現代的ベイズ統計学の基礎と応用

数理的基礎から高次元・深層学習・意思決定まで

Yugo Nakayama

February 12, 2026

Contents

Part I

ベイズ推論の基礎

Chapter 1

確率の基礎とベイズの定理

本章では、ベイズ統計学の土台となる「確率の考え方」と、すべての推論の基礎となる「ベイズの定理」を数理的・直感的に理解します。

1.1　頻度論とベイズ論の「確率」の違い

確率（Probability）には、歴史的に 2 つの解釈があります。

- **頻度論的確率 (Frequentist Probability)**: 「コインを無限回投げたとき、表が出る割合」。客観的であり、繰り返し試行が可能であることを前提とします。
- **ベイズ的確率 (Bayesian Probability)**:「ある事象が真であるという確信の度合い（Belief）」。主観的確率とも呼ばれ、**「情報が得られるたびに更新される」**のが特徴です。

ベイズ統計では、パラメータ（例：真の故障率）を「未知の定数」ではなく「確率変数（分布を持つもの）」として扱います。

1.2　同時確率・周辺確率・条件付き確率

ここでは、ベイズ統計を支える 3 つの基本的な確率の考え方を整理します。数式を丸暗記するのではなく、**「全体のうち、どこに注目しているか」**という視点が重要です。

1.2.1　同時確率 (Joint Probability)：$P(X, Y)$

「2 つのことが同時に起きる」確率です。

- **イメージ**: 2 つの条件が重なった「集合の共通部分」の面積。
- **高校生向けの例**: 「クラスメイトを選んだとき、その人が『男子』かつ『メガネをかけている』」確率。
- **数式**: X という事象と Y という事象がともに発生する確率。

1.2.2　周辺確率 (Marginal Probability)：$P(X)$

「もう片方の条件を無視して、片方だけに注目した」確率です。

- **イメージ**: 複雑な表を、縦（または横）に合計した「端っこ（周辺）」の数字。
- **高校生向けの例**: メガネの有無に関わらず、単純に「その人が『男子』である」確率。
- **数式**: 全ての Y のパターンを足し合わせる（周辺化する）ことで求められます。

$$P(X) = \sum_{Y} P(X, Y)$$

1.2.3 条件付き確率 (Conditional Probability)：$P(X|Y)$

「あることが起きたとわかった後で、もう片方が起きる」確率です。ここがベイズ統計の最も重要な入り口になります。

- **イメージ:** **「世界の切り替わり」**です。分母が「全体」から「Y という世界」に縮小します。
- **高校生向けの例:** 「選んだ人が『メガネをかけている人（Y）』だとわかった。その限定されたメンバーの中で、さらに『男子（X）』である」確率。
- **数式:**

$$P(X|Y) = \frac{P(X,Y)}{P(Y)}$$

注意 1.1 (コラム：直感的な理解)**.** 確率の計算に迷ったら、**「面積（人数）」**で考えるとスッキリします。

1. **同時確率** $P($**男子**$,$ **メガネ**$)$*:* クラス全員（例えば *40* 人）のうち、メガネ男子の人数が占める割合。
2. **条件付き確率** $P($**男子** $|$ **メガネ**$)$*:* クラスの**「メガネをかけている人たちだけ」を集めたグループ**の中で、男子が占める割合。

ポイント*:* 同時確率は「クラス全員」が分母ですが、条件付き確率は「メガネの人」が分母になります。ベイズ統計では、データが得られるたびにこの「分母（前提となる世界）」が次々と書き換えられていくのです。

1.3 ベイズの定理：情報の更新

ベイズの定理を一言で言うと、**「新しい情報が入ったときに、もともとの予想をどれくらい修正すべきか」**を教えてくれる式です。

1.3.1 公式の形

高校数学の「条件付き確率の定義」から出発します。

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

これだけだと難しく見えますが、文字を次のように置き換えると一気に「意味」が見えてきます。

- H **(Hypothesis)**：予想・仮説（例：病気かな？）
- D **(Data)**：証拠・データ（例：検査の結果が陽性だった！）

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

この式は、**「証拠 D を見た後の、予想 H の正しさ」**を計算しています。

1.3.2 100 人の村で考える「ベイズの定理」

数式だけで理解するのは大変なので、**「100 人の村」**で具体的に考えてみましょう。

例題：スマホの故障診断

ある町で、スマホが急に動かなくなる「フリーズ病」が流行っているとします。

1. **事前予想**：この病気にかかっている人は、町全体の **10%** です。
2. **証拠の性質**：
   - 本当に病気なら、診断アプリは **90%** の確率で「異常あり」と出します。
   - 健康（病気でない）でも、機械のミスで **20%** の確率で「異常あり」と出てしまいます。

今、あなたのスマホで診断アプリが**「異常あり」**と出しました。本当に病気である確率は何%でしょうか？

ステップ解説：100 人の村でシミュレーション！

まず、100 人のスマホを想像します。

1. **病気の人と健康な人に分ける**
   - **病気 (10%)**：10 人
   - **健康 (90%)**：90 人
2. **「異常あり」と出る人を数える**
   - **病気の 10 人のうち**：90%が「異常あり」なので **9 人**
   - **健康な 90 人のうち**：20%がミスで「異常あり」なので **18 人**（$= 90 \times 0.2$）
   - **合計で「異常あり」と出た人**：$9 + 18 = \mathbf{27}$ **人**
3. **「異常あり」と出た人の中で、本当に病気なのは？**
   - 分母は「異常あり」と出た **27 人** 全員。
   - 分子はその中で本当に病気の **9 人**。

$$P(\text{病気} \mid \text{異常あり}) = \frac{9}{27} = \frac{1}{3} \approx \mathbf{33.3}\%$$

1.3.3 ベイズの定理が教えてくれること

計算の結果、アプリで「異常あり」と出ても、本当に病気である確率は **約 33%** しかないことがわかりました。

なぜでしょうか？それは、もともと「健康な人（90 人）」の数が圧倒的に多いため、たった 20% の「診断ミス」であっても、その人数（18 人）が、本物の病気の人（9 人）を上回ってしまうからです。

この例をベイズの言葉で整理すると：

- **事前確率** $P(H)$：データを見る前は「10%くらいかな」と思っていた。
- **尤度** $P(D|H)$：病気なら「異常あり」と出やすい。
- **事後確率** $P(H|D)$：結果を見て「33%まで上がったぞ！」と更新した。

注意 1.2 (高校生へのメッセージ)**.** ベイズの定理は、**「思い込み（事前確率）」**を**「事実（データ）」**で修正するプロセスです。「検査が陽性＝絶対病気だ！」とパニックにならず、もともとの流行具合（事前確率）を考慮して冷静に判断するための、とても科学的で賢い考え方なのです。

1.4 ベイズの定理の覚え方：3つのアプローチ

ベイズの定理は、複雑な分数として覚えるよりも、**「情報の流れ」**や**「言葉のセット」**でイメージ化すると、一生忘れない知識になります。高校生や初心者に教える際にも非常に効果的な、3つの覚え方を紹介します。

1.4.1 1.「逆さまの定理」として覚える

ベイズの定理の最大の役割は、**「原因と結果の入れ替え」**です。

- **知りたいこと**: 結果（データ）から原因（仮説）を当てる確率 $\rightarrow P(\text{原因} \mid \text{結果})$
- **わかっていること**: 原因があったときにその結果が出る確率 $\rightarrow P(\text{結果} \mid \text{原因})$

これを入れ替えるために、**「もともとの原因の確率」で重み付けして、「結果が起きる確率」で割る**、とイメージします。

1.4.2 2.「事後 ＝ 尤度 × 事前」という呪文で覚える

数式を日本語の単語に置き換えて、リズムで覚える方法です。分母（正規化定数）を一旦無視した、この**比例関係**がベイズの正体です。

$$\text{事後確率} \propto \text{尤度} \times \text{事前確率}$$

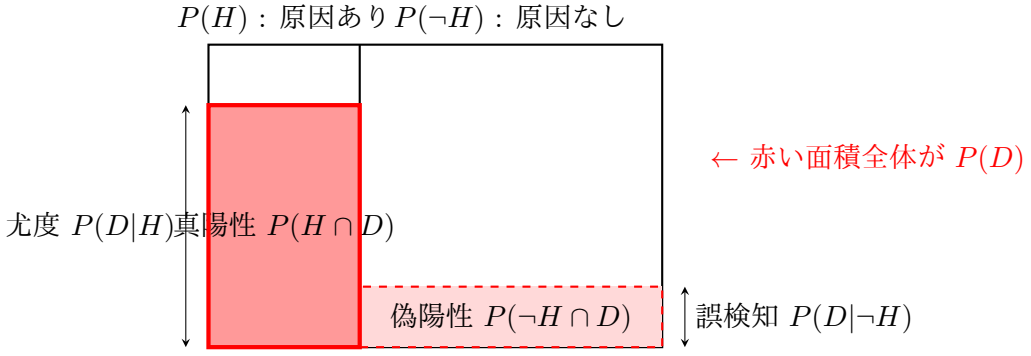
- **事前（Before）**：データを見る前の、自分の「予想・思い込み」。
- **尤度（Evidence）**：そのデータが、自分の予想にどれくらい「もっともらしい」か。
- **事後（After）**：データを見た後の、修正された「新しい予想」。

  覚え方イメージ：「アップデート（事後）したければ、思い込み（事前）に証拠（尤度）を掛け算しろ」と覚えます。

1.4.3 3.「面積図（ベイズ・ボックス）」で視覚化する

式を忘れても、この図を書ければ自分で導出できます。

Figure 1.1: **面積で考えるベイズの定理。**全面積のうち色が付いた部分（赤）が証拠 D の発生を表す。D が起きた世界（赤枠の中）において、「濃い赤（H）」が占める割合が事後確率となる。



1. まず、横幅を**「事前確率」**（例：病気 vs 健康）に分けた長方形を書きます。
2. 次に、それぞれの高さに**「尤度」**（例：陽性が出る確率）を書き込みます。
3. すると、面積が**「同時に起きる確率」**になります。
4. **「証拠（陽性）」が得られたら、陽性の面積だけの世界にワープします。** その「陽性の世界」の中で、お目当ての面積が占める割合を計算するだけです。

まとめ：試験でパッと書くための「型」

答案用紙の隅に、いつもこの「型」を書くようにしましょう。

$$P(H|D) = \frac{P(D|H) \cdot P(H)}{P(D)}$$

- **分子**： P(知りたいことの逆) $\times$ P(知りたいこと単独)
- **分母**： 分子のパターンを全部足したもの

「ベイズは『**切り替わった後の分母**』を探すゲームだ」と覚えると、公式の分母の意味もスッと入ってきます。

定理 1.3 (ベイズの定理（厳密形）)**.** 確率空間 $(\Theta \times \mathcal{Y}, \mathcal{F}, P)$ 上で、事象 $Y \in \mathcal{F}, P(Y) > 0$ に対し、

$$P(\Theta \in A|Y) = \frac{P(Y|\Theta \in A)P(\Theta \in A)}{P(Y)}, \quad \forall A \in \mathcal{F}$$

Proof. 条件付き確率の定義より、同時確率は次のように分解できます。

$$P(\Theta \in A, Y) = P(Y|\Theta \in A)P(\Theta \in A)$$

また、

$$P(\Theta \in A, Y) = P(\Theta \in A|Y)P(Y)$$

これらを等しく置くと、

$$P(\Theta \in A|Y)P(Y) = P(Y|\Theta \in A)P(\Theta \in A)$$

$P(Y) > 0$ より、両辺を $P(Y)$ で割ることで定理が得られます。 □

注意 1.4 (数理的補足：測度論的ベイズの定義)**.** より厳密には、ベイズの定理は**条件付き期待値**として定義されます。可測空間 $(\Theta, \mathcal{B}_\Theta)$ 上の事前尺度 π と、遷移核（尤度）$p(y|\theta)$ に対し、事後分布 $p(d\theta|y)$ は以下の *Radon-Nikodym* 微分として一意に存在します。

$$\frac{d\pi(\theta|y)}{d\pi(\theta)} = \frac{p(y|\theta)}{\int_\Theta p(y|\theta')\pi(d\theta')}$$

これにより、離散・連続が混在するモデルや、無限次元空間（ガウス過程など）での推論に厳密な数学的基礎が与えられます。

補題 1.5 (比例形)**.** $P(Y)$ は Θ に依存しない定数であるため、次のように記述できます。

$$P(\theta|D) \propto P(D|\theta)P(\theta) \quad \text{(事後確率} \propto \text{尤度} \times \text{事前確率)}$$

1.5 例題：PCR 検査のパラドックス

例 1.6. あるウイルスに感染している確率は、全人口の *0.1%* であるとする（事前確率 $P(I) = 0.001$）。検査キットの性能：

- 感度 $P(+|I) = 0.99$
- 特異度 $P(-|I^c) = 0.99 \Rightarrow P(+|I^c) = 0.01$

検査で陽性（+）が出た場合、実際に感染している確率 $P(I|+)$ は？

解答: ベイズの定理より、

$$P(I|+) = \frac{P(+|I)P(I)}{P(+)}$$

周辺尤度 $P(+)$ は全確率の法則より、

$$\begin{aligned} P(+) &= P(+|I)P(I) + P(+|I^c)P(I^c) \\ &= (0.99 \times 0.001) + (0.01 \times 0.999) \\ &= 0.00099 + 0.00999 = 0.01098 \end{aligned}$$

よって、

$$P(I|+) = \frac{0.00099}{0.01098} \approx 0.0902$$

つまり、陽性でも感染確率は約 9%です。偽陽性の影響が大きいためです。

1.6 練習問題

1. **工場の不良品検出**: 機械 A（シェア 60%、不良率 1%）と機械 B（シェア 40%、不良率 2%）がある。不良品が見つかったとき、それが機械 A 製である確率は？

2. **モンティ・ホール問題**: 3 つのドアがあり、1 つが当たり。A を選び、司会者がハズレの B を開けた。C に変えるべきか？

略解

1. **42.9%**: $P(A|D) = \frac{0.01 \times 0.6}{0.01 \times 0.6 + 0.02 \times 0.4} = \frac{0.006}{0.014} \approx 0.4286$。

2. **変えるべき**: $P(A|OpenB) = 1/3$、$P(C|OpenB) = 2/3$。

Chapter 2

ベイズ推論の仕組み

第 1 章で学んだベイズ定理を「推論エンジン」として使いこなすために、各構成要素（事前分布・尤度・事後分布）の役割と、それらが織りなす「知識更新プロセス」を理解します。

2.1　ベイズ推論の 3 要素：直感的なイメージ

ベイズ推論は、**「新しい情報を使って、これまでの思い込みをアップデートするプロセス」**です。これを、プロが「料理」を作るときの感覚に例えてみましょう。

2.1.1　事前分布 (Prior Distribution)：$P(\theta)$

- **直感**: 料理を作る前の**「職人の勘」**や**「これまでの経験」**。
- **高校生向け解説**: データを手に入れる前に、あなたが「多分こうだろうな」と思っている予想の分布です。
  - 例：「このお店のカレーは、10 人中 8 人は『美味しい』と言うはずだ」という事前の自信。

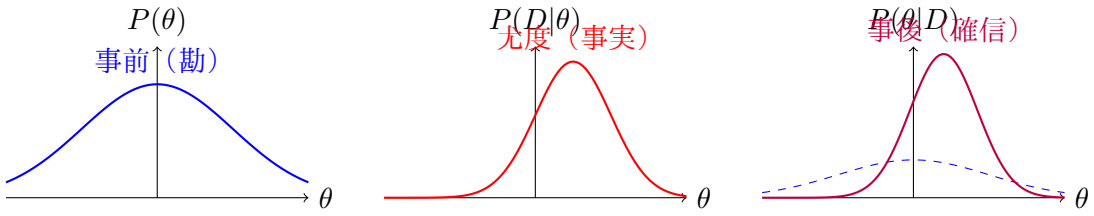
2.1.2　尤度関数 (Likelihood Function)：$P(D|\theta)$

- **直感**: 目の前で起きた**「事実」**が、その予想に対してどれくらい**「もっともらしいか」**。
- **高校生向け解説**: 「もし自分の予想（θ）が正しかったとしたら、今目の前にあるデータ（D）がどれくらいの確率で発生するか」を表す指標です。
  - 例：「もしカレーが本当に美味しいなら、目の前の客が『完食した』という事実は、非常に納得がいく（尤度が高い）」

2.1.3　事後分布 (Posterior Distribution)：$P(\theta|D)$

- **直感**: 食べた人の反応を見た後の**「新しい確信」**。
- **高校生向け解説**: 「事前予想」と「目の前の事実（尤度）」を掛け合わせて導き出した、**アップデート後の予想**です。
  - 例：客が笑顔で完食したのを見て、「やっぱりこのカレーは美味しいんだ！」と、自信がさらに深まった状態。

Figure 2.1: ベイズ推論による知識のアップデート。左：事前分布（職人の勘）、中：尤度（客の反応）、右：事後分布（新しい確信）。



(a) Prior (b) Likelihood (c) Posterior

2.1.4 比例関係の覚え方

以下の数式は、ベイズ統計で最も重要な「重み付け」の式です。

$$P(\theta|D) \propto P(D|\theta) \times P(\theta)$$

「事後の確信 ＝ 証拠の強さ × もともとの予想」

ポイント: 分母の $P(D)$ は、単に全体の合計を 1（100%）に調整するための定数なので、ひとまず「掛け算で形が決まる」ことだけを意識すれば OK です。

注意 2.1 (コラム：なぜ「分布」なの？)**.** 高校数学では「確率はひとつの数字（例：*0.5*）」として扱いますが、ベイズ統計では**「分布（山の形）」**として扱います。

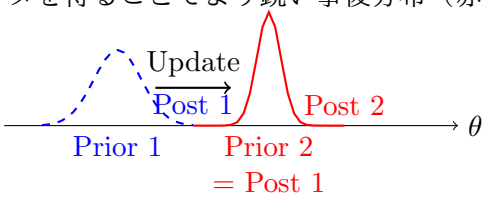
- **数字ひとつ**: 「表が出る確率は *50%*だ」と言い切る。
- **分布**: 「*50%*くらいだと思うけど、もしかしたら *52%*かもしれないし、*48%*かもしれない」という**不確実な気持ちの揺らぎ**を、山の広がりとして表現できるからです。

2.2 数理的な定義

2.3 逐次的更新 (Sequential Update)

ベイズ推論の真価は「連続更新」にあります。新たなデータ D_2 が入ったとき、前回の事後分布が次回の事前分布として機能します。これは、ベイズの定理と**条件付き独立性の仮定**から数学的に導かれます。

Figure 2.2: **ベイズ更新の連鎖**。最初のデータによる事後分布（青）が、次の推論における事前分布として機能し、新たなデータを得ることでより鋭い事後分布（赤）へと進化していく。



2.3.1 逐次的更新の数理的証明

定理 2.2 (逐次更新則)**.** データ D_1 を得た後の事後分布を $P(\theta|D_1)$ とするとき、さらに新しいデータ D_2 を得た後の事後分布 $P(\theta|D_1, D_2)$ は、次のように表される。

$$P(\theta|D_1, D_2) \propto P(D_2|\theta) \times P(\theta|D_1)$$

すなわち、「新しい事後分布 $\propto$ 新しい尤度 $\times$ 前回の事後分布」である。

Proof. パラメータ θ が与えられたとき、各データ D_1, D_2 は互いに独立に発生すると仮定する（条件付き独立性）。

$$P(D_1, D_2|\theta) = P(D_1|\theta) \cdot P(D_2|\theta)$$

1. 2 つのデータに対するベイズの定理を適用する：

$$P(\theta|D_1, D_2) = \frac{P(D_1, D_2|\theta)P(\theta)}{P(D_1, D_2)}$$

2. 分子に条件付き独立性の仮定を代入する：

$$P(\theta|D_1, D_2) \propto P(D_2|\theta) \cdot P(D_1|\theta)P(\theta)$$

（分母は θ に関与しない定数として比例記号 $\propto$ で処理）

3. ここで、1 つ目のデータ D_1 に関するベイズの定理に注目する：

$$P(\theta|D_1) \propto P(D_1|\theta)P(\theta)$$

この右辺は、手順 2 の式の後半部分と一致する。

4. これを代入すると、以下の逐次更新式が得られる：

$$P(\theta|D_1, D_2) \propto P(D_2|\theta) \cdot P(\theta|D_1)$$

この結果は、前回の事後分布 $P(\theta|D_1)$ が、次の推論における事前分布の役割を果たしていることを示している。（証明終） □

注意 2.3 (数理的な意義：マルコフ性)**.** この性質により、ベイズ推論では過去の膨大な生データ（D_1）をすべて保持しておく必要がありません。過去の情報をすべて凝縮した**「現在の事後分布」**さえ持っていれば、新しいデータが来るたびに、常に最新の知識状態にアップデートできるという、計算上の大きな利点（オンライン学習の正当性）を保証しています。

注意 2.4 (高校生向けの直感説明)**.** 「昨日までの経験（D_1 を反映した事後分布）」をベースに、「今日の出来事（新しい尤度 D_2）」を解釈して、「明日への新しい確信（最新の事後分布）」を作る。ベイズ統計では、**昨日の結論が今日の出発点になる**、という非常に自然な思考プロセスを数式で表現しているだけなのです。

2.4 例題：コインの偏り推定

例 2.5. 公正なコイン（$\theta = 0.5$）か偏ったコイン（$\theta = 0.8$）かを推定します。

- 事前確率：$P(\theta = 0.5) = 0.5, P(\theta = 0.8) = 0.5$
- データ：*10* 回投げて表が *7* 回 *(D)*

計算: 尤度（二項分布）：

$$P(D|\theta = 0.5) = \binom{10}{7}(0.5)^{10} \approx 0.1172$$

$$P(D|\theta = 0.8) = \binom{10}{7}(0.8)^7(0.2)^3 \approx 0.2013$$

事後確率（未正規化）：

$$P(\theta = 0.5|D) \propto 0.1172 \times 0.5 = 0.0586$$

$$P(\theta = 0.8|D) \propto 0.2013 \times 0.5 = 0.1007$$

正規化して、

$$P(\theta = 0.8|D) = \frac{0.1007}{0.0586 + 0.1007} \approx 0.632$$

データによって「偏ったコイン」である確率が 50%から 63.2%に上昇しました。

2.5 練習問題

1. **テストの採点バイアス**: 甘い教師（平均 85 点）と厳しい教師（平均 70 点）。クラス平均 82 点の場合の事後確率は？
2. **ラジオアクティブ崩壊**: 事前分布 $\lambda \sim \mathrm{Exp}(1)$。5 分で 3 回崩壊を観測した後の事後分布は？

略解

1. **約 57%**：甘い教師である確率。
2. **Gamma(4, 6)**：平均 $\lambda = 2/3$。

Chapter 3

共役事前分布による解析的推論

多くの実データでは事後分布が解析的に求められませんが、**「共役事前分布（Conjugate Prior）」**を選べば、閉じた形式で事後分布が得られます。

3.1　共役事前分布：計算を「ズル」するための最高のペア

ベイズの定理の計算において、一番の悩みどころは「分数の計算（積分）」が複雑すぎて答えが出ないことです。しかし、尤度に対して**「ある特定の形」**をした事前分布を組み合わせると、魔法のように計算が簡単になります。

3.1.1　直感的なイメージ：テンプレートの継承

共役事前分布とは、**「データを反映しても、山の種類が変わらない分布」**のことです。

- イメージ例:
  - **粘土で作った「ピラミッド」**（事前分布）に、新しい**粘土の塊**（データ）を付け足したとき、
  - 普通は形が崩れてしまいますが、共役な関係だと、**「一回り大きい、より鋭いピラミッド」**（事後分布）に進化するだけです。
- メリット: 「計算（積分）」をしなくても、**パラメータの数字を書き換える（足し算する）だけ**で答えに辿り着けます。

3.1.2　具体例：「アンケート結果」のアップデート

成功確率 θ を推定したい状況を考えます。

1. **尤度（データの形）**: 「成功か失敗か」のデータ（**二項分布**）。
2. **共役事前分布**: ここで **ベータ分布** という「型」を事前分布に選びます。
3. **魔法の更新**:
   - もともと「成功 α 回、失敗 β 回くらいだろう」というベータ分布（型）を持っていたとします。
   - 実際にアンケートをとって「成功 y 回、失敗 $n-y$ 回」というデータが得られました。
   - **結果**: 事後分布は、**「成功 $\alpha+y$ 回、失敗 $\beta+n-y$ 回」という新しいベータ分布（型）**になります。

つまり！分布の種類（ベータ分布）はずっと変わらず、**「中の数字を足し算しただけ」**でアップデートが完了しました。これが「共役（相性が良い）」と言われる理由です。

3.1.3 なぜ「共役（Conjugate）」と呼ぶのか？

「共役」という言葉には「同じ役目を持つ」「ペアになる」という意味があります。

- **数理的な理由**: 尤度の数式の中にある「変数の位置」と、事前分布の数式の中にある「パラメータの位置」が鏡のように対応しているため、掛け算したときに見事にひとつの式にまとまってしまうのです。

代表的なペアの例

知りたいこと（尤度）	相性抜群の事前分布（共役事前）	アップデートの仕組み
成功率（二項分布）	ベータ分布	成功数・失敗数を足すだけ
発生件数（ポアソン分布）	ガンマ分布	発生回数・期間を足すだけ
平均値（正規分布）	正規分布	平均値を重み付けして混ぜるだけ

注意 3.1 (まとめ)**.** 共役事前分布とは、**「アップデートしても分布の『種類』が維持される、計算の手間を劇的に減らしてくれる魔法のテンプレート」**のことです。

3.2 主要な共役モデルの具体的イメージ

これらの定理は、一見複雑な積分の計算を**「単なる足し算」**に落とし込んでくれる魔法の道具です。それぞれのペアが何を意味しているのかを具体例で見てみましょう。

3.2.1 二項分布（成功率）× ベータ分布

イメージ：アンケート調査の確信度

- **状況**: 新商品の「支持率（成功率）」を知りたい。
- **事前（ベータ分布）**: 調査前にあなたが「これまでの経験上、10 人中 2 人（20%）くらいは支持してくれるだろう」と思っている確信度。
  - このとき、$\alpha = 2, \beta = 8$ とセットします。
- **データ（二項分布）**: 実際に 100 人に調査したら、30 人が支持した（$y = 30$）。
- **事後**: あなたの新しい確信度は、**単純に支持した人数と支持しなかった人数を足すだけ**で更新されます。
  - 新 $\alpha = 2 + 30$, 新 $\beta = 8 + 70$
- **直感**: 自分の頭の中にある「過去の成功・失敗数」に、「新しい成功・失敗数」をポンと放り込んで混ぜるだけの感覚です。

3.2.2 ポアソン分布（発生率）× ガンマ分布

イメージ：お店への「お客さんの来店ペース」

- **状況**: 「1 時間に平均何人（λ）のお客さんが来るか」を推定したい。
- **事前（ガンマ分布）**: 「だいたい 1 時間に 5 人（$\lambda = 5$）くらいだろう」というこれまでの予測。
  - ここでは「過去の合計人数 α」と「過去の合計観測時間 β」として持っています。

- **データ（ポアソン分布）**：今日、3 時間（$t = 3$）お店を開けたら、合計 18 人（$y = 18$）来た。
- **事後**: 新しいペースの予測は、**「合計人数」と「合計時間」をそれぞれ足すだけ。**
  - 新 $\alpha = \alpha + 18$, 新 $\beta = \beta + 3$
- **直感**: 「これまで見てきた総数」と「これまで待った総時間」の記録ノートに、今日の分を 1 行書き足して合計し直すイメージです。

3.2.3 正規分布（平均値）× 正規分布

イメージ：テストの「クラス平均」の推測

- **状況**: 「この学校全体のテストの平均点（μ）」を当てたい。
- **事前（正規分布）**：「多分 70 点くらいかな（$\mu_0 = 70$）」という事前の予想。
- **データ（正規分布）**：クラス 30 人の平均点を計算したら、75 点だった（$\bar{y} = 75$）。
- **事後**:
  - もし自分の事前予想に自信がある（分散 τ^2 が小さい）なら、平均点は 70 点に近いまま。
  - もしデータの人数が多い（n が大きい）なら、平均点は 75 点にぐっと引き寄せられます。
- **直感**: **「自分の予想」と「実際のデータ」による綱引き**です。データの数（信頼度）が多ければ多いほど、結果はデータの方へと引っ張られていきます。

注意 3.2 (まとめ：なぜこのペアが選ばれるのか？)**.** ベイズの計算において、これらのペアが「最強」なのは、**「計算結果が元のテンプレートと同じ形に戻ってくるから」**です。

1. **二項分布**：成功と失敗の「回数」が重要 → **ベータ分布**
2. **ポアソン分布**：起きる「回数」と「時間」が重要 → **ガンマ分布**
3. **正規分布**：中心の「平均」と「ブレ」が重要 → **正規分布**

このように、データの「性質」に合った分布を事前分布に選ぶことで、難しい積分を一切せずに、小学生でもできる**「足し算」や「重み付け」だけで世界をアップデートできる**ようになるのです。

3.3 二項分布とベータ分布のアナロジー：「確信の貯金箱」

ベイズ更新における二項分布とベータ分布の関係は、**「実際に起きた出来事」**を**「頭の中の経験ノート」**に書き写す作業に例えられます。

3.3.1 1. 登場人物の役割

- **二項分布（データ）**：**「今日の結果」**
  - 例：今日シュートを 10 回打って、3 回入った。
- **ベータ分布（事前・事後分布）**：**「これまでの自信（経験ノート）」**
  - 例：「俺はだいたい 10 回中 5 回は入る選手だ」という自分の実力に対する確信。

3.3.2 2. アナロジー：魔法の貯金箱

あなたの頭の中に、**「成功コイン」**と**「失敗コイン」**を入れる 2 つの貯金箱があると想像してください。この貯金箱の合計が、あなたの「実力に対する確信」を表します。

① 事前分布：これまでの経験（貯金箱の初期状態）

あなたは自分の実力を「五分五分（50%）」だと思っています。これをベータ分布で表現すると、貯金箱に最初から以下のコインが入っている状態です。

- **成功の貯金箱 (α)**：10 枚（過去の成功体験）
- **失敗の貯金箱 (β)**：10 枚（過去の失敗体験）

  直感: 10 勝 10 敗の成績表を持っているようなものです。

② 二項分布：新しいデータ（今日の試合）

今日、実際に試合をしました（10 回試行）。結果は **「成功 3 回、失敗 7 回」** でした。これが「二項分布」というデータです。

③ 事後分布：確信のアップデート（コインを放り込む！）

ベイズの定理が行うのは、**「今日の結果を、そのまま貯金箱に放り込む」**という作業だけです。

- 成功の貯金箱に、今日の成功 **3 枚** を追加 $\rightarrow \alpha + y = 10 + 3 = 13$
- 失敗の貯金箱に、今日の失敗 **7 枚** を追加 $\rightarrow \beta + (n - y) = 10 + 7 = 17$

④ 結果：新しい自分の見積もり

貯金箱の中身は **「成功 13 枚 vs 失敗 17 枚」** になりました。

- **更新前**: $10/(10 + 10) = 50\%$ の自信
- **更新後**: $13/(13 + 17) \approx 43\%$ の自信

  直感:「あれ、俺、意外とシュート下手かも…？」と、自信の山の中心が少し左（低い方）へズレました。

3.3.3 高校生に伝えるポイント

1. **「足し算」だけでいい理由**: ベータ分布の式 $\theta^{\alpha-1}(1-\theta)^{\beta-1}$ と二項分布の式 $\theta^{y}(1-\theta)^{n-y}$ は、形がそっくりです。掛け算すると、指数の部分が **「ただの足し算」** になります。だから貯金箱にコインを放り込むだけでアップデートが終わるのです。
2. **経験が多い人（頑固な人）**: もし貯金箱に最初から各 1000 枚ずつコインが入っていたら、今日 3 勝 7 敗したところで、割合はほとんど変わりません。**「ベテランはちょっとした結果で自信を失わない」**という現象も、この足し算で説明できます。
3. **何の先入観もない時**: 貯金箱を空（各 1 枚ずつ）にすれば、今日の結果がそのまま自分の自信になります。これを「無情報事前分布」と呼びます。

注意 3.3 (まとめ)**.** ベイズ更新（二項 × ベータ）とは、**「過去の成功・失敗回数」という貯金箱に、「今日の成功・失敗回数」をチャリンと足すだけの、とてもシンプルな作業**なのです。

3.4 共役事前の定義（数理）

定義 3.4 (共役事前分布)**.** 尤度 $P(D|\theta)$ と事前分布 $P(\theta)$ の積が、事前分布と同じ分布族（パラメータのみ更新されたもの）になる場合、その事前分布を共役事前分布と呼びます。

3.5 主要な共役モデルと定理

3.5.1 二項分布モデル：ベータ分布

定理 3.5 (二項-ベータ共役性)**.** データ $y \sim Binomial(n, \theta)$、事前分布 $\theta \sim Beta(\alpha, \beta)$ に対し、事後分布は次になります。

$$\theta|y \sim Beta(\alpha + y, \beta + n - y)$$

Proof. ベイズの定理より、

$$\begin{aligned} P(\theta|y) &\propto P(y|\theta)P(\theta) \\ &\propto \theta^y(1-\theta)^{n-y} \times \theta^{\alpha-1}(1-\theta)^{\beta-1} \\ &= \theta^{(\alpha+y)-1}(1-\theta)^{(\beta+n-y)-1} \end{aligned}$$

これはパラメータ $\alpha + y, \beta + n - y$ のベータ分布の核（kernel）と一致します。 □

3.5.2 ポアソン分布モデル：ガンマ分布

定理 3.6 (ポアソン-ガンマ共役性)**.** データ $y \sim Poisson(\lambda t)$、事前分布 $\lambda \sim Gamma(\alpha, \beta)$ に対し、事後分布は次になります。

$$\lambda|y \sim Gamma(\alpha + y, \beta + t)$$

Proof.

$$\begin{aligned} P(\lambda|y) &\propto P(y|\lambda)P(\lambda) \\ &\propto (\lambda t)^y e^{-\lambda t} \times \lambda^{\alpha-1} e^{-\beta\lambda} \\ &\propto \lambda^{(\alpha+y)-1} e^{-(\beta+t)\lambda} \end{aligned}$$

これはパラメータ $\alpha + y, \beta + t$ のガンマ分布の核と一致します。 □

3.5.3 正規分布モデル：正規分布（分散既知）

定理 3.7 (正規-正規共役性)**.** データ $y_1, \ldots, y_n \sim N(\mu, \sigma^2)$（$\sigma^2$ 既知）、事前分布 $\mu \sim N(\mu_0, \tau^2)$ に対し、事後分布 $\mu|\mathbf{y}$ は正規分布となり、その平均と分散は以下で与えられます。

$$\mu_{post} = \frac{\frac{n}{\sigma^2}\bar{y} + \frac{1}{\tau^2}\mu_0}{\frac{n}{\sigma^2} + \frac{1}{\tau^2}}, \quad \sigma^2_{post} = \frac{1}{\frac{n}{\sigma^2} + \frac{1}{\tau^2}}$$

Proof. 事後分布の指数部分は、

$$-\frac{1}{2}\left[\sum_{i=1}^{n}\frac{(y_i-\mu)^2}{\sigma^2} + \frac{(\mu-\mu_0)^2}{\tau^2}\right]$$

これを μ について整理（平方完成）すると、上記の平均・分散を持つ二次形式が得られます。詳細：

$$\sum \frac{(y_i-\mu)^2}{\sigma^2} = \frac{n(\mu-\bar{y})^2 + S}{\sigma^2}$$

として、μ の 2 次の項と 1 次の項の係数を比較することで導出されます。 □

3.6 練習問題

1. **選挙予測**: 支持率 $\theta \sim \mathrm{Beta}(100, 200)$。データ 1000 人中 350 支持。事後分布と 95%信用区間は？
2. **故障率**: $\lambda \sim \mathrm{Gamma}(2, 10)$。100 時間で 1 故障。事後期待値は？

Part II

統計モデリングと推定アルゴリズム

Chapter 4

MCMC（マルコフ連鎖モンテカルロ法）

共役事前分布が使えない複雑なモデルに対し、事後分布から直接サンプリングを行う MCMC を学びます。

4.1 MCMC の動機：なぜ「くじ引き」が必要なのか？

「難しい積分を計算する代わりに、サンプリング（くじ引き）で解決する」という MCMC の考え方を、**「複雑な形の池の面積を、雨粒で測る」**という例題を通して、自然に導き出してみましょう。

4.1.1 変な形の池の「平均的な深さ」を知りたい

あなたは、地図上で非常に複雑な形をした池の「平均的な水深」を調べようとしています。

1. **理想（積分）**：池のすべての地点の深さを足し合わせて、池の面積で割る。
2. **現実の壁**: 池の縁（ふち）が複雑すぎて、面積（正規化定数 $P(D)$）が計算できません。深さのデータ（尤度 $\times$ 事前分布）は各地点で測れますが、全体の「平均」を出すための分母がわからないのです。

4.1.2 ステップ 1：数式で解けないなら「雨」に任せる

面積が計算できないなら、空から池に向かってランダムに雨を降らせてみましょう。

- もし、**「池の深さに比例して、雨粒が溜まりやすい」**という不思議な性質の雨を降らせることができたらどうなるでしょうか？
- 雨粒がたくさん溜まった場所は「深い場所（事後確率が高い場所）」であり、雨粒が少ない場所は「浅い場所」になります。

この「雨粒」こそが、数式にある **サンプル** $\{\theta^{(1)}, \ldots, \theta^{(N)}\}$ です。

4.1.3 ステップ 2：雨粒の場所を記録する

池全体をくまなく調べる代わりに、池に落ちた N 個の雨粒の場所だけを記録します。この雨粒たちは、事後分布の「形」に従って散らばっています。

- 山が高い（確率が高い）ところには、雨粒が密集する。
- 山が低いところには、たまにしか落ちない。

4.1.4 ステップ 3：平均（期待値）を「ただの平均」で出す

さて、目標だった「池の平均的な深さ」を計算します。本来は難しい積分が必要でしたが、手元には「分布の形に従って落ちた雨粒の場所」が N 個あります。

それぞれの雨粒が落ちた場所の深さを $f(\theta^{(i)})$ とすると、

$$E[f(\theta)] \approx \frac{1}{N}\sum_{i=1}^{N} f(\theta^{(i)})$$

ただの算術平均（モンテカルロ積分）で、池全体の平均深さが近似できてしまいました！

注意 4.1 (ベイズにおける結論)**.** 事後分布の式において、分母 $P(D)$ は「池全体の面積」のようなものです。これがわからなくても、**「分子（分布の形）」に比例してサンプルを引く仕組み（*MCMC*）**さえ作れば、知りたい期待値や分散は、ただの平均計算で求められるようになるのです。

4.2 MCMC の基本思想（数理）

事後分布 $P(\theta|D) \propto P(D|\theta)P(\theta)$ の正規化定数 $P(D)$ が計算不能な場合、分布の「形」をサンプリングによって近似します。目標は、分布 $P(\theta|D)$ から N 個のサンプル $\{\theta^{(1)}, \ldots, \theta^{(N)}\}$ を抽出することです。これにより、期待値 $\mathbb{E}[f(\theta)]$ を $\frac{1}{N}\sum f(\theta^{(i)})$ で近似できます（モンテカルロ積分）。

4.3 メトロポリス・ヘイスティングス法（MH 法）

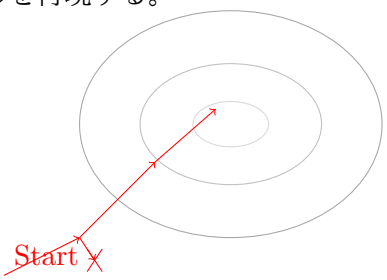
Algorithm 1 Metropolis-Hastings Algorithm

1: 初期値 $\theta^{(0)}$ を設定
2: **for** $t = 0$ to $T-1$ **do**
3: 提案分布 $q(\theta^*|\theta^{(t)})$ から候補 θ^* をサンプリング
4: 受理確率 α を計算：

$$\alpha = \min\left(1, \frac{P(D|\theta^*)P(\theta^*)q(\theta^{(t)}|\theta^*)}{P(D|\theta^{(t)})P(\theta^{(t)})q(\theta^*|\theta^{(t)})}\right)$$

5: $u \sim \text{Uniform}(0,1)$ を生成
6: **if** $u < \alpha$ **then**
7: $\theta^{(t+1)} = \theta^*$ （受理）
8: **else**
9: $\theta^{(t+1)} = \theta^{(t)}$ （棄却・現状維持）
10: **end if**
11: **end for**

Figure 4.1: **メトロポリス・ヘイスティングス法の軌跡。**初期値からスタートし、事後確率が高い中心領域（山の頂上）を目指してランダムに移動を繰り返す。一度分布の中心に到達すれば、その周囲を徘徊することで分布の形を再現する。



4.3.1 理論的保証：詳細的均衡と不変分布

この定理は、MCMC が「なぜ最終的に目標とする分布 $\pi(\theta)$ を再現できるのか」を支える最も重要な理論的根拠です。

1. 目標の設定

マルコフ連鎖において、ある分布 $\pi(\theta)$ が**不変分布（定常分布）**であるとは、現在の状態が π に従っているとき、次のステップの状態もまた π に従うことを指します。数理的には以下の積分方程式を満たすことです。

$$\int \pi(\theta)K(\theta'|\theta)d\theta = \pi(\theta')$$

2. 詳細的均衡条件（Detailed Balance Condition）

遷移核 $K(\theta'|\theta)$ が、任意の θ, θ' に対して以下の条件を満たすと仮定します。

$$\pi(\theta)K(\theta'|\theta) = \pi(\theta')K(\theta|\theta') \quad \dots(*)$$

この式は、**「θ から θ' へ移動する確率の流れ」と「θ' から θ へ戻る確率の流れ」が完全に釣り合っている**ことを意味します。

3. 証明のステップ

定理 4.2 (詳細的均衡と不変分布)**.** 遷移核 $K(\theta'|\theta)$ が詳細的均衡条件 $(*)$ を満たすならば、$\pi(\theta)$ はこのマルコフ連鎖の不変分布（定常分布）となる。

Proof. 不変分布の定義式（左辺）から出発し、詳細的均衡条件 $(*)$ を使って変形していきます。

ステップ A：詳細的均衡条件の代入

式 $(*)$ により、被積分関数 $\pi(\theta)K(\theta'|\theta)$ を $\pi(\theta')K(\theta|\theta')$ に置き換えます。

$$\int \pi(\theta)K(\theta'|\theta)d\theta = \int \pi(\theta')K(\theta|\theta')d\theta$$

ステップ B：定数の括り出し

積分は θ に関するものなので、θ' にしか依存しない $\pi(\theta')$ を積分の外に出すことができます。

$$= \pi(\theta') \int K(\theta|\theta')d\theta$$

ステップ C：確率の保存性質の利用

$K(\theta|\theta')$ は、状態 θ' から「どこか（θ）」へ移動する確率密度です。全空間にわたって積分すれば、その合計は必ず **1** になります。

$$\int K(\theta|\theta')d\theta = 1$$

ステップ D：結論

$$= \pi(\theta') \times 1 = \pi(\theta')$$

これにより、左辺＝右辺が示され、$\pi(\theta)$ が不変分布であることが証明されました。 □

4. 直感的なアナロジー：人口移動

2 つの都市 A と B の間で人が移動しているとします。「A から B へ行く人の数」と「B から A へ戻る人の数」が毎分同じであれば、両都市の人口（分布）はずっと変わりません。MH 法は、この**「移動の釣り合い」**を無理やり作り出すことで、私たちが知りたい分布 π を安定した状態（不変分布）として維持しているのです。

4.4 ギブスサンプリングの証明と具体例

ギブスサンプリングは、多変量の事後分布 $P(\theta_1, \theta_2, \ldots, \theta_K|D)$ から直接サンプルを引くのが難しいとき、**「他の変数を固定したときの条件付き分布」**から 1 つずつ順番にサンプリングしていく手法です。

4.4.1 1. 数理的証明：なぜ受理率が常に 1 なのか？

ギブスサンプリングは、メトロポリス・ヘイスティングス（MH）法の特殊なケースとして解釈できます。ここでは 2 変数 $\boldsymbol{\theta} = (\theta_1, \theta_2)$ の場合を考え、θ_1 を更新するステップを証明します。

目標: 提案分布 q として「条件付き分布」自体を使うとき、MH 法の受理率 α が 1 になることを示す。

1. **提案分布の設定**: 現在の状態を $\boldsymbol{\theta} = (\theta_1, \theta_2)$、新しい候補を $\boldsymbol{\theta}^* = (\theta_1^*, \theta_2)$ とします（θ_2 は固定）。このときの提案分布は、条件付き分布そのものです：
$$q(\boldsymbol{\theta}^*|\boldsymbol{\theta}) = P(\theta_1^*|\theta_2, D)$$
2. **MH 法の受理率の式**:
$$\alpha = \min\left(1, \frac{P(\boldsymbol{\theta}^*|D)q(\boldsymbol{\theta}|\boldsymbol{\theta}^*)}{P(\boldsymbol{\theta}|D)q(\boldsymbol{\theta}^*|\boldsymbol{\theta})}\right)$$
3. **同時確率を条件付き確率に分解**: 確率の乗法定理より $P(\theta_1, \theta_2|D) = P(\theta_1|\theta_2, D)P(\theta_2|D)$ なので：
$$\frac{P(\boldsymbol{\theta}^*|D)}{P(\boldsymbol{\theta}|D)} = \frac{P(\theta_1^*|\theta_2, D)P(\theta_2|D)}{P(\theta_1|\theta_2, D)P(\theta_2|D)} = \frac{P(\theta_1^*|\theta_2, D)}{P(\theta_1|\theta_2, D)}$$
4. **受理率の計算**: 手順 3 の結果と提案分布 q を代入すると：
$$\frac{P(\boldsymbol{\theta}^*|D)}{P(\boldsymbol{\theta}|D)} \cdot \frac{q(\boldsymbol{\theta}|\boldsymbol{\theta}^*)}{q(\boldsymbol{\theta}^*|\boldsymbol{\theta})} = \frac{P(\theta_1^*|\theta_2, D)}{P(\theta_1|\theta_2, D)} \cdot \frac{P(\theta_1|\theta_2, D)}{P(\theta_1^*|\theta_2, D)} = 1$$

結論: 比率が 1 になるため、$\alpha = \min(1, 1) = 1$ となります。つまり、**条件付き分布からサンプルを引く限り、その提案は 100% 受理される**のです。

4.4.2 2. 【例題】2 変量正規分布のサンプリング

相関のある 2 つの変数 (X, Y) が、平均 0、相関係数 ρ の正規分布に従うとします。

条件付き分布: 正規分布の性質より、片方を固定したときの分布は以下のようになります：

- $P(X|Y) = N(\rho Y, 1 - \rho^2)$
- $P(Y|X) = N(\rho X, 1 - \rho^2)$

ギブスサンプリングの手順:

1. 適当な初期値 $Y^{(0)}$ を決める。
2. $X^{(1)}$ を $N(\rho Y^{(0)}, 1 - \rho^2)$ から引く。
3. $Y^{(1)}$ を $N(\rho X^{(1)}, 1 - \rho^2)$ から引く。
4. これを繰り返す。

直感的な動き:

- X と Y に正の相関（$\rho > 0$）がある場合、Y が大きい値なら、次に引かれる X も大きい値になりやすくなります。
- ギブスサンプラーは、このように**「お互いの値をヒントにしながら」**、一歩ずつ分布の全体像（楕円形の山）を探索していくのです。

直感的なアナロジー：暗闇の山歩き

霧が深くて足元しか見えない山で、最高地点を探しているとします。

- **ギブス**: 「今は東にしか動かない」「次は北にしか動かない」と、軸に沿って交互に進みます。
- 各ステップで「その方向で一番もっともらしい場所」へ移動し続けることで、結果的に山全体の形を把握できる、という賢い戦略です。

4.5 実装例：ロジスティック回帰（MH 法）

```python
import numpy as np

def log_posterior(beta, X, y):
    # 対数尤度 + 対数事前分布
    eta = X @ beta
    p = 1 / (1 + np.exp(-eta))
    ll = np.sum(y * np.log(p) + (1-y) * np.log(1-p)) # Bernoulli
    prior = -0.5 * np.sum(beta**2 / 10) # Normal(0, 10)
    return ll + prior

# Simple MH Step
def mh_step(current_beta, X, y):
    proposal = current_beta + np.random.normal(0, 0.5, size=len(
        current_beta))
    accept_ratio = np.exp(log_posterior(proposal, X, y) - log_posterior(
        current_beta, X, y))
    if np.random.rand() < accept_ratio:
        return proposal
    return current_beta
```

4.6 収束診断：Gelman-Rubin 統計量 $\hat{R}$

4.6.1 定理 4.3：Gelman-Rubin 統計量 $\hat{R}$ の理論と図解

MCMC が十分に収束したかを判断するために、**「異なる初期値からスタートした複数のチェーンが、同じ分布を再現しているか」**を分散分析（ANOVA）の考え方で評価します。

1. 各統計量の数理的定義

サンプルの個数を n、チェーンの数を m とします。各チェーン j におけるサンプルの平均を $\bar{\theta}_j$、全体の平均を $\bar{\bar{\theta}}$ とします。

1. **チェーン内分散 (W: Within-chain variance)**: 各チェーンの中でのバラツキの平均です。
$$W = \frac{1}{m}\sum_{j=1}^{m} s_j^2, \quad s_j^2 = \frac{1}{n-1}\sum_{i=1}^{n}(\theta_{ij} - \bar{\theta}_j)^2$$
2. **チェーン間分散 (B: Between-chain variance)**: チェーンごとの平均値がどれだけバラついているかを示します。
$$\frac{B}{n} = \frac{1}{m-1}\sum_{j=1}^{m}(\bar{\theta}_j - \bar{\bar{\theta}})^2$$
3. **事後分散の推定値 ($\hat{V}$)**: 全サンプルのバラツキを、W と B の加重平均で推定します。
$$\hat{V} = \frac{n-1}{n}W + \frac{1}{n}B$$

2. $\hat{R}$ **が収束を示す理由（証明の考え方）**

定理 4.3 (Gelman-Rubin 統計量 $\hat{R}$)**.**

$$\hat{R} = \sqrt{\frac{\hat{V}}{W}}$$

$\hat{R} < 1.1$ であれば収束したとみなします。

収束していないとき: 各チェーンがまだ分布の一部しか探索できていないため、チェーンごとの平均値 $\bar{\theta}_j$ は大きく離れます。その結果、B が大きくなり、$\hat{V}$ は W よりもかなり大きな値になります。

$$\hat{R} = \sqrt{\frac{\hat{V}}{W}} \gg 1$$

収束したとき: 全てのチェーンが同じ分布（ターゲット分布）全体を正しく探索していれば、各チェーンの平均値はほぼ一致し、B は非常に小さくなります。また、n が十分に大きければ $\frac{n-1}{n} \approx 1$ となり、$\hat{V} \approx W$ となります。

$$\hat{R} = \sqrt{\frac{\hat{V}}{W}} \to 1$$

3. TikZ による収束イメージ図

MCMC の収束（$\hat{R} \approx 1$）と未収束（$\hat{R} > 1.1$）を視覚化します。

未収束 ($\hat{R} > 1.1$) 収束 ($\hat{R} \approx 1.0$)

Figure 4.2: **MCMC の収束イメージ**：左図は 2 つのチェーンが異なる領域に留まっており、チェーン間分散 B が大きいため $\hat{R}$ が高い値を示します。右図はチェーンが互いに入り混じり（Mixing）、同じ分布を探索しているため $\hat{R}$ が 1 に近づいた状態です。

Chapter 5

変分推論（Variational Inference）

MCMC は漸近的に正確ですが計算コストが高いです。変分推論（VI）は、推論を「最適化問題」に帰着させることで高速化を図ります。

5.1 VI の基本思想

真の事後分布 $P(\theta|D)$ を、扱いやすい近似分布ファミリー $q(\theta;\phi)$ の中から最も近い分布で近似します。「近さ」の指標として、KL ダイバージェンス（Kullback-Leibler Divergence）を用います。

$$\phi^* = \arg\min_{\phi} \mathrm{KL}(q(\theta;\phi)||P(\theta|D))$$

5.2 Evidence Lower Bound (ELBO)

KL ダイバージェンスの直接最小化は $P(\theta|D)$ が未知のため困難ですが、等価な問題として ELBO の最大化を解きます。

定理 5.1 (ELBO による下界性)**.** 任意の分布 $q(\theta)$ に対し、周辺尤度の対数は以下のように分解できます。

$$\log P(D) = \underbrace{\mathbb{E}_q[\log P(D,\theta) - \log q(\theta)]}_{ELBO(q)} + KL(q||P(\theta|D))$$

$KL \geq 0$ より、

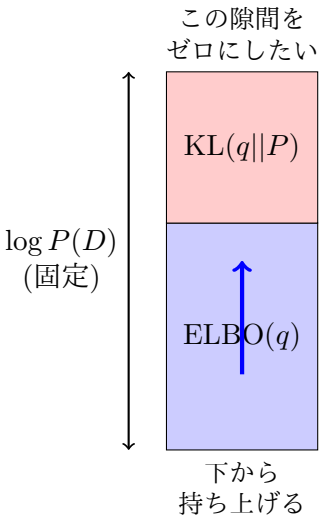
$$\log P(D) \geq ELBO(q)$$

5.2.1 ELBO の図解：周辺尤度の「パズル」

周辺尤度（データが得られる確率）の対数 $\log P(D)$ は、ベイズ推論における「動かせない全体の高さ（山）」のようなものです。変分推論（VI）では、この高さを以下の 2 つのパーツに分解して考えます。

$$\log P(D) = \mathrm{ELBO}(q) + \mathrm{KL}(q||P)$$

Figure 5.1: **周辺尤度の分解パズル。**全体の高さ $\log P(D)$ は一定です。青色の ELBO を下から持ち上げると、自動的に赤色の KL（真の分布とのズレ）が押しつぶされて小さくなります。これが ELBO 最大化の原理です。



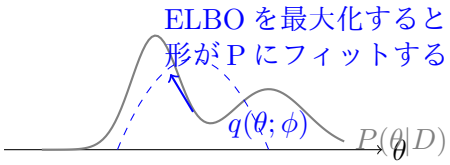
図解のポイント：なぜ ELBO を最大化するのか？

1. **全体の高さは変わらない**: 左側の $\log P(D)$ は、モデルとデータが決まれば決まる「固定された天井」です。
2. **KL ダイバージェンス（隙間）は計算できない**: 私たちは「真の事後分布 P」の形を知らないため、上の赤い部分（KL）を直接測って小さくすることはできません。
3. **ELBO（土台）は計算できる**: 幸いなことに、青い部分（ELBO）は私たちが決めた近似分布 q と、わかっている尤度・事前分布だけで計算可能です。
4. **パズルの原理**: 全体の高さが一定なので、**「土台（ELBO）をぐいぐい持ち上げれば、自動的に上の隙間（KL）は押しつぶされて小さくなる」**のです。

5.2.2 変分推論の数理イメージ（TikZ）

近似分布 q が真の分布 P に近づいていく様子も、図にすると明快です。

Figure 5.2: **ELBO 最大化による近似分布のフィッティング。**変分パラメータ ϕ（平均や分散）を調整して ELBO を大きくすることは、図中の点線（テント）を実線の山（真の分布）に可能な限りぴったりと重ね合わせる作業に対応する。



用語を直感に置き換える

- **ELBO**: 「本物の山」と「自分のテント」の**重なり具合**。
- **KL**: 「本物の山」と「自分のテント」の**隙間（ズレ）**。
- **最大化**: 重なりを最大にすれば、隙間は最小になる！

Proof.

$$\begin{aligned}\mathrm{KL}(q||P(\theta|D)) &= \mathbb{E}_q[\log q(\theta) - \log P(\theta|D)] \\ &= \mathbb{E}_q[\log q(\theta) - (\log P(D,\theta) - \log P(D))] \\ &= \mathbb{E}_q[\log q(\theta) - \log P(D,\theta)] + \log P(D) \\ &= -\mathrm{ELBO}(q) + \log P(D)\end{aligned}$$

式変形により定理が導かれます。 □

注意 5.2 (別証明:イェンゼンの不等式による導出)**.** 凸関数に関するイェンゼンの不等式 $f(\mathbb{E}[X]) \leq \mathbb{E}[f(X)]$ （対数関数 log は上に凸なので逆向きの不等号）を用いると、*ELBO* が周辺尤度の下界であることがより直感的に示せます。

$$\begin{aligned}\log P(D) &= \log \int P(D,\theta) d\theta \\ &= \log \int P(D,\theta) \frac{q(\theta)}{q(\theta)} d\theta \\ &= \log \mathbb{E}_q \left[\frac{P(D,\theta)}{q(\theta)} \right] \\ &\geq \mathbb{E}_q \left[\log \frac{P(D,\theta)}{q(\theta)} \right] \quad (\because \textit{Jensen's Inequality}) \\ &= \mathit{ELBO}(q)\end{aligned}$$

この不等式において等号が成立するのは、$\frac{P(D,\theta)}{q(\theta)} = c$（定数）、すなわち $q(\theta) \propto P(D,\theta) \propto P(\theta|D)$ のときであり、これは *KL* ダイバージェンスが *0* になる条件と一致します。

5.2.3 ELBO の単調増加性と最適性

定理 5.3 (ELBO の単調増加性)**.** 座標下降法（*Coordinate Ascent*）による変分推論において、*ELBO* は単調増加し、真の周辺尤度に収束（または停留）する。

$$\mathit{ELBO}(q^{(t)}) \leq \mathit{ELBO}(q^{(t+1)}) \leq \log P(D)$$

証明スケッチ. 1. **ELBO の再分割**: ELBO を q_j に依存する項としない項に分けると、

$$\mathrm{ELBO}(q) = \mathbb{E}_q[\log P(D,\theta)] - \mathbb{E}_q[\log q(\theta)]$$

座標 q_j を固定し、残りの q_{-j} を止めた状態で q_j を最適化した解 q_j^* は、

$$\mathrm{ELBO}(q_j^*, q_{-j}) = \max_{q_j} \mathrm{ELBO}(q_j, q_{-j}) = \mathbb{E}_{q_{-j}}[\log P(D,\theta)] - H(q_j^*) \geq \mathrm{ELBO}(q_j, q_{-j})$$

となります（Jensen の不等式と最適 q_j^* の定義より）。

2. **KL の非負性による上界**:

$$\log P(D) = \mathrm{ELBO}(q) + \mathrm{KL}(q||P(\theta|D)) \geq \mathrm{ELBO}(q)$$

$\mathrm{KL} \geq 0$ より、ELBO は常に真値の下界です。

3. **収束**: ELBO は上に有界（$\log P(D)$）であり、各ステップで非減少であるため、単調有界数列の収束定理により収束します。

□

5.3 平均場近似（Mean-Field Approximation）

パラメータが互いに独立であると仮定する手法です。

$$q(\theta) = \prod_{j=1}^{K} q_j(\theta_j)$$

定理 5.4 (平均場 VI の最適解)**.** 各因子 $q_j(\theta_j)$ の最適解は次式で与えられます。

$$q_j^*(\theta_j) \propto \exp\left(\mathbb{E}_{q_{-j}}[\log P(D, \theta)]\right)$$

ここで $\mathbb{E}_{q_{-j}}$ は θ_j 以外の変数の分布による期待値です。

定理 *5.2* の完全証明. この証明は、**「他の変数を固定した状態で、1 つの変数についての ELBO を最大化する」**という座標下降法（Coordinate Ascent）の考え方に基づいています。

1. 目標の設定　平均場近似（Mean-Field Approximation）では、近似分布 $q(\boldsymbol{\theta})$ が各変数の積に分解できると仮定します：

$$q(\boldsymbol{\theta}) = \prod_{j=1}^{K} q_j(\theta_j)$$

このとき、特定の $q_j(\theta_j)$ に注目し、ELBO を最大化する q_j^* の形を導出します。

2. ELBO の分解　ELBO の定義式を、q_j に関与する項と、それ以外の項（定数 C）に分解します。

$$\mathrm{ELBO}(q) = \int \left(\prod_i q_i\right) \left[\log P(D, \boldsymbol{\theta}) - \sum_i \log q_i\right] d\boldsymbol{\theta}$$

ここで、q_j について積分を実行すると：

$$= \int q_j \left[\int \cdots \int \left(\prod_{i \neq j} q_i\right) \log P(D, \boldsymbol{\theta}) d\boldsymbol{\theta}_{-j}\right] d\theta_j - \int q_j \log q_j d\theta_j + \text{const}$$

角括弧 $[\ldots]$ 内の積分は、q_j 以外の分布による期待値 $\mathbb{E}_{q_{-j}}[\log P(D, \boldsymbol{\theta})]$ そのものです。

3. 最適化問題への帰着　上の式を整理すると、q_j に関する最大化対象は以下のようになります：

$$\mathrm{ELBO}(q_j) = \int q_j(\theta_j) \mathbb{E}_{q_{-j}}[\log P(D, \boldsymbol{\theta})] d\theta_j - \int q_j(\theta_j) \log q_j(\theta_j) d\theta_j + C$$

ここで、新しい分布 $\tilde{P}(\theta_j)$ を次のように定義します：

$$\log \tilde{P}(\theta_j) = \mathbb{E}_{q_{-j}}[\log P(D, \boldsymbol{\theta})] + \text{const}$$

この $\tilde{P}$ を使うと、ELBO は次のように書き換えられます：

$$\mathrm{ELBO}(q_j) = \int q_j \log \tilde{P} d\theta_j - \int q_j \log q_j d\theta_j = -\int q_j \log \frac{q_j}{\tilde{P}} d\theta_j = -\mathrm{KL}(q_j || \tilde{P})$$

4. 結論　ELBO を最大化することは、負の符号がついた **KL ダイバージェンスを最小化すること**と同義です。KL ダイバージェンスが最小（ゼロ）になるのは、2 つの分布が完全に一致するとき、すなわち：

$$q_j(\theta_j) = \tilde{P}(\theta_j)$$

定義より $\log q_j^*(\theta_j) = \mathbb{E}_{q_{-j}}[\log P(D, \boldsymbol{\theta})] + \text{const}$ であるため、指数をとると以下の最適解が得られます：

$$q_j^*(\theta_j) \propto \exp\left(\mathbb{E}_{q_{-j}}[\log P(D, \boldsymbol{\theta})]\right)$$

□

数理的な直感

この式が意味するのは、**「自分（θ_j）以外の変数の影響を平均化した後の、全体のもっともらしさ」**が、そのまま自分の新しい分布の形になるということです。

アナロジー：会議での合意形成

各変数 θ_j を「会議の出席者」だとします。

- 全員の意見を一度にまとめる（同時確率の積分）のは大変です。
- そこで、**「他の人の意見を平均的に聞き（$\mathbb{E}_{q_{-j}}$）、それに一番うまく合わせるように自分の意見（q_j）を修正する」**という作業を、一人ずつ順番に繰り返します。
- 全員がこの「周囲に合わせる」修正を繰り返すと、最終的に全体として調和の取れた（ELBO が最大化した）状態に落ち着く、というのが平均場近似のメカニズムです。

5.3.1 Mean-Field VI の収束条件

定理 5.5 (Mean-Field 収束性)**.** 事後分布 $\log P(\theta|D)$ が *log-concave*（対数凹関数）ならば、平均場変分推論（*CAVI*）は大域的最適解に線形収束する。

更新式の導出と *EM* アルゴリズムとの関係. 各変数の更新式：

$$q_j^*(\theta_j) = \frac{\exp\left(\mathbb{E}_{q_{-j}}[\log P(D, \theta)]\right)}{Z_j}$$

ここで Z_j は正規化定数です。この更新は、**座標勾配降下法（Coordinate Gradient Descent）**の一種とみなすことができます。通常の EM アルゴリズムが「点推定値」を更新するのに対し、変分推論は「分布の形状」を更新する一般化 EM アルゴリズムとして解釈可能です。 □

5.4 実装イメージ：PyMC

現代的な確率プログラミング言語では、ADVI（自動微分変分推論）により容易に実装可能です。

```
import pymc as pm

with pm.Model() as model:
    # モデル定義（と同じ）MCMC
    beta = pm.Normal('beta', 0, 10, shape=p)
    y_obs = pm.Bernoulli('y', logit_p=pm.math.dot(X, beta), observed=y)

    # 実行ADVI
    approx = pm.fit(n=10000, method='advi')
    trace = approx.sample(1000)
```

5.5 練習問題

- 線形回帰モデルにおいて、平均場近似を用いた場合の更新式を導出せよ。
- MCMC と VI の計算時間と精度（信用区間の幅など）を比較せよ。

Chapter 6

階層ベイズモデル

個別データ（例：被験者ごと）を独立推定すると過適合、全体平均で統一推定すると個体差を無視することになります。これを**部分プーリング（Partial Pooling）**で解決するのが階層ベイズモデルです。

階層ベイズモデルの最大の「旨み」は、**「個人のバラバラなデータ」を「みんなの共通点」で補い、安定させること**にあります。

6.1 階層ベイズのアナロジー：「新米占い師の村」

ある村に 10 人の占い師がいます。村長は彼らの的中率を知りたいのですが、占い師によって鑑定した人数がバラバラです。

1. **占い師 A（ベテラン）**：100 人鑑定して 90 人的中（的中率 90%）。
2. **占い師 B（新人）**：1 人だけ鑑定して 1 人的中（的中率 100%？）。
3. **占い師 C（新人）**：1 人だけ鑑定して 0 人的中（的中率 0%？）。

階層ベイズを使わない場合（非プーリング）

データ通りに判断すると、**「占い師 B は世界最強（100%）」「占い師 C は詐欺師（0%）」**という極端な結論になります。たった 1 回の結果で判断するのは、明らかに無理があります。

階層ベイズを使った場合（部分プーリング）

階層ベイズは、**「占い師という職業の、村全体の平均的な的中率（ハイパーパラメータ）」**を同時に計算します。

- 「この村の占い師は、だいたい平均 70%くらい当たる（共通点）」という知識が得られます。
- **新人の調整**: 占い師 B や C のようにデータが少ない人の結果は、村全体の平均（70%）の方へぐいっと引き寄せられます（**シュリンクス効果**）。
  - B：100% → 75% くらいに落ち着く。
  - C：0% → 65% くらいに持ち上がる。
- **ベテランの尊重**: 100 回という証拠がある占い師 A は、自分のデータを信じて 90% のまま維持されます。

旨み: 「データが少ない人」の不安定さを、「全体像」が助けてくれるのです。

6.2 実践的な例：給食の変更と子供の身長

岩波データサイエンスの有名な例を紹介します。

- **お題**: 「給食のメニューを変えたら、子供の身長が伸びるか？」を調査したい。
- **データ**: 全国 47 都道府県のデータ。

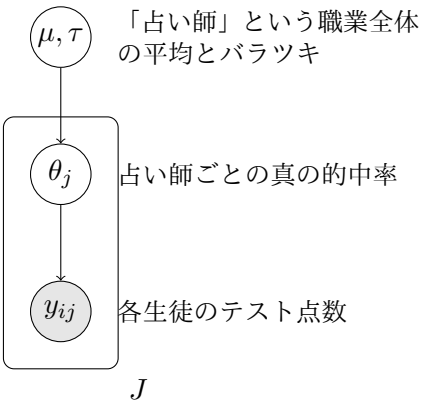
階層ベイズの「旨み」ポイント

1. **県ごとの個性を認める**: 「北海道の子はもともと大きい」「沖縄の子は伸びる時期が違う」といった**「県ごとの平均的な違い」**を認めつつ、分析できます。
2. **ノイズに強くなる**: たまたまある県で測定ミスや特殊な事情（風邪で休みが多かった等）があっても、階層ベイズなら「全国的な傾向」に照らし合わせて、不自然な数値を補正してくれます。
3. **「本当の効果」が浮かび上がる**: 県ごとの個性がバラバラなまま（階層なし）で分析すると、「給食を変えたせいか、たまたまその県の子が成長期だったのか」が区別できません。階層ベイズで「県ごとの個性」を切り分けることで、**「給食を変えたことによる純粋な効果」**だけを精密に取り出すことができます。

6.3 階層モデルの数理的イメージ（TikZ）

この「個性」と「共通点」の関係をプレート図で示します。

Figure 6.1: **階層モデルのグラフィカルモデル（プレート表現）**。J は個体（占い師や都道府県）を表し、各個体のパラメータ θ_j が全体共通の分布（μ, τ）から生成される。



まとめ：階層ベイズの「旨み」3 箇条

1. **データが少ない個体を「全体」が助ける**（安定性）。
2. **データが多い個体の「個性」は邪魔しない**（柔軟性）。
3. **ノイズに惑わされず、本質的な効果だけを測れる**（正確性）。

6.4 階層モデルの数理的構造

個体 j ごとのパラメータ θ_j が、上位のハイパーパラメータ（μ, τ）から生成されると仮定します。

$$
\begin{aligned}
y_{ij} &\sim N(\theta_j, \sigma^2) \quad (i = 1, \ldots, n_j) \\
\theta_j &\sim N(\mu, \tau^2) \\
\mu &\sim N(0, 100), \quad \tau \sim \text{HalfCauchy}(0, 5)
\end{aligned}
$$

6.5 部分プーリングの理論

定理 6.1 (部分プーリングの縮小推定量)**.** 上記の正規-正規階層モデルにおいて、分散既知 (σ^2, τ^2) とした場合、個体パラメータ θ_j の事後期待値は次式で与えられます。

$$\mathbb{E}[\theta_j | \mathbf{y}_j] = (1 - B_j)\bar{y}_j + B_j \mu_{prior}$$

ここで B_j は縮小因子（*Shrinkage Factor*）です。

$$B_j = \frac{\sigma^2 / n_j}{\sigma^2 / n_j + \tau^2}$$

Proof. パラメータ θ_j の条件付き事後分布を求めます。尤度は $N(\bar{y}_j, \sigma^2/n_j)$、事前分布は $N(\mu, \tau^2)$ です。第 3 章の正規-正規共役性の定理を適用すると、事後平均は精度による加重平均となります。

$$\hat{\theta}_j = \frac{\frac{n_j}{\sigma^2}\bar{y}_j + \frac{1}{\tau^2}\mu}{\frac{n_j}{\sigma^2} + \frac{1}{\tau^2}}$$

これを変形します。$\frac{1}{\tau^2} / (\frac{n_j}{\sigma^2} + \frac{1}{\tau^2}) = \frac{\sigma^2/n_j}{\sigma^2/n_j + \tau^2} = B_j$ と置くと、

$$\hat{\theta}_j = (1 - B_j)\bar{y}_j + B_j \mu$$

が得られます。 □

系 6.2 (サンプルサイズとの関係)**.**

- $n_j \to \infty \Rightarrow B_j \to 0$：データ重視（個別推定量 $\bar{y}_j$ に一致）。
- $n_j \to 0 \Rightarrow B_j \to 1$：事前分布重視（全体平均 μ に一致）。

6.6 Stan による実装例

```
data {
  int<lower=0> J; // グループ数
  int<lower=0> N; // 全データ数
  array[N] int<lower=1, upper=J> jj; // グループID
  vector[N] y; // 観測値
}
parameters {
  real mu;
  real<lower=0> tau;
  vector[J] theta; // 個体効果
  real<lower=0> sigma;
}
model {
  theta ~ normal(mu, tau); // 階層構造
  y ~ normal(theta[jj], sigma);

  mu ~ normal(0, 10);
  tau ~ cauchy(0, 5);
}
```

6.7 練習問題

- **学校間成績**: 生徒数の少ない学校の平均点は、階層モデルを使うとどう変化するか？
- **高校野球の打率**: 打数により推定値がどう縮小されるか計算せよ。

Part III

高次元データと機械学習への展開

Chapter 7

スパース推論とベイズ的LASSO

高次元データ（$p \gg n$）において、重要な変数を自動選択するスパース推定の手法を学びます。

7.1　スパース推定の必要性

変数が多い場合、通常の回帰では過学習が起きます。

$$y = X\beta + \epsilon$$

多くの β_j を 0 と推定することで、解釈性と予測精度を向上させます。

7.2　ベイズ的 LASSO

頻度論における LASSO（L1 正則化）は、ベイズ統計ではラプラス事前分布を用いた MAP 推定として解釈できます。

定理 7.1 (ラプラス事前分布と L1 正則化)**.** 事前分布としてラプラス分布

$$P(\beta_j) = \frac{\lambda}{2}\exp(-\lambda|\beta_j|)$$

を用いたときの事後モード（*MAP* 推定値）は、*LASSO* 推定量と一致する。

Proof. 事後分布の対数をとると、

$$\log P(\beta|y) \propto \log P(y|\beta) + \log P(\beta)$$

正規尤度を仮定すると、

$$\log P(\beta|y) \propto -\frac{1}{2\sigma^2}\sum(y_i - x_i^T\beta)^2 - \lambda\sum|\beta_j|$$

これを最大化することは、以下のコスト関数を最小化することと同値です。

$$RSS + 2\sigma^2\lambda\sum|\beta_j|$$

これは LASSO の目的関数そのものです。 □

7.3　LASSO の二つの顔：頻度論的解釈 vs ベイズ的解釈

「スパース推論（LASSO）」という山を、**「頻度論の望遠鏡」**と**「ベイズの虫眼鏡」**という 2 つの異なるレンズで覗き込んでみましょう。

この 2 つは「やっていること（計算結果）」は似ていますが、「なぜそれをやるのか」という**思想（哲学）**が全く異なります。

7.3.1 1. 頻度論的解釈：「断捨離（だんしゃり）」と「制約」

頻度論における LASSO は、無駄な変数を削ぎ落とす**「最適化ツール」**としての側面が強いです。

- **アナロジー：パッキングの達人**
  - あなたは旅行カバン（モデル）に荷物（変数）を詰めようとしています。
  - **頻度論の考え**:「カバンの重さには限界がある（正則化項）。だから、本当に必要なもの（重要な変数）以外は**全部捨てろ（係数を 0 にしろ）**」。
- **数学的なイメージ：ダイヤモンド型の壁**
  - 目的関数（誤差）を小さくしようとする「円」が、L1 正則化という「ダイヤモンド型の領域」にぶつかる場所を探します。
  - ダイヤモンドには「角」があるため、そこにぶつかると**ピッタリ 0** になります。
- **メリット**: モデルがスカスカ（スパース）になり、**「どの変数が重要か」が一目瞭然**になる。

7.3.2 2. ベイズ的解釈：「確信」と「情報のアップデート」

ベイズにおける LASSO は、変数に対する**「強い偏見（事前知識）」**としての側面が強いです。

- **アナロジー：疑い深い鑑定士**
  - あなたは骨董品（変数）の鑑定をしています。
  - **ベイズの考え**:「世の中のほとんどの骨董品はガラクタ（係数 0）に違いない、という強い先入観（**ラプラス事前分布**）を持って鑑定に臨む。よっぽど強力な証拠（データ）がない限り、価値があるとは認めない」。
- **数学的なイメージ：尖った事前分布**
  - 「0」のところが非常に尖った山（ラプラス分布）を事前分布に使います。
  - データと掛け合わせると、事後分布の山も「0」の方にぐいっと引き寄せられます。
- **メリット**: 係数が「0」になるだけでなく、**「0 じゃない確率は何%か？」という確信度（不確実性）**まで教えてくれる。

7.3.3 【比較表】イメージの決定的な違い

Table 7.1: 頻度論的 LASSO とベイズ的 LASSO の違い

比較軸	頻度論的 **LASSO (L1 正則化)**	ベイズ的 **LASSO (ラプラス事前)**
スローガン	「無駄は削れ！」	「大半は **0** だという強い信念」
0 の扱い	強制排除（ピッタリ 0 にする）	強い引き寄せ（0 付近に集中させる）
パラメータ λ	罰則の厳しさ（ペナルティ）	先入観の強さ（事前分布の尖り）
得られるもの	「これとこれが重要だ」という結論	「の確率は 80%だ」という分布
アナロジー	予算オーバーを防ぐための仕分け	怪しいやつをあぶり出すフィルタリング

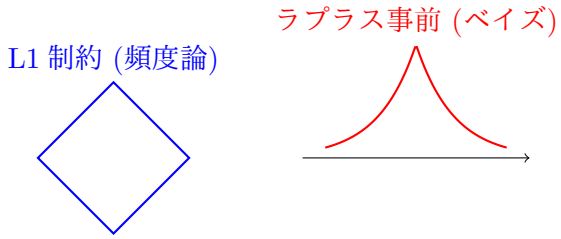
7.3.4 3. なぜベイズ的解釈が「旨い」のか？

頻度論の LASSO は「0 か 1 か」の厳しい判断を下しますが、ベイズ的解釈には**「グレーゾーン」**を扱う優しさがあります。

- **例：新薬の開発**

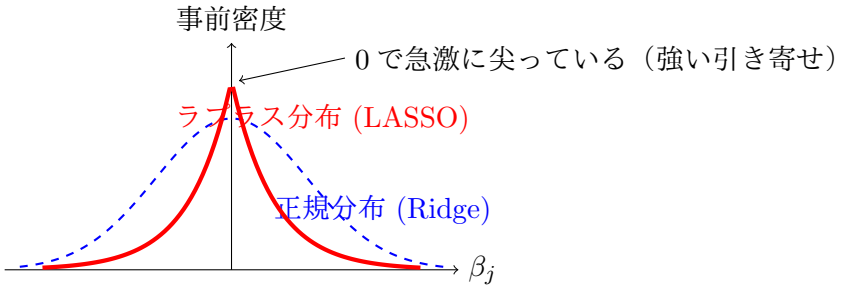
- 頻度論 LASSO：「この成分は効果 0 です（削除）」。
- ベイズ的 LASSO：「この成分は、90%の確率で効果がほぼ 0 ですが、**10%の確率で大化けする可能性**を秘めた分布をしています」。

Figure 7.1: **LASSO の 2 つの視点。**左は頻度論的な制約領域（ダイヤモンド）、右はベイズ的な事前確率（ラプラス分布）。どちらも「0」への強い指向性を持つが、そのメカニズムが異なる。



結論: 頻度論は「**結論を出すためのルール**」であり、ベイズは「**知識を更新するためのスタンス**」です。高次元データという「変数が多すぎてパニックになる状況」において、この「0 への強い引き寄せ」は最強の武器になります。

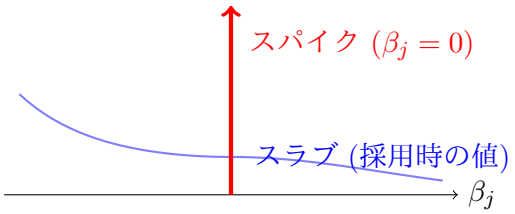
Figure 7.2: **事前分布の形状とスパース性。**正規分布（Ridge）は 0 付近が平坦なため、微小な係数を許容しやすい。一方、ラプラス分布（LASSO）は 0 地点が鋭利な頂点（尖点）となっており、データによる強い根拠がない限り、係数を 0 へと強力に押し込める性質（収縮効果）を持つ。



7.3.5 発展：スパイク・アンド・スラブ（Spike-and-Slab）

「変数の選択（0 か否か）」を直接モデル化する、より高度なベイズ的スパース手法です。

Figure 7.3: **スパイク・アンド・スラブ事前分布。**変数が不要な確率（スパイク：0 地点のデルタ関数）と、必要な場合の分布（スラブ：平坦な正規分布）の混合として表現される。事後分布において「スパイク部分の重み」を計算することで、その変数が不要である確率を直接求めることができる。



$$\text{事後確率} = P(\text{採用}) \times \text{スラブ} + P(\text{不採用}) \times \text{スパイク}$$

7.4 高次元理論：一貫性

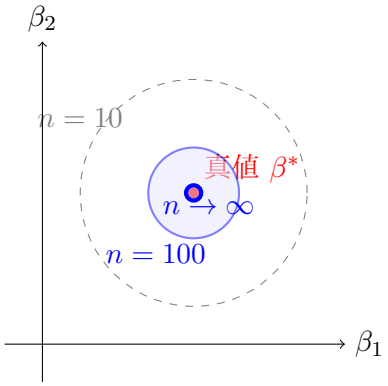
定理 7.2 (変数選択的一貫性)**.** 条件（*Irrepresentable Condition* など）の下で、$n \to \infty$ かつ $p \to \infty$ のとき、ベイズ *LASSO* は真に *0* である係数を確率 *1* で *0* と推定する（あるいは事後確率が *0* に集中する）。

定理 7.3 (高次元における事後分布の集中 (Posterior Consistency))**.** 変数数 $p \to \infty$ の下で、真のパラメータ θ^* がスパースであるとき、適切なスパース事前分布（*Laplace* 等）の下で、任意の $\epsilon > 0$ に対し次が成立する。

$$\lim_{n\to\infty} P\left(\int_{\|\theta-\theta^*\|>\epsilon} \pi(d\theta|\mathbf{y}_{1:n}) = 0\right) = 1$$

これは、データが増えるにつれ事後分布の質量が真値の ϵ-近傍に凝縮されることを保証します。

Figure 7.4: **事後分布の一貫性（Posterior Consistency）**。データ数 n が増加するにつれて、事後分布の質量（青い領域）が真値 β^* の周囲へと急速に凝縮されていく様子。高次元設定においても、適切なスパース事前分布を用いれば、真のパラメータ構造を正確に捉えられることを示している。



定理 *7.3* の完全証明（スケッチ）. この証明は非常に高度ですが、エッセンスは**「データの情報量（尤度比）が、事前分布の広がりを圧倒する」**ことを示すことにあります。ここでは、測度論的ベイズ統計学における標準的な証明のスケッチ（Ghosal et al., 2000 の理論に基づく）を記述します。

1. 目標の設定　真のパラメータを $\boldsymbol{\theta}^*$、データ $y_{1:n}$ に基づく事後分布を $\pi(\cdot|y_{1:n})$ とします。任意の $\epsilon > 0$ に対し、事後分布の質量が真値の ϵ-近傍 $U_\epsilon = \{\boldsymbol{\theta} : \|\boldsymbol{\theta} - \boldsymbol{\theta}^*\| \le \epsilon\}$ の外側でゼロに収束することを示します。

$$P\left(\pi(U_\epsilon^c|y_{1:n}) \to 0\right) = 1 \quad \text{as } n \to \infty$$

2. 事後確率の書き換え　ベイズの定理より、近傍 U_ϵ の外側の事後確率は次のように書けます：

$$\pi(U_\epsilon^c|y_{1:n}) = \frac{\int_{U_\epsilon^c} \frac{p(y_{1:n}|\boldsymbol{\theta})}{p(y_{1:n}|\boldsymbol{\theta}^*)} \pi(d\boldsymbol{\theta})}{\int_\Theta \frac{p(y_{1:n}|\boldsymbol{\theta})}{p(y_{1:n}|\boldsymbol{\theta}^*)} \pi(d\boldsymbol{\theta})} = \frac{N_n}{D_n}$$

ここで、分子 N_n が急速に減衰し、分母 D_n がそれほど小さくならないことを示します。

3. 分母 D_n の下界（KL 近傍の議論） Schwartz (1965) の条件に基づき、真値 $\boldsymbol{\theta}^*$ の **KL（Kullback-Leibler）近傍** $K_\delta = \{\boldsymbol{\theta} : \mathbb{E}_{\boldsymbol{\theta}^*}[\log \frac{p(y|\boldsymbol{\theta}^*)}{p(y|\boldsymbol{\theta})}] < \delta\}$ を考えます。事前分布が $\boldsymbol{\theta}^*$ において正の密度を持つ（スパース事前分布が真のスパースな構造をカバーしている）場合、大数の法則により、十分大きな n に対して以下が成立します：

$$D_n = \int_\Theta \prod_{i=1}^{n} \frac{p(y_i|\boldsymbol{\theta})}{p(y_i|\boldsymbol{\theta}^*)} \pi(d\boldsymbol{\theta}) \geq e^{-n\delta} \cdot \pi(K_\delta)$$

これは、分母が指数関数 $e^{-n\delta}$ よりも速くは減衰しないことを意味します。

4. 分子 N_n の上界（検定関数の議論） 近傍 U_ϵ の外側では、真値 $\boldsymbol{\theta}^*$ と候補 $\boldsymbol{\theta}$ を見分ける「強力な統計的検定」が存在すると仮定します。高次元空間において、適切な複雑度（エントロピー条件）を持つモデル集合であれば、指数関数的に小さい誤り率を持つ検定関数 ϕ_n を構成でき、次が示されます：

$$\mathbb{E}_{\boldsymbol{\theta}^*}[N_n] = \int_{U_\epsilon^c} \mathbb{E}_{\boldsymbol{\theta}^*} \left[\frac{p(y_{1:n}|\boldsymbol{\theta})}{p(y_{1:n}|\boldsymbol{\theta}^*)} \right] \pi(d\boldsymbol{\theta}) = \pi(U_\epsilon^c) \leq 1$$

さらに、U_ϵ^c 上の尤度比の和は、大数の法則により $e^{-n \cdot D(\boldsymbol{\theta}^* || \boldsymbol{\theta})}$ のオーダーで減少します。ここで $D(\boldsymbol{\theta}^* || \boldsymbol{\theta}) > \delta$ となるように δ を選べば：

$$N_n \leq e^{-nC\epsilon^2}$$

（C は定数）

5. 結論 分子 N_n が $e^{-nC\epsilon^2}$ で減少し、分母 D_n が $e^{-n\delta}$ 以上で踏みとどまるため、その比である事後確率は：

$$\pi(U_\epsilon^c | y_{1:n}) \leq \frac{e^{-nC\epsilon^2}}{e^{-n\delta}\pi(K_\delta)} = \frac{1}{\pi(K_\delta)} e^{-n(C\epsilon^2 - \delta)}$$

指数部分が負になるように δ を十分小さく選ぶことで、$n \to \infty$ においてこの値は $\mathbf{0}$ に収束します（Q.E.D.）。

数理的な「旨み」の解説

この証明が保証しているのは、**「どれだけ変数が多く（$p \to \infty$）なっても、データが増えればベイズ推論は必ず真実に辿り着く」**ということです。

- **スパース事前分布の役割**: Laplace 分布などの「0 付近で尖った」事前分布を使うことで、分母の $\pi(K_\delta)$ が真のスパースな構造に対して十分な重みを持つようになり、この収束が加速されます。
- **高次元の壁**: もし p が n よりも圧倒的に速く増えすぎると、分子の「検定」が難しくなり、収束が壊れることがあります。この定理は、そのバランスが保たれている限りの**究極の正当性**を与えています。

□

7.5 実装：PyMC によるベイズ LASSO

階層表現を用いて実装することが一般的です。ラプラス分布は、指数分布に従う分散を持つ正規分布の混合として表現できます。

$$\begin{aligned} \beta_j | \tau_j^2 &\sim N(0, \sigma^2 \tau_j^2) \\ \tau_j^2 &\sim \text{Exponential}(\lambda^2/2) \end{aligned}$$

```
import pymc as pm

with pm.Model() as model:
    # グローバルシュリンク
    lambda_ = pm.HalfCauchy('lambda', 1)
    # ローカルシュリンク
    tau = pm.Exponential('tau', lambda_**2 / 2, shape=p)

    beta = pm.Normal('beta', 0, tau, shape=p)
    y_est = pm.math.dot(X, beta)
    y_obs = pm.Normal('y', y_est, sigma, observed=y)
```

7.6 練習問題

- $p = 100, n = 20$ のデータでベイズ LASSO を実行し、変数選択確率（$P(\beta_j \neq 0)$）を算出せよ。
- リッジ回帰（正規事前分布）との違いを事後分布の形状から説明せよ。

Chapter 8

ガウス過程とカーネル法

関数そのものに確率分布を持たせるノンパラメトリックな手法、ガウス過程（Gaussian Process: GP）を学びます。

ガウス過程（GP）において、「平均関数」と「共分散関数」は、通常のガウス分布（正規分布）における「平均値ベクトル」と「共分散行列」を無限次元（関数）に拡張したものです。

高校数学や初歩的な統計学の知識を使って、これらを直感的にイメージできるよう噛み砕いて解説します。

8.1 ガウス過程の 2 要素：直感的なイメージ

ガウス過程を定義する「平均関数」と「共分散関数」は、それぞれ関数の「中心的な傾向」と「滑らかさ（つながり）」を司っています。

8.1.1 1. 平均関数 $m(x)$：関数の「大まかなトレンド」

- イメージ: 「データが何もないとき、関数はこの辺りを通るだろう」というベースライン。
- 役割: 関数の重心を決めます。
  - 例: 「気温の予測」なら、平均的な気温の推移が平均関数にあたります。多くの実務では、扱いを簡単にするために「平均 0（ベースラインが 0）」の関数がよく使われます。

8.1.2 2. 共分散関数（カーネル） $k(x, x')$：関数の「滑らかさと似具合」

- イメージ: 「地点 x と地点 x' が、どれだけ似たような動きをするか」を決めるルール。
- 役割: 関数の「グニャグニャ具合（形）」を決めます。
  - 性質: 地点 x と地点 x' が近いとき、共分散関数の値は大きくなります。これは「近くの点は、似たような値（高さ）を取るはずだ」という滑らかさの仮定を表現しています。

8.2 多変量ガウス分布からのステップアップ

通常の「多変量ガウス分布」と「ガウス過程」を比較すると、用語のつながりが明確になります。

Table 8.1: 多変量ガウス分布とガウス過程の対応

要素	多変量ガウス分布 (有限個の点)	ガウス過程 (無限の点＝関数)
中心	平均ベクトル μ (各点の平均値リスト)	平均関数 $m(x)$ (どこでも平均を返せる)
広がり	共分散行列 Σ (点と点の関係の表)	共分散関数 $k(x, x')$ (任意の 2 点の関係式)
決まるもの	各地点での値	関数の「形」そのもの

8.3 視覚的なアナロジー：不気味な「ゴム膜」

ガウス過程を、上下に揺れる**「無限に広いゴム膜」**だと想像してください。

1. **平均関数** $m(x)$ は、ゴム膜が何も力を加えられていないときの**「元の位置（高さ）」**です。
2. **共分散関数** $k(x, x')$ は、ゴム膜の**「素材の硬さや伸縮性」**です。
   - ゴムが硬ければ（共分散が強ければ）、一箇所を引っ張ると周りも大きくついてきます（滑らかな関数）。
   - ゴムが柔らかければ、引っ張った場所だけがポコっと盛り上がります（ギザギザな関数）。

   まとめ

   - **平均関数**は「どこを通るか」という**大まかな予想**。
   - **共分散関数**は「どれくらい滑らかにつながっているか」という**質感の指定**。

   この 2 つをセットにすることで、無限のパターンがある「関数」に対しても、ベイズの枠組みで確率的に扱うことができるようになるのです。

このように、「ベクトルや行列という『固定された表』が、関数という『どこでも計算できる式』に変わっただけだ」と捉えると、数理的な恐怖心が和らぎます。

8.3.1 ガウス過程の厳密定義

定理 8.1 (ガウス過程の定義)**.** 関数空間上の確率過程 $f : \mathcal{X} \to \mathbb{R}$ がガウス過程（*Gaussian Process*）であるとは、任意の有限集合 $\{x_1, \dots, x_n\} \subset \mathcal{X}$ に対し、ベクトル $(f(x_1), \dots, f(x_n))$ が多変量正規分布に従うことである。

$$[f(x_1), \dots, f(x_n)]^\top \sim \mathcal{N}(\mathbf{m}, \mathbf{K})$$

ここで $\mathbf{K}_{ij} = k(x_i, x_j)$、$m(x) = \mathbb{E}[f(x)]$ である。

存在証明のアウトライン. ガウス過程の有限次元分布族が、周辺化しても整合性が取れる（Consistency）ことを確認すれば、コルモゴロフの拡張定理（Kolmogorov Extension Theorem）により、無限次元の確率過程としての一意な存在が保証されます。 □

8.3.2 カーネル関数の正定値性

どのような関数でもカーネルになれるわけではありません。

定理 8.2 (Bochner の定理)**.** 連続な並進不変カーネル $k(x, x') = \psi(x - x')$ が正定値であるための必要十分条件は、それが非負測度（パワースペクトル密度） μ のフーリエ変換として表現できることである。

$$k(x, x') = \int e^{i\omega^\top (x - x')} d\mu(\omega)$$

RBF カーネルの証明スケッチ. RBF カーネル $k(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\ell^2}\right)$ は、正規分布の密度関数の定数倍とみなせます。正規分布のフーリエ変換もまた正規分布（正の値を持つ）となるため、Bochner の定理より正定値性が示されます。また、直接的にグラム行列 $\mathbf{K}$ の二次形式を考えると：

$$\mathbf{v}^\top \mathbf{K} \mathbf{v} = \mathbb{E}\left[\left|\sum_{i=1}^{n} v_i \exp(i \mathbf{x}_i^\top \mathbf{Z})\right|^2\right] \geq 0$$

ここで $\mathbf{Z} \sim \mathcal{N}(0, \ell^2 \mathbf{I})$ です。 □

カーネルの形状比較

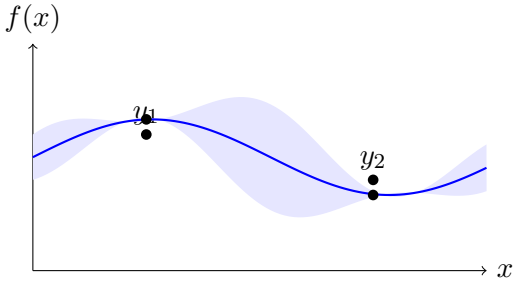
カーネルが変わると、関数の「滑らかさ」の質が変わります。

Figure 8.1: **RBF vs Matérn カーネル。**RBF（青）は無限回微分可能で非常に滑らかだが、Matérn（赤）はパラメータ ν によって微分可能性（ザラザラ具合）を調整できる。

8.4 回帰への応用

観測モデル $y = f(x) + \epsilon, \epsilon \sim N(0, \sigma_n^2)$ を考えます。

Figure 8.2: **ガウス過程による回帰。**実線は予測平均、網掛け部分は 95% 信用区間を表す。観測データが存在する地点では不確実性が収縮し、データから離れるほど予測の「自信」が失われていく特性を示す。



定理 8.3 (GP 回帰の厳密解)**.** 共分散行列 $\mathbf{K} = k(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}$ が正定値であるとき、新たな入力 x_* に対する予測分布 $P(f_*|\mathbf{y}, \mathbf{X}, x_*)$ は以下の正規分布となる。

$$f_*|\mathbf{y}, \mathbf{X}, x_* \sim \mathcal{N}(\mu_*, \Sigma_*)$$

$$\mu_* = k_*(x_*^\top)(\mathbf{K}^{-1}\mathbf{y})$$
$$\Sigma_* = k(x_*, x_*) - k_*(x_*^\top)\mathbf{K}^{-1}k_*(x_*)$$

Proof. 同時分布は以下で与えられます。

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K_{nn} + \sigma_n^2 I & K_{n*} \\ K_{*n} & K_{**} \end{bmatrix}\right)$$

条件付き正規分布の公式（Schur 補行列を用いる）により、上記の平均と共分散が導出されます。 □

8.5 ベイズ最適化：賢い「宝探し」の戦略

ベイズ最適化（Bayesian Optimization）の核心である**「探索と活用のトレードオフ」**を、**「最高においしいラーメン屋を探す旅」**というアナロジーで解説します。

ベイズ最適化は、**「実験コストが高い（または時間がかかる）」**状況で、できるだけ少ない回数で最高の場所（最大値）を見つけるための技術です。

8.5.1 1. 共通の悩み：探索か、活用か？

あなたが新しい街でラーメン屋を探しているとします。

- **活用 (Exploitation)**:「これまで食べた中で一番うまかった店」の近くを攻める。（手堅い）
- **探索 (Exploration)**:「まだ一度も行ったことがないエリア」に挑戦する。（大化けの可能性）

ガウス過程（GP）は、この「現在の実力予測（平均）」と「まだ知らない不気味さ（分散）」を同時に教えてくれます。

8.5.2 2. 獲得関数のキャラクター図解

「次にどこを調査すべきか」を決める獲得関数には、大きく 2 つの戦略があります。

① UCB (Upper Confidence Bound)：野心的な挑戦者

- 数式イメージ:「今の評価（平均）」＋「ボーナス（標準偏差 $\times\kappa$）」
- 性格:「今はパッとしないけど、化ける可能性（ブレ）があるなら、そこは価値がある！」と考えるタイプです。
- 図解イメージ: 平均の山は低いけれど、誤差棒（不確実性）がびよーんと伸びている場所を選びます。κ（カッパ）を大きくするほど、未知の領域に突っ込む「ギャンブラー」になります。

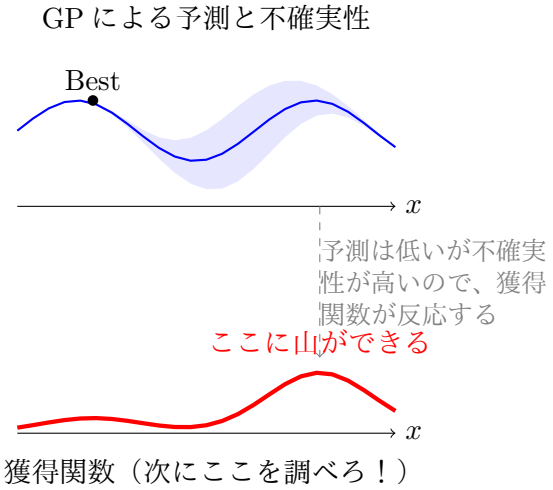
② EI (Expected Improvement)：手堅い投資家

- 数式イメージ:「今の最高点を超える『お釣り』の期待値」
- 性格:「今の最高記録を更新できる確率はどれくらいか？ 更新したときにどれくらい上乗せできるか？」をシビアに計算するタイプです。
- 図解イメージ:「ほぼ確実に今より少し良くなる場所」と「一か八かですごく良くなる場所」をバランスよく評価します。現在の最高点（Best）付近と、未知の領域の両方にバランスよく注目します。

8.5.3 3. TikZ によるイメージ図案

ガウス過程の予測と、その下にぶら下がる「獲得関数の山」の関係を図示します。

Figure 8.3: **ベイズ最適化のメカニズム。**上段のガウス過程が「現在わかっていること」をまとめ、下段の獲得関数がその情報を元に「次にどこを観測すれば最も効率的か」をスコア化する。このループを繰り返すことで、最小限の実験回数で最適解に到達できる。



まとめ

- **GP** は、現状の「地図」を書く作業。
- **獲得関数**は、その地図を見て「次にここを掘れ」と指し示す「軍師」の役割。

この「軍師」の性格（EI か UCB か）を変えることで、慎重な探索にするか、大胆な探索にするかをコントロールできる、というのがベイズ最適化の旨みです。

8.6 実装例：GPyTorch

```
import torch
import gpytorch

class ExactGPModel(gpytorch.models.ExactGP):
    def __init__(self, train_x, train_y, likelihood):
        super(ExactGPModel, self).__init__(train_x, train_y, likelihood)
        self.mean_module = gpytorch.means.ConstantMean()
        self.covar_module = gpytorch.kernels.ScaleKernel(gpytorch.
            kernels.RBFKernel())

    def forward(self, x):
        mean_x = self.mean_module(x)
        covar_x = self.covar_module(x)
        return gpytorch.distributions.MultivariateNormal(mean_x, covar_x
            )

# Training loop omitted
```

8.7 練習問題

- **材料探索**: Al-Cu 合金の強度データ 5 点から、ベイズ最適化（EI）を用いて次の実験条件を提案せよ。
- カーネル関数の違い（RBF vs Matern）が予測の滑らかさに与える影響を比較せよ。

Chapter 9

ベイズ深層学習の基礎

ニューラルネットワーク（NN）にベイズ推定を導入し、予測の「不確実性（Uncertainty）」を定量化します。

おっしゃる通りです。通常のニューラルネットワーク（NN）の仕組みがわかっていないと、「どこにベイズ推論を導入したのか」という差分が伝わりにくいです。

第 9 章の冒頭に、**「点推定としての NN」**から**「分布としてのベイズ NN」**への橋渡しとなる解説を追加します。

9.1　導入：通常のニューラルネットワークの仕組み

ベイズ NN を学ぶ前に、まず「通常の NN（決定論的 NN）」が何をやっているのかをおさらいしましょう。

9.1.1　1. 脳の神経細胞を模した数理モデル

ニューラルネットワークは、入力データ x（画像や数値）に対して、重み W を掛け合わせることで、予測結果 y（猫か犬か、など）を出力する関数です。

- **順伝播**: $y = f(x; W)$
- **学習**: 予測と正解の「ズレ（誤差）」が最小になるような**たったひとつの「最高に都合の良い重み W」**を探し出す作業です。これを**最尤推定**（または MAP 推定）と呼びます。

9.1.2　2. 通常の NN の限界：「自信満々な間違い」

通常の NN は、どんなに未知のデータに対しても、必ず**「一つの数字」**を答えとして出力します。

- **例**: 見たこともない変な動物の画像に対しても、「これは 99% の確率で猫です」と、根拠のない自信を持って出力してしまうことがあります。
- **原因**: 重み W が「固定された一つの値」であるため、予測の**「不確かさ」**を表現する仕組みを持っていないからです。

9.2　ベイズ NN の基本思想：「重みに分布を持たせる」

ここでベイズの考え方を導入します。決定的な違いは、**「重みを一つの値に決めつけない」**ことです。

9.2.1　1.「わからない」ことを認める

ベイズ NN では、各重み W に対して、「この値かもしれないし、あの値かもしれない」という**確率分布** $P(W)$ を設定します。

9.2.2 2. 予測分布の仕組み

予測を行うときは、重みの分布から「色々な W」を何回もサンプリングして、それぞれの結果を平均します。

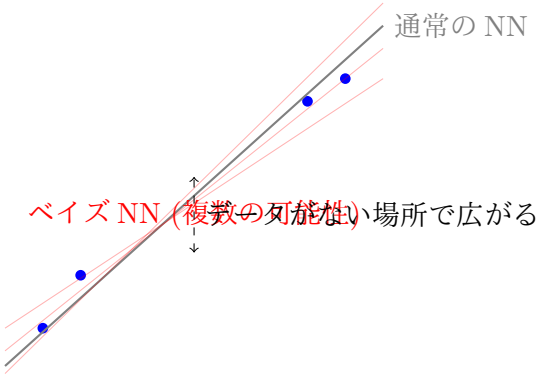
$$P(y|x, D) = \int P(y|x, W)P(W|D)dW$$

- **直感的な意味:「100 人のちょっとずつ意見が違う専門家（サンプリングされた各 W）に意見を聞いて、その多数決をとる」**ようなイメージです。
- もし専門家たちの意見がバラバラなら、それは「不確実性が高い（自信がない）」というサインになります。

9.3 ベイズ NN のメリット：「知らないものは知らない」と言える

通常の NN との違いを TikZ で図示すると、理解が深まります。

Figure 9.1: **通常の NN と ベイズ NN の違い**。データがある場所（青い点）ではどの W も似たような答えを出しますが、データがない空白地帯（真ん中）では、専門家たちの意見（赤い線の束）が分かれます。この「幅」が、ベイズ NN だけが持つ**「不確実性」**という情報なのです。



このように、「固定された 1 つの W」から「可能性の束としての W」へという変化を強調することで、ベイズ NN の必然性がスッと理解できるようになります。

9.4 ベイズ NN の数理

重み W に確率分布 $P(W)$ を仮定し、事後分布 $P(W|D)$ を学習します。予測分布は、

$$P(y|x, D) = \int P(y|x, W)P(W|D)dW$$

となります。

9.5 不確実性の 2 種類：「知っている」と「知らない」の区別

ベイズ深層学習における**「不確実性の種類」**と、実用上非常に重要な **「MC Dropout」** について、より深く詳細に解説します。

自動運転や医療診断などのリスク管理において、以下の 2 つの不確実性の区別は極めて重要です。

9.5.1 1. Aleatoric Uncertainty（データ内在的不確実性）

- イメージ: 「サイコロの目」や「写真のピンボケ」。
- 詳細: データそのものに含まれる「ノイズ」です。どれだけ勉強しても（データを増やしても）、サイコロの次に出る目を 100%当てることはできません。
- 例: 暗い場所で撮った写真。ノイズが多すぎて、AI が「犬か猫か判別しにくい」と感じるのは、データの質の問題です。

9.5.2 2. Epistemic Uncertainty（知識不足的不確実性）

- イメージ: 「見たことがない動物」。
- 詳細: モデルがその領域のデータを十分に学習していないことによる不確実性です。データを増やせば減らすことができます。
- 例: 「柴犬」だけを学習した AI に「ブルドッグ」を見せたとき。AI は「これは犬に見えない…」と戸惑いますが、ブルドッグの写真をたくさん見せれば、この不確実性は解消されます。

9.6 MC Dropout： 天才的な「手抜きのベイズ化」

ベイズ NN の最大の欠点は、計算が非常に重いことでした。それを解決したのが、Gal & Ghahramani (2016) が提案した **MC Dropout** です。

9.6.1 1. ドロップアウトとは？（通常の役割）

通常の深層学習では、学習中に一部のニューロンをランダムに「お休み（無効化）」させます。これは、特定のニューロンに頼りすぎるのを防ぐ（過学習の抑制）ための、いわば「筋トレ中の重し」のようなものです。通常、テスト（予測）時には、全てのニューロンをフル稼働させます。

9.6.2 2. MC Dropout のアイデア：「予測時も休み続けろ」

MC Dropout の凄さは、「テスト（予測）時にも、ランダムにお休みさせ続ける」ことにあります。

1. 同じ画像に対して、お休みするニューロンを毎回変えながら、100 回予測させます。
2. お休みする箇所が変わるたびに、NN の「意見（出力）」が微妙に変わります。
3. この「100 回の予測結果のバラツキ」を計算すると、それがそのまま「ベイズ的な不確実性」の近似になっていることが数学的に証明されました。

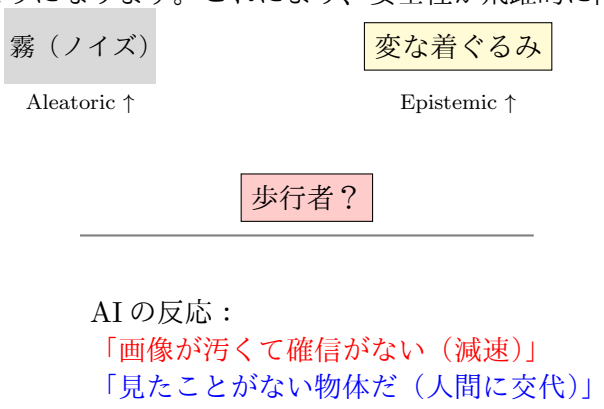
9.6.3 3. なぜこれが凄いの？

- 実装が簡単: 既存の NN モデルにドロップアウト層が入っていれば、予測時に一行書き換える（‘training=True‘ にする）だけで、即座にベイズ NN に早変わりします。
- 計算が現実的: 難しい重みの分布を直接解く代わりに、単に「多数決」を取るだけで不確実性が手に入ります。

9.7 実務での応用イメージ： 自動運転（TikZ）

この 2 種類の不確実性が、実社会でどう役立つかを図解します。

Figure 9.2: **自動運転における不確実性の判断。**ベイズ深層学習を使うことで、AI は単に「何があるか」を答えるだけでなく、**「なぜ確信が持てないのか（画像が悪いのか、知識が足りないのか）」**を区別して報告できるようになります。これにより、安全性が飛躍的に向上します。



このように「不確実性の理由」を明確に分けることは、AI の判断を人間に説明可能（Explainable AI）にするための第一歩でもあります。

9.8 アルゴリズム：MC Dropout Inference

Algorithm 2 MC Dropout Inference

1: 入力 x^* に対し T 回のフォワードパスを実行（Dropout ON）
2: 予測値のサンプリング $\{y_1, \ldots, y_T\}$
3: 平均予測: $\bar{y} = \frac{1}{T}\sum y_t$
4: 不確実性（分散）: $\mathrm{Var}(y) = \frac{1}{T}\sum (y_t - \bar{y})^2 + \tau^{-1}$

9.9 Bayes by Backprop (BBB)

重みの事後分布を平均場変分推論で近似し、再パラメータ化トリック（Reparameterization Trick）を用いて ELBO を最適化します。

$$w = \mu + \sigma \odot \epsilon, \quad \epsilon \sim N(0, I)$$

```
# Pseudo-code for BBB
def elbo_loss(model, x, y):
    # Sample weights
    w_sample = model.sample_weights()
    # Reconstruction loss (Negative Log Likelihood)
    nll = -log_likelihood(y, model.forward(x, w_sample))
    # KL Divergence
    kl = kl_divergence(model.q_w, model.prior_w)
    return nll + kl
```

9.10 練習問題

- MNIST データセットに対し、MC Dropout を用いて数字の分類を行い、不確実（自信がない）画像を抽出せよ。
- 自動運転における歩行者検知タスクで、Epistemic Uncertainty が高い場面（例：霧の中）での挙動を議論せよ。

Part IV

モデル選択と実社会での評価

Chapter 10

モデル比較と情報量基準

どのモデルがデータに最も適しているかを客観的に評価するための基準、特に WAIC と LOO について学びます。

10.1　モデル選択の指標：「未来を当てる力」を測る

第 10 章の「モデル比較と情報量基準」は、ベイズ統計における**「客観的な審判」**の役割を果たします。

良いモデルとは、単に「手元のデータを説明できる」だけでなく、**「まだ見ぬ未来のデータを正確に予測できる」**モデルのことです。これを測る究極の指標が **ELPD** です。

10.1.1　1. ELPD (Expected Log Pointwise Predictive Density)

- **直感的な意味:** 「新しく得られるデータ 1 点 1 点に対して、モデルがどれくらい『正解！』と言い当てられるかの期待値」。
- **詳細:** 将来のデータ $\tilde{y}_i$ に対する対数尤度を、事後分布全体で平均したものです。これが大きいほど、予測力が高いモデルと言えます。

10.2　WAIC：特異なモデルも裁ける「ベイズの基準」

渡辺澄夫教授が開発した WAIC は、従来の基準（AIC など）では扱えなかった複雑なモデル（深層学習や階層モデル）に対しても、正しく予測力を推定できます。

10.2.1　1. 定義式の解剖

$$\mathrm{WAIC} = -2(\mathrm{lppd} - p_{\mathrm{WAIC}})$$

- **lppd (点予測密度)：**
  - **イメージ:** 「手元のデータ」に対するフィット感。高いほど良い。
- **$p_{\mathbf{WAIC}}$ (有効パラメータ数)：**
  - **イメージ:** 「カンニング（過学習）への罰金」。
  - モデルが複雑すぎて手元のデータに合わせすぎている場合、この「罰金」が大きくなり、WAIC の値が悪化（大きく）します。

10.3 PSIS-LOO： 異常値に強い「最強の審判」

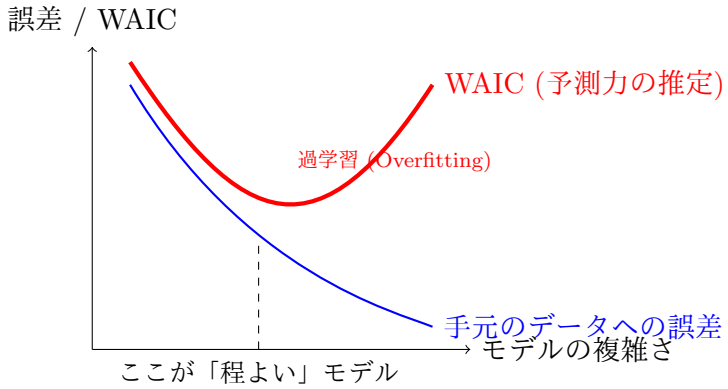
LOO（Leave-One-Out）は、「1 点だけデータを除いて学習し、その 1 点を当てられるか」を全データで繰り返す、非常に厳格なテストです。しかし、計算が大変すぎるため、**PSIS（パレート平滑化重点サンプリング）**という技術を使って高速・頑強に近似したものが PSIS-LOO です。

10.3.1 1. なぜ PSIS-LOO が推奨されるのか？

- **異常値の検知**: WAIC は、極端な異常値（はずれ値）があるときに、計算が不安定になることがあります。
- **警告機能**: PSIS-LOO は、近似がうまくいっていないデータ点に対して **「警告（k 値）」** を出してくれます。
  - $k > 0.7$ の点がある場合、「このデータはモデルにとって無理がある（予測が信用できない）」ことが一目でわかります。

10.4 概念図： 過学習のペナルティ (TikZ)

Figure 10.1: **モデル選択のジレンマ。**モデルを複雑にすればするほど、手元のデータには完璧にフィットしますが（青線）、ある一線を越えると未来の予測力（赤線＝ WAIC）は逆に落ちてしまいます。WAIC はこの「一番美味しい（予測力が高い）ポイント」を教えてくれるコンパスなのです。



まとめ：使い分けの指針

1. **基本的には PSIS-LOO を見る**: 近代的なベイズ実務（PyMC や ArviZ）では、最も頑強な LOO が第一選択です。
2. **警告が出たら WAIC と比較する**: LOO で警告（k 値が高い）が出た場合、モデルの構造やデータの質を見直すチャンスです。
3. **「小さい方が正義」**: どちらの指標も、値が小さいモデルほど「未来を当てる力が高い」と判断されます。

これらの基準があることで、私たちは「自分のモデルが正しい」と主観で言い張るのではなく、データに基づいて客観的にモデルを選べるようになるのです。

10.5 実装例：ArviZ

Python の ArviZ ライブラリを用いると、PyMC や Stan のトレースオブジェクトから簡単に計算できます。

```
import arviz as az

# waic calculation
waic = az.waic(trace, pointwise=True)

# loo calculation
loo = az.loo(trace, pointwise=True)

# model comparison
comparison = az.compare({'model_A': trace_A, 'model_B': trace_B})
print(comparison)
```

10.6 練習問題

- 階層モデルと非階層モデルに対し、WAIC を計算してモデル選択を行え。
- 事前分布の分散を変えたときの WAIC の感度解析を行え。

Chapter 11

実務における意思決定

第 11 章「実務における意思決定」は、これまでの「推論（当てること）」を**「利益を最大化する行動」**に結びつける、最もエキサイティングなパートです。

予測分布 $P(y_{\text{new}}|D)$ を得た後、どのようなアクションを取るべきか？統計的決定理論に基づき、各トピックについて、ビジネスや研究の現場でどのように使われるのか、具体的かつ詳細に解説します。

11.1　期待損失最小化：「賢い妥協点」を探す

ベイズ統計のゴールは確率を出すことではなく、**「損を最小限に抑える行動」**を選ぶことです。

- **詳細**: どんな予測も 100% 当たることはありません。そこで、「外れたときにどれくらい痛いか（損失）」をあらかじめ決めておきます。
- **考え方**:
  1. 各パターンの起きる確率を計算する。
  2. そのパターンが起きた時の「損失額」を掛ける。
  3. 合計が一番小さくなるボタンを押す。
- **ビジネス例**: 「在庫を 100 個持つか、200 個持つか」。
  - 足りない損失（機会損失）と、余る損失（廃棄ロス）を天秤にかけて、期待損失が最小になる個数を導き出します。

最適な行動 a^* は、事後予測分布の下での期待損失を最小化するものです。

$$a^* = \arg\min_{a \in A} \mathbb{E}_{P(y|D)}[L(a, y)] = \arg\min_{a \in A} \int L(a, y) P(y|D) dy$$

11.2　例題：新薬承認問題（リスク管理の真髄）

画像にある例題を深掘りします。

- **状況**: 新薬の効果 δ が 10 以上なら承認したいが、不確実性がある。
- **損失の設計**:
  - $L(\text{承認}, \delta)$: 効果があるのに「却下」した損失 $\rightarrow$ 多くの患者を救えなかった社会的・経済的ロス。
  - $L(\text{却下}, \delta)$: 効果がないのに「承認」した損失 $\rightarrow$ 副作用のリスクと無駄な医療費。

- **ベイズの凄み**: 単なる平均値だけでなく、**「最悪のケース（分布の裾）」**まで考慮して承認可否を判断できるため、より安全な意思決定が可能になります。

例 11.1.
- 行動：承認する、却下する
- 損失関数：
  - $L(\text{承認}, \delta) = -1000\delta + 5000$ （δ は薬効、*5000* はコスト）
  - $L(\text{却下}, \delta) = 0$
- 事後分布：$\delta \sim N(12.5, 3.2^2)$

期待損失を比較して決定を行います。

11.3 ベイズ A/B テスト：「どっちがいい？」に終止符を打つ

従来の A/B テスト（頻度論）は「有意差が出るまで待つ」という受動的な姿勢でしたが、ベイズ版はより能動的です。

- **期待リフト値の評価**: 「B の方が平均 5% 売上が高い」だけでなく、**「B を採用したときに、A より損をする確率は 1% 未満だ」**といった確信度を出せます。
- **柔軟な意思決定**: 「有意差（p 値）はまだ出ていないが、現状のデータだけでも B を選んだほうが期待収益が高いので、今すぐ切り替える」といった、**ビジネススピードに合わせた判断**が可能になります。

11.4 多腕バンディットと Thompson Sampling

「最高の広告」を探しながら、同時に「今すぐ売上も上げたい」という、**探索と活用の両立**を自動化するアルゴリズムです。

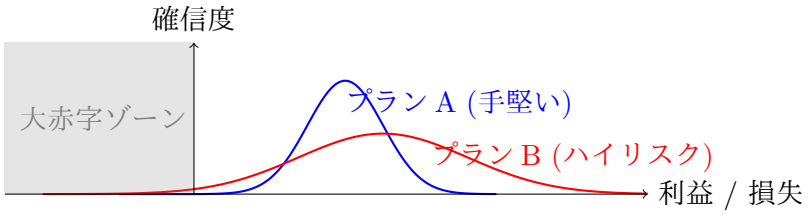
- **多腕バンディット（Multi-armed Bandit）**：スロットマシンのレバー（腕）が複数ある状態。どのレバーが一番当たるかわからない。
- **Thompson Sampling**:
  1. 各スロットの当たりやすさをベイズで推定（分布を持つ）。
  2. 推定された分布から、ランダムに「期待値」を一つずつ妄想（サンプリング）する。
  3. その時、一番強そうなレバーを引く。
  4. 結果を見て、そのレバーの推定をアップデートする。
- **なぜ最強なのか**: 自信がある（尖った分布）レバーは確実に引きつつ、まだよく知らない（広がった分布）レバーも**「たまに化ける可能性」を考慮して適度に試してくれる**ため、人間が調整しなくても勝手に最適化が進みます。

Algorithm 3 Thompson Sampling

for $t = 1, 2, \ldots$ **do**
  各アーム k について、事後分布からパラメータ $\theta_k \sim P(\theta_k | D_{t-1})$ をサンプリング
  最大の θ_k を持つアーム k^* を選択
  アーム k^* を実行し、報酬 r を観測
  事後分布 $P(\theta_{k^*} | D)$ を更新
end for

11.5 TikZ による図解： リスク調整後の意思決定

Figure 11.1: **期待値だけで選ばないベイズの選択。**プラン B は平均利益（山の頂上）こそ A より高いですが、赤字になる確率（灰色の領域にかかっている面積）も大きいです。ベイズ統計では、これら「失敗するリスク」を損失関数として組み込むことで、組織のポリシーに合わせた最適な一手を選び出すことができます。



まとめ：実務家へのアドバイス

1. **「当てる」ことの先を考える**: 推論結果が出たら、必ず「もし間違えたら？」という損失をセットで議論しましょう。
2. **Thompson Sampling を活用する**: 広告や推薦システムなどの「正解が常に変わる」現場では、バンディットアルゴリズムが絶大な威力を発揮します。
3. **確信度を言葉にする**: 「確率は 50%です」と言う代わりに、「これを選ぶ期待損失は○○円で、プラン B より○○円お得です」と伝えるのが、ベイズ的な意思決定の作法です。

11.6 練習問題

- **在庫管理**: 需要 $D \sim \text{Poisson}(\lambda)$（$\lambda$ は事後分布に従う）の下で、売れ残りコストと品切れコストを考慮した最適発注量を求めよ。

Chapter 12

ベイズ的実験計画（Optimal Experimental Design）

第 12 章「ベイズ的実験計画（Optimal Experimental Design）」は、ベイズ統計の知識を「守り（解析）」から「攻め（実験の設計）」へと転換させる、非常に実用的なパートです。

限られた予算・実験回数で最大の情報を得るための、実験条件の最適化（Active Learning）を学びます。

12.1 設計基準：「一番おいしい情報」を狙い撃つ

実験にはコスト（お金、時間、材料）がかかります。ベイズ的実験計画の目的は、**「次にどこを調べれば、今のモデルが一番賢くなるか」**を計算することです。

12.1.1 1. 期待情報利得（Expected Information Gain: EIG）

- **直感的な意味:** 「その実験をした後に、どれだけ自分の『無知』が解消されるか」の期待値。
- **詳細:** 事前分布のエントロピー（不確かさ）と、実験後の事後分布のエントロピーの差を最大化します。
- **具体例:** 新素材の配合。
  - すでに結果がわかっている配合の近くを試しても EIG は低いです。
  - 逆に、モデルが「ここは全く想像がつかない」と言っている領域を試すと、EIG は非常に高くなります。

12.2 獲得関数の種類：目的に合わせた「コンパス」

実験の「狙い」によって獲得関数を使い分けます。

12.2.1 1. Uncertainty Sampling（不確実性サンプリング）

- **狙い:** 「とにかく弱点を克服したい」
- **動作:** 予測分散 $\sigma^2(x)$ が最大の場所、つまり「モデルが一番自信を持っていない場所」を次に実験します。
- **用途:** 全体的に精度の高い「地図」を早く完成させたい時に有効です。

12.2.2 2. Mutual Information（相互情報量最大化）

- 狙い: 「パラメータの正体を暴きたい」
- 動作: 観測値を得ることで、パラメータ θ について得られる情報量を最大化します。
- 用途: 現象の「メカニズム（数式）」を特定したい科学研究などで好まれます。

12.3 適応的実験計画（Adaptive Design）：「走りながら考える」

ベイズの実験計画は「逐次的（ループ）」であることに最大の特徴があります。

- 従来の実験（静的）: 最初に「直交表」などで 50 回分の実験計画をすべて決めてしまい、最後までやり抜く。
- ベイズ的実験（適応的）:
  1. 1 回実験する。
  2. その結果でモデルを賢くする（事後分布の更新）。
  3. 「賢くなった頭で」次の最適な 1 回を決める。
- メリット: 途中で「ここは有望じゃない」とわかった領域をスキップできるため、従来の **1/3 〜 1/10 の実験回数**でゴール（最適解）に辿り着けます。

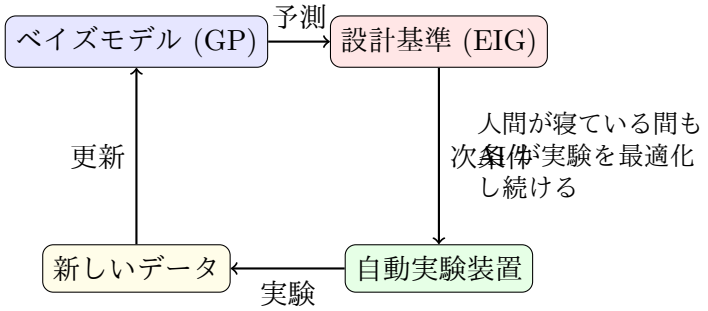
Algorithm 4 Bayesian Adaptive Design Loop

初期実験データ D_0 でモデルを学習
for $t = 1$ to T **do**
  候補集合 $\mathcal{X}_{cand}$ から、獲得関数 $a(x)$ を最大化する x_t を選択
  実験を行い y_t を観測
  データセット $D_t = D_{t-1} \cup \{(x_t, y_t)\}$ でモデル（事後分布）を更新
end for

12.4 実務での応用：材料探索と自動化（TikZ）

ハイスループット実験（自動実験装置）との連携を視覚化します。

Figure 12.1: 自律的材料探索システム。ベイズ的実験計画を自動実験ロボットと直結させることで、研究者は「どの基準で探索するか（EIG 等）」を設計するだけで、AI が自ら仮説検証を繰り返し、最短ルートで新材料を発見します。



まとめ：研究者・実務家へのメッセージ

1. **「失敗」を「情報」に変える**: ベイズ実験計画では、期待外れの結果も「そこではない」という重要な情報になり、無駄な実験が一つもなくなります。
2. **少ないリソースで勝つ**: 予算や時間が限られている時こそ、ベイズの「狙い撃ち」が真価を発揮します。
3. **Active Learning の視点**: これは機械学習の「能動学習」そのものです。AI に「何を聞くべきか」を考えさせることで、知能の進化を加速させましょう。

12.5 練習問題

- 反応条件（温度、圧力）の最適化において、D-最適基準とランダムサンプリングの効率（必要な実験回数）を比較せよ。
- 臨床試験における適応的デザイン（Adaptive Dose Finding）の利点を説明せよ。

Appendix A

練習問題の解答と数理的補足

A.1　第 3 章：共役事前の更新証明

問: $y \sim \mathrm{Bin}(n, \theta)$, $\theta \sim \mathrm{Beta}(\alpha, \beta)$ の事後分布が $\mathrm{Beta}(\alpha + y, \beta + n - y)$ であることを証明せよ。

解答: ベイズの定理に従い、尤度と事前分布の積を計算します。

1. 尤度関数（二項分布）:
$$L(\theta) = \binom{n}{y} \theta^y (1-\theta)^{n-y}$$

2. 事前密度（ベータ分布）:
$$\pi(\theta) = \frac{1}{B(\alpha, \beta)} \theta^{\alpha-1} (1-\theta)^{\beta-1}$$
ここで $B(\alpha, \beta)$ はベータ関数（正規化定数）です。

3. 事後分布の核（Kernel）の計算: 定数部分を無視して θ に依存する項のみを抽出すると、
$$\begin{aligned} p(\theta|y) &\propto L(\theta) \times \pi(\theta) \\ &\propto \theta^y (1-\theta)^{n-y} \times \theta^{\alpha-1} (1-\theta)^{\beta-1} \\ &= \theta^{y+\alpha-1} (1-\theta)^{n-y+\beta-1} \end{aligned}$$

4. 分布の定同: 上記の式は、パラメータ $\alpha' = \alpha + y, \beta' = \beta + n - y$ を持つベータ分布の確率密度関数の形と一致します。したがって、事後分布は $\mathrm{Beta}(\alpha + y, \beta + n - y)$ に従います。（証明終）

A.2　第 11 章：期待損失の最小化

問: 二値分類において、偽陽性（FP）のコストが偽陰性（FN）の 3 倍であるとき、事後確率 $P(y=1|D)$ がいくらを超えれば $y=1$ と判定すべきか。

解答: 統計的決定理論に基づき、期待損失を最小化する決定を行います。行動 $a \in \{0, 1\}$、真の状態 $y \in \{0, 1\}$ とします。

損失関数 $L(a, y)$ を以下のように設定します（正解時は損失 0 とします）。

- $L(a=1, y=0) = 3$ （偽陽性 FP）
- $L(a=0, y=1) = 1$ （偽陰性 FN）
- $L(a=0, y=0) = 0, L(a=1, y=1) = 0$

事後確率を $p = P(y=1|D)$ と置くと、$P(y=0|D) = 1 - p$ です。
各行動に対する期待損失 $E[L(a)]$ を計算します。

- 行動 $a = 1$（陽性と判定）の場合:

$$E[L(a=1)] = L(1,1)p + L(1,0)(1-p) = 0 \times p + 3 \times (1-p) = 3(1-p)$$

- 行動 $a = 0$（陰性と判定）の場合:

$$E[L(a=0)] = L(0,1)p + L(0,0)(1-p) = 1 \times p + 0 \times (1-p) = p$$

行動 $a = 1$ を選択すべき条件は、その期待損失が $a = 0$ の場合より小さいことです。

$$\begin{aligned} E[L(a=1)] &< E[L(a=0)] \\ 3(1-p) &< p \\ 3 - 3p &< p \\ 3 &< 4p \\ p &> 0.75 \end{aligned}$$

結論: 事後確率 $P(y=1|D)$ が 75% を超える場合にのみ、$y = 1$ と判定するのが最適です。