β分布の理解からはじめるベイズ理論

—不確実性の表現・更新・予測・意思決定まで

河合勝彦
名古屋市立大学大学院経済学研究科
kkawai@econ.nagoya-cu.ac.jp

2026年5月21日

概要

本稿は、β分布を入口としてベイズ理論の基本的な考え方を学ぶための入門的チュートリアルである。確率を不確実な量として扱う視点から出発し、β分布の形状、平均・分散、ベルヌーイ試行との共役性、観測データによる事後分布の更新、予測分布、A/Bテスト、意思決定への応用までを順に説明する。数式による定義と直感的な解釈を対応づけながら、Pythonによる可視化例を通じて、ベイズ更新がどのように不確実性を減らし判断を支えるかを確認する。

目次

1 **はじめに読む用語集** 4

2 **記号と式の読み方** 4

3 **はじめに：なぜβ分布からベイズ理論を学ぶのか** 5

4 **β分布の直感：「確率の確率」とは** 6
- 4.1 コインの例で考える . . . 6
- 4.2 離散から連続へ . . . 6
- 4.3 β分布の使われ方の典型像 . . . 7

5 **β分布の定義** 7
- 5.1 確率密度関数（PDF） . . . 7
- 5.2 ベータ関数とガンマ関数 . . . 7

6 **パラメータ α, β の直感的解釈** 8
- 6.1 擬似カウントとしての解釈 . . . 8
- 6.2 形状の代表例 . . . 8
- 6.3 代表的な密度の形 . . . 8

7 **基本的な性質** 9

7.1 期待値・分散・最頻値 9
7.2 形状についての要点 9
7.3 他の分布との関係 9

8 ベイズ理論の基本地図 10
8.1 ベイズ理論が扱うもの 10
8.2 ベイズの定理 10
8.3 β分布が全体像を見せてくれる理由 10

9 ベイズ更新：二項分布の共役事前分布として 11
9.1 設定 11
9.2 事後分布の導出 11
9.3 擬似カウントの再解釈 11
9.4 逐次更新 12
9.5 次の観測を予測する 12

10 推定・信用区間・予測分布 12
10.1 事後分布をどう要約するか 12
10.2 信用区間 12
10.3 複数回先の予測：β-二項分布 13
10.4 逐次学習としての見方 13

11 事前分布の選び方 13

12 ベイズ推論を意思決定につなげる 14
12.1 推定だけでは意思決定にならない 14
12.2 損失関数と代表的な点推定 14
12.3 A/B テストの意思決定 14

13 モデルチェックと感度分析 14
13.1 事前分布への感度 14
13.2 事後予測チェック 15
13.3 独立性と過分散 15

14 ベイズ理論の応用例 15
14.1 A/B テストでのクリック率推定 15
14.2 Thompson サンプリング（多腕バンディット） 16
14.3 プロジェクト管理：PERT 16
14.4 品質管理・不良率推定 16

15 共役性を越えたベイズ理論 16
15.1 共役モデルの位置づけ 17
15.2 階層ベイズ：複数の比率を同時に学ぶ 17

15.3 MCMC と近似推論 17

16 Python による可視化と実装 18
16.1 密度関数の描画 18
16.2 ベイズ更新の可視化 18
16.3 A/B テストの事後比較 19

17 R による可視化と実装 19
17.1 密度関数の描画 19
17.2 ベイズ更新の可視化 20
17.3 A/B テストの事後比較 20

18 スクリプト実行結果と分析 21
18.1 図による確認 21
18.2 ベイズ更新の数値結果 24
18.3 A/B テストの数値結果 24

19 よくある誤解と注意点 25

20 まとめ 25

参考文献・さらに学ぶには 26

1　はじめに読む用語集

この文書では統計・確率の用語を多く使います。最初からすべてを覚える必要はありません。本文中で迷ったら，この節に戻って確認してください。

確率変数　結果によって値が変わる量。コイン投げなら「表なら 1，裏なら 0」とする量など。

確率分布　確率変数がどの値をどれくらい取りやすいかを表すルール。

離散分布　値が 0, 1, 2 のように飛び飛びになる分布。ベルヌーイ分布や二項分布が例。

連続分布　値が実数の範囲を連続的に動く分布。β分布や正規分布が例。

確率密度　連続分布で「その近くに値が出やすいか」を表す高さ。密度そのものは確率ではなく，区間で積分すると確率になる。

パラメータ　分布の形を決める数。β分布では α, β がパラメータ。

期待値　分布の平均的な位置。何度も同じ状況を繰り返したときの長期的な平均のイメージ。

分散　値のばらつきの大きさ。大きいほど不確実性が広い。

最頻値　分布の山が最も高い位置。もっとも「ありそう」に見える値。

ベルヌーイ試行　成功／失敗，表／裏，クリック／非クリックのように結果が 2 通りの 1 回の試行。

二項分布　同じ成功確率でベルヌーイ試行を n 回行ったとき，成功回数が従う分布。

事前分布　データを見る前に，未知量について持っている不確実性を表す分布。

尤度　未知量がある値だったと仮定したとき，観測データがどれくらい自然かを表す量。

事後分布　データを見た後に更新された，未知量についての不確実性を表す分布。

予測分布　未知量の不確実性も含めて，次に観測されるデータを予測する分布。

信用区間　ベイズ推論で，未知量が指定した事後確率で入る範囲。たとえば 95% 信用区間。

共役事前分布　事前分布と事後分布が同じ種類の分布になる便利な組み合わせ。

感度分析　事前分布などの仮定を変えても結論が大きく変わらないか確認する作業。

MCMC　手計算できない事後分布からサンプルを作る数値計算法の総称。

2　記号と式の読み方

統計やベイズ推論の式では，同じ文字が文脈によって少し違う役割を持つことがあります。ここでは本文で使う記号を先にまとめます。特に，未知の成功確率を表す p と，確率密度を表す関数記号 $p(\theta)$ は別の使い方なので，文脈で読み分けます。

記号	意味
p	成功確率，クリック率，不良率など，0 から 1 の値をとる未知量。
θ	一般的な未知パラメータを表す記号。ベイズの定理を抽象的に書くときに使う。
y	観測されたデータ。成功／失敗の列や，成功回数など。
n	試行回数または標本サイズ。
k	成功回数。二項モデルでは $0 \le k \le n$。
α, β	β 分布の形を決めるパラメータ。擬似的な成功数・失敗数としても読める。
$\mathrm{Beta}(\alpha, \beta)$	パラメータ α, β の β 分布。
$B(\alpha, \beta)$	ベータ関数。β 分布の密度を全体で 1 にするための正規化定数。
$\Gamma(x)$	ガンマ関数。階乗を実数へ拡張した関数で，ベータ関数の定義に出てくる。
$f(p)$	p における確率密度の高さ。高さそのものは確率ではない。
$\Pr(A)$	事象 A が起こる確率。たとえば $\Pr(p_B > p_A)$ は B が A より良い確率。
$p(\theta)$	未知量 θ の事前分布。ここでの $p(\cdot)$ は密度または確率質量を表す関数。
$p(y \mid \theta)$	尤度。θ が与えられたとき，データ y がどれくらい自然かを表す。
$p(\theta \mid y)$	事後分布。データ y を見た後の θ への不確実性。
$\binom{n}{k}$	n 回のうち k 回成功する並び方の数。二項分布に出てくる。
$E[p]$	p の期待値。事後分布の平均として点推定に使える。
$\mathrm{Var}(p)$	p の分散。事後分布のばらつきの大きさ。
$\mid$	「条件付き」を表す。$p(\theta \mid y)$ は「y が与えられたもとでの θ」。
$\propto$	比例するという意味。正規化定数を省いて分布の形だけを書くときに使う。
$\int$	積分を表す記号。連続分布で密度を足し合わせて確率を出す。
$\sim$	「従う」と読む。$p \sim \mathrm{Beta}(\alpha, \beta)$ は p が β 分布に従うという意味。
$\approx$	厳密な等号ではなく，近似的に等しいという意味。

式を読むときは，まず「どの量が未知で，どの量が観測済みか」を確認します。β 分布の例では，クリック率や成功確率 p が未知量で，試行回数 n と成功回数 k がデータです。事前分布と尤度を掛け合わせると，データを見た後の事後分布が得られます。

式	読み方
$p \sim \mathrm{Beta}(\alpha, \beta)$	未知の成功確率 p は，パラメータ α, β の β 分布に従う。
$X \mid p \sim \mathrm{Binomial}(n, p)$	成功確率 p が与えられたとき，成功回数 X は二項分布に従う。
$p(\theta \mid y) \propto p(y \mid \theta)p(\theta)$	事後分布は，尤度と事前分布を掛けたものに比例する。
$p \mid y \sim \mathrm{Beta}(\alpha + k, \beta + n - k)$	n 回中 k 回成功を観測した後，p の分布は更新された β 分布になる。
$E[p] = \alpha/(\alpha + \beta)$	β 分布の平均は，α と β の合計に占める α の割合である。
$\Pr(p_B > p_A)$	B 案の成功確率が A 案より高い事後確率。A/B テストの判断で使う。

長い式は，正規化定数・事前分布・尤度・事後分布の部品に分けて読むと理解しやすくなります。たとえば $\mathrm{Beta}(\alpha + k, \beta + n - k)$ は，「事前の成功側パラメータに観測成功数 k を足し，失敗側パラメータに観測失敗数 $n - k$ を足す」という更新則として読むと，式の意味が見えやすくなります。

3 はじめに：なぜ β 分布からベイズ理論を学ぶのか

ベータ分布（Beta distribution；以下，β 分布）は，**0 から 1 の値をとる連続確率変数の確率分布**です。区間 $[0, 1]$ に閉じているという特徴から，

- コインの表が出る確率 p
- ウェブサイトのクリック率（CTR）
- 製品の不良率
- 治療法の成功割合
- アンケートの「賛成」比率

のような**「割合」「比率」「確率そのもの」を表す不確実な量**を表現するのに自然な選択肢になります。

β分布は，ベイズ理論を学ぶ入口としても非常に優れています。理由は，**事前分布・尤度・事後分布・予測分布**というベイズ推論の基本部品が，二項分布との組み合わせではほぼ手計算で追えるからです。抽象的に見えがちな「不確実性を分布として持ち，データで更新し，未来を予測する」という考え方を，クリック率や不良率のような具体例で確認できます。

本書では，まず正規分布などしか知らない初学者を念頭に，

(1) β分布とは何を表す分布か（直感）
(2) 確率密度関数の定義とパラメータの意味
(3) ベイズ理論の基本構造（事前・尤度・事後・予測）
(4) ベルヌーイ／二項分布との関係とベイズ更新
(5) 推定・予測・事前分布の選び方
(6) 意思決定・モデル診断・応用例
(7) 共役モデルから階層ベイズや MCMC への発展
(8) Python ／ R による可視化と実行結果の確認

の順で段階的に説明します。

4 β分布の直感：「確率の確率」とは

4.1 コインの例で考える

ふつう「コインの表が出る確率」と言えば $p = 0.5$ と決まった値を思い浮かべます。ところが現実のコインや，未知のシステムでは「**真の確率 p がいくつかわからない**」という状況がよくあります。

たとえば友人から渡された怪しいコインがあるとしましょう。投げる前にあなたは「p は 0.5 付近かもしれないけど，かなりズレているかもしれない」と考えるかもしれません。これは「p **という量に対してさらに確率を割り当てている**」状況です。

> ベータ分布は，このように $[0, 1]$ 区間にある「未知の確率」や「割合」を変数とみなして，それに対する不確実性を表現するための分布です。

4.2 離散から連続へ

理解の橋渡しとして，まず離散的な例を考えます。

- 「真の確率 p は $\{0.1, 0.2, \ldots, 0.9\}$ のどれかであり，それぞれが等しい確率で起こる」と仮定する

と，これは **離散一様分布** です。

- この粒度を細かくしていき，最終的に p が $[0,1]$ 上の任意の実数値を取れるようにし，各値の「相対的な起こりやすさ」を密度関数で表したものが**ベータ分布**です。

4.3 β 分布の使われ方の典型像

ベータ分布で表せる「信念の形」は次のように多彩です。

- **平坦**：何の情報もなく p は 0 から 1 のどれも等しくありえる。
- **中央に集中**：p はだいたい 0.5 付近である。
- **左寄り**：p は小さい値（例：不良率）に偏っている。
- **右寄り**：p は大きい値（例：高い満足率）に偏っている。
- **U 字型**：p は 0 か 1 に近い（極端値）と信じる。

これら全てを **2 つのパラメータ α, β だけ**で表現できる柔軟性が，ベータ分布の最大の魅力です。

5 β 分布の定義

5.1 確率密度関数（PDF）

定義 5.1 (ベータ分布) 連続確率変数 X が区間 $[0,1]$ 上で次の確率密度関数 f をもつとき，X は**形状パラメータ** $\alpha > 0,\ \beta > 0$ のベータ分布に従うといい，$X \sim \mathrm{Beta}(\alpha, \beta)$ と書く：

$$f(x \mid \alpha, \beta) \;=\; \frac{1}{B(\alpha, \beta)}\, x^{\alpha-1}(1-x)^{\beta-1}, \qquad 0 < x < 1. \tag{1}$$

ここで $B(\alpha, \beta)$ は次節で説明する**ベータ関数**で，密度の総積分を 1 にするための正規化定数である。

端点 $x = 0, 1$ における密度の値は α, β によって有限にも無限にもなります。連続分布では一点そのものの確率は 0 なので，実務上は「区間に入る確率」を密度の積分として扱います。

形が難しく見えても，本体は

$$x^{\alpha-1}(1-x)^{\beta-1}$$

というシンプルな式で，「x を $\alpha-1$ 乗」「$(1-x)$ を $\beta-1$ 乗」して掛け合わせただけだと押さえると見通しが良くなります。

5.2 ベータ関数とガンマ関数

正規化定数は

$$B(\alpha, \beta) = \int_0^1 t^{\alpha-1}(1-t)^{\beta-1}\, dt = \frac{\Gamma(\alpha)\,\Gamma(\beta)}{\Gamma(\alpha+\beta)} \tag{2}$$

と書け，ガンマ関数 Γ は $\Gamma(n) = (n-1)!$ という階乗の連続版です。α, β が正の整数ならば

$$B(\alpha, \beta) = \frac{(\alpha-1)!\,(\beta-1)!}{(\alpha+\beta-1)!}$$

となり，計算は組合せ論の式に帰着します。

6 パラメータ α, β の直感的解釈

6.1 擬似カウントとしての解釈

ベータ分布のパラメータは，しばしば**擬似カウント（pseudo-count）**として解釈されます。ただし文献によって次の 2 つの言い方があるので，どちらの意味で使っているかを区別すると混乱しません。

- 共役更新の式では，α を成功側，β を失敗側の**事前カウント**とみなすと見通しが良い。
- 一様事前 $\mathrm{Beta}(1,1)$ から観測を足した事後分布として読むなら，$\alpha-1$ が成功回数，$\beta-1$ が失敗回数に対応する。

たとえば $\mathrm{Beta}(3,2)$ は，「すでに成功 2 回・失敗 1 回を観察した」状況に対応する事後分布のような形をしています。一方で，事前分布そのものの強さを測るときは $\alpha+\beta$ を使うことが多く，事前平均は $\alpha/(\alpha+\beta)$ です。

6.2 形状の代表例

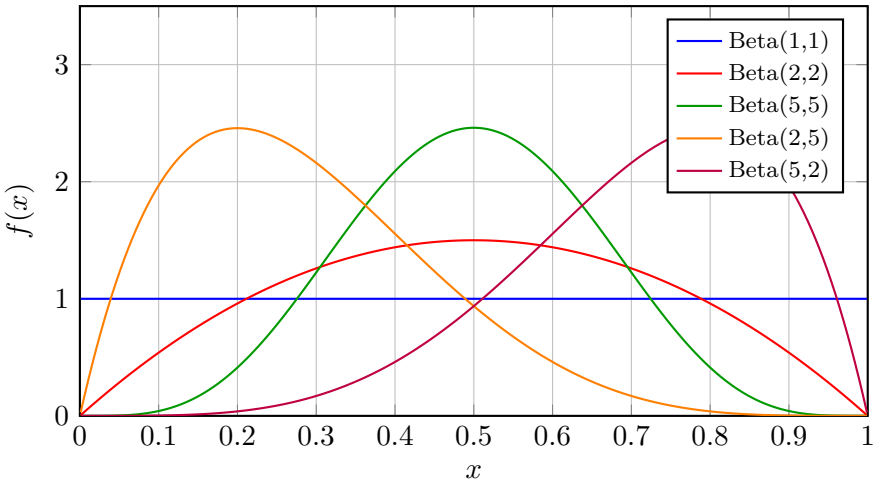
代表的なパラメータと密度の形の対応を表 1 に示します。

表 1　ベータ分布の形状例

(α, β)	形状	解釈・用途
$(1,1)$	一様分布	全く情報がない事前分布
$(2,2)$	中央山型（穏やか）	「だいたい 0.5 付近」とゆるく信じる
$(10,10)$	中央山型（鋭い）	「0.5 付近」をかなり強く信じる
$(2,5)$	左寄り（右に裾）	小さい値の方が起こりやすい
$(5,2)$	右寄り（左に裾）	大きい値の方が起こりやすい
$(0.5,0.5)$	U 字型	0 か 1 付近に集中する極端な信念（ジェフリーズ事前）
$(\alpha,1),\ \alpha>1$	単調増加	$x=1$ に向かって増える分布
$(1,\beta),\ \beta>1$	単調減少	$x=0$ に向かって増える分布

6.3 代表的な密度の形

代表的な密度を一枚にまとめると次のようになります（軸はラフな目盛り）。



（注：Beta$(5,5)$ の係数は $1/B(5,5) = 630$ を用いた。詳細な係数を覚える必要はなく，**形のパターン**だけ覚えれば十分です。）

7　基本的な性質

7.1　期待値・分散・最頻値

性質 7.1 (期待値・分散・最頻値) $X \sim \mathrm{Beta}(\alpha, \beta)$ のとき，

$$\mathbb{E}[X] = \frac{\alpha}{\alpha+\beta}, \tag{3}$$

$$\mathrm{Var}(X) = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}, \tag{4}$$

$$\text{最頻値} = \frac{\alpha-1}{\alpha+\beta-2} \quad (\alpha, \beta > 1 \text{ のとき}). \tag{5}$$

期待値の式は**擬似カウントの感覚と一致**します。「成功比率の見積もり ≈ 成功 / 試行回数」という直観が $\alpha/(\alpha+\beta)$ そのものですね。

7.2　形状についての要点

- $\alpha = \beta$ なら分布は $x = 0.5$ に関して**左右対称**。
- $\alpha + \beta$ が大きいほど分布は**鋭くなる**（不確実性が下がる）。
- $\alpha + \beta$ は**「実効サンプルサイズ」**とも呼ばれ，共役事前分布の信念の強さの指標になる。ただし一様事前からの観測数として数える文脈では $\alpha + \beta - 2$ と解釈することもある。
- $\alpha, \beta < 1$ で **U 字型**になり，$x \to 0, 1$ で密度は発散する（ただし積分は有限）。

7.3　他の分布との関係

- Beta$(1,1)$ は**連続一様分布** $U(0,1)$ と等しい。
- **Dirichlet 分布**はベータ分布の多変量版であり，2 カテゴリの場合は Beta(α, β) に一致する。
- 同じスケールをもつ独立な $Y_1 \sim \mathrm{Gamma}(\alpha)$, $Y_2 \sim \mathrm{Gamma}(\beta)$ に対し $X = Y_1/(Y_1 + Y_2)$ は

$\mathrm{Beta}(\alpha, \beta)$ に従う（ガンマ分布の比による構成）。

8 ベイズ理論の基本地図

ここまでで，β 分布が「$[0, 1]$ 上の未知の比率」を表す分布であることを見ました。次に，その分布をデータでどう更新するかを考えます。

8.1 ベイズ理論が扱うもの

ベイズ理論では，未知量を 1 つの「正解値」としてすぐに固定せず，**未知量に対する不確実性そのもの**を確率分布で表します。たとえばクリック率 p が未知なら，$p = 0.07$ と一点で決め打ちする前に，「0.05 から 0.09 あたりがもっともありそうだが，それ以外も完全には否定しない」という形を分布として持ちます。

この考え方では，推論は次の 4 つの部品に分解されます。

事前分布 データを見る前の未知量への不確実性。

尤度 未知量が仮にある値だったとき，観測データがどれくらい自然か。

事後分布 データを見た後の未知量への不確実性。

予測分布 未知量の不確実性を考慮した，次のデータへの予測。

8.2 ベイズの定理

連続的な未知量 θ とデータ y に対して，ベイズの定理は

$$p(\theta \mid y) = \frac{p(y \mid \theta) p(\theta)}{p(y)} \propto p(y \mid \theta) p(\theta) \tag{6}$$

と書けます。右辺の $p(\theta)$ が事前分布，$p(y \mid \theta)$ が尤度，左辺の $p(\theta \mid y)$ が事後分布です。分母の

$$p(y) = \int p(y \mid \theta) p(\theta)\, d\theta$$

は周辺尤度またはエビデンスと呼ばれ，事後分布を正規化する役割を持ちます。ひとまず「事後分布は，事前分布と尤度を掛けて正規化したもの」と考えれば十分です。

8.3 β 分布が全体像を見せてくれる理由

β 分布を事前分布にし，成功／失敗データをベルヌーイ分布または二項分布で表すと，事後分布も再び β 分布になります。この性質を**共役性**と呼びます。共役性があると，ベイズ理論の核心である「分布を更新する」操作が，複雑な数値計算なしに見えるようになります。

ベイズの部品	β-二項モデルでの形	直感
事前分布	$p \sim \mathrm{Beta}(\alpha, \beta)$	観測前の成功確率への信念
尤度	$K \mid p \sim \mathrm{Binomial}(n, p)$	データの生成モデル
事後分布	$p \mid K = k \sim \mathrm{Beta}(\alpha + k, \beta + n - k)$	データ後の信念
予測分布	$\Pr(Y_{\mathrm{next}} = 1 \mid k)$	次の観測への予測

この小さなモデルを理解すると，ベイズ理論の多くの考え方は「未知量がもっと多い」「手計算できないので数値計算する」という方向に自然に拡張できます。

9 ベイズ更新：二項分布の共役事前分布として

ベータ分布が統計実務で最もよく使われるのは，**ベイズ推論**における**共役事前分布**としての性質です。これは「事後分布も再びベータ分布になる」という非常に便利な性質を指します。

9.1 設定

コインを n 回投げて成功（表）を k 回観測したとします。

- モデル：$K \mid p \sim \mathrm{Binomial}(n, p)$
- 事前分布：$p \sim \mathrm{Beta}(\alpha, \beta)$

9.2 事後分布の導出

ベイズの定理より

$$f(p \mid k) \;\propto\; \underbrace{\binom{n}{k} p^k (1-p)^{n-k}}_{\text{尤度}} \cdot \underbrace{\frac{1}{B(\alpha,\beta)} p^{\alpha-1}(1-p)^{\beta-1}}_{\text{事前}}.$$

p に関係しない係数を除いて整理すると

$$f(p \mid k) \;\propto\; p^{\alpha+k-1}(1-p)^{\beta+(n-k)-1}.$$

これはふたたびベータ分布の核なので，

$$\boxed{p \mid k \;\sim\; \mathrm{Beta}\big(\alpha + k,\; \beta + (n-k)\big).} \tag{7}$$

事後平均は

$$\mathbb{E}[p \mid k] = \frac{\alpha + k}{\alpha + \beta + n} = \frac{\alpha + \beta}{\alpha + \beta + n} \cdot \frac{\alpha}{\alpha + \beta} + \frac{n}{\alpha + \beta + n} \cdot \frac{k}{n}. \tag{8}$$

つまり，事後平均は「事前平均」と「標本比率 k/n」の加重平均です。データ数 n が増えるほど標本比率の重みが大きくなります。

9.3 擬似カウントの再解釈

事後パラメータは「**事前の擬似カウントに，観測値を足すだけ**」という極めて単純な形をしています。

$$\alpha_{\mathrm{post}} = \alpha_{\mathrm{prior}} + (\text{成功数}), \quad \beta_{\mathrm{post}} = \beta_{\mathrm{prior}} + (\text{失敗数}).$$

ベイズ更新がここまで簡単に書けるのは，ベータ分布が二項分布の**共役族**だからです。

9.4 逐次更新

データが時間と共に少しずつ入る場合も，前ステップの事後分布を次の事前分布として再利用するだけで，全データを使ったときと同じ答えが得られます（ベイズ更新の整合性）。これはオンライン推定や A/B テストの逐次評価で重宝される性質です。

9.5 次の観測を予測する

事後分布が $\mathrm{Beta}(\alpha + k, \beta + n - k)$ のとき，次の 1 回が成功する予測確率は

$$\Pr(Y_{\text{next}} = 1 \mid k) = \mathbb{E}[p \mid k] = \frac{\alpha + k}{\alpha + \beta + n}$$

です。未知の p を一点推定してから予測するのではなく，p の不確実性を積分して予測している，というのがベイズ的な見方です。

10 推定・信用区間・予測分布

10.1 事後分布をどう要約するか

ベイズ推論の結果は，本来は事後分布そのものです。しかし実務では，意思決定や報告のために，事後分布をいくつかの数値で要約することが多くあります。β 分布では，代表的な要約量は次のようになります。

要約量	β 分布での形	使いどころ
事後平均	$\dfrac{\alpha + k}{\alpha + \beta + n}$	二乗誤差を重視する点推定
事後中央値	数値的に求める	外れた裾の影響を少し抑えたい場合
事後最頻値	$\dfrac{\alpha + k - 1}{\alpha + \beta + n - 2}$	最も密度が高い値（両形状 > 1 のとき）
事後分散	β 分布の分散式に事後パラメータを代入	不確実性の大きさ

重要なのは，点推定だけで結論を出さないことです。たとえば事後平均が 0.08 でも，分布が鋭く集中している 0.08 と，広くばらついている 0.08 では意味がまったく違います。

10.2 信用区間

ベイズ推論では，事後分布の分位点から**信用区間**を作れます。たとえば $p \mid k \sim \mathrm{Beta}(\alpha + k, \beta + n - k)$ の 95% 信用区間は，

$$[Q_{0.025},\, Q_{0.975}]$$

のように，事後分布の 2.5% 点と 97.5% 点で表せます。この区間は「モデルと事前分布を前提に，p がこの範囲に入る事後確率が 95%」という意味を持ちます。頻度主義の信頼区間とは解釈が違うので，文章で報告するときは「信頼」ではなく「信用」と書き分けるのが無難です。

10.3 複数回先の予測：β-二項分布

次の 1 回だけでなく, 今後 m 回の試行で成功が $\tilde{k}$ 回起こる確率も求められます。事後分布が $\mathrm{Beta}(a, b)$ のとき,

$$\Pr(\tilde{K} = \tilde{k} \mid k) = \binom{m}{\tilde{k}} \frac{B(a + \tilde{k},\, b + m - \tilde{k})}{B(a, b)}, \qquad a = \alpha + k, \quad b = \beta + n - k. \tag{9}$$

これは**β-二項分布**です。単に p を事後平均に固定して二項分布で予測するよりも，p 自体の不確実性を含むため，予測分布は広くなります。少ないデータで将来のクリック数や不良数を予測するとき，この差は実務上かなり重要です。

10.4 逐次学習としての見方

β-二項モデルでは，1 件ずつデータが来るたびに

$$(\alpha, \beta) \leftarrow \begin{cases} (\alpha + 1, \beta) & \text{成功を観測したとき,} \\ (\alpha, \beta + 1) & \text{失敗を観測したとき} \end{cases}$$

と更新できます。これは，ベイズ推論が自然にオンライン学習になっていることを示します。データを全部保存し直さなくても，現在の事後分布が過去データの要約として機能します。

11 事前分布の選び方

事後分布や予測分布を使う前に，事前分布の選び方を確認しておきます。「事前にとくに知識がない」というとき，ベータ分布のどれを使うべきかは実はよく議論されてきました。

表 2　代表的な無情報／弱情報事前分布

名称	形	特徴
一様事前	$\mathrm{Beta}(1, 1)$	最も単純。p の値を一様に評価する。
ジェフリーズ事前	$\mathrm{Beta}(0.5, 0.5)$	Bernoulli モデルで再パラメータ化に不変な事前。
Haldane 事前	$\mathrm{Beta}(0, 0)$	端点を強く重視する極限的な事前。**真の確率分布ではない**（improper）。
弱情報事前	$\mathrm{Beta}(2, 2)$ など	0/1 端点を緩く避けたい場合に使う実用的選択。

結論の目安：データ量が十分に多いなら何を選んでも結論はほぼ変わらない。データが少ない／ゼロ観測がある状況では選択が結果に効いてくるので，$\mathrm{Beta}(1, 1)$ **か** $\mathrm{Beta}(2, 2)$ **を使い，感度分析を行う**のが現実的にバランスが良いでしょう。Haldane 事前を使った事後分布 $\mathrm{Beta}(k, n - k)$ は，成功と失敗が少なくとも 1 回ずつ観測された場合にだけ正則になります。

12　ベイズ推論を意思決定につなげる

12.1　推定だけでは意思決定にならない

ベイズ理論の強みは，未知量を推定するだけでなく，**不確実性を残したまま意思決定に進める**点にあります。ここでいう A/B テストとは，広告文，ボタンの色，バナー，価格表示などの 2 つの案 A と B を同じ時期にランダムにユーザーへ出し分け，クリック率や購入率などの指標を比較する実験です。ランダムに割り付けることで，季節性やユーザー層の違いではなく，案そのものの差を見ようとします。たとえば A/B テストで $\Pr(p_B > p_A) = 0.94$ だったとしても，すぐに B を採用すべきとは限りません。改善幅が小さい，変更コストが大きい，ユーザー体験への副作用がある，といった要素も判断に入るからです。

意思決定を明示するには，行動 a と未知量 θ に対する損失 $L(a, \theta)$ を置き，事後期待損失

$$\mathbb{E}\big[L(a,\theta) \mid y\big]$$

が小さい行動を選びます。この考え方を**ベイズ意思決定理論**と呼びます。

12.2　損失関数と代表的な点推定

点推定は，損失関数を決めたときの最適行動として理解できます。

損失関数	最適な点推定	意味
二乗損失 $(a-\theta)^2$	事後平均	大きな誤差を強く罰する
絶対損失 $\lvert a-\theta\rvert$	事後中央値	誤差の大きさを線形に罰する
0-1 損失	事後最頻値	最もありそうな値を選ぶ

したがって「平均・中央値・最頻値のどれを出すべきか」は，数学的な好みではなく，誤差をどう評価するかという意思決定上の問題です。

12.3　A/B テストの意思決定

B を採用すると 1 表示あたりの価値が v，今後の表示数が N，切り替えコストが C だとします。このとき B 採用の期待純便益は

$$\mathbb{E}\left[Nv(p_B - p_A) - C \mid \text{data}\right]$$

で評価できます。$\Pr(p_B > p_A)$ が高くても，期待リフトが小さければ便益はコストに負けるかもしれません。逆に確率が圧倒的でなくても，上振れ時の価値が大きいなら，追加実験や段階的ロールアウトが合理的になる場合があります。

13　モデルチェックと感度分析

13.1　事前分布への感度

ベイズ推論では，事前分布が結論に影響します。これは欠点ではなく，仮定が明示されるという特徴です。ただし，データが少ないときほど影響は大きくなるので，実務では複数の事前分布で結論を比べ

る**感度分析**が重要です。

確認する事前分布	目的	典型例
一様事前	基準線を作る	$\mathrm{Beta}(1,1)$
弱情報事前	極端な値を少し抑える	$\mathrm{Beta}(2,2)$
悲観的事前	過大評価を避ける	小さい成功率に重心を置く β 分布
楽観的事前	既存知識を反映する	過去実験の平均に重心を置く β 分布

結論が事前分布を変えてもほぼ同じなら，データが十分に強いと判断できます。結論が大きく変わるなら，「まだデータだけでは決めきれない」と報告する方が誠実です。

13.2 事後予測チェック

モデルがデータを十分に説明できているかを見るには，事後分布から未知量をサンプリングし，さらにそこから擬似データを生成します。この擬似データと実データを比べる方法を**事後予測チェック**と呼びます。

β-二項モデルなら，次のような点を確認できます。

- 観測された成功数が，事後予測分布の中心から極端に外れていないか。
- 日ごと・ユーザー群ごとに成功率が大きく違うのに，1 つの p で無理に表していないか。
- 時系列の変化やキャンペーン効果を無視していないか。

モデルチェックの目的は「モデルが真実か」を証明することではありません。どの単純化が結論に効いているかを見つけ，必要ならモデルを改善することです。

13.3 独立性と過分散

β-二項モデルの基本形では，各試行が条件付き独立で，同じ成功確率 p を共有すると考えます。しかし現実には，ユーザーごとの違い，日ごとの変動，広告面の違いなどにより，二項分布が想定するより成功数のばらつきが大きくなることがあります。これを**過分散**と呼びます。

過分散が強い場合，単純な二項モデルは不確実性を過小評価しがちです。その場合は，グループごとに異なる p_j を持つ階層モデルや，説明変数を入れたベイズロジスティック回帰を検討します。

14 ベイズ理論の応用例

14.1 A/B テストでのクリック率推定

A/B テストでは，利用者を A 案を見る群と B 案を見る群に分け，各群で「クリックした／しなかった」のような 2 値データを集めます。ここでは各案の真のクリック率をそれぞれ p_A, p_B とし，観測されたクリック数から「どちらの案が良さそうか」を推定します。

ウェブサイトの 2 つのバナー A, B について

	表示数	クリック数
A	1000	60
B	1000	75

という結果が得られたとします。事前分布として情報のない $\mathrm{Beta}(1,1)$ を採用すれば，各 CTR の事後分布は

$$p_A \sim \mathrm{Beta}(61,\, 941),$$
$$p_B \sim \mathrm{Beta}(76,\, 926).$$

モンテカルロサンプリングで $\Pr(p_B > p_A)$ を計算すれば，「B の方が良い確率」を直接見積もれます。古典的検定とは違い，**比較可能な確率値**が得られるのが大きな利点です。ただしこの確率は事前分布とモデルに依存し，検定の p 値とは意味が異なります。

14.2 Thompson サンプリング（多腕バンディット）

各腕（選択肢）について事後ベータ分布を保持し，各ラウンドで「事後分布から 1 つサンプリングして最大値の腕を選ぶ」という戦略は **Thompson サンプリング**と呼ばれます。ベータ分布の更新が単純な足し算で済むため，オンライン広告や推薦系で高速に動作することが知られています。

14.3 プロジェクト管理：PERT

3 点見積もり（楽観値 a, 最頻値 m, 悲観値 b）を持つ作業時間の推定で

$$\mathbb{E}[T] \approx \frac{a + 4m + b}{6}$$

という有名な式は，区間 $[a, b]$ に線形変換したベータ分布を使う PERT 型の近似です。実装によって形状パラメータの置き方は少し異なりますが，「最頻値を重く見た平均」を作るという考え方は共通しています。

14.4 品質管理・不良率推定

ある製品ロットで n 個検査して不良が k 個見つかったとき, 不良率 p の事後分布 $\mathrm{Beta}(1{+}k,\, 1{+}n{-}k)$ から信用区間を作れば，$k = 0$ でも「事後確率 95% で上限がどの程度か」が自然に求まります（古典的な信頼区間が困難な**ゼロ観測**の場合に特に有用）。

15 共役性を越えたベイズ理論

ここまでの応用例は，β-二項モデルの共役性を使うと非常に見通しよく扱えます。一方，現実の分析では説明変数，グループ差，時系列変化などを入れたくなる場面が多くあります。そのときは，共役性の外へ進む必要があります。

15.1 共役モデルの位置づけ

β-二項モデルは，共役モデルの代表例です。共役モデルは手計算しやすく，ベイズ理論の構造を学ぶには最適ですが，現実の問題が常に共役で書けるわけではありません。代表的な共役関係には次のようなものがあります。

データモデル	共役事前分布	典型的な用途
Bernoulli/Binomial	Beta	成功率・クリック率・不良率
Multinomial	Dirichlet	カテゴリ比率・単語分布
Poisson	Gamma	件数・発生率
Normal（分散既知）	Normal	平均値の推定

共役性がない場合でも，ベイズの定理そのものは変わりません。変わるのは，事後分布を手計算で書けないため，数値計算が必要になる点です。

15.2 階層ベイズ：複数の比率を同時に学ぶ

複数の店舗，広告，医療施設などで成功率を推定したい場合，各グループを完全に別々に推定すると，データが少ないグループの推定が不安定になります。一方，全グループを同じ成功率だとみなすと，個別差を潰しすぎます。

階層ベイズでは，たとえば

$$K_j \mid p_j \sim \mathrm{Binomial}(n_j, p_j), \tag{10}$$
$$p_j \mid \alpha, \beta \sim \mathrm{Beta}(\alpha, \beta) \tag{11}$$

のように，グループごとの成功率 p_j が，共通の母集団分布から来たと考えます。この構造により，データが少ないグループの推定は全体平均へ適度に引き寄せられます。この現象を**部分プーリング**または**縮小推定**と呼びます。

15.3 MCMC と近似推論

階層モデルや回帰モデルでは，事後分布を閉じた式で書けないことが多くあります。その場合は，次のような数値的手法を使います。

- **MCMC**：事後分布からのサンプル列を作り，平均・分位点・確率を近似する。
- **変分推論**：扱いやすい分布で事後分布を近似し，高速に推定する。
- **ラプラス近似**：事後分布の山の近くを正規分布で近似する。

どの手法を使っても，基本は同じです。事前分布と尤度を決め，事後分布を求め，事後分布から予測と意思決定を行います。β分布で学んだ小さな例は，そのまま大きなベイズモデルの設計図になります。

16 Python による可視化と実装

以下のコードは計算の要点です。実行可能な完全版は `scripts/example01_pdf_plot.py`，`scripts/example02_bayesian_update.py`，`scripts/example03_ab_test.py` に置いてあります。Python 版は `uv run` で実行します。

16.1 密度関数の描画

```python
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import beta

x = np.linspace(0, 1, 400)
params = [(1, 1), (2, 2), (5, 5), (2, 5), (5, 2), (0.5, 0.5)]

plt.figure(figsize=(8, 5))
for a, b in params:
    plt.plot(x, beta.pdf(x, a, b), label=f"Beta({a},{b})")
plt.xlabel("x"); plt.ylabel("f(x)")
plt.title("Beta distributions for various (alpha, beta)")
plt.legend(); plt.grid(alpha=0.3); plt.tight_layout()
plt.savefig("beta_pdf.png", dpi=150)
```

16.2 ベイズ更新の可視化

```python
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import beta

# 真の p = 0.3。観測数が増えると事後分布がどう変わるか
rng = np.random.default_rng(0)
true_p = 0.3
n_max  = 200
data   = rng.binomial(1, true_p, size=n_max)

alpha0, beta0 = 1, 1   # 一様事前
snapshots = [0, 5, 20, 50, 200]

x = np.linspace(0, 1, 400)
plt.figure(figsize=(8, 5))
for n in snapshots:
    k = int(data[:n].sum())
    a_post = alpha0 + k
    b_post = beta0  + (n - k)
    plt.plot(x, beta.pdf(x, a_post, b_post),
             label=f"n={n}, k={k}")
plt.axvline(true_p, color="k", linestyle=":")
plt.xlabel("p"); plt.ylabel("posterior density")
plt.title("Sequential Bayesian update of Beta posterior")
plt.legend(); plt.grid(alpha=0.3); plt.tight_layout()
plt.savefig("beta_update.png", dpi=150)
```

データが増えるにつれて事後分布がだんだん真値 $p = 0.3$ のまわりに集中していく様子を観察できます。これがベイズ推論の「学習」です。

16.3 A/B テストの事後比較

```python
import numpy as np
from scipy.stats import beta

# A: 60/1000 クリック, B: 75/1000 クリック
a_A, b_A = 1 + 60, 1 + (1000 - 60)
a_B, b_B = 1 + 75, 1 + (1000 - 75)

rng     = np.random.default_rng(0)
samples = 200_000
p_A = beta.rvs(a_A, b_A, size=samples, random_state=rng)
p_B = beta.rvs(a_B, b_B, size=samples, random_state=rng)

print("P(p_B > p_A) =", (p_B > p_A).mean())
print("expected lift =", ((p_B - p_A) / p_A).mean())

for q in (0.025, 0.5, 0.975):
    diff_q = float(np.quantile(p_B - p_A, q))
    print(f"p_B - p_A quantile {q:.3f} = {diff_q:+.4f}")
```

このように，検定の p 値ではなく**「B が A より良い確率」**を直接出せるのがベイズ的アプローチの大きな利点です。

17 R による可視化と実装

R では `dbeta`, `rbeta`, `rbinom` などの標準関数だけで，Python 版と同じ計算を再現できます。ここでは外部パッケージを使わず，base R の描画機能で実装します。実行可能な完全版は `scripts/example01_pdf_plot.R`, `scripts/example02_bayesian_update.R`, `scripts/example03_ab_test.R` に置いてあります。

17.1 密度関数の描画

```r
x <- seq(0.001, 0.999, length.out = 400)
params <- list(c(1, 1), c(2, 2), c(5, 5), c(2, 5), c(5, 2), c(0.5, 0.5))
cols <- c("black", "red", "blue", "orange", "purple", "darkgreen")

png("beta_pdf_r.png", width = 1200, height = 750, res = 150)
plot(NULL, xlim = c(0, 1), ylim = c(0, 3.5),
     xlab = "x", ylab = "f(x)",
     main = "Beta distributions for various (alpha, beta)")

for (i in seq_along(params)) {
  a <- params[[i]][1]
  b <- params[[i]][2]
  lines(x, dbeta(x, shape1 = a, shape2 = b), col = cols[i], lwd = 2)
}
```

```r

legend("topright",
       legend = vapply(params, function(p) sprintf("Beta(%g,%g)", p[1], p[2]), ""),
       col = cols, lwd = 2, bty = "n")
grid()
dev.off()
```

17.2 ベイズ更新の可視化

```r
set.seed(0)
true_p <- 0.3
n_max <- 200
data <- rbinom(n_max, size = 1, prob = true_p)

alpha0 <- 1
beta0 <- 1
snapshots <- c(0, 5, 20, 50, 200)
x <- seq(0, 1, length.out = 400)
cols <- c("black", "red", "blue", "orange", "purple")
labels <- character(length(snapshots))

png("beta_update_r.png", width = 1200, height = 750, res = 150)
plot(NULL, xlim = c(0, 1), ylim = c(0, 14),
     xlab = "p", ylab = "posterior density",
     main = "Sequential Bayesian update of Beta posterior")

for (i in seq_along(snapshots)) {
  n <- snapshots[i]
  k <- sum(data[seq_len(n)])
  a_post <- alpha0 + k
  b_post <- beta0 + (n - k)
  lines(x, dbeta(x, a_post, b_post), col = cols[i], lwd = 2)
  labels[i] <- sprintf("n=%d, k=%d", n, k)
}

abline(v = true_p, lty = 3)
legend("topright", legend = labels, col = cols, lwd = 2, bty = "n")
grid()
dev.off()
```

17.3 A/B テストの事後比較

```r
# A: 60/1000 クリック, B: 75/1000 クリック
a_A <- 1 + 60
b_A <- 1 + (1000 - 60)
a_B <- 1 + 75
b_B <- 1 + (1000 - 75)

set.seed(0)
samples <- 200000
p_A <- rbeta(samples, shape1 = a_A, shape2 = b_A)
p_B <- rbeta(samples, shape1 = a_B, shape2 = b_B)
```

```r

prob_B_better <- mean(p_B > p_A)
expected_lift <- mean((p_B - p_A) / p_A)

cat(sprintf("P(p_B > p_A) = %.4f\n", prob_B_better))
cat(sprintf("expected lift = %.4f\n", expected_lift))

probs <- c(0.025, 0.5, 0.975)
qs <- quantile(p_B - p_A, probs = probs)
for (i in seq_along(probs)) {
  cat(sprintf("p_B - p_A quantile %.3f = %+.4f\n",
              probs[i], qs[i]))
}
```

R では密度評価に `dbeta`，乱数生成に `rbeta` を使います。Python の `scipy.stats.beta.pdf` と `beta.rvs` に対応する関数だと考えると，両言語の実装を対応づけやすくなります。

18 スクリプト実行結果と分析

この節では，リポジトリ内の Python スクリプトと R スクリプトを実行した結果を確認します。図の出力先は `figures/` です。

```
uv run scripts/example01_pdf_plot.py
uv run scripts/example02_bayesian_update.py
uv run scripts/example03_ab_test.py

Rscript scripts/example01_pdf_plot.R
Rscript scripts/example02_bayesian_update.R
Rscript scripts/example03_ab_test.R
```

18.1 図による確認

この小節では，分布の形と更新の様子を図で確認します。数値として比較したい結果は，次の小節以降に表でまとめます。

密度関数の描画では，Python 版が `figures/ex01_beta_pdf.png`，R 版が `figures/ex01_beta_pdf_r.png` を生成します。Beta$(1,1)$ は一様，Beta$(5,5)$ は 0.5 付近に集中し，Beta$(2,5)$ と Beta$(5,2)$ は左右に歪みます。Beta$(0.5,0.5)$ は 0 と 1 の近くで密度が高く，「極端な値を取りやすい」事前分布として読めます。

ベイズ更新の可視化では，Python 版が `figures/ex02_beta_update.png`，R 版が `figures/ex02_beta_update_r.png` を生成します。観測数が増えるほど事後分布は細くなり，信用区間が狭くなります。ただし，同じ `set.seed(0)` や `default_rng(0)` を使っても，Python と R では乱数生成器が異なるため，観測された成功数は一致しません。

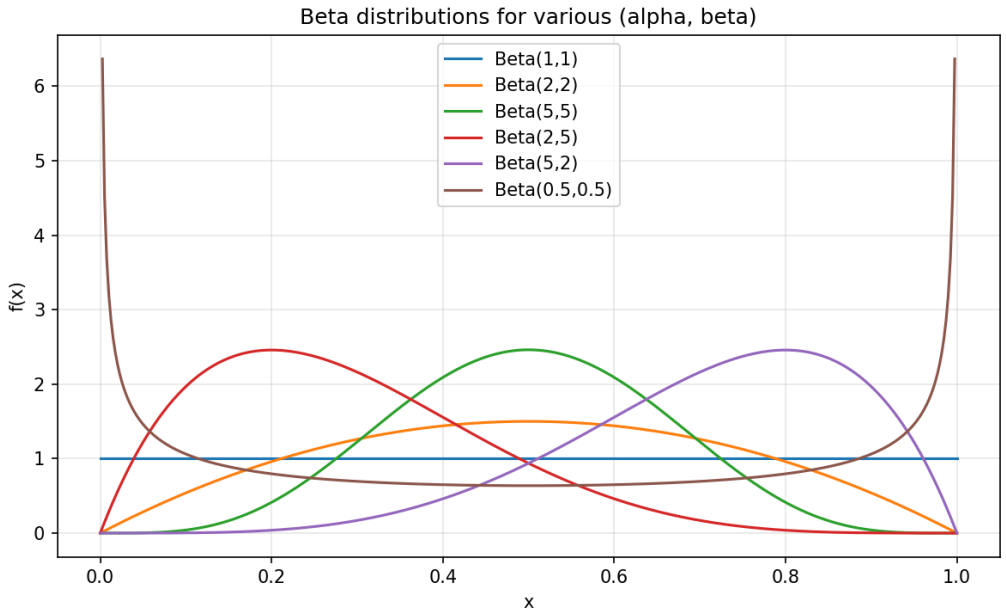


図 1　Python で生成した β 分布の密度曲線

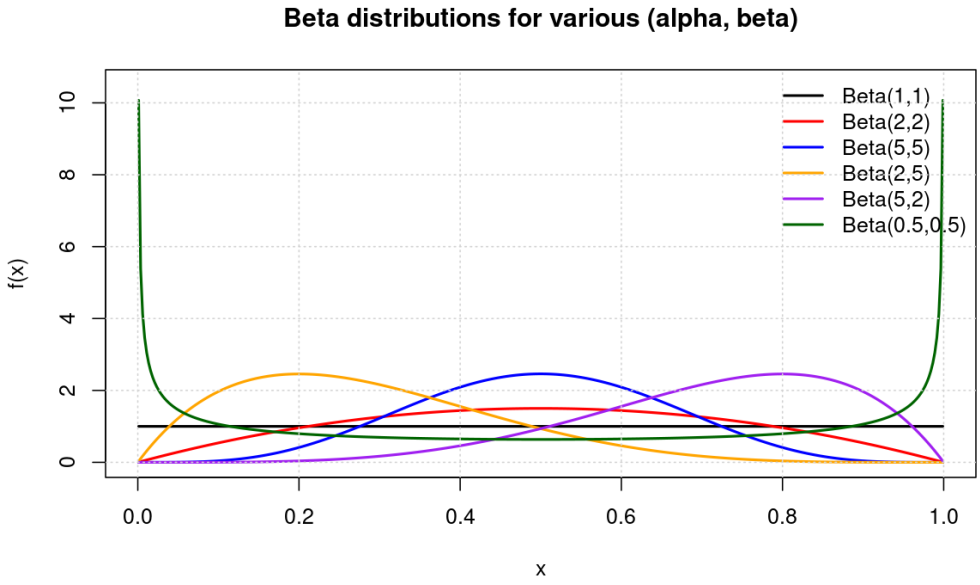


図 2　R で生成した β 分布の密度曲線

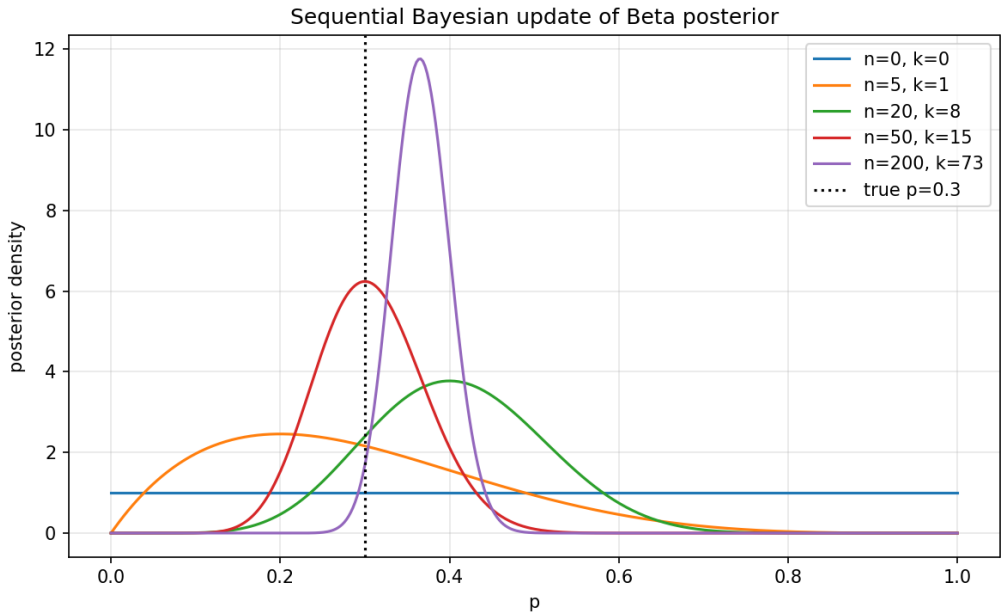


図 3　Python で生成したβ事後分布の逐次更新

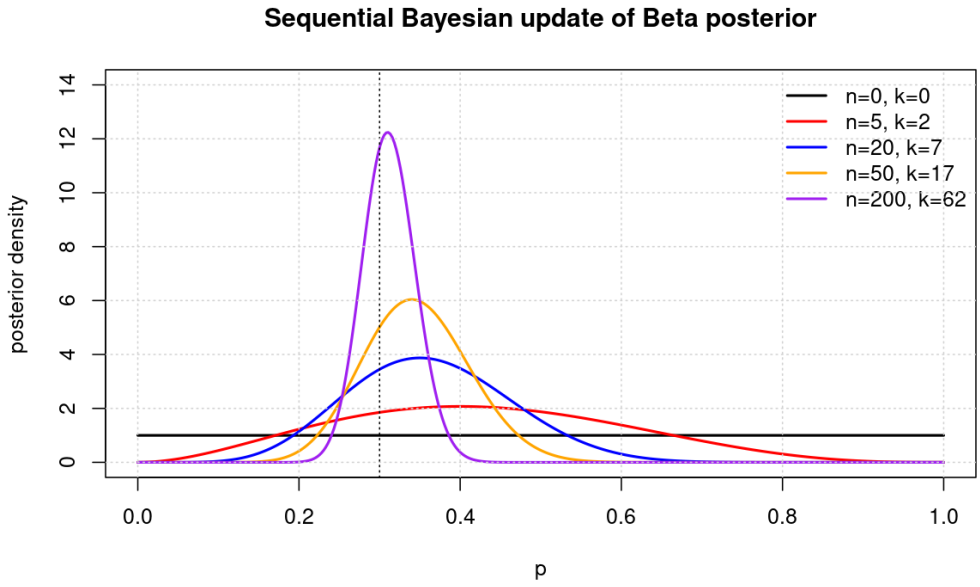


図 4　R で生成したβ事後分布の逐次更新

18.2 ベイズ更新の数値結果

図で見た事後分布の収束を，実行結果の数値で確認します。観測数 n が増えるほど，事後平均は観測された成功率に近づき，95% 信用区間は狭くなります。

表 3　ベイズ更新スクリプトの数値出力

実装	n	成功数 k	事後平均	95% 信用区間
Python	0	0	0.5000	[0.0250, 0.9750]
Python	5	1	0.2857	[0.0433, 0.6412]
Python	20	8	0.4091	[0.2182, 0.6156]
Python	50	15	0.3077	[0.1911, 0.4383]
Python	200	73	0.3663	[0.3014, 0.4338]
R	0	0	0.5000	[0.0250, 0.9750]
R	5	2	0.4286	[0.1181, 0.7772]
R	20	7	0.3636	[0.1811, 0.5697]
R	50	17	0.3462	[0.2243, 0.4792]
R	200	62	0.3119	[0.2500, 0.3773]

18.3 A/B テストの数値結果

A 案は 60/1000 クリック，B 案は 75/1000 クリックでした。Beta$(1,1)$ 事前分布を置くと，A の事後分布は Beta$(61,941)$，B の事後分布は Beta$(76,926)$ になります。

この小節では図を使わず，観測データとモンテカルロ要約を表だけで整理します。表 4 は入力データと事後分布，表 5 は事後分布から計算した比較指標です。

表 4　A/B テストの観測データと事後分布

案	表示数	クリック数	非クリック数	観測 CTR	事後分布
A	1000	60	940	0.060	Beta$(61,941)$
B	1000	75	925	0.075	Beta$(76,926)$

表 5　A/B テストのモンテカルロ要約

実装	$\Pr(p_B > p_A)$	期待リフト	差 2.5%	差 中央値	差 97.5%
Python	0.9087	26.51%	-0.0071	+0.0149	+0.0372
R	0.9100	26.63%	-0.0070	+0.0150	+0.0370

表 5 の「期待リフト」は $(p_B - p_A)/p_A$ の事後平均で，相対的な改善率を表します。一方，「差」は $p_B - p_A$ そのものなので，クリック率の絶対差です。この 2 つを分けると，結果の読み方がはっきりします。

どちらの実装でも，B が A より良い確率は約 91% です。差の中央値は約 0.015，つまりクリック率で約 1.5 ポイントの改善です。一方で，差の 95% 信用区間は下側がわずかに負なので，「B が必ず良い」とまでは言えません。期待リフトは約 26.5% ですが，相対値だけでなく絶対差と変更コストも併せて判

断する必要があります。この結果なら，変更コストが小さい場合は B 採用を検討できますが，実装コストや副作用が大きい場合は追加データを集める判断も合理的です。

19 よくある誤解と注意点

1. α, β **は整数である必要はない。**擬似カウントの比喩からくる誤解。実数なら何でもよい。
2. **ベータ分布は確率分布であり，「確率の確率」という言葉は比喩。**厳密には「$[0,1]$ にとる連続確率変数の分布」。
3. $\mathrm{Beta}(1,1)$ **は完全な無情報ではない。**パラメータ変換に対して不変ではないため，変換後には情報がのる。再パラメータ化に対する不変性が必要なら Jeffreys 事前を検討する。
4. $\alpha = \beta = 0$ **の Haldane 事前は確率分布ではない**（improper）。事後が正則になるには，成功と失敗が少なくとも 1 回ずつ必要。
5. **ベータ分布は** $[0,1]$ **のみ。**他の区間で使いたいときは線形変換を施した**一般化ベータ分布**（あるいは 4 パラメータベータ）を用いる。
6. **独立性が必要。**観測値同士が独立でない場合（時系列・クラスター構造）には単純なベルヌーイ／二項モデルは不適切で，ベータ分布の素直な共役更新も成り立たない。
7. **ベイズ推論は「好きな事前分布で好きな結論を出す方法」ではない。**事前分布は明示し，データとの関係を確認し，感度分析で結論の頑健性を調べる。
8. **事後確率と頻度主義の** p **値は別物。**$\Pr(p_B > p_A \mid \mathrm{data})$ は仮説の事後確率だが，検定の p 値は「帰無仮説のもとで，観測以上に極端なデータが出る確率」。
9. **MCMC はベイズ理論そのものではない。**MCMC は手計算できない事後分布を近似するための数値計算法であり，ベイズ理論の中心は事前分布・尤度・事後分布・予測分布の関係にある。

20 まとめ

- β 分布は $[0,1]$ 上の未知の比率を表す分布であり，ベイズ理論における**事前分布**の役割を直感的に理解しやすい。
- β-二項モデルでは，事前分布・尤度・事後分布・予測分布がすべて明示でき，ベイズ更新は成功数と失敗数を足すだけで表せる。
- 事後平均，信用区間，β-二項予測分布を使うと，点推定だけでなく不確実性を含めた推定と予測ができる。
- ベイズ推論は推定で終わらず，損失関数や期待便益を通じて意思決定に接続できる。
- 事前分布への感度，事後予測チェック，独立性や過分散の確認は，ベイズ分析を実務で使うための重要な診断である。
- 共役性を越えると階層ベイズ，ベイズ回帰，MCMC などが必要になるが，基本構造は β 分布で学んだ小さなモデルと同じである。
- Python の `scipy.stats.beta` や R の `dbeta`/`rbeta` で密度評価・サンプリングが簡単にでき，理

論と実装を往復しながら学べる。

β分布を理解することは，ベイズ理論の入口にすぎません。しかし，この入口はとてもよくできています。小さな β-二項モデルの中に，不確実性の表現，データによる更新，未来の予測，意思決定，モデル診断，階層化への発展がすべて含まれているからです。ここから Dirichlet 分布，ベイズ回帰，階層ベイズモデルへ進むと，扱える問題の幅は大きく広がります。

参考文献・さらに学ぶには

- Gelman, A. et al., *Bayesian Data Analysis*, 3rd ed., CRC Press, 2013.（ベイズ統計の定番教科書）
- McElreath, R., *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*, 2nd ed., CRC Press, 2020.（直感重視で読みやすい）
- Downey, A. B., *Think Bayes: Bayesian Statistics in Python*, 2nd ed., O'Reilly Media, 2021.（Python で動かしながらベイズを学べる）
- Kruschke, J. K., *Doing Bayesian Data Analysis: A Tutorial with R, JAGS, and Stan*, 2nd ed., Academic Press, 2015.（実装と解釈を重視した入門書）
- Murphy, K. P., *Probabilistic Machine Learning: An Introduction*, The MIT Press, 2022.（確率モデル全般への発展）
- 久保拓弥『データ解析のための統計モデリング入門：一般化線形モデル・階層ベイズモデル・MCMC』岩波書店, 2012.（和書のベイズ／ GLM 入門。共役事前にも触れている）