

25861/H/06



TESIS

**ANALISIS BAYESIAN MARKOV CHAIN MONTE CARLO
PADA PEMODELAN *MIXTURE* NORMAL
DENGAN BANYAK KOMPONEN TIDAK DIKETAHUI
(STUDI KASUS: TINGKAT INFLASI DI INDONESIA)**

Oleh :

**DWI MARTIANA WATI
NRP. 1303 201 010**



RTSt
519.542
Wat
9-1

2006

PENGUNJUKAN	
ITS	
Tgl. Terima	11-7-06
Terima Oleh	H
No. Agenda Prp.	925165

**PROGRAM MAGISTER
JURUSAN STATISTIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2006**

**ANALISIS BAYESIAN MARKOV CHAIN MONTE CARLO
PADA PEMODELAN *MIXTURE* NORMAL
DENGAN BANYAK KOMPONEN TIDAK DIKETAHUI
(STUDI KASUS: TINGKAT INFLASI DI INDONESIA)**

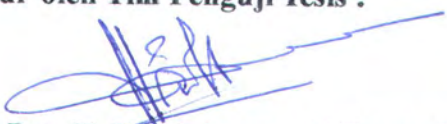
**Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Sains (M.Si.)
di
Institut Teknologi Sepuluh Nopember Surabaya**

Oleh:

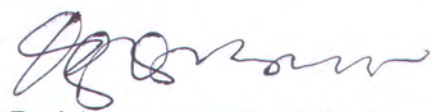
**Dwi Martiana Wati
NRP. 1303 201 010**

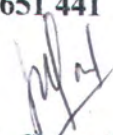
Disetujui oleh Tim Penguji Tesis :

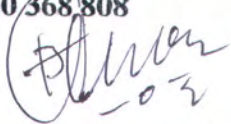
**Tanggal Ujian:
23 Februari 2006**

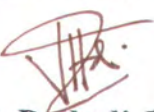
1. 
Prof. Drs. H. Nur Iriawan, M.Ikom., Ph.D.
NIP. 131 782 011

**Periode Wisuda:
September 2006**

2. 
Ir. Dwi Atmono A. W., M.Ikom.
NIP. 131 651 441

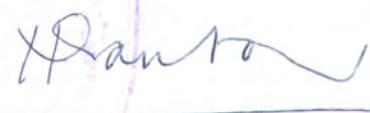
3. 
Prof. Dra. Susanti Linuwih, M.Stats., Ph.D.
NIP. 130 368 808

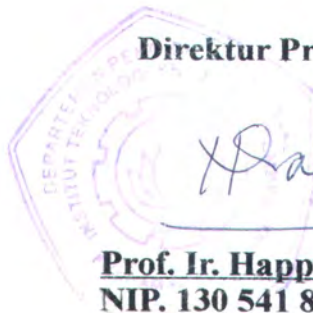
4. 
Dr. Drs. I Nyoman Budiantara, M.S.
NIP. 131 843 382

5. 
Dr. Purnadi, M.Sc.
NIP. 131 652 051

6. 
Drs. I Nyoman Latra, M.S.
NIP. 130 701 283

Direktur Program Pascasarjana


Prof. Ir. Happy Ratna S., M.Sc., Ph.D.
NIP. 130 541 829



**ANALISIS BAYESIAN MARKOV CHAIN MONTE CARLO
PADA PEMODELAN MIXTURE NORMAL
DENGAN BANYAK KOMPONEN TIDAK DIKETAHUI
(STUDI KASUS: TINGKAT INFLASI DI INDONESIA)**

Oleh : Dwi Martiana Wati (1303 201 010)
Pembimbing : 1. Prof. Drs. H. Nur Iriawan, M.Ikom., Ph.D.
2. Ir. Dwi Atmono A.W., M.Ikom.

ABSTRAK

Inflasi merupakan salah satu indikator yang sering digunakan untuk menggambarkan kondisi moneter suatu negara. Adanya perubahan kondisi suatu negara dari waktu ke waktu menyebabkan pola data tingkat inflasi di negara tersebut membentuk kelompok-kelompok data yang mempunyai pola yang berbeda-beda. Pemodelan pada data jenis ini dilakukan dengan menggunakan model *mixture* yang dapat mengakomodasi variabilitas data yang kurang dapat terwakili apabila dimodelkan secara unimodal.

McLachlan dan Basford (1988) menyatakan model *mixture* secara umum sebagai berikut:

$$f_{\text{mix}}(x|w, \theta) = w_1 f_1(x|\theta_1) + w_2 f_2(x|\theta_2) + \dots + w_k f_k(x|\theta_k) \text{ dengan } \sum_{j=1}^k w_j = 1.$$

Pada pemodelan *mixture* dengan banyak komponen tidak diketahui, estimasi parameter model *mixture* dilakukan dengan dua cara, yaitu: pertama, banyak komponen *mixture* ditentukan berdasarkan identifikasi data, sehingga estimasinya dilakukan dengan menggunakan Gibbs Sampler; kedua, banyak komponen *mixture* dianggap tetap tidak diketahui, sehingga estimasinya dilakukan dengan menggunakan Reversible Jump MCMC (RJMCMC).

Pada kasus tingkat inflasi di Indonesia menunjukkan adanya kecenderungan data bersifat multimodal. Estimasi parameter model *mixture* dengan menggunakan metode Gibbs Sampler menghasilkan tiga model *mixture* dengan banyak komponen dua, tiga dan empat. Sedangkan estimasi parameter model *mixture* dengan menggunakan metode RJMCMC menghasilkan model *mixture* dengan banyak komponen satu hingga sepuluh. Semua model *mixture* yang dihasilkan dari setiap metode tersebut dibandingkan untuk mendapatkan model *mixture* yang paling sesuai dengan data tingkat inflasi. Pemilihan model ini dilakukan dengan menggunakan Struktur Perkalian Distribusi (SPD) dan kriteria Bayes faktor.

Hasil perbandingan menunjukkan bahwa perbedaan banyak komponen dalam model *mixture* tersebut tidak signifikan. Namun demikian hasil estimasi model tersebut akan bersifat interpretatif jika disesuaikan dengan kategori inflasi berdasarkan penjelasan Boediono (1982). Dengan demikian model *mixture* yang disarankan untuk data tingkat inflasi adalah model *mixture* dengan banyak komponen dua. Jadi model *mixture* yang disarankan apabila menggunakan Gibbs Sampler adalah:

$$X \sim 0,8949 N(6,839;1,233) + 0,1051 N(17,65;0,45)$$

Sedangkan model *mixture* yang disarankan apabila menggunakan RJMCMC adalah:

$$X \sim 0,89441 N(0,83646;1,2624) + 0,10559 N(17,77451;1,0497)$$

Kata kunci : Model *mixture* normal, Bayesian, MCMC, Gibbs Sampler, RJMCMC, SPD, kriteria Bayes factor

**BAYESIAN MARKOV CHAIN MONTE CARLO ANALYSIS
ON NORMAL MIXTURE MODELLING
WITH AN UNKNOWN NUMBER OF COMPONENTS
(CASE STUDY: INFLATION RATE IN INDONESIA)**

By : Dwi Martiana Wati (1303 201 010)
Supervisors : 1. Prof. Drs. H. Nur Iriawan, M.Ikom., Ph.D.
2. Ir. Dwi Atmono A.W., M.Ikom.

ABSTRACT

Inflation is one of indicators widely used to describe the monetary condition on a country. A conditional changing on the country caused the inflation rate pattern formed several groups of the data. Data consist of several groups can be modeled using mixture model that able to accommodate the variability of data which is less represented if modeled using unimodal.

McLachlan and Basford (1988) define mixture model as:

$$f_{mix}(x|w, \theta) = w_1 f_1(x|\theta_1) + w_2 f_2(x|\theta_2) + \dots + w_k f_k(x|\theta_k) \text{ where } \sum_{j=1}^k w_j = 1 .$$

Parameter estimation using mixture model with an unknown number of components can be done in two ways: first, using Gibbs Sampler method, by specified the number of components based on data identification; second, using Reversible Jump MCMC (RJCMCMC), when the number of components is unknown.

The inflation rate in Indonesia attends that the data is multimodal. Parameter estimation of the mixture model using Gibbs Sampler method obtained that the number of components are two, three or four. While the parameter estimation using RJCMCMC obtained that the number of components are one to ten. All the mixture models obtained from each method are compared to obtain the most representative one to the inflation rate using Multiplicative Joint Density (MJD) and Bayes factor criterion.

The result of computation shows that there are significantly no different between one model to others. It means that all models can be used to represent the inflation rate. But theoretically, the most representative one is mixture model with two number of components (Boediono, 1982). Then the mixture model using Gibbs Sampler method is:

$$X \sim 0,8949 N(6,839;1,233) + 0,1051 N(17,65;0,45)$$

While the mixture model using RJCMCMC is:

$$X \sim 0,89441 N(0,83646;1,2624) + 0,10559 N(17,77451;1,0497)$$

Key words: Normal mixture model, Bayesian, MCMC, Gibbs Sampler, RJCMCMC, MJD, Bayes factor criterion

KATA PENGANTAR

Alhamdulillah, puji syukur hanya kepada Allah SWT yang telah memberikan segala kurnia, rahmat dan hidayahNya sehingga tesis ini dapat terselesaikan.

Penulis juga mengucapkan terima kasih kepada:

1. Pimpinan Universitas Muhammadiyah Jember atas kesempatan yang diberikan untuk mengikuti program S2 di ITS Surabaya.
2. Prof. Drs. H. Nur Iriawan, M.Ikom., Ph.D. atas semua bimbingan yang penuh dengan kesabaran dan pelajaran kehidupan yang banyak.
3. Ir. Dwi Atmono A.W., M.Ikom. atas segala kesabaran selama proses bimbingan.
4. Dr. Drs. I Nyoman Budiantara, MS atas semua budi baiknya.
5. Kel. Bpk. Indra Nur Fauzi atas semua bantuan dan kemudahan selama proses penyusunan tesis.
6. Tim Penguji atas semua masukan dan saran.
7. Semua pihak yang tidak dapat disebutkan satu per satu yang telah membantu selama proses penyusunan tesis ini.

Teristimewa untuk suamiku, Yus Sudarso, dan anakku, Kamilia Azka Zayyana atas segala pengorbanan dan dukungan serta kesabaran yang tiada terkira, serta orang tua dan kakakku atas segala dukungannya. Tesis ini saya dedikasikan untuk mereka. Semoga pekerjaan ini bisa menjadi pendorong semangat bagi kita untuk terus maju dan berjuang di jalan-Nya.

Surabaya, Februari 2006

Penulis

DAFTAR ISI

	Halaman
ABSTRAK	i
KATA PENGANTAR	iii
DAFTAR ISI	iv
DAFTAR GAMBAR	vii
DAFTAR TABEL	viii
DAFTAR LAMPIRAN	ix
BAB I. PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Permasalahan	2
1.3. Tujuan Penelitian	3
1.4. Batasan Masalah	3
1.5. Manfaat Penelitian	3
BAB II. TINJAUAN PUSTAKA	4
2.1. Metode Bayesian	4
2.2. Model <i>Mixture</i>	5
2.3. Distribusi <i>Mixture</i> Normal	6
2.4. Identifikasi Data dengan Metode Histogram	7
2.5. Analisis Bayesian pada Pemodelan <i>Mixture</i>	8
2.5.1. Distribusi Prior	10
2.5.2. Fungsi Likelihood pada Model <i>Mixture</i>	11

2.5.3.	Distribusi Bersyarat Penuh	12
2.5.4.	Struktur Hirarki	12
2.6.	Analisis Bayesian pada Pemodelan <i>Mixture</i> Normal Univariat	13
2.6.1.	Penentuan Distribusi Prior	14
2.6.2.	Pembentukan Struktur Hirarki	15
2.6.3.	Distribusi Bersyarat Penuh	16
2.6.3.1.	Distribusi Bersyarat Penuh untuk Parameter Pembobot	16
2.6.3.2.	Distribusi Bersyarat Penuh untuk Parameter Lokasi	17
2.6.3.3.	Distribusi Bersyarat Penuh untuk Parameter Penyebaran	19
2.6.3.4.	Distribusi Bersyarat Penuh untuk Parameter Alokasi	20
2.6.3.5.	Distribusi Bersyarat Penuh untuk Hiperparameter β	20
2.7.	Markov Chain Monte Carlo	21
2.7.1.	Integrasi Monte Carlo	22
2.7.2.	Rantai Markov	22
2.7.3.	Gibbs Sampler	24
2.7.4.	Metropolis-Hastings.....	28
2.7.5.	Reversible Jump.....	30
2.7.6.	Pemilihan Model Terbaik dengan Struktur Perkalian Distribusi (SPD)	32
2.7.7.	Bayes Faktor	37
BAB III. METODOLOGI PENELITIAN		41
3.1.	Bahan Penelitian	41

3.2. Paket Program	41
3.3. Langkah-langkah Penelitian	42
3.4. Diagram Alir Estimasi Distribusi <i>Mixture</i> Normal	46
BAB IV. HASIL ANALISIS DAN PEMBAHASAN	47
4.1. Identifikasi dan Deskripsi Data	47
4.2. Analisis Bayesian pada Data	49
4.2.1. Banyak Komponen <i>Mixture</i> Ditentukan	50
4.2.1.1. Pembuatan Doodle	50
4.2.1.2. Program WinBUGS	51
4.2.1.3. Estimasi Komponen-komponen <i>Mixture</i>	53
4.2.1.4. Pemilihan Model Terbaik Menggunakan SPD ...	59
4.2.1. Banyak Komponen <i>Mixture</i> Tidak Ditentukan	65
4.2.1.1. Algoritma RJMCMC	65
4.2.1.2. Hasil Iterasi dengan RJMCMC	66
4.3. Pembahasan Pemilihan Model Terbaik	69
BAB V. KESIMPULAN DAN SARAN	71
5.1. Kesimpulan	71
5.2. Saran	73
DAFTAR PUSTAKA	74
LAMPIRAN	76

DAFTAR GAMBAR

Nomor	Judul	Halaman
2.1	Directed Acyclic Graph (DAG) untuk Model <i>Mixture</i> Secara Umum	13
2.2	DAG untuk estimasi distribusi <i>mixture</i> normal dengan banyak komponen k tidak diketahui	15
3.1	Diagram alir estimasi distribusi <i>mixture</i> normal univariat	46
4.1	Histogram data tingkat inflasi	47
4.2	Plot probabilitas normal data tingkat inflasi	49
4.3	Doodle model <i>mixture</i> normal dengan dua komponen	51
4.4	Plot estimasi densitas parameter model <i>mixture</i> normal univariat dengan dua komponen <i>mixture</i>	54
4.5	Plot distribusi <i>mixture</i> normal dengan dua komponen <i>mixture</i>	55
4.6	Plot distribusi <i>mixture</i> normal dengan tiga komponen <i>mixture</i>	58
4.7	Plot distribusi <i>mixture</i> normal dengan empat komponen <i>mixture</i>	58
4.8	Doodle SPD dengan tiga distribusi penyusun untuk data tingkat inflasi	60
4.9	Plot distribusi <i>mixture</i> normal dengan banyak komponen <i>mixture</i> dua dan tiga yang diestimasi dengan RJMCMC	68
4.10	Plot bersama dua densitas <i>mixture</i> dengan dua komponen yang diestimasi dengan Gibbs Sampler dan RJMCMC	70

DAFTAR TABEL

Nomor	Judul	Halaman
2.1	Interpretasi nilai Bayes faktor dalam pemilihan hipotesis	40
4.1	Hasil analisis deskriptif pada data tingkat inflasi	48
4.2	Banyak iterasi dari tiap model <i>mixture</i> normal dengan RJMCMC	66
4.3	Hasil penghitungan Bayes faktor dari model kedua terhadap model lainnya	67
4.4	Estimasi parameter model <i>mixture</i> normal dengan RJMCMC	68

DAFTAR LAMPIRAN

Nomor	Judul	Halaman
1	Data Tingkat Inflasi (dalam %)	76
2	Struktur doodle dan program model <i>mixture</i> data tingkat inflasi Dengan WinBUGS 1.4	77
3	Hasil iterasi program WinBUGS dengan 10000 iterasi	81
4	Inisial dan <i>output</i> program Nmix	84



BAB I

PENDAHULUAN

1.1. Latar Belakang

Salah satu peristiwa moneter yang sangat penting dan dijumpai hampir di semua negara adalah inflasi. Definisi inflasi adalah kecenderungan dari harga-harga untuk menaik secara umum dan terus menerus. Kenaikan harga dari satu atau dua barang saja tidak bisa disebut inflasi, kecuali jika kenaikan tersebut meluas dan mengakibatkan kenaikan sebagian besar harga barang-barang lainnya. Adapun kenaikan harga barang-barang yang disebabkan adanya kondisi musiman, misalnya menjelang hari-hari besar, juga tidak bisa disebut inflasi.

Kondisi inflasi yang senantiasa berubah dari waktu ke waktu membentuk kelompok-kelompok yang mempunyai pola yang berbeda-beda, dimana pola perubahan inflasi ini digambarkan melalui distribusi yang juga berbeda-beda. Adanya tuntutan untuk mengidentifikasi kenyataan ini agar dapat memberikan deskripsi yang lengkap dan nantinya dapat digunakan sebagai bahan masukan untuk analisis dalam pengambilan kebijakan, telah menjadi suatu hubungan yang bersinergi antara ilmu statistika dan ilmu ekonomi, khususnya di bidang moneter.

Perubahan inflasi yang membentuk kelompok-kelompok tersebut akan membentuk sebuah pola baru sebagai suatu *mixture*. Pemodelan pada data jenis ini dilakukan dengan menggunakan model *mixture* yang dapat mengakomodasi variabilitas data yang kurang dapat terwakili apabila dimodelkan secara unimodal. Kondisi

ini sejalan dengan perkembangan ilmu statistika yang menuju pola analisis *data driven*, dimana konsep analisis *data driven* mengutamakan macam dan sifat data sesuai dengan kondisi data sebenarnya.

Pemodelan dengan pendekatan Bayesian diunggulkan sebagai salah satu cara analisis *data driven*. Hal ini dikarenakan pendekatan Bayesian menggunakan perpaduan antara dua sisi informasi, yaitu data pengamatan yang digunakan sebagai penyusun *likelihood* dipadukan dengan data masa lalu sebagai informasi prior. Oleh karena itu pemodelan *mixture* pada data tingkat inflasi dilakukan dengan menggunakan analisis Bayesian yang didukung dengan penggunaan Markov Chain Monte Carlo (MCMC). Penggunaan MCMC ini didasarkan pada analisis yang mengacu pada konsep *data driven* akan mengakibatkan dimensi estimasi model yang dihasilkan dari pendekatan Bayesian semakin kompleks.

1.2. Permasalahan

Berdasarkan uraian latar belakang di atas, permasalahan yang diangkat dalam penelitian ini adalah:

1. Bagaimana cara mengidentifikasi pola data tingkat inflasi yang senantiasa berubah dari waktu ke waktu sehingga membentuk model *mixture normal* ?
2. Bagaimana cara menentukan banyaknya komponen penyusun *mixture* dalam model *mixture normal* ?
3. Bagaimana cara mengestimasi parameter model *mixture normal* ?
4. Bagaimana cara memilih model *mixture* yang paling sesuai untuk data tingkat inflasi?

1.3. Tujuan Penelitian

Sesuai dengan permasalahan di atas, maka tujuan yang hendak dicapai dari penelitian ini adalah:

1. Dapat mengidentifikasi adanya bentuk model *mixture* normal pada data tingkat inflasi.
2. Dapat menentukan banyaknya komponen penyusun *mixture* dalam model *mixture* normal.
3. Dapat mengestimasi parameter model *mixture* normal dengan menggunakan analisis Bayesian dan metode Markov Chain Monte Carlo.
4. Dapat menentukan model *mixture* normal terbaik dengan menggunakan Struktur Perkalian Distribusi dan kriteria Bayes faktor.

1.4. Batasan Masalah

Permasalahan yang dibahas pada tesis ini dibatasi pada pembahasan mengenai pemodelan *mixture* normal univariat, sedangkan pemodelan *mixture* untuk kasus bivariat dan multivariat tidak dibahas. Selain itu data yang digunakan adalah data tingkat inflasi di Indonesia yang dipublikasikan oleh Bank Indonesia.

1.5. Manfaat Penelitian

Hasil penelitian ini dapat dimanfaatkan untuk kajian keilmuan mengenai analisis Bayesian pada pemodelan *mixture* yang didukung dengan MCMC khususnya pada pemodelan *mixture* normal univariat. Sedangkan pada tataran praktis, hasil penelitian ini dapat digunakan sebagai salah satu alat dalam proses pengambilan keputusan atau penentuan kebijakan terkait dengan data yang dianalisis.

BAB II

TINJAUAN PUSTAKA

2.1. Metode Bayesian

Pemodelan dengan pendekatan Bayesian didasarkan pada model posterior yang diperoleh dari perpaduan antara dua informasi, yaitu data pengamatan yang digunakan sebagai penyusun fungsi *likelihood* dipadukan dengan data masa lalu sebagai informasi prior. Dimisalkan terdapat data pengamatan \mathbf{x} yang mempunyai fungsi padat peluang $f(\mathbf{x}|\theta)$, maka fungsi *likelihood* untuk θ adalah $L(\theta) = f(\mathbf{x}|\theta)$. Sedangkan informasi prior dari θ didasarkan pada distribusi θ , yaitu $p(\theta)$, sebelum melakukan pengambilan data \mathbf{x} . Dengan demikian penentuan distribusi posterior θ , yaitu $p(\theta|\mathbf{x})$, didasarkan pada aturan probabilitas dalam teorema Bayes berikut :

$$p(\theta|\mathbf{x}) = \frac{f(\mathbf{x}|\theta)p(\theta)}{f(\mathbf{x})} \quad (2.1)$$

dengan $f(\mathbf{x}) = E[f(\mathbf{x}|\theta)] = \begin{cases} \int_{x \in R} f(\mathbf{x}|\theta)f(\theta)d\theta & \text{jika } \theta \text{ kontinyu} \\ \sum_{x \in B} f(\mathbf{x}|\theta)p(\theta) & \text{jika } \theta \text{ diskrit} \end{cases}$

Oleh karena $f(\mathbf{x})$ berupa konstanta yang biasa disebut sebagai *normalized constant* (Gelman, Carlin, Stern dan Rubin, 1995), maka aturan probabilitas dalam teorema Bayes di atas dapat ditulis sebagai berikut :

$$p(\theta|\mathbf{x}) \propto f(\mathbf{x}|\theta) p(\theta) \quad (2.2)$$

Selanjutnya posterior di atas dinyatakan dalam bentuk yang lebih sederhana sebagai berikut :

$$\pi(\theta) \propto L(\theta) p(\theta) \quad (2.3)$$

dimana $L(\theta) = f(\mathbf{x}|\theta)$. Notasi $\pi(\bullet)$ digunakan untuk menyatakan posterior yang berbeda dengan prior. Dengan demikian persamaan (2.3) menunjukkan bahwa posterior proporsional terhadap perkalian *likelihood* dengan prior parameter model.

2.2. Model Mixture

Model *mixture* merupakan model yang khas. Kekhasan model ini ditunjukkan dengan adanya pola multimodal pada datanya. Dengan demikian model ini digunakan untuk memodelkan data yang diambil dari beberapa subpopulasi atau grup dimana setiap subpopulasi atau grup merupakan komponen penyusun *mixture* dengan proporsi yang bervariasi untuk setiap komponennya ((McLachlan dan Basford,1988) dan (Gelman *et al*, 1995)).

Dimisalkan terdapat data pengamatan $\mathbf{x}' = (x_1, \dots, x_n)$ yang diambil dari sebanyak k subpopulasi dan setiap subpopulasi diberi pembobot yang dinotasikan dengan w_j , sedemikian hingga $\sum_{j=1}^k w_j = 1$ dan $0 \leq w_j \leq 1, \forall j$; maka McLachlan dan Basford (1988) serta Gelman *et al* (1995) menyatakan model *mixture* dari data \mathbf{x} sebagai berikut :

$$f(\mathbf{x}, \boldsymbol{\theta}, \mathbf{w}) = \sum_{j=1}^k w_j f(\mathbf{x}|\theta_j) \quad (2.4)$$

dimana $f(\mathbf{x}|\theta_j)$ menotasikan fungsi densitas dari komponen penyusun *mixture* yang mempunyai parameter model θ_j , sedangkan θ dan \mathbf{w} adalah vektor parameter dengan masing-masing elemen adalah θ_j dan w_j . Adapun setiap parameter model θ_j juga bisa berupa vektor. Model *mixture* yang dinyatakan pada persamaan (2.4) berlaku untuk kasus model *mixture* dengan banyak komponen tertentu, yang disebut dengan *finite mixture model*. Iriawan (2001) dan Wati (2002) telah menggunakan model ini pada estimasi densitas model *mixture* normal univariat dan model *mixture* neo-normal univariat.

2.3. Distribusi *Mixture* Normal

Dimisalkan terdapat sebanyak k subpopulasi yang masing-masing subpopulasi berdistribusi normal, maka berdasarkan persamaan (2.4), densitas *mixture* normal dinyatakan sebagai berikut :

$$f_{\text{mix}}(\mathbf{x}|\mathbf{w}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2) = w_1 f_1(\mathbf{x}|\mu_1, \sigma_1^2) + w_2 f_2(\mathbf{x}|\mu_2, \sigma_2^2) + \dots + w_k f_k(\mathbf{x}|\mu_k, \sigma_k^2) \quad (2.5)$$

dengan densitas dari masing-masing subpopulasi dinyatakan sebagai berikut :

$$f_j(\mathbf{x}|\mu_j, \sigma_j^2) = \frac{1}{\sqrt{2\pi} \sigma_j} \exp\left\{-\frac{(\mathbf{x}-\mu_j)^2}{\sigma_j^2}\right\} \quad ; j = 1, 2, \dots, k \quad (2.6)$$

Jika banyaknya komponen *mixture* empat, maka distribusi *mixture* normal univariat dapat dinyatakan sebagai berikut:

$$\begin{aligned}
 f_{\text{mix}}(x | \mathbf{w}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2) = & w_1 \frac{1}{\sqrt{2\pi} \sigma_1} \exp \left\{ -\frac{(x - \mu_1)^2}{\sigma_1^2} \right\} + \\
 & w_2 \frac{1}{\sqrt{2\pi} \sigma_2} \exp \left\{ -\frac{(x - \mu_2)^2}{\sigma_2^2} \right\} + \\
 & w_3 \frac{1}{\sqrt{2\pi} \sigma_3} \exp \left\{ -\frac{(x - \mu_3)^2}{\sigma_3^2} \right\} + \\
 & w_4 \frac{1}{\sqrt{2\pi} \sigma_4} \exp \left\{ -\frac{(x - \mu_4)^2}{\sigma_4^2} \right\}
 \end{aligned}
 \tag{2.7}$$

2.4. Identifikasi Data dengan Metode Histogram

Metode histogram merupakan metode tertua dan paling sering digunakan dalam estimasi suatu distribusi data. Namun demikian metode ini masih sangat sederhana karena estimasi distribusinya hanya didasarkan pada perkiraan gambar dari histogram yang dibentuk. Silverman (1986) menjelaskan bahwa pembentukan histogram dari data pengamatan hanya memerlukan titik awal x_0 dan lebar kelas interval h , sehingga histogram yang dibentuk akan berada pada interval $[x_0 + mh, x_0 + (m+1)h]$ dengan m adalah bilangan cacah. Dengan demikian keberadaan kelas-kelas interval sangat dipengaruhi oleh penentuan nilai awal x_0 dan lebar kelas h .

Jika dimisalkan terdapat data pengamatan $\mathbf{x}' = (x_1, \dots, x_n)$ maka Silverman (1986) mendefinisikan fungsi densitas data yang menggunakan metode ini sebagai berikut :

$$\hat{f}(x) = \frac{1}{nh} \times p \quad (2.8)$$

dimana n menotasikan banyaknya data x_i dengan $i = 1, \dots, n$ dan p menotasikan banyaknya data x_i yang berada pada kelas yang sama dengan x .

Berdasarkan perumusan di atas, diketahui bahwa untuk membangun distribusi dengan menggunakan metode histogram sangat ditentukan oleh banyaknya data. Padahal banyaknya data pada tiap kelas interval sangat dipengaruhi oleh nilai awal x_0 dan lebar kelas h yang sudah ditentukan. Dengan demikian penentuan titik awal dan lebar kelas interval menjadi sangat berpengaruh.

Adanya ketergantungan pada penentuan titik awal dan lebar kelas interval ini dalam membentuk distribusi data merupakan kelemahan dari metode histogram. Namun demikian metode ini dapat digunakan untuk deteksi awal dari suatu distribusi data secara deskriptif. Iriawan (2001), Prihartanti (2002), Wati (2002) serta Suharjo (2003) telah menggunakan metode ini untuk mendeteksi adanya kecenderungan suatu data akan berdistribusi *mixture*. Oleh karena itu penelitian ini juga akan diawali dengan identifikasi data dengan menggunakan metode ini.

2.5. Analisis Bayesian pada Pemodelan *Mixture*

Gamerman (1997) menyatakan bahwa analisis Bayesian didasarkan pada distribusi posterior yang diperoleh melalui hasil perkalian antara fungsi *likelihood* dengan distribusi priornya. Hal ini memungkinkan bagi para pengguna analisis ini untuk berkreasi dalam mengakomodir dan mengimplementasikan kejadian perubahan pada datanya ke dalam sebuah model yang mengacu pada konsep *data driven*. Sehingga

semakin tinggi level yang dapat dimasukkan dalam struktur pemodelannya menyebabkan model tersebut semakin adaptif dalam mewakili sistem kerja yang dimodelkan. Dengan demikian model tersebut dapat dibentuk ke dalam model hirarki.

Prosedur analisis Bayesian pada model *mixture* memperlakukan semua parameter model sebagai variabel random yang mempunyai distribusi prior tertentu. Pada kondisi banyaknya komponen penyusun *mixture* tidak diketahui, ada 3 komponen yang dijadikan sebagai tujuan dari analisis Bayesian pada model (2.4). Ketiga komponen tersebut adalah: banyaknya komponen penyusun *mixture* k , komponen parameter model θ_j dan komponen pembobot w_j .

Pemodelan *mixture* pada kondisi di atas dilakukan dengan memandang bahwa identitas setiap pengamatan x_i pada setiap subpopulasi tidak diketahui. Pada kondisi ini Richardson dan Green (1997) menggunakan indikator z sebagai variabel random yang menotasikan alokasi setiap pengamatan pada setiap subpopulasi dari model *mixture* pada (2.4). Alokasi setiap pengamatan z_i ditentukan dari distribusi berikut :

$$p(z_i = j) = w_j \quad \text{untuk } j = 1, \dots, k \quad (2.9)$$

Sehingga jika diberikan nilai z_i maka data pengamatan x_i berasal dari setiap subpopulasi yang berdistribusi sebagai berikut:

$$x_i | z_i \sim f(x | \theta_{z_i}) \quad \text{untuk } i = 1, \dots, n \quad (2.10)$$

Dengan demikian posterior yang dihasilkan dari model *mixture* tersebut berbentuk distribusi gabungan dari semua variabel pada model (2.4) yang dinyatakan dalam persamaan berikut :

$$\pi(k, \mathbf{w}, \mathbf{z}, \boldsymbol{\theta}, \mathbf{x}) = p(k) p(\mathbf{w} | k) p(\mathbf{z} | \mathbf{w}, k) p(\boldsymbol{\theta} | \mathbf{z}, \mathbf{w}, k) p(\mathbf{x} | \boldsymbol{\theta}, \mathbf{z}, \mathbf{w}, k) \quad (2.11)$$

Selanjutnya untuk mendapatkan estimasi dari setiap parameter pada posterior di atas dilakukan dengan membentuk distribusi bersyarat penuh dari setiap parameternya. Hal ini akan dipermudah jika struktur hirarki dari posterior di atas dibentuk terlebih dahulu. Richardson dan Green (1997) menggunakan *Directed Acyclic Graph* (DAG) untuk menggambarkan struktur hirarki dari suatu *mixture*.

Pembahasan mengenai penentuan distribusi prior, pembentukan fungsi *likelihood*, distribusi bersyarat penuh dan struktur hirarki akan diberikan pada sub subbab berikut.

2.5.1. Distribusi Prior

Penentuan distribusi prior erat kaitannya dengan distribusi parameter model *mixture*. Berdasarkan hasil identifikasi terhadap pola data, Box dan Tiao (1973) mengelompokkan distribusi prior menjadi dua macam, yaitu distribusi prior konjuget dan distribusi prior non konjuget. Penentuan distribusi prior konjuget didasarkan pada pola yang sekawan dengan distribusi data. Sedangkan distribusi prior non konjuget didasarkan pada pola yang tidak sekawan dengan distribusi data.

Jika dikaitkan dengan penentuan parameter pada distribusi prior, maka Box dan Tiao (1973) mengelompokkan distribusi prior menjadi dua macam, yaitu prior informatif dan prior non informatif. Apabila penentuan parameter distribusi prior didasarkan pada informasi yang diperoleh dari data, maka prior tersebut dikatakan informatif. Sebaliknya, jika penentuan parameter distribusi prior tidak didasarkan pada informasi dari data, maka prior tersebut dikatakan non informatif. Sehingga suatu prior akan diidentifikasi bersifat konjuget dan informatif jika distribusi prior yang digunakan

mempunyai pola sekawan dengan distribusi data dan nilai parameter yang diberikan pada distribusi prior tersebut didasarkan pada informasi yang diperoleh dari data.

2.5.2. Fungsi Likelihood pada Model Mixture

Distribusi *mixture*, biasa juga disebut sebagai model *mixture*, mempunyai struktur yang berbeda dari distribusi lainnya. Dengan demikian fungsi *likelihood* dari distribusi *mixture* juga berbeda dengan fungsi *likelihood* dari distribusi univariat biasa (McLachlan dan Basford, 1988).

Jika terdapat data pengamatan sebanyak n yang terdekomposisi ke dalam k kelompok data (subpopulasi) yang masing-masing mempunyai distribusi, maka fungsi *likelihood* model *mixture* akan tersusun dari beberapa fungsi *likelihood* di setiap subpopulasi menurut distribusinya masing-masing.

Berdasarkan model *mixture* pada persamaan (2.4), McLachlan dan Basford (1988) menyatakan fungsi *likelihood* model *mixture* sebagai berikut:

$$L_{\text{mix}} = \prod_{i_1=1}^{n_1} w_1 f(x_{i_1} | \theta_1) + \prod_{i_2=1}^{n_2} w_2 f(x_{i_2} | \theta_2) + \dots + \prod_{i_k=1}^{n_k} w_k f(x_{i_k} | \theta_k) \quad (2.12)$$

$$L_{\text{mix}} = \sum_{j=1}^k \left(\prod_{i_j=1}^{n_j} w_j f(x_{i_j} | \theta_j) \right) \quad (2.13)$$

2.5.3. Distribusi Bersyarat Penuh

Dimisalkan terdapat variabel θ yang mempunyai distribusi stasioner $\pi(\theta)$. Distribusi bersyarat penuh dari θ dapat diperoleh dengan membentuk partisi dari θ sebagai berikut :

$$\theta = (\theta_k, \theta_{-k}) \quad (2.14)$$

dengan θ_k menotasikan variabel yang akan diestimasi dan θ_{-k} menotasikan komponen dari θ_k .

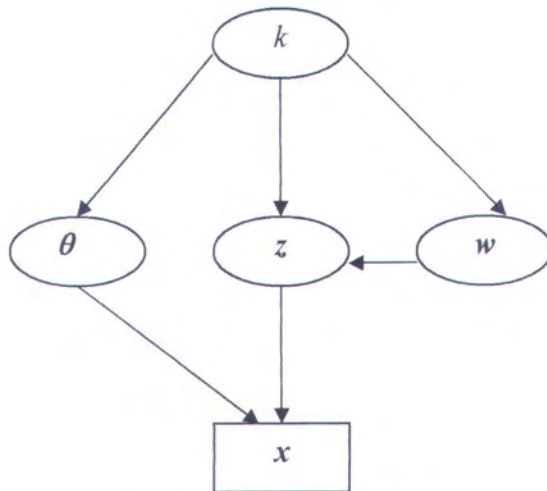
Selanjutnya Gilks (1995) dalam Suharjo (2003) menjelaskan pembentukan distribusi bersyarat penuh melalui penurunan dari bentuk distribusi gabungan semua variabel sebagai berikut :

$$\pi(\theta_k | \theta_{-k}) = \frac{\pi(\theta_k, \theta_{-k})}{\int \pi(\theta_k, \theta_{-k}) d\theta_{-k}} \quad (2.15)$$

2.5.4. Struktur Hirarki

Struktur hirarki dari model *mixture* di atas dapat digambarkan dengan *Directed Acyclic Graph* (DAG). DAG disusun dalam *node* kotak dan *node* ellips. *Node* kotak menunjukkan bahwa parameter tersebut bernilai konstan, sedangkan *node* ellips menunjukkan bahwa parameter tersebut bisa berubah secara stokastik atau *logical*. Perubahan *node* ellips secara stokastik disebut sebagai *node* stokastik sedangkan perubahan *node* ellips secara *logical* disebut sebagai *node logical*. Keduanya dibedakan oleh *directed graph* inputnya yang bergaris tunggal dan ganda.

Richardson dan Green (1997) menyajikan struktur hirarki dari model *mixture* dengan banyak komponen *mixture* tidak diketahui dalam DAG berikut.



Gambar 2.1 : *Directed Acyclic Graph (DAG)* untuk model *mixture* secara umum

Masing-masing parameter pada DAG di atas terdiri dari: variabel k yang menotasikan banyak komponen penyusun *mixture*, variabel w menotasikan pembobot dari setiap komponen penyusun *mixture*, variabel z menotasikan alokasi setiap pengamatan di setiap subpopulasi dan variabel θ menotasikan parameter model *mixture* di setiap komponen penyusun *mixture*. Sedangkan x adalah data pengamatan.

2.6. Analisis Bayesian pada Pemodelan *Mixture* Normal Univariat

Berdasarkan penjelasan pada sub bab 2.5. di atas, maka pembahasan mengenai pemodelan *mixture* normal dilakukan dengan menggunakan langkah-langkah dalam analisis Bayesian sebagai berikut.

2.6.1. Penentuan Distribusi Prior

Penentuan distribusi prior untuk setiap parameter model dilakukan dengan menggunakan pendekatan struktur hirarki. Richardson dan Green (1997) menentukan prior yang akan digunakan dalam pemodelan *mixture* normal sebagai berikut :

$$\begin{aligned}\mu_j &\sim N(\xi, \kappa^{-1}) \quad ; j = 1, 2, \dots, k \\ \sigma_j^{-2} | \beta &\sim \Gamma(\alpha, \beta) \quad ; j = 1, 2, \dots, k \\ \beta &\sim \Gamma(g, h) \\ w &\sim D(\delta, \delta, \dots, \delta)\end{aligned}$$

dimana β adalah hiperparameter untuk σ_j^{-2} dan $\xi, \kappa, \alpha, g, h, \delta$ adalah parameter yang nilainya ditentukan berdasarkan interval variasi data yang diamati dan panjang intervalnya. Nilai presisi κ ditentukan sama dengan $\frac{1}{R^2}$ dimana R adalah rentang data.

Sedangkan untuk hiperparameter β ditentukan berdistribusi $\Gamma(g, h)$ dimana $\alpha > 1$ dan $g < 1$ serta h ditentukan sama dengan $\frac{1}{R^2}$. Adapun setiap data pengamatan x_i

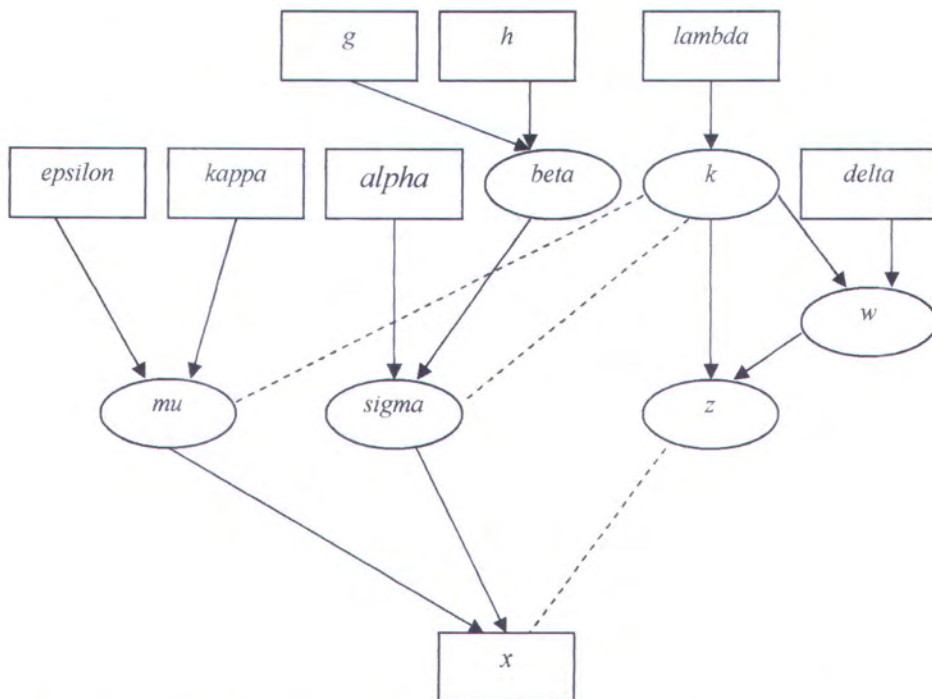
diambil secara independen dari subpopulasi z_i yang tidak diketahui dengan $p(z_i = j) = w_j$ dan $x_i | z_i \sim N(\mu_{z_i}, \sigma_{z_i}^2)$ untuk $i = 1, 2, \dots, n$. Selanjutnya untuk identifiabilitas data, μ_j ditentukan terurut sebagai berikut: $\mu_1 < \mu_2 < \dots < \mu_k$ dengan $j = 1, 2, \dots, k$. Sedangkan untuk parameter pembobot dari tiap komponen *mixture* ditentukan sebagai $w | k \sim D(\delta, \delta, \dots, \delta)$ dengan $k \sim U[1, 2, \dots, k_{\max}]$.

Selanjutnya untuk mempermudah proses estimasi setiap parameter model *mixture* normal, maka prior yang sudah ditentukan di atas akan dibentuk ke dalam

struktur hirarki dengan menggunakan *Directed Acyclic Graph* (DAG). Pembahasan mengenai pembentukan DAG diberikan sebagai berikut.

2.6.2. Pembentukan Struktur Hirarki

Berdasarkan penjelasan mengenai struktur hirarki pada sub subbab 2.5.4, maka struktur hirarki yang menggambarkan pembentukan model *mixture* normal dengan prior yang sudah ditentukan pada penjelasan sebelumnya disajikan pada Gambar 2.2. Gambar tersebut berlaku untuk banyak komponen *mixture* k tidak diketahui. Adapun untuk banyak komponen *mixture* k diketahui, maka DAG pada Gambar 2.2 akan mengalami sedikit perubahan, yaitu pada penentuan *node* dari k ; yang semula *node* k berbentuk ellips karena k tidak diketahui diganti menjadi *node* kotak karena nilai k adalah konstanta.



Gambar 2.2 : DAG untuk estimasi distribusi *mixture* normal dengan banyak komponen k tidak diketahui

2.6.3. Distribusi Bersyarat Penuh

Estimasi untuk setiap parameter model *mixture* normal pada persamaan (2.5) diperoleh melalui bentuk distribusi bersyarat penuh dari setiap parameter modelnya. Sedangkan distribusi bersyarat penuh untuk setiap parameter model diperoleh melalui penurunan dari bentuk distribusi gabungan semua variabel dalam model. Berdasarkan model *mixture* pada persamaan (2.5) dan model pada persamaan (2.11), distribusi gabungan dari semua variabel dinyatakan sebagai berikut :

$$\pi(\mathbf{x}, k, \dots) = p(k) p(w|k) p(z|w, k) p(\mu|z, w, k) p(\sigma|\mu, z, w, k) p(\mathbf{x}|\sigma, \mu, z, w, k) \quad (2.16)$$

Berdasarkan bentuk distribusi gabungan di atas, maka distribusi bersyarat penuh untuk setiap variabel dapat ditentukan dengan menganggap variabel lain sebagai konstanta. Oleh karena itu distribusi bersyarat penuh untuk setiap variabel dinotasikan dengan $\pi(\bullet|\dots)$. Bentuk distribusi bersyarat penuh untuk setiap parameter pada distribusi *mixture* normal dinyatakan pada penjelasan berikut.

2.6.3.1. Distribusi Bersyarat Penuh untuk Parameter Pembobot

Distribusi bersyarat penuh untuk parameter w dengan prior yang sudah ditentukan pada penjelasan sebelumnya, yaitu berdistribusi Dirichlet dinyatakan sebagai berikut :

$$\pi(w|\dots) = \prod_{i=1}^n \left(\frac{\Gamma\left(\sum_{j=1}^k \delta_j\right)}{\prod_{j=1}^k \Gamma(\delta_j)} \prod_{j=1}^k w_j^{\delta_j-1} \right)$$

$$\begin{aligned}
&\propto \prod_{i=1}^n \prod_{j=k}^k w_j^{\delta-1} \\
&= \prod_{j=1}^k \prod_{i=1; z_i=j}^{n_j} w_j^{\delta-1} \quad ; n = n_1 + n_2 + \dots + n_k \\
&= \prod_{j=1}^k w_j^{\delta-1} \times w_j^{n_j} \\
&= \prod_{j=1}^k w_j^{\delta+n_j-1} \\
&\propto \frac{\Gamma\left(\sum_{j=1}^k (\delta+n_j)\right)}{\prod_{j=1}^k \Gamma(\delta+n_j)} \prod_{j=1}^k w_j^{\delta+n_j-1}
\end{aligned}$$

Bentuk terakhir ini merupakan bentuk dari distribusi Dirichlet berikut :

$$D[\delta + n_1, \delta + n_2, \dots, \delta + n_k]$$

Dengan demikian distribusi bersyarat penuh untuk parameter w adalah:

$$w|\dots \sim D(\delta + n_1, \delta + n_2, \dots, \delta + n_k) \quad (2.17)$$

2.6.3.2. Distribusi Bersyarat Penuh untuk Parameter Lokasi

Distribusi bersyarat penuh untuk parameter μ_j dengan prior yang sudah ditentukan pada penjelasan sebelumnya, yaitu berdistribusi normal dinyatakan sebagai berikut :

$$\begin{aligned}
\pi(\mu_j|\dots) &\propto \frac{P(\mu_j, \sigma_j^{-2}|x)}{P(\sigma_j^{-2}|x)} \\
&= \frac{P(x, \mu_j, \sigma_j^{-2})}{P(x, \sigma_j^{-2})} \propto P(x, \mu_j, \sigma_j^{-2})
\end{aligned}$$

sehingga distribusi bersyarat penuhnya adalah :

$$\begin{aligned}
 \pi(\mu_j | \dots) &\propto \exp\left\{-\frac{\sigma_j^{-2}}{2} \sum_{i=1; z_i=j}^{n_j} (x_i - \mu_j)^2\right\} \times \exp\left\{-\frac{\kappa}{2}(\mu_j - \xi)^2\right\} \\
 &= \exp\left\{-\frac{1}{2}\left(\sigma_j^{-2} \sum_{i=1; z_i=j}^{n_j} (x_i - \mu_j)^2 + \kappa(\mu_j - \xi)^2\right)\right\} \\
 &= \exp\left\{-\frac{1}{2}\left(\sigma_j^{-2} \sum_{i=1; z_i=j}^{n_j} x_i^2 - 2\sigma_j^{-2}\mu_j \sum_{i=1; z_i=j}^{n_j} x_i + \sigma_j^{-2} \sum_{i=1; z_i=j}^{n_j} \mu_j^2 + \kappa\mu_j^2 - 2\kappa\xi\mu_j + \kappa\xi^2\right)\right\} \\
 &= \exp\left\{-\frac{1}{2}\left(\sigma_j^{-2} \sum_{i=1; z_i=j}^{n_j} x_i^2 - 2\mu_j\left(\sigma_j^{-2} \sum_{i=1; z_i=j}^{n_j} x_i + \kappa\xi\right) + \mu_j^2(\sigma_j^{-2}n_j + \kappa) + \kappa\xi^2\right)\right\} \\
 &= \exp\left\{-\frac{1}{2}(\kappa + \sigma_j^{-2}n_j) \left(\mu_j^2 - \frac{2\mu_j\left(\sigma_j^{-2} \sum_{i=1; z_i=j}^{n_j} x_i + \kappa\xi\right) + \sigma_j^{-2} \sum_{i=1; z_i=j}^{n_j} x_i^2 + \kappa\xi^2}{\sigma_j^{-2}n_j + \kappa} + \frac{\sigma_j^{-2} \sum_{i=1; z_i=j}^{n_j} x_i^2 + \kappa\xi^2}{\sigma_j^{-2}n_j + \kappa}\right)\right\} \\
 &\propto \exp\left\{-\frac{1}{2}(\sigma_j^{-2}n_j + \kappa) \left(\mu_j - \frac{\left(\sigma_j^{-2} \sum_{i=1; z_i=j}^{n_j} x_i + \kappa\xi\right)}{\sigma_j^{-2}n_j + \kappa}\right)^2\right\}
 \end{aligned}$$

Bentuk terakhir di atas merupakan bentuk distribusi normal. Dengan demikian distribusi bersyarat penuh untuk parameter μ_j adalah :

$$\mu_j \left| \dots \sim N\left(\frac{\sigma_j^{-2} \sum_{i=1; z_i=j}^{n_j} x_i + \kappa\xi}{\sigma_j^{-2}n_j + \kappa}, (\sigma_j^{-2}n_j + \kappa)^{-1}\right) \quad (2.18)$$

dengan $j = 1, 2, \dots, k$.

2.6.3.3. Distribusi Bersyarat Penuh untuk Parameter Penyebaran

Dengan cara yang sama, maka distribusi bersyarat penuh untuk parameter

σ_j^{-2} adalah :

$$\begin{aligned}
 \pi(\sigma_j^{-2} | \dots) &\propto (\sigma_j^{-2})^{\frac{n_j}{2}} \exp\left\{-\frac{\sigma_j^{-2}}{2} \sum_{i=1; z_i=j}^{n_j} (x_i - \mu_j)^2\right\} \frac{\beta^\alpha (\sigma_j^{-2})^{\alpha-1} \exp(-\beta \sigma_j^{-2})}{\Gamma(\alpha)} \\
 &= \frac{\beta^\alpha (\sigma_j^{-2})^{\frac{n_j}{2} + \alpha - 1} \exp\left\{-\frac{\sigma_j^{-2}}{2} \sum_{i=1; z_i=j}^{n_j} (x_i - \mu_j)^2 - \beta \sigma_j^{-2}\right\}}{\Gamma(\alpha)} \\
 &= \frac{\beta^\alpha (\sigma_j^{-2})^{\frac{n_j}{2} + \alpha - 1} \exp\left\{-\sigma_j^{-2} \left(\beta + \frac{1}{2} \sum_{i=1; z_i=j}^{n_j} (x_i - \mu_j)^2\right)\right\}}{\Gamma(\alpha)} \\
 &\propto \frac{\left(\beta + \frac{1}{2} \sum_{i=1; z_i=j}^{n_j} (x_i - \mu_j)^2\right)^{\alpha + \frac{n_j}{2}} (\sigma_j^{-2})^{\alpha + \frac{n_j}{2} - 1} \exp\left\{-\sigma_j^{-2} \left(\beta + \frac{1}{2} \sum_{i=1; z_i=j}^{n_j} (x_i - \mu_j)^2\right)\right\}}{\Gamma\left(\alpha + \frac{1}{2} n_j\right)} \\
 &= \Gamma\left(\alpha + \frac{1}{2} n_j, \beta + \frac{1}{2} \sum_{i=1; z_i=j}^{n_j} (x_i - \mu_j)^2\right)
 \end{aligned}$$

Dengan demikian distribusi bersyarat penuh untuk parameter σ_j^{-2} dinyatakan sebagai

berikut :

$$\sigma_j^{-2} \Big| \dots \sim \Gamma\left(\alpha + \frac{1}{2} n_j, \beta + \frac{1}{2} \sum_{i=1; z_i=j}^{n_j} (x_i - \mu_j)^2\right) \quad (2.19)$$

dengan $j = 1, 2, \dots, k$.

2.6.3.4. Distribusi Bersyarat Penuh untuk Parameter Alokasi

Dengan cara yang sama, maka distribusi bersyarat penuh untuk parameter z adalah :

$$p(z_i = j | \dots) \propto w_j \frac{1}{\sqrt{2\pi}} \sigma_j^{-2} \exp\left(-\frac{\sigma_j^{-2}}{2} (x_i - \mu_j)^2\right)$$

maka distribusi bersyarat penuh untuk parameter z_i adalah :

$$z_i = j | \dots \sim \frac{w_j}{\sigma_j} \exp\left(-\frac{(x_i - \mu_j)^2}{2\sigma_j^2}\right) \quad (2.20)$$

2.6.3.5. Distribusi Bersyarat Penuh untuk Hiperparameter β

Adapun distribusi bersyarat penuh untuk hiperparameter β diperoleh melalui proses penurunan pada persamaan (2.15) dari bentuk distribusi gabungan semua variabel pada persamaan (2.16). Dengan demikian distribusi bersyarat penuh untuk hiper-parameter β dinyatakan sebagai berikut :

$$\begin{aligned} \pi(\beta | \dots) &\propto p(\sigma_j^{-2} | \alpha, \beta) p(\beta | g, h) \\ &= \frac{\beta^{k\alpha} (\sigma_j^{-2})^{\alpha-1} \exp\left\{-\sum_{j=1}^k \beta \sigma_j^{-2}\right\}}{\Gamma(k\alpha)} \times \frac{h^g \beta^{g-1} \exp(-\beta h)}{\Gamma(g)} \\ &\propto \frac{\beta^{k\alpha+g-1} \exp\left\{-\beta\left(h + \sum_{j=1}^k \sigma_j^{-2}\right)\right\} (\sigma_j^{-2})^{k\alpha-1} h^g}{\Gamma(g+k\alpha)} \\ &\propto \frac{\beta^{g+k\alpha-1} \exp\left\{-\beta\left(h + \sum_{j=1}^k \sigma_j^{-2}\right)\right\} \left(h + \sum_{j=1}^k \sigma_j^{-2}\right)^{(g+k\alpha)}}{\Gamma(g+k\alpha)} \\ &= \Gamma\left(g+k\alpha-1, h + \sum_{j=1}^k \sigma_j^{-2}\right) \end{aligned}$$

Jadi distribusi bersyarat penuh untuk hiperparameter β dinyatakan sebagai berikut :

$$\beta | \dots \sim \Gamma \left(g + k\alpha, h + \sum_{j=1}^k \sigma_j^{-2} \right) \quad (2.21)$$

2.7. Markov Chain Monte Carlo

Penggunaan metode Bayesian pada analisis *data driven* akan mengakibatkan dimensi estimasi modelnya akan semakin kompleks. Hal ini dikarenakan banyaknya variabel yang digunakan sebagai penyusun posterior seperti pada persamaan (2.11) dan persamaan (2.16). Banyaknya variabel ini merepresentasikan banyaknya pertimbangan (*decision variable*) yang digunakan dalam pengambilan keputusan. Markov Chain Monte Carlo, yang disingkat dengan MCMC, sebagai metode simulasi stokastik yang menirukan pola kerja Markov Chain akan memberikan bantuan penyelesaian pada permasalahan ini dengan membangkitkan data sampel berdasarkan skenario sampling atau algoritma tertentu.

Dua buah algoritma dalam metode MCMC yang terkenal adalah Gibbs Sampler (Casella dan George, 1992) dan Metropolis-Hastings (Hastings, 1970 dalam Suharjo, 2003). Kedua algoritma tersebut biasa digunakan pada kasus dengan dimensi vektor parameter model posterior bersifat tetap. Selanjutnya Green (1995) mengembangkan algoritma Metropolis-Hastings untuk mengatasi kondisi dengan dimensi vektor parameter model yang bersifat tidak tetap. Hasil pengembangan algoritma ini disebut dengan *Reversible Jump*. Pada tesis ini, algoritma yang akan digunakan untuk memodelkan suatu *mixture* dengan banyak komponen tidak diketahui adalah Gibbs Sampler dan *Reversible Jump*. Gibbs Sampler digunakan untuk memodelkan suatu *mixture*

dengan banyak komponen yang ditentukan berdasarkan hasil identifikasi data dan Reversible Jump digunakan untuk memodelkan suatu *mixture* dengan banyak komponen yang tidak ditentukan.

Penentuan model terbaik untuk model *mixture* yang dihasilkan dari kedua algoritma di atas dilakukan dengan menggunakan Struktur Perkalian Distribusi (SPD) dan kriteria Bayes Faktor ((Iriawan, 2000a), (Iriawan, 2000b) serta Kass dan Raftery (1995)).

2.7.1. Integrasi Monte Carlo

Hogg dan Craig (1978) menjelaskan bahwa integrasi Monte Carlo merupakan teknik integrasi dengan metode komputasi untuk mengestimasi $E[f(X)]$ dengan mengambil sampel $\{x_t, t = 1, \dots, n\}$ dari distribusi tertentu, yaitu $\pi(\bullet)$, kemudian melakukan pendekatan $E[f(X)] \approx \frac{1}{n} \sum_{t=1}^n f(x_t)$. Dengan demikian rata-rata populasi dari $f(X)$ diestimasi dengan rata-rata sampel. Kemudian cara untuk menghasilkan rata-rata sampel yang akurat dalam mengestimasi rata-rata populasi adalah menggunakan rantai markov yang mempunyai distribusi stasioner $\pi(\bullet)$.

2.7.2. Rantai Markov

Rantai Markov merupakan proses stokastik yang bersifat khusus, yang berhubungan dengan karakteristik atau sifat dari sederet variabel random. Gamerman (1997) dan Iriawan (2000a) mendefinisikan rantai Markov pada *state space* S sebagai

sederet variabel random $\{X_t\}_{t \geq 0}$ dimana nilai untuk masing-masing variabel random tersebut berada pada *state space* S dan distribusi dari X_t dengan diberikan semua nilai sebelumnya dari proses, yaitu X_0, X_1, \dots, X_{t-1} , hanya tergantung pada X_{t-1} . Definisi rantai Markov secara matematis diberikan pada definisi di bawah ini.

Definisi 2.1

Suatu proses stokastik $\{X_t | t \in T, t \geq 0\}$ dikatakan sebagai suatu rantai Markov jika diberikan suatu nilai untuk X_t sedemikian hingga distribusi bersyarat X_t dengan $X_1, X_2, X_3, \dots, X_{t-1}$ diketahui hanya akan bergantung pada nilai X_{t-1} saja, atau dapat dituliskan secara matematis sebagai berikut:

$$P(X_t \in A | X_{t-1} \in A_{t-1}, X_{t-2} \in A_{t-2}, \dots, X_0 \in A_0) = P(X_t \in A | X_{t-1} \in A_{t-1})$$

untuk semua $A_0, \dots, A_{t-1}, A \subset S$.

Suatu rantai Markov dikatakan berdistribusi invarian/stasioner $\pi(\bullet)$ dengan probabilitas transisi $P(x, y)$ jika memenuhi persamaan berikut:

$$\sum_{x \in S} \pi(x) P(x, y) = \pi(y), \quad \forall y \in S \tag{2.22}$$

Persamaan (2.22) dapat dinyatakan dalam bentuk perkalian vektor dan matriks berikut:

$\pi_y = \pi_x P$. Sedangkan sifat *irreducible* dari rantai Markov adalah jika terdapat

distribusi probabilitas $\phi(\bullet)$ pada *state space* S sedemikian hingga untuk semua $A \subset S$

maka $\phi(A) > 0$ dan $P(X_t \in A \text{ dan } t > 0 | X_0 = \theta) > 0$ untuk semua $\theta \in A$.

Jika *initial state* X_0 merupakan titik permulaan dari suatu iterasi dan dengan menganggap secara langsung berakibat pada X_t dengan mengabaikan X_1, X_2, \dots, X_{t-1} ; maka dapat dikatakan bahwa X_t bergantung secara langsung pada X_0 . Pada kondisi ini rantai Markov $P^{(t)}(\bullet | X_0)$ secara gradual melupakan *initial state*-nya dan pada akhirnya akan konvergen ke suatu distribusi stasioner $\pi(\bullet)$ dan tidak bergantung pada t atau X_0 . Jika kondisi tersebut terjadi pada iterasi ke- m , maka kondisi ini disebut dengan kondisi *burn-in*. Selanjutnya titik-titik sampel setelah iterasi ke- m , $\{X_t : t = m+1, \dots, n\}$ akan menjadi sampel yang berdistribusi $\pi(\bullet)$. Banyaknya iterasi yang diperlukan untuk mencapai kondisi *burn-in* ini tergantung pada penentuan X_0 dan laju konvergensi dari $P^{(t)}(X_t | X_0)$ ke $\pi(X_t)$ serta kemiripan antara $P^{(t)}(\bullet | \bullet)$ dan $\pi(\bullet)$.

Selanjutnya estimasi untuk $f(X)$ diperoleh dengan membuang sampel sebelum *burn-in*, sehingga hasil estimasi $f(X)$ dapat dinyatakan dalam persamaan berikut :

$$\overline{f(X)} = \frac{1}{n-m} \sum_{t=m+1}^n f(X_t) \quad (2.23)$$

Pola kerja integrasi Monte Carlo dengan memanfaatkan kondisi rantai Markov ini disebut metode Markov Chain Monte Carlo yang biasa disingkat dengan MCMC.

2.7.3. Gibbs Sampler

Gibbs Sampler merupakan bentuk khusus dari Metropolis-Hastings dimana setiap nilai acak yang terpilih akan selalu diterima sebagai sampel ($\alpha(X, Y) = 1$)

(Walsh, 2002). Prinsip kerja dari metode ini adalah penggunaan distribusi bersyarat penuh untuk mengestimasi setiap parameter dari suatu model. Hal ini sesuai dengan penjelasan dari Casella dan George (1992) yang mendefinisikan Gibbs Sampler sebagai suatu teknik simulasi untuk membangkitkan suatu variabel tanpa perlu menghitung densitasnya. Pada beberapa literatur, Gibbs Sampler juga disebut Gibbs Sampling.

Casella dan George (1992), Gamerman (1997) serta Walsh (2002) menjelaskan cara kerja dari metode ini sebagai berikut. Dimisalkan X adalah sampel random yang mempunyai distribusi stasioner $p(\theta|X)$ dimana X dan θ bisa dipartisi menjadi komponen-komponen $(X_1, \dots, X_r) \in S_1 \times \dots \times S_r$ dan $\theta = (\theta_1, \dots, \theta_r)$. Proses sampling secara langsung dari $p(\theta|X) = p(\theta_1, \dots, \theta_r|X)$ tidak dapat dilakukan, karena distribusi tersebut sangat kompleks disebabkan banyaknya parameter dalam model. Untuk mempermudah estimasi terhadap setiap parameter dapat dilakukan dengan cara membangkitkan sampel dari distribusi bersyarat penuh setiap parameter $(\theta_j; j = 1, \dots, r)$ terhadap komplemennya (θ_{-j}) dan data X yaitu $p(\theta_1|X, \theta_2, \dots, \theta_r), \dots, p(\theta_r|X, \theta_1, \dots, \theta_{r-1})$.

Gamerman (1997) dan Walsh (2002) menjelaskan proses *sampling* yang menggunakan metode Gibbs Sampler pada Algoritma 2.1 berikut :

Algoritma 2.1

Step 1. Set nilai awal untuk θ pada $k = 0$ sehingga $\theta^{(0)} = (\theta_1^{(0)}, \dots, \theta_r^{(0)})$

Step 2. Melakukan proses sampling untuk mendapatkan nilai dari θ_j dari distribusi bersyarat berikut :

1) Sampling $\theta_1^{(k+1)}$ dari $p(\theta_1 | X, \theta_2^{(k)}, \dots, \theta_r^{(k)})$

2) Sampling $\theta_2^{(k+1)}$ dari $p(\theta_2 | X, \theta_1^{(k+1)}, \theta_3^{(k)}, \dots, \theta_r^{(k)})$

⋮

r) Sampling $\theta_r^{(k+1)}$ dari $p(\theta_r | X, \theta_1^{(k+1)}, \theta_2^{(k+1)}, \dots, \theta_{r-1}^{(k+1)})$

Step 3. Melakukan iterasi pada step 2 sebanyak M kali dengan $M \rightarrow \infty$

Data yang dibangkitkan dengan menggunakan Algoritma 2.1 menghasilkan data yang konvergen. Jaminan terhadap kekonvergenan data diberikan melalui pembuktian proposisi berikut :

Proposisi 2.1

Jika distribusi bersyarat penuh dari θ_j memberikan probabilitas positif pada berbagai himpunan bagian dari S_j untuk $j = 1, \dots, r$, maka rantai Markov dengan distribusi stasioner $p(\theta_1, \dots, \theta_r | X)$ akan bersifat irreducible.

Pembuktian :

Dimisalkan $\theta_j^{(1)}, \theta_j^{(2)}, \dots, \theta_j^{(M)}$ adalah suatu rantai Markov dengan $j = 1, \dots, r$, sehingga distribusi dari $\theta_j^{(i)}$ untuk $i = 1, \dots, M$ dengan diberikan nilai dari $\theta_j^{(i-1)}, \dots, \theta_j^{(0)}$ hanya akan bergantung pada $\theta_j^{(i-1)}$ saja.

Dengan demikian pembuktian terhadap proposisi 2.1 cukup dilakukan dengan menunjukkan bahwa $p(\boldsymbol{\theta} | \mathbf{X}) = p(\theta_1, \dots, \theta_r | \mathbf{X})$ adalah distribusi stasioner yang memenuhi:

$$\theta^{(i)} \sim \pi \Rightarrow \theta^{(i+1)} \sim \pi$$

Dimisalkan $(\theta_1^{(i)}, \theta_2^{(i)}, \theta_3^{(i)}, \dots, \theta_r^{(i)})$ berdistribusi sesuai dengan distribusi stasionernya, yaitu $p(\theta_1, \dots, \theta_r | \mathbf{X})$, maka pada saat membangkitkan $\theta_1^{(i+1)}$ dalam $(\theta_1^{(i+1)}, \theta_2^{(i)}, \dots, \theta_r^{(i)})$ akan berdistribusi $p(\theta_1 | \mathbf{X}, \theta_2, \dots, \theta_r) p(\theta_2, \dots, \theta_r | \mathbf{X})$ yang sama dengan $p(\theta_1, \dots, \theta_r | \mathbf{X})$.

Demikian juga pada saat membangkitkan $\theta_2^{(i+1)}$ dalam $(\theta_1^{(i+1)}, \theta_2^{(i+1)}, \theta_3^{(i)}, \dots, \theta_r^{(i)})$ akan berdistribusi $p(\theta_2 | \mathbf{X}, \theta_1, \theta_3, \dots, \theta_r) p(\theta_1, \theta_3, \dots, \theta_r | \mathbf{X})$ yang juga sama dengan $p(\theta_1, \dots, \theta_r | \mathbf{X})$.

Demikian seterusnya hingga pada saat membangkitkan $\theta_r^{(i+1)}$ dalam $(\theta_1^{(i+1)}, \theta_2^{(i+1)}, \dots, \theta_r^{(i+1)})$ akan berdistribusi $p(\theta_r | \mathbf{X}, \theta_1, \dots, \theta_{r-1}) p(\theta_1, \dots, \theta_{r-1} | \mathbf{X})$ yang juga sama dengan $p(\theta_1, \dots, \theta_r | \mathbf{X})$.

Berdasarkan uraian di atas terbukti bahwa pada saat membangkitkan $\theta_j^{(i+1)}$ akan menghasilkan $\theta_j^{(i+1)}$ yang berdistribusi sesuai dengan distribusi stasionernya yang bernilai positif. Dengan demikian terbukti bahwa $\theta^{(i)} \sim p(\theta|X) \Rightarrow \theta^{(i+1)} \sim p(\theta|X)$ dengan $p(\theta|X) = p(\theta_1, \theta_2, \dots, \theta_r | X)$ adalah distribusi stasioner dari rantai Markov yang bersifat *irreducible*. \square

Casella dan George (1992) serta Hogg dan Craig (1978) menjelaskan bahwa sampel yang diperoleh melalui metode ini merupakan pasangan sampel yang berdistribusi $p(\theta|X)$ dan setiap parameter yang diestimasi berdistribusi sebagai berikut :

$$\hat{f}(\theta_j) = \frac{1}{N} \sum_{i=1}^N f(\theta_j | \theta_{-j}^{(k)}, X) \quad (2.24)$$

Persamaan di atas merupakan mean dari distribusi bersyarat θ_j dengan diberikan θ_{-j} dan X yang dapat ditulis sebagai $E\left\{f(\theta_j | \theta_{-j}^{(k)}, X)\right\}$.

2.7.4. Metropolis-Hastings

Metode penyusunan rantai Markov dengan menggunakan algoritma Metropolis-Hastings dijelaskan oleh Gamerman (1997) sebagai berikut. Dimisalkan terdapat probabilitas transisi kernel $P(X, Y)$ yang memenuhi persamaan berikut.

$$\pi(X)P(X, Y) = \pi(Y)P(Y, X), \quad \forall (X, Y) \quad (2.25)$$

Selanjutnya probabilitas transisi Kernel $P(X, Y)$ terdiri dari dua komponen, yaitu: $q(X, Y)$ dan $\alpha(X, Y)$ sehingga $P(X, Y) = q(X, Y)\alpha(X, Y)$ untuk $X \neq Y$, dimana

$q(X, Y)$ menotasikan densitas dari X yang akan diajukan untuk mendapatkan rantai Markov berikutnya dan $\alpha(X, Y)$ menotasikan probabilitas penerimaan untuk pergerakan dari X ke Y yang dinyatakan sebagai berikut :

$$\alpha(X, Y) = \min \left[1, \frac{\pi(Y)q(Y, X)}{\pi(X)q(X, Y)} \right] \quad (2.26)$$

Adapun $q(Y, X)$ menotasikan densitas dari Y untuk kembali ke state X .

Jika titik kandidat Y diterima sebagai sampel, maka *state* berikutnya menjadi $X_{t+1} = Y$; sedangkan jika titik kandidat Y ditolak sebagai sampel, maka rantai Markov tidak berubah sehingga $X_{t+1} = X_t$. Gamerman (1997) dan Walsh (2002) memberikan algoritma dari metode Metropolis-Hastings sebagai berikut :

Algoritma 2.2

Inisialisasi X_0 ; set $t = 0$

REPEAT {

Bangkitkan sampel Y dari $q(\bullet | X_t)$

Bangkitkan variabel random $u \sim U(0,1)$

IF $u \leq \alpha(X_t, Y)$ set $X_{t+1} = Y$

ELSE set $X_{t+1} = X_t$

Ulangi sebanyak t }

Dengan demikian probabilitas transisi Kernel $P(X, Y)$ tidak selalu menggambarkan semua transisi dari X ke Y karena ada kemungkinan penolakan transisi dari X ke Y . Probabilitas terjadinya penolakan transisi dari X ke Y dinyatakan sebagai berikut :

$$p(X, X) = r(X) = 1 - \int_{-\infty}^{\infty} q(X, Y) \alpha(X, Y) dY \quad (2.27)$$

2.7.5. Reversible Jump

Pada umumnya penggunaan algoritma MCMC pada proses inferensi Bayesian terbatas hanya pada kasus dimana dimensi vektor parameter model bersifat tetap. Hal ini akan menjadi tidak berlaku pada kondisi dimana dimensi vektor parameter model adalah tidak tetap. Oleh karena itu Green (1995) mengembangkan algoritma Metropolis-Hastings untuk mengatasi permasalahan tersebut yang kemudian disebut dengan Reversible Jump Markov Chain Monte Carlo yang disingkat dengan RJMCMC.

Pada pemodelan *mixture* dengan banyak komponen penyusun *mixture* yang tidak diketahui, Richardson dan Green (1997) memanfaatkan algoritma ini untuk mendapatkan model *mixture* yang tepat bagi data yang terindikasi bersifat multimodal. Algoritma ini memanfaatkan proses *birth/death* dan *split/merge* dalam aplikasinya.

Richardson dan Green (1997) menjelaskan proses RJMCMC ini ke dalam enam tipe pergerakan yaitu:

1. *updating w*
2. *updating θ*
3. *updating z*

4. *updating hyperparameter* β
5. *split/merge* komponen-komponen
6. *birth/death* dari komponen kosong

Proses pada langkah 1 sampai 4 tidak mengubah dimensi vektor parameter model yang terdiri dari (β, θ, k, w, z) ; sedangkan langkah 5 dan 6 mengubah banyaknya komponen penyusun *mixture* k satu per satu.

Pergerakan *split/merge* adalah pilihan random antara *split* ($k \rightarrow k + 1$) atau *merge* ($k + 1 \rightarrow k$).

Langkah-langkah dalam proses *merge* tersebut terdiri dari:

- 1) gabungkan pasangan yang berseberangan $(j_1, j_2) \rightarrow j_*$
- 2) realokasi x_i dengan $z_i \in \{j_1, j_2\}$
- 3) tentukan nilai untuk $(w_{j_*}, \mu_{j_*}, \sigma_{j_*})$ dengan

$$w_{j_*} = w_{j_1} + w_{j_2}$$

$$w_{j_*} \mu_{j_*} = w_{j_1} \mu_{j_1} + w_{j_2} \mu_{j_2}$$

$$w_{j_*} (\mu_{j_*}^2 + \sigma_{j_*}^2) = w_{j_1} (\mu_{j_1}^2 + \sigma_{j_1}^2) + w_{j_2} (\mu_{j_2}^2 + \sigma_{j_2}^2)$$

Sedangkan langkah-langkah dalam proses *split* terdiri dari:

- 1) komponen terpilih secara random dipisah sehingga $j_* \rightarrow j_1 + j_2$
- 2) membangun vektor random $u = (u_1, u_2, u_3)$ untuk membentuk parameter baru yang sesuai dengan persamaan di atas

$$\begin{aligned}
w_{j1} &= w_{j*}u_1 \\
w_{j2} &= w_{j*}(1-u_1) \\
\mu_{j1} &= \mu_{j*} - u_2\sigma_{j*}\sqrt{w_{j2}/w_{j1}} \\
\mu_{j2} &= \mu_{j*} + u_2\sigma_{j*}\sqrt{w_{j1}/w_{j2}} \\
\sigma_{j1}^2 &= u_3(1-u_2^2)\sigma_{j*}^2 w_{j*}/w_{j1} \\
\sigma_{j2}^2 &= (1-u_3)(1-u_2^2)\sigma_{j*}^2 w_{j*}/w_{j2}
\end{aligned}$$

- 3) memeriksa kondisi yang berdekatan sehingga diperoleh $\mu_{j1} < \mu_{j2}$

Adapun pergerakan dalam proses *birth/death* merupakan pilihan random antara *birth* ($k \rightarrow k+1$) atau *death* ($k+1 \rightarrow k$).

Langkah-langkah dalam proses *birth* terdiri dari :

- 1) Membentuk parameter $(w_{j*}, \mu_{j*}, \sigma_{j*})$ untuk komponen baru dengan prior $w_{j*} \sim \text{Beta}(1, k)$
- 2) Melakukan penyekalaan kembali sehingga diperoleh $\sum_{j=1}^k w_j = 1$

Sedangkan langkah-langkah dalam proses *death* terdiri dari :

- 1) Menghapus komponen kosong yang terpilih secara random
- 2) Melakukan penyekalaan kembali dari pembobot yang tersisa

2.7.6. Pemilihan Model Terbaik dengan Struktur Perkalian Distribusi (SPD)

Struktur Perkalian Distribusi (SPD) dikembangkan oleh Iriawan (2000a) dan Iriawan (2000b) sebagai suatu metode pemilihan model terbaik tanpa memperhatikan apakah residualnya memiliki pola tertentu atau tidak. Iriawan (2000a) dan Iriawan (2000b) menjelaskan cara kerja SPD dengan membentuk distribusi gabungan dari beberapa model dengan menggunakan asas perkalian dan tidak memperhatikan asumsi normalitas pada residualnya.

Dimisalkan terdapat dua model berbeda yang akan dibandingkan untuk memperoleh model yang tepat, yaitu $f_1(x, \theta_1)$ dan $f_2(x, \theta_2)$. Selanjutnya dengan menganggap kedua model tersebut saling independen, maka dibentuk densitas baru yaitu $f_{SPD}(x, \lambda, \theta_1, \theta_2)$ dengan cara masing-masing densitas yang akan dibandingkan dipangkatkan dengan parameter λ . Parameter λ digunakan sebagai indikator dominasi model dalam $f_{SPD}(x, \lambda, \theta_1, \theta_2)$. Dengan demikian densitas baru yang dibentuk dari kedua densitas yang akan dibandingkan tersebut dapat dituliskan sebagai berikut:

$$f_{SPD}(\lambda, x, \theta_1, \theta_2) = C(\lambda, \theta_1, \theta_2) f_1^\lambda(x, \theta_1) f_2^{1-\lambda}(x, \theta_2) \quad (2.28)$$

dimana $C(\lambda, \theta_1, \theta_2)$ adalah konstanta normalitas dengan nilai $0 < \lambda < 1$.

Karena $f_1(x, \theta_1)$ dan $f_2(x, \theta_2)$ merupakan fungsi dari x maka $C(\lambda, \theta_1, \theta_2)$ dapat diperoleh dengan cara sebagai berikut:

$$C(\lambda, \theta_1, \theta_2) = \left[\int_{-\infty}^{\infty} f_1^\lambda(x, \theta_1) f_2^{1-\lambda}(x, \theta_2) dx \right]^{-1} \quad (2.29)$$

Persamaan (2.29) menunjukkan bahwa $C(\lambda, \theta_1, \theta_2)$ adalah suatu konstanta. Oleh karena itu persamaan (2.28) dapat ditulis dalam bentuk proporsional sebagai berikut:

$$f_{SPD}(x, \lambda, \theta_1, \theta_2) \propto f_1^\lambda(x, \theta_1) f_2^{1-\lambda}(x, \theta_2) \quad (2.30)$$

Kekuatan dominasi model dalam $f_{SPD}(x, \lambda, \theta_1, \theta_2)$ ditentukan oleh nilai λ . Jika nilai λ mendekati nol, maka $f_{SPD}(x, \lambda, \theta_1, \theta_2)$ akan didominasi oleh $f_2(x, \theta_2)$; sedangkan jika nilai λ mendekati satu, maka $f_{SPD}(x, \lambda, \theta_1, \theta_2)$ akan didominasi oleh $f_1(x, \theta_1)$; dan jika $f_1(x, \theta_1) = f_2(x, \theta_2)$, maka kekuatan dominasi $f_1(x, \theta_1)$ dan

$f_2(x, \theta_2)$ terhadap $f_{\text{SPD}}(x, \lambda, \theta_1, \theta_2)$ adalah sama sehingga nilai $C(\lambda, \theta_1, \theta_2)$ pada persamaan (2.29) sama dengan satu.

Selanjutnya fungsi *likelihood* dan logaritma dari *likelihood* $f_{\text{SPD}}(x, \lambda, \theta_1, \theta_2)$ pada persamaan (2.28) dituliskan sebagai berikut:

$$L_{f_{\text{SPD}}} = (C(\lambda, \theta_1, \theta_2))^n \prod_{i=1}^n f_1^\lambda(x_i, \theta_1) f_2^{1-\lambda}(x_i, \theta_2) \quad (2.31)$$

dan

$$\ln(L_{f_{\text{SPD}}}) = k + \sum_{i=1}^n [\lambda \ln(f_1(x_i, \theta_1)) + (1-\lambda) \ln(f_2(x_i, \theta_2))] \quad (2.32)$$

dimana nilai konstanta k adalah logaritma dari konstanta normalitas $C(\lambda, \theta_1, \theta_2)$.

Dengan demikian pada kasus khusus, yaitu $f_1(x, \theta_1) = f_2(x, \theta_2) = f(x)$, maka persamaan (2.32) akan menjadi:

$$\begin{aligned} \ln(L_{f_{\text{SPD}}}) &= k + (\lambda + (1-\lambda)) \sum_{i=1}^n \ln(f(x_i)) \\ &= k + \ln(L) \end{aligned} \quad (2.33)$$

dengan L adalah fungsi *likelihood* densitas $f(X)$.

Selanjutnya Prihartanti (2002) menjelaskan cara kerja MCMC untuk mengestimasi setiap parameter dalam model SPD dengan menggunakan *likelihood* model $f_{\text{SPD}}(x, \lambda, \theta_1, \theta_2)$ dari data dan menginteraksikannya dengan informasi prior setiap parameter dalam model yang berbentuk distribusi posterior bersyarat penuh (*full conditional posterior*) untuk setiap parameternya. Distribusi posterior bersyarat penuh untuk setiap parameter model tersebut diperoleh melalui turunan dari distribusi

posterior gabungan semua parameter terhadap data. Bentuk distribusi posterior tersebut adalah:

$$f_{\text{SPD}}(\theta_1, \theta_2, \lambda | x) \propto L_{f_{\text{SPD}}}(x | \theta_1, \theta_2, \lambda) p(\theta_1) p(\theta_2) p(\lambda) \quad (2.34)$$

Dengan demikian bentuk distribusi posterior bersyarat penuh untuk setiap parameter model adalah:

$$\ln[f_{\text{SPD}}(\theta_1 | \theta_2, \lambda, x)] = k + \lambda \sum_{i=1}^n \ln[f_1(x_i, \theta_1)] + \ln[p(\theta_1)] \quad (2.35)$$

$$\ln[f_{\text{SPD}}(\theta_2 | \theta_1, \lambda, x)] = k + (1 - \lambda) \sum_{i=1}^n \ln[f_2(x_i, \theta_2)] + \ln[p(\theta_2)] \quad (2.36)$$

$$\ln[f_{\text{SPD}}(\lambda | \theta_1, \theta_2, x)] = k + \lambda \sum_{i=1}^n \ln[f_1(x_i, \theta_1)] + (1 - \lambda) \sum_{i=1}^n \ln[f_2(x_i, \theta_2)] + \ln[p(\lambda)] \quad (2.37)$$

dengan k adalah konstanta.

Selanjutnya jika terdapat m model dalam fungsi SPD, maka bentuk umum dari model SPD dapat ditulis sebagai berikut:

$$f_{\text{SPD}}(x, \Lambda, \Theta) = C(\Lambda, \Theta) \prod_{j=1}^m f_j^{\lambda_j}(x, \theta_j) \\ \propto \prod_{j=1}^m f_j^{\lambda_j}(x, \theta_j) \quad (2.38)$$

dengan $\Lambda = (\lambda_1, \lambda_2, \dots, \lambda_m)$, $\Theta = (\theta_1, \theta_2, \dots, \theta_m)$, $0 < \lambda_j < 1$, $\sum_{j=1}^m \lambda_j = 1$ dan $m > 2$.

Selanjutnya $C(\Lambda, \Theta)$ dihitung sesuai persamaan (2.29). Model *likelihood* persamaan (2.38) adalah:

$$L_{f_{\text{SPD}}} = (C(\Lambda, \Theta))^n \prod_{j=1}^m \prod_{i=1}^n f_j^{\lambda_j}(x_i, \theta_j) \quad (2.39)$$

Dengan demikian distribusi posterior gabungannya adalah:

$$f_{\text{SPD}}(\Lambda, \Theta | x) \propto L(x | \Lambda, \Theta) \prod_{j=1}^m p(\lambda_j) p(\theta_j)$$

$$\propto [C(\Lambda, \Theta)]^n \prod_{j=1}^m \left[\prod_{i=1}^n f_j^{\lambda_j}(x_i, \theta_j) \right] p(\theta_j) p(\lambda_j) \quad (2.40)$$

Distribusi posterior bersyarat penuh untuk setiap parameter model adalah:

$$\ln \left[f_{\text{SPD}}(\theta_{j(s)} | \theta_{-j(s)}, \Lambda, x) \right] = k + \lambda_j \sum_{i=1}^n \ln(f_j(x_i, \theta_{j(s)})) + \ln(p(\theta_{j(s)})) \quad (2.41)$$

$$\ln \left[f_{\text{SPD}}(\lambda_j | \lambda_{-j}, \Theta, x) \right] = k + \lambda_j \sum_{i=1}^n \ln(f_j(x_i, \theta_j)) +$$

$$\left[\left(1 - \sum_{l=1, l \neq j}^m \lambda_l \right) - \lambda_j \right] \sum_{i=1}^n \ln(f_m(x_i, \theta_m)) \quad (2.42)$$

dengan k adalah konstanta dan s adalah banyaknya parameter pada model ke- j .

Prihartanti (2002) dan Suharjo (2003) menyusun algoritma dari SPD dengan dua distribusi penyusun dan lebih dari dua distribusi penyusun yang disajikan pada Algoritma 2.3 dan 2.4 berikut :

Algoritma 2.3 : (MCMC untuk SPD dengan dua distribusi penyusun)

Step 1. Bangkitkan parameter-parameter dari model pertama : θ_{1s} , $s = 1, 2, \dots, q_1$

dengan menggunakan persamaan (2.35)

Step 2. Bangkitkan parameter-parameter dari model pertama : θ_{2s} , $s = 1, 2, \dots, q_2$

dengan menggunakan persamaan (2.36)

Step 3. Bangkitkan parameter λ dengan $0 < \lambda < 1$ dengan menggunakan

persamaan (2.37)

Algoritma 2.4 : (MCMC untuk SPD dengan lebih dari dua distribusi penyusun)

Step 1. {Blok pembangkitan parameter-parameter penyusun SPD, Θ , terdiri dari j blok, $j = 1, 2, \dots, m$, untuk membangkitkan sejumlah q_j parameter dari model ke- j , θ_j , dengan menggunakan persamaan (2.41)}

FOR $j=1$ TO m DO {Blocking model}

FOR $s=1$ TO q_j DO

Bangkitkan $\theta_{j(s)}$ dengan menggunakan persamaan (2.41)

Step 2. {Blok pembangkitan λ dengan menggunakan persamaan (2.42) }

a) FOR $j=1$ TO $(p-1)$ DO

Bangkitkan λ_j dengan menggunakan persamaan (2.42)

b) Hitung $\lambda_m = 1 - \sum_{j=1}^{m-1} \lambda_j$

2.7.7. Bayes Faktor

Penyelesaian analisis data yang menggunakan metode Bayesian untuk uji hipotesis dalam memilih model terbaik yang sesuai dengan data adalah dengan menghitung Bayes faktor dari salah satu model terhadap model lainnya. Bayes faktor merupakan ringkasan dari semua bukti yang diberikan oleh data untuk lebih condong dan sesuai pada salah satu distribusi atau model dari sejumlah m – distribusi atau model yang dibandingkan dengan menggunakan SPD.

Kass dan Raftery (1995) menjelaskan prinsip kerja Bayes faktor sebagai berikut. Dimisalkan ada dua model yang sesuai untuk data x , yaitu M_1 dan M_2 , maka

penentuan model yang paling sesuai untuk data x dilakukan dengan menguji kedua model tersebut menggunakan hipotesis berikut :

$$\begin{aligned} H_1 : x &\sim M_1 \\ H_2 : x &\sim M_2 \end{aligned}$$

dan pengambilan keputusan terhadap kedua hipotesis di atas didasarkan pada probabilitas $\Pr(x|H_1)$ atau $\Pr(x|H_2)$.

Jika diberikan probabilitas prior untuk tiap hipotesis di atas adalah $\Pr(H_1)$ dan $\Pr(H_2) = 1 - \Pr(H_1)$; maka akan diperoleh dua probabilitas posterior yaitu: $\Pr(H_1|x)$ dan $\Pr(H_2|x) = 1 - \Pr(H_1|x)$. Proses ini menunjukkan bahwa prior seakan-akan ditransformasikan ke dalam bentuk posterior melalui interaksi dengan data. Dengan menggunakan bentuk konversi odds sebagai berikut :

$$\text{odds} = \frac{\text{probabilitas}}{1 - \text{probabilitas}} \quad (2.43)$$

maka dapat diperoleh suatu bentuk transformasi yang cukup sederhana. Apabila diberikan bentuk teorema Bayes dari kedua hipotesis di atas sebagai berikut :

$$\Pr(H_j|x) = \frac{\Pr(x|H_j)\Pr(H_j)}{\Pr(x|H_1)\Pr(H_1) + \Pr(x|H_2)\Pr(H_2)} \quad ; j = 1, 2 \quad (2.44)$$

maka akan diperoleh persamaan berikut :

$$\frac{\Pr(H_1|x)}{\Pr(H_2|x)} = \frac{\Pr(x|H_1)\Pr(H_1)}{\Pr(x|H_2)\Pr(H_2)} \quad (2.45)$$

Selanjutnya dari persamaan (2.45) akan diperoleh bentuk transformasi berikut :

$$Bf_{12} = \frac{\Pr(x|H_1)}{\Pr(x|H_2)} \quad (2.46)$$

Persamaan (2.46) merupakan cara penghitungan Bayes faktor untuk kedua model tersebut. Kemudian berdasarkan persamaan (2.45) dan (2.46) dapat dikatakan bahwa :

$$\text{posterior odds} = \text{Bayes faktor} \times \text{prior odds} \quad (2.47)$$

Pada kasus khusus dimana hipotesisnya hanya memuat distribusi yang tunggal dengan tanpa parameter yang independen, maka nilai Bayes faktor, Bf_{12} , dapat diperoleh dengan menghitung nilai *likelihood ratio*-nya (Kass dan Raftery, 1995).

Pada implementasi metode MCMC untuk memilih model terbaik menggunakan SPD telah dibahas mengenai bentuk posterior distribusi gabungan dan distribusi bersyarat penuh untuk tiap parameter model. Posterior untuk parameter λ yang diperoleh melalui persamaan (2.42) menunjukkan dominasi distribusi atau model ke- j di antara m -model atau distribusi yang dibandingkan dalam fungsi SPD. Sehingga apabila banyaknya distribusi atau model ke- j yang muncul dalam N -iterasi diketahui, maka nilai Bayes faktor dari model ke- j terhadap model lainnya dapat dihitung dengan menggunakan persamaan (2.46) ((Carlin dan Chib,1995) dan (Iriawan,2000a)). Jadi nilai Bayes faktor dapat diperoleh dari SPD dengan menggunakan bantuan metode MCMC. Cara penghitungan tersebut diberikan pada persamaan berikut :

a) Untuk dua distribusi penyusun dalam SPD :

$$Bf_{12} = \frac{\text{banyaknya } \lambda_1^{(g)} > 0.5 \text{ yang diperoleh dari } N - \text{iterasi}}{\text{banyaknya } \lambda_1^{(g)} < 0.5 \text{ yang diperoleh dari } N - \text{iterasi}} \quad (2.48)$$

b) Untuk m -distribusi penyusun dalam SPD :

$$Bf_{ij} = \frac{\text{banyaknya } \lambda_i^{(g)} \text{ yang diperoleh dari } N - \text{iterasi}}{\text{banyaknya } \lambda_j^{(g)} \text{ yang diperoleh dari } N - \text{iterasi}} \quad (2.49)$$

dengan λ_1 adalah banyaknya distribusi atau model pertama yang dapat dibangkitkan dari N -kali iterasi MCMC untuk SPD dengan dua distribusi penyusun. Sedangkan $\lambda_j^{(g)}$ adalah banyaknya distribusi atau model ke- j , $j=1,2,\dots,m$, yang dapat dibangkitkan dari N -kali iterasi MCMC untuk SPD dengan m -distribusi penyusun.

Selanjutnya Kass dan Raftery (1995) membentuk kategori untuk menyederhanakan interpretasi yang dapat dibuat dari hasil penghitungan Bayes faktor. Kategori tersebut ditabelkan dalam Tabel 2.1 berikut :

Tabel 2.1 : Interpretasi nilai Bayes faktor dalam pemilihan hipotesis

$2 \times \log(Bf_{12})$	Bf_{12}	Bukti kehandalan H_1 dari H_2
< 0	< 1	Negatif
$0 - 2$	$1 - 3$	Tidak ada
$2 - 5$	$3 - 12$	Positif
$5 - 10$	$12 - 150$	Kuat
> 10	> 150	Sangat kuat

BAB III METODOLOGI PENELITIAN

3.1. Bahan Penelitian

Bahan yang digunakan dalam penelitian ini terdiri dari :

1. Buku dan jurnal yang berhubungan dengan topik penelitian
2. Data yang digunakan pada penelitian ini adalah data tingkat inflasi di Indonesia (dalam %) pada periode Januari 2003 – Desember 2005 yang dipublikasikan oleh Bank Indonesia.

3.2. Paket program

Paket program yang digunakan dalam penelitian ini terdiri dari:

1. Minitab 13 digunakan untuk identifikasi data.
2. Maple 6 digunakan untuk membentuk plot densitas data.
3. WinBUGS 1.4 digunakan untuk memperoleh hasil estimasi parameter pada model *mixture* dengan banyak komponen yang ditentukan.
4. Nmix digunakan untuk memperoleh hasil estimasi parameter dan banyaknya komponen *mixture* pada model *mixture* dengan banyak komponen yang tidak diketahui.

3.3. Langkah-langkah Penelitian

Langkah-langkah pada penelitian ini merupakan penjabaran dari tujuan penelitian yang telah disebutkan pada Bab I.

Pertama : Identifikasi awal data berdistribusi *mixture* normal

1. Membuat plot deskriptif data menggunakan *software* MINITAB 13
2. Melakukan *goodness of fit test* pada data dengan menggunakan distribusi normal univariat.

Kedua : Menentukan banyak komponen penyusun *mixture*

Ada dua cara yang digunakan dalam tesis ini untuk menentukan banyak komponen penyusun *mixture* yang mungkin untuk dibentuk dalam distribusi *mixture* normal. Kedua cara tersebut adalah :

1. Menentukan banyak komponen penyusun *mixture* yang mungkin dari model *mixture* normal berdasarkan hasil identifikasi data dengan menggunakan metode histogram.
2. Menganggap banyak komponen penyusun *mixture* tidak diketahui, sehingga untuk mengetahui banyak komponen yang mungkin dari model *mixture* normal dilakukan dengan membuat algoritma RJMCMC. Algoritma ini digunakan untuk mengestimasi banyak komponen model *mixture* normal sekaligus mengestimasi parameter modelnya.

Ketiga : Estimasi parameter model *mixture* normal

Pada tahap ini parameter model *mixture* normal akan diestimasi, baik untuk model *mixture* dengan banyak komponen yang sudah ditentukan

maupun untuk model *mixture* dengan banyak komponen yang tidak diketahui.

1. Estimasi parameter model *mixture* normal dengan banyak komponen ditentukan

Pada tahap ini perlu dilakukan langkah-langkah untuk membentuk model *mixture* normal secara teoritis sebelum melakukan estimasi parameter model, yang terdiri dari :

- a) Menentukan prior independen untuk semua parameter model :

$$\mu_j \sim N(\xi, \kappa^{-1})$$

$$\sigma_j^{-2} | \beta \sim \Gamma(\alpha, \beta) \text{ dengan } \beta \sim \Gamma(g, h) \text{ dan } \mu_1 < \mu_2 < \dots < \mu_k$$

$$w | k \sim D(\delta, \delta, \dots, \delta) \text{ dan } k \sim U(1, 2, \dots, k_{\max})$$

- b) Menggambarkan struktur hirarki model dengan DAG yang akan digunakan sebagai dasar dalam pembuatan doodle untuk mengestimasi parameter model *mixture*.

Selanjutnya langkah-langkah untuk mengestimasi parameter model *mixture* normal :

- a) Membuat struktur doodle dalam WinBUGS 1.4.
- b) Mengestimasi parameter model dengan menggunakan *software* WinBUGS 1.4.
- c) Menggambarkan model *mixture* yang diperoleh dari hasil estimasi menggunakan *software* Maple.
- d) Mengulang langkah a) – c) untuk jumlah komponen yang lain.

- e) Menyajikan model yang didapat ke dalam gambar dengan menggunakan *software* Maple.
2. Estimasi parameter model *mixture* normal dengan banyak komponen tidak diketahui
- Langkah-langkah pada tahap ini terdiri dari :
- a) Membuat algoritma proses iterasi RJMCMC.
 - b) Mengestimasi parameter model *mixture* normal dengan banyak komponen yang tidak diketahui dengan menggunakan *software* Nmix dalam bahasa FORTRAN.
 - c) Menyajikan model *mixture* yang diperoleh ke dalam gambar dengan menggunakan *software* Maple.

Keempat : Membandingkan hasil estimasi model untuk banyak komponen *mixture* yang ditentukan dengan menggunakan SPD dan Bayes faktor

Langkah-langkah pada tahap ini terdiri dari :

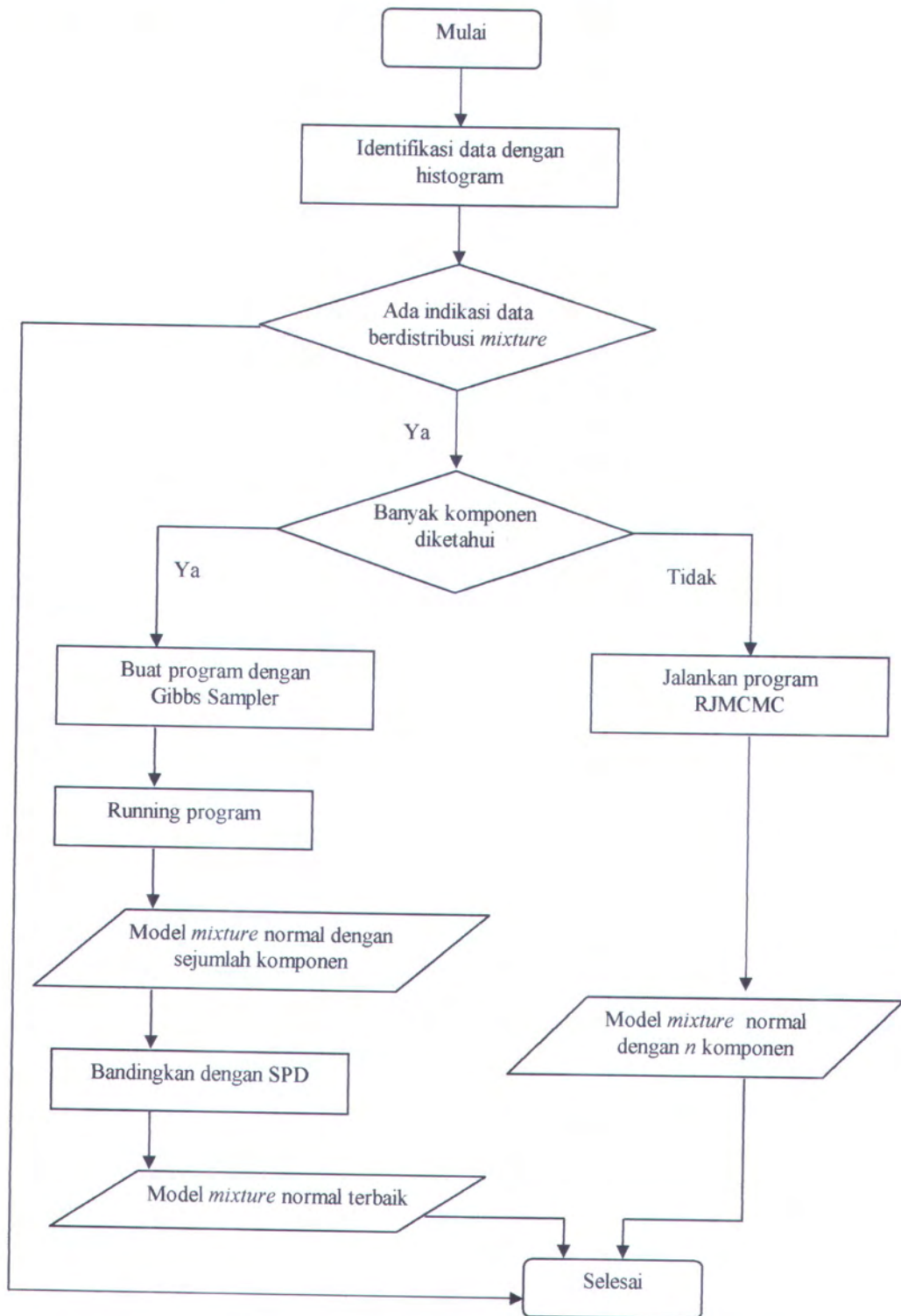
1. Membuat struktur doodle untuk metode SPD dengan beberapa distribusi penyusun.
2. Mengestimasi banyaknya iterasi dari setiap model *mixture* yang dibandingkan.
3. Menghitung Bayes faktor dari model yang dibandingkan untuk mendapatkan model terbaik.
4. Menyajikan model terbaik yang didapat ke dalam gambar dengan menggunakan *software* Maple.

Adapun langkah-langkah pada tahap ini untuk banyak komponen yang tidak diketahui terdiri dari :

1. Membuat algoritma proses iterasi RJMCMC.
2. Mengestimasi parameter model *mixture* dengan banyak komponen *mixture* tidak diketahui dengan menggunakan *software* Nmix dalam bahasa FORTRAN.
3. Menghitung Bayes faktor untuk mendapatkan model yang terbaik yang dihasilkan dari RJMCMC.
4. Menyajikan model terbaik yang didapat ke dalam gambar dengan menggunakan *software* Maple.

Selanjutnya langkah-langkah yang telah disusun tersebut akan disajikan dalam bentuk diagram alir yang disajikan pada Gambar 3.1.

3.4. Diagram Alir Estimasi Distribusi *Mixture* Normal



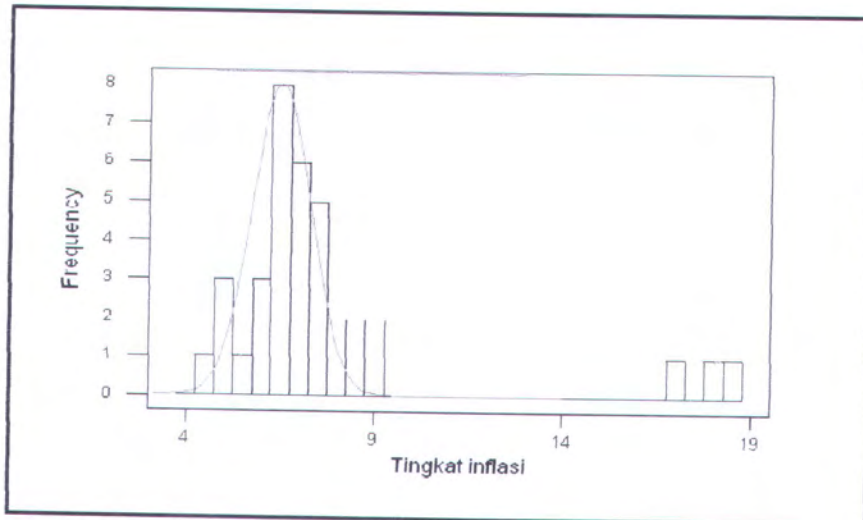
Gambar 3.1 : Diagram alir estimasi distribusi *mixture* normal univariat

BAB IV

HASIL ANALISIS DAN PEMBAHASAN

4.1. Identifikasi dan Deskripsi Data

Berdasarkan langkah-langkah yang sudah ditentukan pada Bab III, maka pembahasan pada bab ini akan dimulai dengan membahas mengenai hasil identifikasi terhadap data yang disajikan pada Lampiran 1. Identifikasi pada data tersebut dilakukan dengan menggunakan metode histogram. Hasil identifikasi ditampilkan pada Gambar 4.1 berikut. Adapun hasil analisis deskriptif terhadap data disajikan pada Tabel 4.1.



Gambar 4.1 : Histogram data tingkat inflasi

Histogram pada Gambar 4.1 menunjukkan bahwa data tingkat inflasi mempunyai kecenderungan berdistribusi *mixture* normal dengan banyak komponen

dua atau tiga atau empat. Hal ini ditunjukkan dengan adanya puncak data (*mode*) yang lebih dari satu.

Tabel 4.1 : Hasil analisis deskriptif pada data tingkat inflasi

Variabel	N	Mean	St dev	Min	Maks
Tingkat inflasi	36	7,75	3,253	4,60	18,38

Selanjutnya untuk mendukung pernyataan di atas, maka identifikasi ini dilengkapi dengan uji *goodness of fit* secara unimodal pada data dengan menggunakan hipotesis berikut :

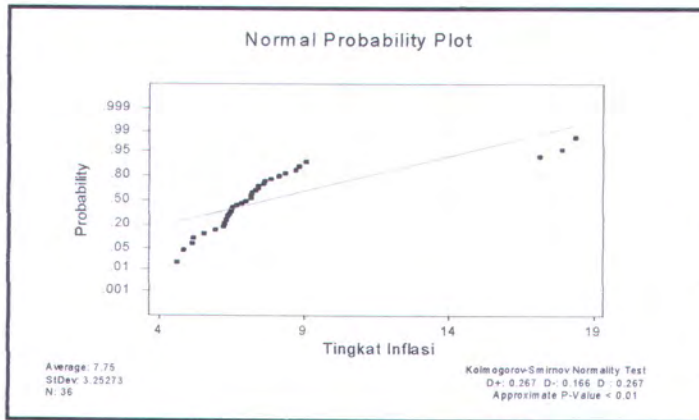
H_0 : Data berdistribusi normal univariat

H_1 : Data tidak berdistribusi normal univariat

Adapun uji yang dilakukan untuk hipotesis di atas menggunakan *Kolmogorov-Smirnov Test*, sehingga statistik uji yang digunakan adalah:

$$D_n = \max \{ D_n^+, D_n^- \}$$

Berdasarkan uji tersebut diperoleh nilai $D_{n=36} = 0,267$ dan $p\text{-value} < 0,010$. Sedangkan berdasarkan tabel statistik uji Kolmogorov pada $\alpha=0,05$ diperoleh nilai $D_{n=36} = 0,221$. Dengan demikian hasil uji ini menunjukkan bahwa data tidak dapat didistribusikan sesuai distribusi normal univariat secara unimodal pada $\alpha=0,05$. Hasil uji ini sesuai dengan plot data pada plot probabilitas normal yang disajikan pada Gambar 4.2.



Gambar 4.2 : Plot probabilitas normal data tingkat inflasi

Berdasarkan hasil identifikasi yang menunjukkan adanya pola multimodal pada data tingkat inflasi, maka pemodelan data tersebut akan dilakukan dengan menggunakan model *mixture* normal sesuai persamaan (2.5). Pembahasan mengenai pembentukan distribusi *mixture* normal univariat diberikan pada subbab 2.6. Selanjutnya distribusi bersyarat penuh untuk setiap parameter model *mixture* normal pada persamaan (2.17) hingga persamaan (2.21) akan digunakan untuk mengestimasi parameter model dengan menggunakan algoritma Gibbs Sampler.

4.2. Analisis Bayesian pada Data

Pembahasan mengenai analisis Bayesian pada data tingkat inflasi akan dibedakan dalam dua bahasan, yaitu pemodelan *mixture* normal dengan banyak komponen ditentukan dan pemodelan *mixture* normal dengan banyak komponen tidak diketahui dan tidak ditentukan. Pada pemodelan *mixture* normal baik dengan banyak komponen ditentukan maupun banyak komponen tidak ditentukan, hasil estimasi modelnya akan dibandingkan untuk menentukan model *mixture* normal yang paling sesuai dengan data tingkat inflasi.

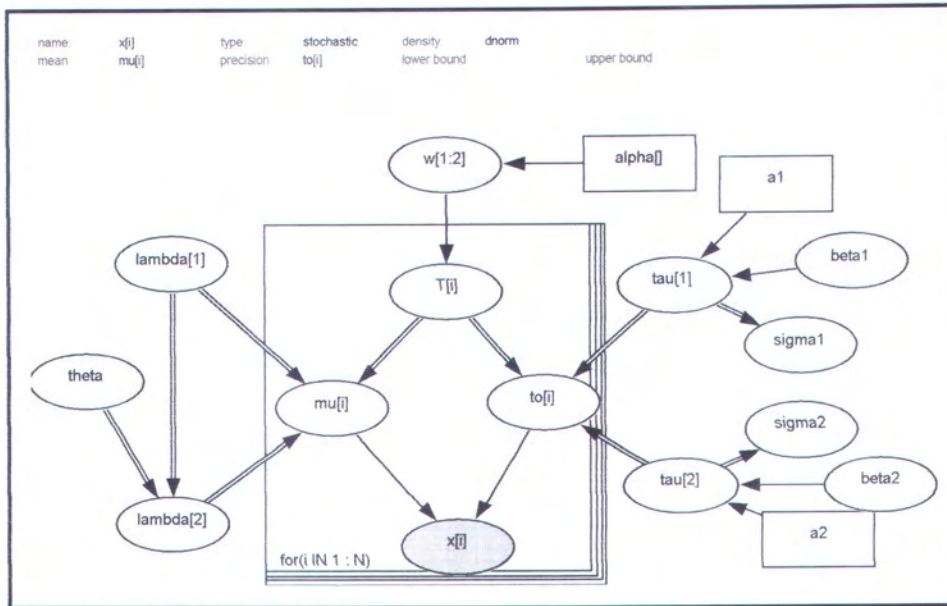
4.2.1. Banyak Komponen *Mixture* Ditentukan

Pemodelan data menggunakan model *mixture* normal univariat dengan banyak komponen ditentukan akan dilakukan dengan menggunakan *software* WinBUGS 1.4. Langkah pertama yang akan dilakukan dalam pemodelan ini adalah membuat doodle yang menunjukkan hubungan berarah antar variabel pada model *mixture* normal yang digambarkan seperti pada pembuatan DAG. Selanjutnya pada pembahasan ini juga akan dilengkapi dengan program pemodelan *mixture* beserta data dan inisialisasi dari iterasi yang akan dilakukan dengan menggunakan metode Gibbs Sampler.

4.2.1.1. Pembuatan Doodle

Pembuatan doodle menggunakan WinBUGS 1.4 akan mempermudah proses estimasi model *mixture* yang menggunakan analisis Bayesian dan metode MCMC. Hal ini didukung dengan adanya fasilitas *automatic programming* yang diberikan oleh WinBUGS sebagai bentuk representasi dari simbolik OOP (*Object Oriented Programming*).

Hasil pembentukan simbolik OOP pada pemodelan *mixture* normal dengan banyak komponen dua diberikan dalam bentuk doodle pada Gambar 4.3. Sifat stokastik dalam MCMC yang mengimplementasikan struktur hirarki dari analisis Bayesian digambarkan dalam node ellips dengan garis penghubung berupa garis tunggal. Sedangkan node ellips dengan garis penghubung berupa garis ganda menggambarkan hubungan yang bersifat *logical*. Selain itu dimungkinkan dalam struktur hirarki Bayesian diakhiri dengan node kotak yang merepresentasikan suatu konstanta.



Gambar 4.3 : Doodle model *mixture* normal dengan dua komponen

4.2.1.2. Program WinBUGS

Hasil estimasi tiap parameter pada model *mixture* normal dengan dua komponen dapat diperoleh jika struktur doodle pada Gambar 4.3 telah diterjemahkan ke dalam program WinBUGS. Program tersebut ditampilkan pada Program 4.1. Selanjutnya untuk memenuhi data dan nilai awal jalannya iterasi yang diperlukan oleh program WinBUGS maka program tersebut dilengkapi dengan masukan data yang dituliskan dalam Program 4.2.

Program 4.1 : Program WinBUGS untuk estimasi model *mixture* dengan dua komponen *mixture*

```

model;
{
  w[1:2] ~ ddirch(alpha[])
  for(i in 1 : N) {
    T[i] ~ dcat(w[1:2])
  }
  for(i in 1 : N) {
    x[i] ~ dnorm(mu[i],to[i])
  }
  for(i in 1 : N) {
    mu[i] <- lambda[T[i]]
  }
  for(i in 1 : N) {
    to[i] <- tau[T[i]]
  }
  lambda[1] ~ dnorm( 3.0,0.05)
  lambda[2] <- lambda[1] + theta
  theta ~ dnorm( 5.0,0.05)I( 0.0,)
  tau[1] ~ dgamma(a1,beta1)
  tau[2] ~ dgamma(a2,beta2)
  beta1 ~ dgamma( 0.2,0.05)
  sigma1 <- 1 / tau[1]
  sigma2 <- 1 / tau[2]
  beta2 ~ dgamma( 0.2,0.05)
}

```

Program 4.2 : Data dan nilai awal dalam pemodelan *mixture* dengan dua komponen *mixture*

```

DATA
list(x=c(4.6,4.82,5.11,5.16,5.53,5.92,6.18,6.22,6.27,6.27,6.33,6.4,
6.47,6.48,6.51,6.67,6.83,6.98,7.15,7.15,7.17,7.2,7.32,7.4,7.42,
7.6,7.62,7.84,8.12,8.33,8.68,8.81,9.06,17.11,17.89,18.38),N=36,
alpha=c(1,1),a1=2,a2=2,
T=c(1,1,1,1,1,1,1,1,1,1,
1,1,1,1,1,1,1,1,1,1,
1,1,1,1,1,1,1,1,1,1,
1,1,1,2,2,2))

INITIAL
list(lambda=c(3.0,NA),beta1=2,beta2=2,theta=5.0)

```

Pembuatan doodle, program WinBUGS serta data dan nilai awal iterasi pada pemodelan *mixture* normal dengan tiga dan empat komponen *mixture* diberikan pada Lampiran 2A dan 2B.

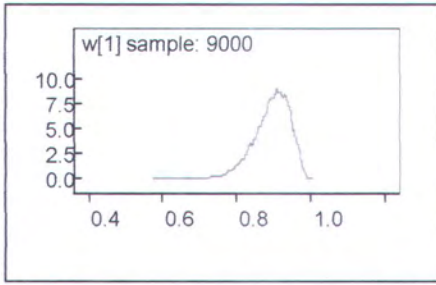
4.2.1.3. Estimasi Komponen-Komponen *Mixture*

Estimasi parameter model *mixture* dilakukan dengan menjalankan iterasi sebanyak 10000 iterasi dengan kondisi *burn in* sebanyak 1000 iterasi. Jadi sampel yang digunakan untuk mengestimasi parameter model diperoleh dari 9000 iterasi. Hasil estimasi parameter model *mixture* normal dengan dua komponen *mixture* ditampilkan pada *Output 4.1*.

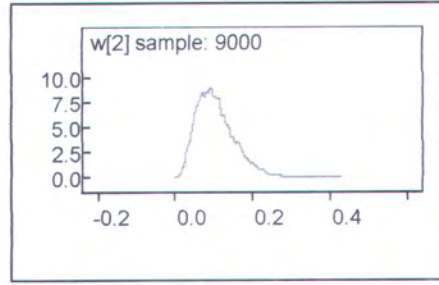
Output 4.1 : Hasil estimasi model *mixture* normal dengan dua komponen *mixture*

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
lambda[1]	6.839	0.1988	0.002448	6.445	6.839	7.234	1001	9000
lambda[2]	17.65	0.7222	0.01137	15.91	17.72	18.9	1001	9000
tau[1]	0.8113	0.2039	0.00257	0.4679	0.7918	1.266	1001	9000
tau[2]	2.218	2.196	0.0528	0.1101	1.538	8.196	1001	9000
w[1]	0.8949	0.04891	4.918E-4	0.781	0.9018	0.9698	1001	9000
w[2]	0.1051	0.04891	4.918E-4	0.03022	0.09823	0.2192	1001	9000

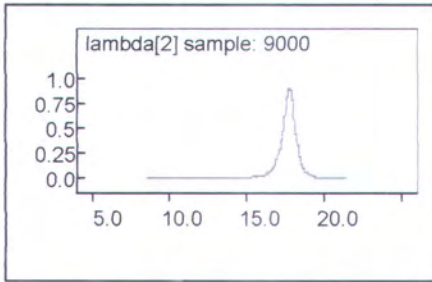
Program WinBUGS juga memberikan hasil estimasi berupa plot densitas tiap parameter yang ditampilkan dalam Gambar 4.4 berikut.



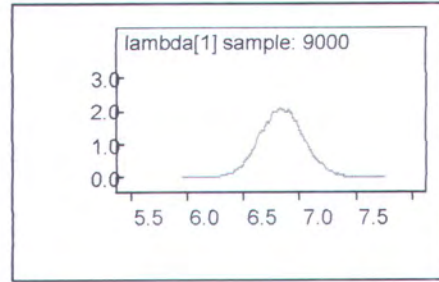
(a)



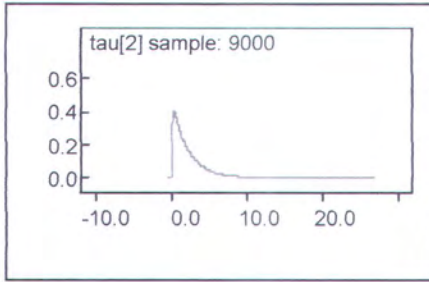
(b)



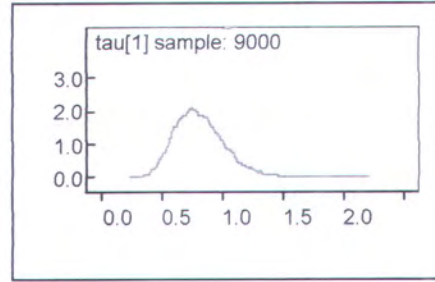
(c)



(d)



(e)



(f)

Gambar 4.4 : Plot estimasi densitas parameter model *mixture* normal univariat dengan dua komponen *mixture*

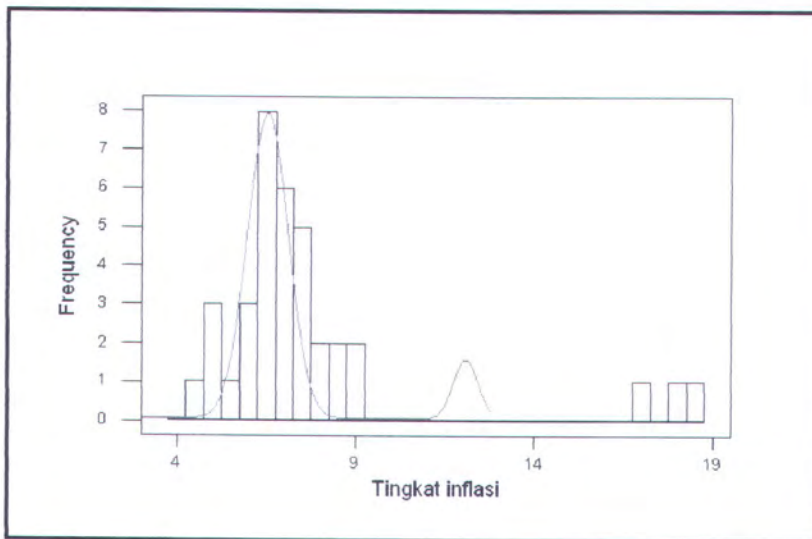
- (a) Plot estimasi densitas parameter pembobot komponen pertama
- (b) Plot estimasi densitas parameter pembobot komponen kedua
- (c) Plot estimasi densitas parameter lokasi komponen pertama
- (d) Plot estimasi densitas parameter lokasi komponen kedua
- (e) Plot estimasi densitas parameter penyebaran komponen pertama
- (f) Plot estimasi densitas parameter penyebaran komponen kedua

Berdasarkan hasil estimasi parameter model *mixture* di atas, maka dapat disimpulkan bahwa pemodelan data tingkat inflasi menggunakan model *mixture* normal univariat dengan dua komponen penyusun *mixture* adalah :

$$X_i \sim 0,8949 N(6,839;1,233) + 0,1051 N(17,65;0,45) \quad (4.1)$$

Model *mixture* di atas menunjukkan bahwa 89,49% data menyebar di kelompok pertama yang berdistribusi normal dengan mean = 6,839 dan variansi sebesar 1,233. Sedangkan sisanya sebesar 10,51% menyebar di kelompok kedua dengan mean = 17,65 dan variansi sebesar 0,45. Oleh karena itu dapat dikatakan bahwa model *mixture* di atas didominasi oleh kelompok pertama dengan interval data kurang dari 10%.

Selanjutnya pemodelan tersebut ditampilkan pada Gambar 4.5 dengan menggambarkan plot densitas *mixture* yang diperoleh dari hasil estimasi model terhadap data yang digambarkan dengan histogram.



Gambar 4.5 : Plot distribusi *mixture* normal dengan dua komponen *mixture*

Gambar di atas menunjukkan bahwa pendekatan model *mixture* normal pada data yang terindikasi bersifat multimodal mampu mengurangi ketidakcocokan yang disebabkan penggunaan pendekatan satu distribusi seperti pada Gambar 4.1.

Pemodelan *mixture* dengan tiga komponen *mixture* juga dilakukan dengan menjalankan iterasi sebanyak 10000 iterasi dengan kondisi *burn in* 1000 iterasi, sehingga sampel yang digunakan untuk mengestimasi model *mixture* ini sebanyak 9000 iterasi. Hasil estimasi parameternya ditampilkan pada *Output* 4.2 berikut.

Output 4.2 : Hasil estimasi parameter model *mixture* dengan tiga komponen *mixture*

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
lambda[1]	5.18	0.2172	0.005454	4.757	5.179	5.619	1001	9000
lambda[2]	7.212	0.1636	0.002078	6.896	7.211	7.534	1001	9000
lambda[3]	17.74	0.4016	0.00417	16.87	17.76	18.52	1001	9000
tau[1]	4.66	2.715	0.04301	1.049	4.118	11.3	1001	9000
tau[2]	1.419	0.394	0.005233	0.7743	1.38	2.308	1001	9000
tau[3]	3.237	2.497	0.04175	0.456	2.584	9.785	1001	9000
w[1]	0.1797	0.0605	6.819E-4	0.07765	0.1744	0.3124	1001	9000
w[2]	0.7166	0.07225	7.19E-4	0.5655	0.72	0.8452	1001	9000
w[3]	0.1037	0.04911	5.315E-4	0.02981	0.09614	0.2173	1001	9000

Model *mixture* normal univariat yang terbentuk dari hasil estimasi di atas adalah:

$$X_i \sim 0,1797 N(5,18;0,4632) + 0,7166 N(7,212;0,8395) + 0,1037 N(17,74;0,558) \tag{4.2}$$

Model pada persamaan (4.2) menunjukkan bahwa model *mixture* dengan tiga komponen penyusun *mixture* masih didominasi oleh kelompok kedua dengan interval data antara 6% hingga 10%, yaitu sebesar 71,66%. Masing-masing komponen berdistribusi normal dengan mean dan variansi berturut-turut: komponen pertama 5,18 dan 0,2146; komponen kedua 7,212 dan 0,7047; komponen ketiga 17,74 dan 0,3089. Hasil estimasi tersebut ditampilkan dalam plot distribusi *mixture* pada Gambar 4.6.

Pemodelan *mixture* dengan empat komponen *mixture* juga dilakukan dengan menjalankan 10000 iterasi dengan kondisi *burn in* sebanyak 1000 iterasi, sehingga sampel yang digunakan untuk mengestimasi parameter model *mixture* tersebut sebanyak 9000 iterasi. Hasil estimasi pemodelan tersebut ditampilkan pada *Output 4.3* berikut.

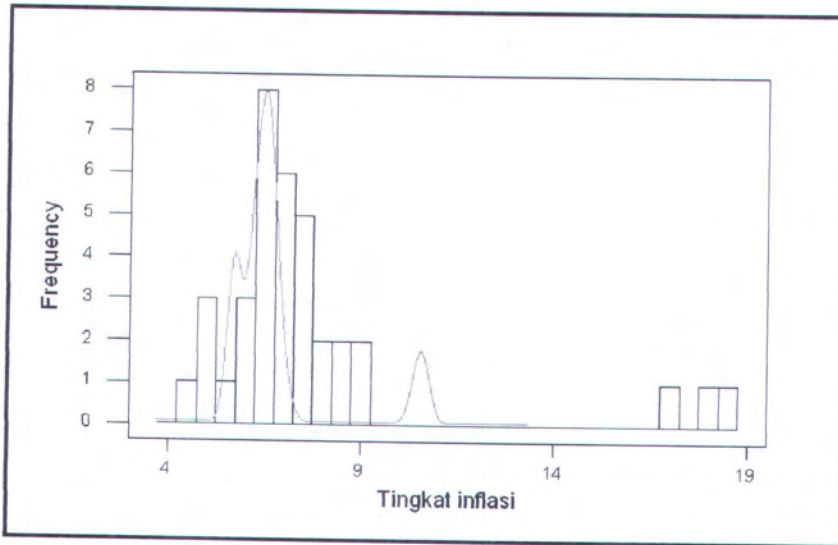
Output 4.3 : Hasil estimasi parameter model *mixture* dengan empat komponen *mixture*

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
lambda[1]	5.158	0.2208	0.00623	4.693	5.163	5.582	1001	9000
lambda[2]	7.2	0.1678	0.002392	6.867	7.201	7.535	1001	9000
lambda[3]	13.35	0.8118	0.02611	11.82	13.33	14.91	1001	9000
lambda[4]	18.18	0.3847	0.007666	17.52	18.15	18.94	1001	9000
tau[1]	4.647	2.729	0.04245	1.003	4.113	11.36	1001	9000
tau[2]	1.413	0.3931	0.004714	0.7607	1.379	2.284	1001	9000
tau[3]	0.08438	0.2629	0.007554	0.001374	0.03322	0.4269	1001	9000
tau[4]	13.07	12.95	0.2351	0.8408	8.893	49.11	1001	9000
w[1]	0.1749	0.06045	6.685E-4	0.07459	0.1688	0.3104	1001	9000
w[2]	0.7007	0.07184	8.646E-4	0.5534	0.7033	0.8301	1001	9000
w[3]	0.04925	0.03362	3.585E-4	0.006399	0.04213	0.1337	1001	9000
w[4]	0.07514	0.04114	4.49E-4	0.01608	0.06833	0.1739	1001	9000

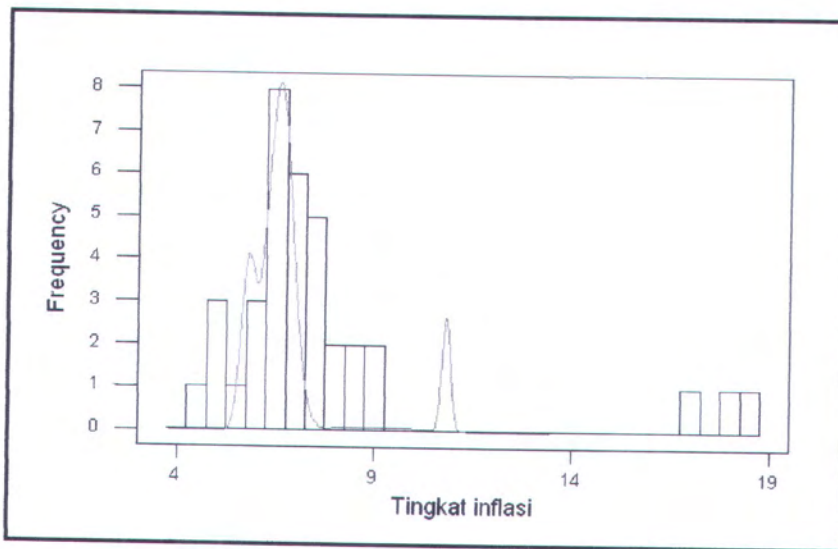
Model *mixture* normal univariat yang terbentuk dari hasil estimasi di atas adalah:

$$X_i \sim 0,1749 N(5,158;0,4639) + 0,7007 N(7,2;0,8412) + 0,04925 N(13,35;3,4425) + 0,07514 N(18,18;0,2766) \quad (4.3)$$

Model pada persamaan (4.3) menunjukkan bahwa model *mixture* dengan empat komponen juga masih didominasi oleh kelompok kedua dengan interval data antara 6% hingga 10%, yaitu sebesar 70,77%. Masing-masing komponen berdistribusi normal dengan mean dan variansi berturut-turut : komponen pertama 5,158 dan 0,2152; komponen kedua 7,2 dan 0,7077; komponen ketiga 13,35 dan 11,851; komponen keempat 18,18 dan 0,0765. Hasil estimasi tersebut ditampilkan pada Gambar 4.7.



Gambar 4.6 : Plot distribusi *mixture* normal dengan tiga komponen *mixture*



Gambar 4.7 : Plot distribusi *mixture* normal dengan empat komponen *mixture*

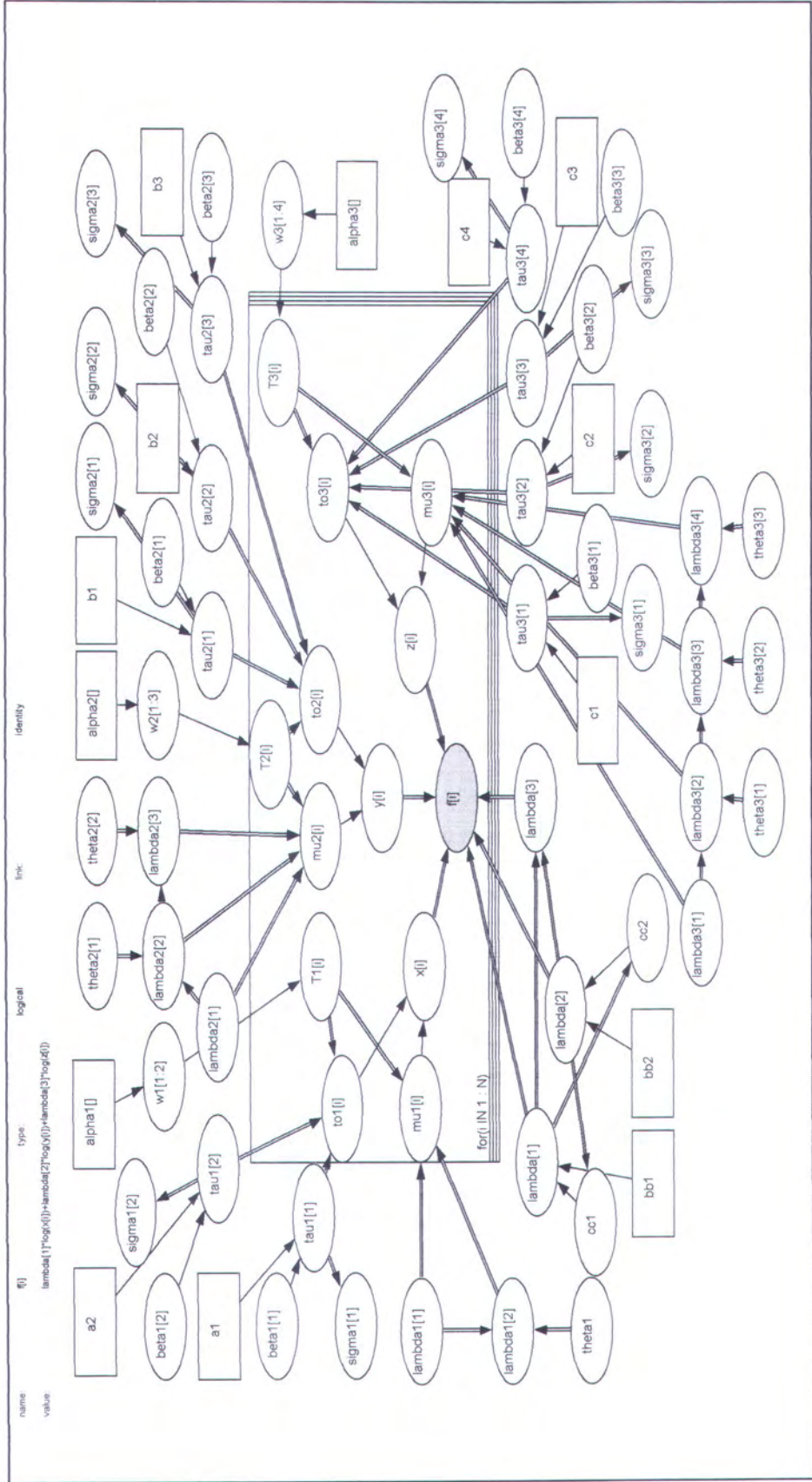
4.2.1.4. Pemilihan Model Terbaik Menggunakan SPD

Hasil pemodelan data menggunakan model *mixture* normal dengan banyak komponen dua, tiga dan empat menunjukkan bahwa ketiga model tersebut sesuai untuk data tingkat inflasi. Namun demikian pada penelitian ini ketiga model tersebut akan dibandingkan untuk mengetahui model yang paling sesuai untuk data tersebut. Pemilihan model dilakukan dengan menggunakan metode SPD dan Bayes faktor.

Sebagaimana pemodelan *mixture* yang sudah dilakukan pada tahap sebelumnya, pemilihan model terbaik juga dilakukan dengan menyusun doodle dari model-model distribusi yang akan dibandingkan. Struktur doodle dari ketiga model yang akan dibandingkan, yaitu: model *mixture* normal dengan dua, tiga dan empat komponen *mixture*, ditampilkan pada Gambar 4.8.

Program WinBUGS dari doodle pada Gambar 4.8 ditampilkan pada Program 4.3. Sedangkan program yang berisi masukan data dan nilai awal yang akan digunakan dalam proses iterasi ditampilkan pada Program 4.4.

Program 4.3 dan Program 4.4 juga dijalankan sebanyak 10000 iterasi dengan kondisi *burn in* sebanyak 1000 iterasi, sehingga sampel yang digunakan untuk mengestimasi parameter λ dalam fungsi SPD sebanyak 9000 iterasi. Hasil estimasi tersebut ditampilkan pada *Output* 4.4.



Gambar 4.8 : Doodle SPD dengan tiga distribusi penyusun untuk data tingkat inflasi

Program 4.3 : Program WinBUGS untuk SPD dengan tiga distribusi penyusun

```
model;
{
  for(i in 1 : N) {
    f[i] <- lambda[1] * log(x[i]) + lambda[2] * log(y[i]) +
      lambda[3] * log(z[i])
  }
  for(i in 1 : N) {
    x[i] ~ dnorm(mu1[i],to1[i])
  }
  for(i in 1 : N) {
    y[i] ~ dnorm(mu2[i],to2[i])
  }
  for(i in 1 : N) {
    z[i] ~ dnorm(mu3[i],to3[i])
  }
  for(i in 1 : N) {
    mu1[i] <- lambda1[T1[i]]
  }
  for(i in 1 : N) {
    to1[i] <- tau1[T1[i]]
  }
  for(i in 1 : N) {
    T1[i] ~ dcat(w1[1:2])
  }
  w1[1:2] ~ ddirch(alpha1[])
  lambda1[1] ~ dnorm( 3.0,0.05)
  lambda1[2] <- lambda1[1] + theta1
  theta1 ~ dnorm( 5.0,0.05)I(0.0)
  sigma1[1] <- 1 / tau1[1]
  beta1[1] ~ dgamma( 0.2,0.05)
  tau1[1] ~ dgamma(a1,beta1[1])
  tau1[2] ~ dgamma(a2,beta1[2])
  beta1[2] ~ dgamma( 0.2,0.05)
  sigma1[2] <- 1 / tau1[2]
  for(i in 1 : N) {
    mu2[i] <- lambda2[T2[i]]
  }
  for(i in 1 : N) {
    to2[i] <- tau2[T2[i]]
  }
  for(i in 1 : N) {
    T2[i] ~ dcat(w2[1:3])
  }
  w2[1:3] ~ ddirch(alpha2[])
  lambda2[1] ~ dnorm( 4.0, 0.6)
  lambda2[2] <- lambda2[1] + theta2[1]
  theta2[1] ~ dnorm( 2.0,0.12)I(0.0)
  lambda2[3] <- lambda2[2] + theta2[2]
  theta2[2] ~ dnorm(6, 0.6)I(0.0)
  tau2[1] ~ dgamma(b1,beta2[1])
  tau2[2] ~ dgamma(b2,beta2[2])
  sigma2[1] <- 1 / tau2[1]
  tau2[3] ~ dgamma(b3,beta2[3])
}
```



```

sigma2[2] <- 1 / tau2[2]
for(i in 1 : N) {
  mu3[i] <- lambda3[T3[i]]
}
for(i in 1 : N) {
  to3[i] <- tau3[T3[i]]
}
for(i in 1 : N) {
  T3[i] ~ dcat(w3[1:4])
}
w3[1:4] ~ ddirch(alpha3[])
beta2[1] ~ dgamma(0.2, 0.6)
beta2[2] ~ dgamma(0.2, 0.12)
sigma2[3] <- 1 / tau2[3]
beta2[3] ~ dgamma(0.2, 0.6)
tau3[1] ~ dgamma(c1, beta3[1])
tau3[2] ~ dgamma(c2, beta3[2])
tau3[3] ~ dgamma(c3, beta3[3])
tau3[4] ~ dgamma(c4, beta3[4])
beta3[1] ~ dgamma(0.2, 0.6)
beta3[2] ~ dgamma(0.2, 0.12)
beta3[4] ~ dgamma(0.2, 2)
sigma3[4] <- 1 / tau3[4]
sigma3[2] <- 1 / tau3[2]
sigma3[1] <- 1 / tau3[1]
sigma3[3] <- 1 / tau3[3]
beta3[3] ~ dgamma(0.2, 0.001)
lambda[1] ~ dunif(bb1, cc1)
lambda[2] ~ dunif(bb2, cc2)
lambda[3] <- 1 - lambda[1] - lambda[2]
cc1 <- 1 - lambda[2]
cc2 <- 1 - lambda[1]
lambda3[1] ~ dnorm(4.0, 0.6)
lambda3[2] <- lambda3[1] + theta3[1]
lambda3[3] <- lambda3[2] + theta3[2]
lambda3[4] <- lambda3[3] + theta3[3]
theta3[1] ~ dnorm(2.0, 0.12)I(0.0,)
theta3[2] ~ dnorm(5.0, 0.001)I(0.0,)
theta3[3] ~ dnorm(5.0, 2.0)I(0.0,)
}

```


Berdasarkan hasil estimasi yang ditampilkan pada *Output* 4.4, maka fungsi SPD yang terbentuk dari tiga distribusi penyusun yang terdiri dari: distribusi pertama yang menotasikan model *mixture* normal dengan dua komponen *mixture*, distribusi kedua menotasikan model *mixture* normal dengan tiga komponen *mixture* dan distribusi keempat yang menotasikan model *mixture* normal dengan empat komponen *mixture*, dinyatakan dalam persamaan berikut.

$$f_{\text{SPD}}(x, \lambda, w, \mu, \sigma) = C (f_{\text{mix}2})^{0,3365} (f_{\text{mix}3})^{0,3327} (f_{\text{mix}4})^{0,3308} \quad (4.10)$$

Selanjutnya penghitungan Bayes faktor dari satu distribusi penyusun SPD terhadap distribusi lainnya dilakukan dengan menghitung iterasi dari tiap distribusi penyusun yang mendominasi SPD. Hasil iterasi tersebut adalah $\lambda_1 = 3029$, $\lambda_2 = 2994$ dan $\lambda_3 = 2977$ dari sebanyak 9000 iterasi. Dengan menggunakan persamaan (2.43) diperoleh Bayes faktor antara model *mixture* dengan dua komponen *mixture* terhadap model *mixture* dengan tiga komponen *mixture* adalah $Bf_{12} = 1,012$; sedangkan Bayes faktor antara model *mixture* dengan dua komponen *mixture* terhadap model *mixture* dengan empat komponen *mixture* adalah $Bf_{13} = 1,0175$; dan Bayes faktor antara model *mixture* dengan tiga komponen *mixture* terhadap model *mixture* dengan empat komponen *mixture* adalah $Bf_{23} = 1,0057$.

Berdasarkan kategori Bayes faktor yang ditampilkan pada Tabel 2.1, maka keseluruhan nilai Bayes faktor di atas berada pada interval 1 sampai 3. Dengan demikian dapat dikatakan bahwa tidak ada model yang mendominasi fungsi SPD pada persamaan (4.10), sehingga ketiga model tersebut dapat digunakan untuk memodelkan data tingkat inflasi.

4.2.1. Banyak Komponen *Mixture* Tidak Ditentukan

Pemodelan *mixture* dengan banyak komponen tidak diketahui dan tidak ditentukan dilakukan dengan menggunakan metode RJMCMC. Keuntungan dari penggunaan metode ini adalah dapat memodelkan suatu *mixture* secara bersama-sama dengan banyaknya komponen penyusun *mixture*. Sebagaimana penjelasan pada Bab II, maka penggunaan metode ini pada pemodelan *mixture* dilakukan dengan membuat algoritma RJMCMC terlebih dahulu.

4.2.1.1. Algoritma RJMCMC

Algoritma RJMCMC merupakan algoritma dengan tipe pergerakan dari enam langkah yang ditampilkan pada algoritma berikut.

Algoritma 4.1

1. *update* w
2. *update* θ dimana $\theta = (\mu, \sigma)$
3. *update* z
4. *update* β
5. *split/merge* komponen-komponen *mixture*
6. *birth/death* komponen-komponen kosong

Selanjutnya Algoritma 4.1 di atas dijabarkan dalam suatu *software* yaitu *Nmix*. *Software* ini ditulis dalam bahasa Fortran dan C dengan struktur program yang terdiri dari program utama dan *subroutine*. Program utama merupakan program umum dari

proses *reversible jump*. Sedangkan *subroutine* yang mendukung program utama terdiri dari tiga *subroutine*, dimana masing-masing *subroutine* berupa program yang terkait dengan distribusi prior parameter model *mixture* normal, yaitu distribusi normal dan gamma.

4.2.1.2. Hasil Iterasi dengan Algoritma RJMCMC

Program penjabaran dari Algoritma 4.1 di atas dijalankan sebanyak 10000 kali. Hasil iterasi tersebut menunjukkan bahwa data tingkat inflasi dapat dimodelkan dengan satu hingga sepuluh komponen. Masing-masing model yang terbentuk dengan satu hingga sepuluh komponen mempunyai banyak iterasi dominan yang berbeda-beda dari keseluruhan 10000 iterasi. Banyaknya iterasi dari tiap model ditampilkan pada Tabel 4.2 berikut.

Tabel 4.2 : Banyak iterasi dari tiap model *mixture* normal dengan RJMCMC

No	Banyak komponen	Banyak iterasi dominan
1	1	5
2	2	5670
3	3	2460
4	4	1042
5	5	450
6	6	222
7	7	87
8	8	42
9	9	16
10	10	6

Hasil iterasi yang ditampilkan pada Tabel 4.2 mengindikasikan bahwa estimasi distribusi *mixture* normal dengan dua komponen merupakan model terbaik karena banyak iterasi dominan sebanyak 5670 kali iterasi. Namun demikian, dugaan tersebut perlu didukung dengan menghitung Bayes faktor dari model kedua yang dibandingkan dengan model lainnya. Hasil penghitungan Bayes faktor tersebut ditampilkan pada Tabel 4.3 berikut.

Tabel 4.3 : Hasil penghitungan Bayes faktor dari model kedua terhadap model lainnya

No	Bayes faktor		No	Bayes faktor	
1	Bf_{21}	1134	6	Bf_{27}	65,17
2	Bf_{23}	2,30	7	Bf_{28}	135
3	Bf_{24}	5,44	8	Bf_{29}	354,375
4	Bf_{25}	12,6	9	Bf_{210}	945
5	Bf_{26}	25,54			

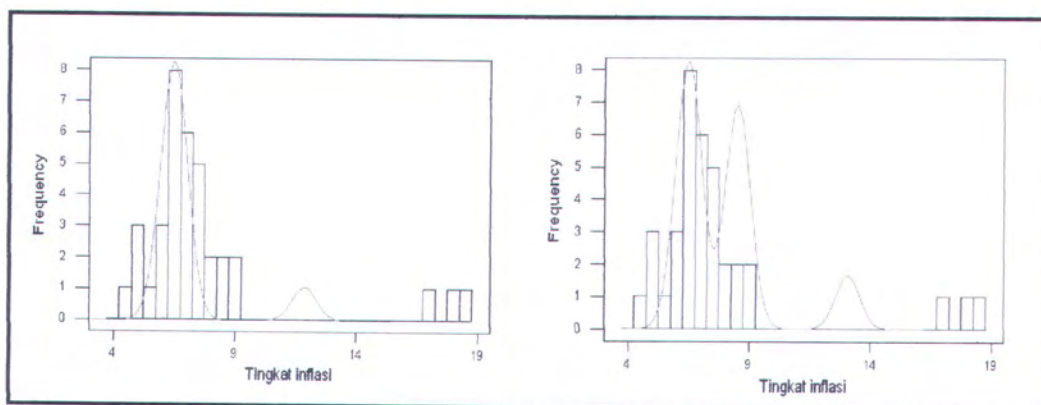
Berdasarkan kategori Bayes faktor pada Tabel 2.1 dan hasil penghitungan Bayes faktor pada Tabel 4.3 menunjukkan bahwa model dengan banyak komponen dua merupakan model yang paling baik dengan nilai Bayes faktor terhadap model dengan banyak komponen satu mencapai 1134. Namun demikian model dengan banyak komponen tiga tidak terdominasi oleh model dengan banyak komponen dua. Oleh karena itu pemodelan *mixture* terhadap data tingkat inflasi direkomendasikan untuk dimodelkan menggunakan model *mixture* dengan banyak komponen dua dan tiga. Selanjutnya berdasarkan hasil iterasi RJMCMC dan nilai Bayes faktor di atas, maka model terbaik yang dapat digunakan untuk memodelkan data tingkat inflasi adalah

model dengan banyak komponen dua dan tiga. Hasil estimasi model-model *mixture* tersebut disajikan pada Tabel 4.4 berikut.

Tabel 4.4 : Estimasi parameter model *mixture* normal dengan RJMCMC

Banyak komponen	Bobot	Mean	Standar deviasi
Dua	0,89441	6,83646	1,12357
	0,10559	17,77451	1,02457
Tiga	0,50043	5,24695	1,09720
	0,40751	9,37368	1,07282
	0,09206	18,62684	1,01025

Selanjutnya berdasarkan hasil estimasi di atas maka representasi dari pemodelan tersebut disajikan pada Gambar 4.9 berikut.



Gambar 4.9 : Plot distribusi *mixture* normal dengan banyak komponen *mixture* dua dan tiga yang diestimasi dengan RJMCMC

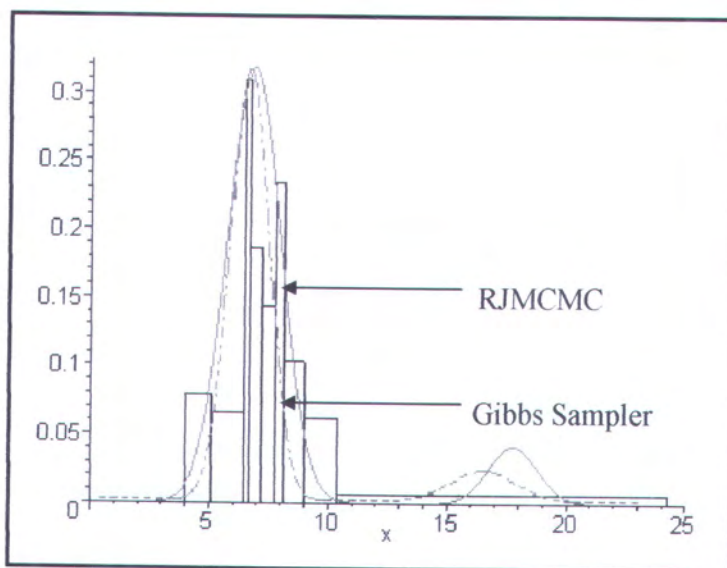


4.4. Pembahasan Pemilihan Model Terbaik

Berdasarkan hasil penghitungan Bayes faktor baik pada model *mixture* normal dengan banyak komponen ditentukan maupun tidak diketahui, dapat dinyatakan bahwa model-model yang diperoleh dari estimasi dapat digunakan untuk memodelkan data tingkat inflasi. Namun demikian, apabila ditinjau dari segi ilmu ekonomi mengenai inflasi, maka model terbaik yang digunakan untuk memodelkan data inflasi adalah model *mixture* normal univariat dengan dua komponen. Hal ini sesuai dengan penjelasan Boediono (1982) mengenai kategori inflasi. Dengan demikian model *mixture* yang diperoleh melalui pemodelan dengan banyak komponen *mixture* yang ditentukan adalah model *mixture* dengan komponen pertama berdistribusi normal univariat dengan nilai mean sama dengan 6,839 dan variansi sebesar 0,8113 pada bobot 0,8949 serta komponen kedua berdistribusi normal dengan mean sebesar 5,18 dan variansi sebesar 6,501 pada bobot 0,1051. Sedangkan model *mixture* yang diperoleh dari pemodelan *mixture* dengan banyak komponen tidak diketahui diperoleh model *mixture* dengan komponen pertama berdistribusi normal univariat dengan nilai mean sebesar 6,83646 dan variansi 1,2624 pada bobot 0,89441 serta komponen kedua berdistribusi normal univariat dengan mean sebesar 17,77451 dan variansi 1,0497 pada bobot 0,10559.

Kedua model yang dipilih di atas sesuai dengan penjelasan Boediono (1982) mengenai kategori inflasi. Boediono (1982) mengkategorikan inflasi menjadi empat macam, yaitu: inflasi ringan dengan tingkat inflasi di bawah 10%, inflasi sedang dengan tingkat inflasi antara 10% – 30%, inflasi berat dengan tingkat inflasi antara 30%- 100%

dan hiperinflasi dengan tingkat inflasi lebih dari 100%. Dengan demikian data inflasi yang digunakan pada penelitian ini terdiri dari inflasi dengan kategori ringan dan sedang. Selanjutnya pemodelan yang diperoleh dari kedua metode yang digunakan dalam penelitian ini digambarkan melalui plot distribusi *mixture* terhadap data tingkat inflasi. Plot distribusi tersebut ditampilkan pada Gambar 4.10 berikut.



Gambar 4.10 : Plot bersama dua densitas *mixture* dengan dua komponen yang diestimasi dengan Gibbs Sampler dan RJMCMC

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil analisis dan pembahasan mengenai pemodelan *mixture* normal univariat pada data tingkat inflasi di Indonesia, maka beberapa kesimpulan yang dihasilkan dari penelitian ini adalah:

1. Hasil identifikasi data menggunakan histogram mengindikasikan data bersifat multimodal. Hal ini ditunjukkan dengan adanya puncak data (*mode*) dari tiap subpopulasi yang lebih dari satu. Kondisi ini juga tampak pada hasil pengujian data secara unimodal menggunakan *Kolmogorov Smirnov Test*. Oleh karena itu pemodelan pada data jenis ini dilakukan dengan menggunakan model *mixture* normal.
2. Pada pemodelan *mixture*, penentuan banyaknya komponen penyusun *mixture* ditentukan berdasarkan hasil identifikasi data menggunakan histogram. Histogram data menunjukkan bahwa data dapat disusun dari dua atau tiga atau empat subpopulasi, dimana setiap subpopulasi merupakan penyusun *mixture*.
3. Pada pemodelan *mixture* normal univariat dengan banyak komponen *mixture* tidak diketahui secara pasti, estimasi parameter modelnya dilakukan dengan menggunakan dua pendekatan, yaitu: pertama, dengan menganggap banyak komponen ditentukan dan kedua, dengan menganggap banyak komponen tidak diketahui. Estimasi parameter model *mixture* berdasarkan pendekatan pertama dilakukan

dengan menggunakan metode Gibbs Sampler, sedangkan estimasi parameter model *mixture* berdasarkan pendekatan kedua dilakukan dengan menggunakan metode Reversible Jump MCMC.

Estimasi model *mixture* dengan menggunakan metode Gibbs Sampler menghasilkan tiga model *mixture* dengan banyak komponen dua, tiga dan empat. Sedangkan estimasi model *mixture* dengan menggunakan metode Reversible Jump MCMC menghasilkan model *mixture* dengan banyak komponen satu hingga sepuluh.

4. Penentuan model terbaik untuk data dilakukan dengan menggunakan metode Struktur Perkalian Distribusi (SPD) dan Bayes Faktor.

Hasil pemilihan model terbaik untuk data berdasarkan hasil estimasi model menggunakan metode Gibbs Sampler menunjukkan bahwa ketiga model yang dihasilkan tersebut dapat digunakan untuk memodelkan data karena tidak ada model yang mendominasi fungsi SPD yang dibentuk dari ketiga model tersebut. Adapun hasil pemilihan model terbaik untuk data berdasarkan hasil estimasi model menggunakan metode Reversible Jump MCMC menunjukkan bahwa model *mixture* dengan banyak komponen dua dan tiga merupakan model terbaik untuk data. Namun demikian hasil estimasi model tersebut akan bersifat interpretatif jika disesuaikan dengan kategori inflasi. Dengan demikian model *mixture* yang disarankan apabila menggunakan metode Gibbs Sampler adalah:

$$X_j \sim 0,8949 N(6,839;1,233) + 0,1051 N(17,65;0,45)$$

dimana model tersebut menunjukkan bahwa tingkat inflasi yang terjadi pada periode Januari 2003 hingga Desember 2005 didominasi oleh data yang berada pada

kelompok pertama (kategori ringan) dengan bobot 89,49 % dan mean sama dengan 6,839 % serta variansi sebesar 0,8113 %.

Sedangkan hasil estimasi model *mixture* yang disarankan dengan menggunakan metode RJMCMC adalah :

$$X_i \sim 0,89441 N(0,83646;1,2624) + 0,10559 N(17,7745;1,0497)$$

dimana model tersebut juga menunjukkan bahwa tingkat inflasi yang terjadi pada periode Januari 2003 hingga Desember 2005 didominasi oleh data yang berada pada kelompok pertama (kategori ringan) dengan bobot 89,441% dan mean sama dengan 0,83646%.

5.2. Saran

Berdasarkan proses penyelesaian tesis dan hasil yang dicapai, ada beberapa hal yang disarankan untuk kelanjutan penelitian ini.

1. Pada tataran teoritis, penelitian ini sangat mungkin untuk dikembangkan pada pemodelan *mixture* normal multivariat dan distribusi *mixture* lainnya dengan banyak komponen tidak diketahui.
2. Perlu dikembangkan sebuah metode untuk membandingkan hasil estimasi model *mixture* dengan banyak komponen ditentukan dan tidak ditentukan untuk mengetahui estimasi yang memberikan hasil yang terbaik.

DAFTAR PUSTAKA

- Bank Indonesia (2006) Indikator Moneter dan Perbankan: Tingkat Inflasi (dalam %). [Online] Diakses tanggal 6 Januari 2006.
<http://www.bi.go.id/web/id/Indikator+Moneter+dan+Perbankan/Inflasi>
- Boediono (1982) *Seri Pengantar Ilmu Ekonomi: Ekonomi Makro*, B P F E , Yogyakarta.
- Box, G.E.P. dan Tiao, G.C. (1973) *Bayesian Inference in Statistical Analysis*, Reading MA: Addison Wesley, Philipina.
- Carlin, B.P. dan Chib, S. (1995) Bayesian Model Choice via Markov Chain Monte Carlo, *Journal Royal Statistical Society, Seri B*, **57**(3), hal. 473-484.
- Casella, G. dan George, E.I. (1992) Explaining Gibbs Sampler, *The American Statistician*, **46**(3), hal. 167-174.
- Gamerman, D. (1997) *Markov Chain Monte Carlo*, Chapman & Hall, London.
- Gelman, A., Carlin, J.B., Stern, H.S. dan Rubin, D.B. (1995) *Bayesian Data Analysis*, Chapman & Hall, London.
- Gilks, W.R. (1995) Full Conditional Distribution. Dalam *Markov Chain Monte Carlo in Practise* (eds W.R. Gilks, S. Richardson dan D.J. Spiegelhalter), hal. 75-88, Chapman & Hall, London.
- Green, P.J. (1995) Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination, *Biometrika*, **82**(4), hal.711-732.
- Hastings, W.K. (1970) Monte Carlo sampling methods using Markov chains and their applications, *Biometrika*, **57**, hal.97-109.
- Hogg, R.V. dan Craig, A.T. (1978) *Introduction to Mathematical Statistics*, Edisi 4, Macmillan, New York.
- Iriawan, N. (2000a) "Computationally Intensive Approaches to Inference in Neo-Normal Linear Models", Ph.D. Thesis, CUT-Australia.
- _____ (2000b) Pemilihan Model Menggunakan Struktur Perkalian Distribusi, *Kappa* **1**(1), hal. 37-44.
- _____ (2001) "Studi tentang 'Bayesian Mixture Normal' dengan Menggunakan Metode Markov Chain Monte Carlo (MCMC)", Laporan Penelitian, LEMLIT -ITS.
- _____ (2003) *Teknik Simulasi*, Modul Ajar, ITS.
- Kass, R.E. dan Raaferty, A.E. (1995) Bayes Factors, *Journal of the American Statistical Association*, **90**(430), hal. 774-795.

- McLachlan, G.J. dan Basford, K. (1988) *Mixture Models: Inference and Application to Clustering*, Marcel and Decker Inc.
- Prihartanti, W. (2002) “Studi Reliabilitas Model Kerusakan Ganda: Pendekatan Mixture Weibull”, Tesis, Pascasarjana – ITS, Surabaya.
- Richardson, S. dan Green, P.J. (1997) On Bayesian Analysis of Mixtures with an Unknown Number of Components (with discussions), *Journal of the Royal Society, Seri B*, **59**(4), hal. 731-792.
- Silverman, B.W. (1986) *Density Estimation for Statistical and Data Analysis*, Chapman & Hall, London.
- Suharjo, B. (2003) “Implementasi Bayesian Markov Chain Monte Carlo pada Distribusi Mixture Normal”, Tesis, Pascasarjana-ITS, Surabaya.
- Walsh, B. (2002) MCMC and Gibbs Sampling, Lecture Notes. [Online]
Diakses tanggal 28 Desember 2005.
<http://www.nitro.biosci.arizona.edu/courses/EEB596/handouts/Gibbs.pdf>
- Wati, D.M. (2002) “Implementasi Metode Bayes dengan Markov Chain Monte Carlo (MCMC) pada Estimasi Densitas Model Mixture Neo-Normal”, Tugas Akhir, FMIPA-ITS, Surabaya.

LAMPIRAN I

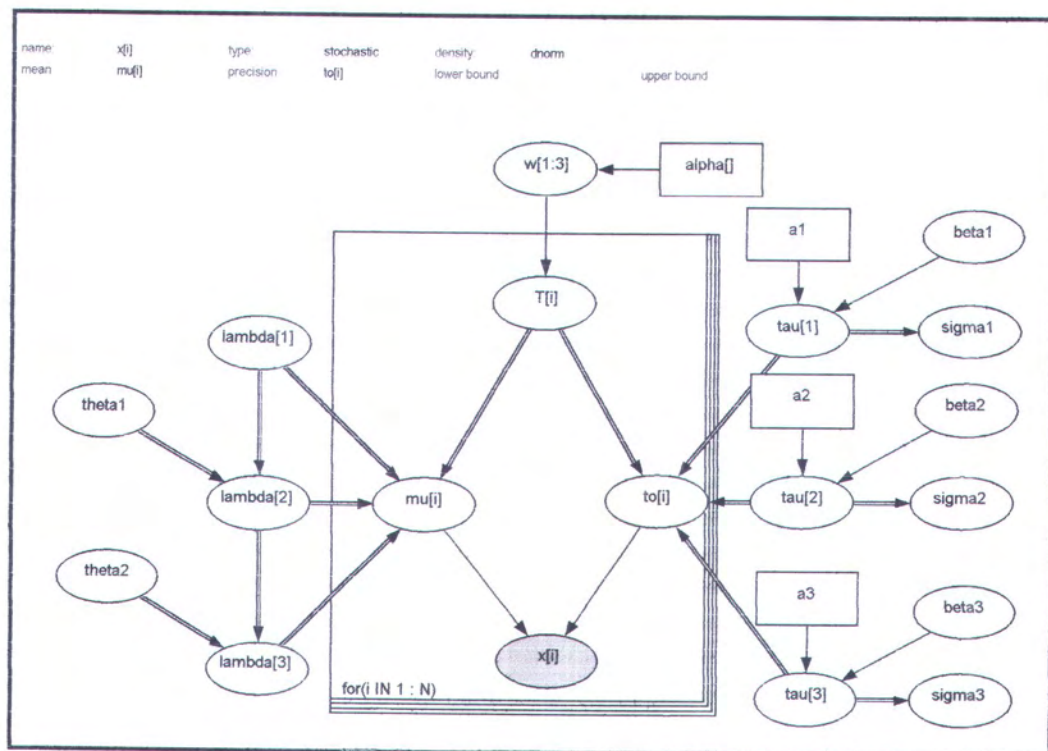
Data Tingkat Inflasi (dalam %)

Periode	Nilai	Periode	Nilai	Periode	Nilai
Jan/2003	8.68	Jan/2004	4.82	Jan/2005	7.32
Feb/2003	7.60	Feb/2004	4.60	Feb/2005	7.15
Mar/2003	7.17	Mar/2004	5.11	Mar/2005	8.81
Apr/2003	7.62	Apr/2004	5.92	Apr/2005	8.12
Mei/2003	7.15	Mei/2004	6.47	Mei/2005	7.40
Jun/2003	6.98	Jun/2004	6.83	Jun/2005	7.42
Jul/2003	6.27	Jul/2004	7.20	Jul/2005	7.84
Ags/2003	6.51	Ags/2004	6.67	Ags/2005	8.33
Sep/2003	6.33	Sep/2004	6.27	Sep/2005	9.06
Okt/2003	6.48	Okt/2004	6.22	Okt/2005	17.89
Nop/2003	5.53	Nop/2004	6.18	Nop/2005	18.38
Des/2003	5.16	Des/2004	6.40	Des/2005	17.11

LAMPIRAN 2

STRUKTUR DOODLE DAN PROGRAM MODEL *MIXTURE* DATA TINGKAT INFLASI DENGAN WINBUGS 1.4

A. Tiga komponen *mixture*




```

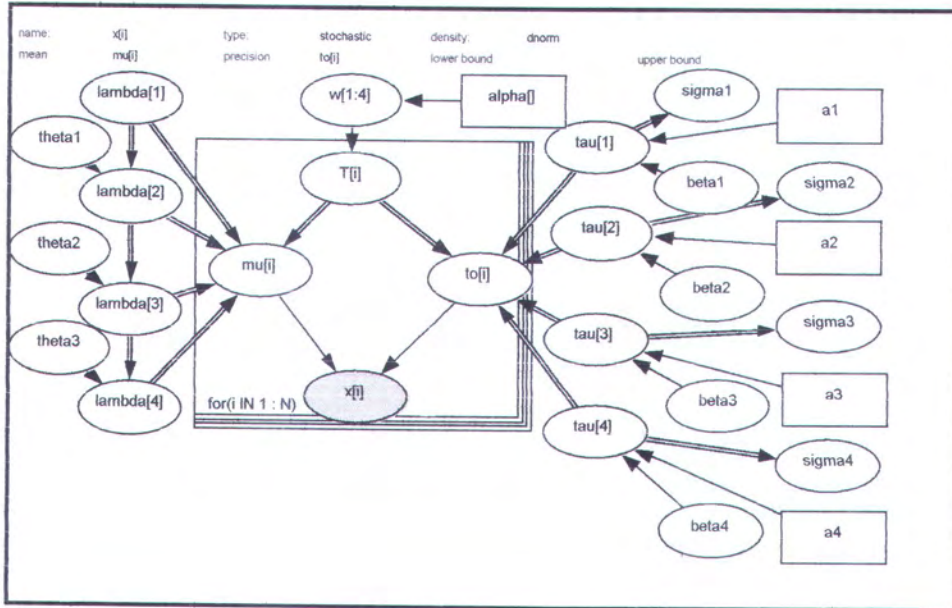
model:
{
  for(i in 1 : N) {
    x[i] ~ dnorm(mu[i],to[i])
  }
  w[1:3] ~ ddirch(alpha[])
  for(i in 1 : N) {
    T[i] ~ dcat(w[1:3])
  }
  for(i in 1 : N) {
    mu[i] <- lambda[T[i]]
  }
  for(i in 1 : N) {
    to[i] <- tau[T[i]]
  }
  lambda[1] ~ dnorm(5.0,0.574)I(0.0,)
  lambda[2] <- lambda[1] + theta1
  lambda[3] <- lambda[2] + theta2
  theta1 ~ dnorm(2.0,0.121)I(0.0,)
  theta2 ~ dnorm(10.0,0.62)I(0.0,)
  tau[1] ~ dgamma(a1,beta1)
  tau[2] ~ dgamma(a2,beta2)
  tau[3] ~ dgamma(a3,beta3)
  beta1 ~ dgamma(0.2,0.574)
  sigma1 <- 1 / tau[1]
  beta2 ~ dgamma(0.2,0.121)
  sigma2 <- 1 / tau[2]
  beta3 ~ dgamma(0.2,0.62)
  sigma3 <- 1 / tau[3]
}

DATA
list(x=c(4.6,4.82,5.11,5.16,5.53,5.92,6.18,6.22,6.27,6.27,6.33,6.4,
6.47,6.48,6.51,6.67,6.83,6.98,7.15,7.15,7.17,7.2,7.32,7.4,7.42,
7.6,7.62,7.84,8.12,8.33,8.68,8.81,9.06,17.11,17.89,18.38),N=36,
alpha=c(1,1,1),a1=2,a2=2,a3=2,
T=c(1,1,1,1,1,1,2,2,2,2,
2,2,2,2,2,2,2,2,2,2,
2,2,2,2,2,2,2,2,2,2,
2,2,2,3,3,3))

INITIAL
list(lambda=c(5.0,NA,NA),beta1=2,beta2=2,beta3=2,theta1=2,theta2=10.0)

```

B. Empat komponen mixture



```

model;
{
  for( i in 1 : N ) {
    x[i] ~ dnorm(mu[i],to[i])
  }
  w[1:4] ~ ddirch(alpha[])
  for( i in 1 : N ) {
    T[i] ~ dcat(w[1:4])
  }
  for( i in 1 : N ) {
    mu[i] <- lambda[T[i]]
  }
  for( i in 1 : N ) {
    to[i] <- tau[T[i]]
  }
  lambda[1] ~ dnorm( 4.0, 0.6)
  lambda[2] <- lambda[1] + theta1
  lambda[3] <- lambda[2] + theta2
  lambda[4] <- lambda[3] + theta3
  theta1 ~ dnorm( 2.0,0.12)I( 0.0,)
  theta2 ~ dnorm( 5.0,0.001)I( 0.0,)
  theta3 ~ dnorm( 5.0, 2.0)I( 0.0,)
  tau[1] ~ dgamma(a1,beta1)
  tau[2] ~ dgamma(a2,beta2)
  tau[3] ~ dgamma(a3,beta3)
  tau[4] ~ dgamma(a4,beta4)
  beta1 ~ dgamma( 0.2, 0.6)
  beta4 ~ dgamma( 0.2,2)
  sigma4 <- 1 / tau[4]
}

```

```

beta3 ~ dgamma(0.2,0.001)
sigma3 <- 1 / tau[3]
beta2 ~ dgamma(0.2,0.12)
sigma2 <- 1 / tau[2]
sigma1 <- 1 / tau[1]
}
DATA
list(x=c(4.6,4.82,5.11,5.16,5.53,5.92,6.18,6.22,
6.27,6.27,6.33,6.4,
6.47,6.48,6.51,6.67,6.83,6.98,7.15,7.15,
7.17,7.2,7.32,7.4,7.42,
7.6,7.62,7.84,8.12,8.33,8.68,8.81,9.06,
17.11,17.89,18.38),N=36,
alpha=c(1,1,1,1),a1=2,a2=2,a3=2,a4=2,
T=c(1,1,1,1,1,1,2,2,2,2,
2,2,2,2,2,2,2,2,2,
2,2,2,2,2,2,2,2,2,
2,2,2,3,4,4))

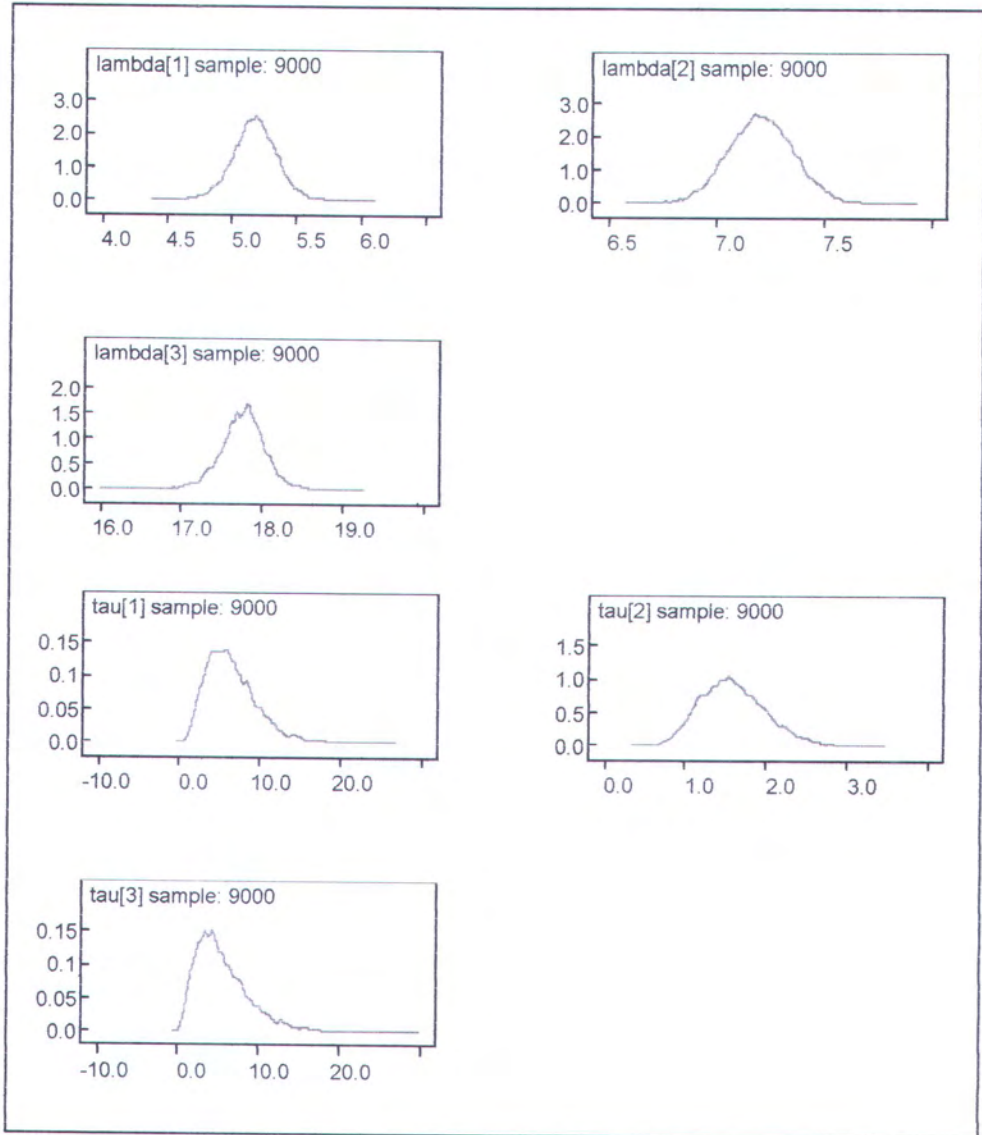
INITIAL
list(lambda=c(3.0,NA,NA,NA),
beta1=2,beta2=2,beta3=2,beta4=2,
theta1=2.0,theta2=5.0,theta3=5)

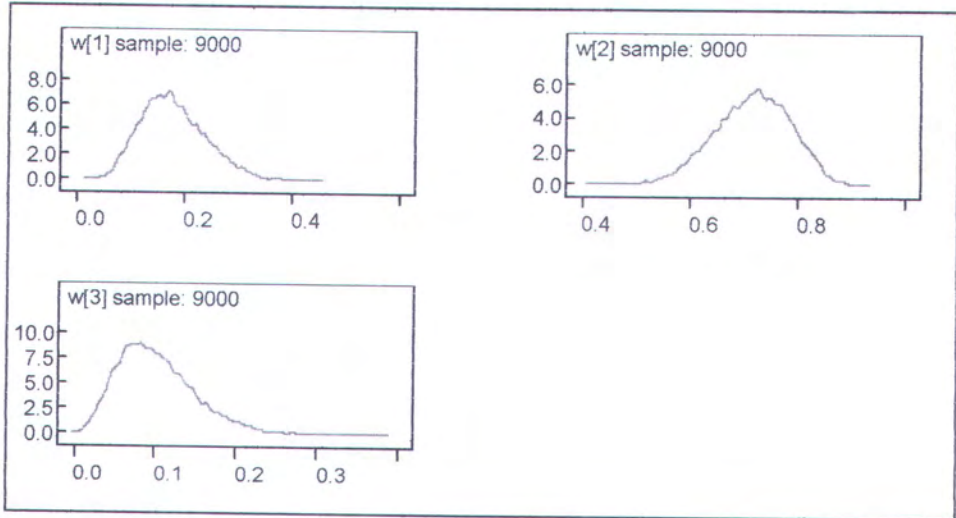
```

LAMPIRAN 3

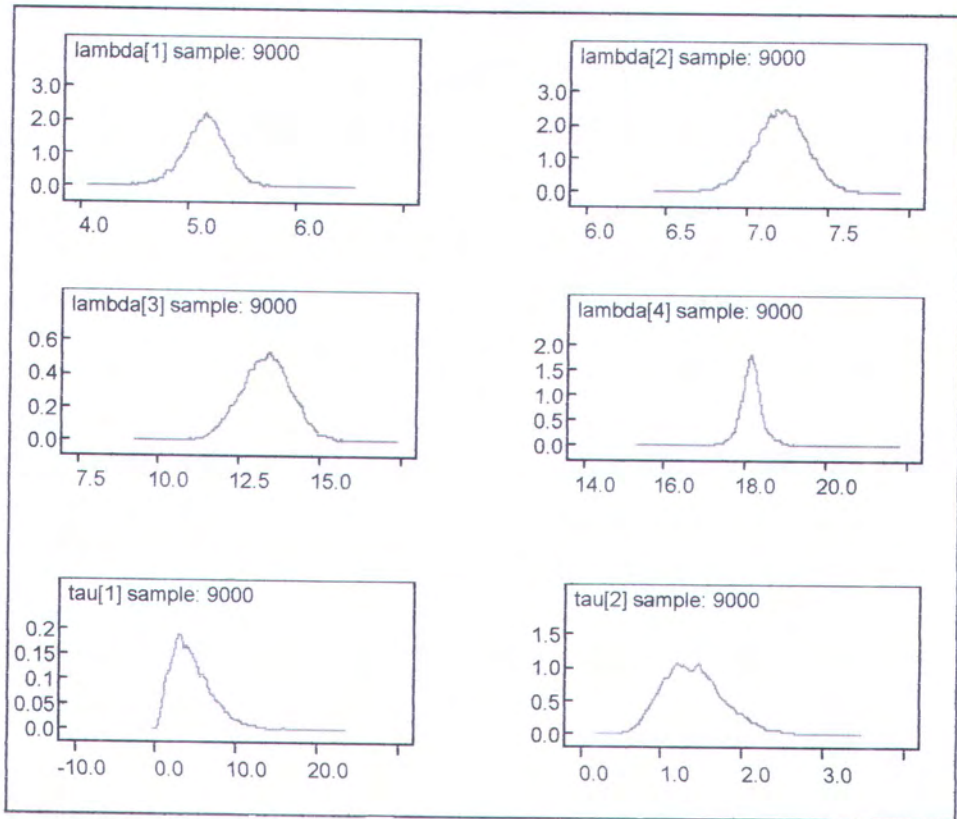
HASIL ITERASI PROGRAM WINBUGS DENGAN 10000 ITERASI

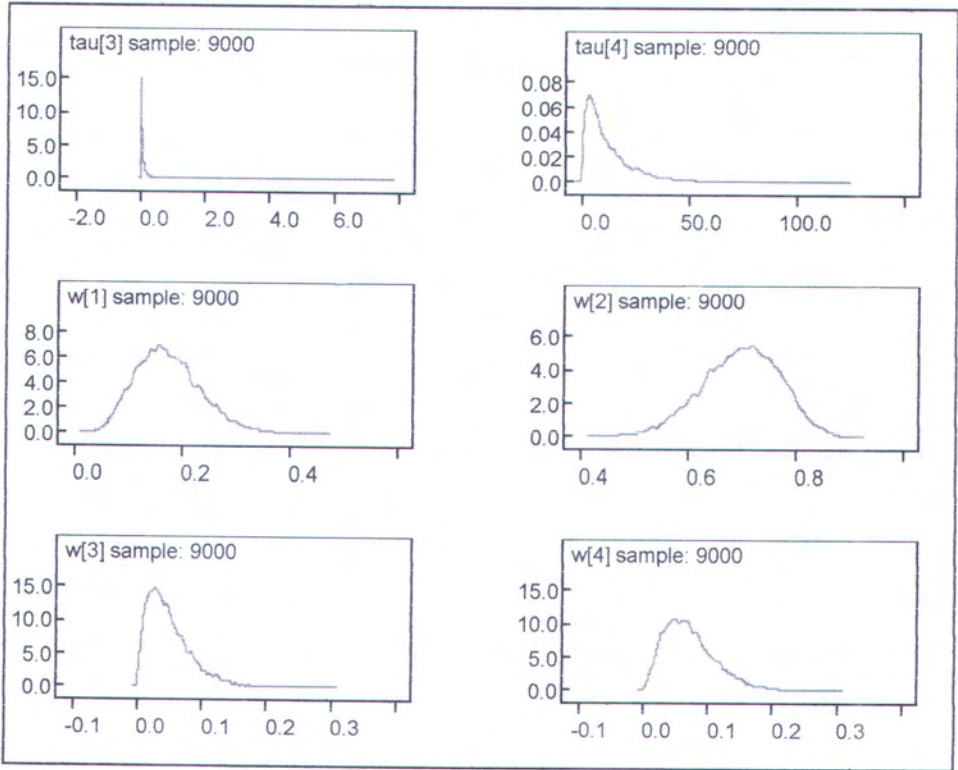
A. Plot densitas model *mixture* dengan tiga komponen





B. Plot densitas model *mixture* dengan empat komponen





LAMPIRAN 4

INISIAL DAN OUTPUT PROGRAM NMIX

```
Run: dwi/1 : ./Nmix dwi
random number seed: 1139211562
xi = 0.000 *R/2+ymid = 11.49
kappa = 1.000 /R^2 = 0.5266E-02
alpha = 2.000
beta is Gamma(g,h) with
  g = 0.2000
  h = 10.00 /R^2 = 0.5266E-01
delta = 1.000
lambda = -1.000
nsweep = 10000
nstep = 6
nburnin = 0
n = 36
ymin = 4.600
ymax = 18.38
  move schedule: swpahb
initial number of components: 1
splits: 560 out of 5049
split rej r: 348
combines: 531 out of 4951
births: 710 out of 5013
deaths: 736 out of 4987
```

Banyak komponen *mixture*, banyak iterasi, estimasi pembobot, mean, dan standar deviasi

1	5		
	1.00000	7.83051	3.36084
2	5670		
	0.89441	6.83646	1.12357
	0.10559	17.77451	1.02457
3	2460		
	0.50043	5.24695	1.09720
	0.40751	9.37368	1.07282
	0.09206	18.62684	1.01025
4	1042		
	0.29934	3.78147	1.03488
	0.38756	7.30916	1.01451
	0.23149	11.27870	1.03458
	0.08160	20.02937	0.99144

Lanjutan Lampiran 4

5	450		
	0.20177	2.37217	1.05488
	0.34760	6.39319	1.00476
	0.23794	9.20589	0.99811
	0.14634	13.65120	1.00753
	0.06634	21.76622	0.95124
6	222		
	0.14589	0.59398	0.95620
	0.27340	5.21667	0.95302
	0.27745	7.75425	0.94061
	0.15224	11.09772	0.93816
	0.09776	15.44258	0.94257
	0.05326	22.20133	0.87134
7	87		
	0.10413	-0.20341	0.90047
	0.20994	4.80227	0.97221
	0.23263	7.03449	0.97588
	0.23216	9.02826	0.83090
	0.10799	12.83079	0.91980
	0.06836	17.50535	0.89042
	0.04479	25.57393	0.96870
8	42		
	0.10154	-0.43734	0.81635
	0.15914	4.79914	0.84244
	0.24313	6.67427	0.96624
	0.19025	8.04938	0.85666
	0.14247	11.03168	0.91642
	0.07004	14.81404	0.96941
	0.05286	19.46928	0.81504
	0.04057	27.62432	0.86559
9	16		
	0.02448	-6.65697	1.15081
	0.14488	3.97937	1.15073
	0.10940	5.24849	0.86243
	0.32888	7.20347	0.93133
	0.15719	10.12993	0.94048
	0.09277	12.18856	1.06936
	0.05955	14.21493	1.17474
	0.05986	18.50611	1.02352
	0.02299	28.87075	1.30114

Lanjutan Lampiran 4

10	6		
0.04040	-6.30754	1.00953	
0.03701	3.58734	1.00590	
0.19780	4.70196	1.06236	
0.38202	6.81473	1.06504	
0.04665	9.95018	0.67136	
0.09136	11.71507	1.22066	
0.07344	14.75169	1.73059	
0.03834	17.79785	1.31829	
0.07008	20.50792	0.84762	
0.02290	30.31724	1.34879	