DISSERTATION

MODULATED RENEWAL PROCESS MODELS WITH FUNCTIONAL PREDICTORS FOR NEURAL CONNECTIVITIES

Submitted by

Hongyu Tan

Department of Statistics

In partial fulfillment of the requirements

For the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Fall 2015

Doctoral Committee:

Advisor: Phillip L. Chapman
Co-Advisor: Haonan Wang

Mary C. Meyer
J. Rockey Luo

ABSTRACT

MODULATED RENEWAL PROCESS MODELS WITH FUNCTIONAL PREDICTORS FOR NEURAL CONNECTIVITIES

Recurrent event data arise in fields such as medicine, business and social sciences. In general, there are two types of recurrent event data. One is from a relatively large number of independent processes exhibiting a relatively small number of recurrent events, and the other is from a relatively small number of processes generating a large number of events. We focus on the second type. Our motivating application is a collection of neuron spike trains from a rat brain, recorded during performance of a task. The goal is to model the intensity of events in the response spike train as a function of a set of predictor spike trains and the spike history of the response itself. We propose a multiplicative modulated renewal processes model that is similar to a Cox proportional hazards model. The model for response intensity includes four components: (1) a baseline intensity, or hazard, function that captures the common pattern of time to next event, (2) a log-linear term that quantifies the impact of the predictor spike histories through coefficient functions, (3) a similar log-linear term for the response history, (4) a log-linear regression-type term for external time dependent variables. The coefficient functions for predictor and response histories are approximated by B-spline basis functions. Model parameters are estimated by partial likelihood. Performance of the proposed methods is demonstrated through simulation. Simulations show that both the coefficient function estimates and the asymptotic standard error function estimates are accurate when the sample size is large. For small samples, simulations show that the smoothly absolute clipped deviation (SCAD) penalty outperforms LASSO penalty and unpenalized partial likelihood approach in identifying functional sparsity under various situations. The proposed methods are illustrated on a real spike train data set, in which substantial non-stationarity is identified.

ACKNOWLEDGEMENTS

I would like to express my deepest appreciation to my committee especially my co-advisors Phillip Chapman and Haonan Wang, who have been inspiring, encouraging, and thoughtful throughout my Ph.D. study. I would also like to thank my fellow graduate students, the faculty, and the staff in the Department of Statistics at Colorado State University for their help.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

Introduction

1.1 Background and motivation

1.1.1 Analysis of recurrent events

Recurrent event data has been widely collected in fields such as biomedicine, business, reliability engineering and the social sciences. The recurrent event data are provided by recurrent event processes which generate events repeatedly over time. In general, there are two basic types of data [Cook and Lawless, 2007]. One is from a relatively large number of processes, usually assumed to be independent, exhibiting a relatively small number of recurrent events. Such data arise frequently in medical studies, where many individuals may experience transient clinical events repeatedly over a period of observation. The literature on the statistical analysis of such data has grown rapidly over the years and a variety of models and methods has been developed, for example, Poisson models based on counts and rate functions, Cox proportional hazards models based on gap times between events. The other type of recurrent events is from a relatively small number of processes, each generating a large number of events. In some cases, there is only one process that generates tens of thousands of events over the observational period. The difference between modeling procedures of both types of data is that the second type is much more challenging due to the possible dependence of an event within a process on previous events as well as other processes. A few models and methods for data from such sets of processes are available, such as the modulated renewal process model, study of the properties and performance of these models are very limited.

In this study, we focus on the second data structure, in particular, a situation in which several processes generate long parallel sequences of events. This study is motived by the need for models that can capture the relationships between simultaneously recorded neuron spike trains, which are commonly viewed as point processes. A neuron spike train is a sequence of event times, and in this case, an event is a “spike" also termed action potential or neural firing. The available data usually consist of neural spike trains that record thousands of spikes from a single subject over a period of observation. A review of neural spike train analysis is given in Section 1.1.2 and an example of neural spike train data is given in Section 1.2.

1.1.2 Analysis of neural spike trains

Multiple electrodes are used to simultaneously record electrical activities of several neurons in a single brain region or across different regions. The electrical activity of a neuron is recorded as a neural spike train. Simultaneously recorded spike trains are used to study how groups of neurons process information and how they interact with each other. Developing statistical methods for the analysis of multiple neural spike-train data is an important and challenging problem [Brown et al., 2004, Reed and Kaas, 2010].

This work involves simultaneously recorded spike trains in the hippocampus. The hippocampus is a brain region that has long been known to play an important role in spatial navigation and the consolidation of information from short-term memory to long-term memory. Damage to the hippocampus by diseases like Alzheimer's disease can cause memory problems and disorientation. People with extensive, bilateral hippocampal damage may experience anterograde amnesia, the inability to form or retain new memories. Developing a neural prosthesis for the damaged hippocampus would be very helpful for people suffering problems caused by hippocampal damage. One of the fundamental principles of cortical brain regions, including the hippocampus, is that information is represented in the ensemble action potentials (nerve impulses) of populations of neurons [Song et al., 2007]. Thus, the first step of developing a neural prosthesis for the damaged hippocampus is to understand the connectivities of the neuron firings in the hippocampus. Analysis of multiple spike trains recorded from neurons in hippocampus allows us to understand such connectivities.

Several methods have been developed for analyzing the relationships between spike trains. The first group of methods is focused on pairwise relationship, such as the cross-correlation function (CCF) [Perkel et al., 1967], the cross-intensity function [Brillinger, 1976a, 1992, Cox, 1972a], product densities, cumulant densities, cumulant spectra, method of moments [Bartlett, 1966, Brillinger, 1975a,b], calculation of the coherence [Brillinger, 1976b, 1992], and the joint peristimulus time histogram (JPSTH) [Aertsen et al., 1989, Gerstein and Perkel, 1969, 1972]. One drawback of these methods is that they fail to consider possible influences from other simultaneously recorded spike trains. The second group of methods is based on multiple regression. In particular, one spike train is chosen to be the response train and the other spike trains are predictor trains. Multiple influences from the predictor spike trains on the response train can be quantified by estimating the regression parameters.

Various regression models have been applied to the study of neuronal connectivities. In general, these models belong to two classes. The first class is the generalized linear models(GLM). This approach assumes the response train is an inhomogeneous Poisson process and the spike intensity is proportional to the exponential of the linear combination of the predictors [Brillinger, 1988, Okatan et al., 2000, Truccolo et al., 2005]. A detailed discussion on this approach can be found in Masud and Borisyuk [2011]. The second class is based on the Cox proportional hazards model [Cox, 1972a]. This approach assumes that hazard function of the time to the next spike of the response train is the product of a baseline hazard function and the exponential of the linear combination of the predictors. The baseline hazard function can be left unspecified and the model can be estimated by maximizing the Cox partial likelihood. It was originally applied to neuroscience data by Borisyuk et al. [1985], and its application to simultaneously recorded spike train data was further studied by Masud and Borisyuk [2011].

To develop useful models for relationships between neural spike trains, e.g., connectivities, several essential characteristics of neurons and their activities must be taken into account. First, the response is a single long sequence of spikes from the same subject. Information is carried in the times of spikes. It is not reasonable to ignore the correlations among the response spike times. Second, neural mechanisms, such as refractory and recovery period, determine whether a spike of a neuron has an impact on the neuron's future activities. Third, the activities of the neurons depend on the subject's activity at time of recording. Since a certain behavior of the subject usually extends over a time period in which a large number of spikes occur, these spike times may have strong correlations. Based on those three characteristics, the model formulation should reflect the dependence of the response neuron's activities on its own spike history.

The relationship between the response spike train and the predictor spike trains can be very complicated due to the following reasons. The activity of the response neuron depends not only on the current activities of other neurons but also on their histories. This requires the model to be dynamic and the connectivities to be functional. Moreover, the connectivity between two neurons can be direct or indirect (Figure 1.1). It is not sensible to assume that different types of connections are the same. We would also like to point out that there is an under-sampling problem, which is due to the fact that the observed spike trains usually represent only a small portion of neuron population. Therefore, the functional connectivity between neurons cannot be directly interpreted as synaptic connections. This implies that the functional connectivities can be very complex due to the impact of the unobserved neurons.

1.1.3 Our approaches

In this dissertation, we model the response spike train as a modulated renewal process [MRP; Cox, 1972b], an extension of the Cox proportional hazards model that can be used to model recurrent events whose occurrence may depend on the previous events. In Cox's MRP model, the conditional intensity function, which describes the rate of response events, given the entire history of the process, is assumed to be the product of a baseline hazard function and an exponential of a linear combination of predictors. These predictors may include not only the previous spike activity of the other trains but also the response spike train's own spike history, as well as other covariates. Under the Cox MRP model, it is assumed that the dependence of the conditional intensity function of the event on the history of the process

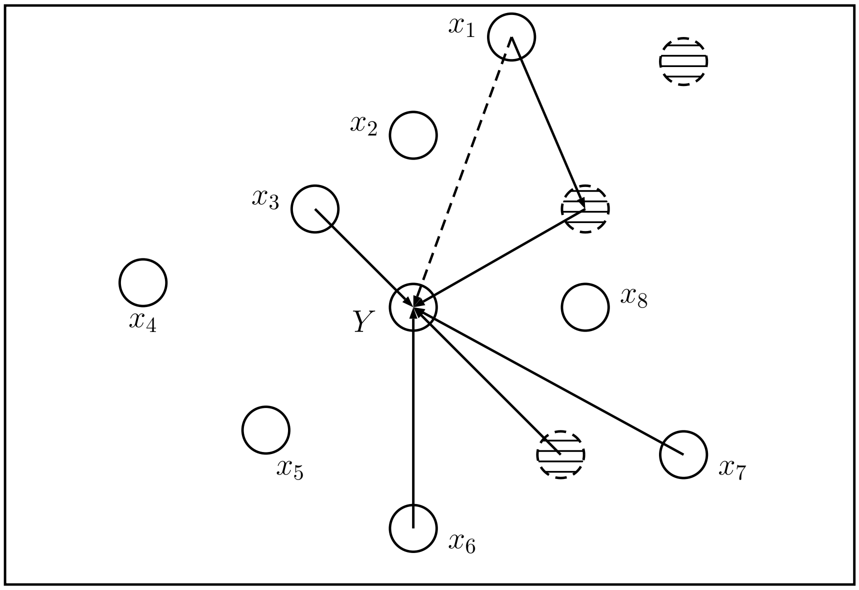

Figure 1.1: Possible connections between neurons: Y is the response neuron, $x_1, x_2, ..., x_8$ are observed predictor neurons. A direct connection is represented by a solid straight line and a indirect connection is represented by a dashed straight line. A shaded circle represents a neuron that is not observed.

is adequately captured by the linear function of the predictors in the model. Like the Cox proportional hazards model, the baseline hazard function can nonparametrically describe complex firing pattern of the response neuron.

Lin et al. [2013] argued that, in practice, the dependence of the current event on the process history may not be adequately captured by a limited number of predictors, so the assumption of the MRP model may be too strong. Those authors used a rate function, defined similarly to the conditional intensity function, which describes the rate of response event occurrences, given the predictors in the model. They did not assume that the predictors in the model can fully capture the dependence of the current event on the process history. The partial likelihood-based inference is much more challenging under the rate model than under the Cox MRP model. The variance of the estimator proposed in Lin et al. [2013] is quite complicated, owing to the unknown dependence structure on the process history that is not explained by the linear function of the predictors. To estimate that variance they adapted block bootstrapping and cluster variance estimators to the partial likelihood.

We believe that the Cox MRP model is appropriate if the predictors can successfully capture a large portion of the information in the process history. The challenge in spike train modeling is that the conditional intensity of a response spike at a given time may depend on the spike history in various ways. Most previous studies, including Lin et al. [2013], divided the history into subintervals and treat each subinterval as a predictor. Then, how well the models capture the dependence of the current spike probabilities on previous spikes depends on the size and number of the subintervals. Ideally, the subintervals can be small enough to approximate the true form of the dependence; however, if the subintervals are too small, the model contains too many parameters (hundreds) to adequately estimate. Thus, important information in the spike history may be lost and the models may not capture the dependence of the current spike on previous spikes.

In this work, we take an approach that does not require the specification of subintervals. In our proposed model, the impact strength of the history of each spike train is modeled by a coefficient function that is approximated by a linear combination of B-splines. Taking this approach, we can keep the number of parameters in the model small without losing much information in the spike history. The flexibility of the coefficient functions provides improved ability to capture the dependence of the current spike intensity on previous spike density. Therefore, the assumptions of the Cox's MRP model should be closer to valid and we can take the advantage of the better studied Cox's MRP model. Our model also allows extrinsic predictors, such as stimulus and behavioral variables, to affect the response spike train. By including those predictors, the functional connections between neural spikes can be better estimated and more interpretable. We take the penalized maximum partial likelihood approach to address functional estimation and sparsity. Functional sparsity refers to coefficient functions that are zero over many sub-periods or the entire period of history. We also adapt the Kolmogorov-Smirnov test based on the time-rescaling theorem [Brown et al., 2002, Haslinger et al., 2010] as a way to evaluate the goodness-of-fit and prediction.

1.2 The neural spike train data

The data used in this dissertation were collected from hippocampus of Long-Evans rats. Neural signals were recorded with an 8-shank silicon probe in layer CA1 of the right dorsal hippocampus while the rats were chasing randomly placed drops of water on a elevated square platform [Mizuseki et al., 2009a,b]. The raw data were preprocessed and transformed into spike trains. Each spike train records the activity of a neuron. It is believed that information is carried in the timings but not the waveform of the spikes, so the durations of the spikes are ignored [Song et al., 2013]. A spike train can be characterized simply by a series of all-or-none point events in time [Gerstner and Kistler, 2002]. In addition to the neural activities, the positions of the rat were also identified by two LEDs on the rat's head. Positions of the rat were extracted from a video file that recorded the experiment. A more detailed description of the data collection can be found in Mizuseki et al. [2009b].

The computational examples in this dissertation are all based on six spike trains from one rat over 4500 seconds. Here we present 500 second data as an example. Denote the six spike trains as S_1, S_2, S_3, S_4, S_5, and S_6. Spikes within 0.002s of the previous spike are believed to be false signals created by the equipment, so they are removed. During this 500 second period, the 6 neurons generated 15893, 13594, 8449, 18212, 1209, and 20275 spikes, respectively. Figure 1.2 shows a portion of the data set recorded over 20 seconds. It can be seen that the overall spike frequencies are different across the six trains; in particular, S_4 has much lower frequency than the other five. This is further seen in the means of the inter-spike times, i.e., the time between two successive spikes, and the number of spikes over 500 seconds (Table 1.1). However, the first quantiles and the medians of the inter-spike time for the six trains are not very different (Table 1.1). Figure 1.3 shows the histograms of the inter-spike times for the inter-spike times that are not greater than 0.2 seconds. The inter-spike time distributions are all unimodal and right skewed. The modes of the density functions are all close to 0, but the density values at the modes are different. From Table 1.1, the 98th percentiles of S_1, S_2, S_3, S_5 are less than 0.2 seconds, so the histograms show more than 98%

of the data for these four spike trains. The distributions of S_4 and S_6 have heavier tails with about 70% and 95% of inter-spike times less than 0.2 seconds respectively. Although S_4 has much lower overall spike frequency; its inter-spike time density still has very small mode and its density at the mode is as large as S_5, which has the highest overall spike frequency. All these results indicate that the spikes tend to cluster together in time. Clustering could be caused by spike rate change related to other neurons and extrinsic variables, such as stimulus and subject's behavior. It could also be caused by the correlation between the spikes of the same neuron.

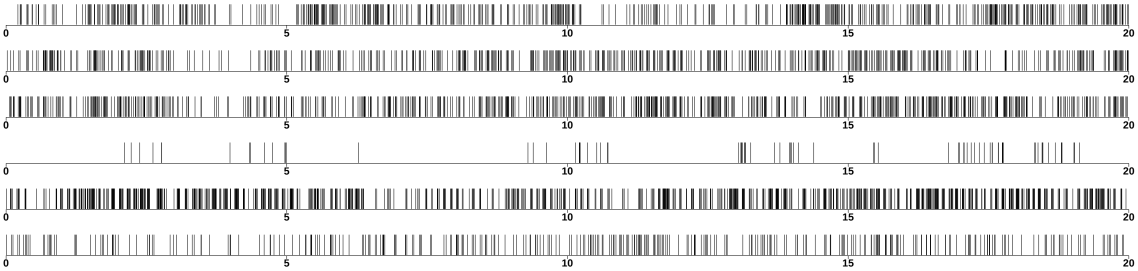

Figure 1.2: Spikes of six neurons (rows) over 20 seconds.

Table 1.1: Summary statistics for the inter-spike times: percentiles, means and number of spikes (n).

	S_1	S_2	S_3	S_4	S_5	S_6
0%	0.0020	0.0020	0.0020	0.0020	0.0020	0.0020
25%	0.0100	0.0101	0.0082	0.0136	0.0054	0.0150
50%	0.0184	0.0198	0.0152	0.0635	0.0095	0.0348
75%	0.0371	0.0463	0.0330	0.2357	0.0235	0.0784
90%	0.0720	0.0912	0.0663	0.9355	0.0773	0.1316
93%	0.0873	0.1033	0.0795	1.4367	0.0894	0.1543
95%	0.1010	0.1149	0.0913	1.9397	0.0981	0.1821
97%	0.1226	0.1348	0.1083	3.2395	0.1105	0.2359
98%	0.1416	0.1600	0.1229	3.8878	0.1200	0.2733
99%	0.1845	0.2224	0.1497	6.1288	0.1457	0.3658
100%	1.1928	0.6983	0.7426	15.9365	0.9157	0.8639
mean	0.0315	0.0368	0.0275	0.4118	0.0247	0.0592
n	15893	13594	18212	1209	20275	8449

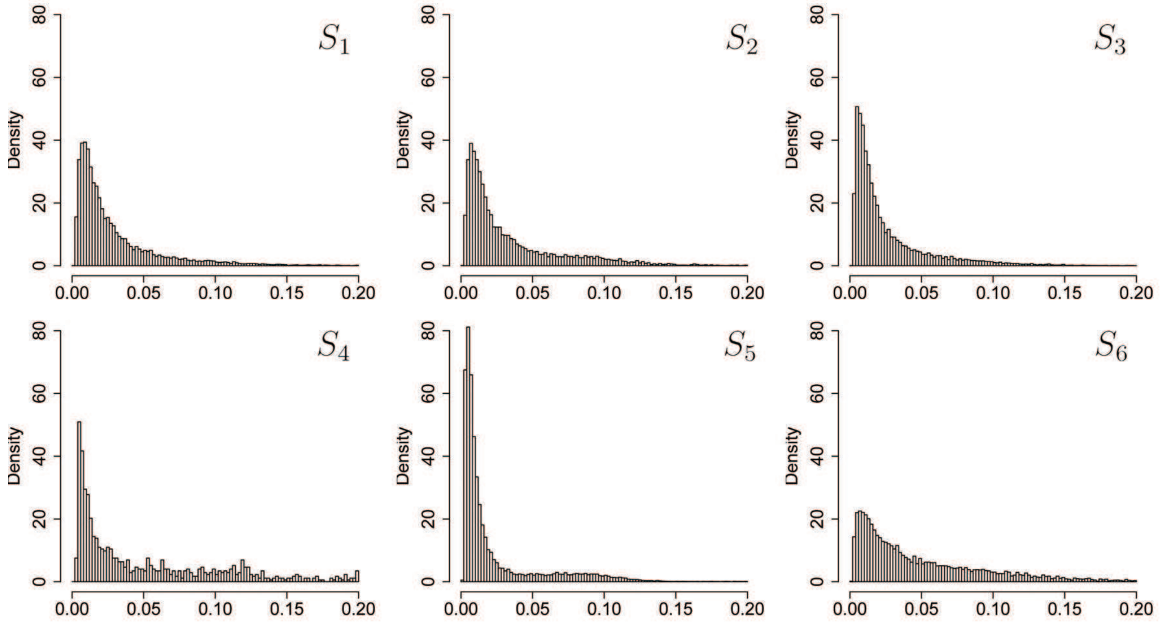

Figure 1.3: Histograms of the inter-spike times for spike trains $S_1, S_2, ..., S_6$, the $\leq$ 0.2s portions.

1.3 Review of renewal process

In this section, we review analysis of recurrent event data based on gap time, i.e., the time between two successive events, and we introduce notation and the framework based on stochastic process. Approaches to modeling of recurrent events are usually described in terms of stochastic processes. The concepts of intensity functions and counting processes are especially useful [Cook and Lawless, 2007]. For a single recurrent event process staring at $T_0 = 0$, let $0 < T_1 < T_2 < ...$ denote the event times, where T_k is the time of the kth event. For any time $t \geq 0$, let $N(t)$ denote the number of events that occur in $(0, t]$. Note that $N(t)$ is a counting process, which is a right continuous function, i.e., $N(t) = N(t+)$, where $N(t+) = \lim_{\Delta \to 0} N(t + \Delta)$ for $\Delta > 0$. $N(t)$ jumps 1 at each event time and is constant otherwise [Daley and Vere-Jones, 1988, Snyder and Miller, 1991]. More generally, $N(s, t) = N(t) - N(s)$ is the number of events occurring over the interval $(s, t]$. The history of the process at t, $H_t = \{N(u), u \in (0, t)\}$ contains all information about the sequence of event times before t. For events occurring in continuous time, one can make the mathematically convenient assumption that two events can not occur simultaneously [Cook and Lawless, 2007]. Models for recurrent events can be specified very generally by considering the *conditional intensity function*, which is the probability distribution for the number of events in short intervals $[t, t + \Delta), \Delta > 0$, given the history of event occurrence before time t. The *conditional intensity function* is defined formally as

$$\lambda(t|H_t) = \lim_{\Delta t \to 0} \frac{Pr\left(N((t + \Delta t)-) - N(t-) = 1|H_t\right)}{\Delta t}, \tag{1.1}$$

where $N(t-)$ is defined as $\lim_{\Delta \to 0} N(t - \Delta)$ for $\Delta > 0$.

Let $Y_j = T_j - T_{j-1}$ denote the gap time between the $j-1$st and jth events. Analyses based on gap times are usually useful in certain settings. An important one is where an individual or system is restored to a similar physical state after each event [Cook and Lawless, 2007]. Renewal process models are useful in such settings. A renewal process is a process whose conditional intensity at t only depends on the time since the most recent event before t, i.e.,

$$\lambda(t|H_t) = h(t - T_{N(t-)}), \tag{1.2}$$

where $t - T_{N(t-)}$ is the time since the most recent event before t, or the backward occurrence time, and $h(y)$ is the hazard function for the gap times Y_j, $j = 1, 2, ..., J$. For renewal processes, the gap times Y_j are independent and identically distributed; that is, Y_j have common density function $f(y)$. Here are some useful functions for describing a renewal process and their connections. A survival function is defined as $S(y) = P(Y \geq y)$, and

$$h(y) = \frac{f(y)}{S(y)} = \lim_{\Delta \to 0} \frac{Pr\left(Y < y + \Delta|Y \geq y\right)}{\Delta}. \tag{1.3}$$

Note that $S(y) = 1 - F(y)$, where $F(y)$ is the cumulative distribution function, then $h(y) = -d(log(S(y)))/dy$ and $S(y) = \exp\{-\int_0^y h(u)du\}$. $CH(y) = \int_0^y h(u)du$ is called the cumulative hazard function and is usually denoted by H; however, here we use CH because we already use H to denote the event history. We assume that the time origin $t = 0$ corresponds to an event time. This may be relaxed so Y_1 is allowed to have a different distribution from $Y_2, Y_3, ...$, with the gap times still being mutually independent.

Covariates may be incorporated into renewal processes in straightforward ways. If fixed covariates $\boldsymbol{Z} = (Z_1, Z_2, .., Z_r)^T$ are associated with independent renewal processes, we can allow the common distribution of the gap times Y_j for a given process to depend on $\boldsymbol{Z}$. Because Y_j are time-to-event variables, regression models in survival analysis may be used. One very popular family of such models is the proportional hazards model, where the hazard function of Y_j is of the form

$$h(y) = h_0(y)\exp(\boldsymbol{Z}^T\boldsymbol{\beta}), \tag{1.4}$$

where $h_0(y)$ is a positive-valued function referred to as the baseline hazard function, and $\boldsymbol{\beta} = (\beta_1, \beta_2, ..., \beta_r)^T$ is the vector of coefficients. In a model like this, the Y_j for a given process are independent but are not identically distributed. Let $f_j(y)$, $F_j(y)$, $S_j(y)$ denote the the probability density function, cumulative distribution function and the survival function of Y_j respectively. Note that $S_j(y) = \exp\left(-\int_{t_{j-1}}^{t_{j-1}+y} \lambda(u|H_u)du\right)$. Assume J events are observed at times $0 < t_1 < t_2 < ... < t_J \leq T$. Because $Y_j = T_j - T_{j-1}$ are independent, the order of the events can be ignored, i.e., the J recurrent events can be viewed as J events from J independent processes generating only one event. The likelihood function is of the form

$$\begin{aligned} L &= \prod_{j=1}^{J} f(y_j)(1 - F_{J+1}(T - t_J)) && (1.5)\\ &= \prod_{j=1}^{J} \lambda(t_j|H_{t_j})S_j(t_j - t_{j-1})S_{J+1}(T - t_J) && (1.6)\\ &= \prod_{j=1}^{J} \lambda(t_j|H_{t_j})\exp\left(-\int_{t_{j-1}}^{t_j} \lambda(u|H_u)du\right)\exp\left(-\int_{t_J}^{T} \lambda(u|H_u)du\right). && (1.7) \end{aligned}$$

When the form of the baseline hazard function $h_0(\cdot)$ is known, $\lambda(t|H_t)$ is parametric and estimate of $\boldsymbol{\beta}$ can be obtained by maximizing the full likelihood function $L(\boldsymbol{\beta})$.

1.4 Review of Cox partial likelihood

When $h_0(\cdot)$ is left unspecified, the use of the likelihood function becomes problematic. Cox [1972a] proposed an approach for estimating the parameters of interest without specifying the form of the baseline hazard by maximizing a component of the likelihood, which is called the Cox partial likelihood, instead of the full likelihood. The Cox partial likelihood is motivated by the survival analysis example in which $y_1, y_2, ..., y_J$ are failure times of J independent experimental units observed in parallel. Conditional on the set of failure times, the probability that the failure observed at y_j is unit j is the hazard function at y_j, divided by the sum of the hazard functions of all units that had not yet failed at time y_j. The risk set is defined as $R(y_j) = \{i : y_i \geq y_j\}$ and the contribution of failure j to the partial likelihood is

$$\frac{h_0(y_j)\exp(\boldsymbol{Z}_j^T\boldsymbol{\beta})}{\sum_{k\in R(y_j)} h_0(y_j)\exp(\boldsymbol{Z}_k^T\boldsymbol{\beta})}. \tag{1.8}$$

For a renewal process, the events are not observed in parallel but in sequence (Figure 1.4). However, because the gap times are assumed to be independent, a sequence of J events can be treated the same way as J parallel events that are generated by J independent processes. Thus, the Cox partial likelihood for a renewal process model can be obtained as the product of the ratio of the hazard value at y_j of the process that is observed to generate an event at y_j and the sum of the hazard values at y_j of the processes in the risk set of y_j, that is

$$PL(\boldsymbol{\beta}) = \prod_{j=1}^{J} \frac{h_0(y_j)\exp(\boldsymbol{Z}_j^T\boldsymbol{\beta})}{\sum_{k\in R(y_j)} h_0(y_j)\exp(\boldsymbol{Z}_k^T\boldsymbol{\beta})} = \prod_{j=1}^{J} \frac{\exp(\boldsymbol{Z}_j^T\boldsymbol{\beta})}{\sum_{k\in R(y_j)} \exp(\boldsymbol{Z}_k^T\boldsymbol{\beta})}. \tag{1.9}$$

The risk set of y_j, denoted as $R(y_j)$, is defined as the set of the processes that do not generate a event before y_j. If we consider the real event time t instead of gap time y, the risk set $R(y_j)$ may contain events that have already occurred before the jth event. Thus, the interpretation of the risk set is not the same as the risk set for parallel events.

When estimate of $\boldsymbol{\beta}$ is obtained by maximizing the Cox partial likelihood function, the cumulative baseline hazard $CH_0(y) = \int_0^y h_0(u)du$ can be estimated by

$$\widehat{CH_0}(y) = \sum_{y_j \leq y} \frac{1}{\sum_{k\in R(y_j)} \exp(\boldsymbol{Z}_k^T\hat{\boldsymbol{\beta}})}. \tag{1.10}$$

Below are some useful results for obtaining the maximizer of the Cox partial likelihood [Cox, 1972a]. The log partial likelihood function is

$$l(\boldsymbol{\beta}) = \sum_{j=1}^{J} \left\{ \boldsymbol{Z}_j^T\boldsymbol{\beta} - \log\left(\sum_{k\in R(y_j)} \exp(\boldsymbol{Z}_k^T\boldsymbol{\beta}) \right) \right\}. \tag{1.11}$$

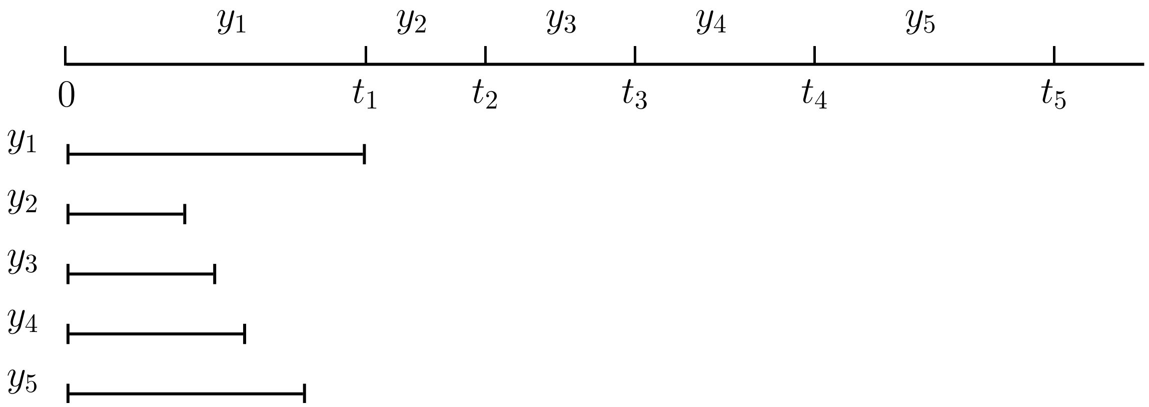

Figure 1.4: A sequence of spike times.

The partial score function is

$$l'(\boldsymbol{\beta}) = \sum_{j=1}^{J} \left\{ \boldsymbol{Z}_j - \frac{\sum_{k \in R(y_j)} \boldsymbol{Z}_k \exp(\boldsymbol{Z}_k^T \boldsymbol{\beta})}{\sum_{k \in R(y_j)} \exp(\boldsymbol{Z}_k^T \boldsymbol{\beta})} \right\}. \tag{1.12}$$

The Hessian matrix of the partial likelihood is

$$l''(\boldsymbol{\beta}) = \sum_{j=1}^{J} \left\{ -\frac{\sum_{k \in R(y_j)} \boldsymbol{Z}_k \boldsymbol{Z}_k^T \exp(\boldsymbol{Z}_k^T \boldsymbol{\beta})}{\sum_{k \in R(y_j)} \exp(\boldsymbol{Z}_k^T \boldsymbol{\beta})} + \frac{\sum_{k \in R(y_j)} \boldsymbol{Z}_k \exp(\boldsymbol{Z}_k^T \boldsymbol{\beta}) \sum_{k \in R(y_j)} \boldsymbol{Z}_k' \exp(\boldsymbol{Z}_k^T \boldsymbol{\beta})}{(\sum_{k \in R(y_j)} \exp(\boldsymbol{Z}_k^T \boldsymbol{\beta}))^2} \right\}. \tag{1.13}$$

Using this score function and Hessian matrix, the partial likelihood can be maximized by the Newton-Raphson algorithm, i.e., start with some $\boldsymbol{\beta}^{(0)}$. For steps $s = 0, 1, 2, ...$, iterate

$$\boldsymbol{\beta}^{(s+1)} = \boldsymbol{\beta}^{(s)} - \frac{l'(\boldsymbol{\beta}^{(s)})}{l''(\boldsymbol{\beta}^{(s)})} \tag{1.14}$$

until convergence.

The inverse of the Hessian matrix, evaluated at $\hat{\boldsymbol{\beta}}$, can be used as an approximate variance-covariance matrix for the estimate and used to produce approximate standard errors for the regression coefficients [Cox, 1972a].

If the fixed covariates $\boldsymbol{Z}$ are time-dependent, i.e., $\boldsymbol{Z}(t)$, then renewal models in which the conditional intensity function is of the form

$$\lambda(t|H_t) = h_0(t - T_{N(t-)}) \exp(\boldsymbol{Z}^T(t) \boldsymbol{\beta}) \tag{1.15}$$

can be considered. In principle, for model (1.15), $\boldsymbol{\beta}$ can be estimated by maximizing a similar partial likelihood function with (1.9) by replacing $\boldsymbol{Z}_j$ and $\boldsymbol{Z}_k$ with $\boldsymbol{Z}(t_j)$ and $\boldsymbol{Z}(t_{k-1}+y_j)$. However, when $\boldsymbol{Z}(t)$ varies continuously and the number of events is large, the model estimation becomes very computationally intensive. We further discuss this issue in Section 3.3.

Modeling a neural spike train as a renewal process can take the advantage of the widely used and well studied Cox proportional hazards model; however, the assumption of independent gap times is untenable in most situations. In situations that involve more complex relationships between event occurrence and prior event history, modulated renewal processes are useful. A modulated renewal process is a process whose conditional intensity depends not only on the time since the last event and the covariates but also on the past event history, i.e.,

$$\lambda(t|H_t) = h\{t - T_{N(t-)}, \boldsymbol{\beta}, T_{N(t-)}, T_{N(t-)-1}, ...\}. \tag{1.16}$$

In this study, we model the response spike train by a modulated renewal process and propose a multiplicative intensity model for a single modulated renewal process in Section 2.1. Without the independent gap times assumption, it is questionable whether treating a sequence of events in the same way as parallel events is appropriate or not. In other words, the performance of the maximum Cox partial likelihood estimator needs to be studied. We further discuss this issue in Section 2.3.3.

1.5 Review of variable selection

Variable selection is vital to statistical modeling. In practice, many covariates are often available that have potential effect on the response variable. At the initial stage of modeling, many predictors are usually introduced. Failing to select significant variables would lead to poor model prediction and interpretation. Therefore, variable selection plays a crucial role in model building and is very challenging in the presence of a large number of predictors [Fan et al., 2005].

Many variable selection approaches have been developed and extended to multiplicative intensity models such as the Cox proportional hazards model. Let us first review recent developments in variable selection for Cox proportional hazards model. Some traditional variable selection criteria, such as Akaike's Information Criterion (AIC; Akaike [1973]) and Bayesian information criterion (BIC; Schwarz [1978]) can be easily extended to Cox proportional hazards model. Tibshirani [1996] proposed the LASSO variable selection procedures for linear regression models and generalized linear models. It was further extended to the Cox proportional hazards model in Tibshirani [1997]. Fan and Li [2001] proposed nonconcave penalized approaches for linear regression, robust linear models and generalized linear models. They demonstrated the smoothly absolute clipped deviation (SCAD) possesses an oracle property, namely, the resulting estimate can correctly identify the true model as if it were known in advance. The LASSO does not possess this oracle property. Fan et al. [2005] derived a nonconcave penalized partial likelihood for the Cox proportional hazards model and the Cox frailty model, and they further illustrated the oracle property of their proposed procedures. The following is a review of LASSO penalty and SCAD penalty.

The LASSO penalty for a scalar parameter β is defined as

$$p_\lambda(\beta) = \lambda|\beta|. \tag{1.17}$$

The LASSO penalized log-partial likelihood criterion for a regression with a vector of parameters can be expressed as

$$\sum_{j=1}^{J} l_j(\boldsymbol{\beta}) - J\lambda \sum_{i=1}^{r} |\beta_i| = \sum_{j=1}^{J} l_j(\boldsymbol{\beta}) - J\lambda \|\boldsymbol{\beta}\|_1. \tag{1.18}$$

The penalty constrains the L_1 norm of the coefficient vector. Maximizing the LASSO penalized log-partial likelihood is equivalent to maximizing the log-partial likelihood subject to $\|\boldsymbol{\beta}\|_1 < s$, then finding the global maximum of the penalized partial likelihood over the values of s. Several algorithms have been developed to solve the LASSO optimization problem, such as the LARS algorithm [Efron et al., 2004], the coordinatewise gradient approach

[Shevade and Keerthi, 2003], and the combination of full gradient ascent optimization with the Newton-Raphson algorithm [Goeman, 2010].

The SCAD penalty for a scalar parameter β is defined as

$$p'_\lambda(|\beta|) = \lambda \left\{ I(|\beta| \leq \lambda) + \frac{(a\lambda - |\beta|)_+}{(a-1)\lambda} I(|\beta| > \lambda) \right\} \qquad \text{for some } a > 2. \tag{1.19}$$

It involves two unknown parameters, a and λ. Figure 1.5 shows LASSO, SCAD and another two popular penalties, which are the bridge penalty, $p_\lambda(|\beta|) = \lambda|\beta|^{0.5}$ and the ridge penalty, $p_\lambda(|\beta|) = \lambda\beta^2$. From Figure 1.5, SCAD is equivalent to LASSO when $|\beta|$ is small, but is constant when $|\beta|$ is large. This guarantees that large coefficients are not excessively penalized. The tuning parameter λ controls the range of β values in which SCAD is equivalent to LASSO and the other tuning parameter a controls the range of β values in which SCAD is constant. Fan and Li [2001] suggested using $a = 3.7$ from a Bayesian statistical point of view; this value will be used throughout this dissertation. The SCAD penalized log-partial

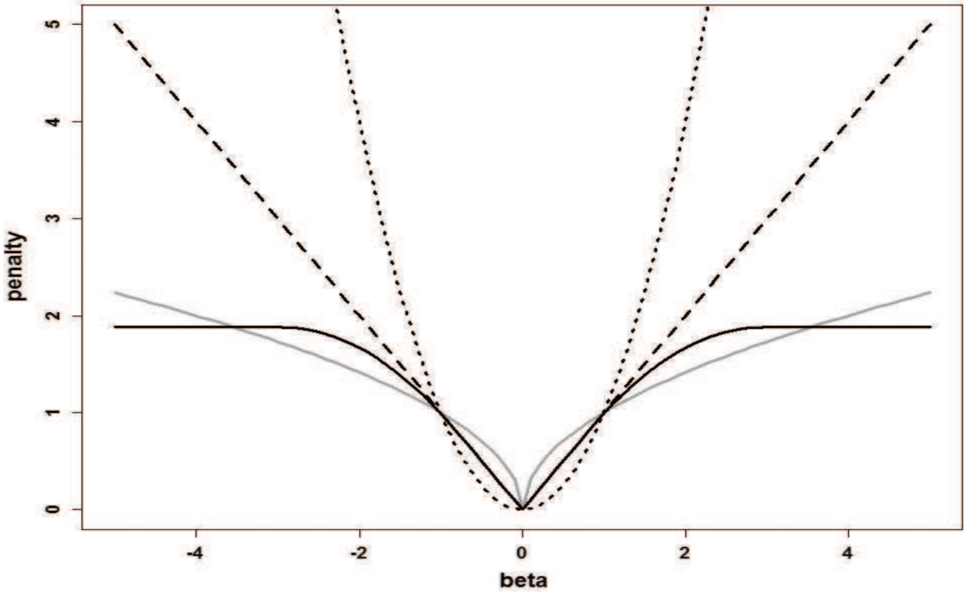

Figure 1.5: Plots of penalty functions: SCAD (Black solid), bridge (gray solid), LASSO (dashed) and ridge (dotted).

likelihood criterion for a regression with a vector of parameters can be expressed as

$$\sum_{j=1}^{J} l_j(\boldsymbol{\beta}) - J\sum_{i=1}^{r} p_\lambda(|\beta_i|). \tag{1.20}$$

As seen in Figure 1.5, the SCAD penalty is non-differentiable at the origin and nonconcave with respect to $\boldsymbol{\beta}$. Therefore, maximizing the SCAD penalized partial likelihood function is difficult. Approaches to solving this problem have been proposed by Fan and Li [2001] and Zou and Li [2008]. Fan and Li [2001] proposed locally approximating penalty function by a quadratic function. Suppose an initial value $\beta^{(0)}$ is given, and it is close to the true value of β, the penalty function is locally approximated by

$$p_\lambda(|\beta_i|) \approx p_\lambda(|\beta_i^{(0)}|) + \frac{1}{2}\{p'_\lambda(|\beta_i^{(0)}|)/|\beta_j^{(0)}|\}(\beta_i^2 - \beta_i^{(0)2}). \tag{1.21}$$

Take the unpenalized maximum partial likelihood estimate as the initial value, $\boldsymbol{\beta}^{(0)}$. For $s = 0, 1, 2, ...$, repeatedly solve

$$\boldsymbol{\beta}^{(s+1)} = \arg\max \left\{ \sum_{j=1}^{J} l_j(\boldsymbol{\beta}) - J\sum_{i=1}^{r} \frac{p'_\lambda(|\beta_i^{(s)}|)}{2|\beta_i^{(s)}|}\beta_j^2 \right\}. \tag{1.22}$$

Stop the iteration if the sequence of $\boldsymbol{\beta}^{(s)}$ converges.

Zou and Li [2008] pointed out that Fan and Li [2001]'s algorithm shared a drawback with backward stepwise variable selection: if a covariate is deleted at any step in the algorithm, it will necessarily be excluded from the final selected model. Zou and Li [2008] also proposed locally approximating the penalty function by a linear function. This approach is as follows. Suppose an initial value $\beta^{(0)}$ is given, and it is close to the true value of β, the penalty function is locally approximated by

$$p_\lambda(|\beta_i|) \approx p_\lambda(|\beta_i^{(0)}|) + p'_\lambda(|\beta_i^{(0)}|)(|\beta_i| - |\beta_i^{(0)}|). \tag{1.23}$$

Figure 1.6 illustrates the quadratic and linear approximations for the SCAD penalty. The optimization problem of the penalized partial likelihood can be solved as follows. Use the unpenalized maximum partial likelihood estimate as the initial value, $\boldsymbol{\beta}^{(0)}$. For $s = 0, 1, 2, ...,$ repeatedly solve

$$\boldsymbol{\beta}^{(s+1)} = \arg\max \left\{ \sum_{j=1}^{J} l_j(\boldsymbol{\beta}) - J \sum_{i=1}^{r} p'_{\lambda}(|\beta_i^{(s)}|)|\beta_i| \right\}. \tag{1.24}$$

Stop the iteration if the sequence of $\boldsymbol{\beta}^{(s)}$ converges [Zou and Li, 2008]. Note that (1.24) is a LASSO optimization problem and can be solved by previously mentioned algorithms.

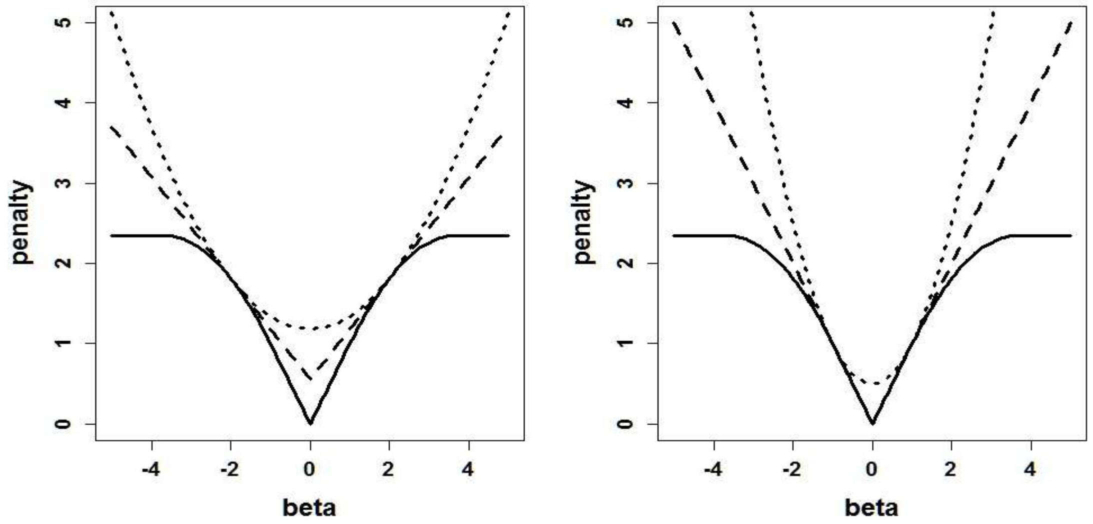

Figure 1.6: Plots of local quadratic approximation (dashed lines) and local linear approximation (dotted lines) at $\beta = 2$ and 1 for SCAD penalty (black solid lines) with $\lambda = 1$, $a = 3.7$.

The performance of the penalized approach depends on the tuning parameter λ, which balances the trade-off between goodness-of-fit and model sparsity. To select an appropriate λ, it is common to fit the model by the penalized approach using a sequence of λ values. Then the fitted models are compared based on a selected criterion, e.g., the minimization of the AIC, BIC, Generalized Cross Validation (GCV, Wahba [1990]) and for high dimension models, the extended BIC (EBIC; Chen and Chen [2008]).

The penalized approach has been widely studied and gained its popularity due to simultaneous parameter estimation and variable selection. In addition to the widely used LASSO and SCAD penalties, a number of recent papers have considered penalties such as the Adaptive LASSO [Zhang and Lu, 2007], Dantzig selector [Candes and Tao, 2007], group LASSO [Yuan and Lin, 2006] and group bridge [Huang et al., 2009].

1.6 Outline of the dissertation

This dissertation contains two topics of regression modeling for modulated renewal process. In Chapter 2, we propose a multiplicative intensity model for neural spike trains. A semi-parametric model estimation approach and a model evaluation method are covered. The large sample performance of the model is evaluated by simulation. Methodologies are applied on modeling a large real neural spike train data. In Chapter 3, we study penalized approaches for the proposed model to achieving sparsity of the coefficient functions when the sample size of the data is relatively small. LASSO and SCAD penalties are applied to the proposed model and their performances are evaluated by simulation. Approaches to reduce the computational complexity of model estimation are covered, too. The methods are applied to a small real neural spike train data. Discussions of future work are in Chapter 4.

CHAPTER 2

Modulated renewal process model for neural connectivity

2.1 The multiplicative modulated renewal process model

In this dissertation, our interest centers on studying the relationship between different spike trains by modeling them as point processes. Let $(0, T]$ denote the observation interval and $0 < t_1 < t_2 < ... < t_{J-1} < t_J \leq T$ be a set of J spike times of a spike train. For any time $t \in (0, T]$, let $N(t)$ denote the number of spikes that occurred in $(0, t]$. Note that $N(t)$ is a counting process, which is a right continuous function that jumps 1 at each spike time and is constant otherwise [Daley and Vere-Jones, 1988, Snyder and Miller, 1991]. The sample path $H_t = \{N(u), u \in (0, t)\}$ contains all information about the sequence of spike times up to, but not including time t [Brown et al., 2003]. Equivalently, we can write $H_t = \{0 < t_1 < t_2 < ... < t_{N(t-)} < t\}$, where $N(t-)$ is the number of spikes that occurred *before* time t. The *conditional intensity function* is widely used and is defined as

$$\lambda(t|H_t) = \lim_{\Delta t \to 0} \frac{Pr\left(N((t + \Delta t)-) - N(t-) = 1|H_t\right)}{\Delta t}, \text{ for } t \in (0, T]. \tag{2.1}$$

In survival analysis, the conditional intensity is termed the hazard function because $\lambda(t|H_t)$ measures the conditional probability density of a failure or death at time t, given that the process has survived up to time t [Brown et al., 2002]. If the conditional intensity function depends only on the time since the last spike, i.e., $\lambda(t|H_t) = \lambda(t - t_{N(t-)})$ then the process is a renewal process [Cinlar, 1969]. An equivalent condition is that the lengths of interspike intervals (increments) are independent and identically distributed. Although renewal processes are often used to model spike trains, in practice, neuronal spike trains rarely satisfy the independent and stationary increments assumptions. The intensity of a neural spike train usually is affected by other spike trains as well as its own history. Here, we choose to use a modulated renewal process to model a neural spike train because it can reflect trends and dependencies in a point process.

To study the relationship among the neural spike trains, we select one spike train as the response train and treat the others as predictor spike trains. Our primary goal is to develop a regression-type model that captures the causal relationships between the response and the predictor spike trains. The response train is modeled by a modulated renewal process; as in most regression studies, the predictor trains are treated as fixed. Let x_i be the set of spike times of the ith predictor spike train and denote the number of spikes up to time t as $n_{x_i}(t)$, $i = 1, 2, ..., p$. $n_{x_i}(t)$ defines a counting measure on the measurable space $\{x_i, \Sigma_{x_i}\}$, where Σ_{x_i} is the σ-field of measurable subsets consisting of all subsets of x_i.

Since a stochastic point process may be characterized by its conditional intensity function, the relationship between the response spike train and the predictor spike trains may be modeled using the conditional intensity function. We consider a multiplicative model that assumes the conditional intensity function of the response spike train at t is the product of a baseline hazard function, a function that captures the impact of the predictor trains, a function that captures the impact of the response train's own history and other predictors such as stimulus and behavioral variables. This model is an extension of the Cox proportional hazards model. The model is defined as

$$\lambda(t|H_t) = \lambda(t, t - t_{N(t-)}, t - t_{N(t-)-1}, .., t - t_1) \tag{2.2}$$

$$= \lambda_0(t - t_{N(t-)})\Psi(t) \prod_{j=1}^{N(t-)} \lambda_1(t - t_j)\Phi(t). \tag{2.3}$$

Equation(2.3) is the product of four factors that affect the conditional intensity. The factors are:

(1) $\lambda_0(\cdot)-$ baseline hazard function;

(2) $\Psi(t)-$ function that captures the impact of the predictor trains;

(3) $\lambda_1(\cdot)-$ function that captures the impact of the response train's own history;

(4) $\Phi(t)-$ function that captures the impact of extrinsic variables.

We consider the following specific model for each of these four factors.

(1) $\lambda_0(\cdot)$ is an unknown baseline hazard function of the time since the most recent spike. We do not add any constrains to the form of $\lambda_0(\cdot)$.

(2) $\Psi(t)$ is a log-linear function of integrals of unknown coefficient functions with respect to the counting measures $n_{x_i}(t)$ over the most recent M seconds of t; that is

$$\Psi(t) = \exp\left\{\sum_{i=1}^{p}\int_{t-M}^{t}\kappa_i(t-u)dn_{x_i}(u)\right\}. \tag{2.4}$$

$\Psi(t)$ models the effects of the predictor spikes in $[t-M, t)$, where M is a positive constant that determines the length of the predictor spikes' "history" that has an impact on the next response spike. A sufficiently large M can be chosen for all predictor spike trains, or a separate M_i value can be chosen for each predictor spike train. $\kappa_i(\cdot)$, $i = 1, 2, ..., p$ is an unknown coefficient function that measures the impact strength of the ith predictor spike train. When M is sufficiently large, $\Psi(t)$ allows the model to capture the dependence of the response spike train on the predictor spikes' history. Note that $\Psi(t)$ is a function of t; that is, the predictors are time-dependent. The coefficient function $\kappa_i(t-u)$ represents the impact strength of a spike in the ith predictor train at time u. Note that the impact strength of a predictor spike at u is determined by its distance from time t. This is illustrated in Figure 2.1.

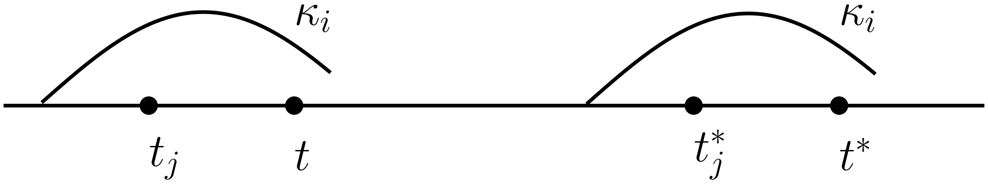

Figure 2.1: Interpretation of κ function. Two spikes in the same predictor spike train has the same impact on future response intensity, as long as the distance between the spike time and the future time is the same.

(3) $\lambda_1(u)$ is an unknown function that reflects the effect of a response spike at time $t-u$ on the conditional intensity function at t. This function is evaluated for all response spikes

in history, so the effect of any past response spike only depends on the difference between its time and t. We further assume

$$\lambda_1(u) = \exp\left\{\kappa_0(u)\right\}, \tag{2.5}$$

where $\kappa_0(\cdot)$ is a unknown coefficient function that represents the impact strength of a past response spike on the conditional intensity at a future time and $\kappa_0(u) = 0, u > M_0$, which may be taken for simplicity as equal to the previous M. Note that if a past response spike t_j is not the most recent response spike prior t, its impact is $\lambda_1(t - t_j)$. However, if it is the most recent response spike prior t, its impact is $\lambda_0(t - t_j)\lambda_1(t - t_j)$. This is illustrated by Figure 2.2. Therefore, $\lambda_0(\cdot)$ describes the unique effect of the most recent response spike and can capture neural properties such as recovery period. An alternative way of setting up the model would be not to treat the most recent response spike as part of the response train's history, i.e. in (2.3), the product of $\lambda_1(\cdot)$s would not include $\lambda_1(t - t_{N(t-)})$, i.e.,

$$\prod_{j=1}^{N(t-)-1} \lambda_1(t - t_j). \tag{2.6}$$

Then the impact of the most recent response spike is described completely by $\lambda_0(\cdot)$. We do not expect the two model setups would be different in predicting future spikes. However, the interpretations of the baseline hazard function are different. In this paper, we choose to use the first model setup because it can show the difference between the impact of the most recent response spike and the impact of other past response spikes directly.

(4) $\Phi(t)$ is a function of predictors that include information beyond neural spiking activities, such as external stimulus and behavioral variables of the subject. The form of $\Phi(t)$ depends on the nature of the predictors, $\boldsymbol{Z} = (Z_1, Z_2, ..., Z_q)^T$. For example, if the predictors describe the status of the subject and do not change over time, then $\Phi(t)$ can be as simple as $\exp(\boldsymbol{\beta}^B \boldsymbol{Z})$, where $\boldsymbol{\beta}^B = (\beta_1^B, ..., \beta_q^B)$ is the vector of coefficients. It can be more complex, such as $\exp(\boldsymbol{\beta}^B \boldsymbol{Z}(t))$, when the predictors are time-dependent, or $\Phi(t) = \exp\left(\sum_{i=1}^q \int_{t-M}^t \kappa_i^B(t-u) Z_i(t) du\right)$ when the predictors describe information in some time interval instead of at certain time. Here, we pick $\Phi(t) = \exp(\boldsymbol{\beta}^B \boldsymbol{Z}(t))$ as an example.

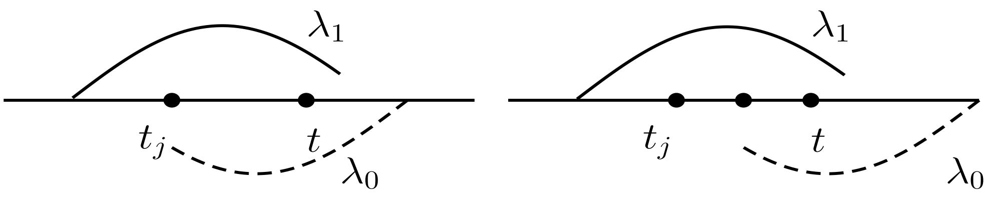

Figure 2.2: Interpretation of $\lambda_1(\cdot)$ (solid curves) and the baseline hazard function $\lambda_0(\cdot)$ (dashed curves). If a response spike at t_j is the most recent response spike, its impact is represented by the product of $\lambda_1(t - t_j)$ and $\lambda_0(t - t_j)$; otherwise, its impact is represented only by $\lambda_1(t - t_j)$.

Combine the four model components, and the model can be written as,

$$\begin{aligned}\lambda(t|H_t) &= \lambda_0(t - t_{N(t-)}) \exp\left\{\sum_{i=1}^{p} \int_{t-M}^{t} \kappa_i(t-u)dn_{x_i}(u) + \sum_{j=1}^{N(t-)} \kappa_0(t-t_j) + \boldsymbol{\beta}^B \boldsymbol{Z}(t)\right\} \\ &= \lambda_0(t - t_{N(t-)}) \exp\left\{\sum_{i=0}^{p} \int_{t-M}^{t} \kappa_i(t-u)dn_{x_i}(u) + \boldsymbol{\beta}^B \boldsymbol{Z}(t)\right\} \qquad (2.7)\end{aligned}$$

where $n_{x_0}(t)$ is the number of response spikes up to time t. Figure 2.3 illustrates the structure of the model.

2.2 Special cases of the multiplicative modulated renewal process model

There are four special cases of our model worth noting, the first two are used in the Section 2.5 Simulation as simple versions of the model. The third and fourth cases are models that have been used for analyzing neuron spike train data in other studies. Special cases:

Case 1 No response history, time independent model. The conditional intensity function is the product of two components,

$$\lambda(t|H_t) = \lambda(t - t_{N(t-)}, t_{N(t-)}) = \lambda_0(t - t_{N(t-)})\Psi(t_{N(t-)})\Phi(t_{N(t-)}). \qquad (2.8)$$

This model does not include the response history component, i.e., it assumes that the inter-spike times are independent. Also, note that the impact of the predictor trains'

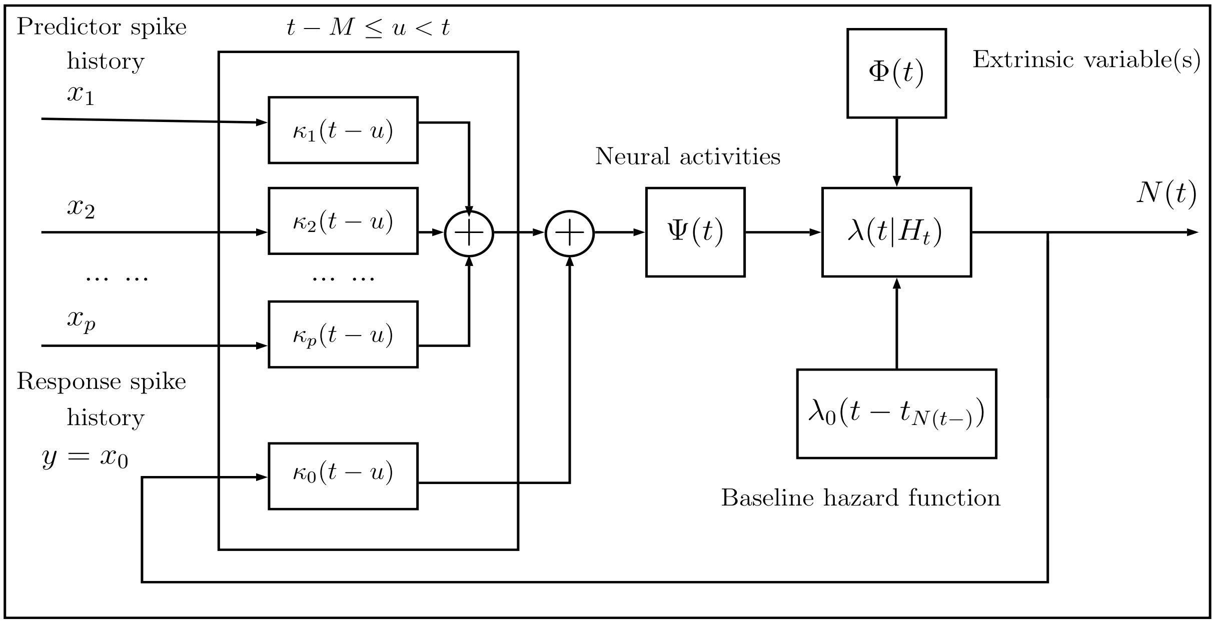

Figure 2.3: Multiplicative modulated renewal process model for neural spike trains.

history is represented by $\Psi(t_{N(t-)})$ and does not vary for $t > t_{N(t-)}$, so for each inter-spike time, the predictors only include the history up to the most recent response spike time before t. We call them time-independent predictors and the “time" is referring to the time since the most recent spike $t - t_{N(t-)}$, not time t. In this case, the inter-spike times are independent random variables whose distributions are determined by the common baseline hazard function, impact of predictor spikes and the extrinsic variables up to the most recent response spike.

Case 2 Response history, time independent model. The difference between this model and the model in Case 1 is that this model includes the response history component, but the predictors are assumed to be time-independent.

$$\lambda(t|H_t) = \lambda_0(t - t_{N(t-)})\Psi(t_{N(t-)}) \prod_{i=1}^{N(t-)-1} \lambda_1(t_{N(t-)} - t_{N(t-)-i})\Phi(t_{N(t-)}). \qquad (2.9)$$

In this case, the inter-spike times are not independent.

Case 3 Inhomogeneous renewal process model, i.e., no response history, time dependent model. In the model (2.7), let $\kappa_0(u) \equiv 0$, so the model does not include the response history component, it becomes

$$\lambda(t|H_t) = \lambda(t, t - t_{N(t-)}) = \lambda_0(t - t_{N(t-)})\Psi(t)\Phi(t), \tag{2.10}$$

then the response spike train is modeled by an inhomogeneous renewal process. The inter-spike times are independent variables whose distributions are determined by the common baseline hazard function, impact of predictor spikes and the extrinsic variables up to time t. The inhomogeneous Markov interval process model proposed by Kass and Ventura [2001] is an example of such models.

Case 4 Inhomogeneous Poisson process model, i.e. no response history, constant baseline hazard model. In the model (2.7), let $\kappa_0(u) \equiv 0$ *and* the baseline hazard function $\lambda_0(u) \equiv C$, C is a constant, the model becomes

$$\lambda(t|H_t) = C\Psi(t)\Phi(t), \tag{2.11}$$

then the response spike train is modeled by an inhomogeneous Poisson process. By including response history as predictor(s), we get an extension of the inhomogeneous Poisson process model as,

$$\lambda(t|H_t) = C\Psi(t) \prod_{j=1}^{N(t-)} \lambda_1(t - t_j)\Phi(t). \tag{2.12}$$

The inter-spike times are not independent. Their conditional distributions given previous response spikes are Poisson distributions. The model proposed by Truccolo et al. [2005] is an example of these models.

2.3 Model estimation

2.3.1 B-spline approximation to the coefficient functions

In the proposed model, estimating the coefficient functions κ_i is challenging, because little information is known about the form of κ_i. We use a nonparametric approach that does not require specifying the form of κ_i. While the form is left unspecified, it is intuitively sensible to assume some degree of smoothness for the κ_i, so we assume that they have $d-1$ continuous derivatives for $d \geq 1$ almost everywhere on $[0, M]$. A natural approximation family for the κ_i, is the linear space spanned by B-splines. Another sensible assumption is: if M is sufficiently large, $\kappa_i(u) = 0$, for $u \geq M$, i.e., the impact of a previous spike is neglectable if it occurs far from the current time.

B-spline basis functions are polynomial segments jointed end-to-end at argument values called knots, breaks or join points. The segments have specifiable smoothness across these breaks. Let $\eta_0 = 0 < \eta_1 < \cdots < \eta_m < \eta_{m+1} = M$ be a knot sequence with m interior knots and d be the degree of B-spline basis, a B-spline basis function can be defined recursively using the Cox-de Boor recursion formula

$$B_{k,d}(u) = \frac{u-\eta_k}{\eta_{k+d}-\eta_k} B_{k,d}(u) + \frac{\eta_{k+d+1}-u}{\eta_{k+d+1}-\eta_{k+1}} B_{k+1,d}(u) \tag{2.13}$$

and

$$B_{k,0}(u) = \begin{cases} 1 & \text{if} \quad \eta_k \leq u < \eta_{k+1}, \\ 0 & \qquad \text{otherwise,} \end{cases} \qquad k = 0, 1, ..., m. \tag{2.14}$$

Figure 2.4 gives an example of B-spline basis functions with degree $d = 3$ and 9 interior knots. B-spline basis functions have the advantages of very fast computation and great flexibility. For more discussion on spline functions, see Schumaker [1980] and de Boor [2001].

The proposed model can be approximated by replacing $\kappa_i(t-u)$ with a linear combination of B-spline basis functions, i.e.,

$$\kappa_i(t-u) = \sum_{k=1}^{K} \beta_{ik}^{A} B_k(t-u), \quad v \in [0, M], \quad i = 0, 1, ...p \tag{2.15}$$

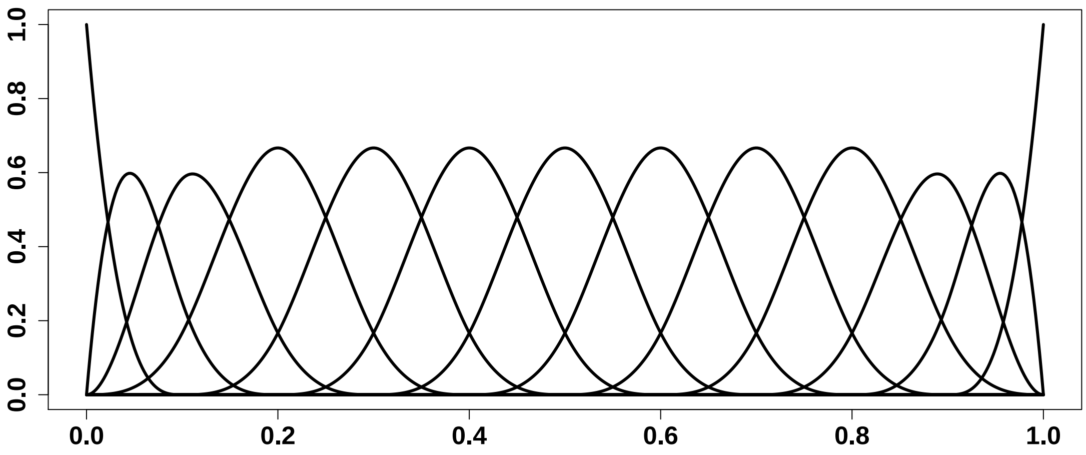

Figure 2.4: B-spline basis functions in line plots. The degree d is 3 and 9 interior knots are evenly space between 0 and 1.

where for fixed d, B_k is the kth B-spline basis function , $k = 1, 2, ..., K = m + d + 1$. β_{ik}^A is the coefficient for the ith spike train and the kth B-spline basis function. Thus,

$$\begin{aligned}\lambda(t|H_t) &= \lambda_0(t - t_{N(t-)}) \exp\left\{\sum_{i=0}^{p} \int_{t-M}^{t} \kappa_i(t-u) dn_{x_i}(u) + \boldsymbol{\beta}^B \boldsymbol{Z}(t)\right\} && (2.16)\\ &\simeq \lambda_0(t - t_{N(t-)}) \exp\left\{\sum_{i=0}^{p} \int_{t-M}^{t} \sum_{k=1}^{K} \beta_{ik}^A B_k(t-u) dn_{x_i}(t) + \boldsymbol{\beta}^B \boldsymbol{Z}(t)\right\} && (2.17)\\ &= \lambda_0(t - t_{N(t-)}) \exp\left\{\sum_{i=0}^{p} \sum_{k=1}^{K} \beta_{ik}^A D_{ik}(t) + \boldsymbol{\beta}^B \boldsymbol{Z}(t)\right\}, && (2.18)\end{aligned}$$

where $D_{ik}(t) = \int_{t-M}^{t} B_k(t-u) dn_{x_i}(u)$, $\boldsymbol{\beta} = \{\beta_{ik}^A, \boldsymbol{\beta}^B\}$ is the vector of coefficients. Denote $\exp\left\{\sum_{i=0}^{p} \sum_{k=1}^{K} \beta_{ik}^A D_{ik}(t) + \boldsymbol{\beta}^B \boldsymbol{Z}(t)\right\}$ by $\Psi^*(t, \boldsymbol{\beta})$. Then $\Psi^*(t, \boldsymbol{\beta})$ represents the combined impact of the neural activities and the extrinsic variables. The conditional intensity can be written as the product of the baseline hazard function and $\Psi^*(t, \boldsymbol{\beta})$, that is

$$\lambda(t|H_t) \simeq \lambda_0(t - t_{N(t-)}) \Psi^*(t, \boldsymbol{\beta}). \qquad (2.19)$$

This is similar with the Cox's proportional hazards model with time dependent covariates. The problem of estimating the coefficient functions becomes estimating coefficients of the B-splines bases.

2.3.2 The full likelihood

When the form of the baseline hazard function $\lambda_0(\cdot)$ is specified, $\lambda(t|H_t)$ is parametric. Then the parameter vector $\boldsymbol{\beta}$ can be estimated by maximizing the likelihood function. As in all likelihood analyses, the likelihood function for a continuous time point process is formulated by deriving the joint probability density of the spike train. Let $\{0 < t_1 < t_2 < ... < t_J \leq T\}$ denote J observed response spike times. $\{0 < t_1 < t_2 < ... < t_J\}$ can be treated as a realization of J dependent random variables $\{0 < T_1 < T_2 < ... < T_J\}$ with joint density $f(t_1, t_2, ...t_J)$. The joint density can be written as the product of conditional densities:

$$f(t_1, t_2, ...t_J) = \prod_{j=1}^{J} f_j(t_j|T_i = t_i, i \leq j-1) \tag{2.20}$$

The conditional intensity function can be written in terms of the spike time conditional density as,

$$\lambda(t|H_t) = \frac{f_{N(t-)+1}(t|H^*_{N(t-)})}{1 - \int_{t_{N(t-)}}^{t} f_{N(t-)+1}(u|H^*_{N(t-)})du}. \tag{2.21}$$

$H^*_s = \{T_i = t_i, i \leq s\}$ contains the first s spike times, then $H^*_{N(t-)} = \{T_i = t_i, i \leq N(t-)\}$ contains all the spike times prior to time t. The likelihood function of $\{0 < t_1 < t_2 < ... < t_J \leq T\}$ is the product of the joint density of the first J spike times and conditional probability that the $J + 1$th spike occurs after T given the first J spike times. It can be written in terms of the conditional intensity functions using the relationship between the conditional density and the conditional intensity function [Barbieri et al., 2001, Brown et al., 2002, Daley and Vere-Jones, 1988],

$$\begin{aligned}
L &= f(t_1, t_2, ...t_J)Pr(T_{j+1} > T|T_i = t_i, i \leq J) && (2.22)\\
&= \prod_{j=1}^{J} f_j(t_j|H^*_{j-1})Pr(T_{j+1} > T|H^*_J) && (2.23)\\
&= \prod_{j=1}^{J} \lambda(t_j|H_{t_j}) \exp\left(-\int_{t_{j-1}}^{t_j} \lambda(u|H_u)du\right) \exp\left(-\int_{t_J}^{T} \lambda(u|H_u)du\right). && (2.24)
\end{aligned}$$

2.3.3 The maximum partial likelihood estimation

When $\lambda_0(\cdot)$ is left unspecified, the use of the likelihood function becomes problematic. Cox [1972a] proposed approach for estimating the parameters of interest in survival analysis without specifying the form of the baseline hazard by maximizing a component of the likelihood called the Cox partial likelihood instead of the full likelihood. In the survival analysis context, t_j is the failure time of the jth independent subject. The Cox partial likelihood is the product of the ratio of the hazard value at t_j of the subject that is observed to fail at t_j and the sum of the hazard values at t_j of the subjects in the risk set of t_j. The risk set of t_j is defined as the set of the subjects that survive until t_j or later.

For our proposed model, the Cox's partial likelihood approach is adapted as follows. Define the observed inter-spike time as $y_j = t_j - t_{j-1}$ and assume $y_i \neq y_j$ for $i \neq j$. Define the risk set of the jth spike as $R(t_j) = \{i, y_i \geq y_j\}$. Note that the risk set is defined using the inter-spike time y_j instead of the spike time t. $\boldsymbol{\beta}$ can be estimated by maximizing the following Cox partial likelihood with Ψ^* as defined in (2.18),

$$PL(\boldsymbol{\beta}) = \prod_{j=1}^{J} \frac{\Psi^*(t_{j-1} + y_j, \boldsymbol{\beta})}{\sum_{k \in R(t_j)} \Psi^*(t_{k-1} + y_j, \boldsymbol{\beta})}. \tag{2.25}$$

And the cumulative baseline hazard can be estimated by

$$\widehat{CH_0}(y) = \sum_{y_j \leq y} \frac{1}{\sum_{k \in R(t_j)} I\{y_k \geq y_j\} \Psi^*(t_{k-1} + y_j, \hat{\boldsymbol{\beta}})}. \tag{2.26}$$

Rigorous developments of the asymptotic theory underlying the Cox's partial likelihood approach for independent event times were given by Andersen and Gill [1982] and Tsiatis [1981]. However, Oakes [1981] pointed out that, for *dependent* event times, the reordering of the timescales invalidates the conditioning argument underlying the partial likelihood, i.e. when the event times are not independent, the estimators $\hat{\boldsymbol{\beta}}$ and $\widehat{CH_0}(t)$ may not have the same limiting distributions that they would have when the event times are independent. A more intuitive way to see this problem is that the risk set $R(t_j) = \{i, y_i \geq y_j\}$ contains some of the time-to-event outcomes that have already occurred at t_j, so they are not really "at risk" at time t_j. It has been shown that under certain conditions, the Cox's partial likelihood is valid for modulated renewal process models [Lin and Fine, 2009, Oakes and Cui, 1994, Pons and de Turckheim, 1988]. However, those conditions are very difficult to check in practice. Generally, the conditions require that the dependence structure of the modulated renewal process should not be too strong, so the process is "stationary" in some sense.

2.4 Evaluation of model goodness-of-fit and prediction power

The modulated renewal process model described in this study estimates the conditional distributions of inter-spike times. One approach to assessing model goodness-of-fit is the Kolmogorov-Smirnov (KS) test based on the time-rescaling theorem [Brown et al., 2002, Haslinger et al., 2010]. According to the time-rescaling theorem, if the data truly came from the estimated distributions of the inter-spike times, they can be rescaled into an independent uniform random variable with simple variable conversions. Therefore, the standard KS plot of the rescaled inter-spike times should show a 45-degree diagonal line. In this case, the standard KS plot is produced as follows,

1. For the jth response spike time, compute $\tau_j = \int_{t_{j-1}}^{t_j} \hat{\lambda}(t|H_t)dt$. This is equal to the value of the estimated cumulative conditional intensity function of the jth inter-spike time at $y_j = t_j - t_{j-1}$.

2. Compute $z_j = 1 - \exp\{\tau_j\}$. This is equal to the value of the estimated conditional cumulative distribution function of the jth inter-spike time at y_j.

3. Order z_j from the smallest to the largest, denoting the ordered values as $z_{(j)}$.

4. Plot the $z_{(j)}$ versus J equally spaced values of the cumulative distribution function of uniform(0,1) distribution, as $b_j = \frac{j-0.5}{J}$ for $j = 1, 2, .., J$

To gain better visualization of the difference between the standard KS curve and the diagonal line, a scaled horizontal KS plot is produced by adding three more steps:

5. Compute the differences between the values of the cumulative distribution function of uniform(0,1) distribution defined as $b_j = \frac{j-0.5}{J}$ for $j = 1, 2, .., J$ and $z_{(j)}$.

6. Scale the differences $b_j - z_{(j)}$ by diving them by half of the width of the 95% confidence interval, which is approximately equal to $1.36/J^{1/2}$[Johnson and Kotz, 1970].

7. Plot the scaled differences against b_j.

A scaled horizontal KS plot between -1 and 1 is within the 95% confidence bounds. Since we treat time as continuous and our model does not require that we discretize the response spike times into bins, i.e., the distribution of the inter-spike time is continuous, this method does not suffer the bias problem pointed out by Haslinger et al. [2010] for binned data. We conduct a small simulation to demonstrate this. 200 data sets are generated using the following modulated renewal process model (2.27).

$$\lambda(t|H_t) = \lambda_0(t - t_{N(t-)}) \exp\left\{\sum_{i=0}^{p} \int_{t-M}^{t} \kappa_i(t-u) dn_{x_i}(u)\right\} \tag{2.27}$$

Four ($p = 4$) neuron spike trains are chosen from the data set described in Section 1.2 as predictor trains. The baseline function $\lambda_0(y)$ is equal to 0 when $y \leq 0.002$, which represents a resting period, and a constant 2.5 when $y > 0.002$. The length of history M is 1 second. The conditional intensity of the response also depends on its own history n_{x_0} through κ_0. To reduce computational complexity, we assume that the history impact changes every 0.004 second instead of changing constantly. The true coefficient functions are in the space spanned by a linear combination of B-spline basis functions with $d = 3$ degrees and 9 equally spaced interior knots. The knots are 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9. For simplicity, we choose the same group of basis functions to represent all the coefficient functions.

The true coefficient functions are shown in Figure 2.5. κ_1 is positive in a short interval close to 0 and constant 0 elsewhere. It represents that a predictor spike close to time t has a strong impact on the condition intensity of the response at t and the impact strength decreases quickly as the distance between the predictor and t increases. κ_2 is also positive in a short interval close to 0 and constant 0 elsewhere, but it has a bell shape and represents increasing then decreasing impact strength of the predictor's history. κ_3 is constant zero, so the third predictor train has no impact on the response train. κ_4 is negative in a short interval close to 0 followed by a positive component. It represents a more complex relationship between the fourth predictor train and the response train. κ_0 is the coefficient function of the response's own history. It is 0 in a short interval close to 0 followed by a bell shape positive component. This means the response's history is positively related to its future activity with a short lag.

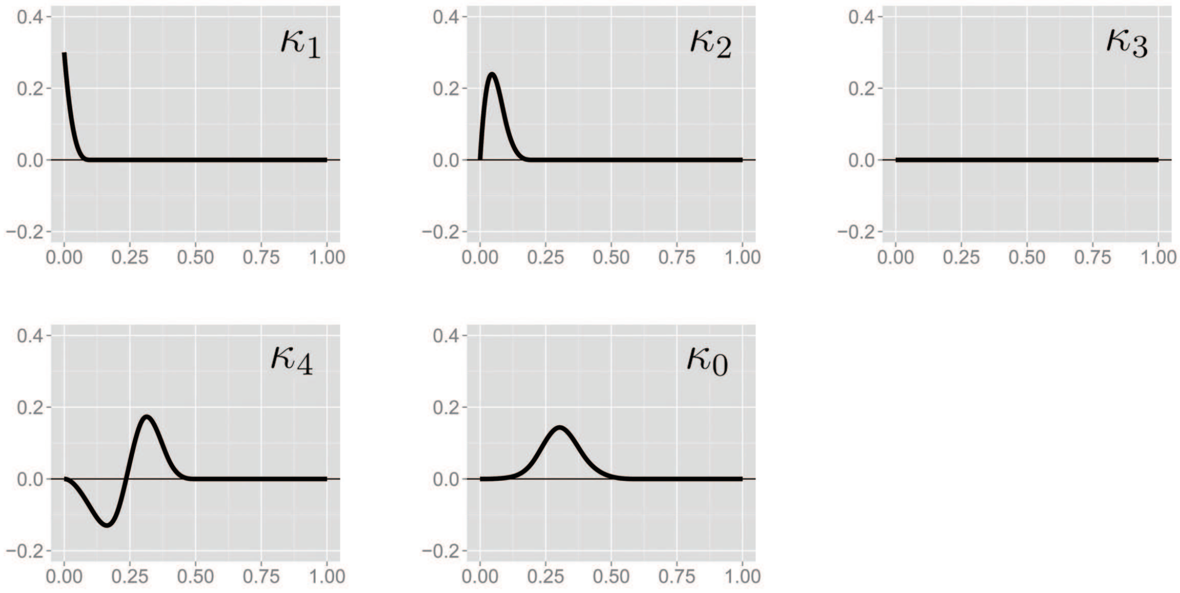

Figure 2.5: True coefficient functions of the predictor trains: κ_1-κ_4 and true coefficient function of the response history, κ_0.

The generated data sets and the true model with true coefficient functions and true baseline hazard function are used to compute 200 unscaled horizontal KS plots. The mean curve of these KS plots is shown in Figure 2.6 and is compared with 50 mean curves of 200 KS curves computed using realizations of Uniform(0,1). From Figure 2.6, the mean KS curve comparing the model generated data with their true distributions falls within the region spanned by the mean KS curves comparing Uniform(0,1) generated data with

Uniform(0,1) distribution. Therefore, on average, if the estimated distribution is close to the true distribution , the previously described method should produce a KS plot that is close to the diagonal line. Also we do not see any bias and as expected, the horizontal KS plot is closer to 0 as the sample size increases.

In some situations, the precision of the spike time is low, then it is reasonable to treat it as discrete time data. A discrete version of the modulated renewal process model can be easily obtained and applicable to discrete time data. In those situations, goodness-of-fit of the model can be evaluated by approaches proposed by Haslinger et al. [2010] based on the discrete time-rescaling theorem.

Since the baseline hazard function is nonparametric, using the in-sample KS plot to evaluate the goodness of fit of the modulated renewal process model can be misleading. The null model can have good in-sample KS plot because it under smooths the baseline hazard function. If the purpose of the model is prediction, the out-of-sample KS plot based on a validation data set would be useful because it shows the prediction performance of the model. If the purpose of the model is to quantify relationships between response and predictors, the distance between the KS plot computed using the fitted model and the KS plot computed using only the estimated baseline function would provide a good measure of the explanatory power of the model.

2.5 Simulation: large sample performance of the maximum partial likelihood estimator

In this section, we use simulation to evaluate the performance of the maximum partial likelihood estimator for the multiplicative modulated renewal process model. We focus on three questions. (1) Can the coefficient functions be adequately estimated by using B-spline approximations? (2) Does the performance of the maximum partial likelihood estimator for a process that depends on its own history differ from the performance for a process that does not depend on its own history? (3) Does the performance change when the predictors are time-dependent? To answer these questions, we conduct simulations under three scenarios

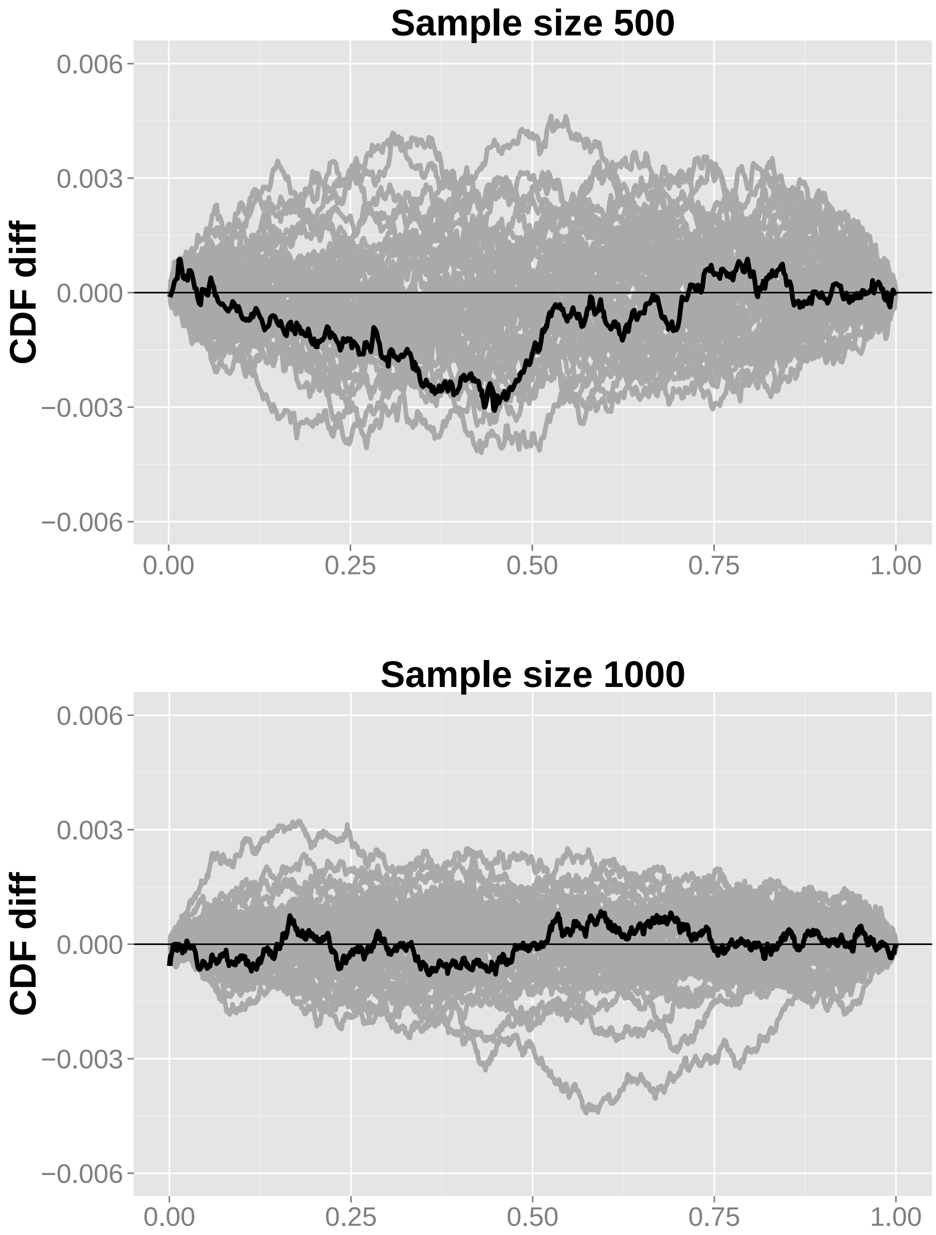

Figure 2.6: Mean horizontal KS plots based on 200 repetitions of 500 and 1000 response spikes (dark curves); 50 mean horizontal KS plots based on 200 repetitions of 500 and 1000 observations generated from $U(0, 1)$ (gray curves).

with different true coefficient functions and in each scenario, three models are used to generate the response train. Thus, there are nine scenario and model combinations. In the first two scenarios, the true coefficient functions are in the space spanned by a linear combination of B-spine basis functions, but in the third scenario the true coefficient functions are not. The first model generates a process that does not depend on its history. The other two generate processes that depend on its history and in addition, the third model includes time-dependent covariates. The generated data are fitted using the models that generate them. Then the estimated coefficient functions are compared with the true coefficient functions to evaluate the performance of the estimator.

Two neuron spike trains are chosen from the data set described in Section 1.2 as predictor trains. Three models are used to generate the response train. The first model is the special Case 1 of our proposed model described in Section 2.2, the no response history and time independent model (2.8), which does not include response history component and the predictors' histories are not time-dependent. The second model is Case 2, the response history and time independent model (2.9). It includes response history component but the histories are still not time-dependent. The third model is our proposed model, model (2.7), which includes the response history component and time-dependent predictors. For all three models (Table 2.1), the baseline function $\lambda_0(y)$ is equal to 0 when $y \leq 0.002$, which represents a resting period, and a constant 2 when $y > 0.002$. The length of history M is 1 second. For the third model, to reduce the computational complexity, we assume that the predictors change every 0.5 second.

Table 2.1: Three models considered in simulation

Model 1	Case 1	No response history, time independent predictors.
Model 2	Case 2	Response history, time independent predictors.
Model 3	Proposed	Response history, time dependent predictors.

In the first two scenarios, all coefficient functions are in the space spanned by a linear combination of B-spline basis functions. B-spline basis functions with degree $d = 3$ and

16 interior knots evenly distributed on $[0,1]$ are used in the simulation. For simplicity, we choose the same group of basis functions to represent or approximate all the coefficient functions, and denote them by $B_1, B_2, ..., B_{20}$. The true values of the B-spline coefficients are shown in Table 2.3 and the true coefficient functions are shown in Figure 2.7 and Figure 2.8. In Scenario 1 (S1), the coefficient functions of the predictors' histories have large positive values near zero and are equal to zero elsewhere. They reflect strong but short lasting history impacts. In Scenario 2 (S2), the coefficient functions of the predictors' histories have smaller positive values than those in Scenario 1. But their nonzero intervals are larger. They reflect weaker but longer lasting history impacts. The coefficient functions of the response's history in Scenario 1 and Scenario 2 both have large positive values near zero and short nonzero intervals. In the third scenario (S3) the coefficient functions are outside the B-spline space. We choose $\kappa(u) = \exp(-\alpha(1-u))(1-\exp(-\beta(1-u))$ as the true coefficient function. In all the three scenarios (Table 2.2), the integrated absolute coefficient functions are equal to 0.02 for the predictors' histories and 0.01 for the response's history. In each scenario, 10,000 response spikes are generated using each of the three models. Then the data are fitted using the true model with unknown coefficient functions and baseline hazard function. This is repeated 100 times.

Table 2.2: Three simulation scenarios

S1	In B-spline space	Strong but short lasting effects
S2	In B-spline space	Moderate and long lasting effects
S3	Out of B-spline space	Both strong and moderate effects

Figure 2.7 shows the estimated coefficient functions for model 1 in the three scenarios (three columns). Model 1 does not include the response train's history as predictors, so there are two coefficient functions (two rows). The estimated coefficient functions for model 2 and model 3 are shown in Figure 2.8 and Figure 2.9. These two models include the response train's history as predictors, so there are three coefficient functions in each scenario. For all the 9 model and scenario combinations, the true coefficient functions lie in the middle

Table 2.3: True coefficient values of two predictor trains and the response for Scenarios 1 and 2.

S1											
κ_1	0	0.341	0	0.170	0	0	0	0	0	0	0,...,0
κ_2	0	0	0.124	0	0.248	0	0	0	0	0	0,...,0
κ_0	0	0.179	0.108	0	0	0	0	0	0	0	0,...,0
S2											
κ_1	0.095	0.095	0.063	0.032	0.016	0.032	0.063	0.047	0.024	0.008	0,...,0
κ_2	0	0	0.010	0.029	0.076	0	0.114	0.076	0.029	0.010	0,...,0
κ_0	0	0.179	0.108	0	0	0	0	0	0	0	0,...,0

of the gray area shaded by the estimated coefficient functions and the gray area is fairly narrow, which indicates the estimation procedure can give good estimates for the coefficient functions.

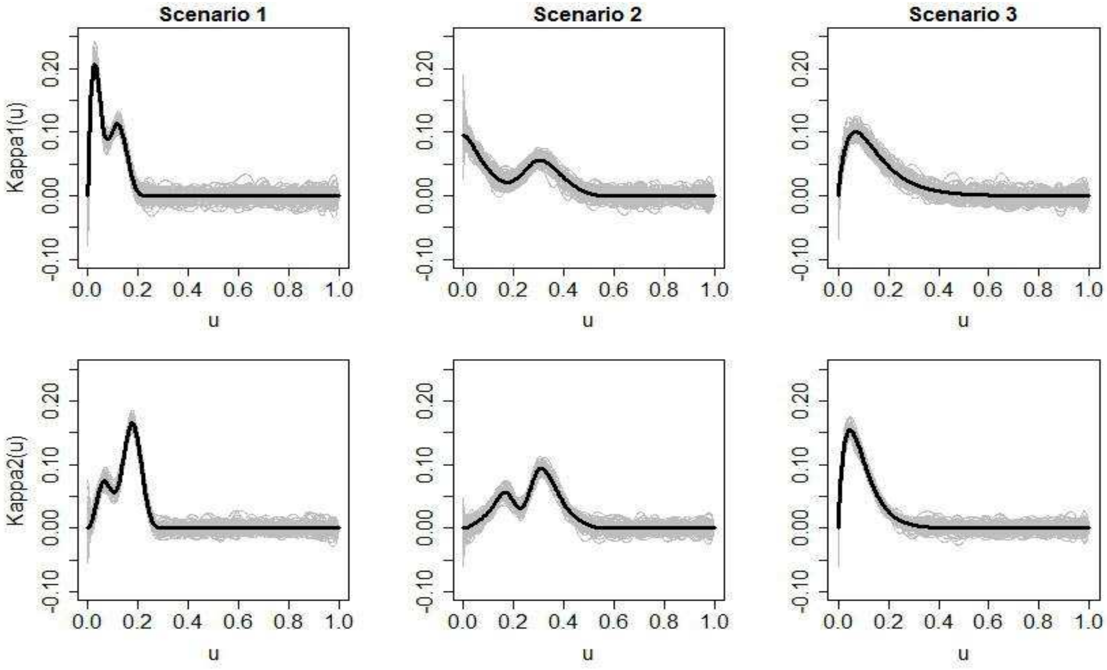

Figure 2.7: Estimated coefficient functions and the true coefficient functions for model 1. The black curves are the true coefficient functions and the grey curves are the estimated coefficient functions.

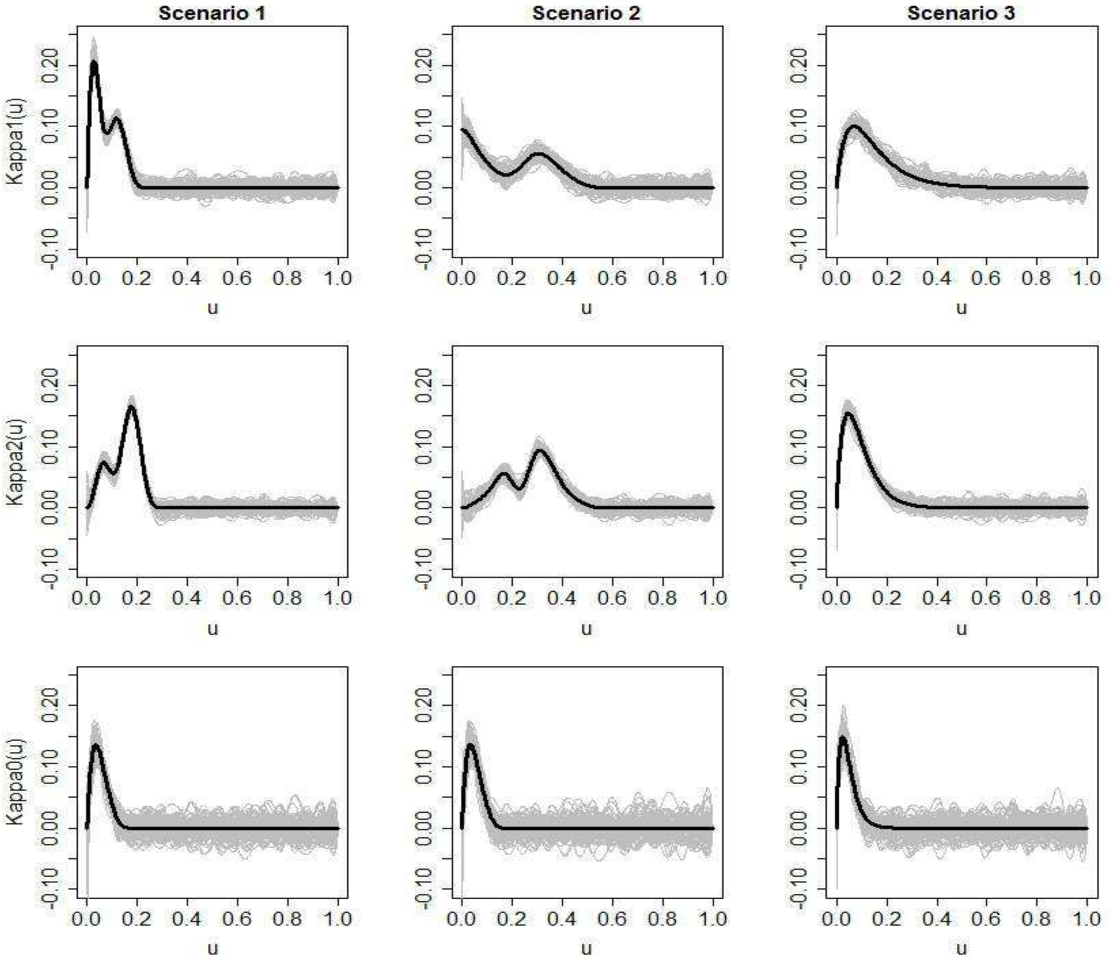

Figure 2.8: Estimated coefficient functions and the true coefficient functions for model 2. The black curves are the true coefficient functions and the grey curves are the estimated coefficient functions.

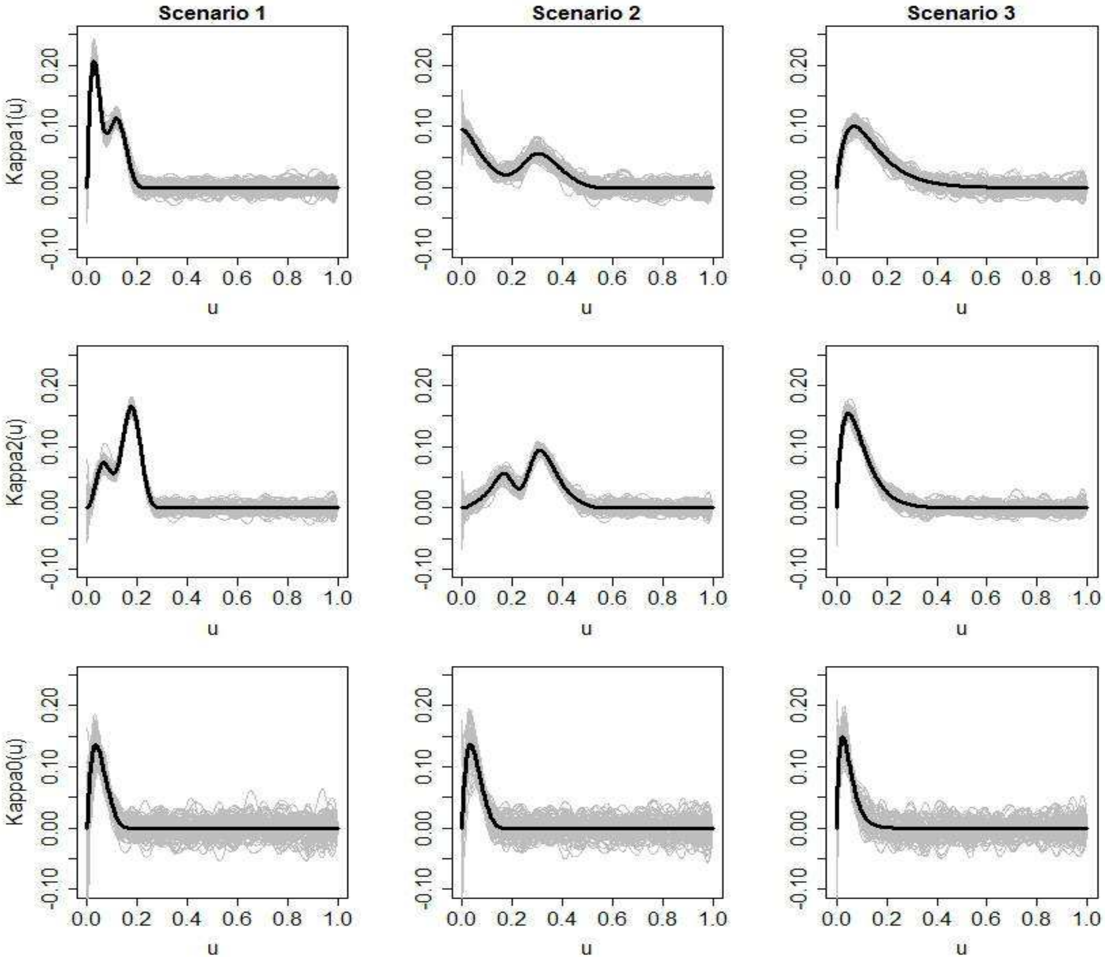

Figure 2.9: Estimated coefficient functions and the true coefficient functions for model 3. The black curves are the true coefficient functions and the grey curves are the estimated coefficient functions.

Denote the estimated coefficient function of the kth repetition as $\widehat{\kappa}_i^j(u)$. Based on the 100 repetitions, the bias function of the estimated coefficient function is computed as

$$Bias_i(u) = \frac{1}{100}\sum_{j=1}^{100}\widehat{\kappa}_i^j(u) - \kappa_i(u) \tag{2.28}$$

We use the integrated absolute bias as a numerical measure of the overall bias of a coefficient function. The integrated absolute bias is computed as

$$IBias_i = \int_0^1 Bias_i(u)du. \tag{2.29}$$

The bias functions are shown in Figures 2.10, 2.11 and 2.12. The IBias values are shown in Table 2.4. The bias values are quite small for the two predictors and larger for the response history component. In addition, in order to assess the usefulness of the asymptotic standard error of the maximum partial likelihood estimator, both asymptotic and empirical standard errors based on 100 repetitions are calculated. The empirical standard error function of the estimated coefficient function is computed as

$$SD_i(u) = \sqrt{\frac{1}{99}\sum_{j=1}^{100}\left(\widehat{\kappa}_i^j(u) - \overline{\widehat{\kappa}_i(u)}\right)^2} \tag{2.30}$$

For each replication, an estimated variance-covariance matrix $\widehat{V}_i^j$ is computed for estimated B-spline coefficients. $\widehat{\beta}_i = \{\widehat{\beta}_{i1}, ...\widehat{\beta}_{i19}\}^{\mathrm{T}}$. Then the asymptotic standard error function of the estimated coefficient function is computed as:

$$\widehat{SD}_i^j(u) = B(u)^{\mathrm{T}}\widehat{V}_i^j B(u) \tag{2.31}$$

where $B(u) = (B_1(u), ...B_{19}(u))^{\mathrm{T}}$. The asymptotic standard error functions are compared with the empirical standard error function. Figure 2.13 shows the standard error functions for model 3 in Scenario 3 as an example. The plot indicates the estimated standard error functions are very close to the empirical standard error function. Therefore, the standard variance estimator can give valid variance estimates for the coefficient functions.

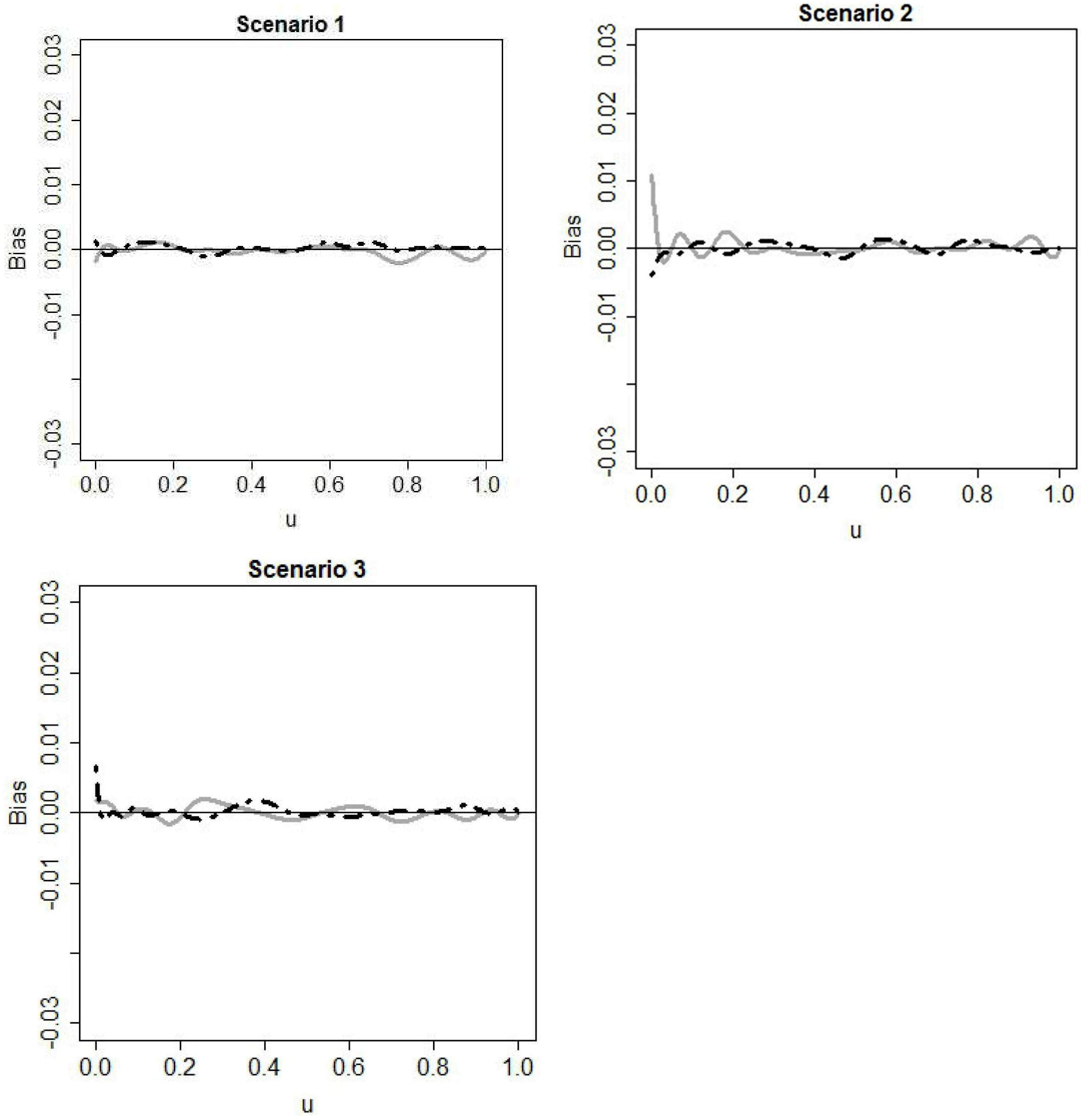

Figure 2.10: Plots of the bias functions for model 1. The gray curves are for predictor 1 and the dot dashed curves are for predictor 2.

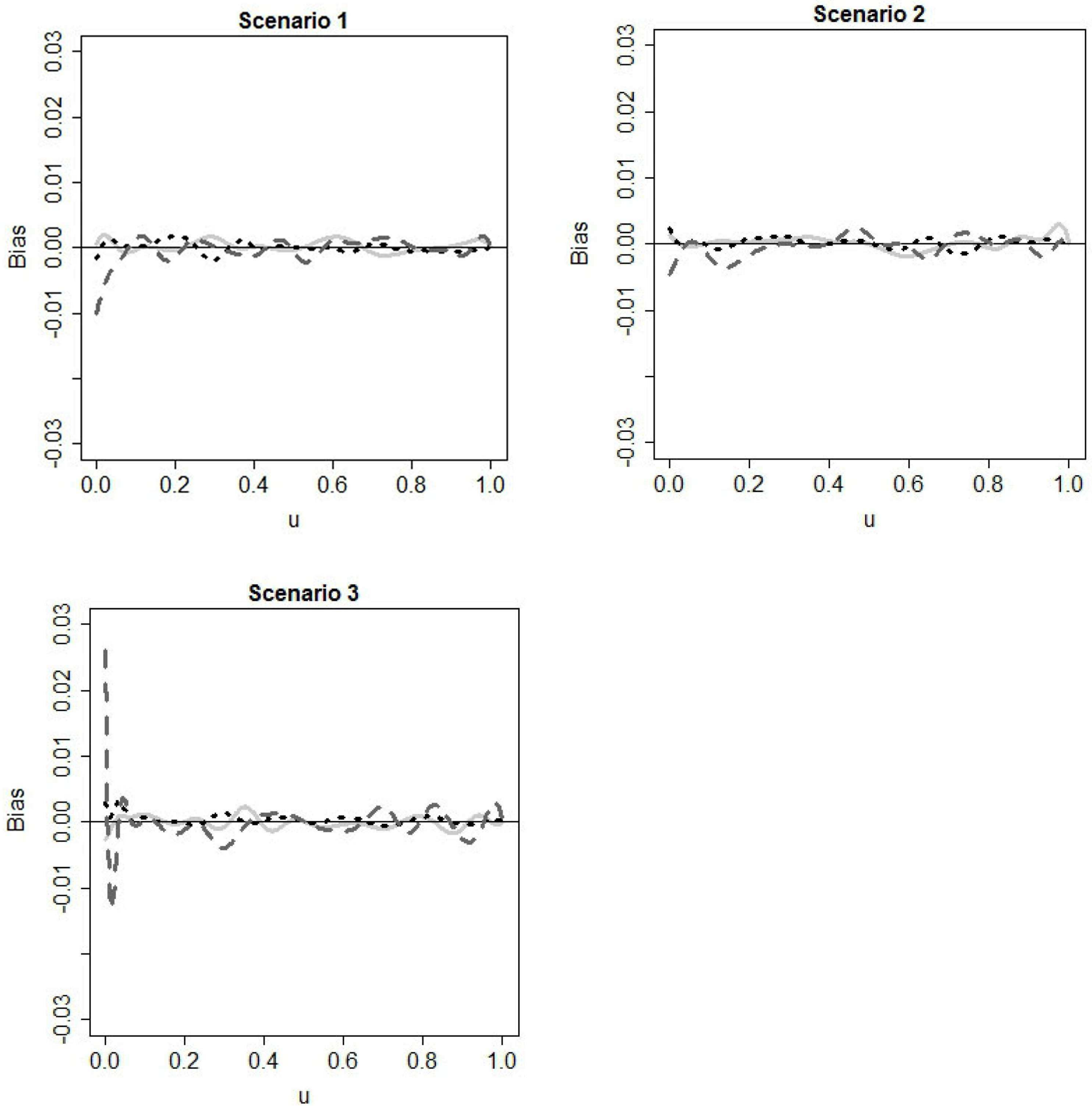

Figure 2.11: Plots of the bias functions for model 2. The gray solid curves are for predictor 1, the dotted black curves are for predictor 2 and the dark gray dashed curves are for the response's history.

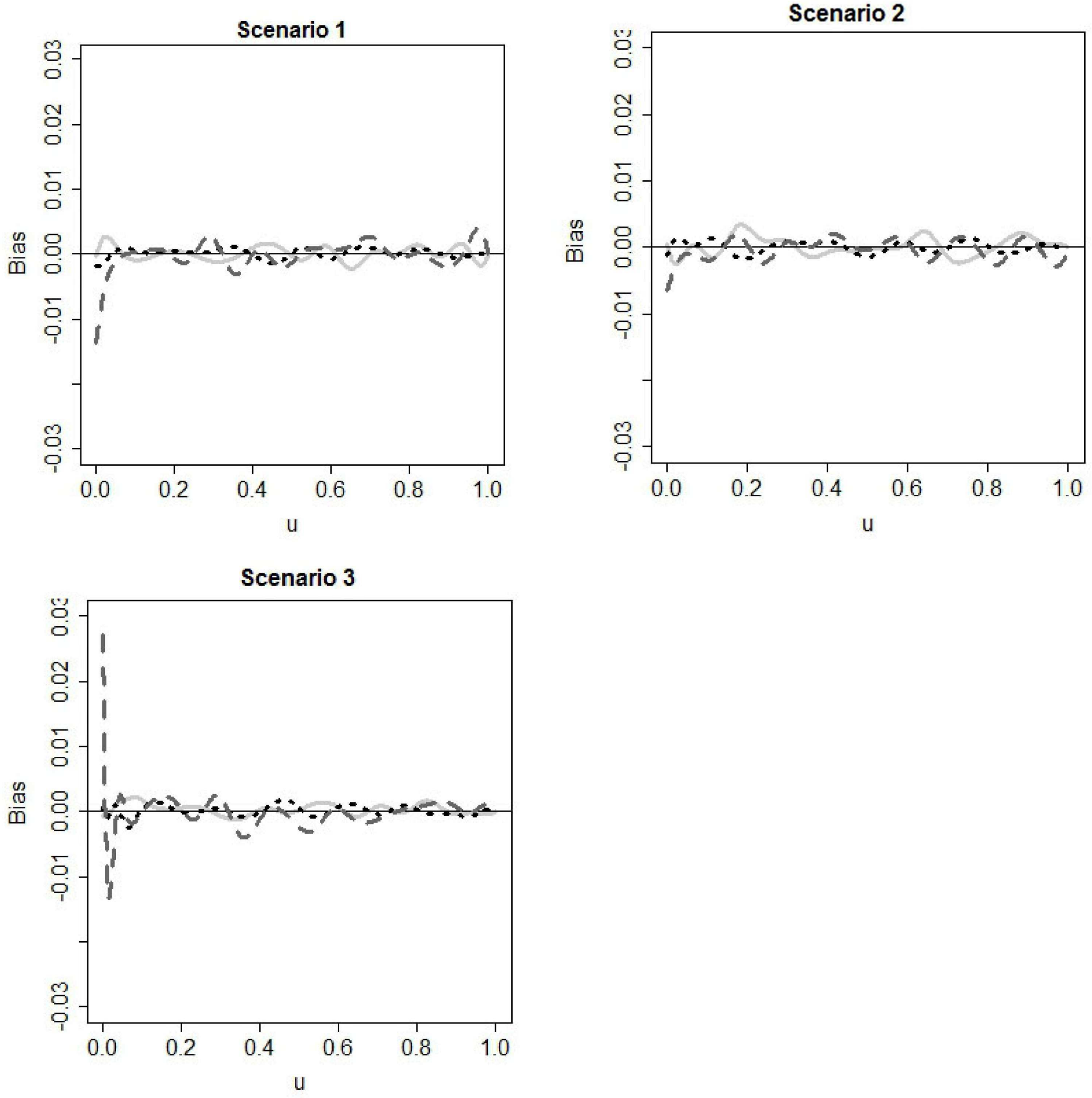

Figure 2.12: Plots of the bias functions for model 3. The gray solid curves are for predictor 1, the dotted black curves are for predictor 2 and the dark gray dashed curves are for the response's history.

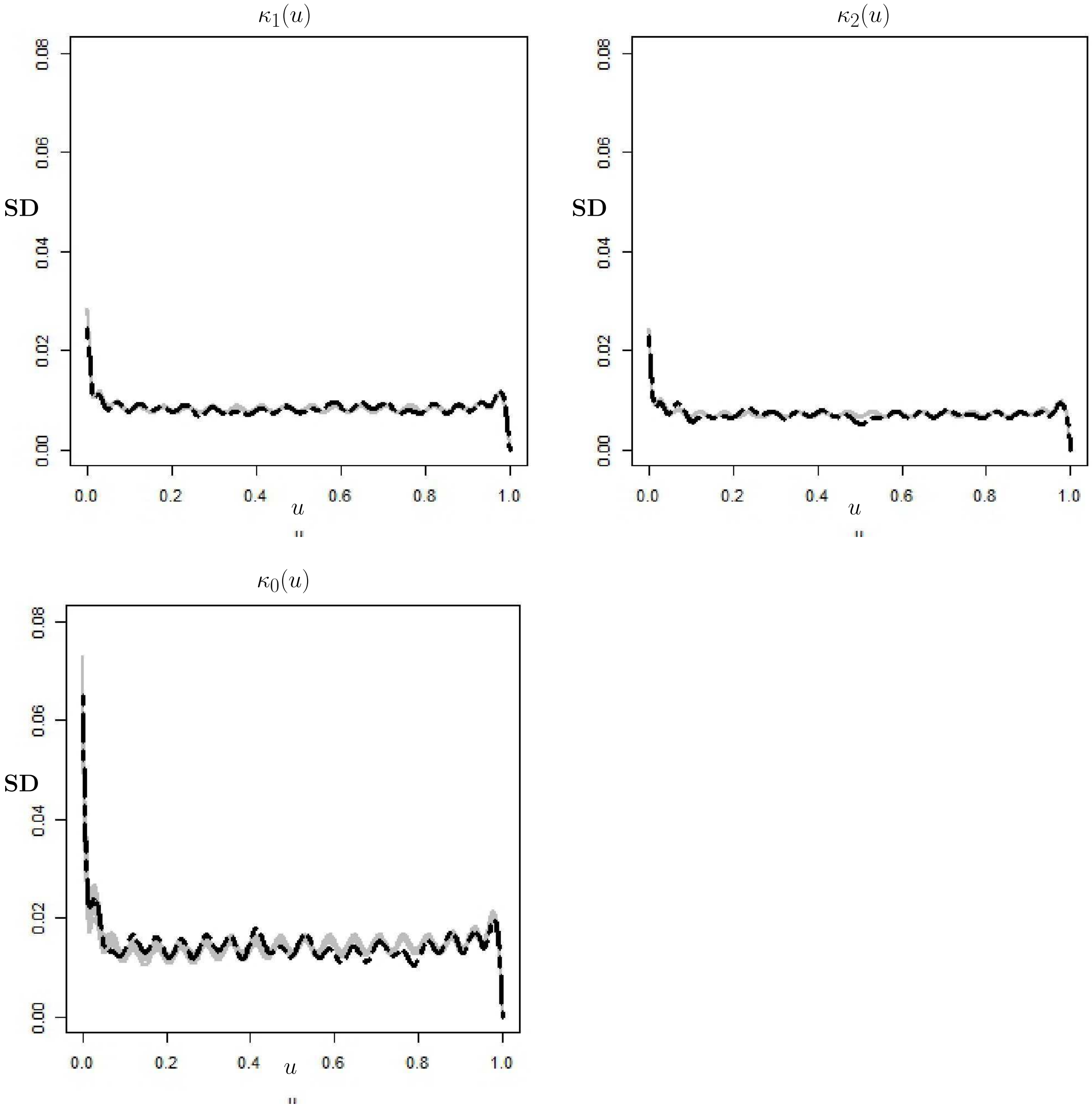

Figure 2.13: Asymptotic standard error functions and empirical standard error functions for model 3 Scenario 3. The black dashed curves are the asymptotic standard error functions and the grey curves are the empirical standard error functions.

Table 2.4: Integrated absolute bias

	Model 1	Model 2	Model 3
Scenario 1			
Predictor 1	0.266	0.316	0.410
Predictor 2	0.248	0.272	0.283
Response	NA	0.583	0.697
Scenario 2			
Predictor 1	0.377	0.347	0.552
Predictor 2	0.348	0.291	0.353
Response	NA	0.521	0.625
Scenario 3			
Predictor 1	0.357	0.345	0.339
Predictor 2	0.260	0.253	0.313
Response	NA	0.814	0.750

In summary, when the sample size is large, the estimation procedure for the multiplicative modulated renewal process model provides good estimates of the coefficient functions. The asymptotic standard error of the maximum partial likelihood estimator is useful as an inference method.

2.6 Real data analysis: large data estimation and non-stationarity

2.6.1 Application to a large neural spike train data

In this section, the proposed model is implemented on the neuron signal data described in Section 1.2. There are six spike trains that record the times of the neuron spikes over 500 seconds from layer CA1 of the right dorsal hippocampus. We call this data set the Real Data 1. Spike train S_6 is chosen as the response and the other 5 spike trains are treated as functional predictors. The modulated renewal process model (2.7) is used to fit the data. This model includes the predictor trains' histories, the response train's history and an additional covariate for the movement speed of the subject. The spike history is updated every 0.004 second, which seems to be a reasonably small value relative to the minimum inter-spike time 0.002 second. The length of the history M is chosen to be 1 second. B-spline basis functions with $d = 3$ and 14 interior knots are used. The knots sequence is: 0, 0.005, 0.01, 0.015, 0.020, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1 ,and the B-spline basis function closest to the 1 end is not used so that the coefficient functions are 0 at $t - 1$ second.

The coefficient functions are estimated by the partial likelihood estimation procedure described in Section 2.3.3. The result is shown in Figure 2.15. Figure 2.15 includes six plots; the first five plots show the estimated coefficient functions of the five predictors, and the last plot shows the coefficient function of the response's own history. Functions $\kappa_1, \kappa_2, \kappa_3$ show a strong positive component for short intervals (< 20 ms). κ_4 is not significantly different from 0 based on the estimated standard errors. κ_5 shows a positive component for short intervals (approximately 0-40 ms), followed by a negative component for longer intervals (approximately 40-100 ms). κ_0, the coefficient function of response's history, is not significantly different from 0 for short intervals (approximately 0-50 ms). However, it shows a strong positive component for longer intervals (approximately 50-200 ms). Figure 2.14 shows the estimated cumulative baseline function ($\widehat{CH_0}(y)$) and the smoothed baseline function ($\hat{\lambda}_0(y)$), which is computed by numerically taking derivative of the estimated cumulative baseline function. $\hat{\lambda}_0(y)$ is small near 0, so the neuron tends to not fire again immediately after a firing (i.e., there is a resting period). $\hat{\lambda}_0(y)$ increases rapidly to its maximum at approximately 15 ms then decrease slowly. Note that $\hat{\lambda}_0(y)$ describes the unique effect of the very last response spike, in additional to the effect of any response spike in the history. The moving speed of the subject is included in the model as a time-dependent predictor. The coefficient estimate is 0.1632 (p-value=0.0056), which implies the subject's moving speed has a positive effect on the response neuron's spike activities.

Figure 2.16 shows that the in-sample KS plot for the model, which is computed using the same data that have been used for model fitting, is within the 95% confidence bounds. It also shows five out-of-sample KS plots computed using data over five randomly select time intervals after the time of the data for model fitting. From Figure 2.16, all the five out-of-sample KS plots are within the 95% confidence bounds. The KS plots show that the model performs well in both the case of model fitting and the case of prediction.

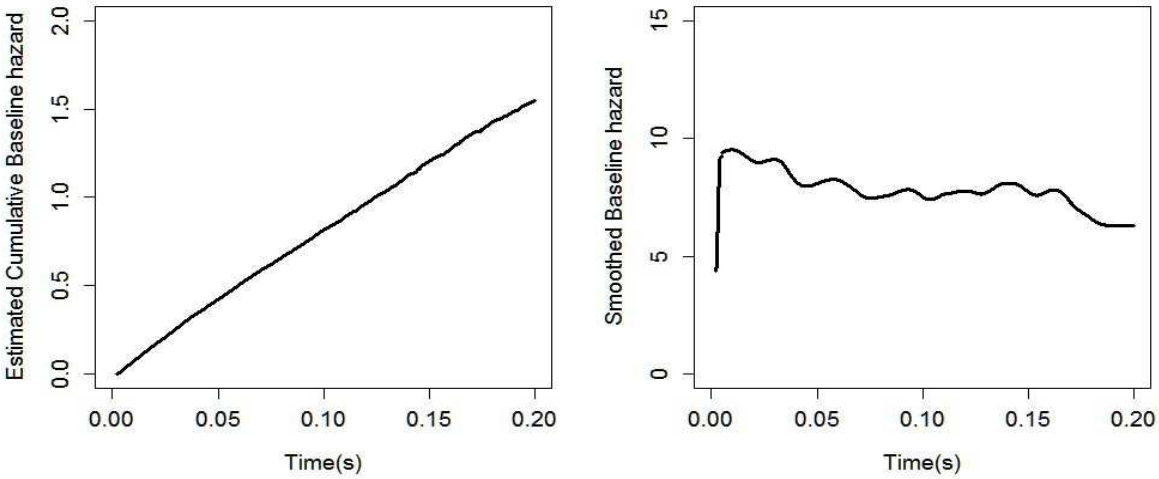

Figure 2.14: Plots of estimated cumulative baseline hazard functions and the smoothed estimated baseline hazard functions based on Real Data 1.

Several reduced models were also fitted. We use BIC as a criterion for comparing the models. Table 2.5 shows the BIC values. The model with both predictors' histories and

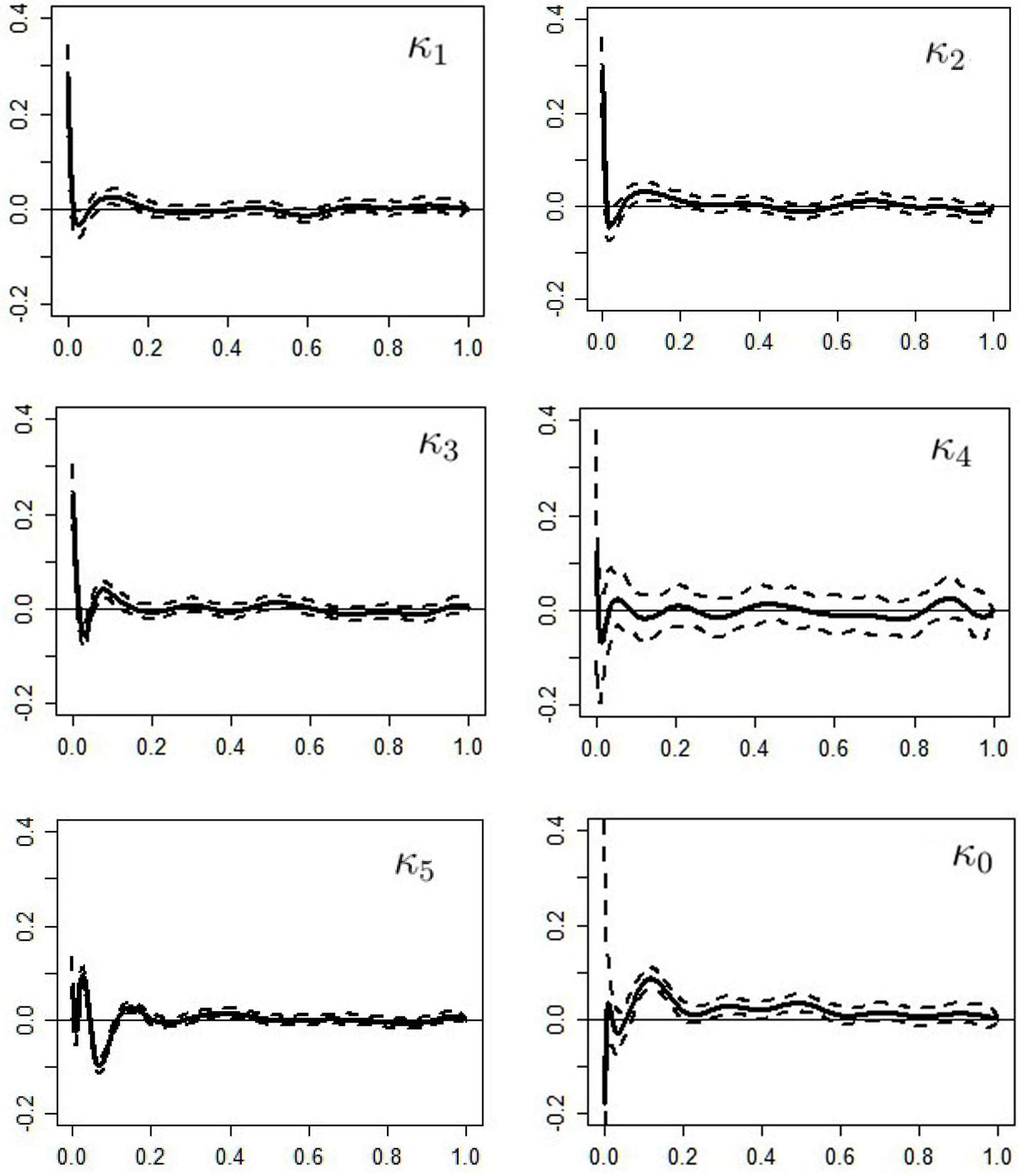

Figure 2.15: Plots of estimated coefficient functions based on Real Data 1. The dashed curves show the estimated coefficient functions +/- 2 times the estimated standard error functions.

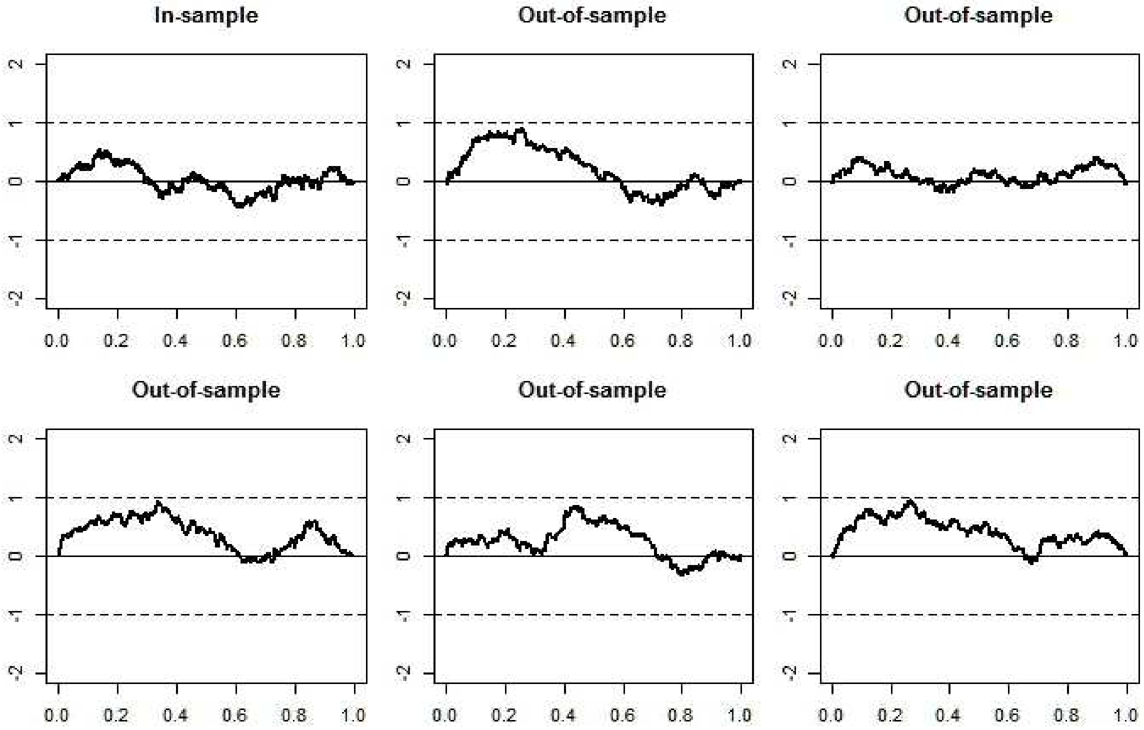

Figure 2.16: Model goodness-of-fit shown with horizontal KS plots of model 1 based on Real Data 1. The left top plot is the in-sample KS plot. The remaining five plots are out-of-sample KS plot using five different data sets, each of which consists 1000 spikes. Dashed black lines represent the 95% confidence bounds.

response's history has the lowest BIC (Model 7, BIC=129138.4). When moving speed is also included as a predictor (Model 1), the BIC is slightly larger (129139.8). This implies that including moving speed does not decrease the partial likelihood by a large amount given, the presence of the neural activities in the model. Note that all models without the predictor trains' histories have much larger BICs. This implies that the predictor trains play an important role in explaining the occurrences of the response spikes. The BIC is further reduced by including the response train's own history. So the future response spike times are related to response train's own history.

Table 2.5: BIC values of models with different predictors.

Model	Model Predictors	BIC
1	predictor trains' and response's histories, speed predictor	**129,139.8**
2	only speed predictor	129,655.2
3	only response's history	129,471.0
4	response's history and speed predictor	129,459.8
5	only predictor trains' histories	129,172.9
6	the predictor trains' histories and the speed predictor	129,164.8
7	the predictor trains' and response's histories	**129,138.4**

2.6.2 Evaluation of non-stationarity

The dataset used in the previous section is seconds 0 through 500 of a larger dataset that records the times of the neuron spikes of six spike trains over 4500 seconds. The 1000-1100s sub-dataset is excluded in the following data analysis, because of missing speed values. To investigate whether the relationship between the response and the predictor trains is stationary, we fit the model using the entire dataset and evaluate goodness-of-fit using 9 sub-datasets based on time, 0-500s, 500-1000s, 1100-1500s, 1500-2000s, 2000-2500s, 2500-3000s, 3000-3500s,3500-4000s, 4000-4500s.

Figure 2.17 shows the nine horizontal KS plots for the nine sub-datasets. It can be seen that there is a pattern over time, the horizontal KS plots go from negative to positive as time increases. A negative KS curve implies that the observed inter-spike time tends to be smaller than the expected inter-spike time based on the fitted model. A positive KS curve implies that the observed inter-spike time tends to be bigger than the expected inter-spike time based on the fitted model. The trend of the KS plots can be better seen when we plot the extreme values and medians of the KS plots vs time (Figure 2.18). This trend may imply that the relationship between the response train and the predictor trains is not stationary, i.e., the coefficient functions changes over time, because the goodness-of-fit performance of the model based on the entire dataset does not change randomly for sub-datasets based on time.

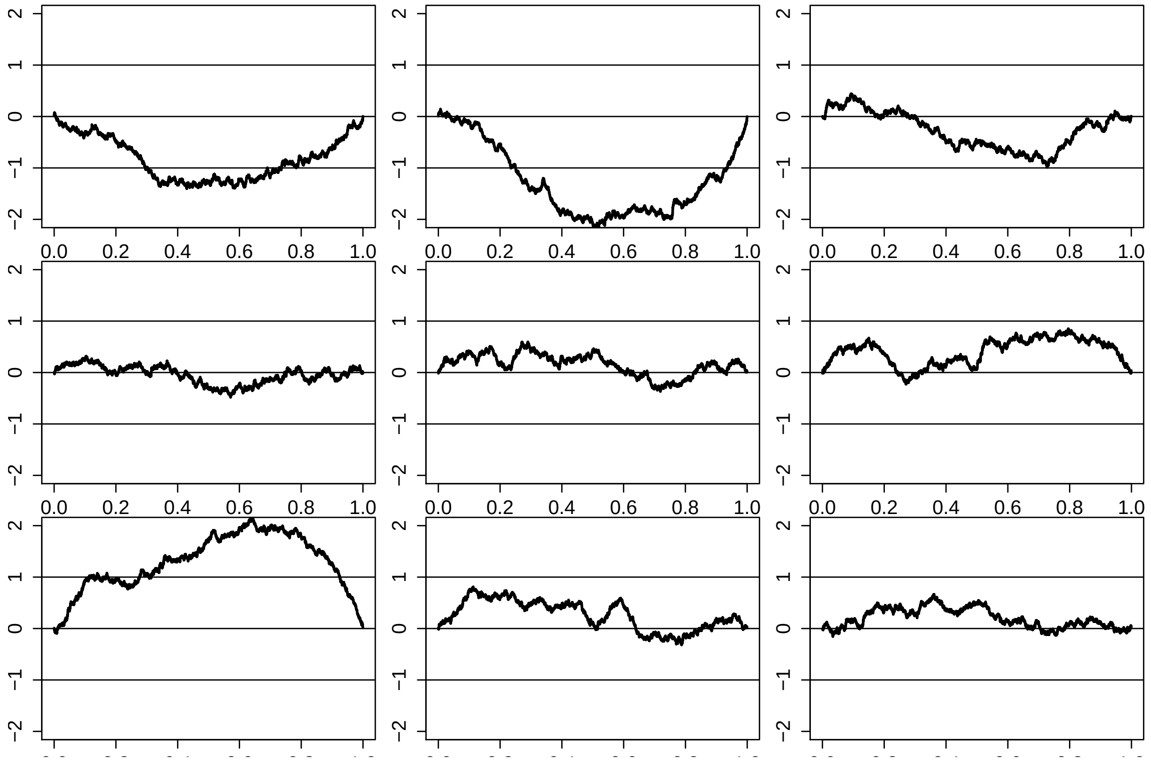

Figure 2.17: How well the model based on the entire dataset fits the nine sub-datasets, evaluated by horizontal KS plots.

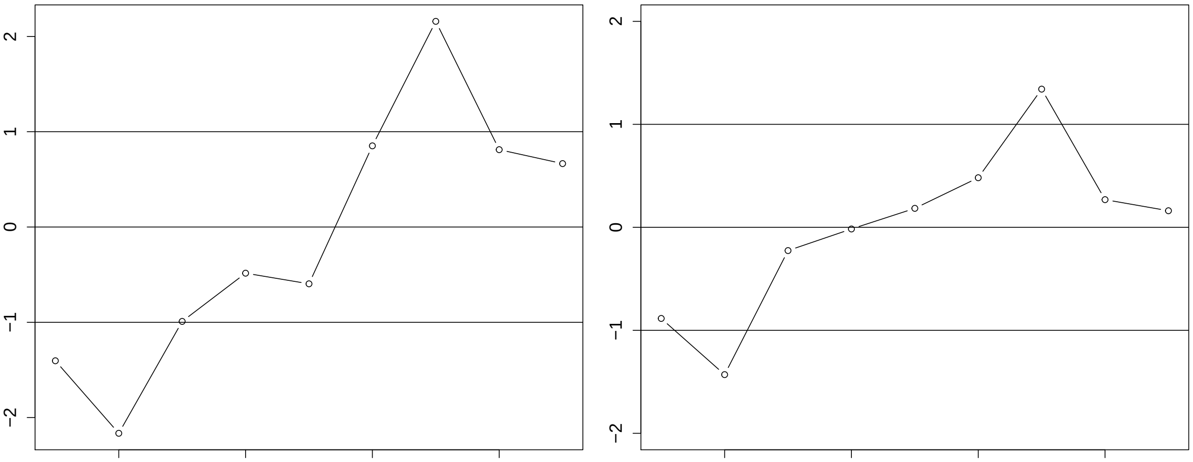

Figure 2.18: Plots of extreme values and medians of the nine horizontal KS plots.

2.7 Discussion

In this Chapter, we formulate a modulated renewal process model for the identification of the neural connectivity using spike train data. This modulated renewal process model has two new features compared to the standard Cox's proportional hazards model. (1) The impact strengths of the spike histories of the predictor trains are modeled by coefficient functions that can be approximated by B-splines to reduce model complexity. (2) The conditional intensity function of the response spike train depends not only on the histories of the predictor trains and the extrinsic covariates but also depends on its own history, i.e., the response spike train is modeled by a modulated renewal process instead of a renewal process.

There are three major advantages of this modulated renewal process model. First, this model does not require discretizing the response spike process by recording response as presence or absence of a spike in 0.002 time interval, which may cause information loss. Second, this model does not require dividing the history interval into subintervals to create covariates based on the subintervals. Moreover, modeling the effect of the history by very flexible coefficient function improves the ability to capture the dependence of the future spikes on the previous spikes. Third, modeling the response spike process by modulated renewal process provides more flexibility to model the distribution of the inter-spike times than a Poisson process.

By simulation, we show that the model estimation procedure based on the partial likelihood has good performance when the sample size is large. The asymptotic estimates of the standard errors from the model are close to the standard error estimated by simulation. They can be used to provide narrow confidence bands for the coefficient functions, which can be used to identify important predictor trains and history intervals. However, in the real data analysis, we see that the relationship between the response train and the predictor trains may not be stationary, i.e., the coefficient functions may vary over the real time (the time from the start of the observation). It may not be appropriate to fit the model using data over a very long time period ignoring the non-stationarity. Since little is known about the structure of the non-stationarity, it is sensible to consider the local model estimation based on observations in a short period in which the relationships among spike trains can be assumed to be stationary. Then, the sample size becomes smaller.

Due to non-stationarity, we need to model shorter segments of time, so we must have efficient estimator of the coefficient functions; B-spline gives us more efficient use of parameters than piecewise constant functions. In Chapter 3, we look at model selection methods to increase the efficiency of parameter usage.

CHAPTER 3

Variable selection in modulated renewal process model with functional predictors

3.1 Sparsity in neural connectivity

In the proposed modulated renewal process model, most of the predictors are functional, i.e., they are observed continuously along certain trajectory such as time. The impact strength of a functional predictor is modeled by a coefficient function instead of a scalar coefficient. A scalar coefficient is either zero or nonzero, but a coefficient function can be zero in some intervals and nonzero in the others. This brings potential difficulty in selecting important functional predictor(s).

In neural network, the relationship between neurons is believed to be sparse in nature [Song et al., 2007]. That is, in our proposed modulated renewal process model, many coefficient functions are constant zero or zero in many intervals. Therefore, the estimated coefficient functions should be sparse in order to capture the true relationship between the neurons. Tu et al. [2012] defined two types of sparsities to distinguish between two cases (1) a coefficient function is zero in the whole interval, (2) a coefficient function is zero only in some subintervals. They called the former global sparsity and the latter local sparsity.

Ideally, we would like to achieve both global and local sparsities of the estimates simultaneously. However, in some situations this goal can be very difficult to achieve. One such situation is when the true coefficient functions are only nonzero in subintervals that are short compared with the whole interval; then the estimated coefficient functions are highly likely to be identified as being zero in the whole interval by enforcing the global sparsity. Therefore, in such situations, one has to consider preference between the two types of sparsities. In this study, the number of predictor spike trains are reasonably small, but for each train, we consider a long period of history to capture any possible history effects. Therefore, we expect the durations of the true effects are much shorter than the whole period considered and we prefer more accurately achieving local sparsity to enforcing global sparsity.

3.2 Penalized partial likelihood estimation

Recall that after applying the B-spline approximation, the proposed modulated renewal process model can be estimated by maximizing the following Cox's partial likelihood.

$$PL(\boldsymbol{\beta}) = \prod_{j=1}^{J} \frac{\Psi^*(t_{j-1}+y_j, \boldsymbol{\beta})}{\sum_{l \in R(t_j)} \Psi^*(t_{l-1}+y_j, \boldsymbol{\beta})} \tag{3.1}$$

where $\Psi^*(t, \boldsymbol{\beta}) = \exp\left\{\sum_{i=0}^{p}\sum_k \beta_{ik}^A D_{ik}(t) + \boldsymbol{\beta}^B \boldsymbol{Z}(t)\right\}$ reflects the overall effect of the covariates on the conditional intensity at time t. It is a function of $\boldsymbol{\beta}$, a vector of parameters including the coefficients of the B-spline approximation and the coefficients of the nonfunctional predictor(s). $R(t_j) = \{i, y_i \geq y_j\}$ is the risk set of response spike time t_j, which is defined using the inter-spike times $y_i = t_i - t_{i-1}$. Maximizing (3.1) is equivalent to maximizing the following log partial likelihood,

$$l(\boldsymbol{\beta}) = \sum_{j=1}^{J}\left\{\sum_{i=0}^{p}\sum_{k} \beta_{ik}^A D_{ik}(t_{j-1}+y_j) + \boldsymbol{\beta}^B \boldsymbol{Z}(t_{j-1}+y_j)\right\} \tag{3.2}$$

$$- \sum_{j=1}^{J} \log\left\{\sum_{l \in R(t_j)} \exp\left\{\sum_{i=0}^{p}\sum_{k} \beta_{ik}^A D_{ik}(t_{l-1}+y_j) + \boldsymbol{\beta}^B \boldsymbol{Z}(t_{l-1}+y_j)\right\}\right\} \tag{3.3}$$

Like the maximum likelihood estimator, the maximum partial likelihood estimator does not guarantee sparsity. One way to enforce sparsity is the penalized likelihood approach, which is very popular because it simultaneously considers model estimation and variable selection. In general, the penalized log-partial likelihood criterion can be expressed as

$$\sum_{j=1}^{J} l_j(\boldsymbol{\beta}) - J\sum_{i=1}^{r} p_\lambda(|\beta_i|) \tag{3.4}$$

where l_j is the log partial likelihood of the jth observation, r is the dimension of $\boldsymbol{\beta}$, $p_\lambda(\cdot)$ is a penalty function for each parameter and λ is a tuning parameter. We consider two widely used penalty functions, LASSO and Smoothly Clipped Absolute Deviation Penalty.

The LASSO penalty is defined as

$$p_\lambda(|\beta|) = \lambda|\beta|. \tag{3.5}$$

Thus, the LASSO penalized log-partial likelihood criterion can be expressed as

$$\sum_{j=1}^{J} l_j(\boldsymbol{\beta}) - J\lambda \sum_{i=1}^{r} |\beta_i| = \sum_{j=1}^{J} l_j(\boldsymbol{\beta}) - J\lambda\|\boldsymbol{\beta}\|_1 \tag{3.6}$$

The penalty constrains the L_1 norm of the coefficient vector. Maximizing the LASSO penalized log-partial likelihood is equivalent to maximizing the log-partial likelihood subject to $\|\boldsymbol{\beta}\|_1 < s$, then finding the global maximum of the penalized partial likelihood over the values of s. A few algorithms have been developed to solve the LASSO optimization problem. We choose the algorithm proposed by Goeman [2010], which is based on a combination of gradient ascent optimization with the Newton-Raphson algorithm.

The SCAD penalty is defined as

$$p'_\lambda(|\beta|) = \lambda \left\{ I(|\beta| \leq \lambda) + \frac{(a\lambda - |\beta|)_+}{(a-1)\lambda} I(|\beta| > \lambda) \right\} \qquad \text{for some } a > 2. \tag{3.7}$$

It involves two tuning parameters, a and λ. The SCAD penalty is equivalent to the LASSO penalty when θ is small, but is constant when $|\beta|$ is large. This guarantees that large estimated coefficients are not excessively penalized. The tuning parameter λ controls the range of β values in which the SCAD penalty is equivalent to the LASSO penalty and the other tuning parameter a controls the range of β values in which the SCAD penalty is constant. Fan and Li [2001] suggested using $a = 3.7$ from Bayesian statistical point of view, and this value will be used throughout this paper. Maximizing the SCAD penalized partial likelihood function is difficult because the penalty is nondifferentiable at the origin and nonconcave with respect to $\boldsymbol{\beta}$. By locally approximating the penalty function with a linear function [Zou and Li, 2008], the SCAD optimization problem can be turned into a sequence of LASSO optimization problems as follows. Use the unpenalized maximum partial likelihood estimate as the initial value, $\boldsymbol{\beta}^{(0)}$. For $w = 0, 1, 2, ...$, repeatedly solve

$$\boldsymbol{\beta}^{(w+1)} = \arg\max \left\{ \sum_{j=1}^{J} l_j(\boldsymbol{\beta}) - J \sum_{i=1}^{r} p'_\lambda(|\beta_i^{(w)}|)|\beta_i| \right\}. \tag{3.8}$$

Stop the iterations if the sequence of $\boldsymbol{\beta}^{(w)}$ converges (Zou and Li [2008]). The performance of the penalized approach depends on the tuning parameter λ, which balances the trade-off between goodness-of-fit and model sparsity. We use the BIC to select λ. For a sequence of λ candidates, fit the model by maximizing the penalized partial likelihood, then compute BIC for each model estimate and choose the λ value that produces the smallest BIC.

We choose LASSO and SCAD penalties instead of the group penalties, which enforce global sparsity by shrinking parameters in the same group together, because we prefer more accurately achieving local sparsity to enforcing global sparsity. Also these two penalties yield more stable estimates than the group penalties.

3.3 Computation: number of B-spline knots and time-dependent covariates updating frequency

One drawback of the maximum penalized partial likelihood methods is the high computational complexity. The proposed modulated renewal process model is more computational intensive than the standard Cox proportional hazards model because of the B-spline approximation and the time-dependent covariates. Therefore, it is important to choose the appropriate number of B-spline knots and the appropriate way of updating the time-dependent covariates in order to balance the model performance and the computational complexity.

Considering the situation when all the predictors in the model are functional, then the number of parameters in the partial likelihood function (2.25) is equal to the product of the number of functional predictors and the number of the B-spline basis functions in each functional predictor. Therefore, using fewer B-spline basis functions can greatly reduce the computational complexity. However, if the number of the B-spline basis functions is too small, the ability to accurately estimate the coefficient functions and to achieve functional sparsity will be lost. We investigate the performance of the model based on various B-spline basis functions by simulation in section 3.4 and compare the unpenalized partial likelihood approach with the penalized partial likelihood approaches. We find that the estimates by penalized partial likelihood approaches are not sensitive to the number of B-spline basis functions, i.e. the estimates do not change much when the number of knots in the B-spline basis functions changes within a reasonable range. However, the estimates by the unpenalized partial likelihood method changes significantly as the number of knots changes.

When the covariates are time-dependent, to compute the term of the Cox's partial likelihood associated with the jth event, it is necessary to compute, for all the events in the risk set of the jth event,

$$\Psi^*(t_{l-1}+y_j,\boldsymbol{\beta}) = \exp\left\{\sum_{i=0}^{p}\sum_{k}\beta_{ik}^{A}D_{ik}(t_{l-1}+y_j)+\boldsymbol{\beta}^{B}\boldsymbol{Z}(t_{l-1}+y_j)\right\} \tag{3.9}$$

$$= \exp\left\{\boldsymbol{\beta}^{T}\boldsymbol{D}(t_{l-1}+y_j)\right\}, \tag{3.10}$$

where $\boldsymbol{\beta} = (\beta_{ik}^{A}, i = 0,1,,,,p, k = 1,2,...K; \boldsymbol{\beta}^{B})$ is a vector of coefficients of length d and $\boldsymbol{D}(t) = (D_{ik}(t), i = 0,1,,,,p, k = 1,2,...K; \boldsymbol{Z}(t))$ is a vector of time-dependent covariates of length d. To compute $\Psi^*(t_{l-1}+y_j,\boldsymbol{\beta})$, we need d terms and to compute the jth component of the Cox's partial likelihood, for each event in the risk set, we need to compute $\Psi^*(t_{l-1}+y_j,\boldsymbol{\beta})$. The number of events in the risk set of the jth ordered event based on the ascending order of the event time is $J-j+1, j = 1,2,...J$. Note that for $j_1 \neq j_2$, $\boldsymbol{D}(t_{l-1}+y_{j_1})$ may not be equal to $\boldsymbol{D}(t_{l-1}+y_{j_2})$. Therefore, the total number of terms required for computing the Cox's partial likelihood is $d\sum_{j=1}^{J}(J-j+1) = dJ(J+1)/2$. When J is large, this number is large.

An alternative approach is to approximate the time-dependent $\boldsymbol{D}(t)$ with step functions. For example, in Figure 3.1, plot (a) shows that for the response spike with the jth smallest inter-spike time, $y_{(j)}$, there are $J-j+1$ events in the risk set, $y_{(j)}, y_{(j+1)}, ..., y_{(J)}$. Let $t_{l^*}, l = 1,2,...,J$ denote the spike time of the response spike with the lth smallest inter-spike time and t_{l^*-1} denote the spike time of the most recent response spike prior t_{l^*}. $J-j+1$ terms, $D(t_{l^*-1}+y_{(j)}), l = j, j+1, ...J$, are needed for computing the partial likelihood. Plot (b) shows that if $\boldsymbol{D}(t)$ is approximated by a step function taking three values over the interval $(y_{(j)}+\min_{l\geq j}t_{l^*-1}, y_{(j)}+\max_{l\geq j}t_{l^*-1})$, then only three terms are needed instead of j terms. For neural spike train data, we take this approach by choosing a frequency of updating $D(t)$, e.g., 250HZ (every 0.004s). This approximation can reduce the number of terms required by the Cox's partial likelihood to less than dJN, where N is the product of the maximum inter-spike time and the updating frequency and N can be much smaller than $(J+1)/2$. However, important information can be lost by using this approximation. Therefore, it must be used with caution. We compare the performance of models based on various updating frequencies by simulation in Section 3.4.4, and find that it is possible to use low updating frequency and still get good coefficient function estimates.

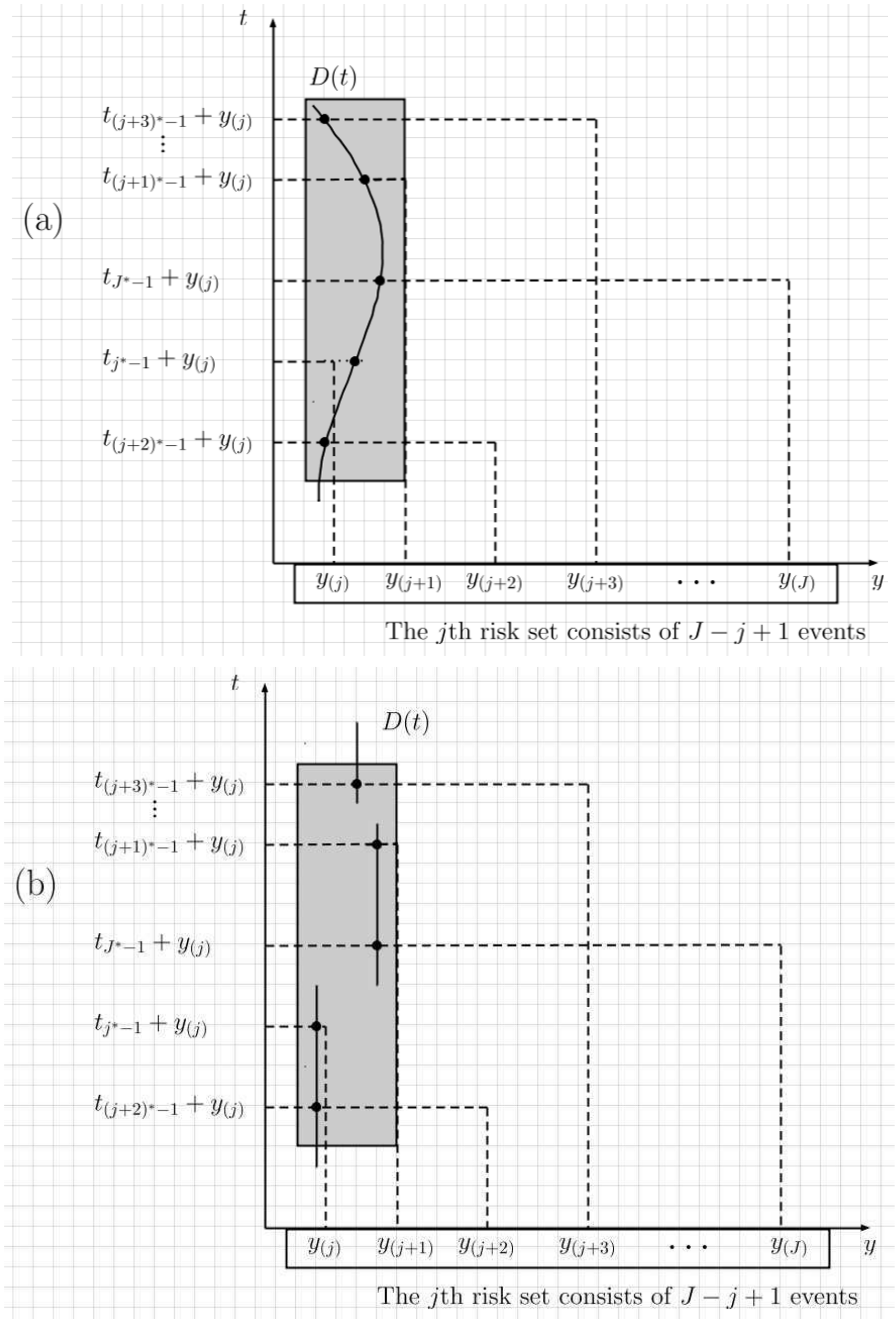
(a)
t
$D(t)$
$t_{(j+3)^*-1} + y_{(j)}$
⋮
$t_{(j+1)^*-1} + y_{(j)}$
$t_{J^*-1} + y_{(j)}$
$t_{j^*-1} + y_{(j)}$
$t_{(j+2)^*-1} + y_{(j)}$
$y_{(j)}$ $y_{(j+1)}$ $y_{(j+2)}$ $y_{(j+3)}$ $\cdots$ $y_{(J)}$
y
The jth risk set consists of $J-j+1$ events
(b)
t
$D(t)$
$t_{(j+3)^*-1} + y_{(j)}$
⋮
$t_{(j+1)^*-1} + y_{(j)}$
$t_{J^*-1} + y_{(j)}$
$t_{j^*-1} + y_{(j)}$
$t_{(j+2)^*-1} + y_{(j)}$
$y_{(j)}$ $y_{(j+1)}$ $y_{(j+2)}$ $y_{(j+3)}$ $\cdots$ $y_{(J)}$
y
The jth risk set consists of $J-j+1$ events

Figure 3.1: Number of terms required for computing the partial likelihood is reduced by updating the time-dependent covariates at fixed frequency.

3.4 Simulation: comparison of PL, LASSO and SCAD

In this section, we use simulation to evaluate the performance of the unpenalized maximum partial likelihood (PL) estimator and the LASSO and SCAD penalized maximum PL estimators for the modulated renewal process model. Four ($p = 4$) neuron spike trains are chosen from the data set described in Section 1.2 as predictor trains. The following modulated renewal process model (3.11) is used to generate the response train,

$$\lambda(t|H_t) = \lambda_0(t - t_{N(t-)}) \exp\left\{ \sum_{i=0}^{p} \int_{t-M}^{t} \kappa_i(t-u) dn_{x_i}(t) \right\}. \tag{3.11}$$

The baseline function $\lambda_0(y)$ is equal to 0 when $y \leq 0.002$, which represents a resting period, and 2.5 when $y > 0.002$. The length of history M is 1 second. To reduce computational complexity, we assume that the history impact changes every 0.004 second instead of changing continuously. The true coefficient functions are in the space spanned by a linear combination of B-spline basis functions with $d = 3$ degrees and $m = 9$ equally spaced interior knots, which generates 13 basis functions. The knots are 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9. For simplicity, we choose the same group of basis functions to represent all the coefficient functions and denote them by $B_1, B_2, ..., B_{13}$. The true values of the B-spline coefficients are shown in Table 3.1. The true coefficient functions are shown in Figure 3.2.

Table 3.1: True coefficient values (β_{ik}) of the four predictor trains and the response.

k	1	2	3	4	5	6	7	8	9	10	11	12	13
κ_1	0.3	0	0	0	0	0	0	0	0	0	0	0	0
κ_2	0	0.4	0	0	0	0	0	0	0	0	0	0	0
κ_3	0	0	0	0	0	0	0	0	0	0	0	0	0
κ_4	0	0	-0.08	-0.2	0.3	0	0	0	0	0	0	0	0
κ_0	0	0	0	0.02	0.2	0.04	0	0	0	0	0	0	0

We conduct the simulation under four scenarios (Table 3.2). To compare the performance of the estimators under different sample sizes, in the first and the second scenarios, 500 response spikes are generated, and in the third and the fourth scenarios, 1000 response spikes are generated. In the first and the third scenarios, the correct B-spline basis functions

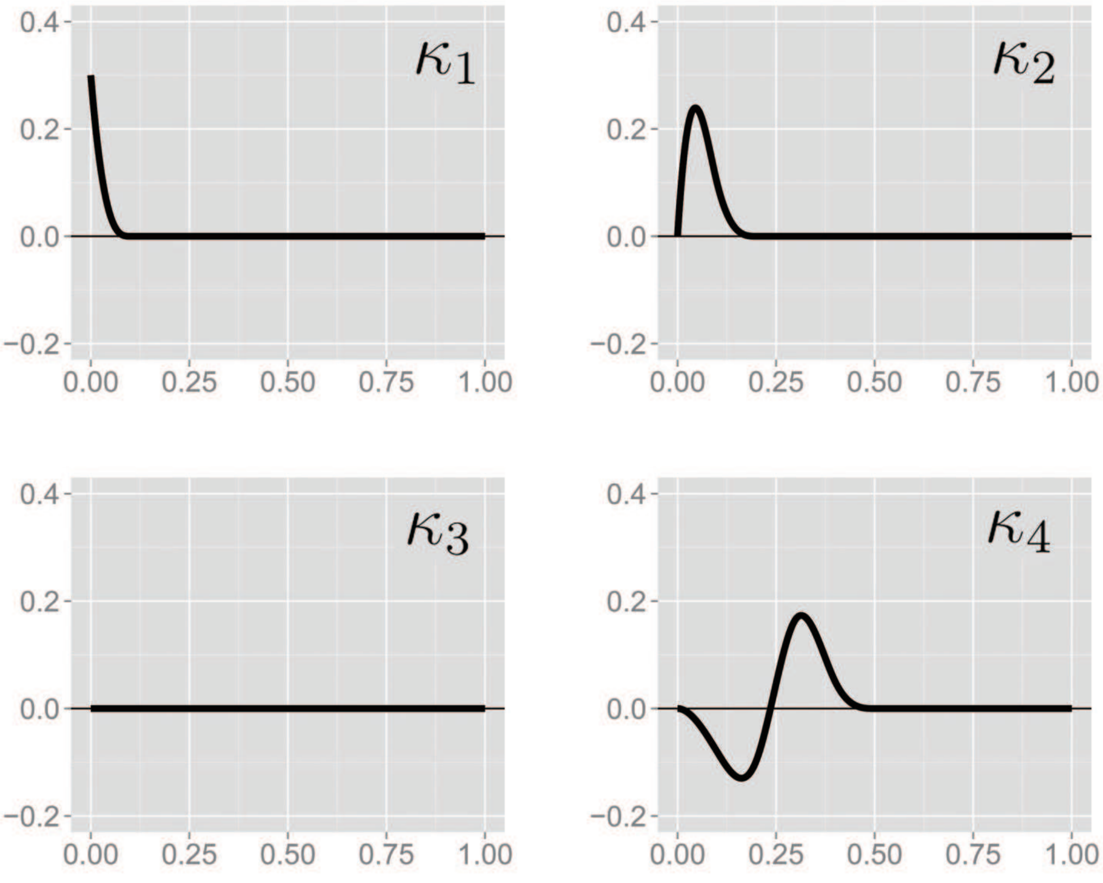

Figure 3.2: True coefficient functions of the predictor trains.

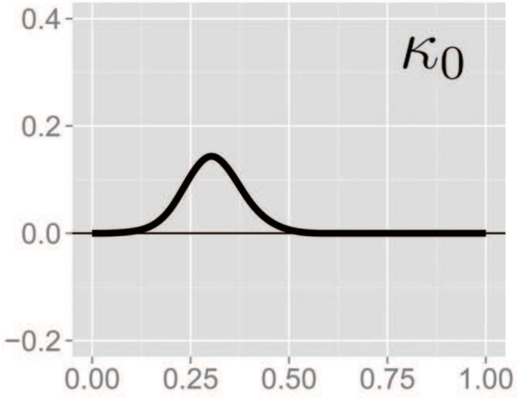

Figure 3.3: True coefficient function of the response history.

are used to fit the model. In the second and the fourth scenarios, B-spine basis functions with correct degree but incorrect number of knots are used. The latter two scenarios are examples of the situation when the true coefficient functions are not in the space spanned by the B-spline basis functions used to fit the model. For each of the four scenarios, 200 data sets are generated. The estimated coefficient functions are compared with the true coefficient functions to evaluate the performance of the estimators.

Table 3.2: Four scenarios

		Sample size	
		500	1000
Number of knots	Correct	Scenario 1	Scenario 3
for B-splines	Incorrect	Scenario 2	Scenario 4

The following measures of performance are used. Denote the ith estimated coefficient function of the jth repetition as $\widehat{\kappa}_i^j(u)$. The bias function for the ith estimated coefficient function is computed as

$$Bias_i(u) = \frac{1}{200}\sum_{j=1}^{200}\widehat{\kappa}_i^j(u) - \kappa_i(u). \tag{3.12}$$

Another numerical measure of performance is the root mean integrated squared error (RMISE). Based on 200 repetitions, RMISE is computed as

$$RMISE_i = \sqrt{MISE_i} = \sqrt{\frac{1}{200}\sum_{j=1}^{200}\int_0^1 \left(\widehat{\kappa}_i^j(u) - \kappa_i(u)\right)^2 du}. \tag{3.13}$$

In addition, we use the following summary measures for comparison of functional sparsity:

- C_0: Correctly identified constant zero coefficient functions(proportion of replications in which κ_3 is correctly identified as zero),
- I_0: Incorrectly identified constant zero coefficient functions(total proportion of times $\kappa_0, \kappa_1, \kappa_2$ and κ_4 were incorrectly identified as zeros),
- $C_{i,0}$: Average length of correctly identified zero intervals for the ith coefficient function,

- $RC_{i,0}$: Average length of correctly identified zero intervals for the ith coefficient function/true length of zero interval for the ith coefficient function,
- $I_{i,0}$: Average length of incorrectly identified zero intervals for the ith coefficient function,
- $RI_{i,0}$: Average length of incorrectly identified zero intervals for the ith coefficient function/true length of nonzero interval for the ith coefficient function.

Note that C_0 and I_0 summarize the ability of the fitting procedure to recognize global sparsity, i.e., coefficient functions that are constant zero. $C_{i,0}$ and $I_{i,0}$ summarize the ability of the fitting procedure to capture local sparsity, i.e., zero intervals of coefficient functions that are partly zero [Tu et al., 2012]. $RC_{i,0}$ and $RI_{i,0}$ are also measures of local sparsity, and they are comparable across the coefficient functions. Model goodness-of-fit is evaluated by horizontal KS plot described in Section 2.4. A numerical summary of KS plot is the mean distance between KS plot and the diagonal line,

$$MDis(u) = \frac{1}{200}\sum_{j=1}^{200} |KS_j(u) - Diag(u)|, \tag{3.14}$$

where $KS_j(u)$ is the KS curve for the jth repetition , $0 \leq u \leq 1$. The results of the simulations are shown in Section 3.4.1-3.4.4.

3.4.1 Scenario 1: comparison of the three methods when correct B-spline basis functions are used and the sample size is 500.

In this section, the generated response spike trains consist of 500 spikes, and the correct B-spline basis functions are used to fit the model. Figure 3.4 and Figure 3.5 show the bias functions of the three methods, and RMISE values for each coefficient function are compared in Table 3.3. In general, for all predictors' coefficient functions, SCAD has smaller bias than PL and LASSO. LASSO has smaller bias than PL over the zero intervals of the true coefficient functions, but PL has smaller bias than LASSO over the nonzero intervals of the true coefficient functions. SCAD has the smallest RMISE values. PL has the largest

RMISE values. For κ_1 and κ_2, the RMISE values of SCAD is about 1/3 of the PL's RMISE values and 1/2 of LASSO's. For κ_3, SCAD has slightly larger RMISE value than LASSO, but its RMISE value is still very small and much smaller than PL's. Recall that κ_3 is constant zero, estimates of SCAD and LASSO are substantially better than PL because PL estimator does not guarantee sparsity. For κ_4, the RMISE values of the three methods are close and SCAD has the smallest value.

For the response's history, the estimates of the three methods all have higher RMISEs than those for predictors' histories. This is sensible, because the effect of the response's own history is harder to be captured. SCAD has larger bias than LASSO near zero. This is because it is not common to observe two response spikes occurring very close with each other, especially when the sample size is small ($J = 500$). With limited information, PL and SCAD tend to overfit and the estimates are not stable, i.e., have large variance. For the same reason, SCAD also has larger RMISE value than LASSO. PL has the largest bias and RMISE values. In summary, SCAD performs better than PL and LASSO except over the interval near 0 of the response's history.

The sparsity measures are shown in Table 3.4 and Table 3.5. PL does not guarantee sparsity, so we focus on comparing LASSO with SCAD. From Table 3.4, SCAD successfully identifies the constant zero coefficient function (κ_3) 95.5% of the time, and this proportion is much larger than the LASSO's (34.5%). For each individual coefficient function, the average length ($C_{i,0}$) of correctly identified zero intervals by SCAD is much larger than by LASSO. For example, for κ_1, the true length of the zero interval is 0.9; the average length of correctly identified zero intervals by SCAD is 0.89, on average, SCAD is able to correctly identify almost all the zero intervals. The average length of correctly identified zero intervals by LASSO is 0.4, on average, LASSO is able to correctly identify less than half of the zero intervals. Table 3.5 shows the incorrect identification results. There are four coefficient functions that are not constant zero. The average proportion of those 4 coefficient functions incorrectly identified as constant zero functions by SCAD is 0.045. This proportion

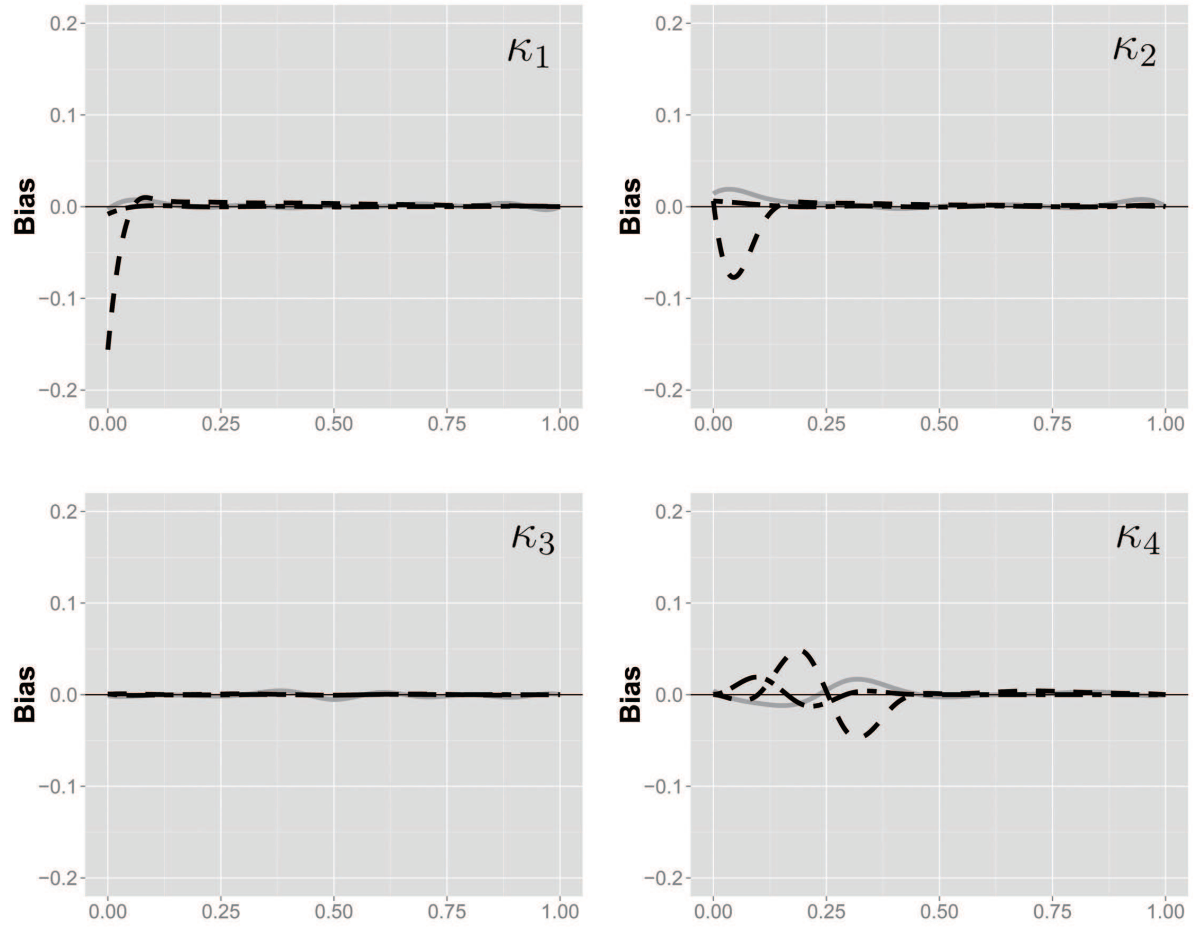

Figure 3.4: Comparison of bias of the coefficient functions of the predicator trains estimated by PL (gray solid), LASSO (black dashed), and SCAD (black dash-dotted) when the sample size is 500.

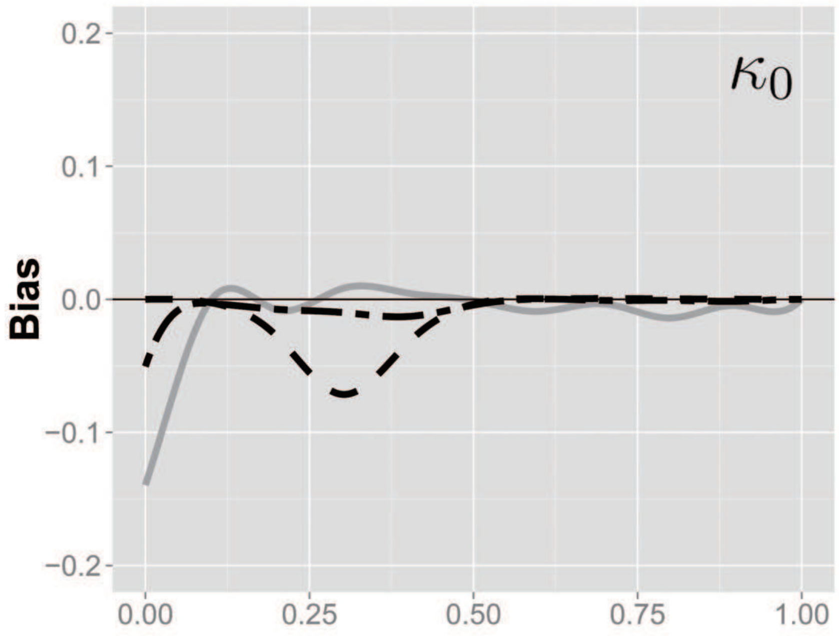

Figure 3.5: Comparison of bias of the coefficient functions of the response's history estimated by PL (gray solid), LASSO (black dashed), and SCAD (black dash-dotted) when the sample size is 500.

of incorrectly identification is very small. LASSO has a slightly larger value, which is 0.055. Also, the average lengths ($I_{i,0}$) of incorrectly identified zero intervals of the five coefficient functions individually by LASSO and SCAD are all very small. Moreover, many of these averages are zero. In summary, SCAD has much better performance than PL and LASSO in achieving both global and local sparsity.

Figure 3.6 shows in-sample model goodness-of-fit and out-of-sample model prediction performance of models estimated by PL, LASSO and SCAD. The PL method performs as well as SCAD for the in-sample case, however, its performance is much worse than SCAD for the out-of-sample case. This is because it fails in achieving sparsity of the estimated coefficient functions and under smooths the baseline hazard function. LASSO performs worse than SCAD and PL for the in-sample case because it shrinks the coefficient functions. However, its performance is closed to SCAD for the out-of-sample case.

Table 3.3: Comparison of RMISEs when sample size is 500.

Estimator	κ_1	κ_2	κ_3	κ_4	κ_0
PL	0.033	0.038	0.032	0.026	0.064
LASSO	0.021	0.022	0.004	0.022	0.030
SCAD	0.010	0.012	0.005	0.017	0.037

Table 3.4: Sparsity summary measures: correctly identified when sample size is 500. The values in the parentheses are the relative values, $RC_{i,0}$.

Estimator	C_0	$C_{1,0}$	$C_{2,0}$	$C_{3,0}$	$C_{4,0}$	$C_{0,0}$
PL	0	0	0	0	0	0
LASSO	0.345	0.400(.44)	0.501(.62)	0.644(.64)	0.194(.38)	0.367(.84)
SCAD	0.955	**0.890(.99)**	**0.781(.97)**	**0.987(.99)**	**0.502(1.00)**	**0.370(.92)**
True 0	1	0.9	0.802	1	0.502	0.402

Table 3.5: Sparsity summary measures: incorrectly identified when sample size is 500. The values in the parentheses are the relative values, $RI_{i,0}$.

Estimator	I_0	$I_{1,0}$	$I_{2,0}$	$I_{3,0}$	$I_{4,0}$	$I_{0,0}$
PL	0	0	0	0	0	0
LASSO	0.055	**0.001(.01)**	0(0)	0(NA)	0(0)	0.177(.3)
SCAD	**0.045**	0.002(.02)	0(0)	0(NA)	0(0)	0.177(.3)
True nonzero	4	0.1	0.198	0	0.498	0.598

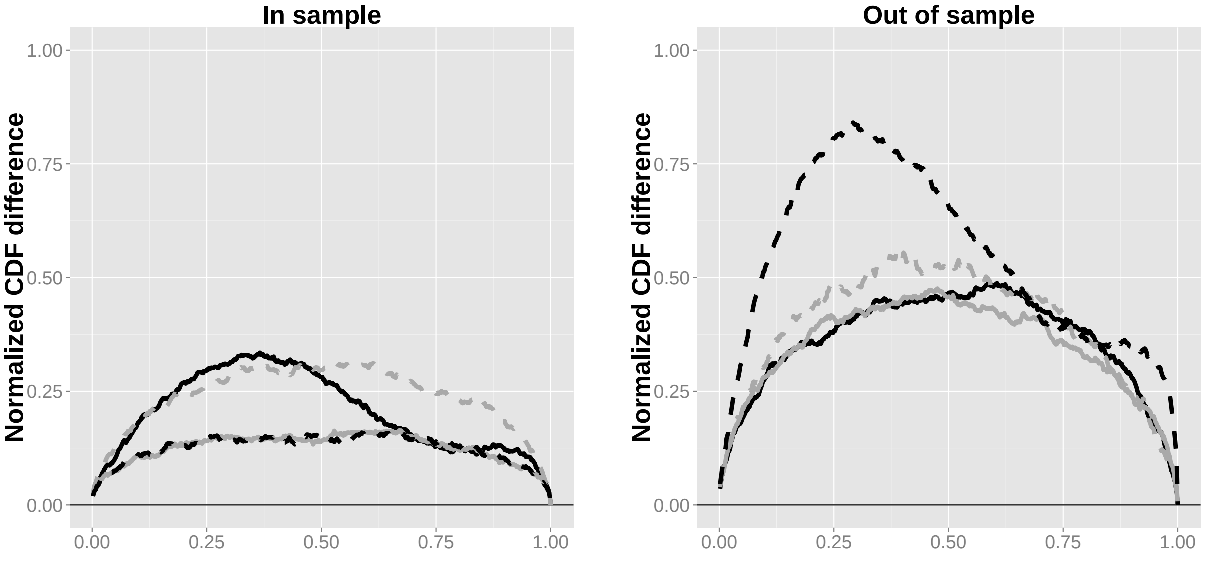

Figure 3.6: Model goodness-of-fit shown with horizontal KS plots of estimated models when the sample size is 500. PL (black dotted), LASSO (black solid), SCAD (gray solid), true model (In-Sample gray dashed) and NULL model (Out-of-Sample gray dashed).

3.4.2 Scenario 2: comparison of the three methods when incorrect B-spline basis functions are used and the sample size is 500.

To evaluate the performance of PL, LASSO and SCAD when the coefficient functions are not in the space spanned by a linear combination of B-splines used to fit the model, the data generated in Section 3.4.1 is fitted using B-spline basis functions with incorrect number of knots. We use B-spline basis functions with 3, 6, 12, 15 and 18 equally spaced interior knots to approximate the coefficient functions. Recall that the correct number of interior knots is 9. The performance of these incorrect basis functions are evaluated in terms of MISE. MISEs of the estimated coefficient functions using 9 knots are used as references. We compute the relative differences between MISEs using k knots and MISEs using 9 knots as

$$RDMISE_i^{(k)} = \frac{(MISE_i^{(k)} - MISE_i^{(9)})}{MISE_i^{(9)}} \tag{3.15}$$

To evaluate the overall local sparsity identification, average relative correctly identified zero interval and average relative incorrectly identified zero interval are computed as

$$ARC^{(k)} = \frac{1}{5}\sum_{i=0}^{4} RC_{i,0}^{(k)}, \tag{3.16}$$

$$ARI^{(k)} = \frac{1}{5}\sum_{i=0}^{4} RI_{i,0}^{(k)}. \tag{3.17}$$

The maximum mean distance (MMD) between KS plot and the diagonal line is used as a goodness-of-fit measure,

$$MMD^{(k)} = \max_u MDis(u)^{(k)} = \max_u \left(\frac{1}{200}\sum_{j=1}^{200} |KS_j^{(k)}(u) - Diag(u)| \right). \tag{3.18}$$

The changes in MISE values of the three methods are shown in Table 3.6. Negative relative difference values imply that the MISE values of the estimates are smaller than the estimates obtained by using the correct B-spline basis functions. From Table 3.6, PL estimates have smaller MISE values when the number of B-spline basis functions is small. This is because PL tends to over-fit, so when the number of parameters is smaller, the over-fitting is less severe. For LASSO and SCAD, most of the relative difference values are positive, so they have the smallest MISE values when the number of knots is equal to 9, the correct number. The MISE values when the number of knots is equal to 9 are also given in the Table 3.6. Recall that SCAD has smaller MISE values for all the predictor trains.

Similar patterns can be seen in the changes of the out-of-sample MMD values (Table 3.7 and Figure 3.7). From Table 3.7, the out-of-sample MMD value of PL increases as the number of knots increases, in particular it increases from 0.560 to 1.095 and this is illustrated by the left plot in Figure 3.7. The MMD values of LASSO and SCAD do not change much. Also SCAD has smaller MMD values in both in-sample case and out-of-sample case than PL and LASSO for most knot numbers.

Table 3.8 shows the sparsity summary measures for various knot-numbers. For the correctly identification measures (C_0 and ARC), SCAD has larger values than LASSO, which implies that on average, SCAD is able to correctly identify more constant zero coefficient functions and zero intervals. For the incorrectly identification measures (I_0 and ARI), SCAD has smaller or larger but still close values than LASSO, which implies that on average, SCAD does not incorrectly identify more than LASSO. In general, SCAD has better performance than LASSO in achieving both global sparsity and local sparsity for all numbers of knots. The best performance is obtained when the number of knots is correct, that is, for the correctly identification measures (C_0 and ARC), SCAD has the largest values (0.96 and 0.97) when the number of knots is 9.

In summary, the performance of PL is greatly affected by the number of knots, while for LASSO and SCAD, the performance does not change much as the number of knots changes, i.e., they are less sensitive to number of knots choice. Moreover, for various knot numbers, SCAD have better performance than PL and LASSO in model estimation, variable selection and model prediction.

Table 3.6: Comparison of RDMISEs for various number of knots values when sample size is 500.

Estimator	knots	κ_1	κ_2	κ_3	κ_4	κ_0
PL MISE	9	0.00110	0.00145	0.00102	0.00067	0.00407
Relative difference	3	-0.573	-0.138	-0.657	1.104	-0.555
	6	-0.327	-0.214	-0.324	-0.104	-0.408
	12	0.300	0.317	0.294	0.269	0.066
	15	0.591	0.566	0.588	0.522	0.337
	18	0.873	0.814	0.853	0.746	0.570
LASSO MISE	9	0.00046	0.00047	0.00002	0.00047	0.00088
Relative difference	3	0.130	1.723	0.500	2.979	0.636
	6	-0.130	0.787	0.500	0.383	-0.034
	12	0.304	0.426	0.000	-0.170	0.341
	15	0.478	0.660	0.000	0.128	0.602
	18	0.522	0.702	0.000	0.383	0.784
SCAD MISE	9	0.00010	0.00014	0.00003	0.00029	0.00136
Relative difference	3	3.200	7.214	-0.333	4.448	0.140
	6	0.600	3.286	-0.333	0.207	-0.309
	12	0.800	1.929	0.000	-0.034	0.051
	15	1.800	2.143	0.667	1.517	0.368
	18	3.200	3.857	3.000	1.966	0.794

Table 3.7: Model goodness-of-fit evaluated with KS test for various number of knots values when sample size is 500.

Number of knots	3	6	9	12	15	18
	In-sample					
PL	0.177	0.166	0.159	0.170	0.163	0.165
LASSO	0.261	0.292	0.335	0.356	0.359	0.366
SCAD	0.163	0.163	0.164	0.174	0.175	0.185
	Out-of-sample					
PL	0.560	0.727	0.841	0.928	1.014	1.095
LASSO	0.517	0.501	0.491	0.493	0.497	0.509
SCAD	0.509	0.502	0.476	0.487	0.502	0.516

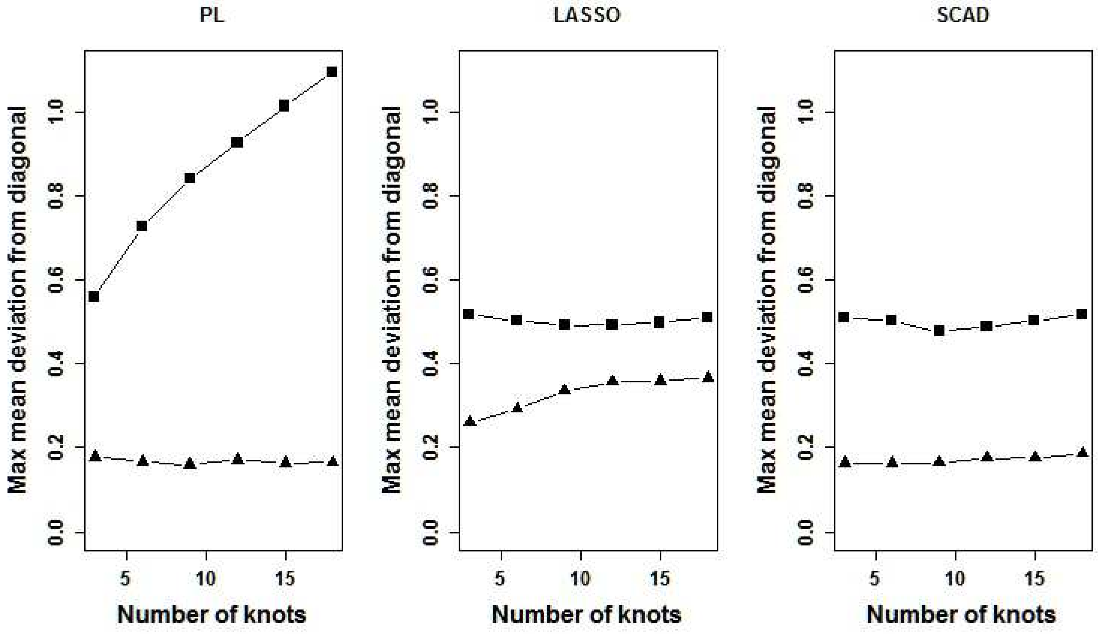

Figure 3.7: Comparison of the max mean deviation of the KS plot from the diagonal line for In-Sample (triangle) and Out-of-Sample (square) for various number of knots values when sample size is 500.

Table 3.8: Comparison of sparsity summary measures for various number of knots values when sample size is 500.

Estimator	C_0						ARC					
	3	6	9	12	15	18	3	6	9	12	15	18
LASSO	0.29	0.34	0.35	0.48	0.45	0.47	0.29	0.50	0.60	0.68	0.72	0.74
SCAD	0.95	0.98	0.96	0.94	0.89	0.78	0.74	0.91	0.97	0.96	0.95	0.93
	I_0						ARI					
	3	6	9	12	15	18	3	6	9	12	15	18
LASSO	0.09	0.09	0.05	0.13	0.15	0.22	0.03	0.02	0.08	0.15	0.17	0.20
SCAD	0.11	0.05	0.06	0.06	0.07	0.02	0.02	0.04	0.08	0.21	0.26	0.28

3.4.3 Scenario 3: comparison of the three methods when correct B-spline basis functions are used and the sample size is 1000.

We increase the number of response spikes generated in each repetition to 1000 to investigate the performance of the three methods for a larger sample size. Again, we start with using the correct number of knots and then evaluate the performance of the methods when incorrect number of knots is used.

B-spline basis functions with $d = 3$ and 9 equally spaced interior knots are used to approximate the coefficient functions. Figure 3.8 shows the biases of coefficient functions estimated by PL, LASSO and SCAD. Comparing with the biases based on 500 response spikes, the biases of all three methods decrease. In particular, the bias of SCAD over the interval near 0 of the response's history decreases greatly and SCAD has the smallest biases over almost all intervals of all five coefficient functions. SCAD also has the smallest RMISEs (Table 3.9). Tables 3.10 and 3.11 show the sparsity measures. SCAD performs better than PL and LASSO in achieving both global and local sparsity. Its performance increases as the sample size increases.

Table 3.9: Comparison of RMISEs when sample size is 1000.

Estimator	κ_1	κ_2	κ_3	κ_4	κ_0
PL	0.021	0.024	0.022	0.018	0.036
LASSO	0.015	0.015	0.003	0.016	0.014
SCAD	0.005	0.007	0.003	0.012	0.014

Table 3.10: Sparsity summary measures: correctly identified, when sample size is 1000. The values in the parentheses are the relative values, $RC_{i,0}$.

Estimator	C_0	$C_{1,0}$	$C_{2,0}$	$C_{3,0}$	$C_{4,0}$	$C_{0,0}$
PL	0	0	0	0	0	0
LASSO	0.370	0.386(.43)	0.524(.65)	0.695(.70)	0.207(.49)	0.339(.91)
SCAD	0.965	**0.896(.99)**	**0.795(.99)**	**0.990(.99)**	**0.502(1.00)**	**0.393(.98)**
True 0	1	0.9	0.802	1	0.502	0.402

The in-sample and out-of-sample performances of models estimated by PL, LASSO and SCAD are shown in Figure 3.10. SCAD performs best in both in-sample and out-of-sample

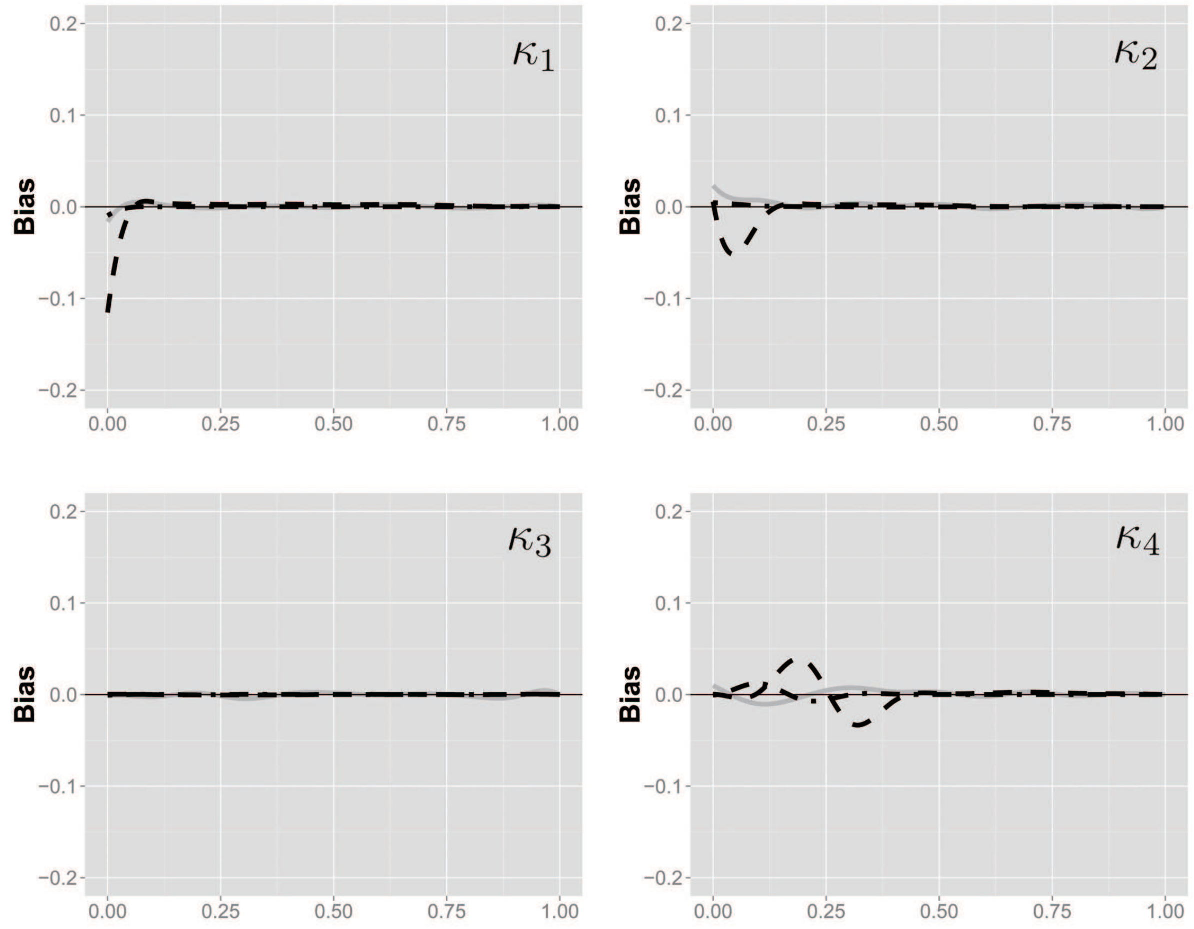

Figure 3.8: Comparison of bias of the coefficient functions of the predicator trains estimated by PL (gray solid), LASSO (black dashed), and SCAD (black dash-dotted) when the sample size is 1000.

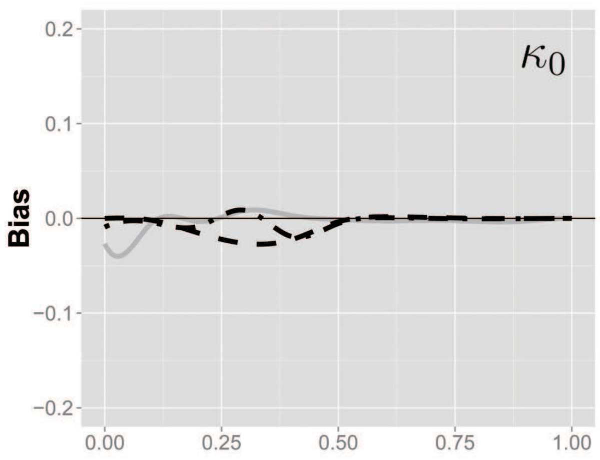

Figure 3.9: Comparison of bias of the coefficient functions of the response's history estimated by PL (gray solid), LASSO (black dashed), and SCAD (black dash-dotted) when the sample size is 1000.

cases. Figure 3.11 compares the performance of models estimated based on 500 response spikes and those based on 1000 response spikes. The performance of all three methods is improved as the sample size increases. PL has the largest improvement.

Table 3.11: Sparsity summary measures: incorrectly identified, when sample size is 1000. The values in the parentheses are the relative values, $RI_{i,0}$.

estimator	I_0	$I_{1,0}$	$I_{2,0}$	$I_{3,0}$	$I_{4,0}$	$I_{0,0}$
PL	0	0	0	0	0	0
LASSO	0	**0(0)**	0(0)	0(NA)	0(0)	0.115(.19)
SCAD	**0**	0(0)	0(0)	0(NA)	0(0)	0.182(.30)
True nonzero	4	0.1	0.198	0	0.498	0.598

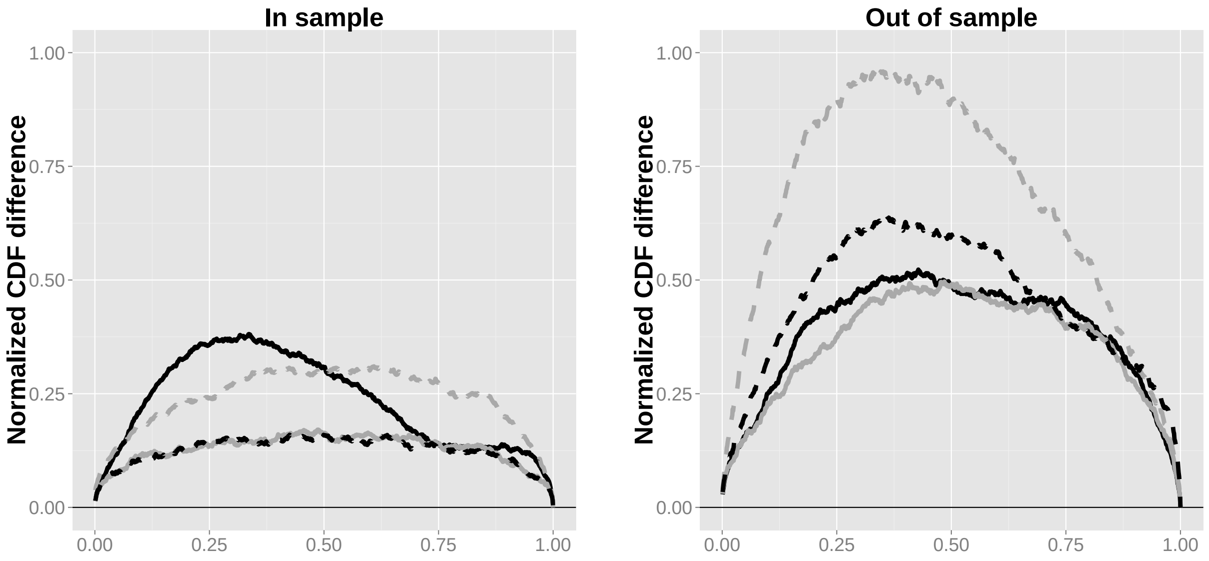

Figure 3.10: Model goodness-of-fit shown with horizontal KS plots of estimated models when the sample size is 1000. PL (black dotted), LASSO (black solid), SCAD (gray solid), true model (In-Sample gray dashed) and NULL model (Out-of-Sample gray dashed).

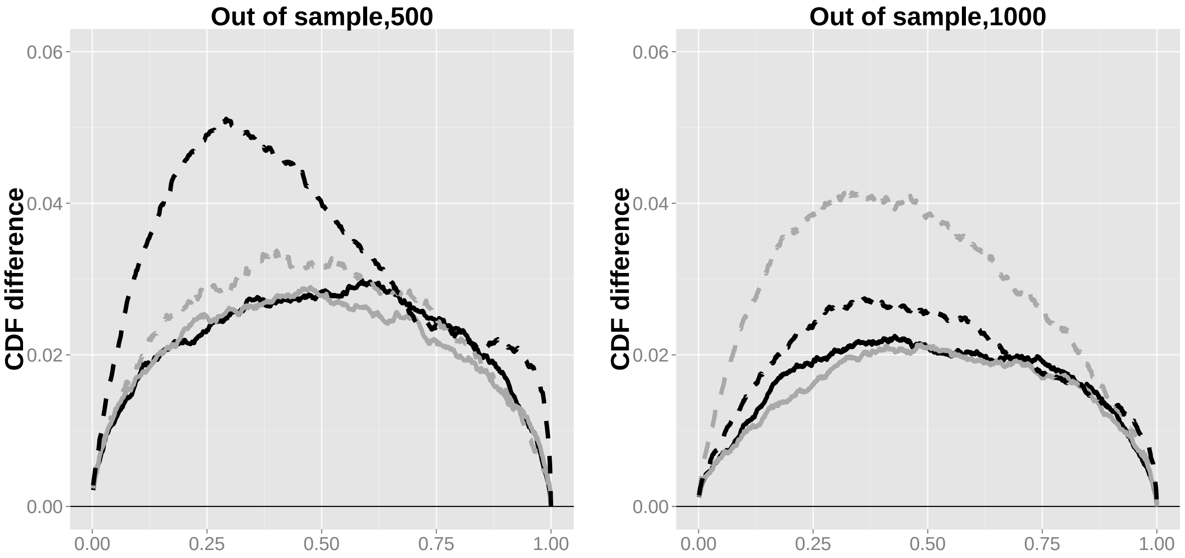

Figure 3.11: Comparison of model goodnees-of-fit shown with horizontal KS plots for different sample sizes 500 and 1000. PL (black dotted), LASSO (black solid), SCAD (gray solid), and NULL model (gray dotted).

3.4.4 Scenario 4: comparison of the three methods when incorrect B-spline basis functions are used and the sample size is 1000.

Similar to Section 3.4.2, we evaluate the performance of PL, LASSO and SCAD when the true coefficient functions are not in the space spanned by a linear combination of B-splines used to fit the model. The sample size is increased to 1000. We use B-spline basis functions with $d = 3$ and 3, 6, 12, 15 and 18 equally spaced interior knots to approximate the coefficient functions. Recall that the correct number of interior knots is 9. Table 3.12 shows the RDMISEs. For LASSO and SCAD, the model using the correct number of knots has the smallest MISEs. For PL, in general, the model using 6 knots has the smaller MISEs than that using 9 knots. The sparsity measures are shown in Table 3.13. SCAD performs better than LASSO for all numbers of knots in achieving both global and local sparsity. Its performance achieves its best when the number of knots is correct. The performance of LASSO is better when the number of knots is larger. The model goodness-of-fit performance is shown in Table 3.14 and Figure 3.12. The results are similar to those based on 500 response spikes. The model predication of PL decreases as the number of knots increases, while the performance of LASSO and SCAD does not change much as the number of knots changes.

In summary, the simulation results demonstrate that for our proposed model, SCAD has smaller bias, RMISE and performs better in achieving both global and local sparsity than PL and Lasso. This advantage exists in both small sample size situation and fairly large sample size situation. In addition, the performance of SCAD is not sensitive to whether the correct number of knots is used for the B-spline basis functions.

Table 3.12: Comparison of RDMISEs for various number of knots values when sample size is 1000.

	knots	κ_1	κ_2	κ_3	κ_4	κ_0
PL MISE	9	0.00047	0.00061	0.00046	0.00032	0.00132
Relative difference	3	-0.340	0.574	-0.630	3.094	-0.189
	6	-0.319	-0.066	-0.283	0.156	-0.326
	12	0.277	0.295	0.283	0.219	0.189
	15	0.553	0.525	0.522	0.406	0.485
	18	0.830	0.754	0.783	0.625	0.773
LASSO MISE	9	0.00023	0.00021	0.00001	0.00026	0.00022
Relative difference	3	0.652	3.905	1.000	5.115	1.955
	6	-0.043	1.857	0.000	0.846	0.227
	12	0.348	0.476	0.000	-0.154	0.591
	15	0.565	0.571	0.000	0.115	1.045
	18	0.696	0.571	0.000	0.346	1.500
SCAD MISE	9	0.00003	0.00004	0.00001	0.00015	0.00021
Relative difference	3	10.333	24.250	-1.000	9.067	4.524
	6	1.667	10.500	-1.000	1.133	0.381
	12	1.000	4.250	0.000	0.133	1.429
	15	3.333	3.000	1.000	1.667	2.619
	18	4.333	6.250	3.000	2.667	3.952

Table 3.13: Comparison of sparsity summary measures for various number of knots values when sample size is 1000.

estimator	C_0						ARC					
	3	6	9	12	15	18	3	6	9	12	15	18
LASSO	0.19	0.33	0.37	0.42	0.46	0.44	0.24	0.50	0.60	0.64	0.72	0.69
SCAD	0.99	0.99	0.98	0.95	0.89	0.83	0.73	0.92	0.99	0.98	0.95	0.95
	I_0						ARI					
	3	6	9	12	15	18	3	6	9	12	15	18
LASSO	0	0	0	0	0	0	0	0.01	0.05	0.08	0.17	0.11
SCAD	0	0	0	0	0	0.01	0	0.01	0.08	0.19	0.26	0.23

Table 3.14: Model goodness-of-fit evaluated with KS test for various number of knots values when sample size is 1000.

Number of knots	3	6	9	12	15	18
	In-sample					
PL	0.184	0.159	0.165	0.162	0.166	0.170
LASSO	0.325	0.355	0.381	0.400	0.416	0.442
SCAD	0.170	0.169	0.171	0.181	0.177	0.181
	Out-of-sample					
PL	0.544	0.609	0.645	0.682	0.720	0.781
LASSO	0.562	0.521	0.523	0.550	0.552	0.562
SCAD	0.548	0.519	0.497	0.490	0.513	0.514

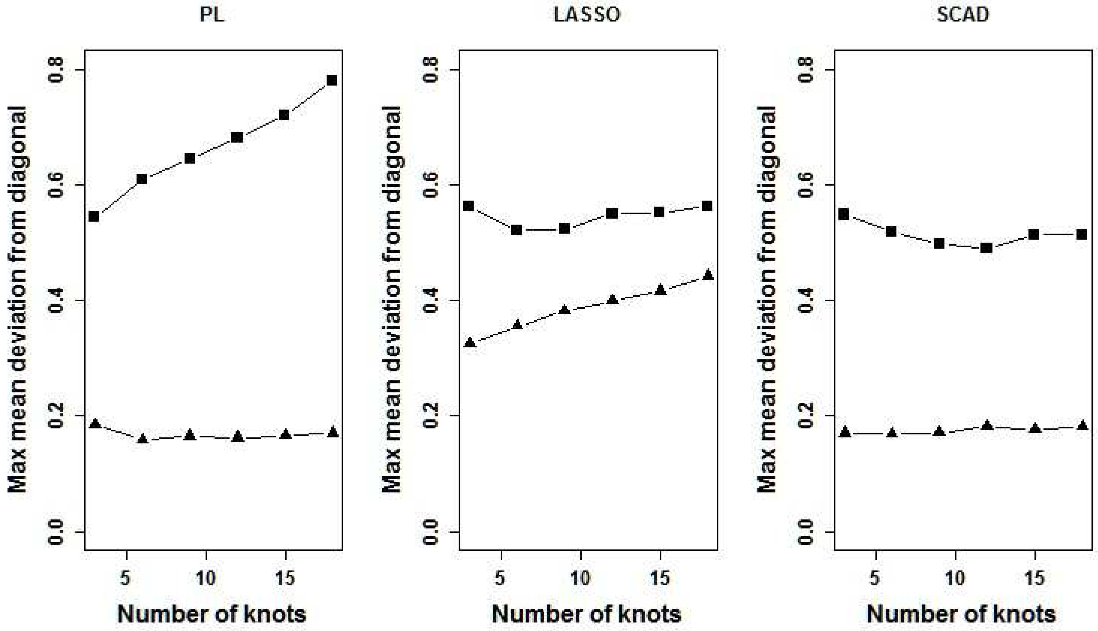

Figure 3.12: Comparison of the max mean deviation of the KS plot from the diagonal line for In-Sample (triangle) and Out-of-Sample (square) for various number of knots values when sample size is 1000.

3.4.5 Updating the time-dependent covariates less often

When the sample size, i.e., the number of response spikes, and the number of unknown parameters in the model are fixed, the computational complexity of model estimation is affected by the frequency of updating the time-dependent covariates. Ideally, we want to update the covariates whenever their values change. However, high updating frequency brings computational challenges and may not lead to significantly better estimates of the coefficient functions. In this section, we use simulation to compare the performances of the models based on different updating frequencies. We use the data generated in Section 3.4.3. Recall that the data consists of 200 repetitions, each of which includes four predictor spike trains and a response spike train of 1000 spikes. The response trains are generated by model (2.7) and covariates are updated every 0.004s in the data generation. In previous simulations, we fit the data using the models with correct updating frequency. Here, we use incorrect updating frequencies: the covariates are updated every 0.008s and 0.016s instead of 0.004s. The computational complexity of these models are less than the model using the correct frequency and reduction of computation is about the same as reducing the sample size to 500 and 250 respectively. The performances of the two models are compared with the models using the correct frequency and based on 500 and 1000 response spikes. Same as in previous simulations, the model performance is evaluated by bias, RMISE, sparsity summary measures and the KS plots.

Figures 3.13-3.15 show the bias curves of the estimated coefficient functions by PL, LASSO and SCAD using 1000 sample size and updating the time-dependent covariates every 0.004s, 0.008s and 0.016s. To illustrate the size of the bias, the true coefficient functions are also plotted. For all three methods, the difference in bias of different updating frequencies is small compared with the true coefficient functions. Table 3.15 shows the RMISE values. For PL estimates, the value increases very little as the updating frequency decrease from per 0.004s to per 0.016s. However, when the sample size is reduced to 500 from 1000, the RMISE value increases quite a bit. For LASSO and SCAD, the RMISE values does not increase very much when the updating frequency changes from per 0.004s to per 0.008s. But when the updating frequency is reduced to per 0.016s, the RMISE values increase to almost as large as those when the sample size is 500.

The sparsity measures are shown in Tables 3.16, 3.17. For both LASSO and SCAD, we do not see much change in those sparsity measures as the updating frequency changes. And with updating per 0.016s and sample size of 1000, the sparsity performance is still better than updating per 0.004s and sample size of 500. The model goodness-of-fit and prediction measures, shown in Table 3.18, support the same conclusion.

In summary, the models that update the covariates every 0.008s and 0.016s based on 1000 response spikes perform better or as well as the model that updates the covariates every 0.004s based on 500 response spikes. This implies that for the generated data, reducing the updating frequency is a good way of reducing the computational complexity without decreasing the model performance, which could be true for some real life data. In real data analysis, since penalized methods are much more computational expensive than PL, fitting models using various updating frequencies by PL and choosing an appropriate updating frequency based on the computational complexity and model performance could be a helpful step before model selection by methods like LASSO and SCAD.

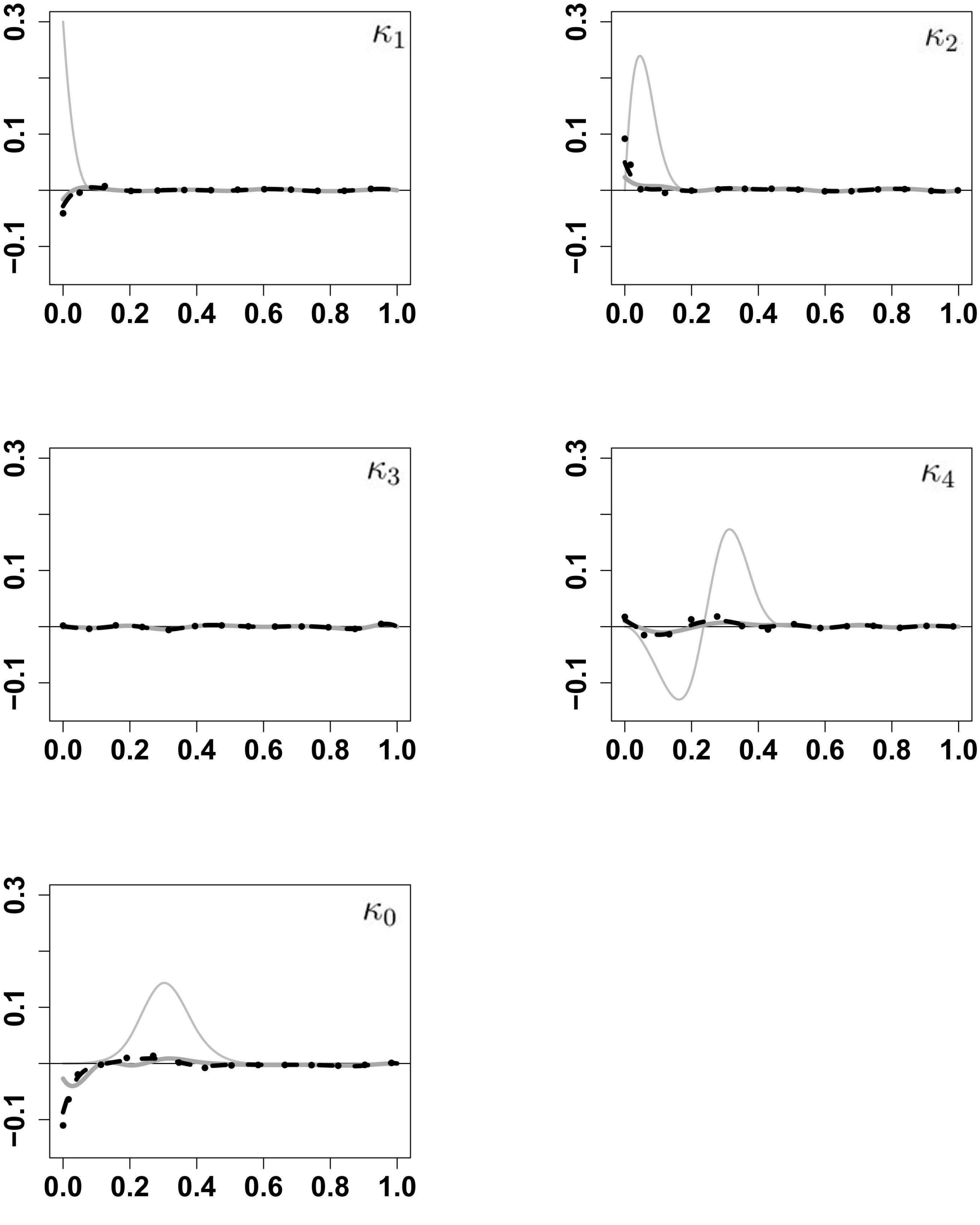

Figure 3.13: Bias comparison of coefficient functions estimated by PL updating the covariates every 0.004s (dark gray solid), 0.008s (black dashed) and 0.016s (black dotted). The gray solid curves are the true coefficient functions.

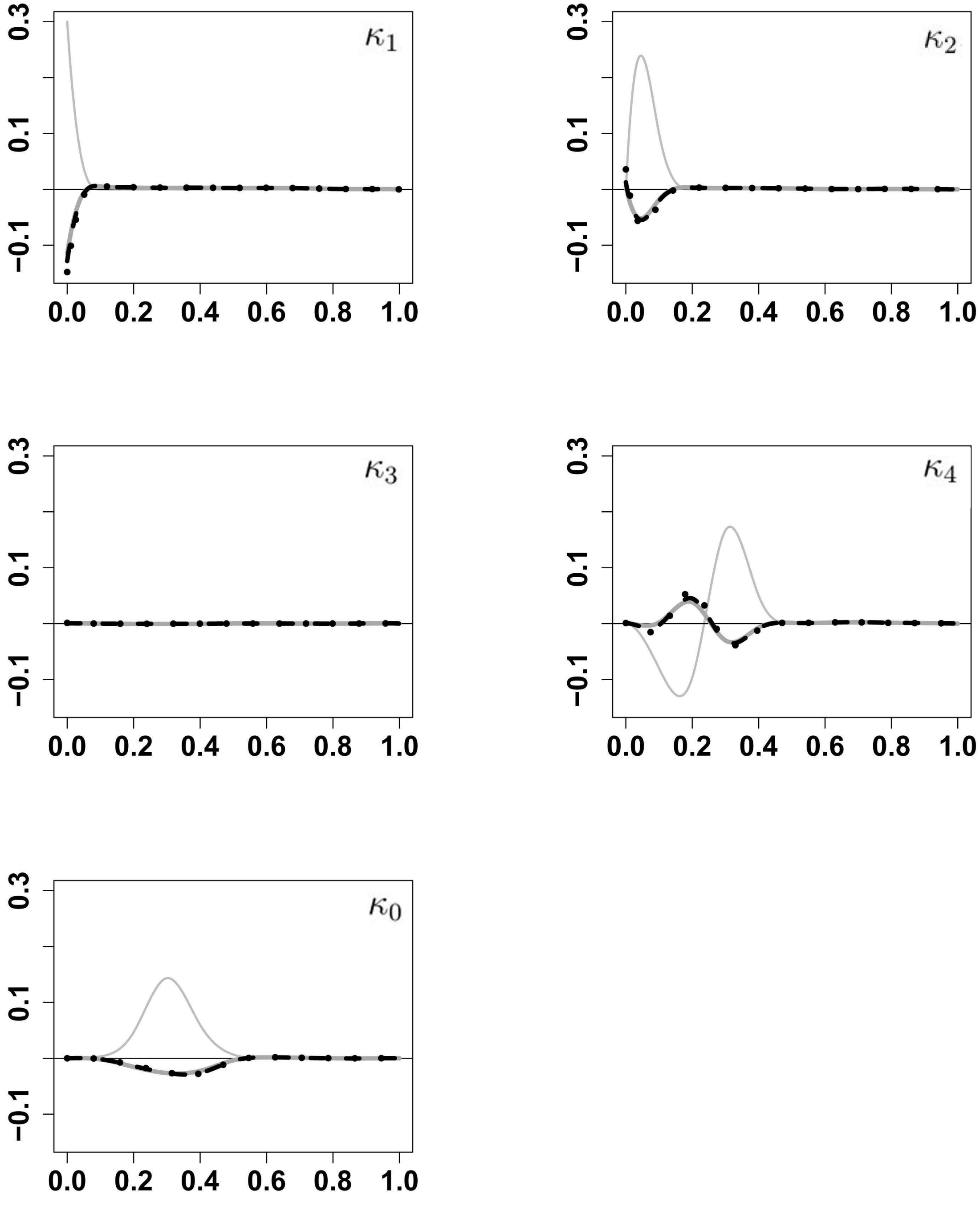

Figure 3.14: Bias comparison of coefficient functions estimated by LASSO updating the covariates every 0.004s (dark gray solid), 0.008s (black dashed) and 0.016s (black dotted). The gray solid curves are the true coefficient functions.

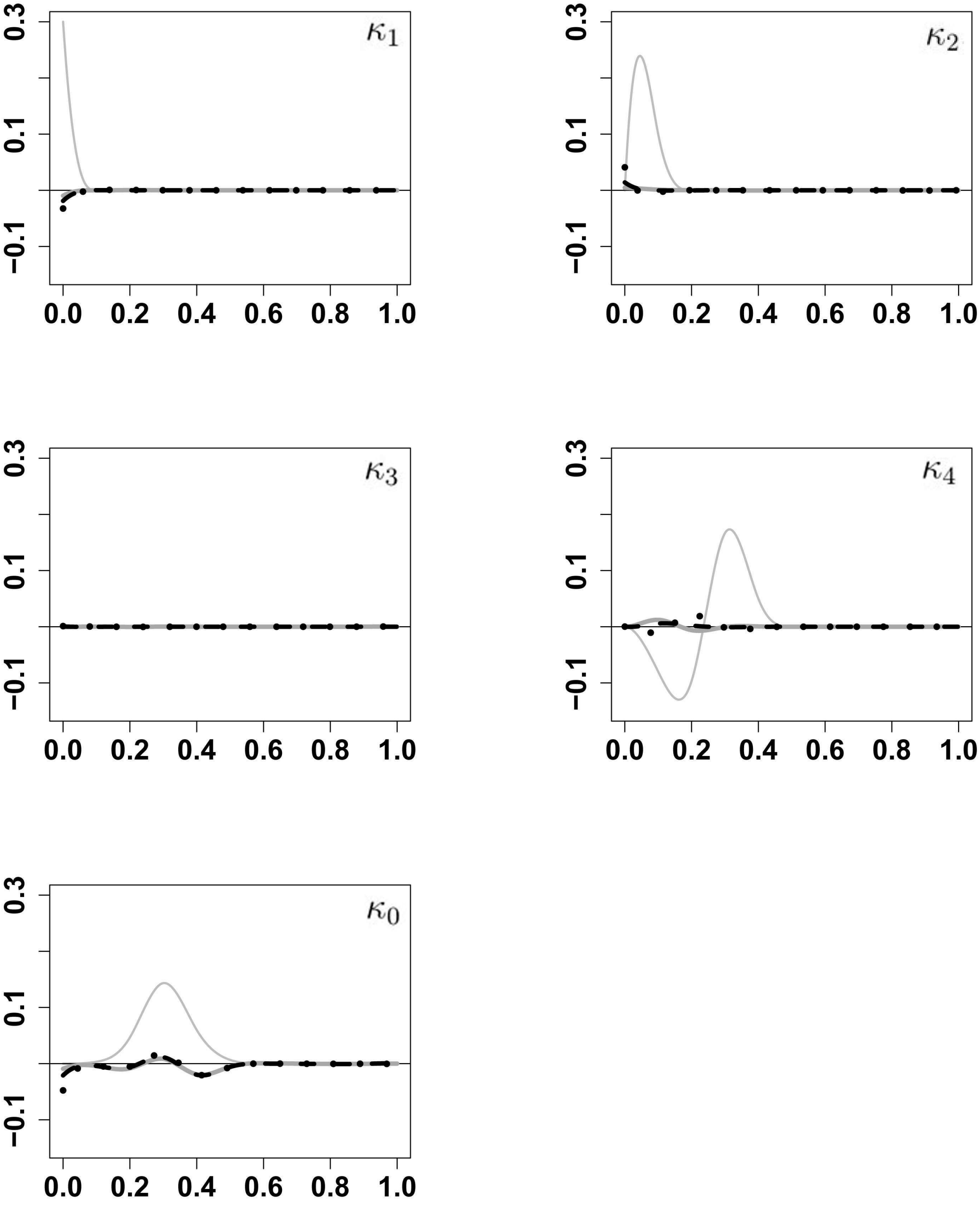

Figure 3.15: Bias comparison of coefficient functions estimated by SCAD updating the covariates every 0.004s (dark gray solid), 0.008s (black dashed) and 0.016s (black dotted). The gray solid curves are the true coefficient functions.

Table 3.15: Comparison of RMISEs for various updating frequency and sample size combinations.

	Sample size	Updating	κ_1	κ_2	κ_3	κ_4	κ_0
PL RMISE	500	0.004	0.033	0.038	0.032	0.026	0.064
	1000	0.004	0.022	0.025	0.021	0.018	0.036
	1000	0.008	0.022	0.026	0.022	0.018	0.036
	1000	0.016	0.023	0.028	0.022	0.020	0.037
LASSO RMISE	500	0.004	0.021	0.022	0.004	0.022	0.030
	1000	0.004	0.015	0.014	0.003	0.016	0.015
	1000	0.008	0.017	0.015	0.003	0.017	0.015
	1000	0.016	0.019	0.017	0.003	0.021	0.015
SCAD RMISE	500	0.004	0.010	0.012	0.005	0.017	0.037
	1000	0.004	0.005	0.006	0.003	0.017	0.014
	1000	0.008	0.005	0.007	0.002	0.013	0.016
	1000	0.016	0.007	0.011	0.003	0.017	0.019

Table 3.16: Comparison of sparsity summary measures: correctly identified, for various updating frequency and sample size combinations.

Estimator	Size	Updating	C_0	$C_{1,0}$	$C_{2,0}$	$C_{3,0}$	$C_{4,0}$	$C_{0,0}$
LASSO	500	0.004	0.345	0.399	0.500	0.644	0.194	0.367
	1000	0.004	0.370	0.386	0.524	0.695	0.207	0.339
	1000	0.008	0.400	0.392	0.498	0.701	0.208	0.342
	1000	0.016	0.355	0.373	0.506	0.690	0.203	0.341
SCAD	500	0.004	0.955	0.890	0.781	0.987	0.502	0.370
	1000	0.004	0.965	0.896	0.795	0.990	0.502	0.393
	1000	0.008	0.975	0.899	0.800	0.997	0.502	0.390
	1000	0.016	0.965	0.896	0.799	0.994	0.502	0.398
True			1	0.9	0.802	1	0.502	0.402

Table 3.17: Comparison of sparsity summary measures: incorrectly identified, for various updating frequency and sample size combinations.

Estimator	Size	Updating	I_0	$I_{1,0}$	$I_{2,0}$	$I_{3,0}$	$I_{4,0}$	$I_{0,0}$
LASSO	500	0.004	0.055	0.001	0	0	0	0.177
	1000	0.004	0	0	0	0	0	0.115
	1000	0.008	0	0	0	0	0	0.123
	1000	0.016	0	0	0	0	0	0.125
SCAD	500	0.004	0.045	0.002	0	0	0	0.177
	1000	0.004	0	0	0	0	0	0.182
	1000	0.008	0.005	0	0	0	0	0.179
	1000	0.016	0.005	0	0	0	0	0.167
True			4	0.1	0.198	0	0.498	0.598

Table 3.18: Model goodness-of-fit evaluated with KS test for various updating frequency and sample size combinations.

		In-sample				Out-of-sample			
	Size	500	1000	1000	1000	500	1000	1000	1000
Estimator	Updating	0.004	0.004	0.008	0.016	0.004	0.004	0.008	0.016
PL		0.0097	0.0071	0.0070	0.0078	0.0512	0.0277	0.0270	0.0290
LASSO		0.0204	0.0164	0.0167	0.0170	0.0299	0.0225	0.0221	0.0222
SCAD		0.0100	0.0074	0.0072	0.0080	0.0290	0.0214	0.0204	0.0203

3.5 Real data analysis: model selection for small sample

The dataset used in this section is a 200s continuous recording of neurons in layer CA1 of the right dorsal hippocampus. It is seconds 1300 to 1500 of the data set describe in Section 2.6.2 and we call it the Real Data 2. It contains 6 neural spike trains that recorded the neural spikes of 6 neuron cells. During this 200 seconds period, the 6 neurons generated 6722, 5874, 7613, 570, 8880 and 3715 spikes respectively. The sixth spike train, denoted as S_6, is chosen as the response and the other 5 spike trains are treated as functional predictors. The modulated renewal process model (2.7) is fitted to the data. This model includes covariates that reflect the predictor trains' history, the response train's history and the movement speed of the subject. The length of the history M is chosen to be 1 second. The history covariates are time-dependent and are updated every 0.004 second, which is a reasonably small value. B-spline basis functions with $d = 3$ and 11 interior knots are used to approximate the coefficient functions for the history covariates. The knots sequence is: 0, 0.020, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1. The interior knots are evenly spaced except the first two. This is different from the simulation in Section 3.4, where the knots are evenly spaced. We put more knots near 0, because it is believed that the history impacts are stronger and more complex near the current time. The B-spline basis function closest to the 1 end is not used because we want the coefficient functions to be 0 at 1. This is a reasonable constrain given that the length of the history is quite large.

The coefficient functions are estimated by PL, LASSO and SCAD methods. For LASSO and SCAD, the tuning parameters are selected by minimizing BIC as described in Section 3.2.

The estimated coefficient functions are shown in Figure 3.17, which consists of six plots. The first five plots show the estimated coefficient functions for the histories of the five predictor trains, κ_1 to κ_5, and the last plot shows the coefficient function for the response's own history (κ_0). PL does not identify any constant zero coefficient function nor zero interval of the nonzero coefficient functions. LASSO and SCAD do not identify any coefficient function as constant zero function. SCAD identifies shorter nonzero intervals than LASSO but larger effect over those nonzero intervals. Because the simulations in Section 3.4 demonstrate that SCAD has better performance than PL and LASSO, we focus on interpreting the SCAD estimates. In κ_1 and κ_2, SCAD estimates show strong positive components for short intervals less than 20ms. The estimated κ_3 shows a weaker positive component for a short interval ($<$ 20ms), followed by a weaker negative component for a short interval of about 30 ms, and then followed by another positive component for a longer interval (about 130 ms). The second positive component is weaker than the first one. In κ_4 and κ_5, SCAD estimates show a weak positive component for short intervals followed by a negative component for longer intervals. However, the nonzero interval (40 ms) in κ_4 is much shorter than that (180 ms) in κ_5. In κ_0, the coefficient function of the response history, SCAD estimate is zero in the first 30 ms interval and shows a weak but long last (280 ms) positive component. The results show that as expected, the effects of the histories are stronger near the current time (near 0) and weaker far from the current time (near 1). These results suggest that further analysis might be done with length of history M reduced from its current value of 1 second to a smaller value.

Figure 3.16 shows the estimated baseline hazard functions by PL, LASSO and SCAD. The baseline hazard function can be interpreted as the unique effect of time since the very last response spike in addition to the effects of the response spikes in the history. The SCAD estimate is small near 0, so the neuron tends to not fire again immediately after a firing (i.e., resting period). It increases rapidly to its maximum at approximate 15 ms and reaches its second peak at approximate 100 ms. The estimated baseline hazard function is not close to a constant function, which supports the notion that it is better to model the response train as a modulated renewal process than an inhomogeneous Poisson process. PL and LASSO estimates are smoother than the SCAD estimate, and they over-smooth the baseline hazard function. This is demonstrated by the out-of-sample KS plot (Figure 3.18) that evaluates the prediction performance of the models. We use the 500 seconds data that immediately follow the 200 seconds data used for fitting the models to construct the out-of-sample KS plot. The plot shows that the SCAD model yield smaller KS score than PL and LASSO models, i.e., SCAD model has better prediction performance.

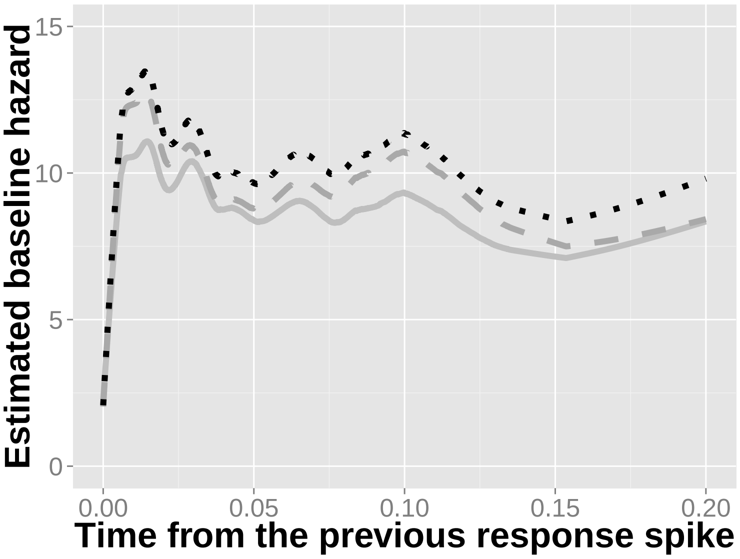

Figure 3.16: Estimated baseline hazard functions: PL estimation (gray solid), Lasso estimation (black dashed) and SCAD estimation (black dotted) based on Real Data 2.

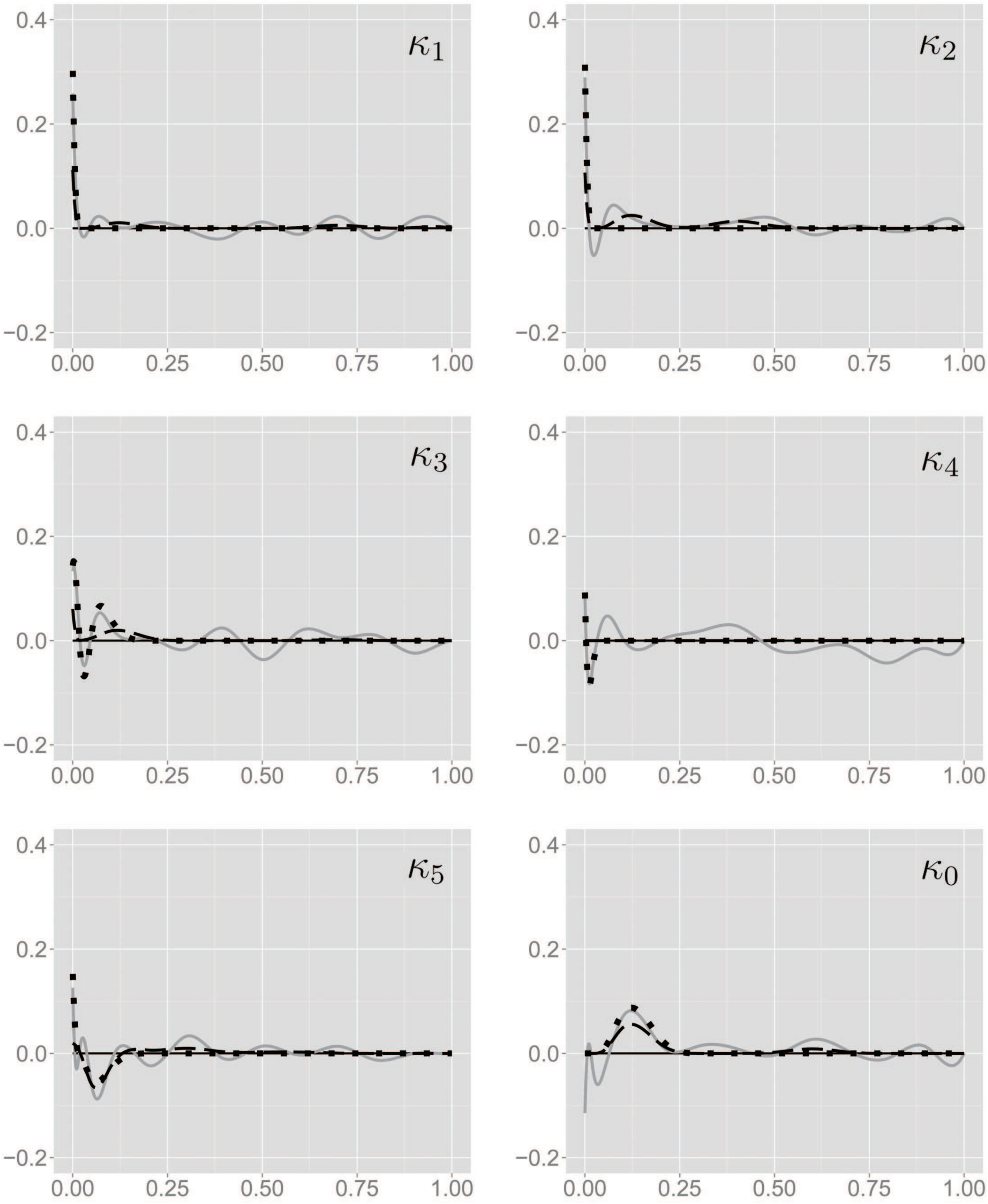

Figure 3.17: PL (black dotted), LASSO (gray solid) and SCAD (black dashed) estimates of coefficient functions for the five predictor trains (the first five subplots) and the response history (the sixth subplot) based on Real Data 2.

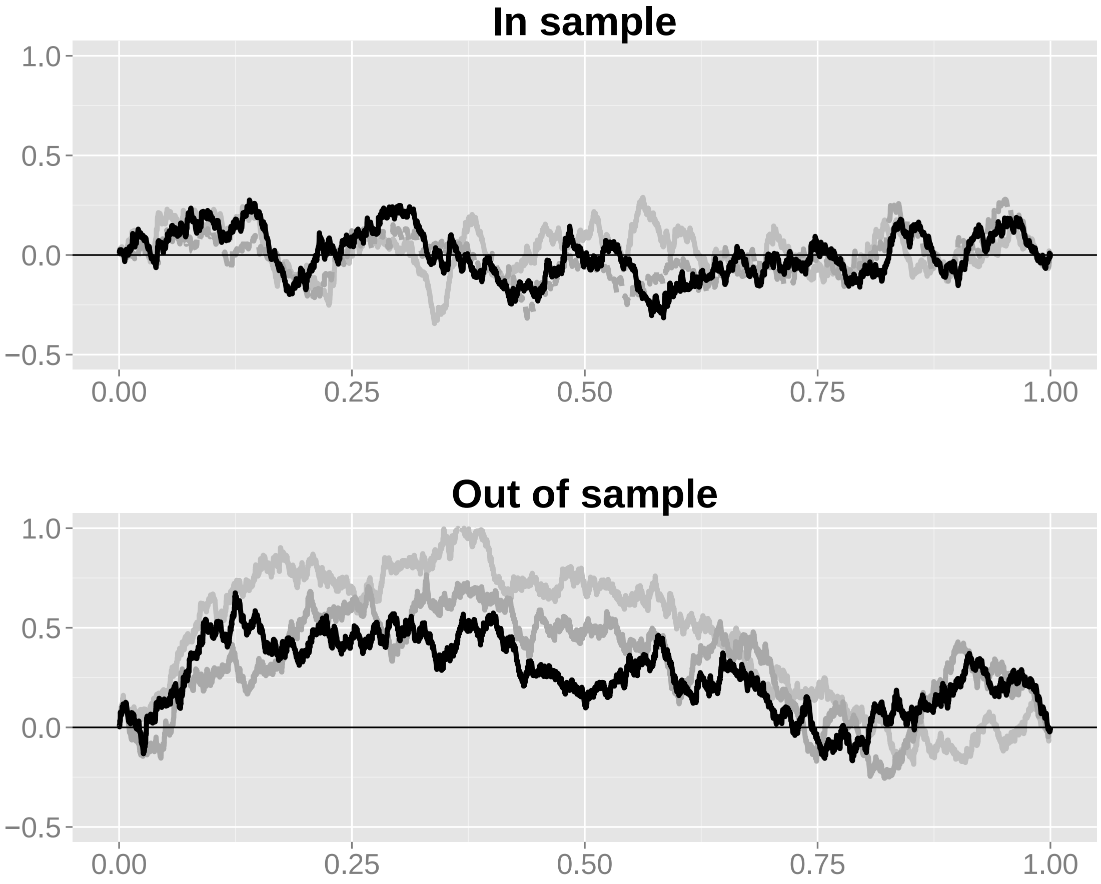

Figure 3.18: In-sample and out-of-sample horizontal KS plots: PL estimation (gray solid), LASSO estimation (black solid) and SCAD estimation (black dotted) based on Real Data 2.

CHAPTER 4

Future work

4.1 Big-data computing

Given the large number of spikes and sparse but complex relationships among neurons, the analysis of neural spike trains is bound to face computational challenges. For the proposed model, the penalized methods require a large amount of memory when the number of spikes is large. For example, greater than 20GB memory is required for 10000 spikes and 6 spike trains. Some studies [Song et al., 2007] showed that including second order interactions in the model can increase the ability of capturing the connections between neurons. However, this will greatly increase the number of parameters in the model and increase the computational complexity. Therefore, for large neural spike train data, approaches to reducing computational complexity are needed. Some available approaches include map-reduce [Dean and Ghemawat, 2008] and coordinate decent [Friedman et al., 2010].

4.2 Non-stationarity modeling

In the real data analysis, we see that the relationships between the response train and the predictor trains may not be stationary, i.e., the coefficient functions may vary over the real time (the time from the start of the observation). Therefore, for neural spike train data over a long observation period, it is reasonable to develop models that can account for the non-stationarity. Such models include:

1. Mixed models

Divide the observation period into sub-periods based on extrinsic information of the subject that may affect the neural activities, for example, location, movement, stimulus, etc., and, include sub-period random effect(s) in the model. There are two main challenges: first, identification of the sub-periods is difficult. Second, the likelihood does not have a closed form due to the integration with respect to the distribution of the random effects.

2. Locally non-stationary models

The goal of this type of models is to better capture the effect of the response train's history. Some differences between the response train's history and the predictor trains' history up to time t are: (1) There are no response spikes between $t_{N(t-)}$ and $t_{N(t-)+1}$, but there can be predictor spikes during that time. (2) The most recent response spike before t may have special effect that is different from the effect of the other previous response spikes. But the predictor spikes should all have the same effect on the future response spikes. Given these two differences, the following extension of the proposed modulated renewal process model may be able to better capture the effect of response train's history.

The first extension is to assume the effect of a response does not depend only on its distance from t; instead, its effect depends on both its distance from the most recent response spike before t as well as the distance between t and the most recent response spike $t_{N(t-)}$. That is, the effect of a response spike at t_j on the conditional intensity at time $t, t > t_j$ is a function of $t - t_{N(t-)}$ and $t_{N(t-)} - t_j$,

$$\begin{aligned}\lambda(t|H_t) &= \lambda_0(t - t_{N(t-)}) \exp\left\{ ... + \sum_{j=1}^{N(t-)} \kappa_0(t_{N(t-)} - t_j, t - t_{N(t-)}) I\{t_{N(t-)} - t_j \leq M\} ... \right\} \\ &= \lambda_0(t - t_{N(t-)}) \exp\left\{ ... + \sum_{u=0}^{m} \kappa_0(u\Delta, t - t_{N(t-)}) x_0(t_{N(t-)} - u\Delta) + ... \right\}. \qquad (4.1)\end{aligned}$$

Comparison of this extension and our proposed model is illustrated in Figure 4.1. Figure 4.1 (a) shows that for our proposed model, the effect of a spike at t_j on the conditional intensity at t is a function of $t - t_j$. Figure 4.1 (b) shows that for the extension, the effect of a spike at t_j on the conditional intensity at t is a function of $t - t_{N(t-)}$ and $t_{N(t-)} - t_j$,

Estimating the two dimensional coefficient surface is difficult. One approach is to approximate it using B-spline basis like this $\kappa_0(u\Delta, t - t_{N(t-)}) \approx \sum_{j_1} \sum_{j_2} \alpha_{j_1,j_2} B_{j_1}(u) B^*_{j_2}(t - t_{N(t-)})$. Note that instead of using the two dimensional B-spline basis function, we use the product of two one dimensional B-spline basis functions. The challenge of the this model is that the number of parameters is increased and the shape of $\kappa_0(u\Delta, t - t_{N(t-)})$ with respect to

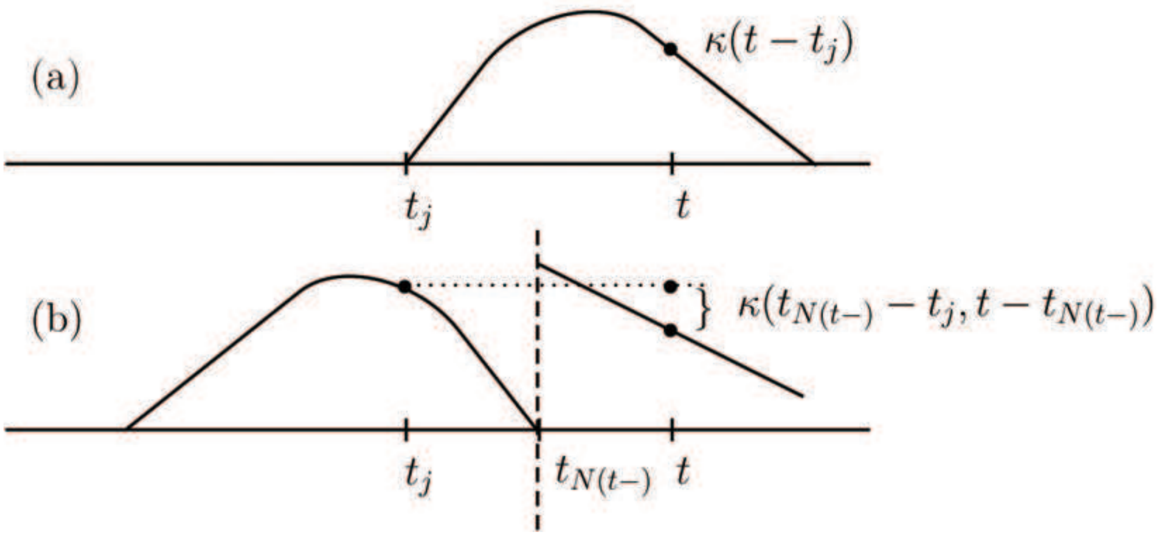

Figure 4.1: Comparison of the proposed modulated renewal process model and the locally non-stationary model.

$t - t_{N(t-)}$ should not be too complex. We call this model locally non-stationary model because it allows the effects of history response spikes to vary according to their distances from the most recent response spike. In principle, the effects of the predictor spikes can be modeled in this way, however, first, it would greatly increase the number of parameters and second, there are predictor spikes after the most recent response spike and they need to be handled differently from those before the most recent response spike.

3. Globally non-stationary models

The goal of this type of model is to model the variation of the coefficient functions along the real time trajectory. This requires information about how the relationships among neurons vary over time. One possible approach is to assume the relationships change according to the status of the subject that can be measure by the extrinsic covariates such as location, movement, stimulus etc. Under this assumption, including interactions between history covariates and extrinsic covariate capture the variation of the coefficient functions of the history covariates. Another approach that seems to be reasonable when information about the subject's status is not available is to assume the effects vary periodically. We extend the one dimensional coefficient function to two dimensional such that the effect of a spike in history does not only depend on its distance from t but also depend on t itself. That is

$$\lambda(t|H_t) = \lambda_0(t - t_{N(t-)}) \exp\left\{\sum_{i=0}^{p}\sum_{u=0}^{m}\kappa_i(u\Delta, t)x_i(t - u\Delta) + ...\right\}. \tag{4.2}$$

Comparison of this extension and our proposed model is illustrated in Figure 4.2. Figure 4.2 (a) shows that for our proposed model, the effect of a spike at t_j on the conditional intensity at t is a function of $t - t_j$. Figure 4.2 (b) shows that for the extension, the effect of a spike at t_j on the conditional intensity at t is a function of $t - t_j$ and t.

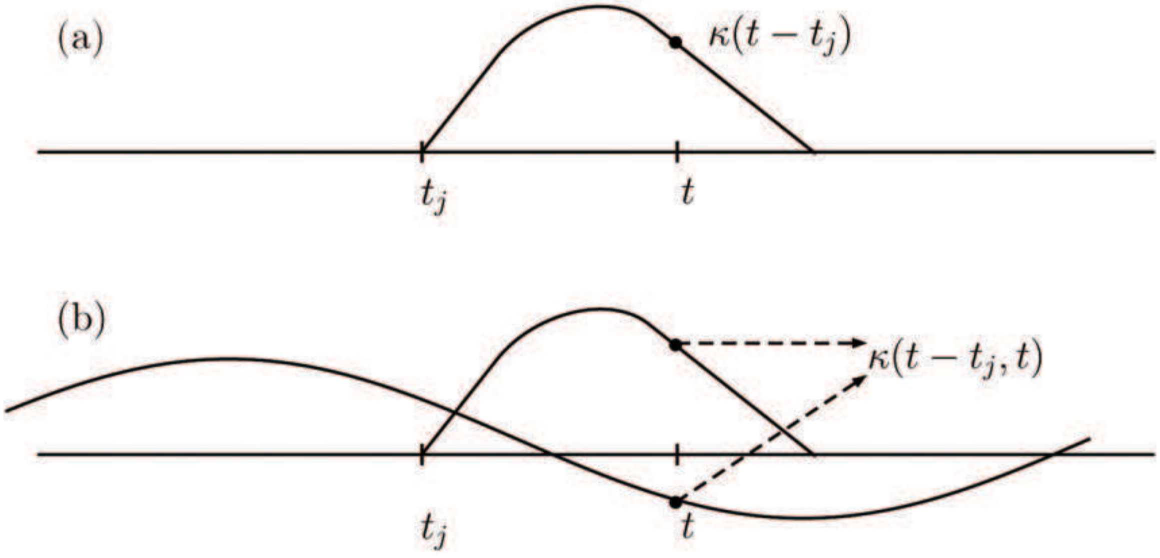

Figure 4.2: Comparison of the proposed modulated renewal process model and the globally non-stationary model.

We approximate the two dimensional coefficient surface by the product of B-spline basis functions and basis functions suitable for periodic functions such as the Fourier basis functions. That is

$$\kappa_i(u\Delta, t) = \sum_{j_1}\sum_{j_2}\alpha_{j_1,j_2}B_{j_1}(u\Delta)B^*_{j_2}(t). \tag{4.3}$$

The main challenge of this model is computational complexity due to the large number of parameters.

BIBLIOGRAPHY

Aertsen, A. M. H. J., Gerstein, G. L., Habib, M. K., and Palm, G. (1989). Dynamics of neuronal firing correlation: modulation of effective connectivity. *J. Neurophysiol.*, 61:900–17.

Akaike, H. (1973). *Information theory and an extension of the maximum likelihood principle*, chapter Proceeding of the 2nd International Symposium on Information Theory, pages 267–281.

Andersen, P. and Gill, R. D. (1982). Cox's regression model for counting processes: a large sample study. *Ann.Statist.*, 10:1100–20.

Barbieri, R., Quirk, M., Frank, L., Wilson, M., and Brown, E. (2001). Construction and analysis of non-poisson stimulus-response models of neural spiking activity. *J. Neurosci. Meth.*, 105:25–37.

Bartlett, M. (1966). *An introduction to stochastic processes.* Cambridge University Press.

Borisyuk, G., Borisyuk, R., Kirillov, A., Kovalenko, E., and Kryukov, V. I. (1985). A new statistical method for identifying interconnections between neuronal network elements. *Biol Cybern*, 52:301–306.

Brillinger, D. R. (1975a). Estimation of product densities. *Computer Science and Statistics: 8th Annual Symposium, Los Angeles*, pages 431–8.

Brillinger, D. R. (1975b). The identification of point process systems. *Annals of Probability*, 3:909–29.

Brillinger, D. R. (1976a). Estimation of the second-order intensities of a bivariate stationary point process. *Journal of the Royal Statistical Society: Series B*, 38:60–6.

Brillinger, D. R. (1976b). Identification of synaptic interactions. *Biol Cybern*, 22:213–28.

Brillinger, D. R. (1988). Maximum likelihood analysis of spike trains of interacting nerve cells. *Biol Cybern*, 59:189–200.

Brillinger, D. R. (1992). Nerve cell spike train data analysis: a progression of technique. *J. Am. Stat. Assoc.*, 87:260–71.

Brown, E., Kass, R., and Mitra, P. (2004). Multiple neural spike train data analysis: state-of-theart and future challenges. *Nat Neurosci*, 7:456–61.

Brown, E. N., Barbieri, R., Eden, U. T., and Frank, L. M. (2003). *Computational Neuroscience: A Comprehensive Approach*. Chapman & Hall/CRC Mathematical & Computational Biology.

Brown, E. N., Barbieri, R.and Ventura, V., Kass, R. E., and Frank, L. M. (2002). The time-rescaling theorem and its application to neural spike train data analysis. *Neural Comput.*, 14(2):325–346.

Candes, E. and Tao, T. (2007). The dantzig selector: Statistical estimation when p is much larger than n (with discussion). *Ann. Statist*, 35:2313–2351.

Chen, J. and Chen, Z. (2008). Extended bayesian information criteria for model selection with large model sspace. *Biometrika*, 95:759–771.

Cinlar, E. (1969). Markov renewal theory. *Advances in Applied Probability*.

Cook, R. J. and Lawless, J. F. (2007). *The Statistical Analysis of Recurrent Events*. Springer.

Cox, D. R. (1972a). Regression models and life-tables. *J.R. Statist. Soc. B*, 34:187–220.

Cox, D. R. (1972b). *The statistical analysis of dependencies in point processes*, pages 55–66. New York: Willey.

Daley, D. and Vere-Jones, D. (1988). *An Introduction to the Theory of Point Processes*. New York: Springer-Verlag.

de Boor, C. (2001). *A Practical Guide to Splines.* New York: Springer-Verlag.

Dean, J. and Ghemawat, S. (2008). Mapreduce: simplified data proprocess on large clusters. *Communications of the ACM*, 51(1):107–113.

Efron, B., Hastie, T.and Johnstone, I., and Tibshirani, R. (2004). Least angle regression. *Annals of Statistics*, 32:407–451.

Fan, J., Li, G., and L, R. (2005). *An Overview on Variable Selection for Survival Analysis*, chapter Contemporary Multivariate Analysis And Design of Experiments, pages 315–336.

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *J.Am. Statist. Assoc.*, 96:1348–1360.

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22.

Gerstein, G. and Perkel, D. (1969). Simultaneously recorded trains of action potentials: analysis and functional interpretation. *Science*.

Gerstein, G. L. and Perkel, D. H. (1972). Mutual temporal relationships among neuronal spike trains: statistical techniques for display and analysis. *Biophysical Journal*.

Gerstner, W. and Kistler, W. M. (2002). *Spiking Neuron Models: Single Neurons, Populations, Plasticity.* Cambridge University Press.

Goeman, J. (2010). L-1 penalized estimation in the cox proportional hazards model. *Biometrical Journal*, 52(1):70–84.

Haslinger, R., Pipa, G., and Brown, E. (2010). Discrete time rescaling theorem: determining goodness of fit for discrete time statistical models of neural spiking. *Neural Comput.*, 22(10):2477–2506.

Huang, J., Ma, S., Xie, H., and Zhang, C. H. (2009). A group bridge approach for variable selection. *Biometrika*, 96:339–355.

Johnson, A. and Kotz, S. (1970). *Distributions in statistics: Continuous univariate distributions*. New York: Wiley, 2 edition.

Kass, R. E. and Ventura, V. (2001). A spike-train probability model. *Neural Computation*, 13:1713–1720.

Lin, F. and Fine, J. P. (2009). Pseudomartingale estimating equations for modulated renewal process models. *J. R. Stat. Soc. B.*, 71:3–23.

Lin, F. C., Truong, Y. K., and Fine, J. P. (2013). Robust analysis of semiparametric renewal process models. *Biometrika*, 100(3):709–26.

Masud, M. S. and Borisyuk, R. (2011). Statistical technique for analysing functional connectivity of multiple spike trains. *J. Neurosci. Methods*, 196:201–19.

Mizuseki, K., Sirota, A., Pastalkova, E., and Buzsaki, G. (2009a). Multi-unit recordings from the rat hippocampus made during open field foraging.

Mizuseki, K., Sirota, A., Pastalkova, E., and Buzsaki, G. (2009b). Theta oscillations provide temporal windows for local circuit computation in the etorhinal-hippocampal loop. *Neuraon*, 62(2):267–80.

Oakes, D. (1981). Survival analysis: aspects of partial likelihood (with discussion). *Int. Statist. Rev.*, 49:235–64.

Oakes, D. and Cui, L. (1994). On semiparametric inference for modulated renewal processes. *Biometrika*, 81:83–90.

Okatan, M., Wilson, M., and Brown, E. N. (2000). Analyzing functional connectivity using a network likelihood model of ensemble neural spiking activity. *Neural Comput.*, 12:2621–53.

Perkel, D., Gerstein, G., and Moore, G. (1967). Neuronal spike trains and stochastic point processes ii. simultaneous spike trains. *Biophys J*, 7:419–40.

Pons, O. and de Turckheim, E. (1988). Cox's periodic regression model. *Ann. Statist.*, 16:678–693.

Reed, J. and Kaas, J. (2010). Statistical analysis of large-scale neuronal recording data. *Neural Networks*, 23:673–84.

Schumaker, L. (1980). *Spline Functions: Basic Theory*. Cambridge: Cambridge University Press.

Schwarz, G. (1978). Estimating the dimensions of a model. *The Annals of Statsitics*, 6:461–464.

Shevade, S. K. and Keerthi, S. S. (2003). A simple and efficient algorithm for gene selection using sparse logisitc regression. *Bioinformatics*, 19:2246–2253.

Snyder, D. and Miller, M. (1991). *Random Point Processes in Time and Space (2nd ed.)*. New York: Springer-Verlag.

Song, D., Chan, R. H. M., Marmarelis, V., Hampson, R., Deadwyler, S., and Berger, T. (2007). Nonlinear dynamic modeling of spike train transformations for hippocampal-cortical prostheses. *Biomedical Engineering, IEEE Transactions on*, 54(6):1053–1066.

Song, D., Wang, H., Tu, C. Y., Marmarelis, V. Z., Hampson, R. E., Deadwyler, S. A., and Berger, T. W. (2013). Identification of sparse neural functional connectivity using penalized likelihood estimation and basis functions. *Journal of Computational Neuroscience*, 35:335–357.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *J. R. Statist. Soc. B*, 58:267–288.

Tibshirani, R. (1997). The lasso method for variable selection in the cox model. *Statist. Med.*, 16:385–395.

Truccolo, W., Eden, U., Fellows, M., Donoghue, J., and Brown, E. (2005). A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects. *J Neurophysiol*, 93:1074–89.

Tsiatis, A. (1981). A large sample study of cox's regression model. *Ann. Statist.*, 9:93–108.

Tu, C. Y., Song, D., Breidt, F. J., Berger, T. W., and Wang, H. (2012). *Fuctional model selection for sparse bianry time series with multiple input.*, chapter Economic Time series: Modeling and seasonality, pages 477–497. Chapman and Hall/CRC.

Wahba, G. (1990). Spline models for obserational data. In *CBMS-NSF Regional Conference Series in Applied Matematics.* Society for Industrial and Applied Mathematics.

Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *J. R. Stat. Soc. B*, 68, Part 1:49–67.

Zhang, H. and Lu, W. (2007). Adaptive lasso for cox's proportional hazards model. *Biometrika*, 94 (3):691–703.

Zou, H. and Li, R. (2008). One-step sparse estimates in nonconcave penalized likeli- hood models (with discussion). *The Annals of Statistics*, 36:1509–1533.