CIDU CONFERENCE ON INTELLIGENT DATA UNDERSTANDING

Proceedings of the 2011

Conference on Intelligent Data Understanding

CIDU 2011

October 19-21, 2011
Mountain View, California

Editors:
Ashok N. Srivastava, General Chair
Nitesh V. Chawla, Program Chair
Amal Shehan Perera, Proceedings Chair

Sponsored by:
The National Aeronautics and
Space Administration:
The Aviation Safety Program and
The Engineering and Safety Center

Foreword

The NASA Conference on Intelligent Data Understanding (CIDU) is applications-oriented, with a focus on Earth Sciences, Space Sciences, and Aerospace and Engineering Systems Applications. The conference originated in 2004 as a small workshop in Cleveland Ohio with about 25 participants. This is the second year that CIDU is publishing full-length papers. This year's papers feature articles from international participants covering a wide range of topics such as smart grids, system health management, discovering climate phenomenon, text mining, and methods appropriate to analyze sky surveys. CIDU is unique in creating a forum for the applications of data mining and machine learning to earth sciences, space sciences, and aerospace and engineering systems. These high quality full-length papers represent the work of practitioners in the application areas as well as established researchers in the data mining and machine learning communities. The conference features invited speakers, poster sessions, oral paper presentations, panel, and networking opportunities for interested researchers and students. The proceedings of CIDU 2011 will be archived in the NASA Center for Aerospace Information and will be indexed by DBLP. Selected papers will be invited for consideration for the CIDU special issue in the Statistical Analysis and Data Mining Journal.

CIDU 2011 is sponsored by the NASA Engineering and Safety Center and the NASA Aviation Safety Program. The organizers are grateful for the support from these organizations.

Ashok N. Srivastava, NASA Ames Research Center (General Chair)
Nitesh V. Chawla, University of Notre Dame (Program Chair)

CIDU 2011 Core Topics

Earth & Environmental Systems Applications

- Climate and ecology data sciences
- Climate modeling
- Sustainability
- Geographic information systems
- Geospatial intelligence
- Spatio-temporal data mining
- Visual analytics for earth science data
- High-performance computing applications
- Evaluation or validation techniques

Space Science Applications

- On-board and real-time machine learning
- Decision making under uncertainty
- Constraint-driven data mining and machine learning
- Event mining and robotic telescopes
- Unsupervised and supervised learning in astrophysics
- Highly scalable algorithms
- Risk management in space missions
- Classification in large sky surveys

Aerospace and Engineering Systems

- Related government engineering applications (DOE, DOD, others)
- Systems health applications
- Anomaly detection, diagnostics, and prognostics from large data sets
- Text mining in aerospace information systems
- Data driven reliability modeling
- Adaptive system monitoring
- System model identification
- Large data set challenges
- Exploratory mining of aerospace data
- Privacy and security issues in aerospace data
- Statistical process control using very large datasets

CIDU 2011 Conference Organization

General Chair:	Ashok N. Srivastava *(NASA AMES Research Center)*
Program Chair:	Nitesh V. Chawla *(University of Notre Dame)*
Earth Science Applications Area Chair:	Claire Monteleoni, *(Columbia University)*
Space Science Applications Area Chair:	Kirk Borne *(George Mason University)*
Aerospace and Engineering Systems Area Chair:	Paul Melby, *(MITRE)*
Posters Chair:	Kanishka Bhadrui *(NASA AMES Research Center)*
Proceedings Chair:	Amal Shehan Perera *(University of Moratuwa, Sri Lanka)*
Publicity Chair:	Karsten Steinhaeuser, *(University of Notre Dame / ORNL)*
Local Arrangements Chair:	Elizabeth Foughty *(NASA AMES Research Center)*
Communications Chair:	Kamalika Das *(NASA AMES Research Center)*

CIDU 2011 Conference Committees

Steering Committee

Stephen Boyd *(Stanford University)*
Jiawei Han *(University of Illinois at Urbana-Champaign)*
Vipin Kumar *(University of Minnesota)*
Eamonn Keogh *(University of California, Riverside)*
Zoran Obradovic *(Temple University)*
Nikunj Oza *(NASA Ames Research Center)*
Raghu Ramakrishnan *(Yahoo!)*
Ramasamy Uthurusamy *(General Motors)*
Ramasubbu Venkatesh *(Netflix Inc.)*
Xindong Wu *(University of Vermont)*

Program Committee Members:

Arindam Banerjee *(University of Minnesota)*
Robert Brunner *(UIUC/NCSA)*
Douglas Burke *(Harvard University)*
Michael Burl *(NASA Jet Propulsion Laboratory)*
Alfredo Cuzzocrea *(ICAR-CNR & University of Calabria, Italy)*
Auroop Ganguly *(Oakridge National Laboratory)*
Jiawei Han *(University of Illinois at Urbana-Champaign)*
Latifur Khan *(University of Texas at Dallas)*
Vipin Kumar *(University of Minnesota)*
Mark Last *(Ben-Gurion University of the Negev)*
Amy McGovern *(University of Oklahoma)*
Tim Oates *(University of Maryland Baltimore County)*
Olufemi Omitaomu *(Oakridge National Laboratory)*
Jeff Scargle *(NASA Ames Research Center)*
Ranga Raju Vatsavai *(Oakridge National Laboratory)*
Mike Way *(NASA Ames and NASA Goddard)*
Xindong Wu *(University of Vermont)*
Zhi-Hua Zhou *(Nanjing University, China)*
Kiri Wagstaff *(NASA, Jet Propulsion Laboratory)*
Dimitris Giannakis *(New York University)*
Tao Shi *(Ohio State University)*
Sara Graves *(University of Alabama)*
Philip Yu *(University of Illinois at Chicago)*
Shyam Boriah *(University of Minnesota)*
Joao Gama *(University of Porto, Portugal)*

Table of Contents

Session 1

Session 2

Session 3

Session 4

Session 5

Session 6

Poster Paper Session

A SUSTAINABLE APPROACH FOR DEMAND PREDICTION IN SMART GRIDS USING A DISTRIBUTED LOCAL ASYNCHRONOUS ALGORITHM

RAJARSHI MALLIK* AND HILLOL KARGUPTA**

Abstract. Energy production, distribution, and consumption play a critical role in the sustainability of the planet and its natural resources. Electric power systems have been going through major changes that are aimed to make the energy infrastructure "smarter", scalable, and more efficient. These new generation of smart energy grids need novel computational algorithms for supporting generation of power from wide range of sources, efficient energy distribution, and sustainable consumption. This paper argues that a fundamentally distributed approach with more local flexibility is a lot more sustainable methodology compared to the traditional centralized frameworks for analyzing and processing data. It considers the problem of predicting power generation and consumption trends over a distributed smart grid. Since power generation from solar, wind, geothermal and other renewable sources are likely to be part of many households in near future, both power generation and consumption data will be generated over a wide area network. Moreover, a good part of the communication links between the household data sources and the central server are likely to be over the wireless networks with low bandwidth and high data-plan cost. Analyzing such data (some of it privacy sensitive) in a centralized is not scalable, sometimes not privacy-preserving, and often not practical because of cost-sensitivity of the applications. This paper presents a more sustainable distributed asynchronous algorithm for constructing energy demand prediction models in a smart grid by multivariate linear regression. The paper offers the algorithm, analysis, and experimental results.

1. Introduction

Sustainability implies resource consumption with little internal or external adverse impact so that it can be practiced for an extended period. A system or a process is sustainable when its input and output have little adverse impact on its environment and therefore can be accepted as a long term practice. A system that is not sustainable often leads to the failure (sometimes catastrophic) of the system itself or other systems in its environment. Sustainability in Electric Power systems is a very important issue since it usually has a significant impact on the environment and other systems that share the same environment. Electric power systems are undergoing a profound change driven by a number of needs. Environmental impact, reliability, operational efficiencies in energy generation and distribution along with alternate power generation technologies and "intelligent" appliances are driving the need for developing the new generation of energy networks — the Smart Grids. A sustainable Smart Grid would deliver high performance at the right cost with little impact on the environment. This paper argues that this demands making the "smart" in the Smart Grid really smart by deploying proper adaptive machine learning and data analysis techniques for energy production and distribution.

The vast amount of sensor data from Smart Grid's physical and operational state along with environmental sensor data are being available as both real time and from archival stores which contains important correlations, trends, and patterns that can be exploited for optimizing operations with respect to sustainability metrics, such as, energy consumption, carbon footprint etc. However, considering the large amount of data produced, manual inspection is virtually impossible and thus automated knowledge discovery and data mining techniques are necessary to synthesize models for enabling sustainable end-to-end operation. These data mining techniques and analytics will extract

*University of Maryland Baltimore County, rmallik1@umbc.edu
**University of Maryland Baltimore County, Agnik LLC, hillol@cs.umbc.edu.

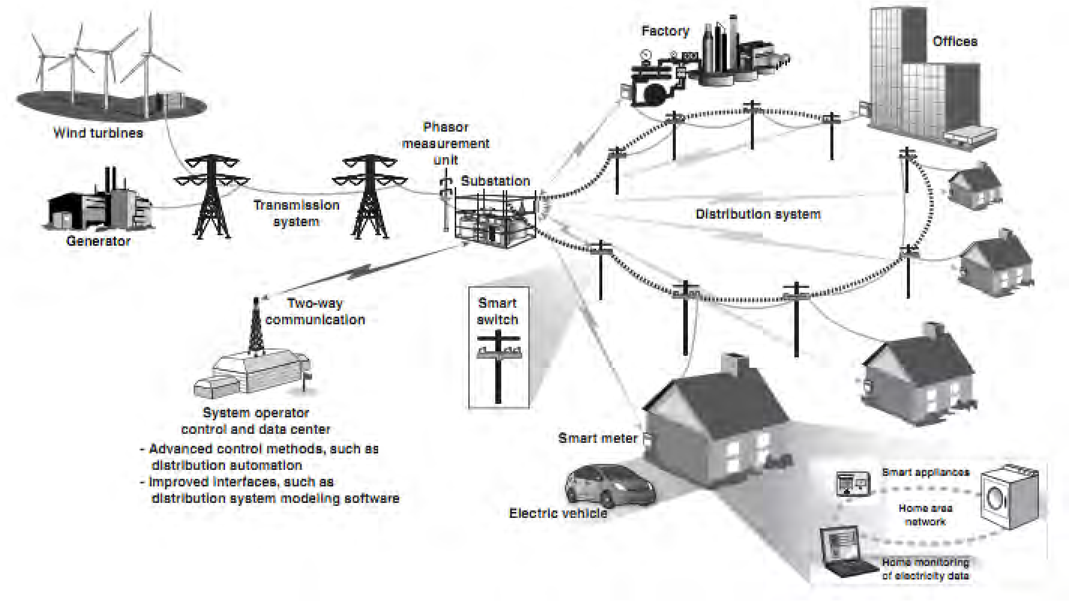

Factory
Offices
Wind turbines
Phasor measurement unit
Substation
Transmission system
Generator
Distribution system
Two-way communication
Smart switch
System operator control and data center
- Advanced control methods, such as distribution automation
- Improved interfaces, such as distribution system modeling software
Smart meter
Electric vehicle
Smart appliances
Home area network
Home monitoring of electricity data

FIGURE 1. Smart Grids - Efficient two-way communication electrical power grids.[1]

prediction rules, which when embedded in distributed, decision support or real time data engines will help shape electric consumption and optimize the grid leading to greater sustenance. But analyzing data in a smart grid is unlikely to be both commercially and physically sustainable if we rely upon centralized architectures since a smart grid is by definition comprised of large number of loosely coupled asynchronous entities. Let us note the following observations:

- It is very unlikely that a single organization will be in charge of the entire Smart Grid and it will be able to collect all the data at a central location. A centralized approach for data analysis would require different competing business entities to share the data of their own customers which is unlikely to be acceptable.
- Users may have reservations regarding giving up their own household data (often privacy sensitive) to a business entity. Wide acceptance of the technology is likely be hindered unless we are sensitive to the privacy issues. A centralized data mining approach is likely to be very difficult to plement while preserving various heterogeneous privacy-preserving regulations from different places.
- Many of the Smart Grid sensors (e.g. energy meter, appliance monitors) communicate with remote servers over wide area wireless modems. Transmitting large volume of data over low-bandwidth wireless modems is very expensive, not scalable, and have higher energy consumption (therefore carbon emission) footprint.

This paper argues that a distributed approach would be more sustainable since it allows little or no sharing of raw data, more suitable for integration with privacy-preserving techniques, and minimizes overall communication among different nodes in the network. The paper specifically considers the problem of multivariate regression in a distributed scenario since learning linear models is a common problem in many statistical data analysis and adaptive learning techniques such as perceptrons and kernel regression. Such techniques are widely used for predicting and forecasting which is directly relevant to the energy demand forecasting problem considered in this paper.

[1] Courtesy U.S Government Accountability Office

Rest of the paper is organized as follows. Section 2 looks at the related works and Section 3 places the problem definition. Section 4 presents the overview of linear regression, its terms and notations. Section 5 describes how normal equations are solved. Section 6 explains the iterative algorithm and the choice of the gossip algorithm. Section 7 presents the experimental results and discusses the simulations and measurement metrics. Finally, Section 8 concludes this paper.

2. Related Work

Predicting demand on a Smart Grid requires forecasting the difference between the energy production and consumption based on several factors like geographical area, weather, type and number of consumers and other patterns which are extracted from a series of events. A survey of challenges related to computational sustainability in general and that of Smart Grids in the area of planning and operating large complex digital ecosystems, controlling and measuring technologies from a producer controlled network to a more decentralized system has been showed by Carla P Gomes et. al [13]. There exists a body of literature dealing with algorithms and systems-related challenges for information processing over smart grids. Hurdles like standards of interoperability in information obtained from these smart grids are addressed by NIST [3]. The problem of estimation of time between failures in electric grids leading to greater sustenance has been previously modeled in [11]. The problem is particularly related to multiple and distributed failure modes and causes with potential explosion of data. Data mining challenges and techniques in sustainable and efficient transportation systems with large volume of data are addressed in [12]. The impact of local consumption, group behavior and its effect are also addressed giving rise to new challenges and opportunities. [17]. Intelligent techniques for smart grids have been explored elsewhere [19]. Various predictive models for smart grid enabling devices like smart meters are explored elsewhere [10].

This paper considers the problem of distributed regression over smart grid. Therefore, it is important to make a note of the existing work related to the distributed regression technique. Hershberger et al.[15] considered the problem of solving global regression coefficients in a vertically partitioned data distribution scenario. This is a synchrnized algorithm where each node computes the wavelet transformation of the data and exchanges the significant coefficients in a synchronous manner in order to regress the global coefficients of the model. Algorithms presented by Guestrin et al.[14] performs linear regression in a network of sensors using in-network processing of messages where instead of transmitting raw data it transmits only constraints thereby reducing communication complexity. Bhaduri et al[5] addresses the problem by providing a scalable local algorithm for multivariate regression. This paper makes use of a convergecast phase where data is sampled from the network to a central post to compute the coefficients based on the samples followed by a broadcast phase where these coefficients are distributed into the network and the results are monitored in-node. Though the root of the convergecast tree is not pre-specified it inherently doesn't steer clear of the centralized approach once the tree is built and subsequently during tie resolution during broadcast phase.

The current work makes use of a distributed sum computation technique which builds on existing work. There has been a lot of work on distributed computation of averages (in general first order statistics) more generally problem of reaching agreement or consensus among peers via distributed computation[16][18]. The aggregation problem approach proposed by Bawa et al.[4] is to count the sum and count queries subject to a model of network behavior and guaranteeing bound errors, however it was explained calculating sums using this method can be costly and the protocols are based mostly on building trees. Boyd et al.[6] presented a simple iterative gossip algorithm and showed a connection between the averaging time of the algorithm and mixing time of an appropriate random walk. Distributed summation is computed in asymptotically minimal rounds by Kempe's et al.[16] Push-Sum algorithm. Here we used Kempe's push sum protocol to compute the linear regression coefficients in a distributed environment.

The following section defines the distributed energy demand prediction problem in the context of distributed regression.

3. Problem Definition

This paper considers the problem of predicting energy demand using distributed energy consumption and production data. It makes the following assumptions about the overall smart grid model of energy management:

(1) With the advent of solar panels, smaller wind turbines, and geothermal technology, energy production is likely to be a household process in the near future. Residences are likely to be producing and supplying surplus energy to the community. In addition, traditional energy companies will be producing and selling energy.
(2) Eventually, a smart grid will most likely be supporting multiple energy company. Therefore, the household energy consumption data may not belong to a single company. Even if it does, privacy protection of the consumer will be an important issue.
(3) A household or a corporate entity may or may not want to allow centralizing data

Energy demand for a household or an area can be computed from the household's energy production and consumption data. Forecasting the demand would require building predictive models from the observed demand data and various other features such as consumption behavior, housing and household characteristics, geographical location, season and time of the day. While there exists many techniques for learning predictive models, multivariate regression is a popular well-understood technique for constructing such predictive models. This paper considers the problem of learning predictive models for the energy demand based on the observed household energy production and consumption data.

Computing a global regression model from the data available at distributed sources can be costly, inefficient and sometimes impractical in many scenarios because of various reasons such as large number of data sources, the asynchronicity and dynamic nature of the networks, multi-organizational structure of the environment and privacy issues among others.

The problem and proposed solution we present in this paper is abstracted on a distributed environment of the grid infrastructure. The data aggregators like smart meters can be seen as nodes in a network represented by the grid, we can construct a network of N nodes of no specific overlay topology. Each node N_i contains data tuples given by X_i^m where m is the number of features of the data. We have assumed that the data present at each node is homogeneous in nature. These data represent the information that is generated at each node by virtue of the consumption and production characteristics of the grid network.Our goal is to learn a multivariate regression model where we compute the approximate linear coefficients for the global regression model locally at each node in a distributed and asynchronous manner.

4. Linear Regression - An Overview

Regression is the task of learning a target function $\widehat{f}$ that maps each attribute set x into a continuous-valued output $f(x)$. The goal is to find the target function that can fit the input data with minimum error. Given a data set that contains n observations $(\vec{x_i}, f(\vec{x_i}))$ where $i = 1, 2, ..., n$ each $\vec{x_i}$ corresponds to the set of attributes of the i^{th} observation and $f(\vec{x_i})$ corresponds to the target variable. For linear regression the idea is to learn $\widehat{f}(\vec{x_i})$ which approximates $f(\vec{x_i})$ for all data observations and is a linear combination of any d specified functions of x given by the general form $\widehat{f}(\vec{x_i}) = \sum_{k=0}^{d-1} w_k X_k(\vec{x_i})$ where $X(\vec{x_i})$ are arbitrary fixed functions of x and w_k are the coefficients or parameters that need to be estimated. Since $\widehat{f}$ is linear, the problem reduces to finding the parameter vector $w \in \mathbb{R}^d$ such that $y = f(x) = w^T x$.

Here in case of general linear regression we choose a linear model which we want to fit as $\widehat{f}(\vec{x_i}) = w_0 + w_1 x_{i1} + w_2 x_{i2} + \ldots + w_m x_{im}$ where w_j's are the coefficients that need to be estimated from the dataset. For every data point in the set of n observations, the squared error is:

$$\begin{gathered}E_1 = [f(\vec{x_1}) - w_0 - \ldots - w_m x_{1m}]^2\\ E_2 = [f(\vec{x_2}) - w_0 - \ldots - w_m x_{2m}]^2\\ \vdots\\ E_n = [f(\vec{x_n}) - w_0 - \ldots - w_m x_{nm}]^2\end{gathered}$$

The total square error over all the data points is
$SSE = \sum_{j=1}^{n} E_j = \sum_{j=1}^{n} E_j [f(\vec{x_j}) - w_0 - \ldots - w_m x_{jm}]$
For linear regression, closed form expressions exist for finding the coefficients w_i's by finding the partial derivatives of SSE with respect to the $w_i's$ and setting them to zero. In the matrix form it can be written as - $Xw = Y$

$$\begin{pmatrix} \sum_{i=1}^{n} x_{i1}^2 & \sum_{i=1}^{n} x_{i1}x_{i2} & \ldots & \sum_{i=1}^{n} x_{i1}x_{im} \\ \sum_{i=1}^{n} x_{i2}x_{i1} & \sum_{i=1}^{n} x_{i2}^2 & \ldots & \sum_{i=1}^{n} x_{i2}x_{im} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{n} x_{im}x_{i1} & \sum_{i=1}^{n} x_{im}x_{i2} & \ldots & \sum_{i=1}^{n} x_{im}^2 \end{pmatrix} \times \begin{pmatrix} w_0 \\ w_1 \\ \vdots \\ w_m \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{n} x_{i1} f(\vec{x_i}) \\ \sum_{i=1}^{n} x_{i2} f(\vec{x_i}) \\ \vdots \\ \sum_{i=1}^{n} x_{im} f(\vec{x_i}) \end{pmatrix}$$

We solve the normal equation as shown above to get the coefficients of linear regression w_i.

5. Solving Normal Equations

Given the data tuples comprising of the vector of *predictor(x)* and *response(y)* variables, finding the *regression coefficients* is solving the linear equation of the form $Aw = b$ where row vector w are regression coefficients.A m-dimensional n data tuples are given by

$$\begin{pmatrix} x_{11} & x_{12} & \ldots & x_{1m} & y_1 \\ x_{21} & x_{22} & \ldots & x_{2m} & y_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{n1} & x_{n2} & \ldots & x_{nm} & y_n \end{pmatrix}$$

and the corresponding correlation matrix is given by

$$X^T.X \;=\; A \;=\; \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1m} \\ a_{21} & a_{22} & \ldots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \ldots & a_{mm} \end{bmatrix}$$

All entries of correlation matrix A and b can be written as the sum of individual predictor variables as shown next

$$A = \begin{bmatrix} \sum_{i=1}^{n} x_{i1}^2 & \sum_{i=1}^{n} x_{i1}x_{i2} & \dots & \sum_{i=1}^{n} x_{i1}x_{im} \\ \sum_{i=1}^{n} x_{i2}x_{i1} & \sum_{i=1}^{n} x_{i2}^2 & \dots & \sum_{i=1}^{n} x_{i2}x_{im} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{n} x_{im}x_{i1} & \sum_{i=1}^{n} x_{im}x_{i2} & \dots & \sum_{i=1}^{n} x_{im}^2 \end{bmatrix} \quad b = \begin{pmatrix} \sum_{i=1}^{n} x_{i1}y_i \\ \sum_{i=1}^{n} x_{i2}y_i \\ \vdots \\ \sum_{i=1}^{n} x_{im}y_i \end{pmatrix}$$

We solve the normal equation of the form $A.w = b$ to get the linear regression co-efficients.The equation shown before are called the normal equations of the least-squares problem i.e., equations that minimizes the sum of square differences between the left and the right sides of the equation. It is called normal because $b - A.w$ is normal to the range of A. Given a matrix equation of the form

$$A.w = b \tag{1}$$

can be solved for the vector of parameters a by standard methods notably LU decomposition and back substitution, Cholesky decomposition or Gauss-Jordan Elimination. In matrix form, this can be written as

$$(A^T.A).w = A^T.b \tag{2}$$

In theory, since $A^T.A$ is symmetric and positive-definite, Cholesky decomposition is the most efficient way to solve the normal equations. Cholesky decomposition is about a factor of two faster than alternative methods for solving linear equations. Cholesky decomposition constructs a lower triangular matrix L whose transpose L^T can itself serve as the upper triangular triangular part. In other words we replace A by

$$A = L.L^T \tag{3}$$

$$\text{i.e,} \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & x_{nm} \end{bmatrix} = \begin{bmatrix} L_{11} & 0 & \dots & 0 \\ L_{21} & L_{22} & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ L_{m1} & L_{m2} & \dots & L_{mm} \end{bmatrix} . \begin{bmatrix} L_{11} & L_{12} & \dots & L_{1m} \\ 0 & L_{22} & \dots & L_{2m} \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & L_{mm} \end{bmatrix}$$

This factorization is sometimes referred to as 'taking square root' of a matrix A, though, because of the transpose, it is not literally that. The component of L^T are of course related to those of L by

$$L_{ij}^T = L_{ji}$$

Writing Equation 3 in components, we can readily obtain

$$L_{ii} = \left(a_{ii} - \sum_{k=1}^{i-1} L_{ik}^2 \right)^{\frac{1}{2}} \tag{4}$$

$$L_{ji} = \frac{1}{L_{ii}} \left(a_{ij} - \sum_{k=0}^{i-1} L_{ik}L_{jk} \right) \tag{5}$$

$$j = i+1, i+2, \dots, m-1$$

The sequence in which L_{ij} is solved is given by

$$\begin{tikzcd}
L_{11} \arrow[d] & & & \\
L_{21} \arrow[r] \arrow[d] & L_{22} \arrow[d] & & \\
L_{31} \arrow[r] \arrow[d] & L_{32} \arrow[r] \arrow[d] & L_{33} \arrow[d] & \\
\dots \arrow[r] & \dots \arrow[r] & \dots \arrow[r] & {}
\end{tikzcd}$$

Now we need to solve the equation of the form $L.L^T.w = b$. Substituting $p = L^T.w$ and solving for p in the equation $L.p = b$ is given by

$$p_i = \frac{1}{L_{ii}} \left[b_i - \sum_{j=1}^{i-1} L_{ij} p_j \right] \tag{6}$$

$$i = 1, 2, \dots, m$$

Finally solving for w, the regression coefficients in the equation $L^T.w = p$, we have

$$w_i = \frac{1}{L_{ii}} \left[p_i - \sum_{j=i+1}^{m} L_{ji} w_j \right] \tag{7}$$

$$i = m, m-1, \dots, 1$$

6. Iterative Approach

In this approach, data matrix at time t given by entries $x_{ij}\, and\, y_i$ where $i = 1 \dots n$, $j = 1 \dots m$.The new data matrix formed by data tuples that arrived at time $(t+1)$ is given by $x_{ij}\, and\, y_i$ where $i = (n+1) \dots z$, $j = (n+1) \dots m$. So the new data matrix formed from the data during the whole period is given by $x_{ij}\, and\, y_i$ where $i = 1 \dots z$, $j = 1 \dots m$

The corresponding correlation matrix remains unchanged except for the fact that the summation terms is now to variable z as opposed to n before,

$$\text{A} = \begin{bmatrix} \sum_{i=1}^{z} x_{i1}^2 & \sum_{i=1}^{z} x_{i1} x_{i2} & \dots & \sum_{i=1}^{z} x_{i1} x_{im} \\ \sum_{i=1}^{z} x_{i2} x_{i1} & \sum_{i=1}^{z} x_{i2}^2 & \dots & \sum_{i=1}^{z} x_{i2} x_{im} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{z} x_{im} x_{i1} & \sum_{i=1}^{z} x_{im} x_{i2} & \dots & \sum_{i=1}^{z} x_{im}^2 \end{bmatrix} \tag{8}$$

Similarly L_{ii} and L_{ji} from equation 4 and 5 at time t and time $(t+1)$ is given by

$$L_{ii(t)} = \left(a_{ii(t)} - \sum_{k=1}^{i-1} L_{ik(t)}^2 \right)^{\frac{1}{2}} \tag{9}$$

$$L_{ii(t+1)} = \left(a_{ii(t+1)} - \sum_{k=1}^{i-1} L_{ik(t+1)}^2 \right)^{\frac{1}{2}} \tag{10}$$

$$L_{ji(t)} = \frac{1}{L_{ii(t)}} \left(a_{ij(t)} - \sum_{k=1}^{i-1} L_{ik(t)} L_{jk(t)} \right) \tag{11}$$

$$L_{ji(t+1)} = \frac{1}{L_{ii(t+1)}} \left(a_{ij(t)} - \sum_{k=1}^{i-1} L_{ik(t+1)} L_{jk(t+1)} \right) \tag{12}$$

We express the term $L_{ii(t+1)}$ as an iteration of $L_{ii(t)}$ in the following way

$$L^2_{ii(t+1)} - L^2_{ii(t)} = a_{ii(t+1)} - a_{ii(t)} \; - \sum_{k=1}^{i-1} L^2_{ik(t+1)} + \sum_{k=1}^{i-1} L^2_{ik(t)}$$

$$\Rightarrow L_{ii(t+1)} = \left(\underbrace{L^2_{ii(t)}}_{Lookup} + \underbrace{\sum_{k=n+1}^{z} x^2_{ki}}_{New\, Data} - \underbrace{\sum_{k=1}^{i-1} L^2_{ik(t+1)}}_{Already\, Calculated} + \underbrace{\sum_{k=1}^{i-1} L^2_{ik(t)}}_{Lookup} \right)^{\frac{1}{2}} \tag{13}$$

$$where\, a_{ii(t+1)} - a_{ii(t)} = \sum_{k=1}^{z} x^2_{ki} - \sum_{k=1}^{n} x^2_{ki} = \sum_{k=n+1}^{z} x^2_{ki}$$

As we see from equation 13 that $L_{ii(t+1)}$ depends on the *new data* that becomes available at time *(t+1)* and the not the whole data matrix. So the calculation of $L_{ii(t+1)}$ depends on *lookup* of previous iteration, the *new data* arriving at time *(t+1)*, the *already calculated* entries of the $L_{ij(t+1)}$ matrix and again a simple *lookup* of the previous iteration of the entry $L_{ij(t)}$.

Similarly we express the term $L_{ji(t+1)}$ as an iteration of $L_{ji(t)}$ as follows

$$L_{ji(t+1)}.L_{ii(t+1)} - L_{ji(t)}.L_{ii(t)} = a_{ij(t+1)} - a_{ij(t)} - \sum_{k=1}^{i-1} L_{ik(t+1)}.L_{jk(t+1)} + \sum_{k=1}^{i-1} L_{ik(t)}.L_{jk(t)}$$

$$\Rightarrow L_{ji(t+1)} = \underbrace{\frac{1}{L_{ii(t+1)}}}_{Already\, calculated} \left(\underbrace{\sum_{k=n+1}^{z} x_{ki} x_{kj}}_{New\, Data} - \underbrace{\sum_{k=1}^{i-1} L_{ik(t+1)}.L_{jk(t+1)}}_{Already\, calculated} + \underbrace{\sum_{k=1}^{i-1} L_{ik(t)}.L_{jk(t)}}_{Lookup} + \underbrace{L_{ji(t)}.L_{ii(t)}}_{Lookup} \right) \tag{14}$$

We see from equation 14 that $L_{ji(t+1)}$ can be calculated iteratively from *already calculated* $L_{ii(t+1)}$, *new data* arriving at *time (t+1)*, *already calculated* entries of matrix $L_{ij(t+1)}$ and a *lookup* of the previous values of $L_{ij(t)}$.

Similarly p_i and w_i from equation 6 and 7 at time t and time $(t+1)$ is given as

$$p_{i(t)} = \frac{1}{L_{ii(t)}} \left[b_{i(t)} - \sum_{j=1}^{i-1} L_{ij(t)} p_{j(t)} \right] \tag{15}$$

$$p_{i(t+1)} = \frac{1}{L_{ii(t+1)}} \left[b_{i(t+1)} - \sum_{j=1}^{i-1} L_{ij(t+1)} p_{j(t+1)} \right] \tag{16}$$

$$w_{i(t)} = \frac{1}{L_{ii(t)}}\left[p_{i(t)} - \sum_{j=i+1}^{m} L_{ji(t)} w_{j(t)}\right] \tag{17}$$

$$w_{i(t+1)} = \frac{1}{L_{ii(t+1)}}\left[p_{i(t+1)} - \sum_{j=i+1}^{m} L_{ji(t+1)} w_{j(t+1)}\right] \tag{18}$$

Expressing $p_{i(t+1)}$ of equation 16 and $w_{i(t+1)}$ of equation 18 as an iteration of their previous values we have

$$p_{i(t+1)}.L_{ii(t+1)} - p_{i(t)}.L_{ii(t)} = b_{i(t+1)} - b_{i(t)} - \sum_{j=1}^{i-1} L_{ij(t+1)} p_{j(t+1)} + \sum_{j=1}^{i-1} L_{ij(t)} p_{j(t)}$$

$$\Rightarrow p_{i(t+1)} = \underbrace{\frac{1}{L_{ii(t+1)}}}_{Lookup}\left(\underbrace{\sum_{k=n+1}^{z} x_{ki} y_k}_{New\ Data} - \underbrace{\sum_{j=1}^{i-1} L_{ij(t+1)} p_{j(t+1)}}_{Already\ calculated} + \underbrace{\sum_{j=1}^{i-1} L_{ij(t)} p_{j(t)}}_{Lookup} + \underbrace{p_{i(t)}.L_{ii(t)}}_{Lookup}\right) \tag{19}$$

and

$$w_{i(t+1)}.L_{ii(t+1)} - w_{i(t)}.L_{ii(t)} = p_{i(t+1)} - p_{i(t)} - \sum_{j=i+1}^{m} L_{ji(t+1)} w_{j(t+1)} + \sum_{j=i+1}^{m} L_{ji(t)} w_{j(t)}$$

(20)

$$\Rightarrow w_{i(t+1)} = \underbrace{\frac{1}{L_{ii(t+1)}}}_{Lookup}\left(\underbrace{p_{i(t+1)} - p_{i(t)}}_{Lookup\ and\ calculate} - \underbrace{\sum_{j=i+1}^{m} L_{ji(t+1)} w_{j(t+1)}}_{Already\ calculated} + \underbrace{\sum_{j=i+1}^{m} L_{ji(t)} w_{j(t)}}_{Lookup} + \underbrace{w_{i(t)}.L_{ii(t)}}_{Lookup}\right)$$

The iterative approach to solve normal equations can be extended to fit computations in distributed environments. Assuming there are N nodes in a network with data matrix at each node represented by X_i^m where m represent the features of the data, then data communication among nodes connected with each other over time to compute the regression coefficients can be perceived as a asynchronous distributed problem. All of these assumptions are based on the fact that the data is homogeneous in nature distributed at different nodes or sites. Calculations of linear co-efficients can be iteratively solved and the expressions are additive in nature so a de-centralized calculation of co-efficients are possible over large distributed environments overlayed by a communication strategy or protocols like gossip based computations. In gossip protocol, a peer of a peer-to-peer network exchanges data or statistics with a random peer. Kempe et al's[16] *Push-Sum* protocol based on gossip communication for computing sum at the nodes of a network is asymptotically optimal with respect to convergence speed. Combining the *Push-Sum* algorithm with the iterative computation of coefficients of linear regression we compute the approximate global coefficients with the same bound of error as shown before. All of the above concepts relies on diffusion speed of uniform gossip based on mass conservation as presented in theorem 3.1 of [16]. As shown the estimate of an average at a node i, at time t is given by $\frac{v_{t,i}.x}{w_{t,i}}$ where $v_{t,i}$ is local contribution vector, x is the node's local value and $w_{t,i}$ is the weight. The relative error at node i is $\frac{|(\frac{v_{t,i}.x}{w_{t,i}}) - \frac{1}{n}\sum_j x_j|}{|\frac{1}{n}\sum_j x_j|}$. By applying the triangle equality under sum (Holder's Inequality), we obtain

$$\begin{aligned}\frac{|(\frac{v_{t,i}.x}{w_{t,i}}) - \frac{1}{n}\sum_j x_j|}{|\frac{1}{n}\sum_j x_j|} &= n.\frac{|(\frac{v_{t,i}}{\|v_{t,i}\|_1} - \frac{1}{n}.1).x|}{|\sum_j x_{x_j}|} \\ &\leq n.\frac{\left\|\frac{v_{t,i}}{\|v_{t,i}\|_1} - \frac{1}{n}.1\right\|_\infty . \|x\|_1}{|\sum_j x_j|} \\ &\leq \epsilon.\frac{\sum_j |x_j|}{|\sum_j x_j|}\end{aligned}$$

The relative error in the estimate of average at any node i is at most $\epsilon.\frac{\sum_j |x_j|}{|\sum_j x_j|}$. So the relative error is at most ϵ when values of x_j have the same sign. To get sum only one node start with weight $w = 1$ and the value computed at the nodes converges to sum. Since the calculations of regression coefficients are additively decomposable we get a convergence of the coefficients to the true global coefficients.

The overall approach of decomposition of the steps for solving normal equations and presenting an iterative model for evaluating the model parameters without having to store the entire data at a single place is the foundation of the algorithm. The algorithm proposed reduces the problem to relatively simple primitive computation and makes use of the distributed asynchronous property to get the desired results for distributed and peer-to-peer systems as against centralized computations. Smart Grids connecting multiple power production and consumption nodes with embedded sensors of various types capable of data storage, retrieval and computations can be seen as sensor networks with decentralized architectures. These data repositories stored at these different nodes are hard to centralize. Traditional off-the-shelf algorithm will not work in such environment since they typically requires first centralizing the data for subsequent analysis. Even a client-server architecture-based distributed system where the server sends the query to the different databases for accessing the data for subsequent analysis is unlikely to scale because of the large volume, latency and also sensitive aspect of the data in some cases. The following algorithm has the intrinsic property of distributed computation and thus is deployable on nodes of such smart-grids.

Algorithm 1: Distributed Asynchronous Regression (DAR)

```
Initialization:$X_n^m$ {The data matrix having m features and n instances}
Compute $L_{ij(t)}$, $p_{i(t)}$, $w_{i(t)}$
while a node receives a message do
    $L_{ij}^{new}$, $p_i^{new}$, $w_i^{new} \leftarrow receive()$
    Compute $L_{ij(t+1)}$, $p_{i(t+1)}$, $w_{i(t+1)}$
end while
for $\phi(n)$ number of times do
    choose a neighbor $nbr$ uniformly at random
    send $L_{ij(t+1)}$, $p_{i(t+1)}$, $w_{i(t+1)}$ to $nbr$
end for
return $w_{i(t)}$
```

The distributed asynchronous regression algorithm (DAR) is shown in algorithm table 1. In the given algorithm each node starts with a $n \times m$ dimensional data matrix X_n^m where m is the number of features and n is the number of instances. Each peer N_i computes its own values of $L_{ij(t)}$, $p_{i(t)}$, $w_{i(t)}$. The message that each node now sends comprises of the above calculated values and similarly each node is equipped with a $receive()$ construct that is set to listen for these values from its neighboring nodes. Once a node receives the new values they are labeled as L_{ij}^{new}, p_i^{new}, w_i^{new}. The values of $L_{ij(t+1)}$, $p_{i(t+1)}$, $w_{i(t+1)}$ are calculated as described in Section 6. To send a message to another node, the current node chooses a neighbor uniformly at random and sends its current calculated values. The number of rounds of such iterations is given by the function $\phi(n)$, where n is the number of nodes in the network as mentioned. The algorithm terminates giving the approximate coefficients of linear regression.

Theorem 1. The space complexity of the algorithm DAR running on each node is of order $O(m^2)$ where m is the number of features and is independent of the number of data tuples.

Proof. As shown in Section 5 Equation 3 the dimensionality of covariance matrix A is $O(m^2)$, the matrix containing the entries $L_{ij(t)}$ is of dimension m^2 from Equations 4 and 5, and the vectors $p_{i(t)}\, and\, w_{i(t)}$ is of dimension m from Equations 6 and 7. At every iteration for calculating coefficients we store and update each of $L_{ij(t)}$, $p_{i(t)}$ and $w_{i(t)}$ and compute new $L_{ij(t+1)}$, $p_{i(t+1)}\, and\, w_{i(t+1)}$ so, the total complexity is given by $O(m^2) + O(m) + O(m) = O(m^2)$. □

The efficiency of this process is due to the fact that usually $m << n$ and the calculation of every iteration of coefficients is a mixture of *lookups* and already calculated components for each computation.

Theorem 2. The communication cost for this algorithm DAR is of order $O(log\, n)$ number of messages where n is the total number of nodes in the network and messages passed are of constant size.

Proof. The messages passed by each node is a vector of size of the dimension of the data tuples as shown in Section 6. The size of such messages is equal to the number of coefficients calculated for the linear regression which is determined by the dimensionality of the data tuples. So the payload of each message is constant dependent on the number of features of the data tuples. On the other hand the number of messages passed before the algorithm converges within ϵ relative error with a probability at least $1 - \delta$ for a network of n nodes is of the order $O(log\, n + log\frac{1}{\epsilon} + log\frac{1}{\delta})$ given by Theorem 2.1 in Kempe et al[16]. □

In gossip-based protocols, each node contacts one or a few nodes in each round and exchanges information. The guarantees obtained from gossip are probabilistic in nature. The key issue is how many rounds are needed for the local values to converge to a global value with sufficient accuracy. The averaging time, which is the time required for the number of rounds to achieve a desired level of accuracy of a gossip algorithm turns out to be closely related to the mixing time of Markov chain which is the time until the Markov chain is close to its steady state distribution defined by a random walk on the graph. The convergence rate is the speed with which this sequence reaches its value. Boyd et al. [6] found that convergence speed depends on the second largest eigenvalue of the stochastic matrix and proposed a subgradient method to optimize the neighbor selection probabilities for each node in order to find the fastest mixing Markov chain on the graph. This method is proved to work on complete graphs, expander graphs and peer-to-peer networks. The averaging time for a randomized gossip algorithm defined by transition matrix P is given by

$$T_{ave}(n,\epsilon) = \underbrace{sup}_{x(0)}\, inf \left\{ k : P\left(\frac{||x(k) - x_{ave}\vec{1}||}{||x(0)||} \geq \epsilon \right) \leq \epsilon \right\}$$

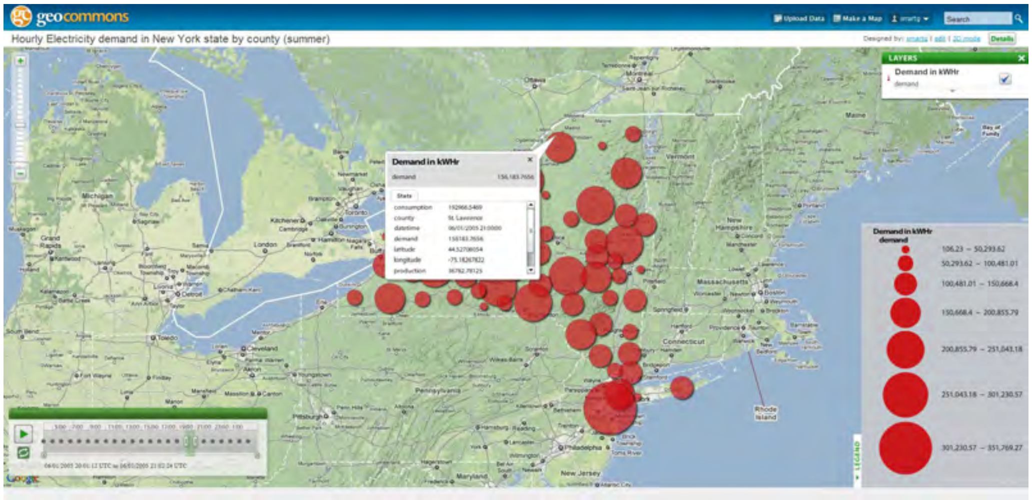
such that $\epsilon > 0$, the ϵ-averaging time is the earliest time at which vector $x(k)$ is ϵ close to the normalized true average with probability greater than 1 - ϵ and $||.||_2$ denotes l_2 norm and $||x(k) - x_{ave}\vec{1}||/||x(0)||$ is referred as the relative error after k rounds.

Theorem 3. The number of rounds of gossip R_{conv} or the rate of convergence of the regression coefficients w_i for the proposed algorithm DAR is bounded by the number of nodes n, the relative error ϵ and the second largest eigenvalue of the transition matrix for the given set of nodes in the network and is expressed by $R_{conv} \leq \frac{\log n + \log \epsilon^{-1}}{1-\lambda_2}$.

Proof. The rate of convergence of Markov chain whose transition matrix is given by P is characterized by the mixing time of a Markov chain with transition matrix $\tilde{P}$ where $\tilde{P} = \frac{1}{2}(I - P)$ as shown in Boyd et al. [6]. We know the mixing time of a Markov chain can be bounded in terms of its eigenvalues. Given P a stochastic matrix and defining $\tilde{P}$ implies it is also stochastic and has largest eigenvalue $\lambda_1 = 1$ and remaining eigenvalues non-negative. It also follows that the mixing time of the Markov chain, which is the measure of the amount of time needed to guarantee each coefficient calculated at each each node is within the ϵ-error of the centralized value, with transition matrix $\tilde{P}$ is bounded in terms of its second largest eigenvalue λ_2 and is given by $T_{mix}(\epsilon, \tilde{P}) \leq \frac{\log n + \log \epsilon^{-1}}{1-\lambda_2}$, Diaconis et al.[9] where $T_{mix}(\epsilon, \tilde{P})$ is the ϵ mixing time or number of rounds of gossip given the transition matrix $\tilde{P}$. Given the knowledge of the number of nodes in the network, the second largest eigenvalue of the transition matrix we see that the rate of convergence of the regression coefficients or the number of rounds of gossip R_{conv} following the DAR algorithm is characterized by the mixing time $T_{mix}(\epsilon, \tilde{P})$ and so R_{conv} can be upper bounded by the same expression and is given as $R_{conv} \leq \frac{\log n + \log \epsilon^{-1}}{1-\lambda_2}$. □

7. Experimental Results

7.1. **Experimental Setup.** We have implemented our algorithms in Distributed Data Mining Toolkit(DDMT)[8] developed by the DIAIDC research lab at UMBC. We use topological information generated by the *Barabasi Albert(BA)* model in BRITE [7] since it is often considered a reasonable model for peer-to-peer infrastructure. On top of the network generated by BRITE, we simulate gossip based communication.



Demand in kWHr

demand 156,183.7656

Stats

consumption 192966.5469
county St. Lawrence
datetime 06/01/2005 21:00:00
demand 156183.7656
latitude 44.52708054
longitude -75.18267822
production 36782.78125

Figure 2. Simulated hourly electricity demand in the state of New york by County in a typical day of Summer.

7.2. **Experimental Data.** The data is collected from the consumption section of Residential Energy Consumption Survey (RECS)[1] 2005 which is a national area-probability sample survey that collects energy related data for occupied housing units. The data attributes included housing unit characteristics like mobile homes, single family detached house and apartment buildings etc., the number of people living in each household, the average energy consumption per house by dryer, dishwasher, refrigerator and other electric appliances. From the yearly consumption data we first characterized the usage in seasonal pattern of summer and winter. We seperated electricity consumption data like that of air-conditioning which is more used in summer and room heating and water heaters which are more used in winter to simulate household consumption behavior. We then brought down the seasonal consumption behavior to per day depending on the time of day introducing peak energy usage during morning and evening and off-peak usage rest of the time in the day. To simulate variational usage we again introduced white Gaussian noise to get the hourly consumption. From the RECS data we choose to simulate the hourly energy consumption of households for the state of New York.

For the production data by photovoltaic(PV) cell we used System Advisor Model(SAM) [2] simulator. With default system array size of 35^2mm per house hold and taking weather conditions like temperature, wind speed, shading factor we simulated the hourly production data for the same region. The demand by region was simulated by finding the difference between the production and consumption per hour for a typical summer day for the state of New York. The simulated demand model over a period of one day is hosted on web service found elsewhere.[2]

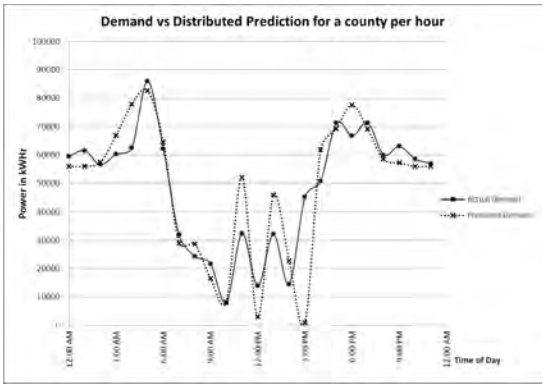


Figure 3. Actual demand vs distributed prediction for a county per hour.

7.3. **Simulation Results.** The experiments were carried out for approximately 7 million residential homes for the state of New York grouped by 62 counties. The consumption and production data were simulated on an hourly basis. From this data for a typical summer day we ran the DAR algorithm to get the predicted demand and plotted it against the actual demand from the simulated data. The results of which are presented in the graphs below. In Figure 3 we have the time of the day on the x axis and the power demand on the y-axis.

The dotted lines represents the predicted demand obtained from the DAR algorithm and the solid lines represent the actual demand from the simulated data. In Figure 4 we have the same axes but here we compare the actual demand and the predicted demand obtained by centralized method. Figure 5 we compare the demand prediction by the centralized method and our proposed distributed asynchronous method. In Figure 6 we show the comparison between communication cost in terms of bytes of data that gets transferred for the centralized approach and distributed approach. As seen from the figures the x axis represents the problem size, here the number of households that are producing the data and the y axis represents the size of the message that are transferred in the network. As seen from the figure the communication cost increases linearly with increase in size for the centralized algorithm whereas communication cost increases logarithmically for the distributed algorithm making it suitable for scalable applications. From the demand prediction figure we see that as the number of rounds of running the algorithm increases for each node the prediction value asymptotically reaches the centralized prediction thereby presenting a small and acceptable error margin.

[2] http://geocommons.com/maps/38838

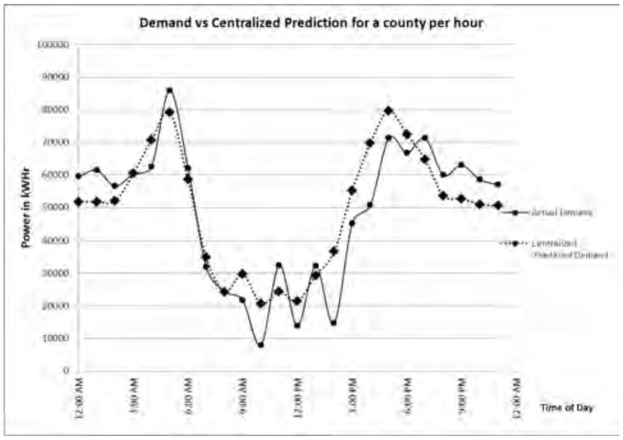


Figure 4. Actual demand vs centralized prediction for a county per hour.

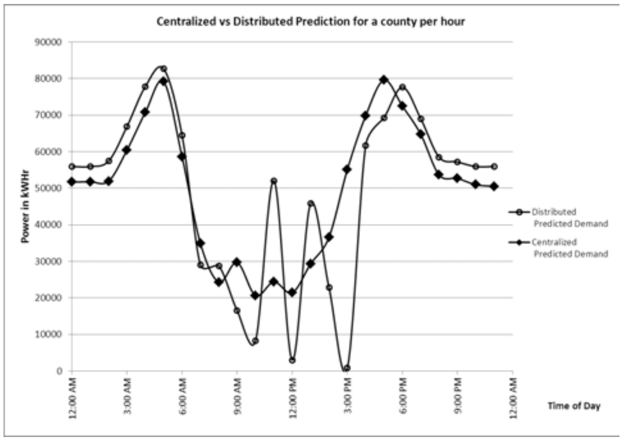


Figure 5. Distributed prediction vs centralized prediction for a county per hour.

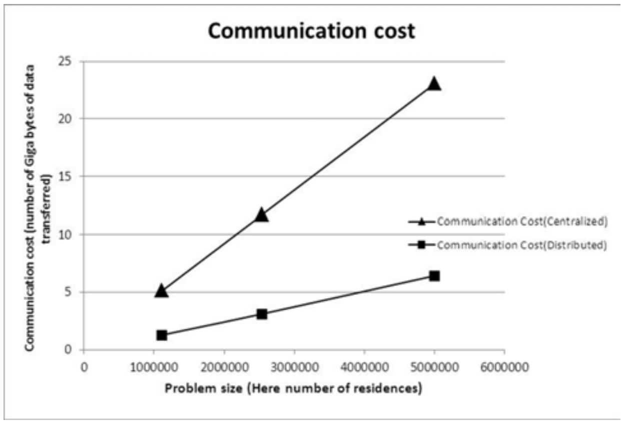


Figure 6. Communication cost for the centralized vs distributed approach.

8. Conclusion

This paper explored the problem of predicting energy demand in a distributed smart grid environment. It argues that the centralized approach to learn and monitor such predictive models is not sustainable for both business and technical reasons. Participation of different business entities, household production of energy, and privacy-sensitive power consumption data are some of the reasons since they often prohibit centralized collection of the the observed data. This paper considers a multi-variate regression-based approach for predicting energy demand and offers a novel technique for learning linear regression models in a distributed and asynchronous way. It analyzes the performance of the algorithm and offers experimental results. Overall, the paper demonstrates that a distributed asynchronous algorithm can be used to learn predictive models over smart grids in a scalable manner.

Acknowledgment

This research is supported by NASA Grant NNX07AV70G.

References

[1] Residential energy consumption survey (recs). http://www.eia.doe.gov/emeu/recs/recspubuse05/pubuse05.html.

[2] System advisor model. https://www.nrel.gov/analysis/sam/.

[3] N. S. P. 1108. *NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 1.0*, January 2010.

[4] M. Bawa, H. garcia Molina, A. Gionis, and R. Motwani. Estimating aggregates on a peer-to-peer network. Technical report, Stanford University, 2003.

[5] K. Bhaduri and H. Kargupta. An efficient local algorithm for distributed multivariate regression in peer-to-peer networks. In *SIAM International Conference on Data Mining*, pages 153–164, 2008.

[6] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Gossip algorithms : Design, analysis and applications. In *INFOCOMM*, pages 1653–1664, 2005.

[7] BRITE. http://www.cs.bu.edu/brite/.

[8] DDMT. http://www.umbc.edu/ddm/Sftware/DDMT.

[9] P. Diaconis and D. Stroock. Geometric bounds for eigenvalues of markov chains. *The Annals of Applied Probability*, 1(1):36–61, 1991.

[10] S. K. Domnic Savio, Lubomir Karlik. Predicting energy measurements of service-enabled devices in the future smartgrid. In *12th International Conference on Computer Modelling and Simulation*, 2010.

[11] H. Dutta, D. Waltz, A. Moschitti, D. Pighin, P. Gross, C. Monteleoni, A. Salleb-Aouissi, A. Boulanger, M. Pooleery, and R. Anderson. Estimating the time between failures of electrical feeders in the new york power grid. In *NGDM09*, 2009.

[12] W. Fan. Mining techniques to sustainable and efficient transportation system - challenges and solutions. Technical report, NGDM09, 2009.

[13] C. P. Gomes. Computational sustainability : Computational methods for a sustainable environment, economy, and society. *The Bridge, National Academy of Engineering*, 39(4), 2009.

[14] C. Guestrin, P. Bodi, R. Thibau, M. Paski, and S. Madden. Distributed regression: an efficient framework for modeling sensor network data. In *IPSN*, pages 1–10, 2004.

[15] D. E. Hershberger and H. Kargupta. Distributed multivariate regression using wavelet-based collective data mining. *Journal of Parallel and Distributed Computing*, 61(3):372–400, 2001.

[16] D. Kempe, A. Dobra, and J. Gehrke. Computing aggregate information using gossip. In *FOCS*, 2003.

[17] E. Maibach. Climate change and energy use:examining the american people and the drivers of population behavior. Technical report, NGDM09, 2009.

[18] D. S. Scherber and H. C. Papadopoulos. Distributed computation of averages over ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 23(4):776–787, 2005.

[19] Y.Ma, L. Zhou, N. Tse, A. Osman, and L. L. Lai. An initial study on computational intelligence for smart grid. In *International Conference on Machine Learning and Cybernetics*, 2009.

P-MATCH and QUBIT – Methods for Extracting Critical Information from Free Text Data for Systems Health Management

ANNE KAO[1], STEPHEN POTEET[2], DAVID AUGUSTINE[3]

ABSTRACT. In order to achieve integrated aircraft health management, multiple data sources regarding aircraft maintenance as well as aviation safety have to be studied together to achieve a holistic view. While some data sources contain only structured data as found in standard relational databases, free text data often provide critical information that cannot be fully expressed in structured data. The wealth of information goes beyond what can be achieved by coding the free text into a set of fixed categories. The text mining problem is fundamentally different from casual exploratory search of the Internet. Analysts need to extract detailed information with high recall from very noisy text data, with lots of non-standard spellings and abbreviations. The extraction task cannot overlook analysts' important domain knowledge. In this paper, we will examine two data sources, one from airplane maintenance logs involving aircraft parts and systems, and the other from the Airplane Safety Reporting System (ASRS), a collection of anonymous aviation safety self-reports involving operational issues. We will illustrate innovative methods we have developed that can aid analysts in extracting critical information in these two different data sources. Our methods use a combination of natural language processing and string matching, and we suggest ways of using machine learning and interactive feedback to provide an easy way for the analysts to utilize their domain knowledge to improve the system and achieve systems health management goals.

1. INTRODUCTION

In order to maximize the safety as well as the economy of a vehicle's performance, whether it is an airplane, automobile or spacecraft, it is crucial to take an integrated view of all vehicle related data. These include on-board sensor data, scheduled and unscheduled maintenance records and reports, reports related to operations of the vehicle, and other environmental data (such as weather condition and how congested a highway or an airport is). In order to achieve success in vehicle health management and keeping operating conditions economic, all of these various factors have to be examined together. The volume of data is vast and the data sources are diverse. On the other hand, the data often have missing and incorrect values. Data mining techniques should be able to offer us a plethora of useful approaches to analyzing this data. However, a lot of technical domain knowledge (e.g. engineering design and manufacturing) is required when applying data mining techniques to this problem. The complexity in the data goes far beyond the cliché market analysis of products and customer demographical data.

In this paper, we will focus on airplane health management. While examples used are drawn from aviation, readers will find that most discussions carry strong commonalities to other types of vehicle health management. Furthermore, while the number of data sources supporting a complete airplane health management system go well beyond the short list mentioned above and is easily in the hundreds, we will focus our discussion on two types of data sources in particular, namely, aviation safety reports and airplane maintenance log book data. These data sources contain both structured and unstructured data. However, the wealth of information is in the unstructured free text data. Our discussion will further focus on analysis of the free text data, and how that can be mined and used in conjunction with structured data.

[1] Boeing Research & Technology, anne.kao@boeing.com
[2] Boeing Research & Technology, stephen.r.poteet@boeing.com
[3] Boeing Research & Technology, david.c.augustine@boeing.com

2. GENERAL PROBLEM DESCRIPTION

Airplane maintenance data and aviation safety data represent two very different types of information and technical challenges. The volume of maintenance data is huge. For one major airplane model (e.g. 747 or 777) at one major airline, there can be over 100,000 maintenance records. The maintenance data can come from an airline, or from the Boeing organized consortium ARMS ISDP (Airplane Reliability and Maintainability System In-Service Data Program), which contains maintenance data from more than twenty major airlines. The data contains unscheduled maintenance data and, sometimes, scheduled maintenance data. For the current discussion, we will focus on log book data, which is the record of line maintenance (i.e. maintenance at the airport gate). The free text fields include (1) the complaint text which records problems reported by the pilot, crew or mechanics, and (2) the resolution text which records the actions taken to fix each problem. The guideline is that each record should record one single problem; however, this is not always true, partly due to the complexity of the problem.

Aviation safety data are collected through open source news reports, internal airline reports, airline reports of operating issues to the manufacturer, and self-reporting by individuals to the Airplane Safety Reporting System (ASRS) managed by NASA. It is important to distinguish these data from accident investigation data. Through major efforts by airplane manufacturers and operators, airplanes are extremely safe and there have been very few accidents. In the rare cases an accident does occur, there would be an in-depth forensic analyses and report on what happened. However, the goal of health management is not to focus on these few accidents, but to analyze near accidents and other unexpected aviation events to identify potential risk areas in order to further reduce the chance of accidents. Unfortunately, this task is impeded by the fact that safety data is often somewhat sketchy and lacks important details. For example, ASRS reports contain a fair amount of operational issues that relate to weather, airport condition, and crew and pilot activities. However, they only contain the year and month of the reports, and not the specific date of the event, and do not give the airline involved, let alone the flight. This is by design to make the information anonymous and protect the reporter’s identity, but it makes it hard to match the report with more specific airport or weather conditions that obtained at the time of the incident, other than what the report explicitly states. There is a lot less safety data than maintenance data; however, it is much harder to categorize all of the records into a fixed set of categories that would be both comprehensive enough to cover all cases and detailed enough to distinguish specific issues.

In order to better understand human factors issues and operational issues, it is crucial to study the safety data. However, in order to ascertain system or part reliability, or spot potential metal fatigue and long term wear and tear, it is important to mine the maintenance data.

Both maintenance and safety data suffer from missing values, as well as duplications and pseudo-duplications. A maintenance record can report on multiple actions performed on a single part or even actions performed on multiple parts (even though the guideline is one part per record). A maintenance problem can be deferred if not flight critical, resulting in multiple records for the same problem so that a simple search would over-identify a problem. Worse, a maintenance problem can take more than one try to get fixed, even though it is closed out after the first try under the mistaken belief that it is resolved. In this case, there is nothing that ties the records together; they are treated as unrelated problems by the system. Similarly, since safety reports are largely self reports, the same problem can be reported multiple times by different people, often with very diverse points of view. The lack of key discriminating information makes it hard to identify duplicates with complete confidence. Missing values are a common problem for both maintenance data and safety data. Different airlines have different practices on how to fill in various data fields. For example, how many times a non-flight critical issue is identified but deferred is not always filled out. Since safety data is self-reporting data, the reporting person may not always fill in information such as the flight phase when the event occurred (take off, cruising, or landing).

3. P-MATCH

In this section, we will illustrate how to use a combination of knowledge-based natural language processing and various string matching algorithms to solve a high impact part name analysis problem.

3.1. The part name reference problem. The log data provides documentation on what problems have occurred and what maintenance actions have been performed. Reliability studies of parts typically are interested in all of the maintenance actions performed. The result helps determine a recommended maintenance schedule for airlines, and how a specific fleet or a specific airplane deviates from its class. Alternatively, a real-time health management application may be only interested in fix effectiveness and thus only focus on the actions that successfully fix each of the problems reported. In addition, the data can be used to study the reliability of parts, proactively identify required upcoming maintenance, and support supply management to avoid over-stocking or under-stocking issues. For this type of application, only replacements would be of interest, and resetting, cleaning or adjusting actions would not be. In a complex business such as the airline industry, it can also help operation management; for example, by keeping the airline from sending an airplane to a station that is not equipped to perform any likely upcoming maintenance.

As noted above, a major part of the problem is that there are numerous spelling and ad hoc abbreviation problems in the data. Here is an example of a part name term, ‘computer’, and some of the variations on how it is expressed in the log data:

computer, comptr, compter, computor, compuer, computo

Furthermore, one variant, ‘comp’, is ambiguous and, in addition to “computer”, can mean:

compressor, compartment, compensator

Compounding the problem, a part name typically consists of multiple words, each exhibiting many variants, and the words in the part name from a given list may not all occur in a maintenance record or may occur in a different order. For example, the part name:

Overhead Panel Bus Controller (L), M23112 (P11)

may occur in a maintenance record as:

COMPLAINT: REF ADD 913 STS MSG ***LEFT O/H PNL BUS CONTROL*** *INTERMITTENT TAGS OFF K02648Y*
RESOLUTION:FIM ACTIONED AS PER MSG 23-48802 ***OPBC*** *REPLACED IAW MM 23-93-01 GRND CHKS AND TESTS C/OUT SATIS TAGS ON B25092G*

The rendering of the part name in the COMPLAINT text leaves off the equipment number (M23112) and the panel (P11) the part is located on (it being redundant with “left” in this case), realizes “L” as “left” and removes the parentheses and relocates it to the beginning of the part name. In the RESOLUTION text, the whole thing is reduced to an acronym, “OPBC”, and any indication of location (“left” or the panel) is left off (since one of these has been mentioned in the COMPLAINT text). Note that it is the less informative of these, the acronym, that is adjacent to the maintenance action performed (“replaced”). All of this makes trying to find a good match even harder.

Clearly searching for an exact match of the string representing the part name is not going to help. While using a synonym list could help with acronyms like “O/H” and “OPBC”, it would not be a very good way of handling all the misspellings and ad hoc abbreviations illustrated by the “computer” example above. The number of parts is very large (in the thousands) and number of spelling variations for the words in part names is also very large (often 10-20 and sometimes over 40). This would make constructing an exhaustive list highly labor intensive if possible at all. Furthermore, it is also hard to foresee what types of spelling variations might occur, given that one major type of error is to fuse two words together (leaving out the space in between). Traditional natural language processing approaches, which would try to parse the sentences of the log record, would fare even worse, since, in addition to requiring an entry in the lexicon for each misspelling, they would have to associate them with grammatical information as well (like what part-of-speech they are) [1].

3.2. Our solution. Our solution is to combine knowledge-based natural language processing techniques and various string matching techniques. We call our approach Partname Matching by Analysis of Text Characteristics, or P-MATCH.

First, we will briefly summarize our approach. We begin with a given list of the part names that we are interested in, the “target part name list”. This is typically a subset of all the parts on an airplane, for example, the parts that can be repaired or replaced during a brief stopover between flights, i.e. the “line replaceable units” or LRUs. We take advantage of the structure of part names, so the first step is to parse the part names on our list according to this structure. Next, for each maintenance record, we use a list of the maintenance actions we are interested in along with the results of the part name parsing to identify a candidate part name string, a sequence of words that likely refers to a part that we are interested in. Then we use one or more fuzzy string matching algorithms to compare this candidate part name string with the part names on the target part name list. We then use the results to select the most likely part name from the target list.

Inspired by linguistic analysis of noun phrases as basically consisting of a head noun and modifiers, we parse part names into a head noun that tells us what general type of part it is (e.g. a switch), essential modifiers that determine the specific type of part it is (e.g. “outflow valve” in “outflow valve switch”), and peripheral modifiers that primarily indicate location (e.g. “aft” in “aft outflow valve switch”). The head is typically the last word in the part name, but sometimes the last word is so general that it is likely that log data may leave it out completely and the second to the last word should be treated as the head, for example “Control Display Unit”, will often be described simply as a “control display” or some variant thereof. In addition, certain types of modifiers may occur after the head in the target list of part names, though they are typically easy to recognize by their form (e.g. alphanumeric) or the fact that they occur in parentheses or after a comma. For example:

Control Display Unit (Center), N34303 (P11)

“N34303” is the equipment number of this part, rather than a part of the part name per se. “Center” is the location and a peripheral modifier. “P11” is the panel it is located on and is typically not included in the log data or, if it is, not necessarily adjacent to the part name proper.

Having parsed the target part names into heads and modifiers, the system then examines the log data. There are a number of ways in which we can start, depending on the task. One task we might be asked to perform with P-MATCH would be to find parts that had been removed, replaced or repaired to help analyze fix effectiveness. For this, we typically begin by searching for variants of the verbs representing these particular maintenance actions. By starting our search with the verb we were able to make the search more efficient. We not only narrow the number of records that have to be examined in detail, but we also anchor our search in the text for the string representing the part name. Note that there can be quite a few variations of these verbs. Some of the variants of “replace” in the data are:

replaced, repl, rplaced, replacd, replced, repaced, replaed, replacement, replac, replaced, rplcmnt, replace, repla, replae

However, it is worth noting that the total number of verbs involved is an order of magnitude smaller than the total number of part names involved. We find this list of variants by searching a list of words extracted from a large set of the data using a fuzzy string matching algorithm.

Often a maintenance record will contain multiple verbs (e.g. the engineer checked one part, adjusted another part, and finally replaced yet another part). When dealing with fix effectiveness, we are typically interested in the last verb in the maintenance message based on the fact that, in process descriptions, actions are typically described in the same temporal order as they occur in the real world (authors of maintenance reports never write "before we replace X, we reseated X") and we want the final action performed. An exception to this is that variants of “checked OK” frequently follow the action of replacing/removing/repairing, so it would be ignored. Another way of finding the appropriate part in a message to analyze fix effectiveness would be to organize maintenance actions into a prioritized set of categories. For example, the categories and their priority might be: removed/ replaced, repaired, deferred, reset, OK, other. In this task, we find all the verbs in the maintenance message and then select the one with the highest priority as the starting point from which to search for the candidate part name. On the other hand, if we are trying to assess part reliability, we may want to find all the parts mentioned whatever the action performed was. In this case, we search for all the verbs and find associated part names near them, often yielding multiple parts per maintenance message.

The next step is to identify a candidate part name string from the maintenance record. The candidate string is a sequence of words in the maintenance text that is likely to name a part in the target part name list. In general, part names are contiguous strings of part name modifiers followed by a part name head; however, there are exceptions to this. As noted above in the case of the “left overhead panel bus controller”, a fuller description of the part (e.g. the location “left”) may occur earlier in the record, or even in a different field (COMPLAINT text) from the action verb that indicates which part we are actually interested in (typically located in the RESOLUTION text). These different descriptions of the part need to be synthesized. In addition, references to location (“right”, “left”, “forward”, “aft”, “Zone A”, panel number) frequently occur after the head (e.g. “ECS CARD R”, where “R” is “right”). Given this, we search first for a part name head in the vicinity of the verb, then search for possible modifiers of that head adjacent to it. Since the verb can be active or passive (e.g. “replaced XXX valve” or “XXX valve replaced”), both sides of the verb are searched. The head and its associated modifiers are derived from the parse of the target part name list described earlier. Because of the possibility of misspelling, the head and the modifiers are searched using a fuzzy string matching criteria. After these modifiers are collected, the rest of the message is searched for other occurrences of the head (or a variant of it, e.g. “vlv” for “valve”) and, if one is found, additional modifiers adjacent to it are collected. Finally, the head and all of its modifiers are synthesized into a single part name candidate string.

Occasionally, peripheral modifiers, in addition to occurring after the head can also occur in non-adjacent positions. For example, in one maintenance message there is a reference to the “Upper Flow Control and Shutoff Valve (L Pack), V21511 (P110)”, but “LEFT PACK” occurs in the COMPLAINT text and “UPPER FLOW CONTROL VALVE” occurs in the RESOLUTION text. While it might be possible to extend the search for these peripheral modifiers to catch some of these, there is always the chance that the location does not refer to the part we are looking for, but to some other part. One way to deal with this would be to ignore or down-weight these location indicators during matching with part names from the target list.

The resulting part name candidate string is then compared with all of the part names on the target list that have the same (or equivalent) head using another fuzzy matching algorithm. We are currently using a subset of the algorithms in SecondString [2]. Similarity algorithms can be divided into several different categories: string, token, and hybrid or two-level [3]. String-based algorithms process the entire string as a sequence of characters, applying one or more approaches to measure the dissimilarity, such as Levenshtein (edit-distance), Jaro (number and order of common characters), Jaro-Winkler (Jaro with added score for common initial substrings), or Jaro-Jones (which takes into account common number-letter substitutions, such as numeral ‘0’ and letter ‘O’). Token-based algorithms divide the string into tokens using one or more ‘space’ or punctuation characters, then compare tokens ignoring their order. Examples include Jaccard (number of tokens common to both strings divided by the number of tokens in either string), TF-IDF (cosine similarity of strings represented as vectors of Inverse “Document” Frequency weighted token frequencies), and the Jensen-Shannon distance (based on the Jensen-Shannon divergence of probability distributions of tokens [4]). Mixed or two-level algorithms tokenize the string, and then apply string similarity algorithms to the individual tokens. Examples include Level2Jaro (the average similarity of matching tokens using the Jaro algorithm) and Soft TFIDF (Jaro-Winkler applied to the individual tokens and TF-IDF applied to the tokens above a certain similarity level). For a more complete description of the wide variety of string similarity algorithms that are available, see [5], [6], and [7]. For Jaro-Jones see [8]. The results of our initial experiments used Jaro, Jaro-Jones, Jaccard, Jensen-Shannon, and Level2Jaro see [9].

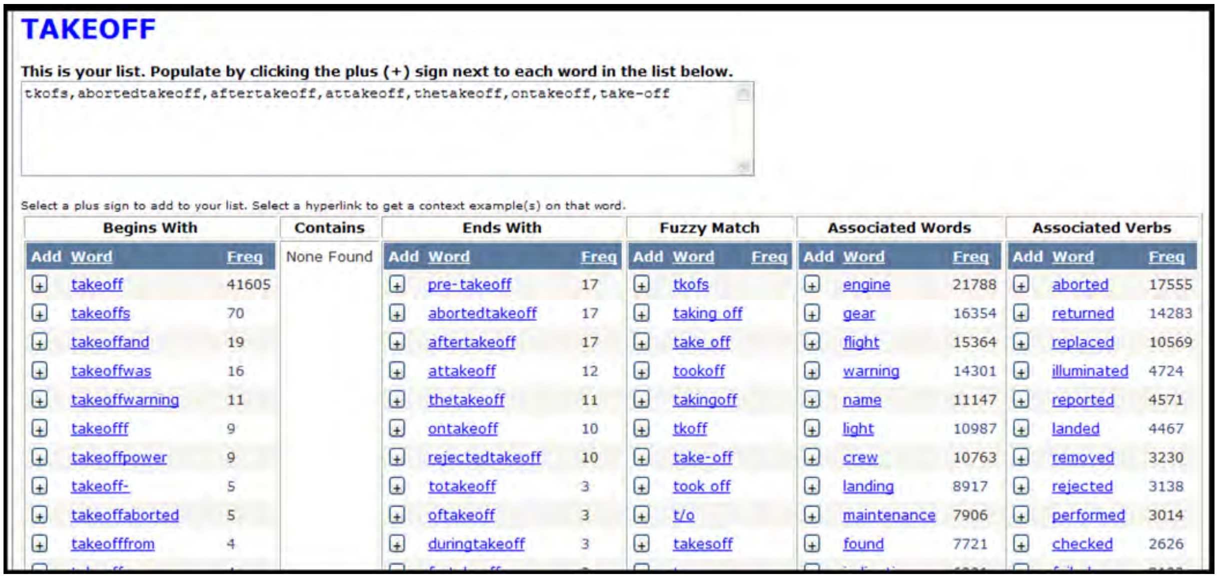
4. QUBIT

In this section, we will illustrate how to support a user’s complex searches by offering a combination of text mining generated suggestions and knowledge compiled suggestions, together with information management techniques to further assist users to research high impact problems involving aviation systems, structures, and operations.

4.1. The ad-hoc query problem. Analysts of health management systems do not always have the luxury of a predefined list of items on which they wish to perform standing queries, like the part-name problem described above. For example, new investigations of aviation safety related issues usually demand users to come up with complex queries that have not been previously performed in short demand. The search may involve parts and systems as well as operational issues and environmental issues (such as volcano eruption). While users are subject matter experts who possess a high degree of engineering knowledge, it is very difficult for them to figure out all of the spelling variations and ad hoc ways of expressing the same concept or the same part or system. The problem goes further. It is not atypical for a complex query to include more than 100 conjuncts and disjuncts in an SQL statement. Trying to rapidly refine queries of this level of complexity to get the accuracy required for the task is also a very daunting task. QUBIT (QUery BuIlder Tool) is designed to solve this problem using two integrated features, which we describe below.

4.2 Suggester. We utilize a number of methods to help the user come up with variations in search terms so that the users can focus on the search concept based on their engineering knowledge. It should be noted that this is not the same as providing a synonym list, which many text databases support. There are several major differences. First, a synonym list is typically obtained by having users enter the knowledge manually or by loading it from an existing knowledge base. While a knowledge base can be, and is, one source of input for a QUBIT Suggester, the goal is to use text mining techniques wherever it makes sense to extract knowledge from the data and minimize the manual work. Secondly, while it is possible for a user to edit the synonym list, a text database is not designed to allow the user to select or reject entries based on the goal of the search or the nature of the data being analyzed. In contrast, by design our suggester provides multiple suggestions for the same term. The user has the option of choosing what makes sense in the current search context.

Some of major methods a suggester may employ are discussed below. In all cases except the knowledge base, the text collection to be searched will be used as sample data for the Suggester to extract suggestions. Depending on the size of the complete set of data, and the processing speed required to make the suggesting process interactive, the whole data set or a subset of it will be used as the sample data for this purpose. Typically, more than one term is used in a search. The Suggester will make suggestions for each search term. It is also possible to have a multiple word search term (e.g. ‘main landing gear’) and the Suggester will treat this as a single term. Figures 1 and 2 show two possible incarnations of the Suggester illustrating some of the functions a Suggester might perform.



TAKEOFF

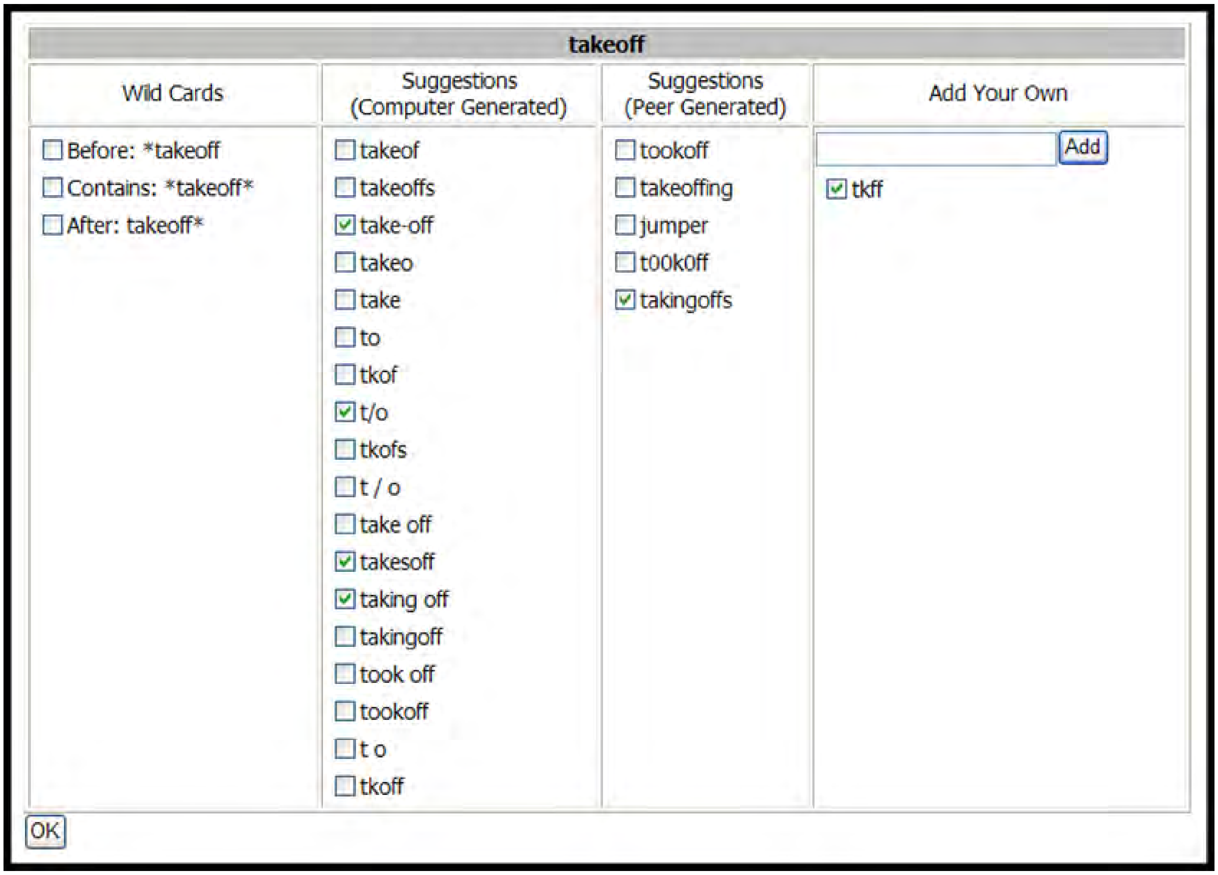
This is your list. Populate by clicking the plus (+) sign next to each word in the list below.

```
tkofs,abortedtakeoff,aftertakeoff,attakeoff,thetakeoff,ontakeoff,take-off
```

Select a plus sign to add to your list. Select a hyperlink to get a context example(s) on that word.

Begins With			Contains	Ends With			Fuzzy Match			Associated Words			Associated Verbs		
Add	Word	Freq	None Found	Add	Word	Freq	Add	Word	Freq	Add	Word	Freq	Add	Word	Freq
⊞	takeoff	41605		⊞	pre-takeoff	17	⊞	tkofs		⊞	engine	21788	⊞	aborted	17555
⊞	takeoffs	70		⊞	abortedtakeoff	17	⊞	taking off		⊞	gear	16354	⊞	returned	14283
⊞	takeoffand	19		⊞	aftertakeoff	17	⊞	take off		⊞	flight	15364	⊞	replaced	10569
⊞	takeoffwas	16		⊞	attakeoff	12	⊞	tookoff		⊞	warning	14301	⊞	illuminated	4724
⊞	takeoffwarning	11		⊞	thetakeoff	11	⊞	takingoff		⊞	name	11147	⊞	reported	4571
⊞	takeofff	9		⊞	ontakeoff	10	⊞	tkoff		⊞	light	10987	⊞	landed	4467
⊞	takeoffpower	9		⊞	rejectedtakeoff	10	⊞	take-off		⊞	crew	10763	⊞	removed	3230
⊞	takeoff-	5		⊞	totakeoff	3	⊞	took off		⊞	landing	8917	⊞	rejected	3138
⊞	takeoffaborted	5		⊞	oftakeoff	3	⊞	t/o		⊞	maintenance	7906	⊞	performed	3014
⊞	takeofffrom	4		⊞	duringtakeoff	3	⊞	takesoff		⊞	found	7721	⊞	checked	2626

Figure 1. QUBIT Suggester, Sample Version 1



takeoff

Wild Cards	Suggestions (Computer Generated)	Suggestions (Peer Generated)	Add Your Own
□ Before: *takeoff	□ takeof	□ tookoff	[ ] Add
□ Contains: *takeoff*	□ takeoffs	□ takeoffing	☑ tkff
□ After: takeoff*	☑ take-off	□ jumper	
	□ takeo	□ t00k0ff	
	□ take	☑ takingoffs	
	□ to		
	□ tkof		
	☑ t/o		
	□ tkofs		
	□ t / o		
	□ take off		
	☑ takesoff		
	☑ taking off		
	□ takingoff		
	□ took off		
	□ tookoff		
	□ t o		
	□ tkoff		

Figure 2. QUBIT Suggester, Sample Version 2

4.2.1 Finding terms with a high co-occurrence with the search term. The Suggester processes the sample data and displays a list of terms that co-occur with the search term, along with frequency counts, (depicted under the "Associated Words" column in Figure 1). Given a list of particular words (e.g. verbs), instances of those associated with the target term can also be shown, (shown in the "Associated Verbs" column in Figure 1). Depending on the data and the application, three more parameters can be set to control this. (a) A proximity window can be set so that, for example, only words that are no more than five words away from the search term are returned. This will allow the suggester to only return co-occurring terms near the search term and therefore more likely to be related to the search term. As a special case, by setting the window to be one, only terms immediately adjacent to the search term will be returned. (b) The directionality of the window can be controlled, so that only co-occurring terms to the left or to the right of the search term are returned. Of course, this can be combined with a proximity window. Based on the syntax of English, the terms immediate to the left of a search term which is a noun will tend to be modifiers of the search term. (c) Finally, a frequency threshold can be set to limit the co-occurring terms to be displayed to the user to those with at least a certain frequency. This can be used to prevent the Suggester from returning random examples. The Suggester ranks the results by frequency and displays them to the user with a count next to each co-occurring term.

4.2.2 Using fuzzy string matching algorithms to generate a potential match.
The Suggester can use string matching algorithms, such as those described in the P-MATCH section, to generate terms similar to the search term, (as depicted in- the "Fuzzy Match" column in Figure 1). Parameters like those discussed under P-MATCH can be employed. These include the type of string matching algorithm to use, a similarity threshold, or the maximum number of results to be returned by the string matching algorithms. The Suggester can have an option to display the similarity measure or the rank of the results returned.

4.2.3 Using regular expressions to generate potential matches. Some analysis tools allow user to use regular expressions to identify additional search terms [10]. However, , this is a very daunting task that is hard for subject matter experts to take on. The Suggester takes a different approach. It automatically generates certain regular expressions for each search term and finds matches in the sample data to display to the user. This is the most powerful way to deal with words fused together by error, which is very typical in noisy text data sets. The Suggester can identify all "words" in the sample data beginning or ending with the search term, or containing it in the middle, without any space in between (as shown in the columns "Begins With", "Ends With", and "Contains" in Figure 1). Alternatively, the Suggester might just match all words in any or all of these categories, as shown in the "Wild Cards" column in Figure 2. Similarly, the suggester can return examples of words with a space arbitrarily inserted into them, which frequently occurs in the data (possibly as a result of the data being moved between data entry systems and data storage systems). The suggester can also make a regular expression substituting the letter "l" with the numeral "1", if the letter "l" occurs in the search term, and similarly the letter "o" with the numeral "0". There are many other possibilities here. For example, the suggester can replace the ending of the word with a wild card, because words are frequently abbreviated in ad hoc ways. Alternatively, regular expressions can be used to search for words that are missing one or more vowels, since that is another common means of abbreviating. The power of the Suggester here is to allow users to benefit from the use of regular expressions, without expecting them to construct regular expressions themselves.

4.2.4 Using latent semantic analysis or other methods to generate terms which are semantically or topically related. The Suggester can use latent semantic analysis [11], for example TRUST [12], to generate terms highly correlated to the search term using the sample data. Semantically highly related terms can be identified using this method or similar text mining methods. For example, it might find words that can occasionally be used in place of each other (i.e. synonyms or near synonyms or words that are otherwise closely related semantically but not in terms of spelling) such as "blower" and "fan", "circuitboard" and "motherboard", or "wiring" and "harness", if they are used sufficiently frequently and in similar contexts.

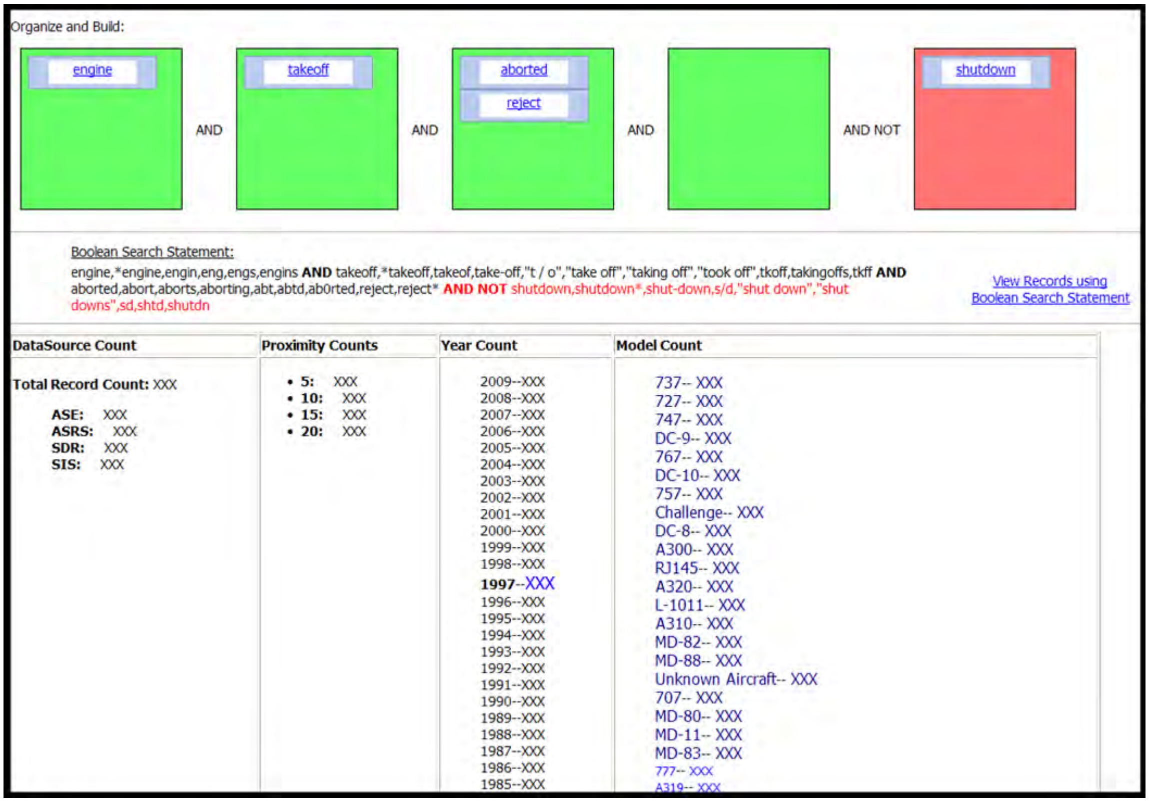
4.2.5 Using a knowledge base. The Suggester can use any knowledge base that is available. This includes what a text database manager may return as a suggestion for a search term. Typically, this is a good method to handle acronyms as well as true synonyms that do not look anything like the original term, both of which are not always easy to identify with the above methods. We are using synonym and acronym lists such as those produced by FAA-ASIAS as well as a Boeing internal thesaurus and acronym list.

4.2.6 Using suggestions entered by peers. The Suggester can also allow users to use terms that have been entered by their peers using the system. There is an option for a user to enter a variant of a term that they know of or have discovered in their search of the data but that has not been suggested by the system. The system will then provide this as a peer generated suggestion to other users, as shown under "Suggestions (Peer Generated)" in Figure 2. This provides a way for the system to let users leverage the expertise and experience of their peers.

4.3 Query Builder. Users in this domain rarely search with just one term. After using the Suggester to determine which terms best express the concepts they are interested in, the user can now concentrate on constructing the actual query based on these concepts. At this point, each concept will consist of a disjunction of terms. Depending on the application domain, there may be an advantage to further categorizing the search concepts. For example, in an application for fix effectiveness, the user often wants to organize the concepts into the following categories: part (or system), the condition of the part, the action taken in fixing the problem, and the result of the fix. In other applications, the search may be less structured and the system would offer a more flexible way to organize the concepts. Either way, the users can now put these concepts into boxes representing different conjuncts of their query and possibly one or more negative concepts which, if found, would exclude a record from being returned. Note that if two concepts are placed in the same conjunct box (depicted as a green box in Figure 3), the system will treat this as a disjunct of those two concepts. For example, the query expressed in Figure 3 is:

"engine AND takeoff AND (aborted OR reject) AND NOT shutdown"

where each of "engine", "takeoff" etc. are concepts consisting of a disjunction of terms, typically synonyms, abbreviations, acronyms, misspellings etc. The system will then generate a complex machine executable query based on the user's selections. The user has the option of editing the generated complex search, though this is not typically done.



Boolean Search Statement:

engine,*engine,engin,eng,engs,engins **AND** takeoff,*takeoff,takeof,take-off,"t / o","take off","taking off","took off",tkoff,takingoffs,tkff **AND** aborted,abort,aborts,aborting,abt,abtd,ab0rted,reject,reject* **AND NOT** shutdown,shutdown*,shut-down,s/d,"shut down","shut downs",sd,shtd,shutdn

DataSource Count

Total Record Count: XXX

ASE: XXX
ASRS: XXX
SDR: XXX
SIS: XXX

Proximity Counts

- **5:** XXX
- **10:** XXX
- **15:** XXX
- **20:** XXX

Year Count

2009--XXX
2008--XXX
2007--XXX
2006--XXX
2005--XXX
2004--XXX
2003--XXX
2002--XXX
2001--XXX
2000--XXX
1999--XXX
1998--XXX
1997--XXX
1996--XXX
1995--XXX
1994--XXX
1993--XXX
1992--XXX
1991--XXX
1990--XXX
1989--XXX
1988--XXX
1987--XXX
1986--XXX
1985--XXX

Model Count

737-- XXX
727-- XXX
747-- XXX
DC-9-- XXX
767-- XXX
DC-10-- XXX
757-- XXX
Challenge-- XXX
DC-8-- XXX
A300-- XXX
RJ145-- XXX
A320-- XXX
L-1011-- XXX
A310-- XXX
MD-82-- XXX
MD-88-- XXX
Unknown Aircraft-- XXX
707-- XXX
MD-80-- XXX
MD-11-- XXX
MD-83-- XXX
777-- XXX
A319-- XXX

Figure 3. QUBIT Query Builder

The Query Builder provides a GUI to help the user formulate her query in a form that is inspired by conjunctive normal form (though not exactly the same). While conjunctive normal form is a classic concept in computer science, to the best of our knowledge, it has only been used in very few cases to support search [13] and no one has employed it in an interactive system to support complex search requiring in-depth domain knowledge. The combination of Suggester and Query Builder allows the user to focus on the concept instead of getting lost in the details of the query syntax.

5. CONCLUSION

In the paper, we have discussed the need to exploit text data in addition to sensor and other numerical and categorical data in order to improve vehicle systems health management. We have discussed some of the most salient problems with the data, mainly the fact that it is extremely noisy, with frequent ad hoc abbreviations and spellings. We have discussed in detail P-MATCH, an approach we have developed to find mentions of part names in this type of data. P-MATCH illustrates how a combination of knowledge-based natural language processing techniques in conjunction with various string matching algorithms, mostly numerical, can leverage the strength of both. The following characteristics are inspired by linguistics-based NLP: (1) Target part names are parsed into head nouns, core modifiers and peripheral modifiers. (2)Based on the fact that in English noun phrases non-phrasal modifiers precede the head noun, our algorithm searches to the left of the head for modifiers to include in the candidate part name, with the data-motivated exception that peripheral modifiers, primarily indicating location, can occur more freely. (3) We use the fact that in English process descriptions, actions are typically described in the same temporal order as they occur in the real world (authors of maintenance reports never write "before we replace X, we reseated X") to determine the final action taken. P-MATCH also illustrates how to solve part name extraction, an important type of entity extraction problem with some different characteristics from typical person and place name extractions. Today's entity extraction tools with person names and place names as their primary focus cannot adequately deal with the special characteristics of specific part name matching. While P-MATCH is applied to the airline industry data today, it can benefit any industry with a large number of parts and large amount of log data.

In addition, we have also discussed QUBIT, a method of supporting interactive ad hoc queries that exploit and rely on the user's domain knowledge. We have discussed how the combination of the user's domain knowledge with the use of a Suggester can be a quite effective and how a structured but flexible GUI can guide the user in constructing complex queries.

REFERENCES

[1] J. F. Allen. Natural language understanding (second edition). Menlo Park, CA, Benjamin Cummings, 1994.

[2] SecondString Project Page. http://secondstring.sourceforge.net/. Retrieved February 20, 2010.

[3] W. W. Cohen, P. Ravikumar, and S. E. Fienberg. A Comparison of String Distance Metrics for Name-Matching Tasks. In Proceedings of the ACM Workshop on Data Cleaning, Record Linkage and Object Identification, Washington DC, 2003.

[4] Wikipedia Jensen-Shannon Entry. http://en.wikipedia.org/wiki/Jensen%E2%80%93Shannon_divergence. Retrieved November 7, 1020.

[5] W. W. Cohen. Record Linkage Tutorial: Distance Metrics for Text. 2006. http://www.cs.cmu.edu/~wcohen/Matching-2.ppt. Retrieved February 20, 2010.

[6]W. E. Winkler. Overview of Record Linkage and Current Research Directions. Research Report Series, Statistics #2006-2, U.S. Census Bureau. http://www.census.gov/srd/papers/pdf/rrs2006-02.pdf, 2006.

[7] SimMetrics Project Page. http://www.dcs.shef.ac.uk/~sam/simmetrics.html. Retrieved February 20, 2010.

[8] M. Ankerst and D. H. Jones. System and method for string distance measurement for alphanumeric indicia. U.S. Patent 7,540,430, filed Sept 27, 2005, and issued June 2, 2009.

[9] A. Kao, S. Poteet, and D. Augustine. Extracting Critical Information from Free Text Data for Systems Health Management. In **Data Mining in Systems Health Management: Detection, Diagnostics, and Prognostics.** Chapman & Hall/CRC Press, Boca Raton, FL, to appear.

[10] Wikipedia Regular Expression Entry. http://en.wikipedia.org/wiki/Regular_expression. Retrieved November 7, 1020.

[11] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. Indexing by Latent Semantic Analysis. Journal of the American Society for Information Science, 41(6):391-40, 1990.

[12] A. Booker, M. Condliff, M. Greaves, F. B. Holt, A. Kao, D. J. Pierce, S. Poteet, Y.-J. J. Wu. Visualizing Text Data Sets. IEEE Computing in Science & Engineering, 1(4):26-35, July 1999.

[13] M. Hearst. Improving Full-Text Precision on Short Queries using Simple Constraints. Proceedings of the Fifth Annual Symposium on Document Analysis and Information Retrieval (SDAIR), Las Vegas, NV, April 1996.

DATA GUIDED DISCOVERY OF DYNAMIC CLIMATE DIPOLES

JAYA KAWALE, STEFAN LIESS, ARJUN KUMAR, MICHAEL STEINBACH, AUROOP GANGULY*,
NAGIZA F. SAMATOVA**, FRED SEMAZZI**, PETER SNYDER, AND VIPIN KUMAR

Abstract. Pressure dipoles in global climate data capture recurring and persistent, large-scale patterns of pressure and circulation anomalies that span distant geographical areas (teleconnections). In this paper, we present a novel graph based approach called shared reciprocal nearest neighbors that considers only reciprocal positive and negative edges in the shared nearest neighbor graph to find dipoles in pressure data. To show the utility of finding dipoles using our approach, we show that the data driven dynamic climate indices generated from our algorithm always perform better than static indices formed from the fixed locations used by climate scientists in terms of capturing impact on land temperature and precipitation. Another salient point of this approach is that it can generate a single snapshot picture of all the dipole interconnections on the globe in a given dataset making it possible to differentiate between various climate model simulations via data driven dipole analysis. Given the importance of teleconnections in climate and the importance of model simulations in understanding the impact of climate change, this methodology has the potential to provide significant insights.

1. Introduction

The Earth is known to exhibit continued changes in atmospheric and ocean circulation by which thermal energy is distributed on the surface of the Earth and which brings about changes in weather and climate on the globe. *Teleconnections* are recurring long distance patterns of climate anomalies related to each other at large distances. Such teleconnections have proven to be important for understanding and explaining climate variability in many regions. Typically, these teleconnections are represented by time series known as *climate indices* [3], which are often used in studies of the impact of climate phenomena on temperature, precipitation, and other climate variables. For instance, the El Niño-Southern Oscillation (ENSO) index captures sea surface temperature (SST) variability in several locations at once; the Pacific-North American teleconnection pattern relates to the El Niño phenomenon, which in turn enables prediction of rainfall, snowfall, droughts, or temperature patterns with a few weeks to a few months lead time in North America. One important class of climate indices are *pressure dipoles*, which are characterized by pressure anomalies of opposite polarity appearing at two different locations at the same time.

Scientists have known of the existence of such dipoles for about a century [26, 16]. Two of the best known pressure dipoles are the North Atlantic Oscillation (NAO) and the Southern Oscillation (SO). NAO, which is traditionally described by the difference in anomalies in sea level pressure (SLP) between Akyureyri in Iceland and Ponta Delgada in the Azores, captures the large-scale atmospheric fluctuations between Greenland and Northern Europe. It was first observed in 1770-1778 [23] and was labeled NAO in 1924 [27]. The Southern Oscillation Index (SOI) is measured as the difference in SLP anomalies at Tahiti and Darwin, Australia and captures fluctuations in SLP around the tropical Indo-Pacific region that correspond to the El Niño Southern Oscillation (ENSO) phenomenon [24]. These dipoles are defined by static locations but the underlying phenomenon is dynamic. Many of the dipoles (e.g., SO, NAO) have been discovered by examining the local data at specific locations. Such manual discovery can miss many dipoles. Ever since the satellite data became widely available in the early 1970s, pattern analysis such as EOF analysis has been used to identify individual dipoles and the climate indices over a limited region, such as Arctic Oscillation (AO) index [25]. However,

*kawale,liess,arkumar,steinbac,snyder,kumar @cs.umn.edu ** gangulyar@ornl.gov *** nagiza, fred_semazzi @ncsu.edu.

there are several limitations associated with EOF and other types of eigenvector analysis; namely, it only finds a few of the strongest signals and physical interpretation of such signals can be difficult.

In this paper, we present a novel graph based approach to discover dipoles using a Shared Reciprocal Nearest Neighbor(SRNN) algorithm. Our approach allows us to detect all dipoles represented in an individual global dataset within the selected time frame and to determine their individual strengths. It makes it possible to discover new dipoles that may not have been seen. It enables tracking the movements of these dipoles and studying their interactions in a much more systematic way. Another important application of global dipole analysis is in the understanding of the skill of various General Circulation Models (GCMs) used for climate prediction. Various GCM models exhibit variability in their predictions of various climate variables, as they use different representations of physical interactions in the climate system. Hence they often diverge in their predictions and sometimes even offer contradicting projections of changes in various regions in response to different greenhouse gas emission scenarios. Our current approach provides a comprehensive view of the dipoles on Earth and, hence, a power to test various models in terms of their ability to capture dipoles. Despite the prevalence and importance of teleconnections in climate science and climate related impacts, an adequate study quantifying the teleconnections in the climate models is still lacking. Similarities or differences in dynamic dipole structure can offer valuable insights to climate scientists on model performance, which further aids in assessing reliability of climate prediction simulations.

1.1. **Related Work and Motivation.** Steinbach *et al.* [18, 19, 20] showed the utility of using Shared Nearest Neighbor(SNN) algorithm to find known climate indices. At first, a climate graph was constructed in which each node represents a region of the Earth and an edge between a pair of nodes represents pairwise correlation between the anomaly time series of the corresponding regions. The clusters were found in the climate graph using SNN and some of the centroid of the clusters corresponded to known climate indices. Further some pairs of discovered clusters also showed high correlation with many SLP based climate indices defined as dipoles. Other researchers, including Tsonis *et al.* [22], Donges *et al.* [6] and Steinhaeuser *et al.* [21], studied the behavior of climate graphs as complex networks and showed correspondence between features in climate graph and major teleconnection patterns such as NAO.

Kawale *et al.* [12] formally defined the notion of a dipole in the context of a climate graph and presented a dipole detection algorithm that focused on the negative correlations in contrast to the previous approaches that either used positive[18] or absolute value correlations[22, 6, 21]. The approach also showed better correlation with the static indices and area-weighted impact on the land anomalies as compared to [18, 19, 20]. Kawale *et al.* noted that many more positive edges than negative edges exist in the climate graphs and most of these positive edges are uninteresting, as they are between nearby regions and thus primarily due to spatial autocorrelation. In contrast, every significant negative correlation represents a potentially interesting teleconnection. Negatively weighted, or simply negative, edges in the climate network can be crucial for finding dipoles, as the dipole regions, by definition, have opposite polarity anomalies. The approach was based upon picking up the most negative edge in the complete graph and building regions around it so that the two regions are negatively connected across each other but within them they are positively connected to each other. The approach then iteratively removed the edges of the dipole found from the climate network to find dipoles in the data.

There are several shortcomings with the iterative algorithm in [12]. First, it is computationally expensive, as it works in edge space. Figure 1 shows the distribution of negative edges in the NCEP/NCAR reanalysis data. At a 2.5 degree resolution, there are about 10000 nodes and 55 million edges. Out of them, there are about 1 million edges with edge weights below -0.4. [1] If we consider a higher threshold, there will be fewer negative edges but we may miss many dipoles. Some

[1] With a finer 0.5 degree resolution, the number of nodes increases 25 times and the number of edges increases even further.

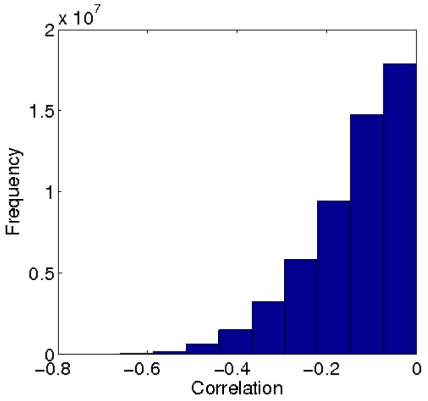


Figure 1. Degree plot of the Negative edges around the globe.

dipoles are inherently weaker in nature as compared to the others. For example, the SOI dipole is much weaker than the AO dipole and most of the negative edges spanning SOI regions have correlation much weaker than the -0.4 threshold used to limit the number of edges[2] Our proposed approach as does simultaneous clustering of nodes instead of iteratively removing edges and that significantly brings down the amount of computation. Second, the number of candidate dipoles generated by the algorithm in [12] is enormous, as we only remove the edges from the climate network that have already been included in dipoles discovered in previous iterations. This results in many surrogate dipoles for a single dipole. Third, the algorithm uses various parameters. There were four parameters in the algorithm. Our current approach has only a single parameter K.

1.2. **Our Contribution.** The main contributions of our paper are as follows:

(1) We present a novel graph based approach to find dipoles in any spatio-temporal data that overcomes the shortcomings of the previous approaches [18, 12].
(2) Our approach allows us to have a single snapshot of all the dipoles on the globe. This was not possible using the previous best approach[12]. The new approach enables us to discover new dipoles and comprehensively study the behavior, interaction, and movement of various dipoles in a more precise manner.
(3) We show an application of dipole analysis to understand the differences between GCMs which are used for climate change prediction.

2. Background and Data Preliminaries

2.1. **Dataset.** We use sea level pressure (SLP) data from the NCEP/NCAR Reanalysis project as well as from the output of the GCMs. SLP is used to find the dipoles because most of the important climate indices are based upon pressure variability. The NCEP/NCAR reanalysis project is a gridded dataset of 2.5 degree resolution for all locations on the Earth created using a mix of observations and interpolations to have data for all the grid points on the Earth. The data spans 1948—present and there are 10512 grid points in the 2.5 degree resolution data. We use monthly mean values for the 60 years of data (corresponding to 720 monthly values). The NCEP/NCAR reanalysis is provided by the NOAA/OAR/ESRL PSD, Boulder, Colorado, USA [11], available for public download at [4].

We use simulation output data from six of the more than 20 general circulation models (GCMs) from the Fourth Assessment Report (AR4) of the IPCC [15]. These models produce large-scale,

[2] In order to find such weaker dipoles, in [12] data smoothing was necessary as a preprocessing step which adds another parameter - the degree of smoothing to be used. In contrast, our proposed approach is able to find all the dipoles using the raw data without any smoothing.

mainly physics-based simulations of the coupled atmosphere/ocean system for understanding the climate and projecting climate changes. Complex climate model simulations are run by about 20 laboratories across the world to make predictions of anticipated future changes in climate and to inform the Intergovernmental Panel on Climate Change (IPCC) [15]. Compared to weather forecast models, which are used for forecasts up to 10 days, these GCMs are designed to make stable projections over many decades or even centuries. GCMs might therefore not capture individual weather events and dynamic oscillations, but only very large-scale patterns and the overall state of the global climate. GCMs predict a global temperature increase over the next century in response to increased greenhouse gas concentration in the atmosphere [15]. In order to evaluate the overall model skill, simulations start in a known period of the past and model results can be compared to readily available observations for this period. This process is known as hindcast. As the model simulations continue into periods beyond present time, where no observations are available yet, the results are known as forecast. A list of the six GCM models used in our study is as follows - CCCMA CGCM 3.1 (Canadian Centre for Climate Modelling and Analysis) , GISS Model E-H (NASA Goddard Institute for Space Studies), CSIRO 3.0 (Commonwealth Scientific and Industrial Research Organisation) , GFDL 2.1 (Geophysical Fluid Dynamics Laboratory), BCCR BCM2.0 (Bjerknes Centre for Climate Research) and UKMO HadCM3 (Hadley Centre for Climate Prediction and Research).

2.2. **Seasonality Removal.** Most of the data in Earth Science is associated with a strong seasonality due to the Earth's revolution. The seasonality forms the strongest signal and it masks out other signals in the data. In order to take care of the seasonality, we construct anomaly time series from the raw data by removing the monthly mean values of the data. This is done as follows:

$$\mu_m = \frac{1}{end - start + 1} \sum_{y=start}^{end} x_y(m), \forall_{m \in \{1..12\}}$$

$$x_y(m) = x_y(m) - \mu_m, \forall_{y \in \{1948..2009\}}$$

In this equation, start and end represent the start and end years to consider for the mean and define the base for computing the mean for subtraction (in our case 1948 and 2009). μ_m is the mean of the month m and $x_y(m)$ represents the value of pressure for the month m and year y. Once we remove the monthly means, the resulting values are the anomaly time series for that location.

2.3. **Network Construction .** Once we get the anomaly series from the raw pressure data, we construct a complete graph out of the data using the approach used earlier by [22, 6, 21, 18, 12] by taking the pairwise correlation between the anomaly time series of all pairs of location on the Earth. The nodes in the graph represent locations on the Earth and the edges represent the correlation between the anomaly timeseries of the two locations on the Earth.

2.4. **Notation.** We represent the undirected weighted graph as $G = (V, E)$, where $V = \{V_1, V_2, ..., V_N\}$ represent the N $(= |V|)$ vertices in the graph and E is a N X N matrix in which cell $E_{i,j}$, $1 \leq i, j \leq N$, indicates the edge weight between vertices V_i and V_j. For every vertex V_i the set $S_i = \{V_{i_1}, V_{i_2}, ..., V_{i_{N-1}}\}$, where $i_1, i_2, ..., i_{N-1}$ is a permutation of the set $\{1, 2, ..., N\} \setminus i$, such that, $E_{i,i_1} \geq E_{i,i_2} \geq ... \geq E_{i,i_{N-1}}$. Let $KNN_i^+ = \{V_{i_1}, V_{i_2}, ..., V_{i_K}\}$ and $KNN_i^- = \{V_{i_{N-K}}, V_{i_{N-K+1}}, ..., V_{i_{N-1}}\}$. The edges from V_i to nodes in KNN_i are referred to as extremal edges.

3. Our Approach

Dipoles are defined as a pair of regions such that locations within each region are highly positively correlated with each other and locations across these regions are negatively correlated to each other. To find dipoles we use a clustering approach that groups together locations on the globe that are similar in terms of i) the locations to which they are most strongly negatively correlated and ii) locations to which they are positively correlated. This first requirement is motivated by the centrality of negative correlation in the definition of dipole, while the second helps to produce spatially contiguous clusters since nearby locations tend to have positive correlations. These clusters can serve as the ends of dipoles and the set of all possible pairs are further evaluated to yield candidate dipoles. Since regions involving dipoles can be of different size, shapes and strength, we use a clustering scheme based on the shared nearest neighbor concept that is particularly effective in addressing such requirements. In this section, we define our approach based upon shared reciprocal neighbors to find the dipoles in climate data.

We model the climate data as an undirected weighted graph $G^C = (V^C, E^C)$, where V^C is the set of nodes representing grid locations on the Earth and E^C is the set of undirected edges between these locations. The edge weight represents the correlation between the anomaly time series of the locations, such that, positive edge weight between two locations indicate that they experience a similar climatic phenomenon and negative edge weight indicates that they exhibit an opposite climatic phenomenon. Our algorithm to compute dipoles consists of four major steps, mentioned as follows:

- Step 1: Construction of reciprocal graph G^R from the climate data graph G^C. This involves forming the list of k nearest positive and negative neighbors of each object using the original similarity measure, where k is a parameter chosen by the user and considering only the edges that are reciprocal, i.e. which lie on each other's nearest neighbor list.
- Step 2: Construction of Shared nearest neighbor graph (G^{SNN_-} and G^{SNN_+}). This is done by redefining the similarity of each pair of objects in terms of the number of their common (shared) nearest reciprocal neighbors.
- Step 3: Merging of G^{SNN_-} and G^{SNN_+} to construct G^{SRNN} graph.
- Step 4: Finding dipoles using density based clustering on G^{SRNN}.

The further details of the algorithm are mentioned as follows.

3.1. **STEP 1: Construction of Reciprocal Graph.** We begin by considering the original graph $G^C = (V^C, E^C)$ as described in Section 2.3. We construct the reciprocal graph $G^R = (V^C, E^R)$, where $E^R \subseteq E^C$ as follows:

$$E^R_{i,j} = \begin{cases} 1 & \text{if } V^C_i \in KNN^+_j \wedge V^C_j \in KNN^+_i \\ -1 & \text{if } V^C_i \in KNN^-_j \wedge V^C_j \in KNN^-_i \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

The main idea behind reciprocal is to pick the K highest positively and negatively correlated locations (*extremal* set) corresponding to a given location and then consider an edge between two locations if they appear in each other's extremal set. From the definition of dipoles, we know that any two regions that actually form dipoles would be in each other's negative extremal set and the nodes within a region would be in their positive extremal set. The benefits of computing the reciprocal graph is manifold: Firstly, it reduces the size of the original graph drastically (asymptotic upper bound of reduction is $\theta(N/K)$ but in practice it is much more); Secondly, it filters noise (such as anomalous locations or regions, weakly correlated locations).

Corollary 3.1. The graph G^R achieves $\theta(N/K)$ reduction in the number of edges over G^C. This is easy to see. Since every node in G^R has at most $2 * K$ neighbors, the number of edges in G^R are $\theta(N * K)$. The number of edges in G^C are $\theta(N * N)$.

Note that building the reciprocal graph is essential to eliminate spurious inter-connections between the locations. The concept of reciprocity holds more importance in negative correlations than in positive correlations as in spatial data, due to autocorrelation, nearby objects are very similar and hence reciprocity exists by nature in positive correlations. But for negative correlations, reciprocity is much more meaningful and helps in weeding out spurious negative correlations. Consider the location Tahiti which is a part of the SO dipole. The KNN^- and the reciprocal edges coming out from Tahiti are shown in the figure 2. From the figure, we see that Tahiti has many edges going to the North pole in the KNN^-, however only the ones going to Darwin in Australia survive in the reciprocal graph.

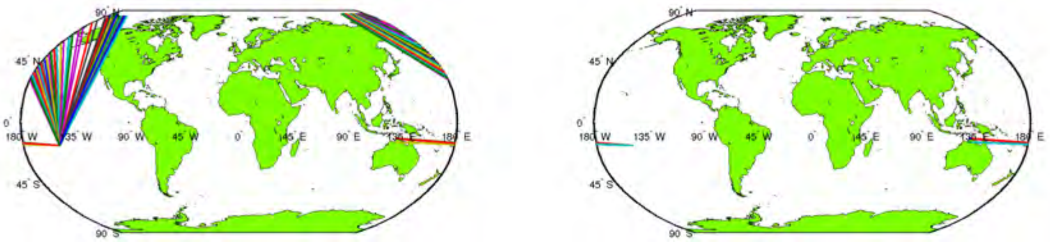


FIGURE 2. All KNN^{-} and only reciprocal edges from Tahiti using K=50

3.2. **STEP 2: Construction of Shared Nearest Neighbor graph (G^{SNN_-} and G^{SNN_+}).** The reciprocal graph G^R retains the edges which are mutually extreme (highly positive or negative) between all the location pairs. G^R essentially captures the dipole regions and their inter-connections yet the extraction of these regions require us to cluster graph nodes into set of regions. Additionally, clustering helps us in identifying spurious regions that result due to a small number of spurious extremal edges (making G^R robust to any choice of $K << N$). We propose a variant of SNN algorithm [18] for clustering the reciprocal graph. The main idea of SNN algorithm is to form groups based on how many shared neighbors two nodes have in the graph. It is important to note that the SNN algorithm alone could not extract the most precise dipoles, as suggested by prior work [12]. This motivates us to propose the following variant of SNN algorithm. We construct two graphs $G^{SNN_+} = (V^C, E^{SNN_+})$ and $G^{SNN_-} = (V^C, E^{SNN_-})$ by running SNN algorithm on positive and negative edges of G^R, respectively. More formally, the edge weights of the two graphs are estimated as follows:

$$E^{SNN_+}_{i,j} = |\{k : V^C_k \in V^C \wedge E^R_{i,k} = 1\} \cap \{k' : V^C_{k'} \in V^C \wedge E^R_{i,k'} = 1\}| \tag{2}$$

$$E^{SNN_-}_{i,j} = |\{k : V^C_k \in V^C \wedge E^R_{i,k} = -1\} \cap \{k' : V^C_{k'} \in V^C \wedge E^R_{i,k'} = -1\}| \tag{3}$$

Equations 2 and 3 estimate the number of shared neighbors two nodes have and considers the number of shared neighbors as the edge weight. The motivation behind two separate graph is that since a node can have two types of neighbors in G^R; those with $+1$ edge weights and others with -1 edge weight. As a result, these neighbors need to be counted separately. It is crucial to treat the two types of edges separately because otherwise a single application of SNN algorithm would allow locations that are close to one of the dipole regions to have significant edge weight even when they do not participate in the dipole phenomenon. The next step in which we combine the above two graphs makes it clear why a single application of SNN would not yield qualitatively and quantitatively good dipoles. In equation 2 and 3, we simply count the number of shared neighbors between all location pairs. Instead, we can also compute a weighted sum, where the weights take into account the ranks of the shared nearest neighbors from the two lists (see [10]). This idea allows us to compute the edge weight as a weighted sum of the reciprocal links shared between the nearest neighbor list of the two nodes. The weight is computed by taking the mean of the ranked order of the reciprocal links in the two neighbor lists. The weighted version performs slightly better than the counting version and we use it throughout to present our results.

Overall, nodes with high edge weights in G^{SNN_+} indicate two things. Firstly, the two locations, corresponding to the nodes, share positive correlation in their climate and this correlation is high for both the nodes (guaranteed by G^R). Secondly, these nodes are part of a cluster where this positive climate phenomenon is maximal (counting of positive neighbors, equation 2). In practice, this cluster corresponds to spatially co-located places on Earth. Similarly, G^{SNN_-} gives us a sense of which negative regions these nodes associate with. It is possible for two nodes to have high edge weight in one graph and yet a low or 0 edge weight in other graph; forming the basis of the next step of our algorithm.

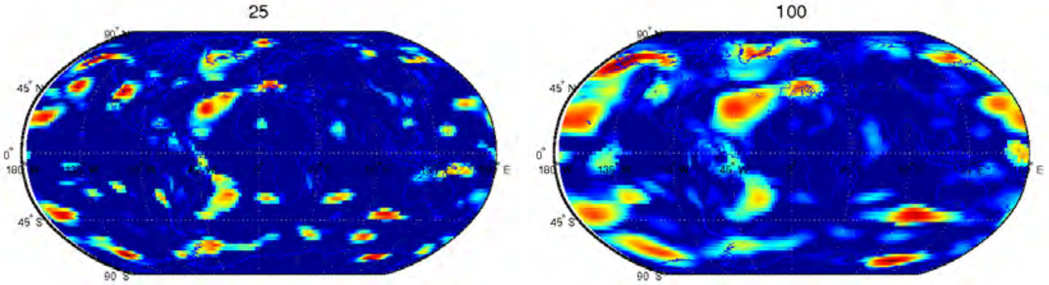


FIGURE 3. Dipoles discovered using our algorithm for $K = 25, 100$ (density plot of sum edge weight of nodes in G^{SRNN}). The red regions represent the regions of high density and the blue regions represent regions of low density.

3.3. **STEP 3: Merging of G^{SNN_-} and G^{SNN_+} to construct G^{SRNN} graph.** The two graphs G^{SNN_+} and G^{SNN_-} form graph components or cliques with high inter clustering coefficient than intra clustering coefficient. It is possible for two nodes to have high edge weight in one graph yet a very low edge weight in other. To illustrate this consider two geographically close points; one inside one end of a dipole (say x) and other outside it (say y). Indeed, x and y would share high positive correlation on climate variables (such as air pressure, temperature) due to spatial autocorrelation. As a result it is possible for the two nodes to have moderate to high $E_{x,y}^{SNN_+}$. On the other hand, the point y would not have very high negative correlation with the other end of the dipole region corresponding to x, as it is not a part of the dipole. It is also possible that two regions have a high edge weight $E_{x,y}^{SNN_-}$ and a low edge weight $E_{x,y}^{SNN_+}$ which indicates that the two locations are spatially distant and cannot be a part of the same end of the dipole. Hence a single application of SNN in step 2 does not yield good results (because then both point x and y are claimed to be part of the dipole region). The example presented above presents an intuitive justification of our merging criteria: multiply the edge weight of G^{SNN_-} and G^{SNN_+} to form $G^{SRNN} = (V^C, E^{SRNN})$. More formally,

$$E_{i,j}^{SRNN} = E_{i,j}^{SNN_-} * E_{i,j}^{SNN_+} \tag{4}$$

Note that the only parameter that our algorithm uses is K (defined in Section 3.1) to compute the extremal edges. A large choice of K would result in a lot of spurious connections, where a small choice of K would surface only the most significant regions within the dipoles. The merging criteria chosen above makes the dipole discovery less sensitive to the choice of K (and robust for moderate values of K). The robustness can be seen in Figure 3 which shows that with increasing the value of K only increases the size of dipole regions and yet it does not surface any spurious region claiming it to be dipole. Steps 1, 2 and 3 of the algorithm are illustrated by the example presented in figure 4.

3.4. **STEP 4: Finding dipoles using density based clustering on G^{SRNN}.** Figure 3 shows the density plot of locations, where density for a location is defined as the weighted degree (sum of edge weights) of that location. From the visual inspection of figure 3, it is clear that the spatially contiguous red regions form a single dipole region. These regions can be extracted using a spatial clustering algorithm over the latitude, longitude and the intensity of the locations. We propose a method which is motivated from the Denclue algorithm [9], which finds clusters in data based upon local density attractors. Specifically, we use latitude and longitude to determine the local attractor (point with the highest density) in the neighborhood of locations. The algorithm proceeds by attaching every node in the graph to its local attractor by moving in the direction of increase in density. In the next step, we hierarchically merge attractors that are very close and have a positive

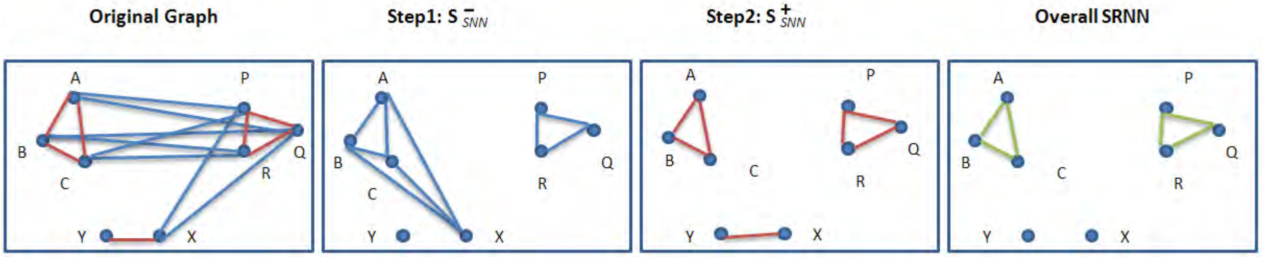


FIGURE 4. Illustration of Steps 1,2 and 3: Blue edges are reciprocal negative and red edges are reciprocal positive. First box on the left shows the original reciprocal graph. Second box shows G^{SNN-}. Note that edges A, B and C get connected as they share negative neighbors P, Q & R. Also the node X gets connected to A, B and C since X shares P and Q with them. Third box has G^{SNN+} and all the nearby nodes get connected. Fourth box shows G^{SRNN} with overall similarity defined as the product of the two. This helps in separating node X from nodes A, B & C.

correlation in order to remove extraneous attractors. The details of the algorithm are presented in Algorithm 1.

The locations that remain in A form the cluster centers and they become the attractor of all the points in their neighborhood (as assigned in LA in algorithm 1). The points that are attracted to a given cluster center are part of the same cluster. Next we compute the correlation of every cluster pair to find the dipoles from the clusters. After this we label all the cluster pairs having a sufficient negative correlation as a dipole, where by sufficient we mean a user provided correlation threshold. However, the threshold does not matter as there are far fewer cluster pairs generated and we can label all the cluster pairs having a correlation < 0 as dipoles. The significance of these cluster pairs can later on be ranked on the basis of their strength or impact on land temperature/pressure anomalies as we see later in the Section. 4.1.

Algorithm 1: Local attractor based clustering.

```
Let, $DS_{i,j}$ be geographical distance between locations $i$ and $j$.
Let, $CORR_{i,j}$ be anomaly correlation between locations $i$ and $j$.
Let, $D_i = \sum_{j=1}^{N} E_{i,j}^{SRNN}, \ \forall i \in \{1, 2, ..., N\}$ (location density).
Let, $A = \{1, 2, ..., N\}$ (local attractor set - initially set to all locations on Earth).
Let $LA_i = i$ (local attractor of all nodes are set to themselves initially).
repeat
  for $i \in A$ do
    $j = \arg\min_k (DS_{i,k} : k \in A \wedge k \neq i)$
    if $DS_{i,j} <$ Distance-Thresh AND $CORR_{i,j} >$ Correlation-Thresh then
      if $D_i \geq D_j$ then
        $A = A \setminus j$ {Eliminate $j$ from attractor set as $i$ is the attractor of $j$}
        $LA_z = i, \forall z \in \{1, 2, ..., N\} \wedge LA_z = j$
      else
        $A = A \setminus i$ {Eliminate $i$ from attractor set as $j$ is the attractor of $i$}
        $LA_z = j, \forall z \in \{1, 2, ..., N\} \wedge LA_z = i$
      end if
    end if
  end for
until convergence {If $A$ doesn't change in two successive iterations, then algorithm converges}
```

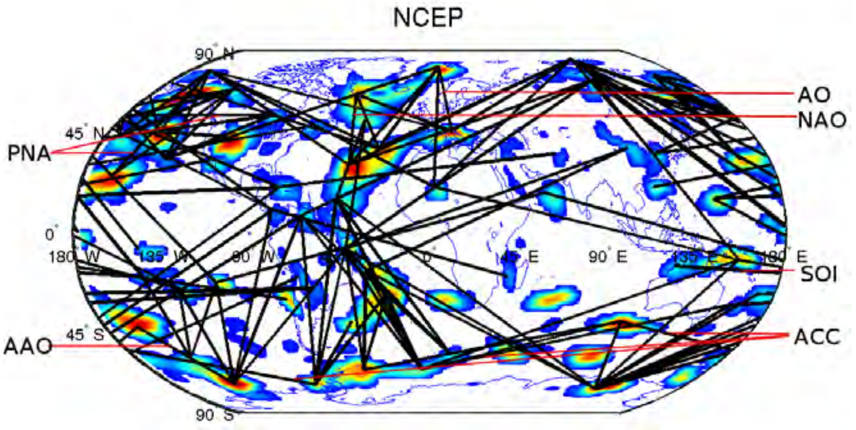


Figure 5. Dipoles in SLP NCEP data from 1948-1967. The color background shows the SRNN density identifying the regions of high activity. The edges represent the dipole connection between two regions.

3.5. **Algorithm Features.** The proposed algorithm runs in $O(N^2)$ space and time. Moreover, our approach can be implemented quite efficiently. *The previously known approach* [12] *takes more than 1 day to run and the proposed approach runs in less than 20 minutes for the NCEP/NCAR Reanalysis SLP dataset at a* $2.5°$ *resolution.* It further improves over the previous approaches by eliminating spurious dipoles and filtering of noise automatically. Additionally, it has only one parameter K (and not sensitive to its choice as well) in contrast to the previous algorithm, which had four parameters. Additional advantages of this approach over the previous ones are that it can find weak dipoles and it produces a much more reasonable number of candidate dipoles as shown further in Section. 4.1.

4. Experimental Evaluation

The goal of our experimental evaluation is three-fold. First, we want to show that the dipoles generated by our approach are similar in terms of their power as compared to the ones found in [12]. Next, we discuss the utility of a global snapshot view of the dipoles. Finally, we show how the technique can be used to study the behavior of various GCMs and understand their predictability for various climate change scenarios.

4.1. **Evaluation of Dipoles.** We construct networks from the NCEP/NCAR data using anomaly time series for a period of 20 years with a sliding window of 5 years so as to study the gradual change in the climate networks. Thus, for the 60 years of NCEP/NCAR data we had 9 networks spanning 20 years each. We ran the dipole detection algorithm for each of the 9 periods. Fig 5 shows the dipole interconnections in the first 20 year periods from 1948-1967.

In order to compute the "goodness" of the dipole clusters generated, we use three measures defined in [12]:

(1) *Dipole correlation with known climate indices*: Strong correlation indicates that the generated dipoles are good representatives of the known climate indices. The known climate indices are provided by the Climate Prediction Centre's (CPC) [1] website.
(2) *Dipole strength*: It is defined as the mean negative correlation of all the edges in the two ends of the dipole. A dipole is stronger if the correlation is more negative.
(3) *Dipoles' impact on land*: This allows us to finally test the utility of dipoles generated by our approach as compared to the static ones used by climate scientists. The impact is computed by taking the aggregate area weighted correlation of the climate index with the temperature anomalies.

Table 1. Correlation of our dynamic indices with known climate indices ($K = 25$)

Start year	**SRNN**				
	SOI	NAO	AO	WP	AAO[1]
1948	0.9035	0.7764	0.8121	0.7290	-
1953	0.7038	0.7689	0.8177	0.7287	-
1958	0.8998	0.7716	0.8065	0.7323	-
1963	0.8895	0.7246	0.7848	0.7341	-
1968	0.9279	0.7500	0.7859	0.7581	-
1973	0.9267	0.7590	0.8400	0.7319	-
1978	0.9452	0.7403	0.7654	0.7361	-
1983	0.9400	0.6625	0.8215	0.7274	0.9193
1988	0.9437	0.7185	0.8121	0.7042	0.9277

Table 2. Strength of the dipoles ($K = 25$)

Start year	**SRNN**				
	SOI	NAO	AO	WP	AAO[1]
1948	-0.2184	-0.4087	-0.4951	-0.3413	-
1953	-0.1663	-0.3804	-0.4395	-0.2814	-
1958	-0.2924	-0.4308	-0.4746	-0.3883	-
1963	-0.3275	-0.1731	-0.4189	-0.3974	-
1968	-0.3510	-0.1726	-0.4286	-0.3547	-
1973	-0.3890	-0.4458	-0.4576	-0.3293	-
1978	-0.3243	-0.3014	-0.5256	-0.3253	-
1983	-0.3582	-0.2173	-0.5667	-0.2557	-0.3578
1988	-0.2621	-0.3324	-0.5253	-0.3606	-0.3530

To test whether the right dipoles are being found using our methodology, we compute the correlation of the dipoles with the static indices known by climate scientists from the CPC website[1]. Table 1 shows the correlation between the static and dynamic climate indices using $K = 25$ nearest neighbors. These results are comparable to [12]. An important point to note is that even though the AO and AAO static indices are defined by climate scientists by taking a huge region(70 degree latitude each) doing a PCA kind of analysis, we are still able to find a region based definition for these dipoles with a correlation > 0.85. Table 2 shows the strength of different dipoles during different network periods. The AO dipole is the strongest dipole in all the network periods. The SOI dipole has a weaker strength than the NAO/AO dipoles. Note that the numbers in [12] were lower as smoothing was used in the data. The real utiliy of data driven dipoles lies in the fact that they are able to capture land temperature and precipitation anomalies related to these dipoles better than the static indices used by climate scientists. In order to show the utility of our dipoles we take an area weighted correlation of land temperature with the static and dynamic indices. Only locations having correlation > 0.2 are considered to compute the area weighted impact and the aggregate impact is divided by the total land area to generate a single number. We also varied the threshold by 0, 0.1, 0.3, 0.4, etc and saw a similar difference in between the static and dynamic indices. Figure 6 shows the aggregate area weighted correlation of land temperature anomalies using the static and dynamic NAO dipoles. The area weighted correlation of land temperature is much higher using a dynamic index as compared to the static index even while using different values of K. Fig 7 shows the correlation of land temperature anomalies using the static and dynamic NAO index. From the figure, we see that both static and dynamic NAO have a similar pattern but the dynamic index shows a much stronger correlation with land temperature anomalies. To validate that the land

[1] The AAO climate index data at the Climate Prediction Center is available only from 1979.

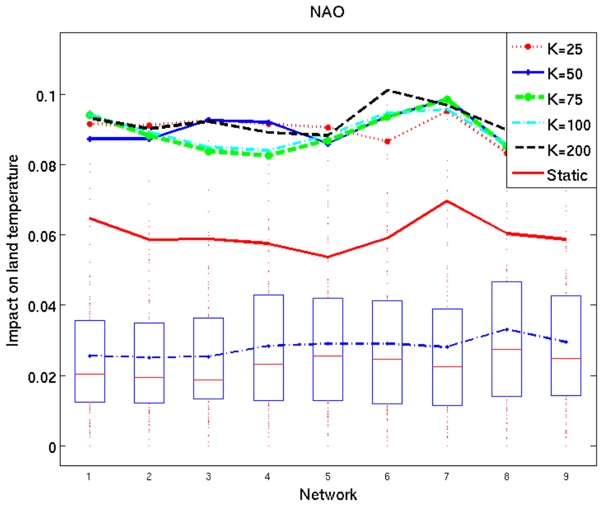


FIGURE 6. Area weighted impact on land temperature using static and dynamic NAO. The boxplot shows the spread of impact on land temperature using 100 random locations.

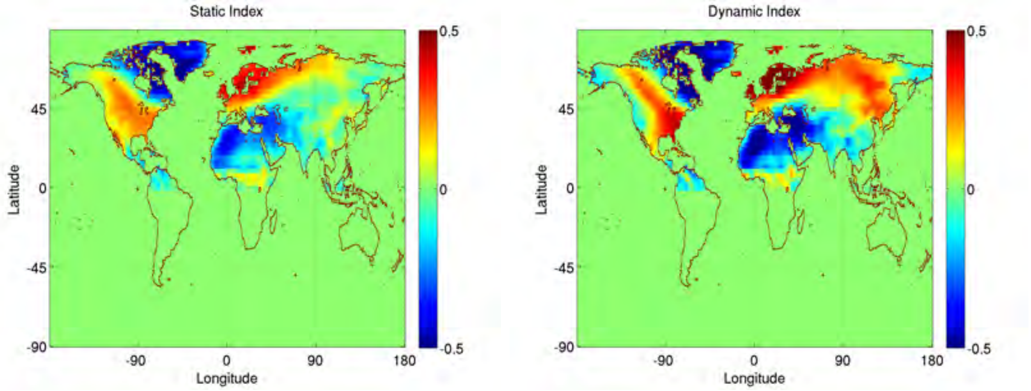


FIGURE 7. Correlation of land temperature anomalies using static and dynamic NAO.

impact generated by our identified dipoles is not spurious, we perform a randomization test. We randomly select 100 positively correlated time series from locations on Earth that are most likely not a part of any dipole. We compute their impact on land temperature anomalies. The boxplot in Figure 6 shows the spread of the impact using the 100 random locations and the blue line in the box shows the mean of the impact using these locations. Note that static and dynamic indices have a much stronger impact as compared to the random baseline. The dynamic index always generates a stronger impact than the static one for different numbers of nearest neighbors K. We also get similar results for the SOI dipole as reported in [12] and again the correlations are higher for dynamic indices than for static ones. We are also able to show a better impact on precipitation anomalies using CRU observational data[2] but do not report the numbers due to space constraints. The biggest advantage of our current approach as compared to [12] is that it allows us to have a comprehensive view of the dipoles and their interactions. Figure 5 illustrates the dipole connections in the first network, which represents the period 1948 to 1967. The figure is generated by connecting the local attractors of all

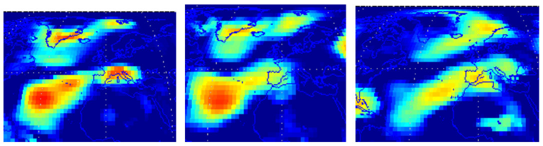


FIGURE 8. NAO/AO interactions in the three periods, 1948-1967, 1968-1987, and 1988-2007.

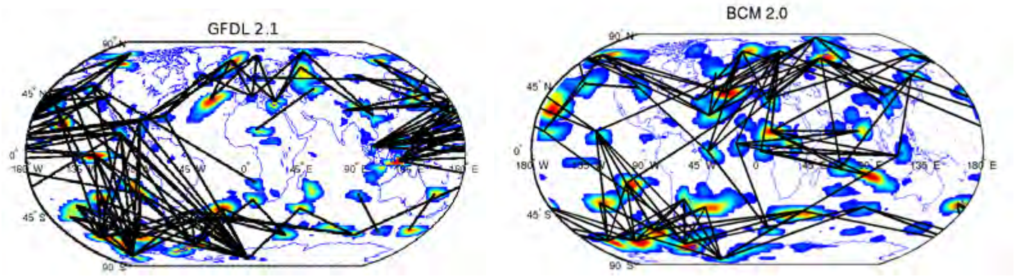


FIGURE 9. GFDL and BCM Hindcast

the cluster pairs labeled as dipoles. The figure shows that the NCEP/NCAR data reproduces the known climate patterns and indices during the first 20-year time range: the Northern Hemisphere pattern from west to east, the Pacific/North-America Pattern (PNA; which is actually a tripole) in the top left corner, the NAO and AO in the central top, and the West Pacific oscillation (WP) on the top right. In the Southern Hemisphere and equatorial region, there are SOI connecting the west Pacific warm pool and eastern Pacific with a line from the central right eastward to the right end of the plot and showing up again in the far left to connect to the eastern Pacific, the South Pacific Convergence Zone to the East of Australia crossing the map to the right and showing up on the left end in the southern Pacific, the South Atlantic Convergence Zone connecting South America and the south Atlantic, a dipole over Africa that relates local rainfall anomalies to ENSO [8], and the Indian Ocean Dipole (IOD) in the southern Indian Ocean. The four peaks over the Southern Ocean are due to high and low pressure systems related to the Antarctic Circumpolar Current.

Our approach's ability to detect and visualize all the dipoles on the globe as in Figure 5 empowers our understanding of climate data in many ways. For example, Figure 8 illustrates dynamic changes in the interactions of the NAO/AO dipoles in the NCEP data when compared for different time periods, 1948-1967, 1968-1987, and 1988-2007.

Moreover, using a technique like this one, we can explore the data from the various model simulations, and quantify the goodness of the models using the simulations as we see further in the next subsection 4.2.

4.2. **Understanding IPCC AR4 Models.** Our SRNN based dipole detection algorithm allows us the ability to compare the performance of the different models by looking at their dipole networks. We detected dipoles in the data from various IPCC climate models using both backward (hindcast) and forward model predictions (forecast or projections). The hindcast and projections data generally cover the period of 1850—2000 and 2000—2100, respectively. We used the hindcast 1948—2000 data to have an overlap with the NCEP data. For model projections, the data for various climate change scenarios is available. We used the IPCC scenario A1B that incorporates IPCC's moderate case assumption for increase in greenhouse gases and thus predicts the moderate amount of warming amongst all scenarios. For the hindcast data, we constructed seven networks for the first seven 20-year periods investigated in the NCEP/NCAR reanalysis. For the 100 years of projection data, we constructed five networks of 20 years without overlap.

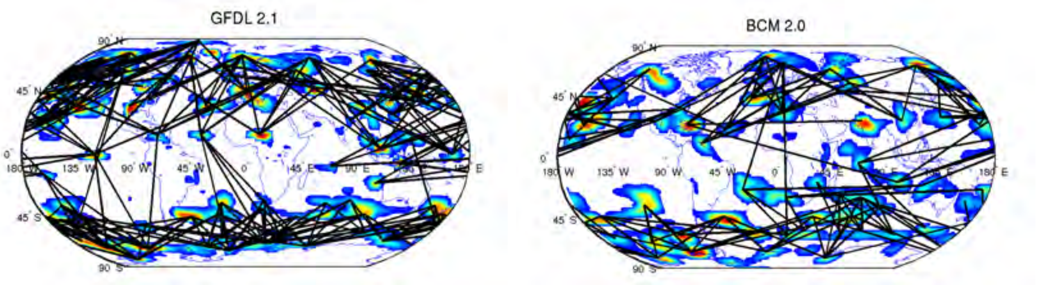


Figure 10. GFDL and BCM Forecast

Table 3. Strength of the NAO dipole in the 20 year networks in Hindcast data

Network	**Start year**	CCCMA	GISS	CSIRO	GFDL	BCM2.0	HadCM3
1	1948	-0.484	-0.4676	-0.4744	-0.4251	-0.55	-0.474
2	1953	-0.4962	-0.4667	-0.4605	-0.4274	-0.5719	-0.4817
3	1958	-0.5187	-0.4193	-0.4441	-0.4472	-0.5465	-0.5149
4	1963	-0.5304	-0.4244	-0.3962	-0.294	-0.5642	-0.4785
5	1968	-0.5191	-0.4263	-0.4056	-0.4263	-0.5296	-0.4558
6	1973	-0.4887	-0.4135	-0.2551	-0.4524	-0.4697	-0.4335
7	1978	-0.4456	-0.447	-0.2621	-0.5301	-0.5193	-0.4593

One way to quantify the output of the climate models is to look at the strength of the dipoles. We study the strength of the two major dipoles NAO and SOI in selected model simulations. From the several cluster pairs declared as dipoles our goal is to identify NAO and SOI. Hence, for every model simulation, we created a static index based on the grid points over Iceland and the Azores for NAO and over Tahiti and Darwin for SOI as per the way they are defined. After that, we picked up the dipole cluster pair that had the highest correlation with the static index for the two static indices and labeled them as NAO/SOI respectively. Tables 3 and 4 show the strength of the NAO dipole in the various models in the hindcast and forecast modes. The table shows that all models reproduce a NAO in hindcast mode, and the dipole strength in NAO forecast mode stays within the range of the hindcasts. Tables 5 and 6 show the strength of the SOI dipole in the hindcast and the forecast models respectively. The Table has only 3 columns as SOI as our algorithm detects SOI in only 3 of the 6 models. This result is consistent with the findings that models differ in their capability to represent different climate indices [14],[13]. The ability to construct detailed spatio-temporal characteristics of dipoles in simulation data can provide great insights on which models will perform better regionally and can be of huge benefit to the modeling community. In the following we discuss global dipole structure of two of the models as shown in Fig . 9.

Fig. 9 shows the dipole connections for hindcast period 1968-1987 for GFDL2.1 and BCM2.0, respectively using a threshold of -0.2 to show the edges. GFDL2.1 shows strong SOI connections from the west Pacific warm pool in the center right of the figure toward the right end of the figure and then showing up again at the far left to make a connection to the equatorial central and eastern Pacific. The AAO in the south Pacific and the Antarctic Circumpolar Current across the Southern Ocean are also well defined. In the north, PNA in the top left and NAO and AO in the top center can be detected. WP is seen in the top right over the west Pacific. The BCM2.0 simulation shows a strong pattern over Africa and over the Indian Ocean to the right (IOD) that is visible but weaker in the NCEP reanalysis (Fig. 5). The pattern in the north roughly resembles PNA, NAO, AO, and WP in GFDL 2.1, and the pattern in the south show a weaker Antarctic Circumpolar Current.

Shukla et. al[17] suggested that the SOI might not necessarily persist in a much warmer climate and instead change to a permanent state in which the SOI dipole is in a locked phase. This permanent

Table 4. Strength of the NAO dipole in Forecast data scenario A1B

Network	**Start year**	CCCMA	GISS	CSIRO	GFDL	BCM2.0	HadCM3
1	2000	-0.4525	-0.405	-0.3529	-0.2753	-0.5141	-0.3844
2	2020	-0.4603	-0.4152	-0.4253	-0.5108	-0.5444	-0.4557
3	2040	-0.5308	-0.4611	-0.348	-0.4883	-0.5614	-0.4874
4	2060	-0.4542	-0.461	-0.3913	-0.443	-0.4493	-0.538
5	2080	-0.4921	-0.4	-0.4195	-0.3488	-0.5332	-0.4047

Table 5. Strength of the SOI dipole in the 20 year networks in hindcast data

Net	Year	CSIRO	GFDL	HadCM3
1	1948	-0.5421	-0.443	-0.2651
2	1953	-0.5122	-0.4603	-0.2835
3	1958	-0.6208	-0.5091	-0.3055
4	1963	-0.568	-0.5132	-0.3172
5	1968	-0.3826	-0.5482	-0.4233
6	1973	-0.3296	-0.5021	-0.4671
7	1978	-0.3638	-0.5186	-0.3187

Table 6. Strength of the SOI dipole in projection data in scenario A1B

Net	Year	CSIRO	GFDL	HadCM3
1	2000	-0.3401	-0.5856	-0.302
2	2020	-0.3559	-0.5412	-0.3278
3	2040	-0.3181	-0.3303	-0.4237
4	2060	-0.3338	-0.4015	-0.3598
5	2080	-0.2856	-0.3563	-0.385

so called El Niño condition produces higher SLP in the eastern Pacific and lower in the western Pacific in a warmer climate based on regional terrestrial paleo-climatic data and general circulation model studies. Our results on the forecast mode show a trend toward reduced SOI strength for the climate models (Table. 6). Further, from the fig 10 we can see the reduced activity of dipoles in the tropics as represented by much less connections than in the results for the present time (Fig.9), e.g. dipoles over Africa and equatorial South America in the centers of Fig. 10 are reduced in the forecast scenario. On the other hand, dipole structures over the mid-latitudes and Arctic regions in the upper and lower parts of these figures are enhanced indicating the stronger dipole activity in these regions. For NAO, our results show that forecast models capture NAO characteristics for the next century. Knowledge like this can inform the climate modeling community about the shortcomings of the models.

5. Discussion and Conclusion

In this paper, we presented a novel systematic shared nearest neighbor based approach to find dipoles in global climate data. The approach is able to find dipoles as accurately as presented in [12] with far fewer parameters and candidate dipoles generated. Furthermore, we can study the interconnections between dipoles and show possible interactions between atmospheric oscillations. Knowledge of these interactions is particularly important for predicting climate extreme events. For example, while the cold winter over Europe in 2010 could be largely explained by NAO and other local indices [5], the cold winter over North America at the same time is largely due to a combination of NAO and ENSO [7], thus knowledge of patterns that span multiple dipoles can be useful. Using this approach we can study the changes in their dynamics and structure in a much more systematic manner.

Further, our approach gives us an alternative method to measure climate model performance. Since the dipoles or teleconnections define the heartbeat of a climate system, we can measure how well the dipoles are represented in the different model simulations. From our preliminary investigation, we see that different models vary in their ability to capture dipoles. Indeed, some models seem to miss some dipoles completely. Climate predictions so far mostly rely on taking averages of the models and since dipoles are prevalent and important in climate data as they are known to be linked to climate variability across the globe, this result is very important in assessing the goodness of a climate model and the value of making regional predictions from the model. Further, this can provide insights into the creation of ensembles of the various models for further climate predictions.

Acknowledgments

This work was supported by NSF grants IIS-0905581 and IIS-1029771. Access to computing facilities was provided by the University of Minnesota Supercomputing Institute.

References

[1] Climate prediction centre, http://www.cpc.ncep.noaa.gov/.

[2] Climatic research unit, http://www.cru.uea.ac.uk/.

[3] http://www.cgd.ucar.edu/cas/catalog/climind/.

[4] http://www.esrl.noaa.gov/psd/data/.

[5] J. Cattiaux, R. Vautard, C. Cassou, P. Yiou, V. Masson-Delmotte, and F. Codron. Winter 2010 in Europe: A cold extreme in a warming climate. *Geophysical Research Letters*, 37(20):L20704, 2010.

[6] J. Donges, Y. Zou, N. Marwan, and J. Kurths. Complex networks in climate dynamics. *The European Physical Journal-Special Topics*, 174(1):157–179, 2009.

[7] J. García-Serrano, B. Rodríguez-Fonseca, I. Bladé, P. Zurita-Gotor, and A. de la Cańara. Rotational atmospheric circulation during north atlantic-european winter: the influence of enso. *Climate Dynamics*, pages 1–17, 2010.

[8] L. Goddard and N. Graham. Importance of the Indian Ocean for simulating rainfall anomalies over eastern and southern Africa. *Journal of Geophysical Research*, 104(D16):19099, 1999.

[9] A. Hinneburg and H. Gabriel. Denclue 2.0: Fast clustering based on kernel density estimation. *Advances in Intelligent Data Analysis VII*, pages 70–80, 2007.

[10] R. Jarvis and E. Patrick. Clustering using a similarity measure based on shared near neighbors. *IEEE Transactions on Computers*, pages 1025–1034, 1973.

[11] E. Kalnay and et al. The ncep/ncar 40-year reanalysis project. *Bull. Amer. Meteor. Soc.*, 77:437–471, 1996.

[12] J. Kawale, M. Steinbach, and V. Kumar. Discovering dynamic dipoles in climate data. In *SIAM International Conference on Data mining, SDM*. SIAM, 2011.

[13] J. Leloup, M. Lengaigne, and J.-P. Boulanger. Twentieth century enso characteristics in the ipcc database. *Climate Dynamics*, 30:277–291, 2008.

[14] J. Lin. Interdecadal variability of ENSO in 21 IPCC AR4 coupled GCMs. *Geophys. Res. Lett*, 34:L12702, 2007.

[15] I. P. on Climate Change. *Fourth Assessment Report: Climate Change 2007: The AR4 Synthesis Report*. Geneva: IPCC, 2007.

[16] C. L. Pekeris. Atmospheric oscillations. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 158(895), 1937.

[17] S. Shukla, M. Chandler, D. Rind, L. Sohl, J. Jonas, and J. Lerner. Teleconnections in a warmer climate: the pliocene perspective. *Climate Dynamics*, pages 1–19, 2011.

[18] M. Steinbach, P. Tan, V. Kumar, S. Klooster, and C. Potter. Discovery of climate indices using clustering. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 446–455. ACM, 2003.

[19] M. Steinbach, P. Tan, V. Kumar, C. Potter, S. Klooster, and A. Torregrosa. Clustering earth science data: Goals, issues and results. In *Proc. of the Fourth KDD Workshop on Mining Scientific Datasets*. Citeseer, 2001.

[20] M. Steinbach, P. Tan, V. Kumar, C. Potter, S. Klooster, and A. Torregrosa. Data mining for the discovery of ocean climate indices. In *Proc of the Fifth Workshop on Scientific Data Mining*. Citeseer, 2002.

[21] K. Steinhaeuser, N. Chawla, and A. Ganguly. An exploration of climate data using complex networks. In *Proceedings of the Third International Workshop on Knowledge Discovery from Sensor Data*, pages 23–31. ACM, 2009.

[22] A. Tsonis, K. Swanson, and P. Roebber. What do networks have to do with climate? *Bulletin of the American Meteorological Society*, 87(5):585–595, 2006.

[23] H. van Loon and J. C. Rogers. The seesaw in winter temperatures between greenland and northern europe. part i: General description. *Monthly Weather Review*, 106(3):296–310, 1978.

[24] G. A. Vecchi and A. T. Wittenberg. El niño and our future climate: where do we stand? *Wiley Interdisciplinary Reviews: Climate Change*, 1(2):260–270, 2010.

[25] H. Von Storch and F. Zwiers. *Statistical analysis in climate research*. Cambridge Univ Pr, 2002.

[26] G. Walker. Correlation in seasonal variations of weather, viii. a preliminary study of world weather. *Memoirs of the India Meteorological Department*, 24(4):75–131, 1923.

[27] G. Walker. Correlation in seasonal variations of weather, ix. a preliminary study of world weather. *Memoirs of the India Meteorological Department*, 24:275–332, 1924.

INCORPORATING NATURAL VARIATION INTO TIME SERIES-BASED LAND COVER CHANGE IDENTIFICATION

VARUN MITHAL*, ASHISH GARG*, IVAN BRUGERE*, SHYAM BORIAH*, VIPIN KUMAR*,
MICHAEL STEINBACH*, CHRISTOPHER POTTER**, AND STEVEN KLOOSTER**

Abstract. The ability to monitor forest related change events like forest fires, deforestation for agriculture intensification, and logging is critical for effective forest management. Time series remote sensing data sets such as MODIS Enhanced Vegetation Index (EVI) can be used to identify these changes. Most existing approaches work on small data sets spanning over a specific geographic region of a homogeneous vegetation type. Also, most of these need training samples or require setting of parameters for each geographic region individually. These limitations make the algorithms unscalable and restrict their global applicability. In this paper, we present a scalable time series based change detection framework that overcomes these limitations of the existing methods. We introduce the concept of natural variation in EVI for a given of location and incorporate it into the change detection paradigm. We evaluate the change events identified by our approach using forest fire validation data in California and Canada. The results of this study demonstrate that the inclusion of a measure of natural variability improves detection accuracy, and makes the paradigm more robust across vegetation types and regions.

1. Introduction

Forests act as a sink of atmospheric carbon while disturbances such as fires and deforestation cause the stored carbon to be released into the atmosphere. In addition, forests are home to many ecosystems and these disturbances cause them severe damage. For efficient and effective management of forest resources, reliable and quantifiable observation of forest cover changes at a global scale is critical [18]. Some nations have allocated resources to monitor disturbances in their forests. As an example, Brazil has developed a system for deforestation monitoring called PRODES. Regional products such as these are infrequent because they require considerable resources. Therefore, there is a need to develop a forest monitoring system to identify global forest disturbances.

Data collected from remote sensing instruments can be used to identify changes in forests. The bulk of work in identifying land cover changes using remote sensing data involves image comparison methods [8, 16]. These methods include classifying locations using reflectance data and using post-classification comparison to identify changes. They require training data sets for supervised classification process. However, labeled data is available only for some regions in the world and classifiers built using training examples from one region perform poorly if applied to another region. Therefore, these techniques are region-specific and not globally applicable. Furthermore, the error in classification gets compounded during change detection. In addition, several characteristics like rate of change and the actual change date cannot be found using these image comparison based methods because these bi-temporal approaches compare snapshots between two dates and information between those dates is not considered. On the other hand, time series-based approaches look at a longer stretch of greater context and therefore can be utilized for providing fine grained information about land cover dynamics that is necessary to quantitatively assess the carbon impact of land cover changes [20]. Publicly available global time series data sets such as MODIS Enhanced Vegetation Index (EVI) can be used to identify changes in forest cover. Hence there is increasing interest in time series-based approaches to change detection in vegetation data [2, 4, 10, 12, 14, 15, 17, 21].

*University of Minnesota, <mithal,ashish,ivan,sboriah,kumar,steinbac>@cs.umn.edu
**NASA Ames Research Center, chris.potter@nasa.gov, sklooster@gaia.arc.nasa.gov.

However, most of these methods have been used on small data sets spanning over a specific geographical area and comprising of a homogeneous vegetation. There is often a parameter setting step that is fine-tuned for performance in that specific geographical region and vegetation. This is a serious limitation that makes these algorithms difficult to apply on a global scale.

There are primarily three types of time series-based land cover change detection approaches. Temporal segmentation methods divide the time series into homogeneuos regions and the boundary of the segments indicate change in vegetation [4, 12]. These approaches aim to identify any change in the vegetation type such as change from one land cover to other or changes in crops. Another approach is to look for trends in the time series spanning over multiple years to identify gradual decrease in vegetation [10, 23]. Such gradual changes represent forest degradation such as due to insect infestation, long-term droughts, etc. The third approach, which is the focus of the paper, is to identify an abrupt decrease in vegetation by predicting EVI values from a learned model for the vegetation time series and using prediction error to identify a change [3, 6, 10, 15, 17, 21]. Roy et al. [21] use a model-based prediction scheme for identifying fires from time series data and use this for generating the global Burned Area Product. Kucera et al. [15] use a CUSUM based technique and model fitting to identify forest fires in Portugal. Hammer et al. [10] use a regression-based technique to model the short and long term trends in NDVI for detecting deforestation in pan-tropical rain forests, while Chandola and Vatsavai [6] use Gaussian Process regression and PARASID (http://www.terra-i.org/) use a neural network for prediction. Lunetta et al. [17] use a spatial anomaly detection method for identifying deforestation. Boriah [3] describe Yearly Delta, a model based approach that uses mean annual EVI difference between successive years to identify changes such as forest fires and show that it is comparable in performance to Recursive Merging proposed in [4] which was shown in [5] to significantly outperform algorithm based on CUSUM and the scheme proposed by Lunetta et al. [17].

Keogh et al. [13] proposed an anomaly detection approach to find discords in longer time series. If an abrupt change occurs in a time series, the subsequence for that year will be unusual with respect to remaining time series and therefore discord discovery can potentially be adapted for finding abrupt changes. In our adaptation, annual discords are computed for each time series and the distance of the discord to its nearest neighbor is considered the change score. In case of an undisturbed forest, all annual segments are highly similar and therefore the discord score is low. For the time series in which a fire occurs, this scheme identifies the change window accurately and flags it as a discord giving it a relatively high discord score. However, performance is severely impacted in presence of time series with no fire-related change but large noise or high inter-annual variations.

A global scale analysis reveals that some vegetation types are highly stable and show a small decrease in EVI when a disturbance occurs, but this decrease can be significant compared to the otherwise stable nature of past EVI values. On the other hand, some vegetation types show random fluctuations in the EVI signal which occur due to atmospheric interference and various natural sources ranging from soil conditions to variation in inter-annual temperature and precipitation. Thus, significance of the Yearly Delta score differs based on the region and vegetation type, and score thresholds need to be adjusted separately for different geographic regions and land cover types to avoid too many false alarms. In this paper, we propose two *time series change detection algorithms* that utilize the temporal structure present in the remote sensing data to incorporate natural variability in the vegetation signal of a location in the Yearly Delta algorithm. The incorporation of the concept of natural variation in the Yearly Delta algorithm improves the change detection accuracy, and makes the paradigm more robust across vegetation types and regions. The evaluation of the proposed method is done quantitatively using validation data on forest fires in California and Yukon. We also compare performance of our algorithms against the output from Burned Area Product, a well-known global NASA product for fire monitoring by Roy et al. [21]. The evaluation results assert the importance of incorporating natural variability of vegetation in change detection. In particular, the experiments illustrate the need for variability modeling if change detection is performed on larger regions with multiple vegetation types.

1.1. **Key contributions.** The key contributions of this paper are: (1) a novel, scalable framework to identify significant abrupt changes in spatial-temporal remote sensing vegetation data sets to address the problem of land cover change detection, (2) introducing a concept of natural variation of EVI in the identification of changes in EVI signal, (3) a method to associate significance to observed annual changes in EVI with respect to the natural variability of the location, and (4) a quantitative evaluation of the performance of the proposed approach using validation data sets available for forest fires in California and Yukon and also comparison with an existing well-known global fire monitoring product.

1.2. **Organization of the paper.** We describe the data used in this study in Section 2. Section 3 presents the proposed change detection framework and the details of the proposed algorithms. Section 4 describes the validation data and evaluation methodology for this paper. Section 5 provides analysis of the results and Section 6 discusses the key challenges in the task of land cover monitoring, limitations of the proposed algorithms and the future research directions.

2. Data and Preprocessing

Global remote sensing data sets are available from a variety of instruments at different spatial resolutions as a sequence of snapshots of measurement values. In principle, the proposed algorithms can be applied to any geospatial dataset that features regular, repeated observations, consistent image registration and well-defined composite indicators of vegetation. In this study, we use the Enhanced Vegetation Index (EVI), a data product derived from measurements taken by the Moderate Resolution Imaging Spectroradiometer (MODIS) sensor on NASA's Terra satellite and distributed through the Land Processes Distributed Active Archive Center [1]. EVI essentially measures the "greenness" signal (area-averaged canopy photosynthetic capacity) as a proxy for the amount of vegetation at a particular location. MODIS data has been used to generate a continuous record of the EVI index at spatial resolution of 250 meters from February 2000 to the present. This index is generated at a temporal frequency of 16 days: each instance in the product is composited using the highest quality data from 16 daily raw observations.

In this study we use MODIS EVI data for California and Yukon. The data for California is at 250 m spatial resolution and 16 day temporal resolution. A spatial mask of MODIS landcover classes [9] was used to separate land cover categories of interest (forests, savannas and shrubs) from other categories like agriculture and urban using the MODIS MCD12Q1. The data set *DSCalifornia* has 3,389,564 pixels and predominant changes include forest fires, deforestation and urban expansion. The other dataset, *DSCanada* is MODIS EVI at 1km spatial resolution for Yukon Province in Canada. The data set has 551,275 pixels with the major change type as forest fires. This data set has significant homogeneity and no MODIS land cover mask was used. Winter months for this data set were pre-processed to an EVI value of 0 because winters are snow-clad at this latitude and have no vegetation response.

3. Algorithm Description

In this section, we describe the two proposed change detection algorithms to identify abrupt decrease in vegetation such as due to forest fires and deforestation. The main intuition behind these algorithms is that in stable forests, EVI values for future time steps tend to be similar to previous years when accounting for seasonal variation. On the other hand, changes like fires and deforestation are characterized by an abrupt decrease in EVI after the event. The algorithms build a model that is used for predicting the expected EVI values for the future years. Deviation of the future observations from the predicted value indicates a change. A measure that quantifies the deviation of future observations from the model prediction is used to assign the change score. One possibility is to predict the EVI for each time step, and use the prediction error as change score. However, this method is susceptible to noise in the data and causes too many false alarms. Figure 1 has some time steps when the observed EVI is lower than usual though there is no change in the

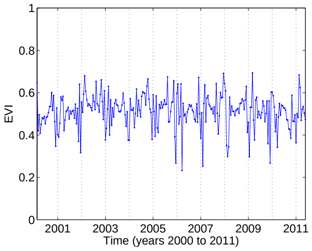


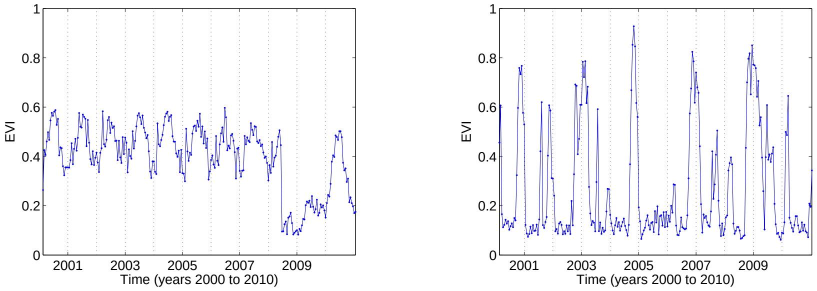
Figure 1. Noise in EVI data can cause false alarms.

location. A methat that only relies on one time step for change detection will falsely identify such locations as changed. One approach to address this issue of noise susceptibility is to use the notion of persistent decrease, and flag a time series for change only if a significant fraction of time steps from a time window of size m exceed the threshold. This idea of persistence is used by Roy et al. [21] in their algorithm to identify fires from daily remote sensing time series. Another approach is to predict a more stable statistic, for example, the mean of the successive time steps over an year is more robust to noise than deviation from prediction at a single time step. This approach is used in the Yearly Delta algorithm discussed in [3] and [18].

In the following, we describe the Yearly Delta algorithm and its two new variations: Variability-Aware Yearly Delta and Vegetation-Independent Yearly Delta. Our focus is to incorporate the natural variation of EVI time series for a location in assigning the change score. This concept of variability proposed in the paper is applicable to both approaches, Burned Area and Yearly Delta.

3.1. **Yearly Delta algorithm (YD).** The main intuition behind this algorithm originally presented in Boriah [3] and also used in Mithal et al. [18] is that for the time step corresponding to the date of abrupt change event, the difference between the annual mean EVI of the previous and following year will be high. YD algorithm considers the previous year as the model and assigns a change score to each time step as the difference between the mean annual EVI of the previous year and the following year. The YD score for a location is the maximum change score across all time steps and the time step with the maximum score is considered the time point for the change. The pixels are ranked based on their YD score and a certain number of the top ranked pixels are considered as changes. This algorithm works under the assumption that the pixels which have an abrupt change event will get a higher score than those which do not have an abrupt change event because the undisturbed pixel typically do not have an unusually large EVI decrease from one year to next. Figure 2(a) shows a location in California with a fire occurrence. We can see an abrupt decrease in EVI value after the fire in year 2008 that will lead to a high mean annual EVI difference and therefore a high YD score.

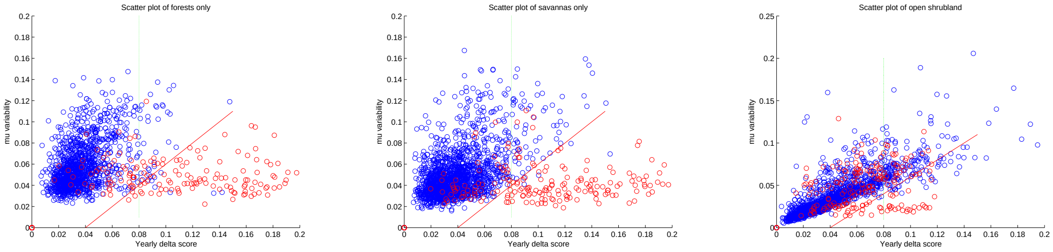
3.2. **Variability-Aware Yearly Delta (VD).** The observations in the future years vary from the model built based on previous years due to natural variability arising from changes in weather, soil conditions, etc. Shrub land cover shows a variety in the natural variation in their EVI signal and are very sensitive to changes in local climate conditions of precipitation and temperature in comparison to forests which are more resilient to such climatic changes. Thus, the mean annual EVI differences are not expected to be always 0 for the unchanged locations. Also, this variability depends on the location's local geography and land cover type. The change score should reflect the significance of the deviation with respect to the natural variation of vegetation response for that location. The YD algorithm does not make use of information about the natural variation in EVI. The same YD score can correspond to a significant loss of vegetation for some vegetation type or can occur due to natural variation in others. To understand this, consider the two time series in Figure 2(a) and 2(b) that have the same YD score. Figure 2(a) corresponds to an actual change in year 2008, while Figure 2(b) has the same YD score for year 2005 due to inter-annual natural variability that exists



(a) EVI time series for a pixel in forest location with YD score of 0.29 and VD score of 0.21 for year 2008.

(b) EVI time series for a pixel in shrub location with YD score of 0.3 and VD score of 0.07 for year 2005.

FIGURE 2. Illustrative examples to understand the power and limitations of the YD algorithm



(a) MODIS Forests

(b) MODIS Savannas

(c) MODIS Shrubs

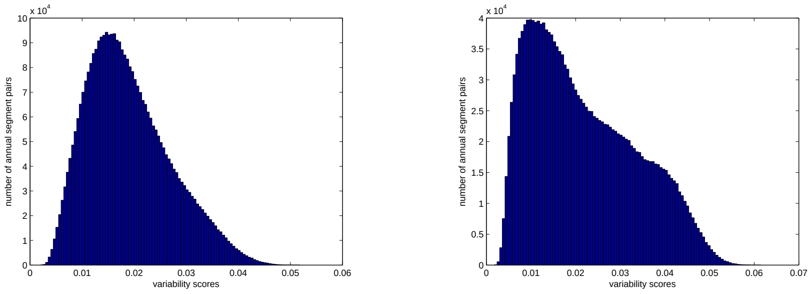
FIGURE 3. Scatter plot of mean variability (μ_{var}) against the YD score for different land cover categories in California.

in shrubs. The limitation of YD to distinguish between the two types of changes illustrated by Figure 2(a) and 2(b) motivates the need to incorporate the notion of natural variation in the change detection paradigm. One possibility is to use the differences in EVI values in the past to model the natural variation of a location. In this approach, each annual segment in the first k years is considered a model and the remaining $k-1$ segments are considered observed values and the mean Manhattan distance for each of the pairs is computed to give a distribution of variability scores for that location. A YD score that lies in this distribution is likely to occur even by random fluctuation. So we modify the score as YD score relative to the mean of this distribution (μ_{var}) for each location. The new score is called the VD score and is computed as VD score = YD score - μ_{var}

To illustrate the advantage of subtracting the μ_{var} from YD score, we show the scatter plots of YD score against μ_{var} for a random sample derived from three different land cover categories in California. Figure 3 shows the unchanged locations as blue circles and changed locations as red circles. The vertical line in green shows the constant YD score of 0.08 and the oblique line in red shows the constant VD score of 0.04. These scores were chosen for the two algorithms because they gave similar number of changed events. Circles lying to the right half of these lines will have change scores higher than the line boundary and will be detected as changes by the algorithms respectively. Thus, the blue circles in this right half are the false positives of the algorithms. We see in Figure 3(a) that both algorithms will correctly identify the fires on the forest land cover though after subtracting variability we will reduce some of the errors. Figure 3(b) shows the same plot for savannas. Here we notice that YD will make more errors as compared to VD and incorrectly label a few unchanged locations as changed. The scatter plot for Open Shrublands is shown in Figure 3(c) and we see that this is a difficult category and performance of both YD and VD is poor. However, the number of mistakes made by VD is significantly lower to those by YD. These scatter plots illustrate the utility of modeling variability in the change score especially in the highly variable land cover types. Similarly, in Figures 2(a) and 2(b), we see that the two locations get the same YD score, but VD is able to incorporate the inherent variability of the locations and gives very different change scores.

Our experiments show that any value for k between 3 and 5 works well. Since the first k years are used for modeling natural variability, change detection starts from $k+1$ year and changes in the first k years are not detected. Also note that this method assumes that the first few years that are used for variability modeling are undisturbed in the location. If a change event occurs during these initial years, it will cause the location to get a high variability score and a later change at that location will go undetected. This limitation can be addressed by using a sliding window of the previous k years instead of the first k years for computing variability under the assumption that abrupt changes such as fires and deforestation do not happen multiple times in k years.

The VD algorithm highlights the importance of incorporating the natural variation of vegetation at a location in computation of the change score. In the discussion above, we see that μ_{var} is a good indicator of expected natural variations in mean annual EVI differences and using the VD score can significantly improve the performance, especially in some land cover types. Boriah et al. [5] illustrated a similar advantage of modeling variability in the context of their segmentation algorithm Recursive Merging.



(a) The distribution of mean Manhattan distances between annual segments for a cluster of locations with smaller spread.

(b) The distribution of mean Manhattan distances between annual segments for a cluster of locations with a wider spread.

FIGURE 4. The distribution of variability scores (mean Manhattan distances between annual segments) of groups of pixels from two different locations in California with same μ_{var} (i.e. around 0.02).

3.3. **Vegetation-Independent Yearly Delta (VID).** Consider the scenario where the distribution of the mean Manhattan distances between annual segments for two locations in different types of vegetation have the same mean (μ_{var}) but different spread. In this scenario, if the same decrease in mean annual EVI was noticed in the two locations, the VD algorithm will give them the same score. However, the probability that the decrease in mean annual EVI was observed by a random chance is different for the two locations. For the location with smaller spread of distribution (i.e. smaller standard deviation) the probability that this mean annual difference is by random chance is lower, and for the location with a wider spread (i.e. higher standard deviation) this probability is higher. VD, which will assign same score, is unable to distinguish between the two cases. This is a serious limitation for VD if it is used for a composite data set with multiple types of vegetation. In such a scenario, locations will have different spread of their variability score distribution and a higher VD score threshold will miss the actual change that occurred in the more stable vegetation and therefore have a poor recall for those vegetation types. On the other hand, a lower VD score threshold will

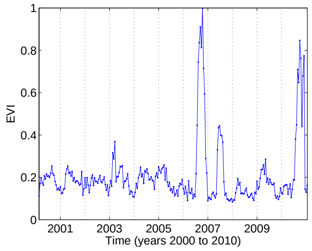


FIGURE 5. EVI time series for a location in California with highly stable initial years (before year 2005) for which the variability modeling was done and larger variations due to climatic variability in later years. Such locations are incorrectly identified as changed by the VID algorithm.

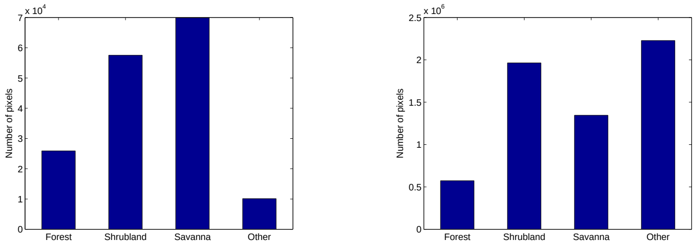
have many false positives from locations that have a wider spread of variability distribution because unchanged pixels will also have same VD score by random chance.

As an illustrative example, two locations in *DSCalifornia* were chosen and the 30 nearest neighbors for the two locations were computed based on Manhattan distance measure between their EVI time series. The mean Manhattan distance between annual segments for each pixel in the two groups were computed and Figure 4 shows the distribution of the mean Manhattan distances between annual segments of the two groups. These groups have similar mean variability (μ_{var}) but different spread in the distribution of the variability scores. The same mean annual EVI decrease observed has a different probability of occurring by natural variation in the two vegetation types. For example, if the YD score was 0.04 then the probability that this would be observed by natural variation in Figure 4(a) is considerably small, but Figure 4(b) has a high probability of getting this score by natural variability. Thus, there is a need to further scale the VD score with the standard deviation of the variability score distribution to accurately estimate the significance of the change.

The VID algorithm tries to address this limitation by including the standard deviation of the variability in the change score. It assumes that the random fluctuations in mean annual EVI for a particular vegetation type are normally distributed for a location and estimates the mean μ_{var} and standard deviation σ_{var} of the variability score distribution as the maximum likelihood estimates for the distribution.
The new score is called the VID score and is computed as VID score = (YD score - μ_{var}) / σ_{var}.

This score can be viewed as the z-statistic from the standard normal distribution. A high VID score threshold implies a lower false positive rate and vice-versa. In addition, fixing the same VID score threshold for all locations will incur same false positive rate across vegetation type. This, however, depends on the assumption that for different vegetation types the variability scores have a near normal distribution and the future EVI values also follow the same distribution if there is no change event. We observe that the assumption is true in most cases and the false positive rate for the algorithm is independent of the vegetation type. Due to climatic and other factors, for some vegetation types this assumption is not true and the false positive rates are higher for these vegetation types for the same score threshold. As an example, see Figure 5 which is the EVI time series of a location in California in which the variability changes with time perhaps due to changes in precipitation.

Note that we add a small number (1% of EVI scale) to the estimate value of σ_{var}. In case the σ_{var} is close to 0 for a very stable location, this avoids a small change from getting an extremely high VID score. This is especially important for locations with highly stable EVI such as in arid and semi-arid areas, where slightly high vegetation response for a single year due to higher precepitation might lead to a false alarm due to a high VID score.



(a) Landcover distribution inside the fire polygons of California.

(b) Landcover distribution in California

Figure 6. The histograms show the number of pixels of each land cover type (using the MODIS land cover map) inside the fire polygons and in entire California.

4. Evaluation

We use the same evaluation strategy as described in Boriah et al. [5] to understand the relative performance of different change detection techniques. The following describes the validation data used in this study and provides a brief overview of the evaluation methodology.

4.1. **Validation Data.** Change detection studies are frequently plagued by the lack of good ground truth data [19] which forces the evaluation process to be more qualitative in nature. In this study, we have utilized high quality validation data for fires generated by an independent source, and are thus able to perform an objective quantitative evaluation. Specifically, we obtained fire boundaries generated by the state of California for the fire seasons for the years 2006 through 2008 and the province of Yukon in Canada for years 2004 to 2008. The validation data is in the form of *polygons* which represent the boundaries of forest fires. Our EVI data is georeferenced by the latitude and longitude value for the pixel center. Thus, a pixel is considered inside a polygon if the pixel center lies inside it, otherwise it is considered outside the polygon.

The histogram in Figure 6(a) shows the distribution of land cover type of the pixels that lie *inside* the validation polygons for California; i.e., these are the pixels which actually burned according to the validation data. The figure shows that shrubland and savannas account for a significant portion of the burned regions in California. The land cover types included in our study are MODIS forests, savannas and shrubs and we exclude the pixels belonging to the "other" MODIS landcover category in our California data set. This is because land cover categories such as agriculture that belong to the "other" category have a vast number of changes and majority of these changes are not related to fires and therefore will be considered as false positives by our validation data. Also, the fraction of pixels belonging to the "other" category in the validation data is small compared to their fraction in the entire California data (as seen in Figure 6(b) that shows the distribution of land cover categories in California). In *DSCanada* we include the entire state of Yukon without any MODIS land cover mask as in this region almost all locations are forest, savanna or shrub. Also note that there are non-fire related changes in the forest, savanna and shrub MODIS categories such as logging and conversion to agriculture which are not covered in the fire polygons. Such changes will be incorrectly considered as false positives. However, we expect that this issue will impact performance of all algorithms similarly and will not change their relative performance.

4.2. **Evaluation Methodology.** The change detection algorithms assign a change score to each location, and the locations are ranked according to the descending order of their change score. The algorithm flags the top n ranked locations as change events and the lower ranked locations as unchanged. By computing the intersection with the validation data, we find the number of true positives (TP_n), false positives (FP_n), true negatives (TN_n) and false negatives (FN_n) as

		Predicted	
		Fire	No Fire
Validation Data	Fire	TP_n	FN_n
	No Fire	FP_n	TN_n

Table 1. Confusion matrix.

shown by Table 1. Our evaluation of the performance of the change detection algorithms is based on computation of the *precision* and *recall* that are two well-known metrics used to evaluate the performance of algorithms in information retrieval, machine learning and data mining [22] and given by:

$$\text{Precision, } p_n = \frac{TP_n}{TP_n + FP_n}$$

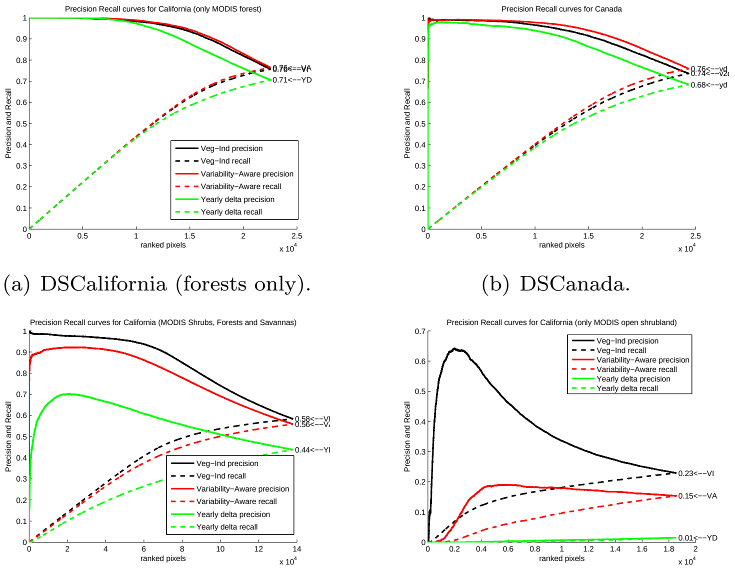
$$\text{Recall, } r_n = \frac{TP_n}{M}$$

To compare the relative performance of different techniques, we plot the precision and recall curve for the ranked list of pixels for the values $1 \leq n \leq M$. An ideal change identification algorithm should have a precision of 1 and a steadily rising recall from 0 to 1 as n increases from 1 to M. M in our case is the actual number of pixels inside fire polygons.

5. Discussion of Experimental Results

The performance of the three algorithms described in Section 3 is evaluated and analyzed in this section. The precision and recall plots for the data set on Yukon and California are used to understand the accuracy differences between the three algorithms on different vegetations. *DSCalifornia* data set comprises of many different land cover types and to illustrate the effect of incorporating variability in the change detection algorithms. This data was subsetted based on MODIS landcover map and results are reported for only forests and shrubs in addition to the entire data for California. A comparison with Burned Area Product is also reported to further highlight the ability of the algorithms to identify forest fires from EVI data.

5.1. **Performance of proposed approaches.** Figure 7(a) and Figure 7(b) show the precision and recall curve for the three algorithms in California (only forests) and Yukon respectively. The performance of the three algorithms is comparable in the data sets that comprise of only MODIS forest land cover category in California and in entire Yukon province. All three algorithms perform well and are able to identify fires in these areas with high recall and precision, but the two algorithms with variability incorporated show slightly better results. The reason behind the good performance of all algorithms on the *DSCalifornia* data set is that forests have lower variability and typically have a stable EVI which has an abrupt decrease primarily in case of an actual land cover change. The decrease in precision occurs primarily because the algorithms identify other changes like logging that also show an abrupt decrease in EVI but as the validation data is limited to forest fires these changes are considered as false positives. This limitation of the validation data is further discussed in Section 6. The slight improvement in performance by modeling natural variability comes because of presence of some non-forested locations in the data set. We use the MODIS forest map in California for this data set but it is inaccurate and includes some shrubs and agriculture lands labeled as forests. The VD and the VID algorithms are more resilient to such misclassifications in land cover map. This is because farms are by nature highly variable due to shifts in cropping dates and other reasons, and as such get a high variability score and are therefore eliminated by algorithms that incorporate variability modeling, but are detected as changes by YD because it gives these locations a higher



(a) DSCalifornia (forests only).

(b) DSCanada.

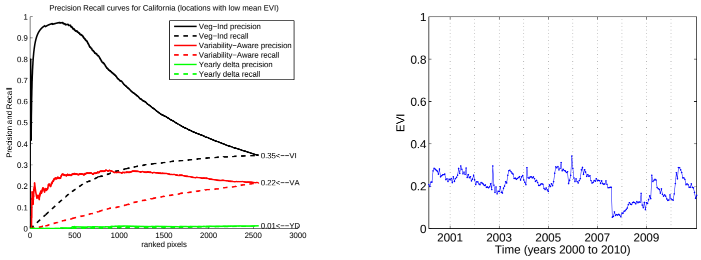
(c) DSCalifornia.

(d) DSCalifornia (open shrubs only).

Figure 7. Precision and Recall curves for three algorithms. *black* for VID, *red* for VD and *green* for YD

change score. The locations in *DSCanada* data set have primarily MODIS forest, savanna and shrub category. The EVI signal for all locations in this data set has considerable homogenity and therefore VD and VID are only slightly better compared to YD. We see that VID performs slightly worse than VD on the *DSCanada*. This is because the EVI time series in Canada have some observations with an unusually high EVI value. If this noise occurs in the first few years that are used to model the variability, the variability of that location will be high and as a result a change in the later years at that location will not get detected due to the high variability score. These outliers negatively impact results by incorrectly increasing variability for such locations and since VID uses variability modeling more strictly as compared to VD, it gets negatively impacted due to these outliers to a greater extent and shows a slightly worse performance than VD as seen in Figure 7(b).

The contrast between the performance of the algorithms becomes evident when we evaluate on the entire *DSCalifornia* data set. This is because this data set has multiple land cover types (including shrubs) and variability modeling becomes essential in the case of some of the land cover categories. Figure 7(c) shows the quantitative performance for the three algorithms on this data set. We notice that the YD algorithm performs significantly poorer than its counterparts that incorporate variability modeling. This is not surprising as shrubs form the dominant land cover type in the data set and they have high variability due to their higher sensitivity to climate variation. YD gives a high change score to many locations even if they are not burned and thus leads to a poor precision on the composite data set (*DSCalifornia*). This fact is further illustrated in Figure 7(d), which shows the performance of the three algorithms on *DSCalifornia* with open shrubs only. The precision of YD on this data set is exceptionally poor and indicates that YD is not a good change detection algorithm for this land cover type though its performance is comparable to other algorithms in forests. Since the number of shrubs in the composite data set *DSCalifornia* is high, the poor precision of YD on this data set is explainable. The high variability of shrubs is also present in the first few years used to compute μ_{var} and thus the VD is able to perform better and shows a considerably improved performance over YD on *DSCalifornia*. Again, this fact is supported by the observation in Figure 7(d) where precision of VD is significantly better than that for YD. The VID algorithm that also models the



(a) Precision and Recall curves for three algorithms on low EVI (less than 0.13). *black* for VID, *red* for VD and *green* for YD.

(b) A location in California with low and stable EVI and a fire with small decrease in EVI.

FIGURE 8. VID performs better on Low EVI locations and correctly detects changes in such locations.

spread of the variability distribution along with the μ_{var}, is able to work well across different land cover types. This is primarily because *DSCalifornia* (open shrubs only) has EVI time series with large variations in the spread of their variability scores (see Figure 4(b)). The VID score takes into account the information about the distribution's spread and therefore avoids several false alarms from this vegetation, while other algorithms make many mistakes on this particular vegetation. Low recall on the *DSCaliforina* (only open shrublands) data set is primarily due the fact that for many pixels inside fire polygons the EVI shows no significant change. Other bands of the spectrum have to be analyzed to be able to identify these burnt locations. An example EVI time series of such a location that was inside a fire polygon but has little change in EVI is shown in Figure 9. The vertical red line marks the date of fire. We see that the decrease in EVI was too small to be identified by these algorithms.

Another observation is that the VID algorithm has much better performance in locations with low mean EVI (see Figure 8(a)). These vegetation types are typically stable and even after a change event, the decrease in EVI response in small in magnitude. Since VID models the standard deviation in the variability, these vegetation types have low σ_{var} and therefore even smaller changes get identified. This is the case with fires in the open shrub land cover for California which typically occur in locations with extremely stable low mean EVI values (an example is shown in Figure 8(b)). We observe that the VID score is independent of the the magnitude of the original time series and has therefore has comparable performance across vegetation types with different mean EVI.

5.2. **Comparison with Burned Area Product.** We use the output of the Burned Area Product to evaluate its performance on *DSCalifornia* and *DSCanada*. A location is considered burned according to the Burned Area Product if it flags a burn in the years under consideration. There is also a resampling step required before using the Burned Area Product which is available at 500 m spatial resolution. For *DSCalifornia* which is at 250 m resolution, all 250 m pixels that lie within a 500 m pixel get the same label. For *DSCanada* which is at 1 km spatial resolution, a 1 km pixel is labeled burned if any of the 500 m pixels that lie within the 1 km pixel are considered burned. The differences in the performance of this product and our algorithms occur due to two reasons: (1) the change detection mechanism used in this product is different from our approaches and (2) the underlying data set used for the generation of this product is thermal band instead of EVI. The Burned Area Product indicated 34,986 locations to have burned in 2006-2008 in *DSCalifornia*. Out of these 24,890 are inside the polygons. The precision and recall are 71.1% and 18.1% for Burned Area Product on *DSCalifornia*. In Canada, the performance of Burned Area Product is better than that in California. The Burned Area Product reports 15,005 pixels burned in 2004-2008 out of which 13,513 are in polygons. The precision and recall are 90% and 55.5% respectively on *DSCanada*. For

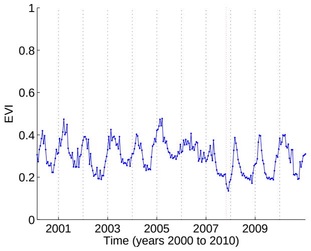


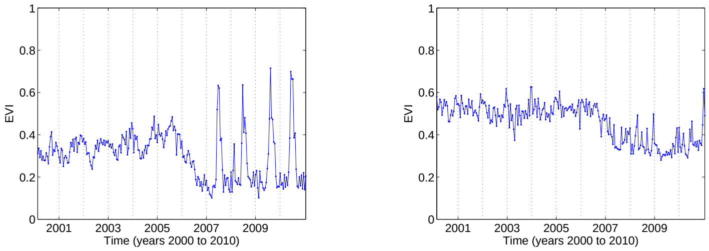
Figure 9. A burned location in California correctly identified by Burned Area Product and went undetected by our approaches because it shows little change in EVI signal. The vertical red line marks the change date.

the same precision as Burned Area, the YD algorithm has a similar recall on both *DSCalifornia* and *DSCanada*. This indicates that YD has a comparable performance to the technique by Roy et al. [21] that is used to generate the Burned Area Product. For a similar precision on the two data sets as the Burned Area, recall for the VID algorithm is around 50% on the *DSCalifornia* and 60% on *DSCanada*. The results on *DSCalifornia* are significantly better for the VID algorithm over the YD and Burned Area. This is because this data set has multiple vegetation types present in it and VID that incorporates the natural variation of EVI time series in the change detection paradigm performs better. The same idea of incorporating variability in the change detection framework can potentially be used with the Roy et al. [21] approach and improve their change detection accuracy. Furthermore, we notice some complementarity between the change events detected by the two approaches. This is primarily due to the different data set (thermal band) used by Burned Area Product. Several changes like Figure 9 which are not prominent in EVI signal are detected in the Burned Area Product.

6. Concluding Remarks

In this paper, we described two novel time series change detection algorithms that can be used to identify abrupt vegetation loss and extend the Yearly Delta algorithm by introducing the concept of natural variation in EVI time series of a location. The results of the study demonstrate the importance of modeling natural variation in the vegetation signal for each location for accurately estimating the significance of the change in EVI signal. The evaluation of the proposed method is done quantitatively using the validation data on forest fires from California and Yukon. The evaluation results demonstrate the ability of our proposed approach in identifying occurrence of forest fires from a remote sensing vegetation dataset (Enhanced Vegetation Index) with high accuracy. These algorithms are computationally fast (3000 timeseries are processed per second using a MATLAB implementation on a desktop), making it possible to process the entire globe at 1 km spatial resolution in less than a day on a standard desktop computer. Since computation for each time series is independent, the algorithms are easily parallelized. In the following, we discuss the limitations of the current work and possible directions for future research.

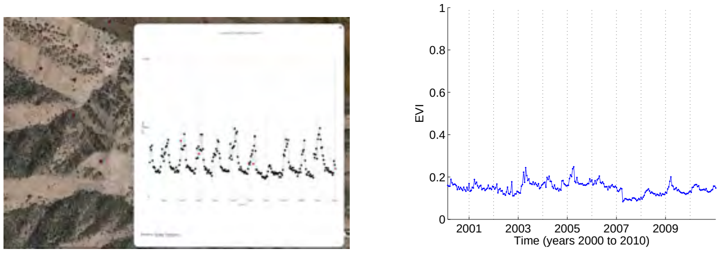
- **Limitation of evaluation methodology:** A careful look at the false positives of the Vegetation-Independent Yearly Delta algorithm reveals that it finds many changes that do not correspond to fires. This is because an abrupt decrease in EVI is also caused by other forest disturbances such as logging, floods, conversion to agriculture, etc (Figure 10(a)). In addition, it often finds gradual decreases in EVI that might occur due to slow forest degradation such as in Figure 10(b). These locations, though genuine forest cover disturbances, are considered false positives because they are absent in the validation data which is restricted to forest fires. Lack of exhaustive validation data for land cover change is a serious challenge with evaluation of forest monitoring algorithms. A fair evaluation is possible if an



(a) Change of forest to other vegetation type.

(b) Gradual decrease in time series.

FIGURE 10. EVI time series for a pixel not present in fire polygons while the time series indicate change.



(a) EVI time series of a shrub location, not present in fire polygons, with an unusually low vegetation for a single year.

(b) EVI time series for a pixel that was inside a fire polygon and not detected by the VID algorithm.

FIGURE 11. Example time series to illustrate the need for including climate variables like precipitation in change detection framework.

additional characterization step to identify the type of the change is included. This is particularly challenging because several types of changes often have similar EVI signature. For example, fires and deforestation often show the same characteristic abrupt decrease in EVI. Exploiting complex spatial and temporal structures present in the EVI data in conjunction with other data sets (eg. thermal band) could help distinguish between such changes.

- **Modeling of variations due to climate variables:** Figure 11(a) shows an example of a location that shows unusually low values for the vegetation index in one of the years. This signature was present in many pixels in shrublands in California and appears to be the result of a drought-like condition. These pixels are flagged as changes by our algorithms since they correspond to sudden drop in vegetation, but they are considered as false positives in the context of detecting fires or deforestation events. Figure 11(b) is an example of a pixel that was not identified as a change, though it was in the validation data. This pixel had a high variability score perhaps because the vegetation is highly sensitive to precipitation etc., hence the relatively smaller decrease was missed. One approach to correctly handle such cases, is to model variations due to changes in climate and incorporate it in the change detection paradigm. Hammer et al. [10] use rainfall in the month as a dependent variable in the regression equation for monthly NDVI to account for changes in rainfall. The extra-seasonal relationship between rainfall and NDVI is captured by this term. In addition, a long term trend in amount of rainfall leading to a trend in NDVI can be captured.

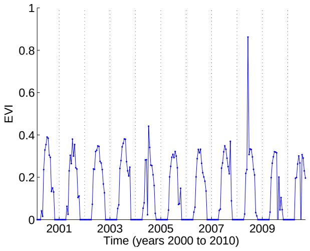
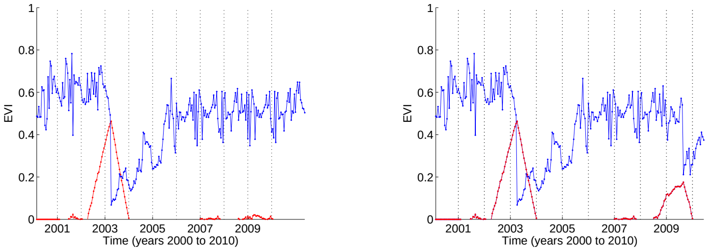


FIGURE 12. EVI time series for a pixel not present in fire polygons but was detected as change due to a noisy value in year 2008.

- **Noise Reduction:** Noise in the EVI time series poses a significant challenge to any change detection algorithm. For example, all algorithms in this study falsely identify changes in cases such as in Figure 12 where there is a noisy observation in year 2008 that increases the mean annual EVI for that year. There is a need to design a noise reduction technique that is cognizant of characteristics of remote sensing data sets and that utilizes the information about quality of observations, cloud and aerosol conditions that are available with the data. The remote sensing community has developed many noise reduction techniques to reduce the impact of noise in these data sets [11]. However, since these techniques were not designed to account for a possibility of an abrupt change, these smoothing-based techniques tend to distort the actual change point [7]. It is therefore not possible to directly apply an off-the shelf noise reduction technique, and new noise reduction techniques need to be developed that are suitable in the context of change detection.



(a) EVI time series (in *blue*) for a location with a single fire and the corresponding YD score time series (in *red*).

(b) EVI time series (in *blue*) for a location with two fires and the corresponding YD score time series (in *red*).

FIGURE 13. EVI time series and corresponding YD time series for single and multiple changes

- **Identifying multiple changes in time series:** Another limitation of the proposed approach is that it can find only a single change in a time series. The ability to find multiple changes will become critical as these time series are increasing in time dimension with more satellite data being collected. The change detection framework needs to be extended to allow for finding multiple changes in the time series. Thus instead of assigning each location a single change score, time steps of a location should get flagged as changes if they correspond to local maximas that are higher than the score threshold. As an example, Figure 13(a) shows the EVI and YD score time series for a single change. The peak in the YD score time series corresponds to the change point. Figure 13(b) shows the EVI time series for a location with two fires. The YD score time series shows two peaks that are separated in time and peak at the time of fire. Both these peaks should be flagged as changes and the location will have two change events.

- **Choice of Model:** The change detection algorithms described in this paper use the previous year as a model for EVI values of the current year. A more robust model can be built using the median of the EVI values in the previous k years. This is especially useful in eliminating false alarms due to noise in data or climate variations such as seen in Figure 5. In this figure the vegetation response is high for two years and the YD score will be high for the next year that has a low response. But if the median score for the previous 5 years was used the score will be low and not get falsely detected as change.

7. Acknowledgements

The research described here is supported by the Planetary Skin Institute, NSF Grant IIS-0713227, NSF Grant IIS-0905581, NASA grant NNX09AL60G and the University of Minnesota MN Futures Program. Access to computing facilities was provided by the University of Minnesota Supercomputing Institute.

References

[1] Land Processes Distributed Active Archive Center. http://lpdaac.usgs.gov.

[2] L. O. Anderson, Y. Malhi, L. E. O. C. Aragão, R. Ladle, E. Arai, N. Barbier, and O. Phillips. Remote sensing detection of droughts in Amazonian forest canopies. *New Phytologist*, 187(3):733–750, 2010.

[3] S. Boriah. *Time Series Change Detection: Algorithms for Land Cover Change.* PhD thesis, University of Minnesota, 2010.

[4] S. Boriah, V. Kumar, M. Steinbach, C. Potter, and S. Klooster. Land cover change detection: A case study. In *KDD '08: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 857–865, 2008.

[5] S. Boriah, V. Mithal, A. Garg, V. Kumar, M. Steinbach, C. Potter, and S. Klooster. A comparative study of algorithms for land cover change. In *Conference on Intelligent Data Understanding*, 2010.

[6] V. Chandola and R. Vatsavai. Scalable time series change detection for biomass monitoring using gaussian process. In *In Proceedings of NASA Conference on Intelligent Data Understanding*, 2010.

[7] X. Chen, V. Mithal, S. R. Vangala, I. Brugere, S. Boriah, and V. Kumar. A study of time series noise reduction techniques in the context of land cover change detection. Technical Report 11-016, Computer Science Department, University of Minnesota, 2011.

[8] P. Coppin, I. Jonckheere, K. Nackaerts, B. Muys, and E. Lambin. Digital change detection methods in ecosystem monitoring: a review. *International Journal of Remote Sensing*, 25(9):1565–1596, 2004.

[9] M. A. Friedl, D. Sulla-Menashe, B. Tan, A. Schneider, N. Ramankutty, A. Sibley, and X. Huang. MODIS Collection 5 global land cover: Algorithm refinements and characterization of new datasets. *Remote Sensing of Environment*, 114 (1):168–182, 2010.

[10] D. Hammer, R. Kraft, and D. Wheeler. FORMA: Forest monitoring for action—rapid identification of pan-tropical deforestation using moderate-resolution remotely sensed data. Working Paper 192, Center for Global Development, 2009.

[11] J. Hird and G. McDermid. Noise reduction of ndvi time series: An empirical comparison of selected techniques. *Remote Sensing of Environment*, 113(1):248 – 258, 2009.

[12] R. Kennedy, Z. Yang, and W. Cohen. Detecting trends in forest disturbance and recovery using yearly landsat time series. *Remote Sensing of Environment*, 2010.

[13] E. Keogh, J. Lin, and A. Fu. Hot sax: efficiently finding the most unusual time series subsequence. In *Fifth IEEE International Conference on Data Mining*, pages 226–233, 2005.

[14] A. Koltunov, S. L. Ustin, G. P. Asner, and I. Fung. Selective logging changes forest phenology in the Brazilian Amazon: Evidence from MODIS image time series analysis. *Remote Sensing of Environment*, 113(11):2431–2440, 2009.

[15] J. Kucera, P. Barbosa, and P. Strobl. Cumulative sum charts-a novel technique for processing daily time series of modis data for burnt area mapping in portugal. In *MultiTemp 2007. IEEE International Workshop on the Analysis of Multi-temporal Remote Sensing Images, 2007.*

[16] D. Lu, P. Mausel, E. Brondízio, and E. Moran. Change detection techniques. *International Journal of Remote Sensing*, 25(12):2365–2401, 2003.

[17] R. S. Lunetta, J. F. Knight, J. Ediriwickrema, J. G. Lyon, and L. D. Worthy. Land-cover change detection using multi-temporal MODIS NDVI data. *Remote Sensing of Environment*, 105(2):142–154, 2006.

[18] V. Mithal, A. Garg, S. Boriah, M. Steinbach, V. Kumar, C. Potter, S. A. Klooster, and J. C. Castilla-Rubio. Monitoring global forest cover using data mining. *ACM TIST*, 2(4):36, 2011.

[19] R. Radke, S. Andra, O. Al-Kofahi, and B. Roysam. Image change detection algorithms: a systematic survey. *IEEE Transactions on Image Processing*, 14(3):294–307, 2005.

[20] N. Ramankutty, H. K. Gibbs, F. Achard, R. Defries, J. A. Foley, and R. A. Houghton. Challenges to estimating carbon emissions from tropical deforestation. *Global Change Biology*, 13:51–66, January 2007.

[21] D. P. Roy, P. E. Lewis, and C. O. Justice. Burned area mapping using multi-temporal moderate spatial resolution data–a bi-directional reflectance model-based expectation approach. *Remote Sensing of Environment*, 83(1-2):263–286, 2002.

[22] P.-N. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining.* Addison-Wesley, Boston, MA, 2006.

[23] J. Verbesselt, R. Hyndman, G. Newnham, and D. Culvenor. Detecting trend and seasonal changes in satellite image time series. *Remote sensing of Environment*, 114(1):106–115, 2010.

PSEUDO-LABEL GENERATION FOR MULTI-LABEL TEXT CLASSIFICATION

MOHAMMAD SALIM AHMED[1], LATIFUR KHAN[1], AND NIKUNJ OZA[2]

ABSTRACT. With the advent and expansion of social networking, the amount of generated text data has seen a sharp increase. In order to handle such a huge volume of text data, new and improved text mining techniques are a necessity. One of the characteristics of text data that makes text mining difficult, is multi-labelity. In order to build a robust and effective text classification method which is an integral part of text mining research, we must consider this property more closely. This kind of property is not unique to text data as it can be found in non-text (e.g., numeric) data as well. However, in text data, it is most prevalent. This property also puts the text classification problem in the domain of *multi-label classification (MLC)*, where each instance is associated with a subset of class-labels instead of a single class, as in conventional classification. In this paper, we explore how the generation of pseudo labels (i.e., combinations of existing class labels) can help us in performing better text classification and under what kind of circumstances. During the classification, the high and sparse dimensionality of text data has also been considered. Although, here we are proposing and evaluating a text classification technique, our main focus is on the handling of the multi-labelity of text data while utilizing the correlation among multiple labels existing in the data set. Our text classification technique is called *pseudo-LSC (pseudo-Label Based Subspace Clustering)*. It is a subspace clustering algorithm that considers the high and sparse dimensionality as well as the correlation among different class labels during the classification process to provide better performance than existing approaches. Results on three real world multi-label data sets provide us insight into how the multi-labelity is handled in our classification process and shows the effectiveness of our approach.

1. Introduction

Classification is an important part of text data analysis as has been pointed out in text research over a long period of time. With the increase of its volume, it has become necessary that we find automated means for text classification. However, text data is different from its non-text counterpart in a number of ways. The first difference that we look into and address in this paper is that text data tends to address multiple topics at the same time. As a result, they can be associated with multiple class labels giving rise to multi-labelity. And, these class labels are not independent of one another, indicating the existence of correlation or label dependence across the class labels. One of the main contributions of this paper is to take this correlation into account during the classification process.

We must also consider the high and sparse dimensionality of text data. All documents in text data sets are written in plain language. Since the vocabulary of any natural language is vast, the dimensionality is very high and compared to the whole vocabulary, only a few words appear in each document which gives rise to the sparseness.

[1] The University of Texas at Dallas, salimahmed@utdallas.edu, lkhan@utdallas.edu
[2] NASA Ames Research Center, nikunj.c.oza@nasa.gov.

Another important consideration during text classification is the availability of labeled data. Manual labeling of data is a time consuming task and as a result, in many cases, they are available in limited quantity. If we consider just the labeled data, then we are sometimes left with too little data to build a classification model that can perform well. On the other hand, if we ignore the class label information of the labeled data, then we are forsaking valuable information that could allow us to build a better classification model.

If we look into the literature, we see that usually, text classification approaches focus on a specific characteristic of text data such as its high dimensionality, multi-labelity or availability of limited labeled data. As a result, many of these methods can not be used universally. Sometimes, the underlying theory of these methods may become incorrect. For example, *Entropy Weighting K Means* approach [12] uses a subspace clustering approach that is based on the entropy of the features or dimensions. If the data is multi-label, then the entropy calculation of that method no longer holds ground. This happens in case of *SISC* [3], too, which is our previously formulated semi-supervised subspace clustering algorithm that considers both the high dimensionality and limited labeled data challenges. In *SISC* [3], however, the measure that becomes incorrect is the class impurity calculation. Also, it is only applicable for multi-class text data, not multi-label data, let alone considerations of label correlation.

In face of all these challenges, traditional as well as state-of-the-art text classification approaches perform poorly on multi-label data sets as we have found through our experiments. In order to address this multi-labelity scenario, we extended *SISC* to formulate *SISC-ML*[4] which is a multi-label variation of *SISC*. However, if we look closely into the data, we find that not all the classes co-occur with the same frequency. Which implies that the correlation among different class labels are not the same. In order to incorporate this correlation information during the clustering process, We, therefore, extended *SISC* [3] further based on this correlation information and formulated *pseudo-LSC* in this paper.

The reason behind choosing *SISC* as our classification method for extension is due to its notion of subspace clustering. Subspace clustering allows us to find clusters in a weighted hyperspace [10] and can aid us in finding documents that form clusters in only a subset of dimensions. In our proposed *pseudo-LSC (pseudo-Label Based Subspace Clustering)* approach, we augment the original class labels in the data set with *pseudo-labels* which are actually combinations of multiple class labels. Assigning such pseudo-labels allows us to use the correlation among different class labels during clustering and to achieve better classification performance.

In short, we have a number of contributions in this paper. First, *pseudo-LSC* is a semi-supervised subspace clustering algorithms that successfully finds clusters in the subspace of dimensions irrespective of the data being multi-class or multi-label. Second, our proposed algorithm performs well in practice even when a very limited amount of labeled training data is available. Third, at the same time, this algorithm minimizes the effect of high dimensionality and its sparse nature during training. Finally, we compare *pseudo-LSC* with other classification and clustering approaches to show the effectiveness of our algorithms over three benchmark multi-label text data sets.

The organization of the paper is as follows: Section 2 discusses related works. Section 3 presents the theoretical background of *pseudo-LSC*, the semi-supervised multi-class text classification approach. Section 4, then describes how *pseudo-LSC* handles multi-labeled data. Section 5 discusses the data sets, experimental setup and evaluation of our approach. Finally, Section 6 concludes with directions to future research.

2. Related Work

We can divide our related work based on the characteristic of our proposed *pseudo-LSC*. *pseudo-LSC* is a semi-supervised approach, it uses subspace clustering, and most important of all, it can handle multi-label data. Therefore, we have to look into the state-of-the-art methods that are already in the literature for each of these categories of research.

Approaches that have been proposed to address multi-label text classification, including margin-based methods, structural SVMs [19], parametric mixture models [21], κ-nearest neighbors (κ-NN) [25], and ensemble pruned methods [16]. One of the most recent works include *RAndom k-labELsets (RAKEL)* [20]. In a nutshell, it constructs an ensemble of *LP (Label Powerset)* classifiers and each *LP* is trained using a different small random subset of the multi-label set. Then, ensemble combination is achieved by thresholding the average zero-one decisions of each model per considered label. *MetaLabeler* [18] is another approach which tries to predict the number of labels using *SVM* as the underlying classifier. Most of these methods utilize the relationship between multiple labels for collective inference. One characteristic of these models is they are mostly supervised [16, 20, 18]. Aside from multi-label text classification, there are also work on regret analysis and loss function for such classification. In [9], Dembczynski et al. compare two loss functions namely subset 0/1 loss and Hamming loss for different multi-label classifiers. They focus mainly on the close connection between conditional label dependence and loss minimization. Unlike their approach, we are utilizing the unconditional label correlation that exists in the data as well as cluster impurity minimization.

Semi-supervised methods for classification is also present in the literature. This approach stems from the possibility of having both labeled and unlabeled data in the data set and in an effort to use both of them in training. In [5], Bilenko et al. propose a semi-supervised clustering algorithm derived from *K-Means*, *MPCK-MEANS*, that incorporates both metric learning and the use of pairwise constraints in a principled manner. There have also been attempts to find a low-dimensional subspace shared among multiple labels [12]. In [24], Yu et al. introduce a supervised *Latent Semantic Indexing (LSI)* method called *Multi-label informed Latent Semantic Indexing (MLSI)*. *MLSI* maps the input features into a new feature space that retains the information of original inputs and meanwhile captures the dependency of output dimensions. Our method is different from this algorithm as our approach tries to find clusters in the subspace. Due to the high dimensionality of feature space in text documents, considering a subset of weighted features for a class is more meaningful than combining the features to map them to lower dimensions [12]. In [7] a method called *LPI* is proposed. *LPI* is different from *LSI* which aims to discover the global Euclidean structure whereas *LPI* aims to discover the local geometrical structure. But *LPI* only handles multi-class data, not multi-label data. In [17] must-links and cannot-links, based on the labeled data, are incorporated in clustering. But, if the data is multi-label, then the calculation of must-link and cannot-link becomes infeasible as there are large number of class combinations and the number of documents in each of these combinations may be very low. As a result, this framework can not perform well when using multi-label text data.

There has been some subspace clustering approaches to minimize the impact of high dimensionality on classification. Subspace clustering can be divided into hard and soft subspace clustering. In case of hard subspace clustering, an exact subset of dimensions are discovered whereas soft subspace clustering determines the subsets of dimensions according to the contributions of the dimensions in discovering corresponding clusters. Examples of hard subspace clustering include *CLIQUE* [2], *PROCLUS* [1], *ENCLUS* [8] and *MAFIA* [11]. A hierarchical subspace clustering approach with automatic relevant dimension selection, called *HARP*, was presented by Yip et al. [23]. *HARP* is based on the assumption that two objects are likely to belong to the same cluster if they are very similar to each other along many dimensions. But, in multi-label and high dimensional text environment, the accuracy of *HARP* may drop as the basic assumption becomes less valid. In [14], a subspace clustering method called *nCluster* is proposed. But, it has similar problems when dealing with multi-label data. In [22], Wang et al. focuses on an ensemble approach and proposes a nonparametric Bayesian clustering ensemble method to discover the number of clusters for consensus clustering. In [13], SciForest has been proposed which uses clustering for finding group of anomalies/outliers. There, Liu et al. employ a split selection criterion to choose a split that separates clustered anomalies from normal points. They also makes use of randomly generated hyper-planes in order to provide suitable projections that separate anomalies from normal points. However, such a clustering is not applicable for multi-label text classification. Other soft clustering include [6] where spectral decomposition of the normalized affinity matrix is performed. The affinity matrix indicates the similarity measure between any two instances in the training set and therefore, depends too much on the quality of the similarity measure. Also, [6] focuses on using such clustering on graph data rather than text data.

pseudo-LSC uses subspace clustering in conjunction with κ-*NN* approach. In this light, it is closely related to the work of Jing et al. [12], Frigui et al. [10] and Ahmed et al. [3]. The closeness is due to the subspace clustering and fuzzy framework respectively. A significant difference with Frigui et al. [10] is that, unlike *pseudo-LSC*, it is unsupervised in nature. Another work that is closely related to ours is the work of Masud et al. [15]. In [15], a semi-supervised clustering approach called *SmSCluster* is used. They have used simple *K-Means Clustering* and it is specifically designed to handle evolving data streams. Finally, *SISC* [3] is another subspace clustering approach which has close resemblance to our approach. But, it is designed for only multi-class data. It has a multi-label variation called *SISC-ML* [4]. However, as mentioned previously, *SISC-ML* does not consider the class label correlation and assumes the class labels to be independent of each other. Our proposed *pseudo-LSC* is different in this respect as it does not make such class label independence assumption. It is also not specific for multi-class or multi-label data as is the case for *SISC* or *SISC-ML*. *pseudo-LSC* can work with a data set irrespective of it being multi-class or multi-label and therefore, addresses many of the challenges associated with text classification simultaneously.

Notation	Range	Explanation
x_i	$i = 1 : n$	i-th data point in the n document data set
d_j	$j = 1 : m$	j-th binary feature of m unigram features for data point x_i
t_i	$i = 1 : p$	i-th class of p classes in the data set where $p = \lvert\mathcal{T}\rvert$
c_l	$l = 1 : k$	l-th cluster of k subspace clusters
w_l	$l = 1 : k$	Membership weight of data point x_i of l-th subspace cluster
Lc_l	-	Total number of labeled points in cluster c_l

Table 1. SISC Notations

3. Multi-Label Classification

In this section, we describe the *MLC* problem in more detail and formalize it from a soft-clustering perspective. Along the way, we introduce the notations used throughout the paper.

3.1. Problem Statement.

Let $\mathcal{X}$ denote the training instance space $\hat{\mathcal{X}}$ denote the test set. Also, let $\mathcal{T} = t_1, t_2, \ldots, t_p$ be a finite set of class labels. We assume that any instance x across the training and test set is associated with a subset of labels $T \in 2^{\mathcal{T}}$; this subset is often called the set of relevant class labels while the complement of T is considered as irrelevant for x. Our goal is to predict the probability of a test instance $\hat{x}_i$ to belong to each class label $t_r, r = 1 : p$. In short, for each test instance $\hat{x}_i \in \hat{\mathcal{X}}$, we generate a class label vector $\boldsymbol{y} = (y_1, y_2, \ldots, y_p)$, in which $y_i = [0, 1]$ and $\sum_{i=1}^{p} y_i = 1$.

In this paper, we define a multi-label classifier $\mathbf{h}$ as an $\mathcal{X} \to \mathcal{Y}$ mapping that assigns a class-label vector $\boldsymbol{y}_i \in \hat{\mathcal{Y}}$ to each test instance $\hat{x}_i \in \hat{\mathcal{X}}$. Therefore, the problem of *MLC* can be stated as follows:

Given training data in the form of a finite set of observations $(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{X} \times \mathcal{Y}$, the goal is to learn a classifier $\mathbf{h} : \mathcal{X} \to \mathcal{Y}$ that generalizes well beyond the training observations. Table 1 specifies some of the notations that will be used throughout this paper.

It should be noted that since we are using a soft subspace clustering formulation, each training instance $x_i \in \mathcal{X}$ is a member of all the k subspace clusters (but with different membership weights). Apart from these notations, the following *two* measures are also used in *pseudo-LSC* as has been defined for *SISC* in [3].

3.2. Description of pseudo-LSC.

In *pseudo-LSC*, each data point may belong to multiple clusters. The weight with which a data point belongs to a particular cluster is referred to as *cluster membership weight.* For a data point, these membership weights across all the clusters sum up to 1. So, the membership weights can be regarded as probabilities with which a data point belongs to a cluster. Also, *pseudo-LSC* applies subspace clustering and the weight of a dimension in a cluster represents the probability of contribution of that dimension in forming that cluster. These dimension weights within a cluster are kept as a vector and the different dimension vectors of the clusters indicate how the clusters are different from one another. We, therefore, have to update three parameters during our clustering process - the *dimension weights* within each cluster, the

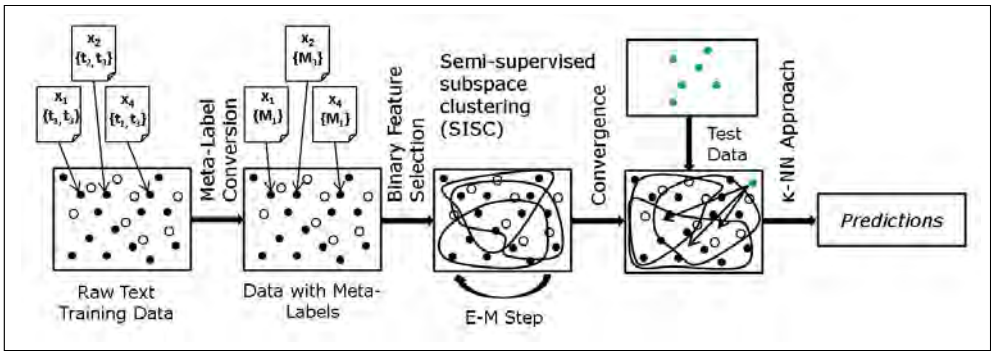


FIGURE 1. pseudo-LSC Top Level Diagram

cluster membership weights of each data point and the *cluster centroids*. *pseudo-LSC* utilizes the *Expectation-Maximization(E-M)* approach that locally minimizes the following objective function.

$$F(W,Z,\Lambda) = \sum_{l=1}^{k}\sum_{j=1}^{n}\sum_{i=1}^{m} w_{lj}^{f}\lambda_{li}^{q}D_{lij} * (1 + Imp_l) + \gamma\sum_{l=1}^{k}\sum_{i=1}^{m}\lambda_{li}^{q}\chi_{li}^{2} \tag{1}$$

where

$$D_{lij} = (z_{li} - x_{ji})^2$$

subject to

$$\sum_{l=1}^{k} w_{lj} = 1, 1 \le j \le n, 1 \le l \le k, 0 \le w_{lj} \le 1$$

$$\sum_{i=1}^{m} \lambda_{li} = 1, 1 \le i \le m, 1 \le l \le k, 0 \le \lambda_{li} \le 1$$

In this objective function, W, Z and Λ represent the *cluster membership*, *cluster centroid* and *dimension weight* matrices respectively. Also, the parameter f controls the fuzziness of the membership of each data point, q further modifies the weight of each dimension (λ_{li}) of each cluster c_l and finally, γ controls the strength of the incentive given to the *Chi Square* component.

Our algorithm is formulated using the E-M approach. In the E-Step, the *dimension weights* and the *cluster membership weights* are updated. Initially, every data point has equal membership weights across the clusters and the dimensions are given equal weights, too. During the *dimension weight* and *cluster membership weight* update, the cluster impurity is calculated using the pseudo-labels, not the original class labels. In the M-Step, the centroids of the clusters are updated and the summary statistics, i.e., the representation (percentage) of each class label present in the cluster, is updated. During the summary calculation, the membership weights are used. In the final step, the κ nearest neighbor clusters are identified for each test point where κ is a user defined parameter. The distance is calculated in the subspace where the cluster resides. If κ is greater than 1, then during the class probability calculation, we multiply the class representation with the inverse of the subspace distance and then sum them up for each class across all the κ nearest clusters.

3.3. **Impurity Measure.** Each cluster $c_l, l = 1 : k$, has an *Impurity Measure* [15] associated with it. This measure quantifies the amount of impurity within each cluster c_l. If the data points belonging to c_l all have the same class label, then the *Impurity Measure* of this cluster Imp_l is 0. On the other hand, if more and more data points belonging to different class labels become part of cluster c_l, the *Impurity Measure* of this cluster Imp_l also increases. This component has been used to modify the dispersion measure for each cluster. Its use helps in generating purer clusters in terms of cluster labels. However, it should be noted that Imp_l can be calculated using only labeled data points. If there are very few labeled data points, then this measure does not contribute significantly during the clustering process. Therefore, we use $1 + Imp_l$, so that unlabeled data points can play a role in the clustering process. Using Imp_l in such a way makes *pseudo-LSC* semi-supervised.

$$Imp_l = ADC_l * Ent_l$$

Here, ADC_l indicates the *Aggregated Dissimilarity Count* and Ent_l denotes the entropy of cluster c_l. In order to measure ADC_l, we first need to define *Dissimilarity Count* [15], $DC_l(x_i, y_i)$:

$$DC_l(x_i, y_i) = |L_{c_l}| - |L_{c_l}(t)|$$

if x_i is labeled and its label $y_i = t$, otherwise $DC_l(x_i, y_i)$ is 0. L_{c_l} indicates the set of labeled points in cluster c_l. In short, it counts the number of labeled points in cluster c_l that do not have label t. Then ADC_l becomes

$$ADC_l = \sum_{x_i \in L_{c_l}} DC_l(x_i, y_i)$$

Summing up the ADC_l for all the class labels provide us with the ADC_l for the entire cluster. The *Entropy* of a cluster c_l, Ent_l is computed as

$$Ent_l = \sum_{t=1}^{|\mathcal{T}|} (-p_t^l * log(p_t^l))$$

where p_t^l is the prior probability of class t, i.e., $p_t^l = \frac{|L_{c_l}(t)|}{|L_{c_l}|}$. It can also be shown that ADC_l is proportional to the *gini index* of cluster c_l, $Gini_l$ [15]. But, in *pseudo-LSC* method, the data points have fuzzy cluster memberships. So, the ADC_l calculation needs to be modified to incorporate this concept into *pseudo-LSC*. Rather than using counts, the membership weights are used for the calculation. This is reflected in the probability calculation.

$$p_t^l = \sum_{j=1}^{n} w_{lj} * j_t$$

where, j_t is 1 if data point x_j is a member of class t, and 0 otherwise. This *Impurity Measure* is normalized using the global impurity measure, i.e., the impurity measure of the whole data set, before using in the subspace clustering formulation.

3.4. **Chi Square Statistic.** During the clustering process, it may happen that the clusters are formed using only a few features. However, if only a few features (e.g., 2 or 3 features) are involved in the clustering with their dimension weights being greater than 0, they may fail to play any role during the label prediction step. This may happen as those few features may never appear in a test document rendering us unable to ascertain the κ-NN clusters of a test data point. To prevent such a scenario to happen, this *Chi Square* component has been included in the objective function so that more features or dimensions have nonzero weights and can participate in the clustering process. It works against the dispersion component of the objective function to create a balancing effect and ensures that clusters are not formed in just a few dimensions. From a clustering perspective, the conventional *Chi Square Statistic* can be defined as,

$$\chi^2_{li} = \frac{m(s_1 s_4 - s_2 s_3)^2}{(s_1 + s_3)(s_2 + s_4)(s_1 + s_2)(s_3 + s_4)}$$

where

$$\begin{aligned}
&s_1 = number\ of\ times\ feature\ d_i\ occurs\ in\ cluster\ c_l \\
&s_2 = number\ of\ times\ feature\ d_i\ occurs\ in\ all \\
&\quad clusters\ except\ c_l \\
&s_3 = number\ of\ times\ cluster\ c_l\ occurs\ without\ feature\ d_i \\
&s_4 = number\ of\ times\ all\ clusters\ except\ c_l\ occur \\
&\quad without\ feature\ d_i \\
&m = number\ of\ dimensions
\end{aligned}$$

This *Chi Square Statistic* χ^2_{li} indicates the measure for cluster c_l and dimension d_i. However, if the conventional approach is used for calculation of s_1, s_2, s_3, s_4 and m, then a threshold has to be specified to determine which point can be regarded as a member of a cluster. This not only brings forth another parameter, but also the membership values themselves are undermined in the calculation. So, *pseudo-LSC* modifies the calculation of these counts to consider the corresponding membership values of each point. The modification is provided below:

$$\begin{aligned}
&s_1 = \sum_{j=1}^{n} \sum_{d_i \in x_j} w_{lj}, \qquad s_2 = 1 - \sum_{j=1}^{n} \sum_{d_i \in x_j} w_{lj} \\
&s_3 = \sum_{j=1}^{n} \sum_{d_i \notin x_j} w_{lj}, \qquad s_4 = 1 - \sum_{j=1}^{n} \sum_{d_i \notin x_j} w_{lj} \\
&m = total\ number\ of\ labeled\ points
\end{aligned}$$

3.5. **Update Equations.** Minimization of F in Eqn. 5 with the constraints, forms a class of constrained nonlinear optimization problems. This optimization problem can be solved using partial optimization for Λ, Z and W. Detailed derivation of the update equations can be found in [3].

3.5.1. *Dimension Weight Update Equation.* Given matrices W and Z are fixed, F is minimized if

$$\lambda_{li} = \frac{1}{M_{lij} \sum_{i=1}^{m} \frac{1}{M_{lij}}} \tag{2}$$

where

$$M_{lij} = \left\{ \sum_{j=1}^{n} w_{lj}^{f} D_{lij} * (1 + Imp_l) + \gamma \chi_{li}^{2} \right\}^{\frac{1}{q-1}}$$

3.5.2. *Cluster Membership Update Equation.* Similar to the dimension update equation, the update equations for cluster membership matrix W can be derived, given Z and Λ are fixed. The update equation is as follows:

(3)
$$w_{lj} = \frac{1}{N_{lij} \sum_{l=1}^{k} \frac{1}{N_{lij}}}$$

where

$$N_{lij} = \left\{ \sum_{i=1}^{m} \lambda_{li}^{q} D_{lij} \right\}^{\frac{1}{f-1}}$$

3.5.3. *Cluster Centroid Update Equation.* The cluster center update formulation is similar to the formulation of dimension and membership update equations. The update equations for cluster center matrix i.e., Z can be derived, given W and Λ are fixed. The update equation is as follows:

(4)
$$z_{li} = \frac{\sum_{j=1}^{n} w_{lj}^{f} x_{ij}}{\sum_{j=1}^{n} w_{lj}^{f}}$$

4. Handling Multi Labeled Data

The previously described *Impurity Measure* calculation is only applicable for multi-class data where each document may belong to only a single class label. This constraint ensures that the calculated probabilities always sum up to 1. However, if each data point may belong to more than one class label, the sum of probabilities may become greater than 1. One way would be to convert the classification problem into T binary class problems and modify the *Impurity Measure* to handle the multi-label data in such a way. But, doing so is only feasible if all the class labels are independent of each other. Our experience with multi-label data also indicate that there is a correlation among the different class labels. In order to handle the multi-labelity as well as the co-occurrence of the class labels, our proposed *pseudo-LSC* generates *pseudo-labels* which are combinations of one or more original class labels in the data set. In short, we transform the multi-label data set into a multi-class data set where each data point can belong to only a single *pseudo-label.* The following example illustrates how the *pseudo-labels* are generated.

As can be seen from the example in Table 2, the 5 data points belong to 3 *pseudo-labels.* And each of the *pseudo-labels* may constitute of one or more original class labels in the data set. After assigning such *pseudo-labels* to the data points, the new data set becomes multi-class. Therefore, the *pseudo-LSC* multi-class algorithm becomes applicable to such a data set. In Figure 3.1, we show how the text data is converted from its original label to *pseudo-labels.* In this example, data points x_1, x_2 and x_3 are assigned *pseudo-labels* p_1, p_2 and p_3 respectively. This pseudo-label generation is

Data	Labels
x_1	t_1, t_3
x_2	t_1, t_2, t_4
x_3	t_2
x_4	t_1, t_3
x_5	t_2

Pseudo Labels	Label Sets
p_1	t_1, t_3
p_2	t_1, t_2, t_4
p_3	t_2

Data	Pseudo Labels
x_1	p_1
x_2	p_2
x_3	p_3
x_4	p_1
x_5	p_3

Table 2. Construction of Pseudo Labels In pseudo-LSC

only applicable during the *Impurity Measure* calculation. The original class labels are used in all other calculations during the classification process.

5. Experiments and Results

We have used a total of three multi-label data sets to verify the effectiveness of our algorithm on multi-label data. In all cases, we used fifty percent data as training and rest as test data in our experiments as part of 2-fold cross-validation. Similar to other text classification approaches, we performed preprocessing on the data and removed stop words from the data. We used binary features as dimensions, i.e. features can only have 0 or 1 values. The parameter γ is set to 0.5. For convenience, we selected 1000 features based on information gain and used them in our experiments. In all the experiments related to a data set, the same feature set was used. We performed multiple runs on our data sets. And in each case, the training set was chosen randomly from the data set.

5.1. **Data sets.** We describe here all the three data sets that we have used for our experiments.

(1) Reuters Data Set: This is part of the Reuters-21578, Distribution 1.0. We selected $10,000$ data points from the $21,578$ data points of this data set and henceforth, this part of the data set will be referred to as simply *Reuters* data set. We considered the most frequently occurring 20 classes in our experiments. Of the $10,000$ data points, $6,651$ are multi-labeled.
(2) 20 Newsgroups Data Set: This data set is also multi-label in nature. We selected $15,000$ documents randomly for our classification experiments. Of them $2,822$ are multi-label documents and the rest are single labeled. We have performed our classification on the top 20 classes of this data set.
(3) NASA ASRS Data Set: We randomly selected $10,000$ data points from the *ASRS* data set and henceforth, this part of the data set will be referred to as simply *ASRS* Data Set. We considered 21 class labels (i.e., anomalies) in our experiments.

5.2. **Base Line Approaches.** We have chosen 3 sets of baseline approaches. First, since we are using κ-nearest neighbor (κ-NN) approach along with clustering approach, we compare our method with the basic κ-NN approach. Second, we compare two subspace clustering approaches. They are *SCAD2* [10] and *K-means Entropy* [12] approaches. The reason behind using them as baseline approaches is that they have similarities in objective functions with our methods. So, a comparison with them will show the effectiveness of our algorithms from a subspace clustering perspective. Finally, we perform experiments using two multi-label methods and compare them to *pseudo-LSC*.

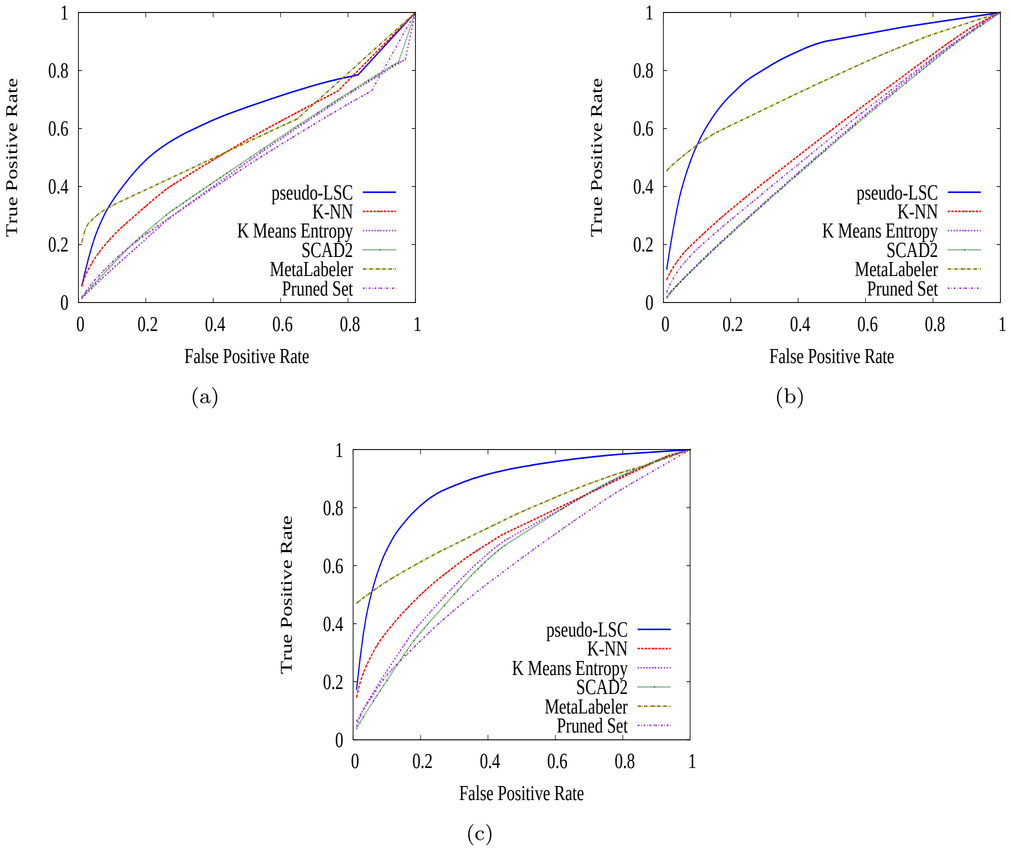


FIGURE 2. ROC Curves for (a) NASA ASRS Data Set (b) Reuters Data Set (c) 20 Newsgroups Data Set.

Methods	ASRS	Reuters	20 Newsgroups
pseudo-LSC	**0.637**	**0.821**	**0.874**
Pruned Set	0.469	0.56	0.60
MetaLabeler	0.58	0.762	0.766
κ-NN	0.552	0.585	0.698
SCAD2	0.482	0.533	0.643
K Means Entropy	0.47	0.538	0.657

TABLE 3. Area Under The ROC Curve Comparison Chart For Multi-Label Classification

They are *Pruned Set* [16] and *MetaLabeler* [18] approaches. Both these methods are state-of-the-art multi-label approaches and use *SVM* as their base classifiers. We, therefore, did not choose *SVM* as a baseline approach as it is already takes part in the comparison through these multi-label approaches. Also, it was not possible to used *SISC* [3] as it is applicable only for multi-class data, not multi-label data. Below we describe these 5 baseline approaches briefly.

5.2.1. *Basic κ-NN Approach.* In this approach, we find the nearest κ neighbors in the training set for each test point. Here κ is a user defined parameter. After finding the neighbors, we find how many of these neighbors belong to the t-th class. We perform this calculation for all the classes. We can then get the probability of the test point belonging to each of the classes by dividing the counts with κ. Finally, using these probabilities, for each class, we generate ROC curves and take their average to compare with our methods.

5.2.2. *K-Means Entropy.* This is another soft subspace clustering approach that we compare with *pseudo-LSC*. Its objective function has two components, the first one is based on dispersion and the second one is based on the negative entropy of cluster dimensions. Another difference between this approach and SCAD2 is that it is not fuzzy in nature. So, a training data point can belong to only a single cluster. The objective function that is minimized, as specified in [12] to generate the clusters, is as follows:

$$F(W, Z, \Lambda) = \sum_{l=1}^{k}\sum_{j=1}^{n}\sum_{i=1}^{m} w_{lj}\lambda_{li}D_{lij} + \gamma\sum_{l=1}^{k}\sum_{i=1}^{m}\lambda_{li}log(\lambda_{li}) \tag{5}$$

5.2.3. *SCAD2. SCAD2* [10] is a soft subspace clustering method with a different objective function than the *pseudo-LSC* method. This clustering method is also fuzzy in nature and can be considered the most basic form of fuzzy subspace clustering. As it does not consider any other factors during clustering except for dispersion. Its objective function has close resemblance to the first term of the *pseudo-LSC* objective function. As mentioned earlier, the reason we have used this method as benchmark is due to this similarity. The objective function of *SCAD2* is as follows:

$$F(W, Z, \Lambda) = \sum_{l=1}^{k}\sum_{j=1}^{n}\sum_{i=1}^{m} w_{lj}^{f}\lambda_{li}^{q}|x_{ij} - z_{li}|^2 \tag{6}$$

After performing this clustering using the same E-M formulation of *pseudo-LSC*, we use κ nearest clusters of each test point to calculate label probabilities.

5.2.4. *MetaLabeler.* This is a multi-label classification approach that learns a function from the data to the number of labels [18]. It involves two steps - i) constructing the meta data set and ii) learning a meta-model. Unlike our formulation of *pseudo-labels* in *pseudo-LSC*, the label of the meta data for this method is the number of labels for each instance in the raw data. There are three ways [18] that this learning can be done. We have applied the *Content-based MetaLabeler* to learn the mapping function from the features to the meta-labels (i.e., the number of class labels). As specified in [18], we consider the meta learning as a multi-class classification problem and use it in conjunction with *One-vs-Rest SVM.* We, therefore, train *T + 1 SVM* classifiers where *T* is the total number of class labels in the data set. Of them, one is a multi-class classifier and the rest are One-vs-Rest *SVM* classifiers for each of the classes. We then normalize the scores of the predicted labels and consider them as probabilities for generating *ROC curves.*

Methods	ASRS	Reuters	20 Newsgroups
pseudo-LSC	0.637	0.821	0.874
pseudo-LSC Without Chi Square	0.455	0.532	0.582

Table 4. Area Under The ROC Curve Comparison Chart For Chi Square Statistic

5.2.5. *Pruned Set.* The main goal of this algorithm is to transform the multi-label problem into a multi-class problem. In order to do so, *Pruned Set* [16] method finds frequently occurring sets of class labels. Each of these sets (or combinations) of class labels are considered as a distinct label. The benefit of using this approach is that, only those class label combinations that occur in the data set and the user does not need to consider an exponential amount of class label combinations. The user specifies parameters like what is the minimum count of a class label combination to consider it as frequent and the minimum size (i.e., class combinations having at least r class labels) of such sets or combinations.

At first, all data points with label combinations having sufficient count are added to an empty training set. This training set is then augmented with rejected data points having label combinations that are not sufficiently frequent. This is done by making multiple copies of the data points, only this time with subsets of the original label set. So, some data points may be duplicated during this training set generation process. This training set is then used to create an ensemble of *SVM* classifiers. We have also varied the number of retained label subsets to add to the training set and chose the best result to report.

5.3. **Evaluation Metric.** In all of our experiments, we use the *Area Under ROC Curve (AUC)* to measure the performance. For all the baseline approaches and our *pseudo-LSC* method, we generate each class label prediction as a probability. Then, for each class we generate an ROC curve based on these probabilities and the original class labels. After generating all the ROC curves, we take the average of them to generate a combined ROC curve. Finally, the area under this combined ROC curve is reported as output. This area can have a range from 0 to 1. The higher the AUC value, the better the performance of the algorithm.

5.4. **Results and Discussion.** As can be seen from Figure 2(a), *pseudo-LSC* performs much better than the baseline approaches. In Table 3, the *AUC* values for *pseudo-LSC* and all the baseline approaches are provided. With the *ASRS* data set, the *AUC* value for *pseudo-LSC* is 0.637. The closest performance is provided by the state-of-the-art *MetaLabeler* approach which is 0.58. Therefore, there is around 5%-8% increase in performance with our approaches.

Similar results can be found for *Reuters* and *20 Newsgroups* data sets. In Figure 2(b) and Figure 2(c), we provide these results. Just like the *ASRS* data set, *pseudo-LSC* provides much better results. For *Reuters* data set, our algorithm achieves AUC values of 0.821 and the nearest baseline approach value is 0.762. And, for *20 Newsgroups* data set, the AUC value achieved is 0.874 whereas, the nearest value is 0.766.

5.5. **Impact of Chi Square Statistic.** We have included the *Chi Square Statistic* in our objective function to achieve better performance by ensuring that more features have nonzero dimension weights as opposed to only a few features having nonzero weights and generating the clusters over a larger subset of dimensions. This increases the probability that a test point will have some of those nonzero features making the distance calculation meaningful. We have performed experiments to determine the impact of this component on the classification performance. To do so, we have removed this component from the objective function and performed the same experiments. We found that using it indeed increases the performance quite significantly. The objective function used in this case is given below.

$$F(W, Z, \Lambda) = \sum_{l=1}^{k} \sum_{j=1}^{n} \sum_{i=1}^{m} w_{lj}^{f} \lambda_{li}^{q} D_{lij} * (1 + Imp_l) \qquad (7)$$

The results are provided in Table 4.

6. Conclusions

In this paper, we have presented *pseudo-LSC*, a semi-supervised text classification approaches based on fuzzy subspace clustering that considers the correlation among different class labels by generating pseudo labels during the clustering process. It provides a unified approach to perform classification on both multi-class and multi-label data. The experimental results on real world multi-labeled data sets like *ASRS*, *Reuters* and *20 Newsgroups*, have shown that *pseudo-LSC* outperforms *κ-NN*, *K-Means Entropy* based method, *SCAD2* and state-of-the-art multi-label text classification approaches like *Pruned Set* and *MetaLabeler* in classifying text data.

References

[1] C. C. Aggarwal, J. L. Wolf, P. S. Yu, C. Procopiuc, and J. S. Park. Fast algorithms for projected clustering. *SIGMOD Rec.*, 28(2):61–72, 1999.

[2] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. *SIGMOD Rec.*, 27(2):94–105, 1998.

[3] M. S. Ahmed and L. Khan. SISC: A text classification approach using semi supervised subspace clustering. *DDDM '09: The 3rd International Workshop on Domain Driven Data Mining in conjunction with ICDM 2009*, Dec. 2009.

[4] M. S. Ahmed, L. Khan, N. Oza, and M. Rajeswari. Multi-label ASRS dataset classification using semi-supervised subspace clustering. *In Conference on Intelligent Data Understanding (CIDU) 2010*, September 2010.

[5] M. B. Basu; and R. J. Mooney. Integrating constraints and metric learning in semi-supervised clustering. In *ICML*, pages 81–88, 2004.

[6] F. Bavaud. Euclidean distances, soft and spectral clustering on weighted graphs. In *Proceedings of the 2010 European conference on Machine learning and knowledge discovery in databases: Part I*, ECML PKDD'10, pages 103–118, Berlin, Heidelberg, 2010. Springer-Verlag.

[7] D. Cai, X. He, and J. Han. Document clustering using locality preserving indexing. *IEEE Transactions on Knowledge and Data Engineering*, 17(12):1624–1637, Dec. 2005.

[8] C.-H. Cheng, A. W. Fu, and Y. Zhang. Entropy-based subspace clustering for mining numerical data. In *KDD '99: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 84–93, New York, NY, USA, 1999. ACM.

[9] K. Dembczyński, W. Waegeman, W. Cheng, and E. Hüllermeier. Regret analysis for performance metrics in multi-label classification: the case of hamming and subset zero-one loss. In *Proceedings of the*

2010 European conference on Machine learning and knowledge discovery in databases: Part I, ECML PKDD'10, pages 280–295, Berlin, Heidelberg, 2010. Springer-Verlag.

[10] H. Frigui and O. Nasraoui. Unsupervised learning of prototypes and attribute weights. *Pattern Recognition*, 37(3):567 – 581, 2004.

[11] S. Goil, H. Nagesh, and A. Choudhary. Mafia: Efficient and scalable subspace clustering for very large data sets. *Technical Report CPDC-TR-9906-010, Northwest Univ.*, 1999.

[12] L. Jing, M. K. Ng, and J. Zhexue Huang. An entropy weighting k-means algorithm for subspace clustering of high-dimensional sparse data. *IEEE Trans. Knowl. Data Eng.*, 19(8):1026–1041, 2007.

[13] F. T. Liu, K. M. Ting, and Z.-H. Zhou. On detecting clustered anomalies using sciforest. In *Proceedings of the 2010 European conference on Machine learning and knowledge discovery in databases: Part II*, ECML PKDD'10, pages 274–290, Berlin, Heidelberg, 2010. Springer-Verlag.

[14] G. Liu, J. Li, K. Sim, and L. Wong. Distance based subspace clustering with flexible dimension partitioning. In *IEEE 23rd International Conference on Data Engineering (ICDE)*, pages 1250–1254, April 2007.

[15] M. Masud, J. Gao, L. Khan, J. Han, and B. Thuraisingham. A practical approach to classify evolving data streams: Training with limited amount of labeled data. In *Eighth IEEE International Conference on Data Mining (ICDM)*, pages 929–934, Dec. 2008.

[16] J. Read, B. Pfahringer, and G. Holmes. Multi-label classification using ensembles of pruned sets. In *Eighth IEEE International Conference on Data Mining (ICDM)*, pages 995–1000, Dec. 2008.

[17] J. Struyf and S. Džeroski. Clustering trees with instance level constraints. In *ECML '07: Proceedings of the 18th European conference on Machine Learning*, pages 359–370, Berlin, Heidelberg, 2007. Springer-Verlag.

[18] L. Tang, S. Rajan, and V. K. Narayanan. Large scale multi-label classification via metalabeler. In *WWW '09: Proceedings of the 18th international conference on World wide web*, pages 211–220, New York, NY, USA, 2009. ACM.

[19] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun. Support vector machine learning for interdependent and structured output spaces. In *ICML '04: Proceedings of the twenty-first international conference on Machine learning*, page 104, New York, NY, USA, 2004. ACM.

[20] G. Tsoumakas and I. Vlahavas. Random k-labelsets: An ensemble method for multilabel classification. In *ECML '07: Proceedings of the 18th European conference on Machine Learning*, pages 406–417, Berlin, Heidelberg, 2007. Springer-Verlag.

[21] N. Ueda and K. Saito. Parametric mixture models for multi-labeled text. *In Advances in Neural Information Processing Systems 15. Cambridge: MIT Press.*, 2003.

[22] P. Wang, C. Domeniconi, and K. B. Laskey. Nonparametric bayesian clustering ensembles. In *Proceedings of the 2010 European conference on Machine learning and knowledge discovery in databases: Part III*, ECML PKDD'10, pages 435–450, Berlin, Heidelberg, 2010. Springer-Verlag.

[23] K. Yip, D. Cheung, and M. Ng. Harp: a practical projected clustering algorithm. *IEEE Transactions on Knowledge and Data Engineering*, 16(11):1387–1397, Nov. 2004.

[24] K. Yu, S. Yu, and V. Tresp. Multi-label informed latent semantic indexing. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 258–265, New York, NY, USA, 2005. ACM.

[25] M.-L. Zhang and Z.-H. Zhou. Ml-knn: A lazy learning approach to multi-label learning. *Pattern Recognition*, 40(7):2038 – 2048, 2007.

ON THE STATISTICS AND PREDICTABILITY OF GO-AROUNDS

MAXIME GARIEL*, KEVIN SPIESER*, AND EMILIO FRAZZOLI*

ABSTRACT. This paper takes an empirical approach to identify operational factors at busy airports that may predate go-around maneuvers. Using four years of data from San Francisco International Airport, we begin our investigation with an analysis of sequence of landing aircraft that may increase the probability of go-around occurrence. Then we take a statistical approach to investigate which features of airborne, ground operations (e.g., number of inbound aircraft, number of aircraft taxiing from gate, etc.) or weather are most likely to fluctuate, relative to nominal operations, in the minutes immediately preceding a missed approach. We analyze these findings both in terms of their implication on current airport operations and discuss how the antecedent factors may affect NextGen. Finally, as a means to assist air traffic controllers, we draw upon techniques from the machine learning community to develop a preliminary alert system for go-around prediction.

1. INTRODUCTION

A missed approach or go-around (GA) occurs when an aircraft aborts its landing and is forced, instead, to land on a subsequent approach. It is tempting to speculate that lack of visibility at decision height or pilot error, that is a pilot's inability to safely land the aircraft in a given situation, are a leading cause of GAs. Indeed, low ceiling and visibility increase the potential for missed-approach and the workload for pilots and controllers [10]. Nevertheless, as it is shown in this paper, weather may only accounts for a small fraction of the total number of GA. However, interviews with air-traffic controllers during visits to Logan International Airport in Boston and Laguardia Airport in New York refute these claims; rather, the controller's testimony suggests operational errors, such as a runway incursion, late runway departure from an aircraft taking off or holding position line violation are the primary causes of GAs. The remainder of this paper analyzes the arrivals at Sand Francisco International Airport (SFO). Before giving an expert's view on GA at SFO, we present some motivation for this study.

To increase airport throughput, NextGen's high-density operations [11] are projected at more airport's than today's class B airports. High-density operations require high performance procedures and aircraft equipage to enable Closely Spaced Parallel Approaches with delegated separation procedures. Despite the increased automation, the technologies, there will always be errors or unexpected events leading to missed approach. Avionics for de-conflicted missed approaches for converging is still in the roadmap of NextGen but has not been addressed yet [9]. To take full advantage for these high throughput operations, the number of missed approaches needs to be minimized, and therefore a thorough understanding of the factors that lead to missed approach is necessary.

Go-arounds increase the workload of air traffic controllers, as landing sequences must be amended to accommodate the aircraft that failed to land [2]. In addition, a GA is a loss of a landing spot, a scarce resource, reducing the capacity of the airport. On a related note, the requisite revamping schedules taxes an airport system that maintains high-safety standards largely through comprehensive planning and delegation,

This research is supported by NASA grant # NNX08AY52A..

The authors are with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA, 02139. {mgariel,kspieser,frazzoli}@mit.edu.

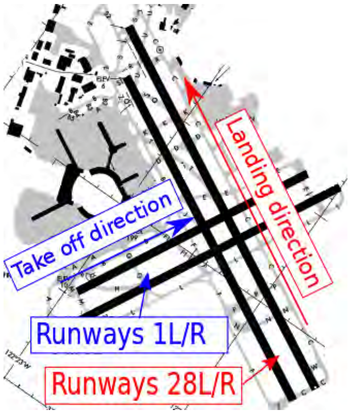


FIGURE 1. Simplified SFO diagram and selected runway configuration

go-arounds are also undesirable from a safety perspective [14]. Finally, go-arounds are costly for airlines, both in terms of the added fuel cost and the logistic delays absorbed from spending extra time airborne [13].

This report is an exploratory study aimed at addressing the following questions i.) can we identify factors that lead to GAs and ii.) if so, can these causes be identified in real-world data sets from major airports to predict GAs? If the factors causing GAs can be identified, then mitigation action can be taken in order to reduce the number of GAs without impacting airport throughput.

Presentation of SFO airport. San Francisco International Airport (SFO) is notorious for two things. Its unpredictable weather and its parallel runways. The weather will be analyzed in the next section. Regarding the runways, SFO has two sets of close parallel runways, as depicted in Figure 1. In the runway configuration illustrated, aircraft take-off from runways 1R-L (parallel runways), and land on runway 28R-L (also parallel runways, separated by only 750 ft); this is the configuration in use approximately 80 percent of the time. Due to the close proximity of the landing runways, the runways cannot be used for simultaneous instrument landings, reducing the hourly landing capacity from 60 to 30 aircraft. The weather conditions are called Visual Meteorological Conditions (VMC) and Instrument Meteorological Conditions (IMC). The modes of operation for the TRACON and the SFO tower differ between Visual Flight Rule (VFR) operations during VMC and Instrument Flight Rules (IFR) during IMC. VMC conditions enable simultaneous parallel landings while IMC required single file landings. Therefore, the causes for GA may differ from one more of operation to the other.

According to a manager from the SFO tower, and from reports from the Northern California TRACON, there are many reasons for aircraft to be sent around. GA may be initiated either by pilots or by Air Traffic Control (ATC). The principal causes are explained below.

Presence of another aircraft on the surface: One of the major factors that trigger GA is the presence of another aircraft on the active runway when a landing aircraft passes over the runway threshold. It is often the preceding landing aircraft that has not cleared off the runway yet. In the configuration depicted on Figure 1, it may also be an aircraft taking-off on runways 1R or L that has not cleared the runway intersection when the landing aircraft passes over the threshold of runway 28 R or L.

Pilot initiated GA: One of the main causes for pilots to initiate a GA is an unstable approach. An unstable approach can be due to a pilot accepting, from ATC, a short turn to final and then realizing that the aircraft is too high or too fast. Such high-energy/unstable approaches present a safety risk and therefore pilots may decide to go-around. Another reason for pilot initiated GA are wind-shear alerts which are safety critical alerts. Finally, and particularly at night, a pilot may initiate a GA after having been cleared for a VFR landing and finding themselves in a cloud, since ATC cannot see the clouds.

Pilot initiated GA, parallel landings: Another factor that trigger GA at SFO are alerts from the Traffic Collision Avoidance System (TCAS) [16]. TCAS is a system that alerts pilots of potential collisions or near mid-air collision with other aircraft. While working well in en-route environment, it presents a high level of nuisance alerts during close parallel approaches [6]. These procedures often trigger a TCAS alerts, which can be disregarded if pilots have the other aircraft in sight and follow ATC instructions. While disregarded by most pilots, some airlines with very strict company procedures mandate pilots to nevertheless execute a GA.

ATC initiated GA, single file landings: During IFR operations, when landings occur in a single file of aircraft, the compression effect may lead to loss of separation. The compression effect is the result of aircraft significantly slowing down just before landing. If the difference in ground speed between an aircraft about to land and its follower becomes to large, then the compression effect is also large and may result in a time separation at landing not long enough. Therefore, ATC have to initiate a GA.

ATC initiated GA, parallel landings: There are two important rules for the parallel landing procedures. These rules regulate overtakings when aircraft are side-by-side. There exist 4 categories of aircraft when dealing with wake vortices: Small, Large, Heavy and 757. The first rule is that no large should overtake a small and the second rule is that no heavy or 757 may overtake any other aircraft. If one of the rule is about to be broken, ATC will require one of the two aircraft to go-around. The aircraft sent around can be either the one over-taking or the one over-taken. The controller decides which option is safer.

Figure 2 presents the causes of GA as recorded by the Northern California TRACON (NCT). These data correspond to all the GA from September 2010 to March 2011, with a total of 356 GA [15]. These reports suggest that 35% of the GA are initiated by pilots or may be due to an equipment problem. Traffic on the runway accounts for 27% of the GA and the compression effect for 22%. Then, re-sequencing mishap account for less than 11% of the GA and finally, other factors such as TCAS alerts or earthquakes account for less than 6%.

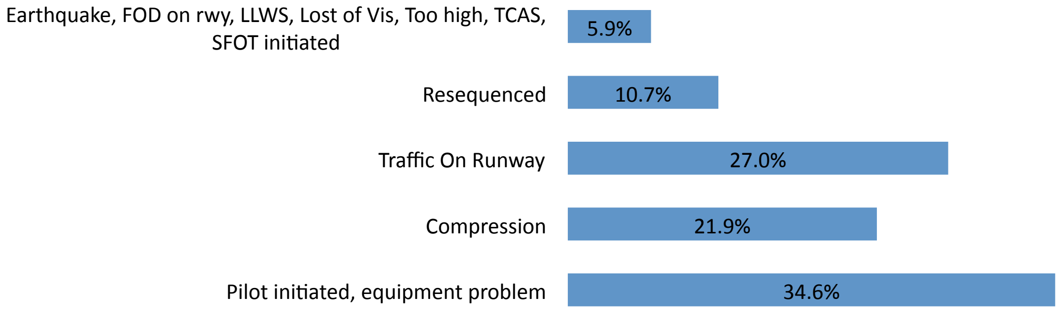


FIGURE 2. Go-Around causes as recorded by the NCT

These reports present an “after-the-facts” analysis of the GA, seen from the air traffic controllers. In this paper, we try to identify conditions that may increase the probability of GA or explanatory factors, in a more detailed manner than simply saying “compression”. For instance, is there a particular traffic configuration that increases the probability of GA due to compression?

The remainder of this report is structured as follows. Section 2 describes the datasets used to construct our corpus of flight data. Section 4 provides a statistical analysis of conditions that may be closely associated with GAs and Section 5 explores the idea of using these findings as a means to predict GAs. Conclusions are presented in Section 6.

2. Data presentation and selection

In an attempt to identify factors that precipitate a GA, data was collected from three complimentary datasets over a four-year period spanning January 2006 to December 2009. Following cleaning and synchronization, the data was combined to create a single master dataset. The objective of this dataset is to create a "state" vector for the system at every minute. In this sense, the system in question is comprised of the airport, the airspace surrounding it, and the weather conditions. The generated state vectors also contain information about the past (averages and changes over 5, 10 and 15 minutes intervals).

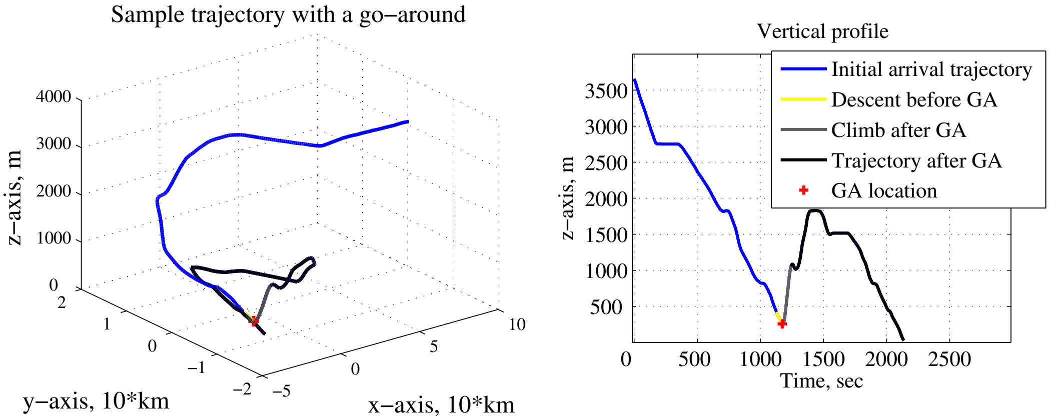
2.1. **Data presentation. Airspace data:** This dataset contains data for all of the flights recorded by the secondary radar located at Oakland International Airport (OAK). It is from the Automated Radar Terminal System (ARTS). A 3 month sample of the data is available for download on DashLink [4] In the full data set, it appears the range of the radar was increased from 45 to 60 NM over the four years of interest. To ensure consistency over the 4 year period only the data in a radius smaller than 45 NM was kept. For each flight, the dataset contains the aircraft's 4-D trajectory as well as metadata such as flight identification, origin airport, destination airport, etc. We kept only flights departing or arriving at one of the three largest airports in the Bay Area, that is San-Francisco International Airport (SFO), Oakland International Airport (OAK) and San Jose International Airport (SJC).

Ground data: The ground data was extracted from the Aviation System Performance Metric (ASPM) flight database. This database contains a record of both the scheduled time and actual time for pullback from gate, takeoff, and landing for each aircraft at each major airport in the United States. This data is available for download from the Federal Aviation Administration (FAA) website [5]. To compliment the airspace fields, the relevant fields for all flights taking off or landing at SFO were extracted from the ASPM dataset.

Weather and runway data: The weather information and the runway configuration for SFO was extracted from the ASPM Airport database. This database is also available on the FAA website [5]. The database includes the weather (visibility, cloud ceiling, wind speed and direction) as well as the runway configuration in use. Figure 1 depicts the layout of SFO. In the configuration illustrated, aircraft take off from runways 1R-L (parallel runways), and land on runway 28R-L (also parallel runways); this is the configuration in use approximately 80 percent of the time. Among weather-related fields, the temperature was not always available at each time instant. In these cases, the temperature measurement used was obtained by interpolation neighboring entries.

Aircraft Category: The landing category of aicraft (heavy, large, small) was obtained from a database containing all the aircraft models and their wake vortex category. This category is determined from ICAO standards.

2.2. **GA identification.** To facilitate our investigation of GAs, we assembled a corpus of samples, each of which is one of two types: i.) samples of the airport state during nominal operations in which no GAs occur and ii.) samples of the airport state during a window in which a GA does occur. The following rule was used to label a flight as containing a GA: a flight contains a GA if during the plane's terminal flight phase, the plane's altitude increases for fifteen consecutive measurements following a period in which the plane descended for at least ten consecutive radar measurements. At the sample rate at which measurements were taken, this corresponds to approximately 70 seconds of continuous increase in altitude, following 45 seconds of continuous descent. This criterion identified nearly all GAs and was discerning enough to exclude the trajectories of helicopters and short-haul flights not associated with GAs. This method of detecting GAs was validated through manual verification on a large sample of trajectories. For example, Figure 3 shows a sample landing trajectory containing a GA. The blue line shows the portion of the trajectory prior to the GA. The yellow segment corresponds to the 45 seconds of descent preceding the GA. The instant at which the GA is initiated is indicated with a red cross. Finally, the grey line corresponds to the period of at least 70 seconds of climb following the GA. The remainder of the trajectory, including the eventual landing, is shown in black.



(a) Trajectory

(b) Vertical profile

FIGURE 3. Sample landing trajectory containing a GA

Figure 4 shows the number of GA gathered from the data, by quarter, starting in 2006. From top to bottom along each bar: red corresponds to GAs occurring when the runway configuration is different that landing on takeoffs on runway 1 R/L and landings on runways 28R/L (e.g. aircraft taking-off and landing on runways 10R/L) blue corresponds to the GA occurring on the selected runway configuration but at night (23:00 to 7:00); yellow corresponds to GA occurring during day time (7:00-23:00), on runways 28 R/L, but during IMC; and green corresponds to the corpus of GAs selected for this study, that is VMC on runways 28R/L, between 7:00 and 23:00. Due to missing data, these numbers do not reflect the exact count of GAs at SFO. The following section presents an analysis of landing sequences during IMC and VMC, and the associated probability of GA occurrence.

3. Landing Sequence Analysis

In this section, we analyze the sequence of aircraft that precedes a GA. The objective is to determine if some landing sequences are more likely to predate a GA.

3.1. **Sequence description.** Spacing between aircraft at landing is determined by aircraft weight categories. There are 4 categories: Small, Large, Heavy and 757 (There also exists a special category, "Super" just for the Airbus A380). Since rules regarding heavy aircraft and 757 are very similar, these two categories were

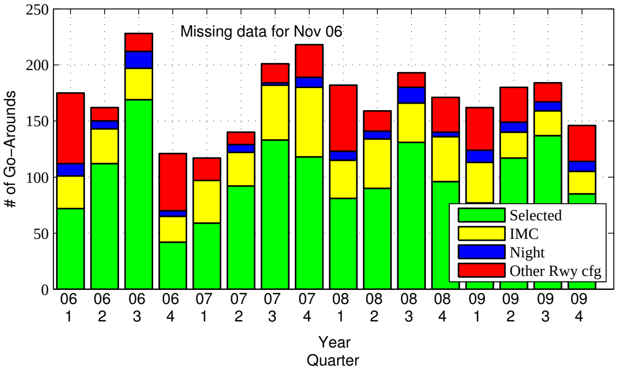


FIGURE 4. Distribution of GA

merged in this analysis under the denomination “heavy”. The separation between aircraft is due to the wake vortex created by aircraft; the heavier the aircraft, the bigger the wake vortex. For each aircraft, we created its preceding landing sequence. Tables 1 and 2 present the prior probability distribution for each type of preceding aircraft, for VMC and IMC, respectively. The prior correspond to the probability of an aircraft executing a GA given the type of preceding aircraft. The probability multiplication factor corresponds to the increase in probability of a GA with respect to the overall probability of occurrence of GA, $P(GA)$. This analysis encompasses all the runway configurations

TABLE 1. Increased probability of GA with respect to average probability of GA in VMC

Preceding aircraft *Prec*	Prior $P(GA\|Prec)$	Probability multiplication factor $P(GA\|Prec)/P(GA)$	Number of occurrences
No aircraft	1.12×10^{-2}	2.6775	24
Small	0.42×10^{-2}	1.0021	299
Large	0.39×10^{-2}	0.9246	798
Heavy	0.53×10^{-2}	1.2751	229
Any	0.42×10^{-2}	1	1350

TABLE 2. Increased probability of GA with respect to average probability of GA in IMC

Preceding aircraft *Prec*	Prior $P(GA\|Prec)$	Probability multiplication factor $P(GA\|Prec)/P(GA)$	Number of occurrences
No aircraft	2.4×10^{-2}	3.3094	7
Small	0.63×10^{-2}	0.8847	79
Large	0.68×10^{-2}	0.9465	284
Heavy	0.96×10^{-2}	1.3431	84
Any	0.71×10^{-2}	1	454

Table 1 shows that during VMC and in average, the risk of executing a GA is multiplied by 2.6 for aircraft with no other aircraft landing in the previous 10 minutes. Nevertheless, this is very unfrequent. It also suggests that the risk of GA is increased by 27.5% when following a heavy/757 aircraft. Table 2 shows the same trends, during IMC. GA are 3.3 times more likely to occur when there is no preceding aircraft than in average. Also, it is 34% more frequent when following a heavy/757 aicraft.

The following section will takes a different approach to the analysis by creating a state vector for the system, for every minute. Then, the empirical probability distribution functions for each parameter are compared between the nominal data set and the GA dataset.

4. Temporal analysis

This section presents a statistical analysis of the distribution of the different variable features for both nominal and GA flights. The nominal samples used in this section consist of 120,000 samples that were randomly taken from points in time no less than 15 minutes away from a flight that performs a GA. The GA corpus contains all 2,512 GA samples. For all figures presented in this section, the values corresponding to the GA flights are shown in red, values corresponding to "nominal" flights are shown in blue. The following discussion highlights operational factors that we found interesting, either because the data showed a significant difference between the nominal and GA sample distributions, or because there was remarkably little difference between the two distributions.

4.1. **State vector creation.** The datasets presented in section 2.1 were used to create a state vectors for the system. The fields of the state vectors are listed in Table 3. Most of the states are not directly available from the dataset and requires preprocessing and analysis. For instance, the number of heavy aircraft landing in the past 15 minutes required the processing of the entire database as well as a secondary database to match aircraft type (e.g. Boeing 747) and the landing category (e.g. heavy). The values of the fields were sampled for every minute in time during the four years of study. If a GA occurs within the one minute sample window we refer to the sample as a GA sample. Otherwise, the sample is referred to as a nominal sample. In the event a portion of the data associated with a particular sample of interest was absent, the sample was removed from the data set in order to ensure uniformity among dataset entries. The study focused on the periods of higher traffic density, that is 7:00 to 23:00 local time.
Table 3 presents the fields that were used as states to represent the system.

4.2. **Single parameter analysis.**

4.2.1. *Weather.* Weather is an important factor for aircraft operation as well as for airport runway configuration and operation. Table 4 presents the number of flights that landed during VMC and IMC as well as the number of GA occurring during each type of condition. The probability of a GA is increased by 71% during IMC versus VMC.

Figure 5 shows the distribution of nominal and GA flights as a function of various weather parameters. Figure 5(a) presents the frequency of the samples as a function of the visibility. A visibility of 10 nmi indicates that the actual visibility was at least 10 nmi. It appears the visibility at SFO is greater than 10 nmi approximatively 83% of the time, but only 75% of go arounds occur during these conditions. GAs appear to occur at a greater rate during low visibility conditions with 25% of GAs occurring in 17% of the time in which visibility is lower. Adverse weather conditions significantly increase the probability of GA, only 25 % of GA occur during poor weather conditions. Figure 5(b) presents the nominal and GA distributions as a function of headwind. A negative headwind corresponds to a tailwind. Most of the flights land with positive to no headwind, and the headwind does not appear to be a significant cause of GAs. In negative headwinds, that is tailwinds, GAs appear to be more frequent. The crosswind, is not depicted but does not seem to have an impact on GAs. Figure 5(c) presents the altitude of the sky's ceiling. All ceiling altitudes over 10,000 ft were trimmed to 10,000 ft. The data suggests a low ceiling is associated with an increase in the likelihood

TABLE 3. Sate vector description

Index	Fields Name: "Airspace data"
1	Time of the day
2-5	Number of ac, SFO inbound, current, average 5, 10, 15 min
6-8	Number of ac, SFO inbound, variation 5, 10, 15 min
9-12	Number of ac, SFO outbound, current, average 5, 10, 15 min
13-15	Number of ac, SFO outbound, variation 5, 10, 15 min
16-29	Number Same as 2-15 for OAK.
30-43	Number Same as 2-15 for SJC
44-46	Landing rate at SFO (ac/min) 5, 10, 15 min
47-49	Time elapsed to land the previous 4, 8, 12 ac at SFO
50-52	Departure rate at SFO (ac/min) 5, 10, 15 min
53-55	Time elapsed to takeoff the previous 4, 8, 12 ac
56-58	Landing rate at OAK (ac/min) 5, 10, 15 min
59-61	Time elapsed to land previous 4, 8, 12 ac at OAK
62-64	Departure rate at OAK 5, 10, 15
65-67	Time elapsed to takeoff the previous 4, 8, 12 ac at OAK
68-70	Landing rate at SJC (ac/min) 5, 10, 15 min
71-73	Time elapsed to land the previous 4, 8, 12 ac at SJC
74-76	Departure rate at SJC (ac/min) 5, 10, 15 min
77-79	Time elapsed to takeoff the previous 4, 8, 12 ac at SJC
80-88	Number of small, large, heavy, ac landing at SFO in past 5, 10, 15 min
89-97	Number of small, large, heavy, ac taking-off from SFO in past 5, 10, 15 min

Index	Fields Name: "Ground data"
98-101	Number of ac taxiing in, current, average 5, 10, 15 min
102-104	Number of ac taxiing in, variation 5, 10, 15 min
105-108	Number of ac taxiing out, current average 5, 10, 15 min
109-111	Number of ac taxiing out, variation 5, 10, 15 min
112-115	Number of ac in the runway queue, current, average 5, 10, 15 min
116-118	Number of ac in the runway queue, variation 5, 10, 15 min
119	Total estimated departure delay
119-124	Number of ac out delayed $>$ 0, 10, 20, 30, 45 min
125	Average delay by aircraft, out
126	Total estimated delay from schedule, arrivals
127-131	Number of ac delayed in $>$ 0, 10, 20, 30, 45 min
132	Average delay by aircraft, in

Index	Fields Name: "Weather data"
133	1 for visual MC and 0 for instrument MC
134-135	Ceiling, Visibility, Temperature
136-139	Wind Angle, windspeed, headwind, crosswind
140	Number of Runway(s) used for landing
141	Number of Runway(s) used for take offs

TABLE 4. Number of flights, GA and probability of GA during VMC and IMC

Conditions	All	Daytime	GA	$P(GA\|Conditions)$	GA & day	$P(GA\|Conditions\&\text{day})$
IMC	116,315	100,961	813	7.0×10^{-3}	752	7.4×10^{-3}
VMC	470,514	438,168	1,926	4.1×10^{-3}	1,860	4.2×10^{-3}
IMC	19.8%	18.7	29.7%	28.8%	-	-
VMC	80.2%	81.3	70.3%	71.2%	-	-

of a GA. Figure 5(d) shows the distributions as a function of temperature. The plots suggest GAs are more likely to occur at "higher" temperatures.

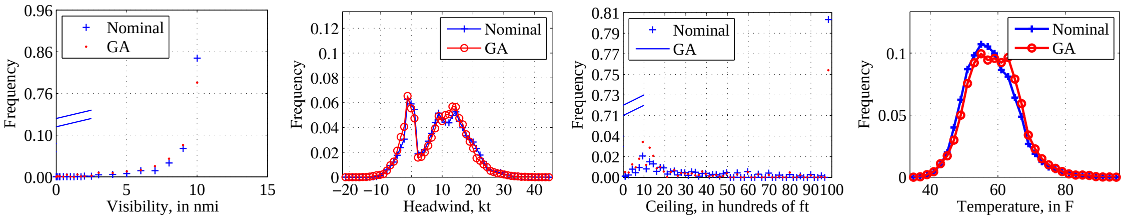


FIGURE 5. Analysis of the weather related parameters

4.2.2. *Time of day.* Figure 6 presents the sample distributions arranged by time of the day from 7:00 to 23:00 local time. The nominal distribution is not uniform on account of the runway configuration used and missing data; on some days, data for the morning was missing for unknown reasons. It appears that there are two peaks where GAs occur more frequently: from 9:00 to 14:00 and then again from 19:00 to 21:00. In the interim, the frequency of GAs achieves a minimum near 15:30.

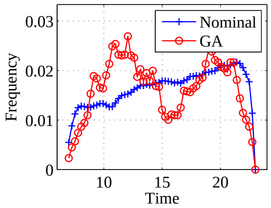


FIGURE 6. Distribution of Time for the nominal observations and the GA

4.2.3. *Number of aircraft inbound for SFO.* Figure 7 shows the distribution corresponding to the number of airborne flights inbound for SFO present in the terminal airspace. Figure 7(a) shows the number of aircraft at the time the sample is taken, Figure 7(b) shows the average number of aircraft during the 15 minutes preceding the sample. Intuitively, these statistics capture a measure of an air traffic controller's current and recent activity level, respectively. The GA and nominal distributions are similar, but there is a visible shift in the mean; the mean of the GA distribution is approximatively 3 aircraft larger than that of the nominal distribution. Figures 7(c) and 7(d) present the distributions for nominal and GA flights as a function of the difference between the number of aircraft in the system at present and the number of aircraft 5 and 15 minutes ago, respectively. The 5 minute variation does not illustrate a significant difference between distributions. The 15 minute distributions suggests that the distribution of GAs is shifted slightly to the right relative to the associated nominal distribution.

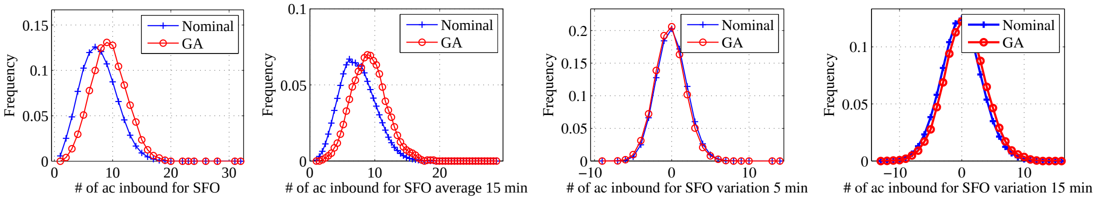


FIGURE 7. Analysis of the number of aircraft incoming to SFO

4.2.4. *Landing/takeoff rates at SFO and aircraft types.* Figure 8 presents the landing rates (number of aircraft landing per minute) over the past 5 and 15 minutes (Figures 8(a) and 8(b)), as well as the number of heavy and large aircraft landing over the past 5 minutes (Figures 8(c) and 8(d)), respectively. It appears that a higher landing rate increases the likelihood of a GA, but not in a very important manner. Moreover, it appears that higher numbers of large and heavy aircraft also tend to increase the likelihood of a GA occurring. Although omitted here, the number of small aircraft landing in the past 5, 10 and 15 minutes displayed no significant difference between nominal and GA distributions. Also omitted, for similar reasons, are the distributions corresponding to the takeoff rate at SFO.

4.2.5. *Number of aircraft outbound from SFO.* Figure 9 depicts the distributions associated with the number of aircraft outbound from SFO, and the variation over 5 minutes. There is no significant difference between the data corresponding to GAs and nominal flights. We omit the associated average and variation plots for 5, 10, and 15 minutes as there are no significant differences between the nominal and GA distributions. The outbound traffic does not appear to have a statistical impact on the occurrence of GAs.

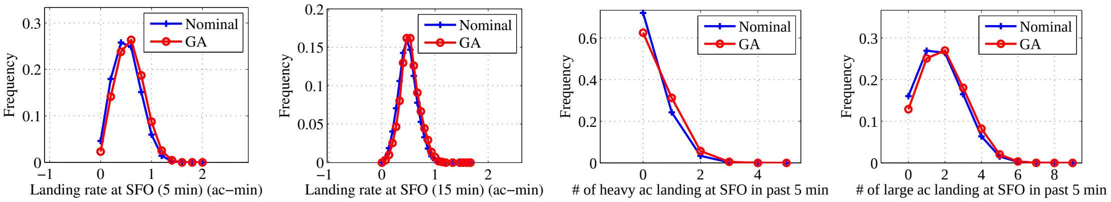


FIGURE 8. Analysis landing/takeoff rates at SFO and aircraft types

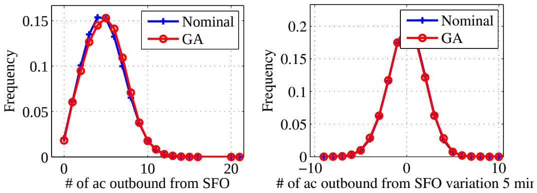


FIGURE 9. Analysis of the number of aircraft outbound from SFO

4.2.6. *Number of aircraft inbound for OAK - Landing rate.* Figure 10 presents the distribution of the number of flights present in the terminal airspace (in the air) and inbound for OAK. Figure 10(a) shows the number of aircraft at the time the sample is taken. Figure 10(b) shows the average number of aircraft during the 15 minutes preceding the time of the sample. These measures reflect the current activity of the controllers and their workload over the past 15 minutes. The nominal and GA distributions do not differ significantly, meaning the number of aircraft inbound for OAK does not appear to have a statistical impact on the occurrence of GAs at SFO. Figures 10(c) and 10(d) present the difference between the number of aircraft simultaneously present at the time of the sample and the number in the system 5 and 15 minutes in the past, respectively. The GA distribution is shifted slightly to the right of the nominal distribution, suggesting GAs occur more frequently when there is an increase in the number of aircraft inbound for OAK during the preceding minutes. Note that the plots are not centered at 0, suggesting a correlation between the runway configuration used at SFO and changes in the traffic volume inbound for OAK. The landing rates at OAK over the preceding 5, 10 and 15 minutes are not presented, since they do not imply any significant statistical impact on GAs at SFO.

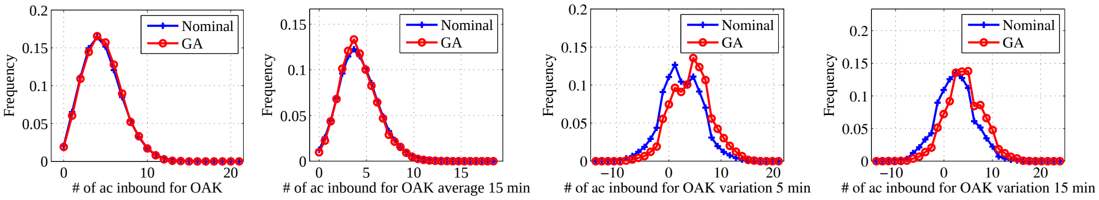


FIGURE 10. Analysis of the number of aircraft incoming to OAK

4.2.7. *Number of aircraft outbound from OAK - Takeoff rate.* Distributions in the number of aircraft outbound from OAK, the variation in the number of aircraft outbound as well as the takeoff rates are not presented; they present no significant difference between the data corresponding to the GA and nominal samples.

4.2.8. *Number of aircraft inbound to/outbound from SJC - Landing/takeoff rates.* In terms of instantaneous or average number of aircraft, the distributions associated with the number of aircraft inbound to and outbound from SJC do not show significant differences. Figure 11 presents the distribution of the difference in the current number of inbound aircraft and the number of inbound aircraft 5 and 15 minutes ago. It appears an increase in the number of aircraft inbound for SJC tends to indicate an increase in the frequency of GAs at SFO. Note that the plots are not centered at 0, suggesting a correlation between the runway configuration used at SFO and the changes in traffic volume inbound for SJC. The landing and takeoff rates at SJC over 5, 10 and 15 minutes are not presented; they do not appear to have a statistical impact on GAs at SFO.

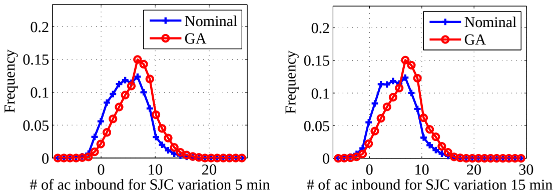


FIGURE 11. Analysis of the changes in number of aircraft incoming to SJC

4.2.9. *Number of aircraft on the airport surface, inbound.* Figure 12 presents the distributions associated with the number of inbound aircraft taxiing at SFO. Figure 12(a) shows the number of aircraft at the time the sample is taken and Figure 12(b) shows the average number of aircraft over the preceding 15 minutes. These measures reflect the current congestion at the airport for aircraft taxiing-in. The shape of nominal and GA distributions are very similar, with the GA distributions being slightly skewed toward higher aircraft counts. Figures 12(c) and 12(d) present distributions for the difference between the number of aircraft simultaneously present at the time of sample and 5 and 15 minutes ago, respectively. The plots would suggest that a high number of incoming aircraft slightly increases the probability of having a GA, but some GAs occur when there are only a few incoming aircraft. The 5 minute variation in the number of inbound aircraft is slightly shifted toward the negative numbers for the GAs, meaning that GAs are more likely to occur when the number of aircraft inbound on the surface diminishes. This effect is not visible in the case of 15 minute variations.

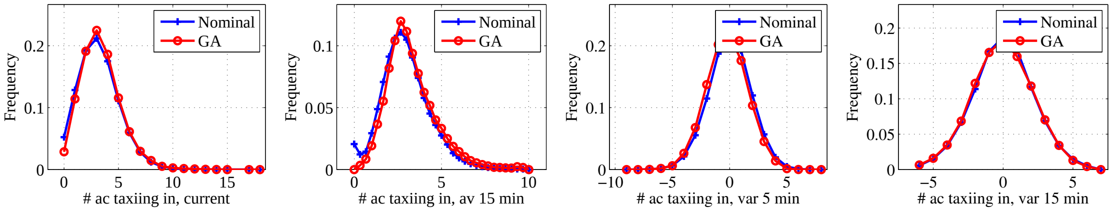


FIGURE 12. Analysis of the number of aircraft taxiing at SFO, inbound

4.2.10. *Number of aircraft on the airport surface, outbound.* Figure 13 presents the distributions of the number of inbound aircraft taxiing at SFO. Figure 13(a) shows the number of aircraft at the time the sample is taken and Figure 13(b) shows the average number of aircraft over the preceding 15 minutes. These measures reflect the current congestion at the airport, for aircraft taxiing-in. The shape of the distribution of the GAs and the nominal samples are very similar; there are slightly more GAs at the higher aircraft counts. It appears that a high number of outbound aircraft has an impact on the probability of having a GA, but some GAs occur when there are only a few incoming aircraft. The 15 minutes plot suggests that having an average of more than 10 aircraft taxiing out has a significant impact on GAs.

The difference between the number of aircraft simultaneously present at the time of the sample and 5, 10 and 15 minutes in the past are not depicted, since they do not show any particularly interesting results.

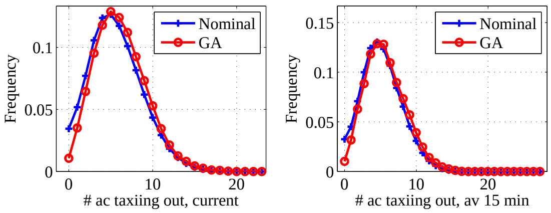


FIGURE 13. Analysis of the number of aircraft taxiing at SFO, outbound

4.2.11. *Delays.* Figure 14 shows the distributions as a function of the number of aircraft delayed. Figure 14(a) shows the distribution of the number of inbound aircraft with a delay, taxiing into the gate. Figure 14(b), presents the distribution for delays greater than 20 minutes. It appears that GAs are less likely to occur when there are no delayed aircraft taxiing to a gate. For delays greater than 20 minutes, although 40% of the time there are no aircraft delayed to this extent, 33% of the GAs occur under these conditions, indicating large delays may contribute to a GA.

Figures 14(c) and 14(d) present the same distributions for the case of aircraft taxiing out. While 25% of the time there are no more than two aircraft with a delay, 15% of the GAs occur under these conditions.

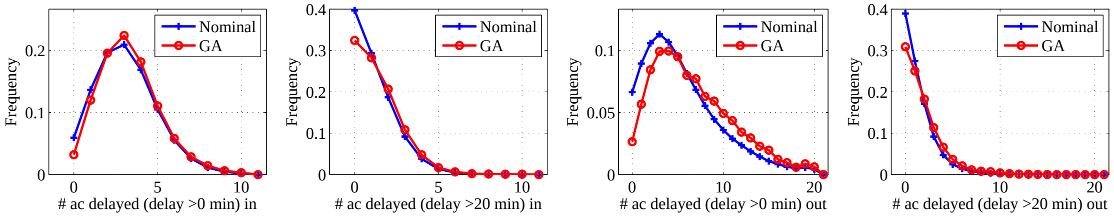


FIGURE 14. Analysis of the number of aircraft delayed at SFO

4.3. **Discussion of results.** From this analysis, three main factors leading to an increased probability of a GA are the weather, the airborne traffic density and aircraft mix, and finally, the ground traffic and its delays.

4.3.1. *Weather:* When the visibility or the ceiling are low, the rate of GAs is much higher than in good weather conditions. A likely explanation is the lack of visibility at decision hight forcing the pilot to initiate a missed approach and return. Wind, including tailwind and crosswind do not appear to have a significant impact on the probability of GAs to occur. The weather has a direct impact on GAs, but at SFO, the number of GAs due to poor weather conditions is only 25%. The temperature appears to have a slight impact on the GA, meaning that GA are more likely to occur during warm days, or during summer. Note that this seasonality factor that does appear in the time distribution of Figure 4

4.3.2. *Traffic density and aircraft mix:* The analysis suggests that having a large number of incoming aircraft increases the probability of having a GA. From a human factor's perspective, a large number of aircraft simultaneously present in the terminal airspace increases the workload of the controllers, probably leading to more "operational errors" and violation of minimum separation at distances. The terminal airspace is rather small and congested, therefore dealing with many aircraft becomes complex very quickly and leads air traffic controllers to vector and reroute aircraft [8]. In a previous study [7], it was shown that limiting the number of aircraft simultaneously present in the TRACON tends to allow for more direct routes, hence reducing the perceived complexity, and eventually, maybe reducing the probability of a GA. However, there are some GAs that occur when there are only a few incoming aircraft, perhaps a testament to the sporadic and haphazard nature of the event.

The aircraft mix appears to have an effect on the likelihood of GA. A high number of large and heavy aircraft landing in the past 5 to 15 minutes increases the probability of a GA. Possible explanations include the separation distance between aircraft becoming too small.

There is also evidence to suggest the variation in the number of planes being metered to OAK has an impact on the likelihood of a GA; a positive change in the number of aircraft incoming into OAK seems to increase the probability of a GA occurring at SFO. A possible explanation is a shift in cognitive perception for the controllers in charge of the sequencing and merging for SFO, OAK and SJC. Since most of the traffic in the TRACON is directed to SFO, a sudden increase in the number of aircraft inbound for OAK or SJC requires a shift of attention from the controller, probably breaking his current mental model [12] of the situation. This analysis shows the coupling effect between the airport, not only because of the traffic that needs to be separated, but also from a controller's point of view. This correlation between GA at SFO and variation in incoming traffic for OAK is really unexpected since SFO and OAK operations are supposedly decoupled, different controllers for each airport.

4.3.3. *Ground traffic and delays.* It appears that a large number of aircraft taxiing out at SFO increases the probability of a GA occurring. In addition, an average over 15 minutes of more than 10 aircraft taxiing out has a visible impact. Note also that delays affecting either inbound or outbound aircraft increase the probability of GAs. It appears that human errors such as runway incursion, holding lines violation or late takeoff from runway are more likely to occur during high density outbound ground traffic and when delays affect ground traffic.

5. An Alerting System for Go-Arounds

In this section, we present a system to evaluate the potential of a GA. The first step is to classify GAs from nominal samples using the available data. The second step is to evaluate the potential of a GA at each time step. We first introduce the issues related to predicting these rare and poorly separated events, before presenting the results of our classification and prediction results based on the method of linear discriminant analysis.

5.1. **Classification and Prediction issues.** There are a number of factors that make classification and temporal prediction of GAs difficult. First, GAs are rare events. Although this is auspicious form an air transportation perspective, it makes learning difficult and introduces a strong statistical bias in our corpus of training examples. To improve learning, it is natural to use a modified training set with roughly an equal number of nominal and GA samples. This can be accomplished by either withholding a large portion of nominal examples from the corpus, up-sampling the GAs, or a combination of the two methods. One of the main issues with this dataset is that nominal and GA samples are very poorly separated in the sample space (as presented in section 4). As false positives (nominal samples that are labelled as GA samples) are especially undesirable in our context, we must seek to learn relationships that separate the training data. Because of the poor separability of the samples, it is not possible to predict GAs and maintain a low rate of false positives. Therefore, we aim at evaluating "high risk" time samples which have a higher probability of having a GA. These high risk time samples will be denoted by an "alert level". The objective is to maximize the number of GAs positively identified during the alert level while minimizing the total number of samples in the alert level.

5.2. **Linear discriminant analysis.** Linear discriminant analysis is a statistical method commonly used to separate samples into several classes [1]. In our case, we are concerned with only two classes of samples: nominal state vector samples with label $y = 0$ and GA state samples with label $y = 1$. We will assume GAs and nominal samples are generated according to a two-label Gaussian mixture model. For this purpose, our corpus of samples, $\{x_i\}_{i=1}^N$, $x_i \in \mathbb{R}^{135}$ is split into two groups, a training group and a test group. A strong assumption made by LDA is that the conditional probability density functions $p(x|y = 0)$ and $p(x|y = 1)$ are both normally distributed with mean and covariance (μ_0, Σ_0) and (μ_1, Σ_1), respectively. Then, a feature vector x is assigned the label 0 if it satisfies

$$(1) \qquad (x-\mu_0)^T \Sigma_{y=0}^{-1} (x-\mu_0) + \ln|\Sigma_{y=0}| - (x-\mu_1)^T \Sigma_{y=1}^{-1} (x-\mu_1) - \ln|\Sigma_{y=1}| < T,$$

where T is a threshold value that reflects the frequency of a label and the $|\Sigma|$ denotes the determinant of Σ. Otherwise, x is assigned the label 1. By sweeping T downward from a very large number to zero, we can progressively decrease the rate at which 0 labels are assigned. That is, we can control the number of GAs that we are able to identify correctly. However, correctly identifying the bulk of the GAs comes at the expense of having a large number of false positives.

5.3. **An alert system for GA.** We used LDA on three years of data to classify GA and nominal samples. A separate year of data was withheld to test the classifier's predicative capabilities. The training set consists of randomly selected nominal samples and all the GA from 2006 to 2008. The test set contains all the available samples from 2009. Figure 15 presents the result of the classification and prediction. The green curve (top) is corresponds to the training set and the blue curve (bottom) to the test set. The horizontal-axis represents the fraction of time predicted as alert-level, that is the fraction of the time where GAs are likely to occur. The y-axis represents the proportion of actual GAs that occurred during a period of alert-level. The crosses on the green line correspond to the different values of the threshold T (Eq. 1). By fixing a value of T, one can choose the number of samples residing in the alert-level. The dashed black line indicates the increased probability of a GA occurring during an alert-level as compared to the remainder of the time. The line 1x indicates complete randomness of the prediction. When the green line is over the 4x line, a GA is 4 times more likely to occur as compared to the remainder of time, that is the time when the alert is off. For instance, the green line intercepts the *9x* line at 15% of the time in threat level. This means that the predictor can be in "threat level" 15% of the time and capture 39% of the GAs. During alert-level time, GAs are 9 times more likely to occur than during the remaining 85% of the time.

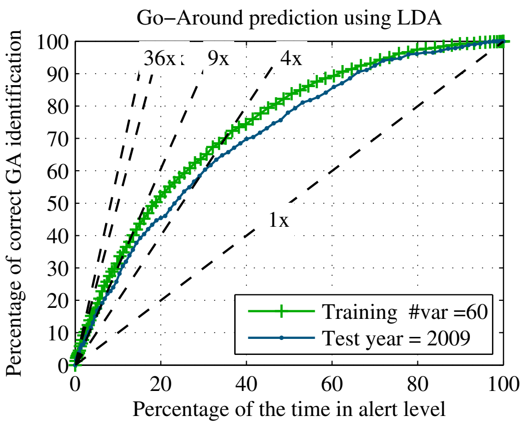


FIGURE 15. Training and test results for classification of GA using LDA

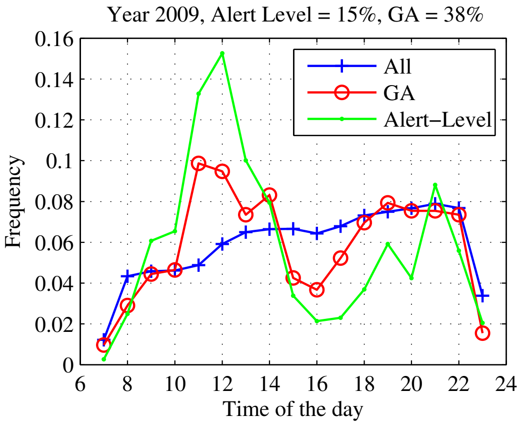


FIGURE 16. Time distribution of nominal samples, GA and alert-level

To further analyze the results of the predictive system, we looked at the distribution over time of the samples predicted in to be in alert levels, that is we ran our alert system on 2009 data, at a given design point. The design parameter, T, was selected so that, during training, 15% of the samples are in threat level, resulting in the capture of 39% of the GAs. Figure 16 presents the time distribution of the nominal samples, GA and samples identified in alert-level. The alert-level curve follows the same pattern as the GA curve over time. The predictive system slightly over-estimate the risk during the morning peak and under-estimates the risks during the evening peak.

5.4. **Other methods.** To improve the classification results, we tried different methods. We also split the dataset in a different manner. Instead of using 2006 to 2008 data for training and 2009 for test, we randomly picked 80% of the data for training and the remaining 20% for test. This explains the difference between Figures 15 and 17 for the results on the initial dataset.

To overcome the problem of the imbalance between the datasets, we then tried to over-sample the GA dataset using using the Synthetic Minority Over-sampling Technique (SMOTE) [3]. SMOTE creates synthetic examples for the under-represented class (GAs in our case). When increasing the number of GA samples to 120,000 (the number of nominal samples). Figure 17 presents the results of the training using SMOTE in grey. The test results are presented in pink and in green. The pink curve contains synthetic test data while the green only contains real data. The large gap between training and test curves indicates that the classifier over-fitted the data during training and performed poorly during testing.

We then tried to use a ℓ_1 logistic regression **??** for attribute selection and prediction. As shown in figure 17, the performance does not improve. The results are very sensitive to the value chosen for the λ parameter.

6. CONCLUSION

This paper investigated a number of airport operational features, each of which is readily accessible to on-duty air traffic controllers, and they fluctuate in the time period preceding a missed approach. We showed the important interconnection between surface operations, airborne operations, airports located in the vicinity of each other, air traffic control procedures, and system delays. By analyzing how the distribution

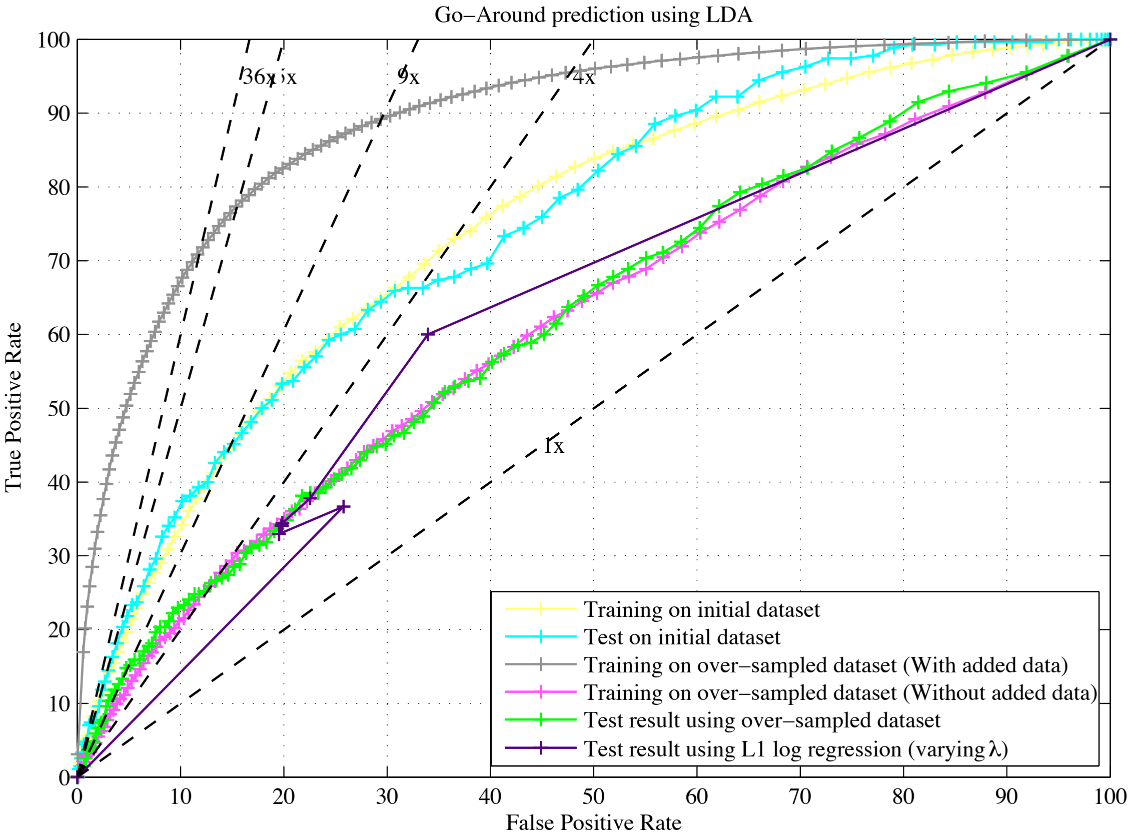


FIGURE 17. Comparison of classification and prediction result for different classifiers

of these features varied between nominal and go-around operations, we provided a statistical mechanism to gain insight into which factors are more likely to be a discernible precursor to go-arounds. Interpretations for these results were provided in terms of the current operational policies in place at busy metropolitan airports. Armed with the new insight afforded by these statistics, we proposed a framework for developing an automated alert system to identify systems in which there is a high potential of having a missed approach. Unfortunately, the machine learning techniques employed to this end would mandate the alert system be "on" for an exorbitantly large amount of time to capture most of the go-arounds. It appears that the prediction of go-arounds is a very challenging task and this study has highlighted unexpected factors that have an impact on the probability of a missed approach. For example, factors such as the airport coupling effect on controllers needs to be accounted for in the design of high-density metroplex operations.

ACKNOWLEDGMENT

This research was sponsored by NASA under NRA NNX08AY52A. The authors would like to thank Greg Kingery from SFO control tower for his valuable insight.

REFERENCES

[1] C. Bishop. *Pattern Recognition and Machine Learning. 2007*. Springer.

[2] L. Boursier, E. Hoffman, L. Rognin, F. Vergne, , and K. Zeghal. Airborne Spacing in the Terminal Area: A study of non-nominal situations. 2006.

[3] N. Chawla, K. Bowyer, L. Hall, and W. Kegelmeyer. SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16(1):321–357, 2002.

[4] DashLink. Flight tracks, northern california tracon. `https://c3.ndc.nasa.gov/dashlink/resources/132/`.

[5] Federal Aviation Administration. Aviation system performance metrics. `http://aspm.faa.gov/`.

[6] Federal Aviation Administration. *Aeronautical Information Manual*, 2009.

[7] M. Gariel, J.-P. Clarke, and E. Feron. A dynamic I/O model for TRACON traffic management. In *AIAA Guidance, Navigation, and Control Conference and Exhibit, Hilton Head, SC*, 2007.

[8] M. Gariel, A. N. Srivastava, and E. Feron. Trajectory clustering and an application to airspace monitoring. *Accepted to IEEE Transactions on Intelligent Transportation*, 2011.

[9] Joint Planning and Development Office. Nextgen avionics roadmap, v1.0, 2008.

[10] Joint Planning and Development Office. Atm-weather integration plan, v2.0, 2010.

[11] J. Planning and D. Offifice. Concept of operations for the next generation air transportation system, version 3.2.

[12] T. Reynolds, J. Histon, H. Davison, and R. Hansman. Structure, Intent & Conformance Monitoring in ATC. *ATM Workshop*, 22-26 September 2002, Capri, Italy.

[13] W. Robertson. Fuel conservation strategies: Descent and approach. *Aero Magazine*, 2010.

[14] C. Thiel and H. Fricke. Collision risk on final approach - a radar data based evaluation method to assess safety. *Proceedings of 4th International Conference on Research In Air Transportation*, 2010.

[15] N. C. TRACON. Go-arounds statistics, Sept 2010 to March 2011.

[16] US Department of Transportation and Federal Aviation Administration. Introduction to TCAS II, version 7, 2000.

SMOOTHED QUANTILE REGRESSION FOR STATISTICAL DOWNSCALING OF EXTREME EVENTS IN CLIMATE MODELING

ZUBIN ABRAHAM*, FAN XIN**, AND PANG-NING TAN*

Abstract. Statistical downscaling is commonly used in climate modeling to obtain high-resolution spatial projections of future climate scenarios from the coarse-resolution outputs projected by global climate models. Unfortunately, most of the statistical downscaling approaches using standard regression methods tend to emphasize projecting the conditional mean of the data while paying scant attention to the extreme values that are rare in occurrence yet critical for climate impact assessment and adaptation studies. This paper presents a statistical downscaling framework that focuses on the accurate projection of future extreme values by estimating directly the conditional quantiles of the response variable. We also extend the proposed framework to a semi-supervised learning setting and demonstrate its efficacy in terms of inferring the magnitude, frequency, and timing of climate extreme events. The proposed approach outperformed baseline statistical downscaling approaches in 85% of the 37 stations evaluated, in terms of the magnitude projected for extreme data points.

1. Introduction

An integral part of climate modeling is downscaling, which seeks to project future scenarios of the local climate based on the coarse resolution outputs produced by global climate models (GCMs). Two of the more common approaches to downscaling are dynamic downscaling and statistical downscaling. Dynamic downscaling uses a numerical meteorological model to simulate the physical dynamics of the local climate while utilizing the climate projections from GCMs as initial boundary conditions. Though it captures the geographic details of a region unresolved by GCMs, the simulation is computationally demanding while its spatial resolution remains too coarse for many climate impact assessment studies. Statistical downscaling establishes the mathematical relationship between the coarse-scale GCM outputs and the fine-scale local climate variables based on observation data. Unlike dynamic downscaling, it is flexible enough to incorporate any predictor variable and is relatively inexpensive. Most of the statistical downscaling approaches employ regression methods such as multiple linear regression, ridge regression, and neural networks to estimate the conditional mean of the future climate conditions. These methods are ill-suited for predicting extreme values of the climate variables.

An alternative approach is to use techniques such as quantile regression, which aims to minimize an asymmetrically weighted sum of absolute errors, to estimate the particular quantile that corresponds to extreme values [26]. Unfortunately, quantile regression tends to overestimate the response variable resulting in a large number of data points being falsely predicted to be extreme. Figure 1 represent the histogram of the distribution of observed temperature at a weather station in Canada. The lines represent the distribution of the predicted values for temperature obtained using multiple linear regression (MLR) and quantile regression. An observation is considered an extreme data point if its response variable is in the top 5 percentile of observations. The shape of the tail of the distribution that represents extreme data points (observed and projected) is shown in Figure 2. It is clear from the figures that methods such as multiple linear regression (green line) that estimate the conditional mean tend to underestimate the tail of observed probability distribution, while quantile linear regression (red line) overestimates the tail part of the probability distribution. As elaborated

*Michigan State University, Dept of Computer Science, abraha84@msu.edu, ptan@cse.msu.edu
**Michigan State University, Dept of Statistic, fanxin@msu.edu.

in Section 5, it was found that for the 37 stations evaluated, at an average, quantile regression predicted a datapoint to be an extreme point more than twice as frequently as the actual frequency of observed extreme data points.

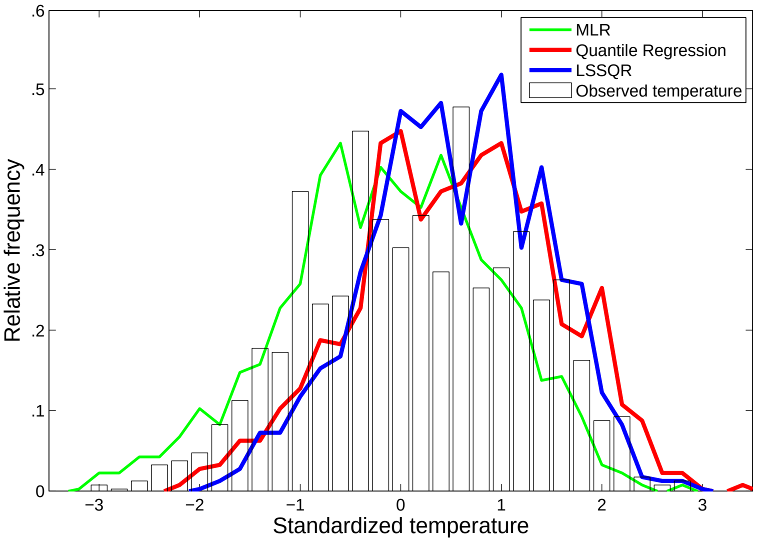


Figure 1. Histogram of observed temperature.

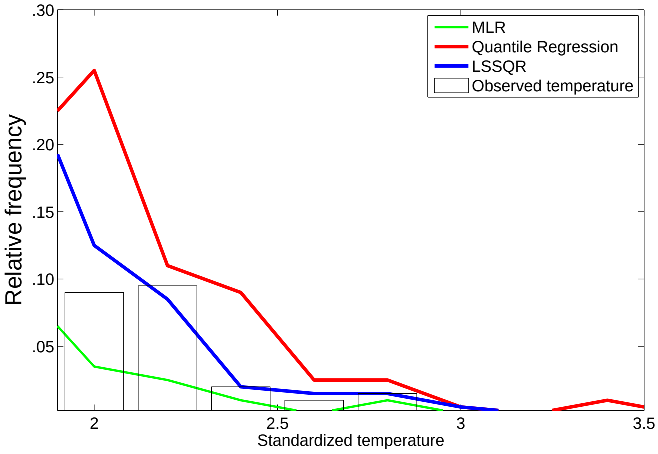


Figure 2. Tail of the histogram.

To address this overestimation, we propose a method known as smoothed quantile regression (LSQR) that reduces the absolute error of extreme data points by introducing a smoothing term that brings the predicted response value of extreme points closer to the value corresponding to the percentile of extreme data points. This smoothing term also provides a means to easily extend the objective function to a semi-supervised learning setting (LSSQR). Semi-supervised learning, in addition to using the training data, can also use the distribution characteristics of the predictor variables of the test set to glean a better estimate of the distribution of data upon which the model will be applied.

In summary, the main contributions of this paper are as follows:

- We demonstrate the limitation of MLR, ridge regression and quantile regression in predicting extreme values.
- We present a smoothed quantile regression framework for extreme values prediction.
- We also extend the framework to a semi-supervised setting.
- We demonstrate the efficacy of our learning framework on climate data (temperature) obtained from the Canadian Climate Change Scenarios Network website [1]. Both the supervised and the semi-supervised proposed frameworks outperformed the baseline methods in 85% of the 37 stations evaluated, in terms of magnitude, frequency and the timing of the extreme events.

The remainder of this paper is organized as follows. Section 2 covers some of the related work. Section 3 introduces the reader to the notations and terminology used in the paper. Relevant approaches, such as quantile regression are also introduced. Section 4 introduces the objective function of the proposed supervised and semi-supervised model, as well as the analysis of the model. This is followed by a detailed description of our algorithm and experimental results in Section 5. Finally, we present our conclusions and suggestions for future work in Section 6.

2. Related Work

Time series prediction has long been an active area of research with applications in finance [40], climate modeling [19][12], network monitoring [10], transportation planning [24], etc. There are several time series prediction techniques available, including least square regression [27], recurrent neural networks [23], Hidden Markov Model Regression [22], and support vector regression [33].

Given the growth in the number of climate models in the earth science domain, extensive research has been done to best utilize these models [31] as well as focus on downscaling surface climate variables like temperature and precipitation time series from these global climate models (GCM) [12, 13, 19, 39]. Identifying and modeling extreme events in climatology has recently gained a lot of traction [7]. Unfortunately, the common regression techniques mentioned earlier that may be used for downscaling, focus on predicting the conditional mean of the response variable, while extreme values are better identified by conditional quantiles that corresponds to the extreme values. Hence, unlike the common regression techniques mentioned earlier that focus on predicting the conditional mean, the motivation behind the presented model is focusing on the conditional quantile, using an approach similar to quantile regression [26].

Variations of quantile regression such as non-parametric quantile regression and quantile regression forests have been used to infer the conditional distribution of the response variable which may be used to build prediction intervals [34, 30]. Also, variants of quantile regression that estimate the median are used due to its robustness to outliers when compared to traditional mean estimate [41]. [21] presented a statistical downscaling approach to estimate censored conditional quantiles of precipitation that uses QR. The conditional probability of the censored variable is estimated using a generalized linear model (GLM) with a logit function to model the nature of the distribution of precipitation and hence cannot be directly applied to model temperature. Mannshardt-Shamseldin et. al. [28] demonstrate another approach to downscaling extremes through the development of a family of regression relationships between the 100 year return value (extremes) of climate modeled precipitation(NCEP and CCSM) and station-observed precipitation values. Generalized extreme value theory based approaches have also be applied to model extreme events like hydrologic and water quality extremes, precipitation, etc [36, 6]. The Pareto distribution [47, 48], Gumbel [49, 50] and Weibull [51] are the more common variants of General extreme value distribution used. But these techniques are probabilistic based that emphasize trends pertaining to the distribution of future extreme events and not the deterministic timing of the occurrence of the extreme event.

The drawback of building a model that primarily focuses on only a particular section of the conditional distribution of the response variable is the limited amount of available data. Hence, the motivation for incorporating unlabeled data during model building. There have been extensive studies on the effect of incorporating unlabeled data to supervised classification problems, including those based on generative models[18], transductive SVM [25], co-training [8], self-training [44] and graph-based methods [5][45]. Some studies concluded that significant improvements in classification performance can be achieved when unlabeled examples are used, while others have indicated otherwise [8, 15, 17, 35, 42]. Blum and Mitchell [8] and Cozman et al. [15] suggested that unlabeled data can help to reduce variance of the estimator as long as the modeling assumptions match the ground truth data. Otherwise, unlabeled data may either improve or degrade the classification performance, depending on the complexity of the classifier compared to the training set size [17]. Tian et al. [35] showed the ill effects of using different distributions of labeled and unlabeled data on semi-supervised learning.

3. Preliminaries

Let $D_l = \{(x_i, y_i)\}_{i=1}^{n}$ be a labeled dataset of size n, where each $x_i \in \mathcal{R}^d$ is a vector of predictor variables and $y_i \in \mathcal{R}$ the corresponding response variable. Similarly, $D_u = \{(x_i, y_i)\}_{i=n+1}^{n+m}$ corresponds to the unlabeled dataset. The objective of regression is to learn a target function $f(x, \beta)$ that best estimates the response variable y. β is the parameter vector of the target function. n represents the number of labeled training points and m represents the number of unlabeled testing points.

3.1. Multiple linear regression (MLR) and ridge regression.

One of most widely used forms of regression is multiple linear regression. It solves a linear model of the form

$$y = x^T \beta + \boldsymbol{\epsilon}$$

where, $\boldsymbol{\epsilon} \sim N(0, \sigma^2)$ is an i.i.d Gaussian error term with variance σ^2. $\boldsymbol{\beta} \in \mathcal{R}^d$ is the parameter vector. MLR minimizes the sum of squared residuals

$$(y - X\beta)^T (y - X\beta)$$

which leads to a closed-form expression for the solution

$$\hat{\boldsymbol{\beta}} = (X^T X)^{-1} X^T y$$

A variant of MLR, called ridge regression or Tikhonov regularization is often used to mitigate overfitting. Ridge regression also provides a formulation to overcome the hurdle of a singular covariance matrix $X^T X$ that MLR might be faced with during optimization. Unlike the loss function of MLR the loss function for ridge regression is

$$(y - X\beta)^T (y - X\beta) + \lambda \beta^T \beta,$$

and its corresponding closed-form expression for the solution is

$$\hat{\boldsymbol{\beta}} = (X^T X + \lambda I)^{-1} X^T y$$

where, the ridge coefficient $\lambda > 0$ results in a non-singular matrix $X^T X + \lambda I$ always being invertible. The problem with both MLR and ridge regression is that they try to model the conditional mean, which is not best suited for predicting extremes.

3.2. Quantile Linear Regression(QR).

The τ^{th} quantile of a random variable Y is given by:

$$Q_Y(\tau) = F^{-1}(\tau) = \inf\{y : F_Y(y) \geq \tau\}$$

where,

$$F_Y(y) = P(Y \leq y)$$

is the distribution function of a real valued random variable Y and $\tau \in [0, 1]$.

Unlike MLR that estimates the conditional mean, quantile regression estimates the quantile (e.g., median) of Y.To estimate the τ^{th} conditional quantile $Q_{Y|X}(\tau)$, quantile regression minimizes an asymmetrically weighted sum of absolute errors. To be more specific, the loss function for quantile linear regression is:

$$\sum_{i=1}^{N} \rho_\tau(y_i - x_i^T \beta)$$

where,

$$\rho_\tau(u) = \begin{cases} \tau u & u > 0 \\ (\tau - 1)u & u \le 0 \end{cases}$$

Unlike MLR and ridge regression that have a closed-formed solution, quantile regression is often solved using optimization methods such as linear programming. Linear programming is used to solve the loss function by converting the problem to the following form.

$$\min_{u,v} \quad \tau 1_n^T u + (1-\tau) 1_n^T v$$

$$\text{s.t.} \quad y - x^T \beta = u - v$$

where, $u_i \ge 0$ and $v_i \ge 0$. But as shown in Figures 1 and 2, quantile regression often overestimates data points resulting in too many false positive extreme events predicted.

4. Framework for smoothed quantile regression

Given that the primary objective of the model is to accurately regress extreme valued data points and quantile regression has been shown to perform relatively better that its least square counterparts that tend to underestimate the frequency and magnitude of extreme data points, the proposed objective approach of the proposed frameworks is modeled around linear quantile regression. Section 4.1 describes smoothed quantile regression (LSQR) and its objective function. Section 4.2 proposes a semi-supervised extension to LSQR which is then followed by mathematical properties of the behavior of the objective function.

4.1. Smoothed quantile regression (LSQR).

We propose a quantile-based linear regression model that is based on the assumption of smoothness, i.e., data points whose predictor variables are similar, should have a similar response. We use this notion of smoothness as an integral part of the framework as experiments provided in Section 5 demonstrate this characteristic in the dataset used. The smoothness assumption could be described as the constraint

$$\sum_{i,j}^{n} w_{ij}(f_i - f_j)^2 < c$$

where w_{ij} is a measure of similarity between data point i and j, f the predicted value of the response variable and c is a constant.

Also, since the framework doesn't restrict the training set only to extreme data points, the smoothing component of the objective function tends to implicitly cluster data points resulting in better distinction of the response variables of an extreme valued data point and a non-extreme valued data point. Empirical results comparing supervised quantile regression to the proposed semi-supervised model illustrate this point as shown in Section 5. The term

$$w_{ij} = \exp(-\frac{||x_i - x_j||^2}{\sigma}) \quad i, j \in [1, 2, \ldots, n]$$

is equivalent to the radial basis function and is used to capture the similarity between the predictor variables of data point i and data point j. σ is a scale parameter used to control the distance above which two data points are not considered as being highly coupled.

Assuming linear regression, $f(x_i, \beta) = x_i\beta$, the smoothing term can be reformulated as

$$\sum_{i,j}^{n} w_{ij}(f(x_i, \beta) - f(x_j, \beta))^2 = f^T \mathbf{\Delta} f = \beta^T \mathbf{\Sigma} \beta$$

where,

$$\mathbf{\Sigma} = X^T \mathbf{\Delta} X$$

$$\mathbf{\Delta} = D - W$$

and D is a diagonal matrix such that $D_{ii} = \sum_{j=1}^{n} w_{ij}$ and $W = \{w_{ij}\}|_{i,j=1}^{n}$.

Coupling smoothing with the objective function of linear qunatile regression, we end up with the following optimization problem.

$$\min_{\beta} \sum_{i=1}^{n} \rho_\tau(y_i - x_i^T \beta) + \lambda \beta^T \mathbf{\Sigma} \beta$$

As can be clearly observed from the objective functions of $LSQR$, $\lambda \to 0$ results in an estimate similar to quantile linear regression while, $\lambda \to \infty$ results in the estimate of the response variable converging towards the target quantile of data. This is because a large λ would penalize any non-zero difference between f_i and f_j very harshly thereby minimizing the error by setting $f_i = \alpha, \forall i \in [1, 2, \ldots, n]$, thereby reducing the error from the second component of the equation to 0. This reduces the loss function to the following

$$f(\beta) = \sum_{i=1}^{n} \rho_\tau(y_i - \alpha), \quad \beta = (\alpha, 0, 0, \ldots, 0)^T$$

The formal proof of this is provided in the following theorem.

Theorem 1: $f(x_i, \beta) \to y_{(n\tau)}$ as $\lambda \to \infty$, $\forall i \in [1, 2, \ldots, n]$.

Proof : Let $y_{(i)}$ be the i^{th} smallest element among $y_k|_{k=1}^{n}$ and $y_{(i)} < \alpha_i <= y_{(i+1)}$. When $\lambda \to \infty$, the loss function can be rewritten in terms of α_i as follows

$$\sum_{k=1}^{i}(1-\tau)(\alpha_i - y_{(k)}) + \sum_{k=i+1}^{n} \tau(y_{(k)} - \alpha_i) + \sum_{i,j=1}^{n} W_{ij}(\alpha_i - \alpha_i)$$

which is equivalent to minimizing

$$\tau \sum_{k=1}^{n} y_{(k)} - \sum_{k=1}^{i} y_{(k)} - (n\tau - i)\alpha_i$$

or maximizing

$$\sum_{k=1}^{i} y_{(k)} + (n\tau - i)\alpha_i = l_i$$

Therefore,

$$l_j - l_{j-1} = y_j - \alpha_{j-1} + (n\tau - j)(\alpha_{j-1} - \alpha_j)$$

Hence, $\forall j : j \leq n\tau, \quad l_j - l_{j-1} >= 0$, since $(y_j - \alpha_{j-1})$, $(n\tau - j)$ and $(\alpha_{j-1} - \alpha_j)$ are all ≥ 0. Similarly, $\forall j : j \geq n\tau$,

$$l_j - l_{j+1} = \alpha_{j+1} - y_{j+1} + (n\tau - j)(\alpha_j - \alpha_{j+1}) \geq 0$$

Hence, if $\exists i : i = n\tau$, then $\alpha = y_{(n\tau)}$. But if, $i < n\tau < (i+1)$, then α is in the interval $[y_{(i)}, y_{(i+1)}]$ □

Figure 3 is a plot that tracks the values of β for different λ values. The figure shows that the regression parameter vector $\boldsymbol{\beta}$ will converge to $(\alpha, 0, 0, \ldots, 0)^T$ as λ increases. β_0 is the regression parameter that corresponds to the column of 1's in the design matrix.

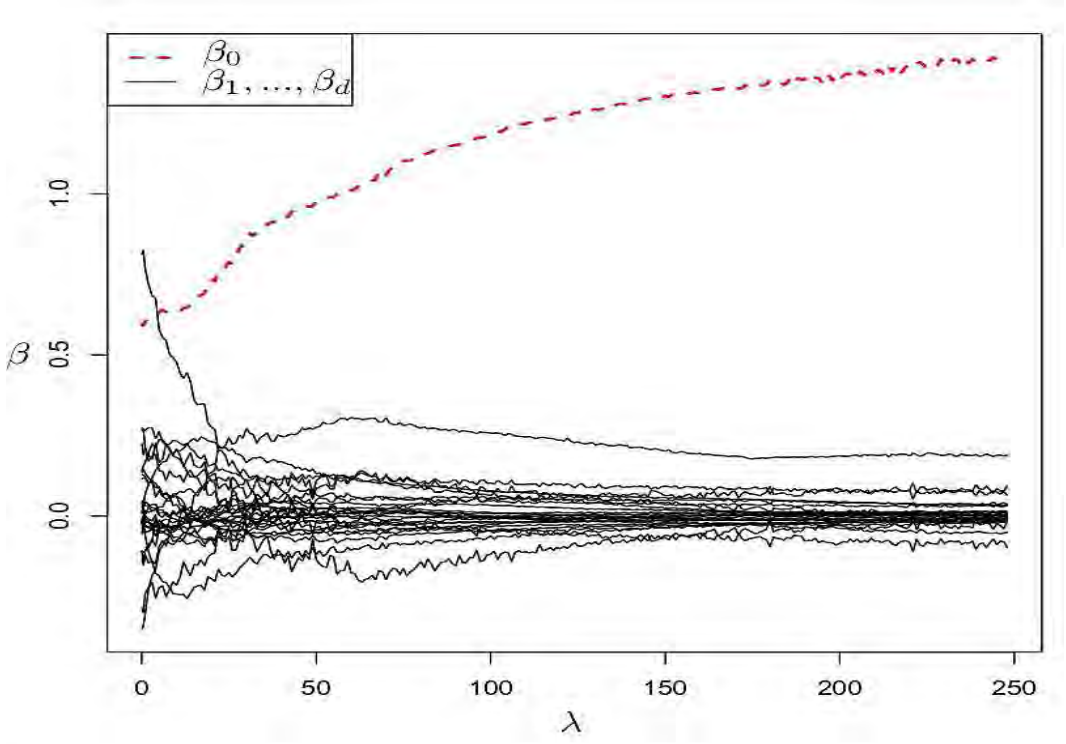


FIGURE 3. Influence of parameter λ on the regression coefficients β in LSQR.

Figures 4 and 5 plots the influence of λ on the predicted values returned from LSSQR. i.e., as the value of λ increases, LSSQR shrinks the prediction range to the quantile τ. Figure 5 is a zoomed-in image, capturing the tail of Figure 4.

4.2. **Linear semi-supervised quantile regression (LSSQR).** The objective function of LSQR can be easily extended to a semi-supervised learning setting since the smoothing factor (the second term in the equation) is independent of y. Therefore, by extending the range of the indices i and j of the smoothing term to span 1 to $n + m$, the predictor variables of the unlabeled data $X_u = [x_{u1}, ..., x_{um}]^T$ can be harvested.

The objective function of the LSSQR is

$$\arg\min_{\beta} \sum_{i=1}^{n} \rho_\tau(y_i - x_i^T \beta) + \lambda \sum_{i,j}^{n+m} w_{ij}(x_i^T \beta - x_j^T \beta)^2$$

5. Experimental Results

In this section, the climate dataset that is used for statistical downscaling is described. This is followed by the experimental setup, which address the inherent properties of the dataset, such as its periodic nature. Once the dataset is introduced, we analyze the behavior of baseline models developed using MLR, ridge regression and quantile regression and contrast them with LSQR and LSSQR. The efficacy of the models in accurately measuring the magnitude, the relative frequency and timing of forecasting a data point as an extreme event is measured.

5.1. **Data.** All the algorithms were run on climate data obtained at 37 weather stations in Canada, from the Canadian Climate Change Scenarios Network website [1]. The response variable to be regressed (downscaled) corresponds to daily temperature values measured at each weather station.

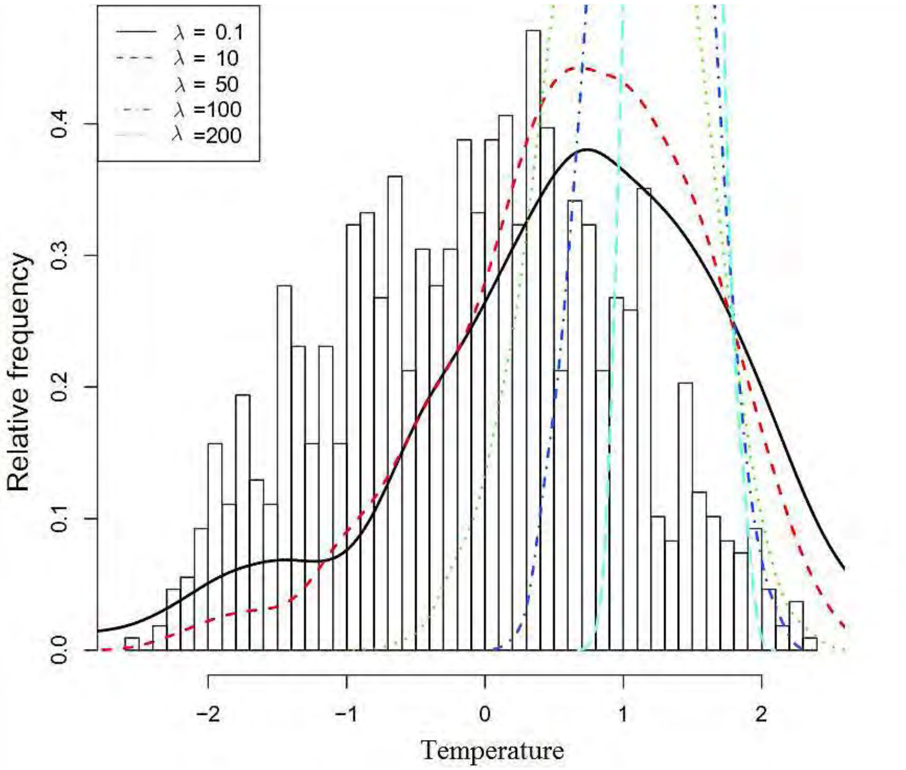


FIGURE 4. Influence of λ on the probability distribution of the predicted values obtained from LSSQR.

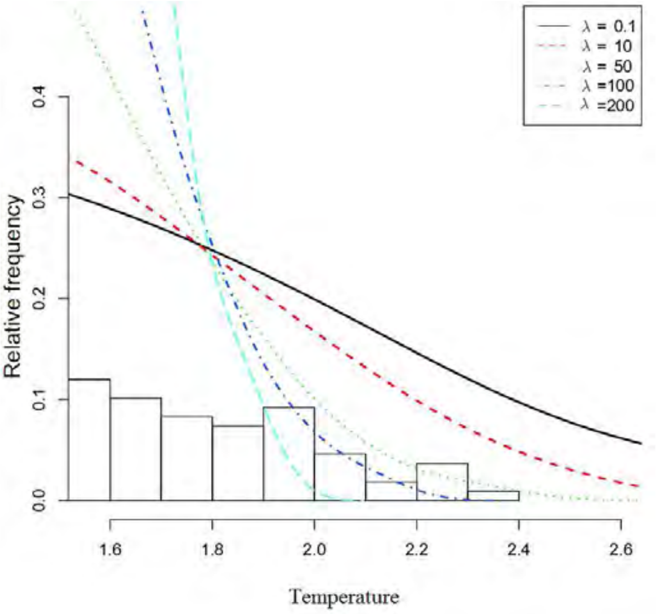


FIGURE 5. Influence of λ on the probability distribution of the predicted extreme values obtained from LSSQR.

The predictor variables for each of the 37 stations correspond to 26 coarse-scale climate variables derived from the NCEP re-analysis data set, which include measurements of airflow strength, sea-level pressure, wind direction, vorticity, and humidity, as shown in Table 1. The predictor variables used for training were obtained from the NCEP re-analysis data set that span a 40-year period (1961 to 2001). The time series was truncated for each weather station to exclude days for which temperature or any of the predictor values are missing.

Table 1. List of predictor variables for temperature prediction.

Predictor Variables	
500 hPa airflow strength	850 hPa airflow strength
500 hPa zonal velocity	850 hPa zonal velocity
500 hPa meridional velocity	850 hPa meridional velocity
500 hPa vorticity	850 hPa vorticity
500 hPa geopotential height	850 hPa geopotential height
500 hPa wind direction	850 hPa wind direction
500 hPa divergence	850 hPa divergence
Relative humidity at 500 hPa	Relative humidity at 850 hPa
Near surface relative humidity	Surface specific humidity
Mean sea level pressure	Surface zonal velocity
Surface airflow strength	Surface meridional velocity
Surface vorticity	Surface wind direction
Surface divergence	Mean temp at 2 m

5.2. **Experimental setup.** As is well known, temperature, which is the response variable in our experiments, has seasonal cycles. To efficiently capture the various cycles, de-seasonalization is performed prior to running the experiments. As is common practice in the field of climatology, a common approach to de-seasonalization is to split the data into 4 seasons (DJF, MAM, JJA, SON) where 'DJF' refers to the months of December-January-February in the temperature timeseries. Similarly, 'MAM' refers to March-April-May, and 'JJA' refers to June-July-August and 'SON', September-October-November. In effect, for each station, we build 4 different models, corresponding to the 4 seasons. The training size used spanned 6 years of data and the test size, 12 years. During validation, the parameter λ was selected using the score returned by RMSE for extreme data points. A data point is considered extreme if its response variable is greater than .95 percentile (Threshold-1) of the whole dataset corresponding to the station. QR was implemented using the interior point algorithm as detailed in [2]. Broyden Fletcher Goldfarb Shanno (BFGS) method was used to solve the LSQR and LSSQR optimization problem.

5.3. **Evaluation criteria.** The motivation behind the selection of the evaluation metrics was the intent to evaluate the different algorithms in terms of accuracy of the prediction of extreme values, the timing of the extreme events as well as the frequency with which a data point is predicted to be an extreme data point. The following metrics are used to capture the above evaluation criteria for the various models:

- Root Mean Square Error (RMSE), which measures the difference in magnitude between the actual and predicted values of the response variable, i.e.: RMSE $= \sqrt{\frac{\sum_{i=1}^{n}(y'_i - f'_i)^2}{n}}$. RMSE was computed on those days that were observed to be extreme data points.
- Precision and recall of extreme events are computed to measure the timing accuracy of the prediction. F-measure, which is the harmonic mean between recall and precision values, will be used as a score that summarizes the precision and recall results. F-measure $= \frac{2 \times Recall \times Precision}{Recall + Precision}$

- The frequency of predicting extreme data point for the various methods was measured by computing the ratio of the number of data points that were predicted to be extreme to the number of observed extreme data points.

To summarize, RMSE is used for measuring the accuracy of the predicted magnitude of the response variable, whereas F-measure can be thought of as measuring the correctness of the timing of the extreme events.

5.4. **Baseline.** We compared the performance of LSQR and LSSQR with baseline models created using multiple linear regression (MLR), ridge regression (Ridge), and quantile regression (QR). All the baselines were run for the same 37 stations and for all the 4 seasons. Also, a comparison of the performance of the proposed supervised framework (LSQR) is made with its semi-supervised counterpart (LSSQR), where LSSQR demonstrated an improved performance over LSQR for the 37 stations evaluated upon as shown in Table 2. Table 2 summarizes the tally of percentage of times LSSQR outperformed LSQR over the 4 seasons for the given 37 stations. As seen in the table, LSSQR showed an improved performance in terms of both RMSE and F-measure.

Table 2. The relative performance of LSSQR compared with LSQR with regard to the extreme data points.

	Win	Loss	Tie
RMSE	68.25%	31.75%	0%
F-measure	60.14%	37.16%	2.7%

5.5. **Results.** As mentioned earlier, experiments were run separately using each of the baseline approaches and LSQR and LSSQR for the 4 seasons (DJF, MAM, JJA, SON) of the year for each of the 37 stations' data. The results over all the seasons and stations are summarized in Tables 3 and 4 while the individual results of each season in Figures 6 and 8. Table 3 summarizes the relative performance of LSQR with respect to the baseline methods in terms of RMSE of extreme data points and F-measure of identification of extreme data points. During testing, a data point is considered extreme, if its response variable is greater than .95 percentile (Threshold-1) of the whole dataset corresponding to the station. For the purpose of analysis, results of using the .95 percentile of the response variable in the training set (Threshold-2) to identify extreme data points are also summarized. The fact that the results obtained by using the two different baselines is an indicator that the training data did capture the distribution of the response variable reasonably well. LSQR consistently outperformed the baselines both in terms of RMSE and F-measure. It must also be noted that LSQR did outperform MLR and Ridge in terms of recall of extreme events comprehensively across each of the 37 stations and seasons.

Table 3. The percentage of stations LSQR outperformed the respective baselines, with regard to the extreme data points.

		MLR	Ridge	QR
RMSE	Threshold-1	88.51%	87.84%	80.40%
	Threshold-2	89.19%	87.84%	79.05%
F-measure	Threshold-1	59.45%	60.13%	72.97%
	Threshold-2	56.08%	58.10%	79.05%

Similarly, Table 4 summarizes the relative performance of LSSQR with respect to the baseline methods in terms of RMSE of extreme data points and F-measure of identification of extreme data points. Like LSQR, LSSQR consistently outperformed the baselines both in terms of RMSE and

TABLE 4. The percentage of stations LSSQR outperformed the respective baselines, with regard to the extreme data points.

		MLR	Ridge	QR
RMSE	Threshold-1	87.16%	85.14%	85.13%
	Threshold-2	87.84%	86.49%	81.76%
F-measure	Threshold-1	60.13%	58.78%	75.67%
	Threshold-2	56.75%	59.45%	81.75%

F-measure. It must be noted that LSSQR outperform MLR and Ridge in terms of recall of extreme events comprehensively across each of the 37 stations and seasons.

Figure 6 gives a breakdown of the performance of the LSSQR over each of the 4 seasons of the 37 stations using Threshold-1 for the purpose of marking a data point as extreme. The figure is a bar chart of percentage of stations that LSSQR outperformed MLR, ridge regression and QR in prediction accuracy for only extreme data points in the test set. RMSE was used to compute the accuracy of each model in predicting extreme value data points, at the 37 stations. As seen in the plot, LSSQR outperforms MLR, ridge regression and QR in each of the four seasons across the 37 stations.

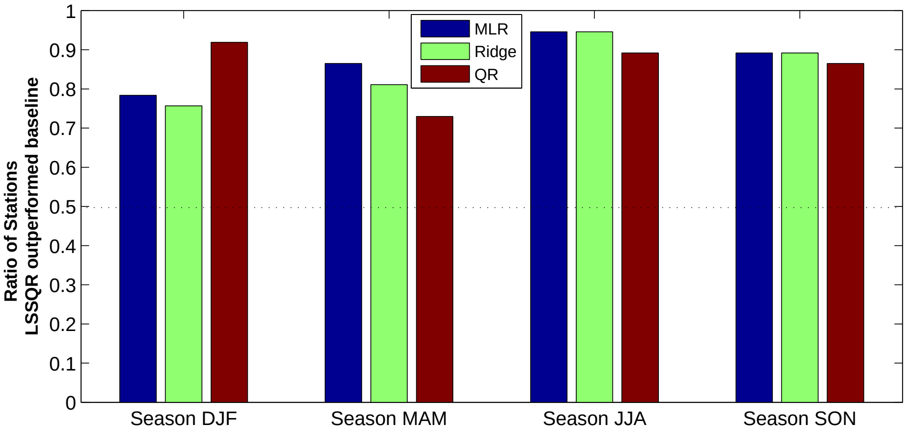


FIGURE 6. Ratio of stations LSSQR outperforming baseline in terms of RMSE of extreme data points.

Figure 7 shows a graph that depicts the percentage of stations LSSQR outperformed MLR, ridge regression and QR in terms of identifying extreme data points over 37 stations. Again, LSSQR comprehensively outperforms MLR and ridge regression over all the 37 stations and 4 seasons. But as expected, QR outperforms LSSQR in terms of recall performance for each of the 4 seasons due to the overestimating nature of QR, which consequently resulted in poor precision and which is reflected in its F-measure score. At an average, quantile regression, predicted a datapoint to be an extreme point more than twice as frequently as the actual frequency of observed extreme data points. In fact, QR lost out to LSSQR in 91% of 37 stations across 4 seasons in terms of precision of identifying extreme data points.

Figure 8 shows a graph that depicts the percentage of stations where LSSQR outperformed MLR, ridge regression and QR in prediction accuracy based on F-measure of the identifying extreme data points over 37 stations. Again, LSSQR outperforms MLR, ridge regression and QR for all the 4 seasons.

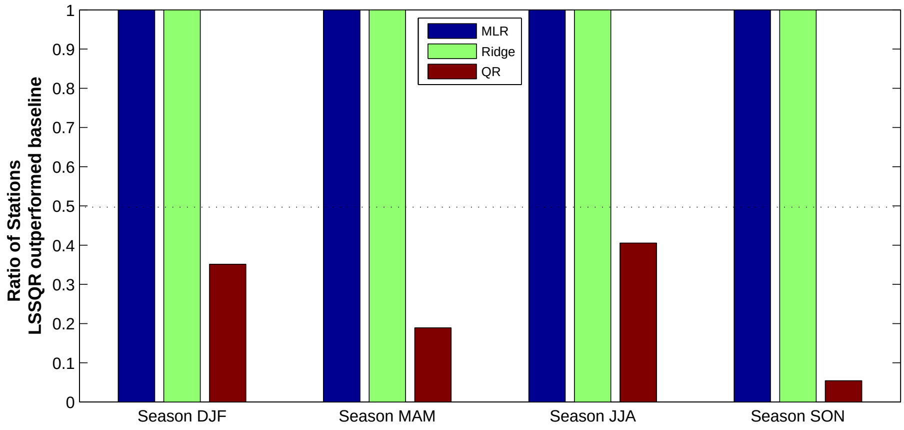


FIGURE 7. Ratio of stations LSSQR outperforming baseline in terms of recall of extreme data points.

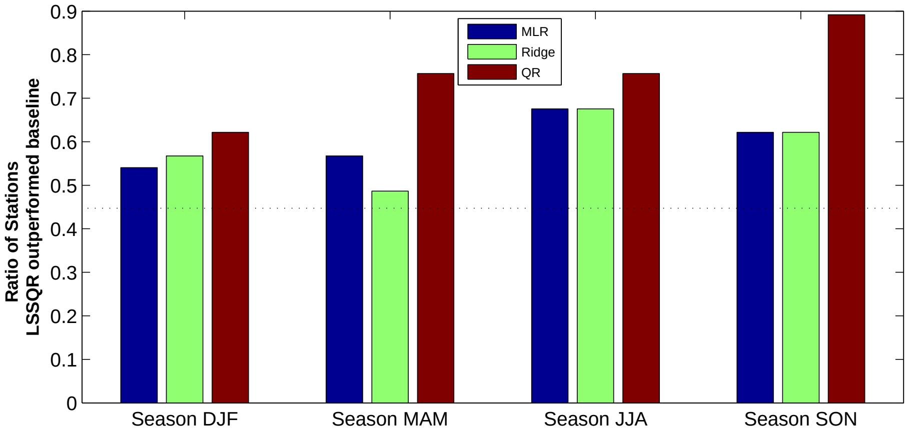


FIGURE 8. Ratio of stations LSSQR outperformin baseline in terms of F-measure of extreme data points.

The performance improvement obtained by LSSQR in terms of predicting the extreme values can be easily visualized in Figure 9. Figure 9 is a plot comparing the predicted response variable of the various methods. The plot is restricted to only extreme data points for a station. As expected, the predicted value of the response variable using multiple linear regression is often underestimating the observed temperature, while quantile regression regularly overestimates the prediction of temperature and LSSQR lies in between MLR and QR and closer to the observed temperature.

6. CONCLUSIONS

This paper presents a semi-supervised framework (LSSQR) for recalling and accurately predicting values for extreme data points. The proposed approach was applied to real world climate data spanning 37 stations and was compared with MLR, ridge regression and quantile regression in terms of the effectiveness the model demonstrated in identifying and predicting extreme temperatures for the given stations. For future work, we will explore a non-linear variant of the smoothed quantile

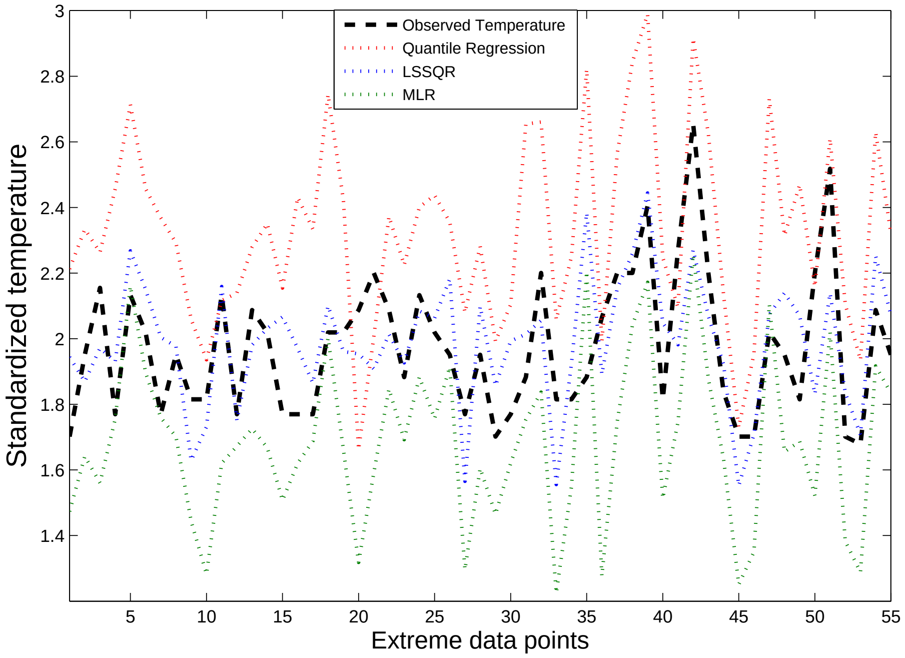


FIGURE 9. Prediction performance of extreme data points using MLR, Ridge, QR, LSSQR.

regression framework. We will also explore a semi-supervised variant of the non-linear smoothed quantile regression model.

7. ACKNOWLEDGMENTS

This work is partly supported by National Science Foundation Dynamics of Coupled Natural and Human Systems competition Program (CNH Award No.-0909378). The authors would like to thank Dr. Julie Winkler for valuable discussion on statistical downscaling for climate change projection.

REFERENCES

[1] Canadian Climate Change Scenarios Network, Environment Canada. http://www.ccsn.ca/

[2] R. Koenker. Quantile Regression Software. http://www.econ.uiuc.edu/ roger/research/rq/rq.html

[3] S. Ancelet, M.-P. Etienne, H. Benot, and E. Parent. Modelling spatial zero-inflated continuous data with an exponentially compound poisson process. *Environmental and Ecological Statistics*, DOI:10.1007/s10651-009-0111-6, April 2009.

[4] S. Barry and A. H. Welsh. Generalized additive modelling and zero inflated count data. *Ecological Modelling*, 157(2-3):179–188, November 2002.

[5] A. Blum and S. Chawla. Learning from labeled and unlabeled data using graph mincuts. In *Proc. of the 18th Int'l Conf. on Machine Learning*, pages 19–26, 2001.

[6] J. Bjornar Bremnes, Probabilistic Forecasts of Precipitation in Terms of Quantiles Using NWP Model Output. In *Monthly Weather Review*, pages 338–347, 2004

[7] L. Feudale. Large scale extreme events in surface temperature during 1950–2003: an observational and modeling study In *Ph.D. Dissertation. George Mason University*

[8] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proc. of the Workshop on Computational Learning Theory*, pages 92–100, 1998.

[9] D. Bohning, E. Dierz, and P. Schlattmann. Zero-inflated count models and their applications in public health and social science. In J. Rost and R. Langeheine, editors, *Applications of Latent Trait and Latent Class Models in the Social Sciences.* Waxman Publishing Co, 1997.

[10] Y.-A. L. Borgne, S. Santini, and G. Bontempi. Adaptive model selection for time series prediction in wireless sensor networks. *Signal Process*, 87(12):3010–3020, 2007.

[11] U. Brefeld, T. Gärtner, T. Scheffer, and S. Wrobel. Efficient co-regularised least squares regression. In *Proc. of the 23rd Int'l Conf. on Machine learning*, pages 137–144, 2006.

[12] S. Charles, B. Bates, I. Smith, and J. Hughes. Statistical downscaling of daily precipitation from observed and modelled atmospheric fields. In *Hydrological Processes*, pages 1373–1394, 2004.

[13] Z. Abraham and P.-N. Tan. An Integrated Framework for Simultaneous Classification and Regression of Time-Series Data. In *Proc of the ACM SIGKDD Int'l Conf on Data Mining*, Colorado, OH, 2010.

[14] H. Cheng and P.-N. Tan. Semi-supervised learning with data calibration for long-term time series forecasting. In *Proc of the ACM SIGKDD Int'l Conf on Data Mining*, Las Vegas, NV, 2008.

[15] I. Cohen, N. Sebe, F. G. Cozman, M. C. Cirelo, and T. S. Huang. Semi-supervised learning of classifiers: Theory and algorithms for bayesian network classifiers and applications to human-computer interaction. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(12):1553–1566, Dec 2004.

[16] C. Cortes and M. Mohri. On transductive regression. In *Advances in Neural Information Processing Systems*, 2006.

[17] F. Cozman and I. Cohen. Unlabeled data can degrade classification performance of generative classifiers. In *Proc. of the 15th Int'l Florida Artificial Intelligence Society Conference*, pages 327–331, 2002.

[18] F. Cozman, I. Cohen, and M. Cirelo. Semi-supervised learning of mixture models. In *Proc of the 20th Int'l Conf. on Machine Learning*, 2003.

[19] W. Enke and A. Spekat. Downscaling climate model outputs into local and regional weather elements by classification and regression. In *Climate Research 8*, pages 195–207, 1997.

[20] D. Erdman, L. Jackson, and A. Sinko. Zero-inflated poisson and zero-inflated negative binomial models using the countreg procedure. In *SAS Global Forum 2008*, pages 1–11, 2008.

[21] P. Friederichs and A. Hense Statistical Downscaling of Extreme Precipitation Events Using Censored Quantile Regression. In *Monthly Weather Review*, pages 2365–2378, 2007

[22] K. Fujinaga, M. Nakai, H. Shimodaira, and S. Sagayama. Multiple-regression hidden markov model. In *Proc. of IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing*, 2001.

[23] C. Giles, S. Lawrence, and A. Tsoi. Noisy time series prediction using a recurrent neural network and grammatical inference. *Machine Learning, 44(1-2)*, pages 161–183, 2001.

[24] W. Hong, P. Pai, S. Yang, and R. Theng. Highway traffic forecasting by support vector regression model with tabu search algorithms. In *Proc. of Int'l Joint Conf. on Neural Networks*, pages 1617–1621, 2006.

[25] T. Joachims. Transductive inference for text classification using support vector machines. In *Proc. of the 16th Int'l Conf. on Machine Learning*, pages 200–209, Bled, SL, 1999.

[26] R. Koenker and K. Hallock. Quantile Regression. *Journal of Economic PerspectivesVolume 15, Number 4*, pages 143-156, 2001.

[27] B. Kedem and K. Fokianos. Regression models for time series analysis. *Wiley-Interscience ISBN: 0-471-36355*, 2002.

[28] E.C. Mannshardt-Shamseldin, R.L. Smith, S.R. Sain, L.D. Mearns and D. Cooley Downscaling extremes: A comparison of extreme value distributions in point-source and gridded precipitation data. In *Annals of Applied Statistics*, pages 484–502, 2010.

[29] T. Martin, B. Wintle, J. Rhodes, P. Kuhnert, S. Field, S. Low-Choy, A. Tyre, and H. Possingham. Zero tolerance ecology: improving ecological inference by modelling the source of zero observations. *Ecology Letters*, 8:1235–1246, 2005.

[30] N. Meinshause. Quantile Regression Forests. *Journal of Machine Learning Research 7*, 7:9839-99, 2006.

[31] C. Monteleoni, G. Schmidt AND S. Saroha Tracking Climate Models. *NASA Conference on Intelligent Data Understanding (CIDU)*, 2010.

[32] A. Ober-Sundermeier and H. Zackor. Prediction of congestion due to road works on freeways. In *Proc. of IEEE Intelligent Transportation Systems*, pages 240–244, 2001.

[33] A. Smola and B. Scholkopf. A tutorial on support vector regression. In *Statistics and Computing*, pages 199–222(24). Spring, 2004.

[34] I. Takeuchi, Q.V. Le, T. Sears and A.J. Smola. Nonparametric Quantile Regressio. In *Journal of Machine Learning Research Nonparamteric Quantile Estimation*,2005.

[35] Q. Tian, J. Yu, Q. Xue, and N. Sebe. A new analysis of the value of unlabeled data in semi-supervised learning for image retrieval. In *Proc. of IEEE Int'l Conf. on Multimedia and Expo.*, pages 1019– 1022, 2004.

[36] E. Towler, B. Rajagopalan, E. Gilleland, R.S. Summers, D. Yates, and R.W. Katz Modeling hydrologic and water quality extremes in a changing climate: A statistical approach based on extreme value theory. In *Water Resources Research*, VOL. 46, W11504, 2010

[37] L. Wei and E. J. Keogh. Semi-supervised time series classification. In *Proc of ACM SIGKDD Int'l Conf on Data Mining*, pages 748–753, Philadelphia, PA, August 2006.

[38] A. H. Welsh, R. Cunningham, C. Donnelly, and D. B. Lindenmayer. Modelling the abundance of rare species: statistical models for counts with extra zeros. In *Ecological Modelling.* Elsevier, Amsterdam, PAYS-BAS (1975) (Revue), 1996.

[39] R. Wilby, S. Charles, E. Zorita, B. Timbal, P. Whetton, and L. Mearns. Guidelines for use of climate scenarios developed from statistical downscaling methods. Available from the DDC of IPCC TGCIA, 2004.

[40] C.-C. Wong, M.-C. Chan, and C.-C. Lam. Financial time series forecasting by neural network using conjugate gradient learning algorithm and multiple linear regression weight initialization. Technical Report 61, Society for Computational Economics, Jul 2000.

[41] L. Youjuan,L. Yufeng, and Ji Z HU Quantile Regression in Reproducing Kernel Hilbert Spaces In *American Statistical Association Vol. 102, No. 477, Theory and Methods*, 2007

[42] T. Zhang. The value of unlabeled data for classification problems. In *Proc of the Int'l Conf. on Machine Learning*, 2000.

[43] Z. Zhou and M. Li. Semi-supervised regression with co-training. In *Proc. of Int'l Joint Conf. on Artificial Intelligence*, 2005.

[44] X. Zhu. Semi-supervised learning literature survey. In *Technical Report,Computer Sciences, University of Wisconsin-Madison*, 2005.

[45] X. Zhu, Z. Ghahramani, and J. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *Proc. of the 20th Int'l Conf. on Machine Learning*, volume 20, 2003.

[46] X. Zhu and A. Goldberg. Kernel regression with order preferences. In *Association for the Advancement of Artificial Intelligence*, page 681, 2007.

[47] C. Dorland, R.S.J. Tol and J.P. Palutikof. Vulnerability of the Netherlands and Northwest Europe to storm damage under climate change. In *Climatic Change*, pages 513-535, 1999.

[48] Y. Hundecha, A. St-Hilaire, T.B.M.J. Ouarda, S. El Adlouni, and P. Gachon. A nonstationary extreme value analysis for the assessment of changes in extreme annual wind speed over the Gulf of St. Lawrence, Canada. In *Journal of Applied Meteorology and Climatology*, pages 2745-2759, 2008.

[49] M.J. Booij, Extreme daily precipitation in Western Europe with climate change at appropriate spatial scales. In *International Journal of Climatology*, 2002.

[50] N.B. Bernier, K.R. Thompson, J. Ou, and H. Ritchie. Mapping the return periods of extreme sea levels: Allowing for short sea level records, seasonality, and climate change. In *Global and Planetary Change*, pages 139-150, 2007.

[51] R.T. Clarke. Estimating trends in data from the Weibull and a generalized extreme value distribution. In *Water Resources Research*, 2002.

TIME SERIES RECONSTRUCTION VIA MACHINE LEARNING: REVEALING DECADAL VARIABILITY AND INTERMITTENCY IN THE NORTH PACIFIC SECTOR OF A COUPLED CLIMATE MODEL

DIMITRIOS GIANNAKIS* AND ANDREW J. MAJDA*

Abstract. Many processes in atmosphere-ocean science develop multiscale temporal and spatial patterns, with complex underlying dynamics and time-dependent external forcings. Because of the possible advances in our understanding and prediction of climate phenomena, extracting that variability empirically from incomplete observations is a problem of wide contemporary interest. Here, we present a technique for analyzing climatic time series that exploits the geometrical relationships between the observed data points to recover features characteristic of strongly nonlinear dynamics (such as intermittency), which are not accessible to classical Singular Spectrum Analysis (SSA). The method utilizes Laplacian eigenmaps, evaluated after suitable time-lagged embedding, to produce a reduced representation of the observed samples, where standard tools of matrix algebra can be used to perform truncated Singular Value Decomposition despite the nonlinear manifold structure of the data set. As an application, we study the variability of the upper-ocean temperature in the North Pacific sector of a 700-year equilibrated integration of the CCSM3 model. Imposing no a priori assumptions (such as periodicity in the statistics), our machine-learning technique recovers three distinct types of temporal processes: (1) periodic processes, including annual and semiannual cycles; (2) decadal-scale variability with spatial patterns resembling the Pacific Decadal Oscillation; (3) intermittent processes associated with the Kuroshio extension and variations in the strength of the subtropical and subpolar gyres. The latter carry little variance (and are therefore not captured by SSA), yet their dynamical role is expected to be significant.

1. Introduction

Coupled atmosphere-ocean processes exhibit variability across a broad range of time scales, including seasonal, interannual, and decadal time scales [19, 27, 20, 21, 26]. There is a strong interest among the climate community in extracting physically-meaningful information about this variability using data from models or observations, with the goal of enhancing our understanding of the underlying dynamics, and improving our predictive capabilities.

A classical way of attacking this problem is through Singular Spectrum Analysis (SSA), or one of its variants [28, 3, 18, 15]. Here, a low-rank approximation of a dynamic process is constructed by first embedding a time series of a scalar or multivariate observable in a high-dimensional vector space H (called embedding space) using the method of delays [25, 24, 14], and then performing a truncated singular-value decomposition (SVD) of the matrix X containing the embedded data [8]. In this manner, information about the dynamical process is extracted from the left and right singular vectors of X with the k largest singular values. The left singular vectors form a set of empirical orthogonal functions (EOFs) which, at each instance of time, are weighted by the corresponding principal components (PCs) determined from the right singular vectors to yield a rank-k reconstruction of X.

A potential drawback of this approach is that it is based on minimizing an operator norm which may be unsuitable for the nonlinear processes encountered in atmosphere-ocean science (AOS). Specifically, the PCs are computed by projecting onto the principal axes of the k-dimensional ellipsoid that best fits the data in the least-squares sense. This construction is optimal when the underlying dynamics are linear, but nonlinear processes will in general give rise to a manifold M in embedding space that deviates significantly from an ellipsoidal shape. Physically, a prominent manifestation

*Courant Institute of Mathematical Sciences, New York University, dimitris@cims.nyu.edu, jonjon@cims.nyu.edu.

of this phenomenon is failure to capture via SSA the intermittent patterns arising in turbulent dynamical systems; i.e., temporal processes that carry low variance but play an important dynamical role [13].

Despite their inherently nonlinear character, such data sets possess a natural linear structure, namely the Hilbert space $L^2(M, \mu)$ of square-integrable functions on M with inner product inherited from the volume element μ of M (the Riemannian measure). This space may be thought of as the collection of all possible weights that can be assigned to the data samples when making a reconstruction, i.e., it is analogous to the space spanned by the right singular vectors in SSA [3]. Similarly, the left singular vectors are naturally identified with elements of the dual space H^* to H. Therefore, it is reasonable to develop algorithms that seek to approximate suitably defined maps from $L^2(M, \mu)$ to H^*. Such maps, denoted here by A, have the advantage of being simultaneously linear and compatible with the nonlinear manifold comprised by the data.

In this paper, we advocate that this approach, implemented via algorithms developed in machine learning, can reveal important aspects of complex AOS data sets which are not accessible to standard SSA. Here, an orthonormal basis for $L^2(M, \mu)$ is constructed through eigenfunctions of the Laplace-Beltrami operator on M, computed efficiently via sparse graph-theoretic algorithms [4, 10]. Projecting the data from embedding space H onto these eigenfunctions gives a matrix representation of A, such that the optimal rank-k reconstruction with respect to the natural norm of maps from $L^2(M, g)$ to H^* is given by standard truncated SVD.

We demonstrate the efficacy of the scheme in an analysis of the North Pacific sector of the Community Climate System model version 3 (CCSM3) [12]. Using a 700-year equilibrated data set of the upper 300 m ocean [1, 26, 7], we identify a number of qualitatively-distinct spatiotemporal processes, each with a meaningful physical interpretation. These include the seasonal cycle, semiannual variability, as well as decadal-scale processes resembling the Pacific Decadal Oscillation (PDO).

Besides these modes, which are familiar from SSA, the spectrum of the manifold-based algorithm also contains modes with a strongly intermittent behavior in the temporal domain, characterized by five-year periods of high-amplitude oscillations with annual and semiannual frequencies, separated by periods of quiescence. Spatially, these modes describe enhanced eastward transport in the Kuroshio extension region, as well as retrograde (westward) propagating temperature anomalies and circulation patterns resembling the subpolar and subtropical gyres. The bursting-like behavior of these modes, a hallmark of strongly-nonlinear dynamics, means that they carry little variance of the raw signal (about an order of magnitude less than the seasonal and PDO modes). As a result, they are not captured by linear SSA.

The plan of this paper is as follows. In Section 2 we describe our theoretical framework. In Section 3 we apply this framework to the upper-ocean temperature in the North Pacific sector of CCSM3. We discuss the implications of these results in Section 4, and conclude in Section 5.

2. Theoretical framework

We consider that we have at our disposal samples of a time-series x_t of a d-dimensional climatic variable sampled uniformly with time step δt. Here, $x_t \in \mathbb{R}^d$ is generated by a dynamical system, but observations of x_t alone are not sufficient to uniquely determine the state of the system in phase space; i.e., our observations are incomplete. For instance, in Section 3 ahead, x_t will be a depth-averaged ocean temperature field restricted in the North-Pacific sector of CCSM3. Our objective is to produce a low-rank reconstruction of x_t taking explicitly into account the fact that the underlying trajectory of the dynamical system lies on a nonlinear manifold M in phase space.

The methodology employed here to address this objective consists of five basic steps: (1) embed the observed data in a vector space of dimension greater than d via the method of delays; (2) map the data from embedding space to a set of orthonormal Laplacian eigenfunctions; (3) evaluate a low-rank approximation of the data in reduced coordinates determined through the eigenfunctions; (4) convert the approximated data back to embedding space; (5) project to physical space $\mathbb{R}^d$ to obtain the reconstructed signal. Below, we provide a summary of each step. Details of the procedure will be presented elsewhere. Hereafter, we shall consider that M is compact and smooth, so that a well-defined spectral theory exists [6]. Even though these conditions may not be fulfilled in practice, eventually we will pass to a discrete, graph-theoretic description [9], where smoothness is not an issue.

Step (1) is familiar from the qualitative theory of dynamical systems [23, 25, 24, 14]. Under generic conditions, the image of x_t in embedding space $H = \mathbb{R}^n$ under the delayed-coordinate mapping,

$$x_t \mapsto X_t = (x_t, x_{t-\delta t}, \dots, x_{t-(q-1)\,\delta t}) \tag{1}$$

lies on a manifold which is diffeomorphic to M (i.e., indistinguishable from M from the point of view of differential geometry), provided that the dimension n of H is sufficiently large. Thus, given a sufficiently-long embedding window $\varDelta t = (q-1)\,\delta t$, we obtain a representation of the nonlinear manifold underlying our incomplete observations, which can be thought of as a curved hypersurface in Euclidean space. That hypersurface inherits a Riemannian metric g, i.e., an inner product between tangent vectors on M constructed from the canonical inner product of H.

Steps (2) and (3) effectively constitute a generalization of SSA, adapted to nonlinear data sets. Recall that SSA is essentially an SVD decomposition,

$$X = U \varSigma V^T, \tag{2}$$

of the data matrix $X = [X_0, X_{\delta t}, \dots, X_{(s-1)\delta t}]$, dimensioned $n \times s$ for s samples in n-dimensional embedding space. Here, the key observation is that the map in Eq. (1) naturally gives rise to two linear vector spaces, which are analogous to the spaces spanned by left and right singular vectors of X [3]. The first is the space $L^2(M, \mu)$ of square-integrable functions on M, where $\mu = (\det g)^{1/2}$ is the volume element (Riemannian density) induced on M through the embedding $M \mapsto H$. The second space of interest is the dual space H^* of H. The elements of H^* are functionals, mapping observed data points in H to the real numbers.

To see the correspondence with SVD, let f be a function in $L^2(M, \mu)$, z an arbitrary vector in H, and consider the dual vector $h \in H^*$ defined by

$$h(z) = \int_M \mu(X_t) f(X_t) \langle X_t, z \rangle, \tag{3}$$

where $\langle \cdot, \cdot \rangle$ is the inner product of H. That is, f assigns a weight proportional to $f(X_t)$ on the dual vector $\langle X_t, \cdot \rangle$, much like the i-th column of V weighs the i-th column of U in Eq. (2). What one gains by phrasing the problem in this manner is a linear map A taking $L^2(M, \mu)$ to H^* via the rule in Eq. (3), viz. $A(f) = h$. Note that this definition is basis-independent. Moreover, unlike the nonlinear manifold M, A is amenable to analysis through the standard tools of linear algebra. In particular, low-rank reconstruction of A is a well-defined notion.

Having the latter as an objective, the role of the Laplacian eigenfunctions in step (2) is to provide an orthonormal basis of $L^2(M, \mu)$, in which the operator norm $\|A\|$ can be straightforwardly computed via the Frobenius norm of its matrix representation [Eq. (6) ahead]. Specifically, it is well known that the eigenfunctions $\{\phi_0, \phi_1, \dots\}$ of the Laplace-Beltrami operator Δ associated with the metric g, defined via $\Delta \phi_i = \lambda_i \phi_i$ (together with appropriate boundary conditions if M has boundaries), lead to an orthogonal decomposition of $L^2(M, \mu)$ into invariant subspaces $\varPhi_i$. That is, we have [6]

$$L^2(M, \mu) = \bigoplus_{i=0}^{\infty} \varPhi_i \quad \text{with } \varPhi_i = \operatorname{span}\{\phi_k : \lambda_k = \lambda_i\}, \tag{4a}$$

$$\int_M \mu(X) f_i(X) f_j(X) = 0 \quad \text{for any } f_i \in \varPhi_i,\ f_j \in \varPhi_j, \text{ and } j \neq i. \tag{4b}$$

The components of A in this basis are

$$A_{ij} = \int_M \mu(X_t)(X_t)_i \phi_j(X_t), \tag{5}$$

with $(X_t)_i$ the i-th element of X_t, giving the operator norm through

$$\|A\|^2 = \sum_{ij} A_{ij}^2. \tag{6}$$

Equation (5) may be interpreted as a Fourier transform on compact manifolds.

In applications, the Laplace-Beltrami eigenfunctions for a finite data set are computed by replacing the continuous manifold M via a weighted graph G, and solving the eigenproblem of a Markov matrix P defined on G, constructed so that in the continuum limit, $s \to \infty$, the generator of P (the graph Laplacian) converges to Δ [4, 11, 10, 5]. Note that the Markov matrix employed in this procedure is highly sparse, which means that the cost of the eigenvalue problem for (λ_i, ϕ_i) grows linearly with the number of samples.

The least-favorable scaling in the eigenfunction calculation involves the pairwise distance calculation between the data samples in embedding space. This scales quadratically with the number of samples if done with brute force, which is the approach adopted here. However, an $s \log s$ scaling may be realized if the dimension of H is small-enough for approximate kd-tree-based algorithms to operate efficiently [2]. In the present study, all eigenfunction calculations were performed on a desktop workstation. The scalability of this class of algorithms to large problem sizes has been widely demonstrated in the machine learning and data mining literature.

In step (3), a rank-k approximation $\tilde{A}$ of A is evaluated by selecting the first r invariant subspaces in order of increasing λ_i (with $l = \sum_{i=1}^{r} \dim \varPhi_i \geq k$), and performing a truncated SVD of the $n \times l$ matrix $\hat{A} = [A_{ij}]_{j \leq l}$. That is, in matrix notation, the nonzero components of $\tilde{A}$ are

$$\tilde{A} = U_k \varSigma_k V_k^T, \tag{7}$$

where $\varSigma_k$ is a $k \times k$ diagonal matrix containing the k-largest singular values σ_i of $\hat{A}$, and U_k and V_k are respectively $n \times k$ and $l \times k$ matrices whose columns are the corresponding left and right singular vectors. The resulting operator is the highest-norm rank-k linear map from $L^2(M, \mu)$ to H^*, whose kernel is the orthogonal complement of $\bigoplus_{i=1}^{r} \varPhi_i$ in $L^2(M, \mu)$.

Step (4) involves computing the reconstructed data $\tilde{X}_t$ in embedding space via the inverse transform [cf. Eq. (5)]

$$(\tilde{X}_t)_i = \sum_{j=1}^{l} \tilde{A}_{ij} \phi_j(X_t). \tag{8}$$

Finally, in step (5), $\tilde{X}_t$ is projected to d-dimensional physical space by writing

$$\tilde{X}_t = (\hat{x}_{t,0}, \hat{x}_{t,\delta_t}, \dots, \hat{x}_{t,(q-1)\,\delta_t}), \tag{9}$$

and taking the average,

$$\tilde{x}_t = \sum_{t',\tau : t'-\tau=t} \hat{x}_{t',\tau}/q. \tag{10}$$

Note that if M is embedded as an ellipsoid in H, then a set of (possibly degenerate) Laplace-Beltrami eigenfunctions will give the projections of X_t on the principal axes of the ellipsoid; i.e., the system trajectory $\phi_i(X_t)$ in the eigenfunction-based coordinates will be equivalent to the right singular vectors in SSA.

3. Modes of variability in the North Pacific sector of CCSM3

We apply the method presented above to study variability in the North Pacific sector of CCSM3; specifically, variability of the mean upper 300 m sea temperature field in the 700-year equilibrated control integration used by Teng and Branstator [26] and Branstator and Teng [7] in work on the initial and boundary-value predictability of subsurface temperature in that model. Here, our objective is to diagnose the prominent modes of variability in a time series generated by a coupled general circulation model.

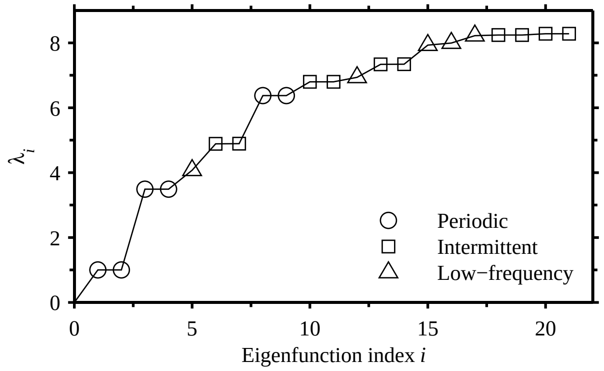


FIGURE 1. Eigenvalues λ_i of the graph Laplacian Δ for the periodic, intermittent, and low-frequency states. Here, we have defined Δ as a positive semidefinite operator, which means that the eigenvalues are non-negative, and obey the ordering $0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \cdots$. Moreover, we have normalized the first non-trivial eigenvalue, λ_1, to unity, since multiplication of the λ_i by the same constant can be absorbed by rescaling the Riemannian metric g.

In this analysis, the x_t observable is the mean upper 300 m temperature field sampled every month at $d = 534$ gridpoints (native ocean grid mapped to the model's T42 atmosphere) in the region 20°N–65°N and 120°E–110°W. Throughout, we work with a two-year embedding window; i.e., the dimension of embedding space is $n = d \times 24 = 12{,}816$. For the calculations of the Laplacian eigenvalues and eigenvectors we used the Diffusion Map algorithm of Coifman and Lafon [10].

Figures 1 and 2 show representative eigenvalues and eigenfunctions of the graph Laplacian. Since we are interested in studying temporal evolution processes, we display the eigenfunctions graphically as plots of $\phi_i(X_t)$ versus t, and also show the corresponding Fourier power spectra. Moreover, to study the spatial patterns associated with the eigenfunctions, we have performed temperature field reconstructions by applying the inverse transform in Eq. (8) with $\tilde{A}_{ij}$ replaced by the operator components A_{ij} from Eq. (5) corresponding to each invariant subspace $\mathit{\Phi}_j$. Figure 3 shows reconstructions based on the eigenfunctions of Figure 2.

Carrying out this procedure systematically for several (~ 100) of the eigenfunctions, we find that they fall into three distinct families of periodic, low-frequency, and intermittent modes, described below. Note that embedding [step (1)] is essential to the separability of the eigenfunctions into these processes; the character of the eigenfunctions is mixed if no embedding is performed.

3.1. **Periodic modes.** The periodic modes come in doubly-degenerate pairs (see Figure 1), and have the structure of sinusoidal waves with phase difference $\pi/2$ and frequency equal to integer multiples of 1 year^{-1}. The leading periodic modes, ϕ_1 and ϕ_2, represent the seasonal cycle in the data. In the physical (spatial) domain [Figure 3(b)], these modes generate an annual oscillation of the temperature anomaly, whose amplitude is largest ($\sim$ 1°C) in the western part of the basin ($\sim$ 130°E–160°E) and for latitudes in the range 30°N–45°N. Together with the higher-frequency overtones, the modes in this family are the standard eigenfunctions of the Laplacian on the circle, suggesting that the data manifold M has the geometry of a circle along one of its dimensions.

3.2. **Low-frequency modes.** The low-frequency modes are characterized by high spectral power over interannual to interdecadal timescales, and strongly suppressed power over annual or shorter time scales. As a result, these modes represent the low-frequency variability of the upper ocean, which has been well-studied in the North Pacific sector of CCSM3 [1, 26]. The leading mode in this family [ϕ_5; see Figure 2(b)], gives rise to a typical PDO pattern [Figure 3(c)], where the most prominent basin-scale structure is a horseshoe-like temperature anomaly pattern developing eastward

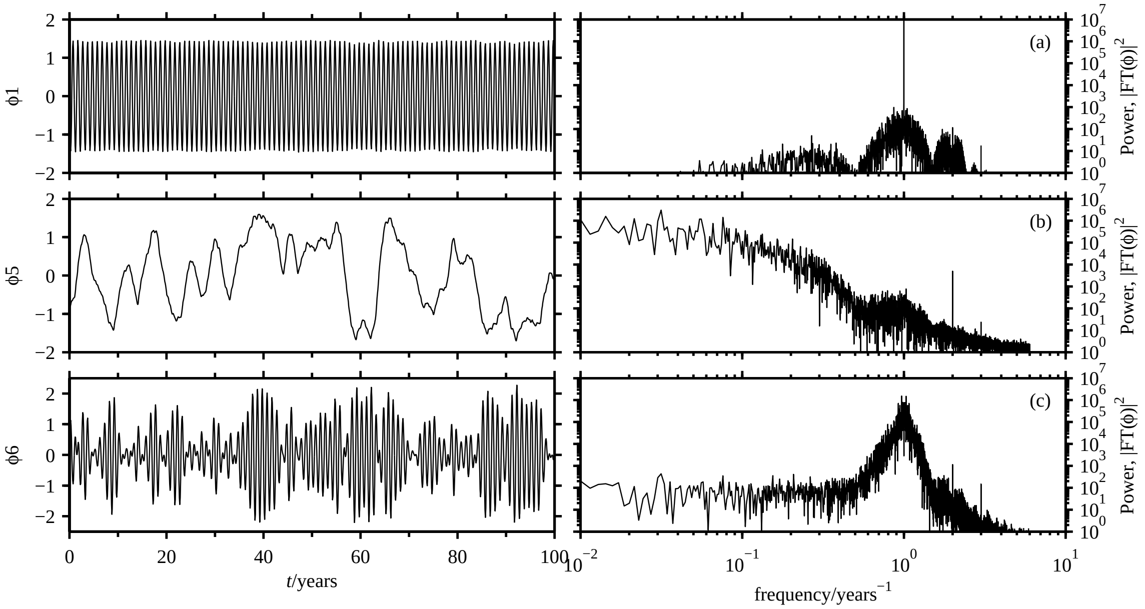


FIGURE 2. Eigenfunctions of the graph Laplacian corresponding to the eigenvalues from Figure 1 plotted in the temporal (left-hand panels) and frequency domains (right-hand panels). (a) Seasonal eigenfunction ϕ_1. (b) First low-frequency eigenfunction, ϕ_5. (c) First intermittent eigenfunction, ϕ_6.

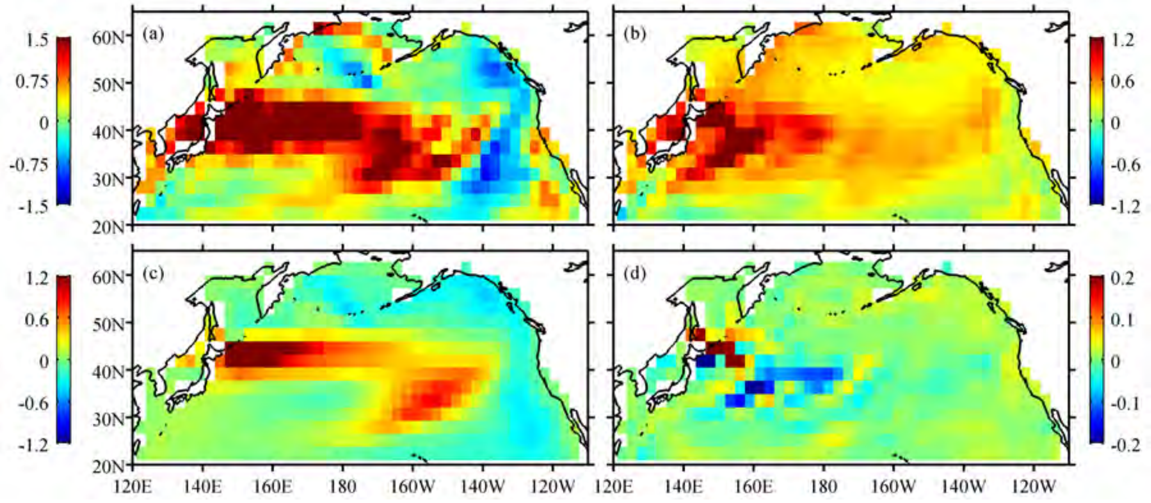


FIGURE 3. Reconstructions of the upper 300 m temperature anomaly field (annual mean subtracted at each gridpoint). Panel (a) shows the raw data in November of year 91 of Figure 2. Panels (b–d) display reconstructions using (b) the seasonal eigenfunctions, ϕ_1 and ϕ_2; (c) the first low-frequency eigenfunction, ϕ_5, describing the PDO; (d) the first two-fold degenerate set of intermittent eigenfunctions, ϕ_6 and ϕ_7, describing variability of the Kuroshio extension.

along the Kuroshio extension, together with an anomaly of the opposite sign along the west coast of North America. The higher modes in this family gradually develop smaller spatial features and spectral content over shorter time scales than ϕ_5, but have no spectral peaks on annual or shorter timescales.

3.3. **Intermittent modes.** As illustrated in Figure 2(c), the key feature of modes of this family is temporal intermittency, arising out of oscillations at annual or higher frequency, which are modulated by relatively sharp envelopes with a temporal extent in the 2–10-year regime. Like their periodic counterparts, the intermittent modes form nearly degenerate pairs (see Figure 1), and their base frequency of oscillation is an integer multiple of 1 year^{-1}. The resulting Fourier spectrum is dominated by a peak centered at at the base frequency, exhibiting some skewness towards lower frequencies.

In the physical domain, these modes describe processes with relatively fine spatial structure, which are activated during the intermittent bursts, and become quiescent when the amplitude of the envelopes is small. The most physically-recognizable aspect of these processes is enhanced transport along the Kuroshio extension region, shown for the leading-two intermittent modes (ϕ_6 and ϕ_7) in Figure 3(d). This process features sustained eastward propagation of small-scale, ~ 0.2 °C temperature anomalies during the intermittent bursts. The intermittent modes higher in the spectrum also encode rich spatiotemporal patterns, including retrograde (westward) propagating anomalies, and gyre-like patterns resembling the subpolar and subtropical gyres.

4. Discussion

4.1. **Intermittent processes and relation to SSA.** The main result of this analysis, which highlights the importance of taking explicitly into account the nonlinear structure of AOS data sets, is the existence of intermittent patterns of variability in the North Pacific sector of CCSM3, which are not accessible through SSA. This type of variability naturally emerges by studying the properties of individual invariant subspaces $\varPhi_i$ of Laplace-Beltrami eigenfunctions on the data manifold (e.g., as done in Figure 3), but in order to produce a more accurate reconstruction, the SVD in Eq. (2) must be applied to combine information from several $\varPhi_i$. Here, we apply this procedure to evaluate a rank $k = 30$ reconstruction based on the leading $l = 55$ Laplace-Beltrami eigenfunctions (in order of increasing λ_i), and compare the results with SSA.

As shown in Figure 4, the leading singular values of $\tilde{A}$ from Eq. (7) fall into four distinct families, separated by spectral gaps; viz. $\{\sigma_1, \sigma_2\}$, coupling almost entirely to the annual eigenfunctions, ϕ_1 and ϕ_2; $\{\sigma_3, \dots, \sigma_{12}\}$, dominated by the low-frequency modes in Figure 1 with weak contributions from the intermittent modes; $\{\sigma_{13}, \sigma_{14}\}$; coupling almost entirely to the semiannual modes, ϕ_3 and ϕ_4; $\{\sigma_{15}, \dots, \sigma_{21}\}$, dominated by the intermittent modes with some coupling to the low-frequency modes with high λ_i.

Typical temperature-anomaly patterns associated with these processes are shown in Figure 5. There, the Kuroshio modes of Figure 3(d) become augmented by temperature anomalies developing along the West Coast of North America, and transported westwards at high latitudes or in the sub-tropics. These features, displayed in Figure 5(f), resemble the subpolar and subtropical gyre. The semiannual modes [Figure 5(e)] also exhibit significant amplitude along the West Coast, which is consistent with semiannual variability of the upper ocean associated with the California current [22]. Note that the semiannual modes appear early in the λ_i spectrum of the Laplacian, but their explained variance, as measured by σ_i, is comparatively small. In separate calculations, we have verified that the SVD decomposition of A is qualitatively robust with respect to the number l of Laplacian eigenfunctions used as basis functions for $L^2(M, \mu)$.

A key point brought out by Figures 4 and 5 is that reconstructions based on machine learning are in close agreement with SSA for the annual and low-frequency modes, but intermittent modes have no SSA counterparts. In particular, instead of the qualitatively-distinct families of processes

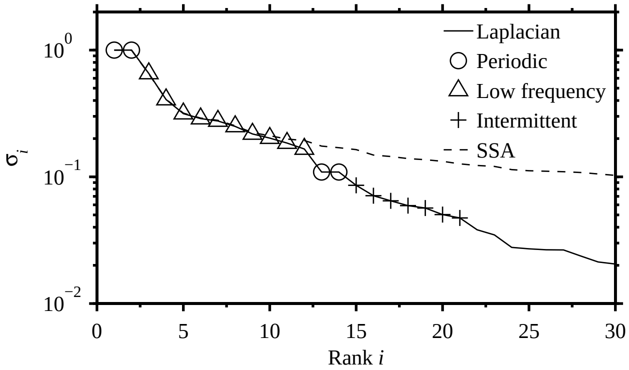


FIGURE 4. Singular values σ_i (normalized so that $\sigma_1 = 1$) evaluated through Laplacian eigenmaps from Eq. (7) (solid line) and SSA (dashed line). The periodic, low-frequency, and intermittent modes indicated here are used in the temperature field reconstructions of Figure 5.

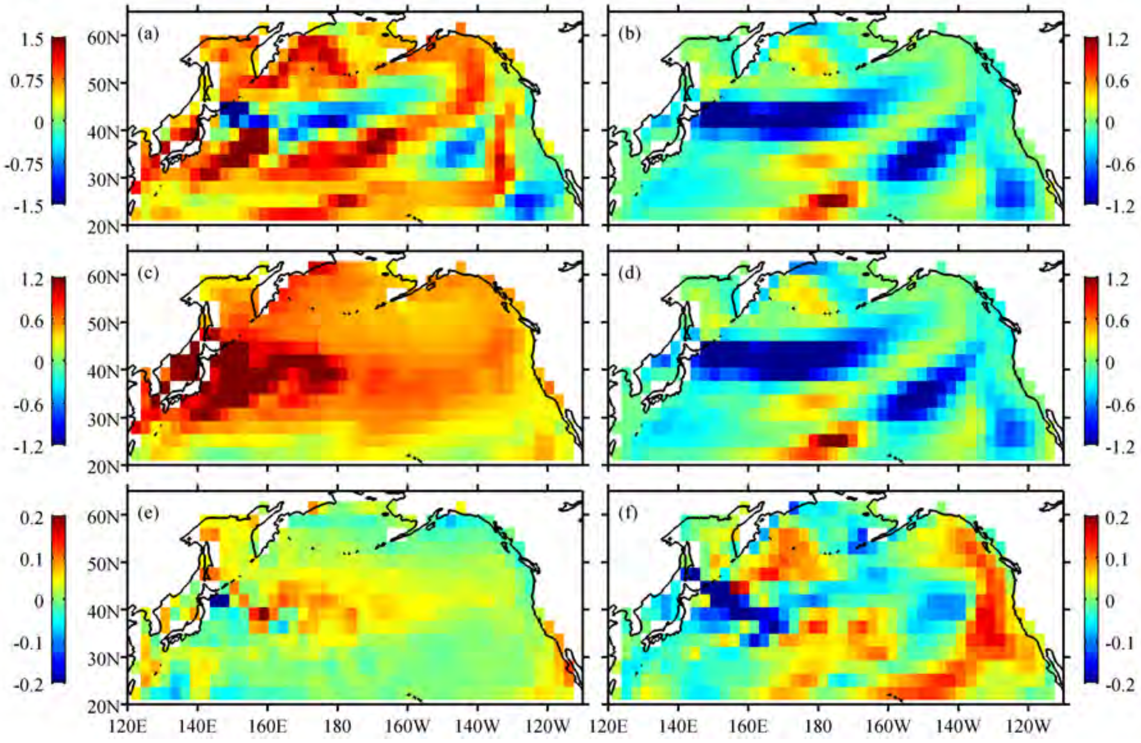


FIGURE 5. Reconstructions of the upper-300 m temperature anomaly field of the 700-year CCSM3 control run through machine learning and SSA. Panel (a) shows the raw data in October of year 45 of Figure 2. Panel (b) displays an SSA reconstruction evaluated using singular vectors (SVs) 3–12 (the low-frequency modes; see Figure 4). Panels (c–f) display reconstructions via Laplacian eigenmaps using (c) the first two SVs of $\tilde{A}$ in Eq. (7) (annual modes); (d) SVs 3–12 (low-frequency modes); (e) SVs 13 and 14 (semiannual modes); (f) SVs 15–21 (intermittent modes).

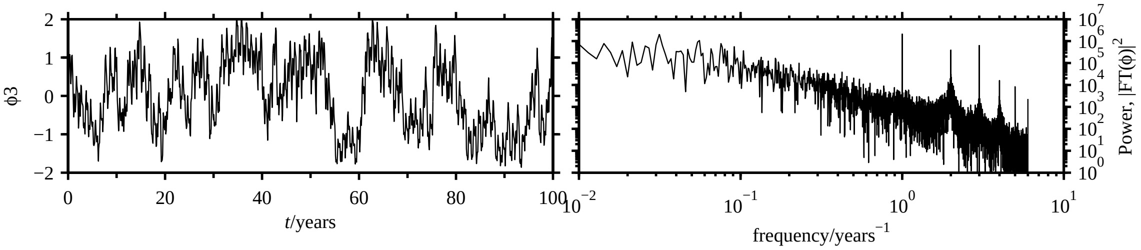


Figure 6. The Laplacian eigenfunction corresponding to the leading "low-frequency" mode evaluated without embedding [cf. Figure (2b)]. Note the pronounced spectral lines with period $\{1, 1/2, 1/3, \dots, 1/6\}$ years.

described above, the SSA spectrum is characterized by a smooth decay involving modes of progressively higher spatiotemporal frequencies, but with no intermittent behavior analogous, e.g., to mode ϕ_6 in Figure 2. Two of the SSA modes exhibit significant semiannual variability, but the frequency content of these modes is not pure, featuring low-frequency beating patterns.

The σ_i values associated with the intermittent modes and, correspondingly, the contributed variance in temperature field reconstructions, is significantly smaller than the periodic or low-frequency modes. However, this is not to say the dynamical significance of these modes is negligible. In fact, intermittent events, carrying low variance, are widely prevalent features of complex dynamical systems [13]. Being able to capture this intrinsically nonlinear behavior constitutes one of the major strengths of the machine-learning based method presented here.

4.2. **The role of lagged embedding.** The embedding in Eq. (1) of the input data x_t in H is essential to the separability of the Laplacian eigenfunctions into distinct families of processes. To illustrate this, in Figure 6 we display the Laplacian eigenfunction that most-closely resembles the PDO mode in Figure 2(b), evaluated without embedding ($q = 1$, $\Delta t = 0$). It is evident from both the temporal and Fourier representations of that eigenfunction that the decadal process recovered in Section 3.2 using a two-year embedding window has been contaminated with high-frequency variability; in particular, prominent spectral lines at integer multiples of 1 year^{-1} down to the maximum frequency of 6/year allowed by the monthly sampling of the data. An even stronger frequency mixing was found to take place in the corresponding temporal SSA modes. In general, representing the dynamical information lost through partial observations via time-lagged embedding, as advocated in the qualitative theory of dynamical systems [23, 25, 8, 24], significantly enhances the quality of time-series reconstructions through either of the machine learning or SSA schemes.

In separate calculations, we have verified that the eigenfunctions separate into periodic, low-frequency, and intermittent processes for embedding windows up to $\Delta t = 10$ years. However, longer embedding windows require more eigenfunctions to produce the same strength of reconstructed signal via Eq. (7).

5. Conclusions

Combining techniques from machine learning and the qualitative theory of dynamical systems, in this work we have presented a scheme for time series reconstruction, which takes explicitly into account the nonlinear geometrical structure of data sets arising in atmosphere-ocean science. Like classical SSA [15], the method presented here utilizes time-lagged embedding and truncated SVD to produce a low-rank reconstruction of time series generated by partial observations of high-dimensional, complex dynamical systems. However, the linear operator used here in the SVD step differs crucially from SSA in that its domain of definition is the Hilbert space of square-integrable functions on the nonlinear manifold M comprised by the data (in a suitable coarse-grained representation via a graph). These functions, analogous to the temporal modes (right singular vectors) in SSA [3], are tailored to the nonlinear geometry of M through its Riemannian measure.

Applying this scheme to the upper-ocean temperature in the North Pacific sector of the CCSM3 model, we find a family of intermittent processes which are not captured by SSA. These processes describe eastward-propagating, small-scale temperature anomalies in the Kuroshio extension region, as well as retrograde-propagating structures at high latitudes and in the subtropics. Moreover, they carry little variance of the raw signal, and display burst-like behavior characteristic of strongly nonlinear dynamics. The remaining identified modes include the familiar PDO pattern of low-frequency variability, as well as annual and semiannual periodic processes.

The nature of the analysis presented here is purely diagnostic. In particular, we have not touched upon the dynamical role of these modes in reproducing the upper ocean dynamics in CCSM3. Here, pertinent open questions are the significance of the intermittent modes in triggering large-scale regime transitions [13], as well as potential improvements of the predictive skill and model error of reduced models utilizing these modes [16, 17]. We plan to study these topics in future work.

Acknowledgments

This work was supported by NSF grant DMS-0456713, ONR DRI grants N25-74200-F6607 and N00014-10-1-0554, and DARPA grants N00014-07-10750 and N00014-08-1-1080. We thank G. Branstator and H. Teng for providing access to the CCSM3 data set used in this analysis.

References

[1] M. Alexander et al. Extratropical atmosphere–ocean variability in CCSM3. *J. Climate*, 19:2496–2525, 2006.

[2] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Wu. An optimal algorithm for approximate nearest neighbor searching. *J. ACM*, 45:891, 1998.

[3] N. Aubry, R. Guyonnet, and R. Lima. Spatiotemporal analysis of complex signals: Theory and applications. *J. Stat. Phys.*, 64(3–4):683–739, 1991.

[4] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput.*, 15(6):1373–1396, 2003.

[5] M. Belkin and P. Niyogi. Towards a theoretical foundation for Laplacian-based manifold methods. *J. Comput. Syst. Sci.*, 74(8):1289–1308, 2008.

[6] P. H. Bérard. *Spectral Geometry: Direct and Inverse Problems*, volume 1207 of *Lecture Notes in Mathematics.* Springer-Verlag, Berlin, 1989.

[7] G. Branstator and H. Teng. Two limits of initial-value decadal predictability in a CGCM. *J. Climate*, 23(23):6292–6311, 2010.

[8] D. S. Broomhead and G. P. King. Extracting qualitative dynamics from experimental data. *Phys. D*, 20(2–3):217–236, 1986.

[9] F. R. K. Chung. *Spectral Graph Theory*, volume 97 of *CBMS Regional Conference Series in Mathematics.* Americal Mathematical Society, Providence, 1997.

[10] R. R. Coifman and S. Lafon. Diffusion maps. *Appl. Comput. Harmon. Anal.*, 21(1):5–30, 2006.

[11] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. Zucker. Geometric diffusions as a tool for harmonic analysis and structure definition on data. *Proc. Natl. Acad. Sci.*, 102(21):7426–7431, 2004.

[12] W. D. Collins et al. The community climate system model version 3 (CCSM3). *J. Climate*, 19:2122–2143, 2006.

[13] D. T. Crommelin and A. J. Majda. Strategies for model reduction: Comparing different optimal bases. *J. Atmos. Sci.*, 61:2206, 2004.

[14] E. R. Deyle and G. Sugihara. Generalized theorems for nonlinear state space reconstruction. *PLoS ONE*, 6(3):e18295, 2011.

[15] M. Ghil et al. Advanced spectral methods for climatic time series. *Rev. Geophys.*, 40(1), 2002.

[16] D. Giannakis and A. J. Majda. Quantifying the predictive skill in long-range forecasting. Part I: Coarse-grained predictions in a simple ocean model. *J. Climate*, 2011. submitted.

[17] D. Giannakis and A. J. Majda. Quantifying the predictive skill in long-range forecasting. Part II: Model error in coarse-grained Markov models with application to ocean-circulation regimes. *J. Climate*, 2011. submitted.

[18] N. Golyandina, V. Nekrutkin, and A. Zhigljavsky. *Analysis of Time Series Structure: SSA and Related Techniques*. CRC Press, Boca Raton, 2001.

[19] M. Latif and T. P. Barnett. Causes of decadal climate variability over the North Pacific and North America. *Science*, 266(5185):634–637, 1994.

[20] M. Latif and T. P. Barnett. Decadal climate variability over the North Pacific and North America: Dynamics and predictability. *J. Climate*, 9:2407–2423, 1996.

[21] N. J. Mantua and S. R. Hare. The pacific decadal oscillation. *J. Oceanogr.*, 58(1):35–44, 2002.

[22] R. Mendelssohn, F. B. Schwing, and S. J. Bograd. Nonstationary seasonality of upper ocean temperature in the California Current. *J. Geophys. Res.*, 109, 2004.

[23] N. H. Packard, J. P. Crutchfield, J. D. Farmer, and R. S. Shaw. Geometry from a time series. *Phys. Rev. Lett.*, 45:712–716, 1980.

[24] T. Sauer, J. A. Yorke, and M. Casdagli. Embedology. *J. Stat. Phys.*, 65(3–4):579–616, 1991.

[25] F. Takens. Detecting strange attractors in turbulence. In *Dynamical Systems and Turbulence, Warwick 1980*, volume 898 of *Lecture Notes in Mathematics*, pages 366–381. Springer, Berlin, 1981.

[26] H. Teng and G. Branstator. Initial-value predictability of prominent modes of North Pacific subsurface temperature in a CGCM. *Climate Dyn.*, 36(9–10):1813–1834, 2010.

[27] K. E. Trenberth and J. W. Hurrell. Decadal atmosphere-ocean variations in the Pacific. *Climate Dyn.*, 9(6):303–319, 1994.

[28] R. Vautard and M. Ghil. Singular spectrum analysis in nonlinear dynamics, with applications to paleoclimatic time series. *Phys. D*, 35:395–424, 1989.

SEMI-SUPERVISED NOVELTY DETECTION WITH ADAPTIVE EIGENBASES, AND APPLICATION TO RADIO TRANSIENTS

DAVID R. THOMPSON[1,2], WALID A. MAJID[2], COLORADO J. REED[2], AND KIRI L. WAGSTAFF[2]

Abstract. We present a semi-supervised online method for novelty detection and evaluate its performance for radio astronomy time series data. Our approach uses adaptive eigenbases to combine 1) prior knowledge about uninteresting signals with 2) online estimation of the current data properties to enable highly sensitive and precise detection of novel signals. We apply the method to the problem of detecting *fast transient* radio anomalies and compare it to current alternative algorithms. Tests based on observations from the Parkes Multibeam Survey show both effective detection of interesting rare events and robustness to known false alarm anomalies.

1. Introduction

Recent discoveries in high time resolution radio astronomy data have drawn attention to a new class of sources. *Fast transients* are rare pulses of radio-frequency energy lasting from microseconds to seconds that might be produced by a variety of exotic astrophysical phenomena [6, 5, 11, 12]. For example, X-ray bursts, neutron stars, active galactic nuclei, and extraterrestrial intelligence (ETI) are all possible sources of short-duration transient radio signals. Such events are often discovered serendipitously in high time resolution data collected for other purposes. These transients are generally faint and difficult to detect, so improved detection algorithms can directly benefit the science yield of all such commensal monitoring. Existing detection approaches rely on a specific *dispersed pulse* model of the signal shape. This paper presents a new method for analyzing real-time high-resolution radio astronomy data that operates without this model assumption. Therefore, it can potentially detect a far broader class of anomalous events in real time, as well as unexpected events that do not match a known profile.

We have formulated fast transient monitoring as statistical anomaly detection in a time series [16, 17]. The main challenges of our domain are:

- **High dimensionality:** Signals of interest span multiple antenna power measurements that could include hundreds of time steps and frequency channels.
- **Real time processing:** With the exception of a few dedicated surveys, most high time resolution data is too voluminous to archive. Therefore, events must be detected in real time to select only the most interesting candidates for storage and later exhaustive analysis.
- **Nonstationarity:** Background noise characteristics change over time on medium to long scales, manifesting as narrow-band noise or large-scale gain fluctuations that can appear and disappear in response to hardware and observing conditions. Detection of anomalous "fast" signals should be robust to these changes.
- **False alarms:** Certain known classes of events, such as momentary Radio Frequency Interference (RFI), are not astronomically interesting but are easily mistaken for fast transients. It is important to avoid flagging these events as novel to avoid filling the detection buffer with these false alarm events. Further, false alarms waste valuable astronomer time in reviewing the results.

[1] david.r.thompson@jpl.nasa.gov

[2] Jet Propulsion Laboratory, California Institute of Technology.

This work proposes a new solution that learns a low-dimensional linear manifold for describing the "normal" data. The novelty of our approach lies in combining basis vectors learned in an unsupervised, online fashion from the data stream with supervised basis vectors learned in advance from known false alarms. We thereby achieve adaptive, data-driven anomaly detection that also exploits prior domain knowledge about signals that may be statistically anomalous but are not interesting and should therefore be ignored. We identify truly interesting anomalies by compressing and reconstructing the data [9] using the combined basis. High reconstruction error indicates a signal that does not match the learned profile of the normal data. The unsupervised component uses the incremental method of Lim et al. [13], an efficient online algorithm that satisfies real-time constraints.

We evaluated the new method using data from the Parkes Multibeam Survey. This data set was originally collected to search for pulsars, which are astronomical sources that emit radio pulses at regular periods. However, several non-pulsar anomalies have recently been discovered in this dataset [3], making it a compelling test case. We found that by explicitly filtering known false alarm patterns, our approach yields significantly better performance than current transient detection methods. This method shows promise for use in current and future astronomical surveys, including data to be collected by the Square Kilometre Array, a radio telescope currently under development that will be 50 times more sensitive than any existing instrument.

2. Related Work

Generic approaches to anomaly detection are data-driven: they typically learn a representation of the "normal" or uninteresting data, then identify any observations that do not match this model. One such method is one-class support vector machine (SVM) classification [18], in which an SVM is trained only on examples from the normal class and then detects any new data belonging to a different, previously unobserved class. More recent efforts seek to include user-labeled examples. Blanchard et al. [2] propose a semi-supervised technique that trains a classifier using two kinds of data: labeled data known to be normal and an additional unlabeled sample that may contain anomalous data. Both approaches aim to train a binary classifier that labels new items as either "normal" or "anomalous." The Blanchard technique further accommodates an upper limit on the false anomaly detection rate. Our approach differs from these methods in that it specifically incorporates known examples of false alarms to further improve the system's precision. In addition, our system is designed for online operation rather than batch processing of previously collected data.

In contrast with statistical novelty detection, radio astronomers generally use physical models of the anticipated events. If the precise shape of the event is known in advance, matched filtering provides maximum sensitivity to detect faint transient pulses. These models reflect the fact that signals from remote astronomical sources are *dispersed*. As the signal travels through the interstellar medium that lies between the source and the observer, it encounters free electrons that absorb some of the signal's energy and delay its propagation. This affects lower frequency components more than higher frequency components. The slight difference accumulates over long distances and ultimately causes a broadband signal to appear dispersed in time, so that the lower frequency components arrive later.

Real-time transient detection uses *incoherent* radio array data organized as a matrix of discretely channelized measurements. In other words, observations occur across a range of radio frequency channels at each time step. This data is typically portrayed as a two-dimensional image in which the axes correspond to time and frequency. The pixel intensity shows observed power, the accumulated squared voltage received by the antenna. Figure 1 (left) shows a pulse from pulsar J0742-2822 that displays a typical dispersed "sweep." Dispersion manifests as a time delay Δt_{delay} that is inversely proportional to the signal's frequency. Following [14]:

$$\Delta t_{\text{delay}} = 4.1\text{ms}\frac{DM\ k}{\Delta \nu^2_{\text{GHz}}} \tag{1}$$

Here $\Delta\nu$ is the frequency difference. The amount of dispersion, or the Dispersion Measure (DM), correlates with the number of interfering electrons present between the source and the observer [1]. It is commonly reported in parsecs per cm^3. For regions of constant electron density, the amount of dispersion suggests the physical distance to the source.

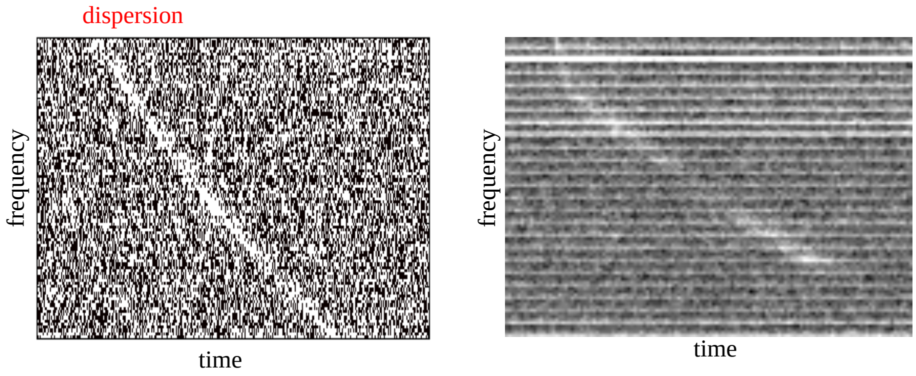


FIGURE 1. Examples of typical and atypical transient signals. The image at left shows a single pulse from pulsar J0742-2822, with a classic dispersed pulse profile. Such signals can be found by inverting the dispersion effect prior to matched filtering. More exotic and poorly understood phenomena, like the *peryton* signal pictured at right, do not match typical dispersion and could benefit from model-free detection strategies with fewer assumptions. This example shows a distinctive "kink" in the curved signal. The narrow horizontal lines are narrow-band interference; such behavior is time-variable but not astronomically relevant and would ideally not affect the detection decision.

The most common approach to detecting remote transient signals is tailored to the known properties of dispersion. Data is exhaustively *dedispersed* using a variety of different candidate DMs [1, 4]. Dedispersion re-aligns the observations in time to undo the effects of a given assumed DM, and then sums the resulting signal across all frequency channels to yield a detection statistic. This tailored summation is equivalent to a matched filter, and increases detection sensitivity over a naive sliding window detection using all frequency channels. By seeking the maximum dedispersed sum across all DMs, one can characterize the signal (and roughly the distance to the source). A dedispersion search can also help separate genuine astronomical signals from Radio Frequency Interference (RFI). Broadband RFI manifests as a vertical signal with no dedispersion (DM = 0); the pulse originates locally and all frequencies arrive simultaneously.

This approach has proven effective for the detection of pulsars and other astronomical phenomena [7, 11, 6, 12]. It can be implemented efficiently to keep up with streaming data using FPGAs, GPUs, or other parallel architectures; dedispersion over multiple DMs is inherently highly parallelizable. The weakness of this approach, however, is that it is sensitive to only one kind of signal. While dispersion is a known phenomenon of all remote signals, some recently-discovered sources (Figure 1, right) exhibit deviations from the expected shape which renders them difficult to detect. Further, it is not known how many other exotic source types may currently be overlooked due to the detection method's dependence on one kind of signal model. The next section presents a more flexible strategy that could operate in parallel with dedispersion searches, providing the capability to detect both dispersed pulses and unanticipated novel events.

3. Approach

We propose a new approach that combines 1) prior knowledge about uninteresting signals with 2) online estimation of the current data properties to enable flexible detection of novel signals. We treat the data as a sequence of observations that arrive sequentially from the antenna. We combine n such observed data points $\mathbf{x}_i \in \mathbb{R}^d$ as columns of a $d \times n$ data matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots \mathbf{x}_n]$. Here, d is the number of frequency channels observed at each time step. The goal is to compute a discriminant function that maps each observation to a novelty score, $f(\mathbf{x}_i) : \mathbb{R}^d \mapsto \mathbb{R}$. The discriminant value should be small for typical data but large for interesting or novel data.

3.1. **Constructing an eigenbasis.** We exploit a popular strategy, detailed in [19] and [9], of measuring the distance from the signal to a low-dimensional manifold learned from the data stream. We will start by describing the simpler case of novelty detection in a static (non-adaptive) subspace. We hypothesize that the "regular" data lies on a linear subspace in $\mathbb{R}^{d'}$ with $d' \ll d$. Subtracting the data mean $\bar{\mathbf{x}}$ yields a translated matrix $\tilde{\mathbf{X}} = [(\mathbf{x}_1 - \bar{\mathbf{x}}), (\mathbf{x}_2 - \bar{\mathbf{x}}), \dots, (\mathbf{x}_n - \bar{\mathbf{x}})]$. Singular Value Decomposition (SVD) provides $\tilde{\mathbf{X}} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$. The columns of $\mathbf{U}$ are the *principal components*: an orthonormal basis with axes in the order of decreasing data variance. We form a low-dimensional basis $\mathbf{A}$ using the first d' columns of $\mathbf{U}$. When $n > d$ the SVD decomposition is undefined, but one can still compute the matrix $\mathbf{A}$ via classical Principal Component Analysis (PCA), e.g., using the eigenvectors corresponding to the largest eigenvalues of the covariance matrix $\tilde{\mathbf{X}}\tilde{\mathbf{X}}^T$.

We quantify the novelty of observation $\mathbf{x}_i$ using the Euclidean distance to the subspace, equivalent to the L2-norm *reconstruction error* after first transforming $\mathbf{x}_i$ into the low-dimensional basis and then reconstructing an approximation $\hat{\mathbf{x}}_i$. This suggests the following discriminant function which is large for novel data and zero for points on the linear manifold.

$$f(\mathbf{x}_i) = \|\mathbf{x}_i - \hat{\mathbf{x}}_i\| = \|(\mathbf{x}_i - \bar{\mathbf{x}}) - \mathbf{A}\mathbf{A}^T(\mathbf{x}_i - \bar{\mathbf{x}}))\|_2 \tag{2}$$

The eigenvalue decomposition makes computing $\mathbf{A}$ difficult for large n. However, it is important that our basis accommodate large data sets and long-timescale changes in the background. One solution is to periodically recompute the entire matrix $\mathbf{A}$ in batch mode using a recent subset of the data. In this work we favor the approach of Lim et al. [13] for efficient online updates to the mean $\bar{\mathbf{x}}$ and eigenbasis $\mathbf{A}$. This approach updates an SVD decomposition defined by some previous data $\tilde{\mathbf{X}}_p = \mathbf{U}_p\mathbf{\Sigma}_p\mathbf{V}_p^T$. Each block update has a data matrix $\mathbf{X}_q$ with mean $\bar{\mathbf{x}}_q$ and decomposition $\tilde{\mathbf{X}}_q = \mathbf{U}_q\mathbf{\Sigma}_q\mathbf{V}_q^T$. This gives a combined dataset $\mathbf{X}_r = [\mathbf{X}_p|\mathbf{X}_q]$. Fortunately one can compute an updated mean $\bar{\mathbf{x}}_r$ and eigenbasis $\tilde{\mathbf{X}}_r = \mathbf{U}_r\mathbf{\Sigma}_r\mathbf{V}_r^T$ without having to store the old data explicitly. We refer the reader to the Lim et al. text for details, but summarize their approach in Algorithm 1 below. It relies on the widely-studied R-SVD procedure [8] which exploits the fact that a low-rank update to the eigenbasis is decomposable into efficient block operations. The method extends R-SVD to the case where the data are not assumed to have zero mean.

Algorithm 1: Lim et al. Algorithm for Sequential Eigenbasis Updates.

```
Input: Previous mean $\bar{\mathbf{x}}_p$
       Previous decomposition $\mathbf{U}_p\mathbf{\Sigma}_p\mathbf{V}_p^T$
       Additional data $\mathbf{X}_q$
Output: Revised mean $\bar{\mathbf{x}}_r$
       Revised decomposition $\mathbf{U}_r\mathbf{\Sigma}_r\mathbf{V}_r^T$
Compute $\bar{\mathbf{x}}_r = \frac{n}{n+m}\bar{\mathbf{x}}_p + \frac{m}{n+m}\bar{\mathbf{x}}_q$, where $n = |\mathbf{X}_p|$ and $m = |\mathbf{X}_q|$
Compute $\mathbf{E} = \left(\mathbf{X}_q - \bar{\mathbf{x}}_r 1_{(1\times m)} | \sqrt{\frac{nm}{n+m}}(\bar{\mathbf{x}}_p - \bar{\mathbf{x}}_q)\right)$
Use $\mathbf{U}_p\mathbf{\Sigma}_p\mathbf{V}_p^T$ with $\mathbf{E}$ as input to R-SVD to compute $\mathbf{U}_r\mathbf{\Sigma}_r\mathbf{V}_r^T$
```

An advantage of the Lim et al. method is that one can downweight the old basis to introduce a forgetting factor that allows the influence of old data to decay gradually as new points are added.

This lets the basis shift to track a nonstationary distribution, and it accommodates observations of arbitrary length.

3.2. **Semi-supervised eigenbases.** Automated novelty detection may need to exclude certain rare events that are known in advance to be uninteresting. For example, there may be known *false alarms* due to rare but recognizable noise. Another case where false alarms can be anticipated in advance is through feedback from a human user based on follow-up processing of previous results. This feedback could determine which previous detections had been erroneous. We incorporate information about known false alarm patterns with a second basis trained to model these known examples. Our semi-supervised novelty detection method uses a combined subspace with both supervised and unsupervised components, shifting with long-term trends while still excluding the false alarms. Algorithm 2 summarizes the procedure. We train an initial low-dimensional basis $\mathbf{U}_s$ using data known to be uninteresting. At runtime, we compute an adaptive mean $\bar{\mathbf{x}_r}$ and basis $\mathbf{U}_r$ using the Lim et al. method, and also define a combined basis $\mathbf{U}_c = [\mathbf{U}_r|\mathbf{U}_s]$ to span both the supervised basis and the unsupervised data. We orthogonalize the new basis using QR decomposition with the Gram-Schmidt method. The reconstruction error with respect to the combined basis yields a more reliable novelty score.

Algorithm 2: Semi-Supervised Eigenbasis Novelty Detection.

```
Input: Supervised training data $\mathbf{X}_s$ of size $l$
        Size $m$ block updates of streaming, unsupervised data $\mathbf{X}_u$
Output: Novelty scores $f(\mathbf{x}_1), f(\mathbf{x}_2), \ldots$
Compute $\tilde{\mathbf{X}}_s = [\mathbf{x}_{s1} - \bar{\mathbf{x}}_s,\ \mathbf{x}_{s2} - \bar{\mathbf{x}}_s,\ \ldots, \mathbf{x}_l - \bar{\mathbf{x}}_s]$
Use PCA with $\tilde{\mathbf{X}}_s\tilde{\mathbf{X}}_s^T$ or SVD with $\tilde{\mathbf{X}}_s = \mathbf{U}_s\mathbf{\Sigma}_s\mathbf{V}_s^T$ to compute an orthonormal basis $\mathbf{U}_s$
Using the first block $\mathbf{X}_{u1}$, compute an initial mean $\bar{\mathbf{x}}_p$ and eigenbasis $\mathbf{U}_p\mathbf{\Sigma}_p\mathbf{V}_p^T$
For each subsequent $\mathbf{X}_u$:
  Compute a revised mean $\bar{\mathbf{x}}_r$ and a revised decomposition $\mathbf{U}_r\mathbf{\Sigma}_r\mathbf{V}_r^T$ using the Lim et al. method
  Define a combined basis $\mathbf{U}_c = [\mathbf{U}_r|\mathbf{U}_s]$
  Use QR decomposition to find a basis $\mathbf{U}'_c$ that makes $\mathbf{U}_c$ orthonormal.
  For each $\mathbf{x}_i \in \mathbf{X}_u$:
    Compute $f(\mathbf{x}_i) = \|(\mathbf{x}_i - \bar{\mathbf{x}}_r) - \mathbf{U}'_c{\mathbf{U}'_c}^T(\mathbf{x}_i - \bar{\mathbf{x}}_r)\|_2$
```

Note that the proposed approach does not preserve the mean of the initial false alarm distribution, which is assumed to drift in a similar fashion as the mean of the dynamic distribution. User feedback would permit a more sophisticated system that also updates the false alarm mean and basis online, but we focus here on the simpler case where all training occurs in advance.

4. Evaluation

This semi-supervised anomaly detection method was motivated by applications in radio astronomy. We performed a comparative evaluation on a test set of radio array data using five linear and nonlinear novelty detection algorithms: the traditional dedispersion approach, kernel PCA novelty detection [9], one-class SVM novelty detection [18], unsupervised adaptive novelty detection using PCA, and the proposed semi-supervised approach.

4.1. **Data set.** We use a selected portion of data from the Parkes Multibeam Survey, an extensive search for Pulsars using the Parkes radio telescope of CSIRO [7, 15, 10]. This instrument views the sky simultaneously through 13 receivers, effectively providing 13 independent antennas covering adjacent, and slightly overlapping, areas in the sky. Receiver measurements are recorded at high time resolution and transformed into channelized power measurements corresponding to the squared voltage response at various discrete frequency channels. This specific data sequence contains examples of events known as *perytons*, first discovered by Burke-Spoloar and Bailes in their analysis of Parkes pulsar surveys [3]. Perytons are still poorly understood, and they are scientifically interesting because they vary in frequency and approximate a dispersion curve. However, they do not exactly match a dispersion profile, and their spatial distribution in the sky suggests that they are of terrestrial (possibly atmospheric) origin.

In addition to these features, structured interference is often visible in the form of channel-specific noise and gain fluctuations appearing as horizontal stripes. Such noise is pervasive and typical for highly sensitive, cryogenically cooled receiver feeds. Our tests focus on approximately five minutes of observation time in each of the 13 receivers. This span includes several tens of thousands of timesteps recorded at a cadence of 0.125 milliseconds in each of 96 frequency channels near 1450 MHz. Figure 2 shows three examples of perytons. The red rectangle shows the size of an example

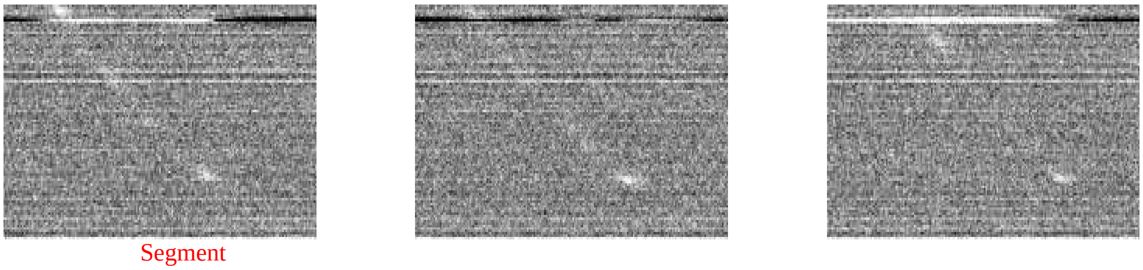


FIGURE 2. True anomalies: Peryton events from the Parkes Multibeam Survey.

Figure 3 shows some examples of false alarms that are statistically uncommon but not scientifically interesting. These specific examples are broadband pulses of Radio Frequency Interference (RFI), probably emitted by some local artificial source. Such features are rare enough that they are not well-represented in an unsupervised eigenbasis, but typical enough that they would dominate novelty detection results if not handled explicitly.

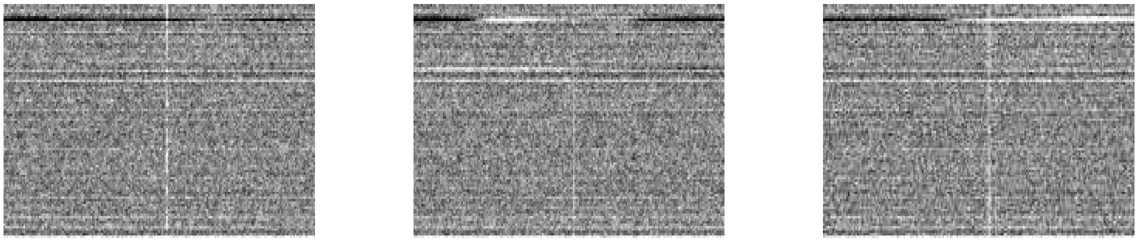


FIGURE 3. False anomalies: Vertical stripes due to broadband RFI that is statistically anomalous but uninteresting.

4.2. **Methodology.** We subsample the data by a temporal factor of 20 so that it has a resolution of 2.5 ms, and then analyze the data as a sequence of short non-overlapping *segments* that cross all 96 vertical frequency channels and 6 horizontal timesteps. This segment width corresponds to a $15ms$ time interval, found empirically to be the best-performing value for all methods. We reorder each segment into a single column vector $\mathbf{x} \in \mathbb{R}^{384}$. Finally, we unify data from all beams into one large dataset, witholding five beams (38%) for training purposes.

We compare five different detection methods that are broadly representative of different linear and nonlinear anomaly detection approaches. First, we consider the proposed semi-supervised method that combines supervised and unsupervised components and reports reconstruction error $f_{ss}(\mathbf{x}_i)$.

Here we trained the subspace $\mathbf{U}_s$ using 30 overlapping segments ($\mathbf{X}_s$) drawn from three manually-selected broadband RFI pulses. The second method is a purely unsupervised eigenbasis approach based on reconstruction error from a low-dimensional basis $f_u(\mathbf{x}_i)$. It does not explicitly account for RFI. The third method is the one-class SVM novelty detection of Scholkopf et al. [18]. Here we use a radial basis kernel function selected with a grid search, and treat each test point's distance to the decision boundary as a real-valued novelty score.

The fourth method is kernel PCA: a non-linear extension of PCA. Kernel PCA novelty detection first maps the data to a higher (generally infinite) dimensional features space, computes the principal components in this space, projects the transformed data to a lower-dimension manifold, and defines a novelty measure as the reconstruction error in the feature-space. Kernel functions allow the reconstruction error to be calculated without explicitly [9]. However, this method never explicitly calculates the principal components so it cannot be used as an adaptive technique in the manner discussed in Algorithm 2. Instead we use the implementation of Hoffmann et al [9]. We use a radial basis kernel function with parameters selected by a grid search.

Finally, we consider a state-of-the-art incoherent dedispersion and summation method which searches DM values from 0 and 500. We correct each time step separately for each DM, use and the maximum response from all DMs as the novelty score $f_d(\mathbf{x}_i)$. Time averaging did not improve performance, so we report the single-timestep result.

In addition to labeling RFI, we obtained the precise locations of all peryton events noted in the study by Burke-Spolaor et al [3]. These appeared to some degree in all antennas, though the signal strength and character varied somewhat even for simultaneous observations. The concatenated dataset provided 88 real novel events for our evaluation. We assigned each peryton an enclosing time interval; any detection in this range counted as having successfully detected the peryton. Note that we use the same time interval for all beams regardless of the actual signal strength. Perytons that are weak in one or more beams penalize all detection methods equally, so we assume events are always present for our (relative) performance assessment.

We evaluated each method by first computing novelty scores for the entire data set, sorting these scores across all beams, and then counting the result of each trigger in order of decreasing novelty. Each peryton can only be captured once, though multiple triggers within the same event do not count as false positives. However, any detection falling outside a peryton interval counts as a false positive.

4.3. **Results.** A visualization of the unsupervised and supervised bases created by our method appears in Figure 4. Here we use 4 principal components as an unsupervised basis with online updates from the data stream. These *eigensignals* (Figure 4, left) show high magnitude in the most variable channels; at the time this eigensignal snapshot was captured, such channels comprised the major axis of variance for the data set. A supervised basis of 10 dimensions models the known broadband RFI. The top five such eigensignals appear in Figure 4, Right; they reflect the vertical profile of momentary RFI pulses. Together, the two can accurately reconstruct a slow shift in channelized RFI conditions along with any additive broadband pulses. This image shows the orthonormal segments after QR factorization.

Figure 5 compares novelty detection scores for the entire observation sequence of the first test beam, computed with a purely unsupervised basis (standard PCA), f_u, and the semi-supervised approach, f_{ss}. Interesting peryton events are noted by black triangles; the other signal spikes correspond to various kinds of RFI. Five peryton signals were barely visible in the reconstruction error of either method, due possibly to the alignment of non-overlapping segments or the inherently weak visibility of those events in this beam. We exclude these five from the diagram for clarity. In general the semi-supervised approach responds to the novelty of peryton events while filtering most of the RFI. In contrast, broadband RFI contaminates the purely unsupervised approach; it accounts for the three strongest responses by f_u for this sequence.

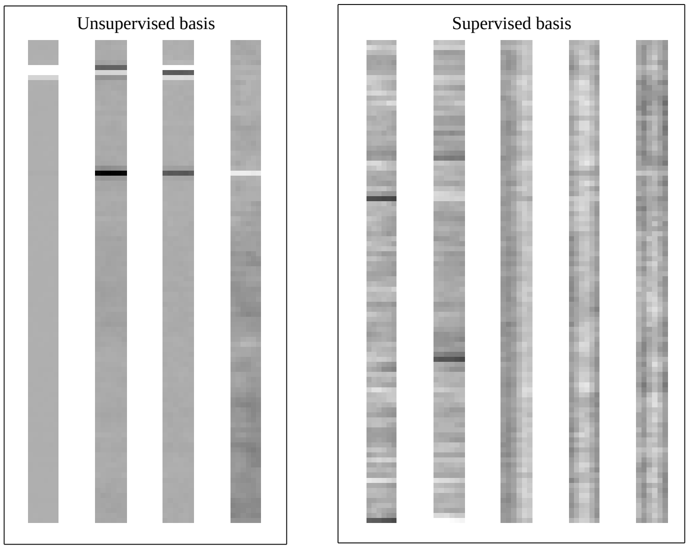


FIGURE 4. Orthonormal principal components used to construct $\mathbf{U}_c$ from $\mathbf{U}_r$ (left) and $\mathbf{U}_s$ (right). The unsupervised portion (left) models channelized interference, while the vertical structures in the supervised portion (right) represent momentary broadband RFI.

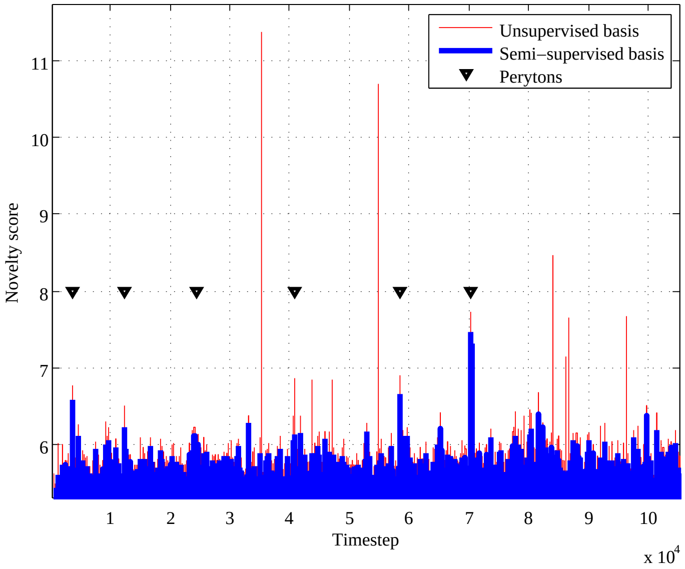


FIGURE 5. Semi-supervised learning filters out RFI events that would otherwise dominate the detection results. This time series plot shows per-timestep novelty evaluated for the first beam in the test set. Not all perytons are clearly distinguishable in this beam.

Figure 6 shows a Receiver Operating Characteristic (ROC) curve describing the tradeoff in precision and recall rates. We report the number of perytons captured for a variety of false positive budgets. For real-time observations, false positive budgets beyond 10 are excessive. Generating more than 10 false positives would represent greater than one detection event for every 5 seconds of observations, imposing an unrealistic burden on manual post-analysis. Future commensal campaigns with constant observations and higher data volumes will demand even stricter limits. For this low error budget, the semi-supervised approach considerably outperforms the competing methods: the top 12 signals detected via f_{ss} are due to perytons, while the kernel PCA technique detects 5 false-positives before the first peryton, and the unsupervised method reports more than 40 false positives before finding the first real peryton. These runner-up methods require 250 and 200 false positives respectively before they match the error-free retrieval rate of the semi-supervised approach. Note that one might improve performance of any method further with additional hand-crafted RFI excision rules, such as a ban on zero-DM detections that are likely to be terrestrial. Naturally, such rules are less general than a learning-based approach and might filter other interesting but unanticipated phenomena.

The preceding results used a data segment size of 15 ms (6 time steps) to compose $\mathbf{x}_i$. We evaluated sensitivity to segment length (see Figure 7). Segments of duration $10 - 15$ ms performed best for this data set. It is possible that smaller segments are susceptible to noise while larger sizes dilute the perytons. It might improve performance for large segments to use a higher-dimensional basis for the unsupervised component. Such models might do a better job of modeling temporal structure (such as switching interference) that begins to appear at these scales.

We also assessed the runtime of each method to determine whether they could be used in a realistic real-time setting. Using a single core of a modern desktop processor, the runtime of the dedispersion search method averaged 0.16 seconds per DM for the entire subsampled sequence, or ≈ 80 seconds for a typical search over 500 DM values. This could be divided easily among multiple processors to provide faster processing for multiple beams. The eigenbasis approaches' runtimes depend strongly on the size of the block updates to the eigenbasis. For a single desktop processor core performing block updates of size $m = 100$, the entire observation from a single beam was processed at $5\times$ real time (≈ 10 seconds/beam for the entire dataset). The time required was slightly larger (up to ≈ 20 seconds/beam) for smaller block updates where constant-time overhead costs had a larger impact. The accuracy of these techniques was nearly indistinguishable for all block update sizes we tried. Varying the segment sizes also affected run time by up to a factor of two. Kernel PCA and one-class SVM performed considerably slower than the dedispersion and eigenbasis approaches as all computations were performed with an RBF kernel representation of the data: a representation of size $|\mathbf{x}|^2 = 384^2$ for this work. In our experiments we found these techniques required $\approx 200 - 400$ seconds/beam with block updates of $m = 400$. Furthermore, unlike the dedispersion and eigenbasis techniques, the Kernel PCA and one-class SVM computation times scale quadratically with the size of m. This reduces the generality of these methods, and in combination with their large computational run-times, makes them unfeasible as real-time techniques. On the other hand, we found the dedispersion and eigenbases approaches to be readily employable for real-time use on general purpose computing hardware.

5. Discussion

Semi-supervised eigenbases have general applicability for anomaly detection in domains with real-time requirements, high-dimensional input, and prior knowledge about false alarm events. Of course, it is not necessary to incorporate this false alarm information directly into the novelty detection model as we have done here. One could perform pre-classification to filter these events prior to a purely unsupervised novelty detection stage. Nevertheless, there may be other advantages to a combined approach. It is simple and easy to implement. The projection shifts to reflect any underlying drift in the mean signal levels, so that a basis trained on previous false alarms remains relevant. Further work will investigate ways to combine multi-scale models when the temporal extent of the interesting events is not known in advance. Finally, application to the broader Parkes survey catalogue will increase practical experience with the technique, and may even reveal additional classes of RFI and astronomical transient events.

6. Acknowledgements

This work was made possible through the assistance of Sarah Burke-Spolaor, formerly of CSIRO and now affiliated with the Jet Propulsion Laboratory. More generally we would also thank Swinburne University and CSIRO for a generous data access policy. We also thank J-P Macquart (CSIRO), and Dayton Jones, Robert Preston and Joseph Lazio (Jet Propulsion Laboratory) whose advice and ideas have influenced our investigation. This research was performed at the Jet Propulsion Laboratory, California Institute of Technology, under the Research and Technology Development Strategic Initiative Program. Copyright 2011 California Institute of Technology. All Rights Reserved. U.S. Government Support Acknowledged.

References

[1] D. Bhattacharya. Detection of radio emission from pulsars. *NATO ASIC Proc. 515: The Many Faces of Neutron Stars*, 1998.

[2] G. Blanchard, G. Lee, and C. Scott. Semi-supervised novelty detection. *Journal of Machine Learning Research*, 11:2973–3009, 2010.

[3] S. Burke-Spolaor, M. Bailes, R. Ekers, J. Macquart, and F. Crawford III. Radio Bursts with Extragalactic Spectral Characteristics Show Terrestrial Origins. *The Astrophysical Journal*, 727:18, 2011.

[4] J. Chennamangalam et al. Software data-processing pipeline for transient detection. *The Low-Frequency Radio Universe, ASP. Conf. Series*, LFRU, 2009.

[5] J. M. Cordes et al. The dynamic radio sky. *New Astronomy Reviews*, 48:1459–1472, 2004.

[6] J. M. Cordes and M. A. McLaughlin. Searches for fast radio transients. *The Astrophysical Journal*, 596:1142–1154, October 2003.

[7] R. T. Edwards, M. Bailes, W. van Straten, and M. Britton. The Swinburne Intermediate Latitude Pulsar Survey. *MNRAS*, 326:358–374, 2001.

[8] G. Golub and C. Van Loan. *Matrix Computations*. Johns Hopkins Univ Pr, 1996.

[9] H. Hoffmann. Kernel PCA for novelty detection. *Pattern Recognition*, 40(3):863–874, 2007.

[10] B. A. Jacoby, M. Bailes, S. M. Ord, R. T. Edwards, and S. R. Kulkarni. A large-area survey for radio pulsars at high galactic latitudes. *The Astrophysical Journal*, 699:401–411, 2009.

[11] E. Keane. The search for nearby RRATs and other transient radio bursts. In *Third Estrela Workshop*, 2008.

[12] J. Lazio, J. S. Bloom, G. C. Bower, J. Cordes, S. Croft, S. Hyman, C. Law, and M. McLaughlin. The dynamic radio sky: An opportunity for discovery. *Astro2010: The Astronomy and Astrophysics Decadal Survey White Paper no. 176*, 2009.

[13] J. Lim, D. Ross, R. Lin, and M. Yang. Incremental learning for visual tracking. *Advances in Neural Information Processing Systems*, 1:793–800, 2004.

[14] A. G. Lyne and F. G. Graham-Smith. *Pulsar Astronomy.* Cambridge University Press, 1998.

[15] R. N. Manchester et al. Parkes multibeam pulsar survey: I. observing and data analysis systems, discovery and timing of 100 pulsars. *MNRAS*, 328:17–35, 2001.

[16] M. Markou and S. Singh. Novelty detection: a review. Part 1: statistical approaches. *Signal Processing*, 83(12):2481–2497, 2003.

[17] M. Markou and S. Singh. Novelty detection: a review. Part 2: neural network based approaches. *Signal Processing*, 83(12):2499–2521, 2003.

[18] B. Schöolkopf, J. Platt, J. Shawe-Taylor, A. Smola, and R. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7):1443–1472, 2001.

[19] S. O. Song, D. Shin, and E. S. Yoon. Analysis of novelty detection properties of auto-associators. In *Proceedings of COMADEM*, pages 577–584, 2001.

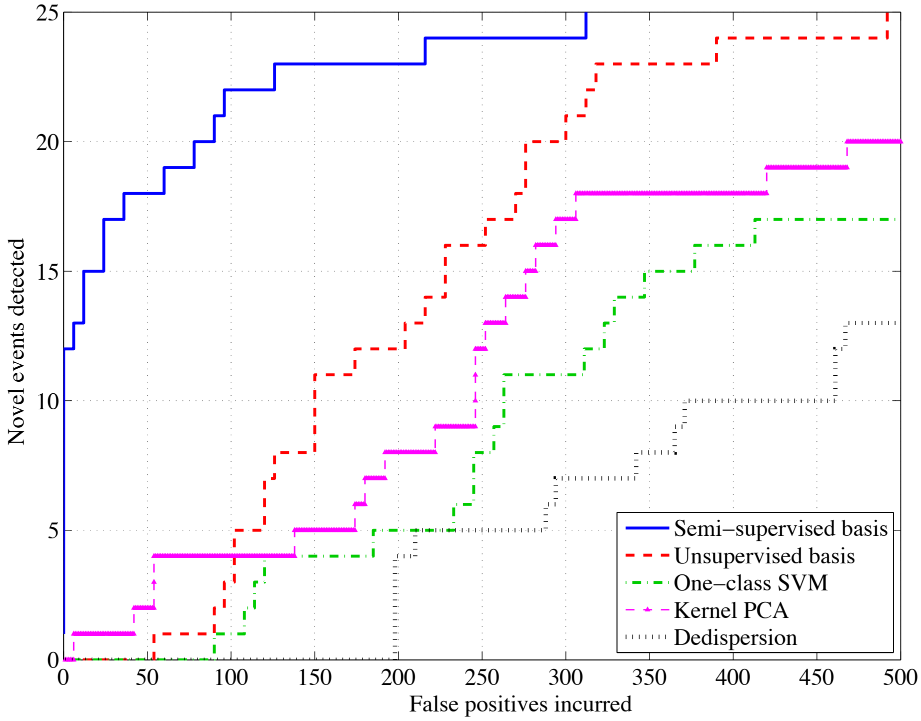


FIGURE 6. ROC curves comparing eigenbasis novelty detection approaches with the traditional dedispersion search.

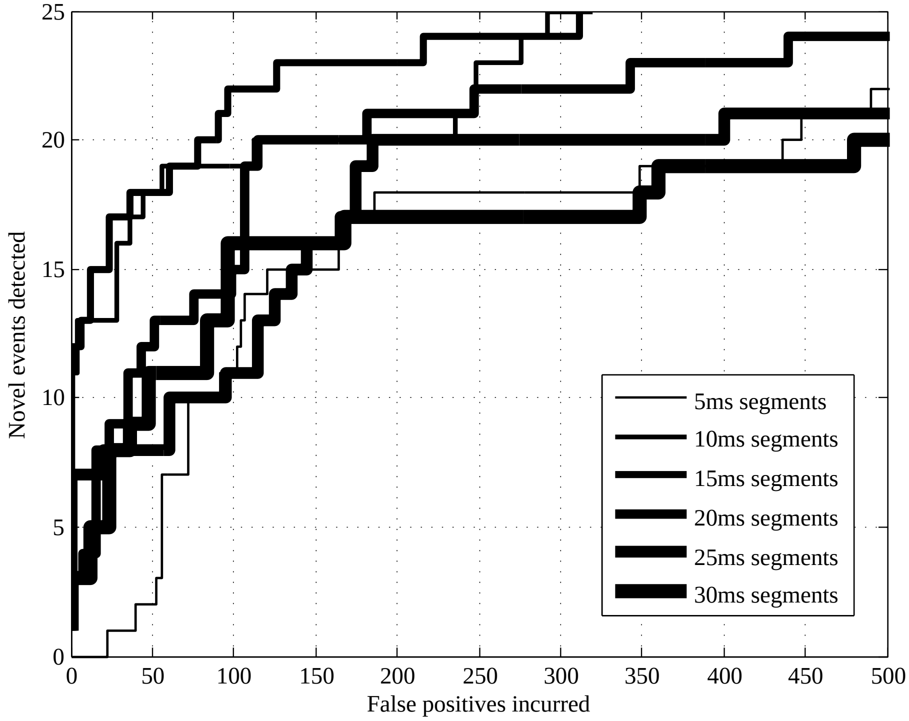


FIGURE 7. ROC curves to assess the sensitivity of semi-supervised detection to data segment sizes.

STATISTICAL INFERENCE BASED ON DISTANCES BETWEEN EMPIRICAL DISTRIBUTIONS WITH APPLICATIONS TO AIRS LEVEL 3 DATA

DUNKE ZHOU* AND TAO SHI**

Abstract. Atmospheric Infrared Sounder (AIRS), a sensor aboard NASA's Aqua satellite, has been collecting temperatures, water vapor mass-mixing ratios, cloud fractions at various atmosphere pressure levels, and other atmospheric observations. AIRS level 2 data has a 45 km ground footprint with global coverage. The AIRS level 3 Quantization (L3Q) product summarizes valid level 2 data in each $5^o \times 5^o$ latitude-longitude grid box during a time period by a set of representative vectors and their associated weights, which can be treated as an empirical distribution. In this paper, we study potential statistical tools using pairwise dissimilarities that are suitable for analyzing this nontraditional type of data. Through theoretical analysis and simulations, we investigate several different dissimilarity measures and find Mallows distance is preferable over others when the locations of the representative vectors are important for the analysis. We apply MultiDimensional Scaling and clustering method to analyze AIRS data collected in December 2002. The results from these studies provide insights on how statistical methods based on Mallows distance may extract more information from the AIRS L3Q data than from the simple sample average summary in each grid box.

1. Introduction

In recent years, scientists working on climate change have seen the explosion of the volume and complexity of available data, both from remote sensing satellites of the National Aeronautics and Space Administration (NASA) and from the output of massive climate model simulations. To reduce the massive data size, data are first divided into subsets by spatial location, time period, or other grouping variables. Then summary statistics, such as sample average and standard deviation, are used to represent each subset, and further analysis is applied to these summaries. This approach works reasonably well when the data distributions in different subsets have similar shapes. However, any information beyond the first two moments of the empirical distribution is lost in this initial data reduction step.

Going beyond the averages and standard deviations, a data reduction method based on clustering was suggested in [4, 5]. After clustering multivariate data vectors in one subset into groups, the data are summarized by a set of representative vectors (mean of each cluster: $m_1, \ldots, m_L$) and their associated weights or probabilities $(w_1, \ldots, w_L)$. In this way, the data in the i-th subset are represented by $S_i \equiv \{m_{i,1}, \ldots, m_{i,L_i}; w_{i,1}, \ldots, w_{i,L_i}\}$. While dramatically reducing the data size, the data summary S_i, like a multivariate empirical distribution, still keeps most information of the joint distribution of the data vectors in the subset.

This strategy has been adopted by the Atmospheric Infrared Sounder (AIRS) project. AIRS collects temperatures, water vapor mass-mixing ratios, cloud fractions, at various atmosphere pressure levels at each 45 km ground footprint in its level 2 data. The level 3 Quantization (L3Q) products summarize valid level 2 data vectors in each $5^o \times 5^o$ latitude-longitude grid box during a month or a season by a list of representative vectors and their frequencies produced from a K-means clustering algorithm. As a result, each summary

*Department of Statistics, The Ohio State University zhou.208@buckeyemail.osu.edu

*Department of Statistics, The Ohio State University taoshi@stat.osu.edu.

S_i represents a record of the regional climate during the period of time. The AIRS L3Q products are publicly available at `http://disc.sci.gsfc.nasa.gov/AIRS/data-holdings` and more details can be found in [4, 5].

Although the clustering based data summary preserves much more distributional information than sample averages, statistical analysis tools for this new type of data summary are not widely available. Traditional multivariate statistical methods have been developed mainly to deal with data vectors in an Euclidean space. Geometry and linear projections play significant roles in those classical methods such as Principal Component Analysis, Factor Analysis, Linear Regression, Canonical Correlation Analysis, et al. However, these methods cannot be directly applied to these clustering based data summaries, since the concepts of directions and projections are not well defined for this type of data.

In recent years, analysis methods using dissimilarity or distances between each pair of observations (vectors) have drawn attentions in Statistic and Machine Learning. Examples include MultiDimensional Scaling (MDS, [7]), Kernel Principal Components Analysis [23], Spectral clustering [28, 24, 19, 26] and estimation [25], manifold learning [1, 20, 27]. In principle, this class of methods is more suitable to be applied to the clustering based data summaries if the pairwise dissimilarity measure is properly chosen. In an initial study reported in [6], MultiDimensional Scaling based on Mallows distances [18] was applied to the AIRS L3Q data to display the year to year variation of local climates. It was illustrated that the leading second MDS dimension may connect to certain physical features of the global climate. Mallows distance has also been employed to help visualize the particle collisions in particle accelerator experiments as in [11] and to analyze univariate histograms as in [13].

Besides the Mallows distance, several dissimilarity measures between distributions have been studied in Statistics and Information Theory. These measures include Hellinger distance, Kullback-Leibler (K-L) divergence, and χ^2 divergence. In this paper, we will investigate the appropriateness of applying statistical methods using certain dissimilarity measures. In particular, this paper concentrates on (1) characterizing and summarizing the properties of these dissimilarity measures and (2) investigating the properties of statistical analysis based on different dissimilarity measures and potential difference among their results.

In Section 2, we review several dissimilarity measures including Mallows distance and f-divergence, where Hellinger distance, K-L divergence, χ^2 divergence are special cases of the f-divergence. The properties of Mallows distance are summarized and a brief comparison between Mallows Distance and f-divergence is included. Mallows distance is found to be good at reflecting the difference in support-dependent features of distributions such as mean and variance. Moreover, Mallows distance has advantage over f-divergence when applied to discrete distributions with potential different supports, such as the AIRS L3Q data. On the other hand, f-divergences, especially K-L divergence, have natural interpretation in information theory. K-L divergence is preferred when such interpretation is desired, such as in predictability analysis in climate dynamical models.

In Section 3, simulation studies are used to explore the potentials of statistical analysis tools such as MDS based on different dissimilarity measures. Given a family of parametric distributions, one dataset is independently drawn from each of the distributions. After summarizing each dataset into an empirical distribution, MDS based on pairwise Mallows distance and MDS based on symmetrized K-L divergence between these empirical distributions are carried out. Applying MDS based on Mallows distance to data sets drawn from a family of univariate mixture Gaussian distributions, we find that one of leading dimensions of MDS based on Mallows distance perfectly corresponds to the mean parameter and the other two leading dimensions together represent the standard deviation and shape of the distributions in a nonlinear fashion. However, MDS based on symmetrized K-L divergence does not show any clear patterns.

Analysis based on Mallows distance using AIRS L3Q data collected in December 2002 is presented in Section 4. We illustrate that compared to mean summaries, AIRS L3Q data contains additional useful information about configurations of regional short-term climates. By incorporating this configuration information, MDS based on Mallows distance forms better low dimensional projections of regional short-term climates than MDS based on Mean distance. Moreover, the configuration information also helps to form more consistent clustering of regional short-term climates and to identify special climate observations through spectral clustering based on Mallows distance. We conclude in Section 5 with conclusions and discussion.

2. Dissimilarity measures between distributions

In this section, we briefly review two types of dissimilarity measures between distributions: Mallows distance and f-divergence. The latter one includes a few commonly used dissimilarity measures such as Kullback-Leibler divergence (K-L divergence), Hellinger distance and χ^2 divergence as special cases. Next, we provide a short comparison between these two different types of measures and their applicabilities to certain types of problems.

2.1. **Mallows distance.**

Mallows distance (or Wasserstein distance) has been studied in many different fields such as Probability, Statistics and recently Computer Science. A brief history of Mallows distance can be found at [22]. In Statistics context, Mallows distance was first introduced in [18] as a metrication tool such that the convergence in Mallows distance is equivalent to the weak convergence and convergence of the q-th moment of a sequence of probability distributions. In short, Mallows distance between two distributions is the minimum expected distance among all pairs of random variables having those two distributions as marginal distributions. Mathematically, the Mallows distance is defined as

$$M_q(F,G) = \inf_P \left\{ (E_P\|X-Y\|^q)^{1/q} : \, (X,Y) \sim P, \, X \sim F, \, Y \sim G \right\}, \quad \text{for} \quad q \in [1,\infty).$$

The Mallows distance is also strongly connected with the Earth Movers' Distance (EMD), which becomes a popular way of measuring dissimilarities between images in computer science [16, 21]. It was shown in [15] that the Mallows distance is exactly the same as the EMD when it is used to measures the difference between two probability distributions. The EMD provide a nice interpretation of Mallows distance which could help us understand the meaning of Mallows distance in applications.

Here we provide a brief survey of some important properties of Mallows distance scattered in different literatures. We first list a few basic properties of Mallows distance [3] with $q \geq 1$.

- Mallows distance $M_q(F,G)$ satisfies the three requirements of a metric;
- $M_q(F_n,F) \to 0$ if and only if $F_n \to F$ weekly and $\int \|x\|^q dF_n(x) \to \int \|x\|^q dF(x)$;
- For F and G defined on $\mathcal{R}$, $M_q(F,G) = \left(\int_0^1 |F^{-1}(u) - G^{-1}(u)|^q du \right)^{1/q}$.

For the commonly used Mallows distance with $q = 2$, it is known that

- For F (mean μ_F and sd σ_F) and G (mean μ_G and sd σ_G) defined on $\mathcal{R}$:

$$M_2^2(F,G) = (\mu_F - \mu_G)^2 + (\sigma_F - \sigma_G)^2 + 2\sigma_F\sigma_G(1 - \rho_{QQ}(F,G)), \tag{1}$$

  where

$$\rho_{QQ}(F,G) = \frac{\int_0^1 (F^{-1}(u) - \mu_F)(G^{-1}(u) - \mu_G)du}{\sigma_F \, \sigma_G} = \frac{\int_0^1 F^{-1}(u)G^{-1}(u)du - \mu_F\mu_G}{\sigma_F \, \sigma_G}$$

  is the correlation of the quantiles of the two distributions as represented in a classical QQ plot [12].
- For multivariate Gaussian distributions $N(\mu_1,\Sigma_1)$ and $N(\mu_2,\Sigma_2)$,

$$M_2^2\left(N(\mu_1,\Sigma_1), N(\mu_2,\Sigma_2)\right) = \|\mu_1 - \mu_2\|^2 + \text{trace}\left(\Sigma_1 + \Sigma_2 - 2(\Sigma_1^{1/2}\Sigma_2\Sigma_1^{1/2})^{1/2}\right). \tag{2}$$

- For multivariate nonGaussians F (mean μ_F) and G (mean μ_G),

(3) $$M_2^2(F,G) = \|\mu_F - \mu_G\|^2 + M_2^2(F_0, G_0)\,,$$

  where F_0 and G_0 are derived by centering F and G at 0 [3]. Further decomposition of Mallows distance would be much more complicated than in univariate case. However, the Mallows distance is lower bounded by mean difference plus the measure of difference in covariance matrix as shown in (2) and the conditions for the equality to hold are given in [9].

From these properties, we clearly see that the Mallows distance can be decomposed into the difference in terms of location and configuration which includes spread, and shape. This decomposition of various aspects of the difference between distributions turns out to be very crucial and helpful in our statistical analysis based distances that will be detailed in Section 3 and Section 4.

2.2. ***f*-divergence.** Another type of dissimilarity measures between distributions is the f-divergence. In general, the f-divergence is defined as

$$D_f(F||G) = \int f\left(\frac{dF}{dG}\right) dG,$$

where f is a convex function that satisfies $f(1) = 0$. It is easy to show that the K-L divergence, the Hellinger distance and the χ^2 divergence are f-divergence corresponding to $f(t) = t\,log(t)$, $f(t) = 1 - \sqrt{t}$ and $f(t) = (1-t)^2$ respectively. Moreover, many other well known divergences are also variants of the f-divergence. For example, Jensen-Shannon divergence is the average of the K-L divergences of two distributions to their average and Bhattacharyya distance is a transformation of Hellinger distance. More details about these distances can be found at [2] and [17].

In statistics literature, these f-divergences are often used in information geometry as distance measures on the space of distributions. Many classical statistical methods such as Maximum Likelihood Estimation would have intuitive geometric interpretations. In addition, geometric methods could be employed to study the statistical properties of those methods. Meanwhile, K-L divergence is well studied in information theory due to the fact that K-L divergence exactly quantifies the redundancy of coding a source with a wrong distribution. For further details of f-divergence, we refer readers to [8].

One important application of K-L divergence in atmosphere science research is to quantify the predictability of future weather from climate dynamical systems. By interpreting the weather prediction as reducing uncertainty about future weather, K-L divergence seems to be a natural way to the quantify the predictability through its relative entropy interpretation in information theory. We refer readers to [14] and [10] for details of application of K-L divergence in predictability analysis.

2.3. **Comparison between Mallows distance and *f*-divergence.** To compare the Mallows distance and the f-divergence, we focus on discrete distributions in this section. Without loss of generality, two distributions can be represented as $F = \{(x_i, p_i), i = 1, ...n\}$ and $G = \{(x_i, q_i), i = 1, ..., n\}$, where $\{x_1, ..., x_n\} \subset \mathcal{R}^d$ is the union of the supports of F and G (some of the p_i's or q_i's could be zero). In this case, the Mallows distance and f-divergence are defined as

(4) $$M_2(F,G) = \min_{\substack{\sum_i p_{ij} = p_i, \forall j \\ \sum_j p_{ij} = p_j \forall i}} \left(\sum_{i=1}^{m}\sum_{j=1}^{n} p_{ij}||x_i - x_j||^2\right)^{\frac{1}{2}},$$

and

(5) $$D_f(F||G) = \sum_{i=1}^{n} q_i\, f\left(\frac{p_i}{q_i}\right),$$

respectively. Since some of the p_i's or q_i's could be zero, it is convenient to define $f(0) = \lim_{t\to 0} f(t)$, and $0f(\frac{a}{0}) = \lim_{t\to 0} tf(\frac{a}{t})$ [8].

A close investigation reveals several differences between these two classes of dissimilarity measures:

- Most importantly, the f-divergences defined in (5) are independent of the locations of distribution support $\{x_1, x_2, \ldots, x_n\}$. Therefore, in general, they are not good at reflecting the difference in mean or variance since such features depend on the support. Meanwhile, the Mallows distance is closely related with the locations of $\{x_1, x_2, \ldots, x_n\}$ and Equations (1), (2), (3) show that the differences in mean and variance are key components of Mallows distance between two distributions. The simulation studies shown in the next section will further illustrate this key property of Mallows distance.

- In case that two discrete distributions F and G do not share the same support (some of the p_i's or q_i's being zero), the f-divergence could be infinity for some choices of f; for example, the K-L divergence and the χ^2 divergence. On the other hand, the Mallows distance is always finite and it can be easily calculated by solving a linear programming problem.

To summarize, the Mallows distance is preferred in applications where support dependent features of distribution, such as mean and variance, are important, especially when the distributions have potentially different supports. It also has an advantage of being a well defined true metric, which will be convenient when the development of rigorous modeling is needed. On the other hand, when the relative locations of points in the support is not meaningful, such as in a "bag-of-features" representation of documents, or when information theoretical interpretation is desired, such as in predictability analysis, the f-divergence might be more appropriate. In the rest of the paper, we will concentrate on studying the properties of statistical analysis tools using Mallows distance with $q = 2$.

3. Statistical analysis on distributions

In statistics literature, most classical multivariate methods have been developed to deal with data vectors in an Euclidean space, in which geometry and linear projections play significant roles. For example, the popular Principle Component Analysis (PCA) seeks for a few orthogonal directions such that the linear projection of data vectors on these directions preserve the largest variation. In this paper, our interest is learning from a set of (empirical) distributions that do not lie on an Euclidean space, so PCA type of methods can not be directly applied. In such situations, statistical methods using only pairwise dissimilarities (distance) measures between subjects are more suitable. MultiDimensional Scaling (MDS) and Spectral Clustering algorithms both fall in this category. In this section, we will mainly focus on dimension reduction using MDS based on pairwise distance between distributions. The relationship between the leading MDS dimensions and the characteristics of distributions will be illustrated through a simulation study. In addition, the impact of the choice of distances will also be discussed. The MDS and Spectral clustering methods will be applied to AIRS L3Q data in Section 4.

To illustrate the potential advantage of analysis based on Mallows distance that recovers information beyond the first two moments, we use a simulation study that involves a family of univariate two-component mixture Gaussian distributions that are indexed by its mixing proportion, mean and standard deviation:

$$\left\{ X \sim F_{\pi,\mu,\sigma} : \begin{array}{c} F_{\pi,\mu,\sigma} = \pi N(\mu_1, \sigma_c^2) + (1-\pi) N(\mu_2, \sigma_c^2) \\ \text{s.t. } 0 \le \pi \le 1, \quad E(X) = \mu, \\ Sd(X) = \sigma > 0, \text{ and } \sigma_c = \frac{1}{2}\sigma \end{array} \right\} \tag{6}$$

One interesting property of the parameterization of this family of mixture distributions is that the shape of each distribution is only connected with the parameter π. In other words, the standardized versions of two distributions with same π are exactly the same. Such independent parametrization of mean, variance and shape would allow us to study the connection between the MDS dimensions and the parameters of distribution more straightforwardly.

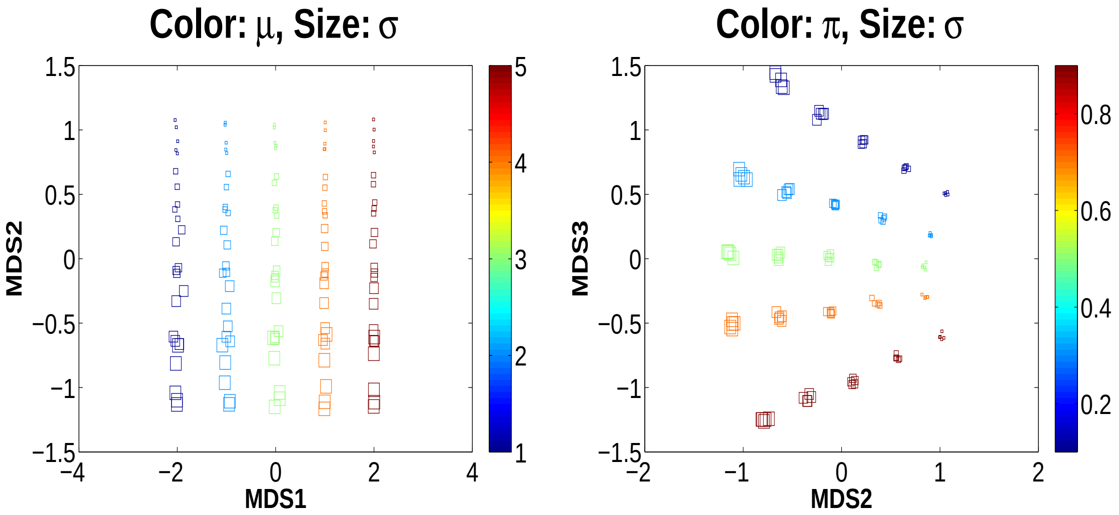


FIGURE 1. MDS projections based on Mallows distance: Left panel shows the scatter plot of MDS1 and MDS2 where μ and σ specify the color and size. Right panel is the scatter plot of MDS2 and MDS3 where π and σ specify the color and size.

Under this scenario, let us first investigate the properties of MDS based on Mallows distance from theoretical perspectives. Given two univariate distributions F and G, Eq.1 shows that

$$M_2^2(F,G) = (\mu_F - \mu_G)^2 + (\sigma_F - \sigma_G)^2 + 2\sigma_F\sigma_G(1 - \rho_{QQ}(F,G))$$

where $1-\rho_{QQ}(F,G)$ equals to the Mallows distance between standardized versions of F and G. For distributions in the family defined by (6), the term $\rho_{QQ}(F,G)$ is only a function of π_F and π_G. Several conclusions can be drawn from this decomposition. First, the mean difference is an independent factor in determining the Mallows distance. Therefore, one MDS dimension will reflect the relative locations of the means, if μ is independent of σ and π in the parameter space of this family, Secondly, difference in σ is another important contributor to the Mallows distance. Meanwhile, the size of σ interacts with the difference in π in determining the size of the Mallows distance. Thus, we do not expect σ and π can be independently represented by any single MDS dimension. Instead, there should be two (or more) MDS dimensions explain the relative locations of σ and π together. Since it is hard to explicitly represent $\rho_{QQ}(F,G)$ in terms of π_F and π_G, we will illustrate how σ and π are reflected in the MDS dimensions based on Mallows distance through a simulation study.

In the simulation study, we setup a grid for $(\mu, \sigma, \pi) \in \{1,2,3,4,5\} \times \{1,1.5,2.5,3,3.5\} \times \{0.1,0.3,0.5,0.7,0.9\}$. For each given $F_{\pi,\mu,\sigma}$, an empirical distribution with 20 centers is constructed based on 1000 *i.i.d.* sampled points. MDS based on the pairwise Mallows distance between these empirical distributions is carried out and the scatter plots based on first three MDS dimensions are shown in Figure 1. The MDS plot in the left panel has axes correspond to first two dimensions and is color coded according to μ and size coded according to σ. The axes of the right panel are the second and third MDS dimensions, where color represents the mixing weights π and symbol size stands for σ.

The left panel of Figure 1 clearly shows that the first MDS dimension exactly corresponds to the mean parameter μ. This result confirms the theoretical interpretation discussed above. Meanwhile the second MDS dimension roughly corresponds to overall standard deviation. However, the plot on the right panel indicates that actually the second and third MDS dimensions together give a better characterization of overall standard deviations and mixing weights. This also agrees with our expectation raised from the theoretical analysis.

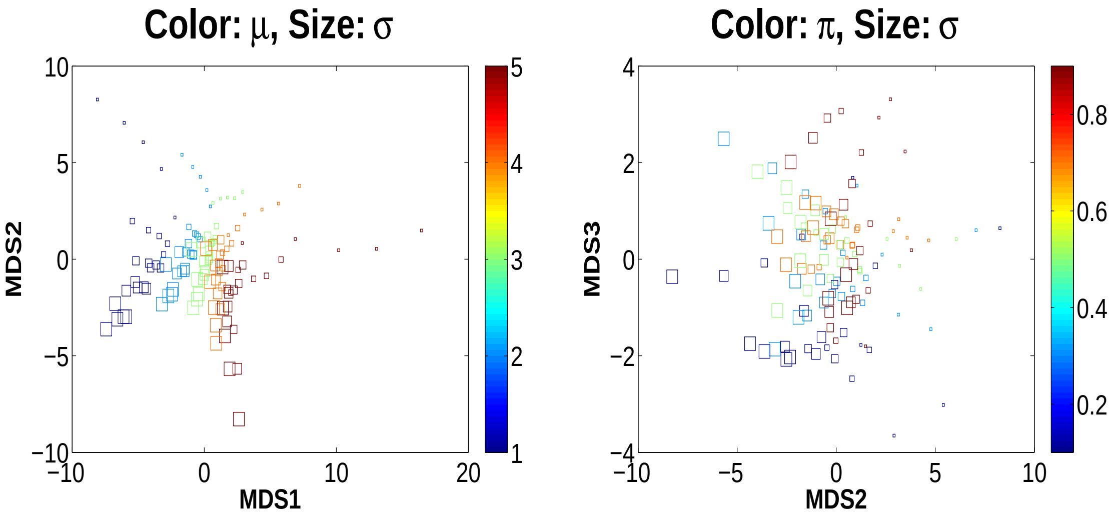


Figure 2. MDS projections based on symmetrized K-L divergence: Left panel shows the scatter plot of MDS1 and MDS2 where μ and σ specify the color and size. Right panel is the scatter plot of MDS2 and MDS3 where π and σ specify the color and size.

The MDS results based on symmetrized K-L divergence using the same dataset are shown in a similar way in Figure 2 for comparison purposes. As shown in the left panel, MDS1 and MDS2 together are related to the μ and σ in a highly nonlinear pattern. On the other hand, none of these three MDS dimensions shows any correlation with π. In addition, two pairs of distributions which differ the same amount in the parameter space are mapped quite differently. Therefore, symmetrized K-L divergence does not properly reflect the difference in the parameter space. This simulation result confirms our discussion in Section 2 that K-L divergence performs badly in applications when the distribution support related parameters, such as mean and variance, are of interests.

So far, we have shown the potential of MDS based on Mallows distance in terms of recovering underlying structures related to the mean and the configuration (including spread and shape). These properties will also help us interpret the MDS dimensions in the real data applications discussed in the next section.

4. Analysis of AIR L3Q data

Launched into Earth-orbit on May 4, 2002, the Atmospheric Infrared Sounder (AIRS) is one of six instruments on board the Aqua satellite. AIRS level 1 data are the observed radiation (emitted and reflected) in 2378 spectral channels from each 45 km ground footprint. Using retrieval algorithms based on Physics knowledge, the level 1 data at each footprint is converted into level 2 data, which contains 11 atmospheric temperatures and 11 water vapor mass-mixing ratios corresponding to the bottom 11 air pressure levels, 10 cloud fraction values corresponding to the same pressure levels except the surface, and other variables. The volume of AIRS level 2 data produced in each year is about 33 GB.

AIRS level 3 Quantization (L3Q) products summarize valid level 2 data in each $5^o \times 5^o$ latitude-longitude grid box collected during a time period (a month or a season) by a list of representative vectors and their frequencies. Therefore, this multivariate empirical distribution in each grid box represents a record of the regional climate during the corresponding period of time. Compared to traditional data reduction methods which summarize each dataset by its first two moments, this clustering based approach has the potential in keeping the distribution configuration beyond sample means. In addition, outlier information in the dataset, which represents special weather pattern, could also be retained. However, how to carry out statistical analysis on such datasets is an open problem, mostly because the data do not lie in an Euclidean space. Fortunately, by interpreting AIRS L3Q data as a set of empirical distributions, statistical analysis based on distance between distributions, such as MDS and spectral clustering, provides us with an alternative.

In this section, we apply MDS and spectral clustering on AIRS L3Q data with different dissimilarity measures and study the difference in the results. Mallows distance between the empirical distributions in each pair of grid boxes is the first dissimilarity measure in our study. Besides the Mallows distance, the Euclidean distance between the mean vectors of the data in each pair of grid boxes is another measure one may use to represent the dissimilarity which ignores the configuration information of empirical distributions. With the AIRS L3Q data, the mean vectors for each grid box can be easily derived.

Analysis based on K-L distance is not suitable for this dataset due to its instability when applied to AIRS L3Q data. We find that the water vapor mass-mixing ratios and cloud fraction variables have small or even no variability between representative vectors within some empirical distributions, while they have significant difference between the mean vectors of empirical distributions. Due to the nature of K-L distance, these variables lead to extremely large K-L distances, even to infinity. Therefore, K-L distance does not seem to be a proper measure of dissimilarity for analyzing the AIRS L3Q data.

4.1. **Multidimensional Scaling.** MDS based on Mallows distances was applied to AIRS L3Q data in winter season of each year in [6], along with a comparison to MDS based on the distance between mean vectors of each grid box. It turned out that the first dimension of MDS based on Mallows distance is almost identical to the first MDS dimension based on the mean vectors. However, the second dimension of MDS based on Mallows distance is much more similar to an important physical process called the vertical velocity than to the second MDS dimension based on the mean vectors. Instead of trying to match the *MDS2-Mallows* with physical process, we concentrate on interpreting such difference from a different angle which might provide further understanding of the MDS results.

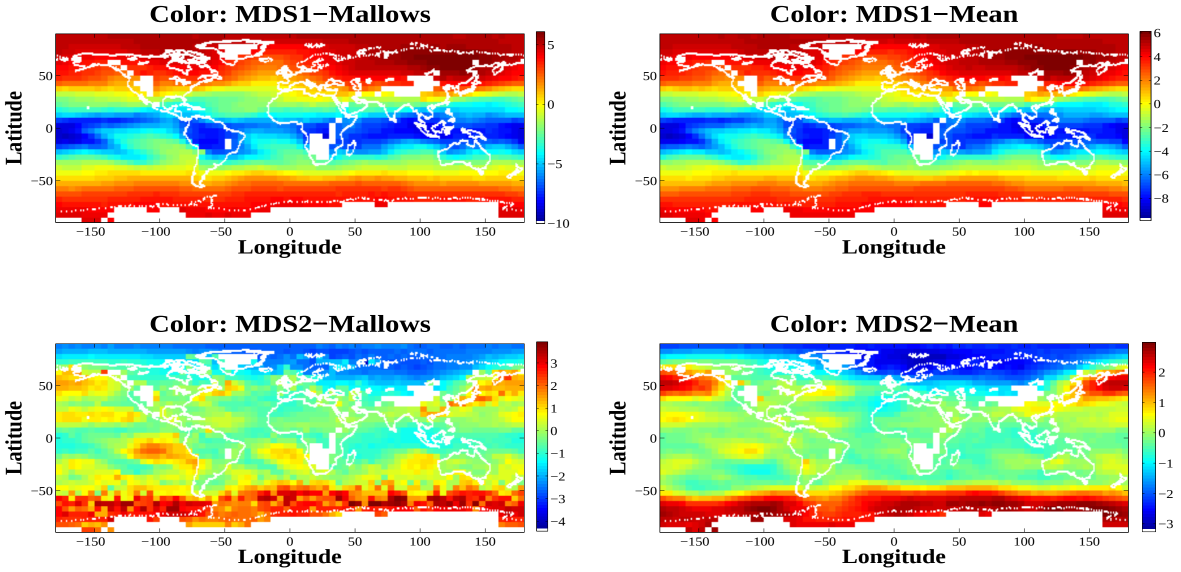


FIGURE 3. MDS dimensions displayed in geographic maps: Left two panels are colored by *MDS1-Mallows* and *MDS2-Mallows*. Right two panels are colored by *MDS1-Mean* and *MDS2-Mean*. White blocks in the maps show the locations where data are missing.

The dataset in this application is the AIRS monthly L3Q data collected in December 2002. The dataset contains the valid empirical distributions in 2338 5^o by 5^o grid boxes. Two adjustments are applied to the data before we perform further analysis. First, we remove the last three indicator variables which contain little information about climate and are mostly used for reference purposes. Secondly, each remaining variable is properly normalized such that weighted average and weighted sample variance of the representative vectors of all empirical distributions are 0 and 1 respectively. With these two modifications, the observation in each 5^o by 5^o grid box is a 32 dimensional discrete distribution. Based on this standardized data, Mallows distances and mean distances between each pair of empirical distributions are computed and used in MDS. To simplify our discussion, we introduce some notations first. The ith MDS dimension based on Mallows distance will be denoted as *MDSi-Mallows* and *MDSi-Mean* corresponds to the one based on mean distance.

We first visualize the leading MDS dimensions using maps. Since each data point in a MDS dimension is associated with the latitude and longitude of the grid box, MDS results can be displayed in maps where color codes the value of MDS dimensions as shown in Figure 3. Similar to the findings presented in [6], *MDS1-Mallows* matches with *MDS1-Mean* almost perfectly while *MDS2-Mallows* disagrees with *MDS2-Mean* in some regions.

Recall that one of properties of the Mallows distance in Eq (3) is

$$M_2^2(F,G) = ||\mu_F - \mu_G||^2 + M_2^2(F_0, G_0),$$

where F_0 and G_0 are derived by centering F and G at 0 . For simplicity, we will call those two parts in the decomposition as mean difference and configuration difference. The configuration of a distribution includes information about its spread and shape. This decomposition may help us better interpret the results based on Mallows distance. By dividing Mallows distance into two parts, we get a new distance matrix which measures the difference in configuration. To be consistent with notations above, the i-th MDS dimension based on configuration distance is represented as *MDSi-Config.*

On the left panel of Figure 4, we show the scatter plots of *MDS1-Mean* and *MDS1-Config* where color codes the *MDS1-Mallows.* No obvious relationship is observed. In addition, as shown in Table 1, the eigenvalue 4.05×10^4 corresponds to *MDS1-Mean* is much larger than 0.27×10^4 for *MDS1-Config.* For the rest of MDS dimensions of configuration distance, their corresponding eigenvalues are too small to have any significant impact on *MDS1-Mallows.* Therefore, *MDS1-Mean* totally dominates the *MDS1-Mallows.*

Now let us concentrate on *MDS2-Mallows.* Plotted on the right panel of Figure 4 is the scatter plot of *MDS2-Mean* and *MDS1-Config* where color codes the *MDS2-Mallows.* We observe that *MDS2-Mean* is negatively correlated with *MDS1-Config* with a correlation -0.361. Table 1 also suggests that their eigenvalues (0.41×10^4 v.s. 0.27×10^4) are also comparable. The color changing direction (*MDS2-Mallows*) in the right panel of Figure 4 shows a linear combination of *MDS2-Mean* and *MDS1-Config.* This implies that we might be able to get further understanding of *MDS2-Mallows* through interpreting *MDS2-Mean* and *MDS1-Config* separately.

Eigenvalues	1	2	3	4	5
MDS-Mallows($\times 10^4$)	4.13	0.50	0.38	0.32	0.25
MDS-Mean($\times 10^4$)	4.05	0.41	0.31	0.10	0.064
MDS-Config($\times 10^4$)	0.27	0.18	0.14	0.09	0.07

Table 1. Eigenvalues of MDS: For each distance, the first five eigenvalues of MDS dimensions are shown.

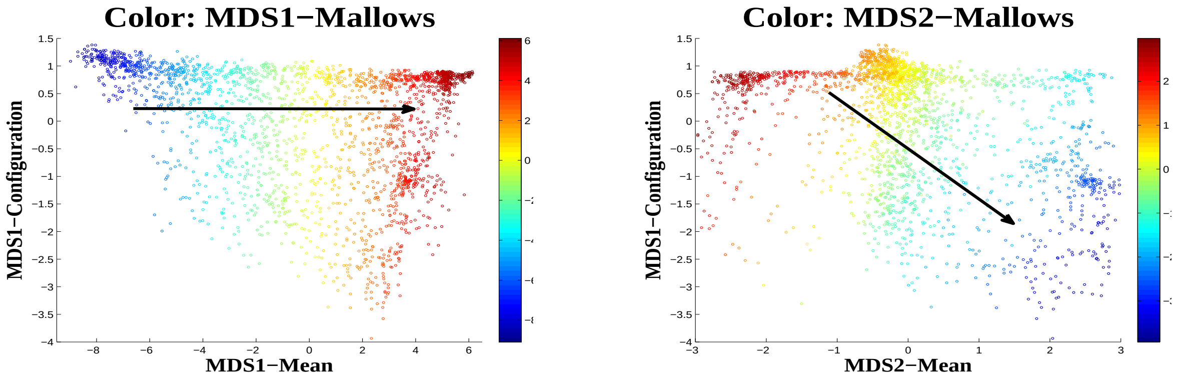


FIGURE 4. Scatter Plots of MDS Dimensions: Left panel shows the scatter plot of *MDS1-Mean* and *MDS1-Config.* Points are colored by *MDS1-Mallows* and the arrow is indicates changing of color. Right panel shows the scatter plot of *MDS2-Mean* and *MDS1-Config.* The color represents *MDS2-Mallows* and the arrow indicates changing of color.

Through our analysis, we illustrate that AIRS L3Q data contains more information of the configuration of the regional short-term climate than the mean summary. Especially we find that the major difference between MDS dimensions of Mallows distance and Mean distance is contributed by variations in configuration. More specifically, *MDS2-Mean* is correlated with *MDS1-Config* and we hypothesize that *MDS2-Mallows* is determined jointly by those two dimensions.

4.2. **Clustering.** As another way to explore the hidden structure in data, clustering analysis tries to identify natural groups in data according to a given dissimilarity measure. Clustering methods could be applied on AIRS L3Q data to identify typical regional short-term climate patterns and potential climate outliers. In this section, we perform clustering on a subset of AIRS L3Q data collected in December 2002. Similar to the MDS analysis, clustering is performed with both the Euclidean distance of mean summaries of AIRS data and Mallows distance between empirical distributions. Significant differences are observed between two clustering results. The clustering based on Mallows distance is more reasonable in a sense that the group members are more consistent with each other in terms of short-term climate. Moreover, the clustering based on Mallows distance detects some potential climate outliers which are missed by clustering using the mean summaries.

In this application, we use a subset of the AIRS L3Q data collected in December 2002, which is a rectangle area covering most of North American. The southwestern corner of the rectangle is located at $[-135^o, 15^o]$ and the northeastern corner is at $[-50, 50]$. Similar to the analysis in MDS, we also remove the three indicator variables and normalize the remaining variables. In addition, we exclude the cloud fraction at the second pressure level from the analysis. That is because the representative vectors in all empirical distributions have constant value in this variable. In all, this subset of data includes 127 empirical distributions in 31 dimensions.

For clustering methods, we employ a recently developed Data Spectroscopic clustering algorithm[26], which is one of many spectral clustering methods. Compared to the K-Means type of algorithm in [16], Data Spectroscopic method is computationally much faster and does not have the problem of finding a local optimal solution. In addition, Data Spectroscopic method has more power in identifying potential outliers. More details of Data Spectroscopic method can be found in [26]. Data Spectroscopic clustering is implemented on the selected data with mean distance and Mallows distance separately. The tuning parameter in Data

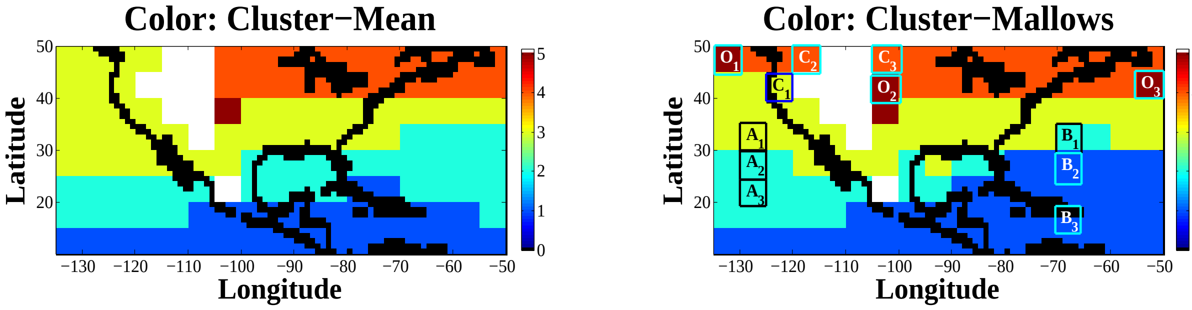


Figure 5. Clustering displayed in geographic maps: Three panels are colored by *Cluster-Mean* and *Cluster-Mallows*. White blocks in the maps show the locations where data are missing.

Spectroscopic method is set to generate clustering with 5 groups, where the fifth group corresponds to the identified outliers. We use *Cluster-Mean* and *Cluster-Mallows* to refer to the clustering results based on those two different dissimilarity measures.

We first show the geographic map of the rectangle area with color coded groups in Figure 5. The left panel shows *Cluster-Mean* and the right panel corresponds to *Cluster-Mallows*. Comparing these two clustering results, we find that *Cluster-Mean* and *Cluster-Mallows* are similar to each other and we can match up the groups with each other. Despite the similarity, two significant differences can still be observed. Firstly, there are several regions being clustered differently based on mean distance and Mallows distance. These regions are mostly on the geographical boundaries of some groups. Secondly, *Cluster-Mean* identifies one grid box as potential outlier while *Cluster-Mallows* detects three more potential outliers. These three regions are labeled as $O_1 - O_3$ in Figure 5. In the rest of this section, we concentrate on explaining these observed differences.

To better explain the differences in clustering results, let us start with describing a visualization method we develop for investigating empirical distributions. Given an empirical distribution F, we separate it into two parts: mean μ_F and configuration F_0. Basically, μ_F is a 31 dimensional vector and F_0 is a set of vectors, each of which is associated with a weight. For each vector, we present it as a horizontal strip composed of 31 blocks. For vectors in configuration, their weights in configuration will determine the heights of their corresponding strips. Meanwhile, each block of strip will be color coded by the value of corresponding mean or representative vector in that variable. Blocks $1-11$ correspond to the values of atmosphere temperatures starting from the surface level, blocks $12-22$ represent the values of water vapor mass-mixing ratios and the last 9 blocks are for cloud fractions starting from the third pressure level. To enhance the contrast between blocks in visualization, all the values greater than 3 or smaller than -3 will be set to 3 or -3 in visualization.

With this visualization tool, we first show the center empirical distributions of Groups $1-4$ in *Cluster-Mallows* in Figure 6. The center empirical distribution of a group is defined as an empirical distribution which has the smallest Mallows distance to all empirical distributions in the group. As in [16], we use an iterative algorithm to find an approximate center empirical distribution for each group. A brief comparison shows that those center empirical distributions not only differ in mean but also have significant difference in configuration, especially in those variables related to cloud fractions. The significant difference in configuration among center empirical distributions indicates that ignoring configuration information might lead to inconsistent clustering of climate observations.

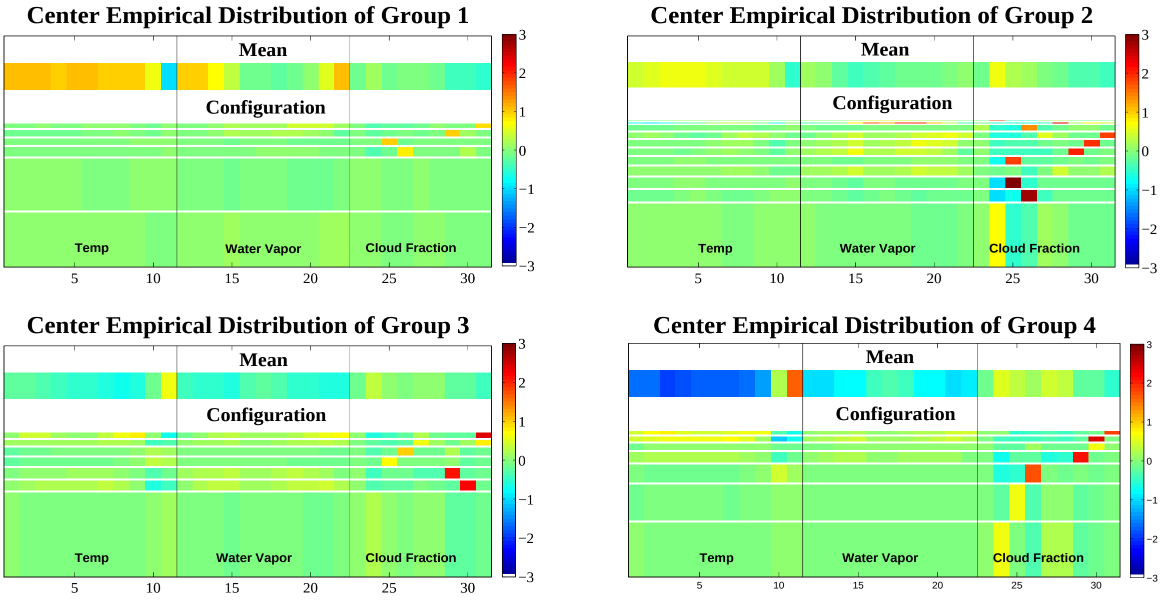


FIGURE 6. Visualization of four center empirical distributions for Group 1-4 in *Cluster-Mallows*.

We now turn our focus to the grid boxes which are classified into different groups in two clustering results. Due to space limitation, we take grid boxes A_2, B_2, and C_2, which are highlighted on the right panel of Figure 5, as examples. As shown in Figure 5, these grid boxes are clustered differently in *Cluster-Mean* and *Cluster-Mallows*. We also choose two reference grid boxes for each of $\{A_2, B_2, C_2\}$ for comparison purpose. Two reference grid boxes, A_1 and A_3, are chosen such that A_1 is in the same group as A_2 in *Cluster-Mean* and A_3 is in the same group as A_2 in *Cluster-Mallows*. In addition, A_1 is assigned to the same group in *Cluster-Mean* and *Cluster-Mallows* and so is A_3. The grid boxes $\{B_1, B_3\}$ and $\{C_1, C_3\}$ are chosen similarly with respect to B_2 and C_2. The geographic locations of the reference grid boxes are marked on the right panel of Figure 5. We visualize the empirical distributions in $\{A_1, A_2, A_3\}$, $\{B_1, B_2, B_3\}$, and $\{C_1, C_2, C_3\}$ on the top three, middle three and bottom three panels of Figure 7 respectively.

Visual comparisons of these empirical distributions in Figure 7 reveal interesting patterns. Let us take $\{A_1, A_2, A_3\}$ as an example. By comparing the mean vectors shown on the top three panels of Figure 7, we find that the empirical distribution in A_2 is closer to that in A_1 in terms of mean. On the other hand, both configurations in A_2 and A_3 have large variability in variables $24 - 26$ which is missing in the configuration in A_1. Therefore, A_2 is much closer to A_3 than to A_1 in terms of configuration. Moreover, by comparing the empirical distribution in A_2 with the center empirical distribution of Group 2 in *Cluster-Mallows*, we observe that they share a similar pattern in configuration that shows large variability in variables $24 - 26$. This indicates that the special pattern of the empirical distribution in A_2 relates to the feature of Group 2 in *Cluster-Mallows*. Therefore, A_2 should be clustered into Group 2 (same as A_3) if clustering regional short-term climates is of our interest. Similar observations are made in the comparison of $\{B_1, B_2, B_3\}$ and $\{C_1, C_2, C_3\}$. From these comparisons, we illustrate that the clustering based on mean vectors will misclassify some empirical distributions while Mallows distance has the ability to form more consistent clusters about regional short-term climate.

Finally, we concentrate on the potential outliers. The three additional outlier regions identified in *Cluster-Mallows* are marked as $O_1 - O_3$ in Figure 5 and their corresponding empirical distributions are plotted in Figure 8. By comparing the mean vectors of empirical distributions in $O_1 - O_3$ with those of empirical

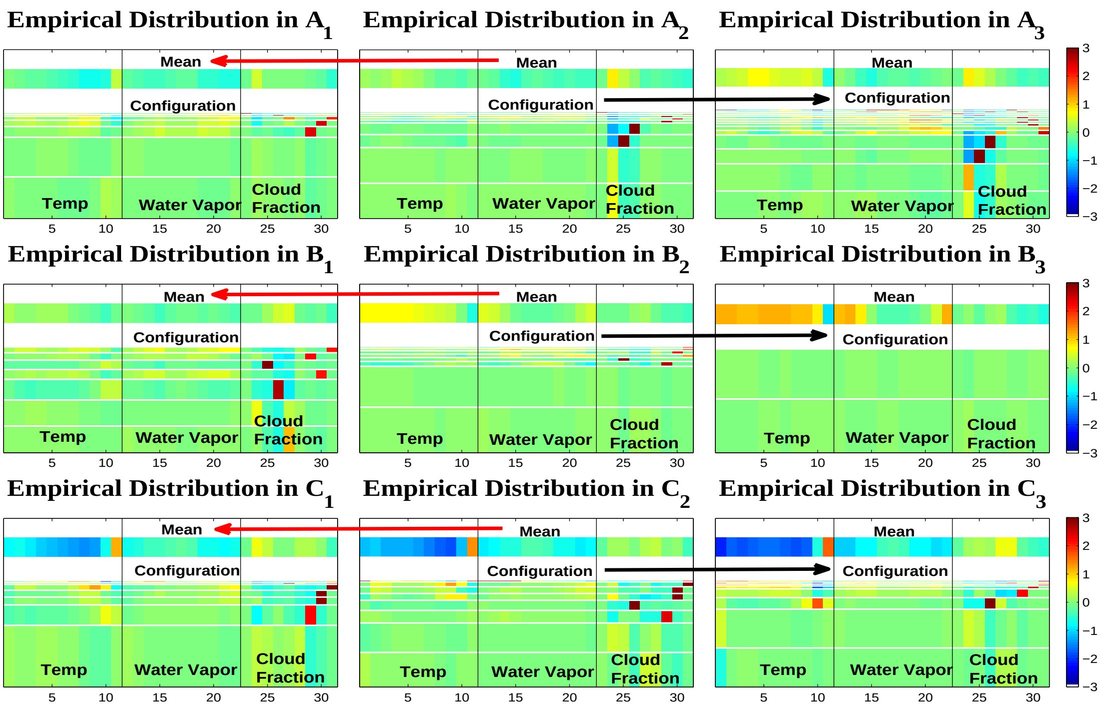


FIGURE 7. Visualization of three groups of empirical distributions: The middle panels are empirical distributions for three grid boxes which are clustered differently. The left and right three panels are the reference empirical distributions for those in the middle panel. All these empirical distributions are marked in Figure 5.

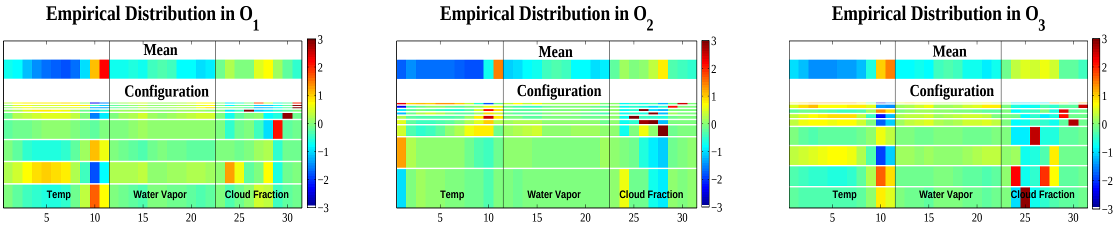


FIGURE 8. Visualization of four potential outliers identified in *Cluster-Mallows*.

distributions in $C_1 - C_3$ and that of center empirical distribution of Group 4 in *Cluster-Mallows* , we do not observe any specialties of mean vectors of those outliers. However, in terms of configuration, empirical distributions in $O_1 - O_3$ show great variability in both temperature variables and cloud fraction variables. Such large variability in temperatures is not obvious in other empirical distributions. Therefore, it is the special pattern in the configuration that makes these three empirical distributions as outliers.

Through our study, we further illustrate that mean vectors do not fully characterize the difference among empirical distributions in AIRS L3Q data. Moreover, the configuration information in the empirical distributions helps us better cluster the regional short-term climate observations and identify potential outliers.

5. Conclusions and discussion

In this paper, we study potential statistical methods to analyze an unconventional type of data, empirical distributions. The class of statistical tools that only depend on pairwise dissimilarity measures is investigated to show their applicability for this task. We summarize the theoretical properties of different dissimilarity measures and find Mallows distance being more attractive than others in quantifying the difference between empirical distributions with potential different supports, especially when the support of distribution matters.

The simulation on learning a family of mixture distributions suggests that the Mallows distance is a smooth function of mean difference and configuration difference. When empirical distributions are constructed from samples from a parametric family of distributions, the metric geometry induced by Euclidean distance between parameters can be locally approximated by the metric geometry induced by the Mallows distance. This implies the potential feasibility of using Mallows distance in some statistical methods such as Kernel PCA, and Spectral clustering, whose results heavily depend on the local distances.

Through the analysis on AIRS L3Q data, we demonstrate that means and configurations both contain useful information for analysis about regional short-term climate. Statistical methods based on Mallows distance are useful tools to visualize and interpret such information. However, in order to answer some scientific questions, statistical modeling is required. From the exploratory analysis shown above, the first two MDS dimensions of Mallows distance exhibit some spatially clustering patterns. In addition, spectral clustering also shows the existence of clustering patterns in empirical distributions. These observations indicate that a mixture model might be a reasonable model for these empirical distributions. We will pursue this direction in our future research.

Acknowledgments

This research is partially supported by NASA grant NNG06GD31G and NSF grant DMS-1007060. The authors would like to thank Dr. Amy Braverman, Dr. Noel Cressie and Dr. Steve Maceachern for helpful discussion. The authors also thank three anonymous reviewers for useful comments.

References

[1] M. Belkin and P. Niyogi. Using manifold structure for partially labeled classification. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems 15*, pages 953 – 960. MIT Press, 2003.

[2] A. Bhattacharyya. On a measure of divergence between two statistical populations defined by their probability distributions. *Bulletin of the Calcutta Mathematical Society*, 35(4):99–109, 1943.

[3] P. J. Bickel and D. Freedman. Some asymptotic theory for the bootstrap. *Annals of Statistics*, 9(1196-1217), 1981.

[4] A. Braverman. Compressing massive geophysical datasets using vector quantization. *Jorurnal of Computational and Graphical Statistics*, 11:44–62, 2002.

[5] A. Braverman, E. Fetzer, A. Eldering, S. Nittel, and K. Leung. Semi-streaming quantization for remote sensing data. *Journal of Computational and Graphical Statistics*, 12(4):759–780, 2003.

[6] A. J. Braverman, E. J. Fetzer, B. H. Kahn, E. M. Manning, R. B. Oliphant, and J. P. Teixeira. Massive data set analysis for nasa's atmospheric infrared sounder. Technical report, Jet Propulsion Laboratory, California Institute of Technology, 2010.

[7] T. Cox and M. Cox. *Multidimensional Scaling*. Chapman and Hall, second edition, 2001.

[8] I. Csiszár and P. Shields. Information theory and statistics: A tutorial. *Foundations and Trend in Communications and Information Theory*, 4:417–528, 2004.

[9] J. A. Cuesta-Albertos, C. Matrán-Bea, and A. Tuero-Diaz. On lower bounds for the l2-wasserstein metric in a hilbert space. *Journal of Theoretical Probability*, 9:263–283, 1996.

[10] T. DelSole and M. K. Tippett. Predictability: Recent insights from information theory. *Reviews of Geophysics*, 45(4):1–74, 2007.

[11] M. Hermann, A. Greß, and R. Klein. Interactive exploration of large event datasets in high energy physics. *Journal of WSCG*, 17(1):41–48, Feb. 2009.

[12] A. Irpino and E. Romano. Optimal histogram representation of large data sets: Fisher vs piecewise linear approximations. *RNTI*, E9(99-110), 2007.

[13] A. Irpino, R. Verde, and Y. Lechevallier. *Dynamic clustering of histograms using Wasserstein metric*, page 869876. Citeseer, 2006.

[14] R. Kleeman. Measuring dynamical prediction utility using relative entropy. *Journal of the Atmospheric Sciences*, 59(13):2057–2072, 2002.

[15] E. Levina and P. Bickel. The earth mover's distance is the mallows distance: Some insights from statistics. In *Proc. International Conference on Computer Vision*, pages 251–256, Vancouver, Canada, 2001.

[16] J. Li and J. Z. Wang. Real-time computerized annotation of pictures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):985–1002, 2008.

[17] J. Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991.

[18] C. Mallows. A note on asymptotic joint normality. *Annals of Mathematical Statistics*, 42(2):508–515, 1972.

[19] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In T. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, pages 955 – 962. MIT Press, 2002.

[20] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326, 2000.

[21] Y. Rubner, C. Tomasi, and L. J. Guibas. A metric for distribution with applications to image databases. In *Proc. International Conference on Computer Vision*, pages 59–66, Bombay, India, 1998.

[22] L. Rüschendorf. *Wasserstein Metric.* Kluwer Academic, 1995.

[23] B. Schölkopf, A. Smola, and K. R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10:1299–1319, 1998.

[24] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[25] T. Shi, M. Belkin, and B. Yu. Data spectroscopy: learning mixture models using eigenspaces of convolution operators. In A. McCallum and S. Roweis, editors, *Proceedings of the 25th Annual International Conference on Machine Learning (ICML 2008)*, pages 936–943. Omnipress, 2008.

[26] T. Shi, M. Belkin, and B. Yu. Data spectroscopy: Eigenspace of convolution operators and clustering. *Annals of Statistics*, 37(6B):3960–3984, 2009.

[27] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[28] Y. Weiss. Segmentation using eigenvectors: a unifying view. *International Conference on Computer Vision*, 1999.

A MODEL-FREE TIME SERIES SEGMENTATION APPROACH FOR LAND COVER CHANGE DETECTION

ASHISH GARG[†]*, LYDIA MANIKONDA[†]*, SHASHANK KUMAR**, VIKRANT KRISHNA*,
SHYAM BORIAH*, MICHAEL STEINBACH*, VIPIN KUMAR*, DURGA TOSHNIWAL**,
CHRISTOPHER POTTER***, AND STEVEN KLOOSTER***

ABSTRACT. Ecosystem-related observations from remote sensors on satellites offer significant possibility for understanding the location and extent of global land cover change. In this paper, we focus on time series segmentation techniques in the context of land cover change detection. We propose a model-based time series segmentation algorithm inspired by an event detection framework proposed in the field of statistics. We also present a novel model-free change detection algorithm for detecting land cover change that is computationally simple, efficient, non-parametric and takes into account the inherent variability present in the remote sensing data. A key advantage of this method is that it can be applied globally for a variety of vegetation without having to identify the right model for specific vegetation types. We evaluate the change detection capacity of the proposed techniques on both synthetic and MODIS EVI data sets. We illustrate the importance and relative ability of different algorithms to account for the natural variation in the EVI data set.

1. INTRODUCTION

The goal of the land cover change detection problem is to detect when the land cover at a given location has been converted from one type to another. It is very important to study land cover change in order to understand its impact on local climate, radiation balance, biogeochemistry, hydrology, and the diversity and abundance of terrestrial species [6, 17]. Such understanding can be very valuable for policy makers, natural resource managers and researchers to address the issues related to global environmental changes. A large body of change detection studies from remotely sensed imagery has focused on comparisons between two images: one before and one after a change [8]. However, such techniques are usually domain or region specific and require expensive training and thus are difficult to scale globally. Recognizing these limitations, several algorithms have been developed [17, 16] to detect changes in the time series of satellite-based observations such as the Enhanced Vegetation Index (EVI) [3]. EVI, which is a product based on measurements taken from the MODIS instrument on NASA's Terra and Aqua satellites, is available globally at 250m and 1km resolution and at a temporal frequency of 16 days, since February 2000.

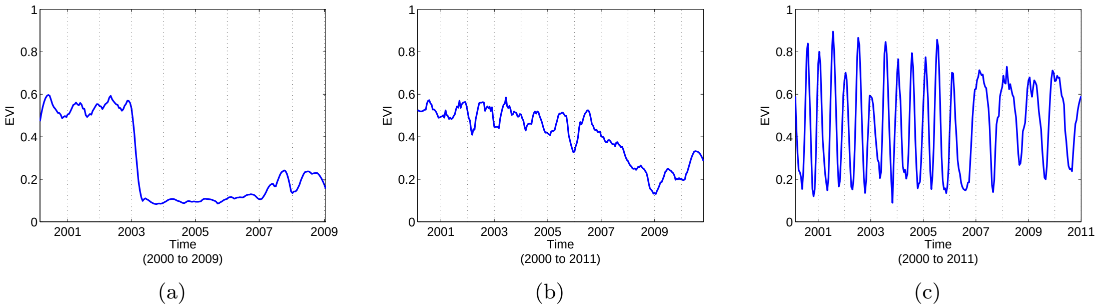
A number of techniques [5, 11] have been developed recently for identifying sudden drops in the vegetation index time series (e.g. in Figure 1(a)) or slow degradation (e.g. in Figure 1(b)) that can occur due to fires or logging etc. However, these techniques are unable to effectively detect changes such as conversion of forested land to crop land, intensification of agriculture, and change in cropping patterns. These changes do not necessarily result in loss of vegetation, for example, see Figure 1(c) for the change in cropping pattern from double to single crop per year. Rather, these changes result in characteristic change in the regular pattern of the EVI time series. The ability to monitor such land cover changes at local, regional and global scale is important due to their potential impact on the environment.

[†] These authors contributed equally to this work.

*University of Minnesota, <ashish,lmani,vikrant,sboriah,steinbac,kumar>@cs.umn.edu

**Indian Institute of Technology, Roorkee <sha28uec,durgafec>@iitr.ernet.in

***NASA Ames Research Center, chris.potter@nasa.gov, sklooster@gaia.arc.nasa.gov .



(a) (b) (c)

Figure 1. EVI vegetation time series (Feb–2000 to Sep–2010 -Vertical lines indicate yearly boundaries) showing (a) Sudden drop in year 2003 at a location in California; (b) Slow degradation from 2006 to 2009 at a location in Washington; (c) Conversion of double to single cropping for a location in Zimbabwe in 2006. Note that the mean EVI for each year is similar for this time series.

The problem of detecting land cover changes can be posed as segmenting a vegetation index time series. The goal of segmentation is to partition the input time series into homogeneous segments such that the subsequence within a segment is homogeneous and the segments are heterogenous with respect to each other. Segmentation thus is essentially a special case of change detection since by definition successive segments are not homogeneous.

In this paper, we focus on time series segmentation techniques in the context of land cover change detection. The key contributions of this paper are:

(1) We propose a model-based time series segmentation algorithm inspired by a statistical event detection framework.
(2) We present a novel model-free change detection algorithm for detecting land cover change that is computationally simple, efficient, non-parametric and takes into account the inherent variability present in the remote sensing data. A key advantage of this technique is that it can be applied globally for a variety of vegetation without having to identify the right model for specific vegetation types.
(3) We evaluate the change detection efficacy of the proposed techniques on two data sets (i) simulated 16-day EVI time series containing phenological changes, and (ii) 16-day MODIS EVI time series for a region in North Carolina for which an independent land cover classification is available from National Oceanic and Atmospheric Administration (NOAA).
(4) We then illustrate the importance and the relative ability of different algorithms to account for the natural variation in the EVI data set due to different vegetation types, climate variability, geographic variability and errors in the data.

Organization of the Paper. Section 2 discusses previous work on segmentation techniques for land cover change. In Section 3, we present the two change detection algorithms. Section 4 describes the data used for experimentation. Section 5 presents the experimental evaluation with both simulated and real input data sets. Section 6 contains concluding remarks. Note that most figures in this paper are best seen in color.

2. Time Series Segmentation Techniques for Land Cover Change Detection: Related Work

In this section, we discuss the time series segmentation algorithms in the context of land cover change detection. In general, effective techniques for land cover change detection must be (i) scalable to handle large scale high resolution data sets; (ii) stable and robust to varying vegetation types; (iii) take into account noise and inherent variability present in the Earth Science data.

One commonly used segmentation-based approach divides a time series into multiple segments such that each segment can be approximately represented by a piecewise linear curve [13, 15]. The two key steps in this approach are to determine the best linear curve within a segment and to determine the number of segments in a time series. These techniques have been used in the remote sensing community for extracting phenology characteristics (e.g. timing of maximum of the growing season, length of growing season, onset of vegetation green-up) of the time series per year [7]. However, our work differs in that its objective is not to extract phenology characteristics but rather to identify the changes in the time series.

Below, we present the next two broad categories, model-based and model-free segmentation approaches and how they can be adapted for land cover change detection. We focus on identifying only one change in the time series though many of the techniques may be extended to find multiple changes. We specifically discuss the relative capabilities of these techniques to handle inherent variability and noise present in the data.

Model-based techniques involve fitting a model to a given time series. One such technique was proposed by Guralnik et al in [12]. It considers segmentation as a problem of either recognizing the change of parameters in the underlying model or the change of the most suitable model fit to the time series. It is an iterative algorithm that fits a model to a time segment, and uses a likelihood criterion to determine if the segment should be partitioned further. This approach is a top down strategy [15] which works by considering every possible partitioning of the time series and splitting it at the best location. Both the segments of the time series are then recursively partitioned in a similar way until a stopping criterion is reached. For single change point detection, the techniques aims at finding the first split. Therefore, in these techniques it is important to choose the correct model to represent the segments and an appropriate threshold as a stopping criterion. We adapt this technique for land cover change detection and evaluate it quantitatively. We also show that the choice of model plays a critical role in the performance of this algorithm.

Another *model-based* approach, Breaks for Additive Seasonal and Trend (BFAST) proposed recently by Verbesselt et al. [18] decomposes a time series into trend, seasonal and residual components. The time series is divided into segments such that intra-segment trend is constant and inter-segment trends are dissimilar. A trend breakpoint is associated with segment boundaries. The seasonal component is handled in a similar fashion. The focus of this work is on a paradigm for identifying multiple changes of different types, therefore we will not be comparing it directly in this paper. However, in the context of finding a single change in the seasonal pattern (which is the focus of our paper), BFAST is similar to the scheme presented in [12].

On the other hand, *model-free* time series segmentation algorithms do not assume any model for the time series but rather work directly with the data values. One such technique is the Recursive Merging algorithm proposed in [6] for land cover change detection. The algorithm starts with each year as the segmentation of original time series of length t, i.e with t/p (where p is the season length) segments. Next, the algorithm computes the cost of merging every adjacent pair of segments and iteratively merges the lowest cost pair. The process is repeated until two segments are left. The cost of merging can be computed in different ways, such as linear interpolation or linear regression. The algorithm also incorporates the notion of variability in the time series and is shown to be more effective than CUSUM [14] based change detection techniques and the change detection technique proposed by Lunetta et al. in [16]. In our paper, we will propose another novel model-free segmentation and compare it quantitatively with recursive merging and our adapted model-based algorithm.

3. Change Detection Techniques

This section describes the two change detection methods proposed in this paper. In Section 3.1 we propose a model-based segmentation technique inspired by a framework proposed in [12]. In Section 3.2 we propose a novel, simple and efficient model-free change detection algorithm which offers scalability and robustness to varying characteristics of time series across the globe.

All these approaches take as input the vegetation index time series and the annual season length for a location and give as output the corresponding change score and change point. The locations under study can be ranked according to the change score given by each algorithm. A good algorithm will give higher ranking to the locations that are more likely to have changed.

Following are the list of notations used in this paper. Let D be a data set with N land locations each of which has a time series of length T. The time series for a location corresponds to T 16-day EVI observations at that location. We also define the following notation:

p: season length (here it is 23)
Y: the number of years of data in the data set $=\frac{T}{p}$
n_i: an individual location n_{ij} : EVI value at time j for the location n_i.
$b_{i1}, \ldots, b_{iy}$: list of annual cycles where, $b_{i1} = [n_{i,1}, n_{i,2}, \ldots, n_{i,23}]$, $b_{i2} = [n_{i,24}, \ldots, n_{i,46}]$,

3.1. **Model-Based Segmentation Algorithm.** This approach follows a top-down segmentation strategy and is inspired from a framework proposed by Guralnik and Srivastava [12]. The technique follows an iterative algorithm that fits a model to a time segment, and uses a likelihood criterion to determine if the segment should be partitioned further, i.e. if it contains a new change-point. In other words, the likelihood criterion determines the statistical significance that a given time series should either be defined using a different set of model parameters or two different models. The need for two different models or a different set of parameters indicates that the time series contains a change point.

In Algorithm 1 we provide the general framework of the change detection scheme and provide specific details in the following paragraph.

```
Let $p$ be the seasonal length
for each time series $ts$ in a given dataset do
  Consider the entire time series $[n_{i,1}, n_{i,2}, n_{i,3}, \ldots, n_{i,T}]$ as a single segment
  Choose a *model that best* fits the time series $ts$
  Calculate the *error of model* $L$ from the original time series
  for each possible candidate timestamp $t = p \times j$, where $j \in [2, 3, \ldots, \frac{length(ts)}{p} - 2]$ do
    Divide the time series into two segments at $t$
    For each segment fit the *best model separately* and calculate the *individual errors* – $L1$, $L2$
  end for
  Choose $\min(L1 + L2)$, which is the minimum of $L1 + L2$ over all possible values of $t$
  Score($S_i$) of $ts$ is $\frac{L - \min(L1+L2)}{L}$;
  Change Point for this time series $ts$ is the index where $min(L1 + L2)$ occurs
end for
```

Algorithm 1: Model-based segmentation approach for time series.

Algorithm 1 has three key aspects which we address below:

(1) *Error computation between the model and the original time series of the segment:* In [12] the error for the model was calculated using residual sum of squares between the fitted model and the original time series. However, EVI time series contains noise due to cloud contamination which results in the sudden rise or fall of values in the time series. Since the residual sum of squares is sensitive to outliers, these spikes in the EVI data make the error computation less robust. Therefore, for EVI time series, we use the Manhattan distance between the model and the segment as the error value.

(2) *Choice of Model to fit the time series:* The choice of appropriate model plays a critical role in the performance of the scheme. There are two key properties that the model should possess in this framework (i) the model should follow the seasonality of the EVI vegetation time series data (ii) the model should not follow the change very well. For example, a piecewise polynomial model, which follows both the seasonality and change, cannot be used as it would

result in low error even if applied to a time series that is changed. Also, a non-seasonal model results in high values of L, $L1$ and $L2$ and thus a lower score even for a changed time series.

In this paper, we use a harmonic model which was inspired from the work by Verbesselt et al. [18] for estimating an EVI time series. The harmonic model follows the seasonality well and is less sensitive to short term data variations and noise. The value of parameter K used in the analysis is 3. We refer to this scheme as *HM-Variability* in this paper, where HM stands for Harmonic Model.

(3) *Score for the Time Series:* In the original scheme, at every iteration, the value of the likelihood criterion was calculated until it fell below a certain threshold. However, since the focus is on a single change, we use the maximum value of the likelihood criterion obtained in the first iteration as the change score. Normalization of the likelihood estimation by L in the above scheme models the inherent variability of the time series. To evaluate the ability of *HM-Variability* to model variability, we evaluate a variation of it that does not perform the normalization step. We refer to that scheme as *HM-NoVariability*.

3.2. **Model-Free Segmentation Algorithm:** The two key characteristics of this algorithm are (i) the technique does not assume any model for the time series but rather works directly with the data values. It can therefore be applied to any periodic data without having to choose an appropriate model (ii) the technique introduces a new method to incorporate the notion of variability in the time series due to both noise in the data and climate variations.

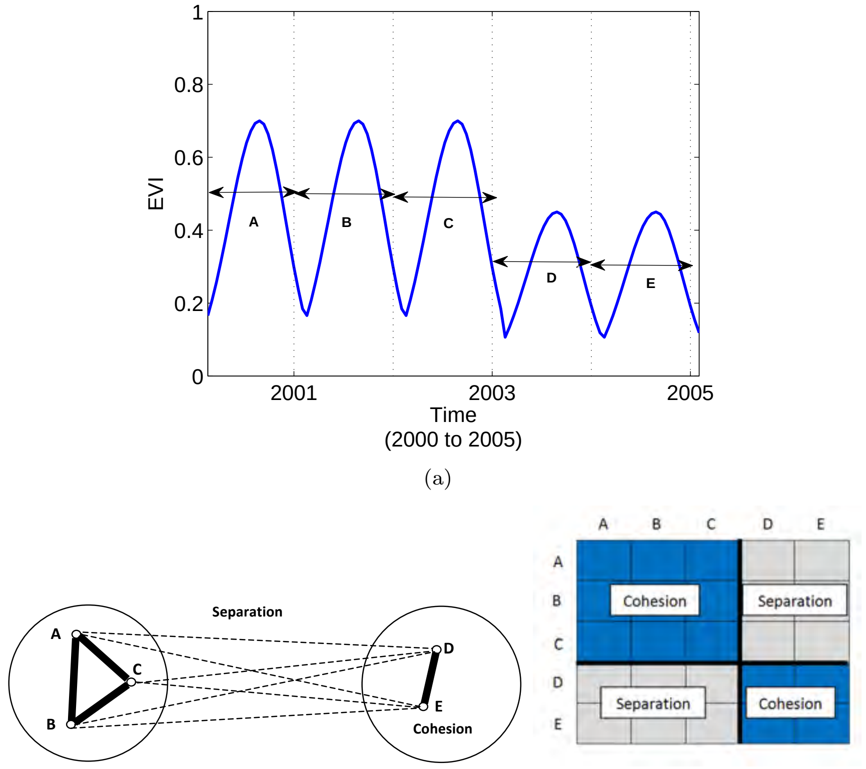
The algorithm assumes that each time series undergoes a maximum of one phenological pattern change. In particular, it assumes that a changed time series follows a certain pattern for the first few years and then follows a different pattern for the next few years. For a non-changed pixel, its time series follows the same pattern throughout its time period. There is a notion of pattern for each annual segment. This technique does not use any model and is non-parametric.

The key idea of the proposed algorithm is to find two continuous segments in the time series such that the annual years (objects) within each segment are very similar to each other while being significantly different from the objects across the segments. The boundary of the segments represents the change point in the time series. To model the similarity and differences between the objects for each segment we calculate two terms: *Cohesion* and *Separation*. The cohesion of a segment is defined as an average of the pairwise distance of all annual years within the segment. Cohesion for the time series of a pixel is defined as an average of the cohesion of both the segments (see Figure 2). The value of cohesion gives an estimate of the natural variability within the time series. Higher values of cohesion indicate higher natural variability of the time series since it means that the distances between the years in the same segment are also high. For example, the value of cohesion for a time series with no noise or fluctuations would be zero since the annual cycles would look exactly same within each segment. Likewise, the separation between two segments can be measured by the sum of the distances from objects in one segment to objects in the other segment. The value of separation indicates how distinct or well-separated the segments are to each other. The combination of cohesion and separation values indicates the amount of change in the time series with respect to the natural variation. In Algorithm 2, we describe in detail how every pixel is assigned a score and a change point.

$$C(i,t) = \frac{\frac{\sum_{p=1:t}\sum_{q=1:t} M(p,q)}{t^2-t} + \frac{\sum_{p=t+1:Y}\sum_{q=t+1:Y} M(p,q)}{(Y-t)^2-(Y-t)}}{2} \tag{1}$$

$$S(i,t) = \frac{\sum_{p=1:t}\sum_{q=t+1:Y} M(p,q)}{(Y-t)\times t} \tag{2}$$

Note that we assume that the change points occur no earlier than the end of second year and no later than the second to last year since we want at least two annual years to be present in each segment to account for inter-annual variability.



(a)

(b)

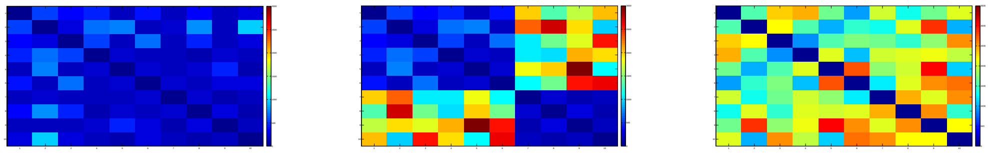
(c)

FIGURE 2. Illustration of Cohesion and Separation. (a) Time series with different years A,. . . ,E; (b) Two different circles containing 3 and 2 points (shown as small circles) represent two segments. The dark edges represent the cohesion and the dotted lines represent separation between the segments; (c) Dissimilarity matrix constructed by using the pairwise distances between years.

```
for each time series $ts$ in a given dataset do
    Create a dissimilarity matrix $M$ for the time series
    Each entry $M(q,r)$ in the matrix contains the distance between the annual segments $b_{iq}$, $b_{ir}$
    We use Manhattan distance between the vectors $b_{iq}$ and $b_{ir}$
    for each possible candidate timestamp $t = p \times i$, where $i \in [2 \cdots \frac{length(ts)}{p} - 2]$ do
        Cohesion $(C(i,t))$ with respect to $t$ is calculated as in Equation 1
        Separation $(S(i,t))$ with respect to $t$ is calculated as in Equation 2
        $Score(i,t) = S(i,t) - C(i,t)$
    end for
    $ChangeScore(i) \equiv max_t Score(i,t)$
    Change Point of this time series $ts$ is the index where $max_t Score(i,t)$ occurs
end for
```

Algorithm 2: Our proposed model-free segmentation algorithm

The key aspect of this algorithm is the use of values of cohesion and separation to distinguish a real change from the natural variability of the time series. Using Figure 3, we illustrate how the distance matrix looks for different types of series and the capability of the method to use all of the existing information to incorporate variability and assign change scores. Consider the following:



(a) Unchanged time series with low variability

(b) Changed time series

(c) Unchanged time series with high variability

FIGURE 3. Dissimilarity matrices for different kinds of time series. The blue values represent low values while the red and yellow values are higher

(1) A time series with no change and low variability: The dissimilarity matrix for such a time series is shown in Figure 3(a). Since each annual segment for such a time series would be very similar, the dissimilarity matrix consists of low values which results in low cohesion and separation, resulting in a overall lower score.
(2) A stable time series with a change: The typical dissimilarity matrix for such a time series is shown in Figure 3(b) Notice that the separation values are high and the cohesion values are low which results in a high score. Visually, notice that dissimilarity matrix has a roughly block diagonal structure since the time series have well-seperated segments.
(3) A highly variable time series with no change: The dissimilarity matrix shown in Figure 3(c) consists of all high values. If we consider only the separation between any two segments, we would obtain a high score for the time series and wrongly label it as change. However, if we consider the cohesion between the time series and the relative difference between cohesion and separation, the time series would be given a low score since all values are relatively similar. Visually also, there is no block-diagonal structure observed in the dissimilarity matrix signifying that the time series does not have well-seperated segments.

The above discussion illustrates the importance of including measures of variability in the analysis of vegetation index data set to effectively distinguish between an unusual event and an event within the normal range of variability. In this paper, we refer to the scheme using only separation as *MF-NoVariability* and using the difference of separation and cohesion i.e., $S(i,t) - C(i,t)$ as *MF-Variability*.

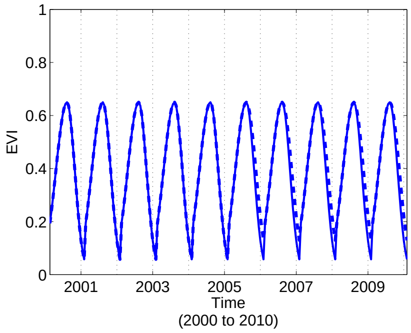
Another way to handle variability in the time series is to examine the distribution of the pairwise distance values between the objects in the same segment and object across the segment. In this paper we use the t-statistic as the scoring function. We refer to this scheme as *MF-T-stat*. The scoring function in Algorithm 2 is replaced by the score below:

$$Score = \frac{tstatistic(S(i,t), C(i,t)_{Seg1}) + tstatistic(S(i,t), C(i,t)_{Seg2})}{2}$$

4. DATA AND EVALUATION METHODOLOGY

Below, we provide details of the simulated and the MODIS EVI data sets used for evaluation.

4.1. **Simulated 16-day EVI Time Series:** Simulated EVI time series are generated by summing simulated seasonal and noise components. This procedure was adapted from [18]. The seasonal component is created using an asymmetric Gaussian function (Equation 3) for each season. Two different kinds of seasonal cycles are created by using $x \in [1, p]$ for single cycles per year and $x \in [1, \frac{p}{2}]$ for double cycles per year, where p is the season length.



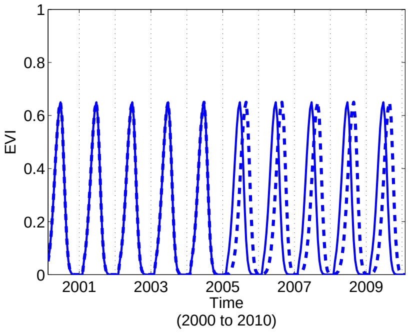


FIGURE 4. Seasonal change by changing the C_1 from 5 0(–) to 75 (dashed): c_2=100, b=12

FIGURE 5. Seasonal change by changing b from 9 (–) to 13 (dashed): $c_1 = 10$, $c_2 = 25$

$$f(x) = \begin{cases} ae^{\frac{-(b-x)^2}{c_1}} & x \geq b, \\ ae^{\frac{-(b-x)^2}{c_2}} & \text{otherwise.} \end{cases} \tag{3}$$

The parameters a and b determine the amplitude and the position of maximum or minimum with respect to the independent time variable t, while c_1 and c_2 determine the width of the left and the right hand side, respectively.

In addition to the seasonal component, the following two noise components were generated (i) *Noise_Seasonal*: Simulates the inter annual seasonal variability observed in the EVI values due to climate variations and was generated using a random number generator that follows a uniform distribution over a pre-defined range (ii) *Noise_Spike*: A noise component that was added to a pre-defined number of time stamps in each time series to simulate cloud contamination. The value of the noise component also followed a uniform distribution between a pre-defined range.

The time series were generated in the following manner:

(1) *Unchanged Time Series:* The same values of the parameters are used in Equation 3 for all individual years. Both noise components were added using the method as described above.
(2) *Changed Time Series:* The changed time series are constructed by changing the parameters within a single time series after a certain year which is chosen randomly between years 2 and 8. Different parameters impact the time series in a different way. For example, Figure 4 illustrates a pattern change introduced from fifth year onwards by changing c_1 from 10 to 100 while keeping all the other parameters fixed. The change in different parameters corresponds to different land cover changes. For example, a change in only the amplitude might represent a degradation of crop productivity or a change in only c_1 or c_2 might indicate a different cropping pattern. However, to easily compare the relative performance of different algorithms we only change the parameter b in the two different segments. Figure 5 shows the effect of changing the parameter b in the two segments.

Using the above procedure different data sets were created which differed in the amount of noise:

DS1: It is a combination of multiple data sets (Table 1) which have similar amplitude range but vary in the levels of noise. This was to simulate the areas with similar vegetation patterns but different characteristics of noise due to geographic locations, climate patterns etc. All the data sets used have the same number of changed and non-changed time series and contained both single & double cycled time series in equal proportion.

DS2: This data set was created to simulate changes occuring in different vegetation phenologies having different noise levels and extent of changes. The constructed data set is the combination of

Name	Amplitude	$Noise_{Seasonal}$	$Noise_{Spike}$	% of time stamps	Changed	Non-Changed
DS-N1	[3000,7000]	[-500,500]	[1200,1500]	10	2000	20,000
DS-N2	[3000,7000]	[-500,500]	[1700,2000]	30	2000	20,000
DS-N3	[3000,7000]	[-1000,1000]	[1200,1500]	10	2000	20,000
DS-N4	[3000,7000]	[-1000,1000]	[1700,2000]	30	2000	20,000
DS-N5	[3000,7000]	[-1500,1500]	[1200,1500]	10	2000	20,000
DS-N6	[3000,7000]	[-1500,1500]	[1700,2000]	30	2000	20,000

Table 1. Summary of different data sets used to create data set **DS1**

the data sets shown in Table 2. Notice that it contains two data sets: DS-N7 with higher amplitude & higher levels of noise and DS-N8 with lower amplitude & lower levels of noise.

Name	Amplitude	$Noise_{Seasonal}$	$Noise_{Spike}$	% of time stamps	Changed	Non-Changed
DS-N7	[3000,7000]	[-1500,1500]	[1700,2000]	30	2000	20,000
DS-N8	[1000,1500]	[-500,500]	[1200,1500]	10	2000	20,000

Table 2. Summary of different data sets used to create data set **DS2**

DS3: These data sets were created to illustrate the importance of choosing an appropriate model in the model based change detection algorithm. DS3 consists of data sets shown in Table 3. DS-N9 is constructed using Assymetric Gaussian function as in Equation 3. DS-N10 is however constructed using Wigner semicircle distribution model as in Equation 4. Changed time series are constructed by changing the values of $R \in [6, 11]$. Both these data sets have 2,000 changed and 20,000 nonchanged time series.

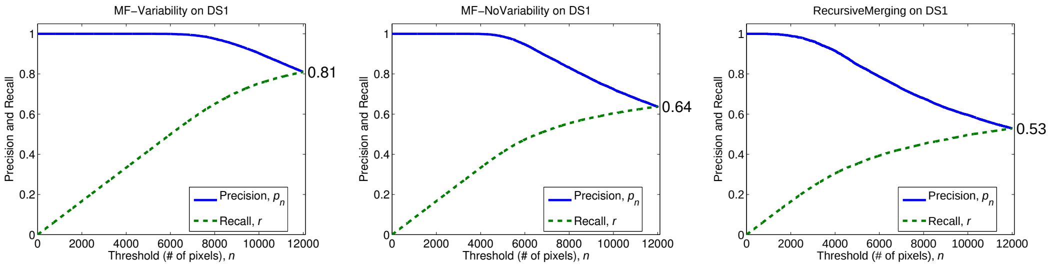
$$f(x) = \begin{cases} \frac{2}{\pi R^2}\sqrt{(R^2 - x^2)} & -R < x < R, \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

Name	Model	Amplitude	$Noise_{Seasonal}$	$Noise_{Spike}$
DS-N9	Asymmetric Gaussian	[8000,12000]	[-500,500]	[1200,1500]
DS-N10	Wigner semicircle	[8000,10000]	[-1500,1500]	[2000,2500]

Table 3. Summary of data set **DS3**

4.2. **16-day MODIS EVI Time Series:** The specific vegetation-related variable used in this analysis was the Enhanced Vegetation Index (EVI) product that serves as a surrogate for the amount of vegetation for a pixel; and is measured by the moderate resolution imaging spectroradiometer (MODIS) instrument. In this paper, the temporal coverage of the data is from the time period February 2000 – February 2010.

We selected a region in North Carolina containing 48,025 pixels of 250m resolution between North 35.99–35.3 and West 76.5–77. We refer to this data set as DSNC. This region was chosen because it is known to have a variety of changes in land cover over the past 10 years. Also, a reasonably good quality land cover classification map of this region is available from NOAA [4] at 30m resolution for 2001 and 2006 that can be used for validation. Each 250m pixel was assigned a set of 30m pixels based on the nearest neighbor and a 250m pixel was considered a change if a certain threshold (10%



(a) (b) (c)

Figure 6. Precision-Recall curve for DS1, Blue curve is Precision, Green curve is Recall. x-axis represents the number of pixels (a) *MF-Variability*; (b) *MF-NoVariability*; (c) *RecursiveMerging*

in our analysis) of the 30m pixels within that 250m pixel had different land cover labels in 2001 and 2006. Using this threshold 7,367 pixels were considered changed. More details of the ground truth generation are provided in the technical report [9].

4.3. **Evaluation Methodology.** Assume that for a time series data set D with N pixels, the change detection technique returns a list of *change scores* of length N, where each change score is a measure of the degree of change for the corresponding pixel. We also have a validation data set which consists of true labels for each of the pixels; let M be the *total* number of actual changes as determined by the validation data set. To evaluate the performance of a given change detection algorithm at rank n, we count the number of true changes in the top n ranked pixels of the sorted change scores of all the pixels, where n is the number of actual changes ($1 \leq n \leq M$). Let TP_n be the number of actual disturbances in the top n predicted disturbances, and FP_n be the number of pixels that are in the top n portion but are not actual disturbances.

We evaluate performance by examining the *sorted* list of change scores. The performance metrics are defined as follows:

$$\text{Precision, } p_n = \frac{TP_n}{TP_n + FP_n} \qquad \text{Recall, } r_n = \frac{TP_n}{M}$$

Note that as n increases, p_n will tend to decrease and r_n will increase. One specific value of interest is the one when n is equal to the number of changed pixels (validation data). At this value of n, $p_n = r_n$. Also, if the change detection algorithm does a perfect job of identifying changes, then p_n will remain at 1 upto this value of n and then start to drop for increasing values of n and r_n will linearly increase from 0 to 1 and then stay at 1 for larger values of n.

5. Experimental Results

5.1. **Observations on Simulated Data Sets:** Below we present precision and recall curves for different algorithms on DS1, DS2, DS3 and DSNC. We particularly focus on the relative capabilities of the algorithm to model natural variation. We also present results to illustrate the dependence of model based algorithms on the choice of model.

5.1.1. ***MF-Variability Significantly Outperforms MF-NoVariability and RecursiveMerging for DS1:***

The precision and Recall curves in Figure 6 for DS1 shows that *MF-Variability* significantly outperform *MF-NoVariability* and *RecursiveMerging*. The primary reason is that since dataset DS1 consists of time series with varying levels of variability (noise), the change detection algorithm must take into account the change with respect to the natural variation. Since *MF-NoVariability* does not depend on the value of cohesion, it is not able to model the natural variation in the time series.

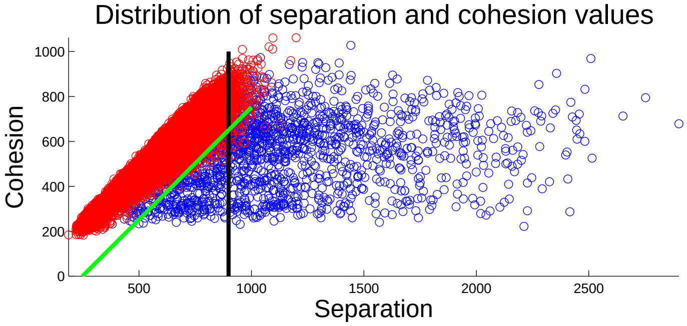


FIGURE 7. Cohesion-Separation value distribution for changed and unchanged pixels. The changed time series are represented by blue circles while the unchanged pixels are represented by red circles.

To illustrate the advantage of subtracting the cohesion from separation in *MF-Variability*, in Figure 7 we show the scatter plots of the Separation and Cohesion values for a random sample of 2,000 changed (blue circles) and 20,000 non-changed time series (red circles) from DS1. The vertical line in black shows the constant *MF-NoVariability* score of 898 and the oblique line in green shows the constant score of *MF-Variability* score of 248. Points lying to the right half of these lines will have scores higher than the respective line. These scores are chosen because they give similar number of changed events. From the Figure 7, we notice that *MF-NoVariability* will make more errors as compared to *MF-Variability* by incorrectly labelling a few unchanged time series as changed.

The discussion above illustrates that the notion of variability is important to incorporate in the change detection algorithm. Using the value of cohesion as an indicator of the natural variation, *MF-Variability* is able to significantly improve the results.

5.1.2. ***MF-T-stat outperforms MF-Variability for DS2:***

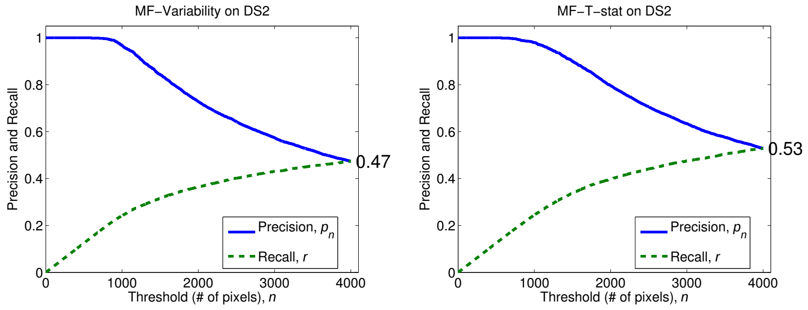
Figure 8(a) shows the precision and recall curve for *MF-Variability* on DS2. Recall that DS2 consists of two different kinds of vegetation patterns: (i) time series with higher amplitude and higher levels of noise (DS-N7) (ii) time series with lower amplitude and lower level of noise (DS-N8). Table 4 shows the number of true and false positives from the individual data sets DS-N7 and DS-N8 when *MF-Variability* is used on DS2. It is seen that only 325 points out of 2,000 changed points are recalled from the data set DS-N8. Also, notice that almost all the false positives are from the dataset DS-N7. This illustrates that because of the higher levels of noise present in DS-N7 and smaller number of changes in DS-N8, *MF-Variability* gives a higher score to unchanged time series in DS-N7 than compared to changed time series in DS-N8. It is therefore important to design a scoring mechanism which takes into account the difference in variation observed in the time series due to different phenological characteristics. As discussed in Section 4, Figure 8(a) illustrates how *MF-T-stat* models the variance of the distribution in cohesion and separation values to significantly improve the results. Table 5 further illustrates that the *MF-T-stat* is able to recall many more points from DS-N8 as compared to *MF-Variability*.

5.1.3. ***HM-Variability outperforms HM-NoVariability for DS2:***

Figure 9 shows the precision recall curve for *HM-Variability* and *HM-NoVariability*. It is seen that *HM-Variability* significantly outperforms *HM-NoVariability*. The primary reason for better performance of *HM-Variability* is similar to as explained above for the comparison of *MF-T-stat* and *MF-Variability*. The normalization step in *HM-Variability* helps to model the difference in variability of the two different combined data sets.

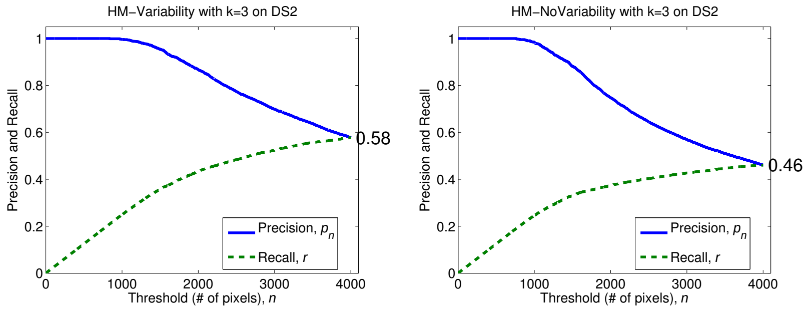
5.1.4. ***Model Choice plays a critical role in the performance of Model Based Algorithm:***

To illustrate the importance of model choice, we show results on DS3 which consists of time series generated from two different models: asymmetric Gaussian (DS-N9) and Wigner Semicircle



(a) (b)

Figure 8. Precision-Recall curve for DS2 (a) *MF-Variability* (b) *MF-T-stat*



(a) (b)

Figure 9. Precision-Recall curve for DS2 (a) *HM-Variability* (b) *HM-NoVariability*

TP or FP	DS-N7	DS-N8
TP	1569	325
FP	2102	4

Table 4. *MF-Variability*

TP or FP	DS-N7	DS-N8
TP	1328	828
FP	925	918

Table 5. *MF-T-stat*

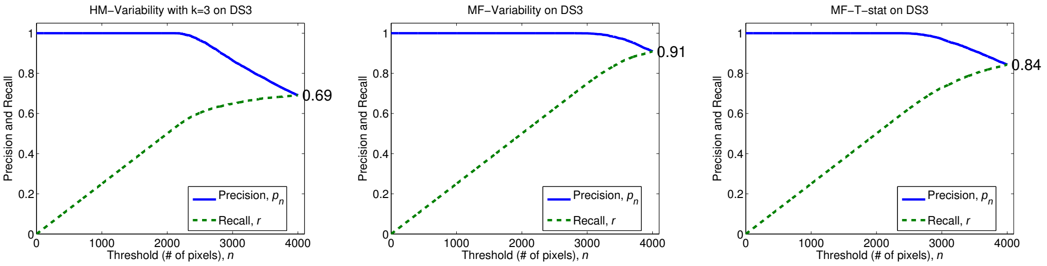
Distribution (DS-N10), as mentioned in Section 4. On this data set, *MF-Variability* significantly outperforms *HM-Variability* as shown in Figure 10. The primary reason is that since the harmonic model used in *HM-Variability* does not appropriately model the time series in DS-N10, the error computation is not accurate. In particular, the error between the fit and the original time series is particularly high for time series in DS-N10, resulting in lower score being assigned to such time series due to the normalization step in *HM-Variability*. This is also represented in the number of true and false positives detected by the algorithms for the individual data sets present in DS3 (shown in Table 6 and Table 7). Note that only a few (767 out of 2,000) changed points are recalled from the data set DS-N10. Also, notice that all of the false positives are from the data set DS-N9. Therefore, the choice of the model in model-based algorithm is critical to its performance. On the other hand, *MF-Variability* does not require any knowledge of model or choice of parameter and therefore is robust to different phenologies and characteristics of time series globally. This is one of the key properties of the *MF-Variability* algorithm for its application in global land cover change detection.

TP or FP	DS-N9	DS-N10
TP	2000	767
FP	1233	0

Table 6. *HM-Variability*

TP or FP	DS-N9	DS-N10
TP	2000	1637
FP	0	363

Table 7. *MF-Variability*



(a)

(b)

(c)

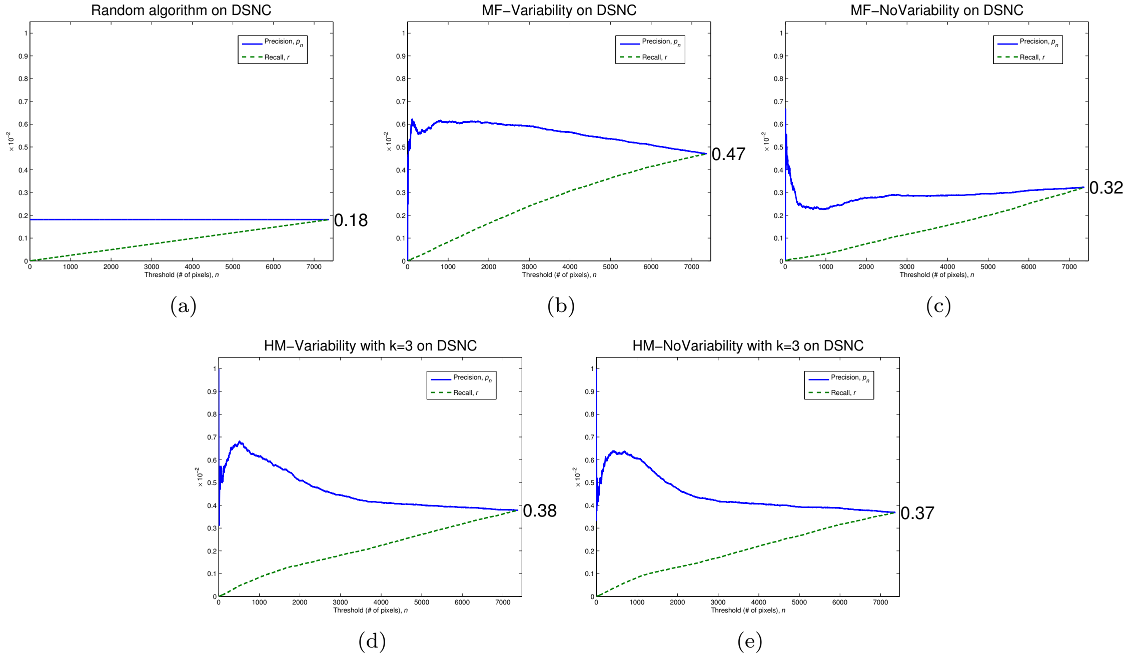
Figure 10. Precision-Recall curves for DS3 (a) *HM-Variability*; (b) *MF-Variability*; (c) *MF-T-stat*

5.2. **Observations on real dataset: DSNC:.** Figure 11 shows the precision recall curve for different algorithms on DSNC. It is observed that none of the algorithms perform very well on this dataset. This is primarily due to various issues associated with the validation data set which complicates the evaluation. First, the resolution difference between the label dataset (30m) and the MODIS EVI data set (250m) results in inaccuracy in assigning the proper set of labels to each 250m pixel. Also, determining the threshold for the number of 30m pixels required to have changed for each 250m pixel to be considered as change is challenging. For example, though a conversion of 10% of 30m pixels within a 250m pixel from forest to barren land could be strongly reflected in the 250m EVI signal, a 10% conversion from forest to pasture might not be reflected. Additional challenges arise from the inability of the EVI signal to distinguish between some particular land cover types. A pixel classified as Secondary Forest in 2001 and Mixed Forest in 2006 is considered changed according to the validation data set but might not show a perceptible change in its EVI signal and thus would not be detected by the change detection algorithm. Conversely, certain changes such as double cropping to single cropping cycles which are clearly reflected in the EVI signal are not considered change according to the ground truth because they have the same LCC label. Such pixels detected by the algorithm are considered as false positives by the evaluation methodology and thus reduces the observed performance of the algorithms.

Despite these challenges, note that all the algorithms still do significantly better than the random curve shown in Figure 11. Also, it is observed that *MF-Variability* performs the best and significantly better than *MF-NoVariability*. However, it is difficult to make precise statements about the relative performance given the uncertainity associated with the validation labels.

6. Conclusion and Future Work

In this paper, we presented two time series segmentation techniques that can be used to identify the pattern changes in the vegetation index time series. The results of this study also demonstrate the importance of modeling the natural variation in the time series for accurately estimating the significance of the change in the EVI signal. Both the techniques significantly outperformed another recently proposed technique by Boriah et al [6]. The proposed model-based segmentation algorithm was shown to be sensitive to the choice of model, however the model-free segmentation algorithm requires no model and gives comparable or better results. The proposed model-free segmentation



(a)

(b)

(c)

(d)

(e)

Figure 11. $x - axis$ shows the number of pixels considered; $y - axis$ shows the precision (range 0-1) and recall (range 0-1); Precision-Recall curves on real data set (a) *Random Algorithm* (b) *MF-Variability*; (c) *MF-NoVariability*; (d) *HM-Variability*; (e) *HM-NoVariability*

algorithm has been applied globally at 1km EVI to detect various land cover changes [10] such as forest to farmland conversions, change in cropping patterns, urbanization and the results are publicly available via the online platform ALERTS [1]. The results indicate the ability of the algorithm to provide rapid, inexpensive, robust, scalable and precise detection of land use change [2].

The proposed algorithms assume that only one pattern change occurs in the time series. However, the ability to find multiple changes becomes critical as the length of the time series increases with the continuous collection of satellite data. Therefore the existing techniques ought to be extended to detect multiple changes. This could be challenging since the presence of multiple change points might hinder the effective detection of the first change point using the top down segmentation approaches. In addition, the techniques need to be adapted to discover changes even in the presence of other changes such as gradual or abrupt drops. BFAST, a recently proposed technique [18] outlines a framework to detect such changes, however the technique is computationally expensive and hence not scalable for global application. The proposed techniques could be extended using similar frameworks to detect such changes. Also, our techniques assume that the pattern changes occur at the yearly boundaries which is not always true in the land cover change domain.

Acknowledgements

The research described here is supported by the Planetary Skin Institute, NSF Grant IIS-0713227, NSF Grant IIS-0905581, NASA grant NNX09AL60G and the University of Minnesota MN Futures Program. Access to computing facilities was provided by the University of Minnesota Supercomputing Institute. We are grateful to Ivan Brugere, Xi Chen, Divya Alla, Sruthi Reddy Vangala and Yashu Chamber for their helpful comments.

References

[1] Planetary Skin.
`http://www.ourplanetaryskin.org/`.

[2] Monitoring Forests: Seeing the world for the trees.
`http://www.economist.com/node/17730208/`.

[3] Land Processes Distributed Active Archive Center.
`http://edcdaac.usgs.gov`.

[4] Coastal Change Analysis Program Regional Land Cover.
`http://www.csc.noaa.gov/digitalcoast/data/ccapregional/`.

[5] S. Boriah. *Time Series Change Detection: Algorithms for Land Cover Change*. PhD thesis, University of Minnesota, 2010.

[6] S. Boriah, V. Kumar, M. Steinbach, C. Potter, and S. Klooster. Land cover change detection: A case study. In *KDD 2008: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 857–865, 2008.

[7] V. Chandola, D. Hui, L. Gu, B. Bhaduri, and R. R. Vatsavai. Using time series segmentation for deriving vegetation phenology indices from modis ndvi data. In *Proceedings of the 2010 IEEE International Conference on Data Mining Workshops*, ICDMW '10, pages 202–208, Washington, DC, USA, 2010. IEEE Computer Society.

[8] P. Coppin, I. Jonckheere, K. Nackaerts, B. Muys, and E. Lambin. Digital change detection methods in ecosystem monitoring: a review. *International Journal of Remote Sensing*, 25(9):1565–1596, 2004.

[9] A. Garg, L. Manikonda, S. Kumar, V. Krishna, S. Boriah, M. Steinbach, V. Kumar, D. Toshniwal, C. Potter, and S. Klooster. Pre-processing of the validation data used in the paper titled "Model-Free Time Series Segmentation Approach for Land Cover Change Detection". *Technical Report 11-017, Computer Science Department, University of Minnesota*, 2011.

[10] A. Garg, V. Mithal, Y. Chamber, I. Brugere, V. Chaudhari, M. Dunham, V. Krishna, S. Krishnamurthy, S. Vangala, S. Boriah, M. Steinbach, V. Kumar, A. Cho, J. Stanley, T. Abraham, J. C. Castilla-Rubio, C. Potter, and S. Klooster. GOPHER: Global observation of planetary health and ecosystem resources. In *IGARSS 2011: Proceedings of the IEEE Geoscience and Remote Sensing Symposium*, 2011.

[11] L. Giglio, G. R. van der Werf, J. T. Randerson, G. J. Collatz, and P. Kasibhatla. Global estimation of burned area using modis active fire observations. *Atmospheric Chemistry and Physics*, 6(4):957–974, 2006.

[12] V. Guralnik and J. Srivastava. Event detection from time series data. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '99, pages 33–42, New York, NY, USA, 1999. ACM.

[13] D. M. Hawkins. Point estimation of the parameters of piecewise regression models. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 25(1):pp. 51–57, 1976.

[14] K. Jan, B. F. Paulo, and S. Peter. Cumulative sum charts - a novel technique for processing daily time series of modis data for burnt area mapping in portugal. In *In MultiTemp 2007: International Workshop on the Analysis of Multi-temporal Remote Sensing Images*, pages 1–6, 2007.

[15] E. Keogh, S. Chu, D. Hart, and M. Pazzani. Segmenting time series: A survey and novel approach. In *In an Edited Volume, Data mining in Time Series Databases. Published by World Scientific*, pages 1–22. Publishing Company, 1993.

[16] R. S. Lunetta, J. F. Knight, J. Ediriwickrema, J. G. Lyon, and L. D. Worthy. Land-cover change detection using multi-temporal modis ndvi data. *Remote Sensing of Environment*, 105(2):142 – 154, 2006.

[17] V. Mithal, A. Garg, S. Boriah, M. Steinbach, V. Kumar, C. Potter, S. Klooster, and J. C. Castilla-Rubio. Monitoring global forest cover using data mining. *ACM Trans. Intell. Syst. Technol.*, 2:36:1–36:24, July 2011.

[18] J. Verbesselt, R. Hyndman, A. Zeileis, and D. Culvenor. Phenological change detection while accounting for abrupt and gradual trends in satellite image time series. *Remote Sensing of Environment*, 114(12): 2970 – 2980, 2010.

SPARSE MACHINE LEARNING METHODS FOR UNDERSTANDING LARGE TEXT CORPORA

LAURENT EL GHAOUI*, GUAN-CHENG LI*, VIET-AN DUONG**, VU PHAM***,
ASHOK SRIVASTAVA****, AND KANISHKA BHADURI****

Abstract. Sparse machine learning has recently emerged as powerful tool to obtain models of high-dimensional data with high degree of interpretability, at low computational cost. This paper posits that these methods can be extremely useful for understanding large collections of text documents, without requiring user expertise in machine learning. Our approach relies on three main ingredients: (a) multi-document text summarization and (b) comparative summarization of two corpora, both using sparse regression or classification; (c) sparse principal components and sparse graphical models for unsupervised analysis and visualization of large text corpora. We validate our approach using a corpus of Aviation Safety Reporting System (ASRS) reports and demonstrate that the methods can reveal causal and contributing factors in runway incursions. Furthermore, we show that the methods automatically discover four main tasks that pilots perform during flight, which can aid in further understanding the causal and contributing factors to runway incursions and other drivers for aviation safety incidents.

1. Introduction

Sparse machine learning refers to a collection of methods to learning that seek a trade-off between some goodness-of-fit measure and sparsity of the result, the latter property allowing better interpretability. In a sparse learning classification task for example, the prediction accuracy or some other classical measure of performance is not the sole concern: we also wish to be able to explain what the classifier means to a non-expert. Thus, if the classification task involves say gene data, one wishes to provide not only a high-performance classifier, but one that only involves a few genes, allowing biologists to focus their research efforts on those specific genes.

There is an extensive literature on the topic of sparse machine learning, with terms such as compressed sensing [12, 5], l_1-norm penalties and convex optimization [42], often associated with the topic. Successful applications of sparse methods have been reported, mostly in image and signal processing, see for example [15, 28, 31]. Due to the intensity of research in this area, and despite an initial agreement that sparse learning problems are more computationally difficult than their non-sparse counterparts, many very efficient algorithms have been developed for sparse machine learning in the recent past. A new consensus might soon emerge that sparsity constraints or penalties actually *help* reduce the computational burden involved in learning.

Our paper makes the claim that sparse learning methods can be very useful to the *understanding* large text databases. Of course, machine learning methods in general have already been successfully applied to text classification and clustering, as evidenced for example by [21]. We will show that sparsity is an important added property that is a crucial component in any tool aiming at providing interpretable statistical analysis, allowing in particular efficient multi-document summarization, comparison, and visualization of huge-scale text corpora.

*EECS Dept., UC Berkeley, (guanchengli,elghaoui)@eecs.berkeley.edu;
**Ecole des Mines d'Alès, School of Production & Systems Engineering, viet-an.duong@mines-ales.org;
***University of Science, VNU-HCMC, Ho-Chi-Minh City, Vietnam, ptvu@acm.org;
****System-Wide Safety and Assurance Technologies Project, NASA,
(ashok.n.srivastava,kanishka.bhaduri-1)@nasa.gov.

To illustrate our approach we focus here on Aviation Safety Reporting System (ASRS) text reports, which is a crucial component of the continuing effort to maintain and improve aviation safety. The text reports are written by members of the flight crew, air traffic controllers, and others on a voluntary basis. The reports are de-identified so that the author and other specific information regarding the flight is not revealed. Each report is a small paragraph describing any incident that the author wishes to discuss and is assigned a category among a set of pre-defined ones by a team of ASRS experts. The ASRS database consists of about 100,000 reports spanning approximately 30 years. Although the report intake fluctuates on a monthly basis, the ASRS report intake for March 2011 was 6148 reports. ASRS data are used by experts to identify deficiencies in the National Aviation System so that they can be corrected. The data are also used to further deepen our understanding of human factors issues in aviation which is a critical component of aviation safety. It is widely thought that over two-thirds of all aviation accidents and incidents have their roots in human performance errors [1].

The ASRS data contains several of the crucial challenges involved under the general banner of "large-scale text data understanding". First, its scale is huge, and growing rapidly, making the need for automated analyses of the processed reports more crucial than ever. Another issue is that the reports themselves are far from being syntactically correct, with lots of abbreviations, orthographic and grammatical errors, and other shortcuts. Thus we are not facing a corpora with well-structured language having clearly defined rules, as we would if we were to consider a corpus of laws or bills or any other well-redacted data set. Finally, in many cases we do not know in advance what to look for because the goal is to discover precursors to aviation safety incidents and accidents. In other words, the task is not about search, and finding a needle in a haystack: in many cases, we cannot simply to monitor the emergence or disappearance of a few keywords that would be known in advance. Instead the task resembles more one of trying to visualize the haystack itself, compare various parts of it, or summarize some areas.

In examining the ASRS data, we would like to be able to pinpoint some emerging issues, highlight some trends, broken down by time, type of flight, incident, or airport. For example, the class of incidents known as "runway incursion" might occur more frequently at some airports; runway incursions might be due to different causes, necessitating differentiated responses (such as improved ground lighting, or changes in taxiway configurations). How can we quickly figure out the *type* of runway incursions involved at each airport, and respond accordingly?

Our paper is organized as follows. Section 2 is devoted to a review of some of the main models and algorithms in sparse machine learning. We explain how these methods can be used in text understanding in section 3. Section 4 illustrates the approach in the context of ASRS data, and also reviews some prior work on this specific data set. Although our focus here is on ASRS data, most of the approaches depicted here have been developed in the context of news data analysis, see [18, 30, 6].

2. Sparse Learning Methods

In this section we review some of the main algorithms of sparse machine learning.

2.1. Sparse classification and regression.

2.1.1. *The LASSO.* Perhaps the most well known example of sparse learning is the variant of least-squares known as the LASSO [41], which takes the form

$$\min_{\beta} \|X^T\beta - y\|_2^2 + \lambda\|\beta\|_1, \tag{1}$$

where X is a $n \times m$ data matrix (with each row a specific feature, each column a specific data point), y is a m-dimensional response vector, and $\lambda > 0$ is a parameter. The l_1-norm penalty encourages

[1] See http://asrs.arc.nasa.gov for more information on the ASRS system. The text reports are available on this website along with analyses performed by the ASRS analysts.

the regression coefficient vector β to be sparse, bringing interpretability to the result. Indeed, if each row is a feature, then a zero element in β at the optimum of (1) implies that that particular feature is absent from the optimal model. If λ is large, then the optimal β is very sparse, and the LASSO model then allows to select the few features that are the best predictors of the response vector.

2.1.2. *Solving the LASSO.* The LASSO problem looks more complicated than its classical least-squares counterpart. However, there is mounting evidence that, contrary to intuition, the LASSO is substantially *easier* to solve than least-squares, at least for high values of λ. As shown later, in typical applications to text classification, a high value of λ is desired, which is precisely the regime where the LASSO is computationally very easy to solve.

Many algorithms have been proposed for LASSO; at present it appears that, in text applications with sparse input matrix X, a simple method based on minimizing the objective function of (1) one coordinate of β at a time is extremely competitive [16, 33]. The so-called safe feature elimination procedure [14], which allow to cheaply detect that some of the components of β will be zero at optimum, enables to treat data sets having millions of terms and documents, at least for high values of λ.

2.1.3. *Other loss functions.* Similar models arise in the context of support vector machines (SVM) for binary classification, where the sparse version takes the form

$$\min_{\beta,b} \; \frac{1}{m}\sum_{i=1}^{m} h(y_i(x_i^T\beta + b)) + \lambda\|\beta\|_1, \tag{2}$$

where now y is the vector of ±1's indicating appartenance to one of the classes, and h is the so-called hinge loss function, with values $h(t) = \max(0, 1-t)$. At optimum of problem (2), the above model parameters (β, b) yield a classification rule, *i.e.* predict a label $\hat{y}$ for a new data point x, as follows: $\hat{y} = \mathbf{sign}(x^T\beta + b)$. A smooth version of the above is sparse logistic regression, which obtains upon replacing the hinge loss with a smooth version $l(t) = \log(1+e^{-t})$. Both of these models are useful but somewhat less popular than the LASSO, as state-of-the-art algorithms are have not yet completely caught up. For our text applications, we have found that LASSO regression, although less adapted to the binary nature of the problem, is still very efficient [30].

2.2. **Sparse principal component analysis.**

2.2.1. *The model.* Sparse principal component analysis (Sparse PCA, see [48, 47] and references therein) is a variant of PCA that allows to find sparse directions of high variance. The sparse PCA problem can be formulated in many different ways, one of them (see [39, 27]) involves a low-rank approximation problem where the sparsity of the low-rank approximation is penalized:

$$\min_{p,q} \; \|M - pq^T\|_F^2 + \lambda\|p\|_1 + \mu\|q\|_1, \tag{3}$$

where M is the data matrix, $\|\cdot\|_F$ is the Frobenius norm, and $\mu \geq 0, \lambda \geq 0$ are parameters.

The model above results in a rank-one approximation to M (the matrix pq^T at optimum), and vectors p, q are encouraged to be sparse due to the presence of the l_1 norms, with high value of the parameters λ, μ yielding sparser results. Once sparse solutions are found, then the rows (resp. columns) in M corresponding to zero elements in p (resp. in q) are removed, and problem (3) is solved with the reduced matrix as input. If M is a term-by-document matrix, the above model provides sparsity in the feature space (via p) and the document space (via a "topic model" q), allowing to pinpoint a few features and a few documents that jointly "explain" data variance.

2.2.2. *Algorithms.* Several algorithms have been proposed for the above problem, for example [23, 39, 8]. In practice, one algorithm that is very efficient (although it is only guaranteed to converge to a local minimum) consists in solving the above problem alternatively over p, q many times [39]. This leads to a modified power iteration method

$$p \to P(S_\lambda(Mq)), \;\; q \to P(S_\mu(M^Tq)),$$

where P is the projection on the unit circle (assigning to a non-zero vector v its scaled version $v/\|v\|_2$), and for $t \geq 0$, S_t is the "soft thresholding" operator (for a given vector v, $S_t(v) = \mathbf{sign}(v) \max(0, |v| - t)$, with operations acting component-wise). We can replace the soft thresholding by hard thresholding, for example zeroing out all but a fixed number of the largest elements in the vector involved.

With $\lambda = \mu = 0$ the original power iteration method for the computation of the largest singular value of M is recovered, with optimal p, q the right- and left- singular vectors of M. The presence of λ, μ modifies these singular vectors to make them sparser, while maintaining the closeness of M to its rank-one approximation. The hard-thresholding version of power iteration scales extremely well with problem size, with greatest speed increases over standard power iteration for PCA when a high degree of sparsity is asked for. This is because the vectors p, q are maintained to be extremely sparse during the iterations.

2.2.3. *Thresholded PCA.* An alternative to solving the above that was proposed earlier for sparse PCA is based on solving a classical PCA problem, then thresholding the resulting singular vectors so that they have the desired level of sparsity. For large-scale data, PCA is typically solved with power iteration, so the "thresholded PCA" algorithm is very similar to the above thresholded power iteration for sparse PCA. The only difference is in how many times thresholding takes place. Note that in practice, the thresholded power iteration for sparse PCA is much faster than its plain counterpart, since we are dealing with much sparser vectors as we perform the power iterations.

2.3. **Sparse graphical models.**

2.3.1. *Covariance selection.* Sparse graphical modeling seeks to uncover a graphical probabilistic model for multivariate data that exhibits some sparsity characteristics. One of the main examples of this approach is the so-called sparse covariance selection problem, with a Gaussian assumption on the data (see [34], and related works such as [17, 29, 45, 40, 26, 24]). Here we start with a $n \times n$ sample covariance matrix S, and assuming the data is Gaussian, formulate a variant to the corresponding maximum likelihood problem:

$$\max_X \log\det X - \mathbf{Tr} SX - \lambda \|X\|_1, \tag{4}$$

where $\lambda > 0$ is a parameter, and $\|X\|_1$ denotes the sum of the absolute values of all the entries in the $n \times n$ matrix variable X. Here, $\mathbf{Tr} SX$ is the scalar product between the two symmetric matrices S and X, that is, the sum of the diagonal entries in the matrix product SX. When $\lambda = 0$, and assuming S is positive-definite, the solution is $X = S^{-1}$. When $\lambda > 0$, the solution X is always invertible (even if S is not), and tends to have many zero elements in it as λ grows. A zero element in the (i, j) entry of X corresponds to the conditional independence property between nodes i and j; hence sparsity of X is directly related to that of the conditional independence graph, where the absence of an edge denotes conditional independence.

2.3.2. *Solving the covariance selection problem.* The covariance selection problem is much more challenging than its classical counterpart (where $\lambda = 0$), which simply entails inverting the sample covariance matrix. At this point it appears that one of the most competitive algorithms involves solving the above problem one column (and row) of X at a time. Each sub-problem can be interpreted as a LASSO regression problem between one particular random variable and all the others [34, 17]. Successful applications of this approach include Senate voting [34] and gene data analysis [34, 11]

Just as in the PCA case, there is a conceptually simple alorithm, which relies on thresholding. If the covariance matrix is invertible, we simply invert it and threshold the elements of the inverse. Some limited evidence points to the statistical superiority of the sparse approach (based on solving problem (4)) over its thresholded counterpart. On the computational front however, and contrarily to the models discussed in the previous two sections, the thresholding approach remains computationally competitive, although still very challenging in the high-dimensional case.

2.4. **Thresholded models.** The algorithms in sparse learning are built around the philosophy that sparsity should be part of the model's formulation, and not produced as an afterthought. Sparse modeling is based on some kind of direct formulation of the original optimization problem, involving, typically, an l_1 penalty. As a result of the added penalty, sparse models have been originally thought to be substantially more computationally challenging than their non-penalized counterparts.

In practice, sparse results can be obtained via the use of *any* learning algorithm, even one that is not necessarily sparsity-inducing. Sparsity is then simply obtained via thresholding the result. This is the case for example with naïve Bayes classification, or Latent Dirichlet Allocation (LDA). In the case of LDA, the result is a probability distribution on all the terms in the dictionary. Only the terms with the highest weights are retained, which amounts in effect to threshold the probability distribution. The notion of *thresholded models* refers to the approach of applying a learning algorithm and obtaining sparsity with a final step of thresholding.

The question about which approach, "direct" sparse modeling or sparse modeling via thresholding, works better in practice, is a natural one. Since direct sparse modeling appears to be more computationally challenging, why bother? Extensive research in the least-squares case shows that thresholding is actually often sub-optimal [30]. Similar evidence has been reported on the PCA case [47]. Our own experiments in section 4 support this viewpoint.

There is an added benefit to direct sparse modeling—a computational one. Originally thresholding was considered as a computational shortcut. As we argued above for least-squares, SVM and logistic regression, and PCA, sparse models can be actually surprisingly easier to solve than classical models; at least in those cases, there is no fundamental reason for insisting on thresholded models, although they can produce good results. For the case of covariance selection, the situation is still unclear, since direct sparse modeling via problem (4) is still computationally challenging.

The above motivates many researchers to "sparsify" existing statistical modeling methodologies, such as Latent Dirichlet Allocation [4]. Note that LDA also encodes a notion of sparsity, not in the feature space, but on the document (data) space: it assumes that each document is a mixture of a small number of topics, where the topic distribution is assumed to have a Dirichlet prior. Thus, depending on the concentration parameter of this prior, a document comprised of a given set of words may be effectively restricted to having a small number of topics.

This notion of sparsity (document-space sparsity) does not constrain the number of features active in the model, and does not limit overall model complexity. As a result, in LDA, the inclusion of terms that have little discrimination power between topics (such as 'and', 'the', etc.) may fall into multiple topics unless they are eliminated by hand. Once a set of topics is identified the most descriptive words are depicted as a list in order of highest posterior probability given the topic. As with any learning method, thresholding can be applied to this list to reveal the top most descriptive words given a topic. It may be possible to eliminate this thresholding step using a modified objective function with an appropriate sparsity constraint. This is an area of very active research, as evidenced by [13].

3. Application to Text Data

3.1. **Topic summarization.** Topic summarization is an extensive area of research in natural language processing and text understanding. For a recent survey on the topic, see [7]. There are many instances of this problem, depending on the precise task that is addressed. For example the focus could be to summarize a single unit of text, or summarize multiple documents, or summarize two classes of documents in order to produce the summaries that offer the best contrast. Some further references to summarization include [19, 20, 32].

The approach introduced in [18] and [30] relies on LASSO regression to produce a summary of a particular topic as treated in multiple documents. This is part of the *extraction* task within a summarization process, where relevant terms are produced and given verbatim [7]. Using predictive models for topic summarization has a long history, see for example [37]; the innovation is the systematic reliance on *sparse* regression models.

The basic idea is to divide the corpora in two classes, one that corresponds to the topic, and the other to the rest of the text corpora. For example, to provide the summary of the topic "China" in a corpora of news articles from *The New York Times* over a specific period, we may separate all the paragraphs that mention the term "china" (or related terms such as "chinese", "china's", etc) from the rest of the paragraphs. We then form a numerical, matrix representation X (via, say, TF-IDF scores) of the data, and form a "response" vector (with 1's if the document mentions China and -1 otherwise). Solving the LASSO problem (1) leads to a vector β of regressor coefficients, one for each term of the dictionary. Since LASSO encourages sparsity, many elements of β are zero. The non-zero elements point to terms in the dictionary that are highly predictive of the appearance of "china" in any paragraph in the corpus.

The approach can be used to contrast to set of documents. For example, we can use it to highlight the terms that allow to best distinguish between two authors, or two news sources on the same topic.

Topic summarization is closely related to *topic modeling* via Latent Dirichlet Allocation (LDA) [4], which finds on a latent probabilistic model to produce a probability distribution of all the words. Once the probability distribution is obtained, the few terms that have the highest probability are retained, to produce some kind of summary in an unsupervised fashion. As discussed in section 2.4, the overall approach can be seen as a form of indirect, thresholding method for sparse modeling.

3.2. **Discrimination between several corpora.** Here the basic task is to find out what terms best describe the differences between two or more corpora. In a sparse classification setting, we may simply classify one of the corpora against all the others. The resulting classifier weight vector, which is sparse, then points to a short list of terms that are most representative of the salient differences between the corpora and all the others. Of course, related methods such as multi-class sparse logistic regression can be used.

3.3. **Visualization and clustering.** Sparse PCA and sparse graphical models can provide insights to large text databases. PCA itself is a widely used tool for data visualization, but as noted by many researchers, the lack of interpretability of the principal components is a challenge. A famous example of this difficulty involves the analysis of Senate voting patterns. It is well-known in political science that, in that type of data, the first two principal components explain the total variance very accurately [34]. The first component simply represents party affiliation, and accounts for a high proportion of the total variance (typically, 80%). The second component is much less interpretable.

Using sparse PCA, we can provide axes that are sparse. Concretely this means that they involve only a few features in the data. Sparse PCA thus brings an interpretation, which is given in terms of which few features explain most of the variance. Likewise, sparse graphical modeling can be very revealing for text data. Because it produces sparse graphs, it can bring an understanding as to which variables (say, terms, or sources, or authors) are related to each other and how.

4. Application to ASRS Data

4.1. **ASRS data sets.** In this section our focus is on reports from the Aviation Safety Reporting System (ASRS). The ASRS is a voluntary program in which pilots, co-pilots, other members of the flight crew, flight controllers, and others file a text report to describe any incident that they may have observed that has a bearing on aviation safety. Because the program is completely voluntary and the data are de-identified, meaning that the author, his or her position, the carrier, and other identifying information is not available in the report. After reports are submitted, analysts from ASRS may contact the author to obtain clarifications. However, the information provided by the reporter is not investigated further. This motivates the use of (semi-) automated methods for the real-time analysis of the ASRS data.

A first data set is the one used as part of the SIAM 2007 Text Mining Competition. The data consists in about 20,000 flight reports submitted by pilots after their flight. Each report is a small paragraph describing any incident that was recorded during flight, and is assigned a category (totaling 22), or type of incident. We refer to this data set as the "category" data set. In the category data set, the airport names, the time stamps and other information has been removed. The documents in this corpora were processed through a language normalization program that performs stemming, acronym expansion, and other basic pre-processing. The system also removes non-informative terms such as place names.

We have also worked with an ASRS data set of raw reports that include airport names and contain the term "runway incursion". Our goal with this data set is to focus on understanding the causal factors in runway incursions, which is an event in which one aircraft moves into the path of another aircraft during landing or takeoff. A key question that arises in the study of runway incursions is to understand whether there are significant distinguishing features of runway incursions for different airports. Although runway incursions are common, the causes may differ with each airport. These are the causal factors that enable the design of the intervention appropriate for that airport, whether it may be runway design, runway lighting, procedures, etc. Unlike the category data set, these data were not processed through a language normalization program.

4.2. **Related work on ASRS data.** In this section we list some previous work in applying data mining/machine learning methods for analyzing ASRS data, along with pointers for further research.

Text Cube [25] and Topic Cube [46] are multi-dimensional data cube structures which provide a solid foundation for effective and flexible analysis of the multidimensional ASRS text database. The text cube structure is constructed based on the TF/IDF (i.e., vector space) model while the topic cube is based on a probabilistic topic model. Techniques have also been developed for mining repetitive gapped subsequences [9], multi-concept document classification [43][44], and weakly supervised cause analysis [1]. The work in [25] has been further extended in [10] where the authors have proposed a keyword search technique. Given a keyword query, the algorithm ranks the aggregations of reports, instead of individual reports. For example, given a query "forced landing" an analyst may be interested in finding the external conditions (e.g. weather) that causes this kind of query and also find other anomalies that might co-occur with this one. This kind of analysis can be supported through keyword search, providing an analyst a ranked list of such aggregations for efficient browsing of relevant reports. In order to enrich the semantic information in a multidimensional text database for anomaly detection and causal analysis, Persing and Ng have developed new techniques for text mining and causal analysis from ASRS reports using semi-supervised learning [36] and subspace clustering [3].

Some work has also been done on categorizing ASRS reports into anomalous categories. It poses some specific challenges such as high and sparse dimensionality as well as multiple labels per document. Oza et al. [35] presents an algorithm called Mariana which learns a one-vs-all SVM classifier per anomaly category on the bag-of-words matrix. This provides good accuracy on most of the ASRS anomaly categories.

Topic detection from ASRS datasets have also received some recent attention. Shan et al. have developed the Discriminant Latent Dirichlet Allocation (DLDA) model [38], which is a supervised version of LDA. It incorporates label information into the generative model using logistic regression. Compared to Mariana, it not only has a better accuracy, but it also provides the topics along with the classification.

Gaussian Process Topic Models (GPTMs) by Agovic and Banerjee [2] is a novel family of topic models which define a Gaussian Process Mapping from the document space into the topic space. The advantage of GPTMs is that it can incorporate semi-supervised information in terms of a Kernel over the documents. It also captures correlations among topics, which leads to a more accurate topic model compared to LDA. Experiments on ASRS dataset show better topic detection compared to LDA. The experiments also illustrate that the topic space can be manipulated by changing the Kernel over documents.

4.3. **Recovering categories.** In our first experiment, we sought to understand if the sparse learning methods could perform well in a blind test. The category data did not contain category *names*, only referring to them with letter capitals. We sought to understand what these categories were about.

Category	term 1	term 2	term 3	term 4	term 5	term 6	term 7
A	MEL	install	maintain	mechanic	defer	logbook	part
B	CATA	CATN	airspace	install	MEL	AN	
C	abort	reject	ATO	takeoff	advance	TOW	pilot
D	grass	CATJ	brake	mud	veer	damage	touchdown
E	runway	taxi	taxiway	hold	tower	CATR	ground control
F	CATH	clearance	cross	hold	feet	runway	taxiway
G	altitude	descend	feet	CATF	flightlevel	autopilot	cross
H	turn	head	course	CATF	radial	direct	airway
I	knotindicator	speed	knot	slow	airspeed	overspeed	speedlimit
J	CATO	CATD	wind	brake	encounter	touchdown	pitch
K	terrain	GPWS	GP	MD	glideslope	lowaltitude	approach
L	traffic	TACAS	RA	AN	climb	turn	separate
M	weather	turbulent	cloud	thunderstorm	ice	encounter	wind
N	airspace	TFR	area	adiz	classb	classdairspace	contact
O	CATJ	glideslope	approach	high	goaraound	fast	stabilize
P	goaround	around	execute	final	approach	tower	miss
Q	gearup	land	towerfrequency	tower	contacttower	gear	GWS
R	struck	damage	bird	wingtip	truck	vehicle	CATE
S	maintain	engine	emergency	CATA	MEL	gear	install
T	smoke	smell	odor	fire	fume	flame	evacuate
U	doctor	paramedic	nurse	ME	breath	medic	physician
V	police	passenger	behave	drink	alcohol	seat	firstclass

Table 1: LASSO images of the categories: each list of terms correspond to the most predictive list of features in the classification of one category against all the others. The meaning of abbreviations is listed in Table 2.

Abbreviation	Meaning	Abbreviation	Meaning
aborted take-off	ATO	minimumdescent	MD
aircraftnumber	AN	minimumequipmentlist	MEL
airtrafficcontrol	ATC	noticestoairspace	NTA
gearwarningsystem	GWS	resolutionadvisory	RA
groundproximity	GP	trafficalertandcollisionavoidancesystem	TACAS
groundproximitywarningsystem	GPWS	takeoffclear	TOC
groundproximitywarningsystemterrain	GPWS-T	takeoffwarning	TOW
knotsindicatedairspeed	KIAS	temporaryflightrestriction	TFR
medicalemergency	ME		

Table 2: Some abbreviations used in the ASRS data.

To this end, we have solved one LASSO problem for each category, corresponding to classifying that category against all the others. As shown in Table 1, we did recover a very accurate and differentiated image of the categories. For example, the categories M, T, U correspond to the ASRS categories *Weather/Turbulence*, *Smoke/Fire/Fumes/Odor,* and *Illness*. These categories names are part of the ASRS Events Categories as defined in `http://asrs.arc.nasa.gov/docs/dbol/ASRS_Database_Fields.pdf`. This blind test indicates that the method reveals the correct underlying categories using the words in the corpus alone.

The analysis reveals that there is a singular category, labelled B. This category makes up about 50% of the total number of reports. Its LASSO images points to two terms, which happen to be two categories, A (mechanical issues) and N (airspace issues). The other terms in the list are common to either A or N. The analysis points to the fact that category is a "catch-all" one, and that many reports in it could be re-classified as A or N.

4.4. **Sparse PCA for understanding.** A first exploratory data analysis step might be to plot the data set on a pair of axes that contain a lot of the variance, at the same time maintaining some level of interpretability to each of the four directions.

We have proceeded with this analysis on the category data set. To this end we have applied a sparse PCA algorithm (power iteration with hard thresholding) to the category data matrix M (with each column an ASRS report), and obtained Fig. 1. We have not thresholded the direction q, only the direction p, which is the vector along which we project the points, so that it has at most 10 positive and 10 negative components. The sparse PCA plot shows that the data involves four

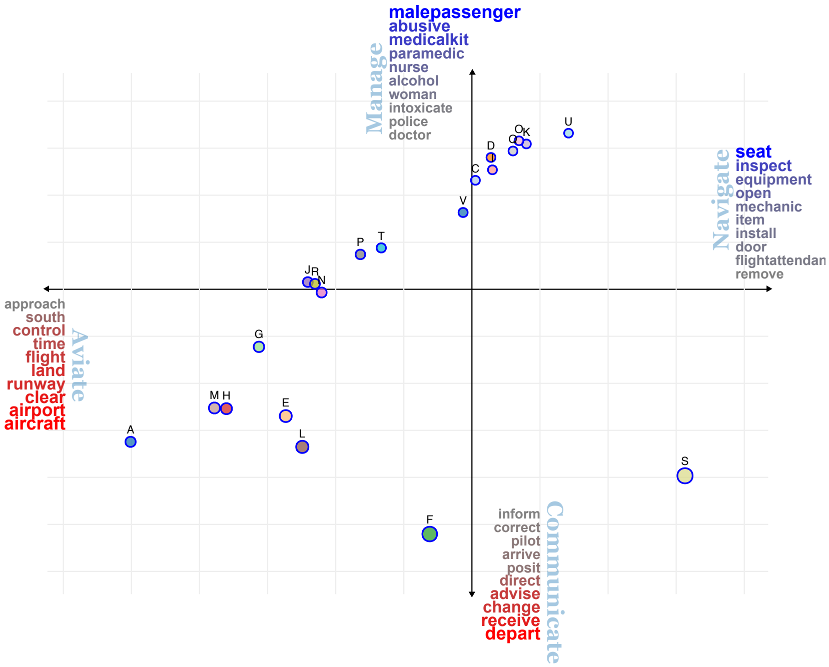


Figure 1: A sparse PCA plot of the category ASRS data. Here, each data point is a category, with size of the circles consistent with the number of reports in each category. We have focussed the axes and visually removed category B which appears to be a catch-all category. Each direction of the axes is associated with only a few terms, allowing an easy understanding of what each means. Each direction matches with one of the missions assigned to pilots in FAA documents (in light blue).

different themes, each corresponding to the positive and negative directions of the first two sparse principal components.

Without any supervision, the sparse PCA algorithm found themes that are consistent with the four missions of pilots, as is widely cited in aviation documents [22]: *Aviate*, *Navigate*, *Communicate*, and *Manage Systems*. These four actions form the basis of flight training for pilots in priority order. The first and foremost activity for a pilot is to aviate, *i.e.*, ensure that the airplane stays aloft and in control. The second priority is to ensure that the airplane is moving in the desired direction with appropriate speed, altitude, and heading. The third priority is to communicate with other members of the flight crew and air traffic control as appropriate. The final priority is to manage the systems (and humans involved) on the airplane to ensure safe flight. These high-level tasks are critical for pilots to follow because of their direct connection with overall flight safety. The algorithm discovers these four high-level tasks as the key factors in the category data set.

We validated our discovery by applying the Latent Dirichlet Allocation algorithm to the ASRS data and set the desired number of topics equal to 4. Because there is currently no method to discover the 'correct' number of topics, we use this high-level task breakdown as for an estimate of the number of topics described in the documents. While the results did not reveal the same words as sparse PCA, it revealed a similar task breakdown structure.

A a second illustration we have analyzed the runway data set. Fig 2 shows that two directions remain associated with the themes found in the category data set, namely "aviate" (negative horizontal direction) and "communicate". The airports near those directions, in the bottom left quadrant of the plot (CLE, DFW, ORD, LAX, MIA, BOS) are high-traffic ones with relatively bigger number of reports, as is indicated by the size of the circles. This is to be expected from airports where large amounts of communication is necessary (due to high traffic volume and complicated layouts).

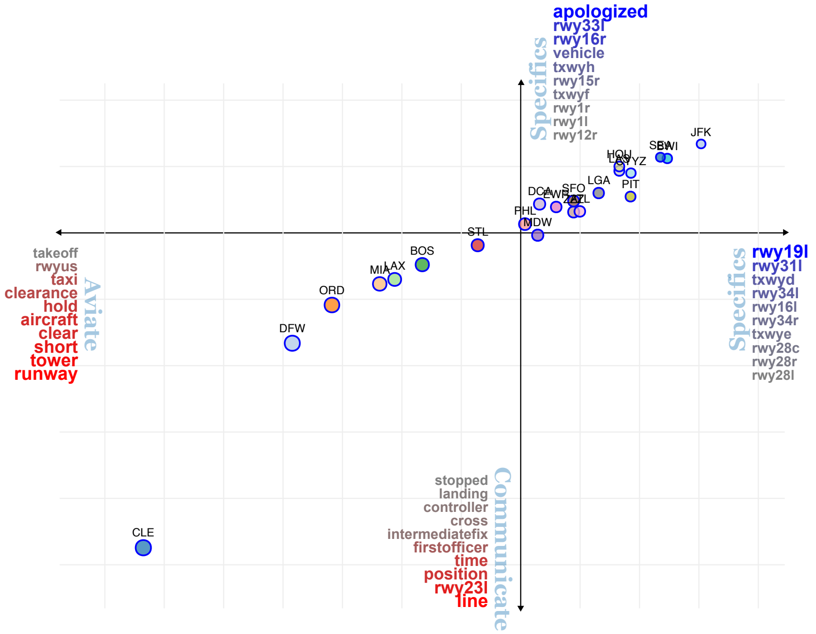


Figure 2: A sparse PCA plot of the runway ASRS data. Here, each data point is an airport, with size of the circles consistent with the number of reports for each airport.

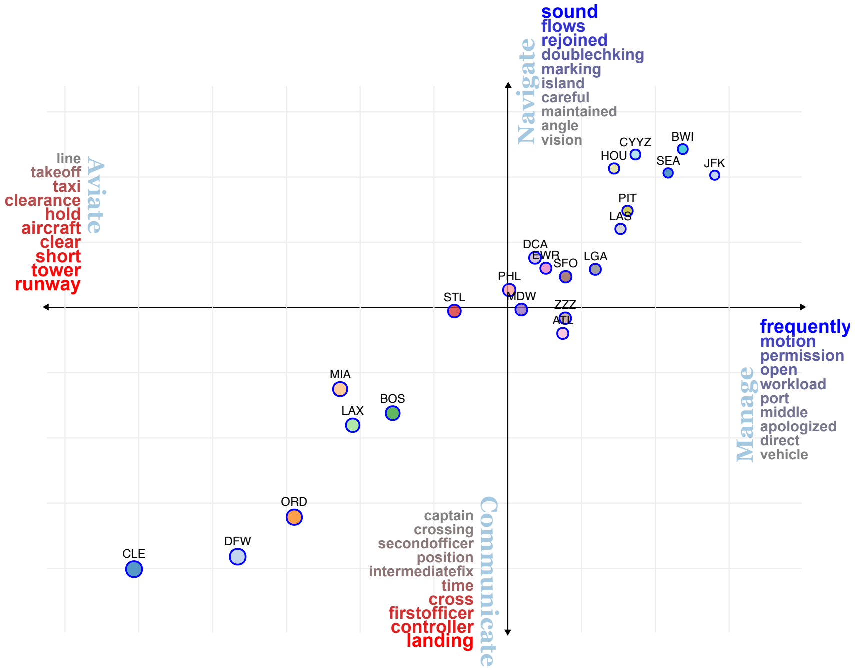


Figure 3: A sparse PCA plot of the runway ASRS data, with runway features removed.

Another cluster (on the NE quadrant) corresponds to the two remaining directions, which we labelled "specifics" as they related to specific runways and taxiways in airports. This other cluster of airports seem to be affected by issues related to specific runway configuration that are local to each airport.

In a second plot (Fig. 3) we redid the analysis after removal of all the features related to runways and taxiways, in order to discover what is "beyond" runway and taxiway issues. We recover the four themes of *Aviate*, *Navigate*, *Communicate* and *Manage*. As before, high-traffic airports remain affected mostly by aviate and communicate issues. Note that the disappearance of passenger-related issues within the *Manage* theme, which was defining the positive-vertical direction in Fig 1. This is to be expected, since the data is now restricted to runway issues: what involved passenger issues in the category data set, now becomes mainly related to the other humans in the loop, pilots ("permission"), drivers ("vehicle") and other actors, and their actions or challenges ("workload, open, apologized").

A look at the sparse PCA plots (Figs. 3 and 1) reveals a commonality: the themes of *Aviate* and *Communicate* seem to go together in the data, and are opposed to the other sub-group of *Navigate* and *Manage Systems*.

How about thresholded PCA? Fig. 4 shows the total explained variance by the two methods (sparse and thresholded PCA) as a function of the number of words allowed for the axes, for the category data set. We observe that thresholded PCA does not explain as much variance (in fact, only half as much) as sparse PCA, with the same budget of words allowed for each axis. This ranking is reversed only after 80 words are allowed in the budget. The two methods do reach the maximal variance explained by PCA as we relax our word-budget constraint. Similar observations can be made for the runway data set.

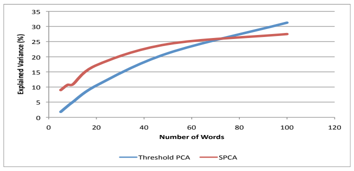


Figure 4: Explained variance.

4.5. **LASSO images of airports.** Our goal here is to use the runway data to help understand what specific runway-related issues affect each airport. To do this, we consider a specific airport and separate the runway incursion data in two sets: one set corresponds to the ASRS reports that contain the name of the airport under analysis; the other contains all the remaining ASRS documents in our corpus.

Using LASSO we can classify these two data sets, and discover the few features (terms in the dictionary) that are strong predictors of the differences. Hence we are able to single out a short list of terms that are strongly associated with the specific airport under consideration. Repeating this process for every airport provides a global, differentiated view of the runway incursion problem, as reported in the corpus analyzed. We have selected for illustration purposes the top twenty airports, as ordered by the number of reports that mention their name. The resulting short lists for a few of the airports are shown in Table 3. As expected, some airports' images point to the runways of that airport, and more importantly, to a few specific taxiways. The image of other airports, such as YYXZ (Toronto), points to other problems (lines, in the case of YYZ), and taxiways issues are less prevalent.

The LASSO analysis mostly points to specific runways for each airport. In order to go beyond this analysis, we focus on a single airport (say DFW). In the left panel of Fig 5, we propose a two-stage LASSO analysis allowing to discover a tree structure of terms. The inner circle corresponds to the LASSO image of DFW. Then, for each term in that image, we re-ran a LASSO analysis, comparing all the documents in the DFW-related corpus containing the term against all the other documents in the DFW-related corpus.

The tree analysis highlights which issues are pertaining to specific runways, and where *attention* could be focussed. In the airport diagram 6, we have highlighted some locations discussed next.

airport	term 1	term 2	term 3	term 4	term 5	term 6	term 7	term 8
CLE	Rwy23L	Rwy24L	Rwy24C	Rwy23R	Rwy5R	Line	Rwy6R	Rwy5L
DFW	Rwy35C	Rwy35L	Rwy18L	Rwy17R	Rwy18R	Rwy17C	cross	Tower
ORD	Rwy22R	Rwy27R	Rwy32R	Rwy27L	Rwy32L	Rwy22L	Rwy9L	Rwy4L
MIA	Rwy9L	TxwyQ	Rwy8R	Line	Rwy9R	PilotInCommand	TxwyM	Takeoff
BOS	Rwy4L	Rwy33L	Rwy22R	Rwy4R	Rwy22L	TxwyK	Frequency	Captain
LAX	Rwy25R	Rwy25L	Rwy24L	Rwy24R	Speed	cross	Line	Tower
STL	Rwy12L	Rwy12R	Rwy30L	Rwy30R	Line	cross	short	TxwyP
PHL	Rwy27R	Rwy9L	Rwy27L	TxwyE	amass	TxwyK	AirCarrier	TxwyY
MDW	Rwy31C	Rwy31R	Rwy22L	TxwyP	Rwy4R	midway	Rwy22R	TxwyY
DCA	TxwyJ	Airplane	turn	Captain	Line	Traffic	Landing	short
SFO	Rwy28L	Rwy28R	Rwy1L	Rwy1R	Rwy10R	Rwy10L	b747	Captain
ZZZ	hangar	radio	Rwy36R	gate	Aircraft	Line	Ground	Tower
ERW	Rwy22R	Rwy4L	Rwy22L	TxwyP	TxwyZ	Rwy4R	papa	TxwyPB
ATL	Rwy26L	Rwy26R	Rwy27R	Rwy9L	Rwy8R	atlanta	dixie	cross
LGA	TxwyB4	ILS	Line	notes	TxwyP	hold	vehicle	Taxiway
LAS	Rwy25R	Rwy7L	Rwy19L	Rwy1R	Rwy1L	Rwy25L	TxwyA7	Rwy19R
PIT	Rwy28C	Rwy10C	Rwy28L	TxwyN1	TxwyE	TxwyW	Rwy28R	TxwyV
HOU	Rwy12R	Rwy12L	citation	Takeoff	Heading	Rwy30L	Line	Tower
BWI	TxwyP	Rwy15R	Rwy33L	turn	TxwyP1	Intersection	TxwyE	Taxiway
CYYZ	TxwyQ	TxwyH	Rwy33R	Line	YYZ	Rwy24R	short	toronto
SEA	Rwy34R	Rwy16L	Rwy34L	Rwy16R	AirCarrier	FirstOfficer	TxwyJ	SMA
JFK	Rwy31L	Rwy13R	Rwy22R	Rwy13L	vehicle	Rwy4L	amass	Rwy31R

Table 3: The terms recovered with LASSO image analysis of a few airports in the “runway” ASRS data set.

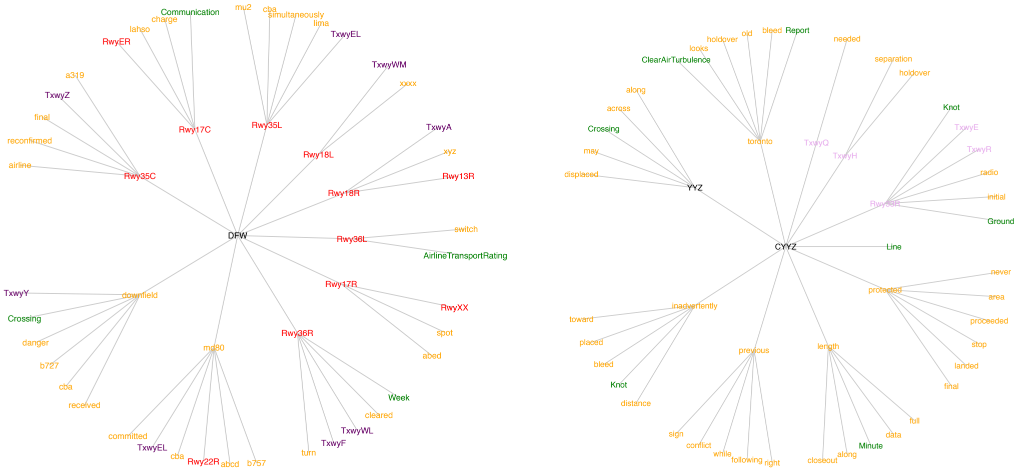


Figure 5: A tree LASSO analysis of the DFW (left panel) and CYYZ (right panel) airports, showing the LASSO image (inner circle) and for each term in that image, a further image.

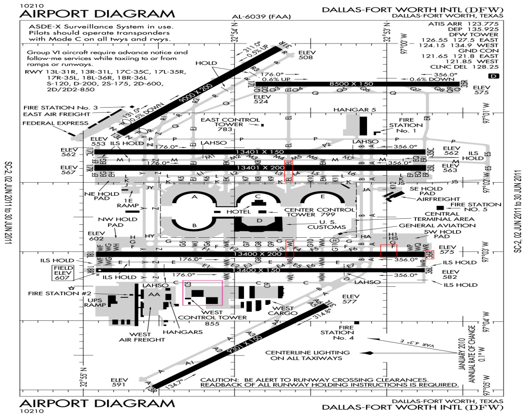


Figure 6: Diagram of DFW.

For example, as highlighted in red in the airport diagram 6, the major runway 35L crosses the taxiway EL, and the term in the tree image "simultaneously" evokes a risk of collision; similar comments can be made for the runway 36R and its siblings taxiway WL and F. At those particular intersections, the issues seem to be about obtaining "clearance" to "turn" from the tower, which might be due to the absence of line of sight from the tower (here we are guessing that the presence of the west cargo area could be a line-of-sight hindrance). The tree image is consistent with the location of DFW in the sparse PCA plot (Fig. 3), close to the themes of *Aviate* and *Communicate*.

Similar comments can be made about the tree image of the CYYZ airport, as shown in the right panel of Fig. 5. Note here that there is no mention of "ice" or other weather-related issues, which indicates that the measures taken to address them seem to work properly there.

5. Conclusions and future work

We have discussed several methods that explicitly encode sparsity in the model design. This encoding leads to a higher degree of interpretability of the model without penalizing, or even improving, the computational complexity of the algorithm. We demonstrated these techniques on real-world data from the Aviation Safety Reporting System and showed that they can reveal contributing factors to aviation safety incidents such as runway incursions. We also show that the sparse PCA and LASSO algorithms can discover the underlying task hierarchy that pilots perform.

Sparse learning problems are formulated as optimization problem with explicit encoding of sparsity requirements, either in the form of constraint or penalty. As such, the results have an explicit tradeoff between accuracy and sparsity based on the value of the sparsity-controlling parameter that is chosen. In comparison to thresholded PCA or similar methods, which provide "after-the-fact" sparsity, sparse learning methods offer a principled way to explicitly encode the tradeoff in the optimization problem. Thus, the enhanced interpretability of the results is a direct result of the optimization process.

In the safety monitoring of most critical, large-scale complex systems, from flight safety to nuclear plants, experts have relied heavily on physical sensors and indicators (temperature, pressure, etc). In the future we expect that human-generated text reporting, assisted by automated text understanding tools, will play an ever increasing role in the management of critical business, industry or government operations. Sparse modeling, by offering a great trade-off between user interpretability and computational scalability, appears to be well equipped to address some of the corresponding challenges.

6. Acknowledgments

A.N.S. thanks the NASA Aviation Safety Program, System Wide Safety and Assurance Technologies project for supporting this work, and Dr. Irving Statler, Linda Connell, and Charles Drew for their valuable insights regarding the Aviation Safety Reporting System and related discussions. L.E.G.'s work is partially supported by the National Science Foundation under Grants No. CMMI-0969923 and SES-0835531.

References

[1] M. A. U. Abedin, V. Ng, and L. Khan. Cause identification from aviation safety incident reports via weakly supervised semantic lexicon construction. *J. Artif. Int. Res.*, 38:569–631, May 2010.

[2] A. Agovic and A. Banerjee. Gaussian process topic models. In *Proceedings of the Twenty-Sixth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-10)*, pages 10–19, Corvallis, Oregon, 2010.

[3] M. S. Ahmed and L. Khan. SISC: A Text Classification Approach Using Semi Supervised Subspace Clustering. In *Proceedings of the 2009 IEEE International Conference on Data Mining Workshops*, ICDMW '09, pages 1–6, 2009.

[4] D. Blei and J. McAuliffe. Supervised topic models. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 121–128. MIT Press, Cambridge, MA, 2008.

[5] E. Candès and Y. Plan. Near-ideal model selection by ℓ_1 minimization. *Annals of Statistics*, 37:2145–2177, 2009.

[6] X. Dai, J. Jia, L. El Ghaoui, and B. Yu. SBA-term: Sparse bilingual association for terms. In *Fifth IEEE International Conference on Semantic Computing*, Palo Alto, CA, USA, 2011.

[7] D. Das and A. F. T. Martins. A survey on automatic text summarization, 2007.

[8] A. d'Aspremont, F. Bach, and L. El Ghaoui. Optimal solutions for sparse principal component analysis. *Journal of Machine Learning Research*, 9:1269–1294, 2008.

[9] B. Ding, D. Lo, J. Han, and S.-C. Khoo. Efficient Mining of Closed Repetitive Gapped Subsequences from a Sequence Database. In *Proceedings of the 2009 IEEE International Conference on Data Engineering*, pages 1024–1035, 2009.

[10] B. Ding, B. Zhao, C. X. Lin, J. Han, and C. Zhai. Topcells: Keyword-based search of top-k aggregated documents in text cube. *Data Engineering, International Conference on*, pages 381–384, 2010.

[11] A. Dobra, C. Hans, B. Jones, J. R. Nevins, G. Yao, and M. West. Sparse graphical models for exploring gene expression data. *Journal of Multivariate Analysis*, 90(1):196 – 212, 2004. Special Issue on Multivariate Methods in Genomic Data Analysis.

[12] D. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006.

[13] J. Eisenstein, A. Ahmed, and E. P. Xing. parse additive generative models of text. In *International Conference on Machine Learning (ICML)*, 2011.

[14] L. El Ghaoui, V. Viallon, and T. Rabbani. Safe feature elimination for the LASSO. *Journal of Machine Learning Research*, 2011. Submitted, April 2011.

[15] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15(12):3736–3745, 2006.

[16] J. Friedman, T. Hastie, H. Höfling, and R. Tibshirani. Pathwise coordinate optimization. *Annals of Applied Statistics*, 1(2):302–332, 2007.

[17] J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432, 2008.

[18] B. Gawalt, J. Jia, L. Miratrix, L. El Ghaoui, B. Yu, and S. Clavier. Discovering word associations in news media via feature selection and sparse classification. In *Proc. 11th ACM SIGMM International Conference on Multimedia Information Retrieval*, 2010.

[19] J. Goldstein, V. Mittal, J. Carbonell, and M. Kantrowitz. Multi-document summarization by sentence extraction. In *NAACL-ANLP 2000 Workshop on Automatic summarization*, pages 40–48, 2000.

[20] L. Hennig. Topic-based multi-document summarization with probabilistic latent semantic analysis. In *Recent Advances in Natural Language Processing (RANLP)*, 2009.

[21] T. Joachims. *Learning to Classify Text Using Support Vector Machines – Methods, Theory, and Algorithms*. Kluwer/Springer, 2002.

[22] J. E. Jonsson and W. R. Ricks. Cognitive models of pilot categorization and prioritization of flight-deck information. Technical report, 1995.

[23] M. Journée, Y. Nesterov, P. Richtárik, and R. Sepulchre. Generalized power method for sparse principal component analysis. *arXiv:0811.4724*, 2008.

[24] M. Kolar, A. Parikh, and E. Xing. On Sparse Nonparametric Conditional Covariance Selection. *International Conference on Machine Learning*, 2010.

[25] C. X. Lin, B. Ding, J. Han, F. Zhu, and B. Zhao. Text Cube: Computing IR Measures for Multidimensional Text Database Analysis. *IEEE International Conference on Data Mining*, pages 905–910, 2008.

[26] Z. Lu, R. Monteiro, and M. Yuan. Convex optimization methods for dimension reduction and coefficient estimation in multivariate linear regression. *Mathematical Programming*, 9(1):1–32, 2010.

[27] L. Mackey. Deflation methods for sparse PCA. *Advances in Neural Information Processing Systems*, 21:1017–1024, 2009.

[28] J. Mairal, G. Sapiro, and M. Elad. Learning multiscale sparse representations for image and video restoration. *SIAM Multiscale Modeling and Simulation*, 7(1):214–241, 2008.

[29] N. Meinshausen and B. Yu. Lasso-type recovery of sparse representations for high-dimensional data. *Annals of Statistics*, 37(1):246–270, 2008.

[30] L. Miratrix, J. Jia, B. Gawalt, B. Yu, and L. El Ghaoui. Summarizing large-scale, multiple-document news data: sparse methods and human validation. submitted to JASA.

[31] B. Moghaddam, Y. Weiss, and S. Avidan. Fast Pixel/Part Selection with Sparse Eigenvectors. *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8, 2007.

[32] B. L. Monroe, M. P. Colaresi, and K. M. Quinn. Fightin' words: Lexical feature selection and evaluation for identifying the content of political conflict. *Political Analysis*, 16(4):372–403, 2008.

[33] Y. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *CORE Discussion Papers*, 2010.

[34] O.Banerjee, L. El Ghaoui, and A. d'Aspremont. Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *Journal of Machine Learning Research*, 9:485–516, March 2008.

[35] N. C. Oza, J. P. Castle, and J. Stutz. Classification of Aeronautics System Health and Safety Documents. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, 39(6):670–680, 2009.

[36] I. Persing and V. Ng. Semi-supervised cause identification from aviation safety reports. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, pages 843–851, 2009.

[37] F. Schilder and R. Kondadadi. Fastsum: fast and accurate query-based multi-document summarization. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, HLT-Short '08, pages 205–208, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics.

[38] H. Shan, A. Banerjee, and N. C. Oza. Discriminative Mixed-Membership Models. In *Proceedings of the 2009 Ninth IEEE International Conference on Data Mining*, pages 466–475, Washington, DC, USA, 2009.

[39] H. Shen and J. Z. Huang. Sparse principal component analysis via regularized low rank matrix approximation. *J. Multivar. Anal.*, 99:1015–1034, July 2008.

[40] J. Songsiri and L. Vandenberghe. Topology selection in graphical models of autoregressive processes. *Journal of Machine Learning Research*, 2010.

[41] R. Tibshirani. Regression shrinkage and selection via the LASSO. *Journal of the Royal statistical society, series B*, 58(1):267–288, 1996.

[42] J. A. Tropp. Just relax: Convex programming methods for identifying sparse signals. *IEEE Trans. Inform. Theory*, 51(3):1030–1051, Mar. 2006.

[43] C. Woolam and L. Khan. Multi-concept Document Classification Using a Perceptron-Like Algorithm. In *Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology - Volume 01*, pages 570–574, 2008.

[44] C. Woolam and L. Khan. Multi-label large margin hierarchical perceptron. *IJDMMM*, 1(1):5–22, 2008.

[45] M. Yuan and Y. Lin. Model selection and estimation in the Gaussian graphical model. *Biometrika*, 94(1):19, 2007.

[46] D. Zhang, C. Zhai, J. Han, A. Srivastava, and N. Oza. Topic modeling for OLAP on multidimensional text databases: topic cube and its applications. *Stat. Anal. Data Min.*, 2:378–395, December 2009.

[47] Y. Zhang, A. d'Aspremont, and L. El Ghaoui. Sparse PCA: Convex relaxations, algorithms and applications. In M. Anjos and J. Lasserre, editors, *Handbook on Semidefinite, Cone and Polynomial Optimization: Theory, Algorithms, Software and Applications.* Springer, 2011. To appear.

[48] H. Zou, T. Hastie, and R. Tibshirani. Sparse Principal Component Analysis. *Journal of Computational & Graphical Statistics*, 15(2):265–286, 2006.

TOWARDS AN AUTOMATED CLASSIFICATION OF TRANSIENT EVENTS IN SYNOPTIC SKY SURVEYS

S. G. DJORGOVSKI[1,5], C. DONALEK[1], A.A.MAHABAL[1], B. MOGHADDAM[2], M. TURMON[2],
M.J. GRAHAM[3], A.J. DRAKE[3], N. SHARMA[4], Y. CHEN[4]

ABSTRACT. We describe the development of a system for an automated, iterative, real-time classification of transient events discovered in synoptic sky surveys. The system under development incorporates a number of Machine Learning techniques, mostly using Bayesian approaches, due to the sparse nature, heterogeneity, and variable incompleteness of the available data. The classifications are improved iteratively as the new measurements are obtained. One novel feature is the development of an automated follow-up recommendation engine, that suggest those measurements that would be the most advantageous in terms of resolving classification ambiguities and/or characterization of the astrophysically most interesting objects, given a set of available follow-up assets and their cost functions. This illustrates the symbiotic relationship of astronomy and applied computer science through the emerging discipline of AstroInformatics.

1. INTRODUCTION

A new generation of scientific measurement systems (instruments or sensor networks) is now generating exponentially growing *data streams*, now moving into the Petascale regime, that can enable significant new discoveries. Often, these consist of phenomena where a rapid change occurs, that have to be identified, characterized, and possibly followed by new measurements in the real time. The requirement to perform the analysis rapidly and objectively, coupled with huge data rates, implies *a need for automated classification and decision making*.

This entails some special challenges beyond traditional automated classification approaches, which are usually done in some feature vector space, with an abundance of self-contained data derived from homogeneous measurements. Here, the input information is generally sparse and heterogeneous: there are only a few initial measurements, and the types differ from case to case, and the values have differing variances; the contextual information is often essential, and yet difficult to capture and incorporate in the classification process; many sources of noise, instrumental glitches, etc., can masquerade as transient events in the data stream; new, heterogeneous data arrive, and the classification must be iterated dynamically. Requiring a high completeness (don't miss any interesting events) and low contamination (a few false alarms), and the need to complete the classification process and make an optimal decision about expending

[1] California Institute of Technology, [george,donalek,aam]@astro.caltech.edu
[2] Jet Propulsion Laborattory, [baback,turmon]@jpl.nasa.gov
[3] California Institute of Technology, [mjg,ajd]@cacr.caltech.edu
[4] California Institute of Technology, [nihar,cheny]@caltech.edu
[5] Distinguished Visiting Professor, King Abdulaziz University, Jeddah, Saudi Arabia

classification of the transient sources is the key to their interpretation and scientific uses, and in many cases scientific returns come from the follow-up observations that depend on scarce or costly resources (e.g., observing time at larger telescopes). Since the transients change rapidly, a rapid (as close to the real time as possible) classification, prioritization, and follow-up are essential, the time scale depending on the nature of the source, which is initially unknown. In some cases the initial classification may remove the rapid-response requirement, but even an archival (i.e., not time-critical) classification of transients poses some interesting challenges.

A number of synoptic astronomical surveys are already operating [see, e.g., 1,2,3,7,17,25,26,43], and much more ambitious enterprises [4,5] will move us into the Petascale regime, with hundreds of thousands of transient events per night, implying a need for an automated, robust processing and follow-up, sometimes using robotic telescopes. Thus, *a new generation of scientific measurement systems is emerging* in astronomy, and many other fields: connected sensor networks which gather and analyze data automatically, and respond to outcome of these measurements in the real-time, often redirecting the measurement process itself, and without human intervention.

We are developing a novel set of techniques and methodology for an automated, real-time data analysis and discovery, operating on massive and heterogeneous data streams from robotic telescope sensor networks, fully integrated with Virtual Observatory (VO) [39,40,42]. The system incorporates machine learning elements for an iterative, dynamical classification of astronomical transient events, based on the initial detection measurements, archival information, and newly obtained follow-up measurements from robotic telescopes. A key novel feature, still under development, will be the ability to define and request particular types of follow-up observations in an automated fashion. Our goal is to increase the efficiency and productivity of a number of synoptic sky survey data streams, and enable new astrophysical discoveries.

2. THE CHALLENGE OF AN AUTOMATED, REAL-TIME EVENT CLASSIFICATION

A full scientific exploitation and understanding of astrophysical events requires a rapid, multi-wavelength follow-up. The *essential enabling technologies* that need to be automated are robust classification and decision making for the optimal use of follow-up facilities. They are the key for exploiting the full scientific potential of the ongoing and forthcoming synoptic sky surveys.

The first challenge is to associate classification probabilities that any given event belongs to a variety of known classes of variable astrophysical objects and to update such classifications as more data come in, until a scientifically justified convergence is reached [24]. Perhaps an even more interesting possibility is that a given transient represents a previously unknown class of objects or phenomena, that may register as having a low probability of belonging to any of the known data models. The process has to be *as automated as possible, robust, and reliable*; it has to operate from *sparse and heterogeneous data*; it has to maintain a *high completeness* (not miss any interesting events) yet a *low false alarm rate*; and it has to *learn* from the past experience for an ever improving, evolving performance. The next step is development and implementation of an automated follow-up event prioritization and decision making mechanism, which would actively determine and request follow-up observations on demand, driven by the event data analysis. This would include an automated identification of the most discriminating potential measurements from the available follow-up assets, taking into account their relative cost functions, in order to optimize both classification discrimination, and the potential scientific returns.

An illustration of an existing, working system for a real-time classification of astrophysical event candidates in a real synoptic sky survey context is shown in Fig. 2. This is an Artificial Neural Network (ANN) based classifier [18] that separates real transient sources from a variety of spurious candidates caused by various data artifacts (electronic glitches, saturation, cross-talk, reflections, etc.), that operated as a part of the Palomar-Quest (PQ) survey's [7,26] real time data reduction pipeline. While this is a very specialized instance of an automated event classifier for a particular sky survey experiment, it illustrates the plausibility and the potential of this concept. A similar approach, using Support Vector Machine (SVM) techniques [11], has been deployed successfully by the Lawrence Berkeley National Laboratory Nearby Supernova Factory [10,27]. Use of image morphology for astronomical image classification via machine learning has long been used successfully, e.g., [12,19,20]. Here we deploy it in a real-time data reduction pipeline.

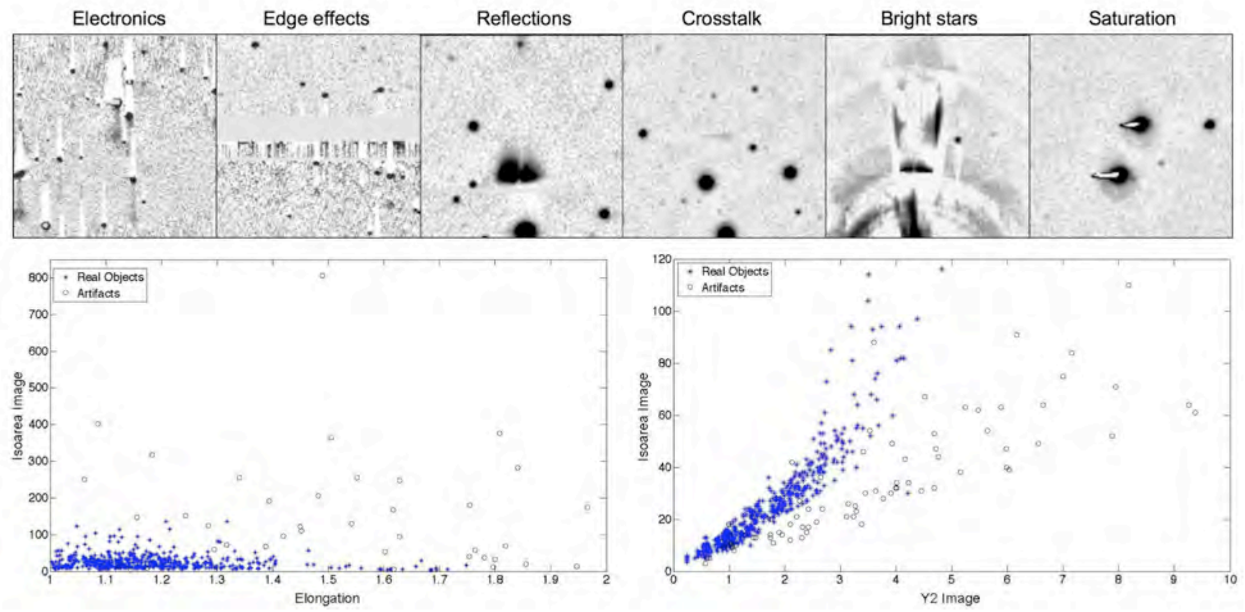


Figure 2. Automated classification of candidate events, separating real astronomical sources from a variety of spurious candidates (instrument artifacts) is operational within the Palomar-Quest) survey's real time data pipeline [26,31]. Image cutouts on the top show a variety of instrumental and data artifacts which appear as spurious transients, since they are not present in the baseline comparison images. The two panels on the bottom show a couple of morphological parameter space projections, in which artifacts (○) separate well from genuine objects (∗). A multi-layer perceptron (MLP) ANN is trained to separate them, using 4 image parameters, with an average accuracy of ~ 95%. From Donalek et al., [31] and in prep.

However, the problem here is more complex and challenging: it is an astrophysical classification of genuine transient events, all of which would look the same in the images (star-like), so that information other than image morphology must be used. One problem is that in general, not all parameters would be measured for all events, e.g., some may be missing a measurement in a particular filter, due to a detector problem; some may be in the area on the sky where there are no useful radio observations; etc. Broader approaches to automated classification of transients and variables include, e.g., [28,30,31,44,45,46,47,48].

A more insidious problem is that many observables would be given as upper or lower limits, rather than as well defined measurements; for example, "the increase in brightness is > 3.6 magnitudes", or "the radio to optical flux ratio of this source is < 0.01". One approach is to treat them as missing data, implying a loss of the potentially useful information. A better approach is to reason about "censored" observations, that can be naturally incorporated through a Bayesian model by choosing a likelihood function that rules out values violating the bounds.

3. A BAYESIAN APPROACH TO EVENT CLASSIFICATION

We identify two core problems: *classification* (physical interpretation of an event), learning from compiled knowledge obtained by linking observations to phenomena, and *recommendation* (what are the optimal follow-up observations for this particular event).

The main astronomical inputs are in the form of observational and archival parameters for individual objects, which can be put into various, often independent subsets. Examples include fluxes measured at different wavelengths, associated colors or hardness ratios, proximity values, shape measurements, magnitude characterizations at different timescales, etc. The heterogeneity and sparsity of data makes the use of Bayesian methods for classification a natural choice.

Distributions of such parameters need to be estimated for each type of variable astrophysical phenomena that we want to classify. Then an estimated probability of a new event belonging to any given class can be evaluated from all of such pieces of information available, as follows. Let us denote the feature vector of event parameters as x, and the object class that gave rise to this vector as y, $1 \le y \le K$. While certain fields within x will generally be known, such as sky position and brightness in selected filters, many other parameters will be known only sporadically, e.g., brightness change over various time baselines. In a Bayesian approach, x and y are related via

$$P(y=k \mid x) = P(x \mid y=k)P(k)/P(x) \propto P(k)P(x \mid y=k) \approx P(k)\prod_{b=1}^{B} P(x_b \mid y=k)$$

Because we are only interested in the above quantity as a function of k, we can drop factors that only depend on x. We assume that, conditional on the class y, the feature vector *decomposes* into B roughly independent blocks, generically labeled x_b. These blocks may be singleton variables, or contain multiple variables, e.g., sets of filters that are highly correlated. The resulting algorithm is called *naive Bayes* because of its assumption that we may decouple the inputs in this way [8,9].

This decoupling is advantageous to us in two ways. First, it allows us to circumvent the "curse of dimensionality," because we will eventually have to learn the conditional distributions $P(x_b \mid y = k)$ for each k. As more components are added to x_b, more examples will be needed to learn the corresponding distribution. The decomposition keeps the dimensionality of each feature block manageable. Second, such decomposition allows us to cope easily with ignorance of missing variables. We simply drop the corresponding factors from the product above.

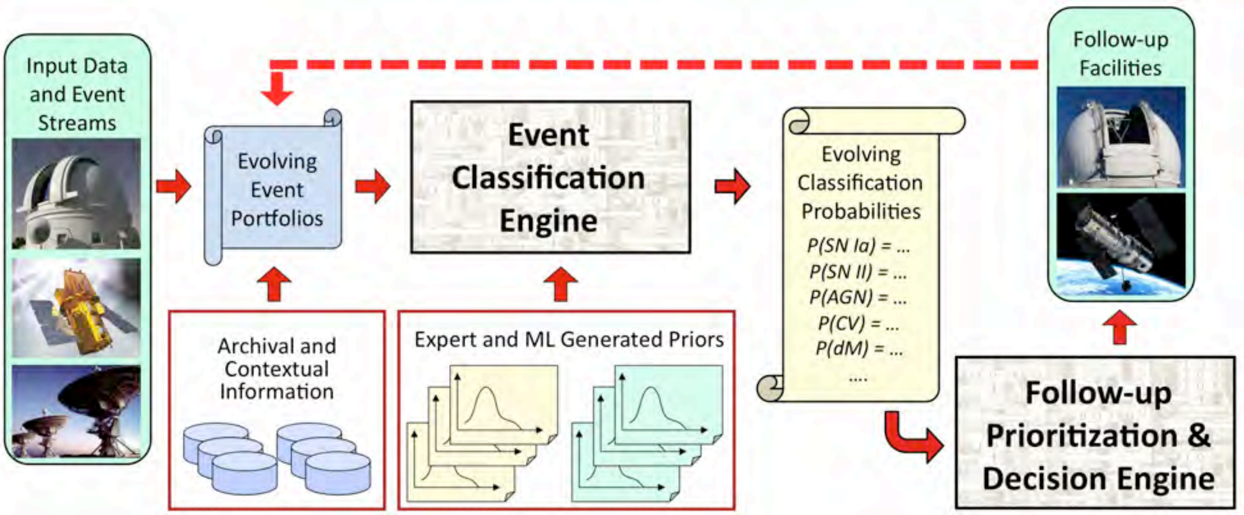

Input Data and Event Streams
Evolving Event Portfolios
Event Classification Engine
Evolving Classification Probabilities
P(SN Ia) = ...
P(SN II) = ...
P(AGN) = ...
P(CV) = ...
P(dM) = ...
....
Follow-up Facilities
Archival and Contextual Information
Expert and ML Generated Priors
Follow-up Prioritization & Decision Engine

Figure 3. A conceptual outline of the system. The initial input consists of the generally sparse data describing transient events discovered in sky surveys, supplemented by archival heterogeneous measurements from external, multi-wavelength archives corresponding to this spatial location, if available (e.g. radio flux and distance to nearest galaxy). Data are collected in evolving electronic portfolios containing all currently available information for a given event. These data are fed into the Event Classification Engine; another input into the classification process is an evolving library of priors giving probabilities for observing these particular parameters if the event was belonging to a class X. The output of the classification engine is an evolving set of probabilities of the given event belonging to various classes of interest, which are updated as more data come in, and classifications change. This forms an input into the Follow-up Prioritization and Decision Engine, which would prioritize the most valuable follow-up measurements given a set of available follow-up assets (e.g., time on large telescopes, etc.), and their relative cost functions. What is being optimized is: (a) the new measurements which would have a maximum discrimination for ambiguous classifications, and/or (b) the follow-up measurements which would likely yield most interesting science, given the current best-guess event classification? New measurements from such follow-up observations will be fed back into the event portfolios, leading to dynamically updated/iterated classifications, repeating the cycle.

As a simple demonstration of the technique, we have been experimenting with a prototype Bayesian Network (BN) model [32,33]. We use a small but homogeneous data set involving colors of ~ 1,000 reliably classified transients detected in the CRTS survey [17,25], as measured at the Palomar 1.5-m telescope. We have used multinomial nodes (discrete bins) for 3 colors, with provision for missing values, and a multinomial node for Galactic latitude which is always present and is a probabilistic indicator of whether an object is Galactic or not. The current priors used are for six distinct classes, cataclysmic variables (CVs; these are binary star system in which a compact stellar remnant such as a white dwarf or a neutron star accretes material from its companion in a fairly stochastic fashion), supernovae (SN; these are exploding stars, and while there are several distinct types, the overall behavior is very similar), blazars (beamed active galactic nuclei, or AGN, where we are looking into their relativistic jet), other variable AGN, UV Ceti stars (dwarf stars undergoing gigantic equivalent of the Solar flares), and all else bundled into a sixth pseudo-class, called Rest. Testing is done with a 10-fold cross validation, in order to assess how good it will perform on an independent data set.

Using a sample of 316 SNe, 277 CVs, and 104 blazars, and a *single* epoch measurement of colors, in the relative classification of CVs vs. SNe, we obtain a completeness of ~ 80% and a contamination of ~ 19%, which reflects a qualitative color difference between these two types of transients. In the relative classification of CVs vs. blazars, we obtain a completeness of ~ 70 – 90% and a contamination of ~ 10 – 24% (the ranges corresponding to different BN experiments), which reflects the fact that colors of these two types of transients tend to be similar, and that some additional discriminative parameter is needed. Eventually we will use a BN with an order of magnitude more classes, including divisions of different types of SNe, AGN, and a large variety of variable star types (there are literally hundreds of varieties of variable stars, but only a few tens may be relevant for the present transients search), with more measured parameters, and additional BN layers. Measurements from multiple epochs should improve considerably the classifications. The end result will be the posteriors for the "Class" node from the marginalized probabilities of all available inputs for a given object.

In this framework the priors come from a set of observed parameters like distribution of colors, distribution of objects as a function of Galactic latitude, frequencies of different types of objects etc. The posteriors we are interested in are determining the type of an object based on, say, its (*r-i)* color, Galactic latitude and proximity to another object etc.

Sparse and/or irregular light curves (LC) from any given object class can have sufficient salient structure that can be exploited by automated classification algorithms. We have experimented with Gaussian Process Regression [34], and found it to be useful for parameter estimation for a certain types of LCs that can be represented by a standard data model (e.g., Supernovae).

We are now experimenting with a different approach. By *pooling* many instances of an object class's LCs we can effectively represent and encode their characteristic structure *probabilistically*, and construct an empirical probability distribution function (PDF) that can be used for subsequent classification of new event observations. This comparison can be made incrementally over time as new observations "trickle in", with the final classification scores growing more confident with each additional set of observations that is accumulated.

Since the telescope's (flux-only) observations come primarily in the form of single magnitude changes over time increments – *e.g.*, an observed (Δt, Δm) pair – we focus on modeling the joint distribution of all such pairs of data points for a given LC (Note: we consider all possible *causal* increments available, corresponding to Δt > 0). By virtue of being *increments*, these data and their empirical PDF will be invariant to absolute magnitude (the distance to the event generally being unknown) and time (the onset of the event not being known) shifts. Additionally, these densities allow flux upper limits to be encoded as well – *e.g.*, under poor seeing conditions, we may only obtain bounded observations such as m > 18. We currently use smoothed 2D histograms to model the distribution of (Δt, Δm) pairs. This is a computationally simple, yet effective way to implement a non-parametric density model that is flexible enough for all object classes under our consideration. Figure 4 shows the joint 2D histograms for 3 classes of objects and how a given probe LC measurements fit these 3 class-specific histograms.

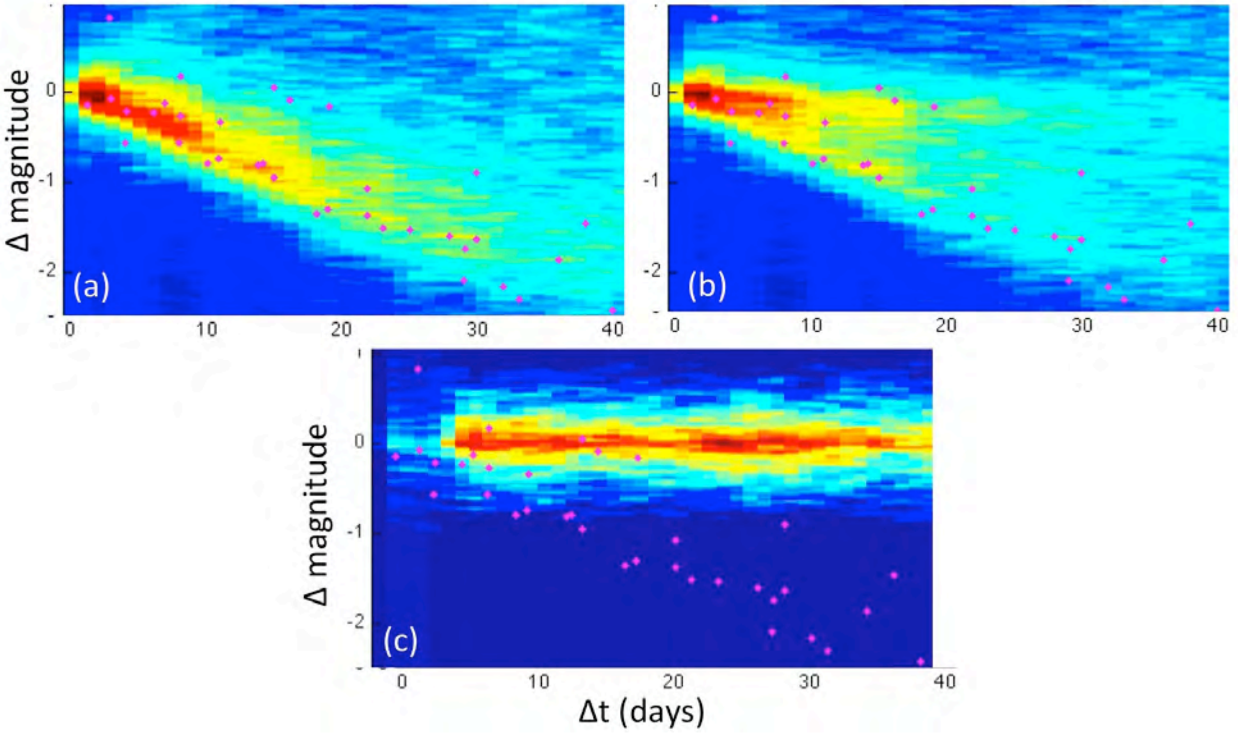


Figure 4. Examples of (Δm, Δt) pairs PDFs for three types of astrophysical transients: (a) SN Ia, (b) SN IIP and (c) RR Lyrae, using bins of width Δt = 1 day, and Δm = 0.01. The histograms were smoothed with a 3-tap triangular Δt kernel = [0.25 0.5 0.25] and a Gaussian Δm kernel of FWHM = 0.05 mag. The set of diamonds superimposed on each panel are from a single test case of a SN Ia's LC. Note that PDFs for the two SN types form a better "fit" to the observed data (diamonds) than the RR Lyrae's PDF (and SN Ia is a better fit than SN II P). Various metrics on probability distributions can be used to automatically quantify the degree of fitness.

In our preliminary experimental evaluations with a small number of object classes (single outburst like SN, periodic variable stars like RR Lyrae and Miras, as well as stochastic like blazars and CVs) we have been able to show that our gap event density models are potentially a powerful classification method from sparse/irregular time series like typical observational LC data.

4. INCORPORATING THE CONTEXTUAL INFORMATION

Contextual information can be highly relevant to resolving competing interpretations: for example, the light curve and observed properties of a transient might be consistent with both it being a cataclysmic variable star, a blazar, or a supernova. If it is subsequently known that there is a galaxy in close proximity, the supernova interpretation becomes much more plausible. Such information, however, can be characterized by high uncertainty and absence, and by a rich structure – if there were two candidate host galaxies, their morphologies, distance, etc., become important, e.g., is this type of supernova more consistent with being in the extended halo of a large spiral galaxy or in close proximity to a faint dwarf galaxy? The ability to incorporate such contextual information in a quantifiable fashion is highly desirable. In a separate project we are investigating the use of crowdsourcing as a means of harvesting the human pattern recognition skills, especially in the context of capturing the relevant contextual information, and turning them into machine-processible algorithms.

A methodology employing contextual knowledge forms a natural extension to the logistic regression and classification methods mentioned above. Ideally such knowledge can be expressed in a manipulable fashion within a sound logical model, for example, it should be possible to state the rule that "a supernova has a stellar progenitor and will be substantially brighter than it by several order of magnitude" with some metric of certainty and infer the probabilities of observed data matching it. *Markov Logic Networks* (MLNs, [36]) are such a probabilistic framework using declarative statements (in the form of logical formulae) as atoms associated with real-valued weights expressing their strength. The higher the weight, the greater the difference in log probability between a world that satisfies the formula and one that does not, all other thing being equal. In this way, it becomes possible to specify 'soft' rules that are likely to hold in the domain, but subject to exceptions - contextual relationships that are likely to hold such as supernovae may be associated with a nearby galaxy or objects closer to the Galactic plane may be stars.

A MLN defines a probability distribution over possible worlds with weights that can be learned generatively or discriminatively: it is a model for the conditional distribution of the set of query atoms Y given the set of evidence atoms X. Inferencing consists of finding the most probable state of the world given some evidence or computing the probability that a formula holds given a MLN and set of constants, and possibly other formulae as evidence. Thus the likelihood of a transient being a supernova, depending on whether there was a nearby galaxy, can be determined.

The structure of a MLN – the set of formulae with their respective weights – is also not static but can be revised or extended with new formulae either learned from data or provided by third parties. In this way, new information can easily be incorporated. Continuous quantities, which form much of astronomical measurements, can also be easily handled with a hybrid MLN [37].

5. COMBINING AND UPDATING THE CLASSIFIERS

An essential task is to derive an optimal event classification, given inputs from a diverse set of classifiers such as those described above. This will be accomplished by a fusion module, currently under development, illustrated schematically in Fig. 5.

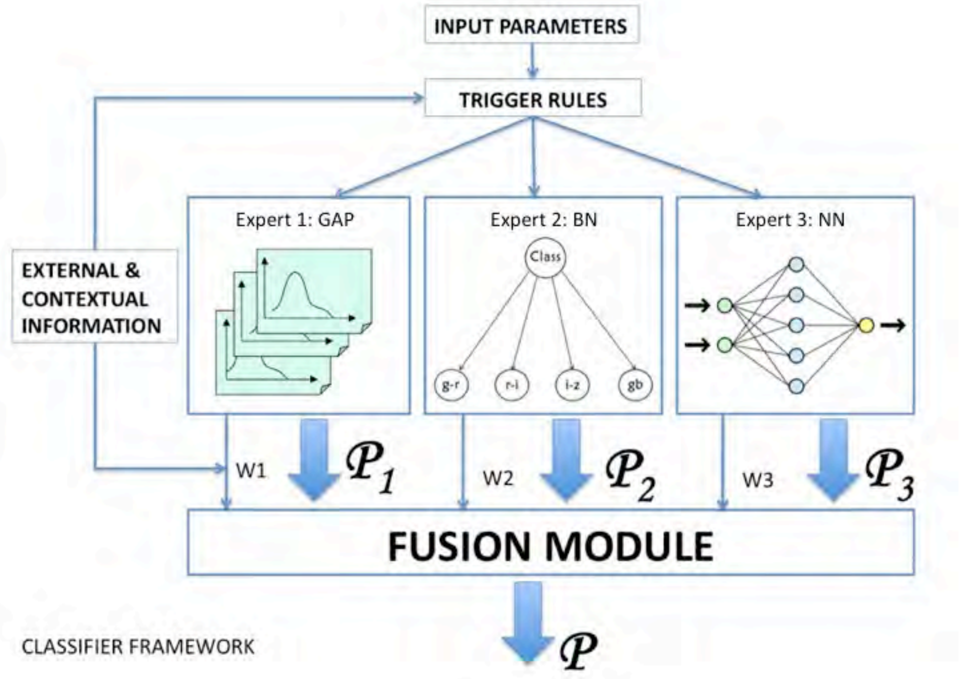


Figure 5. A schematic illustration of the event classifier combination challenge, to be implemented by the classification fusion module. Different aspects of available event information trigger different classifiers. In some cases more than one classifier can be used. How to combine the different outcomes is a subject of the ongoing work.

A MLN approach could be used to represent a set of different classifiers and the inferred most probable state of the world from the MLN would then give the optimal classification. For example, a MLN could fuse the beliefs of different ML-based transient classifiers – 4 give a supernova and 3 give a cataclysmic variable, say – to give a definitive answer.

We are experimenting with the so-called "sleeping expert" [35] method. A set of different classifiers each generally works best with certain kinds of inputs. Activating these optionally only when those inputs are present provides an optimal solution to the fusion of these classifiers. Sleeping expert can be seen as a generalization of the IF-THEN rule: IF this condition is satisfied THEN activate this expert, e.g., a specialist that makes a prediction only when the instance to be predicted falls within their area of expertise. For example, some classifiers work better when certain inputs are present, and some work only when certain inputs are present. It has been shown that this is a powerful way to decompose a complex classification problem. External or *a priori* knowledge can be used to awake or put experts to sleep and to modify online the weights associated to a given classifier; this contextual information may be also expressed in text.

A crucial feature of the system should be the ability to update and revise the prior distributions on the basis of the actual performance, as we accumulate the true physical classifications of events, e.g., on the basis of follow-up spectroscopy. Learning, in the Bayesian view, is precisely the action of determining the probability models above – once determined, the overall model (1) can be used to answer many relevant questions about the events. Analytically, we formulate this as determining unknown distributional parameters θ in parameterized versions of the conditional distributions above, $P(x \mid y = k; \theta)$. (Of course, the parameters depend on the object class *k*, but we suppress this below.) In a histogram representation, θ is just the probabilities associated with each bin, which may be determined by computing the histogram itself. In a Gaussian representation, θ would be the mean vector μ and covariance matrix Σ of a multivariate Gaussian distribution, and the parameter estimates are just the corresponding mean and covariance of the object-*k* data. When enough data is available we can adopt a semi-parametric representation in which the distribution is a linear superposition of such Gaussian distributions,

$$P(x_d \mid y = k; \theta) = \sum_{m=1}^{M} \lambda_m N(x_d; \mu_m, \Sigma_m)$$

This generalizes the Gaussian representation, since by increasing *M*, more distributional characteristics may be accounted for. The corresponding parameters may be chosen by the Expectation-Maximization algorithm [13]. Alternatively, kernel density estimation could be used, with density values compiled into a lookup table [14,21].

We can identify three possible sources of information that can be used to find the unknown parameters. They can be from the *a priori* knowledge, e.g. from physics or monotonicity considerations, or from examples that are labeled by experts, or from the feedback from the downstream observatories once labels are determined. The first case would serve to give an analytical form for the distribution, but the second two amount to the provision of labeled examples, (*x*, *y*), which can be used to select a set of *k* probability distributions.

6. AUTOMATED DECISION MAKING FOR AN OPTIMIZED FOLLOW-UP

We typically have sparse observations of a given object of interest, leading to classification ambiguities among several possible object types (e.g., when an event is roughly equally likely to belong to two or more possible object classes, or when the initial data are simply inadequate to generate a meaningful classification at all). Generally speaking, some of them would be of a greater scientific interest than others, and thus their follow-up observations would have a higher scientific return. Observational resources are scarce, and always have some cost function associated with them, so a key challenge is to determine the follow-up observations that are most useful for improving classification accuracy, and detect objects of scientific interest.

There are two parts to this challenge. First, what type of a follow-up measurement – given the *available* set of resources (e.g., only some telescopes/instruments may be available) – would yield the maximum information gain in a particular situation? And second, if the resources are finite and have a cost function associated with them (e.g., you can use only so many hours of the telescope time), when is the potential for an interesting discovery worth spending the resources?

We take an information-theoretic approach to this problem [15] that uses Shannon entropy to measure ambiguity in the current classification. We can compute the entropy drop offered by the available follow-up measurements – for example, the system may decide that obtaining an optical light curve with a particular temporal cadence would discriminate between a Supernova and a flaring blazar, or that a particular color measurement would discriminate between, say, a cataclysmic variable eruption and a gravitational microlensing event. A suitable prioritized request for the best follow-up observations would be sent to the appropriate robotic (or even human-operated) telescopes.

Note that the system is suggesting follow-up observations that may involve imperfect observations of a block of individual variables. This is a more powerful capability than rank-ordering individual variables regarding their helpfulness. Furthermore, we will ascertain that the framework accounts for the varying degrees of accuracy of different observations. The key to quantifying the classification uncertainty is the conditional entropy of the posterior distribution for y, given all the available data. Let $H[p]$ denote the Shannon entropy of the distribution p, which is always a distribution over object-class y. (The classification is discrete, so we only need to compute entropies of discrete distributions.) Then, when we take an additional observation x_+, uncertainty drops from $H[p(y \mid x_0)]$ to $H[p(y \mid x_0, x_+)]$. We want to choose the source x_+ so that the expected final entropy is lowest. To choose the best refinement in advance, we look for the largest expected drop in entropy.

Because all observing scenarios start out at the same entropy $H[p(y \mid x_0)]$, maximizing entropy drop is the same as minimizing expected final entropy, $E[H[p(y \mid x_0, x_+)]]$. The expectation is with respect to the distribution of the new variable x_+, whose value is not yet known. Therefore, this entropy is a function of the *distribution* of x_+, but not the value of the random variable x_+. The distribution captures any imprecision and noise in the new observation. In our notation, the best follow-on observation thus minimizes, over available variables x_+,

$$H[\,p(y \mid x_+, x_0)] = -\sum_{y, x_+} p(y, x_+ \mid x_0) \log p(y \mid x_+, x_0)\,.$$

This is equivalent to maximizing the conditional mutual information of x_+ about y, given x_0; that is, $I(y; x_+ \mid x_0)$ [22]. *The density above is known within the context of our assumed statistical model.* Thus, we can compute, within the context of the previously learned statistical model, a rank-ordered list of follow-on observations, which will lead to the most efficient use of resources.

Alternatively, instead of maximizing the classification accuracy, we consider a scenario where the algorithm chooses a set of events for follow-up and subsequent display to an astronomer. The astronomer then provides information on how interesting the observation is. The goal of the algorithm is to learn to choose follow-up observations which are considered most interesting. This problem can be naturally modeled using *Multi-Armed Bandit* algorithms (MABs) [38]. The MAB problem can abstractly be described as a slot machine with *k* levers, each of which has different expected returns (unknown to the decision maker). The aim is to determine the best strategy to maximize returns. There are two extreme approaches: (1) exploitation - keep pulling the lever which, as per your current knowledge, returns most, and (2) exploration – experiment with different levers in order to gather information about the expected returns associated with each lever. They key challenge is to trade off exploration and exploitation. There are algorithms [47] guaranteed to determine the best choice as the number of available tries goes to infinity.

In this analogy different telescopes and instruments are the levers that can be pulled. Their ability to discriminate between object classes forms the returns. This works best when the priors are well assembled and a lot is already known about the type of object one is dealing with. But due to the heterogeneity of objects, and increasing depth leading to transients being detected at fainter levels, and more examples of relatively rarer subclasses coming to light, treating the follow-up telescopes as a MAB will provide a useful way to rapidly improve the classification and gather more diverse priors. An analogy could be that of a genetic algorithm which does not get stuck in a local maxima because of its ability to sample a larger part of the parameter space.

This work is supported in part by the NASA grant 08-AISR08-0085, the NSF grant AST-0909182, and the U.S. Virtual Astronomical Observatory; NS and YC were supported in part by the Caltech SURF program. We thank numerous collaborators and colleagues for stimulating discussions.

REFERENCES

[1] CSS: http://www.lpl.arizona.edu/css/

[2] NEAT: http://neat.jpl.nasa.gov/

[3] LINEAR: http://www.ll.mit.edu/LINEAR/

[4] Pan-STARRS: http://pan-starrs.ifa.hawaii.edu/public/

[5] LSST: http://www.lsst.org/

[6] VOEventNet: http://www.voeventnet.org/

[7] Palomar-QUEST: http://palquest.org

[8] Titterington, D. M., et al. (1981), Comparison of discrimination techniques applied to a complex data set of head injured patients (with discussion), *J. Royal Stat. Soc. Ser. A*, **144**, 145.

[9] Hand, D., & Yu, K. (2001), Idiot's Bayes not so stupid after all?, *Intl. Statistical Review*, **69**, 385.

[10] R. Romano, C. Aragon, and C. Ding (2006), Supernova Recognition using Support Vector Machines LBNL-61192, Proc. 5th Int'l. Conf. Machine Learning Applications.

[11] Cristianini, N. & Shawe-Taylor, J. (2000), *An Introduction to Support Vector Machines and other kernel-based learning method*s, Cambridge University Press.

[12] Ball N.M., & Brunner R.J., (2010), Data Mining and Machine Learning in Astronomy, *Int. J. Mod. Phys. D*, arXiv/0906.2173

[13] Turmon, M., Pap, J. M, & Mukhtar, S. (2002), Statistical pattern recognition for labeling solar active regions: Application to SoHO/MDI imagery, *Astrophys. J.*, **568**, 396.

[14] Silverman, B. W. (1986), *Density Estimation for Statistics and Data Analysis*, London: Chapman and Hall publ.

[15] Loredo, T. J. & Chernoff, D. F. (2003), Bayesian Adaptive Exploration, *Statistical Challenges in Modern Astronomy III*, eds. E. D. Feigelson and G. J. Babu, Berlin: Springer Verlag, p. 57.

[16] Djorgovski, S. G., et al. (2001), Exploration of Large Digital Sky Surveys, in: *Mining the Sky*, eds. A.J. Banday et al., ESO Astrophysics Symposia, p. 305, Berlin: Springer Verlag.

[17] Djorgovski, S.G., Drake, A., et al. (the CRTS survey team) (2011), The Catalina Real-Time Transient Survey (CRTS), in "The First Year of MAXI: Monitoring Variable X-ray Sources", eds. T. Mihara & N. Kawai, Tokyo: JAXA Special Publ., in press. [18] Ripley, B. D. 1996, *Pattern Recognition and Neural Networks*, Cambridge Univ. Press.

[18] Ripley, B. D. (1996), *Pattern Recognition and Neural Networks*, Cambridge Univ. Press.

[19] Weir, N., Fayyad, U., Djorgovski, S.G., & Roden, J. (1995), The SKICAT System for Processing and Analyzing Digital Imaging Sky Surveys, *PASP*, **107**, 1243-1254.

[20] Odewahn, S., et al. (2004), The Digitized Second Palomar Observatory Sky Survey (DPOSS). III. Star-Galaxy Separation, *Astron. J.*, **128**, 3092-3107.

[21] John, G., & Langley, P. (1995), Estimating continuous distributions in Bayesian classifiers, *Proc. Eleventh Conf. on Uncertainty in Artificial Intelligence (UAI)*, 338–345, Morgan-Kaufmann publ.

[22] Cover, T, & Thomas, J. (1991), *Elements of Information Theory*, New York: Wiley Publ.

[23] Djorgovski, S. G., et al. (2001), Exploration of Parameter Spaces in a Virtual Observatory, in: *Astronomical Data Analysis*, eds. J.-L. Starck & F. Murtagh, *Proc. SPIE*, **4477**, 43.

[24] Djorgovski, S. G., et al. (2006), Some Pattern Recognition Challenges in Data-Intensive Astronomy, in: *Proc. 18th International Conference on Pattern Recognition (ICPR 2006)*, Vol. 1, eds. Y.Y. Tang et al., *IEEE Press*, p. 856.

[25] Catalina Sky Survey (CRTS): http://crts.caltech.edu/

[26] Djorgovski, S. G., et al. (PQ survey team) (2008), The Palomar-Quest digital synoptic sky survey, *Astonomische Nachrichten*, **329**, 263.

[27] Bailey, S., et al. (2007), How to Find More Supernovae with Less Work: Object Classification Techniques for Difference Imaging, *Astrophys. J.*, **665**, 1246.

[28] Mahabal, A., et al. (PQ survey team) (2008), Automated probabilistic classification of transients and variables, *Astonomische Nachrichten*, **329**, 288.

[29] Paczynski, B. (2000), Monitoring All Sky for Variability, *Publ. Astron. Soc. Pacific*, **112**, 1281.

[30] Mahabal, A., et al. (2008), Towards Real-Time Classification of Astronomical Transients, *AIP Conf. Ser.*, **1082,** 287**.**

[31] Donalek, C., et al. (2008), New Approaches to Object Classification in Synoptic Sky Surveys, *AIP Conf. Ser.*, **1082,** 252**.**

[32] Heckerman, D. (1999), A Tutorial on Learning with Bayesian Networks, in: *Learning in Graphical Models*, ed. M. Jordan, Cambridge, MA: MIT Press.

[33] Cowell, R., Dawid, P., Lauritzen, S., & Spiegelhalter, D. (2007), Probabilistic Networks and Expert Systems: Exact Computational Methods for Bayesian Networks, New York: Springer.

[34] Rasmussen, C., & Williams, C., (2006), *Gaussian Processes for Machine Learning,* Cambridge, MA: MIT Press.

[35] Cesa-Bianchi, N., & Lugosi, G. (2006). *Prediction, Learning and Games*. Cambridge University Press.

[36] Richardson M., & Domingos P., 2006, Markov logic networks, *Machine Learning*, **62**, 107-136.

[37] Wang J., & Domingos P., (2008), Hybrid Markov logic networks, *Proc. Twenty-Third National Conference in Uncertainty in Artificial Intelligence*, Chicago, IL, AAAI Press.

[38] Robbins, H. (1952), Some Aspects of the Sequential Design of Experiments, *Bull. Am. Math. Soc.*, **58**, 527-535.

[39] U.S. Virtual Astronomical Observatory (VAO), http://www.us-vo.org/

[40] International Virtual Observatory Alliance (IVOA), http://www.ivoa.net/

[41] Vermorel, J. and Mohri, M. (2005), ECML, **3720**, pp 437-448.

[42] National Virtual Observatory Science Definition Team (2002): http://www.us-vo.org/sdt/

[43] Rau, A., et al. (the PTF team), (2009), Exploring the Optical Transient Sky with the Palomar Transient Factory, *PASP*, **121**, 1334-1351,

[44] Mahabal, A., Wozniak, P., Donalek, C., & Djorgovski, S.G. (2009), Transients and Variable Stars in the Era of Synoptic Imaging, in: *LSST Science Book*, eds. Z. Ivezic, et al., Ch. 8.4, p. 261; available at http://www.lsst.org/lsst/scibook

[45] Mahabal, A., et al. (2010), Mixing Bayesian Techniques for Effective Real-time Classification of Astronomical Transients, in: *Proc. ADASS XIX*, ed. Y. Mizumoto, *ASP Conf. Ser.*, **434**, 115.

[46] Mahabal, A., et al. (2010), Classification of Optical Transients: Experiences from PQ and CRTS Surveys, in: *Gaia: At the Frontiers of Astrometry*, eds. C. Turon, et al., *EAS Publ. Ser.* **45**, 173, Paris: EDP Sciences.

[47] Mahabal, A., et al. (2010), The Meaning of Events, in: *Hotwiring the Transient Universe*, eds. S. Emery Bunn, et al., Lulu Enterprises Publ. http://www.lulu.com/, p. 31.

[48] Bloom, J., & Richards, J. (2011), Data Mining and Machine-Learning in Time-Domain Discovery & Classification, in: *Advances in Machine Learning and Data Mining for Astronomy*, in press; arXiv/1104.3142

ANOMALY CONSTRUCTION IN CLIMATE DATA: ISSUES AND CHALLENGES

JAYA KAWALE*, SNIGDHANSU CHATTERJEE***, ARJUN KUMAR*, STEFAN LIESS**,
MICHAEL STEINBACH*, AND VIPIN KUMAR*

Abstract. Earth science data consists of a strong seasonality component as indicated by the cycles of repeated patterns in climate variables such as air pressure, temperature and precipitation. The seasonality forms the strongest signals in this data and in order to find other patterns, the seasonality is removed by subtracting the monthly mean values of the raw data for each month. However since the raw data like air temperature, pressure, etc. are constantly being generated with the help of satellite observations, the climate scientists usually use a moving reference base interval of some years of raw data to calculate the mean in order to generate the anomaly time series and study the changes with respect to that.

In this paper, we evaluate different measures for base computation and show how an arbitrary choice of base can skew the results and lead to a favorable outcome which might not necessarily be true. We perform a detailed study of different base selection criterion and base periods to highlight that the outcome of data mining can be sensitive to choice of the base. We present a case study of the dipole in the Sahel region to highlight the bias creeping into the results due to the choice of the base. Finally, we propose a generalized model for base selection which uses Monte-Carlo based methods to minimize the expected variance in the anomaly time-series of the underlying datasets. Our research can be instructive for climate scientists and researchers in temporal domain to enable them to choose the right base which would not bias the outcome of the results.

1. Introduction

An important component of Earth Science data is the seasonal variation in the time series. Seasons occur due to the revolution of the Earth around the Sun and the tilt of the Earth's axis. The change in seasons brings about annual changes in the climate of the Earth such as increase in temperature in the summer season and decrease in temperature in the winter season. The seasonality component is the most dominant component in the Earth science data. For example, consider the time series of monthly values of air temperature at Minneapolis from 1948-1968 as shown in Figure 1. From the figure, we see that there is a very strong annual cycle in the data. The peaks and valleys in the data correspond to the summer and winter season respectively and occur every year. The seasonal patterns even though important are generally known and hence uninteresting to study. Mostly, scientists are interested in finding non-seasonal patterns and long term variations in the data. As a result of the effect of seasonal patterns, other signals in the data like long term decadal oscillations, trends, etc. are suppressed and hence it is necessary to remove them. Climate scientists usually aim at studying deviations beyond the normal in the data.

In order to remove seasonality from the raw data, climate scientists generally remove the monthly mean value from the raw data. For example, although more than 100 years of data are available for the temperature anomaly time series at the National Climatic Data Center, only the 100 years 1901-2000 are used to calculate the annual cycle [3]. Often, climate scientists only take 30 years as a reference interval and construct anomalies with respect to that interval. There are several important results and implications derived from the anomalies constructed using a short reference base. In general, climate data has complex structures due to spatial and temporal autocorrelation.

*Department of Computer Science, University of Minnesota, kawale, arkumar, steinbac, kumar@cs.umn.edu
Department of Soil, Water and Climate, liess@umn.edu * School of Statistics. University of Minnesota chatterjee@stat.umn.edu.

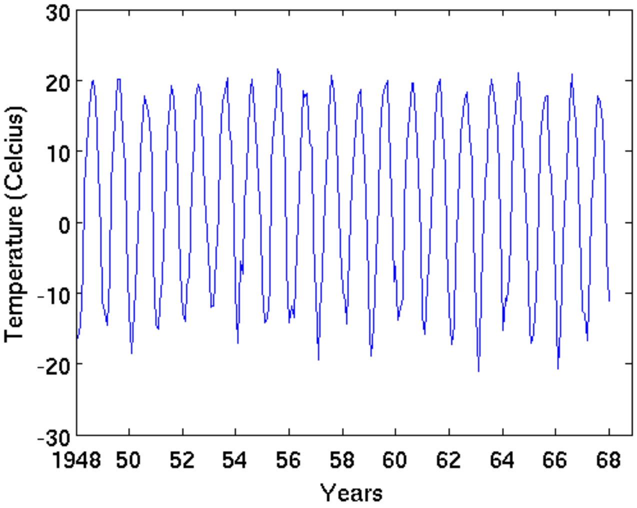


FIGURE 1. The figure shows the monthly mean air temperature at Minneapolis for a 20 year period. From the figure we can see that there is a very high annual cycle and the temperatures go up and down with the change of seasons.

The choice of the base significantly impacts the patterns that can be discovered from it and some really important climate phenomenons are computed using a fixed base. For example, teleconnections or long distance connections between two regions on the globe are represented by time series called *climate indices*. Climate indices are time series that summarize the behavior of the selected regions and are used to characterize the factors impacting the global climate. These climate indices are computed by the Climate Prediction Center [1] using a moving 30-year base period and currently they use a base period of 1981-2010. Another important set of results computed using a fixed base are incorporated in the International Panel on Climate Change (IPCC) Fourth Assessment Report on understanding climate change[13].

In this paper, we show how an arbitrary choice of base can skew the results and lead to favorable outcome which might not necessarily be true. We examine four simple criterions for base selection and empirically evaluate the differences in them. Our empirical evaluation of the different measures reveals that the z-score measure is quite different from the other measures like mean, median and jackknife. We further study the impact of using different base period to highlight that the outcome of further analysis can be sensitive to the choice of the base. We present a case study of the Sahel region to show that the dipole in precipitation in the region moves around and even disappears with the choice of a different base. Finally, we propose a generalized model for base selection which uses Monte-Carlo based sampling methods to minimize the expected variance of the underlying datasets. Our research can be especially instructive to climate scientists in helping them construct a generalized anomaly that does not create a bias in their analysis. Further, other researchers in temporal domain can also benefit from our work and it will enable them to choose a bias-free base. The main contributions of our paper are as follows:

- We present a systematic evaluation of four different measures of computing the base to construct the anomalies. Our evaluation shows that using the mean for anomaly computation might not be the right thing to do.

- We show that using a short base reference introduces a bias in the variance and show an alternative approach to take care of the bias.

- We present a case study of Sahel region to highlight that outcome of further analysis like dipole detection can be sensitive to the selection of the base.

- We propose an algorithm based on Monte Carlo sampling for automatic selection of the appropriate base which minimizes the variance across the time series. The algorithm suggests using weighted base of 55 year time period rather than 30 years for our data spanning 62 years.

The paper is organized as follows: In Section 2, we define the related work examining the issues with base construction. In Section 3, we describe the dataset used for our study. In Section 4, we examine the different measures and different time periods that can be used for anomaly construction. In Section 5, we describe our experiments evaluating the different measures and time periods. We also present a case study of the Sahel region to show the impact of the different base period in dipole analysis. In Section 6, we present a generalized approach for anomaly construction that computes a weighted anomaly using Monte Carlo sampling and also present our results based on the approach. Section 7 includes the discussions and the future road map for our work.

2. Related Work

Anomaly computation is a fundamental problem in climate science as most of the analysis of climate data relies upon computation of the anomalies as the first step. There have been some studies in the climate domain analyzing anomalies. Climate scientists mostly use a 30 year period to construct the anomalies and remove the annual cycle. Other ways to remove the annual cycle are 1) computing second moment statistics over each individual season by removing the first two harmonics of the respective time series; and 2) averaging the second moment statistics over all years. More techniques to remove the annual cycle include removing the first two or three harmonics (periods of 365.25, 182.625, and 121.75 days) e.g., [8] and [4]. Some of the less common practices involve looking at more sophisticated techniques like removing the cyclostationary empirical orthogonal function [11] or bandpass filtering, e.g. using a low-pass filter with 0.5 cycle/year [15]. More general methods are described in Wei et al. [19].

However, these procedures fail to take into account the natural interannual variability that should remain visible in the data. Therefore the procedures result in biased estimates of certain statistics [17]. In particular, lag-autocorrelations are systematically negatively biased, which indicates that uncertainty is added to climate data. Trenberth [17] shows for first order autoregressive time series that the autocorrelations computed after the annual cycle is removed become negative after just a few days lag. Consequently, the stochastic character of meteorological time series can result in less statistically significant analysis. Kumar *et al.* [12] state that the analysis of observed climate data often lacks separation of the total seasonal atmospheric variance into its external and internal components, with external components being the influence of atmospheric initial conditions, the coupled air-sea interactions, and boundary conditions other than sea surface temperatures, whereas internal components are described by the atmospheric variability over time. Removing the annual cycle should provide insight into the internal variability while leaving the external forcing intact.

Tingley *et al.* [16] discuss the impact of using a short reference interval in anomaly construction. They show that using a short reference period, the variance of the records at the time interval is reduced and inflated elsewhere. They show that the choice of the reference interval has a significant impact on the second spatial moment of the time series in the temperature data set whereas the first moment of the time-series is largely unaffected . They further use two factor ANOVA model within a Bayesian inference framework.

Despite the importance of anomalies in the further impact on the results, there is no firm consensus on how to deal with the systematic construction of anomalies and their impact on the various results.

Apart from this, the authors are not aware of any systematic study comparing the different aspects of anomaly construction in the climate data.

3. Dataset

We use the data from the NCEP/NCAR Reanalysis project provided by the NOAA / OAR/ ESRL PSD, Boulder, Colorado, USA [9]. The goal of the NCEP project is to produce a comprehensive atmospheric analysis using historical data (1948 onwards) from observations as well as other analysis like projection. As a result of these analysis, there is a complete data assimilation for every grid point on the Earth.

The NCEP/NCAR Reanalysis project has data assimilated from 1948 – present and is available for public download at [2]. We use the monthly time resolution of data and it has a grid resolution of 2.5° longitude x 2.5° latitude on the globe. We use the precipitation, air temperature and sea level pressure data for our analysis as they represent the most important climate variables. In all, we have 62 years of data (corresponding to 744 monthly values) for 10512 grid locations on the globe.

4. Different aspects of Anomaly Construction

We examine two aspects of anomaly construction: 1) the measure for anomaly construction and 2) the period used for anomaly construction in the following subsections.

4.1. Different measures for Anomaly Construction.

The central idea behind anomaly construction is to split the data into two parts: (a) data with expected behavior, and (b) anomaly data that shows the variability from the expected, which is generally used for understanding climate change phenomenon. For a given location i, its anomaly times series f'_i is constructed from the raw time series f_i by removing a base vector b_i from it as follows:

$$f'_i = f_i - b_i \tag{1}$$

A simple measure of computing the base b_i is by taking the mean of all data $(\overline{f}_i)$ present for location i. However the sample mean would not be a good measure as the Earth science data is associated with a large amount of seasonality. In order to account for this the base b_i is computed by taking a monthly mean for each month separately. It is not yet clear whether the mean is the right way to compute the base or if there is a better measure to compute the base. We examine four simple measures of base computations as follows:

- **Mean**: In this measure, the monthly mean values of the raw data are considered as the base and subtracted from the data to get the anomaly series.
- **Z-score**: Another possible way to construct the anomalies is to remove the monthly z-score values from the raw data. The z-score also accounts for the standard deviation in the monthly values.
- **Median**: This is constructed by removing the monthly median values instead of the monthly means as median can be a more robust measure when the data is skewed.
- **Jackknife**: This approach involves considering all points apart from the point itself in the computation of the mean and variance measures and it produces an unbiased estimation of variance just like Maximum aposterior Estimate (MAP).

We elaborate these measures and how they are computed in the following sub-sections.

4.1.1. *Mean.* Monthly mean computation is the most widely used method to extract the anomalies from the raw data. The mean subtraction makes the anomaly time series to have a zero mean. More formally,

$$f'_i(t,m) = f_i(t,m) - \mu_m, \forall t \in \{total - start, \ldots, total - end\}, \forall m \in month \tag{2}$$

where total-start and total-end values represent the actual size of the data. In general it is known that taking mean would minimize the variance of the resulting series but it can also lead to over fitting and conclusions that might not be true. Further, instead of using the entire data for base computation, a short reference interval can be chosen. For example, if the data begins from 1900 to 2010, the base start and end years could be chosen as 1960-1990. We further discuss the issue of choosing a short base in Section.4.2.

4.1.2. *Z-score.* The z-score normalization ensures that the resulting anomaly series has mean = 0 and standard deviation = 1. As a result, z-score can be considered to be more robust than the mean but at the same time z-score based standardization can eliminate variations across different locations on Earth which might not be desirable. The z-score measure is computed as follows:

$$f'_i(t,m) = \frac{f_i(t,m) - \mu_m}{\sigma_m}, \forall t \in \{total - start, \ldots, total - end\}, \forall m \in month \tag{3}$$

4.1.3. *Median.* In scenarios where data is skewed, mean can be sensitive to outliers. In such settings, median is typically considered to be more robust to outliers. As a result, we consider median as a method for base computation:

$$f'_i(t,m) = f_i(t,m) - median_m, \forall t \in \{total - start, \ldots, total - end\}, \forall m \in month \tag{4}$$

4.1.4. *Jackknife estimate.* The Quenouille Tukey jackknife approach [20] is a useful nonparametric estimate of mean and variance. The basic idea behind the jackknife estimator is to systematically compute the mean estimate by leaving out one observation at a time from the sample set. Let $f_1, f_2, \ldots, f_n$ be the n points in the time series of a location x. The jackknife mean estimate is computed at point f_i by taking the mean of all points except f_i as follows:

$$Mean(f_i) = \frac{f_1 + \ldots + f_{i-1} + f_{i+1} + \ldots + f_n}{n-1} \tag{5}$$

Thus the anomalies are constructed by excluding the value at each point f^i_x. We however still use all the monthly values only to compute the jackknife estimate at each point. The variance measure using the jackknife approach turns out to be:

$$Variance = \left(\frac{n}{n-1}\right)^2 \times Variance(f_1, \ldots, f_n) \tag{6}$$

In order to see this consider $f_1, \ldots, f_n$ to be variable during a given month. Then we have to following:

$$\begin{aligned} Variance &= \frac{1}{n}\sum_i (f_i - Mean(f_i))^2 \\ &= \frac{1}{n}\sum_i (f_i - \frac{n}{n-1} \times Mean + \frac{1}{n-1} \times f_i)^2 \\ &= \frac{1}{n}\sum_i \frac{n^2}{(n-1)^2} \times (f_i - Mean)^2 \\ &= \left(\frac{n}{n-1}\right)^2 \times Variance(f_1, \ldots, f_n) \end{aligned}$$

The variance essentially turns out to be an unbiased estimate and is similar to the maximum aposterior probability (MAP) estimate of the model. MAP is similar maximum likelihood estimate (MLE) but also incorporates a prior distribution over the quantity one wants to estimate. MAP estimation can therefore be seen as a more robust form of MLE estimation. However the main problem with an approach based on jackknife is that it requires a lot of computation.

4.2. **Different Time Periods for Anomaly Construction.** As mentioned earlier, an anomaly series is constructed from the raw time series by removing a base value from it. The base value is generally considered to be the mean of the data. Since the true theoretical mean is not known, the base value is created by taking the sample mean of the data. However, most of the times a short reference interval is chosen to compute the base and changes with respect to that are studied. There is no absolute truth or guidelines available to choose the reference interval. Climate scientists generally choose the base as a moving 30 year period and study the changes with respect to that. However a moving short reference interval is problematic and can result in spurious results and conclusions. In order to highlight the problems associated with picking an arbitrary short base, we consider an example of teleconnections looking into the drought of the Sahel region in Africa in the Section 5.3.

5. Experiments and Results

5.1. **Comparison of Different Measures of Anomaly Construction.** Our first task is to empirically evaluate the differences in the four different measures described in Section 4.1. We use the precipitation data for our analysis. Using all the 62 years of data from the NCEP/NCAR website, we first construct an anomaly series for each location on the Earth using the four different measures. Further, we also construct complex networks by taking pairwise correlation between all locations on the Earth as used by several researchers like [14], [10], [5], [18] to find patterns in climate data. The nodes in the graph represent all the locations on the Earth and the edges represent pairwise correlation between the anomaly time series of all the nodes on the Earth. Our goal is to evaluate whether there are statistically significant differences between different measures to compute anomalies. In order to measure the statistically significant difference, we consider the following three criterion:

- *Mean based difference*: We compute the mean of anomaly time series using different measures and then compute the difference in mean for each pair of measure. The mean difference would be statistically significant if we can say with 95% confidence that the mean of difference is non-zero.
- *Correlation based difference*: Here we compute the correlation of every point with respect to other points on the globe using the four measures and check if the correlation values are impacted by using different measures for anomaly construction.
- *Monthly variance based difference*: Here we check if the monthly variance of the anomaly time series at each location is different for pairs of anomaly computation measure or not.

We use *t-test* to test if the difference between two measures follows a Gaussian distribution with $mean = 0$ and unknown variance. Thereby, our null hypothesis, $H0$ is that two measures lead to the same result and alternate hypothesis, Ha is that the two measures are different.

Tables 1, 2 and 3 show the number of locations where two measures lead to significant differences in the anomaly time series. Here we make an observation that z-score and median lead to significant differences from each other as well as the mean and the jackknife. Z-score based base computation yields the most significant difference as it leads to statistically significant changes in correlations and monthly variances at more than 9000 grid locations on the Earth. The z-score measure also stands out if we look at the monthly variance of each point. On the other hand, mean and jackknife seem to be similar. Median differs from the two over the mean difference based comparison. This is perhaps expected as the median and the mean values are not the same and all the other bases have zero monthly mean. Overall this result indicates that different measures used to compute base can lead to drastically different results. This result makes it intuitively clear that z-score might not be the best way to compute the base. In order to compare the mean and the median, we examine the skew in the anomaly time series after using the mean to construct the anomalies. To determine the skew, we check the *kurtosis* of the anomaly series at each location on the Earth. The kurtosis falls within the range of 2.6-3.5 for more than half of the locations on the Earth which is acceptable for a normal distribution. However, some locations have a very high skew and the kurtosis value is as high as 10. Fig 2 shows the histogram of kurtosis for all the locations and also shows the skew in the time series at a random location on the Earth. This suggests that the mean might not a good measure to compute the anomalies and median might be a better choice. However, further investigations are still needed to understand the right measure for anomaly computation.

Method	Mean	Median	z-score	Jackknife
Mean	-	692	0	0
Median	-	-	1281	674
z-score	-	-	-	0
Jackknife	-	-	-	-

Table 1. Number of locations that rejected the null hypothesis at 95% confidence interval in the two sample t-test examining the **anomalies** at the different locations for precipitation.

Method	Mean	Median	z-score	Jackknife
Mean	-	0	5303	0
Median	-	-	5152	0
z-score	-	-	-	5303
Jackknife	-	-	-	-

Table 2. Number of locations that rejected the null hypothesis at 95% confidence interval in the two sample t-test examining the **correlation of each location** with the different locations for precipitation.

Method	Mean	Median	z-score	Jackknife
Mean	-	0	9152	0
Median	-	-	9152	0
z-score	-	-	-	8998
Jackknife	-	-	-	-

Table 3. Number of locations that rejected the null hypothesis at 95% confidence interval in the two sample t-test examining the **monthly variance** at the different locations for precipitation.

5.2. **Comparison of different time periods for anomaly construction.** The previous results show that for a fixed base period there exists different ways to compute the base which can lead to drastic differences in the anomaly time series. *Here we try to see if we can fix a measure (say mean) then check if varying the base period affects the anomaly time series.* In order to do this, we examine three base periods: a) first 20 years b) entire 62 years and c) last 20 years. We also experiment with base period length of 30 years but that leads to similar results so for the sake of presenting the extremes, we present results choosing 20 years as a base.

We construct the anomaly series for each location corresponding to the given base periods. We selected mean as the measure of computing the base. In order to compare the time series, we used KL-divergence criteria to see if different base periods have different effects on anomaly time series.

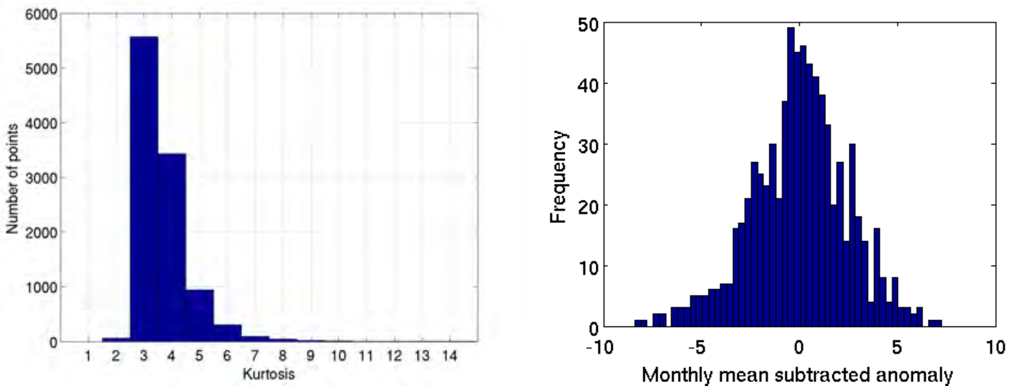


Figure 2. a) Kurtosis histogram b) The mean subtracted anomaly shows a skew in the data.

The KL-divergence is defined as follows:

$$D_{\mathrm{KL}}(P\|Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)} \tag{7}$$

KL-divergence of 0 means that the two series are exactly the same. A KL-divergence value indicates that the series are quite different. We plot the divergence value for each location on the globe in Figure 3. The white region shows that these locations are severely affected by our choice of base. In general last 20 years vs 62 years (second figure) has a light shade of gray indicating that all the locations on earth would be affected (in their anomaly series) if we make a choice between last 20 years vs all 62 years.

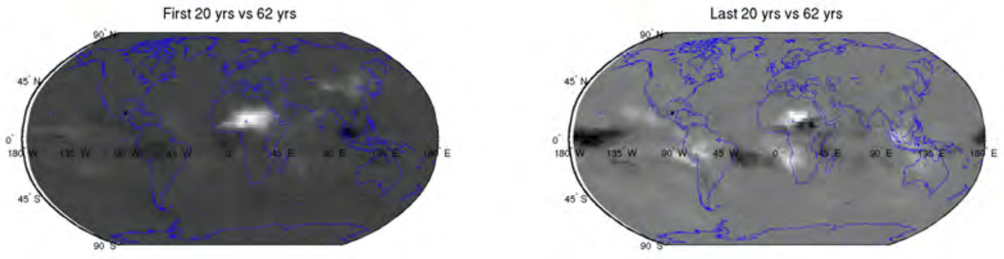


Figure 3. KL-divergence of the anomaly series the different bases a) first 20 years vs entire 62 years b) last 20 years vs entire 62 years and c) first 20 years vs last 20 years. The white shaded regions represent regions of maximum divergence.

Also the variance in the anomaly time series changes when we move the 20 year base period across the entire length of the time-series. Fig. 4 shows the change in variance at two random locations by picking up different 20 year base periods by varying the starting times from 1948-1988. From the figure, we see that average variance in the anomalies at different points varies using different start times for the base periods. This makes the problem complex as different regions show minimum variance in different windows of the time period.

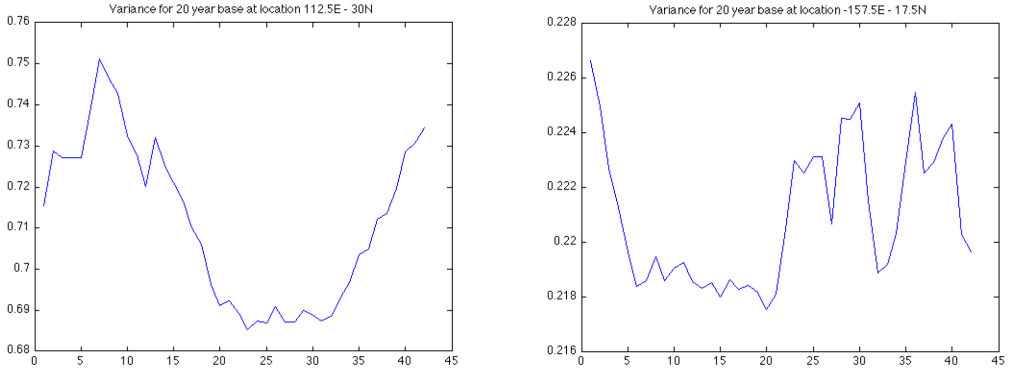


Figure 4. Change in variance of two random locations on the Earth choosing a 20 year reference period and moving the starting year from 1948-1988.

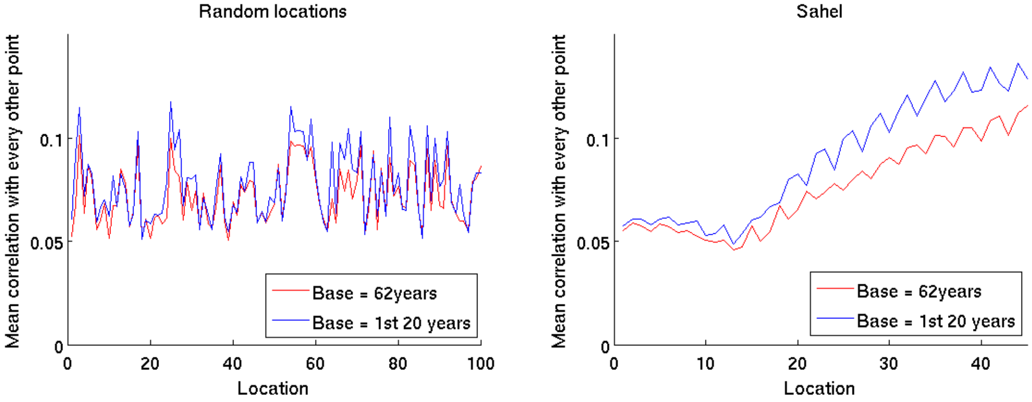


Figure 5. a) Mean correlation of 100 random points with all the points in the globe for precipitation using the entire 62 years as the base and only the first 20 years as the base. b) Mean correlation of 100 random points with all the points in the globe for precipitation using the entire 62 years as the base and only the first 20 years as the base. The difference in correlation (red and blue) is much more pronounced in Sahel as compared to the random locations.

In order to further analyze the impact of the choice of short reference base on the correlation of anomaly time-series, we consider two anomaly construction scenarios using the base as: a) first 20 years from 1948-1967 and b) using the entire 62 year time period from 1948-2009. We examine the changes in the mean correlations of locations with respect to every other location on the Earth using the two base period. Figure 5 shows the change in mean correlation of 100 random locations and the locations in Sahel using the two base periods to compute the anomalies. From the figures, we see that the locations in Sahel are much more impacted by the change in the base period as compared to the 100 random locations. We also find similar trends in other variables like pressure and temperature in Sahel but do not report it due to lack of space. These results underline the fact that a reference interval is crucial in the computation of the anomalies. In the next section, we show a case study on teleconnections where the actual analysis results and implications are impacted by the choice of the reference base.

5.3. **Case study of the Sahel dipole.** *Teleconnections* are long distance connections connecting the climate of two places on the Earth. One such class of teleconnections are the dipoles which consist of two regions having anomalies in the opposite direction and thus having negative correlation. The climate in Sahara and Sahel region of Africa has undergone some radical shift in the past century. The region received heavy rainfall till about 1969 until when it went into a period of severe drought for about 30 years which brought a *regime shift* in the region. The drought in the region and its environmental causes and consequences have been well studied in the past [6].

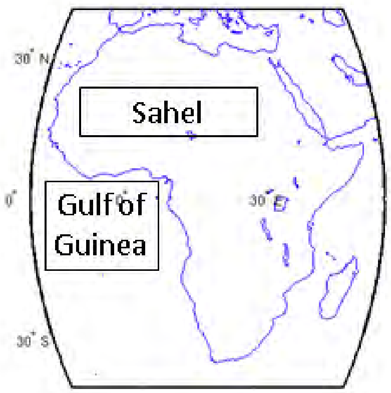


FIGURE 6. Sahel and the Gulf of Guinea in Africa.

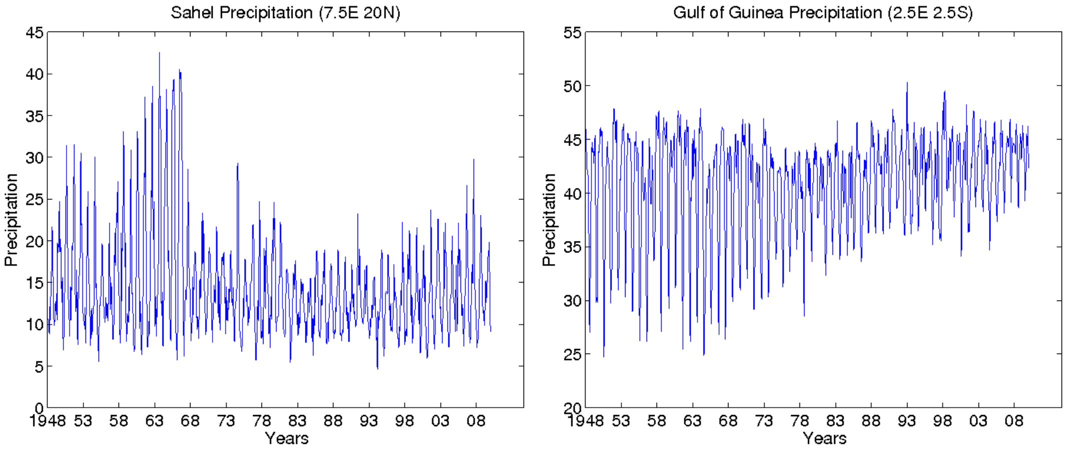


FIGURE 7. Raw precipitation time-series at Sahel and the Gulf of Guinea.

The precipitation in the region has recovered slightly but not enough to come back to the same levels as that before 1969. The severe loss of precipitation at Sahel was accompanied with a heavy increase in precipitation at the same time in the Gulf of Guinea around Africa, thus forming a dipole in precipitation [7]. The two regions Sahel and the Gulf of Guinea are marked in the Fig.6. The raw precipitation time series of the two locations in Sahel (7.5E, 20N) and the Gulf of Guinea (2.5E, 2.5S) are shown in the Fig. 7.

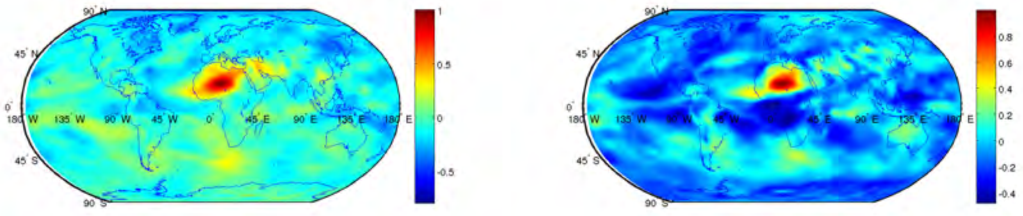


Figure 8. Correlation of precipitation time series for different places on the Earth with respect to a single point in Africa using the two base a)1948-1967 and b)1987-2008 for the 1st network (1948-1967). The figure shows the presence of a dipole (positive and negative correlations as shown by red and blue regions) in the **right** picture and not the left one.

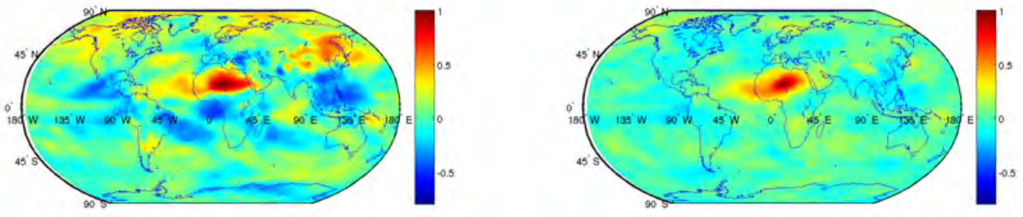


Figure 9. Correlation of precipitation time series for different places on the Earth with respect to a single point in Africa using the two base a)1948-1967 and b)1987-2008 for the 2nd network (1968-1987).

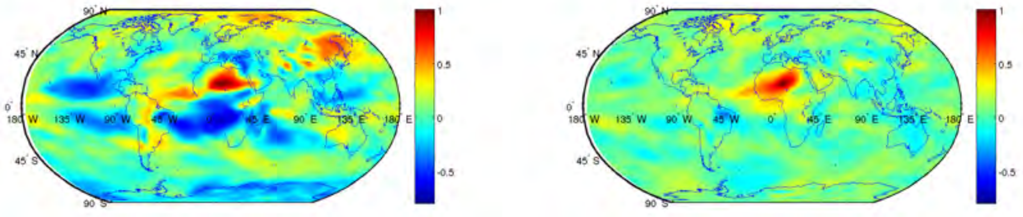


Figure 10. Correlation of precipitation time series for different places on the Earth with respect to a single point in Africa using the two base a)1948-1967 and b)1987-2008 for the 3rd network (1987-2007). The figure shows the presence of a dipole (positive and negative correlations as shown by red and blue regions) in the **left** picture and not the right one.

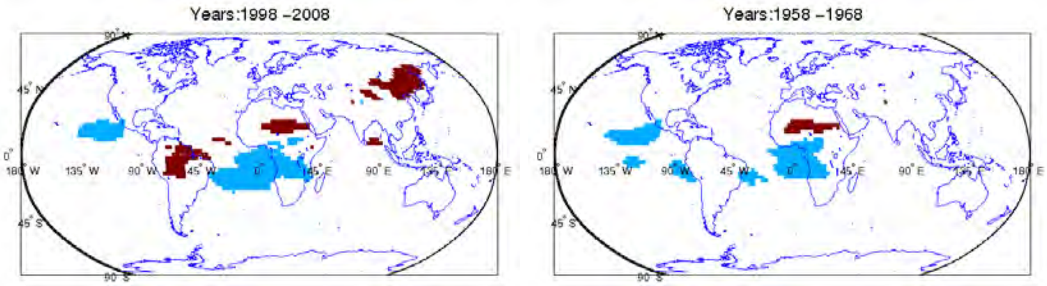


FIGURE 11. Different regions and different time periods are identified as dipoles in precipitation using the first 20 years as the base and the last 20 years as the base. (The red and blue regions represent two ends of the dipole and have negative correlation in their anomalies.)

From the figure, we can see the dramatic decrease in precipitation in Sahel and an increase in precipitation in the Gulf of Guinea around the 1970s. Now using the 62 years of NCEP precipitation data, we choose two base years, the first 20 years (1948-1967) and the last 20 years(1988-2007). We further construct three networks by taking pairwise correlation between the anomaly time series of all locations on the Earth for a 20 year time period each (1948-1967, 1968-1987 and 1988-2007). Consider the point A(7.5E, 20N) in Sahel. Let us examine the correlations of this point with all the regions on the Earth. Fig. 8, 9 and 10 show the correlation of all the points on the Earth with respect to a single point in Sahel for the three time periods 1948-1967,1968-1987 and 1988-2007 respectively. From the figures, we see that the if we choose the base period to be the first 20 years, the Sahel dipole is clearly visible (positive and negative correlations as shown by red and blue regions in the figures) in the period 1988-2007, however if we choose the last 20 years as the base period the dipole is seen in the interval 1948-1967. Further we use the dipole detection algorithm on the complete network as given in [10]. The algorithm begins by picking up the most negative edge on the Earth and grows the two ends of the negative edge into two regions such that they are negatively correlated with each other and positively correlated within each other. Using the algorithm, we see that the Sahel dipole appears in different time periods and also in different regions as also shown in the Fig.11. Thus the choice of a base period severely impacts the results and subsequently the interpretations that can be drawn from the results. Hence extra caution needs to be exercised while constructing anomalies in order to avoid spurious conclusions to be drawn from the results.

6. A Generalized Approach for Anomaly Construction

In the previous section, we saw that there is a bias introduced in the results upon considering different measures of the base and different durations. So the primary question arises, *What is the right base to choose for anomaly construction?*. In this section, we discuss our approach to handle the problem of the anomaly construction. The intuition behind our approach is to have a weighted mean to construct the anomalies and use an objective criteria to pick up the right set of weights using Monte Carlo sampling. The weighted base for anomaly construction for a location i is created as follows:

$$b_i(t, w) = \sum_{t=t0}^{t0+k} w_t * f_i(t) \quad subject\ to \sum_{t=t0}^{t0+k} w_t = 1 \tag{8}$$

where $t0$ represents the starting time period, k represents the length of the time period. We further assume that the weights w_t are the same for each year and do not depend upon the month in the year. Further, the anomalies are constructed by removing the weighted base for each month as

Algorithm 1: A Generalized approach for Anomaly construction.

```
Let $f_i(t)$ be the monthly values of raw time series of location $i$
Let, $N$ = Length of total time period.
Let, $T_{base}$ = Shortest length of reference interval.
Let, $NumSimulations$ = Number of simulations to run.
Initialize $GlobalVariance$, $OptimalWeights$ to $\infty$
repeat
  for $k \in T_{base}, ...T_N$ do
    for $t \in T_0, ...T_N$ do
      for $i \in 1....NumSimulations$ do
        Compute weight vector $w_1, w_2, ......, w_t$
        subject to the constraint $w_1 + w_2 + .... + w_t = 1.$ using a Dirchlet prior.
        Compute the weighted base as $b_i(t, w) = \sum_{t=t0+1}^{t0+k} w_t * f_i(t)$
        Compute the Anomalies from the weighted base as $f_i'(t, w) = f_i(t) - b_i(t, w)$
        Compute the Variance of all the anomaly time-series across the globe.
        if $Variance < GlobalVaraince$ then
          Update the $GlobalVariance$ and $OptimalWeight$
          $GlobalMedian = Median$
          $OptimalWeight = w_1, w_2, ......, w_t$
        end if
      end for
    end for
  end for
until convergence
```

follows:

$$f_i'(t, w) = f_i(t) - b_i(t, w) \tag{9}$$

We run Monte Carlo simulations to get the right set of the weights w_t and define the objective function as minimizing the variance of the anomaly time series over time and space. By minimizing the variance, we are trying to enforce uniformity over the data. There can be some other objective functions like the median of the lowest 10% of the correlations. The intuition behind this objective function is that for computing dipoles, we need to examine the most negative correlations. Hence we want to find a weight and a base vector corresponding to our criterion for dipoles. However, we consider a general objective function that is not dependent upon the problem. The further details of the algorithm are present in Algorithm 1.

6.1. **Results.** We use the precipitation data and run our Monte Carlo based simulation algorithm to get the right reference base period. Figure 12 represents the final converged weights. The other parameters of the final convergence of the algorithm are as mentioned in Table 4. Using our new weighted anomaly, we re-construct the correlation plots around the Sahel to get a sense of the dipole in the Sahel. Fig 13 shows the new results using the dipole in the Sahel using the weighted anomaly. Using a bias free base gives us confidence about the non-spuriousness of the discovered climate pattern or climactic phenomena such as a dipole. It implies that a dipole does exist in the region and that bases chosen which result in the dipole appear vanishing are not good bases. This objective function thus helps us in observing phenomena which would be more prominent if favorable bases are assumed but a bias-free base gives us a worst case scenario and more confidence in the results.

7. Discussion and Conclusions

The issue of anomaly construction is a fundamental problem in climate science as most of the analysis and results are derived after the raw data is transformed into an anomaly series. However there are no current guidelines available on anomaly construction and climate scientists usually

Table 4. Final algorithm convergence details.

Parameter	Value
Period	55
Starting year	1948

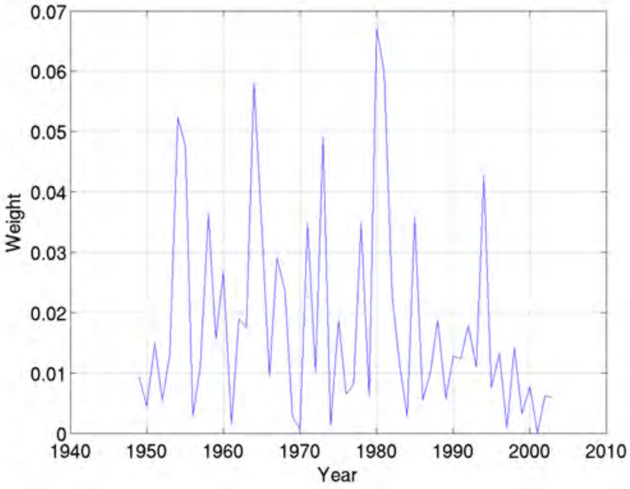


Figure 12. Final converged weight vector.

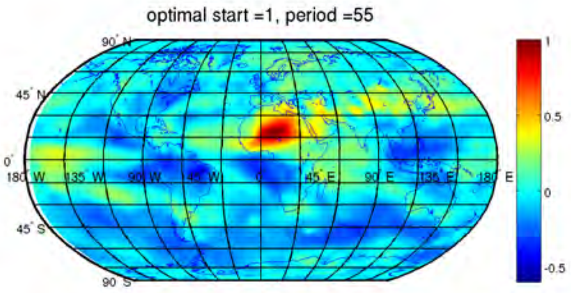


Figure 13. A correlation map as seen from the Sahel location A(7.5E, 20N).

rely upon computing a moving reference base for anomaly construction. In this paper, we examine the various issues pertaining to the construction of the anomalies. We assess the four methods of anomaly construction i.e. mean, median, z-score and jackknife. Our results show that if z-score is used as a measure for anomaly computation then the correlation values across different locations come out to be significantly different at 95% confidence interval. The mean, median and the jackknife measure do not show significant differences. However, due to the skewness in the data, the mean might not be a good measure and the median might be a good measure in such a case. However, further investigation is required to understand the right measure should be used.

We further show the bias in results introduced due to a choosing a short reference interval and show the difference in conclusions and results using a case study of the Sahel dipole. It is important to handle the bias introduced due to a short base as subsequent conclusions derived from it get affected. We further propose a generalized algorithm to handle the the issue of a bias-free base. Using our algorithm, we get the optimal base period to be 55 years. The algorithm can be modified to have different objective functions to handle different specific scenarios. As a part of our future work, we will examine different approaches to learn the weight vector as opposed to using the Monte Carlo simulations. We will also evaluate different objective measures and their impact on the base construction.

Acknowledgments

This work was supported by NSF grants IIS-0905581 and IIS-1029771. Access to computing facilities was provided by the University of Minnesota Supercomputing Institute.

References

[1] Climate prediction centre, http://www.cpc.ncep.noaa.gov/.

[2] Ncep data download link, http://www.esrl.noaa.gov/psd/data/.

[3] Temperature anomaly time series, national climatic data center, noaa, http://www.ncdc.noaa.gov/ghcnm/time-series/index.php.

[4] G. Compo, G. Kiladis, and P. Webster. The horizontal and vertical structure of east asian winter monsoon pressure surges. *Quarterly Journal of the Royal Meteorological Society*, 125(553):29–54, 1999.

[5] J. Donges, Y. Zou, N. Marwan, and J. Kurths. Complex networks in climate dynamics. *The European Physical Journal-Special Topics*, 174(1):157–179, 2009.

[6] J. Foley, M. Coe, M. Scheffer, and G. Wang. Regime shifts in the sahara and sahel: interactions between ecological and climatic systems in northern africa. *Ecosystems*, 6(6):524–532, 2003.

[7] A. Giannini, R. Saravanan, and P. Chang. Oceanic forcing of sahel rainfall on interannual to interdecadal time scales. *Science*, 302(5647):1027, 2003.

[8] C. Jones and J. Schemm. The influence of intraseasonal variations on medium-to extended-range weather forecasts over south america. *Monthly Weather Review*, 128(2):486–494, 2000.

[9] E. Kalnay, M. Kanamitsu, R. Kistler, W. Collins, D. Deaven, L. Gandin, M. Iredell, S. Saha, G. White, J. Woollen, Y. Zhu, M. Chelliah, W. Ebisuzaki, W. Higgins, J. Janowiak, K. C. Mo, C. Ropelewski, J. Wang, A. Leetmaa, R. Reynolds, R. Jenne, and D. Joseph. The ncep/ncar 40-year reanalysis project. *Bull. Amer. Meteor. Soc.*, 77:437–471, 1996.

[10] J. Kawale, M. Steinbach, and V. Kumar. Discovering dynamic dipoles in climate data. In *SIAM International Conference on Data mining, SDM*. SIAM, 2011.

[11] K.-Y. Kim and C. Chung. On the evolution of the annual cycle in the tropical pacific. *Journal of Climate*, 14(5):991–994, 2001.

[12] A. Kumar, B. Jha, Q. Zhang, and L. Bounoua. A new methodology for estimating the unpredictable component of seasonal atmospheric variability. *Journal of Climate*, 20(15):3888–3901, 2007.

[13] I. P. on Climate Change. *Fourth Assessment Report: Climate Change 2007: The AR4 Synthesis Report*. Geneva: IPCC, 2007.

[14] M. Steinbach, P. Tan, V. Kumar, S. Klooster, and C. Potter. Discovery of climate indices using clustering. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 446–455. ACM, 2003.

[15] D. Thomson. Dependence of global temperatures on atmospheric co2 and solar irradiance. *Proceedings of the National Academy of Sciences of the United States of America*, 94(16):8370, 1997.

[16] M. Tingley. A bayesian anova scheme for calculating climate anomalies, submitted 2011.

[17] K. Trenberth. Some effects of finite sample size and persistence on meteorological statistics. part i: Autocorrelations. *Mon. Wea. Rev*, 112:2359–2368, 1984.

[18] A. Tsonis, K. Swanson, and P. Roebber. What do networks have to do with climate? *Bulletin of the American Meteorological Society*, 87(5):585–595, 2006.

[19] L. Wei, N. Kumar, V. Lolla, E. Keogh, S. Lonardi, and C. Ratanamahatana. Assumption-free anomaly detection in time series. In *Proceedings of the 17th international conference on Scientific and statistical database management*, pages 237–240. Lawrence Berkeley Laboratory, 2005.

[20] C. Wu. Jackknife, bootstrap and other resampling methods in regression analysis. *The Annals of Statistics*, 14(4):1261–1295, 1986.

MITEXCUBE: MICROTEXTCLUSTER CUBE FOR ONLINE ANALYSIS OF TEXT CELLS

DUO ZHANG*, CHENGXIANG ZHAI*, AND JIAWEI HAN*

Abstract. A fundamental problem for analysis of multidimensional text databases is efficient and effective support of various kinds of online applications, such as summarizing the content of text cells and comparing the contents across multiple text cells. In this paper, we propose a new infrastructure called MicroTextCluster Cube (or *MiTexCube*) to support efficient online text analysis on multidimensional text databases by introducing *micro-clusters* of text documents as a compact representation of text content. Experimental results on a real multidimensional text database show that applications based on the proposed materialized *MiTexCube* are more efficient than the baseline method of direct analysis based on document units in each cell, without sacrificing much quality of analysis, and *MiTexCube* naturally accommodates flexible tradeoff between efficiency and quality of analysis.

1. Introduction

As large amount of unstructured text becomes available in multidimensional databases, it is increasingly important to support efficient online analysis of text data. While a search engine is useful to satisfy a user's *ad hoc* information needs, allowing a user to retrieve relevant documents through a keyword query, it is inadequate for analysis of bulky text information, which is necessary in many online applications. For example, while it is easy for a user to find documents discussing opinions about iPhone in a review database based on a search engine, it is hard to compare opinions expressed in different time periods or by different user groups. In contrast, if we can manage text data together with structured data with attributes such as time and user groups in a multidimensional database, we would be able to flexibly explore text data corresponding to different combinations of time and user groups and compare opinions across different contexts (e.g., different time periods).

As an example of multidimensional text database, consider the ASRS database, the largest repository of aviation safety information provided by the frontline personnel [1]. It has both structured data (e.g., time, airport, and light condition) and text data such as narratives about an anomalous event written by a pilot or flight attendant as illustrated in Table 1. A text narrative usually contains hundreds of words.

Table 1. An example of text database in ASRS

ACN	**Time**	**Airport**	⋯	**Light**	**Narrative**
101285	199901	MSP	⋯	Daylight	Document 1
101286	199901	CKB	⋯	Night	Document 2
101291	199902	LAX	⋯	Dawn	Document 3

In many applications, we need to analyze the text information in such a multidimensional text database with consideration of structured data in the standard dimensions. To support such analysis in a general way, it has been proposed in recent work to construct a Cube for text data [10, 15], which would enable an analyst to flexibly explore and analyze text cells, which are groups of text data corresponding to certain constraints on the standard dimensions.

*Department of Computer Science, University of Illinois at Urbana-Champaign
dzhang22@cs.uiuc.edu, czhai@cs.uiuc.edu, hanj@cs.uiuc.edu.

Many interesting online analysis tasks can be done on top of text cells. For example, an expert may be interested in the major anomalous events within a specific context. So she forms a query like (Time="1999", Location="LA") and tries to digest the content of all the narratives associated with these specified time and location values. A desirable system would return a *summary* of the content in the specific text cell (e.g. clusters of documents with major content words in each cluster or a small set of representative documents) so that the expert does not need to read all the documents. In another scenario, the expert may be interested in a particular topic within a text cell, e.g. the anomalous events related to "altitude deviation" at "LA" during "1999". In this case, a set of documents, generated as a summary for a text cell *given a query topic*, should ideally be both relevant to the topic and representative in covering most content of the text data in the cell (many duplicates in selected documents will cause the summary cover only partial content of a text cell). Furthermore, the expert may also be interested in *comparing* the content of multiple cells, e.g. a group of cells with different locations, and it would be desirable for the system to generate a comparison of the content covered by all these returned cells to reveal some common topics discussed within these cells and the different coverage of these common topics in each cell. Similar application scenarios can also be found in many other domains such as product review analysis, IT service ticket investigation, and disease symptom diagnosis.

Table 2. An Example of a *MiTexCube*

Cell	Doc ID	Content	Micro-Text-Clusters
(Time=1999, Location=TX)	d_1	... due to stronger than forecasted winds and weather going ...	(weather 2.5, wind 1.2, ...), 3
	d_2	... I think that the weather, headwinds, shrinking dewpoint/temperature contributed to the fuel emergency ...	
	d_3	... After an hour, the weather had not much improved. We were in the clear for a bit and then hit another cloud bank ...	
	d_4	... so that if we saw the ARPT, we could land ...	(land 2.1, rule 0.9, ...), 2
	d_5	... we were in class G and the IFR rules tell us to land ...	

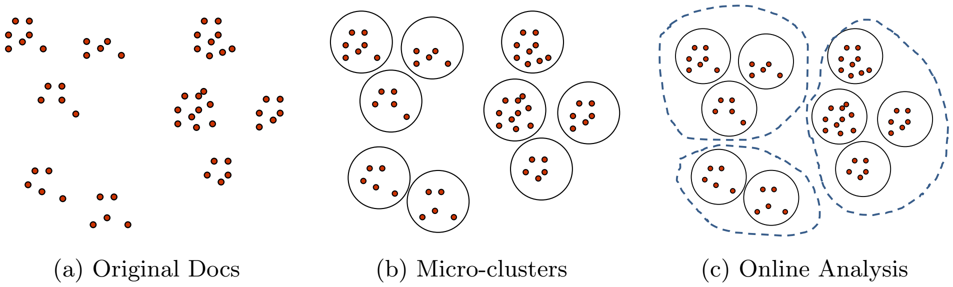
Since all these analysis tasks need to be done efficiently online, how to develop a *general* infrastructure to support all these tasks efficiently is a very interesting and challenging research question. Intuitively, we want to do as much offline pre-computing as possible to minimize the cost of online computation. However, there are two major technical challenges in implementing this general idea: (1) Many analysis tasks cannot be pre-specified in advance, making it impossible to pre-compute all the answers or even partial answers. For example, query-specific summarization can only be done after seeing the query, thus a naive solution of computing and storing summaries of all the cells offline is simply not feasible. Indeed, it is a significant challenge to factor out the computation that can be done offline. (2) Different analysis tasks need different computations (e.g., summarization and topic comparison have different needs). It is unclear how to provide a *general* support for many such tasks to enable efficient online processing.

One possible solution to the two challenges is to build a global Clustering Feature (CF) tree as proposed in [16]. In this approach, all the documents in the multidimensional text database can be first clustered into a global CF tree offline. Then, online analysis can take advantage of the clusters stored in the CF tree to reduce the computational cost. However, such a global CF tree is not suitable for an OLAP scenario, because in an OLAP text analysis task we mainly focus on local contents of a text cube. When we change the context and do text analysis in different text cells, the rigid global clustering structure cannot serve well in various local cells, since the clustering results of documents in a local text cell could be very different from their clustering results in a global CF tree. For example, if we cluster all the reviews in a commercial text database, the global CF tree may cluster reviews based on different brands of products. But when we do OLAP analysis in a text cell of a certain location, the reviews within that text cell may be clustered according to different time periods. Similarly, if we do OLAP analysis in a text cell of a certain time period, the reviews may also be clustered according to different locations. So a global clustering structure based on brands is not suitable for analysis in different local text cells.

The recent work on Text Cube [10] proposed methods for analyzing a text cube by materializing each text cell with vectors of documents. This approach can support several different analysis tasks, but it does not scale up well; indeed even a simple clustering analysis of the documents within one text cell is still expensive, especially when the number of documents is large.

In this paper, we propose a new general infrastructure called *MicroTextCluster Cube* to organize text content in a multi-dimensional text database so as to support a variety of online text analysis tasks efficiently. To solve the two major challenges above, our key idea is to represent text contents of each **local cell** in a "compressed" way which can retain the essential semantic information in text, so that online operations can be supported efficiently by performing them on the compressed representation rather than the original representation.

Specifically, we cluster documents in each cell into *micro-clusters* which serve as a compact, though coarse, representation of the content in the cell. The set of documents in a micro-cluster can be regarded as a big "pseudo-document" with a compact representation. Since the number of micro-clusters in each cell is usually much smaller than the number of individual documents, it allows us to dynamically analyze any text cell (*e.g.*, clustering documents in a text cell) much more quickly based on the micro-clusters in the cell. Intuitively, the online computation effort is reduced substantially by offline micro-clustering of similar documents, as shown in Figure 1, where we see that online clustering can be done based on micro-clusters instead of the original documents. Since a common characteristic in many analysis tasks is that they focus more on the characteristics of groups of documents rather than the concrete content of each individual document, the micro-cluster model essentially captures and leverages this kind of redundant information to achieve a concise representation that enables many online analysis tasks to be done efficiently.



(a) Original Docs (b) Micro-clusters (c) Online Analysis

FIGURE 1. Illustration of micro-clusters and their uses for summarization.

We materialize a *MiTexCube* with a progressive strategy, which aims at both saving the disk cost of a *MiTexCube* and supporting efficient analysis of a large set of documents in a high level or large text cell with flexible tradeoff between efficiency and quality of analysis. Basically, one cell is materialized with micro-clusters only when the number of micro-clusters aggregated from its sub cells is too large to perform efficient online operation. In that case, the cell is materialized by re-clustering those micro-clusters in its sub cells into a small set of larger micro-clusters. During online analysis, we can either use the micro-clusters within the target cell or we can use more finer granularity micro-clusters aggregated from its sub cells if time cost is affordable. In an extreme case, we can also use single-document micro-clusters as the analysis units. Therefore, our approach makes it possible to control the efficiency-quality tradeoff through adjusting the resolutions of micro-clusters, and accommodate the different needs of different analysis tasks.

We represent each micro-cluster by a centroid vector of weighted terms, associated with certain statistics such as the size of the micro-cluster (*i.e.*, the number of documents inside the micro-cluster). Note that how to represent micro-clusters and how to form micro-clusters offline is quite flexible, as long as they can effectively compress the content of text cells in a reasonable way. In our paper, we use weighted term vectors to represent micro-clusters and use k-means algorithm to form micro-clusters.

As a general infrastructure for representing text information in text cells in a compressed way, the micro-cluster text cube can potentially support many online analysis tasks efficiently. As case studies, in this paper, we propose methods to leverage *MiTexCube* to support three common analysis tasks: query-independent summarization, query-dependent summarization, and comparative analysis of text cells. We evaluate the proposed model and methods with the NASA:ASRS (Aviation Safety Report System) database. Experimental results show that the proposed cube structure can efficiently support summarization and comparative analysis of text cells, outperforming baseline methods that directly work on the documents in each cell without using micro-clusters, and it enables flexible tradeoff between efficiency and quality of analysis.

In sum, the contributions of this paper include: (1) We proposed a general novel cube structure that can support a variety of online text analysis tasks efficiently. (2) We proposed a materialization algorithm for constructing a *MiTexCube* to support flexible time-quality tradeoff for online applications and also save the disk cost of the cube. (3) We proposed methods for three applications, i.e. query-independent summarization, query-dependent summarization, and multi-cell comparison, which are customized towards leveraging the *MiTexCube* infrastructure.

2. Related Work

Online Analytical Processing (OLAP) [3, 5, 8] has found widespread applications in multiple domains [7, 11, 14]. Handling text in OLAP has recently drawn much attention (*e.g.*, [6, 2, 13, 10, 15]). Both [6] and [2] are systems related to text OLAP. They both use classification methods to classify documents into categories. Then each document is attached with a class label, and such category labels would allow users to drill down or roll up along the category dimension, achieving OLAP on text. In [6], the system also uses an optimized variant of k-means algorithm to do online clustering of documents. The main difference between our work and their work is that we aim at providing an infrastructure so that not only online clustering but also other analysis can be done more efficiently.

In [13], the authors try to combine keyword search with OLAP aggregation in order to efficiently analyze and explore large amounts of content. Their method is to first retrieve relevant documents from the database according to a query. Then they use both metadata and extracted phrases as dimensions to construct a data cube online, so that users can explore the results in an OLAP manner. Our work is different from theirs in that we first construct a text cube based on multidimensional text databases *offline*, and then provide an infrastructure for efficient text data analysis when doing OLAP in such a text cube.

Two new models related with text OLAP have been proposed in [10] and [15]. In [10], a text cube is partially materialized with two information retrieval-related measures, namely term frequency and inverted index, and a term hierarchy is also built in order to support OLAP operations. In their model, when a query arrives, the cube will return a set of relevant keywords in the result cell. The major difference between our work and their model is that we introduce micro-clusters into the cube, which materializes a cube with offline computed micro-clusters of documents rather than a vector of term frequency for each document. By grouping relatively similar documents together, the micro-cluster based infrastructure substantially reduces the online computation cost. Compared with text cube, our experimental results show that various kinds of online analysis of text cells become much more efficient with *MiTexCube*. In the topic cube proposed in [15], a hierarchical topic tree is predefined and a cell is materialized with statistics of topics mined from the documents in the cell. The purpose of a topic cube is to allow users to analyze predefined topics in text with different granularities. Our work differs from this work in that we are not restricted to predefined topics, and different analysis based on micro-clusters can adapt to arbitrary cells.

The materialization strategy used for materializing the *MiTexCube* is inspired by the BIRCH algorithm described in [16], but the purpose of using micro-clusters is quite different. In our work, we build micro-clusters in an OLAP environment, and the micro-clusters are used as coarse representations of the content of each local text cell. We do not need to maintain a global Clustering Feature tree as BIRCH, which is designed for incrementally and dynamically clustering incoming multi-dimensional metric data points.

3. MicroTextCluster Cube

The main idea of *MiTexCube* is to speed up online analysis of text cells by doing as much preprocessing as possible during offline stage. Specifically, we preprocess the documents by generating a good number of micro-clusters to "compress" similar documents. These micro-clusters are materialized and stored in *selected* text cells. Since these micro-clusters can roughly represent the original documents, in the online stage, we can mostly work on the micro-clusters to carry out analysis of text cells quickly.

3.1. **Definition of MiTexCube.**

Conceptually, *MiTexCube* extends a simple model, Document Cube. We thus first introduce the concept of document cube defined on a multidimensional text database.

Definition 3.1. Document Cube: A document cube is a data cube built based on the standard dimensions of a multidimensional text database. The measure stored in each cell is a document set which is the union of the documents (records in the database) aggregated from its subcells.

In general, a multidimensional text database is made of two parts: *standard fields* and a *text field.* The standard fields correspond to the attributes in a structured database (e.g., time, location) and can be viewed as the *context* of the associated text documents. Thus conceptually, the Document Cube allows us to naturally partition all the documents in the text field according to the combinations of values in the standard fields. Unfortunately, it is not feasible to store all the document lists in cells. For example, in the apex cuboid, we need to store all the documents in its document list, which would be too expensive space-wise. Thus the measure in a document cube is only a "conceptual measure"; in practice, when a user inputs a query, a document cube would use the value(s) specified on the standard fields to fetch all the matching records in the database and return the union of the corresponding documents as the measure of the cell.

MiTexCube essentially extends Document Cube by storing an additional measure that captures all the micro-clusters in a cell.

Definition 3.2. MiTexCluster: A micro text cluster (or MiTexCluster) is a coherent cluster of text documents that serves as a compressed representation of document content. These clusters are called micro-clusters because compared with the size of the corresponding cell that they represent, their sizes are relatively small, which ensures that the micro-clusters serve well as an approximation of the content in a cell for the purpose of analysis.

Definition 3.3. MiTexCube: A *MiTexCube* is a data model that extends a document cube to support efficient online analysis of text cells. Two kinds of measures are stored in cells of a *MiTexCube.* One is a document set aggregated from the base cells, which is the same as in a document cube. The other is either the statistics of a set of micro-clusters or a set of subcells from which the documents in the current cell can be efficiently computed based on aggregation.

One important thing to be noted in Definition 3.3 is that information about micro-clusters (*i.e.*, "content measures") is stored in two different ways. We define the *MiTexCube* in this way in order to save the disk storage as much as possible. To better explain the idea behind this, we use an example to show the correspondence between cells and their measures in Table 3. From this table, we can see that there are mainly two types of cells. One is materialized with concrete micro-clusters, and we call this type of cells *concrete cells* (type 1). Examples of concrete cells are C_{71} and C_{100}. In

TABLE 3. An example of the materialization of a *MiTexCube*

Cell ID	Type	Measure	Size
C_1	0	$\{d_1\}$	1
C_2	0	$\{d_2\}$	1
...	...	...	...
C_{70}	0	$\{C_1, C_2, \ldots, C_{10}\}$	10
C_{71}	1	$\{mean_1, 20\} \ldots \{mean_5, 30\}$	5
...	...	...	...
C_{99}	0	$\{C_{70}, C_{71}, \ldots, C_{76}\}$	56
C_{100}	1	$\{mean_1, 35\} \ldots \{mean_5, 32\}$	5
...	...	...	...

their measures, we store five micro-clusters for each of them. Each micro-cluster contains its mean vector and the size of the cluster. In our study, each document is represented by a vector of weighted terms, and the weight for each term is the TF-IDF value of this term within the document [12]:

$$\vec{d} = (c_d(w_1) * idf_{w_1}, c_d(w_2) * idf_{w_2}, \ldots, c_d(w_V) * idf_{w_V})$$

where $c_d(w_i)$ is the term frequency of word w_i in document d and idf_{w_i} is the inverse document frequency (IDF) of word w_i in the whole document set in the database. The mean vector of a micro-cluster is also a vector of weighted terms, and the weight for each term is the average weight for this term over all the documents that belong to this micro-cluster:

$$mean(mc_i) = \frac{1}{|mc_i|} \sum_{d \in mc_i} \vec{d}.$$

The other type of cells is materialized with a list of subcells, from which we can easily aggregate the micro-clusters in these subcells to form a set of micro-clusters for the current cell at the time of online processing. We call this type of cells *non-concrete cells* (type 0). For example, C_{99} is a non-concrete cell. Its measure contains a set of subcells, *i.e.*, $\{C_{70}, C_{71}, \ldots, C_{76}\}$. If we need to cluster the documents in cell C_{99}, we would fetch the micro-clusters contained in $\{C_{70}, C_{71}, \ldots, C_{76}\}$, and use them for clustering. In general, in order to save disk space, we would choose to not materialize a cell such as C_{99} as long as we can efficiently carry out analysis based on the micro-clusters contained in its subcells. However, had it been too expensive to do online analysis of the micro-clusters in $\{C_{70}, C_{71}, \ldots, C_{76}\}$, we would have further grouped these micro-clusters into larger micro-clusters and store them in the cell C_{99}, which would make it a concrete cell rather than a non-concrete cell.

In practice, storing the complete cell list in a non-concrete cell is still costly. Thus, we use a dimension and its level to indicate which set of subcells we should use for aggregation. For example, in Table 4, a cell (ID="laptop", Time="*", Location="*") can be either aggregated from subcells like {ID="laptop", Time="1st Quarter", Location="*"} or from subcells like {ID="laptop", Time="*", Location="TX"}. So we use "{Time, Quarter Level}" or "{Place, State Level}" as a compact representation of the corresponding list of subcells. In the next section, we will discuss the criteria for choosing the dimension for aggregation.

The star schema of a *MiTexCube* is shown in Figure 2. In the schema, if we ignore Measure 2, it would become the star schema of a document cube. As we discussed above, Measure 1 in this schema is just a conceptual measure.

3.2. **Progressive Materialization.** Materialization of a *MiTexCube*means we need to precompute the micro-clusters offline and store the micro-clusters in the *MiTexCube.* A good materialization is important because (1) with sufficient materialization, the online analysis can be done efficiently; (2) the overhead of materialization should be reasonable. There are two important parameters to be set for materializing a *MiTexCube.* One is the total number of micro-clusters K in each cell. A larger

Table 4. An example of subcell selection

Cell ID	Subcell Set	Selection
(laptop, *, *)	(laptop, 1st Quarter, *)	{Time, Quarter Level}
	(laptop, 2nd Quarter, *)	
	(laptop, 3rd Quarter, *)	
	(laptop, 4th Quarter, *)	
	(laptop, Jan., *)	{Time, Month Level}
	(laptop, Feb., *)	
	. . .	
	(laptop, Dec., *)	
	(laptop, *, CA)	{Place, State Level}
	(laptop, *, TX)	
	. . .	
	(laptop, *, WA)	

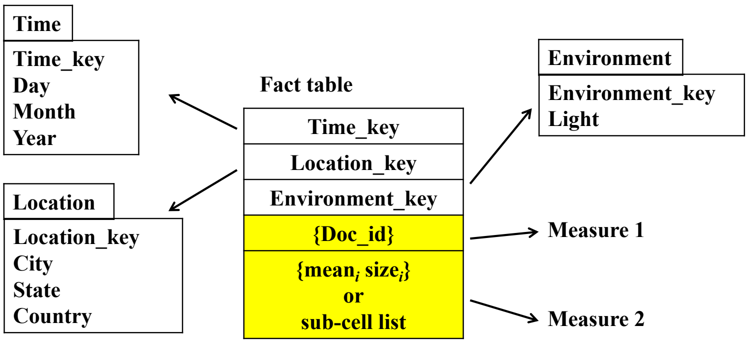


Figure 2. Star Schema of a *MiTexCube*

K will result in finer granularity of micro-clusters, so the result of online processing like clustering will be closer to that of clustering documents directly at the price of slower online processing. On the other hand, a smaller K will result in larger micro-clusters of documents in each cell, which can speed up online processing at the price of achieving coarser approximation of the content and not being able to summarize a cell at a finer granularity level of topics. Thus there is an inherent tradeoff here between approximation accuracy and time efficiency of online summarization and this tradeoff is controlled by the parameter K, which can be empirically set according to specific application needs.

The other important parameter is the total number of micro-clusters M that we can deal with efficiently for online processing. This parameter controls the tradeoff between time efficiency and space overhead. If M is small, then the number of cells needed to be materialized will be large, thus more disk storage would be needed. On the other hand, if M is large, then the online processing will be more time consuming, but we would be able to save more space since the number of cells to be materialized would be smaller. Thus M provides a flexible way to control this tradeoff. For example, we may optimize the value of M based on whether we can efficiently cluster M micro-clusters online.

Once these two parameters are set, our algorithm for materializing a *MiTexCube* works in a bottom-up manner to progressively process each cell. The process is illustrated in Figure 3.

Specifically, we would start materializing the cube from the base cells, each of which contains only one document. As we aggregate a set of base cells into the next level of cube (*i.e.*, cuboid ABCD, ABDE, *etc.*.), we would test the number of documents in each cell. If the number is larger than the threshold M, we would group this set of documents into K micro-clusters, and store these micro-clusters as measures, as illustrated in cell $(a_1, b_1, c_1, d_1, *)$. Based on these micro-clusters, we

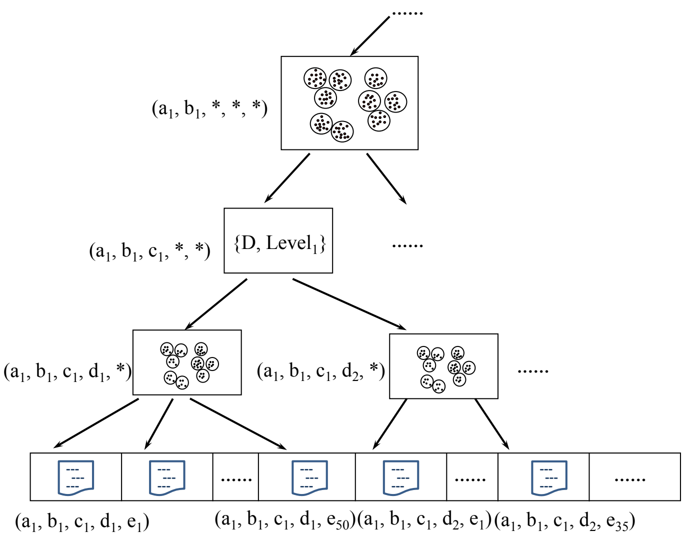

......
$(a_1, b_1, *, *, *)$
$(a_1, b_1, c_1, *, *)$
$\{D, Level_1\}$
......
$(a_1, b_1, c_1, d_1, *)$
$(a_1, b_1, c_1, d_2, *)$
......
......
......
......
$(a_1, b_1, c_1, d_1, e_1)$
$(a_1, b_1, c_1, d_1, e_{50})(a_1, b_1, c_1, d_2, e_1)(a_1, b_1, c_1, d_2, e_{35})$

Figure 3. Materialization of a *MiTexCube*

can further aggregate into the next level of the cube (cuboid ABC, ABD, *etc.*). If the number of micro-clusters aggregated into one cell is no larger than the threshold M, we would only need to store a sub-cell list represented by the aggregation dimension and the level of this dimension, from which we will be able to aggregate the micro-clusters from the subcells, *e.g.*, cell $(a_1, b_1, c_1, *, *)$, thus saving the space needed to store the complete list of subcells. As shown in Table 4 and discussed in the previous section, there are several possibilities to aggregate subcells into a super cell. In our algorithm, we choose to store the dimension from which the subcells of the super cell would give the least number of micro-clusters; the rationale is to delay the need for re-clustering as much as we can, thus also saving more space. As we reach the next level of cube (*i.e.*, cuboid AB, BC, *etc.*), we first calculate the number of micro-clusters in each cell. For example, in cell $(a_1, b_1, *, *, *)$, the micro-clusters are aggregated from cell $(a_1, b_1, c_1, *, *)$, *etc.*. Since cell $(a_1, b_1, c_1, *, *)$ is non-concrete, the micro-clusters aggregated from this cell are actually from its own subcells. Assuming that, at this time, the number of micro-clusters in cell $(a_1, b_1, *, *, *)$ is larger than the threshold M, we thus group these small micro-clusters into K larger micro-clusters, as shown in the figure.

In general, to group micro-clusters from subcells into larger micro-clusters, we may use any clustering algorithm. In our experiments, we use a k-means based algorithm to group those micro-clusters based on their means. One advantage of k-means is that we can stop at any iteration to obtain clustering results, and thus can flexibly trade off time with quality of clusters. For each small micro-cluster we have the mean and size of it. Therefore, we can use these statistics to calculate the statistics of the new micro-clusters. For example, suppose we group micro-clusters $mc_1, mc_2, ..., mc_l$ into a bigger micro-cluster. The mean of the new micro-cluster can be computed as

$$\frac{\sum_{i=1}^{l} mean(mc_i) * size(mc_i)}{\sum_{i=1}^{l} size(mc_i)}, \tag{1}$$

and the size of the new micro-cluster is

$$\sum_{i=1}^{l} size(mc_i) \tag{2}$$

The pseudo code of the materialization algorithm using k-means clustering is given in Figure 4. Here, variable min_{mc} is used to store the minimum number of micro-clusters aggregated into the current cell. Variables $min_{dimension}$ and min_{level} are used to store the corresponding dimension and level. Variable num_{mc} on line 8 represents the number of micro-clusters aggregated from subcells in the current dimension and level.

Figure 4. Pseudo Code for Materialization

```
Input: A multidimensional text database with $n$ standard fields and one text field
Output: Materialized MiTexCube
Algorithm:
1 From base cuboid to Apex cuboid
2   For each cell in current cuboid
3     $min_{mc} = \infty$;
4     $min_{dimension}$ = null;
5     $min_{level}$ = null;
6     For each aggregation dimension in current cell
7       For each level in this dimension
8         Calculate $num_{mc}$ in subcells aggregated from current dimension and level
9         if($num_{mc} < min_{mc}$)
10            $min_{mc} = num_{mc}$;
11            $min_{dimension}$ = current dimension;
12            $min_{level}$ = current level;
          end of if
        end of for
      end of for
13    if($min_{mc} > M$ )
14      regroup these micro-clusters in $K$ larger micro-clusters, and store the mean and size of each new micro-cluster
15    else
16      store the $min_{dimension}$ and $min_{level}$ as measures
```

4. Online Analysis of Text Cells

After a *MiTexCube* is materialized, we can carry out various kinds of online analysis based on this infrastructure. In this section, we discuss three representative online analysis tasks.

4.1. Standard (Neutral) Cell Summarization.

Standard (*i.e.*, topic-neutral) cell summarization means to give analysts an overview of the content in any given text cell by grouping all the documents in that text cell into P different clusters, where P is the desired number of clusters specified by an analyst. Based on the *MiTexCube* model, one can efficiently generate such a standard cell summary by clustering the already formed micro-clusters instead of clustering all the documents from scratch. Specifically, assume one cell has in total MC micro-clusters ($MC > P$). We can use the mean vector of each micro-cluster as a data point (as if it were a document vector) and use the k-means algorithm to partition them into P clusters. When we cluster several micro-clusters into one big cluster, we can use Eq. (1) and Eq. (2) to update the mean and size of this big cluster. Thus algorithm-wise, our method for standard cell summarization is similar to re-clustering in the materialization algorithm except that we now generate fewer macro-clusters for the purpose of online analysis of a text cell's content.

Since the k-means method is an iterative method, the time complexity of the baseline method of clustering all the documents from scratch (which we denote by *GS-Base*) is $O(D * P * n)$, where D is the total number of documents to be clustered, n is the total number of iterations. With *MiTexCube*, the time complexity of our method (denoted by *GS-MC*) is $O(MC * P * m)$, where MC is the number of micro-clusters and m is the number of iterations. When $MC \ll D$, we can expect that *GS-MC* should be much faster than *GS-Base* (the number of iterations m is comparable with n). While it is inevitable that *GS-MC* would be inferior to *GS-Base* in clustering quality, we can expect the sacrifice of quality to be insignificant since documents in a micro-cluster are generally similar to each other in content. Indeed, since we can adjust the size of a micro-cluster (thus also the number of micro-clusters), *MiTexCube* enables flexible efficiency-quality tradeoff.

4.2. **Query-Specific Cell Summarization.** The purpose of query-specific cell summarization is to customize a summary based on the topic preference that a user may have. Specifically, given a set of documents in one text cell as well as a topic keyword query q, the task of query-specific summarization is to generate a summary with P documents selected from the cell that are both representative of the cell and relevant to the query, where P is a number specified by a user to indicate the desired number of documents in the summary. This is different from a traditional information retrieval task, which only considers the relevance of documents to a query. For a summarization task, we also want the selected documents cover well the major content in the cell.

With *MiTexCube*, we can leverage the available micro-clusters to optimize the coverage of the documents in the cell by forcing the summary to include documents distributed over all the distinct micro-clusters. Intuitively, micro-clusters tell us where the redundancy is, because documents within the same micro-cluster are believed to be similar. Specifically, suppose there are K micro-clusters in a cell, given a topic query q we first rank all the documents into a candidate list based on their relevance to the query. Then, in the first round, we select documents from the most relevant one, and if one document is selected, all the documents in the same micro-cluster will be removed from the candidate list and not be considered for selection. The next document to be considered is the most relevant document remained in the list. So this ensure that we select relevant documents distributed over all the micro-clusters. If we need more representative documents (i.e. $P > K$), we just get back all those non-selected documents and do another round of selection.

An *indirect* way to generate a query-specific summary for a text cell is to use a greedy algorithm called Maximal Marginal Relevance (MMR) [4] to avoid redundancy in the selected documents. MMR reranks a list of documents by using the following formula to select the next document to reduce redundancy in the selected documents:

$$argmax_{D_i \in R \setminus S} \left[\lambda Sim_1(D_i, Q) - (1-\lambda) \max_{D_j \in S} Sim_2(D_i, D_j) \right] \tag{3}$$

Here, R is the candidate document set, S is the current selected document set, D_i is a candidate document to be considered as the next selected document, Q is a user specified query, Sim_1 is a function used to measure the similarity between a query and a document, and Sim_2 measures the similarity between two documents.

Compared with the *MiTexCube*-based method (denoted as *QS-MC*), the MMR approach (denoted as *QS-Base*) is less efficient because it requires computation of pair-wise similarity for potentially many document pairs on the fly; besides, the MMR approach may not achieve representativeness well because avoiding redundancy does not always lead to representative topics, while the *MiTexCube* method achieves representativeness more *directly* through the structure based on micro-clusters.

4.3. **Common Topic Comparison.** Another analysis task is to compare multiple text cells to reveal the difference of their coverage on common topics. The standard cell summarization of the text cells cannot easily quantify the coverage of a common topic in different cells, because the result clusters may not be comparable across different cells. A better way to support common topic comparison is to pool the text documents in all the text cells to be compared and cluster them into P clusters, which can then be assumed to be P common topics covered in these cells and serve as a common basis for comparison of different cells. With these P topic clusters as a basis, we can measure the content of each cell by a vector of weights corresponding to the numbers or percentages of documents in the cell that belong to each of the P clusters. Intuitively, such a weight vector (in P-dimensional space) indicates the coverage of each common topic in the corresponding cell, thus comparing these weight vectors across cells can easily reveal which cell covers which topic more and generate trends of topic coverage in any standard dimension with ordinal variables (e.g., location or time).

Once again, *MiTexCube* can speed up this clustering process as we only need to cluster all the micro-clusters in these cells instead of all the documents in them. Without *MiTexCube*, we would have to pool all the documents together and then cluster them from scratch into P common topics. As discussed earlier in the case of standard cell summarization, *MiTexCube* can be potentially much faster than this baseline approach and it also naturally allows us to flexibly take a tradeoff between efficiency and clustering quality.

5. Experimental Results

In this section, we will evaluate how well our *MiTexCube* model supports multiple representative analysis tasks.

5.1. **Data Sets.** We used the ASRS database [1] for our experiments. We downloaded and extracted two years (1998, 1999) of the data from the database, giving us a total of 4073 records for our experiments. We selected 7 dimensions from the database to construct our *MiTexCube*, and the number of distinct values in each dimension is summarized in Table 5.

Table 5. Number of distinct values in each dimension

State	Flight Condition	Light	Operator	FAR	Flight Phase	Affiliation
3	5	2	8	8	32	10

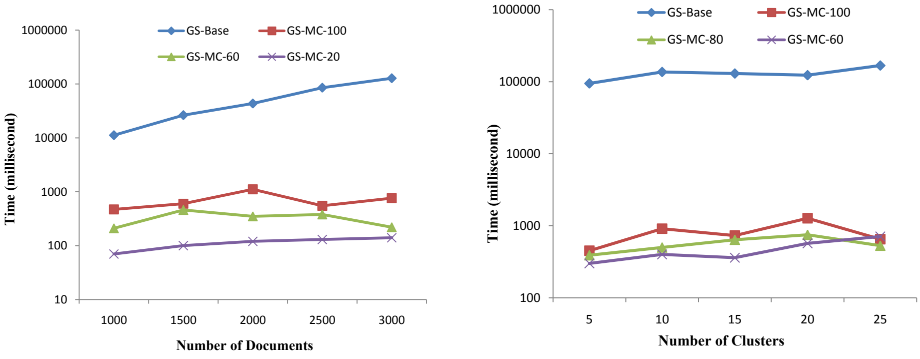
5.2. **Evaluation of Representative Analysis Tasks.**

5.2.1. *Standard Cell Summarization.* We first look at standard cell summarization and compare our *GS-MC* method, which is based on *MiTexCube*, with the baseline method *GS-Base*, which works directly on the documents in a cell, in terms of both efficiency and effectiveness. We vary the parameter K to generate different settings of *GS-MC* with different numbers of micro-clusters K in each cell, which will be denoted by a suffix indicating the value of K. For example, *GS-MC-100* refers to the setting of $K = 100$.

Efficiency: In Figure 5(a), we compare the speed of clustering documents or micro-clusters when we vary the size of a cell from 1,000 to 3,000. The target number of clusters in this experiment is 10, which means that we use 10 clusters to summarize the content of documents in each cell. From Figure 5, we can see that for all the three different settings (i.e., K=20, 60, and 100), *GS-MC* is much faster than the *GS-Base* method, and for some cells, *GS-MC* is 100 times faster than *GS-Base*. Moreover, as the number of documents increases, the *GS-Base* method slows down dramatically, but the time cost of *GS-MC* does not increase much. In general, the larger the number of micro-clusters K is, the slower the *GS-MC* method is; this is the price we pay for obtaining a finer granularity representation of content, which gives us better approximation of content.

In Figure 5(b), we further compare the two methods by varying the number of targeted clusters. Here, we also test three different settings for *GS-MC*, corresponding to setting K to 60, 80, and 100, respectively. In this experiment, we use a cell which has 2000 documents in it. From the figure, we can make the same conclusion as in Figure 5(a), i.e. *GS-MC* is much faster than *GS-Base* in all the settings.

Quality of clustering: Since there is always a tradeoff between the efficiency and accuracy, we expect our method *GS-MC* to have inferior quality to the baseline method *GS-Base*, and our main goal is to see how well *GS-MC* can support flexible tradeoff between efficiency and quality. (Indeed, we may view *GS-Base* as a special case of our *GS-MC* when we have each document as a micro-cluster.) Table 6 shows the comparison result in a text cell with 2000 documents, and the number of target cluster P is set to 10 and 5. For each method, we compute its clustering quality as well as its time cost, and the numbers are the average result of 10 runs for each method on each test case. Here, the quality of a clustering result is the sum of cosine similarity between each document vector



(a) (b)

Figure 5. Efficiency Comparison between *GS-Base* and *GS-MC*

and its cluster's mean vector, which intuitively captures the coherence of a cluster, and the larger the better.

We tried two strategies to improve the quality of the clustering result.

1. Increasing the number of micro-clusters: During online analysis, when we need finer granularity of micro-clusters to analyze a text cell, we can always go down to its sub cells, which result in a set of larger number of micro-clusters. In our experiment, we test our method with different number of micro-clusters (i.e. K80, K500, K1000), and the result in the table shows that as the number of micro-clusters increase we can get improvement on the quality by sacrificing some time.

2. Additional iterations of document based clustering: After running *GS-MC*, we may also further improve the quality of clustering by starting from the results of *GS-MC* and running additional iterations of k-means on document vectors, as shown in the last six rows of the table where we show the results of running one additional iteration and two additional iterations. For example, K80 + 1 means we do one iteration of document-based clustering after clustering all the 80 micro-clusters into P target clusters, using mean vectors of the result P clusters as the starting point. The result also shows that by additional iterations of document vector based clustering, the quality of clusters can be improved.

Overall, although the baseline method gets very high quality of cluster, the time cost of it is also the highest. With the help of *MiTexCube*, *GS-MC* can indeed support flexible tradeoff between efficiency and quality of clustering.

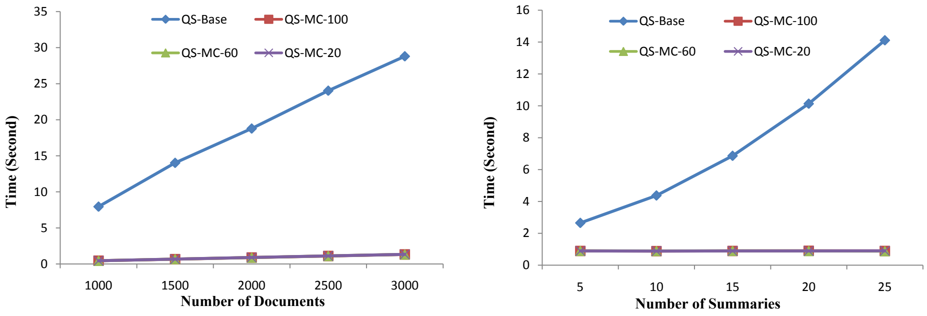
Table 6. Quality Comparison for Standard Cell Summarization

	P= 10		P= 5	
Method	Quality	Time	Quality	Time
Baseline	491.84	52.36	444.09	47.38
K80	445.59	0.57	408.02	0.50
K500	456.22	6.55	420.82	6.31
K1000	469.87	17.83	430.60	14.86
K80 + 1	463.88	3.53	422.35	2.77
K500 + 1	473.98	9.78	432.84	8.71
K1000 + 1	482.36	21.15	437.90	17.29
K80 + 2	468.11	6.46	427.01	4.98
K500 + 2	477.12	12.97	434.19	11.03
K1000 + 2	484.30	24.42	438.48	19.69

5.2.2. *Query-Specific Cell Summarization.* We now look at query-specific cell summarization, and here we compare our method *QS-MC* with the MMR baseline method *QS-Base* again in terms of both efficiency and quality. We use a query ("flight", "system") to test the performance of the two methods (similar conclusions can be drawn with other queries). To calculate the similarity between a document and a query, we use the KL-divergency retrieval model [9].

Efficiency: Figure 6(a) and Figure 6(b) show the experimental results for different cell sizes and different numbers of summary documents, respectively. From these figures, we can see that the time cost of *QS-Base* increases linearly as we increase either the total number of documents in a cell or the number of target summary documents. If we look at Eq. 3, we can find out that in MMR, whenever a top ranked document is selected, it would need to update the score of all the rest documents, and this is the reason why the time cost of *QS-Base* increases linearly as shown in Figure 6.

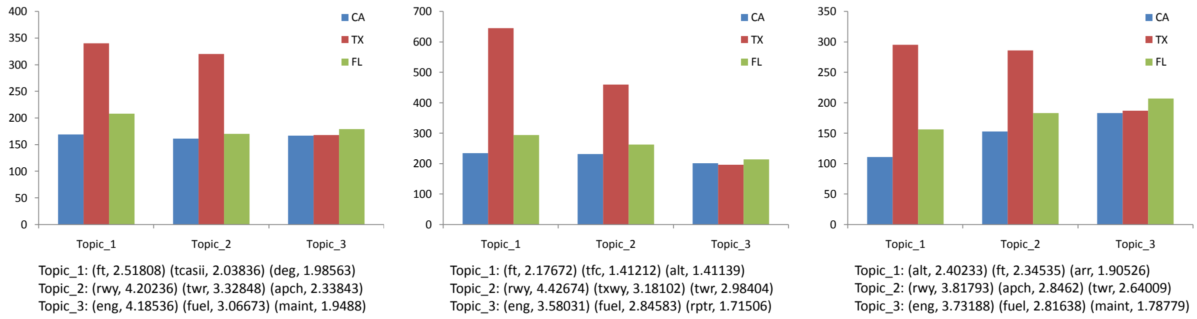
In contrast, the time cost of *QS-MC* only increases very little when the total number of documents increases, as shown in Figure 6(a), or when the number of target summary documents increases, as shown in Figure 6(b). Moreover, the performance of different settings of the number of micro-clusters K do not have very much difference, so that their curves overlap with each other. Actually, from previous section we can know that the time cost of *QS-MC* mainly depends on the process of document ranking, and almost independent of the number of target summary documents and the setting of the number of micro-clusters K. So overall, the *QS-MC* method is much faster than the *QS-Base* method.



(a) (b)

Figure 6. Efficiency Comparison between *QS-Base* and *QS-MC*

Quality: Table 7 shows the quality comparison result of one query when we retrieve 20 document as a summary based on two measures: (1) coverage and (2) relevance. The coverage is calculated using the following method: for each unselected document, we calculate the highest cosine similarity of this document with the selected 20 documents as its score, which intuitively captures how well this document is covered by the selected 20 documents. Then, we sum over the scores of all the unselected documents as the coverage. Relevance is the total similarity of all the selected document to a query. The top three rows are results of MMR (i.e., *QS-Base*), and the bottom three rows are result of our method (i.e., *QS-MC*), where K is the number of micro-clusters in the cell and λ is the weight parameter used in MMR. The total number of document is 2000. From these results, we can see that *QS-MC* consistently outperforms *QS-Base* in coverage due to better capturing the representative topics in the cell through micro-clusters, confirming our hypothesis that direct modeling topics through micro clusters is more effective for selecting documents representing the cell well than the indirect way through eliminating redundancy used in MMR. However, we also note that *QS-MC* has lower relevance than *QS-Base*, which indicates a tradeoff between relevance and coverage as well as a tradeoff between relevance and efficiency (as discussed earlier, *QS-Base* is much less efficient than *QS-MC*). Note that here again *MiTexCube* allows us to make flexible tradeoff between relevance and efficiency since as K get larger, we get better relevance.



(a) (b) (c)

Figure 7. Common Topic Comparison

Table 7. Quality Comparison for Topic-biased Cell Summarization

λ	0.2	0.4	0.6	0.8	1
relevance	-283.27	-282.278	-282.278	-282.218	-282.21
coverage	258.28	257.919	257.919	259.707	259.707

K	10	20	30	50	100
relevance	-284.86	-284.0996	-282.8749	-282.6589	-282.5442
coverage	264.3687	269.9924	271.238	264.84	267.6433

5.2.3. *Sample results of comparative analysis of text cells.* We use sample results to show the effectiveness of *MiTexCube*in the *Common Topic Comparison* task. We use the total 4071 documents within three cells for the comparison, which have different locations(states), namely CA, TX, and FL. The number of common topics to be compared is set to 10.

Figure 7(a) shows the comparison result based on document units[1]. Figure 7(b) is the result based on micro-clusters in which each cell has 100 micro-clusters inside, and Figure 7(c) is the result where each cell has 500 micro-clusters. The y-axis is the number of documents that belong to one topic within a cell. The three topics on the x-axis are the top three major topics within the 10 common topics. The top weighted terms are also listed under each graph. We can see that the two micro-cluster based methods got similar comparison result to the document unit based method. When the number of micro-clusters of each cell increases, the comparison results are much closer to the document unit based approach. For example, for the comparison of topic_3 over different states, Figure 7(c) is more accurate than Figure 7(b). In addition, compared with the K100 based approach, the K500 based approach has more similar top weighted terms to the document based approach. The time cost for these three methods are: 215.77, 18.09, and 62.35 seconds, which shows the advantage of micro-cluster based methods in terms of efficiency.

6. Conclusions and Future Work

In this paper, we proposed a novel cube called MicroTextCluster Cube (*MiTexCube*) to enable efficient online analysis of text cells in several applications. We propose a progressive materialization algorithm for this novel cube and methods to leverage *MiTexCube* for three analysis tasks, including standard cell summarization, query-specific cell summarization, and common topic coverage comparison. Experimental results on a real multidimensional text database show that applications based on the proposed materialized *MiTexCube* are more efficient than the baseline methods of direct analysis

[1] Abbreviations: (ft: Feet), (tcasii: Traffic Alert and Collision Avoidance System), (deg: Degree), (rwy: Runway), (twr: Tower), (apch: Approach), (eng: Engine), (maint: Maintenance), (tfc: Traffic), (alt: Altitude), (txwy, Taxiway), (rptr: Reporter)

based on document units in each cell, without sacrificing much quality of analysis. The proposed *MiTexCube* has several parameters to accommodate flexible tradeoffs between time and space as well as effectiveness and efficiency.

As for our future work, exploring how to leverage *MiTexCube* for tasks in other domains like production review analysis is very interesting. Also, the basic idea of using multi-resolution micro-clusters to achieve a compact semantic representation of data is general and can be used in other multidimensional text database analysis. We plan to further explore these directions in the future.

7. Acknowledgement

The work was supported in part by NASA NRA-NNH10ZDA001N, U.S. Air Force Office of Scientific Research MURI award FA9550-08-1-0265, U.S. National Science Foundation grants IIS-0905215 and CNS-1028381. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agencies.

References

[1] Aviation safety reporting system.
http://asrs.arc.nasa.gov/.

[2] Megaputer's polyanalyst.
http://www.megaputer.com/.

[3] S. Agarwal, R. Agrawal, P. Deshpande, A. Gupta, J. F. Naughton, R. Ramakrishnan, and S. Sarawagi. On the computation of multidimensional aggregates. In *VLDB*, pages 506–521, 1996.

[4] J. Carbonell and J. Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *SIGIR '98*, pages 335–336, New York, NY, USA, 1998. ACM.

[5] S. Chaudhuri and U. Dayal. An overview of data warehousing and olap technology. *SIGMOD Rec.*, 26(1):65–74, 1997.

[6] W. F. Cody, J. T. Kreulen, V. Krishna, and W. S. Spangler. The integration of business intelligence and knowledge management. *IBM Syst. J.*, 41(4):697–713, 2002.

[7] F. M. fei Jiang, J. Pei, and A. W. chee Fu. Ix-cubes: iceberg cubes for data warehousing and olap on xml data. In *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 905–908, New York, NY, USA, 2007. ACM.

[8] J. Gray, A. Bosworth, A. Layman, and H. Pirahesh. Data cube: A relational aggregation operator generalizing group-by, cross-tab, and sub-totals. *ICDE*, 00:152, 1996.

[9] J. Lafferty and C. Zhai. Document language models, query models, and risk minimization for information retrieval. In *SIGIR '01*, pages 111–119, New York, NY, USA, 2001. ACM.

[10] C. X. Lin, B. Ding, J. Han, F. Zhu, and B. Zhao. Text cube: Computing ir measures for multidimensional text database analysis. In *ICDM*, pages 905–910, 2008.

[11] E. Lo, B. Kao, W.-S. Ho, S. D. Lee, C. K. Chui, and D. W. Cheung. Olap on sequence data. In *SIGMOD '08: Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 649–660, New York, NY, USA, 2008. ACM.

[12] G. Salton and M. McGill. *Introduction to Modern Information Retrieval.* McGraw-Hill, 1983.

[13] A. Simitsis, A. Baid, Y. Sismanis, and B. Reinwald. Multidimensional content exploration. *Proc. VLDB Endow.*, 1(1):660–671, 2008.

[14] Y. Tian, R. A. Hankins, and J. M. Patel. Efficient aggregation for graph summarization. In *SIGMOD '08: Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 567–580, New York, NY, USA, 2008. ACM.

[15] D. Zhang, C. Zhai, and J. Han. Topic cube: Topic modeling for olap on multidimensional text databases. In *SDM*, 2009.

[16] T. Zhang, R. Ramakrishnan, and M. Livny. Birch: an efficient data clustering method for very large databases. *SIGMOD Rec.*, 25(2):103–114, 1996.

A STOCHASTIC METHODOLOGY FOR PROGNOSTICS UNDER TIME-VARYING ENVIRONMENTAL FUTURE PROFILES

LINKAN BIAN AND NAGI GEBRAEEL*

Abstract. We present a stochastic model of a sensor-based degradation signal for predicting, in real time, the residual lifetime of individual components subjected to a time-varying environment. We consider future environmental profiles that evolve in a deterministic manner. Unique to our model is the union of historical data with real time sensor-based data to update the degradation model and the residual life distribution (RLD) of the component within a Bayesian framework. The performance of our model is evaluated based on degradation signals from both numerical experiments and a case study using real bearing data. The results show that our approach provides more accurate estimates of the RLD, compared with benchmark models.

1. Introduction

Sensor technology with condition monitoring techniques has been widely used in monitoring critical engineering components in complex systems, such as complex wind turbine systems, aircraft components, smart structures, manufacturing equipment, and so on. The resulting real-time sensory data, known as degradation signals, are usually correlated with the underlying physical degradation process, which might be unobservable. The environmental or operating conditions may have a significant impact on the remaining lifetime of a component. For example, increasing the load and the speed of rotating machinery may accelerate the degradation of its components, such as roller bearings. The effects of time-varying environments and the uncertainties associated with components' degradation processes pose challenges on accurately forecasting the remaining life distributions of components. Therefore, it is important to develop a prognostic model that can incorporate the effect of environmental or operating conditions.

In this paper, we propose a stochastic model for the evolution of degradation signals from a fielded component, which operates in time-varying environments; and predict the residual useful life of the component in real time. This paper examines the effects of the severity of the current environment on the rate of degradation. Furthermore, our model is unique because it accounts for the reality that the transitions in environments may induce upward or downward jumps in the amplitude of the degradation signal, depending on the nature of the changes. We assume that the component operates in a time-varying environment that the environmental profile is deterministic and known (i.e., there is no uncertainty about how the environment transitions in the future). Unlike the traditional reliability approaches, our approach incorporates historical data of a population of similar components with real-time sensor data that updates the residual life distribution in real time.

The remainder of the paper is organized as follows. In Section 2, we review the extensive literature related to lifetime estimation that includes models with and without environmental effects. Section 3 describes the degradation models which assumes the environmental or operating conditions evolve dynamically, but in a deterministic manner. In Section 4, we describe a simulation study to compare the results of our model with other existing models in the current literature. In Section 5, we discuss a case study with the implementation of bearing data. Finally, in Section 6, we provide some concluding remarks and the future extensions of this paper.

*Milton H. Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332, nagi.gebraeel@isye.gatech.edu.

2. Relevant Literature

Residual life estimation for components operating in static environments has been studied extensively in the literature. We highlight a number of important contributions here [16, 15, 5]. For example, in [16], the authors developed Brownian motion models of the degradation process of testing units and used lifetime data to estimate the lifetime distribution of a population of units. Besides, [5] considered the utilization of real-time degradation signals for updating the residual life distributions (RLDs) of partially degraded components. The authors modeled a degradation signal as a Brownian motion process which leads to a closed-form results for the residual life distribution. None of the models described here consider the effect of the unit's operating environment.

The literature related to models for the lifetime distributions of components operating in dynamic environments can be partitioned into two groups. Papers in the first group utilize Cox's proportional hazard model (PHM). Due to its generality and flexibility, the PHM has been widely utilized to relate the hazard function to the environmental conditions. For example, [9] utilized Cox's PHM in optimizing condition-based maintenance. The authors modeled the environmental conditions as the time-varying covariates of the hazard function. Along this line, [2] considered an environmental process, which evolved as a Markov process. The authors applied an approximation technique to assess the distribution of failure time. The resulting expression of the failure time distribution was represented in a complex integral form. Computational issues associated with this problem were further investigated in [1] where the authors proposed a general numerical methods to approximate the failure time distribution. Other extensions and applications of PHMs can be found in [7, 12, 19].

The second group of literature focuses on modeling the degradation process or its manifestations. These processes are usually characterized by stochastic processes such as Brownian motion, general Markov processes and others. For example, [4] applied Brownian motion with a stress-dependent drift to accelerated life test experiments and developed the failure time distribution. [6] considered a similar problem with deterministic environmental profiles. Furthermore, [10] examined the problem wherein the state of environmental condition evolved as a Markov process. [11] extended the model in [10] to consider the addition of Poisson shocks, each of which induces a random amount of damage to the system. However, [10, 11] did not account for the possibility of shocks that may occur at environment transition epochs.

In contrast to research described here, this work is concerned with the modeling of the degradation signal of a component that is monitored in real time. By observing the real-time degradation signal and the environment, we update the residual life distribution of the component in real time. Therefore, the predicted residual life distribution depends on the prior information as well as the future environmental conditions. Furthermore, we explicitly include upward and downward jumps in the degradation signal stemming from transitions of the environment process, which has not been considered in previous works. Section 3 presents our degradation signal model for the case when the environment is time-varying but deterministic.

3. Degradation in a Deterministic and Dynamic Environment

In this section, we present the degradation model and a method for estimating the residual life distribution (RLD) of the component in real time via Bayesian updating. Here, we assume that the environment is time-varying but deterministic. We begin with an elucidation of the notation and a few preliminaries. For each $t \geq 0$, let $S(t)$ be the degradation signal at time t, and let $S(0)$ be the initial signal observation. We assume that a population of identical components begins with the same initial degradation signal. At any time $t \geq 0$, the component's environment can occupy one of the states in a set $S = \{1, 2, \ldots, m\}, m < \infty$. Deciding the appropriate number of environment states m, and a meaningful ordering of the states in S, are important aspects of our modeling framework discussed in the following subsection. Let $\psi : [0, \infty) \to S$ be an S-value deterministic and piecewise constant function so that $\psi(t)$ is the state of the environment at time t. That is, the environment visits the states in S in a deterministic way. Besides, we denote by $r(\psi(t))$ the component's rate of degradation at time t such that whenever $\psi(t) = j \in S$, the component degrades at rate $r(j)$. Finally, we account for the reality that in typical applications, the degradation signal exhibits jumps at environment transition epochs. To this end, we denote a mapping $J : S \to \mathbb{R}$ so that $J(\psi(v))$ is a function of the jump (either upward or downward) that occurs at time v. Specifically, for these models, the jump magnitude is a deterministic quantity that depends on the environment state just before and after the jump epoch. The mapping J can assume a variety of forms; however, in this research we assume that the jump magnitude is proportional to the current state of the environment.

With these definitions and notation, the model of the degradation signal is

$$S(t) = S(0) + \int_0^t r(\psi(v))dv + \int_0^t J(\psi(v)) + \gamma W(t), \tag{1}$$

where $\{W(t) : t \geq 0\}$ is a standard Brownian motion (BM) process, and $\gamma(\gamma > 0)$ is its diffusion parameter. That is, for each $t \geq 0$, $\gamma W(t) \sim N(0, \gamma^2 t)$, where $N(a, b)$ denotes a normal random variable with mean a and variance b. This term models degradation effects that cannot be attributed to the environment process. Figure 1 graphs a sample path of the degradation process $S(t)$ and illustrates the effect of the deterministic environment on its evolution.

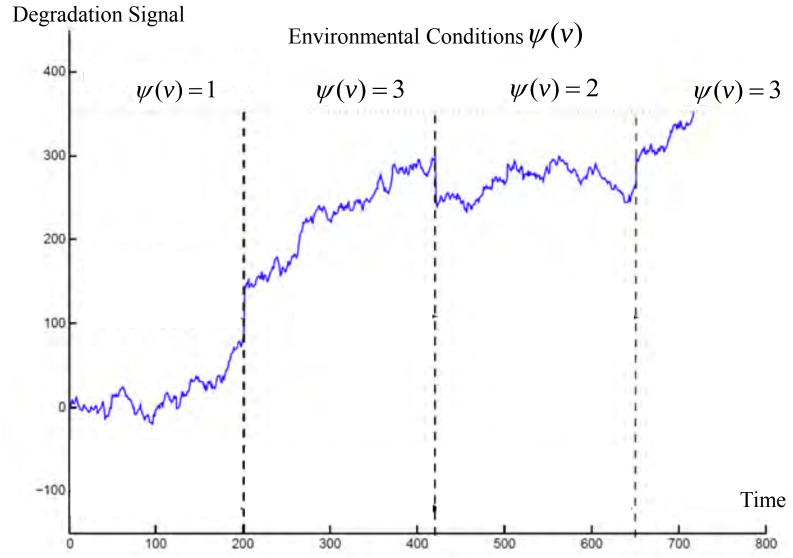


Figure 1. Simulation Results: Deterministic Environments

The component's time to failure corresponds to the first time the degradation signal $\{S(t) : t \geq 0\}$ crosses a fixed, deterministic threshold D, i.e., the failure time, T_D, is the first passage time,

$$T_D = \inf\{t > 0 : S(t) \geq D\}.$$

The primary objective of this paper is to provide a framework for online updating the remaining life distribution of the component based on discrete observations of the signal process $S(t)$ over time. Specifically, given a sequence of $k+1$ realized signal observations $\{s(t_i) : i = 0, 1, 2, \ldots, k\}$, let R_k denote the remaining time needed for the signal to first reach the threshold D, given the set of signal observations up to time t_k. Our aim is to estimate the distribution of R_k namely

$$\mathbb{P}(R_k \leq t - t_k | s(t_0), \ldots, s(t_k)), \quad t > t_k.$$

An important distinction needs to be made here regarding the random variable R_k and the standard residual life distribution. Assume for the moment that the distribution of T_D is known in advance. Then the residual life distribution is defined by

$$\mathbb{P}(T_D > t + t_k | T_D > t_k).$$

However, for real applications computing the residual life distribution in this way does not have much value because (1) the true distribution of T_D is not typically known in advance, and (2) it does not exploit available information about the current condition of the component information that can drastically affect the estimate of the remaining useful lifetime of the component.

The novelty of our approach is the updating of the residual life distribution using real-time sensor data to dynamically estimate parameters of the signal model $S(t)$ within a Bayesian framework. This distinguishes our hybrid stochastic model from other failure models that either do not update parameter estimates in real time, or do not consider the evolution of the environment and its effects on the component. Next, we describe an important aspect of the modeling framework, namely determining the set of environment states and arranging the elements of the set by ordering them according to their level of severity.

3.1. **Determining and Ordering the Environment States.** The degradation of fielded components are affected by many factors, some of which may result in higher rates of degradation than others. For example, the degradation of a rotating bearing is affected by the rotational speed of the bearing and the current load being applied to it, the current ambient temperature as well as humidity. However, not all of the factors are necessarily significant, and some of these possible combinations have a similar effect on the degradation rate so that states can be aggregated. Since an extremely large number of environmental states might raise possible computational issues, we focus on the scenario, in which the number of environmental factors is reasonably moderate. In what follows we briefly discuss a means, by which we choose the significant factor and order the environment states.

Suppose there are initially N factors, $X_1, \ldots, X_N$. We consider the response of degradation rate under a variety of settings to determine the impact of factors and/or their combinations. There are a number of ways to do this, and we use a linear regression model of the form

$$r = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_N X_N + \epsilon,$$

where $\beta_i, i = 1, 2, \ldots, N$ are unknown coefficients estimated by real data and ϵ denotes a zero-mean, normally distributed error term with homogeneous variance. To determine which factors (or their interactions) are most significant, one can employ any number of standard techniques (e.g. stepwise regression, best subset selection, a design-of-experiments approach, etc.).

Next, let us assume that some number of factors, say $B < N$, has been chosen for inclusion in the model, and denote these B significant factors by $Y_1, Y_2, \ldots, Y_B$. An environment condition is considered as a specific combination of the levels of the significant factors so that $E(l_1, \ldots, l_B)$ denotes the environment condition when Y_1 assumes level l_1, Y_2 assumes level l_2, and so forth. We now propose an algorithm to order the environmental conditions $E(l_1, \ldots, l_B)$ with regard to their impact on the corresponding degradation rate, denoted by $r(l_1, \ldots, l_B)$.

We discuss the sorting algorithm for two cases. First, we can first apply engineering expertise to sort the severity of environmental factors. For any two vectors, $(l_1, \ldots, l_B)$ and $(l'_1, \ldots, l'_B)$, define the partial order $(\geq)$ by

$$(l_1, \ldots, l_B) \geq (l'_1, \ldots, l'_B) \tag{2}$$

if $l_k \geq l'_k$, for $k = 1, 2, \ldots, B$. If the factor levels can be ordered in such a way, then it is clear that $r(l_1, \ldots, l_B) \geq r(l'_1, \ldots, l'_B)$. However, if the partial ordering of (2) cannot be established, then one can perform a hypothesis test of the form

$$H_0 : r(l_1, \ldots, l_B) \geq r(l'_1, \ldots, l'_B) \quad \text{against} \quad H_1 : r(l_1, \ldots, l_B) < r(l'_1, \ldots, l'_B)$$

using a simple experiment or observations obtained from a database of historical observations (if available). If there is sufficient evidence to reject H_0 at the α level of significance, we conclude that environment condition $E(l_1, \ldots, l_B)$ is less severe than the environmental condition $E(l'_1, \ldots, l'_B)$.

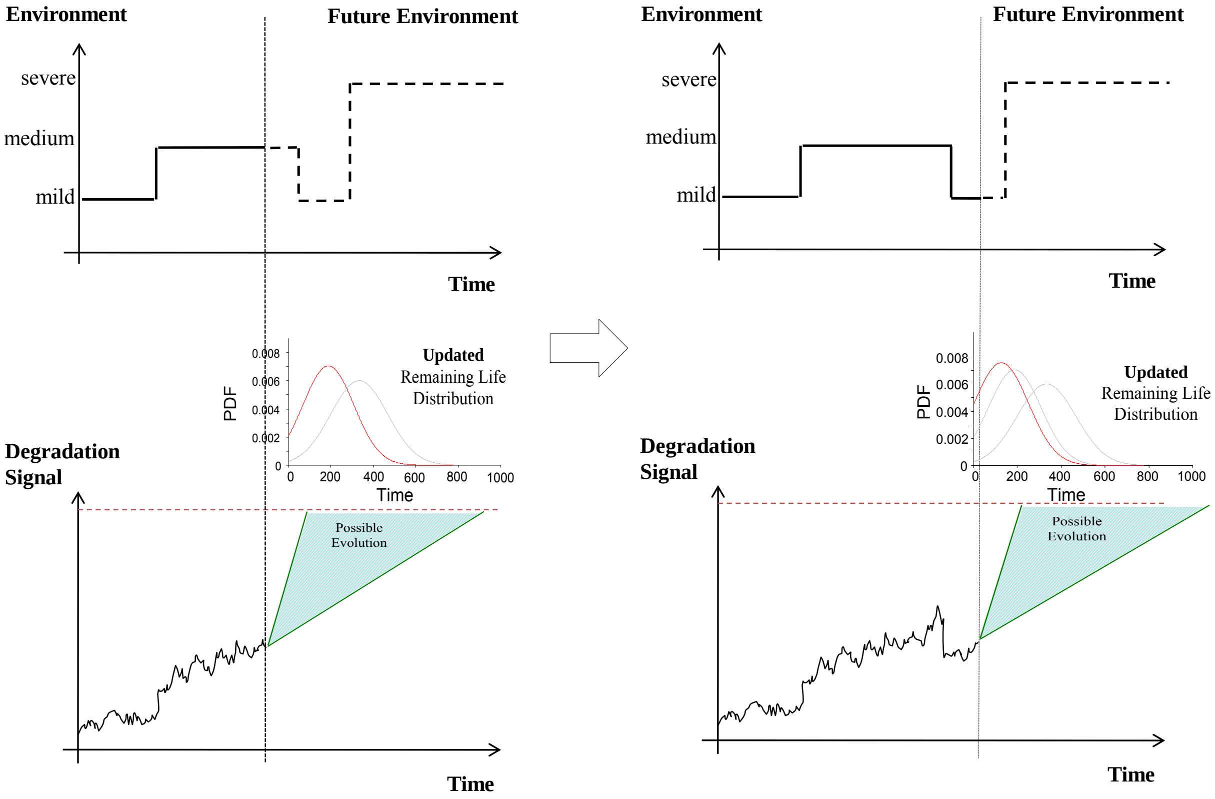


FIGURE 2. A Bayesian Updating Scheme at Two Observation Times

This comparison procedure can be applied pairwise to all environment conditions that do not satisfy the partial order of (2). Therefore, we can order all of the critical environment states in S from low to high by severity so that, for any two $i, j \in S, i < j$ implies that state i induces a smaller degradation rate than state j. The proposed approach investigates the effects of environmental factors on the rate of degradation. This approach can be naturally generalized in future research to incorporates the multiplicative interactions of environmental factors. The resulting extension will account for the possibility that significant environmental factors may vary for different environments. However, in this paper, we focus on effects of independent environmental factors on the degradation rate.

3.2. **Bayesian Updating of the Degradation Model.** In this subsection, we describe our Bayesian approach for updating the degradation model using prior information estimated from historical data in conjunction with real-time degradation signal observations obtained from a fielded component. For many applications, a historical database of degradation signals and environmental conditions is available for the estimation of prior information. However, even identical components can exhibit significant differences due to variations in the components' quality, etc. By combining both historical and real-time data, we are able to account for these inherent differences. Figure 2 shows the Bayesian updating scheme we propose in the work. The plots on the left and right columns illustrate the observed real-time information at two observation times. As we monitor more degradation signals and environmental information, we update the degradation model as well as the estimate of the residual life distribution. The real-time updating of the degradation signal $S(t)$ hinges upon the updating of the degradation rate function r, the mapping J, and the drift parameter γ. Let us denote the joint prior distribution of (r, J, γ) by $\pi_s(r, J, \gamma)$, where we suppress the dependence of r and J on the environment state $\psi(t)$ for notational convenience. By monitoring in real time the degradation signal of a fielded component (via sensors), along with the current state of the environment, we will update the prior distribution π_s. Suppose the degradation signal is monitored at times $t_0, t_1, \ldots, t_k$ such that $0 = t_0 < t_1 < \ldots < t_k$, and let $s(t_i)$ denote the observed signal at observation time t_i (the i^{th} observation epoch). We represent the set of observations by a vector $\mathbf{s_k} \in \mathbb{R}^{k+1}$, where $\mathbf{s_k} = (s(0), s(t_1), \ldots, s(t_k))'$. Additionally, we also observe the magnitude of jumps occurring at environment transition epochs. Therefore, in addition to the vector $\mathbf{s_k}$, we observe the ordered pairs,

$$\{(v_j, \psi(v_j^+)) : j = 1, 2, \ldots, n(t_k)\},$$

where v_j is the time of the n^{th} environment transition, $\psi(v_j^+)$ is the state of the environment just after the j^{th} environment transition where for some $\epsilon > 0$, $v_j^+ = \lim_{\epsilon \to 0} v_j + \epsilon$, and $n(t_k)$ is the cumulative number of environment transitions up to time t_k. Using this convention, the environment maintains state u_i over the interval $[v_{i-1}, v_i), i = 1, 2, \ldots, n(t_k)$.

Next, we denote the likelihood function of the degradation signal by $f_s(\mathbf{s_k}|r, J, \gamma)$. In the basic Bayesian framework, the posterior distribution of (r, J, γ) is computed by

$$\nu_s(r, J, \gamma|\mathbf{s_k}) = \pi_s(r, J, \gamma) f_s(\mathbf{s_k}|r, J, \gamma). \tag{3}$$

In the next subsection, we show how to use the signal and environment observations to dynamically estimate the residual life distribution of the component as it degrades over time.

3.3. Estimating the Residual Life Distribution.

When the future environmental profile is deterministic, the distribution of the residual life can be obtained using boundary crossing probabilities for a standard Brownian motion (BM) process. In particular, we consider a boundary that is piecewise linear over an interval $[0, T]$. We decompose the degradation signal into its deterministic and stochastic components, respectively so that

$$S(t) = \zeta(t) + \gamma W(t),$$

where

$$\zeta(t) = s(0) + \int_0^t r(\psi(v))dv + \int_0^t J(\psi(v))dv$$

is the deterministic portion of the signal, and $\gamma W(t)$ is the stochastic component. As shown in Figure 3, the probability that the signal is below the threshold D at time t is given by

$$\mathbb{P}(S(t) < D) = \mathbb{P}(\gamma W(t) < D - \zeta(t)),$$

where, by virtue of our modeling framework, the function $D - \zeta(t)$ is linear in t. For convenience, we denote this function by $d(t) = D - \zeta(t)$, where the slope of $\zeta(t)$ is $r(\psi(t))$. The probability that the degradation signal does not exceed D on $[0, T]$ is equivalent to the complementary probability that a standard BM process crosses a linear boundary whose slope depends explicitly on the current environment state. Boundary crossing probabilities for BM processes have been well-studied in the literature [17, 18]. For instance, if the function $d(t)$ is linear on $[0, T]$, [17] derived the (conditional) probability that a BM process crosses the linear boundary in this interval. This result was extended to piecewise linear functions without jump discontinuities on $[0, T]$ by [18]. Theorem 1 below extends Theorem 1 of [18] to consider the case when the function $d(t)$ is piecewise linear with jump discontinuities at finitely-many deterministic points. To this end, we partition the interval $[0, T]$ so that

$$[0, T] = \bigcup_{j=1}^{n} [v_{j-1}, v_j),$$

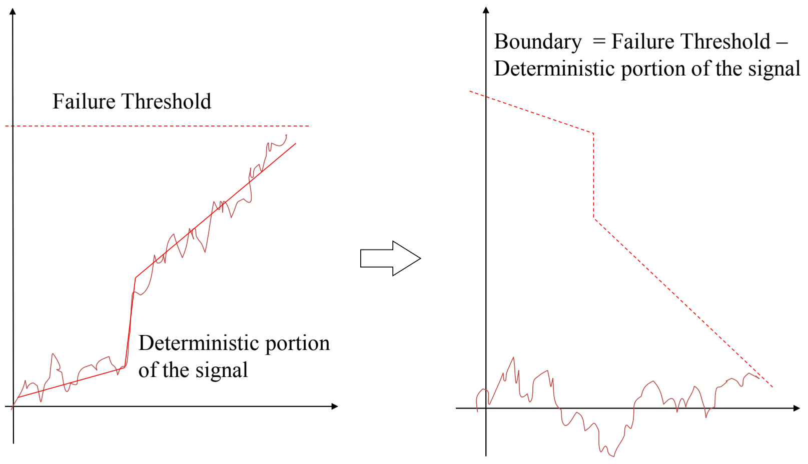

Failure Threshold
Deterministic portion of the signal
Boundary = Failure Threshold – Deterministic portion of the signal

FIGURE 3. Relations between the RLD and the crossing probability of a BM

where v_j denotes the time of the n^{th} jump in the signal process. It is important to note that both upward and downward jumps can occur. Therefore, to simplify notation in Theorem 1, let $m_j = \min\{d_j, d_j^-\}$ where $d_j = d(v_j)$ and $d_j^- = d(v_j^-), j = 0, 1, \ldots, n$, and let $\mathbf{d} = (d_1, d_1^-, d_2, d_2^-, \ldots, d_n, d_n^-)'$.

Theorem 3.1. Let $0 = v_0 < v_1 < \ldots < v_n = T$ denote n fixed jump times, and suppose $d(v)$ is linear on $[v_{j-1}, v_j), j = 1, 2, \ldots, n$ with $d(0) > 0$. Then for each $v \in [0, T]$, the complement of the crossing probability of a Brownian motion process, $\gamma W(v)$, with diffusion parameter γ is given by

$$\mathbb{P}(\gamma W(v) < d(v)) = \mathbb{E}[h(W(v_1), \ldots, W(v_n); \mathbf{d})], \tag{4}$$

where

$$h(x_1, x_2, \ldots, x_n; \mathbf{d}) = \prod_{j=1}^{n} \mathbf{1}(x_j < m_j/\gamma)\Delta(v_j, v_{j-1})$$

with

$$\Delta(v_j, v_{j-1}) = 1 - \exp\left[-\frac{2[d_{j-1}/\gamma - x_{j-1}][d_j^-/\gamma - x_j]}{v_j - v_{j-1}}\right],$$

and $\mathbf{1}(A)$ is the indicator function for condition A.

Proof. The details of the proof can be referred to [3]. □

Suppose the degradation signal has been sampled at k distinct times, $t_1, \ldots, t_k$, and the current time is $t_k < T$. The deterministic process, $\{\psi(t) : t_k < t \leq T\}$, is the future environmental profile from time t_k up to some future time T. On the interval $(t_k, T]$, the deterministic component of the degradation signal is

$$\zeta^k(t) = s(t_k) + \int_{t_k}^{T} r(\psi(v))dv + \int_{t_k}^{T} J(\psi(v))dv. \tag{5}$$

Define by R_k the residual life of the component at time t_k, given that the degradation signal has not crossed the threshold on the interval $[0, t_k]$. Applying Theorem 1, we estimate the distribution of R_k as follows:

(6) $$\mathbb{P}(R_k \leq T|\mathbf{s_k}) = 1 - \mathbb{E}[h(W(v_1), \dots, W(v_n); \mathbf{d_k})],$$

where $v_1, \dots, v_n$ are the transition epochs of the environment process $\{\psi(t) : t_k < t \leq T\}$, and $\mathbf{d_k}$ indicates the dependence of $\mathbf{d}$ on the observation time t_k. Equation (6), though simple in form, is not easy to compute due to the multidimensional integration requirement of Equation (4). To circumvent this integration, we propose a Monte-Carlo simulation approach to estimate $\mathbb{E}[h(W(v_1), \dots, W(v_n); \mathbf{d_k})]$. The algorithm is as follows:

Algorithm 1: Monte-Carlo Simulation Approach

Input: A sufficiently large number of realizations M', e.g., $M' = 5000$.

Output: The Monti-Carlo estimate of $\mathbb{P}(R_k \leq T|\mathbf{s_k})$.

Step 1 (simulation): For each $j = 1, \dots, M'$, generate n independent normal random variables, say $X_1, \dots, X_n$ such that for $i = 1, \dots, n$, $X_i \sim N(0, \gamma^2(v_i - v_{i-1}))$ with $v_0 = 0$ and

$$w_i^j = \sum_{k=1}^{i} X_k, i = 1, 2, \dots, n.$$

The vector $(w_1^j, \dots, w_n^j)$ is the j^{th} realization of $(W(v_1), \dots, W(v_n))$.

Step 2 (averaging): By applying the strong law of large numbers (SLLN), for sufficiently large M', we can estimate the residual lifetime distribution at time t_k by

$$\mathbb{P}(R_k \leq T|\mathbf{s_k}) \approx 1 - \frac{1}{M'} \sum_{j=1}^{M'} h(w_1^j, \dots, w_n^j; \mathbf{d_k}).$$

3.4. **An Illustrative Example.** We now illustrate the updating of the residual life distribution by describing a model with a specific form of the degradation rate function and the environment-dependent jump process. The rate of degradation, as a function of the environment state, is given by

$$r(\psi(v)) = \alpha + \beta\psi(v),$$

and the impact of jumps is captured by the function

$$J(\psi(v)) = \eta\psi(v),$$

where α, β, and η are the parameters of the degradation signal model as is γ, the diffusion coefficient. The prior marginal distributions of these parameters are

$$\alpha \sim N(\mu_1, \sigma_1^2),\ \beta \sim N(\mu_2, \sigma_2^2),\ \eta \sim N(\mu_3, \sigma_3^2),\ \gamma \sim N(\mu_4, \sigma_4^2).$$

The parameters are assumed to be mutually independent random variables. To estimate the posterior distribution of $(\alpha, \beta, \eta, \gamma)$, or equivalently of (r, J, γ), we next derive the likelihood function of degradation model. The likelihood function, conditioned on the parameter vector $(\alpha, \beta, \eta, \gamma)$, is denoted by

$$\mathbf{L}(\mathbf{s_k}|\alpha, \beta, \eta, \gamma) = \prod_{i=1}^{k} \phi_i[s(t_i) - s(t_{i-1})],$$

where ϕ_i is the probability density function of a normal distribution with mean

$$\int_{t_{i-1}}^{t_i} [\alpha + \beta\psi(v)]dv + \eta \int_{t_{i-1}}^{t_i} \psi(v)dv$$

and variance $\gamma^2(t_i - t_{i-1})$. To simplify notation, let $\mathbf{G}_{t_k} = \{\psi(v) : 0 \leq v \leq t_k\}$. The posterior distribution of $(\alpha, \beta, \eta, \gamma)$ is

$$\nu_s(\alpha, \beta, \eta, \gamma|\mathbf{s_k}, \mathbf{G}_{t_k}) = \pi_s(\alpha, \beta, \eta, \gamma) \times \prod_{i=1}^{k} \phi_i[s(t_i) - s(t_{i-1})],$$

where

$$\pi_s(\alpha,\beta,\eta,\gamma) = \varphi_1(\alpha)\varphi_2(\beta)\varphi_3(\eta)\varphi_4(\gamma),$$

and

$$\varphi_i(x) = \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left[-\frac{(x-\mu)^2}{2\sigma_i^2}\right], \quad i = 1,2,3,4.$$

The updated residual life distribution at time t_k is given by

$$\begin{aligned}
\mathbb{P}(R_k \le T|\mathbf{s_k}, \mathbf{G}_{t_k}) &= \int_{\alpha,\beta,\eta,\gamma} \mathbb{P}(R_k \le T|\mathbf{s_k},\alpha,\beta,\eta,\gamma) \times \nu_s(\alpha,\beta,\eta,\gamma|\mathbf{s_k},\mathbf{G}_{t_k}) \\
&= \int_{\alpha,\beta,\eta,\gamma} \mathbb{P}(R_k \le T|\mathbf{s_k},\alpha,\beta,\eta,\gamma)\pi_s(\alpha,\beta,\eta,\gamma) \times \prod_{i=1}^{k} \phi_i[s(t_i) - s(t_{i-1})] \\
&= \mathbb{E}_{\pi_s}\left[\mathbb{P}(R_k \le T|\mathbf{s_k},\alpha,\beta,\eta,\gamma) \prod_{i=1}^{k} \phi_i[s(t_i) - s(t_{i-1})]\right], \qquad (7)
\end{aligned}$$

where $\mathbb{E}_{\pi_s}$ is the expectation operator with respect to measure π_s. In a manner similar to that described for estimating $\mathbb{P}(R_k \le T|\mathbf{s_k})$, Equation (7) can be estimated using Markov chain Monte-Carlo (MCMC) techniques. Specifically, a sufficiently large number (say M') of realizations of $(\alpha,\beta,\eta,\gamma)$ can be simulated from the joint density $\pi_s(\alpha,\beta,\eta,\gamma)$ in order to estimate the corresponding values of

$$\mathbb{P}(R_k \le T|\mathbf{s_k},\alpha,\beta,\eta,\gamma) \prod_{i=1}^{k} \phi_i[s(t_i) - s(t_{i-1})].$$

Applying the SLLN, for sufficiently large M', the updated RLD is estimated by

$$\mathbb{P}(R_k \le T|\mathbf{s_k}, \mathbf{G}_{t_k}) \approx \frac{1}{M'} \sum_{(\alpha,\beta,\eta,\gamma)} \left[\mathbb{P}(R_k \le T|\mathbf{s_k},\alpha,\beta,\eta,\gamma) \prod_{i=1}^{k} \phi_i[s(t_i) - s(t_{i-1})]\right].$$

Numerical examples illustrating the quality of these estimates will be provided in Section 4.

4. Simulation Study

In this section, we will discuss the simulation study for the degradation model under deterministic environments. We compare our results with two benchmarks: [6] and [4]. In [6], the authors model degradation signals with time-varying degradation rate and jumps when environmental transition occurs. The authors assume that future environmental conditions remain the same as current environment when predicting residual life distributions. In [6], the authors consider a degradation model in which time-varying environments affect the degradation rate only. The authors develop an expression for the lifetime distribution under time transformation. However, possible jumps in degradation signals caused by transitions of environmental conditions are not considered. We will show that RLD prediction using our proposed method is more accurate than these two benchmark models. To evaluate performance of our proposed models we compute the prediction error as the follows:

$$\text{Prediction Error} = \frac{|\text{Actual Lifetime} - \text{Estimated Lifetime}|}{\text{Actual Lifetime}}.$$

We continue with the model in the illustrative example and simulate degradation signals up to failure using parameters listed in Table 1. Besides, we let the deterministic environmental condition evolve as the following step-wise function:

$$\psi(t) = \begin{cases} 1, & 0 \le t < 100 \\ 2, & 100 \le t < 200 \\ 1, & 200 \le t < 300 \\ 2, & 300 \le t \end{cases}.$$

Table 1. Parameter setup for the simulation study.

Prior Distribution of α	μ_1	0.3	σ_1^2	0.03
Prior Distribution of β	μ_2	0.5	σ_2^2	0.05
Prior Distribution of γ	μ_4	3	σ_4^2	0.3
Failure Threshold D	350			

To analyze the effects of jumps at the transition times, which distinct our model from most current research, we conducted simulation studies for $\mu_3 = 0, 10, 20$ with $\sigma_3^2 = 0.1\mu_3$, where μ_3 and σ_3^2 are the prior mean and variance of the jump factor η, respectively. For each setup of parameters, we simulate 1000 degradation signals, which are divided into two groups. The first group of signals are used to as training data, from which we estimate prior distributions of model parameters; and the second group is used as testing data, which we use to test online prediction of residual life distributions. The degradation signals are simulated via the following procedure:

(1) We simulate samples of α, β, η and γ using parameters listed in Table 1. The resulting realizations are denoted by $\alpha_i, \beta_i, \eta_i$ and γ_i.

(2) Using $\alpha_i, \beta_i, \eta_i, \gamma_i$ and $\psi(t)$, we simulate degradation signal $s_i(t)$ with the formula

$$s_i(t) = \int_0^t (\alpha_i \psi(v) + \beta_i) dv + \eta_i \int_0^t d\psi(v) + \gamma_i W(t)$$

until it hits the failure threshold D.

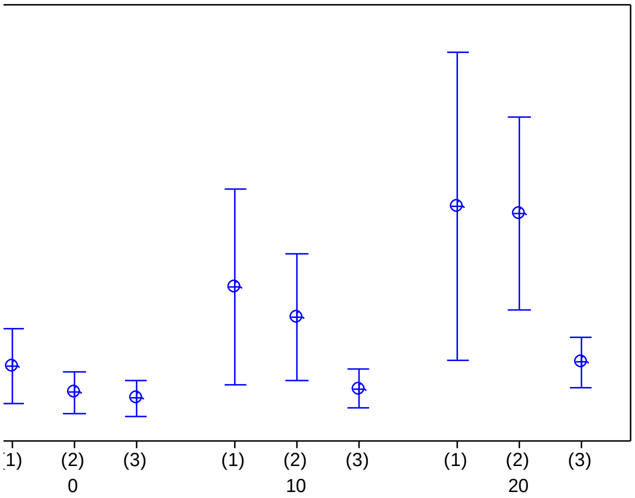
(3) For each simulated signal $s_i(t)$, we resample degradation signals at discrete epochs $t = 1, 2, 3, \ldots, k_i$, where k_i is the actual lifetime of signal $s_i(t)$.

To estimating the prior distributions of $(\alpha, \beta, \eta, \gamma)$, i.e., (μ_j, σ_j^2) for $i = 1, 2, 3, 4$, we applied an two-stage method as proposed in [13] to the training data. Based on these estimated prior distributions, we test online prediction of RLD using the testing data. Residual lifetimes are computed at the 50^{th} and 90^{th} percentiles of lifetimes; the sample mean and variance of prediction error are computed. We compare the results of our model with the following two benchmarks [4] and [6]. The result is presented in Figure 4, where (1) represents results from [4]; (2) represents results from [6]; and (3) represents results from our proposed method. For each approach, we estimate the RLD in three cases: the prior mean of the jump factor η equals 0, 10, and 20.

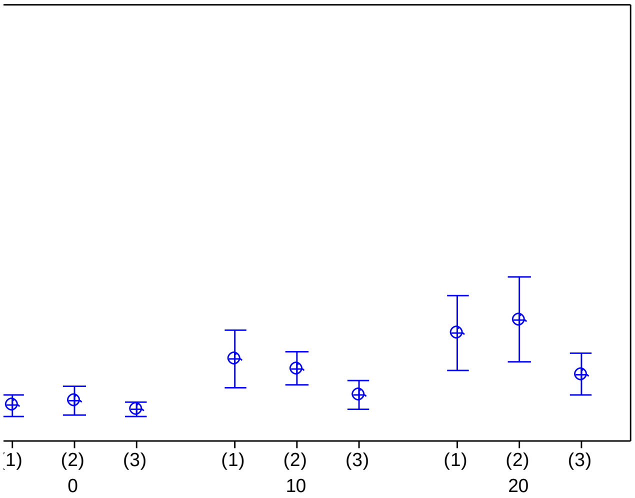
We observed that the prediction accuracy of our proposed approach is higher than that of [6], in which assume that the future environmental condition remains the same as the current environmental condition as they predict the RLD. With regard to [4], the authors forecast the lifetime distribution of components without considering the possible shocks caused by environmental changes. As $\eta = 0$, the transitions of environments do not cause jumps in the amplitude of degradation signals. The resulting prediction accuracy of our model and that of [4] are very close. However, for larger jumps (the mean of η equals 10 and 20), our proposed method has the smallest prediction error because we completely characterize the features of simulated signals and utilize online data.

5. A Case Study

In this subsection, we present a case study that involves ball bearings operating under deterministic environmental profiles. Bearings are a crucial component in nearly all rotating machinery, such as hot rolling mills, steam and wind turbines, and many other applications. The failure of bearings has been widely studied in the literature, and vibration monitoring is considered as one of the most widely used techniques for monitoring bearing degradation ([8]). We use vibration-based



(a) Prediction Error: 50th Percentile



(b) Prediction Error: 90th Percentile

Figure 4. Simulation Results: Deterministic Environments

degradation signals generated from an experimental test rig that is designed to perform accelerated degradation tests on ball bearings using different loads and rotational speeds.

We conduct vibration analysis and construct the vibration-based degradation signals based on the fact that the bearing's vibration amplitude of defective frequencies is associated with its health condition. In particular, we compute the average amplitude of the defective frequency and its first five harmonics. We limit ourselves to the first five harmonics since higher-order harmonics have been observed to behave erratically. Furthermore, we define bearing failure based on the root mean square (RMS) value of the overall vibration of the test rig. According to industrial standards for machinery vibration, ISO 2372, 2.0–2.2 G (G denotes gravitational acceleration) represents a vibration-based danger level for applications involving general purpose mid-size machinery. We use this standard to identify a corresponding failure threshold of 0.025 Vrms (Root Mean Square Volts).

In this study, we examine the effects of two environmental factors: the load applied to the bearing and the rotational speed of the bearing. In particular, two different loads (400 lbs and 500 lbs) and two different rotational speeds (2,200 rpm and 2,600 rpm) are considered; therefore, initially there are four distinct environmental conditions: (2,200 rpm, 400 lbs), (2,200 rpm, 500 lbs), (2,600 rpm, 400 lbs), and (2,600 rpm, 500 lbs). To construct the mapping from the environmental conditions to the environmental state space, we apply the procedure in Section 3.1 that determines and orders the environmental states so that state 1 represents the state with the lowest degradation rate and state 4 the highest degradation rate. Let $r(s,l)$ denote the degradation rate when the rotational speed is s (rpm), and the load is l (lbs). Since higher load or speed accelerates the degradation of bearings ([14]), we obtain the following inequalities of degradation rates in various environment states:

(1) r(2,200 rpm, 400 lbs) $<$ r(2,200 rpm, 500 lbs) $<$ r(2,600 rpm, 500 lbs),

(2) r(2,200 rpm, 400 lbs) $<$ r(2,600 rpm, 400 lbs) $<$ r(2,600 rpm, 500 lbs).

To establish a complete ordering of the degradation rates in all four environmental conditions, we evaluate r(2,200 rpm, 500 lbs) and r(2,600 rpm, 400 lbs) using the vibration data. Our analysis, which is based on the hypothesis testing procedure presented in Section 3.1, indicates that r(2,200 rpm, 500 lbs) $<$ r (2,600 rpm, 400 lbs). Therefore, the final ordering of degradation rates (from least severe to most severe) is $r(2200, 400) < r(2200, 500) < r(2600, 400) < r(2600, 500)$. The resulting environmental states included in **S** are summarized in Table 2.

Table 2. Definition of ordered environmental states.

Environmental condition	Environmental state
(2,200 rpm, 400 lbs)	1
(2,200 rpm, 500 lbs)	2
(2,600 rpm, 400 lbs)	3
(2,600 rpm, 500 lbs)	4

We conducted two groups of bearing tests. The first set of 12 experiments was used to estimate prior distribution parameters for the degradation model, and these are designated as ID 1 to 12. The second set of 3 experiments are used for validation, and these are labeled as ID 13 to 15. The experimental setups for these two groups are summarized in Table 3. The bearings in validation tests are run to failure so that we can observe the actual lifetime and compare it with our estimated results. We estimate the prior distributions of model parameters using the degradation signals from experiments 1-12 and assess online prediction of the RLD using the degradation signals from experiments 13-15. The RLDs are estimated at the 30th, 60th and 90th percentiles of the components' lifetimes. The means of the estimated lifetimes and the corresponding prediction errors are presented in Table 4. We observe that the prediction errors at the 90th percentile of the lifetime are relatively small. This is, in part, because the environmental condition remains constant for all of the three online experiments after the 90th percentile of the lifetime.

6. Conclusion

In this paper, we have presented a stochastic degradation modeling framework that computes the RLD of partially-degraded components operating under time-varying environmental or operating

Table 3. Experiments for prior information and online validation.

Experiment ID	Operating conditions	Number of bearings
1	(2,200 rpm, 400 lbs)	4
2	(2,200 rpm, 500 lbs)	4
3	(2,600 rpm, 400 lbs)	4
4	(2,600 rpm, 500 lbs)	4
5	(2,200 rpm, 400 lbs) → (2,200 rpm, 500 lbs)	2
6	(2,200 rpm, 500 lbs) → (2,200 rpm, 400 lbs)	2
7	(2,600 rpm, 400 lbs) → (2,600 rpm, 400 lbs)	2
8	(2,600 rpm, 400 lbs) → (2,600 rpm, 400 lbs)	2
9	(2,200 rpm, 400 lbs) → (2,600 rpm, 400 lbs)	2
10	(2,600 rpm, 400 lbs) → (2,200 rpm, 400 lbs)	2
11	(2,600 rpm, 400 lbs) → (2,200 rpm, 400 lbs)	2
12	(2,200 rpm, 400 lbs) → (2,600 rpm, 400 lbs)	2
13	(2,200 rpm, 400 lbs) → (2,600 rpm, 400 lbs)	1
14	(2,600 rpm, 400 lbs) → (2,200 rpm, 400 lbs)	1
15	(2,200 rpm, 400 lbs) → (2,200 rpm, 500 lbs)	1

Table 4. Prediction of lifetime for validation data.

ID	Actual Lifetime	30th Percentile	60th Percentile	90th Percentile
13	283	318.28 (12.5% error)	301.31 (6.5% error)	289.81 (2.4% error)
14	546	489.56 (10.3% error)	575.14 (5.3% error)	563.32 (3.1% error)
15	402	440.24 (9.5% error)	432.21 (7.5% error)	387.86 (3.8% error)

conditions. This framework uses historical and real-time signals related to the environmental conditions, as well as the underlying physical degradation process. In contrast to most existing models, we compute the components RLD in real time by utilizing the potential profile of future environmental conditions that the component is likely to experience. We develop a degradation model for the components under deterministic environments, which incorporates information from(1) historical degradation signals and the corresponding environmental conditions from a population of similar components, (2) real-time degradation signals and the corresponding environmental conditions from the components of interest, and (3) knowledge of future environmental conditions.

Our proposed framework raised a few interesting and important questions that are worthy of further consideration. First, as stated in Section 3.1, additional developments are needed to investigate environmental factors along with their interactions when the number of environmental states is large. Moreover, it is possible that the future environmental profile might not be deterministic in real world applications. Extensions of this work will include an examination of degradation models when the future environmental condition is unknown.

References

[1] D. Banjevic and A. Jardine. Calculation of reliability function and remaining useful life for a markov failure time process. *IMA Journal of Management Mathematics*, 17(2):115, 2006.

[2] D. Banjevic, A. Jardine, V. Makis, and M. Ennis. A control-limit policy and software for condition-based maintenance optimization. *INFOR-OTTAWA-*, 39(1):32–50, 2001.

[3] L. Bian, N. Gebraeel, and J. Kharoufeh. Degradation models in deterministic and randomly varying future environments. *submitted to Operations Research*, 2011.

[4] K. Doksum and A. Hóyland. Models for variable-stress accelerated life testing experiments based on wiener processes and the inverse gaussian distribution. *Technometrics*, 34(1):74–82, 1992.

[5] N. Gebraeel, M. Lawley, R. Li, and J. Ryan. Residual-life distributions from component degradation signals: A bayesian approach. *IIE Transactions*, 37(6):543–557, 2005.

[6] N. Gebraeel and J. Pan. Prognostic degradation models for computing and updating residual life distributions in a time-varying environment. *Reliability, IEEE Transactions on*, 57(4):539–550, 2008.

[7] A. Ghasemi, S. Yacout, and M. Ouali. Evaluating the reliability function and the mean residual life for equipment with unobservable states. *Reliability, IEEE Transactions on*, 59(1):45–54, 2010.

[8] T. Harris, M. Kotzalas, and I. ebrary. *Rolling bearing analysis.* Wiley, 1984.

[9] A. Jardine, D. Banjevic, and V. Makis. Optimal replacement policy and the structure of software for condition-based maintenance. *Journal of Quality in Maintenance Engineering*, 3(2):109–119, 1997.

[10] J. Kharoufeh. Explicit results for wear processes in a markovian environment. *Operations Research Letters*, 31(3):237–244, 2003.

[11] J. Kharoufeh, D. Finkelstein, and D. Mixon. Availability of periodically inspected systems with markovian wear and shocks. *Journal of Applied Probability*, 43(2):303–317, 2006.

[12] C. Liao and S. Tseng. Optimal design for step-stress accelerated degradation tests. *Reliability, IEEE Transactions on*, 55(1):59–66, 2006.

[13] C. Lu and W. Meeker. Using degradation measures to estimate a time-to-failure distribution. *Technometrics*, 35(2):161–174, 1993.

[14] W. Nelson. *Accelerated testing: statistical models, test plans and data analyses.* Wiley Online Library, 1990.

[15] C. Park and W. Padgett. New cumulative damage models for failure using stochastic processes as initial damage. *Reliability, IEEE Transactions on*, 54(3):530–540, 2005.

[16] L. Pettit and K. Young. Bayesian analysis for inverse gaussian lifetime data with measures of degradation. *Journal of Statistical Computation and Simulation*, 63(3):217–234, 1999.

[17] D. Siegmund. Boundary crossing probabilities and statistical applications. *The Annals of Statistics*, 14(2):361–404, 1986.

[18] L. Wang and K. Potzelberger. Boundary crossing probability for brownian motion and general boundaries. *Journal of Applied Probability*, 34(1):54–65, 1997.

[19] X. Zhao, M. Fouladirad, C. Bérenguer, and L. Bordes. Condition-based inspection/replacement policies for non-monotone deteriorating systems with environmental covariates. *Reliability Engineering & System Safety*, 95(8):921–934, 2010.

SPARSE INVERSE GAUSSIAN PROCESS REGRESSION WITH APPLICATION TO CLIMATE NETWORK DISCOVERY

KAMALIKA DAS* AND ASHOK N. SRIVASTAVA**

Abstract. Regression problems on massive data sets are ubiquitous in many application domains including the Internet, earth and space sciences, and finances. Gaussian Process regression is a popular technique for modeling the input-output relations of a set of variables under the assumption that the weight vector has a Gaussian prior. However, it is challenging to apply Gaussian Process regression to large data sets since prediction based on the learned model requires inversion of an order n kernel matrix. Approximate solutions for sparse Gaussian Processes have been proposed for sparse problems. However, in almost all cases, these solution techniques are agnostic to the input domain and do not preserve the similarity structure in the data. As a result, although these solutions sometimes provide excellent accuracy, the models do not have interpretability. Such interpretable sparsity patterns are very important for many applications. We propose a new technique for sparse Gaussian Process regression that allows us to compute a parsimonious model while preserving the interpretability of the sparsity structure in the data. We discuss how the inverse kernel matrix used in Gaussian Process prediction gives valuable domain information and then adapt the inverse covariance estimation from Gaussian graphical models to estimate the Gaussian kernel. We solve the optimization problem using the alternating direction method of multipliers that is amenable to parallel computation. We demonstrate the performance of our method in terms of accuracy, scalability and interpretability on a climate data set.

1. Introduction

In many application domains, it is important to predict the value of one feature based on certain other measured features. For example, in the Earth Sciences, predicting the precipitation at one location given the humidity, sea surface temperature, cloud cover, and other related factors is an important problem in climate modeling. For such problems, simple linear regression based on minimization of the mean squared error between the true and predicted values can be used for modeling the relationship between the input and the target features. In decision support systems which use these predictive algorithms, a prediction with low confidence may be treated differently than if the same prediction was given with high-confidence. Thus, while the predicted value from the regression function is clearly important, the confidence in the prediction is equally important. A simple model such as linear regression does not provide us with that information. Also, models like linear regression, in spite of being easy to fit and being highly scalable, fail to capture nonlinear relationships in the data. Gaussian Process regression (GPR) is one regression model that can capture nonlinear relationships and outputs a distribution of the prediction where the variance of the predicted distribution acts as a measure of confidence in the prediction. Moreover, the inverse kernel (or covariance) matrix has many interesting properties along the gaussian graphical model perspective, that can be exploited for better understanding relationships within the training examples. Depending on the nature of the data, these relationships can indicate dependencies (causalities) for certain models.

However, predictions based on GPR method, requires inversion of a kernel (or covariance) matrix of size $n \times n$, where n is the number of training instances. This kernel inversion becomes a bottleneck for very large datasets. Most of the existing methods for efficient computation in GPR involve numerical approximation techniques that exploit data sparsity. While this does speed up

*SGT Inc., NASA Ames Research Center, Kamalika.Das@nasa.gov, **NASA Ames Research Center, ashok.n.srivastava@nasa.gov
.

GPR computations, one serious drawback of these approximations is that the resulting GPR model loses interpretability. Even if we get reasonably accurate predictions, we fail to unearth significant connections between the training points or identify the most influential training points for a specific set of test points.

In this paper we propose a sparse GPR algorithm which not only scales to very large datasets but also allows us to construct a complete yet sparse inverse covariance matrix, thereby facilitating interpretability. The method proposed in this paper induces sparsity by introducing a regularizer in a pseudo negative log likelihood objective used for covariance selection. This forces the algorithm to seek a parsimonious model for GPR prediction having excellent interpretability. One of the highlights of the solution technique used in this paper is a completely parallelizable framework for solving the inverse covariance estimation problem using the alternating direction method of multipliers (ADMM) that allows us to exploit modern parallel and multi-core architectures. This also addresses the situation where the entire covariance matrix cannot be loaded into memory due to size limitations.

The rest of the paper is organized as follows. In the next section (Section 2) we present some background material related to GPR and some existing methods of solving the GPR problems. In Section 3 we discuss the equivalence between inverse kernel and covariance matrices. Next we present our new sparse inverse covariance matrix using ADMM technique (Section 4). Experimental results are discussed in Section 5. We conclude the paper in Section 6.

2. Background: Gaussian Process Regression

Since this paper proposes a technique of model fitting using Gaussian Process regression, we start with a brief review of it here. Rasmussen and Williams [15] provide an excellent introduction on this subject. Gaussian Process regression is a generalization of standard linear regression. If $\mathbf{X}$ is the training data set having n multidimensional observations (rows) $\mathbf{x}_1, \ldots, \mathbf{x}_n$, with each $\mathbf{x}_i \in \mathbb{R}^D$ and the corresponding target is represented by a $n \times 1$ vector $\mathbf{y}$, then the standard linear regression model is:

$$f(\mathbf{x}) = \mathbf{x}\mathbf{w}^T, \qquad y = f(\mathbf{x}) + \epsilon$$

where $\mathbf{w}$ is a D-dimensional weight vector of parameters and ϵ is additive Gaussian noise such that $\epsilon \sim \mathcal{N}(0, \sigma^2)$. Assuming that we choose the prior distribution of the weights to be Gaussian with mean zero and covariance Σ_p, the posterior distribution of the weights, following Bayesian inferencing techniques, can be written as:

$$p(\mathbf{w}|\mathbf{X}, \mathbf{y}) \sim \mathcal{N}\left(\frac{1}{\sigma^2}A^{-1}\mathbf{X}^T y, A^{-1}\right)$$

where $A = \sigma^{-2}\mathbf{X}^T\mathbf{X} + \Sigma_p^{-1}$. Given the posterior and the likelihood, the predictive distribution of a test input $\mathbf{x}^*$ is obtained by averaging over all possible models ($\mathbf{w}$) to obtain:

$$p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{X}, \mathbf{y}) \sim \mathcal{N}\left(\frac{1}{\sigma^2}\mathbf{x}^* A^{-1} X^T \mathbf{y}, \mathbf{x}^* A^{-1}\mathbf{x}^{*T}\right)$$

Using a kernel (covariance) function $k(\mathbf{x}_i, \mathbf{x}_j)$ in place of a mapping from input space to an N-dimensional space, and applying some algebraic manipulations, we can write the predictive mean and variance of the posterior distribution as

$$\widehat{\mathbf{y}}^* = K^*(\sigma^2 I + K)^{-1}\mathbf{y} \tag{1}$$

$$C = K^{**} - K^*(\sigma^2 I + K)^{-1}K^{*T} \tag{2}$$

where the ij^{th} entry of K is $k(\mathbf{x}_i, \mathbf{x}_j)$ and K^* and K^{**} are similarly the cross covariance matrices involving the test point $\mathbf{x}^*$. Equations 1 and 2 pose significant computational challenge due to the requirement of inverting the covariance matrix K of size n^2. If the number of observations n is large, the $O(n^3)$ operation can be a bottleneck in the process of using Gaussian Process regression.

In the next section, we discuss several techniques that have been proposed in the literature for approximating the inverse matrix for large datasets.

2.1. **Existing methods for efficient GP computation.** Approximations are introduced in the Gaussian Process literature for either finding closed-form expressions for intractable posterior distributions or for gaining computational advantage for large data sets. Here we are interested in the second goal and, therefore, briefly discuss the existing research in this area. Smola and Bartlett [16] describe a sparse greedy method that does not require evaluating the full covariance matrix K and finds an approximation to the maximum aposteriori estimate by selecting an 'active' subset of columns of K by solving an expensive optimization problem. The running time of the numerical approximation is reduced from $O(n^3)$ to $O(nm^2)$ where m ($m \ll n$) is the rank of the matrix approximation.

A related approach of low rank matrix approximation called the subset of regressors method [21] involves selecting the principal sub-matrix of the unperturbed covariance matrix K by matrix factorization. Though this method has been found to be numerically unstable, recent research by Foster *et al.* [8] has shown that if we use partial Cholesky decomposition to factorize the covariance matrix and perturb the low rank factor such that independent rows and columns form the principal sub-matrix, then the approximation we get is numerically stable. The authors report excellent accuracy using their approximation calculations when the rank of the reduced matrix is a small factor (5) times the rank of the original data matrix X.

The generalized Bayesian committee machine [20] is another approach for reducing the computational complexity of any kernel-based regression technique, by dividing the data arbitrarily into M almost equal sized partitions, training a different estimator on each partition, and combining the estimates given by the different estimators using the inverse of the variance to ensure that least certain predictions are given the smallest weights in the final prediction. This method allows us to choose M to be equal to $K\alpha$ so that it becomes linear in K in computational complexity. The Bayesian Committee Machine weights the training data based on the test points using a block diagonal approximation and, therefore, the model needs to be retrained every time a new test set comes in. A related method recently proposed by Das and Srivastava [4] works for multimodal data. It partitions the input space into multiple clusters, with each one corresponding to one mode of the data distribution. Then, each cluster is modeled using a normal distribution and all points which are not modeled by any of the normal distributions are grouped using a separate cluster. Each cluster learns a separate GP model and a weighted sum based prediction is used for the gating.

A recent development is the ℓ_1 penalized GPR method (GPLasso) introduced by Yan and Qi [22] in which the authors explore sparsity in the output rather than the input. They propose a GPR technique that minimizes the Kullback-Leibler divergence between the posterior distributions of the exact and the sparse solutions using a ℓ_1 penalty on the optimization. They pose this problem as a LASSO optimization [19] and solve a rank reduced approximate version of this using the Least Angle Regression (LARS) method [7]. The authors present this work as a pseudo output analogy of the work by Snelson *et al.* [17]. Quiñonero-Candela and Rasmussen [14] provide a unifying view of all sparse approximation techniques for Gaussian Process regression by analyzing the posterior and reinterpreting each algorithm as an exact inferencing method using approximate priors.

All the methods discussed in this section apply some form of numerical approximation technique to reduce the rank of the kernel matrix for efficient matrix inversion. As a result, they often lose model interpretability — a value at any position of the reduced rank inverted matrix cannot be traced back to any cell of the original kernel. In many domains, however understanding the sparsity structure is important. For example, in Earth Sciences, it is not only important to get good predictions from the GPR model, but it is also important to understand how different geographical regions are connected and how these locations influence one another. Unfortunately, none of the efficient GPR techniques allow this. Our proposed technique in the next section not only learns a sparse GP model but also allows domain scientists to draw conclusions about the sparsity structure by studying the inverse covariance matrix.

3. SPI-GP:Sparse Gaussian Process using inverse covariance estimation

Let $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ be a set of multi-dimensional gaussian observations such that

$$\mathbf{x}_i \sim \mathcal{N}(\mu, \Sigma) \in \mathbb{R}^d$$

where $\mu \in \mathbb{R}^d$ and $\Sigma \in \mathbb{R}^{d \times d}$ are the mean and covariance matrices. While the mean μ measures the center of the distribution, the covariance matrix Σ measures the pairwise (linear) relationship between the variables. It is well known that a value of 0 at any cell of Σ implies independence of the observations:

$$\Sigma_{i,j} = 0 \Rightarrow P(\mathbf{x}_i \mathbf{x}_j) = 0$$

which means $\mathbf{x}_i$ and $\mathbf{x}_j$ are independent. In many cases, we may be interested in studying how two variables influence each other when the information about the other variables are taken into consideration. One way of doing this is by studying the inverse covariance matrix, also known as the concentration matrix or precision matrix denoted by Σ^{-1}. Unlike Σ, a value of 0 in any cell of Σ^{-1} implies conditional independence among those variables [1]. For example, $\mathbf{x}_i$ and $\mathbf{x}_j$ are conditionally independent, given all the other variables, if Σ^{-1}=0. Mathematically,

$$\Sigma^{-1}_{i,j} = 0 \Rightarrow P(\mathbf{x}_i \mathbf{x}_j | \mathbf{x}_{-i,-j}) = 0$$

where $\mathbf{x}_{-i,-j}$ denotes all the variables other than $\mathbf{x}_i$ and $\mathbf{x}_j$. Note that independence of elements implies conditional independence but not vice-versa *i.e.* a value of 0 at any cell of Σ implies that the corresponding location of Σ^{-1} is also 0; but a non-zero value at any cell of Σ matrix does not imply that the corresponding cell of Σ^{-1} will also be non-zero. The reason for studying Σ^{-1} rather than Σ, is for many gaussian distributed variables, there is more sparsity in the inverse covariance matrix than in the covariance matrix and this sparsity reveals interesting data relationships. It has been shown in [9], that inverting a covariance matrix (with the additional assumption that the inverse is sparse) is equivalent to learning a graphical model, where each node in the model corresponds to a feature and the absence of an edge between any two signifies that those features are conditionally independent.

In the case of GPR, the kernel matrix between the observations (see Eqn. 1 and 2) can be viewed as a covariance matrix among the function outputs. Formally, a gaussian process is defined as a collection of random variables, any finite number of which is jointly gaussian. Hence, it is a distribution over functions, completely specified by its mean function and covariance function as,

$$f(\mathbf{x}_i) \sim GP(m(\mathbf{x}_i), k(\mathbf{x}_i, \mathbf{x}_j))$$

where $m(\mathbf{x}_i) = E[f(\mathbf{x}_i)]$ and $k(\mathbf{x}_i, \mathbf{x}_j) = E[f(\mathbf{x}_i) - m(\mathbf{x}_i)][f(\mathbf{x}_j) - m(\mathbf{x}_j)]$ are the mean function and covariance function of some real process $f(\mathbf{x}_i)$. Note that $f(\mathbf{x}_i)$ are random variables and GP fits a distribution over all possible $f(\mathbf{x}_i)$. In our case since $f(\mathbf{x}_i)$'s are linear functions $f(\mathbf{x}_i) = \mathbf{x}_i \mathbf{w}^T$, the mean and covariance of GP can be stated as,

$$m(\mathbf{x}_i) = E[f(\mathbf{x}_i)] = \mathbf{x}_i E[\mathbf{w}^T] = 0$$

$$k(\mathbf{x}_i, \mathbf{x}_j) = E[f(\mathbf{x}_i) f(\mathbf{x}_j)] = \mathbf{x}_i E[\mathbf{w}^T \mathbf{w}] \mathbf{x}_i^T = \mathbf{x}_i \Sigma_p \mathbf{x}_i^T$$

where $\mathbf{w} \sim N(0, \Sigma_p)$ denotes the prior distribution of the weights. The covariance function k, also known as the kernel function specifies the covariance between a pair of random variables

$$\mathrm{cov}(f(\mathbf{x}_i), f(\mathbf{x}_j)) = E[f(\mathbf{x}_i) f(\mathbf{x}_j)] = k(\mathbf{x}_i, \mathbf{x}_j)$$

Therefore, a kernel function computed over the pairwise input points is equivalent to a covariance between the outputs. There are several choices of the kernel functions available. In this paper we have used the widely used gaussian radial basis function (rbf) kernel:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right)$$

where σ is known as the bandwidth parameter which is typically learned from the data.

In many GPR applications, it is not only important to get good prediction accuracy, but also understand the model. For example, in Earth Sciences teleconnections [11] reveal important symmetric and sometimes causal relationships among different events observed in geographically distant locations and can be studied by exploiting sparsity in the inverse kernel in GPR. Another possible application area is the study of climate networks [18]. Fig. 1 (left) shows the observed precipitation data of the world overlaid on a 360 × 720 grid. Figs. 1 (center and right) show a kernel or similarity matrix generated from the data and the corresponding inverse covariance matrix. Each cell in the kernel (except the diagonal) denotes the similarity between the precipitation values of a grid location (lower resolution). The highlighted row and column correspond to the location marked in white on the world map. In this paper we are interested in studying the sparsity pattern of the inverse covariance matrix, with the information that sparsity patterns in the inverse covariance matrix leads to conditional independence among the locations of interest.

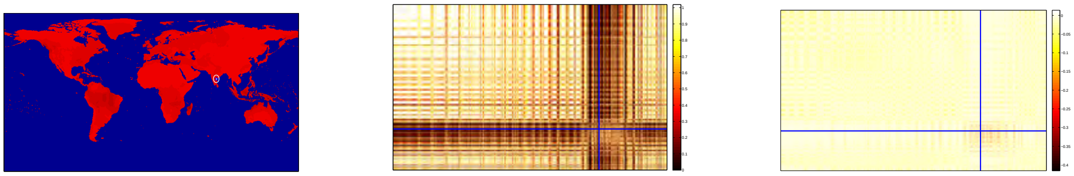


FIGURE 1. Precipitation data of the world map (top figure). Note that the data is only available for land (the ocean locations have fill values of -9999). The figure in the center shows a kernel in which similarity is computed between every pair of locations from the precipitation data. Note the location marked with a circle on the left figure corresponds to the row and column in blue on the center and right figure. The right figure shows the inverse kernel matrix.

4. Sparse covariance selection

There exist several techniques in the literature for solving the inverse covariance estimation problem also known as the covariance selection problem.

Given a dataset containing d features, Meinshausen *et al.* [13] infers the graphical model (and therefore the inverse covariance matrix) by taking one variable at a time and then finding all the connections of that variable with all of the other ones. For each variable d_i in the dataset, the method constructs a lasso regression problem by taking all the other variables as inputs and d_i as the target with an additional sparsity constraint on the solution weights. The non-zero entries of the weight vector signifies a connection between that feature and the target d_i. To deal with inconsistencies among the connections, the authors have proposed two schemes: (1) in the **AND** technique, an edge is established in the graphical model between any two features d_i and d_j iff both d_i and d_j have non-zero entries in the weight vector when they are each used as target in different lasso problems, and (2) in the **OR** scheme, an edge is established if either d_i or d_j has a non-zero weight when the other is taken as the target. One serious drawback of this method is the number of independent lasso problems increases linearly with the size of the feature space.

Banerjee *et al.* [1] propose a different solution to the inverse covariance selection problem. They show that based on Dempster's theory [5], estimating the inverse covariance matrix is equivalent to minimizing the pesudo negative log likelihood. The objective function takes the form:

$$\mathbf{Tr}(KS) - \log det(S)$$

where K is the empirical covariance (or kernel) matrix and S is the desired inverse of K i.e. $S = K^{-1}$, $\mathbf{Tr}(\cdot)$ is the trace of a matrix, and $det(\cdot)$ is the matrix determinant. Solution to the above equation is stable when an additional sparsity constraint is imposed on the inverse, *i.e.*

$$\mathbf{Tr}(KS) - \log det(S) + \lambda \|S\|$$

where λ controls the degree of sparsity. This is a convex optimization problem and in order to solve this, the authors propose a block-wise interior point algorithm.

Friedman *et al.* [9] generalizes both these papers and present a very efficient algorithm based on the lasso technique. Their objective function is the same as used by Banerjee *et al.* [1] *i.e.* they try to maximize the log likelihood of the model with the additional sparsity constraint. They show that the solution proposed by Meinshausen [13] is an approximation of the log likelihood estimate proposed by Banerjee *et al.* [1]. They propose a new algorithm based on coordinate descent to solve the same trace minimization problem. This algorithm is based on recursively solving lasso subproblems for each variable until convergence. The authors note that this new algorithm is at least 50 to 4000 times faster than existing techniques and therefore scales to much larger data sets.

However, there is one drawback common to all these optimization techniques. All these techniques assume that the data can be loaded in computer memory for the analysis. Unfortunately, in applications such as Earth Sciences, most datasets are massive — they contain millions of observations (locations) and therefore constructing a full covariance matrix in memory is itself impossible, leaving aside the computational power necessary to run these optimization techniques for inverse estimation. To solve the large scale inverse covariance estimation problems which do not fit into the memory of one machine, in this paper we propose our SPI-GP method which works by distributing the workload among a network of machines. The technique we follow is based on the method of Alternating Direction Method of Multipliers (ADMM) which is a distributable algorithm for solving very large convex optimization problems. We give a brief overview of ADMM technique in the next section.

4.1. **Alternating Direction Method of Multipliers for convex problems.** Alternating Direction Method of Multipliers (ADMM) [10][6][2] is a decomposition algorithm for solving separable convex optimization problems of the form:

$$\min \quad G_1(x) + G_2(y) \quad \text{subject to} \quad Ax - y = 0, \quad x \in \mathbb{R}^n, \quad y \in \mathbb{R}^m$$

where $A \in \mathbb{R}^{m \times n}$ and G_1 and G_2 are convex functions. The algorithm derivation is as follows. First, the augmented Lagrangian is formed:

$$L_\rho(x, y, z) = G_1(x) + G_2(y) + z^T(Ax - y) + \rho/2 \|Ax - y\|_2^2$$

where ρ is a positive constant known as the penalty parameter. ADMM iterations can then be written as:

$$x^{t+1} = \min_x \left\{ G_1(x) + z^{tT}Ax + \rho/2 \left\|Ax - y^t\right\|_2^2 \right\} \tag{3}$$

$$y^{t+1} = \min_y \left\{ G_2(y) - z^{tT}y + \rho/2 \left\|Ax^{t+1} - y\right\|_2^2 \right\} \tag{4}$$

$$z^{t+1} = z^t + \rho\left(Ax^{t+1} - y^{t+1}\right) \tag{5}$$

This is an iterative technique where t is the iteration counter, and the initial vectors y^0 and z^0 can be chosen arbitrarily. ADMM can be written in a different form (known as the scaled form) by combining the linear and quadratic terms of the Lagrangian:

$$z^T(Ax - y) + \rho/2 \|(Ax - y)\|_2^2 \quad = \quad \rho/2 \|(Ax - y) + (1/\rho)z\|_2^2 - 1/(2\rho) \|z\|_2^2$$

Now scaling the dual variable $p = (1/\rho)z$, the iterations of ADMM become:

$$x^{t+1} = \min_x \left\{ G_1(x) + \rho/2 \left\| Ax - y^t + p^t \right\|_2^2 \right\} \tag{6}$$

$$y^{t+1} = \min_y \left\{ G_2(y) + \rho/2 \left\| Ax^{t+1} - y + p^t \right\|_2^2 \right\} \tag{7}$$

$$p^{t+1} = p^t + \rho \left(Ax^{t+1} - y^{t+1} \right) \tag{8}$$

It has been argued [10] that ADMM is very slow to converge especially when high accuracy is desired. However, ADMM converges within a few iterations when moderate accuracy is desired. This can be particularly useful for many large scale problems similar to the one we consider in this paper.

Critical to the working and convergence of the ADMM method is the termination criterion. The primal and dual residuals are:

$$r_p^{t+1} = Ax^{t+1} - y^{t+1} \quad \text{(primal residual)}$$

$$r_d^{t+1} = \rho A(y^{t+1} - y^t) \quad \text{(dual residual)}$$

A reasonable termination criterion is when either the primal or the dual residuals are below some thresholds *i.e.*

$$\left\| r_p^{t+1} \right\|_2 \leq \epsilon_p \quad \text{and} \quad \left\| r_d^{t+1} \right\|_2 \leq \epsilon_d.$$

where ϵ_p and ϵ_d are the primary and dual feasibility tolerances. Using user-defined values for ϵ_1 and ϵ_2, these tolerances can be stated as,

$$\epsilon_p = \epsilon_1 \sqrt{m} + \epsilon_2 \max \left(\left\| Ax^{t+1} \right\|_2, \left\| y^{t+1} \right\|_2 \right)$$

$$\epsilon_d = \epsilon_1 \sqrt{n} + \epsilon_2 \left\| A^T p^{t+1} \right\|_2.$$

In the next section we discuss the ADMM update rules for the sparse inverse covariance estimation problem.

4.2. **Alternating Direction Method for sparse inverse kernel estimation.** We start with the prior assumption that the inverse kernel matrix K^{-1} is sparse. This is a reasonable assumption when studying climate data, because given a location *i.e.* any row of the inverse kernel matrix, there are few major locations which influence this location.

With such an assumption, the ADMM algorithm is as follows. Let K be the observed kernel matrix between the grid locations. For a moderate sized K, one can search over all sparsity patterns, since for a fixed sparsity pattern the log likelihood estimate of K is a tractable problem. However, this becomes very challenging for large K. One technique which has been used earlier for sparse covariance selection problem [1] is to minimize the negative log likelihood of $S = K^{-1}$ with respect to the observed data with a penalty term added to induce sparsity. This resulting objective function can be written as

$$\min \quad \mathbf{Tr}(KS) - \log det(S) + \lambda \left\| S \right\|_1$$

where $\left\| \cdot \right\|_1$ is the ℓ_1-norm or the sum of the absolute values of the entries of a matrix and λ is a constant which determines the amount of sparsity. Larger the value of λ, sparser is the solution S. The ADMM version of this problem can be written as follows:

$$\min \quad \mathbf{Tr}(KS) - \log det(S) + \lambda \left\| Y \right\|_1 \quad \text{subject to} \quad S - Y = 0$$

By constructing the augmented Lagrangian and using the derivations given in Section 4.1 for the scaled version of the problem, the ADMM updates for the above estimation problem are:

$$S^{t+1} = \min_x (\mathbf{Tr}(KS) - \log det(S) + \rho/2 \left\| S - Y^t + P^t \right\|_F) \tag{9}$$

$$Y^{t+1} = \min_y \left(\lambda \left\| Y \right\|_1 + \rho/2 \left\| S^{t+1} - Y + P^t \right\|_F \right) \tag{10}$$

$$P^{t+1} = P^t + \left(S^{t+1} - Y^{t+1} \right) \tag{11}$$

with $\|\cdot\|_F$ denoting the Frobenius norm of a matrix. These updates can be simplified further. Taking the derivative of Eqn. 9 and setting it to 0 we get,

$$\begin{aligned} & K - S^{-1} + \rho(S - Y^t + P^t) = 0 \\ \Rightarrow \quad & \rho S - S^{-1} = \rho(Y^t - P^t) - K \end{aligned}$$

Now let $Q\Lambda Q^T$ be the eigen decomposition of $\rho(Y^t - P^t) - K$. Therefore, continuing from the previous step,

$$\begin{aligned} & \rho S - S^{-1} = \rho(Y^t - P^t) - K \\ \Rightarrow \quad & \rho S - S^{-1} = Q\Lambda Q^T \\ \Rightarrow \quad & \rho Q^T S Q - Q^T S^{-1} Q = Q^T Q \Lambda Q^T Q \\ \Rightarrow \quad & \rho \widehat{S} - \widehat{S}^{-1} = \Lambda \quad [\text{since } Q^T Q = Q Q^T = I] \end{aligned} \tag{12}$$

where $\widehat{S} = Q^T S Q$. Solution to Eqn. 12 can easily be found noting that the right hand side is a diagonal matrix of the eigenvalues λ_i's. For each diagonal entry of $\widehat{S}_{ii}$, $\forall i = 1 : n$, we have

$$\rho \widehat{S}_{ii} - \widehat{S}_{ii}^{-1} = \lambda_i$$

which, using the formula of finding the roots of a quadratic equation is

$$\widehat{S}_{ii} = \frac{\lambda_i + \sqrt{\lambda_i^2 + 4\rho}}{2\rho}$$

Therefore, $S = Q\widehat{S}Q^T$ is the optimal value of the S minimization step.

Eqn. 10 can also be simplified further and can be written as the element-wise soft thresholding operation:

$$Y_{ij}^{t+1} = \Im_{\lambda/\rho}\left(S_{ij}^{t+1} + P_{ij}^t\right)$$

In the next section we describe the SPI-GP algorithm in details.

4.3. **SPI-GP: algorithm description.** The SPI-GP algorithm is based on the ADMM technique described in the earlier section. Alg. 1 presents the pseudo-code of the algorithm. The inputs are the kernel K, algorithm parameters λ and ρ, number of iterations $numIter$ and the error tolerances ϵ_1 and ϵ_2. The output of the algorithm is the estimated inverse of K in $S = K^{-1}$. The algorithm proceeds in an iterative fashion. In every iteration, an eigen decomposition is performed of the matrix

$$[Q \quad \Lambda] = \rho(Y^{t-1} - P^{t-1}) - K.$$

The eigenvalues Λ and eigenvectors Q are used to update the S variable. The Y-update is a soft thresholding operation of $\left(S^t + P^{t-1}\right)$ with threshold λ/ρ. Finally, the P-update is a linear dual variable update. Also during each iteration, the primal and dual residuals r_p and r_d are computed along with the corresponding error thresholds. Whenever the residuals become less than the error thresholds, the algorithm stops. The result is returned in the matrix S. In our experiments we have chosen $rho = 1$

Running time of ADMM: Since the algorithm requires eigen decomposition for every S update, and the Y and P updates are constant time operations, the runtime complexity is $O(mn^3)$, where m is the number of iterations and n is the size of the dataset (training points).

Convergence of ADMM: In order to ensure convergence of ADMM, two basic assumptions are necessary: (1) the functions G_1 and G_2 are closed, proper and convex, and (2) the unaugmented Lagrangian has a saddle point. Based on these two conditions, it can be shown that [2]:

- primal residual approaches 0 i.e. $r^t \to 0$ as $t \to \infty$
- the objective function approaches the optimal value
- dual variable P approaches feasibility

```
Input: $K$, $\rho$, $\lambda$, $numIter$, $\epsilon_1$, $\epsilon_2$
Output: $S = K^{-1}$
Initialization: $Y^1 = 0, P^1 = 0$
begin
    for $t$=2 to $numIter$ do
        $[Q \quad \Lambda] = \mathbf{evd}[\rho(Y^{t-1} - P^{t-1}) - K]$;
        for $i$=1 to $n$ do
            $\widehat{S}_{ii} = \frac{\lambda_i + \sqrt{\lambda_i^2 + 4\rho}}{2\rho}$;
        end
        $S^t = Q\widehat{S}Q^T$;
        $Y^t = \mathbf{softThreshold}[(S^t + P^{t-1}), \lambda/\rho]$;
        $P^t = P^{t-1} + (S^t - Y^t)$;
        $r_p = \|S^t - Y^t\|_F$;
        $r_d = \left\|-\rho(S^t - Y^{t-1})\right\|_F$;
        $\epsilon_p = \epsilon_1\sqrt{n} + \epsilon_2 \max(||S^t||_F, ||Y^t||_F)$;
        $\epsilon_d = \epsilon_1\sqrt{n} + \epsilon_2||\rho P^t||_F$;
        if $(r_p < \epsilon_p)$ AND $(r_d < \epsilon_d)$ then
            break;
        end
    end
end
```

Algorithm 1: SPI-GP: ADMM for Sparse Kernel Inversion

In practice however, ADMM may be slow to converge. This type of algorithms, are therefore, more useful when moderate accuracy is necessary within a relatively few iterations. Although this algorithm is slow and sometimes has convergence issues, it is the only method that is amenable to parallel computing which is essential for many large data sets that do not fit in the main memory of a single machine.

4.4. **SPI-GP: distributed implementation.** As we have discussed earlier, ADMM is amenable to distributed computation in a network of machines. This becomes particularly important when the data does not fit into the memory of one machine. This form of ADMM is known as consensus optimization. In this form, the objective function G_1 needs to be decomposable across ℓ nodes $M_1, \ldots, M_\ell$ as follows:

$$\min \quad \textstyle\sum_{i=1}^{\ell} G_1(x_i) + G_2(y) \text{ subject to } \quad Ax_i - y = 0, \quad x_i \in \mathbb{R}^n, \quad y \in \mathbb{R}^m$$

where x_i is the i-th block of data and is stored at machine M_i. The solution to this optimization is the same as given in Section 4.1. The update rules can be written as,

$$\begin{aligned}
x_i^{t+1} &= \min_{x_i} \left\{ G_1(x_i) + z^{tT} A x_i + \rho/2 \left\| A x_i - y^t \right\|_2^2 \right\} \\
y^{t+1} &= \min_{y} \left\{ G_2(y) + \sum_{i=1}^{\ell} \left(-z_i^{tT} y + \rho/2 \left\| A x_i^{t+1} - y \right\|_2^2 \right) \right\} \\
z_i^{t+1} &= z_i^t + \rho \left(A x_i^{t+1} - y^{t+1} \right)
\end{aligned}$$

Unfortunately, the above method cannot be applied for the optimization of the inverse covariance matrix in our case. This is because $\log det(S)$ is not a decomposable function.

Therefore, to solve this problem for large kernel matrices, we use the ScaLAPACK routine of Matlab. It allows the kernel matrix to be distributed across different machines, but still compute the eigen decomposition correctly. For a Matlab implementation, this is done using the co-distributed array data structure and an overloaded *eig* function. It should be noted here that this method *does* *not* attempt to speed up the GPR process. Instead, it makes GPR possible for extremely large data sets where the entire kernel matrix cannot be loaded in the main memory due to size limitations.

5. Experimental results

For the performance study of SPI-GP, the experimental results are reported on a synthetic multivariate Gaussian distribution data and a real life climate domain data set. For generating the multivariate Gaussian, we fix the number of dimensions and samples. We then generate a sparse inverse covariance matrix with all zeros and ones along the diagonal. We randomly insert 1 at certain locations in our inverse covariance. We make this inverse matrix symmetric and positive definite (by making the min eigenvalue positive). Finally we invert this matrix and draw Gaussian samples with zero mean which becomes our covariance matrix. Using this data set we demonstrate the scalability of the distributed SPI-GP method on a cluster of computing nodes.

Our second data set is a historical climate domain data set which consists of NCEP/NCAR features available at `http://www.cdc.noaa.gov/data/gridded/data.ncep.reanalysis.html` [12] and cross-matched normalized difference vegetation index (NDVI) data (NDVI) from the National Oceanic and Atmospheric Administrations Advanced Very High Resolution Radiometer (NOAA/AVHRR). The climate variables used in this study include pressure (hg1000 and hg500), sea surface temperature (sst), Temperature (temp) and precipitation (pre). We use this data set to demonstrate a Gaussian Process regression task where our goal is to take as inputs the first five variables and predict/model precipitation (output) using our SPI-GP method. We have used data from years 1982 - 2002 (21 years). Each variable is observed at a 0.5° resolution over the entire grid. The data used here are composites of observations over a month. Thus there are 360×720=259200 values for each variable vectorized and stored as a single row corresponding to a time point (a month). Therefore, each variable has $12 \times 21 = 252$ rows in the data set, each having 259200 columns. Note that some variables are observed only in land while others only in ocean. For any variable, the locations which do not contain any meaningful data has a fill value of -9999.0.

If we want to use all five variables (hg1000, hg500, sst, temp, ndvi) for predicting precipitation, then we have to create a GPR model which takes the five variables as input and precipitation as the output. Since there are missing values for each variable, the locations where all values are present are the coastlines of the continents (only approximately 8500 points). This means that if we build a model based on only these points, the other data points cannot be used in the model. Instead we use a multiple kernel approach as follows. Let K_{hg1000}, K_{hg500}, K_{ndvi}, K_{sst}, and K_{temp} be the kernels computed from each of the 5 input variables separately after removing the fill values. We create a global training kernel as,

$$K_{global} = K_{hg1000} + K_{hg500} + K_{sst} + K_{temp} + K_{ndvi}$$

Similarly we create the test kernel as,

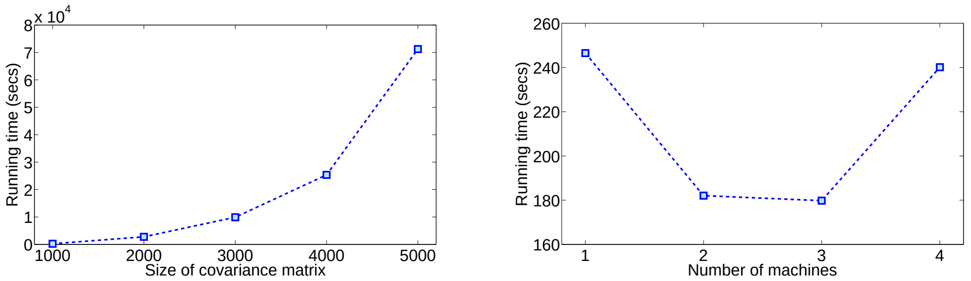
$$K^*_{global} = K^*_{hg1000} + K^*_{hg500} + K^*_{sst} + K^*_{temp} + K^*_{ndvi}$$

In both these global kernels, we normalize the values by making their range between 0 and 1. We then use the following two GPR equations

$$\widehat{\mathbf{y}}^* = K^*_{global}(\sigma^2 I + K_{global})^{-1}\mathbf{y} \tag{13}$$

$$C = K^{**}_{global} - K^*_{global}(\sigma^2 I + K_{global})^{-1}K^{*T}_{global} \tag{14}$$

using the kernels just computed by combining the individual kernels. By construction, K_{global} is a matrix on the entire set of grid locations ($n \times n$). K^*_{global} is a test kernel of size $m \times n$, where m is the number of test locations. y is the training output of size $n \times 1$. As a result $\widehat{y}^*$ becomes of size $m \times 1$. One issue is in using the entire y vector. For our prediction problem, this corresponds to precipitation and hence has only values on land. So when we use the y vector for the entire world's data, it contains missing values of -9999.0 (about 40% of the total size of y). To circumvent this problem, we replace the fill values with an average value of the feature y.



(a) Running time in seconds for different sizes of the training data on 4 processors

(b) Running time in seconds for different number of processors on a 1000 × 1000 matrix

Figure 2. Scalability study of SPI-GP on synthetic data

5.1. **Study 1: Scalability study on synthetic data.** In this study we report the scalability of the SPI-GP algorithm. The metric we use is running time (in seconds). We report results from two different experiments. In our first experiment we fix the number of cores on which we run our experiment and vary the size of the training data. Figure 2 (left) shows the result. We used the Matlab Parallel Computing toolbox and a local scheduler for multicore architecture. For this experiment we chose 4 processors on a single CPU to simulate the distributed computing environment. We experimented with five different sizes of the covariance matrix starting from 1000×10000 to 5000×50000 and notice that the growth in the running time is less than cubic in spite of the eigen decomposition step. This is due to the distributed eig function usage which makes the method complexity $\mathrm{O}(nr^2)$ where r is the chunk (rank) of the matrix for the covariance matrix partition in any one of the processors. Figure 2 (right) reports the results of running SPI-GP on a 1000×10000 matrix on a varying number of processors starting from 1 to 4. The result is counter-intuitive since we see that a single processor takes the highest time while there is no clear trend in the time as we increase the number of processors, keeping the data fixed. This is because there is considerable overhead in distributing a job over the parallel computing framework and there is an optimal number of processors for a fixed partitioning of the data. The performance degrades with deviation from the optimal.

5.2. **Study 2: Precipitation prediction in the Indian subcontinent.** In the climate study, we observe which geographical regions are most similar to the precipitation pattern of India. We want to identify these points and study how these points change over a time period of 20 years. Since all climatic connections change very slowly with time, we construct the relevant network connections for Indian precipitation every 5 years. Fig. 4 shows the results. Each plot in Fig. 4 is for the average of one year's data. The variable shown in the figures is precipitation. The black markers are the locations in India. The yellow markers indicate the the top 10 areas which influence India. These are the points which have the highest values in the estimated inverse kernel matrix corresponding to test points for India. As Figure 4 shows, there are certain regions which remain similar to our test set for the entire period of 20 years, while others have a more disparate pattern. Some locations which show consistent influence pattern include the west coast of South America, west coast of Africa, and east coast of Australia. Some less consistent locations include areas in China. To illustrate how this method can be used for studying climate networks, we represent a portion of the precipitation-based inverse covariance matrix as a network. As can be seen in Figure 3, the true inverse covariance is difficult to understand or interpret, given the huge amount of network connections for any particular node in the graph. The reference node in this study is denoted in red in both the left and right subfigures in Figure 3. The right figure, which a sparse variant of the same graph shows only the important connections to the colored node, and enhances interpretability of learnt models like in Gaussian Process regression.

(a) Network representing a sub-matrix of the inverse covariance matrix

(b) Network representing a sub-matrix of the sparse inverse covariance matrix estimated using SPI-GP

Figure 3. Interpretability of sparse inverse covariance matrix

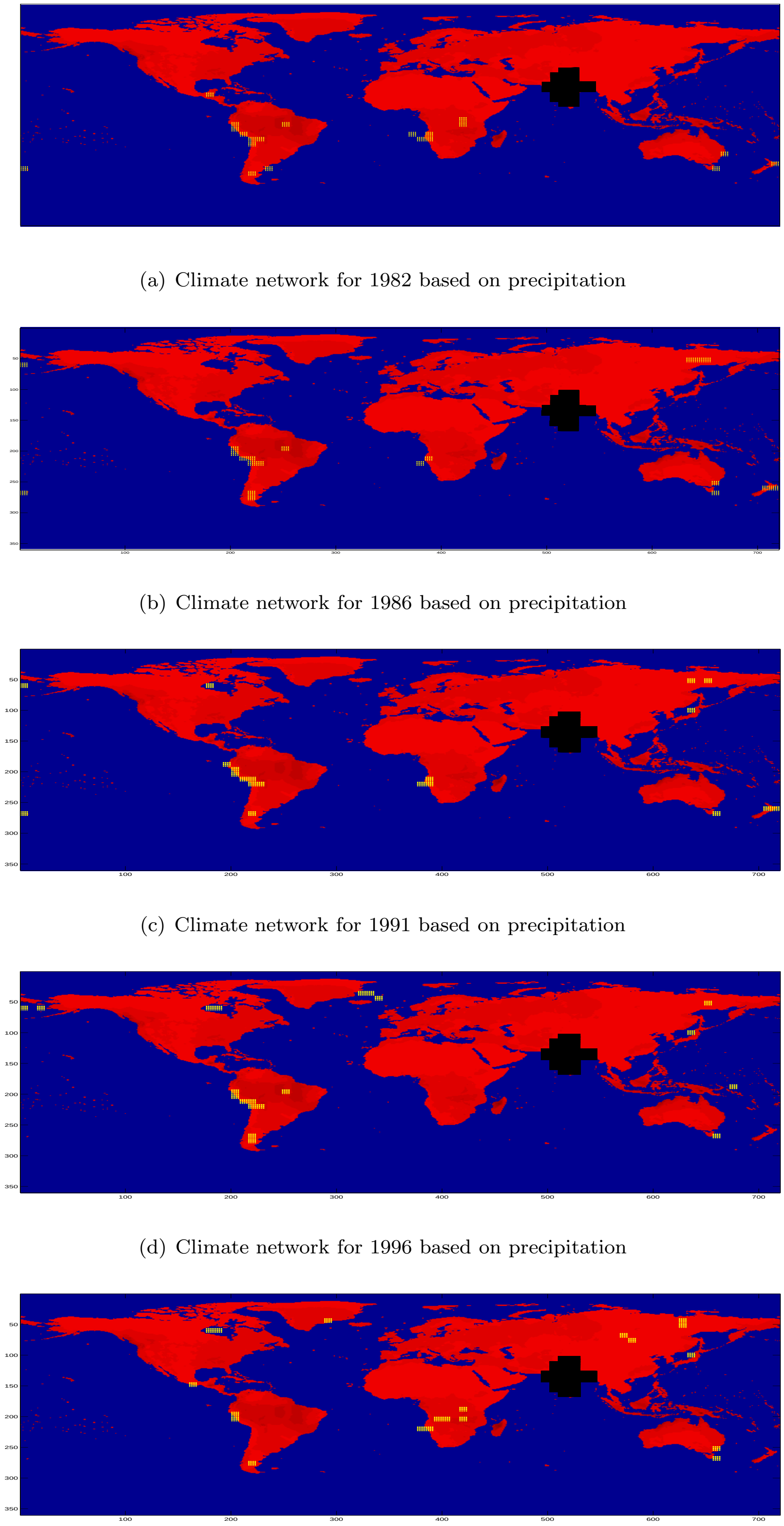
As we will observe in section 5.3, this regression problem performs poorly due to the immense amount of missing data in the different modalities used to predict precipitation. Therefore, for demonstrating the fact, that the poor regression results are only due to the nature of data available, and not due to the technique discussed here, we study a a different regression problem where we want to predict the precipitation in the Indian subcontinent based on only precipitation data from four weeks in advance. In this study, we use only precipitation data to predict precipitation for a delay of 1 month. This study is also performed for over a 20 year period at intervals of every 5 years. For every year we study the prediction problem quarterly.

5.3. **NMSE of SPI-GP.** If a set of points are very similar to the points representing rainfall in the Indian subcontinent, then it is intuitive that those points should be very good predictor of precipitation in India. Our next study tries to verify this intuition. For this, we choose the top k locations of the world that are most similar to the precipitation in the Indian subcontinent for each of the years 1982, 1986, 1990, 1994, and 1998 and build GPR models by taking only this subset as the training examples. We test on year 2002. As a baseline comparison, we train a separate GPR on the entire world's data (Full-GP). For both these methods, we use the same locations of India as test sets. We build these two GPR's separately for each of the five years mentioned before. The first row of Table 5.3 shows the normalized mean squared error (NMSE) values for these two GPR methods for each of the five years, where NMSE is defined as

$$NMSE = \frac{\sum_{i=1}^{n}(\widehat{y}_i^* - y_i)^2}{n \times var(\widehat{\mathbf{y}}^*)}.$$

The value of k is chosen to be $n/2$ where n is kernel dimension. For this study, for each of the five years, the NMSE value for the GPR model of top k values from SPI-GP is better than the Full-GP. This happens because the most similar points capture more information and less of noise as has been verified earlier in [4]. However, as it can be noted the improvement in NMSE observed is not significant. Not only that, even for the improvement that is observed, the NMSE values are quite high (approximately 1). Now, a value of 1 for NMSE implies that the prediction is equal to the mean of the target. This explains the observed NMSE in our experiments. Since approximately 40% of the target data used in our experiments were actually fill values and were replaced by the mean of the target. Therefore, the NMSE that we see is largely an artifact of the data preprocessing for this data set since the mean-based smoothing technique applied here may have failed to capture the dynamics in the data.

To verify that the high NMSE values are not an artifact of the technique, but the data, we perform similar experiments for the precipitation based regression study. The second row of Table 5.3 shows the NMSE values when we predict rainfall for August of 2002 based on rainfall in July for each of the years 1982, 1986, 1990, 1994, and 1998. We can notice that the NMSE values are much lower



(a) Climate network for 1982 based on precipitation

(b) Climate network for 1986 based on precipitation

(c) Climate network for 1991 based on precipitation

(d) Climate network for 1996 based on precipitation

(e) Climate network for 2001 based on precipitation

FIGURE 4. Evolution of the climate network over 20 years based on precipitation data.

		1982	1986	1990	1994	1998
Regression using all variables	Full-GP	1.085	1.218	1.115	1.883	1.138
	SPI-GP	0.902	0.811	1.05	1.072	0.979
Regression using precipitation	Full-GP	0.695	0.664	0.611	0.651	0.669
	SPI-GP	0.6912	0.664	0.605	0.650	0.667

Table 1. NMSE of GPR for 2002 when entire world's data is used (Full-GP) vs. top few similar points in SPI-GP. For the first regression scenario, each column shows the NMSE for that year. For the second scenario, each column shows the NMSE for prediction of rainfall in August for that year.

Training years	Training months							
	1		4		7		10	
	Full-GP	SPI-GP	Full-GP	SPI-GP	Full-GP	SPI-GP	Full-GP	SPI-GP
1982	0.237	0.283	0.454	0.439	0.426	0.426	0.371	0.361
1983	0.258	0.292	0.492	0.492	0.658	0.658	0.374	0.374
1984	0.261	0.273	0.451	0.451	0.818	0.819	0.374	0.368
1985	0.196	0.208	0.475	0.450	0.396	0.396	0.385	0.385

Table 2. NMSE of GPR for 1986 when entire world's data is used (Full-GP) vs. top few similar points in SPI-GP.

Training years	Training months							
	1		4		7		10	
	Full-GP	SPI-GP	Full-GP	SPI-GP	Full-GP	SPI-GP	Full-GP	SPI-GP
1982	0.311	0.293	0.554	0.563	0.706	0.706	1.23	1.22
1986	0.325	0.295	0.587	0.595	0.81	0.809	1.301	1.3
1991	0.281	0.278	0.564	0.586	0.782	0.781	1.15	1.15

Table 3. NMSE of GPR for 1996 when entire world's data is used (Full-GP) vs. top few similar points in SPI-GP.

compared to the first study. However, it should be noted that the precipitation prediction problem that we are studying is a difficult one since the data does not have reasonably high predictability. The linear correlations for different data subsets and different test sets can vary from -0.2 (very poor) to 0.88 (high correlation) accounting for the high variability in the NMSE values for the different test scenarios. Tables 5.3 and 5.3 document the NMSE values for predicting precipitation in India for months February, May, August and November for the years 1986 and 1996 respectively. NMSE values in the table range from as low as .19 to as high as 1.3 indicating the difficulty level of different prediction scenarios. For example year 1986 has reasonably good predictability and has lower variation in the NMSE values thatn year 1996. Although February, May and August have quite low NMSE values for 1996, the month of November does not have that since the prediction is working as poorly as random for the different training years. The year 2002 is worse than 1986 in terms of average predictability, but data is more consistent across the different training years.

6. Conclusion

In this paper we discuss a method for sparse inverse Gaussian Process regression that allows us to compute a parsimonious model while preserving the interpretability of the sparsity structure in the data. We discuss how the inverse kernel matrix used in Gaussian Process prediction gives valuable information about the regression model and then adapt the inverse covariance estimation from

Gaussian graphical models to estimate the Gaussian kernel. We solve the optimization problem using the alternating direction method of multipliers that is amenable to parallel computation. This sparsity exploiting GPR technique achieves two goals: (i)it provides valuable insight into the regression model and (ii)it allows for parallelization so that the entire kernel matrix need not be loaded into a single main memory, thereby removing the size related constraints plaguing large scale analysis. We perform experiments on historical climate data of 20 years. The climate network study shows evolution of the most influential points over time for predicting precipitation in the Indian subcontinent. The NMSEs reported are relatively high due to the mean-based smoothing adopted in the preprocessing. For future work, we plan to pursue other spatial smoothing processes such as the ones proposed by Cressie and Wikle [3]. We also want to pursue teleconnection study using different climate data for specific climate scenarios.

Acknowledgements

This work was supported by the Systemwide Safety and Assurance Technologies project of Aviation Safety Program under the NASA Aeronautics Mission Directorate.

References

[1] O. Banerjee, L. Ghaoui, A. d'Aspremont, and G. Natsoulis. Convex optimization techniques for fitting sparse Gaussian graphical models. In *Proceedings ICML-06*, pages 89–96, 2006.

[2] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends in Machine Learning*, 2011.

[3] N. Cressie and C. K. Wikle. *Statistics for Spatio-Temporal Data*. Wiley, 2011.

[4] K. Das and A. Srivastava. Block-GP: Scalable Gaussian Process Regression for Multimodal Data. In *The 10th IEEE International Conference on Data Mining, ICDM 2010*, pages 791–796, 2010.

[5] A. P. Dempster. Covariance Selection. *Biometrics*, 28:157–175, 1972.

[6] J. Eckstein and D. Bertsekas. An alternating direction method for linear programming. Technical Report LIDS-P ; 1967, MIT, 1990.

[7] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Annals of Statistics*, 32(2):407–499, 2004.

[8] L. Foster, A. Waagen, N. Aijaz, M. Hurley, A. Luis, J. Rinsky, C. Satyavolu, M. Way, P. Gazis, and A. Srivastava. Stable and Efficient Gaussian Process Calculations. *JMLR*, 10:857–882, 2009.

[9] J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics Journal*, 9(3):432–441, 2008.

[10] M. Fukushima. Application of the alternating direction method of multipliers to separable convex programming problems. *Computational Optimization and Applications*, 1:93–111, 1992.

[11] M. H. Glantz, R. W. Katz, and N. Nicholls. *Teleconnections linking worldwide climate anomalies : scientific basis and societal impact*. Cambridge University Press, 1991.

[12] E. Kalnay, M. Kanamitsu, R. Kistler, W. Collins, D. Deaven, L. Candin, S. Saha, G. White, J. Woollen, Y. Zhu, M. Chelliah, W. Ebisuzaki, W. Higgins, J. Janowiak, K. C. Mot, C. Ropelewski, A. Leetmaa, R. Reynolds, R. Jenne, and D. Joseph. The NCEP/NCAR 40-Year Reanalysis Project. *B. Am Metrolo. Soc.*, 77(3):437–471, 1996.

[13] N. Meinshausen, P. Bhlmann, and E. Zrich. High Dimensional Graphs and Variable Selection with the Lasso. *Annals of Statistics*, 34:1436–1462, 2006.

[14] J. Quiñonero-Candela and C. E. Rasmussen. A Unifying View of Sparse Approximate Gaussian Process Regression. *JMLR*, 6:1939–1959, 2005.

[15] C. E. Rasmussen. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[16] A. J. Smola and P. Bartlett. Sparse Greedy Gaussian Process Regression. In *Proc. of NIPS 13*, pages 619–625, 2000.

[17] E. Snelson and Z. Ghahramani. Sparse gaussian processes using pseudo-inputs. In *Proceedings of NIPS 18*, 2005.

[18] K. Steinhaeuser, N. Chawla, and A. Ganguly. An exploration of climate data using complex networks. *SIGKDD Explorations Newsletter*, 12:25–32, November 2010.

[19] R. Tibshirani. Regression Shrinkage and Selection Via the Lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288, 1994.

[20] V. Tresp. The generalized bayesian committee machine. In *Proc. of KDD*, pages 130–139, 2000.

[21] G. Wahba. *Spline Models for Observational Data*. SIAM, 1990.

[22] F. Yan and Y. Qi. Sparse Gaussian Process Regression via ℓ_1 Penalization. In *Proceedings of ICML-10*, pages 1183–1190, 2010.

A NOVEL TIME SERIES BASED APPROACH TO DETECT GRADUAL VEGETATION CHANGES IN FORESTS

YASHU CHAMBER*, ASHISH GARG*, VARUN MITHAL*, IVAN BRUGERE*, MICHAEL LAU*, VIKRANT KRISHNA*, SHYAM BORIAH*, MICHAEL STEINBACH*, VIPIN KUMAR*, CHRIS POTTER** AND STEVE KLOOSTER**

Abstract. It is well-known that forests play a vital role in maintaining biodiversity and the health of ecosystems across the Earth. This important ecological resource is under threat from both anthropogenic and biogenic pressures, ranging from insect infestations to commercial logging. Detecting, quantifying and reporting the magnitude of forest degradation are therefore critical to efforts towards minimizing the loss of one of Earth's most crucial resources. Traditional approaches that use image-based comparison for detecting forest degradation are frequently domain- or region-specific, which require expensive training, and are thus not suited for application at global scale. More recently, time series based change detection methods applied on remote sensing datasets have gained much attention because of their scalability, accuracy, and monitoring capability at frequent regular intervals. In this paper, we propose a novel approach to identify regions where forest degradation occurs gradually. The proposed approach complements traditional domain- and region-specific approaches by providing information on where degradation is occurring, and during what time, at a global scale.

1. Introduction

Forests play a vital role in maintaining biodiversity and the health of ecosystems across the Earth. However, this important ecological resource is under threat of degradation by both anthropogenic and biogenic pressures. Forest degradation occurs due to a number of different causes ranging from insect infestations to logging. Such reduction in forest cover not only has implications on the global carbon cycle, but also causes adverse effects on the ecosystem which are often realized by decrease in biodiversity, increase in the frequency of floods, droughts, changes in rainfall patterns, etc. [15, 10, 16]. Thus, detecting, quantifying and reporting the magnitude of forest degradation is critical to efforts towards minimizing this loss of one of Earth's most crucial resources.

Remote sensing offers rich data sets that are very well-suited for monitoring forests around the globe, in a regular fashion across time. A large variety of techniques and tools have been developed for detecting changes in forest cover, and more generally land cover [4, 11]. However, detecting gradual forest degradation (as opposed to abrupt changes caused by fires etc.) is particularly challenging because the reduction in forest cover occurs very slowly and the amount of reduction observed across time is small compared to natural variations.

Traditional approaches that use image based comparison for detecting forest degradation are frequently domain-specific or region-specific [5] which require expensive training, and are thus not suited for application at global scale. More recently, time series based methods applied on remote sensing datasets have gained much attention to detect deforestation because of their scalability, accuracy, and forest monitoring capability at frequent intervals. However, even most of the current time series based approaches for detecting vegetation loss in forests are aimed at only certain types of changes (e.g. due to fires), which are characterized by sudden and severe vegetation loss [12].

A number of approaches have been proposed for identifying gradual changes in a time series. Kucera et al. [9] describe the use of the well-known CUSUM technique for land cover change detection. CUSUM follows a simple approach of determining deviation in the values of a time series

* University of Minnesota, <chamber,ashish,mithal,ivan,mwlau,krishna,sboriah,steinbac,kumar>@cs.umn.edu
**NASA Ames Research Center, chris.potter@nasa.gov, sklooster@gaia.arc.nasa.gov.

from an expected value, and the change score, giving the magnitude of change, is determined as the maximum cumulative deviation. Another approach presented recently by Verbesselt et al. [17], Breaks for Additive Seasonal and Trend (BFAST), decomposes a time series into trend, seasonal and residual components. The time series is divided into segments such that intra-segment trend is constant, while inter-segment trends are dissimilar. A trend breakpoint is associated with segment boundaries. The seasonal component is handled in a similar fashion.

In this paper, we present a novel approach to identify regions where forest degradation is occurring gradually (either due to biogenic or anthropogenic causes). The approach is robust, scalable and easy to apply across different regions and vegetation types. The proposed method represents an adaptation of CUSUM for the problem of gradual change detection. While CUSUM only identifies a time series as changed or not, the proposed approach also identifies the period of change, in addition to having a considerable improvement in performance.

We begin by describing the underlying remote sensing data and preprocessing procedure in Section 2. Sections 3 and 4 discusses the concepts behind the development of the new approach. We formally present our method in Section 5 followed by a discussion in Section 6. We then evaluate the performance of the proposed approach in Section 7 using independent validation data sets in two regions of the world where the degradation has entirely different causes. Finally, in section 8, we comparatively evaluate the proposed approach, CUSUM and BFAST for detecting gradual changes.

2. Data and Preprocessing

The time series data set used for this study is the Enhanced Vegetation Index (EVI), which is a product based on measurements taken from MODIS instrument on NASA's Terra satellite, and is available for download from the Land Processes Distributed Active Archive Center (LPDAAC) [1]. EVI essentially measures the "greenness" signal as a proxy for the amount of vegetation at a location. The spatial resolution of the dataset is 250 meters and the temporal resolution is 16 days (23 time steps per year), and covers the time period from from February 2000 to the present. The range of EVI is 0 to 1, where 0 indicates no vegetation and 1 indicates vegetation saturation. Figure 1 shows an example of an EVI time series.

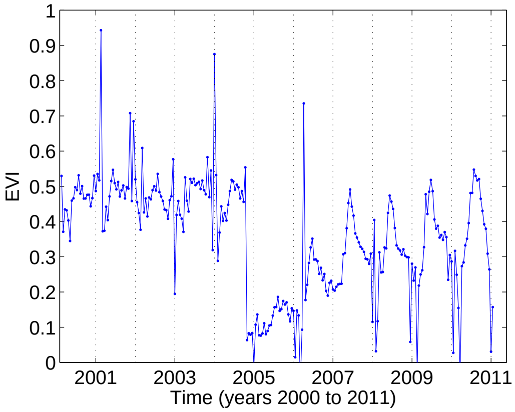


Figure 1. Example of an EVI time series (with noise in observations).

Remote sensing data sets are frequently subject to contamination due to clouds, haze, pixel geometry and other factors. We preprocess the EVI time series data set in order to remove undesired fluctuations in EVI (such as the sharp increases in Figure 1). This improves the efficiency of identifying signatures of interest. For smoothing purposes, we have used the Savitzky-Golay smoothing filter [13], which uses two parameters: polynomial *degree* desired for smoothing, and *frame size*. The smoothing filter fits a polynomial function of the indicated degree over a window equal to the frame size over each time step, the current time step being at the center of the window; the EVI value of the current time step is then replaced with the polynomial fit.

3. Detecting forest degradation: problem formulation and a CUSUM approach

A reduction in forest cover is often reflected as a decrease in the EVI value. In fact, many existing schemes compute difference in EVI (or related indices) between different years to identify changes. However, the values of vegetation indices such as EVI can have a high degree of variability due to seasonality (e.g. vegetation is greener in the summer than in the winter in the temperate zones), as well as due to natural variation in vegetation growth caused by environmental factors such as temperature and precipitation.

Most existing methods have handled seasonality by comparing vegetation index values at (or around) the same date in different years. Natural variability is much harder to handle since it can result in too many false positives. The problem becomes even more acute for many non-forest covers such as shrubs, since natural variability tends to be much larger in these cases. Although our focus is on identification of degradation in forests, it is not possible to completely exclude non-forests from any study due to the unavailability of highly precise forest maps [6].

Given an EVI time series dataset, we are interested in identifying time series such as the one shown in Figure 2, where there is a perceptible decrease in the signal, along with determining the approximate period of decrease (as shown by the vertical lines in Figure 2). Identifying a time series with a gradual decrease in vegetation is challenging due to a number of reasons: distinguishing vegetation loss from natural seasonal variations; differentiating between a spurious decrease due to noise or environmental factors and a genuine decrease from degradation on ground; correctly determining the period of decrease (start and end time steps) especially when there is a high degree of variability in the time series. There could also be a phenological change during the decrease period or across the decrease, and the algorithm must be able to handle such cases and extract the decrease period appropriately.

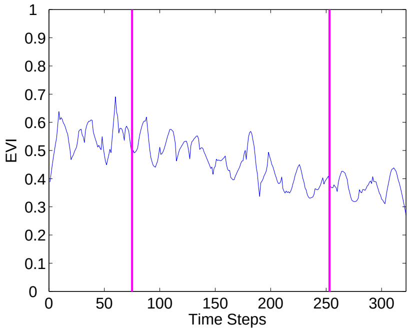


FIGURE 2. Example of a decreasing time series. Vertical lines enclose a gradually decreasing segment.

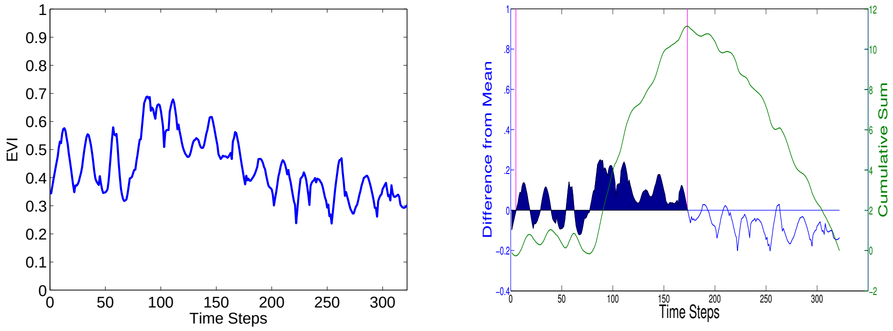
3.1. **Notation.** Table 1 defines notation used in this paper.

n	The number of time steps in a time series.
S	The number of time steps corresponding to one year of data (we also call this the season length). For biweekly data $S = 23$, and $S = 12$ for monthly data.
t_1	First time step.
t_i	ith time step.
v_{t_i}	Data value at the time step t_i
$v_{t_i}...v_{t_j}$	All values between time steps t_i and t_j.
T	A sample time series $= v_1 v_2 v_3 ... v_i ... v_n$
$v_{t_i...t_j}$	$mean(v_{t_i}...v_{t_j})$
Δ_i	$v_{t_{i-S+1}...t_i} - v_{t_{i+1}...t_{i+S}}$
Δ-series	$\Delta_S \Delta_{S+1} \Delta_{S+2} ... \Delta_{n-S}$

TABLE 1. Notation for time series change detection.

3.2. **CUSUM Method for detecting decreasing time series.** CUSUM is a well-known change detection algorithm that was originally developed in the domain of process control. It is one of the earliest change detection algorithms developed, proposed by Page [14]. One of the defining features of CUSUM is its ability to detect *small* and *gradual* changes in the process. The basic CUSUM scheme has an expected value μ for the process. It then compares the *deviation* of every observation to the expected value, and maintains a running *statistic* (the cumulative sum) CS of deviations from the expected value. If there is no change in the process, CS is expected to be approximately 0; if CS exceeds a user-defined threshold at any time step, the time series is flagged as changed.

There are multiple ways in which a change score can be assigned to a time series, the simplest of which is to use $\max\{CS_1, CS_2, \ldots, CS_n\}$. However, this score can be sensitive to noise [3]. Kucera



(a) A sample time series with decreasing period between time steps 100 and 210.

(b) Corresponding difference series, D, (blue) and cumulative sum series, Q, (green).

Figure 3. Measures computed by CUSUM.

et al. [9] developed a CUSUM approach for land cover change detection which uses a more robust technique to compute the change score. Specifically, a bootstrap procedure is used to determine the confidence of CS by determining the degree to which such a score can occur by chance. The bootstrap procedure involves randomly permuting the input time series to obtain a distribution of change scores $\mathcal{R}$ (CUSUM is run on each randomization). The confidence of the drop is determined by the relative frequency of CS being greater than the randomized distribution, i.e. $\frac{|CS>\mathcal{R}|}{|\mathcal{R}|}$.

Kucera et al. [9] take the expected value μ as the mean of the entire time series. Other measures may also be used to compute μ such as the value of the first time step, or the mean of the first S values. The advantage of using the mean value across a periodic cycle over a single time step is that the mean value is independent of the fluctuations in a time period (or seasonal variation in case of the MODIS EVI time series).

We illustrate some drawbacks of the scoring mechanism of CUSUM described above:
(1) *Change point of drop and period of decrease not identified.* This method only identifies a score corresponding to the maximum deviation in cumulative sum time series, and does not give the period of change. Figure 3 shows the scoring process using CUSUM. It identifies the maximum value in the cumulative sum series as the score. Thus, no change point of drop or period of drop is identified.
(2) *Computed score may not be associated with the decreasing period.* Computed score is the maximum cumulative deviation from the expected value, which may or may not depict the amount of EVI lost during the decrease period. This can again be noticed from Figure 3b.

4. Adapting CUSUM for Gradual Degradation

In the original CUSUM approach, the expected value is always fixed in a time series regardless of the way it is computed. In this paper, we propose a different strategy: If we take the expected value at any time step t_{i+1} as the value at time step t_i, then the deviation of values at each time step from its expected value would give the amount of drop or rise from its previous value. However, such a model would be dependent on the intra-periodic variation and the resulting deviation could be due to the natural periodicity of the time series. In order to make this process independent of periodicity, averaging over a periodic cycle can be used. Therefore, instead we take the mean value of the current periodic cycle as the expected mean value for the next periodic cycle. The deviation between mean values of successive periodic cycles is also equivalent to the drop in EVI across a time step t_i that marks the boundary between these periodic cycles, i.e. the one that ends at t_i, and the other that begins at t_{i+1}. We refer to this type of differencing (computing drop from previous periodic cycle in succession) as $Successive\ Differencing\ (SD)$, which is different from computing the deviation from a fixed expected value as done in CUSUM, which we refer to as $Fixed\ Differencing\ (FD)$.

If T is a given time series, n is the number of time steps in T, and S is the periodicity of T, we can define SD and FD methods as:

FD: $\Delta_i^c = v_{t_i} - \mu \quad \forall \quad i \in 1 \cdots n$

SD: $\Delta_i^s = \Delta_i \quad \forall \quad i \in S \cdots n - S \qquad (Refer \quad to \quad Table\,1 \quad for \quad notation)$

For MODIS EVI time series, Δ_i^s's are computed as difference between the mean values of two successive years (two consecutive sets of 23 values since $S = 23$).

FD gives deviation in values relative to the fixed expected value, while SD gives the drop relative to the previous year. Also, in the first equation, individual data values are used for differencing while in the second equation, averaged value over a seasonal cycle is being subtracted. Computing drop from a previous value in succession can provide trend information in a time series, which is what successive differencing does. Also, subtracting averaged values instead of individual values make the trend information more robust to seasonal variations and noisy outliers. On the other hand, Δ_i^c's fluctuates with the seasonal variations, even when there is no decrease in vegetation. Also, it does not provide trend information which is vital for identifying decreasing period in a time series.

Using Successive Differencing. As we have seen above, successive differencing using mean value of annual segments can be used to determine trends in a time series. Therefore, we use SD instead of FD for our approach. Below, we mention some possibilities in which SD could be used:

Consider a method for detecting gradual decrease that tries to identify the window in a time series that has the largest drop: given a time series, identify two years, i.e. two sets of 23 consecutive time steps, y_1 and y_2, such that the difference between the mean EVI of y_1 and y_2 is maximum in the time series. This is similar to identifying the window where the sum of Δ_i^s is maximum. This method will work well for consecutively decreasing time series. However, there are some disadvantages of this method when applied to a time series with high variation. The primary disadvantage is that this method loses information about the time steps in between y_1 and y_2. For example, given the time series shown in Figure 4, this method would identify the decrease as having occurred between years 2 and 11 even though it is clear that the time series increased significantly after year 7. Specifically, if the time series rises and falls in between then such a time series is highly variable and it should not be considered as changed. Another disadvantage is that if there are large spikes in a year due to noise that distorts the mean EVI for that year in an otherwise stable time series, this time series will be given a high score (drop from y_1 to y_2) by this method, even though this change is spurious.

To overcome drawbacks of the above method, yet another method to detect gradual decrease could be to compute the difference between successive yearly sets (Δ_i^s), and determine the longest continuous window of positive Δ_i^s. This method again has a major disadvantage that if there is a spurious rise in time series due to noise, the drop window will fall short of that false rise and thus could be determined much smaller than it actually is (e.g. for the time series in Figure 4, this method would incorrectly detect end of degradation in year 5).

Building on the concepts described in this section, we propose a novel time series change detection method, *Persistent-*Δ *Approach*, or $PDELTA$. It uses successive differencing as the base to compute Δ_i's. The key property of successive differencing is that as long as there is a decrease in the time series from one year to the next, Δ_i would be positive. If the decrease is at an almost constant rate, the Δ_i would be almost constant. As soon as Δ_i becomes zero, it means that there has been no vegetation change from past year to the present year. But it could be too soon to say that the change in vegetation has stopped since this could be due to some noisy time steps and it's possible that after very few time steps, Δ_i's become positive and stay positive for a couple of years or more. Thus the change didn't really stop, but continued after a short time. Therefore, the primary objective of PDELTA is to determine the window of *maximum reliable drop*. This method tolerates natural variation which may cause small increases in individual years during an extended period of degradation. For example, in Figure 4, the technique correctly identifies the period between years 2 and 6 when degradation has occurred since it accounts for the perturbations in the intervening years. However, if this rise in the time series violates a *reliability condition*, the approach differentiates it from the natural variation and does not consider this in the changed phase. The next section describes the PDELTA method in detail.

5. Description of PDELTA

As mentioned in the previous section, the main objective of the PDELTA approach is to identify the window of maximum reliable drop in a time series. It need not be a continuous drop, and some amount of intermittent rise can be allowed as long as a decreasing trend is persistent. The amount of intermittent rise allowed is controlled by a simple, but strong condition (*Reliability Condition*) which essentially limits the amount of every intermittent rise during an extended period of degradation. The remainder of this section describes the approach in detail.

As a first step, we compute Δ_i, as described in the previous section, at each time step beginning at the end of the first year (since we don't have sufficient information to compute Δ_i during the first year) and terminating before the start of the last year (again due to insufficient information during the last year). Let the series composed of Δ_i's, $S \leq i \leq n-S$, be Δ-series (Delta-series) where S are the number of time steps in a year, and n are the number of time steps in the entire time series. Next, we identify those time steps in the time series that have the characteristic to become the extremes of the drop window. For this effect, we compute a Γ-series (Gamma-series) from Δ-series, with each time step represented by γ_i, using the following transformation:

$$\gamma_i = \begin{cases} 1 & \Delta_i > 0 \\ -1 & \Delta_i \leq 0 \end{cases}$$

We say that those ith time steps are candidates for drop start for which γ_{i-1} is -1 and γ_i is 1 (transition to 1). Similarly, those ith time steps are candidates for drop end for which γ_{i-1} is 1 and γ_i is -1 (transition from 1). The bottom plot in Figure 4 show an example of the Δ-series, Γ-series scaled by a factor of 0.07, and candidate start and end time steps (b_1, e_1, etc.).

Let there be K candidate start and stop time steps identified in time series T, which are denoted by b_k and e_k respectively, $\forall k \in 1 \dots K$. Between every b_k and e_k there is a decrease in EVI values (decreasing trend), and between every e_k and b_{k+1} there is an increase in EVI values (increasing trend). For each b_k we are interested in identifying the farthest e_l ($1 \leq k \leq l \leq K$) such that the time series pattern within these limits in general has a decreasing trend, even if there are mild rises in between. If a drop starts at b_k, then an intermediate rise occurs between every e_l and b_{l+1} ($k \leq l < K$). In order to ensure that the decreasing trend is followed across these intermediate rises, we test for a drop reliability condition at every e_l which must be satisfied before allowing the rise between e_l and b_{l+1}. This reliability condition is given below.

Reliability Condition (RC): It states that after the commencement of a drop at a time step b_k, the rise occurring at a certain time step e_l ($e_k \leq e_l < e_K$) would not be considered as drop termination if this rise is bounded by a fraction ($x\%$) of the drop that has already occurred, and, if there is an *overall decrease* in the EVI values during the time steps in the *limited future* of e_l. The *limited future* is defined as the time steps following e_l during which the rise does not exceed $x\%$. The motivation for this is that if the current drop window has accumulated a large sum of Δ_i, a greater room for rise is allowed as long as the time series in general still has a decreasing trend.

As long as this condition is satisfied, e_l can be extended. As soon as this condition fails at a certain candidate stop time step, we stop at that e_l, and the maximum reliable drop that began at time step b_k is terminated at e_l. This becomes a candidate drop window (cw_p) for time series T. Since there could be more such windows in the same time series that begin at other candidate start time steps, b_j, $j \neq k$, we repeat the above process for the remaining candidate start time steps. Thus, we would have at most K candidate drop windows cw_p, $\forall p \in 1 \cdots K$.

To identify the best drop window, we compute a score for each candidate window using one of the methods described in Section 6. The maximum scoring window is determined as the representative window of the time series. Currently, we identify a single representative window of a time series because we are interested in identifying time series that have undergone change at least once. Thus,

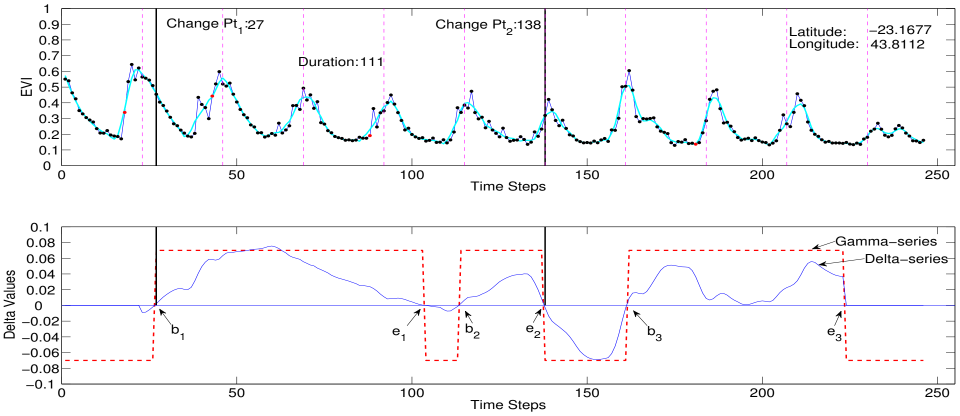


FIGURE 4. The top plot shows the original time series (line connecting the dots), smoothened time series, and the identified changed period (between solid vertical lines). The bottom plot shows the corresponding Δ-series (continuous curved line) as well as the Γ-series (broken line) scaled by a factor of 0.07.

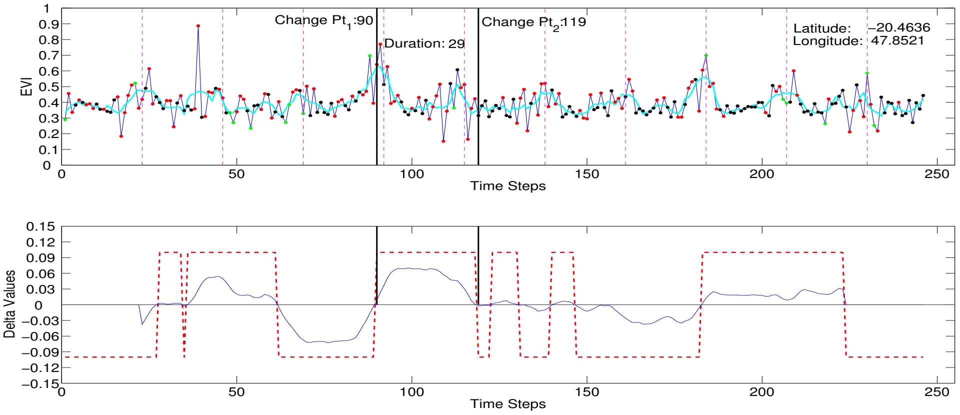


FIGURE 5. A time series in Madagascar showing a spurious rise in EVI.

.

the significance of change in each time series is given by the score computed for their corresponding representative change window. The higher the value, the more severe the change.

The example time series in Figure 4 captures the effectiveness of this approach. The top plot shows the EVI time series, the smoothed time series, and the two vertical solid lines identifying the drop period. It can be noticed that the time series gradually decreased over many years and then stabilized. The bottom plot shows the corresponding Δ-series in solid curved line and the Γ-series scaled by a factor of 0.07 by a broken line. Notice that the first drop in Δ-series below zero is included in the maximum reliable window since the reliability condition is not violated.

Let us also consider a case of a spurious rise as shown in Figure 5. In such cases the drop window resulting from this rise will be small since the subsequent period will not be able to satisfy the reliability condition. Furthermore, the score of the identified drop window according to our methodology (as would be described shortly) would be low. Hence these type of drops would be easily differentiated from the genuine drops.

6. Scoring Mechanisms and Discussion on Reliability Condition

Scoring Mechanisms. Once the candidate change windows are determined in a time series, the next step is to quantify the change in each window. Some of the methods used are: (i) **Drop in**

EVI, which is the difference between the mean annual EVI just before the start of the drop and the mean annual EVI just after the termination of the drop; (ii) **Length of the Drop Window**: The length can be a powerful indicator of the confidence of the change. A decrease of longer duration, even with a small *Drop in EVI*, can be of high confidence. Beetle infestation in Colorado (Figure 7) is a good example of this scenario; (iii) **Total Loss in EVI**: If we assume that the drop didn't take place in an actually changed time series, we can suppose that the EVI pattern, P_1, representing the year before the start of the drop would have continued. In such a scenario, the total loss in EVI would be the area enclosed between two time series, one that should have been had no vegetation loss occurred, and the other which is the actual time series in which there is a loss in vegetation.

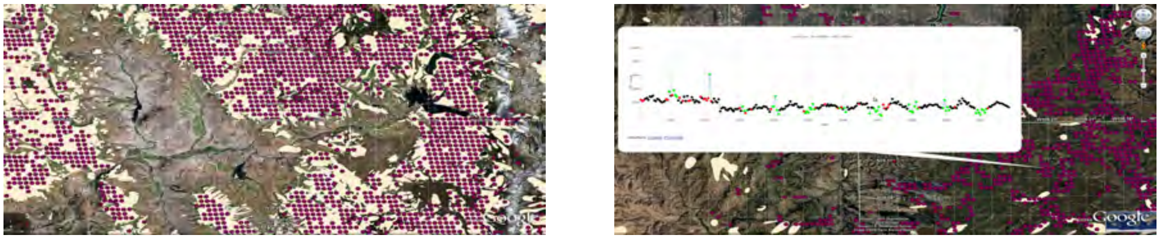
Here, we also introduce the concept of a *third* change point (the first being the drop start time step and the second being the drop end time step). After the drop occurs, if one wants to determine how long it takes for the time series to have a *significant* recovery, the third change point can be used. The third change point is positioned at a time step after the second change point such that the EVI values have risen a significant percent (say, 50%) of what it has dropped during the drop window. The position of the third change point can also be an indicator of the confidence of the drop. If the third change point is realized after many time steps following the drop window, the drop is trustworthy because the vegetation stays low for a long time. On the other hand, if the third change point occurs soon after the second change point, it might mean that either the vegetation indeed recovered very quickly, or the drop was actually spurious and short-lived.

The different scoring schemes could also be incorporated into a single *cost function.* The new cost function could be constructed such that it gives a minimum cost to a drop window that receives a high score from all the above mentioned schemes, as well as high cost to a drop window getting a low score from each scheme. A single cost threshold could be set which differentiates a genuine change from a spurious one. This cost function must be designed with care such that it's applicable globally, and we leave this for investigation in future work.

Accounting for Variability in a Time Series. Though we have briefly mentioned variability in a time series before, here we discuss it in the context of quantifying the EVI loss. Natural variability occurs in EVI values from one year to the other due to changes in environmental conditions such as temperature, precipitation, cloud cover, etc., or imprecision in measurement. Such a change in a time series should not be regarded as a loss in vegetation. Furthermore, a true loss in vegetation would be over and above the natural variability because a loss in vegetation equal to the natural variability would actually be a common signature of that vegetation. A way to model the natural variability is to take a mean of pairwise distance between EVI values of annual segments either during the first few years (before the first change point) or immediate previous few years before the first change point. The *city-block* distance measure (L_1 norm) works well for computing this variability. This variability is subtracted from the *Drop in EVI* in order to reflect a true drop in vegetation on the ground. Similarly, the EVI values of the pattern P_1 should be lowered by this variability before computing the *Total Loss in EVI.* For evaluation presented in this paper, we have used *Total Loss in EVI* as the scoring scheme with compensation for variability.

Potential Reliability Condition Augmentation. In the previous section, we described a method to determine the start and end time steps of a candidate drop window. As it would appear, these change points are dependent on the reliability condition used. We have described a simple reliability condition that limits the amount of rise in the time series once the drop has begun. If the rise is greater than a threshold, the drop is terminated by positioning a second change point before this rise and a new drop window is initiated at the next candidate start time step. Finally, the best drop window is selected as the representative drop window of the time series.

The advantage of using the above reliability condition is that it is simple, as well as it has only one parameter. But this condition may handle some time series changes differently from what one might prefer. For instance, if there is a time series that drops during the third year, stays almost constant



(a) A snapshot showing locations identified as changed by the proposed approach (red circles) overlaid on the validation data polygons.

(b) Snapshot showing a large region in Colorado where there was a drop in EVI due to Fire in the year 2002. Most of this region is not covered by the polygons.

FIGURE 6. Snapshots showing regions in Colorado where vegetation loss was detected.

for the next five years, and then drops again during the eighth and ninth years, then the start and end change points would be identified around the second year and the ninth year respectively. It would overlook that the time series is not decreasing at all for many intermittent years. It might be more desirable to include these two drops in separate drop windows instead of one. But such cases are not a limitation, as the proposed reliability condition can be adapted to handle these cases. It can be taken as a base condition over which other conditions are added, which may or may not be region specific. In the context of the above example problem, one possible adaptation could be to add a condition that the drop window must have recurrent drops, say every y years or so. This wouldn't allow a period of stagnation for more than y years. Note that this modification would result in the use of another parameter, y, thus deviating further from simplicity.

7. EVALUATION

Evaluation of a scheme for detecting changes in forest cover is challenging due to the lack of high-quality ground truth. The most reliable methods for generating ground truth (e.g. ground surveys) are very expensive and are thus only available for small regions.

In the absence of such gold standard ground truth, less reliable labels generated by some other scheme or via aerial surveys can still be used for validation, but care must be taken to check if "false positives" (i.e. changes found by the scheme but not in the validation data) are indeed false, since they could have been missed by the scheme used to generate labels. Similarly, one needs to check if "false negatives" (i.e. changes noted in the validation data but not found by the scheme) are indeed changes on the ground as they could be incorrectly classified as changed by the other scheme.

We evaluated our approach on two regions; Northern Colorado and Southern Madagascar for which moderate quality labels are available in the form of polygons covering degraded areas of the forests. These regions are interesting because they have completely different vegetation types, and the degradation is caused by different mechanisms; specifically, insect damage in Colorado, and logging in Madagascar. During analysis in both regions we emphasize the strength of this algorithm in detecting changes that are difficult to identify, as well as the capability of this approach to capture many changes that are missed by the validation data.

7.1. Evaluation on Colorado Forests.

The first region of analysis is forests in northern Colorado (the region bounded by 39°N—41°N; 108°W—104.5°W). The US Forest Service and its partners [2] maintain data sets which map the regions of forest cover that have degraded between years 2002 and 2008 in northern Colorado. The objective was to detect regions of forest degradation using our approach and evaluate it against the above validation data (which has been transformed to polygons). We present our analysis as follows:

True positives are points detected by our approach that also lie in the polygons. As seen in Figure 6a, there is a very good overlap of the detected points with the validation polygons. The algorithm is also able to correctly identify the period of degradation. Colorado is a difficult region

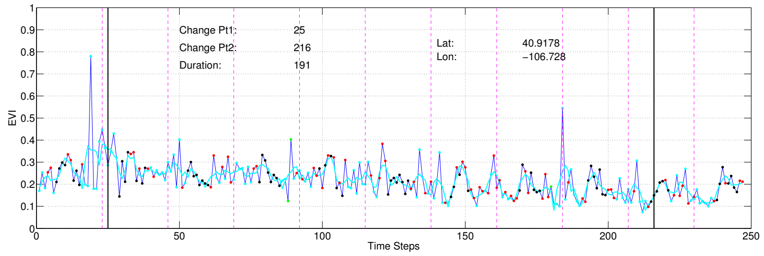
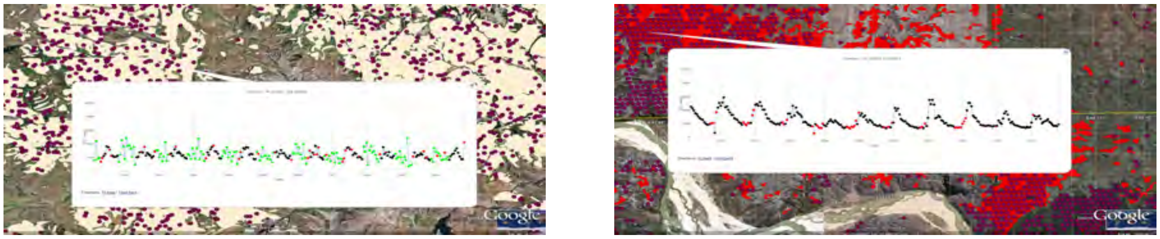


Figure 7. A typical gradual drop in Colorado due to beetle infestation.



(a) This figure shows an example time series in Colorado that has no perceptible change but which fall inside the ground truth polygon.

(b) Time series of a region in Madagascar showing gradual decrease in EVI starting from year 2001, which lie inside the ground truth polygon.

Figure 8. Snapshots of false negatives in Colorado and true positives in Madagascar.

because changes here are often very gradual, sometimes to the extent that there is no visible change in EVI signal upon manual inspection. Nevertheless, our approach identified a significant number of points. Figure 7 shows the typical EVI time series in this region.

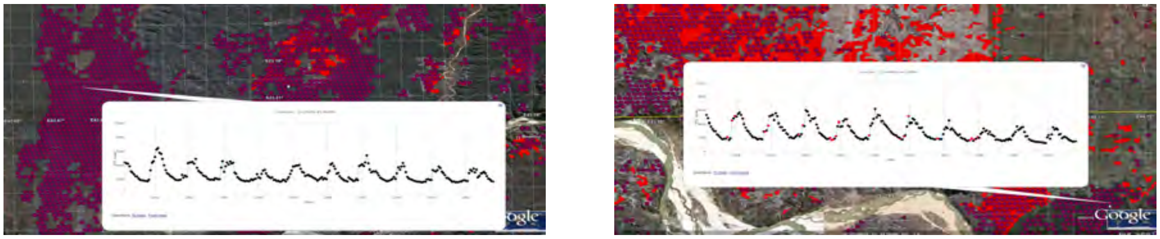
False positives are points that we detected as change but do not lie in any of the polygons. There could be several reasons for this: (i) the decrease in vegetation in these areas is caused by factors other than those considered in constructing the validation data; (ii) it is known that the polygons can be inaccurate (iii) these points are in fact not changed, but due to noise in EVI appear as changed and thus given a high score by the proposed approach. Our manual inspection shows that majority of false positives with high scores are due to (i) and (ii). For example, consider the region shown in Figure 6b. This region is not part of any polygon even though the change is quite apparent and is likely due to fire [7].

False Negatives are points that we did not detect as changed but which lie inside the polygons. Figure 8a, shows an example of such a time series. This time series does not show any change in the EVI signal. There are numerous time series in this region that show little perceptible change and are in the polygons. So either the vegetation loss here is too gradual to be detected by our approach, the change on the ground is not captured by the EVI signal, or the polygons are inaccurate.

7.2. **Evaluation on Madagascar Forests.** The second region of analysis is southern Madagascar (the region bounded by 25.6°S—20°S). The validation data is obtained from Center for Applied Biodiversity Science (CABS) at Conservation International (CI), whose analysis is based on bitemporal Landsat image comparison between years 2001 and 2005 [8]. Hence, the validation data (or polygons) cover changes only between these two years.

True Positives and False Negatives: Figure 8b shows an example of a true positive. The image clearly shows the gradual drop starting in the year 2001. Most points in the validation polygons show similar behavior, although with varying decay rate and duration. Hence, effectively there are few false negatives since most points in the validation polygons can be found by our algorithm.

False Positives: Most of the false positives were observed due to the following reasons:



(a) A snapshot showing a large region in Madagascar where vegetation loss started to occur in the year 2001.

(b) This figure shows an area where vegetation loss occurred after year 2005.

Figure 9. False Positives in Madagascar having a decreasing EVI signal.

(1) Vegetation degradation occurred during the period of analysis (2000-2005) but the vegetation recovered during 2005, which causes it to be missed by the technique used in generating the validation data (Figure 9a). The entire cluster of points to the left in the figure has similar time series signature but lies outside the validation polygons. This illustrates the limitation of the technique that are based on the comparison of images taken on two different dates. (2) Significant vegetation degradation is visible only after 2005 hence was not included in the validation data set. Figure 9b shows an example of such a region. This type of identification also highlights the capability of our approach to find changes with high temporal precision in a continuous manner. This is opposed to the image-based methods where analysis is usually done on snapshots of images generally few years apart.

8. Comparative Evaluation on Synthetic Dataset

In this paper, we compare our proposed technique with two other approaches for gradual change detection, CUSUM and BFAST [17] using data sets with simulated noise and change characteristics. The BFAST technique is designed to detect long-term changes in satellite image time series. It decomposes a time series into trend, seasonal, and residual components, such that the intra-segment models are constant, while inter-segment models are dissimilar. BFAST identifies the optimal position of trend and seasonal breakpoints by minimizing the residual sum of squares (RSS), and the optimal number of breaks can be determined by minimizing an information criterion. Before estimating the breakpoints, the ordinary least squares residuals-based moving sum test is used to identify if any breakpoints are occurring in the time series. As output, BFAST provides the trend breakpoints and associated trends, seasonal breakpoints and associated seasonal models, and logical values indicating whether the time series is considered changed in the seasonal or trend components. We do not consider the seasonal component in this paper since we are looking for a decreasing trend.

Evaluating our algorithm against BFAST is not straightforward since BFAST looks not only for drops, but any type of trend change in a time series. Also, simply consulting the logical vector values that labels a time series as changed was not feasible for two reasons: (i) BFAST appears to be sensitive to noise and frequently finds different trends even in a stable time series and labels them as changed. (ii) BFAST would label a time series as changed if any type of trend change is present, notwithstanding the absence of a decreasing trend. In addition, BFAST also requires some parameter settings such as the minimum segment size and maximum number of breakpoints desired. These parameters are not mandatory, but not setting them makes it quite sensitive to noise, resulting in breaking even a single trend into multiple segments. Therefore, construction of the synthetic datasets had to be in consonance with the parameter values of BFAST.

We constructed two types of datasets, DS1 and DS2, the first containing three different trends (two trend breakpoints), and the other containing four different trends (three trend breakpoints). The maximum number of breakpoints set in BFAST for these two types of datasets were two and three respectively, and it was expected that BFAST would correctly identify all the given trends. Since we are interested in identifying the decreasing trend, our trend of interest among the ones returned by BFAST is the one which has the largest decrease across it. Its change score (which also represents the score of the time series) is computed in the same manner in which we compute the score for our proposed approach. Below, we describe the creation process of the synthetic dataset.

8.1. **Synthetic Data Generation.** The datasets, DS1 and DS2 are comprised of 1100 time series each, in which a gradual decrease phase was inserted in 80 time series for DS1, and 120 time series for DS2. In DS1, 40 time series also have an increasing trend. The remaining stable time series in both the datasets are identical (total: 980). Each time series has 322 time steps, with a seasonal period of 23 time steps (in order to mimic the MODIS EVI time series having 14 years of data). The seasonality in a time series is created using a function of the form:

$$F(x) = A * e^{\frac{-|x-m|}{B}}$$

where A controls the amplitude, x varies between time steps of a particular year, m controls the position of the peak in that year, and B controls the curve. The shape of $F(x)$ mimics a typical seasonal vegetation pattern of a forested region (or farming cycle) as reflected in an EVI time series.

Each time series has different types of noise added to it. We define these below, followed by the characteristics of the changed and stable time series.

Noise characteristics Two types of noise are introduced in the dataset. w_1 is white noise that is added to each time step in the time series. w_2 is outliers, that results in very high (upward spikes) or very low (downward spikes) values at certain time steps as compared to that of its neighbors.

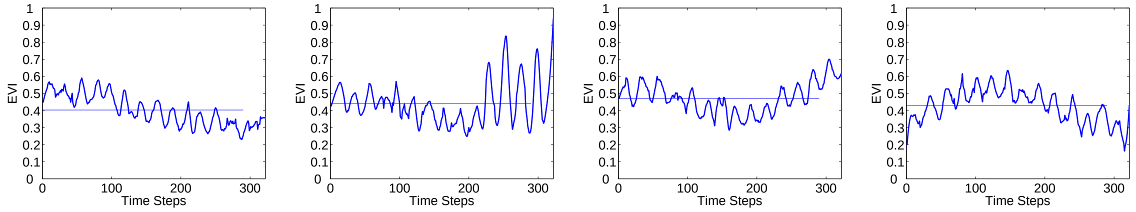
Characteristics of a changed time series There are three phases in these time series. (1) *beforePhase* is the period in the time series before a drop. Here, seasonal cycles (pattern during one year) are represented by $F(x)$. This phase may have an increasing trend or a stable trend. Noise w_1 and w_2 is added to each time step. All introductions in this phase are probabilistic as a Gaussian distribution within sufficient ranges specified in advance. This includes the values of w_1, w_2, duration of this phase, height of the data values during each year, duration of the increasing trend if any. (2) *changePhase* starts as soon as *beforePhase* ends. The majority of these time series have a decreasing trend. The base level of successive years in this phase is reduced gradually from starting of this phase till its end. The duration of this phase and the amount of drop introduced are probabilistic within a certain range. Noise w_1 and w_2 are added to this phase as well. In a small fraction of time series, an increasing trend is added during this phase instead of a decreasing trend to include more variety of time series. However, since these time series do not contain a decreasing trend, they are considered as false positives if detected by any algorithm. (3) *afterPhase* starts after the *changePhase* ends. Each year in this phase is also represented by $F(x)$ with w_1 and w_2 added.

Characteristics of a stable (unchanged) time series These time series have one phase with a constant level whose value is probabilistic within a certain range. Each seasonal cycle in these time series is also represented by $F(x)$ with noise w_1 and w_2 added.

8.2. **Evaluation Strategy.** PDELTA, CUSUM, and BFAST are applied to these datasets after preprocessing as described in Section 2. We compare the performance of our approach with CUSUM and BFAST separately. It is because these two approaches return different information about the drop, and we adjust our evaluation according to this information. For CUSUM, we combine the samples of time series from datasets DS1 and DS2 into a single dataset DS0.

8.3. **Comparison with CUSUM.** We evaluated the performance of PDELTA and CUSUM on the dataset DS0. There are four different types of time series in DS0 that have a decreasing trend (Figure 10). Each pattern has 50 different samples. Overall, this dataset has 200 decreasing time series, and 1020 stable time series (Total: 1220). The time series patterns included in DS0 are common in the real world datasets. Pattern one (Figure 10a) is an example of a stable forest that degrades over many years. Pattern two (Figure 10b) is an example of conversion of forests to farm lands, as could be noticed from the typical farming cycles during the later part of the time series. Figure 10c reflects some plantation following a deforestation. Figure 10d could depict a failed reforestation.

The precision-recall curve of the result of the two algorithms is shown in Figure 11a. CUSUM performed best on the samples of the pattern shown in Figure 10a. Admittedly, CUSUM performs



(a) Pattern one (b) Pattern two (c) Pattern three (d) Pattern four

FIGURE 10. Different decreasing patterns in D0 dataset. Each pattern has fifty samples (total two-hundred). The horizontal line shows the mean of the time series.

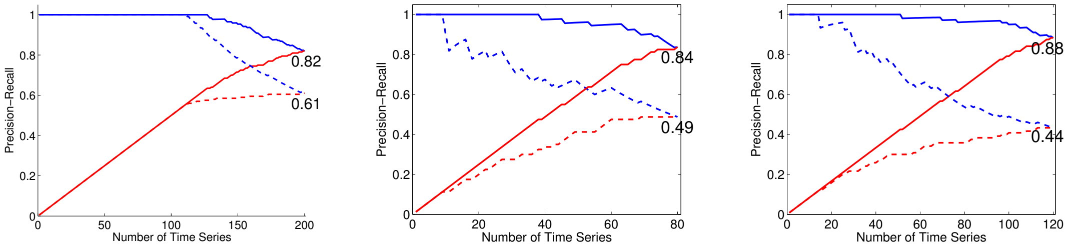
well for any gradually decreasing time series that accumulates a large score during the beginning time steps, which would happen if most of the beginning values are placed above the expected value μ. However, CUSUM would perform equally poorly on time series having an opposite signature. This highlights a major drawback of CUSUM. Consider the patterns two and three shown in Figures 10b and 10c, on which CUSUM performed poorly. In these time series, most of the values in the beginning are below μ, which is taken as the mean of the time series, implying that the majority of the later values are above μ. Such time series would never be able to accumulate a high cumulative sum since it incurs a large loss in the beginning due to negative deviations, and therefore would be given a low score. Many such scenarios could be constructed where there is a decreasing trend present but the time series never accumulates a high enough score for it to be significant.

We also investigated alternative ways of computing the expected value, but these variations either repeated some of the above drawbacks, or other drawbacks were discovered in them. For instance, by taking μ as the mean value of the first year, CUSUM performed poorly on patterns two, three, and four (Figure 10). Note that if we consider this variation, we are looking for the minimum cumulative sum (instead of the maximum) since an ideally decreasing time series would have a highly negative cumulative sum. The main disadvantage of this variation is that the cumulative sum is highly dependent on the first year values. If the first year values are noisy, it can drastically affect the algorithmic output. As an example, if μ is even slightly high due to noise, a stable time series could get a high score. In contrast, if μ is low, decreasing time series can go undetected (Figure 10d).

The precision-recall curve suggests that PDELTA performed considerably better than CUSUM on this dataset. Additionally, PDELTA also identifies the period of decrease.

8.4. **Comparison with BFAST.** Our evaluation with BFAST is based on two factors: (1) Precision-recall curves for PDELTA and BFAST, formed by ranking the time series in decreasing order of scores, when evaluated on DS1 and DS2 (Figures 11b and 11c). (2) Scatter plots of the deviation of the change points identified by the two approaches from the actual positions where the breakpoints were introduced in the synthetic datasets (Figure 12).

The scatter plots show that the distribution of the deviation of the change points identified by PDELTA is closer to zero than that of BFAST. BFAST segments a time series based on RSS and a Bayesian Information Criteria (BIC), which has no bias towards identifying decreasing periods. If identifying the decreasing trend as a separate segment minimizes RSS, BFAST will correctly identify the decreasing trend. Otherwise, it may combine a part of the decreasing trend with an adjacent trend. Also, BFAST appears to be sensitive to noise in a time series. It correctly detected the trends introduced for many time series (Figure 13a), but it segmented stable time series into different trends as well (Figure 13b). This suggests that BFAST might not be very suitable for highly variable time series, where noise levels can distort the ideal seasonal pattern enough. Such highly variable time series are characteristic of the tropical belt such as in Para (Brazil), Peru, Congo, etc.

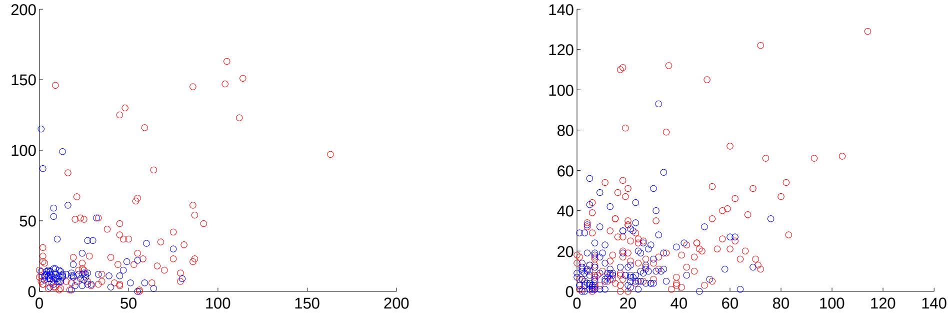


(a) DS0 - Precision-Recall for PDELTA (solid - precision: 0.82) and CUSUM (broken - precision: 0.61). Noise $w_1 = 200$, $w_2 = 10\%$.

(b) DS1 - PDELTA (0.84). BFAST (0.49)

(c) DS2 - PDELTA (0.88). BFAST (0.44)

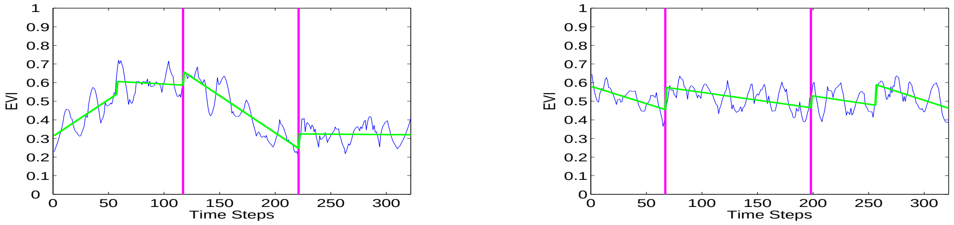
Figure 11. Precision curves (blue) and recall curves (red) for PDELTA (solid curves), CUSUM (dashed curves), and BFAST (dashed curves).



(a) DS1 - PDELTA (Blue), BFAST (Red).

(b) DS2 - PDELTA (Blue), BFAST (Red).

Figure 12. Scatter plots of the deviation of change points detected by PDELTA and BFAST from actual drop start (x-axis) and drop end (y-axis) time steps.



(a) Correctly detecting a decreasing trend.

(b) Many trends identified in stable time series.

Figure 13. Trends using BFAST. Vertical lines identify the period of maximum drop.

9. Conclusion

In this paper, we presented a globally scalable novel approach, PDELTA, for detecting a gradually decreasing EVI time series that can capture changes caused by a variety of sources. PDELTA can be considered an adaptation of CUSUM with the added capability of identifying the period of decrease and quantifying the magnitude of drop in a time series, while being more robust in the presence of noise and spurious changes. We demonstrated the efficacy of the proposed approach using independent validation data sets in Colorado and Madagascar. It was also shown that genuine changes were detected by our technique which were missed by other approaches, as well as points identified as changed by other approaches with no perceptible EVI signal were not detected. We comparatively evaluated our technique with CUSUM, and the state of the art BFAST technique. BFAST in its present form is computationally very expensive, whereas both PDELTA and CUSUM are quite fast. PDELTA can also identify reforested areas depicted by increase in vegetation simply by reversing a time series before applying this algorithm. Future extensions of this work include adapting PDELTA to detect more general types of changes (e.g. abrupt changes). Also, while this paper focuses on identifying the single most significant drop in a time series, PDELTA is able to identify multiple decreasing segments. Thus, other decreasing segments could also be identified as separate changes if the drop within them is also significant (multiple change detection). This aspect of PDELTA needs to be further explored and developed.

10. Acknowledgments

The research described here is supported by the Planetary Skin Institute, NSF Grant IIS-0713227, NSF Grant IIS-0905581, NASA grant NNX09AL60G and the University of Minnesota MN Futures Program. Access to computing facilities was provided by the University of Minnesota Supercomputing Institute.

References

[1] Land Processes Distributed Active Archive Center. `http://edcdaac.usgs.gov`.

[2] U.S. Department of Agriculture, Forest Service, Rocky Mountain Region, Forest Health Management. `http://www.fs.fed.us/r2/fhm`.

[3] S. Boriah. *Time Series Change Detection: Algorithms for Land Cover Change*. PhD thesis, University of Minnesota, 2010.

[4] P. Coppin, I. Jonckheere, K. Nackaerts, B. Muys, and E. Lambin. Digital change detection methods in ecosystem monitoring: a review. *International Journal of Remote Sensing*, 25(9):1565–1596, 2004.

[5] L. Eklundh, T. Johansson, and S. Solberg. Mapping insect defoliation in Scots pine with MODIS time-series data. *Remote sensing of Environment*, 113(7):1566–1573, 2009.

[6] G. M. Foody. Status of land cover classification accuracy assessment. *Remote Sensing of Environment*, 80(1):185–201, 2002.

[7] R. T. Graham. Hayman fire case study. Technical Report RMRS-GTR-114, U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station, 2003.

[8] G. J. Harper, M. K. Steininger, C. J. Tucker, D. Juhn, and F. Hawkins. Fifty years of deforestation and forest fragmentation in madagascar. *Environmental Conservation*, 34(4):325–333, 2007.

[9] J. Kucera, P. Barbosa, and P. Strobl. Cumulative sum charts-a novel technique for processing daily time series of modis data for burnt area mapping in Portugal. In *MultiTemp 2007: International Workshop on the Analysis of Multi-temporal Remote Sensing Images*, pages 1–6. IEEE, 2007.

[10] W. A. Kurz, C. C. Dymond, G. Stinson, G. J. Rampley, E. T. Neilson, A. L. Carroll, T. Ebata, and L. Safranyik. Mountain pine beetle and forest carbon feedback to climate change. *Nature*, 452(7190): 987–990, 2008.

[11] D. Lu, P. Mausel, E. Brondízio, and E. Moran. Change detection techniques. *International Journal of Remote Sensing*, 25(12):2365–2401, 2003.

[12] V. Mithal, A. Garg, S. Boriah, M. Steinbach, V. Kumar, C. Potter, S. Klooster, and J. Castilla-Rubio. Monitoring global forest cover using data mining. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(4):36, 2011.

[13] S. J. Orfanidis. *Introduction to signal processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, 1995.

[14] E. S. Page. Continuous inspection schemes. *Biometrika*, 41(1/2):100–115, 1954.

[15] G. R. van der Werf, D. C. Morton, R. S. DeFries, J. G. J. Olivier, P. S. Kasibhatla, R. B. Jackson, G. J. Collatz, and J. T. Randerson. Co2 emissions from forest loss. *Nature Geoscience*, 2(11):737–738, 2009.

[16] P. J. van Mantgem, N. L. Stephenson, J. C. Byrne, L. D. Daniels, J. F. Franklin, P. Z. Fulé, M. E. Harmon, A. J. Larson, J. M. Smith, A. H. Taylor, and T. T. Veblen. Widespread Increase of Tree Mortality Rates in the Western United States. *Science*, 323(5913):521–524, 2009.

[17] J. Verbesselt, R. Hyndman, G. Newnham, and D. Culvenor. Detecting trend and seasonal changes in satellite image time series. *Remote sensing of Environment*, 114(1):106–115, 2010.

A MACHINE LEARNING APPROACH FOR PROBABILISTIC DROUGHT CLASSIFICATION

Ganeshchandra Mallya[1], Shivam Tripathi[2], Rao S Govindaraju[1]

ABSTRACT. Current methods of drought assessment utilize drought indices, such as the standardized precipitation index and Palmer drought severity index, that rely on subjective thresholds and hence cannot be universally applied across different climatic regions. In addition, most of the existing drought indices are not amenable to probabilistic treatment which is essential for quantifying model uncertainties in drought classification. This study applies a machine learning tool, the hidden Markov model (HMM), for probabilistic drought classification. The HMM-based drought index (HMM-DI) developed in this study, does not require specification of subjective thresholds and model parameters are determined from historical data during parameter estimation. The drought classifications obtained using HMM-DI are compared with SPI results. The HMM-DI reveals new insights into the frequency and severity of droughts and their spatio-temporal variations. The effectiveness of HMM-DI is assessed by its application to monthly precipitation data over India. The results suggest that HMM-DI can be a promising alternative to conventional drought indices.

1. INTRODUCTION

Droughts are assessed using *drought indices* that provide a numerical standard for comparing drought characteristics over time and over different regions. According to World Meteorological Organization (WMO), a drought index is “an index which is related to some of the cumulative effects of prolonged and abnormal moisture deficiency” [1]. Numerous drought indices have been proposed in the literature [2]; some of the early indices include: Munger’s index [3], Blumenstock’s index [4], Antecedent Precipitation Index [5], Palmer Drought Severity Index (PDSI) [6], Crop Moisture Index (CMI) [7] and Surface Water Supply Index [8]. Although these indices use different forms of water deficits to characterize droughts, the results often do not correspond well among the indices owing to the complex physics that involves precipitation, infiltration, evapotranspiration, groundwater, base flow and direct runoff. Another popular index - the standardized precipitation index (SPI) [9] has gained wide recognition because of its computational simplicity and versatility in comparing different hydro-meteorological variables at different time scales. In SPI, historical observations are used to compute the probability distribution of the monthly and seasonal (2 months, 3-months, etc., up to 48 months) precipitation totals. The fitted probability distributions are then normalized using the standard inverse Gaussian function to calculate the SPI. A negative value of SPI indicates precipitation less than median rainfall, and the magnitude of departure from zero represents the severity of a drought. McKee et al. [9] suggested a classification scale in which (i) *extreme drought* occurs when SPI value is less than -2.0, (ii) *severe drought* when SPI value is between -1.5 and -2.0, and (iii) *moderate drought* when SPI value lies between -1.5 and -1.0.

[1] School of Civil Engineering, Purdue University, West Lafayette, IN 47907, USA. govind@purdue.edu

[2] Department of Civil Engineering, Indian Institute of Technology, Kanpur – 208016, India. shiva@iitk.ac.in

The SPI has several limitations when applied to precipitation data over India as described in Mallya et al. [10] and are briefly given below:

(a) The SPI cannot identify drought prone areas - in SPI, the precipitation data are transformed to a standard normal distribution, and therefore the frequency of drought remains same irrespective of the region. Further, the frequency of drought remains same irrespective of the duration of the drought.

(b) For regions where precipitation exhibits small variability, even a small anomaly in the precipitation can lead to large negative SPI values.

(c) In estimating SPI, it is generally assumed that the precipitation values are independent. This independence assumption may not hold true when estimating longer duration droughts (window size greater than twelve months).

This study uses hidden Markov model to develop a drought index (HMM-DI) for probabilistic classification of drought states. Unlike SPI, the HMM-DI does not require subjective thresholds because they are determined from historical data during parameter estimation. The HMM-DI is developed to overcome some of the aforementioned limitations of SPI.

This paper does not provide any new algorithm development or methodological innovation. Rather, the goal of the paper is to use an HMM to achieve two important aspects that are not reflected in current drought indices, namely (i) representation of temporal dependence in drought states, especially for longer duration droughts, and (ii) a probabilistic classification of drought status. The focus of this study is on application of this method to precipitation data, and to analyze the results for seeking new insights into drought patterns over India. The remainder of the paper is structured as follows: the mathematical formulation of the new drought index is first outlined. The data used in the study are then described. Subsequently, the results obtained are discussed, and a set of conclusions are presented.

2. MATHEMATICAL FORMULATION

2.1 Hidden Markov Model

An HMM is a statistical model in which observations from a system are assumed to be conditioned on the state of the system [11]. The state is hidden and satisfies the Markov property. The HMM was developed in late 1960s and early 1970s for speech recognition, and it has since been used successfully in many applications including hydrology [12]. The mathematical formulation of the HMM used in this work is described in detail by Tripathi and Govindaraju [13], and is briefly presented in the following paragraphs.

Let the rainfall at time t be denoted by x_t, $t = 1, \ldots, N$ $\{x_t \in \Re$ and $\boldsymbol{x} = [x_1, \ldots, x_N]^{\mathrm{T}}\}$. In a HMM, the rainfall x_t is assumed to depend only on the state variable z_t $\{\boldsymbol{Z} = [z_1, \ldots, z_N]^{\mathrm{T}}\}$ that denotes a drought or wet state, is hidden (not observed), and follows the first order Markov property. The state variable z_t is a K-dimensional binary random variable. If the number of states, K, are known *a priori*, the standard HMM can be parameterized using the following three distributions:

1. The conditional distribution of rainfall given the drought state, $p(x_t \mid z_t)$, referred to as the *emission distribution*.
2. The conditional distribution of the present drought state given the previous state i.e. $p(z_t \mid z_{t-1})$. Because z_t is a K dimensional binary variable, the conditional distribution is given by a $K \times K$ transition matrix $\boldsymbol{A}$ whose element $A_{jk} = p(z_{tk} = 1 \mid z_{t-1,j} = 1)$.
3. The marginal distribution of the drought state at the first time step, $p(z_1)$, given by a K dimensional vector $\boldsymbol{\Pi}$ whose element $\pi_k = p(z_{1k} = 1)$.

For a drought index, a definition of drought states that remains unaltered irrespective of the location or the time of a drought is desirable. To achieve this property, the following two steps were taken:

(a) The rainfall data at any desired time scale (from one month to several years) were standardized by subtracting the data from its mean and dividing it by its standard deviation. The standardization brings the data from different locations and time scales to a common platform. The HMM model was applied to the transformed data.

(b) The emission distribution was selected to be a mixture of Gaussian distributions of the form

$$p(x_t \mid z_t) = \prod_{k=1}^{K} \mathcal{N}(x_t \mid \mu_k, \sigma_k^2)^{z_{tk}} \qquad (1)$$

where μ_k and σ_k^2 are the mean and the variance of a Gaussian distribution, respectively. Since the results of the developed drought index are compared with SPI, the number of states (components in the Gaussian mixture) K was set to 7 (3 drought states + 1 normal state + 3 wet states). In Mallya et al. [10], the μ_k's and σ_k's were fixed *a priori* bringing subjectivity in HMM-DI. In this study, the μ_k's and σ_k's were considered to be free parameters and were estimated along with other parameters of HMM.

The parameters of the HMM were estimated by the method of *maximum likelihood* using Baum-Welch algorithm [14].

3. DATA USED IN THE STUDY

Daily rainfall data at a spatial resolution of 1° for both latitude and longitude were obtained from India Meteorological Department (IMD) and are based on a total of 1384 stations distributed over India that have at least 70% availability for the period 1901-2004 [15]. The gridded data consisting of 357 grid points have been obtained by interpolating raingage data. This data set is an extension of Rajeevan et al. [16] data set that was available for the period 1951-2004.

4. RESULTS AND DISCUSSION

The HMM based drought index and SPI were estimated at each grid point for 1, 3, 6, 12, 24 and 36 months windows. For brevity, the results for window ending in September are discussed here.

Figure 1 presents the emission distributions for extreme, severe and moderate drought states estimated for 3-months window ending in September at Rajasthan (grid 255) and the Western Ghats (grid 25). The geographical locations of the grid points are shown in Fig. 4. The probability density function (pdf) of the standardized cumulative precipitation at the grid-points for 3-months window is determined using non-parametric kernel density estimation method and is shown using black thick line. The pdf at both the grid points are positively skewed. The pdf at gird 255 has a steeper rising limb compared to the pdf at grid 25 indicating that low rainfall values are relatively rarer in grid 25. The emission distribution for the extreme drought state at grid 25 is broad and diffusive compared to that at grid 255, and consequently extreme droughts are more likely to occur over grid 255 than over grid 25. Further, since the emission distributions have smaller variance at grid 255 there would be less uncertainty in the determination of drought states compared to grid 25. The proposed HMM-DI utilizes this information contained in the precipitation data. This information could not be engaged in drought classification by SPI with fixed thresholds or by HMM-DI with fixed emission distributions as was done in previous studies [9, 10].

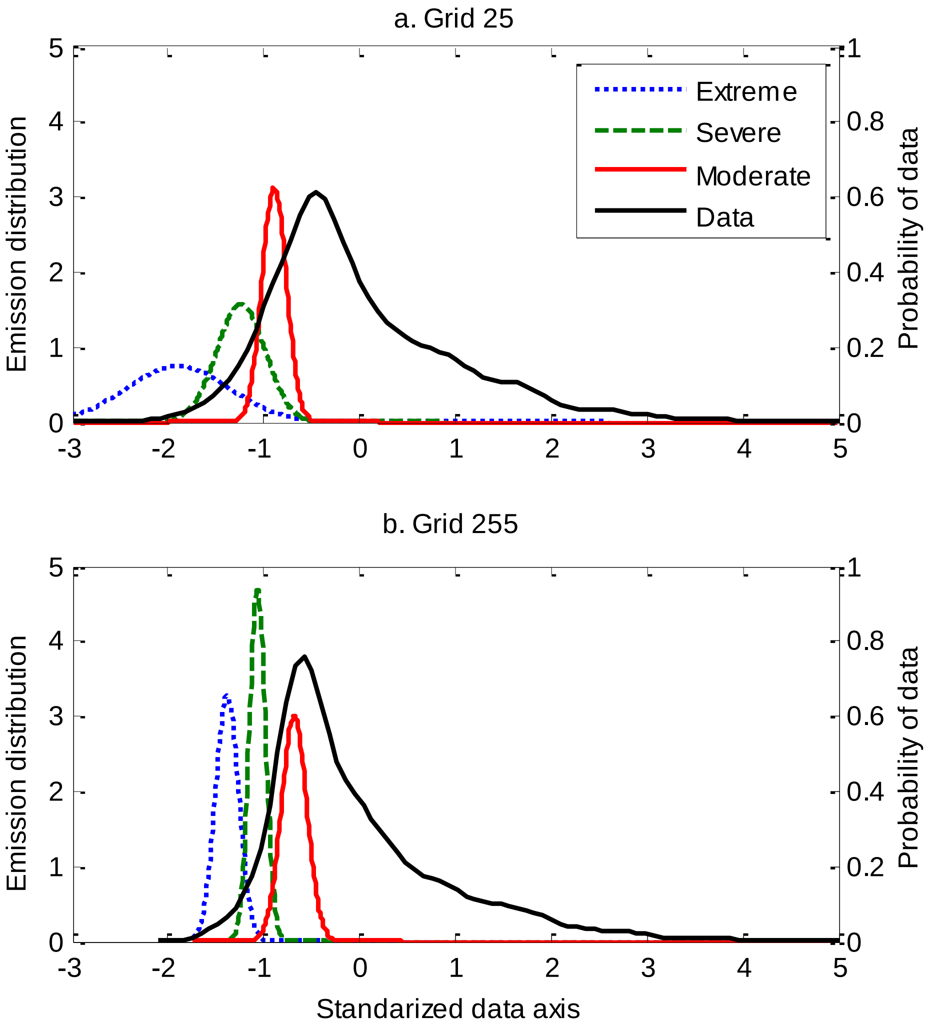


Figure 1, The emission distributions for extreme, severe and moderate drought states estimated for Grids 25 and 255. The emission distributions correspond to the 3-month rainfall window ending in September. The black thick line represents the probability density function (pdf) of the cumulative precipitation in that window. The pdf is determined using non-parametric kernel density estimation method.

Figures 2 and 3 compare the performance of HMM and SPI for grid points 255 and 25, respectively. Both the models classify droughts into three categories - moderate, severe, and extreme; the HMM provides probabilistic classification while SPI yields a discrete classification.

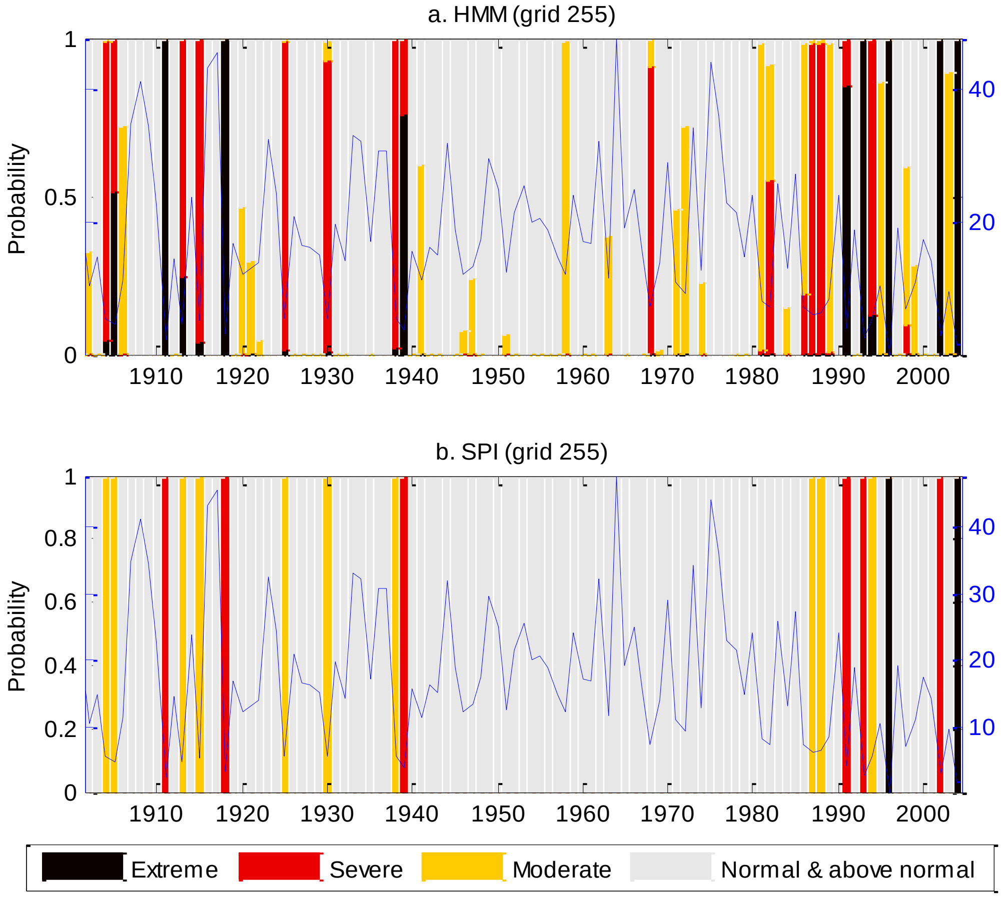


Figure 2. Drought states identified by the HMM model (top panel) and SPI (bottom panel) for grid 255 located over Rajasthan. The results correspond to 3-months window ending in September. The cumulative rainfall over the window period is shown using a solid blue line. The legends used are shown in the bottom panel.

For Rajasthan, the SPI classifies 1938 and 1939 as moderate and severe drought years, respectively, even though the magnitude of precipitation among those years differs by only a few centimeters. In contrast, the HMM classifies both the years under severe to extreme drought category with certain probabilities. Similarly, the 1986 and 1987 precipitation over grid 255 differs by just over a centimeter; however, the SPI classifies one under normal and the other under moderate drought category. The HMM classifies both the years to be in moderate to severe drought categories with certain probabilities. These examples highlight that even a small difference in precipitation may lead to two different drought categories by the SPI, a problem which the HMM-DI can avoid owing to its probabilistic formulation, thus providing a more realistic assessment of drought status.

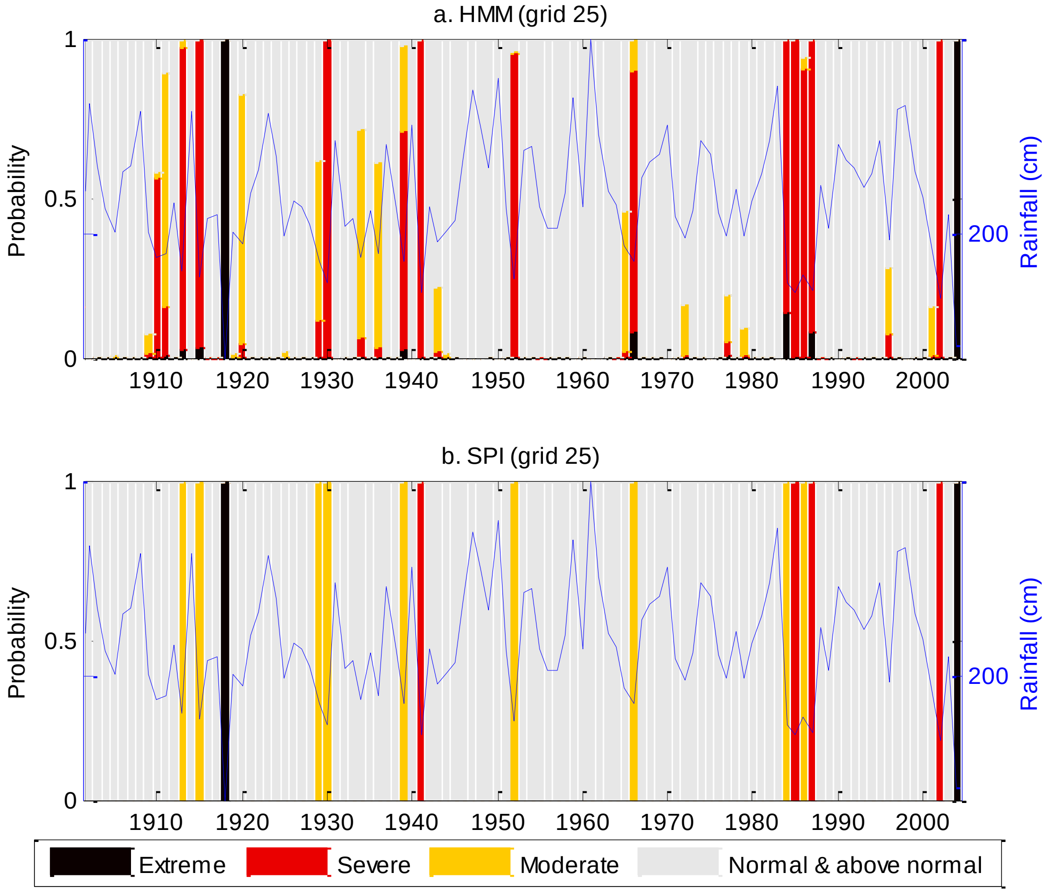


Figure 3. Drought states identified by the HMM model (top panel) and SPI (bottom panel) for grid 25 located over the Western Ghats. The results correspond to 3-months window ending in September. The cumulative rainfall over the window period is shown using a solid blue line. The legends used are shown in the bottom panel.

For the grid 25 located in the Western Ghats, there are a few similarities between the results given by the HMM and the SPI. The year 2002 is classified under severe and 2004 under extreme drought categories by both the models. The cumulative precipitation amounts over grid 25 during the selected 3-months window for 1986 and 1987 differ only by a few cm, and it is expected that both the years will be under the same category. The HMM gives this intuitive result; the SPI, however, classifies 1986 as a moderate drought year and 1987 as a severe drought year indicating that the SPI may sometimes lead to misleading conclusions.

Figures 4 and 5 show the drought states over India for two extremely dry years, 2002 and 2004. For 2002, the HMM model suggests that most of the country, except for some parts of West Bengal and Northeast India, is under drought. The SPI incorrectly indicates that large portions of Central and North India are under normal monsoon conditions. Additionally, the drought patterns given by the HMM possess spatial contiguity, a feature

I

ll

e

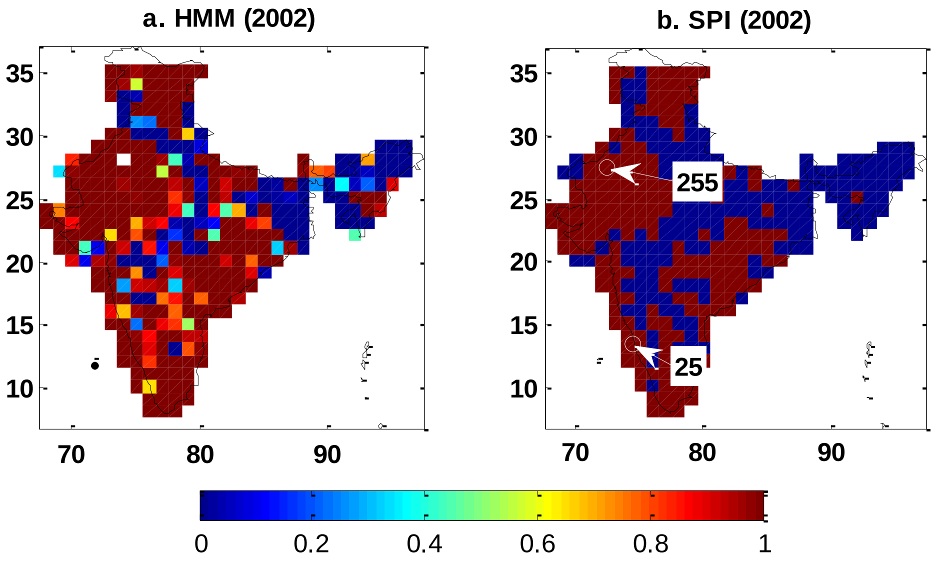


Figure 4. Moderate to extreme drought states identified by the HMM model (left panel) and the SPI (right panel) for 2002. The locations of grid points 25 (in the Western Ghats) and 255 (in Rajasthan) are also shown in the figure. The blue shaded grid points represent normal or above normal precipitation. The color shades on the left panel represent probability of drought according to the color bar.

The number of extreme, severe, and total (including moderate, severe and extreme) drought years during the available historical records (1901 to 2004) are shown in Figures 6, 7 and 8, respectively. The SPI transforms precipitation into a standard normal distribution, and hence each grid point has equal probability of having extreme or severe drought events. This is evident from the right panels of Figures 6, 7 and 8. Thus, SPI, owing to its formulation, cannot distinguish drought prone areas. In contrast, the HMM can identify drought prone areas - Fig. 6 indicates that Western Rajasthan and the Kutch region of Gujarat are more susceptible to extreme droughts, while Fig. 7 suggests that the remaining portion of Rajasthan, Gujarat, and north-interior Karnataka are more likely to have severe droughts.

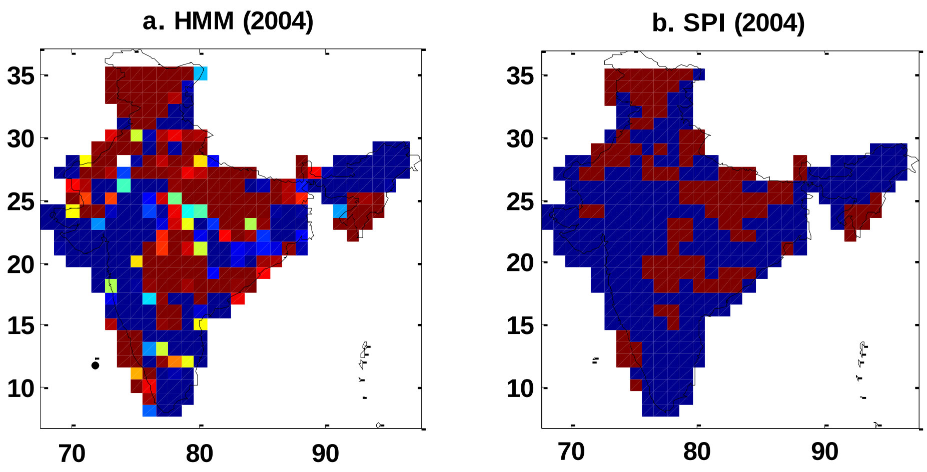


Figure 5. Moderate to extreme drought states identified by the HMM model (left

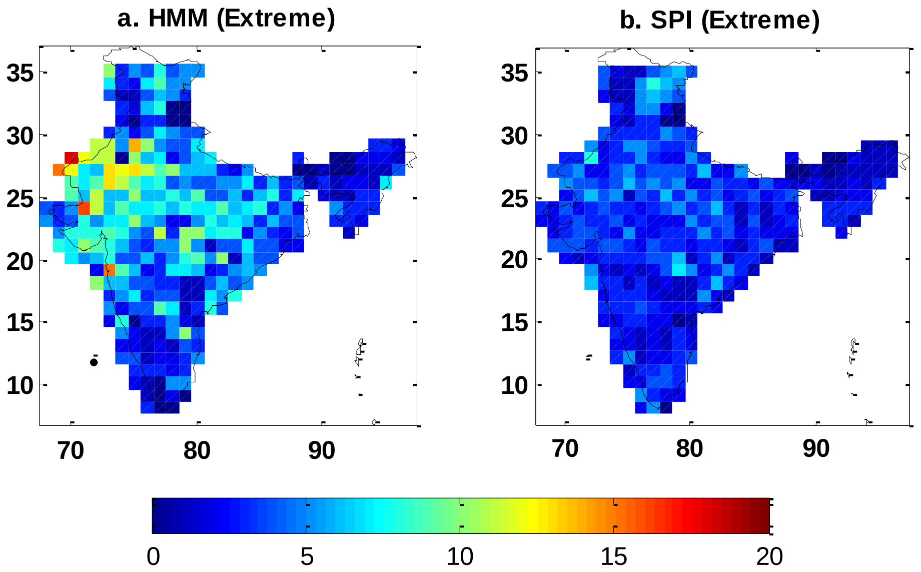


Figure 6. Number of extreme drought years during 1901 to 2004 identified by the HMM model (left panel) and SPI (right panel). For the HMM model, the years with probability of extreme drought greater than 0.9 are counted. For SPI, the years with SPI value less than -2.0 are counted.

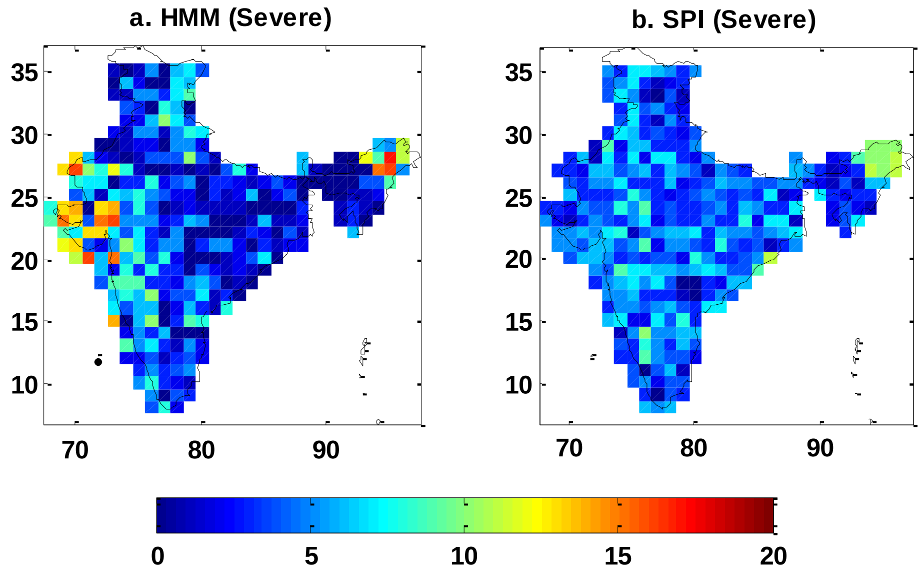


Figure 7. Number of severe drought years during 1901 to 2004 identified by

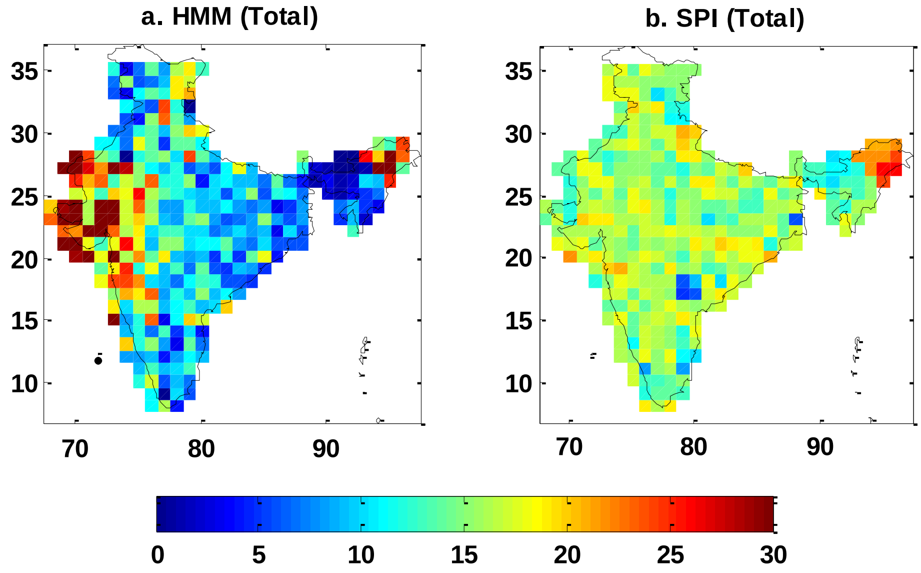


Figure 8. Number of total drought years (including moderate, severe and extreme droughts) during 1901 to 2004 identified by the HMM model (left panel) and SPI (right panel). For the HMM model, the years with sum of probability of moderate, severe and extreme droughts greater than 0.9 are counted. For SPI, the years with SPI value less than -1.0 are counted.

The number of drought years during 1901 to 2004 identified by SPI and HMM for different window sizes (1 to 36 months) all ending in September are shown in Fig. 9. For SPI, the years during which SPI value is less than -1.0 are counted at each grid point and for each window size. For HMM, the years during which the sum of probability of moderate, severe and extreme droughts exceeds 0.9 are counted. It is expected that with increase in window size, the number of drought years would increase because the droughts for longer time windows naturally exhibit prolonged durations. Since SPI drought classification is based on predefined thresholds and under an independence assumption of the data, the number of drought years is of the same order irrespective of the time scale. This is evident in Fig. 9 that shows boxplots of number of drought years for different time scales (1 to 36 months) at all grid points over India. The SPI, owing to its independence assumption, cannot distinguish between time-scales, while HMM shows an increasing trend in the number of severe drought years with increasing window size.

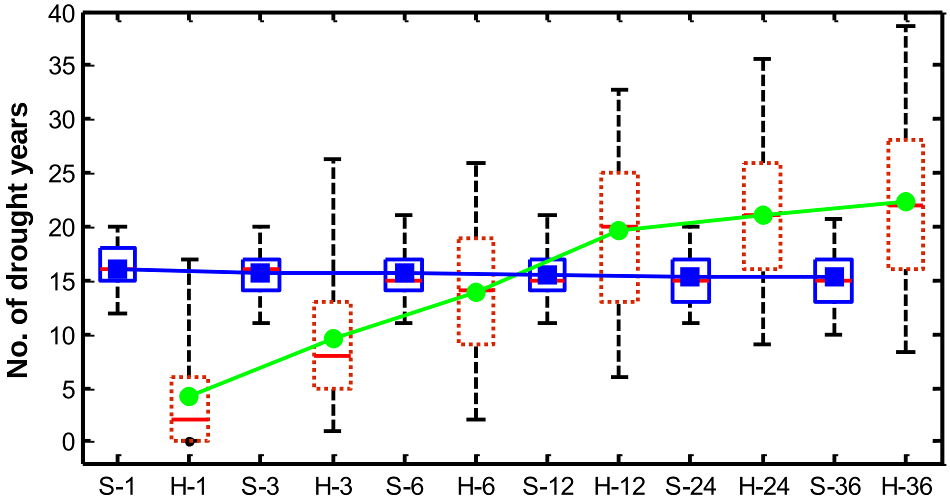


Figure 9. Number of drought years (including moderate, severe and extreme droughts) during 1901 to 2004 identified by the HMM (red dashed box) and SPI (blue bold box) over all 357 grid points for different time scales (1 month to 36 months). The symbol S-*w* and H-*w* demote SPI and HMM-DI for window size of *w* moths, respectively. The number of drought years averaged over all grid points identified by HMM-DI and SPI are denoted by green circles and blue squares, respectively.

5. CONCLUDING REMARKS

A hidden Markov model (HMM) based directed acyclic graph was used to develop a new index for assessing drought characteristics. The parameters of the HMM were estimated using the method of maximum-likelihood. The developed drought index (HMM-DI) was applied to precipitation data over India and the results were compared with the standard precipitation index (SPI). The results suggest that the HMM-based index has some advantages over the SPI - (i) the HMM can identify drought prone areas, (ii) the probabilistic classification of drought states by the HMM avoids some non-intuitive results given by the SPI, and (iii) the HMM relaxes the independence assumption made in the SPI, which is particularly useful for assessing longer duration droughts (window size greater than twelve months).

The developed HMM index appears to be a promising alternative to the existing drought indices. In addition to the above discussed advantages, the HMM index can assess drought characteristics in real-time and can be used to generate data for simulating droughts. Additionally, the graphical representation of HMM-DI can be exploited for investigating the relationships between droughts based on precipitation, streamflow and soil moisture, and for space-time identification of drought triggers.

ACKNOWLEDGMENTS

This study was supported in part by the National Science Foundation under Grants OCI 0753116 and AGS 1025430. This support is gratefully acknowledged. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] WMO. *International Meteorological Vocabulary*, Publication No. 182, second edition, World Meteorological Organization, Geneva, Switzerland, 1992.

[2] R. R. Heim. A review of twentieth-century drought indices used in the United States. *Bulletin of the American Meteorological Society*, 83(8): 1149–1165, 2002.

[3] T. T. Munger. Graphical method of representing and comparing drought intensities. *Mon. Weather Review,* 44, 642-643, 1916.

[4] G. Blumenstock Jr. Drought in the United States analyzed by means of the theory of probability. *USDA Tech. Bull.*, 819, 1942.

[5] J. McQuigg. A simple index of drought conditions. *Weatherwise,* 7, 64-67, 1954.

[6] W. C. Palmer. Meteorological drought. *Res. Paper No. 45, Weather Bureau, Washington, D. C.,* 1965.

[7] W. C. Palmer. Keeping track of crop moisture conditions, nationwide: The new Crop Moisture Index. *Weatherwise,* 21, 156-161, 1968.

[8] B. A. Shafer and L. E. Dezman. Development of a surface water supply index (SWSI) to assess the severity of drought conditions in snowpack runoff areas. *Proceedings of the Western Snow Conference,* 164-175, Colorado State University, Fort Collins, Colorado, 1982.

[9] T. B. McKee, N. J. Doesken and J. Kleist. The relationship of drought frequency and duration to time scales. *Proceedings of the Eighth Conference on Applied Climatology*, 179–184, Anaheim, CA, 1993.

[10] G. Mallya, S. Tripathi and R.S. Govindaraju. Assessment of drought characteristics using graphical models. *Frontiers of Interface Between Statistics and Sciences*, Hyderabad, India, pp. 565-575, 2009.

[11] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2): 257–286, 1989.

[12] M. Thyer and G. Kuczera. A hidden Markov model for modelling long-term persistence in multi-site rainfall time series 1. Model calibration using a Bayesian approach. *Journal of Hydrology*, 275(1–2): 12–26, 2003.

[13] S. Tripathi and R. S. Govindaraju. On the identification of intra-seasonal changes in the Indian summer monsoon. *Proceedings of the Third International Workshop on Knowledge Discovery from Sensor Data* (Sensor-KDD 2009), Paris, France, pp. 62–70, 2009.

[14] C. M. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics, first edition, Springer, New York, USA, 2006.

[15] M. Rajeevan, J. Bhate and A.K. Jaswal. Analysis of variability and trends of extreme rainfall events over India using 104 years of gridded daily rainfall data. *Geophysical Research Letters*, 35, L18707, doi:10.1029/2008GL035143, 2008.

[16] M. Rajeevan, J. Bhate, J. D. Kale and B. Lal. High resolution daily gridded rainfall data for the Indian region: analysis of break and active monsoon spells. *Current Science*, 91(3): 296–306, 2006.

[17] S. Gadgil, M. Rajeevan and R. Nanjundiah. Monsoon prediction-why yet another failure? Current Science 88:1389–1400, 2005.

Automated Orbital Mapping of Mars

Brian P. Kent, Alessandro Rinaldo, and David Wettergreen

Abstract:

The Mars geologic mapping process lags behind the collection of Mars imagery and elevation data. We develop and test a statistical system that automatically generates geomorphic maps of uncharted Mars regions in order to help close this gap. By providing “first draft” maps, we hope to allow geologists to avoid repetitive mapping tasks and to focus on areas of high scientific interest.

We view the mapping process as a supervised learning problem. In the most general form, our system uses a set of Martian scenes and a statistical classifier to learn the relationship between spectral and topographic information on one hand and landform classes on the other, then uses the relationship to predict the geomorphic map for an uncharted scene. We explore how the predictive accuracy of the system varies depending on the choices of training set, classification method, included covariates, and scene segmentation.

Previous work by Ghosh, Stepinski, and Vilalta [1, 2] showed that multiple classification algorithms could accurately predict the landform type of test superpixels—contiguous regions of uniform landform treated as a single data point —when trained on topographic information from superpixels in the same scene. We build on these results by defining a more universal partition of landform classes, by training our classifiers on many different scenes and testing on an entirely new scene, and by augmenting topographic information from the Mars Orbiter Laser Altimeter (Mars Global Surveyor) with spectral intensity data from the High Resolution Stereo Camera (Mars Express). These changes allow our system to predict more general scenes.

When our classifiers are trained and tested on superpixels from the same scene and only topographic information is used, we achieve slightly less accurate predictions than Ghosh, et al., although the results are still considerably better than the naïve most common class (MCC) prediction. When we train the classifiers on many scenes at once and predict an entirely different scene the accuracy for each classifier drops but still exceeds the MCC accuracy for most methods. Furthermore, when we add spectral information to the training set, prediction accuracies for the Adaboost, k-nearest neighbors, and support vector machine methods jump substantially to about 80%, compared to 50% accuracy for the baseline MCC method (Figure 1).

Future work will focus on expanding the training set to more than seven scenes while maintaining high data quality, using clustering techniques to identify a data-driven partition of Mars geomorphology into discrete classes, and reincorporating previous efforts to use more complex superpixel features and belief propagation to share predictive information between neighboring superpixels.

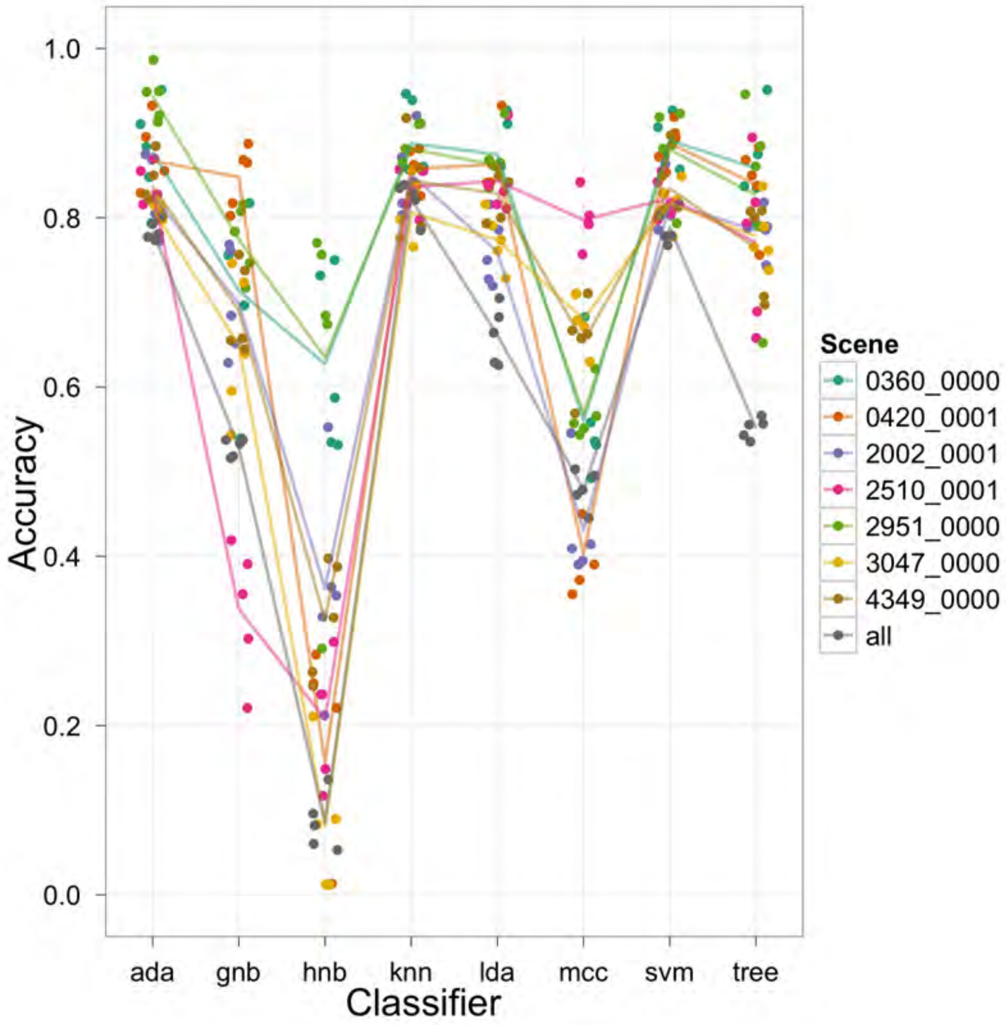


Figure 1: 5-fold cross-validation prediction accuracy for several classifiers, including Adaboost (ada), Gaussian naive Bayes (gnb), histogram naive Bayes (hnb), k-nearest neighbors (knn), linear discriminant analysis (lda), support vector machine (svm), and decision tree (tree). The baseline method is to predict all test superpixels as the most common class in the training set (mcc). Color indicates the training and testing scene, except for gray which indicates cross-validation runs where each classifier is tested on an entirely new scene. The solid lines indicate mean accuracy for each scene.

[1] S. Ghosh, T.F. Stepinski, and R. Vilalta. *Automatic Annotation of Planetary Surfaces With Geomorphic Labels*. IEEE Transactions on Geoscience and Remote Sensing. **48** (2010), no. 1, 175-185.

[2] T.F. Stepinski, S. Ghosh, and R. Vilalta. *Automatic Recognition of Landforms on Mars Using Terrain Segmentation and Classification*. In Lecture Notes in Artificial Intelligence, 4265, pp. 255-266, Springer Berlin/Heidelberg, 2006.

A STUDY OF TIME SERIES NOISE REDUCTION TECHNIQUES IN THE CONTEXT OF LAND COVER CHANGE DETECTION

XI CHEN[†]*, VARUN MITHAL[†]*, SRUTHI REDDY VANGALA[†]*, IVAN BRUGERE[†]*, SHYAM BORIAH*, AND VIPIN KUMAR*

The purpose of this study is to introduce concepts relevant to performance of (i) change detection algorithms within (ii) various regional contexts with differing noise characteristics according to (iii) differing strategies of noise reduction. The relevant interrelations of these three elements are presented, and focused analysis is presented from the perspective of varying (i) and (iii) for a comparative analysis across (ii).

Six smoothing methods has been studied in this work: Savitzky-Golay (SG) method [7], The Savitzky-Golay method iterated to upper envelope (SG-Itr) [3], Harmonic Analysis of Time Series (HANTS) [6], Double Logistic function fitting method (DL) [1], Data Assimilation method(DA) [5]and a naive outlier identification and imputation scheme (SO).

In this work, we enumerate three general data characteristics, especially relevant in the MODIS EVI data, which a given noise reduction technique may take advantage of: neighborhood coherence, quality annotation and background model.

For a noise reduction technique we identify the following two questions to be of relevance:

- Which observations in the time series should be imputed?
- How are these observations to be imputed?

Based on the first question, the reviewed methods can then be organized into (1) *selective* and (2) *non-selective* imputation methods. Selective methods identify some observations that they consider noisy and ought to be imputed. On the other hand, in the non-selective methods every observation is imputed. We consider the selective methods to be more conservative as they modify fewer observations as opposed to the non-selective methods which modify every observation and therefore no processed data value corresponds to the real observation. Intuitively, if an observation is not clearly anomalous and is annotated as a high quality observation, the value reported by the MODIS is as trustworthy as can be ascertained. Time series smoothing methods should thus be considered the most aggressive because generally every observation of the original time series is modified without identifying trustworthy observations. Note that typically the imputations of selective methods will modify the observation by large magnitude because large outlier values are imputed in this case. The non-selective methods are less conservative and modify each value but the total modification in the value itself is generally of smaller magnitude for most observations.

Imputation is done primarily based on the three characteristics of neighborhood coherence, quality annotation, and background model. Most of the non-selective methods rely only on neighborhood coherence and use function fitting on temporal neighbors to eliminate the noise. In contrast, some of the selective methods such as DA does not account for the temporal coherence. While all three properties play an important role when removing noise yet there is no method that uses them all.

In our study, we present two noise characteristics in varying degress in different regions. The effectiveness of noise reduction for change detection methods is closely related to the susceptibility of these methods to these characteristics. First, *unbiased noise* of relatively small amplitude exists as a component of each observation due to variations in atmospheric conditions or instrument imprecision. This noise causes neighboring observations to be arbitrarily different from each other due

[†] These authors contributed equally to this work.

*University of Minnesota, <chen,mithal,vangala,ivan,sboriah,kumar>@cs.umn.edu.

to phenomena other than vegetation growth, where no land cover change has occurred. Second, the presence of relatively large, positively or negatively *biased noise* produces anomalous observations which do not follow the phenological trend of the time series. Often these observations are annotated with a low quality flag (QA) but sometimes may not be recorded accurately in the QA annotation.

Change Detection methods are impacted by both biased and unbiased noise in the data. A naive algorithm using observation-wise comparisons between the same months in two years will be severely impacted by biased noise and raise many false alarms. Therefore, most algorithms, including those used in this study, consider a more robust statistic like the average over an entire year for change detection. The Manhattan Delta [4] and Yearly Delta algorithms [2] are impacted by biased noise as it can increase the distance considerably and give a false appearance of change in EVI. The Yearly Delta algorithm is robust to unbiased noise in the data as averaging of it tends to be approximately zero. However, the Manhattan Delta algorithm is additively impacted by the unbiased noise.

In this paper we have shown that the interrelations between noise characteristics endemic to differing data regions, change detection methods, and noise reduction methods. We have provided contrasts between selective and non-selective imputation methods and their effects on biased and unbiased noise characteristics. We conclude that less conservative, non-selective noise reduction methods generally follow more conservative, selective methods to improve results. Conversely, we conclude that non-selective methods tend to perform poorly in the presence of positively or negatively biased noise. Depending on the susceptibility of the change detection method to each of these noise characteristics, either smoothing or outlier detection may not be necessary.

For an extended version of this study which includes a detailed discussion on noise reduction algorithms, noise characteristics of vegetation index data, as well as a comprehensive experimental evaluation, we refer the reader to [4].

Acknowledgements

The research described here is supported by the Planetary Skin Institute, NSF Grant IIS-0713227, NSF Grant IIS-0905581, NASA grant NNX09AL60G and the University of Minnesota MN Futures Program. Access to computing facilities was provided by the University of Minnesota Supercomputing Institute.

References

[1] P. S. Beck, C. Atzberger, K. A. Høgda, B. Johansen, and A. K. Skidmore. Improved monitoring of vegetation dynamics at very high latitudes: A new method using modis ndvi. *Remote Sensing of Environment*, 100(3):321 – 334, 2006.

[2] S. Boriah. *Time Series Change Detection: Algorithms for Land Cover Change*. PhD thesis, University of Minnesota, Twin Cities, April 2010.

[3] J. Chen, P. Jonsson, M. Tamura, Z. Gu, B. Matsushita, and L. Eklundh. A simple method for reconstructing a high-quality ndvi time-series data set based on the savitzky-golay filter. *Remote Sensing of Environment*, 91(3-4):332 – 344, 2004.

[4] X. Chen, V. Mithal, S. R. Vangala, I. Brugere, S. Boriah, and V. Kumar. A study of time series noise reduction techniques in the context of land cover change detection. Technical Report 11-016, University of Minnesota, Department of Computer Science and Engineering, 2011.

[5] J. Gu, X. Li, C. Huang, and G. S. Okin. A simplified data assimilation method for reconstructing time-series modis ndvi data. *Advances in Space Research*, 44(4):501 – 509, 2009.

[6] G. J. Roerink, M. Menenti, and W. Verhoef. Reconstructing cloudfree ndvi composites using fourier analysis of time series. *International Journal of Remote Sensing*, 21(9):1911 – 1917, 2000.

[7] A. Savitzky and M. Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, 336(8):1627 – 1639, 1964.

UNDERSTANDING THE SEA SURFACE TEMPERATURE - TROPICAL CYCLONE RELATIONSHIP: A DATA-DRIVEN APPROACH

JAMES H FAGHMOUS*, STEFAN LIESS**, AUROOP GANGULY***, MICHAEL STEINBACH*,
FRED SEMAZZI****, AND VIPIN KUMAR*

Extended Abstract

Global climate change and its effect on Atlantic tropical cyclone (TC) activity has become one of the most contested issues in climate science. The difficulty of attributing a change in TC frequency to global climate change stems from the lack of reliable historical data as well as the large amplitude fluctuations in present-day storms. Understanding future TC activity is crucial, especially in light of TC's potential role in the ocean's poleward heat transport [10, 2], impact on marine ecosystems [6], and increasing destructiveness [1, 12].

Currently, a theory of TC formation (cyclogenesis) under climate change is still not fully understood [7, 2], which makes predicting future TC frequency highly uncertain. Existing high-resolution climate models fail to consistently predict an increase or decrease in the total number of TCs in a warming environment. Globally, the majority of global circulation models (GCMs) forecast a decrease in the total number of TCs as the atmosphere continues to warm. At the individual basin level, however, regional circulation models (RCMs) have been significantly more uncertain with projected changes of up to $+/-50\%$.

In this work we attempt to gain a better understanding into Atlantic cyclogenesis by leveraging the recently available climate and TC data. First, we show that not all regions in the Atlantic are equally important when it comes to cyclogenesis. Previous work monitoring TC trends employed basin-wide averaging to study the sea surface temperature (SST)-TC relationship. Webster *et al.* [12] conducted a basin-wide analysis of SST and TC trends in all major basins and concluded that the recent increase in Atlantic TC activity could not be attributed to SST alone.

Our work proposes that instead of analyzing trends across an entire basin, which spans thousands of miles, it might be more informative to focus on smaller and more meaningful regions in the Atlantic to better capture SST's relationship to Atlantic TCs. To accomplish this, we designed a systematic search across the Atlantic basin to find SST and cyclogenesis regions with the highest seasonal SST and TC frequency correlation. To identify such regions, we implement a linear optimization algorithm that searches the Atlantic for pairs of SST-Cyclogenesis regions of any (reasonable) size that better explain the SST-TC relationship compared to basin-wide averaging.

When we run our algorithm on the SST of months preceding the TC season (May-June), we find the region off the West African coast near 20°- 30°N to have a significant correlation (0.55; $p < 0.05$) with TC frequency of the following season (June-October). Similarly, when we apply our algorithm to seasonal (June-October) SST averages we find the region westward of $10° - 20°$N between 18° and 60°W has the highest correlation of seasonal TC counts (0.66; $p < 0.05$). Both of these regions correlate better with seasonal TC counts that basin-wide averaging (0.43).

Interestingly, the region westward of $10° - 20°$N between 18° and 60°W is part of the Atlantic's main development region (MDR), a region from where nearly 60% of all tropical storms and 85% of major hurricanes originate [4]. The majority of storms within the MDR form from atmospheric

*Department of Computer Science, The University of Minnesota.
**Department of Soil, Water, and Climate, The University of Minnesota.
*** Oak Ridge National Laboratory.
****Department of Marine, Earth, and Atmospheric Sciences, North Carolina State University.

African easterly waves (AEW). Therefore seasonal TC activity is highly sensitive to the climatology of the MDR and AEW activity. Although the number of AEWs per year is fairly constant [3], the percentage of developing AEWs is not [4, 11]. Recently, Sall *et al.* [8] and Hopsch *et al.* [5] found that AEWs are weakened by dry mid-to-upper level air traveling from higher latitudes (Europe and North Africa). More specifically, AEWs exiting from the African continental landmass tend to ingest dry air descending from middle latitudes and dissipate without experiencing cyclogenesis. We propose that the warming of the West African coast near 20°- 30°N prior to the TC season (May-June) provides the additional moisture necessary to counter the higher latitude dry air linked to suppressing AEWs and therefore increase the chance that AEWs advance deep into the Atlantic and develop into TCs. In a similar fashion the warming of the region along the AEW path (westward of $10° - 20°$N between 18° and 60°W) provides favorable conditions for cyclogenesis and therefore can explain the SST-TC relationship better than basin-wide averaging.

These findings suggest that the warming of the Atlantic off the West African coast near 20°- 30°N prior to the TC season, as well as the warming westward of $10° - 20°$N between 18° and 60°W during the TC season have a pronounced effect on TC formation. Furthermore, abnormal SST averages in the regions highlighted above could explain (in)active TC seasons. Our approach therefore, can be used to objectively identify more meaningful regions than mere basin-wide averaging. Finally, the difference in TC activity between the Atlantic and Pacific highlighted in [12] could be explained by the difference in AEW and their Pacific counterparts. On the one hand, Pacific easterly waves (PEWs) are driven primarily by convective heating, which depends on SST. On the other hand, the barotropic to baroclinic conversion, which is the energy transport from the mean flow toward the rotational flow component, dominates AEWs. This means that AEWs are strongly associated with rotation and therefore cyclogenesis, unlike PEWs, which are related to convection (*i.e.* the vertical flow) [9].

Acknowledgments. This work was supported by NSF grants IIS-0905581 and IIS-1029711. James H Faghmous is supported by an NSF Graduate Research Fellowship. We would like to thank Jaya Kawale for help with the TC count data and Ryan Haasken for generating figures for the poster.

References

[1] K. Emanuel. Increasing destructiveness of tropical cyclones over the past 30 years. *Nature*, 436(7051):686–688, 2005.

[2] K. Emanuel. The hurricane-climate connection. *Bulletin of the American Meteorological Society*, 89(5), 2008.

[3] N. Frank. Atlantic tropical systems of 1974. *Monthly Weather Review*, 103:294, 1975.

[4] S. Goldenberg, C. Landsea, A. Mestas-Nuñez, and W. Gray. The recent increase in atlantic hurricane activity: Causes and implications. *Science*, 293(5529):474, 2001.

[5] S. Hopsch, C. Thorncroft, and K. Tyle. Analysis of african easterly wave structures and their role in influencing tropical cyclogenesis. *Monthly Weather Review*, 2009.

[6] I. Lin, W. Liu, C. Wu, G. Wong, C. Hu, Z. Chen, W. Liang, Y. Yang, and K. Liu. New evidence for enhanced ocean primary production triggered by tropical cyclone. *Geophys. Res. Lett*, 30(13):1718, 2003.

[7] R. Pielke Jr, C. Landsea, M. Mayfield, J. Laver, and R. Pasch. Hurricanes and global warming. *Bulletin of the American Meteorological Society*, 86(11):1571–1575, 2005.

[8] S. M. Sall, H. Sauvageot, A. T. Gaye, A. Viltard, and P. D. Felice. A cyclogenesis index for tropical Atlantic off the African coasts. *Atmospheric Research*, 79(2):123–147, 2006.

[9] Y. Serra, G. Kiladis, and M. Cronin. Horizontal and vertical structure of easterly waves in the Pacific ITCZ. *Journal of the Atmospheric Sciences*, 65(4):1266–1284, 2008.

[10] R. Sriver and M. Huber. Observational evidence for an ocean heat pump induced by tropical cyclones. *Nature*, 447(7144):577–580, 2007.

[11] C. Thorncroft and K. Hodges. African easterly wave variability and its relationship to Atlantic tropical cyclone activity. *Journal of Climate*, 14(6):1166–1179, 2001.

[12] P. J. Webster, G. J. Holland, J. A. Curry, and H. Chang. Changes in tropical cyclone number, duration, and intensity in a warming environment. *Science*, 309(5742):1844 –1846, 2005.

INTERACTIONS BETWEEN TELECONNECTIONS

JAYA KAWALE*, STEFAN LIESS**, MICHAEL STEINBACH*, PETER SNYDER**, AND VIPIN KUMAR*

Abstract. Dipoles represent a class of teleconnections or long range spatio-temporal dependencies in climate data characterized by anomalies of opposite polarity at two locations at the same time. This dipole phenomenon has been known to occur for more than a century and the dipoles are crucial as they impact climate changes throughout the globe. For example, the El Niño Southern Oscillation (ENSO) is responsible for remote climate variations like temperature changes, increased rainfall, thunderstorms, tropical cyclones and droughts. Despite the importance of these dipole teleconnections for predicting regional climate anomalies and severe hydro-meteorological events, they have so far been mainly studied in isolation, and interactions between dipoles have been considered to be weak. In this paper, we study the interactions between the different pressure dipoles by examining the dipole activity of four major dipoles during the three phases of the ENSO namely El Niño , La Niña and neutral. Our results show significantly different dipole characteristics during the three phases.

1. Introduction

Pressure dipoles represent important long distance teleconnection phenomena characterized by two locations having pressure anomalies in the opposite direction. Dipoles are often defined by climate scientists using two fixed locations. For e.g., the El Niño Southern Oscillation is defined based on two locations: Tahiti and Darwin, Australia. The strength of a dipole is measured by its *index* which is computed by taking a difference in the pressure anomaly series of the two locations. The two phases of the ENSO and are called El Niño, which corresponds to warming in the eastern equatorial Pacific, and La Niña, which corresponds to the cooling in this region. Understanding these dipoles and their interplay is crucial to understand the variability of the global climate system. In this paper, we study the impact of the two phases of ENSO on the four dipoles - North Atlantic Oscillation (NAO), Arctic Oscillation (AO), Western Pacific (WP) and Pacific North American pattern (PNA).

2. Experiments and Results

2.1. **Methodology.** In this paper we use the approach by Kawale *et. al*[1] to find and study dipoles. It is a Shared Reciprocal Nearest Neighbor clustering based approach which finds all the dipole cluster pairs in the data in a single snapshot image and thus overcoming the limitations of the previous iterative approaches.

2.2. **Results.** For our dipole analysis, we use pressure climate data from the NCEP/NCAR Reanalysis which has data assimilated from 1948 – present which is available for public download. In order to find dipoles we focus on sea level pressure. We also use the precipitation and air temperature data to study the impact of the dipoles on land. In order to study the interactions between the ENSO and the other dipoles, we first separated the data into the three different phases, namely (1) El Niño, (2) La Niña and (3) neutral phase. Out of the 62 years of data, El Niño data spans over 16 years, La Niña spans over 14 years and the neutral phase spans around 30 years.

*Department of Computer Science, University of Minnesota kawale, steinbac, kumar@cs.umn.edu
Department of Soil, Water and Climate, liess, pksnyder@umn.edu.

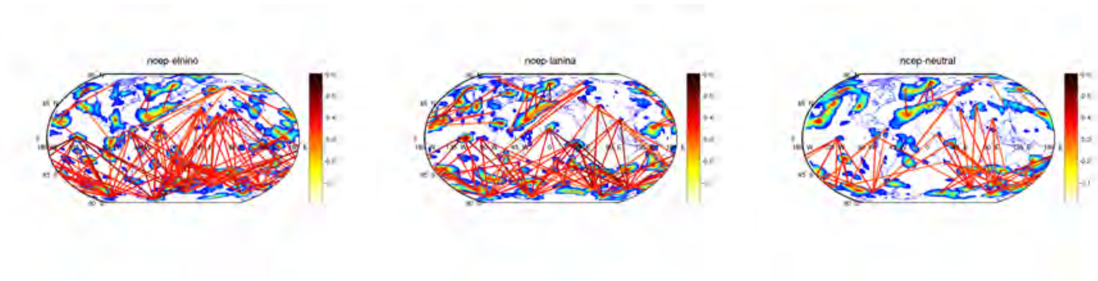


FIGURE 1. Dipole connections on the globe during the three periods and the overall period using a correlation threshold of -0.3

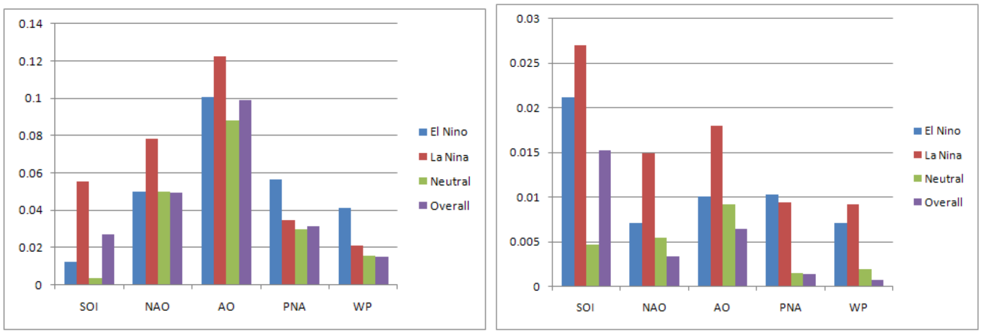


FIGURE 2. Left figure shows the impact on land temperature and the right figure shows the impact on precipitation during the 4 periods.

2.2.1. *Comparison of Network Characteristics.* We constructed complex network consisting of pairwise correlation among all the nodes for all the data in the three periods respectively. Next, we find the dipoles using the Shared Reciprocal Nearest Neighbour(SRNN) based algorithm. Figure 1 shows the interconnections between the different dipole centers. We observe that the number of connection is much more in El Niño and La Niña compared to the neutral phase. This indicates that the dipole activity is much stronger during these two phases.

2.2.2. *Impact on land.* The most crucial aspect of dipoles is that they may cause regional climate anomalies and extreme climate events. In order to measure the impact, we consider the area weighted correlation of the dipole index on temperature and precipitation anomalies of land. Figure 2 shows the aggregate area weighted impact of the different dipoles on land temperature and precipitation during different phases using the static dipole index used by climate scientists. Overall this result shows that the impact of SO, NAO, AO on temperature during the La Niña phase is much higher as compared to the other phases. Also the impact of PNA, WP on land temperature is strikingly higher in the El Niño phase. The NAO and AO have a dramatically higher impact on precipitation during the La Niña phase as compared to the El Niño and neutral phase. Also PNA and WP show a much higher impact during both El Niño and La Niña phase.

2.3. **Acknowledgments.** This work was supported by NSF grants IIS-0905581 and IIS-1029771. Access to computing facilities was provided by the University of Minnesota Supercomputing Institute.

REFERENCES

[1] J. Kawale, S. Liess, A. Kumar, M. Steinbach, N. Ganguly, A. Samatova, F. Semazzi, P. Snyder, and V. Kumar. Data driven discovery of dynamic dipoles in climate data. In *Submitted*, 2011.

MINING AVIATION SAFETY REPORTS USING PREDICTIVE WORD SEQUENCES

PAUL MELBY*

Abstract. This paper describes a text mining sofware program that has been developed for use by the Aviation Safety Information Analysis and Sharing (ASIAS) program, an industry-wide effort to promote aviation safety. The text mining program, called The ASIAS Information Retrieval and Extraction System (AIRES), implements an algorithm that discovers word sequences from aviation safety incident reports that are highly predictive of a user-specified safety topic. This algorithm is similar to an Apriori algorithm, except that it uses predictive power, rather than frequency, as the pruning criteria. We show that this method reduces the number of word sequences that require evaluation by up to a factor of 10, while still discovering over 90% of all the highly predictive word sequences. Additionally, we discuss the software implementation of this algorithm in AIRES, a tool designed to allow subject matter experts an efficient process for searching through incident reporting data, as well as recording their own categorization of the reports.

1. Introduction

The Aviation Safety Information Analysis and Sharing (ASIAS) program is an industry-wide effort to promote commercial aviation safety. The primary objective of ASIAS is to discover common, systemic safety problems that span multiple airlines, fleets and regions of the national air transportation system. ASIAS leverages Federal Aviation Administration (FAA) data, de-identified airline safety data and other government and publicly available data sources. Many of these data sources are textual. A primary source of these are Aviation Safety Action Program (ASAP) reports that are shared with ASIAS by participating airlines. ASIAS also utilizes NASA's Aviation Safety Reporting System (ASRS) data, Service Difficulty Reports (SDR) filed by the airlines and provided by the FAA, and other sources of textual data. In many of these data sources, there are extensive taxonomies that represent the safety issues and incidents in the reports as well as contributing factors and demographic information. For example, the ASIAS ASAP taxonomy includes well over 400 fields of structured information.

One recurring challenge with any of these data sources is the poor quality of the structured fields. In many cases, a subject matter expert reading the narrative portions of an ASAP report would identify several contributing factors or safety incidents that are discussed in the report, but are not marked as such in the corresponding structured field. These omissions of categorization cause a challenge for both finding all the reports on a given topic area, as well as performing analysis on the structured fields, such as finding which contributing factors commonly co-occur with each incident type.

This abstract describes a software program developed to solve this problem, the ASIAS Information Retrieval and Extraction System (AIRES). AIRES makes a comparison of positively and negatively labeled records and discovers words and word sequences that have predictive power for selecting positively labeled records. The algorithm discovers word sequences such as *('crossed', 'hold', 'short')* which represent a sequence where the word *'crossed'* is followed by *'hold'* which is followed by *'short'*. The sequences may contain extra words, or gaps, in between and still match the sequence, so a statement such as *'crossed just over the hold short'* would match the sequence despite the extra words *'just over the'* in between *'crossed'* and *'hold'*. The discovered word sequences

*The MITRE Corporation, 7515 Colshire Dr., McLean, VA 22102, pmelby@mitre.org.

have a very high precision and can be useful in classifying reports that are positively labeled in a category. In addition, these sequences extrapolate well to find reports that should be positively labeled but are not without returning too many reports for the analyst to review. This enables an active learning approach where a subject matter expert can review only the most relevant reports, improving the overall efficiency of the search and validation process [**?**].

There have been many previous approaches to finding highly predictive word sequences in a document collection. In order to deal with the high dimensionality of word sequences, many of these focus on finding frequent sequences [**?**, **?**, **?**]. Variations include finding maximal frequent word sequences [**?**, **?**] and closed frequent word sequences [**?**]. While these approaches are useful, in many cases in aviation safety reports, there are highly precise word sequences that relatively rare. These can often be identified by subject matter experts but this requires a lot of manual effort. In this paper, we describe an alternative approach to finding highly predictive word sequences that grows patterns based on their predictive power, not their frequency.

2. Discovering Highly Predictive Word Sequences

The underlying algorithm for the discovery of predictive word sequences, or patterns, relies on the fact that all of the categories in the ASAP archive are relatively rare, which makes comparison of positively labeled reports and negatively (or unlabeled) reports a fruitful method of finding good search patterns. In this way, at any point in time, measures of the predictive power of a particular word or pattern, such as relative information gain or f-measure, while not entirely accurate, should still be able to find useful features in the data. To illustrate the princple, if labeled positive reports for a class only constitute 1% of the total reports, then even if there are an additional 1% of reports that are unlabeled, a term that exists in all 2% of true positive reports still provides a large amount of information about the classification since it increases the fraction of reports in the positive class from 1% to 50%. This is a key assumption for this approach and may not apply for data sets that have balanced classes.

One of the challenges in finding word sequences is the vast number of possible word combinations that exist when gaps are allowed to exist between the words. Many approaches to finding such word sequences therefore focus on finding frequent word sequences, using an Apriori algorithm to search for all sequences of at least a specified frequency within the text. Then, given all frequent word sequences, a feature selection algorithm can choose those that have the highest predictive power. While this approach can find very useful features, including those of very long length, it may not be able to discover highly precise sequences that occur in relatively few reports. In contrast, AIRES uses the predictive power of each candidate pattern as the pruning criteria and is therefore able to find highly predictive patterns that do not occur frequently. Algorithm 1 illustrates the workings of the word sequence discovery algorithm.

Algorithm 1: Word Sequence Discovery.

```
Input: A list of categorized reports, the mininmum word frequency, $N$, and the minimum
       information gain for a word sequence, $IG_{min}$, the maximum number of words in a
       sequence, $L$ and the length of the gaps allowed between words, $X$
Output: A list of gapped word sequences that are highly predictive of the target category
Determine all words that exist within the data;
Optionally apply synonyms, stopwords, and stemming;
Words = all words that occur greater than N times;
Candidate Patterns = words that have a minimum relative information gain of $IG_{min}$;
while The length of the longest Candidate Patterns $< L$ do
    Find larger patterns like ⟨word⟩ followed by ⟨pattern⟩ within $X$ words or ⟨pattern⟩ followed
    by ⟨word⟩ within $X$ words;
    Eliminate the candidate patterns that occur less than N times;
    Eliminate candidate patterns with less than $IG_{min}$ relative information gain;
end
```

The minimum information gain, IG_{min} is determined by the user but is typically set at 0.5% of the entropy of the categorization. Although there is a pruning of infrequent patterns, the threshold is typically set very low to remove unique patterns that would not have high predictive power. In a typical analysis, there may be 10-15,000 words that occur more than 3-5 times, but only 200-500 that have greater than 0.5% relative information gain. Because the sequences are pruned based on predictive power, there is no guarantee that all highly predictive word sequences will be discovered. Figure 1a shows the effect of varying IG_{min} on the percentage of all highly predictive word sequences that are discovered. These percentages found by comparing the number of sequences found with a specified information gain (in this case, 2% and 5%) with no pruning ($IG_{min} = 0$) to the number of sequences found with pruning at various IG_{min}'s. The results in the figure are averaged over 3 different test cases with ASRS data. As can be seen, even for a very high level of pruning, $IG_{min} = 1.5\%$, over 90% of all the sequences with over 5% information gain are discovered. The impact on sequences with less predictive power is greater, but still over 90% for $IG_{min} = 0.5\%$. Figure 1b shows the corresponding reduction in the number of sequences that require evaluation during the discovery process. Even at the lowest level of pruning depicted in the figure, there is a savings of nearly a factor of three over the case with no pruning at all. For the highest level of pruning, there is an order of magnitude difference in the required number of sequences to be evaluated, with only 9.8% the number of sequence evaluations being required.

3. Results and Implementation

To implement this approach, a software program was developed that streamlines the process and allows subject matter experts to search for reports using the discovered word sequences. Figure **??** shows a screenshot of the GUI interface for the tool. Users are presented with patterns that were discovered to have high predictive power and several statistical measures of that predictive power (information gain, precision, recall and weighted f-measure). Additionally, there are statistics on the overall performance of all patterns in the set to find relevant reports. The report viewer shows the report to the user and highlights the patterns where they occur. The graphical interface allows the user to select subsets of the data to focus on. The analyst can therefore focus on the "false positives," for example, that may be the most likely reports to be positives that were not labeled that way in the raw data. Each report is given a relevance score by adding up the information gain for each sequence that is contained in the report. The reports selected to be viewed are then sorted by this relevance score. In this way, the most relevant reports are viewed by the analyst first. Additionally, the user can validate the categorization of reports. This allows the discovery algorithm to perform

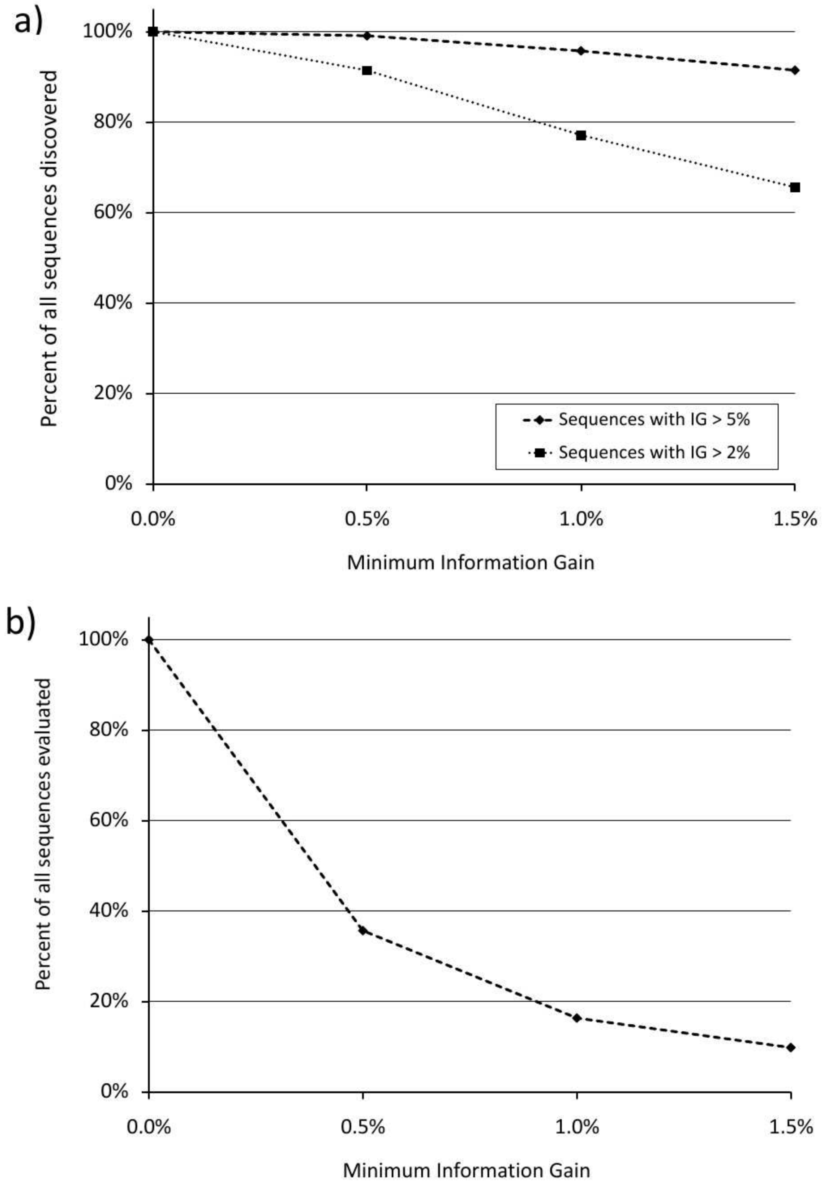


FIGURE 1. a. Average percentage of all highly predictive sequences discovered as a function of the minimum information gain threshold parameter. Percentages are shown for sequences with a relative information gain of greater than 5% and 2%. b. Percent of all sequences of length 3 or less that are evaluated by the algorithm as a function of the minimum information gain threshold parameter.

better, as it has an improved "best guess" of the categorization for each document. When the user is satisfied with the overall performance of the search process, the results can be exported for further review or analysis.

Figure 2 shows the results of performing the search process on three different topics within the ASIAS ASAP database. The user began each of these with a query based on an existing structured field, then reviewed the "false positives" for whether they should be included as part of the topic. In the figure, Precision is the cumulative precision given all reports read and validated by the user after having read n reports. Recall is the ratio of the number of positive reports identified in n to the total number identified in the entire search process. For all three topics, the precision starts out at 1, which indicates that the reports given the highest relevancy ranking tended to be in the topic. As the user reviewed the reports, fewer and fewer positive reports were found and the precision began to slip. Overall, the final precision before the user decided that enough positive reports were found ranged from 0.42 to 0.46. For the three topics, the initial percent of reports labeled as positive were 2.8%, 4.8%, and 4.7%. The search process found an additional 1.7%, 1.8% and 1.2% of the total reports that matched the corresponding topic.

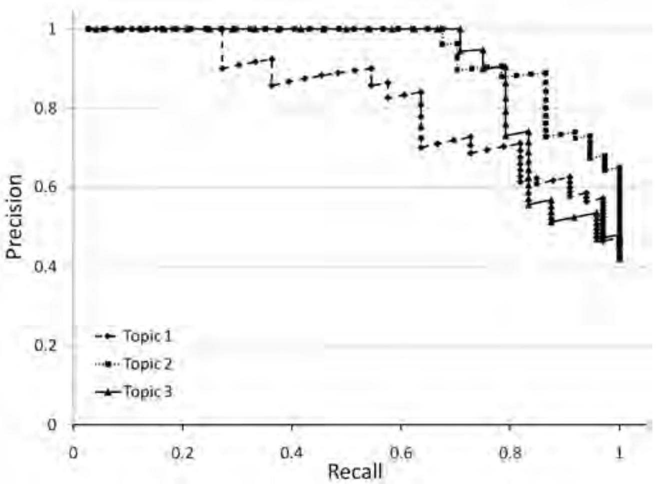


Figure 2. Precision versus Recall for user-reviewed reports in three different ASAP topics. Each data point represents a single report reviewed by the user. As the user reviews reports and adds them to the category, the overall recall gets higher, while the precision tends to drop over time as fewer and fewer additional positive reports are found.

4. Summary

In this paper, we described an approach to finding highly predictive word sequences from text that grows the word sequences based on their predictive power, rather than frequency of occurence. This approach requires many fewer evaluations than what would be required for finding all sequences of a specified frequency. The resulting speedup has enabled the development of an interactive software program that can be used by subject matter experts in order to search for and validate safety incident reports. In the current implementation, information gain is suggested as the measure of predictive power, but other measurements of predictive power can be used as well, such as F-measure or lift.

One advantage of using word sequences as features over other methods, such as individual words or feature-vectors such as from Latent Semantic Analysis is that they are very easily understood by the subject matter experts who use the tool to search for and validate reports. Many of these word sequences are actually standard aviation phraseology and resonate with the user. Additionally, the highlighting of the report text corresponds in a one-to-one fashion with the decision process the program used which makes it easy for the user to understand why a report is being presented for review.

In addition to a user-defined safety topic, the user-defined query could select the most recent reports as "positives." In this case, the word sequences that are discovered will often times be related to emerging trends in the data, or new air traffic control procedures. This mode is useful for discovering new safety issues that may need to be monitored. A future release of AIRES will include a more robust interface for searching for emerging safety issues. Additionally, the word sequences that are discovered can be used as features for classification and clustering algorithms.

5. Acknowledgements

The author would like to thank Justin Boesel, Keith Campbell, Michelle Harper, Mitchell Serber, Zohreh Nazeri, Eric Bloedorn, John Kuchenbrod and Ali Hadjarian of the MITRE Corporation for their input on the algorithm, the user interface and for helpful comments on the preparation of this paper.

The contents of this material reflect the views of the author and/or the Director of the Center for Advanced Aviation Systems Development. Neither the Federal Aviation Administration nor the Department of Transportation makes any warranty or guarantee, or promise, expressed or implied, concerning the content or accuracy of the views expressed herein. Approved for Public Release: 11-2622. Distribution Unlimited.

METADATA RETRIEVAL: ANNOTATION OF GEO-REFERENCED MAPS WITH SOCIAL METADATA IN SUPPORT TO UNMANNED AIRCRAFT VEHICLES MISSIONS

ROSA MEO*, ELENA ROGLIA**, AND ENRICO PONASSI*

1. Introduction to MetaData Retrieval

The natural disasters and episodes of environment pollution require the territory surveillance by the agencies of territory protection. SMAT[1] is a distributed system for the territory surveillance by means of missions performed by Unmanned Aircraft Vehicles (UAVs). A UAV is equipped with different payload sensors that will download streaming video of the target territory to the ground components of the system. The ground components are constituted by control stations that are responsible for the UAV tactical control (flight operations, sensor activities) and perform data gathering and transmission to a central station (Supervision and Coordination Station - SSC).

The SSC gives support to the generation and integration of the mission plan of each single UAV, controls the mission execution, post-processes the data from the mission and gives support to the final users from the civil protection force to elaborate the history of the data and plan the next goals. The operators can provide maps with additional annotations, metadata extracted from external sources. For these purposes the system performs data storage and near-real-time data fusion and provides both a geo-spatial and a temporal reference to the stored information.

The software functionality that we describe is MetaData Retrieval (MDR). It is one of the geo-spatial services provided by the SSC. The main focus is to provide additional information on the locations included in the cartographic maps. Cartographic maps are often thematic and do not contain all the information that is needed by any user. Furthermore, the maps need to be kept up-to-date with fresh information. On the Web there exists a large amount of information on the geographical areas generated by Volunteered Graphic Information projects (VGI) by the everyday experience of the users of the Web 2.0 applications through handhelds or mobile phones. These information on the spatial data are referred to as metadata. They are both spatially and temporally referenced and are presented by MDR in an interactive map that is useful to the mission operators for summarizing the information on the mission targets and the route way-points. Metadata can be essential for monitoring the evolution in time of the spatial objects and help in environmental emergencies when the retrieval of the past information on a certain spatial area is quickly needed. For these purposes MDR has a data warehouse that collects the history of annotations on the locations and allows to query it by a multi-dimensional, spatio-temporal query.

The information on locations is downloaded through web-services in the form of XML-based files. Web services are provided by open, collaborative projects like OpenStreetMap and GeoNames. OpenStreetMap has the aim to create a free editable map of the whole world from the contribution of the open, collaborative network. GeoNames provides the description and definition of over eight million named locations taken from projects like Wikipedia.

A Service Oriented Architecture (SOA) is the architectural choice for the system. It allows the integration of different independent systems and respond with a variety of services to different users needs. Geo-spatial Web services can be called on demand, allow an easy distribution of geo-spatial data and applications and guarantee interoperability among them.

*University of Torino, Italy, rosa.meo@di.unito.it, enrico.ponassi@educ.di.unito.it

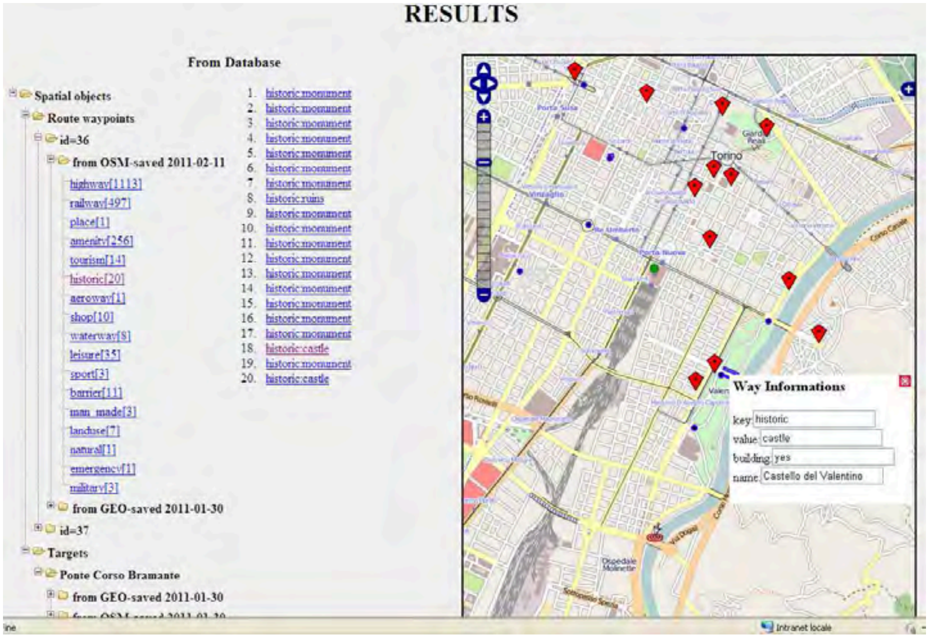
**ITHACA, Torino, Italy, elena.roglia@ithaca.polito.it.

[1] SMAT project is composed of Universities and Research centres (Univ. Torino, Politecnico Torino, ISMB), three Industries (Alenia Aeronautica, Selex Galileo and Altec) and eleven Small Medium Enterprises in Piemonte, Italy.

Any user's request to MDR searches for the metadata of some spatial objects. The MDR graphical user interface is a sort of Query By Example that allows the user to specify in a transparent way for which spatial objects the annotations are requested. The specification is not simple since missions involve a large quantity of objects and the identification of a spatial object is a multidimensional problem. For these reasons the user might act in an exploratory way and specify by means of a combination of dimensions values the spatial objects of interest. As regards the output annotations, the user specifies the type of the spatial objects and the maximum distance between the spatial objects and the nearby locations whose annotations will be displayed.

The information on locations is checked by MDR to be well-formed. In addition, MDR provides a characterization of the map in terms of the concepts corresponding to the users' annotations. MDR applies a statistical filter to the tags in order to select the annotations that are valid with a high degree of certainty. The filter is especially needed when a big number of tags is present, when some of them are the result of a user mistake with a misleading effect similar to the superimposition of noise on the valuable information. The filter compares the frequency of occurrence of each tag encountered in the given area, with the distribution of the frequencies of the same tag in the surrounding geographical areas. According to the property of spatial auto-correlation of the features, most of the tag categories are expected to occur in the neighborhood with a similar frequency. Those tags that confirm the expectations are not surprising and therefore are judged non-particularly interesting. On the contrary, the frequencies of the tag categories that are outliers of the tag frequency distribution in the surrounding areas constitute a discontinuity in space and are highlighted as the map characteristics. The map characterization that is generated by this method discriminates the map area with the neighborhood and guarantees statistical significance to the annotations and as such, an increased level of reliability.

In Figure 1 we show an interactive map, annotated by MDR. The example shows the historical monuments surrounding the target of a mission (a bridge in Torino, Italy). The left-hand side of the window displays the annotations in a tree-like arrangement that helps the user to browse the annotations for the spatial objects of the query. Annotations are ordered by location, time and category. Each annotation on the left is geo-referenced by an icon in the map on the right.



RESULTS

From Database

- Spatial objects
  - Route waypoints
    - id=36
      - from OSM-saved 2011-02-11
        - highway[1113]
        - railway[497]
        - place[1]
        - amenity[256]
        - tourism[14]
        - historic[20]
        - aeroway[1]
        - shop[10]
        - waterway[8]
        - leisure[35]
        - sport[3]
        - barrier[11]
        - man_made[3]
        - landuse[7]
        - natural[1]
        - emergency[1]
        - military[3]
      - from GEO-saved 2011-01-30
    - id=37
  - Targets
    - Ponte Corso Bramante
      - from GEO-saved 2011-01-30
      - from OSM-saved 2011-01-30

1. historic:monument
2. historic:monument
3. historic:monument
4. historic:monument
5. historic:monument
6. historic:monument
7. historic:monument
8. historic:ruins
9. historic:monument
10. historic:monument
11. historic:monument
12. historic:monument
13. historic:monument
14. historic:monument
15. historic:monument
16. historic:monument
17. historic:monument
18. historic:castle
19. historic:monument
20. historic:castle

Way Informations

key: historic
value: castle
building: yes
name: Castello del Valentino

FIGURE 1. Interactive map with annotations from the historical database

Pattern Analysis in Wind Power Time Series - Early Results

Mandoye Ndoye and Chandrika Kamath
Lawrence Livermore National Laboratory, Livermore, CA 94551

I. Introduction

Renewable resources, such as wind, are providing an increasing percentage of our energy requirements. However, integrating wind energy on the power grid is challenging for several reasons. Control room operators find it difficult to schedule wind power as it is an intermittent resource. They typically use 0-6 hour ahead forecasts, along with the actual generation in the previous hours, to determine the amount of energy to schedule for the hour ahead. These forecasts are obtained from numerical weather prediction simulations or based on estimates of wind speed in the region of the wind farms. However, the forecasts can be inaccurate, especially for ramp events where the generation suddenly increases or decreases by a large amount in a short time.

In our previous work [1], we considered the use of feature selection techniques to identify important weather conditions associated with ramp events. We wanted to identify variables that the control room operators could monitor to see if a ramp event was imminent. In the current work, we are interested in the situation where the energy forecasts are inaccurate. In such cases, the control room operators consider the energy generation for the previous few days and hours, and based on their experience and expertise, estimate the energy they should schedule for the upcoming hour.

In examining wind power time series, we have observed that there is frequently a diurnal pattern. The generation may be low and flat on days with little wind, or it may be high and flat on days when the wind speed is at a sustained high level for most of the hours in the day. Or, the speed may be high in the early hours, drop down to near zero by noon, and rise again in the late evening. It is obvious to ask if there are a limited number of these patterns for the wind generation at a site? If so, can we associate these patterns with the weather conditions for the day? If this is indeed possible, we can then provide the control room operators additional information they can use to make better informed decisions on the amount of wind energy they should schedule on the grid.

II. Methodology

To answer these questions, we analyzed the 2007-2008 wind power generation time series from a Southern California Edison (SCE) wind farm located in the Tehachapi Pass region. We refer to these data as the SCE-2007 and SCE-2008 datasets. In both years, measurements were taken at the rate of four samples per hour. Figure 1 shows the SCE-2007 dataset, with the embedded plot providing details for a two-week-long data segment shown in red in the year-long time series. In our work, we used the SCE-2007 dataset for exploratory data analysis and development of the methodology, and the SCE-2008 dataset for subsequent testing and validation.

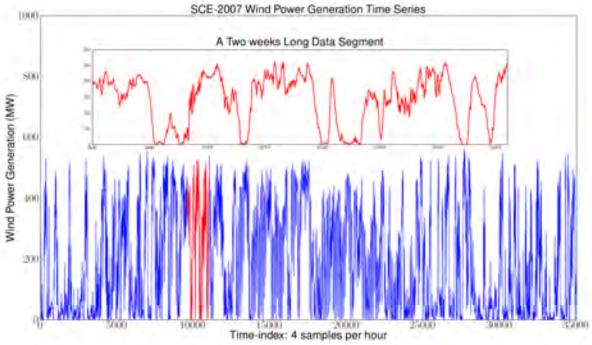


Fig. 1. The SCE-2007 data, showing details for a two-week segment in red.

Our analysis of the wind power time series indicated the presence of features at a range of scales: high frequency measurement noise, short-term signal variations, and the previously mentioned diurnal patterns. So, the time series can be represented by the N-length discrete sequence

$$Y(n) = T(n) + s(n) + w(n), \ n = 0, 1, \ldots, N$$

where $T(n)$ denotes the trend-signal containing the prospective diurnal patterns, $s(n)$ denotes the short-term variations that correspond to the small-scale features in the data, and $w(n)$ denotes high frequency noise contributions.

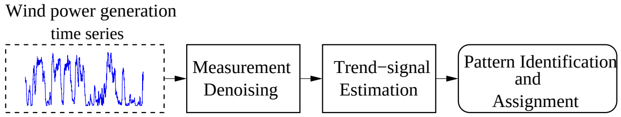


Fig. 2. Data processing steps for pattern identification and assignment.

To identify the patterns in the trend signal, $T(n)$, we use the three-step process shown in Figure 2. There are several ways in which each of these steps can be implemented. We next describe one such approach:

Measurement denoising: We expect the measurement noise to have a lower energy but higher frequency content than the actual wind power generation. To remove this noise, we first obtain a Fourier decomposition of the time series. We then reconstruct the data using the frequency components associated with the K largest Fourier coefficients. We choose

LLNL-ABS-491136: This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

the parameter K to preserve some percentage, θ, of the energy of the original time series. For our data, a value of θ in the range 95-to-99 % worked well.

Trend-signal estimation: Next, we remove the small-scale variations, which typically last from several minutes to an hour or two, using Gaussian smoothing, where the scaling parameter (i.e., the standard deviation, σ) is chosen heuristically. An alternative is to determine the intrinsic scale of the data using techniques such as scale-space theory [2] or the undecimated wavelet transform [3]. The former creates a scale-space representation of a signal by smoothing it with Gaussian kernels with increasing values of σ. The premise is that for scaling values near the intrinsic scale σ_o (to be determined) of the data, the changes in the smoothed signal are minimal. We used automatic scale selection concepts [4] to find σ_o for the de-noised time series. We then obtained the trend signal by filtering this time series with a Gaussian kernel with $\sigma = \sigma_o$. Figure 3 shows the effects of the data pre-processing steps. We note that the use of a denoising step, though unnecessary in light of the subsequent smoothing, allows us to obtain a more robust estimation of the intrinsic scale.

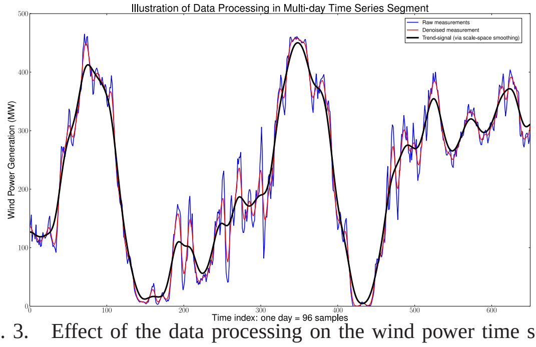


Fig. 3. Effect of the data processing on the wind power time series.

Pattern identification and assignment: Our analysis of the estimated trend signal indicated several patterns: *flat*, with approximately constant power generation, or concatenations of up to two periods of upward (*up*) and downward (*down*) generation. To identify these, we first found the start and end of the *up* and *down* generation periods using a threshold-based, peak-and-valley finder. We chose the threshold so that small changes in the threshold would lead to minimal changes in the number of peaks and valleys detected. We then assigned patterns to each day by first identifying the *flat* patterns as one where the generation was constrained to within a certain range of the minimum. Then, we assigned the remaining patterns based on the number of *up* and *down* periods contained in a day-long observation. An observation which was not assigned any of the five patterns (*flat*, *up*, *down*, *up-down* and *down-up*), was assigned to the pseudo-pattern *others*.

III. Preliminary results and discussion

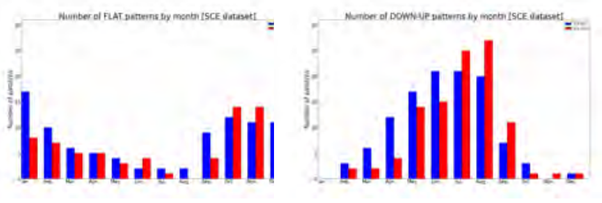
We applied the method described in Section II which was developed in the context of the 2007 dataset, to both the 2007 and the 2008 datasets. Table I shows the percentage of time each of the five patterns occurs in the years 2007-2008. The distribution is somewhat similar for the two years, though this may not always be the case. We also observe that a small dictionary of daily patterns could be used to represent the wind power generation time series at a given wind farm site.

TABLE I
Percentage of patterns in the 2007-2008 datasets

Pattern	SCE-2007	SCE-2008
flat	32.32 % (118 days)	28.14 % (103 days)
up	13.42 % (49 days)	15.57 % (57 days)
down	6.30 % (23 days)	9.83 % (36 days)
up-down	5.20 % (19 days)	5.73 % (21 days)
down-up	29.04 % (106 days)	26.50 % (97 days)
others	13.69 % (50 days)	14.20 % (52 days)

We next considered the monthly distribution for the patterns to determine if this would provide control room operators additional information for use in scheduling. We found that certain patterns occured more frequently during certain seasons, as shown in Figure 4. For example, *flat* occurs rarely in the summer, when the *up-down* pattern is more prevalent.

Fig. 4. Monthly distribution of the *flat* and *up-down* patterns for 2007-2008



(a) (b)

IV. Future work

We are currently investigating the use of feature selection and classification techniques to determine if we can use weather conditions in the region of the wind farm to predict days when a particular type of pattern is likely to occur. This work is challenging as the weather data tend to be noisy, with missing or incorrect values, and is available at sites which may be far from the wind farm. Further, as shown in Table I, some patterns occur quite frequently, while others, such as *up-down* are rare. Given this unbalanced training set, it is unlikely we will be able to obtain a high accuracy for predicting the rarer pattterns. Our early work in this area is promising; as we apply our techniques and refine the assignment of patterns, we are hopeful that we may be able to achieve reasonable prediction accuracy, at least for the more frequently occuring patterns.

References

[1] C. Kamath, Associating Weather Conditions with Ramp Events in Wind Power Generation, *2011 IEEE PES Power Systems Conference & Exposition*, Phoenix, Arizona, March 20 - 23, 2011.

[2] A. P. Witkin, Scale-space filtering, *Proc. 8th Int. Joint Conf. Art. Intell.*, pages 1019-1022, Karlruhe, Germany, 1983.

[3] S. Mallat, A Wavelet Tour of Signal Processing, Academic Press, San Diego , 1999.

[4] T. Lindeberg, Principles for automatic scale selection, *Handbook of Computer Vision and Applications, Volume 2*, pages 239-244, Academic Press, Boston, USA, 1999.

GLIDER: Satellite Data Mining Made Easy

Rahul Ramachandran, Sara Graves, Todd Berendes, Manil Maskey
Information Technology and Systems Center
University of Alabama Huntsville

The large volume of publicly available satellite imagery has the potential to provide a wealth of information for both civilian and military decision makers. Often, sophisticated data mining analyses such as land-use and land cover change detection are implemented to provide "actionable intelligence" at regional and global scales. Mining satellite data requires tools that can handle spatial, temporal and spectral analyses. Complex analyses such as thematic image classification are often performed manually and can be a cumbersome, step-by-step process requiring extensive user interaction.

UAHuntsville has developed a freely available tool that simplifies mining of satellite imagery. The Globally Leveraged Integrated Data Explorer for Research (GLIDER) provides an integrated plug-in based software workbench with a set of visualization and analysis tools that facilitate sophisticated analysis of satellite imagery. Visualization modes such as three band color composite and look-up table color display allow easy interactive image exploration and aid in identification of image features. Imagery can be displayed in a 2-D native swath view or overlaid on a 3-D globe display. Pixel level data can be plotted using scatter plots, histograms, spectral profiles and spatial transect profiles. Additionally, pixel level data can be interactively sampled and extracted from the imagery and used to train supervised learning classifiers for use in land-use studies and other analyses. An entire suite of data mining algorithms is integrated within GLIDER and a workflow composition tool is provided. This poster showcases some of GLIDER's powerful features and provides some case studies showing practical applications of data mining for land-use, cloud and aerosol detection and decision support.

SPIKE DETECTION IN FLIGHT QUICK ACCESS RECORDER EVENT RATES USING CONTROL CHARTS BASED ON THE BETA DISTRIBUTION

ANIL YELUNDUR* AND KEITH CAMPBELL**

1. Introduction

Proportions control charts based on the Beta distribution have produced positive feedback from operational experts when used to detect spikes in safety-related flight event rates derived from Flight Quick Access Recorders, which record operational parameters on board an aircraft.

Monitoring of safety-related events to identify emerging aviation safety concerns is one goal of the Aviation Safety Information and Sharing (ASIAS) program, a collaborative government-aviation industry program. In this context, a 'spike' refers to a sudden, relatively large increase in event rates.

Because ASIAS stakeholders wish to monitor many hundreds of time series, for example particular airports and aircraft types, automated screening is essential. Spikes are reviewed by teams of operational experts with limited time, so screening needs to produce a small number of alerts that are likely to be actionable.

2. Control Charts a logical Monitoring Tool

As tools designed for detecting changes in a process, Control Charts are a natural tool for spike detection. ASIAS flight data is organized by month over a three-year period. Reviews occur quarterly, and the review teams prefer to view time series aggregated by quarter. Accordingly, the control charts use 3-period groupings from months to quarters.

We are interested in detecting sharp changes in the quarterly rate such that they are much higher than historic trends - we want to detect upward spikes. Monitoring focuses on identifying spikes in the most recent two quarters, with prior periods being treated as the training period.

Because events are measured in terms of rates, e.g. ground proximity warnings per 100,000 flights, Proportions Control Charts [1] were a logical initial choice. Unfortunately, Proportions Charts (assumes a Binomial distribution i.e. a constant event rate) performed poorly with our data. The reason being that the observed event rates are over-dispersed relative to the Binomial distribution hence resulting in the generation of an excessive number of alerts.

3. Control Chart using Beta Distribution

The Beta distribution models the uncertainty in the proportion p of a Binomial distribution by taking into account the sample mean and variance. Hence fitting a Beta distribution to the data addresses the issue of over-dispersion relative to a Binomial distribution.
A Beta distribution is defined by two positive shape parameters : α and β.
The probability density function for the proportion p given the two shape parameters is given by:

$$P(p|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1}$$

*MITRE/CAASD, ayelundur@mitre.org
**MITRE/CAASD, keithc@mitre.org.

3.1. **Parameter Estimation.** The two shape parameters i.e. α and β are estimated from the data corresponding to the training period. The training period consists of adjusted quarterly rates over the entire time period except the latest two quarters. We are using the Maximum Likelihood Estimation Method (MLE) to estimate the two shape parameters with initial values set to those obtained via Method of Moments (MOM).

3.2. **Control Limits.** Once α and β are determined, the control chart parameters, namely the mean (i.e. baseline) and variance, are calculated using the corresponding formulas for the Beta distribution.
For $\alpha > 1$ and $\beta > 1$, the Upper Control Limit (UCL) is calculated using the 99.87th percentile of the Beta distribution i.e. 3 Standard Deviations (SD) away from the mean of a Normal distribution:

$$n_{sd} = \frac{(Q_{Beta} - baseline)}{\sqrt{variance}}$$

$$UCL = baseline + n_{sd}\sqrt{variance}$$

where Q_{Beta} denotes the 99.87th percentile of the corresponding Beta distribution.
We select the MLE estimates only when $\alpha > 1$ and $\beta > 1$ because the Beta distribution is unimodal i.e. the minimization algorithm is guaranteed to have converged to a global minimum. Else we revert back to the MOM estimates and set n_{sd} is set equal to 3. Also note that the Beta distribution is skewed when $\alpha \neq \beta$ i.e. it is not symmetric about the mean (right skewed for our data since $\beta > \alpha$). This makes it necessary to specifically estimate the parameter n_{sd}. If α happens to be ≤ 1 then we set n_{sd} equal to 3. If the minimization algorithm does not converge, then the MOM estimates are used for α and β and n_{sd} is set equal to 3.

3.3. **Example Chart.** Figure 1 illustrates an example series that produced an alert. The last two quarters for the subject airport has event rates that lie between 2 SD and 3 SD relative to the baseline. Applying the Western Electric rules, two consecutive time series samples that lie between 2 SD and 3 SD from the baseline indicate an alarm i.e. a violation (represented in the figure as a red circle).

4. Conclusions

4.1. **Implementation Experience is Largely Positive.** Spike detection using the Beta method has been applied to the ASIAS quarterly review process since early 2011. The Beta charts are being used to monitor system-wide and airport-specific rates for three metrics. About two hundred airports are being tracked, so about 600 series in total are being monitored.

Reaction from operational experts has been largely positive. The initial round of reports produced twelve alerts, few enough for the review committee to briefly examine each. Alerts consistently indicated clear spikes, and were considered relevant by domain experts. One alert was selected for in-depth review. That rate increase was clearly associated with a change in operating procedures at a major airline, increasing the level of confidence in the alerts.

4.2. **Rare Events Issue and Alternate Methodologies.** The Beta method is limited in its ability to cope with rare events. For example, a fourth metric that is part of the quarterly process is a candidate for monitoring but has a large proportion of airports with zero or one events during the three-year period, too few to construct meaningful control limits with our current approach.

We are investigating alternate methods that can cope with rare events. Currently Parametric Empirical Bayes (PEB) method appear to be most promising. We have developed two PEB models namely, a Poisson-Gamma model and a Beta-Binomial model. Initial results suggest that both models provide improved performance compared to the Beta charts when events are rare and equivalent performance for non-rare events.

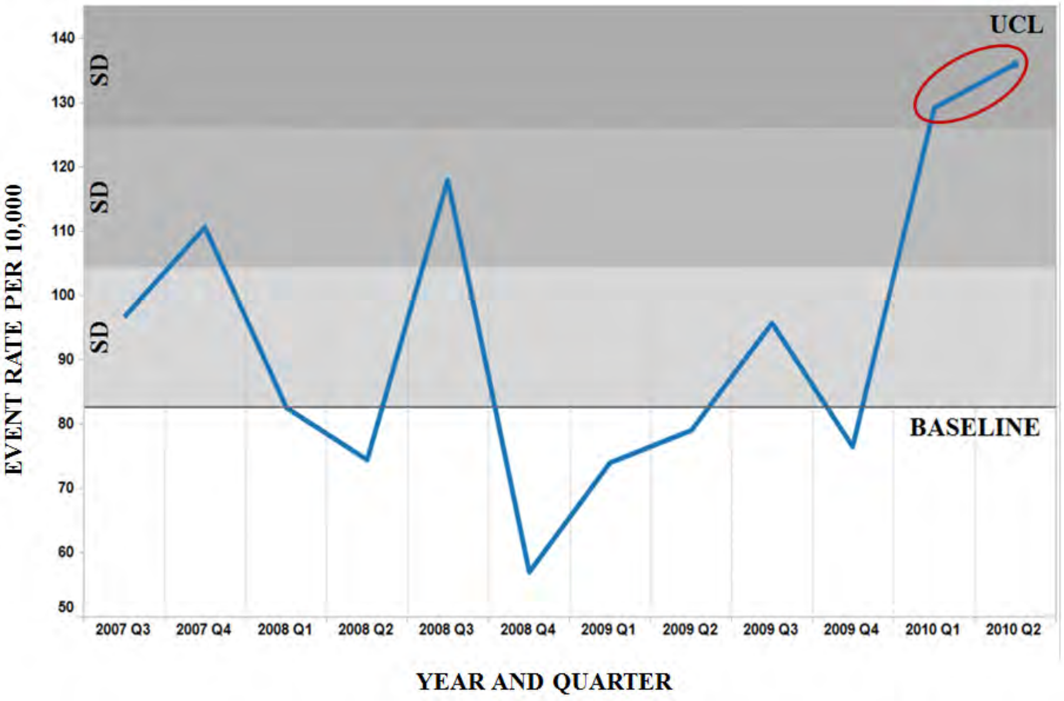


Figure 1. Spike Detection (data for major US airport between Q3 2007 and Q2 2010) using Beta Distribution.

5. Acknowledgement

The authors would like to thank Clay Shwery and Jessica Rothbart for testing the spike detection methodologies on various safety metrics. We appreciate the guidance received from Justin Boesel, Ed Walsh and Paul Melby, which helped focus this effort. Finally we appreciate the support of the Federal Aviation Administration in sponsoring this work.

The contents of this material reflect the views of the author and/or the Director of the Center for Advanced Aviation Systems Development. Neither the Federal Aviation Administration nor the Department of Transportation makes any warranty or guarantee, or promise, expressed or implied, concerning the content or accuracy of the views expressed herein.

Approved for Public Release: 11-2480. Distribution Unlimited.

References

[1] *e-Handbook of Statistical Methods.* NIST/SEMATECH, Gaithersburg, MD, 2006.