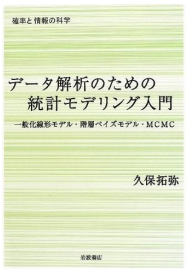
統計モデリング入門 2016 (a)

An Introduction to Statistical Modeling

観測されたパターンを説明する統計モデル

久保拓弥（北海道大・環境科学）
kubo@ees.hokudai.ac.jp



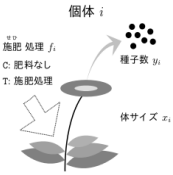


図 3.1 この例題に登場する架空植物の第 i 番目の個体．この植物の体サイズ（個体の大きさ）x_i と肥料をやる施肥処理 f_i が種子数 y_i にどう影響しているのかを知りたい．

この授業の目的:

「データにあわせて

統計モデルを作る」

…という考えかたに慣れる

「脱」ぶらっくぼっくす統計！

「統計モデル」って何?

内容がわからなくてもソフトウェアにまるなげ

“人工知能”？

↑

その実態：機械学習?

↑

その部品の一部：「統計モデル」

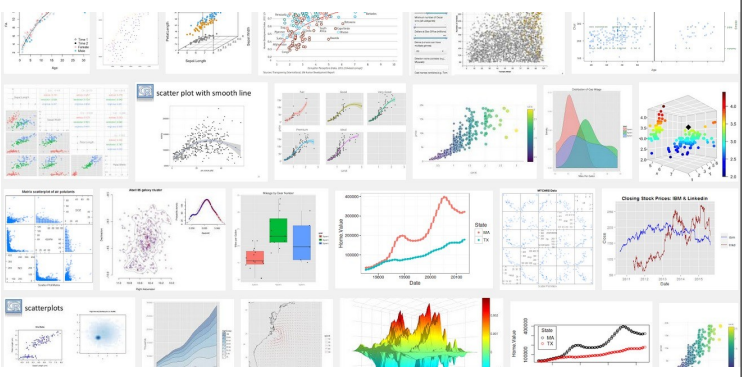
…ぐらいの理解でよいかと…

データ解析で

もっとも重要なことは?

統計モデル?…必ずしもそうではない

データを図示すること!!



google 画像検索の結果の一例

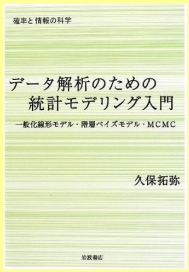
作図のないデータ解析はありえない!

じゃ，データ図示の授業やったら?

- うーむ…作図は art?
  - 自分の中では体系化されていない
  - ダメな作図は指摘できる
  - よい作図の方針はよくわからない
- 統計モデリングは science
  - 簡単なものから高度なものへステップアップ
  - 何がダメか, 比較的明瞭

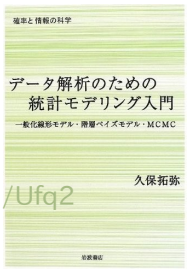
ただし，明日は作図の練習ですよ！

教科書とソフトウェア

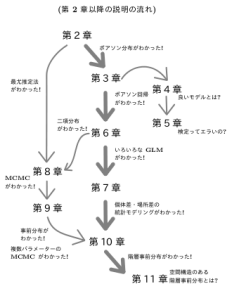


この授業は「統計モデリング入門」にそった内容を説明します

著者：久保拓弥
出版社：岩波書店
2012-05-18 刊行
価格 3990 円



http://goo.gl/Ufq2



割引販売 3000 円!!

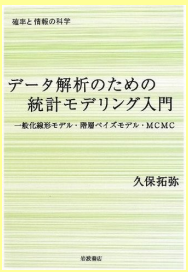
統計ソフトウェア R

統計学の勉強には良い統計ソフトウェアが必要!

- 無料で入手できる
- 内容が完全に公開されている
- 多くの研究者が使っている
- 作図機能が強力

この教科書でも R を使って問題を解決する方法を説明しています

統計モデルとは何か?



「統計モデル」とは何か?

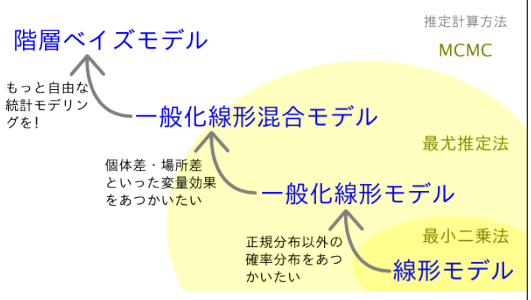
どんな統計解析においても
統計モデルが使用されている

- 観察によってデータ化された現象を説明するために作られる
- 確率分布が基本的な部品であり，これはデータにみられるばらつきを表現する手段である
- データとモデルを対応づける手つづきが準備されていて，モデルがデータにどれぐらい良くあてはまっているかを定量的に評価できる

「統計モデリング入門」の主張

「何でも正規分布」じゃないだろ!

線形モデルの発展



たとえばこんなデータがあったしましょう

（次の時間の例題）

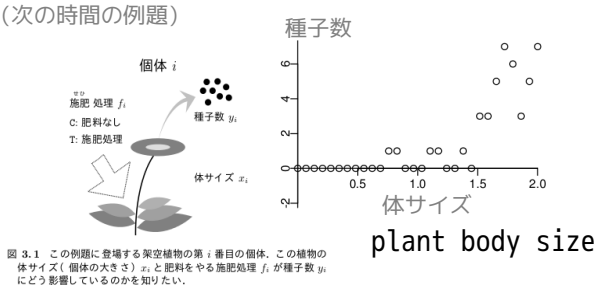
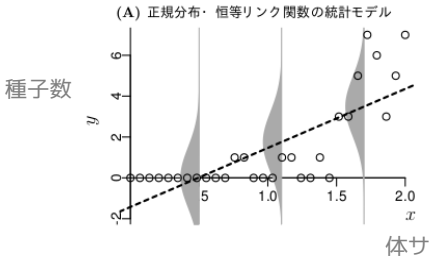


図 3.1 この例題に登場する架空植物の第 i 番目の個体．この植物の体サイズ（個体の大きさ）x_i と肥料をやる施肥処理 f_i が種子数 y_i にどう影響しているのかを知りたい．

残念なデータ解析



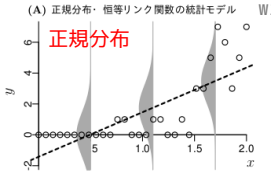
ブラックボックス
統計の典型

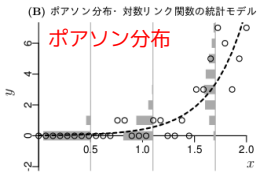
統計モデルを
考えていない

0 個，1 個，2 個と数える
種子数が「正規分布」……??

一般化線形モデル - ばらつきをよく見る

Don't use the normal distribution without seeing data!





線形モデルの発展

階層ベイズモデル

もっと自由な統計モデリングを！

一般化線形混合モデル

個体差・場所差といった変量効果をあつかいたい

一般化線形モデル

正規分布以外の確率分布をあつかいたい

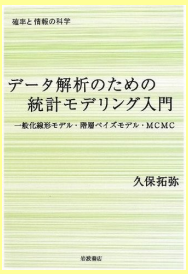
線形モデル

推定計算方法
MCMC
最尤推定法
最小二乗法

0 個，1 個，2 個と数えられる種子数が「正規分布」なわけないだろ!!

3.9 回帰モデルと確率分布の関係．また別の架空データに対して GLM をあてはめた例．破線は x とともに変化する平均値．グレイで

誰のための教科書・講義?



あまり勉強しない「理系」院生のための…

「理系」院生の秘密：あまりよく考えない

内容がわからなくてもソフトウェアにまるなげ

- ブラックボックス統計解析
- とにかく「ゆーい差」さえ出せばよいという発想(「理系」だけにありがちな思考?)

統計モデルを理解する
→「脱」ブラックボックス

10/6 の概要

(a, b) 08:50-10:20 概要，確率分布と最尤推定
(c) 10:30-12:00 一般化線形モデルとモデル選択
(d) 13:00-14:30 検定とロジスティック回帰
(e) 14:40-16:10 階層ベイズモデル
(f) 16:20-17:50 時間変化データの階層ベイズモデル
（先端科学トピックス）

10/7 の概要

- R 実習：データの操作と作図（基本編）
- 前日に説明した GLM 推定と R 作図などなど

茨城大集中講義 2016 (b)

確率分布と最尤推定

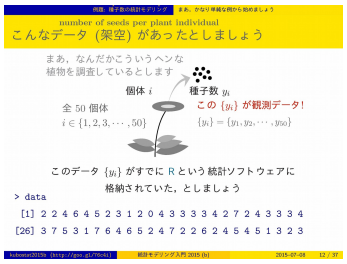
久保拓弥 kubo@ees.hokudai.ac.jp

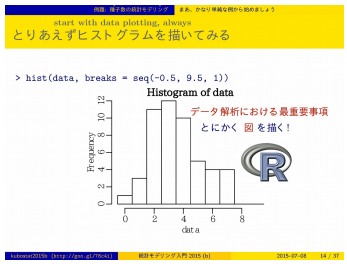
筑波大の講義 http://goo.gl/aFLLHZ

2016–10–06

ファイル更新時刻: 2016–09–26 13:55

単純化した例題





カウントデータはポアソン分布を使って説明できないかを調べる

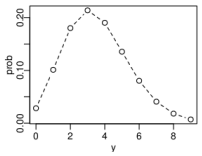


図 4 平均 $\lambda = 3.56$ のポアソン分布．種子数 y とその確率 prob の関係が示されている．図 2 の表を図にしたもの．R の plot() 関数の引数，type = "b" によって「丸と折れ線による図示」，lty = 2 によって「折れ線は破線で」と指示している．

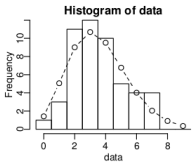
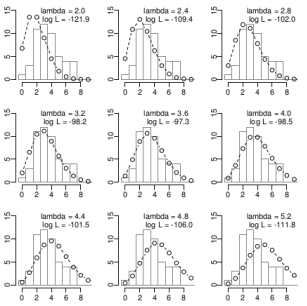


図 5 観測データと確率分布の対応をながめる．ヒストグラムは図 2 と同じ．それに重ねられている丸と破線は y 個の種子をもつ個体数の予測．平均 3.56 の図 1 のポアソン分布の確率分布に全個体数 50 をかけて得られる．

さいゆう 最尤推定という考えかたを説明します



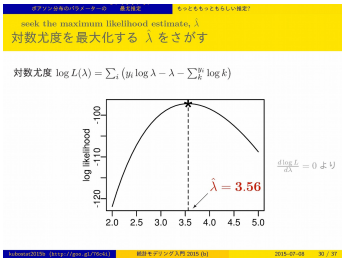


図 7 平均 λ（lambda）を変化させていったポアソン分布と，観測データへのあてはまりの良さ（対数尤度 logL）．すべてのヒストグラムは図 2 と同じ．

茨城大集中講義 2016 (c)

一般化線形モデル (ポアソン回帰) とモデル選択

久保拓弥 kubo@ees.hokudai.ac.jp

筑波大の講義 http://goo.gl/aFLLHZ

2016–10–06

ファイル更新時刻: 2016–09–28 14:41

ここで登場する ---
「何でも正規分布」ではダメ！という発想

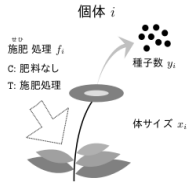


図 3.1 この例題に登場する架空植物の第 i 番目の個体．この植物の体サイズ（個体の大きさ）x_i と肥料をやる施肥処理 f_i が種子数 y_i にどう影響しているのかを知りたい．

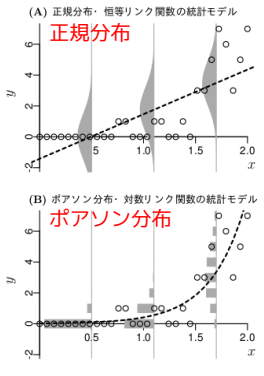


図 3.9 回帰モデルと確率分布の関係．また別の架空データに対して GLM をあてはめた例．破線は x とともに変化する平均値．グレイで

Free の統計ソフトウェア R で統計モデリング

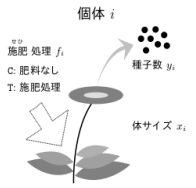
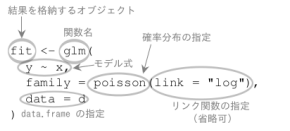


図 3.1 この例題に登場する架空植物の第 i 番目の個体サイズ（個体の大きさ）x_i と肥料をやる施肥処理 f_i にどう影響しているのかを知りたい．



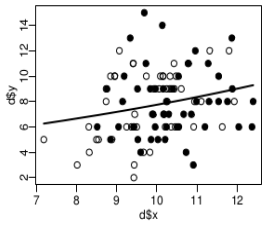
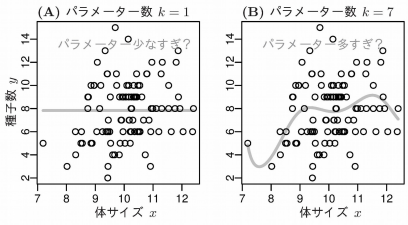


図 17 平均種子数 λ の予測．図 [12] に λ の予測値（実線）を上がきしたもの．

Q．モデル選択とは何か？

もくじ

パラメーター数は多くても少なくてもヘン？



What is the **"best?"** parameter number k?

茨城大集中講義 2016 (d)

統計学的検定 と ロジスティック回帰

久保拓弥 kubo@ees.hokudai.ac.jp

筑波大の講義 http://goo.gl/aFLLHZ

2016–10–06

ファイル更新時刻: 2016–09–28 14:41

A．より良い予測をする統計モデルを探すこと

統計学的な検定 そして，その非対称性

But their procedures are similar

しかしモデル選択と検定の手順は途中まで同じ

統計モデルの検定 AIC によるモデル選択 ←こっちだ!

検定はモデル選択じゃない!

解析対象のデータを確定

⇓

データを説明できるような統計モデルを設計

(帰無仮説・対立仮説) (単純モデル・複雑モデル)

⇓

ネストした統計モデルたちのパラメーターの 最尤（さいゆう） 推定計算

⇓ ⇓

帰無仮説棄却の危険率を評価 モデル選択規準 AIC の評価

統計学って「検定」のこと?

「検定」って何なの?

「検定」ってエラいの?

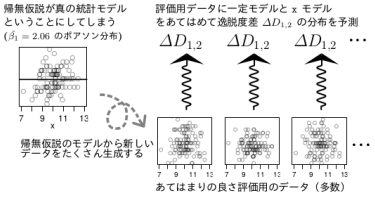
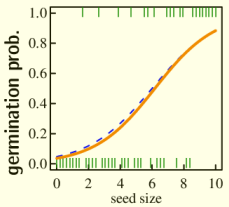


図 6 尤度比検定に必要な $\Delta D_{1,2}$ の分布の生成．まず帰無仮説である一定モデル（$\beta_1 = 2.06$, p.[2] 参照）が真の統計モデルだと仮定し，そこから得られるデータを使って逸脱度差 $\Delta D_{1,2}$ がどのような分布になるかを調べる．

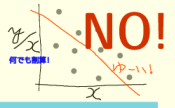
GLM のひとつ，ロジスティック回帰を使おう

データにあわせたより良い統計モデリングを!

おススめできないデータ解析を回避するための注意点



- むやみに 区画わけしない！
- 何でも 割り算するな！
- たくさん 図を描く
- 「観測データを説明する 確率分布は何か?」 を考える



コツ: 不自然にデータをこねくりまわさない
データの性質・構造にあったモデリングを!

2012-11-02 k4 (2012-10-26 17:07 修正版) 43/ 44

茨城大集中講義 2016 (e)

階層ベイズモデル – 個体差・場所差のモデリング

久保拓弥 kubo@ees.hokudai.ac.jp

筑波大の講義 http://goo.gl/aFLLHZ

2016–10–06

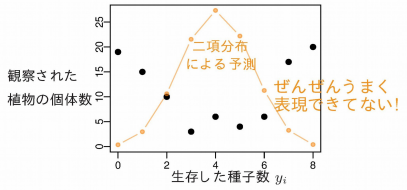
ファイル更新時刻: 2016–09–28 14:41

GLM ではうまく説明できないデータ!?

GLMM は階層ベイズモデルの一種 事前分布をどう選ぶかが重要

また別の観測データ：二項分布だめだめ?!

100 個体の植物の合計 800 種子中 **403** 個の生存が見られたので，平均生存確率は 0.50 と推定されたが……



さっきの例題と同じようなデータなのに?
(「 統計モデリング入門」第 10 章の最初の例題)

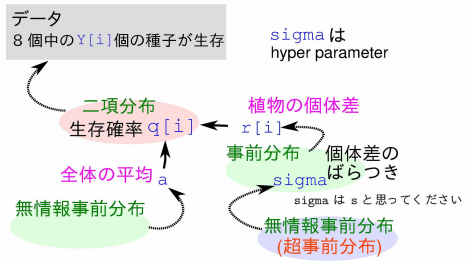
第 6 回と同じような例題を，こんどはベイズモデルを使ってモデリングします

GLM を階層ベイズモデル化して対処

GLMM は階層ベイズモデルの一種 事前分布をどう選ぶかが重要

なぜ「 階層」ベイズモデルと呼ばれるのか?

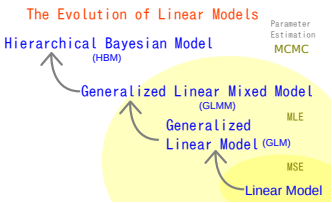
超事前分布 → 事前分布という 階層があるから



矢印は手順ではなく，依存関係をあらわしている

なぜ階層ベイズモデルまで勉強するの?

- 個体差・店舗差・地区差・空間相関・時間相関などめんどうな「細かい差異」をあつかわないといけない



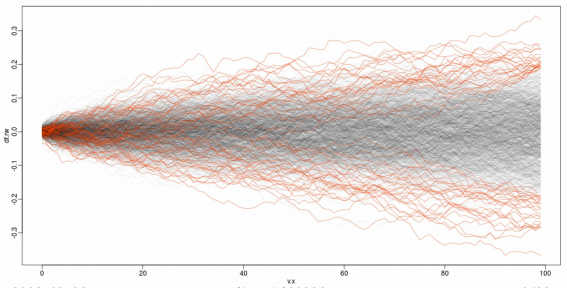
そういう難しい状況では……

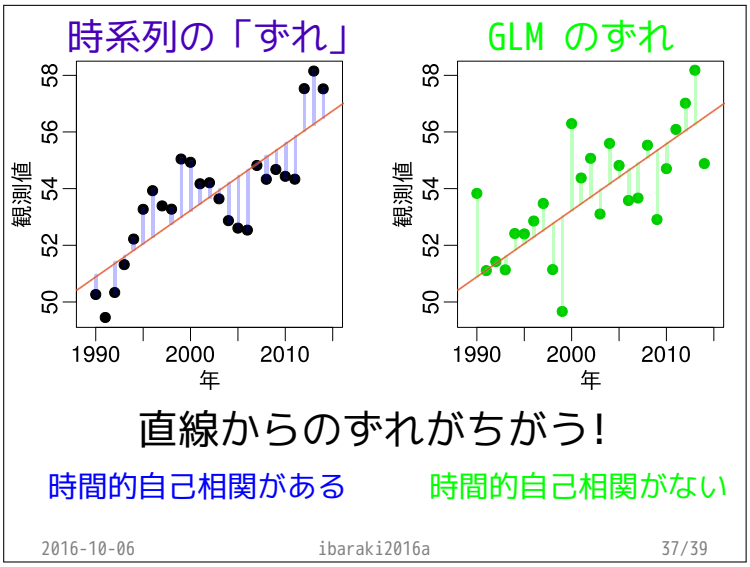
- 「差異」の階層ベイズモデル化
- そのパラメーターの事後分布を MCMC 法を使って推定するのが無難

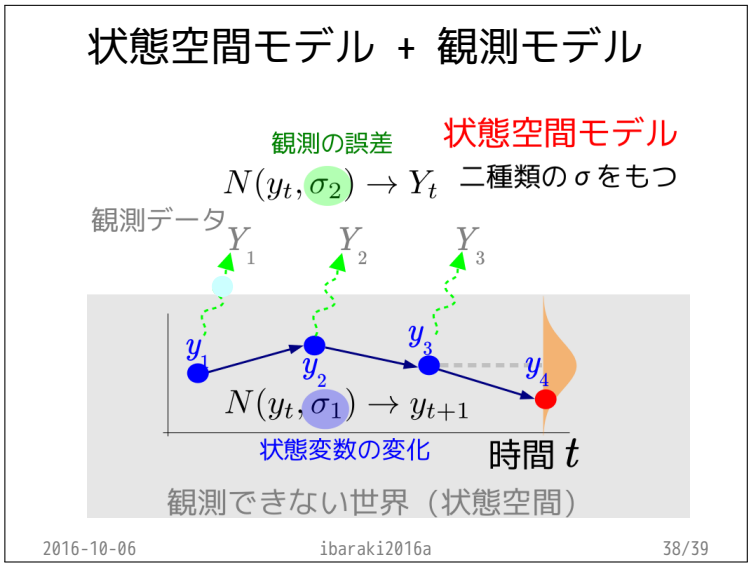
(f) 階層ベイズモデル – 時間変化の統計モデリング

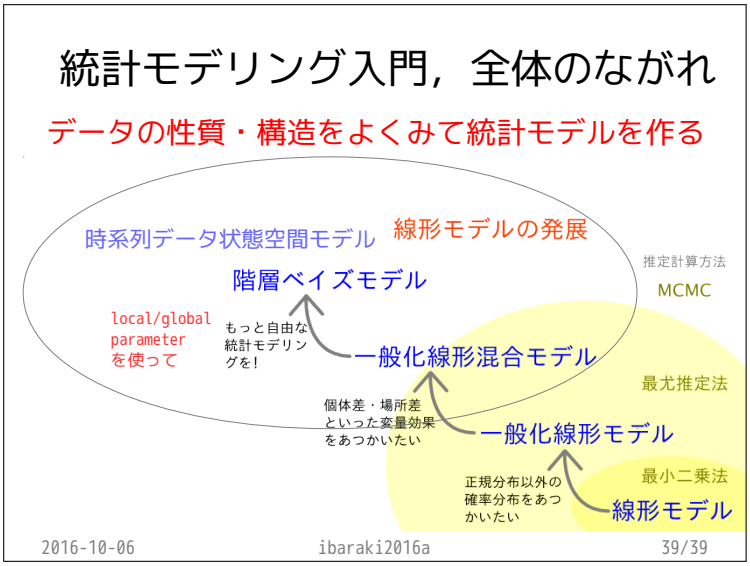
生態学の時系列データ解析でよく見る
『あぶない』モデリング

久保拓弥 mailto:kubo@ees.hokudai.ac.jp




時系列の「ずれ」
GLM のずれ
観測値
年
1990
2000
2010
50
52
54
56
58
直線からのずれがちがう!
時間的自己相関がある
時間的自己相関がない
2016-10-06 ibaraki2016a 37/39


状態空間モデル + 観測モデル
観測の誤差
状態空間モデル
$N(y_t, \sigma_2) \to Y_t$
二種類の σ をもつ
観測データ
Y_1
Y_2
Y_3
y_1
y_2
y_3
y_4
$N(y_t, \sigma_1) \to y_{t+1}$
状態変数の変化
時間 t
観測できない世界（状態空間）
2016-10-06 ibaraki2016a 38/39


統計モデリング入門，全体のながれ
データの性質・構造をよくみて統計モデルを作る
時系列データ状態空間モデル
線形モデルの発展
階層ベイズモデル
推定計算方法
MCMC
local/global parameter を使って
もっと自由な統計モデリングを!
一般化線形混合モデル
最尤推定法
個体差・場所差といった変量効果をあつかいたい
一般化線形モデル
正規分布以外の確率分布をあつかいたい
最小二乗法
線形モデル
2016-10-06 ibaraki2016a 39/39

茨城大集中講義 2016 (b)

確率分布と最尤推定

久保拓弥 kubo@ees.hokudai.ac.jp

筑波大の講義 http://goo.gl/aFLLHZ

2016–10–06

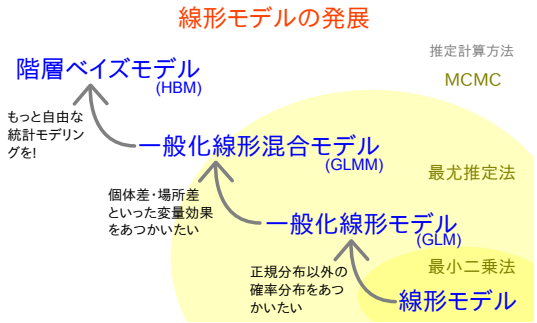
ファイル更新時刻: 2016–09–29 13:32

今日のハナシ I

1. 例題: 種子数の統計モデリング
   まあ，かなり単純な例から始めましょう
2. データと確率分布の対応
   probability distribution, the core of statistical model
3. ポアソン分布のパラメーターの 最尤推定(さいゆうすいてい)
   もっともらしい推定?
4. 統計モデルの要点
   乱数発生・推定・予測

統計モデリング授業前半の主題は「線形モデルを発展させる」こと

この授業であつかう統計モデルたち

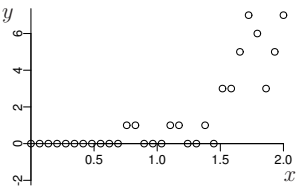


データの特徴にあわせて線形モデルを改良・発展させる

0 個, 1 個, 2 個と数えられるデータ

カウントデータ ($y \in \{0, 1, 2, 3, \cdots\}$ なデータ)

応答変数



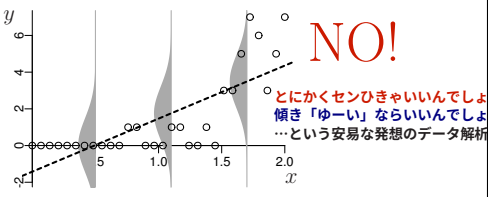
説明変数

- たとえば x は植物個体の大きさ，y はその個体の花数
- 体サイズが大きくなると花数が増えるように見えるが……
- この現象を表現する統計モデルは?

正規分布を使った統計モデル …… ムリがある?

正規分布・恒等リンク関数の統計モデル

応答変数



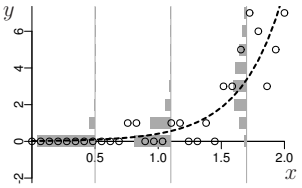
説明変数

- タテ軸のばらつきは「正規分布」なのか?
- y の値は 0 以上なのに ……
- 平均値がマイナス?

ポアソン分布を使った統計モデルなら良さそう?!

ポアソン分布・対数リンク関数の統計モデル

応答変数



YES!

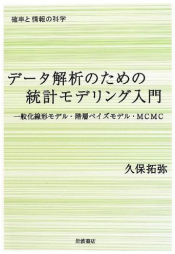
説明変数

- タテ軸に対応する「ばらつき」
- 負の値にならない「平均値」
- 正規分布を使ってるモデルよりましだね

データの性質をよくみる 確率分布という部品を選ぶ 「ぶらっくぼっくす」にしない!

今日の内容と「統計モデリング入門」との対応

http://goo.gl/Ufq2



今日はおもに「第 2 章 確率分布と統計モデルの最尤推定」の内容を説明します.

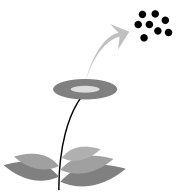
- 著者: 久保拓弥
- 出版社: 岩波書店
- 2012–05–18 刊行

2. 例題: 種子数の統計モデリング

まあ, かなり単純な例から始めましょう

R でデータをあつかいつつ

この授業では架空植物の架空データをあつかう



理由: よけいなことは考えなくてすむので

現実のデータはどれも授業で使うには難しすぎる……

こんなデータ (架空) があったとしましょう

まあ, なんだかこういうヘンな植物を調査しているとします

個体 i

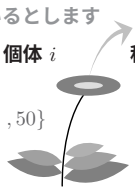
種子数 y_i

全 50 個体

$i \in \{1, 2, 3, \cdots, 50\}$

この $\{y_i\}$ が観測データ!

$\{y_i\} = \{y_1, y_2, \cdots, y_{50}\}$



このデータ $\{y_i\}$ がすでに R という統計ソフトウェアに格納されていた, としましょう

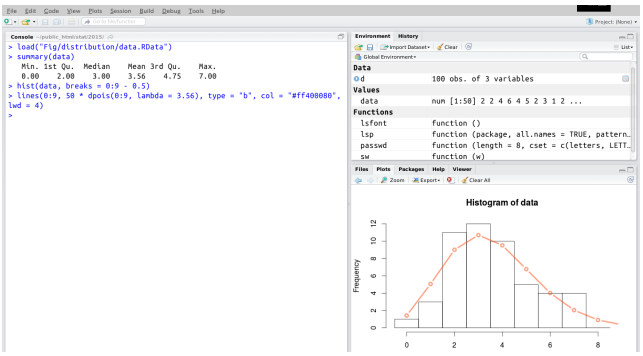
```
> data
 [1] 2 2 4 6 4 5 2 3 1 2 0 4 3 3 3 3 4 2 7 2 4 3 3 3 4
[26] 3 7 5 3 1 7 6 4 6 5 2 4 7 2 2 6 2 4 5 4 5 1 3 2 3
```

これ使いましょう：統計ソフトウェア R

http://www.r-project.org/

- いろいろな OS で使える free software
- 使いたい機能が充実している
- 作図機能も強力
- S 言語によるプログラミング可能
- Rstudio http://www.rstudio.com/

RStudio



http://www.rstudio.com/

R でデータの様子をながめる

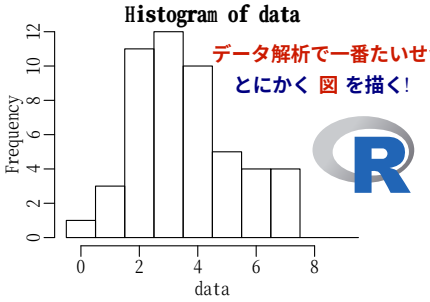

の table() 関数を使って種子数の頻度を調べる

```
> table(data)
 0  1  2  3  4  5  6  7
 1  3 11 12 10  5  4  4
```

(種子数 5 は 5 個体，種子数 6 は 4 個体 ……)

とりあえずヒストグラムを描いてみる

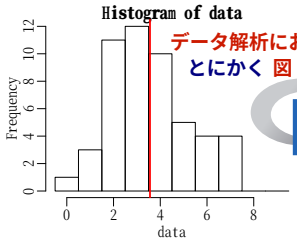
```
> hist(data, breaks = seq(-0.5, 9.5, 1))
```



データ解析で一番たいせつなこと
とにかく 図 を描く!

How to evaluate mean value using R?

```
> mean(data)
[1] 3.56
> abline(v = mean(data))
```



データ解析における最重要事項
とにかく 図 を描く!

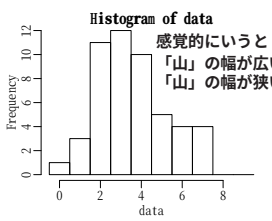
「ばらつき」の統計量

あるデータの ばらつき をあらわす標本統計量の例: 標本分散

```
> var(data)
[1] 2.9861
```

標本標準偏差 とは標本分散の平方根 $(\mathrm{SD} = \sqrt{\mathrm{variance}})$

```
> sd(data)
[1] 1.7280
> sqrt(var(data))
[1] 1.7280
```



感覚的にいうと
「山」の幅が広い: 分散が大きい
「山」の幅が狭い: 分散が小さい

3. データと確率分布の対応

probability distribution, the core of statistical model

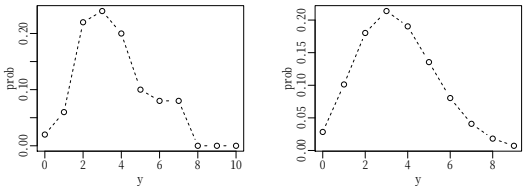
確率分布は統計モデルの重要な部品

Empirical VS Theoretical Distributions

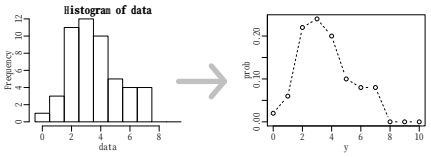
統計モデルの部品である **確率分布** には

“データそのまま” な **経験分布** (cf. サイコロ) と

数式で定義される**理論的な分布** がある



“データそのまま” な経験分布




```
> data.table <- table(factor(data, levels = 0:10))
> cbind(y = data.table, prob = data.table / 50)
    y  prob
0   1  0.02
1   3  0.06
2  11  0.22
3  12  0.24
4  10  0.20
5   5  0.10
6   4  0.08
7   4  0.08
8   0  0.00
9   0  0.00
10  0  0.00
```

- 確率分布とは **発生する事象** と **発生する確率** の対応づけ
- “たまたま手もとにある” データから “発生確率” を決める確率分布が**経験分布**

なるほど経験分布は “直感的” かもしれないが……

- データが変わると確率分布が変わる?
- 種子数 $y = \{0, 1, 2, \cdots\}$ となる確率が，個々におたがい無関係に決まる?
- パラメーターは $\{p_0, p_1, p_2, \cdots, p_{99}, p_{100}, \cdots\}$ 無限個ある?

道具として使うには，ちょっと不便かもしれない……

なにか理論的に導出された確率分布のほうが便利ではないか?

- 少数のパラメーターで分布の “カタチ” が決まる
- “なめらかに” 確率が変化する
- いろいろと数理的な道具が準備されている (パラメーター推定方法など)

確率分布 (ポアソン分布) を数式で決めてしまう

種子数が y である確率は以下のように決まる，と考えている

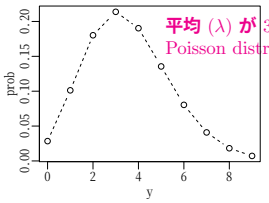
$$p(y \mid \lambda) = \frac{\lambda^y \exp(-\lambda)}{y!}$$

- $y!$ は y の階乗で，たとえば $4!$ は $1 \times 2 \times 3 \times 4$ をあらわしています．
- $\exp(-\lambda) = e^{-\lambda}$ のこと $(e = 2.718\cdots)$
- ここではなぜポアソン分布の確率計算が上のようになるのかは説明しません— まあ，こういうもんだと考えて先に進みましょう

数式で決められたポアソン分布?

とりあえず R で作図してみる

```
> y <- 0:9                          # これは種子数（確率変数）
> prob <- dpois(y, lambda = 3.56)   # ポアソン分布の確率の計算
> plot(y, prob, type = "b", lty = 2)
```

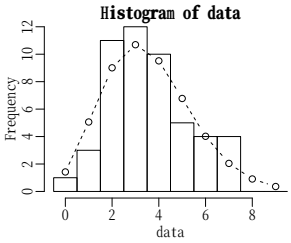


平均 (λ) が 3.56 である
Poisson distribution

```
> # cbind で「表」作り
> cbind(y, prob)
```

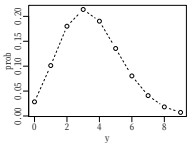
	y	prob
1	0	0.02843882
2	1	0.10124222
3	2	0.18021114
4	3	0.21385056
5	4	0.19032700
6	5	0.13551282
7	6	0.08040427
8	7	0.04089132
9	8	0.01819664
10	9	0.00719778

データとポアソン分布を重ね合わせる




```
> hist(data, seq(-0.5, 8.5, 0.5))     # まずヒストグラムを描き
> lines(y, prob, type = "b", lty = 2) # その「上」に折れ線を描く
```


パラメーター λ はポアソン分布の平均




```
> # cbind で「表」作り
> cbind(y, prob)
```

	y	prob
1	0	0.02843882
2	1	0.10124222
3	2	0.18021114
4	3	0.21385056
5	4	0.19032700
6	5	0.13551282
7	6	0.08040427
8	7	0.04089132
9	8	0.01819664
10	9	0.00719778

- 平均 λ はポアソン分布の唯一のパラメーター
- 確率分布の平均は λ である（$\lambda \geq 0$）
- 分散と平均は等しい: λ = 平均 = 分散
- $y \in \{0,\ 1,\ 2,\ \cdots,\ \infty\}$ の値をとり，すべての y について和をとると 1 になる

$$\sum_{y=0}^{\infty} p(y \mid \lambda) = 1$$

どういう場合にポアソン分布を使う?

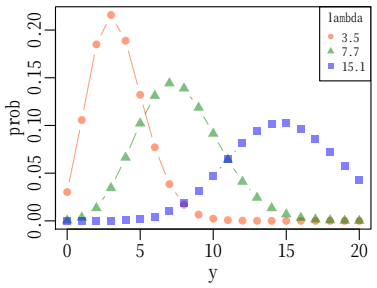
統計モデルの部品としてポアソン分布が選んだ理由:

- データに含まれている値 y_i が $\{0,\ 1,\ 2,\ \cdots\}$ といった非負の整数である（カウントデータである）
- y_i に下限（ゼロ）はあるみたいだけど上限はよくわからない
- この観測データでは平均と分散がだいたい等しい
  - このだいたい等しいがあやしいのだけど，まあ気にしないことにしましょう

ポアソン分布の λ を変えてみる

$$p(y \mid \lambda) = \frac{\lambda^y \exp(-\lambda)}{y!}$$

λ は平均をあらわすパラメーター



4. ポアソン分布のパラメーターの 最尤推定（さいゆうすいてい）

もっとももっともらしい推定?

「あてはめる」ことは推定すること

尤度 (likelihood) とは何か?

- 最尤推定法では，尤度 というあてはまりの良さをあらわす統計量に着目
- 尤度はデータが得られる確率をかけあわせたもの
- この例題の場合，パラメーター λ を変えると尤度が変わる
- もっとも「あてはまり」が良くなる λ を見つけたい
- たとえば，いまデータが 3 個体ぶん，たとえば，$\{y_1,\ y_2,\ y_3\} = \{2,\ 2,\ 4\}$，これだけだった場合，尤度はだいたい $0.180 \times 0.180 \times 0.19 = 0.006156$ といった値になる

尤度 $L(\lambda)$ はパラメーター λ の関数

この例題の尤度:

$$\begin{aligned} L(\lambda) &= (y_1\text{が } 2 \text{ である確率}) \times (y_2\text{が } 2 \text{ である確率}) \\ &\quad \times \cdots \times (y_{50}\text{が } 3 \text{ である確率}) \\ &= p(y_1 \mid \lambda) \times p(y_2 \mid \lambda) \times p(y_3 \mid \lambda) \times \cdots \times p(y_{50} \mid \lambda) \\ &= \prod_i p(y_i \mid \lambda) = \prod_i \frac{\lambda^{y_i} \exp(-\lambda)}{y_i!}, \end{aligned}$$

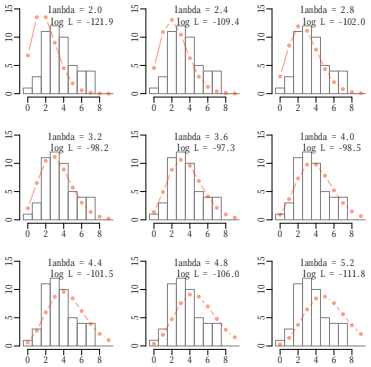
尤度はしんどいので対数尤度を使う

尤度は確率 (あるいは確率密度) の積であり，あつかいがふべん (大量のかけ算!)
そこで，パラメーターの最尤推定では，対数尤度関数 (log likelihood function) を使う

$$\log L(\lambda) = \sum_i \left(y_i \log \lambda - \lambda - \sum_k^{y_i} \log k \right)$$

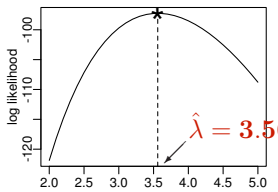
対数尤度 $\log L(\lambda)$ の最大化は尤度 $L(\lambda)$ の最大化になるから
まずは，平均をあらわすパラメーター λ を変化させていったときに，ポアソン分布のカタチと対数尤度がどのように変化するのかを調べてみましょう

λ を変えるとあてはまりの良さが変わる



対数尤度を最大化する $\hat{\lambda}$ をさがす

対数尤度 $\log L(\lambda) = \sum_i \left(y_i \log \lambda - \lambda - \sum_k^{y_i} \log k \right)$

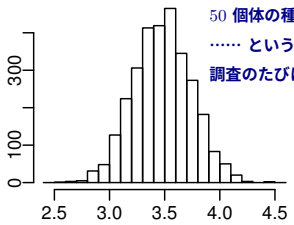


$\frac{d \log L}{d\lambda} = 0$ より

- 最尤推定量 (ML estimator): $\sum_i y_i / 50$ 標本平均値!
- 最尤推定値 (ML estimate): $\hat{\lambda} = 3.56$ ぐらい

最尤推定を使っても真の λ は見つからない

真の λ が 3.5 の場合



50 個体の種子数を調べる
…… ということを 3000 回くりかえし
調査のたびに $\hat{\lambda}$ を最尤推定した

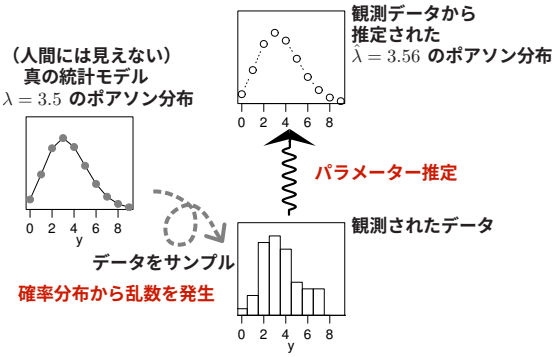
データは有限なので真の λ はわからない

5. 統計モデルの要点

乱数発生・推定・予測

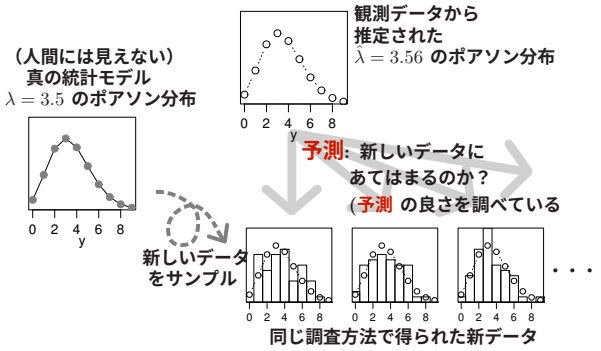
統計モデルとデータの対応づけ

確率分布: 乱数発生 と 推定



データ？…ここでは確率・統計モデルが生成していると仮定

推定されたモデルを使った 予測



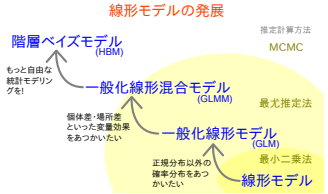
この講義で登場する確率分布

- **ポアソン分布**: $y \in \{0, 1, 2, 3, \cdots\}$ となるデータ，「y 回なにかがおこった」
- **二項分布**: $y \in \{0, 1, 2, \cdots, N\}$ となるデータ，「N 個のうち y 個で何かがおこった」
- **正規分布**: $-\infty < y < \infty$ の連続値をとるデータ
- その他あれこれ — ちょっと登場するだけ

そんなに多くの確率分布は登場しません

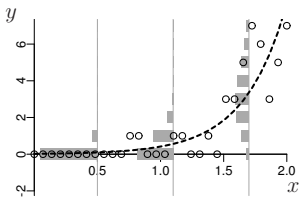
いろいろな確率分布があるけれど……

- この講義では多種多様な確率分布を**あつかいません**
- しかし **確率分布を混ぜあわせる** ことによって，自分で確率分布を作り出すことができます
- ハナシの後半に登場する GLMM や階層ベイズモデル



次回予告

The next topic



YES!

一般化線形モデルのひとつ: ポアソン回帰

Poisson Regression, a Generalized Linear Model (GLM)

茨城大集中講義 2016 (c)

一般化線形モデル (ポアソン回帰) とモデル選択

久保拓弥 kubo@ees.hokudai.ac.jp

筑波大の講義 http://goo.gl/aFLLHZ

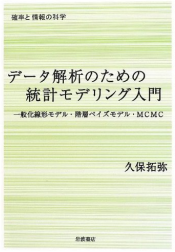
2016–10–06

ファイル更新時刻: 2016–09–29 14:35

今日のハナシ I

今日の内容と「統計モデリング入門」との対応

http://goo.gl/Ufq2



今日はおもに「第 3 章 一般化線形モデル (GLM)」の内容を説明します.

- 著者: 久保拓弥
- 出版社: 岩波書店
- 2012–05–18 刊行

一般化線形モデルって何だろう?

一般化線形モデル (GLM)

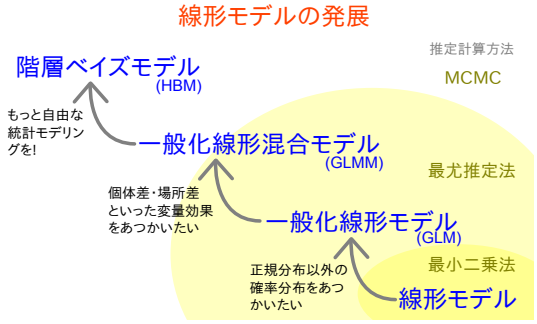
- ポアソン回帰 (Poisson regression)
- ロジスティック回帰 (logistic regression)
- 直線回帰 (linear regression)
- ……

1. ポアソン回帰の統計モデル

応答変数 y と説明変数 x

一般化線形モデルにとりくんでみる

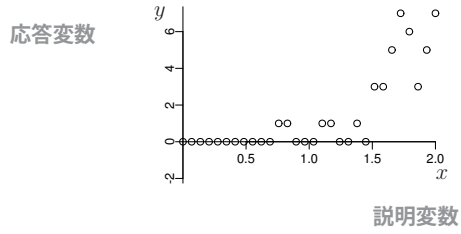
この授業であつかう統計モデルたち



データの特徴にあわせて線形モデルを改良・発展させる

0 個, 1 個, 2 個と数えられるデータ

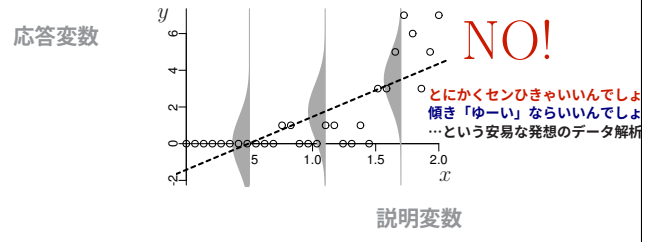
カウントデータ ($y \in \{0, 1, 2, 3, \cdots\}$ なデータ)



- たとえば x は植物個体の大きさ，y はその個体の花数
- 体サイズが大きくなると花数が増えるように見えるが……
- この現象を表現する統計モデルは?

正規分布を使った統計モデル …… ムリがある?

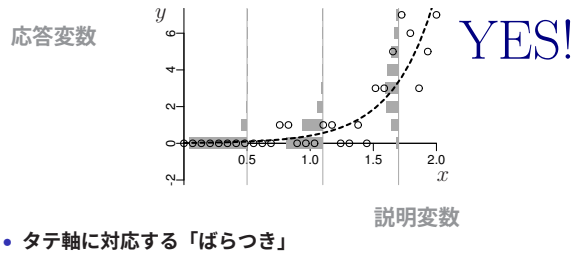
正規分布・恒等リンク関数の統計モデル



- タテ軸のばらつきは「正規分布」なのか?
- y の値は 0 以上なのに ……
- 平均値がマイナス?

ポアソン分布を使った統計モデルなら良さそう?!

ポアソン分布・対数リンク関数の統計モデル



- タテ軸に対応する「ばらつき」
- 負の値にならない「平均値」
- 正規分布を使ってるモデルよりましだね

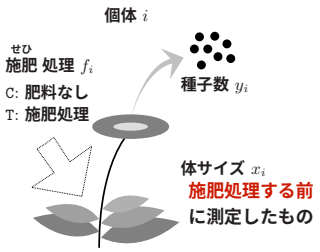
7. ポアソン回帰の例題: 架空植物の種子数データ

植物個体の属性，あるいは実験処理が種子数に影響?

まずはデータの概要を調べる

個体サイズと実験処理の効果を調べる例題

- 応答変数: 種子数 $\{y_i\}$
- 説明変数:
  - 体サイズ $\{x_i\}$
  - 施肥処理 $\{f_i\}$



標本数

- 無処理 (f_i = C): 50 sample ($i \in \{1, 2, \cdots 50\}$)
- 施肥処理 (f_i = T): 50 sample ($i \in \{51, 52, \cdots 100\}$)

データファイルを読みこむ

data3a.csv は CSV (comma separated value) format file なので，R で読みこむには以下のようにする:

```
> d <- read.csv("data3a.csv")
```

データは d と名付けられた data frame (表みたいなもの) に格納される

とりあえず data frame d を表示

```
> d
    y     x  f
1   6  8.31  C
2   6  9.44  C
3   6  9.50  C
... (中略) ...
99  7 10.86  T
100 9  9.97  T
```

data frame d を調べる: 連続値と整数値

```
> d$x
  [1]  8.31  9.44  9.50  9.07 10.16  8.32 10.61 10.06
  [9]  9.93 10.43 10.36 10.15 10.92  8.85  9.42 11.11
... (中略) ...
 [97]  8.52 10.24 10.86  9.97

> d$y
  [1]  6  6  6 12 10  4  9  9  9 11  6 10  6 10 11  8
 [17]  3  8  5  5  4 11  5 10  6  6  7  9  3 10  2  9
... (中略) ...
 [97]  6  8  7  9
```


data frame d を調べる: “因子型” のデータ

施肥処理の有無をあらわす f 列はちょっと様子がちがう

```
> d$f
  [1] C C C C C C C C C C C C C C C C C C C C C C C C C
 [26] C C C C C C C C C C C C C C C C C C C C C C C C C
 [51] T T T T T T T T T T T T T T T T T T T T T T T T T
 [76] T T T T T T T T T T T T T T T T T T T T T T T T T
Levels: C T
```

因子型データ: いくつかの水準をもつデータ
ここでは C と T の 2 水準

R のデータのクラスとタイプ

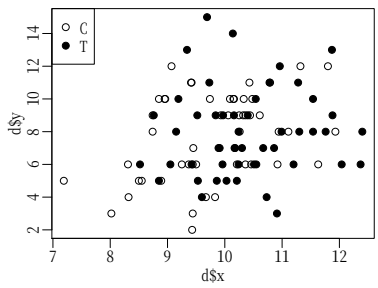
```
> class(d) # d は data.frame クラス
[1] "data.frame"
> class(d$y) # y 列は整数だけの integer クラス
[1] "integer"
> class(d$x) # x 列は実数も含むので numeric クラス
[1] "numeric"
> class(d$f) # そして f 列は factor クラス
[1] "factor"
```


data frame の summary()

```
> summary(d)
       y               x          f
 Min.   : 2.00   Min.   : 7.190   C:50
 1st Qu.: 6.00   1st Qu.: 9.428   T:50
 Median : 8.00   Median :10.155
 Mean   : 7.83   Mean   :10.089
 3rd Qu.:10.00   3rd Qu.:10.685
 Max.   :15.00   Max.   :12.400
```


データはとにかく図示する!

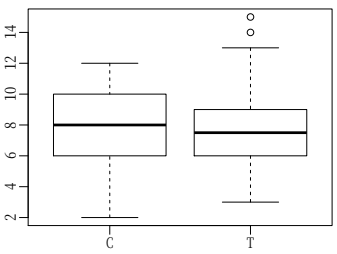
```
> plot(d$x, d$y, pch = c(21, 19)[d$f])
> legend("topleft", legend = c("C", "T"), pch = c(21, 19))
```



散布図

施肥処理 f を横軸とした図

```
> plot(d$f, d$y)
```



箱ひげ図

3. GLM の詳細を指定する

確率分布・線形予測子・リンク関数

ポアソン回帰では log link 関数を使うのが便利

一般化線形モデルを作る

一般化線形モデル (GLM)

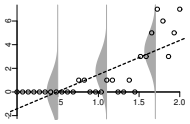
- 確率分布は?
- 線形予測子は?
- リンク関数は?

GLM のひとつである直線回帰モデルを指定する

直線回帰のモデル

- 確率分布: 正規分布
- 線形予測子: e.g., $\beta_1 + \beta_2 x_i$

  直線の式: (切片) + (傾き) $\times x_i$
- リンク関数: 恒等リンク関数



結果 ← 原因 (かも?) を表現する線形モデル

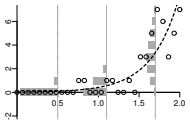
- 結果: 応答変数
- 原因: 説明変数
- 線形予測子 (linear predictor):

(応答変数の平均) = 定数 (切片)
+ (係数 1) × (説明変数 1)
+ (係数 2) × (説明変数 2)
+ (係数 3) × (説明変数 3)
+ ···

GLM のひとつであるポアソン回帰モデルを指定する

ポアソン回帰のモデル

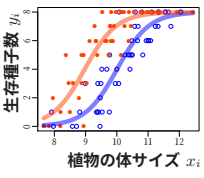
- 確率分布: ポアソン分布
- 線形予測子: e.g., $\beta_1 + \beta_2 x_i$
- リンク関数: 対数リンク関数



GLM のひとつである logistic 回帰モデルを指定する

ロジスティック回帰のモデル

- 確率分布: 二項分布
- 線形予測子: e.g., $\beta_1 + \beta_2 x_i$
- リンク関数: logit リンク関数

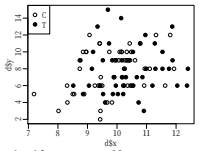


R で一般化線形モデル (GLM) の推定を……

	確率分布	乱数発生	GLM あてはめ
(離散)	ベルヌーイ分布	rbinom()	glm(family = binomial)
	二項分布	rbinom()	glm(family = binomial)
	ポアソン分布	rpois()	glm(family = poisson)
	負の二項分布	rnbinom()	glm.nb() in library(MASS)
(連続)	ガンマ分布	rgamma()	glm(family = gamma)
	正規分布	rnorm()	glm(family = gaussian)

- glm() で使える確率分布は上記以外もある
- GLM は直線回帰・重回帰・分散分析・ポアソン回帰・ロジスティック回帰その他の「よせあつめ」と考えてもよいかも

さてさて，種子数の例題にもどって



種子数 y_i は平均 λ_i のポアソン分布にしたがう としましょう

$$p(y_i \mid \lambda_i) = \frac{\lambda_i^{y_i} \exp(-\lambda_i)}{y_i!}$$

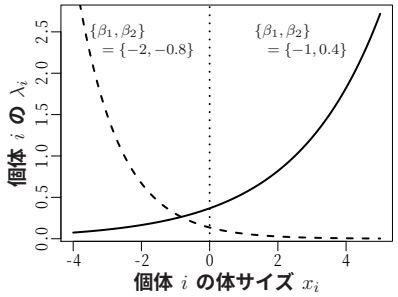
個体 i の平均 λ_i を以下のようにおいてみたらどうだろう……?

$$\lambda_i = \exp(\beta_1 + \beta_2 x_i)$$

- β_1 と β_2 は係数 (パラメーター)
- x_i は個体 i の体サイズ，f_i はとりあえず無視

指数関数ってなんだっけ?

$$\lambda_i = \exp(\beta_1 + \beta_2 x_i)$$



GLM のリンク関数と線形予測子 ← (直線の式)

個体 i の平均 λ_i

$$\lambda_i = \exp(\beta_1 + \beta_2 x_i)$$

$$\Updownarrow$$

$$\log(\lambda_i) = \beta_1 + \beta_2 x_i$$

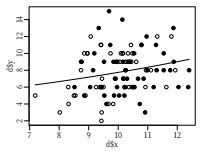
$$\log(\text{平均}) = \text{線形予測子}$$

log リンク関数とよばれる理由は，上のようになっているから

この例題のための統計モデル

ポアソン回帰のモデル

- 確率分布: ポアソン分布
- 線形予測子: $\beta_1 + \beta_2 x_i$
- リンク関数: 対数リンク関数



4. R で GLM のパラメーターを推定

あてはまりの良さは対数尤度関数で評価

推定計算はコンピューターにおまかせ

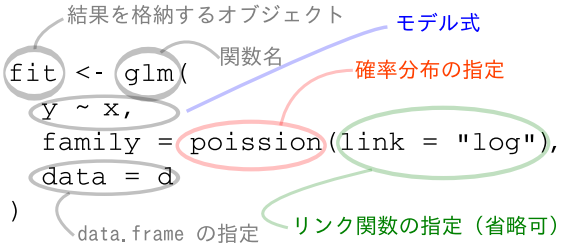
glm() 関数の指定

```
> d
    y     x  f
1   6  8.31  C
2   6  9.44  C
3   6  9.50  C
... (中略) ...
99  7 10.86  T
100 9  9.97  T
```

これだけ!

```
> fit <- glm(y ~ x, data = d, family = poisson)
```


glm() 関数の指定の意味



- モデル式 (線形予測子 z): どの説明変数を使うか?
- link 関数: z と応答変数 (y) 平均値 の関係は?
- family: どの確率分布を使うか?

glm() 関数の出力

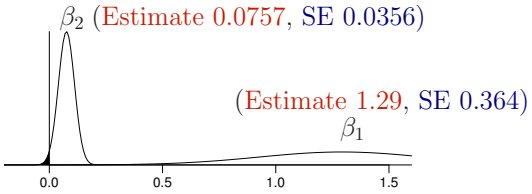
```
> fit <- glm(y ~ x, data = d, family = poisson)

all:  glm(formula = y ~ x, family = poisson, data = d)

Coefficients:
(Intercept)            x
     1.2917       0.0757

Degrees of Freedom: 99 Total (i.e. Null);  98 Residual
Null Deviance:^^I    89.5
Residual Deviance: 85 ^^IAIC: 475
```


glm() 関数のくわしい出力

```
> summary(fit)
Call:
glm(formula = y ~ x, family = poisson, data = d)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-2.368  -0.735  -0.177   0.699   2.376

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   1.2917     0.3637    3.55  0.00038
x             0.0757     0.0356    2.13  0.03358
```

…… (以下，省略) ……

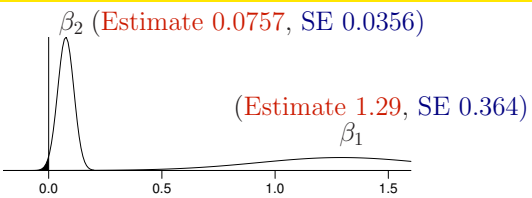
推定値と標準誤差のいめーじ (かなりいいかげんな説明)



- 確率 p は ゼロからの距離 をあらわしている
- p がゼロに近いほど 推定値 $\hat{\beta}$ はゼロから離れている
- p が 0.5 に近いほど 推定値 $\hat{\beta}$ はゼロに近い

(注: 頻度主義的な信頼区間の正しい解釈はもっとめんどくさい)

推定値と標準誤差のいめーじ (何がめんどくさいの?)

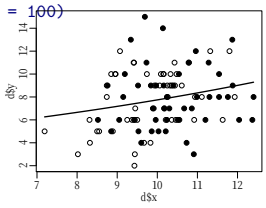


- 区間 95% 内に「ゼロ」があるとしよう $\rightarrow$ 「だから何？」
- 多数のパラメーターがある場合には?
- 授業の後半であつかうベイズ統計モデルでの解釈は 簡単 ……になるはず……

モデルの予測

```
> fit <- glm(y ~ x, data = d, family = poisson)
...
Coefficients:
(Intercept)            x
     1.2917       0.0757

> plot(d$x, d$y, pch = c(21, 19)[d$f]) # data
> xp <- seq(min(d$x), max(d$x), length = 100)
> lines(xp, exp(1.2917 + 0.0757 * xp))
```

ここでは観測データと予測の関係を見ているだけ，なのだが

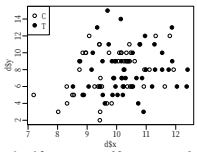


5. 処理をした・しなかった 効果も統計モデルに入れる

GLM の因子型説明変数

数量型 + 因子型 という組み合わせで

肥料の効果 f_i もいれましょう



種子数 y_i は平均 λ_i のポアソン分布にしたがう としましょう

$$p(y_i \mid \lambda_i) = \frac{\lambda_i^{y_i} \exp(-\lambda_i)}{y_i!}$$

個体 i の平均 λ_i を次のようにする

$$\lambda_i = \exp(\beta_1 + \beta_2 x_i + \beta_3 d_i)$$

- β_3 は施肥処理の効果 の 係数
- f_i のダミー変数

$$d_i = \begin{cases} 0 & (f_i = \texttt{C} \text{ の場合}) \\ 1 & (f_i = \texttt{T} \text{ の場合}) \end{cases}$$

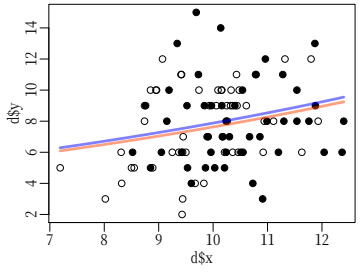
glm(y ~ x + f, ...) の出力

```
> summary(glm(y ~ x + f, data = d, family = poisson))
...(略)...

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   1.2631     0.3696    3.42  0.00063
x             0.0801     0.0370    2.16  0.03062
fT           -0.0320     0.0744   -0.43  0.66703
```

……（以下，省略）……

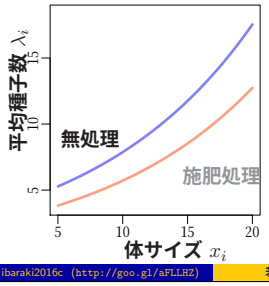
x + f モデルの予測

```
> plot(d$x, d$y, pch = c(21, 19)[d$f]) # data
> xp <- seq(min(d$x), max(d$x), length = 100)
> lines(xp, exp(1.2631 + 0.0801 * xp), col = "blue", lwd = 3) # C
> lines(xp, exp(1.2631 + 0.0801 * xp - 0.032), col = "red", lwd = 3) # T
```



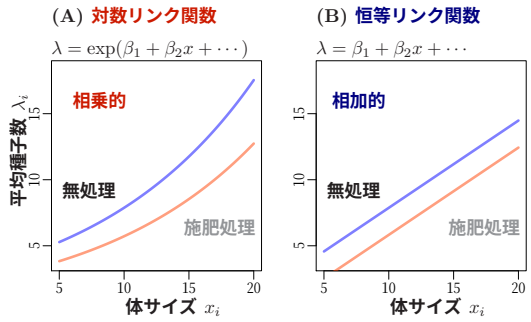
複数の説明変数をいれた場合の統計モデル

- $f_i = \texttt{C}$: $\lambda_i = \exp(1.26 + 0.0801 x_i)$
- $f_i = \texttt{T}$: $\lambda_i = \exp(1.26 + 0.0801 x_i - 0.032)$
  $= \exp(1.26 + 0.0801 x_i) \times \exp(-0.032)$

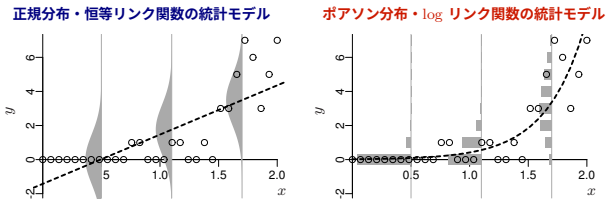


施肥効果である $\exp(-0.032)$ は**かけ算**できくことに注意!

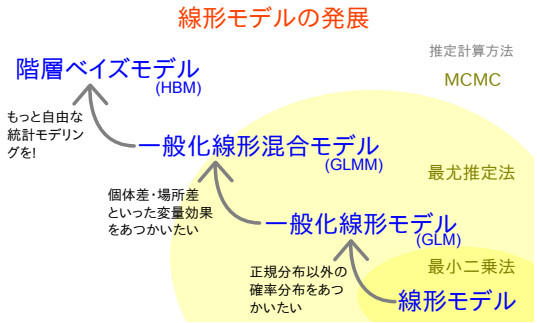
リンク関数が違うとモデルの解釈が異なる



GLM: 適切な確率分布 とリンク関数を選ぶ



この講義であつかう統計モデルたち



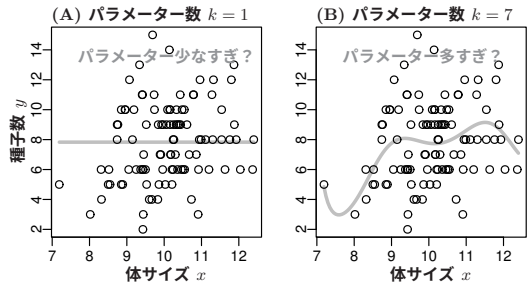
データの特徴にあわせて線形モデルを改良・発展させる

6. モデル選択

予測力のよい統計モデルはどれか?

予測の悪さの基準 AIC

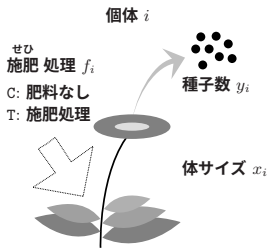
パラメーター数 k は多くても少なくてもヘン?



“良いモデル” とはなにか? k も重要なのか?

個体サイズと実験処理の効果を調べる例題

- **応答変数: 種子数** $\{y_i\}$
- **説明変数:**
  - **体サイズ** $\{x_i\}$
  - **施肥処理** $\{f_i\}$



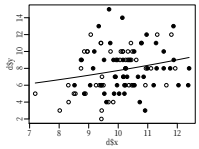
標本数

- **無処理** (f_i = C): 50 sample ($i \in \{1, 2, \cdots 50\}$)
- **施肥処理** (f_i = T): 50 sample ($i \in \{51, 52, \cdots 100\}$)

この例題のための統計モデル

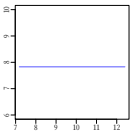
ポアソン回帰のモデル

- 確率分布: **ポアソン分布**
- 線形予測子: $\beta_1 + \beta_2 x_i + \beta_3 f_i$
- リンク関数: **対数リンク関数**



4 つの可能なモデル候補: (A) constant λ

$$\lambda_i = \exp(\beta_1)$$

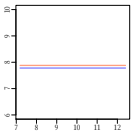


あてはまりの良さを対数尤度 (log likelihood) **で評価する**

```
> logLik(glm(y ~ 1, data = d, family = poisson))
'log Lik.' -237.64 (df=1)
```


4 つの可能なモデル候補: (B) f model

$$\lambda_i = \exp(\beta_1 + \beta_3 f_i)$$

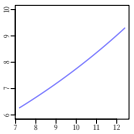


あてはまりの良さを対数尤度 (log likelihood) **で評価する**

```
> logLik(glm(y ~ f, data = d, family = poisson))
'log Lik.' -237.63 (df=2)
```


4 つの可能なモデル候補: (C) x model

$$\lambda_i = \exp(\beta_1 + \beta_2 x_i)$$

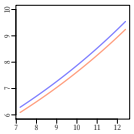


あてはまりの良さを対数尤度 (log likelihood) **で評価する**

```
> logLik(glm(y ~ x, data = d, family = poisson))
'log Lik.' -235.39 (df=2)
```


4 つの可能なモデル候補: (D) x + f model

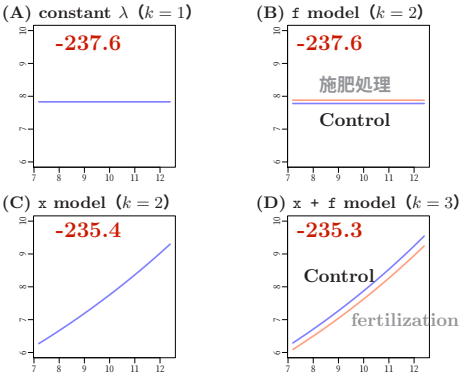
$$\lambda_i = \exp(\beta_1 + \beta_2 x_i + \beta_3 f_i)$$



あてはまりの良さを対数尤度 (log likelihood) **で評価する**

```
> logLik(glm(y ~ x + f, data = d, family = poisson))
'log Lik.' -235.29 (df=3)
```


パラメーター数が多いとあてはまりが良い



7. AIC を使ったモデル選択

あてはまりの悪さ: deviance

そして予測の悪さ: AIC

R の glm() は deviance を出力

```
> glm(y ~ x + f, data = d, family = poisson)

Call:  glm(formula = y ~ x + f, family = poisson, data = d)

Coefficients:
(Intercept)            x           fT
     1.2631       0.0801      -0.0320

Degrees of Freedom: 99 Total (i.e. Null);  97 Residual
Null Deviance:      89.5
Residual Deviance: 84.8          AIC: 477
```

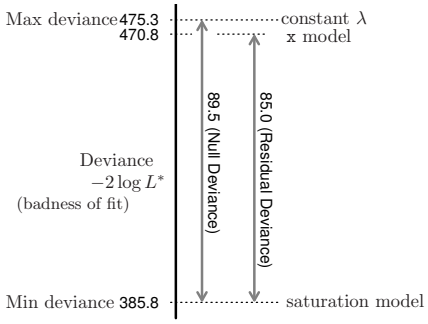
Residual Deviance? Null Deviance? AIC?

deviance $D = -2 \times \log L^*$

- Maximum log likelihood $\log L^*$: goodness of fit
- Deviance $D = -2 \log L^*$: badness of fit

model	k	$\log L^*$	Deviance $-2 \log L^*$	Residual deviance
constant λ	1	-237.6	475.3	89.5
f	2	-237.6	475.3	89.5
x	2	-235.4	470.8	85.0
x + f	3	-235.3	470.6	84.8
saturation	100	-192.9	385.8	0.0

Null deviance, Residual deviance, ...



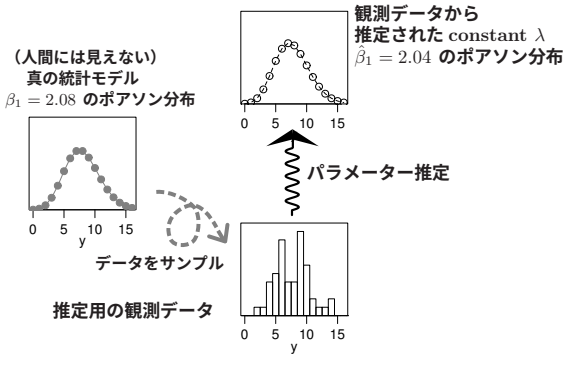
予測の悪さ: AIC $= -2 \log L^* + 2k$

AIC 最小のモデルを選ぶ

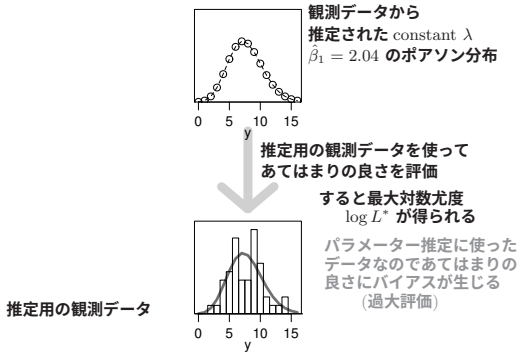
model	k	$\log L^*$	Deviance $-2 \log L^*$	Residual deviance	AIC
constant λ	1	-237.6	475.3	89.5	477.3
f	2	-237.6	475.3	89.5	479.3
x	**2**	**-235.4**	**470.8**	**85.0**	**474.8**
x + f	3	-235.3	470.6	84.8	476.6
saturation	100	-192.9	385.8	0.0	585.8

AIC: A (or Akaike) information criterion

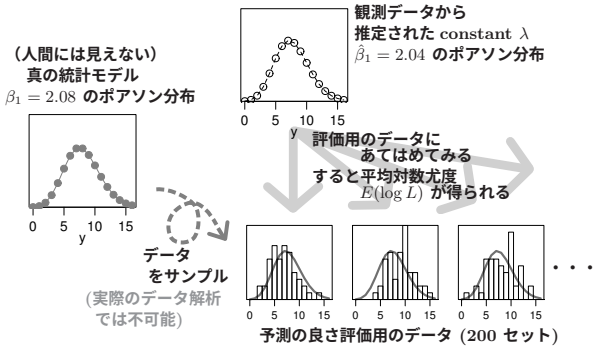
統計モデルによる推測って何だっけ?



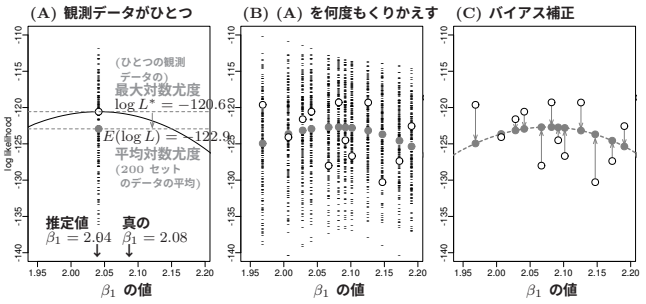
推定に使ったデータであてはまりを評価している?



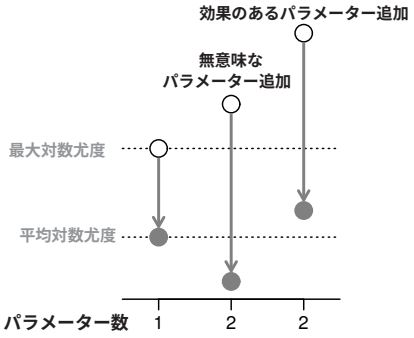
重要なこと: 新データがあてはまるかどうか



シミュレイションで予測の良さを調べる



バイアス補正を図示してみる



FAQ モデル選択

http://hosho.ees.hokudai.ac.jp/~kubo/ce/FaqModelSelection.html

次回予告

統計学的検定
と
ロジスティック回帰 (GLM)

茨城大集中講義 2016 (d)

統計学的検定 と ロジスティック回帰

久保拓弥 kubo@ees.hokudai.ac.jp

筑波大の講義 http://goo.gl/aFLLHZ

2016–10–06

ファイル更新時刻: 2016–09–29 15:04

今日のハナシ I

1. 統計学的な検定

そして，その非対称性

ここでは 尤度比検定 を紹介

モデル選択と検定の手順は途中まで同じ

統計モデルの検定	AIC によるモデル選択

解析対象のデータを確定

⇓

データを説明できるような統計モデルを設計

(帰無仮説・対立仮説) (単純モデル・複雑モデル)

⇓

ネストした統計モデルたちのパラメーターの 最尤(さいゆう) 推定計算

⇓ ⇓

帰無仮説棄却の危険率を評価 モデル選択規準 AIC の評価

⇓ ⇓

帰無仮説棄却の可否を判断 予測の良いモデルを選ぶ

モデル選択 と統計学的検定 は

その目的がぜんぜんちがう

目的?

モデル選択: よい予測をするモデルの探索

統計学的検定: 帰無仮説の排除

統計学的な検定 (Neyman-Pearson framework)

statistical test

Null hypothesis
帰無仮説
glm(y ~ 1)
is better!

VS

Alternative hypothesis
対立仮説
glm(y ~ x)
is better!

どうでもいい
… 興味ない…

重要！これを
主張したい！

非対称性 asymmetricity?

統計学的な検定 (Neyman-Pearson framework)

statistical test

Null hypothesis
帰無仮説
glm(y ~ 1)
is better!

VS

Alternative hypothesis
対立仮説
glm(y ~ x)
is better!

test!
(if ...)
reject 棄却 ------ support 支持

非対称性 asymmetricity?

統計学的な検定 (Neyman-Pearson framework)

statistical test

Null hypothesis
帰無仮説
glm(y ~ 1)
is better!

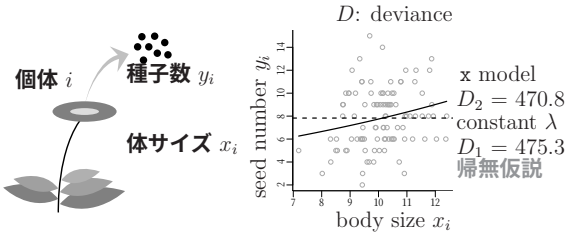
VS

Alternative hypothesis
対立仮説
glm(y ~ x)
is better!

test!
(if ...)
NOT reject ------ Say Nothing!?

非対称性 asymmetricity?

また同じ例題



(施肥処理は無視!)

検定統計量 $\Delta D_{1,2}$

difference in deviance $\Delta D_{1,2} = D_1 - D_2 = \mathbf{4.51} \approx 4.5$

likelihood ratio? — $\log \frac{L_1^*}{L_2^*} = \log L_1^* - \log L_2^*$

model	k	$\log L^*$	Deviance $-2 \log L^*$	
constant λ	1	-237.6	$D_1 = 475.3$	帰無仮説
x	2	-235.4	$D_2 = 470.8$	対立仮説

検定の非対称性: **帰無仮説はゴミあつかい**
……にもかかわらず，帰無仮説だけをしつこく調べる

帰無仮説のつくりかた

対立仮説の中に帰無仮説がある (ネストした関係)

- カウントデータ $\{y_i\}$ は平均である λ_i のポアソン分布に従う
- 対立仮説の一例: $\log \lambda_i = \beta_1 + \beta_2 x_i$
- ネストした 帰無仮説: $\log \lambda_i = \beta_1$ (切片だけのモデル)

検定の目的: 帰無仮説  の棄却

観察された逸脱度差 $\Delta D_{1,2} = 4.5$ は……

↓帰無仮説は	「めったにない差」(帰無仮説を棄却)	「よくある差」(棄却できない)
真のモデルである	第一種の過誤	(問題なし)
真のモデルではない	(問題なし)	第二種の過誤

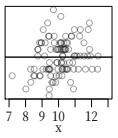
↓  is ...	significant (Reject )	not significant (Not reject )
TRUE	Type I error	(no problem)
NOT true	(no problem)	Type II error

検定の非対称性: 第一種の過誤だけに注目

$\Delta D_{1,2}$ の分布を生成: ブートストラップ尤度比検定

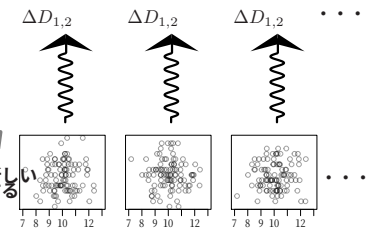
帰無仮説  が真のモデルであるとしよう!

帰無仮説が真の統計モデルということにしてしまう ($\hat{\beta}_1 = 2.06$ のポアソン分布)



帰無仮説のモデルから新しいデータをたくさん生成する

評価用データに constant λ と x model をあてはめて逸脱度差 $\Delta D_{1,2}$ の分布を予測



あてはまりの良さ評価用のデータ（多数）

ブートストラップ法って何?

コンピューターに大量の乱数を発生させるチカラまかせの方法

- 計算機に莫大な数の乱数を発生させる→パターン生成
- (例 1): 確率分布の乱数の和 → 正規分布?
- (例 2): この回の例題の $\Delta D_{1,2}$ の確率分布

How to generate $\Delta D_{1,2}$ under  is TRUE?

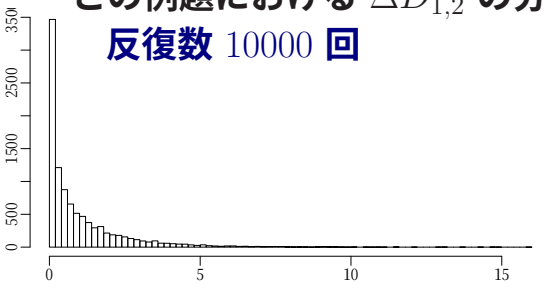
```
> d$y.rnd <- rpois(100, lambda = mean(d$y))
> fit1 <- glm(y.rnd ~ 1, data = d, family = poisson)
> fit2 <- glm(y.rnd ~ x, data = d, family = poisson)
> fit1$deviance - fit2$deviance
```

- `rpois()` によるポアソン乱数の生成 (架空データ)
- 架空データを使って `glm()` あてはめ

パラメトリック・ブートストラップの結果

この例題における $\Delta D_{1,2}$ の分布

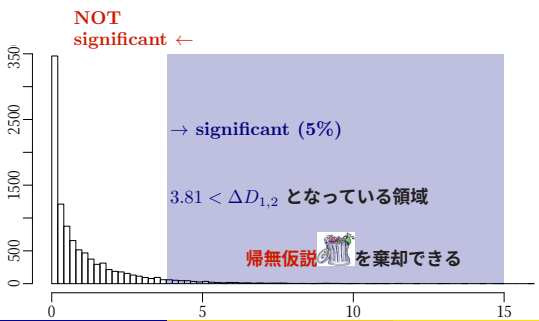
反復数 10000 回



あらかじめ棄却域を決めておく

たとえば 5% とか? — (注) "5%" には 何の意味も正当化もない

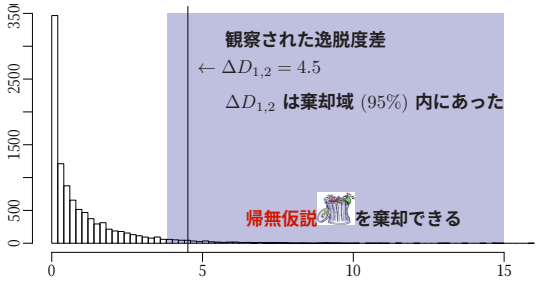
…… てきとーに決めただけ ……



A random $\Delta D_{1,2}$ generator in R

```
get.dd <- function(d) # データの生成と逸脱度差の評価
{
  n.sample <- nrow(d) # データ数
  y.mean <- mean(d$y) # 標本平均
  d$y.rnd <- rpois(n.sample, lambda = y.mean)
  fit1 <- glm(y.rnd ~ 1, data = d, family = poisson)
  fit2 <- glm(y.rnd ~ x, data = d, family = poisson)
  fit1$deviance - fit2$deviance # 逸脱度の差を返す
}
pb <- function(d, n.bootstrap)
{
    replicate(n.bootstrap, get.dd(d))
}
```


Generated distribution of $\Delta D_{1,2} = D_1 - D_2$



constant λ と x model の逸脱度の差 $\Delta D_{1,2}$

(R code is in the next page)

Probability$\{\Delta D_{1,2} \geq 4.5\} = \frac{332}{10000} = 0.0332$

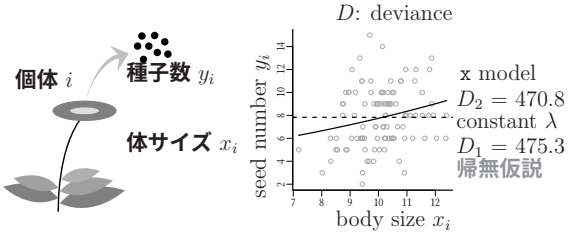
```
> source("pb.R") # reading "pb.R" text file
> dd12 <- pb(d, n.bootstrap = 10000)
> hist(dd12, 100) # to plot histogram
> abline(v = 4.5, lty = 2)
> sum(dd12 >= 4.5)
[1] 332
```

so-called "P-value" is 0.0332.

この例題では 帰無仮説 は棄却された

So we can state that 対立仮説 can be accepted.

x model is better than constant λ.



In case that $P > 0.05$ ...?

何も結論できない ……

λ 一定のモデルが良いとは言えない

検定の非対称性: 帰無仮説 はけっして受容されない

「検定」問題あれこれ

- 統計学的な検定はうまいアイデアだが，誤用も多い
- 帰無仮説は何があっても受容されない
- $p = 0.01$ は $p = 0.0001$ より「えらい」わけではない
- 統計モデルをまちがえると p 値の分布がゆがむ
- 無意味な $p < 0.05$ にこだわるあまり p hacking という詐術が発達 — $p = 0.04$ ぐらい，という論文がやたらと多い

2. 統計学的な検定

そして，その非対称性

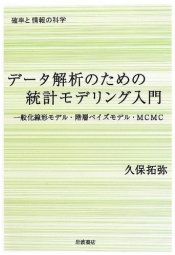
ここでは 尤度比検定 を紹介

今日の内容と「統計モデリング入門」との対応

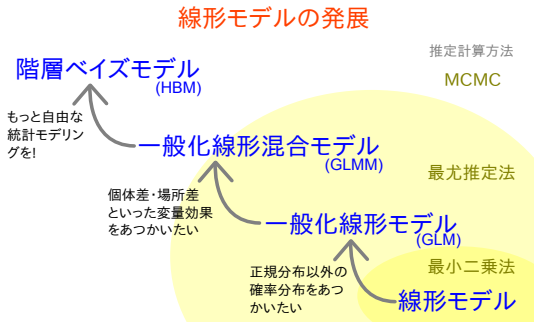
http://goo.gl/Ufq2

今日はおもに「第 6 章 GLM の応用範囲をひろげる」の内容を説明します．

- 著者: 久保拓弥
- 出版社: 岩波書店
- 2012–05–18 刊行



この授業であつかう統計モデルたち



データの特徴にあわせて線形モデルを改良・発展させる

一般化線形モデルって何だろう?

一般化線形モデル (GLM)

- ポアソン回帰 (Poisson regression)
- ロジスティック回帰 (logistic regression)
- 直線回帰 (linear regression)
- ……

一般化線形モデルを作る

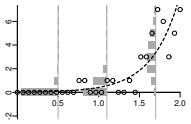
一般化線形モデル (GLM)

- 確率分布は?
- 線形予測子は?
- リンク関数は?

GLM のひとつであるポアソン回帰モデルを指定する

ポアソン回帰のモデル

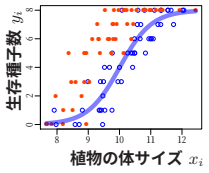
- 確率分布: ポアソン分布
- 線形予測子: e.g., $\beta_1 + \beta_2 x_i$
- リンク関数: 対数リンク関数



GLM のひとつである logistic 回帰モデルを指定する

ロジスティック回帰のモデル

- **確率分布**: **二項分布**
- **線形予測子**: e.g., $\beta_1 + \beta_2 x_i$
- **リンク関数**: logit **リンク関数**



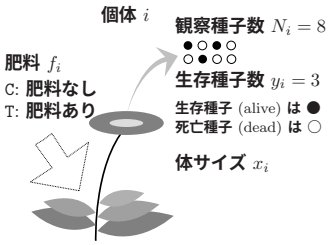
3. “N 個のうち k 個が生きてる” タイプのデータ

上限のあるカウントデータ

$y_i \in \{0, 1, 2, \cdots, 8\}$

またいつもの例題? …… ちょっとちがう

8 個の種子のうち y 個が **発芽可能** だった! …… というデータ



データファイルを読みこむ R

data4a.csv は CSV (comma separated value) format file なので，R で読みこむには以下のようにする:

```
> d <- read.csv("data4a.csv")
```

or

```
> d <- read.csv(
+ "http://hosho.ees.hokudai.ac.jp/~kubo/stat/2015/Fig/binomial/data4a.csv")
```

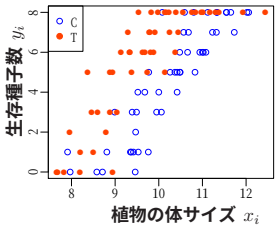
データは d **と名付けられた** data frame (**「表」みたいなもの**) **に格納される**

data frame d を調べる

```
> summary(d)
       N          y               x          f
 Min.   :8   Min.   :0.00   Min.   : 7.660   C:50
 1st Qu.:8   1st Qu.:3.00   1st Qu.: 9.338   T:50
 Median :8   Median :6.00   Median : 9.965
 Mean   :8   Mean   :5.08   Mean   : 9.967
 3rd Qu.:8   3rd Qu.:8.00   3rd Qu.:10.770
 Max.   :8   Max.   :8.00   Max.   :12.440
```


まずはデータを図にしてみる

```
> plot(d$x, d$y, pch = c(21, 19)[d$f])
> legend("topleft", legend = c("C", "T"), pch = c(21, 19))
```



今回は施肥処理 がきいている?

4. ロジスティック回帰の部品

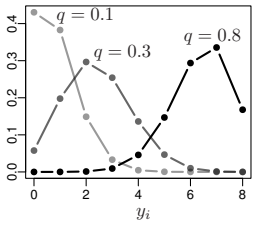
二項分布 binomial distribution と logit link function

二項分布: N 回のうち y 回，となる確率

$$p(y \mid N, q) = \binom{N}{y} q^y (1-q)^{N-y}$$

$\binom{N}{y}$ は「N 個の観察種子の中から y 個の生存種子を選びだす場合の数」

確率 $p(y_i \mid 8, q)$

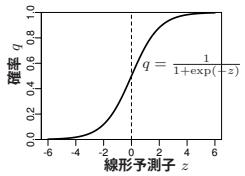


ロジスティック曲線とはこういうもの

ロジスティック関数の関数形 (z_i: 線形予測子，e.g. $z_i = \beta_1 + \beta_2 x_i$)

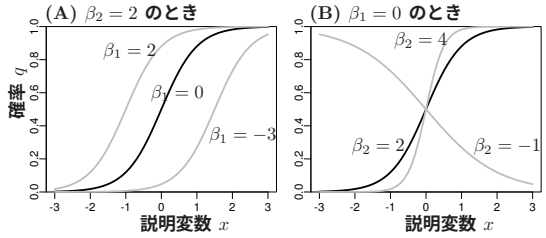
$$q_i = \mathrm{logistic}(z_i) = \frac{1}{1+\exp(-z_i)}$$

```
> logistic <- function(z) 1 / (1 + exp(-z)) # 関数の定義
> z <- seq(-6, 6, 0.1)
> plot(z, logistic(z), type = "l")
```



パラメーターが変化すると……

黒い曲線は $\{\beta_1, \beta_2\} = \{0, 2\}$. (A) $\beta_2 = 2$ と固定して β_1 を変化させた場合．(B) $\beta_1 = 0$ と固定して β_2 を変化させた場合．



パラメーター $\{\beta_1, \beta_2\}$ や説明変数 x がどんな値をとっても確率 q は $0 \le q \le 1$ となる便利な関数

logit link function

- logistic 関数

$$q = \frac{1}{1+\exp(-(\beta_1 + \beta_2 x))} = \mathrm{logistic}(\beta_1 + \beta_2 x)$$

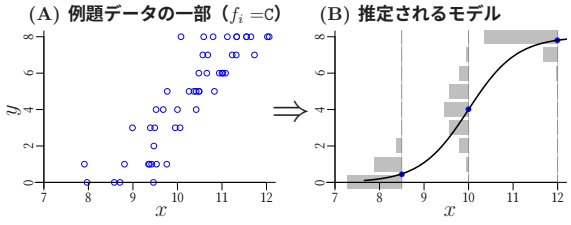
- logit 変換

$$\mathrm{logit}(q) = \log \frac{q}{1-q} = \beta_1 + \beta_2 x$$

logit は logistic の逆関数，logistic は logit の逆関数

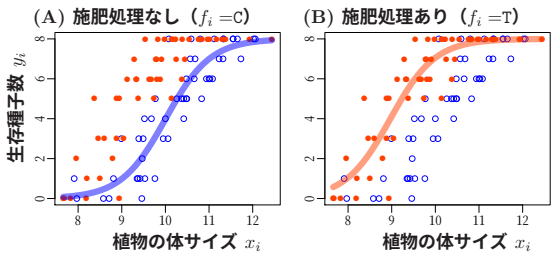
logit is the inverse function of logistic function, vice versa

R でロジスティック回帰 — β_1 と β_2 の最尤推定




```
> glm(cbind(y, N - y) ~ x + f, data = d, family = binomial)
...
Coefficients:
(Intercept)           x          fT
    -19.536       1.952       2.022
```

統計モデルの予測: 施肥処理によって応答が違う



5. ちょっとだけ交互作用項 について

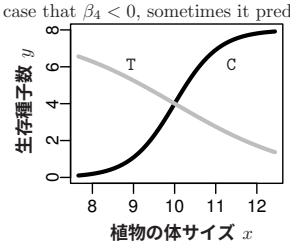
線形予測子の中の複雑な項

ロジスティック回帰を例に

交互作用項とは何か?

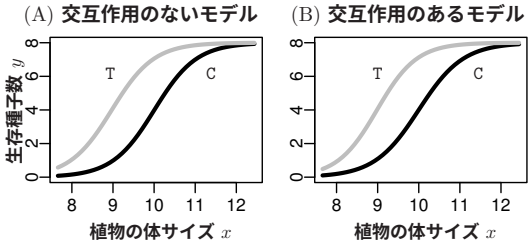
$$\mathrm{logit}(q) = \log \frac{q}{1-q} = \beta_1 + \beta_2 x + \beta_3 f + \beta_4 x f$$

... in case that $\beta_4 < 0$, sometimes it predicts ...



この例題データの場合，交互作用はない

^^I `glm(y ~ x + f, ...)` `glm(y ~ x + f + x:f, ...)`



差がほとんどない

6. 何でも「割算」するな!

「脱」割算の offset 項わざ

ポアソン回帰を強めてみる

割算値ひねくるデータ解析はなぜよくないのか?

- 観測値 / 観測値 がどんな確率分布にしたがうのか見とおしが悪く，さらに説明要因との対応づけが難しくなる
- 情報が失われる: 「10 打数 3 安打」 と「200 打数 60 安打」，「どちらも 3 割バッター」と言ってよいのか?
- 割算値を使わないほうが見とおしのよい， 合理的なデータ解析ができる (今回の授業の主題)
- したがって割算値を使ったデータ解析は不利な点ばかり， そんなことをする必要性はどこにもない

避けられるわりざん

- **避けられる割算値**
  - 確率

    例: N 個のうち k 個にある事象が発生する確率

    対策: ロジスティック回帰など二項分布モデルで
  - 密度などの指数

    例: 人口密度，specific leaf area (SLA) など

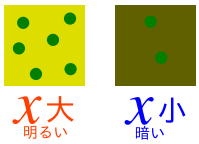
    対策: offset 項わざ — このあと解説!

避けにくいわりざん

- **避けにくい割算値**
  - 測定機器が内部で割算した値を出力する場合
  - 割算値で作図せざるをえない場合があるかも

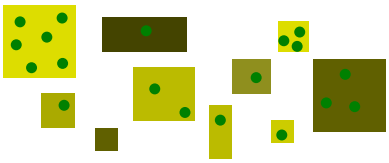
offset 項の例題: 調査区画内の個体密度

- 何か架空の植物個体の密度が「明るさ」 x に応じて どう変わるかを知りたい
- 明るさ は $\{0.1, 0.2, \cdots, 1.0\}$ の 10 段階で観測した



これだけなら単純に `glm(..., family = poisson)` とすればよいのだが ……

「場所によって調査区の面積を変えました」?!



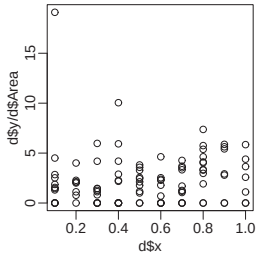
- 明るさ x と面積 A を同時に考慮する必要あり
- ただし「密度 = 個体数 / 面積」といった割算値解析はやらない!
- `glm()` の `offset` 項わざでうまく対処できる
- ともあれその前に観測データを図にしてみる

R の data.frame: 面積 Area, 明るさ x, 個体数 y

```
> load("d2.RData")
> head(d, 8) # 先頭 8 行の表示
       Area   x y
1  0.017249 0.5 0
2  1.217732 0.3 1
3  0.208422 0.4 0
4  2.256265 0.1 0
5  0.794061 0.7 1
6  0.396763 0.1 1
7  1.428059 0.6 1
8  0.791420 0.3 1
```


明るさ vs 割算値図の図

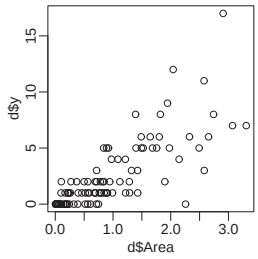
```
> plot(d$x, d$y / d$Area)
```



いまいちよくわからない ……

面積 A vs 個体数 y の図

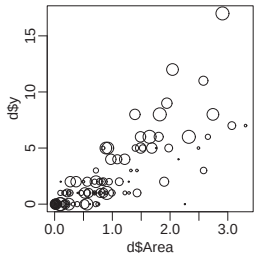
```
> plot(d$Area, d$y)
```



面積 A とともに区画内の個体数 y が増大するようだ

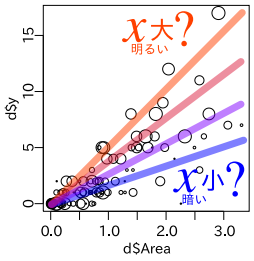
明るさ x の情報 (マルの大きさ) も図に追加

```
> plot(d$Area, d$y, cex = d$x * 2)
```



同じ面積でも明るいほど個体数が多い?

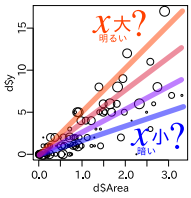
密度が明るさ x に依存する統計モデル



- 区画内の個体数 y の平均は面積 $\times$ 密度
- 密度は明るさ x で変化する

「平均個体数 = 面積 $\times$ 密度」モデル

1. ある区画 i の応答変数 y_i は平均 λ_i のポアソン分布にしたがうと仮定:
   $y_i \sim \mathrm{Pois}(\lambda_i)$
2. 平均値 λ_i は面積 A_i に比例し, 密度は明るさ x_i に依存する

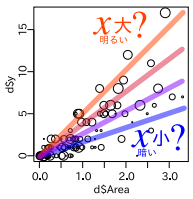


$$\lambda_i = A_i \exp(\beta_1 + \beta_2 x_i)$$

つまり $\lambda_i = \exp(\beta_1 + \beta_2 x_i + \log(A_i))$ となるので
$\log(\lambda_i) = \beta_1 + \beta_2 x_i + \log(A_i)$ 線形予測子は右辺のようになる

このとき $\log(A_i)$ を offset 項とよぶ (係数 β がない)

この問題は GLM であつかえる!



- `family`: `poisson`, ポアソン分布
- `link` 関数: `"log"`
- モデル式 : `y ~ x`
- `offset` 項の指定 : `log(Area)`

- 線形予測子 $z = \beta_1 + \beta_2$ `x` $+$ `log(Area)`
  a, b は推定すべきパラメーター
- 応答変数の平均値を λ とすると $\log(\lambda) = z$
  つまり $\lambda = \exp(z) = \exp(\beta_1 + \beta_2$ `x` $+$ `log(Area)`$)$
- 応答変数 は平均 λ のポアソン分布に従う:

`glm()` 関数の指定

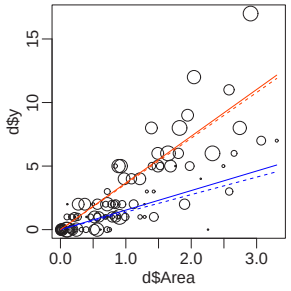
```
fit <- glm(
  y ~ x,
  family = poission(link = "log")
  data = d,
  offset = log(Area)
)
```

結果を格納するオブジェクト
関数名
モデル式
確率分布の指定
offset の指定
リンク関数の指定（省略可）

R の glm() 関数による推定結果

```
> fit <- glm(y ~ x, family = poisson(link = "log"), data = d,
  offset = log(Area))
> print(summary(fit))

Call:
glm(formula = y ~ x, family = poisson(link = "log"), data = d,
    offset = log(Area))

(...略...)

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)    0.321      0.160    2.01    0.044
x              1.090      0.227    4.80  1.6e-06
```


推定結果にもとづく予測を図にしてみる



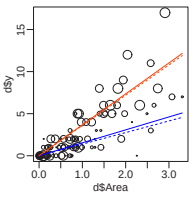
$x = 0.9$
light environment

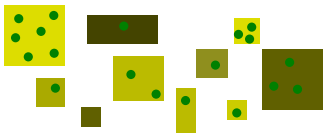
$x = 0.1$
dark environment

- 実線は glm() の推定結果にもとづく予測
- 破線はデータ生成時に指定した関係

まとめ: glm() の offset 項わざで「脱」割算

- 平均値が面積などに比例する場合は， この面積などを offset 項 として指定する
- 平均 = 面積 × 密度，というモデルの密度 を exp(線形予測子) として定式化する





統計モデルを工夫してわりざんやめよう

- 避けられる割算値
  - 確率

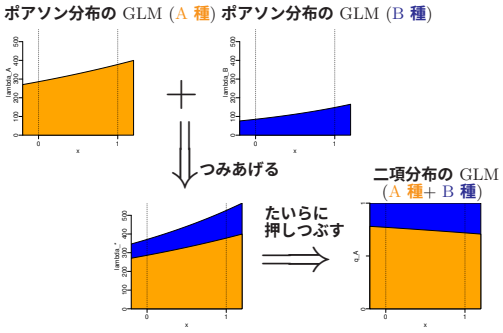
    例: N 個のうち k 個にある事象が発生する確率

    対策: ロジスティック回帰など二項分布モデルで
  - 密度などの指数

    例: 人口密度，specific leaf area (SLA) など

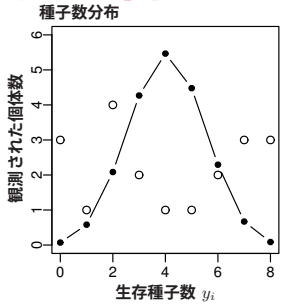
    対策: offset 項わざ — 統計モデリングの工夫!

ポアソン分布 GLM と二項分布 GLM のつながり



三項分布・多項分布で威力を発揮!

次回予告 The next topic



N 個のうち y 個
…という形式のデータなのに
二項分布ではまったく
説明できない?

階層ベイズモデル

Hierarchical Bayesian Model (HBM)

茨城大集中講義 2016 (e)

階層ベイズモデル – 個体差・場所差のモデリング

久保拓弥 kubo@ees.hokudai.ac.jp

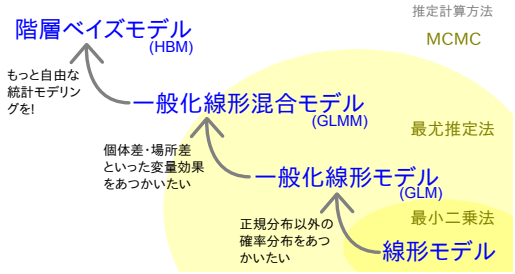
筑波大の講義 http://goo.gl/aFLLHZ

2016–10–06

ファイル更新時刻: 2016–09–29 13:32

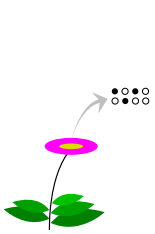
今日の統計モデル: 階層ベイズモデル

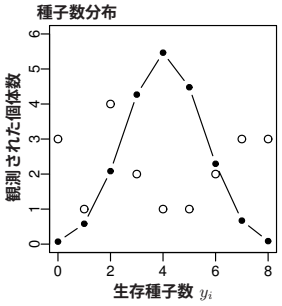
線形モデルの発展



そして Markov Chain Monte Carlo (MCMC)
を使った Bayesian Estimation (ベイズ推定)

GLM ではうまく説明できない観測データ





N 個のうち y 個
…という形式のデータ
なのに
二項分布ではまったく
説明できない?

階層ベイズモデルが必要!

Apply Hierarchical Bayesian Model (HBM)!

今日のハナシ

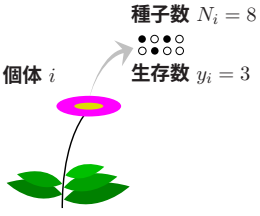
1. MCMC サンプリングのための例題

logistic regression: binomial distribution

and logit link function

例題: 植物の種子の生存確率

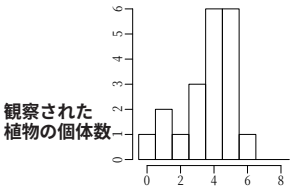
- 架空植物の種子の生存を調べた
- 種子: 生きていれば発芽する
  - どの個体でも 8 個 の種子を調べた
- 生存確率: ある種子が生きている確率



- データ: 植物 20 個体，合計 160 種子の生存の有無を調べた
- 73 種子が生きていた — このデータを統計モデル化したい

たとえばこんなデータが得られたとしましょう

個体ごとの生存数	0	1	2	3	4	5	6	7	8
観察された個体数	1	2	1	3	6	6	1	0	0



生存していた種子数 y_i

これは個体差なしの均質な集団

生存確率 q と二項分布の関係

- 生存確率を推定するために二項分布 という確率分布を使う
- 個体 i の N_i 種子中 y_i 個が生存する確率

$$p(y_i \mid q) = \binom{N_i}{y_i} q^{y_i} (1-q)^{N_i - y_i},$$

- ここで仮定していること
  - 個体差はない
  - つまり すべての個体で同じ生存確率 q

尤度 (ゆうど): 20 個体ぶんのデータが観察される確率

- 観察データ $\{y_i\}$ が確定しているときに
- パラメータ q は値が自由にとりうると考える
- 尤度 は 20 個体ぶんのデータが得られる確率の積，パラメータ q の関数として定義される

$$L(q|\{y_i\}) = \prod_{i=1}^{20} p(y_i \mid q)$$

個体ごとの生存数	0	1	2	3	4	5	6	7	8
観察された個体数	1	2	1	3	6	6	1	0	0

対数尤度方程式と最尤推定

- この尤度 $L(q \mid$ データ$)$ を最大化するパラメータの推定量 $\hat{q}$ を計算したい
- 尤度を対数尤度になおすと

$$\log L(q \mid \text{データ}) = \sum_{i=1}^{20} \log \binom{N_i}{y_i} + \sum_{i=1}^{20} \{y_i \log(q) + (N_i - y_i) \log(1-q)\}$$

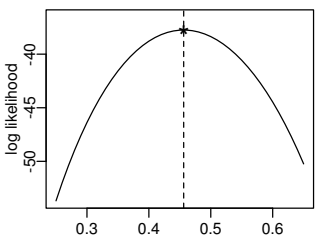
- この対数尤度を最大化するように未知パラメーター q の値を決めてやるのが最尤推定

最尤推定 (MLE) とは何か

- 対数尤度 $L(q \mid$ データ$)$ が最大になるパラメーター q の値をさがしだすこと
- 対数尤度 $\log L(q \mid$ データ$)$ を q で偏微分して 0 となる $\hat{q}$ が対数尤度最大

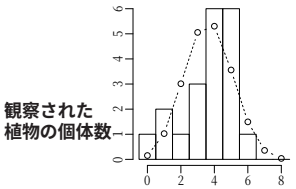
  $\partial \log L(q \mid \text{データ}) / \partial q = 0$
- 生存確率 q が全個体共通の場合の最尤推定量・最尤推定値は

$\hat{q} = \frac{\text{生存種子数}}{\text{調査種子数}} = \frac{73}{160} = 0.456$ ぐらい



二項分布で説明できる 8 種子中 y_i 個の生存

$\hat{q} = 0.46$ なので $\binom{8}{y} 0.46^y \ 0.54^{8-y}$



生存していた種子数 y_i

2. 同じような推定を MCMC でやってみる

最尤推定と Markov chain Monte Carlo (MCMC) はちがう!

そして “なんとなーく” ベイズ統計モデルと関連づけ

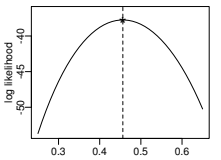
ここでやること: 尤度と MCMC の関係を考える

- さきほどの簡単な例題 (生存確率) のデータ解析を
- 最尤推定ではなく
- Markov chain Monte Carlo (MCMC) 法のひとつである**メトロポリス法** (Metropolis method) であつかう
- 得られる結果: 「パラメーターの値の分布」……??

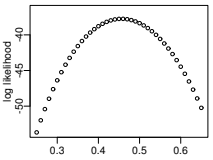
MCMC をもちださなくてもいい簡単すぎる問題
説明のためあえてメトロポリス法を適用してみる

メトロポリス法を説明するための準備

連続的な対数尤度関数 $\log L(q)$



離散化: q がとびとびの値をとる

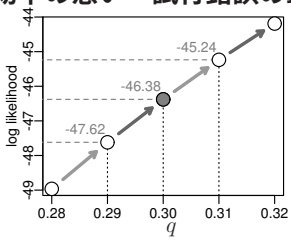


説明を簡単にするため
生存確率 q の軸を離散化する

(実際には離散化する必要などない)

試行錯誤による q の最尤推定値の探索

ちょっと効率の悪い「試行錯誤の最尤推定」



1. q の値の「行き先」を「両隣」どちらかにランダムに決める
2. 「行き先」が現在の尤度より高ければ，q の値をそちらに変更
3. 尤度が変化しなくなるまで (1), (2) をくりかえす

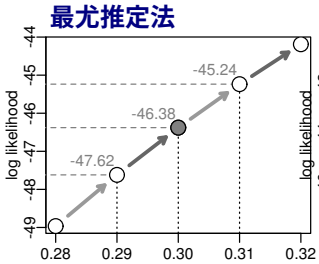
メトロポリス法のルール: この例題の場合

1. パラメーター q の初期値を選ぶ
   (ここでは q の初期値が 0.3)
2. q を増やすか減らすかをランダムに決める
   (新しく選んだ q の値を q_{new} としましょう)
3. q_{new} における尤度 $L(q_{\text{new}})$ ともとの尤度 $L(q)$ を比較
   - $L(q_{\text{new}}) \geq L(q)$ (あてはまり改善): $q \leftarrow q_{\text{new}}$
   - $L(q_{\text{new}}) < L(q)$ (あてはまり改悪):
     - 確率 $r = L(q_{\text{new}})/L(q)$ で $q \leftarrow q_{\text{new}}$
     - 確率 $1 - r$ で q を変更しない
4. 手順 2. にもどる

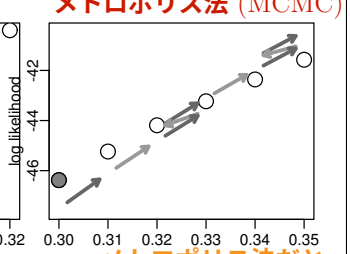
($q = 0.01$ や $q = 0.99$ でどうなるんだ，といった問題は省略)

メトロポリス法のルールで q を動かす

最尤推定法



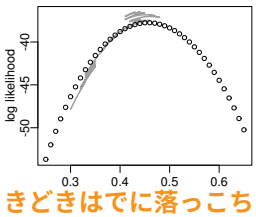
メトロポリス法 (MCMC)



メトロポリス法だと
「単調な山のぼり」にはならない

対数尤度関数の「山」でうろうろする q の値

メトロポリス法 (そして一般の MCMC) は
最適化ではない

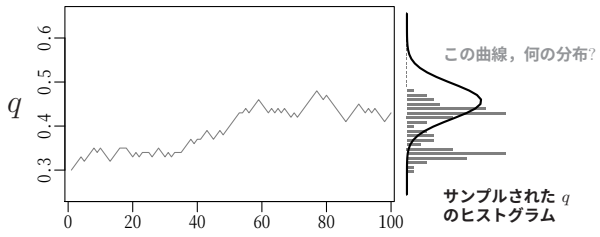


ときどきはでに落っこちる

何のためにこんなことをやるのか?

q の変化していく様子を記録してみよう

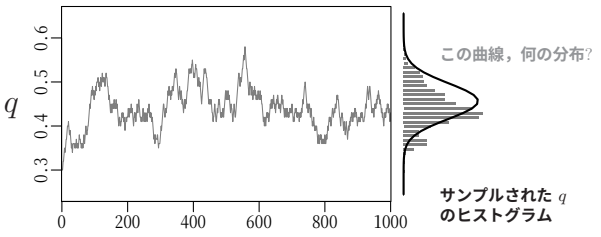
ステップごとに q の値をサンプリング



MCMC 試行錯誤の回数

もっと試行錯誤してみたほうがいいのかな?

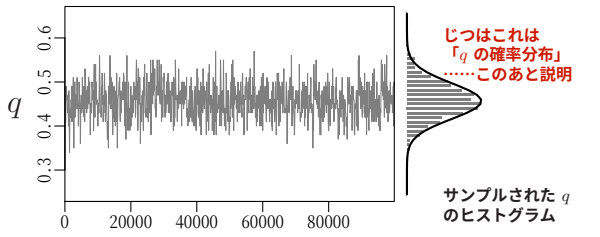
もっと長くサンプリングしてみる



MCMC 試行錯誤の回数

まだまだ……?

もっともっと長くサンプリングしてみる

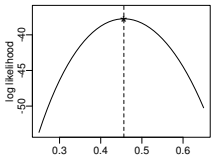


MCMC 試行錯誤の回数

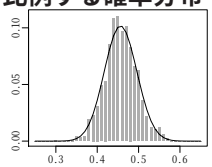
なんだか，ある「山」のかたちにまとまったぞ?

MCMC は何を**サンプリング**している?

対数尤度 $\log L(q)$



尤度 $L(q)$ **に**
比例する確率分布

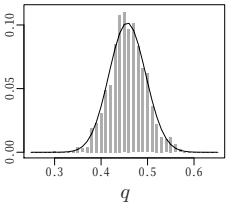


尤度に比例する確率分布からのランダムサンプル

最尤推定はパラメーターの値の点推定

MCMC は "パラメーターの事後分布" (←推定したいこと)
は こういう分布ですよ と推定している

MCMC の結果として得られた q の経験分布

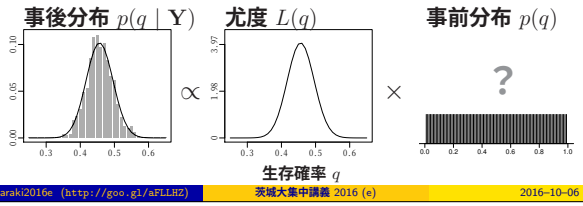


- データと統計モデル (二項分布) を決めて，MCMC サンプリングすると，$p(q)$ からのランダムサンプルが得られる
- このランダムサンプルをもとに，q の平均や 95% 区間などがわかる— **便利じゃないか!**

ベイズ統計モデルの推定

統計モデルとデータにもとづいて事後分布の推定

- パラメーター数の少ないベイズモデルであれば，尤度の数値計算やメトロポリス法で可能
- パラメーター数の多い複雑な統計モデルであれば，**あとで説明する** サンプリングソフトウェアを使用する



3. Softwares for MCMC sampling

"Gibbs sampling" などが簡単にできるような……

事後分布から効率よくサンプリングしたい

統計ソフトウェア R

http://www.r-project.org/

簡単な GLMM なら R だけで推定可能

- R にはいろいろな GLMM の最尤推定関数が準備されている ……
  - `library(glmmML)` の `glmmML()`
  - `library(lme4)` の `lmer()`
  - `library(nlme)` の `nlme()` (正規分布のみ)
- しかし **もうちょっと複雑な** GLMM，たとえば個体差 + 地域差をいれた統計モデルの最尤推定は **かなり難しい** (ヘンな結果が得られたりする)
- 積分がたくさん入っている尤度関数の評価がしんどい

どのようなソフトウェアで MCMC 計算するか?

1. 自作プログラム
   - 利点: 問題にあわせて自由に設計できる
   - 欠点: 階層ベイズモデル用の MCMC プログラミング，けっこうめんどう
2. R のベイズな package
   - 利点: 空間ベイズ統計など便利な専用 package がある
   - 欠点: 汎用性，とぼしい
3. "BUGS" で "Gibbs sampler" なソフトウェア
   - 利点: 幅ひろい問題に適用できて，便利
   - 欠点というほどでもないけど，多少の勉強が必要
   - えーっと "Gibbs sampler" って何?

さまざまな MCMC アルゴリズム

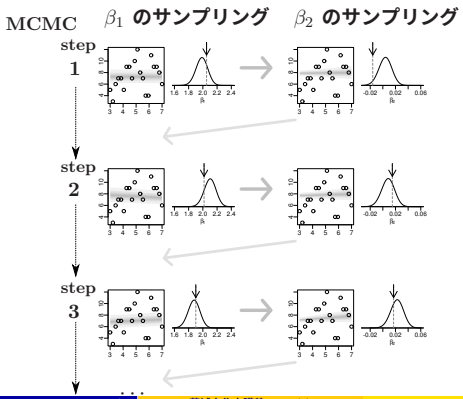
いろいろな MCMC

- **メトロポリス法**: 試行錯誤で値を変化させていく MCMC
  - メトロポリス・ヘイスティングス法: その改良版
- **ギブス・サンプリング**: 条件つき確率分布を使った MCMC
  - 複数の変数 (パラメーター・状態) を効率よくサンプリング

Gibbs sampling とは何か?

- MCMC アルゴリズムのひとつ
- 複数のパラメーターの MCMC サンプリングに使う
- 例: パラメーター β_1 と β_2 の Gibbs sampling
  1. β_2 に何か適当な値を与える
  2. β_2 の値はそのままにして，その条件のもとでの β_1 の MCMC sampling をする (条件つき事後分布)
  3. β_1 の値はそのままにして，その条件のもとでの β_2 の MCMC sampling をする (条件つき事後分布)
  4. 2. – 3. をくりかえす
- 教科書の第 9 章の例題で説明

図解: Gibbs sampling (統計モデリング入門の第 9 章)

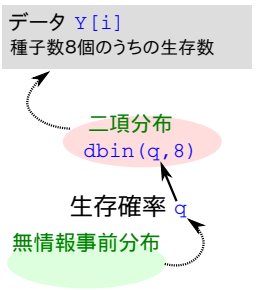


便利な "BUGS" 汎用 Gibbs sampler たち

- BUGS 言語 (+ っぽいもの) でベイズモデルを記述できるソフトウェア
  - WinBUGS — 歴史を変えて……さようなら?
  - OpenBUGS — 予算が足りなくて停滞?
  - JAGS — お手軽で良い，どんな OS でも動く
  - Stan — いま一番の注目
    — 今日は紹介しませんが ……
- リンク集: `http://hosho.ees.hokudai.ac.jp/~kubo/ce/BayesianMcmc.html`

えーと……BUGS 言語って何?

このベイズモデルを BUGS 言語で記述したい



BUGS 言語コード

```
for (i in 1:N.sample) {
  Y[i] ~ dbin(q, 8)
}
q ~ dunif(0.0, 1.0)
```

矢印は手順ではなく，依存関係をあらわしている

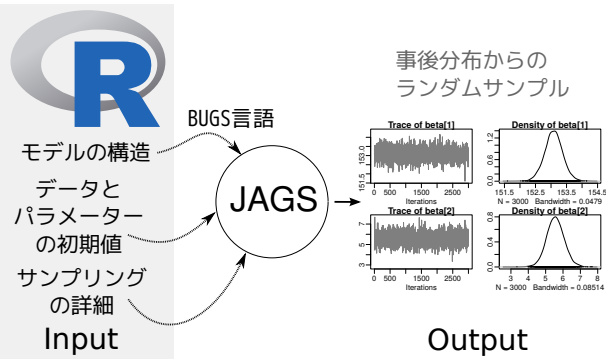
BUGS 言語: ベイズモデルを記述する言語

Spiegelhalter et al. 1995. BUGS: Bayesian Using Gibbs Sampling version 0.50.

いろいろな OS で使える JAGS4.2.0

- R core team のひとり Martyn Plummer さんが開発
  - Just Another Gibbs Sampler
- C++ で実装されている
  - R がインストールされていることが必要
- Linux, Windows, Mac OS X バイナリ版もある
- 開発進行中
- R からの使う: `library(rjags)`

JAGS を R の "したうけ" として使う



R から JAGS にこんなかんじで仕事を命じる (1 / 3)

```
library(rjags)
library(R2WinBUGS) # to use write.model()

model.bugs <- function()
{
  for (i in 1:N.data) {
    Y[i] ~ dbin(q, 8) # 二項分布にしたがう
  }
  q ~ dunif(0.0, 1.0) # q の事前分布は一様分布
}
file.model <- "model.bug.txt"
write.model(model.bugs, file.model) # ファイル出力

# 次につづく……
```

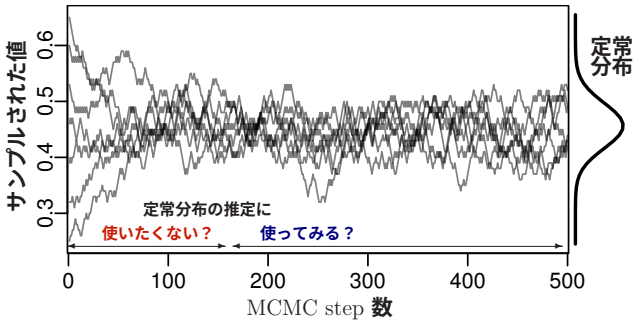

R から JAGS にこんなかんじで仕事を命じる (2 / 3)

```
load("mcmc.RData") # (data.RData ではなく mcmc.RData!!)
list.data <- list(Y = data, N.data = length(data))
inits <- list(q = 0.5)
n.burnin <- 1000
n.chain <- 3
n.thin <- 1
n.iter <- n.thin * 1000

model <- jags.model(
  file = file.model, data = list.data,
  inits = inits, n.chain = n.chain
)

# まだ次につづく……
```

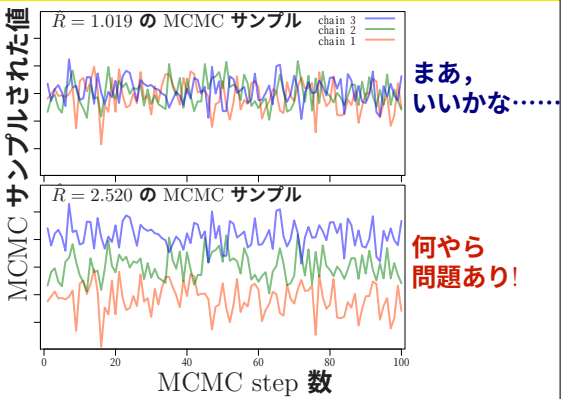

R から JAGS にこんなかんじで仕事を命じる (3 / 3)

```
# burn-in
update(model, n.burnin) # burn in

# サンプリング結果を post.mcmc.list に格納
post.mcmc.list <- coda.samples(
  model = model,
  variable.names = names(inits),
  n.iter = n.iter,
  thin = n.thin
)
#  おわり
```


burn in って何? → 「使いたくない」長さの指定



試行間で差がないかを「診断」する

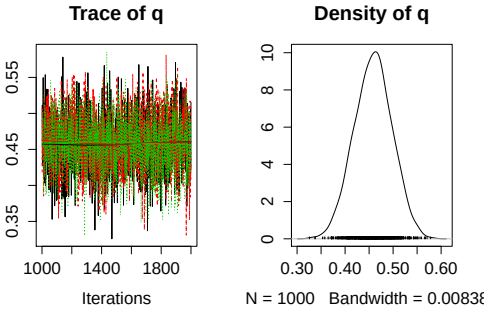


収束診断の $\hat{R}$ 指数

- `gelman.diag(post.mcmc.list)` → 実演表示
- R-hat は Gelman-Rubin の収束判定用の指数
  - $\hat{R} = \sqrt{\frac{\hat{\mathrm{var}}^+(\psi|y)}{W}}$
  - $\hat{\mathrm{var}}^+(\psi|y) = \frac{n-1}{n}W + \frac{1}{n}B$
  - W : サンプル列内の variance の平均
  - B : サンプル列間の variance
  - Gelman et al. 2004. Bayesian Data Analysis. Chapman & Hall/CRC

Gibbs sampling → 事後分布の推定

- `plot(post.mcmc.list)`



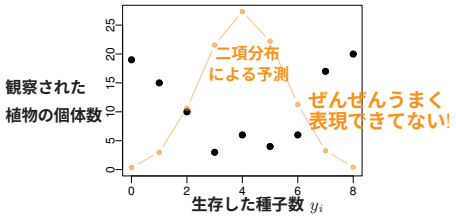
4. 個体差の階層ベイズモデル

個体差のばらつきをあらわす

階層事前分布の設定

二項分布では説明できない観測データ!

100 個体の植物の合計 800 種子中 403 個の生存が見られたので，平均生存確率は 0.50 と推定されたが……

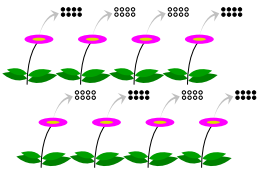


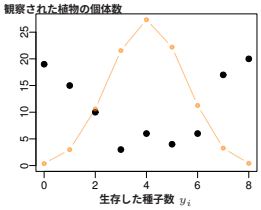
さっきの例題と同じようなデータなのに?

(「統計モデリング入門」第 10 章の最初の例題)

個体差 → 過分散 (overdispersion)

極端な過分散の例





- 種子全体の平均生存確率は 0.5 ぐらいかもしれないが……
- 植物個体ごとに種子の生存確率が異なる: 「個体差」
- 「個体差」があると overdispersion が生じる
- 「個体差」の原因は観測できない・観測されていない

モデリングやりなおし: 個体差を考慮する

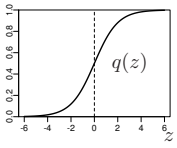
- 生存確率を推定するために 二項分布という確率分布を使う
- 個体 i の N_i 種子中 y_i 個が生存する確率は二項分布

$$p(y_i \mid q_i) = \binom{N_i}{y_i} q_i^{y_i} (1-q_i)^{N_i - y_i},$$

- ここで仮定していること
  - 個体差があるので個体ごとに生存確率 q_i が異なる

GLM わざ: ロジスティック関数で表現する生存確率

- 生存確率 $q_i = q(z_i)$ をロジスティック関数 $q(z) = 1/\{1+\exp(-z)\}$ で表現



- 線形予測子 $z_i = a + r_i$ とする
  - パラメーター a: 全体の平均
  - パラメーター r_i: 個体 i の個体差 (ずれ)

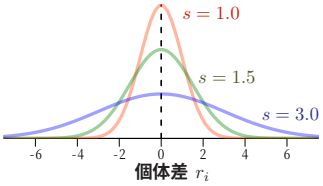
個々の個体差 r_i を最尤推定するのはまずい

パラメーター数 $>$ サンプルサイズ

- 100 個体の生存確率を推定するために**パラメーター 101 個**（a と $\{r_1, r_2, \cdots, r_{100}\}$）を推定すると……
- 個体ごとに生存数 / 種子数を計算していることと同じ！（「データのよみあげ」と同じ）

そこで，次のように考えてみる

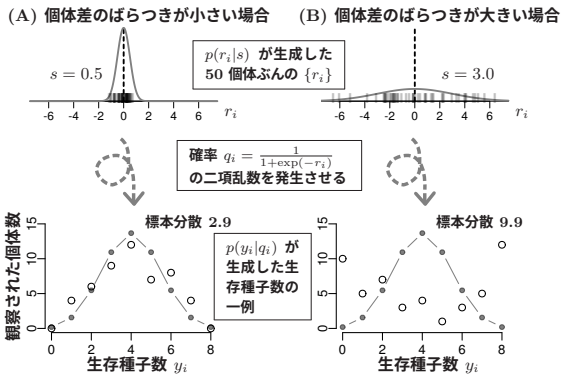
$\{r_i\}$ のばらつきは正規分布だと考えてみる



$$p(r_i \mid s) = \frac{1}{\sqrt{2\pi s^2}} \exp\left(-\frac{r_i^2}{2s^2}\right)$$

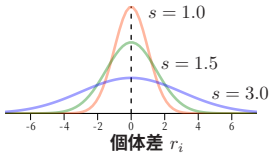
この確率密度 $p(r_i \mid s)$ は r_i の「出現しやすさ」をあらわしていると解釈すればよいでしょう．r_i がゼロにちかい個体はわりと「ありがち」で，r_i の絶対値が大きな個体は相対的に「あまりいない」．

ひとつの例示: 個体差 r_i の分布と過分散の関係



これは r_i の事前分布の指定，ということ

前回の講義で $\{r_i\}$ は正規分布にしたがうと仮定したが
ベイズ統計モデリングでは「100 個の r_i たちに
共通する事前分布として正規分布 を指定した」
ということになる



$$p(r_i \mid s) = \frac{1}{\sqrt{2\pi s^2}} \exp\left(-\frac{r_i^2}{2s^2}\right)$$

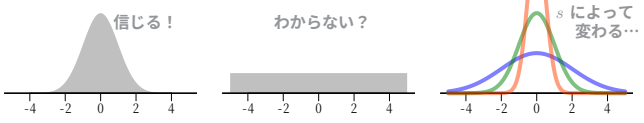
ベイズ統計モデルでよく使われる三種類の事前分布

たいていのベイズ統計モデルでは，ひとつのモデルの中で複数の種類の事前分布を混ぜて使用する．

(A) 主観的な事前分布
（できれば使いたくない!）

(B) 無情報事前分布

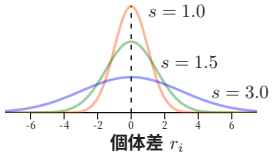
(C) 階層事前分布



r_i の事前分布として階層事前分布を指定する

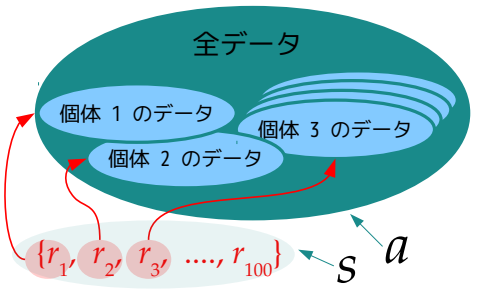
階層事前分布の利点

「データにあわせて」事前分布が変形!



$$p(r_i \mid s) = \frac{1}{\sqrt{2\pi s^2}} \exp\left(-\frac{r_i^2}{2s^2}\right)$$

統計モデルの大域的・局所的なパラメーター



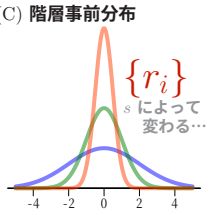
データのどの部分を説明しているのか?

パラメーターごとに適切な事前分布を選ぶ

(B) 無情報事前分布

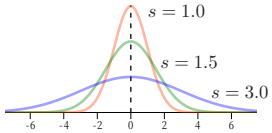
a, s
わからない?

(C) 階層事前分布



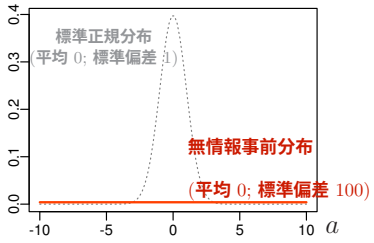
パラメーターの種類	説明する範囲	事前分布
全体に共通する平均・ばらつき	大域的	無情報事前分布
個体・グループごとのずれ	局所的	階層事前分布

個体差 $\{r_i\}$ のばらつき s の無情報事前分布



- s はどのような値をとってもかまわない
- そこで s の事前分布は **無情報事前分布** (non-informative prior) とする
- たとえば一様分布，ここでは $0 < s < 10^4$ の一様分布としてみる

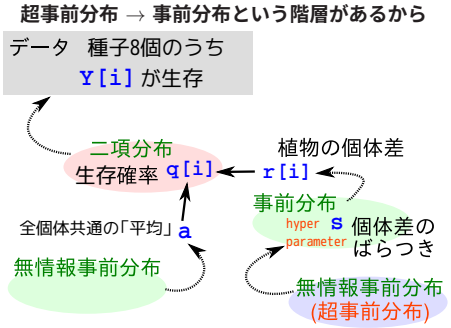
全個体の「切片」 a の無情報事前分布



「生存確率の (logit) 平均 a は何でもよい」と表現している

階層ベイズモデル: 事前分布の階層性

超事前分布 → 事前分布という階層があるから



矢印は手順ではなく，依存関係をあらわしている

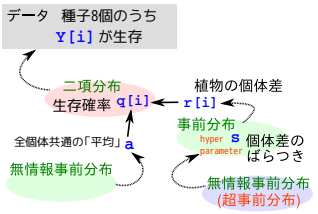
5. 階層ベイズモデルの推定

ソフトウェア JAGS を使ってみる

R の “したうけ” として JAGS を使う

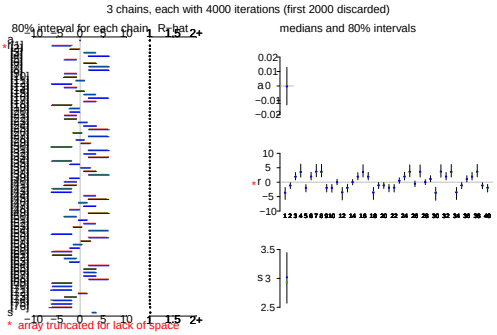
階層ベイズモデルを BUGS コードで記述する

```
model
{
  for (i in 1:N.data) {
    Y[i] ~ dbin(q[i], 8)
    logit(q[i]) <- a + r[i]
  }
  a ~ dnorm(0, 1.0E-4)
  for (i in 1:N.data) {
    r[i] ~ dnorm(0, tau)
  }
  tau <- 1 / (s * s)
  s ~ dunif(0, 1.0E+4)
}
```



JAGS で得られた事後分布サンプルの要約

```
> source("mcmc.list2bugs.R")                    # なんとなく便利なので…
> post.bugs <- mcmc.list2bugs(post.mcmc.list) # bugs クラスに変換
```

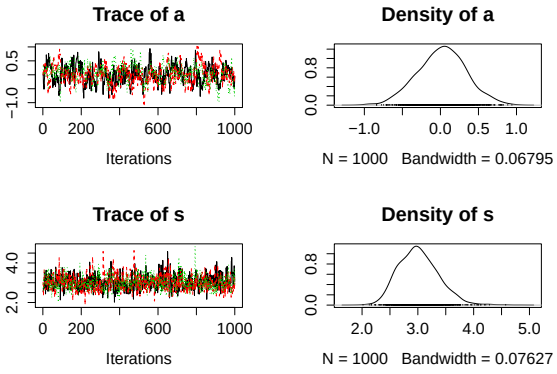


bugs オブジェクトの post.bugs を調べる

- `print(post.bugs, digits.summary = 3)`
- **事後分布の 95% 信頼区間などが表示される**

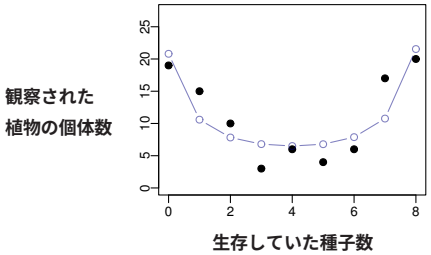
```
3 chains, each with 4000 iterations (first 2000 discarded), n.thin = 2
n.sims = 3000 iterations saved
        mean    sd   2.5%    25%    50%    75%  97.5%  Rhat n.eff
a       0.020 0.321 -0.618 -0.190  0.028  0.236  0.651 1.007   380
s       3.015 0.359  2.406  2.757  2.990  3.235  3.749 1.002  1200
r[1]   -3.778 1.713 -7.619 -4.763 -3.524 -2.568 -1.062 1.001  3000
r[2]   -1.147 0.885 -2.997 -1.700 -1.118 -0.531  0.464 1.001  3000
r[3]    2.014 1.074  0.203  1.282  1.923  2.648  4.410 1.001  3000
r[4]    3.765 1.722  0.998  2.533  3.558  4.840  7.592 1.001  3000
r[5]   -2.108 1.111 -4.480 -2.775 -2.047 -1.342 -0.164 1.001  2300
...（中略）
r[99]   2.054 1.103  0.184  1.270  1.996  2.716  4.414 1.001  3000
r[100] -3.828 1.766 -7.993 -4.829 -3.544 -2.588 -1.082 1.002  1100
```


各パラメーターの事後分布サンプルを R で調べる

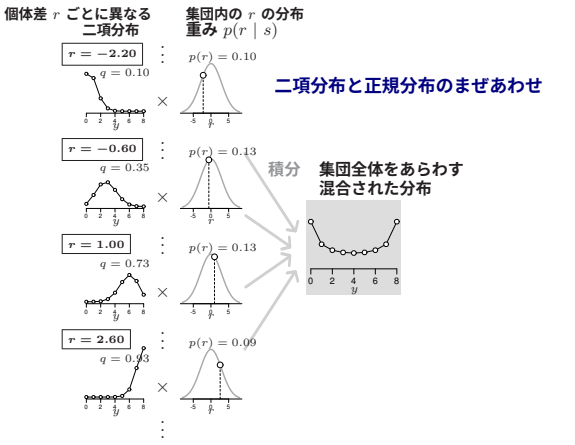


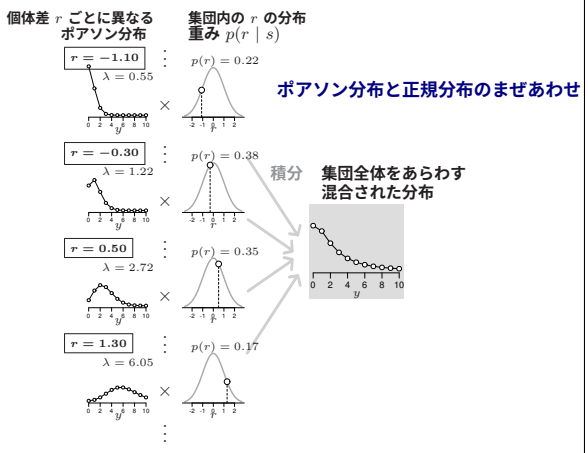
得られた事後分布サンプルを組みあわせて予測

- `post.mcmc <- to.mcmc(post.bugs)`
- **これは `matrix` と同じようにあつかえるので，作図に便利**
- **……このあとごちゃごちゃと計算する必要あるけど，省略……**



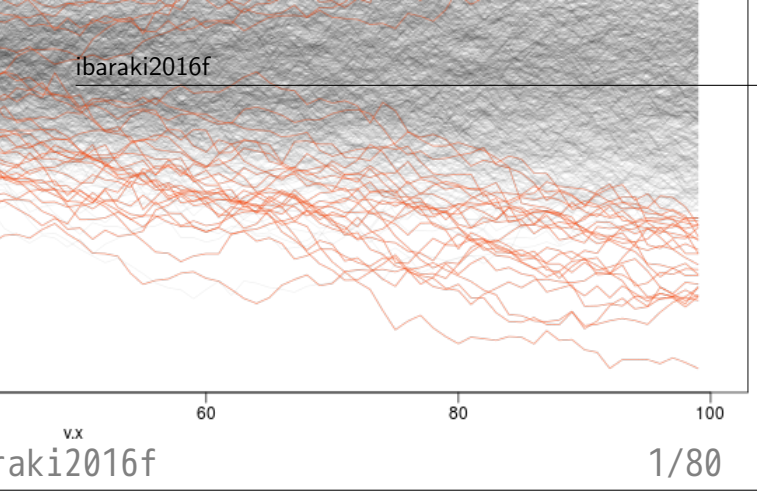
個体差 r_i について積分するということは
二項分布と正規分布をまぜあわせること





次回予告

時間変化データの階層ベイズモデル



（危 2）時系列 Y_t ～

各時刻の個体

（これは次回

データを GLM で

「ゆーいな傾き」を
ねつぞうする原因

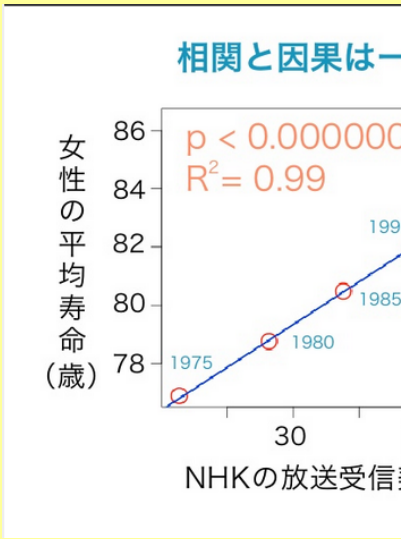
傾きの検定やめて
AIC モデル選択
しても同様になる

とかそういう問題ではない
デルがおかしい?

（危 2）時系列 Y_t

「相関は因果関係

問題の一部：に



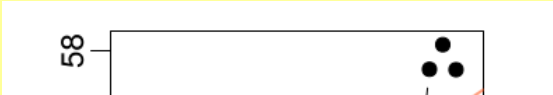
http://www.slideshare.net/takehik

の統計モデリング

」してはいけない

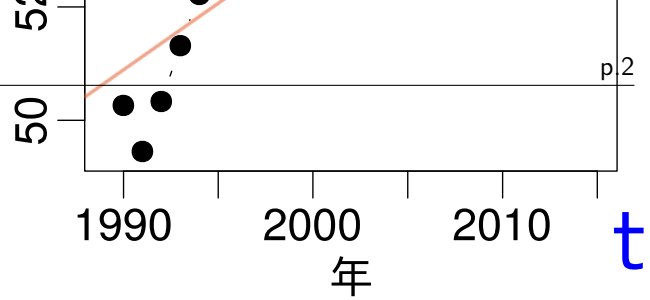
ークモデルが基本

（危 1）時系列デ



連続値ばかり，と
いうことで）

t



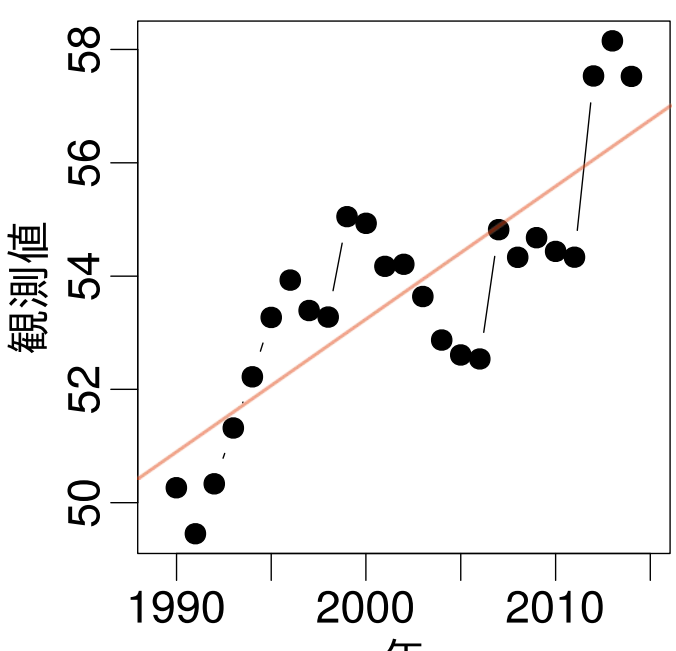
t

ーいだ!!」……??

```
formula = y ~ t))

uals:
1Q    Median        3Q       Max
83  -0.0817    0.9860    2.0188


stimate Std. Error t value Pr(>|t|)
14.5655    71.4761   -5.80  6.6e-06
 0.2339     0.0357    6.55  1.1e-06
```

glm(時系列Y ~ 時間 t)

時系列の各点は

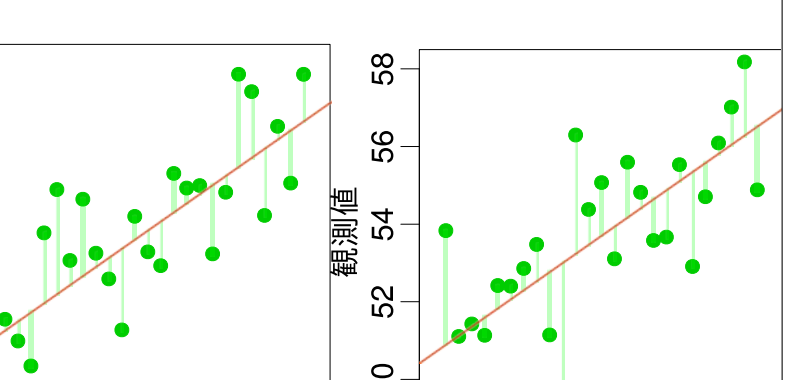


「

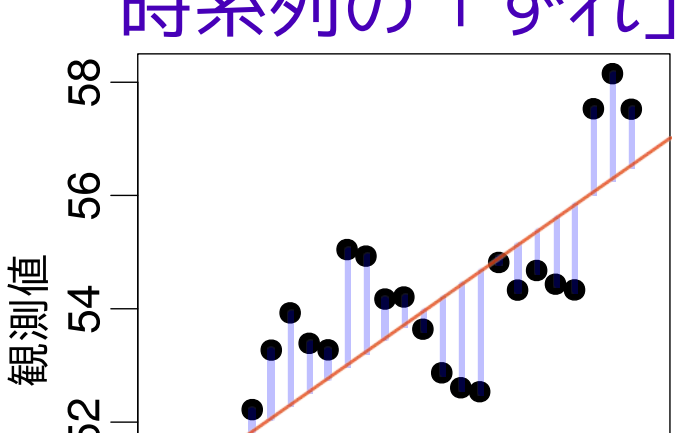
検定とかモデル選択とか

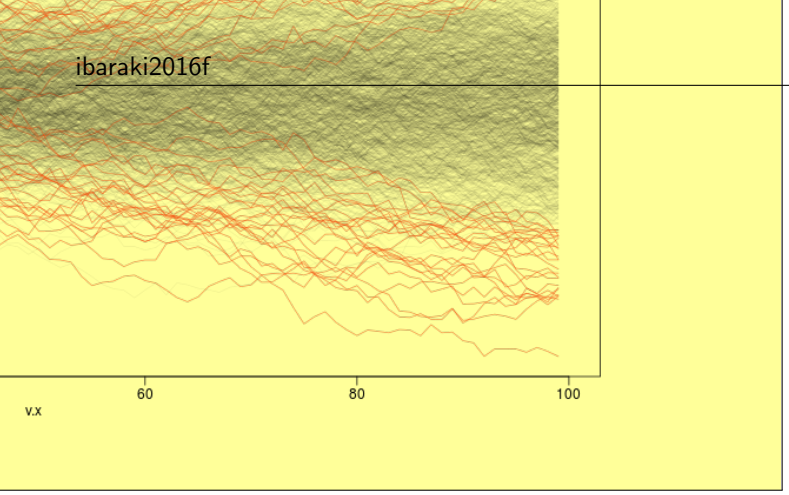
統計モデルと

GLM のずれ

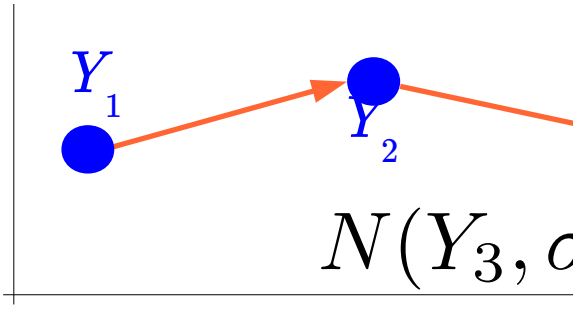


時系列の「ずれ」



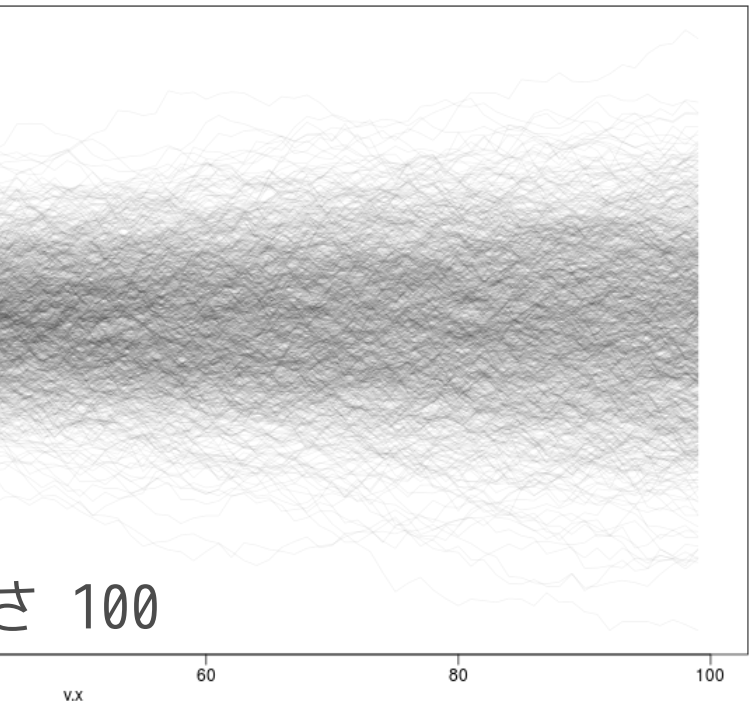


$N(Y_2, \sigma) \rightarrow$



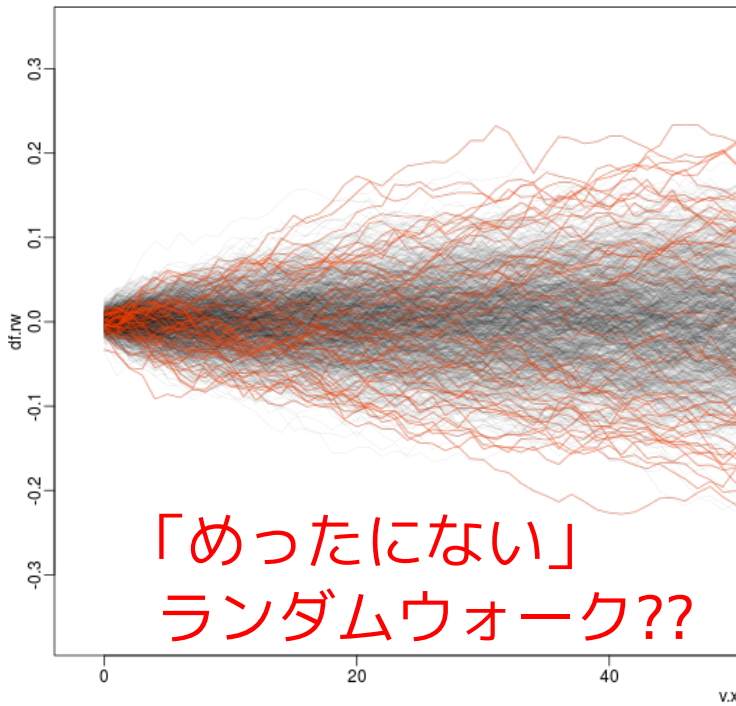
フなサンプル時系列

ほど生成してみました



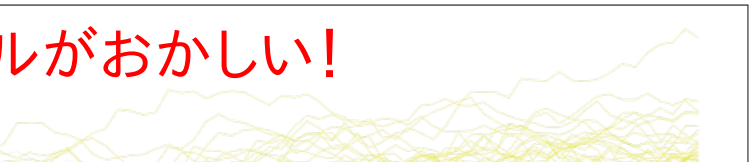
例外的な時系列とし

たとえば $t = 100$ でか



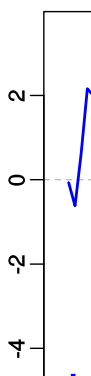
GLM あてはめると…

ての場合で「ゆーい」！



ちょっとでも傾い

実際には
こんなデータ
なのに

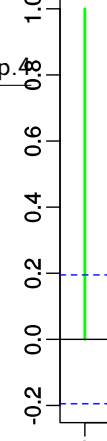


$$\mathrm{ov}(y_t, y_{t-k})$$

ibaraki2016f

$$(y_t) \cdot \mathrm{Var}(y_{t-1})$$

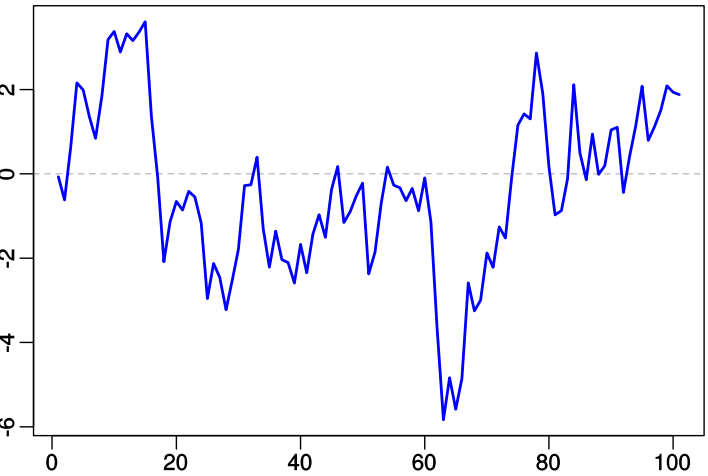
```
plot(acf(ts(Y)))
```

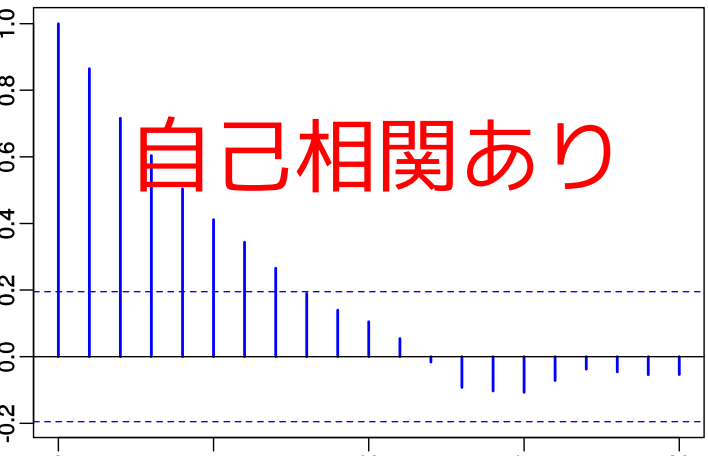


2016-10-06

ibaraki2

の様子を図示





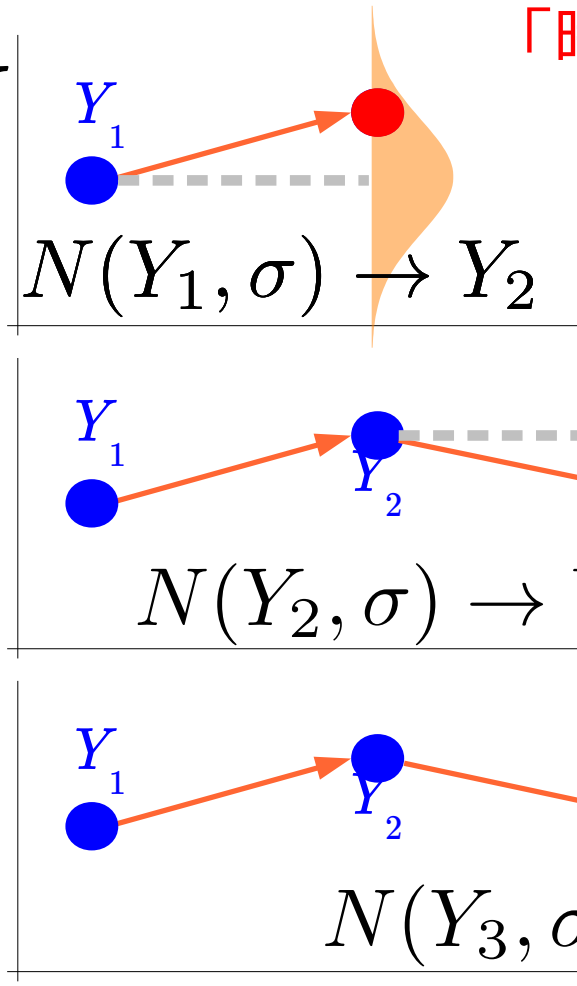
自己相関あり

raki2016f

21780

変数 Y

「日

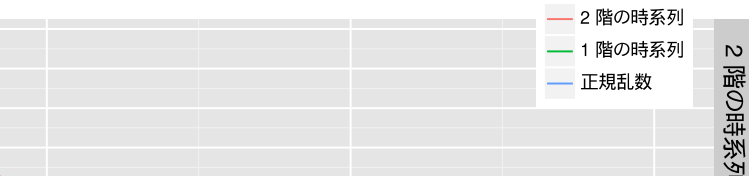


2016-10-06

ibaraki2

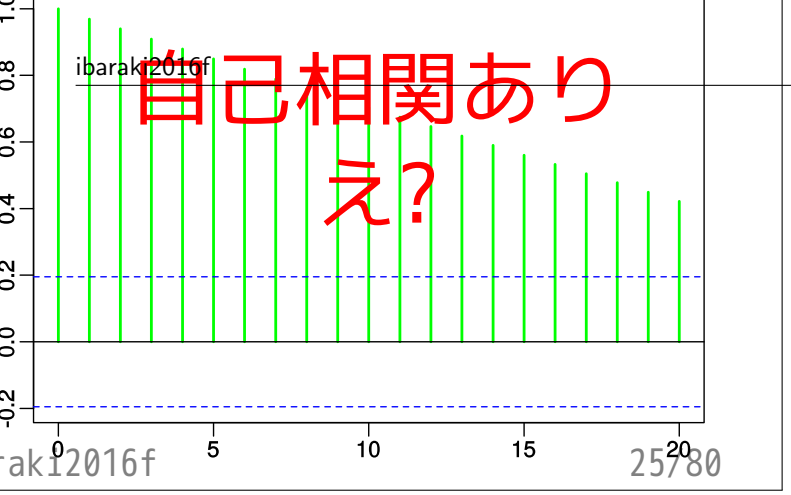
の「差分」をみよう

ど差分を調べるのが基本



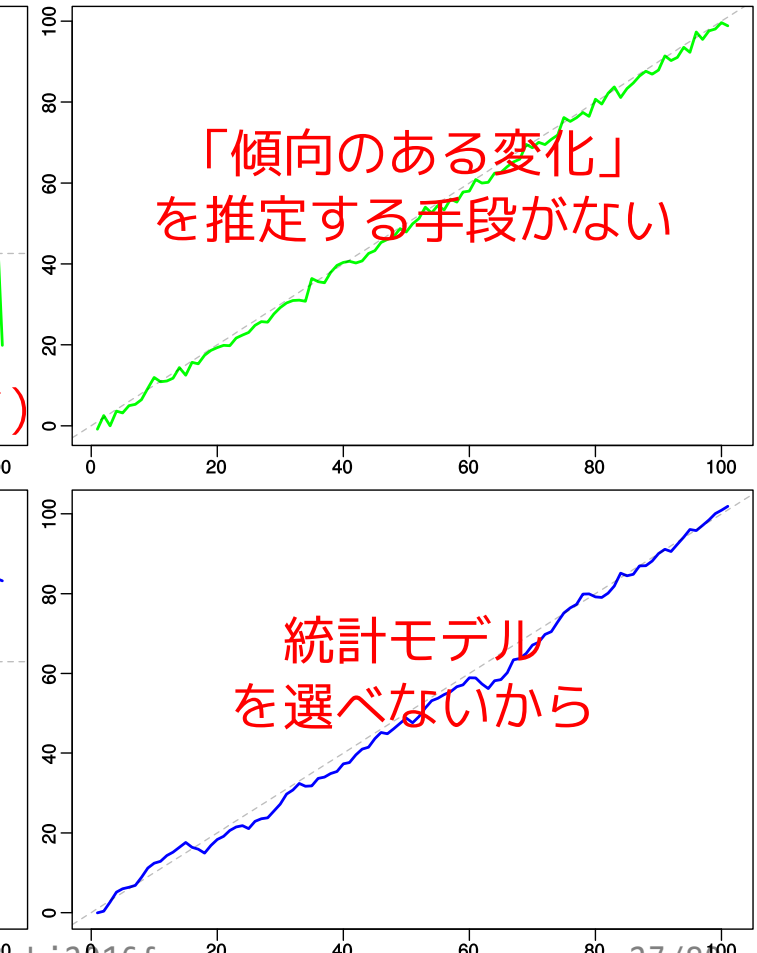
時間的自

いつも役にたつ

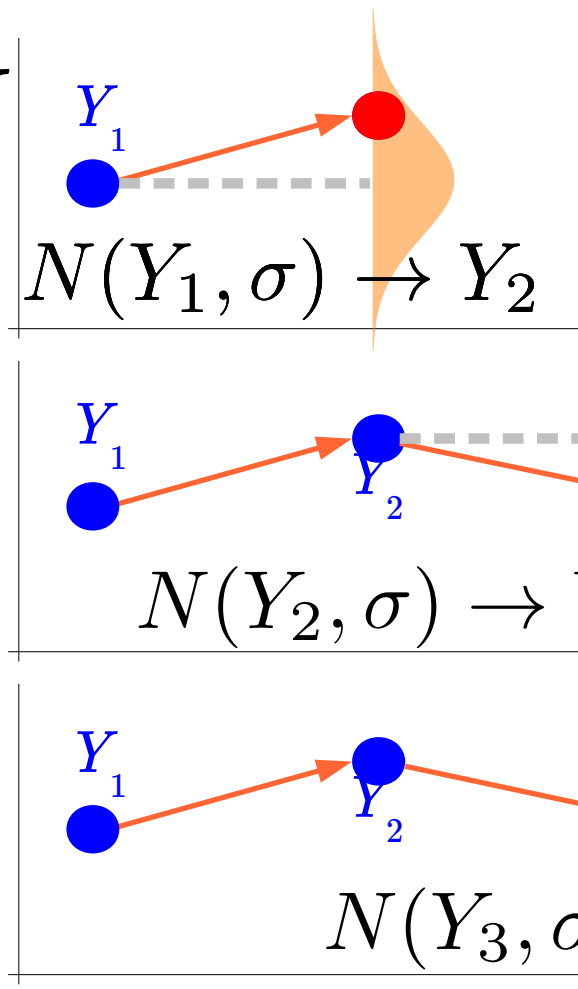


plot(acf(ts(Y)))

ても区別がつかない



変数 Y



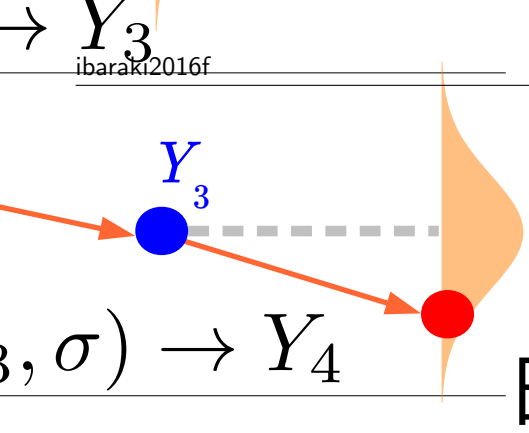
ルでたちむかう

データ解析

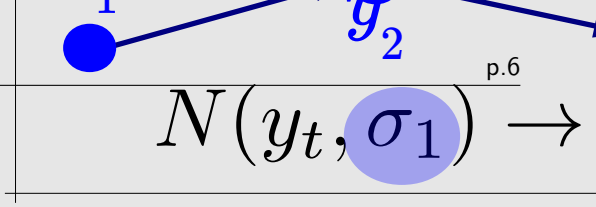
時系列データを

時系列データ解析の

- モデルがあれこ
- 経済学よりのモ



時間 t



$N(y_t, \sigma_1) \rightarrow$

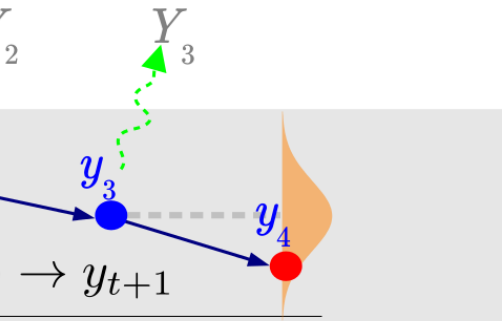
状態変数の変

観測できない世界

誤差

状態空間モデル

$\rightarrow Y_t$ 二種類の σ をもつ

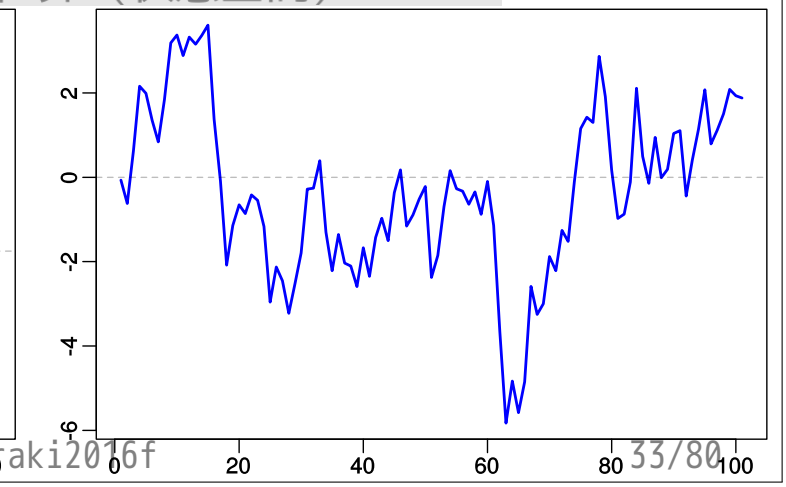


の変化

時間 t

世界（状態空間）

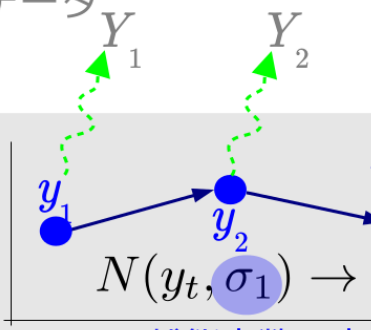
σ_2 小
σ_1 大



観測の誤差

$N(y_t, \sigma_2) \rightarrow$

観測データ

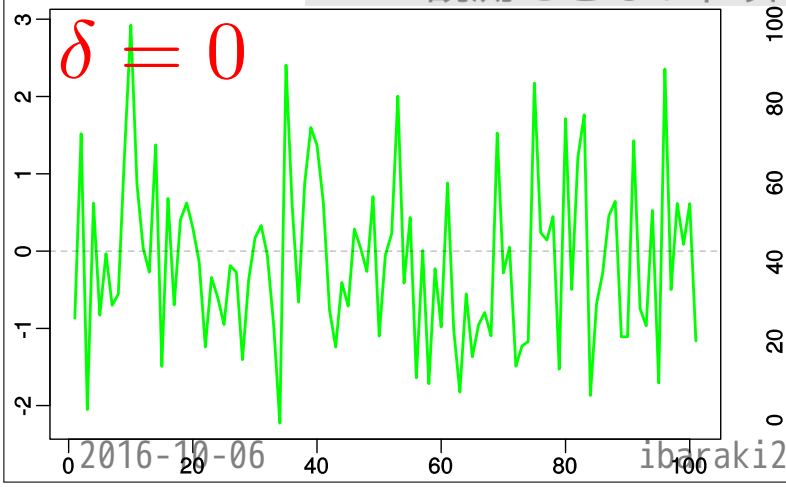


$N(y_t, \sigma_1) \rightarrow$

状態変数の変

観測できない世界

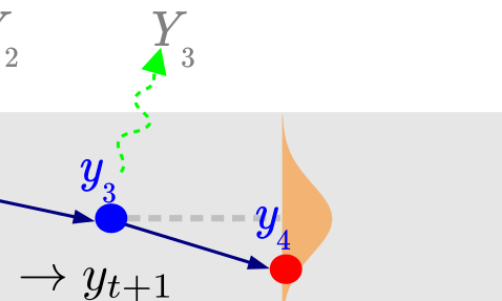
σ_2 大
σ_1 小



誤差

状態空間モデル

$\rightarrow Y_t$ 二種類の σ をもつ



状態空間モデ

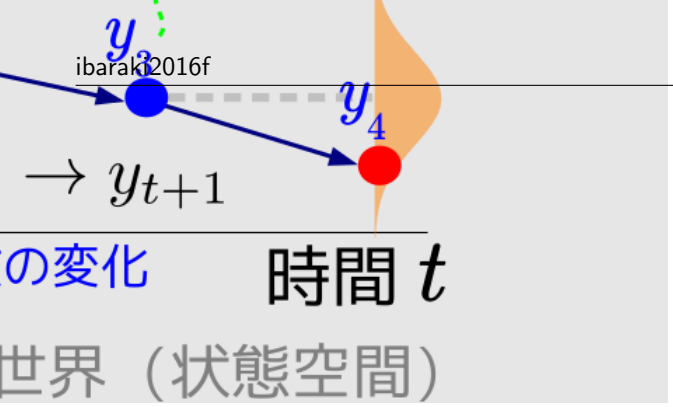
この部分にポアソン分布や

二項分布をいれる

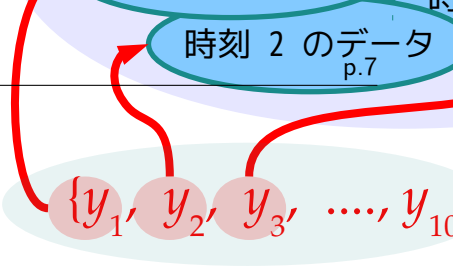
誤差

$N(y_t, \sigma_2) \rightarrow$

観測データ



世界（状態空間）



局所的パラメータ

（たくさんの時点・個体・調査地……

デルをあてはめる?

態空間モデルの

いろいろある

y(dlm)

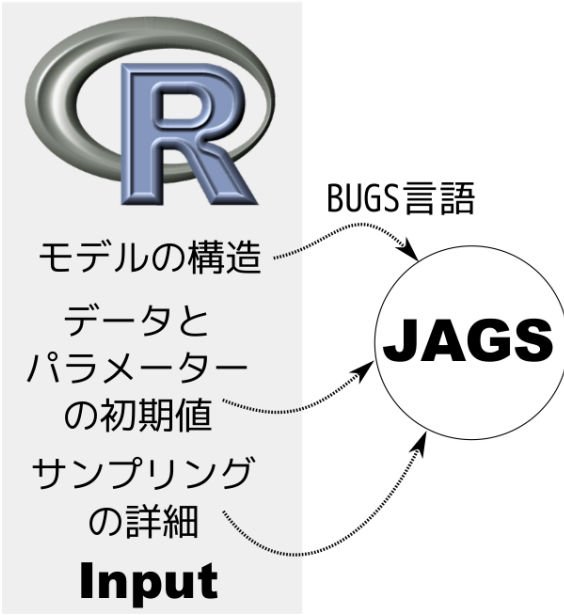
y(KFAS)

めんどう?)

こういう問題

BUGS 言語で

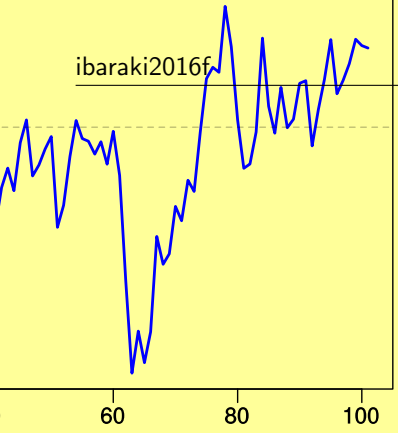
階層ベイズモデ



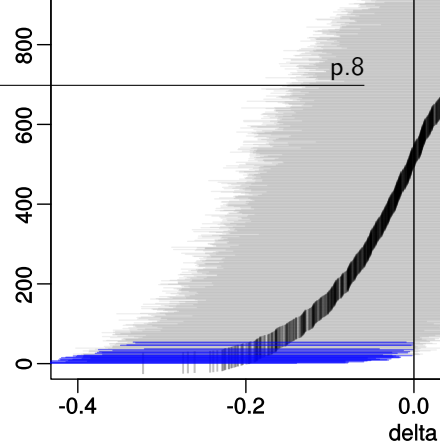

```
e <- 0.0001
, tau[2])
au.Noninformative)
{
y[t], tau[2])
```

1000 個の架空

いろいろなランダムウ



真のδは 0

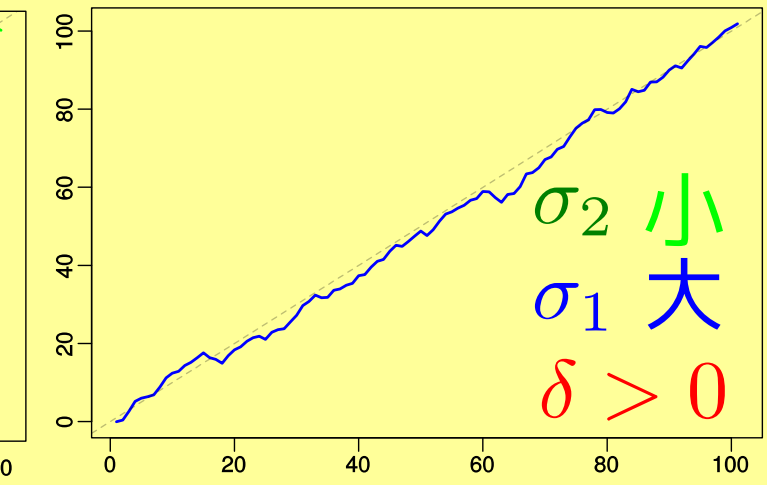


2016-10-06 ibaraki2

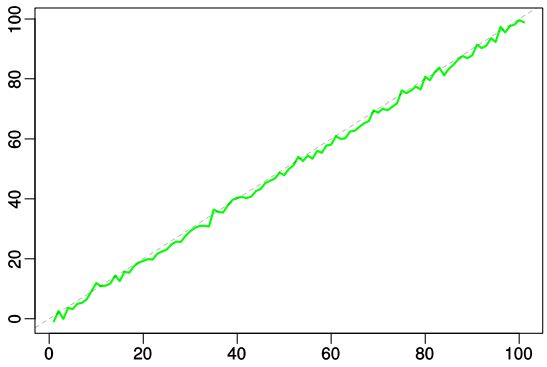
間モデルを

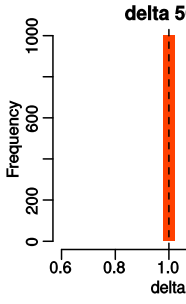
ランダムウォーク

タにあてはめる

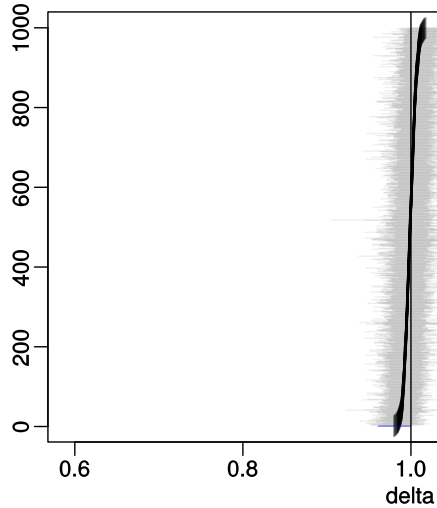


「傾き」δの事後



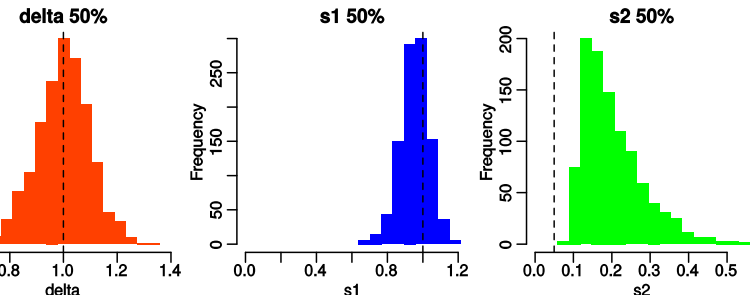


真のδは 1



2016-10-06 ibaraki2

事後分布を見る



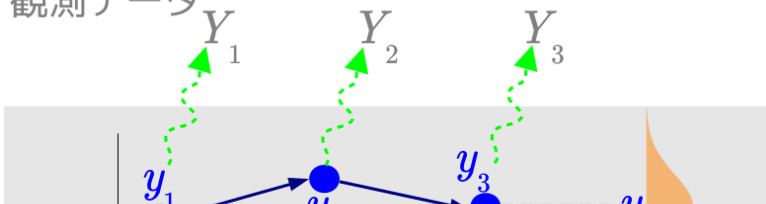
とりあえ

状態空間モデ

観測の誤差

$N(y_t, \sigma_2) \rightarrow Y_t$ 二種類の σ を

観測データ



ら解析」やめよう

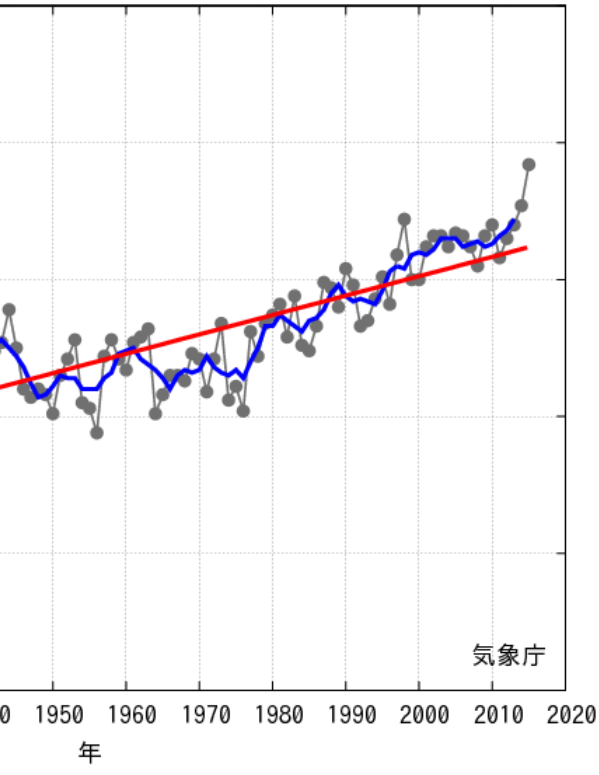
回帰」への対策

ズモデル!

気象庁のデ

向（トレンド）の解説

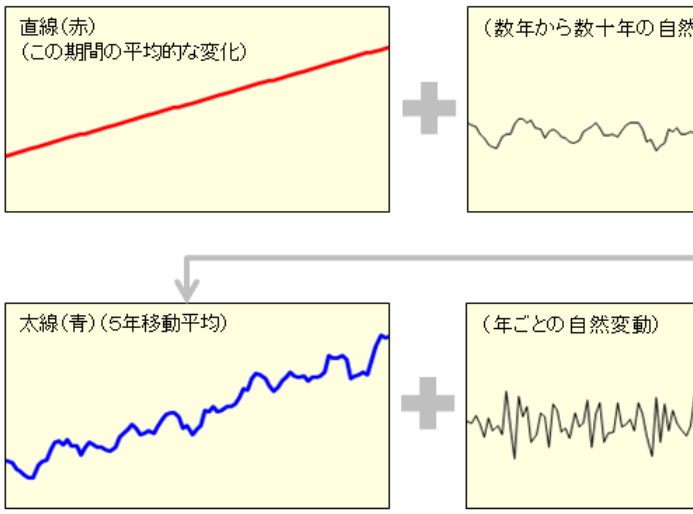
の年平均気温偏差

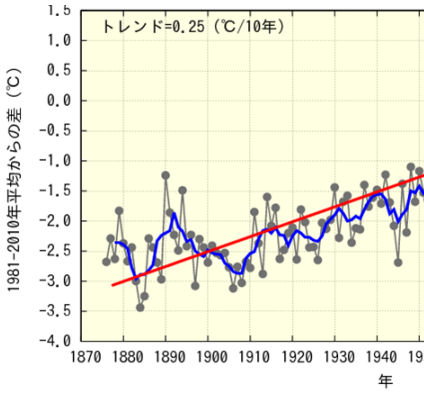


o.jp/cpdinfo/temp/an_wld.html

気象庁の長期変化傾向

http://www.data.jma.go.jp/

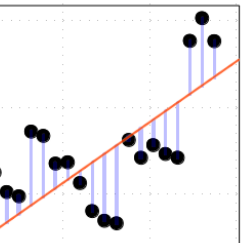




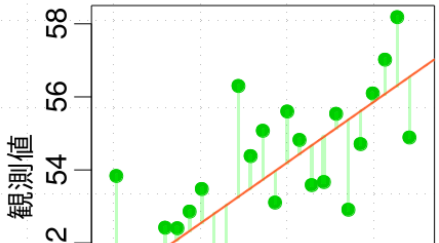
ソッド何がまずいか?

o.jp/cpdinfo/temp/trend.html

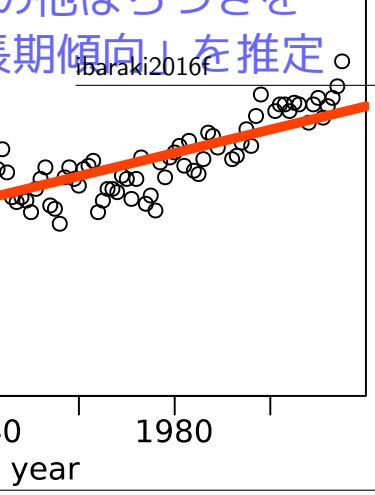
列の「ずれ」



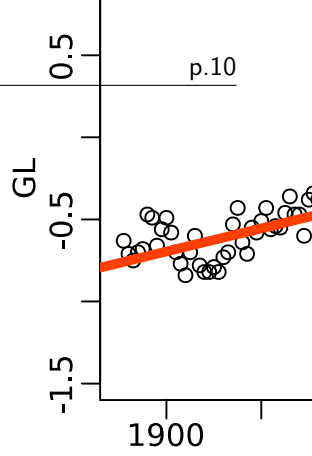
GLM のずれ



公開データを

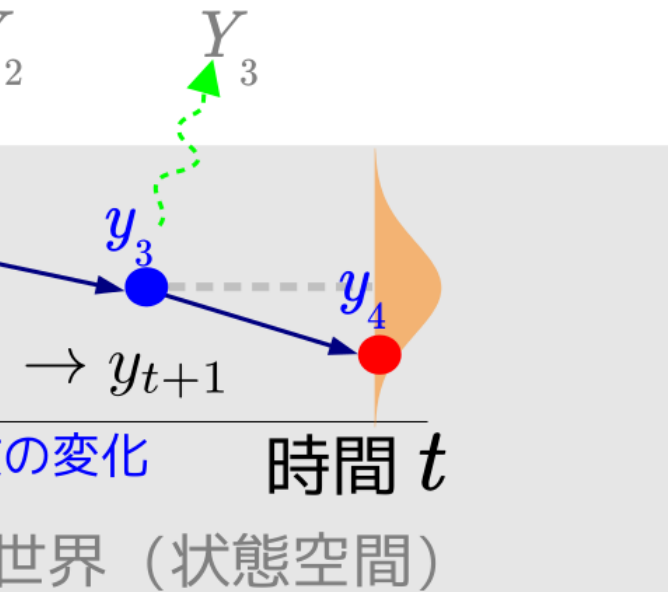


100年
あたり
0.70℃



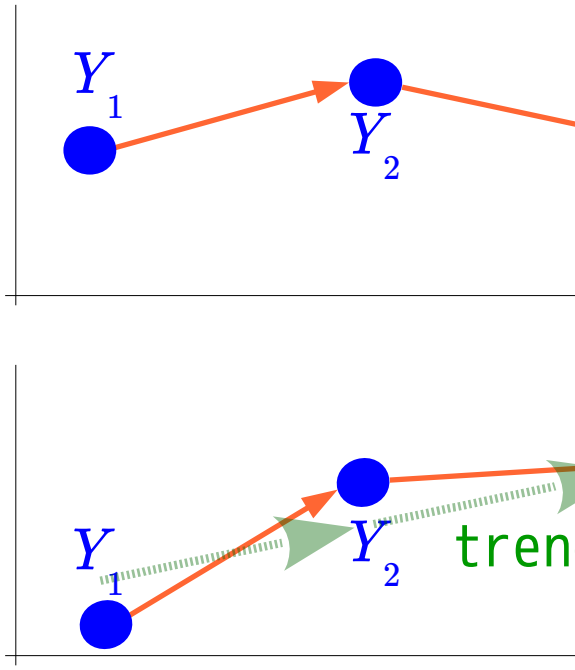
すべてを同時に推定

+各年独立なノイズ



状態空間モデル：すべ

ランダムウォーク+各

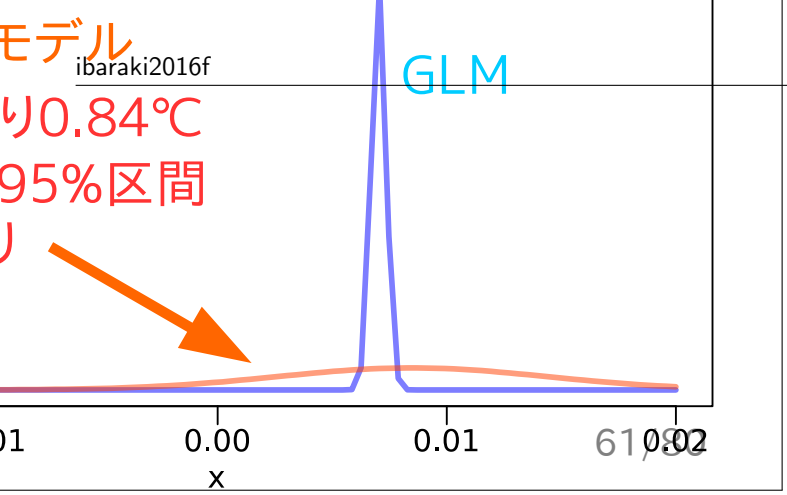


すべてを同時に推定

```
1], tau[2])
0, Tau.Noninformative)
) {
m(y[t], tau[2])
m(m[t], tau[1])
ta + y[t - 1]
```

状態空間モデルが予

```
> summary(glm(GL ~ year, d

Coefficients:
             Estimate Std.
(Intercept) -1.41e+01   6.
```



time series $Y \sim$

の統計モデリング

うがいいこと

とか Y(t) ~ X(t)

測値の四則演算

解析

現 – 再測は時系列

raki2016f 63/80

「見せかけの回帰」

spurious_regression.R

Source on Save Run Sourc

```
1 x <- cumsum(rnorm(100))
2 y <- cumsum(rnorm(100))
3 plot(ts(x), col = "blue", ylim = range(x, y))
4 lines(ts(y), col = "red")
5 print(summary(glm(y ~ x))$coefficients)
```

5:40 (Top Level) R Scri

Console ~/

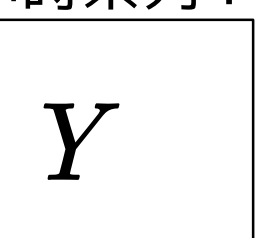
```
> plot(ts(x), col = "blue", ylim = range(x, y))

> lines(ts(y), col = "red")

> print(summary(glm(y ~ x))$coefficients)
             Estimate Std. Error t value   Pr(>|t|)
(Intercept) -1.67120    0.90288 -1.8510 6.7186e-02
x            0.64551    0.10803  5.9753 3.7127e-08
```

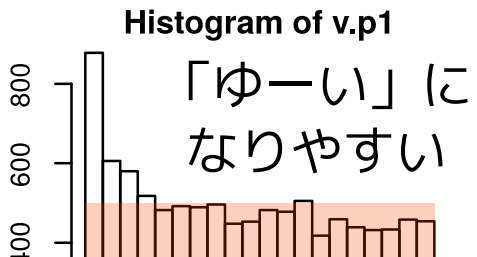
ちょっとだけ実

2016-10-06 ibaraki2

系列にうもれたワナ？

い時系列？





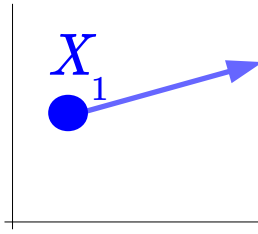
「ゆーい」になりやすい

$Y \sim X$

疑わしい回帰

spurious regression

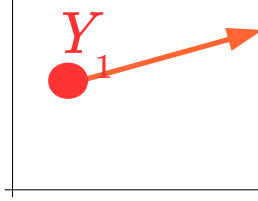
$$X_{t+1} \sim N(X_t, s_x)$$



ンダムウォーク

X_t は独立

$$Y_{t+1} \sim N(Y_t, s_y)$$ 一変量



二変量の正規

Bivariate case

In the 2-dimensional nonsingular case (k = rank(Σ) = $[X\ Y]'$ is:

$$f(x,y) = \frac{1}{2\pi\sigma_X\sigma_Y\sqrt{1-\rho^2}} \exp\left(-\frac{1}{2(1-\rho^2)}\left[\frac{(x-}{\sigma}\right.\right.$$

where ρ is the correlation between X and Y and where

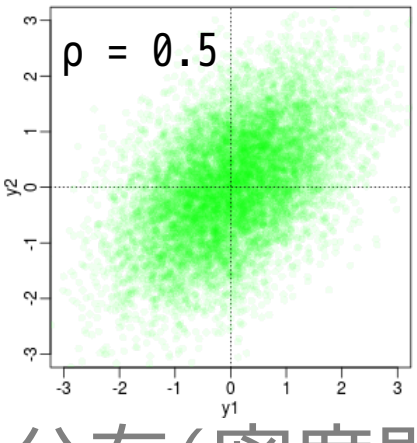
$$\boldsymbol{\mu} = \begin{pmatrix} \mu_X \\ \mu_Y \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} \sigma_X^2 & \rho\sigma_X\sigma_Y \\ \rho\sigma_X\sigma_Y & \sigma_Y^2 \end{pmatrix}.$$

分散共分散行列

https://en.wikipedia.org/wiki/Mult

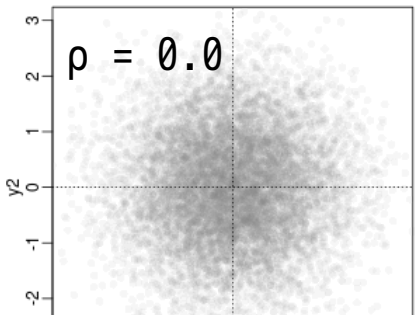
で

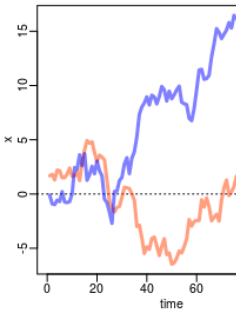
ないか?



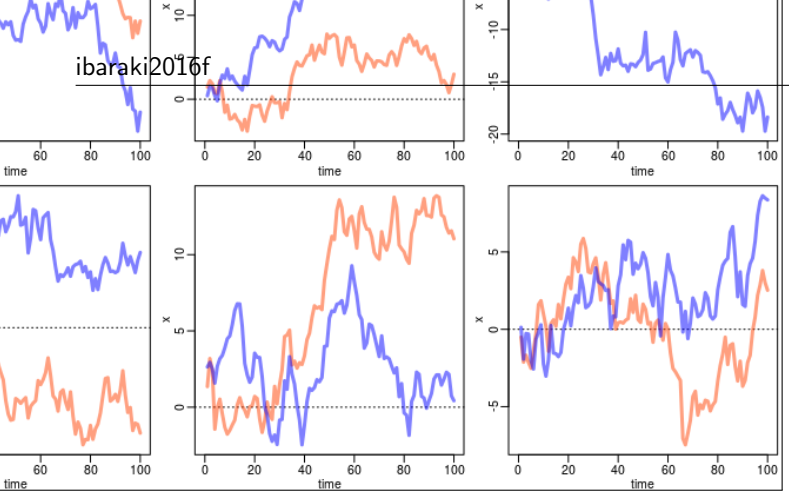
正の相関

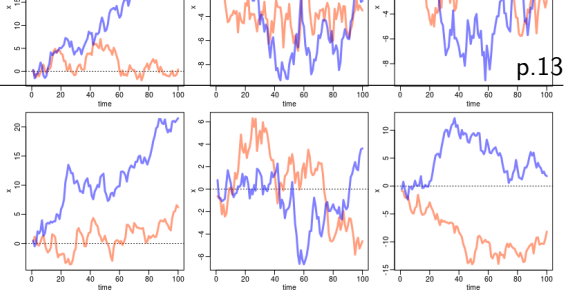
二変量正規分布とラン





確率分布(密度関数)





ρ = 0.0

2016-10-06

品とする状態空間モデル

```
rm(mu[1:2], Omega[1:2, 1:2])

4, 1.0E+4)
4, 1.0E+4)
nverse(VarCov[1:2, 1:2])
a[1] * sigma[1]
a[1] * sigma[2] * rho
a[2] * sigma[1] * rho
a[2] * sigma[2]
, 1.0E+4)
, 1.0E+4)
0)
```

で実演）

75/80

階層ベイズモ

状態空間

から得られた

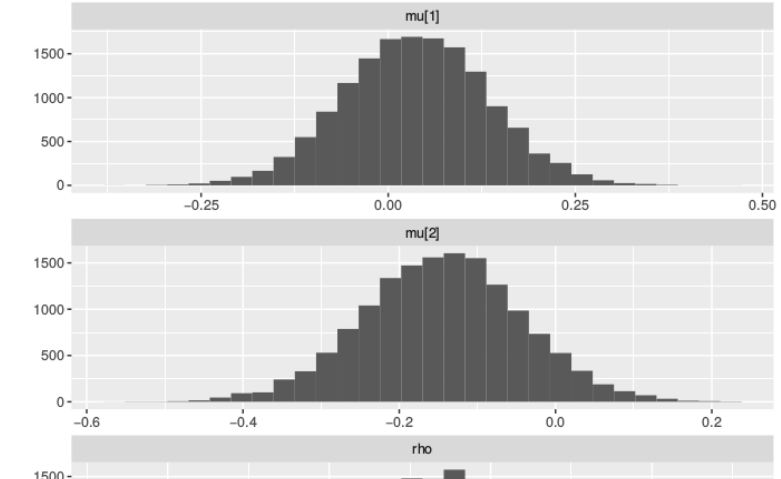
```
 3 chains, each with 5200 iterations (
 n.sims = 15000 iterations saved
           mean    sd   2.5%    25%
mu[1]    -0.122 0.110 -0.342 -0.195 -0
mu[2]    -0.157 0.100 -0.355 -0.224 -0
sigma[1]  1.091 0.079  0.949  1.036  1
sigma[2]  0.993 0.074  0.864  0.941  0
rho       0.568 0.070  0.420  0.523  0
```

ふたつの時系列デ

相関しているかどう

図示すると

2016-10-06



追加スライド

状態空間モデルの

パラメーターの

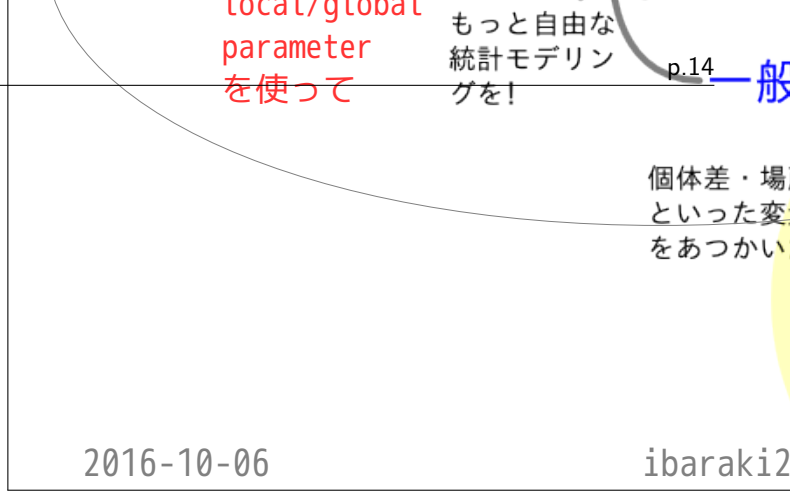
事後分布（ρ = 0.5）

生成する時系列

識する

デルで推定

状態空間モデル



R の練習 (r1) 2016–10–07

久保拓弥 kubo@ees.hokudai.ac.jp

この授業の web page: http://goo.gl/aFLLHZ

統計ソフトウェア R は研究にたいへん役にたつ free software (無料で入手でき，しかも内部を自由に調べられる) です．今回は R のデータ操作・作図の基本わざを説明します．

R を使ったデータ解析の基本的な流れは次のようになります:

1. データを読みこむ (データフレイム data.frame を作る)
2. 読みこんだデータをいろいろ整理する (データフレイムの操作)
3. データをさまざまな方法で**図示**する
4. 統計モデリングの設計・あてはめを行う
5. あてはめの結果やモデルの予測を**図示**する
6. 解析結果をさまざまな方法で出力し，保存する

今日は時間も限られているので，データの読みこみ，基本的ななデータフレイム操作，簡単な図示について説明します．上述の授業 web site のあちこちを見て，さらに発展したわざも勉強してください．

1 R でデータフレイムの操作

1.1 データを読みこんで data.frame を作り，それを表示する

```
> d <- read.csv("data.csv")
> d
   treatment size seed
1    control 21.3    9
2       trtX 24.2   19
3    control 12.0    1
4       trtX 16.1    4
5    control 21.8   13
6       trtX 20.2    6
7    control 22.7    8
8       trtX 23.8    8
9    control 19.5    7
10      trtX 26.4   22
11   control 20.1    3
12      trtX 27.3   31
```

```
13   control 22.5   14
14      trtX 21.8   19
15   control 18.6    4
16      trtX 25.3   26
17   control 23.5   11
18      trtX 19.7    6
19   control 27.9   22
20      trtX 22.0   17
```

```
> head(d) # 最初の 6 行が表示される
  treatment size seed
1   control 21.3    9
2      trtX 24.2   19
3   control 12.0    1
4      trtX 16.1    4
5   control 21.8   13
6      trtX 20.2    6
```

```
> head(d, 3) # 最初の 3 行が表示される
  treatment size seed
1   control 21.3    9
2      trtX 24.2   19
3   control 12.0    1
```

```
> tail(d, 3) # 最後の 3 行が表示される
   treatment size seed
18      trtX 19.7    6
19   control 27.9   22
20      trtX 22.0   17
```

```
> edit(d) # d を編集する
```

1.2 data.frame から行と列をとりだす

```
> d[1:3,] # 1 行めから 3 行めをとりだす
  treatment size seed
1   control 21.3    9
2      trtX 24.2   19
3   control 12.0    1
```

```
> d[c(1, 3, 5),] # 1, 3, 5 行めをとりだす
  treatment size seed
```

```
1   control 21.3    9
3   control 12.0    1
5   control 21.8   13

> d[, 1] # 1 列めをとりだす
 [1] control trtX    control trtX    control trtX  ... 略
Levels: control trtX

> d[4:6, 2:3] # 4-6 行めの 2-3 列めをとりだす
  size seed
4 16.1    4
5 21.8   13
6 20.2    6

# 列の選びかたに 3 とおりある（どれも重要）
> d[, 3] # 3 列めをとりだす
 [1]  9 19  1  4 13  6  8  8  7 22  3 31 14 19  4 26 11  6 22 17

> d$seed # 上とおなじことをやっている
 [1]  9 19  1  4 13  6  8  8  7 22  3 31 14 19  4 26 11  6 22 17

> d[, "seed"] # これも同じ
 [1]  9 19  1  4 13  6  8  8  7 22  3 31 14 19  4 26 11  6 22 17
```

1.3 data.frame から条件つきデータとりだし

treatment が trtX のデータ

```
> d[d$treatment == "trtX",]
   treatment size seed
2       trtX 24.2   19
4       trtX 16.1    4
6       trtX 20.2    6
8       trtX 23.8    8
10      trtX 26.4   22
12      trtX 27.3   31
14      trtX 21.8   19
16      trtX 25.3   26
18      trtX 19.7    6
20      trtX 22.0   17
```

size が 25.0 より大きいのデータ

```
> d[d$size > 25.0,]
```

```
   treatment size seed
10      trtX 26.4   22
12      trtX 27.3   31
16      trtX 25.3   26
19   control 27.9   22
```

seed が 6 以下であるデータ

```
> d[d$seed <= 6,]
3    control 12.0    1
4       trtX 16.1    4
11   control 20.1    3
15   control 18.6    4
...
```

seed が 6 以下，かつ 2 より大

```
> d[d$seed <= 6 & d$seed > 2,]
...
```

seed が 6 より大，または 2 以下

```
> d[d$seed > 6 | d$seed <= 2,]
...
```

1.4 data.frame 内でのならびかえ

```
> d <- d[order(d$size),] # d$size の小さい順に並べかえる
> d <- d[rev(order(d$size)),] # d$size の大きい順に並べかえる
```

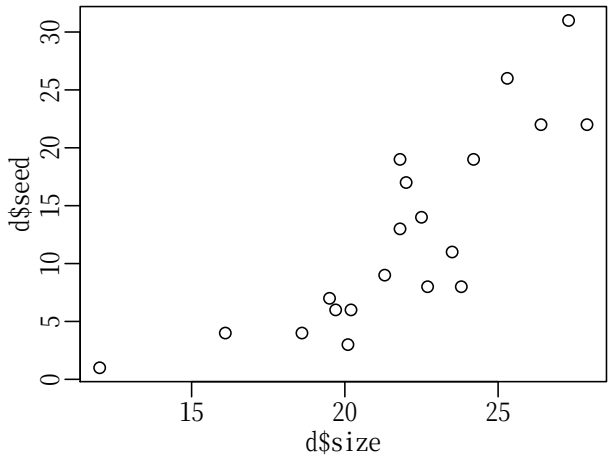
2 R で作図

R 作図の基本 (plot() 関数を使う場合)

- いっぺんに図を作ろうとするのではなく，必要な要素を足していく
- plot() で「わく」を描く
- points(), lines(), legend() で必要なものを追加していく
- par(new = TRUE) による方法は使わないほうがよい (わくを何重にも描くことになったりするから)

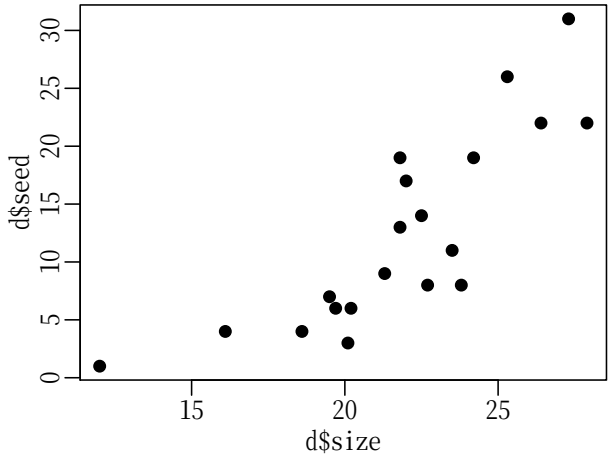
2.1 data.frame のデータを表示する

```
> d <- read.csv("r1.csv")
> plot(d$size, d$seed)
```



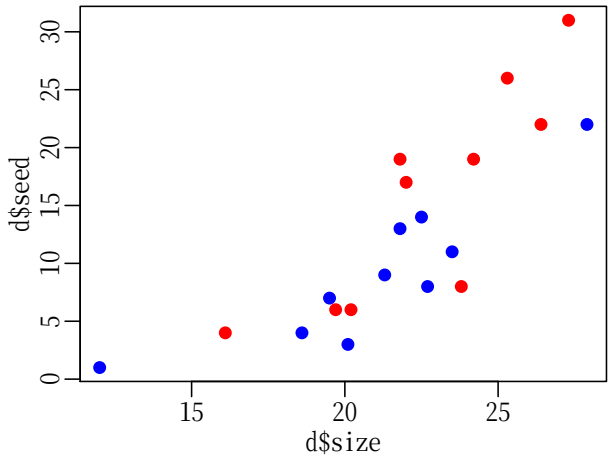
pch 引数で点の種類を変える

```
> plot(d$size, d$seed, pch = 19)
```



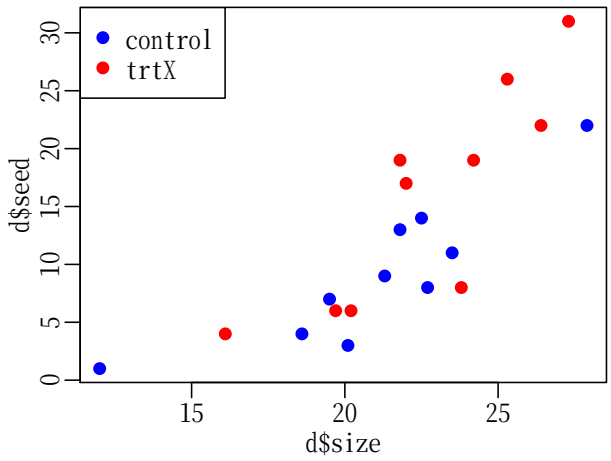
col 引数で点の色を変える

```
> plot(d$size, d$seed, pch = 19, col = c("blue", "red")[d$treatment])
```



legend() 関数で凡例を追加

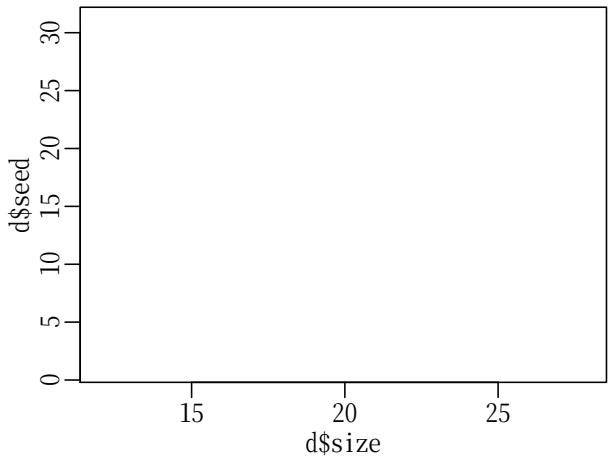
```
# 上の図に legend を追加
> legend("topleft", legend = levels(d$treatment), pch = 19, col = c("blue", "red"))
```



2.2 図を順にかさねていくわざ

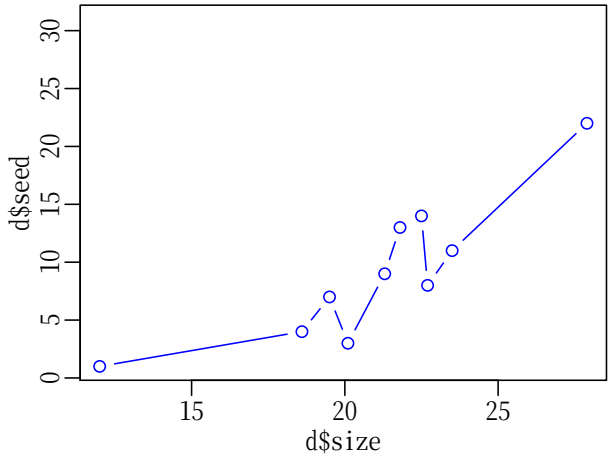
最初にわくだけ描く

```
> plot(d$size, d$seed, type = "n") # わくだけ描く
```



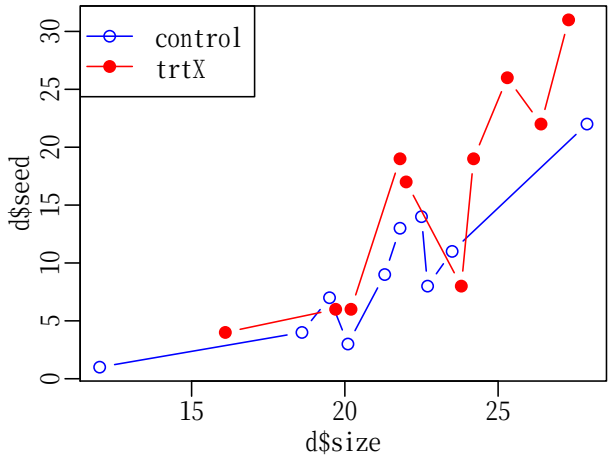
処理が control である線だけ描く

```
> dC <- d[d$treatment == "control",] # treatment が control のデータだけ
> dC <- dC[order(dC$size),] # size 順にならびかえる
> lines(dC$size, dC$seed, pch = 21, col = "blue") # 線を追加
```



次に処理が trtX である線を描き，凡例を追加する

```
> dX <- d[d$treatment == "trtX",] # treatment が trtX のデータだけ
> dX <- dX[order(dX$size),] # size 順にならびかえる
> lines(dX$size, dX$seed, pch = 21, col = "red") # 線を追加
> legend("topleft", legend = levels(d$treatment),
        pch = c(21, 19), col = c("blue", "red"), lwd = 1)
```



3 その他あれこれ

- `pdf()`, `jpg()`, `png()` といった device 指定でいろいろな形式で図を出力できる
- R 作図に慣れてきたら，`library(lattice)` や `library(ggplot2)` で，より「全体像のみやすい」図を作ろう
  - `library(lattice)` を使った条件ごとプロットの例:

```
> d <- d[order(d$size),] # size 順にデータをならびかえ
> print(xyplot(seed ~ size | treatment, data = d, type = "b"))
```