Pythonで学ぶ
ベイズ推論ハンズオン（後編）
【ベイズ統計モデリングに慣れる】

https://github.com/Ronmemo/Beyes

本日の予定

7/28（日）後編【ベイズ統計モデリングに慣れる】
ベイズ推論の基礎続き
・事前分布と事後分布の関係
・MCMC法とは何か，その他の近似法
・予測分布の求め方
・Python（PyMC3）によるベイズ推論の実装，より詳しく中身をみる

さまざまなモデルにおけるベイズ推論
・正規分布モデル
・線形回帰モデル
・ロジスティック回帰モデル
・階層モデル
・混合分布モデル

ベイズ推論の基礎
（続き）

ベイズ推論の流れ（再掲）

学習

1. データの特徴をつかむ

2. モデルを決める

3. 事後分布を求める

ベイズの定理

$$p(\theta \mid D) = \frac{p(D \mid \theta)p(\theta)}{p(D)}$$

推論

4. 予測分布を求める

$$p(x_* \mid D) = \int p(x_* \mid \theta)p(\theta \mid D)d\theta$$

1. データの特徴をつかむ

例：コインを投げて，5回中2回表が出た
→表が出る確率は？

表1，裏0とすると
0,1,0,0,1
と表せる

2. モデルを決める

ベルヌーイ分布（二項分布）

でモデリングする

尤度

$$L(\theta|X) = p(X|\theta) = Bern(X|\theta)$$

→データを示す

事前分布

$$p(\theta) = Beta(\theta|a, b)$$

→既知の情報を示す

ベータ分布は
二項分布に対して
共役である

＊ X：訓練データ集合 θ ：パラメータ

事前分布としてのベータ分布

https://ryosuke-okubo.hatenablog.com/entry/2019/03/07/210000

3. 事後分布を求める

条件付き確率

$$p(X, p) = p(X\,|\,\theta)p(\theta)$$

ベイズの定理

事前分布

$$p(\theta\,|\,X) = \frac{p(X\,|\,\theta)p(\theta)}{p(X)}$$

周辺分布

共役事前分布の利点

X1を観測した後の事後分布

$$p(\theta | X_1) \propto p(X_1 | \theta) p(\theta)$$

その後，X2を観測した後の事後分布

$$p(\theta | X_1, X_2) \propto p(X_2 | \theta) p(\theta | X_1)$$

共役事前分布を用いると，
全て同じ形の確率分布が
出力される

解析解を出すのが楽

例：ベルヌーイ分布

https://ryosuke-okubo.hatenablog.com/entry/2019/03/11/210000_1

コイン投げ問題の例

[1]　一般モデルの推定

まず偏りの概念を一般化する。

- 偏りが0→裏しか出ない
- 偏りが1→表しか出ない
- 偏りが0.5→表と裏が半々出る

偏りのパラメータをθ，N回投げて表が出た回数をyとすると，次の式で表される。

$$p(\theta|y) \propto p(y|\theta)p(\theta)$$

[2]　尤度の選択

コイン投げの試行において，尤度は**2項分布**となる。

$$p(y|\theta) = \binom{N}{y}\theta^{y}(1-\theta)^{N-y}$$

[3]　事前分布の選択

ベータ分布を用いる。

$$p(\theta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\theta^{\alpha}(1-\theta)^{\beta-1}$$

ベータ分布を使う理由として，

- 0と1の区間に制限されている
- パラメータによって様々な形をとる
- 2項分布の共役事前分布である

ことがあげられる。

[4]　事後分布の計算

ベイズの定理より，

$$p(\theta|y) \propto \binom{N}{y}\theta^{y}(1-\theta)^{N-y}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\theta^{\alpha}(1-\theta)^{\beta-1}$$

θに依存する項のみで簡略化すると，

$$p(\theta|y) \propto \theta^{\alpha-1+y}(1-\theta)^{\beta-1+N-y}$$

つまり，次のベータ分布で表される。

$$p(\theta|y) = Beta(\alpha_{prior}+y, \beta_{prior}+N-y)$$

実際，コンピューターでは

近似解を求める方法

- **サンプリング（MCMC）**

  メトロポリス-ヘイスティング

  ノーUターンサンプラー

  ギブスサンプリング

  ハミルトニアンモンテカルロ

  逐次モンテカルロ

- **簡単な式に変換する周辺化計算の近似**

  変分推論

  ラプラス近似

  期待値伝播

MCMC法とは？

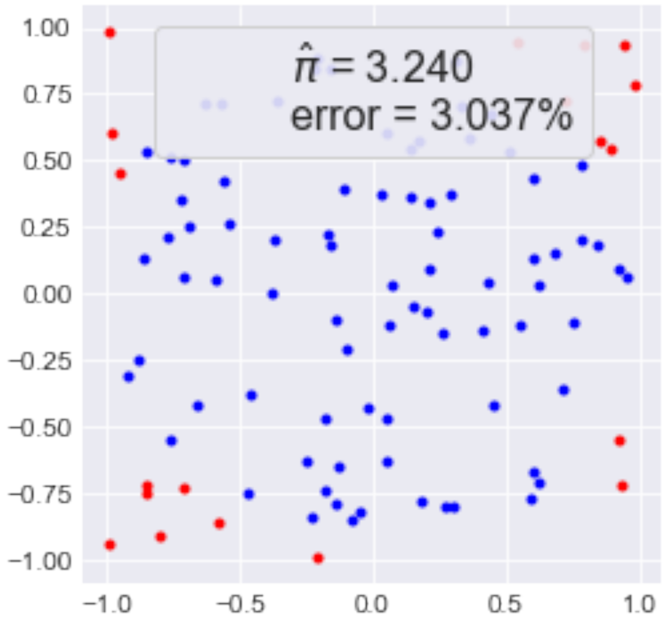
Monte Carlo + Markov chain

モンテカルロ法
→乱数を用いた近似方法

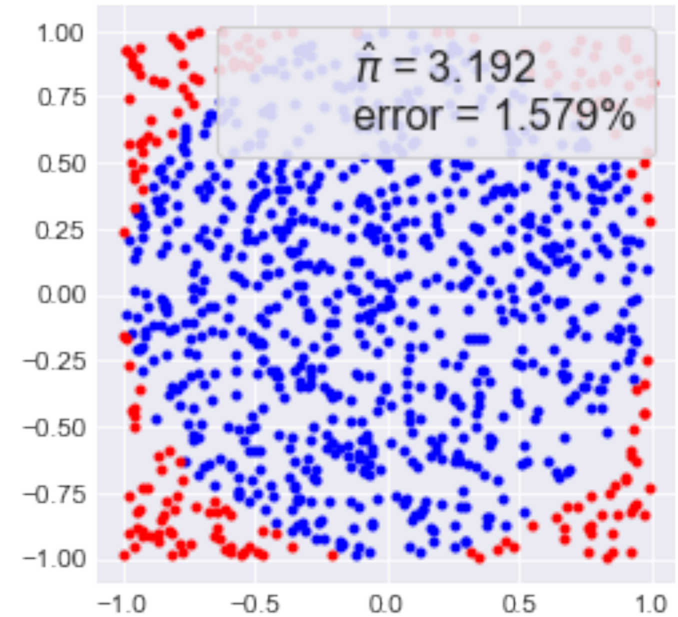
マルコフ連鎖
→現在の状態が決まっていれば、
過去および未来の状態は独立である
確率過程（離散）

$$P(x_{t+1} \mid x_1, x_2, \ldots, x_t) = P(x_{t+1} \mid x_t)$$

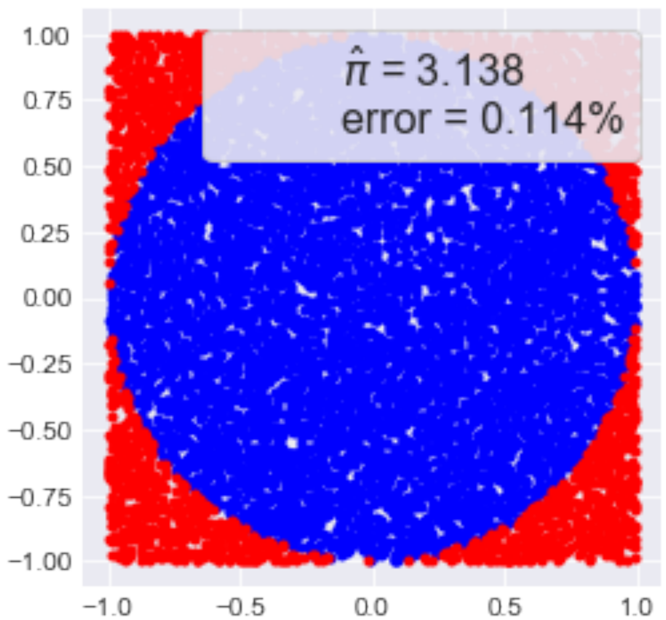
モンテカルロ法の例



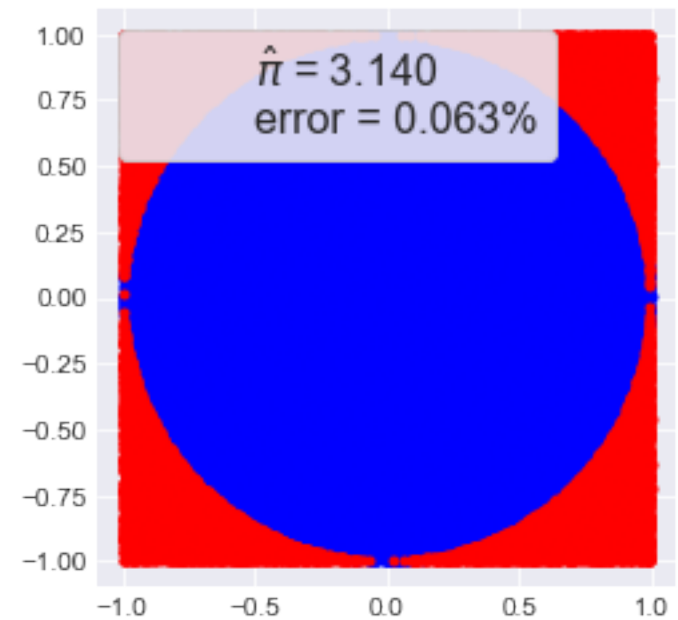
N = 100



N = 1000



N = 10000



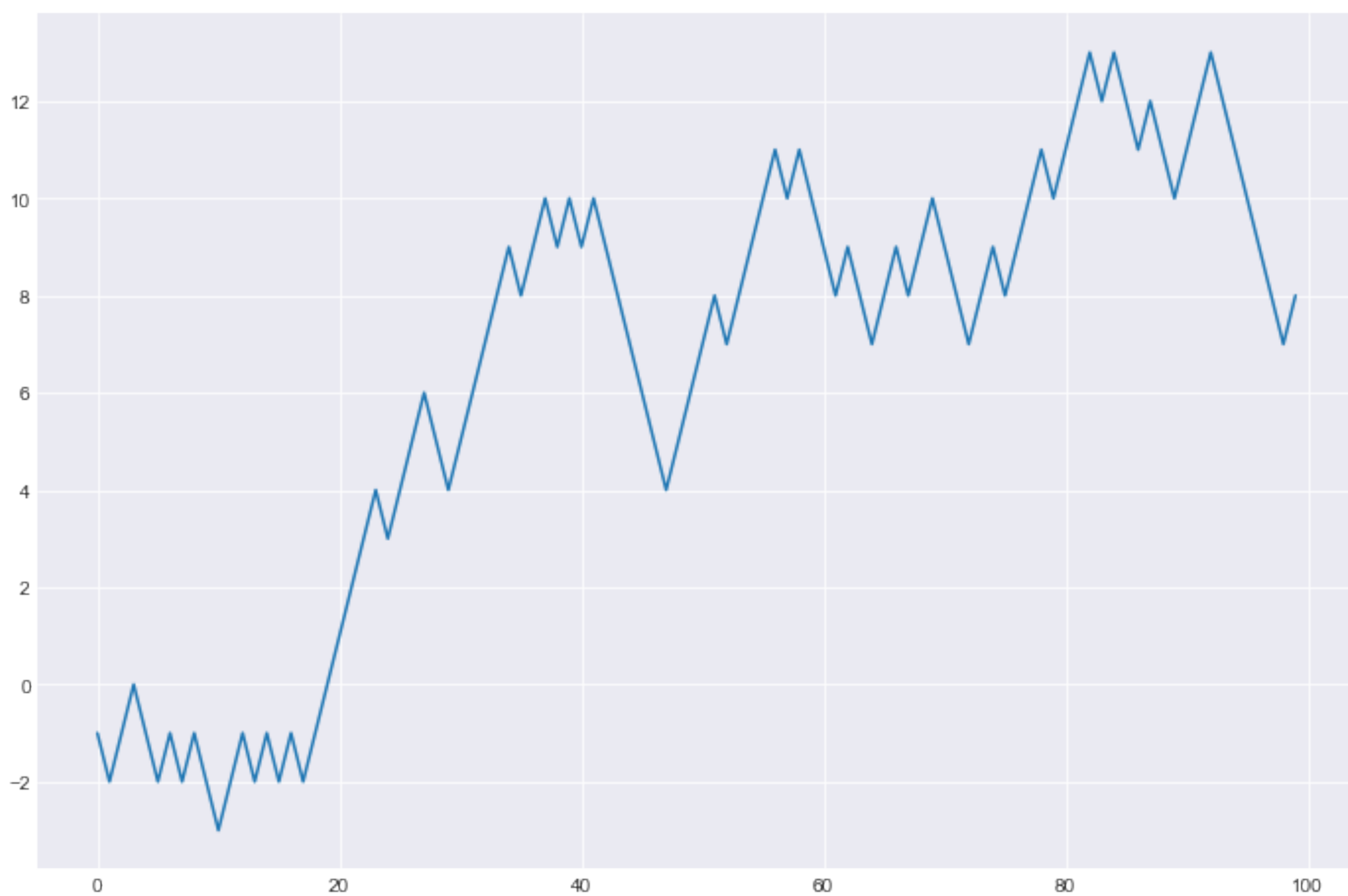
N = 100000

マルコフ連鎖の例

ランダムウォーク

$$P(x_{t+1} \mid x_1, x_2, \ldots, x_t)$$
$$= P(x_{t+1} \mid x_t)$$

$\rightarrow$**1 or -1**



メトロポリス-ヘイスティング

ベイジアン分析は，主に**マルコフ連鎖モンテカルロ**（MCMC）法によって実行される。マルコフ連鎖において，メトロポリス-ヘイスティングス・アルゴリズムが汎用される。アルゴリズムの手順を以下に示す。

1. パラメータx_iの初期値を決める
2. 新しいパラメータの値x_{i+1}を決め，サンプリングが簡単な分布（例：正規分布）$Q(x_{i+1}|x_i)$からサンプルを得る。
3. メトロポリス-ヘイスティングスの基準

   $$p_a(x_{i+1}|x_i) = min\left(1, \frac{p(x_{i+1})q(x_i|x_{i+1})}{p(x_i)q(x_{i+1}|x_i)}\right)$$

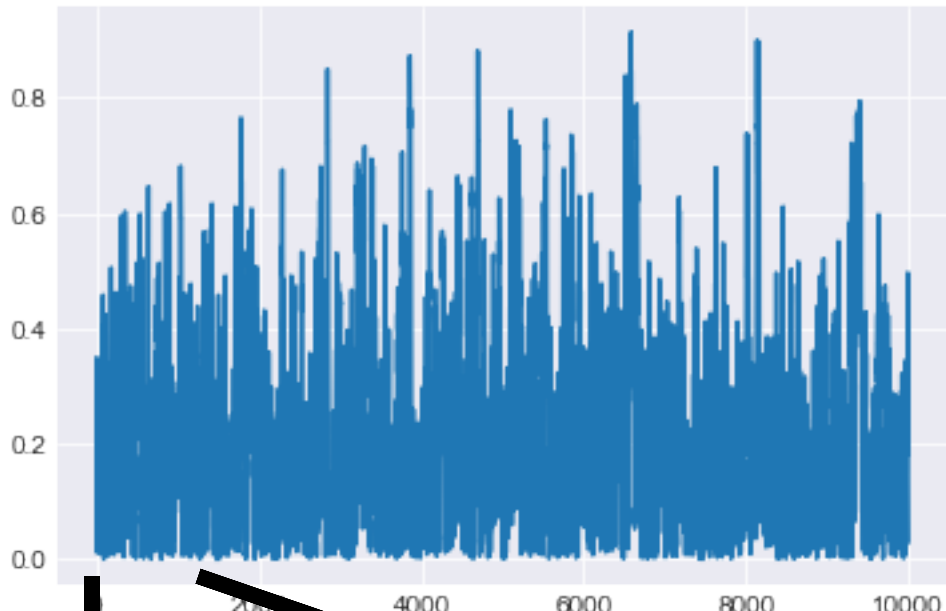
   を使って，新しいパラメータの値を受け入れる確率を計算する。

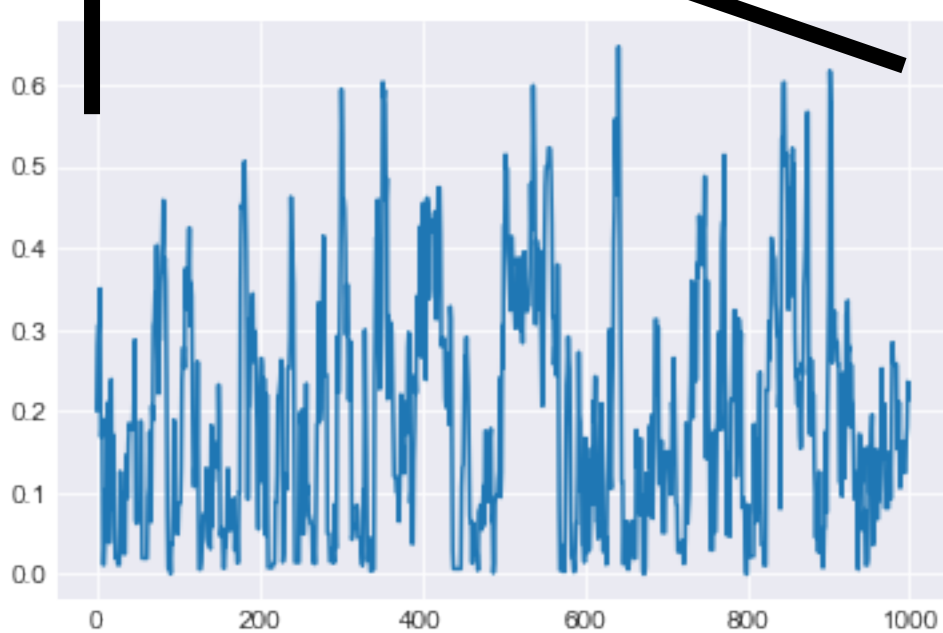
4. 3.で計算された確率が区間[0,1]の一様分布から得られる値より大きい場合，新しい状態を受容して，そうでなければ古い状態に留まる。
5. 1.から4.を繰り返す。

最終的に，**サンプルチェーン**（**サンプルトレース**）という数値リストを得る。これは事後分布の近似である。

サンプリングの様子

$\beta(1,4)$

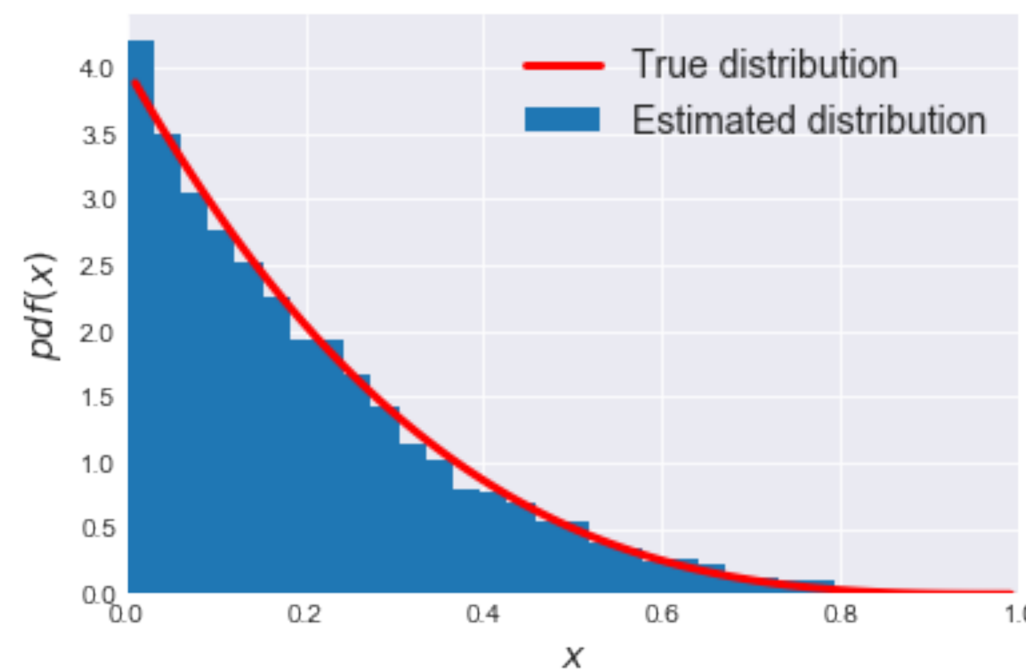




最初の1000ステップ

より高い確率の方に
移動しようとする

集計すると,
確率分布の近似が
得られる



4. 予測分布を求める

$$p(x_* \mid D) = \int p(x_* \mid \theta) p(\theta \mid D) d\theta$$

未知の
データ

→さまざまなθについて
モデルの平均

Python（PyMC3）による
ベイズ推論の実装,
より詳しく中身をみる

さまざまなモデルにおけるベイズ推論