数理生物学演習

第11回　数理モデルを用いた植物デンプン代謝の解析
（数理モデルのパラメータ推定）

工藤 秀一
shuichi8040@gmail.com
九州大学システム生命科学府
数理生物学研究室(佐竹グループ)

本日の内容

数理モデルを用いた時系列データの解析
（実験データからモデルのパラメータを推定）

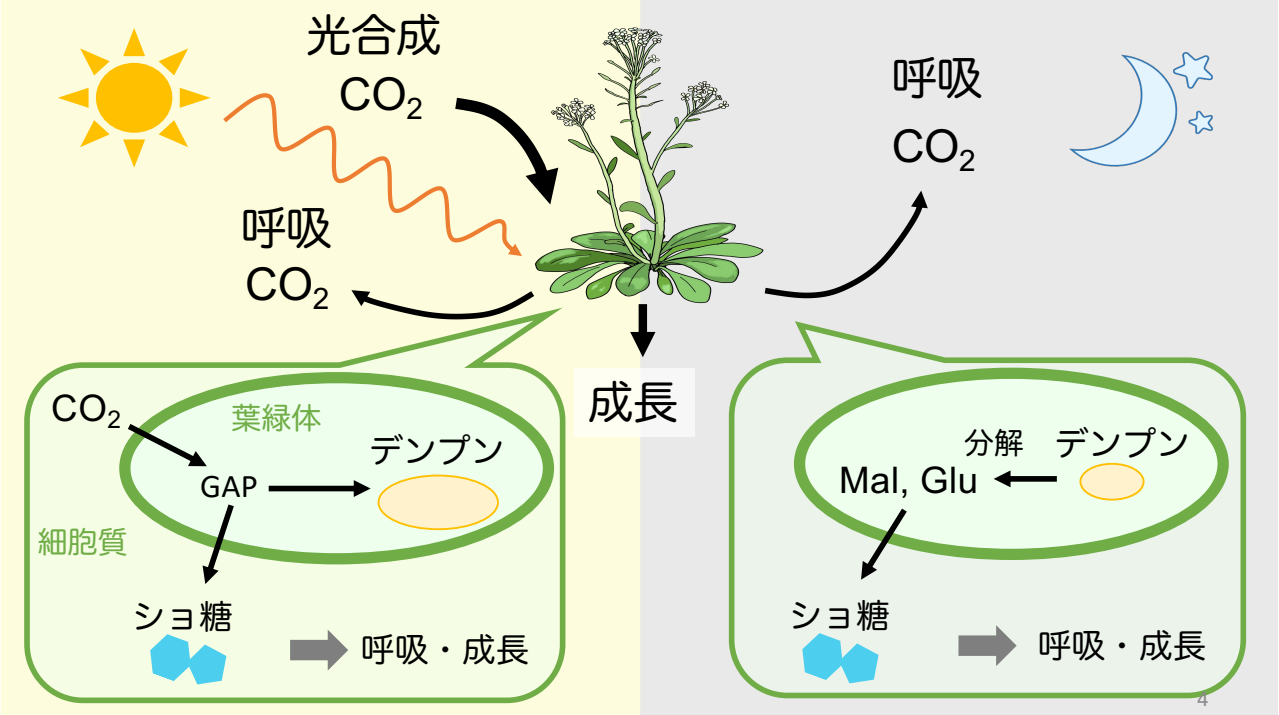
1. 研究紹介
   数理モデルを用いた植物デンプン代謝の解析

2. パラメータ推定
   ベイズ推定の基礎とPyMCによる実装

研究紹介
植物のデンプン代謝の解析

植物のデンプン代謝

多くの植物は夜に備えて葉にデンプンを蓄える



植物のデンプンの役割

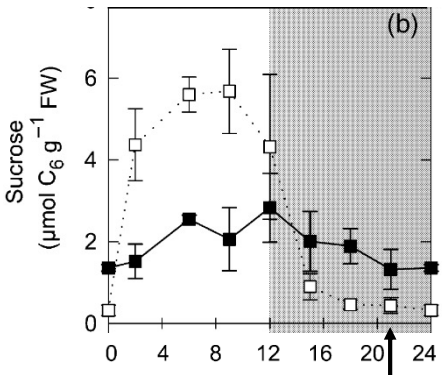
デンプンは昼夜を通したショ糖の変動を抑えるため
のバッファーとして機能する（ショ糖恒常性）

光合成産物

ショ糖　　デンプン

ショ糖が多ければデンプン
にしておく
足りない時はデンプンを
分解して補充する

葉のショ糖の日内変動



Gibon et al (2004)

デンプンが無い
と昼夜の変動が
大きい

□ *pgm*
デンプンが合成
できない

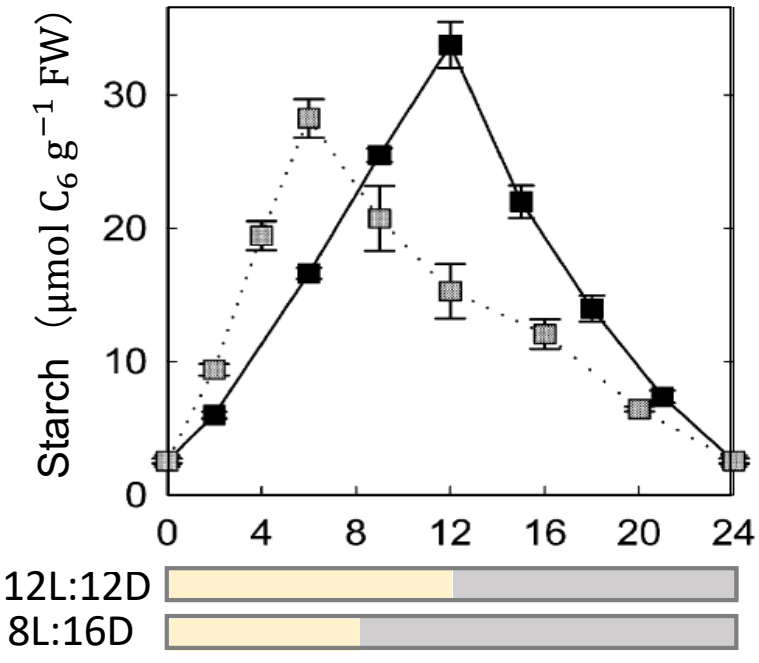
■ Col-0
野生型

夜間のショ糖枯渇は成長を阻害する

継続的なショ糖の供給により安定した成長を維持できる

デンプン量の日周変化

1日を通して葉のデンプンは直線的に増減する



日長

■ 12L：12D

■ 8L：16D

異なる日長の下でも
過不足なくデンプン
を使う(日長応答)

どうやって??

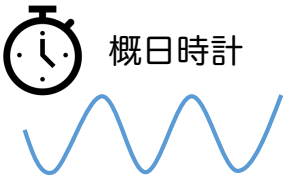
Gibon et al (2004)

デンプン分解速度はどう制御されるか?

デンプンを夜の長さに応じて過不足なく分解するためには…

時間と量の情報が必要
(どれくらいの時間でどのくらい分解するのか)

時間に関する情報



分解量に関する情報

デンプンを測る?

ショ糖を測る?

デンプン分解速度は概日時計によって制御されている

- 時計の変異体はデンプン分解を調節できない

デンプン分解速度はおそらくショ糖によって制御されている

- ショ糖が過剰になるとデンプン分解が抑制される
- 水溶性のショ糖の方が適している

デンプン代謝の数理モデル

適切なデンプン分解速度はどのように決まるのか?

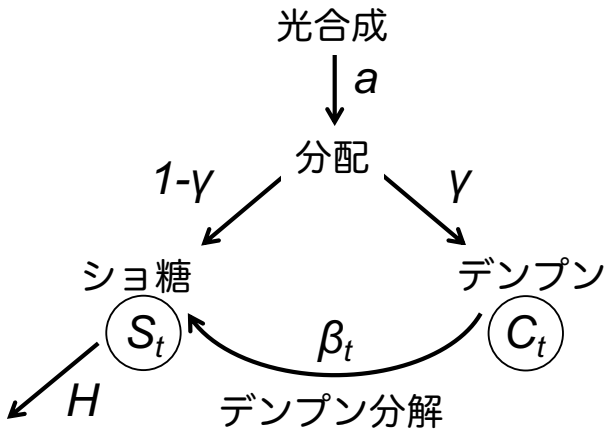
考えられるメカニズム

(1) 植物はショ糖の変化を感知し、ショ糖量を一定に保とうとする
(2) 概日時計によって、デンプン分解速度は時間依存的(時間の関数)に制御される

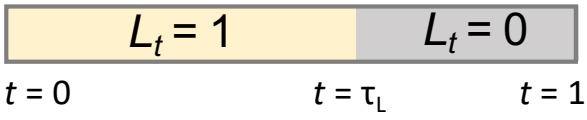
↓

仮説をもとに数理モデルを立ててみる

デンプン代謝の数理モデル



光条件



デンプンの時間変化

$$\frac{dC_t}{dt} = \underbrace{a\gamma L_t}_{\text{光合成}} - \underbrace{\beta_t C_t^{\kappa}}_{\text{デンプン分解}}$$

κ : デンプン粒の形状を表す ($\kappa = 2/3$ 球状)

ショ糖の時間変化

$$\frac{dS_t}{dt} = \underbrace{a(1-\gamma)L_t}_{\text{光合成}} + \underbrace{\beta_t C_t^{\kappa}}_{\text{デンプン分解}} - \underbrace{HS_t}_{\text{輸送・呼吸}}$$

光の状態

$$L_t = \begin{cases} 1 & (0 \le t \le \tau_L) \quad \text{昼} \\ 0 & (\tau_L < t \le 1) \quad \text{夜} \end{cases}$$

⟶ 理想的なβ_tの形は？

デンプン代謝の数理モデル

日長T_Lの下でショ糖が一定となるようなβ_tを求める

デンプンの時間変化

$$\frac{dC_t}{dt} = a\gamma L_t - \beta_t C_t^{\kappa}$$

ショ糖の時間変化

$$\frac{dS_t}{dt} = a(1-\gamma)L_t + \beta_t C_t^{\kappa} - HS_t$$

⟶ ショ糖が一定、つまり$\frac{dS_t}{dt} = 0$とおいてβ_tについて解く

これを計算すると最終的に以下が示せる

$$\frac{dS}{dt} = 0 \iff \frac{dC}{dt} = aL_t - HS_0$$

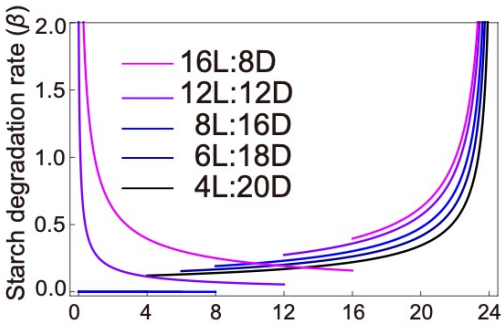
ショ糖の恒常性　　デンプン増減の直線性

$$\beta_t = \frac{a(\gamma - 1 + \tau_L)}{\{a(1-\tau_L)t + C_0\}^{\kappa}} L_t + \frac{a\tau_L}{\{a\tau_L(1-t) + C_0\}^{\kappa}}(1 - L_t)$$

理想的なデンプン分解速度

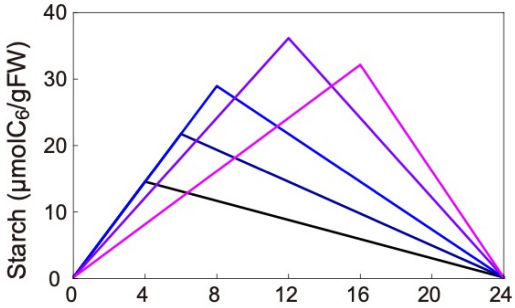
デンプン代謝の数理モデル

日長ごとの理想的なデンプン分解速度

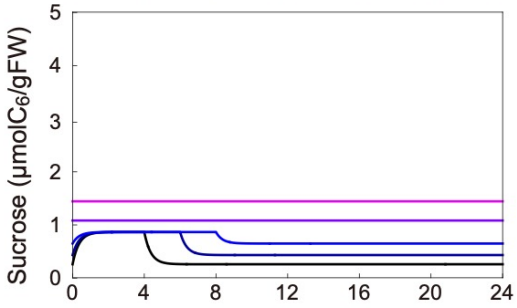


$$\beta_t = \frac{a(\gamma-1+\tau_L)}{\{a(1-\tau_L)t+C_0\}^\kappa} L_t + \frac{a\tau_L}{\{a\tau_L(1-t)+C_0\}^\kappa}(1-L_t)$$

デンプンの直線的な増減



ショ糖の変動最小



パラメータを変えることで様々なデンプン代謝の状態を表現できる

デンプン代謝モデルをデータ解析に応用

生物学的な要請
デンプン分解を制御する分子メカニズムが知りたい
候補遺伝子の機能を(例えば変異体を作って)評価したい

植物分子生理学者Dr. Camila Caldana (ドイツ Max Planck Institute)との共同研究

デンプン分解制御に関わる候補遺伝子の変異体を使ってデンプンとショ糖の濃度を測定し、時系列データを取得

↓

数理モデルを使って実験データから各変異体のデンプン・ショ糖代謝の状態、日長応答の有無を推定し、各遺伝子の寄与を評価する

数理モデルは役に立つのか？

数理モデルをデータ解析に用いる利点

- データを解析するための視点を与える
  (例：日長応答を見るためにショ糖の変動に注目する)
- 定量的に評価できる
  (例：変異体Aでは野生型より2時間日長がずれている)

ただし注意も必要

真に正しいモデルは存在しない

数理モデルはあくまでも現実の系を近似したもの

自分のデータに適用できるかどうか考える必要がある

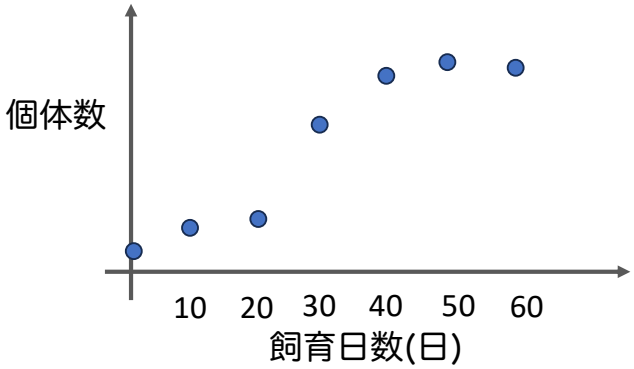
数理モデルのパラメータ推定

データをうまく説明するパラメータは？

ある数理モデルで説明される現象について実験データが得られたとする

実験データからパラメータを推定できるか?

例: ロジスティック増殖する昆虫の個体数



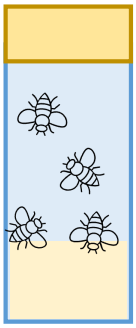
ロジスティックモデル

$$\frac{dx}{dt} = rx\left(1 - \frac{x}{K}\right)$$

生物学的に重要な情報

自然増加率 r ＝?

環境収容力 K ＝?



パラメータ推定の方法

色々ある

最適なパラメータを求める(評価関数＋最適化)

- 最小二乗法：データと予測の二乗誤差を最小化
- 最尤推定法：尤度を最大化

パラメータの分布を求める

- ベイズ推定：パラメータの確率分布を推定

ベイズ推定

データXが得られたという条件の下で，パラメータがある値θをとる条件付き確率$P(\theta|X)$を計算する

ベイズ推定の特徴

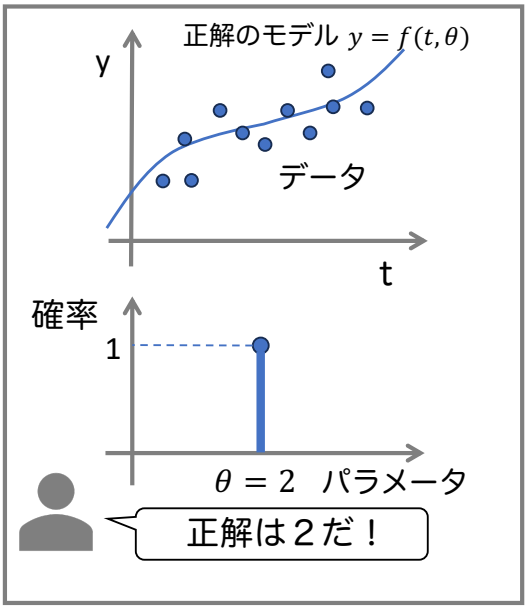
(1)パラメータの値そのものではなくパラメータが特定の値をとる確率(分布)を求める

(2) パラメータに関する事前知識を考慮する

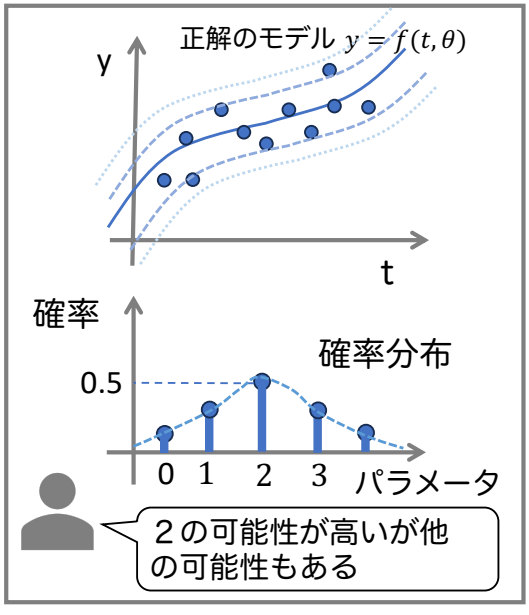
ベイズ推定

(1) パラメータの値そのものではなくパラメータが特定の値をとる確率分布を求める

正解を一つに定めるのではなく…



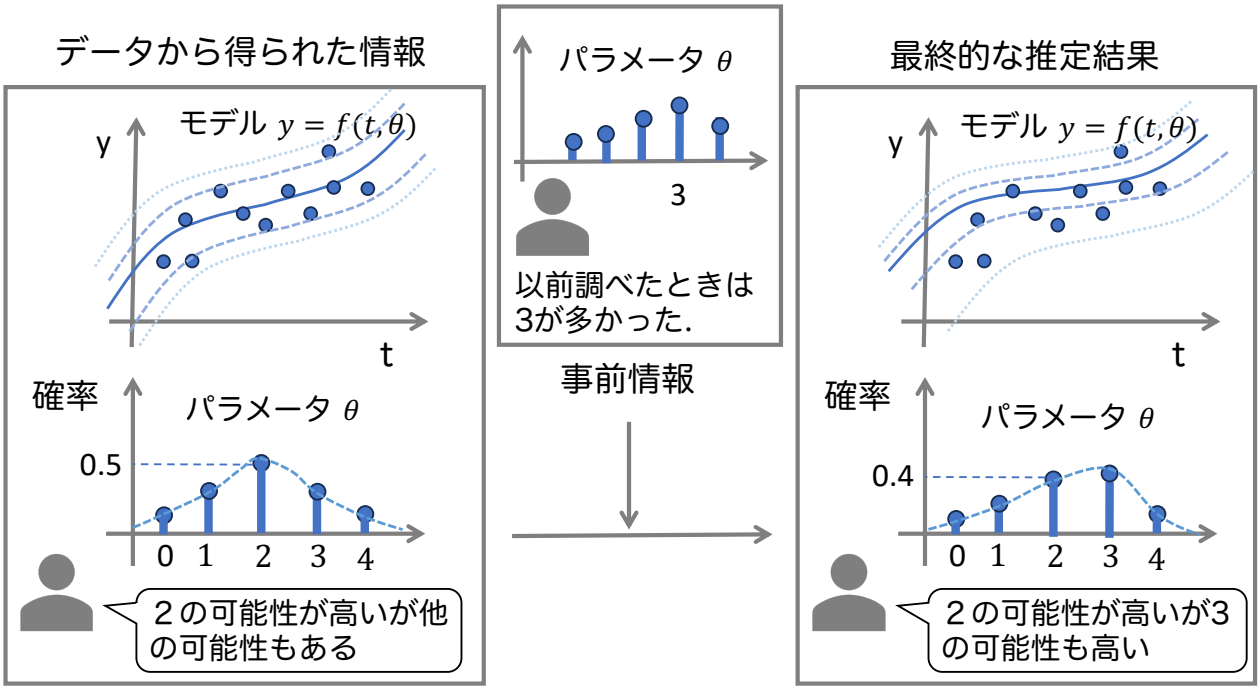
不確実性も含めた分布として推定する



推定した結果がどの程度の不確実性を含んでいるのかが分かる

ベイズ推定

(2) パラメータに関する事前の知識を考慮する



過去の経験を加味したパラメータの推定ができる

ベイズ推定

ベイズ推定では事前の情報やデータから得られる情報を確率分布を使って表現する

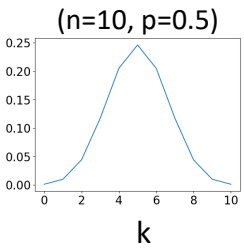
確率分布 $P(X)$：確率変数がある値をとる確率を定める関数.

確率変数Xを代入すると対応する確率(密度)を返す.
パラメータを持つときは$P(X; \ldots)$と書く.

二項分布

$Binomial(k; n, p)$

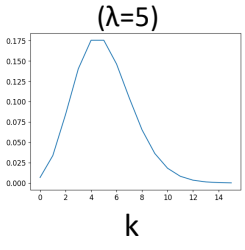
確率pで表が出るコインをn回投げたとき表がk回出る確率



ポワソン分布

$Poisson(x; \lambda)$

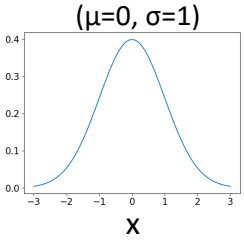
1日に平均λ回発生する事故が1日にk回発生する確率



正規分布

$Normal(x; \mu, \sigma)$

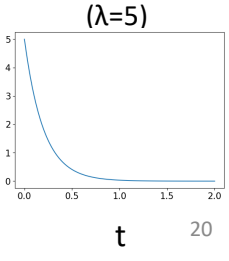
最もよく使われる連続型確率分布
実験誤差、身長など



指数分布

$Exponential(x; \lambda)$

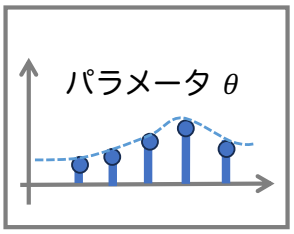
1日に平均λ回発生する事故がt日後に初めて発生する確率



ベイズ推定の定式化

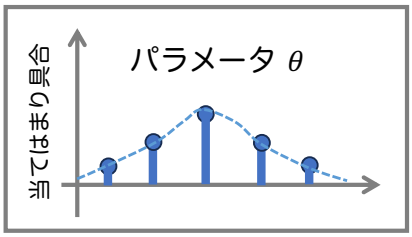
ベイズ推定では事前の情報やデータから得られる
情報を確率分布を使って条件付き確率として表現

事前情報



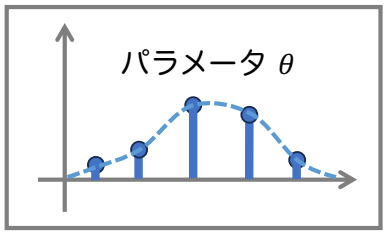
×

データから得られる情報



→

最終的な推定結果



データが得られる前の
パラメータの確率分布

$P(\theta)$

事前分布

とりうるパラメータごとに
データが得られる確率を計算

$P(X|\theta)$

尤度

データと前提条件を考慮
したパラメータの確率分布

$P(\theta|X)$

事後分布

条件付き確率に関する
ベイズの定理　$P(\theta|X) = P(\theta)P(X|\theta)/P(X)$

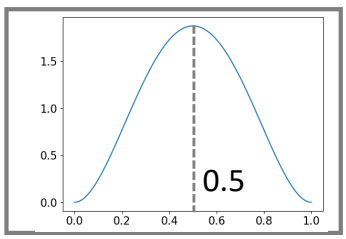
$P(X)$は定数なので分布の形に影響しない

簡単な例 コイン投げ

例題1

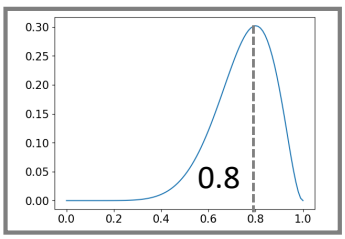
ある工場で作られたコインを10回投げたところ8回表が出た.
このデータに基づいてコインの表が出る確率θの事後確率
分布を求めよ.
ただし経験上θは0.5程度であることが多い.

事前情報



×

データとの当てはまり具合



→

最終的な推定結果

?

ベータ分布を仮定

$Beta(\theta; \alpha = 3, \beta = 3)$

※ベータ分布は二項分布と
セットでよく使われる
表:裏が$\alpha - 1 : \beta - 1$の時のθ
の分布と解釈できる

二項分布を使う

$Binomial(k = 8; n = 10, \theta)$
$= {}_{10}C_8\, \theta^8 (1-\theta)^2$

表が出る確率θのコインを
n回投げてk回表が出る確率

???

※本来は回数kの関数だが、
ここではパラメータθの関数
として見る=尤度関数

簡単な例 コイン投げ

Pythonを使って事後分布を推定してみよう

確率分布の計算にはSciPyのstats
モジュールを利用する

scipy.stats
科学系の計算用のライブラリSciPyに
含まれる統計関数を集めたモジュール

パッケージの読み込み

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

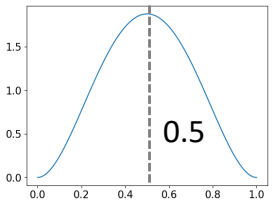
scipy全部ではなくstatsだけ
をimportする

事前分布の設定と可視化

```
#事前分布
theta = np.linspace(0,1,101)
prior = stats.beta.pdf(theta, 3, 3)
plt.plot(theta, prior)
plt.show()
```

0から1までを0.01刻みで
101個入れた配列

ベータ分布を仮定



$Beta(\theta;\alpha,\beta)$

表:裏が$\alpha-1:\beta-1$の時
のθの分布と解釈できる

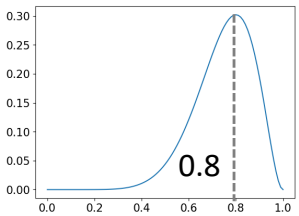
簡単な例 コイン投げ

Pythonを使って事後分布を推定してみよう

尤度の計算と可視化

```
#二項分布による尤度の計算
k = 8
n = 10
likelihood = stats.binom.pmf(k, n, theta)
plt.plot(theta, likelihood)
plt.show()
```

二項分布

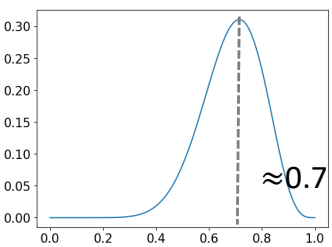


$${}_nC_k\,\theta^k(1-\theta)^{n-k}$$

θのコインをn回投げて
k回表が出る確率

事後分布の計算と可視化

```
#事後分布(定数倍は無視)
posterior = prior * likelihood
plt.plot(theta, posterior)
plt.show()
```

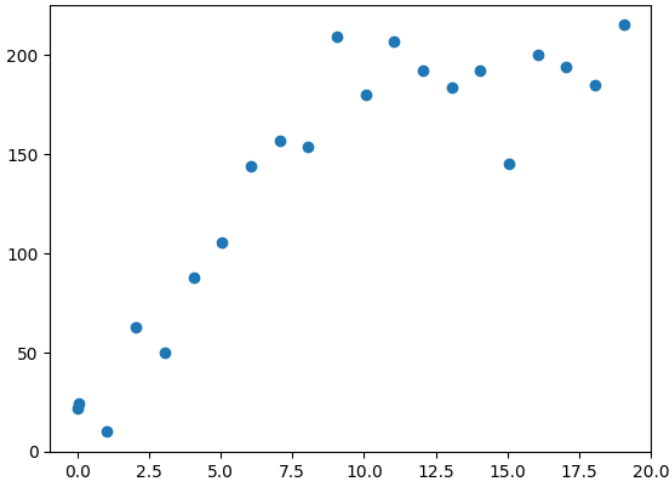


事前分布と尤度の中間くらいにピーク

本題：微分方程式のパラメータ推定

ある生物の個体数変化を調べると図のような結果が得られた.

この生物の個体数変化がロジスティックモデルに当てはめることができるとすると、パラメータr, Kを推定できるか?



ロジスティックモデル

$$\frac{dx}{dt} = rx\left(1 - \frac{x}{K}\right)$$

生物学的に重要な情報

自然増加率 r = ?

環境収容力 K = ?

やることは同じ. 事前分布と尤度関数を定義する!

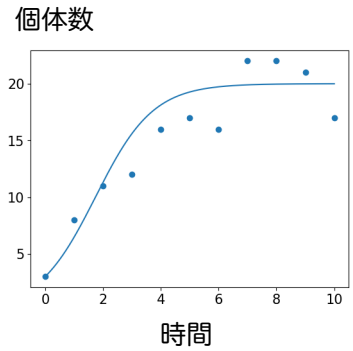
ベイズ推定で微分方程式のパラメータ推定

どうやって尤度を定義するか?

データのばらつきを確率モデルで表現する

データ = モデルの理論値 + 確率的なばらつき

確率モデルは問題に応じて選ぶ
例えば正規分布やポワソン分布を使う



ここでは正規分布を使って表現してみる (最適ではないかもしれない)

時刻t_n における個体数X_nは平均$x(t_n)$, 標準偏差σ_xの正規分布に従う

微分方程式の解　ばらつきの強さ

尤度 $P(X_n|r, K, x_0, \sigma_x) = Normal(X_n; \mu = x(t_n, r, K, x_0), \sigma = \sigma_x)$

計算が大変なので専用のパッケージを使おう

ベイズ推定用ライブラリを利用する

PyStan
Stanと呼ばれるベイズ推定に特化した言語をPythonから扱うためのライブラリ

stanコードの例(コイン投げ)

```
stan_model = """
data {
int n;
int X;
}
parameters {
real<lower=0, upper=1> p;
}
model {
p ~ beta(7,3);
X ~ binomial(n,p);
}
"""
```

基本的な使い方

(1)ライブラリのインポート
(2)stanコードを作成
(3)stanに入れるデータを定義
(4)stanを実行
(5)結果の表示

データブロック、パラメータブロック、モデルブロックなど、ブロックごとに書いていく

ベイズ推定で微分方程式のパラメータ推定

サンプルコード(pystan_logistic.ipynb)をダウンロードし実行してみよう

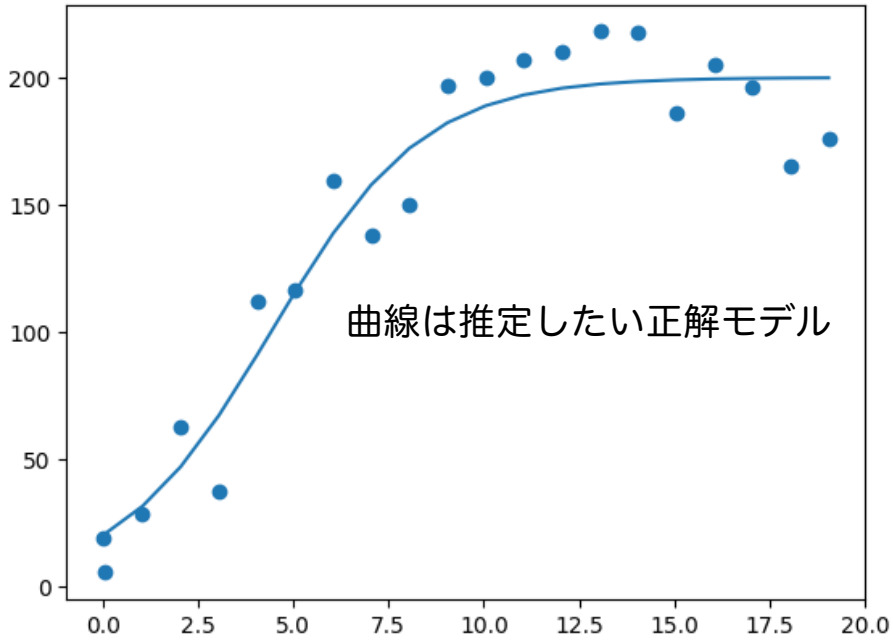
以下のリンクからダウンロードし, Google driveにアップロードして開く
https://github.com/shuichi-kudo/Computational-Biology2023

サンプルコードの中身

正解パラメータ設定
↓
正解パラメータから擬似データを作成
↓
stanを使ってパラメータ推定
↓
推定結果の可視化

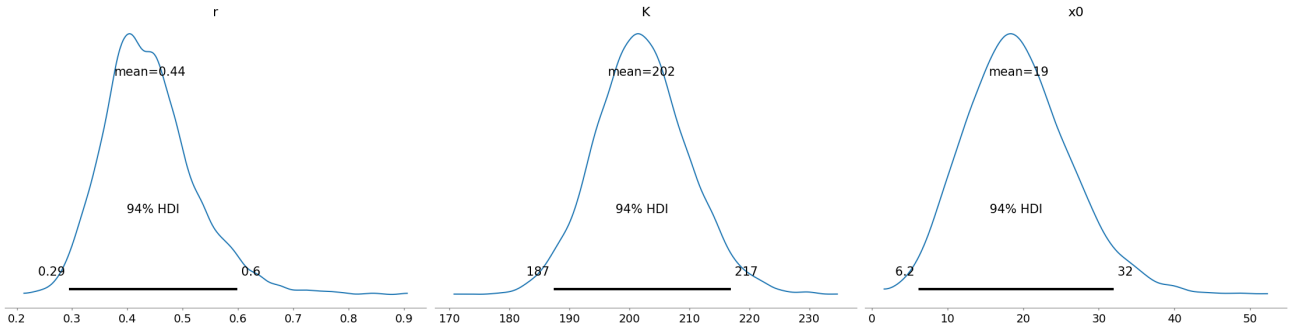
ベイズ推定で微分方程式のパラメータ推定

擬似データを作成

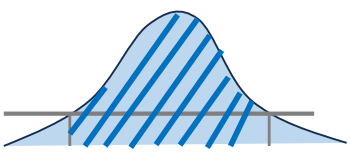


ベイズ推定で微分方程式のパラメータ推定

最終的な出力の一例



平均meanと94%High Density Interval (HDI)が表示される



左の交点から右の交点までの区間の面積が全体の94%になる高さ

ベイズ推定の注意点

尤度関数や事前確率分布の設定など柔軟なモデリングが可能な反面、解析者が決めないといけない要素が多い.
また、複雑なモデルになるほど推定が不安定になる.

よくある問題
- 事前の情報が限られているときに事前分布をどう設定するか?　恣意的でないか?
- パラメータのとり方は適切か？
- 持っているデータはパラメータ決定に十分な情報を持っているか?

問題・データに応じて適切なモデリングを行う必要がある

本日の課題　ノーマル

1. 10回コインを投げたら8回表が出た。このコインの表が出る確率θについて、事前分布をベータ分布$Beta(\theta, \alpha = 3, \beta = 2)$とし事前分布、尤度関数、事後分布をそれぞれプロットせよ.
2. 100回コインを投げたら80回表が出た場合について同じ事前分布$Beta(\theta, \alpha = 3, \beta = 2)$を使い、事前分布、尤度関数、事後分布をそれぞれプロットせよ.
3. 2.のとき事後分布はどのように変化したか。事前分布が推定結果に与える影響の観点から考察せよ.
4. 質問, 感想, 要望をどうぞ.

課題をノートブック(.ipynbファイル)にまとめて，Moodleにて提出すること
ファイル名は[回数，01~15]_[難易度，ノーマル nかハード h].ipynb.例.11_n.ipynb

本日の課題　ハード

1. サンプルコード(pystan_logistic.ipynb)を実行し、ロジスティックモデルのベイズ推定を行い, 各パラメータの事後分布をプロットせよ.
2. 1.で真のパラメータセットを色々変えて, どのような時に推定がうまくいかないことがあるか調べよ.
3. 推定がうまくいかない理由について考察せよ.
4. 質問, 感想, 要望をどうぞ.

課題をノートブック(.ipynbファイル)にまとめて，Moodleにて提出すること
ファイル名は[回数，01~15]_[難易度，ノーマル nかハード h].ipynb.例.11_h.ipynb

乱数を使った事後分布の計算

パラメータ数が多くなった場合や複雑なモデリングを
行った場合、ベイズの定理から直接事後分布を計算する
ことは困難になる

↓

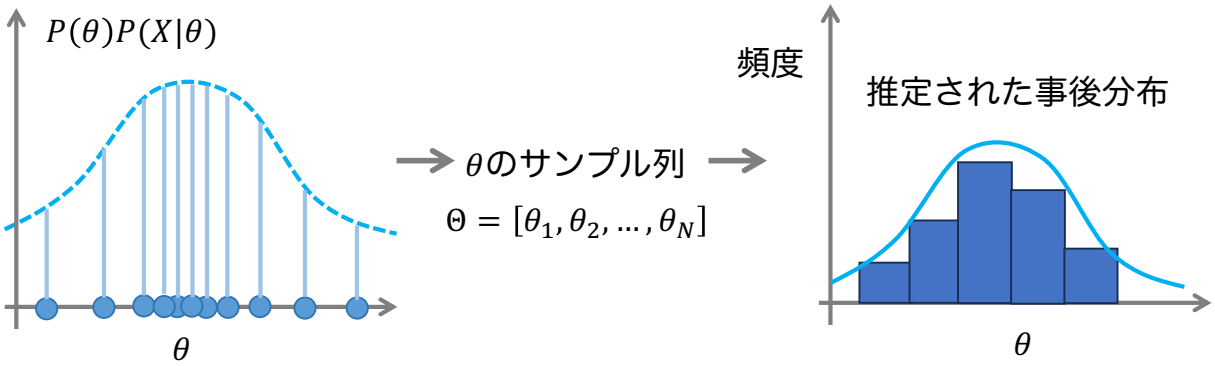
乱数を使って事後分布に従うパラメータのサンプリング
を行い、サンプルから分布の平均や広がりを計算する
(マルコフ連鎖モンテカルロ法(MCMC))

ベイズ推定を専門に行うソフトウェア(PyStan・PyMC)では
MCMCやその派生系が実装されている

マルコフ連鎖モンテカルロ法(MCMC)

乱数を使って複雑な事後分布を計算する

$P(\theta)P(X|\theta)$の全体像は分からない (代入によって局所的な情報は分かる)
θをランダムに生成し, $P(\theta)P(X|\theta)$が大きいものほどたくさん記録する



θのサンプリングはθのとりうる範囲内を歩き回るようにして行う
(1) 最初のサンプルをθ_0とする
(2) θ_0から少し離れたパラメータをランダムにとり次の候補θ^*とする
(3) θ_0とθ^*で$P(\theta)P(X|\theta)$の値を比較し候補θ^*を採択するか(確率的に)決める
(4) (2),(3)を繰り返す