Chapter 8

The Monte Carlo origin (MCMC)

Markov Chain Monte Carlo (MCMC) has two components:

- The Monte Carlo,
- The Markov Chain.

CLT

If we sample $X_1, X_2, \ldots, X_n$ random variables that are independent and identically distribution the

$$\frac{X_1 + X_2 + \ldots + X_n}{n} \rightarrow \mathbf{N}(\mu, \frac{\sigma}{\sqrt{n}}),$$

where $\mu = \mathbb{E}[X]$. How does this help analyzing output from a Monte Carlo algorithm?

A Markov chain, X_t, is a time series with the following property:

Memoryless

Given $X_0, X_1, \ldots, X_t$ the distribution of X_{t+1} satisfies

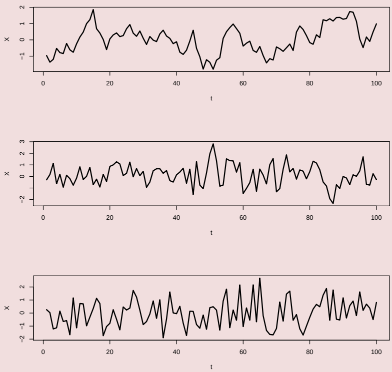
$$f(X_{t+1}|X_t, X_{t-1}, \ldots, X_0) = f(X_{t+1}|X_t).$$

Figure: Andrey Markov

Three examples of AR(1) processes:

$$X_t = aX_{t-1} + \epsilon_t$$
$$\epsilon_t \sim N(0, \sigma)$$

1. $a = 0.9, \sigma = \sqrt{1 - 0.9^2}$
2. $a = 0.1, \sigma = \sqrt{1 - 0.1^2}$
3. $a = 0, \sigma = 1$



Stationary distribution

- For all three processes if we can thin the series:

$$X_T, X_{2T}, X_{3T}, \ldots$$

  where T is large.
- It turns out that all three series have the same **stationary distribution**, p.

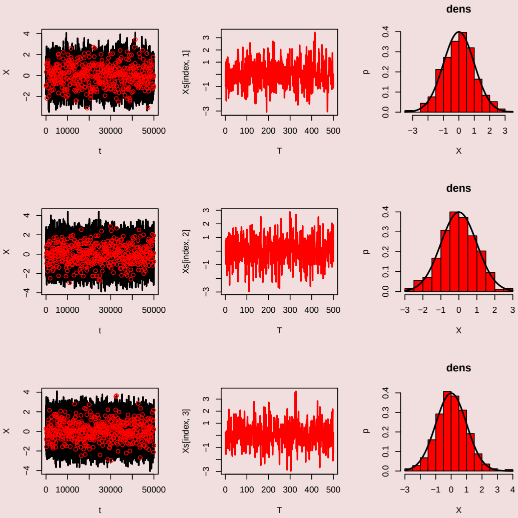
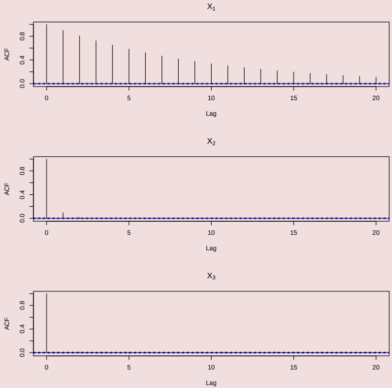


Figure: $T = 500$

For time series one often look at the autocorrelation (ACF) function:



Transition

A Markov chain has a transition density $f(x|X_t)$. The transition density is the density of X_{t+1} given you know X_t.

Stationary

A Markov chain has a stationary density $p(x)$. The stationary density is the density of observations taken far enough from each other.

Generate a Markov Chain with stationary distribution p That is we choose a density f such that the stationary distribution is p.

MCMC the origin

THE JOURNAL OF CHEMICAL PHYSICS VOLUME 21, NUMBER 6 JUNE, 1953

Equation of State Calculations by Fast Computing Machines

NICHOLAS METROPOLIS, ARIANNA W. ROSENBLUTH, MARSHALL N. ROSENBLUTH, AND AUGUSTA H. TELLER,
Los Alamos Scientific Laboratory, Los Alamos, New Mexico

AND

EDWARD TELLER,* *Department of Physics, University of Chicago, Chicago, Illinois*

(Received March 6, 1953)

A general method, suitable for fast computing machines, for investigating such properties as equations of state for substances consisting of interacting individual molecules is described. The method consists of a modified Monte Carlo integration over configuration space. Results for the two-dimensional rigid-sphere system have been obtained on the Los Alamos MANIAC and are presented here. These results are compared to the free volume equation of state and to a four-term virial coefficient expansion.

Biometrika (1970), **57**, 1, *p.* 97
Printed in Great Britain

Monte Carlo sampling methods using Markov chains and their applications

BY W. K. HASTINGS
University of Toronto

SUMMARY

A generalization of the sampling method introduced by Metropolis *et al.* (1953) is presented along with an exposition of the relevant theory, techniques of application and methods and difficulties of assessing the error in Monte Carlo estimates. Examples of the methods, including the generation of random orthogonal matrices and potential applications of the methods to numerical problems arising in statistics, are discussed.

1. INTRODUCTION

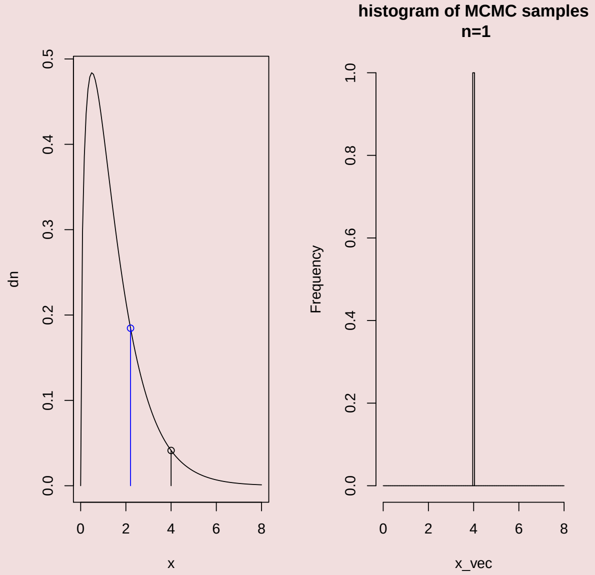
Metropolis-Hastings algorithm

Here needs better explanation. One iteration of a symmetric random walk

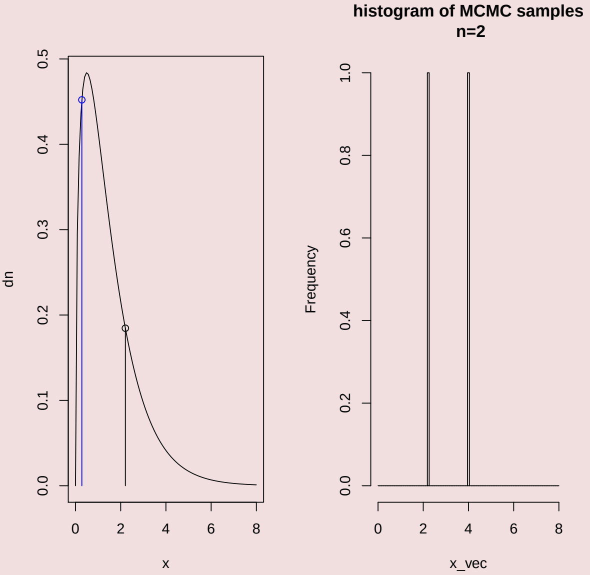
- Generate a symmetric variable centered around the previous value, most common Normal $X^* \sim N(X^{old}, \sigma)$.
- Generate $U \sim U[0,1]$.
- The new value is

$$X^{new} = \begin{cases} X^* & \text{if } U \leq \frac{p(X^*)}{p(X^{old})}, \\ X^{old} & \text{otherwise.} \end{cases}$$

In Action

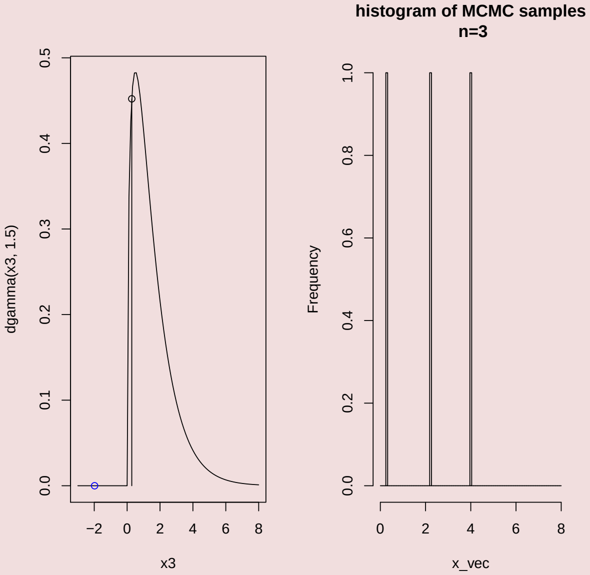

dn
0.0
0.1
0.2
0.3
0.4
0.5
0
2
4
6
8
x
histogram of MCMC samples
n=1
Frequency
0.0
0.2
0.4
0.6
0.8
1.0
0
2
4
6
8
x_vec

In Action



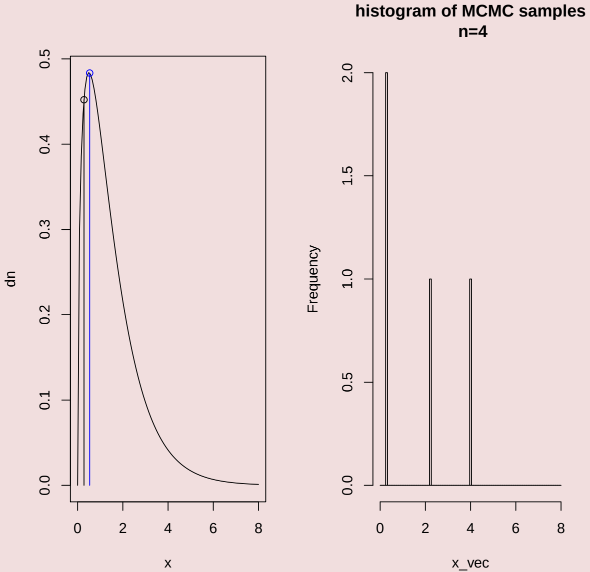
dn
0.0
0.1
0.2
0.3
0.4
0.5
0
2
4
6
8
x
histogram of MCMC samples
n=2
Frequency
0.0
0.2
0.4
0.6
0.8
1.0
0
2
4
6
8
x_vec

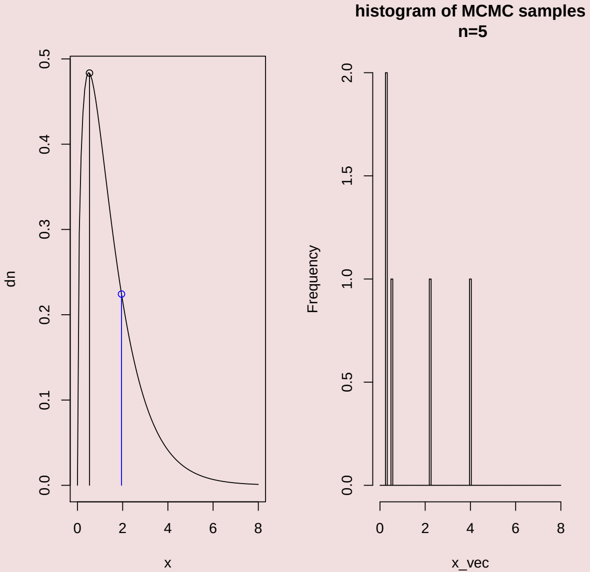
In Action

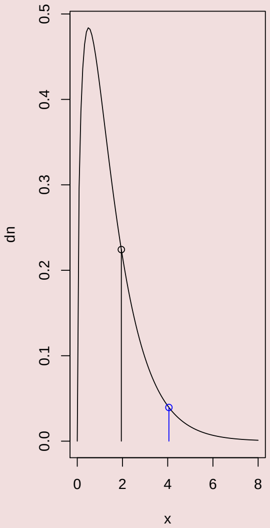


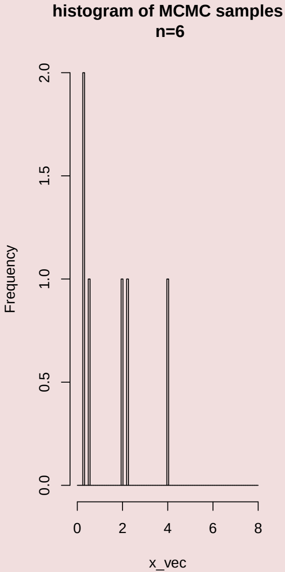
histogram of MCMC samples
n=3
dgamma(x3, 1.5)
x3
Frequency
x_vec
0.0
0.1
0.2
0.3
0.4
0.5
−2
0
2
4
6
8
0.0
0.2
0.4
0.6
0.8
1.0
0
2
4
6
8

In Action

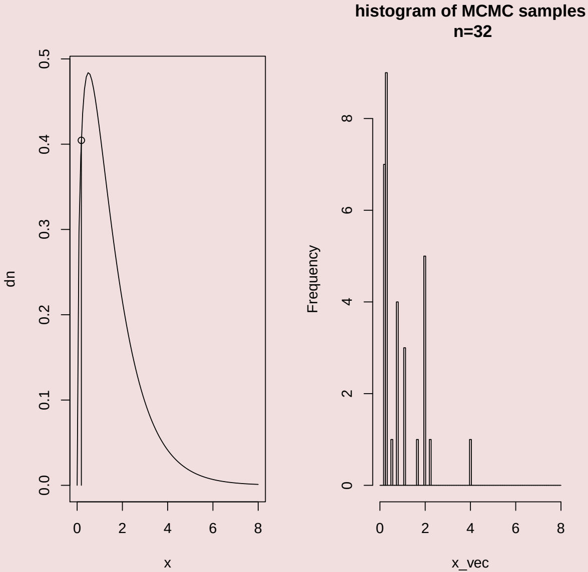

histogram of MCMC samples
n=4
dn
x
0.0
0.1
0.2
0.3
0.4
0.5
0
2
4
6
8
Frequency
x_vec
0.0
0.5
1.0
1.5
2.0

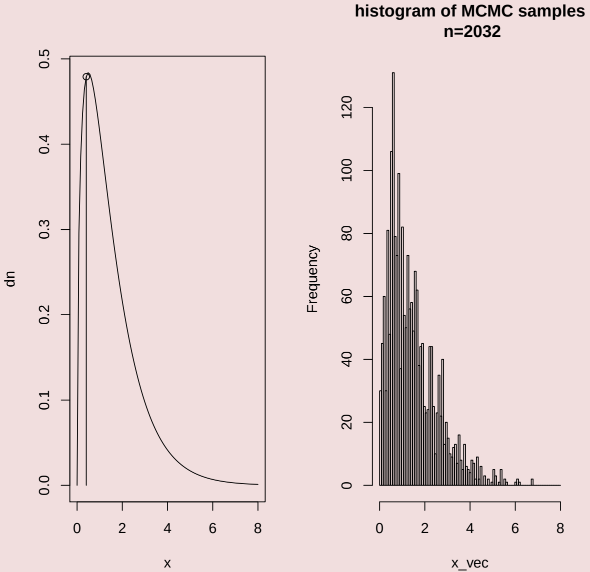

histogram of MCMC samples
n=5
dn
0.0
0.1
0.2
0.3
0.4
0.5
0
2
4
6
8
x
Frequency
0.0
0.5
1.0
1.5
2.0
x_vec


dn
0.0
0.1
0.2
0.3
0.4
0.5
0
2
4
6
8
x

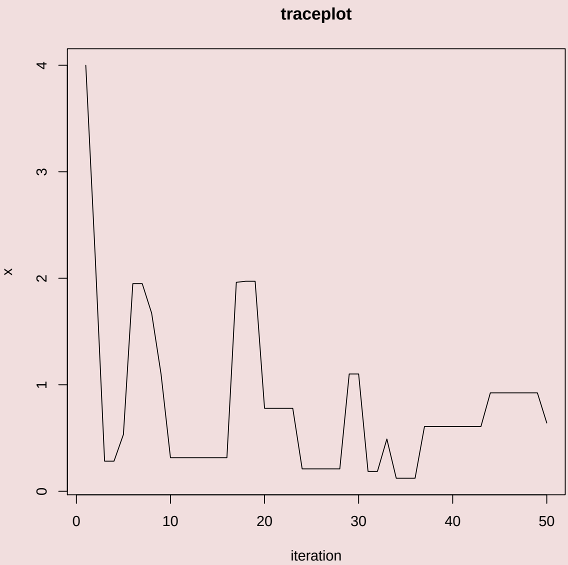

histogram of MCMC samples
n=6
Frequency
0.0
0.5
1.0
1.5
2.0
0
2
4
6
8
x_vec

In Action

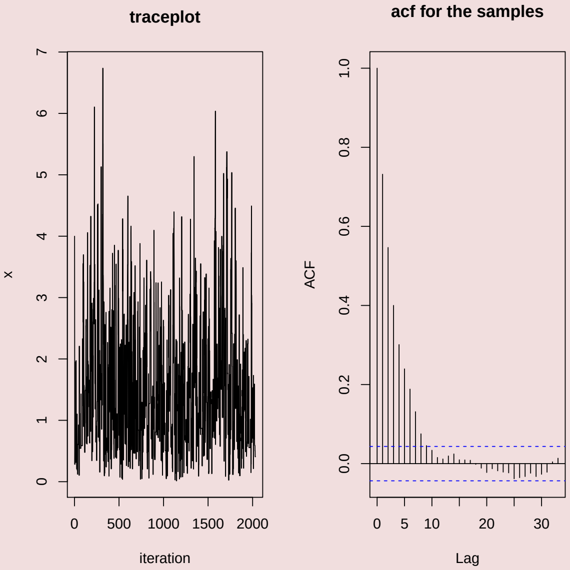

histogram of MCMC samples
n=32
dn
0.0
0.1
0.2
0.3
0.4
0.5
0
2
4
6
8
x
Frequency
0
2
4
6
8
0
2
4
6
8
x_vec


histogram of MCMC samples
n=2032
dn
x
0.0
0.1
0.2
0.3
0.4
0.5
0
2
4
6
8
Frequency
x_vec
0
20
40
60
80
100
120

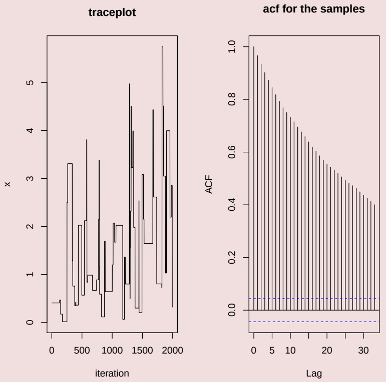
Traceplot



Traceplot


traceplot
acf for the samples
x
0
1
2
3
4
5
6
7
0
500
1000
1500
2000
iteration
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
20
30
Lag

- The choice of σ extremely important for the mixing (the ACF) of the algorithm.
- In stan there are several parameters that are set in the MCMC this is done during the **warmup**.



	culture	population	contact	total_tools
1	Malekula	1100	low	13
2	Tikopia	1500	low	22
3	Santa Cruz	3600	low	24
4	Yap	4791	high	43
5	Lau Fiji	7400	high	33
6	Trobriand	8000	high	19
7	Chuuk	9200	high	40
8	Manus	13000	low	28
9	Tonga	17500	high	55
10	Hawaii	275000	low	71

Poisson model

$$
\begin{aligned}
tools_i &\sim Po(\lambda_i) \\
g(\lambda_i) &= \alpha + \log(population_i)\beta_p + contact_i\beta_c \\
\alpha &\sim N(0, 10) \\
\beta_p &\sim N(0, 10) \\
\beta_c &\sim N(0, 10)
\end{aligned}
$$

- Density:

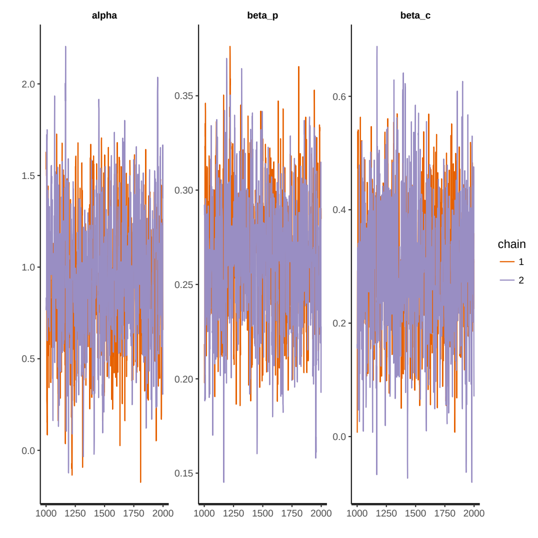
$$p(\alpha, \beta_c, \beta_p | t) \propto N(\alpha; 0, 10) N(\beta_c; 0, 10) N(\beta_p; 0, 10) \cdot \prod_{i=1}^{n} Po(t_i; g^{-1}(\alpha + \log(p_i)\beta_p + c_i\beta_c)).$$

Hence,

$$p(\alpha, \beta_c, \beta_p | t) = N(\alpha; 0, 10) N(\beta_c; 0, 10) N(\beta_p; 0, 10) \cdot \prod_{i=1}^{n} Po(t_i; g^{-1}(\alpha + \log(p_i)\beta_p + c_i\beta_c)) \cdot \left(\int N(\tilde{\alpha}; 0, 10) N(\tilde{\beta}_c; 0, 10) N(\tilde{\beta}_p; 0, 10) \cdot \prod_{i=1}^{n} Po(t_i; g^{-1}(\tilde{\alpha} + \log(p_i)\tilde{\beta}_p + c_i\tilde{\beta}_c)) d\tilde{\alpha} d\tilde{\beta}_p d\tilde{\beta}_c \right)^{-1}$$

- Direct sampling (Monte Carlo) will requires evaluation of $p(\alpha, \beta_c, \beta_p)$.
- What does MCMC

Checking the chains



Checking the chains

```
print(simple_fit, probs=c(0.1,0.9), digits=2,pars=c("alpha","beta_p","beta_c"))
```

```
Inference for Stan model: poisson_stan.
2 chains, each with iter=2000; warmup=1000; thin=1;
post-warmup draws per chain=1000, total post-warmup
draws=2000.

        mean se_mean   sd  10%  90% n_eff Rhat
alpha   0.94    0.01 0.35 0.49 1.40   627    1
beta_p  0.26    0.00 0.03 0.22 0.31   655    1
beta_c  0.30    0.00 0.12 0.15 0.45   797    1
```

Formulas

Let X_{ij} be samples $i = 1, 2, \ldots, n$ and chains $j = 1, 2, \ldots, m$.

$$n_{eff} = \frac{n}{1 + \sum_{i=1}^{\infty} acf(i)}$$

Let X_{ij} be samples $i = 1, 2, \ldots, n$ and chains $j = 1, 2, \ldots, m$.

$$W = \frac{1}{m} \sum_{i=1}^{m} S_i^2,$$

where

$$s_j^2 = \frac{1}{n-1} \sum_{i=1}^{n} (X_{ij} - \bar{X}_j)^2.$$

Let X_{ij} be samples $i = 1, 2, \ldots, n$ and chains $j = 1, 2, \ldots, m$.

$$B = \frac{n}{m-1} \sum_{i=1}^{m} (\bar{\bar{X}} - \bar{X}_j)^2$$

where

$$\bar{\bar{X}} = \frac{1}{m} \sum_{j=1}^{m} \bar{X}_j.$$

Between Chain variance

Let X_{ij} be samples $i = 1, 2, \ldots, n$ and chains $j = 1, 2, \ldots, m$.

$$\hat{\mathbb{V}}[X] = (1 - \frac{1}{n})W + \frac{1}{n}B.$$

where

$$\hat{R} = \sqrt{\frac{\hat{\mathbb{V}}[X]}{W}}$$

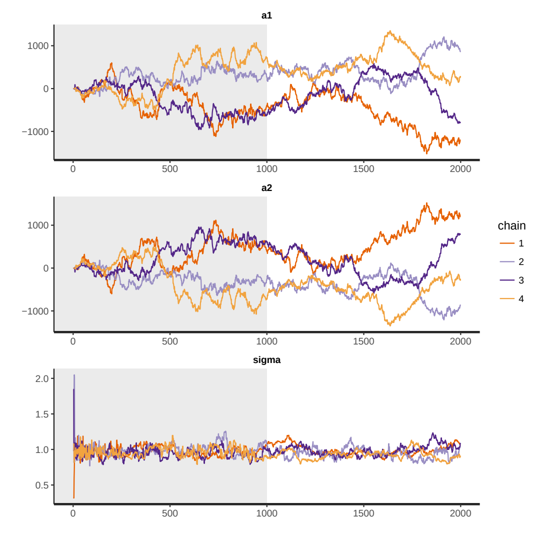
when $\hat{R}$ is high above 1.1, then it indicates that the chain does not have the same distribution.

Detour, dumb model

$$y_i \sim N(\mu, \sigma)$$
$$\mu = \alpha_1 + \alpha_2$$

Non proper prior $p(\alpha_1, \alpha_2, \sigma) \propto 1$.

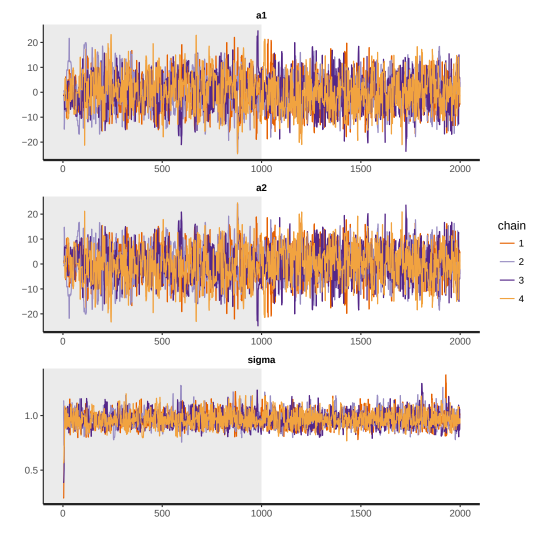
Detour, how does failure look

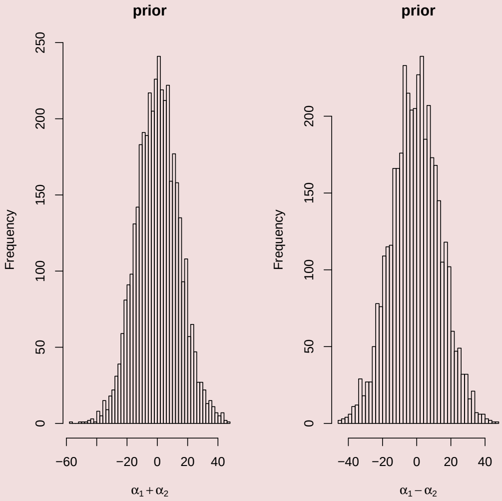


Detour, dumb model, dumb fix

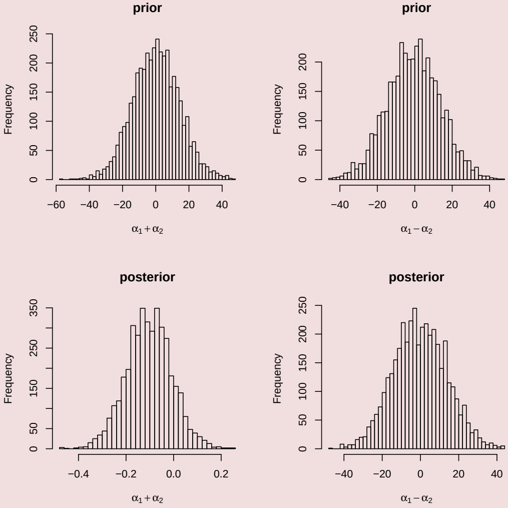
$$
\begin{aligned}
y_i &\sim N(\mu, \sigma) \\
\mu &= \alpha_1 + \alpha_2 \\
\alpha_1 &\sim N(0, 10) \\
\alpha_2 &\sim N(0, 10)
\end{aligned}
$$

trace post fix



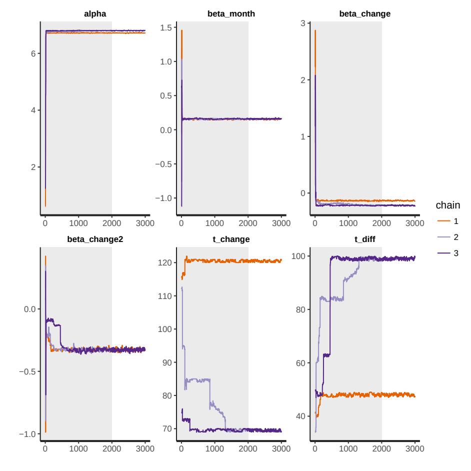

prior
Frequency
0
50
100
150
200
250
−60
−20
0
20
40
$\alpha_1 + \alpha_2$
prior
Frequency
0
50
100
150
200
−40
−20
0
20
40
$\alpha_1 - \alpha_2$

posterior

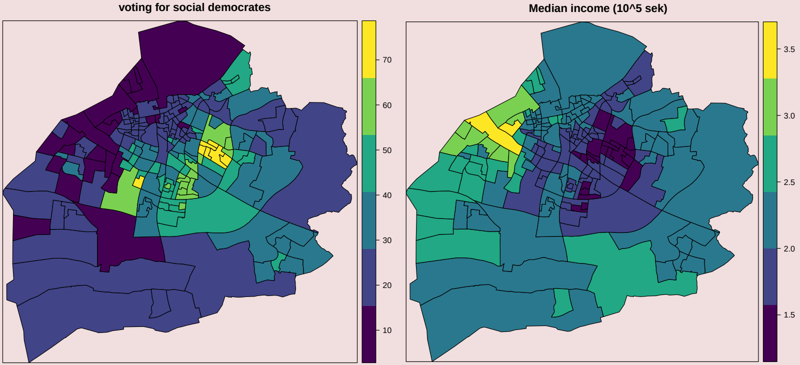


prior
Frequency
0
50
100
150
200
250
−60
−40
−20
0
20
40
$\alpha_1+\alpha_2$
prior
Frequency
0
50
100
150
200
−40
−20
0
20
40
$\alpha_1-\alpha_2$
posterior
Frequency
0
50
150
250
350
−0.4
−0.2
0.0
0.2
$\alpha_1+\alpha_2$
posterior
Frequency
0
50
100
150
200
250
−40
−20
0
20
40
$\alpha_1-\alpha_2$

Checking the chains, failure number 2

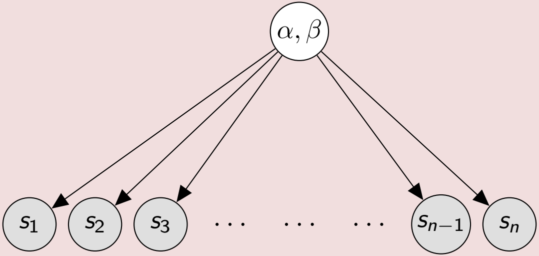


Voting in Malmö, data

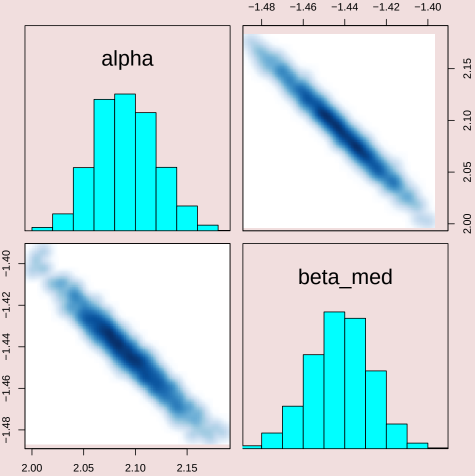


$$
\begin{aligned}
s_i &\sim bin(n_i, p_i), \\
g(p_i) &= \alpha + med_i \beta, \\
\alpha &\sim N(0, 10) \\
\beta &\sim N(0, 10)
\end{aligned}
$$

Independent model DAG



Posterior parameter



Predictions

- By the model the prediction given the data is

$$\hat{Y}_i \sim Bin(n_i, p_i),$$
$$p_i \sim p(\cdot | y_1, y_2, \ldots, y_n)$$

Predictions

- By the model the prediction given the data is

$$\hat{Y}_i \sim Bin(n_i, p_i),$$
$$p_i \sim p(\cdot|y_1, y_2, \ldots, y_n)$$

- The variance is:

$$V[\hat{Y}|p_i, n_i] = n_i(1 - p_i)p_i$$
$$V[\frac{\hat{Y}}{n_i}|p_i, n_i] = \frac{(1 - p_i)p_i}{n_i}$$

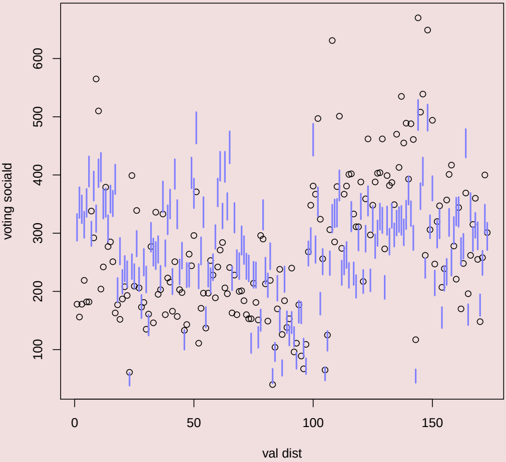


Figure: Predicition by district

Overdispertion

- Both binomial and Poisson has only one parameter.
- These models are extremely sensitivity to incorrect parameter.
- They can not adjust it variance to the data.

Solution

- This is typically solved by overdispersion model. Like Beta-binomial.

Solution

- This is typically solved by overdispersion model. Like Beta-binomial.
- For each observation one adds a random non-negative parameter:

$$p(y_i|n_i) = \int Bin(y_i|n_i, p_i)h(p_i|p,\theta)p(p,\theta)dp_i dp d\theta,$$

Then one puts covariates on p not p_i.

Solution

- This is typically solved by overdispersion model. Like Beta-binomial.
- For each observation one adds a random non-negative parameter:

$$p(y_i|n_i) = \int Bin(y_i|n_i, p_i)h(p_i|p, \theta)p(p, \theta)dp_i dp d\theta,$$

  Then one puts covariates on p not p_i.
- overdisperation is typically a multilevel model.

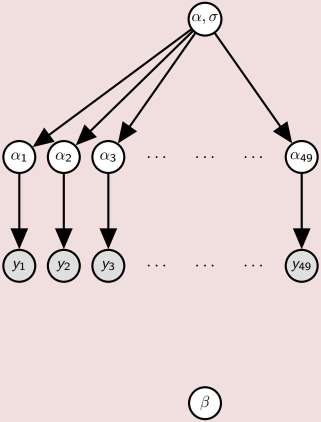
$$
\begin{aligned}
y_i &\sim Bin(n_i, p_i) \\
g(p_i) &\sim \alpha_0 + med_i \beta + Z_i \\
Z_i &\sim N(0, \sigma) \\
\alpha_0 &\sim N(0, 10) \\
\sigma &\sim HC(0, 5).
\end{aligned}
$$

$$
\begin{aligned}
y_i &\sim Bin(n_i, p_i) \\
g(p_i) &\sim \alpha_0 + med_i\beta + Z_i \\
Z_i &\sim N(0, \sigma) \\
\alpha_0 &\sim N(0, 10) \\
\sigma &\sim HC(0, 5).
\end{aligned}
$$

or equivalently

$$
\begin{aligned}
y_i &\sim Bin(n_i, p_i) \\
g(p_i) &\sim \alpha_i + med_i\beta \\
\alpha_i &\sim N(\alpha_0, \sigma) \\
\alpha_0 &\sim N(0, 10) \\
\sigma &\sim HC(0, 5).
\end{aligned}
$$

Hierarchicals model DAG



PI for model

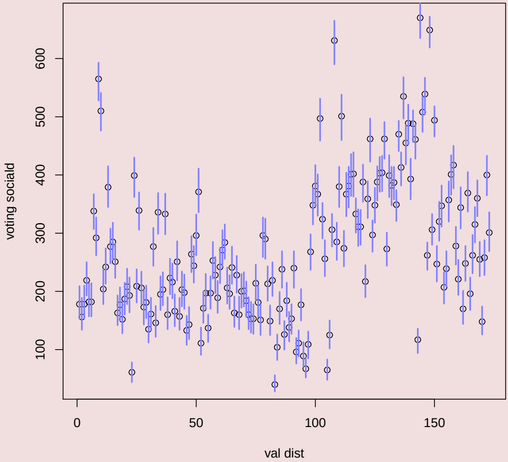


Figure: Prediction by district multilevel

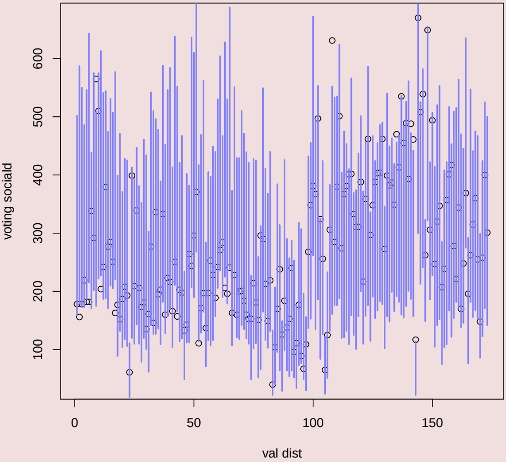


Figure: Prediction unconditional by district multilevel

Heuristically

- The variance is:

$$V[\hat{Y}|n_i] \approx n_i(1-\hat{p}_i)\hat{p}_i + n_i^2\tilde{\sigma}$$
$$V[\frac{\hat{Y}}{n_i}|n_i] \approx \frac{(1-\hat{p}_i)\hat{p}_i}{n_i} + \tilde{\sigma}$$

Where $\tilde{\sigma}$ is the variation from

PI for multilevel without cheating

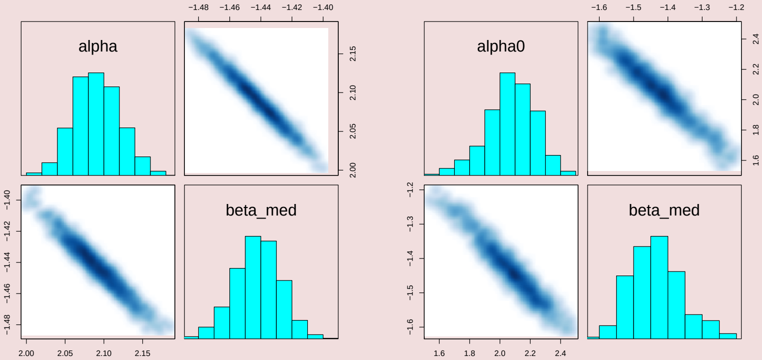


Figure: Look at parameter certainty