Department of Mathematics and Statistics

On Generalised Additive Model Using MCMC Model Selection and Simulated Annealing

Zeyu Zhou

Supervised by
Dr. Guoqi Qian

October 2011
To my dear parents,

Zhou Zhen Bang,
& Pan Li Da.
Abstract

In supervised learning, an important problem is to select a subset of $p$ candidate predicting variables to best explain the underlying true model. Markov Chain Monte Carlo (MCMC) implements an information model selection criterion, such as AIC & BIC. But it has the potential to deal with the cases where up to $2^{2000}$ candidate models available for selection, which cannot be done by using an information criterion in a standard way.

In this thesis, an MCMC model selection diagnosis toolkit using Generalized Additive Models (GAM) with local polynomial fitting is presented. Our original contribution is to implement simulated annealing techniques into the MCMC model selection procedure. By different ways of controlling the temperature parameter $T$, the diagnosis toolkit is capable of dealing with much larger $p$ (e.g. $p = 100$), accelerating the standard MCMC model selection process and extracting more information of important predicting variables.
Acknowledgments

I am heartily thankful to my supervisor, Guoqi Qian, whose encouragement, guidance and support from the initial to the final level enabled me to develop this thesis.

Lastly, I offer my regards and blessings to all of those who supported me in any respect during the completion of the project.

Zeyu Zhou
# Contents

1 Introduction ................................................. 1  
  1.1 Introduction ........................................... 1  
  1.2 Generalized additive models with local polynomial fitting ... 3  
  1.3 MCMC model selection procedures ........................ 4  
  1.4 Diagnose toolkit using simulated annealing ............. 5  
  1.5 Simulation Study ........................................ 6  

2 GAM with local polynomial fitting ......................... 9  
  2.1 Local Polynomial Fitting ............................... 9  
    2.1.1 Framework ........................................ 9  
    2.1.2 Bias and variance ................................ 13  
    2.1.3 Equivalent Kernels ............................... 14  
    2.1.4 Ideal selection of optimal bandwidth ............. 18  
    2.1.5 Optimal bandwidths used practical approach ...... 19  
    2.1.6 Some remarks on application ...................... 19  
  2.2 Generalized Additive Models ............................ 20  
    2.2.1 Framework ........................................ 20  
    2.2.2 Local Polynomial as a smoother .................. 21  
    2.2.3 Fitting the model ................................ 21  
    2.2.4 Estimating the degrees of freedom ............... 22  

3 Markov Chain Monte Carlo ................................ 25  
  3.1 Some preliminaries on Markov chain..................... 25  
  3.2 Gibbs Sampler ......................................... 30  
    3.2.1 Gibbs Sampler Algorithm ........................ 30
List of Tables

2.1 Widely used kernel functions ........................................... 13

4.1 Variance comparation between A.O & A.R. ......................... 64

5.1 5 best models in exhaustive search ................................. 74
5.2 Gelman & Rubin’s convergence test results ....................... 75
5.3 Marginal frequencies in FTSs ......................................... 77
5.4 Best models in CSS with \( N = 10000 \) .......................... 81
5.5 Marginal Frequencies in CSS ......................................... 81
5.6 Average Odds under temperatures on 1000-chain ............... 82
5.7 10 Best Models in FTS with temperature \( T = 1 \) ............... 83
5.8 Marginal frequencies in FTSs ......................................... 83
5.9 10 Best models in CSS ................................................ 84
5.10 Marginal Frequency of selected variables in CSS ............... 84
5.11 Comparisons between different machine learning methods .... 87
List of Figures

2.1 An example of local polynomial fitting 11
4.1 Another convergence diagnose example 56
4.2 Relationship between A.O & A.R 62
4.3 Convergence property of A.O & A.R. 65
5.1 Artificial dataset summary 73
5.4 Trend of Marginal Frequencies in FTSs 78
5.5 Convergence of Marginal frequencies when T = 8 79
5.7 Plot of two important predicting variables from diagnostic toolbox 85
5.2 I-chart convergence test 89
5.3 Distribution of best 50 models in FTSs 90
5.6 Movement BIC values in a cooling schedule 91
Chapter 1

Introduction

1.1 Introduction

This thesis investigates an optimisation approach for statistical model selection using Markov Chain Monte Carlo (MCMC). Statistical model selection is an important supervised learning method having applications in many data mining related disciplines. We have found that simulated annealing, a stochastic global optimisation technique, can be used to significantly improve the efficiency and effectiveness in statistical model selection. We have also found a significant connection between simulated annealing and Gibbs sampler, a founding MCMC method, in the context of model selection. All these will be explored in this thesis.

Thanks to modern information technology, it is much easier to collect and process data nowadays than before. At the same time, the complexity and quantity of the collected data also grow exponentially, posing a challenger to statisticians and data analysts on effectively analysing them. To take on this challenge, a new discipline, called data mining also called machine learning or statistical learning, has been rapidly developed in recent 15 years by statisticians, mathematicians and computing scientists. The discipline provides a collection of new data analysis methods to tackle high-dimensional and nonstandard data, and is still growing.
This thesis aims to bring together some model selection methods, especially those involving a large number of candidate models.

Use of model selection is required in at least two different situations. The first is where a model with the best prediction ability is looked for. This task may not be achievable by utilising the information provided from the dataset alone. Knowledge external to the dataset may also provide insights on the model building process, which is often the case in applications such as physics and biology. This perspective underlies the second way of using model selection where one just finds a subset of available explanatory variables which best supervise the system mechanism of the response variable. The actual model detailing the relationship between the selected explanatory variables and the response variable is to be established by further utilising the subject information underlying the problem. This thesis is to focus on the variable selection aspect of model selection with model predictability also being considered but as secondary.

We have constructed a particular MCMC model selection procedure in the context of selecting generalized additive models (GAM). In the procedure, each candidate GAM generated is fit by local polynomials of relevant explanatory variables, and is assessed by model selection criteria AIC or BIC, where models having smaller AIC and BIC values are ranked better than those having larger values. When the model space contains an astronomical number of candidate models, it is computationally infeasible to fit and assess all of them for model selection. Instead, a Gibbs sampler is used to generate a random sample of candidate models from the model space. The probability distribution governing the random generation of the candidate models is determined by the model selection criterion used. This ensures that the model having the smallest AIC or BIC value has probability of 1 in the limit to be included in the generated sample under strong regulation setting. It also ensures that the odds of generating effecting explanatory variables against non-effecting ones in the generated model sequence goes to $\infty$ in the limit under weak regulation setting.
In addition to the model selection procedure we have developed a diagnostic toolkit to monitor and facilitate the convergence of the Markov chain generation involved. The pivot of this toolkit is the temperature parameter which has been widely used in simulated annealing to tune the convergence rate and search depth in global optimisation. We have extended the use of such temperature parameter to suit Gibbs sampler model selection, resulting in two search algorithms Fixed Temperature Search (FTS) and Cooling Schedules Search (CSS). It has been found that FTS is more powerful on identifying all effecting explanatory variables from the available, whereas CSS is more powerful on finding the best candidate model.

1.2 Generalized additive models with local polynomial fitting

Let \( \{Y, X_1, X_2, \ldots, X_p\} \) be a data underlying regression problem. The ordinary way to model this problem is through solve it is linear regression

\[
E(Y) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p
\]

To make allowance for different probability distributions for \( Y \), we may instead use generalized linear model

\[
E(Y) = g^{-1}(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p)
\]

where \( g^{-1} \) is a given link function. Generalised additive model further extends the generalised linear model by replacing each \( X_i \) with some function \( f_i(X_i) \), resulting in the following non-parametric form using non-parametric models.

\[
E(Y) = g^{-1}\left(\alpha + f_1(X_1) + f_2(X_2) + \cdots + f_p(X_p)\right)
\]
The above GAM can be fit by a local polynomial weighted least squares approach. The local polynomial fitting is some weighted local fitting involving a bandwidth defining the neighbour, a degree of freedom indicating the complexity of local polynomial, and a kernel function specifying the actual weight of each data point. Comparing with the usual kernel smoothing method, local polynomial fitting has less variance and boundary effects, but has more parameters to be estimated and is more computing-intensive especially when the degree of polynomial is high.

1.3 MCMC model selection procedures

MCMC model selection is a stochastic search procedure. Given a model space where each model is identified by a subset of explanatory variables formulated as a GAM, the idea of MCMC model selection is to simulate a Markov Chain from the model space with the property that the best models appear more often in the chain than the others when equilibrium is reached. By best models, we mean they have the smallest values in terms of an information criterion; e.g. AIC or BIC, etc.

Specifically, suppose there is a particular model space \( \mathcal{A} \) consisting of all candidate generalized additive models for the given data. We construct a discrete probability measure on this model space according to, say, Akaike Information Criteria (AIC) of each model.

\[
P(\alpha) = \frac{\exp(-AIC_\alpha)}{H} \quad \text{for } \alpha \in \mathcal{A}
\]

where \( H = \sum_{\alpha' \in \mathcal{A}} \exp(-AIC_{\alpha'}) \). Obviously, a model with lower AIC value has a large probability. However, it may be infeasible to evaluate \( H \) when \( \mathcal{A} \) has an astronomical number of models.

Let’s say there are \( p \) explanatory variables. Each model in \( \mathcal{A} \) can be denoted by a \( p \) dimensional binary vector, where the \( i \)th entry of the vector equals either 1 or 0 indicate whether or not the \( i \)th explanatory variable is in the
1.4. DIAGNOSE TOOLKIT USING SIMULATED ANNEALING

model. There will be up to $2^p$ models in $\mathcal{A}$, implying it is difficult to evaluate $H$ and generate a sample directly from $P(\alpha)$. However, Gibbs sampler can be used instead to generate a Markov chain from $\mathcal{A}$, where only conditional distributions of the $p$-dimensional binary vector are required and the evaluation of $H$ is not required. The stationary distribution of the Markov chain generated can be shown to be $P(\alpha)$.

Analyzing the simulated Markov chain enable us to select the best model. There are usually two options. One is to pick the model with the lowest AIC or BIC value in the chain. Another is to select a sub set of variables each of which appears more often in the models in the chain, i.e. with a higher marginal frequency than variables not selected. By this stochastic search, MCMC model selection is capable of dealing with a model space with large $p$. Since a better model has greater chance to appear in the chain and appear early in the chain. Comparing to exponential order of the size of $\mathcal{A}$, the size of the simulated Markov chain is expected to be of a polynomial order before the best models are generated.

1.4 Diagnose toolkit using simulated annealing

Although MCMC model selection is good at handling model space with large $p$, there are some practical issues to be addressed when $p$ keeps growing, for example, $p > 50$. The expected length of informative chain required increases in a polynomial rate when $p$ increases. In some cases, when the goodness of models are pretty close to each other in terms of AIC or BIC, the expected time to see the best model can be very long. In other words, when we have a very large $p$, the only way to gain confidence of our current model is to run a much longer chain. This could be very painful.

In this thesis, we present a diagnostic toolkit to ease these problems. Simulated annealing techniques is implemented to enhance the performance of MCMC model selection by accelerating the convergence. By adding a parameter
temperature $T$, simulated annealing allows us to alter the structure of the
Gibbs measure in the direction of our preference so that good models have
additional weights to be generated.

We present two strategies to utilize the annealing technique. One is Fixed
Temperature Search (FTS) where a suitable temperature is used for MCMC.
Another is Cooling Schedules Search (CSS) with a periodic temperature func-
tion. The first strategy would accelerate the convergence of Markov chain
to its equilibrium if using some cold temperature but still preserve the aper-
tiodic Markov chain property, while the second is very good at finding local optimums. Further more, in the marginal frequency search, we have discov-
ered that the marginal frequency of an important variable tends to increase
when temperature decreases. This property can be used to diagnose impor-
tant variables with relative less computational cost.

In addition, a term called Average Odds is introduced in this thesis to eval-
uate the sensitivity of MCMC to different temperature settings. It is shown
that only relatively short sequence of models need to be generated to obtain
a reliable estimation of it. In summary, our toolkit allows us to search and
diagonse with more efficiency and effectiveness.

1.5 Simulation Study

There are three datasets in this thesis which we have used to study the prop-
erties of MCMC under different temperature settings, the ability of choosing
model with large number of candidate variables, and the models' prediction
ability.

The first dataset is an artificially simulated regression problem, so that
the true model is known to us. An exhaustive search is then used to obtain
the full information of the model space, which literally does regression for
every model in the model space. Base on that result, we find out that BIC
1.5. SIMULATION STUDY

successfully picks out the model with all and only the true effecting variables, while AIC selects some additional noise variables.

Further more, given a specific temperature $T$, the actual distribution $P_T(\alpha)$ used in Gibbs sampler can be calculated. To investigate how FTS imporves MCMC model selection, a comparison study between $P_T(\alpha)$ and the empirical distributions deduced from simulated sample chains under different temperature settings is done. Consistent with what is expected from an asymptotic analysis, when temperature is high, the sampler tends to choose as many good models as it can ignoring their difference; when temperature is low, the sampler tends to stay around a local optimum. The combination of GAM with local polynomial fitting and BIC successfully picked out the true model in the data set. In the marginal frequency search, we found an interesting phenomenon that for true effecting variables, their marginal frequencies increase as the temperature decreases. We will provide some explanation and use it as an informative indicator in studying the other two datasets.

As for CSS, we setup a schedule periodically moving from high temperature to cold. And it turns out to be as good at finding out the model with lowest BIC values as FTS, but much faster, and the true model appears more often.

The second dataset, community crime study, has a response variable crime rate in each community and 101 explanatory variables which are various measures and attributes about that community. Many of the explanatory variables are highly correlated of each other. It would take a very long chain to reach the Markov chain equilibrium in default MCMC model selection setting. Actually, the best models are not attainable in this setting. However, using our toolkit, we are able to pick models and important variables with high confidence. In this study, we are interested in finding out the most important variables having effects on the crime rate, which can be used in further study. It turns out that the annealing MCMC model selection has done a satisfactory job here.
The last dataset is email-spam dataset. It is has been analysed by many data mining methods in [Hastie et al. [2005]]. Using the MCMC model selection procedure, we have obtained results that are competent with those by using the other data mining methods.
Chapter 2

GAM with local polynomial fitting

2.1 Local Polynomial Fitting

Local polynomial fitting was systematically studied by Stone [1977, 1980] and [Cleveland, 1979]. Most of studies in local polynomial regression can be found in the monographs written by Fan and Gijbels [1996]. Besides, the software LOCFIT and some nice applications in Loader [1999] are also worth notice. In this section, we are going to review the framework of local polynomial fitting and introduce some computing procedures using statistical language R. This section is essentially a review on Fan and Gijbels [1996]'s book.

2.1.1 Framework

Suppose \((x_i, y_i), i = 1, \ldots, n\) are i.i.d. samples from a population \((X, Y)\). We are interested in estimating the regression function

\[
m(x) = E(Y|X = x) \quad \& \quad \sigma^2 = \text{Var}(Y|X = x)
\]  

(2.1)

This implies the population admits the following model:

\[
Y = m(X) + \sigma(X)\epsilon
\]  

(2.2)
where $E(\epsilon) = 0$, $\text{Var}(\epsilon) = 1$, and $X$ and $\epsilon$ are assumed to be independent of each other.

In local polynomial fitting polynomials are used to approximate the function $m(x)$ at around each $x_0$ locally. Suppose the $(k + 1)^{th}$ derivative of $m(x)$ exists at the point $x_0$. Then by Taylor expansion, we have

$$m(x) \approx m(x_0) + m'(x_0)(x - x_0) + \frac{m''(x_0)}{2!}(x - x_0)^2 + \cdots + \frac{m^{(k)}(x_0)}{k!}(x - x_0)^k + \frac{m^{(k+1)}(\tilde{x})}{(k+1)!}(x - x_0)^{k+1}$$

(2.3)

where $\tilde{x}$ is between $x$ and $x_0$.

This results in the local polynomial model for each $x_0$:

$$m(x) = \beta_0 + \beta_1(x - x_0) + \beta_2(x - x_0)^2 + \cdots + \beta_p(x - x_0)^k$$

(2.4)

where $m^{(v)} = v!\beta_v$, $v = 1, 2, \ldots k$.

This model is fitted by minimizing the locally weighted least squares:

$$\min \sum_{i=1}^{n} \left( y_i - \sum_{j=0}^{p} \beta_j(x_i - x_0)^j \right)^2 K_h(x_i - x_0)$$

(2.5)

where the weight is determined by a kernel function $K_h = K(\frac{x - x_0}{h})/h$ with the bandwidth parameter $h$ controlling the size of the neighborhood.

We introduce some matrix notation here. In matrix form,

$$X = \begin{pmatrix}
1 & (x_1 - x_0) & \cdots & (x_1 - x_0)^k \\
\vdots & \vdots & & \vdots \\
1 & (x_n - x_0) & \cdots & (x_n - x_0)^k
\end{pmatrix}, \quad (2.6)$$
and  

\[ y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \quad \text{and} \quad \beta = \begin{pmatrix} \beta_0 \\ \vdots \\ \beta_k \end{pmatrix} \]  

(2.7)

Then (2.5) can be written as:

\[ \min_{\beta} (y - X\beta)^TW(y - X\beta) \]  

(2.8)

where \( W = \text{diag}\{K_h(x_i - x_0)\} \).

This is a typical weighted linear regression problem. Hence, we obtain our estimated coefficients \( \hat{\beta} \) as

\[ \hat{\beta} = (X^TWX)^{-1}X^TWy. \]  

(2.9)

Figure 2.1: An example of local polynomial fitting

In fact, this is only an estimation of \( Y \) at one particular point \( x_0 \). In order
to give a curve prediction of $Y$ within a range of $X$, we make a partition of range, and do a local polynomial fit on the partition. Figure 2.1 demonstrates how local polynomial method fit a continuous function $f(x) = 2x + \frac{\exp(-64x^2)}{3}$ with degree of 2. The solid line is the fitted curve, while the dashed line suggests the real mean curve of $Y$. Local polynomial doesn’t very much capture the peak of $Y$ around 0 pretty good. However, this can be improved by adjusting parameters, which we will see in later chapters when we perform model selection.

There are three major parameters in controlling the local polynomial fitting: choice of bandwidth $h$, choice of order of the local polynomial $k$, and choice of kernel function $K$.

- Bandwidth $h$ plays a critical role here. It controls the smoothness of the fitting. It can be a simple constant or change dynamically according to the density of data, e.g. a nearest neighbor bandwidth uses the $h$ under which the number of data points in the associated neighbour is a given constant.

- The choice of order $k$ has a minor effect on the bias-variance trade-off unlike the bandwidth does. The higher the order the better approximation we get, but will not change in a continuous way. In practice, we mostly choose $k$ to be less than 4.

- Kernel function is typically chosen to be a bell-shaped function. Gasser et al. [1985] shows that Epanechnikov kernel, $K(x) = \frac{3}{4}(1 - x^2)_{+}$, asymptotically minimizes MSE in the context of local polynomial fitting. Table 2.1 lists some popular kernel functions used in practice. The Gaussian function has a domain over $\mathbb{R}$ while others are defined on interval $(-1, 1)$. However, in practice, the benefit from choosing the right kernel function is quite limited, not as important as bandwidth. Thus choosing an optimal kernel function is usually not the first option to reduce the prediction error.
2.1 LOCAL POLYNOMIAL FITTING

<table>
<thead>
<tr>
<th>Name</th>
<th>$K(x)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epanechnikov</td>
<td>$\frac{3}{4}(1-x^2)I_{</td>
</tr>
<tr>
<td>Gaussian</td>
<td>$\frac{1}{\sqrt{2\pi}}\exp\left{-\frac{x^2}{2}\right}$</td>
</tr>
<tr>
<td>Biweight</td>
<td>$\frac{15}{32}(1-x^2)^2I_{</td>
</tr>
<tr>
<td>Triweight</td>
<td>$\frac{1}{2}I_{</td>
</tr>
<tr>
<td>Uniform</td>
<td>$\frac{70}{80}(1-</td>
</tr>
</tbody>
</table>

Table 2.1: Widely used kernel functions

2.1.2 Bias and variance

The mean and variance of $\hat{\beta}$ in (2.9) are

$$E(\hat{\beta}|X) = (X^TWX)^{-1}X^T W m$$
$$= \beta + (X^TWX)^{-1}X^T W r$$
$$\text{var}(\hat{\beta}|X) = (X^TWX)^{-1}(X^T \Sigma X)(X^TWX)^{-1}$$

where $m = \{m(x_1), \ldots, m(x_n)\}^T$, $\beta = \{m(x_0), \ldots, m^{(k)}(x_0)/k!\}^T$, $r = m - X\beta$, and $\Sigma = \text{diag}\left\{K^2_h(x_i-x_0)\sigma^2(x_i)\right\}$.

The conditional mean and variance of the fitted mean $\hat{m}(x_0)$ are however difficult to find. But their asymptotic approximation, can be derived. To present the asymptotic results, we use the following notations:

$$\mu_j = \int u^j K(u)du \quad \text{and} \quad \nu_j = \int u^j K^2(u)du$$

$S = (s)_{j,l}$, $\tilde{S} = (\tilde{s})_{j,l}$, $S^* = (s)_{j,l}$ are matrices with

$$s_{j,l} = \mu_{j+l} \text{ for } 0 \leq j, l \leq k$$
$$\tilde{s}_{j,l} = \mu_{j+l+1} \text{ for } 0 \leq j, l \leq k$$
$$s^*_{j,l} = \nu_{j+l} \text{ for } 0 \leq j, l \leq k$$

$c_k = (\mu_{k+1}, \ldots, \mu_{2k+1})^T$
$$\tilde{c}_k = (\mu_{k+2}, \ldots, \mu_{2k+2})^T$$
Furthermore, \( e_{\nu+1} = (0, \ldots, 0, 1, 0, \ldots, 0)^T \) a unit vector with 1 on the \((\nu + 1)^{th}\) position.

**Theorem 2.1.** by [Fan and Gijbels 1996]. Assume \( X \) has a pdf \( f(x) \) which together with \( m^{(k+1)} \) and \( \sigma^2(\cdot) \) are continuous around \( x_0 \). Furthermore, assume \( h \to 0 \) and \( nh \to \infty \). Let \( \hat{m}_\nu(x_0) \) be the local polynomial estimator on \( x_0 \). Then, the asymptotic conditional variance of \( \hat{m}_\nu(x_0) \) is given by

\[
\text{Var} \left\{ \hat{m}_\nu(x_0) | X \right\} = e_{\nu+1}^T S^{-1} S^* S^{-1} \frac{\nu! \sigma^2(x_0)}{f(x_0) nh^{1+2\nu}} + o_P \left( \frac{1}{nh^{1+2\nu}} \right) \tag{2.12}
\]

The asymptotic conditional bias when \( p - \nu \) is odd is

\[
\text{Bias} \left\{ \hat{m}_\nu(x_0) | X \right\} = e_{\nu+1}^T S^{-1} S^* S^{-1} \frac{\nu!}{(p + 1)!} m^{(p+1)}(x_0) h^{k+1-\nu} + o_P(h^{k+1-\nu}) \tag{2.13}
\]

When \( p - \nu \) is even, the asymptotic conditional bias is

\[
\text{Bias} \left\{ \hat{m}_\nu(x_0) | X \right\} = e_{\nu+1}^T S^{-1} S^* S^{-1} \frac{\nu!}{(k+2)!} \left\{ m^{(k+2)}(x_0) \right\} + (k+2) m^{(k+1)}(x_0) \frac{f'(x_0)}{f(x_0)} \} h^{k+1-\nu} + o_P(h^{k+2-\nu}), \tag{2.14}
\]

provided that \( f'(\cdot) \) and \( m^{(k+2)}(\cdot) \) are continuous in a neighborhood of \( x_0 \) and \( nh^3 \to \infty \).

### 2.1.3 Equivalent Kernels

A major property of local polynomial fitting is asymptotically equivalent to a kernel smoothing method [Fan and Gijbels 1996]. In this subsection, we filled some gaps of the derivation in that book. At a fixed data point \( x_0 \), we have the estimation:

\[
\hat{m}(x_0) = \hat{\beta}_0 = e_1^T (X^T W X)^{-1} X^T W y
\]
2.1. LOCAL POLYNOMIAL FITTING

where \( e_1 \) is the \((k+1) \times 1\) vector with first entry is 1 and zero elsewhere.

Denote \( S_n = X^T W X \).

\[
\hat{\beta}_0 = e_1^T \hat{\beta} = e_1^T S_n^{-1} X^T W y \\
= e_1^T S_n^{-1} \begin{pmatrix} 1 & \ldots & 1 \\ \vdots & \ddots & \vdots \\ (x_1 - x_0)^k & \ldots & (x_n - x_0)^k \end{pmatrix} \text{diag}\{K_h(x_i - x_0)\} y \\
= e_1^T S_n^{-1} \begin{pmatrix} K \left( \frac{x_1 - x_0}{h} \right) h^{-1} & \ldots & K \left( \frac{x_n - x_0}{h} \right) h^{-1} \\ K \left( \frac{x_1}{h} \right) h^{-1} \left( \frac{x_1 - x_0}{h} \right) & \ldots & K \left( \frac{x_n}{h} \right) h^{-1} \left( \frac{x_n - x_0}{h} \right) \end{pmatrix} y \\
= e_1^T S_n^{-1} \begin{pmatrix} \sum_i K \left( \frac{x_i - x_0}{h} \right) h^{-1} y_i \\ \sum_i K \left( \frac{x_i}{h} \right) h^{-1} \left( \frac{x_i - x_0}{h} \right)^k y_i \end{pmatrix} \\
= \sum_i e_1^T S_n^{-1} \begin{pmatrix} \frac{x_i - x_0}{h} \hfill h^{-1} y_i \\ \frac{x_i}{h} \hfill \left( \frac{x_i - x_0}{h} \right)^k \hfill h^{-1} y_i \end{pmatrix} \\
= \sum_{i=1}^n W_1^n \left( \frac{x_i - x_0}{h} \right) y_i \\
\tag{2.15}
\]

where \( W_1^n(t) = e_1^T S_n^{-1} \{1, t, \ldots, (t^h)^k\}^T K(t) h^{-1} \).

This \( W_1^n \) corresponds to the kernel function used in kernel smoothing. Instead of being a constant, \( W_1^n \) is actually determined by all \( \{x_i\} \) and the location of \( x_0 \). This property explains why local polynomial fit can adapt automatically to different designs and boundary conditions.
Recall, $S_n$ is a $(k+1) \times (k+1)$ matrix with $(s_{n,j+l})_{0 \leq j,l \leq k}$. We denote $S_{n,j}$ as the $j$th row of matrix $S_n$:

$$S_{n,j} = \sum_{i=1}^{n} K_h(x_i - x_0)(x_i - x_0)^j$$  \hspace{1cm} (2.16)

We assume all $\{x_i\}$ are i.i.d. with density function $f(x)$. We set $u = \frac{x_1 - x_0}{h}$. To study the asymptotic properties, we assume $h \to 0$ and $nh \to \infty$. It follows that ([Fan and Gijbels, 1996])

$$S_{n,j} = ES_{n,j} + O_P \left\{ \sqrt{\text{Var}(S_{n,j})} \right\}$$

$$= n \int K(\frac{x - x_0}{h})/h \cdot h^j \left( \frac{x - x_0}{h} \right)^j f(x) \, dx$$

$$+ O_P \left( \sqrt{n} \mathbb{E} \left\{ (x_1 - x_0)^{2j} K_h^2 (x_1 - x_0) \right\} + \left( \mathbb{E} K_h (x_1 - x_0)(x_1 - x_0)^2 \right)^2 \right)$$

$$= nh^j \int u^j K(u) f(x_0 + hu) \, du$$

$$+ O_P \left( \sqrt{n} \mathbb{E} \left\{ (x_1 - x_0)^{2j} K_h^2 (x_1 - x_0) \right\} \right).$$

The first term in the last equation can be simplified using the property

$$\int u^j K(u) f(x_0 + hu) \, du$$

$$= f(x_0) \int u^j K(u) du + O \left( h \int u^{j+1} K(u) f'(x_0) \, du \right)$$

$$= f(x_0) \mu_j + O(h).$$
Also,

\[
O_p \left( \sqrt{n} \mathbb{E} \left\{ (x_1 - x_0)^{2j} K_h^2 (x_1 - x_0) \right\} \right)
\]

\[
= O_p \left( \sqrt{n} \int (x - x_0)^{2j} K^2 \frac{(x - x_0)}{h} / h^2 f(x) dx \right)
\]

\[
= n h' O_p \left( \sqrt{\frac{1}{n h}} \int u^{2j} K^2(u) f(x_0 + h u) du \right)
\]

\[
= n h' O_p \left( \frac{1}{\sqrt{n h}} \right).
\]

It follows that \( S_{n,j} \):

\[
S_{n,j} = n h' \left\{ f(x_0) \mu_j + o(1) + O_p \left( 1 / \sqrt{n h} \right) \right\}
\]

\[
= n h' f(x_0) \mu_j \left\{ 1 + o_p(1) \right\}
\]

(2.17)

The following asymptotic form of matrix \( S_n \) can then be obtained,

\[
S_n = n f(x_0) H S H \left\{ 1 + o_p(1) \right\}
\]

(2.18)

where \( H = \text{diag}(1, h, \ldots, h^p) \). Hence, we can update \( W_1^n \) to

\[
W_1^n(t) = \frac{1}{n h f(x_0)} e^T S^{-1} (1, t, \ldots, t^k)^T K(t) \{ 1 + o_p(1) \}
\]

Therefore,

\[
\hat{\beta}_0 = \frac{1}{n h f(x_0)} \sum_{i=1}^{n} K^*_1 \left( \frac{x_i - x_0}{h} \right) y_i \left\{ 1 + o_p(1) \right\},
\]

(2.19)

where

\[
K^*_1(t) = e^T + S^{-1} (1, t, \ldots, t^k)^T K(t),
\]

(2.20)
2.1.4 Ideal selection of optimal bandwidth

There are several options to deal with the choice of bandwidth: a constant bandwidth (global bandwidth), global variable bandwidth, and local variable bandwidth. Here we discuss about the local variable bandwidth. The idea behind this is to shrink at the peaks and expand in the flat region. The optimal local bandwidth is obtained by minimizing the conditional Mean Squared Error (MSE) given by

$$\left[ \text{Bias} \left\{ \hat{m}(x_0)|X \right\} \right]^2 + \text{Var} \left\{ \hat{m}(x_0)|X \right\}$$

Using the result for MSE, the optimal local bandwidth minimizing MSE has the form

$$h_{opt}(x_0) = C_k(K) \left[ \frac{\sigma^2(x_0)}{\left\{ m^{(k+1)}(x_0) \right\} 2f(x_0)} \right]^{1/(2k+3)} n^{-1/(2k+3)} \quad (2.21)$$

where

$$C_k(K) = \left[ \frac{(k + 1)! \int K_1^*(t) dt}{2(k + 1) \left\{ \int t^{k+1}K_1^*(t) dt \right\}^2} \right]^{1/(2k+3)}$$

It is easy to compute $C_k$, since it only depends on the kernel function $K$. Another version of $h_{opt}$ can be obtained by minimizing the conditional weighted MISE:

$$\int \left( \left[ \text{Bias} \left\{ \hat{m}_v(x_0)|X \right\} \right]^2 + \text{Var} \left\{ \hat{m}_v(x_0)|X \right\} \right) w(x) dx$$

The ideal asymptotic results under MISE is:

$$h_{opt} = C_k(K) \left[ \frac{\int \sigma^2(x)w(x)/f(x) dx}{\int \left\{ m^{(k+1)}(x) \right\} 2w(x) dx} \right]^{1/(2k+3)} n^{-1/(2k+3)} \quad (2.22)$$

Both (2.21) and (2.22) are called the ideal version of $h_{opt}$ as it involves the unknown terms $\sigma^2, m^{(k+1)}$ and $f(\cdot)$. 
### 2.1.5 Optimal bandwidths used practical approach

There are two possible ways to implement the ideal $h_{opt}$ in practice. Plug-in Method which replaces those unknown terms in (2.21) and (2.22) by their estimates. Another way is to use the Residual squares criterion. The simplest way to estimate $m(x)$ is by a polynomial of order $k+3$ globally, which is called “pilot estimates”.

$$\hat{m}(x) = \hat{a}_0 + \cdots + \hat{a}_{k+3} x^{k+3} \quad (2.23)$$

It is quite easy to get the standardized residual sum of squares, $\hat{\sigma}^2$ and the derivative function $\hat{m}^{(k+1)}(x)$ (of quadratic order) from this parametric fit. Furthermore, we can take $w(x) = f(x)w_0(x)$ for some specific function $w_0$. With help of these estimate statistics, we can find the pilot estimate of $h_{opt}$.

$$h_{opt} = C_k(K) \left[ \frac{\hat{\sigma}^2 \int w_0(x)/f(x)dx}{n \int \{\hat{m}^{(k+1)}(x)\}^2 w_0(x)f(x)dx} \right]^{1/(2k+3)} \quad (2.24)$$

The denominator also can be estimated by

$$\sum_{i=1}^{n} \left\{\hat{m}^{(k+1)}(x_i)\right\}^2 w_0(x_i y)$$

Finally, we obtain the “rule of thumb bandwidth selector”

$$\hat{h}_{ROT} = C_k(K) \left[ \frac{\hat{\sigma}^2 \int w_0(x)/f(x)dx}{\sum_{i=1}^{n} \left\{\hat{m}^{(k+1)}(x_i)\right\}^2 w_0(x_i y)} \right]^{1/(2k+3)} \quad (2.25)$$

### 2.1.6 Some remarks on application

Local polynomial fitting is a non-parametric univariate regression method. Making a prediction on each data point is not quite economical, especially when data points are many and quite close to each other. One general accepted way is to make prediction on some evenly distributed grid.

The most useful function in R is `loess` by B. D. Ripley. Although it doesn’t
haven't many tweaking parameters introduced previously, it is include in the base of R, and it runs very quick comparing to other program packages, locfit, etc. Furthermore, it has support to Hastie’s “GAM” package. locfit also claims it supports the generalized additive modeling package GAM in S, but it does not work in R.

Here are some illustration of the default settings. \( \text{span} = 0.75 \) means there’s 75% of the data is evaluated during each local fit. “\( \text{family} = \text{gaussian} \)” means it uses least squares method to fit the data. In fact, it uses an adjusted tricubic kernel to weigh data points in the neighbourhood.

\[
f(u) = \frac{35}{32} \left(1 - \left(\frac{u}{d}\right)^2\right)^3
\]

where \( d \) is the maximum distance of the data points.

## 2.2 Generalized Additive Models

### 2.2.1 Framework

For multi-variable supervised learning. Suppose we have \( p \) predictors, \( X_1, X_2, \ldots, X_p \), and a response variable \( Y \). An additive model is defined as following:

\[
Y = \alpha + \sum_{j=1}^{p} f_j(X_j) + \varepsilon \tag{2.26}
\]

where \( \mathbb{E}(\varepsilon) = 0 \), \( \text{var}(\varepsilon) = \sigma^2 \), and \( f_j \) is a univariate function of \( X_j \) to be estimated. We assume \( \mathbb{E}\{f_j(X_j)\} = 0 \) so that \( \alpha = \mathbb{E}(Y) \).

The above additive model suggests that

\[
\mathbb{E}(Y | X_1, X_2, \cdots, X_p) = \mu(X) = \alpha + f_1(X_1) + f_2(X_2) + \cdots + f_p(X_p) \tag{2.27}
\]

In a generalized additive model, we instead assume
2.2. GENERALIZED ADDITIVE MODELS

\[ g[\mu(X)] = \alpha + f_1(X_1) + \cdots + f_p(X_p) \]  \hspace{1cm} (2.28)

and we call \( g(\cdot) \) a link function. For example, \( g(\mu) = \text{logit}(\mu) \) is the logit link used for binomial response data. \( g(\mu) = \log(\mu) \) is the log link used for Poisson response data. Note that GAM cannot account for any interaction between variables.

2.2.2 Local Polynomial as a smoother

A smoother can be regarded as a regression tool for capturing the trend of a response measurement \( Y \) as a function of the predictor measurements \( X_1, \ldots, X_p \). The objectives of smoothers are stabilising the prediction and reducing the variance as the name “smooth” suggests. Smoothers are non-parametric by its nature.

Usually, a univariate smoother is used for each predictor variable in an additive model. The sum of these smoothers are denoted as \( s = S(Y | X) \). Based on the characteristic of \( S(Y | X) \) and the data, a smoothing matrix (smoothing operator) can be constructed.

2.2.3 Fitting the model

A general iterative algorithm called “backfitting” is applied to fit an additive model. Previously, we have shown that local polynomial fitting is equivalent to some kernel function smoothing. Initially, set \( \hat{\alpha} = n^{-1} \sum_{i=1}^{n} Y_i \); \( \hat{f}_k = f_0^k \), \( k = 1, \ldots, p \).

For \( k = 1, \ldots, p, 1, \ldots, p, 1 \ldots \)

\[
\hat{f}_k = S_k \left\{ Y - \hat{\alpha} - \sum_{j \neq k} \hat{f}_j(X_j) | X_k \right\} \quad \text{where } S_k \text{ is the local polynomial fit,}
\]

until \( \{f_k\} \) converges.
2.2.4 Estimating the degrees of freedom

The degrees of freedom refer to the independent pieces of information involved in the estimation. It plays an important role as the penalty term of a model in information criteria does for model selection. In non-parametric statistics, degrees of freedom is usually represented by the effective degrees of freedom, some real number instead of an integer.

One typical way to calculate the effective degrees of freedom is to calculate the trace of the hat matrix $S$, or smoothing matrix. Let $y$ be the vector of data points we want to smooth. $S$ is called a smoothing matrix, which is a linear operator mapping $y$ to $\hat{y}$:

$$\hat{y} = Sy.$$

The effective degrees of freedom is defined to be

$$df = \text{trace}(S).$$

In local polynomial fitting, for each $y_j \in y$, we have $\hat{y}_i = \beta_{j,0}$, where

$$\beta_{j,0} = e_1^T \hat{\beta}_j = e_1^T \left(X^T W_j X\right)^{-1} X^T W_j y.$$

$e_1$ is a length $n$ vector with the first entity being 1 and rest of them being zero. Here, $\beta_{j,0}$ is the first element of $\beta_j$, which is the intercept of the local polynomial regression around $y_j$.

In fact, we can write

$$\hat{y} = (\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n)^T$$

$$= \sum_{j=1}^n \left\{ E_{j,1} \left(X^T W_j X\right)^{-1} X^T W_j y \right\}$$

$$= \sum_{j=1}^n \left\{ E_{j,1} \left(X^T W_j X\right)^{-1} X^T W_j \right\} y.$$
where $E_{i,j}$ is an $n \times n$ matrix with the $(i,j)$ entry being 1 and rest being zero.

After all we obtain the equivalent smoothing matrix $S_{lp}$ for local polynomial fitting:

$$S_{lp} = \left\{ \sum_{j=1}^{n} E_{j,1} \left( X^T W_j X \right)^{-1} X^T W_j \right\}$$

(2.29)

Accordingly, the effective degrees of freedom $df$ for local polynomial fitting with respect to $x$ is

$$df = \sum_{j=1}^{n} \left\{ e_{j,1} \left( X^T W_j X \right)^{-1} X^T W_j \right\}$$

(2.30)

As we can see, the degree of freedom is only dependent on $x$ and the parameters of local polynomial fitting. In GAM, backfitting algorithm is used where each explanatory variable requires one local polynomial fitting. For example, if $p$ variables are involved in the GAM, the degrees of freedom of GAM is the sum of the degrees of freedom of each variable, i.e.

$$df_{GAM} = \sum_{i=1}^{p} df_i.$$  

(2.31)

where $df_i$ is the degrees of freedom of variable $x_i$ using local polynomial fitting.
Chapter 3

Markov Chain Monte Carlo

Markov chain Monte Carlo (MCMC) is a family of simulation methods, which generate samples of Markov Chain processes. Gibbs Sampler, one of the MCMC method, is an algorithm to generate a sequence of samples from some joint distribution of a random vector. In this section, we set up a framework to use Gibbs sampler to solve certain optimization problems, and further adopted simulated annealing technique to accelerate the process. This chapter is more a preparation for MCMC model selection mainly discussed in next chapter.

3.1 Some preliminaries on Markov chain

In this section, we introduced the definition and some properties of Markov chain process. A brief proof of the marginal distribution

**Definition.** A discrete-time random process $\xi = \{\xi_0, \xi_1, \cdots\}$ taking values in some countable set $S$ is a Markov chain if the current status only depends on the previous one, i.e.

$$
P (\xi_n = s | \xi_0 = x_0, \cdots, \xi_{n-1} = x_{n-1}) = P (\xi_n = s | \xi_{n-1} = x_{n-1}) \quad (3.1)
$$

for all $n \geq 1$ and all $s, x_1, \cdots, x_{n-1} \in S$
Definition. State $i$ is called persistent (or recurrent) if
\[ P(\xi_n = i \text{ for some } n \geq 1 | \xi_0 = i) = 1 \]  
(3.2)
If this probability is strictly less than 1, the state is called transient.

Denote the probability that from state $i$, the first visit to state $j$, for precise $n$ steps.
\[ f_{i,j}(n) = P(\xi_1 \neq j, \ldots, \xi_{n-1} \neq j, \xi_n = j | \xi_0 = i) \]
Further define
\[ f_{i,j} = \sum_{n=1}^{\infty} f_{i,j}(n) \]
to be the probability that the chain hits $j$ all the time, starting from $i$.

Definition. The mean recurrence time $\mu_i$ of a state $i$ is defined as
\[ \mu_i = \mathbb{E}(T_i | \xi_0 = i) = \begin{cases} \sum_n nf_{ii}(n) & \text{if } i \text{ is persistent}, \\ \infty & \text{if } i \text{ is transient}. \end{cases} \]  
(3.3)
where $T_j = \min\{n \geq 1 : \xi_n = j\}$.

Definition. A matrix $P$ is called a transition matrix of a chain $\xi$, if $P = (p_{i,j})$ where
\[ p_{i,j} = P(\xi_{n+1} = j | \xi_n = i) \quad i, j \in \{1, 2, \ldots, |S|\}. \]  
(3.4)

Definition. The vector $\pi = \{\pi_j : j \in S\}$ is called a stationary distribution of some chain $\xi$ if

\[ (i) \quad \pi_j \geq 0 \text{ for all } j, \text{ and } \sum_j \pi_j = 1, \]

\[ (ii) \quad \pi = \pi P, \text{ i.e. } \pi_j = \sum_j \pi_j p_{i,j} \text{ for all } j. \]
3.1. SOME PRELIMINARIES ON MARKOV CHAIN

**Definition.** For a persistent state $i$,

\[
\text{state } i \text{ is called } \begin{cases} 
\text{null} & \text{if } \mu_i = \infty, \\
\text{non-null} & \text{if } \mu_i < \infty.
\end{cases}
\]

Some notations: Denote $\rho_i(k)$ be the mean number of visits of the chain to the status $i$ between two successive visits to state $k$; that is, $\rho_i(k) = E(N_i|\xi_0 = k)$ where

\[
N_i = \sum_{n=1}^{\infty} I_{\{\xi_n = i\} \cap \{T_k \geq n\}}
\]

and $T_k$ is the time of the first return to state $k$, as before. Note that $N_k = 1$ so that $\rho_k(k) = 1$, and that

\[
\rho_i(k) = \sum_{n=1}^{\infty} P(\xi_n = i, T_k \geq n|\xi_0 = k).
\]

the probability that the chain reaches $i$ in $n$ steps but with no intermediate return to its starting point $k$.

\[
l_{k,i}^n = P(\xi_n = i, T_k \geq n|\xi_0 = k),
\]

**Lemma 3.1.** For any state $k$ of an irreducible persistent chain, the vector $\rho(k)$ satisfies $\rho_i(k) < \infty$ for all $i$, and furthermore $\rho(k) = \rho(k)P$.

**Proof:**

\[
f_{k,k}(m+n) \geq l_{k,i}(m)f_{i,k}(n)
\]

\[
\rho_i(k) = \sum_{m=1}^{\infty} l_{k,i}(m) \leq \frac{1}{f_{i,k}(n)} \sum_{m=1}^{\infty} f_{k,k}(m+n) \leq \frac{1}{f_{i,k}(n)} < \infty
\].
for the second statement

\[ l_{k,i} = p_{k,i} \]

\[ l_{k,i}^n = \sum_{j:j \neq k} P(\xi_n = i, \xi_{n-1} = j, T_k \geq n | \xi_0 = k) = \sum_{j:j \neq k} l_{k,i}^{n-1} p_{j,i} \quad \text{for } n \geq 2, \]

Conditioning on the value of \( \xi_{n-1} \)

\[ \rho_k^i = p_{k,i} + \sum_{j:j \neq k} \left( \sum_{n \geq 2} l_{k,j}^{n-1} \right) p_{j,i} = \rho_k^k p_{k,i} + \sum_{j:j \neq k} \rho_j^k p_{j,i} \]

\( \rho_k^k = 1 \), hence lemma is proved.

**Theorem 3.2.** For an irreducible chain, if all the states are non-null persistent, then it has a stationary distribution \( \pi \).

**Proof.** Define \( \pi_i = \rho_i^k / \mu_k \), we get \( \pi = \pi P \).

**Theorem 3.3.** For an irreducible ergodic Markov chain limit \( p_i \) to \( p_j \) exists Furthermore, if all its states are non-null persistent, we have the follow result

\[ \pi_j = \lim_{n \to \infty} P_{i,j}^n, \quad j \geq 0. \quad (3.5) \]

**Proof.** Use coupling to prove.

Let \( Z = (\xi, Y) \) be a coupled chain, where \( \xi, Y \) are independent Markov chain with same states space \( S \) and transition matrix \( P \). As we can see that \( Z_n \) is in states space \( S \times S \), and the transition matrix has \( p_{ij,kl} = p_{i,k} P_{j,l} \). Since \( \xi \) is irreducible and aperiodic, there exists \( N \) such that \( p_{i,k}^{n} P_{j,l}^{n} > 0 \) for all \( n \geq N \), which shows that \( Z \) is also irreducible(where the aperiodic of \( \xi \) is needed).

\( \xi \) is non-null persistent. Then according to the previous theorem, there exists a unique stationary distribution \( \pi \) of \( \xi \). And \( Z \) also has a stationary distribution \( \nu = (\nu_{i,j}, j \in S) \) where \( \nu_{i,j} = \pi_i \pi_j \). Hence \( Z \) is also non-null persistent.(here we need the full theorem iff)
Without lose of generality, we assume $Z_0 = (i, j)$ for some $i, j \in S$. For any state $s \in S$ let

$$T_s = \min \{ n \geq 1 : Z_n = (s, s) \}.$$  

since $f_{ii} = 1$ for irreducible and persistent MC, $\mathbf{P}(T < \infty) = 1$.

Conditional on $\{T \leq n\}$, $\xi_n$ and $Y_n$ have the same distribution.

$$p^n_{i,k} = \mathbf{P}(\xi_n = k) \quad (3.6)$$

$$= \mathbf{P}(\xi_n = k, T \leq n) + \mathbf{P}(\xi_n = k, T > n) \quad (3.7)$$

$$= \mathbf{P}(Y_n = k, T \leq n) + \mathbf{P}(\xi_n = k, T > n) \quad (3.8)$$

$$\leq \mathbf{P}(Y_n = k) + \mathbf{P}(T > n) \quad (3.9)$$

$$= p^n_{j,k} + \mathbf{P}(T > n). \quad (3.10)$$

$$|p^n_{i,k} - p^n_{j,k}| \leq \mathbf{P}(T > n) \to 0 \quad \text{as } n \to \infty \quad (3.11)$$

Show the existence of the limit.

$$\pi_k - p^n_{j,k} = \sum_i \pi_i (p^n_{i,k} - p^n_{j,k}) \to 0 \quad \text{as } n \to \infty \quad (3.12)$$

It goes to 0 when $F$ approaches to $S$. \hfill \Box

**Theorem 3.4.** Let $P$ be the transition matrix of an irreducible chain $\xi$, and suppose that there exists a distribution $\pi$ such that $\pi_i p_{i,j} = \pi_j p_{j,i}$ for all $i, j \in S$. Then $\pi$ is a stationary distribution of the chain. Furthermore $\xi$ is reversible in equilibrium.

**Proof.**

$$\sum_i \pi_i p_{i,j} = \sum_i \pi_j p_{j,i} = \pi_j \sum_i p_{j,i} = \pi_j$$
which gives us $\pi = \pi P$.

Theorem 3.2, 3.3, 3.4 will be used to check that the Gibbs sampler in MCMC model selection forms an irreducible ergodic Markov chain, and sample chain converges to the stationary distribution we designed in Chapter 4.

### 3.2 Gibbs Sampler

Gibbs sampler is a random vector sampling algorithm which does not require the complete information of the target multivariate joint distribution. It utilizes conditional distributions of one dimension depending on other dimensions to generate a value of that dimension. After values of all dimensions have been updated, a sample is generated. By repeat doing this, a sequence of samples can be obtained. This sample is a realization of some Markov chain process the stationary distribution of which is exactly the target multivariate joint distribution we are after. It was first developed by Geman and Geman [1984]. The name, Gibbs, is came from the gibbs distribution they used in image reconstruction. Gibbs sampler will be computationally feasible if the conditional distributions are ready to be simulated. Comparing another MCMC method, Metropolis-Hastings algorithms [Metropolis et al., 1953], Gibbs sampler is capable of dealing with random vectors instead of random variable. It is widely used in Bayesian analysis and physics, etc.

#### 3.2.1 Gibbs Sampler Algorithm

Suppose we want to generate a sample chain of a random vector $U = (U_1, U_2, \cdots, U_p)$ with joint probability density distribution function $f(u)$, where $u = (u_1, \ldots, u_p)$.

Denote $U_{-k} = (U_1, \ldots, U_{k-1}, U_{k+1}, \cdots, U_p)$. Then we denote $f_{(k)}$ as conditional marginal pdf of $U_k$.
3.2. GIBBS SAMPLER

\[ f_{(k)}(u_k | u_1, \ldots, u_{k-1}, u_{k+1}, \ldots, u_p) = P\left\{ U_k = u_k | U_{-k} = (u_1, \ldots, u_{k-1}, u_{k+1}, \ldots, u_p) \right\} \]

Gibbs Sampler Algorithm

Step i: Choose an arbitrary initial vector \( u_0 = (u_1^{(0)}, u_2^{(0)}, \ldots, u_p^{(0)}) \)

Step ii: 
1. Generate \( u_1^{(1)} \) from conditional pdf \( f_{(1)}(u_1 | u_2^{(0)}, \ldots, u_p^{(0)}) \)
2. Generate \( u_2^{(1)} \) from conditional pdf \( f_{(2)}(u_2 | u_1^{(1)}, u_3^{(0)}, \ldots, u_p^{(0)}) \)
   
   : 
   
   : 

p. Generate \( u_p^{(1)} \) from conditional pdf \( f_{(p)}(u_p | u_1^{(1)}, \ldots, u_{p-1}^{(1)}) \)

p+1. Obtain sample \( u^{(1)} = (u_1^{(1)}, \ldots, u_p^{(1)}) \)

Step iii: For \( j \in \{2, 3, \ldots, N + m\} \), we repeat Step ii to generate \( u^{(j)} \).

Step iv: We set the first \( m \) vectors as burnin period, and take rest \( N \) vectors \( \{u^{(m+1)}, \ldots, u^{(m+N)}\} \) as our sample chain.

Some remarks

- The feasibility of this algorithm is based on the set of conditional distributions.

- Gibbs sampler is in fact a well-defined Markov chain process. Given enough time, after it reaches its equilibrium, the sample chain will have a marginal distribution converges to its stationary distribution. The sample chain can be regarded as sample of the stationary distribution but with dependence, i.e. not i.i.d. However, in model selection case, the dependence is not a concern. In this thesis, we will only provide a proof of this statement in the model selection case, which the Markov chain is defined on a finite discrete state space.
There's no universal rule to determine how long the burnin period should be. In other words, The length of burnin period before the Markov Chain process reaches it equilibrium varies case by case. We will introduce several convergence diagnosis tools to handle this problem.

### 3.3 Diagnose the convergence of Gibbs sampler

The term convergence is quite ambiguous here. Actually, Gibbs Sampler is a well defined Markov chain, a stochastic process, itself. This process doesn’t alter during the whole simulation. “Convergence”, literally, is referred to the convergence of the marginal distribution of each state to the stationary distribution of this Markov chain. In another word, Test of convergence means to test whether the process reaches its equilibrium.

Although the equilibrium of Markov Chain Monte Carlo (MCMC) methods can be reached eventually, it is a tough job to determine the convergence after some certain burn-in period, not even considering the case when Markov chain has a great number of states. Appropriate diagnostics of the “convergence” will not only enhance the accuracy of the sampling but also save considerable computation power.

There are many of methods can check the convergence. However it is always possible to raise a counter-example for each methods even with a strong conclusion. It is widely believed that a mixture of tests can provide more reliable evidence of convergence. Four tests will be introduced here considering effectiveness and ease of use. They are [Gelman and Rubin, 1992], [Geweke, 1992] and [Qian and Zhao, 2007].

**i-Chart** is a graphical test introduced by [Qian and Zhao, 2007]. They set up a boundary with respect to the samples. If the number of samples go beyond the boundary is less then certain proportion of the sample size, then the process is regarded as reached its equilibrium. Let $X$ be a random
variable. Let $X$ be a random variable, with density $f$. Figure 5.2 shows an example of this diagnosis method.

$$P \left\{ X - \min X \geq b \sqrt{\text{Var}(X) + (E(X) - \min X)^2} \right\} \leq b^{-2} \quad (3.13)$$

**Proof.** Let $\sigma^* = \sqrt{\text{Var}(X) + (E(X) - \min X)^2}$,

$$E \left\{ \left( \frac{X - \min X}{b\sigma^*} \right)^2 \right\} = \int_0^\infty \left( \frac{s - \min X}{b\sigma^*} \right)^2 f(s)ds$$

$$= \frac{1}{b^2\sigma^*} \int_0^\infty (s - \min X)^2 f(s)ds$$

$$= \frac{1}{b^2\sigma^*} \int_0^\infty (s - E(X) + E(X) - \min X)^2 f(s)ds$$

$$= \frac{1}{b^2\sigma^*} \left\{ \text{Var}(X) + (E(X) - \min X)^2 \right\}$$

$$+ \int_0^\infty 2(s - E(f))(E(X) - \min X)f(s)ds = \frac{1}{b^2} \quad \square$$

**Gelman and Rubin** [1992] propose a test based on normal theory approximations involving two steps. Firstly, construct an over-dispersed estimate of the stationary distribution to generate some starting points of independent chains. Second step is to do an ANOVA-like on these independent chains to see whether they come from the same distribution. The convergence is revealed by the parameter $\sqrt{\hat{R}}$, namely

$$\sqrt{\hat{R}} = \sqrt{\left( \frac{n - 1}{n} + \frac{(m + 1)B}{mnW} \right) \frac{df}{df - 2}} \quad (3.14)$$

Where $m$ stands for number of chains simulated, $n$ is the length of each chain, $B$ is the variance between the chains, $W$ describes the within-chain
variances, $df$ is the degrees of freedom of the approximating Student $t$ distribution.

If $\sqrt{R}$ is close to 1, then it is believed that the convergence is reached.

Geweke [1992]'s idea is to test the equality of two subsequences' location measures. After certain burn-in period, according to the assumption, the first $m$ outcomes and last $m$ outcomes should share the same location measures.

$$Z_n = \frac{(\bar{\theta}_A - \bar{\theta}_B)}{\sqrt{(1/n_A)\hat{S}_A^0(0) + (1/n_B)\hat{S}_B^0(0)}} \to N(0, 1)$$

where $n_B$ is the length of the second subsequence, $\bar{\theta}_A = (1/n_A)\sum_{t=1}^{n_A} \theta^t$, $\bar{\theta}_B = (1/n_B)\sum_{t=1}^{n_B} \theta^t$. $\hat{S}_A^0(0)$ and $\hat{S}_B^0(0)$ are estimators of the variance of $\theta$ based on $\{\theta^t: t = 1, \ldots, n_A\}$ and $\{\theta^t: t = n^*, \ldots, n\}$.

### 3.4 MCMC for optimization problems

**Definition.** A optimization problem can be formalized as a pair $(\mathcal{S}, f)$, where $\mathcal{S}$ is the solution space denoting finite number of possible solutions, and $f$ is a cost function maps $\mathcal{S}$ to $\mathbb{R}$.

In case of minimization problems, a solution $s_{opt} \in \mathcal{S}$ which satisfies

$$f(s_{opt}) \leq f(s_i), \text{ for all } s_i \in \mathcal{S}$$

is what we are looking for.

In order to apply MCMC, we construct a probability measure on the solution space with respect to lost function. One widely used option is Gibbs distribution defined as follows:

**Definition.** Gibbs distribution for optimization problem $(\mathcal{S}, f)$.

$$P(s_i) = q_i = \frac{\exp(-f(s_i))}{\sum_j \exp(-f(s_j))}, \quad \forall s_i \in \mathcal{S} \quad (3.15)$$
As we can see, the optimal state will have the greatest probability. But due to the complexity of the design, Gibbs distribution is very hard to compute. This is where MCMC come in place. We can use MCMC to simulate a sample of this distribution. And the optimal will appear most frequently in the sample.

3.5 Simulated annealing

Simulated annealing (SA) is a probabilistic method for finding global minimum of some cost function introduced by [Kirkpatrick et al. 1983]. It searches local minima, and finally stays at the global minimum given enough time. This sampling method was originally extended from Metropolis Algorithm [Metropolis et al. 1953] by implanting a temperature function $T$. $T$ is used to control the difficulty for the stochastic sampler to escape from a local minimum and reach the global optimal for a non-optimal state. In this thesis, we implemented it in Gibbs sampler.

3.5.1 Framework

**Definition.** Temperature sensitive Gibbs distribution with respect to Temperature $T$:

$$P(s_i, T) = q_i(T) = \frac{\exp\left(-\frac{f(s_i)}{T}\right)}{\sum_j \exp\left(-\frac{f(s_j)}{T}\right)} \quad \forall s_i \in \mathcal{S} \quad (3.16)$$

**Definition.** A cooling schedule is a function $T(t)$ describing how temperature varies with respect to time $t$.

Typically, cooling schedules are monotonically decreasing functions of $t$. When temperature $T$ is high, probabilities are evenly distributed among all states. On the other hand, when $T$ is close to 0, the set of optimal states occupies all the probabilities.

**Proposition 3.5.** Asymptotic behavior, when temperature $T$ goes down to zero, the best models will appear with probability one. Denote $\mathcal{S}_{opt}$ as the set of globally best models minimizing $f(\alpha)$, then
\[ \lim_{T \to 0} q_i(T) = \frac{1}{\# \{ \mathcal{S}_{opt} \} I_{\{i \in \mathcal{S}_{opt} \}}} \]  

(3.17)

Proof.

\[ \lim_{T \to 0} q_i(T) = \lim_{T \to 0} \frac{\exp(-f(s_i)/T)}{\sum_j \exp(-f(s_j)/T)} = \lim_{T \to 0} \frac{\exp(f_{opt} - f(s_i)/T)}{\sum_j \exp(f_{opt} - f(s_j)/T)} \]

\[ = \lim_{T \to 0} \frac{1}{\sum_j \exp(f_{opt} - f(s_j)/T)} I_{\{i \in \mathcal{S}_{opt} \}} + \lim_{T \to 0} \frac{\exp(f_{opt} - f(s_i)/T)}{\sum_j \exp(f_{opt} - f(s_j)/T)} I_{\{i \notin \mathcal{S}_{opt} \}} \]

\[ = \frac{1}{\# \{ \mathcal{S}_{opt} \} I_{\{i \in \mathcal{S}_{opt} \}}} + 0 \]

where \#\{ \mathcal{S}_{opt} \} is the number of elements in \( \mathcal{S} \), \( I \) is a indicate function.

**Proposition 3.6.** Asymptotic behavior, when temperature \( T \) goes up infinity, each model will be equally likely been chosen.

\[ \lim_{T \to \infty} q_i(c) = \frac{1}{\# \{ \mathcal{S} \} } \text{ for any } i \]  

(3.18)

**Remarks** Comparing to MCMC for optimization problems in Section 3.4, simulated annealing use temperature \( T \) to control the probability measure on the state space. When \( T \) is low, the state with lower cost function values gains greater probability. This makes global minimum more obvious in the sample chain. The gradually declining cooling schedules insures that the process will jump out of local minima and finally stays in the global minimum. In other words, simulated annealing accelerates the MCMC optimization process by gradually squeezing out useless information (the state that is not global minimum).

Furthermore, looking from an information theory point of view, the entropy (4.2) of this annealed probability measure reaches it's maximum entropy
when $T$ goes to infinity, and reaches minimum when $T$ goes to zero. This is a very interesting result consistent with the cooling down concept. In a cooled system, the structure (information) are more stable, while particle vibration (noise) reduced.

**Implementation in Gibbs sampler** A simulated annealed Gibbs sampler can be easily derived from the temperature sensitive Gibbs distribution. Let $T(t)$ be a cooling schedule function of time $t$. Suppose at time $t$ we have a $p$-dimension random vector $U^{T(t)} = \left[U^{T(t)}_1, \ldots, U^{T(t)}_p \right]$ with joint probability density distribution function $f^{T(t)}(u)$, where $u = \left(u_1, \ldots, u_p \right)$. Then we denote $f^{(k)}_{(k)}$ as the $k$th conditional marginal pdf of $U^{T(t)}$ under temperature $T$

$$f^{T(t)}(u_k \mid u_1, \ldots, u_{k-1}, u_{k+1}, \ldots, u_p) = P\left(U^{T(t)}_k = u_k \mid U^{T(t)}_{-k} = \left(u_1, \ldots, u_{k-1}, u_{k+1}, \ldots, u_p \right) \right).$$

**Gibbs Sampler Algorithm**

**Step i:** Choose an arbitrary initial vector $u_0 = \left(u^{(0)}_1, u^{(0)}_2, \ldots, u^{(0)}_p \right)$, set time $t = 0$

**Step ii:**
1. Generate $u^{(1)}_1$ from conditional pdf $f^{T(t+1)}_{(1)}(u^{(0)}_1, u^{(0)}_2, \ldots, u^{(0)}_p)$
2. Generate $u^{(1)}_2$ from conditional pdf $f^{T(t+1)}_{(2)}(u^{(1)}_1, u^{(0)}_3, \ldots, u^{(0)}_p)$
   \[ \vdots \]
\[ p. \] Generate $u^{(1)}_p$ from conditional pdf $f^{T(t+1)}_{(p)}(u^{(1)}_1, \ldots, u^{(1)}_{p-1})$
\[ p+1. \] Obtain sample $u^{(1)} = \left(u^{(1)}_1, \ldots, u^{(1)}_p \right)$

**Step iii:** For $j \in \{2, 3, \ldots, N + m\}$, we repeat Step ii and set $t = j$ to generate $u^{(j)}$.

**Step iv:** We set the first $m$ vectors as burnin period, and take rest $N$ vectors $\left\{u^{(m+1)}, \ldots, u^{(m+N)}\right\}$ as our sample chain.

The new simulated annealing Gibbs sampling is no longer a time-homogenous Markov chain. Actually, it is time-inhomogenous, since the temperature $T$
keeps changing. In this case, for each time \( t \), there is a different stationary distribution with respect to temperature \( T(t) \). The convergence here is referred to convergent to the asymptotic stationary distribution (3.17).

### 3.5.2 Convergence of SA

We already knew that Gibbs sampler is a time-homogenous Markov chain, but simulated annealing Gibbs sampler is a time-inhomogenous Markov chain. Geman and Geman [1984] gave the first proof of convergence of simulated annealing under a specific type of cooling schedules:

\[
T(t) = \frac{d}{\ln(t)},
\]

(3.19)

where \( d \) is some difficult constant measures the difficult of the process to jump out of a local minimum. Mitra et al. [1986] and Hajek, 1988 developed the theory and weakened the conditions using the same cool schedules which is stated below.

**Theorem 3.7.** Convergence of simulated annealing [Hajek, 1988]. We say that state \( s_i \) communicates with \( S^* \) at height \( h \) is there exists a path in \( S \) that starts at \( s_i \) and ends at some element of \( S^* \), and the greatest value of \( f \) along the path is \( f(s_i) + h \). Denote \( d^* \) be the lower bound of \( d \) such that \( \forall s_i \in S \) communicates with \( S^* \) at height \( d^* \). Then the SA algorithm converges if and only if

\[
\lim_{t \to \infty} T(t) = 0 \quad \& \quad \sum_{i=1}^{\infty} e^{-d^*/T(t)} = \infty
\]

By Theorem [3.7], Using (3.19) as cooling schedules, SA converges if and only if \( d \geq d^* \). This theorem provides some theoretical assurance that simulated annealing can select the global minimum. However, these cooling schedules decline very slowly. In practice, we may use other cooling schedules to gain more efficiency.
3.5. Other annealing methods

**Simulated quenching** is to use more aggressive cooling schedules, which converges to 0 much faster then (3.19). For example, exponential cooling schedules:

\[ T(t) = \gamma^t T_0 \quad \text{where } \gamma \in (0, 1) \quad (3.20) \]

In these cases, convergence is no longer guaranteed, i.e. the simulated annealing stays in some local minimum instead of the global one.

**Fast annealing** [Szu and Hartley [1987]] introduced a new annealing method with new probability measure, Cauchy distribution (3.21), and cooling schedules. Actually the new measure is not defended on the problems space but the difference of cost function \( \Delta x = f(s) - f(s') \), where \( s, s' \in \mathcal{S} \).

\[ g(\Delta x) = \frac{T}{(\Delta x^2 + T^2)^{(D+1)/2}} \quad (3.21) \]

\[ T(t) = \frac{T_0}{t} \quad (3.22) \]

The \( D \) is some shape parameter, and \( T_0 \) is just the initial temperature. As for Cauchy distribution, the density increases much faster when cost function decreases, which increases the chances that the process jump from local minima to global minimum. Tested by [Szu and Hartley [1987]], fast annealing is exponential faster then simulated annealing methods using Gibbs distribution in many cases.

**Re-annealing** Re-annealing is simply do annealing several times with same or different parameters, such as difficulty parameter, initial temperature, etc. In the next Chapter, we used the concept of re-annealing to construct some time-homogenous process which some inference can be done.

**Adaptive annealing** [Ingber [1996]] presents a very fast annealing solution combining quenching and re-annealing techniques. It provides C codes with
many option flags can be adapted in varieties of problems.
Chapter 4

Model Selection Procedures

In this chapter, a model selection framework was first set up. This framework emphasizes on selecting the best subsets of candidate predicting variables rather than selecting a more appropriate regression method. For each subset of candidate predictors, a model is fitted by an identical model method, GAM with local polynomial fitting introduced previously. The aim of the model selection procedure would be finding the model best explains the underling information in some dataset from a model space formed by all subsets of candidate predicting variables.

To assess the goodness of models, we used information criteria, estimations of Kullback Leibler divergence (4.3), i.e. AIC, BIC. This transforms model selection problem into an optimization problem. When \( p \) is small, an exhaustive search of all possible models can be done. However, when \( p \) goes larger, exhaustive search just becomes infeasible. MCMC model selection overcomes this difficulty. Gibbs sampler, as one of the MCMC method, is constructed on the model space. It generates a Markov chain sample, in which the models with less information model selection criterion values have greater chance to appear. Analyzing the sample chain would enable us to locate a few best models and important predicting variables. Furthermore, we implemented simulated annealing techniques into the Gibbs sampler, which would accelerate the convergence of MCMC process, and also
provide more diagnostic information of the important predicting variables.

### 4.1 Model selection Framework

Let $Y$ is the response random variable in interest. We assume $Y$ can be predicted through some candidate explanatory random vector $X$ following unknown distribution $G(Y|X)$. A statistical model is another distribution $F(Y_n|X_n)$ formed by some observed dataset $\{Y_n, X_n\}$ with $n$ data points. It is an estimation of the true distribution $G(Y|X)$. Model selection, generally speaking, is a methodology that selects the best estimation among some set of candidate models.

Usually, we have no idea what the predicting random vector $X$ is. However, there are set of candidate predicting random variables $\{X_i| i = 1, 2, \ldots, p\}$ available for chosen. Then it leaves us two directions to search models. One is to select a right or suitable regression method. Another one is to choose an appropriate subset of these candidate predicting variables. In this thesis, we use GAM with local polynomial fitting as our regression method, and focus on selecting the subset of candidate predicting variables which would give us a closed estimation of the underlying true model. The major object in this thesis is to develop a systematic procedure to select the subset of true or important explanatory variables.

**Definition.** A GAM model space $\mathcal{A}$ is a set of generalised additive models formed by different subsets of the candidate explanatory variables.

Say $Y$ maybe influenced by a set of number $p$ predicting variables $\{X_1, X_2, \ldots, X_p\}$. Since we always use the same regression method, GAM with local polynomial fitting, the models in $\mathcal{A}$ can be specified by the subset of predicting variables involved. Thus a model $\alpha \in \mathcal{A}$ can be simply denoted by a binary vector

$$\alpha = (v_1, v_2, \ldots, v_p),$$

where $v_j = I\{X_j \text{ is in the model}\}$ tells us whether $X_j$ is included in model $\alpha$. Therefore, since we use an identical modeling method, the model space $\mathcal{A}$ here
actually can be regarded as a $p$-dimension binary vector space with a number of $2^p$ states.

### 4.1.1 Information Criteria

“All models are wrong; some models are useful.” – George Box. In traditional statistics modeling, the aim of modelling is to construct a model approximating the underlying unknown true model as accurately as possible given some observed dataset. Nowadays, people tend to treat statistical models as tools of extraction of information. Instead of finding the closed model, finding the most useful model more or less has become our main interest. In information theory, entropy is used to measure the amount of ultimate data compression of certain piece of information. For instance, a dataset can be viewed as a piece of encrypted message, while a model is a tool decoding the message. With a good model, we are able to obtain more information the message carries; entropy here measure the ultimate possible informations can be carried be this message. We may assume there is a proper way to decode the message. Relative entropy is a measure of difference between the entropies of two distributions. We use it to evaluate the goodness of a model to see how much useful information can be extracted from the dataset comparing to the unknown true model. To archive this, we introduce two estimation of the relative entropy, AIC and BIC.

**Definition.** The entropy $H(Y)$ for a discrete random variable $Y$ with probability measure $p$ and range $\mathcal{Y}$, is defined as

$$H(Y) = - \sum_{y \in \mathcal{Y}} p(y) \ln p(y) = \mathbf{E} \ln p(Y) \quad (4.2)$$

**Definition.** Let $Y, \hat{Y}$ with distributions $q, p$ be two random variables. The
relative entropy, Kullback-Leibler (KL) Information, is defined as

\[ D(p||q) = \sum_{y \in \mathcal{Y}} p(y) \ln \frac{p(y)}{q(y)} \] (4.3)

Assuming \( y = \{y_1, \ldots, y_n\} \) is an i.i.d. sample from \( Y \) following distribution \( p(\cdot) \) which is unknown but fixed. We estimate \( Y \) using some \( \hat{Y} \) have distribution \( q(\cdot) \). Using KL information as a selection criteria, the better model \( \hat{Y} \) should have a smaller information. Since \( \{y_i\} \) are i.i.d. we can estimate KL information by

\[ D(p||q) \approx \frac{1}{n} \left\{ \sum_i \ln p(y_i) - \sum_i \ln q(y_i) \right\} \]

Actually the only model dependent part is \( \sum_i \ln q(y_i) \), the log likelihood with respect to \( q \). Given data \( y \) we assume \( f(y|\theta) \) to be the likelihood function given some \( k \times 1 \) parameter vector \( \theta \), and \( \ell(\theta) \) be the likelihood. Since \( \theta \) is unknown, we use the maximum likelihood estimator \( \hat{\theta} \) instead. There are two widely used estimation of KL information, AIC and BIC. Akaike information criterion [Akaike, 1973]:

\[ \text{AIC} = -2 \cdot \text{maximum log likelihood} + 2k = -2 f(y|\hat{\theta}) + 2k, \] (4.4)

and Bayesian information criterion [Schwarz, 1978]:

\[ \text{BIC} = -2 \cdot \text{maximum log likelihood} + k \ln n = -2 f(y|\hat{\theta}) + k \ln n \] (4.5)

where \( k \) is the number of parameters estimated in the model. It is also called degrees of freedom. Generally speaking, BIC tends penalize more heavily on complexity of the model, i.e. higher degrees of freedom. On the other hand, the model selected according to AIC may have higher prediction rate.

For model selection in this thesis, we either choose AIC or BIC as our model
selection criterion depends on the what kind of information we need, i.e. prediction, the set of important predicting variables. With the help of information criteria, model selection is in fact a optimization problem defined on some model space \( \mathcal{A} \). In order to rank models by their information criterion values, we index models according to their binary vector form (4.1). Recall, in model selection framework, there is a supervising variable \( Y \), and \( p \) corresponding variables \( \{X_i | i = 1, \ldots, p\} \). We denote a particular model \( \alpha \) in GAM model space \( \mathcal{A} \) as a \( p \)-dimensional binary vector. Furthermore, we index all \( 2^p \) models by transforming the binary vector form as a binary number to decimal number.

Suppose the dataset has \( n \) data points, we use \( Y_n, X_n \) to denote the data set. For model \( \alpha_i \), let \( f_i = f(\alpha_i | Y_n, X_n) \) denote its information criterion value, i.e. AIC or BIC, where \( f \) is the information criterion function. Since better model has a lower information criterion value, the model selection problem has become an optimization problem, i.e. find the model \( \alpha_{opt} = \min_i f(\alpha_i | Y_n, X_n) \).

### 4.1.2 Derivation of BIC

In Keynesian’s framework, we say \( \theta \) follows some prior distribution \( \pi(\theta) \). Then we have the likelihood of data \( y \) to be

\[
p(y) = \int f(y|\theta)\pi(\theta)d\theta, \tag{4.6}
\]

\[
= \int \exp\{\ln f(y|\theta)\}(\theta)d\theta \tag{4.7}
\]

\[
= \int \exp\{\ell(\theta)\}\pi(\theta)d\theta
\]

The Taylor expansion of the log-likelihood function \( \ell(\theta) \) around maximum
likelihood estimator $\hat{\theta}$ yields
\[
\ell(\theta) = \ell(\hat{\theta}) - \frac{n}{2} (\theta - \hat{\theta})^T J(\hat{\theta}) (\theta - \hat{\theta}) + \cdots, \tag{4.8}
\]
where
\[
J(\hat{\theta}) = -\frac{1}{n} \frac{\partial^2 \ell(\theta)}{\partial \theta \partial \theta^T} \bigg|_{\theta = \hat{\theta}}. \tag{4.9}
\]
Similarly, we expand the prior distribution $\pi(\theta)$, and simplified it using the convergence property of $\hat{\theta}$ to $\theta$.
\[
\pi(\theta) = \pi(\hat{\theta}) + (\theta - \hat{\theta})^T \frac{\partial \pi(\theta)}{\partial \theta} \bigg|_{\theta = \hat{\theta}} + \cdots = \pi(\hat{\theta}) + O_p(n^{-1/2}). \tag{4.10}
\]
Substituting (4.8) and (4.10) into (4.7) and apply Laplace approximation for integrals for large $n$, we have
\[
q(x_n) = \int \exp \left\{ \ell(\hat{\theta}) - \frac{n}{2} (\theta - \hat{\theta})^T J(\hat{\theta}) (\theta - \hat{\theta}) + \cdots \right\} \left\{ \pi(\hat{\theta}) + O_p(n^{-1/2}) \right\} d\theta \\
\approx \pi(\hat{\theta}) \int \exp \left\{ \ell(\hat{\theta}) - \frac{n}{2} (\theta - \hat{\theta})^T J(\hat{\theta}) (\theta - \hat{\theta}) \right\} d\theta \\
\approx \exp \{ \ell(\hat{\theta}) \} \pi(\hat{\theta}) \int \exp \left\{ -\frac{n}{2} (\theta - \hat{\theta})^T J(\hat{\theta}) (\theta - \hat{\theta}) \right\} d\theta. \tag{4.11}
\]
The integration can be regarded as expectation of normal distributed r.v., which can be simplified as
\[
\int \exp \left\{ -\frac{n}{2} (\theta - \hat{\theta})^T J(\hat{\theta}) (\theta - \hat{\theta}) \right\} d\theta \\
= \left( \frac{2\pi}{n} \right)^{p/2} |J(\hat{\theta})|^{-1/2} \int \left( \frac{n}{2\pi} \right)^{p/2} |J(\hat{\theta})|^{1/2} \exp \left\{ -\frac{n}{2} (\theta - \hat{\theta})^T J(\hat{\theta}) (\theta - \hat{\theta}) \right\} d\theta \\
= \left( \frac{2\pi}{n} \right)^{p/2} |J(\hat{\theta})|^{-1/2}
\]
Consequently, the likelihood $p(y)$ can be approximated as follows when $n$ becomes large.
\[ p(y) \approx \exp \left\{ \ell(\hat{\theta}) \right\} \left( \frac{2\pi}{n} \right)^{p/2} |J(\hat{\theta})|^{-1/2} \pi(\hat{\theta}) \]  

(4.12)

Taking nature logarithm of (4.12) and multiplying it by \(-2\), we obtain

\[ -2 \ln q(y) = -2 \ln \left\{ \int f(y|\theta)\pi(\theta) d\theta \right\} \]

\[ \approx -2\ell(\hat{\theta}) + k \ln n + \ln |J(\hat{\theta})| - k \ln(2\pi) - 2 \ln \pi(\hat{\theta}). \]  

(4.13)

Then, by ignoring terms with order less than \(O(1)\) with respect to the sample size \(n\), the model evaluation criterion BIC is obtained.

### 4.2 An exhaustive search

An exhaustive search essentially does regressions all model in the model space. It selects model by choosing the model with the lowest information criterion value. This strategy covers all subsets of predicting variables, and guarantees to find the best model in the current framework. However, it is feasible only when \(p\), the number of candidate predicting variables are small, since the size of the model spaces simply grows exponentially when \(p\) increases.

### 4.3 MCMC model selection

MCMC model selection is a procedure use Gibbs sampler, one of the MCMC method, to solve the optimization problem of model selection. It was first introduced by \cite{QianField2002} to solve model selection problems for logistic models. It generates a Markov chain sample of models in \(\mathcal{M}\). When the Markov chain reaches its equilibrium, we will have a stationary distribution with the property that the better model appears more often. We take advantage of this property to select model when there is a large number of explanatory variables. In this thesis, we further applied this method to
model selection of GAM with local polynomial fitting.

First we define a measure on $\mathcal{A}$, the Gibbs distribution

$$P(\alpha_i) = \frac{\exp(-f(\alpha_i|Y_n, X_n))}{\sum_{\alpha_j \in \mathcal{A}} \exp(-f(\alpha_j|Y_n, X_n))}$$

(4.14)

As for $f$, commonly, we can either choose AIC or BIC introduced previously. The probability assigned to each model is consistent with the goodness of the model. The lower the $f(\alpha_i)$ is, the greater chance it got selected. In other words, this stochastic process favors good models. What MCMC model selection does is to construct a Markov chain which has this exact Gibbs distribution as its stationary distribution.

Once we obtain a sample chain, we can estimate the best model from a sample in following ways:

- Estimate the best model by the one having the smallest information criteria value in the sample.
- Estimate the best model by the one appearing most frequently in the sample.
- Estimate the best model by choosing variables with large estimates of the probability of "success" (be selected).

Actually, the first two criteria are asymptotically equivalent.

### 4.3.1 Gibbs sampler

Gibbs sampler, a Markov Chain Monte Carlo method, generates Markov chain sample of a random vector using conditional probability of each dimension sequentially. In model selection framework, each model $\alpha \in \mathcal{A}$ can be denoted as a $p$-dimension binary vector (4.1). Therefore, Gibbs sampler generates values in each dimension conditioning other dimensions are fixed in some proper order. And after all $p$ dimensions has been updated, a
new model is added into the sample chain. By repeating the previous step, a sample chain can be attained.

One great advantage of Gibbs sample is that there is no need to calculated the complicated normalized constant 
\[ H = \sum_{\alpha_j \in A} \exp \left( -f(\alpha_j | Y_n, X_n) \right) \]. The conditional probability is quite easy to calculated. Actually, it only involves two models. For example, the kth condition distribution of \( \alpha = (v_1, v_2, \ldots, v_p) \). Without loss of generality, we may assume for the current model \( \alpha, v_k = 0 \) and the alternative one \( \alpha' \) has \( v_k = 1 \) fixing other variables. By the law of total probability, the conditional distribution of kth variable can be derived as:

\[
P(\alpha' | v_1, \cdots, v_{k-1}, v_{k+1}, \cdots, v_p) \]
\[
= \frac{P(v_1, \cdots, v_k = 1, \cdots, v_p)}{P(v_1, \cdots, v_{k-1}, v_{k+1}, \cdots, v_p)}
\]
\[
= \frac{P(\alpha')}{P(\alpha) + P(\alpha')}
\]
\[
= \frac{\exp(-f(\alpha' | Y_n, X_n))}{\sum_{\alpha_j \in A} \exp(-f(\alpha_j | Y_n, X_n))}
\]
\[
= \frac{\exp(-f(\alpha' | Y_n, X_n)) + \sum_{\alpha_j \in A} \exp(-f(\alpha_j | Y_n, X_n))}{\sum_{\alpha_j \in A} \exp(-f(\alpha_j | Y_n, X_n)) + \sum_{\alpha_j \in A} \exp(-f(\alpha_j | Y_n, X_n))}
\]

We purpose two simulation order here. One is sequential which updates dimensions one by one in order. Another is random, which updates dimensions randomly with equal probability. Random theme has more volatility, but each step can be viewed as a Markov chain, while it take \( p \) steps to form a valid Markov chain for sequential theme. Hence, random theme is more computational efficient.
Sequential theme  Base on the idea of Gibbs sampler, we first choose a particular variable to simulated. While keep other variables fixed, we calculate the conditional probability on those fixed ones. Then we simulated the next model.

- Arbitrarily choose a starting model \( v^{(0)} = (v_1^{(0)}, \cdots, v_p^{(0)}) \).
- Repeat for \( j = 1, \cdots, K \),
  - Repeat for \( i = 1, \cdots, p \), generate \( v_i^{(j)} = 1 \) with probability
    \[
    P \left( v_i^{(j)} \mid v_1^{(j)}, \cdots, v_{i-1}^{(j)}, v_{i+1}^{(j-1)}, \cdots, v_p^{(j-1)} \right) = \frac{\exp(-f(\alpha_1))}{\exp(-f(\alpha_0)) + \exp(-f(\alpha_1))} \tag{4.15}
    \]
    where \( \alpha_1 = \left( v_i^{(j)}, \cdots, v_{i-1}^{(j)}, 1, v_{i+1}^{(j-1)}, \cdots, v_p^{(j-1)} \right) \), \( \alpha_0 = \left( v_i^{(j)}, \cdots, v_{i-1}^{(j)}, 0, v_{i+1}^{(j-1)}, \cdots, v_p^{(j-1)} \right) \).
- Return the model sequence \( \{ v^{(1)}, v^{(2)}, \cdots, v^{(K)} \} \).

As a Markov chain  Let’s check the characteristics of this algorithm.

- the new sample in a chain only depends on the previous sample in the chain only. This fits the definition of Markov chain.
- For each sample in the chain, there’s a positive probability that the next sample would still stay in the same state, which gives us aperiodic.
- For each state, there is positive probability that it can be reach from any other state. This gives us positive recurrent. Together with aperiodic, this Markov chain is ergodic.
- By Theorem 3.2, we know this Gibbs sampler defined on \( \mathcal{A} \) is a well defined Markov and has a stationary distribution \( \pi \). And we denote the transition matrix of Gibbs sampler as \( P \).
Let’s have a close look at the algorithm. For any \( j \), in the \( i \)th inner step, the Gibbs sampler either stay at the current state or move to a new state which altered the \( i \)th dimension of current state. Furthermore, probability jump to the new model \( 4.15 \) only depends on the current model and the candidate model, which suggests that for \( i \)th inner step, given the same model as current state, the probability will remain same as well. This means for \( i \)th inner step alone, the Gibbs sampler can be viewed as a new Markov chain with transition matrix \( P^{(i)} \). \( P^{(i)} \) is a sparse matrix. In each row, there are only two non-zero positive values, which indicates the probability stays at the current state and the probability jump to the candidate state respectively.

For \( p \) inner steps, we will have \( P^{(1)}, \ldots, P^{(p)} \) transition matrices. They are independent of \( j \). If we compound all \( p \) inner steps together, we will obtain the Markov chain process for the sample chain, the entire Gibbs sampler algorithm, with transition matrix

\[
P = P^{(1)}P^{(2)}\cdots P^{(p)} = \prod_{i=1}^{p} P^{(i)}. \tag{4.16}
\]

Following proposition shows that this stationary distribution \( \pi \) is exactly the Gibbs distribution \( 4.14 \).

**Proposition 4.1.** The stationary distribution of this Markov chain is exactly the Gibbs distribution \( 4.14 \).

**Proof.** We use Theorem \[\text{3.1}\] to verify this. Let \( \pi \) be the Gibbs distribution, where \( \pi_i = \frac{\exp(-f(\alpha_i))}{\sum_{j \in S} \exp(-f(\alpha_j))} \). For some sequential transition matrix \( P^{(k)} \), denote \( \pi^* = P^{(k)}\pi \). Since, for model \( \alpha_i \), \( P^{(k)} \) only assign probabilities to states \( \alpha_i \) and \( \alpha'_i \), where \( \alpha_j = \alpha'_i \) is \( \alpha_i \) altered the \( k \)th variable. Thus we have:
\[ \pi_i = \frac{\exp(-f(\alpha_i))}{\sum_{\alpha_j \in A} \exp(-f(\alpha_j))} \exp\left(\frac{-f(\alpha_i)}{\sum_{\alpha_j \in A} \exp(-f(\alpha_j))}\right) + \exp\left(\frac{-f(\alpha_i)}{\sum_{\alpha_j \in A} \exp(-f(\alpha_j))}\right) \]

which gives \( \pi = P^{(k)} \pi \) for all \( k \). Furthermore, we have \( \pi = \prod_{i=1}^{p} p^{(i)} \pi = P \pi. \)

**Algorithm 1** Gibbs Sampler Algorithm for MCMC Model Selection under sequential theme

\[ v \leftarrow \left(v_1^{(0)}, \ldots, v_p^{(0)}\right). \{\text{An arbitrary starting model}\} \]

\[ i \leftarrow 1 \]

for \( j = 1 \rightarrow K \) do

\[ v' \leftarrow \left(v_1, \ldots, (1 - v_i), \ldots, v_p\right) \{\text{The candidate model}\} \]

Calculate the information criteria \( f_v, f_{v'} \) for models \( \alpha_v \) and \( \alpha_{v'} \)

Generate \( p \) from Unif(0,1)

if \( p < \frac{\exp(-f_{v'})}{\exp(-f_v) + \exp(-f_{v'})} \) then

\[ v \leftarrow v' \]

end if

if \( i \geq p \) then

\[ i \leftarrow 1 \]

else

\[ i \leftarrow i + 1 \]

end if

\[ v^{(j)} \leftarrow v \]

end for

return Chain \( \{v^{(1)}, \ldots, v^{(K)}\} \)

In the standard Gibbs sampler algorithm, in order to obtain a sample in chain, \( p \) addition regressions need to be done. There are chances that the best model may appear in these inner steps instead of in the final output sample chain recorded. Furthermore, in order to maximized computational
efficiency, we suggest to record the model selected in each inner step as well. Algorithm 1 is the algorithm of the modified Markov chain. Proposition 4.2 tells us that it still can be used to as samples of the Gibbs distribution. As a drawback, the samples in this new sample chain will have a much stronger dependency.

**Proposition 4.2.** Algorithm 1 forms a time-inhomogeneous Markov chain with the same gibbs distribution as the stationary distribution

**Proof.** The new chain is actually a time-inhomogeneous Markov chain, since the transition matrix is not identical, but every $p$ models in the chain forms a subchain of some time-homogeneous Markov with the Gibbs distribution as its stationary distribution. Therefore, the new chain can be evenly divided into $p$ subchains. Each subchain is some sample chain with the same stationary distribution, the Gibbs distribution. Then, we can conclude that this new chain generated by 1 will have the Gibbs distribution as its stationary distribution as well.

**Random Theme** Sequential theme searches models one variable by another in order. It takes $p$ outcomes of Gibbs sampler to obtain a well-defined time-homogeneous Markov chain sample. In random theme, the dimension being updated in randomly chosen. In this case, the transition matrix would be identical in each step, which makes this Gibbs sampler itself a time-homogeneous Markov chain. Furthermore, the sample generated by this Gibbs sampler is less dependent.
Algorithm 2  Gibbs Sampler Algorithm for MCMC Model Selection under random theme

\[ v \leftarrow \left( v_1^{(0)}, \ldots, v_p^{(0)} \right). \{ \text{An arbitrary starting model} \} \]

for \( j = 1 \rightarrow K \) do

\[ i \leftarrow \text{integers between } 1 \text{ and } p \text{ with equal chances} \]

\[ v' \leftarrow \left( v_1, \ldots, (1-v_i), \ldots, v_p \right). \{ \text{The candidate model} \} \]

Calculate the information criteria \( f_v, f_{v'} \) for models \( \alpha_v \) and \( \alpha_{v'} \)

Generate \( p \) from Unif(0,1)

if \( p < \frac{\exp(-f_{v'})}{\exp(-f_v)+\exp(-f_{v'})} \) then

\[ v \leftarrow v' \]

end if

end for

return Chain \( \{ v^{(1)}, \ldots, v^{(K)} \} \)

---

**Corollary 4.3.** Gibbs distribution is also the stationary distribution for random theme Markov chain.

*Proof.* For model \( \alpha_i \), this Markov chain assigns probabilities to \( p \) models instead of 2. By adding up the probabilities from \( p \) related models, we obtain \( \pi^* = \pi \) again. \( \square \)

### 4.3.2 Another convergence diagnosis method

**Inspiration from Yu.** As we all know, the diagnosis of MCMC convergence is always a difficult task, which is usually achieved by a combination of different methods. Apart from varieties of MCMC convergence diagnosis methods introduced in Section 3.3, here we present a specific diagnosis procedure modified from [Yu, 1994]. Her idea was for a specific form of stationary distribution

\[
\frac{f(x)}{\kappa}
\]

where \( f(\cdot) \) is known, while \( \kappa \) is a normalization constant very hard to compute. Instead of estimating the stationary itself, she suggests to estimate \( \kappa \), by means of calculating the mean ratio of kernel density over function \( f(\cdot) \). This is also referred as Rao-Blackwellised MCMC estimator in some litera-
ture. Then the convergence of MCMC can be test by the difference between kernel density and estimated stationary distribution in some sense.

**Another diagnosis method.** In our model selection scenario, we simplified her method into discrete form. The stationary distribution of our MCMC is some Gibbs distribution (4.14) with some unknown normalization constant $\kappa$ in the denominator

$$\kappa = \sum_{\alpha_j \in \mathcal{A}} \exp \left( - f(\alpha_j | Y_n, X_n) \right)$$

(4.17)

This means, the shape of the stationary distribution is already known. As long as an estimation of $\kappa$ is obtained, we are allowed to the difference compare the stationary distribution and our sample distribution. Here, we apply the same Rao-Blackwellised MCMC estimator of $\kappa$

$$\hat{\kappa} = \frac{1}{\#(\mathcal{B})} \sum_{\alpha_i \in \mathcal{B}} \frac{\exp \left( - f(\alpha_i) \right)}{\text{Sample frequency of } \alpha_i}$$

(4.18)

where $\mathcal{B}$ is the set of states appeared in the sample chain.

Here we present a simple graphical diagnosis procedure. This method simply compares the sample frequency with the estimated probabilities among states, i.e.

$$\frac{\exp(-f(\alpha_i))}{\hat{\kappa}} \text{ vs Sample frequency of } \alpha_i$$

The results of this diagnosis is actually very strong comparing with the previous methods introduced. First of all, it is state-wised (convergence in probability) instead of focusing on the distribution of selection criteria (convergence in distribution). Secondly, the object of convergence is fixed in some sense, rather than some Cauchy convergence methods. From another point of view, since we utilized function $f(\cdot)$, this diagnosis access more information than other methods, thus it is expected to have better performance.
A demonstration. Figure 4.1 is the graphic diagnosis of the 20 states with largest sample frequency respectively of some single chain (from Section 5.1) with different lengths. The first graph with chain size 500 is the first 500 outputs in the chain. The second graph with chain size 8160 is the rare part of a 10200 chain cutting the first one-fifth as burnin period. The solid line is the estimated state probability while the dots are sample frequencies. This comparison demonstrates what a good convergence would look like.

However, since this diagnosis is too powerful, it may need a very long chain to reach its convergence conclusion, which may not be affordable in some cases. Fortunately, as for model selection, we are not interested in distribution of all states, but those with highest probabilities, i.e. smaller AIC, BIC. Therefore, we may loosen our diagnose criteria. Instead of looking for proof of convergence in probability, we only check the first few outstanding states only, and use the solid line to access whether the variance is acceptable. Al-
though by doing this we have little confidence of convergence of the process as whole, the estimator the first few best models are still reliable, and that's all we are care about.

**Confidence in the results.** Moreover, once the convergence is diagnosed, we can calculate a confidence of the best model in the sample is the best model in the model space through geometric distribution assuming weak dependence:

\[
\left(1 - \frac{\exp(-f(\alpha_{\text{best}}))}{\hat{\kappa}}\right)^M
\]

Usually, when the difference of AICs between two models is more than 2, the difference is regarded as significant. Another confidence can be calculated.

\[
\left(1 - \frac{\exp(-f(\alpha_{\text{best}}) + 2)}{\hat{\kappa}}\right)^M
\]

where \(M\) is the sample chain length after burn-in period, \(\alpha_{\text{best}}\) is the best model in the sample chain.

### 4.3.3 Discussion

MCMC model selection use Gibbs sampler work pretty well with many potential predicting variables, say \(p = 20\). However, when \(p\) keeps growing, Gibbs sampler gradually loses its power.

For example, if there are 40 potential predicting variables, then \(\mathcal{M}\) will have \(2^{40} > 10^{12}\) different models in total. One terrible but possible scenario is that even with a sample chain with \(10^5\) in length, there are \(10^5\) different models in it. This may suggest the procedure may need a much longer sample chain to obtain a meaningful result. In other words, a workable length of sample chain grows as fast as the size of model space \(\mathcal{M}\) when \(p\) increases. Furthermore, when we include more variables, each regression fitting will require more time. These facts make Gibbs sample less efficient when with dealing with huge number of potential predicting variables.
Qian and Field [2002] presented a two-step random search procedure to handle this difficulty. In the first step, it does a normal MCMC model search. Regardless of the fact that the chain may still not reach its equilibrium, in the second step, the variables appeared in the best model seen in the first step are used to construct a new sub model space, and another MCMC model selection procedure is done on that subspace. One defect of this method is that it is difficult to do inference on the result, i.e. with how much confidence can we say that the model we have in hand is the best model with lowest information criterion values in the model space. In next section, we are going to apply some concepts from simulated annealing to improve the Gibbs sampler for model selection from another prospect of view.

4.4 Simulated annealing model selection diagnostic toolkit

4.4.1 Introduction

In MCMC model selection, there are usually two ways of choosing models, pick the model with lowest information criterion value in the sample chain or choose a set of variables appear most often in the models in the sample chain. However, there are practical problems lies in these strategies.

- In the case of search the lowest information criterion values, it is very hard to verify the convergence of Markov chain sample. Without certain degree of confidence in the convergence of the Markov chain, it’s quite meaningless to look optimal model in the chain. Thus the only way to gain more confidence is to run a much longer simulation.

- When using the marginal frequency of explanatory variables, it’s difficult to set a cutoff level. Variables can locate very close to each other in terms of marginal frequency. And in some situations where explanatory variables are highly correlated, some variables with high
marginal frequency are just acting as substitutions of some true variables, while some important variables with relatively lower marginal frequency may get excluded from the selection process.

In this section, we present a diagnostic toolkit to ease these problems to some extent. Basically, we introduce a temperature parameter $T$ from simulated annealing technique into the Gibbs distribution. By controlling $T$, we are able to accelerate the convergence of the Markov Chain, and extract more information from the marginal frequency selection.

Simulated annealing is originally attached to the Metropolis-Hasting (HM) algorithm by introducing a parameter temperature $T$ to control the likelihood of accepting a new candidate state mentioned by Aarts and Korst [1988]. It is also feasible to assemble it into Gibbs sampler for model selection. Gibbs sampler here is defined on a discrete space, In each step, we generated a new sample from a conditional distribution. Since it is a binary space, each time we have only two outcomes, either stays in the current model or jump to the new model. This is very similar to the HM algorithm.

Previously, we use (3.16) as our new Gibbs distribution. Then immediately we have the conditional distribution at temperature $T$:

$$
\Pr_T(\alpha'|\alpha \setminus v_k) = \frac{\exp\left(-\frac{f(\alpha')}{T}\right)}{\exp\left(-\frac{f(\alpha)}{T}\right) + \exp\left(-\frac{f(\alpha')}{T}\right)}
$$

(4.21)

where $\alpha'$ is the candidate model altered $k$th variable from $\alpha$, the current model.

Here we are proposing utilizing Simulated Annealing in two ways, Fixed Temperature Search (FTS) and with Cooling Schedules Search (CSS). In FTS, by choosing an appropriate temperature constant $T$, we can increase the probability of better models, thus making them appear sooner and more often. For FTS, we try $T$ as a function of time $t$. It is a more aggressive
option, which searches local minima in a very fast speed, but loses the power of marginal frequency search of predicting variables.

4.4.2 Fixed Temperature Search

From Proposition 3.17 and 3.18, we know asymptotically the stationary distribution would favor the model with lowest information values when temperature $T$ decreases. However the connectivity between states gets weaken when as well. The Markov chain will lose ergodicity eventually when temperature is low enough. Here we introduce some statistic of the sample chain, average odds (AO) to capture affection of temperature on the Gibbs sampler.

Acceptance ratio $\chi_A$ presented by [Aarts and Korst 1988] is one of the simple ways to access the response of annealing process based on MH algorithm to temperatures.

**Definition.** Acceptance ratio $\chi(T)$

$$\chi_A(T) = \frac{\text{number of accepted new models}}{\text{number of proposed new models}}$$

(4.22)

This ratio was originally resembled in Metropolis-Hasting method, hence $\chi_A(t)$ has a range of $(0, 1)$. Furthermore, $\chi_A$ is actually an empirical parameter come alone with simulations. In Short, it describes the volatility variation among change of temperatures. However, in MCMC model selection context, we use Gibbs sampler as the Markov chain process. Here we introduce another statistic called Average Odds (AO). We found out that it has less variance given a sample chain of short length.

**Temperature and odds** Let’s have a close look at the mechanics of Gibbs sampler. In every inner step $i$ (Algorithm 1), We start from a current model $\alpha$ with score $f_0$. Then a candidate model $\alpha'$ is proposed with score $f_1$. We
choose the new model with probability

\[ p(\alpha, \alpha', T) = \frac{\exp(-f(\alpha'))}{\exp(-f(\alpha')) + \exp(-f(\alpha)/T)} \]

Thus we can calculate the odds of choosing \( \alpha' \) under temperature \( T \).

\[ O_T(\alpha', \alpha) = \frac{p(\alpha, \alpha', T)}{1 - p(\alpha, \alpha', T)} = \frac{\exp(-f(\alpha'))}{\exp(-f(\alpha')) + \exp(-f(\alpha)/T)} \]

\[ = \left\{ O_1(\alpha', \alpha) \right\}^{1/T} = \exp\left\{ \frac{\Delta f}{T} \right\} \]

where \( O_1(\alpha', \alpha) \) is the odds when temperature is 1, and \( \Delta f = f(\alpha) - f(\alpha') \).

Furthermore, we define \( O^*, O^*_T \) to be the odds for worse model instead of the better model. Immediately, we have \( O^* \leq 1 \), and \( O^*_T(T) \) is a monotonically increasing function of \( T \). Also, asymptotically, we have

\[ \lim_{T \to 0} O^*_T = 0 \quad \text{and} \quad \lim_{T \to \infty} O^*_T = 1 \]

Hence, when \( T \to 0 \), the process with choose the better model for sure. When \( T \to \infty \), the process with choose two models equally likely. This is consistent with the asymptotic result of the stationary distribution. To access the over all affection of temperature on Gibbs sample in stead of in a particular inner step, we simple take the average of \( O^*_T \)s.

**Definition.** Average odds for worse model \( \chi \)

\[ \chi(T) = \frac{1}{n} \sum_{i} O^*_i(\alpha_i, \alpha'_i) = \frac{1}{n} \sum_{i} \left( \frac{p_i(T)}{1 - p_i(T)} \right)^{2I_{[p_i(T) > 0.5]}} \]

where \( p_i(T) = p(\alpha_i, \alpha'_i, T) \) is the conditional probability of choosing the candidate model in \( i \)th step with model \( \alpha_i \) and candidate model \( \alpha'_i \); \( O^*_i(T) \)
is the odds of staying in poorer model in $i$th step with temperature $t$. Since the better model always gets more than 50% chances to be chosen, the $O_i^t(T) \in (0, 1)$.

Figure 4.2 illustrates how the average odds and acceptance ratio response to temperature change for some simulation study.

Figure 4.2: Relationship between A.O. and A.R. under different temperatures

**Proposition 4.4.** *When $\chi_0 = 1$, the Gibbs sampler will have the widest spread, i.e. the maximum entropy.*

**Proof.** The marginal distribution of a model space with $p$ variables starting from $\alpha_0 = (0,0,\ldots,0)$ after $p$ inner steps. In the first inner step, it assigns $\frac{1}{2}$ to $\alpha_0$ and $\frac{1}{2}$ to $(1,0,\ldots,0)$. In the second inner step, conditioning on $(1,0,\ldots,0)$ it assigns $\frac{1}{2}$ to $(1,1,0,\ldots,0)$ and $\frac{1}{2}$ to $(1,0,0,\ldots,0)$. At this stage, it assigns $\frac{1}{4}$ to four models. In $i$ inner step, conditioning on each model in previous step, a new model will be assign $\frac{1}{2}$ by adding the $i$th variable, and $\frac{1}{2}$ to itself again. In total, models in $(i - 1)$th inner step will
have a probability $2^{-i}$, and there would be $2^{i-1}$ new models assigned $2^{-i}$ in $i$th step. Finally, after $p$ inner steps, there would be $2^p$ models assigned with same probability $2^{-p}$. Actually, this is a uniform distribution on $\mathcal{A}$, which has maximum entropy.

In fact, temperature $T$ exactly controls the entropy of the whole regression model space $\mathcal{A}$. When temperature keeps going up, $\chi_o \to 1$, each model becomes equally important, i.e. having equal probability. Contrarily, when temperature keeps going down, $\chi_o \to 0$, Gibbs sampler will jump to better model for sure.

**Average Odds vs acceptance ratio**  Average Odds is more reliable than acceptance ratio when the length of sample chain $n$ is small due to it collects more information. Firstly, Average Odds reveals the process’s preference for worse model, while acceptance ratio only illustrates preference for new model. Secondly, Average Odds is calculated through the difference of information criterion values between the original model and candidate model in each step, while accepted ratio is calculated by counts only.

The following table 4.1 compares the variance of A.O. and A.R. of 200-length sample chain in the first simulation study Section 5.1. We assume A.R. is the average of some Bernoulli r.v. And use central limited theorem to estimate the variance of A.O. where the variance of Odds estimated from a $10^5$ sample chain. In the table, the variance of A.O. is almost 10 times less then the variance of A.R. consistently. This result is exactly what we expected. Figure 4.3 gives a more detailed illustration of how do A.O. and A.R converges asymptotically when the sample size increases. Actually, a 40 length chain would give enough precision to estimated A.O. for this simulation study.
4.4.3 Cooling Schedules Search

In this subsection, we first defined a cooling schedule and derived its algorithm. Then we discussed its Markov chain properties and vaguely described what the stationary distribution would be.

A cooling schedule is a temperature function $T(t)$ of time $t$. It defines how temperature moves from the initial temperature to the terminal temperature. This is core idea of simulated annealing. In general, a cooling schedule has to satisfied three criteria. Firstly, stay in a zone of temperature high enough long enough so that the process could gain enough volatility to jump out of any local minima. Secondly, stay in a zone of temperature cold enough and long enough so that the process could reach the local minimum. Lastly, it should have a smooth transition from time to time.

In this thesis, we constructed an exponential periodic cooling schedules combining techniques of quenching and re-annealing. Since temperature is always changing in the algorithm. The Markov chain process is no longer time-homogeneous. This may break many nice property of Gibbs sampler in model selection, especially for convergence diagnosis. In stead of focusing on asymptotically convergence to the optimal solution, we intend to construct a new time-homogeneous Markov chain process consisting of quite a few time-inhomogeneous Markov chains. It preserves some properties of

<table>
<thead>
<tr>
<th>Temp.</th>
<th>A.O.</th>
<th>A.R.</th>
<th>$\text{Var(A.O.)}$</th>
<th>$\text{Var(A.R.)}$</th>
<th>$\text{Var(Odds)}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.013</td>
<td>0.010</td>
<td>0.0006</td>
<td>0.0035</td>
<td>0.0087</td>
</tr>
<tr>
<td>8</td>
<td>0.055</td>
<td>0.070</td>
<td>0.0008</td>
<td>0.0083</td>
<td>0.0124</td>
</tr>
<tr>
<td>12</td>
<td>0.141</td>
<td>0.220</td>
<td>0.0011</td>
<td>0.0128</td>
<td>0.0158</td>
</tr>
<tr>
<td>18</td>
<td>0.300</td>
<td>0.335</td>
<td>0.0016</td>
<td>0.0154</td>
<td>0.0222</td>
</tr>
<tr>
<td>30</td>
<td>0.461</td>
<td>0.425</td>
<td>0.0024</td>
<td>0.0170</td>
<td>0.0345</td>
</tr>
<tr>
<td>75</td>
<td>0.700</td>
<td>0.465</td>
<td>0.0033</td>
<td>0.0175</td>
<td>0.0468</td>
</tr>
<tr>
<td>250</td>
<td>0.874</td>
<td>0.475</td>
<td>0.0017</td>
<td>0.0177</td>
<td>0.023</td>
</tr>
<tr>
<td>750</td>
<td>0.949</td>
<td>0.490</td>
<td>0.0004</td>
<td>0.0177</td>
<td>0.006</td>
</tr>
</tbody>
</table>

Table 4.1: Variance comparation between A.O & A.R.
4.4. SIMULATED ANNEALING MODEL SELECTION DIAGNOSTIC TOOLKIT

Figure 4.3: Convergence property of A.O & A.R.
the original Gibbs sampler so that some diagnosis may still be done for the
sample chain.

For convenience, we divide a annealing period into three segments. They
are Hot Zone characterized by $T_0$, Cold Zone characterized by $T_n$, and Trans-
ition Zone which cover the rest. In Hot Zone, the annealed Gibbs sampler
behaves approximate to uniform random sampler. The idea is that the pro-
cess is able to jump out of the area dominated by current local minimum. In
Cold Zone, the randomness of annealed Gibbs sample is almost eliminated.
It chooses better model almost surely. In Transition Zone, annealed Gibbs
sample moves smoothly from hot to cold where all the characteristics of our
Gibbs sampler is carried.

For quenching, we use exponential cooling schedules. For re-annealing,
we treat the cooling schedules as a periodic function. In each period, tem-
perature declines from the Hot Zone to Cold Zone exponential, and start
over in the next period. By fixing length the period instead of terminate
the declining by some stopping criteria, each period become an identical
stochastic process. Actually this is some new time-homogeneous Markov
Chain. All the handful properties be used for inference again, while sample
space has become much smaller, i.e. a space of local minima.

Some notations used in CSS:

- $n$ is the totally length of a single annealing period, within which $n$
different temperatures are applied.

- $M$ be the number of annealing periods simulated.

- $T_0$ be the initial Temperature.

- $T_n$ be the terminal Temperature, as well as the coldest Temperature.
The cooling schedules within each period is defined as:

\[ T(t) = T_0 \left( \frac{T_n}{T_0} \right)^{t/n}. \]  \hspace{1cm} (4.25)

As an exponential cooling schedule, the decreasing factor \( \gamma \) is defined as

\[ \gamma = \left( \frac{T_n}{T_0} \right)^{1/n} \]  \hspace{1cm} (4.26)

The algorithm of CSS can be summarized as following:

**Algorithm 3** CSS Algorithm for MCMC Model Selection under sequential theme

\[ v \leftarrow \left( v_1^{(0)}, \ldots, v_p^{(0)} \right). \text{ \{An arbitrary initial model\} } \]
\[ i \leftarrow 1 \]
\[ \text{for } j = 1 \rightarrow K \text{ do} \]
\[ v' \leftarrow \left( v_1, \ldots, 1 - v_i, \ldots, v_p \right). \text{ \{The candidate model\} } \]
\[ \text{Calculate the information criteria } f_v, f_{v'} \text{ for models } \alpha_v \text{ and } \alpha_{v'} \]
\[ \text{Generate } p \text{ from Unif}(0,1) \]
\[ \text{Calculate Temperature } T(i) \]
\[ \text{if } p < \frac{\exp(-f_{v'}/T(i))}{\exp(-f_v/T(i)) + \exp(-f_{v'}/T(i))} \text{ then} \]
\[ v' \leftarrow v \]
\[ \text{end if} \]
\[ \text{if } i \geq p \text{ then} \]
\[ i \leftarrow 1 \]
\[ \text{else} \]
\[ i \leftarrow i + 1 \]
\[ \text{end if} \]
\[ v^{(j)} \leftarrow v \]
\[ \text{end for} \]
\[ \text{return } \text{Chain } \left\{ v^{(1)}, \ldots, v^{(K)} \right\} \]

Choosing the temperature parameters by Average Odds  \hspace{1cm} We use Average Odds as a guideline to select the initial temperature \( T_0 \) and terminal temperature \( T_n \) for our cooling schedules. AO increases to 1 when temperature keeps climbing, while the algorithm tends to reject worse models with greater probability when temperature declines. By referencing it, enable us
to choose suitable initial temperature $T_0$ such that $O_{T_0}^* \to 1$, and terminal temperature $T_f$ such that $O_{T_f}^* \to 0$.

After $T_0, T_f$ is chosen, we need to setup the length of each single annealing period $n$. The decreasing factor $\gamma$ can be immediately calculated (4.26). In other words, the greater the $n$ is, the more smoothing the temperature declines.

**Proposition 4.5.** Given fixed length cooling schedule with length $n$, SA for model selection under sequential theme is a time-homogeneous Markov chain regarding each period as a single step, if $n = kp$ for some $k \in \mathbb{N}$.

**Proof.** Gibbs sampler for model selection under sequential theme has transition matrix $P$ (4.16). In SA, the transition matrix $P_T^{(i)}$ is a function of temperature $T$ for $i$th inner step in the algorithm. In a single period, the compound transition matrix $P_{SA}$ is

$$P_{SA} = P_T^{(1)} P_T^{(2)} \cdots P_T^{(p)} \cdots P_T^{(1)} P_T^{(2)} \cdots P_T^{(1)} P_T^{(2p)} \cdots P_T^{(1)} .$$

In the next period, it starts from $P_T^{(1)}$ and ends at $P_T^{(p)}$ again. Hence $P_{SA}$ is identical for each period, which forms a time-homogeneous Markov chain from certain point of view.

With the suitable parameters, i.e. $T_0$ is hot enough, $T_f$ is cold enough, $n$ is long enough to overcome some minor local optimal, we will have some nice properties of this Markov chain.

**Stationary distribution of Cooling Schedules Search** Starting from arbitrary model $\alpha$, since the existence of Hot Zone, $\alpha$ is communicated to any model in the model space. And due to Cold Zone, the outcome at the end of each period is a local optimal. Take one period as burning period, SA can be viewed as a Markov chain on the space of local optimums. After all, the conditions of Theorem 3.2 is satisfied. Hence we know it has a stationary distribution. The follow Proposition 4.6 shows that with our exponential pe-
4.4. SIMULATED ANNEALING MODEL SELECTION DIAGNOSTIC TOOLKIT

Periodic cooling schedules, the stationary probability of local minima increase when temperature decreases.

**Proposition 4.6.** Given some arbitrary measure \( \pi_0 \) on model space \( \mathcal{A} \) with information criteria \( f \). Say \( \alpha_k \) is a local minimum. Denote \( [\pi]_k = P(\alpha_k) \) under measure \( \pi \). We have

\[
[\pi_0P_{T_1}^{(j)}]_k < [\pi_0P_{T_2}^{(j)}]_k
\]

for \( j \in \{1, 2, \ldots, p\} \) if \( T_1 \geq T_2 \).

**Proof.**

\[
[\pi_0P_{T_1}^{(j)}]_k = [\pi_0]_kP_{T_1, kk}^{(j)} + [\pi_0]_jP_{T_1, jk}^{(j)}
\]

\[
= [\pi_0]_k \frac{\exp \left( -\frac{f(\alpha_k)}{T_1} \right)}{\exp \left( -\frac{f(\alpha_k)}{T_1} \right) + \exp \left( -\frac{f(\alpha'_k)}{T_1} \right)}
\]

\[
+ [\pi_0]_j \frac{\exp \left( -\frac{f(\alpha_k)}{T_1} \right)}{\exp \left( -\frac{f(\alpha_k)}{T_1} \right) + \exp \left( -\frac{f(\alpha'_k)}{T_1} \right)}
\]

\[
= \left( [\pi_0]_j + [\pi_0]_k \right) \frac{1}{1 + \exp \left( \frac{f(\alpha_k) - f(\alpha'_k)}{T_1} \right)}
\]

\[
\leq \left( [\pi_0]_j + [\pi_0]_k \right) \frac{1}{1 + \exp \left( \frac{f(\alpha_k) - f(\alpha'_k)}{T_2} \right)} = [\pi_0P_{T_2}^{(j)}]_k
\]

where \( \alpha'_k \) is adding or removing \( j \)th variable of model \( \alpha_k \). And \( f(\alpha_k) < f(\alpha'_k) \), since \( \alpha'_k \) is in the neighborhood of local minimum \( \alpha_k \). \( \square \)

In a single period of our cooling schedules, the temperature is always decreasing. No matter what initial model is, the probability of the simulated annealed Gibbs sampler choosing a local minimum at the end of period is always increasing. This proves that for CSS, the stationary distribution will more likely to assign probability to local minima, which explains why CSS is a much faster method in searching global minimum.
4.4.4 Remarks

If the MCMC has reached its equilibrium, given enough time, the marginal frequency of each state would converge to its stationary distribution. From that point, we can regard the sample chain as a sample of the Gibbs distribution we construct previously. In that distribution, a better model has a lower information criterion value, will have a higher marginal frequency in the sample. Say we have obtained a length of $N$ sample after equilibrium. Denote $\alpha_{(1)}$ for the best model in the sample chain, $\hat{\theta}_{(1)}$ for its marginal frequency within the sample. We would expected that $\hat{\theta}_{(1)} \leq \hat{\theta}_i$ for all other model $\alpha_i$ appeared in the sample, otherwise the MCMC is not convergent. If $\alpha_{(1)}$ is not the true best model $\alpha_{\text{best}}$ then we would expect that $\theta_{\text{best}} \geq \theta_{(1)}$. And the probability of the true best model $\alpha_{\text{best}}$ is not inside the sample can be calculated assuming the dependence of samples are weak

$$P(\alpha_{\text{best}} \text{ didn’t appear}) \geq (1 - \hat{\theta}_{(1)})^N$$

Asymptotically, when $N \to \infty$, $P(\alpha_{\text{best}} \text{ didn’t appear}) \to 0$. Given the convergence of MCMC, we can obtain a confidence estimator of finding the true best model, which makes diagnosis of convergence to be a crucial step from drawing a conclusion the random search procure.
Chapter 5

Simulation study

In this chapter, we studied the performance of MCMC Model Selection Procedure through from 3 aspects:

• Compare the simulation results with theoretical facts.

• Test on data set involving large number of explanatory variables.

• Do an comparative study with other data mining methods in terms of mis-classification rate.

In each case, we both applied Fix Temperature Search and Cooling Schedules Search from our diagnostic toolbox. It turns out that our toolbox did a satisfactory jobs in accelerating the process and extracting more useful information.

5.1 Simulated dataset

This is a simulation study on an artificial dataset, which means the true model is known. We first compared AIC & BIC as model selection criteria in finding the best model in the model space. Then for FTS, we compared MCMC model selection results under different temperature settings. There is a trade-off between the degree of acceleration and ergodicity of the Markov chain process. We tried to locate some sweet spot of temperature
range for FTS according to the corresponding Average Odds so that both high degree of acceleration and ergodicity is achieved. Furthermore, we observed an interesting phenomenon, the marginal frequencies of important variables tend to increase when the temperature is decreasing. We gave some explanation of that and further use that as an informative method to pick out important variables. For CSS, we set up a cooling schedule according to the information of temperature and Average Odds obtained in previous FTS. CSS did a brilliant job in finding local and global minima.

5.1.1 Data description

In this simulation, we construct 300 data points with 20 explanatory variables, which are uniformly generated from \([-1, 1]\). The true model The \(y\) observation is generated by equation

\[
Y = f_1(x_5) + f_2(x_{10}) + f_3(x_{15}) + \epsilon, \quad (5.1)
\]

where \(\epsilon \sim N(0, 1)\),

\[
f_1(x) = 2x + \frac{4\exp(-64x^2)}{3}, \quad (5.2)
\]

\[
f_2(x) = 2\sin(2\pi x), \quad (5.3)
\]

\[
f_3(x) = \max(2 - 8x^2, 0). \quad (5.4)
\]

These three functions (5.2) (5.3) (5.4), showed in Figure 5.1, are not local polynomial functions. We would like to find out whether local polynomial can fit these function though a relative small dataset good enough to pick out the corresponding variables in GAM.

Furthermore, we implement some degree of correlations among variables to how GAM with local polynomial fittings handle highly correlated variables. For example, \(x_3, x_4\) has correlations 0.99, 0.95 with \(x_5\). \(x_8, x_{20}\) have correlations 0.78, 0.97 with \(x_{10}\). \(x_3, x_4\) have correlations 0.99, 0.95 with \(x_5\). \(x_8, x_{20}\) have correlations 0.78, 0.97 with \(x_{10}\). This bizarre order of arranging vari-
5.1. SIMULATED DATASET

Figure 5.1: Artificial dataset summary

ables is designed to construct some local minima when we use sequential Gibbs sampler in MCMC Model selection.

5.1.2 Exhaustive search result

For exhausted search, we did regressions on a number of $2^{20} = 1,048,576$ models which constructed by different subset of potential predicting variables. The five best models are listed in Table 5.1. BIC successfully picked out the GAM model with all three true variables, while model with lowest AIC value includes one more noise variable $x_{20}$ high correlated to the true variable $x_{10}$.

Furthermore, all regression results are restore in order to calculate the exact
Gibbs distribution (4.14) used in Gibbs sampler for MCMC model selection. It would also allow us to calculate different exact Gibbs distribution (3.16) under different temperature settings. With the exact stationary distribution of the Markov chains, we are able to access the convergence of MCMC much easier.

### 5.1.3 Fixed temperature search

We did MCMC model selection under different temperature settings using sequential searching theme. We obtained 510 samples for each temperature. Since there are 20 potential predicting variables in this dataset, it takes 20 inner steps to get a sample. In order to take maximum usage of computational power, we simply record all models appeared in inner steps. In each chain, there are in fact 10200 models with 10200 BIC values. In this subsection, we searched the chains for the best model and best subset of predicting variables. And a formal procedure for FTS was presented and demonstrated. First, let check the convergence of MCMC chains.

**Diagnose of convergence**  As a MCMC method, the output can be viewed as valid sample only if the process has reached its equilibrium. Here we conduct I-chart method (3.13) and Gelman and Rubin’s method (3.14) to check convergence.

Table 5.2 is generated by the CODA package in R by Plummer et al. [2006]. The potential scales reduction factor are all close to 1, which suggest very

<table>
<thead>
<tr>
<th>Rank</th>
<th>Models</th>
<th>AIC</th>
<th>BIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$x_5, x_{10}, x_{15}$</td>
<td>454.66</td>
<td>503.11</td>
</tr>
<tr>
<td>2</td>
<td>$x_5, x_{10}, x_{15}, x_{20}$</td>
<td>441.88</td>
<td>506.86</td>
</tr>
<tr>
<td>3</td>
<td>$x_5, x_{10}, x_{15}$</td>
<td>459.62</td>
<td>508.07</td>
</tr>
<tr>
<td>4</td>
<td>$x_5, x_{10}, x_{15}, x_{20}$</td>
<td>447.62</td>
<td>512.61</td>
</tr>
<tr>
<td>5</td>
<td>$x_5, x_6, x_{10}, x_{15}$</td>
<td>452.49</td>
<td>518.69</td>
</tr>
</tbody>
</table>

Table 5.1: 5 best models in exhaustive search by BIC values
Table 5.2: Gelman & Rubin’s convergence test results

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Point Est.</th>
<th>97.5% CI Upperbound</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1.008</td>
<td>1.008</td>
</tr>
<tr>
<td>5</td>
<td>1.012</td>
<td>1.057</td>
</tr>
<tr>
<td>10</td>
<td>1.003</td>
<td>1.016</td>
</tr>
<tr>
<td>20</td>
<td>1.026</td>
<td>1.113</td>
</tr>
<tr>
<td>50</td>
<td>1.008</td>
<td>1.023</td>
</tr>
<tr>
<td>250</td>
<td>1.004</td>
<td>1.007</td>
</tr>
</tbody>
</table>

good convergence. And Figure 5.2 draws I-Chart test results. Almost all BIC values are below the boundary, the thick curve, which suggest good convergence as well. Another thing worth notice is that the boundary proportional to the variance of BIC values increase as temperature goes up. This is consistent with previous study on average odds and acceptance ratio, i.e. the higher the temperature is, the more volatility the process gets. After all, all 6 chains under different temperature settings passed both tests. However this doesn’t mean all 6 chains had converged.

Selecting models by the lowest BIC value Theoretically, the probability of best models increases while temperature decreases. However, in simulation, this property is only feasible in some range of temperatures. When temperature is too low, the process will fail to jump out of some local minimum. When temperature is too high, An very long sample chain is needed for the best model to pop-up in the chain, since the difference between each states is almost unobservable.

Figure 5.3 shows the difference between theoretical probabilities and the probabilities estimated from sample chains, and the best 50 models are listed. The probability of the best model model drops from above 0.6 to $3 \times 10^6$, when temperature increases from 3 to 250. When temperature is 3, the process misses the best model and over 80% of samples in the chain lie in another single state. On the other hand, the process misses almost all important models, when temperature is greater than 50. Temperature
(5, 10) with average odds around (0.05, 0.15) seems a sweet stop for FTS, since the best model has the largest estimated probability and the residuals has less deviance.

If we do the normal MCMC model selection, i.e. $T = 1$, it would totally fail, since the temperature is too cold and the process will be stuck in a local minimum. When using our diagnosis toolbox, only based on some short trial chains, we will be able to choose some temperature between 5 to 10. In that temperature setting, the best model can be easily find and verified, since its probability in range of (0.05, 0.35). This accelerates process significantly.

**Selecting by marginal frequencies** Another important and practical way is selecting predicting variables by its marginal frequencies in the sample chain. Especially, when model spaces is large, i.e. more than 30 explanatory variables, the chances of Gibbs sampler to reach the best model in a sample chain is tiny. The marginal frequencies of variables suggest the preference of the Gibbs sampler in searching for better models, i.e. if including a predicting variable would form a better model, then this variable has larger chances to be included in the models in the sample chain.

Table 5.3 records the marginal frequencies of the true variables $x_5, x_{10}, x_{15}$ and two high correlated variables $x_3, x_{20}$. The A.O. and A.R. are referred to the average odds and acceptance ratio. $\#\alpha_o$ counts the number of appearance of the best model in each sample chain. “Cover.” denoting coverage is the number of different models appeared in the samples. When temperature increases, both A.O., A.R. and coverage increase. The process misses the global minimum until the temperature is greater then 5. And the number of occurrences of the true model decreases after that. And the process failed to sample the true model when temperature is higher then 30.
### Table 5.3: Marginal frequencies in FTSs under different temperatures

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x_3</td>
</tr>
<tr>
<td>1.0</td>
<td>0.00054</td>
<td>0.0016</td>
<td>0</td>
<td>13</td>
<td>1.00</td>
</tr>
<tr>
<td>1.5</td>
<td>0.00244</td>
<td>0.0026</td>
<td>0</td>
<td>13</td>
<td>1.00</td>
</tr>
<tr>
<td>2.0</td>
<td>0.00527</td>
<td>0.0078</td>
<td>0</td>
<td>13</td>
<td>1.00</td>
</tr>
<tr>
<td>2.3</td>
<td>0.00721</td>
<td>0.0102</td>
<td>0</td>
<td>13</td>
<td>1.00</td>
</tr>
<tr>
<td>3.0</td>
<td>0.01247</td>
<td>0.0164</td>
<td>0</td>
<td>22</td>
<td>1.00</td>
</tr>
<tr>
<td>4.0</td>
<td>0.02249</td>
<td>0.0280</td>
<td>0</td>
<td>34</td>
<td>1.00</td>
</tr>
<tr>
<td>5.0</td>
<td>0.03800</td>
<td>0.0522</td>
<td>4066</td>
<td>97</td>
<td>0.17</td>
</tr>
<tr>
<td>6.0</td>
<td>0.05472</td>
<td>0.0772</td>
<td>2880</td>
<td>169</td>
<td>0.19</td>
</tr>
<tr>
<td>7.0</td>
<td>0.07396</td>
<td>0.1055</td>
<td>2148</td>
<td>318</td>
<td>0.20</td>
</tr>
<tr>
<td>8.0</td>
<td>0.09509</td>
<td>0.1367</td>
<td>1408</td>
<td>498</td>
<td>0.30</td>
</tr>
<tr>
<td>9.0</td>
<td>0.11791</td>
<td>0.1620</td>
<td>886</td>
<td>659</td>
<td>0.47</td>
</tr>
<tr>
<td>10.0</td>
<td>0.14037</td>
<td>0.1894</td>
<td>586</td>
<td>840</td>
<td>0.45</td>
</tr>
<tr>
<td>12.0</td>
<td>0.18519</td>
<td>0.2362</td>
<td>176</td>
<td>1257</td>
<td>0.53</td>
</tr>
<tr>
<td>15.0</td>
<td>0.24692</td>
<td>0.2917</td>
<td>35</td>
<td>1804</td>
<td>0.46</td>
</tr>
<tr>
<td>18.0</td>
<td>0.30192</td>
<td>0.3185</td>
<td>50</td>
<td>2114</td>
<td>0.43</td>
</tr>
<tr>
<td>20.0</td>
<td>0.33409</td>
<td>0.3332</td>
<td>57</td>
<td>2312</td>
<td>0.44</td>
</tr>
<tr>
<td>25.0</td>
<td>0.40484</td>
<td>0.3780</td>
<td>0</td>
<td>2702</td>
<td>0.49</td>
</tr>
<tr>
<td>30.0</td>
<td>0.46180</td>
<td>0.4040</td>
<td>3</td>
<td>2872</td>
<td>0.45</td>
</tr>
<tr>
<td>50.0</td>
<td>0.60461</td>
<td>0.4455</td>
<td>0</td>
<td>3313</td>
<td>0.65</td>
</tr>
<tr>
<td>75.0</td>
<td>0.69751</td>
<td>0.4572</td>
<td>0</td>
<td>3493</td>
<td>0.65</td>
</tr>
<tr>
<td>100.0</td>
<td>0.75192</td>
<td>0.4720</td>
<td>0</td>
<td>4298</td>
<td>0.56</td>
</tr>
<tr>
<td>250.0</td>
<td>0.87184</td>
<td>0.4986</td>
<td>0</td>
<td>5043</td>
<td>0.54</td>
</tr>
<tr>
<td>500.0</td>
<td>0.92738</td>
<td>0.5027</td>
<td>0</td>
<td>5071</td>
<td>0.54</td>
</tr>
<tr>
<td>750.0</td>
<td>0.94919</td>
<td>0.5019</td>
<td>0</td>
<td>5111</td>
<td>0.57</td>
</tr>
</tbody>
</table>
Interestingly, the marginal frequencies were also sensitive to different temperatures. First of all, when temperature was high enough, all frequencies were close to 0.5. Secondly, frequencies of important variables tended to decrease when temperature increased, while others tended to increase. Figure 5.4 shows the movement of frequencies along log temperature. The solid lines stands for the marginal frequencies of the true predicting variables, while the dashed line suggests otherwise. The temperature starts from 5, since the process is not ergodic lower than it. One interpretation is that, by lemma 4.6, some local minimums (including the best model) appear more often when temperature decreases. Since these minimums usually contain the important variables, the frequencies of these variables also increases.
Procedure summary  A routine procedure for FTS can be summarized as follows.

- Firstly, we run small test samples on several temperature settings to get an estimation of average odds. A size of $4p$ would be fine.

- Secondly, based on the previous estimation, we choose a temperature $T$ which would give us an average odds around 0.05 to 0.15. Then we run simulation with this temperature setting for long chain.

- Exam the number of appearance of best models, see whether it decrease as BIC increases. Then check the marginal frequencies of each variable.

- If there’s some suspicious variables, i.e. the marginal frequencies is high, but not much higher than others. We can run a new simulation with a much lower temperature to see whether this frequencies remain high or even grows higher.

![Marginal frequencies of variables](image)

Figure 5.5: Convergence of Marginal frequencies when $T = 8$

For example, we choose temperature $T = 8$. Figure 5.5 illustrates how marginal frequencies changes when sample size grows larger. The true variables $x_5, x_{10}, x_{15}$ have obvious high frequency. The frequency of interfering variable $x_3$ is pretty high at the beginning, but finally falls. With the
right choice of temperature, a 1000-chain is already good enough to dig out important information. In sum, the Gibbs sampler does a very good job selecting the important variables here.

### 5.1.4 Cooling schedules Search

Based on previous analysis and study on temperature and average odds. We set the initial temperature $T_0 = 200$ where the average odds is around 0.8, and the terminal temperature $T_f = 1$ where the average odds is less than 0.0001. Furthermore, the period length $n = 60$, where it could be used to generate 3 valid samples in FTS.

Figure 5.6 demonstrates how CSS searches local minima. The figures both records the movement of the BIC values of models in the same part of the chain. The solid line is the BIC values, while the gray dots in the upper figure is the temperature. There are a little more than 3 periods in the figures, and in each period, the temperature drops fairly quickly and then gradually sinks to the terminal temperature $T = 1$. At the beginning, when the temperature is very low, the BIC values goes down to some local minimum. At the beginning of another period, the temperature is high, and the process jumps out of that local minimum. Then the process gradually moves down and search for another local minimum, when temperature becomes low again. In the lower figure, the solid line is still the same BIC chain, but the gray dots are the $O^*$, the odds of choosing the worse model instead of the better one in each step, here. $O^*$ has the similar periodic property, and moves in the same direction as the temperature does, but with more variance.

According to Proposition 4.5, each period of CSS forms a time-homogeneous Markov chain. Therefore, we only studies models appeared at the end the periods. The result is surprising good (table 5.4). The true model appeared in the end of the second period, which is the 121$^{th}$ step of the chain. Further more, the true model appears most often among the outputs. What worth
notice is that model \((x_5, x_{10}, x_{15}, x_{20}, x_3)\), model \((x_{10}, x_{15}, x_{20})\) don’t appear often considering their high BIC ranking. This is because they are in the neighbourhood of the first two models, and CSS has a large preference on local minima.

<table>
<thead>
<tr>
<th>Model</th>
<th>BIC rank</th>
<th>Frequency</th>
<th>First appearance</th>
</tr>
</thead>
<tbody>
<tr>
<td>(x_5, x_{10}, x_{15})</td>
<td>1</td>
<td>0.541</td>
<td>121</td>
</tr>
<tr>
<td>(x_3, x_{10}, x_{15})</td>
<td>3</td>
<td>0.388</td>
<td>61</td>
</tr>
<tr>
<td>(x_5, x_{10}, x_{15}, x_{20})</td>
<td>2</td>
<td>0.035</td>
<td>3241</td>
</tr>
<tr>
<td>(x_2, x_3, x_{10}, x_{15})</td>
<td>46</td>
<td>0.006</td>
<td>3541</td>
</tr>
<tr>
<td>(x_1, x_5, x_{10}, x_{15})</td>
<td>23</td>
<td>0.006</td>
<td>9601</td>
</tr>
<tr>
<td>(x_3, x_5, x_{10}, x_{15})</td>
<td>30</td>
<td>0.006</td>
<td>7261</td>
</tr>
<tr>
<td>(x_3, x_8, x_{10}, x_{15})</td>
<td>29</td>
<td>0.006</td>
<td>9001</td>
</tr>
<tr>
<td>(x_5, x_8, x_{10}, x_{15})</td>
<td>13</td>
<td>0.006</td>
<td>8701</td>
</tr>
<tr>
<td>(x_3, x_{10}, x_{15}, x_{20})</td>
<td>4</td>
<td>0.006</td>
<td>361</td>
</tr>
</tbody>
</table>

Table 5.4: Best models in CSS with \(N = 10000\)

<table>
<thead>
<tr>
<th>Variables</th>
<th>(x_{10})</th>
<th>(x_{15})</th>
<th>(x_5)</th>
<th>(x_3)</th>
<th>(x_4)</th>
<th>(x_2)</th>
<th>(x_1)</th>
<th>(x_8)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td>0.99</td>
<td>0.97</td>
<td>0.6</td>
<td>0.51</td>
<td>0.29</td>
<td>0.29</td>
<td>0.26</td>
<td>0.25</td>
</tr>
</tbody>
</table>

Table 5.5: Marginal Frequencies in CSS

5.2 Communities and Crime Data Set

5.2.1 Data description

We did some data cleaning, i.e., removing uncorrelated variables and missing data. Finally, there are 101 variables with 1993 entities in the final data warehouse. Overall speaking, this is a data mining regression problem. Since variables of from facts of communities, we find some strong connections between variables. In other words, the best model with lowest BIC is not that meaningful after all. We expect to find some important variables which is related to crime rate for further sociology studies.

5.2.2 Fixed temperature search

Following the procedure summarized previously, Average Odds under different temperatures based on 1000-chains is calculated as listed in TABLE 5.6. We find the temperatures between 0.5 and 1 maybe a sweet spot for FTS, since the Average Odds are around 0.1.

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Average Odds</th>
<th>Temperature</th>
<th>Average Odds</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>9.13e-7</td>
<td>2.50</td>
<td>0.293</td>
</tr>
<tr>
<td>0.05</td>
<td>1.70e-3</td>
<td>3.00</td>
<td>0.342</td>
</tr>
<tr>
<td>0.50</td>
<td>6.14e-2</td>
<td>5.00</td>
<td>0.486</td>
</tr>
<tr>
<td>1.00</td>
<td>0.125</td>
<td>7.00</td>
<td>0.582</td>
</tr>
<tr>
<td>1.50</td>
<td>0.184</td>
<td>10.00</td>
<td>0.674</td>
</tr>
<tr>
<td>2.00</td>
<td>0.243</td>
<td>20.00</td>
<td>0.813</td>
</tr>
</tbody>
</table>

Table 5.6: Average Odds under temperatures on 1000-chain

A 10,000-chain is simulated under temperatures $T = 1$. Since there are $p = 101$ candidate predicting variables, there are less then 100 valid Markov chain samples. With the same number of regressions, the number of valid samples are proportional decreasing when $p$ increases. But on the other hand, the number of states in the model space increases exponentially, which means the ratio of valid samples to model space decreases exponential as well. This is why the standard MCMC model selection gradually loses its power when $p$ is very big.
5.2. COMMUNITIES AND CRIME DATA SET

Table 5.7: 10 Best Models in FTS with temperature $T = 1$

<table>
<thead>
<tr>
<th>BIC</th>
<th>Model</th>
<th># Appearance</th>
</tr>
</thead>
<tbody>
<tr>
<td>46.36861</td>
<td>$x_{16}, x_{34}, x_{45}, x_{92}$</td>
<td>3</td>
</tr>
<tr>
<td>46.7601</td>
<td>$x_{16}, x_{34}, x_{42}, x_{45}, x_{92}$</td>
<td>6</td>
</tr>
<tr>
<td>46.95732</td>
<td>$x_{16}, x_{34}, x_{39}, x_{45}, x_{92}$</td>
<td>4</td>
</tr>
<tr>
<td>47.02452</td>
<td>$x_{16}, x_{34}, x_{41}, x_{45}, x_{70}, x_{92}$</td>
<td>35</td>
</tr>
<tr>
<td>47.3232</td>
<td>$x_{16}, x_{34}, x_{42}, x_{45}, x_{48}, x_{92}$</td>
<td>1</td>
</tr>
<tr>
<td>47.54892</td>
<td>$x_{16}, x_{41}, x_{45}, x_{48}, x_{70}, x_{92}$</td>
<td>5</td>
</tr>
<tr>
<td>47.65738</td>
<td>$x_{16}, x_{34}, x_{41}, x_{45}, x_{48}, x_{70}, x_{92}$</td>
<td>2</td>
</tr>
<tr>
<td>47.86762</td>
<td>$x_{16}, x_{39}, x_{45}, x_{70}, x_{85}, x_{87}$</td>
<td>2</td>
</tr>
<tr>
<td>47.92093</td>
<td>$x_{16}, x_{45}, x_{56}, x_{70}, x_{73}, x_{92}$</td>
<td>15</td>
</tr>
<tr>
<td>48.16035</td>
<td>$x_{16}, x_{34}, x_{35}, x_{39}, x_{45}, x_{92}$</td>
<td>66</td>
</tr>
</tbody>
</table>

Table 5.7 lists 10 best models in terms of BIC with their number of appearance. All the BIC values are pretty close, and the number of appearance is not consistent with Gibbs distribution, which means the process is far from its equilibrium. As a result, it is not meaningful to search models in this sample by their BIC values at all.

On the other hand, searching important variables by their marginal frequency is still feasible. Table 5.8 shows the marginal frequencies of 8 variables under two temperature settings. It’s obvious that $x_{45}, x_{16}$ are outstanding variables. Their marginal frequencies are high and increases as temperature decreases. On the other hand, other variables' frequencies are always less than 0.5 suggesting much less important. Therefore, searching models by predictors' marginal frequency is more robust to large $p$.

<table>
<thead>
<tr>
<th>T</th>
<th>AR</th>
<th>AO</th>
<th>Min. BIC</th>
<th>$x_{45}$</th>
<th>$x_{16}$</th>
<th>$x_{42}$</th>
<th>$x_{48}$</th>
<th>$x_{73}$</th>
<th>$x_{70}$</th>
<th>$x_{44}$</th>
<th>$x_{75}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.139</td>
<td>0.125</td>
<td>46.37</td>
<td>0.71</td>
<td>0.6</td>
<td>0.46</td>
<td>0.45</td>
<td>0.44</td>
<td>0.43</td>
<td>0.41</td>
<td>0.4</td>
</tr>
<tr>
<td>0.5</td>
<td>0.070</td>
<td>0.061</td>
<td>46.29</td>
<td>0.93</td>
<td>0.83</td>
<td>0.38</td>
<td>0.25</td>
<td>0.33</td>
<td>0.4</td>
<td>0.15</td>
<td>0.19</td>
</tr>
</tbody>
</table>

Table 5.8: Marginal frequencies in FTSs
5.2.3 Cooling Schedules Search

A CSS is ran under the setting: \( T_0 = 2, T_f = 0.0001, n = 600 \). There are 3 models with lower BIC than those in FTS found in CSS. Furthermore, their number of appearance is consistent with underlying Gibbs distribution, i.e. better model appears more often. What is worth noticing is that \( x_{45}, x_{16} \) appeared in almost all lowest BIC models, which is consistent with the marginal frequency search result in FTS. In this dataset, CSS, as an aggressive method searching local minima did its job.

As the result of the model selection, \( x_{45} \), “percentage of kids in family housing with two parents”, and \( x_{16} \), “households with investment/rent income in 1989”, are the two important variables we found negatively related to community crime rate. In a short summary, our diagnostic toolbox have brought models with lower BIC values than the standard MCMC model selection would do. And more evidence about those two important variables are collected at an acceptable cost of addition computation.

<table>
<thead>
<tr>
<th>BIC</th>
<th>Model</th>
<th># appearance</th>
</tr>
</thead>
<tbody>
<tr>
<td>46.27614</td>
<td>( x_{16}, x_{45}, x_{44}, x_{92} )</td>
<td>2</td>
</tr>
<tr>
<td>46.28978</td>
<td>( x_{16}, x_{45}, x_{92} )</td>
<td>2</td>
</tr>
<tr>
<td>46.34675</td>
<td>( x_{16}, x_{45}, x_{82} )</td>
<td>1</td>
</tr>
<tr>
<td>46.62806</td>
<td>( x_{16}, x_{41}, x_{45}, x_{87}, x_{92} )</td>
<td>1</td>
</tr>
<tr>
<td>46.65254</td>
<td>( x_{16}, x_{42}, x_{45}, x_{87} )</td>
<td>1</td>
</tr>
<tr>
<td>46.71364</td>
<td>( x_{16}, x_{42}, x_{45}, x_{85} )</td>
<td>1</td>
</tr>
<tr>
<td>46.78293</td>
<td>( x_{16}, x_{43}, x_{45}, x_{70}, x_{87} )</td>
<td>1</td>
</tr>
<tr>
<td>47.71161</td>
<td>( x_{4}, x_{16}, x_{42}, x_{44}, x_{70} )</td>
<td>1</td>
</tr>
<tr>
<td>47.77711</td>
<td>( x_{16}, x_{44}, x_{45}, x_{92} )</td>
<td>1</td>
</tr>
<tr>
<td>47.84254</td>
<td>( x_{4}, x_{16}, x_{39}, x_{44}, x_{70} )</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 5.9: 10 Best models in CSS

<table>
<thead>
<tr>
<th>( x_{16} )</th>
<th>( x_{45} )</th>
<th>( x_{44} )</th>
<th>( x_{92} )</th>
<th>( x_{87} )</th>
<th>( x_{42} )</th>
<th>( x_{4} )</th>
<th>( x_{41} )</th>
<th>( x_{39} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.69</td>
<td>0.5</td>
<td>0.5</td>
<td>0.47</td>
<td>0.44</td>
<td>0.42</td>
<td>0.4</td>
<td>0.36</td>
<td>0.33</td>
</tr>
</tbody>
</table>

Table 5.10: Marginal Frequency of selected variables in CSS
5.3. Spam Data

5.3.1 Data description

This data set was in a study to screen emails for spam. It can be obtained from [ftp.ics.uci.edu](ftp.ics.uci.edu) donated by George Forman From Hewlett-Packard laboratories, Palo Alto, California. There are 4601 email messages with 57 explanatory variables all treated as continuous variables here.

- 48 quantitative predictors as the percentage of words in the email that match a given word.
- 6 quantitative predictors as the percentage of characters in the email that match a given character.
- The average length of uninterrupted sequences of capital letters: CAPAVE.
- The length of the longest uninterrupted sequence of capital letters: CAPMAX.
- The sum of the length of uninterrupted sequences
This data set is widely tested for many data mining methods in the textbook [Hastie et al., 2005] as an evaluation tool. In order to be comparable, we do the same log-transform for each variable \( \log(x + 0.1) \) to adjust the thick tail of the distributions. And for the same reason, smoothing curve is applied in generalized additive models instead of local polynomials.

5.3.2 Model comparison

In terms of prediction, the model selected by AIC usually outperforms the one selected by BIC. AIC has less penalty on degree of freedoms than BIC, which tends to include more predicting variables. This may explains why AIC is preferred searching models for prediction. Therefore, in this section we use AIC as our model selection criterion. In addition, in [Hastie et al., 2005]’s textbook, they used smoothing splines to fit the GAM. In order to be comparable and show improvement after deployed our MCMC model selection procedure, for this dataset, we use smoothing splines instead of local polynomial fitting.

Using our MCMC model selection diagnostic toolbox, we present two models selected by FTS and CSS respectively. First of all, we ran some short sample chains with different temperature settings to have a general idea how does the Gibbs distribution defined on model spaces react to different temperatures. Then we use Average Odds as an reference guideline to choose the temperature for FTS, and set up the cooling schedule for CSS.

For FTS, we chose temperature \( T = 1.5 \). After obtained a 10,000 chain, we formed our model \( \alpha_{FTS} \) by selecting predicting variables with marginal frequencies greater than 0.6. For CSS, we set the cooling schedule (4.25) with the following parameters. Initial temperature \( T_0 = 200 \), terminal temperature \( T_n = 0.001 \), and a single period is of length \( n = 285 \). Again, we generated a 10,000-chain with CSS. We pick the model with lowest AIC value and call it \( \alpha_{CSS} \).
A comparison of misclassification error on this dataset between our models $\alpha_{\text{FTS}}$, $\alpha_{\text{CSS}}$ and models mentioned in [Hastie et al. 2005] is listed in Table 5.11. We used same cross-validation method to calculated the misclassification error of $\alpha_{\text{FTS}}$, $\alpha_{\text{CSS}}$ and directly quote the result from the textbook.

<table>
<thead>
<tr>
<th>Model</th>
<th>Misclassification error</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha_{\text{FTS}}$</td>
<td>4.92 %</td>
</tr>
<tr>
<td>$\alpha_{\text{CSS}}$</td>
<td>5.05 %</td>
</tr>
<tr>
<td>Full GAM</td>
<td>5.5 %</td>
</tr>
<tr>
<td>Linear logistic regression</td>
<td>7.6 %</td>
</tr>
<tr>
<td>Decision Tree</td>
<td>9.3%</td>
</tr>
<tr>
<td>Multivariate Adaptive Regression Splines</td>
<td>5.5 %</td>
</tr>
<tr>
<td>Gradient boosting method</td>
<td>4.5 %</td>
</tr>
<tr>
<td>Random forest classifier</td>
<td>4.88 %</td>
</tr>
<tr>
<td>Bagging</td>
<td>5.4%</td>
</tr>
</tbody>
</table>

Table 5.11: Comparisons between different machine learning methods

There is an observable difference (about 0.5%) between $\alpha_{\text{FTS}}$, $\alpha_{\text{CSS}}$ and the full GAM including all possible predicting variables. It shows there is improvement of full GAM can be made using our model selection. Gradient boosting method has much lower misclassification error, however, $\alpha_{\text{FTS}}$, $\alpha_{\text{CSS}}$ have much simpler structures and are much easier for human to read, i.e. to reveal which predicting variables are more important than others. Meanwhile, comparing to another human friendly method, decision tree method, $\alpha_{\text{FTS}}$, $\alpha_{\text{CSS}}$ are much better in prediction. Although random forest as a mixture of decision trees has similar prediction ability, it is too sophisticated to tell the relationships between predicting variables and responding variable in a simple form.

In sum, GAM selected by MCMC model selection and simulated annealing is comparative with the modern machine learning methods in terms of prediction. In addition it has a very simple structure, which can tell how the prediction variables and the responding variables are related, and how important those predicting variables are respectively. Furthermore, our model diagnostic toolbox, MCMC model selection with simulated annealing, can be
further implemented into many other supervised machine learning methods. As long as they are sensitive to the set of predicting variables, such as random tree.
Figure 5.2: I-chart convergence test
Figure 5.3: Distribution of best 50 models in FTSs
Figure 5.6: Movement BIC values in a cooling schedule.
Chapter 6

Conclusion

In this thesis, we presented a diagnostic toolbox for MCMC model selection using simulated annealing techniques on generalized additive model with local polynomial fitting. This toolkit can, in addition to select models both competitive in prediction among other modern machine learning methods, also provide more human friendly information about the dataset.

MCMC model selection on GAM is capable of not only selecting models but collecting more information of relationships between the predicting variables and responding variables, i.e. their relationships through the smoothing functions, how important each predicting variable is through the marginal frequencies. Our diagnostic toolkit presented two searching methods, Fixed Temperature Search (FTS) & Cooling Schedules Search (CSS). FTS accelerates the standard MCMC model selection procedure by reshaping the stationary distribution on model space, while CSS aggressively searches local minima of information criterion values. Moreover, we also observe a negative relationship between the marginal frequencies of the important predicting variables and temperature; and a positive relationship between the noise variables’ and the temperature. This interesting phenomenon can be utilized to confirm the selected important predicting variables in FTS, which is a feature not provided by using the standard MCMC model selection procedures.
This thesis is more of a working framework with some heuristic thinking than a systematic study. Plenty of work can be done related to this toolkit, i.e. the bias and variance study of Average Odds, whether there is a universal relationship between marginal frequencies and temperature, an automatic procedure to establish a cooling schedule, etc. Furthermore, the toolkit can be implemented into other supervising machine learning methods which are sensitive to the sets of predicting variables.
Bibliography


