

# マルチカノニカルMCMCによる分割表の数の推定

伊庭 幸人

モデリング研究系 教授

(データ同化研究開発センター, 統計思考院 兼任)

マルコフ連鎖モンテカルロ法的一种であるマルチカノニカルMCMCは多峰性の分布からのサンプリングのための道具として有用であり, 特にレアイベントの確率計算や離散構造の数の推定に向いている.

Kitajima and Kikuchi(2015)はPinn and Wiczerkowski(1998)の手法を拡張して30x30までの魔方陣の数を推定した. ここではこの方法を一般の分割表のサンプリングに適用する. 「制約をゆるめる」方法により複雑なMarkov baseの考察を省略できるのが利点である.

**マルチカノニカルMCMC**  
(Berg 1991,1992)

状態:  $x$ , 注目する統計量:  $U(x)$

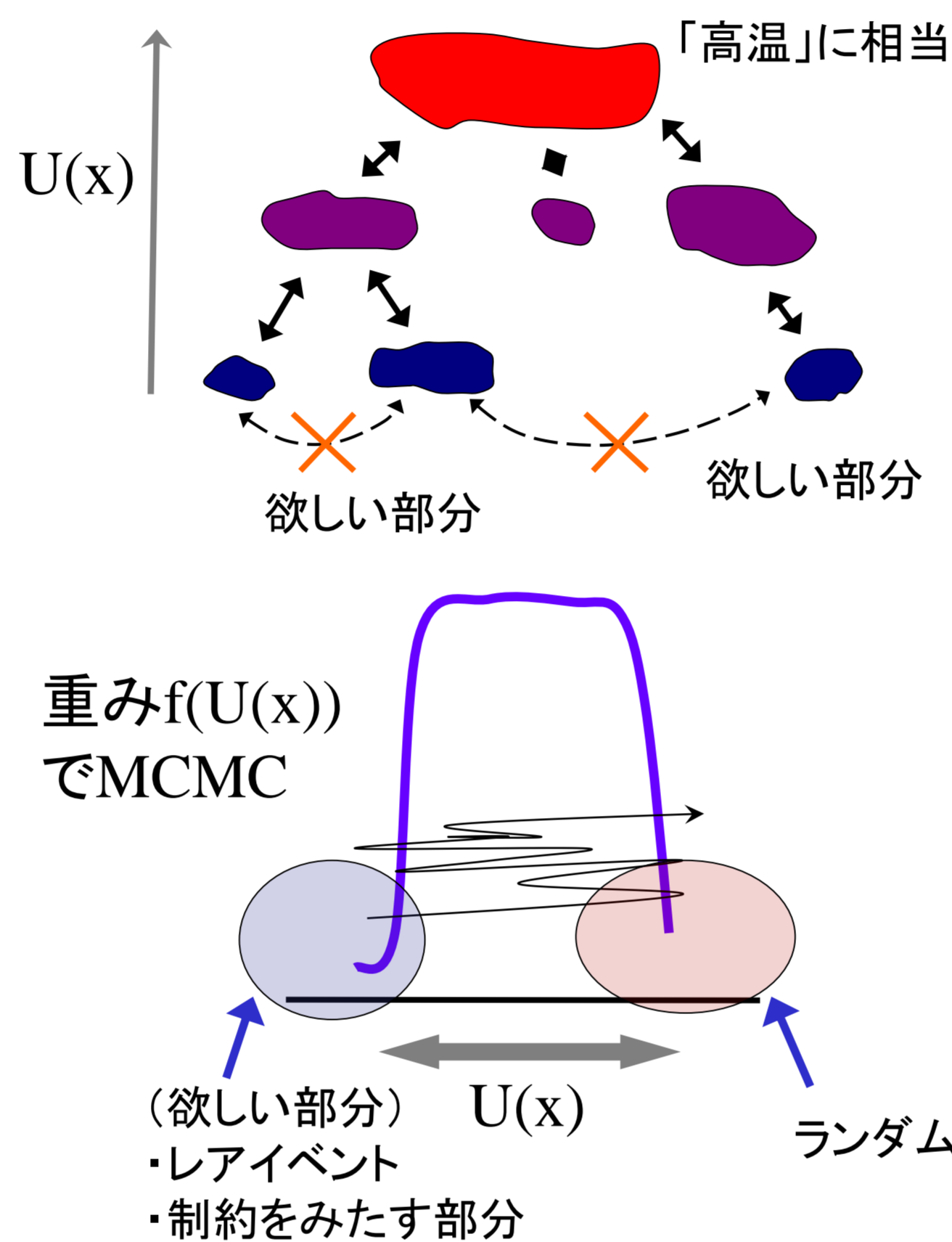
$f(U)$ の関数形をうまく選ぶ  
→  $f(U(x))$ を重みとしてMCMC

↓

$U$ の値の広い区間をほぼ一様に生成するような $x$ のサンプリングを実現

※  $x$ をランダムに生成してもそうならないことに注意(大数の法則!)

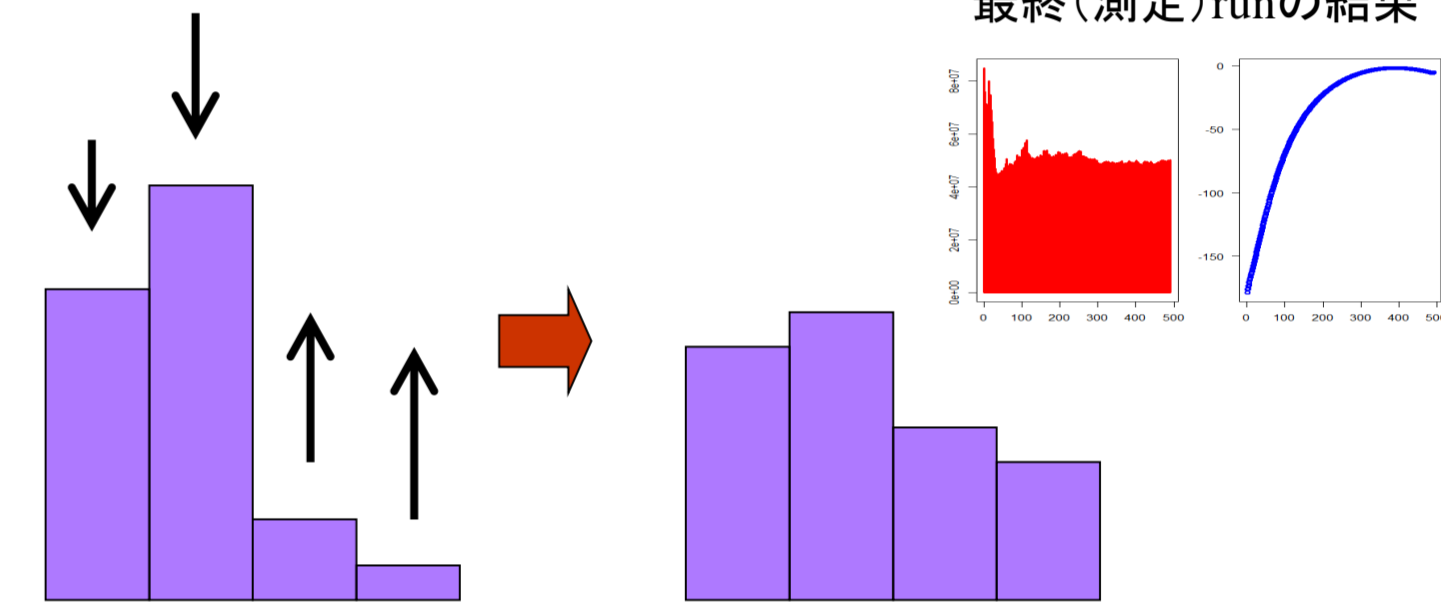
- レアイベントを高確率で生成(右下)
- 局所的な解にトラップされにくい (アニーリング効果: 右上)
- 正しい確率でサンプリングできる
- レアイベントの確率(個数)の推定



[適切な $f(U)$ をどうやって求めるか]

→ 実際にやってみて学習する

最も簡単な学習法(エントロピック・サンプリング)  
シミュレーション→ヒストグラムを繰り返す  
沢山出たら重みを下げる



高次元の $x$ でなく1次元の $U(x)$ のヒストグラム  
[  $f(U(x))$ は高次元の $x$ の関数;  $f(U)$ は1変数 ]

より効率的な手法として  
ワン・ランダウ法がよく使われる

**魔方陣(対角線2本の制約が特徴)**

$U(x) = \sum | \text{行の和} - C | + \sum | \text{列の和} - C | + \sum | \text{対角線の和} - C |$   
( $x$ : 各数字1個使用の範囲)  $| \cdot |$ は絶対値

$U(x)=0 \Leftrightarrow$  配列 $x$ が魔方陣

魔方陣の制約を完全に満たす

Kitajima and Kikuchi(2015)  
30x30 ~  $6.56 \times 10^{2056}$  個

**「ダーウィンフィンチの表」(12x17)**

縦横の周辺和が左と等しい01表の個数を勘定

|| = 14  
= ..

|| ||  
3 :

$U(x)$ : 魔方陣と同様  
・対角線の和がない  
・ $C$ が行・列に依存  
( $x$ : この場合は要素が0か1)

$6.71 \times 10^{16}$  (厳密)  
 $6.76 \times 10^{16}$  (3min /laptop)  
 $6.71 \times 10^{16}$  (6min+/laptop)  
i7-5600U 2.6GHz

SIS法で10秒 (2005, Chen et al, JASA)  
Markov baseが簡単(MCMC)・比較的やさしい問題

**Latin cube (4x4x4, 5x5x5)**

ラテン立方体の拡張  
縦・横・奥行きに色が各1  
⇔ 4x4x4x4, 5x5x5x5の01分割表

[4x4x4]  $64 \times 4! \times 3! \times 3! = 55296$   
54646 (19min+/laptop)  
55456 (33min /laptop)

[5x5x5]  $40246 \times 5! \times 4! \times 4! = 2.78 \times 10^9$   
 $2.76 \times 10^9$  348 min (6時間弱)  
i7-5930K 3.5GHz

総研大大学院生募集 他大学の院生も在籍のまま受け入れ可能 (1~2年: 特別共同利用研究員制度)  
— 具体的な問題を扱いながら, 統計モデリングやモンテカルロ法の使い方が学べます —

医学データの階層ベイズ/ベイズ生存時間分析  
外部の保健・医学研究者と組んで具体的なデータを解析

レアイベントサンプリングで解ける問題各種

組み合わせの数の計算や時間逆転シミュレーション,  
リスク解析などさまざまな手法や応用が考えられます

政党支持率データの状態空間モデル解析  
メディアによる違いや時間変動を統一的に扱う手法を開発