



Swiss Federal Institute of Technology Zurich

Seminar for Statistics

Department of Mathematics

Master Thesis

Summer 2015

Paweł Morzywołek

Non-parametric Methods for Estimation of Hawkes Process for High-frequency Financial Data

Submission Date: August 1st 2015

Co-Advisers: Dr. Vladimir Filimonov
Prof. Dr. Didier Sornette
Adviser: Prof. Dr. Peter Bühlmann

Acknowledgements

I would like to express my deepest gratitude to Dr. Vladimir Filimonov for his guidance of my work and granting me the access to his code on the Hawkes model. I sincerely thank Prof. Didier Sornette and Prof. Peter Bühlmann, who have allowed me to work on such an interesting topic. Special thanks go to Spencer Wheatley, who provided me his implementation of the EM Algorithm and was always ready to discuss issues regarding the Hawkes model and statistical analysis. I would like to thank Zhuli He for allowing me to use his implementation of the INAR method. I would like to thank Stefan Rustler for providing me a figure from his Master Thesis. Special thanks for Jakub Szymanek for his IT support and Sarah Crook for language corrections. I would also like to thank Philip Berntsen, Ioannis Georgiadis, Robert Kuert, Jeroen Rozendaal and other master students for a pleasant atmosphere in the office. At last I would like to express my sincere gratitude to my parents for their constant support.

Abstract

Due to its ability to represent clustered data well the popularity of the self-excited Hawkes model has steadily grown in recent years. After originally being applied for earthquake prediction it has been also used to anticipate flash crashes in finance, epidemic type of behavior in social media such as Twitter and YouTube or criminality outbursts in big cities.

The aim of this work is to conduct a comprehensive comparison study of the existing non-parametric techniques for estimation of the Hawkes model, which without making any *a priori* assumptions on the correlation structure of the observables, provide us insights into the data. To best of my knowledge such work has not been done so far. The first considered method is the widely used in non-parametric statistics EM Algorithm, adjusted to the case of a Hawkes process. The second calibration procedure is based on the estimation of a conditional expectation of the Hawkes model's counting process and then solving a Wiener-Hopf type integral equation to obtain the kernel function of the model. The last estimation technique uses representation of the Hawkes model as an integer-valued autoregressive model and subsequently applies tools from theory of time series to obtain parameters of the model.

The methods were tested on synthetic data generated from the Hawkes model with different kernels and different parameters. I investigated how the size of the sample and the overlapping of point clusters influences performance of different estimation methods. When conducting the analysis, I did not restrict myself only to the case of the most commonly used exponential and power law kernels, but also considered less typical step and cut-off kernels. After the comparison on synthetic data has been accomplished I proceeded with empirical data analysis. For this purpose I tested the estimation methods on the high-frequency data of price changes of E-mini S&P 500 and Brent Crude futures contracts.

Contents

1	Introduction	3
2	The Hawkes Process	5
2.1	Theoretical Background	5
2.2	Kernels	6
2.3	Point Clusters	7
2.4	Data Simulation	9
3	Estimation of the Kernel	11
3.1	Parametric Estimation Methods	11
3.2	Non-parametric Estimation Methods	11
3.2.1	WH Method	12
3.2.2	EM Algorithm	16
3.2.3	INAR Method	19
4	Comparison of the Estimation Methods on Synthetic Data	21
4.1	Performance of the Estimation Methods with respect to Sample Size	21
4.1.1	Separated Clusters	23
4.1.2	Overlapping Clusters	23
4.2	Performance of the Estimation Methods with respect to the Overlap of Point Clusters	28
4.3	Performance of the Estimation Methods with respect to the Branching Ratio	28
4.4	Slowly Decaying Memory Kernel	33
4.5	Performance of Estimation Methods on Other Types of Kernels	39
5	Comparison of the Estimation Methods on Empirical Data	47
5.1	Data Source	47
5.2	Data Preparation - Randomization Procedure	48
5.3	Residual Analysis	48
5.4	Results on Empirical Data	48
6	Summary	73
	Bibliography	75

Chapter 1

Introduction

For centuries people have had great interest in trading and finance, trying to predict future outcomes and price movements. Stock markets, being a great source of data and randomness, have inspired many generations of mathematicians to try to better understand mechanisms behind the price formation. One of the first attempts to apply advanced mathematical tools in finance was a famous PhD Thesis "Théorie de la spéculation" of Louis Bachelier ([Bachelier, 1900](#)). Based on the results of Bachelier, Paul Samuelson and Eugene Fama defined a concept of an efficient market ([Samuelson, 1965](#)), ([Fama, 1970](#)). The idea is that all available information about the assets is incorporated automatically in the prices, hence they reflect the true value. The Efficient Market Hypothesis (EMH) has many proponents among financial economists, but there have been a lot of criticisms towards it. The doubts come from the fact that many events, such as financial bubbles, trading herding, abrupt price jumps, just to name few, are observed in the financial markets even though their presence is disclosed under EMH.

An alternative approach to understand how financial markets work is a concept of market's reflexivity popularized by George Soros ([Soros, 2003](#)). This notion captures the idea of interplay of exogenous events (economic fundamentals) with endogenous events (price changes due to previous price changes). That allows us to speak about the level of endogeneity of a market, which gives us information about what part of all price movements is endogenous in their nature. The question that arises is how to measure the endogeneity of a market. ([Filimonov and Sornette, 2012](#)) suggest methodology based on fitting a Hawkes model to the financial data. Due to its self-exciting behavior the Hawkes model, being a generalization of a Poisson process, seems particularly suitable to model financial data with its clustered volatility. The model has already been successfully used in a number of different application areas such as seismology ([Ogata, 1998](#)), criminology (Mohler, Short, Brantingham, Schoenberg, and Tita, [2012](#)) and social networks ([Crane and Sornette, 2008](#)).

One of the biggest challenges in analyzing financial markets in recent years has been the attempt to capture the activity of high-frequency and algorithmic traders. This fairly new phenomenon was observed to grow intensively since the technological development came to stock markets. This new way of trading allows making huge amounts of transactions in extremely short time periods without any human interaction. Because of the fact that algorithms use trading strategies based fully on analysis of previous price changes, while practically fully neglecting economic fundamentals and due to huge volumes of their transactions, we can almost completely associate endogenous price movements with activity of

algorithmic and high-frequency traders. In this way, once we manage to quantify the level of endogeneity of the market, we can try to address a highly interesting question of what part of all transactions done in the stock markets nowadays is executed by algorithms.

Fitting a Hawkes model to financial data provides some problems, to a large extent because of the concerns about correlation structure of the price movements. The question of whether the price dependence structure is of a short-term or long-term nature seems still open and not easy to answer. In order to avoid making any prior assumptions, which could influence the outcome of the analysis, there has been a growing need for good and robust non-parametric estimation methods. The main goal of this thesis is to test a variety of available non-parametric methods in different circumstances and judge their advantages and application potential.

After a brief introduction of the Hawkes model and a presentation of the considered non-parametric estimation methods, I conduct a comparison study on synthetic data generated from most widely used in applications exponential and power law kernels. I further extend my work to the case of less regular kernels such as step and cut-off kernel. Once the analysis on synthetic data is accomplished I proceed by applying the methods on high-frequency data of E-Mini S&P 500 and Brent Crude futures contracts. I believe the obtained results of the model's fit to Brent Crude futures price changes look very promising. At this point I shall stress that my focus was more on the methodological side of the work, therefore the results on empirical data should be considered only as a preliminary analysis.

Chapter 2

The Hawkes Process

The Hawkes process was introduced for the first time in (Hawkes, 1971). Its original application field was seismology (see (Ogata, 1998), (Helmstetter and Sornette, 2003)). The idea was to provide a model incorporating self-exciting behavior, which would allow to model parallelly randomly occurring main shocks (exogenous events) and the aftershocks (endogenous events) caused by them.

2.1 Theoretical Background

A linear Hawkes process is a point process $\{t_i\}_{i \in \mathbb{N}_{>0}}$ with conditional intensity given by

$$\lambda(t|\mathcal{F}_{t-}) = \mu(t) + \int_{-\infty}^t \varphi(t-s)dN(s). \quad (2.1.1)$$

The conditional intensity is defined as

$$\lambda(t|\mathcal{F}_{t-}) := \lim_{h \rightarrow 0} \frac{\mathbb{E}[N(t+h) - N(t)|\mathcal{F}_{t-}]}{h} \quad (2.1.2)$$

for a given counting process N and the filtration \mathcal{F}_{t-} . The counting process is of the form

$$N(t) = \max\{i : t_i \leq t\} \quad (2.1.3)$$

and the filtration given by

$$\mathcal{F}_{t-} = \{t_1, \dots, t_i : \forall i < N(t)\} \quad (2.1.4)$$

represents the history of the process N until time t . Intuitively the conditional intensity is an infinitesimal expected rate at which the events occur around time t given the history of the process N before t . The background intensity μ is responsible for accounting for the arrival of exogenous events and the kernel function φ , satisfying causality condition

$\varphi(t) = 0$ for $t < 0$, is determining the correlation properties of the process. Furthermore, we define a branching ratio

$$n := \int_0^\infty \varphi(t) dt. \quad (2.1.5)$$

Rewriting the differential of the counting process as a sum of delta functions

$$dN(t) = \sum_{t_i < t} \delta(t - t_i) \quad (2.1.6)$$

we can obtain the following form of the conditional intensity

$$\lambda(t|\mathcal{F}_{t-}) = \mu(t) + n \sum_{t_i < t} h(t - t_i), \quad (2.1.7)$$

where

$$h(t) := \varphi(t)/n \quad (2.1.8)$$

is called a bare kernel. Let us notice that bare kernel is a probability density function (PDF). The Hawkes process is stationary for $n < 1$. Assuming stationarity and taking $\mu(t) = \mu$ to be constant over time we can calculate an average total intensity

$$\Lambda := \mathbb{E}[\lambda(t|\mathcal{F}_{t-})] = \mathbb{E}\left[\mu(t) + \int_{-\infty}^t \varphi(t-s)dN(s)\right] = \mu + \Lambda \int_0^\infty \varphi(\tau)d\tau \quad (2.1.9)$$

and this implies

$$\Lambda = \frac{\mu}{1-n}. \quad (2.1.10)$$

The branching ratio n is crucial for analyzing the dynamics of the Hawkes process. It is the average number of first-generation daughters of a single mother. We can notice that when $n = 0$ we just obtain an inhomogeneous Poisson process. This means that we can see the Hawkes process as a generalization of a Poisson process dependent on the time and the history of the process. In case of $n > 1$ the intensity may explode. The critical case $n = 1$ separates subcritical ($n < 1$) and supercritical ($n > 1$) regimes. In further considerations I will concentrate on subcritical case.

2.2 Kernels

In many applications we have an intuition (based on empirical observation) of how the kernel should look like or what properties should it have. The most widely used are exponential and power law kernel. The power law kernel defined as

$$\varphi_{pow}(t) = \frac{n\theta c^\theta}{(t+c)^{1+\theta}} \chi(t) \quad (2.2.1)$$

is particularly often used in geophysical applications due to the aftershock rates dependence called the Omori law (Utsu and Ogata, 1995). The function χ is the Heaviside function, which guarantees the causality. (Kagan and Knopoff, 1981) suggested using modified version of the power law kernel - so called cut-off kernel - given by

$$\varphi_{cut}(t) = \frac{n\epsilon\tau_0}{t^{1+\epsilon}} \chi(t - \tau_0). \quad (2.2.2)$$

In order to facilitate the approximation of (2.2.2) an approximation as a sum of exponential functions given by

$$\varphi_{se} = \frac{n}{Z} \left[\sum_{i=0}^{M-1} \frac{1}{\xi_i^{1+\epsilon}} \exp\left(-\frac{t}{\xi_i}\right) - S \exp\left(-\frac{t}{\xi_{-1}}\right) \right] \quad (2.2.3)$$

has been suggested in (Hardiman, Bercot, and Bouchaud, 2013), where $\xi_i = \tau_0 m^i$ and the coefficients S and Z are chosen so that $\varphi_{se}(0) = 0$ and $\int_0^\infty \varphi_{se}(t) dt = n$. The values of M and m has been chosen empirically by the authors to be $M = 15$ and $m = 5$. The second most often occurring kernel - exponential - is defined as

$$\varphi_{exp}(t) = \frac{n}{\tau} \exp\left(-\frac{t}{\tau}\right) \chi(t). \quad (2.2.4)$$

There has been an intensive scientific discussion which kernel represents best the dependencies between the price changes in financial markets. Some researchers claim that the price correlation is of the long-term nature and hence should be modeled using power law kernel (Hardiman et al., 2013). Other argue that due to non-stationarity of financial data the dependence should be only considered on small time windows and therefore short-memory exponential kernel is better suited for this purpose (Filimonov and Sornette, 2012), (Filimonov and Sornette, 2015). Recently (Rambaldi, Pennesi, and Lillo, 2014) suggested using the double exponential kernel defined as

$$\varphi_{de}(t) = \left(\alpha_1 \exp\left(-\frac{t}{\tau_1}\right) + \alpha_2 \exp\left(-\frac{t}{\tau_2}\right) \right) \chi(t), \quad (2.2.5)$$

where α_i and τ_i for $i = 1, 2$ are constant parameters, providing very good fits of the model to the FX financial data. This ambiguity about choice of the kernel function shows great need for further development of non-parametric estimation methods, which without any *a priori* assumptions would be able to provide us more insights into the structure of the data.

2.3 Point Clusters

Probably the most important feature of the Hawkes process is the fact that due to its self-exciting nature it generates point clusters, where each one originates from a single immigrant event. Depending on the choice of the parameters of the model clusters can be fairly well-separated or overlap on each other. In what follows, I will try to analyze how the overlapping of clusters influences the estimation performance of different estimation

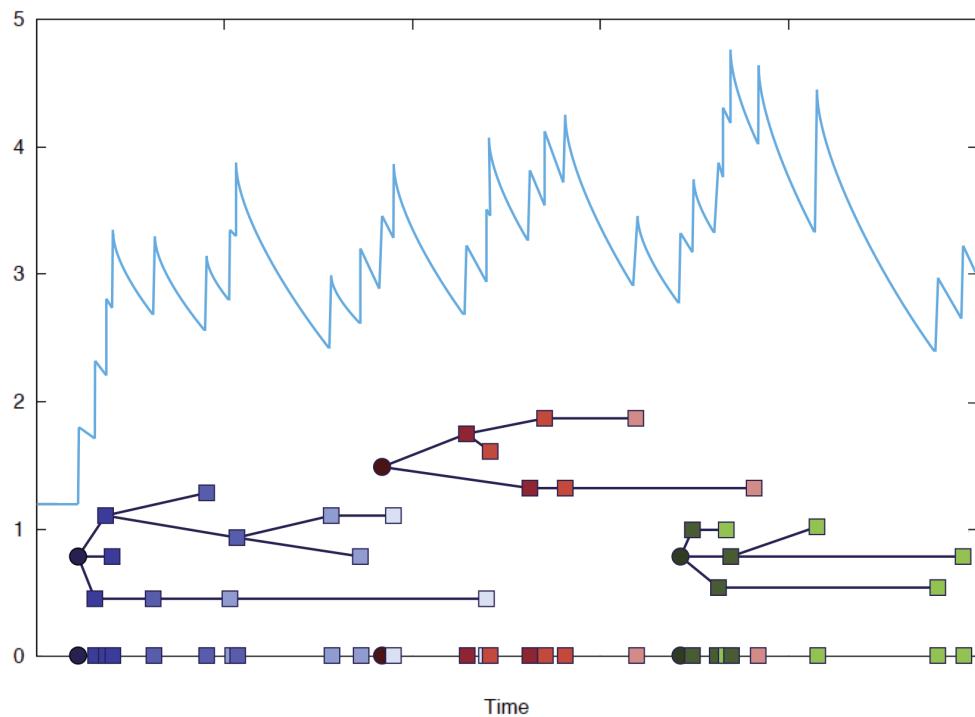


Figure 2.1: Plot of the conditional intensity of a Hawkes process together with a presentation of the cluster structure of the model. The points on the x axis are the realization of the Hawkes model as we would observe it. The picture above shows how we can imagine Hawkes process with its cluster structure. Different colors (blue, red, green) represent different clusters. Black dots mean immigrant events. Different shades of the descendants' colors show different generations of the events. [Reproduced with permission from Stefan Rustler.]

methods in case of Hawkes process with exponential kernel. In order to do that I will define below the clusters separation measure. Before I do that we have to introduce another type of kernel, namely renormalized kernel. The bare kernel that we have seen above is nothing else than the probability of a mother event triggering a daughter event. In some cases it is easier though to capture the responses of the points in the cluster with respect to the immigrant event (the earliest mother of the cluster) than the response with respect to the direct mother. The kernel function R , called renormalized kernel, is a function describing those responses. The bare kernel and the renormalized kernel are related by the following integral equation

$$h(t) = R(t) - n \int_0^t h(t-s)R(s)ds. \quad (2.3.1)$$

In the case of exponential kernel (2.2.4) the renormalized kernel is given by

$$R(t) = \frac{1}{\tau} \exp\left(-\frac{t(1-n)}{\tau}\right). \quad (2.3.2)$$

In this case it is easy to see that renormalized kernel is not a PDF. In general we have

$$\int_0^\infty R(t)dt = \frac{1}{1-n}. \quad (2.3.3)$$

For derivation of this result and further discussion of renormalized kernel see ([Rustler and Filimonov, 2014](#)). In order to analyze how much the clusters overlap we have to know the average distance between immigrants and the average length of the cluster. Since immigrants are generated from a homogenous Poisson process with intensity μ it is a well-known fact that the average distance between points is $\frac{1}{\mu}$. For more detailed presentation consult ([Daley and Vere-Jones, 2003](#)). Moreover, since the occurrence of descendants with respect to the earliest mother is described by (2.3.2), hence the average length of the cluster is given by

$$\int_0^\infty t(1-n)R(t)dt = \int_0^\infty \frac{(1-n)}{\tau} t \exp\left(-\frac{(1-n)}{\tau}t\right) dt = \frac{\tau}{1-n}, \quad (2.3.4)$$

where additional factor $(1-n)$ accounts for the fact that renormalized kernel is not a PDF. Now we can define clusters separation measure κ to be the ratio of average length of the cluster and average distance between immigrants. This means that $\kappa < 1$ will indicate well-separated point clusters and $\kappa \geq 1$ will be a sign of overlapping clusters. In case of exponential kernel we have

$$\kappa = \frac{\tau/(1-n)}{1/\mu} = \frac{\mu\tau}{1-n}. \quad (2.3.5)$$

2.4 Data Simulation

There are two currently known ways to generate Hawkes process. The first technique called the thinning method has been described in ([Lewis and Shedler, 1978](#)) and has

been originally used for simulating inhomogeneous Poisson processes. It was then applied in ([Ogata, 1981](#)) to the case of a Hawkes process. The idea is to generate the data points t_1, \dots, t_N from a homogenous Poisson process with intensity λ_{maj} being majorant to the conditional intensity $\lambda(t)$ of the Hawkes process that we want to generate, i.e. $\lambda_{maj} \geq \lambda(t) \ \forall t$. Subsequently we apply the acceptance-rejection method to the previously generated points. For a given point t_i ($i \in \{1, \dots, N\}$) we accept it with probability p_i given by

$$p_i = \frac{\lambda(t_i)}{\lambda_{maj}} \quad (2.4.1)$$

and otherwise we discard it from the sample. The numerical complexity of the method is $\mathcal{O}(N^2)$, which makes it inefficient in case of a long time series of near-critical Hawkes process (see ([Filimonov and Sornette, 2015](#))). Another disadvantage of this method is that the branching structure of the process is unknown, i.e. we do not know which event is an immigrant and which event has been triggered by one of the previous events. A more efficient way to generate a Hawkes process is to simulate all the clusters in parallel generation by generation. This means that we simulate first all immigrant events (0^{th} generation of the cluster) from a homogenous Poisson process with the intensity $\lambda(t)$ equal to the background intensity μ of a Hawkes process. Once we have obtained all immigrants we proceed with construction of next generations of the clusters. For a given immigrant with a timestamp t_i we generate the first generation of its cluster from an inhomogeneous Poisson process with the intensity $\lambda_i(t) = \varphi(t - t_i)$. Analogously once we have constructed the k^{th} generation of the cluster we simulate the points of the $(k+1)^{th}$ generation from inhomogeneous Poisson processes with the intensities $\lambda_{k,i}(t) = \varphi(t - t_{k,i})$, where $t_{k,i}$ is a timestamp of the i^{th} point in the k^{th} generation of the cluster. We do this procedure for all the clusters in parallel, until there is no more offspring. The method has been presented in ([Møller and Rasmussen, 2005](#)) and ([Møller and Rasmussen, 2006](#)). The numerical complexity goes as $\mathcal{O}(\mu T K)$, where T is a size of simulation time window and K is the number of generations in the window (see ([Filimonov and Sornette, 2015](#))). An additional advantage is the fact that we are able to reconstruct whole branching structure of the process. In the simulations that follow, I have used the latter simulation method.

Chapter 3

Estimation of the Kernel

3.1 Parametric Estimation Methods

There are well-developed methods for parametric estimation of the kernel, where the most widely used is Maximum Likelihood Estimation ([Ogata, 1998](#)). The log-likelihood function of the Hawkes process is given by

$$\log L(t_1, \dots, t_N) = \sum_{i=1}^N \log \lambda(t_i | \mathcal{F}_{t_{i-}}) - \int_0^T \lambda(t | \mathcal{F}_{t-}) dt, \quad (3.1.1)$$

where $t_1, \dots, t_N \in (0, T]$ are observed timestamps. The parameters of the Hawkes model can be computed numerically by maximization of the log-likelihood function for both exponential and power law kernel.

3.2 Non-parametric Estimation Methods

As already mentioned, there are situations when we do not have an idea of the shape of the kernel function or we would like to justify our intuition regarding the kernel function using a non-parametric estimation. The non-parametric estimation methods used in the context of Hawkes process are the class of EM Algorithms ([Dempster, Laird, and Rubin, 1977](#)), ([Marsan and Lengliné, 2008](#)), ([Veen and Schoenberg, 2008](#)), ([Lewis and Mohler, 2011](#)), ([Wheatley, Filimonov, and Sornette, 2014](#)), the estimation method based on solving of Wiener-Hopf type integral equation relating the kernel function of the Hawkes process with conditional expectation of its counting process (further "WH method") ([Bacry, Dayri, and Muzy, 2012](#)), ([Bacry and Muzy, 2014a](#)), ([Bacry and Muzy, 2014b](#)), ([Bacry, Jaisson, and Muzy, 2014](#)) and an estimation procedure using representation of the Hawkes process as an integer-valued autoregressive model of degree p (INAR(p) model) and then applying tools from time series theory to estimate the parameters (further "INAR method") ([Kirchner, 2014](#)).

3.2.1 WH Method

This method has been originally presented in (Bacry et al., 2012) in case of a symmetric multivariate Hawkes process. It has been further developed for general Hawkes process in (Bacry and Muzy, 2014a) and (Bacry and Muzy, 2014b). The extension of the procedure for the case of slowly decaying kernels has been done in (Bacry et al., 2014). This estimation method uses the conditional expectation of the counting process of the Hawkes model, which is fairly easy to estimate, and its relation to the kernel of the process given by a Wiener-Hopf integral equation.

Definition 3.2.1.1. (*Def. 2.3. in (Bacry et al., 2014)*) *The conditional expectation g of counting process of the Hawkes model is defined as the non-singular part of the measure $\mathbb{E}[dN_t|dN_0 = 1]$*

$$g(t)dt := \mathbb{E}[dN_t|dN_0 = 1] - \delta(t) - \Lambda dt. \quad (3.2.1)$$

The following proposition links the conditional expectation and the kernel of the Hawkes process.

Proposition 3.2.1.2. (*Prop. 2.4. in (Bacry et al., 2014)*) *Given the conditional expectation g of the counting process of the Hawkes model, its kernel is the unique solution of the Wiener-Hopf integral equation*

$$g(t) = \varphi(t) + g * \varphi(t) \quad \forall t > 0. \quad (3.2.2)$$

Let us point out that for all $t < 0$ we have $g(t) = g(-t)$.

Numerical Estimation of the Kernel

The estimation procedure consists of the following few steps:

i.) **Estimate empirically the average intensity Λ .**

Estimation of Λ is an easy thing to do. We only have to count the empirical number of events in a unit time, i.e.

$$\hat{\Lambda} = \frac{\#\{t_i : t_i \in [0, T]\}}{T}. \quad (3.2.3)$$

ii.) **Compute an estimate \hat{g} of the conditional expectation g .**

Given the timestamps of the events $\{t_k\}_{1 \leq k \leq N}$ we want to estimate empirically the conditional expectation g . We can do that following two approximation steps. First we have to choose the time grid $\{x_l\}_{1 \leq l \leq K}$ in a way that allows us to approximate $g\left(\frac{x_l+x_{l+1}}{2}\right)$ by $\frac{1}{x_{l+1}-x_l} \int_{x_l}^{x_{l+1}} g(s)ds$. Furthermore, we can approximate the integral

$$\int_{x_l}^{x_{l+1}} g(s)ds = \int_{x_l}^{x_{l+1}} \mathbb{E}[dN_s|dN_0 = 1] - \delta(s) - \Lambda ds \quad (3.2.4)$$

with a histogram of all interevent times

$$\frac{1}{n} \sum_{k=1}^n \sum_{k'=1, k' \neq k}^n \mathbb{1}_{\{t_k - t_{k'} \in [x_l, x_{l+1}]\}} - \Lambda \times (x_{l+1} - x_l). \quad (3.2.5)$$

Combining both approximation steps together we finally obtain:

$$\hat{g}\left(\frac{x_l + x_{l+1}}{2}\right) = \frac{1}{x_{l+1} - x_l} \frac{1}{n} \sum_{k=1}^n \sum_{k'=1, k' \neq k}^n \mathbb{1}_{\{t_k - t_{k'} \in [x_l, x_{l+1}]\}} - \Lambda \quad (3.2.6)$$

and interpolate affinely afterward.

iii.) **Solve the integral equation using a quadrature scheme to obtain $\hat{\varphi}$.**

Under the assumption that the kernels are piecewise affine on $[x_k, x_{k+1}]$, i.e.

$$\varphi(t) = \varphi(x_k) + \frac{t - x_k}{x_{k+1} - x_k} (\varphi(x_{k+1}) - \varphi(x_k)) \quad (3.2.7)$$

we get the following system of equations

$$\begin{aligned} \hat{g}(x_n) &= \hat{\varphi}(x_n) + \sum_{k=0}^{K-1} \hat{\varphi}(x_k) \int_{x_n - x_{k+1}}^{x_n - x_k} \hat{g}(u) du \\ &\quad + \sum_{k=0}^{K-1} \frac{(\hat{\varphi}(x_{k+1}) - \hat{\varphi}(x_k))(x_n - x_k)}{x_{k+1} - x_k} \int_{x_n - x_{k+1}}^{x_n - x_k} \hat{g}(u) du \\ &\quad - \sum_{k=0}^{K-1} \frac{\hat{\varphi}(x_{k+1}) - \hat{\varphi}(x_k)}{x_{k+1} - x_k} \int_{x_n - x_{k+1}}^{x_n - x_k} u \hat{g}(u) du. \end{aligned} \quad (3.2.8)$$

Now we just have to solve the system in order to obtain the values of $\hat{\varphi}$ in grid points and then interpolate the solution.

iv.) **Calculate an estimate $\hat{\mu}$ of the background intensity μ .**

Finally we can obtain the estimate \hat{n} of parameter n by integrating the kernel $\hat{\varphi}$ and then calculating the estimate $\hat{\mu}$ of background intensity μ using (2.1.10).

Choice of the Grid $\{x_l\}_{1 \leq l \leq K}$.

The appropriate choice of the grid is an important part of the estimation process, since it allows the approximation of $g\left(\frac{x_l + x_{l+1}}{2}\right)$ by $\frac{1}{x_{l+1} - x_l} \int_{x_l}^{x_{l+1}} g(s) ds$ (in step ii. of above numerical estimation) and representation of \hat{g} using system of equations (step iii.). In case of a short memory kernel (i.e. exponentially decaying kernel) we can choose a uniform grid with the grid-size $\delta < 1$

$$\{x_l\}_{1 \leq l \leq K} = \{0, \delta T, 2\delta T, \dots, T\}. \quad (3.2.9)$$

In case of a slowly decaying kernel (i.e. power law kernel) we have to be more cautious with our choice. The problem is that we want to have a fine grid at the low time scale in order to have a good estimation of the head of the distribution, but at the same time we would like to avoid that the tail of the distribution varies too much on the large time

scale. A solution to this issue is setting a uniform time grid on interval $[0, T_{min}]$ and a log-uniform time grid on $[T_{min}, T]$:

$$\{x_l\}_{1 \leq l \leq K} = \{0, \delta T_{min}, 2\delta T_{min}, \dots, T_{min}, T_{min}e^\delta, T_{min}e^{2\delta}, \dots, T\}. \quad (3.2.10)$$

In order to choose the appropriate grid-size, I will apply AIC (Akaike Information Criterion). It is a standard statistical tool, which allows us to choose a model with best performance and least complexity at the same time ([Akaike, 1974](#)).

Analytical Solution of the Integral Equation in the case of an Exponential Kernel

In case of an exponential kernel ([\(2.2.4\)](#)) we can solve the integral equation ([\(3.2.2\)](#)) analytically in order to get the explicit formulation of g (similar computation as in ([Hawkes, 1971](#))):

$$\begin{aligned} g(t) &= \varphi(t) + g * \varphi(t) = \varphi(t) + \int_{-\infty}^t \varphi(t-s)g(s)ds \\ &= \varphi(t) + \int_0^t \varphi(t-s)g(s)ds + \int_0^\infty \varphi(t+s)g(s)ds \quad \forall t > 0 \end{aligned} \quad (3.2.11)$$

where in the last step I have used the symmetry of g . Applying Laplace transform we obtain by Convolution Theorem

$$\begin{aligned} \mathcal{L}\{g(t)\}(s) &= \mathcal{L}\{\varphi(t)\}(s) + \mathcal{L}\left\{\int_0^t \varphi(t-x)g(x)dx\right\}(s) + \mathcal{L}\left\{\int_0^\infty \varphi(t+x)g(x)dx\right\}(s) \\ &= \mathcal{L}\{\varphi(t)\}(s) + \mathcal{L}\{\varphi(t)\}(s) \times \mathcal{L}\{g(t)\}(s) + \mathcal{L}\left\{\int_0^\infty \varphi(t+x)g(x)dx\right\}(s) \end{aligned} \quad (3.2.12)$$

We can compute easily the last term of the sum in case of exponentially decaying kernel.

$$\begin{aligned} \mathcal{L}\left\{\int_0^\infty \varphi(t+x)g(x)dx\right\}(s) &= \mathcal{L}\left\{\int_0^\infty \frac{n}{\tau} \exp\left(-\frac{t+x}{\tau}\right) g(x)dx\right\}(s) \\ &= \int_0^\infty \exp(-st) \int_0^\infty \frac{n}{\tau} \exp\left(-\frac{t+x}{\tau}\right) g(x)dxdt \\ &= \int_0^\infty \frac{n}{\tau} \exp\left(-\left(s + \frac{1}{\tau}\right)t\right) dt \times \int_0^\infty \exp\left(-\frac{x}{\tau}\right) g(x)dx \\ &= \frac{n}{s\tau + 1} \mathcal{L}\{g(t)\}\left(\frac{1}{\tau}\right) \end{aligned} \quad (3.2.13)$$

Substituting this result into equation ([\(3.2.12\)](#)) and also using the fact that $\mathcal{L}\left\{\frac{n}{\tau} \exp\left(-\frac{t}{\tau}\right)\right\}(s) = \frac{n}{s\tau + 1}$ we get

$$\mathcal{L}\{g(t)\}(s) = \frac{n}{s\tau + 1} \left(1 + \mathcal{L}\{g(t)\}(s) + \mathcal{L}\{g(t)\}\left(\frac{1}{\tau}\right)\right). \quad (3.2.14)$$

Hence we have

$$\mathcal{L}\{g(t)\}(s) = \frac{n}{s\tau + 1 - n} \left(1 + \mathcal{L}\{g(t)\} \left(\frac{1}{\tau} \right) \right). \quad (3.2.15)$$

Now setting $s = \frac{1}{\tau}$ we can solve this equation to get

$$\mathcal{L}\{g(t)\} \left(\frac{1}{\tau} \right) = \frac{n}{2(1-n)}. \quad (3.2.16)$$

Substituting this result back to the equation (3.2.15) we get

$$\mathcal{L}\{g(t)\}(s) = \frac{n(2-n)}{2(1-n)(s\tau + 1 - n)}. \quad (3.2.17)$$

Finally we can obtain the conditional expectation g applying the Inverse Laplace transform

$$g(t) = \frac{(2-n)}{2(1-n)} \frac{n}{\tau} \exp \left(-(1-n) \frac{t}{\tau} \right). \quad (3.2.18)$$

Comparison of Analytical and Numerical Solution in the case of an Exponential Kernel

Let us compare the estimates of the conditional expectation \hat{g} and the kernel function $\hat{\varphi}$ provided by the numerical algorithm described above with analytical expressions for g and φ in the case of an exponential kernel. Figure 3.1 presents the obtained result. The data points have been simulated from Hawkes process with an exponential kernel with parameters $[\mu, n, \tau] = [0.05, 0.8, 0.3]$ on the time window $[0, 10^6]$. The sample size was 250957. From the plot we can see that numerically estimated conditional expectation \hat{g} matches perfectly the analytical solution of the integral equation (3.2.2) given by (3.2.18). Similarly the estimate $\hat{\varphi}$ obtained by solving system of equations (3.2.8) fit perfectly the plot of the kernel (2.2.4).

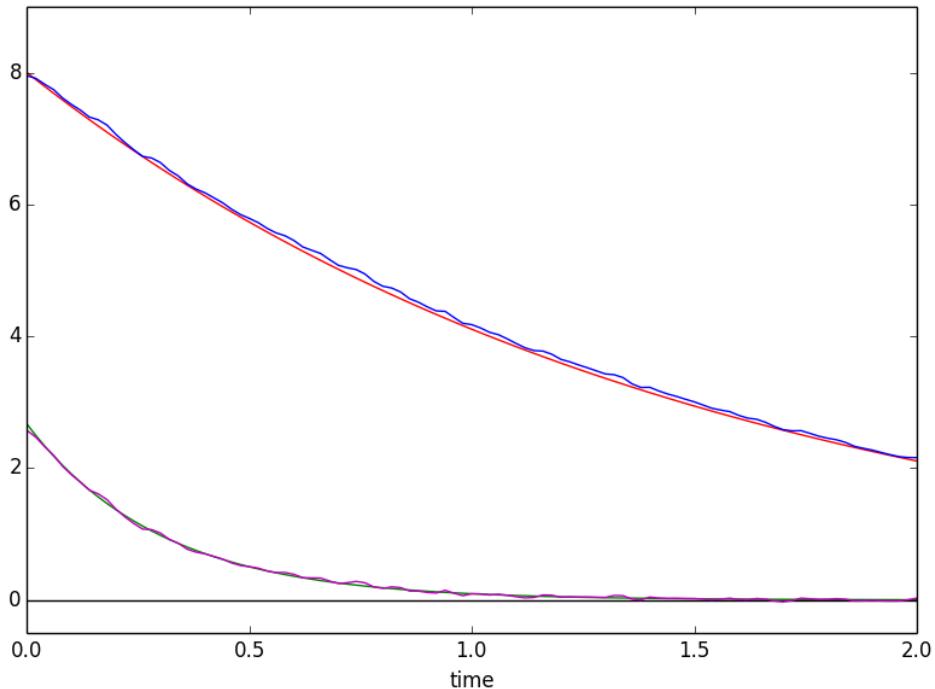


Figure 3.1: Plot comparing numerical estimation with analytical solution of the integral equation (3.2.2) in case of exponential kernel. The exponential kernel function φ (2.2.4) (green line), analytical solution g (3.2.18) of the integral equation (3.2.2) (red line), estimated conditional expectation \hat{g} (blue line) and estimated kernel function $\hat{\varphi}$ (magenta line) are given.

3.2.2 EM Algorithm

EM Algorithm - General Introduction

An Expectation Maximization (EM) Algorithm has been introduced in (Dempster et al., 1977). Its application in the case of Hawkes process has been first described in (Marsan and Lengliné, 2008) and then further developed in (Veen and Schoenberg, 2008) and (Lewis and Mohler, 2011). EM Algorithm estimation in the case of a generalized Hawkes process with renewal immigration has been done in (Wheatley et al., 2014). The EM Algorithm is an iterative method to compute numerically Maximum Likelihood Estimator. It is used if the part of data is missing or in case of missing data, if known, would allow to compute a likelihood more easily. Let us denote observed data as X and missing data (sometimes also called latent data) as Z . X and Z together are complete data. Furthermore, we will consider the function $L(\theta; X, Z)$, which is called the complete data likelihood function. Maximization of $\log L(\theta; X, Z)$ with respect to θ will provide us an estimate of a parameter. Since Z is not known we will account for missing data probabilistically using an iterative procedure:

- i.) Choose initial estimate of a parameter $\hat{\theta}^{[0]}$.
- ii.) **E-Step** - Calculate the expected value of the complete data log-likelihood with respect to conditional distribution of missing data Z , given the observed data X and

current estimate of parameter $\hat{\theta}^{[m]}$:

$$Q(\theta|X, \hat{\theta}^{[m]}) = \mathbb{E}_{Z|X, \hat{\theta}^{[m]}} [\log L(\theta; X, Z)]. \quad (3.2.19)$$

- iii.) **M-Step** - Maximize the expected likelihood (3.2.19) in order to get a new estimate of the parameter θ :

$$\hat{\theta}^{[m+1]} = \arg \max_{\theta} Q(\theta|X, \hat{\theta}^{[m]}). \quad (3.2.20)$$

- iv.) Proceed by iterating E and M steps until the parameter estimates stabilize.

A natural question at this point would be why does EM Algorithm actually work, since in each iteration we increase the value of function $Q(\theta|X, \hat{\theta}^{[m]})$, even though we are really interested in maximization of $\log L(\theta; X, Z)$. As it has been shown in (Dempster et al., 1977) the latter follows the former, hence we have

$$\log L(\hat{\theta}^{[m+1]}; X, Z) \geq \log L(\hat{\theta}^{[m]}; X, Z) \quad \forall m \in \mathbb{N}, \quad (3.2.21)$$

which means that we improve the estimate of MLE in each iteration.

EM Algorithm for Hawkes Process

Since we are interested in a non-parametric estimation we do not assume that kernel φ has any particular functional form. In case of Hawkes process the data that we observe are the timestamps of the events t_1, \dots, t_N . The data that we miss is a branching structure of the process. In order to analyze the structure of the Hawkes process we define random variables Z_{ij} such that for $i \neq j$

$$Z_{ij} = \begin{cases} 1 & \text{if } j \text{ is an endogenous event caused by } i \\ 0 & \text{otherwise} \end{cases} \quad (3.2.22)$$

and

$$Z_{jj} = \begin{cases} 1 & \text{if } j \text{ is an exogenous event} \\ 0 & \text{otherwise.} \end{cases} \quad (3.2.23)$$

If we knew the values of random variables Z_{ij} then the reconstruction of the process would be limited to two separated problems of estimating μ and φ that are easy to pursue. Since we do not know the values of Z_{ij} we may only consider their expectations $\mathbb{E}[Z_{ij}] = p_{ij}$, which are equal to the probabilities that event i caused an event j (for $i \neq j$) or that event i is an exogenous event (for $i = j$). We will now apply E and M steps iteratively:

- **E-Step:**

In E-Step calculate new estimates for the probabilities p_{ij}^k :

$$p_{ij}^k = \frac{\hat{\varphi}^k(t_j - t_i)}{\hat{\lambda}^k(t_j)} = \frac{\hat{\varphi}^k(t_j - t_i)}{\hat{\mu}^k + \sum_{l=1}^{j-1} \hat{\varphi}^k(t_j - t_l)} \quad (3.2.24)$$

and

$$p_{jj}^k = \frac{\hat{\mu}^k}{\hat{\lambda}^k(t_j)} = \frac{\hat{\mu}^k}{\hat{\mu}^k + \sum_{l=1}^{j-1} \hat{\varphi}^k(t_j - t_l)}. \quad (3.2.25)$$

- **M-Step:**

In M-Step we actualize the estimates of μ and φ using the probabilities obtained in the last E-Step. The new background intensity is of the form

$$\hat{\mu}^{k+1} = \frac{1}{T} \sum_{j=1}^n p_{jj}^k. \quad (3.2.26)$$

Intuitively this is clear. If we knew the branching structure of the process we would approximate the background intensity counting the number of exogenous events in a unit of time. Since the branching structure is unknown we sum up the probabilities of each event being exogenous in a unit of time. Moreover, we approximate the kernel with a histogram. Given a time grid $\{0, \delta T, 2\delta T, \dots, T\}$ we approximate $\hat{\varphi}_m^{k+1} := \hat{\varphi}^{k+1}(m\delta T)$ as

$$\hat{\varphi}_m^{k+1} = \frac{1}{\delta T} \sum_{(i,j) \in A_m} p_{ij}^k \quad (3.2.27)$$

where A_m is the set of pairs of events such that $m\delta T \leq |t_i - t_j| \leq (m+1)\delta T$.

Using the histogram in M-Step is the most naive way to approximate the kernel. We can easily generalize it using a method called Kernel Density Estimation in order to get a smooth approximation of a kernel of Hawkes process. The method works as following. We take a function K (also called kernel function in theory of density estimation) satisfying the condition

$$\int_{-\infty}^{\infty} K(x) dx = 1. \quad (3.2.28)$$

The most widely used kernels are the boxcar kernel

$$K(x) = \frac{1}{2} \mathbb{1}_{[-1,1]}(x), \quad (3.2.29)$$

the Gaussian kernel of the form

$$K(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right) \quad (3.2.30)$$

and Epanechnikov kernel given by

$$K(x) = \frac{3}{4}(1 - |x|^3)^3 \mathbb{1}_{[-1,1]}(x). \quad (3.2.31)$$

In my computations I have used a Gaussian kernel. Now the approximation of the kernel of the Hawkes process is of the form

$$\hat{\varphi}(t) = \frac{1}{2\delta T} \sum_{i=1}^N \sum_{j=1}^N K\left(\frac{|t_i - t_j|}{h}\right), \quad (3.2.32)$$

where h is a smoothing parameter. (Silverman, 1986) offers a nice introduction to Kernel Density Estimation. A more sophisticated method to estimate kernel function of a Hawkes model is called Logspline Density Estimation. The idea is to use maximum likelihood estimation to estimate $\log(\varphi)$ by a cubic spline function having finite number of knots and being linear in the tail. The procedure has been thoroughly described in (Kooperberg and Stone, 1991). In my simulations I have been using EM Algorithm with all described estimation methods (histogram, Kernel Density Estimation and Logsplines), but due to low computational efficiency and significantly worse results of the histogram method as compared to two other EM methods, I decided only to present results for EM Algorithm methods using Kernel Density Estimation and Logsplines.

3.2.3 INAR Method

The INAR estimation method has been introduced in (Kirchner, 2014). Intuitively it works as following. We divide the time line into bins and count the number of points in each of them. The number of events in consecutive bins form an integer-valued autoregressive model. We then apply tools from theory of time series to obtain the parameters of the model and hence of the Hawkes process. Let us introduce a few definitions in order to formalize our intuition.

Definition 3.2.3.1. (Def. 2.1 in (Kirchner, 2014)) For an integer-valued random variable Y and a constant $\alpha \geq 0$ define the thinning operator \circ by

$$\alpha \circ Y = \sum_{k=1}^Y \xi_k^{(\alpha)}, \quad (3.2.33)$$

where $\xi_1^{(\alpha)}, \xi_2^{(\alpha)}, \dots$ are i.i.d. and independent of Y with $\xi_1^{(\alpha)} \sim \text{Poisson}(\alpha)$. We use convention that $\sum_{k=1}^0 \xi_k^{(\alpha)} = 0$.

Definition 3.2.3.2. (Def. 2.2 in (Kirchner, 2014)) Let $p \in \mathbb{N}$ and $\alpha_k \in [0, 1]$, $k = 1, 2, \dots, p$. An integer-valued autoregressive model of order p (INAR(p)) is a stationary sequence of integer-valued random variables $(X_n)_{n \in \mathbb{Z}}$ that is a solution to the stochastic difference equations

$$X_n - \sum_{k=1}^p \alpha_k \circ X_{n-k} = \epsilon_n, \quad n \in \mathbb{N}, \quad (3.2.34)$$

where each \circ operates independently (over k and n) and $(\epsilon_n)_{n \in \mathbb{N}}$ i.i.d., $\epsilon_0 \sim \text{Poisson}(\alpha_0)$ for some $\alpha_0 > 0$.

The main idea of the estimation method is based on the connection between Hawkes process and INAR time series given in the following theorem:

Theorem 3.2.3.3. (*Thm. 2.5 in (Kirchner, 2014)*) Let N be a univariate Hawkes process with background intensity μ and piecewise continuous excitement function φ such that $\sum_{k=1}^{\infty} \varphi(k\Delta)\Delta \leq 1$ for all $\Delta \in (0, 1)$. Furthermore, let $(X_n^{(\Delta)})$ be a univariate INAR(∞) sequence with innovation term $\alpha_0^{(\Delta)} = \Delta\mu$ and thinning coefficients $\alpha_k^{(\Delta)} = \Delta\varphi(k\Delta)$, $k = 1, 2, \dots, p$. We define a family of point processes by

$$N^{(\Delta)}(a, b) = \sum_{k:k\Delta \in (a,b]} X_k^{(\Delta)}, \quad a < b, \quad \Delta \in (0, 1). \quad (3.2.35)$$

Then we have that $N^{(\Delta)} \xrightarrow{w} N$, $\Delta \rightarrow 0^+$.

Now using the fact that the INAR(∞) sequence can be approximated by INAR(p) sequence, for $p \rightarrow \infty$, we come to conclusion that for a small bin size $\Delta > 0$ and large support $s = p\Delta$ we can estimate Hawkes process by fitting the INAR(p) model to the bin-count sequence of the timestamps. In order to do that we apply the conditional least squares estimation (CLS).

Definition 3.2.3.4. (*Def. 3.3 in (Kirchner, 2014)*) Let $(x_k)_{k=1,\dots,n}$ be an \mathbb{R} -valued sequence. Fix $p \in \mathbb{N}$ and define the conditional least-squares estimator (CLS estimator) as

$$\begin{aligned} \hat{\theta}_{CLS}^{(p)}: \mathbb{R}^n &\rightarrow \mathbb{R}^{p+1} \\ (x_1, \dots, x_n) &\rightarrow \hat{\theta}_{CLS}^{(p)}(x_1, \dots, x_n) := YZ^T(ZZ^T)^{-1} \end{aligned} \quad (3.2.36)$$

where

$$Z(x_1, \dots, x_n) := \begin{bmatrix} x_p & x_{p+1} & \cdots & x_{n-1} \\ x_{p-1} & x_p & \cdots & x_{n-2} \\ \cdots & \cdots & \cdots & \cdots \\ x_1 & x_2 & \cdots & x_{n-p} \\ 1 & 1 & \cdots & 1 \end{bmatrix} \quad (3.2.37)$$

is the design matrix and

$$Y(x_1, \dots, x_n) := (x_{p+1}, x_{p+2}, \dots, x_n) \in \mathbb{R}^{n-p}. \quad (3.2.38)$$

Combining the above theorem and definitions together we obtain the estimation method for Hawkes process.

Definition 3.2.3.5. (*Def. 3.6 in (Kirchner, 2014)*) Let t_1, \dots, t_N be a sample on $[0, T]$ from a Hawkes process with background intensity μ and kernel function φ . For some $\Delta > 0$ construct a sequence of bin-counts

$$x_k^{(\Delta)} := (\#\{t_i \in [(k-1)\Delta, k\Delta]\}), \quad k = 1, 2, \dots, m := \lfloor T/\Delta \rfloor. \quad (3.2.39)$$

Define the Hawkes estimator by applying the CLS estimator with respect to some support s , $\Delta < s < T$, and $p := \lceil s/\Delta \rceil$ on the sequence of bin-counts

$$\hat{\varphi}^{(\Delta, s)} := \frac{1}{\Delta} \hat{\theta}_{CLS}^{(p)} \left(\left(x_k^{(\Delta)} \right)_{k=1,\dots,m} \right). \quad (3.2.40)$$

Chapter 4

Comparison of the Estimation Methods on Synthetic Data

The main goal of this work is to compare the non-parametric estimation procedures described in the previous chapter. When comparing the performance of the methods I will look at them in various different circumstances. The analysis will be done on synthetic data generated from Hawkes model with specified kernel and parameters. I will be mostly interested in how the estimation outcome changes depending on the size of the sample and what impact the separation of point clusters has on the estimation performance. This is where cluster separation measure given by (2.3.5) will be used. Later I will also pursue the analysis of the cases of slowly decaying (i.e. power law) and some less common kernels like step or cut-off kernel. The comparison of the estimation methods on empirical data will follow in the next chapter.

4.1 Performance of the Estimation Methods with respect to Sample Size

In this section I want to analyze how the size of the sample influences the estimation performance of different methods. The comparison will be made on two types of data sets. The first is generated from Hawkes process with an exponential kernel with parameters $[\mu, n, \tau] = [0.05, 0.7, 4]$. In this case the clusters are well separated ($\kappa = 0.66$). On the other hand I will consider a second data set generated from Hawkes process with an exponential kernel with parameters $[\mu, n, \tau] = [0.05, 0.7, 18]$ ($\kappa = 3$), hence having strongly overlapping point clusters. Increasing the value of parameter τ , while keeping parameters μ and n fixed, causes bigger overlap of the clusters which makes the estimation more difficult (see figure 4.1). I have conducted my analysis on data samples of size 2000 and 4000. The reason for such a choice was the fact that my main interest was to test applicability of the methods in context of financial data and this is the range of sample sizes, which we would usually use to fit the model to real data.

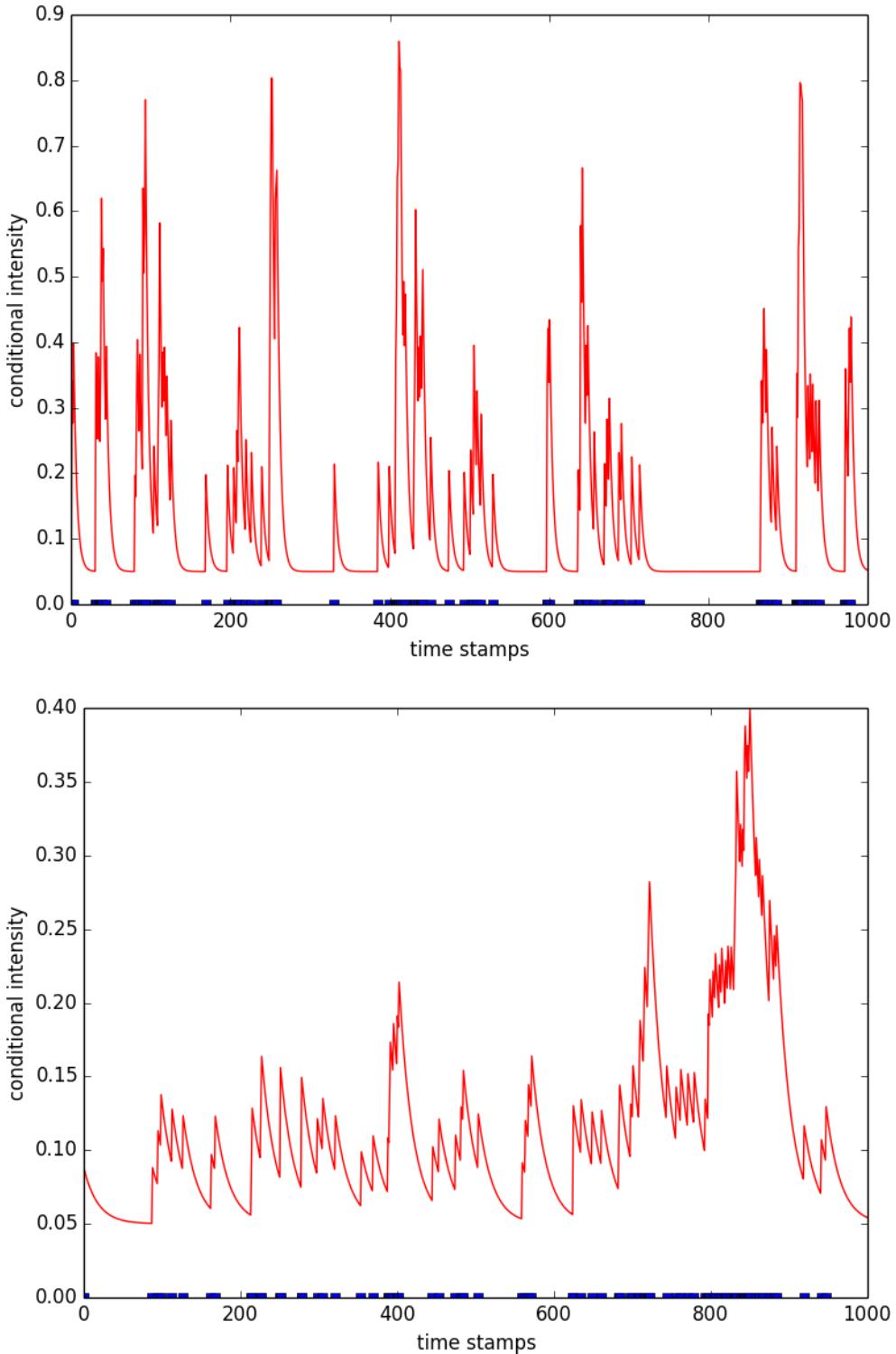


Figure 4.1: Conditional intensity of the Hawkes process with an exponential kernel. On the upper plot the parameters are $[\mu, n, \tau] = [0.05, 0.7, 4]$ (well-separated point clusters with $\kappa = 0.66$), whereas on the bottom plot the parameters are $[\mu, n, \tau] = [0.05, 0.7, 18]$ (overlapping point clusters with $\kappa = 3$).

4.1.1 Separated Clusters

In order to analyze the performance of the methods depending on the sample size in case of well-separated clusters I have generated the data from the Hawkes model with an exponential kernel and parameters $[\mu, n, \tau] = [0.05, 0.7, 4]$, which gives $\kappa = 0.66$. In figure 4.2 we can see the result of kernel fitting on a small sample of size 2000 (upper plot) and medium sample size consisting of 4000 points (bottom plot). In both cases EM Algorithm with Logsplines provide good estimation following closely the theoretical kernel. EM Algorithm with Kernel Density Estimation, WH and INAR methods perform slightly worse, but still oscillate around true curve.

In order to further compare the results, one can consider bias and efficiency of estimates of branching ratio n and background intensity μ . In order to do that I have generated 50 times data samples of sizes from 1000 to 4000 from Hawkes process with an exponential kernel and parameters $[\mu, n, \tau] = [0.05, 0.7, 4]$. Then I have applied all the methods to obtain the estimates of parameters. In figure 4.3 I present the median of a bias of estimates, but also 25- and 75- percent quantiles to capture their efficiency. The first thing we can notice in the plots is that the estimation results for each method seems stable for different sample sizes, with bias decreasing and efficiency improving as the sample size grows. Furthermore, we can see that in the case of well-separated data clusters the estimates obtained from the WH method and EM Algorithm with Logsplines outperform other two methods in terms of bias. Furthermore, the EM Algorithm seems to have slightly better efficiency compared to the WH method. The performance of the INAR method seems to be the worst having both the highest bias and the worst efficiency.

Furthermore, I compared the error with respect to cumulated theoretical kernel $\int_0^t \varphi(s)ds$ of the cumulated estimated kernels $\int_0^t \hat{\varphi}(s)ds$ obtained applying 50 times all the estimation methods to the data sample of size 2000 (upper plot) and 4000 (bottom plot) generated from Hawkes process with exponential kernel and parameters $[\mu, n, \tau] = [0.05, 0.7, 4]$. The results are presented in figure 4.4. Visually both plots looks similar, but the outcomes are less noisy in the bigger sample. All the methods underestimate the head of kernel, but capture the tail quite well. The performance of INAR at the small time scale looks quite dramatic, whereas WH method seems to perform best overall.

4.1.2 Overlapping Clusters

In order to analyze the performance of the methods depending on the sample size in the case of overlapping clusters I have generated the data from the Hawkes model with an exponential kernel and parameters $[\mu, n, \tau] = [0.05, 0.7, 18]$, which means $\kappa = 3$. Overall we can see in figure 4.5 that higher overlapping of point clusters seems to significantly influence the performance of all methods, causing worse estimation as compared to figure 4.2. Moving from small sample size to a medium sample size improves the results slightly as we can see at the bottom plot of figure 4.5.

The analysis of bias and efficiency of estimates of branching ratio and background intensity obtained by fitting the model on data samples of different size generated 50 times from the Hawkes model with an exponential kernel and parameters $[\mu, n, \tau] = [0.05, 0.7, 18]$ shows us that the estimation of the parameters is quite stable with respect to the sample size for all methods. In figure 4.6 we see that the bias is overall higher as compared to figure 4.3. Interesting is the fact that EM Algorithm with Kernel Density Estimation seems to

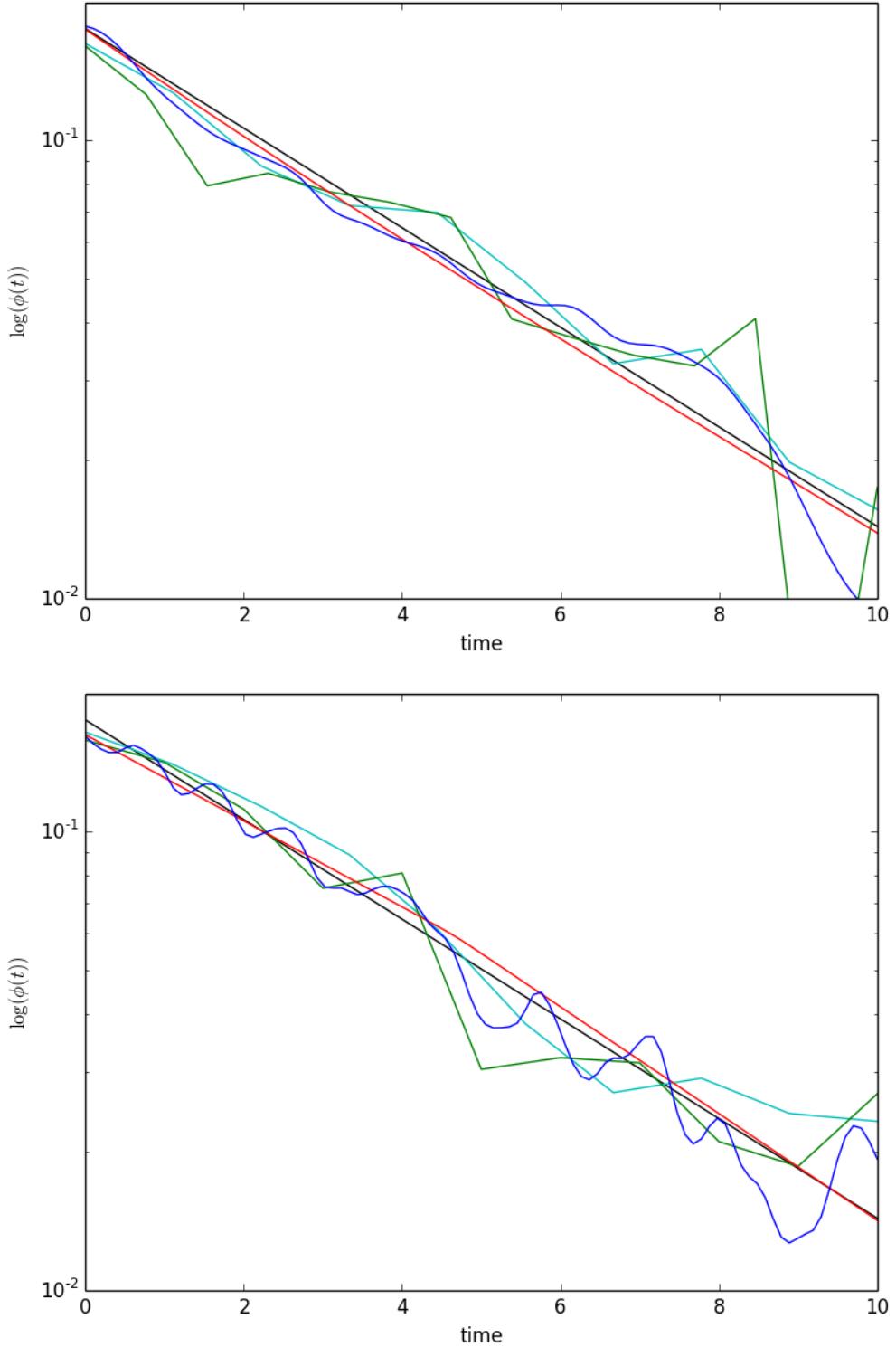


Figure 4.2: Estimated kernels of the models fitted to the data sample of size 2000 (upper plot) and 4000 (bottom plot) generated from Hawkes process with an exponential kernel with parameters $[\mu, n, \tau] = [0.05, 0.7, 4]$ ($\kappa = 0.66$) using the WH method (cyan line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation (blue line). The theoretical kernel (black line) is also given.

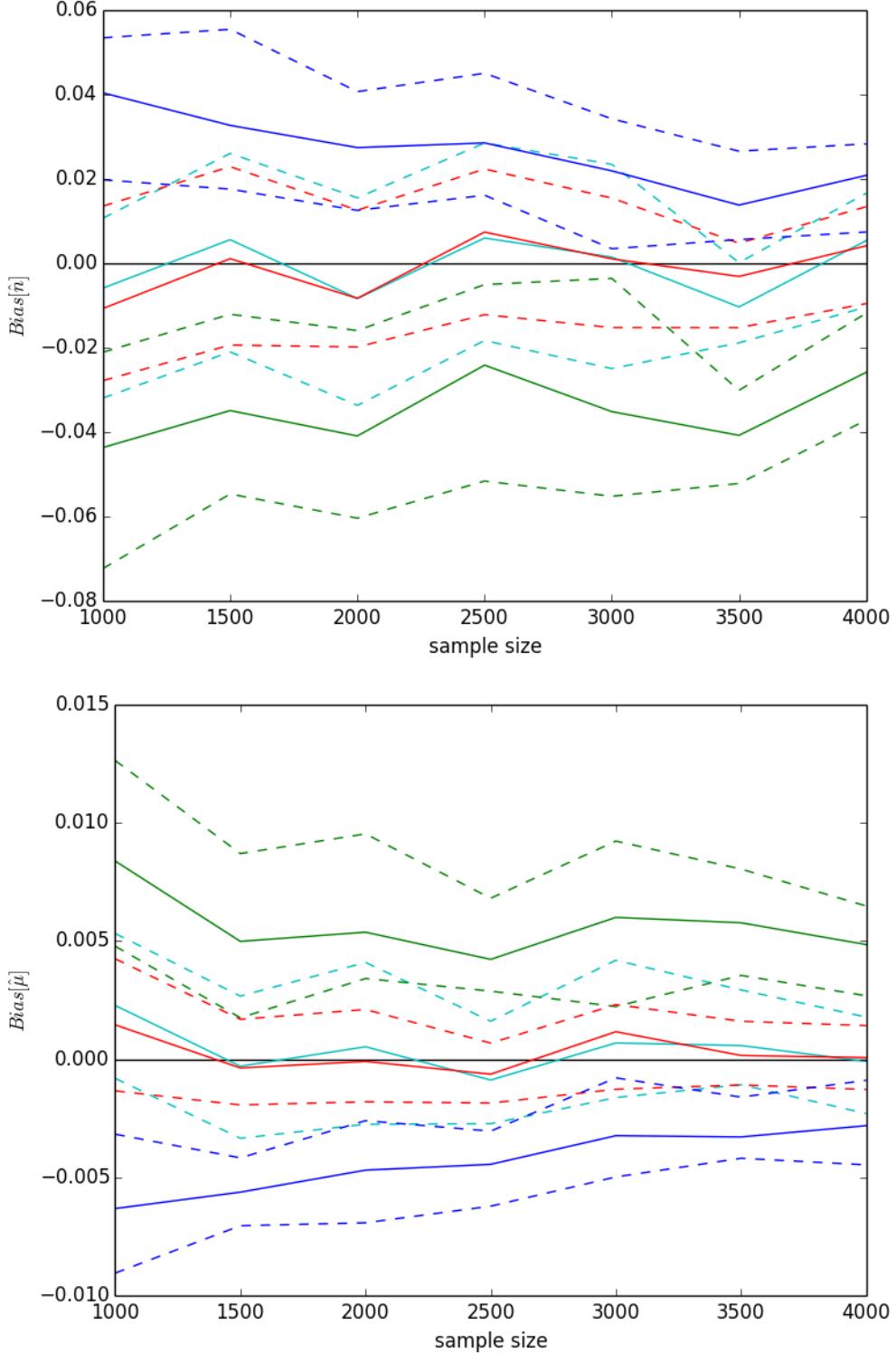


Figure 4.3: Bias and efficiency of estimates of branching ratio n and background intensity μ of the models fitted using the WH method (cyan line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation (blue line) on the data generated from Hawkes process with an exponential kernel with parameters $[\mu, n, \tau] = [0.05, 0.7, 4]$ ($\kappa = 0.66$) with respect to sample size. Median (solid line), 25-, 75-, percent quantiles (dashed line) are given.

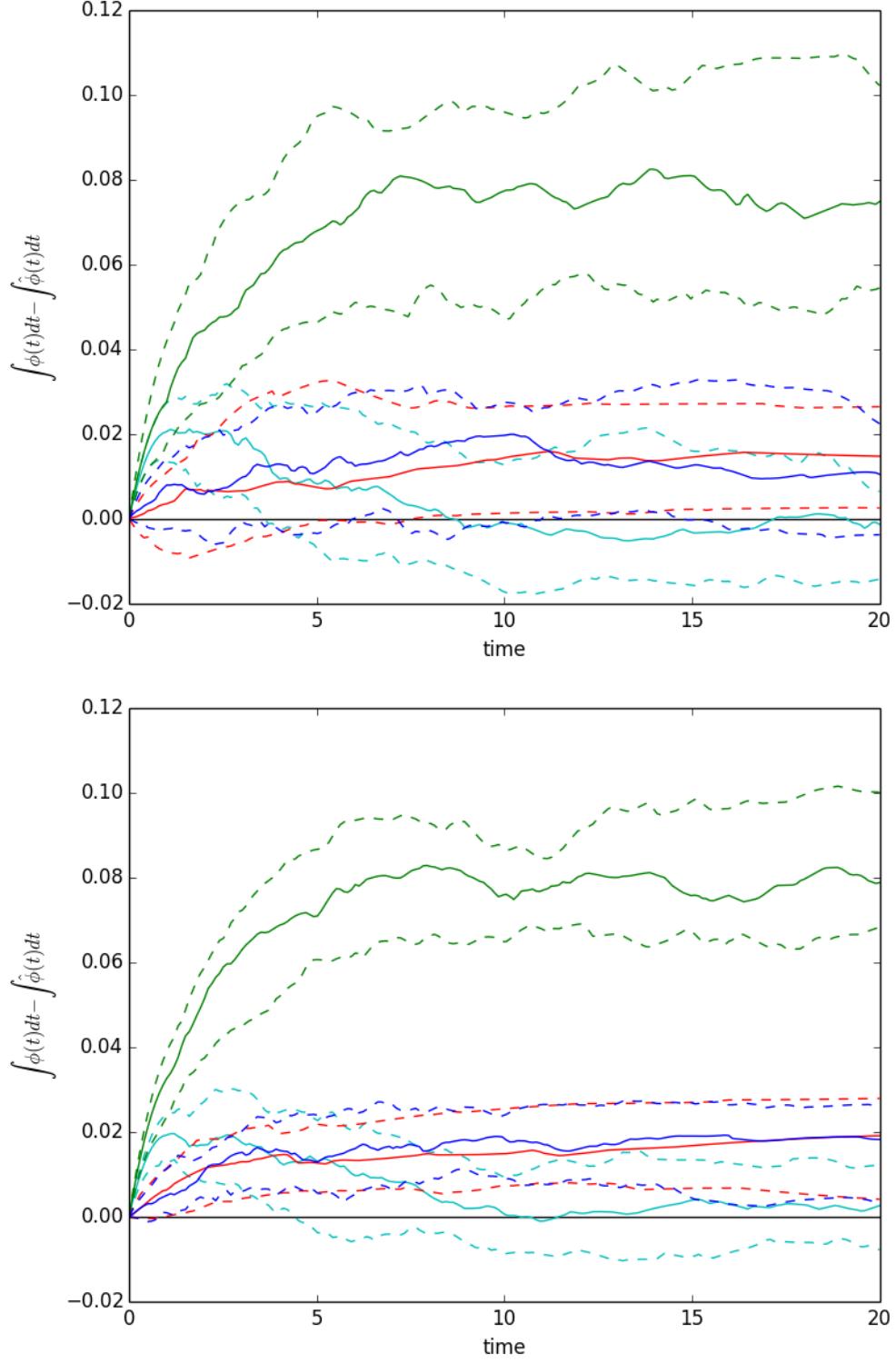


Figure 4.4: Error of the cumulated estimated kernel $\int_0^t \hat{\varphi}(s)ds$ obtained using the WH method (cyan line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation (blue line) on the data sample of size 2000 (upper plot) and 4000 (bottom plot) generated from Hawkes process with an exponential kernel with parameters $[\mu, n, \tau] = [0.05, 0.7, 4]$ ($\kappa = 0.66$). Median (solid line), 25-, 50-, 75-, percent quantiles (dashed line) are given.

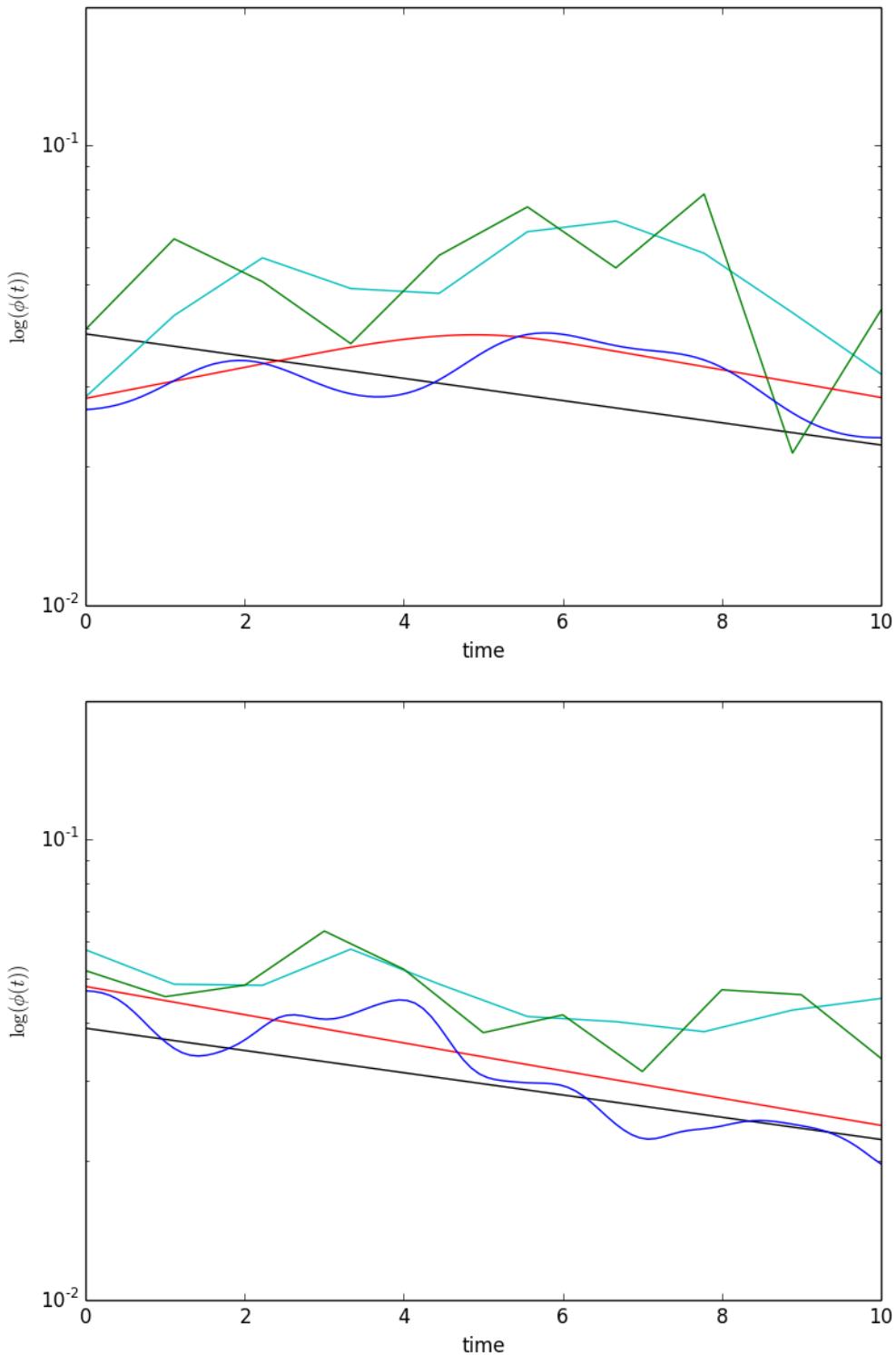


Figure 4.5: Estimated kernels of the models fitted on the data sample of size 2000 (upper plot) and 4000 (bottom plot) generated from Hawkes process with an exponential kernel with parameters $[\mu, n, \tau] = [0.05, 0.7, 18]$ ($\kappa = 3$) using the WH method (cyan line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation(blue line). The theoretical kernel (black line) is also given.

perform best in the case of overlapping clusters, whereas on separated clusters it did not work very well relative to other methods. Conversely the WH method, which produced superb results in the previous case, now performs poorly as compared to other methods.

Further conclusions come from figure 4.7 showing errors of cumulated kernels $\int_0^t \hat{\varphi}(s)ds$. Similarly as before the cumulated kernels has been estimated 50 time on a data sample of size 2000 (upper plot) and 4000 (bottom plot) generated from Hawkes process with an exponential kernel with parameters $[\mu, n, \tau] = [0.05, 0.7, 18]$. Here again the cumulated kernel from EM Algorithm with Kernel Density Estimation seems to follow the cumulated theoretical kernel most closely. Not surprisingly the WH and INAR cumulated kernels are doing poorly especially in terms of approximating the tail of the theoretical kernel.

4.2 Performance of the Estimation Methods with respect to the Overlap of Point Clusters

In order to analyze further how the performance of the estimation methods is influenced by a degree of separation of the clusters, let us compare the top and bottom plots of figure 4.4 and figure 4.7. The first thing we notice is that in the case of separated clusters all the methods underestimate head of kernel, whereas in the case of overlapping clusters they capture it fairly well. When we consider the tail of the kernel though the situation is opposite. In the case of $\kappa = 0.66$ all the methods approximate tail well, but for $\kappa = 3$ the WH and INAR methods significantly overestimate it. More insight into an issue of cluster separation impact on estimation performance can provide us analysis of figure 4.8 presenting biases of estimates. We can clearly see for both parameters that when cluster overlap increases then the difference between estimated and true values also grows. Moreover figure 4.8 illustrate well the fact discussed already above that for well separated clusters (κ ranging from 0.5 to 1) the EM Algorithm with Logsplines and WH methods perform best, whereas for overlapping clusters EM Algorithm with Kernel Density Estimation seems to be closest to true kernel followed by EM Algorithm with Logsplines. In the latter case the performance of WH gets significantly worse.

4.3 Performance of the Estimation Methods with respect to the Branching Ratio

I want to analyze how the value of branching ratio influences the performance of the estimation methods. In order to consider that I will keep the parameters $[\mu, \tau] = [0.05, 10]$ fixed and I will vary the value of branching ratio from 0.2 to 0.8. First let us compare plots in figure 4.9 showing results of fitting of a model for the branching $n = 0.3$ (upper plot) and $n = 0.7$ (bottom plot). Increase of branching ratio does not seem to have an influence on performance of EM Algorithm methods. WH and INAR methods improve estimation results for higher value of branching ratio, but still in both scenarios they seem to be worse then the other two methods.

Plots of the bias of estimates in figure 4.10 show us that EM Algorithm methods seem to overestimate branching ratio for small values of parameter n , but underestimate it when the branching ratio is high. WH and INAR methods constantly underestimate the value of parameter n having bigger bias then EM methods. In case of background intensity the

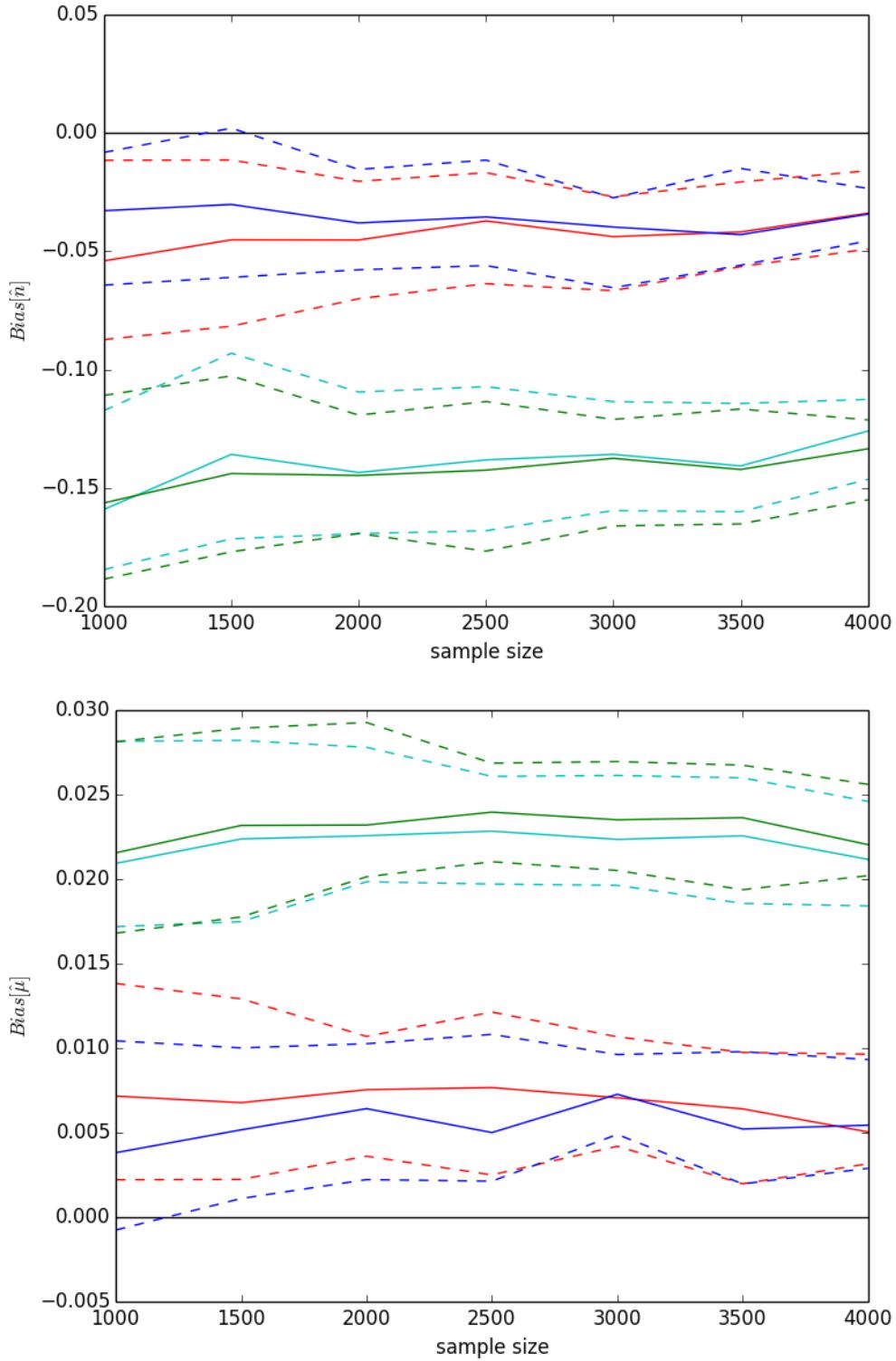


Figure 4.6: Bias and efficiency of estimates of branching ratio n and background intensity μ of the models fitted using the WH method (cyan line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation (blue line) on the data generated from Hawkes process with an exponential kernel with parameters $[\mu, n, \tau] = [0.05, 0.7, 18]$ ($\kappa = 3$) with respect to sample size. Median (solid line), 25-, 75-, percent quantiles (dashed line) are given.

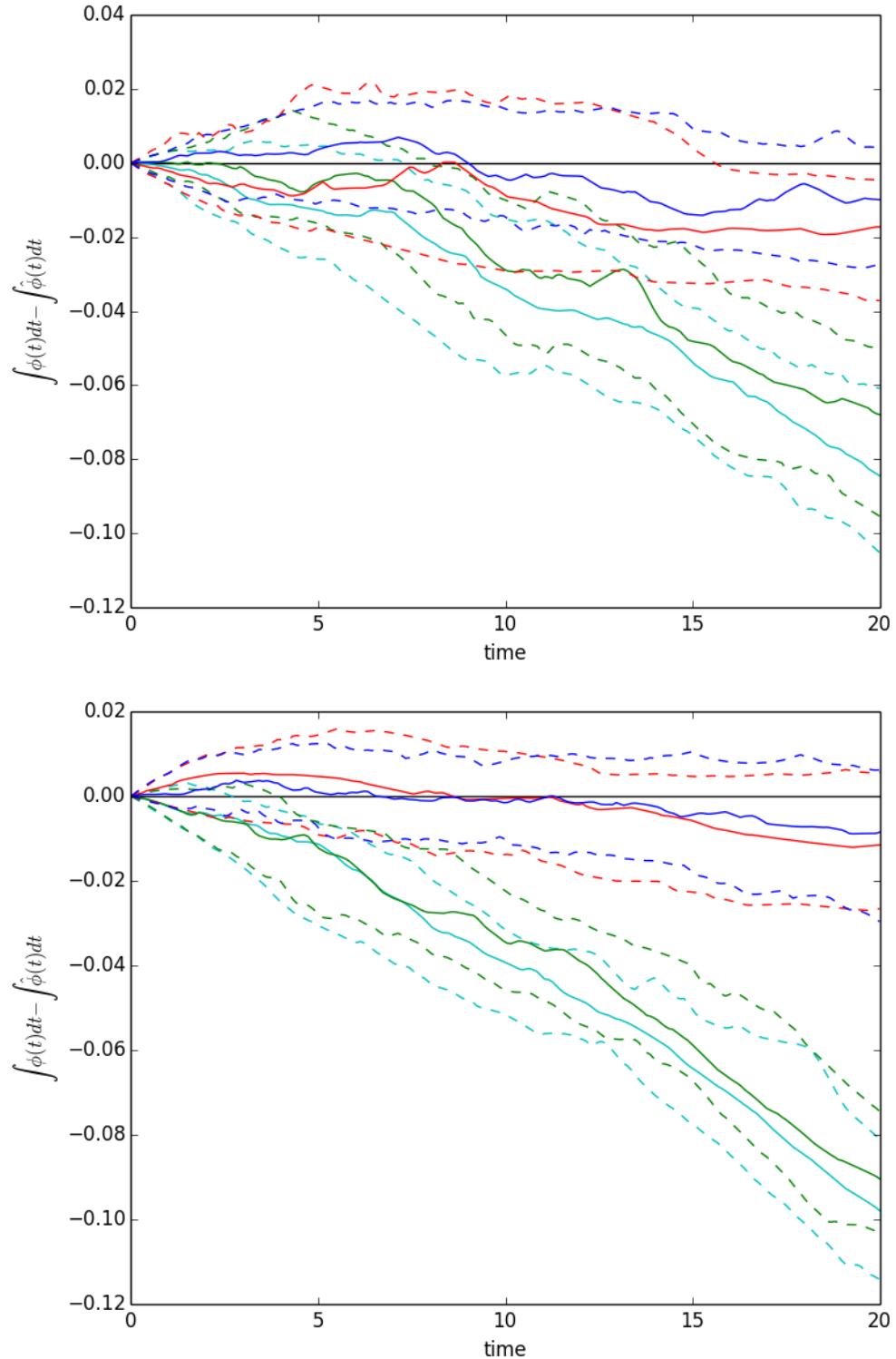


Figure 4.7: Error of the cumulated estimated kernel $\int_0^t \hat{\varphi}(s)ds$ obtained using the WH method (cyan line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation (blue line) on the data sample of size 2000 (upper plot) and 4000 (bottom plot) generated from Hawkes process with an exponential kernel with parameters $[\mu, n, \tau] = [0.05, 0.7, 18]$ ($\kappa = 3$). Median (solid line), 25-, 50-, 75- percent quantiles (dashed line) are given.

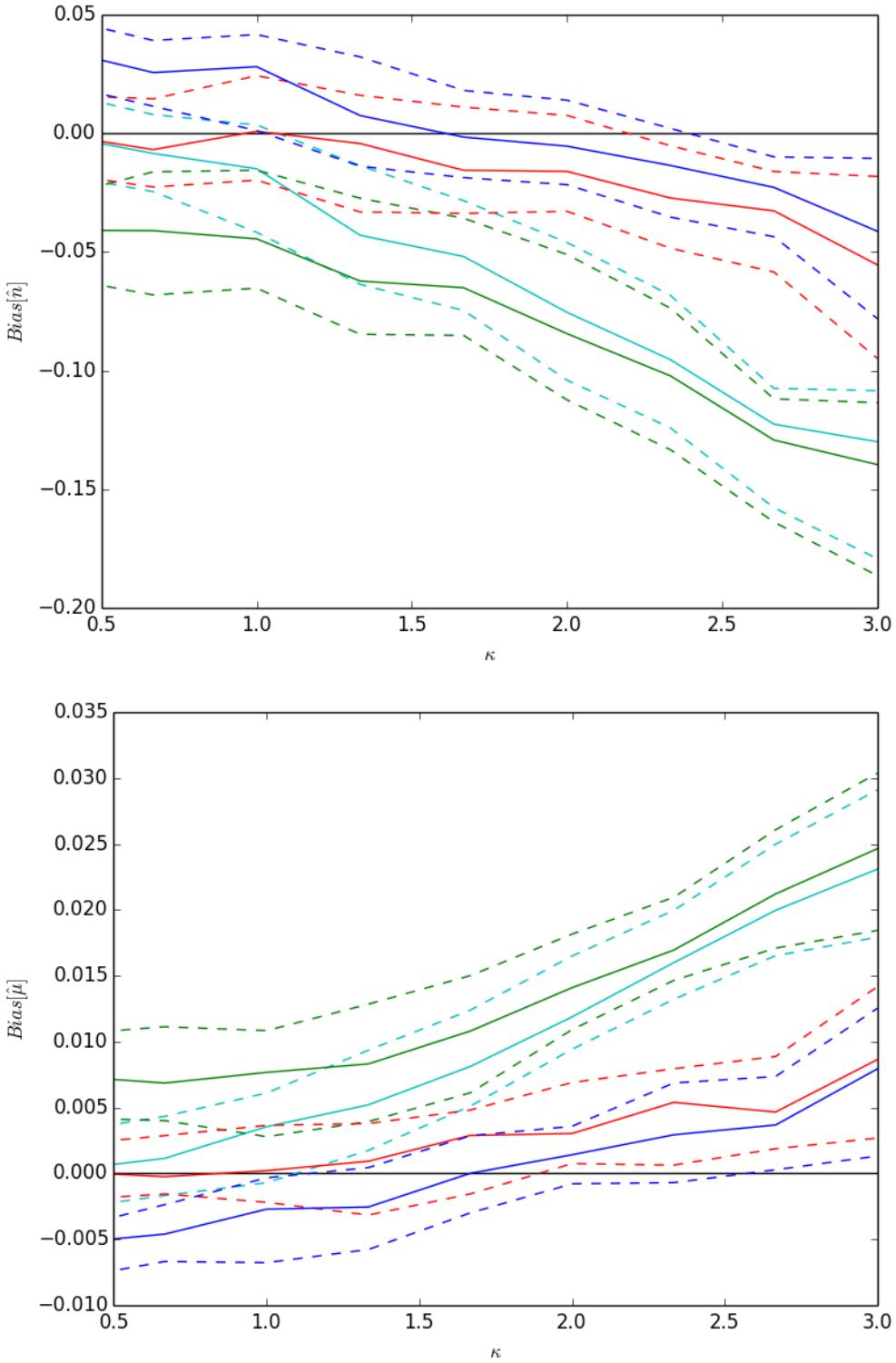


Figure 4.8: Bias and efficiency of estimates of branching ratio n and background intensity μ of the models fitted using the WH method (cyan line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation (blue line) on the data sample of size 2000 generated from Hawkes process with an exponential kernel with parameters $[\mu, n] = [0.05, 0.7]$ and parameter τ from 2 to 18, which gives the range of cluster separation measure κ from 0.5 to 3. Median (solid line), 25-, 75-, percentiles (dashed line) are given.

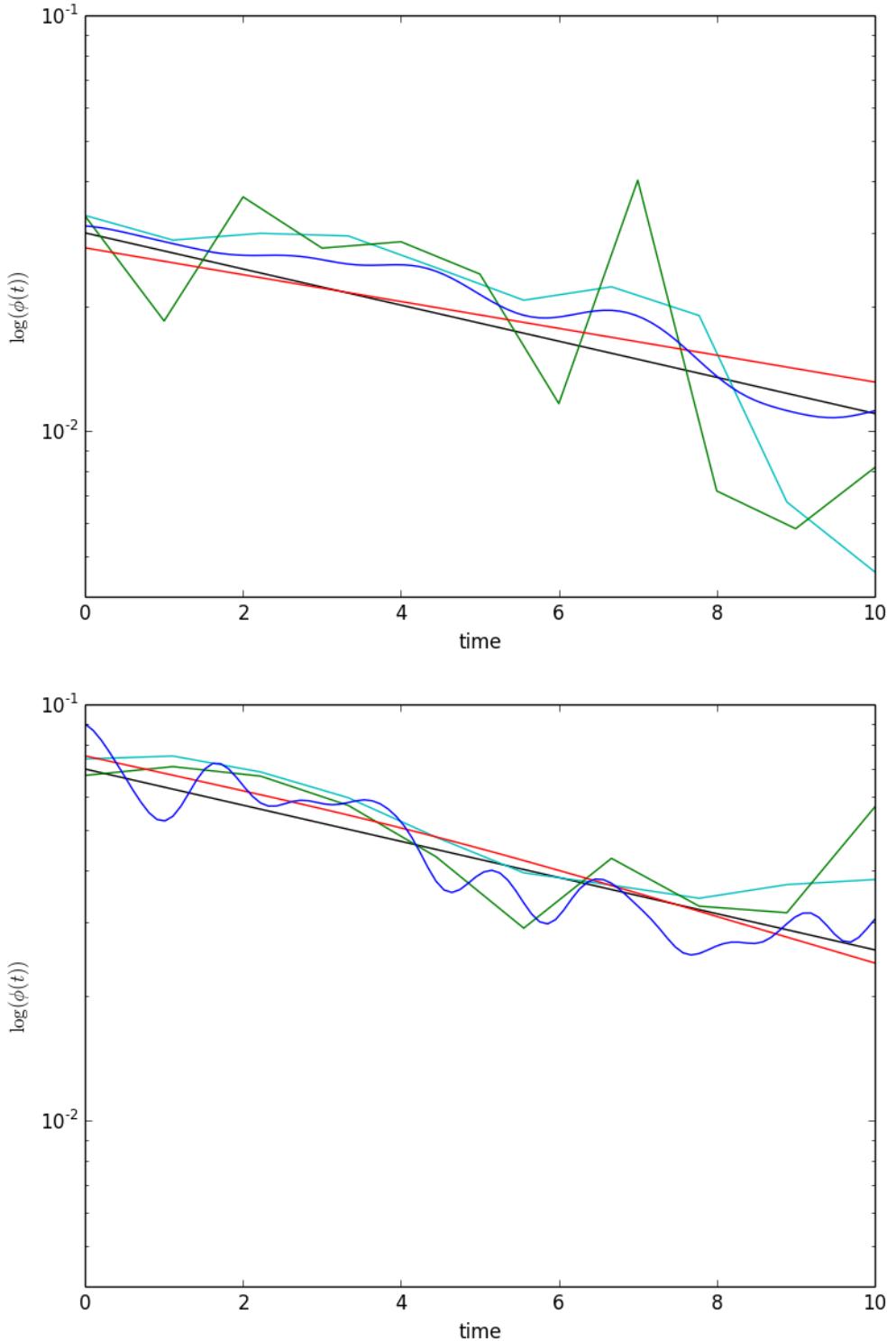


Figure 4.9: Estimated kernels of the models fitted on the data sample of size 2000 generated from Hawkes process with an exponential kernel with parameters $[\mu, \tau] = [0.05, 10]$ and branching ratio $n = 0.3$ (upper plot) and $n = 0.7$ (bottom plot) using the WH method (cyan line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation(blue line). The theoretical kernel (black line) is also given.

estimate provided by EM Algorithm with Logsplines is best, whereas both WH and INAR methods slightly overestimate its value.

For a small branching ratio we can see in the upper plot of figure 4.11 that cumulated kernels estimated from EM Algorithm methods follow almost perfectly the theoretical one. The WH method seems to approximate kernel in the head quite well, but tail badly. On the other hand the INAR method performs badly in head of the kernel, but behaves quite well in the tail. In case of branching ratio $n = 0.7$ both EM Algorithms slightly underestimate head of kernel. WH also performs well in the head but misses the tail, whereas INAR performs poorly overall.

4.4 Slowly Decaying Memory Kernel

I would like to treat the case of the slowly decaying memory kernels (i.e. power law memory kernels) separately. Long-memory kernels occur often in applications. In what follows I will work on the data samples generated from the Hawkes model with a power law kernel (see (2.2.1)) with parameters $[\mu, n, c, \theta] = [0.05, 0.9, 1, 0.3]$. For slowly decaying kernels, I introduce one additional method to those previously used, namely modification of WH method using a semi-uniform grid (3.2.10) (as opposed to one using a uniform grid (3.2.9)). In figure 4.12 we see that curves produced by the EM Algorithm with Logsplines and modified WH method follow the theoretical curve well (though the WH curve is much more noisy). The EM Algorithm with Kernel Density Estimation behaves well in the head, but gets very noisy and off the curve for larger times. Moreover, from results provided by the original WH and INAR method we can see a drawback of using a uniform grid when estimating long memory kernels. If we wanted to get more precision in the head of the kernel we would need to set bin-size very small which would mean a huge number of bins, because we want to make estimation over long time. This would also result in poor approximation of the tail, because a single event falling into a bin would cause a spike in estimated kernel tail.

On figure 4.13 we see that the estimates provided by all the methods have constantly high bias independently of the sample size with the EM Algorithm with Logsplines performing slightly better than the rest.

On the plot of cumulated estimated kernels (figure 4.14) we see that both the EM Algorithm with Logsplines and modified WH method perform well. The WH method underestimates the tail of the kernel slightly in both samples sizes. What is surprising is the fact that in a smaller sample it captures the tail relatively well, but misses it in a bigger sample. The performance of the EM Algorithm with Kernel Density Estimation and INAR method is poor. The former method strongly underestimates tail of kernel, whereas the latter approximates the head badly.

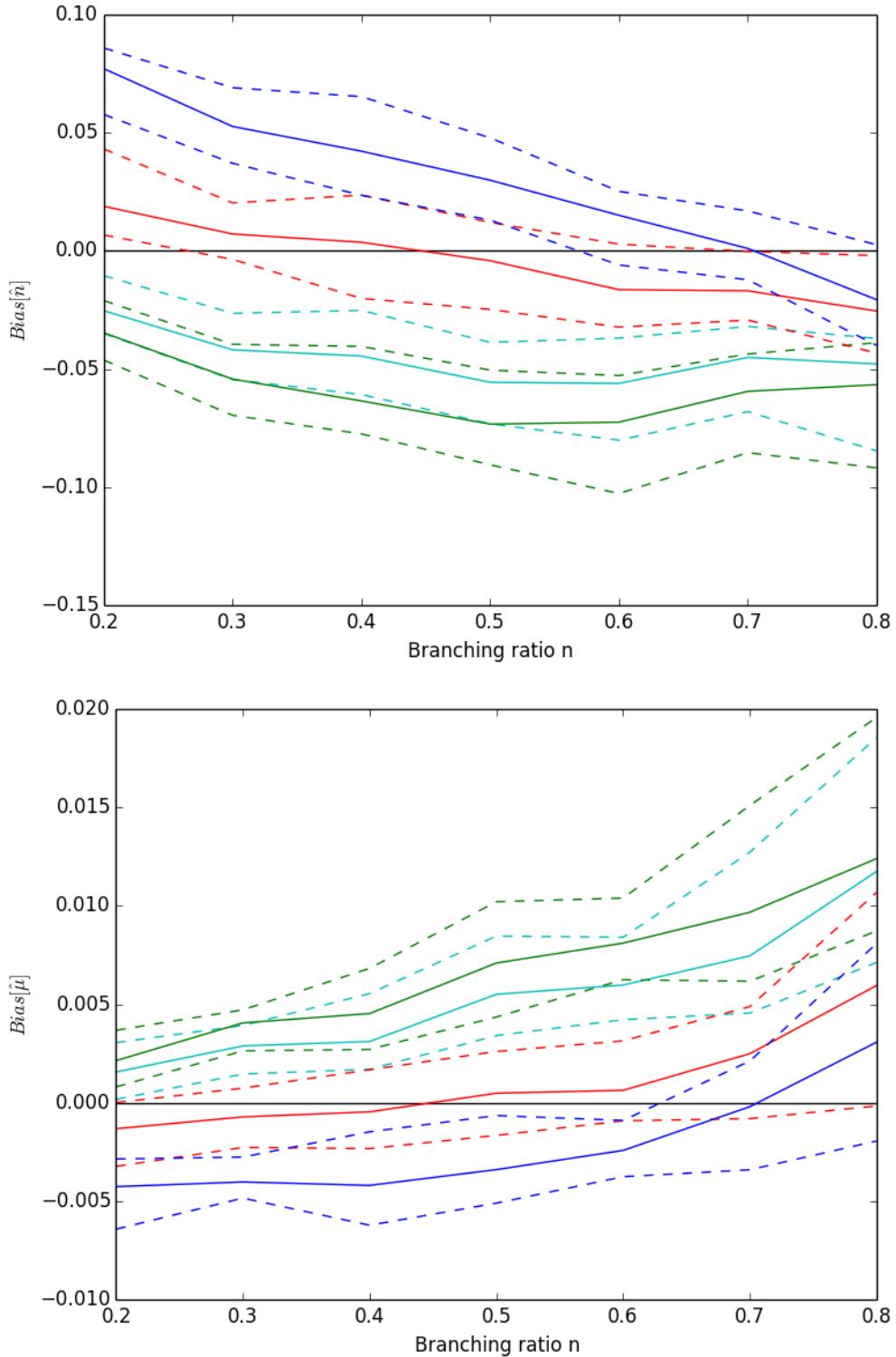


Figure 4.10: Bias and efficiency of estimates of branching ratio n and background intensity μ of the models fitted using the WH method (cyan line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation (blue line) on the data sample of size 2000 generated from Hawkes process with an exponential kernel with parameters $[\mu, \tau] = [0.05, 10]$ and branching ratio n ranging from 0.2 to 0.8. Median (solid line), 25-, 75-, percent quantiles (dashed line) are given.

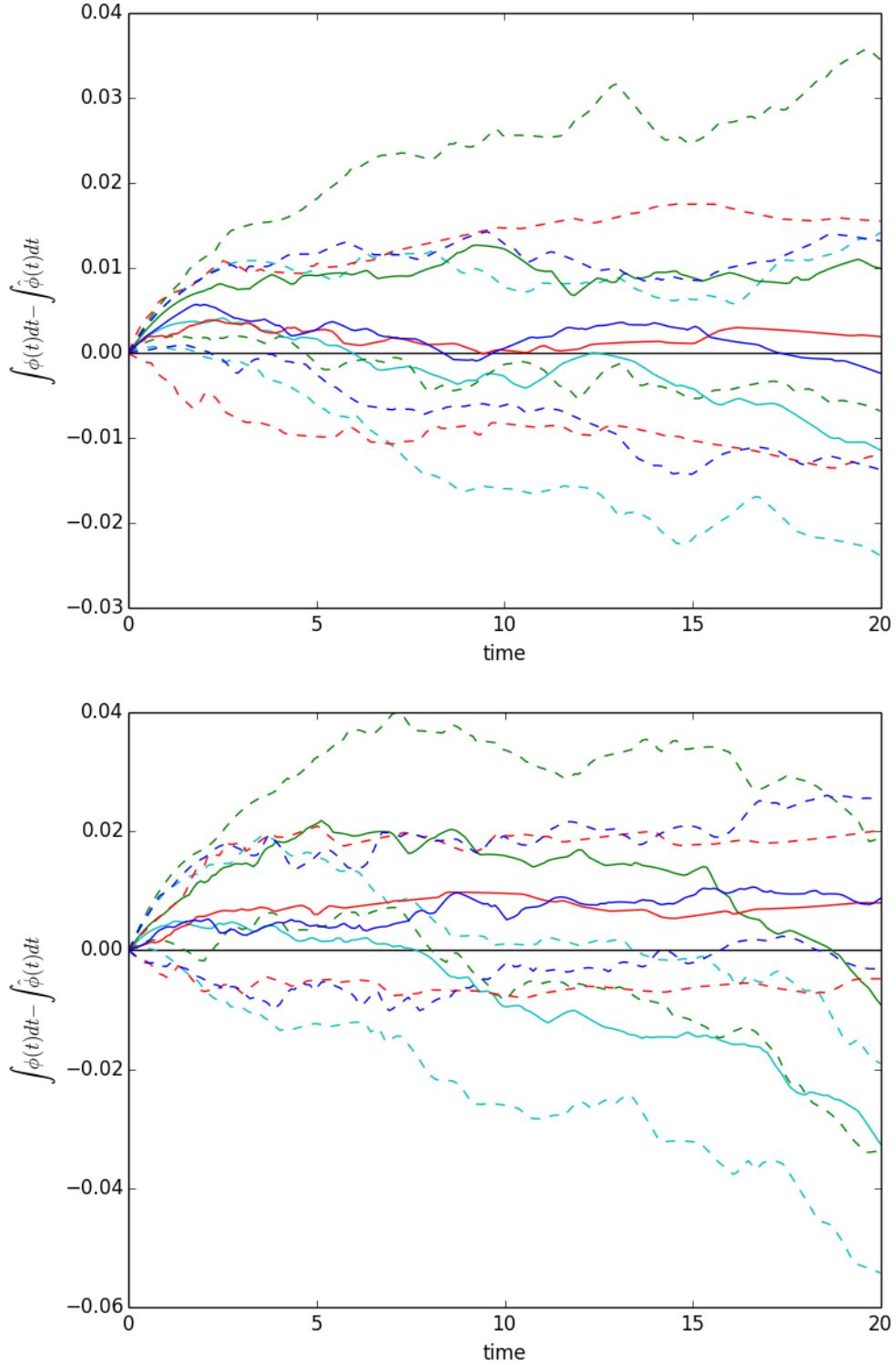


Figure 4.11: Error of the cumulated estimated kernel $\int_0^t \hat{\varphi}(s) ds$ obtained using the WH method (cyan line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation (blue line) on the data sample of size 2000 generated from Hawkes process with an exponential kernel with parameters $[\mu, \tau] = [0.05, 10]$ and branching ratio $n = 0.3$ (upper plot) and $n = 0.7$ (bottom plot). Median (solid line), 25-, 75-, percent quantiles (dashed line) are given.

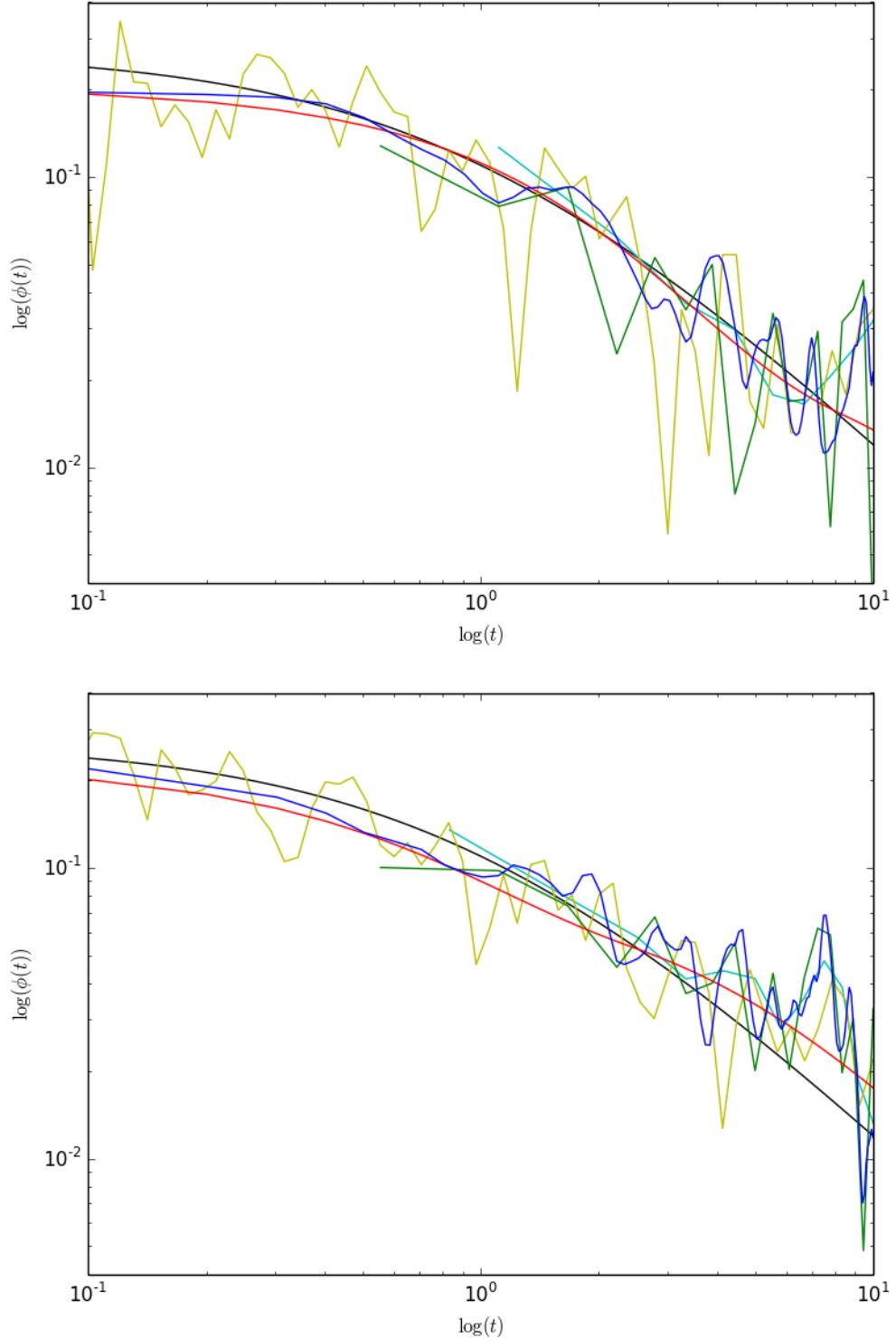


Figure 4.12: Estimated kernels of the models fitted on the data sample of size 2000 (upper plot) and 4000 (bottom plot) generated from Hawkes process with a power law kernel with parameters $[\mu, n, c, \theta] = [0.05, 0.9, 1, 0.3]$ using the WH method on the uniform grid (cyan line), WH method on the semi-uniform grid (yellow line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation (blue line). The theoretical kernel (black line) is also given.

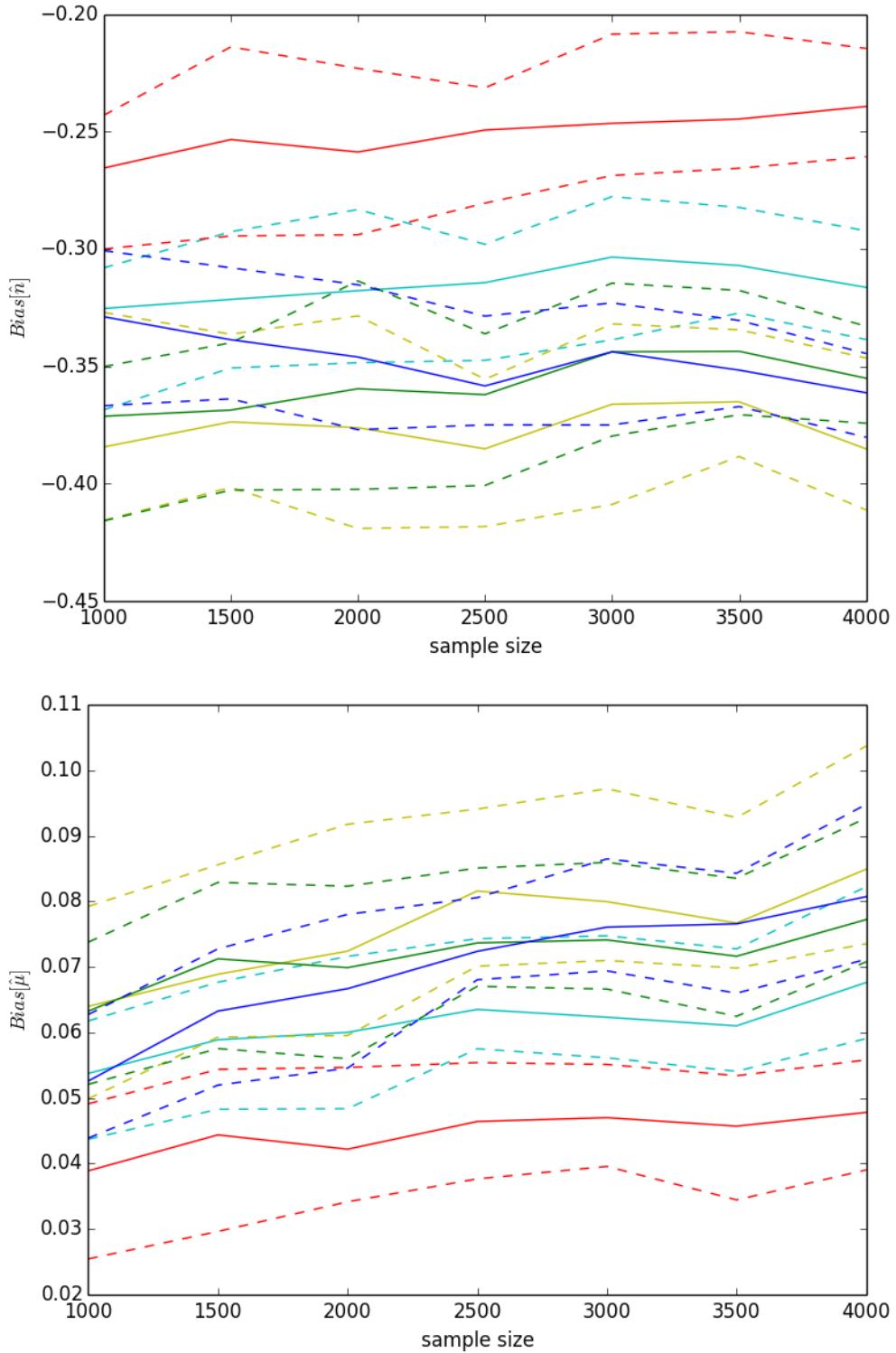


Figure 4.13: Bias and efficiency of estimates of branching ratio n and background intensity μ of the models fitted using the WH method on the uniform grid (cyan line), WH method on the semi-uniform grid (yellow line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation (blue line) on the data generated from Hawkes process with a power law kernel with parameters $[\mu, n, c, \theta] = [0.05, 0.9, 1, 0.3]$ with respect to sample size. Median (solid line), 25-, 75-, percent quantiles (dashed line) are given.

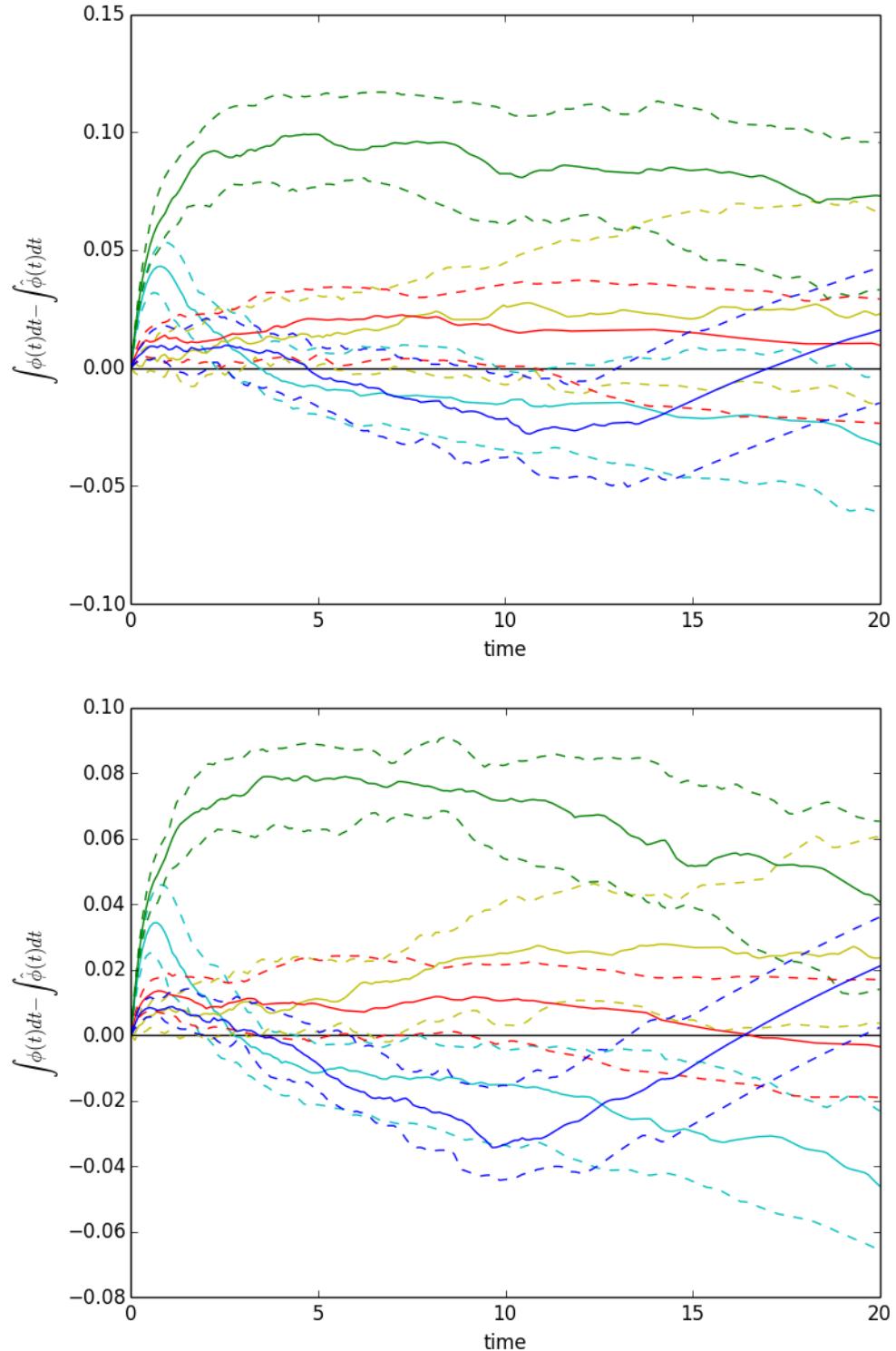


Figure 4.14: Error of the cumulated estimated kernel $\int_0^t \hat{\phi}(s)ds$ obtained using the WH method on the uniform grid (cyan line), WH method on the semi-uniform grid (yellow line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation (blue line) on the data sample of size 2000 (upper plot) and 4000 (bottom plot) generated from Hawkes process with a power law kernel with parameters $[\mu, n, c, \theta] = [0.05, 0.9, 1, 0.3]$. Median (solid line), 25-, 75-, percent quantiles (dashed line) are given.

4.5 Performance of Estimation Methods on Other Types of Kernels

In this section I would like to check the performance of the estimation on some less regular kernels such as the step function kernel given by

$$\varphi_{ab}(t) = \frac{n}{b-a} \mathbb{1}_{\{t \in [a,b]\}}(t), \quad (4.5.1)$$

the cut-off kernel (2.2.2) and its approximation being sum of exponential functions (2.2.3). In the case of approximation of a step function I will be interested how the shift of the step influences the outcome of the estimation. In figure 4.15 we can see the approximation of the kernel with a small shift ($a = 1$). We see that for both sample sizes all the methods capture the shape of the kernel relatively well, but the EM Algorithm with Logsplines is almost perfect, missing only the top part of the step slightly.

Similarly when we consider the step function with a bigger shift ($a = 2$) in figure 4.16 we see that all the methods reproduce the rectangular shape fairly well. Again, as in the case of small shift, the EM Algorithm with Logsplines can adjust itself best to the shape of the theoretical curve.

In figure 4.17 we can see the estimation methods applied to the data generated from the Hawkes model with a cut-off kernel with parameters $[\mu, n, \tau_0, \epsilon] = [0.05, 0.9, 1, 0.3]$. The crucial parameter is τ_0 also called a cut-off parameter, since it regulates the size of cut-off close to 0. Similarly, as in the case of the step function, one interesting question will be how the value of the cut-off parameter influences results of estimation. In order to answer this question I will look on the data samples of size 2000 and 4000. In figure 4.17 we see that for $\tau_0 = 1$ the EM Algorithm with Logsplines almost perfectly reproduces the spike. The WH and INAR methods also seem to capture the shape of the kernel quite well, whereas Kernel Density Estimation is not able to reproduce the spike.

Presented in figure 4.18 are plots in the case of a bigger cut-off, where I have used $\tau_0 = 2$. The sample sizes are again 2000 (upper plot) and 4000 (bottom plot). In this case all the methods seem to perform worse than for $\tau_0 = 1$. The EM Algorithm with Logsplines approximates the curve quite well, but overestimates the spike. The INAR method follows the theoretical kernel closely, but is very noisy. The other two methods represent the spike poorly.

Let us see how the estimation methods work on the approximation of cut-off function (2.2.3) with parameters $[\mu, n, \tau_0, \epsilon] = [0.05, 0.9, 1, 0.3]$. The results are presented in figure 4.19. Since the kernel is smooth it is not surprising that EM Algorithm methods both using smoothing tools approximate the theoretical curve well. The other two methods are much worse and actually missing the bump completely on both sample sizes.

When using parameters $[\mu, n, \tau_0, \epsilon] = [0.05, 0.9, 2, 0.3]$ in the approximation of cut-off we encounter similar problems as in figure 4.18. The EM Algorithm with Kernel Density Estimation seems to be the only one to follow the theoretical curve, though its performance is not particularly good (see 4.20). The WH and INAR methods seem again to completely miss the bump.

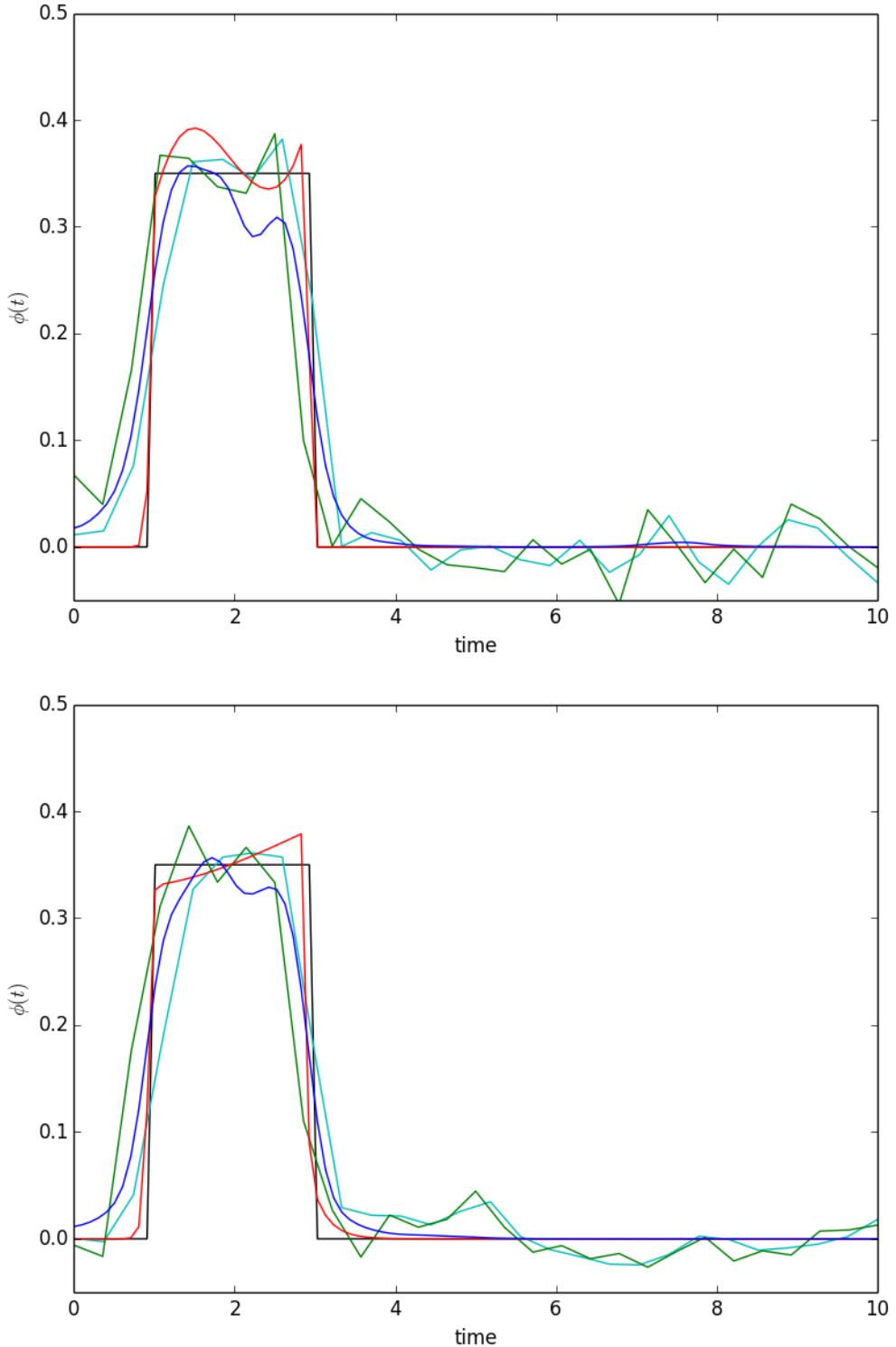


Figure 4.15: Estimated kernels of the models fitted on the data sample of size 2000 (upper plot) and 4000 (bottom plot) generated from Hawkes process with a step function kernel with parameters $[\mu, n] = [0.05, 0.7]$ using the WH method (cyan line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation (blue line). The theoretical kernel (black line) is also given.

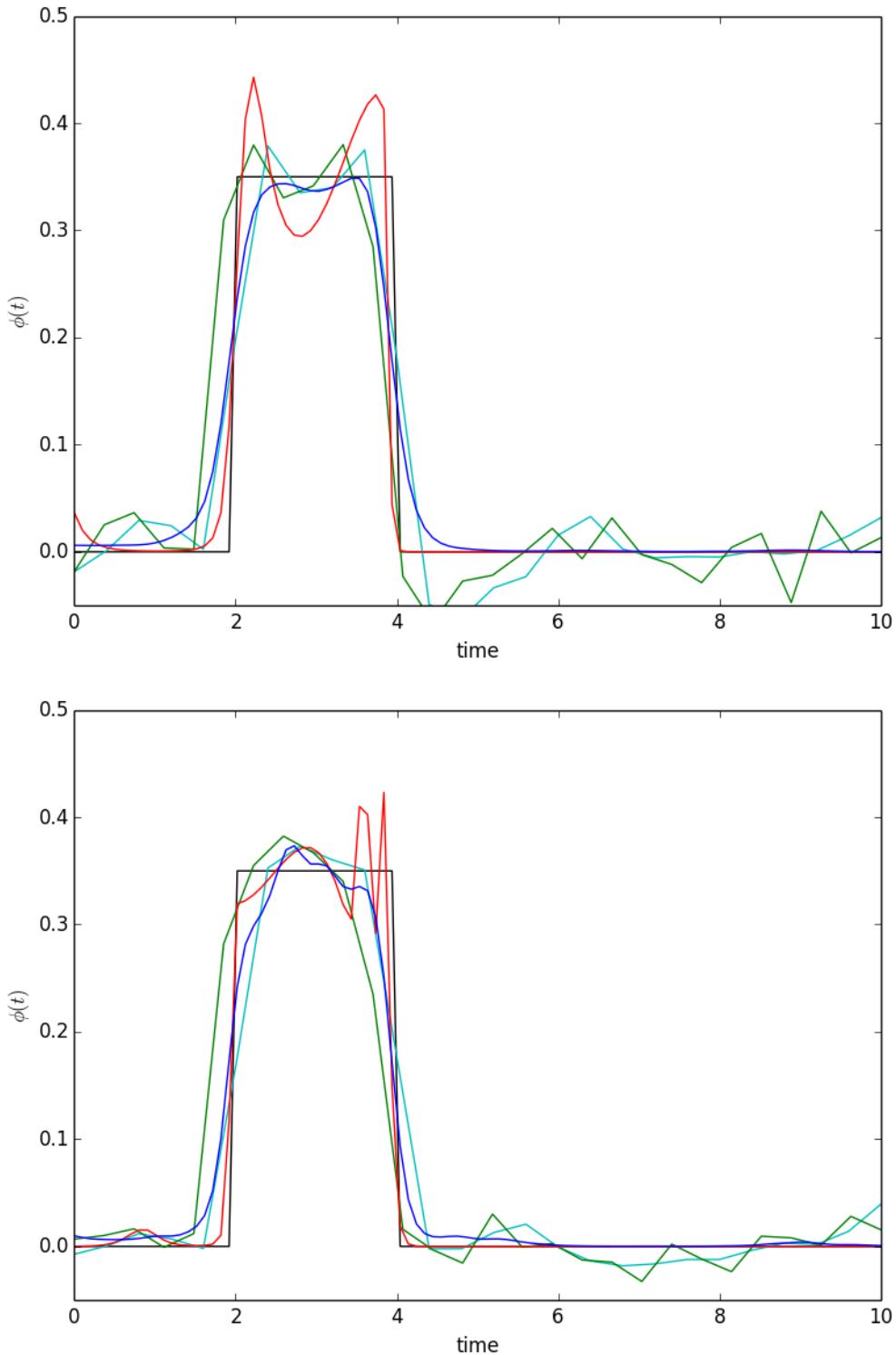


Figure 4.16: Estimated kernels of the models fitted on the data sample of size 2000 (upper plot) and 4000 (bottom plot) generated from Hawkes process with a step function kernel with parameters $[\mu, n] = [0.05, 0.7]$ using the WH method (cyan line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation (blue line). The theoretical kernel (black line) is also given.

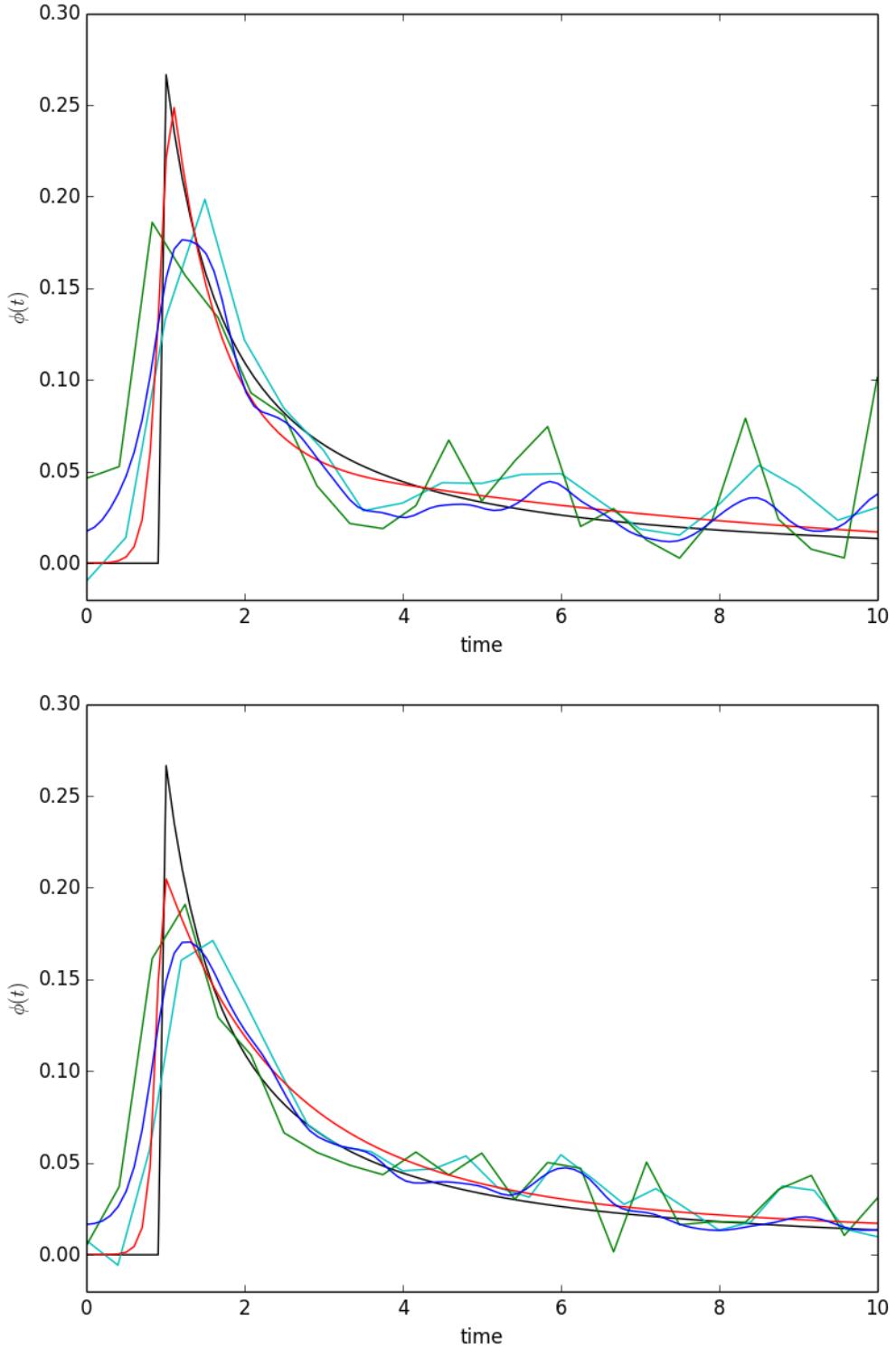


Figure 4.17: Estimated kernels of the models fitted on the data sample of size 2000 (upper plot) and 4000 (bottom plot) generated from Hawkes process with a cut-off memory kernel with parameters $[\mu, n, \tau_0, \epsilon] = [0.05, 0.9, 1, 0.3]$ using the WH method (cyan line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation (blue line). The theoretical kernel (black line) is also given.

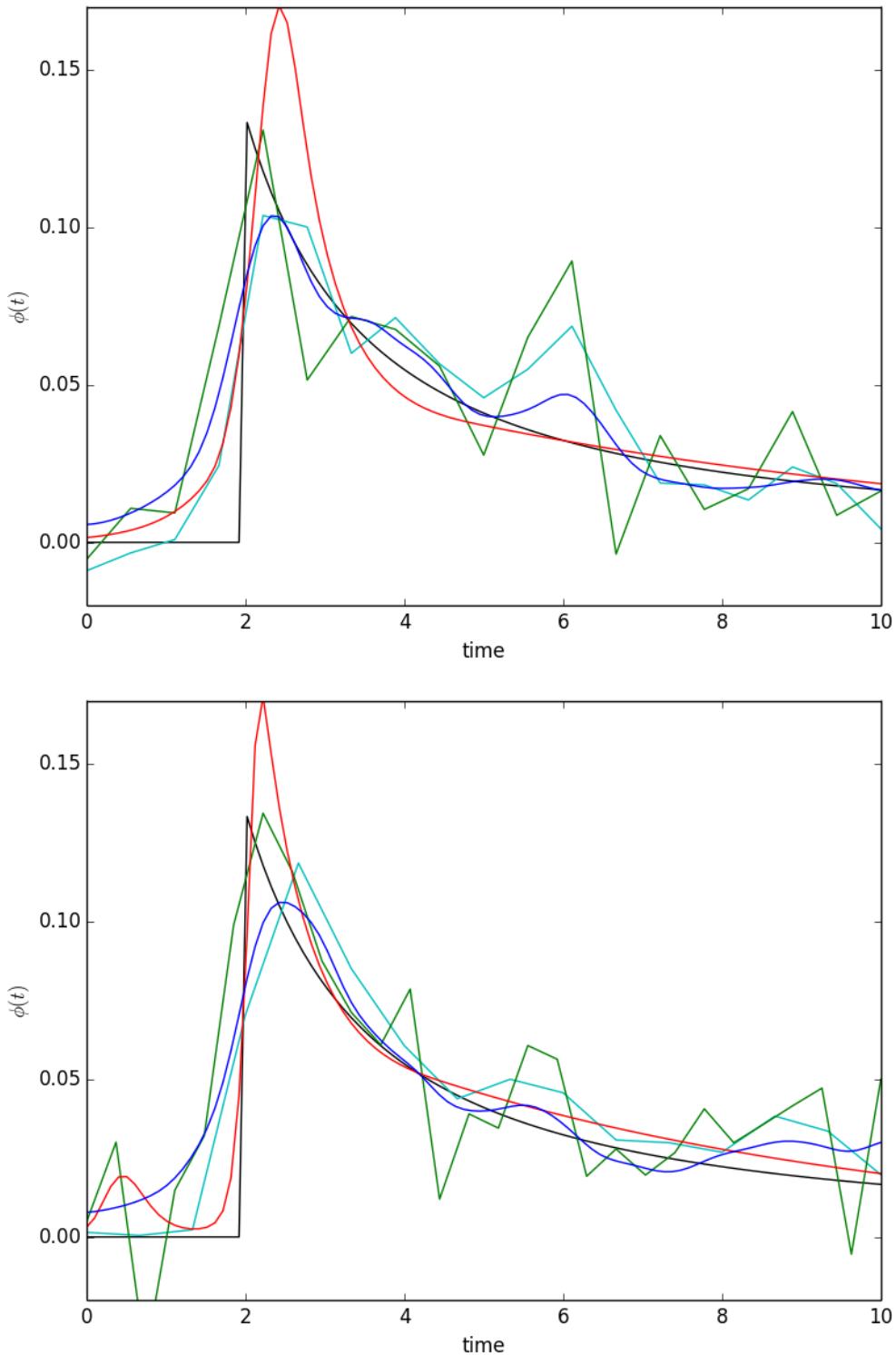


Figure 4.18: Estimated kernels of the models fitted on the data sample of size 2000 (upper plot) and 4000 (bottom plot) generated from Hawkes process with a cut-off memory kernel with parameters $[\mu, n, \tau_0, \epsilon] = [0.05, 0.9, 2, 0.3]$ using the WH method (cyan line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation(blue line). The theoretical kernel (black line) is also given.

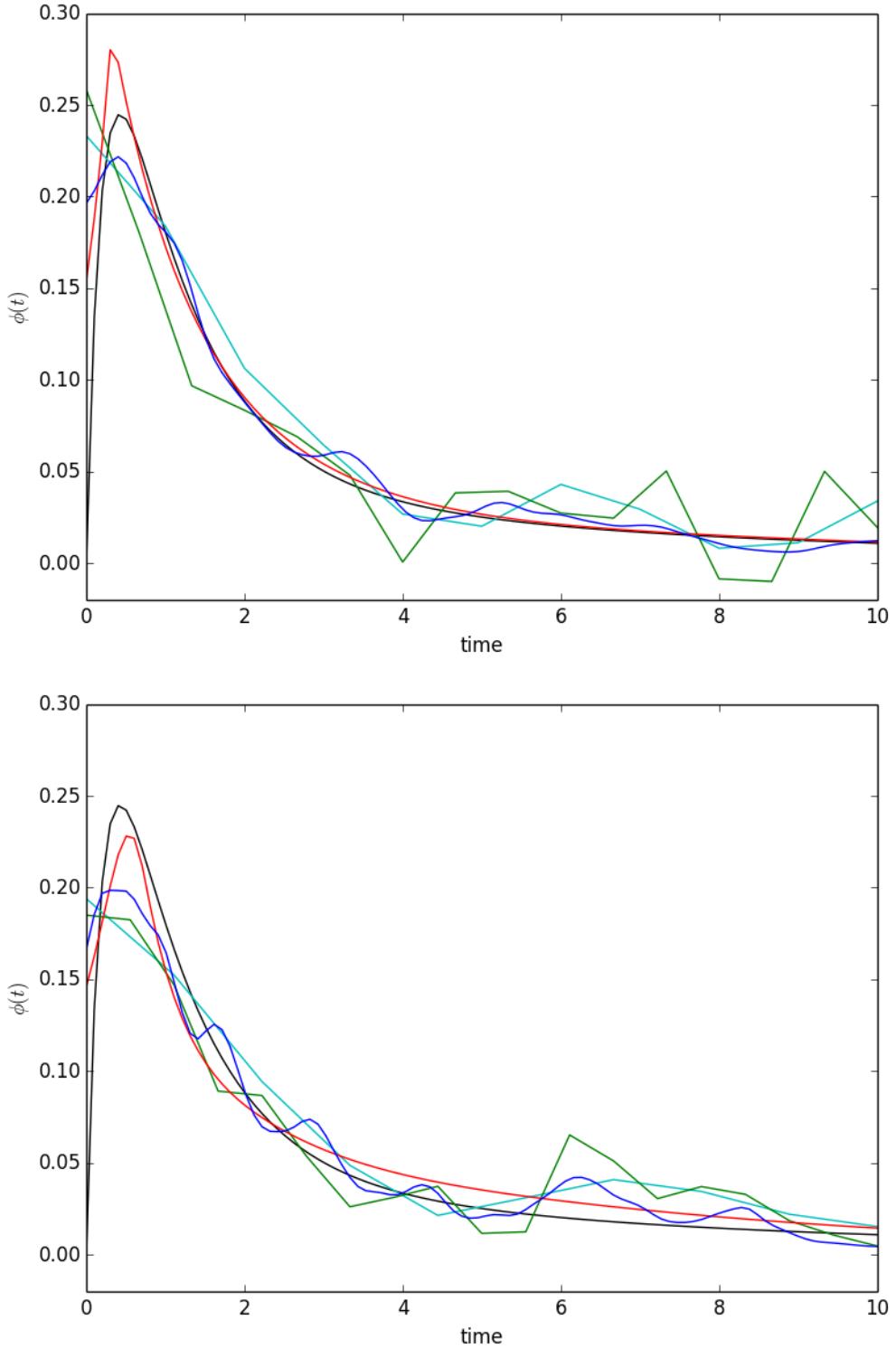


Figure 4.19: Estimated kernels of the models fitted on the data sample of size 2000 (upper plot) and 4000 (bottom plot) generated from Hawkes process with a kernel being sum of exponential functions with parameters $[\mu, n, \tau_0, \epsilon] = [0.05, 0.9, 1, 0.3]$ using the WH method (cyan line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation(blue line). The theoretical kernel (black line) is also given.

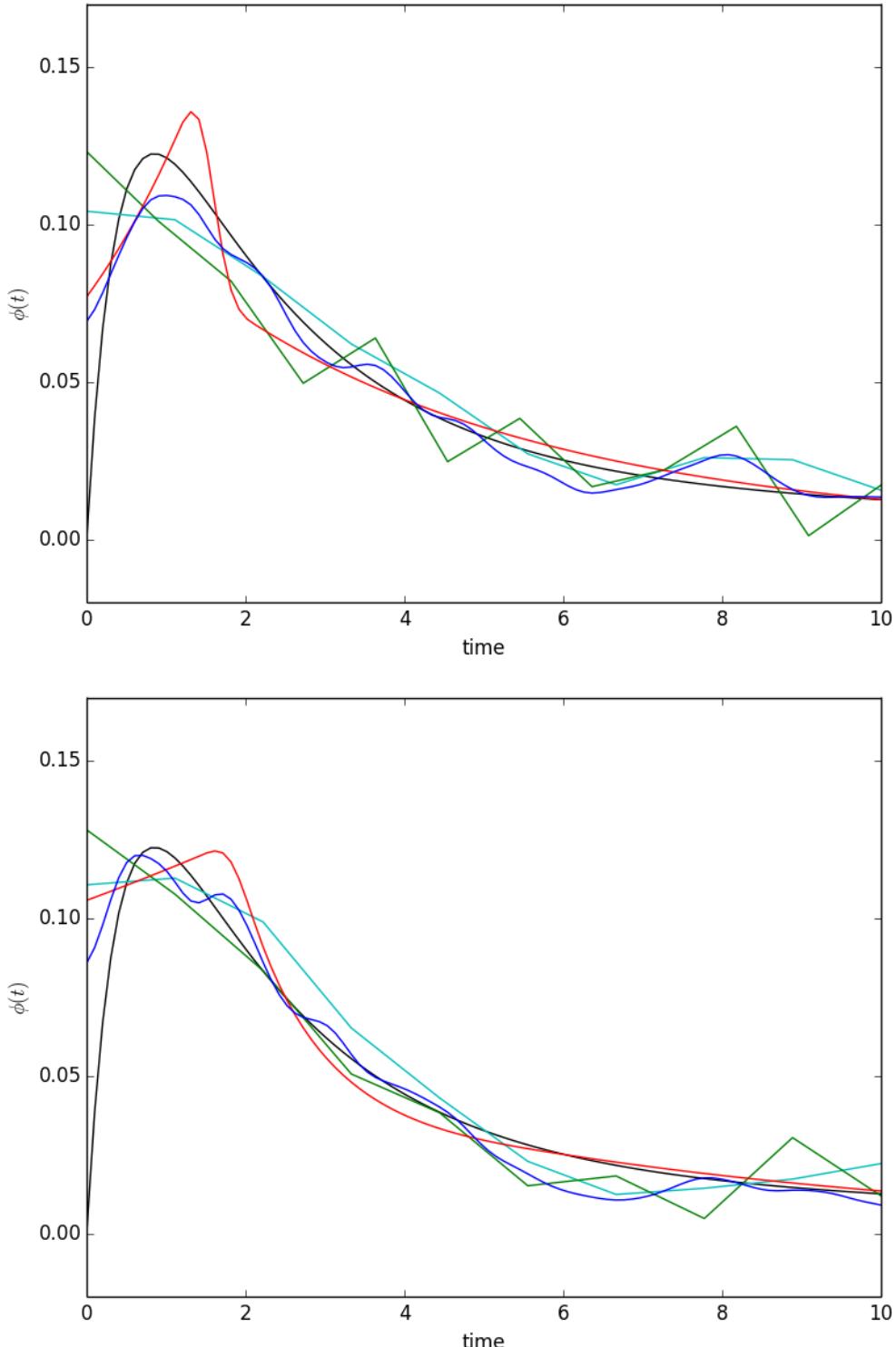


Figure 4.20: Estimated kernels of the models fitted on the data sample of size 2000 (upper plot) and 4000 (bottom plot) generated from Hawkes process with a kernel being sum of exponential functions with parameters $[\mu, n, \tau_0, \epsilon] = [0.05, 0.9, 2, 0.3]$ using the WH method (cyan line), INAR method (green line), EM Algorithm with Logsplines (red line) and EM Algorithm with Kernel Density Estimation(blue line). The theoretical kernel (black line) is also given.

Chapter 5

Comparison of the Estimation Methods on Empirical Data

5.1 Data Source

For the analysis on empirical data I have used high-frequency data of price changes of E-mini S&P 500 (ESc1) and Brent Crude (LCOc1) futures contracts. Futures contracts (also called shortly futures) are one of the most popular financial instruments nowadays. Futures is a contract between two parties to buy or sell an asset at a certain price at a certain time in the future. For this definition and an introduction to theory of financial instruments see ([Hull, 2006](#)). E-mini S&P 500 is a stock market index futures contract traded on the Chicago Mercantile Exchange's Globex electronic trading platform. After its introduction in 1997 E-mini S&P 500 quickly became the most popular index futures contract in the world. Brent Crude futures are the contracts to buy or sell crude oil. They are traded on the electronic Intercontinental Exchange (known as ICE) based in London. For my analysis I have chosen a few trading days from recent years. In order to avoid abnormal market behavior I have tried to skip roll-over periods (i.e. periods shortly before the expiration of a contract) and days of relevant economic events or news releases (such as FED's Monetary Policy Report), when choosing a date of data to analyze. The events that I will look at are the price changes in the time window of one hour. The chosen size of the window allow to collect enough data points for a meaningful analysis and at the same time should allow for stationarity assumption in the time series. Since for a given asset one can distinguish at any time t three different prices: (i) the last transaction price $p_{tr}(t)$, at which the last order was executed, (ii) the best ask price $a(t)$ and (iii) the best bid price $b(t)$, it is not clear what we should consider as the price change. An intuitive choice of the last transaction price p_{tr} might not be a good idea. Let us imagine situation where bid and ask prices of an asset stay constant over some period of time during which buy and sell orders are alternately executed. In such a case we would observe an oscillation of the last transaction price between bid and ask prices, which does not reflect any real changes in price, but is just connected to the order of occurrences of transactions (this phenomena is called bid-ask bounce). In order to avoid this kind of noise I will use a price notion which better reflects price changes namely mid-price (called also mid-quote price) defined as $p(t) = [a(t) + b(t)]/2$. For a discussion of a choice of price changes see ([Filimonov and Sornette, 2012](#)).

5.2 Data Preparation - Randomization Procedure

An important issue when dealing with high-frequency data is a randomization procedure. When one records the high-frequency data there is always a limit of the precision (nowadays on the mili- or even microsecond level). This means that events occurring in a time distance shorter than the precision limit will have the same timestamps. In this way we will observe artificial clustering of the data points only due to the limitations of data recording, which might significantly bias our estimation results. I have addressed this issue by using a randomization procedure suggested in ([Filimonov and Sornette, 2012](#)), which distributes the points with the same timestamps uniformly within a precision limit.

5.3 Residual Analysis

The calibration of the Hawkes model can be validated by goodness of fit using residual analysis introduced in ([Ogata, 1988](#)). The idea is based on the fact that a point process $\{t_i\}_{1 \leq i \leq N}$ with a conditional intensity $\lambda(t)$ can be mapped bijectively to a residual process $\{\tau_i\}_{1 \leq i \leq N}$ defined as

$$\tau_i := \int_0^{t_i} \lambda(t) dt, \quad (5.3.1)$$

which has a distribution of a Poisson process with intensity 1 (see ([Papangelou, 1972](#))). Hence in order to check whether the data can be well explained with a fitted model I will first estimate conditional intensity of the Hawkes model $\hat{\lambda}$ and then compute $\{\hat{\tau}_i\}_{1 \leq i \leq N}$. In order to verify whether obtained residuals follow the Poisson process with intensity 1 we can use following transformation

$$U_i = 1 - \exp(-(\hat{\tau}_i - \hat{\tau}_{i-1})) \quad (5.3.2)$$

and test whether obtained random variables are uniformly distributed in the interval $[0, 1]$.

5.4 Results on Empirical Data

In the figures [5.1-5.20](#) I present the outcomes of estimation. Calibration on the E-mini S&P 500 price changes is shown in figures [5.1-5.10](#), whereas the results for Brent Crude futures data are in figures [5.11-5.20](#). The first point to mention is that even though I have always considered a time window of one hour the size of the samples varies a lot from 843 to 6718 in case of E-mini S&P 500 contracts and from 2133 to 4724 for Brent Crude futures data. Quite interestingly the results for different dates visually look very similar. Most of the obtained curves seem to follow a straight line on the log-log plots for bigger times, what suggests a power law behavior in the tail. I have applied linear regression to recover the slope of the lines. The obtained values of the slopes give us estimates of tail's exponent $\beta := 1 + \theta$ (see [2.2.1](#)). When we look at the estimation outcome of the WH method on a uniform grid (cyan line) it does not seem to perform better than other methods, but when we use the modified WH method on a semi-uniform grid (yellow line) the results improve a lot, allowing much higher precision in the head of the kernel. This suggests that

for histogram type estimation methods counting number of points in bins of a grid (like WH and INAR methods) a semi-uniform grid should be used (or a version of it) instead of a uniform grid, when applied to heavy-tailed distributions. Similarly the EM Algorithm with Kernel Density Estimation (blue line) was not flexible enough to reproduce the head of the kernel well. This is mostly due to the fact that Kernel Density Estimation has been used with a fixed smoothing parameter. One way to improve the results of the EM Algorithm method would be to apply Adaptive Kernel Density Estimation, which could adjust the size of the bandwidth depending on the time scale. This would allow us to have a small bandwidth in the head of the kernel, where there are many points and a large bandwidth in the tail, where the data points are scarce.

In order to further evaluate the results of estimation, I have applied residual analysis, which has been presented on the evenly numbered figures 5.2-5.20. For E-mini S&P 500 contracts the results of the residual analysis look very badly clearly rejecting the models on all the dates (residuals of the modified WH and INAR methods on 10-02-2015 improve, but still are unsatisfactory). The outcomes of the residual analysis for Brent Crude futures look much better. Only models provided by modified WH method on 18-04-2011 and 30-10-2014 fit the data badly. For the other methods the residuals seem to be uniformly distributed, hence indicate models well suiting the data.

Tables 5.1-5.6 collect the estimates of branching ratio n , background intensity μ and tail's exponent β obtained from the WH method on a uniform grid (3.2.9), the WH method on a semi-uniform grid (3.2.10), INAR method and EM Algorithm with Kernel Density Estimation on different dates for E-mini S&P 500 (ESc1) and Brent Crude (LCOc1) futures price changes. Because the data is highly heavy-tailed I was not able to use the EM Algorithm with Logsplines due to problems with convergence of the method. We can notice that the estimates of the tail's exponent vary a lot for each method on different dates. They confirm that data is highly heavy-tailed with the exponent less than 2 in most of the cases. We can notice that for INAR method there are few negative estimates of β , which suggests that estimated kernel is not of power law type. The estimates of branching ratio also differs significantly between different methods. For ESc1 contracts both WH methods estimate parameter n to be around 0.7-0.8. The EM Algorithm suggests higher branching ratio in the range of 0.85-0.9. On the other hand the INAR method estimates \hat{n}_{INAR} are very low at around 0.35-0.4. In the case of LCOc1 contract the branching ratio's estimates are generally lower, being 0.55-0.65 for the WH methods, 0.68-0.75 for the EM Algorithm with Kernel Density Estimation and 0.4-0.5 for the INAR method.

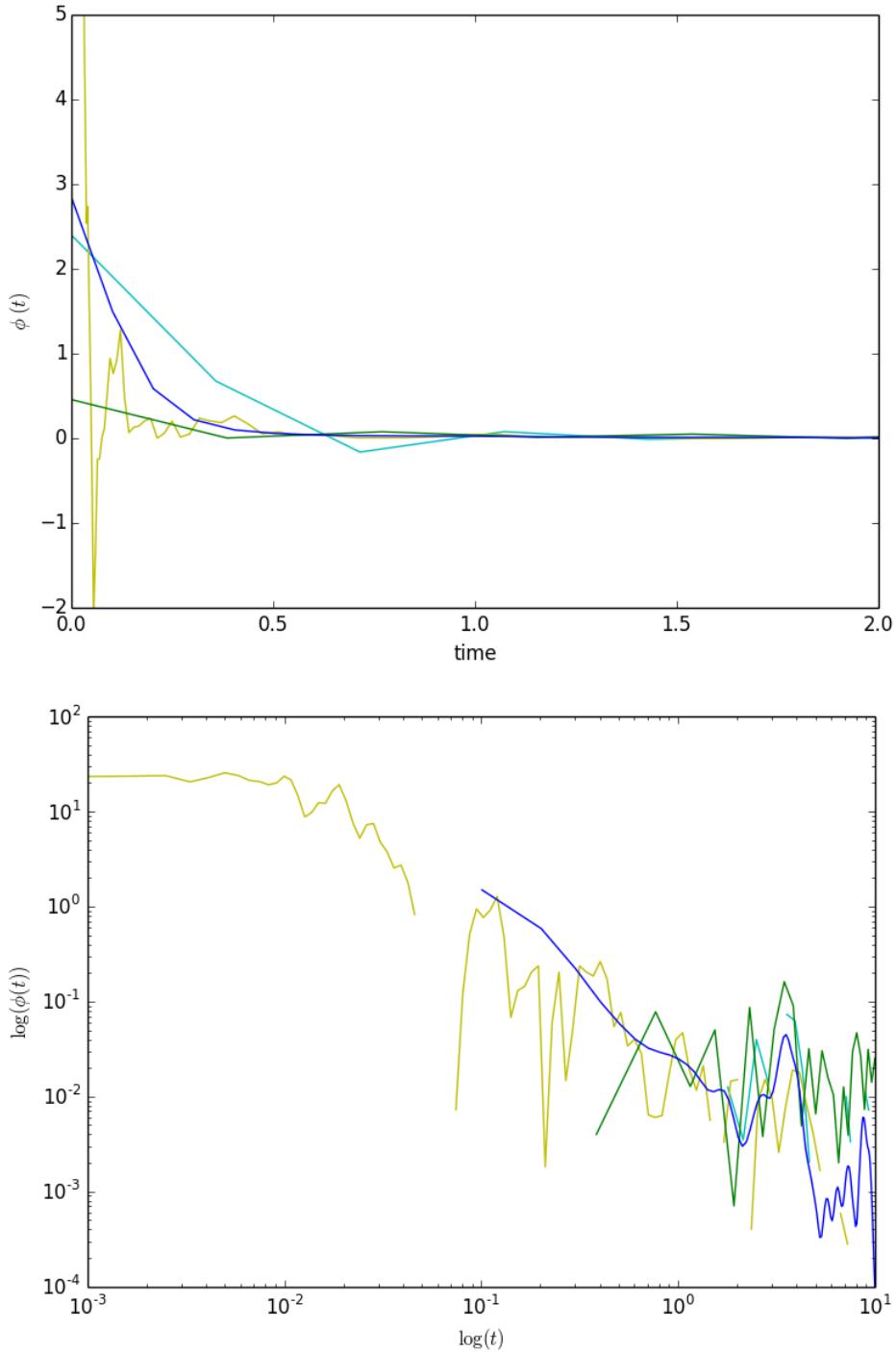


Figure 5.1: The linear-linear (upper plot) and log-log (bottom plot) plots of the estimated kernels fitted on the data sample of size 1664 representing the price changes of a contract ESc1 on 12-05-2011 from 15:00 to 16:00 using the WH method on the uniform grid (cyan line), WH method on the semi-uniform grid (yellow line), INAR method (green line) and EM Algorithm with Kernel Density Estimation (blue line).

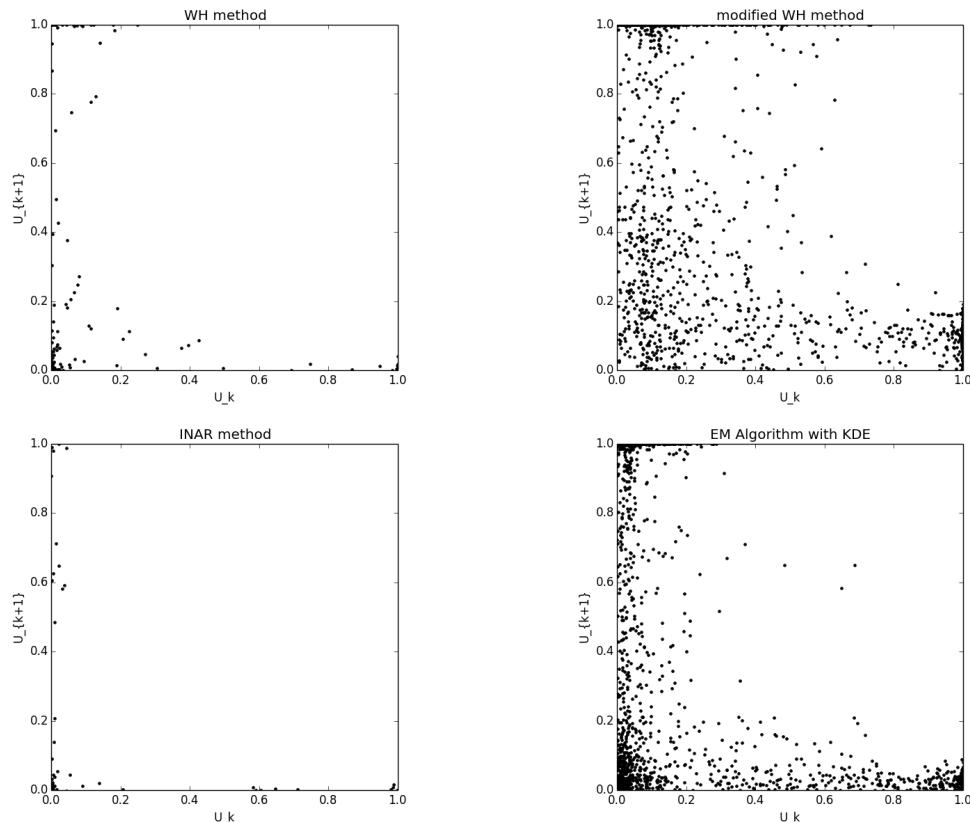


Figure 5.2: The residual plots of the models fitted on the data sample of size 1664 representing the price changes of a contract ESc1 on 12-05-2011 from 15:00 to 16:00 using the WH method on the uniform grid (top left), WH method on the semi-uniform grid (top right), INAR method (bottom left) and EM Algorithm with Kernel Density Estimation (bottom right).

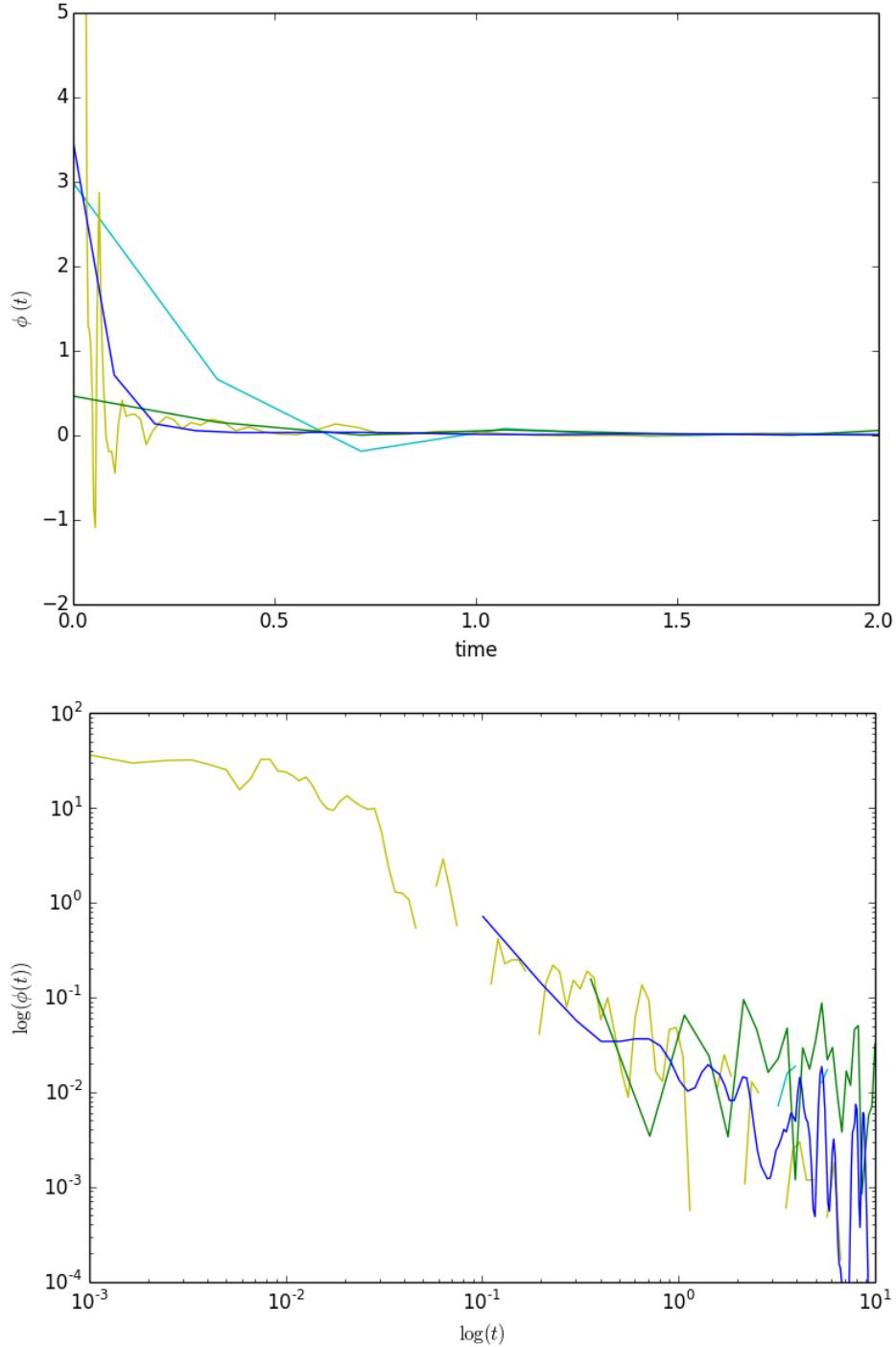


Figure 5.3: The linear-linear (upper plot) and log-log (bottom plot) plots of the estimated kernels fitted on the data sample of size 843 representing the price changes of a contract ESc1 on 18-07-2012 from 19:00 to 20:00 using the WH method on the uniform grid (cyan line), WH method on the semi-uniform grid (yellow line), INAR method (green line) and EM Algorithm with Kernel Density Estimation (blue line).

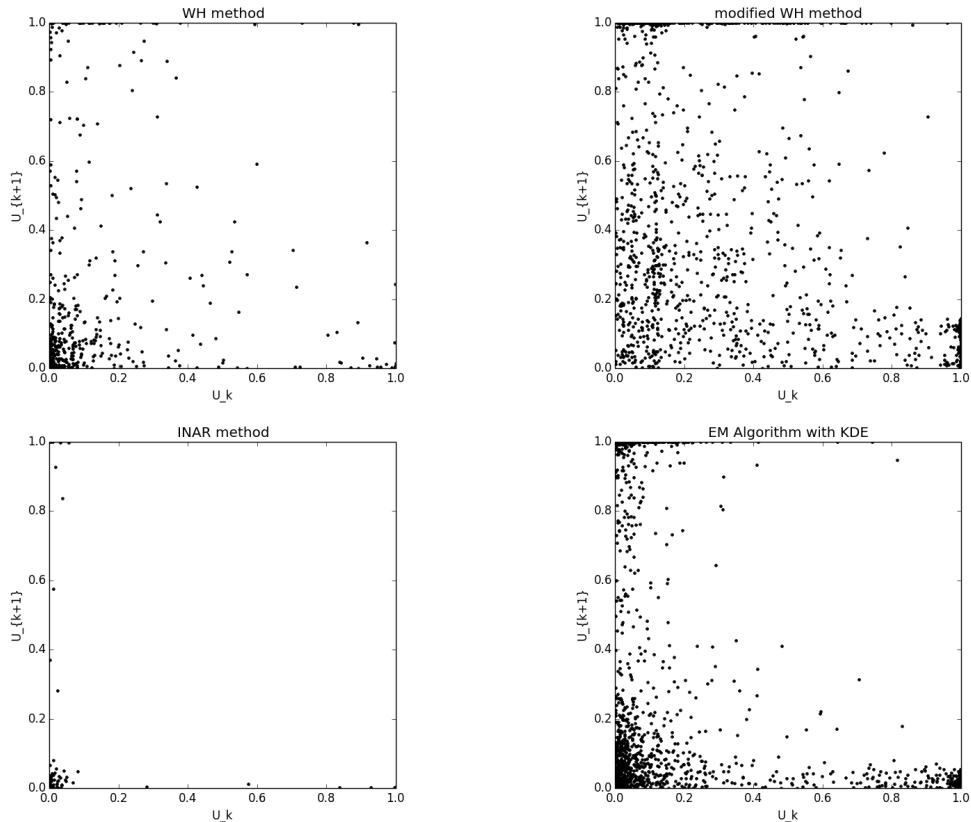


Figure 5.4: The residual plots of the models fitted on the data sample of size 843 representing the price changes of a contract ESc1 on 18-07-2012 from 19:00 to 20:00 using the WH method on the uniform grid (top left), WH method on the semi-uniform grid (top right), INAR method (bottom left) and EM Algorithm with Kernel Density Estimation (bottom right).

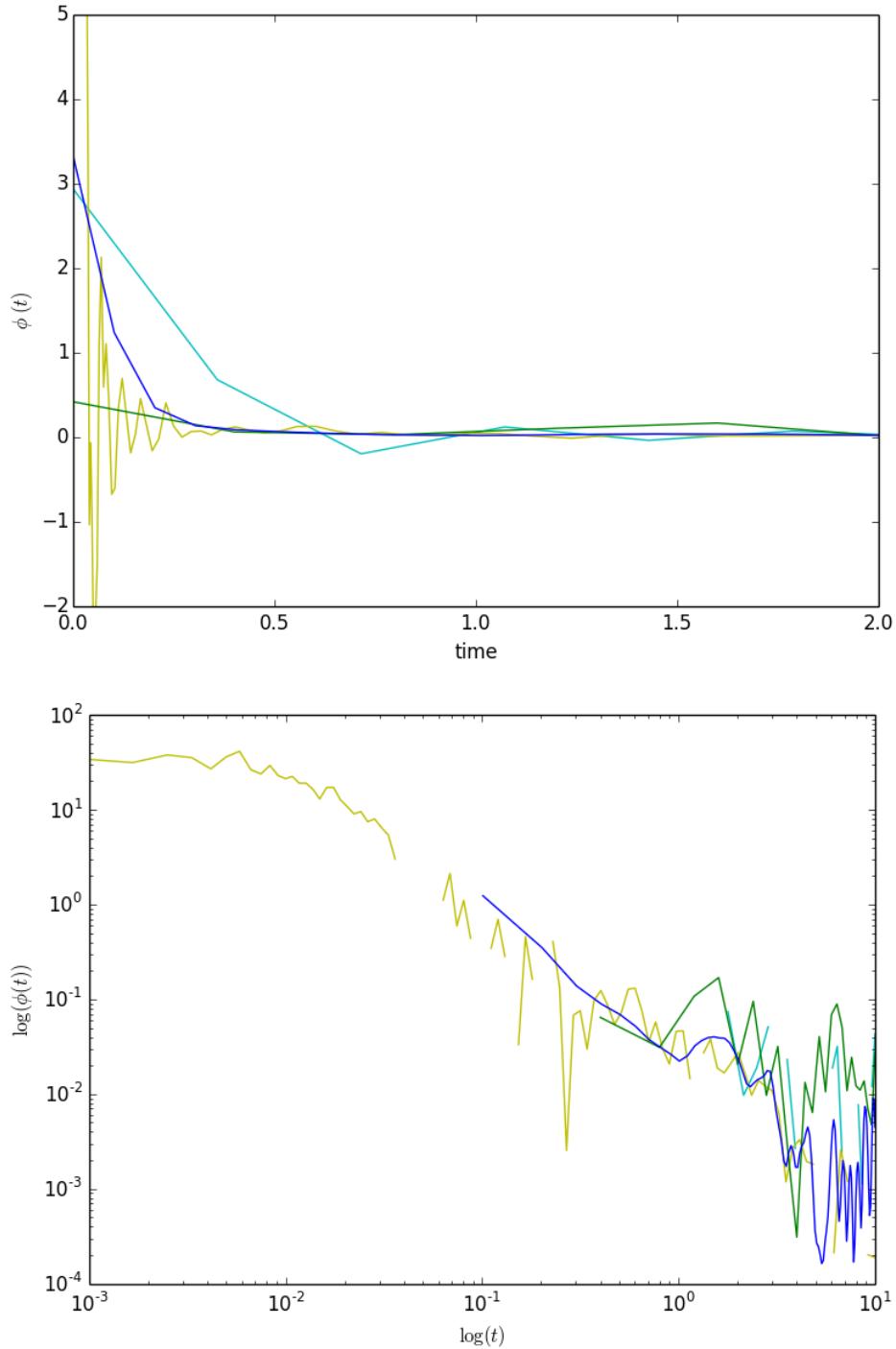


Figure 5.5: The linear-linear (upper plot) and log-log (bottom plot) plots of the estimated kernels fitted on the data sample of size 933 representing the price changes of a contract ESc1 on 24-01-2013 from 17:00 to 18:00 using the WH method on the uniform grid (cyan line), WH method on the semi-uniform grid (yellow line), INAR method (green line) and EM Algorithm with Kernel Density Estimation (blue line).

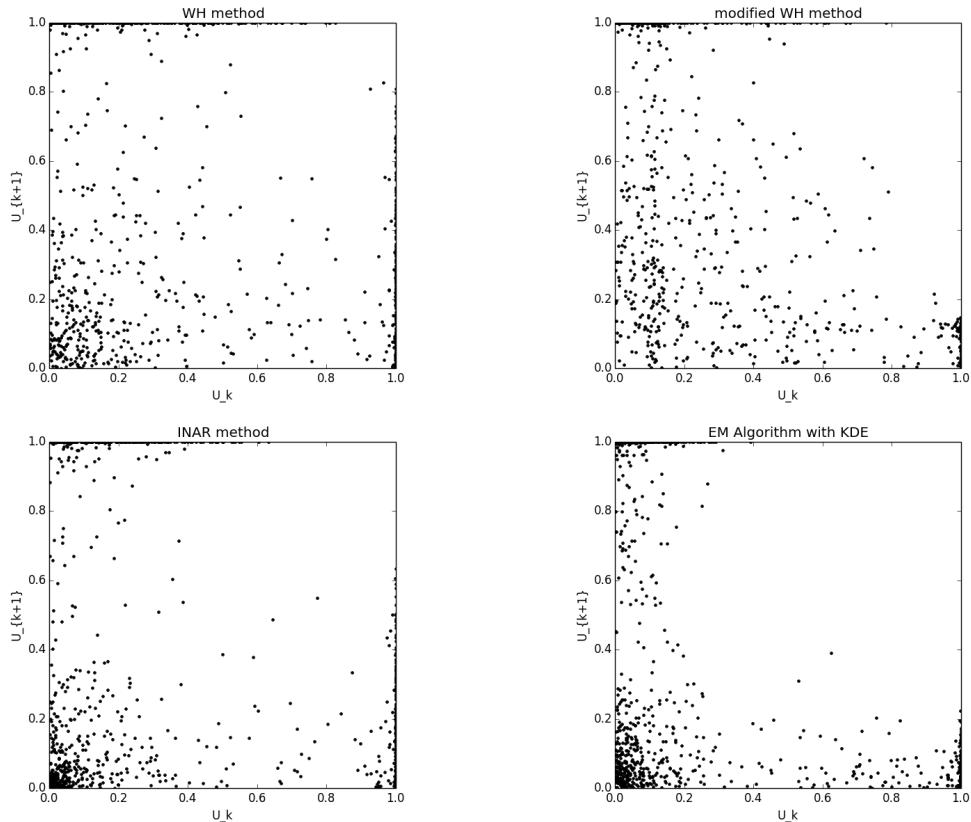


Figure 5.6: The residual plots of the models fitted on the data sample of size 933 representing the price changes of a contract ESc1 on 24-01-2013 from 17:00 to 18:00 using the WH method on the uniform grid (top left), WH method on the semi-uniform grid (top right), INAR method (bottom left) and EM Algorithm with Kernel Density Estimation (bottom right).

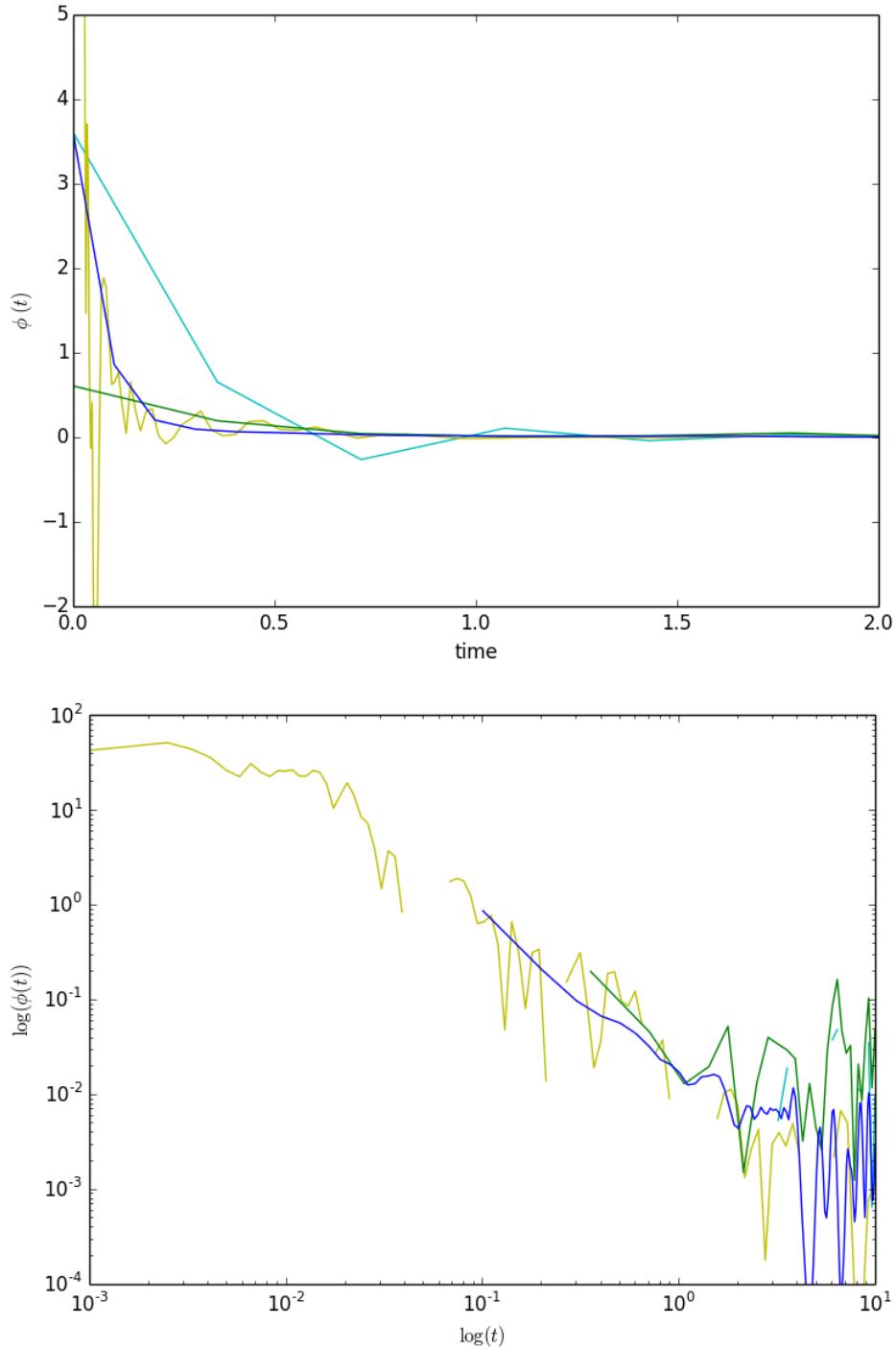


Figure 5.7: The linear-linear (upper plot) and log-log (bottom plot) plots of the estimated kernels fitted on the data sample of size 2202 representing the price changes of a contract ESc1 on 21-11-2014 from 16:00 to 17:00 using the WH method on the uniform grid (cyan line), WH method on the semi-uniform grid (yellow line), INAR method (green line) and EM Algorithm with Kernel Density Estimation (blue line).

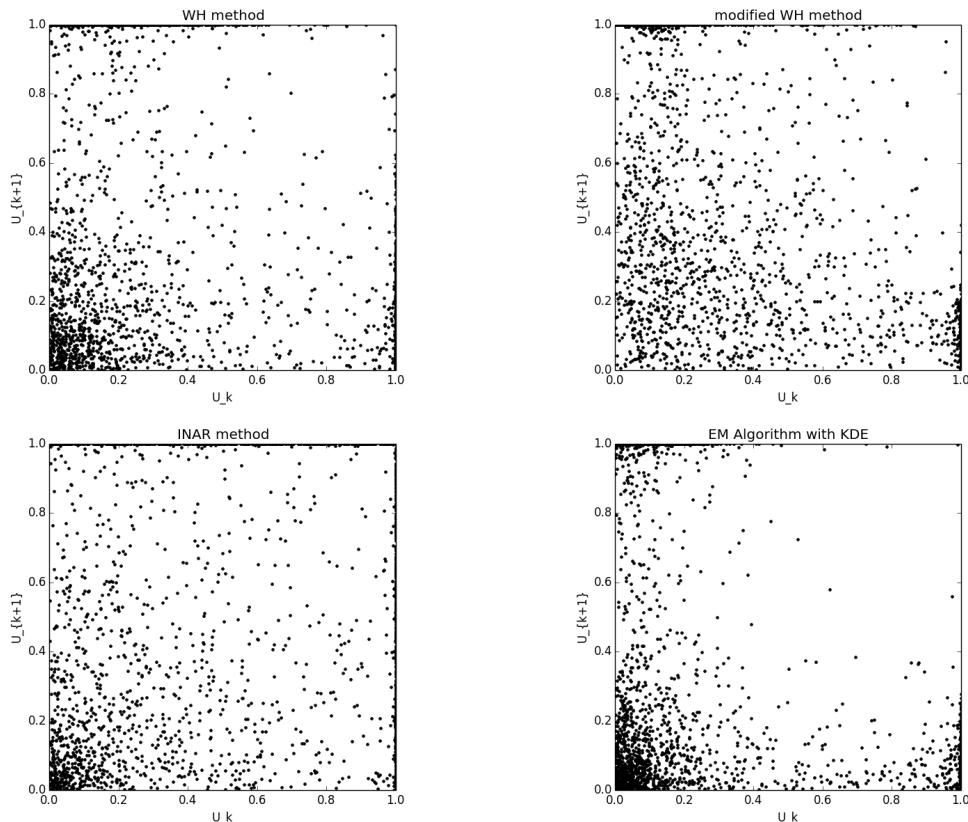


Figure 5.8: The residual plots of the models fitted on the data sample of size 2202 representing the price changes of a contract ESc1 on 21-11-2014 from 16:00 to 17:00 using the WH method on the uniform grid (top left), WH method on the semi-uniform grid (top right), INAR method (bottom left) and EM Algorithm with Kernel Density Estimation (bottom right).

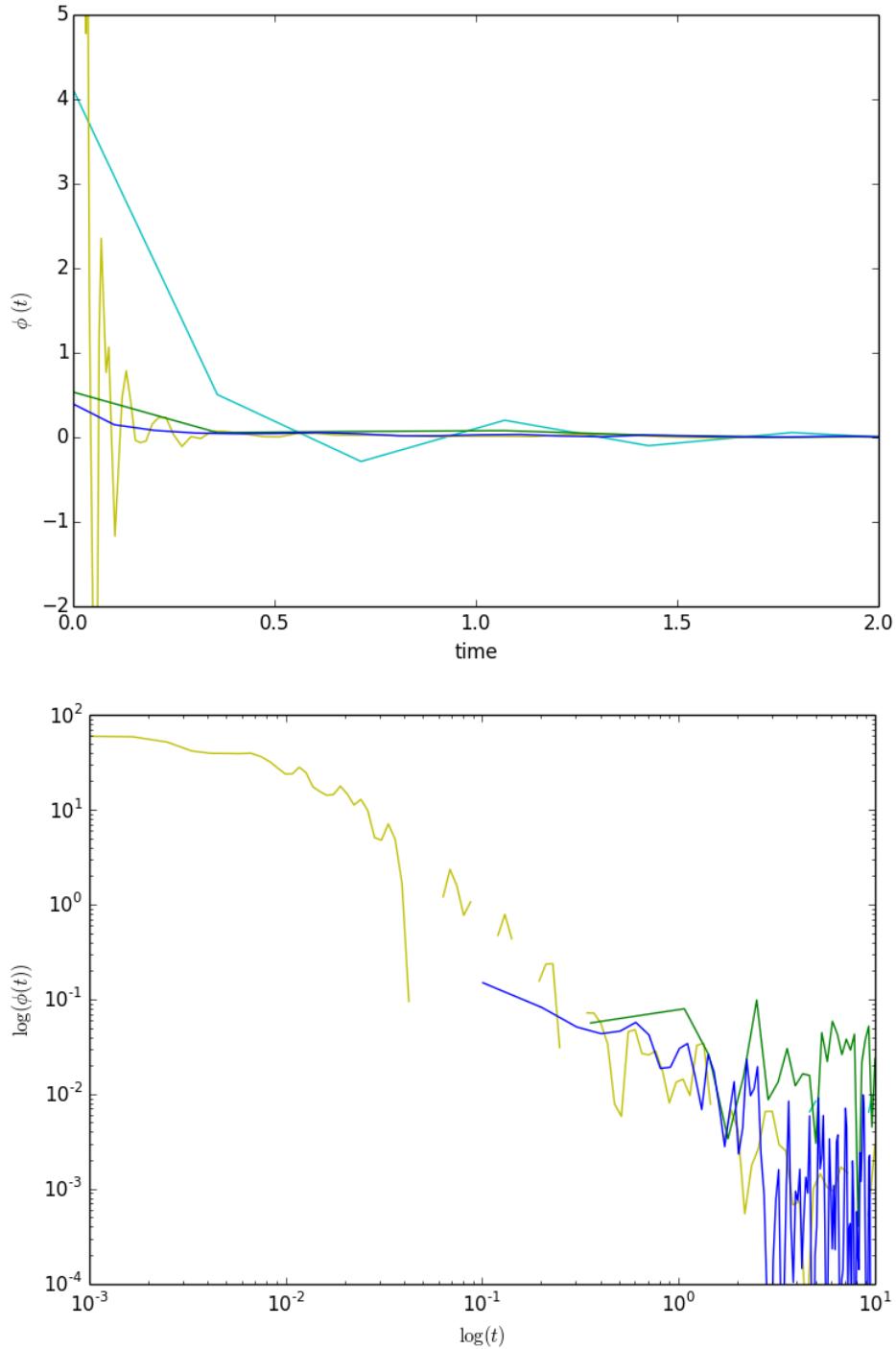


Figure 5.9: The linear-linear (upper plot) and log-log (bottom plot) plots of the estimated kernels fitted on the data sample of size 6718 representing the price changes of a contract ESc1 on 10-02-2015 from 17:00 to 18:00 using the WH method on the uniform grid (cyan line), WH method on the semi-uniform grid (yellow line), INAR method (green line) and EM Algorithm with Kernel Density Estimation (blue line).

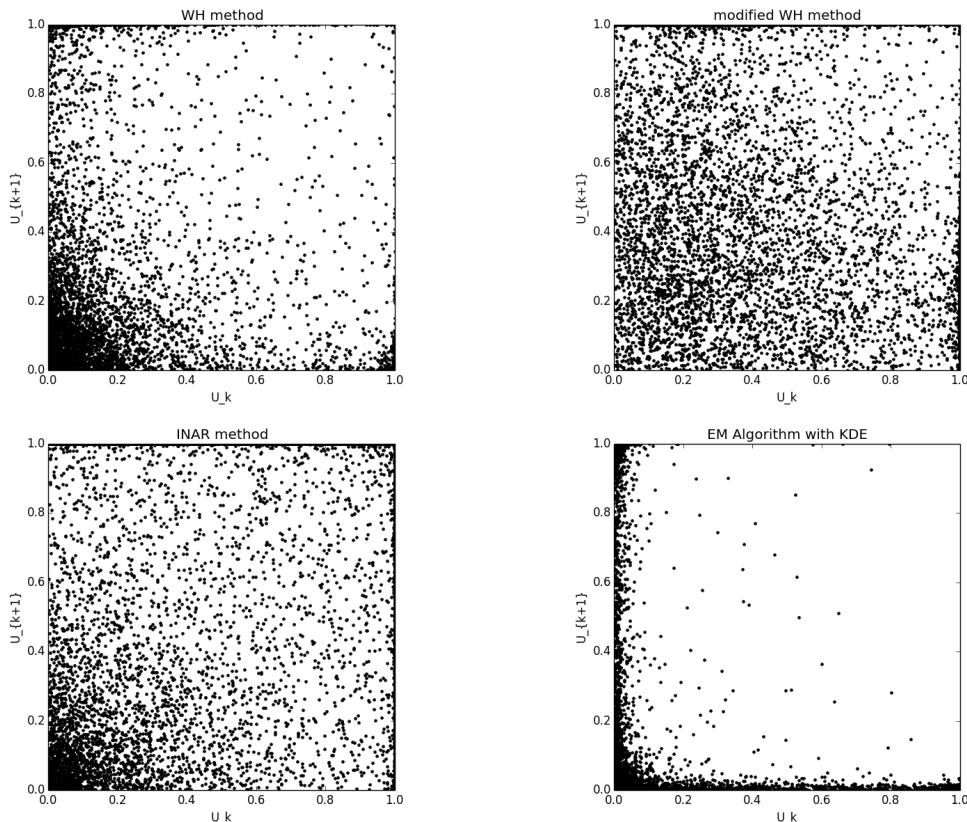


Figure 5.10: The residual plots of the models fitted on the data sample of size 6718 representing the price changes of a contract ESc1 on 10-02-2015 from 17:00 to 18:00 using the WH method on the uniform grid (top left), WH method on the semi-uniform grid (top right), INAR method (bottom left) and EM Algorithm with Kernel Density Estimation (bottom right).

ESc1	\hat{n}_{WH}	\hat{n}_{WH-mod}	\hat{n}_{INAR}	\hat{n}_{EMKDE}
12-05-2011 15:00-16:00	0.69	0.62	0.32	0.84
18-07-2012 19:00-20:00	0.74	0.68	0.27	0.84
24-01-2013 17:00-18:00	0.79	0.71	0.37	0.87
21-11-2014 16:00-17:00	0.85	0.77	0.43	0.88
10-05-2015 17:00-18:00	0.89	0.8	0.4	0.9

Table 5.1: Branching ratio estimated on the E-mini S&P 500 futures contract data on a given dates using the WH method on the uniform grid ("WH"), modified WH method on the semi-uniform grid ("WH-mod"), INAR method ("INAR") and EM Algorithm with Kernel Density Estimation ("EMKDE").

ESc1	$\hat{\mu}_{WH}$	$\hat{\mu}_{WH-mod}$	$\hat{\mu}_{INAR}$	$\hat{\mu}_{EMKDE}$
12-05-2011 15:00-16:00	0.14	0.17	0.31	0.07
18-07-2012 19:00-20:00	0.11	0.14	0.31	0.07
24-01-2013 17:00-18:00	0.05	0.08	0.16	0.03
21-11-2014 16:00-17:00	0.09	0.14	0.35	0.07
10-05-2015 17:00-18:00	0.2	0.37	1.12	0.19

Table 5.2: Background intensity estimated on the E-mini S&P 500 futures contract data on a given dates using the WH method on the uniform grid ("WH"), modified WH method on the semi-uniform grid ("WH-mod"), INAR method ("INAR") and EM Algorithm with Kernel Density Estimation ("EMKDE").

ESc1	$\hat{\beta}_{WH}$	$\hat{\beta}_{WH-mod}$	$\hat{\beta}_{INAR}$	$\hat{\beta}_{EMKDE}$
12-05-2011 15:00-16:00	0.7	1.32	-0.18	1.62
18-07-2012 19:00-20:00	0.67	1.32	0.57	2.56
24-01-2013 17:00-18:00	1.1	2.43	0.5	1.79
21-11-2014 16:00-17:00	0.62	0.81	-0.4	1.14
10-05-2015 17:00-18:00	1.14	1.6	-0.22	2.2

Table 5.3: Estimates of the tail's exponent obtained on the E-mini S&P 500 futures contract data on a given dates using the WH method on the uniform grid ("WH"), modified WH method on the semi-uniform grid ("WH-mod"), INAR method ("INAR") and EM Algorithm with Kernel Density Estimation ("EMKDE").

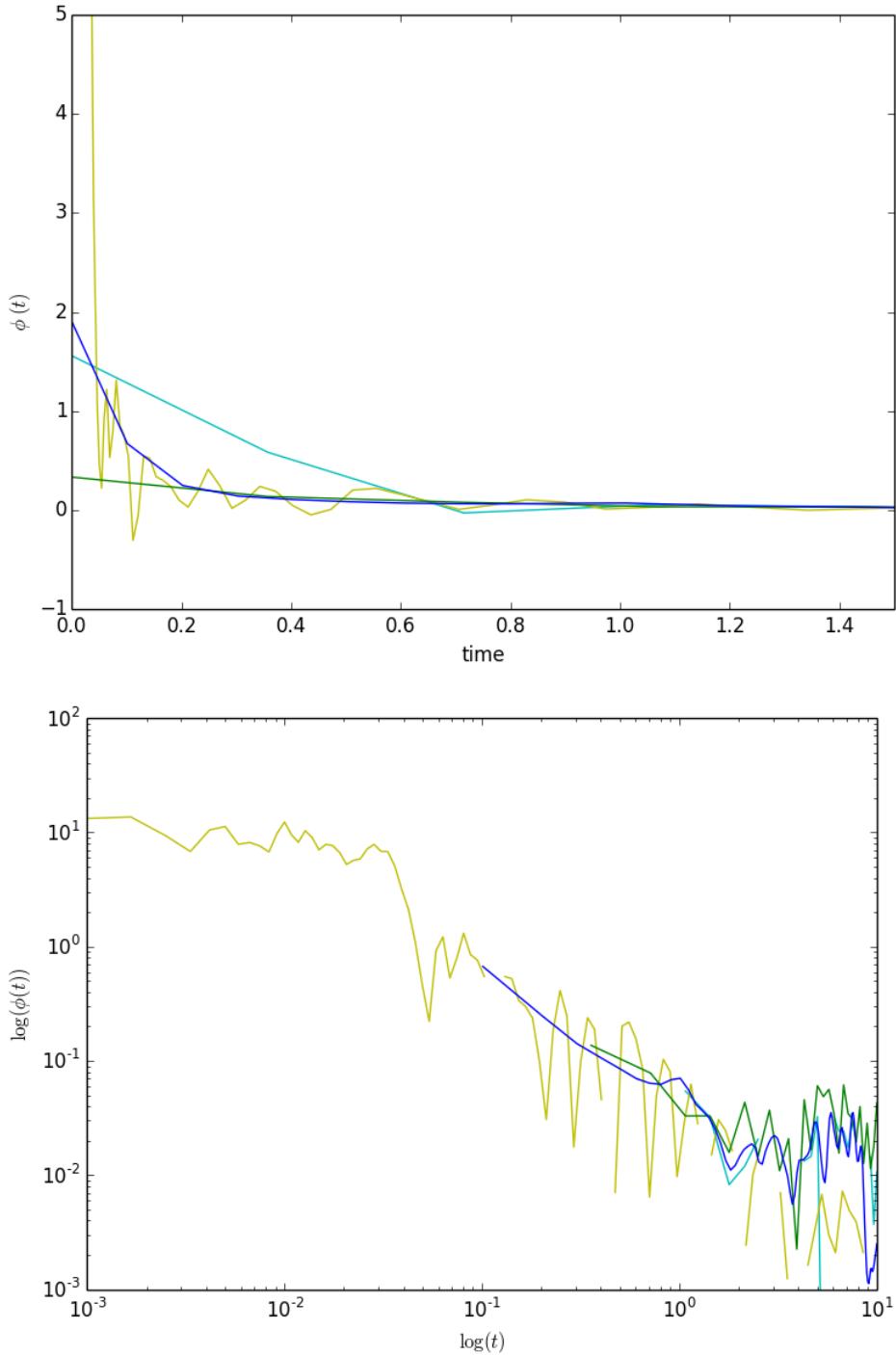


Figure 5.11: The linear-linear (upper plot) and log-log (bottom plot) plots of the estimated kernels fitted on the data sample of size 2133 representing the price changes of a contract LCOc1 on 18-04-2011 from 17:00 to 18:00 using the WH method on the uniform grid (cyan line), WH method on the semi-uniform grid (yellow line), INAR method (green line) and EM Algorithm with Kernel Density Estimation (blue line).

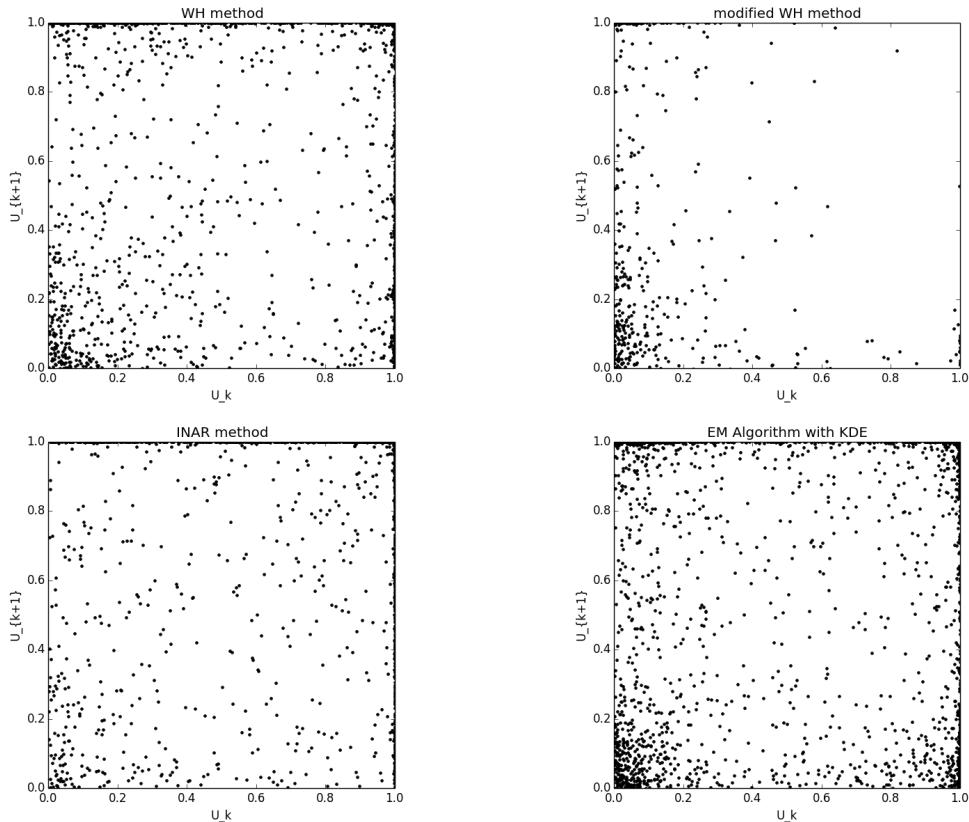


Figure 5.12: The residual plots of the models fitted on the data sample of size 2133 representing the price changes of a contract LCOc1 on 18-04-2011 from 17:00 to 18:00 using the WH method on the uniform grid (top left), WH method on the semi-uniform grid (top right), INAR method (bottom left) and EM Algorithm with Kernel Density Estimation (bottom right).

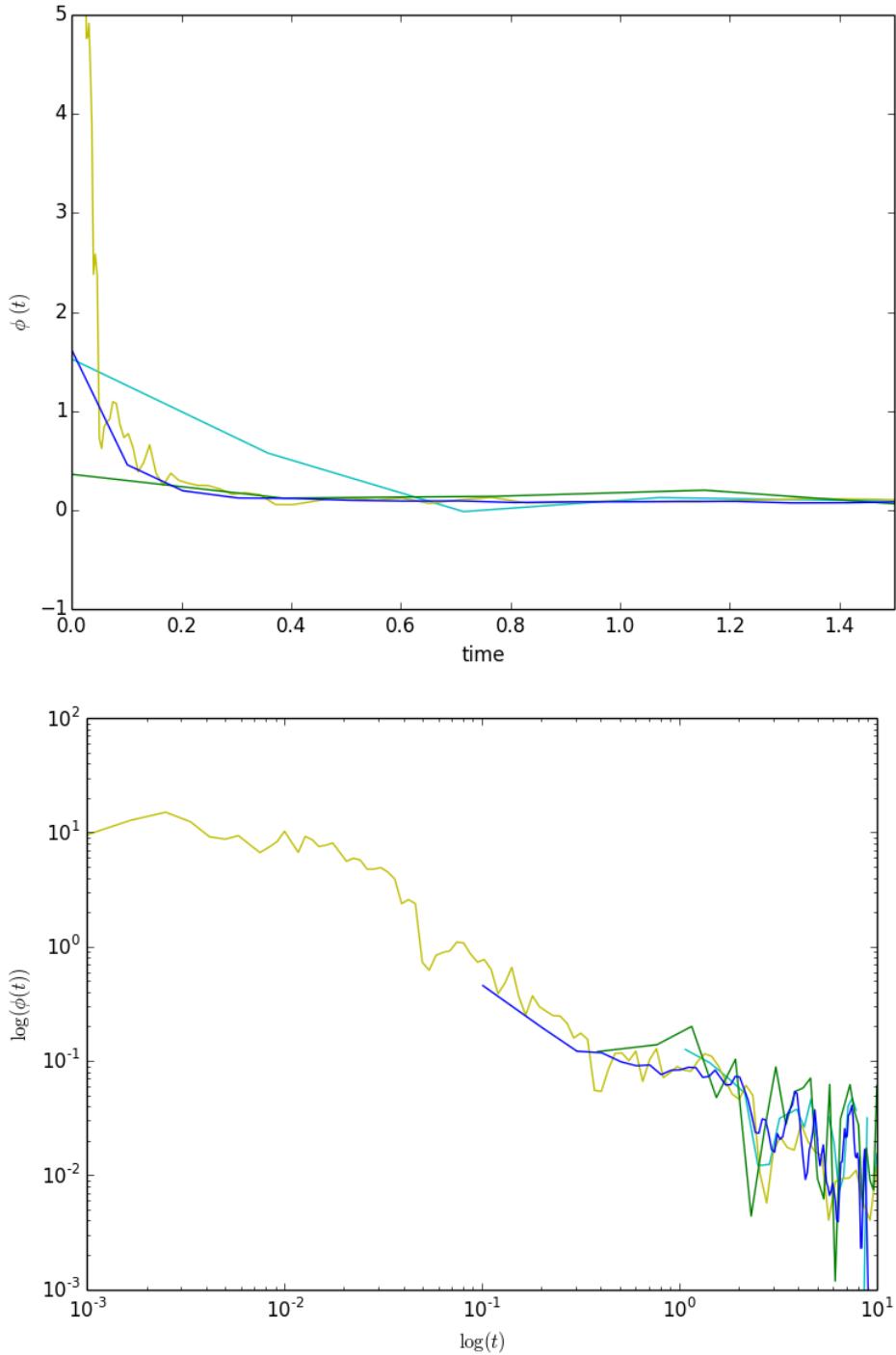


Figure 5.13: The linear-linear (upper plot) and log-log (bottom plot) plots of the estimated kernels fitted on the data sample of size 4724 representing the price changes of a contract LCOc1 on 22-03-2012 from 15:00 to 16:00 using the WH method on the uniform grid (cyan line), WH method on the semi-uniform grid (yellow line), INAR method (green line) and EM Algorithm with Kernel Density Estimation (blue line).

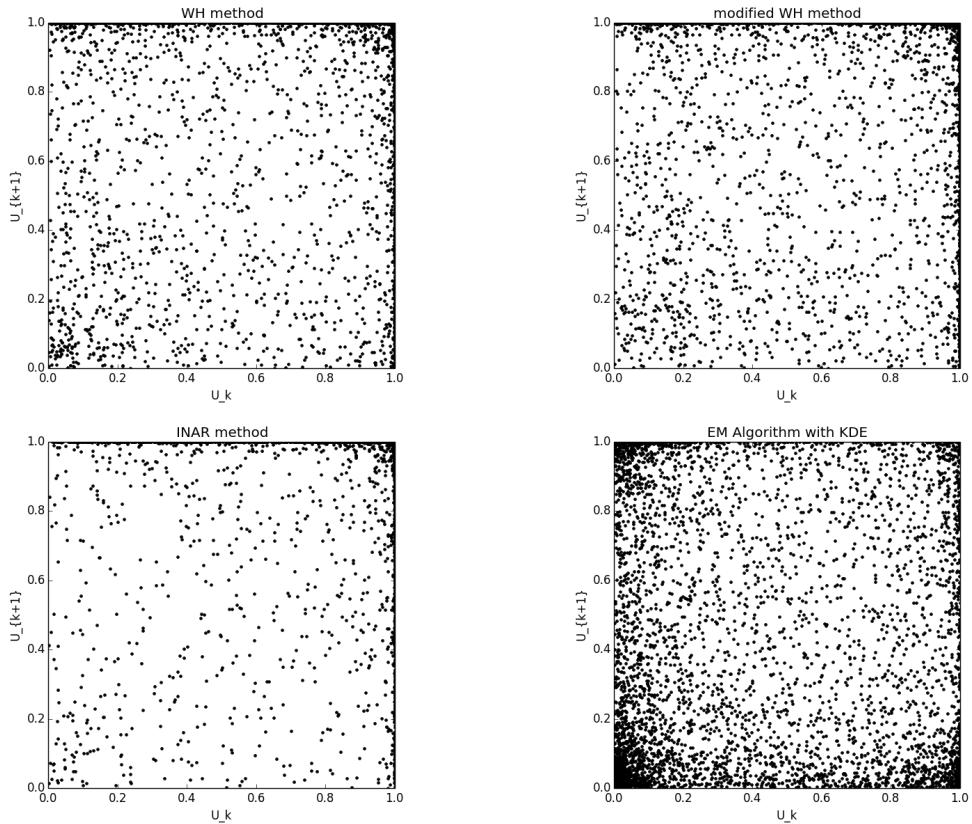


Figure 5.14: The residual plots of the models fitted on the data sample of size 4724 representing the price changes of a contract LCOc1 on 22-03-2012 from 15:00 to 16:00 using the WH method on the uniform grid (top left), WH method on the semi-uniform grid (top right), INAR method (bottom left) and EM Algorithm with Kernel Density Estimation (bottom right).

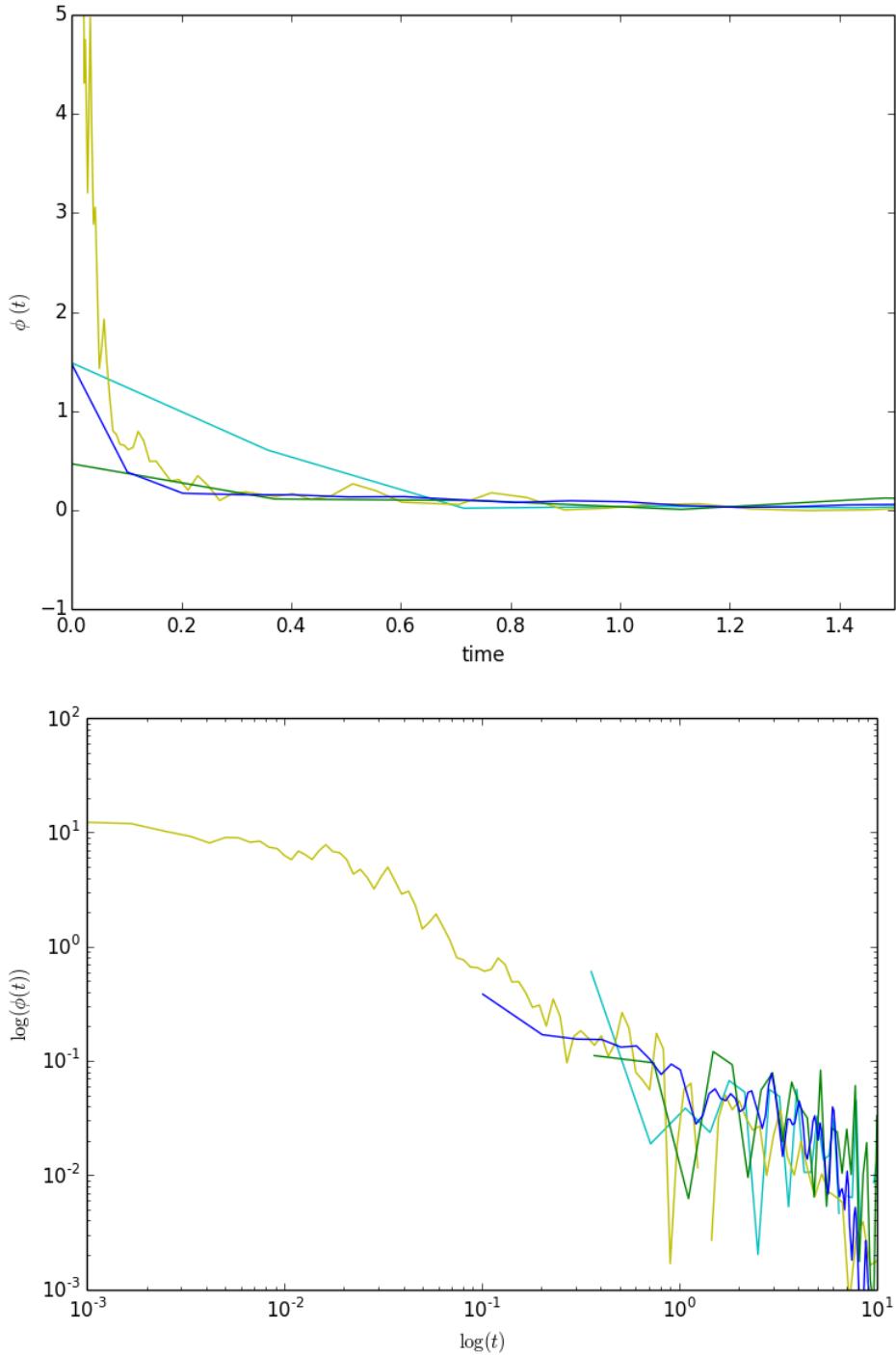


Figure 5.15: The linear-linear (upper plot) and log-log (bottom plot) plots of the estimated kernels fitted on the data sample of size 3174 representing the price changes of a contract LCOc1 on 05-07-2013 from 16:00 to 17:00 using the WH method on the uniform grid (cyan line), WH method on the semi-uniform grid (yellow line), INAR method (green line) and EM Algorithm with Kernel Density Estimation (blue line).

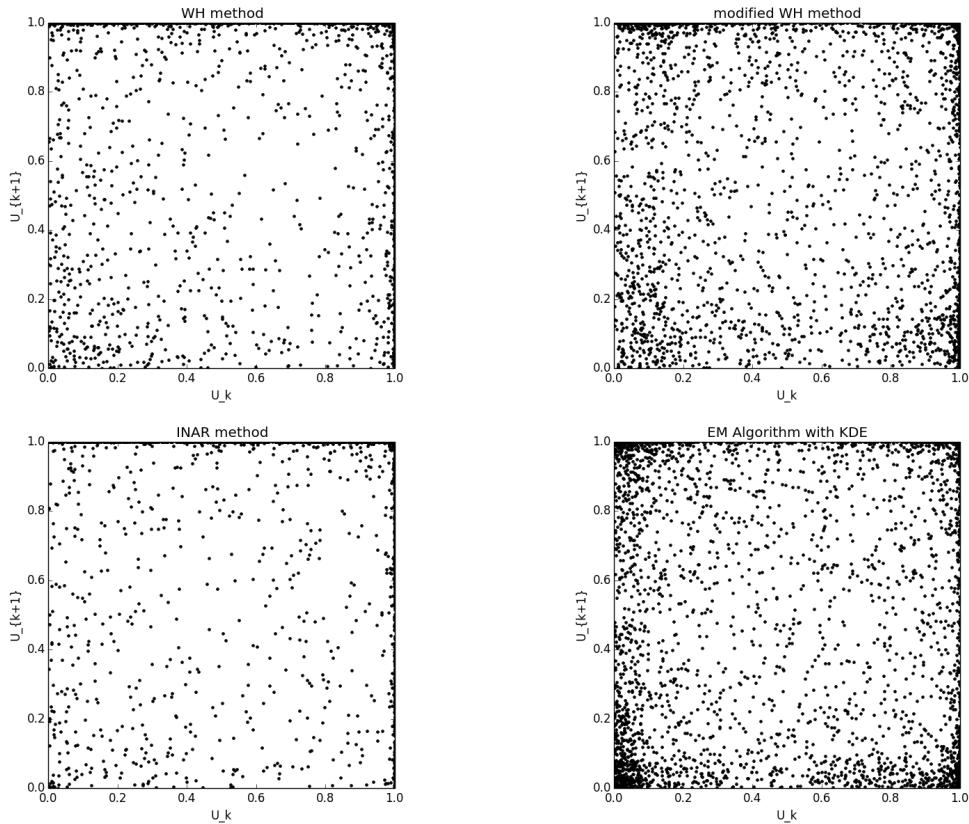


Figure 5.16: The residual plots of the models fitted on the data sample of size 3174 representing the price changes of a contract LCOc1 on 05-07-2013 from 16:00 to 17:00 using the WH method on the uniform grid (top left), WH method on the semi-uniform grid (top right), INAR method (bottom left) and EM Algorithm with Kernel Density Estimation (bottom right).

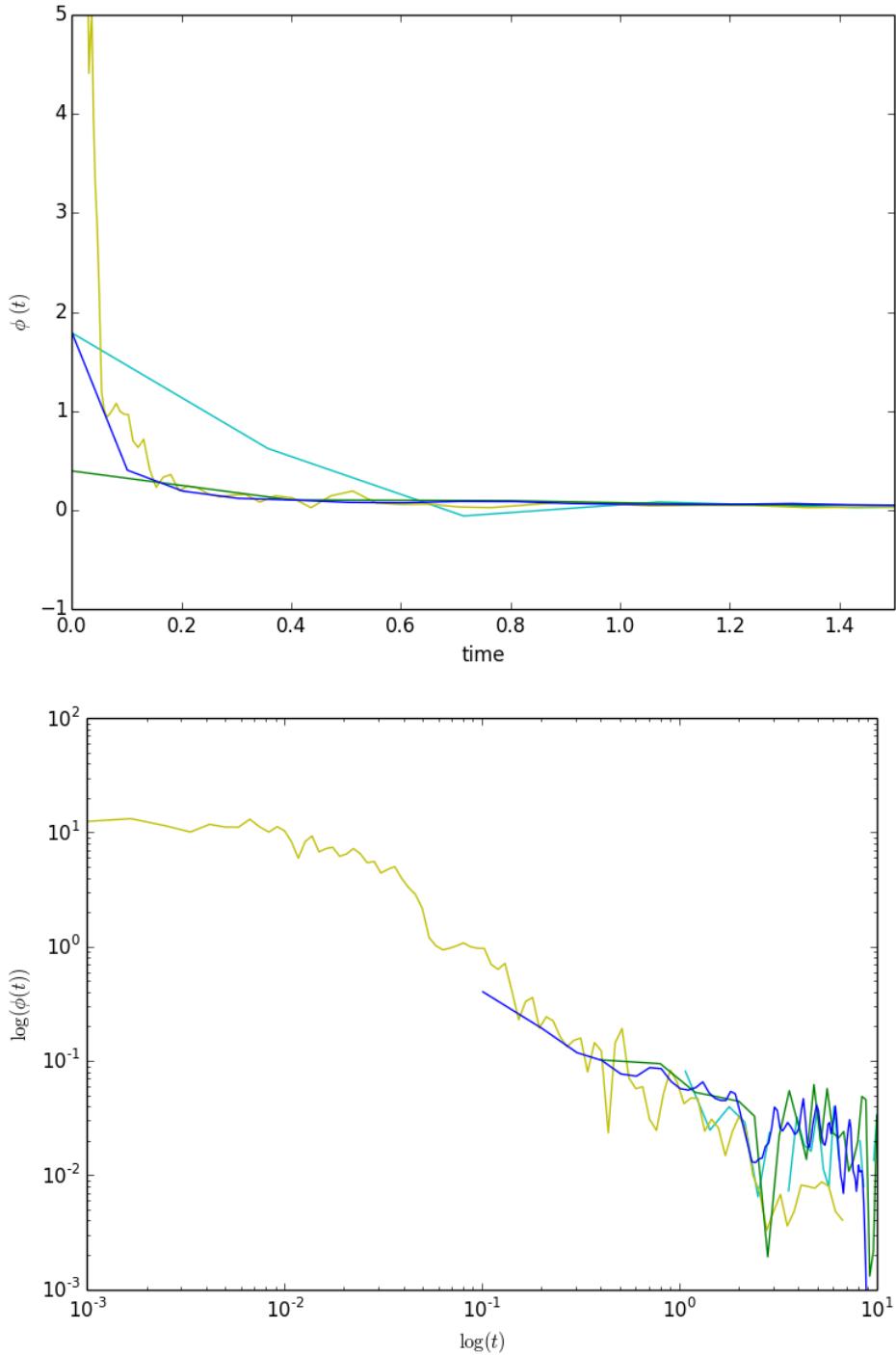


Figure 5.17: The linear-linear (upper plot) and log-log (bottom plot) plots of the estimated kernels fitted on the data sample of size 4190 representing the price changes of a contract LCOc1 on 30-10-2014 from 15:00 to 16:00 using the WH method on the uniform grid (cyan line), WH method on the semi-uniform grid (yellow line), INAR method (green line) and EM Algorithm with Kernel Density Estimation (blue line).

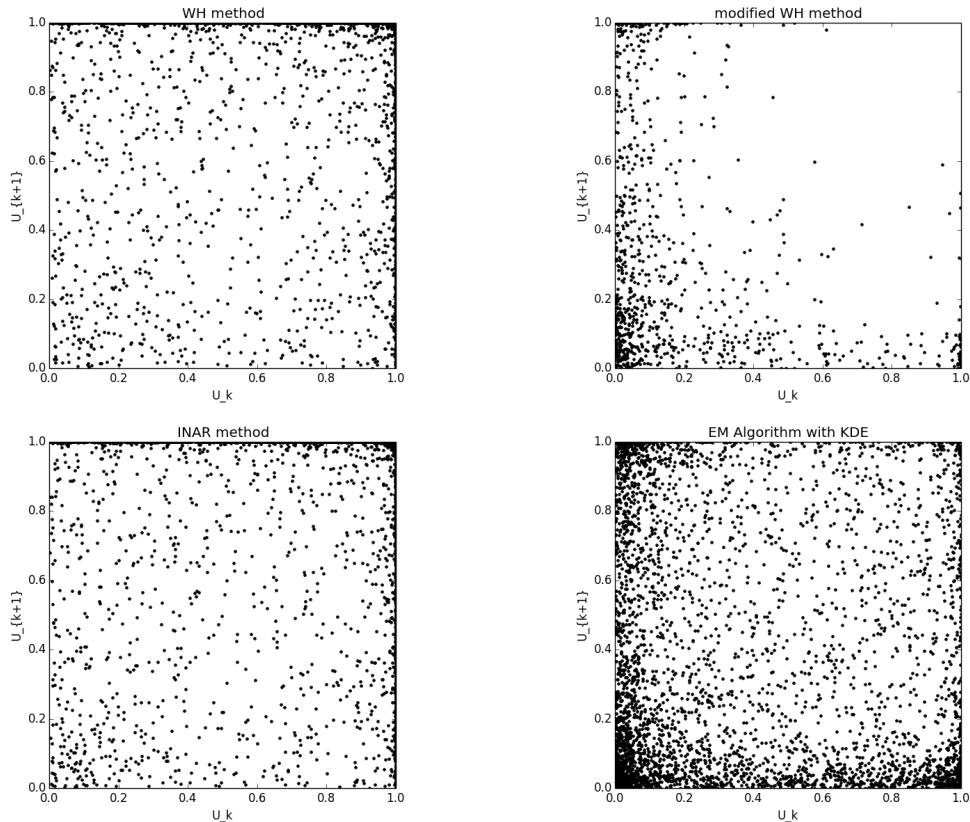


Figure 5.18: The residual plots of the models fitted on the data sample of size 4190 representing the price changes of a contract LCOc1 on 30-10-2014 from 15:00 to 16:00 using the WH method on the uniform grid (top left), WH method on the semi-uniform grid (top right), INAR method (bottom left) and EM Algorithm with Kernel Density Estimation (bottom right).

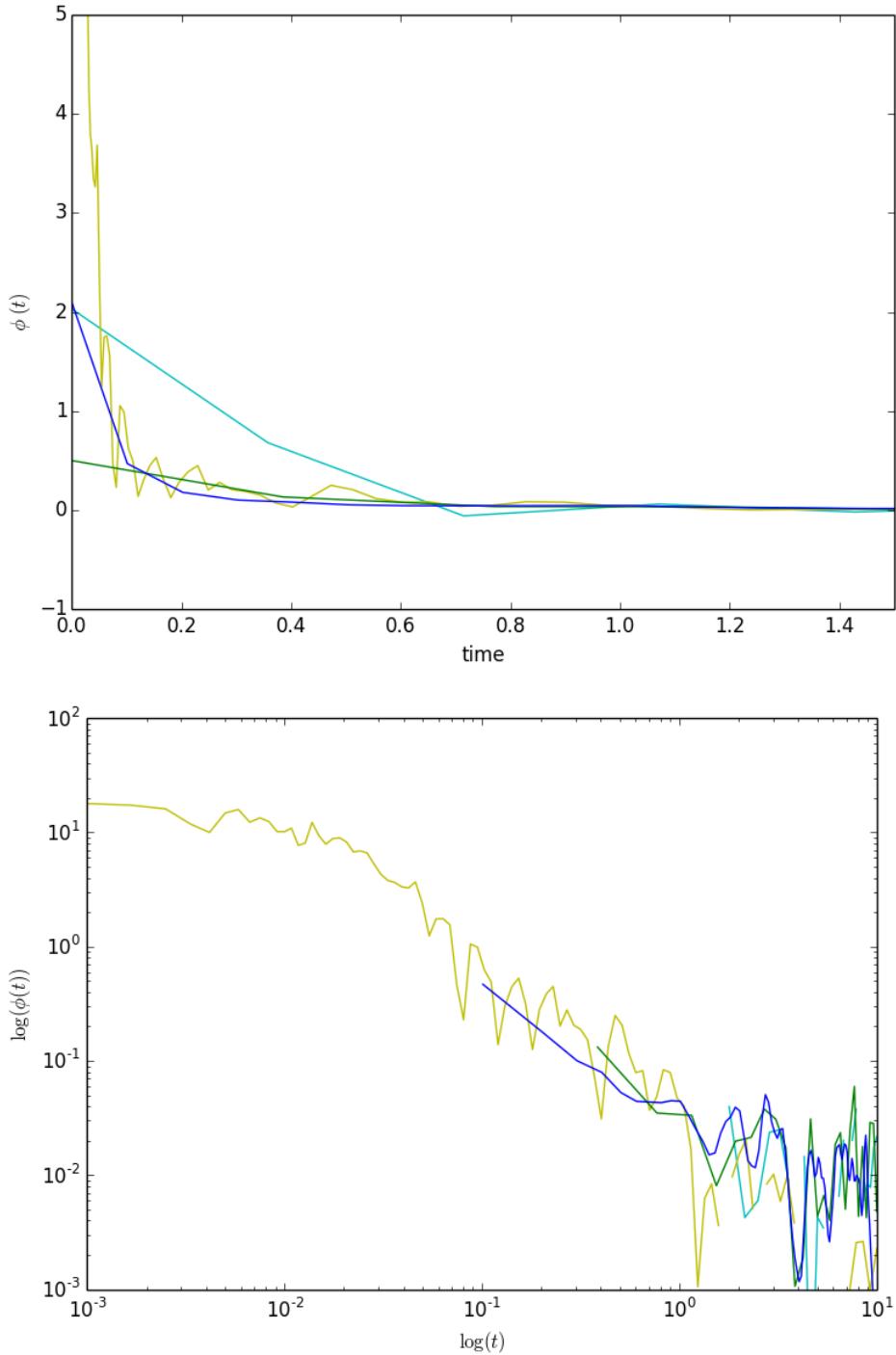


Figure 5.19: The linear-linear (upper plot) and log-log (bottom plot) plots of the estimated kernels fitted on the data sample of size 2600 representing the price changes of a contract LCOc1 on 26-01-2015 from 18:00 to 19:00 using the WH method on the uniform grid (cyan line), WH method on the semi-uniform grid (yellow line), INAR method (green line) and EM Algorithm with Kernel Density Estimation (blue line).

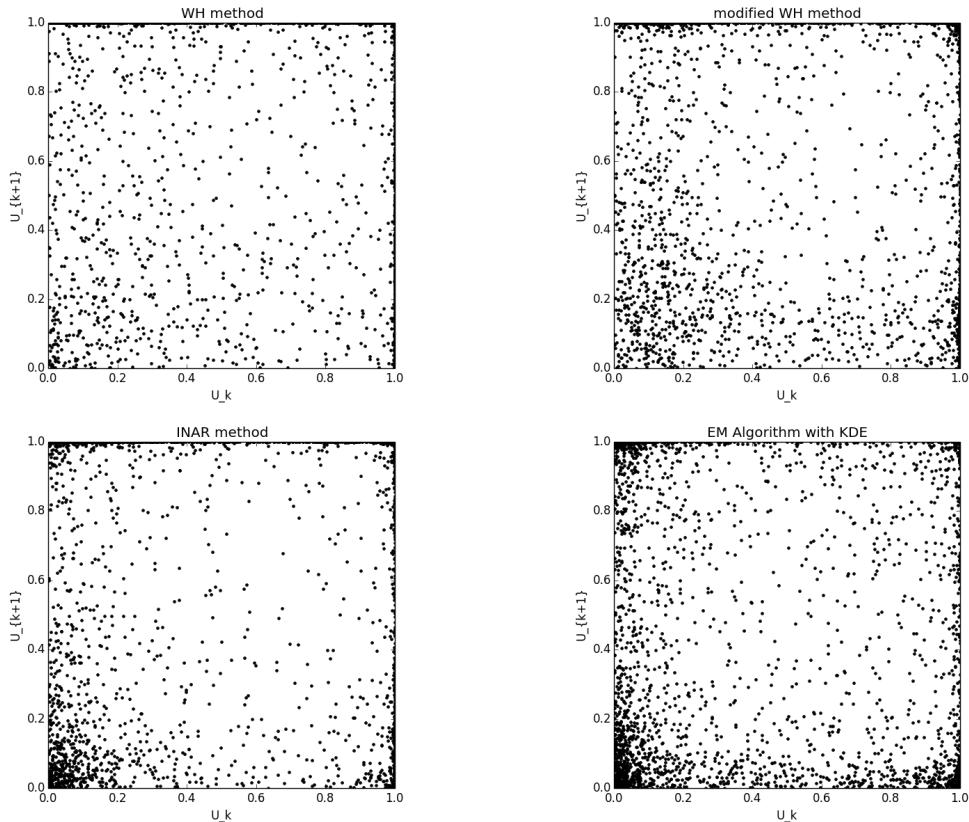


Figure 5.20: The residual plots of the models fitted on the data sample of size 2600 representing the price changes of a contract LCOc1 on 26-01-2015 from 18:00 to 19:00 using the WH method on the uniform grid (top left), WH method on the semi-uniform grid (top right), INAR method (bottom left) and EM Algorithm with Kernel Density Estimation (bottom right).

LCOc1	\hat{n}_{WH}	\hat{n}_{WH-mod}	\hat{n}_{INAR}	\hat{n}_{EMKDE}
18-04-2011 17:00-18:00	0.58	0.5	0.34	0.68
22-03-2012 15:00-16:00	0.74	0.66	0.6	0.74
05-07-2013 16:00-17:00	0.68	0.58	0.51	0.69
30-10-2014 15:00-16:00	0.67	0.56	0.44	0.74
26-01-2015 18:00-19:00	0.68	0.59	0.39	0.67

Table 5.4: Branching ratio estimated on the Brent Crude futures contract data on a given dates using the WH method on the uniform grid ("WH"), modified WH method on the semi-uniform grid ("WH-mod"), INAR method ("INAR") and EM Algorithm with Kernel Density Estimation ("EMKDE").

LCOc1	$\hat{\mu}_{WH}$	$\hat{\mu}_{WH-mod}$	$\hat{\mu}_{INAR}$	$\hat{\mu}_{EMKDE}$
18-04-2011 17:00-18:00	0.25	0.29	0.4	0.19
22-03-2012 15:00-16:00	0.34	0.44	0.52	0.35
05-07-2013 16:00-17:00	0.28	0.37	0.43	0.28
30-10-2014 15:00-16:00	0.38	0.51	0.66	0.3
26-01-2015 18:00-19:00	0.23	0.3	0.44	0.23

Table 5.5: Background intensity estimated on the Brent Crude futures contract data on a given dates using the WH method on the uniform grid ("WH"), modified WH method on the semi-uniform grid ("WH-mod"), INAR method ("INAR") and EM Algorithm with Kernel Density Estimation ("EMKDE").

LCOc1	$\hat{\beta}_{WH}$	$\hat{\beta}_{WH-mod}$	$\hat{\beta}_{INAR}$	$\hat{\beta}_{EMKDE}$
18-04-2011 17:00-18:00	0.34	0.9	-0.21	0.77
22-03-2012 15:00-16:00	0.95	1.33	0.52	2.07
05-07-2013 16:00-17:00	0.75	1.19	1.42	2.28
30-10-2014 15:00-16:00	0.43	1.81	0.53	2.16
26-01-2015 18:00-19:00	0.45	1.22	0.2	0.97

Table 5.6: Estimates of the tail's exponent obtained on the Brent Crude futures contract data on a given dates using the WH method on the uniform grid ("WH"), modified WH method on the semi-uniform grid ("WH-mod"), INAR method ("INAR") and EM Algorithm with Kernel Density Estimation ("EMKDE").

Chapter 6

Summary

The goal of this work was to compare existing non-parametric methods to calibrate the Hawkes model. First the analysis of estimation was conducted on synthetic data generated from the Hawkes model with the most commonly used exponential and power law memory kernels, but also with less popular step and cut-off kernels. The parameters of the model have been chosen in a way, which would allow to assess the impact of different factors like the value of the branching ratio and separation of point clusters on the estimation outcome. After the analysis of the synthetic data was accomplished I performed a study on high-frequency data of price changes of E-mini S&P 500 and Brent Crude futures contracts.

First I will summarize the results on synthetic data. The EM Algorithm using Logsplines outperformed the other methods in most of the cases. The WH method was working well for separated data clusters and low values of branching ratio. When the clusters were overlapping or had a high branching ratio, it was observed that the EM Algorithm with Kernel Density Estimation was the second best performing. The INAR method seemed to constantly provide poorest results. In the case of slowly decaying kernels, both EM Algorithm methods kept performing well. The original WH method on a uniform grid was not very successful, but its modification on a semi-uniform grid could follow the theoretical curve well. Unfortunately the provided solution was quite noisy. For less regular step and cut-off kernels, once again the EM Algorithm with Logsplines provided the best results with other methods being closely behind. At this point it shall be mentioned that the sample sizes that I considered were of relatively small size. This gave an edge to the EM Algorithm methods, which used more sophisticated kernel estimation methods. It seems to me that the WH and INAR methods would have more potential to improve if we were to increase the sample size.

Analysis on real financial data resulted in less promising results with respect to the EM Algorithm with Logsplines. Because of the highly heavy-tail nature of the data, convergence problems of the Logsplines method occurred, which I was unable to overcome. Among the remaining procedures the modified WH method due to its flexible grid could provide much higher precision in the head of the kernel than other methods. The residual analysis for E-mini S&P 500 futures showed bad fit of the models to the data for all the methods. On the other hand the results on Brent Crude contract data seem very promising providing uniform residuals in most of the cases.

I believe there are few ways to continue my work. The comparison of the methods on step and cut-off kernels provided couple of very good fits of the kernels, especially in the

case of the EM Algorithm with Logsplines. I think the more exhaustive study is worth being performed. The analysis on data generated from the slowly decaying kernels showed defects of the INAR method and the EM Algorithm with Kernel Density Estimation as not being flexible enough to approximate well the head and the tail of the kernel at the same time. The reason for that was the fact that they use a uniform grid in the case of the INAR method and a fixed size bandwidth in the EM Algorithm. I believe an interesting extension of those methods would be to apply the INAR method on semi-uniform grid (3.2.10)(or some version of it), similarly as it has been done for the WH method and incorporate Adaptive Kernel Density Estimation into the EM Algorithm. I suppose the flexibility obtained in this way would allow much better performance of those methods on heavy-tailed data. Finally, the study performed on Brent Crude futures data show very nice fit of the models obtained from non-parametric calibration to the data. I would strongly recommend performing more comprehensive analysis in order to further investigate this matter.

Bibliography

- Akaike, H. (1974). A new look at the statistical model identification. *Automatic Control, IEEE Transactions on* 19(6), 716–723.
- Bachelier, L. (1900). *Théorie de la spéculation*. Gauthier-Villars.
- Bacry, E., K. Dayri, and J.-F. Muzy (2012). Non-parametric kernel estimation for symmetric hawkes processes. application to high frequency financial data. *The European Physical Journal B* 85(5), 1–12.
- Bacry, E., T. Jaisson, and J.-F. Muzy (2014). Estimation of slowly decreasing hawkes kernels: Application to high frequency order book modelling. *arXiv preprint arXiv:1412.7096*.
- Bacry, E. and J.-F. Muzy (2014a). Hawkes model for price and trades high-frequency dynamics. *Quantitative Finance* 14(7), 1147–1166.
- Bacry, E. and J.-F. Muzy (2014b). Second order statistics characterization of hawkes processes and non-parametric estimation. *arXiv preprint arXiv:1401.0903*.
- Crane, R. and D. Sornette (2008). Robust dynamic classes revealed by measuring the response function of a social system. *Proceedings of the National Academy of Sciences* 105(41), 15649–15653.
- Daley, D. and D. Vere-Jones (2003). An introduction to the theory of point processes, volume i: Elementary theory and methods of probability and its applications.
- Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, 1–38.
- Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work*. *The journal of Finance* 25(2), 383–417.
- Filimonov, V. and D. Sornette (2012). Quantifying reflexivity in financial markets: Toward a prediction of flash crashes. *Physical Review E* 85(5), 056108.
- Filimonov, V. and D. Sornette (2015). Apparent criticality and calibration issues in the hawkes self-excited point process model: application to high-frequency financial data. *Quantitative Finance* (ahead-of-print), 1–22.
- Hardiman, S. J., N. Bercot, and J.-P. Bouchaud (2013). Critical reflexivity in financial markets: a hawkes process analysis. *The European Physical Journal B* 86(10), 1–9.
- Hawkes, A. G. (1971). Spectra of some self-exciting and mutually exciting point processes. *Biometrika* 58(1), 83–90.

- Helmstetter, A. and D. Sornette (2003). Predictability in the epidemic-type aftershock sequence model of interacting triggered seismicity. *Journal of Geophysical Research: Solid Earth (1978–2012)* 108(B10).
- Hull, J. C. (2006). *Options, futures, and other derivatives*. Pearson Education India.
- Kagan, Y. Y. and L. Knopoff (1981). Stochastic synthesis of earthquake catalogs. *Journal of Geophysical Research: Solid Earth (1978–2012)* 86(B4), 2853–2862.
- Kirchner, M. (2014). An Estimation Procedure for the Hawkes Process. 00(00), 1–24.
- Kooperberg, C. and C. J. Stone (1991). A study of logspline density estimation. *Computational Statistics & Data Analysis* 12(3), 327–347.
- Lewis, E. and G. Mohler (2011). A nonparametric em algorithm for multiscale hawkes processes. *preprint*, 1–16.
- Lewis, P. A. and G. S. Shedler (1978). Simulation of nonhomogeneous poisson processes by thinning. Technical report, DTIC Document.
- Marsan, D. and O. Lengliné (2008). Extending earthquakes’ reach through cascading. *Science* 319(5866), 1076–1079.
- Mohler, G. O., M. B. Short, P. J. Brantingham, F. P. Schoenberg, and G. E. Tita (2012). Self-exciting point process modeling of crime. *Journal of the American Statistical Association*.
- Møller, J. and J. G. Rasmussen (2005). Perfect simulation of hawkes processes. *Advances in applied probability*, 629–646.
- Møller, J. and J. G. Rasmussen (2006). Approximate simulation of hawkes processes. *Methodology and Computing in Applied Probability* 8(1), 53–64.
- Ogata, Y. (1981). On lewis’ simulation method for point processes. *Information Theory, IEEE Transactions on* 27(1), 23–31.
- Ogata, Y. (1988). Statistical models for earthquake occurrences and residual analysis for point processes. *Journal of the American Statistical Association* 83(401), 9–27.
- Ogata, Y. (1998). Space-time point-process models for earthquake occurrences. *Annals of the Institute of Statistical Mathematics* 50(2), 379–402.
- Papangelou, F. (1972). Integrability of expected increments of point processes and a related random change of scale. *Transactions of the American Mathematical Society* 165, 483–506.
- Rambaldi, M., P. Pennesi, and F. Lillo (2014). Modeling fx market activity around macroeconomic news: a hawkes process approach. *arXiv preprint arXiv:1405.6047*.
- Rustler, S. and V. Filimonov (2014). Shocks and self-excitation in twitter.
- Samuelson, P. A. (1965). Proof that properly anticipated prices fluctuate randomly. *Industrial management review* 6(2), 41–49.
- Silverman, B. W. (1986). *Density estimation for statistics and data analysis*, Volume 26. CRC press.
- Soros, G. (2003). *The alchemy of finance*. John Wiley & Sons.

- Utsu, T. and Y. Ogata (1995). The centenary of the omori formula for a decay law of aftershock activity. *Journal of Physics of the Earth* 43(1), 1–33.
- Veen, A. and F. P. Schoenberg (2008). Estimation of space–time branching process models in seismology using an em-type algorithm. *Journal of the American Statistical Association* 103(482), 614–624.
- Wheatley, S., V. Filimonov, and D. Sornette (2014). Estimation of the hawkes process with renewal immigration using the em algorithm. *Swiss Finance Institute Research Paper* (14-53).

Declaration of Originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor .

Title of work (in block letters):

NON-PARAMETRIC METHODS FOR ESTIMATION
OF HAWKES PROCESS FOR HIGH-FREQUENCY
FINANCIAL DATA

Authored by (in block letters):

For papers written by groups the names of all authors are required.

Name(s):

MORZYWOŁEK

First name(s):

PAWEŁ

With my signature I confirm that

- I have committed none of the forms of plagiarism described in the **Citation etiquette** information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work .
- I am aware that the work may be screened electronically for plagiarism.
- I have understood and followed the guidelines in the document *Scientific Works in Mathematics*.

Place, date:

ZÜRICH, 30.07.2015

Signature(s):

Paweł Morzywołek

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.