

ベイズ統計学

理論編

宇畑 優太(1260404)

2024-07-21

目次

準備	1
1 Beyes	1
1.1 ベイズの定理	1
1.2 事後確率分布	3
1.3 Kernel(θ)	3
1.4 正規化定数	5
1.5 ベイズ推論の難点	6
2 MCMC	6
2.1 モンテカルロ積分	7
2.2 マルコフ連鎖	7
2.3 メトロポリス・ヘイスティング法 (MH 法)	10
3 stan	13
4 補足 信頼区間と予測区間の違い	18

準備

1 Beyes

1.1 ベイズの定理

$P(A \cap B)$ という同時確率を考える. 確率の性質より以下の等式が成り立つ.

$$\begin{aligned} P(A \cap B) &= P(A|B)P(B) \\ &= P(B|A)P(A) \end{aligned}$$

上記の等式について、 $P(B)$ を移行することで**ベイズの定理**が導出できる。

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

A と B を D と θ に変えておく。

$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{P(D)}$$

$$\text{事後確率} = \frac{\text{尤度} \cdot \text{事前確率}}{\text{周辺尤度}}$$

周辺尤度 $P(D)$ について、以下のように変形が可能である。

$$\begin{aligned} \sum_{i=1}^N P(D|\theta_i)P(\theta_i) &= \frac{\sum_{i=1}^N P(D \cap \theta_i)}{P(\theta_i)} P(\theta_i) \\ &= \sum_{i=1}^N P(D \cap \theta_i) \\ &= P(D \cap \theta_1) + P(D \cap \theta_2) + \dots + P(D \cap \theta_N) \\ &= P(D) \end{aligned}$$

注意

わかりやすさを重視して式変形の順序を逆にしている

このような変形 ($\sum_y p(x, y) = p(x)$) を**周辺化**と呼ぶ。

上記の表現を用いて事後確率を再び記載する

$$\begin{aligned} \text{事後確率} &= \\ P(\theta|D) &= \frac{P(D|\theta)P(\theta)}{\sum_{i=1}^N P(D|\theta_i)P(\theta_i)} \end{aligned}$$

ここで用語を整理しておく

- 事前確率：データが得られる前に想定された確率
- 事後確率：データが得られた後に想定する確率
- ベイズ更新：ベイズの定理を用いて事前確率を事後確率に更新すること

事後 (事前) 確率があるのなら、当然その確率の確率分布たる**事後 (事前) 分布**なるものも存在する。

- 事前確率分布：データが得られる前に想定する分布
- 事後確率分布：データが得られた後に想定する分布

1.2 事後確率分布

事後確率分布は以下のように表現できる.

$$\begin{aligned}\text{事後確率分布} &= \frac{\text{尤度関数} \cdot \text{事前確率分布}}{\text{周辺尤度関数}} \\ f(\theta|D) &= \frac{f(D|\theta) \cdot f(\theta)}{f(D)} \\ &= \frac{f(D|\theta) \cdot f(\theta)}{\int_{-\infty}^{\infty} f(D|\theta)f(\theta)d\theta}\end{aligned}$$

上記の等式が成り立つのは, 事後確率の等式が成り立つのと同じ理屈である.

事前分布に関して何らかの想定が置けないときは**無情報事前分布**を指定する. 例えば分散が非常に大きい正規分布など. 分散が大きいということは, 「どの値になる確率にも, 満遍なく低い確率を割り当てる」という状態である.

! 2024_7_20 上記を修正

- 昔は確かにそのようにしていました. しかし, この方法で構成した「無情報」事前分布は全然無情報ではない(「分散が大きい」という「強い」情報がある)ので, 無情報事前分布はつかってはいけません.
- 複数の事前分布を使い感度分析を行う方法^{*1}と, データから得られた統計量を使う方法がある. 後者は stan の規定事前分布

- 確率分布のパラメータ (θ) を含む部分のことを**カーネル**と呼ぶ. 上記の式では分子の尤度関数・事前確率分布 ($f(D|\theta) \cdot f(\theta)$) に当たる. $\text{kernel}(\theta)$ と表記する.
- 確率分布のパラメータ (θ) を含まない部分のことを**正規化定数**と呼ぶ. 上記の式では分母の周辺尤度関数 ($f(D)$) に当たる. 読んで字の如く**正規化**をしてくれるのである. 正規化とは事後確率分布の確率密度関数の積分値を 1 になるように調整することであり. 周辺尤度関数 ($f(D)$) を計算することで正規化ができる.

⚠ 注意

詳しくは「ベイズ推論の難点」に書いた

1.3 Kernel(θ)

事後分布の具体的な計算に入ってく. サンプルサイズが N のデータ x_1, x_2, \dots, x_N が得られたとする. 正規分布を用いた確率モデルを想定する.

$$X \sim \text{Normal}(\theta, 1)$$

^{*1} 感度分析に関する記述は教科書の 2 部 6 章にある

計算を簡略化するために母分散は 1 であるとし、パラメータ θ の事後分布を得ることを目指す。

- 事前分布

事前分布としては分散が 100^2 の正規分布を想定する。無情報分布というには分散が小さすぎるが、計算を簡単にするためにこの値を使う。

! 2024_7_20 上記を修正

- 上でコメントしたとおり、こんなに大きな分散を想定するということは、それはもはや「無」情報ではありません。

この事前分布を数式で表すことにする。パラメータ θ の事前分布の確率密度関数 $f(\theta)$ は以下のようになる

$$f(\theta) = \frac{1}{\sqrt{20000\pi}} \exp\left(-\frac{\theta^2}{20000}\right)$$

$(x-\theta)^2$ ではなく単に θ^2 となっているのは、事前確率分布はデータ X が得られる前に想定されるものだから。(多分,,,)

! 2024_7_20 上記を修正

- これは誤りです。
 $\theta \sim \text{Normal}(0, 100)$ (100 は標準偏差)
を θ の事前分布と仮定しているからです。

- 尤度関数

尤度関数は次のように定義されている。

「一般に、パラメータ θ を持つ密度関数 $f_x(x; \theta)$ から無作為標本 $\{x_1, x_2, \dots, x_N\}$ が得られたときに、その同時密度関数を θ の関数とみなして

$$L(\theta|x) = f_x(x_1|\theta)f_x(x_2|\theta) \times \dots \times f_x(x_N|\theta)$$

を**尤度関数**と呼ぶ」

! 2024_7_20 上記を修正

- 他の箇所でも $(D | \theta)$ のように表記しているので、 $(x | \theta)$ のように縦線で書くべきです。(; でもいいが、一貫した書き方をすべきという意味)
- 普通は $L(\theta | x)$ と表記します。

では、この尤度関数を数式で表現することを目指す。平均が θ で分散が 1 である正規分布の確率密度関数は以下のようになる。

$$\text{Normal}(X|\theta, 1) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x-\theta)^2}{2}\right)$$

今回は N 個のデータが得られているため、尤度は「 x_1 というデータが得られる確率」 \times 「 x_2 というデータが得られる確率」 $\times \dots$ と N 回分の結果をかけ合わせる必要がある。律儀に書くと以下の通り

$$\text{尤度関数} = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x_1 - \theta)^2}{2}\right) \times \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x_2 - \theta)^2}{2}\right) \times \dots \times \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x_N - \theta)^2}{2}\right)$$

総乗の記号 \prod を使い、尤度関数 $f(D|\theta)$ を以下のように表記する。

$$f(D|\theta) = \prod_{i=1}^N \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x_i - \theta)^2}{2}\right)$$

突然ではあるが、パラメータ θ の事後確率密度関数 $f(\theta|D)$ は、「尤度 \times 事前分布」つまりは $\text{Kernel}(\theta)$ に比例する。

$$\begin{aligned} f(\theta|D) &\propto f(D|\theta)f(\theta) \\ &= \left[\prod_{i=1}^N \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x_i - \theta)^2}{2}\right) \right] \cdot \left[\frac{1}{\sqrt{20000\pi}} \exp\left(-\frac{\theta^2}{20000}\right) \right] \\ &= \text{Kernel}(\theta) \end{aligned}$$

データが得られた後であるため、 $\text{Kernel}(\theta)$ は θ の関数である。(尤度関数の定義でも似たようなことを言うてる。)

分母たる周辺尤度を無視しているのは 2 つ理由がある。

1. 周辺尤度の計算がめんどくさい
2. 周辺尤度を直接求めなくても良い理由がある s

1.4 正規化定数

周辺化とは何であったかを思い出す。

- 離散の場合 $\sum_y p(x, y) = p(x)$
- 連続の場合 $\int_y p(x, y) dy = p(x)$

であった。周辺化を用いることで周辺尤度の確率密度関数 $f(D)$ は次のように変形できる

$$\begin{aligned} \int_{-\infty}^{\infty} f(D|\theta)f(\theta)d\theta &= \int_{-\infty}^{\infty} \frac{f(D, \theta)}{f(\theta)} f(\theta)d\theta \\ &= \int_{-\infty}^{\infty} f(D, \theta)d\theta \\ &= f(D) \end{aligned}$$

⚠ 注意

わかりやすさを重視して式変形の順序を逆にしている

$\int_{-\infty}^{\infty} f(D|\theta)f(\theta)d\theta$ を計算するのは非常に困難である。積分計算によって θ が消えるためもはや θ の関数ではなく、データと確率モデルが与えられた後には定数になる。

1.5 ベイズ推論の難点

ベイズの定理を用いることで事後確率分布 (正確にはその確率密度関数) を得ることができた

$$\begin{aligned}\text{事後確率分布} &= \frac{\text{事前確率分布} \cdot \text{尤度関数}}{\text{周辺尤度}} = \frac{\text{Kernel}(\theta)}{\text{正規化定数}} \\ f(\theta|D) &= \frac{f(D|\theta)f(\theta)}{f(D)} \\ &= \frac{f(D|\theta)f(\theta)}{\int_{-\infty}^{\infty} f(D|\theta)f(\theta)d\theta} \\ &= \frac{\left[\prod_{i=1}^N \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x_i-\theta)^2}{2}\right) \right] \cdot \left[\frac{1}{\sqrt{20000\pi}} \exp\left(-\frac{\theta^2}{20000}\right) \right]}{\int_{-\infty}^{\infty} \left[\prod_{i=1}^N \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x_i-\theta)^2}{2}\right) \right] \cdot \left[\frac{1}{\sqrt{20000\pi}} \exp\left(-\frac{\theta^2}{20000}\right) \right] d\theta}\end{aligned}$$

ここで正規化定数の意味を思い出してほしい。

i 正規化定数

「読んで字の如く**正規化**をしてくれるのである。正規化とは事後確率分布の確率密度関数の積分値が 1 になるように調整することであり、周辺尤度関数 ($f(D)$) を計算することで正規化ができる。」

ここで、事後確率分布の確率密度関数を用いて「パラメータ θ が a から B の範囲に入る確率」を計算したいとする。つまり

$$\begin{aligned}P(a \leq \theta \leq b) &= \int_a^b f(\theta|D)d\theta \\ &= \int_a^b \left(\frac{\left[\prod_{i=1}^N \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x_i-\theta)^2}{2}\right) \right] \cdot \left[\frac{1}{\sqrt{20000\pi}} \exp\left(-\frac{\theta^2}{20000}\right) \right]}{\int_{-\infty}^{\infty} \left[\prod_{i=1}^N \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x_i-\theta)^2}{2}\right) \right] \cdot \left[\frac{1}{\sqrt{20000\pi}} \exp\left(-\frac{\theta^2}{20000}\right) \right] d\theta} \right) d\theta\end{aligned}$$

を計算するのであるが、最後の項の分母と分子に注目してほしい。分子が分母 (**正規化定数**) よりも大きい値をとることは無い、ということが見てわかる。仮に全区間で全体を積分したとしても、分母と分子は同等の値となり積分値は 1 になるであろうことは容易に想像がつく。事後確率分布 ($f(\theta|D)$) を全区間で積分した答えが 1(になりそう) ということは、 $f(\theta|D)$ を確率密度関数として扱うことができるということである。これが**正規化定数**の意味である。

ここまでの議論で事後分布を得ることに成功した。あとは積分するだけなのだが、これが非常にめんどくさい。だけでなく、できないかもしれない。それを解決してくれるのが **MCMC** である。

2 MCMC

MCMC はマルコフ連鎖モンテカルロ法を省略したものである。マルコフ連鎖を利用して乱数を生成し、その乱数を使ってモンテカルロ積分をする、という一連のプロセスを指す。

議論がややこしいので、MCMC についてのおおまかな流れを先に記載する

事後分布を普通の方法で積分するのはめんどくさい

⇓

代わりに、モンテカルロ積分をしよう。 $\frac{\sum_{i=1}^{1000} \hat{\theta}_i}{1000}$ を計算する。これで θ の期待値がわかる。

⇓

モンテカルロ積分をするには、事後分布に従う乱数 $\hat{\theta}$ を得る必要がある。

⇓

乱数はマルコフ連鎖を利用して生成する

2.1 モンテカルロ積分

MCMC を行う最大の理由は事後分布が非常に複雑な数式となってしまう、積分が困難であるからである。積分ができないということは θ の期待値も計算できないということである。

では、普通に積分をすることは諦めて、代わりにモンテカルロ積分をしよう。もしパラメータ θ の事後分布に従う乱数 $\hat{\theta}$ が 1000 個得られたのなら、 $\frac{\sum_{i=1}^{1000} \hat{\theta}_i}{1000}$ を計算することで θ の期待値がわかる。

$$\int_{-\infty}^{\infty} f(\theta|D) \cdot \theta \, d\theta \approx \frac{\sum_{i=1}^N \hat{\theta}_i}{N}$$

ただし、

$$\hat{\theta}_i \sim \text{事後分布}$$

では、ここからは乱数を生成する仕組みを考える。

2.2 マルコフ連鎖

先述したように、乱数はマルコフ連鎖を利用して生成するのだが、このマルコフ連鎖とは一体何なのか。正体は以下の等式を満たす $[X_t]$ のことである。

$$P(X_t | X_{t-1}, X_{t-2}, \dots, X_1) = P(X_t | X_{t-1})$$

- 左辺は、全期間にわたって結果がわかっているときの $[X_t]$ の確率分布
- 右辺は、1 時点前の結果のみわかっているときの $[X_t]$ の確率分布

左辺と右辺の条件付き確率が一致するのは、マルコフ連鎖は 1 時点前の値しか考慮しないからである。

1 時点前の値を所与とした条件付き確率（つまりは右辺）のことを遷移核と呼ぶ。

スマホを例に挙げよう。iPhone ユーザーが来期も iPhone を購入する確率、Android に買い換える確率、Android ユーザーが来期も Android を購入する確率、iPhone に買い換える確率は以下のように遷移核が表現できる。

$$\begin{aligned}P(X_t = \text{iPhone} | X_{t-1} = \text{iPhone}) &= 0.4 \\P(X_t = \text{Android} | X_{t-1} = \text{iPhone}) &= 0.6 \\P(X_t = \text{Android} | X_{t-1} = \text{Android}) &= 0.1 \\P(X_t = \text{iPhone} | X_{t-1} = \text{Android}) &= 0.9\end{aligned}$$

この遷移核に従いユーザー比率は変化し続ける、かと思えば途中から全体の比率は動かなくなる。このような、変化しなかった確率分布を定常分布という。

行列での理解を目指す。表にまとめるとこんな感じ

前\後	iPhone	Android
iPhone	0.4	0.6
Android	0.9	0.1

行列にすると

$$P = \begin{pmatrix} 0.4 & 0.6 \\ 0.9 & 0.1 \end{pmatrix}$$

この行列をひたすら掛けまくる。

```
t <- 0.1      #初期の iPhone のシェアを表す
a <- matrix(c(t, 1 - t, 1-t, t), nrow = 2)
a
```

```
      [,1] [,2]
[1,]  0.1  0.9
[2,]  0.9  0.1
```

```
p <- matrix(c(0.4, 0.9, 0.6, 0.1), nrow = 2)
p #遷移核
```

```
      [,1] [,2]
[1,]  0.4  0.6
[2,]  0.9  0.1
```

```
p1 <- a%*%p
p1 #t=1
```

```
      [,1] [,2]
[1,] 0.85 0.15
[2,] 0.45 0.55
```

```
p2 <- p1%*%p
p2 #t=2
```



```
      [,1] [,2]
[1,] 0.475 0.525
[2,] 0.675 0.325
```

```
p3 <- p2%*%p
p3 #t=3
```

```
      [,1] [,2]
[1,] 0.6625 0.3375
[2,] 0.5625 0.4375
```

```
p4 <- p3%*%p
p4 #t=4
```

```
      [,1] [,2]
[1,] 0.56875 0.43125
[2,] 0.61875 0.38125
```

```
p5 <- p4%*%p
p5 #t=5
```

```
      [,1] [,2]
[1,] 0.615625 0.384375
[2,] 0.590625 0.409375
```

```
p6 <- p5%*%p
p6 #t=6
```

```
      [,1] [,2]
[1,] 0.5921875 0.4078125
[2,] 0.6046875 0.3953125
```

```
p7 <- p6%*%p
p7 #t=7
```

```
      [,1] [,2]
[1,] 0.6039063 0.3960938
[2,] 0.5976563 0.4023438
```

初期の値, シェアがいくつであっても最終的に一定の比率に落ち着くことがわかる. 全体としては定常分布に落ち着くのだが, 一人一人個人の乗り換え履歴に関しては完全にランダムで決まる. この履歴を得ることは, マルコフ連鎖によって得られた乱数を獲得していることに等しい. さらに, この乱数は定常分布に従っていると考えられる.

ここまでの議論で, マルコフ連鎖を利用し遷移核の確率分布たる定常分布から乱数を得る方法を知った. が,

我々が欲しいのは定常分布に従う乱数ではなく、事後分布に従う乱数である。

整理しよう

遷移核を定める

⇓

遷移核の確率分布たる定常分布が得られる

⇓

定常分布に従う乱数を得ることができる

我々がやりたいことは、,,

乱数を得たい

⇓

乱数は事後分布に従って欲しい

⇓

事後分布は事後確率の確率分布である


⇓

遷移核をうまく設定すれば良さそう？

遷移核について、iPhone と Android を考えてきたが、パラメータ θ は連続であると想定される。

2.3 メトロポリス・ヘイスティングス法 (MH 法)

MH 法は乱数生成のアルゴリズムの一種である。ややこしいが、パラメータ θ は種類なのだが、「パラメータ θ の事後分布に従う乱数 $\hat{\theta}$ 」は複数生成される。そのため、これらを見分けるために添字を使う。乱数の初期値は $\hat{\theta}_1$ である。

 注意!!!

$\hat{\theta}_1$ や $\hat{\theta}_2$ には具体的な値が入る

ここからは、事後分布が $\text{Kernel}(\theta)$ に比例するということが重要になってくる。

$$f(\theta|D) \propto f(D|\theta) f(\theta) = \text{Kernel}(\theta)$$

1. $\hat{\theta}_1$ を適当に決める。本当になんでも良い
2. $N(0, \sigma^2)$ に従う乱数を、別に生成する

3. $\hat{\theta}_1 + \text{乱数}$ の値を便宜上 $\hat{\theta}_2^{\text{提案}}$ とする

4. $\frac{\text{Kernel}(\hat{\theta}_2^{\text{提案}})}{\text{Kernel}(\hat{\theta}_1)} = \text{rate}$ を計算する

1. ここで少し解説をする. 上記の計算式は $f(\hat{\theta}_2^{\text{提案}}|D)$ と $f(\hat{\theta}_1|D)$ という 2 つの事後確率分布の比を計算しているのである
2. 比をとることで共通の分母である正規化定数が消えるのである.

$$\begin{aligned}\frac{f(\hat{\theta}_2^{\text{提案}}|D)}{f(\hat{\theta}_1|D)} &= \frac{\frac{\text{Kernel}(\hat{\theta}_2^{\text{提案}})}{\text{正規化定数}}}{\frac{\text{Kernel}(\hat{\theta}_1)}{\text{正規化定数}}} \\ &= \frac{\text{Kernel}(\hat{\theta}_2^{\text{提案}})}{\text{Kernel}(\hat{\theta}_1)} \\ &= \text{rate}\end{aligned}$$

5. 1. もしも rate が 1 よりも高かったのであれば, それは提案されたパラメータの値 $\hat{\theta}_2^{\text{提案}}$ の方が密度が高かったということである. 密度が高いということは, その値は比較的発生しやすい値であることを意味する. (1 番発生しやすいであろう値 (θ の期待値) に一步近づいた). なので $\hat{\theta}_2^{\text{提案}}$ は晴れて $\hat{\theta}_2$ になるのである. 提案の文字が消えて, 正式に乱数と認められた
2. もしも rate が 1 よりも小さかったのであれば,
- 確率 rate で $\hat{\theta}_2^{\text{提案}}$ が $\hat{\theta}_2$ になる
 - 確率 (1-rate) で $\hat{\theta}_1$ が $\hat{\theta}_2$ を兼任する (これらの措置をとることで, ある程度起こりやすそうな値が乱数として採用されやすいようになる)

⚠ 再び注意!!!

$\hat{\theta}_1$ や $\hat{\theta}_2$ には**具体的な値**が入る. 混乱しないように

6. $\hat{\theta}_2$ を初期値として 2. からの計算をもう一度行う

7. これらを何度も行い, 多数の乱数を得る.

$\hat{\theta}_t$ は一時点前の $\hat{\theta}_{t-1}$ に基づいて得られている. つまりはマルコフ連鎖の形になっている.

せっかくなので, 乱数生成を繰り返した結果を見てみよう (stan で)

#こんなデータが得られたとする

```
y <- rnorm(n = 2024, mean = 2.71, sd = 3.14)
```

#データの生成も乱数にした

```
myd <- y
```

#2024/7/20 日変更

```
sample_size <- length(y)
```

```
data_list <- list(y = y, N = sample_size)
```

```
# mcmc の実行

#生成回数が 50
mcmc_result_50 <- stan(
  file = "stan/1.stan",
  data = data_list,
  seed = 1,
  iter = 50,
  chains = 1
)

#生成回数が 2000
mcmc_result_2000 <- stan(
  file = "stan/1.stan",
  data = data_list,
  seed = 1,
  iter = 2000,
  chains = 1
)
```

! 2024_7_20 上記を修正

- データ生成もシミュレーションの一部にして、ベイズ推定で正しいパラメタが推定できるかどうか確かめるとより良いですね。

```
trace1 <- traceplot(mcmc_result_50
  ,pars = c("mu"),
  inc_warmup = T)
trace2 <- traceplot(mcmc_result_2000,
  pars = c("mu"),
  inc_warmup = T)
plot(trace1/trace2)
```

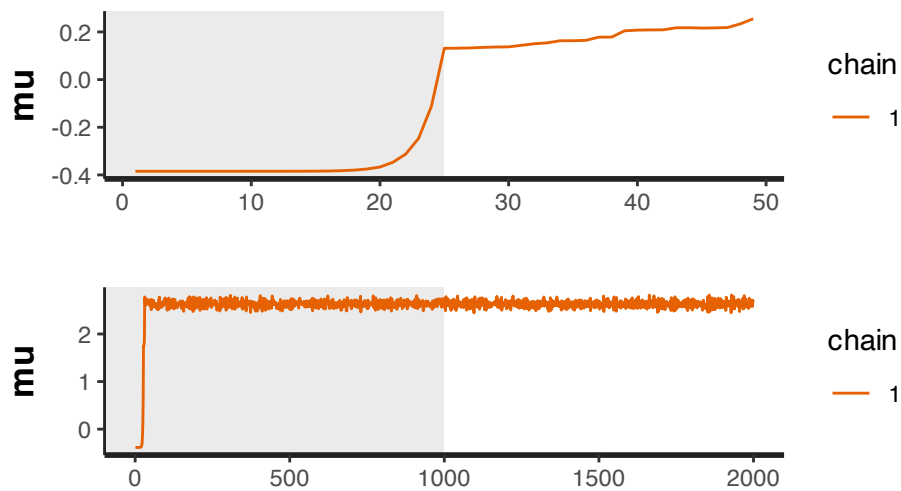


図 1: MCMC による乱数生成

定常分布に収束したとみて良いだろう。

3 stan

なにごとにも実際にやってみないと実感できない。stan で乱数を生成し、その分布を見てみよう。

3.0.1 MCMC の実行

正規分布を用いた確率モデルを想定する。

$$X \sim \text{Normal}(\theta, 1)$$

θ を求めることを目標にする。

まずは stan に統計モデルの情報を教えてあげる。(stan ファイルに記述する)

記述ができたならモデルをコンパイルする

```
mu_norm <- cmdstan_model("stan/1.stan")
```

stan ファイルの中身を見てみる

```
mu_norm$print()
```

```
data {
  int<lower=0> N;
  vector[N] y;
}
```

```
parameters {
```

```

    real mu;
    real sigma;
  }

  model {

    //統計モデルの指定
    y ~ normal(mu, sigma);
  }

```

以下のようなデータが得られたとする.

```
y <- rnorm(n = 2024, mean = 2.71, sd = 3.14)
```

このデータを stan に渡すために list 形式にする. ついでに事前分布の指定もする

```
list <- list(N = length(y),
             y = y)
```

これらのデータを stan に渡して推定を行う.

```
fit <- mu_norm$sample(
  data = list,
  seed = 1,
  chains = 4,
  refresh = 1000,
  iter_warmup = 1000,
  iter_sampling = 3000
)
```

結果を表示する

```
#mu
fit$summary("mu")
```

```
# A tibble: 1 x 10
  variable mean median    sd    mad    q5   q95  rhat ess_bulk ess_tail
  <chr>    <dbl>  <dbl>  <dbl>  <dbl>  <dbl> <dbl> <dbl>    <dbl>    <dbl>
1 mu      2.70   2.70 0.0714 0.0707  2.58  2.82  1.00   8681.   7931.
```

```
#sigma
fit$summary("sigma")
```

```
# A tibble: 1 x 10
  variable mean median    sd    mad    q5   q95  rhat ess_bulk ess_tail
```

```

      <chr>      <dbl>  <dbl>  <dbl>  <dbl> <dbl> <dbl> <dbl>      <dbl>      <dbl>
1 sigma      3.23    3.23 0.0503 0.0501  3.15  3.31  1.00    9547.    7848.

```

3.0.2 可視化

MCMC の役割は事後分布に従う乱数を生成することであった。まずはその乱数を取り出す

```

post <- fit$draws() |>
  as_draws_df()
glimpse(post)

```

Rows: 12,000

Columns: 6

```

$ lp__      <dbl> -3382.97, -3382.99, -3383.02, -3383.12, -3384.38, -3382.79, ~
$ mu        <dbl> 2.66220, 2.71881, 2.72421, 2.63384, 2.72640, 2.67148, 2.699~
$ sigma     <dbl> 3.25580, 3.18815, 3.26584, 3.21445, 3.13633, 3.21089, 3.274~
$ .chain     <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
$ .iteration <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, ~
$ .draw      <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, ~

```

こうすることで事後分布から取り出した乱数をデータフレームに格納することができた。

分布が本当に収束しているのか視覚的に理解するために、トレースプロットを作る

```

#mu
trace1 <- ggplot(post,
  aes(x = .iteration,
      y = mu,
      group = as.factor(.chain))) +
  geom_line(aes(colour = as.factor(.chain)))+
  ylab("mu") +
  xlab("実行回数")

#sigma
trace2 <- ggplot(post,
  aes(x = .iteration,
      y = sigma,
      group = as.factor(.chain))) +
  geom_line(aes(colour = as.factor(.chain)))+
  ylab("sigam") +
  xlab("実行回数")

plot(trace1/trace2)

```

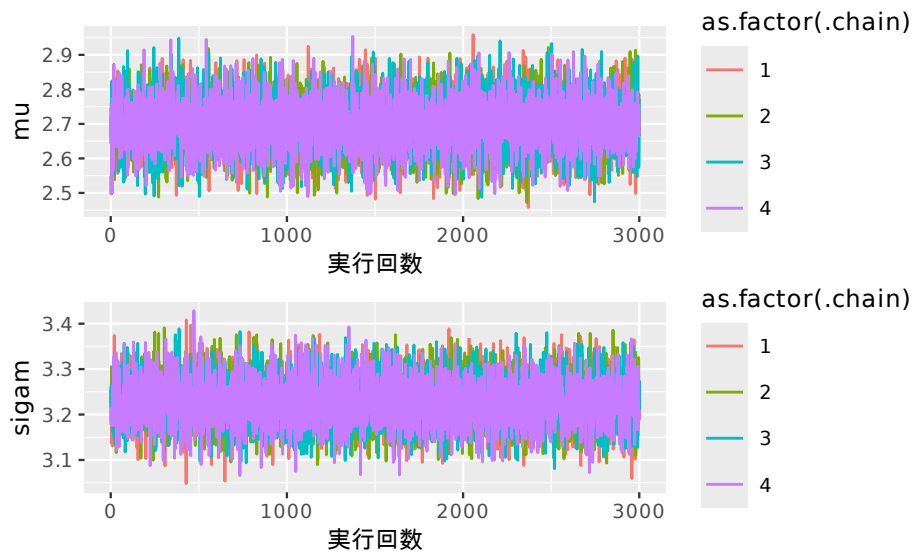


図 2: トレースプロットの図示

毛虫になったので問題なし.

θ の事後分布を確認しよう

```
#mu
plt1 <- ggplot(post,
  aes(x = mu,
      y = after_stat(density))) +
  geom_histogram(colour = "black")+
  labs(x = expression(mu),
      y = "事後確率密度")+
  geom_vline(xintercept = 2.71,
      color = "red")

#sigma
plt2 <- ggplot(post,
  aes(x = sigma,
      y = after_stat(density))) +
  geom_histogram(colour = "black")+
  labs(x = expression(sigma),
      y = "事後確率密度")+
  geom_vline(xintercept = 3.14,
      color = "red")
```



```
plot(plt1/plt2)
```

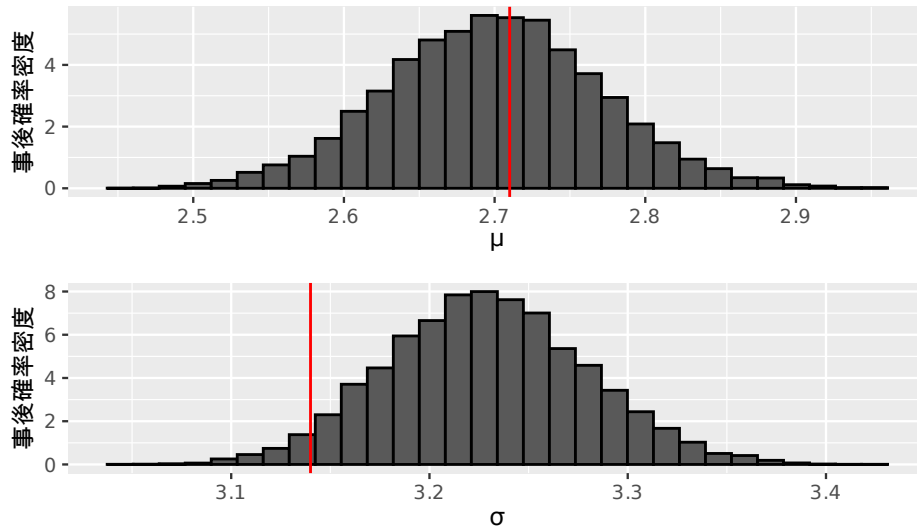


図 3: 事後分布の確認

ちょっとズレる.

このサンプル (乱数) を使うことで平均や信用区間を求めることができる.

```
#mu  
#平均  
mean(post$mu)
```

```
[1] 2.699359
```

```
#mu  
#95% 信用区間  
quantile(post$mu,probs = c(0.025,0.975))
```

```
      2.5%      97.5%  
2.557927 2.842821
```

```
#sigam  
#平均  
mean(post$sigma)
```

```
[1] 3.228867
```

```
#sigma  
#95% 信用区間  
quantile(post$sigma,probs = c(0.025,0.975))
```

2.5% 97.5%
3.133369 3.330211

分布は少しずれていたが、平均と信用区間をみるとかなり近い値が出ている。

4 補足 信頼区間と予測区間の違い

4.0.1 ベイズ信頼区間 [75,119,129]

事後分布に従う乱数を小さいものから順番に並べて、2.5% 点から 97.5% 点に該当する範囲を並べることで求める。

4.0.2 予測区間 [137,178]

データ D が与えられた時の、将来的な観測値 pred^{*2} が従う確率分布の確率質量関数を $f(\text{pred}|D)$ とする。これが予測分布である。例えば、パラメータ θ が与えられた時の観測値 pred の確率分布の確率質量関数は $f(\text{pred}|\theta)$ となる。この確率質量 (密度) 関数はポアソン分布かもしれないし、正規分布かもしれない。 θ の事後分布の確率密度関数を $f(\theta|D)$ とおくと、予測分布は以下のように求められる

$$f(\text{pred}|D) = \int f(\text{pred}|\theta)f(\theta|D)d\theta$$
$$\text{予測分布} = \int (\theta \text{が与えられてたときに} \text{pred} \text{が従う分布})(\theta \text{の事後分布})d\theta$$

この計算はめんどくさいので、mcmc を活用する。まずは θ の MCMC サンプルを得る。そして、 θ の MCMC サンプルを母数とした確率分布 (ポアソン, 正規, etc,,,) に従う乱数を得る。この乱数こそが観測値 pred の MCMC サンプルになる。

*2 pred は prediction(予測) の略