Statistical Methods for Multivariate Functional Data Clustering, Recurrent Event Prediction, and Accelerated Degradation Data Analysis

Zhongnan Jin

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Statistics

Yili Hong, Chair
Xinwei Deng
Inyoung Kim
Laura P. Sands
Xiaowei Wu

August 8 2019
Blacksburg, Virginia

Keywords: accelerated destructive degradation test, clustering, functional principal component analysis, geyser eruption, multivariate analysis, recurrent process, sensory data, thermal index, variable selection.

Statistical Methods for Multivariate Functional Data Clustering, Recurrent Event Prediction, and Accelerated Degradation Data Analysis

Zhongnan Jin

Abstract

In this dissertation, we introduce three projects in machine learning and reliability applications after the general introductions in Chapter 1. The first project concentrates on the multivariate sensory data, the second project is related to the bivariate recurrent process, and the third project introduces thermal index (TI) estimation in accelerated destructive degradation test (ADDT) data, in which an R package is developed. All three projects are related to and can be used to solve certain reliability problems. Specifically, in Chapter 2, we introduce a clustering method for multivariate functional data. In order to cluster the customized events extracted from multivariate functional data, we apply the functional principal component analysis (FPCA), and use a model based clustering method on a transformed matrix. A penalty term is imposed on the likelihood so that variable selection is performed automatically.

In Chapter 3, we propose a covariate-adjusted model to predict next event in a bivariate recurrent event system. Inspired by geyser eruptions in Yellowstone National Park, we consider two event types and model their event gap time relationship. External systematic conditions are taken account into the model with covariates. The proposed covariate adjusted recurrent process (CARP) model is applied to the Yellowstone National Park geyser data.

In Chapter 4, we compare estimation methods for TI. In ADDT, TI is an important index indicating the reliability of materials, when the accelerating variable is temperature. Three methods are introduced in TI estimations, which are least-squares method, parametric model and semi-parametric model. An R package is implemented for all three methods. Applications of R functions are introduced in Chapter 5 with publicly available ADDT datasets.

Chapter 6 includes conclusions and areas for future works.

Statistical Methods for Multivariate Functional Data Clustering, Recurrent Event Prediction, and Accelerated Degradation Data Analysis

Zhongnan Jin

General Audience Abstract

This dissertation focuses on three projects that are all related to machine learning and reliability. Specifically, in the first project, we propose a clustering method designated for events extracted from multivariate sensory data. When the customized event is corresponding to reliability issues, such as aging procedures, clustering results can help us learn different event characteristics by examining events belonging to the same group. Applications include diving behavior segmentation based on vehicle sensory data, where multiple sensors are measuring vehicle conditions simultaneously and events are defined as vehicle stoppages. In our project, we also proposed to conduct sensor selection by three different penalizations including individual, variable and group. Our method can be applied for multi-dimensional sensory data clustering, when optimal sensor design is also an objective.

The second project introduces a covariate-adjusted model accommodated to a bivariate recurrent event process system. In such systems, events can occur repeatedly and event occurrences for each type can affect each other with certain dependence. Events in the system can be mechanical failures which is related to reliability, while next event time and type predictions are usually of interest. Precise predictions on the next event time and type can essentially prevent serious safety and economy consequences following the upcoming event. We propose two CARP models with marginal behaviors as well as the dependence structure characterized in the bivariate system. We innovate to incorporate external information to the model so that model results are enhanced. The proposed model is evaluated in simulation studies, while geyser data from Yellowstone National Park is applied.

In the third project, we comprehensively discuss three estimation methods for thermal index. They are the least-square method, parametric model and sem-parametric model. When temperature is the accelerating variable, thermal index indicates the temperature at which our materials can hold up to a certain time. In reality, estimating the thermal index precisely can prolong lifetime of certain product by choosing the right usage temperature. Methods evaluations are conducted by simulation study, while applications are applied to public available datasets.

Dedicated to my mother.

Acknowledgements

I want to show my deepest gratitude to my advisor, Dr. Yili Hong. Not only has he passed the academic intelligence to me throughout the past five years, he is also a great educator who teaches in accordance with individual's aptitude. I would like to thank him for having faith in me and helping me become a better person.

I also would like to thank all my committee members, Dr. Xinwei Deng, Dr. Inyoung Kim, Dr. Xiaowei Wu for not only great lectures you offer, but also for the advice you provide on my research work. Special thanks to Dr. Laura P. Sands, for guiding my interdisciplinary research in the area of public health. She not only shows me how making good use of statistics can help people in need, but also widen my horizon by sharing her life experiences. I also would like to thank Dr. Jeffrey B. Birch who generously offered this great opportunity for me to be in the Department of Statistics from the very beginning.

I also appreciate all my group members for their genuine help, Dr. Yimeng Xie, Dr. Miao Yuan, Dr. Khaled F. Bedair, Li Xu, Yueyao Wang, and Hung-Ping Tung. I want to thank colleagues who have offered to help along this amazing journey, Dr. Danni Lu, Ruijin Lu, and Wenzhuo Pan.

Last but not least, I want to thank my family for the love and belief they have for me, supporting me unconditionally all the time.

Contents

List of Figures

List of Tables

Chapter 1 General Introduction

1.1 Multi-dimensional Sensory Data Clustering with Variable Selection

With development of technology, sensory data are widely used nowadays. One of the applications is to use multivariate functional data generated by a multi-dimensional sensor system. According to this multivariate functional data, customized events can be defined. Clustering techniques are applied to put events into different groups, so that event characteristics in each groups are learnt separately. This will lead to potential explanations of data generating mechanism. In this section, we introduce some key concepts used in Chapter 2.

1.1.1 Sensory Data

Sensory data are usually referred to the collection of measurements gathered by sensors. Depending on sensor functionalities, various characteristics can be measured. For example, temperature, humidity and light measurements can be collected by some sensors in continuous bases (Wu and Clements-Croome, 2007). In reliability, conditional events of interest are usually obtained by sensory measurements. Studies are conducted on building relations between sensory data and reliability. In the example above, sensory data are used to monitor home environment and detect possible appliance malfunctions. Because of the high frequency sensor measurements obtained, measurements can be considered as functional data (Ramsay and Silverman, 2007).

1.1.2 Multivariate Functional Data

Typically, multiple sensors are working together in a system, offering sources of functional data that are measuring different features of systems. This collection of multiple sensor

measurements will lead to the multivariate functional data. Because of large numbers of sensors there can be, and the high rate of measurement, it is usually challenging to use this high volume multivariate functional data. In the literature, Eubank (1999) proposes to use the spline method to conduct data smoothing, while selections of tuning parameters in B-spline and cubic spline are discussed in Eilers and Marx (1996) and Wood (2000). In applications, Berrendero et al. (2011) propose to use multivariate functional principal component analysis to transform the original data into a coefficient matrix. The dimension of multivariate functional data is reduced, so that the transformed data are much easier to deal with.

1.1.3 Feature Extraction

Events of interest can be defined and event features are usually restored in the multivariate functional data. In order to obtain event features comprehensively, we abstract measurements from all sensors within the same period of time where events occur. Measurements from different sensors can interact with each other, and together they characterize events dynamically. In Wu and Clements-Croome (2007), the in-home sensory measurements such as temperature and humidity are affecting each other, while abnormal readings from either sensor indicate possible household emergency like fire and flood. We are usually interested in recurrent events, while studying their extracted features can help us identify different event characteristics.

1.1.4 Sensory Data Clustering

One of the sensory data applications is clustering. Customized events from functional data are put into different groups based on their functional characteristics. With multivariate functional principal component analysis, we transform the multivariate functional data clustering to simpler multivariate data clustering problems. Non-parametric method like k-nearest-neighbor (KNN) is proposed (Cover and Hart, 1967) and applied on the financial data clustering (Tang et al., 2018), while model based clustering methods are quite popular as well. In Chapter 2, we consider the transformed coefficient matrix as they are from some mixture Gaussian models (Jacques and Preda, 2014), and use a model based clustering method on the

transformed multivariate functional data.

Since not all sensors are contributing in the event clustering, variable selection is meaningful in the process of clustering. We impose penalty terms on the likelihood so that variables that contribute less will be removed. In Chapter 2, we introduce different penalty terms according to our needs, they are individual penalty, variable penalty and group penalty. In literature, Wang and Zhu (2008) and Park et al. (2017) propose the use of penalty terms in model based clustering.

1.2 Reliability Estimations and Predictions

Recurrent event data and degradation data are very common in reliability applications, as well as in many other areas. Chapters 3 to 5 deals with prediction and estimation problems for those two types of data. Specifically, Chapter 3 introduces a covariate adjusted model to deal with data from bivariate recurrent event systems, and we predict the next event time and type for the system. Chapter 4 comprehensively compares methods to tackle an estimation problem in Accelerated Destructive Degradation Test (ADDT) data. Three methods to estimate thermal index (TI) are introduced, they are, the least-squares method, parametric method and semi-parametric method. In Chapter 5, an R package is implemented based on these methods. In following sections, we introduce some terms involved in these models.

1.2.1 Recurrent Event Process Time Prediction

Recurrent processes are usually used to model events that occur repeatedly over time. An important quantity for the recurrent process is gap times between two consecutive events. The gap time variable is often used to characterize the recurrent process. Distributional assumptions are usually put on this gap time variable. When there exists more than one event type, it becomes a recurrent event system. Compared to the recurrent event process with a single type, the recurrent event system is challenging to work with because we need to quantify not only the marginal behavior for each event type, but also possible dependence

among them. In literature, Yang et al. (2013) and Yang et al. (2017) apply lognormal and copula models on the multivariate recurrent event system.

In Chapter 3, we consider a case in which there are two events types in the recurrent system. In the recurrent event process system, it is of interest to predict the next event time and event type. When the recurrent events are malfunctions and adversities, precise predictions on event times and types can help us be preventive as well as be economically efficient. We use an innovative covariate adjusted model to quantify the gap time variable. Both copula models and lognormal models are introduced, which is new to literature.

1.2.2 Geyser System Eruption Prediction

We apply the proposed recurrent event process system model on a geyser system from the Yellowstone National Park. Geyser eruptions are signature attractions of Yellowstone National Park which attract tourists around the world. However, it is not always easy to witness this spectacular natural phenomenon, since geyser eruptions times can be hard to predict. Geyser eruptions are affected by earth tidal forces, barometric pressure, and tectonic stresses (Rinehart, 1972), while eruptions among nearby geysers are also closely related. Our model takes this among-geyser dependence into account as well as the marginal behaviors. For the first time, geyser eruption time predictions are considered from a pure statistical perspective.

1.2.3 Thermal Index Estimation

Thermal index is used as a measure of material strength in accelerated destructive degradation test (ADDT). The larger TI is for some materials, the longer materials can be held in certain temperature. A good estimate of TI will provide us better a better understanding of the reliability of certain materials. There are three methods to estimate TI, the traditional least-squares method, parametric method using likelihood function and semi-parametric method with use of spline (King et al., 2018 and Xie et al., 2018). In Chapter 4, we summarize these methods comprehensively.

1.2.4 R Package Implementation

In Chapter 5, three methods mentioned above are implemented in the R package "ADDT" to estimate TI from ADDT data. We provide instructions on how to use R functions in the package (Jin et al., 2017). Applications are described based on publicly available datasets.

1.3 Outline of the Dissertation

The rest of the dissertation is organized as follows. Chapter 2 introduces methods to solve multi-dimensional sensory data clustering problems. The proposed method makes use of multivariate functional principal component analysis to simplify the high volume sensory data. Mixture model assumption is imposed to the transformed data, while penalty incorporated likelihood accomplishes the variable selection automatically. Chapter 2 is mainly based on Jin et al. (2019b). In Chapter 3, we introduce the covariate adjusted recurrent process model motivated by geyser data from the Yellowstone National Park. Recurrent process is implemented on the geyser system, while a dependent recurrent process with covariate adjustment is described in general cases as well. The next event time in recurrent systems is predicted, and the model performances are evaluated in simulation study as well in geyser data application. Chapter 3 is based on Jin et al. (2019a). Chapter 4 describes an estimation problem for ADDT. Methods to estimate TI are discussed in Xie et al. (2017), which are the least-squares, parametric and semi-parametric methods. All TI estimation methods are implemented in an R package named "ADDT", where the use of R functions and data applications of the package are introduced in Chapter 5.

Bibliography

J. R. Berrendero, A. Justel, and M. Svarc. Principal components for multivariate functional data. *Computational Statistics and Data Analysis*, 55:2619–2634, 2011.

T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13:21–27, 1967.

P. H. C. Eilers and B. D. Marx. Flexible smoothing with B-splines and penalties. *Statistical Science*, 11:89–121, 1996.

R. L. Eubank. *Nonparametric regression and spline smoothing*. CRC press, 1999.

J. Jacques and C. Preda. Model-based clustering for multivariate functional data. *Computational Statistics and Data Analysis*, 71:92–106, 2014.

Z. Jin, Y. Xie, Y. Hong, and J. H. Van Mullekom. ADDT: An R package for analysis of accelerated destructive degradation test data. In D. G. Chen, Y. L. Lio, H. K. T. Ng, and T. R. Tsai, editors, *Statistical Modeling for Degradation Data*, chapter 14. Springer, NY: New York, 2017.

Z. Jin, K. F. Behair, and Y. Hong. Covariate adjusted recurrent processes for bivariate systems and an application to geyser eruption prediction. Manuscript, 2019a.

Z. Jin, Y. Hong, P. Du, and Q. Yang. Multivariate functional data clustering with automatic variable selection. Manuscript, 2019b.

C. B. King, Y. Xie, Y. Hong, J. H. Van Mullekom, S. P. DeHart, and P. A. DeFeo. A comparison of traditional and maximum likelihood approaches to estimating thermal indices for polymeric materials. *Journal of Quality Technology*, 50:117–129, 2018.

C. Park, M. C. Wang, and E. B. Mo. Probabilistic penalized principal component analysis. *Communications for Statistical Applications and Methods*, 24:143–154, 2017.

J. O. Ramsay and B. W. Silverman. *Applied functional data analysis: methods and case studies*. Springer, 2007.

J. S. Rinehart. Fluctuations in geyser activity caused by variations in earth tidal forces, barometric pressure, and tectonic stresses. *Journal of Geophysical Research*, 77:342–350, 1972.

L. Tang, H. Pan, and Y. Yao. K-nearest neighbor regression with principal component analysis for financial time series prediction. In *Proceedings of the 2018 International Conference on Computing and Artificial Intelligence*, pages 127–131. ACM, 2018.

S. Wang and J. Zhu. Variable selection for model-based high-dimensional clustering and its application to microarray data. *Biometrics*, 64:440–448, 2008.

S. N. Wood. Modelling and smoothing parameter estimation with multiple quadratic penalties. *Journal of the Royal Statistical Society*, 62:413–428, 2000.

S. Wu and D. Clements-Croome. Understanding the indoor environment through mining sensory dataxa case study. *Energy and Buildings*, 39(11):1183–1191, 2007.

Y. Xie, Z. Jin, Y. Hong, and J. H. Van Mullekom. Statistical methods for thermal index estimation based on accelerated destructive degradation test data. In D. G. Chen, Y. L. Lio, H. K. T. Ng, and T. R. Tsai, editors, *Statistical Modeling for Degradation Data*, chapter 12. Springer, NY: New York, 2017.

Y. Xie, C. B. King, Y. Hong, and Q. Yang. Semiparametric models for accelerated destructive degradation test data analysis. *Technometrics*, 60:222–234, 2018.

Q. Yang, N. Zhang, and Y. Hong. Statistical reliability analysis of repairable systems with dependent component failures under partially perfect repair assumption. *IEEE Transactions on Reliability*, 62:490–498, 2013.

Q. Yang, Y. Hong, N. Zhang, and J. Li. A copula-based trend-renewal process model for analysis of repairable systems with multitype failures. *IEEE Transactions on Reliability*, 66:590–602, 2017.

Chapter 2 Multivariate Functional Data Clustering with Automatic Variable Selection

Abstract

Multi-dimensional sensory data are widely available nowadays and one potential application is the clustering of certain events of interest. It is usually challenging to work with high volume multivariate functional data in which there exists sensors that do not contribute to clustering. In this chapter, we conduct multivariate functional data clustering in an unsupervised manner while considering automatic variable selection to remove uninformative sensors. Dimensions of high volume data are reduced by using multivariate functional principal components. The functional principal component analysis (FPCA) enables us to transform multi-dimensional sensory data into a coefficient matrix. We model this transformed data using Gaussian mixture distribution with K distinct centers to do model based clustering. We use penalty based maximum likelihood to conduct automatic variable selection. Individual, variable and group penalties are considered in the variable selection procedure. Performance of the proposed methods are investigated in the simulation study. An application of engineering sensory data is carried out. Conclusions and remarks on our methods are included.

Key Words: adaptive penalty, functional data clustering, functional principal component analysis, group lasso, sensory data, variable selection.

2.1 Introduction

2.1.1 Background

With development of sensor and communication technologies, high dimensional sensory data clustering becomes increasingly interesting. Applications are available including driving behaviour segmentations from vehicle sensors (Ozguner et al., 2007) and life threatening emergency identification from implantable body sensors (Van Laerhoven et al., 2004). In a system with active sensors, multi-dimensional sensor data can be obtained on-line at a high rate (e.g., per second or millisecond). Various conditions of the system such as malfunctions can be obtained by sensory measurements. In a single-sensor system, suppose the sensor measures the system temperature, it indicates possible systematic malfunctions when temperatures reach certain degree. In a multi-sensor system, systematic malfunctions are determined by multiple sensory measurements together. For example, there are three sensors measuring the system temperature, humidity and pressure, respectively. Any abnormal reading of these sensors could indicate possible systematic malfunctions. In multi-dimensional sensor cases, system conditions are usually affecting each other dynamically. In the example above, changes of temperature will affect the humidity and pressure. Because of this, it is more difficult to detect systematic conditions compared to a single sensor case. In a system with multiple sensors and continuous measurement, multivariate functional data are generated. In such data, each dimension represents measurements from one particular sensor. It is challenging to work with multivariate functional data with a large number of sensors and high rates of sensory measurements.

Different events can be extracted from multivariate functional data, while one of the applications is to cluster customized events into different groups. Clustering results will help us examine grouped events separately, reveal events similarities, and essentially explain the data generating mechanism. In clustering processes, data are from multiple sensors, and usually in different measurement units. We have to integrate data intelligently and make use of in-

formation restored in the multi-dimensional functional data. Further challenges occur when data are from sensors with a subset of which do not contribute to clustering. In reality, some sensors in a multi-dimensional sensor system are irrelevant to clustering of certain events. It's more efficient to remove data from those uninformative sensors in the process of clustering. In addition, selected variables of importance help us gain a better understanding of factors causing the customized events differentiation. This clustering result based on selected important factors essentially offers us initial understanding on this massive multivariate functional data.

In this chapter we propose a novel method for multi-dimensional sensory data clustering. The objective is to conduct clustering using high volume multivariate functional data, and perform automatic variable selection simultaneously, which is new to literature.

2.1.2 Engineering System Sensory Data

One example of multivariate sensory data is from an engineering system. Engineering sensory data can be observed from various applications. For instance, aircraft engines generate multivariate sensory data where sensors are located at different components of an engine. These multivariate sensory data are created while the engine is running. Another example of multivariate sensory data is from vehicle sensors. Sensors measuring different parts of vehicles are actively gathering data when vehicles are running. In both cases, interesting recurrent events can occur, where characteristics of each event observation can be different. For instance, we are usually interested in aircraft engine failures. Multiple degradation types can be observed prior to failures and each type needs different types of management. In addition, vehicle stoppages are of interest since stoppage procedure can be used as an indicator of driving behaviors. In both cases, it is often more interesting to examine time around events. These time windows can be used to learn multivariate functional data, since characteristics of events are restored in the multivariate functional data.

In Figure 2.1, we present sensory data examples from an engineering system. Specifically, each plot represents a particular sensor and each line stands for one observation. For each ob-

servation, the event is observed at measure point 30, so that the procedure prior to each event is stored in data. Among all installed sensors, only some of them reflect different event characteristics, while others are irrelevant. In the clustering procedure, including irrelevant sensors will not only increase the computational burden, but also make interpretations misleading. On the other hand, with the selected sensors impacting the event clustering procedure, it helps to reveal the data generating mechanism. In this study, we need to use information from all sensors intelligently, while removing those uninformative ones.

2.1.3 Related Literature

In literature, functional data have been widely studied. Ramsay and Silverman (2007) provide a classic resource on handling the functional data in general. They suggest choosing a dimension reduction method according to the underlying data. Specifically, if the data shows seasonality as in the finance and ecology data in Henderson (2006), the Fourier smoothing can be used. Otherwise we usually employ the spline method. Eubank (1999) provides a systematic way to smooth the original functional data using splines. Even though the splines are flexible in fitting the raw data, the choice of knots and tuning parameters can be difficult. Marx and Eilers (1998) discuss the selection of tuning parameters for the penalized B-spline. Wood (2000) describes the cubic spline with multiple penalty terms so that parameters are simplified in an automatic manner. Other smoothing methods are considered in various applications. For example, Panaretos et al. (2013) use the functional Discrete Fourier Transform (fDFT) method, where stationary time series data are required. In this chapter, we use functional principal component analysis (FPCA) to conduct dimension reduction, where fewer constraints are needed.

FPCA methods are widely applied in functional data analysis. Locantore et al. (1999) and Viviani et al. (2005) apply FPCA to the corneal data and fMRI data in brain research. FPCA reduces the data dimension by using only principal components that explain most of the variation in the original data. Ways to choose principal components are discussed in Li et al. (2013). In this way, the original high volume functional data are characterized into the

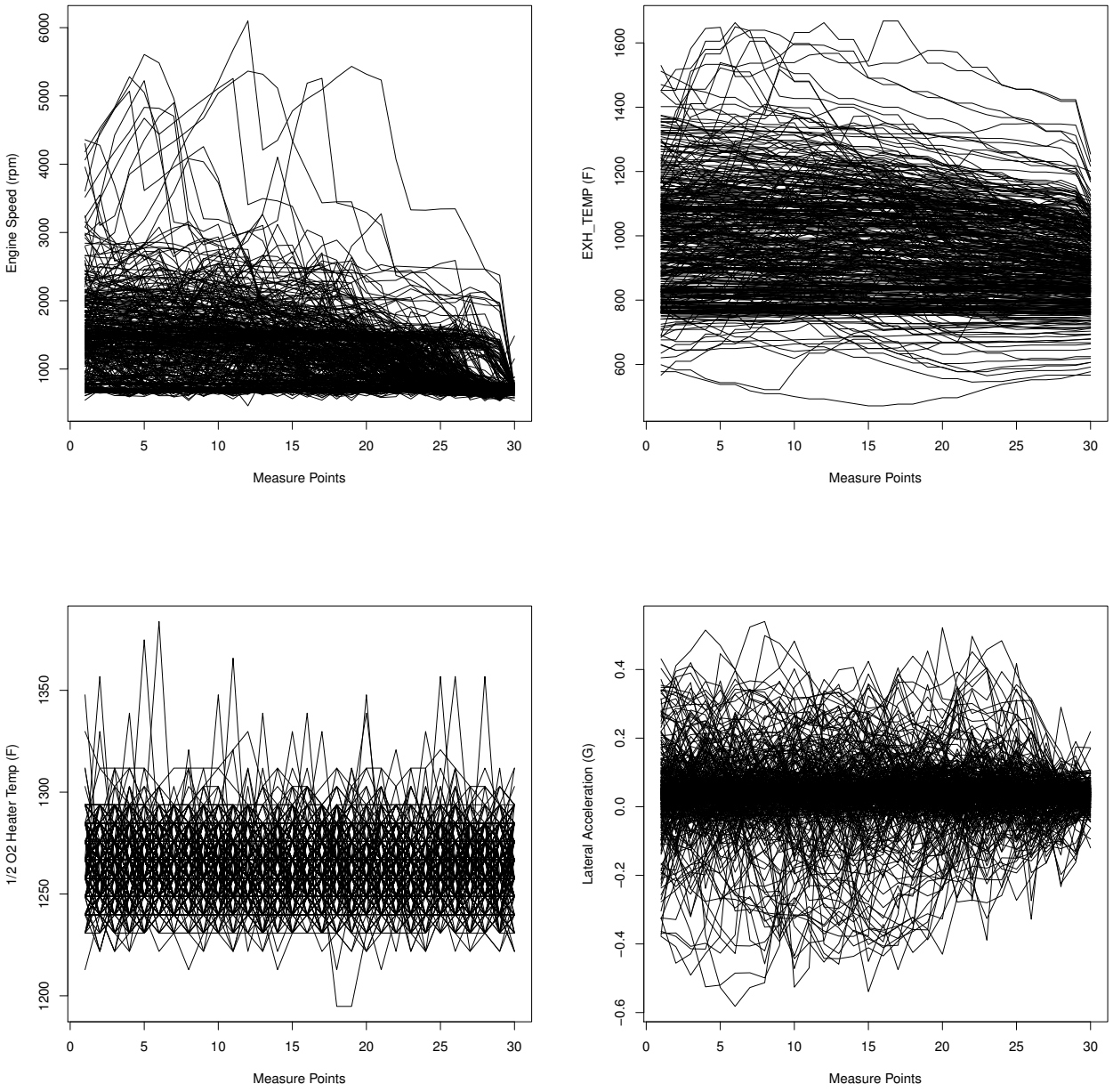


Figure 2.1: Examples of engineering sensory data. Measurements from four sensors are shown where events happen at measure point 30. Measure time window is 30 measure points prior to the defined event.

principal component coefficients. With the use of FPCA, clustering techniques can be applied on the transformed data such as the coefficient matrix.

Traditional clustering methods include the distance based k-nearest neighbour (KNN) introduced by Cover and Hart (1967). Recently, Tang et al. (2018) apply KNN clustering method on the principal components from finance data. Kamath and Mahato (2009) conduct similar applications to study colonic mucosal tissue fluorescence spectra. In this chapter, we assume the transformed coefficient matrix follows some distributions, so that the model based clustering method can be employed. Bouveyron and Brunet-Saumard (2014) conduct a complete review of model based clustering methods. Nguyen and Gelfand (2011) assume a Dirichlet process for clustering, while McNicholas and Murphy (2010) use Gaussian mixture models. However, all work above is regarding to univariate functional data.

Regarding multivariate functional data, Berrendero et al. (2011) use the multivariate functional principal component analysis (MFPCA) as an extension to the FPCA, and Happ and Greven (2018) apply MFPCA when functional resources are in different domains. In this chapter, we use FPCA on each dimension of multivariate functional data, where the Karhunen-Loeve expansion is applied with a finite truncation.

Clustering methods for single source functional data can also be applied to the multivariate functional data after the FPCA transformation. Jacques and Preda (2014) introduce model based clustering methods for multivariate function data, where mixture model assumptions are applied. However, they make use of all variables in the data including ones likely to have no contribution to the clustering. In multivariate functional data, with the assumption that not all variables are contributing to the clustering procedure, some authors also implement penalty terms to the likelihood function. As in Wang and Zhu (2008) and Park et al. (2017), authors force some variables to be removed from the clustering procedure .

In classic variable selection, penalization methodologies include the ridge penalty (Hoerl and Kennard, 1970) and the lasso penalty (Tibshirani, 1996). Zou (2006) proposes adaptive lasso where a different weight is assigned on to tuning parameters.

In FPCA transformed matrix, variables from the same functional source represent similar

information, so that they should be considered as a whole part in the penalization. Yuan and Lin (2006) use the grouped lasso penalization in the model selection, where the grouped variables are in and out of the model at the same time. While adaptive penalization methods are applied to group lasso by Wang and Leng (2008), Xie et al. (2008) extend this grouped penalization in the clustering process. However, there is no method dealing with multivariate functional data clustering with automatic variable selection.

In this chapter, we focus on the multivariate functional data clustering with automatic variable selection. We use the FPCA and adaptive penalized likelihood with mixture Gaussian distributions. Hyperparameter selections regarding adaptive penalties are barely discussed in literature where special cases of weight parameter are usually considered. In this chapter, we choose all hyperparameters simultaneously based on adjusted BIC.

2.1.4 Overview

The rest of the chapter is organized as follows. Section 2.2 introduces the multi-dimensional sensory data and notation used for subsequent analysis. Section 2.3 describes the developed method. Performances of proposed methods are evaluated by simulation study in Section 2.4. An application on engineering sensory data is discussed in Section 2.5. Conclusion and remarks are described in Section 2.6.

2.2 Multi-dimensional Sensory Data

In this section, we introduce notation and data that are used in our methods. Suppose there are p sensors in the measurement process. Each sensor provides continuous measurements when the sensor is active. Specifically, $X_j(t)$ represents the sensor reading for the j^{th} sensor at time t. Here $t \in [0, T]$ is the period of time when sensors are active, $t = 0$ is the first point of measurement and $t = T$ is the last measurement. We denote the measurement of p sensors with a p dimensional vector $\boldsymbol{X}(t) = [X_1(t), \cdots, X_p(t)]'$, where each dimension represents the sensor measurement at a particular time t.

Events can be defined as any acute recurrent phenomenon where event occurrences can be observed repeatedly. Suppose that there are n events and they are observed at time points T_i where $i = 1, \cdots, n$. Event times satisfies

$$0 < T_1 < T_2 < \cdots < T_n < T.$$

For the i^{th} event, we abstract a time window of length t_w prior the event occurrence, e.g., $[T_i - t_w, T_i]$. Measurements of p sensors in this time window is referred as the multi-dimensional sensory observation for this event. Specifically, we denote the sensory observation of the i^{th} event to be $\boldsymbol{X}_i(t) = [X_{i1}(t), \cdots, X_{ip}(t)]'$ where $t \in [T_i - t_w, T_i]$. We apply the same window of length t_w for all events. For rest of the chapter, we use the notation $\boldsymbol{X}_i$ to represent sensory observations from the i^{th} event, where $\boldsymbol{X}_i = \{\boldsymbol{X}_i(t) : t \in [T_i - t_w, T_i]\}$. Event details are restored in the multi-dimensional sensory measurements prior each event. When there exist different characteristics among events, they could be obtained by different sensory observations $\boldsymbol{X}_i$ where $i = 1, \cdots, n$. In this chapter, we denote all observations from a single sensor j as $\boldsymbol{X}^j(t) = \{X_{ij}(t) : i = 1, \ldots, n \text{ and } t \in [T_i - t_w, T_i]\}$.

In the engineering sensor example, the event is defined as moments when system speeds are reduced to zero. Time point T_i are system stoppage moments, while the length of each measure time window is 30, i.e., $t_w = 30$ for all observations. Measurements in $[T_i - t_w, T_i]$ records details prior each stoppage where $i = 1, \cdots, 419$. Our objective is to apply event clustering methods on observed multi-dimensional sensory data. Figure 2.2 shows $\boldsymbol{X}^j(t)$ for speed sensor in the engineering sensory data.

2.3 The Proposed Clustering Method

We propose a two-step clustering method: a) use FPCA to transform multivariate functional data to multivariate coefficient matrix, b) employ model based algorithm with penalty terms to do clustering and automatic variable selection. In this section, we introduce this method in details.

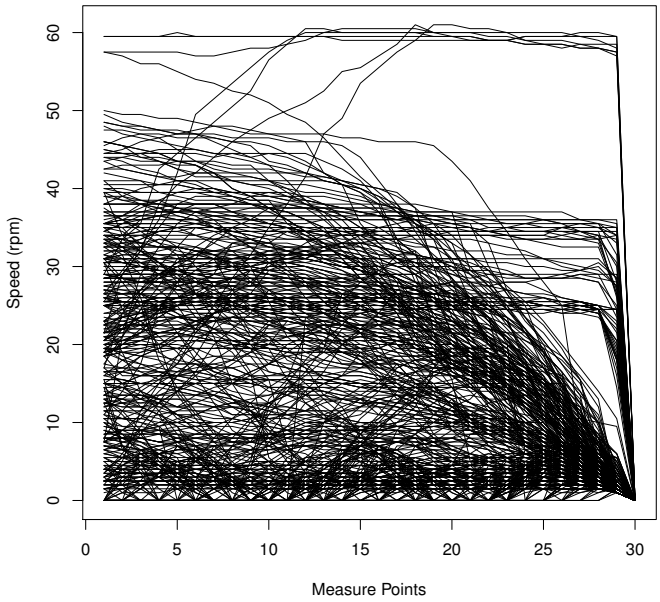


Figure 2.2: Measurements of 419 distinct system stoppages for the speed sensor in engineering sensory data.

2.3.1 Functional Principal Component Transformation

Here we use FPCA to transform functional data to multivariate coefficient matrix. Functional data from a single sensor can be decomposed by the Karhunen-Loeve expansion. For the j^{th} sensor, the i^{th} observation is expanded as

$$X_{ij}(t) = \mu_j(t) + \sum_{l=1}^{\infty} C_{ilj} f_{lj}(t), \quad t \in [0, T]. \tag{2.1}$$

Here, $\mu_j(t)$ is mean function, $f_{lj}(t)$ are functional principal functions for sensor j, and $C_{ilj}, i = 1, \cdots, n$ are principal component coefficients. For principal functions in (2.1), it has the constraint that

$$\int_0^T f_{lj}(t) f_{l'j}(t) dt = \begin{cases} 1, & \text{when} \quad l = l' \\ 0, & \text{when} \quad l \neq l' \end{cases}.$$

We denote $\boldsymbol{C}_{lj} = (C_{1lj}, \cdots, C_{nlj})'$ to be the principal component coefficient vector for the j^{th} sensor and l^{th} principal component. For different l, $\boldsymbol{C}_{lj}$ are uncorrelated random variables

with mean zero and variance λ_{lj}. Here λ_{lj} are eigen values from the FPCA with correspond to data $\boldsymbol{X}^j(t)$. Principal component function $f_{lj}(t)$ and coefficient C_{ilj} follow the relation where

$$C_{ilj} = \int_0^T [X_{ij}(t) - \mu_j(t)] f_{lj}(t) dt.$$

In practice, the expansion in (2.1) is often to truncate the series with finite addition. That is,

$$X_{ij}(t) \approx \mu_j(t) + \sum_{l=1}^{L_j} C_{ilj} f_{lj}(t), \quad t \in [0, T]. \tag{2.2}$$

In this chapter, the choice of L_j is large enough so that at least 95% of the data variation is preserved. With data $\boldsymbol{X}^j$, principal components coefficients $\boldsymbol{C}_{lj}$ can be estimated as $\boldsymbol{c}_{lj}$ using functional data analysis techniques. In this way, functional data for each event $\boldsymbol{X}_i$ are transformed to a vector where elements in the vector are c_{ilj}. Here $l = 1, \cdots, L_j$ and $j = 1, \cdots, p$. With all n events $\boldsymbol{X}_i$ in the data, we obtain a coefficient matrix by aggregating each transformed coefficient vector. The transformed coefficient matrix has dimension $n \times q$, where $q = \sum_{j=1}^{p} L_j \times p$. Each event corresponds to a row in the matrix, and characteristics from the j^{th} sensor are restored in L_j columns. Clustering methods discussed in following sections are applied upon this coefficient matrix. For rest of the chapter, we denote rows from this coefficient matrix as $\boldsymbol{c_i} = (c_{i1}, \cdots, c_{iq})'$ where $i = 1, \cdots, n$.

2.3.2 Model Based Clustering with Variable Selection

Here we use model based algorithm with a penalty term to do clustering. By applying penalty terms to likelihood functions, automatic variable selection will be performed. Suppose data are generated by a Gaussian mixture distribution, the probability density function (pdf) is,

$$g(\boldsymbol{c}_i) = \sum_{k=1}^{K} \pi_k f_k(\boldsymbol{c}_i; \boldsymbol{\mu}_k, \Sigma).$$

Here the constant K is a pre-set number of cluster, π_k are proportions satisfying $\sum_{k=1}^{K} \pi_k = 1$, and f_k is a normal pdf for the k^{th} cluster with mean $\boldsymbol{\mu}_k$ and covariance matrix Σ. That is,

$$f_k(\boldsymbol{c}_i; \boldsymbol{\mu}_k, \Sigma) = \frac{1}{(2\pi)^{\frac{q}{2}} |\Sigma|^{\frac{1}{2}}} \exp\left[-\frac{1}{2} (\boldsymbol{c}_i - \boldsymbol{\mu}_k)' \Sigma^{-1} (\boldsymbol{c}_i - \boldsymbol{\mu}_k) \right].$$

We denote $\boldsymbol{\mu}_k = (\mu_{k1}, \cdots, \mu_{kq})'$ as the mean component of the k^{th} cluster, where q is the number of variables. We assume Σ are the same across clusters. Let δ_{ik} be the indicator for observation i in cluster k such that

$$\delta_{ik} = \begin{cases} 1, & \text{if } \boldsymbol{c}_i \text{ is from cluster } k \\ 0, & \text{if } \boldsymbol{c}_i \text{ is from cluster } u \text{ where } u \neq k \end{cases}.$$

The log-likelihood of Gaussian mixture distribution with n observations is

$$l(\boldsymbol{\theta}) = \sum_{i=1}^{n} \sum_{k=1}^{K} \delta_{ik} \Big\{ \log(\pi_k) + \log\left[f_k(\boldsymbol{c}_i; \boldsymbol{\mu}_k, \Sigma) \right] \Big\}. \tag{2.3}$$

Typically, with more number of clusters K and variables q included, log-likelihood in (2.3) will be larger. In order to perform variable selection on top of clustering, we apply a penalty term to (2.3) and it becomes

$$l_P(\boldsymbol{\theta}) = \sum_{i=1}^{n} \sum_{k=1}^{K} \delta_{ik} \Big\{ \log(\pi_k) + \log\left[f_k(\boldsymbol{c}_i; \boldsymbol{\mu}_k, \Sigma) \right] \Big\} - p_\lambda(\boldsymbol{\theta}). \tag{2.4}$$

Penalty term $p_\lambda(\boldsymbol{\theta})$ in (2.4) can have different forms according to the need. In this chapter, we discuss three types of penalties including individual, variable and group penalties. Each penalty term makes use of mean components differently. Specifically, the individual penalty penalizes each mean component separately, the variable penalty focuses on the largest mean component corresponding to each variable in all clusters. The group penalty utilizes prior knowledge on the natural group information for all variables, and mean components from the same group are penalized simultaneously. Individual, variable and group penalties are defined

as

$$\text{Individual penalty:} \quad p_\lambda(\boldsymbol{\theta}) = \lambda \sum_{j=1}^{q} \sum_{k=1}^{K} w_{kj} \cdot |\mu_{kj}|,$$

$$\text{Variable penalty:} \quad p_\lambda(\boldsymbol{\theta}) = \lambda \sum_{j=1}^{q} w_j \cdot \max_s(|\mu_{sj}|),$$

$$\text{Group penalty:} \quad p_\lambda(\boldsymbol{\theta}) = \lambda \sum_{k=1}^{K} \sum_{m=1}^{M} w_{k_m} \sqrt{k_m} ||\boldsymbol{\mu}_k^m||.$$

In group penalty, we dissect the mean component vector by $\boldsymbol{\mu}_k = (\boldsymbol{\mu}_k^{1'}, \boldsymbol{\mu}_k^{2'} \cdots \boldsymbol{\mu}_k^{M'})'$, so that each $\boldsymbol{\mu}_k^m$ represents a natural variable group for $m = 1, \cdots, M$. Suppose dimensions for each subgroup are $\dim(\boldsymbol{\mu}_k^m) = k_m$, and they satisfy $\sum_{m=1}^{M} k_m = q$. In each penalty, w is referred as the weight. It is working closely with hyperparameter λ to make the penalization more flexible. Specifically, λ controls at the universal level since it remains the same for all variables and clusters. Weight w is adjusting the penalization at a lower level because we can apply different w_{kj} for each variable j and cluster k for $j = 1, \cdots, p$ and $k = 1, \cdots, K$. In the individual penalty, one can define the weight according to mean components for each variable and cluster as in Zou (2006). That is,

$$w_{kj} = \frac{1}{|\tilde{\mu}_{kj}|^\gamma}, \quad \text{where} \quad \tilde{\mu}_{kj} = \frac{\sum_{i=1}^{n} \delta_{ik} c_{ij}}{\sum_{i}^{n} \delta_{ik}} \tag{2.5}$$

is the mean component estimates without penalizations. In the variable penalty, we define $w_j = w_{kj}$ where $k = \arg\max_s(|\mu_{sj}|)$. Here w_{kj} are same with ones in the individual penalty. In the group penalty, data $\boldsymbol{c}_i$ can be dissected into a vector $(\boldsymbol{c}_i^1, \cdots, \boldsymbol{c}_i^M)'$ similar to the mean component decomposition. We define the weight as

$$w_{k_m} = \frac{1}{||\tilde{\boldsymbol{\mu}}_k||^\gamma}, \quad \text{where} \quad \tilde{\boldsymbol{\mu}}_k^m = \frac{\sum_{i=1}^{n} \delta_{ik} \boldsymbol{c}_i^m}{\sum_{i=1}^{n} \delta_{ik}}. \tag{2.6}$$

where q and K are the number of variables and clusters, respectively. Here $|\cdot|$ and $||\cdot||$ are L-1 and L-2 norms respectively. γ is a hyperparameter corresponding to weights. Calculations of (2.5) and (2.6) are shown in Appendix 2.6. Implementation of the weight enhances the

variable selection performances tremendously. With large λ and γ, it penalizes variables with mean components close to zero more with heavily weights. On the other hand, variables with large mean components are not easily removed since the weight becomes small as well as the penalty term. In each penalty term, λ and γ are hyperparameters used for different penalization perspectives. Specifically, λ controls the overall scale and γ adjusts the weight for mean components from each clusters. In this study, we need to choose hyperparameters λ and γ simultaneously. Hyperparameter selection is discussed in Section 2.3.4.

Individual, variable and group penalties all remove variables who barely contribute to the clustering procedure. However, according to ways each penalty are built, they acts differently when removing variables from clustering. In the next section, we introduce the penalty details and variable removal criterion for all penalties.

(1) Individual Penalty $p_\lambda(\boldsymbol{\theta}) = \lambda \sum_{j=1}^{p} \sum_{k=1}^{K} w_{kj} \cdot |\mu_{kj}|$. This penalty term separately controls mean component μ_{kj} for different cluster and variable. For example, μ_{kj} and $\mu_{k'j}$ for $k \neq k'$ are not related in the estimation even μ_{kj} and $\mu_{k'j}$ are both corresponding to the j^{th} variable. In individual penalty, we remove the j^{th} variable from clustering procedure when corresponding mean components are forced to zero for all K clusters. That is,

$$\mu_{1j} = \mu_{2j} = \cdots = \mu_{Kj} = 0.$$

For variables with little contribution to clustering, mean components will be small or similar across K clusters. With large enough λ and γ, penalty term values will be large and corresponding variables will be removed automatically.

(2) Variable Penalty $p_\lambda(\boldsymbol{\theta}) = \lambda \sum_{j=1}^{p} w_j \cdot \max_s(|\mu_{sj}|)$. It sets mean components μ_{kj} equal to zero for all $k = 1, 2, \cdots, K$, if the maximum mean component among K clusters is restricted to zero. As a result, the variable removal criterion is

$$\max_s(|\mu_{sj}|) = 0.$$

Compare to the individual penalty, a variable penalty considers mean components from the same variable simultaneously since they share similar variable information. Considering this variable homogeneity, variable selection only depends on the $\max_s(|\mu_{sj}|)$. With large hyper-parameters λ and γ, $\max_s(|\mu_{sj}|)$ will be forced to zero and the j^{th} variable will be removed.

(3) Group Penalty $p_\lambda(\boldsymbol{\theta}) = \lambda \sum_{k=1}^{K} \sum_{m=1}^{M} w_{km} \sqrt{k_m} ||\boldsymbol{\mu}_k^m||$. Our variables can sometimes be grouped naturally based on prior knowledge. For instance, L variables from Karhunen-Loeve decomposition in (2.2) come from the same functional data, so that we can assign corresponding L variables into the same group. Instead of single variable selection in the first two penalties, variables in a group will be in or out of the clustering procedure together for group penalty. The variable removal criterion is when all variables elements are set to zero for $k = 1, \cdots, K$. That is,

$$\boldsymbol{\mu}_k^m = \mathbf{0}.$$

With large enough λ and γ, all three penalties will automatically set mean components to zero for those variables without much contribution to clustering. Closed-form parameter estimations exist for all penalty terms. We introduce the estimation procedure in Section 2.3.3.

2.3.3 Estimation Procedure

For the Gaussian mixture distribution in (2.3), latent indicators δ_{ik} are included for each observation. It is common to apply the expectation maximization (EM) algorithm to estimate parameters. In an EM procedure, each parameter is updated iteratively towards the one with smallest loss function. Parameters in our study include π_k, $\boldsymbol{\mu}_k$, and Σ, where $k = 1, \cdots, K$. With usage of different penalty terms, parameter update methods vary. We introduce both similarities and differences for the EM algorithm among individual, variable and group penalties.

In general, the EM algorithm is consist of expectation and maximization procedures. The expectation procedure estimates indicator for observations δ_{ik} and proportion indicator π_k.

The maximization procedure updates distributional parameters $\boldsymbol{\mu}_k$ and Σ based on indicators from the expectation procedure. The EM algorithm used in our study is shown as follows.

(i) Parameter Initialization Given number of cluster K and data $\boldsymbol{c}_i$, we initialize parameters π_k^0, $\boldsymbol{\mu}_k^0$ and Σ^0 for $k = 1, \cdots, K$. Although various methods can be employed, we use values returned from the KNN (Dempster et al., 1977) to initialize the EM algorithm.

(ii) Expectation Steps For the r^{th} EM iteration, we estimate indicator for observation δ_{ik} as τ_{ik} using the Bayes rule. Specifically,

$$\tau_{ik}^{r+1} = \frac{\pi_k^r f_k(\boldsymbol{c}_i; \boldsymbol{\mu}_k^r, \Sigma^r)}{\sum_{k=1}^K \pi_k^r f_k(\boldsymbol{c}_i; \boldsymbol{\mu}_k^r, \Sigma^r)},$$

where $\boldsymbol{\mu}_k^r$ and Σ^r are estimated from the last step. The proportion indicator is updated as

$$\pi_k^{r+1} = \frac{\sum_{i=1}^n \tau_{ik}^{r+1}}{n}.$$

(iii) Maximization Steps With the updated τ_{ik}^{r+1} and π_k^{r+1}, distributional parameters are calculated through the maximization step. Σ is a diagonal matrix whose j^{th} element on the diagonal is estimated as

$$\hat{\sigma}_j^{2,(r+1)} = \sum_{k=1}^K \sum_{i=1}^n \hat{\tau}_{ik}^{(r)} (c_{ij} - \hat{\mu}_{kj}^{(r)})^2 / n. \tag{2.7}$$

The calculation (2.7) is shown in Appendix 2.6. In this study, expectation steps for Σ are the same for all three penalty terms since penalty terms do not include the covariance matrix. On the other hand, maximization procedures for the mean component vary according to different penalties applied on the log-likelihood. Maximization steps for mean components under different penalties are introduced as follows.

(1) Individual Penalty With adaptive weights, the sufficient and necessary condition for $\hat{\mu}_{kj}$ to globally maximize function $l_P(\boldsymbol{\theta})$ is that

$$\begin{cases} \frac{\sum_{i=1}^n \tau_{ik} c_{ij}}{\sum_{i=1}^n \tau_{ik}} = \left(\frac{\lambda w_{kj}}{\sum_{i=1}^n \tau_{ik}} + |\hat{\mu}_{kj}| \right) \text{sign}(\hat{\mu}_{kj}), & \text{if and only if } \hat{\mu}_{kj} \neq 0 \\ \frac{|\sum_{i=1}^n \tau_{ik} c_{ij}|}{\sigma_j^2} \leq \lambda, & \text{if } \hat{\mu}_{kj} = 0 \end{cases}.$$

With an additional weight term w_{kj} as defined in (2.5), the mean component update is different for each variable and cluster. In the r^{th} step we update μ_{kj} by

$$\hat{\mu}_{kj}^{(r+1)} = \frac{\sum_{i=1}^n \hat{\tau}_{ik}^{(r)} c_{ij}}{\sum_{i=1}^n \hat{\tau}_{kj}^{(r)}} \left(1 - \frac{\lambda w_{kj} \hat{\sigma}_j^{2,(r)}}{|\sum_{i=1}^n \hat{\tau}_{kj}^{(r)} c_{ij}|} \right)_+, \tag{2.8}$$

where $(x)_+$ is an indicator function such that

$$(x)_+ = \begin{cases} x, & \text{if} \quad x > 0 \\ 0, & \text{if} \quad x \leq 0 \end{cases}.$$

In (2.8), both $\hat{\sigma}_j^{2,(r)}$ and $\hat{\tau}_{ik}^{(r)}$ are calculated from previous steps, while c_{ij} are data.

(2) Variable Penalty For the variable penalty, we modify the weight applied to each variable and cluster. Specifically, for the j^{th} variable, if there exist $k_1, k_2, \cdots, k_r$, such that $|\hat{\mu}_{k_1 j}| = \cdots = |\hat{\mu}_{k_r j}| > |\hat{\mu}_{kj}|$ for $k \notin \{k_1, \cdots, k_r\}$. Then we have

$$\hat{\mu}_{kj} = \begin{cases} \tilde{\mu}_{kj}, & \text{when} \quad k \notin \{k_1, \cdots, k_r\} \\ \text{sign}(\tilde{\mu}_{kj}) \Big(\sum_{s=1}^r \frac{\tau_{\cdot k_s}}{\sum_{s=1}^r \tau_{\cdot k_s}} |\tilde{\mu}_{k_s} j| - \frac{\lambda w_{k_s j} \sigma_j^2}{\sum_{s=1}^r \tau_{\cdot k_s}} \Big)_+, & \text{when} \quad k \in \{k_1, \cdots, k_r\} \end{cases},$$

where $\tau_{\cdot k_s} = \sum_{i=1}^n \tau_{ik_s}$. Weight w_{kj} and unconstrained mean component estimate $\tilde{\mu}_{kj}$ is defined in (2.5). For variable penalty, we need to determine r, and the set $\{k_1, \cdots, k_r\}$ before the parameter estimation. Without loss of generality, r is set to be 1 in our study, so that only $\max_s(|\mu_{sj}|)$ needs to be updated for variable j.

(3) Group Penalty In group penalty case, variables are divided into M groups. FPCA introduce independency among variables, so the covariance matrix can be written as diag($\boldsymbol{V}_{k1}$,

$\boldsymbol{V}_{k2}, \cdots, \boldsymbol{V}_{kM})$ where each $\boldsymbol{V}_{km}$ is a $k_m \times k_m$ diagonal matrix. With adaptive weights, we have the sufficient and necessary condition for $\boldsymbol{\mu}_k^m$ being a unique maximizer of $l_P(\boldsymbol{\theta})$ as

$$\begin{cases} \boldsymbol{V}_{km}^{-1}\big[\sum_{i=1}^n \tau_{ik}\boldsymbol{c}_i^m - (\sum_{i=1}^n \tau_{ik})\boldsymbol{\mu}_k^m\big] = \lambda w_{k_m}\sqrt{k_m}\frac{\boldsymbol{\mu}_k^m}{||\boldsymbol{\mu}_k^m||}, & \text{if and only if } \boldsymbol{\mu}_k^m \neq \mathbf{0} \\ ||\sum_i^n \tau_{ik}\boldsymbol{c}_i^m \boldsymbol{V}_{km}^{-1}|| \leq \lambda w_{k_m}\sqrt{k_m}, & \text{if } \boldsymbol{\mu}_k^m = \mathbf{0} \end{cases}.$$

As a result, in the maximization step for group penalty, $\boldsymbol{\mu}_k^m$ is updated as

$$\hat{\boldsymbol{\mu}}_k^m = \left[\text{sign}\left(1 - \frac{\lambda w_{k_m}\sqrt{k_m}}{||\sum_{i=1}^n \tau_{kj}\boldsymbol{c}_i^m V_{km}^{-1}||}\right)\right]_+ v_k^m \tilde{\boldsymbol{\mu}}_k^m, \tag{2.9}$$

where

$$v_k^m = \left(\mathbf{I} + \frac{\lambda\sqrt{k_m}}{||\sum_{i=1}^n \tau_{kj}\boldsymbol{c}_i^m V_{km}^{-1}||}\right)^{-1}.$$

In (2.9), $\boldsymbol{c}_i^m$ are data corresponding to the m^{th} variable group while the weight is defined in (2.6).

(iv) Convergence We repeat the steps introduced in (ii) and (iii) until it converges. In this study, we set the stopping criterion as when parameter changes in two consecutive steps are negligible. Specifically,

$$||\hat{\boldsymbol{\theta}}^{r+1} - \hat{\boldsymbol{\theta}}^r|| \leq 0.0001.$$

The EM algorithm is summarized in Algorithm 1.

2.3.4 Hyperparameter Selection

In this study, there are three hyperparameters need to be chosen prior the estimation process. They are λ, γ and K. Specifically, λ and γ are for penalty terms, and K is the number of clusters. One way to choose hyperparameters is based on the adjusted BIC which is introduced in Xie et al. (2008). That is,

$$\text{BIC} = -\log[L(\hat{\boldsymbol{\theta}})] + \log(n)d_e, \tag{2.10}$$

Algorithm 1 EM algorithm

procedure EM($\boldsymbol{c}_i, \pi_k, \boldsymbol{\mu}_k, \Sigma$)
  for $r = 1$ **do** ▷ Start iteration with r being the iteration index
    Assign initial values to parameter set $\pi_k, \boldsymbol{\mu}_k.\Sigma$ ▷ Apply KNN
    Update π_k^{r+1}
    Update $\boldsymbol{\mu}_k^{r+1}$
    Update Σ^{r+1} ▷ Assume the same Σ for all clusters
  end for
  while $r > 1$ and have not converge **do**
    for `i = 1,2,`$\cdots$`,n` **do**
      Estimate δ_{ik} as τ_{ik} by Gaussian distribution $f_k(\boldsymbol{\mu}^r, \Sigma^r)$ ▷ Using Bayes rule
    end for
    Update π_k^{r+1}
    for `k = 1,2,`$\cdots$`,K` **do**
      Update $\boldsymbol{\mu}_k^{r+1}$
    end for
    Update Σ^{r+1}
  end while
  return $\pi_k, \boldsymbol{\mu}, \Sigma, \tau$ ▷ Final parameter estimations are returned
end procedure

where d_e is the effective degrees of freedom, i.e., $d_e = K + p + K \times p - 1 - n_0$. Here n_0 is the number of mean component who are forced to be zero.

In literature, choices of λ and K are well studied while the hyperparameter γ in the weight is usually set to be 1. In order to use more flexible weights in our study, all three hyperparameters are iteratively used and adjusted BIC are calculated by the converged EM algorithm. The hyperparameter combination with minimum adjusted BIC is chosen to use in the clustering. It's easy to see from (2.10) that when there are more mean components set to be zero, adjusted BIC is becoming smaller. Since using more clusters and variables will always lead to larger log-likelihood, and penalty terms control the cluster and variable numbers. The chosen hyperparameter reaches balance between log-likelihood and penalty terms.

2.4 Simulation Study

In this section, we conduct simulations on various scenarios to evaluate performances of both clustering and variable selections. Parameters used to generate functional data are referred as

true parameters. True parameter values include signal noise ratio, signal strength and number of cluster. The signal noise ratio is reflected by different numbers signal and noisy sensors. Signal sensors contribute to the clustering procedure, while noisy sensors do not. Variances of B-spline coefficients are used to adjust the signal strength. Here B-spline coefficients are comparable to c_{ij} as introduced in previous section. Coefficient means and variances similar to $\boldsymbol{\mu}_k$ and Σ are used to characterized functional data. In this chapter, three clusters with equal cluster proportions are used. We conduct clustering and variable selection under the optimal hyperparameters chosen by the adjusted BIC. Estimated results are compared with true values and model performances with different penalties are discussed.

2.4.1 Setup

For both signal sensors and noisy sensors, we use a B-spline with 2 bases and order of 2 to simplify the setting. Domains of all simulated sensory data are from 0 to 30 where there are 31 measurements on each unit. With the same basis functions, 2 coefficients are used for each sensor. In this study, we denote coefficients as $\boldsymbol{\beta} = (\beta_{s1}, \beta_{s2}, \cdots, \beta_{n1}, \beta_{n2}, \cdots)'$, where β_{sy} and β_{nz} are coefficients for signal and noisy sensors, respectively. Here $y = 1, \cdots, 2n_s$ and $z = 1, \cdots, 2n_n$ where n_s and n_n are number of signal and noisy sensors. We use the mixture Gaussian distributions to generate $\boldsymbol{\beta}$ and the pdf is

$$g(\boldsymbol{\beta}) = \sum_{k=1}^{K} \pi_{\beta,k} f_k(\boldsymbol{\beta}; \boldsymbol{\mu}_{\beta,k}, \Sigma_\beta).$$

In the simulation study, we use equal proportions where $\pi_{\beta,k} = 1/3$ for $k = 1, 2$ and 3. Covariance matrices $\Sigma_\beta = \text{diag}(\sigma_{\beta_{s1}}, \sigma_{\beta_{s2}}, \cdots, \sigma_{\beta_{n1}}, \sigma_{\beta_{n2}}, \cdots)$ are diagonal and the same across different clusters. Here $\sigma_{\beta_{s1}}$ and $\sigma_{\beta_{s2}}$ are coefficient variances for signal sensors while $\sigma_{\beta_{n1}}$ and $\sigma_{\beta_{n2}}$ are noisy sensor coefficient variances. We only apply one set of coefficient variance for sensors of the same kind. Mean components $\boldsymbol{\mu}_{\beta,k}$ are set different for each cluster. Specifically, for signal sensors, we use different coefficients for each cluster, while all 0 coefficients are applied to the noisy sensors means in all cluster. In an example where there are three clusters

and 2 sensors for both signal and noisy, coefficient mean components are

$$\boldsymbol{\mu}_{\beta,1} = (-1,-1,-1,1,0,0,0,0)^{'},$$

$$\boldsymbol{\mu}_{\beta,2} = (1,1,1,-1,0,0,0,0)^{'},$$

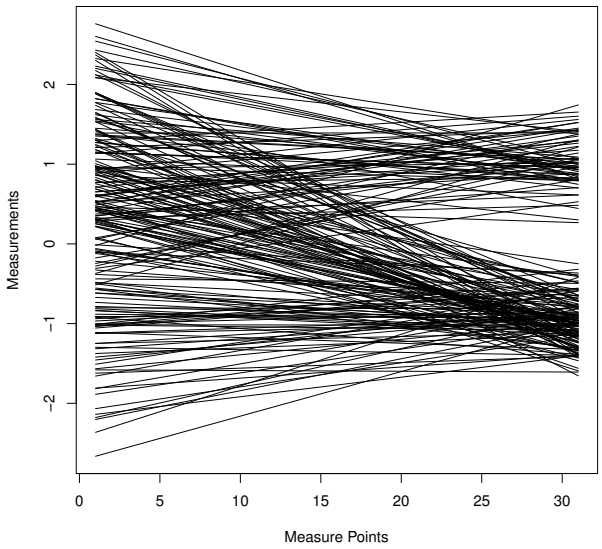
$$\boldsymbol{\mu}_{\beta,3} = (1,-1,1,1,0,0,0,0)^{'}.$$

For the mean component in the first cluster $(-1,-1)^{'}$ and $(-1,1)^{'}$ are coefficient means from two signal sensors while $(0,0)^{'}$ are from noisy sensors.

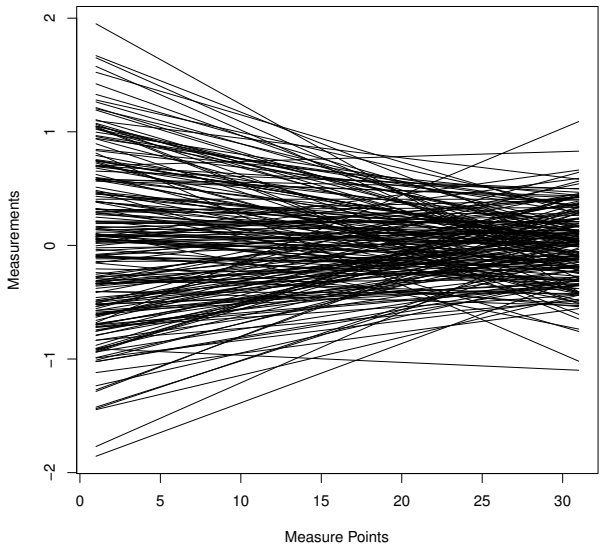
In this chapter, we investigate the effect of sample sizes, noisy signal ratio and signal strengths. In terms of sample size, we use $n = 50$, 100, 200 and 500 to generate data, while $n_s = n_n = 2$. We apply the same signal strength where the coefficient variances are $(\sigma_{\beta_{s1}}, \sigma_{\beta_{s2}})^{'} = (\sigma_{\beta_{n1}}, \sigma_{\beta_{n2}})^{'} = (0.5, 0.1)^{'}$.

We study the signal noise ratio by changing the number of noisy sensors while the number of signal sensor are fixed. We set $n_s = 2$ while using $n_n = 2, 4, 8$ and 64. Coefficient mean components from signal sensor are the same with ones in $\boldsymbol{\mu}_{\beta,1}, \boldsymbol{\mu}_{\beta,2}$ and $\boldsymbol{\mu}_{\beta,3}$. Noisy sensors have the mean component as $(0,0)^{'}$. We use $n = 200$ and $(\sigma_{\beta_{s1}}, \sigma_{\beta_{s2}})^{'} = (\sigma_{\beta_{n1}}, \sigma_{\beta_{n2}})^{'} = (0.5, 0.1)^{'}$ for all cases. Signal sensory data in the generated data present different characteristics under each cluster, while noisy sensory data show similarities among clusters. Examples of simulated signal and noisy sensory data are shown in Figure 2.3.

Another attribute that affects the clustering and variable selection is signal strength. We investigate this by varying coefficient variances in functional data generations. Smaller coefficient variances make corresponding functional data closer together. On the other hand, larger coefficient variances generate functional data that are sparsely located. Simulated functional data with weak and strong signals are shown in Figure 2.4. In the simulation study, $(\sigma_{\beta_{s1}}, \sigma_{\beta_{s2}})^{'} = (\sigma_{\beta_{n1}}, \sigma_{\beta_{n2}})^{'}$ are changed as $(0.5, 0.1)^{'}$, $(0.25, 0.05)^{'}$, $(0.20, 0.04)^{'}$ and $(0.05, 0.01)^{'}$. We use $n_s = 2, n_n = 8$ and $n = 200$ for all cases.

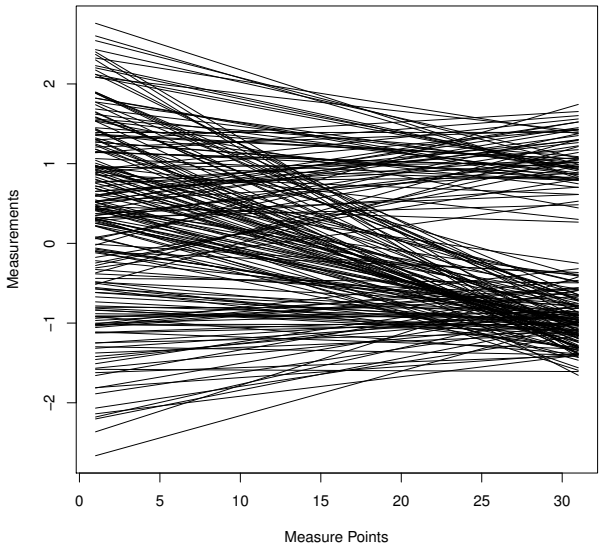


(a) Signal sensor functional data.

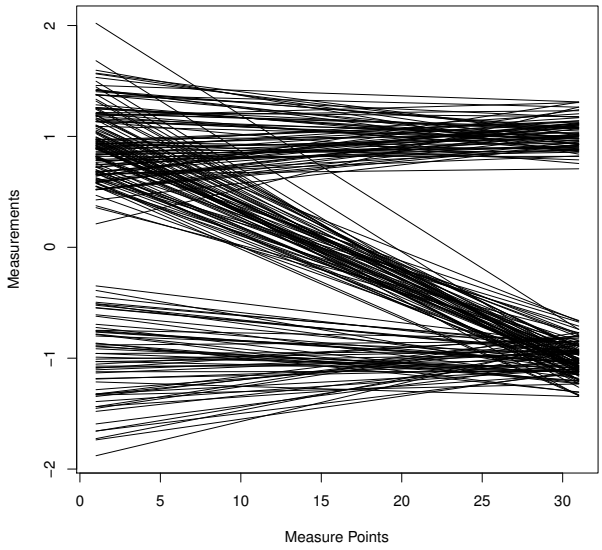


(b) Noisy sensor functional data.

Figure 2.3: Signal and noisy functional data for three clusters. In (a) Coefficient means are $(-1,-1)^{'}$, $(1,1)^{'}$ and $(1,-1)^{'}$ for each cluster. In (b), coefficient means are $(0,0)^{'}$ for all clusters. Coefficient variances are $(0.5,0.1)^{'}$ in both cases.



(a) Functional data with weak signals.



(b) Functional data with strong signals.

Figure 2.4: Examples of weak and strong signals functional data. Coefficients have means $(-1,-1)^{'}$, $(1,1)^{'}$ and $(1,-1)^{'}$ for three clusters respectively. Variances for coefficients are $(0.5,0.1)^{'}$ for all clusters.

2.4.2 Results

In this section, we present simulation results where sample size, signal noise ratio and signal strength are changed. In Table 2.1, effect of sample sizes are illustrated. For 200 simulated samples under each sample size n, we calculate the mean absolute error (MAE) as

$$\frac{1}{200}\sum_{i=1}^{200}|\hat{K}_i - K_{True}|,$$

where $\hat{K}_i$ the estimated number of clusters for the i^{th} sample and K_{True} is the true number of clusters. In this study, K_{True} is set to 3 across different setups. In addition, we calculate the mean number of variables and sensors removed. Variables from signal sensors are considered as removed falsely if they are identified as noises. In Table 2.1, with larger sample sizes, MAEs of estimated K become smaller. In addition, number of removed sensors are closer to the true value (i.e.,2) when the sample size increases. Among three penalty terms, variable penalty outperforms the other 2 in terms of estimated K and removed sensors. In the variable selection procedure, both variables have to be removed in order to remove the corresponding sensor. Group penalty term has a better performance in sensor selection compare to individual penalty, even though variable selections performances are similar.

In Table 2.2, we show clustering results under different signal noise ratios. Variable penalty has the best sensor selection performance, while it removes the most number of noises. Group and individual penalties are not as good in terms of the correct number of sensor removed. Given this particular setup, methods with all penalty terms are under penalized with no falsely removed sensors in the simulation study except for one dataset. With more noisy sensors included, numbers of sensors selected are deviating from the true value. Other than sample size and signal noise ratio, estimated results are affected by signal strength as well.

Table 2.3 shows that stronger signals return better clustering results. When coefficient variances get smaller, MAEs of estimated K are decreasing. In addition, mean number of sensor removed are closer to the true value for all penalty terms. In the strong signal case,

Table 2.1: MAEs of estimated K, average numbers of removed variables and sensors under sample sizes 50, 100, 200 and 500. For each simulated dataset, optimal hyperparameters are used to evaluate clustering results.

n \ Evaluation	Penalty	K MAE	Variable removed	Sensor removed	Variable removed falsely
50	Individual	0.32	2.9	0.98	0.01
	variable	0.225	3.3	1.29	0.02
	group	0.27	1.9	0.95	0
100	Individual	0.21	3.08	1.16	0
	variable	0.135	3.60	1.61	0
	group	0.165	2.87	1.44	0
200	Individual	0.17	3.39	1.43	0
	variable	0.085	3.79	1.79	0
	group	0.12	3.41	1.71	0
500	Individual	0.17	3.60	1.62	0
	variable	0.10	3.87	1.87	0
	group	0.115	3.66	1.83	0

methods with variable and group penalties still outperform the individual penalty. There is no falsely removed signal sensor.

EM algorithm used in the clustering estimation assigns labels randomly each time. When comparing estimated results towards true ones, (A, A, B, C, C, C) and (C, C, A, B, B, B) can be falsely considered as different clustering results, while they are actually the same. One remedy is to evaluate the clustering precision based on the adjusted Rand index (ARI). ARI is a non-parametric correlation estimation, where higher similarity between two variables returns an ARI closer to 1. In the above example clustering result, suppose (A, A, B, C, C, C) is the estimated results and (C, C, A, B, B, B) is the true value. ARI between these is 1, correctly indicating an identical results. In simulation studies, using ARI as our criterion enables us to compare estimated and true clustering labels more precisely. Figure 2.5, 2.6 and 2.7 shows ARI with different sample sizes, number of noisy sensors and coefficient variances, respectively. On the other hand, stronger signals lead to better clustering results in terms of ARI. Specifically, Figure 2.5 indicates that larger sample size leads to higher ARI. Figure 2.6 shows that with a fix number of signal sensors, more noisy sensors will reduce the ARI. In Figure 2.7, we see ARI estimated from strong signal sensors are higher than those from weak

Table 2.2: With different number of noisy sensors, MAEs of estimated K, mean number of removed variables and sensors are estimated with optimal hyperparameters in 200 samples. Coefficients variance are set to be $(\sigma_{\beta_{s1}}, \sigma_{\beta_{s2}})' = (\sigma_{\beta_{n1}}, \sigma_{\beta_{n2}})' = (0.5, 0.1)'$ for all sensors. Number of signal sensors is $n_s = 2$.

n_n \ Evaluation	Penalty	K MAE	Variable removed	Sensor removed	Variable removed falsely
2	Individual	0.17	3.39	1.43	0
	Variable	0.085	3.79	1.79	0
	Group	0.12	3.41	1.71	0
4	Individual	0.24	6.59	2.69	0
	Variable	0.165	7.35	3.37	0
	Group	0.215	6.52	3.26	0
8	Individual	0.39	12.8	5.1	0.005
	Variable	0.25	14.3	6.3	0
	Group	0.285	12.2	6.1	0
64	Individual	0.78	107.5	45.2	0
	Variable	0.84	114.9	51.1	0
	Group	0.76	96.7	48.4	0

Table 2.3: Clustering results from individual, variable and group penalties with different coefficient variances. MAEs of estimated K, mean numbers of removed variables and sensors are estimated under optimal hyperparameters for 200 samples. We use $n_s = 2$ and $n_n = 8$ for all setups.

Strength \ Evaluation	Penalty	K MAE	Variable removed	Sensor removed	Variable removed falsely
(0.5, 0.1)	Individual	0.39	12.8	5.1	0.005
	Variable	0.25	14.3	6.3	0
	Group	0.285	12.2	6.1	0
(0.25, 0.05)	Individual	0.255	14.0	6.1	0
	Variable	0.175	15.4	7.4	0
	Group	0.15	14.7	7.4	0
(0.20, 0.04)	Individual	0.17	14.5	6.5	0
	Variable	0.125	15.6	7.6	0
	Group	0.14	15.1	7.5	0
(0.05, 0.01)	Individual	0.04	15.3	7.3	0
	Variable	0.01	15.9	7.9	0
	Group	0.01	15.8	7.9	0

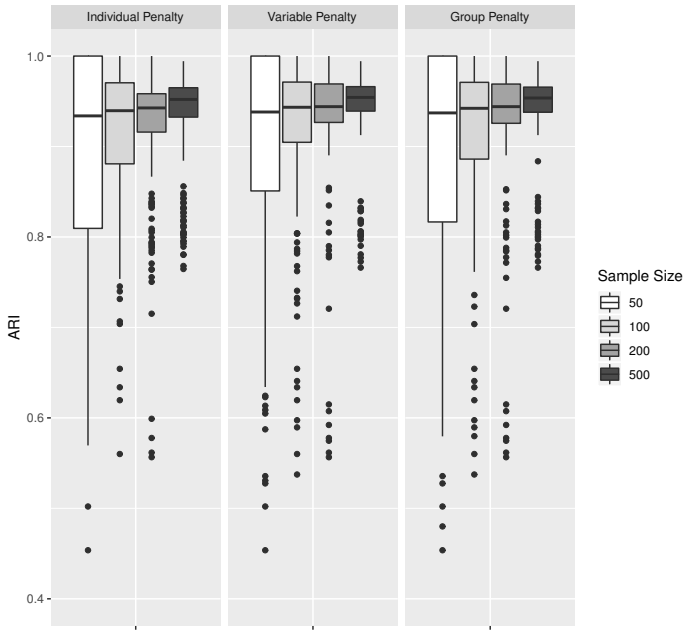


Figure 2.5: ARI under sample sizes 50, 100, 200 and 500 with individual, variable and group panelties.

signal sensors. ARI can eventually be translated into clustering precision. An ARI that is closer to 1 indicates an almost perfect clustering. Among three penalty terms, ARI estimates from variable and group penalties are consistently better than ones from individual penalty, where ARI results between variable and group penalties are similar. This conclusion is more obvious when n is large, n_n is small and signal strength is strong.

2.5 Application

In this section, we apply the proposed method on the engineering sensor data. Methods with different penalty terms are employed under hyperparameters chosen by the adjusted BIC. Estimation procedures based on individual, variable and group penalties are shown as follows. In this system, 65 sensors are installed, and status of the system are recorded on various locations. For instance, there are sensors to monitor engine and system temperatures. In addition, some sensors are set to measure operating speeds of the system. The engineering system with active sensors produces a multivariate functional data with a total of 65 columns

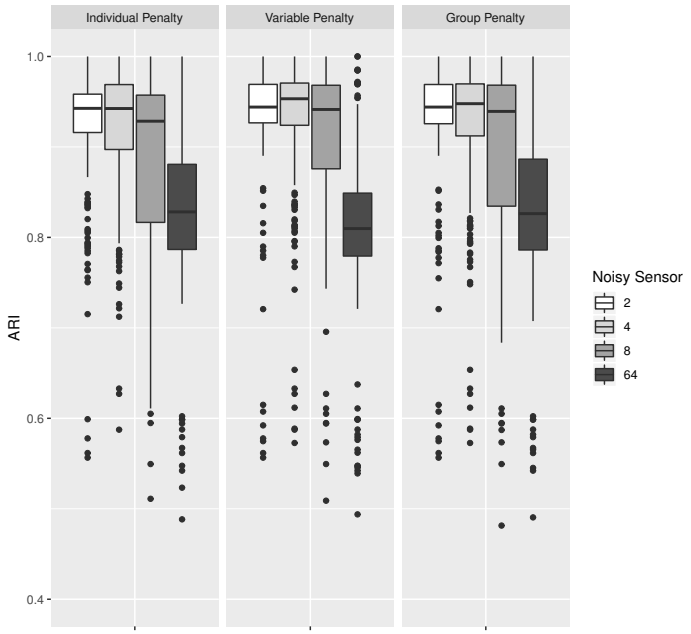


Figure 2.6: ARI under individual, variable and group penalties for $n_n = 2, 4, 8$ and 64, while $n_s = 2$ for all cases.

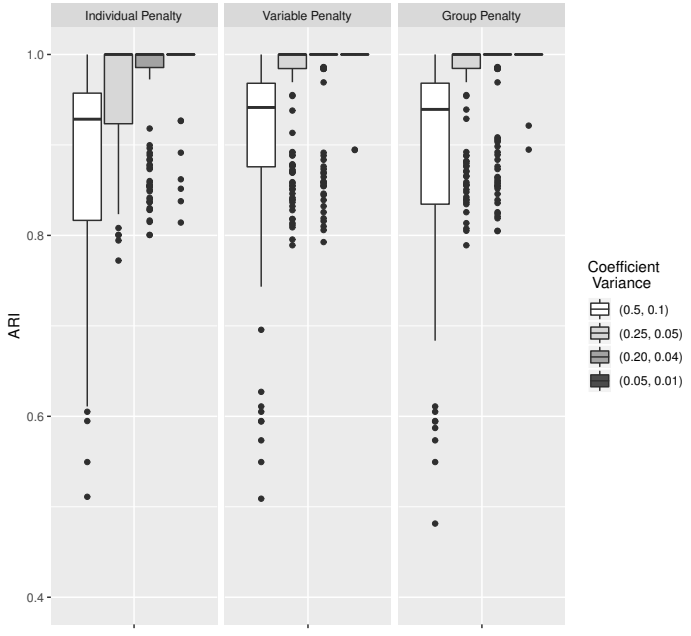


Figure 2.7: ARI under different coefficient variances (i.e., $(\sigma_{\beta_{s1}}, \sigma_{\beta_{s2}})'$ and $(\sigma_{\beta_{n1}}, \sigma_{\beta_{n2}})'$) with individual, variable and group panelties.

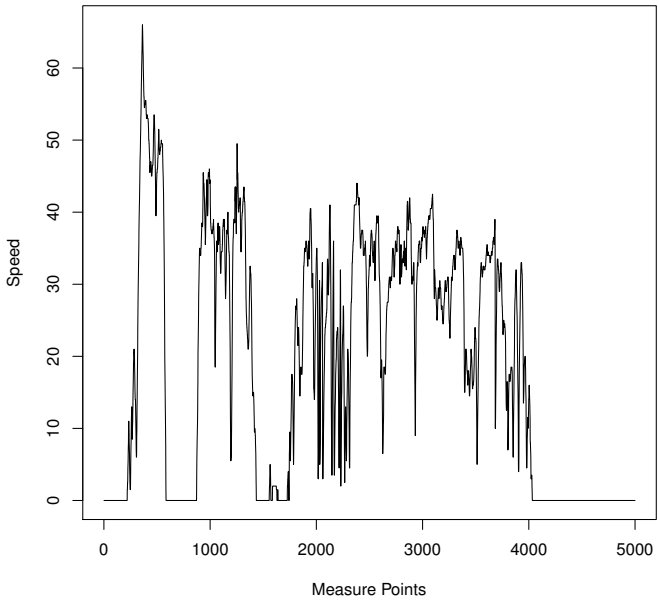


Figure 2.8: First 5,000 measurements on the operation speed sensor from engineering sensory data example.

and 125,980 rows. Each column represents one sensor, while each row stands for one measurement. For each sensor, it measures the corresponding status on average of 2 times per second. In Figure 2.8, we show the first 5,000 measure points for the sensor recording system operation speed in a 2.5 hours span. Out of 125,980 measurement points, there are 419 stoppage events, where each stop is in a 30×65 matrix. This engineering sensory data allows us to conduct the clustering on stoppages so that various stoppage characteristics can be further studied. In Figure 2.9, we show four additional sensors' measurements. Some of sensors are related to operation speeds, while the other sensors are systematic status that have no correlations with system stops. In order to apply the proposed method, we stack this multi-dimensional sensory measurements where stoppages are the last measurement point for each observation. Data cleaning also include removing sensors whose measurement have no variation. In the engineering sensory data analysis, we remove 18 sensors and use the rest to conduct clustering.

With the prepared data suitable to our methods, we choose hyperparameters to use for individual, variable and group penalties, respectively. Different K, λ and γ are iterated, and

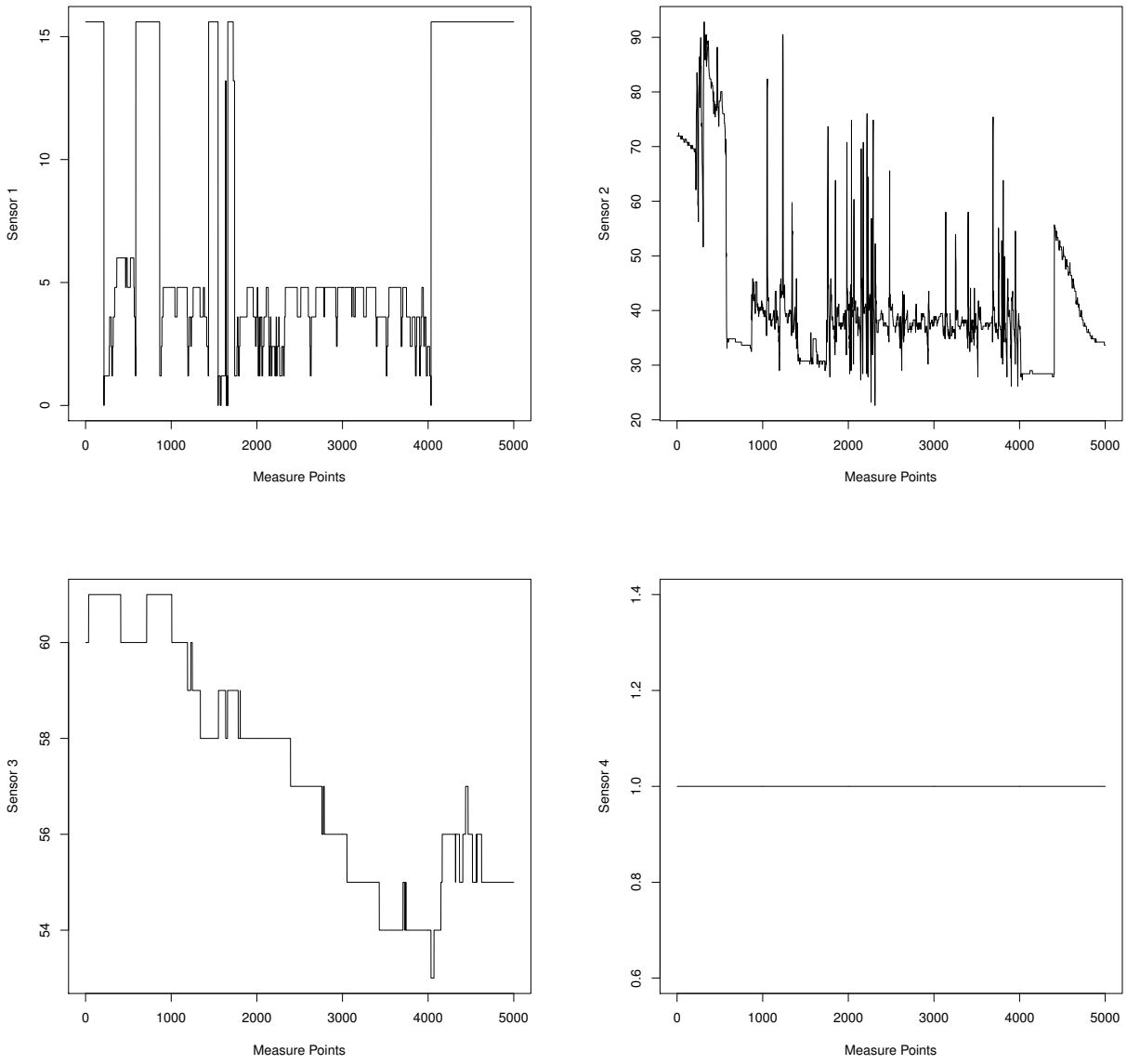


Figure 2.9: Four different sensor measurements for the first 5000 measurement points. The plots on the top rows are related to the event, while plots on the bottom row are noises.

Table 2.4: Clustering results for engineering sensor data from individual, variable and group penalties. Each row represents number of observations in each clusters for a certain penalty term.

Penalty	Cluster 1	Cluster 2	Cluster 3
Individual	158	63	198
Variable	160	63	196
Group	161	63	195

one (K, λ, γ) combination with the smallest adjusted BIC are selected. As a result, individual penalty with ($K = 3$, $\lambda = 0.05$, $\gamma = 5$), variable penalty with ($K = 3$, $\lambda = 0.5$, $\gamma = 5$) and group penalty with ($K = 3$, $\lambda = 0.05$, $\gamma = 10$) are hyperparameters we use in the data analysis. Based on the chosen parameters, clustering results are similar among all three penalties. Number of observations in each cluster is shown in Table 2.4. Since estimated labels are similar among three penalties, we visualize results from one penalty. Using the clustering result by variable penalty, we assign colors to different clusters and present them in Figure 2.10. Specifically, green lines indicate hard breaks where speeds are dropped drastically fast shortly before stoppages. Blacks lines are prepared stoppage, it includes stoppages where speeds are reduced gradually. Stoppages denoted by red lines illustrate slow speeds.

Variable selection results for the three penalties are quiet different. In this study, we use two principal component coefficients to represent one sensor. In order to remove a sensor in variable selection, we have to remove both coefficient variables corresponding to the sensor. Number of sensors removed are 10, 14 and 17, respectively for individual, variable and group penalties. We present some variables that are selected by all three penalties in Figure 2.11. They are engine speed, exhale temperature, longitudinal sensor and MAP vacuum. In Figure 2.12, we show sensory plot who are removed by all penalties. Specifically, they are heater temperature and lateral acceleration. Sensors in Figure 2.11 shows similar patterns within same clusters, while Figure 2.12 exhibits no group features.

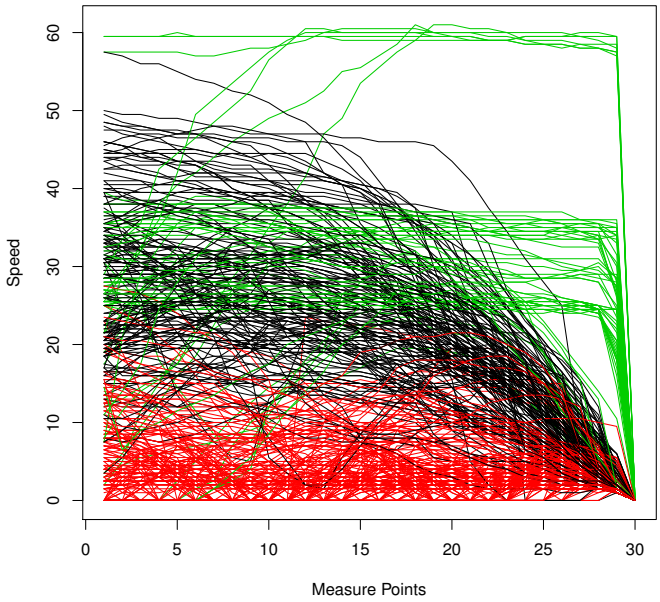
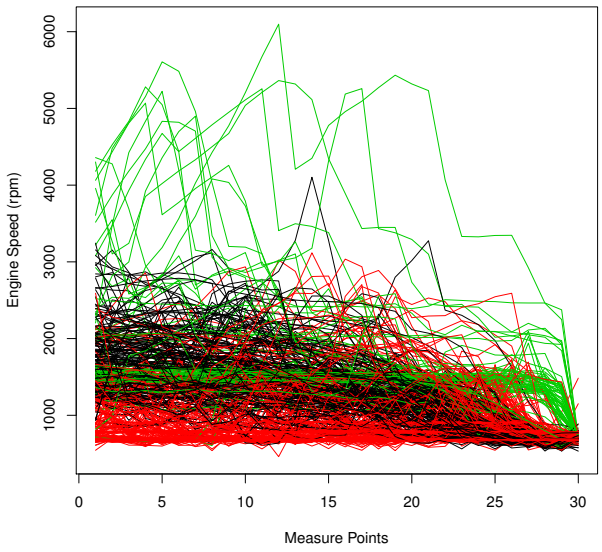


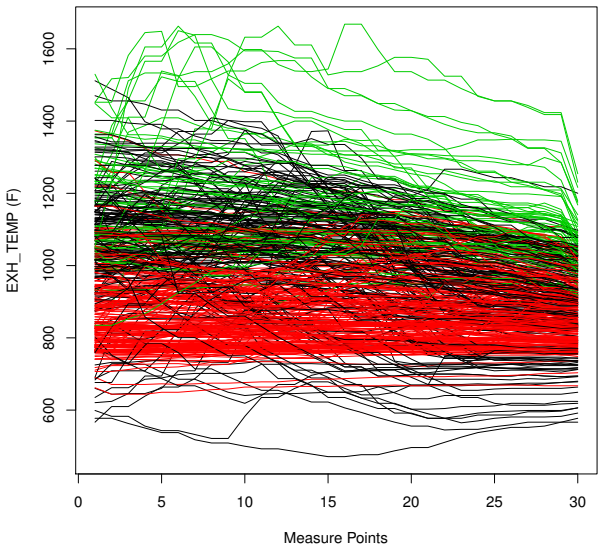
Figure 2.10: Clustering results of 419 distinct stops for the speed sensor in engineering sensory data. Red, green and black lines represent three different clusters respectively.

2.6 Conclusion and Remarks

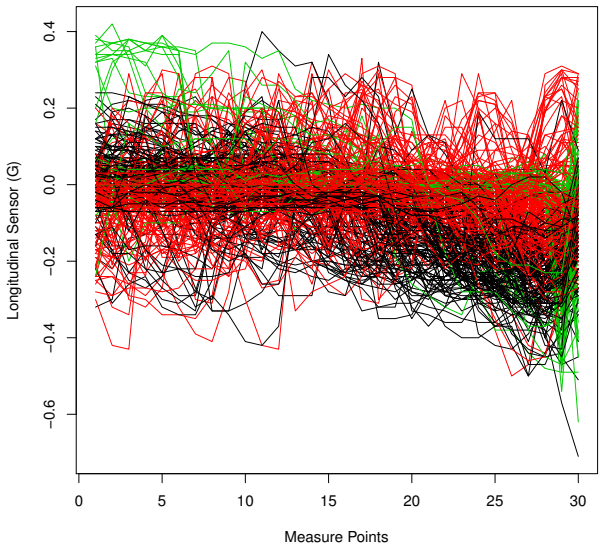
In this chapter, we introduce a method to conduct clustering on multivariate functional data. An important feature is this method also performs automatic variable selection. Three penalty terms are introduced including individual, variable and group penalties. Their performances are discussed in simulation studies. Clustering results are evaluated by ARI and variable selection precision. Among three penalties introduced, variable penalty has the best performance in variable selection and clustering. Group penalty is slightly worse compared to variable penalty, while individual penalty is the worst. Advantages of group penalties can be explained by making use of the coefficient homogeneity introduced by FPCA. The reason that variable penalty performs the best is because it considers variable similarities among different clusters. The usage of weight term enhances the flexibility in penalizations. It is novel to choose λ and γ together with number of clusters K in this functional clustering problem with variable selection.



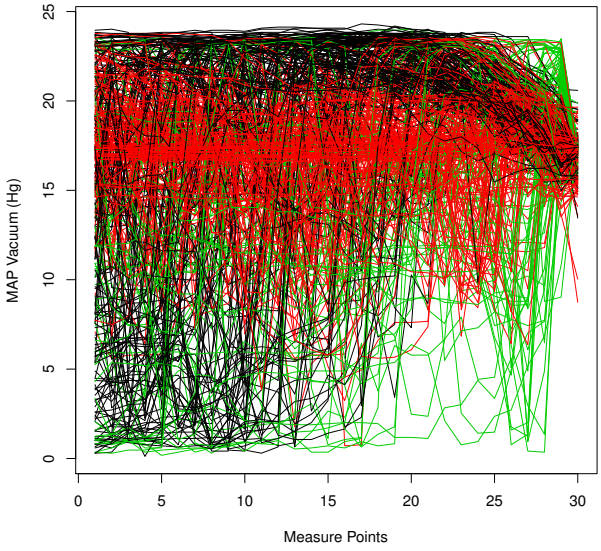
(a) Engine Speed.



(b) Exhale Temperature.

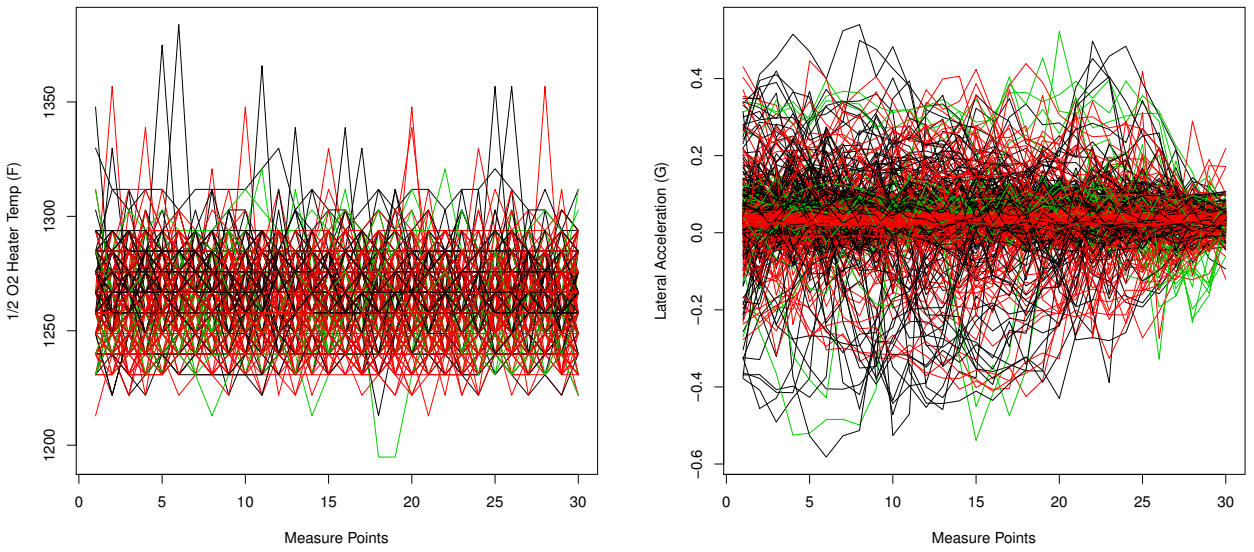


(c) Longitudinal measure.



(d) MAP Vacuum.

Figure 2.11: Examples of selected sensors based on all penalty terms.



(a) Heater temperature.

(b) Lateral acceleration.

Figure 2.12: Examples of removed sensors by all penalties.

A sensory data application is considered in this chapter, where multi-dimensional sensory measurements are observed in an engineering system. However, applications on this method is not restricted to sensory data only. Our method applies to any multi-dimensional continuous measurement who has need of clustering with variable selection. For example, implantable body sensors are used to collect multi-dimensional sensory data, from which life threatening events can be identified. In Van Laerhoven et al. (2004), only K-means clustering is employed regarding this online obtained data, while our method can enhance the clustering by utilizing information thoroughly. In addition, indoor environmental safety issues occur more frequently because of the increase usage of home appliances. Multi-sensor system is applied to monitor possible emergencies such as fire and smoke appearances (Wu and Clements-Croome, 2007). Our method can be applied on this multi-dimensional sensory data to recognize various risks at homes.

Selected variables highlight important factors contributing in the clustering process. In an unsupervised clustering setup, variable selection improves the computational efficiency. Selected sensors can be used in the initial screening process of a sensor scheduling and selection.

As a result, sensor utilizations are economically efficient when we focus on the informative sensors with limited resources.

Appendix A

Mean Component Estimates for an Unconstrained Log-likelihood

Here we only derive the calculation of $\tilde{\mu}_{kj}$ and $\tilde{\boldsymbol{\mu}}_k^m$ is simply an vector extension of it. When there is no constraint on the likelihood, let $\tilde{\mu}_{kj}$ be the maximizer for (2.3) and $\tilde{\mu}_{kj}$ satisfies

$$\left.\frac{\partial l(\boldsymbol{\theta})}{\partial \mu_{kj}}\right|_{\mu_{kj}=\tilde{\mu}_{kj}} = 0.$$

Here we calculate the partial derivative in a vector version μ_k and mean component of any j can be easily applied.

$$\begin{aligned}
\frac{\partial l(\boldsymbol{\theta})}{\partial \boldsymbol{\mu}_k} &= \frac{\partial}{\partial \boldsymbol{\mu}_k}\sum_{i=1}^{n}\delta_{ik}\Big\{\log(\pi_k) + \log\big[f_k(\boldsymbol{c}_i;\boldsymbol{\mu}_k,\Sigma)\big]\Big\} \\
&= -\sum_{i=1}^{n}\delta_{ik}\frac{1}{f_k(\boldsymbol{c}_i;\boldsymbol{\mu}_k,\Sigma)}\Sigma^{-1}(\boldsymbol{c}_i-\boldsymbol{\mu}_k)\frac{1}{(2\pi)^{\frac{q}{2}}|\Sigma|^{\frac{1}{2}}}\exp\big[-\frac{1}{2}(\boldsymbol{c}_i-\boldsymbol{\mu}_k)'\Sigma^{-1}(\boldsymbol{c}_i-\boldsymbol{\mu}_k)\big] \\
&= -\sum_{i=1}^{n}\delta_{ik}\Sigma^{-1}(\boldsymbol{c}_i-\boldsymbol{\mu}_k). \qquad (2.11)
\end{aligned}$$

With a non-singular covariance matrix Σ , we obtain the estimator of $\boldsymbol{\mu}_k$ by letting (2.11) equal to zero and multiply Σ on both sides of the equation. That is,

$$\sum_{i=1}^{n}\delta_{ik}(\boldsymbol{c}_i-\boldsymbol{\mu}_k) = \mathbf{0}.$$

Then estimates for the mean component from cluster k is

$$\tilde{\boldsymbol{\mu}}_k = \frac{\sum_{i=1}^{n}\delta_{ik}\boldsymbol{c}_i}{\sum_i^n \delta_{ik}},$$

where dimension j and group m are $\tilde{\mu}_{kj}$ and $\tilde{\boldsymbol{\mu}}_k^m$ as in (2.5) and (2.6), respectively.

Calculate Covariance Matrix in the EM Algorithm

Similar to the calculation in Appendix 2.6, $\hat{\sigma}_j^2$ are calculated by taking partial derivative to the penalized log-likelihood function with respective to Σ. The maximizer is the parameter who makes this partial derivative to be zero.

$$\begin{aligned}
\frac{\partial l_P(\boldsymbol{\theta})}{\partial \Sigma} &= \frac{\partial}{\partial \Sigma}\left\{ \sum_{i=1}^{n}\sum_{k=1}^{K} \delta_{ik}\left\{ \log(\pi_k) + \log\left[f_k(\boldsymbol{c}_i;\boldsymbol{\mu}_k,\Sigma)\right]\right\} - p_\lambda(\boldsymbol{\theta})\right\} \\
&= \sum_{i=1}^{n}\sum_{k=1}^{K} \delta_{ik}\frac{1}{f_k(\boldsymbol{c}_i;\boldsymbol{\mu}_k,\Sigma)}\frac{1}{(2\pi)^{\frac{q}{2}}|\Sigma|^{\frac{3}{2}}}\exp\big[-\frac{1}{2}(\boldsymbol{c}_i-\boldsymbol{\mu}_k)'\Sigma^{-1}(\boldsymbol{c}_i-\boldsymbol{\mu}_k)\big]|\Sigma|\Sigma^{-1} - \\
&\quad \sum_{i=1}^{n}\sum_{k=1}^{K} \delta_{ik}\frac{1}{(2\pi)^{\frac{q}{2}}|\Sigma|^{\frac{1}{2}}}\exp\big[-\frac{1}{2}(\boldsymbol{c}_i-\boldsymbol{\mu}_k)'\Sigma^{-1}(\boldsymbol{c}_i-\boldsymbol{\mu}_k)\big]\Sigma^{-1}(\boldsymbol{c}_i-\boldsymbol{\mu}_k)(\boldsymbol{c}_i-\boldsymbol{\mu}_k)'\Sigma^{-1} \\
&= \sum_{i=1}^{n}\sum_{k=1}^{K} \delta_{ik}\Sigma^{-1}\Big[1-(\boldsymbol{c}_i-\boldsymbol{\mu}_k)(\boldsymbol{c}_i-\boldsymbol{\mu}_k)'\Sigma^{-1}\Big]. \qquad (2.12)
\end{aligned}$$

According to the definition, $\sum_{i=1}^{n}\sum_{k=1}^{K}\delta_{ik} = n$, total number of observations. Let (2.12) to be $\mathbf{0}$, we have the j^{th} diagonal element in Σ calculated as

$$\hat{\sigma}_j^2 = \sum_{k=1}^{K}\sum_{i=1}^{n}\delta_{ik}(c_{ij}-\mu_{kj})^2/n.$$

Bibliography

J. R. Berrendero, A. Justel, and M. Svarc. Principal components for multivariate functional data. *Computational Statistics and Data Analysis*, 55:2619–2634, 2011.

C. Bouveyron and C. Brunet-Saumard. Model-based clustering of high-dimensional data: A review. *Computational Statistics and Data Analysis*, 71:52–78, 2014.

T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13:21–27, 1967.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39:1–38, 1977.

R. L. Eubank. *Nonparametric regression and spline smoothing*. CRC press, 1999.

C. Happ and S. Greven. Multivariate functional principal component analysis for data observed on different (dimensional) domains. *Journal of the American Statistical Association*, pages 1–11, 2018.

B. Henderson. Exploring between site differences in water quality trends: a functional data analysis approach. *Environmetrics: The official journal of the International Environmetrics Society*, 17:65–80, 2006.

A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

J. Jacques and C. Preda. Model-based clustering for multivariate functional data. *Computational Statistics and Data Analysis*, 71:92–106, 2014.

S. D. Kamath and K. K. Mahato. Principal component analysis (PCA)-based k-nearest neighbor (K-NN) analysis of colonic mucosal tissue fluorescence spectra. *Photomedicine and Laser Surgery*, 27:659–668, 2009.

Y. Li, N. Wang, and R. J. Carroll. Selecting the number of principal components in functional data. *Journal of the American Statistical Association*, 108:1284–1294, 2013.

N. Locantore, J. Marron, D. Simpson, N. Tripoli, J. Zhang, K. Cohen, G. Boente, R. Fraiman, B. Brumback, C. Croux, et al. Robust principal component analysis for functional data. *Test*, 8:1–73, 1999.

B. D. Marx and P. H. Eilers. Direct generalized additive modeling with penalized likelihood. *Computational Statistics and Data Analysis*, 28:193–209, 1998.

P. D. McNicholas and T. B. Murphy. Model-based clustering of microarray expression data via latent gaussian mixture models. *Bioinformatics*, 26:2705–2712, 2010.

X. Nguyen and A. E. Gelfand. The dirichlet labeling process for clustering functional data. *Statistica Sinica*, pages 1249–1289, 2011.

U. Ozguner, C. Stiller, and K. Redmill. Systems for safety and autonomous behavior in cars: The darpa grand challenge experience. *Proceedings of the IEEE*, 95(2):397–412, 2007.

V. M. Panaretos, S. Tavakoli, et al. Fourier analysis of stationary time series in function space. *The Annals of Statistics*, 41:568–603, 2013.

C. Park, M. C. Wang, and E. B. Mo. Probabilistic penalized principal component analysis. *Communications for Statistical Applications and Methods*, 24:143–154, 2017.

J. O. Ramsay and B. W. Silverman. *Applied functional data analysis: methods and case studies*. Springer, 2007.

L. Tang, H. Pan, and Y. Yao. K-nearest neighbor regression with principal component analysis

for financial time series prediction. In *Proceedings of the 2018 International Conference on Computing and Artificial Intelligence*, pages 127–131. ACM, 2018.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

K. Van Laerhoven, B. P. Lo, J. W. Ng, S. Thiemjarus, R. King, S. Kwan, H.-W. Gellersen, M. Sloman, O. Wells, P. Needham, et al. Medical healthcare monitoring with wearable and implantable sensors. In *Proc. of the 3rd International Workshop on Ubiquitous Computing for Healthcare Applications*, 2004.

R. Viviani, G. Grön, and M. Spitzer. Functional principal component analysis of fMRI data. *Human Brain Mapping*, 24:109–129, 2005.

H. Wang and C. Leng. A note on adaptive group lasso. *Computational Statistics & Data Analysis*, 52(12):5277–5286, 2008.

S. Wang and J. Zhu. Variable selection for model-based high-dimensional clustering and its application to microarray data. *Biometrics*, 64:440–448, 2008.

S. N. Wood. Modelling and smoothing parameter estimation with multiple quadratic penalties. *Journal of the Royal Statistical Society*, 62:413–428, 2000.

S. Wu and D. Clements-Croome. Understanding the indoor environment through mining sensory dataxa case study. *Energy and Buildings*, 39(11):1183–1191, 2007.

B. Xie, W. Pan, and X. Shen. Penalized model-based clustering with cluster-specific diagonal covariance matrices and grouped variables. *Electronic Journal of Statistics*, 2:168, 2008.

M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society*, 68:49–67, 2006.

H. Zou. The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006.

Chapter 3 Covariate Adjusted Recurrent Processes for Bivariate Systems and an Application to Geyser Eruption Prediction

Abstract

Geyser eruption is one of the signature attractions at Yellowstone National Park. With the motivation to predict next eruption times for geyser groups, we propose parametric models for eruption gap times with covariate adjustment. In this chapter, we discuss a general recurrent event process with two event types. A bivariate lognormal distribution and Gumbel copula with different marginal distributions are implemented to accommodate for the dependence of various types of failure events. We apply the maximum likelihood method to estimate model parameters. Based on the developed method, predictions of event time and event type given all failure history and covariates information are obtained. A comprehensive simulation study is conducted to verify the performance of the developed methodology. Proposed models are illustrated by application to the Yellowstone geyser eruptions data.

Key Words: copula, competing risks, event dependence, geyser eruptions, recurrent events.

3.1 Introduction

Geyser eruption is one of the signature attractions in Yellowstone National Park. Tourists around the world crave to witness this natural phenomenon which is hard to encounter, since eruption time for some geysers are hard to predict. In order to predict geyser eruptions, we introduce a recurrent process which is used to model events that can repeatedly occur over time. Recurrent processes are widely observed in many areas. Examples include vehicle failures in warranty study (Lawless, 1995), relapse biomarkers in cancer research (Schaubel and Cai, 2004), and sports injury analysis (Ullah et al., 2014). Interval/gap times between two consecutive events are used to capture the characteristic of failure event frequencies. Tremendous loss can be caused by mechanical failures, therefore it is of financial benefits to get ready for the events beforehand by predicting the next event time and type.

Traditionally, proportional intensity models (Cox, 1972, and Andersen and Gill, 1982) are used to quantify gap times resulted from a single caused recurrent process. In more sophisticated cases, recurrent events observed from the system are in multiple types, whereas events in any type will result in system failures. Generally, in a multi-type event process, event gaps from different event types are influencing each other. One common failure dependence is that events from one type will cause events from the other types to occur more frequently.

In this chapter, we apply our model on a special recurrent event process, which consists of geyser eruptions collected from Yellowstone National Park in 2008. Whereas, failures of the system can result from failures of either components. In this case, any consecutive failures can result from different components or from the same component. In order to describe this bivariate recurrent process, we not only need to understand the marginal behavior for each component, but also we need to capture the failure dependence between components. A bivariate distributional assumption is made on the gap times among successive failures in the system. In such recurrent processes, predictions of next event times will be more dynamic, whereas gap times for each event type, as well as gap times for the system need to be quantified.

In some applications, it is necessary to make use of external information other than only

event times. In such cases, the recurrent process is affected by other systematic conditions. For example, some mechanical failures tend to happen more frequently with higher temperatures, humidity, and/or pressures. Here, we incorporate covariates into the proposed model for gap times to reflect the effect of these systematic conditions. We call this procedure as covariate adjustment.

Recurrent event processes are widely studied in the areas of reliability and public health. For instance, Lawless (1995) provides a comprehensive summary on methods regarding recurrent event processes. In this chapter, we focus on the recurrent process with bivariate event types. The variable of interest in such recurrent processes varies in different studies. For example, Dauxois and Sencey (2009) considered the risks of two nosocomial infections for patients admitted to hospitals and Bouaziz et al. (2013) provided a nonparametric method to estimate the intensity function of a recurrent process. Other than hazard functions and intensity functions, survival status in time is of interest as well. Huang and Liu (2007) studied the disease free survival rate in a recurrent heart failure data. Zeng and Lin (2009) and Garre et al. (2008) discussed terminal events in recurrent systems. Gap times of multivariate failure time data were introduced in Schaubel and Cai (2004). Yang et al. (2017) considered a parametric model for the multi-type event recurrent system in a car body process. In this chapter, we study eruption gap time variable in a two-geyser system.

In medical research, Liu and Huang (2009) presented repeated measurements of biomarker to determine the HIV survival status in a recurrent event system. Sun et al. (2006) applied covariate adjusted additive hazard model for the data, which is involving recurrent gap times. Prasad and Rao (2002) used a proportional hazard function with covariate adjustment in a repairable system. In another application of recurrent event data, Huzurbazar and Williams (2010) incorporated covariates in a flowgraph model. Yang et al. (2013) introduced multivariate lognormal assumption on event gap times of different event types. Yang et al. (2017) developed copula function on gap times in a recurrent process. Whereas, both papers didn't consider covariate information to adjust their proposed models.

Geyser eruption is rarely studied by statistical models, while research work has been con-

ducted on other natural phenomena such as flood and earthquake prediction. Wijesundera et al. (2013) used a Bayesian framework to conduct cyclone-induced flooding prediction. Ogata (2013) applied hierarchical space-time Epidemic-type aftershock sequence (ETAS) model to predict earthquakes. For geyser eruption study, Fournier (1969) built a physical model for one of the geysers in Yellowstone National Park, Old Faithful Geyser. Rinehart (1972) showed the fact that Old Faithful Geyser activity is affected by earth tidal forces, barometric pressure, and tectonic stresses. In this chapter, we develop a statistical method for modeling gap times of bivariate recurrent processes for geyser system data. The proposed model will be applied to two geyser eruptions, namely, West Triplet Geyser and Grotto Geyser, which are collected from Yellowstone National Park and have most complete record for eruptions. A prediction procedure for the next geyser eruption time is described. The developed model accommodate for covariates information and will capture the dependence among different failure types.

The rest of this chapter is organized as follows. Section 3.3 introduces model details on multi-type recurrent system with data incorporation. Section 3.4 describes the parameter estimation algorithm with examples. Simulation study is implemented in Section 3.5 with model comparisons under different parameter setups. An application on a two-geyser system is shown in Section 3.6. Section 3.7 contains some concluding remarks.

3.2 Geyser Data

We apply our model on the Yellowstone geyser eruption data. The public available data were collected in 2008 by the Geyser Observation and Study Association (GOSA). With underground sensors setup, water levels are measured continuously, and occurrences of geyser eruptions are detected automatically. For each geyser, the data include times of each eruption and durations of corresponding eruptions. In the bivariate geyser system, we merge two geysers' eruption records in temporal order under time in between June 2008 and November 2008.

In this chapter, we choose the West Triplet Geyser and the Grotto Geyser to create a

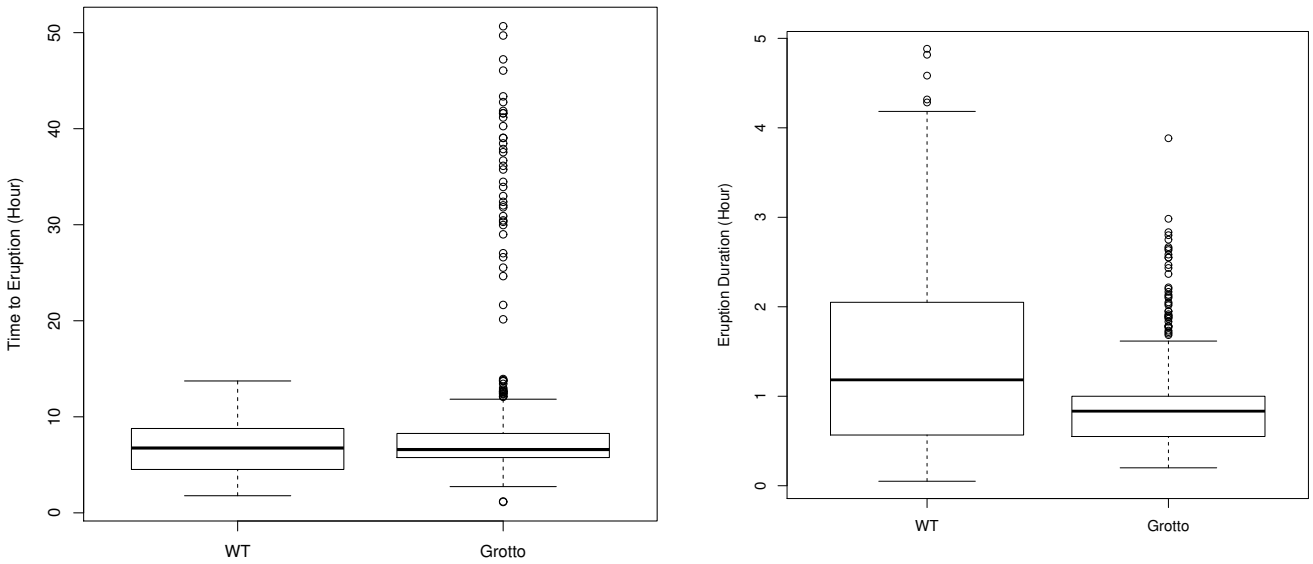
bivariate system. Data used in this study include the exact date and time when eruptions occur, the eruption duration, and the geyser indicator that informs the location of the eruption. Overall, the West Triplet Geyser erupts more frequently than the Grotto Geyser. For the West Triplet Geyser, the average eruption gap time is 6.8 hours with a standard deviation of 2.8, while Grotto Geyser has an average eruption gap time of 9.3 hours with a standard deviation of 8.6. For the duration variable, the West Triplet Geyser has longer durations than the Grotto Geyser, with the average duration 1.5 hours versus 0.9 hours and standard deviation 1.0 and 0.5 for the West Triplet and Grotto, respectively. An application of the model on this data is conducted in Section 3.6.

Distributions of age and duration for both geysers are shown in Figures 3.1a and 3.1b, respectively. The Grotto Geyser has some significantly long event gap times and duration times, while for the West Triplet Geyser, eruption gap times are more stable. In this chapter, our model utilizes the eruption history, and quantify the eruption gap time with the adjustment of eruption durations as the covariate. In Figure 3.2, we illustrate the first three eruptions in the two-geyser system, where the eruption gap time and duration variables are denoted as W and x, repectively. Model notations are introduced in Section 3.3. One objective of the chapter is to predict the next geyser eruption time and location. In order to adapt to our model, the data require an event time variable, an event type variable, and possible covariates.

3.3 Data Setup and Model

3.3.1 Data Setup

Suppose that in a bivariate recurrent process with n events, systematic event time is described as variable T_i. We define the system installation time variable as T_0, while the last event time in the system as T_n. In a bivariate system, there are two types of events, and we use an indicator variable $\Delta_i \in \{1, 2\}$ to represent the source of event. For each event, covariates are denoted as variable $\boldsymbol{X}_i$, where length of $\boldsymbol{X}_i$ depends on numbers of event types. In a bivariate recurrent process, we use $\boldsymbol{X}_i$ as a vector of length of two. Therefore, each event



(a) Time to eruption (Hour)

(b) Eruption duration (Hour)

Figure 3.1: Boxplots for time to eruption and eruption duration for the West Triplet Geyser and Grotto Geyser from June to November in 2008.

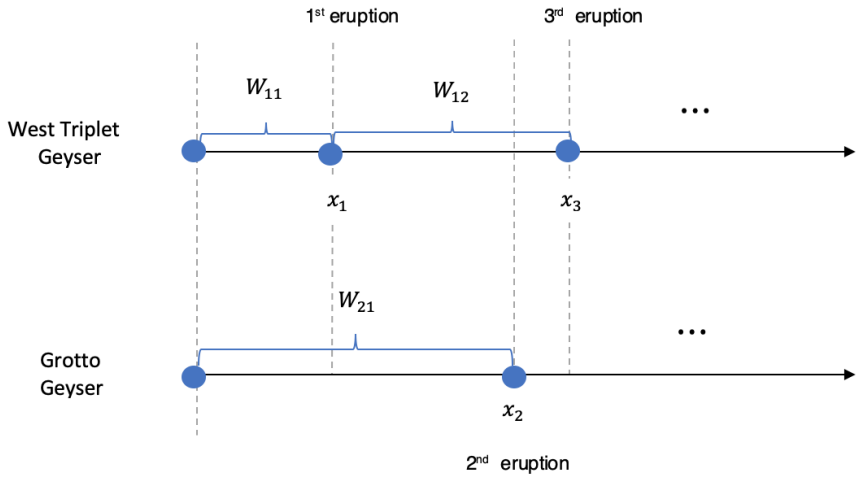


Figure 3.2: Illustration of geyser eruptions with West Triplet and Grotto Geysers.

can be represented by a vector $(T_i, \Delta_i, \boldsymbol{X}_i)'$, where $i = 1, \cdots, n$. For example, an observation $(t_i, \delta_i, \boldsymbol{x}_i)'$ means the i^{th} event occurs at time t_i, it is from event type δ_i, and the covariates are measured as $\boldsymbol{x}_i$, where $i = 1, \ldots, n$. Here, n is the total number of events in an observed process. In the observed recurrent process, time corresponding to the last event is t_n, and it is smaller than the pre-defined stop time κ.

This sequence of events can be simplified as a cumulative process $\{N(t) : t \geq 0\}$. Despite different event types, $N(t)$ is the cumulative number of events at time t. Similarly, we define the cumulation process for event type j as $\{N_j(t) : t \geq 0\}$. In a bivariate recurrent process, $j = 1$ and 2. As a result, up to a time point $s \leq \kappa$, the event history for the system can be denoted as $\mathcal{H}_s = \{N(t) : t \leq s\}$. Similarly, covariates history is expressed as $\mathcal{X}_s = \{\boldsymbol{x}_t : t \leq s\}$. For further discussion, we define $\mathcal{F}_s = \{\mathcal{H}_s, \mathcal{X}_s\}$ as the data history.

For a bivariate recurrent process system, event time variable T defined above satisfies the relation

$$0 = T_0 < T_1 < \cdots < T_i < \cdots < T_n < \kappa.$$

Similarly, for events corresponding to type j, event time variables are defined as T_{lj}, where $l = 1, \ldots, n_j$ and n_j is the number of events from type j. In a bivariate recurrent system, we have the relation $n_1 + n_2 = n$. The time variable T_{lj} is also ordered. For event of type $j = 1$ and $j = 2$,

$$0 = T_{0j} < T_{1j} < \cdots < T_{lj} < \cdots < T_{n_j j} < \kappa.$$

Based on ordered event times, the event gap time variable can be defined for events from the same event type. We define the gap time variable for the j^{th} event type as $W_{lj} = T_{l+1,j} - T_{lj}$, where $l = 0, 1, \ldots, n_j - 1$. It is calculated as time differences between two consecutive events from the same type. In a bivariate recurrent system, W_{lj} can be denoted as $(W_{l1}, W_{l2})'$. For the i^{th} observation, the two-dimensional event gap time variable is described as

$$\mathbf{W}_i = (\mathbf{W}_{l_{i1},1}, \mathbf{W}_{l_{i2},2})',$$

where l_{ij} is the cumulative number of events for j^{th} event type upon time t_i, i.e., $N_j(t_i)$. For an observed recurrent process, at installation point where $t = 0$, the event gap time vector is $\boldsymbol{w}_0 = (w_{0,1}, w_{0,2})'$. Suppose the first event results from type 1 at time t_1, then the second event gap time vector is $\boldsymbol{w}_1 = (w_{1,1}, w_{0,2})'$. Time of the first event type is updated to t_1, while the second one remained the same with no events occur from type 2.

In order to build the relationship between the event gap time variable $\mathbf{W}_i$ and the event time variable T_i. An age variable is introduced as the time difference between the event time T_i and the time of last event for each failure type. In a bivariate recurrent process,

$$\mathbf{A}_i = [A_1(T_i), A_2(T_i)]',$$

where $A_j(t) = t - T_{N_j(t),j}$, and $j = 1$ and 2. For an observed event at time t_i, the observed age vector is $\boldsymbol{a}_i = [a_1(t_i), a_2(t_i)]'$, where $a_j(t) = t - t_{N_j(t),j}$. For example, at installation time where $t = 0$, the age vector is $\boldsymbol{a}_0 = (0, 0)'$. If the first event results from type 1 at t_1, then $\boldsymbol{a}_1 = (0, t_1)'$. The age variable then bridges the gap between real time in the system and the event gap variable $\mathbf{W}_i$ through the relation $\mathbf{W}_i \geq \mathbf{A}_i$. For notation convenience, we define vector comparison to be element-wise comparisons. For example, the comparison of two arbitrary vectors $\boldsymbol{x} \leq \boldsymbol{y}$ is defined to be element-wise comparisons $(x_1 \leq y_1, \cdots, x_n \leq y_n)'$. To motivate our general model, we first introduce the following definition.

3.3.2 Model

The CARP is defined as follows. After an event at t_i (or after the system installation at $t_0 = 0$), event gap times follow,

$$\mathbf{W}_i | \mathcal{F}_{t_i} \sim F_W(\boldsymbol{w} | \boldsymbol{a}_i, \boldsymbol{x}_i), \quad i = 0, 1, 2, \cdots, n, \tag{3.1}$$

where $\boldsymbol{w} = (w_1, w_2)'$, and

$$F_W(\boldsymbol{w}|\boldsymbol{a}_i, \boldsymbol{x}_i) = \Pr(\mathbf{W}_i \leq \boldsymbol{w}|\mathbf{W}_i > \boldsymbol{a}_i, \boldsymbol{x}_i) \tag{3.2}$$

is the joint conditional cumulative distribution function (cdf). At each event time, age is set to be zero for the event type in which the event results in. The event gap time variable $\mathbf{W}_i$ is adjusted by covariates $\boldsymbol{x}_i$, while details are discussed in Section 3.3.3. The subsequent event time and event type are determined by

$$T_{i+1} = T_i + \min_j\{W_{l_{ij},j} - A_j(T_i)\} \quad \text{and} \quad \Delta_{i+1} = \text{argmin}_j\{W_{l_{ij},j} - A_j(T_i)\},$$

respectively, according to the conditional distribution in (3.1). We set our focus on the event gap time $\mathbf{W}_i$, and use it with $\mathbf{A}_i$ to demonstrate the occurrence of the next event. Specifically, according to the current event in a bivariate system at T_i, the next event is determined by the realization of the event gap time variable $W_{l_{ij},j}$, adjusted for the current age value $A_j(T_i)$. We claim the next event comes from event type with a shorter adjusted gap time, where the adjusted gap time is calculated as $W_{l_{ij},j} - A_j(T_i)$. In the process of label prediction for any given time $t \leq \kappa$, the predicted event type of the next event must have a higher probability compared to the other event type. The details of label prediction process is introduced in Section 3.4.2.

3.3.3 Dependence Modeling and Covariate Adjustment

Dependence between events from different types in a bivariate system are modeled by implementing distributional assumptions on variable $\mathbf{W}_i = (W_{i1}, W_{i2})'$. In this chapter, we use a bivariate lognormal distribution and a copula function to model the random vector $\mathbf{W}_i$, where in both models, covariates $\boldsymbol{x}_i$ are used for adjustment. We refer the CARP model under these assumptions as CARP MLN and CARP copula, respectively. Here MLN is short for multivariate lognormal.

CARP MLN

For the lognormal distribution case,

$$\mathbf{W}_i \sim \text{MLN}[\boldsymbol{\mu}(\boldsymbol{x}_i), \boldsymbol{\Sigma}],$$

where the location parameter in the bivariate lognormal distribution is expressed as a linear form of covariates $\boldsymbol{x}_i$. That is,

$$\boldsymbol{\mu}(\boldsymbol{x}_i) = \boldsymbol{\mu}_0 + \boldsymbol{B}\boldsymbol{x}_i. \tag{3.3}$$

In the linear expression above, $\boldsymbol{\mu}_0$ is a vector of baseline location parameters and $\boldsymbol{B}$ is a 2 $\times$ 2 coefficient matrix. In the bivariate lognormal assumption, we use a covariance matrix $\boldsymbol{\Sigma}$ to capture the event dependence between events from two event types. Specifically, the diagonal elements in $\boldsymbol{\Sigma}$ represent marginal variances while the off diagonal element stands for the covariance. When using the bivariate lognormal distribution, the covariance matrix is defined as $\boldsymbol{\Sigma} = \boldsymbol{C}\boldsymbol{C}'$ to ensure $\boldsymbol{\Sigma}$ to be positive definite, where

$$\boldsymbol{C} = \begin{pmatrix} \sigma_1 & 0 \\ \eta & \sigma_2 \end{pmatrix}. \tag{3.4}$$

The correlation ρ is described by σ_2 and η, and the relationship is

$$\rho = \frac{\eta}{\sqrt{\sigma_2^2 + \eta^2}}.$$

In a bivariate lognormal distribution, the marginal distribution of each dimension in fact follows a lognormal distribution. On the contrary, when observed marginal distributions are far from lognormal, or the dependence can not be characterized by our scale parameter $\boldsymbol{\Sigma}$, our assumptions for the bivariate lognormal distribution are violated. One of the alternatives to deal with this insufficiency is to define the W_1 and W_2 in separate distributions and combine

them through a more flexible copula function. This is referred as the CARP copula model.

CARP copula

Let the marginal cdf for two event types be F_1 and F_2. There always exists a copula function C so that the joint cdf for the two dimensional variable $(W_1, W_2)'$ can be written as

$$F(W_1, W_2) = C[F_1(W_1), F_2(W_2)], \tag{3.5}$$

where for any unitary uniform variable $U_j, j = 1, 2$, a bivariate copula is defined as

$$C(u_1, u_2) = \Pr(U_1 \leq u_1, U_2 \leq u_2).$$

This is also known as the Sklar's Theorem. When variables are continuous, the copula function C is unique. With the use of copula function, we are able to choose marginal distributions separately for each event type. For instance, a Gamma distribution and a Weibull distribution can be used as marginal distributions for events corresponding to event type 1 and event type 2, respectively. With selected F_1 and F_2, one can combine marginal distributions by using different copulas.

In literature, various copula functions are introduced and they emphasize on different dependence patterns among marginal distributions. In this chapter, only a special case from Archimedean copula family is discussed, which is the Gumbel copula. We refer the CARP model with the use of Gumbel copula as CARP copula model for the rest of the chapter. In fact, the CARP MLN model is a special case of the CARP copula model, where in CARP MLN, the Gaussian copula is applied.

The CARP MLN model characterizes dependence among event types through the covariance matrix $\mathbf{\Sigma}$, while the CARP copula model quantifies dependence using the copula parameter. Specifically in the Gumbel copula model, the parameter is denoted as α. As a result, in CARP MLN and CARP copula, we have different parameters to character event de-

pendence. In order to compare dependence from different models, we introduce the Kendall's tau.

In CARP copula cases, the Kendall's tau is

$$\tau = 1 - \frac{1}{\alpha},$$

where α is the copula coefficient in Gumbel copula. In CARP MLN cases, the Kendall's tau is calculated as

$$\tau = \frac{2}{\pi} \arcsin\left(\frac{\eta}{\sigma_2}\right),$$

where η and σ_2 can be found in (3.4).

Similar to CARP MLN model, for CARP copula model, we also use a linear form of the covariates $\boldsymbol{x}_i$ to represent location parameters in marginal distributions as in (3.3). In the CARP MLN, we use a two dimensional vector $\boldsymbol{\mu}(\boldsymbol{x}_i) = [\mu_1(\boldsymbol{x}_i), \mu_2(\boldsymbol{x}_i)]'$ to represent the location parameter of the lognormal distribution. While in the CARP copula, $\mu_1(\boldsymbol{x}_i)$ and $\mu_2(\boldsymbol{x}_i)$ stand for location parameters for the first and second marginal distributions, respectively.

3.3.4 Properties of CARP

For event gap time variable $\mathbf{W}_i$, we define the survival function (sf), cdf and hazard function as follows. We denote the joint sf of $\mathbf{W}_i$ to be

$$S(\boldsymbol{v}) = \Pr(W_{i1} > v_1, W_{i2} > v_2). \tag{3.6}$$

The joint cdf is

$$F_W(\boldsymbol{v}) = \Pr(W_{i1} \leq v_1, W_{i2} \leq v_2),$$

and the corresponding joint probability density function (pdf) is denoted by $f_W(\boldsymbol{v})$. According to (3.2), the joint pdf of event gap time variable given all historical events $\mathbf{W}_i|\mathcal{F}_{t_i}$ is

$$f_W(\boldsymbol{w}|\boldsymbol{a}_i.\boldsymbol{x}_i) = \frac{f_W[a_1(c_1), a_2(c_2)]}{S(\boldsymbol{a}_i)}, \quad w_j > a_j(t_i), \quad j = 1, 2, \tag{3.7}$$

where $c_j = t_{l_{ij},j} + w_j, j = 1, 2$. The denominator in (3.7) takes age condition into account, while the numerator builds the relationship among the event gap time w_j, the event time $t_{l_{ij},j}$ and the age $a_j(c_j)$. Covariates $\boldsymbol{x}_i$ are used to adjust the gap time variable $\mathbf{W}_i$ as discussed in Section 3.3.3.

For further discussions, let $\mathcal{F}_{t^-}$ be the event history up to time t. Note that $T_{N_j(t^-),j}$ gives the most recent event time for type j upon time t. For event type j, the age variable prior to time t is denoted as $A_j^-(t) = t - T_{N_j(t^-),j}$. It calculates the cumulative running time upon time t since the last event. For any observed recurrent process, we denote $\boldsymbol{a}^-(t) = [a_1^-(t), a_2^-(t)]'$ for any given time t. Special cases for event times t_i, the age vector is denoted as $\boldsymbol{a}_i^- = \boldsymbol{a}^-(t_i)$. For example, at time 0, the vector is $\boldsymbol{a}_0^- = (0, 0)'$. If the event type is $\delta_1 = 1$ at t_1, then $\boldsymbol{a}_1^- = (t_1, t_1)'$. Note that $\boldsymbol{a}_1$ updates the age to be zero for the corresponding event type, while $\boldsymbol{a}_1^-$ does not. This notation is used to complete the likelihood in Section 3.4, also to define the hazard function as follows.

In literature, the sub-intensity function (i.e., cause-specific event intensity function) is often used to characterize a failure event process. In particular, the sub-intensity function for event type j is defined as

$$h_j(t) = \lim_{\Delta t \to 0} \frac{\Pr[T \in (t, t + \Delta t), \Delta = \delta_j | \mathcal{F}_{t^-}]}{\Delta t}, \tag{3.8}$$

where T is the event time, and Δ is the event type. The sub-cumulative intensity function is $H_j(t) = \int_0^t h_j(s)ds$. The intensity function and cumulative hazard function for the system are

calculated as the summation for two event types,

$$h(t) = \sum_{j=1}^{2} h_j(t) \quad \text{and} \quad H(t) = \sum_{j=1}^{2} H_j(t).$$

The sub-intensity function in (3.8) is calculated as

$$h_j(t) = \frac{D_j[\boldsymbol{a}^-(t)]}{S[\boldsymbol{a}^-(t)]}, \quad \text{where} \quad D_j(\widetilde{\boldsymbol{v}}) = -\frac{\partial S(\boldsymbol{v})}{\partial v_j}\bigg|_{\boldsymbol{v}=\widetilde{\boldsymbol{v}}}, \tag{3.9}$$

and $\widetilde{\boldsymbol{v}} = (\widetilde{v}_1, \widetilde{v}_2)'$ is a vector with given values. The calculation of $D_j(\widetilde{\boldsymbol{v}})$ under different model assumptions are introduced in Section 3.4.1.

3.4 Parameter Estimation and Statistical Inference

3.4.1 Parameter Estimation

The maximum likelihood (ML) approach is used to estimate model parameters. Parameters in the model include parameters in the joint distribution function, parameters in copula function and those in linear covariate transformation function. Given all event history $\mathcal{F}_\tau$, the likelihood function is constructed as follows:

$$L(\boldsymbol{\theta}|\mathcal{F}_\tau) = \prod_{i=1}^{n} L_i(\boldsymbol{\theta}|\boldsymbol{a}_i, \boldsymbol{x}_i), \tag{3.10}$$

where

$$L_i(\boldsymbol{\theta}|\boldsymbol{a}_i, \boldsymbol{x}_i) = \Pr[T_i \in (t_i, t_i + \Delta t), \Delta_i = \delta_i|\mathcal{F}_{t_{i-1}}], \quad \text{where} \quad i = 1, \cdots, n. \tag{3.11}$$

Parameter set $\boldsymbol{\theta}$ denotes all parameters in our model. The estimated parameters $\widehat{\boldsymbol{\theta}}$ are asymptotically normally distributed with a large sample assumption based on the ML theory (Casella and Berger, 2002). The calculation of L_i for CARP models is shown as follows. For any observed recurrent process with n total events, the likelihood contribution for $i = 1, \cdots, n$ in

(3.11) is

$$L_i(\boldsymbol{\theta}|\boldsymbol{a}_i, \boldsymbol{x}_i) = \frac{D_{\delta_i}(\boldsymbol{a}_i^-)}{S(\boldsymbol{a}_{i-1})}. \tag{3.12}$$

In (3.12), the numerator $D_{\delta_i}(\boldsymbol{a}_i^-)$ is introduced in (3.9), it calculates partial derivative of the bivariate sf $S(\boldsymbol{a}_i^-)$. Covariates $\boldsymbol{x}_i$ are used to adjust the distribution of $\mathbf{W}_i$. Likelihood calculations are the same upon this point for CARP MLN and CARP copula models. However, we have different ways to calculate the $D_j(\boldsymbol{a}_i^-)$ for two CARP models, according to how they calculate the survival functions. We show details to calculate the sf for proposed models as follows.

CARP MLN

In CARP MLN model, the sf is calculated in closed form as discussed in (3.6), and the partial derivative with regard to the j^{th} event type can be written as

$$D_j(\boldsymbol{a}_i^-) = -\frac{\partial S(v_j, v_{j'})}{\partial v_j}\bigg|_{\boldsymbol{v}=\boldsymbol{a}_i^-} = f_j[a_j^-(t_i)] \times \Pr[v_{j'} \geq a_{j'}^-(t_i); j' \neq j | v_j = a_j^-(t_i)], \tag{3.13}$$

where $f_j[a_j^-(t_i)]$ is the j^{th} marginal density function from a bivariate lognormal distribution. In (3.13), the conditional probability can be calculated from the conditional normal distribution with a logarithm transformation. Calculation details can be found in the Appendix 3.7.

CARP Copula

In the CARP copula model, the likelihood function is calculated with the relationship between the bivariate sf $S(\boldsymbol{a}_i^-)$ and the cdf $F(\boldsymbol{a}_i^-)$. The partial derivative to the sf is transformed to the partial derivative to the cdf. In bivariate cases, the sf and cdf relation follows:

$$S[a_1(t_i), a_2(t_i)] = 1 - F[a_1(t_i), \infty] - F[\infty, a_2(t_i)] + F[a_1(t_i), a_2(t_i)],$$

where the joint cdf can be calculated by the copula as in (3.5). We need to choose the copula function and marginal distributions. In this study, we only use the Gumbel copula. ML estimates $\widehat{\boldsymbol{\theta}}$ are obtained by maximizing the likelihood function in (3.10).

3.4.2 Next Event Time Prediction

In this section, we introduce the prediction procedure in a bivariate recurrent process. That is, based on all event history and covariates, we predict the next event time and label. Covariates adjustments are made to both CARP MLN and copula models to incorporate external information.

To predict the next event time, let $T_i^* \in (0, \infty)$ be the next event time since the last event at T_{i-1}. By defining next event time variable this way, the next event can then be described as the survival time of the system. This can also be referred as the remaining life time of the system, which is calculated as $\mathbf{E}[S(T^*|\mathcal{F}_{t^-})]$. In the expression, S is the sf. The relationship between event gap time variable $\mathbf{W}_i$, the observed age at last event $\mathbf{A}_{t_{i-1}}$ and the next event time T_i^* is

$$T_i^* = \mathbf{W}_i - \mathbf{A}_{t_{i-1}},$$

where the distribution of $\mathbf{W}_i$ is known as discussed in Section 3.3. Then for any given value $t^* \in (0, \infty)$, the sf is calculated as

$$\begin{aligned} S_{T^*}(t^*|\mathcal{F}_{t^-}) &= \Pr(T^* > t^*|\mathcal{F}_{t^{*-}}) \\ &= \Pr(W_{i1} - \boldsymbol{a}_{t_{i-1},1} > t^*, W_{i2} - \boldsymbol{a}_{t_{i-1},2} > t^*|\mathcal{F}_{t^{*-}}) \\ &= \Pr(W_{i1} > t^* + \boldsymbol{a}_{t_{i-1},1}, W_{i2} > t^* + \boldsymbol{a}_{t_{i-1},2}|\mathcal{F}_{t^{*-}}), \end{aligned}$$

which can be calculated as a probability with respect to the gap time variable $\mathbf{W}_i$ discussed in (3.6). Specifically,

$$\mathbf{E}_{T^*}\{S[t^*|\mathcal{F}_{t^{*-}}, \boldsymbol{a}^-(t^*)]\} = \mathbf{E}_{\mathbf{W}_i}[S(t^* + \boldsymbol{a}_{t_{i-1}})],$$

where the distribution of $\mathbf{W}_i$ is discussed in Section 3.3.3. Covariates $\boldsymbol{x}_i$ are adjusted, and $\boldsymbol{x}_i$ is incorporated in $\mathcal{F}_{t^{*-}}$. Suppose the predicted next event time is $\widehat{t^F}$, its corresponding label is determined by the sub-cdf from two event types. The sub-cdf is introduced as follows.

Let the sub-cdf from event type j to be $F_j(t^*)$, at any given $t^* \in (0, \infty)$, the sub-cdf is expressed as

$$F_j(t^*) = \Pr(W_{ij} \leq t^* + \boldsymbol{a}_{t_{i-1},j}, W_{ij'} \geq W_{ij}, j \neq j'). \tag{3.14}$$

The cdf for the system is summation of sub-cdf across event types, i.e., $F(t^*) = \sum_{j=1}^{2} F_j(t^*)$. The calculation of sub-cdf is shown in Appendix 3.7. The next event label is defined by comparing sub-cdf of each event type at the predicted next event time. Suppose the predicted next event time is $\widehat{t^F}$, the corresponding next event label $\widehat{\Delta}_{i+1}$ is claimed as the type who has the larger sub-cdf at time $\widehat{t^F}$, i.e,

$$\widehat{\Delta}_{i+1} = \text{argmax}_j\{F_j(\widehat{t^F})\}.$$

3.5 Simulation Study

In this section, we simulate bivariate recurrent system from different parameter sets and models. We apply our proposed models on simulated data, and evaluate the model performances. Various parameters are considered in the simulation, including ones in CARP MLN model and ones in CARP copula model. We evaluate the model goodness of fit by calculating average AIC and mean squared error (MSE).

For each generated data, the CARP MLN model and CARP copula model are both applied and results are evaluated. The effect of covariate adjustment is studied by using models with or without coefficient $\boldsymbol{B}$ to generate and estimate data. Models we use to generate data is referred as the true model, while models to fit the data are referred to as the fitted model. In this section, sample sizes and Kendall's tau are changed for different scenarios.

When changing sample sizes, we use 200, 500, 1000 and 2000 to generate data with both

CARP copula model and CARP MLN model. In CARP copula model, lognormal marginal distributions are used for both event types with the Gumbel copula. Location and scale parameters for lognormal marginal distributions in the true model are ($\mu_1 = 1, \sigma_1 = 0.25$) and ($\mu_2 = 1.5, \sigma_2 = 0.25$). For the linear transformation matrix $\boldsymbol{B}$, we use a 2×2 matrix where

$$\boldsymbol{B} = \begin{pmatrix} 1.5 & 0 \\ 0 & 0.1 \end{pmatrix}. \tag{3.15}$$

We keep the Gumbel copula parameter α to be 1.5 for all sample sizes. In CARP MLN generation, we use same location and scale parameters, and adjust the correlation η so that the Kendall's tau in both models are the same.

While changing the Kendall's tau, we rotate the Gumbel copula coefficient α in true models as 1, 1.12, 1.5 and 2.22, while all other parameters remain the same as ones we use in sample size cases. Sample sizes are 1000 in true models. Since CARP MLN and CARP copula models quantify the Kendall's tau differently. In CARP MLN generation, we use η as 0, 0.0443, 0.1445 and 0.299, so that the Kendall's tau from both CARPs are 0, 0.11, 0.33 and 0.55, respectively.

The effect of covariate adjustment is studied by using different types of $\boldsymbol{B}$. True models include CARP copula and MLN for both non-zero and zero $\boldsymbol{B}$ cases. In non-zero $\boldsymbol{B}$ case, we use coefficient matrix in (3.15) to generate data. In zero $\boldsymbol{B}$ true models, $\boldsymbol{B} = \boldsymbol{0}$ is used, where all elements in $\boldsymbol{B}$ is 0. We apply all true models to also fit generated data. The Kendall's tau is set to be 0.33, while other parameters are the same as ones in the sample size case. A sample size of 1000 is used across true models.

Results under different true models and fitted models are shown in this section. For each true model, we repeat the simulation process for 500 times with different randomizations. Under each fitted model, the average AIC is computed. In order to evaluate the performance of parameter estimates, we calculate MSE for location parameter μ, scale parameter σ and linear transformation parameter $\boldsymbol{B}$ for different sample sizes.

In Table 3.1, we show the average AIC in fitted models. Fitted models are CARP MLN and

Table 3.1: Average AIC from CARP MLN and CARP copula calculated by 500 repeated samples on true models generated by both CARP MLN and copula models. Sample sizes from true models are changed from 200, 500, 1000 and 2000.

True Model \ Fitted Model		Copula Generation		MLN Generation	
		MLN	Copula	MLN	Copula
Sample size N	200	701.6	700.1	721.9	721.9
	500	1753.0	1749.1	1805.6	1808.9
	1000	3513.9	3505.3	3609.5	3616.4
	2000	7029.9	7015.6	7216.1	7230.6

Table 3.2: Average AIC by CARP MLN and copula under different Kendall's tau. α and η are used to adjust the Kendall's tau in the true models for CARP MLN and CARP copula, respectively.

True Model \ Fitted Model		Copula Generation		MLN Generation	
		MLN	Copula	MLN	Copula
Kendall's tau (τ)	0	3675.8	3675.2	3455.6	3455.3
	0.11	3512.6	3509.2	3492.9	3494.6
	0.33	3510.1	3519.2	3610.8	3617.6
	0.55	3447.1	3442.0	3840.7	3856.3

Table 3.3: Average AIC from different true models and fitted models to evaluate effect of covariate adjustment. True and fitted models are copula and MLN with or without coefficient $\boldsymbol{B}$.

True Model \ Fitted Model		Copula		MLN	
		Non-zero $\boldsymbol{B}$	Zero $\boldsymbol{B}$	Non-zero $\boldsymbol{B}$	Zero $\boldsymbol{B}$
Copula	Non-zero $\boldsymbol{B}$	3510.8	4057.4	3520.1	4063.7
	Zero $\boldsymbol{B}$	2481.2	2476.4	2487.8	2485.5
MLN	Non-zero $\boldsymbol{B}$	3614.9	4167.1	3608.2	4178.1
	Zero $\boldsymbol{B}$	2572.5	2568.4	2564.5	2560.5

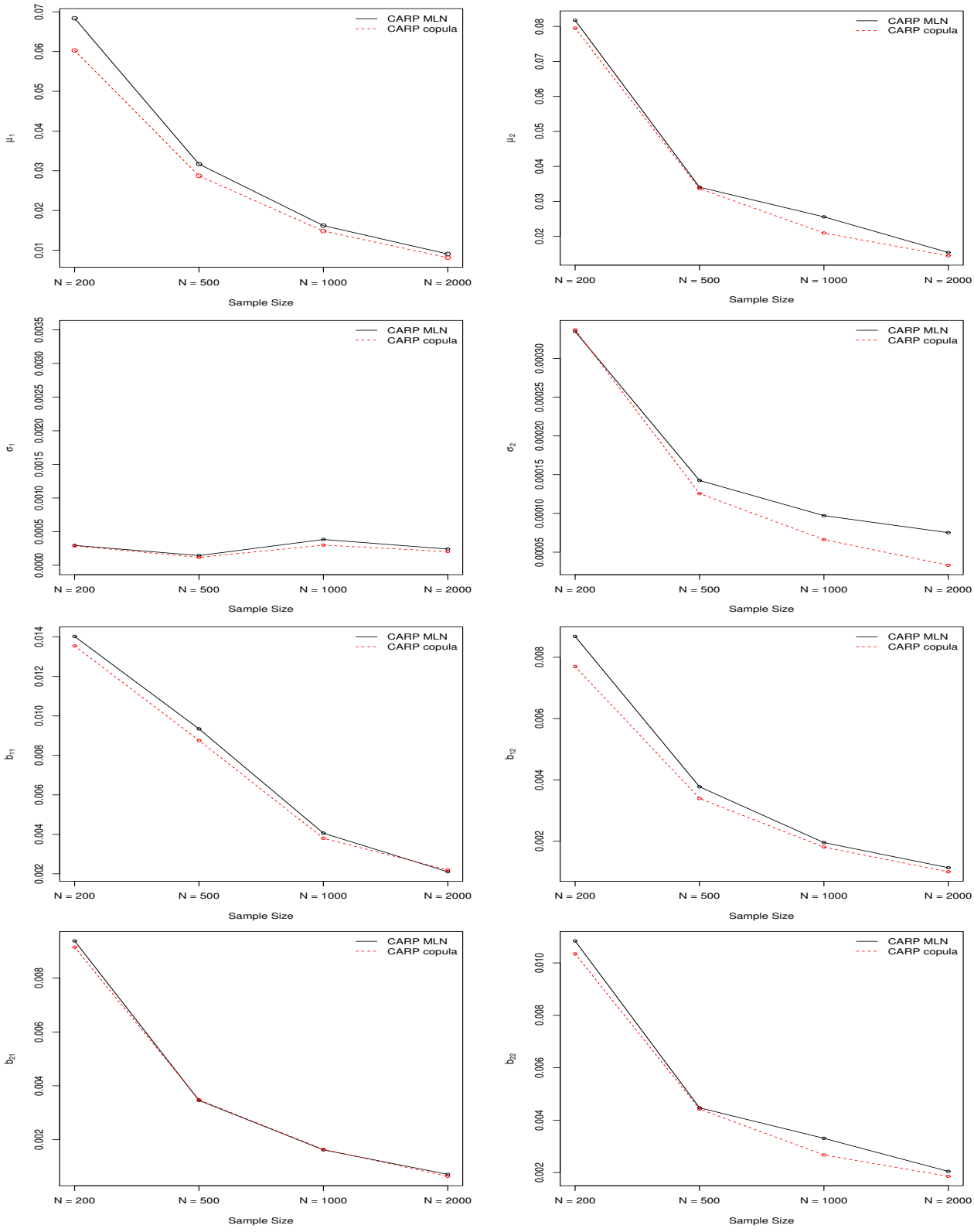


Figure 3.3: MSE for the location and parameter μ_1, μ_2, scale parameter σ_1, σ_2 and coefficient $\boldsymbol{B}$, calculated by both CARP MLN and copula model with different sample sizes. The true model is CARP copula with lognormal marginal distributions.

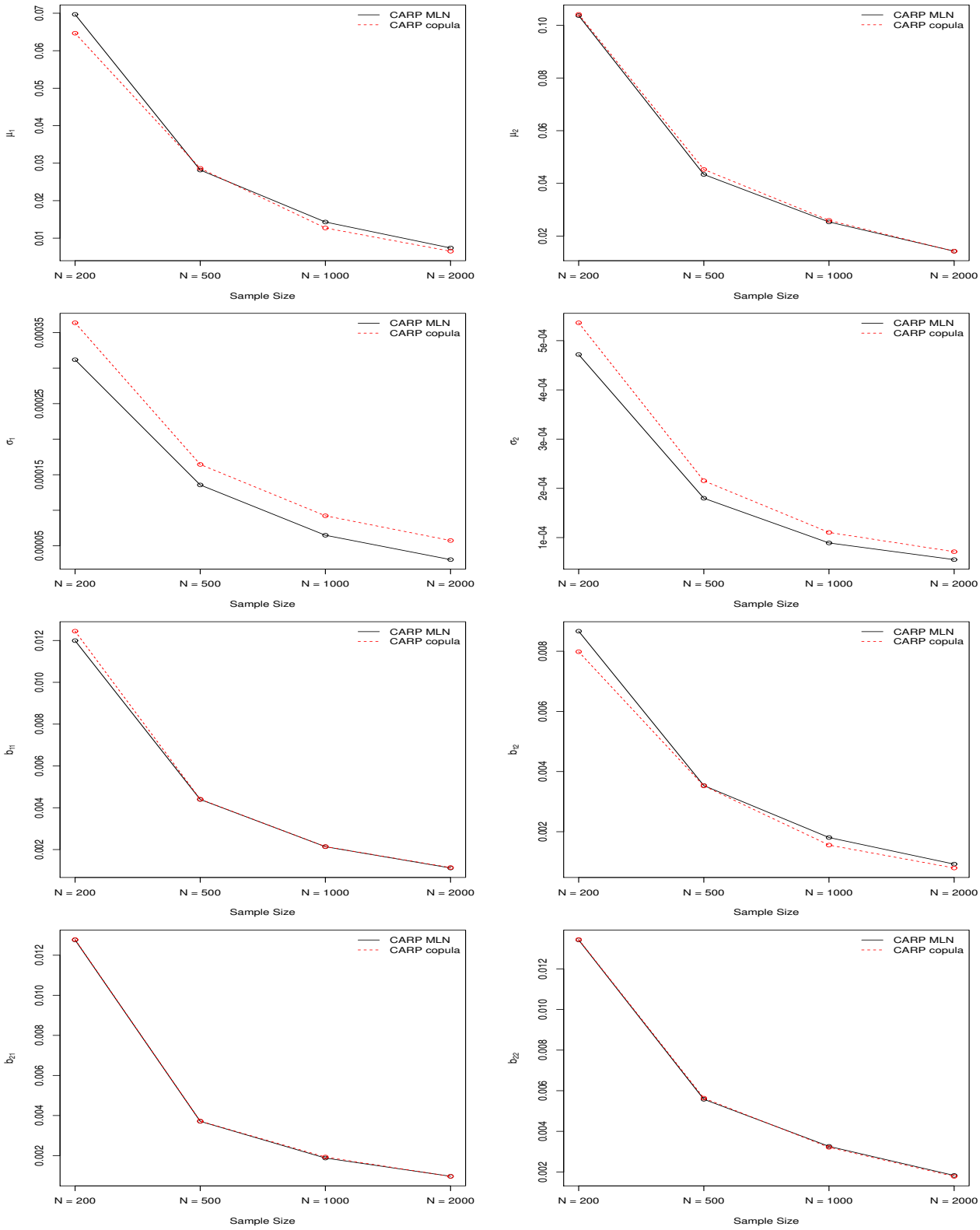


Figure 3.4: MSE for the location parameter μ_1, μ_2, scale parameter σ_1, σ_2 and $\boldsymbol{B}$, calculated by both CARP MLN and copula model with different sample sizes. The true model used to generated data is CARP MLN.

CARP copula models, while true models are from CARP MLN and CARP copula models. For given sample sizes, the fitted CARP model that is used as true models always outperforms the other model according to the average AIC. Specifically, if data are generated by CARP MLN, then the fitted CAPR MLN always has smaller AICs compared to the fitted CARP copula. On the other hand, the fitted CARP copula model has smaller AICs when true models are CARP copula. Figures 3.3 and 3.4 show results of MSE for parameters in both fitted models respectively. Location parameter μ, scale parameter σ and linear transformation coefficient $\boldsymbol{B}$ are presented in those plots. In both plots, the black solid line represents the MSE calculated from the CARP MLN model under different sample sizes, while the red dashed line is the MSE calculated from the CARP copula model. Two major conclusions are: (1) MSE for all parameters are decreasing with the increase of sample sizes in true models. The more data we generate and use to fit models, more reliable model results are. In other words, parameter estimates are closer to true values on average with the increase of sample size. (2) In terms of parameter estimates, both CARP MLN and CARP copula work better under the data generated by the corresponding true models. When the true underlying model is generated by CARP copula and far from bivariate lognormal, CARP copula fits data better. In Figure 3.3, MSEs from CARP copula are consistently smaller than ones from CARP MLN. On the other hand, when true underlying models are bivariate lognormal, both CARP copula and MLN work relatively well. In Figure 3.4, MSEs from CARP copula and CARP MLN are close to each other, while MSEs from CARP MLN are slightly smaller.

In addition, we vary the Kendall's tau for the dependence between two event types. Both CARP MLN and CARP copula are used as true models to generate data with Kendall's tau at 0, 0.11, 0.33 and 0.55. When Kendall's tau is not zero, there exists dependence between two marginal variables. Similar to the sample size case above, the particular CARP model performs better when the fitted model is also used as the true model. For the case where Kendall's tau is zero, marginal distributions are independent. As a result, CARP MLN and copula models are similar. In Table 3.2, for cases where dependence exists, AIC is smaller for fitted model which is also used as the true model. For independent cases, fitted models obtain

similar AICs regardless of true models.

Table 3.3 shows model results under different coefficient $\boldsymbol{B}$. We simulate data with both non-zero and zero $\boldsymbol{B}$. Both CARP MLN and copula are used as true models. For each simulated data, we apply both CARP MLN and CARP copula models with non-zero and zero $\boldsymbol{B}$. When the true model is also used as the fitted model, it has the smallest AIC among all fitted models. The use of covariate adjustment appears to improve model results significantly. In Table 3.3, when non-zero $\boldsymbol{B}$ is applied in true models, fitted models with non-zero $\boldsymbol{B}$ have much smaller AICs compared to ones with zero $\boldsymbol{B}$. On the other hand, for true models with zero $\boldsymbol{B}$ used, using non-zero $\boldsymbol{B}$ fitted models only gains slightly larger AICs than the zero $\boldsymbol{B}$ model. As a result, covariate adjustment should always be recommended according to AIC results.

3.6 Applications

We apply CARP models on a bivariate geyser system from Yellowstone National Park. Two adjacent geysers including West Triplet and Grotto Geyser are chosen to form the bivariate recurrent process. Geyser eruptions are highly related to underground water levels, which can be affected drastically by a nearby geyser eruption. In addition, the duration length for each eruption will affect the corresponding waiting time until the next eruption. In particular, the longer the eruption duration is for the current eruption, the more waiting time we will undergo for the next eruption. Since longer eruptions usually indicate more water consumed. After longer eruptions, we also need longer water gathering times to reach the next eruption.

We use CARP MLN model as well as CARP copula model with Weibull marginal distributions on the geyser data. In the first case, parameters are $\boldsymbol{\theta} = (\mu_1, \mu_2, \sigma_1, \sigma_2, \eta, b_{11}, b_{12}, b_{21}, b_{22})'$, where $\mu_1, \mu_2, \sigma_1, \sigma_2$ and η define baseline location parameters and scale parameters in the bivariate lognormal distribution. In the second case, parameters are $\boldsymbol{\theta} = (s_1, s_2, \sigma_1, \sigma_2, \alpha, b_{11}, b_{12}, b_{21}, b_{22})'$, where s_1, s_2, σ_1 and σ_2 are shape and scale parameters in marginal Weibull distributions, α is the parameter in the Gumbel copula. In both cases, the 2×2 matrix $\boldsymbol{B}$ is

Table 3.4: Parameter estimates and 95% confidence intervals from CARP MLN model calculated using the geyser data.

Parameter	Estimates	95% lower	95% upper
μ_1	0.978	0.870	1.086
μ_2	1.643	1.551	1.746
b_{11}	1.243	1.103	1.383
b_{21}	0.074	−0.041	0.190
b_{12}	−0.003	−0.007	0.000
b_{22}	0.093	0.088	0.098
η	0.012	−0.024	0.048
σ_1	0.337	0.356	0.399
σ_2	0.265	0.247	0.283
τ	0.028	0.028	0.029

the linear coefficient, and it can be denoted as

$$\boldsymbol{B} = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}.$$

In order to make sure shape parameters are positive for both marginal Weibull distributions s_1 and s_2, we apply an exponential transformation on both parameters, i.e., $\mu_1 = \log(s_1)$ and $\mu_2 = \log(s_2)$.

Table 3.4 shows estimation results from CARP MLN model, while Table 3.5 presents estimation results from CARP copula model. For simplicity, we use $\widehat{\boldsymbol{\theta}}_{\text{MLN}}$ and $\widehat{\boldsymbol{\theta}}_{\text{CP}}$ to represent parameter estimates from CARP MLN and CARP copula models, respectively.

Estimated linear coefficients $\widehat{\boldsymbol{B}}$ are shown in (3.16) for both models. They indicate that the longer the eruption time is, the longer waiting time until the next eruption. Specifically, in $\widehat{\boldsymbol{B}}_{\text{MLN}}$, $\widehat{b}_{11} = 1.243$ means the marginal duration effect of West Triplet Geyer on the location parameter is 1.243 on average.

Table 3.5: Parameter estimates and the corresponding 95% confidence intervals from CARP copula model using Weibull marginal distributions.

Parameter	Estimates	95% lower	95% upper
μ_1	-0.096	-0.423	0.231
μ_2	1.529	1.303	1.755
b_{11}	1.649	1.188	2.110
b_{21}	-0.044	-0.324	0.236
b_{12}	-0.003	-0.010	0.003
b_{22}	-0.120	-0.135	-0.104
σ_1	8.898	8.409	9.386
σ_2	7.415	7.179	7.734
α	1.000	0.940	1.060
τ	0.000	-0.002	0.002

$$\widehat{\boldsymbol{B}}_{\text{MLN}} = \begin{pmatrix} 1.243 & \text{-0.003} \\ 0.074 & 0.093 \end{pmatrix} \quad \text{and} \quad \widehat{\boldsymbol{B}}_{\text{CP}} = \begin{pmatrix} 1.649 & \text{-0.003} \\ \text{-0.044} & \text{-0.120} \end{pmatrix}. \tag{3.16}$$

For CARP MLN model, the estimated covariance matrix is

$$\widehat{\boldsymbol{\Sigma}} = \begin{pmatrix} 0.142 & 0.004 \\ 0.004 & 0.071 \end{pmatrix},$$

where the correlation estimation is calculated as $\widehat{\eta}/\sqrt{\widehat{\sigma}_2^2 + \widehat{\eta}^2} = 0.17$, which indicates a slightly positive correlation between event gap times from West Triplet and Grotto Geyser. The longer waiting time for West Triplet Geyser, the longer waiting time for Grotto Geyser to erupt. The Kendall's tau provides an unique measure on the event dependence for both models. The estimates from CARP MLN and copula models are $\widehat{\tau}_{\text{MLN}} = 0.03$ and $\widehat{\tau}_{\text{CP}} = 0$, respectively. the calculated AICs for CARP MLN and copula models are 4424.4 and 4837.3, indicating CARP MLN model is a better fit for the geyser data.

One way to evaluate the goodness of fit to this data is to use the prediction precision matrix. In particular, given each observation in our data, we predict the next eruption time

Table 3.6: Prediction precision matrix based on CARP MLN model

Observed label \ Predicted label	Grotto	West Triplet
Grotto	324	97
West Triplet	82	497

Table 3.7: Prediction precision matrix based on CARP copula model

Observed label \ Predicted label	Grotto	West Triplet
Grotto	353	68
West Triplet	241	338

and label. The corresponding fitted model is used with observed covariates. Predicted and observed labels are used to calculate the precision matrix shown as in Tables 3.6 and 3.7. The prediction precision rate for CARP MLN is 82%, while the CARP copula gives an 69% precision rate to this particular dataset. Rows in Tables 3.6 and 3.7 indicates the observed labels while columns in those tables are the predicted labels.

Another measure for the goodness of fit is the estimated cumulative intensity function as compared to the observed one.Estimated and observed cumulative intensity functions from both models are shown in Figures 3.5 and 3.6. Specifically, the estimated cumulative intensity functions are red dashed lines, while observed ones are black solid lines.

3.7 Remarks

CARP MLN and CARP copula models are introduced in this study. With covariates adjustment, the CARP model provides a new approach to deal with the bivariate recurrent system. Results show that both models are sensitive to true models. When the underlying data are close to a bivariate lognormal distribution, both models work relatively well. However, when the real data are far from bivariate lognormal, the CARP copula model is recommended to improve the model performance. A special case of Archimedean copulas, i.e., Gumbel is considered in this chapter while other copulas are available for our model. One example is the Frank copula. Compared to Gumbel copula where only positive dependence can be quan-

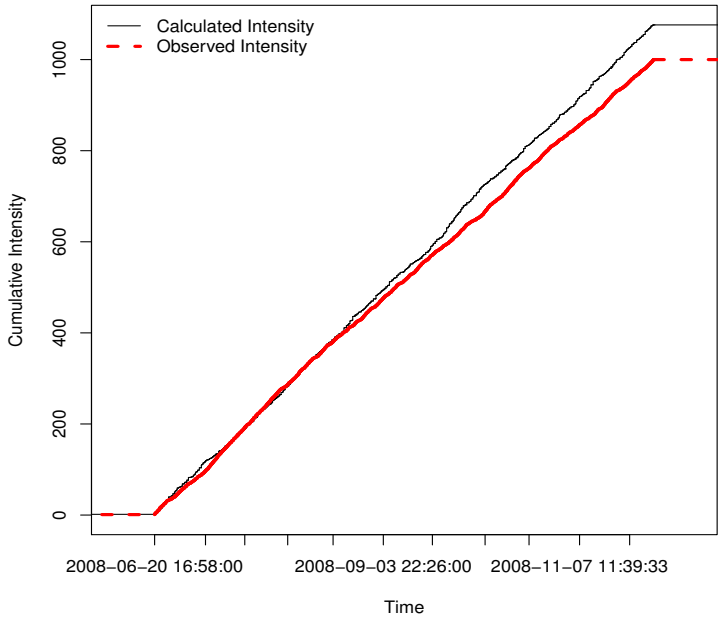


Figure 3.5: Cumulative intensity function from fitted CARP MLN model using the geyser data.

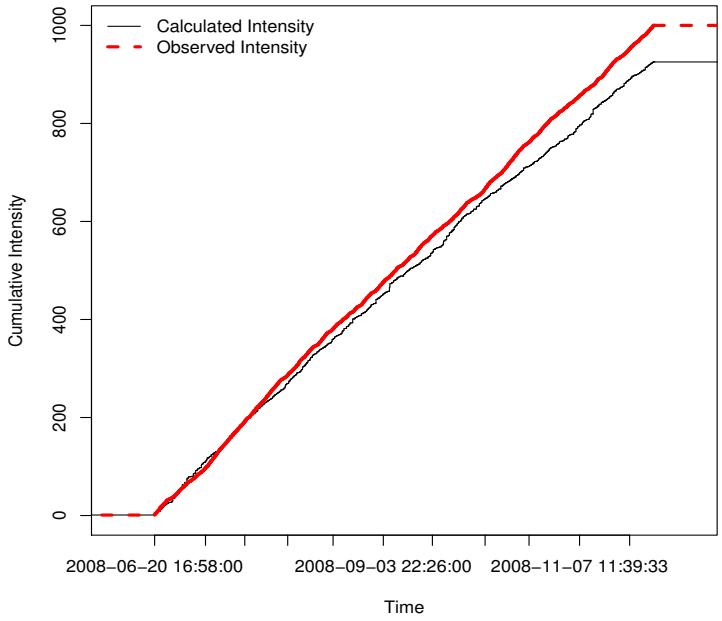


Figure 3.6: Cumulative intensity function calculated from the CARP copula model with the Weibull marginal distributions.

tified, the Frank copula can calculate both positive and negative dependence between two event types. On the other hand, the Gumbel copula has an asymmetric dependence structure where correlations on the tail can be very different than both Gaussian and Frank copulas. In addition, differences of joint density functions between Frank and Gaussian copulas are negligible when marginal distributions used are the same.

Adjustment of effective covariates significantly improves model results. When the true model include a significant covariate effect, using covariate adjustment significantly improves model fitting. On the other hand, if there is no covariate effect in the true model, including covariate adjustment for the fitted model does not increase AICs too much. In reality, the covariate adjustment should always be recommended when true models are unknown.

The choice of marginal distributions in CARP copula model is based on AICs. Specifically, marginal distributions with the minimum AIC is used in our model where marginal distributions for different event types can be different. This also adds flexibility to the CARP copula model. When true models include two different marginal distributions other than lognormal, our CARP model will handle the situation well.

A multivariate CARP model can be considered for future work when there are more than two components in the recurrent system. In the geyser data application, covariates other than eruption duration can be used when there exists a correlation between eruption time and duration.

Appendix B

Conditional Lognormal Probability

If a bivariate random variable $\boldsymbol{y} = (y_1, y_2)'$ follows a lognormal distribution $\text{MLN}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ where

$$\boldsymbol{\mu} = (\mu_1, \mu_2)' \quad \text{and} \quad \boldsymbol{\Sigma} = \begin{pmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{pmatrix},$$

then the conditional distribution of $\log(y_1)|\log(y_2)$ follows a normal distribution $\mathrm{N}(\mu_c, \sigma_c)$ with location and scale parameter as

$$\mu_c = \mu_1 + \sigma_{12}\sigma_{22}^{-1}[\log(y_2) - \mu_2] \quad \text{and} \quad \sigma_c = \sigma_{11} - \sigma_{12}\sigma_{22}^{-1}\sigma_{21}.$$

Calculation of the Sub-cdf

The calculation of (3.14) can be transformed into an integral to a probability product. That is,

$$\begin{aligned}&\Pr(W_{ij} \le t^* + \boldsymbol{a}_{t_{i-1},j}, W_{ij'} \ge W_{ij}, j \ne j')\\ &= \int_0^{t^*+\boldsymbol{a}_{t_{i-1},j}} \Pr(W_{ij'} \ge s | W_{ij} = s) f(s) ds,\end{aligned}$$

where $f(s)$ is the marginal density function and $\Pr(W_{ij'} \ge s|W_{ij} = s)$ is the conditional probability with respect to the event gap time variable $\mathbf{W}_i$. In the integral, the density $f(s) = \Pr(s \le W_{ij} \le s + ds)$ can be obtained with the following relationship:

$$f(s) = \frac{\Pr(s \le W_{ij} \le s + ds)}{ds},$$

where ds can be ignored in the calculation of likelihood without affecting final results.

Confidence Interval for Kendall's tau

For lognormal cases, the Kendall's tau estimator can be expressed as $\widehat{\tau} = (2/\pi)\arcsin(\widehat{\eta} / \sqrt{\widehat{\sigma}_2^2 + \widehat{\eta}^2})$ where the asymptotic distribution is known from the ML estimator. Using the delta method,

$$\mathrm{Var}(\widehat{\tau}) = \left(\frac{\partial\widehat{\tau}}{\partial\widehat{\eta}}\right)^2 \mathrm{Var}(\widehat{\sigma}_2) + \left(\frac{\partial\widehat{\tau}}{\partial\widehat{\sigma}_2}\right)^2 \mathrm{Var}(\widehat{\eta}) + 2\left(\frac{\partial\widehat{\tau}}{\partial\widehat{\eta}}\right)\left(\frac{\partial\widehat{\tau}}{\partial\widehat{\sigma}_2}\right)\mathrm{cov}(\widehat{\eta}, \widehat{\sigma}_2),$$

where

$$\frac{\partial \widehat{\tau}}{\partial \widehat{\eta}} = \frac{2}{\pi} \frac{1}{\sqrt{1 - \dfrac{\widehat{\eta}^2}{\widehat{\sigma}_2^2 + \widehat{\eta}^2}}} \left[(\widehat{\eta}^2 + \widehat{\sigma}_2^2)^{-\frac{1}{2}} - \frac{1}{2} \widehat{\eta} (\widehat{\eta}^2 + \widehat{\sigma}_2^2)^{-\frac{3}{2}} \right],$$

and

$$\frac{\partial \widehat{\tau}}{\partial \widehat{\sigma}_2} = -\frac{2}{\pi} \frac{1}{\sqrt{1 - \dfrac{\widehat{\eta}^2}{\widehat{\sigma}_2^2 + \widehat{\eta}^2}}} \left[-\widehat{\eta} \widehat{\sigma}_2 (\widehat{\eta}^2 + \widehat{\sigma}_2^2)^{-\frac{3}{2}} \right].$$

For the Gumbel copula case,

$$\widehat{\tau} = 1 - \frac{1}{\widehat{\alpha}}.$$

Similarly with the lognormal case, the variance for the estimator can be written by using the delta method. That is,

$$\mathrm{Var}(\widehat{\tau}) = \left[\frac{\partial g(\widehat{\alpha})}{\partial \widehat{\alpha}} \right]^2 \mathrm{Var}(\widehat{\alpha}),$$

where

$$g(x) = 1 - \frac{1}{x}.$$

Bibliography

P. K. Andersen and R. D. Gill. Cox's regression model for counting processes: a large sample study. *The Annals of Statistics*, 10:1100–1120, 1982.

O. Bouaziz, F. Comte, and A. Guilloux. Nonparametric estimation of the intensity function of a recurrent event process. *Statistica Sinica*, pages 635–665, 2013.

G. Casella and R. L. Berger. *Statistical Inference*, volume 2. Duxbury Pacific Grove, CA, 2002.

D. R. Cox. Regression models and life tables (with discussion). *Journal of the Royal Statistical Society, Series B*, 34:187–220, 1972.

J.-y. Dauxois and S. Sencey. Non-parametric tests for recurrent events under competing risks. *Scandinavian Journal of Statistics*, 36:649–670, 2009.

R. O. Fournier. Old faithful: A physical model. *Science*, 163:304–305, 1969.

F. G. Garre, A. H. Zwinderman, R. B. Geskus, and Y. W. Sijpkens. A joint latent class changepoint model to improve the prediction of time to graft failure. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 171:299–308, 2008.

X. Huang and L. Liu. A joint frailty model for survival and gap times between recurrent events. *Biometrics*, 63:389–397, 2007.

A. V. Huzurbazar and B. J. Williams. Incorporating covariates in flowgraph models: applications to recurrent event data. *Technometrics*, 52:198–208, 2010.

J. Lawless. The analysis of recurrent events for multiple subjects. *Applied Statistics*, pages 487–498, 1995.

L. Liu and X. Huang. Joint analysis of correlated repeated measures and recurrent events processes in the presence of death, with application to a study on acquired immune deficiency syndrome. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 58:65–81, 2009.

Y. Ogata. A prospect of earthquake prediction research. *Statistical Science*, pages 521–541, 2013.

P. Prasad and K. Rao. Reliability models of repairable systems considering the effect of operating conditions. In *proceedings of Annual of Reliability and Maintainability Symposium*, pages 503–510. IEEE, 2002.

J. S. Rinehart. Fluctuations in geyser activity caused by variations in earth tidal forces, barometric pressure, and tectonic stresses. *Journal of Geophysical Research*, 77:342–350, 1972.

D. E. Schaubel and J. Cai. Non-parametric estimation of gap time survival functions for ordered multivariate failure time data. *Statistics in medicine*, 23:1885–1900, 2004.

L. Sun, D.-H. Park, and J. Sun. The additive hazards model for recurrent gap times. *Statistica Sinica*, pages 919–932, 2006.

S. Ullah, T. J. Gabbett, and C. F. Finch. Statistical modelling for recurrent events: an application to sports injuries. *Br J Sports Med*, 48:1287–1293, 2014.

I. Wijesundera, M. N. Halgamuge, T. Nirmalathas, and T. Nanayakkara. A geographic primitive-based bayesian framework to predict cyclone-induced flooding. *Journal of Hydrometeorology*, 14:505–523, 2013.

Q. Yang, N. Zhang, and Y. Hong. Statistical reliability analysis of repairable systems with dependent component failures under partially perfect repair assumption. *IEEE Transactions on Reliability*, 62:490–498, 2013.

Q. Yang, Y. Hong, N. Zhang, and J. Li. A copula-based trend-renewal process model for analysis of repairable systems with multitype failures. *IEEE Transactions on Reliability*, 66:590–602, 2017.

D. Zeng and D. Lin. Semiparametric transformation models with random effects for joint analysis of recurrent and terminal events. *Biometrics*, 65:746–752, 2009.

Chapter 4 Statistical Methods for Thermal Index Estimation Based on Accelerated Destructive Degradation Test Data

Abstract

Accelerated destructive degradation test (ADDT) is a technique that is commonly used by industries to access material's long-term properties. In many applications, the accelerating variable is usually the temperature. In such cases, a thermal index (TI) is used to indicate the strength of the material. For instance, one may interpret a TI of 500 C as the material can be expected to maintain a specific property at a temperature of 500°C for 100,000 hours. A higher TI indicates the particular material has a stronger resistance to thermal damage. In literature, there are three methods available to estimate the TI based on ADDT data, which are the traditional method based on the least-squares approach, the parametric method, and the semiparametric method. In this chapter, we provide a comprehensive review of the three methods and illustrate how TI can be estimated based on different models. We also conduct comprehensive simulation studies to show the properties of different methods. We provide thorough discussions on the pros and cons of each method. The comparisons and discussion in this chapter can be useful for practitioners and future industrial standards.

Key Words: Adhesive Bond B, Arrhenius model, degradation process, least squares, polymeric materials, semi-parametric.

4.1 Introduction

4.1.1 Background

Polymeric materials are common in various industrial applications. In current industrial practice, a thermal index (TI) is often used to describe the long-term performance of polymeric materials. As specified in industrial standard UL746B (2013), the TI of a polymeric material can be considered a measure of the material's ability to maintain a specific property (e.g., physical and electrical properties) over a prolonged period of time (usually 100,000 hours) under exposure to elevated temperatures. The interpretation of the TI is as follows. A material with at TI value of 500°C is expected to maintain the rated property for the exposure to a temperature of 500°C for 100,000 hours. Thus, a material with a higher TI rating is expected to pertain a stronger resistance to thermal exposure as compared to those with a lower TI ratings. The TI can also be used to determine if a material is suitable for a particular application, and for comparing multiple materials. When a material is introduced to a field, its TI can be compared to a list of similar materials with known TI values, which can give insights for the long term performance of the new material. Therefore, estimating the TI for a material is an important task in material property demonstration.

To estimate the TI, necessary data need to be collected, which track the material property over time. Such data are often referred to as degradation data. However, the degradation of the material performance is often gradual and can take years to observe deteriorations. To collect information in a timely manner, accelerated degradation test (ADT) is often used. In the setting of TI estimation, temperature is often the accelerating variable. When measuring the material performance, such as the tensile strength, the sample will be stretched until it breaks (i.e., the sample is destroyed in the testing procedure). In this way only one data point can be collected form one sample. Such ADT is called accelerated destructive degradation testing (ADDT). ADDT is a commonly used technique for evaluating long-term performance of polymeric materials, due to the nature of the testing. Examples of ADDT data include the

Adhesive Bond B data in Escobar et al. (2003), the Polymer data in Tsai et al. (2013), the Seal Strength data in Li and Doganaksoy (2014), and the Formulation K data in Xie et al. (2018).

To use the ADDT data for the TI estimation, a statistical method is needed. In literature, there are three methods available to estimate the TI based on ADDT data, which are the traditional approach based on the least-squares method, the parametric approach based on maximum likelihood (ML) method, and the semi-parametric approach based on splines method. The traditional procedure is the one that is currently specified in the industrial standards UL 746B, which is commonly used in applications. The traditional approach is a two-step approach using polynomial fittings and least-squares methods. In the statistical literature, the parametric method is also commonly used to model the ADDT degradation paths, and the ML method is used for parameter estimation. Recently, a semiparametric method is proposed to analyze ADDT data in Xie et al. (2018). The basic idea of the semiparametric method is to use monotonic spline method to model the baseline degradation path and use a parametric method to model the effect of accelerating variable.

The objective of this chapter is to provide a comprehensive review of the three methods and illustrate how the TI can be estimated based on different models. We also conduct comprehensive simulation studies to show the properties of different methods. Then, we provide thorough discussions on the pros and cons of each method. The comparisons and discussions in this chapter can be useful for practitioners and future industrial standards.

4.1.2 Related Literature

Degradation data were used to access products and material reliability in early work such as Nelson (1990, Chapter 10), and Lu and Meeker (1993). There are two types of degradation data: repeated measures degradation (RMDT) data and ADDT data. For RMDT data, multiple measurements can be taken from the same unit. For ADDT data, only one measurement can be taken from the same unit due to the destructive nature of the measuring procedure. Different types of methods are used to analyze RMDT and ADDT data. The majority of the

degradation literature is on RMDT data analysis, features two major classes of models: the general path model [e.g., Meeker and Escobar (1998), and Hong et al. (2015)] and stochastic process models [e.g., Whitmore (1995), Park and Padgett (2005), and Wang and Xu (2010)]. A review of statistical degradation models and methods are available in Meeker et al. (2011), and Ye and Xie (2015).

This chapter focuses on the analysis of ADDT data and their corresponding TI estimation. Regarding ADDT analysis, the traditional approach for TI estimation using the least-squares method is described in UL746B (2013). Parametric models are quite common in ADDT analysis, for example, in Tsai et al. (2013), Escobar et al. (2003), and Li and Doganaksoy (2014). King et al. (2018) apply both the traditional and parametric approaches to ADDT data analysis and TI estimations. King et al. (2018) also conduct a comprehensive comparisons for the two approaches in TI estimations. Xie et al. (2018) develop a semiparametric approach for ADDT data analysis, in which the monotonic splines are used to model the baseline degradation path and they use the Arrhenius relationship to describe the temperature effect. However, the TI estimation procedure is not developed in Xie et al. (2018).

In this chapter, we develop the TI estimation based on the semiparametric method after providing a review of the existing methods in TI estimations. We also conduct comprehensive simulations to compare the three methods. In terms of software implementation, Hong et al. (2016) implement the three methods and their corresponding TI estimation procedures into an R package "ADDT". Details and illustrations of the R package ADDT is available in Jin et al. (2017).

4.1.3 Overview

The rest of this chapter is organized as follows. Section 4.2 introduces the concept of ADDT, examples of ADDT data, and the concept of TI. Section 4.3 presents the three different methods that can be used to model ADDT data and their corresponding procedures for TI estimation. The three different methods are the traditional methods, the parametric method, and the semiparametric method. Section 4.5 conducts extensive simulations to compare per-

Table 4.1: Illustration sample size allocation for an ADDT.

Temperature	Measuring Points (Hours)					
(°C)	0	552	1008	2016	3528	5040
-	10					
250		5	5	5	5	5
260		5	5	5	5	5
270		5	5	5	5	5
280		5	5	5	5	5

formances of the estimation procedures. Section 4.6 provides a comprehensive discussion on the pros and cons of each method, and suggestions for practitioners.

4.2 Accelerated Tests and Thermal Index

In this section, we give a more detailed introduction to ADDT and TI.

4.2.1 Test Plans

The test plan of an ADDT consists of temperature levels, measuring time points, and number of samples assigned to each temperature levels and measuring time points combinations. Table 4.1 illustrates a test plan for an ADDT. Four elevated temperature levels are considered in the test, which are 250°C, 260°C, 270°C, and 280°C. There are five measuring time points considered in this plan, which are 552 hours, 1008 hours, 2016, hours, 3528 hours, and 5040 hours. At the initial time (time zero), there are ten sample units tested under the normal temperature level to serve as the baseline. Then for each combination of temperature level and time points, there are five sample units tested to obtain needed measurements for the material property. To measure some properties like tensile strength, the unit will be destroyed after the measurement. Note that equal sample allocation is used in Table 4.1. However, unequal sample size allocation is also applied in practice. See King et al. (2018) for more detailed discussion on the test plans.

4.2.2 Data and Notation

The ADDT data record the material property (e.g., the tensile strength of the material) for each unit. Here, we use the Adhesive Bond B example in Escobar et al. (2003) to illustrate the ADDT data. Figure 4.1 shows a scatter plot of the Adhesive Bond B data. In general, we observe a decreasing trend over time. Under higher temperature level, the rate of decreasing is faster than those under lower temperature levels.

Here we introduce some notations to the ADDT data that will be necessary for the development of statistical methods. Let n be the number of temperature levels and n_i be the number of measuring time points for temperature level i. The value of the ith temperature level is denoted by A_i. The corresponding time points are denoted by t_{ij}, $j = 1, \cdots, n_i$. Note that it is possible measure time points are different for different temperature levels. Let n_{ij} be the number samples tested at time t_{ij} for temperature level i. Note that the number of samples tested at each time point t_{ij} can also vary. We denote the degradation measurement by y_{ijk} for the kth sample at level i of the temperature and measuring time t_{ij}, $i = 1, \cdots, n$, $j = 1, \cdots, n_i$, and $k = 1, \cdots, n_{ij}$. The total number of measured samples are $N = \sum_{i=1}^{n} \sum_{j=1}^{n_i} n_{ij}$.

4.2.3 Thermal Index

In this section, we introduce the general concept of the thermal index (TI). In the following, we will use the tensile strength as the interested material property. In a common framework of degradation modeling, the failure time is defined as the first time when the degradation level passes the failure threshold. For example, a failure is said to have occurred when the sample tensile strength reaches a certain percentage of the original tensile strength (e.g., 50%).

For degradation processes that are accelerated by temperature, the Arrhenius relationship is widely used to model the relationship between the degradation and temperature. In particular, the Arrhenius model uses the following transformed temperature,

$$h(A) = \frac{-11605}{A + 273.16}, \tag{4.1}$$

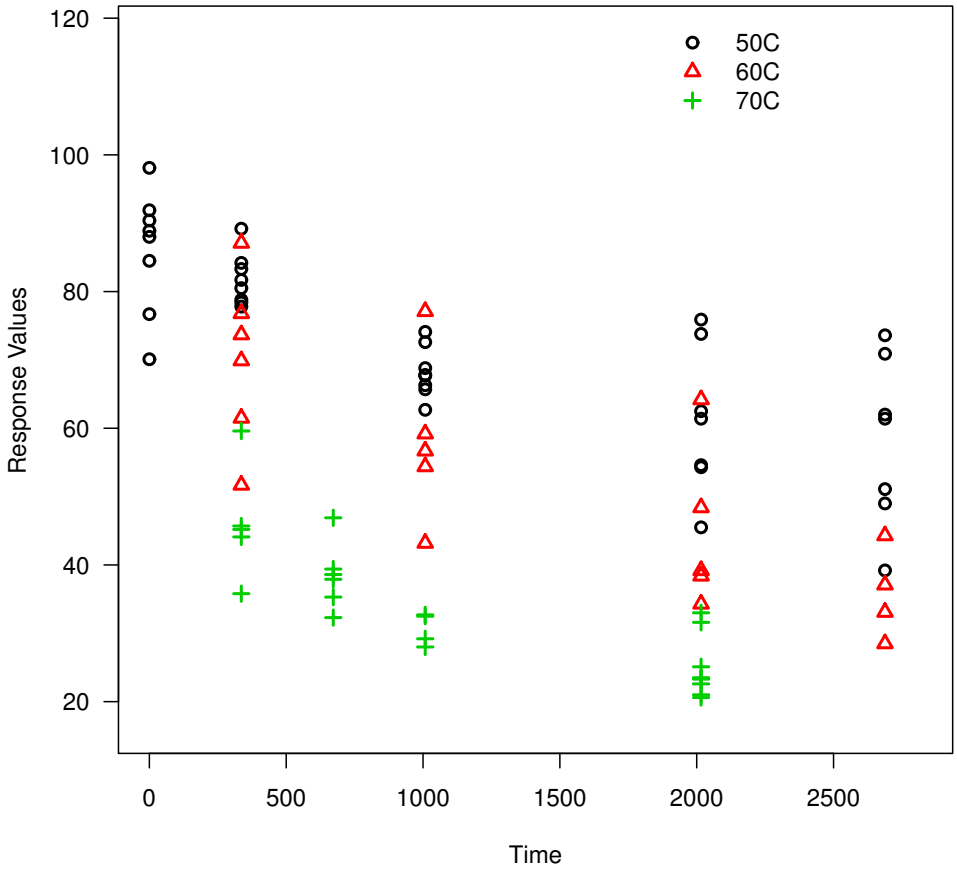


Figure 4.1: Scatter plot of the Adhesive Bond B data. The x-axis is time in hours and the y-axis is strength in Newtons.

where A is the temperature value in degrees Celsius, the constant 11605 is the reciprocal of the Boltzmann's constant (in units of eV). Note that the constant 273.16 is for converting the Celsius temperature to the Kelvin temperature scale. For the convenience of modeling, we define,

$$x = \frac{1}{A + 273.16}, \quad \text{and} \quad x_i = \frac{1}{A_i + 273.16}.$$

Through the modeling of degradation data, which will be detailed in Section 4.3, the mean time to failure at x can be described by a relationship $m(x)$. For targeted time to failure t_d (e.g., $t_d = 100{,}000$ hours), the corresponding temperature level R can be obtained by solving x_d from $m(x_d) = t_d$. Because

$$x_d = m^{-1}(t_d) = \frac{1}{R + 273.16},$$

we obtain the corresponding temperature value R as

$$R = \frac{1}{m^{-1}(t_d)} - 273.16. \tag{4.2}$$

The temperature level R in (4.2) is defined as the TI for the material. Figure 4.2 illustrates the temperature-time relationship based on the Arrhenius and corresponding TI.

Note that the targeted time to failure is not necessary to be fixed at 100,000. For example, if there is an existing material with a known TI (e.g., 220°C), its targeted time to failure t_d^{old} can be obtained. For a new material, its TI can be obtained by using t_d^{old} as the targeted time. In this case, the TI for the new material is called relative TI because it compares to the existing material, see King et al. (2018) for more details.

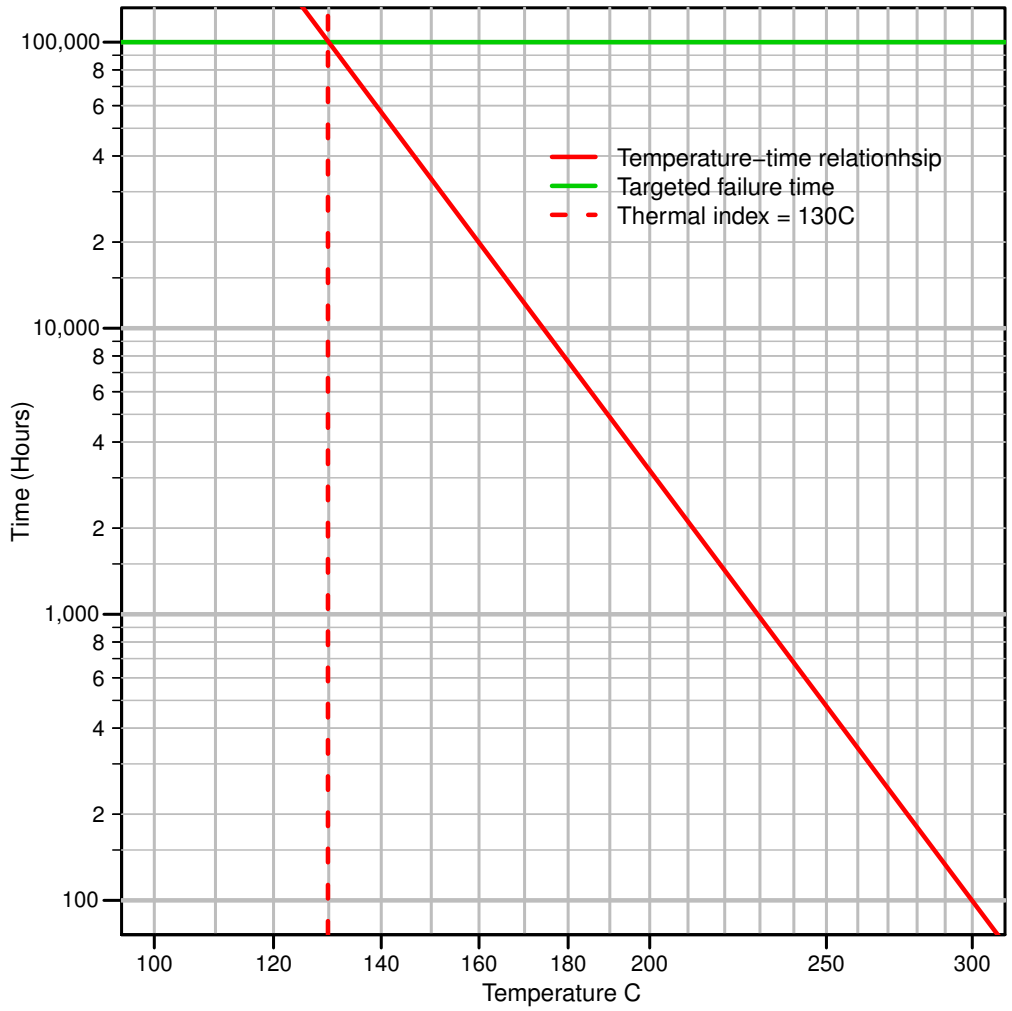


Figure 4.2: Illustration of temperature-time relationship and TI. The x-axis is temperature A on the scale of $1/(A + 273.16)$, and the y-axis is time in hours on base 10 logarithm scale.

4.3 Statistical Methods for Thermal Index Estimations

This section covers the statistical methods for the TI estimation. We first review the traditional and the parametric methods as described in King et al. (2018). Then, we derived the TI estimation based on the semiparametric model in Xie et al. (2018).

4.3.1 The Traditional Method

The traditional method is the methodology described in UL746B (2013), which is the current accepted standard for ADDT data analysis in industry. The procedure essentially is a two-step approach. As descried in UL746B (2013), the basic idea is to find an appropriate model to link the time to failure to the level of degradation, and then estimate the parameters by applying the least-squares technique. The estimated failure time is done by interpolating the fitted curves. When there is no material-specific knowledge on the degradation relationship, the UL standards recommend using a third-order polynomial fitting.

Specifically, for temperature level i, we first compute the points $\{t_{ij}, \overline{y}_{ij.}\}$, $j = 1, \cdots, n_i$, where

$$\overline{y}_{ij.} = \frac{1}{n_{ij}} \sum_{k=1}^{n_{ij}} y_{ijk}$$

is the batch average for observations at time t_{ij} for temperature level i. A third order polynomial $a_{0i}+a_{1i}t+a_{2i}t^2+a_{3i}t^3$ is used to fit the data points $\{t_{ij}, \overline{y}_{ij.}\}$, $j = 1, \cdots, n_i$, separately for each temperature level. Here, $(a_{0i}, a_{1i}, a_{2i}, a_{3i})'$ are the polynomial coefficients to be estimated by the least-squares method.

After obtaining the estimates of $(a_{0i}, a_{1i}, a_{2i}, a_{3i})'$, the mean failure time m_i for temperature level i can be obtained through interpolation. In particular, one needs to solve,

$$a_{0i} + a_{1i}m_i + a_{2i}m_i^2 + a_{3i}m_i^3 = y_f, \tag{4.3}$$

where y_f is the failure threshold. The failure threshold is usually set at 50% of the initial strength, though different values may be set according to specifications of different applica-

tions.

Through the polynomial interpolation, a set of data points $\{x_i, m_i\}, i = 1, \cdots, n$ are obtained where x_i is the transformed temperature as defined in (4.1). The least-squares method is used again to fit a straight line to data points $\{x_i, \log_{10}(m_i)\}, i = 1, \cdots, n$. That is, to fit the following model,

$$\log_{10}(m_i) = \beta_0 + \beta_1 x_i, i = 1, \cdots, n,$$

to obtain the estimates of β_0 and β_1. Note that the base 10 logarithm is used here because it is more popular in engineering literature. In the traditional method the temperature-time relationship is represented as

$$\log_{10}[m(x)] = \beta_0 + \beta_1 x. \tag{4.4}$$

With the fitted temperature-time relationship in (4.4), the TI based on the traditional method is obtained as

$$R = \frac{\beta_1}{\log_{10}(t_d) - \beta_0} - 273.16\,. \tag{4.5}$$

where t_d is the target time and $t_d = 100{,}000$ is often used.

The traditional method is fairly intuitive and straightforward to compute, which is of advantages. Here we provide some other considerations for the traditional method. Because the method is based on interpolation, it requires the degradation level to reach the failure threshold so that m_i can be obtained for level i. Otherwise, all data collected at level i can not be used for analysis. The number of temperature levels for the ADDT is usually small (i.e., around 4), thus only a few number of observations are available to fit the model in (4.4). Because it is a two-step approach, so far there is no method to quantify the uncertainty associated with the TI estimation for the traditional method. For ADDT data, one would expect higher temperature levels to yield shorter lifetimes. Due to randomness in the data

and the flexibility of polynomials, the traditional method can produce estimated failure times that are not monotonically increasing with temperature, which might be unrealistic. With parametric models, most of the concerns can be addressed.

4.3.2 The Parametric Method

In statistical literature, parametric methods are prevalent ADDT data analysis such as in Tsai et al. (2013), Escobar et al. (2003), and Li and Doganaksoy (2014). In the parametric method, the primary estimation and inference method is based on a parametric model and maximum likelihood theory. Here, we give a brief description for the parametric method summarized in King et al. (2018).

In this setting, the parametric model for the degradation measurement is represented as

$$y_{ijk} = \mu(t_{ij}; x_i) + \epsilon_{ijk}, \tag{4.6}$$

where $\mu(t_{ij}; x_i)$ is the underly degradation path and ϵ_{ijk} is an error term. Because the tensile strength is decreasing over time, the function $\mu(t; x_i)$ is a decreasing function of t. Since a higher temperature usually causes a higher rate of degradation, the function $\mu(t; x_i)$ is also a decreasing function of the temperature.

For a specific x, the mean time to failure $m(x)$ can be solved from

$$\mu[m(x); x] = y_f,$$

leading to the temperature-time relationship as

$$m(x) = \mu^{-1}(y_f; x).$$

The TI can be solved from $m(x_d) = t_d$, which is equivalent to solve x_d from

$$\mu(t_d; x) = y_f.$$

The TI can be computed from the solution x_d. That is,

$$R = \frac{1}{x_d} - 273.16.$$

To proceed with the modeling, one needs to be specific about the form of $\mu(t; x_i)$. For polymer materials, the parametric form in Vaca-Trigo and Meeker (2009) is often used. In particular,

$$\mu(t; x) = \frac{\alpha}{1 + \left[\frac{t}{\eta(x)}\right]^{\gamma}}, \tag{4.7}$$

where α is the initial degradation level, $\eta(x) = \exp(\nu_0 + \nu_1 x)$ is the scale factor based on the Arrhenius model, and γ is the shape parameter determining the steepness of the degradation path.

Let $p = y_f/\alpha$ be the proportion of decreasing for the failure threshold from the initial degradation level. Based on the model in (4.7), the mean time to failure at x, $m(x)$, is obtained by solving $\mu[m(x); x] = p\alpha$. Specifically, the temperature-time relationship is

$$\log_{10}[m(x)] = \beta_0 + \beta_1 x, \tag{4.8}$$

where

$$\beta_0 = \frac{\nu_0}{\log(10)} + \frac{1}{\gamma \log(10)} \log\left[\frac{1-p}{p}\right], \quad \text{and} \quad \beta_1 = \frac{\nu_1}{\log(10)}.$$

When $p = 1/2$, β_0 reduces to $\nu_0/\log(10)$. As a result, the TI at t_d can be computed as

$$R = \frac{\beta_1}{\log_{10}(t_d) - \beta_0} - 273.16. \tag{4.9}$$

The estimation of the model in (4.6) is done by ML method. The error term is modeled

as

$$\varepsilon_{ijk} \sim \mathrm{N}(0, \sigma^2), \quad \text{and} \quad \mathrm{Corr}(\varepsilon_{ijk}, \varepsilon_{ijk'}) = \rho, \ k \neq k'. \tag{4.10}$$

The parameter ρ represents the within-batch correlation. The unknown parameters are denoted by $\boldsymbol{\theta} = (\nu_0, \nu_1, \alpha, \gamma, \sigma, \rho)'$. The likelihood is

$$L(\boldsymbol{\theta}) = \prod_{i,j} (2\pi)^{-\frac{n_{ij}}{2}} |\Sigma_{ij}|^{-\frac{1}{2}} \exp\left\{ -\frac{1}{2} [\boldsymbol{y}_{ij} - \boldsymbol{\mu}(t_j, x_i)]' \Sigma_{ij}^{-1} [\boldsymbol{y}_{ij} - \boldsymbol{\mu}(t_j, x_i)] \right\}, \tag{4.11}$$

where $\boldsymbol{y}_{ij} = (y_{ij1}, \cdots, y_{ijn_{ij}})'$ is the corresponding degradation measurements and they follow the multivariate normal distribution with mean vector $\boldsymbol{\mu}(t_{ij}; x_i)$, an $n_{ij} \times 1$ vector of $\mu(t_{ij}; x_i)$'s, and covariance matrix Σ_{ij}. Here Σ_{ij} is an $n_{ij} \times n_{ij}$ matrix with σ^2 on the diagonal elements and $\rho\sigma^2$ on the off-diagonal elements. Parameter estimates $\widehat{\boldsymbol{\theta}}$ are obtained by maximizing (4.11). The estimate of R is obtained by evaluating (5.5) at the estimate of $\widehat{\boldsymbol{\theta}}$.

The parametric model can overcome the shortcoming of the traditional method, and allows for statistical inference. However, for the parametric method, one needs to find an appropriate form for $\mu(t_{ij}; x_i)$.

4.3.3 The Semiparametric Method

Xie et al. (2018) propose the following semi-parametric functional forms for $\mu(t_{ij}; x_i)$ for the model in (4.6). That is,

$$\mu(t_{ij}; x_i) = g\left[\eta_i(t_{ij}; \beta); \boldsymbol{\gamma}\right], \tag{4.12}$$

$$\eta_i(t; \beta) = \frac{t}{\exp(\beta s_i)}, \quad s_i = x_i - x_{\max}, \tag{4.13}$$

Here, $g(\cdot)$ is a monotonic decreasing function with parameter vector $\boldsymbol{\gamma}$, and β is the parameter

for the temperature effect. The quantity

$$x_{\max} = \frac{1}{\max_i\{A_i\} + 273.16}$$

is the transformed value of the highest level of temperature. At the highest temperature level, $s_{\max} = x_{\max} - x_{\max} = 0$, then

$$\mu(t; x_{\max}) = g(t; \boldsymbol{\gamma}).$$

Thus, the function $g(\cdot)$ is interpreted as the baseline degradation path. The advantage of using the maximum temperature level as the baseline is that its degradation level will reach the failure threshold in most ADDT. The $g(\cdot)$ is constructed nonparametrically by monotonic splines, which is the nonparametric component of the model. The use of the monotonic splines retains the physical meaning of the degradation mechanism (i.e., monotonicity), and it is also flexible because one does not need to find a parametric form for the degradation paths. The Arrhenius model is used for describing the acceleration effect, which is the parametric component of the model. Thus, the model in (4.12) is called a semiparametric model.

The distribution of the error terms ε_{ijk} are specified in (4.10). Let $\boldsymbol{\theta} = (\boldsymbol{\gamma}', \beta, \sigma, \rho)'$ be the vector containing all unknown parameters. The estimation of $\boldsymbol{\theta}$ is through an iterative procedure that maximizes the loglikelihood function. Detail of monotonic spline construction and parameter estimation are referred to Xie et al. (2018).

Here we derive the TI estimation based on the semiparametric model in (4.12). Let $g_0 = g(0)$ be the initial degradation level and p be the proportion reducing from the initial degradation (i.e., $p = y_f/g_0$). The mean time to failure for the temperature level x is denoted by $m(x)$, which can be solved from

$$g\left[\frac{m(x)}{\exp[\beta(x - x_{\max})]}\right] = pg_0.$$

We obtain the temperature time relationship as,

$$m(x) = g^{-1}(pg_0)\exp[\beta(x - x_{\max})],$$

which is equivalent to

$$\log_{10}[m(x)] = \beta_0 + \beta_1 x. \tag{4.14}$$

Here,

$$\beta_0 = \log_{10}[g^{-1}(pg_0)] - \frac{\beta x_{\max}}{\log(10)}, \quad \text{and} \quad \beta_1 = \frac{\beta}{\log(10)}.$$

The TI is computed as,

$$R = \frac{\beta_1}{\log_{10}(t_d) - \beta_0} - 273.16. \tag{4.15}$$

The estimates of the TI R can be obtained by substituting the estimate of $\boldsymbol{\theta}$ into (4.15).

4.4 An Illustration of Thermal Index Estimation

In this section, we provide an illustration for the TI estimation using the Adhesive Bond B data introduced in Escobar et al. (2003). The computing was done by using the R package "ADDT" by Hong et al. (2015). An detailed introduction to the package is available in Jin et al. (2017).

4.4.1 Degradation Path Modeling

We apply the traditional method, the parametric method, and the semiparametric method to the Adhesive Bond B data. For the traditional method, Figure 4.3 shows the polynomial interpolation for the Adhesive Bond B data, when the failure threshold is set to $p = 50\%$. For the temperature level 50°C, the degradation level has not reached the failure threshold yet. The estimated time to failure m_{50} is not available. Thus, data from this level is discard for

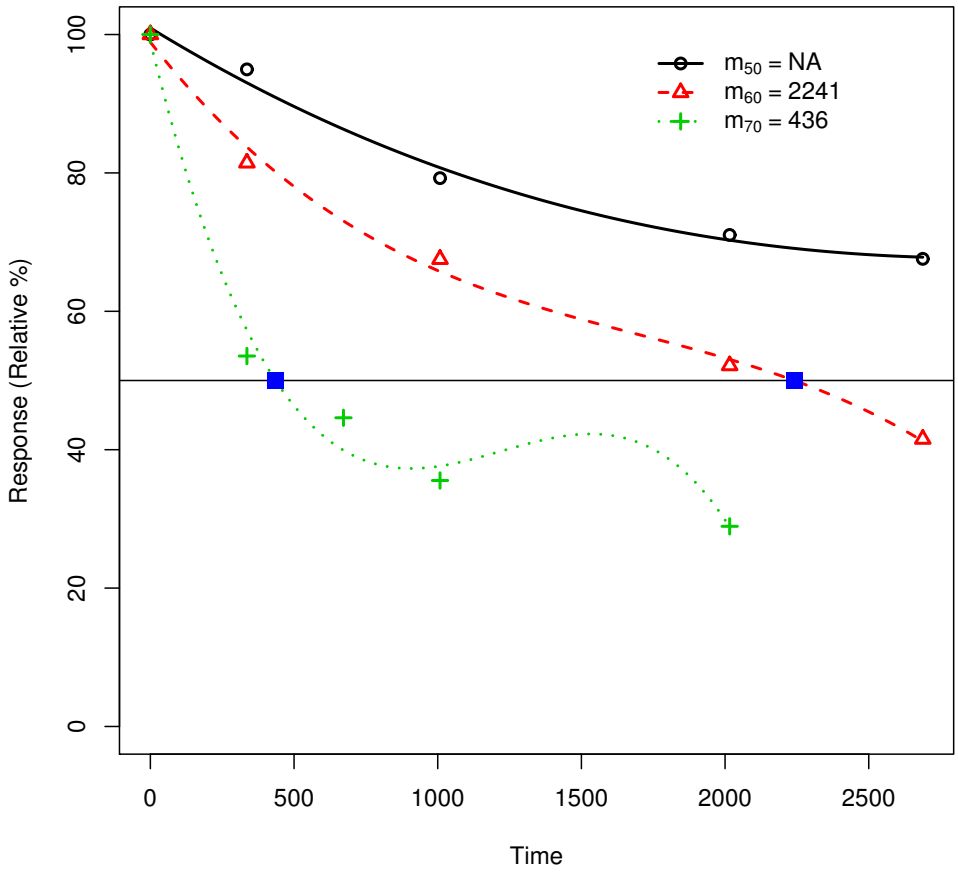


Figure 4.3: Polynomial interpolation for the traditional method applied on the Adhesive Bond B data. The failure threshold is $p = 50\%$.

the analysis. For the parametric and semiparametric methods, however, all data can be used.

Figure 4.4 shows the fitted degradation paths using the parametric method for the Adhesive Bond B data, while Figure 4.5 shows similar results based on the semiparametric method. Both methods provide good fits to the data. The results in Xie et al. (2018) show that the semiparametric method tends to have a better performance to the degradation data.

4.4.2 TI Estimation

For illustration, we compute the TI based on the three methods presented previously. Table 4.2 shows the estimated parameters for the temperature-time relationship, and the corresponding

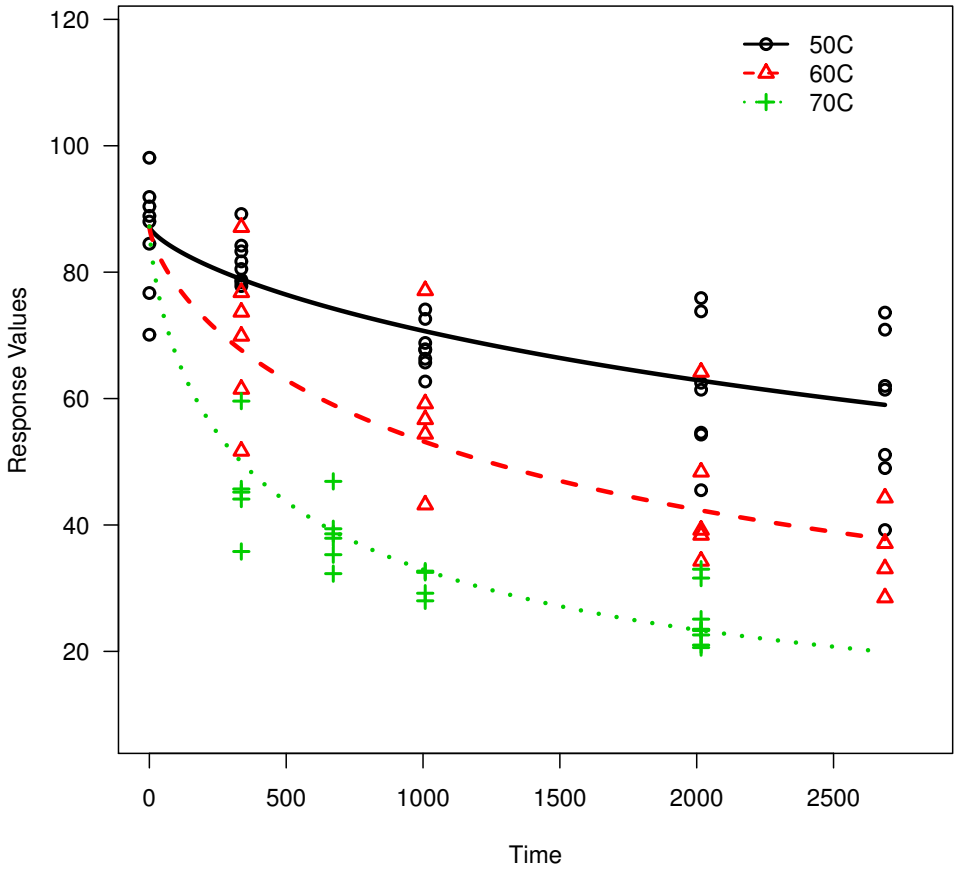


Figure 4.4: Fitted degradation paths using the parametric method for the Adhesive Bond B data (Escobar et al., 2003). The x-axis is time in hours and the y-axis is strength in Newtons.

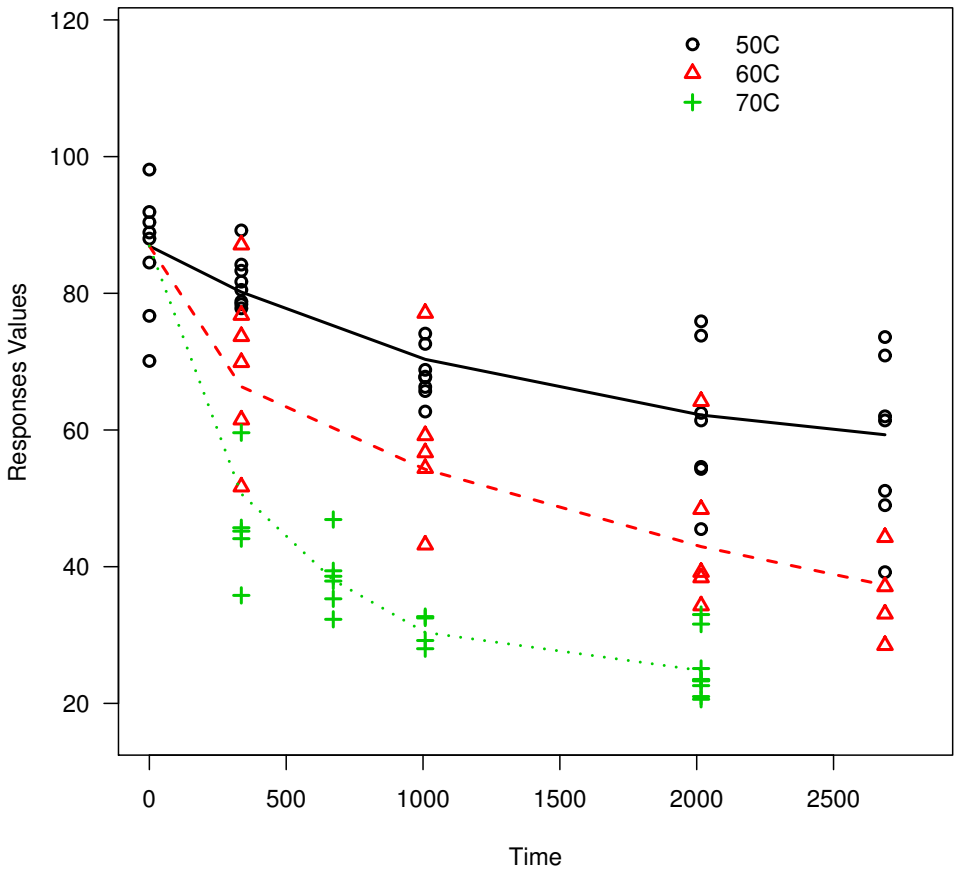


Figure 4.5: Fitted degradation paths using the semiparametric method for the Adhesive Bond B data (Escobar et al., 2003). The x-axis is time in hours and the y-axis is strength in Newtons.

Table 4.2: Estimated parameters for the temperature-time relationship and TI based on the traditional method (TM), the parametric method (PM), and semiparametric method (SPM) for the Adhesive Bond B data, when $t_d = 100{,}000$ and $p = 50\%$.

Methods	β_0	β_1	TI
TM	-21.05	8128.4	39
PM	-16.18	6480.4	33
SPM	-16.81	6697.1	34

TI for the Adhesive Bond B data. In the computing, we use $t_d = 100{,}000$ and $p = 50\%$. Figure 4.6 shows the fitted temperature-time relationship lines using the three methods and the corresponding estimated TI for the Adhesive Bond B data. The results based on the parametric method and semiparametric method are quite close to each other, while the results from traditional method are different from these two methods. Section 4.5 will conduct a simulation study to evaluate the estimation performance.

4.5 Simulation Studies

In this section, simulations are carried out to compare performances of the traditional method, the parametric method, and the semiparametric method in terms of TI estimating. We will consider two settings, under which the parametric model is correctly specified, the parametric model is incorrectly specified.

4.5.1 Simulation Settings

For the first setting (Setting I), we generate degradation from the parametric model in (4.7), and still use the same model in (4.7) to fit the data, which is corresponding to the case that the model is correctly specified. For model (4.7), the parameter values used in the simulation are $\alpha = 9000, \nu_0 = -16, \nu_1 = 12500$, $\gamma = 2$, $\sigma = 1000$, and $\rho = 0.0$. The failure threshold was set to be $p = 50\%$ of the initial degradation level and we used $t_d = 100,000$ hours. Under this configuration, the true TI is $R = 181°\text{C}$. Here we set the correlation to be $\rho = 0$, and fit models without correlations to speed up the simulations.

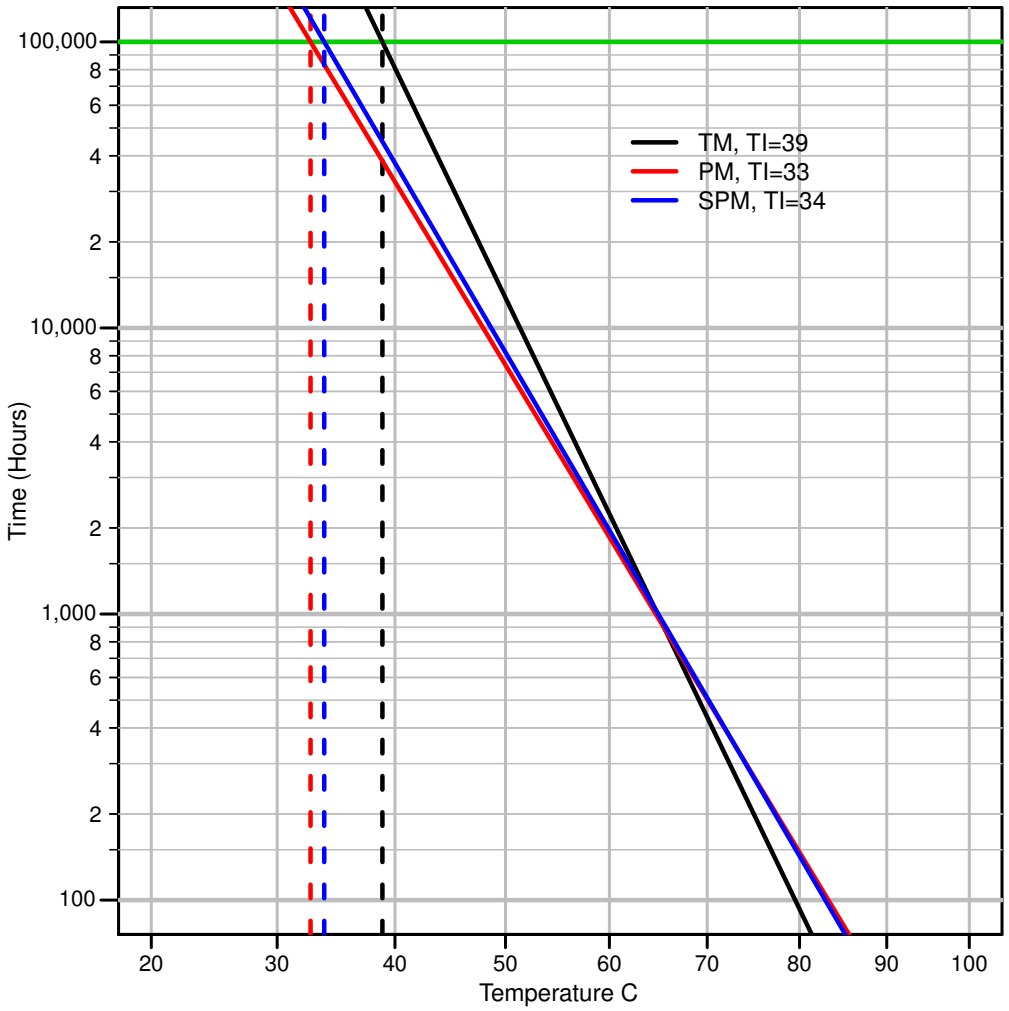


Figure 4.6: Fitted temperature-time relationship lines using the traditional method (TM), the parametric method (PM) and the semiparametric method (SPM), and the corresponding estimated TI for the Adhesive Bond B data.

Table 4.3: The temperature levels and measuring time points for simulation scenarios we use.

Scenarios	Temperature Levels (°C)					Time Points (Hours)				
1: Temp. 3, Time 4		250	260	270		552	1008	2016	3528	
2: Temp. 4, Time 4		250	260	270	280	552	1008	2016	3528	
3: Temp. 4, Time 4	240	250	260	270		552	1008	2016	3528	
4: Temp. 5, Time 4	240	250	260	270	280	552	1008	2016	3528	
5: Temp. 3, Time 5		250	260	270		552	1008	2016	3528	5040
6: Temp. 4, Time 5		250	260	270	280	552	1008	2016	3528	5040
7: Temp. 4, Time 5	240	250	260	270		552	1008	2016	3528	5040
8: Temp. 5, Time 5	240	250	260	270	280	552	1008	2016	3528	5040

For the second setting (Setting II), we generate degradation from a parametric model that is different from (4.7), but the model in (4.7) to fit the data, which is corresponding to the case that the model is incorrectly specified (i.e., model misspecification). In particular, the following model was used to generate data for Setting II,

$$\mu(t;x) = \alpha \exp\left\{-\left[\frac{t}{\eta(x)}\right]\right\}, \tag{4.16}$$

which was used in Li and Doganaksoy (2014) to describe the degradation of polymer strength. Here, $\eta(x) = \exp(\nu_0 + \nu_1 x)$. For model (4.16), the parameter values were set to $\alpha = 9000$, $\nu_0 = -15.6$, $\nu_1 = 12471$, $\sigma = 1000$, and $\rho = 0$. Those values were chosen so that the mean time to failure under 270°C and the true TI are the same as in Setting I.

For each setting, we consider eight scenarios. For different scenarios, we change the number of time points and also vary the temperature levels. Table 4.3 lists the configuration for each scenario. We considered both four time points and five time points to check the sensitivity to different time constraints. The number of temperature levels is from three to five to check the sensitivity to different temperature factors. We also considered the range of temperature levels with higher or lower temperature to check the effect of temperature level in terms of distance from use levels. Similar specification was used in King et al. (2018).

Table 4.4: Estimated results of mean, bias, SD, and RMSE of the TI estimators for the traditional method (TM), the parametric method (PM), and the semiparametric method (SPM) for Setting I: the parametric model is correctly specified.

Scenarios	True TI	Mean			Bias			SD			RMSE		
		TM	PM	SPM	TM	PM	SPM	TM	PM	SPM	TM	PM	SPM
1: Temp. 3, Time 4	181	170	179	179	11	2	2	14	9	9	18	9	9
2: Temp. 4, Time 4	181	178	180	181	3	1	1	8	6	6	8	6	6
3: Temp. 4, Time 4	181	171	181	181	11	0	0	13	5	6	17	5	6
4: Temp. 5, Time 4	181	178	181	181	4	0	0	8	4	4	9	4	4
5: Temp. 3, Time 5	181	179	179	179	2	2	2	9	9	9	10	9	9
6: Temp. 4, Time 5	181	182	180	181	1	1	0	5	5	6	5	5	6
7: Temp. 4, Time 5	181	177	180	181	4	1	1	6	5	5	7	5	5
8: Temp. 5, Time 5	181	180	181	182	1	1	0	4	4	4	4	4	4

4.5.2 Results under the Correct Model

Table 4.4 shows the estimated results of TI estimators for the traditional method (TM), the parametric method (PM), and the semiparametric method (SPM). Specifically, mean, bias, standard deviation (SD), and root of mean squared error (RMSE) for setting I in which the parametric setting is correctly specified. Figure 4.7 visualizes the results in Table 4.4. We observe that those scenarios with more time points and temperature levels tend to have better precision in estimating TI for all methods. Testing at higher temperature levels tends to provide better precision for all the methods. Among the three methods, the traditional method tends to perform worse than the other two methods. This observation for the traditional method is consistent with the findings in King et al. (2018). The performance of the newly added semiparametric is comparable to the parametric method.

4.5.3 Results under a Misspecified Model

Table 4.5 shows the estimated result of mean, bias, SD, and RMSE of the TI estimators for the traditional method, the parametric method, and the semiparametric method for Setting II: the parametric model is incorrectly specified. Figure 4.8 visualizes the results in Table 4.5. We observe similar patterns to Setting I. That is, those scenarios with more time points

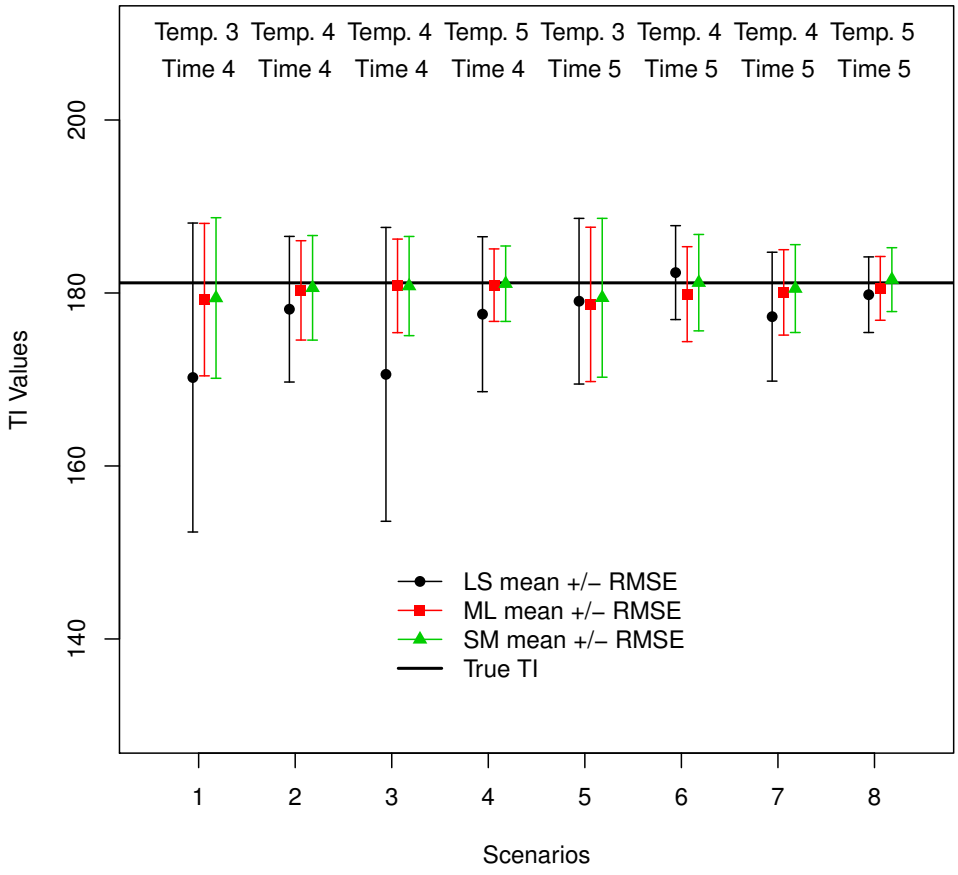


Figure 4.7: Plot of the estimated results of mean, bias, SD, and RMSE of the TI estimators for the traditional method (TM), the parametric method (PM), and the semiparametric method (SPM) for Setting I: the parametric model is correctly specified.

Table 4.5: Estimated results of mean, bias, SD, and RMSE of the TI estimators for the traditional method (TM), the parametric method (PM), and the semiparametric method (SPM) for Setting II: the parametric model is incorrectly specified.

Scenarios	True TI	Mean			Bias			SD			RMSE		
		TM	PM	SPM	TM	PM	SPM	TM	PM	SPM	TM	PM	SPM
1: Temp. 3, Time 4	181	178	180	179	3	1	2	16	11	12	17	11	12
2: Temp. 4, Time 4	181	179	180	180	1	1	1	10	7	8	10	8	8
3: Temp. 4, Time 4	181	176	180	179	4	0	1	17	7	8	17	7	8
4: Temp. 5, Time 4	181	178	180	180	2	0	1	10	5	6	10	5	6
5: Temp. 3, Time 5	181	179	178	178	2	2	3	12	10	11	12	10	12
6: Temp. 4, Time 5	181	178	179	180	3	2	1	8	7	7	9	7	7
7: Temp. 4, Time 5	181	179	181	180	2	0	1	8	6	6	8	6	6
8: Temp. 5, Time 5	181	178	180	180	3	0	0	6	4	5	6	4	5

and temperature levels tend to have better precision in estimating TI for all methods, and the traditional method tends to perform worse than the other two methods, which is also consistent with the findings in King et al. (2018). Surprisingly the parametric method performs well even under model misspecification. Similarly, the performance of the newly added semiparametric is comparable to the parametric method.

4.6 Discussions

In literature, there are three methods available to estimate the TI based on ADDT data, which are the traditional method, the parametric method, and the semiparametric method. In this chapter, we provide a comprehensive review of the three methods and illustrate how the TI can be estimated based on different models. We also conduct a simulation study to show the properties of different methods. The comparisons and discussions in this chapter can be useful for practitioners and future industrial standards. Here, we provide a summary on the pros and cons of each method.

Regarding estimation performance, if there are fewer number of temperature levels/number of time points, the traditional method tends to not performance well. When there are five temperature levels and five time points, the traditional method works well. Both the para-

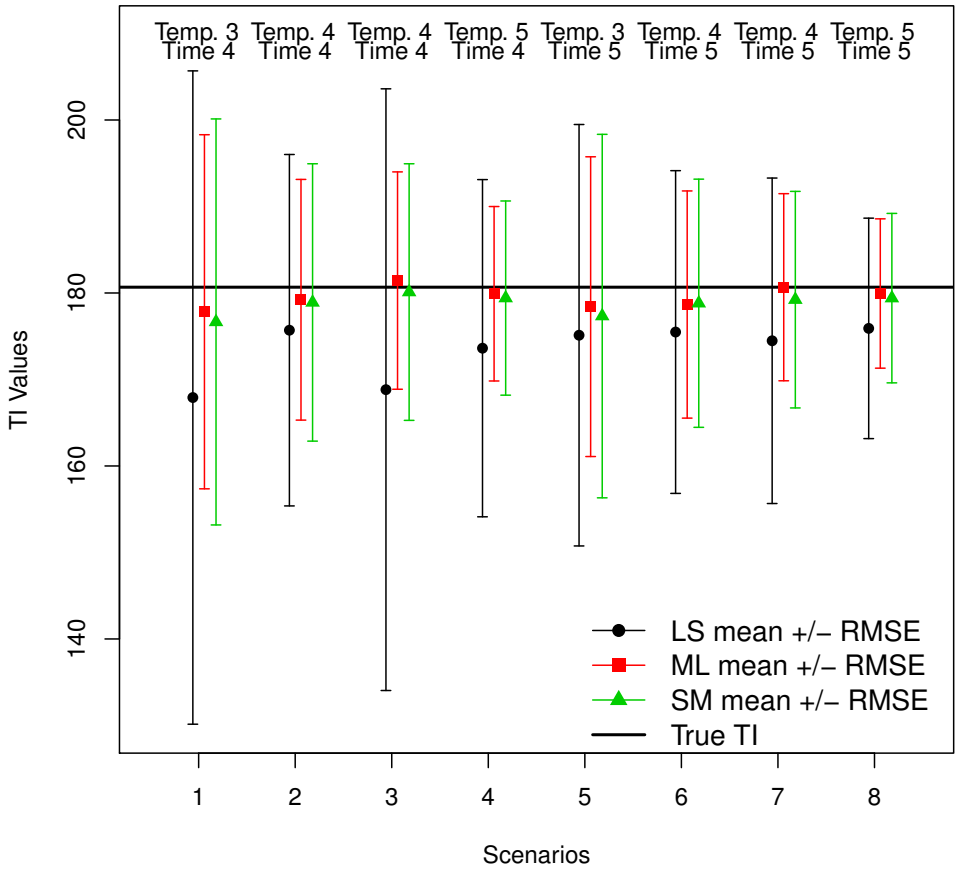


Figure 4.8: Plot of the estimated results of mean, bias, SD, and RMSE of the TI estimators for the traditional method (TM), the parametric method (PM), and the semiparametric method (SPM) for Setting II: the parametric model is incorrectly specified.

metric and semiparametric methods perform better than the traditional methods and their performance are comparable to each other.

Regarding model assumption, the traditional method does not require specific forms for the underlying degradation path because it uses polynomial interpolation. The semiparametric method does not require a specific form but assume the underlying path is monotone and smooth. The parametric method assumes a specific form, which requires the strongest assumption. However, the simulation study show that the parametric is flexible to some extent in model misspecification.

Regarding data use, both the parametric and semiparametric method use all data for analyses, including those have not reached the failure threshold yet. The traditional method will discard the data from the temperature which has not reached the failure threshold yet.

Both the parametric and semiparametric can quantify the uncertainties in the estimation (see King et al. (2018) and Xie et al. (2018) for details). Because the traditional method requires two steps to estimate the TI, it is challenging to quantify the statistical uncertainties.

The semiparametric method is the most computationally intensive one, and the parametric method is in the middle in term of computational time. All the three methods is implemented in an R package "ADDT". Chapter 5 gives a detailed illustration for the use of the package.

In summary, it is of advantages to use the parametric and semiparametric methods in the ADDT analysis and TI estimation. In practice, one can compare the model fitting of both the parametric and semiparametric methods (e.g., AIC values) to determine which models can provide a better description to the ADDT data. Details of model comparisons can be found in Xie et al. (2018).

Bibliography

L. A. Escobar, W. Q. Meeker, D. L. Kugler, and L. L. Kramer. Accelerated destructive degradation tests: Data, models, and analysis. In B. H. Lindqvist and K. A. Doksum, editors, *Mathematical and Statistical Methods in Reliability*, chapter 21. World Scientific Publishing Company, River Edge, NJ, 2003.

Y. Hong, Y. Duan, W. Q. Meeker, D. L. Stanley, and X. Gu. Statistical methods for degradation data with dynamic covariates information and an application to outdoor weathering data. *Technometrics*, 57:180–193, 2015.

Y. Hong, Y. Xie, Z. Jin, and C. King. *ADDT: A Package for Analysis of Accelerated Destructive Degradation Test Data*, 2016. URL `http://CRAN.R-project.org/package=ADDT`. R package version 1.1.

Z. Jin, Y. Xie, Y. Hong, and J. H. Van Mullekom. ADDT: An R package for analysis of accelerated destructive degradation test data. In D. G. Chen, Y. L. Lio, H. K. T. Ng, and T. R. Tsai, editors, *Statistical Modeling for Degradation Data*, chapter 14. Springer, NY: New York, 2017.

C. B. King, Y. Xie, Y. Hong, J. H. Van Mullekom, S. P. DeHart, and P. A. DeFeo. A comparison of traditional and maximum likelihood approaches to estimating thermal indices for polymeric materials. *Journal of Quality Technology*, 50:117–129, 2018.

M. Li and N. Doganaksoy. Batch variability in accelerated-degradation testing. *Journal of Quality Technology*, 46:171–180, 2014.

C. J. Lu and W. Q. Meeker. Using degradation measures to estimate a time-to-failure distribution. *Technometrics*, 34:161–174, 1993.

W. Q. Meeker and L. A. Escobar. *Statistical Methods for Reliability Data*. John Wiley & Sons, Inc., New York, 1998.

W. Q. Meeker, Y. Hong, and L. A. Escobar. Degradation models and data analyses. In *Encyclopedia of Statistical Sciences*. Wiley, 2011.

W. B. Nelson. *Accelerated testing: statistical models, test plans, and data analysis*. John Wiley & Sons, 1990.

C. Park and W. J. Padgett. Accelerated degradation models for failure based on geometric Brownian motion and gamma processes. *Lifetime Data Analysis*, 11:511–527, 2005.

C.-C. Tsai, S.-T. Tseng, N. Balakrishnan, and C.-T. Lin. Optimal design for accelerated destructive degradation tests. *Quality Technology and Quantitative Management*, 10:263–276, 2013.

UL746B. *Polymeric Materials - Long Term Property Evaluations, UL 746B*. Underwriters Laboratories, Incorporated, 2013.

I. Vaca-Trigo and W. Q. Meeker. A statistical model for linking field and laboratory exposure results for a model coating. In J. Martin, R. A. Ryntz, J. Chin, and R. A. Dickie, editors, *Service Life Prediction of Polymeric Materials*, chapter 2. Springer, NY: New York, 2009.

X. Wang and D. Xu. An inverse Gaussian process model for degradation data. *Technometrics*, 52:188–197, 2010.

G. A. Whitmore. Estimation degradation by a Wiener diffusion process subject to measurement error. *Lifetime Data Analysis*, 1:307–319, 1995.

Y. Xie, C. B. King, Y. Hong, and Q. Yang. Semiparametric models for accelerated destructive degradation test data analysis. *Technometrics*, 60:222–234, 2018.

Z. Ye and M. Xie. Stochastic modelling and analysis of degradation for highly reliable products. *Applied Stochastic Models in Business and Industry*, 31:16–32, 2015.

Chapter 5 ADDT: An R Package for Analysis of Accelerated Destructive Degradation Test Data

Abstract

Accelerated destructive degradation tests (ADDT) are often used to collect necessary data for assessing the long-term properties of polymeric materials. Based on the collected data, a thermal index (TI) is estimated. The TI can be useful for material rating and comparisons. The R package `ADDT` provides the functionalities of performing the traditional method based on the least-squares approach, the parametric method based on maximum likelihood estimation, and the semiparametric method based on spline models for analyzing ADDT data and then estimating the TI for polymeric materials. In this chapter, we provide a detailed introduction for the `ADDT` package. We provide a step-by-step illustration for the use of functions in the package. Publicly available datasets are used for illustrations.

Key Words: Adhesive Bond B, Arrhenius model, degradation process, least squares, polymeric materials, semi-parametric.

5.1 Introduction

Accelerated destructive degradation tests (ADDT) are commonly used to collect data to access the long-term properties of polymeric materials (e.g., UL746B (2013)). Based on the collected ADDT data, a thermal index (TI) is estimated using a statistical model. In practice, the TI can be useful for material rating and comparisons. In literature, there are three methods available for ADDT data modeling and analysis: the traditional method based on the least-squares approach, the parametric method based on maximum likelihood estimation, and the semiparametric method based on spline models. The chapter in Xie et al. (2017) provides a comprehensive review for the three methods for ADDT data analysis and compares the corresponding TI estimation procedures via simulations.

The R package `ADDT` in Hong et al. (2016) provides the functionalities of performing the three methods and their corresponding TI estimation procedures. In this chapter, we provide a detailed introduction for the `ADDT` package. We provide a step-by-step illustration for the use of functions in the package. We also use publicly available datasets for illustrations.

The rest of the chapter is organized as follows. Section 5.2 introduces the three methods, the corresponding TI procedures, and the implementations in the R package. The Adhesive Bond B data (Escobar et al. (2003)) is used to do a step-by-step illustration. Section 5.3 provides a full analysis for the Seal Strength data (Li and Doganaksoy (2014)) so that users can see a typical ADDT modeling and analysis process. Section 5.4 contains some concluding remarks.

5.2 The Statistical Methods

5.2.1 Data

In most applications, an ADDT dataset typically includes degradation measurements under different measuring time points, and accelerating variables such as temperature and voltage. In the `ADDT` package, there are four publicly available datasets ready for users to do analysis,

which are the Adhesive Bond B data in Escobar et al. (2003), the Seal Strength data in Li and Doganaksoy (2014), the Polymer Y data in Tsai et al. (2013), and the Adhesive Formulation K data in Xie et al. (2018). Users can load those datasets by downloading, installing the package `ADDT` and appropriately calling the *data* function. The following gives some example R codes.

```
>install.packages("ADDT")
>library(ADDT)
>data(AdhesiveBondB)
>data(SealStrength)
>data(PolymerY)
>data(AdhesiveFormulationK)
>AdhesiveBondB
>SealStength
```

Table 5.1 shows the Adhesive Bond B dataset. The first column is the acceleration variable, temperature in Celsius. Time points that used to measure the degradation and the degradation values are listed in columns 2 and 3 correspondingly. We illustrate the Adhesive Bond B data in Fig 5.1. To use the R `ADDT` package, users need to format the data in the same form as the dataset shown in Table 5.1.

Another dataset that has been frequently used is the Seal Strength data where the strength from ten different seals were measured at five different time points under four different temperature levels. Seal Strength data is shown in Table 5.2. We will use the Adhesive Bond B data and Seal Strength data to illustrate the use of the `ADDT` package.

5.2.2 The Traditional Method

The traditional method using the least-squares approach is widely accepted and used in various industrial applications. The traditional method utilizes two-step approach to describe the accelerating variable and degradation response relationship as well as the failure time and

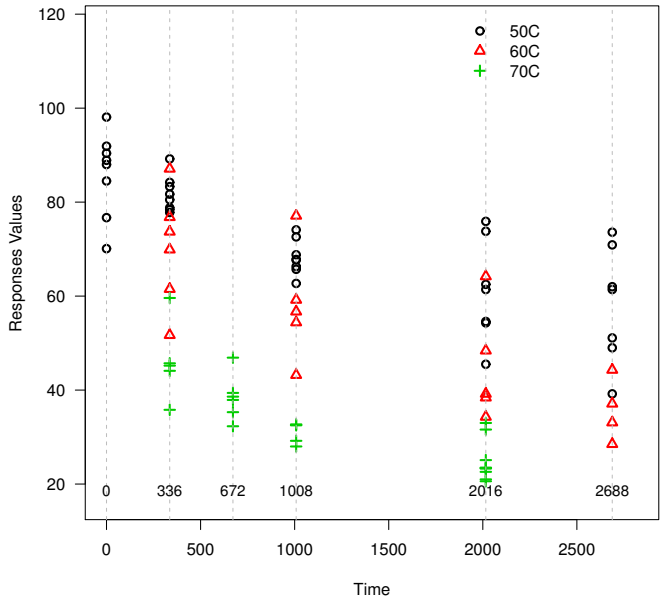


Figure 5.1: Graphical representation of the Adhesive Bond B dataset. The x-axis stands for the time in hour while y-axis represents the degradation values.

Table 5.1: The Adhesive Bond B data from Escobar et al. (2003), which contains the testing of results of an ADDT for the strength of Adhesive Bond B.

TempC	TimeH	Response	TempC	TimeH	Response	TempC	TimeH	Response
50	0	70.1	50	2016	62.5	60	2688	37.1
50	0	76.7	50	2016	73.8	60	2688	44.3
50	0	84.5	50	2016	75.9	70	336	35.8
50	0	88.0	50	2688	39.2	70	336	44.1
50	0	88.9	50	2688	49.0	70	336	45.2
50	0	90.4	50	2688	51.1	70	336	45.7
50	0	91.9	50	2688	61.4	70	336	59.6
50	0	98.1	50	2688	62.0	70	672	32.3
50	336	77.8	50	2688	70.9	70	672	35.3
50	336	78.4	50	2688	73.6	70	672	37.9
50	336	78.8	60	336	51.7	70	672	38.6
50	336	80.5	60	336	61.5	70	672	39.4
50	336	81.7	60	336	69.9	70	672	46.9
50	336	83.3	60	336	73.7	70	1008	28.0
50	336	84.2	60	336	76.8	70	1008	29.2
50	336	89.2	60	336	87.1	70	1008	32.5
50	1008	62.7	60	1008	43.2	70	1008	32.7
50	1008	65.7	60	1008	54.4	70	2016	20.6
50	1008	66.3	60	1008	56.7	70	2016	21.0
50	1008	67.7	60	1008	59.2	70	2016	22.6
50	1008	67.8	60	1008	77.1	70	2016	23.3
50	1008	68.8	60	2016	34.3	70	2016	23.4
50	1008	72.6	60	2016	38.4	70	2016	23.5
50	1008	74.1	60	2016	39.2	70	2016	25.1
50	2016	45.5	60	2016	48.4	70	2016	31.6
50	2016	54.3	60	2016	64.2	70	2016	33.0
50	2016	54.6	60	2688	28.5			
50	2016	61.4	60	2688	33.1			

Table 5.2: The Seal Strength data in Li and Doganaksoy (2014). The table shows the strength of seal samples that were measured at five different time points under four different temperature levels.

TempC	TimeH	Response	TempC	TimeH	Response	TempC	TimeH	Response
100	0	28.74	300	1680	10.34	250	3360	14.23
100	0	25.59	300	1680	13.24	250	3360	12.83
100	0	22.72	300	1680	8.57	250	3360	13.02
100	0	22.44	300	1680	11.93	250	3360	16.74
100	0	29.48	300	1680	13.76	250	3360	12.11
100	0	23.85	300	1680	16.44	250	3360	12.24
100	0	20.24	300	1680	14.81	250	3360	18.97
100	0	22.33	300	1680	11.50	250	3360	15.29
100	0	21.70	300	1680	11.92	250	3360	14.38
100	0	27.97	300	1680	10.30	250	3360	14.80
200	840	52.52	350	1680	5.78	300	3360	2.89
200	840	30.23	350	1680	5.90	300	3360	3.31
200	840	31.90	350	1680	6.99	300	3360	1.81
200	840	33.15	350	1680	7.94	300	3360	1.61
200	840	34.26	350	1680	7.06	300	3360	2.65
200	840	31.82	350	1680	5.13	300	3360	2.83
200	840	27.10	350	1680	5.80	300	3360	2.70
200	840	30.00	350	1680	6.20	300	3360	2.79
200	840	26.96	350	1680	5.30	300	3360	1.83
200	840	42.73	350	1680	6.34	300	3360	3.08
250	840	28.97	200	2520	9.47	350	3360	1.24
250	840	35.01	200	2520	13.61	350	3360	1.57
250	840	27.39	200	2520	8.95	350	3360	2.06
250	840	36.66	200	2520	8.61	350	3360	1.56
250	840	27.91	200	2520	10.16	350	3360	1.94
250	840	31.03	200	2520	8.82	350	3360	1.39
250	840	32.65	200	2520	8.84	350	3360	1.91
250	840	35.08	200	2520	10.73	350	3360	1.44
250	840	28.05	200	2520	10.63	350	3360	1.61
250	840	33.54	200	2520	7.70	350	3360	1.50
300	840	10.63	250	2520	9.59	200	4200	14.53
300	840	8.28	250	2520	14.37	200	4200	17.95
300	840	13.46	250	2520	12.08	200	4200	11.90
300	840	13.47	250	2520	11.79	200	4200	17.00
300	840	9.44	250	2520	17.69	200	4200	15.56
300	840	7.66	250	2520	14.05	200	4200	18.07
300	840	11.16	250	2520	17.08	200	4200	13.96
300	840	8.70	250	2520	11.52	200	4200	13.57
300	840	9.44	250	2520	13.03	200	4200	16.35
300	840	12.23	250	2520	18.37	200	4200	18.76
350	840	13.79	300	2520	3.86	250	4200	14.75
350	840	15.10	300	2520	4.76	250	4200	11.54
350	840	20.58	300	2520	5.32	250	4200	11.57
350	840	18.20	300	2520	3.74	250	4200	10.83
350	840	16.64	300	2520	4.58	250	4200	12.78
350	840	10.93	300	2520	3.62	250	4200	10.14
350	840	12.28	300	2520	3.58	250	4200	11.45
350	840	18.65	300	2520	3.47	250	4200	12.91
350	840	20.80	300	2520	3.29	250	4200	13.06
350	840	15.04	300	2520	3.63	250	4200	6.76
200	1680	31.37	350	2520	1.34	300	4200	1.95
200	1680	37.91	350	2520	0.92	300	4200	1.55
200	1680	38.03	350	2520	1.31	300	4200	2.19
200	1680	42.21	350	2520	1.76	300	4200	2.00
200	1680	32.64	350	2520	1.30	300	4200	2.00
200	1680	32.10	350	2520	1.47	300	4200	2.33
200	1680	32.37	350	2520	1.11	300	4200	1.80
200	1680	33.59	350	2520	1.25	300	4200	2.34
200	1680	26.46	350	2520	1.02	300	4200	1.88
200	1680	33.69	350	2520	1.30	300	4200	2.66
250	1680	14.29	200	3360	26.72	350	4200	0.27
250	1680	20.16	200	3360	21.24	350	4200	0.20
250	1680	22.35	200	3360	22.76	350	4200	0.26
250	1680	21.96	200	3360	24.39	350	4200	0.26
250	1680	13.67	200	3360	15.93	350	4200	0.27
250	1680	14.40	200	3360	23.90	350	4200	0.18
250	1680	22.37	200	3360	22.09	350	4200	0.13
250	1680	13.08	200	3360	23.69	350	4200	0.20
250	1680	17.81	200	3360	23.67	350	4200	0.13
250	1680	17.82	200	3360	20.94	350	4200	0.21

accelerating variable relationship (i.e., the temperature-time relationship). The TI can be obtained by using the fitted temperature-time relationship. In particular, for each temperature level, indexed by i, we find the mean time to failure m_i satisfies the following equation.

$$a_{0i} + a_{1i}m_i + a_{2i}m_i^2 + a_{3i}m_i^3 = y_f, i = 1, \cdots, n, \tag{5.1}$$

where time y_f is the failure threshold fixed at a certain proportion of the initial degradation value measurement according to particular applications, and $(a_{0i}, a_{1i}, a_{2i}, a_{3i})'$ are coefficients. Here n is the number of temperature levels. The temperature-time relationship is expressed as

$$\log_{10}(m_i) = \beta_0 + \beta_1 x_i, i = 1, \cdots, n, \tag{5.2}$$

which is based on the Arrhenius relationship to extrapolate to the normal use condition. With the parameterizations in this temperature-time relationship, the TI, denoted by R, can be estimated as:

$$R = \frac{\beta_1}{\log_{10}(t_d) - \beta_0} - 273.16\,. \tag{5.3}$$

where β_0 and β_1 are the same with the coefficients from equation 5.2, and t_d is the target time, usually $t_d = 100{,}000$ is used.

In the R package `ADDT`, we implement the traditional method by using:

```
>addt.fit.lsa<-addt.fit(Response~TimeH+TempC,data=Adh
esiveBondB, proc="LS", failure.threshold=70)
```

The *addt.fit* function in `ADDT` package fits the traditional model automatically when users specify *proc = "LS"* argument. In function *addt.fit*, other arguments include:

- *formula*: We use *Response* $\sim$ *TimeH+TempC* to present the model formula. The *Response*, *TimeH*, and *TempC* specify the response, time, and temperature columns in the

dataset, respectively. Note that the order of *TimeH* and *TempC* can not be exchanged in the formula.

- *data*: The name of the dataset for analysis. The dataset should have the same layout as the Adhesive Bond B in Table 5.1. Specifically, the order of the three columns should be the same as Adhesive Bond B, which is TempC, TeimH, and Response.

- *initial.value*: We need response measurements at time point 0 to compute the initial degradation level in the model. If the data does not contain that information, user must supply the *initial.value*, otherwise,the function will give an error message.

- *failure.threshold*: This argument set the point when soft failure occurs. The default value of the soft failure threshold is 70% of the initial value in the `ADDT` package examples. Note that in industrial standard such as UL746B (2013), the failure threshold is usually 50%.

- *time.rti*: The *addt.fit* function allows users to specify the expected time associated with the TI. The default value for *time.rti* is t_d =100,000 hours.

- *method*: This argument specifies the method that is used in the optimization process. Details can be found in *optim* function in R. The default value is *"Nelder-Mead"*.

- *subset*: This argument allows the users to specify a subset of the dataset for modeling.

The above arguments are the basic model inputs to run *addt.fit*, when *proc="LS"*. Other methods, *proc ="ML"* (parametric method) and *proc ="SemiPara"* (the semiparametric method) also require the same arguments. However, there are additional arguments for the other two methods and we will introduce them in Sections 5.2.3 and 5.2.4.

We store the model fitting results in the *addt.fit.lsa* in this example. Users are able to print the model summary table and plots upon appropriate call. Examples are listed below:

```
> summary(addt.fit.lsa)
```

```
Least Squares Approach:
beta0     beta1
-13.7805 5535.0907
est.TI: 22
Interpolation time:
Temp      Time
[1,]   50 2063.0924
[2,]   60  797.1901
[3,]   70  206.1681
```

The *summary* function for *proc ="LS"* provides the parameter estimates and interpolated mean time to failure for the corresponding temperature levels. In the Adhesive Bond B example, the parameter estimates are $\hat{\beta}_0 = -13.7805$ and $\hat{\beta}_1 = 5535.0907$ for the temperature-time relationship. Estimated mean time to failure for temperature level 50°C, 60°C, and 70°C, are 2063.092, 797.190 and 206.168 hours, respectively. The estimated TI is 22°C in this example. Figure 5.2 shows the fitted polynomial curves for each temperature levels and the corresponding interpolated mean time to failure, according to least-squares method. The R code that is used to plot the results is shown below.

```
>plot(addt.fit.lsa, type="LS")
```

5.2.3 The Parametric Method

Different from the two-step approach in the traditional method, for the parametric method, one uses a parametric model to describe the degradation path. The maximum likelihood (ML) method is then used to estimate the unknown parameters in the model. In particular, we assume that degradation measurement y_{ijk} at time t_{ij} for temperature level i follows the model:

$$y_{ijk} = \mu(t_{ij}; x_i) + \epsilon_{ijk}, i = 1, \cdots, n, j = 1, \cdots, n_i, k = 1, \cdots, n_{ij},$$

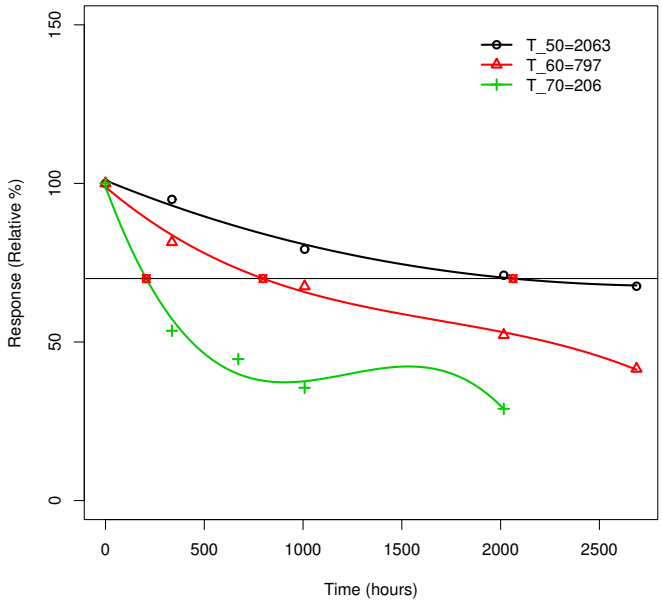


Figure 5.2: Plot of the fitted polynomial curves for each temperature level, and the corresponding interpolated time to failures. The horizontal dark line presents the failure threshold. The y-axis shows the relative value of material strength.

where

$$x_i = \frac{1}{\text{TempC}_i + 273.16},$$

TempC_i is the temperature level, the value 273.16 is used to convert the temperature to Kelvin temperature scale. Here n is the number of temperature levels, n_i is the number of time points for level i, and n_{ij} is the number samples tested under the temperature time combination and ϵ_{ijk} is the error term. For polymer materials, the following parametric assumption for $\mu(t;x)$ (e.g., Vaca-Trigo and Meeker (2009)) is used

$$\mu(t;x) = \frac{\alpha}{1 + [\frac{t}{\eta(x)}]^{\gamma}}, \tag{5.4}$$

where α represents the initial degradation, and γ is a shape parameter. Here,

$$\eta(x) = \exp(\upsilon_0 + \upsilon_1 x).$$

is the scale factor that is based on the Arrhenius relationship. By the parametric specification, the ML method is then used to estimate the parameters. King et al. (2018) performed a comprehensive comparison between the traditional method and the parametric method. Xie et al. (2017) performed a comprehensive comparison among the three methods in term of TI estimation.

For the model in (5.4), the TI is calculated as follows:

$$R = \frac{\beta_1}{\log_{10}(t_d) - \beta_0} - 273.16, \tag{5.5}$$

where β_0 and β_1 are defined as:

$$\beta_0 = \frac{\nu_0}{\log(10)} + \frac{1}{\gamma \log(10)} \log\left[\frac{1-p}{p}\right], \quad \text{and} \quad \beta_1 = \frac{\nu_1}{\log(10)}.$$

To fit the parametric model, one can use the following command:

```
> addt.fit.mla<-addt.fit(Response~TimeH+TempC,data=Adh
esiveBondB,proc="ML", failure.threshold=70)
```

Similar to the "LS" case, here we provide an example of *ML* method based on the parametric method implemented in R. Using the same dataset Adhesive Bond B, we now change the *proc* argument to *proc ="ML"* so that the parametric model is used. The model results are stored in *addt.fit.mla*. Argument setups are almost the same as those in *addt.fit* for the case of *proc = "LS"* except for additional arguments: *"starts"* and *"fail.thres.vec"*. In particular,

- *starts*: It provides a set of starting values for the ML estimation procedure. If this value is not supplied, the function will use the least-squares method to estimate for a set of starting values for the ML estimation.

- *fail.thres.vec*: If the user does not specify *starts* argument, the user may instead provide a vector of two different *failure.thresholds*. The least-squares procedure is then used for the two different failure thresholds to produce starting values for the ML procedure.

For the model results in *addt.fit.lma*, we not only have the parameter estimates as in the *LS* example, but also have confidence intervals for the model parameters as well as the TI. The following shows the summary information of the model fitting.

```
> summary(addt.fit.mla)

Maximum Likelihood Approach:
Call:
lifetime.mle(dat = dat0, minusloglik = minus.loglik.ki
netics,  starts = starts, method = method, control =
list(maxit = 1e+05))

Parameters:
mean       std  95% Lower  95% Upper
alpha    87.2004    2.5920    82.2653    92.4315
beta0   -37.2360    4.6450   -46.3401   -28.1318
beta1 14913.1628 1561.1425 11853.3235 17973.0022
gamma     0.7274    0.0870     0.5753     0.9195
sigma     8.2017    0.6405     7.0377     9.5581
rho       0.0000    0.0003    -0.0006     0.0006

Temperature-Time Relatonship:
beta0      beta1
-16.6830   6478.5641
```

```
TI:
est       std 95% Lower 95% Upper
25.6183    3.0980   19.5465   31.6902

Loglikelihod:
[1] -288.9057
```

By applying *summary* function to the *addt.fit* results, we have the ML estimates for $\alpha, \upsilon_0, \upsilon_1, \gamma, \sigma$, and ρ along with their standard deviation as well as the associated 95% confidence intervals based on large-sample approximations. The log likelihood values for the final model is also printed for model comparisons.

The summary table will perform the TI estimates and confidence interval calculation automatically by assigning the default confidence level as 95%. Users are able to change the confidence level to other values by using the function *addt.confint.ti.mle* and specifying the desired value for *conflevel*. In particular,

```
> addt.confint.ti.mle(addt.fit.mla, conflevel = 0.99)
```

provides an example of customizing confidence level for TI estimates. It shows that the 99% confidence interval for TI and the confidence interval is wider than using 95% as the confidence level. The results are shown as follows.

```
est.      s.e.     lower     upper
25.618     3.097     17.638    33.598
```

Similar to *LS* method, we offer users options to visualize the model results. For *ML* method, one can plot the fitted lines along with the data by employing *plot.addt.fit*. Figure 5.3 shows the illustration of the fitting results of *plot.addt.fit*.

```
> plot(addt.fit.mla, type="ML")
```

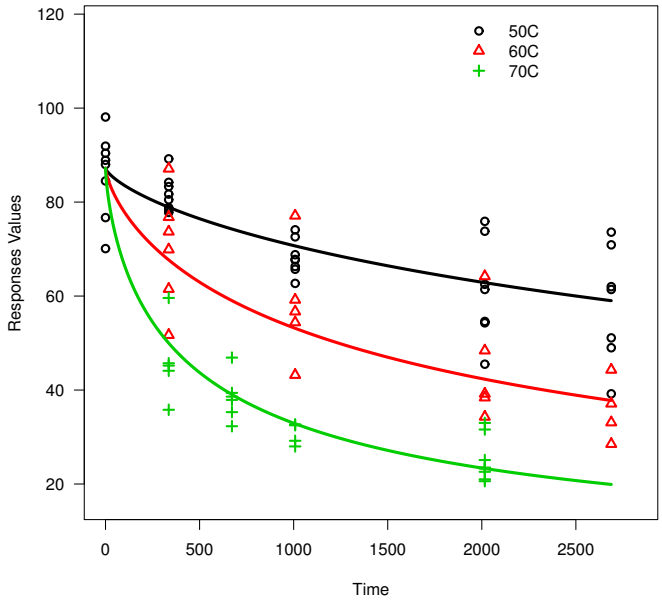


Figure 5.3: Plot of the original dataset of Adhesive Bond B as well as the fitted degradation paths based on the parametric model. The black line, red line and green line stand for fitted lines at 50, 60 and 70 degree, respectively.

5.2.4 The Semiparametric Method

Different from the traditional method and parametric method introduced in Sections 5.2.2 and 5.2.3, with the use of a nonparametric model for the baseline degradation path, the semiparametric method is applicable to different materials. In addition, parametric part of the model using the Arrhenius relationship retains the extrapolation capacity when at use condition of the material that is of interest. Similarly to the parametric model, we assume degradation measurement follows the model:

$$y_{ijk} = \mu(t_{ij}, x_i; \theta) + \epsilon_{ijk},$$

where

$$x_i = -\frac{11605}{\text{TempC}_i + 273.16},$$

and $\boldsymbol{\theta}$ stands for all the parameters in the model. We use the semiparametric model structure to describe the degradation path. In particular, the degradation path is modeled as

$$\mu(t_{ij}, x_i) = g[\eta_i(t_{ij}; \beta); \gamma], \tag{5.6}$$

and the scale factor is

$$\eta_i(t; \beta) = \frac{t}{\exp(\beta s_i)}, \tag{5.7}$$

with acceleration parameter β. In equation 5.7, we define $s_i = x_{max} - x_i$ where x_{max} is the transformed value of the highest level of temperature. We assume the error terms follow normal distribution with variance σ^2 and the correlations between two error terms are ρ. That is,

$$\epsilon_{ijk} \sim \mathrm{N}(0, \sigma^2),$$

and

$$\mathrm{Corr}(\epsilon_{ijk}, \epsilon_{ijk'}) = \rho. \tag{5.8}$$

We assume $k \neq k'$ in the error terms correlations in (5.8). In (5.6), $g(\cdot)$ is a monotonically decreasing function modeled by splines model with parameter vector γ. See Xie et al. (2018) for more details on the semiparametric method.

As a more flexible method designated to a wide variety of materials, the non-parametric component is used to build the baseline degradation path. With inner knots $d_1 \leq d_2 \leq \cdots \leq d_N$ and boundary knots d_0, d_{N+1}, the l-th B-spline basis function with a degree of q can be

expressed at z by recursively building the following models:

$$\begin{aligned} B_{0,l}(z) &= 1(d_l \leq z \leq d_{l+1}) B_{q,l}(z) \\ &= \frac{z - d_l}{d_{l+q} - d_l} B_{q-1,l}(z) + \frac{d_{l+q+q} - z}{d_{l+q+1} - d_{l+1}} B_{q-1,l+1}(z). \end{aligned}$$

The degradation can be expressed as follows.

$$y_{ijk} = \sum_{l=1}^{p} \gamma_l B_{q,l}[\eta_i(t_{ij}; \beta)] + \epsilon_{ijk},$$

where $\eta(t; \beta)$ accounts for the parametric part while $g(\cdot)$ is the non-parametric component which is constrained to be monotonically decreasing to retain the meanings of the degradation process.

Similarly to the "LS" and "ML" methods, we implement the semiparametric model in R. In *addt.fit*, *proc="SemiPara"* enables users to fit a semiparametric model to the degradation data as we discussed above. In particular,

```
>addt.fit.semi<-addt.fit(Response~TimeH+TempC,data=Adh
esiveBondB,proc="SemiPara",failure.threshold=70)
```

Other than the arguments we introduced for *proc = "LS"* and *proc = "ML"*, there is an other unique option in the *addt.fit* when *proc = "SemiPara"* is called. That is:

- *semi.control*: This argument contains a list of control parameters regarding the *Semi-Para* option. Users are able to specify the model assumptions like correlation *rho*. In *semi.control = list(cor = F, ...)*, the default value is to exclude the correlation term in the model (i.e., $\rho = 0$). If *cor = T*, then there will be a correlation term in the semiparametric model.

Summary results of semiparametric model object given by *addt.fit* include $\hat{\beta}$, $\hat{\rho}$, knots that were used by the model, log-likelihood and AICc for the final model, which are both model evaluation quantities. Note that in the example shown below, we use the default set up for semiparametric model fit on the Adhesive Bond B data.

```
> summary(addt.fit.semi)

Semi-Parametric Approach:

Parameters Estimates:
betahat
1.329

TI estimates:
TI.semi    beta0    beta1
26.313  -17.363 6697.074

Model Evaluations:
Loglikelihood      AICC
-288.135   586.269

B-spline:
Left Boundary   knots   Right Boundary
0.00   180.66         2016.00
```

We can also call *plot.addt.fit* to present model fitting results.

```
plot(addt.fit.semi, type="SEMI")
```

Figure 5.4 shows the plot of the original dataset of Adhesive Bond B data as well as the fitted degradation mean values using the semiparametric model. Here we assume that there is no correlation ρ between two error terms. Note that for *plot.addt* function, *type* argument should be compatible with the *addt.obj*, meaning that type used in plot function should be the same with *proc* argument in the function *addt.fit*, otherwise error messages will be generated.

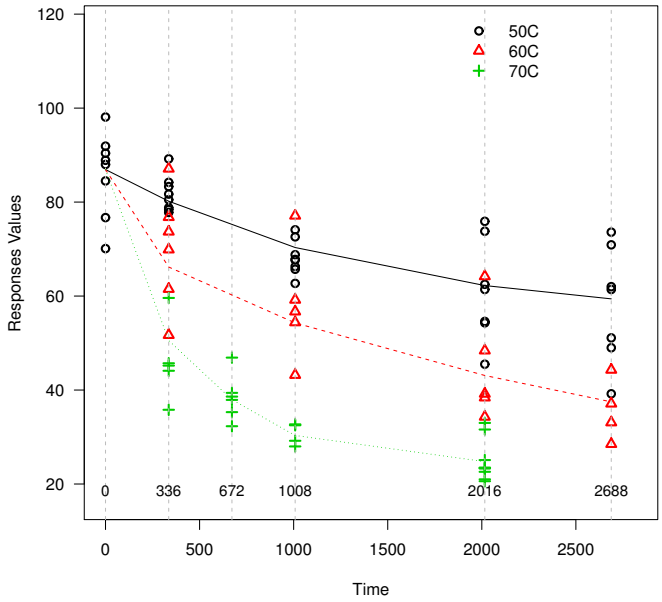


Figure 5.4: Plot of the original dataset of Adhesive Bond B data as well as the fitted degradation mean values using the semiparametric model.

We illustrate the comparisons among least-squares, maximum likelihood and semiparametric methods in terms of TI estimation in Figure 5.5. Temperature-time relationship lines are plotted for all three methods in black, red and blue lines correspondingly.

5.3 Data Analysis

In this section, we present a complete ADDT data analysis using the Seal Strength data to illustrate the use of functions in Section 5.2. The details of the Seal Strength data is available in Li and Doganaksoy (2014). The first ten observations are listed below. Note that in the Seal Strength data, temperatures at time point 0 are modified to 200 degrees while those in the original Seal Strength dataset in Table 5.2 are 100 degrees. Changing temperatures at time point 0 to the lowest temperature is a model computing risk and will not affect results, as at fitting time 0, temperature effect has not kicked in from the model.

```
>head(SealStrength, n=10)
```

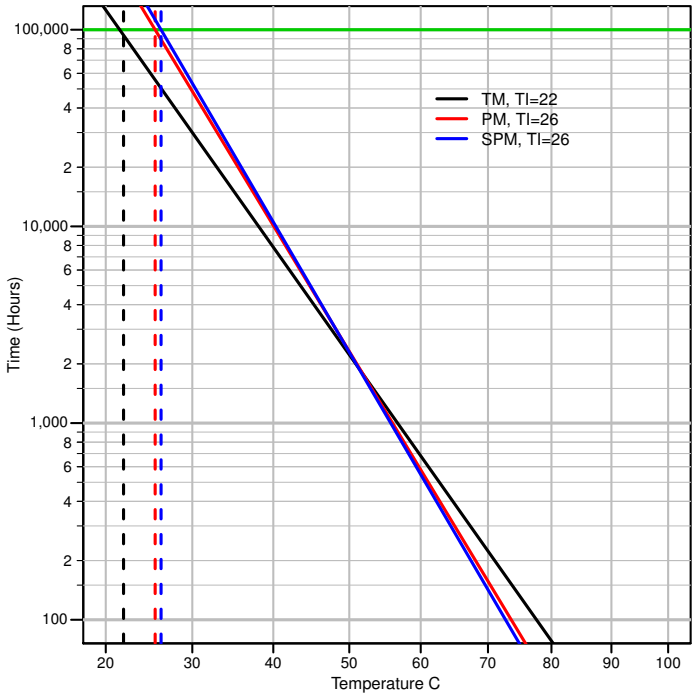


Figure 5.5: Temperature-time relationship lines for Adhesive Bond B data from least-squares (LS), maximum likelihood (ML) and Semi-parametric model (SemiPara) method respectively with a failure threshold of 70%.

	TempC	TimeH	Response
1	200	0	28.74
2	200	0	25.59
3	200	0	22.72
4	200	0	22.44
5	200	0	29.48
6	200	0	23.85
7	200	0	20.24
8	200	0	22.33
9	200	0	21.70
10	200	0	27.97

A graphical representation of the data is useful for users to obtain a general idea of the degradation paths. Using the *addt.fit.mla* object from *addt.fit* with *proc="ML"*, one can plot

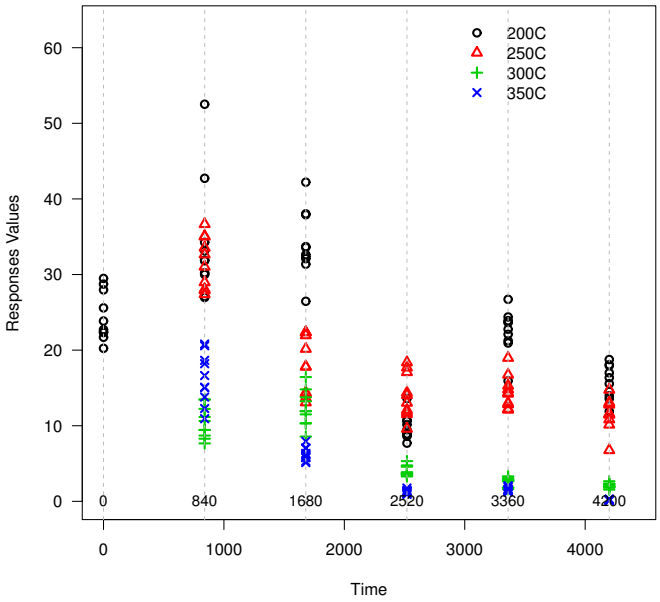


Figure 5.6: Plot of the Seal Strength data. Degradations were measured at six different time points under three different temperatures.

with *type="data"*.

```
>plot(addt.fit.mla, type="data")
```

Figure 5.6 shows the plot of the Seal Strength data, in which the degradations were measured at six different time points under three different temperatures. For Seal Strength data, we observe an average decrease in degradation measurements as time increases. Degradation measurements decrease with the accelerating variable, temperature as well.

Three different *addt.fit* models can be fitted, which are *proc ="LS"*, *proc = "ML"*, and *proc= "SemiPara"*, respectively.

```
>addt.fit.lsa<-addt.fit(Response~TimeH+TempC,data=Seal
Strength,proc="LS",failure.threshold=70)

>addt.fit.mla<-addt.fit(Response~TimeH+TempC,data=Seal
Strength,proc="ML",failure.threshold=70)
```

```
>addt.fit.semi<-addt.fit(Response~TimeH+TempC,data=Seal
Strength,proc="SemiPara",failure.threshold=70)
```

Alteratively, users can specify all three methods via one call of *addt.fit* by setting *proc = "All"*. All three methods object will be stored in the *addt.fit.all*.

```
> addt.fit.all<-addt.fit(Response~TimeH+TempC,data=Seal
Strength,proc="All",=failure.threshold=70)
```

To view the results of all three models, users can call *summary* function:

```
> summary(addt.fit.all)
Least Squares Approach:
beta0     beta1
0.1934 1565.1731
est.TI: 52
Interpolation time:
Temp      Time
[1,]  200 2862.3430
[2,]  250 2282.3303
[3,]  300  509.2084
[4,]  350  622.0857

Maximum Likelihood Approach:
Call:
lifetime.mle(dat = dat0, minusloglik = minus.
loglik.kinetics, starts = starts, method =
method, control = list(maxit = 1e+05))

Parameters:
```

```
mean      std 95% Lower 95% Upper
alpha   30.5898   3.4550   24.5152   38.1697
beta0    0.2991   1.7013   -3.0355    3.6337
beta1 3867.7170 899.5312 2104.6360 5630.7981
gamma    1.6556   0.4171    1.0105    2.7127
sigma    5.5456   0.6521    4.4041    6.9831
rho      0.7306   0.0664    0.6004    0.8607

Temperature-Time Relatonship:
beta0     beta1
-0.0942 1680.4055

TI:
est       std 95% Lower 95% Upper
56.6920   28.1598    1.4997  111.8842

Loglikelihood:
[1] -555.0169

Semi-Parametric Approach:

Parameters Estimates:
betahat
0.282

TI estimates:
TI.semi    beta0    beta1
32.768    0.362 1418.833
```

```
Model Evaluations:
Loglikelihood          AICC
-639.206      1288.412

B-spline:
Left Boundary  knots  knots  knots   knots
0.00 268.60 527.17 840.00 1394.55
Right Boundary
4200.00
```

Results shown here are the same when users call *summary* for three different models separately. The *add.fit.all* and *summary* for *addt.fit.all* provides an alternative way to analyze the data simultaneously.

Similar to Section 5.2, We illustrate the results from least-squares (LS), maximum likelihood (ML) and the semiparametric (SemiPara) method in Figures 5.7, 5.8, 5.9 and 5.10, respectively. Note that in Figures 5.9 and 5.10, we illustrate semiparametric (SemiPara) method results for without ρ and with ρ cases.

In addition, users can specify the *semi.control* argument in the *SemiPara* fit option. The *semi.control* contains a list of arguments that regards the *SemiPara* option in the model. For example, whether or not to include a correlation ρ in the model. When *semi.control = list(cor = T)*, the model will fit the correlation model with ρ. Otherwise, when default value *semi.control = list(cor = F)* or *semi.control* is not specified, no correlation will be fitted. Note that for option *SemiPara* in the function *addt.fit*, including the correlation ρ in the model may require more computing time, but potentially it will provide a better fit.

Here we compare the model results from the traditional method, the parametric method, and the semiparametric method for the Seal Strength data. In the results from *summary*, TI estimates are 52°C, 56°C and 47°C, respectively. With β_0 and β_1 estimates, the TI plot is presented in Fig 5.11. The black line is the TI from the traditional model, the red line is the

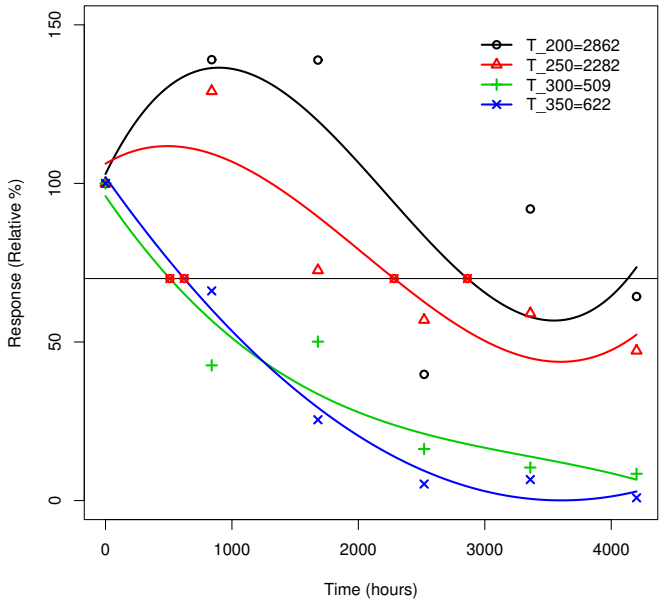


Figure 5.7: Plot of the Seal Strength parametric lines with least-square (LS) method. The red, green, blue, light blue lines represent 200, 250, 300 and 350 degrees Celsius interpolated curves, respectively.

parametric model TI estimates, and the blue line stands for the results from semiparametric method.

In the results from two methods, without and with the correlation ρ, $\hat{\beta}$ are 0.282 and 0.323, while TI estimates are 32.768 and 47.338, respectively. The differences come from the assumption of ρ in the model. From the AICc value, the model with correlation provides a better fit to the data becasue it provides a smaller AICc value. The details of the model outputs are shown as follows.

```
>addt.fit.semi.no.cor<-addt.fit(Response~TimeH
+TempC,data=SealStrength,proc="SemiPara",
failure.threshold=70)

>addt.fit.semi.cor<-addt.fit(Response~TimeH
+TempC,data=SealStrength,proc="SemiPara",
```

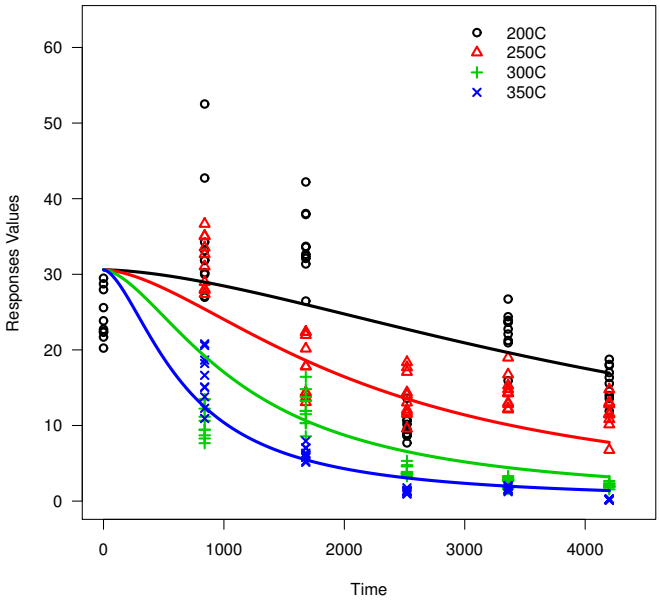


Figure 5.8: Plot of the fitted mean function using maximum likelihood method for the Seal Strength data. The 200, 250, 300 and 350 degrees Celsius estimated curves are represented by red, green, blue and light blue lines, respectively.

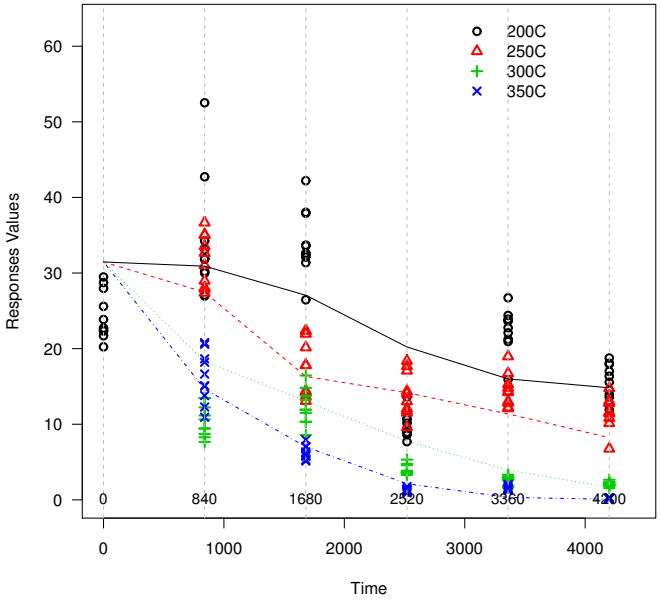


Figure 5.9: Plots of fitted lines using the semiparametric (SemiPara) method for the Seal Strength data, for model without ρ.

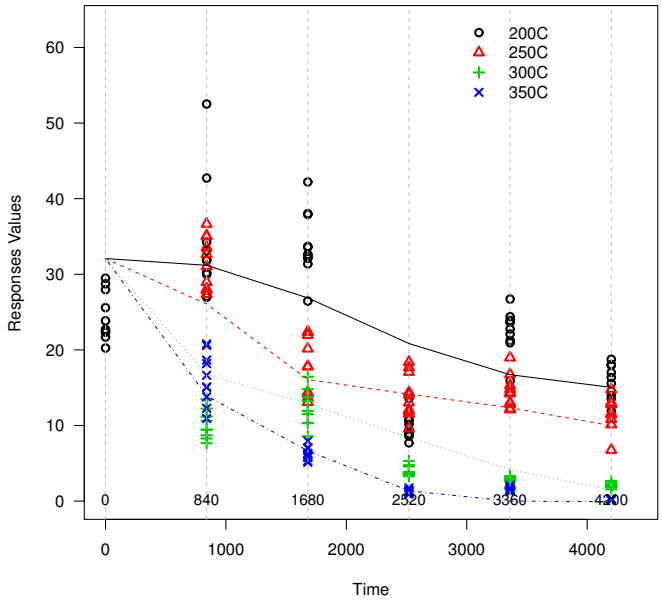


Figure 5.10: Plots of fitted lines using the semiparametric (SemiPara) method for the Seal Strength data, for model with ρ.

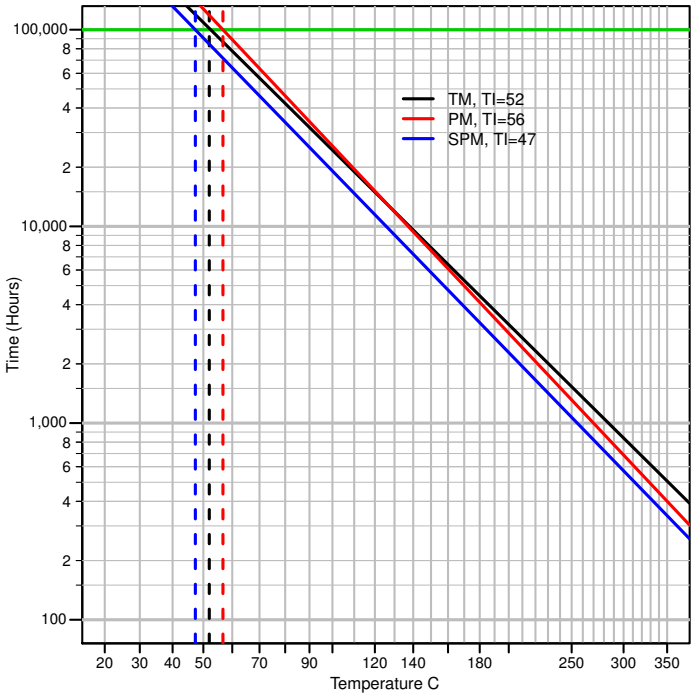


Figure 5.11: Temperature-time relationship lines for Seal Strength data using the traditional method, maximum likelihood method and semiparametric method, respectively, with a failure threshold of 70%.

```
failure.threshold=70, semi.control = list(cor=T))
```

- Model without correlation ρ:

```
> summary(addt.fit.semi.no.cor)
Semi-Parametric Approach:

Parameters Estimates:
betahat
0.282

TI estimates:
TI.semi    beta0    beta1
32.768    0.362 1418.833

Model Evaluations:
Loglikelihood     AICC
-639.206 1288.412

B-spline:
Left Boundary    knots     knots     knots    knots
0.00   268.60    527.17    840.00  1394.55
Right Boundary
4200.00
```

- Model with correlation ρ:

```
> summary(addt.fit.semi.cor)

Semi-Parametric Approach:
```

```
Parameters Estimates:
betahat     rho
0.323   0.714

TI estimates:
TI.semi    beta0    beta1
47.338   -0.087 1630.282

Model Evaluations:
Loglikelihood     AICC
-552.662  1117.323

B-spline:
Left Boundary    knots     knots     knots     knots
0.00   265.59    520.02    840.00   2483.29
Right Boundary
4200.00
```

5.4 Concluding Remarks

In this chapter, we provide a comprehensive description with illustrations for the ADDT methods implemented in the `ADDT` package. Functions such as the *addt.fit* and *summary* are illustrated for the traditional method, the parametric method, and the semiparametric method. We also show R examples using the Adhesive Bond B data and the Seal Strength data under various function options like *proc* and *semi.control*. Results from three different models are provided and visualized. Users can consult the reference manual in Hong et al. (2016) for further details regarding the software package.

Bibliography

L. A. Escobar, W. Q. Meeker, D. L. Kugler, and L. L. Kramer. Accelerated destructive degradation tests: Data, models, and analysis. In B. H. Lindqvist and K. A. Doksum, editors, *Mathematical and Statistical Methods in Reliability*, chapter 21. World Scientific Publishing Company, River Edge, NJ, 2003.

Y. Hong, Y. Xie, Z. Jin, and C. King. *ADDT: A Package for Analysis of Accelerated Destructive Degradation Test Data*, 2016. URL `http://CRAN.R-project.org/package=ADDT`. R package version 1.1.

C. B. King, Y. Xie, Y. Hong, J. H. Van Mullekom, S. P. DeHart, and P. A. DeFeo. A comparison of traditional and maximum likelihood approaches to estimating thermal indices for polymeric materials. *Journal of Quality Technology*, 50:117–129, 2018.

M. Li and N. Doganaksoy. Batch variability in accelerated-degradation testing. *Journal of Quality Technology*, 46:171–180, 2014.

C.-C. Tsai, S.-T. Tseng, N. Balakrishnan, and C.-T. Lin. Optimal design for accelerated destructive degradation tests. *Quality Technology and Quantitative Management*, 10:263–276, 2013.

UL746B. *Polymeric Materials - Long Term Property Evaluations, UL 746B*. Underwriters Laboratories, Incorporated, 2013.

I. Vaca-Trigo and W. Q. Meeker. A statistical model for linking field and laboratory exposure results for a model coating. In J. Martin, R. A. Ryntz, J. Chin, and R. A. Dickie, editors, *Service Life Prediction of Polymeric Materials*, chapter 2. Springer, NY: New York, 2009.

Y. Xie, Z. Jin, Y. Hong, and J. H. Van Mullekom. Statistical methods for thermal index estimation based on accelerated destructive degradation test data. In D. G. Chen, Y. L. Lio,

H. K. T. Ng, and T. R. Tsai, editors, *Statistical Modeling for Degradation Data*, chapter 12. Springer, NY: New York, 2017.

Y. Xie, C. B. King, Y. Hong, and Q. Yang. Semiparametric models for accelerated destructive degradation test data analysis. *Technometrics*, 60:222–234, 2018.

Chapter 6 General Conclusions and Areas for Future Work

6.1 Conclusions

In this dissertation, we study a clustering problem in multi-dimensional sensory data, propose next event time prediction techniques in recurrent event process system, and compare methods of thermal index estimation for materials. All three projects are related to applications in reliability analysis. Specifically, in the first project, clustered events reveals natural grouping mechanisms for events defined from multi-dimensional sensory data. This can be used as a guideline of irregular event screenings in reliability, when the data is from multi-dimensional sensory data. In the second project, we predict the next system event time, which is equivalent with the system remaining life time in reliability analysis. In the third project, we provide a comprehensive comparison on estimation methods of an important measure in reliability, i.e., thermal index. There are different contributions for each one of our projects.

In the first project, we propose an innovative way to conduct clustering for events defined by multi-dimensional sensory data. Three penalty terms on the likelihood work differently on removing variables from the transformed coefficient matrix. We also achieve automatic variable selection in the clustering process.

For the second project, we compare covariate adjusted recurrent process (CARP) model with different assumptions, i.e., MLN and copula. CARP model performances are sensitive to the true underlying model, while covariate adjustment should always be recommended, as shown in simulation study. Both CARP models are applied to the Yellowstone National Park geyser data.

In the third project, comprehensive comparisons on methods of thermal index estimation are provided. Advantages and disadvantages are discussed in Chapter 4, while an R package is introduced to implement different methods.

6.2 Future Directions

In simulation study of Chapter 2, compared to the linear B-spline we currently use, we will generate data with higher B-splines orders for models under three different penalty terms. Because higher order B-splines can be used to characterize more complicated sensory measurements. In addition, a penalized FPCA will be implemented in order to avoid potential identifiability issues on tuning parameter selections. Variable selection results will be consulted by specialists before putting into applications. Variables with meaningful interpretations in reality need to be considered carefully regardless of our method selection results. In the engineering sensory data example, sensor costs will be employed as weights when conducting sensor selection. Specifically, expensive sensors will obtain higher weights so that they are relatively easier to be removed, while cheap sensors tend to stay with similar information presented. In Chapter 3, we will study the accuracy of next event time and label prediction in simulation study. In the geyser data application, additional covariates such as rainfall and temperatures will be considered since they bring additional external information to the geyser eruption times. In addition, we will consider a multivariate recurrent process as an extension to our proposed bivariate model, it is useful when there are more event types in the system.