Data Driven Methods for Complex Systems

Rohit Mohan

November 29, 2016

Contents

List of Figures

List of Tables

Acknowledgements

First of all I would like to acknowledge the wisdom bestowed upon me by Dr K.P Soman. Without his encouraging words and the conversations we had both in class and over tea, I wouldn't have gotten over my fear of mathematical thinking.

I am also greatly indebted to Dr Gopalakrishnan E.A, for introducing me to the wonderful subject of nonlinear dynamics and chaos, and for his role as a guide and mentor for this project.

Further, I would like to thank Mr Vijayakrishna Menon, for the deep and enriching discussions we had on programming and philosophical topics, giving me many parallel ways to look at problem solving, and Dr D.Govind for knowledgable inputs regarding the art of presentation and paper-writing. Also I thank, Mr Sajith Variyar V.V for practical inputs.

Gratitude is in order, for the Ph.D scholars in the department, for their kind assistance, companionship and nourishment, during the late nights spent working in the department. Finally, I would like to thank my friends and family for being there to support me at all times.

Abstract

Data driven modelling is an emerging domain that aims at identifying system characteristics purely from data, and representing them in a mathematical framework. There are, in general two types of system identification frameworks - equation-free modelling and identification of governing equations. Dynamic Mode Decomposition (DMD) is an equation-free modelling technique, originally used in fluid dynamics community. Sparse Identification of Nonlinear Dynamics (SINDy), is a technique aimed at deducing governing equations from data. Recent research efforts into these techniques have produced promising results, establishing them as the primary techniques used for system identification. It is the objective of this work, to understand the workings of these two techniques by applying them on datasets, that are of practical and pedagogical importance. To this end, DMD is applied to real world rainfall data, to try and obtain the dominant dynamic modes, which can then be subject to further analysis. SINDy is used to predict the governing equations of supercritical and subcritical systems, which are normal forms of dynamics, that are simple, yet appear in many practical systems.

Chapter 1

Introduction

"The only constant is change" wrote Heraclitus of Ephesus. This proverb embodies the omnipresence of dynamical behaviour in every system, natural or man-made, whether it is of real or abstract existence. The most obvious common factor between all natural systems ranging from quantum mechanical to celestial regime, is change with respect to time. Human life itself is driven by constant change, with evolution shaping us up at a slow rate, and our own technological advancements altering our life at a much more rapid pace. Technology and financal systems, which are already affected by the dynamics of nature, have developed their own dynamics. This showcases the ability of the dynamical systems framework to bestow upon abstract and non-biological entities, a notion of 'life'. It can be seen that understanding the dynamical behaviour of a given system enables us to better understand, analyse, predict and even control the actions of the system. This endevour has already shown success, in revealing complex behaviour within simple systems (chaos theory), and showing that simple models can explain the behaviour of highly complex systems (complex systems theory). But some important domains, like weather, stock market, turbulence etc., have been extremely difficult to completely comprehend, let alone model. This has led research of dynamical systems into new avenues and contact with other fields like optimization theory, data science and machine learning to name a few [4] [5].

1.1 Difficulty in Analysis of Dynamical Systems

Building a satisfactory mathematical model is of prime importance in the study of any system. Further analysis of the system or a particular phenomenon exhibited by the system is then done on this model. Models may or may not be in the form of an equation. These mathematical models can be widely classfied into two. For a system that causes a transformation represented by A, if

$$A(\alpha_1 x + \alpha_2 y) = \alpha_1 Ax + \alpha_2 Ay \text{ (Superposition principle)}$$

The the system is said to be a linear system. These systems are intuitively more understandable and have been studied very extensively. As a result, the theory behind them rests upon the mathematically sound framework of linear algebra. However these systems are not of great interest in the study of dynamical systems. The more complex and fascinating patterns occur in the behaviour of systems that do not follow the superpoistion princple. They are called nonlinear systems. Though these systems have also been widely studied, the frameworks developed are not as complete as it is for linear systems. One of the approaches taken is to express nonlinear systems as infinite dimensional linear systems as with the Koopman operator theory [6].

Most dynamical systems of interest today are not just nonlinear, but they consist of a lot of components interacting with one another, producing an output that is qualitatively different from the summation of outputs from the individual components it is composed of. These types of systems are called complex systems. Complex systems have a multi-disciplinary origin. Mathematics has contributed the most to the study of chaos, quantum chemistry has provided foundation for the principle of emergence, while the theory of self-organization has it's origins in nonequilibrium thermodynamics, which are all part of complex system theory [15]. While these developments have conveyed the possibility of studying many systems of practical and philosophical interest, the nonlinearity denies access to a solid framework for carrying out analysis. But apart from that, the sheer complexity and number of components involved makes it difficult to arrive at a mathematical model of the system through plain observation and intuition. So, in order to analyse, predict or control these systems, methods have to be developed first to deduce an appropriate model for it.

1.2 Development of Machine Learning

Machine learning developed as a branch of artificial intelligence, but now has an existence of it's own. The major aim of machine learning is to make computers sense the presence of patterns prevalent in large amounts of data. The relevance of machine learning to the study of complex systems is immediately obvious. Since mathematical models are just a terse representation of some pattern of behaviour of the system, machine learning, in theory, should be able to deduce the mathematical model for a system, given enough data generated by the system.

Development of machine learning has been greatly driven by the outpour of data due to the advent of social media platforms, e-commercial platforms and reduction in price and size of sensors have greatly contributed to this. Along with this, the increase in processing capacity of computing machinery has greatly expanded the arsenal of techniques that scientists can consider when devising a learning algorithm. For example, machine learning frameworks like artificial deep neural networks (ADNN), still take months to produce results, even with advanced GPU and distributed computing support, but produce cutting-edge results in many application areas. Computation of this scope could not have been carried out 10-15 years back.

Machine learning itself can be braodly classified into two, regresssion-based learning and classification-based learning. Since most complex systems of interest have continuous valued output, regression-based learning plays a more significant role. Also, since the output data from the system is usually available, supervised learning is mostly chosen over unsupervised learning, for model building.

1.3 Commonly Used Tools of the Trade

One of the first successes of data driven methods can be attibuted to Takens embedding theorem [14]. It showed that complex behaviour of a system can ,under certain conditions, be modelled using data from time-series of a single measurement. Following this, mathematical models pertaining to dynamical systems have been of two types - statistical and geometric. More recently, a third approach relying on operator theory has been actively pursued [2].

Dynamic Mode Decomposition (DMD) and Sparse Identification of Nonlinear Dynmamics (SINDy) are two of the most recently developed tools that show promise. The exploration of these, is the main premises of this report. The growth of tensor (multi-array) methods are also interesting to watch for, since they allow more freedom of representation and manipulation of data than ordinary matrices do. New corelations could be found if data could be arranged in that manner (like exploiting both spatial and temporal closeness of data at the same time).

1.4 Organization of Report

This report outlines the basic theory and practical exploration of DMD and SINDy. To this end, section 2 glances at the historic development and relevant literature related to these two techniques. Section 3 deals with underlying theory and interpretation of these two techniques. Section 4 walks through implementation of computer experiments based on above mentioned techniques on some form of data, for validation of it's correctness. The section also shows relevant results. Section 5 briefs the takeaways from this project, and section 6 enumerates few of the possible avenues of research in this topic.

Chapter 2

Related Work

2.1 General discussion on data-driven methods

Till the advent of powerful computers, data-driven techniques were the mostly of concern to only statistisians. Computer science and artificial intelligence dealt mostly with evolutionary algorithms. Other branches of science followed the 'knowledge-based' approach, drawing from creativity, intuition and mathematical logic of the practitioners involved, to build models. Thankfully, the increase in computing power has kept up with the advent of big data in almost every field of science. This is one of the main reasons for the emergence of data-driven modelling. Data driven modelling derives techniques from areas like computational intelligence, machine learning and data mining, to build models that enhance or replace knowledge-based models. Optimization techniques like genetic algorithms or sparsity based techniques are used to fine-tune the models. This is important because parsimonious models are generally in tune with natural phenomena.

2.2 History of Dynamic Mode Decomposition(DMD)

DMD falls under a more general class of methods called modal decomposition. They are concerned with seperating data into relevant modes, which reveal patterns of interest. The predecessor to DMD, in many respects, is the POD method [8]. These methods have their origin in the fluid dynamics community, and have been used extensively to study various flows. The satisfactory results in the field, has kept the research for more sophisticated modal decomposition techniques alive.

DMD was introduced in 2008 in a conference by Schmid and Sesterhenn with an article following in 2010 [12]. It was soon interpreted in two different ways. Rowley et al. [10], approached the theory from the perspective of Koopman operator theory [6]. By considering DMD as a finite dimensional approximation of the infinite dimensional Hilbert space operator, they argued that nonlinear behaviour can be approximated by DMD. In contrast, Schmid et al. [12] took an approach that noted the similarity between DMD and the Arnoldi method. [12] also introduced an SVD based algorithm, which was much more stable than others. This is the standard DMD algorithm in use today.

2.3 History of Sparse Identification of Nonlinear Dynamics(SINDy)

SINDy falls under the category of system identification methods. Usually system identification methods try to assume a linear model of nonlinear systems, and thereby loose important information. For example, a linear system cannot have more than one attractor. SINDy builds on symbolic regression and sparse representation techniques to avoid this pitfall. The balance between complexity and exact representation is struck by following the parsimony argument, like Pareto front.

Symbolic regression finds functions that reate input data with the output, using machine learning techniques. This technique was first introduced to predict dynamic system equations by Schmidt et.al [13]. Since overfitting is a serious problem with these types of models, the parsimony argument is very important. Sparse representation is obtained using compressed sensing techniques [3]. This is a shift from previous approaches that used genetic algorithms to find a sparse representation [7].

Chapter 3

Theory

3.1 Theoretical Background for DMD

Let us consider dynamic systems of the form,

$$x_{k+1} = f(x_k) \tag{3.1}$$

where x_i is an n-dimensional vector of measurements, made at an instant i, and f(.) is some function that contains the dynamics of these measurements.

With evenly sampled, m measurement vectors, we can form two column shifted matrices,

$$X = \begin{bmatrix} | & | & \dots & | & | \\ x_1 & x_2 & \dots & x_{m-2} & x_{m-1} \\ | & | & \dots & | & | \end{bmatrix} \text{ and } X' = \begin{bmatrix} | & | & \dots & | & | \\ x_2 & x_3 & \dots & x_{m-1} & x_m \\ | & | & \dots & | & | \end{bmatrix}$$

The dynamics of the measurements used as data, should then be captured by a matrix A, such that,

$$X' = AX \tag{3.2}$$

The solution (A), for this equation can be obtained using pseudoinverse of matrix X, obtained from it's singular value decompositon(SVD) like this,

$$A = X'X^{\dagger} \tag{3.3}$$

A is the best-fit operator that represents the dynamics. Then, the eigendecomposition of A will give the required dynamic modes of the system, of which the measurements were taken. However, when number of measurements, $n >> 0$, the matrix A which will be of the size $n \times n$, is very large. Performing an eigendecomposition on this matrix is then infeasible. So an indirect method to find atleast the dominant eigenvalues and eigenvectors of A has to be employed. The commonly used method, which first appeared in [12] is as follows,

- SVD is performed on the data matrix X and an r-rank approxiamtion is made,

$$X \approx \tilde{U}\tilde{\Sigma}\tilde{V}^* \tag{3.4}$$

- Project the matrix A onto the column space of $\tilde{U}$ to obtain $\tilde{A}$(since we do not know A, we use the previous step to substitute for it),

$$\tilde{A} = \tilde{U}^* A \tilde{U} = \tilde{U}^* X' \tilde{V} \tilde{\Sigma}^{-1} \tag{3.5}$$

- Compute the eigendecomposition of $\tilde{A}$,

$$\tilde{A}W = W\Lambda \tag{3.6}$$

  The eigenvalues of $\tilde{A}$ is the same as the dominant eigen values of A.

- Reconstruct the dominant eigen modes of A (given by columns of Φ) using the eigen modes of $\tilde{A}$,

$$\Phi = \tilde{U}W \tag{3.7}$$

These eigen modes are the dynamic modes that capture the essential information about the system's change.

Moreover, reconstruction of the entire dataset, and also future predictions can be made by using the formula,

$$Y(t) = \Phi \Lambda^(t/\delta t) \Phi^\dagger X(0) \tag{3.8}$$

3.2 Limitations of DMD

From the Koopman operator theory perspective we can say that we are finding a finite dimensional approximation , A, for the infinite dimensional Hilbert space operator. This yields an accurate solution only when the dynamics of the measurements is confinied within a finite subspace. Otherwise, the result of the method holds only until the measurements eventually fall out of the subspace represented by A. Obviously, limitation to a finite subspace would undermine the assumption that the nonlinear behaviour is being captured in infinite space. So in essence, the method provides fully accurate results only for linear systems. Effort has been made to extend DMD to accomodate nonlinear dynamics better [16]. The usage of pseudoinverse in the algorithm causes loss of some accuracy due to projection into null space. Moreover, rank truncation of SVD also removes information.

3.3 Theoretical Background for SINDy

Let us consider a dynamical system of the form,

$$\dot{x}(t) = f(x(t)) \tag{3.9}$$

where $x(t) = [x_(t)x_2(t) \ldots x_n(t)]$ is a state vector, and $\dot{x}$ is the rate of change of state. If we assemble the values taken on by both $x(t)$ and $\dot{x}(t)$, we get two matrices,

$$X = \begin{bmatrix} x_1(t_1) & x_2(t_1) & \ldots & x_{n-1}(t_1) & x_n(t_1) \\ x_1(t_2) & x_2(t_2) & \ldots & x_{n-1}(t_2) & x_n(t_2) \\ \vdots & \vdots & \ldots & \vdots & \vdots \\ x_1(t_m) & x_2(t_m) & \ldots & x_{n-1}(t_m) & x_n(t_m) \end{bmatrix} \text{ and } \dot{X} = \begin{bmatrix} \dot{x_1}(t_1) & \dot{x_2}(t_1) & \ldots & \dot{x_{n-1}}(t_1) & \dot{x_n}(t_1) \\ \dot{x_1}(t_2) & \dot{x_2}(t_2) & \ldots & \dot{x_{n-1}}(t_2) & \dot{x_n}(t_2) \\ \vdots & \vdots & \ldots & \vdots & \vdots \\ \dot{x_1}(t_m) & \dot{x_2}(t_m) & \ldots & \dot{x_{n-1}}(t_m) & \dot{x_n}(t_m) \end{bmatrix}$$

From X, it is possible to build a library $\Theta(X)$ such that,

$$\Theta(X) = \begin{bmatrix} | & | & | & & | & | & \\ 1 & X & X^2 & \ldots & sin(X) & cos(X) & \ldots \\ | & | & | & & | & | & \end{bmatrix} \tag{3.10}$$

All functions of X should ideally be included into the library.

Now we say that,

$$\dot{X} = \Theta(X)\Xi \tag{3.11}$$

This denotes the fact that $\dot{X}$ is some linear combination of nonlinear functions of X. So, now Ξ needs to be found. The fact that most nonlinear systems only have a few nonlinear terms in their governing equation, allows us to use sparsity techniques to arrive at an answer. Here compressed sensing techniques like the LASSO, can be used.

3.4 Limitations of SINDy

The main limitation of SINDy is the fact that the functions that go into the library have to be carefully chosen. This requires the knowledge-based intuition, which is what is being tried to be replaced here. If the functions of required for the governing equation are not present, then a sparse representation won't be possible and the best-fit answer will be spread across almost all functions similar to spectral leakage in Fourier transforms. Sometimes, if a lot of extra functions are added, the method may wrongly identify components of those functions in the data.

Chapter 4

Methodology and Results

4.1 Rainfall data analysis using DMD

Rainfall data being used here was obtained from [9]. It consists of monthly rainfall data recorded from 36 base stations across india. The temporal dynamics of this dataset is studied using DMD. Note that since DMD works within the column space of the dataset (and also the truncation after taking the SVD), the dataset limits the DMD algorithm to only 35 time instances (in this case, months). This is because the dimensionality of the data matrix then limits the maximum possible rank of the matrix. So we resort to taking 35 month-chunks of the data and subject it to analysis using DMD (this can be improved with a dataset with higher spatial resolution). This is because DMD was originally formulated to study datasets which form "tall" matrices.

First, we extract data from the spreadsheet file and convert it to a numpy array for future convenience. It is then saved on the disk in pickle format. Now to perform temporal analysis of the data, we extract the time slices under consideration (following the constraints mentioned above). Then DMD algorithm is applied on this subset of data.

Critical information about the data, like the dataset's singular values, eigen values, dynamic modes and their time evolution are studied to look for trends. Short term prediction is also possible with this information. This is in tune with the fact that most nonlinear systems can have a local linear approximation, and DMD provides a representation for this approximation in A. To check if the dynamic modes we extracted truly model the dataset sufficiently, we reconstruct the data matrix with these modes. Visual similarity in the original and reconstructed data matrix can confirm that the dynamic modes represent the trend in original data.

First we submit data for 100 months to DMD algorithm, and analyse it by taking the 20 most prominent components of the SVD performed on the data matrix. The original and reconstructed results are shown in figure 4.1 and figure 4.2. As we can see, the reconstruction fails to resemble the original data. This means the linear tangent space represented by A matrix cannot predict very much further from the first column of data, (which is used as reference for reconstruction). This is as expected. 99 column vector equations cannot be expected to be satisfied by a 35×35 matrix A, especially if the source of these equations is nonlinear in nature. In other words, a 99 variable system of equations cannot be solved with only 35 equations.

Next we subject data of 24 months to the DMD algorithm and use the first 10 modes for reconstruction. The original data and reconstructed data are visualized in figure 4.3 and 4.4 respectively. Here again the reconstruction is not good, because of the limited number of modes chosen. Here the problem is that the modes required to satisfactorily reconstruct the data matrix is more than the 10 we considered.

Finally we consider subjecting data for 24 months to DMD algorithm, with 20 modes taken into consideration. Now we see that the data matrix is approximately reconstructed by the dynamic modes. The original data matrix is the same as figure 4.3 and the reconstructed matrix is shown in figure 4.5. Examining the time evoultion of the 20 dominant modes (with special focus on the largest contributing modes), show us the trend that is seen in the data matrix. This is shown in figure 4.6. We can see that there are two large dominant modes. One of them is oscillatory in nature, with a slight increase. The other one is completely negative, but after an inital dip, shows an increase to lesser negative values. Both of them suggest a slight increase in rainfall pattern, over the 24 month period taken into consideration.

4.2 Computing governing equations using SINDy

The objective of SINDy is to be able to compute the governing equation of a system purely from the data generated by that system. Inorder to perform this, we need to use a sparse regression method. The sparse regression framework used here is a simple one. Even though LASSO method is said to work, in practice it gave poor results on implementation (with the help on cvxpy library). The method used in the original paper [1], however, gave good results and is used to obtain the results shown. This method involves applying least square projection, followed by pruning of values below a cut-off, iteratively until a sparse representation is obtained. Care should be taken in selecting the cut-off, as well as selecting the library functions.

In order to test the SINDy algorithm, two dynamical system models were chosen, which arise in practical applications quite often. These models are called supercritical and subcritical systems. They normally appear in cases where the system's stability has some sort of symmetry involved. First we consider the supercritical model. The noticable qualities of this model include invariance under change of sign of parameter. The normal form of this type of model is,

$$\dot{x} = rx - x^3 \tag{4.1}$$

where r is the model parameter.

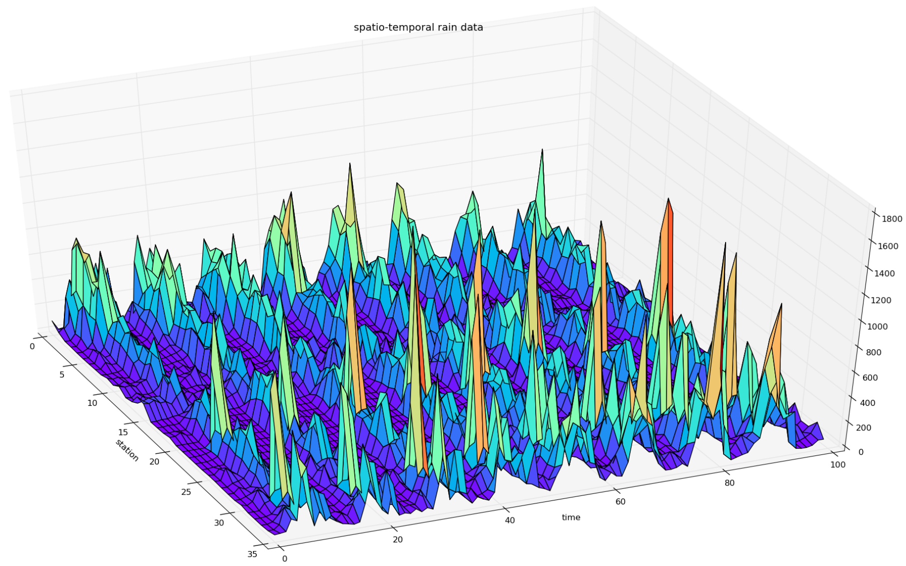

Figure 4.1: Original timeseries for data spanning 100 months with 20 leading modes

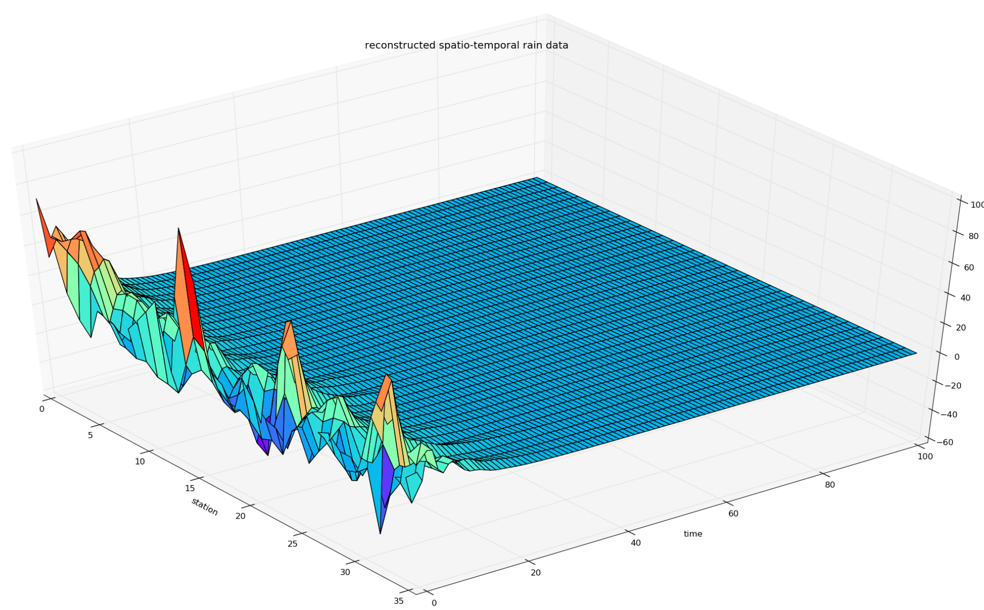

Figure 4.2: Reconstructed timeseries for data spanning 100 months with 20 leading modes

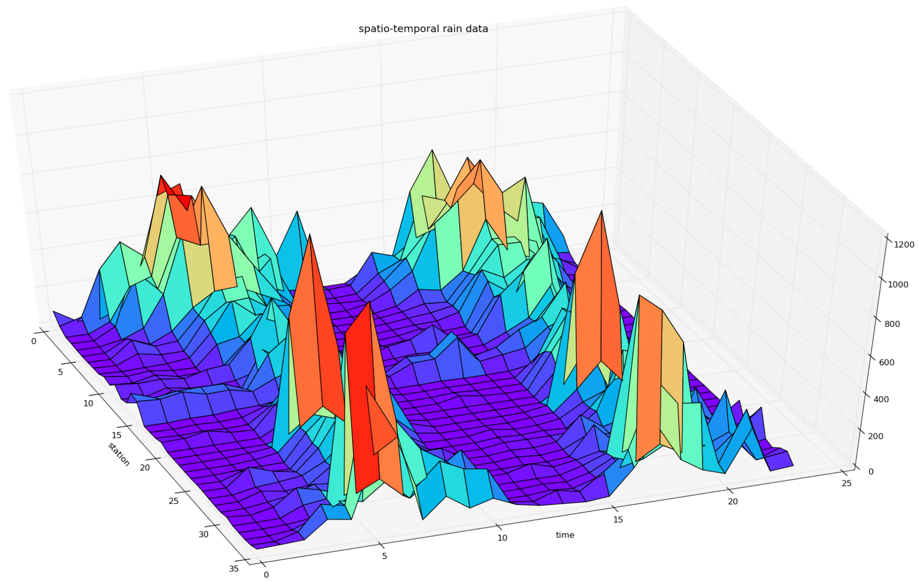

Figure 4.3: Original timeseries for data spanning 24 months with 10 leading modes

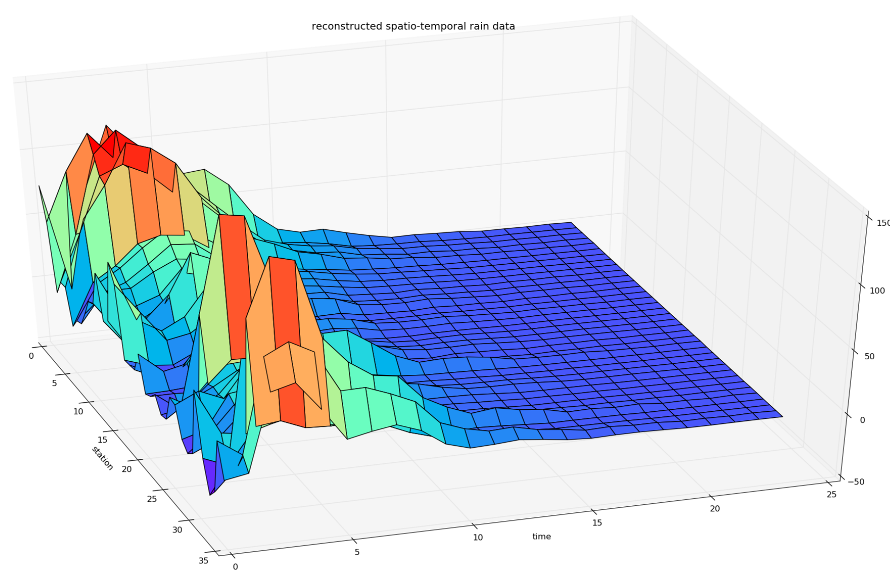

Figure 4.4: Reconstructed timeseries for data spanning 24 months with 10 leading modes

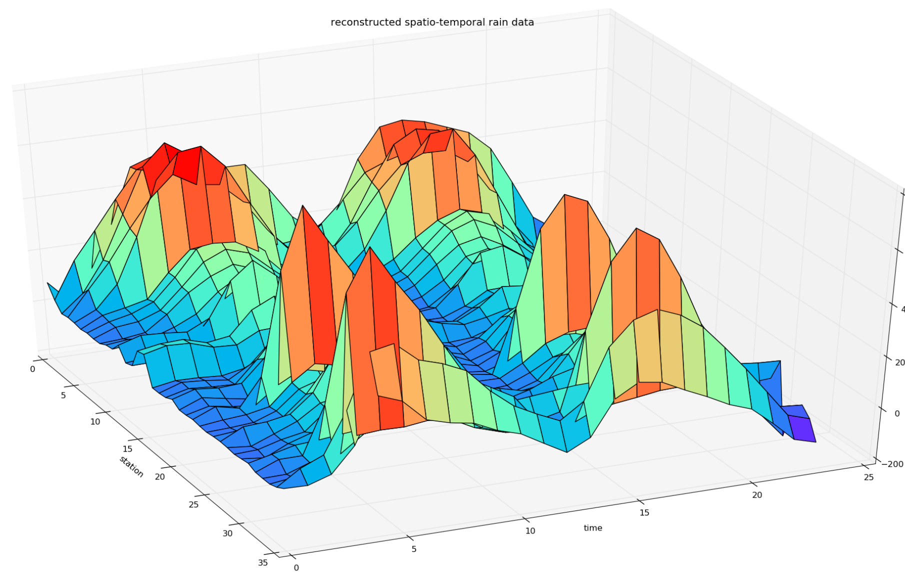

Figure 4.5: Reconstruced timeseries for data spanning 24 months with 20 leading modes

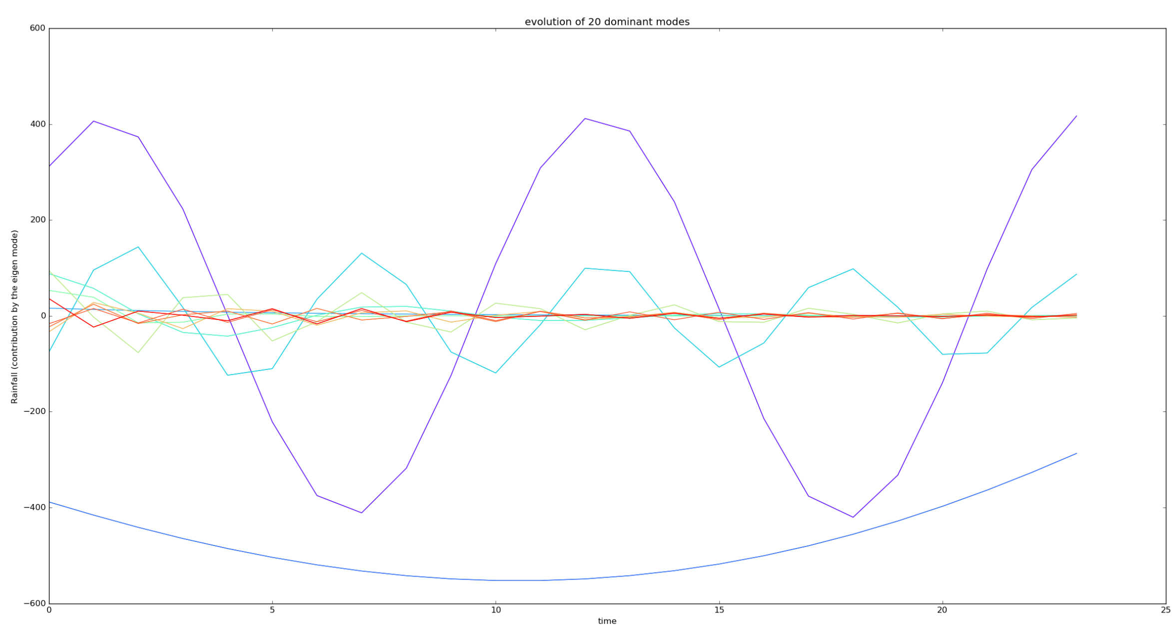

Figure 4.6: Time evolution of the 20 dominant modes

Table 4.1: Output of SINDy for different parameters r

r	x	x^2	x^3	x^4	x^5
-0.1	-0.1001	0	-0.9980	0	0
-0.2	-0.2004	0	-0.9978	0	0
0.0	0	0	-0.9989	0	0
0.1	0.0997	0	-0.9986	0	0
0.2	0.1996	0	-0.9987	0	0

Table 4.2: Output of SINDy for different parameters r

r	x	x^2	x^3	x^4	x^5
-0.1	-0.1001	0	-0.9980	0	0
-0.2	-0.2004	0	-0.9978	0	0
0.0	0	0	-0.9989	0	0
0.1	0.0997	0	-0.9986	0	0
0.2	0.1996	0	-0.9987	0	0

First we simulate the time-series for this model using the Runge-Kutta fourth order method. In this manner, we obtain both the $\dot{x}$ and x time-series. Then we create the library for SINDy with the time-series of x. This is then put into our sparse regression framework. The results for prediction of the supercritical normal mode formula is given in table-4.1.

Next we perform the same procedure on the subcritical model whos normal mode equation (along with a damping term) is,

$$\dot{x} = rx + x^3 - x^5 \tag{4.2}$$

The results obtained is depicted in table-4.2.

Chapter 5

Summary and Conclusion

It can be seen from the above experiments that DMD and SINDy are techniques which could act as building block towards purely data-driven modelling. There are still factors in both the techniques that require input from an expert in the application domain from which the data is taken. In the form depected in this report, these techniques are still useful. They can be used to verify the findings and knowlegable guesses of field experts.

Since both these techniques are popular in the nonlinear dynamics and time series analysis communities, there have been further developments in the field. More complex systems, than the ones depicted in this report, have been used to test and validate these methods. For example, SINDy has been used to reconstruct the Lorentz attractor. DMD has been used to analyse the trends in stock market data.

The analysis of rainfall data using DMD is still a challenging one. The spatio-temporal resolution of the data itself plays an important role in the accuracy of the dynamic modes found. On the other hand, application of SINDy on the pitchfork bifurcation systems is only done for exposition to the technique. But since, like already mentioned, these systems are common occurances, they are likely to be found in many real-life systems.

In short, both the techniques offer unique ways to look at nonlinear system identification, analysis and prediction. Though they haven't yet achived perfection, upon further exploration and modification, they could develop into fully data driven modelling tools.

Chapter 6

Future Work

Since both DMD and SINDy are comparitively nascent tools for system identification, there are many routes that can be taken for future work. Some of them are,

- Using SINDy on pratial differential equations. This work has already been initated by Rudy et. al [11].
- Using SINDy to predict governing equations of dynamic modes given by DMD.
- Exploring the connection of DMD with Koopman operator theory further.
- Application of above techniques to varied datasets with better spatio-temporal resolutions.
- Exploring the combination of above techniques with tensor methods to make use of higher dimensional corelations than that which is offered by normal matrices.
- Ultimately, using these system identifcation techniques, tensor method and novel machine learning algorithms to create a generic framework for system identification, applicable to a wide range of qualitatively diverse applications.

Bibliography

[1] Steven Brunton, Joshua Proctor, and Nathan Kutz. Sparse identification of nonlinear dynamics (sindy). *Bulletin of the American Physical Society*, 61, 2016.

[2] Steven L Brunton, Bingni W Brunton, Joshua L Proctor, Eurika Kaiser, and J Nathan Kutz. Chaos as an intermittently forced linear system. *arXiv preprint arXiv:1608.05306*, 2016.

[3] Emmanuel J Candès et al. Compressive sampling. In *Proceedings of the international congress of mathematicians*, volume 3, pages 1433–1452. Madrid, Spain, 2006.

[4] J Doyne Farmer, Norman H Packard, and Alan S Perelson. The immune system, adaptation, and machine learning. *Physica D: Nonlinear Phenomena*, 22(1):187–204, 1986.

[5] Zoubin Ghahramani and Sam T Roweis. Learning nonlinear dynamical systems using an em algorithm. *Advances in neural information processing systems*, pages 431–437, 1999.

[6] B.O Koopman. Hamiltonian systems and transformations in hilbert space. Proceedings of the National Academy of Sciences, 17(5):315, 1931.

[7] John R Koza. *Genetic programming: on the programming of computers by means of natural selection*, volume 1. MIT press, 1992.

[8] John L Lumley. *Stochastic tools in turbulence*. Courier Corporation, 2007.

[9] Government of India. All india area weighted monthly, seasonal and annual rainfall (in mm). [Online; accessed 1-10-16].

[10] Clarence W Rowley, Igor Mezić, Shervin Bagheri, Philipp Schlatter, and Dan S Henningson. Spectral analysis of nonlinear flows. *Journal of fluid mechanics*, 641:115–127, 2009.

[11] Samuel H Rudy, Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. Data-driven discovery of partial differential equations. *arXiv preprint arXiv:1609.06401*, 2016.

[12] Peter J Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656:5–28, 2010.

[13] Michael Schmidt and Hod Lipson. Distilling free-form natural laws from experimental data. *science*, 324(5923):81–85, 2009.

[14] Floris Takens. Detecting strange attractors in turbulence. In *Dynamical systems and turbulence, Warwick 1980*, pages 366–381. Springer, 1981.

[15] Wikipedia. Complex systems — Wikipedia, the free encyclopedia, 2004. [Online; accessed 25-11-2016].

[16] Matthew O Williams, Ioannis G Kevrekidis, and Clarence W Rowley. A data–driven approximation of the koopman operator: Extending dynamic mode decomposition. *Journal of Nonlinear Science*, 25(6):1307–1346, 2015.