# Semiparametric Bayesian Approach using Weighted Dirichlet Process Mixture For Finance Statistical Models

Peng Sun

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

Doctor of Philosophy
in
Statistics

Inyoung Kim, Chair
Pang Du
Feng Guo
Hongxiao Zhu

February 12, 2016
Blacksburg, Virginia

KEYWORDS: Additive Model; Bayes factor; Cubic Splines; Dual-Semiparametric Regression; Generalized Pólya urn; Geometric ergodicity; Gibbs sampling; Metropolis-Hastings; Nonparametric Bayesian Model; Ordinal data; Parameterization; Semiparametric Regression; Stochastic volatility; Stock return; Weighted Dirichlet process

# Semiparametric Bayesian Approach using Weighted Dirichlet Process Mixture For Finance Statistical Models

Peng Sun

(ABSTRACT)

Dirichlet process mixture (DPM) has been widely used as flexible prior in nonparametric Bayesian literature, and Weighted Dirichlet process mixture (WDPM) can be viewed as extension of DPM which relaxes model distribution assumptions. Meanwhile, WDPM requires to set weight functions and can cause extra computation burden. In this dissertation, we develop more efficient and flexible WDPM approaches under three research topics. The first one is semiparametric cubic spline regression where we adopt a nonparametric prior for error terms in order to automatically handle heterogeneity of measurement errors or unknown mixture distribution, the second one is to provide an innovative way to construct weight function and illustrate some decent properties and computation efficiency of this weight under semiparametric stochastic volatility (SV) model, and the last one is to develop WDPM approach for Generalized AutoRegressive Conditional Heteroskedasticity (GARCH) model (as an alternative approach for SV model) and propose a new model evaluation approach for GARCH which produces easier-to-interpret result compared to the canonical marginal likelihood approach.

In the first topic, the response variable is modeled as the sum of three parts. One part is a linear function of covariates that enter the model parametrically. The second part is an additive nonparametric model. The covariates whose relationships to response variable are unclear will be included in the model nonparametrically using Lancaster and $\check{S}$alkauskas bases. The third part is error terms whose means and variance are assumed to follow nonparametric priors. Therefore we denote our model as dual-semiparametric regression because we include nonparametric idea for both modeling mean part and error terms. Instead of assuming all of the error terms follow the same prior in DPM, our WDPM provides multiple candidate priors for each observation to select with certain probability. Such probability (or weight) is modeled by relevant predictive covariates using Gaussian kernel. We propose several different WDPMs using different weights which depend on distance in covariates. We provide the efficient Markov chain Monte Carlo (MCMC) algorithms and also compare our WDPMs to parametric model and DPM model in terms of Bayes factor using simulation and empirical study.

In the second topic, we propose an innovative way to construct weight function for WDPM and apply it to SV model. SV model is adopted in time series data where the constant variance assumption is violated. One essential issue is to specify distribution of conditional return. We assume WDPM prior for conditional return and propose a new way to model the weights. Our approach has several advantages including computational efficiency compared to the weight constructed using Gaussian kernel. We list six properties of this proposed weight function and also provide the proof of them. Because of the additional Metropolis-Hastings steps introduced by WDPM prior, we find the conditions which can ensure the uniform geometric ergodicity of transition kernel in our MCMC. Due to the existence of zero values in asset price data, our SV model is semiparametric since we employ WDPM prior for non-zero values and parametric prior for zero values.

On the third project, we develop WDPM approach for GARCH type model and compare different types of weight functions including the innovative method proposed in the second topic. GARCH model can be viewed as an alternative way of SV for analyzing daily stock prices data where constant variance assumption does not hold. While the response variable of our SV models is transformed log return (based on log-square transformation), GARCH directly models the log return itself. This means that, theoretically speaking, we are able to predict stock returns using GARCH models while this is not feasible if we use SV model. Because SV models ignore the sign of log returns and provides predictive densities for squared log return only. Motivated by this property, we propose a new model evaluation approach called back testing return (BTR) particularly for GARCH. This BTR approach produces model evaluation results which are easier to interpret than marginal likelihood and it is straightforward to draw conclusion about model profitability by applying this approach. Since BTR approach is only applicable to GARCH, we also illustrate how to properly calculate marginal likelihood to make comparison between GARCH and SV. Based on our MCMC algorithms and model evaluation approaches, we have conducted large number of model fittings to compare models in both simulation and empirical study.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Outline of this Dissertation

Many researches have been fascinated by the fact that nonparametric priors enable people to capture arbitrary shape of distribution that data may display. Starting from Ferguson (1973), numerous studies have been conducted based on Dirichlet Process Mixture (DPM). Motivated by the idea and limitation of DPM, we further develop the nonparametric Bayesian approach under Weighted DPM (WDPM). Therefore in this dissertation, the introduction of WDPM and three research problems are illustrated.

In the first problem we propose an efficient and flexible Bayesian approach in semiparametric regression models which have both parametric and nonparametric functions of covariates. Our Bayesian approach is developed by assuming that the distributions of the response variables are unknown, therefore we model them using WDPM. Our approach is especially useful to real applications that have heterogeneity of measurement errors or come from mixture of distributions. In our semiparametric regression, some of the covariates are formulated as parametric linear function and other covariates are modeled as unknown function forms which are estimated using cubic smoothing splines. We propose several different WDPMs using different weights which depend on distance between covariates. We derive the marginal likelihood of them and provide the computation of marginal likelihood for WDPM. Efficient MCMC algorithms are provided. We compare our WDPMs with parametric error model and DPM error model in terms of Bayes factor using simulation study, suggesting better performance of our approach. The advantage of our approach is also demonstrated using credit rating data.

In the second problem, we propose the semiparametric Bayesian approach under Stochastic Volatility (SV) model to study the stock daily return. In SV models, the important part is to specify the distribution of the error term which is the difference between the target variable and log-volatility. However, when it comes to the data of daily returns of individual stock, the distribution should be discreetly chosen because a non-ignorable proportion of zero returns can make the data deviate a lot from the specified distribution if it is not properly chosen. The semiparametric method, which is introduced in Delatola and Griffin (2011), use a normal prior for zero return case and a DPM prior for the non-zero return case. We find that under this semiparametric SV model, the canonical form of weight, which is specified in Dunson et al (2007), requires heavy computation burden and does not display convergence of parameters in MCMC. Therefore, we develop a new way to construct the weights which can greatly reduce the computational burden in MCMC and contains a good property, that a nearer candidate is always more likely to be chosen than a farther one. The empirical results suggest that our weighted approach produces better marginal likelihoods.

In the third problem we develop a flexible nonparametric Bayesian approach to analyze Generalized AutoRegressive Conditional Heteroskedasticity (GARCH) model. Our model flexibility is obtained from relaxing the assumption that prior distribution of error term parameters is the same for every observation. Adopting WDPM, we can provide multiple candidate priors and let observations to favor different candidates. We model such difference by carefully chosen explanatory covariates and explore different ways of constructing weights.

Since GARCH models the log return, it enables us to predict stock price. We develop an empirical way of model evaluation by taking the complexity of real stock trading into consideration and refer it to the back testing return (BTR) approach. This BTR approach simulates the real process of stock price prediction and help us to provide decision making. Compared to marginal likelihood, our BTR approach provides model evaluation result which is easier to interpret. We have conducted large number of model comparisons among parametric and nonparametric Bayesian approaches of SV model and GARCH using marginal likelihood and using our BTR approach. It is found that WDPM provides us great number of choices for selecting better model in terms of both fitting accuracy and profitability.

Hence, this dissertation focuses on these three research problems of WDPM. The outline of the dissertation is specified as:

- In Chapter 2, we apply WDPM to semiparametric regression. In this Chapter, the following major issues will be discussed:
  - We will specify different weight functions in the Introduction of WDPM.
  - We will introduce the cubic spline regression model upon which WDPM is performed.
  - We will introduce the very important posterior distribution which is applicable to arbitrary form of WDPM in MCMC sampling.
  - The results of simulation and empirical study will be discussed.
- In Chapter 3, we apply WDPM to SV model by proposing a new weight. In this Chapter, the following major issues will be discussed:
  - We will introduce SV model.

  - We will propose a new way to model weights which displays several good properties.
  - We will introduce the framework of sampling and provide some regulations about the priors to ensure the geometric ergodicity in MCMC sampling.
  - The results of simulation and empirical study will be discussed.

- In Chapter 4, we develop WDPM model and propose a new model evaluation approach for GARCH. In this Chapter, the following major issues will be discussed:
  - We will introduce GARCH model.
  - We will explain how to apply WDPM to GARCH model.
  - We will propose the new approach (BTR) of model evaluation for GARCH which produces easy-to-interpret result about model profitability.
  - We will introduce several useful facts in our computation approach for WDPM GARCH in MCMC sampling.
  - We will illustrate how to properly calculate marginal likelihood to make SV and GARCH comparable between each other.
  - The results of simulation and empirical study will be discussed.

# Chapter 2

# Dual-Semiparametric Regression using Weighted Dirichlet Process Mixture

## 2.1 Introduction

Semiparametric regression has been widely used in many fields including economics, finance, biostatistics, where some covariates have known relationship with response while the others do not. It mixes the parametric part and the nonparametric part together in the regression. The nonparametric part can be estimated using cubic smoothing splines (Durrleman and Simon, 1989; Green and Silverman, 1994; Pagan and Ullah, 1999; Marsh and Cormier, 2002; Li and Racine, 2006; Chib and Greenburg, 2010). Although semiparametric regression has a flexibility to model both parametric and nonparametric parts, it also needs strong assumptions about the distribution of the unobserved error or the distribution of the underlying latent variable. The error distribution is usually assumed to be parametric, especially when the outcome is not continuous. In ordinal models for categorical outcomes, the model is almost always specified with logit or probit links. These assumptions are often not satisfied in real applications that have heterogeneity of measurement errors or come from the mixture of heterogeneous distribution. As a result the model may easily turn out to be misspecified and thus influences the statistical inference. Therefore, we adopt nonparametric Bayesian approach to allow the semiparametric regression to be more flexible. Our nonparametric Bayesian approach can be applicable to both continuous and ordinal response.

First let us consider that we have $p+q$ covariates and continuous response variable $y$. The first $p$ covariates can be parametrically modeled with unknown parameters $\boldsymbol{\beta}_p$ and the other $q$ covariates $w_1, \ldots, w_q$ should be nonparametrically modeled with $g_j(\cdot)$, $j = 1, \ldots, q$. Our semiparametric model can be written as:

$$y_i = \mathbf{x}_i^{'}\boldsymbol{\beta}_p + g_1(w_{i1}) + g_2(w_{i2}) + \cdots + g_q(w_{iq}) + \varepsilon_i \quad i = 1, \ldots, n, \tag{2.1}$$

where $\mathbf{x}_i^{'}\boldsymbol{\beta}_p$ is the parametric function, $g_j(\cdot)$ $(j \leq q)$ are unknown functions which are estimated using the cubic smoothing splines by adopting the basis illustrated in Lancaster et al (1986) and Wood (2006), and $\varepsilon$ is the error term whose shape of distribution needs to be captured.

In parametric Bayesian approach, it is typical to assume that the distribution of $\varepsilon_i$ has some known form $f_e(\varepsilon_i|\theta_i)$, where $\theta_i$ denotes the parameters of the distribution of error term. The prior of $\theta_i$ is also in some known form $G_0$. However, it is relatively challenging to derive convincing evidence about what the distribution of error is and prior distribution really follows. The Dirichlet process, that was formally introduced by Ferguson (1973), enables us to construct the prior of $\theta_i$ in a nonparametric way. If we assume the prior distribution of $\theta_i$, say G, is sampled from a Dirichlet process, this Dirichlet Process Mixture

Model (DPM) allows more flexibility than assuming G belongs to some known family of distributions. Therefore, many literatures have adopted DPM in their study (Chib and Greenburg, 2010; Hannah et al, 2011; Jensen and Maheu, 2013). It is found that if the error terms follow normal distribution and $\theta_i$ is the parameters in the normal distribution for the $i^{th}$ observation, a prior G of $\theta_i$ that is sampled from Dirichlet process is able to approximate any unknown distribution (Ferguson, 1983). However in parametric Bayesian approach, such flexibility is often unachievable.

Although DPM has such flexibility, it still does not take account the covariates' information into the prior of $\theta_i$. DPM assumes that all the $\theta_i'$s follow the same prior distribution G. Therefore DPM cannot be useful when the error distributions are the mixture of heterogeneous distributions. In order to overcome this limitation, we can use weighted Dirichlet process mixture (WDPM). The idea of WDPM was enlightened by Zellner (1986) and applied in Dunson et al (2007). The concept of WDPM comes from the idea that one can add the information provided by covariates into the construction of prior distributions. Such concept will relax the constraint that all the observations share the same prior for the error term parameter. Instead of a single prior, there will be multiple candidate priors. The observations that have similar values of predictors (covariates) are more likely to share the same prior which is one of the candidate priors available. Therefore, we can see that WDPM allows higher degree of heterogeneity of observations compared to DPM or parametric Bayesian model. As a summary, we have that in parametric Bayesian, we have only one prior for all the $\theta_i'$s and such prior must belongs to some known distribution family. In DPM, there is only one prior as well although it is not necessarily some known distribution. However, when the observations do even not share same prior, DPM is not appropriate. WDPM can be an efficient alternative approach of DPM where model based on single prior assumption fails to produce adequate amount of accuracy.

In this Chapter, we incorporate WDPM to semiparametric regressions and propose several different WDPM models using different weight functions. Our weighted models are called dual-semiparametric models because we adopt semiparametric regression for mean part and nonparametric Bayesian approaches for error terms. We derive the marginal likelihood of them for statistical inference. Efficient MCMC algorithms are provided. To the best of our knowledge, our approach is the first one incorporating WDPM to semiparametric regression, proposing efficient weights, and providing the marginal likelihood derivation. Our weight functions in WDPM are compared with the weight function proposed by Dunson et al (2007). We further compare our WDPM models with parametric and DPM models in terms of Bayes factor using both simulation study and real data and suggesting some outperformance of our approach.

The rest of this Chapter is organized as follows. In Section 2.2, we briefly review the basic idea of WDPM and propose several weight functions for WDPM. In Section 2.3, we introduce the potential reason why WDPM can produce better marginal likelihood based on P$\acute{o}$lya urn for WDPM. In Section 2.4, we explain how to incorporate WDPM to the semiparametric regression. In Section 2.5, the posterior computation is illustrated. In Section 2.6, the marginal likelihood computation is explained so that it can be used for statistical inference. In Section 2.7, we conduct simulation study to understand the performance of our approach. In Section 2.8, we apply our method to credit rating data illustrated in Verbeek (2008). Finally, Section 2.9 provide our concluding remarks.

## 2.2 Weighted Dirichlet Process Mixture

WDPM can be treated as a generalized DPM method that include the covariates' information into the prior of parameter $\boldsymbol{\theta} = (\theta_1, \cdots, \theta_n)$. Instead of having only one unique G for all the $\theta_i$'s, we provide multiple candidate priors $(\mathrm{G}_1^c, \cdots, \mathrm{G}_c^Q)$ for each observation to select. We need to specify a location in the covariate space $X_S$, say $\boldsymbol{x}_q^c$, for each candidate prior $\mathrm{G}_q^c$, and all the $\mathrm{G}_q^c$'s are independently sampled from the same $\mathrm{DP}(\zeta, G_0)$, where $\zeta$ is the constant precision parameter to the base measure $G_0$. Therefore, for an given observation with covariate vector $\boldsymbol{x}$, the probability of this observation selecting $\mathrm{G}_q^c$ as its true prior can be modeled by some function $\pi(\cdot,\cdot)$ of $\boldsymbol{x}$ and $\boldsymbol{x}_q^c$, that is, $\Pr(\text{prior} = \mathrm{G}_q^c) = \pi(\boldsymbol{x}, \boldsymbol{x}_q^c)$. Such function is usually built on the idea that if the covariates' values of two observations are exactly the same or very close to each other, the probability that the two error term's parameters follow the same prior should be higher than that of the case where the two observations are very far away from each other in terms of covariate values. We know that having the same prior of the parameter means having the same marginal distribution on error term. Therefore, the weight function is constructed in the way that favors the assumption that error terms sharing similar covariate informations tend to share similar distribution. We can consider the following the weight function:

$$\pi(\boldsymbol{x}, \boldsymbol{x}_q^c) = \frac{\gamma_q e^{-\psi \left\| \boldsymbol{x}_i - \boldsymbol{x}_q^c \right\|^2}}{\sum_{l=1}^{Q} \gamma_l e^{-\psi \left\| \boldsymbol{x}_i - \boldsymbol{x}_l^c \right\|^2}} \qquad q = 1, \cdots, Q, \quad \forall \boldsymbol{x} \in X_S \tag{2.2}$$

where both $\gamma_l$'s and $\psi$ are positive parameters of the weight function. We can see that the weight is decreasing function of the distance in covariate vector between the given observation and the location of the $q^{th}$ candidate prior, so this follows the basic idea mentioned above in the construction of the weight function.

Since both the number of candidate priors and locations of these candidates affect the weights $\left\{\pi_{iq} \triangleq \pi(\boldsymbol{x}_i, \boldsymbol{x}_q^c)\right\}_{n\times Q}$ that are finally calculated based on the $n$ observations in the given data, how to choose proper number and locations of the candidate priors is worthy of consideration. If the number and locations are poorly chosen, they can produce misleading weights although the weight function is in a legitimate form such as (2.2). For example, consider that most of the observations' covariate vectors are very close to each other and form a cluster while several outliers are far away from this cluster but very close among themselves to form another cluster. In this case, the weight function will lose power to reflect the covariate information of this data if all the candidate locations are chosen to be around the middle point of the two clusters. Dunson et al (2007) propose to choose $Q = n$ and use the $n$ observations' covariate vectors as the locations of the candidate priors. We apply this approach in our empirical study and find that the model performance will not change significantly if we randomly sample the candidate locations rather than selecting the locations given by the observations. We also explore how the number of candidate priors affects the model performance, and find that a small $Q$ leads to poor result while the model performance becomes insensitive as $Q$ approaches to $n$. In another word, neither a large number of $Q$ or randomization of candidate locations makes significant model improvement. Therefore, we turn to different specification of weights and priors of hyper parameters in the weight function. The following WDPM models have been studied:

- Dunson's WDPM:

$$\pi(\boldsymbol{x}, \boldsymbol{x}_q^c) = \frac{\gamma_q e^{-\psi \left\|\boldsymbol{x}-\boldsymbol{x}_q^c\right\|^2}}{\sum_{l=1}^n \gamma_l e^{-\psi \left\|\boldsymbol{x}-\boldsymbol{x}_l^c\right\|^2}} \qquad q = 1, \cdots, n, \quad \forall x \in X_S \tag{2.3}$$

  This is denoted as WDPMD;

- Efficient WDPM: In (2.3) $\gamma_l$ and $\psi$ are not identifiable under one of the following situtations: (i) $\gamma_l \to 0$; (ii) $1/\psi \to 0$ and $\gamma_l \sim O(\psi^c)$ for any positive $c$; or (iii) $\psi \to 0$. To avoid this identifiable situation, we propose the following weight function

$$\pi(\boldsymbol{x}, \boldsymbol{x}_q^c) = \frac{e^{-\psi \left\|\boldsymbol{x}-\boldsymbol{x}_q^c\right\|^2}}{\sum_{l=1}^n e^{-\psi \left\|\boldsymbol{x}-\boldsymbol{x}_l^c\right\|^2}}, \tag{2.4}$$

The one using weight function (2.4) will be called EWDPM (short as efficient WDPM) hereafter;

- WDPMG: We can also consider the weight that depends on variable $\psi_j$'s rather than a fixed $\psi$ for all the observation:

  $$\pi(\boldsymbol{x}, \boldsymbol{x}_q^c) = \frac{e^{-\psi_q \left\|\boldsymbol{x}-\boldsymbol{x}_q^c\right\|^2}}{\sum_{l=1}^{n} e^{-\psi_l \left\|\boldsymbol{x}-\boldsymbol{x}_l^c\right\|^2}}; \tag{2.5}$$

  The hyper parameters enter the weight function linearly in (2.3) and in contrast they enter the function exponentially in (2.5). The types of hyper priors for $\psi_j$'s in (2.5) is gamma distribution. We refer this WDPM to WDPMG;

- WDPME: It is the same as (2.5) except that hyper prior is following exponential distribution; We refer this WDPM to WDPME;

- WDPMH: It is the same as as (2.5) except for the hyper prior is following half cauchy distribution (Carvalho and Polson, 2010) which is also called as horse shoes prior distribution. We refer this WDPM to WDPMH.

The weighted DPM is a generalization of DPM in the sense that there is only one candidate prior in DPM while there are multiple candidates in weighted method. We have to allocate the $n$ observations to the $Q$ candidate priors based on $\{\pi_{iq}\}_{n\times Q}$ in weighted models. But in DPM, we have that $\pi_{i1} = 1$ and $\pi_{iq} = 0$ for $i = 1, \cdots, n$ and $q > 1$ and ignore the information provided by the predictive covariates. We will compare the performances of these WDPM models with DPM model based on both simulated and empirical data.

## 2.3 P$\acute{o}$lya urn for WDPM

For MCMC sampling, we need to generalize the P$\acute{o}$lya urn result (Blackwell and MacQueen, 1973; MacEachern et al, 1998) to WDPM model. In DPM, the conditional prior $f_p(\cdot|\cdot)$ for parameter $\theta_i$ given the basis distribution of $G_0$ and all the other values in $\boldsymbol{\theta}$ is:

$$f_p(\theta_i|\boldsymbol{\theta}_{-i},\zeta) = \frac{\zeta}{\zeta+n-1}G_0 + \frac{1}{\zeta+n-1}\sum_{j\neq i}\delta_{\theta_j}, \tag{2.6}$$

where $\boldsymbol{\theta}_{-i}$ is vector of parameters excluding the $i^{th}$ element and $\delta_j$ is a zero point mass at the $j^{th}$ parameter.

Let $\boldsymbol{Z} = (Z_1, Z_2, \cdots, Z_n)$ be the latent vector which indicates the allocation of the $n$ observations to the $Q$ candidate priors. In DPM model, there is only one candidate prior and $\boldsymbol{Z} = \mathbf{1}_n$. In WDPM model, the number of possible outcomes of $\boldsymbol{Z}$ is $Q^n$. The conditional prior in WDPM model is:

$$f_p(\theta_i|\boldsymbol{Z},\boldsymbol{\theta}_{-i},\zeta) = \frac{\zeta}{\zeta+\sum_{j\neq i}\mathbf{1}(Z_j=Z_i)}G_0 + \frac{1}{\zeta+\sum_{j\neq i}\mathbf{1}(Z_j=Z_i)}\sum_{j\neq i}\mathbf{1}(Z_j=Z_i)\delta_{\theta_j}. \tag{2.7}$$

We can interpret the potential advantage that WDPM model contains in producing a better marginal likelihood by comparing (2.6) and (2.7). Suppose that $\theta_i$ represents the precision of the normal distribution that $\varepsilon_i$ follows, i.e. $\varepsilon_i|\theta_i \sim \mathrm{N}(0, \theta_i^{-1})$. We know that in DPM, $\theta_i|\boldsymbol{\theta}_{-i}$ either comes from $G_0$ or repeats one of the values in $\boldsymbol{\theta}_{-i}$. However in WDPM, $\theta_i|\boldsymbol{\theta}_{-i}$ will only repeat one of the values that are assigned to the same candidate prior with itself. Based on the weight function, we know that if two observations are close in covariate values, the chance that their error term parameters follow the same candidate prior will be higher compared to the case that they are far away from each other. Therefore, it is more likely that $\theta_i|\boldsymbol{\theta}_{-i}$ focuses only on $G_0$ and the values in $\boldsymbol{\theta}_{-i}$ whose observations are close to the

$i^{th}$ one in covariate values. This may be a more reasonable mechanism than simply taking all the values in $\boldsymbol{\theta}_{-i}$ into consideration, because we know that it is common assumption that the two observations with the same covariate values follow the normal distribution with same variance. Although (2.7) is just the prior of $\theta_i|\boldsymbol{\theta}_{-i}$ and we need to modify it to derive the posterior of $\theta_i|\boldsymbol{\theta}_{-i}$ in MCMC sampling, the idea of including covariate information and favoring close candidate may still lead to better model accuracy.

## 2.4 Weighted Dirichlet Process Mixture in Semiparametric Regression

In this section, we explain how WDPM can be applied to semiparametric regression model with continuous and ordinal response variables.

### 2.4.1 Semiparametric regression

Our semiparametric regression model with continuous response variable can then be written as (2.1), where $\epsilon_i$ is the error term whose distribution parameters are assumed to follow parametric or nonparametric prior in our study.

When the response is ordinal, $y_i$ takes one of the ordered values $\{0, \ldots, J-1\}$. We write the model using a latent variable $y^*$ and ordered category cut points, $c_1 < \ldots < c_{J-2}$, where $y_i = j$ if $c_{j-1} < y_i^* \leq c_j$. The cut points divide the support of the latent variable into a sequence of intervals. The response variable labels these intervals from 0 to $J-1$, where $J$ is the number of intervals. Using these setting, the model for ordinal case is the same as that of continuous case except that we replace $y_i$ by $y_i^*$ in (2.1).

It should be mentioned that the ordinal outcome model can be viewed as the extension of the binary outcome case that is introduced in Basu and Chib (2003). The cut points are often of great interest in the ordinal model because finding the accurate cut points is very crucial in producing a better marginal likelihood. Let $\boldsymbol{c} = (c_1, \cdots, c_{J-2})$ be the vector of the free cut points (for J-outcome case, we need J-1 cut points. We can always set $c_0$ to 0 and adjust the intercept of the regression. So there $J-2$ free cut points). Denoting $a_1 = \log(c_1)$ and $a_j = \log(c_j - c_{j-1})$ for $2 \leq j \leq J-2$, we assume $\boldsymbol{a} \sim \mathrm{N}(\boldsymbol{a}_{00}, A_{00})$ where the mean vector and variance matrix follow the setting specified by Chib and Greenburg (2010).

### 2.4.2 Modeling error terms

DPM can be applied to model the error terms in both continuous and ordinal outcome cases. For example, it can be assumed in the ordinal model that $\varepsilon_i|\lambda_i \sim \mathrm{N}(0, \lambda_i^{-1})$, $\lambda_i|\mathrm{G} \sim \mathrm{G}$ and $\mathrm{G} \sim \mathrm{DP}(\zeta, G_0)$. This is a DPM for ordinal outcome. If $G_0$ is directly used as the prior of $\lambda_i$ instead of $\mathrm{G} \sim \mathrm{DP}(\zeta, G_0)$, we can derive that the marginal distribution of the latent variable $y^*$ is a Student-t distribution if $G_0$ is a gamma distribution. Therefore, the Student-t model is a parametric Bayesian model and the models developed based on DPM are nonparametric Bayesian model. Extending DPM to WDPM, we have:

$$
\begin{aligned}
\varepsilon_i|\lambda_i &\sim \mathrm{N}(0, \lambda_i^{-1}); \\
\lambda_i|[(Z_i = q)|\boldsymbol{\theta_H}] &\sim \mathrm{G}_q^c; \\
\Pr(Z_i = q|\boldsymbol{\theta_H}) &= \pi(\boldsymbol{x_i}, \boldsymbol{x_q^c}); \\
\mathrm{G}_q^c &\sim \mathrm{DP}(\zeta, G_0); \\
G_0 &= \mathrm{Gamma}(\lambda_i|\frac{v}{2}, \frac{v}{2}).
\end{aligned}
$$

where $\boldsymbol{\theta_H}$ is $(\psi, \gamma_1, \cdots, \gamma_n)$ in WDPMD, $\psi$ in EWDPMD and $(\psi_1, \cdots, \psi_n)$ in WDPMG, WDPME and WDPMH, and $v$ is the hyper parameter that specifies $G_0$.

### 2.4.3 Modeling unknown function

The unknown functions can be modeled using cubic splines, $\Phi_{mj}(w)$ and $\Psi_{mj}(w)$, where

$$\Phi_{mj}(w) = \begin{cases} 0, \text{ if } w < \tau_{m-1,j}; \\ -(2/h_{mj}^3)(w - \tau_{m-1,j})^2(w - \tau_{mj} - 0.5h_{mj}) \text{ if } \tau_{m-1,j} \leq w < \tau_{mj}; \\ (2/h_{mj}^3)(w - \tau_{m+1,j})^2(w - \tau_{mj} + 0.5h_{m+1,j}) \text{ if } \tau_{m,j} \leq w < \tau_{m+1,j}; \\ 0, \text{ if } w \geq \tau_{m+1,j}, \end{cases}$$

$$\Psi_{mj}(w) = \begin{cases} 0, \text{ if } w < \tau_{m-1,j}; \\ (1/h_{mj}^2)(w - \tau_{m-1,j})^2(w - \tau_{mj}) \text{ if } \tau_{m-1,j} \leq w < \tau_{mj}; \\ (1/h_{m+1,j}^2)(w - \tau_{m+1,j})^2(w - \tau_{mj}) \text{ if } \tau_{m,j} \leq w < \tau_{m+1,j}; \\ 0, \text{ if } w \geq \tau_{m+1,j}, \end{cases}$$

$\boldsymbol{\tau}_j = (\tau_{1j}, \cdots, \tau_{M_j j})$ are the $M_j$ knot points for the $j^{th}$ unknown function $g_j(w)$, $\{\Phi_{mj}(w)\}_{m=1}^{M_j}$ and $\{\Psi_{mj}(w)\}_{m=1}^{M_j}$ are the two type of collections of cubic splines, and $h_{mj} = \tau_{mj} - \tau_{m-1,j}$ is the width between the $(m-1)$st and $m^{th}$ knots.

Then the unknown function $g_j(\cdot)$ for the $l^{th}$ observation is represented as:

$$g_j(w_{lj}) = \sum_{m=1}^{M_j} \{\Phi_{mj}(w_{lj}) f_{mj} + \Psi_{mj}(w_{lj}) s_{mj}\}. \tag{2.8}$$

Since $g_j(w)$ is natural cubic spline, we need its second derivative to be 0 at the lower and upper bounds. We also require $g_j(w)$ to be continuous at the knot points. These all place restrictions on $s_{mj}$. Using these restrictions, Lancaster et al (1986) show that the coefficients of one type of basis function ($\{\Psi_{mj}(w)\}_{m=1}^{M_j}$) can be represented by the coefficients of the other type of basis function. Denoting $(\Phi_{1j}(w_{lj}), \cdots, \Phi_{M_j j}(w_{lj}))$ by $(\Phi_j(w_{lj})^T)$, we can rewrite $g_j(w_{lj})$ as:

$$g_j(w_{lj}) = (\Phi_j(w_{lj})^T + \Psi_j(w_{lj}) A_j^{-1} C_j) \boldsymbol{f}_j \tag{2.9}$$

where

$$A_j = \begin{bmatrix} 2 & 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \eta_{2j} & 2 & \mu_{2j} & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & \eta_{3j} & 2 & \mu_{3j} & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \cdots & \ddots & \ddots & \ddots & \cdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & \eta_{M_j-1,j} & 2 & \mu_{M_j-1,j} \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 2 \end{bmatrix},$$

$$C_j = 3 \begin{bmatrix} -\frac{1}{h_{2j}} & \frac{1}{h_{2j}} & 0 & 0 & \cdots & 0 & 0 & 0 \\ -\frac{\eta_{2j}}{h_{2j}} & \frac{\eta_{2j}}{h_{2j}} - \frac{\mu_{2j}}{h_{3j}} & \frac{\mu_{2j}}{h_{3j}} & 0 & \cdots & 0 & 0 & 0 \\ 0 & -\frac{\eta_{3j}}{h_{3j}} & \frac{\eta_{3j}}{h_{3j}} - \frac{\mu_{3j}}{h_{4j}} & \frac{\mu_{3j}}{h_{4j}} & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & -\frac{\eta_{M_j-1,j}}{h_{M_j-1,j}} & \frac{\eta_{M_j-1,j}}{h_{M_j-1,j}} - \frac{\mu_{M_j-1,j}}{h_{M_j,j}} & \frac{\mu_{M_j-1,j}}{h_{M_j,j}} \\ 0 & 0 & 0 & 0 & \cdots & 0 & -\frac{1}{h_{M_j,j}} & \frac{1}{h_{M_j,j}} \end{bmatrix},$$

$\eta_{mj} = h_{mj}/(h_{mj} + h_{m+1,j})$, $\mu_{mj} = 1 - \eta_{mj}$ and $\boldsymbol{f}_j = (f_{1j}, \cdots, f_{M_j j})^T$.

Defining ${\boldsymbol{z}_{lj}}^T = \Phi_j(w_{lj})^T + \Psi_j(w_{lj})A_j^{-1}C_j$, we can finally transfer (2.1) into:

$$y = X_0\boldsymbol{\beta}_0 + Z_{np}\boldsymbol{f} + \varepsilon \tag{2.10}$$

where $Z_{np} = \left\{{\boldsymbol{z}_{lj}}^T\right\}_{n\times\sum_{j=1}^{q} M_j}$ and $\boldsymbol{f} = (\boldsymbol{f}_j^T, \cdots, \boldsymbol{f}_j^T)^T$. Although it looks same as a linear model once $(\boldsymbol{w}_1, \cdots, \boldsymbol{w}_q)$ that enter the model non-parametrically are transformed into $Z_{np}$ data matrix, there is still difference between the coefficients of $X_0$ and $Z_{np}$ in terms of how the priors are given. For the parametric part, we assume the prior of the coefficients is multivariate normal whose mean vector and variance matrix are specified. For the nonparametric part, on the other hand, we assume the prior of the coefficients is multivariate normal but this distribution is formulated in the following two ways; (1) the differences in slopes of the coefficients (i.e. $\Delta(\frac{f_m - f_{m-1}}{h_m})$) follow a normal distribution with zero mean and certain variance ($\sigma_d^2$) for the interior knots; the two slopes themselves on the boundary (i.e. $\frac{f_2 - f_1}{h_1}$ and $\frac{f_M - f_{M-1}}{h_M}$) follow another normal distribution with zero mean and a different variance ($\sigma_e^2$). Note that $\sigma_e^2$ and $\sigma_d^2$ control the smoothness of the coefficients in the nonparametric part since large variance means a possibly large jump between two adjacent coefficient while small variance increases the smoothness among the coefficients. The priors of $\sigma_e^2$ and $\sigma_d^2$ are typically inverse gamma distributions which make efficient way to construct priors for especially nonparametric part. For example, if we consider three independent variables into the parametric part, we may specify the normal prior for these four (including the intercept) coefficients. But when we model them into nonparametric part, the number of coefficients will be much larger even there are only three unknown functions. If the number knots for these three functions are 8,5 and 5, for instance, the number of coefficients will be $8 + 5 + 5 - 3 = 15$. It will seem dogmatic to directly specify the 15-dimension multivariate normal prior for these coefficients. If we introduce the smoothness parameters instead, we only have to specify two inverse gamma distributions for each unknown function.

Geweke (1993) find that within certain range, the change of the numbers and locations of the knots do not have significance effect on the unknown function approximation. Based on this, we assume equal bandwidth to simplify our model:

$$h_{mj} = \frac{\max(w_j) - \min(w_j)}{M_j - 1} \quad \forall m = 1, \cdots, M_j \tag{2.11}$$

## 2.5 Prior Specification and Posterior Computation

Since the $Z_{np}$ matrix does not have full rank, we need to apply the constraint that $\sum_{m=1}^{M_j} f_{mj} = 0$ for $j = 1, \cdots, q$. This means that each row vector $\boldsymbol{z}_{lj}^T$'s first element is dropped and all the other elements are subtracted by that dropped element. We denote that the $Z_{np}$ matrix is transformed in this way into $X_{np}$. Combining $X_0$ and $X_{np}$, we have the coefficient matrix $X \triangleq (X_0, X_{np})$. This coefficient matrix enables us to write our semiparametric model in a neat form $\boldsymbol{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, where $\boldsymbol{\beta}$ is $(\beta_0^T, (f_{21}, \cdots, f_{M_1 1})^T, \cdots, (f_{2q}, \cdots, f_{M_q q})^T)^T$.

The parameters of our interests are the cut point vector $\mathbf{a}$, the coefficient vector $\boldsymbol{\beta}$, the smoothness parameters $\sigma_e^2$, $\sigma_d^2$, parameters $\lambda_i$'s in error term, the allocation indicator vector $\boldsymbol{Z}$, the hyper parameters $\boldsymbol{\theta_H}$ in the weight functions and the precision parameter $\zeta$ for WDPM. Let $(a_{je0}, \delta_{je0})$, $(a_{jd0}, \delta_{jd0})$, $(\mathbf{b}_0, B_0)$, $(a_0, d_0)$, $(\mu_\psi,\ \sigma_\psi^2)$, $(a_\gamma,\ b_\gamma)$, $(a_\psi)$, and $(c_{\psi,0}$, $c_{\psi,00})$ be values that specify the priors of $\sigma_{ej}^2$, $\sigma_{dj}^2$, $\boldsymbol{\beta}$, $\zeta$ and $\boldsymbol{\theta_H}$ for WDPMD, WDPMG, WDPME, and WDPMH, respectively. Using these notations, the prior distributions and weight functions are specified as the following:

$$\begin{aligned}
\sigma_{ej}^2 &\sim \mathrm{IG}(a_{je0}, \delta_{je0}), \\
\sigma_{dj}^2 &\sim \mathrm{IG}(a_{jd0}, \delta_{jd0}), \\
\boldsymbol{\beta} &\sim \mathrm{N}(\mathbf{b}_0, B_0),
\end{aligned}$$

$$
\begin{aligned}
\Pr(Z_i = q|\boldsymbol{\theta_H}) &= \pi_{iq}, \\
&= \begin{cases}
\frac{\gamma_q e^{-\psi\left\|\boldsymbol{x}_i-\boldsymbol{x}_q^c\right\|^2}}{\sum_{l=1}^n \gamma_l e^{-\psi\left\|\boldsymbol{x}_i-\boldsymbol{x}_l^c\right\|^2}} & \text{if WDPMD;} \\
\frac{e^{-\psi\left\|\boldsymbol{x}_i-\boldsymbol{x}_q^c\right\|^2}}{\sum_{l=1}^n e^{-\psi\left\|\boldsymbol{x}_i-\boldsymbol{x}_l^c\right\|^2}} & \text{if EWDPM;} \\
\frac{e^{-\psi_q\left\|\boldsymbol{x}_i-\boldsymbol{x}_q^c\right\|^2}}{\sum_{l=1}^n e^{-\psi_l\left\|\boldsymbol{x}_i-\boldsymbol{x}_l^c\right\|^2}} & \text{if WDPMG or WDPME or WDPMH,}
\end{cases} \\
\boldsymbol{\theta_H} &\sim \begin{cases}
\text{log-N}(\psi|\mu_\psi, \sigma_\psi^2) \times \prod_{q=1}^n \text{Gamma}(\gamma_q|a_\gamma, b_\gamma) & \text{if WDPMD;} \\
\text{log-N}(\psi|\mu_\psi, \sigma_\psi^2) & \text{if EWDPM;} \\
\prod_{q=1}^n \text{Gamma}(\psi_q|a_\gamma, b_\gamma) & \text{if WDPMG;} \\
\prod_{q=1}^n \text{Exp}(\psi_q|a_\psi) & \text{if WDPME;} \\
\prod_{q=1}^n \text{C}^+(\psi_q|c_{\psi,0}, c_{\psi,00}) & \text{if WDPMH,}
\end{cases} \\
\lambda_i|(Z_i = q) &\sim \text{G}_q^c, \\
\text{G}_q^c &\sim \text{DP}(\zeta, G_0), \quad q = 1, \cdots, Q, \\
G_0 &= \text{Gamma}(\lambda_i|\frac{v}{2}, \frac{v}{2}), \\
\zeta &\sim \text{Gamma}(a_0, d_0), \\
\mathbf{a} &\sim \text{N}(\boldsymbol{a}_{00}, A_{00}).
\end{aligned}
$$

Suppose there are $p$ distinct values in $\boldsymbol{Z}$ and $(n_1, \cdots, n_p)$ denotes the numbers of observations that come from these $p$ candidate priors. Suppose the unqiue values among the $\lambda_i$'s that share the same candidate prior $\text{G}_r^c$ $(r = 1, \cdots, p)$ is $k_r$. We also denote the total number of distinct values in $(\lambda_1, \cdots, \lambda_n)$ as $k$ (i.e. $k = \sum_{r=1}^p k_r$) and the distinct values in $(\lambda_1, \cdots, \lambda_n)$ as $(\lambda_1^*, \cdots, \lambda_k^*)$. The indicator vector which allocates $(\lambda_1, \cdots, \lambda_n)$ to the $k$ distinct values is denoted as $\boldsymbol{S}$. Let $\pi(\boldsymbol{\theta_H})$ be the prior for $\boldsymbol{\theta_H}$ and $w(\boldsymbol{Z}|\boldsymbol{\theta_H})$ be the corresponding weight function. Based on these notations, the complete joint density will be proportional to:

$$
\begin{aligned}
&\prod_{j=1}^{q_1}(\frac{1}{\sigma_{ej}^2})^{\frac{\alpha_{ej0}}{2}-1}e^{-\frac{\delta_{ej0}}{2\sigma_{ej}^2}}][\prod_{j=1}^{q_1}(\frac{1}{\sigma_{dj}^2})^{\frac{\alpha_{dj0}}{2}-1}e^{-\frac{\delta_{dj0}}{2\sigma_{dj}^2}}][\prod_{j=1}^{q_1}(\frac{1}{\sigma_{ej}^2})(\frac{1}{\sigma_{dj}^2})^{\frac{M_j-3}{2}}][e^{-\frac{1}{2}(\boldsymbol{\beta}-\mathbf{b_0})^T B_0^{-1}(\boldsymbol{\beta}-\mathbf{b_0})}]\\
&\times [w(\boldsymbol{Z}|\boldsymbol{\theta_H})][\pi(\boldsymbol{\theta_H})]\\
&\times [\zeta^{a_0-1}e^{-d_0\zeta}]\Pr(k_1,\cdots,k_p|\zeta,\boldsymbol{S},\boldsymbol{Z})\Pr(\boldsymbol{S}|k)\prod_{l=1}^{k}dG_0(\lambda_l^*)\\
&\times [\prod_{i=1}^{n}\lambda_i^{\frac{1}{2}}e^{-\frac{1}{2}\lambda_i(y_i^*-\boldsymbol{x}_i^T\boldsymbol{\beta})^2}]\\
&\times [e^{-\frac{1}{2}(\mathbf{a}-\mathbf{a_{00}})^T A_{00}^{-1}(\mathbf{a}-\mathbf{a_{00}})}][\prod_{j'=0}^{J-1}\mathbf{1}_{\{c_{j'-1}<y_i^*<c_{j'}\}}\mathbf{1}_{\{y_i=j'\}}].
\end{aligned}
$$

It should be addressed that the second line of this joint density is only necessary for WDPM models. For Student-t or DPM model, it should be dropped or fixed at 1. The third line of this density can be simplified as $[\zeta^{a_0-1}e^{-d_0\zeta}]\Pr(k|\zeta,\boldsymbol{S})\Pr(\boldsymbol{S}|k)\prod_{l=1}^{k}dG_0(\lambda_l^*)$ in DPM model. As for Student-t model, we know that $\zeta$ will go to $\infty$ and this line can be further simplified as $\prod_{l=1}^{n}dG_0(\lambda_l^*)$. The forth line is the likelihood of the response (or latent response) given all the parameters, and this part appears in all the models' full likelihoods. The last line is only necessary for ordinal outcome case, and in continuous outcome model it should be dropped or fixed at 1. In addition, for the continuous response case, we do not have to assume zero means for the error terms. By defining $\phi_i = (\mu_i, \sigma_i^2)$, we can have $\varepsilon_i|\phi_i \sim N(\mu_i, \sigma_i^2)$ and $G_0 = N(\mu_i|0, \sigma_i^2) \times \text{inv gamma}(\sigma_i^2|\frac{a}{2}, \frac{b}{2})$. We also define $\phi_i$ to be $\lambda_i$ in ordinal response case. By doing so, $\phi_i$ will be parameter(s) whose prior can be constructed by DPM or WDPM in both continuous and ordinal cases.

Our MCMC can be viewed as the extension of framework of the DPM MCMC sampling. We use the ordinal case to illustrate because it can be generalized to continuous case straightforwardly. Once we have sampled $\mathbf{a}$ using Metropolis-Hastings (M-H) and $\boldsymbol{\beta}$, $\sigma_e^2$ and $\sigma_d^2$ using

Gibbs samplings, the following Steps 1-3 are applied:

Step 1: Sample the error term parameters $\lambda_i$'s and the allocation indicator vector $\boldsymbol{Z}$; It seems convenient to directly sample $\boldsymbol{Z}$ based on the weights (i.e. the prior of $\boldsymbol{Z}$). Unfortunately, this is not applicable if $\Pr(k_1, \cdots, k_p|\zeta, \boldsymbol{S}, \boldsymbol{Z})$ is not constant of $\boldsymbol{Z}$; Hence we conduct the following steps (a)-(c):

(a) Sample $\boldsymbol{S}$ based on generalized P$\acute{o}$lya urn result introduced in Section 2.3.

(b) Update the $k$ distinct values in the $\boldsymbol{\lambda}$ vector based on its posterior, which are simply several Gamma distributions.

(c) Update $\boldsymbol{Z}$ based on the result of Sun et al (2015). We denote $\boldsymbol{C} = (C_1, \cdots, C_k)$ as the allocation of the $k$ distinct values sampled in (b) to the $n$ candidate priors. For example, $C_h = q$ means all of the observations which are assigned to the $h_{th}$ unique value of $\lambda$ select the $q^{th}$ candidate prior as their true prior. We can see that updating $\boldsymbol{C}$ is equivalent to updating $\boldsymbol{Z}$, because the observations that share the same unique value of $\boldsymbol{\lambda}$ must share the same candidate prior (this is the very important fact introduced in Section 2.3). The following posterior of $\boldsymbol{C}$ is true for any WDPM (not only the ones we propose in our study), and can be viewed as the global rule for sampling $\boldsymbol{Z}$ in WDPM:

$$\Pr(C_h = q|\boldsymbol{C}_{-h}, \boldsymbol{S}, \boldsymbol{y}, \zeta, \theta_H) \propto \frac{\Gamma[(\zeta + n_{q(-h)}]}{\Gamma[\zeta + n_{q(-h)} + n_{(h)}]} \prod_{i=1}^{n_{(h)}} \pi_{iq} \tag{2.12}$$

In (2.12) $n_{q(-h)}$ is the number of observations which select the $q^{th}$ candidate and are not assigned to the $h^{th}$ unique value in $\boldsymbol{\lambda}$. This number is fixed once $\boldsymbol{C}_{(-h)}$ is given. $n_{(h)}$ is the number of observations that are assigned to the $h^{th}$ unique value. This result will be listed as a theorem in Chapter 3 and the proof can be found in Sun et al (2015).

Step 2: Sample the hyper parameters that are used to construct the weight functions; For (2.4) and (2.5) we can apply M-H in sampling $\psi$. For (2.3), several Gibbs sampling steps will be added to sample $\gamma_j$'s according to the method enlightened by Dunson and Stanford (2005) and Holmes and Held (2006) and introduced in Dunson et al (2007).

Step 3: Sample the precision parameter $\zeta$, which can be viewed as the extension of the result in Escobar and West (1995).

The MCMC sampling procedures for semiparametric model with WDPM error are summarized in Appendix A.

## 2.6 Marginal Likelihood Computation for WDPM

The marginal likelihood computation enables us to make model comparison through which we select the best model among the feasible choices. In this section, we explain how to derive marginal likelihood for WDPM which is the generalized form of DPM and the extension of the work by Basu and Chib (2003).

Let $f(\boldsymbol{y}|X)$ be the marginal likelihood. Define $\Theta = (\sigma_e^2, \sigma_d^2, \mathbf{a}, \zeta, \boldsymbol{\beta})$. Then log of the marginal likelihood can be written as:

$$\log\{f(\boldsymbol{y}|X)\} = \log\{f(\boldsymbol{y}|X, \Theta)\} + \log\{\pi(\Theta)\} - \log\{f(\Theta|\boldsymbol{y}, X)\}, \tag{2.13}$$

where $\pi(\Theta)$ is calculated using the prior distribution of $\Theta$, which is simply plugging the posterior mean of $\Theta$ into the priors; The calculation of the likelihood $f(\Theta|\boldsymbol{y}, X)$ is straightforward as well. But the calculation of $f(\boldsymbol{y}|\Theta, X)$ needs extra MCMC sampling applying the method introduced in Chib (1995), and this computation is the most challenging part of the marginal likelihood computation and is essentially realized by sequential importance sampling.

We provide the algorithm for computing the marginal likelihood for ordinal outcome case. Let $(c_1^*, \cdots, c_{J-2}^*; \boldsymbol{\beta}^*; \boldsymbol{\pi_1^*}, \cdots, \boldsymbol{\pi_n^*})$ be the estimation of the cut points, coefficients in the semiparametric model and the weight vectors for the $n$ observations (i.e. $\boldsymbol{\pi_i^*} = (\pi_{i1}, \cdots, \pi_{iQ})^T$), and $G_0$ =gamma$(\frac{v}{2}, \frac{v}{2})$. The marginal likelihood can be calculated by the following Steps M1-M5:

Step M1: Set $u_1 = T_v(c_{y_1}^* - \boldsymbol{x}_1^T\boldsymbol{\beta}^*, 1) - T_v(c_{y_1-1}^* - \boldsymbol{x}_1^T\boldsymbol{\beta}^*, 1)$. Draw $y_1^* \sim t_v(x_1^T, 1)\mathbf{1}(c_{y_1-1}^*, c_{y_1}^*)$, $Z_1$ ~multinom$(\boldsymbol{\pi_1^*})$, and $S_1 = 1$. $T_v(\cdot)$ is the cdf of student-t with degree of freedom $v$.

Step M2: For each $i$ observation, $Z_i$ ~multinom$(\boldsymbol{\pi_i^*})$; denote the observations $\{1, \dots i-1\}$ that share the same value in $\boldsymbol{Z}$ with observation $i$ as $(i-1) \sim Z_i$ and check the $\boldsymbol{S}$ values for these observations. Suppose that there are $k_{(i-1)\sim Z_i}$ distinct values of $\boldsymbol{S}$ among these observations. For the $r^{th}$ value of these $k_{(i-1)\sim Z_i}$ distinct values, we denote it as $S_{(i-1)\sim Z_i,r}$ and denote $n_{(i-1)\sim Z_i,r}$ as the number of observations that fall into this cluster. Set $a_{(i-1)\sim Z_i,r} = v + n_{(i-1)\sim Z_i,r}$ and $b_{(i-1)\sim Z_i,r} = v + \sum_{\substack{l\in(i-1)\sim Z_i \\ S_l = S_{(i-1)\sim Z_i,r}}} (y_l^* - \boldsymbol{x}_l^T\boldsymbol{\beta}^*)^2$. Then we have:

$$
\begin{aligned}
u_i =& \frac{\zeta^*}{\zeta^* + \sum_{j<i}\mathbf{1}(Z_j = Z_i)}[T_v(c_{y_i}^* - \boldsymbol{x}_i^T\boldsymbol{\beta}^*, 1) - T_v(c_{y_i-1}^* - \boldsymbol{x}_i^T\boldsymbol{\beta}^*, 1)] \\
& \frac{1}{\zeta^* + \sum_{j<i}\mathbf{1}(Z_j = Z_i)} \sum_{r=1}^{k_{(i-1)\sim Z_i}} n_{(i-1)\sim Z_i,r}[T_{a_{(i-1)\sim Z_i,r}}(c_{y_i}^* - \boldsymbol{x}_i^T\boldsymbol{\beta}^*, a_{(i-1)\sim Z_i,r}^{-1} b_{(i-1)\sim Z_i,r}) \\
& - T_{a_{(i-1)\sim Z_i,r}}(c_{y_i-1}^* - \boldsymbol{x}_i^T\boldsymbol{\beta}^*, a_{(i-1)\sim Z_i,r}^{-1} b_{(i-1)\sim Z_i,r})]
\end{aligned}
$$

Step M3: Sample the $\boldsymbol{S}$ value for the $i^{th}$ observation. Draw

$$
\begin{aligned}
y_i^* \sim & \frac{\zeta^*}{\zeta^* + \sum_{j<i} \mathbf{1}(Z_j = Z_i)} t_v(\boldsymbol{x}_i^T \boldsymbol{\beta}^*, 1) \\
& + \frac{1}{\zeta^* + \sum_{j<i} \mathbf{1}(Z_j = Z_i)} \sum_{r=1}^{k_{(i-1)\sim Z_i}} n_{(i-1)\sim Z_i, r} t_{a_{(i-1)\sim Z_i, r}}(\boldsymbol{x}_i^T \boldsymbol{\beta}^*, a_{(i-1)\sim Z_i, r}^{-1} b_{(i-1)\sim Z_i, r})
\end{aligned}
$$

which is truncated to $(c_{y_1-1}^*, c_{y_1}^*)$ and if $\sum_{j<i} \mathbf{1}(Z_j = Z_i) \neq 0$, obtain

$$
S_i = \begin{cases} r \quad \text{with} \;\; p \propto \frac{n_{(i-1)\sim Z_i, r}}{\zeta^* + \sum_{j<i} \mathbf{1}(Z_j = Z_i)} t_{a_{(i-1)\sim Z_i, r}}(y_i^* | \boldsymbol{x}_i^T \boldsymbol{\beta}^*, a_{(i-1)\sim Z_i, r}^{-1} b_{(i-1)\sim Z_i, r}) \\ k_{i-1} + 1 \quad \text{with} \;\; p \propto \frac{\zeta_*}{\zeta^* + \sum_{j<i} \mathbf{1}(Z_j = Z_i)} t_v(y^* | \boldsymbol{x}_i^T \boldsymbol{\beta}^*, 1); \end{cases}
$$

If $\sum_{j<i} \mathbf{1}(Z_j = Z_i) = 0$, $S_i = 1$

Step M4: Repeat Steps M2-M3 for $i = 2, \ldots, n$

Step M5: Derive the conditional likelihood as $f(\boldsymbol{y}|\Theta, X) = \prod_{i=1}^n u_i$

Step M6: Repeat Steps M1-M5 for $B$ times and calculate the average of log-likelihood.

It is shown in Irwin et al (1994) that the coefficient of variation is approximately proportional to $B^{-0.5}$ for a given standard error of log-likelihood. We set $B = 5000$ to make the coefficient of variation lower than 0.02.

## 2.7 Simulation Study

We conduct simulation study to investigate the performance of several schemes of weights for WDPM in terms of the marginal likelihood. We consider seven candidate models for error terms $\varepsilon$ and denote them as (1) Student t, (2) DPM, (3) WDPMD, (4) EWDPM, (5) WDPMG, (6) WDPME and (7) WDPMH. Both $x$ and $w_j$'s are sampled independently from Uniform(0,1), $j = 1, \ldots, 3$. The true model is:

$$y_i = 5 + 3x_i + g_1(w_{i1}) + g_2(w_{i2}) + g_3(w_{i3}) + \varepsilon_i \quad i = 1, \ldots, n \tag{2.14}$$

where $n$ is the sample size and the underlying forms of the unknown functions are given by:

$$\begin{aligned} g_1(w_1) &= 8w_1 + \sin(4\pi w_1), \\ g_2(w_2) &= -1.5 - w_2 + \exp[-30(w_2 - 0.5)^2], \\ g_3(w_3) &= 6w_3^3(1 - w_3). \end{aligned}$$

To generate error terms, we use $G_0 = \mathrm{N}(\mu_i|0, \sigma_i^2) \times$ inv-Gamma$(\sigma_i^2|2.25, 0.75)$ as $G_0$ in continuous response case and use Gamma(2.5,2.5) as $G_0$ to generate $\lambda_i$'s in ordinal response case. For Student-t model, $G_0$ is directly used as the prior of error term parameters. For other models which include sampling one or multiple distributions from this base distribution, the sampling is conducted by using "stick-breaking" algorithm (Sethuraman, 1994). In DPM model, only one distribution is sampled and $\phi_i$'s are independently generated from this sampled G($\cdot$). In the weighted models, multiple candidate prior distributions are sampled independently from $G_0$ for each model. The entire process of generating the error terms for WDPM can be summarized as:

Step S1: Sample the candidate priors of $\phi_i$ independently from $G_0$;

Step S2: Sample the hyper parameters in the weight functions based on the assumed hyper priors, and calculate the weights using specified weight function;

Step S3: For each observation, randomly choose the prior of $\phi_i$ from the candidates based on the weights;

Step S4: Sample $\phi_i$ based on its prior;

Step S5: Sample $\varepsilon_i$ based on $\varepsilon_i|\phi_i \sim \mathrm{N}(\mu_i, \sigma_i^2)$.

The hyper parameters for weight function are sampled via:

$$\boldsymbol{\theta_H} \sim \begin{cases} \text{log-N}(\psi|0.262, 0.05) \times \prod_{q=1}^{n} \text{Gamma}(\gamma_q|0.1, 100) \ \text{ if WDPMD;} \\ \text{log-N}(\psi|0.262, 0.05) \ \text{ if EWDPM;} \\ \prod_{q=1}^{n} \text{Gamma}(\psi_q|2, 1) \ \text{ if WDPMG;} \\ \prod_{q=1}^{n} \text{Exp}(\psi_q|2.42) \ \text{ if WDPME;} \\ \prod_{q=1}^{n} \text{C}^{+}(\psi_q|0, 0.3) \ \text{ if WDPMH.} \end{cases}$$

The precision parameter $\zeta$ is chosen to be 5 in the "stick-breaking" algorithm and the numbers of knots for the three unknown functions are $(8, 5, 5)$.

We apply MCMC sampling and marginal likelihood calculation introduced in Section 2.5 and Section 2.6 to make model comparison. The first 2500 draws of the MCMC are discarded and the following $20,000$ draws are used to derive posterior mean. This setting of burn-in and maximum number of iterations is applied to MCMC of each model fitting in both simulation and empirical study. We compile C++ functions in R and utilize parallel computation on multiple CPU cores to improve computation efficiency. It takes 49 minutes to complete MCMC and marginal likelihood derivation of a simulated continuous-response data set with $n = 500$ fitted by EWDPM on a single core of Intel Xeon E5-2687 CPU.

In both continuous and ordinal case, we generate 200 data sets from each candidate model and fit each data set by the seven candidate models. In the continuous response case, we set $n = 1000$; In ordinal response case, we set $n = 500$ because sampling cut points requires likelihood maximization in each MCMC iteration.

For each combination of true model and fit model, we record the log of marginal likelihood (log(ML)) of each fitting. For each true model, we can compare the fit models through log(ML) based on the 200 repetitions. We find that paired-t test is not proper because the differences in log(ML) violate normality assumption, therefore we adopt Wilcoxon signed rank test. Table 2.1 contains the results of continuous case. The purpose of this table is to see whether data sets generated from a true model can be best explained by the same model. Here "best explained" means producing the maximum marginal likelihood. In each row of the table, the fit model which produces highest average log(ML) is labeled as "best model". In each cell of this table, we display the average log(ML) along with the p-value of signed rank test which compares the corresponding fit model to the best model of the same row.

Table 2.1: Simulation results for continuous case; In each cell of this table, we display the average log(ML) along with the p-value of signed rank test which compares the corresponding fit model to the best model of the same row.

| Fit Model<br>True Model | Student-t | DPM | WDPMD | EWDPM | WDPMG | WDPME | WDPMH |
|---|---|---|---|---|---|---|---|
| Student-t | -685.14<br>best model | -690.86<br>(2.651e-06) | -695.53<br>(7.693e-11) | -689.70<br>(1.992e-05) | -691.44<br>(3.759e-06) | -690.69<br>(6.017e-06) | -690.44<br>(4.581e-06) |
| DPM | -716.06<br>(3.039e-15) | -693.67<br>best model | -697.11<br>(5.233e-05) | -697.43<br>(4.415e-05) | -709.22<br>(2.989e-16) | -698.30<br>(1.790e-09) | -697.21<br>(1.656e-07) |
| WDPMD | -722.93<br>(0) | -700.30<br>(0.3556) | -700.75<br>(1.055e-04) | -702.34<br>(7.188e-05) | -700.01<br>best model | -701.57<br>(1.522e-05) | -708.82<br>(3.954e-15) |
| EWDPM | -718.12<br>(0) | -696.72<br>(1.937e-06) | -697.51<br>(7.166e-05) | -694.08<br>best model | -695.73<br>(0.1914) | -695.25<br>(0.2651) | -696.17<br>(1.795e-06) |
| WDPMG | -714.30<br>(0) | -691.36<br>(3.587e-07) | -694.24<br>(6.740e-10) | -693.10<br>(2.601e-08) | -689.68<br>best model | -691.62<br>(9.267e-07) | -692.79<br>(9.953e-08) |
| WDPME | -714.03<br>(0) | -694.60<br>(8.005e-06) | -694.29<br>(3.824e-06) | -692.96<br>(0.4117) | -694.63<br>(7.580e-06) | -692.27<br>best model | -693.75<br>(2.694e-04) |
| WDPMH | -717.09<br>(0) | -691.21<br>(3.249e-08) | -693.62<br>(5.091e-10) | -689.38<br>(7.100e-06) | -689.09<br>(4.293e-05) | -689.81<br>(0.0018) | -687.24<br>best model |

In Table 2.1, we can see that all of the true models are best explained by themselves except WDPMD, where WDPMG is the best fit model. We can also observe that the best models in most rows are significantly better than other models of the same row. The exception is in the WDPMD row where DPM does not significantly deviate from WDPMG, and in the WDPME row where WDPME is not significantly better than EWDPM.

In ordinal case. We set cut points $(6.31, 9.04, 11.95)$ for the seven types of true models. Similar as Table 2.1, the results are displayed in Table 2.2.

Table 2.2: Simulation results for ordinal case; In each cell of this table, we display the average log(ML) along with the p-value of signed rank test which compares the corresponding fit model to the best model of the same row.

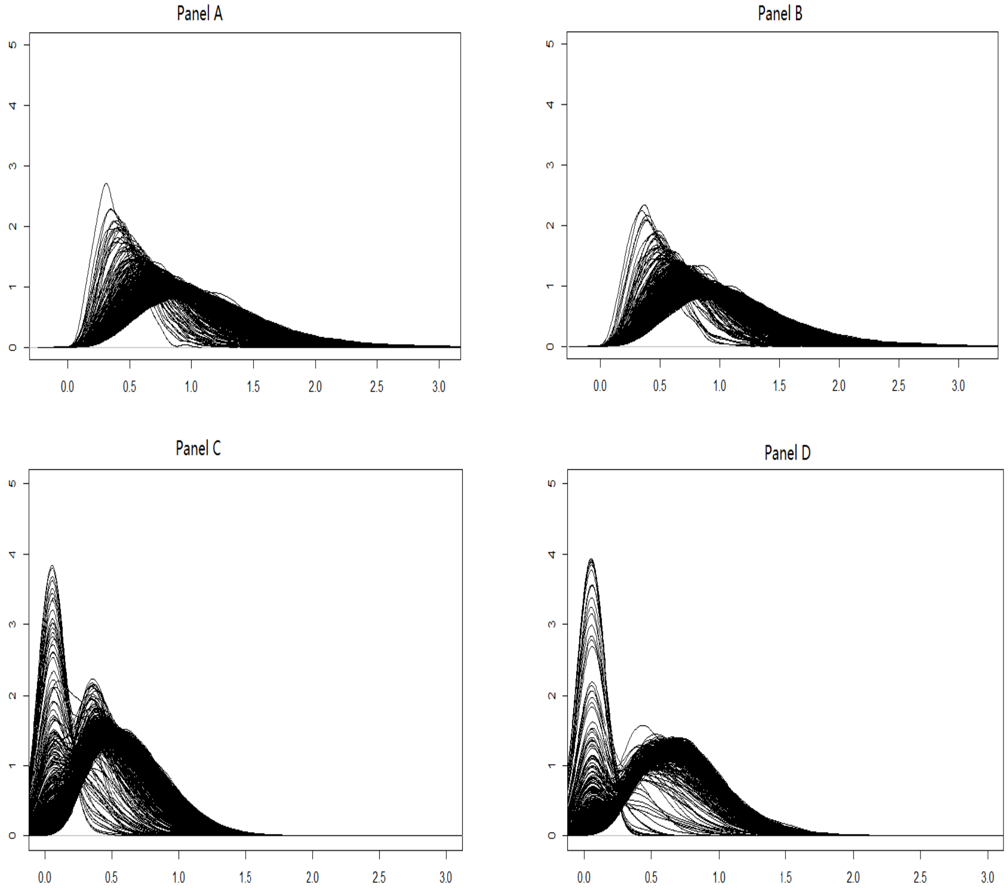
| Fit Model<br>True Model | Student-t | DPM | WDPMD | EWDPM | WDPMG | WDPME | WDPMH |
|---|---|---|---|---|---|---|---|
| Student-t | -263.57<br>best model | -267.82<br>(4.518e-09) | -269.10<br>(5.097e-13) | -265.54<br>(3.383e-06) | -267.49<br>(2.108e-08) | -266.21<br>(7.732e-07) | -265.05<br>(1.937e-05) |
| DPM | -274.02<br>(3.701e-11) | -257.81<br>best model | -271.32<br>(6.988e-07) | -270.53<br>(1.136e-06) | -269.73<br>(5.628e-06) | -269.94<br>(7.094e-06) | -268.09<br>(0.2323) |
| WDPMD | -280.91<br>(0) | -272.67<br>(8.379e-07) | -267.59<br>best model | -269.41<br>(9.613e-05) | -270.01<br>(2.035e-05) | -269.27<br>(2.618e-04) | -268.75<br>(4.023e-04) |
| EWDPM | -274.68<br>(2.793e-14) | -266.04<br>(0.3551) | -268.86<br>(2.627e-06) | -265.53<br>best model | -268.63<br>(3.944e-06) | -267.94<br>(9.738e-06) | -266.15<br>(0.2954) |
| WDPMG | -272.22<br>(2.781e-12) | -264.53<br>(4.750e-05) | -264.74<br>(2.338e-04) | -264.31<br>(6.895e-04) | -265.12<br>(7.252e-05) | -263.47<br>(0.3817) | -262.89<br>best model |
| WDPME | -274.60<br>(0) | -269.81<br>(9.487e-07) | -267.61<br>(4.324e-05) | -267.13<br>(1.790e-05) | -266.36<br>(8.611e-05) | -265.93<br>(0.4174) | -265.22<br>best model |
| WDPMH | -278.19<br>(0) | -469.89<br>(5.842e-06) | -266.73<br>(1.168e-04) | -265.91<br>(0.1625) | -266.08<br>(0.0537) | -266.20<br>(6.734e-04) | -265.74<br>best model |

In Table 2.2, we can see that four out of the seven true models can be best explained by themselves. For the other three cases, the best explanatory model is WDPMH. We can also observe that the overall performance of WDPMH is better than other models because even in the four cases where WDPMH is not the best, this model is either closest to best models (the first and third row) or produce no significant deviation from best models (the second and fourth row). As a conclusion of simulation study, it shows that when the error terms are simulated by WDPM, we do need weighted models to guarantee accurate fitting.

It is also worthy of mentioning that the concentration parameter $\zeta$ is severely underestimated in ordinal case. For example, the posterior mean of $\zeta$ in the continuous case is 5.13 by DPM model, but it is 1.15 when it comes to ordinal case by the same model. The true value of $\zeta$ is 5, so the model comparison in the ordinal case is less convincing than that in the continuous case since parameter estimation is less accurate.

Since DPM and WDPM are essentially clustering observations by unique values (in $\boldsymbol{\lambda}$ for this study), we are interested to check whether the clusters do exist in the posterior densities of $(\lambda_1, \cdots, \lambda_n)$. Figure 2.1 displays such results in four cases. Panel A is the result of data generated by Student-t fitted by Student-t (continuous response and $n = 1000$); Panel B is the result of data generated by Student-t fitted by DPM (ordinal response and $n = 1000$); Panel C is the result of data generated by WDPMG fitted by WDPMG (continuous response and $n = 1000$); Panel D is the result of data generated by DPM fitted by DPM (ordinal response and $n = 500$). We can see that data generated from DPM or WDPMP (Panel C or Panel D) display more clear pattern of multiple clusters compared to data generated from Student-t (Panel A or Panel B). There are at least two major modes in Panel C or Panel D and a clear trough between the two modes. Even if we use DPM to fit Student-t data, we can not see such significant visual evidence of multiple clusters (in Panel B). We can conclude that existence of multiple clusters depend more on the true model than on the fitting approach.

Figure 2.1: Plots of posterior density curves of $\lambda_t$ $(t = 1, \cdots, n)$; In each of the panels, there are $n$ curves, where $n$ is the sample size; Each curve represents the posterior density of an observation's $\lambda$ value based on $20,000$ MCMC draws; The y-axis is kernel density estimated value, and x-axis is the range of $\lambda_t$'s.



## 2.8 Application: Credit Rating Data

We apply weighted DPM methods to the credit rating data illustrated in Verbeek (2008). We have the observed values of the 921 farms Standard and Poor credit rating (the response), book value of debt divided by assets ($\boldsymbol{w}_1$), earnings before interest, taxes divided by total assets ($\boldsymbol{w}_2$), log of sales ($\boldsymbol{w}_3$), retained earnings divided by total assets ($\boldsymbol{w}_4$) and working capital divided by total assets ($\boldsymbol{w}_5$). The response values are ordinal (from 0 to 4). The five covariates ($\boldsymbol{w}_1, \cdots, \boldsymbol{w}_5$) will enter the model nonparametrically and there is only the intercept for the parametric part.

For EWDPM and WDPMD, the prior for $\psi$ is specified as log-N(-2.22, 0.5). In WDPMD, the prior for $\gamma_q$'s is given as gamma(1, $n$), where $n$ is the number of observations of this empirical data set. The three types of priors for $\psi_q$'s in WDPMG,WDPME and WDPMH are correspondingly gamma(1.03, 2.17), Exp(0.62) and $C^+(0, 0.13)$. One can refer to Chib and Greenburg (2010) to check the details about the other priors' settings. We apply our MCMC algorithm and the marginal likelihood computation described in Section 2.6 in order to perform model comparison.

We consider using different numbers and locations of candidate priors to seek model performance. Randomized location means that we do not use the observations' locations as candidate location. Instead, we derive the range of each covariate based on data and sample the five components of the location from their corresponding range uniformly. Each candidate's location is sampled in this way independently. We consider such randomized location prior for $Q$=10, 200 ,1000 and 1200. We also consider using subsets of the observations rather than all to specify location. In Table 2.3, WDPMD-200ob means that we use WDPMD and randomly select 200 observations' locations as the candidate locations. WDPMD-1000r means we randomly sample 1000 locations as the candidate locations. Using the number of knots being equal to 4, 5 and 6 for each unknown function and applying WDPMD and WDPMG, we consider different combinations of $Q$, location, the number of knots and WDPM prior. In Table 2.3 we display the comparison results of these combinations using Bayes factor. We set Student-t model using 4 knots for each unknown function as the benchmark model (i.e. its Bayes factor is 1) and calculate other models' Bayes factor with respect to it. Bayes factor is essentially the ratio of marginal likelihood.

Table 2.3: Bayes factors of WDPMD and WDPMG with different $Q$ and candidate locations with respect to Student-t model using 4 knots for each unknown function.

| Number of knots<br>Candidate Models | 4 | 5 | 6 |
|---|---|---|---|
| t | 1.000 | 120.226 | 114.815 |
| WDPMD-10r | 0.009 | 4.365 | 0.224 |
| WDPMD-200r | 0.017 | 8.318 | 0.468 |
| WDPMD-1000r | 0.047 | 12.882 | 0.603 |
| WDPMD-1200r | 0.051 | 7.413 | 0.603 |
| WDPMD-10ob | 0.011 | 2.818 | 0.550 |
| WDPMD-200ob | 0.017 | 3.311 | 0.759 |
| WDPMD-921ob | 0.069 | 30.903 | 1.660 |
| WDPMG-10r | 0.013 | 5.888 | 0.513 |
| WDPMG-200r | 0.018 | 13.183 | 1.096 |
| WDPMG-1000r | 0.102 | 14.791 | 1.698 |
| WDPMG-1200r | 0.040 | 15.488 | 1.514 |
| WDPMG-10ob | 0.004 | 7.413 | 0.562 |
| WDPMG-200ob | 0.019 | 20.417 | 0.813 |
| WDPMG-921ob | 0.229 | 34.674 | 1.175 |

In Table 2.3 we can see that $Q = 10$ leads to the worst model performance in each column. $Q = 200$ results in model improvement, but the marginal likelihoods are still significantly worse than that of Student-t model. If we use $Q = 1000$ for randomized locations or $Q = 921$ for observations' locations, the model performances will be better than those with $Q = 200$, but such improvement is not adequate because the Bayes factors of Student-t with respect to the weighted models using the same number of knots are all higher than 3. We also observe that the ability to improve model performance reaches to a limit as $Q$ goes beyond 1000. For $Q = 1200$ with randomized locations, we actually do not see any model improvement compared to $Q = 1000$. The marginal likelihoods of the models using observations to specify the candidate locations are generally better than those using randomized locations, and such result is not sensitive to the number of knot.

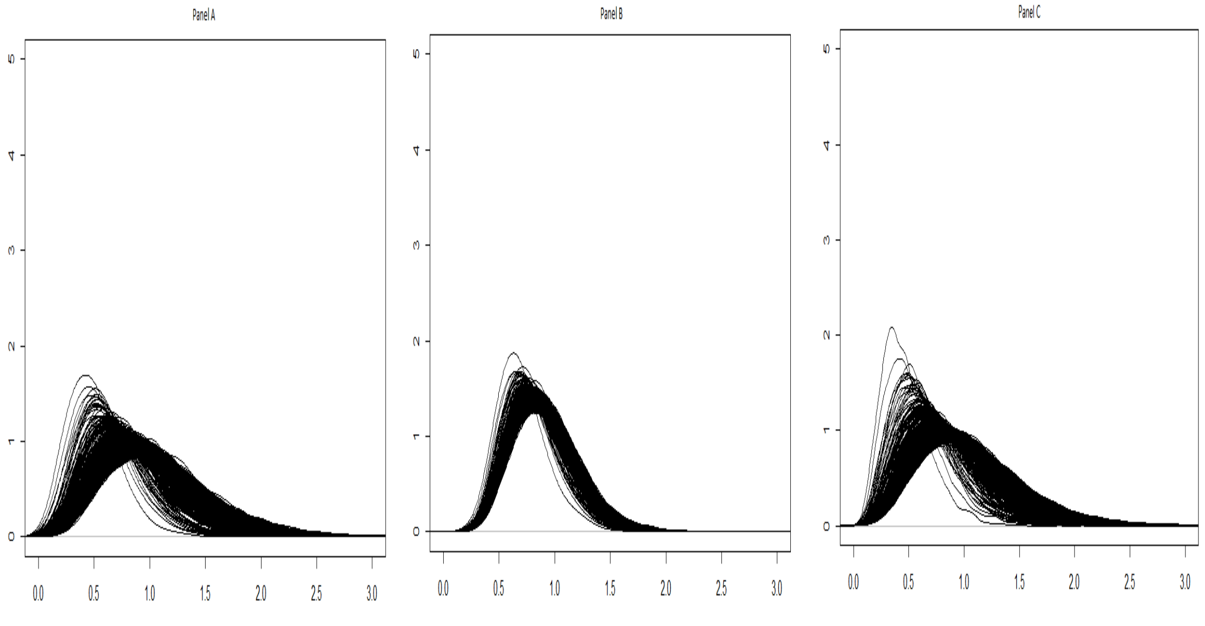
Based on these results, we use $Q = n$ and observations' locations as proposed by Dunson et al (2007) and explore different weight functions to seek further model performance. In Table 2.4, seven types of models in three knot-number cases are compared. We still set Student-t model using 4 knots for each unknown function as the benchmark model, and the Bayes factors are displayed. For the names of weighted approach models, we drop the extensions because they are all followed by "-921ob". Similar as Table 2.3, the models using 5 knots for each unknown function are better than those using 4 or 6. In all of the three cases the best three models are always Student-t, EWDPM and WDPMH. For 4-knot case WDPMH is favored over EWDPM and for 6-knot case EWDPM performs better than WDPMH. All the weighted DPM models produce better marginal likelihood than DPM in all of the three cases. In 4-knot and 5-knot cases, EWDPM and WDPMH are the two weighted approaches with respect to which the Bayes factor of Student-t is less than 3.

Table 2.4: Bayes factors of candidate models with respect to Student-t model using 4 knots for each unknown function.

| Number of knots<br>Candidate Models | 4 | 5 | 6 |
|---|---|---|---|
| t | 1.000 | 120.226 | 114.815 |
| DPM | 0.010 | 5.623 | 0.457 |
| WDPMD | 0.069 | 30.903 | 1.670 |
| EWDPM | 0.372 | 47.863 | 20.893 |
| WDPMG | 0.229 | 34.674 | 1.175 |
| WDPME | 0.246 | 13.490 | 0.575 |
| WDPMH | 0.676 | 56.234 | 5.129 |

The result that Student-t models perform better than DPM or WDPM models in this empirical data can be explained by the visual evidence in Figure 2.2, where Panel A is the result of Empirical data (Verbeek, 2008) fitted by Student-t and Panel B is the result of same data fitted by DPM, and Panel C is the result of WDPME. As we did in simulation study, we check whether multiple clusters exist in the posterior densities of $(\lambda_1, \cdots, \lambda_n)$. We can see that the Panel A and C are similar while Panel B is somewhat different from them. This reflects the results that the marginal likelihood of WDPM is closer than DPM to that of Student-t model. What is more important is that one of the panels gives us evidence that multiple clusters exist among $(\lambda_1, \cdots, \lambda_n)$. It suggests that the real data may favor the assumption that posterior densities of $\lambda_t$'s are based on parametric prior.

Figure 2.2: Plots of posterior density curves of $\lambda_t$ $(t = 1, \cdots, n)$; In each of the panels, there are $n$ curves, where $n$ is the sample size; Each curve represents the posterior density of an observation's $\lambda$ value based on $20,000$ MCMC draws; The y-axis is kernel density estimated value, and x-axis is the range of $\lambda_t$'s.



We furthermore conduct model comparison based on fitting accuracy. Once we have the estimated values of $\boldsymbol{\beta}$, the cut point vector $\mathbf{a}$ and error term parameter $\lambda_i$'s, we can calculate the probability distribution of a certain observation's credit rating based on its covariate information and the model estimation:

$$\Pr(y_i = j|\boldsymbol{\lambda}, \boldsymbol{\beta}, \boldsymbol{a}) = \Phi([c_j - \boldsymbol{x}_i^T\boldsymbol{\beta}]\lambda_i^{1/2}) - \Phi([c_{j-1} - \boldsymbol{x}_i^T\boldsymbol{\beta}]\lambda_i^{1/2}), \quad j = 0, 1, 2, 3, 4, \tag{2.15}$$

where $c_j$ is the cut-point that are calculated based on $\mathbf{a}$ as mentioned in Section 2.4. Based on the fitted probability distribution of credit rating, we can assign a certain observation to the category of rating which it is most likely to fall into and check the proportion of the observations that are correctly rated based on the observed credit rating. Based on Table 2.4 we can see that the model performances in 5-knot case are generally better than those in the other cases, so we display the result of fitting accuracy comparison for only 5-knot case in Table 2.5. Since the fitted values are derived based on all of the 921 observations, this accuracy is related to which of the seven candidate models is most powerful for this particular credit rating data.

Table 2.5: The proportion of the observations that are correctly rated based on the observed credit rating.

| | t | DPM | WDPMD | EWDPM | WDPMG | WDPME | WDPMH |
|---|---|---|---|---|---|---|---|
| Fitting Accuracy | 54.72 | 54.07 | 53.31 | 54.40 | 54.29 | 53.20 | 55.05 |

It turns out that WDPMH produces the best fitting result (55.05%) based on Table 2.5, the second best one is Student-t model (54.72%), and the third one is EWDPM. The best two models with 5 knots for each unknown function also produce the best marginal likelihood. Hence we view these two models as the most competitive models for this credit data. Table 2.6 displays these two models' parameter inferences based on MCMC for the three free cut points $(c_1, c_2, c_3)$. Posterior mean, standard deviation, median, lower and upper quantiles of MCMC samples are described. The credit rating for observation $i$ will be 0 if $y_i^* < 0$, 1 if

$0 \leq y_i^* < c_1$, 2 if $c_1 \leq y_i^* < c_2$, 3 if $c_2 \leq y_i^* < c_3$ and 4 if $y_i^* \geq c_3$. We can see that results of the two models are similar to each other.

Table 2.6: Estimated three free cut points using the best two models for credit rating data; Posterior mean, standard deviation (SD), median, lower and upper quantiles of MCMC draws are displayed.

| | Student-t with 5 knots for each unknown function | | | | |
|---|---|---|---|---|---|
| | Posterior mean | SD | Median | 2.5 % quantile | 97.5 % quantile |
| $c_1$ | 1.753 | 0.094 | 1.751 | 1.578 | 1.944 |
| $c_2$ | 3.224 | 0.127 | 3.225 | 2.988 | 3.485 |
| $c_2$ | 5.197 | 0.195 | 5.206 | 4.830 | 5.599 |
| | WDPMH with 5 knots for each unknown function | | | | |
| | Posterior mean | SD | Median | 2.5 % quantile | 97.5 % quantile |
| $c_1$ | 1.731 | 0.089 | 1.732 | 1.564 | 1.906 |
| $c_2$ | 3.190 | 0.121 | 3.196 | 2.959 | 3.449 |
| $c_3$ | 5.133 | 0.199 | 5.128 | 4.771 | 5.546 |

Figure 2.3 and Figure 2.4 illustrate how the expected values of $g_j|\boldsymbol{y}$, $E(g_j|\boldsymbol{y})$, changes as the value of the $j^{th}$ covariate changes. These two figures help analyze how the five covariates contribute to the latent response $\boldsymbol{y}^*$. A positive slope of a curve, for example, means that increasing the value of this covariate helps to obtain a higher rating. We can see that the two models' $E(g_j|\boldsymbol{y})$ are almost the same based on the visual evidence.

Figure 2.3: The estimated expected value of $g_j|\boldsymbol{y}$ using Student-t model of credit rating data

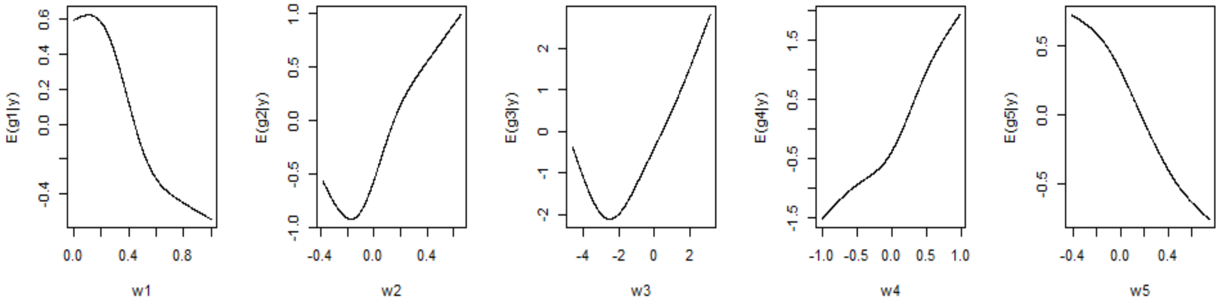


Figure 2.4: The estimated expected value of $g_j|\boldsymbol{y}$ for WDPMH of credit rating data

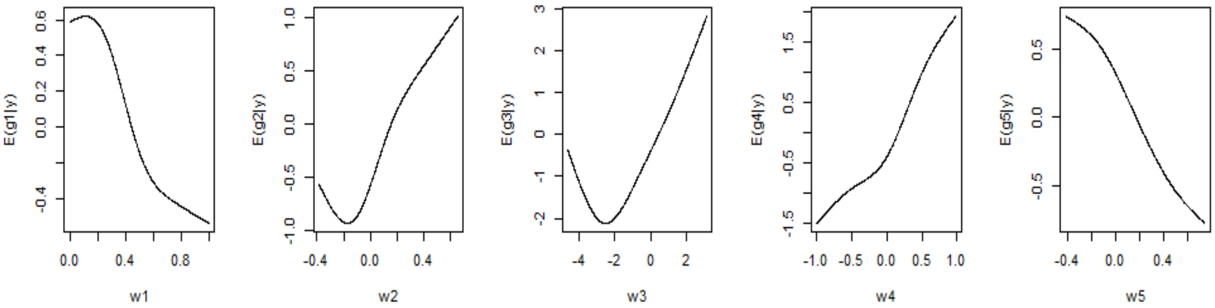
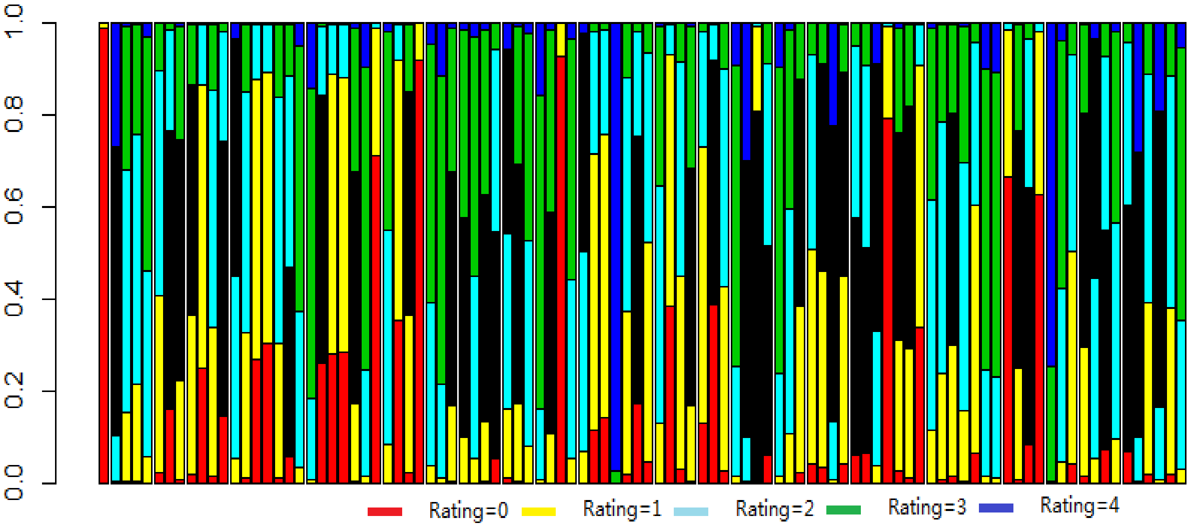


Figure 2.5 shows the how the credit rating probability distribution changes from observation to observation. This is the result from the 101st to the 200th observations based on WDPMH model, which produces the highest fitting accuracy.

Figure 2.5: observation-wise fitting result (the $101^{st}$-$200^{th}$ observations)



The x-axis is the index of observation and y-axis is the probability. There are 100 bars that have the same length of 1. Each bar is decomposed into five parts whose lengths represent the fitted values of $\Pr(y_i = j|\boldsymbol{\lambda}, \boldsymbol{\beta}, \boldsymbol{a})(j = 0, \cdots, 4)$. These five parts are distinguished by different colors. For the $i^{th}$ observation, the category that is associated with the largest $\Pr(y_i = j|\boldsymbol{\lambda}, \boldsymbol{\beta}, \boldsymbol{a})$ will be specified as the fitted credit rating, and the black area in the graph represents those largest $\Pr(y_i = j|\boldsymbol{\lambda}, \boldsymbol{\beta}, \boldsymbol{a})$ that induce mis-specifications. For example, the first bar (counting from left) represents the $101^{st}$ observation and we can see that the fitted value of $\Pr(y_i = 0|\boldsymbol{\lambda}, \boldsymbol{\beta}, \boldsymbol{a})$ is very close to 1 and it turns out the actual credit rating of this firm is also 0. In the second bar, the green part is covered by black because the fitted rating is 3 but the actual rating is not.

## 2.9 Summary/Future Research

In this Chapter, WDPM is applied in the construction of the error term parameters' priors for additive cubic spline regression. The important property of WDPM method is that it uses the predictive information provided by the covariates to construct the prior distribution. WDPM priors, as we have seen in the empirical study, contain potential advantage compared to the idea that proposing a prior ignoring the covariates' information. We can also see that, in the cubic spline regression, both the posterior distribution and the marginal likelihood of weighted DPM model can be derived conveniently by the straightforward extension of the algorithms designed for DPM model. Based on our simulation study with continuouse case, the overall performance of our EWDPM, WDPMG, WDPME, and WDPMH are similar and also is better than other models on Student-t. DPM, and WDPMD in terms of the marginal likelihood. Our WDPMH perform well in simulation study with ordinal case. Hence we would suggest using WDPMH for both continuous and ordinal cases to achieve model accuracy.

In future research, the function form of the weights and some possible priors of the hyper parameters in it should be taken into consideration for model comparison when applying our approach to other empirical data sets. It is also worthwhile to study on the optimal selection of $Q$ for WDPM in the future.

# Chapter 3

# Semiparametric Bayesian Approach for Stochastic Volatility Model

## 3.1 Introduction

The stochastic volatility (SV) model is popular in the field of finance, economics, applied mathematics, and statistics for a description of data from financial markets or modeling return of asset price such as exchange rate (Schotman and Mahieu, 1994; Taylor, 1994; Mahieu and Schotman, 1998), stock price (Danielsson, 1994; Sandmann and Koopman, 1998; Jensen, 2004) and option price (Bates, 1996; Baski et al, 1997; Carr and Sun, 2007) because the autoregressive process explains the inconstant variances well.

The return of asset price ($R_t$) at time $t$ depends on both price ($P_{t-1}$) at time $t-1$ and price ($P_t$) at time $t$. It is calculated as $R_t = 100 \times \log(P_t/P_{t-1})$. The log-volatility of returns

$(h_t)$ is modeled by AR(1). The SV model can then be written as

$$\begin{aligned} y_t &\overset{\triangle}{=} \log(R_t^2 + c) = h_t + a_t, \qquad t = 1, \ldots, n, \\ h_{t+1} &= \mu + \phi(h_t - \mu) + \sigma_\eta \eta_t, \end{aligned} \tag{3.1}$$

where $c$ is a small offset value making zero returns well defined in log function, $\phi$ is in $(-1, 1)$ to ensure stationary process, $a_t$ is called conditional return, and $\eta_t$ is the ranom white noise following $\mathrm{N}(0, 1)$.

The important part of SV model is the distribution of $a_t$. Applying Kalman filtering, Harvey et al (1994) finds that normal approximation is inaccurate; Kim et al (1998) manage to approximate the distribution of $a_t$ well (which is $\log\chi_1^2$) by a mixture of seven normals; Jacquier et al (2004) and Nakajima and Omori (2009) consider the marginal distribution of $\exp(a_t)$ to be Student-t distribution. Delatola and Griffin (2011) and Jensen and Maheu (2014) adopt Dirichlet process mixture (DPM) prior (Ferguson, 1973) to model $a_t$.

Furthermore, numerous ways of modifying (3.1) in seek of better model performance have been proposed. For example, Yu (2005) and Omori et al (2007) assume $a_t$ and $\eta_t$ to be correlated and apply leverage effect. Chib et al (2002) adopt jump effect in order to let the fitted conditional return be approximated by the proposed distribution more accurately. Eraker et al (2003) and Malik and Pitt (2009) combine both leverage and jump effect to construct model. They have provided a great effort in adding correction terms to SV model but have also adopted the implicit assumption that $a_t$'s follow the same marginal distribution because there is alway single prior for the parameters which specify the conditional distribution of $a_t$. This assumption may prevent people from managing to capture the complexity of real data such as stock prices and exchange rates, because stochastic volatility only deal with $h_t$. With more variable distribution of $a_t$ we may achieve greater model flexibility.

Hence, unlike these existing studies, we do not assume that the distribution of $a_t$ is constant of time. We allow each $a_t$ has its own distribution and can vary over time. We develop our approach under the nonparametric Bayesian frame so that our approach allows each $a_t$'s distribution to be automatically built and defined as unique via weighted DP mixture model (WDPM). The key idea of WDPM is to generate multiple (say $Q$) candidate priors for $a_t$ and let each observation to choose its favorite one using different weights. Thus different observation will have different marginal distribution of $a_t$ due to different weights.

The study of weight function for WDPM are quite limited. Dunson et al (2007) and Sun et al (2014) use Gaussian kernel type weight function forms which depend on distance between covariates. However this Gaussian kernel type weight functions have several limitations. It does not guarantee monotonicity and uniqueness which will be further explained in Section 3.2. It also cause considerable computation burden of MCMC sampling.

Motivated by the current limitations, we propose a new weight function for WDPM in Section 3.2. Compared to Gaussian kernel type weights, our newly proposed method greatly reduce the number of hyper-parameters in the weight function which have unknown form of posterior distribution: Most of the hyper-parameters follow Beta posterior if prior is properly chosen and there is only one parameter which requires Metropolis-Hastings (M-H). We prove that the ergodicity of our Markov chain in sampling can be ensured by imposing a bounded support for this parameter.

Our new weight function not only greatly reduces computation burden but also contains several good properties. For example: It can guarantee that one observation will always be more likely to choose a closer candidate than a further one; It does not require rescaling because the weights naturally sum to be 1; It converges to reasonable bounds as the parameters take extreme values.

Therefore the goal of this Chapter is to explain the new weight function under WDPM and further develop SV model under our WDPM. Our WDPM for SV model allows the distribution of $a_t$ is not constant of time. Construction of our weight function and its properties are illustrated in Section 3.2 and 3.3. To the best of our knowledge, no such weight function have been established for WDPM. We refer our weight function to the "probabilistic weight function" in the rest of the Chapter.

The remainder of this Chapter is organized as follows: Section 3.2 is the introduction of our proposed weight function and Section 3.3 illustrates the properties of this weight function. In Section 3.4 we apply weighted approach to SV model; Section 3.5 is the procedures of MCMC and in Section 3.6 we introduce the model evaluation method. Section 3.7 and 3.8 contain the simulation and empirical results. Section 3.9 briefly concludes.

## 3.2 Construction of our probabilistic weight function

### 3.2.1 Limitation of existing weight function

For modeling $a_t$, we suppose each observation randomly select its distribution from $Q$ candidates with weight $\boldsymbol{\pi}_t = (\pi_{1t}, \cdots, \pi_{Qt})$, where the sum of the entries in $\boldsymbol{\pi}_t$ is 1. Dunson et al (2007) propose the following way to model such multinomial weight:

$$\pi_{qt} = \frac{\gamma_q e^{-\psi \left\| x_t - x_q^c \right\|^2}}{\sum_{l=1}^{Q} \gamma_l e^{-\psi \left\| x_t - x_l^c \right\|^2}}, \quad q = 1, \ldots, Q, \tag{3.2}$$

where $x_t$ and $x_q^c$ are the explanatory covariate vector contained by observation $t$ and candidate $q$, $\|\cdot\|$ is the Euclidean distance, and $(\gamma_1, \cdots, \gamma_Q; \psi)$ are the hyper parameters.

The idea of (3.2) is to assign high weights to the candidates that are close to the observations in covariate information. However, this weight function has several limitations. The first one is that the weight of closer candidate is not guaranteed to be larger than the weight of further one because it depend on hyper parameters. The second is that although $(\gamma_1, \cdots, \gamma_Q; \psi)$ increase the flexibility in modeling weight (for example we can control the sparsity of weights by letting some elements of $\gamma_q$'s to be 0), we see redundancy of such parameterization because the weight value will be the same if all of the $\gamma_q$'s are multiplied by a positive constant. The third one is the computation challenge of MCMC sampling caused by the lack of close form of posterior of these hyper parameters. Sun et al (2014) find the slow mixing of MCMC caused by sequential update using M-H or Adaptive Rejection Metropolis Sampling (Gilks et al, 1995), and then propose the following weight function

$$\pi_{qt} = \frac{e^{-\psi_q \left\| x_t - x_q^c \right\|^2}}{\sum_{l=1}^{n} e^{-\psi_l \left\| x_t - x_l^c \right\|^2}}; \tag{3.3}$$

and found that this weight function improves MCMC efficiency. However, (3.3) can not ensure the monotonicity illustrated as the first limitation either.

### 3.2.2 Probabilistic weight function

For our weight function, let $(\pi_{(1)}, \cdots, \pi_{(Q)})$ denote the probabilities of the sorted candidates being selected. That is, $\pi_{(1)}$ is the probability of the nearest candidate being selected and $\pi_{(Q)}$ is the probability of the furthest candidate being selected. To achieve the monotonicity, $\pi_{(1)} \geq \pi_{(2)} \geq \cdots \geq \pi_{(Q)}$, we need to make the sorted probabilities decay faster than $(\pi_1, \cdots, \pi_Q)$ which are generated from "stick-breaking", where $\pi_1 = V_1$, $\pi_q = V_q \prod_{j=1}^{q-1}(1 - V_j)$ for $q = 2, \cdots, Q$ and $V_j \sim \text{Beta}(1, M)$ for $j = 1, \cdots, Q$. By introducing a deflate parameter $\gamma$, our weight function can be created as $\pi_{(1)} = 1 - \gamma(1 - V_1)$, $\pi_{(q)} = [1 - \gamma(1 - V_q)] \prod_{j=1}^{q-1}[\gamma(1 - V_j)]$. The deflate parameter $\gamma$ takes value between 0 and 1, and it is straight forward to verify that our weighted process to create our weight function becomes "stick-breaking" when $\gamma = 1$. Hence our weight process is the generalized version of stick breaking. The sorted probabilities decay faster than the ones generated by "stick-breaking". And also the length of stick available for further split, that is $\pi^c_{(q)} = 1 - \pi_{(1)} - \cdots - \pi_{(q)}$, also decay faster in our weight process.

*Theorem*-1 Suppose $(\pi_1, \cdots, \pi_Q)$ are generated as $\pi_1 = V_1$, $\pi_q = V_q \prod_{j=1}^{q-1}(1 - V_j)$ and $(\pi_{(1)}, \cdots, \pi_{(Q)})$ are generated as $\pi_{(1)} = 1 - \gamma(1 - V_1)$, $\pi_{(q)} = [1 - \gamma(1 - V_q)] \prod_{j=1}^{q-1}[\gamma(1 - V_j)]$, where $V_j \sim \text{Beta}(1, M)$ for $j = 1, \cdots, Q$. Then the following (i)-(iii) statements are true for any given $\gamma$: (i) $\pi_{(1)} \geq \pi_q$; (ii) $\text{E}(\pi_{(q)}) \leq (M+1)\gamma^{q-1}\text{E}(\pi_q)$ for $q = 2, \cdots, Q$; (iii) $\pi^c_{(q)} = \gamma^q \pi^c_q$ for $q = 2, \cdots, Q$.

The proof of Theorem 1 is straightforward: (i) is true because $1 - \gamma(1 - V_1)$ takes the minimum value $V_1$ when $\gamma = 1$; (ii) is true because, through the expectation calculation of the independent Beta distributed variables, we have $\text{E}(\pi_{(q)})/\text{E}(\pi_q) = (M + 1 - \gamma M)\gamma^{q-1}$ which takes the maximum value $(M+1)\gamma^{q-1}$ at $\gamma = 0$; (iii) is true because $\text{E}(\pi^c_q)$ is $\prod_{j=1}^{q}(1 - V_j)$ while $\text{E}(\pi^c_{(q)})$ is $\gamma^q \prod_{j=1}^{q}(1 - V_j)$.

Based on these three results, we propose to use $(\pi_{(1)}, \cdots, \pi_{(Q)})$ as the structure of our weights and use proper functions of sorted distance $(\left\| x_t - x^c_{(1)} \right\|, \cdots, \left\| x_t - x^c_{(Q)} \right\|)$ to replace $V_1, \cdots, V_Q$. Since we know that in stick breaking process, $V_j$s are between 0 and 1, and represent the proportion of cut of the current available stick which has not been cut in the previous cuts, we adopt conditional probabilities $p_{(1)t}, p_{(2)t}, \ldots, p_{(q)t}$ to replace $V_1, \cdots, V_q$ for observation $t$, where

$$p_{(q)t} = \frac{\left\| x_t - x^c_{(q)} \right\|^{-\psi}}{\sum_{l=q}^{Q} \left\| x_t - x^c_{(l)} \right\|^{-\psi}}.$$

This $p_{(q)t}$ $(q > 1)$ can be viewed as the probability that observation $t$ chooses its $q^{th}$ nearest candidate distribution given that its $q-1$ nearest candidates are not selected. $p_{(1)t}$ can be viewed as the probability that the nearest candidate is selected. Hence, we call our weight function as probabilistic weight function. Here $\psi > 0$ plays the similar role as those in (3.2) and (3.3) do.

We now summarize how to create our weight function in the following steps:

**Step-1** For observation $t$, we sort the candidates by their distance to this observation and re-label them from 1 to $Q$ in the ascending order. Thus we have the sorted distances $(\left\|x_t - x^c_{(1)}\right\|, \cdots, \left\|x_t - x^c_{(Q)}\right\|)$ for $t = 1, \cdots, n$;

**Step-2** Construct the conditional probability matrix $\left\{p_{(q)t}\right\}_{Q\times n}$ as:

$$p_{(q)t} = \frac{\left\|x_t - x^c_{(q)}\right\|^{-\psi}}{\sum_{l=q}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi}} \tag{3.4}$$

**Step-3** Use the conditional probability matrix to calculate the weights:

$$\begin{aligned}
\pi_{(1)t} &= 1 - \gamma_t(1 - p_{(1)t}),\\
\pi_{(2)t} &= \gamma_t(1 - p_{(1)t})[1 - \gamma_t(1 - p_{(2)t})],\\
\pi_{(3)t} &= \gamma_t^2(1 - p_{(1)t})(1 - p_{(2)t})[1 - \gamma_t(1 - p_{(3)t})],\\
&\vdots\\
\pi_{(Q)t} &= \gamma_t^{Q-1}[\prod_{l=1}^{Q-1}(1 - p_{(l)t})][1 - \gamma_t(1 - p_{(Q)t})],
\end{aligned} \tag{3.5}$$

We set $\gamma_t \in [0, 1]$ for every $t$ and $\psi \in [0, \infty]$.

## 3.3 Properties of the probabilistic weight function

The weights defined by (3.5) contain the following properties:

*Property 1:* $\pi_{(q)t} \geq 0$ for every $q$ and $t$.

*Property 2:* $\sum_{q=1}^{Q} \pi_{(q)t} = 1$ for every $t$.

*Property 3:* Denoting $(\pi_{(1)t}, \cdots, \pi_{(Q)t})$ as $\boldsymbol{\pi}_t$, we have: $\boldsymbol{\pi}_t \to (\frac{\left\|x_t - x^c_{(1)}\right\|^{-\psi}}{\sum_{l=1}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi}}, \cdots, \frac{\left\|x_t - x^c_{(Q)}\right\|^{-\psi}}{\sum_{l=1}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi}})$ as $\gamma_t \to 1$, and $\pi_t \to (1, 0, \cdots, 0)$ as $\gamma_t \to 0$ for $\forall \psi \in [0, \infty)$.

*Property 4:* $\boldsymbol{\pi}_t \to (1, 0, \cdots, 0)$ as $\psi \to \infty$ for $\forall \gamma_t \in [0, 1]$.

*Property 5:* If $\gamma_t > 0$, $Q \geq 2$, $\psi < \infty$, and $\left\|x_t - x^c_{(q)}\right\|$ is not the same for all $q$, then for a given $\gamma_t$ (or $\psi$), $\boldsymbol{\pi}_t$ is uniquely determined by $\psi$ (or $\gamma_t$).

*Property 6:* $\pi_{(1)t} \geq \pi_{(2)t} \geq \cdots \geq \pi_{(Q)t}$ is always true for every $t$.

The validation of property-1 is straightforward and this guarantees the non-negativity of the weights. The proof of property-2 is also trivial because $p_{(Q)t} = \frac{\left\|x_t - x^c_{(Q)}\right\|^{-\psi}}{\left\|x_t - x^c_{(Q)}\right\|^{-\psi}} = 1$, and this property means that we can directly use the weights as probabilities without re-scaling them. The weight functions (Dunson et al, 2007; Sun et al, 2014) based on Gaussian Kernel do not satisfy this property.

Property-3 is apparent because $\pi_t(\gamma_t, \psi)$ is a continuous function of $\gamma_t$ and the limits are equivalent to $\pi_t(1, \psi)$ and $\pi_t(0, \psi)$. Based on this property-3, we can see that $\gamma_t$ controls the degree of concentration in $\boldsymbol{\pi}_t$ around the few nearest candidates. The highest degree of concentration (i.e. $\boldsymbol{\pi}_t = (1, 0, \cdots, 0)$) means that observation $t$ always choose its nearest candidate prior. A smaller value of $\gamma_t$ means higher degree of concentration. It should be noted that the vector $\boldsymbol{\gamma} = (\gamma_1, \cdots, \gamma_n)$ in (3.5) are essentially different from $\boldsymbol{\gamma}$ in (3.2). Dunson et al (2007) assign $\boldsymbol{\gamma}$ to each candidate, while we assign $\boldsymbol{\gamma}$ to each observation.

Property-4 can be validated by the fact that $p_{(l)t} \to 1$ as $\psi \to \infty$ for $\forall l \in \{1, \cdots, Q\}$. This property means that the concentration is also controlled by $\psi$ and a larger value of $\psi$ means higher degree of concentration.

Property-5 implies the conditional uniqueness in our definition of weights. The condition that $\left\| x_t - x^c_{(q)} \right\|$ is not the same for all $q$ can always be satisfied because the distance is continuous random variable. This conditional uniqueness property is not satisfied by (3.2) as illustrated in the first limitation. The proof of property-5 is provided in Appendix B.

The monotonicity stated in property-6 ensures that for a single observation its nearer candidate is always more likely to be chosen than its farther candidate is. This property solves the first issue mentioned above and its proof is provided in Appendix C.

Besides these six properties, our weight function is able to reduce the computation burden caused by sampling hyper parameters because of two reasons. One is that we do not need to re-scale the weights in updating $\psi$. A much more important reason is that each element in $\boldsymbol{\gamma}$ in our function is proportional to a random variable whose full conditional distribution is Beta distribution if priors are properly chosen.

## 3.4 Probabilistic WDPM for Stochastic Volatility Model

### 3.4.1 Explanatory Covariates

As introduced in Section 3.2, we need to specify covariate values to construct weights. We use stock prices themselves with time lags as explanatory covariates. Stock price autocorrelation has been studied since Cowles and Jones (1937) and Cootner (1964). Despite of the famous "Weak-form Efficiency" hypothesis, Jegadeesh and Titman (1995), Llorente et al (2002) and Anderson et al (2012) have found the existence of autocorrelation of individual stocks which is related to size of firm and trading volume. The evidence of autocorrelation in composite index is even more compelling according to Mech (1993) and Franses and van Dijk (2000). In the latter literature it is found that daily autocorrelation of Frankfort and Tokyo index remain significant for at least 5 lags.

Therefore, the covariate values of observation $y_t$ should be $x_t = (R_{t-p}, \cdots, R_{t-1})$ once $p$ is specified. We propose to select candidate locations from $\{-3, 0, 3\}^p$ instead of random selection. We can sort the elements in $\{-3, 0, 3\}^p$ by their average distance to the observations in ascending order and use the first $Q$ elements. Using this method, both observations' and candidates' locations are fixed once $y$ is given. The advantage in model performance of this method over random picking is displayed in simulation study. Besides, it makes business sense. For example, letting $p = 3$, we know that $(-3, -3, -3)$ is one element in $\{-3, 0, 3\}^p$ and it represents the stock has been decreasing drastically in the past three days.

### 3.4.2 Weighted Dirichlet Process Mixture

Let $W$ be the probability of producing zero return. The unknown distribution $\mathrm{G}(\cdot) \sim \mathrm{DP}(\zeta, G_0)$, where $G_0$ is the base measure and $\zeta$ is the concentration parameter. Let $\mu_t$ be the conditional mean of $a_t$. The distribution of $a_t$ can be expressed as the follows:

$$p(a_t) = W\mathrm{N}(a_t|\mathrm{log}c, \sigma_0^2) + (1-W)\int \mathrm{N}(a_t|\mu_t, \sigma_1^2)\mathrm{G}(\mu_t)d\mu_t \tag{3.6}$$

where $\sigma_0^2$ and $\sigma_1^2$ are the variances in zero and non-zero components. We can see the density of $a_t$ is semiparametric because it is the mixture of a normal (related to zero return) and DPM (related to non-zero return). To ensure efficient MCMC draws of $W$, we set $\mathrm{log}c = -20$ and $\sigma_0^2 = 0.05$. As $\zeta \to \infty$, DPM falls back to parametric model assuming $\int \mathrm{N}(a_t|\mu_t, \sigma_1^2)\mathrm{G}_0(\mu_t)d\mu_t$ be the marginal distribution of $a_t$. As $\zeta$ goes to 0, $\mathrm{G}(\cdot)$ converge to a point mass.

Unlike DPM which consider single prior $G(\cdot)$ for all of the observations, our Weighted approach (WDPM) assign the $Q$ candidate priors $G^c_q(\cdot)$ with weight $\pi_{qt}$, that is,

$$\begin{aligned} \Pr(\mathrm{G}_t(\cdot) \equiv \mathrm{G}^c_q(\cdot)) &\propto \ \pi_{qt} \quad q = 1, \cdots, Q;\ t = 1, \cdots, n \\ \mathrm{G}^c_q(\cdot) &\overset{i.i.d}{\sim}\ \mathrm{DP}(\zeta, G_0) \end{aligned} \tag{3.7}$$

We define a configuration vector $\boldsymbol{Z}$ which allocates the $n$ observations to the $Q$ candidate priors, that is, $Z_t = q$ means $\mathrm{G}_t(\cdot) \equiv \mathrm{G}^c_q(\cdot)$. Using this vector, we can generalize P$\acute{o}$lya urn result (Blackwell and MacQueen, 1973) to WDPM. WDPM is more flexible than DPM because the latter one is the special case of the former one where the elements of $\boldsymbol{Z}$ take the same value.

WDPM can be viewed as the balance between parametric model and DPM. Using parametric prior, $\boldsymbol{\mu} \overset{\triangle}{=} (\mu_1, \cdots, \mu_n)$ will simply have $n$ unique values at each iteration of MCMC sampling. Using DPM, especially when $\zeta$ is small, the unique values in $\boldsymbol{\mu}$ will be way lower than $n$. However, using WDPM, the number of unique values may be between the two extremes. We can show that the prior mean of unique values in $(\mu_1, \cdots, \mu_n)$ of WDPM is larger than that of DPM.

*Theorem*-2 Let $kz$ be the total number of clusters in $Z$, and $(n_1, \cdots, n_{kz})$ be the numbers of observations that fall into each cluster. It is always true that:

$$\sum_{j=1}^{kz} E(k_j|n_1, \cdots, n_{kz}, \zeta) \geq E(k|n, \zeta) \tag{3.8}$$

where $(k_1, \cdots, k_{kz})$ are the numbers of unique values of $\boldsymbol{\mu}$ in each cluster and $k$ is the number of unique values of $\boldsymbol{\mu}$ if all the observations are in the same cluster.

Therefore, WDPM prior allows more flexibility than DPM prior does. If data is in favor of small number of unique values in $\boldsymbol{\theta}$, we can simplify WDPM to DPM by letting $Q = 1$. If data is in favor of a relatively large number of unique values in $\boldsymbol{\theta}$ (but not as large as sample size $n$), WDPM becomes the balance between DPM (small number of unique values) and parametric model ($n$ unique values). This allows us more choices to seek an accurate model.

### 3.4.3 SV Model Specification

To simplify the model, we can treat $h_t - \mu$ in (3.1) as the new log-volatility $h_t$ and merge $\mu$ into $a_t$. Therefore, SV model based on WDPM can be written as:

$$\begin{aligned}
y_t &= h_t + a_t \qquad 1 \leq t \leq n, \\
h_t &= \phi h_{t-1} + \sigma_\eta \eta_t, \\
a_t|\mu_t &\sim W\mathrm{N}(\log c, \sigma_0^2) + (1-W)\mathrm{N}(\mu_t, \sigma_1^2), \\
\mu_t &\sim \mathrm{G}_t, \\
\Pr(\mathrm{G}_t \equiv \mathrm{G}_q^c) &= \pi_{qt}, \\
\mathrm{G}_q^c &\sim \mathrm{DP}[\zeta, \mathrm{N}(\mu_0, \sigma_2^2)] \quad 1 \leq q \leq Q.
\end{aligned} \tag{3.9}$$

The prior of $\phi$ is $\mathrm{N}(\phi|0, 10)$ truncated to $(-1, 1)$. To solve identifiability issue, we impose the constraint $\frac{\sigma_1^2}{\alpha} = \frac{\sigma_2^2}{1-\alpha} = \sigma_z^2$ and sample $\sigma_z^2$ with flat prior. As suggested by Delatola and Griffin (2011), we set $\alpha = 0.05$. The prior for $\zeta$ is Gamma(1,0.2) and the prior for $\sigma_\eta^2$ is inverse Gamma(2.5,0.025).

## 3.5 MCMC Algorithm

Define $\Theta_0 \triangleq (\phi, \mu, \sigma_\eta^2)$, $\Theta_1 \triangleq (\mu_0, \sigma_z^2, \zeta, W)$ and $\Theta_2 \triangleq (\gamma, \psi)$. Let $\boldsymbol{S}$ be the vector that assigns the unique values in $\boldsymbol{\mu}$ to observations. With the initialized values, the MCMC sampling is performed based on the following framework:

Step-1: update $\boldsymbol{h}|\mathbf{y}, \boldsymbol{\mu}, \boldsymbol{Z}, \boldsymbol{S}, \Theta_0, \Theta_1, \Theta_2$

Step-2: update $\boldsymbol{S}|\mathbf{y}, \boldsymbol{h}, \boldsymbol{\mu}, \boldsymbol{Z}, \Theta_0, \Theta_1, \Theta_2$

Step-3: update $\boldsymbol{Z}|\mathbf{y}, \boldsymbol{h}, \boldsymbol{\mu}, \boldsymbol{S}, \Theta_0, \Theta_1, \Theta_2$

Step-4: update $\Theta_2|\mathbf{y}, \boldsymbol{h}, \boldsymbol{\mu}, \boldsymbol{Z}, \boldsymbol{S}, \Theta_0, \Theta_1$

Step-5: update $\boldsymbol{\mu}|\mathbf{y}, \boldsymbol{h}, \boldsymbol{Z}, \boldsymbol{S}, \Theta_0, \Theta_1, \Theta_2$

Step-6: update $\Theta_1|\mathbf{y}, \boldsymbol{h}, \boldsymbol{\mu}, \boldsymbol{Z}, \boldsymbol{S}, \Theta_0, \Theta_2$

Step-7: update $\Theta_0|\mathbf{y}, \boldsymbol{h}, \boldsymbol{\mu}, \boldsymbol{Z}, \boldsymbol{S}, \Theta_1, \Theta_2$

We display the joint distribution for WDPMP in Appendix D and the detailed procedure in Appendix E.

### 3.5.1 Sampling of $Z$

At step 3, the following $Theorem$-2 depicts how we sample $\boldsymbol{Z}$. The sampling $\boldsymbol{Z}$ is equivalent of sampling $\boldsymbol{C}$, which is the vector which allocates the $k$ unique values in $\boldsymbol{S}$ to the $Q$ candidate priors, that is, $C_j = q$ means that all of the observations who share the $j_{th}$ unique value of $\boldsymbol{S}$ will select the $q_{th}$ candidate prior.

We have derived the posterior distribution of $\boldsymbol{C}$. This theorem is not only applicable to our weights but also other weights. Hence this result is the general algorithm of sampling $\boldsymbol{Z}$ for WDPM with any form of weights:

*Theorem*-3 Suppose the probability that the $i_{th}$ $(1 \leq i \leq n)$ observation select the $q_{th}$ $(1 \leq q \leq Q)$ candidate prior is proportional to the constructed weight $w_{iq}$. If the likelihood of observations $f(\mathbf{y}|\Theta, \boldsymbol{S})$ is not conditional on $\boldsymbol{C}$, then the posterior of $C_j$ $(1 \leq j \leq k)$ is given as:

$$p(C_j = q|C_{(-j)}, \boldsymbol{S}, \mathbf{y}, \Theta) \propto \frac{\Gamma[(\zeta + n_{q(-j)}]}{\Gamma[\zeta + n_{q(-j)} + n_{(j)}]} \prod_{i=1}^{n_{(j)}} w_{iq} \tag{3.10}$$

where $n_{q(-j)}$ is the number of observations which select the $q_{th}$ candidate and are not assigned to the $j_{th}$ unique value in $\boldsymbol{S}$ and $n_{(j)}$ is the number of observations that are assigned to the $j_{th}$ unique value.

### 3.5.2 Sampling $\psi$

At step 4, we need to apply M-H to sample $\psi$ and show the ergodicity. In MCMC sampling, there is probability that the draws of $\psi$ be increasing to arbitrarily large values if there is no bound imposed on $\psi$ and the initial value of $\psi$ is not properly chosen. The following theorem suggests us to apply a bounded $\psi$ in the MCMC sampling in order to have uniform ergodicity.

*Theorem*-4 (Uniform Ergodicity) Let $q(\psi'|\psi)$ be the M-H proposal of $\psi$ and the corresponding $n$-step Markov transition kernel be $p^n(\psi, A) = \Pr(\psi_{n+j} \in A|\psi_j = \psi)$ which is constant of $j$. Let $f(\pi, \Theta_{-\psi})$ be the joint density where $\Theta_{-\psi}$ is all of the parameters except $\psi$ involved in MCMC sampling. By defining $\int \pi(dx)P(x, A)$ as $\pi P(A)$, we have that $\pi P = \pi$. Suppose the following conditions are satisfied:

(a) The support of the prior of $\psi$ (i.e. $\pi_\Psi(\psi)$) is $[0, M_\psi]$ where $M_\psi > 0$.

(b) $q(\psi'|\psi)$ is continuous function of $\psi'$ and $q(\psi'|\psi) > 0$ for $\forall \psi, \psi' \in [0, M_\psi]$.

(c) $\pi_\Psi(\psi) > 0$ is continuous and bounded in its support.

(d) $f(\pi, \Theta_{-\psi})$ is bounded and positive in the support of $\psi$ and $\Theta_{-\psi}$.

Then there exists a function $V(\psi) : [0, M_\psi] \rightarrow R$ and $0 < t < 1$ such that for $\pi$-almost every $\psi \in [0, M_\psi]$:

$$\|p^n(\psi, \cdot) - \pi(\cdot)\|_{TV} \leq V(\psi)t^n \tag{3.11}$$

and $\sup_\psi(V(\psi)) < \infty$, where $\|p^n(\psi, \cdot) - \pi(\cdot)\|_{TV}$ is defined to be $\sup_{A\in\mathcal{F}}|p^n(\psi, A) - \pi(A)|$.

This is also called $P$ is uniformly ergodic. The proof of $Theorem$-4 is in Appendix F. In this theorem, condition (b)-(d) are usually satisfied, so the most important regulation is the bounded support of $\psi$. Under these conditions, we are ensured that the $n$-step transition kernel will converge to target density so far as the initial value is within the support of $\psi$.

## 3.6 Model Evaluation

For model evaluation, we adopt the one-step-ahead log predictive score (LPS), which has been used in many literature such as Jensen and Maheu (2014) and Virbickaite et al (2014). The derivation of LPS requires the following five steps:

Step 1: Derive the predictive densities of $y_t - h_t$ (one for each candidate) through MCMC. The $q_{th}$ one is:

$$\begin{aligned}\hat{f}_q(\cdot) = \frac{1}{N_{max} - N_{bn}} \sum_{it=N_{bn}}^{N_{max}} \{\, & W_{(it)}\mathrm{N}(\cdot|\log c, \sigma_0^2) + (1 - W_{(it)})[\sum_{j=1}^{k_{q(it)}} \frac{n_{q_j(it)}}{\zeta + n_{q(it)}}\mathrm{N}(\cdot|\mu^*_{q_j(it)}, \alpha\sigma^2_{z(it)}) \\ & + \frac{\zeta}{\zeta + n_{q(it)}}\mathrm{N}(\cdot, |\mu_{0(it)}, \sigma^2_{z(it)})]\,\}\,,\end{aligned} \tag{3.12}$$

where $N_{max}$ and $N_{bn}$ are the number of iterations and burn-in number; $n_q$ is the number of observations which select the $q_{th}$ candidate prior and are labeled as non-zero return; Among these observations, $n_{q_j}$ is the number of observations which belong to the $q_j$-th unique value in $\boldsymbol{\mu}$ (i.e. $\mu^*_{q_j}$); The unique values other than log$c$ in $\boldsymbol{\mu}$ whose observations belong to the $q_{th}$ candidate prior is denoted as $(\mu^*_{q_1}, \cdots, \mu^*_{q_{k_q}})$ and $(it)$ means the value of $it$-th iteration of MCMC sampling.

Step 2: Use the posterior mean $\hat{\phi}$ and $\hat{\sigma^2_\eta}$ after burn-in as the estimation of $\phi$ and $\sigma^2_\eta$, and sample $h$ by $h_1 \sim N(0, \hat{\sigma^2_\eta}/(1-\hat{\phi}^2))$,$h_{t+1}|h_t \sim N(\hat{\phi}h_t, \hat{\sigma^2_\eta})$ for $t = 1, \cdots, n-1$. We can repeat this $B$ times to get $(\boldsymbol{h}^1, \cdots, \boldsymbol{h}^B)$.

Step 3: Assign weights to the $Q$ predictive densities for each observation:

$$\hat{f}(y_t) = \sum_{q=1}^{Q} \Pr(Z_t = q|\hat{\psi}) \frac{1}{B} \sum_{b=1}^{B} \hat{f}_q(y_t - h_t^b) \tag{3.13}$$

where we integrate weights over $\gamma_t$ to get $\Pr(Z_t = q|\hat{\psi})$ because $\gamma_t$ is contained by new observation and not estimable in real prediction:

$$\begin{aligned}
\Pr(Z_t = q|\hat{\psi}) &= \int_0^1 \pi_t(\gamma_t, \hat{\psi}) d\gamma_t \\
&= \int_0^1 \gamma_t^{q-1}[\prod_{l=0}^{q-1}(1-\hat{p}_{(l)t})][1-\gamma_t(1-\hat{p}_{(q)t})]d\gamma_t \\
&= \frac{\prod_{l=0}^{q-1}(1-\hat{p}_{(l)t})}{q} - \frac{[\prod_{l=0}^{q-1}(1-\hat{p}_{(l)t})](1-\hat{p}_{(q)t})}{q+1}
\end{aligned} \tag{3.14}$$

where $q_t$ is the $q_{th}$ nearest candidate prior of observation $t$.

Step 4: Finally, we derive LPS is $-\frac{1}{n}\sum_{t=1}^{n} \text{log}\hat{f}(y_t)$.

## 3.7 Simulation

We conduct simulation to compare our WDPM with other approaches:

- SV_PM: parametric model proposed in Kim et al (1998)
- SV_DPM: DPM model proposed in Delatola and Griffin (2011)
- SV_EWDPM: weighted DPM model using the weight function based on (3.3) assuming $\psi_1 = \cdots = \psi_Q$
- SV_WDPMH: weighted DPM model using the weight function based on (3.3) the one assuming half-cauchy priors for $(\psi_1, \cdots, \psi_Q)$
- SV_WDPMP: our probabilistic weighted DPM with fixed candidate locations
- SV_WDPMPr: our probabilistic weighted DPM with randomized candidate locations

To reduce inefficiency of MCMC draws, we disregard the first $50,000$ draws and keep one of every 10 draws of the following $50,000$ draws to derive parameter inferences and calculate LPS. We value computation efficiency because we have large number of model fittings in both simulation and empirical study. Fitting one sample of $n = 500$ simulated by parametric model by DPM in Intel Xeon E5-2687 3.10GHz CPU, the computation time of R, matlab and c++ (in RcppArmadillo) are $19,377$; 2045 and 381 seconds respectively. Therefore, we adopt c++ code in our research. We conducted $13,800$ model fittings in simulation study and 404 model fittings in empirical study. Each model fitting include $100,000$ iterations of MCMC sampling and $10,000$ iterations of LPS calculation. We integrated an R package by c++ code to enable parallel computation in multi-cores CPUs, and the total CPU time cost by these fittings was 69 hours.

### 3.7.1 Simulation setting

Simulated stock returns $(R_1, \cdots, R_n)$ based on WDPM model can be generated by the following steps:

**Step 1** Apply "stick-breaking"(Sethuraman, 1994) to generate $Q$ candidate measures $G_1^c(\cdot), \cdots, G_Q^c(\cdot)$ from $\mathrm{DP}[\zeta, \mathrm{N}(\mu_0, (1-\alpha)\sigma_z^2)]$.

**Step 2** Calculate weights $\boldsymbol{\pi}_t$ for each observation using weight function, hyper parameters and covariate values given by observation $(x_t)$ and candidates $(x_1^c, \cdots, x_Q^c)$.

**Step 3** Each observation selects its basis measure $G_t(\cdot)$ from $G_1^c(\cdot), \cdots, G_Q^c(\cdot)$ based on $\boldsymbol{\pi}_t$.

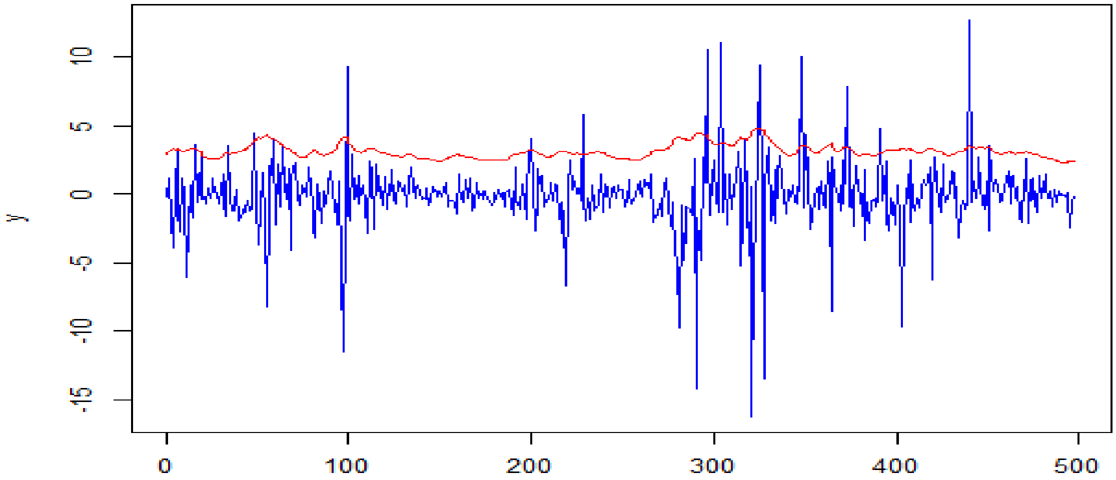
**Step 4** Generate $\mu_t$ from $G_t(\cdot)$ and sample $a_t$ from $\mathrm{N}(\mu_t, \alpha\sigma_z^2)$.

**Step 5** Sample log-volatility from $h_1 \sim N(0, \sigma_\eta^2/(1-\phi^2))$, $h_{t+1}|h_t \sim N(\phi h_t, \sigma_\eta^2)$ for $t = 1, \cdots, n-1$

**Step 6** $R_t$ is given by $e^{(h_t+a_t)/2}$. With probability $W$, we set $R_t$ to 0.

The above steps can be simplified for PM and DPM models. We set the values of the parameters used by true models as $\phi = 0.96$, $\sigma_\eta^2 = 0.10$ for all, $\mu = 0$ for PM only, $\sigma_z^2 = 3$, $\mu_0 = -0.2$, $\zeta = 2$, $W = 0.01$ for DPM and WDPM only, and $\psi = 2$ for WDPM only.

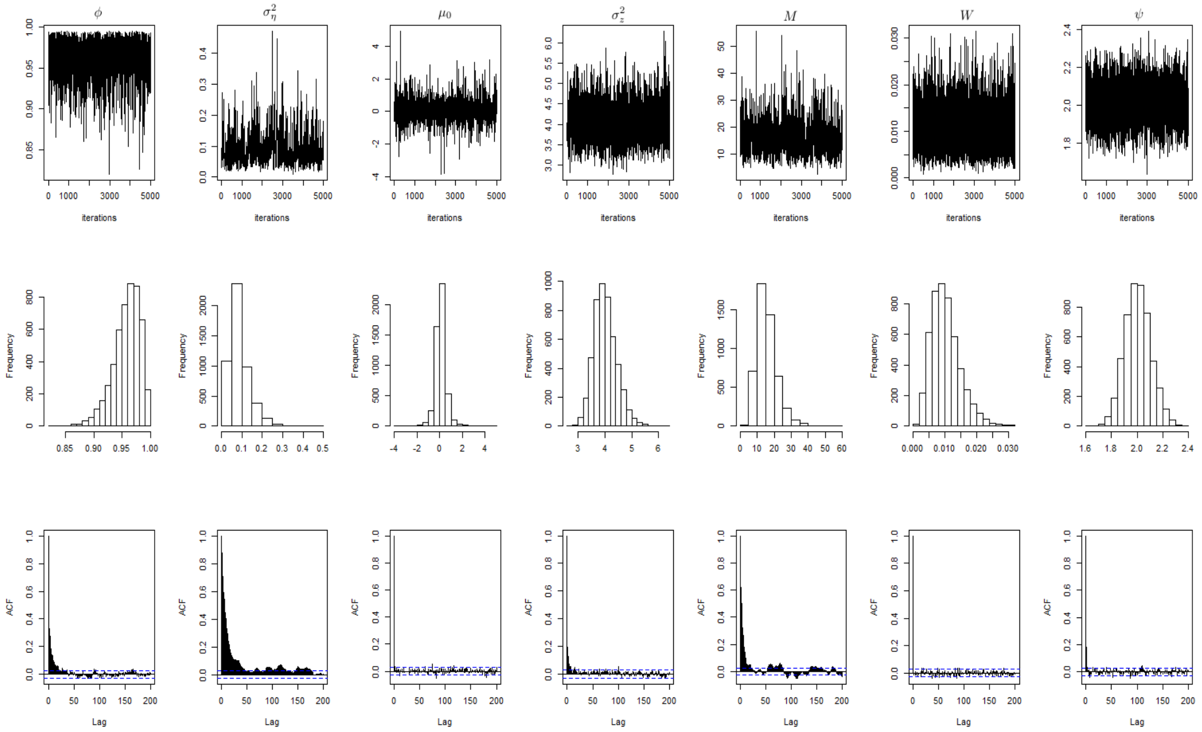
Figure 3.1: Stock returns simulated from SV_WDPMP with $p = 3$ $Q = 27$ with fitted volatility $exp(\hat{\boldsymbol{h}}/2)$ given by itself; The blue part is stocks prices, and red line is $exp(\hat{\boldsymbol{h}}/2)$ plus a constant (=2); We use this constant to make the two parts be separated and easy to view; The $y$ unit of stock prices is percentages, and "1" for fitted volatility.



In Figure 3.1 we display data simulated from SV_WDPMP with $p = 3$ $Q = 27$ using $n = 500$ and the fitted volatility. We can see that the fitted volatility manage to capture the inconstant variance of stock returns.

We also find that our MCMC sampling mix fast. Figure 3.2 is one randomly selected example of MCMC draws, histograms and sample autocorrelation function (acf) plots. In addition, we calculate inefficiency score, $1+2\sum_{l=1}^{K}\rho(l)$, where $\rho(l)$ is sample autocorrelation of lag $l$, of MCMC draws for each fitting.

Figure 3.2: Trace plots, histograms and acf's of MCMC draws; We disregard the first $50,000$ draws and keep one of every 10 draws of the following $50,000$ draws to be displayed in the trace plot; The plots are produced by a randomly selected fitting result of WDPMH with $p=3$, $Q=27$ fitted by WDPMP with $p=3$, $Q=27$ in simulation study.



Our simulation consist three simulation studies. In Chapter 2, we simulate data using PM, DPM, WDPMP with $p=2$ and $p=3$ as the true model respectively. For each case, 200 data sets are generated with sample size $n=500$. Each data set is fitted by six approaches including PM, DPM, and WDPMP with $p$ ranging from 1 to 4. In this Chapter, we generate data from five models including WDPMP with $p=3$ and four WDPMPr models using $p=2$ or $p=3$ and $Q=10$ or $Q=20$. Each model generates 200 data sets and each data set is fitted by the five models. In the third study, the true models are PM, WDPMP with $p = 3$, EWDPM with $p = 3$ and WDPMH with $p = 3$. We also generate 200 data sets from each model and each data set is fitted these four models.

### 3.7.2 Simulation study 1

In this study, we focus on comparing WDPMP to DPM and PM. For each true model, we calculated LPS using different methods. The results are displayed in Table 3.1 which contains the average LPS value (the higher the better) based on 200 fittings. Values in the parenthesis is $p$-value of Wilcoxon signed-rank test. We find that LPS values produced by different repetitions of the same approach tend to follow normal distribution, but the difference of LPS between different approaches usually violate normal assumption based on Shapiro-Wilk test. Therefore we adopt Wilcoxon signed-rank test in each row of Table 3.1 to check the null hypothesis that "the average LPS of certain fitting model is equal to the average LPS produced by the best model". from this result, WDPMP with $p = 3$ is the best model except for the case where the true model is PM.

Table 3.1: Model comparison results for simulation part-1 with $n = 500$; Each true model generates 200 samples and each sample is fitted by six approaches; In each cell, the top part is the average LPS value and the bottom part is the $p$-value of Wilcoxon signed-rank test produced by the comparison to the best model of the same row (except for the best models themselves, where the bottom parts are labeled by "best model"); The four true models are SV_PM, SV_DPM, SV_WDPMP with $p = 2$, and SV_WDPMP with $p = 3$; The six fitting models are SV_PM, SV_DPM, SV_WDPMP with $p$ ranging from 1 to 4.

| Fit Model<br>True Model | SV_PM | SV_DPM | SV_WDPMP<br>$p = 1$ | SV_WDPMP<br>$p = 2$ | SV_WDPMP<br>$p = 3$ | SV_WDPMP<br>$p = 4$ |
|---|---|---|---|---|---|---|
| SV_PM | -2.2220<br>best model | -2.2508<br>(4.300e-14) | -2.2441<br>(1.279e-12) | -2.2367<br>(3.365e-11) | -2.2331<br>(6.689e-09) | -2.3873<br>(0) |
| SV_DPM | -2.0744<br>(0) | -1.9581<br>(4.402e-10) | -1.9539<br>(7.095e-07) | -1.9416<br>(0.3758) | -1.9402<br>best model | -2.1033<br>(0) |
| SV_WDPMP<br>$p = 2$ | -2.1976<br>(0) | -2.1333<br>(4.115e-16) | -2.1006<br>(8.626e-13) | -2.0559<br>(7.352e-06) | -2.0403<br>best model | -2.2863<br>(0) |
| SV_WDPMP<br>$p = 3$ | -2.1775<br>(0) | -2.1265<br>(0) | -2.1118<br>(9.791e-15) | -2.0932<br>(1.703e-10) | -2.0685<br>best model | -2.1935<br>(0) |

We also note that one can find that the LPS values produced by WDPMP with $p = 4$ are dubiously worse than all the other models under any of the true models. Through investigation, we find that this is actually not caused by model mis-specification but by the sample size. Using $n = 500$, WDPMP with $p = 4$ is still the worst even if the true model is itself. Interestingly, if we use $Q = 27$ for $p = 4$ instead of $Q = 81$, the model performance will be even better than that of $p = 3$, $Q = 27$. Furthermore, we investigate simulated data of $n = 2000$ and find WDPMP with $p = 4$ even performs better than WDPMP with $p = 3$ based on 200 fittings using WDPMP with $p = 3$ as the true model. This means that $Q = 81$ may be too large for $n = 500$. We do not include the detail of this investigation because similar results can be seen in real data case.

### 3.7.3 Simulation study 2

In this study, we are interested to see if the model flexibility can be retained when $Q$ and candidate locations are mis-specified. Recall that candidate location is a vector of length $p$. Using SV_WDPMPr, we let a candidate to randomly select each of its $p$ components from $[-5, 5]$ uniformly. Based on simulation study 1, SV_WDPMP with $p = 3$ is the best model in most cases, therefore we let this model further compete with four SV_WDPMPr using $p = 2$ or $p = 3$ and $Q = 10$ or $Q = 20$. The results are in Table 3.2.

Table 3.2: Model comparison results for simulation part-2 with $n = 500$; Each true model generates 200 samples and each sample is fitted by its true model and the other four approaches.

| Fit Model<br>True Model | SV_WDPMPr<br>$p = 2, Q = 10$ | SV_WDPMPr<br>$p = 2, Q = 20$ | SV_WDPMPr<br>$p = 3, Q = 10$ | SV_WDPMPr<br>$p = 3, Q = 20$ | SV_WDPMP<br>$p = 3$ |
|---|---|---|---|---|---|
| SV_WDPMPr<br>$p = 2, Q = 10$ | -2.0416<br>(1.103e-11) | -2.0194<br>(0.009826) | -2.0420<br>(3.868e-14) | -2.0327<br>(4.278e-07) | -2.0136<br>best model |
| SV_WDPMPr<br>$p = 2, Q = 20$ | -2.0669<br>(0) | -2.0441<br>best model | -2.0707<br>(0) | -2.0581<br>(2.131e-11) | -2.0460<br>(0.07584) |
| SV_WDPMPr<br>$p = 3, Q = 10$ | -2.1220<br>(3.850e-12) | -2.1089<br>(7.294e-08) | -2.1201<br>(1.777e-11) | -2.1138<br>(4.029e-08) | -2.0924<br>best model |
| SV_WDPMPr<br>$p = 3, Q = 20$ | -2.1132<br>(0) | -2.1062<br>(3.142e-13) | -2.1066<br>(2.813e-11) | -2.0951<br>(0.4693) | -2.0927<br>best model |
| SV_WDPMP<br>$p = 3$ | -2.0780<br>(2.246e-16) | -2.0627<br>(3.651e-13) | -2.0713<br>(2.679e-15) | -2.0600<br>(3.961e-12) | -2.0290<br>best model |

We can see that SV_WDPMP with $p = 3$ is still the best model in most cases. The only exception is that when the true model is SV_WDPMPr with $p = 2$ and $Q = 20$, the best model is true model. However even in this case, SV_WDPMP with $p = 3$ shows no significant difference compared to the best model based on signed-rank test.

### 3.7.4 Simulation study 3

Our simulation in the last part focuses on the comparison among different weight functions. Since $p = 3$, $Q = 27$ and fixed candidate locations is the best combination of weighted DPM for $n = 500$ based on simulation part-1 and part-2, we use this setting for WDPMP, EWDPM and WDPMH. The results are in Table 3.3.

Table 3.3: Model comparison results for simulation part-3 with $n = 500$; Each true model generates 200 samples and each sample is fitted by its true model and the other four approaches.

| Fit Model<br>True Model | SV_PM | SV_WDPMP<br>$p = 3$ | SV_EWDPM<br>$p = 3$ | SV_WDPMH<br>$p = 3$ |
|---|---|---|---|---|
| SV_PM | -2.2237<br>best model | -2.2334<br>(0.005915) | -2.2496<br>(3.780e-09) | -2.2673<br>(2.368e-11) |
| SV_WDPMP<br>$p = 3$ | -2.1936<br>(1.663e-13) | -2.0789<br>best model | -2.0983<br>(5.813e-10) | -2.1358<br>(3.361e-11) |
| SV_EWDPM<br>$p = 3$ | -2.1748<br>(0) | -1.9955<br>best model | -2.0465<br>(5.637e-16) | -2.0591<br>(1.086e-15) |
| SV_WDPMH<br>$p = 3$ | -2.2222<br>(0) | -2.1453<br>best model | -2.1607<br>(1.378e-11) | -2.1490<br>(0.1364) |

We find that SV_WDPMP is still significantly better than other models unless the true model is PM. This means that our approach contains extra model flexibility which is not displayed by either EWDPM or WDPMH.

### 3.7.5 Summary of Simulation studies

Using LPS, we can conclude that we do see the model flexibility of WDPMP compared to other models. It is also found that $Q$ appear to be more important than $p$ and locations of candidates in terms of seeking better LPS. We learn that $Q$ should not be too small or too large for a certain sample size in order to achieve the best LPS. We also learn that fixed candidate location as we proposed is better than randomized location. For $n = 500$, SV_WDPMP with $p = 3$ is the optimal model among all models we have considered, but it may not be the best one as sample size increases.

As for the parameter inference, we include the results of two examples of true model fitted by itself and one model mis-specification in Table 3.4. The results of each example are based on 200 simulated data sets. In three kinds of fittings A, B, C, we display the point estimates and 95% credit interval of the parameters. A is SV_WDPMP with $p = 3$ fitted by SV_WDPMP with $p = 3$; B is SV_WDPMPr with $p = 2, Q = 20$ fitted by SV_WDPMPr with $p = 2, Q = 20$; C is SV_WDPMH with $p = 3$ fitted by SV_WDPMP with $p = 3$. Each point estimate is an average value of the 200 posterior means, and each 95% credit interval is specified by the $6^{th}$ and $195^{th}$ smallest values of these 200 posterior means. We also calculate the inefficiency score $1 + 2\sum_{l=1}^{K} \rho(l)$ (where $\rho(l)$ is sample autocorrelation of lag $l$) for each parameter's MCMC draws along with its 95% credit interval. We find that estimates are more accurate when true model is fitted by itself, and that the inefficiency score of $\sigma_\eta^2$ and $\zeta$ are usually higher than that of the other parameters.

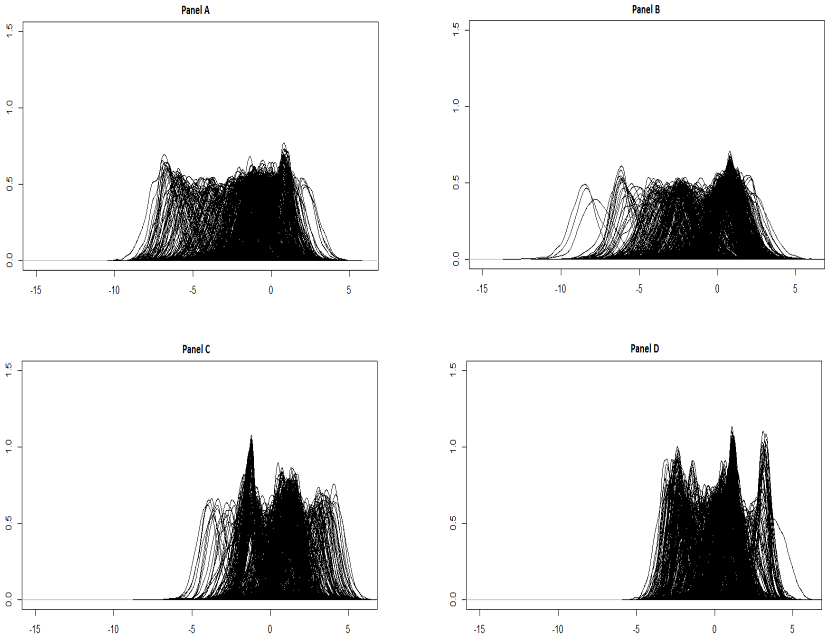
Table 3.4: Parameter estimates and inefficiency summary.

| | True Value | A | | B | | C | |
|---|---|---|---|---|---|---|---|
| | | Estimates | Inefficiency | Estimates | Inefficiency | Estimates | Inefficiency |
| $\sigma_\eta^2$ | 0.10 | 0.101<br>(0.035,0.173) | 21.66<br>(8.13,56.91) | 0.098<br>(0.041,0.192) | 20.19<br>(9.06,41.79) | 0.084<br>(0.028,0.149) | 26.33<br>(7.33,83.67) |
| $\phi$ | 0.96 | 0.951<br>(0.921,0.976) | 6.24<br>(2.71,18.55) | 0.954<br>(0.920,0.977) | 7.03<br>(3.01,16.25) | 0.944<br>(0.898,0.970) | 4.98<br>(2.25,13.98) |
| $\mu_0$ | -0.20 | -0.177<br>(-1.030,0.286) | 1.27<br>(1.00,1.84) | -0.154<br>(-0.989,0.751) | 1.29<br>(1.00,2.34) | -0.207<br>(-0.985,0.601) | 1.57<br>(1.00,3.12) |
| $\sigma_z^2$ | 3.0 | 3.09<br>(2.33,3.97) | 6.26<br>(2.69,11.07) | 3.07<br>(2.14,4.29) | 5.03<br>(2.45,10.48) | 3.31<br>(2.07,4.16) | 15.74<br>(5.15,61.81) |
| $M$ | 2.0 | 2.38<br>(0.83,5.55) | 14.01<br>(8.10,26.09) | 2.59<br>(0.46,6.83) | 10.79<br>(5.88,16.57) | 0.819<br>(0.256,1.849) | 17.42<br>(7.74,45.88) |
| $W$ | 0.01 | 0.010<br>(0.002,0.020) | 1.000<br>(1.000,1.000) | 0.010<br>(0.002,0.021) | 1.000<br>(1.000,1.000) | 0.010<br>(0.002,0.020) | 1.002<br>(1.000,1.000) |
| $\psi$ | 2.0 | 1.99<br>(1.87,2.05) | 12.33<br>(6.58,17.75) | 2.03<br>(1.98,2.11) | 10.95<br>(5.19,19.30) | 1.94<br>(1.88,2.10) | 1.33<br>(1.00,1.91) |

We have also checked the existence of multiple clusters in $(\mu_1, \mu_2, \cdots, \mu_n)$ based on their posterior draws since we are using DPM and weighted DPM. We display four examples in Figure 3.3. There are n (sample size $n = 500$ for simulation study) density curves in each of the four panels. Each curve is plotted based on $50,000$ MCMC draws of $\mu_t$. The x-axis is the range of $\mu_t$ and y-axis stands for the kernel density estimated value.

When data generated by PM is fitted by WDPMP model (Panel A), we see that the clusters may exist, but the separation among these clusters is vague. If the true model becomes DPM (Panel B), then the existence of multiple clusters look more apparent than true model PM case. But Panel C and D, where data generated from WDPMP models are

Figure 3.3: Posterior density curves of $\mu_t$ $(t = 1, \cdots, n)$; The four panels are simulation results: Panel A is SV_PM fitted by SV_WDPMP with $p = 3, Q = 27$; Panel B is SV_DPM fitted by SV_WDPMP with $p = 3, Q = 27$; Panel C is SV_WDPMP with $p = 3, Q = 27$ fitted by itself; Panel D is SV_WDPMPr with $p = 3, Q = 20$ fitted by SV_WDPMP with $p = 3, Q = 27$.



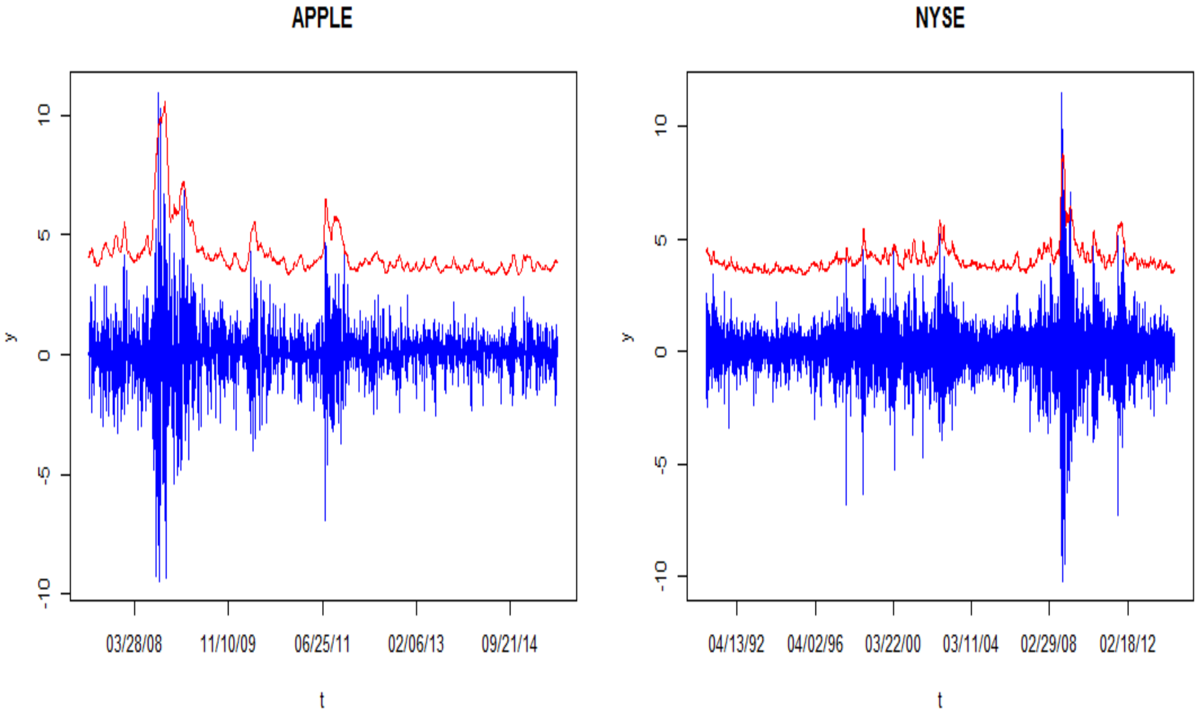
fitted by WDPMP models, display even more clear evidence of multiple clusters than what is displayed in Panel A or B. We also check data sets generated from other WDPM models, and find the similar results. We can conclude that WDPM data is more likely to display pattern of multiple clusters than DPM and PM data.

## 3.8 Application

In the empirical study, our data sets consist of 100 randomly selected NASDAQ individual stocks' daily prices; two companies (APPLE and EXXON) with the largest market capitalization in the world (according to Financial Times Global 500's first quarter report of 2015); three composite indexes (NYSE composite index, Standard & Poor's 500 and Deutscher

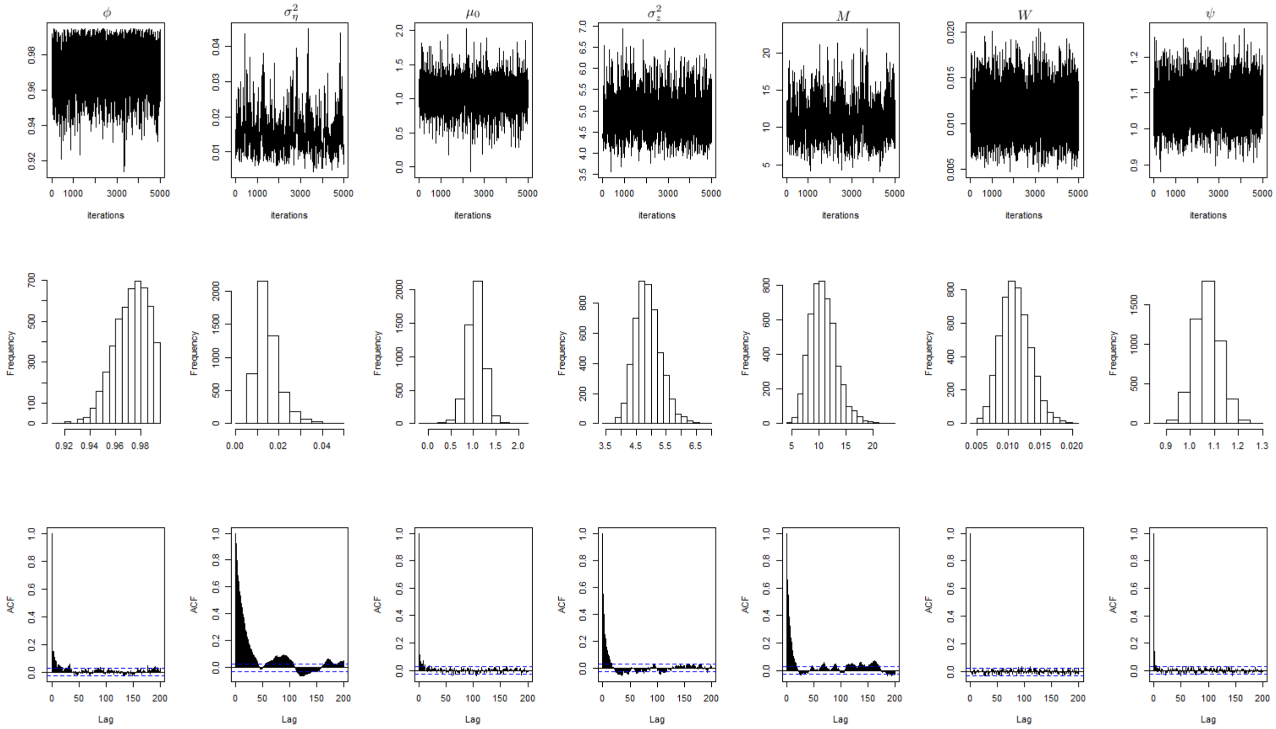
Aktienindex). We use $n = 2000$ for all of them except NYSE index. For NYSE, we set $n = 6000$. In Figure 3.4, we display two examples of empirical data sets with the fitted volatility. We can see that the fitted volatility manage to capture the inconstant variance of stock returns.

Figure 3.4: Stock returns with fitted volatility $exp(\hat{\boldsymbol{h}}/2)$ given by SV_WDPMP (with $p = 3$ $Q = 27$ for the APPLE data and $p = 6$ $Q = 200$ for NYSE). The blue part is stocks prices, and red line is $exp(\hat{\boldsymbol{h}}/2)$ plus a constant (=3). We use this constant to make the two parts be separated and easy to view. The $y$ unit of stock prices is percentages, and "1" for fitted volatility. Panel A is APPLE data, and B is NYSE data.



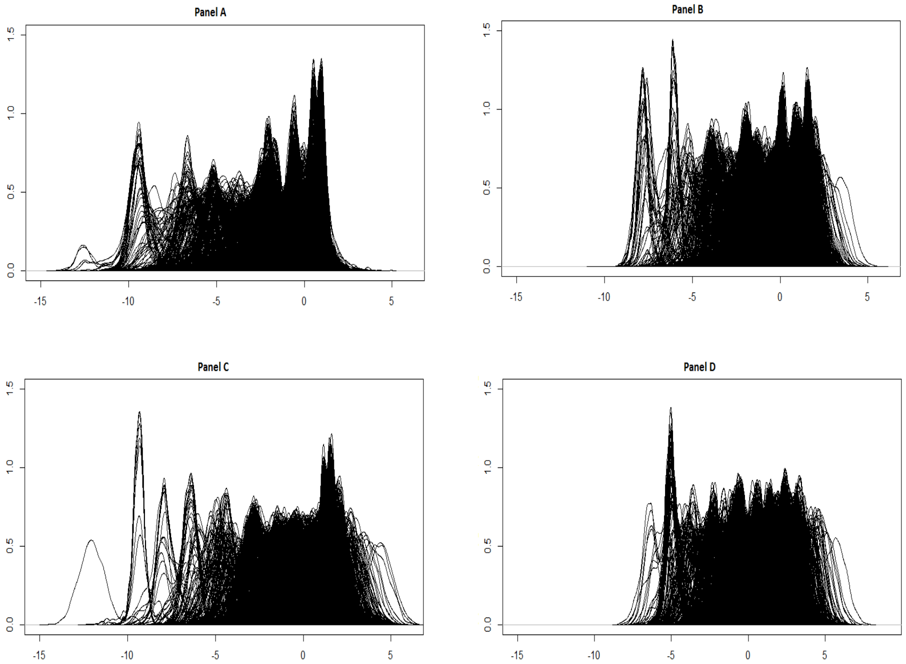
We find that our MCMC samplings for empirical study also mix fast. Figure 3.5 is one randomly selected example of MCMC draws, histograms and acf plots. Besides the visual evidence, we check inefficiency score of MCMC draws for each fitting.

Figure 3.5: Trace plots, histograms and acf's of MCMC draws; We disregard the first $50,000$ draws and keep one of every 10 draws of the following $50,000$ draws to be displayed in the trace plot; The plots are produced by a randomly selected NASDAQ data (MXWL) fitted by WDPMP with $p = 3$ and $Q = 27$.



In addition, we are interested to check the existence of multiple clusters in $(\mu_1, \cdots, \mu_n)$ for empirical data as well. In Figure 3.6, we display four examples the same way as we did in Figure 3.3. All of the four data sets (S&P 500, Exxon, Apple and PFBC) show clear evidence of multiple clusters (PFBC is one of the 100 NASDAQ individual stocks which will be checked in Section 3.8.1). Furthermore, there appear to be more clusters in empirical data compared to simulation study: each of these four empirical data sets contain at least 7 major clusters and we can see more than 10 clusters in the panel of PFBC.

Figure 3.6: Posterior density curves of $\mu_t$ $(t = 1, \cdots, n)$; The four panels are empirical results: Panel A is S&P500 data fitted by SV_EWDPM with $p = 5, Q = 40$; Panel B is Exxon data fitted by SV_WDPMP with $p = 5, Q = 80$; Panel C is Apple data fitted by SV_WDPMP with $p = 5, Q = 100$; Panel D is PFBC data fitted by SV_WDPMP with $p = 3, Q = 27$; Note that PFBC is one of the 100 NASDAQ individual stocks checked in Section (3.8.1).



## 3.8.1 NASDAQ Individual Stocks

We use the NASDAQ stock list which include 3023 stock names to randomly pick stocks. The 100 stocks are selected sequentially because each selected stock need to have available daily prices between 07/20/2007 and 06/30/2015 on Yahoo Finance website. At each step, we pick one stock among the unselected ones remain in the list with equal probabilities, and check the data availability to determine if the selected stock can be added into our data set. Such step was repeated 187 times until we have 100 stocks. Since SV_WDPMP with $p = 3$ performs well in simulation study, we let this model to compete with SV_PM and SV_DPM based on these 100 samples of real stocks. Table 3.5 and 3.6 provides the detail of how the three models perform in each of the stocks.

Table 3.5: Summary of model comparison among SV_PM, SV_DPM and SV_WDPMP with $p = 3$ using randomly selected 100 stocks in NASDAQ; This table display the results of first 50 stocks; The values in each cell is the LPS; The stock code is abbreviation of company name adopted by NASDAQ; The 100 stocks are selected sequentially because each selected stock need to have available daily prices between 07/20/2007 and 06/30/2015 on Yahoo finance website.

| Stock Code | PM | DPM | WDPMP | Stock Code | PM | DPM | WDPMP |
|---|---|---|---|---|---|---|---|
| SWIR | -2.264 | -2.235 | -2.207 | BRCD | -2.310 | -2.270 | -2.243 |
| BIIB | -2.235 | -2.242 | -2.213 | TFSL | -2.262 | -2.240 | -2.127 |
| FEIM | -2.443 | -2.326 | -2.299 | AIRT | -2.446 | -2.284 | -2.251 |
| CSPI | -2.704 | -2.362 | -2.291 | DIOD | -2.219 | -2.229 | -2.205 |
| CVCO | -2.252 | -2.263 | -2.246 | AMWD | -2.262 | -2.284 | -2.241 |
| FAST | -2.276 | -2.284 | -2.263 | PACW | -2.305 | -2.298 | -2.268 |
| BRLI | -2.233 | -2.227 | -2.195 | INTU | -2.285 | -2.292 | -2.279 |
| CDZI | -2.333 | -2.294 | -2.254 | FCTY | -3.114 | -2.363 | -2.314 |
| ISBC | -2.347 | -2.300 | -2.255 | PBIP | -3.011 | -2.531 | -2.451 |
| CBMX | -2.647 | -2.383 | -2.328 | BPOPN | -2.798 | -2.680 | -2.411 |
| EEI | -2.587 | -2.373 | -2.333 | CHY | -2.328 | -2.291 | -2.195 |
| WTBA | -2.335 | -2.320 | -2.294 | HEES | -2.269 | -2.280 | -2.241 |
| PFBC | -2.459 | -2.399 | -2.346 | BWINB | -2.258 | -2.256 | -2.222 |
| DARA | -2.861 | -2.376 | -2.319 | MXWL | -2.226 | -2.227 | -2.187 |
| QQXT | -2.403 | -2.333 | -2.329 | SIFY | -2.509 | -2.339 | -2.238 |
| SRCE | -2.282 | -2.284 | -2.247 | MGRC | -2.313 | -2.308 | -2.268 |
| SVBI | -2.864 | -2.538 | -2.372 | ODFL | -2.238 | -2.248 | -2.218 |
| THRM | -2.233 | -2.233 | -2.208 | SIFI | -2.839 | -2.589 | -2.450 |
| BECN | -2.270 | -2.272 | -2.244 | STMP | -2.283 | -2.286 | -2.254 |
| AEZS | -2.706 | -2.274 | -2.230 | DXPE | -2.260 | -2.271 | -2.249 |
| ISCA | -2.257 | -2.250 | -2.217 | ENTG | -2.347 | -2.307 | -2.277 |
| ISIS | -2.243 | -2.267 | -2.196 | PLCE | -2.237 | -2.241 | -2.218 |
| PCCC | -2.328 | -2.308 | -2.293 | GLNG | -2.291 | -2.308 | -2.270 |
| FSBK | -2.514 | -2.367 | -2.311 | HWKN | -2.332 | -2.312 | -2.288 |
| DYAX | -2.305 | -2.240 | -2.198 | BELFA | -2.994 | -2.552 | -2.481 |

In summary, SV_WDPMP with $p = 3$ produce better LPS than SV_PM does in all of the stocks, and there are only two stocks where SV_DPM is better than SV_WDPMP.

Table 3.6: Summary of model comparison among SV_PM, SV_DPM and SV_WDPMP with $p = 3$ using randomly selected 100 stocks in NASDAQ; This table display the results of last 50 stocks; The values in each cell is the LPS; The stock code is abbreviation of company name adopted by NASDAQ; The 100 stocks are selected sequentially because each selected stock need to have available daily prices between 07/20/2007 and 06/30/2015 on Yahoo finance website; The two stocks where DPM produces better LPS than WDPMP does are marked red.

| Stock Code | PM | DPM | WDPMP | Stock Code | PM | DPM | WDPMP |
|---|---|---|---|---|---|---|---|
| SYMC | -2.258 | -2.255 | -2.229 | SQNM | -2.304 | -2.259 | -2.205 |
| AMD | -2.298 | -2.227 | -2.200 | HFFC | -2.852 | -2.811 | -2.353 |
| MHGC | -2.427 | -2.364 | -2.293 | AKRX | -2.361 | -2.337 | -2.281 |
| XCRA | -2.357 | -2.293 | -2.266 | ILMN | -2.234 | -2.243 | -2.214 |
| AETI | -2.919 | -2.640 | -2.316 | CLDX | -2.312 | -2.296 | -2.235 |
| BASI | -3.088 | -2.444 | -2.410 | FNLC | -2.420 | -2.389 | -2.314 |
| NTAP | -2.244 | -2.251 | -2.231 | AMAT | -2.232 | -2.221 | -2.217 |
| CDNS | -2.283 | -2.244 | -2.228 | PRXI | -2.617 | -2.288 | -2.295 |
| HTBK | -2.431 | -2.401 | -2.320 | FCTY | -3.107 | -2.361 | -2.257 |
| CBLI | -2.474 | -2.317 | -2.254 | RELL | -2.447 | -2.403 | -2.293 |
| AXPW | -3.282 | -3.245 | -2.259 | FELE | -2.217 | -2.223 | -2.204 |
| CPSI | -2.238 | -2.237 | -2.209 | CPLP | -2.329 | -2.286 | -2.243 |
| CLNE | -2.239 | -2.245 | -2.201 | KOOL | -2.617 | -2.205 | -2.211 |
| WSFS | -2.287 | -2.295 | -2.270 | TESO | -2.256 | -2.249 | -2.228 |
| ARCI | -2.564 | -2.345 | -2.326 | LYTS | -2.261 | -2.228 | -2.203 |
| PLPC | -2.285 | -2.302 | -2.263 | VNDA | -2.405 | -2.331 | -2.327 |
| OCC | -2.614 | -2.291 | -2.234 | WDC | -2.251 | -2.251 | -2.236 |
| FORD | -2.508 | -2.219 | -2.163 | BMTC | -2.352 | -2.332 | -2.327 |
| FOX | -2.209 | -2.197 | -2.177 | CALD | -2.337 | -2.262 | -2.238 |
| WAVX | -2.621 | -2.233 | -2.208 | IRIX | -2.674 | -2.424 | -2.382 |
| PTC | -2.235 | -2.232 | -2.223 | AIQ | -2.336 | -2.279 | -2.247 |
| CRUS | -2.220 | -2.212 | -2.191 | SP | -2.246 | -2.241 | -2.225 |
| FLWS | -2.363 | -2.362 | -2.256 | TTGT | -2.396 | -2.344 | -2.306 |
| MTSN | -2.345 | -2.213 | -2.186 | CCRN | -2.251 | -2.212 | -2.181 |
| CMPR | -2.254 | -2.251 | -2.234 | MARK | -2.700 | -2.443 | -2.409 |

Table 3.7 summarizes the results we see in Table 3.5 and 3.6.

Table 3.7: Proportion of data sets that one model perform better than another based on the model comparison among SV_PM, SV_DPM and SV_WDPMP with $p = 3$ using randomly selected 100 stocks in NASDAQ.

| | |
|---|---|
| DPM better than PM | 77% |
| WDPMP better than PM | 100% |
| WDPMP better than DPM | 98% |

Furthermore, the highest inefficiency score produced by WDPMP among these 100 fittings is the inefficiency of $\sigma_z^2$ of AEZS, which is 96.3, while there are 43 stocks where DPM produce higher than 200 inefficiency score in one or multiple parameters' MCMC draws. This means the posterior draws of parameters in WDPMP models mix faster than those in DPM or parametric models, which indicates better performance of MCMC sampling of weighted models.

### 3.8.2 Big Companies and Composite Indexes

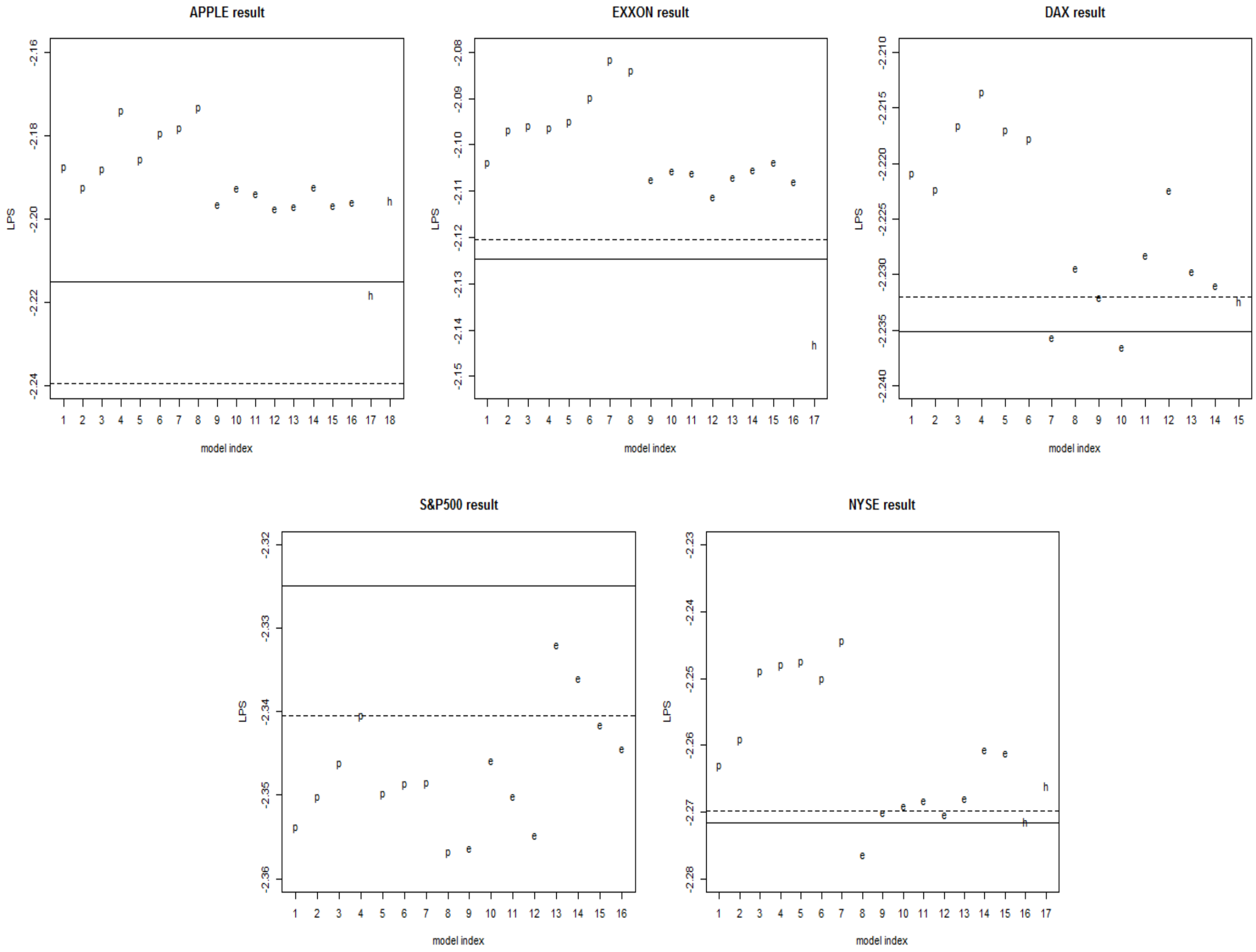
We compare WDPMP to EWDPM and WDPMH using different $p$ and$Q$ based on the five data sets. Note that these data have different time range and the sample size of NYSE index ($n = 6000$) is larger than the other four ($n = 2000$). Table 3.8 provides the detail of data range and LPS values. We learn from initial exploration of real data fitting that $p \leq 2$ (which means $Q \leq 9$) is worse than larger $Q$ choices for $n \geq 2000$ case, therefore we start from $p = 3$, $Q = 27$ for $n = 2000$ data. We further find that $p = 3$, $Q = 27$ is worse than models with larger $Q$ for $n = 6000$ case, therefore we start from $p = 4$, $Q = 81$ for $n = 6000$ data. In Table 3.8 we can see that the models using WDPMP are generally better than other models including PM and DPM except in the case of S&P500 data.

Table 3.8: Summary of model comparison between different weight function; Data range of EXXON is from 01/06/2000 to 12/17/2007, APPLE and S&P500 is from 08/06/2007 to 07/15/2015, DAX is from 02/09/2000 to 12/17/2007 and NYSE is from 09/13/1990 to 07/08/2014. In the Index columns: if one model produces higher than 100 inefficiency score in some parameter(s), it is labeled as '×"; if one model produce less than 100 inefficiency score in all of the parameters, its LPS value is plotted in Figure 3.7 and the numerical value of index is the model index used by Figure 3.7; The indexes of PM and DPM are left blank because they are used as benchmark lines in Figure 3.7.

| Model | APPLE | | EXXON | | DAX | | S&P500 | | Model | NYSE | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | LPS | Index | LPS | Index | LPS | Index | LPS | Index | | LPS | Index |
| SV_PM | -2.214982 | | -2.124732 | | -2.235122 | | -2.324854 | | SV_PM | -2.271581 | |
| SV_DPM | -2.239405 | | -2.120565 | | -2.232036 | | -2.340498 | | SV_DPM | -2.269792 | |
| SV_WDPMP<br>$p = 3$, $Q = 27$ | -2.187644 | 1 | -2.104094 | 1 | -2.224892 | × | -2.353863 | 1 | SV_WDPMP<br>$p = 4$, $Q = 81$ | -2.263140 | 1 |
| SV_WDPMP<br>$p = 4$, $Q = 40$ | -2.192531 | 2 | -2.096867 | 2 | -2.221011 | 1 | -2.350219 | 2 | SV_WDPMP<br>$p = 5$, $Q = 80$ | -2.261873 | × |
| SV_WDPMP<br>$p = 4$, $Q = 50$ | -2.188230 | 3 | -2.095943 | 3 | -2.222442 | 2 | -2.346304 | 3 | SV_WDPMP<br>$p = 5$, $Q = 100$ | -2.259251 | 2 |
| SV_WDPMP<br>$p = 4$, $Q = 81$ | -2.174099 | 4 | -2.096455 | 4 | -2.216675 | 3 | -2.340508 | 4 | SV_WDPMP<br>$p = 5$, $Q = 150$ | -2.249084 | 3 |
| SV_WDPMP<br>$p = 5$, $Q = 40$ | -2.185998 | 5 | -2.095069 | 5 | -2.206859 | × | -2.349826 | 5 | SV_WDPMP<br>$p = 5$, $Q = 200$ | -2.248067 | 4 |
| SV_WDPMP<br>$p = 5$, $Q = 50$ | -2.179777 | 6 | -2.089919 | 6 | -2.213659 | 4 | -2.348702 | 6 | SV_WDPMP<br>$p = 6$, $Q = 100$ | -2.247511 | 5 |
| SV_WDPMP<br>$p = 5$, $Q = 80$ | -2.178335 | 7 | -2.081823 | 7 | -2.217120 | 5 | -2.348542 | 7 | SV_WDPMP<br>$p = 6$, $Q = 150$ | -2.250222 | 6 |
| SV_WDPMP<br>$p = 5$, $Q = 100$ | -2.173400 | 8 | -2.084022 | 8 | -2.217886 | 6 | -2.356897 | 8 | SV_WDPMP<br>$p = 6$, $Q = 200$ | -2.244547 | 7 |
| SV_EWDPM<br>$p = 3$, $Q = 27$ | -2.196537 | 9 | -2.107450 | 9 | -2.235694 | 7 | -2.356329 | 9 | SV_EWDPM<br>$p = 4$, $Q = 81$ | -2.266359 | 8 |
| SV_EWDPM<br>$p = 4$, $Q = 40$ | -2.192622 | 10 | -2.105545 | 10 | -2.229470 | 8 | -2.345894 | 10 | SV_EWDPM<br>$p = 5$, $Q = 80$ | -2.260073 | 9 |
| SV_EWDPM<br>$p = 4$, $Q = 50$ | -2.193895 | 11 | -2.106103 | 11 | -2.232098 | 9 | -2.350097 | 11 | SV_EWDPM<br>$p = 5$, $Q = 100$ | -2.259185 | 10 |
| SV_EWDPM<br>$p = 4$, $Q = 81$ | -2.197677 | 12 | -2.111311 | 12 | -2.236556 | 10 | -2.354828 | 12 | SV_EWDPM<br>$p = 5$, $Q = 150$ | -2.258347 | 11 |
| SV_EWDPM<br>$p = 5$, $Q = 40$ | -2.197165 | 13 | -2.107103 | 13 | -2.228297 | 11 | -2.331984 | 13 | SV_EWDPM<br>$p = 5$, $Q = 200$ | -2.260351 | 12 |
| SV_EWDPM<br>$p = 5$, $Q = 50$ | -2.192334 | 14 | -2.105493 | 14 | -2.222434 | 12 | -2.335996 | 14 | SV_EWDPM<br>$p = 6$, $Q = 100$ | -2.257998 | 13 |
| SV_EWDPM<br>$p = 5$, $Q = 80$ | -2.196730 | 15 | -2.103664 | 15 | -2.229735 | 13 | -2.341514 | 15 | SV_EWDPM<br>$p = 6$, $Q = 150$ | -2.250683 | 14 |
| SV_EWDPM<br>$p = 5$, $Q = 100$ | -2.196003 | 16 | -2.108054 | 16 | -2.231009 | 14 | -2.344434 | 16 | SV_EWDPM<br>$p = 6$, $Q = 200$ | -2.251108 | 15 |
| SV_WDPMH<br>$p = 3$, $Q = 27$ | -2.221196 | × | -2.120838 | × | -2.231009 | 15 | -2.344434 | × | SV_WDPMH<br>$p = 4$, $Q = 81$ | -2.268358 | × |
| SV_WDPMH<br>$p = 4$, $Q = 40$ | -2.218191 | × | -2.142888 | 17 | -2.231002 | × | -2.335611 | × | SV_WDPMH<br>$p = 5$, $Q = 80$ | -2.259390 | × |
| SV_WDPMH<br>$p = 4$, $Q = 50$ | -2.218081 | 17 | -2.110234 | × | -2.225842 | × | -2.325789 | × | SV_WDPMH<br>$p = 5$, $Q = 100$ | -2.261457 | 16 |
| SV_WDPMH<br>$p = 4$, $Q = 81$ | -2.223953 | × | -2.150509 | × | -2.237982 | × | -2.327060 | × | SV_WDPMH<br>$p = 5$, $Q = 150$ | -2.255317 | × |
| SV_WDPMH<br>$p = 5$, $Q = 40$ | -2.195402 | 18 | -2.130149 | × | -2.227389 | × | -2.333147 | × | SV_WDPMH<br>$p = 5$, $Q = 200$ | -2.256094 | 17 |
| SV_WDPMH<br>$p = 5$, $Q = 50$ | -2.216525 | × | -2.117678 | × | -2.224952 | × | -2.330580 | × | SV_WDPMH<br>$p = 6$, $Q = 100$ | -2.258670 | × |
| SV_WDPMH<br>$p = 5$, $Q = 80$ | -2.253658 | × | -2.116305 | × | -2.230707 | × | -2.330760 | × | SV_WDPMH<br>$p = 6$, $Q = 150$ | -2.259303 | × |
| SV_WDPMH<br>$p = 5$, $Q = 100$ | -2.229295 | × | -2.111057 | × | -2.236428 | × | -2.332103 | × | SV_WDPMH<br>$p = 6$, $Q = 200$ | -2.254812 | × |

We also check the efficiency of MCMC draws of these fittings, and find that some of the models produce high inefficiency scores. Based on visualization of MCMC trace plot, we think an inefficiency score larger than 100 is close to failure of burn-in. Therefore for each data set, we exclude the models that produce higher than 100 inefficiency in any of the parameters' draws, and use the rest (as efficient models) to visualize the LPS comparison in Figure 3.7.

Figure 3.7: Visualization of model comparison results of Table 3.8; In each plot, the x-axis is model index specified in Table 3.8, and y-axis is LPS value; The solid line is the level of LPS produced by SV_PM, and the dotted line is the level of LPS given by SV_DPM; "p" represents a WDPMP model, "e" represents a EWDPM model and "h" represents a WDPMH model.



Most of WDPMH models are excluded from this plot because of high inefficiency. We find that efficient weight (EWDPM) models produce lower inefficiency scores than WDPMP models do. Despite of this, 29 of 32 WDPMP models manage well as shown in Figure 3.7

and the three exceptions are associated with relatively small $Q$ among the ones we have used. WDPMP models with $Q > 50$ tend to ensure efficient MCMC draws of parameters for $n = 2000$, and $Q > 100$ tend to ensure for $n = 6000$. The advantage of our WDPMP is visualized in Figure 3.7 and look apparent in terms of higher LPS.

For each data set, we pick the best model and display its parameter inferences and inefficiency scores of MCMC draws in Table 3.9.

Table 3.9: Parameter estimates and inefficiency summary.

| | A | | B | | C | | D | | E | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Est. | Ineff. | Est. | Ineff. | Est. | Ineff. | Est. | Ineff. | Est. | Ineff. |
| $\sigma_\eta^2$ | 0.033<br>(0.0155,0.0591) | 30.3 | 0.023<br>(0.114,0.402) | 20.5 | 0.034<br>(0.0193,0.0550) | 21.2 | 0.056<br>(0.0369,0.0837) | 7.4 | 0.033<br>(0.0227,0.0585) | 25.1 |
| $\phi$ | 0.972<br>(0.942,0.993) | 4.8 | 0.965<br>(0.934,0.991) | 4.1 | 0.974<br>(0.951,0.994) | 2.1 | 0.972<br>(0.947,0.989) | 1.8 | 0.975<br>(0.949,0.994) | 1.4 |
| $\mu_0$ | 0.433<br>(-0.931,0.071) | 1.9 | $-0.660$<br>(-1.016,-0.323) | 2.8 | $-1.46$<br>(-2.084,-0.870) | 3.6 | NA | NA | $-2.12$<br>(-2.505,-1.729) | 2.9 |
| $\sigma_z^2$ | 5.35<br>(4.60,6.33) | 14.3 | 3.89<br>(3.31,4.61) | 9.57 | 5.31<br>(4.20,6.73) | 15.8 | NA | NA | 5.47<br>(4.81,6.29) | 8.3 |
| $M$ | 4.53<br>(2.84,6.90) | 14.4 | 5.37<br>(3.36,8.15) | 12.4 | 3.13<br>(1.78,4.99) | 15.1 | NA | NA | 3.14<br>(2.44,5.81) | 13.3 |
| $W$ | 0.00157<br>(0.0003,0.0037) | 1 | 0.010<br>(0.006,0.015) | 1 | 0.000052<br>(0,0.000498) | 1 | NA | NA | 0.00134<br>(0.0005,0.0024) | 1 |
| $\psi$ | 1.11<br>(0.95,1.37) | 8.3 | 1.18<br>(0.96,1.35) | 2.1 | 0.884<br>(0.803,0.956) | 1.9 | NA | NA | 0.865<br>(0.730,0.992) | 2.5 |

In the five cases A-E of Table 3.9, we display the point estimates and 95% credit interval of the parameters. A is Apple data fitted by its best model: SV_WDPMP with $p = 5$, $Q = 100$; B is Exxon data fitted by its best model: SV_WDPMP with $p = 5$, $Q = 80$; C is DAX data fitted by its best model: SV_WDPMP with $p = 5$, $Q = 40$; D is S&P500 data fitted by its best model: SV_PM; E is NYSE data fitted by its best model: SV_WDPMP with $p = 6$, $Q = 200$. Each point estimate is an average value of 5000 MCMC draws, and each 95% credit interval is specified by the $126^{th}$ and $4875^{th}$ smallest values of these 5000 draws. We also calculate the inefficiency score $1 + 2\sum_{l=1}^{K} \rho(l)$ for each parameter's MCMC draws, and put the score as the subscript of posterior mean.

As we have seen in simulation result, in WDPMP the inefficiency of $\sigma_\eta$ and $\zeta$ is higher than those of the other parameters. But even the largest inefficiency (contained by the sampling of $\sigma_\eta^2$) is much lower than 100, indicating our MCMC sampling in empirical data cases performs well and the posterior draws of parameters mix fast.

## 3.9 Summary/Future Research

In this Chapter we propose a new weight function for WDPM. Our new weights can be used as probabilities without re-scaling since their sum is 1 in nature, and they also satisfy several other properties such as monotonicity, uniqueness, and close form of posterior of most parameters. Using proposed weight function, we utilize historical prices to construct candidate densities and estimate weights for each observations. Based on large number of fittings, our model produce robustly better performance in LPS compared to PM, DPM and WDPM with Gaussian Kernel weights. Hence we suggest using WDPMP in order to get higher marginal likelihood. In terms of computation time, we see that WDPMP costs only about 20% extra time compared to EWDPM and 4% compared to WDPMH. The advantage of WDPMP can offset its disadvantage. Therefore, WDPMP is the optimal type of model based on our study.

Through our simulation and empirical data analyses, we also find that the number of candidates $Q$ plays very important role in WDPMP model performance because a too small or too large $Q$ can under-specify or over-specify the diversity of $a_t$'s among observations. For example at $n = 500$, we find that $Q = 27$ is close to the optimal choice. At $n = 2000$ we find $Q \geq 50$ is necessary while we do not suggest go beyond $Q = 100$. Hence, it is worthwhile to derive the optimal choice of $Q$ in the future and find the relationship among $Q$, $p$ and $n$. It will require to show theoretical justification and intensive simulation studies.

Our approach is not limited within nonparametric Bayesian models. The idea of specifying multiple candidate densities and assigning weights to each observation can be applied to parametric model as well. Our weight function can be applicable to other models such as GARCH, using which we can directly model the standardized return $R_t/h_t^{0.5}$ by applying weighted DPM rather than model the transformed return (i.e. $\log[(R^2+c)/h_t]$ in SV model). We can construct candidate priors of $R_t/h^0.5_t$ using infinite of mixtures of normal distributions and assign weights for each observation to select to check whether this approach is able to capture the complexity in the distribution of $R_t/h_t^{0.5}$.

# Chapter 4

# Weighted Dirichlet Process Mixture GARCH Model for Predicting Stock Price

## 4.1 Introduction

The Generalized autoregressive conditional heteroskedasticity (GARCH) model was proposed by Bollerslev (1986) and has been widely adopted in analyzing the autocorrelated volatility of financial asset prices. The essential idea of GARCH (p,q) model is to model the volatility of observation $t$ (denoted as $h_t$) as the follows:

$$h_t = \alpha_0 + \alpha(\mathrm{B})y_t^2 + \beta(\mathrm{B})h_t,$$

where $y_t$ is the log-return of observation $t$ and $\alpha(\mathrm{B})$, $\beta(\mathrm{B})$ are the polynomial of back-shift operator $B$ with power $p$ and $q$. This original form of GARCH model was firstly adopted to study daily exchange rates in Bollerslev (1987). There are numerous ways that have been proposed to modify or extend the original model as well. Jorion (1988) introduces jump effect into GARCH model. Vlaar and Palm (1993), Nieuwland et al (1994), Chan and Maheu (2002) and Maheu and McCurdy (2004) also further employed the idea of Jorion (1988). Nelson (1991) consider the log of volatility as linear combination of previous standardized error terms and develops exponential GARCH model. Malmsten (2004) evaluates this type of model by testing it against alternative types of GARCH such as GJR-GARCH (short for Glosten, Jagannathan & Runkle GARCH) proposed in Glosten et al (1993), where they assume asymmetric response to previous shocks and assume that volatility has extra correlation to positive shocks that are not contained by negative ones. Engle and Ng (1993) extend the idea of asymmetric response and introduce shift parameter into the square of error terms, and Sentana (1995) further generalizes it to a quadratic form of the error term vector. Instead of assuming linear response function of previous shocks, Hagerud (1997), Gonzalez-Rivera (1998) and Anderson et al (1999) adopt non-linear response function based on smooth transition and explain in detail how parameter inference can be derived using quasi-likelihood.

The main technique in analyzing GARCH model is maximum likelihood approach. Compared to classical method, the previous studies related to Bayesian GARCH model are relatively inadequate. Kim et al (1998) rewrite GARCH (1,1) of $y_t$ as an ARMA(1,1) process of $y_t^2$ and introduce prior distributions for its parameters to compare the marginal likelihood of GARCH model to the one produced by their Bayesian Stochastic Volatility (SV) model (which models the log volatility of $y_t^2$ as autocorrelated process and can be viewed as an alternative approach of GARCH model). While major proliferation of frequentist GARCH models focus on how to specify the response function that determines $h_t$ using information prior to time point $t$, Bayesian GARCH model among previous studies involve more contribution in modifying the distribution of standardized error term which is $y_t h_t^{-1/2}$ in the GARCH model. A popular modification is to assume fat tail distribution of standardized error term such as Student-t (Bauwens and Lubrano, 1998; Wago, 2004; Asai and Watanabe, 2004) instead of Gaussian distribution.

Bayesian approach can be viewed as a promising alternative way of classical method in analyzing GARCH model because it is less sensitive on the sample size and is more accurate to capture complexity of asset return fluctuation. That is, Bayesian approaches are less sensitive to sample size while classical methods usually require large sample size to obtain accurate parameter estimates. Also since the predictive density in Bayesian approach is not conditional on that all of the parameters are fixed at certain values, some of the parameters can be dropped because the derivation of predictive density involves the integration of those parameters over their supports. Therefore, compared to classical methods of parameter estimation that are based on maximizing likelihood (or quasi-likelihood), the inference provided by Bayesian approached can better capture the complexity of asset return fluctuation (Ausín and Galeano, 2007).

While parametric approaches explicitly specify the type of distribution that standardized error terms follow, Dirichlet process mixture (DPM) introduced in Ferguson (1973) provides us a nonparametric way to model such distribution using infinite mixture of Gaussian distribution. DPM contains extra model flexibility because the parameters that determine the conditional distribution of error terms are given randomized prior distribution. Such randomized prior means that the expectation and variance of each component (in the infinite mixture of normals) and their corresponding probabilities depends are subject to the change of data. Therefore, different data can provide different marginal distribution of error terms while parametric approach fixes the marginal distribution once priors are specified. Few studies on relating DPM to GARCH can be found and a recent one is MacEachern

(2000). Although DPM introduces model flexibility, its implicit assumption is that all of the observations follow the same prior which is sampled from base measure. To relax such assumption, MacEachern (2000) viewed DPM as the “stick-breaking” process (Sethuraman, 1994) and propose that locations of atoms depend on covariate values. Griffin and Steel (2006) further introduce order-based dependent DP which assumes that not only the locations but also the weights of the atoms are subject to the change of covariate values. Dunson et al (2007) introduced the generalization of DPM in a different way and propose weighted DPM (WDPM) which is essentially a mixture of $n$ prior distributions, each of which are independently sampled from base measure for each observation. This is equivalent to providing $n$ candidate priors and constructing the distribution of selecting these candidates (i.e. the weight function) using distance in covariate values. Sun et al (2015) propose that the number of candidates do not have to be as large as $n$ and the candidate priors do not have to be sampled at observation location either. They introduce a new weight function other than Gaussian Kernel type and show that computational efficiency and model accuracy can be achieved at the same time if the weight function and the number of candidate priors (say $Q$ and $Q \ll n$) is properly chosen.

In this Chapter we will develop WDPM for GARCH model because WDPM provides model flexibility and accuracy and GARCH focus on the return rather than log-squared return. The advantage of such weighted nonparametric approach in analyzing daily stock prices has been found in Chapter 3 where we model log-squared return. Instead of this, GARCH models focus on the return rather than log-squared return. We are interested to see whether the modeling directly on the return will amplify the advantage of WDPM approach. Another motivation for proposing WDPM for GARCH is that we can adopt a new model evaluation approach (instead of marginal likelihood) which can produce straightforward result about model profitability, and WDPM provides us much more choices in selecting the better model in terms of such profitability. We will focus on GARCH (1,1) since it is found that this simplest form of GARCH model is able to capture the autocorrelated property of the volatility displayed in time series of asset prices.

The rest of this Chapter is organized as: Section 4.2 illustrates the limitation of Student-t and DPM GARCH model and Section 4.3 introduces the GARCH model based on WDPM; In Section 4.4 we propose our model evaluation approach for GARCH which produces intuitive result about model profitability; We introduce an alternative way based on stochastic volatility (SV) to analyze asset daily returns in Section 4.5 and explain how to conduct MCMC sampling for WDPM GARCH and how to compare GARCH to SV based on marginal likelihood in Section 4.6; Section 4.7 and 4.8 include correspondingly the results from simulations and empirical studies; Section 4.9 briefly concludes.

## 4.2 Limitation of Student-t and DPM Garch Model

Let $P_{t-1}$ and $P_t$ be the return prices at time $t-1$ and $t$. Then the log-return of daily closing price $y_t$ can be calculated as $y_t \triangleq 100 \times \log(P_t/P_{t-1})$. Let $\epsilon_t$ be the white noise with the precision parameter $\lambda_t$. Enlightened by Geweke (1993) and Ardia and Hoogerheide (2010), we specify the GARCH(1,1) model with Student-t error terms as:

$$\begin{aligned}
&y_t = \varepsilon_t h_t^{1/2} \quad t = 1, \cdots, n \\
&h_t = \alpha_0 + \alpha_1 y_{t-1}^2 + \beta h_{t-1} \\
&\varepsilon_t \sim \mathrm{N}(0, \lambda_t^{-1}) \\
&\lambda_t \sim \mathrm{Gamma}(\nu/2, \nu/2)
\end{aligned} \tag{4.1}$$

where unknown parameters, $\alpha_0 \geq 0$ and $\alpha_1, \beta > 0$, for modeling volatility $h_t$.

Integrating over $\lambda_t$, we know that the marginal distribution of $\varepsilon_t$ is Student-t with degree of freedom $\nu$. Since $h_t$ in (4.1) represent the volatility of $y_t$, we set $\alpha_0 \geq 0$ and $\alpha_1, \beta > 0$ to ensure that $h_t$ is always positive. We also need $\alpha_1 + \beta < 1$ to guarantee the time series $\{h_t\}$ is stationary. In DPM model we explain the error terms as:

$$\begin{aligned} \varepsilon_t &\sim \mathrm{N}(\mu_t, \alpha\sigma_e^2) \\ \mu_t &\sim G(\cdot) \\ G(\cdot) &\sim \mathrm{DP}(\zeta, G_0) \\ G_0(\cdot) &\equiv \mathrm{N}(\mu_0, (1-\alpha)\sigma_e^2) \end{aligned} \tag{4.2}$$

where $\alpha$ is fixed at a value close to 0 to avoid identifiability problem (Sun et al, 2015). The distribution $G(\cdot)$ is randomly sampled from $\mathrm{DP}(\zeta, G_0)$ where $\zeta$ is the concentration parameter and $G_0$ is the base measure. The MCMC algorithm for DPM model is based on P$\acute{o}$lya urn scheme proposed in Blackwell and MacQueen (1973) and a further study of it for posterior distribution in MacEachern et al (1998).

The difference between Student-t and DPM is that the former one tries to explain the complexity of $\varepsilon_t$ by heavy tail known distribution while the latter one tries to do so using Dirichlet Process. However, both Student-t and DPM models share the same assumption that all of the observations' error term (i.e. $\varepsilon_t$) follow the same marginal distribution. This is actually a strong assumption because sharing the same marginal distribution means that all of the observations share the same predictive density of $\varepsilon_t$ provided by our Bayesian inference. This may make the model unable to capture the complexity of the stock prices and lead to inaccurate results.

Hence one limitation of Student-t and DPM model is that their model performances are not easy to be tuned because they do not leave us too much choices in model selection. Although we can modify the value of $\nu$ in Student-t and $\alpha$ in DPM model, it is found that they are not able to result in adequate model improvement once they are around their optimal level (in practice we find $\nu = 7$ and $\alpha = 0.05$ are about the best choices based on accuracy and speed of mixing in MCMC sampling). Another limitation in DPM model is that we find it can sometimes produce MCMC draws which fail to converge within maximum number of iterations. Motivated by these limitations of Student-t and DPM model, we propose WDPM for GARCH.

## 4.3 Weighted DPM Garch Model

We can relax the assumption that all of the observations share the same marginal distribution by developing DPM to weighted approach, where we provides $Q$ candidate prior distributions and each observation randomly pick its prior from these candidates following certain probability distribution. Relaxing such assumption can be crucial because it proliferates the model flexibility and may potentially capture the complexity of $\varepsilon_t$'s. It can also bring us much more candidate models among which we can seek for greater model improvement.

Suppose that we have $Q$ candidate priors and the probability that the $t^{th}$ observation select the $q^{th}$ candidate prior as $G_t(\cdot)$ is modeled by the weight function $\pi(\boldsymbol{x_t}, \boldsymbol{x_q^c})$, where $\boldsymbol{x_t}$ and $\boldsymbol{x_q^c}$ represent the explanatory covariate vectors contained by observation $t$ and candidate $q$. The explanatory covariates are assumed to be related to such probability, and we assign high weight when $\boldsymbol{x_t}$ is close to $\boldsymbol{x_q^c}$ in terms of Euclidean norm. Then the full model specification for WDPM GARCH model is:

$$
\begin{aligned}
y_t &= \varepsilon_t h_t^{1/2} \quad t = 1, \cdots, n, \\
h_t &= \alpha_0 + \alpha_1 y_{t-1}^2 + \beta h_{t-1}, \\
\varepsilon_t &\sim \mathrm{N}(\mu_t, \alpha\sigma_e^2) \\
\mu_t &\sim G_t(\cdot) \\
G_q^c(\cdot) &\overset{i.i.d}{\sim} \mathrm{DP}(\zeta, G_0(\cdot)) \quad q = 1, \cdots, Q, \\
\Pr(G_t(\cdot) \equiv G_q^c(\cdot)) = &\ \pi(\boldsymbol{x_t}, \boldsymbol{x_q^c}) \quad q = 1, \cdots, Q;\ t = 1, \cdots, n, \\
\sum_{q=1}^{Q} \pi(\boldsymbol{x_t}, \boldsymbol{x_q^c}) = &\ \ 1 \quad t = 1, \cdots, n \\
G_0(\cdot) &\equiv \mathrm{N}(\mu_0, (1-\alpha)\sigma_e^2).
\end{aligned}
\tag{4.3}
$$

For this WDPM GARCH model, three questions need to be addressed: the first is what covariate should be used to explain weight which; the second is how to specify covariate value for candidate priors (The observations' covariate values are usually given by data, but we do need to specify such values for candidates since their existences are proposed by our approach); the third is how to use covariate values to calculate weight (i.e. we need to specify the weight function). We explain how to address these issues in Section 4.3.1-4.3.3.

### 4.3.1 Explanatory Covariate

In this study we use historical returns as explanatory covariate. This means we assume historical prices contain predictive information about the marginal distribution of $\varepsilon_t$. Therefore, for observation $t$, the covariate vector is given as $x_t = (R_{t-p}, \cdots, R_{t-1})$ which contains the latest $p$ days' returns prior to trading day $t$. The stock price autocorrelation has been studied since Cowles and Jones (1937), and evidences against the famous "weak-form market efficiency" hypothesis (Fama, 1970) have been found in literature such as Jegadeesh and

Titman (1995), Llorente et al (2002) and Anderson et al (2012), which motivate us to use historical prices as the covariate for weight function.

### 4.3.2 Specification of Candidate Locations

For the convenience of illustration, we denote $\boldsymbol{x_t}$ as the "location" of observation $t$. Using historical returns, we also need the locations of candidates to construct weight function. There are multiple choices for $p$, but once it is fixed, both observations and candidates' locations must be the vectors of same length $p$. Since the observations' locations are essentially $n$ elements of $R^p$, we want the candidates' locations to be $Q$ typical representatives of this space. Based on an initial investigation of stock price data, it is found that $-1, 0, 1$ are correspondingly typically low, medium and high daily returns. Therefore, we denote $-1, 0, 1$ as $S_x$ to construct $S_x^p$. Since there are $3^p$ elements in $S_x^p$, $Q$ should be no larger than $3^p$. If $Q = 3^p$, all of the elements in $S_x^p$ are adopted as candidate locations. But if $Q < 3^p$, we need to do selection: For each element in $S_x^p$, we can calculate its $n$ distances to the observations and derive the average distance. Thus each element in $S_x^p$ can be associated with one average distance. We can sort these elements by such average distance and select the $Q$ smallest ones as the candidates locations.

### 4.3.3 Weight Function

Having the observations' locations $(\boldsymbol{x_1}, \cdots, \boldsymbol{x_n})$ and candidates' locations $(\boldsymbol{x_1^c}, \cdots, \boldsymbol{x_Q^c})$, we need to construct a probability mass function for each observation whose possible choices are the $Q$ candidates. The weight function is essentially a function of $R^p \times R^p \rightarrow [0, 1]$ which will be decreasing as the Euclidean norm $\left\|\boldsymbol{x_t} - \boldsymbol{x_Q^c}\right\|$ increases. Dunson et al (2007) adopt

Gaussian Kernel and propose:

$$\pi(\boldsymbol{x_t}, \boldsymbol{x_q^c}) \propto \frac{\gamma_q e^{-\psi \left\| x_t - x_q^c \right\|^2}}{\sum_{l=1}^{Q} \gamma_l e^{-\psi \left\| x_t - x_l^c \right\|^2}} \qquad \text{for } \ q = 1, \cdots, Q \tag{4.4}$$

where $(\gamma_1, \cdots, \gamma_Q; \psi)$ are hyper-parameters taking nonnegative values. In Chapter 2 we have discussed the potential issues associated with this weight function and proposed:

$$\pi(\boldsymbol{x_t}, \boldsymbol{x_q^c}) \propto \frac{e^{-\psi_q \left\| x_t - x_q^c \right\|^2}}{\sum_{l=1}^{n} e^{-\psi_l \left\| x_t - x_l^c \right\|^2}} \qquad \text{for } \ q = 1, \cdots, Q \tag{4.5}$$

and provide four alternative ways to model weights: EWDPM (short for efficient WDPM) which assumes all of the components in $(\psi_1, \cdots, \psi_Q)$ take the same value and WDPMG, WDPME, WDPMH which correspondingly assumes Gamma, Exponential, Half-Cauchy priors for $(\psi_1, \cdots, \psi_Q)$. They display the advantage of these four weighted approaches compared to the one based on (4.4) in both model accuracy and computational efficiency.

In Chapter 3 we propose a new way (WDPMP) to model weight based on sorting the $Q$ distances from a given observation to the candidates and use conditional probabilities to construct weights as:

$$\begin{aligned} \pi(\boldsymbol{x_t}, \boldsymbol{x_{(q)}^c}) &= \gamma_t^{q-1}[\prod_{l=1}^{q-1}(1 - p_{(l)t})][1 - \gamma_t(1 - p_{(q)t})] \qquad \text{for } \ q = 1, \cdots, Q \\ p_{(l)t} &= \frac{\left\| x_t - x_{(l)}^c \right\|^{-\psi}}{\sum_{r=l}^{Q} \left\| x_t - x_{(r)}^c \right\|^{-\psi}} \end{aligned} \tag{4.6}$$

where: $\pi_{(q)t}$ represents the probability that observation $t$ selects its $q^{th}$ nearest candidate as prior; and $p_{(0)t} = 1$.

We proved that (4.6) satisfies several good properties such as it does not need to rescale weights to make their sum equal 1 and that closer candidate is always more likely to be chosen no matter how the hyper-parameters change. In fact, these properties are not satisfied by either (4.4) or (4.5). Recall that this approach is based on conditional probabilities, therefore it is denoted as WDPMP (short for WDPM with probabilistic weight function). We will apply EWDPM, WDPMG, WDPME, WDPMH and WDPMP to GARCH, and compare these weighted models through simulation and empirical studies.

## 4.4 Back Testing Return

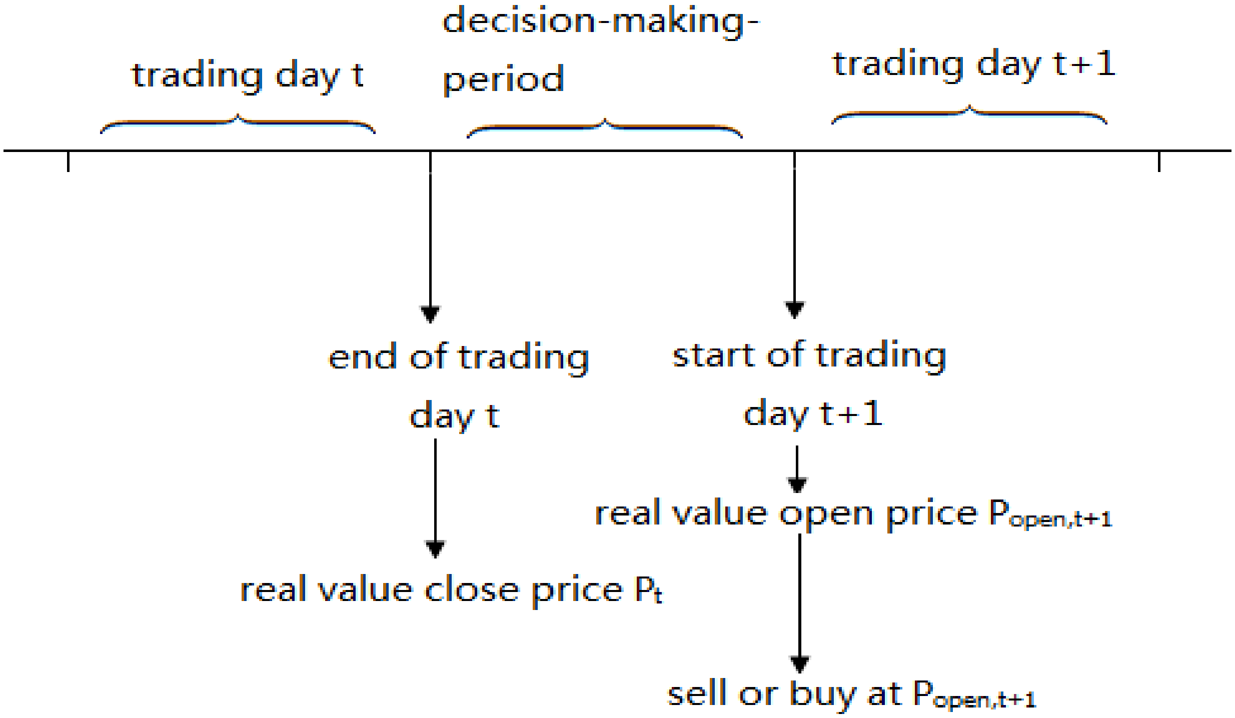
In this section, we propose back testing return (BTR) which is an empirical way of model evaluation. back testing is commonly used in financial analyses to examine trading strategy. The essential idea is to apply our strategy on historical data and simulate the real process of investment. Although the real focus is how the strategy will perform in the future, it will take a very long period to collect sufficient observations if one wants to test it using future data. Therefore back testing based on previous observations we have already seen is one useful tool to check the profitability and risk associated to a strategy.

Enlightened by the idea of back testing, we develop an empirical way of model evaluation which applies model prediction to historical data and produce the corresponding return of trading within a certain period. We noted that the authentic back testing used in financial area is to test different trading strategies while our model evaluation check how the differences in prediction produced by different models affect the return by using the same strategy. That is, all of our models will adopt the same strategy which is introduced in this section and the differences in return is caused by model prediction only.

### 4.4.1 Decision-Making-Period

Decision-making-period is defined to be a certain period which starts from the time we see the closing price of trading day $t$ and ends at the time we see the open price of trading day $t + 1$. Figure 4.1 graphically illustrates the meaning of Decision-Making-Period.

Figure 4.1: Graphical illustration of decision-making-period.



In our BTR approach, we fix the length between two decision-making-period to be 10 trading days. Therefore, if we include $k$ decision-making-period, it means that we are checking the model within a historical period which includes $10k$ observations. We use $n_p$ to denote such number (i.e. $n_p = 10k$). For example, when $n_p = 2000$, there will be 200 decision-making-periods. Trading (either to buy or to sell) can only happen at the end point of a decision-making-period.

### 4.4.2 Trading Strategy

The key idea of our trading strategy is to predict $P_{t+10}$ during decision-making-period and decide the action to be taken at the starting point of trading day $t+1$ once we observe the open price of trading day $t+1$. We denote the amount of money in the investment pool at the end of trading day $t$ to be $\mathrm{Mn}_t$ and the money at the starting point of trading day $t+1$ to be $\mathrm{Mn}^*_{t+1}$. As time goes on, the amount the money in the investment pool may change due to trading. We further let $\delta_t = 1$ means that we are holding the stock on trading day $t$ and $\delta_t = 0$ means not holding. Our trading strategy can be derived by maximizing the conditional mean of the amount of money ten days later which is

$$\mathop{\mathbf{Max}}_{\delta_{t+1}} \ \mathrm{E}(\mathrm{Mn}_{t+10}|\delta_t, \delta_{t+1}), \tag{4.7}$$

where $t$ means that there is a decision-making-period between day $t$ and day $t+1$.

To determine $\mathrm{E}(\mathrm{Mn}_{t+10}|\delta_t)$ we need both $\delta_t, \delta_{t+1}$ and the prediction of $P_{t+10}$. For example, if $\delta_t = 1$ and $\delta_{t+1} = 1$, then $\mathrm{E}(\mathrm{Mn}_{t+10}|\delta_t, \delta_{t+1})$ should be $\frac{\mathrm{Mn}_t}{P_t}\hat{P}_{t+10}$, where $\hat{P}_{t+10}$ is the predicted price of $t+10$. If $\delta_t = 1$ but we choose to sell (i.e. $\delta_{t+1} = 0$), then $\mathrm{E}(\mathrm{Mn}_{t+10}|\delta_t, \delta_{t+1})$ should be $\frac{\mathrm{Mn}_t}{P_t}P_{\mathrm{open},t+1} - C_0$, where $P_{\mathrm{open},t+1}$ is the open price of day $t+1$ and $C_0$ is the cost of trading (buy or sell) which is a fixed charge regardless of trading volume. $P_{\mathrm{open},t+1}$ plays important roll in our back testing method the actions (to sell, to buy, to hold or to stay away) are taken at the starting point of trading day $t+1$. We can see that even if we stop holding from day $t+1$ the amount of money will be slightly different from $\mathrm{Mn}_t$ because of $C_0$ and the difference between $P_t$ and $P_{\mathrm{open},t+1}$. In simulation study we can assume $P_{\mathrm{open},t+1}$ is always equal to $P_t$. But if we apply our approach to real market data, these two prices are usually different. We have observed that such difference fluctuate around zero and tend to have zero mean in the long run.

Our proposed trading strategy implies that we are risk neutral and use only expected profit to determine action. Based on (4.7), we can derive the trading strategy and quantified it as:

$$\delta_{t+1} = \begin{cases} 0, \text{ if } \delta_t = 1 \text{ and } \frac{\mathrm{Mn}_t}{P_t}(\hat{P}_{t+10} - P_{\text{open},t+1}) + C_0 < 0; \\ 1, \text{ if } \delta_t = 1 \text{ and } \frac{\mathrm{Mn}_t}{P_t}(\hat{P}_{t+10} - P_{\text{open},t+1}) + C_0 > 0; \\ 0, \text{ if } \delta_t = 0 \text{ and } \frac{\mathrm{Mn}_t}{P_{\text{open},t+1}}(\hat{P}_{t+10} - P_{\text{open},t+1}) - C_0 < 0; \\ 1, \text{ if } \delta_t = 0 \text{ and } \frac{\mathrm{Mn}_t}{P_{\text{open},t+1}}(\hat{P}_{t+10} - P_{\text{open},t+1}) - C_0 > 0; \end{cases} \tag{4.8}$$

where one can recall that $\hat{P}_{t+10}$ is the predicted price of $t + 10$, $P_{\text{open},t+1}$ is the open price of day $t + 1$ and $C_0$ is the cost of trading (buy or sell) which is a fixed charge regardless of trading volume. We consider the commission fee $C_0$ is fixed charge per order (either to sell of to buy) and it includes all of the related charges for trading. (4.8) implies that we use all of the money in investment pool to trade and do not consider opportunity cost. It also implies that we can always buy or sell at the open price once we observe it, which is very close to reality for high trading volume stock. Another advantage of focusing on high trading volume stock is that we can assume our trading volume is not large enough have any effect on target stock price.

### 4.4.3 Real Value Prices

Before we perform BTR, we need to process the raw data because the market closing prices does not represent the real values of stock. This is because of the existence of events such as dividend and stock split. Yahoo Finance provides the adjusted closing prices which can represent the real values of stocks, but we should not use this kind of data, otherwise it would mean that we have included dividends and splits that happen after day $t$ to do prediction of future price at day $t$, which is obviously not the simulation of real prediction.

To derive the proper actual value prices, we need to collect all of the observations needed. Suppose that the sample size for BTR is $n_p$ ($n_p$ is larger than 10 and can be divided by 10). Then there are $n_p/10$ decision-making-periods, and we use $n_0$ observations at each of these periods to fit model and do prediction. Therefore, we need to collect $n_p + n_0$ observations of daily prices in total. The first decision-making-period is between the end of day 0 and the start of day 1. Therefore we denote these observations as $t = -n_0 + 1, \dots, 0, 1, \dots, n_p$. To simplify the problem, we only consider two kinds of events that affect actual value of stock shares: dividend and stock split. We use the market closing price of day $-n_0 + 1$ as benchmark and the real value prices can be calculated as:

$$P_j = P_j^* \prod_{r_1=1}^{S_j} m_{r_1} + \sum_{r_2=1}^{D_j} (\Delta_{r_2} \cdot \prod_{t(r_1)<t(r_2)} m_{r_1}), \tag{4.9}$$

where: $P_j$ and $P_j^*$ are the real value price and market closing price of trading day $j$; $S_j$ is the number of stock split happened between day $t - n_0$ and day $j$, and among these $S_j$ splits, the $r_1^{th}$ split is $1 : m_{r_1}$ (meaning each share in hand is split into $m_{r_1}$ shares); $D_j$ is the number of dividends happened between day $t - n_0$ and day $j$, and among these $D_j$ dividends, the $r_2^{th}$ dividend is $\Delta_{r_2}$ dollar for each share in hand; $t(r_1) < t(r_2)$ means the stock splits happened before the $r_2^{th}$ dividend. Table 4.1 provides an example of how the real value price is derived from market closing price and how it is different from adjusted closing price and market closing price.

Table 4.1: Example of how the real value price and Yahoo finance adjusted closing price are different from each other; The trading days are labeled from 1 to 8; The trading day 1 is the earliest one while trading day 8 is the latest one; The real value price is given by (4.9); For example, the real value price of trading day 8 is $13.27 * 2 + 0.08 * 2 + 0.13 = 26.83$; The derivation of Yahoo adjusted closing price can be found on https://help.yahoo.com/kb/SLN2311.html.

| Trading Day | Market Closing Price | Event | Real Value Price | Yahoo Adjusted Closing Price |
|---|---|---|---|---|
| 8 | 13.27 | | 26.83 | 13.27 |
| 7 | 13.11 | dividend 0.08 | 26.51 | 13.11 |
| 6 | 13.20 | | 26.53 | 13.12 |
| 5 | 13.51 | share split 1:2 | 27.15 | 13.43 |
| 4 | 26.71 | | 26.84 | 13.27 |
| 3 | 25.80 | dividend 0.13 | 25.93 | 12.82 |
| 2 | 26.19 | | 26.19 | 12.95 |
| 1 | 25.32 | | 25.32 | 12.52 |

Because of the existence of dividend, we need to assume that the value transferred to dividend can be applied to investment immediately to simplify the problem. This may seem far away from real cases because it usually takes a while for the dividend to be transferred to investors and become available money, and it is subject to tax as well. However this can be easily achieved by having certain amount of back-up money which is available at any time. We can match the amount of dividend, say $\Delta_0$ whenever there is one and transfer this amount of money from back-up pool to investment pool without any difficulty.

### 4.4.4 BTR Procedure

Once we collect the observations, we can derive the log returns as the final data used in model fitting and prediction. Then the BTR can be derived by the following steps:

Step-1: For each $t = 10k$ $(k = 0, 1, \cdots, n_p/10)$, apply model to derive parameter estimates and predictive densities.

Step-2: For each $t = 10k$ $(k = 0, 1, \cdots, n_p/10)$, use parameter estimates and predictive densities to perform MCMC sampling for $P_{t+10}$. In weighted approach, the sampling is as following: $\hat{h}_{j+1} = \hat{\alpha}_0 + \hat{\alpha}_1 y_j^2 + \hat{\beta}\hat{h}_j$, $\varepsilon_j \sim \hat{f}_j(\cdot)$ and $y_{j+1} = \hat{h}_{j+1}\varepsilon_j$ for $j = t, t+1, \cdots, t+9$. $\hat{f}_j(\cdot)$ is the predictive density for $\varepsilon_j$, the derivation of which will be introduced in Section 4.6.2. We can easily derive the approximation of inverse cdf

$F_j^{-1}(\cdot)$ from predictive density, and $\varepsilon_j$ can be sampled by $\hat{F}_j^{-1}(u)$ where $u \sim \mathrm{U}[0,1]$. Having $(y_{t+1}, \cdots, y_{t+10})$ sampled, we can derive $P_{t+10}$ which can be viewed as the result of sampling.

Step-3: Repeat Step-6 for 5000 times and calculate the average of MCMC draws of $P_{t+10}$ for $t = 10k$ $(k = 0, 1, \cdots, n_p/10)$. Use the averages $\hat{P}_{t+10}$ as predicted real value prices of $t + 10$.

Step-4: Denote the initial amount of actual money in the investment pool at the end of trading day 0 as $\mathrm{Mn}_0$. Repeat the following Step-5 and 6 for $t = 10k$ $(k = 0, 1, \cdots, n_p/10)$:

Step-5: At the start of trading day $t+1$, convert the market open prices $P^*_{\text{open},t+1}$ to real value price $P_{\text{open},t+1}$ using (4.9). If we are not holding the stock, we buy it if $\frac{\mathrm{Mn}_t}{P_{\text{open},t+1}}(\hat{P}_{t+10} - P_{\text{open},t+1}) - C_0 > 0$ and choose to not to buy otherwise; If we are holding the stock, we sell it if $\frac{\mathrm{Mn}_t}{P_t}(\hat{P}_{t+10} - P_{\text{open},t+1}) + C_0 < 0$ and continue to hold it otherwise.

Step-6: Update the actual money in investment pool at the end of trading day $t + 10$. If we were holding the stock at the end of trading day $t$ and did not sell it on day $t + 1$, we have: $\mathrm{Mn}_{t+10} = \frac{\mathrm{Mn}_t}{P_t} P_{t+10}$; If we were holding and did sell it on day $t + 1$, $\mathrm{Mn}_{t+10} = \frac{\mathrm{Mn}_t}{P_t} P_{\text{open},t+1} - C_0$; If we were not holding and did buy it on day $t + 1$, $\mathrm{Mn}_{t+10} = \mathrm{Mn}_t \frac{P_{t+10}}{P_{\text{open},t+1}} - C_0$; If we were not holding and did not buy it, $\mathrm{Mn}_{t+10} = \mathrm{Mn}_t$

Step-7: Denote the amount of actual money in investment pool after the last update as $\mathrm{Mn}_n$. Calculate BTR:

$$\mathrm{BTR} = \frac{\mathrm{Mn}_n - \mathrm{Mn}_0}{\mathrm{Mn}_0} \times 100\% \tag{4.10}$$

Based on the implementation of this approach, we find that more than 90% of the computation burden lies on Step-1 while Step-3 also takes a small but considerable proportion of

CPU time. However, the advantage of this approach is that we can adopt parallel computation for Step-1 because all of the observations are historical data and have been collected. The $n_p/10$ model fittings can be performed at the same time so far we have enough CPU cores. Once Step-1 is finished, the predictions in different decision-making-periods realized by Step-3 can also be implemented by parallel computation. Such advantage enables us to greatly reduce the computation burden.

At Step-5, we find that there is one thing left for us to specify exogenously: whether we are holding the stock or not at day 0. This should not cause big difference if the open price of day 1 and closing price of day 2 are close and the commission fee takes a small proportion of $\mathrm{Mn}_0$. Despite of this, we will still check both cases in our empirical studies.

## 4.5 Candidate Models

We will compare different types of models in the following sections. To denote GARCH models, we use "G_" followed by the corresponding approach. Our candidate GARCH models are G_t, G_DPM, G_WDPMP, G_EWDPM, G_WDPMG, G_WDPME and G_WDPMH. For weighted GARCH models, we can provide more choices of models by using different $p$ and $Q$. Besides GARCH, we consider SV model introduced in Chapter 3 as an alternative way to analyze volatility of stock prices:

$$\begin{gathered}\log(R_t^2 + c) = h_t + a_t \quad t = 1, \cdots, n \\ h_t = \mu + \phi(h_{t-1} - \mu) + \sigma_\eta \eta_t\end{gathered} \tag{4.11}$$

where $h_t$ represents log-volatility because the target variable in SV model is log of squared return; $c$ is off-set parameter which keep zero returns from being directly put into log function; $a_t$ is the conditional return; $\mu$ is the global mean of log-volatility; $\phi$ is between -1 and 1 to ensure stationarity; $\eta_t \sim \mathrm{N}(0, 1)$.

Following the same semiparametric approach for SV introduced in Chapter 3 and the same notation introduced in Section 3.7, we add SV_PM, SV_DPM, SV_WDPMP, SV_EWDPM, SV_WDPMG, SV_WDPME and SV_WDPMH into the set of candidate models. Since GARCH is similar to SV in the sense that we decompose stock return into two parts (random innovation and volatility) and model the volatility by autoregressive process, we compare GARCH with SV.

However, there are two key difference between these two approaches. The first is that GARCH consider $h_t$ to be fixed once $y_{t-1}$ and $h_{t-1}$ are given while in SV model there is random innovation of day $t$ that determines the log volatility of day $t$. This means that we do not need to sample $h_t$ in MCMC algorithm of GARCH while algorithms such as Forward Filtering Backward Sampling is needed for SV model to sample the log-volatility. The second difference is that in SV model we use $\log(y_t^2)$ as the response and assume the log-volatility to be autocorrelated. We are actually ignoring the sign of $y_t$ following this approach. Yet in GARCH model, the log-return $y_t$ is directly modeled instead using the log of the square of it. In another word, using GARCH model, we can do prediction about daily return rather than only the square of the return based on parameter inferences. This property means the BTR approach we proposed in Section 4.4 is only for GARCH models and such evaluation is not applicable to SV models.

## 4.6 Bayesian Inference

### 4.6.1 MCMC Algorithm

We develop the MCMC algorithm for weighted DPM Garch model to derive the inferences of parameter which include $(\alpha_0, \alpha_1, \beta, \mu_0, \sigma_e^2, \psi, \boldsymbol{\gamma}, \zeta)$. To ensure that all of the elements in $\boldsymbol{h}$ are positive and the stationarity of this time series, we need the constraints $\alpha_0 \geq 0, \alpha_1, \beta > 0$ and $\alpha_1 + \beta < 1$. These constraints can cause potential difficulties in sampling $\alpha_0, \alpha_1, \beta$ because we need to adopt Metropolis-Hastings (M-H) due to the lack of closed form posteriors. It is hard to choose proposal when the support is truncated, therefore, we introduce the following transformations:

$$\begin{aligned}
\alpha_0 &= \omega^2 \\
\alpha_1 &= \frac{e^{\rho_1}}{1+e^{\rho_1}} \cdot \frac{1}{1+e^{\rho_2}} \\
\beta &= \frac{e^{\rho_1}}{1+e^{\rho_1}} \cdot \frac{e^{\rho_2}}{1+e^{\rho_2}}
\end{aligned} \tag{4.12}$$

Using $\omega, \rho_1, \rho_2$ instead of $\alpha_0, \alpha_1, \beta$, we can easily apply M-H. The prior of $\omega, \rho_1, \rho_2$ are $\mathrm{N}(\omega|0, 4)$, $\mathrm{N}(\rho_1|2, 4)$ and $\mathrm{N}(\rho_2|2, 4)$. Since the volatility vector is fixed once all of the parameters and $h_0$ are fixed (which is different from SV model where $\boldsymbol{h}$ can be randomly sampled at each iteration), we find it necessary to carefully choose $h_0$. Therefore, we treat it as a parameter and adopt the transformation $h_0 = e^{\tau}$ to guarantee $h_0$ is always positive. We impose prior $\mathrm{N}(\tau| - 2, 4)$ for $\tau$. Denoting $\boldsymbol{\theta} \triangleq (\omega, \rho_1, \rho_2, \tau)$, we can update the elements in $\boldsymbol{\theta}$ simultaneously using M-H. Although the parameters that are actually sampled in MCMC are $(\omega, \rho_1, \rho_2, \tau)$, we can easily transfer them back to $(\alpha_0, \alpha_1, \beta, h_0)$ and do inference on the original parameters.

For the sampling of hyper-parameters in weight function, we use Beta(1,1) priors for each element in $\boldsymbol{\gamma}$ and Gamma(2,1) for $\psi$ since quadratic function of norm is the canonical way of modeling distance. We introduce a vector $\boldsymbol{Z}$ where $Z_t = q$ means that observation $t$ select the $q^{th}$ candidate as its prior. Once the hyper-parameters are sampled at each iteration, we need to update the weights and use them to sample $\boldsymbol{Z}$. The prior of $\zeta$ and $\psi$ are correspondingly Gamma(1,0.2) and Gamma(0.2, 0.1). We can follow the approach of Sun et al (2015) to sample $\boldsymbol{\gamma}$, $\psi$, $\boldsymbol{Z}$ and $\zeta$.

At each iteration of sampling, in summary, we will update the parameters and unknown vectors following the order: $\boldsymbol{\theta}$, $\mu_0$, $\sigma_e^2$, $\zeta$, $\boldsymbol{\mu}$, $\boldsymbol{\gamma}$, $\psi$ and $\boldsymbol{Z}$. The sampling of $\boldsymbol{\mu}$ can be realized by generalizing the P*ó*lya urn results to WDPM. We display the joint density in Appendix G and provide the detail of MCMC procedures for G_WDPMP in Appendix H based on joint density. We disregard the first $10,000$ iterations for burn-in and use the following $40,000$ iterations for parameter inference. To reduce correlation of MCMC draws, we apply thinning algorithm and keep only one of every ten consecutive draws to calculate posterior means and inefficiency scores $1 + 2\sum_{l=1}^{K} \rho(l)$ (where $\rho(l)$ is sample autocorrelation of lag $l$ and $K$ is the highest lag after which autocorrelation becomes insignificant).

### 4.6.2 Marginal Likelihood

To calculate marginal likelihood for both SV and GARCH models, we adopt the approach introduced in Chib (1995):

$$\log\{f(\boldsymbol{y})\} = \log\{\pi(\Theta)\} + \log\{f(\boldsymbol{y}|\Theta)\} - \log\{f(\Theta|\boldsymbol{y})\} \tag{4.13}$$

which decomposes the log of marginal likelihood into three parts: The first part is log density of prior distribution of $\Theta$ (parameters used in model) and this is straightforward to calculate once the priors are specified; The second part is log of conditional likelihood and the third part is log posterior density of $\Theta$ which can be derived by extra MCMC iterations where the parameters are consequently removed from the set of target variables to be sampled and fixed at their posterior means of previous MCMC draws.

To calculate $\log\{f(\boldsymbol{y}|\Theta)\}$ for GARCH model, we set $\Theta$ as $\boldsymbol{\theta}$. We can first derive the conditional density of $y_t/\sqrt{h_t}$ for G_WDPMP which can be calculated as:

$$\begin{aligned}\hat{f}(\cdot|\boldsymbol{\theta}) = \frac{1}{N_{max} - N_{bn}} \sum_{it=N_{bn}}^{N_{max}} \sum_{q=1}^{Q} \{ \Pr(Z_t = q|\psi_{(it)}, \boldsymbol{\gamma}_{(it)}) [ \sum_{j=1}^{k_{q(it)}} \frac{n_{q_j(it)}}{\zeta + n_{q(it)}} \mathrm{N}(\cdot|\mu^*_{q_j(it)}, \alpha\sigma^2_{e(it)}) \\ + \frac{\zeta}{\zeta + n_{q(it)}} \mathrm{N}(\cdot, |\mu_{0(it)}, \sigma^2_{e(it)})] \}, \end{aligned} \tag{4.14}$$

where $N_{max}$ and $N_{bn}$ are correspondingly the number of MCMC iterations and burn-in; $Q$ is the number of candidate priors; $n_q$ represents the number of observations sharing the $q_{th}$ candidate prior and $k_q$ is the number of unique values of $\boldsymbol{\mu}$ among these observations; $n_{q_j}$ is the number of observations sharing the $q_j$-th unique value in $\boldsymbol{\mu}$; $(it)$ means the variables take their corresponding values at the $it^{th}$ iteration. Replacing the $\psi_{(it)}, \boldsymbol{\gamma}_{(it)}$ in (4.14) by other proper weight parameters, we can calculate such conditional density for other weighted DPM GARCH. Based on the conditional density of $y_t/\sqrt{h_t}$, one can calculate the second part of the right hand side of (4.13) using the fact $f(y_t|\boldsymbol{\theta}) = \hat{f}(y_t/\sqrt{h_t}|\boldsymbol{\theta}) \cdot |h_t|^{-1}$, where $h_t$ can be calculated by plugging in the posterior means of $\boldsymbol{\theta}$.

The next step is to calculate $\log\{f(\Theta|\boldsymbol{y})\}$ and there are two ways to achieve this. One is based on kernel smoothing as introduced in Chib and Greenburg (1998), and the other is based on extra MCMC iterations which is illustrated in Chib and Jeliazkov (2001). We will adopt the latter approach because the accuracy of kernel smoothing can be potential issue especially in cases of high dimension of $\Theta$. Therefore we have:

$$\hat{f}(\boldsymbol{\theta}^*|\boldsymbol{y}) = \frac{G_1^{-1} \sum_{g=1}^{G_1} \alpha(\boldsymbol{\theta}^g, \boldsymbol{\theta}^*|\boldsymbol{y}) q(\boldsymbol{\theta}^*|\boldsymbol{\theta}^g, \boldsymbol{y})}{G_2^{-1} \sum_{j=1}^{G_2} \alpha(\boldsymbol{\theta}^*, \boldsymbol{\theta}^j|\boldsymbol{y})}$$

where $\boldsymbol{\theta}^*$ is the full MCMC posterior mean of $\boldsymbol{\theta}$. $G_1$ is the number iterations from the original MCMC sampling where we calculate the probability of accepting $\boldsymbol{\theta}^*$ (i.e. $\alpha(\boldsymbol{\theta}^g, \boldsymbol{\theta}^* | \nu^*, \psi^*, \boldsymbol{y})$) given the current location of $\boldsymbol{\theta}$ at the $g^{th}$ draw and $q(\boldsymbol{\theta}^* | \boldsymbol{\theta}^g, \nu^*, \psi^*, \boldsymbol{y})$ is the proposal density of $\boldsymbol{\theta^*}$. $G_2$ is the number of iterations of a reduced MCMC where we fix $\boldsymbol{\theta}$ at their posterior means. In this reduced MCMC run, we propose a new value of $\boldsymbol{\theta}$ (i.e. $\boldsymbol{\theta}^j$) at each iteration and calculate the accepting rate without actually accepting $\boldsymbol{\theta}^j$.

For SV type models, we can calculate $\log\{f(\boldsymbol{y}|\Theta)\}$ in a similar way as (4.13). In SV model $\Theta$ is $(\sigma_\eta^2, \phi)$. To calculate $\log\{f(\Theta|\boldsymbol{y})\}$, we first apply kernel smoothing and use the MCMC draws of $\phi$ to derive $f(\phi^*|\boldsymbol{y})$. Then we conduct a reduced MCMC run to calculate $f(\sigma_\eta^{2*}|\psi^*, \boldsymbol{y})$. In the reduced MCMC run, $\phi$ is fixed at the posterior mean and $\sigma_\eta^2$ has closed form of full conditional distribution (inverse Gamma).

We need one more step to make SV and GARCH comparable because the two type of models use different response. Denoting the response for GACRH as $\boldsymbol{y}_{\mathrm{G}}$ and the response for SV $\boldsymbol{y}_{\mathrm{S}}$, we know that $y_{\mathrm{S},t} = \log(y_{\mathrm{G},t}^2 + c)$ for $t = 1, \cdots, n$. We further denote the marginal density of $\boldsymbol{y}_{\mathrm{G}}$ and $\boldsymbol{y}_{\mathrm{S}}$ as $f_{\boldsymbol{y}_{\mathrm{G}}}(\cdot)$ and $f_{\boldsymbol{y}_{\mathrm{S}}}(\cdot)$. We know that we should compare $f_{\boldsymbol{y}_{\mathrm{S}}}(\boldsymbol{y}_{\mathrm{S}}|\mathrm{SV})$ to $f_{\boldsymbol{y}_{\mathrm{S}}}(\boldsymbol{y}_{\mathrm{S}}|\mathrm{GARCH})$ rather than compare $f_{\boldsymbol{y}_{\mathrm{S}}}(\boldsymbol{y}_{\mathrm{S}}|\mathrm{SV})$ to $f_{\boldsymbol{y}_{\mathrm{G}}}(\boldsymbol{y}_{\mathrm{G}}|\mathrm{GARCH})$. Because the derivative of transform function is 0 when the return is 0, we can only compare SV and GARCH by marginal likelihood of non-zero returns. We denote the non-zero returns in $\boldsymbol{y}_{\mathrm{G}}$ as $\boldsymbol{y}_{\mathrm{G}}^+$ and the corresponding transformed non-zero return as $\boldsymbol{y}_{\mathrm{S}}^+$. It should be noted that the comparison between SV and GARCH is essentially the comparison between $f_{\boldsymbol{y}_{\mathrm{S}}}(\boldsymbol{y}_{\mathrm{S}}^+|\mathrm{SV})$ and $f_{\boldsymbol{y}_{\mathrm{S}}}(\boldsymbol{y}_{\mathrm{S}}^+|\mathrm{GARCH})$. We need to utilize (4.14) twice to derive both $f_{\boldsymbol{y}_{\mathrm{G}}}(\boldsymbol{y}_{\mathrm{G}}^+|\mathrm{GARCH})$ and $f_{\boldsymbol{y}_{\mathrm{G}}}(-\boldsymbol{y}_{\mathrm{G}}^+|\mathrm{GARCH})$ to make GARCH comparable to SV. Finally we have:

$$\log[f_{\boldsymbol{y}_{\mathrm{S}}}(\boldsymbol{y}_{\mathrm{S}}^+|\mathrm{GARCH})] = \log[f_{\boldsymbol{y}_{\mathrm{G}}}(\boldsymbol{y}_{\mathrm{G}}^+|\mathrm{GARCH})] + \log[f_{\boldsymbol{y}_{\mathrm{G}}}(-\boldsymbol{y}_{\mathrm{G}}^+|\mathrm{GARCH})] - \sum_{y_{\mathrm{G},t} \neq 0} \log|\frac{2 \cdot y_{\mathrm{G},t}}{y_{\mathrm{G},t}^2 + c}|$$

Therefore, we use different marginal likelihood to evaluate models. For the comparison within GARCH type, we use $\log[f_{\boldsymbol{y}_{\mathrm{G}}}(\boldsymbol{y}_{\mathrm{G}}|\mathrm{GARCH})]$ and we call this "log of marginal likelihood of return", or log(MLR). For comparison between GARCH and SV, we have to use $\log[f_{\boldsymbol{y}_{\mathrm{S}}}(\boldsymbol{y}_{\mathrm{S}}^{+}|\mathrm{GARCH})]$ and $\log[f_{\boldsymbol{y}_{\mathrm{S}}}(\boldsymbol{y}_{\mathrm{S}}^{+}|\mathrm{SV})]$ and we call them "log of marginal likelihood of log-squared return", or log(MLL).

In summary, the complete GARCH model fitting includes $50,000$ iterations of full MCMC sampling ($10,000$ iterations for burn-in followed by $40,000$ for parameter inference) and $5,000$ iterations of reduced MCMC run for accomplishing marginal likelihood calculation.

## 4.7 Simulation Studies

### 4.7.1 Simulation setting

To study how the models perform in different scenarios, we generate data sets by GARCH or SV model using Student-t, DPM or WDPM error terms, and apply the candidate models to these data sets to perform model comparison. The generation of daily returns from G_WDPM includes the following steps:

**Step 1** Independently generate the $Q$ candidate prior measures $G_1^c(\cdot), \cdots, G_Q^c(\cdot)$ from $\mathrm{DP}(\zeta, G_0)$ using "stick-breaking" technique, where $G_0$ is $\mathrm{N}(\mu_0, (1-\alpha)\sigma_e^2)$.

**Step 2** Sample $R_{1-p}, R_{2-p}, \cdots, R_0$ from multivariate Student-t distribution with degree of freedom 3.

**Step 3** Derive weight vector $\boldsymbol{\pi_t}$ for each observation based on the given weight function and covariate vector $\boldsymbol{x_t}$. Rescale the weight vectors to let the components sum up to 1 if necessary.

**Step 4** Let each observation randomly select its prior $G_t(\cdot)$ for $\mu_t$ from the $Q$ candidate priors using the weight vector $\boldsymbol{\pi_t}$.

**Step 5** Sample $\mu_t \sim G_t(\cdot)$.

**Step 6** Sample $\varepsilon_t \sim \mathrm{N}(\mu_t, \alpha\sigma_e^2)$.

**Step 7** Derive $h_t = \alpha_0 + \alpha_1 y_{t-1}^2 + \beta h_{t-1}$.

**Step 8** $R_t = \varepsilon_t h_t^{1/2}$.

**Step 9** Repeat step 3-8 for $t = 1, \cdots, n$.

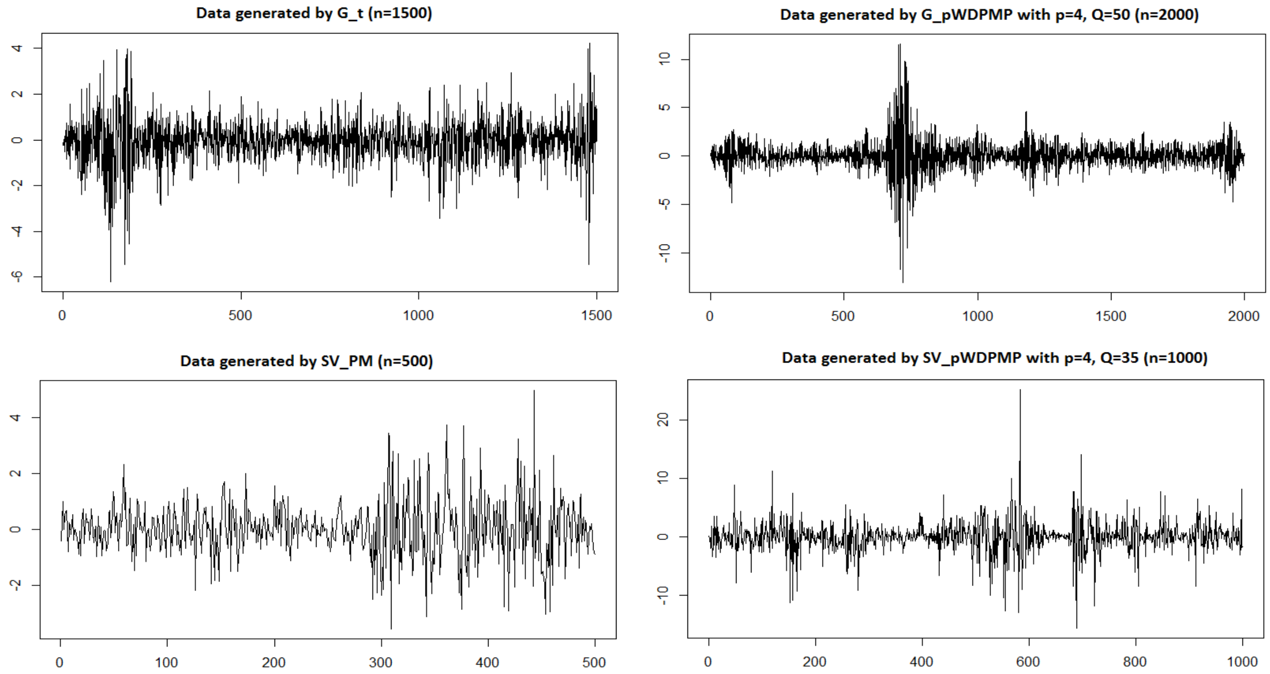
The parameters are specified as $\alpha_0 = 0.03$, $\alpha_1 = 0.1$, $\beta = 0.82$, $h_0 = 0.03$, $\mu_0 = 0$, $\nu = 7$, $\sigma_e^2 = 1.5$, $\zeta = 6$ and $\psi = 2$. We can simplified the above procedures for DPM or Student-t models. In DPM, step 1-4 can be waived and we only have to generate one prior measure from $\mathrm{DP}(\zeta, G_0)$. For Student-t, we can set $G_t(\cdot) \equiv t_\nu(\cdot)$ for every $t$ and directly start from step 5.

### 4.7.2 Model Comparison

We have generated and fitted 76,600 data sets with different sample size $n$ in simulation studies. These can be decomposed into four parts: The first is to compare G_WDPM to G_t and G_DPM to investigate the necessity of adding weighted approach. We set $n = 1000$ for this part. The second will be the comparison between G_WDPMP and GARCH models with other weight functions in four scenarios where n are correspondingly set as 500, 1000, 1500 and 2000. In the third part we compare GARCH models to SV models at $n = 500$ and $n = 1000$. The generation of data sets from SV models will follow the approach specified in Section 3.7.1. All of the first three parts use marginal likelihood for model performance. In the last part, we set $n = 500$ and display the examples of using BTR to compare GARCH models. Utilizing C++ codes and parallel computation on more than 150 cores of Intel Xeon E5-2687 3.10 GHz CPU, the total actual CPU time spent on these 76,600 fittings is 247 hours.

In Figure 4.2 we display two examples of data sets generated from GARCH type models

Figure 4.2: Two examples of data sets generated from GARCH type models and two from SV type models; The x-axis is index of trading days and the y-axis is log return of daily prices.

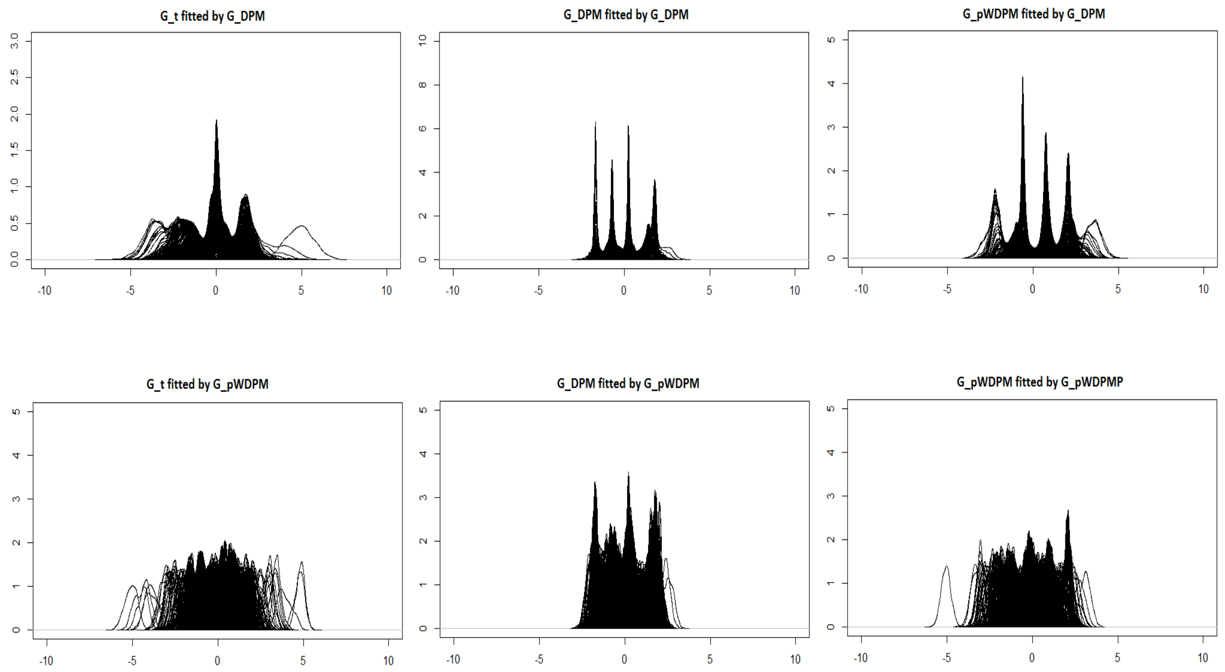


and two from SV type models. We can see in both types that the simulated daily returns contain the pattern of autocorrelated volatilities: the fluctuation lasts high for multiple days once it becomes large and it lasts low for multiple days once it becomes small.

We have also displayed several examples of the clustering check on the $\mu_t$'s as introduced in (4.3). The results are plotted in Figure 4.3. We generate one data set from G_t, G_DPM and G_WDPMP with p=3, Q=27 correspondingly, and fit the three data sets by G_DPM and G_WDPMP with p=3, Q=27. It should be noted that there is no $\mu_t$ in Student-t model fitting, therefore we only fit the three data sets by the two nonparametric models.

We can see that the existence of multiple clusters are more apparent when the data sets are fitted by G_DPM compared to fitted by G_WDPMP. The results provided by G_WDPMP fittings look more reasonable because the separation among different clusters in G_t data is vague and we expect it to be vague since Student-t model does not contain multiple clusters

Figure 4.3: Posterior density curves of $\mu_t$ $(t = 1, \cdots, n)$ for simulated data sets; We set n=1000. There are n density curves in each of the six panels; Each curve is plotted based on $40,000$ MCMC draws of $\mu_t$; The x-axis is the range of $\mu_t$ and y-axis stands for the kernel density estimated value; There are three data sets generated to produce these panels



of $\mu_t$. In contrast, the fittings of G_DPM seem to be not able to reflect the nature of data because even fitting Student-t data produce clear evidence of clusters. The fittings provided by G WDPMP model is also more reasonable in the sense that the signal of existence of different clusters in G_WDPMP data is weaker than that in G_DPM data and stronger than that in G_t data. This can be clearly seen by comparing the three panels of the second row in Figure 4.3. This result is consistent with the findings in Chapter 3 that "WDPM can be viewed as the balance between parametric model and DPM".

#### 4.7.2.1 Comparison among G_WDPM, G_t, and G_DPM

To compare weighted GARCH to DPM or Student-t model, we adopt six models as true models to generate data: Student-t, DPM, WDPM with $p = 1$ to $p = 4$. Note that we set $Q$ always equal to $3^p$ in this part of simulation because the goal is to compare weighted approach to DPM and Student-t rather than to investigate how $Q$ affect weighted approach. Each of the six true models generates 200 data sets with sample size $n = 1000$, therefore we have 1200 data sets in total. Each of these 1200 data sets is fitted by the six models listed above and our focus is on whether true models are always best explained by themselves. Table 4.2 displays the related results. In each cell of this table, based on 200 repetitions of certain true model fitted by certain fit model, we include the average of log(MLR) and $p$-value of Wilcoxon signed-rank test (produced by comparing to the best model of the same row) in each cell of this table. There is no $p$-value for the best model of each row and instead such model is labeled out.

Table 4.2: Model comparison based on simulation with $n = 1000$; All models are GARCH type so we ignore the GARCH in the model names; Every true model generates 200 samples and each sample is fitted by 6 fit models; We include the average log(MLR) and $p$-value of Wilcoxon signed-rank test (produced by comparing to the best model of the same row) in each cell of this table; There is no $p$-value for the best model of each row and instead such model is labeled red; The six candidate models are Student-t, DPM, and WDPMP with $p = 1, 2, 3, 4$, which are also the fit models.

| Fit Model<br>True Model | Student-t | DPM | WDPMP<br>$p = 1$ | WDPMP<br>$p = 2$ | WDPMP<br>$p = 3$ | WDPMP<br>$p = 4$ |
|---|---|---|---|---|---|---|
| Student-t | -1409.77<br>best model | -1574.23<br>(1.097e-14) | -1498.27<br>(7.136e-13) | -1556.70<br>(0) | -1442.46<br>(3.356e-08) | -1439.81<br>(3.168e-07) |
| DPM | -1401.36<br>(5.601e-13) | -1398.25<br>(2.298e-16) | -1253.55<br>(3.697e-08) | -1291.47<br>(6.117e-07) | -1216.49<br>best model | -1235.20<br>(9.194e-05) |
| WDPMP<br>$p = 1$ | -1450.19<br>(0) | -1426.65<br>(0) | -1283.90<br>(8.756e-07) | -1263.57<br>(2.541e-07) | -1227.21<br>best model | -1274.44<br>(1.363e-07) |
| WDPMP<br>$p = 2$ | -1417.69<br>(0) | -1481.23<br>(0) | -1275.58<br>(8.005e-06) | -1252.10<br>(0.2171) | -1231.45<br>best model | -1283.03<br>(9.337e-06) |
| WDPMP<br>$p = 3$ | -1461.31<br>(0) | -1375.74<br>(0) | -1306.74<br>(6.653e-10) | -1290.53<br>(2.904e-06) | -1249.11<br>best model | -1286.07<br>(1.391e-05) |
| WDPMP<br>$p = 4$ | -1401.85<br>(0) | -1417.96<br>(0) | -1303.87<br>(7.134e-09) | -1260.02<br>(0.03537) | -1235.03<br>best model | -1299.84<br>(5.904e-10) |

We can see that all of the models are best fitted by G_WDPMP with $p = 3$ except that Student-t are best fitted by itself. Therefore we can conclude that WDPMP with $p = 3$ does display model flexibility which is not contained by the other five models when we are using nonparametric approach to generate data. When the true model is Student-t, on the other hand, weighted approaches do not show such model flexibility since G_t performs better than the five nonparametric approaches. However, since we do not know true model, WDPMP with $p = 3$ will be more useful and appropriate model in general case.

#### 4.7.2.2 Comparison between WDPM Garch with different weights

In this part, we focus on the comparison among different weight functions. The five candidate weight functions are WDPMP, EWDPM, WDPMG, WDPME and WDPMH. We apply these weighted approaches to GARCH and each model generates 200 data sets. Each of the generated 1000 data sets is fitted by the five weighted approaches as well. Table 4.3-4.6 display the results of such model comparison under different choices of $n$, $p$ and $Q$, and such results can be interpreted in the same way as we do for Table 4.2.

Table 4.3: Comparison among different weight specifications using $n = 500$, $p = 3$ and $Q = 27$; All models are GARCH type so we ignore the GARCH in the model names; The five candidate models are: WDPMP, EWDPM, WDPMG, WDPME, WDPMH; Every true model generates 200 samples and each sample is fitted by the five candidate models models; We include the average log(MLR) and $p$-value of Wilcoxon signed-rank test (produced by comparing to the best model of the same row) in each cell of this table; There is no $p$-value for the best model of each row and instead such model is labeled red.

| n=500, p=3, Q=27 | | | | | |
|---|---|---|---|---|---|
| Fit Model<br>True Model | WDPMP | EWDPM | WDPMG | WDPME | WDPMH |
| WDPMP | -577.63<br>best model | -617.21<br>(0) | -609.24<br>(4.118e-12) | -635.89<br>(0) | -594.60<br>(7.069e-08) |
| EWDPM | -713.56<br>best model | -759.42<br>(3.087e-14) | -736.28<br>(5.229e-08) | -769.10<br>(0) | -774.05<br>(0) |
| WDPMG | -653.97<br>best model | -694.40<br>(1.269e-10) | -701.25<br>(7.947e-13) | -703.22<br>(0) | -692.70<br>(3.151e-12) |
| WDPME | -769.03<br>(0) | -712.28<br>(0.2515) | -740.28<br>(6.375e-11) | -709.93<br>best model | -732.87<br>(1.182e-10) |
| WDPMH | -617.69<br>(7.855e-16) | -648.53<br>(0) | -594.75<br>(0.1870) | -603.20<br>(5.096e-10) | -584.39<br>best model |

In the case of $n = 500$, $p = 3$ and $Q = 27$ (Table 4.3), We can see that three of out five true models are best fitted by G_WDPMP while G_EWDPM and G_WDPMH are best fitted by themselves. Therefore we can conclude that WDPMP is the most flexible weighted approach among the candidates in the case of $n = 500$, $p = 3$ and $Q = 27$.

Table 4.4: Comparison among different weight specifications using $n = 1000$, $p = 3$ and $Q = 27$; The way to construct this table is the same as Table 4.3.

| n=1000, p=3, Q=27 | | | | | |
|---|---|---|---|---|---|
| Fit Model<br>True Model | WDPMP | EWDPM | WDPMG | WDPME | WDPMH |
| WDPMP | -1265.99<br>best model | -1345.73<br>(0) | -1295.02<br>(3.189e-12) | -1291.61<br>(5.003e-12) | -1286.64<br>(7.946e-11) |
| EWDPM | -1475.79<br>best model | -1503.22<br>(3.425e-06) | -1483.91<br>(0.4953) | -1519.42<br>(1.285e-13) | -1540.13<br>(0) |
| WDPMG | -1319.74<br>best model | -1385.46<br>(7.535e-15) | -1372.25<br>(6.194e-12) | -1368.85<br>(1.072e-11) | -1350.59<br>(3.314e-08) |
| WDPME | -1503.54<br>(2.096e-06) | -1517.08<br>(8.658e-12) | -1492.71<br>(0.1663) | -1475.30<br>best model | -1537.84<br>(0) |
| WDPMH | -1195.47<br>best model | -1317.52<br>(0) | -1209.55<br>(0.07523) | -1248.41<br>(5.095e-13) | -1212.61<br>(7.734e-04) |

When we increase the sample size to 1000 and use the same $p$ and $Q$, we observe more convincing evidence in Table 4.4 that WDPMP is the best weight function compared to others. This is because WDPMP becomes the best model even when the true model is G_WDPMH.

Table 4.5: Comparison among different weight specifications using $n = 1500$, $p = 4$ and $Q = 50$. The way to construct this table is the same as Table 4.3.

| n=1500, p=4, Q=50 | | | | | |
|---|---|---|---|---|---|
| Fit Model<br>True Model | WDPMP | EWDPM | WDPMG | WDPME | WDPMH |
| WDPMP | -1910.26<br>best model | -2088.37<br>(0) | -1977.83<br>(3.390e-11) | -1987.12<br>(4.572e-12) | -1950.49<br>(9.791e-06) |
| EWDPM | -2091.57<br>best model | -2179.80<br>(1.958e-15) | -2105.93<br>(0.3981) | -2098.85<br>(0.6233) | -2143.64<br>(8.764e-11) |
| WDPMG | -1996.53<br>best model | -2037.59<br>(1.727e-07) | -2044.16<br>(5.830e-09) | -2011.28<br>(0.00359) | -2062.79<br>(2.519e-12) |
| WDPME | -2176.18<br>(0.3915) | -2253.01<br>(0) | -2212.09<br>(8.363e-09) | -2164.52<br>best model | -2249.87<br>(6.538e-13) |
| WDPMH | -1822.39<br>best model | -2037.65<br>(0) | -1904.77<br>(5.309e-16) | -1885.37<br>(7.584e-09) | -1871.09<br>(3.469e-08) |

In Table 4.5 we display the weight function comparison in two settings ($n = 1500, p = 4, Q = 50$ and $n = 2000, p = 4, Q = 81$). It is learned from Chapter 3 that the optimal choice of $Q$ increases as sample size gets larger. Based on this and the detail of their findings, we consider $Q = 50$ as a proper choice for $n = 1500$. We use $p = 4$ because it is necessary that $p > 3$ to achieve $Q > 27$. We see that WDPMP is the best model in all true model cases except when the true model is WDPME, which is similar results as shown in n=1000 case. The two differences compared to $n = 1000$ case are: on one hand, WDPMP does not deviate significantly ($p$-value is 0.3915) from the best model when the true model is WDPME at n=1500; on the other hand, WDPMP is not significantly ($p$-value is 0.6233) better than EWDPM when the true model is EWDPM at $n = 1500$.

Table 4.6: Comparison among different weight specifications using $n = 2000$, $p = 4$ and $Q = 81$; The way to construct this table is the same as Table 4.3.

| n=2000, p=4, Q=81 | | | | | |
|---|---|---|---|---|---|
| Fit Model<br>True Model | WDPMP | EWDPM | WDPMG | WDPME | WDPMH |
| WDPMP | -2597.82<br>best model | -2793.90<br>(0) | -2671.79<br>(5.638e-11) | -2630.82<br>(3.719e-05) | -2614.05<br>(0.2254) |
| EWDPM | -2907.51<br>(0.7550) | -2944.48<br>(6.432e-08) | -2979.42<br>(1.965e-13) | -2985.62<br>(0) | -2881.13<br>best model |
| WDPMG | -2553.08<br>best model | -2611.76<br>(5.993e-07) | -2625.33<br>(2.284e-09) | -2640.06<br>(1.729e-11) | -2629.17<br>(6.060e-09) |
| WDPME | -3078.54<br>best model | -3249.36<br>(0) | -3157.83<br>(4.789e-09) | -3167.92<br>(5.391e-10) | -3135.40<br>(9.487e-06) |
| WDPMH | -2457.76<br>best model | -2761.33<br>(0) | -2493.89<br>(0.1697) | -2531.14<br>(2.005e-10) | -2501.82<br>(7.838e-06) |

In the last case the sample size is increased to 2000 and we choose a larger $Q(=81)$ for this sample size and use $p = 4$ to perform model comparison. We can find in Table 4.6 that all of the models are best fitted by WDPMP except using EWDPM to generate data, where the best model (WDPMH) is not significantly ($p$-value is 0.7550) better than WDPMP. In summary, we can see that WDPMP contains the extra model flexibility in all of the cases compared to the other four weighted approaches.

#### 4.7.2.3 Comparison between Garch and SV

In this part we compare GARCH models to SV models in two cases ($n = 500$ and $n = 1000$). For $n = 500$, we have six competitors (three from each type): GARCH Student-t, G_WDPMP with $p = 3$ and $Q = 27$, G_WDPMP with $p = 4$ and $Q = 35$, SV_PM, SV_WDPMP with $p = 3$ and $Q = 27$, SV_WDPMP with $p = 4$ and $Q = 35$. For $n = 1000$ the only difference is that two models (G_WDPMP with $p = 4$ and $Q = 35$, SV_WDPMP with $p = 4$ and $Q = 35$) are replaced by G_WDPMP with $p = 4$ and $Q = 50$ and SV_WDPMP with $p = 4$ and $Q = 50$. We choose these settings of $p$ and $Q$ because we view them as the choices close to optimal settings for the particular sample sizes (500 and 1000) based the results of Section 3.7 and 4.7.2.1. In both cases, each competitor generate 200 data sets which are fitted by itself and the other five models. The results are given in Table 4.7 and 4.8.

Table 4.7: Comparison between GARCH and SV models; For $n = 500$ case, the six candidate models are GARCH Student-t, G_WDPMP with $p = 3$ and $Q = 27$, G_WDPMP with $p = 4$ and $Q = 35$, SV_PM, SV_WDPMP with $p = 3$ and $Q = 27$, SV_WDPMP with $p = 4$ and $Q = 35$; Every model generates 200 samples and each sample is fitted by the six candidate models; We include the average of log(MLL) and $p$-value of Wilcoxon signed-rank test (produced by comparing to the best model of the same row) in each cell of this table; There is no $p$-value for the best model of each row and instead such model is labeled red.

| n=500 | | | | | | |
|---|---|---|---|---|---|---|
| Fit Model<br>True Model | GARCH<br>Student-t | G_WDPMP<br>$p = 3$,$Q = 27$ | G_WDPMP<br>$p = 4$,$Q = 35$ | SV_PM | SV_WDPMP<br>$p = 3$,$Q = 27$ | SV_WDPMP<br>$p = 4$,$Q = 35$ |
| GARCH<br>Student-t | -1135.87<br>(0.1531) | -1114.22<br>best model | -1129.15<br>(0.6270) | 1171.39<br>(7.361e-10) | -1129.56<br>(0.5477) | -1157.80<br>(3.629e-07) |
| G_WDPMP<br>$p = 3$,$Q = 27$ | -1109.52<br>(0) | -1073.14<br>best model | -1094.62<br>(3.463e-05) | -1195.18<br>(0) | -1106.78<br>(8.925e-08) | -1107.49<br>(4.151e-09) |
| G_WDPMP<br>$p = 4$,$Q = 35$ | -1120.27<br>(7.580e-12) | -1082.99<br>best model | -1087.05<br>(0.8758) | -1171.35<br>(0) | -1109.89<br>(9.481e-13) | -1115.43<br>(0) |
| SV_PM | -1169.67<br>(0) | -1135.12<br>(6.705e-13) | -1144.34<br>(0) | -1109.21<br>best model | -1120.77<br>(1.285e-13) | -1123.01<br>(7.070e-15) |
| SV_WDPMP<br>$p = 3$,$Q = 27$ | -1037.95<br>(0) | -1022.80<br>(0) | -1019.76<br>(0) | -1012.52<br>(0) | -989.36<br>best model | -996.17<br>(5.063e-08) |
| SV_WDPMP<br>$p = 4$,$Q = 35$ | -1142.65<br>(0) | -1049.67<br>(0) | -1046.15<br>(5.789e-11) | -1102.29<br>(0) | -1035.02<br>best model | -1041.39<br>(9.174e-06) |

It should be noted that in the previous two parts we use log(MLR) to evaluate models while we use log(MLL) in this part. We see that GARCH models perform generally better than SV models when the true model is in GARCH type and vice versa. All of the GARCH models are best fitted by G_WDPMP with $p = 3$ and $Q = 27$ at n=500 while the true

GARCH models are best fitted by themselves at $n = 1000$. As for SV type true models, parametric model are best fitted by itself and weighted DPM models are best fitted by SV_WDPMP with $p = 3$ and $Q = 27$ at both $n = 500$ and $n = 1000$.

Table 4.8: Comparison between GARCH and SV models; This table is constructed in the same way as Table 4.7 except that for $n = 1000$ case two models (G_WDPMP with $p = 4$ and $Q = 35$, SV_WDPMP with $p = 4$ and $Q = 35$) are replaced by G_WDPMP with $p = 4$ and $Q = 50$ and SV_WDPMP with $p = 4$ and $Q = 50$.

| n=1000 | | | | | | |
|---|---|---|---|---|---|---|
| Fit Model<br>True Model | GARCH<br>Student-t | G_WDPMP<br>$p = 3,Q = 27$ | G_WDPMP<br>$p = 4,Q = 35$ | SV_PM | SV_WDPMP<br>$p = 3,Q = 27$ | SV_WDPMP<br>$p = 4,Q = 35$ |
| GARCH<br>Student-t | -2109.77<br>best model | -2142.46<br>(3.356e-08) | 2135.29<br>(1.501e-06) | -2195.00<br>(8.364e-13) | -2193.12<br>(5.398e-12) | -2190.81<br>(4.791e-12) |
| G_WDPMP<br>$p = 3,Q = 27$ | -2261.31<br>(0) | -2049.11<br>best model | -2075.82<br>(6.703e-07) | -2205.72<br>(0) | -2159.24<br>(8.001e-13) | -2160.57<br>(2.622e-15) |
| G_WDPMP<br>$p = 4,Q = 35$ | -2193.25<br>(0) | -2057.80<br>(0.00257) | -2019.47<br>best model | -2208.93<br>(0) | -2191.12<br>(0) | -2201.65<br>(0) |
| SV_PM | 2317.40<br>(0) | -2322.89<br>(0) | -2311.31<br>(3.126e-16) | -2282.47<br>best model | -2289.41<br>(7.799e-06) | -2393.30<br>(2.734e-10) |
| SV_WDPMP<br>$p = 3,Q = 27$ | -2112.95<br>(0) | -2048.33<br>(7.076e-12) | -2061.09<br>(5.190e-16) | -2069.81<br>(0) | -2020.57<br>best model | -2026.94<br>(7.538e-05) |
| SV_WDPMP<br>$p = 4,Q = 35$ | -2264.39<br>(0) | -2102.92<br>(0) | -2096.35<br>(0) | -2093.35<br>(0) | -2058.45<br>best model | -2064.62<br>(0.02783) |

Based on these results, we can conclude that there is no clear evidence about which type of model is better than the other type in terms of marginal likelihood, because the advantage of accuracy depends on what true model is. However, as we illustrated in Section 4.5, the congenital advantage of GARCH is that we can utilize it to do prediction on the stock return directly rather than only on the square of return.

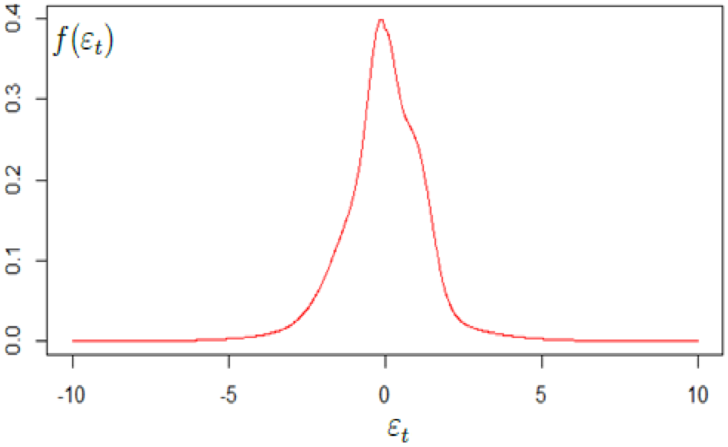
#### 4.7.2.4 Comparison with Garch models based on BTR

The last part of simulation is to display the examples of comparing GARCH models based on BTR. Although parallel computation can greatly improve the efficiency, we need to focus on only one type of true model and use $n_p = 500$ to achieve sufficient number of repetitions to produce robust model comparison conclusion, because producing one BTR takes $n_p/10$ model fittings. The candidate fit models are G_t, G_DPM, G_WDPMP, G_EWDPM, G_WDPMG, G_WDPME and G_WDPMH. For the weighted approaches, we use $p = 3$, $Q = 27$ because the sample size of model fitting in each decision-making-period is $n_0 = 500$ as well. Since we will use only one type of true model, to make the comparison fair, we do not use any of the fit models as true model. Instead, we generate the error term $\varepsilon_t$ using the following approach:

$$\begin{aligned}
&\Pr(\varepsilon_t = \varepsilon_{t(j)}) = p_j, \quad j = 1, \cdots, 5, \\
&(p_{1,2}, p_3, p_4, p_5) = (0.24, 0.37, 0.2, 0.095, 0.095), \\
&\varepsilon_{t(1)} \sim \mathrm{N}(-0.08, 0.4^2), \quad \varepsilon_{t(2)} \sim \mathrm{N}(-0.5, 1), \quad \varepsilon_{t(3)} \sim \mathrm{N}(1, 0.5^2), \\
&\varepsilon_{t(4)} \sim \mathrm{Gamma}(2, 1), \quad -\varepsilon_{t(5)} \sim \mathrm{Gamma}(2, 1),
\end{aligned} \tag{4.15}$$

and generate volatility vector $\boldsymbol{h}$ in the same way as other GARCH models do. Figure 4.4 provides the probability density curve of the error term $\varepsilon_t$ generated from this approach.

Figure 4.4: The curve of actual probability density $f(\varepsilon_t)$ which is used to generate $\varepsilon_t$ for BTR comparison in simulation study.



We generate 100 data sets from this true model with $n_0 = n_p = 500$. There are $500/10 = 50$ "decision-making-periods" for each data set and during each of these periods we include 500 previous observations to fit model. Therefore there are 1000 observations actually generated from (4.15) for data set although only 500 of them will be used for checking BTR.

In our simulation study, we assume commission fee $C_0$ is 0.01% of the original amount of money $\mathrm{Mn}_0$. We also need to assume an open price we see at the end of a "decision-making-period" is always the same as its previous trading day's market closing price. Based on this assumption, it is not necessary to worry about the difference in BTRs brought the initial status of holding or not, since such difference is caused by the gap between the open price day 1 and closing price of day 0. Therefore, we will only focus on the case of "start with not holding" for simulation study.

We are interested to compare the returns brought by using models to the returns achieved by the simplest strategy: buy the stock at the starting point and keep holding the stock till the end. The return associated to this strategy can be given as $(P_{\text{last}} - 1.01 \times P_{\text{start}})/P_{\text{start}} \times 100\%$ (denote this as "overall return"). Here $P_{\text{start}}$ and $P_{\text{last}}$ are the real value prices simulated for the first and last trading day.

Table 4.9: BTR comparison; 100 data sets are generated and each data set is analyzed by the seven candidate models: G_t, G_DPM, G_WDPMP, G_EWDPM, G_WDPMG, G_WDPME and G_WDPMH. For the weighted approaches, we use $p = 3$, $Q = 27$ for $n_0 = 500$. Average BTR, standard error and VaR are calculated based on the 100 repetitions. The 5% VaR is defined to be $\Pr(\text{BTR} < -\text{VaR}) = 0.05$, and we use the negative of the average of the $5^{th}$ and $6^{th}$ lowest BTR among the 100 repetitions as 5% VaR. The number of observations used for checking BTR for each data set is $n_p = 500$. The BTRs produced by different models are compared to the "simple" strategy: buy the stock at the starting point and keep holding the stock till the end.

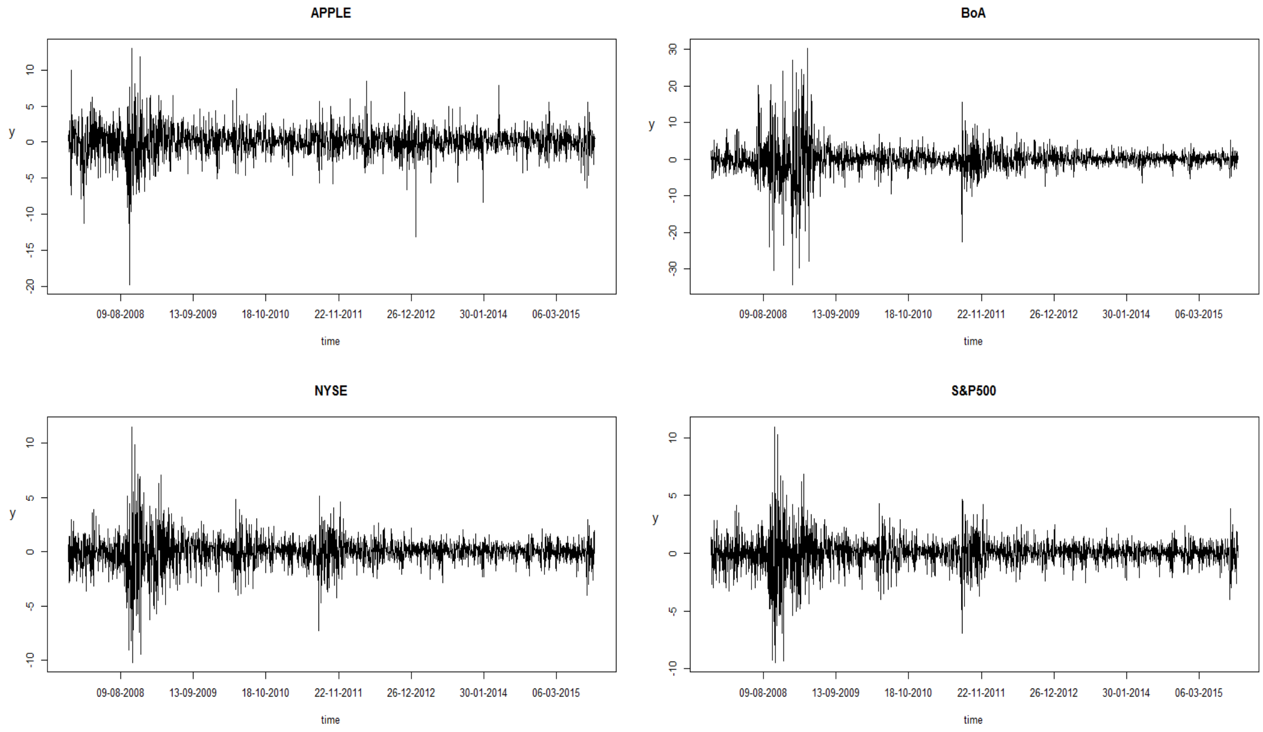
| | Average BTR | Standard Error | Value at Risk |
|---|---|---|---|
| Simple | 2.3% | 51.8% | 57.1% |
| G_t | 4.5% | 29.4% | 27.5% |
| G_DPM | 4.9% | 45.2% | 50.8% |
| G_WDPMP | 7.6% | 27.0% | 29.2% |
| G_EWDPM | -1.5% | 28.1% | 37.7% |
| G_WDPMG | 8.8% | 25.9% | 20.7% |
| G_WDPME | 5.9% | 22.5% | 24.3% |
| G_WDPMH | 2.1% | 42.8% | 4.1% |

In Table 4.9 we display the results of BTR comparison for the seven candidate models. We can see that the average overall return of the 100 simulated data sets is 2.3%. G_EWDPM and G_WDPMH produce average returns lower than 2.3% while the other five models produce average BTR higher than 2.3%. We also calculate the standard error and the 5% Value at Risk (VaR) of each model's BTR based on the 100 repetitions. The 5% VaR is defined to be $\Pr(\text{BTR} < -\text{VaR}) = 0.05$, and we use the negative of the average of the $5^{th}$ and $6^{th}$ lowest BTR among the 100 repetitions as 5% VaR. A smaller VaR means a lower downside risk. The ideal model should be the one that contains the highest BTR with the lowest standard error and VaR. Yet we find that there is no such model based on Table 4.9. In terms of BTR, G_WDPMG is the best; in terms of standard error, G_WDPME is the best; in terms of the VaR, G_WDPMG is the best. But we can draw a definite conclusion that it is necessary to use weighted GARCH models because G_WDPMP, G_WDPMG and G_WDPME produce higher average BTR and lower standard error and VaR compared to the simple strategy (keeping holding till the end), G_t and G_DPM.

## 4.8 Real Application

We obtain the daily returns of four companies (1) APPLE, (2) Bank of America (BoA), (3) NYSE index, and (4) S&P 500 index. APPLE and BoA are individual stocks with large volume and the other two are composite indices. The observations are between 10/25/2007 and 10/06/2015 and the sample size of each data set is $2,000$. Denoting "overall return" as $(P_{\text{last}} - P_{\text{start}})/P_{\text{start}} \times 100\%$, these data sets are plotted in Figure 4.5.

Figure 4.5: Plots of the daily returns of four companies (Apple, Banke of America, NYSE, and S&P); These data sets are collected from Yahoo finance; The x-axis is calendar time of trading days and the y-axis is log return of daily adjusted closing prices; The observations of adjusted closing price are made between 10/25/2007 and 10/06/2015.



We can see the pattern of inconstant variance in each of the four graphs. We also observe that the individual stock of BoA displays negative overall return (lower than $-60\%$); the individual stock of APPLE displays positive overall return (higher than $300\%$). One composite index of NYSE displays near-zero overall return (about $-0.3\%$). One composite index of S&P 500 displays positive overall return (more than 30%).

We will compare our candidate models using marginal likelihood and BTR. There are $4,326$ model fittings conducted for these comparisons. Utilizing the computation techniques introduced in Section 4.7, the total actual CPU time spent on these model fittings is 37 hours.

### 4.8.1 Marginal Likelihood Comparisons

For each data set, we will apply G_t, G_DPM, SV_PM and SV_DPM. Additionally, we consider all of the possible combinations of: two types (SV or GARCH); five weight functions (WDPMP, EWDPM, WDPMG, WDPME, WDPMH) and 8 model settings:

1: $p = 3$, $Q = 27$; 2: $p = 4$, $Q = 40$

3: $p = 4$, $Q = 50$; 4: $p = 4$, $Q = 81$

5: $p = 5$, $Q = 40$; 6: $p = 5$, $Q = 50$

7: $p = 5$, $Q = 80$; 8: $p = 5$, $Q = 100$

Therefore, there will be $2 \times 5 \times 8 = 80$ weighted models used. We apply the 84 candidate models to each data set and calculate log(MLL). We need to calculate marginal likelihood of log-squared return because we compare GARCH to SV. For every stock, we choose the best 20 models based on marginal likelihood and list them on Table 4.10.

Table 4.10: Best 20 models for APPLE, BoA, NYSE and S&P 500 data; The models are sorted by their marginal likelihood and labeled with rank; A "✓" in "S" column means SV type model and in "G" column means GARCH type; For example, the best model for APPLE data is G_WDPMP with $p = 4$ and $Q = 81$. "log(MLL)" means log of marginal likelihood of log-squared data.

| APPLE | | | | | BoA | | | | | NYSE | | | | | S&P500 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Rank | G | S | Model | log(MLL) | Rank | G | S | Model | log(MLL) | Rank | G | S | Model | log(MLL) | Rank | G | S | Model | log(MLL) |
| 1 | ✓ | | WDPMP<br>(4,81) | -4085.25 | 1 | ✓ | | WDPMH<br>(5,80) | -4044.77 | 1 | ✓ | | WDPMP<br>(5,100) | -4102.09 | 1 | ✓ | | WDPMG<br>(4,81) | -4252.91 |
| 2 | ✓ | | WDPMP<br>(5,80) | -4090.44 | 2 | ✓ | | WDPMP<br>(5,100) | -4046.02 | 2 | ✓ | | WDPMP<br>(4,81) | -4105.12 | 2 | ✓ | | WDPMH<br>(4,81) | -4254.28 |
| 3 | ✓ | | WDPMP<br>(5,100) | -4090.79 | 3 | ✓ | | WDPMH<br>(5,50) | -4052.73 | 3 | ✓ | | WDPMP<br>(5,80) | -4106.58 | 3 | ✓ | | WDPMH<br>(5,80) | -4255.06 |
| 4 | ✓ | | WDPMH<br>(5,80) | -4091.01 | 4 | ✓ | | WDPME<br>(4,50) | -4053.18 | 4 | ✓ | | WDPMP<br>(5,50) | -4106.78 | 4 | ✓ | | WDPMH<br>(5,100) | -4258.29 |
| 5 | ✓ | | WDPMP<br>(4,40) | -4091.25 | 5 | ✓ | | WDPMP<br>(5,80) | -4062.74 | 5 | ✓ | | WDPMP<br>(4,50) | -4108.21 | 5 | ✓ | | WDPME<br>(5,100) | -4265.11 |
| 6 | ✓ | | WDPMP<br>(5,40) | -4093.87 | 6 | ✓ | | WDPMH<br>(5,100) | -4064.90 | 6 | ✓ | | WDPMP<br>(4,40) | -4112.06 | 6 | | ✓ | PM | -4266.92 |
| 7 | ✓ | | WDPMP<br>(5,50) | -4094.40 | 7 | ✓ | | WDPMG<br>(4,81) | -4067.23 | 7 | ✓ | | WDPMP<br>(5,40) | -4117.24 | 7 | ✓ | | WDPMG<br>(5,80) | -4273.86 |
| 8 | ✓ | | WDPMH<br>(5,80) | -4095.63 | 8 | ✓ | | WDPMH<br>(4,40) | -4068.59 | 8 | ✓ | | WDPMP<br>(3,27) | -4118.97 | 8 | ✓ | | WDPMG<br>(5,100) | -4274.17 |
| 9 | ✓ | | WDPMG<br>(5,40) | -4098.30 | 9 | ✓ | | WDPMH<br>(4,50) | -4070.63 | 9 | ✓ | | WDPMH<br>(5,80) | -4124.30 | 9 | ✓ | | WDPMG<br>(4,50) | -4275.23 |
| 10 | ✓ | | WDPMP<br>(4,50) | -4098.64 | 10 | ✓ | | WDPMP<br>(5,50) | -4075.33 | 10 | ✓ | | WDPME<br>(5,80) | -4127.42 | 10 | ✓ | | WDPMG<br>(5,50) | -4275.29 |
| 11 | ✓ | | WDPMH<br>(4,50) | -4101.29 | 11 | ✓ | | WDPMP<br>(4,81) | -4075.51 | 11 | ✓ | | WDPME<br>(5,80) | -4130.50 | 11 | | ✓ | EWDPM<br>(5,40) | -4279.40 |
| 12 | | ✓ | WDPMP<br>(5,100) | -4103.15 | 12 | ✓ | | WDPMP<br>(4,50) | -4078.19 | 12 | ✓ | | WDPMH<br>(5,100) | -4139.87 | 12 | ✓ | | WDPMG<br>(4,40) | -4291.79 |
| 13 | ✓ | | WDPMP<br>(3,27) | -4105.17 | 13 | ✓ | | WDPME<br>(5,100) | -4083.09 | 13 | ✓ | | WDPMH<br>(5,40) | -4140.04 | 13 | ✓ | | WDPMG<br>(5,40) | -4291.95 |
| 14 | ✓ | | WDPME<br>(5,50) | -4109.02 | 14 | ✓ | | WDPMH<br>(4,81) | -4085.26 | 14 | ✓ | | WDPMH<br>(4,50) | -4141.75 | 14 | | ✓ | EWDPM<br>(5,80) | -4293.61 |
| 15 | | ✓ | WDPMP<br>(5,50) | -4109.66 | 15 | ✓ | | WDPMH<br>(5,40) | -4087.88 | 15 | ✓ | | WDPME<br>(5,50) | -4143.56 | 15 | | ✓ | EWDPM<br>(5,100) | -4294.08 |
| 16 | ✓ | | WDPMH<br>(4,81) | -4110.53 | 16 | | ✓ | WDPMH<br>(4,81) | -4095.72 | 16 | ✓ | | WDPMH<br>(4,81) | -4152.39 | 16 | | ✓ | EWDPM<br>(4,81) | -4294.97 |
| 17 | ✓ | | WDPMH<br>(5,50) | -4116.01 | 17 | ✓ | | WDPMH<br>(3,27) | -4097.43 | 17 | | ✓ | WDPMP<br>(5,100) | -4157.49 | 17 | ✓ | | WDPMH<br>(5,50) | -4295.00 |
| 18 | ✓ | | WDPMH<br>(5,100) | -4117.96 | 18 | ✓ | | WDPME<br>(5,80) | -4099.81 | 18 | ✓ | | WDPME<br>(4,81 | -4159.53 | 18 | ✓ | | WDPME<br>(3,27) | -4297.73 |
| 19 | | ✓ | WDPMP<br>(5,80) | -4124.10 | 19 | ✓ | | WDPME<br>(4,81) | -4101.69 | 19 | | ✓ | WDPMP<br>(5,50) | -4172.19 | 19 | ✓ | | WDPME<br>(4,40) | -4300.15 |
| 20 | ✓ | | WDPMH<br>(4,40) | -4127.27 | 20 | ✓ | | WDPMG<br>(5,80) | -4104.83 | 20 | ✓ | | WDPME<br>(5,40) | -4179.84 | 20 | ✓ | | WDPME<br>(5,40) | -4305.04 |

We can see that GARCH models display dominating advantage in all of the data sets compared to SV. In each of these four data sets, GARCH type produces 15 or more models among the best 20 and all of the best 10 models are from GARCH. Such advantage in BoA is the most impressive since only one SV model appears in the best 20 list. We can also find that weighted approaches perform better for these data sets because all of their best 20 models are weighted ones except SV_PM for S&P 500 data. Furthermore, we see that SV_WDPMP models are generally better than other models being applied to APPLE and

NYSE data. For S&P 500 and BoA data, on the other hand, WDPMH models are generally more accurate than WDPMP models.

### 4.8.2 Parameter Estimates and Cluster Visualization

In Table 4.11 we provide the summary of the parameter estimates and inefficiency scores of MCMC draws in real data application. The four cases are the four data sets fitted by their best model based on the marginal likelihood comparison in the previous section. In the "Est." columns, each point estimate is a posterior mean of 4000 MCMC draws, and each 95% credit interval is specified by the $101^{st}$ and $3900^{th}$ smallest values. In the "Ineff." columns, we provide inefficiency score $1 + 2\sum_{l=1}^{K} \rho(l)$ for each parameter's MCMC draws.

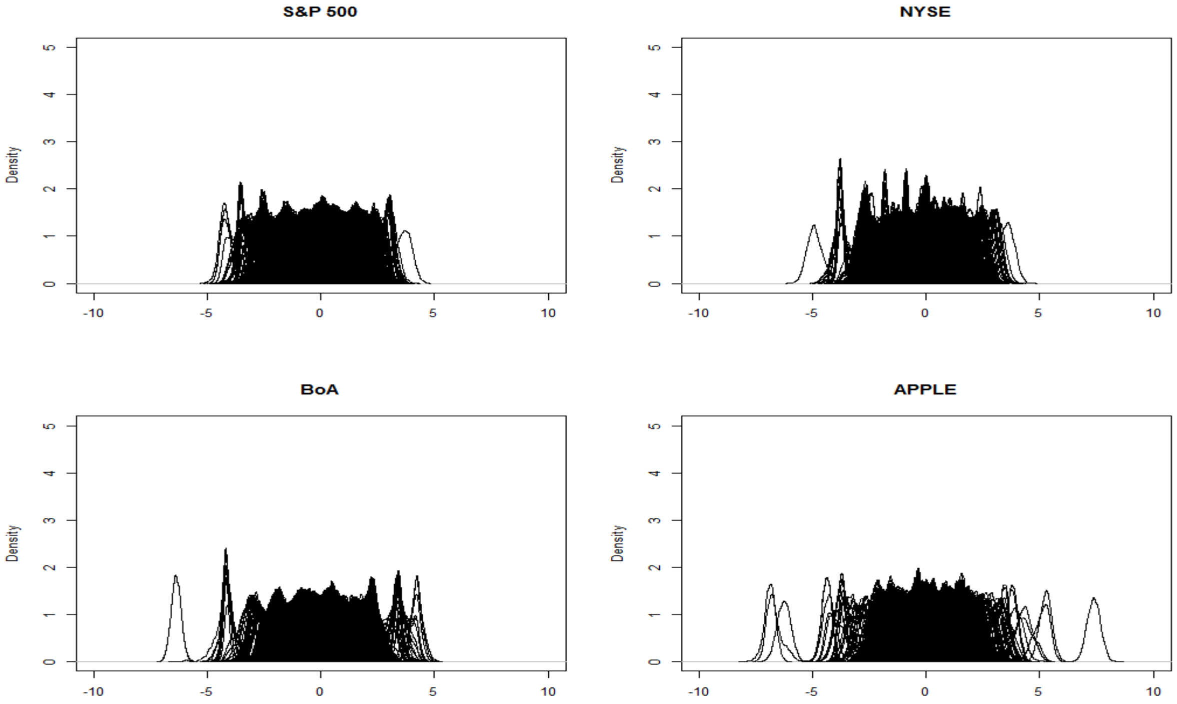
Table 4.11: Parameter estimates and inefficiency summary for the empirical results; The 4 cases are the four data sets fitted by their best GARCH model based on marginal likelihood: Case-1 is APPLE fitted by G_WDPMP with $p = 4, Q = 81$; Case-2 is BoA fitted by G_WDPMH with $p = 5, Q = 50$; Case-3 is NYSE fitted by G_WDPME with $p = 4, Q = 81$; Case-4 is S&P fitted by G_EWDPM with $p = 4, Q = 81$.

| | Case-1 | | Case-2 | | Case-3 | | Case-4 | |
|---|---|---|---|---|---|---|---|---|
| | Est. | Ineff. | Est. | Ineff. | Est. | Ineff. | Est. | Ineff. |
| $\alpha_0$ | 0.0413<br>(0.015,0.090) | 20.7 | 0.0458<br>(0.016,0.083) | 12.3 | 0.0129<br>(0.005,0.022) | 4.5 | 0.0149<br>(0.007,0.025) | 4.9 |
| $\alpha_1$ | 0.0454<br>(0.027,0.070) | 13.0 | 0.0732<br>(0.050,0.100) | 5.2 | 0.0781<br>(0.056,0.105) | 2.4 | 0.0843<br>(0.062,0.111) | 4.3 |
| $\beta$ | 0.920<br>(0.872,0.952) | 18.4 | 0.893<br>(0.857,0.924) | 8.0 | 0.885<br>(0.850,0.915) | 4.1 | 0.873<br>(0.838,0.905) | 5.5 |
| $h_0$ | 8.885<br>(1.374,24.197) | 8.3 | 5.924<br>(0.431,18.401) | 7.5 | 2.724<br>(0.352,7.561) | 8.8 | 2.682<br>(0.381,7.732) | 6.2 |
| $\mu_0$ | 0.0817<br>(-0.023,0.183) | 1.1 | -0.0053<br>(-0.189,0.172) | 2.4 | -0.0112<br>(-0.137,0.112) | 1.7 | 0.0335<br>(-0.112,0.178) | 2.7 |
| $\sigma_e^2$ | 1.77<br>(1.56,2.04) | 20.1 | 1.90<br>(1.54,2.38) | 16.9 | 1.58<br>(1.35,1.86) | 6.7 | 1.50<br>(1.26,1.79) | 11.1 |
| $\zeta$ | 7.13<br>(5.05,9.64) | 5.7 | 3.02<br>(1.53,5.21) | 26.9 | 3.85<br>(2.25,5.84) | 13.9 | 2.47<br>(1.21,4.10) | 35.9 |

Among the four cases, we can see that the parameter MCMC draws of APPLE data (Case-1) are less efficient than the other three data sets except in sampling $\zeta$. In the other three cases, the inefficiency scores of $\zeta$ are much higher than the other parameters'. But in APPLE data case, the draws of $\alpha_0$ becomes the least efficient one. We can also see that the posterior mean of $\zeta$ in APPLE is much higher than those in the other three cases, which implies that APPLE data favor more clusters in $\mu_t$'s. We do not include the estimates of $\psi$ because the estimate of $\psi$ are not comparable among different weight functions and in Case-2 and Case-3 $\boldsymbol{\psi}$ is actually not even a scalar but a vector.

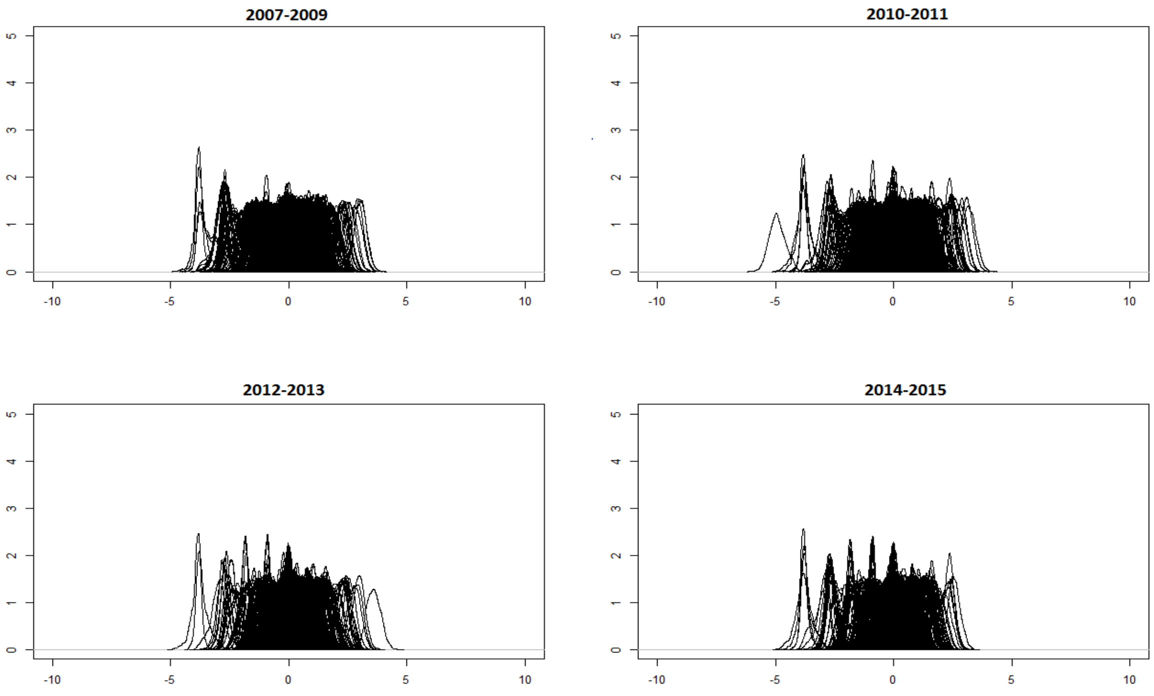
Besides the parameter estimates, we have also provided the cluster visualization of $\mu_t$'s as we did in simulation part. The four real data sets are fitted by their best models based on Section 4.8.1 and the posterior density curves of $\mu_t$ are plotted in Figure 4.6.

Figure 4.6: Posterior density curves of $\mu_t$ $(t = 1, \cdots, n)$ for real data sets, where $n = 2000$; There are $n$ density curves in each of the four panels; Each curve is plotted based on $40,000$ MCMC draws of $\mu_t$; The x-axis is the range of $\mu_t$ and y-axis stands for the kernel density estimated value; The four panels are the real data sets fitted by their best model; The four data sets are correspondingly S&P 500, NYSE, BoA and APPLE, and the best model are correspondingly G_WDPMG with p=4 Q=81, G_WDPMP with p=5 Q=100, G_WDPMH with p=5 Q=100 and G_WDPMP with p=4 Q=81.



We do see the clear evidence that multiple clusters exist among the $\mu_t$'s in every data sets. It is true in each plot that the separation between different clusters is not as clear as the one displayed in G_DPM-generated data fitted by G_WDPMP in simulation study. The results we see here is more similar to the one provided by G_WDPMP fitted by G_WDPMP, and this could imply that the nature of the four real data sets we have checked is close to the nature of data set generated by weighted DPM GARCH. We also observe that the pattern of the clusters tend to be stable across different time periods. For each of the four data sets, we divide the observations into different time periods and check the clusters in each time period. We find that there is no major difference among different periods. In Figure 4.7 we display one example (NYSE data). It can be found that the cluster plots in 2007-2009, 2010-2011, 2012-2013 and 2014-2015 are basically the same despite some slight differences.

Figure 4.7: Posterior density curves of $\mu_t$ for NYSE data sets; Fitting model is G_WDPMP with p=5 and Q=100; Each curve represents an observations and is plotted based on $40,000$ MCMC draws of $\mu_t$; The x-axis is the range of $\mu_t$ and y-axis stands for the kernel density estimated value; The four panels divide the observations in the NYSE data into four time period: 2007-2009, 2010-2011, 2012-2013 and 2014-2015.



### 4.8.3 BTR Comparisons for GARCH

We further compare GARCH models using the four empirical data sets to produce straightforward results about which models are more profitable. We will consider both "start with not holding" and "start with holding" cases because the open prices on trading day 1 are different from the closing prices on day 0. It should be noted that the total number of fittings are not doubled by considering both cases, because the model predictions are based on the same historical data. The only difference between the two cases is that we need to use different decision formula introduced in (4.8) in the first "decision-making-period". For each case of each data set, we consider using the best three weighted GARCH models in terms of marginal likelihood (based on the results in Table 4.10) along with G_t and G_DPM, and derive BTRs for these five models. We also calculate the return brought by the simplest strategy: buy the stock if we are not holding at the starting point and keep holding the stock till the end. The "simple" BTR is equal to the overall return of the stock during the period where the observations are made if we are holding the stock at the beginning, and will be slightly lower than overall return if we are holding and buy in at the beginning (because of commission fee). We use the "simple" BTR as benchmark to compare with BTRs produced by GARCH models. There are 200 "decision-making-periods" for each model applied to each data sets (i.e. $n_p = 2000$), and these 200 model fittings are conducted simultaneously using $n_0 = 300$ previous daily returns. We choose a smaller $n_0$ than the one in simulation because it is found that a relatively small $n_0$ can not only reduce computation burden, but also improve BTR results for most of the candidate models in this BTR comparison. As we did in simulation study, we assume commission fee $C_0$ is 0.01% of the original amount of money $\mathrm{Mn}_0$.

In Table 4.12 we display the results of BTR comparisons described above.

Table 4.12: BTR comparison among candidate models; We check BTR using each of the four data sets by their best three models based on marginal likelihood results in Table 4.10 along with G_t and G_DPM; The BTRs produced by different models are compared to the "simple" strategy in both "start with not holding" and "start with holding" cases; "simple" means the simplest strategy: buy the stock if we are not holding at the starting point and keep holding the stock till the end; The "simple" BTR is equal to (or very close to) the overall return of the stock during the period where the observations are made.

| Start with not holding | | | | | | | |
|---|---|---|---|---|---|---|---|
| APPLE | | BoA | | NYSE | | S&P 500 | |
| Approach | BTR | Approach | BTR | Approach | BTR | Approach | BTR |
| Simple | 351.6% | Simple | -63.6% | Simple | -0.3% | Simple | 30.7% |
| G_t | 233.9 % | G_t | -25.8 % | G_t | 1.6 % | G_t | 5.2 % |
| G_DPM | 215.7 % | G_DPM | -16.4% | G_DPM | -9.4 % | G_DPM | -0.4% |
| G_WDPMP p=4, Q=81 | 304.7% | G_WDPMH p=5, Q=80 | -65.9% | G_WDPMP p=5, Q=100 | 3.3% | G_WDPMG p=4, Q=81 | -7.5% |
| G_WDPMP p=5, Q=80 | 253.2% | G_WDPMP p=5, Q=100 | -55.5% | G_WDPMP p=4, Q=81 | -15.8% | G_WDPMH p=4, Q=81 | 64.8% |
| G_WDPMP p=5, Q=100 | 269.5% | G_WDPMH p=5, Q=50 | -11.7% | G_WDPMP p=5, Q=80 | -19.1% | G_WDPMH p=5, Q=80 | 39.6% |
| Start with holding | | | | | | | |
| APPLE | | BoA | | NYSE | | S&P 500 | |
| Approach | BTR | Approach | BTR | Approach | BTR | Approach | BTR |
| Simple | 354.1% | Simple | -63.4% | Simple | -0.3% | Simple | 30.7% |
| G_t | 235.6% | G_t | -24.9% | G_t | 1.6% | G_t | 5.2% |
| G_DPM | 217.3% | G_DPM | -15.7% | G_DPM | -9.5% | G_DPM | -0.4% |
| G_WDPMP p=4, Q=81 | 306.9% | G_WDPMH p=5, Q=80 | -63.1% | G_WDPMP p=5, Q=100 | 3.3% | G_WDPMG p=4, Q=81 | -7.3% |
| G_WDPMP p=5, Q=80 | 255.1% | G_WDPMP p=5, Q=100 | -54.3% | G_WDPMP p=4, Q=81 | -15.8% | G_WDPMH p=4, Q=81 | 65.1% |
| G_WDPMP p=5, Q=100 | 217.5% | G_WDPMH p=5, Q=50 | -10.6% | G_WDPMP p=5, Q=80 | -19.1% | G_WDPMH p=5, Q=80 | 39.9% |

We can see that BTRs in both cases are very similar although the ones in "start with holding" case are slightly higher than their counterpart in "start with not holding" cases. This is because it is true for all of the four stocks that open price of day 1 is slightly higher than closing price of day 0. For the four data sets (APPLE, BoA, NYSE and S&P500), there are correspondingly 0, 4, 2 and 2 models out of five which produce higher BTR than "simple" strategy if we start with not holding. If we start with holding, the slight difference is that all of the five models in BoA case produce better BTR than "simple" strategy. Compared to the APPLE data case where no GARCH models performs better than "simple" strategy, the BoA case displays the necessity of using GARCH when "simple" strategy brings highly negative return. Among the four cases, the best model in terms of marginal likelihood is still the model with highest BTR only in NYSE data. We can see that producing a higher marginal likelihood does not necessarily means the highest profitability. One can definitely explore more models on more data sets following the examples we provide.

## 4.9 Summary/Future Research

In this Chapter, we describe how to develop WDPM for GARCH models and how to calculate marginal likelihood for weighted GARCH and SV models. We also propose the BTR approach which produce intuitive result for GARCH models. Since SV model ignores the sign of data and model the log-squared return instead, this proposed empirical model evaluation method is not applicable to SV models.

In both cases of $n = 500$ and $n = 1000$ of simulation studies, we do see that neither type of SV and GARCH display advantage over the other type. But in real application, we find convincing evidence that WDPM GARCH models are more accurate than WDPM SV models in terms of marginal likelihood. We have also found that G_WDPMP models produce higher marginal likelihood than other models in simulation study, and that most of the best 20 models come from G_WDPMP and G_WDPMH in real application. Therefore, we suggest using G_WDPMP and G_WDPMH to fit data in terms of seeking better marginal likelihood. For $n = 500$ and 1000, we suggest using $p = 3, Q = 27$. For $n = 2000$, it is better to choose $p = 4$ or 5 and $Q = 80$ or 100. If one take computation time into consideration, G_WDPMP should be the optimal model because it saves around 30% of CPU time compared to G_WDPMH with the same $n, p$ and $Q$. In the future research, it is worthwhile to derive the optimal choice of $Q$ and find the relationship among $Q$, $p$ and $n$. It will require to have theoretical derivation and intensive simulation studies.

We also observe that a GARCH model with better marginal likelihood does not necessarily produce higher BTR, but WDPM allows us to seek a more profitable GARCH model by choosing different $n$ (fitting data sample size), $p$, $Q$ and weight functions. Our WDPM for GARCH can be combined with other extensions of GARCH such as leverage and jump effect in order to seek even better model (in terms of marginal likelihood or BTR) in future study.

Our BTR approach can also be easily implemented using future data rather than historical data. The only difference is that we need to run model at different time points as new observations are collected, while the multiple runs can be completed at the same time if we use historical data. It is interesting to test models in this dynamic way using future data and see the profit of trading updated as time goes on. We can simulate the real trading with no real money invested and compare model based on trading return.

# Bibliography

Anderson, H. M., Nam, K. and Vahid, F. (1999) Asymmetric Nonlinear Smooth Transition GARCH Models. P. Rothman (ed.), Nonlinear time series analysis of economic and financial data, Kluwer, Boston, 191-207

Anderson, R.M., Eom, K.S., Hahn, S.B. and Park, J.H. (2012) Sources of Stock Return Autocorrelation. Working Paper

Ardia, D. and Hoogerheide, L.F. (2010) Bayesian Estimation of the GARCH(1,1) Model with Student-t Innovations. The R Journal, vol.2, No.2, 41-47

Asai, M. and Watanabe, T. (2004) Comparison of MCMC Methods for Estimating GARCH Models, COE discussion paper series, No.18, Tokyo Metropolitan University

Ausín, M.C. and Galeano, P. (2007) Bayesian Estimation of the Gaussian Mixture GARCH Model. Computational Statistics & Data Analysis, vol.51, issue 5

Bakshi, G., Cao, C. and Chen, Z. (1997) Empirical Performance of Alternative Options Pricing Models, Journal of Finance, vol 52, No.5, 2003-2049

Basu, S. and Chib, S. (2003) Marginal Likelihood and Bayes Factors for Dirichlet Process Mixture Models. Journal of the American Statistical Association, vol.98 (461), 224-235

Bates, D.S. (1996) Jumps and Stochastic Volatility: Exchange Rate Processes Implicit in Deutsche Mark Options. The Review of Financial Studies, 9, 69107

Bauwens, L. and Lubrano, M. (1998) Bayesian Inference on GARCH Models Using the Gibbs Sampler, Econometrics Journal, 1, c23-c46

Blackwell, D. and MacQueen, J. B. (1973) Ferguson Distributions via Plya Urn Schemes. The Annals of Statistics, vol. 1, 353-355

Bollerslev, T. (1986) Generalized Autoregressive Conditional Heteroskedasticity. Journal of Econometrics, vol.31, issue 3, 307-327

Bollerslev, T. (1987) A Conditionally Heteroskedastic Time Series Model for Speculative Prices and Rates of Return. Review of Economics and Statistics, vol.69, issue 3, 542-547

Carr, P. and Sun, J. (2007) A new approach for option pricing under stochastic volatility. Review of Derivatives Research, 2007, vol. 10, issue 2, 87-150

Carter, C. K. and Kohn, R. (1994) On Gibbs Sampling for State Space Models. Biometrika, 81, 541-553

Carvalho, C. and Polson, N. G. (2010) The Horseshoe Estimator for Sparse Signals. Biometrika, vol.97, 465-480

Chan, W. H. and Maheu, J. M. (2002) Conditional Jump Dynamics in Stock Market Returns. Journal of Business and Economic Statistics, 20, 377-389

Chib, S. (1995) Marginal Likelihood from the Gibbs Output. Journal of the American Statistical Association, vol.90 (432), 1313-1321

Chib, S. and Greenberg, E. (1998) Analysis of Multivariate Probit Models. Biometrika, vol.85, issue 2, 347-361

Chib, S. and Greenberg, E. (2010) Additive Cubic Spline Regression with Dirichlet Process Mixture Errors. Journal of Econometrics, 156, 322-336

Chib, S., Jeliazkov, I. (2001) Marginal Likelihood from the MetropolisHastings Output. Journal of the American Statistical Association, vol.96, 270-281

Chib, S., Nadari, F. and Shephard, N. (2002) Markov Chain Monte Carlo Methods for Stochastic Volatility Models. Journal of Econometrics, 108, 281-316

Cootner, P.H. (1964) The Random Character of Stock Market Prices. M.I.T. Press

Cowles, A. and Jones, H.E. (1937) Some a Posteriori Probabilities in Stock Market Action. Econometrica, 5, 280-294

Danielsson, J. (1994) Stochastic Volatility in Asset Prices: Estimation with Simulated Maximum Likelihood. Journal of Econometrics, 61, 375-400

Delatola, E.I. and Griffin, J.E. (2011) Bayesian Nonparametric Modelling of the Return Distribution with Stochastic Volatility. Bayesian Analysis, 6, No.2, 901-926

Dunson, D.B., Pillai, N. and Park, J. (2007) Bayesian Density Regression. Journal of Royal Statistical Society, Series B, vol.69, Part 2, 163-183

Dunson, D. B. and Stanford, J. B. (2005) Bayesian Inferences on Predictors of Conception Probabilities. Biometrics, vol.61, 126-133

Durrleman, S. and Simon, R. (1989) Flexible Regression Models with Cubic Splines. Statistics in Medicine, vol.8, Issue 5, 551-561

Engle, R. F. and Ng, V. K. (1993) Measuring and Testing the Impact of News on Volatility. Statistics in Medicine, Journal of Finance, 48, 1749-1777

Eraker, B., Johannes, M. and Polson, N. (2003) The Impact of Jumps in Volatility and Returns. Journal of Finance, VOL. LVIII, NO. 3, 1269-1300

Escobar, M.D. and West, M. (1995) Bayesian Density Estimation and Inference Using Mixtures. Journal of the American Statistical Association, vol. 90, 577-588

Fama, E.F. (1970) Efficient Capital Markets: A Review of Theory and Empirical Work. The Journal of Finance, Vol. 25, No. 2, 383-417

Ferguson, T.S. (1973) A Bayesian Analysis of Some Nonparametric Problems. The Annals of Statistics, vol.1, 209-230

Ferguson, T.S. (1983) Bayesian Density Estimation by Mixtures of Normal Distributions. Recent Advances in Statistics: Papers in Honor of Herman Chernoff on His Sixtieth Birthday. Academic Press, 287-302

Franses, P.H. and van Dijk, D (2000) Non-Linear Time Series Models in Empirical Finance. Cambridge University Press (pg 30-31)

Gelfand, A. E., Sahu, S., and Carlin, B. (1995) Efficient Parametrization for Normal Linear Mixed Effects Models. Biometrika, 82, 479-488

Gewek, J.F. (1993) Bayesian Treatment of the Independent Student-t Linear Model. Journal of Applied Econometrics, 8(S1), S19-S40

Gilks, W.R., Best, N.G. and Tan, K.K.C. (1995) Adaptive Rejection Metropolis Sampling within Gibbs Sampling. Journal of the Royal Statistical Society, Series C (Applied Statistics), vol.44, No.4, 455-472

Glosten, L. R., Jagannathan, R. and Runkle, D. (1993). On the Relation between the Expected Value and the Volatility of Nominal Ecxess Returns on Stocks. Journal of Finance, 48, 1779-1801

Gonzalez-Rivera, G. (1998). Smooth Transition GARCH models, Studies in Nonlinear Dynamics and Econometrics, 3, 161-178.

Green, P.J., Silverman, B.W. (1994) Nonparametric Regression and Generalized Linear Models: A Roughness Penalty Approach. Chapman & Hall/CRC Monographs on Statistics & Applied Probability 58

Griffin, J. E.(2010) Default Priors for Density Estimation with Mixture Models. Bayesian Analysis, 5(1), 45-64

Griffin, J.E. and Steel, M.F.J. (2006) Order-based Dependent Dirichlet Processes. Journal of the American Statistical Association, vol.101, issue 473, 179-194

Hagerud, G. (1997) A New Non-Linear GARCH Model. EFI Economic Research Institute, Stockholm

Hannah, L.A., Blei, D.M. and Powell, W.B. (2011) Dirichlet Process Mixtures of Generalized Linear Models. Journal of Machine Learning Research, vol.12, 1923-1953

Harvey, A.C., Ruiz, E. and Shephard, N. (1994) Multivariate Stochastic Variance Models. The Review of Economic Studies, vol.61, No.2, 247-264

Holmes, C. C. and Held, L. (2006) Bayesian Auxiliary Variable Models for Binary and Multinomial Regression. Bayesian Analysis, vol.1, 145-168

Irwin, M., Cox, N. and Kong, A. (1994) Sequential Imputation for Multilocus Linkage Analysis. Proceedings of the National Academy of Sciences of the United States of America, vol.91, 11684-11688

Jacquier, E., Polson, N.G., Rossi, P.E. (1994) Bayesian Analysis of Stochastic Volatility Models (with discussion). Journal of Business and Economic Statistics, 12, 371-417

Jacquier, E., Polson, N. G., and Rossi, P.E. (2004) Bayesian Analysis of Stochastic Volatility Models with Fat-tails and Correlated Errors. Journal of Econometrics, 122, 185-212

Jegadeesh, N. and Titman, S. (1995) Short-Horizon Return Reversals and the Bid-Ask Spread. Journal of Financial Intermediation, vol.4, issue 2, 116-132

Jensen, M. J. (2004) Semiparametric Bayesian Inference of Long-memory Stochastic Volatility Models. Journal of Time Series Analysis, 25(6), 895-922

Jensen, M.J. and Maheu, J.M. (2013) Bayesian Semiparametric Multivariate GARCH Modeling Journal of Econometrics, vol.176(1) 3-17

Jensen, M.J. and Maheu J.M. (2014) Estimating a Semiparametric Asymmetric Stochastic Volatility Model with a Dirichlet Process Mixture. Journal of Econometrics, vol.178, part 3, 523-538

Jones, G.L., Roberts, G.O. and Rosenthal, J.S. (2014) Convergence of Conditional Metropolis-Hastings Samplers. Advances in Applied Probability, vol.46, No.2, 422-445

Jorion, P. (1988) On Jump Processes in the Foreign Exchange and Stock Markets. Review of Financial Studies, vol.1, No.4, 427-445

Kim, J., Shephard, N. and Chib, S. (1998) Stochastic Volatility: Likelihood Inference and Comparison with ARCH Models. The Review of Economic Studies, vol.5, No.3, 361-393

Knorr-Held, L. and Rue, H. (2002) On Block Updating in Markov Random Field Models for Disease Mapping. Scandinavian Journal of Statistics, 29, 597-614

Lancaster, P. and Šalkauskas, K. (1986) Curve and Surface Fitting: An Introduction. Academic Press, San Diego

Li, Q., and Racine J.S. (2006) Nonparametric Econometrics: Theory and Practice. Princeton University Press, Princeton

Llorente, G., Michaely, R., Saar, G. and Wang, J. (2002) Dynamic Volume-Return Relation of Individual Stocks. Review of Financial Studies, vol.15, issue 4, 1005-1047

MacEachern, S.N. (2000) Dependent Dirichlet Processes. Technical Report, Department of Statistics, Ohio State University.

MacEachern, S.N. and M*ü*ller, P. (1998) Estimating Mixtures of Dirichlet Process Models. Journal of Computational & Graphical Statistics, vol.7, 223-238

Maheu, J. M. and McCurdy, T. H. (2004) News Arrival, Jump Dynamics, and Volatility Components for Individual Stock Returns. Journal of Finance, 59, 755-793

Mahieu, R. J. and Schotman, P. C. (1998) An Empirical Application of Stochastic Volatility Models. Journal of Applied Econometrics, vol. 13(4), 333-360

Malik, S. and Pitt, M.K. (2009) Modelling Stochastic Volatility with Leverage and Jumps: A Simulated Maximum Likelihood Approach via Particle Filtering. The Warwick Economics Research Paper Series, 897. Working Paper

Malmsten, H. (2004) Evaluating Exponential GARCH Models. SSE/EFI Working Paper Series in Economics and Finance, No.564

Marsh,L.C. and Cormier, D.R. (2002) Spline Regression Models. Series: Quantitative Applications in the Social Sciences (v.137)

Mech, T.S. (1993) Portfolio Return Autocorrelation. Journal of Financial Economics, vol. 34, issue 3, 307-344

Nakajima, J. and Omori, Y. (2009) Leverage, Heavy-tails and Correlated Jumps in Stochastic Volatility Models. Computational Statistics & Data Analysis, 53, 2335-2353

Nelson, D.B. (1991) Conditional Heteroskedasticity in Asset Returns: A New Approach. Econometrica, vol.59, No.2, 347-370

Nieuwland, F.G.M.C., Verschoor, W.F.C. and Wolff, C.C.P. (1994) Stochastic Trends and Jumps in EMS Exchange Rates. Journal of International Money and Finance, vol.13, issue 6, 699-727

Omori, Y., Chib, S., Shephard, N. and Nakajima, J. (2007) Stochastic Volatility with Leverage: Fast and Efficient. Journal of Econometrics, 140, 425-449

Pagan, A. and Ullah, A. (1999) Nonparametric Econometrics. Themes in Modern Econometrics, Cambridge University Press, Cambridge

Papaspiliopoulos, O., Roberts, G. O. and Sköld, M. (2007) A General Framework for the Parametrization of Hierarchical Models. Statistical Science, 22, 59-73

Sandmann, G. and Koopman, S. J. (1998) Estimation of Stochastic Volatility Models via Monte Carlo Maximum Likelihood. Journal of Econometrics, 87(2): 271-301.

Schotman, P.C. and Mahieu, R. (1994) Stochastic Volatility and the Distribution of Exchange Rate News. Federal Reserve Bank $of$ Minneapolis, Discussion Paper 96

Sentana, E. (1995) Quadratic ARCH Models. Review of Economic Studies, 62, 639-661

Sethuraman, J. (1994) A Constructive Definition of Dirichlet Priors. Statistica Sinica, vol.4, 639-650

Sun, P., Kim, I. and Lee, K. (2014) Weighted Dirichlet Process Mixture for Semiparametric Regression. Working Paper

Sun, P., Kim, I. and Lee, K. (2015) New Weighted Approach for Stochastic Volatility Model. Working Paper

Taylor, S.J. (1994) Modelling stochastic volatility. Mathematical Finance, 4, 183204

Verbeek, M., 2008. A Guide to Modern Econometrics, third ed. John Wiley & Sons, Ltd., Chichester.

Virbickaite, A., Lopes, H.F., Auśin, C. and Galeano, P. (2014) Particle Learning for Bayesian Non-parametric Markov Switching Stochastic Volatility Model. Universidad Carlos III de Madrid. working paper 14-28, Statistics and Econometrics Series (19)

Vlaar, P.J.G and Palm, F.C. (1993) The Message in Weekly Exchange Rates in the European Monetary System: Mean Reversion, Conditional Heteroskedasticity, and Jumps. Journal of Business & Economic Statistics, vol.11, No.3, 351-360

Wago, H. (2004) Bayesian Estimation of Smooth Transition GARCH Model Using Gibbs Sampling. Mathematics and Computers in Simulation, 64, 63-78

Wood, S.N. (2006) Generalized Additive Models: An Introduction with R. In: Texts in Statistical Science, Chapman & Hall/CRC, Boca Raton, FL.

Yu, J. (2005) On Leverage in a Stochastic Volatility Model. Journal of Econometrics, 127: 165-178

Zellner, A. (1986) On Assessing Prior Distributions and Bayesian Regression Analysis with g-prior Distributions. In Bayesian Inference and Decision Techniques: Essays in Honor of Bruno de Finetti, 233243. Amsterdam: North-Holland

# Appendix A

# Procedures of MCMC Sampling for Chapter 2

Step 1 Sample cut-point vector **a** using Metropolis-Hastings: to do this, derive

$$\begin{aligned} m &= \underset{a}{\text{argmax}} \; \log \Pr(y|\lambda, \beta, a) \\ V &= \left\{ \frac{-\partial \log \Pr(y|\lambda, \beta, a)}{\partial a \partial a^T} \right\}^{-1} \end{aligned}$$

and propose $\mathbf{a}_{new} \sim t_{15}(a|m, V)$. We then update **a** by $\mathbf{a}_{new}$ with probability $\alpha_{MH}$, where

$$\alpha_{MH} = \min \left\{ \tfrac{\pi(a_{new}) \Pr(y|\lambda,\beta,a_{new}) t_{15}(a|m,V)}{\pi(a) \Pr(y|\lambda,\beta,a) t_{15}(a_{new}|m,V)}, 1 \right\},$$

and derive the vector of free cut-points **c**

Step 2 Sample the continuous latent response $\{y_i^*\}$ from $\mathrm{N}(x_i^T \beta, \lambda_i^{-1})$ truncated to the interval $(c_{j-1}, c_j)$ for $y_i = j$

Step 3 Sample $\boldsymbol{\beta}$ from the following normal distribution,

$$\mathrm{N}_k[\beta | B(B_0^{-1} b_0 + \textstyle\sum_i \lambda_i x_i y_i^*), B)]$$

where $B = (B_0^{-1} + \sum_i \lambda_i x_i x_i^T)^{-1}$, $B_0 = diag\left\{B_{00}, \Delta_1^{-1}T_1(\Delta_1^{-1})^T, \cdots, \Delta_q^{-1}T_q(\Delta_q^{-1})^T\right\}$,$b_0 = (b_{00}^T, 0_{k_1 \times 1})^T$, $k_1 = \sum M_j - q$, $b_{00}$ is the mean vector of the normal prior for $\beta_0$, $B_{00}$ is the covariance matrix of the normal prior for $\beta_0$ , $T_j = \text{diag}\left\{\sigma_{ej}^2, \sigma_{dj}^2 I_{M_j-3}, \sigma_{ej}^2\right\}$,

Step 4 Update $\sigma_e^2$ and $\sigma_d^2$ using the following inverse gamma distributions,

$$\begin{aligned} \sigma_{ej}^2 &\sim \text{inv gamma}(\frac{\alpha_{je0}+2}{2}, \frac{\delta_{je0} + \beta_j^T \Delta_j^T D_{0j} \Delta_j \beta_j}{2}), \\ \sigma_{dj}^2 &\sim \text{inv gamma}(\frac{\alpha_{jd0}+M_j-3}{2}, \frac{\delta_{jd0} + \beta_j^T \Delta_j^T D_{1j} \Delta_j \beta_j}{2}), \end{aligned}$$

where $\alpha_{je0}, \alpha_{jd0}, \delta_{je0}, \delta_{jd0}$ are the hyper parameters of the gamma priors for $\sigma_{ej}^2$ and $\sigma_{dj}^2$, and

$$D_{0j} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0_{M_j-3} & 0 \\ 0 & 0 & 1 \end{bmatrix}, D_{0j} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & I_{M_j-3} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Step 5 Update $S$:

$$\Pr(S_t = s | Z_t = q) \propto \begin{cases} \text{N}(y_i^* | x_i^T \beta, \lambda_h^{-1}) \cdot \frac{n_{s(-t)}}{n_q + \zeta - 1} & s \in \left\{q_1, \cdots, q_{k_j}\right\} \\ t_v(y_i^* | x_i^T \beta, 1) \cdot \frac{\zeta}{n_q + \zeta - 1} & s = \max(S^{-t}) + 1 \end{cases}$$

where $n_q$ is the number of observations that select the $q_{th}$ candidate prior (denote the subset of $S$ including these observations as $S_{Z=q}$ ), and $n_{s(-t)}$ is the number of observations except the $t_{th}$ one that belong to the $s_{th}$ unique value in $\boldsymbol{\lambda}$. $S^{-t}$ means $S / \{S_t\}$ and $\left\{q_1, \cdots, q_{k_j}\right\}$ is the set of unique values in the intersection of $S^{-t}$ and $S_{Z=q}$.

Step 6 Sample the $k$ distinct values $(\lambda_1^*, \cdots, \lambda_k^*)$ in $\lambda$,

$$\lambda_h^* \sim \text{gamma}(\tfrac{v + \sum \mathbf{1}(S_i = h)}{2}, \tfrac{v + \sum_{S_i = h}(y_i - x_i^T \beta)^2}{2}), \quad h = 1, \cdots, k$$

Step 7 Sample $C$ vector directly from the weights

$$\Pr(C_h = q | C_{-h}, S, y, \zeta, \theta_H) \propto \frac{\Gamma[(\zeta + n_{q(-h)}]}{\Gamma[\zeta + n_{q(-h)} + n_{(h)}]} \prod_{i=1}^{n_{(h)}} b_{iq}$$

Step 8 Sample the hyper parameters $\psi$ for weight function described in (a) and update $\gamma_j$ explained in (b):

(a) For the unique $\psi$ case, use

$$\underset{\psi}{\text{argmax}} \; \log\{w(Z|\psi)\} = \underset{\psi}{argmax} \; \log(\prod_{i=1}^{n} \frac{e^{-\psi \left\| x_i - x^c_{Z_i} \right\|^2}}{\sum_{q=1}^{Q} e^{-\psi \left\| x_i - x^c_q \right\|^2}})$$

and the negative inverse of the second order derivative as the location ($m$) and scale ($\sigma^2$) parameters of the Student-t with degree of freedom 15, which is the proposal distribution of the Metropolis-Hastings algorithm fro sampling $\psi$. The accepting probability is

$$\min\Big\{ \tfrac{\pi(\psi_{new}) w(Z|\psi_{new})}{\pi(\psi) w(Z|\psi)} \cdot \tfrac{t_{15}(\psi | m, \sigma^2)}{t_{15}(\psi_{new} | m, \sigma^2)}, 1 \Big\}.$$

And for the $\psi = (\psi_1, \cdots, \psi_Q)$ case, we apply sub-optimal proposal. The target density is proportional to $\pi(\psi_q) w(Z|\psi_q)$ and our proposal $q(\psi_{new,q}|\psi_q) \propto \psi_{new,q}^{-1} e^{-(\log\psi_{new,q} - \log\psi_q)^2/2} \mathbf{1}_{[0,10]}$.

(b) This step is only necessary for WDPMD model, where we have to update $\gamma_j$'s as Dunson et al (2007) propose. For a given $\psi$, let $K^*_{ij} = \frac{\exp(-\psi \left\| x_i - x^c_j \right\|^2)}{\sum_{l \neq j} \gamma_l \exp(-\psi \left\| x_i - x^c_j \right\|^2)}$. Then $\gamma_j$ can be sampled by the following Gibbs sampling:

i. $S^*_{ij} \sim$ Poisson $(\gamma_j \xi_{ij} K^*_{ij})$ if $Z_i = j$, otherwise $S_{ij} = 0$,

ii. $\xi_{ij} \sim$ gamma $(1 + S^*_{ij}, 1 + \gamma_j K^*_{ij})$,

iii. $\gamma_j \sim$ gamma $(a_\gamma + \sum_{i=1}^{n} S^*_{ij}, b_\gamma + \sum_{i=1}^{n} \xi_{ij} K^*_{ij})$,

where $S^*_{ij}$ and $\xi_{ij}$ is auxiliary variables that are introduced to realize the Gibbs sampling, and $(a_\gamma, b_\gamma)$ specify the gamma hyper priors that are used for $\gamma_j$'s.

Step 9 Sample $\zeta$: let $kz$ be the distinct values in $Z$ and $(n_1, \cdots, n_{kz})$ denote the numbers of observations that come from these $kz$ candidate priors. For among $\lambda_i$'s that come from the $r_{th}$ one of these $kz$ candidate prior $(r = 1, \cdots, kz)$, suppose that there are $k_r$ distinct values of $\lambda$. Then we have:

$$\Pr(k_1, \cdots, k_{kz}|\zeta) \propto \zeta^{\sum k_r} \prod_{r=1}^{kz} \frac{\Gamma(\zeta)}{\Gamma(\zeta + n_r)}.$$

We use $\underset{\zeta}{\text{argmax}} \ \log\{\Pr(k_1, \cdots, k_p|\zeta)\}$. For sampling $\zeta$ using MH, we use the proposal distribution as the multivariate Student-t with degree of freedom 15 with the location $m$ and scale $V$ which are setted as the negative inverse of the corresponding Hessian matrix of $\underset{\zeta}{\text{argmax}} \ \log\{\Pr(k_1, \cdots, k_p|\zeta)\}$. Hence the accepting probability is

$$\min\left\{\frac{\pi(\zeta_{new})\Pr(k_1, \cdots, k_p|\zeta_{new})}{\pi(\zeta)\Pr(k_1, \cdots, k_p|\zeta)} \cdot \frac{t_{15}(\zeta|m, V)}{t_{15}(\zeta_{new}|m, V)}, 1\right\}.$$

REMARK: since our procedures are general case, they can be applied for other models such as (i) for continuous case, Step 1 will be dropped and we directly use $y$ instead of the latent response $y^*$; (ii) for DPM, Step 7 and 8 should be dropped and Step 5-6 will be simplified into one step:

$$\begin{aligned}\lambda_i|\lambda_{-i}, y^*, \beta, G_0 \quad \sim \quad & \frac{\zeta t_v(y_i^*|x_i^T\beta, 1)}{\zeta t_v(y_i^*|x_i^T\beta, 1) + \sum_{j\neq i} \mathrm{N}(y_i^*|x_i^T\beta, \lambda_j^{-1})}\text{gamma}(\frac{v+1}{2}, \frac{v + (y_i - x_i^T\beta)^2}{2}) \\ & + \sum_{j\neq i} \frac{\mathrm{N}(y_i^*|x_i^T\beta, \lambda_j^{-1})}{\zeta t_v(y_i^*|x_i^T\beta, 1) + \sum_{j\neq i} \mathrm{N}(y_i^*|x_i^T\beta, \lambda_j^{-1})}\delta_{\lambda_j}\end{aligned}$$

and (iii) for $t$ model, Step 7-9 will be dropped and Step 5-6 can be further simplified:

$$\lambda_i|\lambda_{-i}, y^*, \beta, G_0 \sim \text{gamma}(\frac{v+1}{2}, \frac{v + (y_i - x_i^T\beta)^2}{2}).$$

# Appendix B

# Proof of Property-5 in Section 3.3

For a fixed $\gamma_t$, suppose $\pi_t$ is determined by $\gamma_t$ and some $\psi$. We need to prove that there is a unique value of $\psi$ which generates $\pi_t$. We can rewrite Eq.(3.9) as a set of linear equations:

$$\begin{bmatrix} 1 & -\gamma_t & 0 & \cdots & \cdots & 0 \\ 0 & \gamma_t & -\gamma_t^2 & 0 & \cdots & 0 \\ \vdots & 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & \gamma_t^{Q-1} & -\gamma_t^Q \\ 0 & 0 & \cdots & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ \prod_{l=1}^{1}(1-p_{(l)t}) \\ \vdots \\ \vdots \\ \prod_{l=1}^{Q}(1-p_{(l)t}) \end{bmatrix} = \begin{bmatrix} \pi_{(1)t} \\ \pi_{(2)t} \\ \vdots \\ \vdots \\ \pi_{(Q)t} \\ 0 \end{bmatrix}$$

It should be noted that the last equation above is added to make the coefficient matrix a $Q \times (Q+1)$ matrix. The last equation is validated by the fact that $1 - p_{(Q)t} = 0$ (therefore $\prod_{l=1}^{Q}(1-p_{(l)t}) = 0$). We can see that the coefficient matrix is non-singular because it is a upper triangular matrix and the diagonal elements are all non-zero if $\gamma_t \neq 0$. This means there is a unique solution to this set of equations, thus $(p_{(1)t}, \cdots, p_{(Q)t})$ is fixed once $\gamma_t$ and $\pi_t$ are fixed. Since $p_{(q)t}$ is a function of $\psi$ based on Eq.(3.8), let us consider $\psi, \psi' > 0$ such that $p_{(q)t}(\psi) = p_{(q)t}(\psi')$ is true for $q = 1, \cdots, Q$.

Then the following is true for $Q \geq 2$:

$$(1 - p_{(1)t}(\psi))p_{(2)t}(\psi) = (1 - p_{(1)t}(\psi'))p_{(2)t}(\psi') \tag{AP.1}$$

Based on Eq.(3.8) we know that $(1 - p_{(1)t}(\psi))p_{(2)t}(\psi) = \frac{\left\|x_t - x^c_{(2)}\right\|^{-\psi}}{\sum_{l=1}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi}}$. Using Eq.(AP.1) and the fact $p_{(1)t}(\psi) = p_{(1)t}(\psi')$, we have:

$$\begin{aligned} \frac{\left\|x_t - x^c_{(1)}\right\|^{-\psi}}{\sum_{l=1}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi}} &= \frac{\left\|x_t - x^c_{(1)}\right\|^{-\psi'}}{\sum_{l=1}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi'}} \\ \frac{\left\|x_t - x^c_{(2)}\right\|^{-\psi}}{\sum_{l=1}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi}} &= \frac{\left\|x_t - x^c_{(2)}\right\|^{-\psi'}}{\sum_{l=1}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi'}} \end{aligned} \tag{AP.2}$$

From Eq.(AP.2) we know that $\frac{\left\|x_t - x^c_{(1)}\right\|^{-\psi}}{\left\|x_t - x^c_{(2)}\right\|^{-\psi}} = \frac{\left\|x_t - x^c_{(1)}\right\|^{-\psi'}}{\left\|x_t - x^c_{(2)}\right\|^{-\psi'}}$. Using this method we can similarly derive that $\frac{\left\|x_t - x^c_{(q)}\right\|^{-\psi}}{\left\|x_t - x^c_{(q+1)}\right\|^{-\psi}} = \frac{\left\|x_t - x^c_{(q)}\right\|^{-\psi'}}{\left\|x_t - x^c_{(q+1)}\right\|^{-\psi'}}$ is true for $q = 1, \cdots, Q-1$. Since $\left\|x_t - x^c_{(q)}\right\|$ is not the same for all $q$, there exists $q_0$ such that $\frac{\left\|x_t - x^c_{(q_0)}\right\|}{\left\|x_t - x^c_{(q_0+1))}\right\|} \neq 1$ while $(\frac{\left\|x_t - x^c_{(q_0)}\right\|}{\left\|x_t - x^c_{(q_0+1))}\right\|})^{\psi-\psi'} = 1$. So we have $\psi = \psi'$.

For a fixed $\psi$, suppose $\pi_t$ is determined by $\psi$ and some $\gamma_t$. We know $p_{(1)t} < 1$ if $Q \geq 2$ and $\psi < \infty$. Therefore $\gamma_t = \frac{1-\pi_{(1)t}}{1-p_{(1)t}}$ is unique.

# Appendix C

# Proof of Property-6 in Section 3.3

for $\forall$ integer $a$ that satisfies $1 \leq a \leq Q-1$ and $\forall t \in \{1, \cdots, n\}$, we have:

$$\pi_{(a)t} = \gamma_t^{a-1}[\prod_{l=0}^{a-1}(1-p_{(l)t})](1-\gamma_t(1-p_{(a)t}))$$
$$\pi_{(a+1)t} = \gamma_t^{a}[\prod_{l=0}^{a}(1-p_{(l)t})](1-\gamma_t(1-p_{(a+1)t}))$$

To ensure the above two equations are true for $a = 1$, we define $p_{(0)t} = 0$. Eliminating the common positive terms that $\pi_{(a)t}$ and $\pi_{(a+1)t}$ share, we can see that the sign of $\pi_{(a)t} - \pi_{(a+1)t}$ is determined by:

$$\begin{aligned} & 1 - \gamma_t(1-p_{(a)t}) - \gamma_t(1-p_{(a)t})[1-\gamma_t(1-p_{(a+1)t})] \\ = & 1 - 2\gamma_t(1-p_{(a)t}) + \gamma_t^2(1-p_{(a)t})(1-p_{(a+1)t}) \overset{\triangle}{=} \Delta(\gamma_t) \end{aligned}$$

Taking the derivative of $\Delta(\gamma_t)$ we have:

$$\Delta^{'} = -2(1-p_{(a)t}) + 2(1-p_{(a)t})(1-p_{(a+1)t}) = -2p_{(a+1)t}(1-p_{(a)t}) \leq 0$$

So we know that $\Delta(\cdot)$ is a decreasing function. Since $\gamma_t \in [0,1]$, we have that $\Delta(\gamma_t) \geq \Delta(1)$. And:

$$\begin{aligned}
\Delta(1) &= 1 - 2(1 - p_{(a)t}) + (1 - p_{(a)t})(1 - p_{(a+1)t}) \\
&= p_{(a)t} + p_{(a)t}p_{(a+1)t} - p_{(a+1)t} \\
&= \frac{\left\|x_t - x^c_{(a)}\right\|^{-\psi}}{\sum_{l=a}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi}} + \frac{\left\|x_t - x^c_{(a)}\right\|^{-\psi}}{\sum_{l=a}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi}} \frac{\left\|x_t - x^c_{(a+1)}\right\|^{-\psi}}{\sum_{l=a+1}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi}} - \frac{\left\|x_t - x^c_{(a+1)}\right\|^{-\psi}}{\sum_{l=a+1}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi}} \\
&= \frac{\left\|x_t - x^c_{(a)}\right\|^{-\psi}\sum_{l=a+1}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi} + \left\|x_t - x^c_{(a)}\right\|^{-\psi}\left\|x_t - x^c_{(a+1)}\right\|^{-\psi} - \left\|x_t - x^c_{(a+1)}\right\|^{-\psi}(\left\|x_t - x^c_{(a)}\right\|^{-\psi} + \sum_{l=a+1}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi})}{\left\{\sum_{l=a}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi}\right\} \times \left\{\sum_{l=a+1}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi}\right\}} \\
&= \frac{(\left\|x_t - x^c_{(a)}\right\|^{-\psi} - \left\|x_t - x^c_{(a+1)}\right\|^{-\psi})\sum_{l=a}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi}}{\sum_{l=a}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi}\sum_{l=a+1}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi}} \\
&= \frac{\left\|x_t - x^c_{(a)}\right\|^{-\psi} - \left\|x_t - x^c_{(a+1)}\right\|^{-\psi}}{\sum_{l=a}^{Q}\left\|x_t - x^c_{(l)}\right\|^{-\psi}}
\end{aligned}$$

Since $\left\|x_t - x^c_{(a)}\right\| \leq \left\|x_t - x^c_{(a+1)}\right\|$, so $\left\|x_t - x^c_{(a)}\right\|^{-\psi} \geq \left\|x_t - x^c_{(a+1)}\right\|^{-\psi}$. Therefore we derive that $\Delta(1) \geq 0$. This means that $\Delta(\gamma_t)$ is not negative, so we have $\pi_{(a)t} \geq \pi_{(a+1)t}$ for $\forall\gamma_t \in [0,1]$, $\psi \geq 0$ and integer $a$ between 1 and $Q$.

# Appendix D

# Joint Distribution for Chapter 3

The joint distribution of everything for SV_WDPMP can be written as:

$$
\begin{aligned}
&\pi(\Theta_0, \Theta_1, \Theta_2) p(Z|\Theta_2) p(\vec{k}|\zeta, \vec{n}) p(S|\zeta, \vec{k}, \vec{n}) [\prod_{t=1}^{n} f(y_t, h_t|\boldsymbol{\mu}, \Theta_0, \Theta_1, S)][\prod_{s=1}^{k} dG_0(\mu_s^*|\Theta_1)] \propto \\
&e^{-(\phi-0.95)^2} (\sigma_\eta^2)^{-2.5-1} e^{-0.025/\sigma_\eta^2} e^{-\zeta/5} W^{0.1-1} (1-W)^{0.9-1} e^{-\psi} \\
&\times \prod_{t=1}^{n} \gamma_t^{q_t-1} [\prod_{l=0}^{q_t-1} (1-p_{(l)t})][1-\gamma_t(1-p_{(q_t)t})] \\
&\times W^{n-\sum_{j=1}^{kz} n_j} (1-W)^{\sum_{j=1}^{kz} n_j} \zeta^{\sum_{j=1}^{kz} k_j} [\prod_{j=1}^{kz} \Gamma(\zeta)/\Gamma(\zeta+n_j)] \\
&\times \prod_{t=1}^{n} [(\sigma_0^2)^{-0.5} e^{-(y_t-h_t-logc)^2/2\sigma_0^2} \mathbf{1}_{S_t=0} + (\alpha\sigma_z^2)^{-0.5} e^{-(y_t-h_t-\mu_{S_t}^*)^2/(2\alpha\sigma_z^2)} \mathbf{1}_{S_t>0}] \\
&\times (1-\phi^2)^{0.5} (\sigma_\eta^2)^{-0.5} e^{-(1-\phi^2)h_1^2/(2\sigma_\eta^2)} [\prod_{t=2}^{n} (\sigma_\eta^2)^{-0.5} e^{-(h_t-\phi h_{t-1})^2/(2\sigma_\eta^2)}] \\
&\times \prod_{s=1}^{k} [(1-\alpha)\sigma_z^2]^{-0.5} e^{-(\mu_s^*-\mu_0)^2/[2\sigma_z^2(1-\alpha)]}]
\end{aligned}
\tag{AP.3}
$$

In Eq.(AP.3) $n_j$ represents the numbers of observations that are labeled as non-zero return and select the $j_{th}$ candidate prior, and $k_j$ is the number of unique values of $\boldsymbol{\mu}$ other than log$c$ in the $j_{th}$ cluster. $\vec{n}$ is $(n_1, \cdots, n_Q)$ and $\vec{k}$ is $(k_1, \cdots, k_Q)$. $k$ is the number of unique values other than log$c$ in $\boldsymbol{\mu}$ in all of the clusters. $q_t$ means that observation $t$ select its $q_t$-$th$ nearest candidate as $G_t$.

# Appendix E

# Procedures of MCMC Sampling for Chapter 3

- Update log-volatility vector $\boldsymbol{h}$: We apply Forward Filtering Backward Sampling (FFBS) algorithm (Carter and Kohn, 1994; Kim et al, 1998) to update $\boldsymbol{h}$. Compared to the M-H algorithm proposed by Jacquier et al (1994) which update one element of $\boldsymbol{h}$ at a time, FFBS can avoid highly correlated draws of log-volatility.

- Update $\boldsymbol{S}$:

$$\Pr(S_t = s|Z_t = q) \propto \begin{cases} W(\sigma_0^2)^{-0.5} e^{-(y_t - h_t - \log c)^2/(2\sigma_0^2)} & s = 0 \\ (1-W)\frac{n'_{s(-t)}}{n_q+\zeta-1}(\alpha\sigma_z^2)^{-0.5} e^{-(y_t-h_t-\mu_s^*)^2/(2\alpha\sigma_z^2)} & s \in \left\{q_1, \cdots, q_{k_j}\right\} \\ (1-W)\frac{\zeta}{n_q+\zeta-1}(\sigma_z^2)^{-0.5} e^{-(y_t-h_t-\mu_0)^2/(2\sigma_z^2)} & s = \max(\boldsymbol{S}^{-t}) + 1 \end{cases}$$

  where $S_t = 0$ means that this observation is labeled as zero return, $n_q$ is the number of observations who select the $q_{th}$ candidate prior and are labeled as non-zero returns (denote the subset of $\boldsymbol{S}$ including these observations as $S^+_{Z=q}$ ), and $n'_{s(-t)}$ is the number of observations except the $t_{th}$ one that belong to the $s_{th}$ unique value in $\boldsymbol{\mu}$. $\boldsymbol{S}^{-t}$ means $\boldsymbol{S}/\left\{\boldsymbol{S}_t\right\}$ and $\left\{q_1, \cdots, q_{k_j}\right\}$ is the set of unique values in the intersection of $\boldsymbol{S}^{-t}$ and $S^+_{Z=q}$.

- Update $\boldsymbol{Z}$: Suppose $\boldsymbol{C}$ is the vector that allocates the $k$ unique values in $\boldsymbol{\mu}$ other than log$c$ to the $Q$ candidate priors (if $\mu_t$ = log$c$, this means observation $t$ is labeled as zero return). We can see that once $\boldsymbol{C}$ is specified, every non-zero return observation will be assigned to a candidate. For $S_t = 0$, we sample $Z_t$ using the weight vector $\pi_t$ defined by Eq.(3.5). For $S_t > 0$, we sample $\boldsymbol{C}$ based on Eq.(3.10):

$$p(C_j = q|\boldsymbol{C}_{(-j)}, \boldsymbol{S}, \boldsymbol{y}, \Theta) \propto \frac{\Gamma[(\zeta + n_{q(-j)}]}{\Gamma[\zeta + n_{q(-j)} + n_{(j)}]} \prod_{i=1}^{n_{(j)}} w_{iq} \quad q = 1, \cdots, Q$$

It should be noted that $n_{q(-j)}$ is the number of observations which select the $q_{th}$ candidate and are labeled as non-zero return while not assigned to the $j_{th}$ unique value.

- Update $\gamma_t$: $(1 - p_{(q_t)t})\gamma_t \sim$ Beta$(q_t, 2)\mathbf{1}_{(0,1-p_{(q_t)t})}$.

- Update $\psi$: We apply M-H. The proposal is $q(\psi_{new}|\psi) \propto \mathrm{N}(\psi_{new}|\psi, 1) \cdot \mathbf{1}_{[0,4]}$ and the prior of $\psi$ is Gamma(0.2,0.1).

- Update $\boldsymbol{\mu}$: The unique values for non-zero returns in $\boldsymbol{\mu}$ can be sampled as:

$$\mu_s^* \sim \mathrm{N}(\frac{\frac{\sum_{S_t=s}(y_t - h_t)}{\alpha} + \frac{\mu_0}{1-\alpha}}{\frac{n_s^{'}}{\alpha} + \frac{1}{1-\alpha}}, \frac{\sigma_z^2}{\frac{n_s}{\alpha} + \frac{1}{1-\alpha}}) \quad s = 1, \cdots, k$$

where $n_S^{'}$ is the number of non-zero observations that belong to the $s_{th}$ unique value. Then we can assign these unique values to $\boldsymbol{\mu}$ based on $\boldsymbol{S}$. If $S_t = 0$, then $\mu_t$ = log$c$.

- To avoid highly correlated MCMC draws, we adopt the reparameterization approach introduced in Gelfand et al (1995) and Papaspiliopoulos et al (2007). let $CP = 1$ means that we use centered parameterization and $CP = 0$ otherwise. We choose $CP = 1$ with a probability of 0.5. If $CP = 1$, we set $\mu_s^{**} = \mu_s^* - \mu_0$ for $s = 1, \cdots, k$ and $\boldsymbol{h}^* = \boldsymbol{h} + \mu_0$. An alternative approach is joint updating introduced in Knorr-Held and Rue (2002).

- Update $\mu_0$: If $CP = 1$, we have:

$$\mu_0 \sim \mathrm{N}(\frac{(1-\phi^2)h_1^* + (1-\phi)\sum_{t=1}^{n-1}(h_{t+1}^* - \phi h_t^*)}{(n-1)(1-\phi)^2 + (1-\phi^2)}, \frac{\sigma_\eta^2}{(n-1)(1-\phi)^2 + (1-\phi^2)})$$

If $CP = 0$:

$$\mu_0 \sim \mathrm{N}(\frac{\sum_{s=1}^{k} \mu_s^*}{k}, \frac{(1-\alpha)\sigma_z^2}{k})$$

- Update $\sigma_z^2$: If $CP = 1$ we have:

$$\sigma_z^2 \sim \mathrm{IG}(\frac{n - n_0 + k}{2} + 1, 3 + \frac{1}{2}[\frac{\sum_{S_t>0}(y_t - h_t^* - \mu_s^{**})^2}{\alpha} + \frac{\sum_{s=1}^{k}(\mu_s^{**})^2}{1-\alpha}])$$

where $n_0$ is the number of observations labeled as zero return. If $CP = 0$:

$$\mathrm{IG}(\frac{n - n_0 + k}{2} + 1, 3 + \frac{1}{2}[\frac{\sum_{S_t>0}(y_t - h_t - \mu_s^*)^2}{\alpha} + \frac{\sum_{s=1}^{k}(\mu_s^* - \mu_0)^2}{1-\alpha}])$$

- Update $\zeta$: We can generalize the approach of Escobar and West (1995) to sample $\zeta$. Suppose there are $kz$ unqiue values in $\boldsymbol{Z}$, and the number of observations that fall into each of these $kz$ clusters is $(n_1, \cdots, n_{kz})$. Suppose there are $k$ unique values in $\boldsymbol{S}$. The prior of $\zeta$ is Gamma$(a_0, d_0)$. We first generate $kz$ auxiliary variables $(\eta_1, \cdots, \eta_{kz})$:

$$\eta_q \sim \mathrm{Beta}(\zeta + 1, n_q) \quad q = 1, \cdots, kz$$

The next step is to find the coefficients $(c_0, c_1, \cdots, c_q)$ which satisfy:

$$c_0 + c_1\zeta + \cdots + c_{kz}\zeta^{kz} \equiv \prod_{q=1}^{kz}(\zeta + n_q)$$

Using these coefficients, we can calculate the probabilities:

$$P_r \propto \frac{c_r \cdot \Gamma(a_0 + k - kz + r)}{(d_0 - \sum_{q=1}^{kz}[\log(\eta_q)])^r} \quad r = 0, \cdots, kz$$

Finally, $\zeta$ is sampled from a mixture of $kz + 1$ Gamma distributions:

$$\zeta \sim \sum_{r=0}^{kz} P_r \cdot \mathrm{Gamma}(a_0 + k + r - kz, d_0 - \sum_{q=1}^{kz}[\log(\eta_q)])$$

Delatola and Griffin (2011) provides an alternative way of updating $\zeta$ using the priors suggested by

Griffin (2010). We find our Gibbs sampling is more efficient than it.

- Update $W$: $W \sim \text{Beta}(n_0 + 0.1, n - n_0 + 0.9)$ for both $CP = 0$ and 1.

- Update $\phi$: We should apply M-H to sample. If $CP = 1$, the proposal is:

$$\text{N}(\frac{\sum_{t=1}^{n-1}(h_t^* - \mu_0)(h_{t+1}^* - \mu_0)}{\sum_{t=1}^{n-1}(h_t^* - \mu_0)^2}, \frac{\sigma_z^2}{\sum_{t=1}^{n-1}(h_t^* - \mu_0)^2})\mathbf{1}_{(-1,1)}$$

and the target density is proportional to:

$$e^{-(\phi)^2/20}(1-\phi^2)^{0.5}e^{-(1-\phi^2)(h_1^*-\mu_0)^2/(2\sigma_\eta^2)}[\prod_{t=2}^{n} e^{-[(h_t^*-\mu_0)-\phi(h_{t-1}^*-\mu_0)]^2/(2\sigma_\eta^2)}]$$

If $CP = 0$, then the proposal is:

$$\text{N}(\frac{\sum_{t=1}^{n-1} h_t h_{t+1}}{\sum_{t=1}^{n-1} h_t^2}, \frac{\sigma_z^2}{\sum_{t=1}^{n-1} h_t^2})\mathbf{1}_{(-1,1)}$$

And the target density is proportional to:

$$e^{-(\phi)^2/20}(1-\phi^2)^{0.5}e^{-(1-\phi^2)h_1^2/(2\sigma_\eta^2)}[\prod_{t=2}^{n} e^{-(h_t-\phi h_{t-1})^2/(2\sigma_\eta^2)}]$$

- Update $\sigma_\eta^2$: If $CP = 1$, we have:

$$\sigma_\eta^2 \sim IG(2.5 + \frac{n}{2}, 0.025 + \frac{(h_1^* - \mu_0)^2(1-\phi^2) + \sum_{t=1}^{n-1}[(h_{t+1}^* - \mu_0) - \phi(h_t^* - \mu_0)]^2}{2})$$

If $CP = 0$:

$$\sigma_\eta^2 \sim IG(2.5 + \frac{n}{2}, 0.025 + \frac{h_1^2(1-\phi^2) + \sum_{t=1}^{n-1}(h_{t+1} - \phi h_t)^2}{2})$$

- Transform $\boldsymbol{h}^*$ and $\mu_s^{**}$ to $\boldsymbol{h}$ and $\mu_s^*$ by the updated $\mu_0$ as $\boldsymbol{h} = \boldsymbol{h}^* - \mu_0$ and $\mu_s^* = \mu_s^{**} + \mu_0$.

# Appendix F

# Proof of $Theorem$-4 in in Section 3.5

We will apply the theorem 3 of Jones et al (2014) to prove this. We can denote all of the targets except $\psi$ which are sampled in our MCMC as $\Theta_{-\Psi}$(which takes the value of $\theta_{-\psi}$ as given condition for sampling $\psi$). The first step is to check if $\frac{f_{\Psi|\Theta_{-\Psi}}(\psi'|\theta^{'}_{-\psi})}{q(\psi'|\psi)}$, as a function of $\psi'$, is bounded for any given $\psi$ and $\theta^{'}_{-\psi}$. We have $f_{\Psi|\Theta_{-\Psi}}(\psi'|\theta^{'}_{-\psi}) \propto \pi_{\Psi}(\psi') \prod_{t=1}^{n} \pi_t(\psi', \gamma_t')$. The weights $\pi_t \in [0, 1]$, and under condition (c), we know that $f_{\Psi|\Theta_{-\Psi}}(\psi'|\theta^{'}_{-\psi})$ is bounded. Under condition (b) we know that $\frac{1}{q(\psi'|\psi)}$ is also bounded, therefore $\frac{f_{\Psi|\Theta_{-\Psi}}(\psi'|\theta^{'}_{-\psi})}{q(\psi'|\psi)}$ is bounded function of $\psi'$ in $[0, M]$ for any given $\psi$ and $\theta^{'}_{-\psi}$. The second step is to check the transition kernel $k(\psi', \theta'_{-\psi}|\psi, \theta_{-\psi}) = f_{\Theta_{-\Psi}|\Psi}(\theta'_{-\psi}|\psi) f_{\Psi|\Theta_{-\Psi}}(\psi'|\theta'_{-\psi})$. Based on the full joint distribution, we can see that $f_{\Theta_{-\Psi}|\Psi}(\theta'_{-\psi}|\psi)$ is bounded and positive in a compact subset of the support of $\Theta_{-\Psi}$. It is trivial to find a such compact subset with positive measure and we denote it as $B_{\Theta_{-\Psi}}$. Therefore the compact set $B \triangleq \left\{\theta_{-\psi}, \psi : \theta_{-\psi} \in B_{\Theta_{-\Psi}}, \psi \in [0, M]\right\}$ also has a positive measure. For any fixed $\theta'_{-\psi}$, we treat $f_{\Theta_{-\Psi}|\Psi}(\theta'_{-\psi}|\psi)$ as a function of $\psi$ defined on $[0, M]$, and this is a bounded function because the only relevant part is $\pi_{\Psi}(\psi) \prod_{t=1}^{n} \pi_t(\psi, \gamma_t)$. We can also see that this is a continuous function of $\psi$ in $[0, M]$. Therefore, based on extreme value theorem, we know that there exists $\psi_0 \in [0, M]$, such that $f_{\Theta_{-\Psi}|\Psi}(\theta'_{-\psi}|\psi) \geq f_{\Theta_{-\Psi}|\Psi}(\theta'_{-\psi}|\psi_0) > 0$. Since what value of $\psi$ minimizes $f_{\Theta_{-\Psi}|\Psi}(\theta'_{-\psi}|\psi)$ is determined by $\theta'_{-\psi}$, we can also write $\psi_0$ as $\psi_0(\theta'_{-\psi})$. We know that $f_{\Psi|\Theta_{-\Psi}}(\psi'|\theta'_{-\psi})$ is positive because both $f_{\Theta_{-\Psi}|\Psi}(\theta'_{-\psi}|\psi')$ and $\pi_{\Psi}(\psi')$ is positive. Denoting $g(\theta'_{\psi}, \psi') \triangleq f_{\Theta_{-\Psi}|\Psi}(\theta'_{-\psi}|\psi_0(\theta'_{-\psi})) f_{\Psi|\Theta_{-\Psi}}(\psi'|\theta'_{-\psi})$, we finally find the function

$g(\theta'_{\psi}, \psi')$ which takes positive value on $B$ ($\mu(B) > 0$) and satisfies:

$$k(\psi', \theta'_{-\psi}|\psi, \theta_{-\psi}) \geq g(\theta'_{\psi}, \psi')$$

Based on this inequality and that $\frac{f_{\Psi|\Theta_{-\Psi}}(\psi'|\theta^{'}_{-\psi})}{q(\psi'|\psi)}$ is bounded in the support of $\psi'$, we know that $P$ is uniformly ergodic.

# Appendix G

# Joint Distribution for Chapter 4

The joint distribution for G_WDPMP can be written as:

$$\pi(\boldsymbol{\theta}, \mu_0, \sigma_e^2, \boldsymbol{\gamma}, \psi, \zeta)\Pr(\boldsymbol{Z}|\boldsymbol{\gamma}, \psi)\Pr(\vec{k}|\zeta, \vec{n})\Pr(\boldsymbol{S}|\zeta, \vec{k}, \vec{n})f(\boldsymbol{y}|\boldsymbol{h}, \boldsymbol{\mu}^*, \boldsymbol{S}, \sigma_e^2)\prod_{s=1}^{k} dG_0(\mu_s^*|\mu_0, \sigma_e^2) \propto$$

$$e^{-\frac{1}{8}[\omega^2+(\rho_1-2)^2+(\rho_2-2)^2+(\tau+2)^2]-0.8\log(\psi)-0.1\psi-0.2\zeta-2\log(\sigma_e^2)-3/\sigma_e^2}$$

$$\times \prod_{t=1}^{n} \gamma^{b_{tq}-1}[\prod_{l=0}^{b_{tq}-1}(1-p_{(l)t})][1-\gamma_t(1-p_{(b_{tq})t})]$$

$$\times \zeta^{\sum_{j=1}^{kz} k_j}[\prod_{j=1}^{kz}\Gamma(\zeta)/\Gamma(\zeta+n_j)]$$

$$\times \prod_{t=1}^{n}\frac{1}{\sqrt{2h_t\alpha\sigma_e^2}}e^{-(y_t-\mu_{S_t}^*\sqrt{h_t})^2/(2h_t\alpha\sigma_e^2)}\prod_{s=1}^{k}\frac{1}{\sqrt{2(1-\alpha)\sigma_e^2}}e^{-(\mu_s^*-\mu_0)^2/[2\sigma_e^2(1-\alpha)]}$$

where $b_{tq}$ means that $Z_t = q$ and the $q^{th}$ candidate is the $b_{tq}^{th}$ nearest one to observation $t$. $\boldsymbol{S}$ is the label vector which assigns the $k$ unique values in $\boldsymbol{\mu}$ to the $n$ observations, and $\boldsymbol{\mu}^*$ is the vector of these unique values. Therefore $\mu_t$ can be rewritten as $\mu_{S_t}^*$. $kz$ is the number of clusters in $\boldsymbol{Z}$. $\vec{n} = (n_1, \cdots, n_{kz})$ and $\vec{k} = (k_1, \cdots, k_{kz})$ are the vectors which correspondingly contain the numbers of observations and the numbers of unique values of $\boldsymbol{\mu}$ in the $kz$ clusters.

The joint likelihood of other weighted GARCH models can be similarly derived replacing $\Pr(\boldsymbol{Z}|\boldsymbol{\gamma}, \psi)$ by other weight functions and changing the priors of hyper-parameters.

# Appendix H

# Procedures of MCMC Sampling for Chapter 4

The MCMC algorithm for G_WDPMP:

- Sample $\boldsymbol{\theta} \overset{\triangle}{=} (\omega, \rho_1, \rho_2, \tau)$ using M-H. The proposal is multivariate normal distribution $\mathrm{N}(\hat{\boldsymbol{\theta}}, \Sigma_{\hat{\boldsymbol{\theta}}})$ where $\hat{\boldsymbol{\theta}}$ maximizes $\prod_{t=1}^{n} \frac{1}{\sqrt{2h_t(\omega,\rho_1,\rho_2,\tau)\alpha\sigma_e^2}} e^{-(y_t-\mu_{S_t}^* \sqrt{h_t(\omega,\rho_1,\rho_2,\tau)})^2/(2h_t(\omega,\rho_1,\rho_2,\tau)\alpha\sigma_e^2)}$ and $\Sigma_{\hat{\boldsymbol{\theta}}}$ is the corresponding inverse of the negative of Hessian Matrix.
- Update $\mu_0 \sim \mathrm{N}(\frac{\sum_{s=1}^{k} \mu_s^*}{k}, \frac{(1-\alpha)\sigma_z^2}{k})$
- Update $\sigma_e^2 \sim \mathrm{IG}(\frac{n+k}{2} + 1, 3 + \frac{1}{2}[\frac{\sum_{S_t>0}(y_t/\sqrt{h_t}-\mu_s^*)^2}{\alpha} + \frac{\sum_{s=1}^{k}(\mu_s^*-\mu_0)^2}{1-\alpha}])$.
- Sample $\zeta$. Based on the joint likelihood, we can realize the sampling of $\zeta$ by introducing auxiliary variables:

$$\eta_q \sim \mathrm{Beta}(\zeta + 1, n_q) \quad q = 1, \cdots, kz$$

And derive the coefficients $(c_0, c_1, \cdots, c_q)$ which satisfy:

$$c_0 + c_1\zeta + \cdots + c_{kz}\zeta^{kz} \equiv \prod_{q=1}^{kz}(\zeta + n_q)$$

Using these coefficients, we can derive the probabilities:

$$P_r \propto \frac{c_r \cdot \Gamma(a_0 + k - kz + r)}{(d_0 - \sum_{q=1}^{kz}[\log(\eta_q)])^r} \quad r = 0, \cdots, kz$$

where $a_0$ and $d_0$ are the parameters that specify the Gamma prior of $\zeta$. $\zeta$ can be sampled via Gibbs sampling:

$$\zeta \sim \sum_{r=0}^{kz} P_r \cdot \text{Gamma}(a_0 + k + r - kz, d_0 - \sum_{q=1}^{kz}[\log(\eta_q)])$$

- Update $\boldsymbol{S}$:

$$\Pr(S_t = s|Z_t = q) \propto \begin{cases} \frac{n'_{s(-t)}}{n_q+\zeta-1}(\alpha\sigma_z^2)^{-0.5}e^{-(y_t/\sqrt{h_t}-\mu_s^*)^2/(2\alpha\sigma_e^2)} & s \in \left\{q_1, \cdots, q_{k_j}\right\} \\ \frac{\zeta}{n_q+\zeta-1}(\sigma_e^2)^{-0.5}e^{-(y_t/\sqrt{h_t}-\mu_0)^2/(2\sigma_z^2)} & s = \max(\boldsymbol{S}^{-t}) + 1 \end{cases}$$

where $n_q$ is the number of observations who select the $q_{th}$ candidate prior (denote the subset of $\boldsymbol{S}$ including these observations as $\boldsymbol{S}_{Z=q}$ ), and $n'_{s(-t)}$ is the number of observations except the $t_{th}$ one that belong to the $s_{th}$ unique value in $\boldsymbol{\mu}$. $\boldsymbol{S}^{-t}$ means $\boldsymbol{S}/\left\{\boldsymbol{S}_t\right\}$ and $\left\{q_1, \cdots, q_{k_j}\right\}$ is the set of unique values in the intersection of $\boldsymbol{S}^{-t}$ and $\boldsymbol{S}_{Z=q}$.

- Update $\boldsymbol{\mu}$: The unique values in $\boldsymbol{\mu}$ can be sampled as:

$$\mu_s^* \sim \text{N}(\frac{\frac{\sum_{S_t=s}(y_t/\sqrt{h_t})}{\alpha} + \frac{\mu_0}{1-\alpha}}{\frac{n'_s}{\alpha} + \frac{1}{1-\alpha}}, \frac{\sigma_z^2}{\frac{n_s}{\alpha} + \frac{1}{1-\alpha}}) \quad s = 1, \cdots, k$$

where $n'_S$ is the number of observations that belong to the $s_{th}$ unique value. Then we can assign these unique values to $\boldsymbol{\mu}$ based on $\boldsymbol{S}$.

- Sample $\boldsymbol{\gamma}$: $(1 - p_{(b_{tq})t})\gamma_t \sim \text{Beta}(b_{tq}, 2)\mathbf{1}_{[0,1-p_{(q_t)t}]}$ for $t = 1, \cdots, n$.

- Sample $\psi$ using M-H. The proposal is $q(\psi_{new}|\psi) \propto \text{N}(\psi_{new}|\psi, 1) \cdot \mathbf{1}_{[0,4]}$ and the target density is proportional to $e^{-\psi}\text{Pr}(\boldsymbol{Z}|\psi)$. Notice that we using the marginal density by integrating $\text{Pr}(\boldsymbol{Z}|\boldsymbol{\gamma}, \psi)$ over $\boldsymbol{\gamma}$.

- Sample $\boldsymbol{Z}$. We introduce $\boldsymbol{D}$ as the vector allocating the unique values in $\boldsymbol{\lambda}$ to the $Q$ candidate priors. $D_1 = q$ means that all of the observations who share the first unique value in $\boldsymbol{\lambda}$ share the $q^{th}$ candidate. Sampling $\boldsymbol{D}$ is equivalent to sampling $\boldsymbol{Z}$ because the observations sharing the same unique value in $\boldsymbol{\lambda}$ can never come from different candidate priors. The posterior of $\boldsymbol{D}$ is:

$$p(D_j = q|\boldsymbol{D}_{(-j)}, \boldsymbol{S}, \boldsymbol{y}, \boldsymbol{\gamma}, \psi) \propto \frac{\Gamma[(\zeta + n_{q(-j)}]}{\Gamma[\zeta + n_{q(-j)} + n_{(j)}]} \prod_{i=1}^{n_{(j)}} \pi_{d_i q} \quad q = 1, \cdots, Q$$

where $n_{(j)}$ is the number of observations assigned to the $j^{th}$ unique value of $\boldsymbol{\gamma}$. We denote these observations are the $d_1, d_2, \cdots, d_{n_{(j)}}$-th observations in the original data. $n_{q(-j)}$ is the number of observations other than these who share the $q^{th}$ candidate prior. $\pi_{d_i q}$ is the probability that the $d_i^{th}$ observation select candidate $q$. This probability is determined by weight function and hyper-parameters.