Uncovering Underlying Features

for State Transition Modeling

By

ASHKAN ALIABADI FARAHANI[1]

Presented to the faculty of the graduate school of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements for the degree of

DOCTOR OF PHILOSOPHY

UNIVERSITY OF TEXAS AT ARLINGTON

AUGUST 2019

Supervising Committee:

Victoria C.P. Chen, Supervising Professor

Doyle L Hawkins

Jay Rosenberger

Shouyi Wang

[1] Department of Industrial, Manufacturing and Systems Engineering. The University of Texas at Arlington. Arlington, TX 76019 USA ashkan.aliabadifarahani@mavs.uta.edu

Copyright © by Ashkan Aliabadi Farahani 2019

All Rights Reserved

Acknowledgments

This material is based upon work partially supported by the National Science Foundation under Grant CMII-1434401.

Abstract

Uncovering Underlying Features for State Transition Modeling

Ashkan Aliabadi Farahani

The University of Texas at Arlington,

2019

"All effort to bring order into disorder is disorder."

David Bohm

Supervisor Professor: Victoria C.P. Chen

Modeling of a dynamic system is the representation of the interconnectivity of system state variables and their evolutionary trajectory over time. In this dissertation, the terminology "state transition modeling" refers to a situation when the system state transitions and its evolution is unknown and needs to be estimated. There are situations in many application settings where one does not simply observe the behavior of the system, but also has a desire to take action, intervene, and manipulate one or more system variables, and is interested in seeing the causal effect of the intervention. These interventions within a purely observational setting, as opposed to a randomized controlled trial, have induced a controversial debate between statistics and econometrics. This dissertation presents approaches that seek to uncover the true underlying features for state transition modeling given multiple decisions and set of covariates in a complex situation: finite horizon with a very few stages, non-stationary nature of transitions, and a highly correlated feature space.

First, existing state space modeling methodologies are reviewed, and then the proposed methodology is presented in two settings. First, a simplified observational setting is studied with no interventions (i.e., no treatment variables), and then, second, a more complex observational setting is studied with multiple treatment variables under time-varying confounding and multicollinearity.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere gratitude to my advisor, Dr. Victoria Chen, for her patience and continuous support, her inspiration, and compassion in sharing her valuable knowledge and expertise during the years of my Ph.D. study. Her unique method of teaching, sharp knowledge in the field, combined with a caring nature, makes her a role model in professional academic environment. I could not have imagined having a better advisor and mentor for this chapter of my life.

Besides my advisor, I would like to thank the rest of my dissertation committee; namely: Dr. Doyle L Hawkins, for the continuous support through all these years, Dr. Jay Michael Rosenberger, and Dr. Shouyi Wang, for their constructive comments and encouragement, and also for the questions which motivated me to widen the perspective of my research.

I am also grateful for the help of my friends and close colleagues, Nilabh Ohol, Amith Viswanatha, and Shirish Rao, who helped me extensively throughout my dissertation.

Last but not least, I would like to thank my beloved parents, who filled me with indefinite love and support throughout my years at the University of Texas at Arlington. I could not have achieved this without their support.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

Introduction

Modeling systems with temporal setting, where we are interested in "reasoning" about the state of the system as it evolves over time, has applications in a wide variety of fields, such as mathematics, physics, biology, chemistry, engineering, economics, and medicine. The evolution rule of the dynamical system is a function that describes what future states follow from the current state (i.e., state transition functions). We can model such settings in terms of a system state, whose value at time "*t*" is a snapshot system state of the relevant attributes (hidden or observed) of the system at time "*t*." [1] In another words, states of dynamical systems are those critical variables that provide a complete representation of the internal condition or status of the evolution of the system at a "given time."

These transition functions could either be known a priori or unknown. For instance, a retail inventory system tracks the number of items in stock for each product type. The state is the current number of items in stock, and the transition to the next time period would simply subtract the number of items that were sold to customers and add the number of items that were ordered from the manufacturer. This formula that enables calculation of the next state given the current state is the state transition function.

Now imagine, we face a new system with no knowledge about its state attributes or about its state transition evolution. All we know is based on information and data gathered from the domain expert, including information on potential state and treatment variables. Then we begin our study in a pure observational setting, and at each stage or time slice, record the value of the candidate states. There are also treatments or interventions in the data that are dynamically occurring and not controlled as in a randomized clinical trial. For our studies, data are assumed to be collected over a finite horizon with very few discrete stages, and it is allowable for the state transitions to be nonstationary. With these data, it is desired to understand the evolution of the system and how interventions affected the system. In other words, we would like to model the unknown state transitions of the system from one stage to another, not only evolving from natural evolution but also

due to the interventions. One of the applications studied is that of an adaptive interdisciplinary pain management program. It is desired to understand the effects of a set of treatments on a patient's chronic pain metrics that are observed at three points in time, pre-treatment, mid-treatment, and post-treatment [3].

Without having the state transition function, and thereby the evolution of the system dynamics, any further analysis will be impossible, let al.one optimization of the system, for instance finding the most effective set of treatments in pain management study. Hence, this dissertation provides a step towards uncovering true features needed to properly represent the state transition in a dynamic simulation or optimization.

1.1 Definition of State Transition Modeling

Here we describe the notation and setting of the problem in more detail.

$$Y_{t+1} = f(Y_{1:t}) + f(U_{1:t}) + f(X_t) + \epsilon_t \quad (1)$$

Y_{t+1} = state of interest at time $t+1$

$Y_{1:t}$ = state of interest up to time t

X_t = set of Exploratory variables at time t

$U_{1:t}$ = decision or control variable up to time t

ϵ_t = error at time t

Equation (1) is the general state transition model representation for our study. More complex forms are a challenge for future work. In particular, equation (1) shows dependency of the state on previous states, on possibly previous treatments/interventions/actions, and on a set of covariates that are state variables. Now, depending on the system setting and other limitations, this model form encompasses a wide variety of models in statistics and state space literature. Hence, we clarify the purpose of our study, and specify the system of interest, while also pointing out some limitations.

1.2 Definition of the Problem: Purpose, characteristics, and limitations

1. Unknown State Transition Model

As the first and foremost aim of this study, this task conceptually can be separated into two subproblems. First, a common problem in the statistics and computer science literature is prediction. For this subproblem, we seek to predict future states from the past history and to identify the precision of this prediction. Second, a causal modeling problem seeks to uncover the true underlying state transition model of the system. Part of this subproblem is to identify, among the candidate state variables and control variables, the true features that contribute to evolving the state in the next stage. This problem of discovering the true underlying model allows us better represent reality and utilize the causal evolution to enable better interventions for controlling the system. While the literature focuses more on the former, our focus in this dissertation is on the latter.

2. Finite Horizon

This study only focuses on dynamic systems with a limited number of stages. Our first case study has only **three stages** similar to figure 1.

3. Linearity

We assume that function *f* in equation (1) is linear or can be approximated linearly.

4. Non-stationary

The state transition of one stage differs from state transition of another stage. That is natural when we only have a few stages. Stationarity often exists or is assumed with infinite horizon problems or those with a large enough number of stages [4, 5].

5. Discrete or continuous

In general, the state space of the system can be a mixture of continuous and discrete variables. However, in this study, the response or outcome variables is assumed to be continuous.

6. Offline or batch analysis

The proposed approach conducts an offline or batch analysis, for which all the data has been collected and is analyzed in one batch.

7. Intervention and causal effect

In addition to observing the behavior of the system, a key element in causal modeling is understanding the effect of the treatment or intervention, i.e., the efficient estimation of the causal effect of the intervention. In another words, we have decision or control variables and would like to see rather the effect they have on system behavior, subject to several other variables that describe the state space of the system.

8. Pure Observational Setting

We emphasize again that our methodology is developed in a pure observational setting. There is a debate between statistics and econometrics in the last decade to address "causal reasoning" in an observational setting [6, 7, 8, 9, 10]. When we model the unknown state transition of a system, we would like to get as close as possible to the true underlying relationship among variables that are contributing to the change in system behavior over time. *We do not claim that our methodology is capable of causal inference in an observational setting, but rather exploring and inventing ways to take a step closer to that concept regardless of the ongoing debate between statistics and econometrics* [11, 12].

9. Highly correlated state space

Apart from the issue of time-varying confounding that exists in the pain management case study and which is handled with weighting methods developed previously by Ohol et al. [3, 13], there exist high correlations between state variables (covariates) and controls (treatment). Detecting and extracting important interpretable features in this setting is a challenge. We motivate our simulated case studies induced by this issue, and we provide more details in Chapter 3.

10. Parsimony on control variables

The result of the state transition modeling in this study may be used to optimize the system. The optimization task is to dynamically adjust the controls in each stage so to yield the desired system behavior. One of the challenges is that the control variables (i.e., *U* in state-space models [14]) are affected by state variables. When uncovering the true features underlying the unknown state transitions there could be many potential state variables that are unimportant features; hence, parsimony is desired. While fewer treatment control variables are anticipated to be unimportant, parsimony is also sought among the treatments.

11. Order unknown

The order of dependency in this study, like many other specifications of this system, is unknown. Thus, the matter of the order exploration of the system depends on the availability of data for not just one, but several runs. We defer this discussion to the end of the literature review.

1.3 Some Comparative Terminology between Statistics and Computer Science

In our literature review, because we cover a broad class of models from both the statistics and computer science literature, we provide the reader with a translation of equivalent terminology. The usage of the term **learning** and **inference** depends upon the field of study. Confusion usually arises when the words are used casually without reference to a particular field. The process of observing data and learning from it is an intuitive definition of inference.

When **statisticians** talk about inference, they usually talk about statistical inference. In statistical inference, we observe some data, and we would like to build knowledge about the process that generated these data. Hence, predictions, estimating error bars, hypothesis testing, and parameter estimation would all be part of statistical inference. Notice how parameter estimation is also included under statistical inference.

On the other hand, traditional machine learning researchers from a computer science background often like to make a distinction between learning and inference. Learning is associated with parameter estimation and is not explicitly thought of as an inference problem. Hence, the conceptualization of the term "inference" is narrower than that of a statistician.

Table 1. Equivalent Terminology between Statistics and Computer Science

Computer Scientist	**Statistician**	**Meaning**
Inference e.g., sampling algorithm, Kalman forward and backward algorithm	**Statistical Inference** Predictions, Confidence Intervals, hypothesis testing	Use the estimated model or learned Parameters to predict some value
Parameter Learning e.g., Gradient decent algorithm	**Parameter Estimation** Least squares	Using data to estimate an Unknown parameter
Structure Learning	**Model Selection**	Select the best model that predicts well or represents the features closest to that of the true model.

Commonly, inference is thought of as making some sort of prediction. For example, in linear regression, given some features and some learned parameters, we may want to predict some real-valued variable. Or, in an image processing problem, given an image with many missing pixel values, we may want to fill in the most probable values for the missing pixels from our learned joint distribution. Both of these predictions would be called inferences. An advantage of making a distinction between learning and inference is that it naturally separates learning algorithms from inference algorithms. Although for some problems, parameters can be estimated analytically, most problems require a learning algorithm, such as a gradient descent type algorithm. Similarly, in some inference problems, such as the image processing example above, the prediction is usually not a closed-form formula and requires one to use an inference algorithm, such as a sampling algorithm, to compute the prediction. Things become even more interesting in models with latent variables,

where often times an inference algorithm is nested within a learning algorithm, as seen in Markov chain Monte Carlo (MCMC) Expectation-Maximization (EM) algorithms.

For more clarification, estimation and learning are basically the same, but the flavor of terms is a little different. Estimation usually means that one specifies underlying distributions and then estimates their parameters. Learning may be distribution free, perhaps solely optimizing some target function, and it applies in situations with complex structured data when it not reasonable or possible to build a distributional model. To summarize, the difference between inference and learning depends on the eye of the modeler. If you think like a statistician, then learning/parameter estimation is a type of inference. If you think like a traditional machine learning researcher, then learning is usually parameter estimation and inference is usually prediction. Different perspectives are useful in different situations.

The remainder of this research is organized as follows. Chapter 2 provides a literature review on common modeling techniques that handle state transitions in a temporal setting. These models are state-space models (SSM), Kalman filter models (KFM), Hidden Markov Models (HMMs), Dynamic Bayesian Networks (DBNs), and longitudinal data analysis. Then we discuss the shortcomings of these methods in addressing our application case, and then provide our motivation to fill this gap. Chapter 3 presents and motivates our approach based on penalized regression and demonstrates in performance for additive underlying models that have varying levels of correlation structure. Chapter 4 integrates the approach in Chapter 3 with that of Ohol's [13] to enable handling of time-varying confounding in adaptive treatment regimes. Finally, Chapter 5 presents conclusions and future work.

Chapter 2

Literature Review

The data generated by a dynamical system is either sequential data or time series data or simply put, repeated measures of the same variable over time. Methods found in the literature to handle these kinds of settings are classical approaches, such as time series prediction using linear models (ARIMA, ARMAX, etc., see [5]) and longitudinal data analysis for repeated measures [15, 16]. Or methods, such as State Space Models (SSM), and its different representations of Kalman Filters (KFMs), Hidden Markov Models (HMMs) and more of a general representation, Dynamic Bayesian Networks(DBNs) [14, 1].

One thing worth mentioning here is that the family of SSMs is originally designed for infinite sequences or sufficiently long finite sequences, so that one can assume stationarity of the state transitions. The inference and learning algorithms in this field rely heavily on this set of long sequences. Consequently, in spite of the fact that these methods structurally seem appealing, they are not appropriate for cases with very few stages or nonstationary transitions. We first give a review of SSMs and their variations and then make an effort to redefine our problem in the body of state-space models for the purpose of an interactive review.

2.1 State-space Models

The terminology state-space models is often times translated as KFMS. Murphy [14] introduces the terminology for a broader range of models as KFM, HMM, and DBN since they are basically trying to accomplish the same thing in a different setting. State-space models are better than classical time-series modeling approaches in many respects [17]. In particular, they overcome all of the problems in time series [18, 19, 20, 21, 22]: they do not suffer from finite-window effects that force us to base our prediction of the future on only a finite window into the past; they can easily handle discrete and multivariate inputs and outputs; and they can easily incorporate prior knowledge. For instance, there could be variables that we cannot measure, but whose state we would like to estimate; such variables are called hidden or latent. Including these variables allows us to create models which may be much closer to the “true” causal structure of the domain

we are modeling [7]. Even if we are only interested in observable variables, introducing "fictitious" hidden variables often results in a much simpler model. For example, the apparent complexity of an observed signal may be more simply explained by imagining it is a result of two simple processes, the "true" underlying state, which may evolve deterministically, and our measurement of the state, which is often noisy. We can then "explain away" unexpected outliers in the observations in terms of a faulty sensor, as opposed to strange fluctuations in "reality."

In the following subsections, we discuss representation of state-space models in general terms, how to use them to update the belief state and perform other related inference problems, and how to learn and build such models from data. We then discuss the two most common kinds of state-space models, namely Hidden Markov Models (HMMs) and Kalman Filter Models (KFMs). In Murphy [14] the representation, inference, and learning of more general state-space models, called Dynamic Bayesian Networks (DBNs) are discussed [14]. Any state-space model must define

1- Prior or initial state, $P(S_1)$. (2)

2- State transition function, $P(S_t|S_{t-1})$. (3)

3- Observation function $P(O_t|S_t)$. (4)

Sometimes we have some control over the system we are monitoring. In this case, we would like to predict future outcomes as a function of our intervention (policy /decision). In the controlled case, equations (3) and (4) become respectively, $P(S_t|S_{t-1}, U_{t-1})$ and $P(O_t|S_t, U_t)$; we allow the observation to depend on the control so that we can model change with intervention, known as active perception [23].

In a state-space model, we assume that there is some underlying hidden state of the world (S) that generates the observations (O), and that this hidden state evolves in time, possibly as a function of our inputs (U). A state-space model is a model of how S_t generates or "causes" O_t and S_{t+1}. The goal of inference is to invert this mapping, i.e., to infer $S_{1:t}$ given $O_{1:t}$, i.e., $P(S_t|o_{1:t}, u_{1:t})$. The concept of hidden is a key concept in SSMs, and if misunderstood, then SSMs cannot be of help to a researcher. We defer the practical

discussion of this to 2.4 for full observability and the appendix for partial observability, where we define different versions of our problem in the body of SSMs.

2.1.1 Fundamental state-space models assumptions

1. **Time slice**. The first simplification is to discretize the timeline into a set of time slices that are measurements of system states at intervals that are regularly spaced with predetermined time granularity Δ. This assumption simplifies our problem from representing distributions over a continuum of random variables to representing distributions over countably many random variables, sampled at discrete intervals.
2. **Markov assumption** on the hidden state (S) and the observation (Y). One very natural approach is to assume that the future is conditionally independent of the past given the present.

$$P(S_t|S_{1:t-1}) = P(S|S_{t-1}) \qquad (S_t \perp S_{1:t-1} \mid S_{t-1}) \tag{5}$$

$$P(O_t|O_{1:t-1}, S_t) = P(O_t|O_{t-1}, S_t) \qquad (O_t \perp O_{1:t-1} \mid O_{t-1}) \tag{6}$$

   We need only consider whether the Markov assumption is a sufficiently reasonable approximation to the dependencies in our distribution. *In most cases, if we use a reasonably rich state description, the approximation is quite reasonable* [1].
3. **Stationarity, also called time-invariant or homogenous.** If we assume that the transition functions, equation (2), are the same over time, then the model is said to be time-invariant or homogeneous. The reason behind this assumption is in case of infinite-horizon the process can continue indefinitely, then equation (3) still leaves us with the task of acquiring an infinite set of conditional distributions, or a very large one, in the case of finite-horizon processes. Therefore, we usually make one last simplifying assumption. *Without this assumption, we could not model infinitely long sequences.* A remedy to parameter change over time is to just add them to the state space, and treat them as additional random variables [1].

There are many ways of representing state-space models (SSMs), the most common are Hidden Markov Models (HMMs) for discrete state spaces, Kalman Filter Models (KFMs) for continuous state spaces, and switching state space for a mixture of discrete-continuous state spaces. Finally, all of the state-space models can be generalized as DBNs, so that HMMs and KFMS are just special cases [14]. A brief representation of KFMs, HMMs, and DBNs is in the Appendix.

2.2 Inference and Learning of State-Space Models

Previously, we summarized the representation of state-space models, and it seems that our problem of interest, namely state transition modeling lies somewhere in equation (3), the state transition model of a state-space model. Notice that in many early applications of SSMs, like tracking an object, we are studying a physical system with a scientifically known transition from physics law, which means that we know the transition models and its parameters upfront from apriori facts. In facing the system with unknown structure and parameters, the EM algorithm [21, 24] and its extensions are used. In machine learning and computer science estimating the hidden state given the observations and the model (i.e., inference), and learning the model parameters is considered as two separate problems of respectively, Inference and Learning or *system identification* in the engineering literature [25].

There are mainly two cases in state-space models. In the first case, we know what hidden states cause the observations and just want to estimate them. These are examples that come first in any state-space models tutorial, such as the aforementioned tracking problems in linear dynamic systems (KFMs). Often times we can have the initial state, state transition matrices, and the observation function based on the knowledge of the problem structure or physics. The aim then is to obtain **accurate or exact inferences** of the unobserved information from the data, which is also known as **state estimation**.

The main kind of **inferences** that we might want to perform using state-space models are shown in Figure 1 [26, 27].

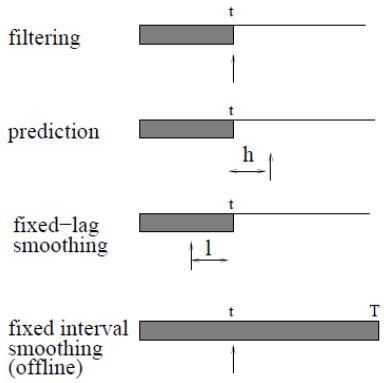


Figure 1. The main kinds of inference for state-space models. The shaded region is the interval for which we have data. The arrow represents the time step at which we want to perform inference. t is the current time, and T is the sequence length.

Filtering: The goal of filtering is computing the probability of the current hidden state X_t given the sequence of inputs and outputs up to time *t*. $P(S_t|Y_{1:t}\ ,\ U_{1:t})$.

Smoothing: The goal of smoothing is to compute the probability of X_t given the sequence of inputs and outputs up to time *T*, where *T* > *t*. $P(S_t|Y_{1:T}\ ,\ U_{1:T})$.

Prediction: Finally, the goal of prediction is to compute the probability of future states and observations given observations up to time *t*.

The second case, which is our case**,** is to explore possible explanation or causes for the observed data without having any explicit model for what should be the state transition evolution and observation function.

Parameter and structure learning: Discovering the true underlying model: In this case, the observation and state transitions are mostly or entirely unknown. The goal instead, is on **learning** the *structure* and a few *parameters* that model the observed data well (with high likelihood). In other words, for instance for the KFMs given only an observed sequence of the outputs (or perhaps several sequences), find the parameters of the SSM $\{A, C, Q, R, \mu_1, Q_1\}$ which maximize the likelihood of the observed data. This is where the concept of discovering the true underlying model is motivated since there is absolutely no prior knowledge about the true underlying transitions of the system. In general, in the sense of a directed acyclic graph (DAG), structure

learning refers to learning the graph topology (arcs between nodes or dependency structure) no matter what parameterization is used. This is often called model selection in statistics.

In state-space models, algorithms for solving parameter and structure learning depend mainly on three factors, fully observed or partially observed, Frequentist or Bayesian estimation, offline or online learning.

Fully observed or partially observed. Partially observed refers to the case where the values of some of the nodes in some of the cases are unknown. This may be because some data are missing, or because some nodes are latent/hidden. Learning in the partially observed case is much harder; the likelihood surface is multimodal, so one usually has to settle for a locally optimal solution, obtained using EM or gradient methods.

Frequentist or Bayesian. A frequentist tries to learn a single best parameter/model. In the case of parameters, this can either be the maximum likelihood (ML) or the maximum a posteriori (MAP) estimate. In the case of structure, it must be a MAP estimate, since the ML estimate would be the fully connected graph. By contrast, a Bayesian tries to learn a distribution over parameters/models. This gives one some idea of confidence in one's estimate and allows for predictive techniques, such as Bayesian model averaging. Although more elegant, Bayesian solutions are usually more computationally expensive to obtain.

Offline or online learning. Offline learning refers to estimating the parameters/structure given a fixed batch of data which is the focus of the study here. Online learning refers to sequentially updating an estimate of the parameters/structure as each data point arrives. (Bayesian methods are naturally suited to online learning.) We emphasize again that our study here is *learning parameters and structure of the model that causes the observations.* In a DAG of our problem in figure 1, if we assume that intra-slice connectivity is fixed, then the goal of transition structure learning is to learn inter-slice connectivity arcs, choose the parents of nodes in time slice t from time slice t - 1, which is equivalent to the variable or feature selection problem.

2.2.1 Inference and learning of linear dynamic systems or KFMs

Inference: The recursive algorithm to compute the three types of inference probabilities of filtering, smoothing, and prediction (with **known parameters)** is known as Kalman Filtering [28]. Kalman Smoothing

recursions or Rauch-Tung-Streibel (RTS) smoother (combined forward and backward recursions for smoothing) [29]. Filtering and smoothing over continuous states have been extensively studied in the work of Kalman [30] and Rauch [29], but this literature is not very well known to the machine learning community. To understand more about filtering and smoothing problem of LDS please refer to [25, 31].

Learning: The learning of the parameters of dynamic linear systems is known as *system identification* in engineering terminology. The idea of *online learning* is desired in real-time adaptive control situations and is obtained by gradient algorithms [32]. Other than gradient-based methods, the EM algorithm is mostly used for offline learning with unknown parameters [33, 34, 35, 36, 24].

Caveats: KFMs assume the system is jointly Gaussian (observation and state). This means the belief state must be unimodal, which is inappropriate for many problems, especially those involving qualitative (discrete) variables. For example, some systems have multiple modes or regimes of behavior; in these cases, KFMs cannot capture multi-modal patterns. This is the idea behind the switching KFMs that we discuss in 2.2.3.

2.2.2 Inference and learning of HMMs (discrete states)

Inference: There are two algorithms commonly used to solve two kinds of inference problems [37]. First, same as KFMs, is finding the posterior probability of the hidden states give the observations using a recursive algorithm known as the *forward-backward algorithm* that is analogous to Kalman filtering and smoothing. This algorithm is a special case of exact algorithms for probabilistic graphical models [14, 38, 6]. The second kind of inference algorithm is finding the single most likely sequence of the hidden states. The solution is a Viterbi Algorithm, which also consists of a forward and backward path in the model [39, 14].

Learning: For learning the maximum likelihood of Parameters of HMMs given the sequence of observation, the Baum-Welch algorithm [40], which is a special case of EM algorithm, is often used.

To understand about different augmented models of HMMS and comprehensive review of their inference and learning algorithms, please read [14, 1].

Caveats: Let's say we have *K* possible states, in a problem with *N* objects, ends up with K^N possible values for the state space. Since we must form a Cartesian product of the state space of each individual object, to specify the transition and observation model, we need a lot of data to learn the model (high sample complexity) and exponential time $O(Tk^{2N})$, for forward-backward inference algorithms (computational complexity). DBNs help to mitigate both of the problems.

2.2.3 Inference and learning of switching state-space models or hybrids

In many applications, the state space is the mixture of continuous and discrete variables, so that neither pure KFM nor pure HMM state transitions can appropriately capture the temporal structure of the data. For recent developments and representations in switching state-space models, please refer to [41, 42]. This family of models, also known as Hybrid Models, Jump-linear system, jump-Markov model and Switching KFMs [14], opened a new door for handling not only mixed states of continuous and discrete, but using the "switch" concept to model piece-wise linear or nonlinear state-space models [43], non-stationary state transition as a special case of switching state-space models [35], and non-Gaussian noise models. The term "jump" or "switch" implies that the state is discrete, whereas linear allows the possibility that the state is continuous. Note that representations of this kind of model can differ, based on the concept of the "switch" parameter in the application of interest, and one should probably develop a problem-specific switching state space rather than conventional developments in this field. For some examples of applications, please refer to [41] and [14].

Inference and learning: Unfortunately the exact inference algorithms in switching KFMs is intractable, therefore approximate inference methods are used. There are two approximate inference approaches: deterministic approximation and stochastic approximation. For details refer to [14].

2.3. Longitudinal Data Analysis

We now change our perspective and focus to the statistics literature and give a brief review of the family of statistical models known as longitudinal data analysis that is designed to handle the mechanism of the change of time or effect of covariates of research unit or subjects. In each of these subjects, the data are collected repeatedly over time. Therefore, the key feature of longitudinal data is that each individual's measurements are correlated to each other.

2.3.1. Mathematical notation

Suppose y_{ij} are the *j*th of n_i measurements on the *i*th of *n* subjects. Let

$$y_i = (y_{i1}, y_{i2}, \dots, y_{in_i})^T \text{ , } i = 1,2, \dots, n \ \ where \ \sum_i n_i = N \tag{7}$$

be the observed response vector of subject *i* and

$$t_i = (t_{i1}, t_{i2}, \dots, t_{in_i})^T \tag{8}$$

be the subject observation time points for subject *i.* Let

$$\Sigma_i = \begin{bmatrix} \mathrm{Var}(y_{i1}) & \mathrm{Cov}(y_{i1}, y_{i2}) & \cdots & \mathrm{Cov}(y_{i1}, y_{in_i}) \\ \mathrm{Cov}(y_{i2}, y_{i1}) & \mathrm{Var}(y_{i2}) & \dots & \mathrm{Cov}(y_{i2}, y_{in_i}) \\ \vdots & \dots & \ddots & \vdots \\ \mathrm{Cov}(y_{n_i 1}, y_{i1}) & \mathrm{Cov}(y_{in_i}, y_{i2}) & \cdots & \mathrm{Var}(y_{in_i}) \end{bmatrix} \tag{9}$$

be a *positive definite matrix*, the covariance matrix of y_i, and $\mu_i = (\mu_{i1}, \dots, \mu_{in_i})^T$ is the population mean vector of the *i*th subject. Then the data can be defined as (10):

$$y_i \sim (\mu_i, \Sigma_i) \tag{10}$$

We start with the classical approach of modeling longitudinal data analysis for the purpose of understanding and then extending the literature later in 2.3.3. The simplest way of modeling is multivariate data analysis to parameterize the population means of the data by a multivariate linear regression approach. If we believe some *p* variables may affect μ_i, then the linear regression model can be expressed as:

$$y_i = X_i \beta + \varepsilon_i \tag{11}$$

where, X_i is a design matrix ($n_i \times p$) and $\beta = (\beta_1, \beta_2, \dots, \beta_p)^T$ is a vector of parameters representing how much the covariates could affect the response vector y_i. The error term ε_i is a vector of length n_i of unobservable random errors, $\varepsilon_i = (\varepsilon_{i1}, \varepsilon_{i2}, \dots, \varepsilon_{in_i})^T$. The covariates vector has dimension p and can be discrete or continuous and time-independent, if they do not change by time, or time-dependent, if they do.

2.3.2 Major issue in longitudinal data analysis: Covariance structure specification

Longitudinal data was born with the natural ***within-subject-correlation***, so this correlation has to be incorporated into the within-subject covariance. If we suppose that random errors ε_i have mean zero and the following simple independent covariance structure in (12):

$$\Sigma_i = \begin{pmatrix} \sigma^2 & 0 & \cdots & 0 \\ 0 & \sigma^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma^2 \end{pmatrix}_{n_i \times n_i} \tag{12}$$

Then following the process of ordinary least squares, the estimator of β, $\widehat{\beta}$, could be expressed as:

$$\widehat{\beta} = \left(\sum_{i=1}^{n} X_i' X_i \right)^{-1} \left(\sum_{i=1}^{n} X_i' y_i \right) \tag{13}$$

Ordinary least squares estimation is quite simple in computation, but the assumption about the covariance structure Σ_i is very restrictive and not realistic for longitudinal data, where within-subject measurements are correlated. Following the process of Generalized Least Squares (GLS), with the form of Σ_i assumed to be a *known positive definite matrix*, the estimators β are expressed as:

$$\widehat{\beta} = \left(\sum_{i=1}^{n} X_i' \Sigma_i^{-1} X_i \right)^{-1} \left(\sum_{i=1}^{n} X_i' \Sigma_i^{-1} y_i \right) \tag{14}$$

From the generalized least square estimators, we could see that the estimation of β depends on the covariance Σ_i. Good estimates of Σ_i increase the efficiency of the estimates of parameters β and therefore the means. In practice, Σ_i is always unknown and needs to be estimated. Therefore, the estimation of covariance is essential for the mean in longitudinal data analysis.

2.3.2.1 Stationary covariance structure

For most of the models in statistics, time series analysis, machine learning, signal processing, and state space modeling, for the covariance with unknown parameters, a specific stationary structure or in another words, a constrained stationary parameterization of covariance, such as AR(1), Compound Symmetry and so on, is chosen in advance even when the data do not support it. These methods all suffer when the covariance structure is misspecified or is not stationary. Wang and Carey [44] found that the misspecification of covariance may lead to great loss of efficiency of the mean parameter estimation.

2.3.2.2 Non-Stationary Covariance Structure

To tackle the problem of non-stationarity, an unconstrained parameterization of covariance structure, a 'data-driven' method based on modified Cholesky decomposition was developed by Pourahmadi [45, 46]. The formulation of modified Cholesky decomposition has three desirable features: the first is modeling non-stationary dependence structure; the second is the explicit statistical interpretation of the parameters; the third is the assurance of positive-definiteness of the covariance structure [45]. This general model for longitudinal data analysis could be divided into three types, roughly: parametric model, nonparametric model, and semi-parametric model. For our purposes, we only review the parametric models. For non-parametric and semi-parametric please refer to Huang [47].

2.3.3 Parametric modeling of longitudinal data analysis

Diggle et al. [15] and Verbeke et al. [16] provided an excellent overview of the parametric longitudinal methodology for discrete and continuous data, respectively.

2.3.3.1 Normal-based longitudinal data

The first fundamental development for parametric longitudinal analysis is the Linear Mixed-Effects Model (LMM). This model was first suggested by Laird and Ware [48] for normally-distributed longitudinal data.

$$y_i = X_i\beta + Z_i b_i + \varepsilon_i \tag{15}$$

where $b_i \sim N(0.D)$ and $\epsilon_i \sim N(0.R_i),\ i = 1.\cdots,n.$

Here β and b_i represent the fixed and random effects, respectively. The random effects, b_i, are introduced to characterize the individual properties, namely 'heterogeneity,' of the data. With the EM algorithm to estimate the parameters of mean and variance, the LMM is still one of the most popular models for the analysis of normal-based longitudinal data.

2.3.3.2 Non-normal longitudinal data

An intuitive extension of LMMs is to build models which could fit non-normal longitudinal data, such as discrete data. Two main different models are developed to address this purpose: one is the marginal model, the other is the mixed-effect model.

Non -normal longitudinal data: Marginal Models

The development of marginal models originated from Generalized Linear Models (GLMs, Nelder, and Wedderburn [49]). GLM is a unified tool for analysis of either *continuous* or *discrete* distributed data *without correlation*. Liang and Zeger [50] extended GLM to Generalized Estimating Equations (GEEs). In this model, the working correlation matrix $R(\alpha)$ with parameters α are added into the estimating equations, which enable GEEs to accommodate the correlated data patterns, such as longitudinal or functional data. There are many papers that discuss and extend GEE models, which makes GEEs one of the hottest and most developed tools for longitudinal data analysis.

Non -normal longitudinal data: **mixed-effect Models**

In contrast to marginal models in which the mean response measurements only depend on fixed-effects covariates, mixed-effects models parameterize the mean response by both the fixed-effects covariates and

random-effects covariates. One main model of this kind is the Generalized Linear Mixed Model (GLMM), which is just the intuitive extension of LMM (Stiratelli, Laird, and Ware [51], Zeger, Liang and Albert [52], Schall [53], Breslow and Clayton [54]). Up to this point, we have reviewed the Linear Model (LM), Linear Mixed Model (LMM), Generalized Linear Model (GLM), Generalized Linear Mixed Model (GLMM) and Generalized Estimating Equation (GEE) [47]. In addition, another type of mixed-effect model named non-linear mixed-effects model, which can be interpreted as a two-stage form of model, was proposed to accommodate non-linear longitudinal data. For more information, Davidian and Giltinan [55] and Vonesh, Chinchilli and Pu [56] give a detailed illustration of this model.

2.3.4 Closest form of the longitudinal model to state transition modeling

Apart from the non-linear mixed effects model, there are also some other parametric models for non-normal distributed longitudinal data analysis. One of them is the so-called *transition model*, where the mean response measurements are set to be dependent on the previous response measurements, in addition to covariates (Diggle et al. [15]). The idea of the transition model could be traced back to Markov models and autoregressive models in time series. However, since the order of the dependence of response measurements on previous responses is very sensitive, this model has restricted applications (Fitzmaurice [57]). In our study if we assume that first order dependency is enough, and we also can assume that treatments are a special kind of covariates and that we can be parsimonious on treatments, then applying transition models is not possible, as it is obvious that they are developed only for cases with discrete and non-normal response and noise. Our study in terms of longitudinal data analysis is modeling the current continuous response dependent on the previous continuous response and a set of mixed discrete-continuous covariates with normal errors. The only applicable form is the linear mixed effect model (LMM). However, LMM does not explicitly model the dependency of the response on past responses, but rather captures this dependency implicitly by assuming or testing a structured covariance matrix, such as AR(1).

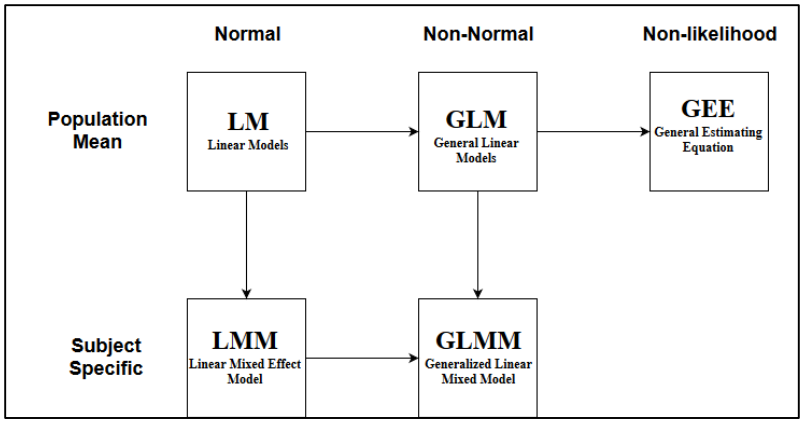


Figure 2. Development of Longitudinal Data Analysis

2.4 Motivation and Contribution

Now we would like to see where our problem of interest lies in the body of state-space models and what is the gap that motivated us for this dissertation.

2.4.1 Defining the problem as a state-space model

Assuming the reader has prior knowledge about directed acyclic graphs (DAG) or causal diagrams [1, 14, 58, 6] , we have a graphical representation (DAG) of the general problem of interest in figure 3.

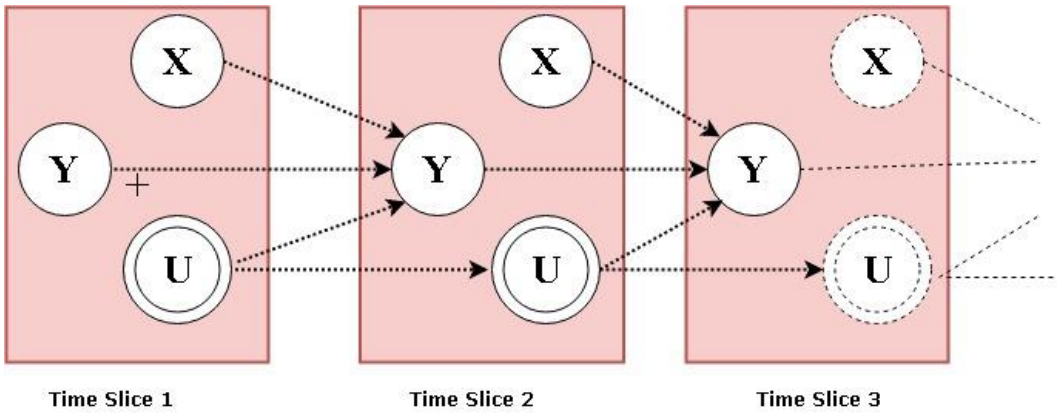


Figure 3. A highly simplified version of our problem in DAG or causal graph. The arcs or edges are dotted line to emphasize that these are hypothesized influence. The Graph is only unrolled in time. That means, X, U and Y can be unrolled based on the problem. U and Y can be unrolled based on the problem. U is double circle here to emphasize its control nature.

Figure 3 is not fully representative of our problem, but is highly simplified and general in terms of dependency structure. Let *Y* be a continuous response of interest. Let *X* be a set of predictors, which is a mixture of continuous-discrete variables, which can be time-invariant or time-variant. The time-variant case is shown in figure 3, and the edge between *X* and *U* is removed, where *U* is the intervention in the system (a.k.a, decision, control or treatment), containing multiple binary control variables (more than one treatment can be activated at a time), and we would like to see its effect on the transition of *Y*. The figure is only unrolled in time slices. That means, *X*, *U*, and *Y* at each time slice can be unrolled to multiple variables of its own and relationships or edges can exist among them. For simplicity, we keep them unrolled here. One difference in the control variable, as opposed to most of the state-space models, is that we also have dynamics between controls (*U*). The edge in figure 3 between treatments is representing the same concept. One can think of *U* as state variables since they are also evolving in the system. We assume for now that the observation and state are jointly Gaussian. We also assume first-order stationarity and Markov. Then we represent DAG in figure 3 in KFMs in two forms, fully observed and partially observed for interested reader in appendix.

2.4.1.1 Fully observed state space

Similar to any other SSM, we define our problem in three steps.

1- The initial state is the past records of the *X*, *Y* in time slice 1.

We take the first action *U* at time slice 1 and observe *Y* in the next stage or time slice 2.

2- Now let $S_t = (X_t, Y_t, U_t)$ represent the state of the system at time *t*. by assuming S_t as all Gaussian continuous, so that we can use Kalman Filter.

The **state transition function** is:

$$\begin{bmatrix} Y_t \\ U_t \\ X_t \end{bmatrix} = \begin{bmatrix} A_1 & A_2 & A_3 \\ 0 & A_4 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} Y_{t-1} \\ U_{t-1} \\ X_{t-1} \end{bmatrix} + V_t \tag{16}$$

$$V_t \sim \mathrm{N}(0, Q), \qquad Q \; is \; the \; following \; covariance \; matrix.$$

$$Q = \begin{bmatrix} Q_y & Q_{y,u} & Q_{y,x} \\ Q_{u,y} & Q_u & Q_{u,x} \\ Q_{x,y} & Q_{x,u} & Q_x \end{bmatrix} \tag{17}$$

As observed in the equation above, the parameters of matrix *A*, or the state transition matrix in this study, are unknown (A_1, A_2, A_3, A_4) and need to be estimated. This is the main focus of our study, state transition modeling, in case of the unknown parameters.

3- Observation Model: $Y_t^O = (1 \quad 0 \quad 0)\begin{pmatrix} Y_t \\ U_t \\ X_t \end{pmatrix}$

All the variables (*X*, *Y*, and *U*) in figure 3 are assumed to be observed and there is no observation noise ($W_t = 0$) *or* $Y_t^O = Y_t$. Hence, there is no need to do state estimation because there is no hidden state. Prediction is straightforward, since there are no future observations on which to condition. The problem from inference and learning shrinks down to only the "learning of *A* and *Q* parameters." This is equivalent to structure learning (model selection) or discovering the true underlying model. From now on in this study, we assume full observability, therefore, from the next section and throughout, we focus on the problem of true underlying feature selection as the first step of uncovering the true underlying model. However, if we assume that the observed behavior of the system is affected by a hidden state or in plain English if we assume partial observability, then the entire model representation, inference and learning, will change. This is out of the scope of our study and for interested readers, the discussion on partially observed is given in Appendix D.

2.4.2 Contribution

In this dissertation, inspired by the pain management case at the Eugene McDermott Center for Pain Management at the University of Texas Southwestern Medical Center in Dallas, we endeavor to develop a feature selection technique that targets "true underlying features" as opposed to "predictive features" in pure observational setting. We first simplified the case and removed the interventions (treatments), and proposed the Iterative Elastic Net (ITELNET) with a unique stopping rule of "similarity condition" for true underlying feature selection under multicolinearity in pure observational setting. Then in the second paper we extend ITELNET to Outcome-adaptive ITELNET (OA-ITELNET) to handle the interventions (treatments) in the observational setting accounting for time-varying confounding and selection bias. The OA-ITELNET

uncovers the causal outcome features and subsequently estimates the unbiased causal effect of treatments, in (1) a multiple treatments setting, (2) with highly correlated state space both among treatments and covariates (3) under the time-varying confounding. We also suggest a way to conduct balance diagnostics for inverse probability of treatment weighting (IPTW) in case of multiple treatments, as a necessary post-evaluation task for every IPTW weighting method.

2.5 Major tools

2.5.1 Regularized (Penalized) Linear Regression

Least Absolute Shrinkage and Selection Operator or LASSO is modern regression technique that was proposed by Tibshirani [59] and performs estimation and variable selection simultaneously. Modern regression methods based on LASSO introduce

some bias but significantly reduce the variance in the non-orthogonal setting, originally designed for better prediction accuracy. Most modern methods for regression can be expressed as:

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \|y - X\beta\|_2^2 \quad \text{subject to } R(\beta) \leq t, \tag{18}$$

Or equivalently:

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \|y - X\beta\|_2^2 + \lambda \cdot R(\beta). \tag{19}$$

The R term is called a penalty or regularizer, and modifying the regression problem in this way is called applying regularization. The original LASSO solves the l_1-penalized regression of finding:

$$\hat{\boldsymbol{\beta}}(\text{lasso}) = \arg\min_{\boldsymbol{\beta}} \left\| \mathbf{y} - \sum_{j=1}^{p} \mathbf{x}_j \beta_j \right\|^2 + \lambda \sum_{j=1}^{p} |\beta_j|, \tag{20}$$

The form of the l_1-regularizer allows estimation and shrinkage of some coefficients to zero, which amounts to feature selection [60]. A number of authors have studied the ability of the LASSO and related penalized methods to recover the correct model.

Despite the fact that LASSO is consistent in prediction or some say "persistent," it is only a consistent variable selector under the "irrepresentability" condition on the design matrix [60, 61]. Let S index the subset of features with non-zero coefficients in the true underlying model, and X_S are the columns of design matrix X corresponding to those features. Similarly S^c are the features with true coefficients equal to zero, and X_{S^c} the corresponding columns. The "irrepresentability" condition says that the least squares coefficients for the columns of X_{S^c} on X_S are not too large, that is, the good variables S are not too highly correlated with the nuisance variable S^c. For example, the orthogonal design guarantees the irrepresentability necessary condition for consistency of the LASSO selection, which is far from reality in many real-world case scenarios with high correlation among predictors.

The adaptive LASSO [62] uses a weighted penalty of the form,

$$\hat{\beta}^{*(n)} = \arg\min_{\beta} \left\| \mathbf{y} - \sum_{j=1}^{p} \mathbf{x}_j \beta_j \right\|^2 + \lambda_n \sum_{j=1}^{p} \hat{w}_j |\beta_j|. \tag{21}$$

where $\hat{w}_j = \frac{1}{|\hat{\beta}|^{\gamma}}$ $\gamma > 0$ and $\hat{\beta}$ can be $\hat{\beta}(\text{OLS})$. When we face high-dimensional data $p > n$, we can use univariate regression coefficients in place of ordinary least squares (OLS). Zou [62] claims and proves the oracle property and consistency of that adaptive LASSO in discovering the true underlying model under a milder condition compared to LASSO. The concept of adaptive comes from the $\hat{w}_j$ which acts as a tuning parameter for different predictors. It makes the penalization more flexible by assigning different degrees of shrinkage to different factors. When using OLS estimators to estimate the adaptive weights, intuitively, a larger degree of penalty is applied to the zero coefficients and a relatively smaller amount is used for coefficients that remain in the model.

Now imagine a situation like genomic applications where there are often strong correlations among the variables with thousands of variables (genes). The LASSO penalty is somewhat indifferent to the choice among a set of strong, but correlated variables and tends to select one feature among the correlated group with no care on the choice. The ridge penalty or l_2-regularizer, on the other hand, outperforms LASSO in this

situation and tends to shrink the coefficient of correlated variables toward each other [63]. Now the attention goes toward a family of models that has a grouping effect, which tends to select correlated variables in a group. The Elastic Net is a compromise, convex combination of the LASSO and ridge penalties and has the form,

$$\hat{\beta}_{(\text{Elastic Net})} = \underset{\beta}{argmin} \left\| y - \sum_{j=1}^{p} x_j \beta_j \right\|^2 + \sum_{j=1}^{p} (\alpha |\beta_j| + (1-\alpha)\beta_j^2). \quad (22)$$

The second term encourages highly correlated features to be averaged, while the first term encourages a sparse solution in the coefficients of these averaged features. In addition to the grouping effect, Elastic Net is particularly useful when we have the curse of high dimensionality, where the LASSO result is not a satisfactory variable selection method in this situation since LASSO cannot select more than $\min(n, p)$. Simulation studies have shown that Elastic Net outperforms LASSO in such situations [64].

Now if we expand the concept of group selection, there are problems for which the predictors belong to "pre-defined groups." In genomic applications, where genes that belong to the same biological pathway or where a collection of indicator (dummy) variables for categorical predictors should be treated as a group. In these situations, it may be desirable to shrink and select the members of a group together. Group LASSO is one way to achieve this and was suggested by Yuan and Lin [65]. Suppose that, d predictors $x_i \in \mathbb{R}^d$ are divided into p pre-defined groups, $x_i = (x_{i1}^T, \ldots, x_{ip}^T)^T$, where $x_{ij} = (x_{ij1}, \ldots, x_{ijd_j})^T \in \mathbb{R}^{d_j}$ and d_j is the size of each group. Then the Group LASSO has the form,

$$\sum_{i=1}^{n} \frac{1}{2} \left(y_i - \sum_{j=1}^{p} x_{ij}^{\top} \beta_j \right)^2 + n\lambda \sum_{j=1}^{p} \|\beta_j\|, \quad (23)$$

where $\|\,.\,\|$ is the l_2-norm. Suppose $d_j = 1$, then the above penalized function reduces to the LASSO. The group LASSO yields sparsity at group level, meaning it either keeps a group or drops it.

The standard Group LASSO algorithm suggested by Yuan and Lin [65] uses coordinate descent and assumes that the design matrix in each group is orthonormal (not just orthogonal), and uses simple soft-thresholding. This is a restrictive assumption; for example, if we have a categorical predictor coded by

indicator variables, we can orthonormalize without changing the problem only if the number of observations in each category is the same. Yang and Zou [66] derive a unified algorithm that handles general design matrix without the orthonormal requirement. However, Group LASSO suffers from inconsistency in estimation and feature selection [62, 67]. Likewise, an adaptive approach for Group LASSO has been developed by Wang and Leng [68] and has the form,

$$Q(\beta) = \sum_{i=1}^{n} \frac{1}{2}\left(y_i - \sum_{j=1}^{p} x_{ij}^{\top}\beta_j\right)^2 + n\sum_{j=1}^{p}\lambda_j\|\beta_j\|. \tag{24}$$

The consistent property of adaptive group LASSO depends on the estimate of $\lambda_j = \lambda\left\|\widehat{\beta}_j\right\|^{-\gamma}$, which can be estimated in the same OLS manner as Zou [62].

Friedman et al. [69] combined LASSO with the Group LASSO to relax the orthonormal design matrix assumption, and the resulting model not only gives sparsity at the group level, but adding LASSO provides sparsity within the group with a number of nonzero coefficients within a nonzero group.

$$\min_{\beta}\frac{1}{2n}\left\|y - \sum_{l=1}^{m} X^{(l)}\beta^{(l)}\right\|_2^2 + (1-\alpha)\lambda\sum_{l=1}^{m}\sqrt{p_l}||\beta^{(l)}||_2 + \alpha\lambda||\beta||_1. \tag{25}$$

where $\alpha \in [0,1]$, a convex combination of the LASSO and Group LASSO. When $\alpha = 0$, equation (25) is equivalent to Group LASSO and when $\alpha = 1$, it is LASSO. Back to the purpose of this study in 1.2, we are employing penalized methods for uncovering the true underlying features (item 1).

2.5.2 Choice of tuning parameter: Uncovering the true underlying model (structure learning)

Every regularization method has an associated tuning parameter: λ in equation (19) or t in equation (18). The tuning parameter controls the amount of regularization, so choosing a good value for the tuning parameter is crucial. In general, if λ or amount of shrinkage increases, the bias also increases, but the variance decreases; also known as the bias-variance trade off. Because each tuning parameter value corresponds to a fitted model,

we also refer to this task as model selection or structure learning in the computer science community. What we might consider a good choice of tuning parameter, however, depends on whether our goal is prediction accuracy or recovering the correct underlying model. Choosing the tuning parameter for the latter purpose is a harder task [60]. Most of the literature, mainly in the computer science community, has focused on the regularized method for the purpose of prediction accuracy; therefore, achieving minimal prediction error is their goal. For that, one of the most frequent metrics in use is *K*-fold cross-validation which is an estimate for prediction error at any fixed value of λ. To achieve this goal, the best tuning parameter is then the one with minimum *K*-fold cross-validation error. The value of the parameter that achieves the smallest cross-validation error often corresponds to not enough regularization for the purpose of recovering the true model. This motivated the development of a unique criteria in Chapter 3 to choose a tuning parameter for the purpose of uncovering the correct features.

Chapter 3

Iterative Elastic Net for Uncovering Underlying State Transition Model Features

Abstract

In the pure observational setting, causal variable selection or uncovering the true underlying model is a practical challenge and a controversial task to accomplish. This is even harder when the data under study are highly correlated and the aim is to detect causation. Under this condition, we propose a methodology that outperforms other current LASSO type modeling techniques in discovering the true underlying model. We examined our proposed method using an experimental design study with 324 simulated cases. Variables are generated from a multivariate normal distribution, and observations are created by defining a linear relationship between the response and defined "causal variables." The most important study factor of interest for design and generation of the data is the unique correlation structure, which controls not only the magnitude of the correlation within causal predictors but also controls the intensity of the correlation between causal predictors and non-causal variables. This allows us to monitor the ability of the methodology in differentiating causation from correlation. The rest of the study factors are the proportion of the number of causal predictors to the total number of variables, the magnitude of regression coefficients and finally the total number of observations. We adopted the confusion matrix suggested by Kubat et al. [70] to evaluate the performance of our suggested methodology compared to current LASSO-type models.

Keywords: Causal variable selection; True Underlying Model, Penalized regression, Tuning parameter selection, LASSO, Elastic Net, Adaptive LASSO, Multicollinearity.

3.1 Introduction and motivation

While predictive models help us to forecast what is going to happen, causal or true models on the contrary are capable of control and change. Back to the famous saying "correlation does not imply causation" [71], it is most common to employ randomized controlled trials to identify causal models. However, there are many real-world situations in which observational data are the only option to develop a decision. Regardless of which statistical methodology is used, building graphical causal models [10] [7] [8] [1] with the help of domain knowledge is a strong tool of causal inference in the observational setting.

In modern regression methods, there are three important factors in recovering the true underlying model. First, the careful design of the penalized function can increases the ability to capture the true underlying model while examining the path of the tuning parameter (λ). Notice that this task entirely depends on the characteristics of the problem and the structure of the observational data. To shed some light on this matter we will present a brief review on how these modern regression functions have been developed and what is the logical motive behind them. Originally, Least Absolute Shrinkage and Selection Operator or LASSO, which was proposed by Tibshirani [59], solves the l_1-penalized regression of finding Eq. (1),

$$\widehat{\beta}^{\text{LASSO}} = \underset{\beta}{\operatorname{argmin}} \|y - X\beta\|^2 + \lambda \sum_{j=1}^{p} |\beta_j| \tag{1}$$

which allows estimation and shrinkage of some coefficients to zero or feature selection simultaneously [59]. [60]. LASSO and its extensions introduce some bias, but significantly reduce the variance in the non-orthogonal setting and increase prediction accuracy, which is the most common goal of its application in practice.

Despite the fact that LASSO is "consistent in prediction" or say "persistent," it is only a consistent variable selector under the "irrepresentability" condition on the design matrix [60] [61]. We open up this discussion for the reader since we inherit this concept in design of the correlation structure in section 3. Let

S index the subset of features with non-zero coefficients in the true underlying model, and X_S be the columns of design matrix X corresponding to those features. Similarly S^c are the features with true coefficients equal to zero, and X_{S^c} the corresponding columns. The "irrepresentability" condition says that correlation coefficients for the column of X_{S^c} on X_S are not too large, that is, the good variables S are not too highly correlated with the spurious variable S^c. The orthogonal design guarantees the irrepresentability necessary condition for consistency of the LASSO selection, which is far from reality in the pure observational setting, where high correlation among variables is present. To relax the irrepresentability condition, with a small change in LASSO function, Zou proposed the adaptive LASSO [62] that uses a weighted penalty in Eq. (2),

$$\hat{\boldsymbol{\beta}}^{*(n)} = \arg\min_{\boldsymbol{\beta}} \left\| \mathbf{y} - \sum_{j=1}^{p} \mathbf{x}_j \beta_j \right\|^2 + \lambda_n \sum_{j=1}^{p} \hat{w}_j |\beta_j|. \tag{2}$$

where $\hat{w}_j = \frac{1}{|\hat{\beta}|^\gamma}$ $\gamma > 0$ and $\hat{\beta}$ can be $\hat{\beta}(ols)$. Zou [62] claims and proves the oracle property and consistency of that adaptive LASSO in variable selection under a milder condition compared to LASSO.

Now imagine a situation where there are strong correlations among the variables and there are thousands of variables from which to choose. LASSO and adaptive LASSO penalties are somewhat indifferent to the choice among a set of strong, but correlated variables, and tends to select one feature among the correlated group with no care about the choice. There are problems for which the predictors belong to "pre-defined groups." For instance, in genomic applications, where genes that belong to the same biological pathway or where a collection of indicator (dummy) variables for categorical predictors should be treated as a group. In these situations, it may be desirable to shrink and select the members of a group together. Group LASSO is one way to achieve this and was suggested by Yuan and Lin [65]. Suppose that, d predictors $x_i \in \mathbb{R}^d$ are divided into p pre-defined groups, $x_i = (x_{i1}^T, \ldots, x_{ip}^T)^T$ where $x_{ij} = (x_{ij1}, \ldots, x_{ijd_j})^T \in \mathbb{R}^{d_j}$ and d_j is the size of each group. Then the Group LASSO has the form,

$$\sum_{i=1}^{n} \frac{1}{2} \left(y_i - \sum_{j=1}^{p} x_{ij}^{\top} \beta_j \right)^2 + n\lambda \sum_{j=1}^{p} \|\beta_j\|, \tag{3}$$

where $\| . \|$ is the l_2-norm. Suppose $d_j = 1$, then the above penalized function reduces to the LASSO. The Group LASSO yields sparsity at group level, meaning it either keeps a group or drops it.

If we shift the grouping need from "pre-defined" to "a natural grouping effect," the combination of ridge penalty or l_2-regularizer [63], and LASSO penalty creates a family of models called "Elastic Net," which is a compromise, convex combination of LASSO and ridge penalties and has the form,

$$\hat{\beta}^{\text{Elastic Net}} = \underset{\beta}{\operatorname{argmin}} \left\| y - \sum_{j=1}^{p} x_j \beta_j \right\|^2 + \sum_{j=1}^{p} (\alpha \| \beta_j \|_1 + (1-\alpha) \| \beta_j \|^2) \quad (4)$$

The second term encourages highly correlated features to be averaged, while the first term encourages a sparse solution in the coefficients of these averaged features. In addition to the natural grouping effect, Elastic Net is particularly useful when we have the curse of high dimensionality, where the LASSO result is not a satisfactory variable selection method since LASSO cannot select more than $\min(n, p)$. Simulation studies have shown that Elastic Net outperforms LASSO in such situations [64].

The second most important factor in discovering the true underlying model in modern regression techniques is the choice of the metric to choose the tuning parameter. A great number of publications in modern regression techniques, place an emphasis on prediction accuracy when it comes to comparing the performance. If a researcher's goal is to obtain a close estimation of the true underlying model for "generalization and causal inference," the model selection based solely on the best fit and prediction accuracy, often selects an unnecessarily complex model that overfits the data and, hence, generalizes poorly on other data generated from the same underlying model [72].

Obtaining minimum error by *K*-fold cross validation is the most popular method used by the computer science community for building predictive models. However, it has shown poor performance in true model recovery and true feature selection [61]. On the other hand, the one standard error (1SE) metric proposed by Breiman, Friedman, Ohlsen and Stone [73] provides more parsimony without harming accuracy, by selecting

the smallest model for which the cross validation error lies within one standard error of minimum CV. Simulation studies have demonstrated its supremacy in capturing the true underlying model compared to minimum CV [61]. The simplest reason is that the true model often times requires and corresponds to a larger lambda, however, the tuning parameter selected by minimum *K*-fold CV is not large enough. There are other model selection techniques (e.g., information criteria) that are beyond the scope of this paper, and we encourage the interested reader to refer to [61] for a quick review of model selection metrics.

The third and final factor in recovering the true model is the adaptive concept $\widehat{w}_j$ from Eq. (2), which enables different tuning parameters for different predictors. It makes the penalization more flexible by assigning different degrees of shrinkage to different factors. When using OLS estimators to estimate the adaptive weight, intuitively, a larger degree of penalty is applied to the zero coefficients and a relatively smaller amount is used for coefficients that remain in the model. In other words, true predictors stand a higher chance for selection. Therefore, this concept should be an inextricable part and backbone of any proposed penalized method that tries to capture true or causal model. When high correlation is present, it is suggested to incorporate ridge or close to ridge estimates as the adaptive weights.

What we endeavor here is to incorporate and combine these concepts and develop a LASSO extension methodology that outperforms other methods in discovering the true underlying model features in the presence of high multicollinearity. Elastic Net and other modern regression methods with a grouping effect (e.g., group LASSO), can shrink the effect towards correlated variables and select a group. However, there arises a question when the true underlying features are surrounded by a more complex correlation structure, for instance, when a high correlation exists between some of causal (relevant) and spurious (irrelevant) variables. Then, is there any method that increases the probability of distinguishing causation from correlation? We propose the Iterative Elastic Net (ITELNET) as a special case of an iterative penalized regression with a unique stopping criteria to augment the true feature selection property. Our ITELNET approach is compared against some current LASSO based methods in a designed simulation study.

The rest of the paper is organized as follows. Section 2 describes the phases of the iterative penalized regression algorithm, in general. The simulation design, performance measure, and candidate penalized

methods are given in section 3. Simulation results and discussion are presented in section 4. Section 5 represents the real case study, and section 6 concludes the paper with a summary and future research.

3.2 General description of Iterative Penalized Regression

Preparation

Input. The full model

Like any other algorithm that starts with an input, at this phase the input for ITELNET is the full model.

Specify the base penalized function

We explained in the introduction how the penalized techniques are applied and chronologically developed in practice as a direct consequence of the complexities of the problem, the properties of the data and finally the goal of the study. This pre-step acts as an input base penalized function which repeats during the iterative process of the algorithm until it achieves the stopping criterion. Not choosing the right penalized based function, consequently, will lead to a chain of poor results in the body of iterative process.

Since we are addressing the situation of highly correlated data, we should take into account the family of LASSO methods with a grouping effect to make sure no true predictor is dropped in phase 2. The reason is that when high correlation exists between a true predictor and a spurious variable, it is possible that the irrelevant variable is selected while the true predictor is dropped. If we were to choose Group LASSO and its extensions in the first phase, we would have to know which variables are pre-defined in the same group. The adaptive Elastic Net in Eq. (5) is used in the body of ITELNET:

$$\min_{\beta} \left\| y - \sum_{j=1}^{p} x_j \beta_j \right\|^2 + \lambda \sum_{j=1}^{p} w_j (\alpha |\beta_j| + (1-\alpha)|\beta_j|^2) \quad (5)$$

Since the focus here is sparsity within the grouping effect, $\alpha = 0.5$ is suggested by Hastie and Qian [74].

3.2.1 Phase 1. Compute the adaptive weights

In phase 1, the adaptive weights are calculated by estimating the coefficients β. The features entering this phase for estimating β are the full model in the first iteration and the reduced model in the second iteration and subsequent iterations. Again, the modeling method to obtain the weights depends on nature of the observational data and can be obtained by OLS, ridge or close to ridge estimates, if we have a correlated data space, and if needed, weighted least squares (WLS). Considering high correlation in the setting under study here, we use close to ridge estimates for adaptive weights $\widehat{w}_i$. This can be achieved by $\alpha = \varepsilon$ for some very small $\varepsilon > 0$ in the Elastic Net in Eq. (4).

3.2.2 Phase 2. Employ the adaptive penalized method chosen in input and choose features by minimum K-fold CV and one standard error (1SE)

The idea in the first phase is to guarantee that the first set of selected features "includes" the true underlying model with an excellent chance. It is shown that the "true" model tends to be a subset of the one that maximizes our estimate of predictive performance [75, 76, 77] . That is to say if the true model is the set of predictors A, and the penalized model selects the subset $\hat{A}(\lambda)$, then we want to make sure that in the first phase with strong probability we achieve (6);

$$A \subseteq \hat{A}(\lambda) \tag{6}$$

Simulation studies have shown that although minimum CV has poor performance in true variable selection, it guarantees condition (4) with high likelihood [61, 75, 76, 77]. In addition, the features selected by one standard error (1SE) $|\hat{A}_{\lambda}^{1SE}|$ are also recorded for active use in the stopping rule in the next step. The path of the iterations seeks to drop and shrink more variables until we get as close as possible to the true model with minimum inclusion and exclusion of spurious variables and true predictors, respectively.

3.2.3 Phase 3. Check the "similarity condition" $|\widehat{A}_{\lambda}^{\text{minimum CV}}| - |\widehat{A}_{\lambda}^{\text{1SE}}| \leq h$?

We need to provide a logical explanation behind the convergence rule first. Minimum CV seeks to find the value of the tuning parameter (λ) that minimizes the average error over all folds. Suppose the set $\{1, \dots, n\}$ is divided into *K* folds of roughly equal size randomly, $F_1, \dots, F_K$.

For $k = 1, \dots, K$, consider training on $(x_i, y_i), i \notin F_K$, and validating on $(x_i, y_i), i \in F_K$. $\hat{\mathrm{F}}_{\lambda}^{-\mathrm{k}}$ is the estimate on the training set. For each tuning parameter (λ), we compute the average error over all folds,

$$CV(\lambda) = \frac{1}{n}\sum_{k=1}^{K}\sum_{i\epsilon F_k}(y_i - \hat{f}_{\lambda}^{-k}(x_i))^2 \tag{7}$$

Minimum CV chooses the value of lambda that minimizes $CV(\lambda)$.

$$\hat{\lambda} = \underset{\lambda\epsilon\{\lambda_1, \lambda_2, \dots, \lambda_m\}}{\operatorname{argmin}} CV(\boldsymbol{\lambda}) \tag{8}$$

To obtain the standard errors for cross validation, we first average the validation errors in each fold by Eq. (9).

$$CV_k(\lambda) = \frac{1}{n_k}\sum_{i\epsilon F_k}(y_i - \hat{f}_{\lambda}^{-k}(x_i))^2 \tag{9}$$

where $n_k$is the number of points in the *k*th fold. Finally, we estimate the standard error of $CV(\lambda)$ by Eq. (10).

$$SE(\lambda) = \sqrt{\frac{\sum_{k=1}^{K}(CV_k(\lambda) - CV(\lambda))^2}{K}} \tag{10}$$

The one standard rule results in a sparser model by increasing the regularization from minimum CV as much as it can, such that the cross-validation error curve $CV(\lambda)$ is still within one standard error of minimum CV [73]. Notice that $CV(\lambda)$ in Eq. (6) is in fact the mean of the validation errors in each fold. If the standard error $SE(\lambda)$ of the validation errors is too small, the deviation of all folds from the mean is too small on average, thereby the values of lambda chosen by CV and 1SE are very close to each other, suggesting that the model and features selected by two methods are in fact behaving similarly in all folds

and stands as an indication of getting closer to the true underlying model. By the same logic, we use the concept of distance between 1SE and minimum CV, instead of using each one of them alone, in an iterative process to develop a new feature selection criteria that targets recovering the true underlying model. Subsequently, this can be interpreted in another way. Literally when the selected lambda values by two methods are very close, they choose almost the same number of features. Therefore, at each stage we check if:

$$|\hat{A}_{\lambda}^{\text{minimum CV}}| - |\hat{A}_{\lambda}^{\text{1SE}}| \leq h \tag{11}$$

We call this the "similarity condition." If we set the difference to zero, we might force the model to drop too many variables than necessary. One might evaluate the ideal difference in a separate study, however, the default $h = 1$ seems the reasonable value since we need minimum CV and 1SE to be close, but not so much $(h = 0)$ that overpenalizing could result in the dropping of too many true predictors. The cardinality of the selected subset of features by minimum CV varies from the cardinality of selected features by 1SE method with maximum one unit in Eq. (10). This idea is embedded in the iterative process depicted in figure 4, where it starts with full model, and ends at a reduced model that satisfies Eq. (11).

If the number of selected features by minimum CV and 1SE is not equal at the end of each iteration, in the next iteration, the algorithm starts at phase 1 with the "reduced model" chosen by minimum CV in the previous iteration and updates the adaptive weights by the new features and then repeats phase 2 until it satisfies Eq. (10). If the stopping criterion is satisfied, we extract the selected model by minimum CV at this stage as the best estimate of true underlying model. By this framework, with the help of 1SE and the iterative process, we turn minimum CV into a true feature selection metric. Figure 4 illustrate the iterative process at a glance.

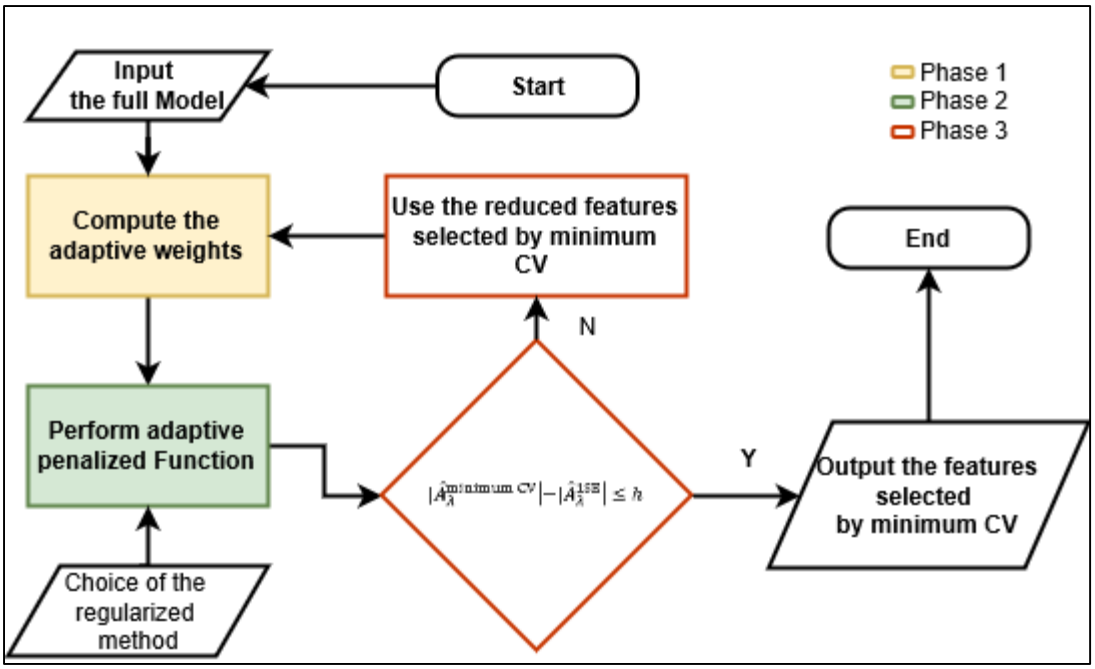


Figure 4. The process of iterative penalized regression in general

3.3 Design of the experimental study and performance measure

3.3.1 Simulation design

We generate datasets by assuming the true outcome follows an additive linear relationship with true predictors.

$$y_i = \sum_{j=1}^{`p} \beta_j x_{ij} + \varepsilon_i \qquad \text{where } \varepsilon_i \sim N(0,1) \tag{12}$$

The cases have been designed considering five factors:

(1) total number of variables p= {10,20}

(2) number of observations n={100,250,500}

(3) proportion of true (causal) predictors among all variables. Prop = {0.3, 0.5, 0.7}

(4) correlation structure.

(5) magnitude of the effects (coefficients) whether [0.4, 0.6] or [0.8, 1].

Respectively from (1) to (5), we have full factorial combination of 2*3*3*(3*3)*2 =324 cases. For each case, 100 replications are generated to evaluate the performance of the true model recovery on average.

3.3.2 Performance measure

If the true model is the set of predictors A and the penalized model selects the subset $\hat{A}(\lambda)$, then these measures are given by,

1- The probability of choosing the correct subset (PCS) [61].

$$\mathrm{PCS}(\lambda) = \mathrm{P}(\hat{A}(\lambda) = A) \tag{13}$$

2- The probability that correct model is the subset of selected model by penalized function.

$$\mathrm{PIS}(\lambda) = \mathrm{P}\left(A \subseteq \hat{A}(\lambda)\right) \tag{14}$$

Ideally, discovering the correct subset (CS) is desired in causal feature selection and is very hard task to achieve in reality. This is the reason that a milder condition is imposed in the second metric (IS). However, a better compromise metric is desired. Hence, for evaluation of the overall performance of the methods, we adopt the confusion matrix in table 1, which is similar to type I and II errors of hypothesis testing in statistics. This table contains information about true and estimated classes and is adopted from the information retrieval community [78] [79] [80]. From table 1, accuracy, sensitivity and specificity are respectively defined as Eqs. (15)- (17).

$$\text{Accuracy} = (a + d) / (a + b + c + d) \tag{15}$$

$$\text{Sensitivity} = d / (c + d) \tag{16}$$

$$\text{Specificity} = a / (a + b) \tag{17}$$

Accuracy is a biased measure when classes are not balanced [78], and it is also a challenge to have a single measure that enables easiert comparison. We use the geometric mean of sensitivity and specificity in Eq. (18), suggested by Kubat et al. [78], which is a robust measure for imbalanced proportions of true and spurious predictors.

Table 2. Confusion Matrix

Confusion Matrix		Predicted Model	
		Spurious	**True**
True Model	**Spurious**	a	b
	True	c	d

$$\text{g-mean} = \sqrt[2]{\text{Sensitivity} * \text{Specificity}} \tag{18}$$

Sensitivity or positive accuracy is the proportion of selected true features among all true features. Specificity or negative accuracy is the proportion of unselected spurious variables among all spurious variables. The value of g-mean moves between zero and one. If g-mean gives a value close to 1, it implies that most of the variables are classified correctly.

3.3.3 Candidate methods for comparison

Among modern regression techniques we compare our methodology to three methods. For the purpose of recovering the true underlying model in a highly correlated space, all of the candidate LASSO type methods should have two proper ties. First, all of them must incorporate the adaptive concept, as mentioned in section 1. Second, they should be a consistent variable selector in a correlated state space. The adaptive LASSO is shown to be a consistent variable selector when there are high correlations between true predictors and spurious [62]. From the model selection perspective, both minimum CV and 1SE are being tested on the adaptive LASSO. The last technique to compare is the Elastic Net with identical parameter settings to our base model in the body of ITELNET. The reason is that in our proposed iterative process, Elastic Net is employed as the base penalized function. One might wonder at the cost of the iterative computational time, whether the ITELNET outperforms the Elastic Net and whether the new model selection technique discussed in 2.3 will outperform the 1SE rule in the task of true feature selection. Close

to ridge estimate is used for adaptive weights in all methods. We conclude this section by listing the candidate penalized regressions:

(a) Adaptive LASSO with close to ridge adaptive weights (minimum CV)

(b) Adaptive LASSO with close to ridge adaptive weights (1SE)

(c) Adaptive Elastic Net (alpha=0.5) with close to ridge adaptive weights (1SE)

(d) ITELNET (alpha = 0.5) with close to ridge adaptive penalty at each stage (proposed methodology).

3.4 Simulation results

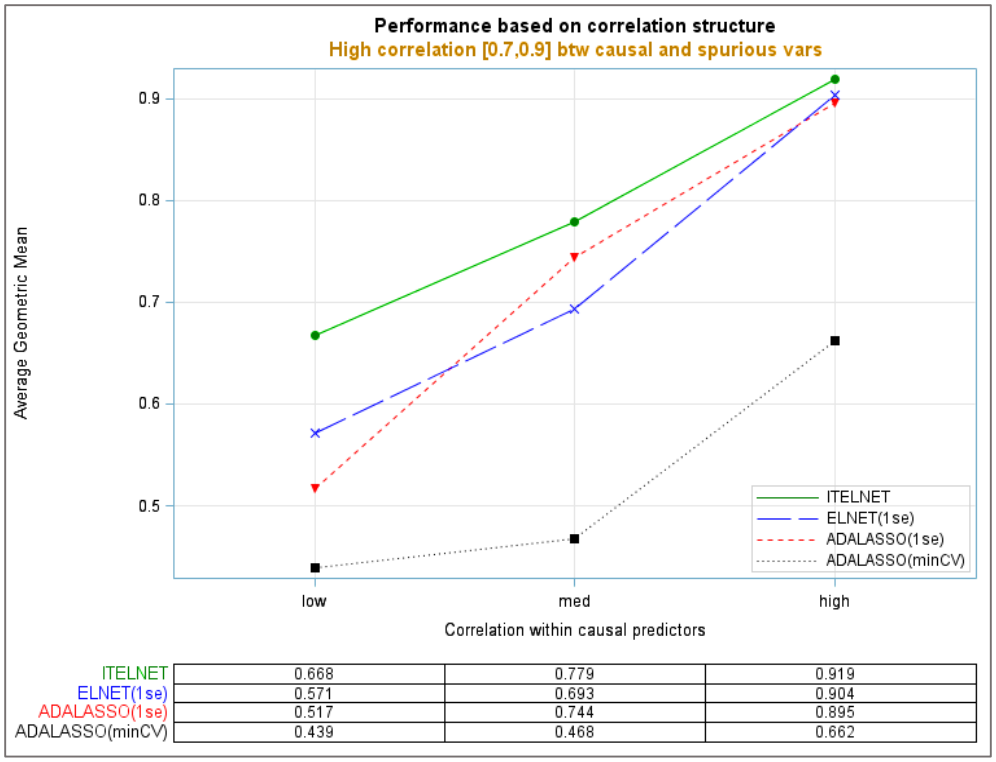
As mentioned before, 100 replications for each of the 324 simulated cases are created to compare the performance of the aforementioned methods in recovering true underlying model features. At each replication, the confusion matrix was formed and g-mean was calculated. Then for each study factor of interest (e.g., the correlation structure), g-mean was averaged for comparison. We also provide PCS and PIS as complimentary pieces of information for evaluating overall comparative results when studying the number of observations. For the rest of the factors, the result is presented only based on average g-mean. Apart from true model recovery metrics, the root mean square prediction error (RMPSE) of the predicted response was also obtained to examine the task of prediction performance versus true model recovery.

3.4.1 Comparison based on g-mean

Before we provide a detailed interpretation of the result, by a quick look at figures 2-7 illustrate the overall superiority of our proposed ITELNET across all study factor combinations. Now let us discuss each study factor of interest.

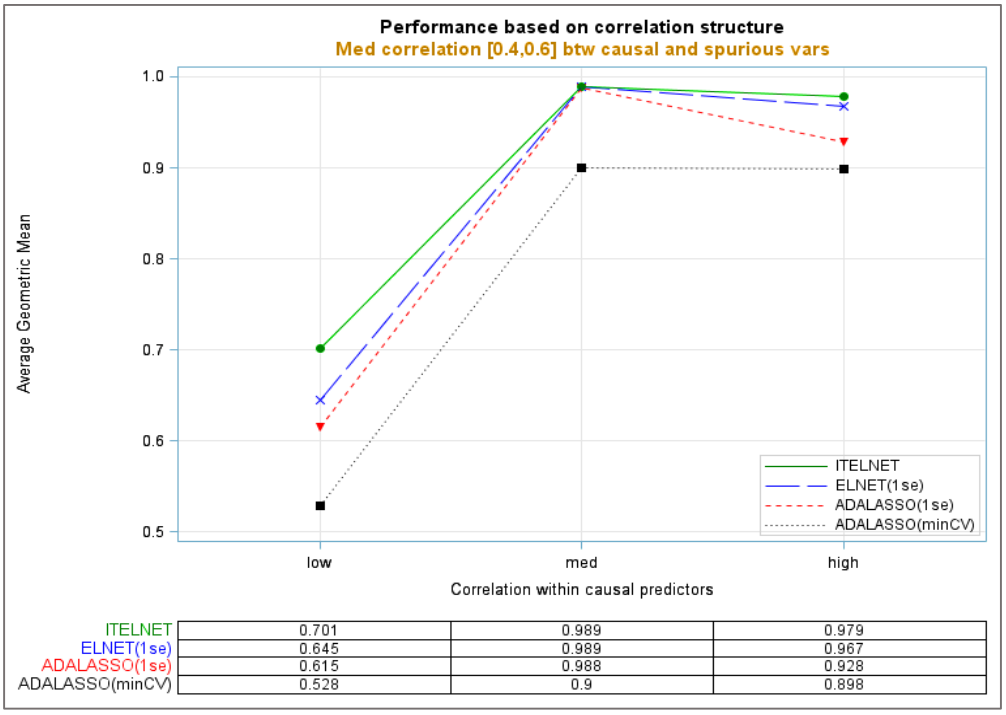
3.4.2 Correlation structure

Figures 2-4 are formed based on three levels of the correlation structure between causal and spurious variables, low, medium and high. The associated averaged g-mean values are provided in an attached table below each figure. In addition, within each block the magnitude of correlation within causal predictors also varies from low to high. The worst case scenario of detecting causal features in reality happens in figure 5, when there is a high correlation between causal and spurious variables. This condition is milder in figures 6-7. The proposed method, ITELNET with the green solid line, is observed above all other methods in all three figures, suggesting better model recovery performance, most significantly in the worst case scenario in figure 5. In addition, it seems that all these modern regression techniques follow some general patterns. The highest performance peak of each method happens when between correlation matches the within correlation level, and it decreases as these two correlation structures become farther from each other. This explains the upward and downward trend in figure 5 and figure 7, respectively, and the medium peak at figure 6. It also tells us that if the overall correlation level in a data set is observed to be in a certain range of low, medium or high, we can expect all methods in this study to have relatively high performance. In particular, ADALASSO(minCV) clearly has the lowest performance in all structures in figures 5-7. This confirms that predictive models based on minimum CV do not achieve the modeling purpose of discovering true features. Figure 7 is an indication of an overall very close and high performance of all methods when the between correlation is very low. If we look at the performance between methods from figures 5 to 7, if we exclude the adaptive LASSO with minimum CV, we realize that when between correlation gets stronger the performance become more distinguishable.



ITELNET	0.668	0.779	0.919
ELNET(1se)	0.571	0.693	0.904
ADALASSO(1se)	0.517	0.744	0.895
ADALASSO(minCV)	0.439	0.468	0.662

Figure 5. Averaged g-mean performance based on correlation structure: High correlation between true and spurious predictors



ITELNET	0.701	0.989	0.979
ELNET(1se)	0.645	0.989	0.967
ADALASSO(1se)	0.615	0.988	0.928
ADALASSO(minCV)	0.528	0.9	0.898

Figure 6. Averaged g-mean performance based on correlation structure: Medium correlation between true and spurious predictors

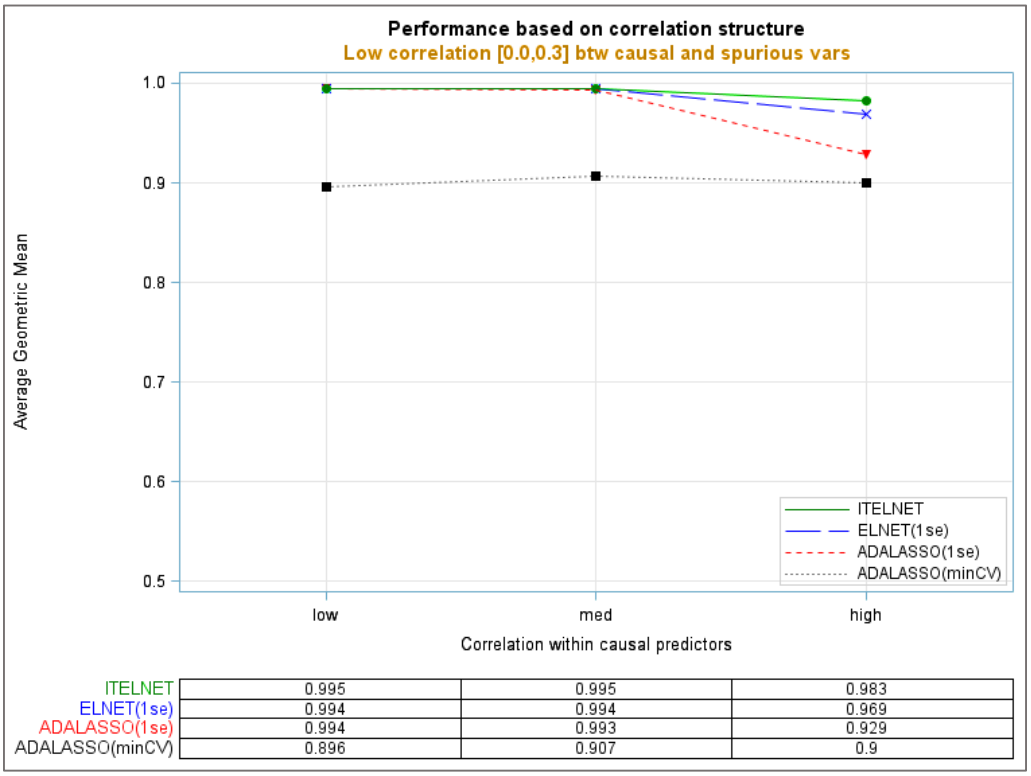


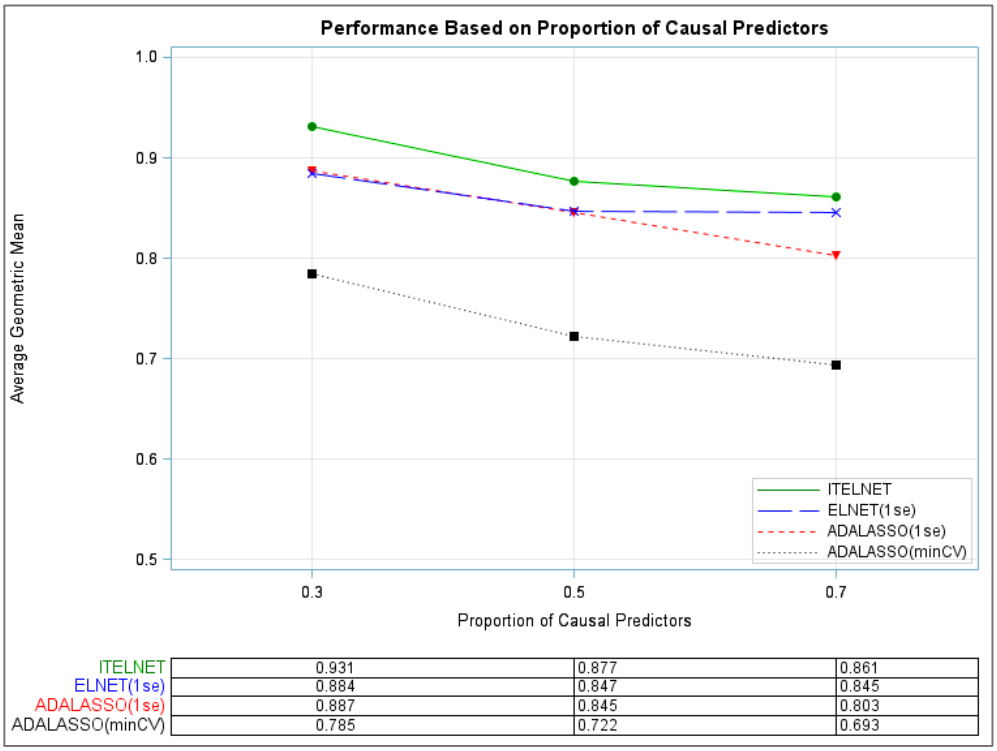
Figure 7. Averaged g-mean performance based on correlation structure: Low correlation between true and spurious predictors

3.4.3 Proportion of true predictors

Figure 8 presents the g-mean performance according to the proportion of causal predictors along with associated values at the bottom of the figure. From the proportion perspective factor, similar to the correlation structure factor, the superiority of ITELNET and inferiority of minimum CV are observable. This figure also suggests the performance decreases as the proportion increases.

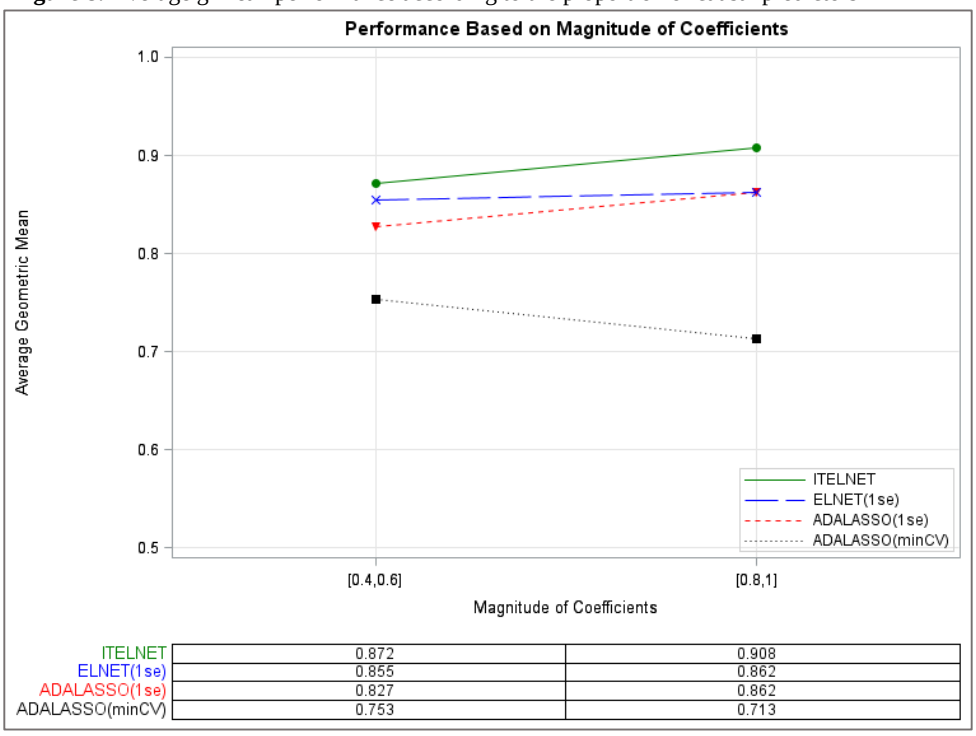
3.4.4 Magnitude of the coefficient

From figure 9, as expected, the overall performance increases with magnitude of the effect, except for adaptive LASSO with minimum CV. This downward odd behavior cannot be explained, but is also evident in another simulation study by Chong et al. [79].



ITELNET	0.931	0.877	0.861
ELNET(1se)	0.884	0.847	0.845
ADALASSO(1se)	0.887	0.845	0.803
ADALASSO(minCV)	0.785	0.722	0.693

Figure 8. Average g-mean performance according to the proportion of causal predictors



ITELNET	0.872	0.908
ELNET(1se)	0.855	0.862
ADALASSO(1se)	0.827	0.862
ADALASSO(minCV)	0.753	0.713

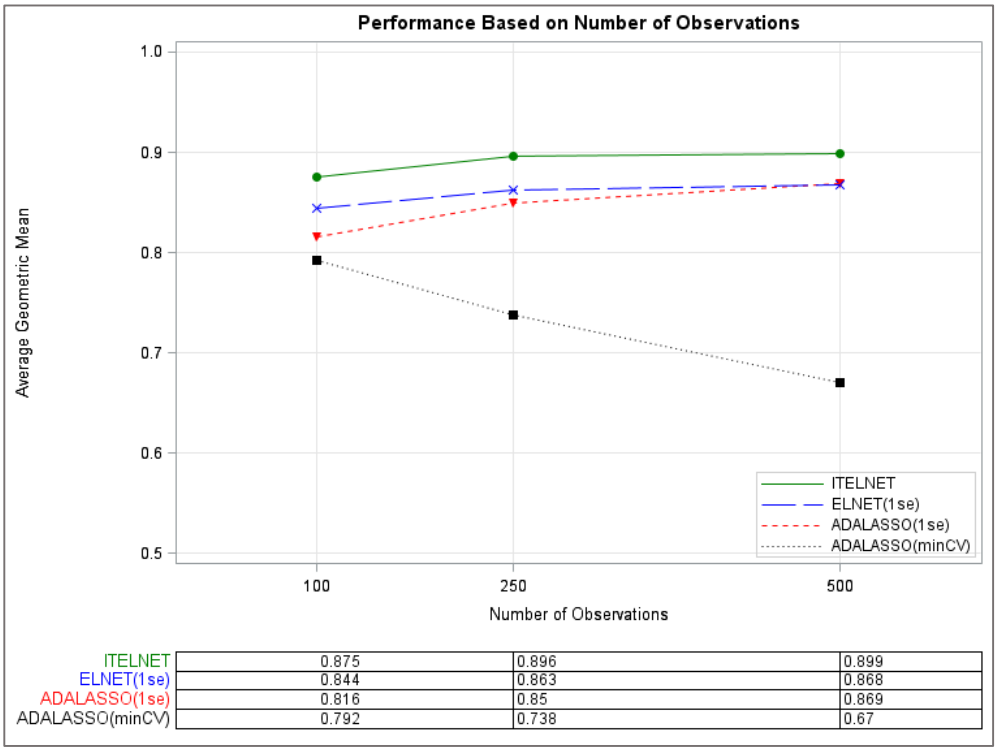
Figure 9. Average g-mean performance according to the magnitude of coefficients

3.4.5 Number of observations

The compiled result based on number of observations in figure 10, provides an overall view of the methods altogether. Here we look at g-mean and compare our findings to PCS and PIS to get an additional perspective on overall performance. Again, the ITELNET outperforms the other methods, followed by, in order, the Elastic Net, the adaptive LASSO with 1SE, and the adaptive LASSO with minimum CV. The slight increase in performance when N goes up is also sensible. What is interesting is that there is only a slight change in the ITELNET model recovery performance when the number of observations increase.

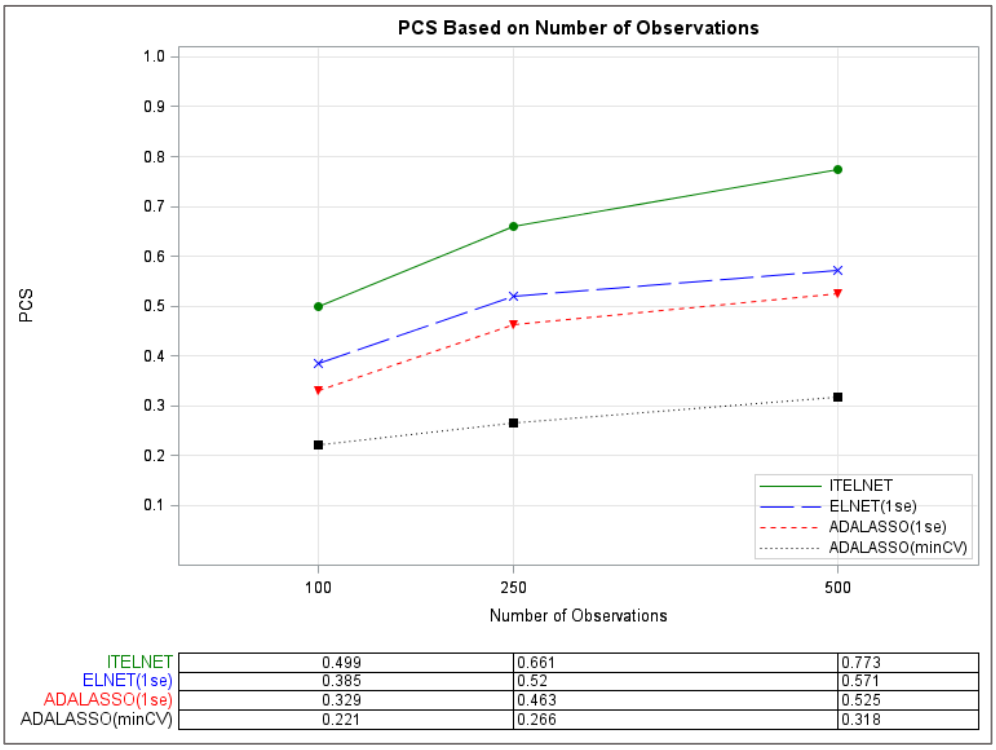
Figure 11 demonstrates the overall performance from the PCS perspective. Notice that if someone was to form a confusion matrix for PCS, the value of g-mean would equal to 1, meaning that we reach the ideal when all the variables are classified correctly. The large gap between ITELNET and the next method suggests its strength in true feature selection. Conversely, from the PIS perspective in figure 12, the results are somewhat reversed.

If we form the g-mean for PIS, we find out that the g-mean value only depends on "a" in the confusion matrix, that is, the correct classification of spurious variables. The fact that minimum CV is superior in PIS is telling us that as N goes up, minimum CV captures more "IS" models but with a greater number of spurious variables or false positives. This confirms the fact that minimum CV is capturing more complex models that may be suitable for prediction of response, rather than interpretation. This also explains the odd downward trend in figure 10; as N goes up, the number of false positives in minimum CV also increases, which consequently results in dropping the average g-mean performance.



ITELNET	0.875	0.896	0.899
ELNET(1se)	0.844	0.863	0.868
ADALASSO(1se)	0.816	0.85	0.869
ADALASSO(minCV)	0.792	0.738	0.67

Figure 10. Average g-mean performance according to the number of observations



ITELNET	0.499	0.661	0.773
ELNET(1se)	0.385	0.52	0.571
ADALASSO(1se)	0.329	0.463	0.525
ADALASSO(minCV)	0.221	0.266	0.318

Figure 11. PCS compiled at number of observations

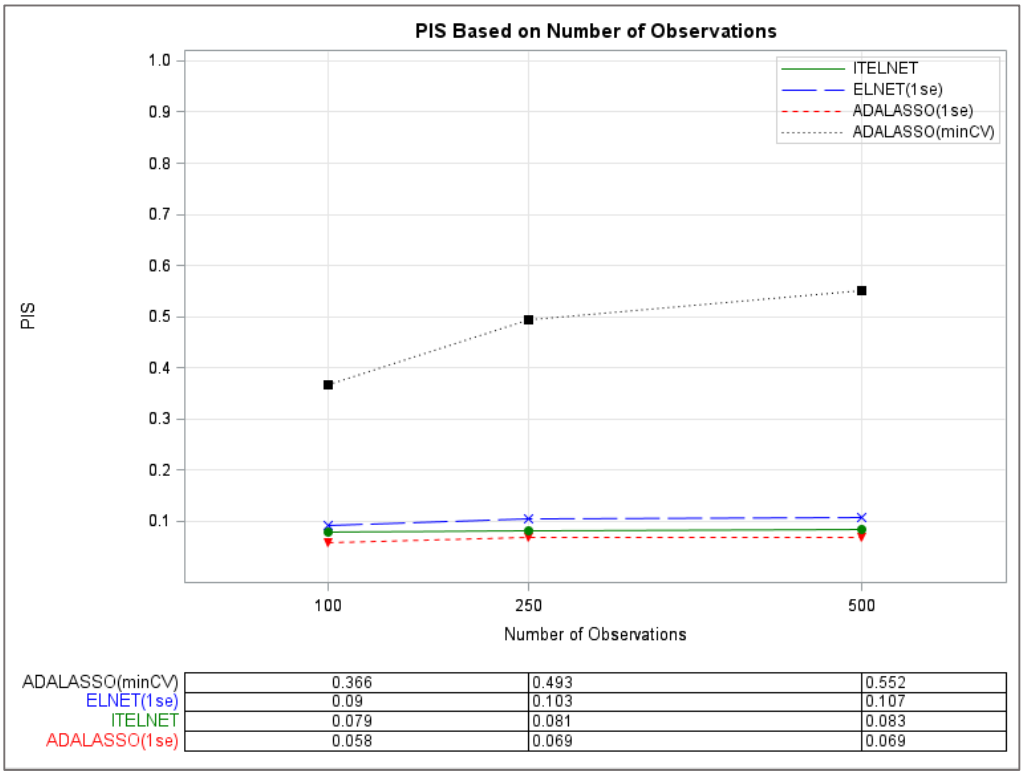


Figure 12. PIS compiled at number of observations

3.4.6 Comparison of prediction performance based on RMSE and RMSPE

One interesting question about which one might be speculating is how the model recovery performance is related to prediction performance. We can compile the simulation result now on an entirely different measure. Root mean square error (RMSE) obtained by Eq. (19) is illustrated in Figure 13. Notice that the values of the y-axis at each case index is the average value of RMSE for all 100 replicated data sets of the same setting. Interestingly, the exact opposite pattern to g-mean is evident in figure 13, suggesting that methods with the priority of high prediction are to be dominated in the true model recovery task and causal model building.

$$\text{RMSE} = \sqrt{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2/n} \tag{19}$$

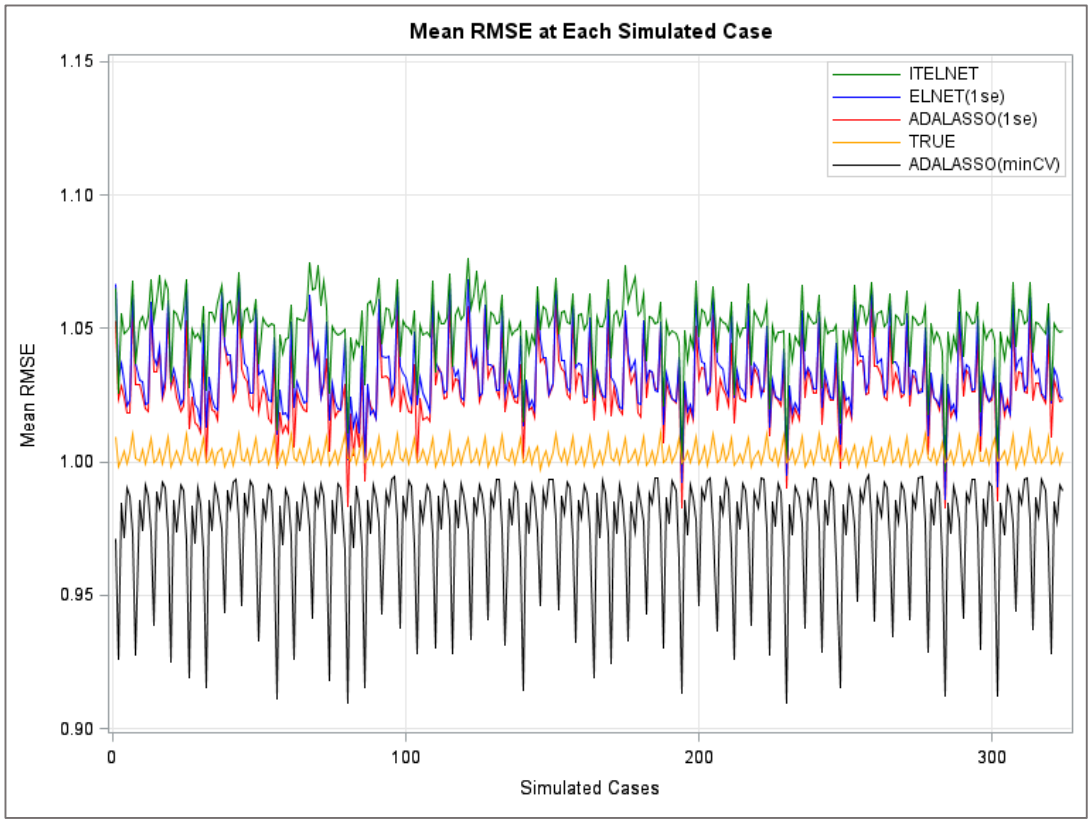


Figure 13. Mean RMSE for the training data at each simulated case

In addition to the training set, by generating the new test set from the same functional underlying model, the actual root mean square prediction error (RMSPE) can be estimated by Eq. (20). A total of 100 test sets are created within which the number of observations is set to 500. Figure 14 sheds light on another perspective in the modeling that might look counter-intuitive, generalizability. It seems that minimizing the training error is not necessarily "generalizable" to the actual test error and might even act conversely. Observing and comparing the patterns in the compiled result (figures 5-14), gives us a similar insight as [72] on how the best fitted model methods, such as minimum CV, are more likely to build an over-complex estimated model (high PIS) that is far from the true underlying model and less generalizable and less-interpretable.

$$\text{RMSPE} = \sqrt{\sum_{i=1}^{n}(y_i{}^{\mathcal{N}} - \hat{y}_i)^2 / n^{\mathcal{N}}} \qquad (20)$$

where $(y^{\mathcal{N}})$ is the new data set.

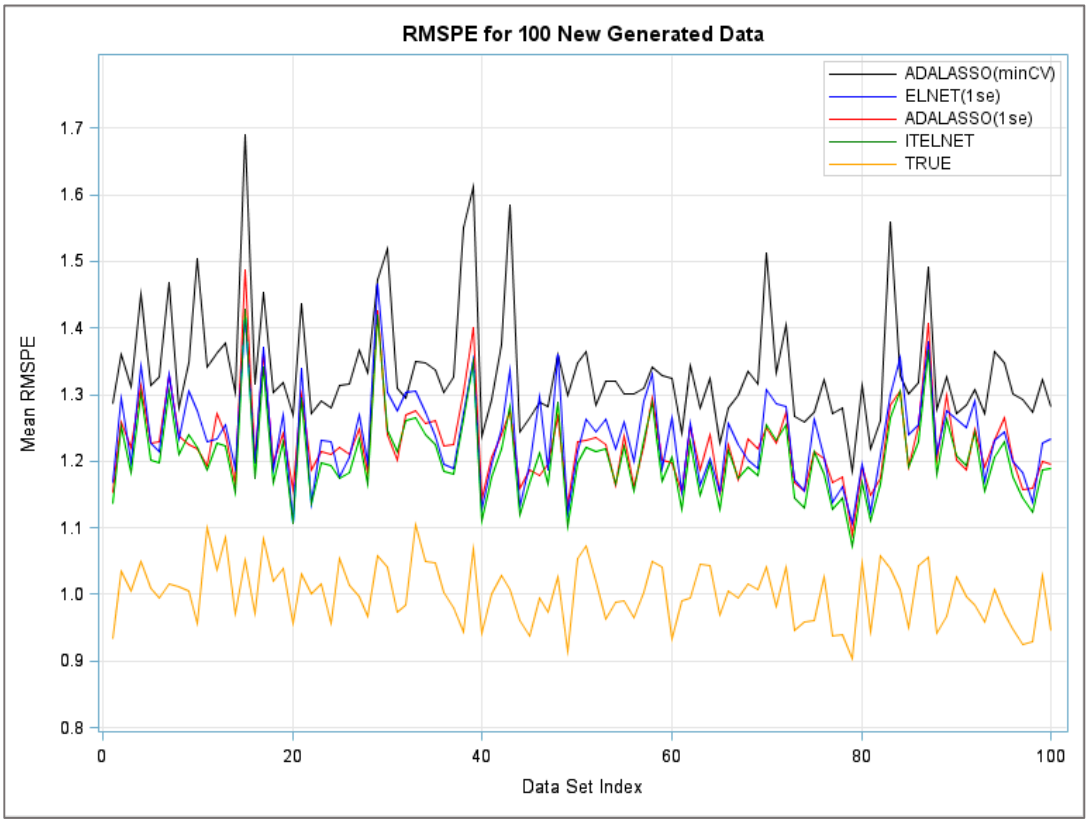


Figure 14. RMSPE for the test data sets (100 sets each 500 observations)

3.5 Summary and future research

In this paper, we proposed a "general iterative regularized method" with a unique "similarity condition" for uncovering the true underlying features under a variety of data structures. We then proposed the use of ITELNET in the highly complex correlated data space. The compiled computational results for the full combination of study factors, suggests that ITELNET outperforms other LASSO-type methods, extending true feature selection of regularized method to a higher level. This study also suggests some deeper understanding of "true predictive models" versus models that minimize the training error and so-called "predictive models" in computer science community. Uncovering the true underlying model has a direct relationship with the modeling approach, which prioritizes interpretability and generalizability over prediction accuracy and complexity.

There is still room for further investigation of the "true model form" (e.g., interaction terms). One possible way to do so is to put the iterative process as a true feature selection tool in the body of the broader model building framework, where we treat the selected features as non-droppable main effects, and then by the use of traditional regression techniques (e.g., OLS) we adjust the model form by investigating the other possible terms generated by those features.

Acknowledgments

This work was partially supported by the National Science Foundation Grant CMII-1434401.

Chapter 4

Outcome-Adaptive Iterative Elastic Net: Causal Variable Selection Under Time-varying Confounding

Abstract

Randomized control trials are not always the feasible solution for selecting and estimating the causal effect of the interventions or treatments in practice. The use of observational data, however, requires the methods to effectively account for the bias created by either treatment assignment or other complexities in the observational setting. In this paper, we focus on the sequential treatment structure known as adaptive treatment regime, where the dynamic nature of the problem causes a bias called endogeneity or time-varying confounding. While previous studies handled the one treatment case, we are focusing on the multiple treatment case where treatments are correlate and the aim of the study is to uncover the true underlying outcome model features that consistently estimates the effect of the causal treatments on the outcome of interest at each stage. We design two-stage adaptive treatment simulated cases, where treatments are binary, and covariates are the mixture of binary and normal, and the response is continuous. At each stage, observations are created by defining a linear relationship between the response and "outcome" predictors, the confounders, and the "causal" treatments. In the simulation design we take full factorial combination of the study factors: (1) the total number of variables in the model consists of treatments and covariates; (2) the proportion of causal treatment variables, true outcome predictors and confounders among all treatments and covariates, respectively; (3) the correlation structure that controls the correlation within causal treatments and the correlation between causal treatments and spurious treatments; and (4) interaction terms that create treatment-treatment, treatment-past treatment, treatment-outcome covariate and treatment-confounder terms. To uncover the true underlying outcome model, we propose the Outcome-Adaptive Iterative Elastic Net in the body of a suggested outcome-modeling framework. We justify its performance based on the computational results of the simulation.

Keywords: Causal variable selection; True Underlying Model, time-varying confounding, Penalized regression, Tuning parameter selection, LASSO, Elastic Net, Adaptive LASSO, Multicollinearity.

4.1 Introduction and motivation

During the adaptive treatment plan, the level and type of treatments are modified over time based on patient information, patient risk factors and past treatments, and patient response to the treatment [81, 82, 83, 84, 85]. What inspired this study is the two-stage interdisciplinary pain management program at the Eugene McDermott Center for Pain Management at The University of Texas Southwestern Medical Center at Dallas. To clarify the definition of the problem, we seek to take into account two important biases that are present in this study, confounding and time-varying confounding. For simplicity, the causal diagram [86, 87] in figure 15 represents the "one treatment" clinical case. To avoid clutter, the causal diagram only starts from the treatment at the stage 1 (T_1) and ends with recording the patient pain outcome (Y), at the end of stage 2. The causal path ($T_1 - Y_1 - T_2 - Y_2$) along with ($Y_1 - Y_2$) in figure 1, showing that the prescribed treatment in the stage 2 (T_2), is modified in type or dose according to the patient outcome, as a time-varying confounder at the end of the previous stage (Y_1), which itself is affected by past treatment (T_1) and a common cause of (T_2) and (Y_2). This creates a bias called time-varying confounding in the causal treatment effect on the outcome of interest [88, 89]. In figure 15, two outward edges from (X) indicates another type of bias created by the confounder (X), a variable associated with both treatment and outcome [90]. In observational studies and epidemiology, this is discussed extensively as selection bias [91]. In this study, the confounders are time-invariant during the course of treatment plan.

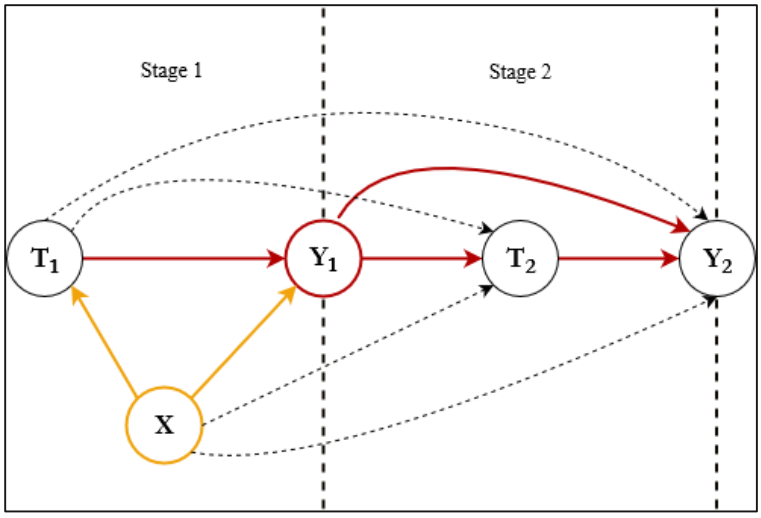


Figure 15. Time-varying confounding and the confounder, in case of one treatment

In the presence of only one treatment, to account for both aforementioned biases, many methods in use require either modeling of the outcome conditional on treatment, past outcome, and potential confounders, known as an outcome model (e.g., regression techniques) or specifying the model for treatment, known as n exposure or treatment model (e.g., propensity score matching and inverse probability of treatments weighting (IPTW) [92, 93, 94, 95, 96, 97]) or both, known as doubly robust methods [98, 99, 100, 101]. The advancements in these methods concentrate on "consistent selection" of true outcome-predictors and true confounders, the correct inclusion of interaction terms, known as "effect-measure modifiers," and finally the "consistent estimation" of the causal treatment effect by adjusting for time-varying confounding.

One important assumption in all these methods are "no unmeasured confounders," meaning that all the outcome-treatment confounders are accounted for in the inverse probability of treatment weighting (IPTW) or exposure model (i.e., propensity score) . This is the motivation for why analysts are using the domain expert knowledge to incorporate as many covariates as possible in the model to avoid serious bias caused by unmeasured potential confounders. On the other hand, while this this approach helps, recent studies have shown inclusion of spurious variables and covariates only associated with treatment in the propensity score model causes efficiency loss [102, 103, 104, 105] and variance inflation and possibly additional bias [105,

106]. Conversely, the study in [107] has shown that the inclusion of outcome covariates in the IPTW or propensity score model can potentially improve the precision and efficiency of the estimation. This suggests a variable selection technique that prioritizes uncovering the true underlying features of the outcome over prediction performance as opposed to most of the methods in practice. All the previous studies addressing the problem of variable selection for causal inference are limited to only one treatment, while we are developing a causal variable selection method for multiple treatments when high correlation is present. In addition, these studies evaluated their proposed method under only a flat correlation structure, which is not realistic. In reality, one might face a situation with spurious variables that are highly correlated with outcome covariates. We particularly take this into account and designed a correlation structure that separately controls the correlation level for (1) within outcome predictors and (2) between spurious and outcome predictors. This provide an ideal platform to analyze the performance of methods in detecting causation versus correlation. The multiple correlated treatments create these difficulties:

(1) To obtain IPTW weights instead of calculating the probability of single treatment, the joint portability of multiple treatments given covariates, past treatments, and past outcome should be obtained. We are using Ohol's approach [13] in which the novel MIMIC algorithm [108] was employed to estimate the joint probability of treatments.

(2) The diagnostics of the IPTW weights is a cumbersome task where exposed observations no longer can be separated in the multiple treatment setting. We propose a new way to calculate the weighted absolute mean difference.

In general we propose the Outcome-adaptive Iterative Elastic Net (OA-ITELNET) for discovering the true underlying model and achieving consistent estimation of causal treatments. Our approach augments the consistent selection of true outcome-predictors, true confounders and more importantly causal treatments in the outcome model. Adjusting for time varying-confounding is achieved via IPTW and a proposed IPTW diagnostic tool for the multiple treatments case. Finally, our experimental design study

investigates the feature selection performance in the presence of effect-modifiers or interactions in the outcome model.

The rest of this paper is organized as follows. Section 4.2 describes step by step phases of the proposed approach. The simulation design, performance measure, and candidate methods are given in section 4.3, 4.4, 4.5 respectively. The result and discussion is encapsulated in section 4.6. Section 4.7 represents the real case study, and section 4.8 concludes the paper with summary and future research.

4.2 Outcome-Adaptive Iterative Elastic Net (OA-ITELNET): A tool for causal variable selection

In the simulation study in Chapter 3, it is shown how the iterative process with the help of the proposed model selection technique improves the recovery of the true underlying features. We extend ITELNET to outcome-adaptive ITELNET to take into account the nature of time-varying confounding and selection bias created by confounders. The whole process of the method is depicted in figure 16 and explained in phases.

The Input: The full model

The algorithm starts with the full model and sparsify the features until it reaches the stopping criteria.

4.2.1 Phase 1. Adjusting the folds

Step 1. Preliminary feature selection

Studies have shown that the IPTW weights performs very poorly when the true model is far from the full model OLS estimates, meaning that a preliminary variable selection is required before obtaining usable weights. In the case of high correlation, this could be achieved using the Elastic Net and minimum CV as model selection metric.

Step 2. Obtain the IPTW weights

In phase 1, the IPTW weights are calculated for the multiple treatment case by the method proposed by Ohol [13] using the MIMIC algorithm [108]. The features entering this phase for estimating IPTW are to remove bias and improve efficiency and precision. For the first iteration, the algorithm starts with the input of the full model, and for the i^{th} iteration ($i > 1$), the reduced model at the end of the $(i-1)^{th}$ iteration enters phase 1.

Step 3. Specify the fold membership for each observation by *K*-Stratified sampling from *K*-quantiles (strata) of the associated IPTW weights

The philosophy behind the general use of balanced randomized assignment of the observations to folds is the balanced distribution of the folds. However, the nature of the adaptive treatment setting creates imbalance and rare cases in the assignment of treatment, and therefore, in the distribution of observations. Accordingly, small IPTW weights conceptually represent the patients with high probability of receiving the treatments given the patient's state variables and vice versa. A unified approach to causal variable selection and estimation is to extend the adjusting concept to the distribution of observations by incorporating the joint probability of treatments given past treatments, past outcome and the covariates in assigning each observation to the k^{th} fold, as follows.

(1) Sort and divide the data set into *K* quantiles or strata according to the associated IPTW weight magnitude.

(2) Conduct *K*-stratified sampling from *K*-strata in step 1 to determine the fold membership of each observation.

4.2.2 Phase 2. Feature selection

Step 1. Compute and update the adaptive weights by β^{ridge}

Instead of OLS estimates, the circumstance of the adaptive treatment setting and correlated data space suggests to employ the β^{ridge} or close to ridge estimates as suggested in the literature. The effect of utilizing either β^{ridge} or β^{WLS}(ridge) estimates are compared in 4.6.1.

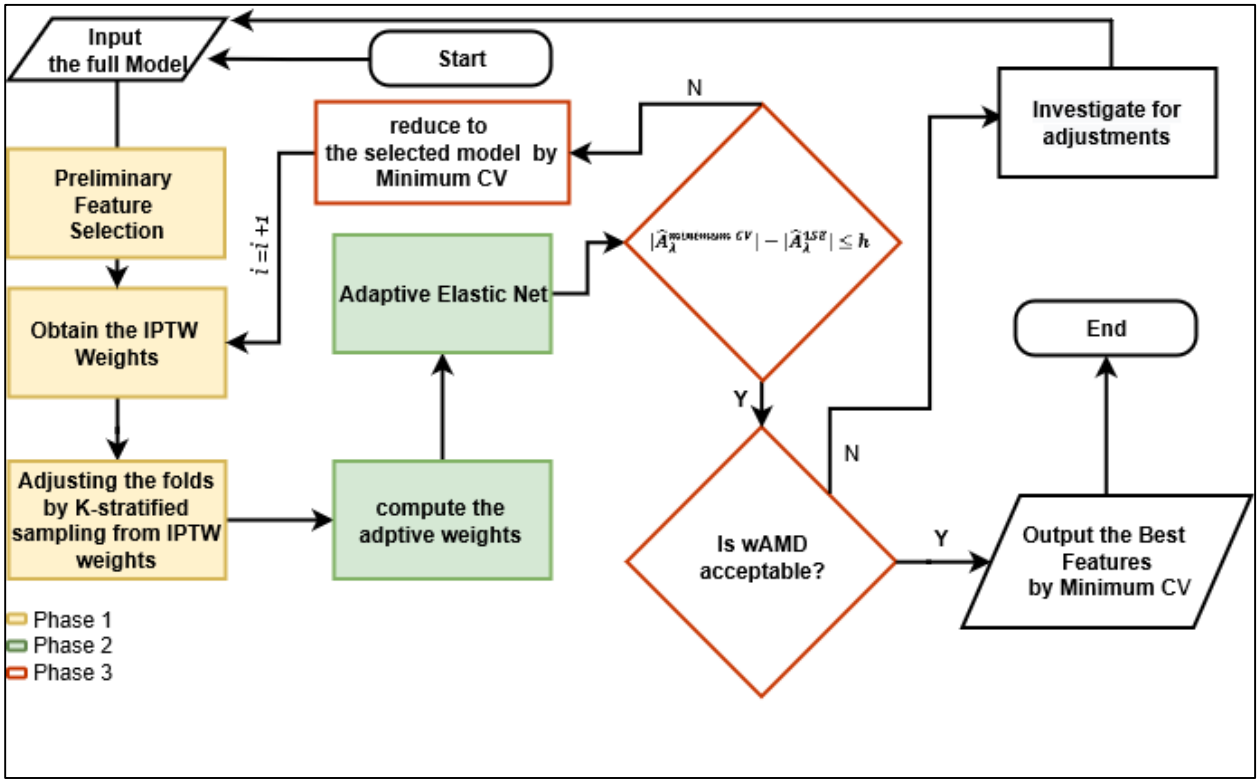


Figure 16. Outcome-Adaptive ITELNET Process

Step 2. Employ the adaptive Elastic Net and choose features by minimum CV and one standard error (1SE) using adjusted *K*-fold CV

From here on, we follow the same logical steps in the algorithm as in ITELNET, proposed in Chapter 3. The penalized function is provided in Eq. (1).

$$\min_{\beta} \left\| y - \sum_{j=1}^{p} x_j \beta_j \right\|^2 + \lambda \sum_{j=1}^{p} w_j (\alpha |\beta_j| + (1-\alpha) |\beta_j|^2) \quad (1)$$

where w_j are adaptive weights.

4.2.3 Phase 3. Check for convergence

Step 1. Check the stopping condition: Is the similarity condition $|\widehat{A}_{\lambda}^{\text{minimum CV}}| - |\widehat{A}_{\lambda}^{\text{1SE}}| \leq h$ satisfied?

It is shown in the iterative process in Chapter 3, where standards error $SE(\lambda)$ of minimum cross validation $CV(\lambda)$ is too small, the features selected are more likely to be close to the true underlying features. This is equivalent to satisfy the similarity condition. They also suggest to use h = 1 for general applications. Another reason for adjusting for the folds is that imbalance in the folds can cause misrepresentation of the similarity condition. If the similarity condition is satisfied the algorithm stops and return $\widehat{A}_{\lambda}^{\text{minimum CV}}$. Otherwise, the algorithm continuous to the next iteration $(i + 1)$.

Step 2. Diagnostic check on the magnitude of the $wAMD_i$ at $\widehat{A}_{\lambda}^{\text{minimum CV}}$

The weighted absolute mean difference (wAMD) as the diagnostic tool for IPTW weights has a direct and mutual relationship with the causal true underlying outcome model, as shown by Shortreed et al. [109] and many other studies mentioned earlier.

4.2.3.1 Weighted absolute mean difference (wAMD) for multiple treatments

Using inverse probability of treatment weighting (IPTW) for estimating the causal effect of treatment in observational studies is only valid if there are no significant differences in distribution of observed state variables between exposed (treated) and unexposed (control) in the weighted sample [110] . This is called balance diagnostics, and the goal is to create a weighted sample or a pseudo-population in which the distribution of outcome covariates and confounders (X_O, X_{OT}) is the same between exposed and control

subjects. With everything else held equal, the model with minimum difference or higher balance is more competitive in the adaptive treatment setting and therefore should be taken as a secondary performance measure. One quantitative way to measure balance is the weighted absolute mean difference (wAMD), which has only developed for the single treatment case. We extend AMD to mean (AMD) for multiple treatments in Eq. (2). This is for comparing continuous covariates and justifiably can be used for dichotomous variables [111]. For the j^{th} covariate in $j = \{1,2,\dots,d\}$, the mean absolute mean difference for V treatments is:

$$\text{mean(AMD)} = \frac{\sum_{v=1}^{V} \theta_v \zeta_j \frac{(\bar{x}_{jv}^{\text{treatment}} - \bar{x}_{jv}^{\text{control}})}{\sqrt{s_{jv}^{2\,\text{treatment}} + s_{jv}^{2\,\text{control}}}/2}}{\sum_{v=1}^{V} \theta_v} \times 100 \tag{2}$$

$$\text{where} \begin{cases} \theta_v = 0 & \text{if } v^{th} \text{ treatment is dropped from the model} \\ \zeta_j = 0 & \text{if } j^{th} \text{covariate is dropped from the model} \\ \theta_v = 1 & \text{if } v^{th} \text{ treatment remains in the model} \\ \zeta_j = 1 & \text{if } j^{th} \text{covariate remains in the model} \end{cases}$$

The mean weighted absolute mean difference is then defined by replacing $\bar{x}\ by\ \bar{x}_{\text{weight}} = \frac{\sum w_i x_i}{\sum w_i}$ and s^2 by $s^2_{\text{weight}} = \frac{\sum w_i}{(\sum w_i)^2 - \sum w_i^2} \sum w_i (x_i - \bar{x}_{\text{weight}})$, where w_i is the IPTW weight estimated and assigned to object i.

If the true features are selected, we are adjusting for the true confounders, which technically means IPTW given these confounders accounts for the confounding bias to its full potential, and this causes wAMD to have ideally zero or a very small value. Now imagine an extreme situation where all coefficients in the model are zero. This could happen in the extreme case of strong penalization, $\lambda = \infty$. In this case, the estimated propensity score would just have an intercept term. All individuals in the exposed group would receive a constant weight (i.e., constant across all values of covariates), and all individuals in the unexposed group would receive a constant weight (also constant across all values of covariates), which may be the same or different than the weights in the exposed group, depending on the marginal probability of

exposure on the sample. In this case, the IPTW weights are not accounting for any confounding, since they are constant over all values of the covariates. If the weights are not accounting for the confounding then there will be differences in the covariate distributions in the exposed and unexposed groups in the weighted sample. This would mean that the wAMD would be large. Therefore, in the case of convergence by the similarity condition in step 1, if the magnitude of the wAMD is too high, then we should consider making adjustments to our model and investigate for the reason.

4.3 Experimental setting

4.3.1 Simulation design

Inspired by the adaptive treatment setting at the Eugene McDermott Center, the linear association in the simulated case studies are created, as shown by arrows in figure 17 and described in the following logic. Here, the first goal of modeling is to capture the closest to true causal features. Let X_O denote covariates associated with outcome, X_{OT} predictors of both as confounders, X_T the predictors of exposure (treatment), and finally X_S nuisance or spurious covariates. All the covariates are correlated with a designated correlation matrix and generated by a modified version of the BinNor package in R. Later on, we describe the correlation structure in detail. The probability of each binary treatment is calculated by linear association with some selection of $X_{OT}, X_T,$ the past treatments and the previous outcome by Eq. (3)

$$P_{\mathrm{trt}_t} = \frac{\exp(\beta^T X_T + \beta_{trt}^{OT} X_{OT} + \eta_{t-1} trt_{t-1} + \gamma_{t-1} Y_{t-1})}{1 + \exp(\beta^T X_T + \beta^{OT} X_{OT} + \eta_{t-1} trt_{t-1} + \gamma_{t-1} Y_{t-1})} \quad (3)$$

Then by predefined correlation matrix and approximate multivariate normal, correlated binary treatments are created. The algorithm divides the data in two groups: effective and non-effective treatments.

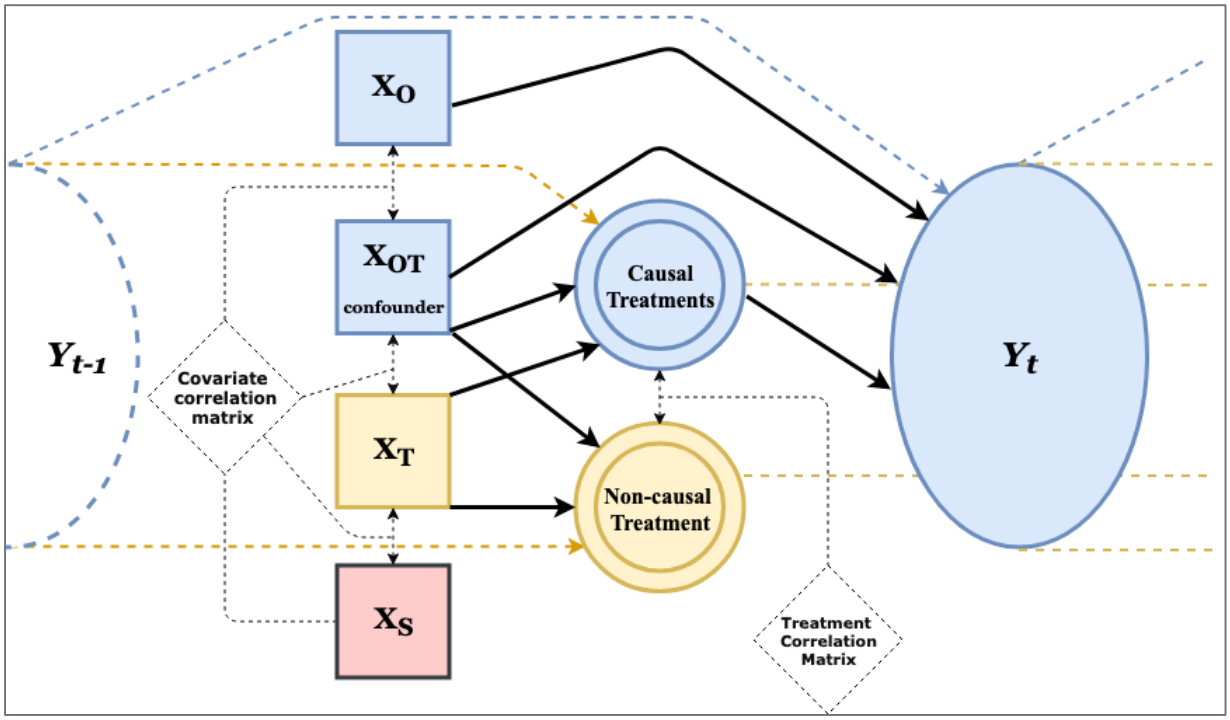


Figure 17. Process map and causal diagram for creation of simulation cases.

Finally, the outcome at the end of stage t is created by linear dependence on the effective treatment, X_O, X_{OT} and the previous outcome and some interaction terms in Eq. (4).

$$Y_t = \beta^O X_O + \beta_Y^{OT} X_{OT} + \eta_t\, trt_t + \eta_{t-1}\, trt_{t-1} + \gamma_{t-1} Y_{t-1} + \text{interaction terms} \quad (4)$$

$$\text{Where } \varepsilon_i \sim N(0,1)$$

We used R programming to generate the simulation, and the codes are available at:

https://github.com/ashkanfa/Generate.adaptive.data.

The cases have been designed considering five study factors.

(1) total number of covariates and treatments in the model. To be realistic, in such studies in collecting data too many potential confounders are recorded to avoid unmeasured confounders. Hence, even after cleaning the data, the total number of covariates are more than pool of available treatments, same pattern in Eugene McDermott Center for pain management. Inspired from that, these combinations are considered for covariates and treatment respectively, {16,8}, {20,8}, and{16,12}.

(2) proportion of causal treatment among all treatments, {0.5, 0.75}

(3) proportion of outcome covariates and confounders to the total number of covariates. {0.25,0.5}. Mimicking real situation, the proportion of treatments are higher than proportion of covariates.

(4) the correlation structure shown in figure 18 gives the researcher the leverage to effectively study the performance of the methods in distinguishing the causation from correlation, especially when the correlation between causal and spurious variables are high. For each block of correlation we consider three magnitude of correlation; low [0, 0.2], medium [0.3, 0.6], and high [0.7, 0.9]. We did not however consider separate structure for treatments and covariates since cross-production of separate consideration would yield 81 combination. In addition, since the within spurious correlation structure is not the focus in this paper, it is set to a constant range from low to high.(5) Interaction structure of (a) no interaction and simultaneous combination of (b) treatment-treatment, (c) treatment-past treatment, (d) treatment-outcome covariate and (e) treatment-confounder are generated to get a more realistic estimate of the real model and provide the opportunity for us to compare the methodology in uncovering the true features under more complex model form. There is no simultaneous treatment-outcome covariate and treatment-confounder interaction terms in the simulation setting. The reason is that we are to investigate how the nature of the interaction terms will affect the causal variable selection.

Magnitude of coefficient and number of observations are set to [0.6,1] and 100 respectively due to the fact that in the previous study in the Chapter 3, the effect of these factors in causal variable selection has been studied and they are far from the motivation of this paper. Respectively from (1) to (5), we have full factorial combination of 3*2*2*9*5 equals to 540 cases. 100 replications of each case will lead to comparative performance analysis of 54,000 data set. The methods however applied to fractional factorial when compiling the result on a factor of interest.

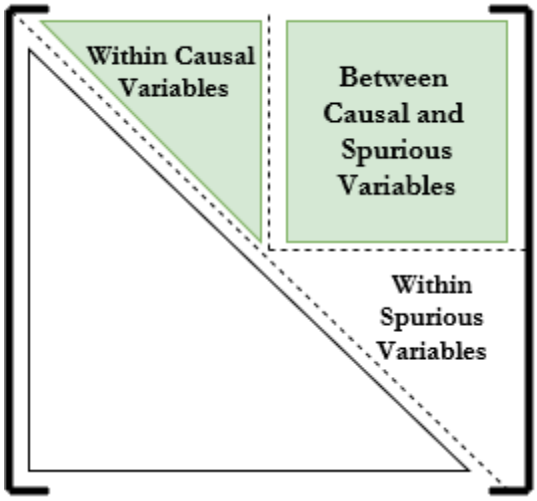


Figure 18. Design of the Correlation Structure for simulated cases: the shaded are controlled structure

4.4 Performance measure

4.4.1 Confusion matrix

For evaluation of the overall causal variable selection performance, we adopt the confusion matrix in table 3 from the information retrieval community [78, 79, 80].

Table 3. Confusion Matrix

Confusion Matrix		**Predicted Model**	
		Spurious	**Causal**
True Model	**Spurious**	a	b
	Causal	c	d

Geometric mean of sensitivity and specificity suggested by Kubat et al. [78] is a robust measure to imbalanced proportion of true and spurious predictors and used in this paper.

$$\text{Sensitivity} = d / (c + d) \tag{5}$$

$$\text{Specificity} = a / (a + b) \tag{6}$$

$$\text{g-mean} = \sqrt[2]{\text{Sensitivity} * \text{Specificity}} \tag{7}$$

If g-mean turns out to a value close to 1, it implies that most of the variables classified correctly. In this study we investigate the result of g-mean for treatments and covariates separately.

4.5 Candidate methods for causal variable selection

In addition to extensive literature on causal variable selection for one treatment, there is almost no clear guideline available for the multiple treatment case. Hence in the effort of collecting competitive methods, some the current methods in the literature was adjusted in order to be qualified for comparison.(1) Outcome-adaptive LASSO is first introduced by Shortreed [109], in the one treatment setting, with competitive performance in low to medium correlation magnitude [0.2,0.5]. The use of wAMD as a model selection is proposed. We extend the OAL to outcome-adaptive Elastic Net and generalize their suggested wAMD to Eq.(8) to adjust for the presence of high correlation and multiple treatments setting respectively.

$$\text{wAMD}(\lambda_n) = \sum_{v=1}^{V} |\eta_v| \sum_{j=1}^{d} |\beta_j| \frac{(\overline{x^w}_{jv}^{\,\text{treatment}} - \overline{x^w}_{jv}^{\,\text{control}})}{\sqrt{{s^w}^2_{\text{pooled}}}} \tag{8}$$

Where η_k and β_j are the coefficients of the treatment k and covariate j accordingly, and $\bar{x}^w$ and s^{w2} are weighted mean and weighted variance. (2) We also include the normal adaptive Elastic Net with 1SE rule as base model selection technique. (3) Before we enter our proposed model for comparative performance study, the performance of some of the adjusted and proposed elements in the outcome-adaptive ITELNET al.gorithm is evaluated (a) first we check if the IPTW adjusted folds in phase 1 makes a difference (b) Although it was not considered in our study, we also compare what if scenarios of the adaptive $\beta^{WLS(ridge)}$ against β^{ridge} in computing the adaptive weights in the first step of phase 2.

4.6 Simulation results

4.6.1 Investigation within Outcome-Adaptive ITELNET

First we investigate the effect of two elements within the outcome-adaptive Elastic Net; (1) IPTW adjusted folds. (2) The possible effect of using $\beta^{\mathrm{WLS(ridge)}}$ vs. β^{ridge} in forming the adaptive weights. In all combinations we separate the result for covariates and treatments. Figure 19 to 20 compares the g-mean performance of the OA-ITELNET where folds are adjusted by IPTW weights against the usual randomized *K*-fold assignment from a uniform distribution. The plots are consists of 108 no-interaction cases. Simulation results suggests slight improvement in feature selection where folds are adjusted by IPTW weights (green solid line vs. the red dashed line).

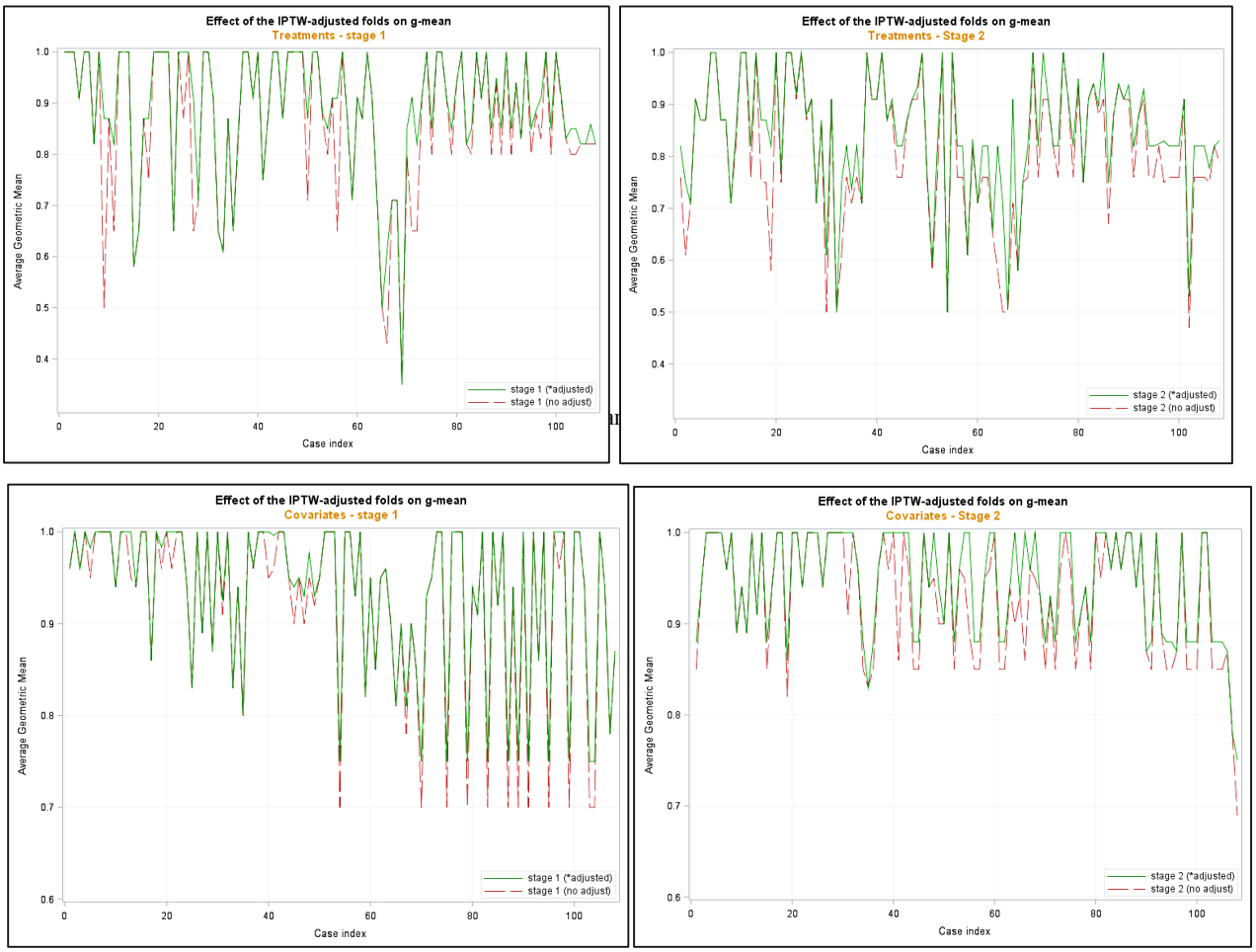


Figure 20. The effect of IPTW adjusted folds on average g-mean of covariate selection by stages

Figure 21 on the other hand is an indication of no difference if IPTW WLS estimates are used for the adaptive weights (The green line over writes the red dashed). The justification of the phases let us to continue to the comparative result in the next section.

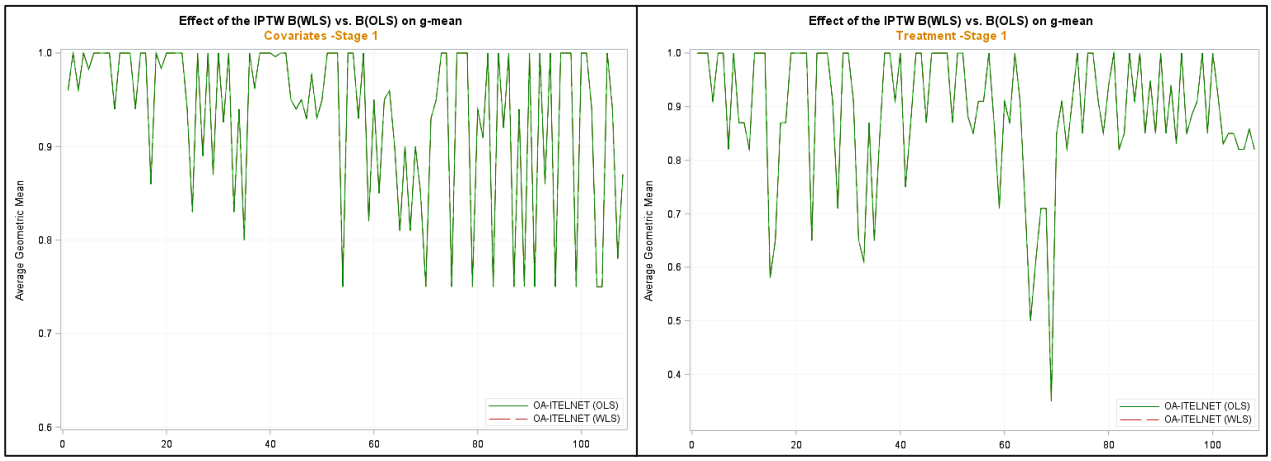


Figure 21. The effect of IPTW WLS adaptive weights on average g-mean

4.6.2 Comparative result on g-mean in cases with no interaction

There are 108 cases with no interaction, each 100 replications, all with 100 number of observations. Figure 22-24, represents the average g-mean in no interaction cases, from the proportion perspective and correlation structure. The associated average g-mean is also summarized in a table attached at the bottom of each figure. In all the plots the proposed method (OA-ITELNET, green solid line) has the best feature selection performance regardless of the factor of interest. This pattern follows by ELASTIC NET(ELNET, red solid line), and at the bottom comes Outcome-adaptive LASSO (OA-ELNET). In addition, it is observed that stage 2 follows the same performance pattern with irregularities compared to the first stage. This could happen due to the escalated complex association among the variables, the effect of confounder and time-varying confounding.

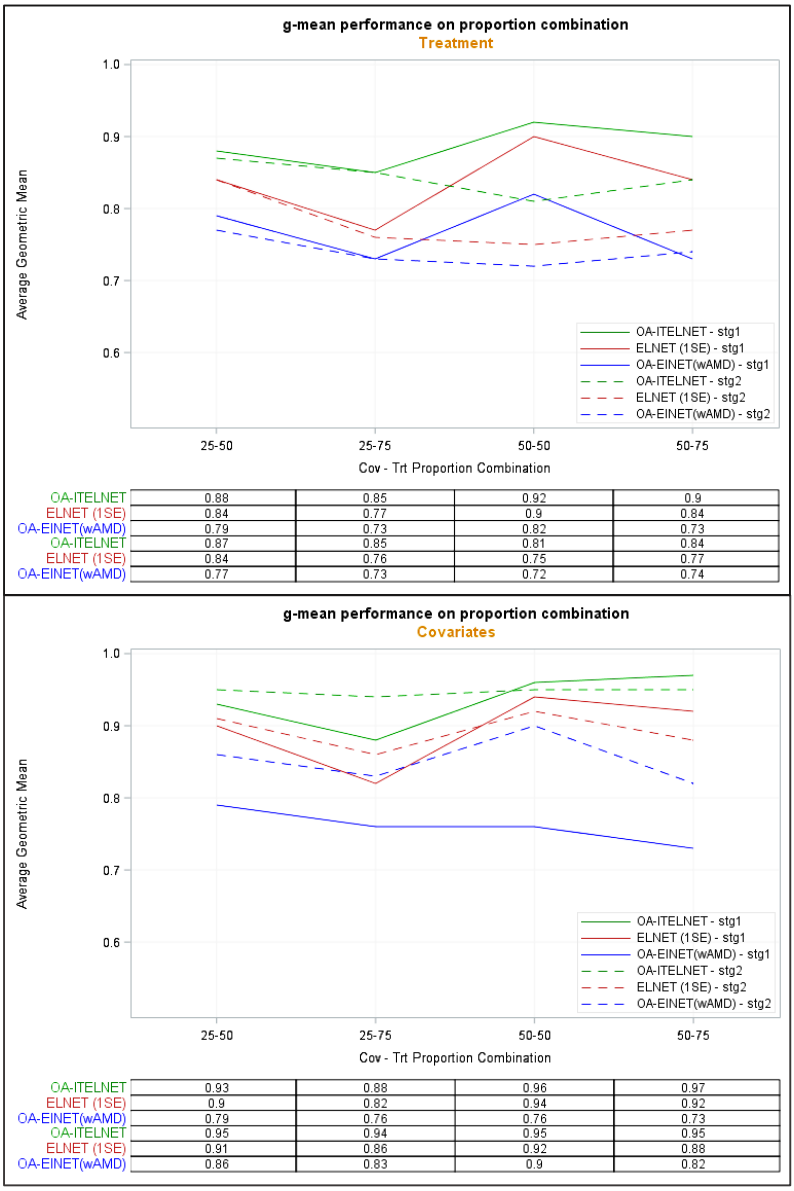


Figure 22. Average g-mean performance based on true proportion combination of covariate-treatment (e.g., 25-50 means proportion of true covariates is 0.25 and proportion of true treatments is 0.50).

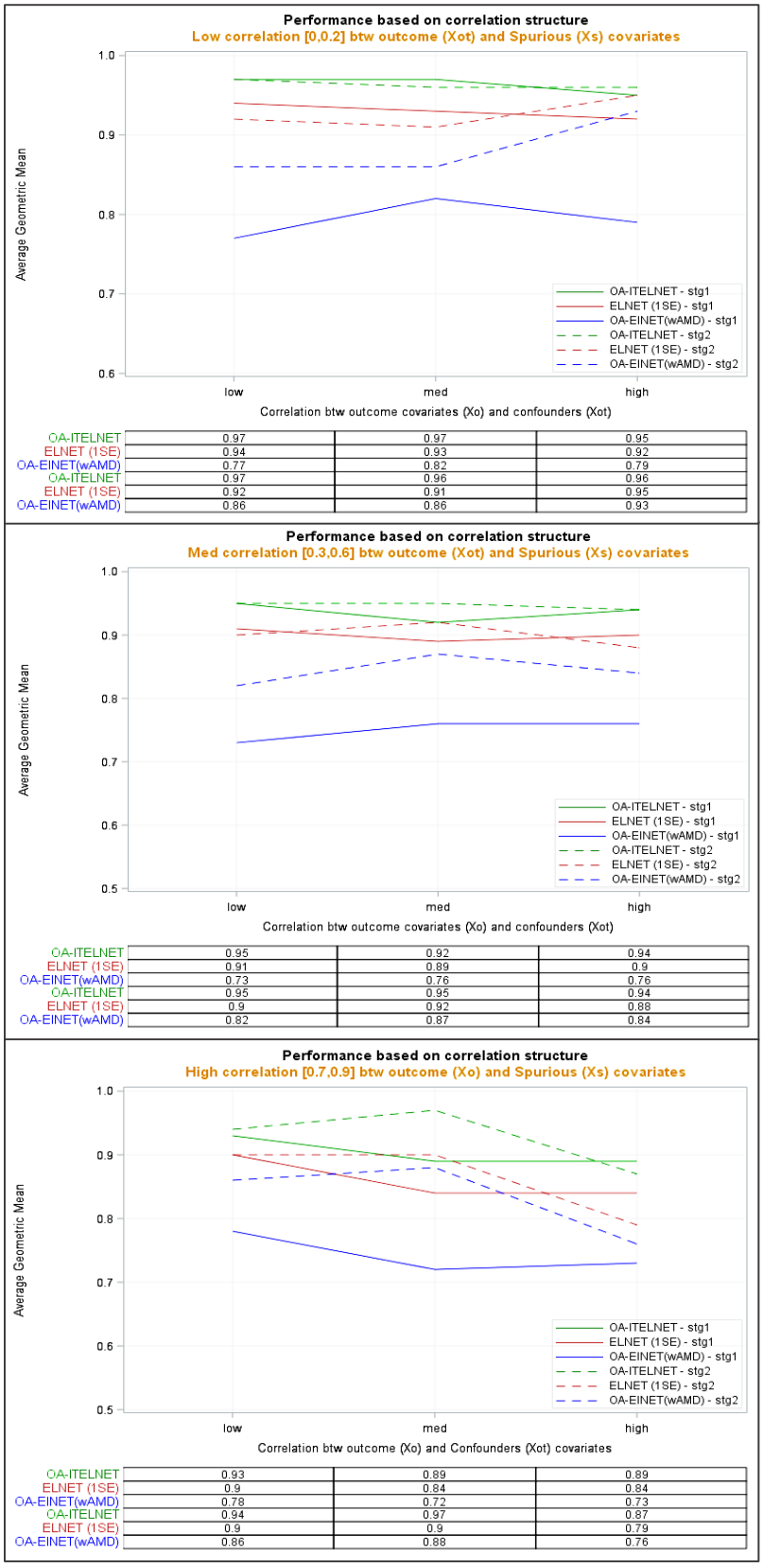
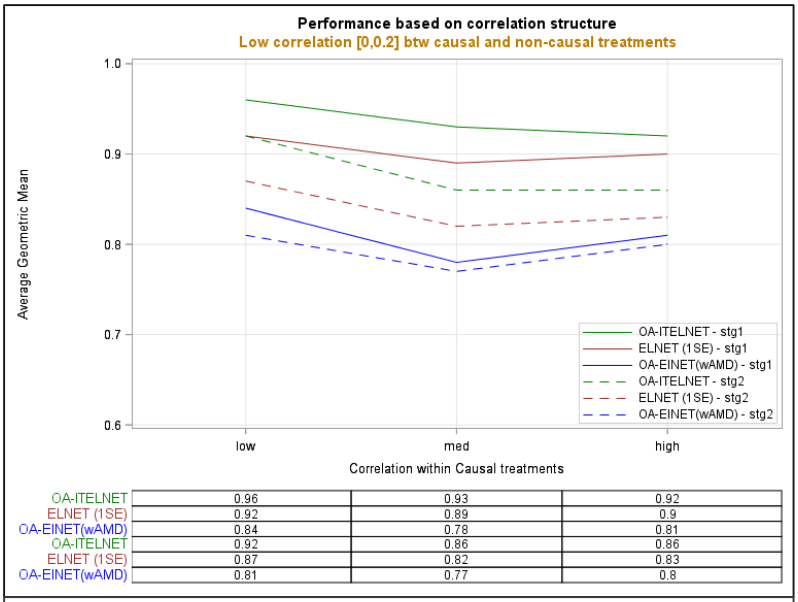
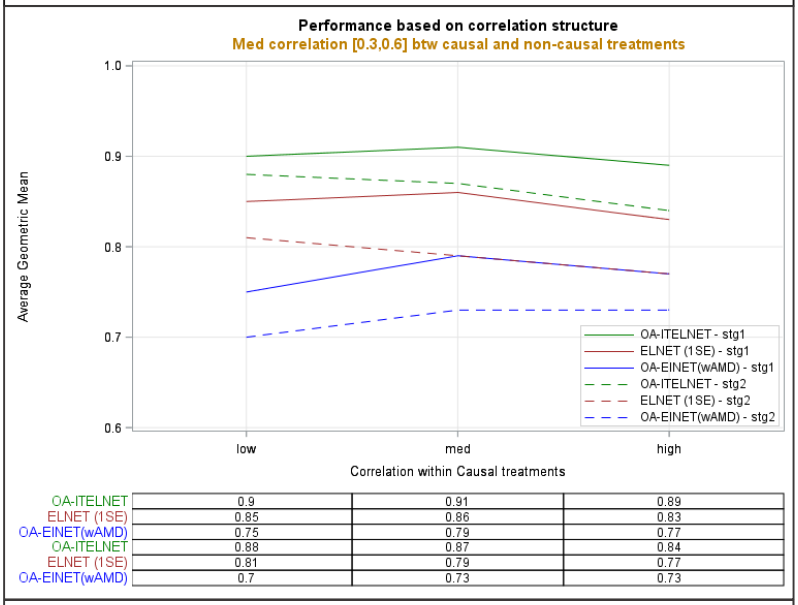


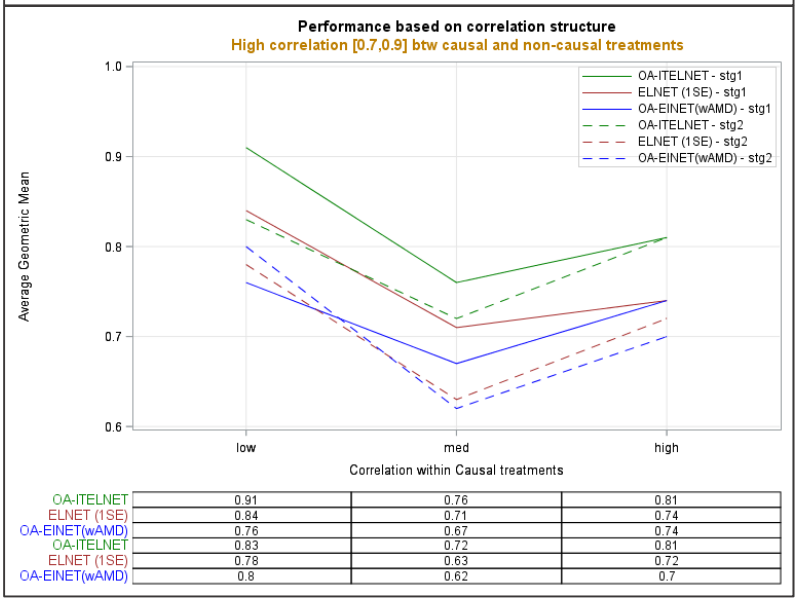
Figure 23. Average g-mean performance based on covariate correlation structure



OA-ITELNET	0.96	0.93	0.92
ELNET (1SE)	0.92	0.89	0.9
OA-EINET(wAMD)	0.84	0.78	0.81
OA-ITELNET	0.92	0.86	0.86
ELNET (1SE)	0.87	0.82	0.83
OA-EINET(wAMD)	0.81	0.77	0.8



OA-ITELNET	0.9	0.91	0.89
ELNET (1SE)	0.85	0.86	0.83
OA-EINET(wAMD)	0.75	0.79	0.77
OA-ITELNET	0.88	0.87	0.84
ELNET (1SE)	0.81	0.79	0.77
OA-EINET(wAMD)	0.7	0.73	0.73



OA-ITELNET	0.91	0.76	0.81
ELNET (1SE)	0.84	0.71	0.74
OA-EINET(wAMD)	0.76	0.67	0.74
OA-ITELNET	0.83	0.72	0.81
ELNET (1SE)	0.78	0.63	0.72
OA-EINET(wAMD)	0.8	0.62	0.7

Figure 24. Average g-mean performance based on treatment correlation structure

4.6.3 Comparative result on g-mean in cases with interaction

Discovering how sensitive the feature selection methods react to the true model form is very crucial since the true model, especially in clinical research more likely to be more complex and possibly have interaction terms or effect-modifiers. At this stage we are not able to detect the interaction terms but examine how the selection of the same features are affected under a true model with interaction terms. In figure 25, the x-axis from left to right represents "no interaction,"" treatment-treatment and treatment -$\boldsymbol{X_O}$ interaction" and finally "treatment-treatment and treatment-$\boldsymbol{X_{OT}}$ interaction." In the second stage (dashed line in figure 25) the same scenario applies except adding another interaction term of treatment-past treatment. These interaction terms are designed to account for common true model form complexities in adaptive treatments setting [112]. It is interesting how the performance pattern takes the opposite role for Elastic Net (1SE) compared to Elastic Net (wAMD) as we move from no interaction case to cases with interaction terms. If someone was to make a decision solely based on the additive setting in the previous section, the performance of the Elastic Net with wAMD dominated all the time. Although the proposed method (OA-ITENET), distinguishes itself with a significant distance, if we were to set the h parameter to 1, similar to chapter 3, the performance was dropped. This is where we notice the sensitivity of the h parameter in similarity condition to the true model form.

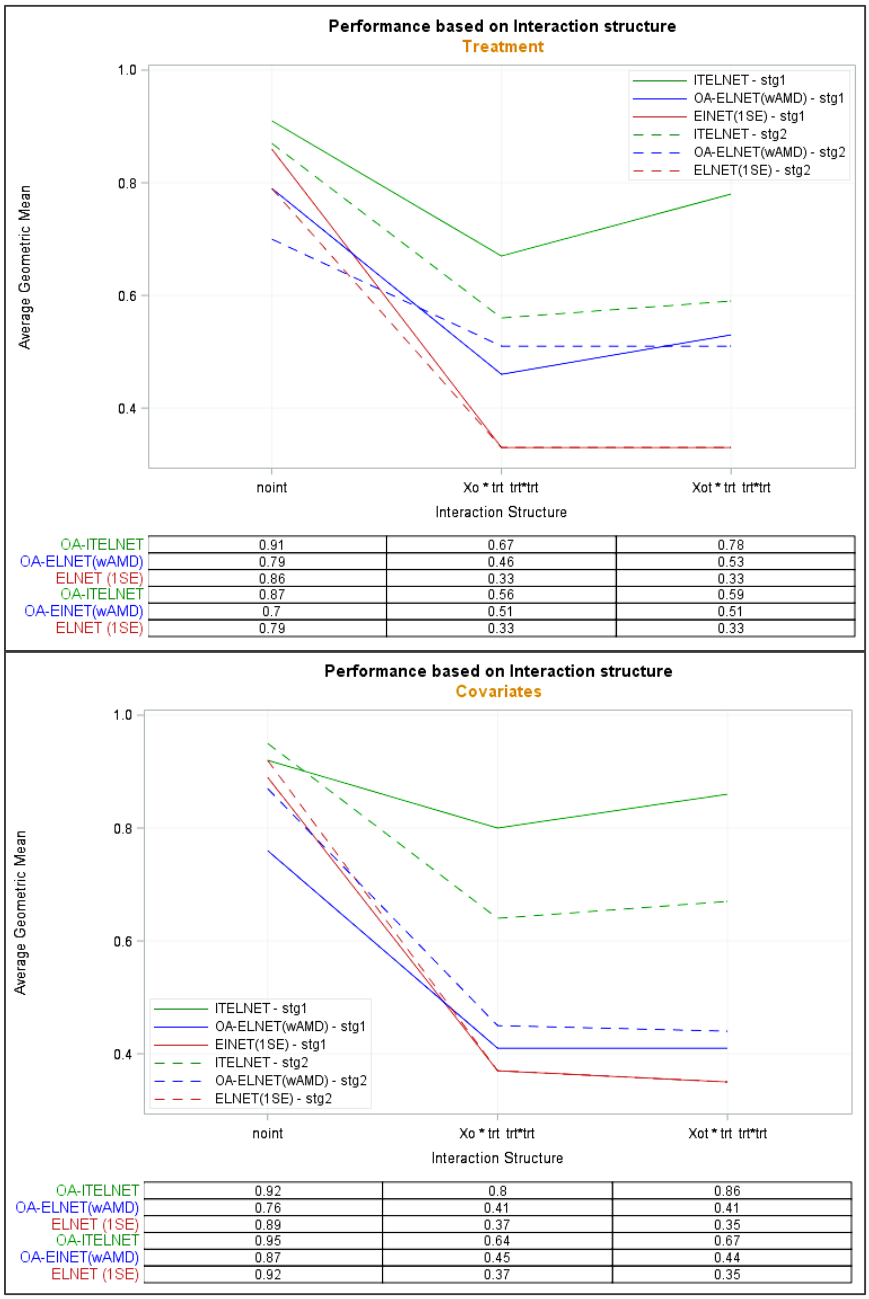


Figure 25. Average g-mean performance based on interaction structure

4.7 Pain management case study

This section discusses the application of the outcome-adaptive ITELNET to a case study from Eugene McDermott Center of Pain at University of Texas Southwestern in Dallas. After cleaning the data, the final

set consists of 235 observations with 25 patient variables including mix of binary and continuous variables as age, gender, litigation, marital status, race, physical history, past diagnosis, surgical history, and etc [85, 84]. Table 2 is referring to the 14 binary treatments in the study which are divided in two groups, 8 pharmaceutical and 6 procedural treatments. Some of the treatments and patient variables are grouped to handle data separation issue which is extensively discussed in [13, 85, 113].Each patient should receive at least one treatment during the course of the study which can be prescribed from either one or both groups. The associated pain outcome is the continuous variable, Oswestry Pain Disability measure (OSW) [114] which is observed and recorded in a "pure observational setting" at the pre-, mid- and post-evaluation. The previous research on this application concerned adjusting for time-varying confounding in the multiple treatment case [85, 13, 84], outcome modeling and finally developing a two-stage stochastic dynamic programming that incorporates the estimated outcome model to find the most effective treatments [113].

A model entering the optimization must be close to the true model that describes the true underlying relationship between outcome, confounders, and treatments. This a modeling approach that captures the true underlying causal model with the highest probability in the observational setting. The MIMIC approach by [13, 108] is used for computation of IPTW weights and the proposed outcome-adaptive ITELNET is applied to the pain management data; the selected features and the associated wAMD is summarized in table 5.

Table 4. List of treatments in the pain management case study

Code	Procedural Treatments	Code	Pharmaceutical Treatments
ProcGr1	Injections	RxGr1	Tramadol
ProcGr2	Block Procedures	RxGr2	NSAIDs
ProcGr4	Stimulation Procedure	RxGr3	Narcotic
ProcGr9	Cognitive Therapy	RxGr4	Muscle Relaxant
ProcGr10	Physical Therapy	RxGr5	Antidepressant
ProcGr11	Number of Additional Procedures	RxGr6	Tranquilizer
		RxGr7	Sleeping Pills
		RxGr8	Others

Table 5. Feature selected by OA-ITELNET and associated wAMD

Variables	Stage 1	Stage 2
Confounders	Gender Race Phydx33 ProcGR12_0 Pastdx6 Marital_2 Children Pre_OSW	RxGr6_1 Race phydx34 pastdx7 pastdx33 marital_3 mid_OSW
Treatments	ProcGr12_1 RXGr5_1 RxGr9_1 RxGr10_1	RxGr2_2 RxGr6_2 RxGr10_2

Figure 26 represents the result of the balance measured by absolute mean difference, in weighted samples (wAMD) using reduced features via OA-ITELNET (green solid line), compared to the absolute mean difference (AMD – black solid line) in unweighted sample. We also included the wAMD achieved by the extreme case of the full features (red solid line) to have an idea what would happen if the importance of feature selection in obtaining the IPTW was ignored. Figure 26 suggests that the features selected by OA- ITELNET reasonably reduces the imbalance in samples to a great degree compared to full model, except for Phdyx33 in stage 1 and marital_3 in stage 2. For these two cases, one should use the stabilized IPTW weights versus the unstabilized version which we applied here and see if the odd result persists. In that case we should re-investigate the process and get feedbacks from a domain expert regarding the selected state variables and treatments.

Figure 26. IPTW balance diagnostics for important covariates in pain management case.

4.8 Summary and future research

Compared to other LASSO type feature selection techniques, the Outcome-adaptive ITELNET (OA-ITELNET) has made a significant improvement in causal feature selection under time-varying confounding. The superiority of the proposed method become more significant where different complicated correlation structure and different true model forms are tested, in particular where some of these structures are mimicking the worst real-world scenarios. The promising results under these conditions, opens up a new window for family of modern feature selection techniques in which the iterative process improves the contradictory task of uncovering the causal model in the pure observational setting to a great extent. We observed in 4.6.3 that the feature selection performance is insensitive to the true model form if we choose the right value for the h parameter. This means as a future direction, OA-ITELNET can be embedded in a larger perspective of uncovering the "causal model" by first uncovering the features and then building the model form based on selected features. In order to accomplish that, the first step is to establish a pre-phase of tuning for h parameter in the iterative process.

Chapter 5

Conclusion and Future Research

In this dissertation, inspired by the pain management case at Eugene McDermott Center of Pain at University of Texas Southwestern in Dallas, we endeavor to develop a feature selection technique that targets true underlying features as opposed to prediction purpose. We first simplified our case and removed the interventions (treatments) from the first paper, and proposed the ITELNET for true underlying feature selection under different scenarios in particular we targeted the dominant complexity of multicolinearity in the observational setting. The proposed method has shown some significant improvement in correct classification of true features from just highly correlated features. Then in the second paper we extend ITELNET to Outcome-adaptive ITELNET to handle the interventions (treatments) in the observational setting accounting for time-varying confounding and selection bias. We updated our simulated cases to the observational setting with interventions under a more complex setting and checked the sensitivity of the method to the true model form by adding interactions. OA-ITELNET has provided promising result for causal variable selection compared to other penalized method in practice.

However, like every method, the flip side of the good model recovery performance is the high computational time due to its iterative nature. If it takes the algorithm for instance 18 iterations to converge, it means 18 times convergence time of any usual regularized method. To mitigate this, one might suggest using the similarity condition in the single iteration searching the whole path of lambda for the values that satisfies the similarity condition. Although it is free for future direction, we took some precaution and diligence in the iterative design. At each step by the use of minimum CV we assure that the true model is the subset of selected features with high probability. And from one step to another we take cautiously tiny step to not ignoring the optima point if it can be obtained by similarity condition and the iterative process. From the optimization perspective, taking a greedy step might exhaust the performance in climbing the hill. Another direction worth of mentioning is investigating the effect of different tuning parameter $\boldsymbol{\alpha}$, different number of folds, different increment values for lambda, finally and more importantly tuning the h

parameter in similarity condition. One can think of it as another tuning parameter. Establishing a pre-phase for the iterative process, tuning techniques for h seems an interesting future direction.

The challenge is how one can measure the best value of the h parameter. One suggestion is using a secondary model selection metrics (e.g., BIC), keeping the rest of parameters in the base penalized function at a constant value, and obtain h parameter that optimizes (e.g., minimize) the selected metrics. The secondary metrics preferably should align with the purpose of this study which is the causal model recovery rather than prediction. Finally, one might examine the effect of non-parsimony against treatments on the causal feature selection. The reason is that researchers are often time interested to see the effect of the controls or treatments in the system rather than removing the unimportant ones.

Remember that we assumed full observability at the beginning (2.4.1) and built our entire research upon structure learning only. It is a very interesting future direction to change the perspective and instead, assume partial observability in the SSM, and find the similarities and differences in both approaches.

Appendix A

Hidden Markov Models (HMMs)

Here, we give a compact description of Hidden Markov Models (HMMs). HMMs define probability distributions over sequences of observations given a discrete hidden state S_t.

A.1 Representation

S_t here denote the *hidden discrete state* with K passible states. O_t , as always the observation.

1- ***Initial state*** $\pi(i) = P(S_1 = i). \pi(.)\ is\ multinomila\ distribution$

2- ***The transition model*** $\mathrm{A}(i,j) = P(S_t = j|S_{t-1} = i)$ A is a stochastic matrix (each row sums to one), a conditional multinomial distribution

3- ***The observation model*** $P(O_t|S_t)$

If the observations are discrete, we can represent the observation model as a matrix, $B(i,K) = P(O_t = K|S_t = i)$

If the observations are continuous in $\mathbb{R}_L$, it is very common to represent $P(O_t|S_t)$ as Gaussian: $P(O_t = o|S_t = i) = \mathrm{N}(O; \mu_i, \Sigma_i)$ or a mixture of Gaussian [14].

Appendix B

Kalman Filter Models (KFMs) or Linear Dynamical Systems (LDS)

Here, we give a compact description of Kalman Filter Models (KFMS) known as Linear Dynamic Systems (LDSs). These models have been initially developed and heavily in use in the control theory community [30] [115] [116] [28].

B.1 Representation

The basic model here is the discrete time linear dynamical systems with Gaussian noise. Despite that in LDS we assume that state of the system under study is continuous, it can be described by K-vector of state variables or "causes" S which we cannot observe directly, at each stage an output p-vector O is observed and accessible.

1- $S_t \in \mathbb{R}_{N_S}$, $O_t \in \mathbb{R}_{N_o}, U_t \in \mathbb{R}_{N_u}$

Initial state: $S_1 = \mathrm{N}(\mu_1, Q_1)$

Transition and observation function are Linear-Gaussian.

2- State Transition Function

$S_t = AS_{t-1} + BU_t + V_o \quad Where \quad V_o \sim \mathrm{N}(\mu_S, Q)$

Or in other words,

$P(S_t = s_t | S_{t-1} = s_{t-1}, U_t = u) = \mathrm{N}(s_t; As_{t-1} + Bu + \mu_s, Q)$

A is a $N_s \times N_s$ matrix, B is a $N_s \times N_u$ matrix, Q is a $N_s \times N_s$ positive semi-definite (PSD) matrix called the process noise. Since the Q noise is Gaussian and its dynamics are linear, S_t is a first-order Gauss-Markov Random Process. In another word, the state S is assumed to evolve according to simple first-order Markov dynamics.

3- Observation Function

$O_t = CS_t + DU_t + W_o \quad Where \quad W_o \sim \mathrm{N}(\mu_O, R)$

Or in other words,

$P(O_t = o|S_t = s, U_t = u) = \mathrm{N}(o; Cs + Du + \mu_o, R)$

C is a $N_O \times N_s$ matrix, D is a $N_o \times N_u$ matrix

Each output vector O is generated from the current state by a simple linear function plus a Gaussian noise.

4- W_o *and* V_o *are independent,* $W_\mathrm{o} \perp V_\mathrm{o}$.

5- W_o *and* V_o *are assumed to be temporally white or no serial correlation*

$\forall\, t \neq t',\ W_\mathrm{o}^t \perp W_\mathrm{o}^{t'}$ *and* $V_\mathrm{o}^t \perp V_\mathrm{o}^{t'}$

6- Time-invariant (stationary) Parameters.

7- Transition and observation function are Linear-Gaussian.

8- Without loss of generality, we can assume μ_S and μ_O are 0, since we can always augment S_t with the constant 1, and add μ_S (μ_O) to the first column of A (C) respectively.

9- We can assume Q or R is **diagonal**; see [25] for details.

10- Note that the noise is noted as V_o and W_o instead of V_t and W_t, to show that noise process does not have the knowledge of the time index [25].

11- Note that the noises are the key elements of the model: without the process noise V_o, the state S_t would always either shrink exponentially to zero or blow up exponentially in the direction of the leading eigenvector of A; same for the absence of the observation noise W_o, the state would no longer be hidden [25].

Appendix C

Dynamic Bayesian Networks (DBNs)

Now we are concerned with the representation of a class of models called dynamic Bayesian networks (DBNs), of which HMMs and KFMs are just special cases. By using DBNs, we are able to represent, and hence learn, much more complex models of sequential data, which hopefully are closer to "reality." The price to be paid is increased algorithmic and computational complexity. One thing to consider here is that DBNs have their biggest payoff in a discrete setting: combining multiple continuous variables together results in a polynomial increase in complexity, but combining multiple discrete variables results in an exponential increase in complexity, since the new "mega" state space is the cross product of the individual variables' state spaces; DBNs help ameliorate this combinatorial explosion. This the reason why DBNs mostly used for discrete or mixed discrete-continuous states [14]. DBNs are represented by Bayesian net or DAG (Directed Acyclic Graph).

In a directed acyclic graph (DAG) *G or in Bayesian Network*, nodes are the random variables in our domain and edges correspond, intuitively, to direct influence of one node on another. In another word, an arc from A to B can be informally interpreted as indicating that A "causes" B. Sometimes DAGs are called "Causal Diagrams" in other domains such as epidemiology [58] and that helps to understand the problem and possible causal relationship regardless of whether Bayesian inferences or frequentist inferences are used. For more information on Bayesian Network or tutorial on DAGs and graphical representation of HMMs, KFMs and switching state-space models refer to [1, 14] [6] [38] [117] [58]. There is no reason for us to duplicate the work of Murphy and Kohler here; therefore for comprehensive representation, inference and learning of State-space models as DBNs and their algorithms go to [14] and [1] respectively.

Appendix D

Partially Observed State

Now, in this case, assumes that initial state and state transition representation remains the same as 2.4.1 except the observation model that has the noise W_t.

This assumes that what we observe Y_t^O is dependent on a hidden state variable Y_t. The DAG of the problem will change and shown in figure 27.

3- Observation Model: $Y_t^O = (1 \quad 0 \quad 0)\begin{pmatrix} Y_t \\ U_t \\ X_t \end{pmatrix} + W_t \quad where\ W_o \sim \mathrm{N}(0, R)$

Hence, in the case of hidden state, we should do state estimation or inferences along with parameter and structure learning.

Unlike what assumed, we have a mixture of continuous and discrete state variables known as SSM with switching regime. Therefore, neither pure KFMs nor HMMs algorithms can handle the modeling. Hence, let's expand the problem and relax the assumption of joint Gaussian for the state variables and observation.

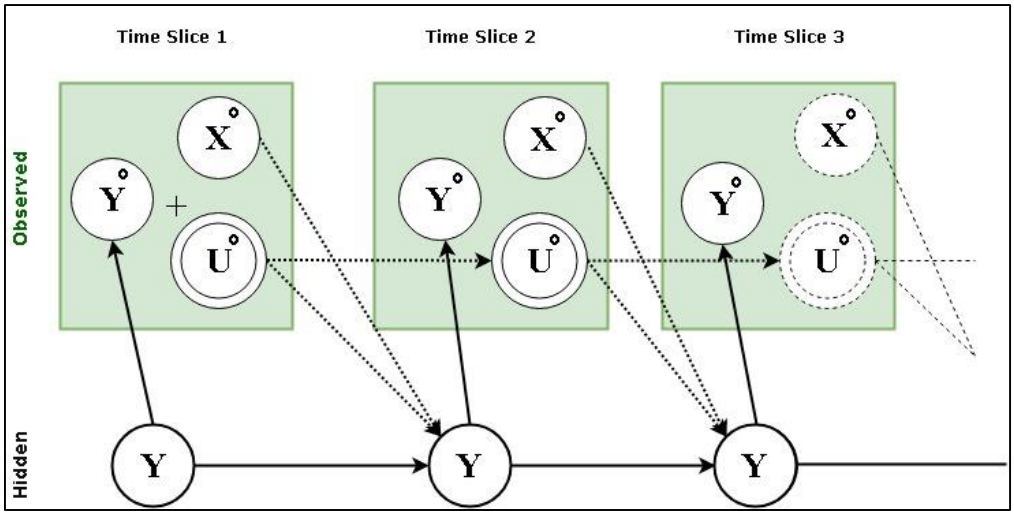


Figure 27. DAG of the problem with the assumption of a hidden variable: Partially Observed

We know That U includes multiple binary decisions or treatments. So if we add treatments as "switch" variables, The DAG is shown in figure 28.

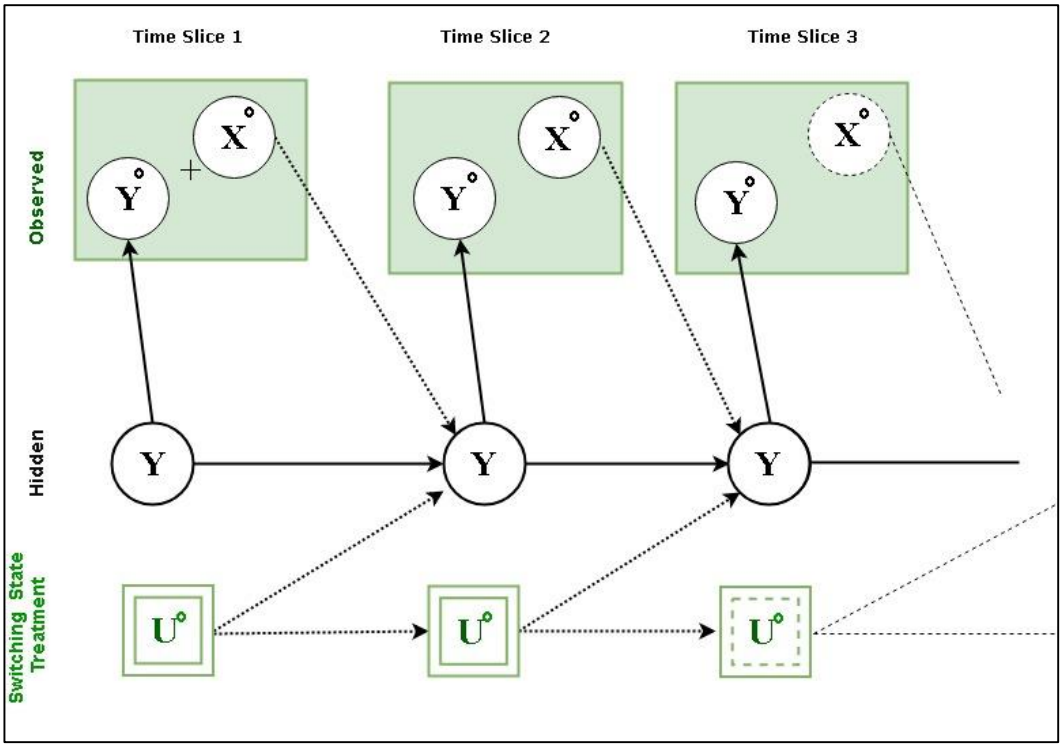


Figure 28. DAG of the problem - partially observed with control as a switch variable. square nodes are discrete, round nodes are continuous

Switching state-space models are useful for modeling "modes" of behavior. If binary treatments or interventions in our study is effective then they might switch the linear transition behavior of the hidden Y from one to another. We also know that treatments depend on previous treatments in this study. Therefore the model described in Figure 28 is a closer representation of the problem. Now let's define the transition and observation functions again.

1- Transition models:

$$P(Y_t = y_t | Y_{t-1} = y_{t-1}, X^0 = x, U_t^0 = i,) = \mathrm{N}(y_t; A_i y_{t-1} + B_i x + \mu_s, Q_i)$$

$$P(U_t^0 = j | U_{t-1}^0 = i) = A(i,j)$$

2- Observation model

$$P\left({Y_t}^0 = {y_t}^0 | Y_t = y_t\right) = \mathrm{N}({y_t}^0; C y_t, R)$$

Note that U could also be hidden if we assume it's not also affected by previous U but another hidden state. One can redefine the state space equations and it's DAG by different assumption and doing so is entirely depends on the researcher and reality of the situation.

References

[1] D. Koller and N. Friedman, Probabilistic Graphical Models: Principles and Techniques, The MIT Press, 2009.

[2] Z. YANG and V. C. Chen, "Mining and modeling for a metropolitan Atlanta ozone pollution decision-making framework," *IIE Transactions,* p. 607–615, 2007.

[3] A. LeBoulluec, N. Ohol, V. Chen, L. Zeng, J. Rosenberger and R. Gatchel, "Handling time-varying confounding in state transition models for dynamic optimization of adaptive interdisciplinary pain management," *IISE Transactions on Healthcare Systems Engineering,* 2018.

[4] T. J. Sargent, Dynamic Macroeconomic Theory, Harvard University Press, 1987.

[5] J. Hamilton, Time Series Analysis, Wiley, 1994.

[6] J. Pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, San Mateo, CA.: Morgan Kaufmann, 1988.

[7] J. Pearl, Causality: Models, Reasoning and Inference, Cambridge University Press, 2000.

[8] J. Pearl, M. Glymour and J. N.P., Causal Inference in Statistics: A Primer, Wiley, 2016.

[9] J. Pearl and D. Mackenzie, The Book of Why: The New Science of Cause and Effect, New York: Basic Books, 2018.

[10] J. Pearl, "An Introduction to Causal Inference," *The International Journal of Biostatistics - Causal Inference,* vol. 6, no. 2, p. Article 7, 2010.

[11] A. Gelman, "Review Essay: Causality and Statistical Learning," *American Journal of Sociology,* vol. 117, no. Number 3, pp. 955-966, 2011.

[12] N. Cartwright, Hunting Causes and Using Them: Approaches in philosophy and economics, London: Cambridge, 2007.

[13] N. Ohol, "ADJUSTING FOR TIME VARYING CONFOUNDING IN ADAPTIVE INTERDISCIPLINARY PAIN MANAGEMENT PROGRAM," University of Texas at Arlington, Arlington, 2018.

[14] K. P. Murphy, *Dynamic Bayesian Networks: Representation, Inference and Learning,* Berekey: UC Berkeley, 2002.

[15] P. J. Diggle, P. Heagerty, K. Y. Liang and S. L. Zeger, Analysis of Longitudinal Data, Oxford University Press., 2002.

[16] G. Verbeke and G. Molenberghs, Linear Mixed Models for Longitudinal Data, Belgium: Springer Series in Statistics, 2000.

[17] J. Hamilton, Time Series Analysis, Princeton University Press, 1994.

[18] M. Aoki, State space modeling of time series, Springer, 1987.

[19] A. Harvey, Forecasting, Structural Time Series Models, and the Kalman Filter, Cambridge University Press, 1989.

[20] M. West and J. Harrison, Bayesian forecasting and dynamic models, Springer, 1997.

[21] J. Durbin and S. J. Koopman, "Time series analysis of non-Gaussian observations based on state space models from both classical and Bayesian perspectives," *Journal of the Royal Statistical Society,* p. To aprear, 2000.

[22] J. Durbin and S. Koopman, Time Series Analysis by State Space Methods., Oxford University Press, 2001.

[23] S. Tong and D. Koller, "Active Learning for Parameter Estimation in Bayesian Networks," in *Neural Information Processing Conference*, 2000.

[24] R. Shumway and D. Stoffer, "An Appraoch to time series smoothing and forecasting using the EM algorithm," *Time series Analysis,* pp. 253-264, 1982.

[25] Z. Ghahramani and S. Roweis, "A unified review of linear gaussian models," *Neural Computation,* vol. 11, no. 2, pp. 305-345, 1999.

[26] B. D. Andesron and J. B. Moore, Optimal Filtering, New Jeresey: Prentice-Hall, 1979.

[27] G. C. Goodwin, Adaptive Filtering Prediction and Control, 1984.

[28] K. R. E. and R. Bucay, "New Results in Kalman Filtering and Prediction," *Journal of Basic Engineering (ASME),* pp. 95-108, 1961.

[29] H. Rauch, "Solutions to the linear smoothing problem," *IEEE Transactions on Automatic Control,* vol. 8, pp. 371-372, 1963.

[30] R. Kalman, "A New Approach to Linear Filtering and Prediction problems," *Journal of Basic Engineering,* pp. 35-45, 1960.

[31] T. Minka, "From Hidden Markov Models to Linear Dynamical Systems.," MIT, 1999.

[32] L. Ljung and T. Soderstrom, Theory and practice of recursive identification, Cambridge: MIT Press, 1983.

[33] Dempster A, Laird N, Rubin DB, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society,* vol. B39, pp. 1-38, 1977.

[34] Z. Ghahramani and G. Hinton, "Parameter Estimation for linear dynamical sysyems," Technical Report CRG-TR-96-2, Toronto, 1996.

[35] V. Digalakis, J. Rohlicek and M. Ostendorf, "ML Estimation of a stochastic linear sytem with the EM algorithm and its application to speech recognition.," *IEEE Transaction on Speech and Audio Processing,* vol. 1, no. 4, pp. 431-442, 1993.

[36] R. Shumway and D. Stoffer, "An approach to time series smoothing and forecasting using the EM Algorithm," *Journal of time series analysis,* pp. 253-264, 1982.

[37] B. H. Juang and L. R. Rabiner, "Hidden Markov models for speech recognition," *Technometrics,* vol. 33, pp. 251-272, 1991.

[38] S. Lauritzen and D. Spiegelhalter, "Local Computations with probabilities on graphical structures and their application to expert systems.," *Royal Statistical Society B,* vol. 50, no. 2, pp. 157-224, 1988.

[39] A. J. Viterbi, "Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm," *IEEE Transactions on Information Theory.,* vol. 13, no. 2, p. 260–269, 1967.

[40] L. Baum, T. Petrie, G. Soules and N. Weiss, "A maximization techinique occurring in the statistical analysis of probabilistic functions of Markov chains.," *The Annals of Mathematical Statistics,* vol. 41, pp. 164-171, 1970.

[41] Z. Ghahramani and G. Hinton, "Switching State-Space Models," University of Toronto, Toronto, 1996.

[42] Z. Ghahramani and G. Hinton, "Variational Learning for Switching State-Space Models," *Neural Computation,* vol. 12, no. 4, pp. 831-864, 2000.

[43] G. Goodwin and K. Sin, Adaptive Filtering prediction and control, Prentice-Hall, 1984.

[44] Y. Wang and V. Carey, "Working correlation structure misspecification, estimation, and covariate design: Implications for generalized estimating equations performance.," *Biometrika,* vol. 90, pp. 29-41, 2003.

[45] M. Pourahmadi, "Joint mean-covariance models with applications to longitudinal data: Unconstrained parameterisation.," *Biometrika,* vol. 86, pp. 677-90, 1999.

[46] M. Pourahmadi, "Maximum likelihood estimation of generalised linear models for multivariate normal covariance matrix.," *Biometrika,* vol. 87, pp. 425-435, 2000.

[47] C. Huang, "JOINT MEAN-COVARIANCE MODELLING AND VARIABLE SELECTION FOR LONGITUDINAL DATA ANALYSIS," University of Manchester, Manchester, 2010.

[48] N. M. Laird and J. H. Ware, "Random-effects models for longitudinal data," *Biometrics,* vol. 38, pp. 963-974, 1982.

[49] J. Nelder and R. Wedderburn, "Generalized Linear Models.," *Journal of the Royal Statistical Society,* vol. 135, 1972.

[50] K. Liang and S. Zeger, "Longitudinal data analysis using generalized," *Biometrika,* vol. 73, pp. 13-22, 1986.

[51] R. Stiratelli, N. M. Laird and J. H. Ware, "Random e®ects models for serial observations with binary response," *Biometrics,* vol. 40, pp. 961-971, 1984.

[52] S. Zeger, K. Liang and P. Albert, "Models for longitudinal data: a generalized estimating equation approach," *Biometrics,* vol. 44, pp. 1049-1060, 1988.

[53] R. Schall, "Estimation in generalized linear models with random e®ects," *Biometrika,* vol. 78, pp. 719-727, 1991.

[54] N. Breslow and D. Clayton, "Approximate Inference in Generalized Linear Mixed Models," *Journal of the American Statistical Association,* vol. 88, pp. 9-25, 1993.

[55] M. Davidian and D. Giltinan, Nonlinear Models for Repeated Measurement Data, Chapman & Hall, 1995.

[56] E. Vonesh, V. Chinchilli and K. Pu, "Goodness-Of-Fit in Generalized Nonlinear Mixed-Effects Models," *Biometrics,* vol. 52, pp. 572-587, 1996.

[57] G. Fitzmaurice, M. Davidian, G. Verbeke and G. Molenberghs, Longitudinal data analysis, Chapman and Hall, 2009.

[58] S. Greenland, J. Pearl and J. Robins, "Causal Diagrams for Epidemiologic Research," *Lippincott Williams & Wilkins,* vol. 10, no. 1, pp. 37-48, 1999.

[59] R. TIBSHIRANI, "Regression shrinkage and selection via the LASSO," *Journal of the Royal Statistical Society: Series B (Statistical Methodology),* vol. 58, no. 1, p. 267–288, 1996.

[60] R. Tibshirani, "Regression shrinkage and selection via the lasso: a retrospective," *J. R. Statist. Soc. B,* vol. 73, no. part 3, pp. 273-282, 2011.

[61] L.-A. Kirkland, F. Kanfer and S. Millard, "LASSO TUNING PARAMETER SELECTION," in *Proceedings of the 57th Annual Conference of SASA*, 49-56.

[62] H. ZOU, "The Adaptive Lasso and Its Oracle Properties," 2006.

[63] A. E. Hoerl and R. Kennard, "American Society for QualityRidge Regression: Biased Estimation for Nonorthogonal Problems," *Technometrics,* vol. 12, no. 1, pp. 55-67, 1970.

[64] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *J. R. Statist. Soc. B,* vol. 67, no. part 2, pp. 301-320, 2005.

[65] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *J. R. Statist. Soc. B,* vol. 68, no. Part 1, pp. 49-67, 2006.

[66] Y. Yang and H. Zou, "A Fast Unified algorithm for solving group-lasso penalized learning problems," 2014.

[67] J. Fan and R. Li, "\Variable selection via nonconcave penalized likelihood and its oracle properties," *Journal of the American Statistical Association,* vol. 96, pp. 1348-1360, 2001.

[68] H. Wang and C. Leng, "A note on adaptive group lasso," *Computational Statistics & Data Analysis,* vol. 52, no. 12, pp. 5277-5286, 2008.

[69] J. Friedman, T. Hastie and R. Tibshirani, "A note on the group lasso and a sparse group lasso," *arXiv:1001.0736,* 2010.

[70] M. KUBAT, R. C. HOLTE and S. MATWIN, "Machine Learning for the Detection of Oil Spills in Satellite Radar Images," *Machine Learning,* p. 195–215, 1998.

[71] J. M. Rohrer, "Thinking Clearly About Correlations and Causation: Graphical Causal Models," *Advances in Methods and Practices in Psychological Science,* vol. 1, no. 1, pp. 27-42, 2018.

[72] I. J. Myung, "The Importance of Complexity in Model Selection," *Journal of Mathematical Psychology,* vol. 44, pp. 190-204, 2000.

[73] L. Breiman, J. Friedman, C. J. Stone and R. Olshen, Classification and Regression Trees, Monterey: The Wadsworth statistics/probability series, 1984.

[74] T. Hastie and J. Qian, "Glmnet Vignette," 26 June 2014. [Online]. Available: https://web.stanford.edu/~hastie/glmnet/glmnet_alpha.html.

[75] J. Shao, "Linear Model Selection by Cross-Validation," *Journal of the American Statistical Association ,* vol. 88, pp. 486-494 , 1993.

[76] Y. Yang, "COMPARING LEARNING METHODS FOR CLASSIFICATION," *Statistica Sinica,* vol. 16, pp. 635-657, 2006.

[77] Y. Yang, "Consistency of cross validation for comparing regression procedures," *Ann. Statist.,* vol. 35, no. 6, pp. 2450-2473, 2007.

[78] M. KUBAT, R. C. HOLTE and S. MATWIN, "Machine Learning for the Detection of Oil Spills in Satellite Radar Images," *Springer, Machine learning,* vol. 30, pp. 195-215, 1998.

[79] I.-G. Chong and C.-H. Jun, "Performance of some variable selection methods when multicollinearity is present," *Chemometrics and intelligent laboratory systems,* vol. 78, pp. 103-112, 2005.

[80] D. G. W. Lewis, "A Sequential Algorithm for Training Text Classifiers," in *Proceedings of the Seventeenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1994.

[81] P. Lavori and R. Dawson, "A design for testing clinical strategies: Biased individually tailored within subject randomization," *J Royal Statist Soc Series A,* vol. 163, pp. 29-38, 2000.

[82] S. Murphy, "An experimental design for the development of adaptive treatment strategies," *Stat Med,* vol. 24, pp. 1455-1481, 2005.

[83] S. Murphy, "Optimal dynamic treatment regimes," *Journal of Royal Statistical Society, Series B,* vol. 65, no. 2, p. 31–355, 2003.

[84] A. LeBoulluec, N. Ohol, V. Chen, L. Zeng and J. Rosenberger, "Handling time-varying confounding in state transition models for dynamic optimization of adaptive interdisciplinary pain management," *IISE Transactions on Healthcare Systems Engineering,* vol. 0, no. 0, pp. 1-10, 2018.

[85] A. LeBoulluec, "Outcome and state transition modeling for adaptive interdisciplinary pain management," The University of Texas at Arlington, Arlington, 2013.

[86] H. M. R. JM., Causal Inference, Chapman & Hall, 2017.

[87] G. S, P. J and R. JM., "Causal diagrams for epidemiologic research," *Epidemiology,* pp. 359-374, 1999.

[88] J. Robins, "Association, causation, and marginal structural models.," *Synthese,* vol. 121, no. 1, pp. 151-179, 1999.

[89] J. Robins, M. Hernn and B. Brumback, "Marginal structural models and causal inference in epidemiology," *Epidemiology,* vol. 11, no. 5, pp. 550-560, 2000.

[90] G. S and M. H, "Confounding in health research.," *Annu Rev Public Health. ,* vol. 22, pp. 189-212, 2001.

[91] H. MA, H.-D. S and R. JM, "A structural approach to selection bias.," *Epidemiology,* vol. 15, no. 5, pp. 615-625, 2004.

[92] D. Rubin, "The use of matched sampling and regression," *Biometrics,* vol. 29, p. 184–203, 1973.

[93] D. Rubin, "Estimating causal effects of treatment in randomized and nonrandomized studies.," *Journal of Educational Psychology,* vol. 66, p. 688–701, 1974.

[94] P. Rosenbaum and D. Rubin, "The central role of the propensity score in observational studies for causal effects," *Biometrika,* vol. 70, p. 41–55, 1983.

[95] M. Hernn, B. Brumback and J. and Robins, "Marginal structural models to estimate the joint causal effect of non-randomized treatments," *Journal of the American Statistical Association,* vol. 96, no. 454, p. 440–448, 2001.

[96] M. Joffe, T. TenHave, H. Feldman and S. andKimmel, "Model selection,confounder control, and marginal structural models: Review and new applications," *American Statistician,* vol. 58, no. 4, pp. 272-279, 2004.

[97] M. Hernan and J. Robins, "Estimating causal effects in epidemiological data," *Journal of Epidemiology and Community Health,* vol. 60, p. 578–586, 2006.

[98] M. J. Van der Laan and S. Gruber, "Collaborative double robust targeted maximum likelihood estimation," *The International Journal of Biostatistics,,* vol. 6, no. Article 17, 2010.

[99] J. KANG and J. SCHAFER, "Demystifying double robustness: a comparison of alternative strategies for estimating a population mean from incomplete data," *Statistical Science,* vol. 22, pp. 523-539, 2007.

[100] R. NEUGEBAUER and M. VAN DER LAAN, "Why prefer double robust estimators in causal inference?," *Journal of Statistical Planning and Inference,* vol. 129, pp. 405-426, 2005.

[101] M. Cefalu, F. Dominici, N. D. Arvold and G. Parmigiani, "Model averaged double robust estimation," *Biometrics,* vol. 73, pp. 410-421, 2017.

[102] S. Greenland, "Invited commentary: Variable selection versus shrinkage in the control of multiple confounders," *American Journal of Epidemiology,* vol. 167, p. 523–529, 2008.

[103] E. Schisterman, S. Cole and R. Platt, "Overadjustment bias and unnecessary adjustment in epidemiologic studies," *Epidemiology,* vol. 20, no. 4, pp. 488-495, 2009.

[104] A. Rotnitzky, L. Li and X. Li, "A note on overadjustment in inverse probability weighted estimation," *Biometrika,* vol. 97, pp. 1-5, 2010.

[105] A. Patrick, S. Schneeweiss, M. Brookhart, R. Glynn, K. Rothman, J. Avorn and e. al., "The implications of propensity score variable selection strategies in pharmacoepidemiology: An empirical illustration.," *Pharmacoepidemiology and Drug Safety,* vol. 20, pp. 551-559, 2011.

[106] X. De Luna, I. Waernbaum and T. and Richardson, "Covariate selection for the nonparametric estimation of an average treatment effect," *Biometrika,* vol. 98, p. 861–875, 2011.

[107] M. Brookhart, S. Schneeweiss, K. Rothman, R. Glynn and J. Avorn, "Variable selection for propensity score models," *American Journal of Epidemiology,* vol. 163, p. 1149–1156, 2006.

[108] J. S. De Bonet, C. L. Isbell and J. P. Viola, "MIMIC: Finding Optima by Estimating Probability Densities," MIT PRESS, Cambridge, MA, 1997.

[109] S. M. Shortreed and A. Ertefaie, "Outcome-Adaptive Lasso: Variable Selection for Causal Inference," *Biometrics,* pp. 1111-1122, 2017.

[110] P. C. Austin and E. A. Stuart, "Moving towards best practice when using inverse probability of treatment weighting (IPTW) using the propensity score to estimate causal treatment effects in observational studies," *Statstics in Medicine,* vol. 34, no. 28, pp. 3661-3679, 2015.

[111] A. PC., "Using the standardized difference to compare the prevalence of a binary variable between two groups in observational research.," *Communications in Statistics - Simulations and Computation,* vol. 38, pp. 1228-1234, 2009.

[112] W. Ahrens and I. Pigeot, Handbook of Epidemiology, New York: Springer, 2005.

[113] G. M. D. Iqbal, "Multi-objective two-stage stochastic programming for adaptive interdisciplinary," The University of Texas at Arlington, Arlington,TX, 2017.

[114] J. Artner, S. Kurz, B. Cakir, H. Reichel and F. and Lattig, "Intensive interdisciplinary outpatient pain management program for chronic back pain: A pilot study," *J Pain Res,* vol. 5, pp. 209-216, 2012.

[115] Bertsekas, Dynamic Programming and Optimal Control, Boston: Athena Scientific, 2005.

[116] R. Bellman, "The Theory of Dynamic Programming," *Bulletin of the American Mathematical Society,* pp. 503-515, 1954.

[117] J. Whittaker, Graphical Models in Applied Multivariate Statistics., Chichester: England, 1990.

[118] Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering,* pp. 35-45, 1960.

[119] D. Simon, Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches, John Wiley & Sons, 2006.

[120] Z. Ghahramani, "Learning dynamic Bayesian networks," *Adaptive Processing of Sequences and Data Structures,* pp. 168-197, 1998.

[121] Chen Z, Barbieri R, Brown EN., " State-space modeling of neural spike train and behavioral data. In Oweiss K (Ed.)," *Statistical Signal Processing for Neuroscience and Neurotechnology,* pp. 161-200, 2010.

[122] McCullagh P, Nelder JA., Generalized Linear Models(2nd Edition), Chapman & Hall/CRC Press, 1989.

[123] Zhe Chen, Emery N. Brown, "State Space Model," 2013. [Online]. Available: http://www.scholarpedia.org/article/State_space_model.

[124] C Cervellera, VCP Chen, A Wen, "Optimization of a large-scale water reservoir network by stochastic dynamic programming with efficient state space discretization," *European Journal of Operational Research,* vol. 171, no. 3, pp. 1139-1151, 2006.

[125] S. Gal, "Optimal management of a multireservoir water supply system," *Water resource research,* vol. 15, no. 4, pp. 737-749, August 1979.

[126] L. S.-Y. Wu, J. S. P. Pai and J. Hosking, "An algorithm for estimating parameters of state-space models," *Statistics and probability letters,* pp. 99-106, 1995.

[127] P. McCullagh and J. Nelder, Generalized Linear Models, Chapman& Hall, 1989.

[128] L. Collins and S. Wugalter, "Latent Class Models for stage-sequentional dynamic latent variables," *Multivariate Behavorial Research,* pp. 131-157, 1992.

[129] H. P. Heagerty, "Marginalized transition models and likelihood inference for longitudinal categorical data.," *Biometrics,* pp. 342-51, 2002.

[130] P. Heagerty and S. L. Zeger, "Marginalized Multilevel Models and likelihood inference," *Statistical Science,* pp. 1-26, 2000.

[131] A. Azzalini, "Logistic Regression for Autocorrelated Data with Application to Repeated Measures," *Biometrika,* pp. 767- 775, 1994.

[132] S. Fruhwirth-Schnatter, Finite Mixture and Markov Switching Models, Austria: Springer Series in Statistics, 2006.

[133] B. L. D. Koch, "Statistical Methods for Variable Selection in Causal Inference," THE UNIVERSITY OF MINNESOTA, 2018.

[134] H. Bang and J. Robins, "Doubly Robust Estimation in Missing Data and Causal Inference Models," *Biometrics,* vol. 61, no. 4, pp. 962-973, 2005.

[135] S. M. Shortreed and A. Ertefaie, "Outcome-Adaptive Lasso: Variable Selection for Causal Inference," *Biometrics,* p. 1111–1122, 2017.

[136] G. Govaert and M. Nadif, "Block clustering with bernoulli mixture models: Comparison of different approaches.," *Computational Statistics & Data Analysis,* vol. 52, no. 6, p. 233–3245, 2008.

[137] G. Govaert and M. Nadif, "Latent block model for contingency table.," *Communications in Statistics - Theory and Methods,* vol. 39, no. 3, p. 416–425, 2010.

[138] G. Govaert and M. Nadif, "Clustering with block mixture models.," *Pattern Recognition,* vol. 36, no. 2, pp. 463 – 473,, 2003.

[139] P. S. Bhatia, S. Lovleff and G. Govaert, "blockcluster: An R Package for Model-Based Co-Clustering," *Journal of Statistical Software,* vol. 79, no. 6, pp. 1-24, 2017.

[140] L. Kirkland, F. Kanfer and S. Millard, "LASSO Tuning Parameter Selection," in *Proceedings of the 57th Annual Conference of the South African Statistical Association (SASA)*, 2015.

[141] B. EFRON, T. HASTIE, I. JOHNSTONE and R. TIBSHIRANI, "Least angle regression.," *The Annals of Statistics,* vol. 32, no. 2, p. 407–451, 2004.

[142] R. J. TIBSHIRANI and J. TAYLOR, "Degrees of freedom in LASSO problems," *The Annals of Statistics,* vol. 40, no. 2, p. 1198–1232, 2012.

[143] C. Lin, A. LeBoulluec, L. Zeng, V. Chen and R. and Gatchel, "An adaptive pain management framework.," *Health Care Management Science,* p. 17(3):270–283., 2013.