

# Chapter 4

Numerical session on matlab

# Chapter 4

- Introduction (p. 3)
- Simple Gibbs and MH MCMC (p. 9)
- CP-AR models (p. 27)
  - Carlin, Gelman and Smith : Griddy-Gibbs
  - Chib's algorithm

# Introduction

# Introduction

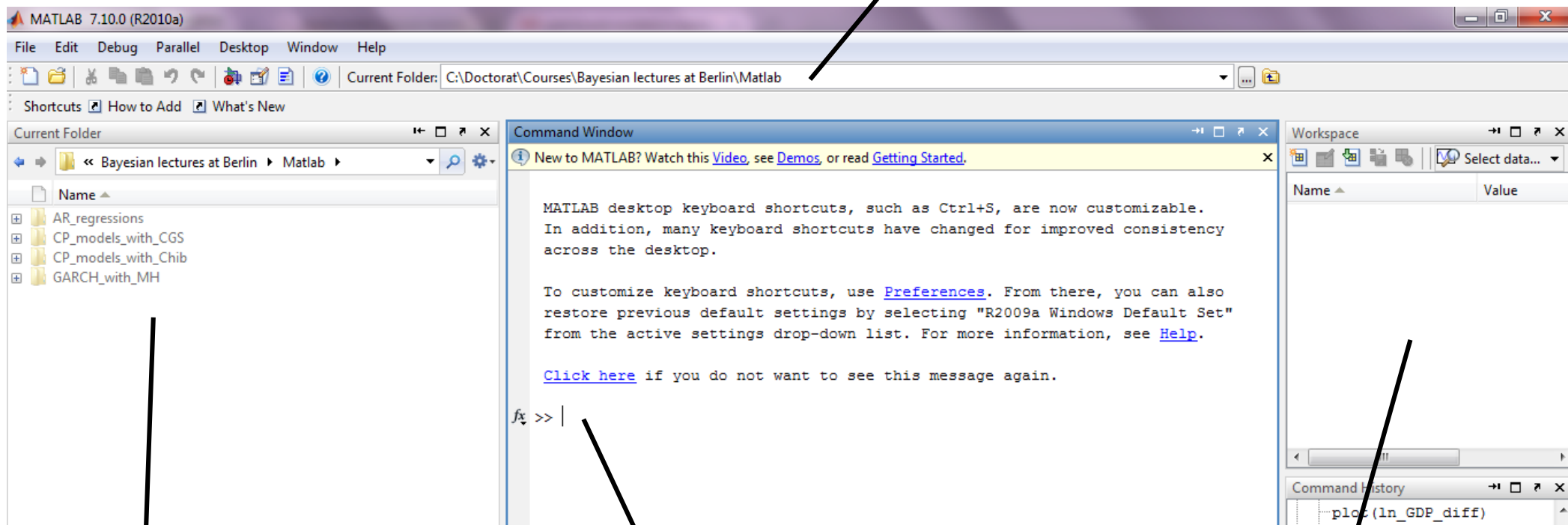
## Quick start

- Four directories :
  - AR\_regressions - *MCMC for simulating posteriors of AR models*
  - GARCH\_with\_MH - *MH MCMC for simulating posteriors of a GARCH(1,1) model*
  - CP\_models\_with\_CGS - *MCMC for drawing posteriors of AR models with 2 regimes using the Carlin, Gelman and Smith approach*
  - CP\_models\_with\_Chib - *MCMC for posterior distributions of AR models with  $k$  regimes based on Chib's algorithm*

# Introduction

How to use the programs ?

Current folder : Matlab directory



Navigator

Command window

Workspace

# Introduction

Go to the AR\_regressions directory  
Click on US\_GDP\_percentage

Current folder :  
Matlab directory

The screenshot shows the MATLAB 7.10.0 (R2010a) interface. The top menu bar includes File, Edit, View, Debug, Parallel, Desktop, Window, and Help. The current folder is set to C:\Doctorat\Courses\Bayesian lectures at Berlin\Matlab\AR\_regressions. The Navigator window on the left shows a list of files in the 'AR\_regressions' directory, with 'US\_GDP\_percentage.mat' selected. The Command Window in the center displays a message about customizable keyboard shortcuts and a prompt 'f>>'. The Workspace window on the right shows variables 'US\_GDP\_growth' (210x1 double) and 'date\_US\_GDP' (1x210 cell). The Command History window at the bottom right shows a sequence of commands including 'plot(ln\_GDP\_diff)', 'date\_US\_GDP = date\_Q;', 'clear all', '3\* US\_Tbill = PPT1;', 'US\_Tbill = PPT1;', 'date\_US\_Tbill = date;', 'max(randn(10,1),1)', and 'Simu = launch\_CP\_AR\_Ca'.

Navigator

Command window

Workspace with data

# Introduction

## How to run a program ?

- **Open the program :**  
`launch_AR_regression_with_Gibbs_sampler`
  - **Comments to run the function are in green**
- **Copy and paste the first line without 'function' in the command window**

```

Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.

MATLAB desktop keyboard shortcuts, such as Ctrl+S, are now customizable.
In addition, many keyboard shortcuts have changed for improved consistency
across the desktop.

To customize keyboard shortcuts, use Preferences. From there, you can also
restore previous default settings by selecting "R2009a Windows Default Set"
from the active settings drop-down list. For more information, see Help.

Click here if you do not want to see this message again.

fx >> [Simu] = launch_AR_regression_with_Gibbs_sampler(US_GDP_growth,1,1000)|
  
```

Instead of y :  
the data

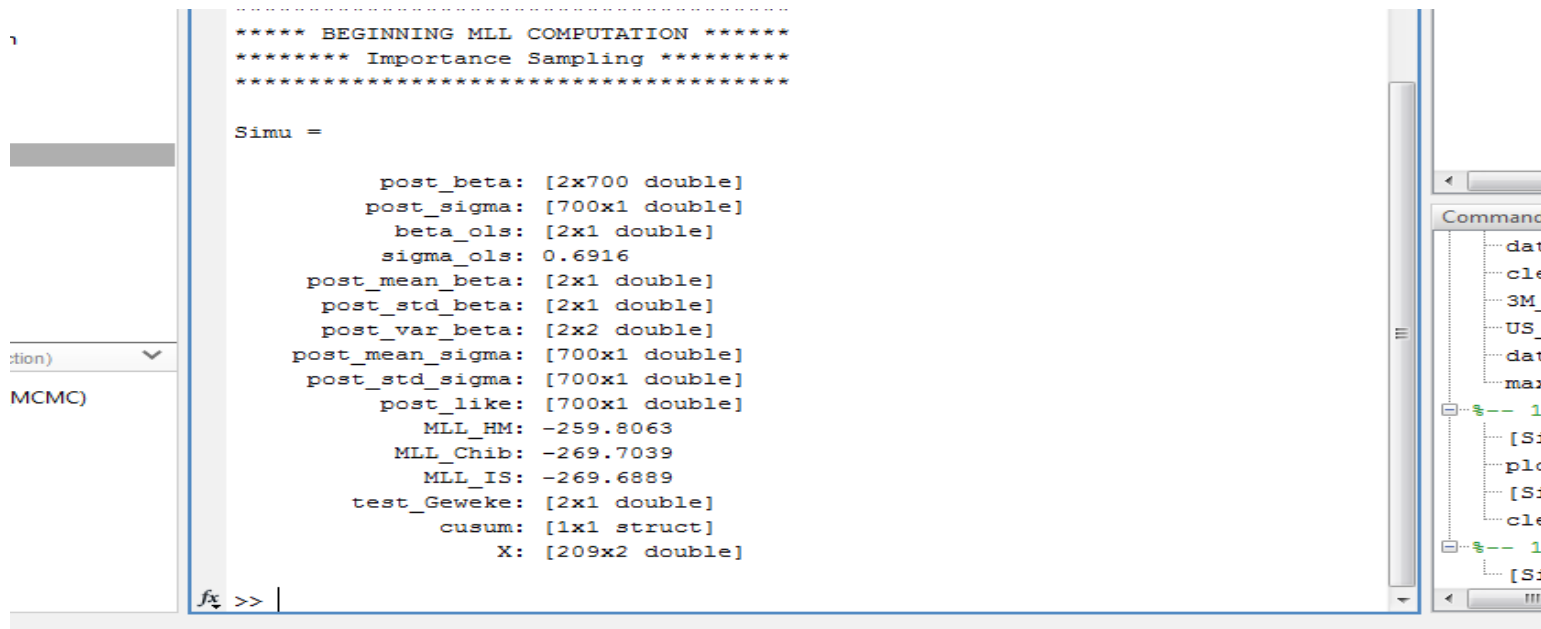
nb\_MCMC : 1000 iterations

Instead of  
AR\_lags : the  
order of the  
AR process

# Introduction

## What is a structure ?

- **Simu is a matlab structure containing many tables**



```
***** BEGINNING MLL COMPUTATION *****
***** Importance Sampling *****
*****

Simu =

    post_beta: [2x700 double]
    post_sigma: [700x1 double]
    beta_ols: [2x1 double]
    sigma_ols: 0.6916
    post_mean_beta: [2x1 double]
    post_std_beta: [2x1 double]
    post_var_beta: [2x2 double]
    post_mean_sigma: [700x1 double]
    post_std_sigma: [700x1 double]
    post_like: [700x1 double]
    MLL_HM: -259.8063
    MLL_Chib: -269.7039
    MLL_IS: -269.6889
    test_Geweke: [2x1 double]
    cusum: [1x1 struct]
    X: [209x2 double]
```

- To access the matrix 'post\_beta' :

→ **Type in command window : `Simu.post_beta`**



# Simple MCMC

# Gibbs sampler

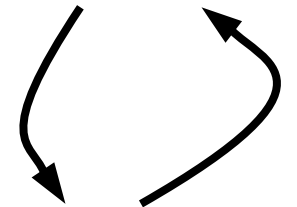
- **The model**

$$\begin{cases} y_t = \beta' x_t + \epsilon_t \\ \epsilon_t \sim \text{i.i.d.} N(0, \sigma^2) \end{cases}$$

- **The prior distributions**

$$\beta \sim N(\beta_0, \Sigma_0) \quad \sigma^2 \sim IG(IG_a, IG_b)$$

- **Conditional posterior distributions :**

$$\pi(\theta | Y_{1:T}, \sigma^2) \sim N(\bar{\mu}, \bar{\Sigma}) \quad \begin{cases} \bar{\Sigma} = [\sigma^{-2} \sum_{t=1}^T (x_t x_t') + \Sigma_0^{-1}]^{-1} \\ \bar{\mu} = \bar{\Sigma} [\sigma^{-2} \sum_{t=1}^T x_t y_t + \Sigma_0^{-1} \beta_0] \end{cases}$$


$$\pi(\sigma^2 | Y_{1:T}, \theta) \sim IG(IG_a + T/2, IG_b + \sum_{t=1}^T \epsilon_t^2 / 2)$$

# Gibbs sampler

- Initial values :

$$\beta = \left( \sum_{t=1}^T x_t x_t' \right)^{-1} \left( \sum_{t=1}^T x_t y_t \right)$$

**MLE estimates**

$$\sigma^2 = \sum_{t=1}^T (y_t - \beta' x_t)^2 / T$$

- Gibbs sampler :

$$\beta \sim \beta | Y_{1:T}, \sigma^2 \quad \sigma^2 \sim \sigma^2 | Y_{1:T}, \beta$$

**Discard the first draws as burn-in period**

**Use the rest as a sample of the posterior distribution :**

$$\beta, \sigma^2 | Y_{1:T}$$

# Gibbs sampler

- Open the function : *Gibbs\_regression\_with\_MLL*
  - Gibbs sampler of a simple regression
  - Provide the marginal log-likelihood (MLL) estimated from three different approaches (Chib, Importance sampling and Harmonic mean)
    - **Harmonic mean : very bad estimate**
- The prior distributions

```

47
48
49
50
51
52
53
54
55 -
56 -
57 -
58 -
59 -
60 -

```

```

#####
#### Set the prior values
#####
### Prior :
### beta ~ N(beta_0, Sigma_0)
### sigma^2 ~ IG(a,b)
#####
IG_b = 1;
IG_a = 1;
var_uninformative = 100; %% We fix the variance of each beta equal to
beta_0 = zeros(dimension,1);
Sigma_0 = diag(ones(dimension,1)*(var_uninformative));
inv_Sigma_0 = diag(ones(dimension,1)*(1/var_uninformative));

```

Hyper-parameters of the mean parameters

Hyper-parameters of the variance

# Gibbs sampler

```

7  #####
8  ##### MCMC starting values
9  #####
10 - beta = beta_ols;
11 - sigma = sigma_ols;
12 - iter = 1;
13 - for i=1:nb_MCMC
14 -     progressbar(i/nb_MCMC);
15 -     #####
16 -     ##### Saving posterior samples
17 -     #####
18 -     if(i>burn_in)
19 -         beta_post(:,iter) = beta;
20 -         sigma_post(iter) = sigma;
21 -         like_post(iter) = (-T/2)*log(2*pi*sigma) - (y-X*beta)'*(y-X*beta)/(2*sigma);
22 -         iter = iter +1;
23 -     end
24 -     #####
25 -     ### Update of sigma|beta
26 -     #####
27 -     IG_post_b_MCMC = IG_b + 0.5*sum((y-X*beta).^2);
28 -     inv_sigma = gamrnd(IG_post_a_MCMC,1/IG_post_b_MCMC);
29 -     sigma = 1/inv_sigma;
30 -     #####
31 -     ### Update of beta|Sigma
32 -     #####
33 -     Sigma_cond = inv(inv_sigma*Mat_XX + inv_Sigma_0);
34 -     mu_cond = Sigma_cond*(inv_sigma*Vec_Xy + inv_Sigma_0*beta_0);
35 -     beta = mvnrnd(mu_cond,Sigma_cond)';
36 - end
37 - #####

```

} Starting values

→ Saving posterior draws

} First block -  
the variance

} First block -  
the mean parameters

# Gibbs sampler

```

7  #####
8  ##### MCMC starting values
9  #####
10 - beta = beta_ols;
11 - sigma = sigma_ols;
12 - iter = 1;
13 - for i=1:nb_MCMC
14 -     progressbar(i/nb_MCMC);
15 -     #####
16 -     ##### Saving posterior samples
17 -     #####
18 -     if(i>burn_in)
19 -         beta_post(:,iter) = beta;
20 -         sigma_post(iter) = sigma;
21 -         like_post(iter) = (-T/2)*log(2*pi*sigma) - (y-X*beta)'*(y-X*beta)/(2*sigma);
22 -         iter = iter +1;
23 -     end
24 -     #####
25 -     ### Update of sigma|beta
26 -     #####
27 -     IG_post_b_MCMC = IG_b + 0.5*sum((y-X*beta).^2);
28 -     inv_sigma = gamrnd(IG_post_a_MCMC,1/IG_post_b_MCMC);
29 -     sigma = 1/inv_sigma;
30 -     #####
31 -     ### Update of beta|Sigma
32 -     #####
33 -     Sigma_cond = inv(inv_sigma*Mat_XX + inv_Sigma_0);
34 -     mu_cond = Sigma_cond*(inv_sigma*Vec_Xy + inv_Sigma_0*beta_0);
35 -     beta = mvnrnd(mu_cond,Sigma_cond)';
36 - end
37 - #####

```

} Starting values

→ Saving posterior draws

} First block -  
the variance

} First block -  
the mean parameters

# Gibbs sampler

- Open the function : *launch\_AR\_regression\_with\_Gibbs\_sampler*

In the command window, type :

```
[Simu] =
```

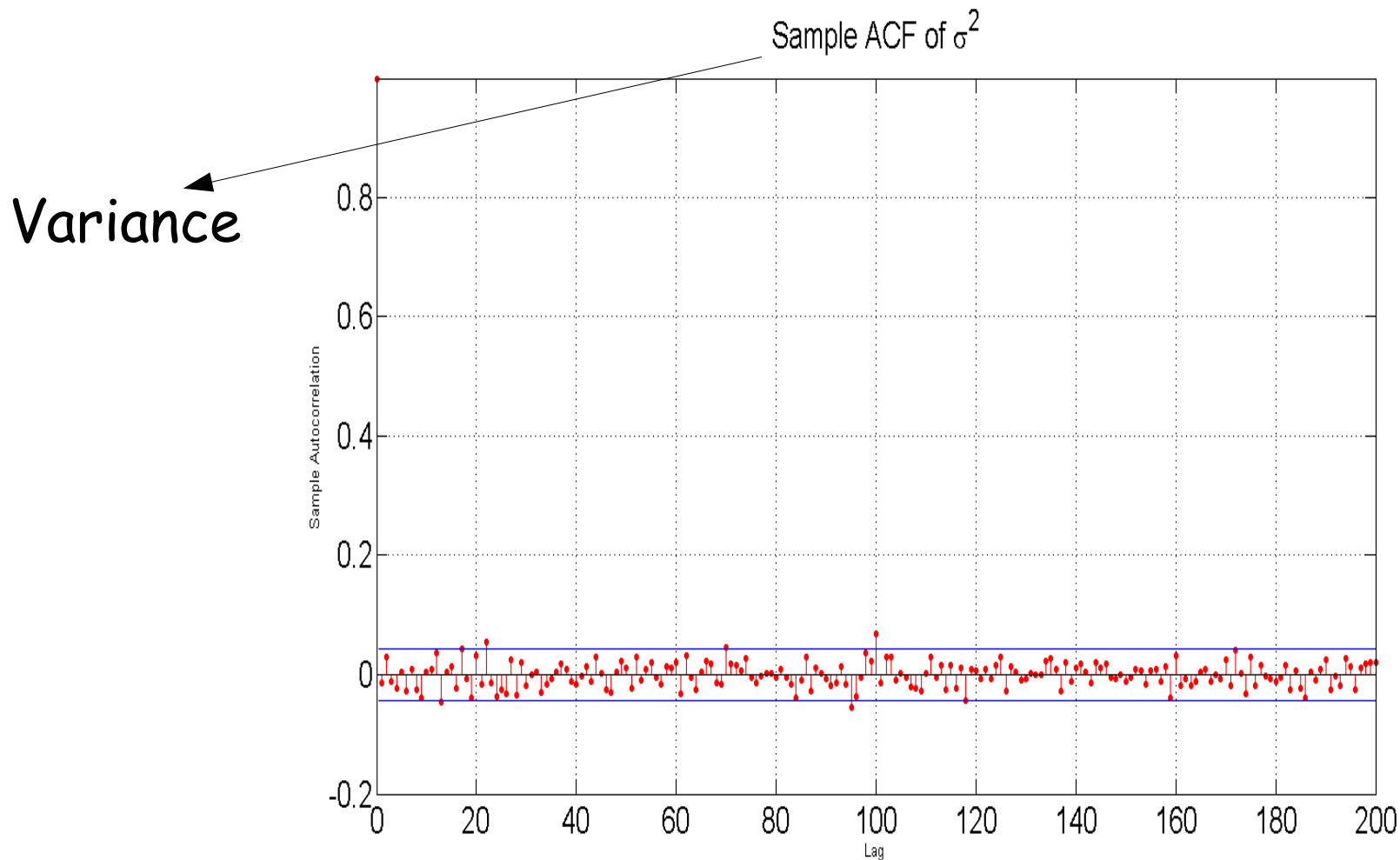
```
launch_AR_regression_with_Gibbs_sampler(US_GDP_growth  
h,1,3000,1)
```

- The data : US\_GDP\_growth
- AR order : 1
- Nb MCMC iterations : 3000
- Convergence Graphics : on

# Gibbs sampler

- Graphics :

## Empirical correlations between MCMC draws



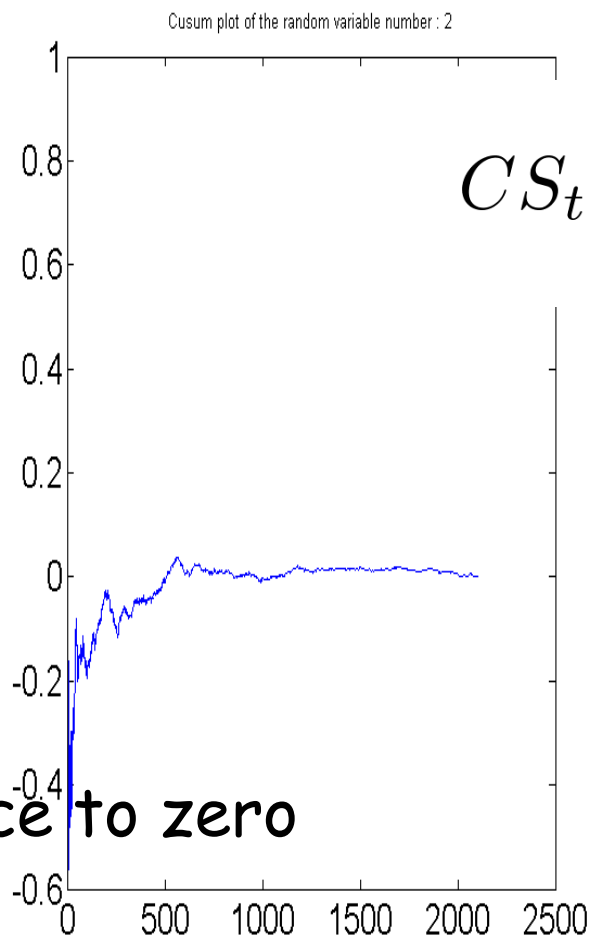
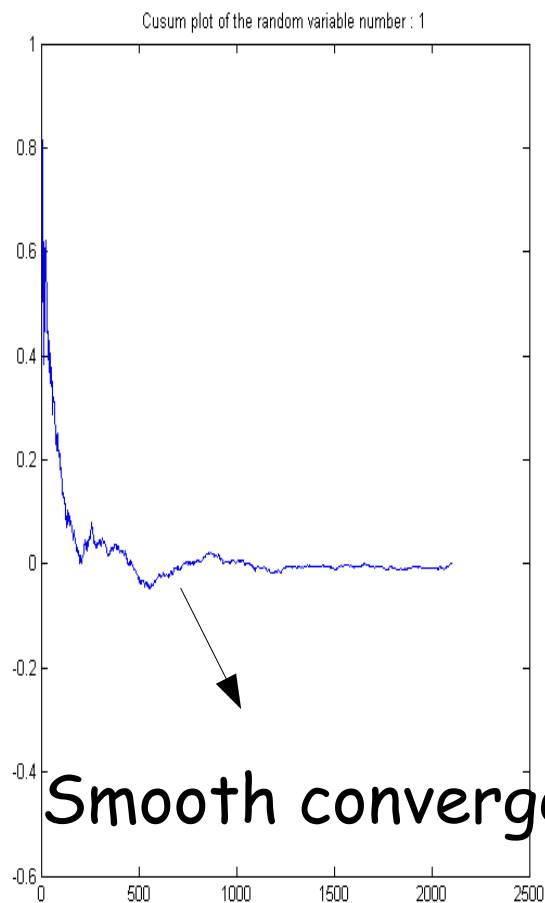


# Gibbs sampler

- Graphics :

## Cumsum plots of the two mean parameters

Empirical std of  
the parameter



Smooth convergence to zero

$$CS_t = \left( \frac{1}{t} \sum_{j=1}^t \theta^j - m_\theta \right) / s_\theta$$

Empirical mean of  
the parameter

$CS_N = 0$

# Gibbs sampler

- Geweke's test of MCMC convergence :

Comparison of two empirical means well separated

In the command window :

`Simu.test_Geweke`

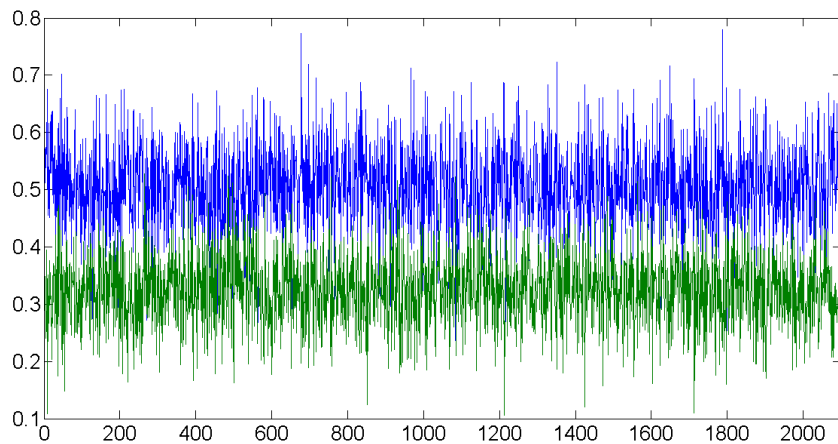
→ Test for each mean parameters and return zero if the hypothesis is not rejected at 95%

```
// set(gca, 'fontSize',20),  
>> set(gca,'fontSize',20);  
>> Simu.test_Geweke  
  
ans =  
    0  
    0 } Convergence  
  
fx >>  
<
```

# Gibbs sampler

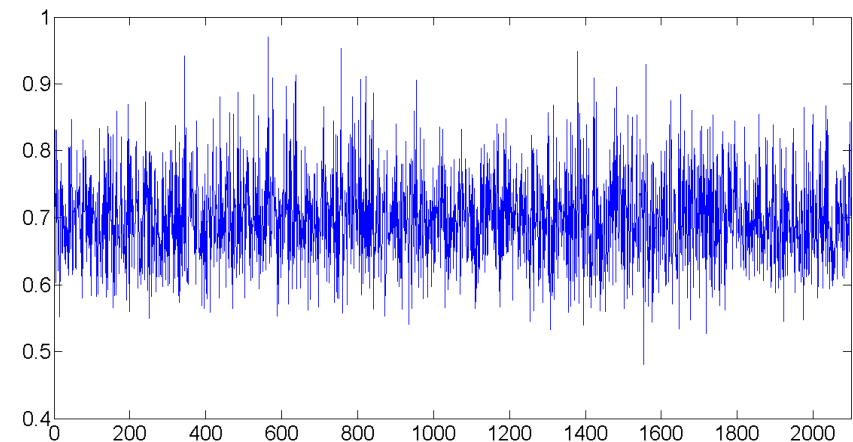
- Results :

`plot(Simu.post_beta')`



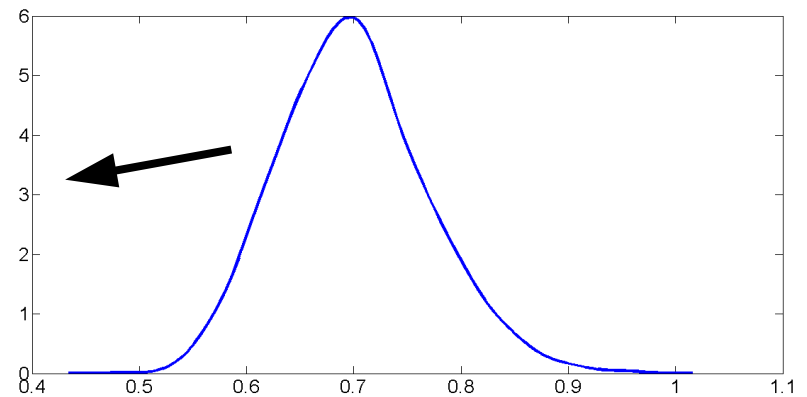
$\text{mean}(\text{Simu.post\_beta}') = 0,50 ; 0,32$   
 $\text{std}(\text{Simu.post\_beta}') = 0,08 ; 0,06$

`plot(Simu.post_sigma')`



$\text{mean}(\text{Simu.post\_sigma}') = 0,70$   
 $\text{std}(\text{Simu.post\_sigma}') = 0,07$

Marginal distribution  
of the variance



# Gibbs sampler

- Testing for the order of the AR process :
  - Launch several times the program with different # of lags
  - Compare the MLL obtained from the different simulations

	AR(0)	AR(1)	AR(2)	AR(3)	AR(4)
MLL HM	-274,16	-259,89	-256,36	-257,16	-254,75
MLL Chib	-279,12	-269,7	-269,71	-273,72	-275,46
MLL IS	-279,12	-269,7	-269,71	-273,72	-275,46

- Harmonic mean : not reliable
- Same Estimates from the local and the global formula

# MH sampler

- Open the function : *MH\_regression\_with\_MLL*
  - MH sampler of a simple regression
  - Provide the marginal log-likelihood (MLL) estimated from three different approaches (Chib, Importance sampling and Harmonic mean)
- Running an estimation of AR model with MH sampler :  
*launch\_AR\_regression\_with\_Gibbs\_sampler*

# Gibbs sampler

- **Adaptation of the proposal distribution**  
(Atchadé and Rosenthal (2005) )

$$\bar{\Sigma}_i = \bar{\Sigma}_{i-1} + (\phi_r^{i-1} - \phi_{target}) / (i^c) \quad \text{if } \Sigma_{Low} < \bar{\Sigma}_i < \Sigma_{High}$$

```

140
141 -   for q=1:dimension+1
142 -       adapt_rate(q) = max(min_max_var(1), adapt_rate(q) + (accept_rate(q)/i - 0.45)/(i^eta));
143 -       if(adapt_rate(q) > min_max_var(2))
144 -           adapt_rate(q) = min_max_var(2);
145 -       end
146 -   end
147

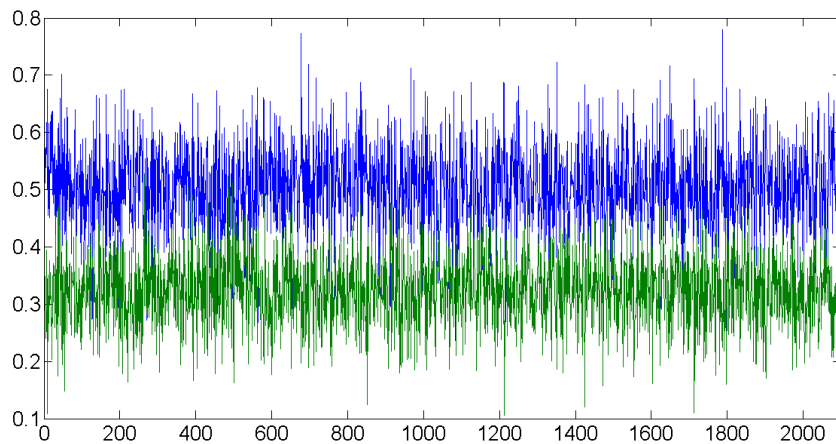
```

↓  
 $\phi_{target}$

# Comparison Samplers

- AR(1)

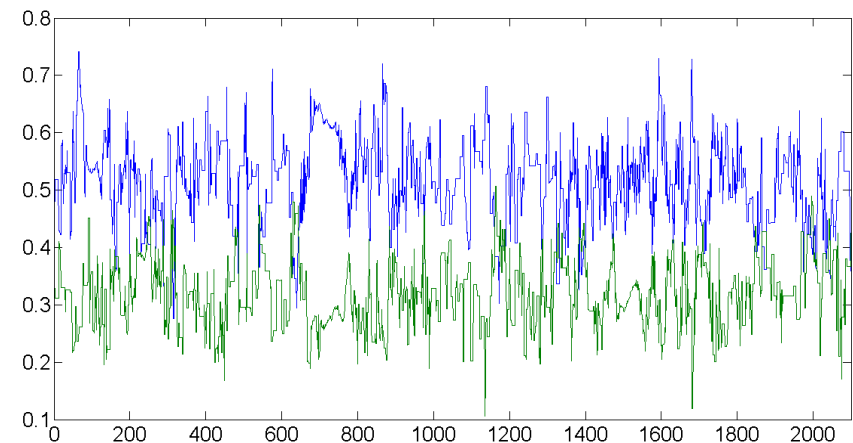
## Gibbs sampler



Mean= 0,50 ; 0,32

Std = 0,08 ; 0,06

## Metropolis-Hastings



Mean = 0,51 ; 0,32

Std = 0,07 ; 0,06

Acceptance rate :  
45 % ; 44 % ; 45 %

# MH sampler

- Testing for the order of the AR process :
  - Launch several times the program with different # of lags
  - Compare the MLL obtained from the different simulations

	AR(0)	AR(1)	AR(2)	AR(3)	AR(4)
MLL Chib (MH)	-279,12	-269,7	-269,71	-273,72	-275,46
MLL Chib (Gibbs)	-279,12	-269,7	-269,71	-273,72	-275,46

Same Estimates from Gibbs or MH samplers

**Marginal likelihood depends on the prior!**

- Test the order of the AR process with different priors



# GARCH estimation by MH MCMC

- Change the directory and go to *GARCH\_with\_MH*
- Import financial data by clicking on *SP500\_percentage\_returns.mat*
- Main matlab program *MCMC\_GARCH\_RW*

Estimate a GARCH(1,1) model with

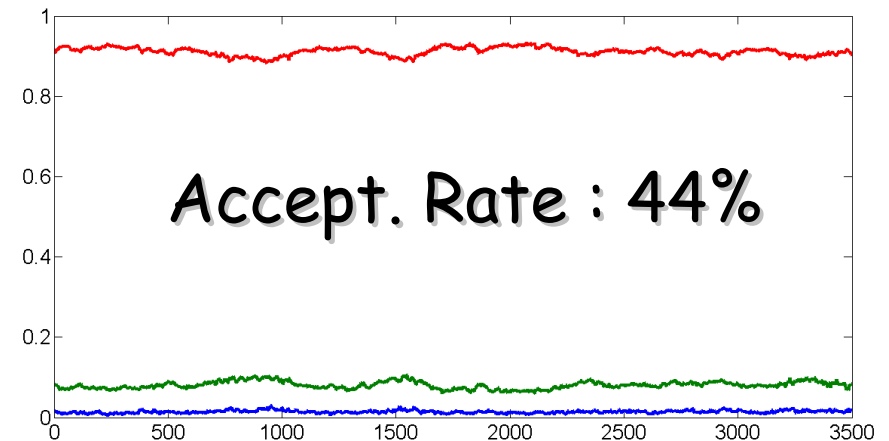
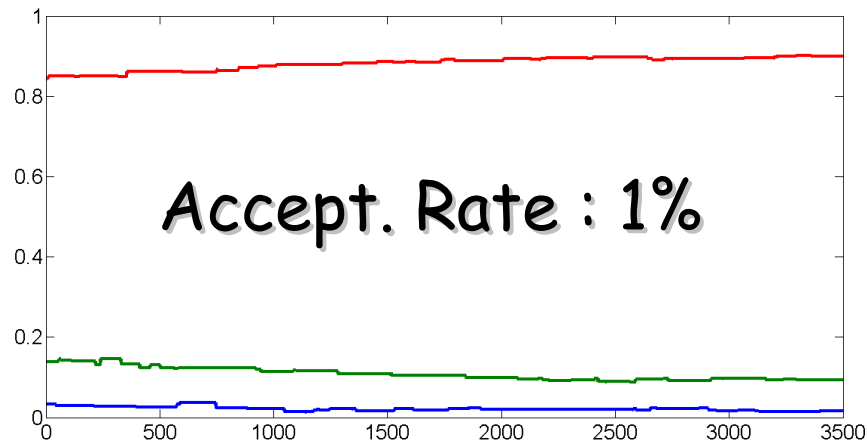
- non adaptive RW
- adaptive RW

## Inputs :

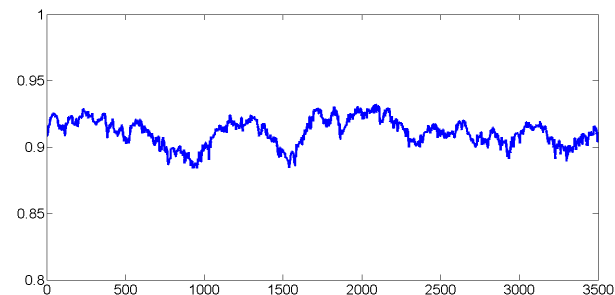
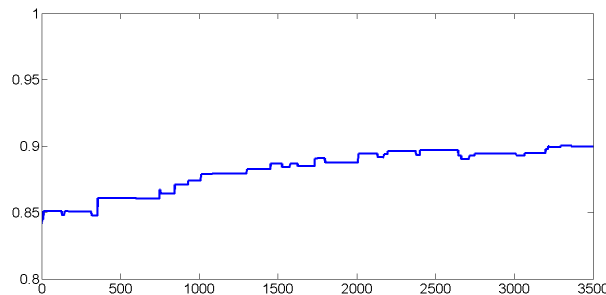
- $Y$  : financial time series (here SP500)
- $nb\_MCMC$  : number of MCMC iterations
- $RW\_step$  : Variance of the proposal distribution
- $Graph$  : Convergence graphics

# GARCH estimation by MH MCMC

- Choose a RW variance and run the program
- `[Simu] = MCMC_GARCH_RW(SP500,5000,0.1)`



- Focus on parameter  $\beta$  in  $\sigma_t^2 = \omega + \alpha y_{t-1}^2 + \beta \sigma_{t-1}^2$



# Change-Point AR models

# CP-AR models with CGS

- Change the directory and go to *CP\_models\_with\_CGS*
- Import data by clicking on *US\_GDP\_percentage.mat*
- Main matlab program  
*Gibbs\_regression\_Carlin\_Gelman\_Smith*
  - Estimates a CP model with 2 regimes using CGS's algorithm

## Inputs :

- Y : a time series (here *US\_GDP\_growth*)
- X : explanatory variables
- nb\_MCMC : number of MCMC iterations
- Program for estimating a CP-AR(q) model  
*Launch\_CP\_AR\_Carlin\_Gelman\_Smith*

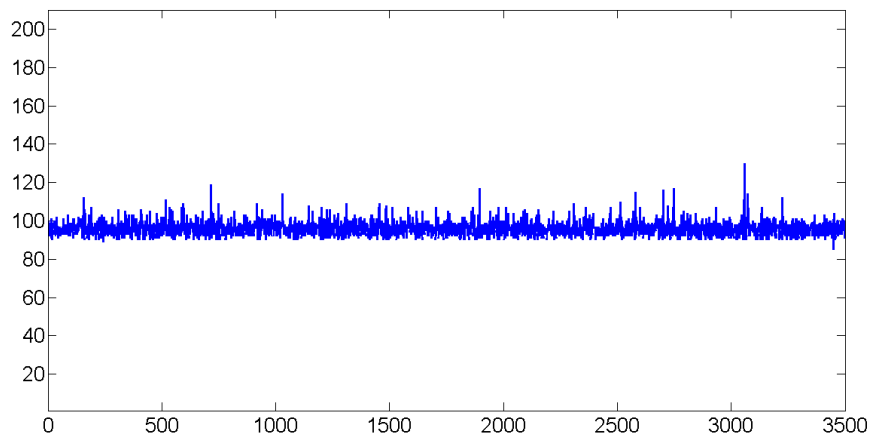
# CP-AR models with CGS

- Run an estimation of a CP-AR(1) model

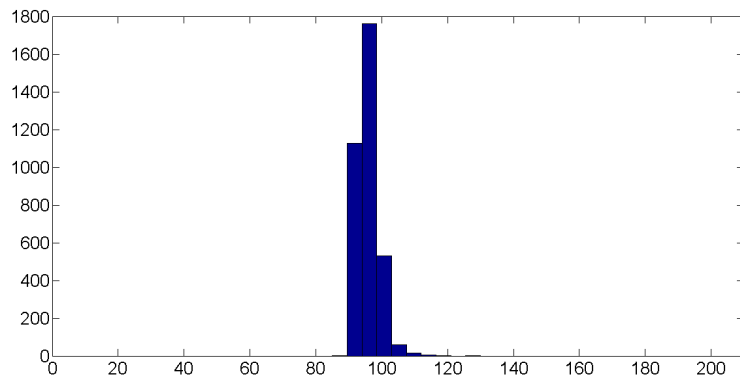
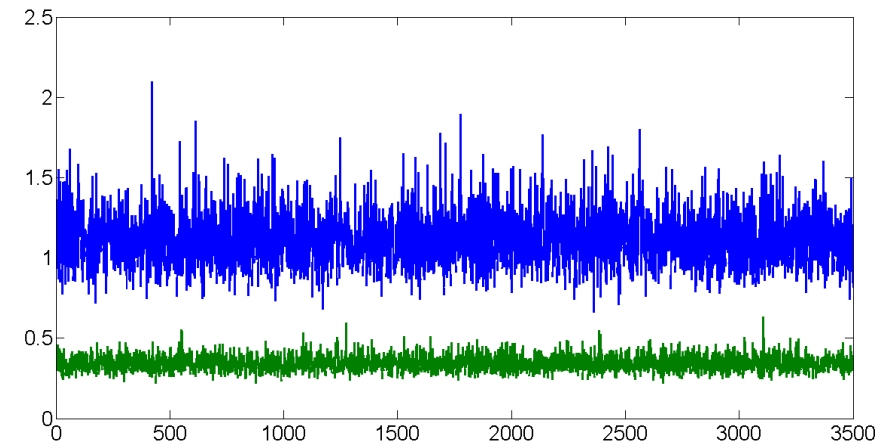
*[Simu] =*

*launch\_CP\_AR\_Carlin\_Gelman\_Smith(US\_GDP\_growth,1,5000)*

*plot(Simu.post\_tau')*



*plot(Simu.post\_sigma')*



- Great moderation
- Not a symmetric distribution

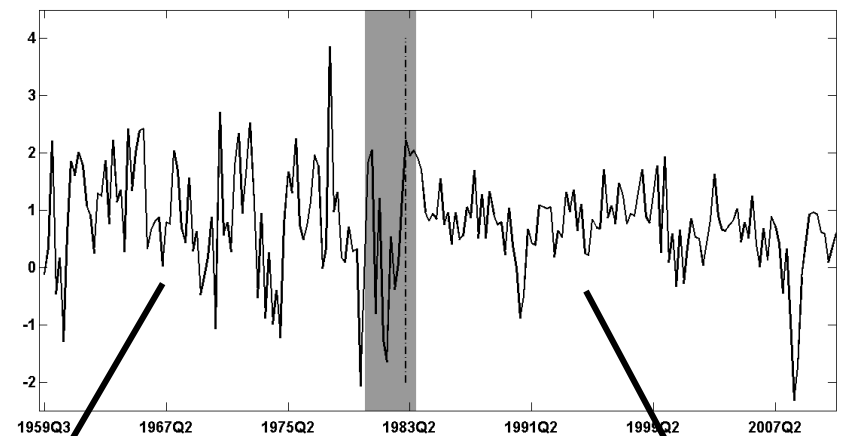
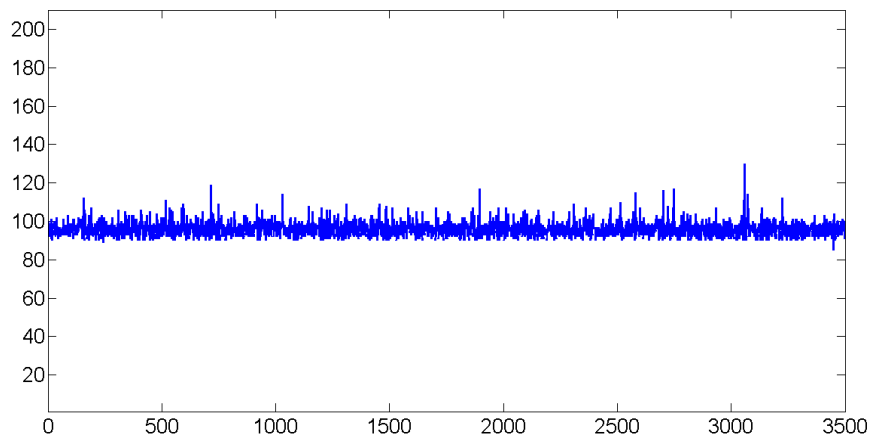
# CP-AR models with CGS

- Run an estimation of a CP-AR(1) model

*[Simu] =*

*launch\_CP\_AR\_Carlin\_Gelman\_Smith(US\_GDP\_growth,1,5000)*

*plot(Simu.post\_tau')*



$$E(\sigma_1^2 | Y_{1:T}) \approx 1.11 \quad E(\sigma_2^2 | Y_{1:T}) \approx 0.34$$

# CP-AR models with Chib

- Change the directory and go to *CP\_models\_with\_Chib*
- Import data by clicking on *US\_GDP\_percentage.mat*
- Main matlab program *Gibbs\_regression\_chib*
  - Estimates a CP model with k regimes using Chib's algorithm

## Inputs :

- Y : a time series (here *US\_GDP\_growth*)
- X : explanatory variables
- Regime : number of regimes
- nb\_MCMC : number of MCMC iterations
- MLL\_computation : =1 if MLL must be estimated

# CP-AR models with Chib

- **Main matlab program** *Gibbs\_regression\_chib*
  - Estimates a CP model with k regimes using Chib's algorithm
- **Matlab program** *launch\_CP\_model\_estimations*
  - Estimates CP-AR models from one up to k regimes

## Inputs :

- Y : a time series (here *US\_GDP\_growth*)
- AR\_lags : The order of the AR process
- upper\_bound\_regime : Max. considered number of regimes
- nb\_MCMC : number of MCMC iterations



# CP-AR models with Chib

- Run an estimation of CP-AR(1) models from one to 3 regimes

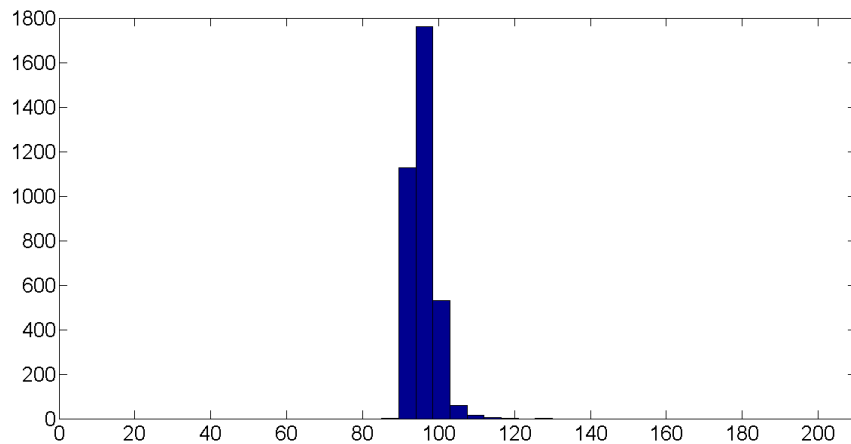
*[Simu MLL] =*

*launch\_CP\_model\_estimations(US\_GDP\_growth,1,3,10000)*

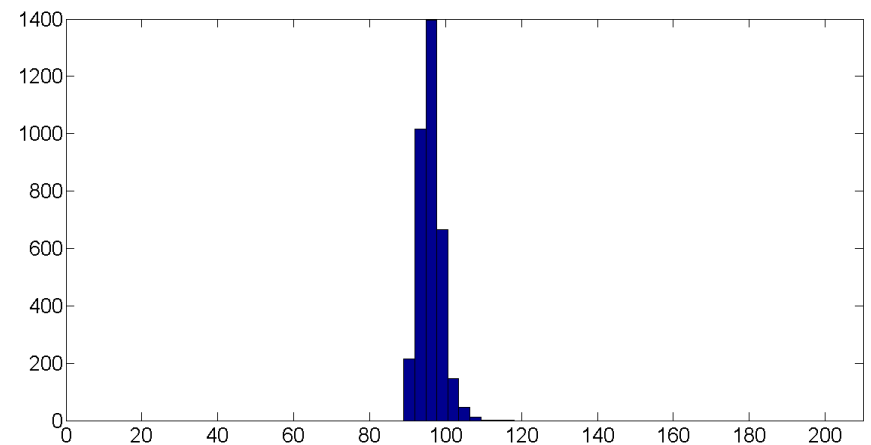
*nb\_MCMC*

- Two regimes

**Griddy-Gibbs**



**Chib**



# Gibbs sampler

- **The model**  $\left\{ \begin{array}{l} y_t = \beta'_{s_t} x_t + \epsilon_t \\ \epsilon_t \sim \text{i.i.d.} N(0, \sigma_{s_t}^2) \end{array} \right.$

- **The prior distributions**

$$\beta_i \sim N(\beta_0, \Sigma_0) \quad \forall i \in [1, K + 1] \quad \text{and} \quad \sigma_i^2 \sim IG(IG_a, IG_b) \quad \forall i \in [1, K + 1]$$

Transition prob. From  $i$  to  $i$ :  $p_{ii} \sim \text{Beta}(\alpha_p, \beta_p)$

Trans. State :  $\left\{ \begin{array}{l} P(s_t = s_{t-1} | s_{t-1}, P) = p_{s_{t-1}, s_{t-1}} \\ P(s_t = s_{t-1} + 1 | s_{t-1}, P) = 1 - p_{s_{t-1}, s_{t-1}} \quad \forall s_{t-1} \in [1, K] \end{array} \right.$

# Gibbs sampler

- The model

$$\begin{cases} y_t = \beta'_{s_t} x_t + \epsilon_t \\ \epsilon_t \sim \text{i.i.d.} N(0, \sigma_{s_t}^2) \end{cases}$$

## Priors in the program : Gibbs\_regression\_chib

```

70 #####
71 ##### Set the Hyper-parameters
72 #####
73 ### Prior :
74 ### beta_i ~ N(beta_0, var_uninformative)
75 ### sigma^-2 ~ G(a,b)
76 ### p_ii ~ Beta(alpha,beta)
77 #####
78 - var_uninformative = 10; %% We fix the variance of each beta equal to var_u
79 - beta_0 = zeros(dimension,1);
80 - inv_Sigma_0 = diag(ones(dimension,1)*(1/var_uninformative));
81 - det_inv_Sigma_0 = det(inv_Sigma_0);
82
83 ### hyper-parameters for the variances
84 - IG_b = 1;
85 - IG_a = 1; } sigma_i^2 ~ IG(IG_a, IG_b)
86
87 ### hyper-parameters for each probability of staying in the same regime
88 - alpha_p = 10;
89 - beta_p = 1; } p_ii ~ Beta(alpha_p, beta_p)

```

$$\beta_i \sim N(\beta_0, \Sigma_0)$$

# Forward-Backward

- **Gibbs step** :  $S_{1:T} | Y_{1:T}, \beta_1, \dots, \beta_{K+1}, \sigma_1^2, \dots, \sigma_{K+1}^2, P$

Program for sampling a state vector : **Forward\_Backward**

Two steps :

- 1) Compute the forward prob.  $f(s_t | Y_{1:t}) \forall t \in [1, T]$
- 2) Sample a state using the decomposition

$$\pi(S_{1:T} | Y_{1:T}) = \pi(s_T | Y_{1:T}) \pi(s_{T-1} | Y_{1:T}, s_T) \dots \pi(s_1 | Y_{1:T}, S_{2:T})$$

```

74
75
76     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
77     %%% Backward algorithm : sampling a state vector
78     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
79 -   sn = zeros(T,1);
80 -   sn(T) = regime;
81 -   for t=T-1:-1:1
82 -       etat_fut = sn(t+1);
83 -       back = forward(t,:)'.*P(:,etat_fut);
84 -       back = back/sum(back);
85 -       sn(t) = multinomialrnd(back);
86 -   end
87
88

```

$$\pi(s_t | Y_{1:T}, S_{t+1:T}) \propto f(s_t | Y_{1:T}) f(s_{t+1} | s_t)$$


# Marginal likelihood

- Local formula :

Likelihood (by F-B)

Priors

$$f(Y_{1:T}) = \frac{f(Y_{1:T} | \beta_1^*, \dots, \beta_{K+1}^*, \sigma_1^{2,*}, \dots, \sigma_{K+1}^{2,*}, P^*) f(\beta_1^*, \dots, \beta_{K+1}^*, \sigma_1^{2,*}, \dots, \sigma_{K+1}^{2,*}, P^*)}{\pi(\beta_1^*, \dots, \beta_{K+1}^*, \sigma_1^{2,*}, \dots, \sigma_{K+1}^{2,*}, P^* | Y_{1:T})}$$

- Third term :

$$\pi(\beta_1^*, \dots, \beta_{K+1}^*, \sigma_1^{2,*}, \dots, \sigma_{K+1}^{2,*}, P^* | Y_{1:T}) = \pi(P^* | Y_{1:T}) \pi(\sigma_1^{2,*}, \dots, \sigma_{K+1}^{2,*} | P^*, Y_{1:T}) \pi(\beta_1^*, \dots, \beta_{K+1}^* | \sigma_1^{2,*}, \dots, \sigma_{K+1}^{2,*}, P^*, Y_{1:T})$$

$$\mathbf{1)} \quad \pi(P^* | Y_{1:T}) = \int \pi(P^* | Y_{1:T}, S_{1:T}) \pi(S_{1:T} | Y_{1:T}) dS_{1:T}$$

$$\pi(P^* | Y_{1:T}) \approx \frac{1}{N} \sum_{i=1}^N \pi(P^* | Y_{1:T}, S_{1:T}^i)$$

# Marginal likelihood

$$\begin{aligned}
 \mathbf{2)} \quad \pi(\sigma_1^{2,*}, \dots, \sigma_{K+1}^{2,*} | P^*, Y_{1:T}) &= \int \pi(\sigma_1^{2,*}, \dots, \sigma_{K+1}^{2,*} | P^*, \beta_1, \dots, \beta_{K+1}, S_{1:T}, Y_{1:T}) \\
 &\quad \pi(S_{1:T}, \beta_1, \dots, \beta_{K+1} | P^*, Y_{1:T}) dS_{1:T} d\beta_1 \dots d\beta_{K+1} \\
 &\approx \frac{1}{G_1} \sum_{i=1}^{G_1} \pi(\sigma_1^{2,*}, \dots, \sigma_{K+1}^{2,*} | P^*, \beta_1^i, \dots, \beta_{K+1}^i, S_{1:T}^i, Y_{1:T})
 \end{aligned}$$



- Run an auxiliary MCMC with fixed  $P^*$

$$\begin{aligned}
 \mathbf{3)} \quad \pi(\beta_1^*, \dots, \beta_{K+1}^* | P^*, \sigma_1^{2,*}, \dots, \sigma_{K+1}^{2,*}, Y_{1:T}) &= \int \pi(\beta_1^*, \dots, \beta_{K+1}^* | S_{1:T}, \sigma_1^{2,*}, \dots, \sigma_{K+1}^{2,*}, Y_{1:T}) \\
 &\quad \pi(S_{1:T} | P^*, \sigma_1^{2,*}, \dots, \sigma_{K+1}^{2,*}, Y_{1:T}) dS_{1:T} \\
 &\approx \frac{1}{G_2} \sum_{i=1}^{G_2} \pi(\beta_1^*, \dots, \beta_{K+1}^* | S_{1:T}^i, \sigma_1^{2,*}, \dots, \sigma_{K+1}^{2,*}, Y_{1:T})
 \end{aligned}$$



- Run second auxiliary MCMC with fixed  $P^*, \sigma_1^{2,*}, \dots, \sigma_{K+1}^{2,*}$

# Model selection

- Run an estimation of CP-AR(1) models from one to 3 regimes

*[Simu MLL] =*

*launch\_CP\_model\_estimations(US\_GDP\_growth,1,3,10000)*

Uninformative prior :

$$\beta_i \sim N(\underline{0}, 1000I_d) \quad \sigma_i^{-2} \sim G(0.01, 100) \quad p_{ii} \sim \text{Beta}(0.5, 0.5)$$

#Regime	1	2	3
MLL	-275.57	-269.25	-275.10

Informative prior :

$$\beta_i \sim N(\underline{0}, I_d) \quad \sigma_i^{-2} \sim G(1, 1) \quad p_{ii} \sim \text{Beta}(0.5, 0.5)$$

#Regime	1	2	3
MLL	-265.28	-249.96	-243.52

# Model selection

- Choose your prior according to **'your break sensitivity'**
- Or use another criterion such as the predictive likelihood

→ **Less impacted by the prior distributions**

$$f(Y_{t+1:T} | Y_{1:t}) = \frac{f(Y_{1:T})}{f(Y_{1:t})}$$

**No prior distributions!  
Impact through the  
posterior**

$$= \frac{f(Y_{1:T} | \theta^*) f(\theta^*)}{\pi(\theta^* | Y_{1:T})} \frac{\pi(\theta^* | Y_{1:t})}{f(Y_{1:t} | \theta^*) f(\theta^*)}$$

$$= \frac{f(Y_{1:T} | \theta^*)}{f(Y_{1:t} | \theta^*)} \frac{\pi(\theta^* | Y_{1:t})}{\pi(\theta^* | Y_{1:T})}$$