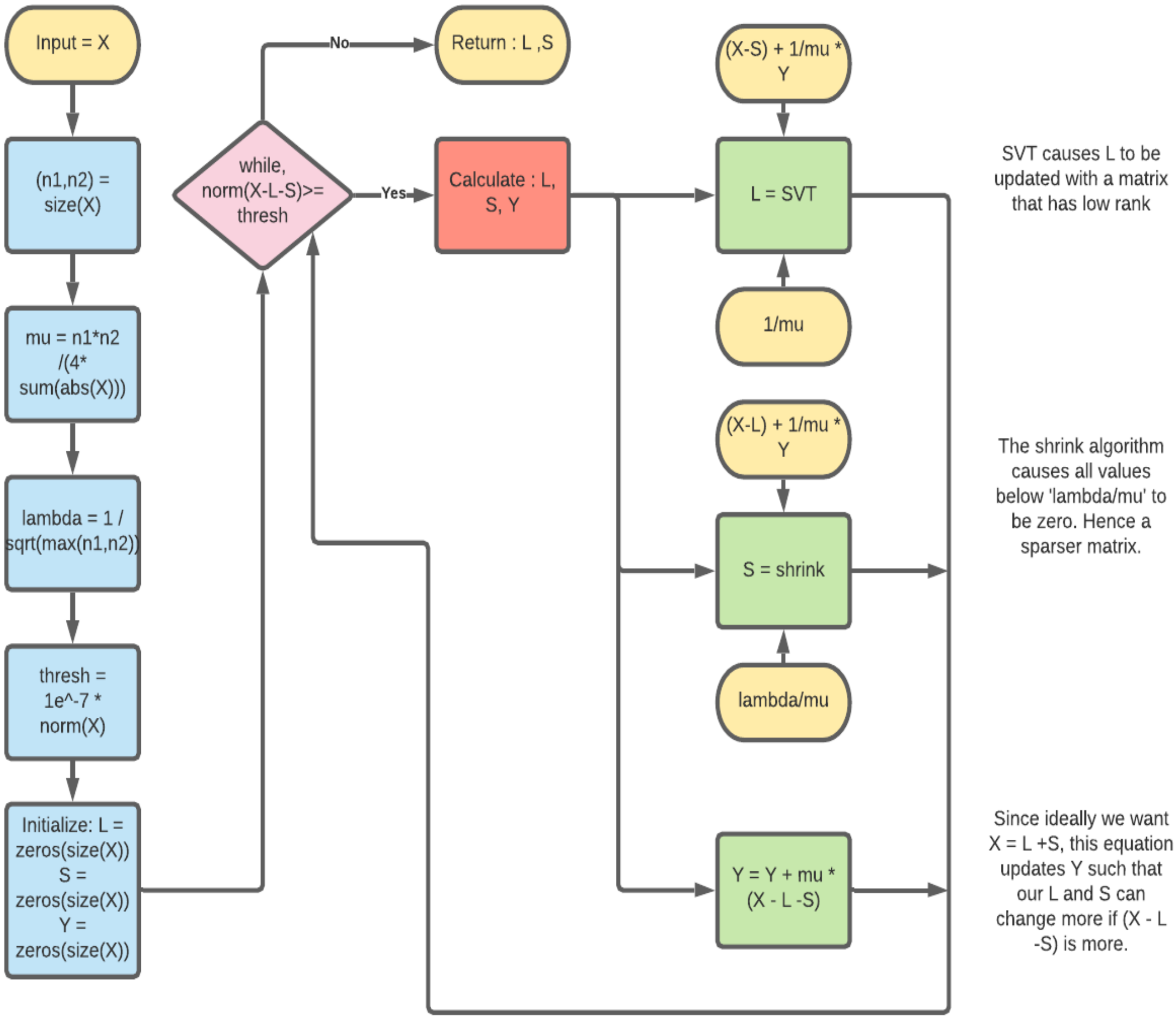

Figure 3: Flowchart RPCA with ADM

Dataset Description

In **Figure 4**, we have created a synthetic dataset with the equation:

$$f(x,t) = 0.5\sin(x) + 2\,\mathrm{csch}(x)\tanh(x)\exp(i2.8t)$$

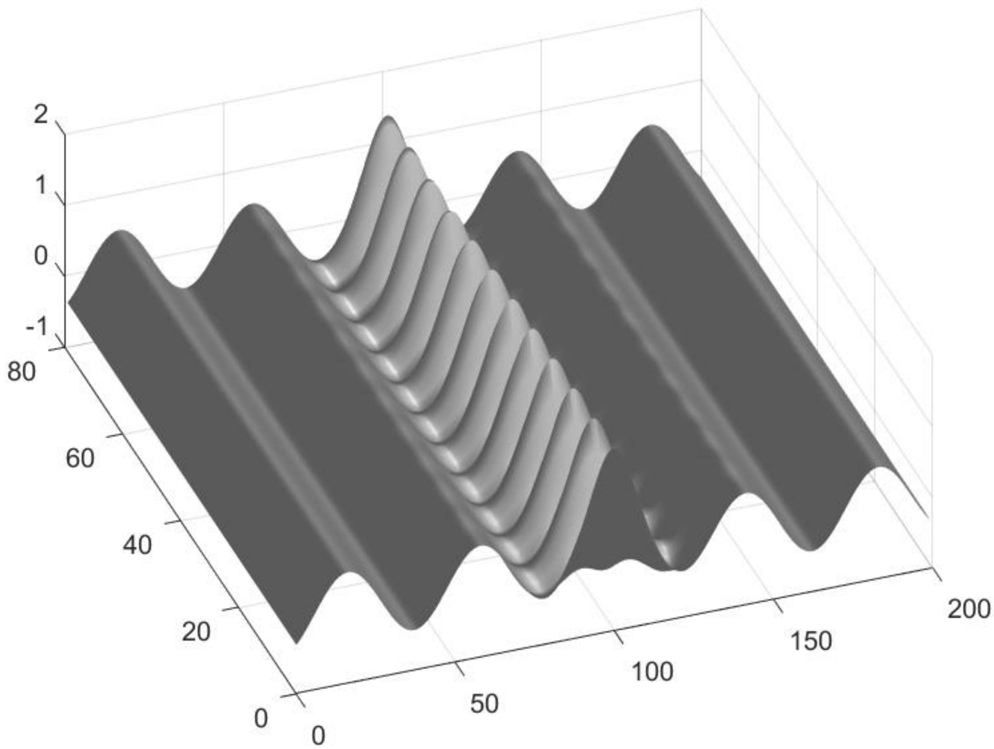

Figure 4: Plot of Synthetic Data

In **Figure 5,** we can see the distribution of the data from 1960-2019. The temperature has a general trend of going upwards. The data from 1960 to 2010 is for training and the data from 2011 to 2019 is for testing.

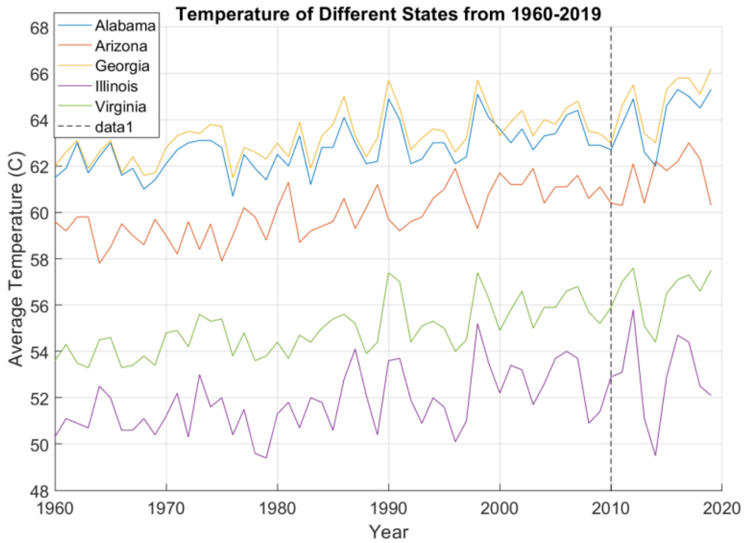

Figure 5: Annual Average Temperature from 1960-2019

Methodology

In this project, we first filtered the data using RPCA algorithm ADM. This will allow us to remove the noise from the data. This process ensures that the model created will be able to capture the true characteristics of the data.

Figure 6: Overall Flowchart for Training DMD Model

The data is then fed to a DMD model that will extract its characteristics and will give us a DMD model that will be able to predict the annual average temperatures in Alabama, Arizona, Georgia, Illinois and Virginia. The model will then be used to reconstruct our training data (1960-2010) and to predict the temperature from 2011-2019 (testing data). We will then compare these projections with our actual data and analyze them.

Implementation

Synthetic Data:

In **Figure 7**, we can see the plot after white noise has been added. The data with white noise was filtered using RPCA with ADM to get two data: Low rank data and Sparse data. We can see that the low rank data is very close to the original data.

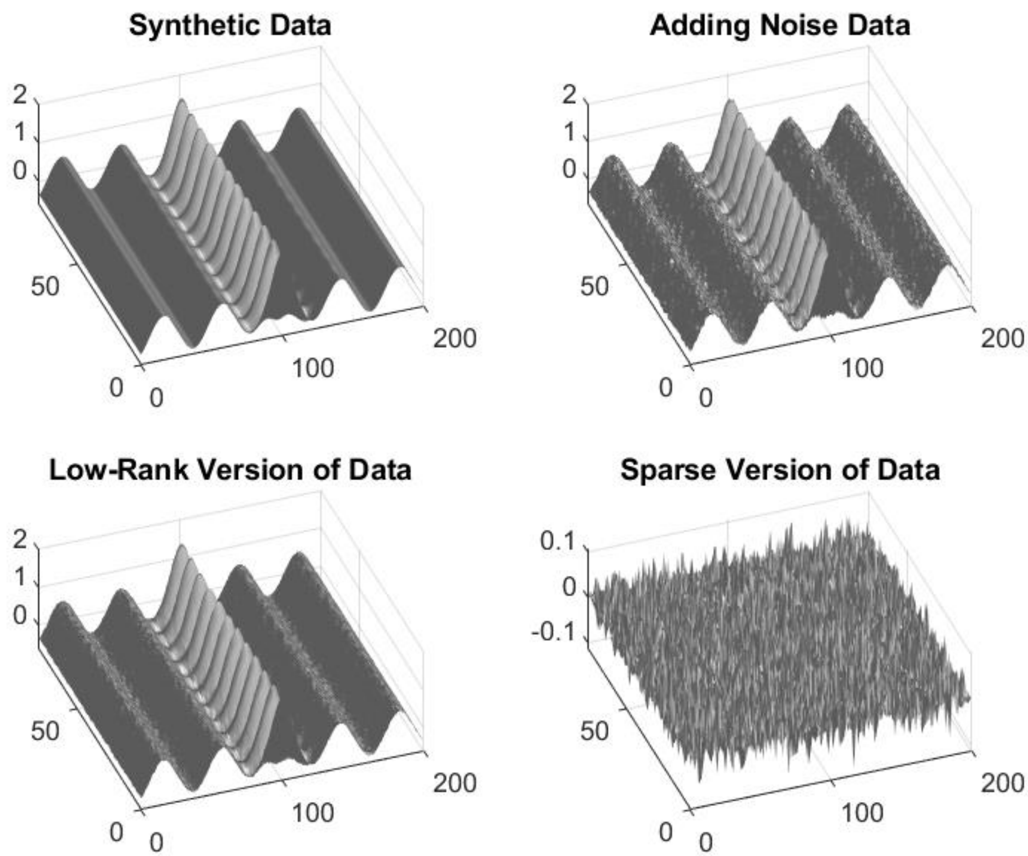

Figure 7: *RPCA with ADM on Noisy Synthetic Data*

In **Figure 8** shows the SVD plot of the training data. We can see that even a rank-2 truncation will have most of the information.

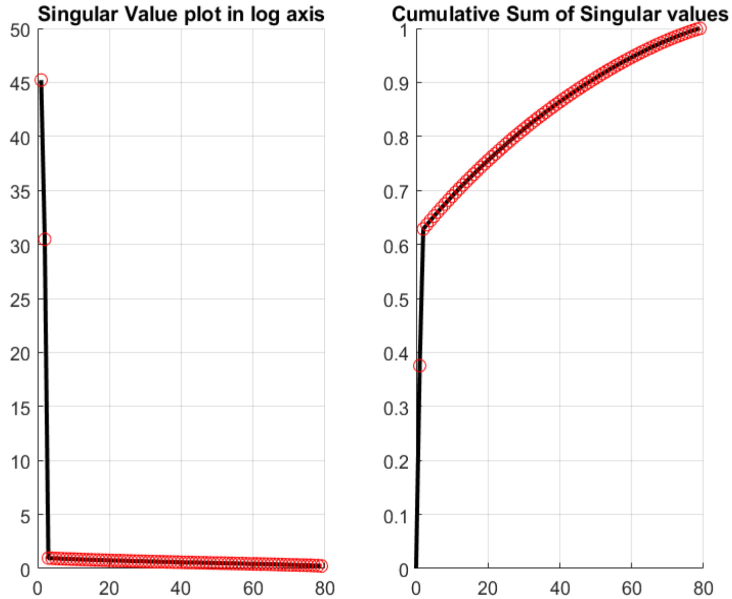

Figure 8: Singular Value Plot of Synthetic Data

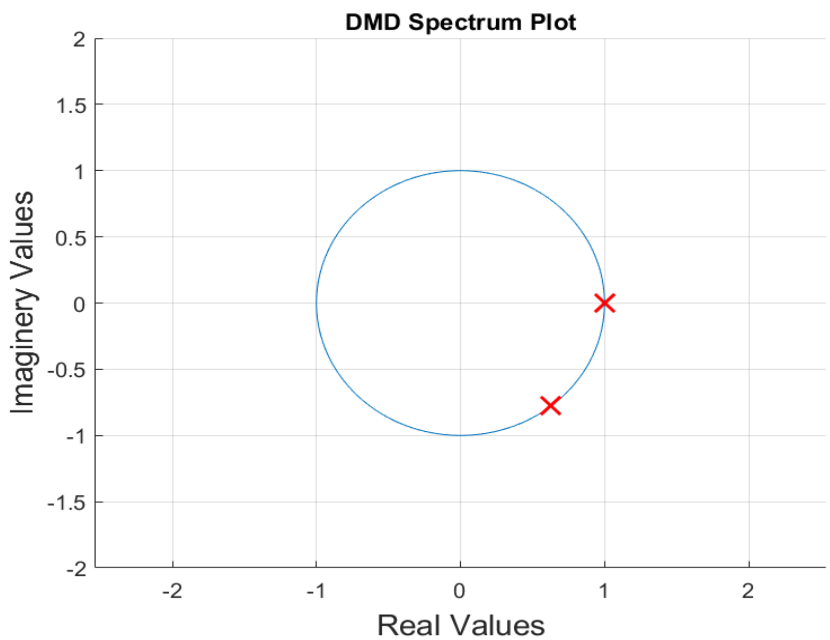

Figure 9: DMD Spectrum Plot of Synthetic Data

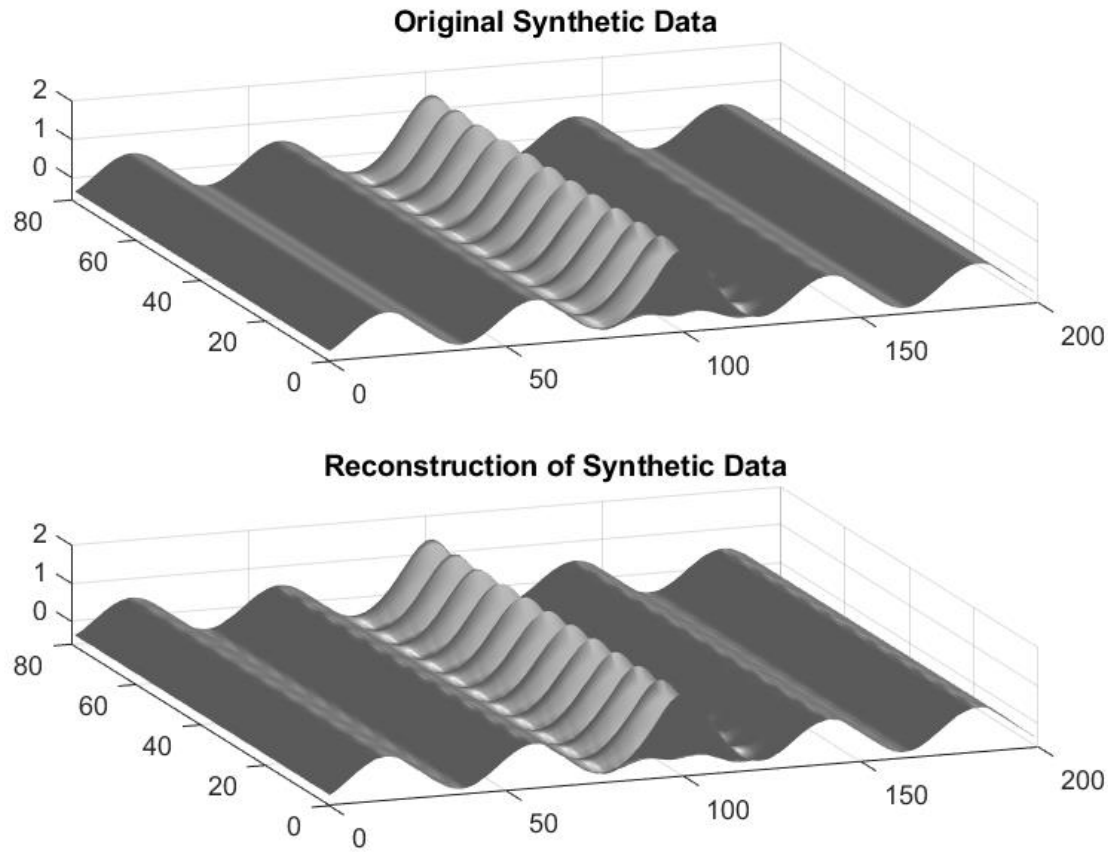

Figure 10: DMD Reconstruction for Synthetic Data

In **Figure 10,** we can see that our DMD model perfectly reconstructed the synthetic data

Result

Table 1: Performance Analysis of Filter-RPCA with ADM and DMD Reconstruction

	RMSE	CC
Filter-RPCA with ADM	0.015	0.99
DMD Reconstruction	0.005	0.99

Table 2: Rank of Synthetic Data after Each State

	Noisy Data	Denoised Data	DMD Reconstruction
Rank	80	59	2

Temperature Data

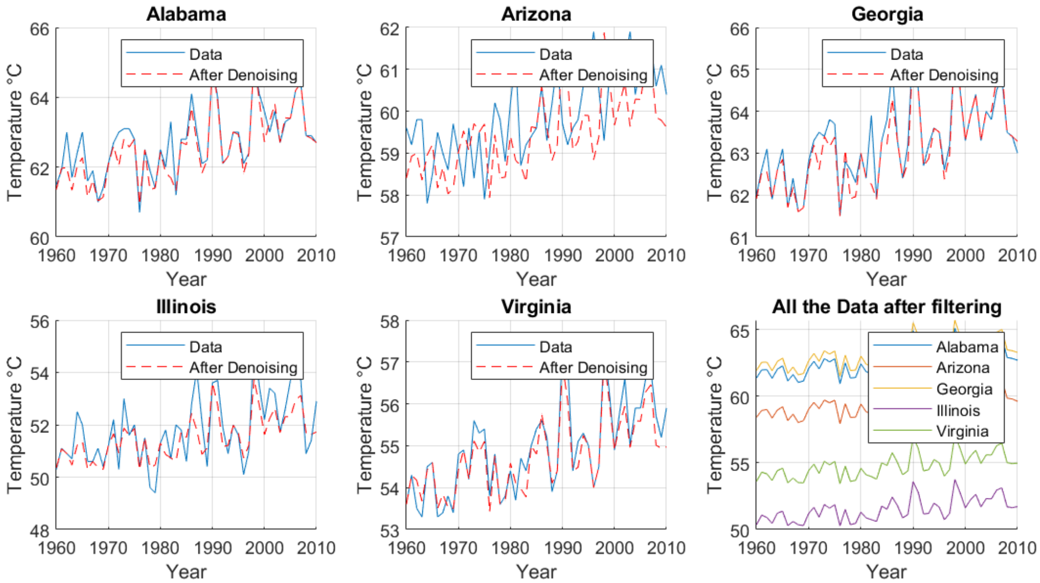

Figure 11: Applying RPCA with ADM to filter the noise

The data from 1960 to 2010 has been passed through a filter. The denoised signal and the raw signal is shown in **Figure 11**.

Normal DMD

In **Figure 12** shows the SVD plot of the training data. We can see that even a rank-1 truncation will have most of the information.

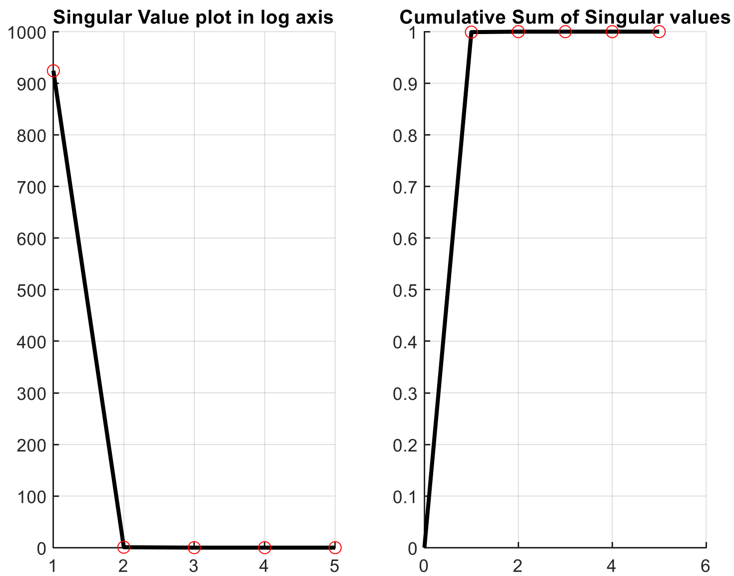

Figure 12: Singular Value of the Training

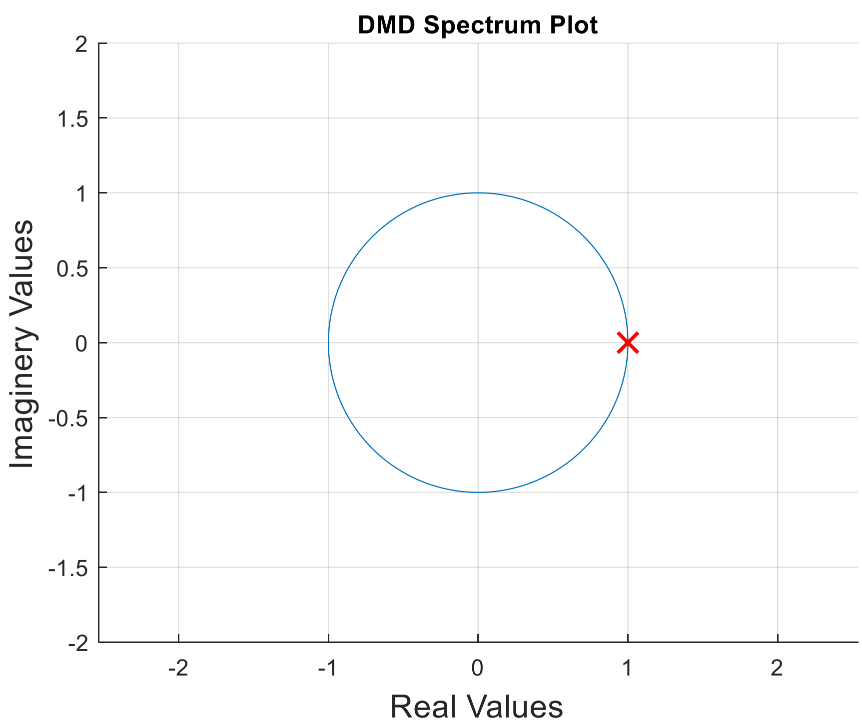

Figure 13: DMD Spectrum Plot

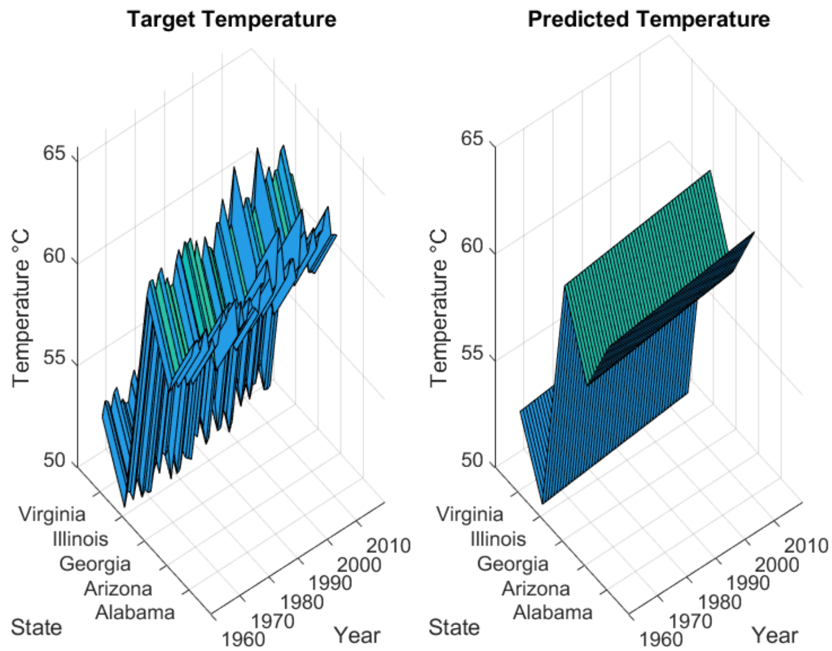

Figure 14: DMD Reconstruction for the year 1960-2010

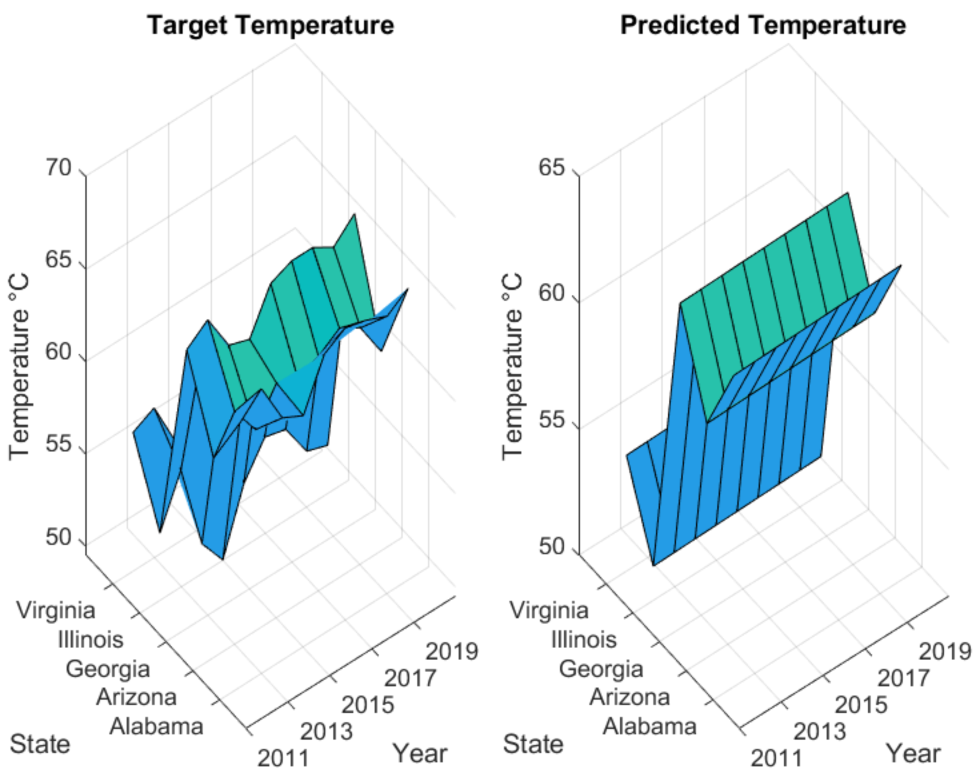

Figure 15: DMD Reconstruction for the year 2011-2019

Result

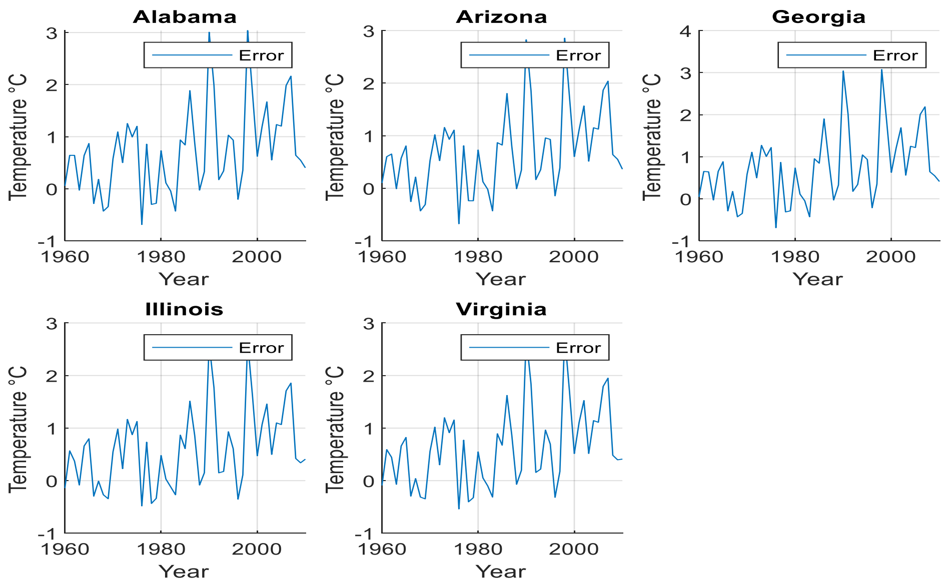

Figure 16: Error Plot for Training Data

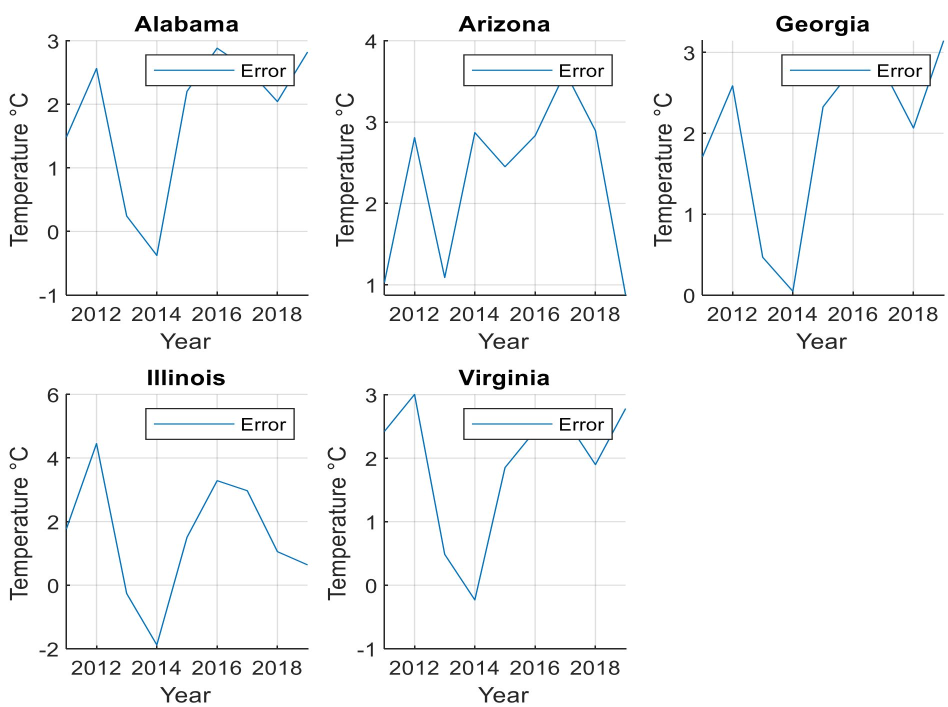

Figure 17: Error Plot for Testing Data

Table 3: Performance Evaluation of DMD Model on Training Data

STATE	RMSE	CC
Alabama	1.09	0.64
Arizona	1.03	0.64
Georgia	1.11	0.63
Illinois	0.95	0.61
Virginia	0.99	0.62
ALL	**1.03**	**0.98**

Table 4: Performance Evaluation of DMD Model on Testing Data

STATE	RMSE	CC
Alabama	2.13	0.49
Arizona	2.46	0.26
Georgia	2.23	0.52
Illinois	2.35	-0.04
Virginia	2.17	0.25
ALL	**2.27**	**0.96**

Table 5: Rank of Temperature Data in Normal DMD

	Noisy Data	Denoised Data	DMD Reconstruction
Rank	5	2	1

Delay Coordinates

In **Figure 18** shows the SVD plot of the training data. We can see that even a rank-2 truncation will have most of the information.

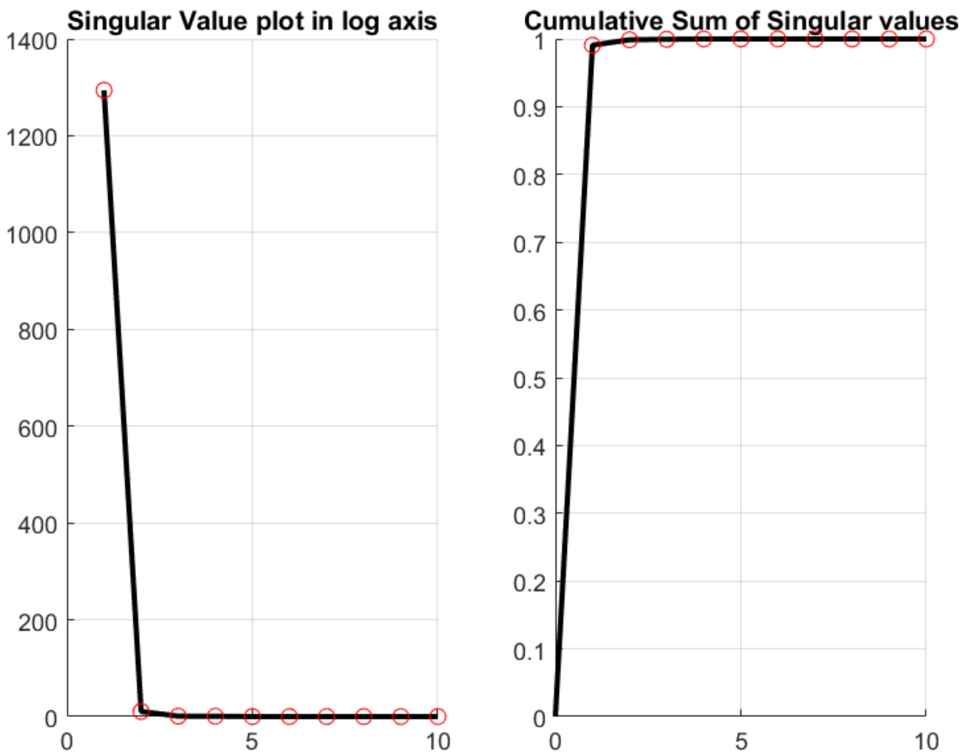

Figure 18: Singular Value of the Training Data

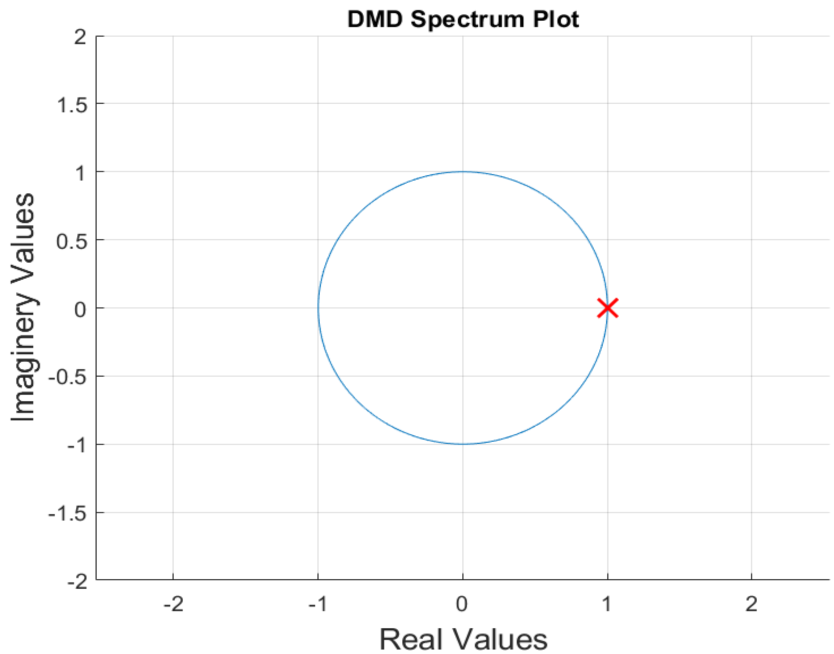

Figure 19: DMD Spectrum

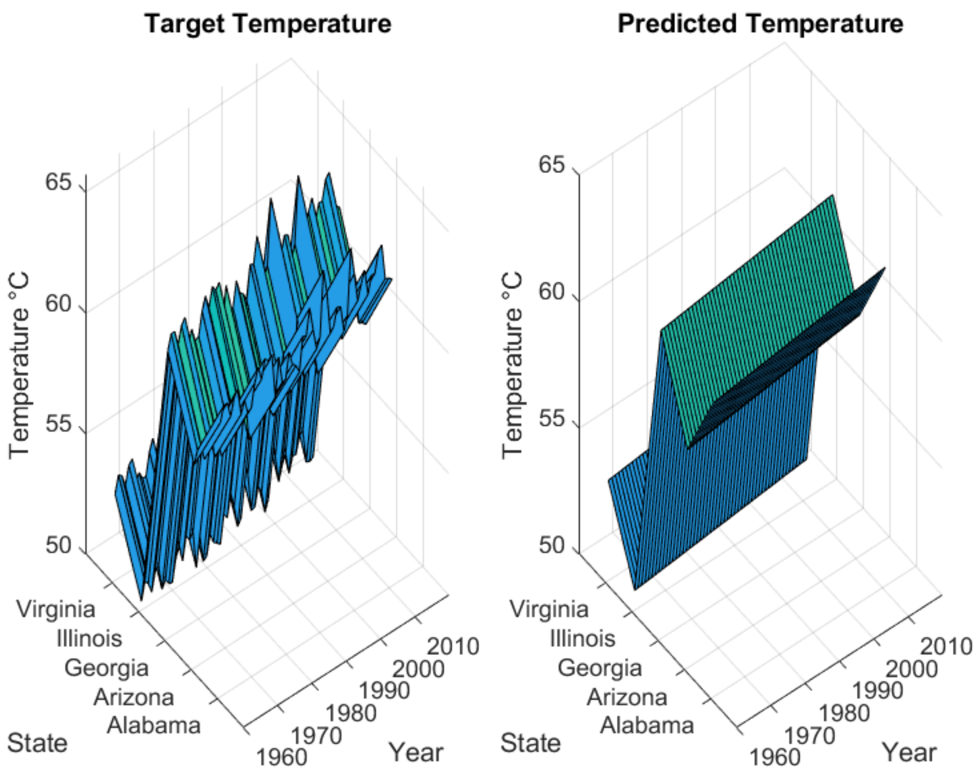

Figure 20: DMD Reconstruction for the year 1960-2010

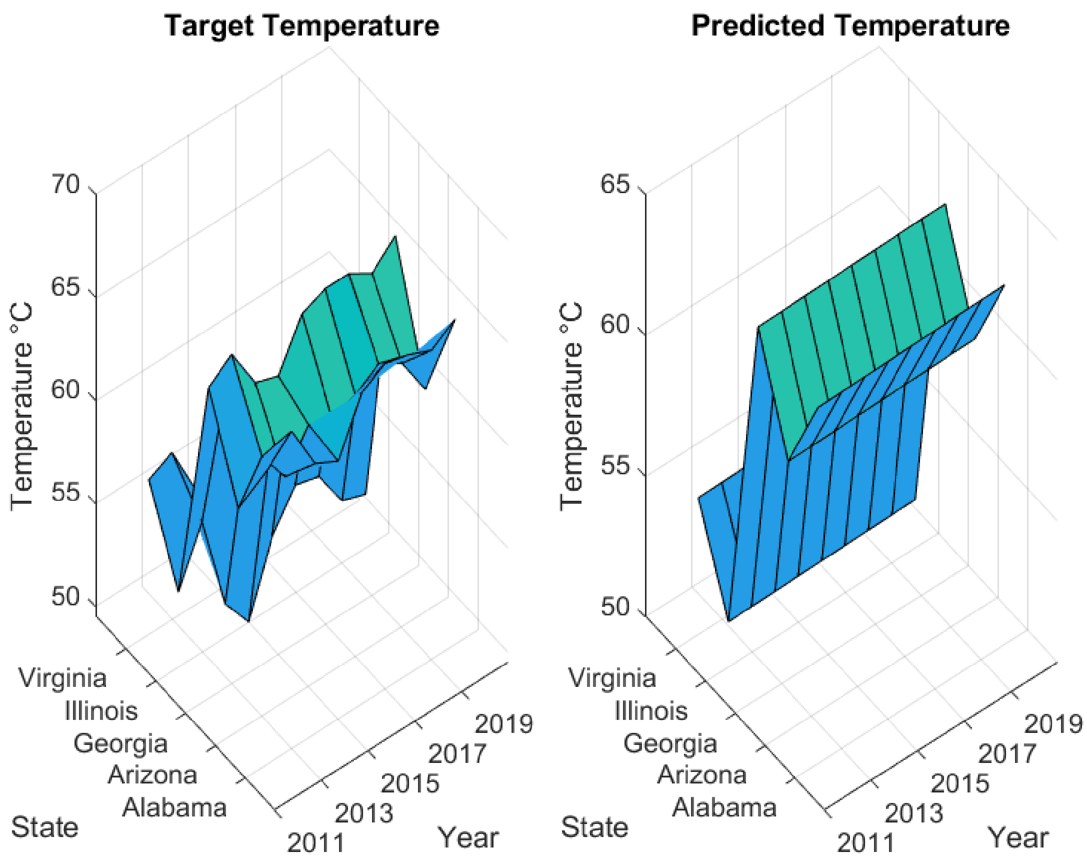

Figure 21: DMD Reconstruction for the year 2011-2019

Result

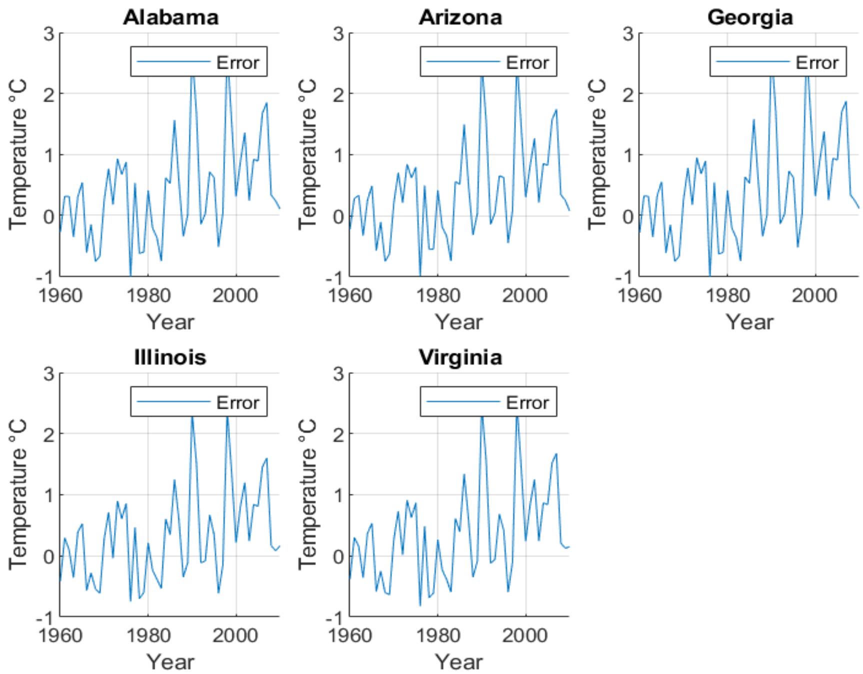

Figure 22: Error Plot for Training Data

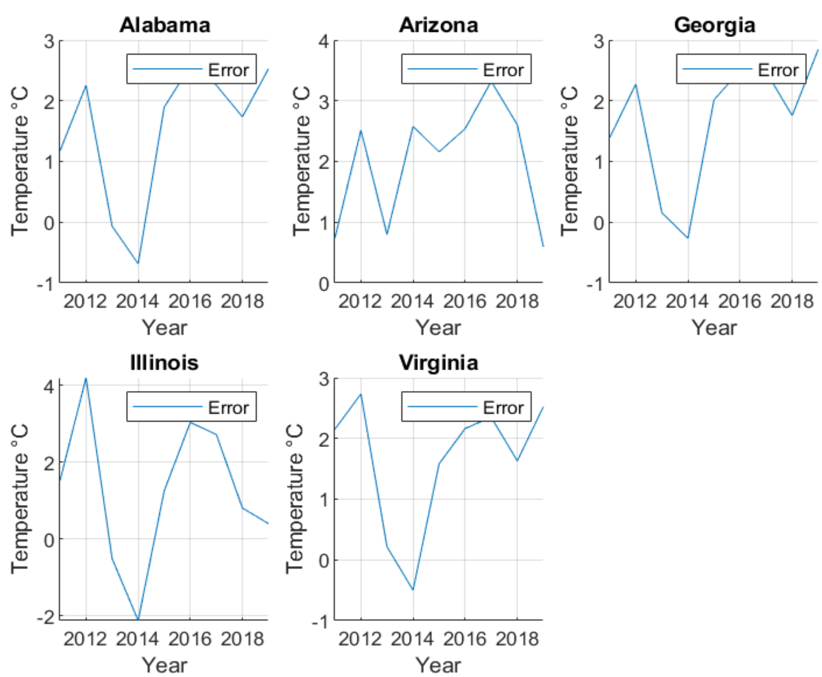

Figure 23: Error Plot for Testing Data

Table 6: Performance Evaluation of DMD with Delay Coordinate Model on Training Data

STATE	RMSE	CC
Alabama	0.92	0.63
Arizona	0.86	0.64
Georgia	0.93	0.63
Illinois	0.81	0.61
Virginia	0.84	0.61
ALL	**0.87**	**0.98**

Table 7: Performance Evaluation of DMD with Delay Coordinate Model on Testing Data

STATE	RMSE	CC
Alabama	1.87	0.49
Arizona	2.19	0.30
Georgia	1.96	0.51
Illinois	2.20	-0.04
Virginia	1.94	0.24
ALL	**2.04**	**0.96**

Table 8: Rank of Temperature Data with Delay Coordinates

	Noisy Data	Denoised Data	DMD Reconstruction
Rank	5	2	1

Discussion

When looking at the synthetic data, we can see that the filter and DMD were very successful. The noisy data had a rank of 80 which after denoising became 59. DMD was able to reduce that to 2. Furthermore, we can see that the DMD reconstruction has a very low RMSE when compared to the real data. Hence we were successful in that regard.

In our implementation of DMD on the Average Annual Temperature, we were able to achieve some success. The real data (all real life data are noisy by default) had a rank of 5 which was reduced by the filter (RPCA with ADM) to 2 and DMD was able to reconstruct the data with a rank of 1. This was the case for both normal DMD and DMD with delay coordinates which was remarkable. Both the normal DMD and DMD with delay coordinates had an RMSE of 2.51 and 2.04 respectively. The results were very good.

However when we compare these results with our previous work where we used DMD and RPCA with Inexact ALM, our previous work outperforms this work. We had previously achieved an RMSE of 1.70 with the test data.

Conclusion

From this report we can see that RPCA with ADM is not that good in filtering the data. But in the theory from which we took it from had great result. The inference that we can draw here is that it works well with image data. Furthermore, we can see that delay coordinates can help us get a better DMD model. Furthermore, we have noticed that DMD itself has a capability of filtering data. This was shown in our previous work where we separated foreground and background. That algorithm worked on the same basis as RPCA : X = L + S.

So in our future work, we can apply delay coordinates on data with RPCA and Inexact ALM. We can also apply DMD on the raw data and see if the performance is comparable to the one with filter.

Appendix

dmd_syn_data.m : DMD on Synthetic Data

dmd_present.m : DMD with RPCA-ADM

dmd_delay_coordinates.m : DMD with Delay Coordinates on data filtered by RPCA-ADM