2010-12-01

九州大学・GCOE 統計・データ解析セミナー

— WinBUGS・ベイズ統計モデリング勉強会 —

統計モデルと GLM

久保拓弥 kubo@ees.hokudai.ac.jp

http://goo.gl/bUKrB

このセミナー全体の予定

- 12/1 (水) 午後
  - 統計モデリング，GLM, R
  - MCMC とベイズモデル, WinBUGS
- 12/2 (木) 午前
  - 階層ベイズモデル，WinBUGS
  - WinBUGS まわりの細かいこと
- 12/2 (木) 午後
  - データ解析実演?
- 12/3 (金) 午前
  - データ解析実演のつづき?
  - 個別こんさる?

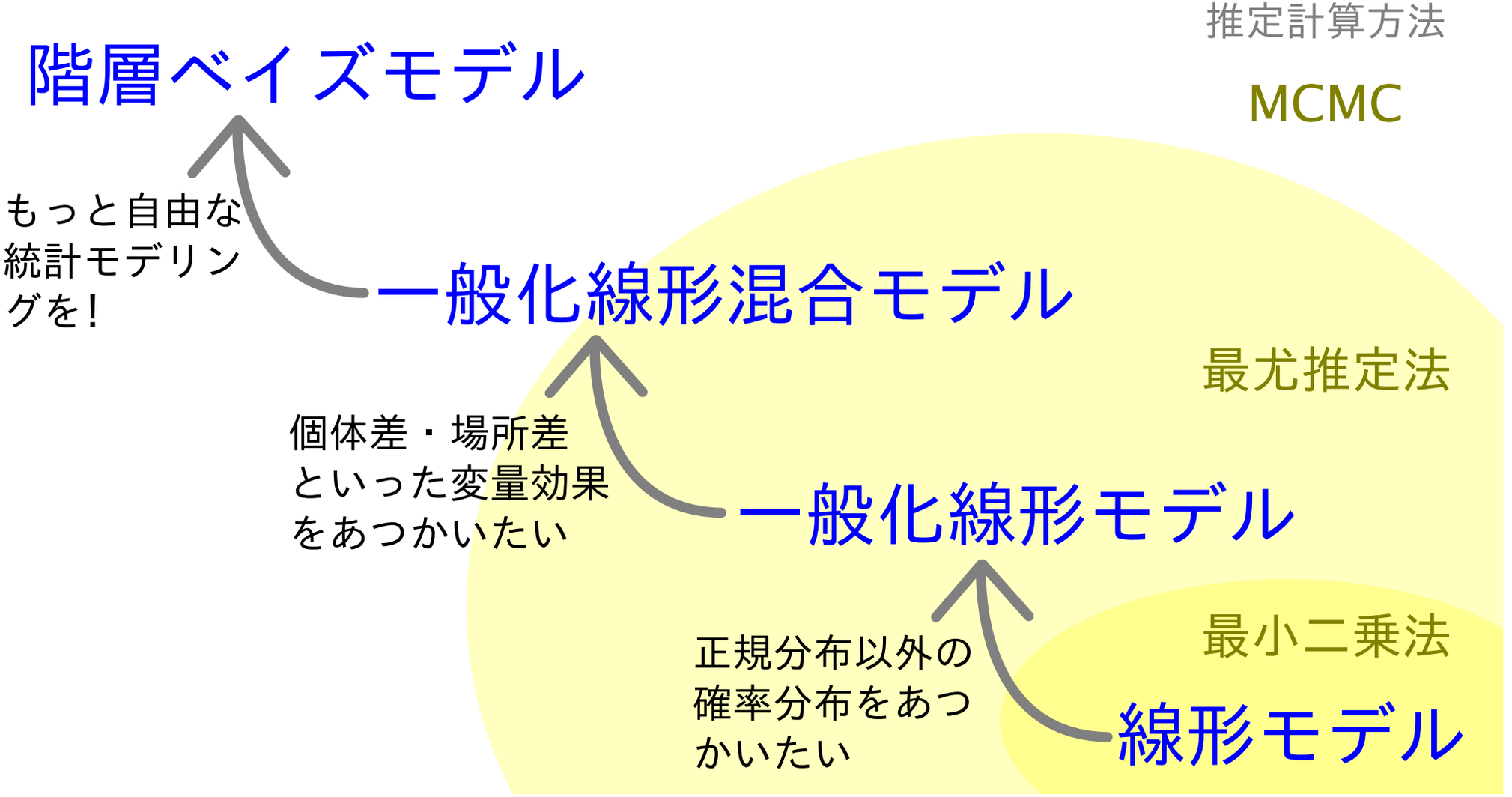
このセミナーの目的

参加者の皆さんが……

- データと統計モデルの部品 (**確率分布**) の対応について考えるようになる
- **線形モデルを拡張**する道すじがわかる
- 「個体差」のような **random effects** がわかる
- ベイズ統計モデルの**事前分布**が何なのか見当がつく
- **BUGS 言語**による統計モデル表現にとりかかれる

今回の説明全体のながれ

線形モデルの発展



この時間のハナシ:

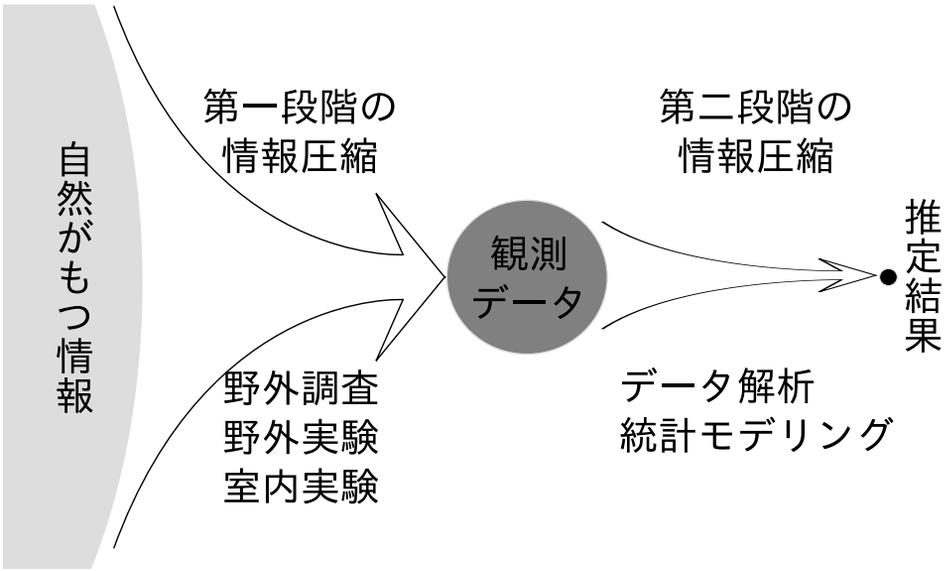
1. 統計モデルや GLM って何なの?
2. この時間の例題 A: ベキ関数 (power function) とポアソン回帰
3. 割算解析やめましょう
4. 「脱」割算の offset 項わざ: ポアソン回帰を強めてみる

1. 統計モデルや GLM って何なの?

自然科学研究における二段階の情報損失

第一段階: 自然現象 → 数値データ

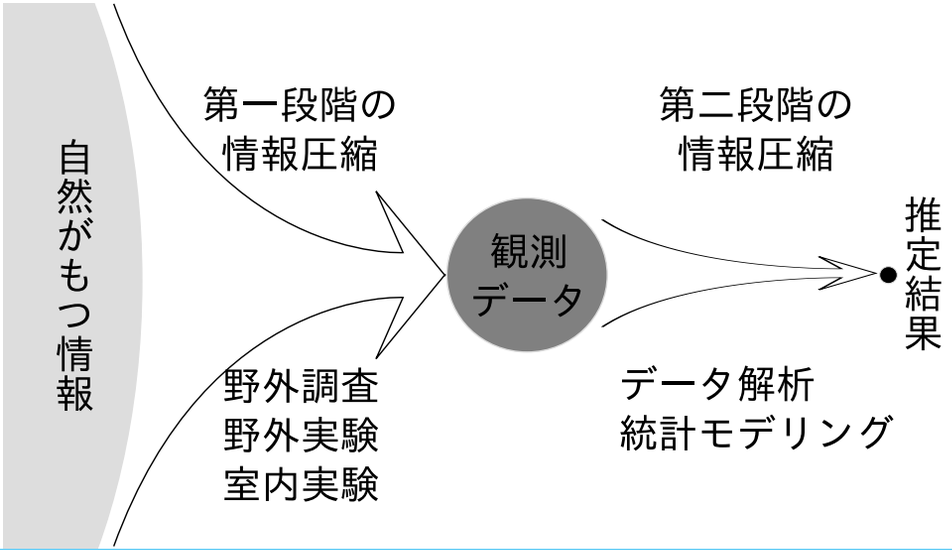
- 観察・実験による情報損失
- 人間が自然現象からとりだせる数値データはごくわずか
- (とくに野外調査では) 厳密に「同じ」データを再びとれない



自然科学研究における二段階の情報損失

第二段階: 数値データ → 統計学的な解析結果

- 統計解析による情報損失
- 人間のアタマは大量の数値データも把握できない
- この情報損失過程には再現性がある(“客観的” に検討できる)



自然科学の研究をするためには，この過程のしくみを理解する必要がある

Ecologists' data-cooking (until recently)

http://www.wstephens.net/

nice data!

Ecologists' data-cooking (until recently)

nice data!

http://www.wstephens.net/

data-chopping

http://alispagnola.blogspot.com/

Ecologists' data-cooking (until recently)

nice data!

http://www.wstephens.net/

data-chopping

http://alispagnola.blogspot.com/

http://chan4chan.com/

data-flaming

http://howdoifeelaboutthis.tumblr.com/

研究は内輪で腐る Academic inbreeding depression

統計モデリングとは何か?

データ解析とは統計モデリングのことだ

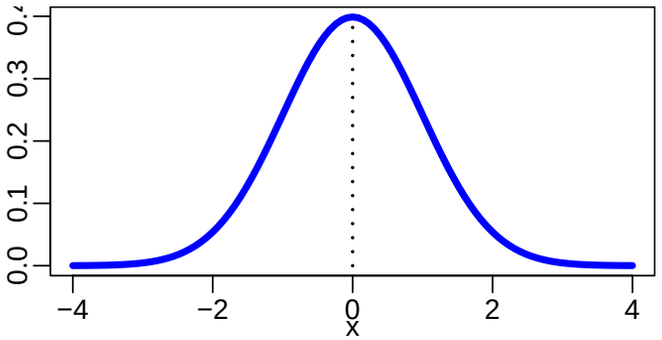
- 統計モデルは (解析したい) 観測データと対象に関する先験的な知識・情報にもとづいて構築される
- 統計モデルは観測データのパターンをうまく**説明**できるようなモデル
- 統計モデルの基本的な部品は**確率分布**，確率分布のカタチはパラメーターによって決まる
- 観測データをうまく説明できるようにパラメーターの値を決めることを「統計モデルの**あてはめ**」または「統計モデルによる**推定**」という

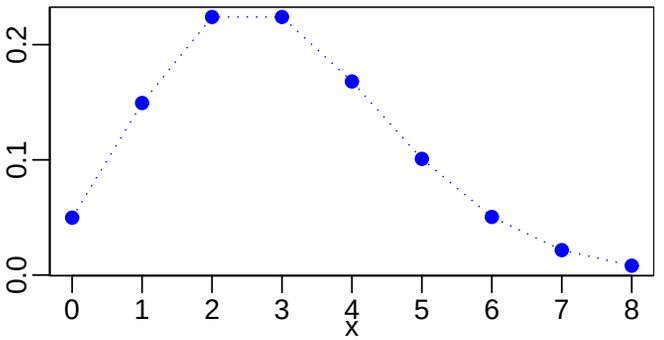
自然科学でははばらつきのある自然現象を……

背後にある確率論的モデルによって生成された，と仮定する

直接は見えない世界

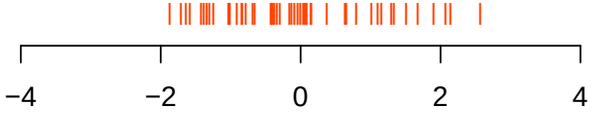
- モデル
- 確率分布
- 母集団

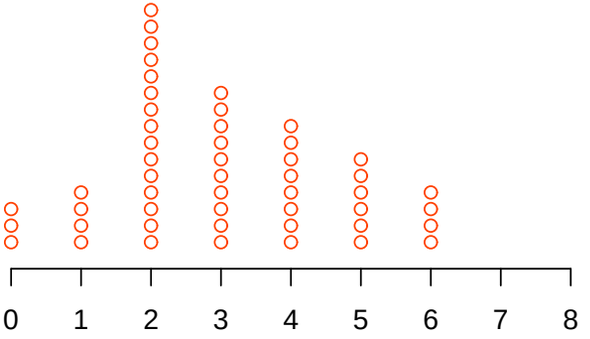




サンプリング ⇓⇑ (パラメーター) 推定

- データ
- 乱数
- 標本集団





見ることのできる世界

「結果 ← 原因」関係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):

$$\begin{aligned}(\text{応答変数の平均}) = \quad & \text{定数 (切片)} \\ & + \quad (\text{係数 } 1) \times (\text{説明変数 } 1) \\ & + \quad (\text{係数 } 2) \times (\text{説明変数 } 2) \\ & + \quad (\text{係数 } 3) \times (\text{説明変数 } 3) \\ & + \quad \cdots\end{aligned}$$

統計モデリング: 観測データのモデル化

- 統計モデルは観測データのパターンをうまく**説明**できるようなモデル
- 基本的部品: **確率分布** (とそのパラメーター)
- データにもとづくパラメーター推定，**あてはまりの良さ**を定量的に評価できる

今回は，「結果 ← 原因」関係を一番単純に表現している**線形モデル**のみを検討する

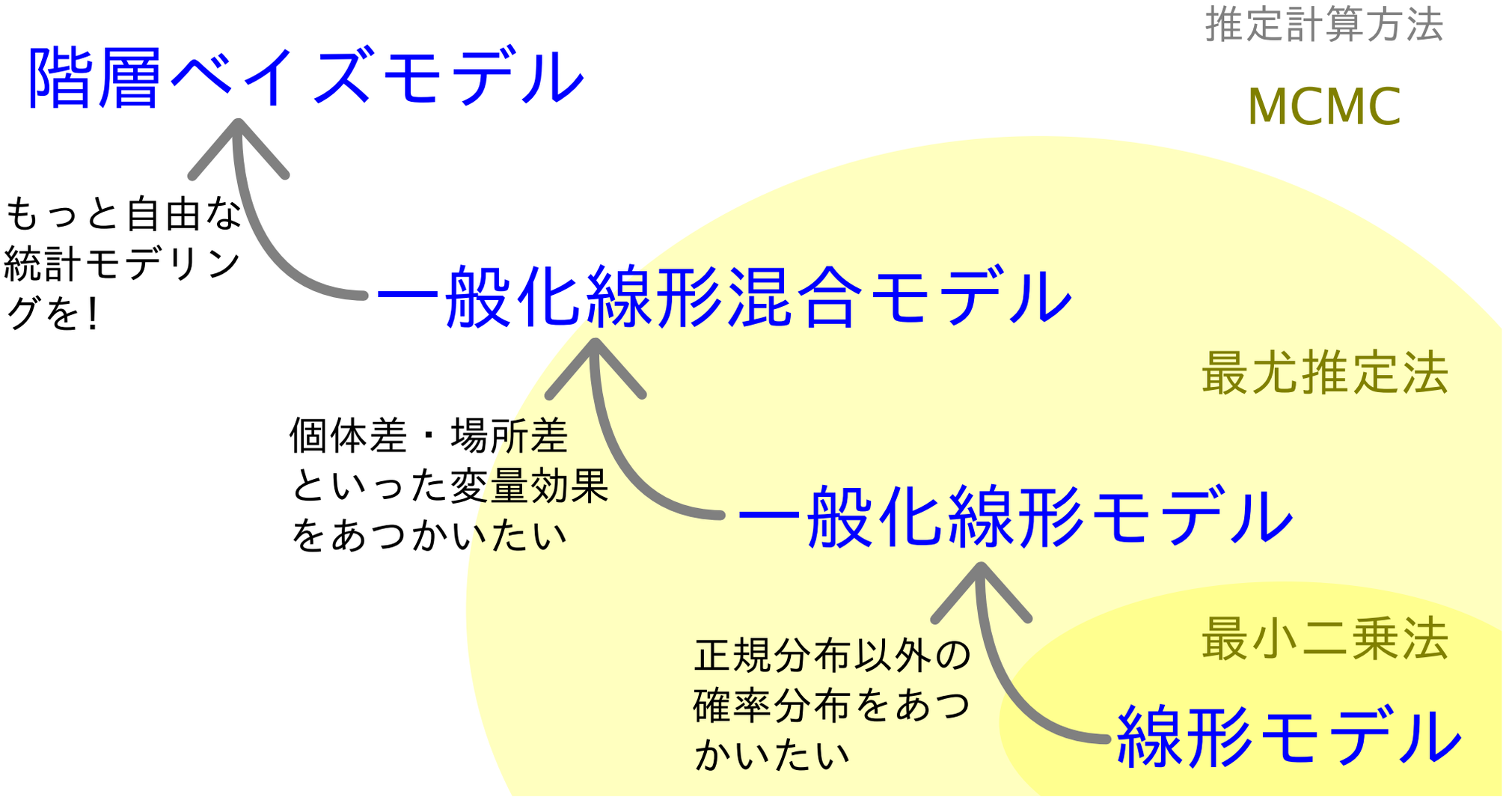
「結果 ← 原因」関係を表現する線形モデル

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

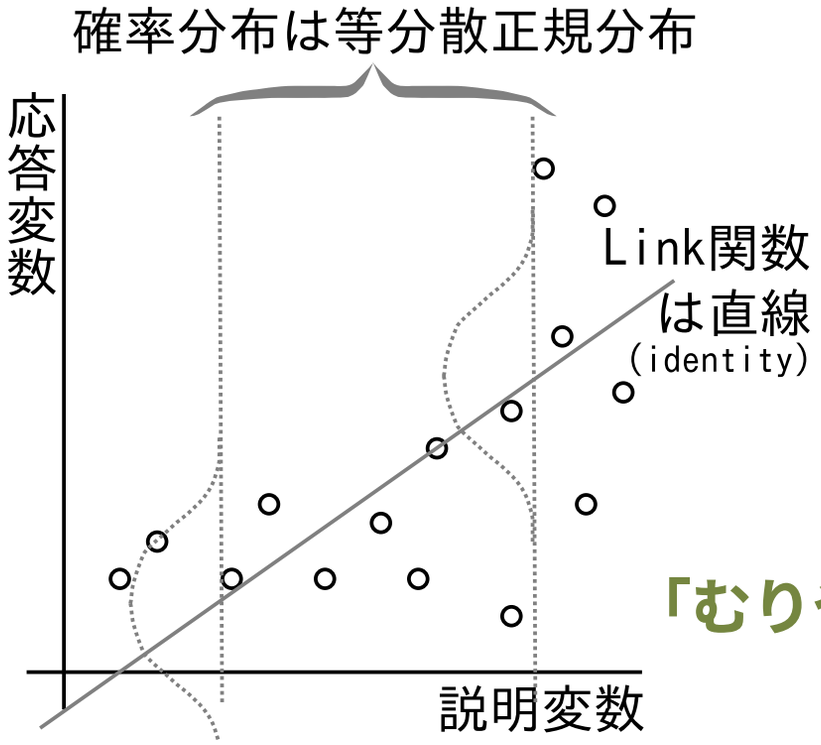
$$\begin{aligned}\text{(応答変数の平均)} = \quad & \text{定数 (切片)} \\ & + \quad \text{(係数 1)} \times \text{(説明変数 1)} \\ & + \quad \text{(係数 2)} \times \text{(説明変数 2)} \\ & + \quad \text{(係数 3)} \times \text{(説明変数 3)} \\ & + \quad \cdots\end{aligned}$$

(交互作用項については粕谷さんが説明してくれます)

線形モデルの発展


推定計算方法
階層ベイズモデル
MCMC
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

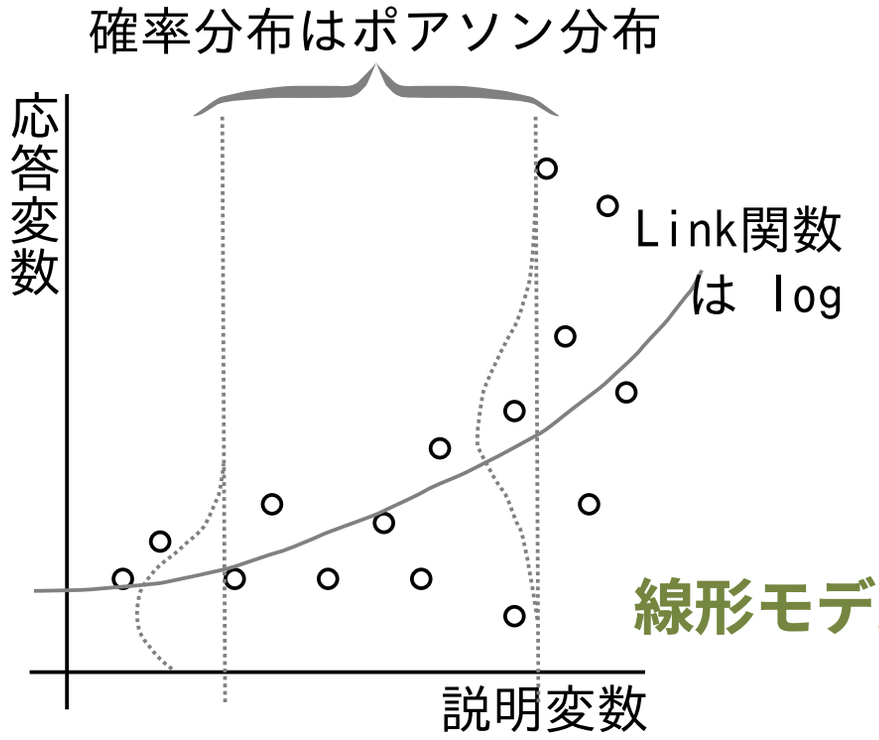
統計モデル: いつでも「直線回帰」でいいのか?



「むりやり線形モデル」の限界!

- もしこの観測データ (縦軸) がカウントデータだったら?
- まずい点: 等分散ではないに直線回帰?
- まずい点: モデルによる予測は「負の個体密度」?

カウントデータならポアソン回帰で!



線形モデル → GLM への拡張

- ポアソン回帰は一般化線形モデルの一部
- 平均値とともに増大する分散に対応
- モデルによる予測はつねに非負

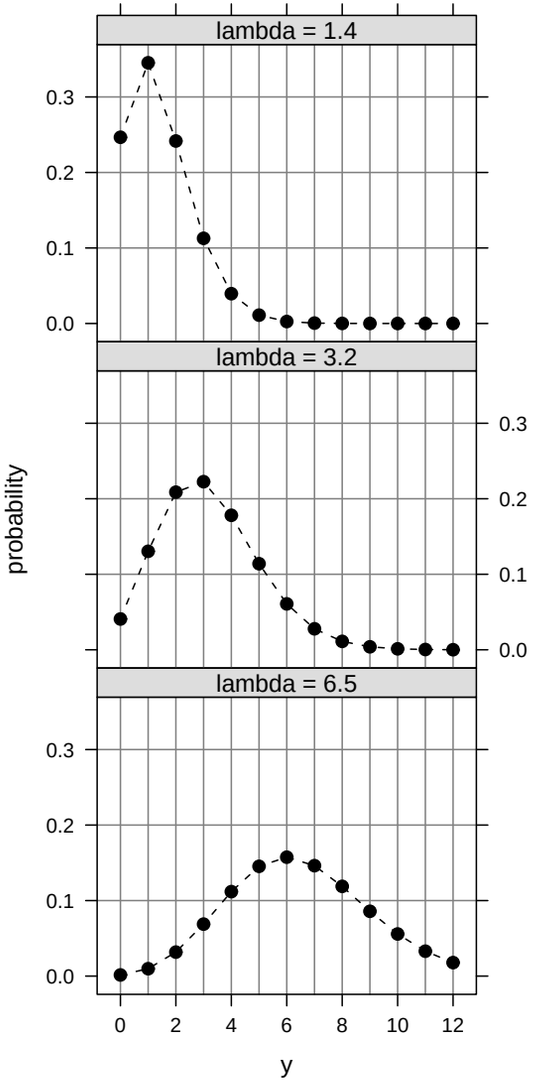
ポアソン分布 (Poisson distribution) とは何か?

- 離散分布 $y_i \in \{0, 1, 2, \cdots, \infty\}$
- 確率密度関数 (paramter: λ)

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

- 期待値 λ，分散 λ

- 上限を設定できないカウントデータに
- 例: 産卵数・種子数・個体数……



一般化線形モデル (generalized linear model; GLM)

確率分布・link 関数・線形予測子を指定して特定できる統計モデル

- 確率分布: 応答変数のばらつきとして正規分布，ポアソン分布，二項分布その他を指定できる
- link 関数を $f()$ とすると，確率分布の平均値 $= f$(線形予測子) という関係がある
- 線形予測子: $\beta_0 + \beta_1 x_1 + \beta_2 + \cdots$，ただし x_i は 説明変数 で β_i は x_i の係数 (coefficient)
  - 観測データ ($\{x_i\}$ と $\{y_i\}$) にもとづいて $\{\beta_i\}$ を最尤推定するのが，GLM によるパラメーター推定

この時間も統計ソフトウェア R 利用前提のハナシで

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

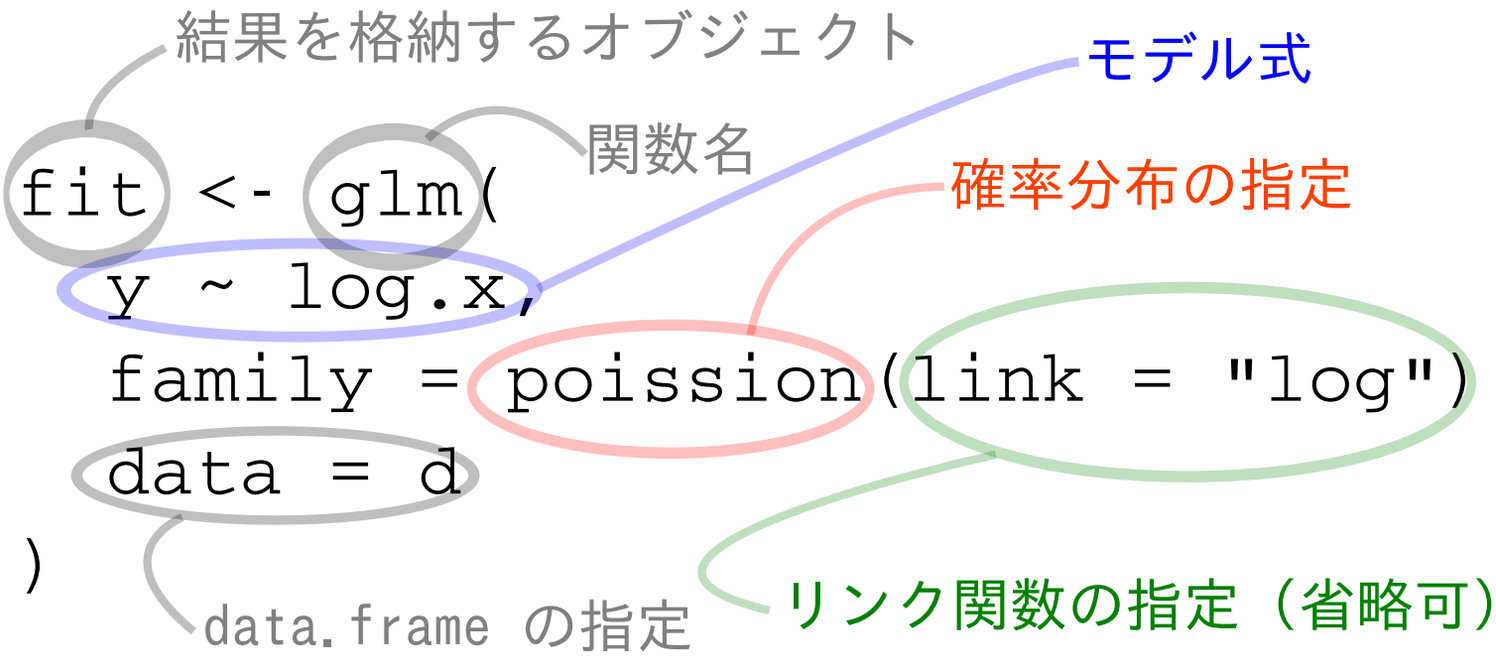
- いろいろな OS で使える free software
- 使いたい機能が充実している
- 作図機能も強力
- S 言語によるプログラミング可能
- よい教科書が出版されつつある
  - すでに多すぎて列挙できません
  - ネット上のあちこち

R で一般化線形モデル: glm() 関数

	確率分布	乱数生成	パラメーター推定
(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
- この時間はポアソン回帰 を使った GLM だけ紹介します

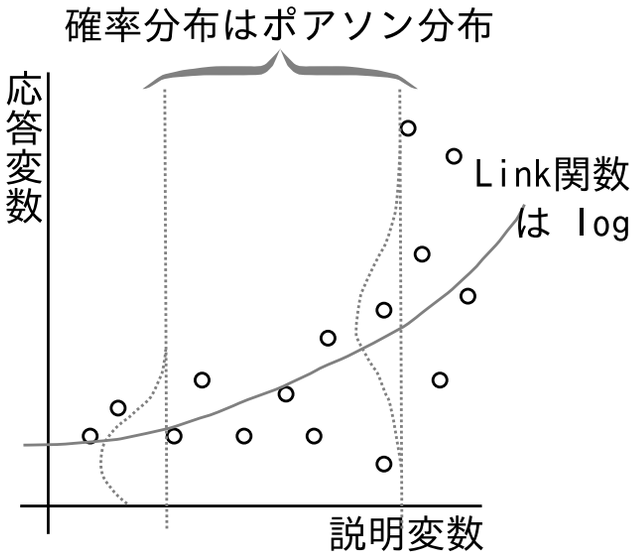
R の glm() 関数: 何を指定すればいい?



- **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?**

ポアソン回帰の glm() 指定 (1)

- family: poisson, ポアソン分布
  - カウントデータ $(0, 1, 2, \cdots$ と数えられるデータ) の場合はポアソン分布で説明してみる
- link 関数: "log"
  - これは family = poisson 時の「おススめ」 link 関数

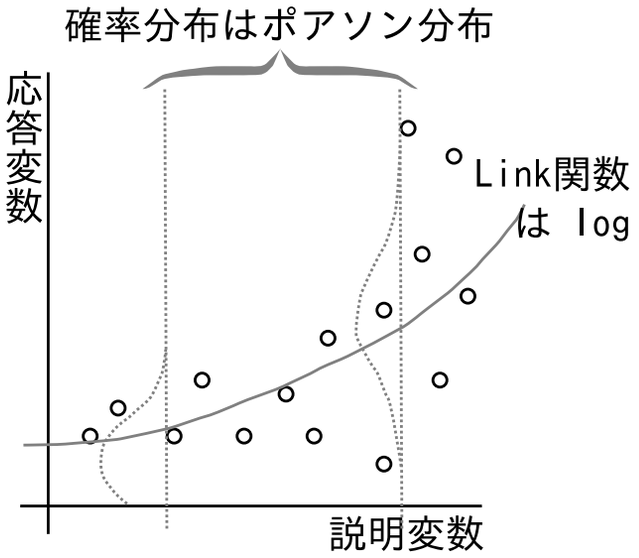


- モデル式 (線形予測子 z): たとえば y ~ x と指定したとする

family = poisson(link = "log") 指定とは何をやっているのだろうか?

ポアソン回帰の glm() 指定 (2)

- family: poisson, ポアソン分布
- link 関数: "log"
- モデル式 (線形予測子 z): たとえば
  y ~ x と指定したとする



- 線形予測子 $z = a + bx$

  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 $\log(\lambda) = z$

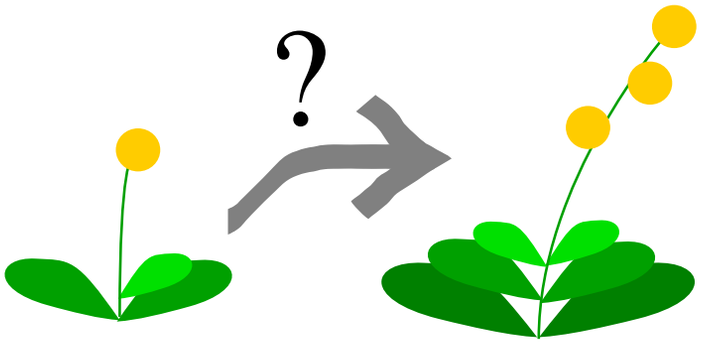
  つまり $\lambda = \exp(z) = \exp(a + bx)$
- 応答変数 は平均 λ のポアソン分布に従う: $y \sim \mathrm{Pois}(\lambda)$

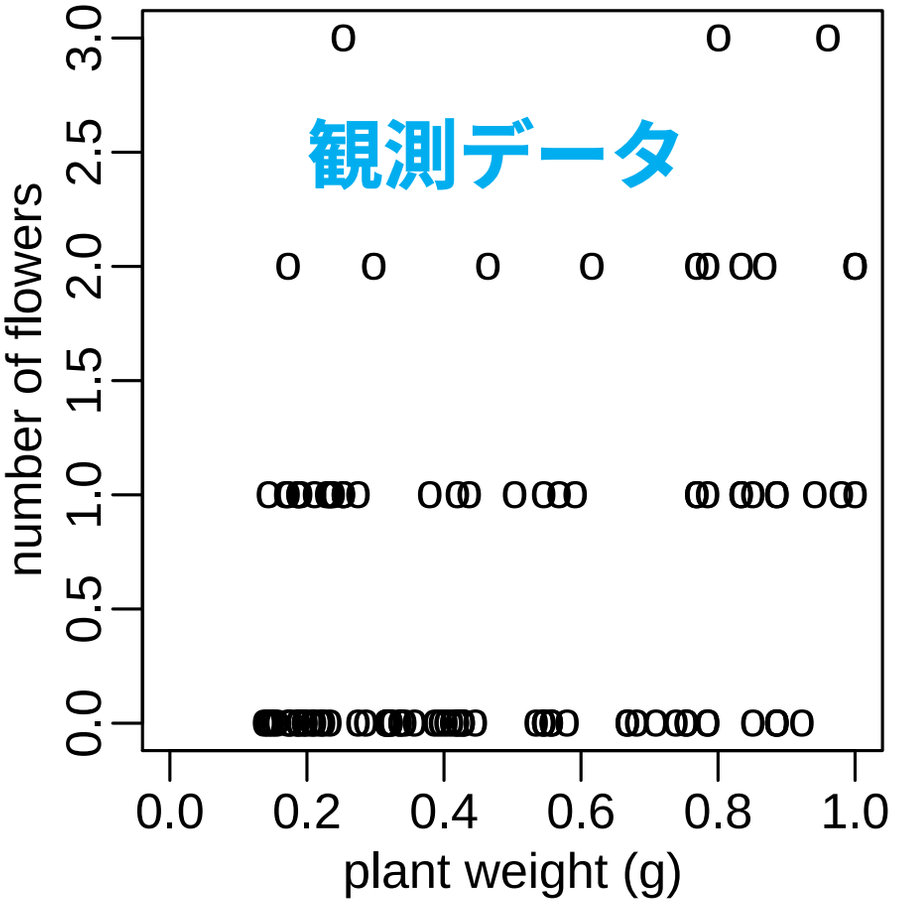
2. GLM の例題 A:

ベキ関数 (power function) とポアソン回帰

この時間の例題 A: サイズと花数の関係?

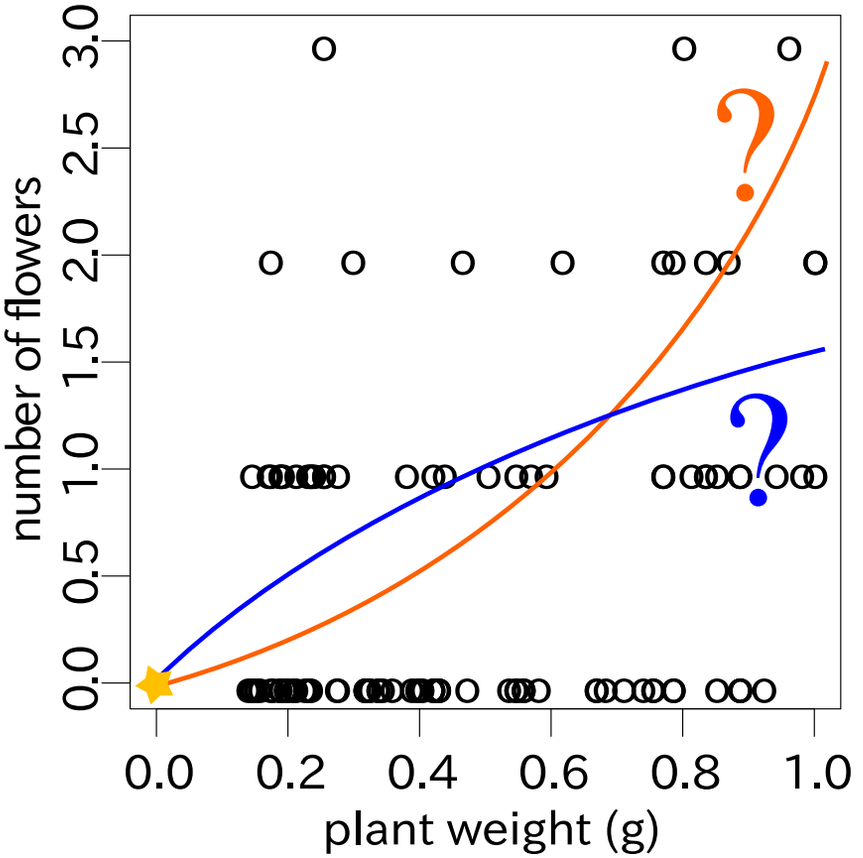
地上部の重量 x が増加するにつれて花数 y は増加するだろうか?





- **調べた個体数は 100 個体: $i = 1, 2, \cdots, 100$**
- **説明変数は地上部の重量 x_i**
- **応答変数は花数 y_i**

統計モデリング: x と y の関係は?



- とりあえず「サイズ x とともに増大」と仮定し ……
- さらに原点 $(0, 0)$ はとおる，と仮定しよう ……
- **とくに知りたいこと**: 関数型は急上昇? アタマうち?

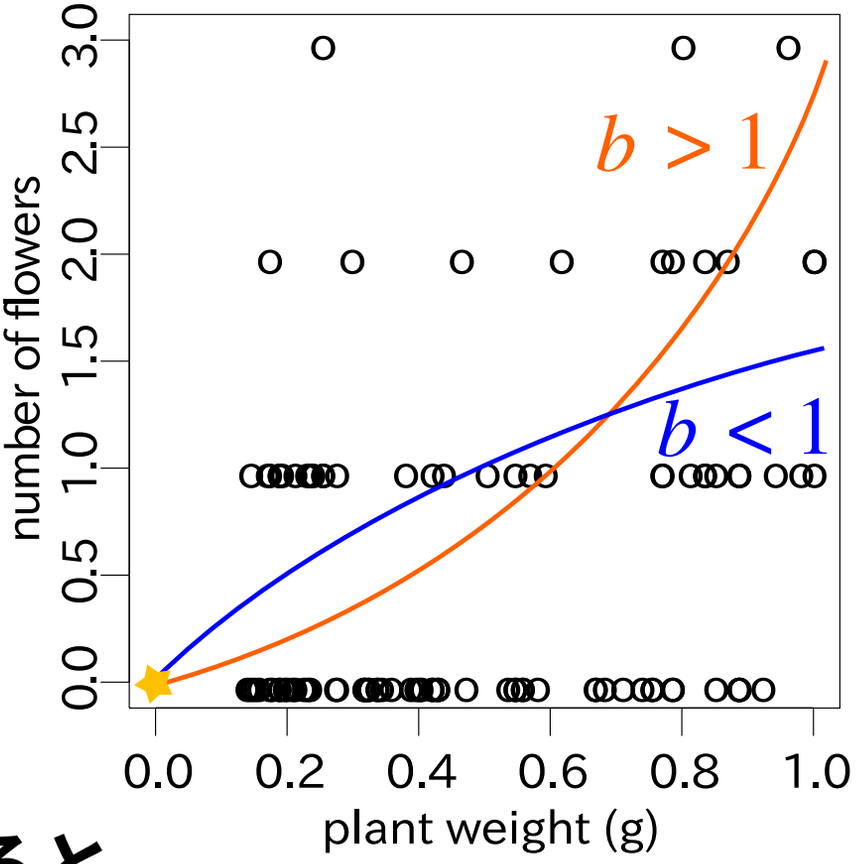
“アロメトリック” なモデルが良さそう

1. 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. ポアソン分布の平均 λ_i は x_i のベキ関数であると仮定:

$$\lambda_i = A x_i^b$$



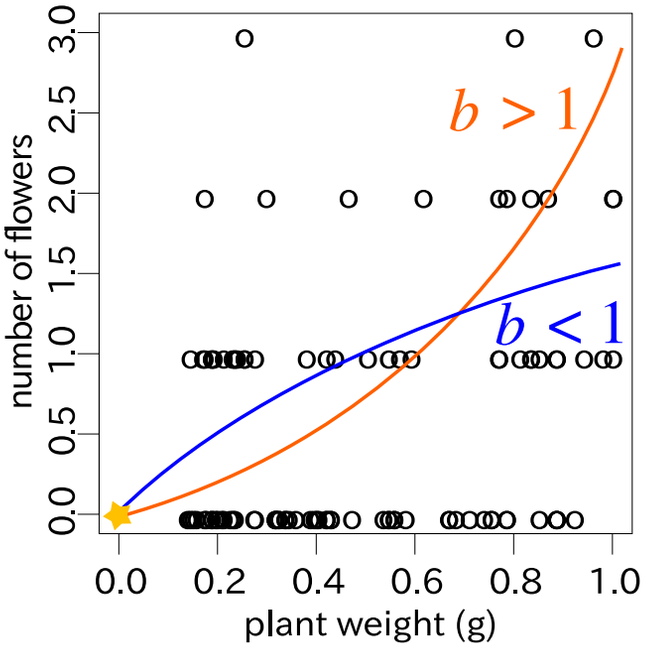
$\lambda_i = A x_i^b$ を変形してみると

$$\lambda_i = \exp(\log(A) + b \times \log(x_i))$$

$a = \log(A)$ とすると，$\log(\lambda_i) = a + b \times \log(x_i)$

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 ~ log.x` と指定，ただし重量 `x` の対数を `log.x` する



- **線形予測子** $z = a + b$ `log.x`

  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 $\lambda = \exp(z) = \exp(a + b$ `log.x`$)$
- **応答変数** は平均 λ のポアソン分布に従う: $y \sim \mathrm{Pois}(\lambda)$

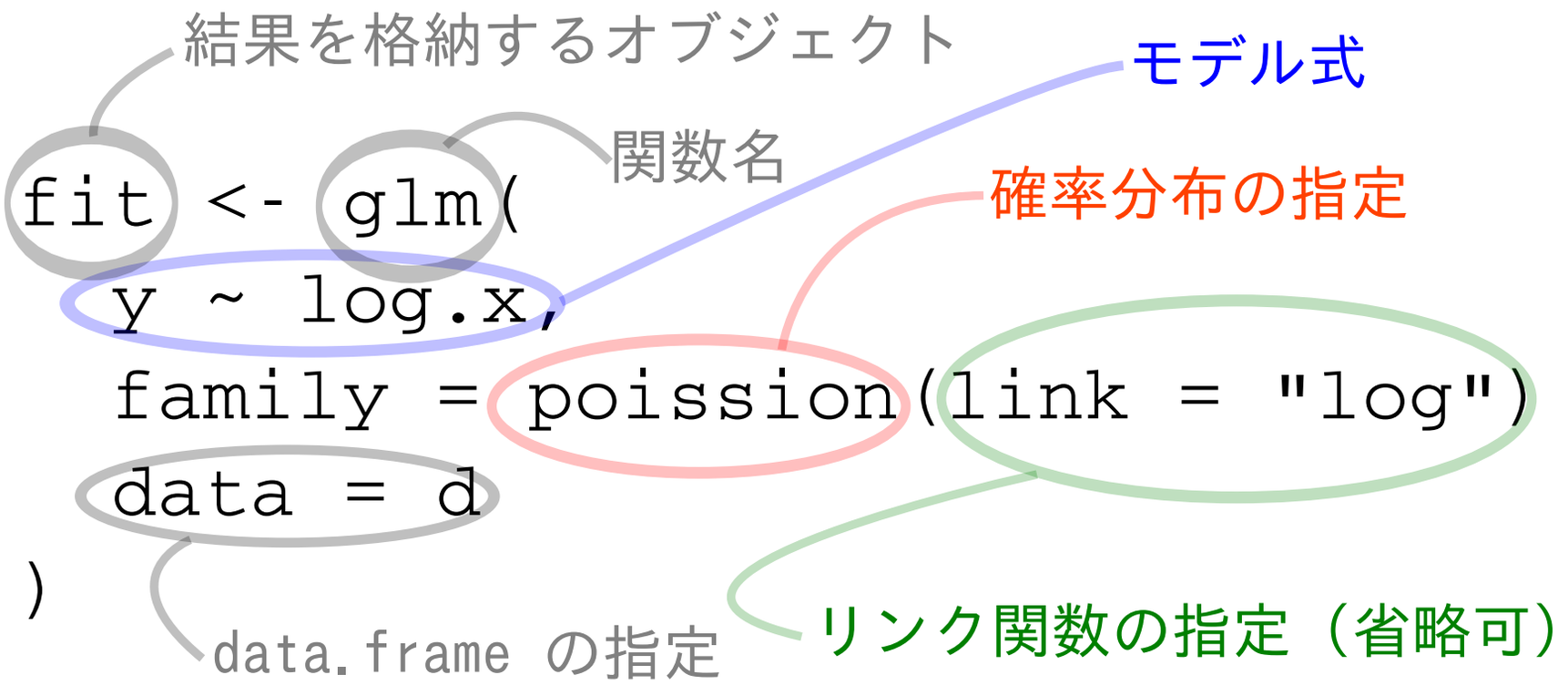
R に格納されたデータセットを操作する

編集前の data.frame "d" $\Longrightarrow$ log.x **列を追加する**

```
> load("d.RData")
> head(d) # 先頭 6 行の表示
        x y
1 0.66762 0
2 0.85077 0
3 0.68124 0
4 0.14379 1
5 0.25316 1
6 0.88585 0
```

```
> d$log.x <- log(d$x)
> head(d)
        x y    log.x
1 0.66762 0 -0.40404
2 0.85077 0 -0.16162
3 0.68124 0 -0.38384
4 0.14379 1 -1.93939
5 0.25316 1 -1.37374
6 0.88585 0 -0.12121
```

glm() 関数の指定


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
fit <- glm(
y ~ log.x,
family = poission(link = "log")
data = d
)
data.frame の指定
リンク関数の指定（省略可）

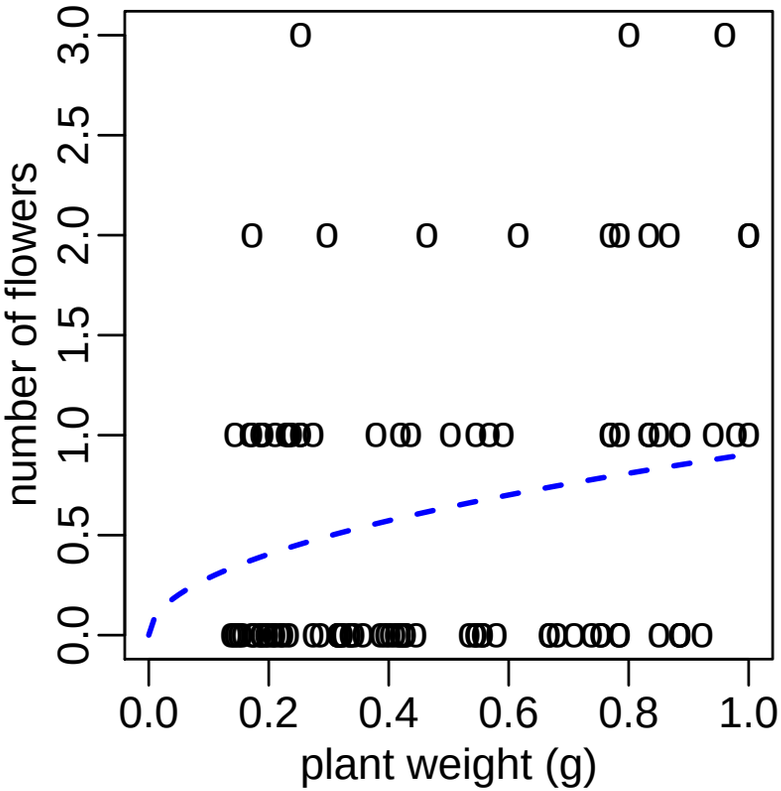
R の glm() 関数による推定結果

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

Call:
glm(formula = y ~ log.x, family = poisson, data = d)
(... 略...)

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   -0.115      0.204   -0.56    0.573
log.x          0.476      0.222    2.14    0.032
(... 略...)
```

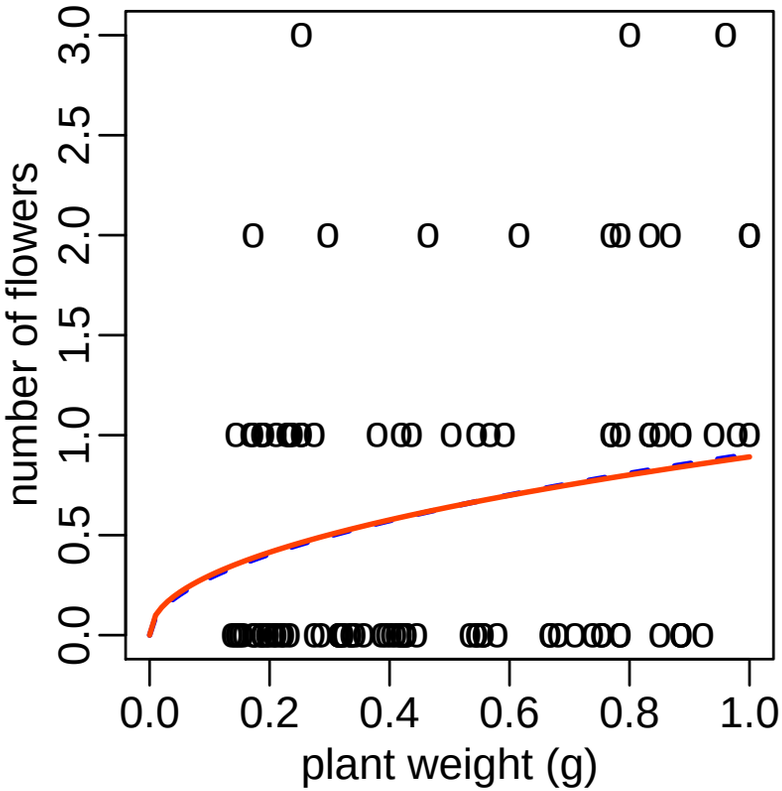
Coefficients は説明変数の係数という意味

GLM の推定結果を図示してみる

「ホント」の
重量 $\rightsquigarrow$ 平均花数



推定された
重量 $\rightsquigarrow$ 平均花数



ここまでのまとめ

1. 統計モデル: 確率分布を使って観測データに見られるパターンを説明
2. GLM の部品: 確率分布, link 関数, 線形予測子
3. データをよくみて統計モデルの確率分布を選び，R の `glm()` を使いこなそう

すみませんが，deviance や尤度のハナシはまた別の機会に……

3. 割算解析やめましょう

ワリ算な統計学について……

世間でよくみかける**おススめできない**作法の例

- ある調査地 i で N_i 本の樹木のうち k_i 本で開花していた
- 調査地 i の開花確率を $p_i = k_i/N_i$ とした
- 別の調査地 j の開花確率を $p_j = k_j/N_j$ とした
- 調査地 i と j の間で開花確率が異なるかどうか，p_* が正規分布にしたがうと仮定して「ゆーい差を検定」した
- 確率は正規分布ではない，と指摘されたのでノンパラメトリック検定で「ゆーい差を検定」した

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，避けにくいわりざん

- **避けられる割算値**
  - 密度などの指数

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

    対策: offset 項わざ
  - 確率

    例: N 個のうち k 個にある事象が発生する確率

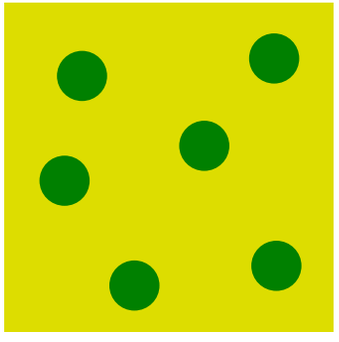
    対策: ロジスティック回帰など二項分布モデルで
- **避けにくい割算値**
  - 測定機器が内部で割算した値を出力する場合
  - 割算値で作図せざるをえない場合があるかも

4. 「脱」割算の offset 項わざ

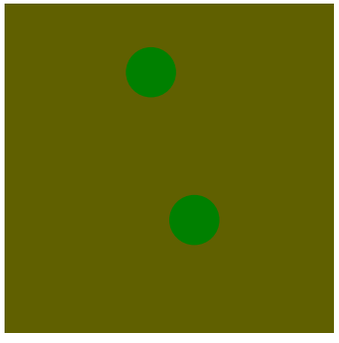
ポアソン回帰を強めてみる

この時間の例題 B: 調査区画内の個体数は明るさで変わるか?

- 何か架空の植物個体の数が「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



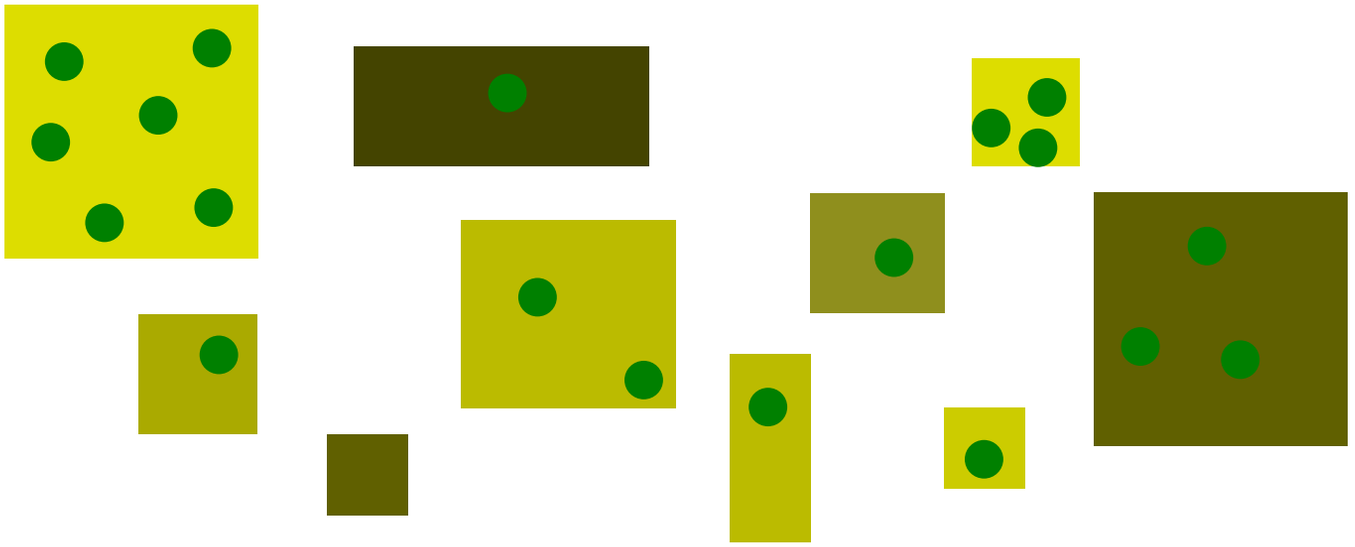
x 大
明るい



x 小
暗い

これだけなら単純に `glm(..., family = poisson)` 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」?!!



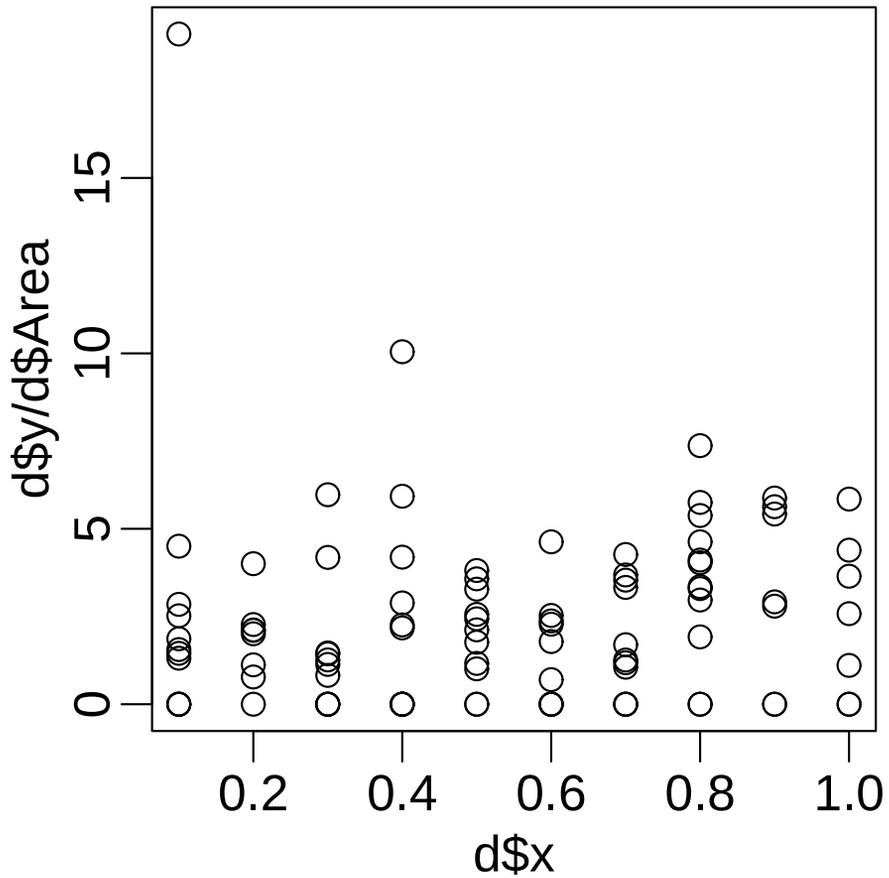
- 明るさ x と面積 A を同時に考慮する必要あり
- ただし「密度 $=$ 個体数 / 面積」といった割算値解析はやらない!
- `glm()` の `offset` 項わざでうまく対処できる
- ともあれその前に観測データを図にしてみる

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
```

ありがちな明るさ vs 割算値の図

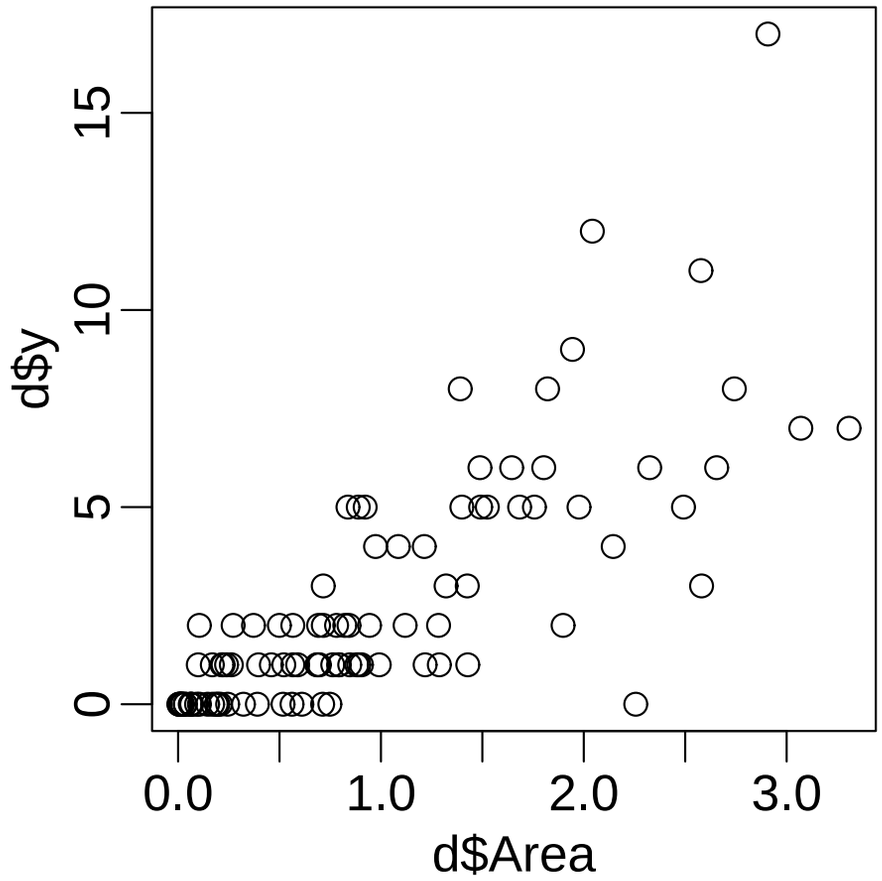
```
plot(d$x, d$y / d$Area)
```



- **いまいちよくわからない……?**

割算値ヤメて面積 A vs 個体数 y の図

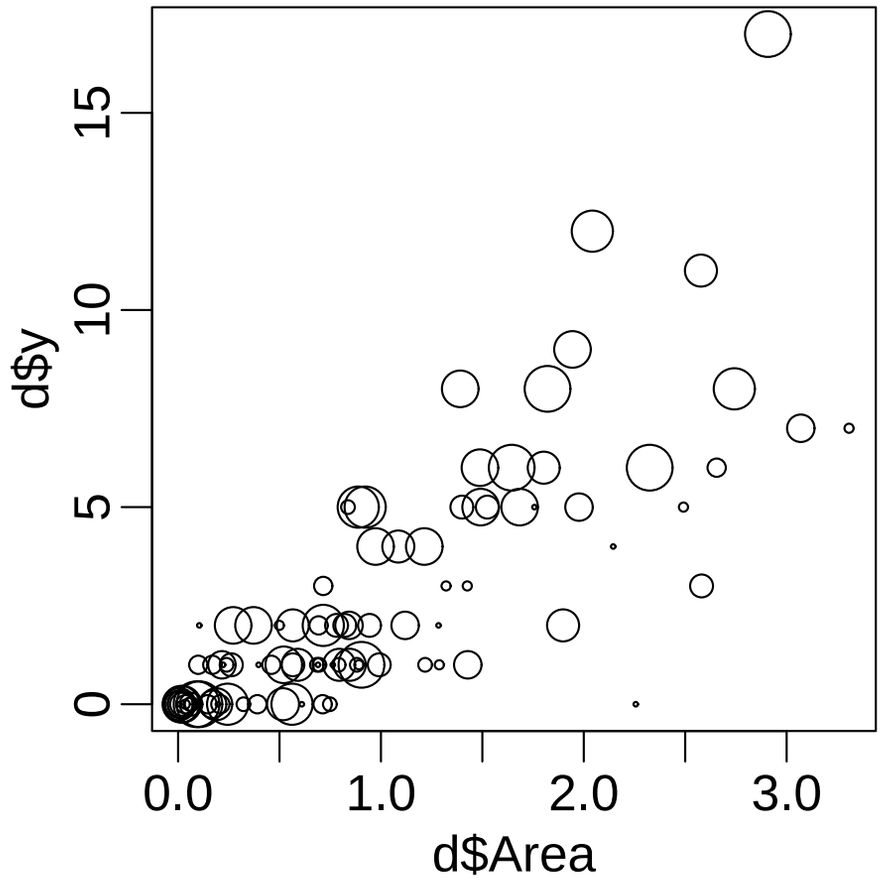
```
plot(d$Area, d$y)
```



- 面積 A とともに区画内の個体数 y が増大するようだ

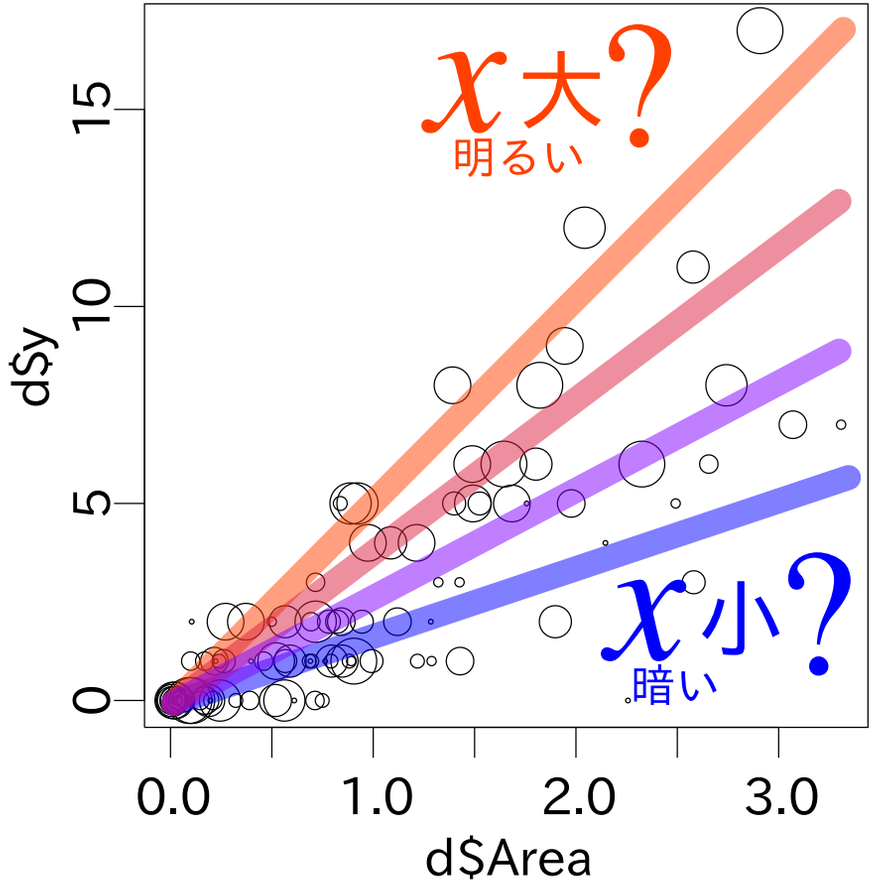
明るさ x の情報 (マルの大きさ) も図に追加

```
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
- ……といった統計モデルを作りたい!

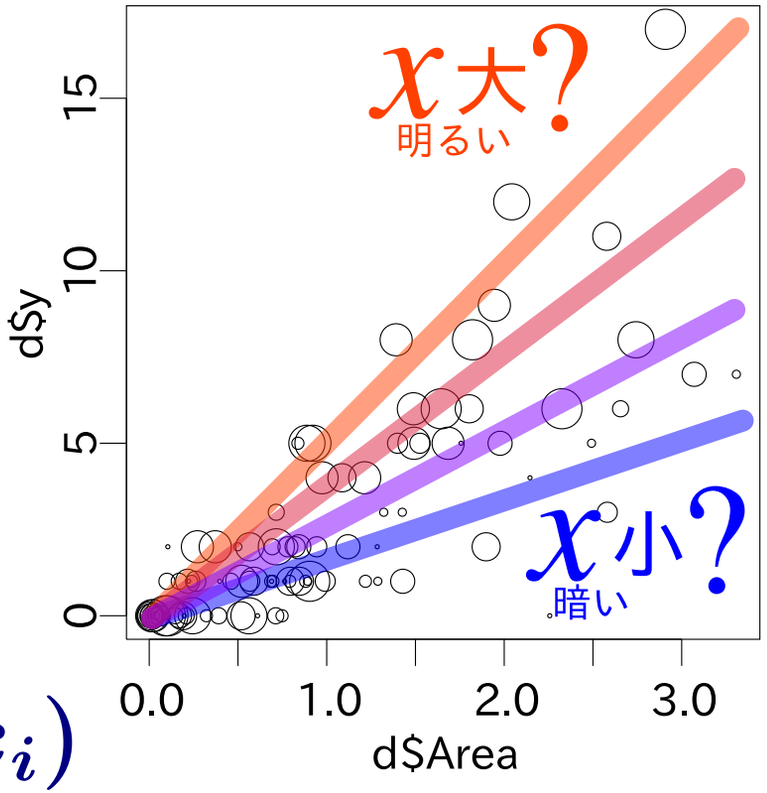
「平均個体数 = 面積 × 密度」モデル

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

$$\lambda_i = A_i \exp(a + b x_i)$$



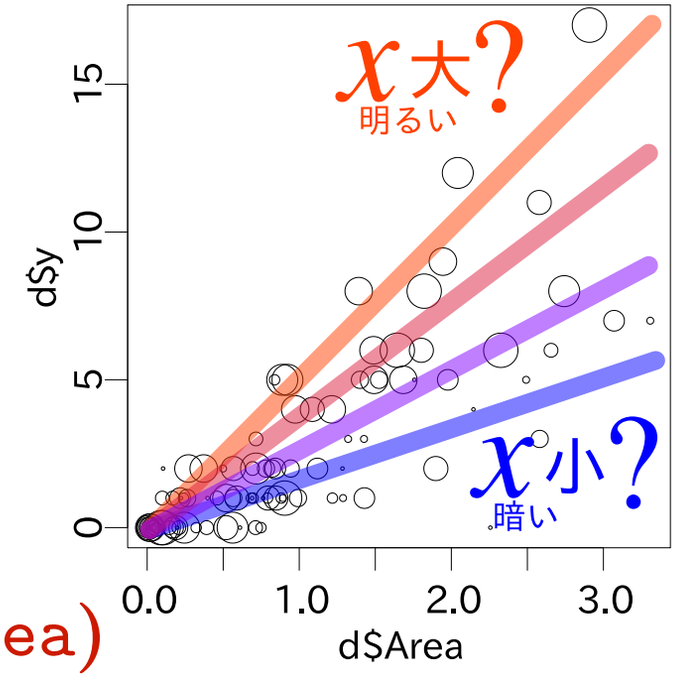
$$\lambda_i = \exp(a + b x_i + \log(A_i))$$

$$\log(\lambda_i) = a + b x_i + \log(A_i)$$

$\log(A_i)$ を offset 項とよぶ

この問題は GLM であつかえる! offset 項ワザ

- 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)

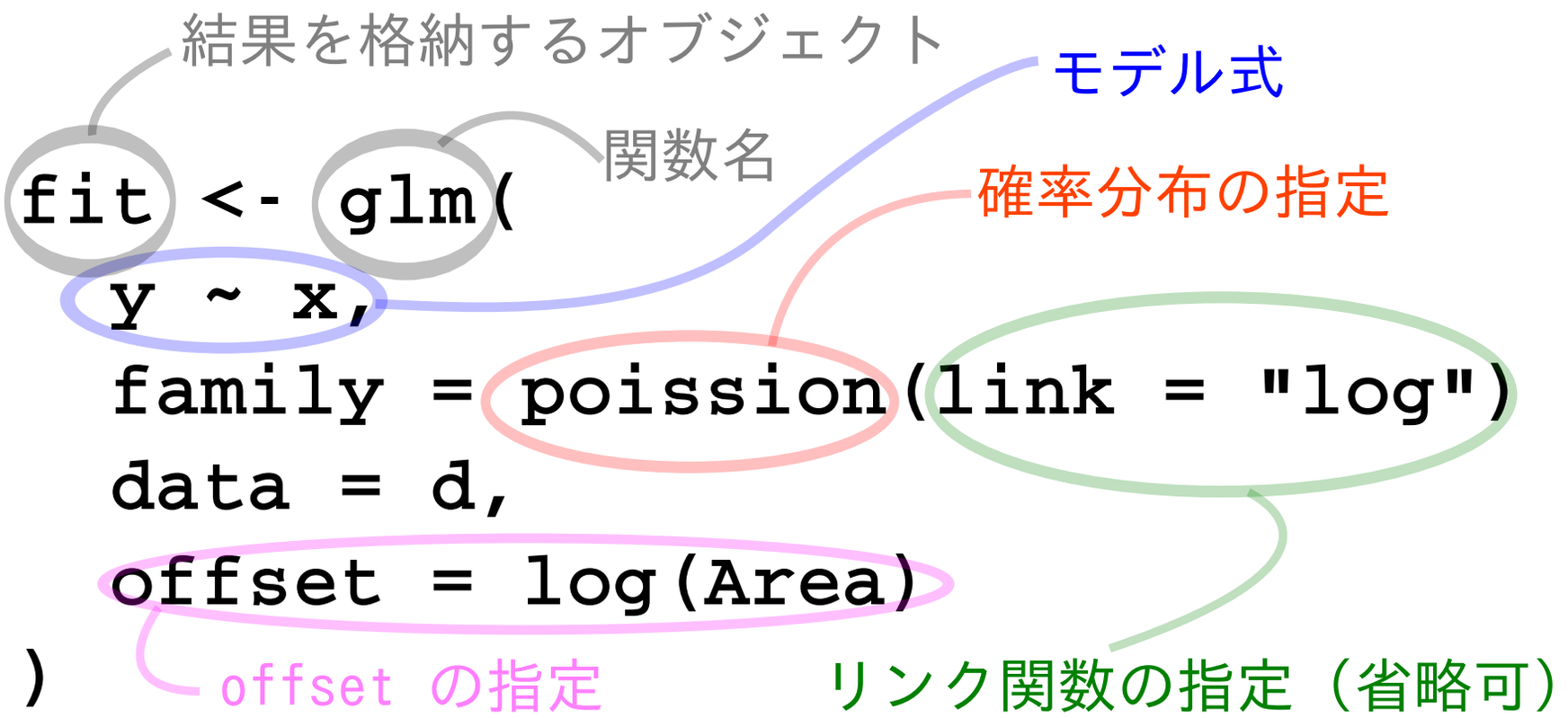


- **線形予測子** $z = a + b\,\mathrm{x} +$ log(Area)

  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 $\lambda = \exp(z) = \exp(a + b\,\mathrm{x} +$ log(Area)$)$
- **応答変数** は平均 λ のポアソン分布に従う: $y \sim \mathrm{Pois}(\lambda)$

glm() 関数の指定

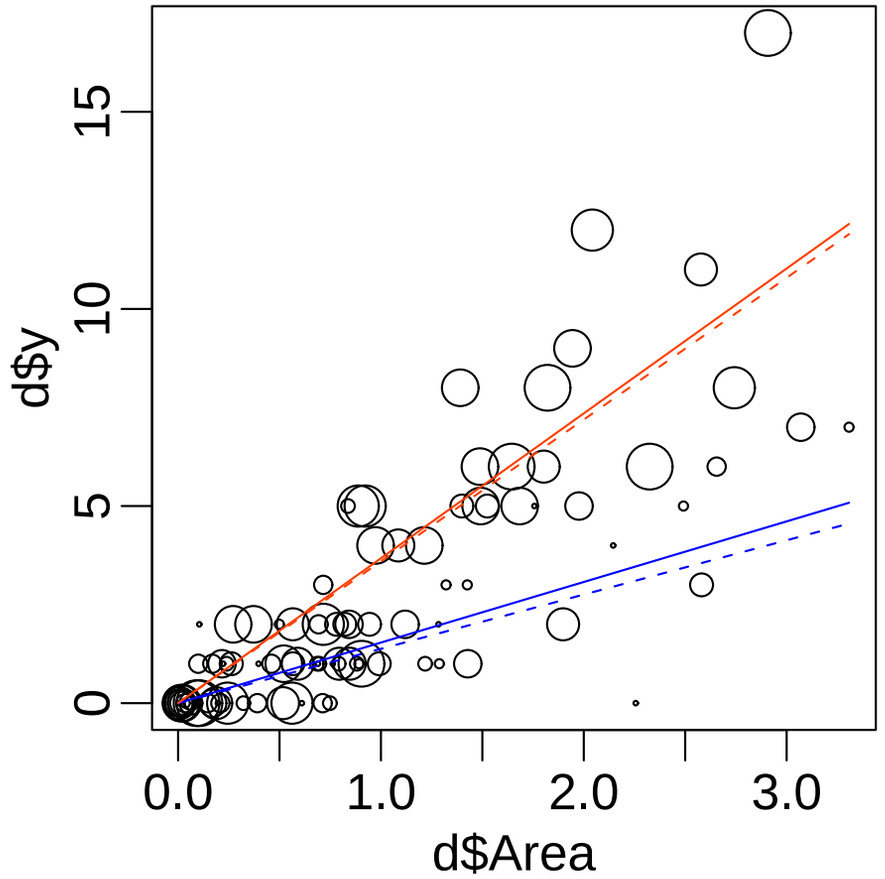

結果を格納するオブジェクト
関数名
モデル式
確率分布の指定
fit <- glm(
y ~ x,
family = poission(link = "log")
data = d,
offset = log(Area)
)
offset の指定
リンク関数の指定（省略可）

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
```

Coefficients は説明変数の係数という意味

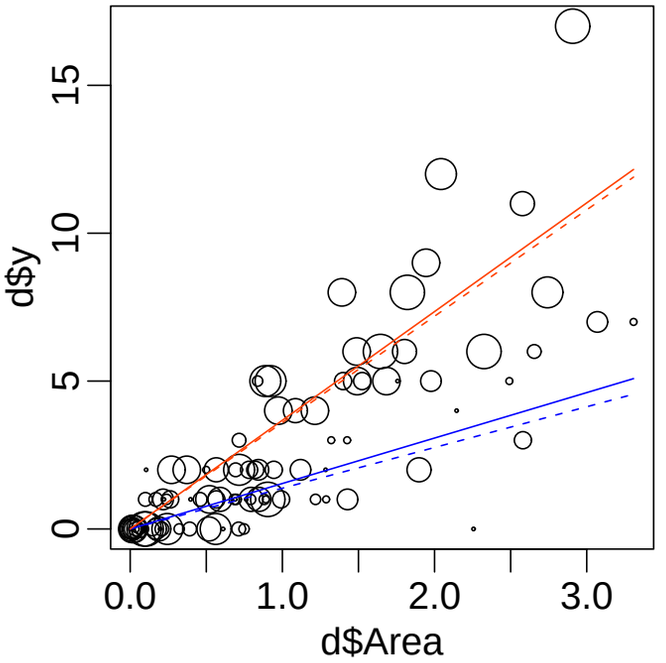
推定結果を図にしてみる

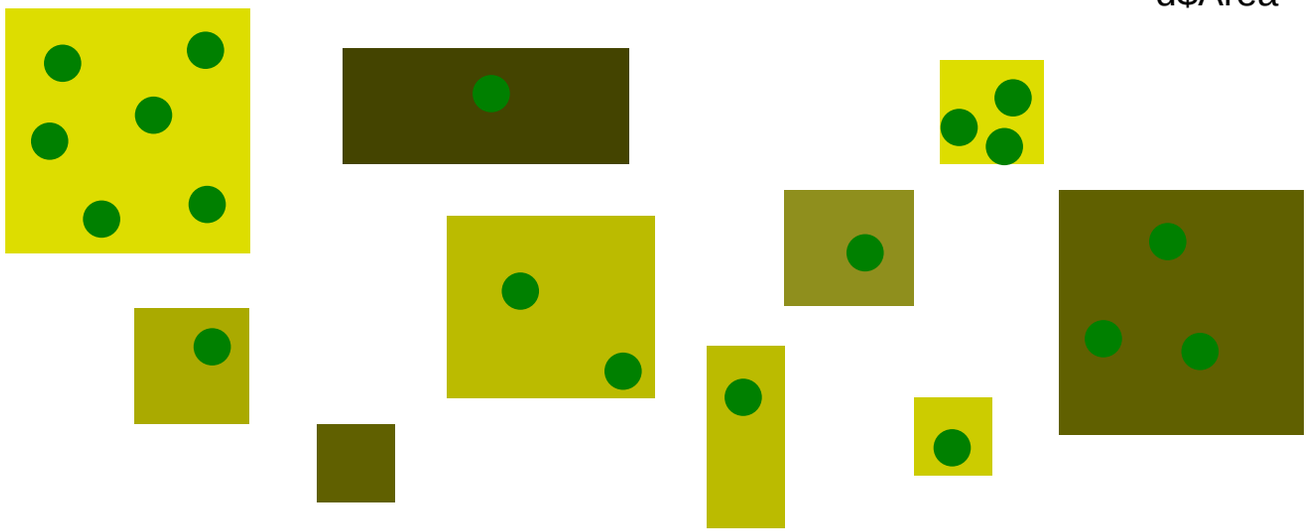


- 赤は明るさ $x = 0.9$, 青は $x = 0.1$
- 実線は `glm()` の推定結果，破線はデータ生成時に指定した関係

まとめ: glm() の offset 項わざで「脱」割算

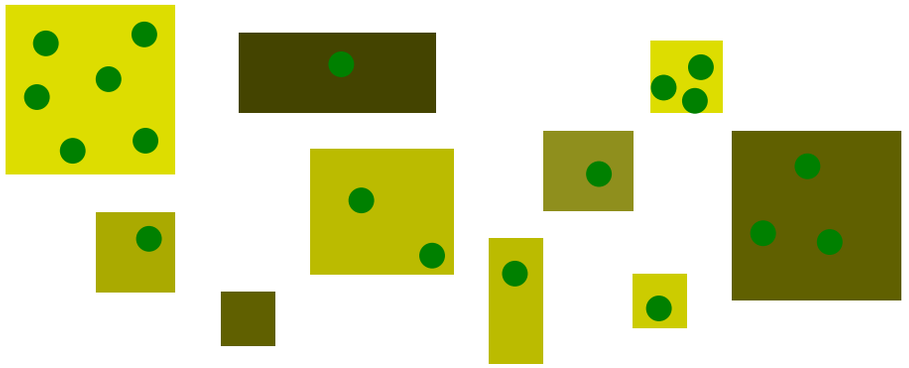
- 平均値が面積などに比例する場合は，この面積などを **offset 項** として指定する
- 平均 = 面積 × 密度，というモデルの**密度** を exp(線形予測子) として定式化する

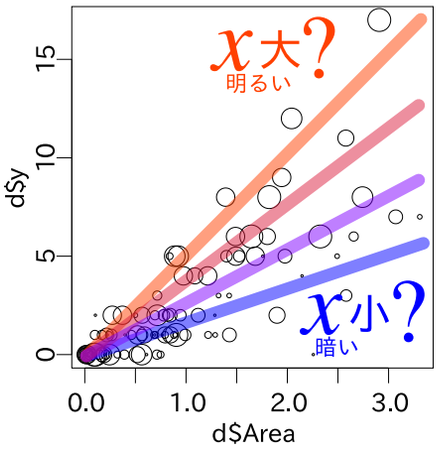




ここまでのまとめ

1. データ解析は統計モデリングだ
2. 割算するとわけわからなくなる
3. 統計モデリングの工夫 (リンク関数や offset 項わざなど) で現象を再現できるような統計モデリングを試みる





今回あつかわなかった GLM 関連事項

- ロジスティック回帰
- `summary(glm(...))` したときに表示される Wald 統計量と `Pr` の解釈
- 最尤推定法と deviance
- AIC によるモデル選択
- 過分散 (overdispersion) と GLMM への道

……などなど……

GLM よくある質問 (1) 「確率分布わからん」

どうやって確率分布を選べばいいんですか?

応答変数のタイプに注目して選んでください

- $y = 0, 1, 2, 3, \cdots$ (y の上限不明) なら**ポアソン分布** (`family = poisson`)
- $y = \{0, 1\}$, $y = \{0, 1, 2, \cdots, N\}$ なら**二項分布** (`family = binomial`)
- **連続かつ正値なら**ガンマ分布 (`family = Gamma`)
- **それ以外の連続値なら**正規分布 (`family = gaussian`)

R で一般化線形モデル: glm() 関数

	確率分布	乱数生成	パラメーター推定
(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.nb() は MASS library 中にある
- 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

GLM よくある質問 (2) 「もっとヘンな分布を!」

私のデータの確率分布はもっとヘンなんです!

GLMM や階層ベイズモデルに「ぱわーあっぷ」だ!

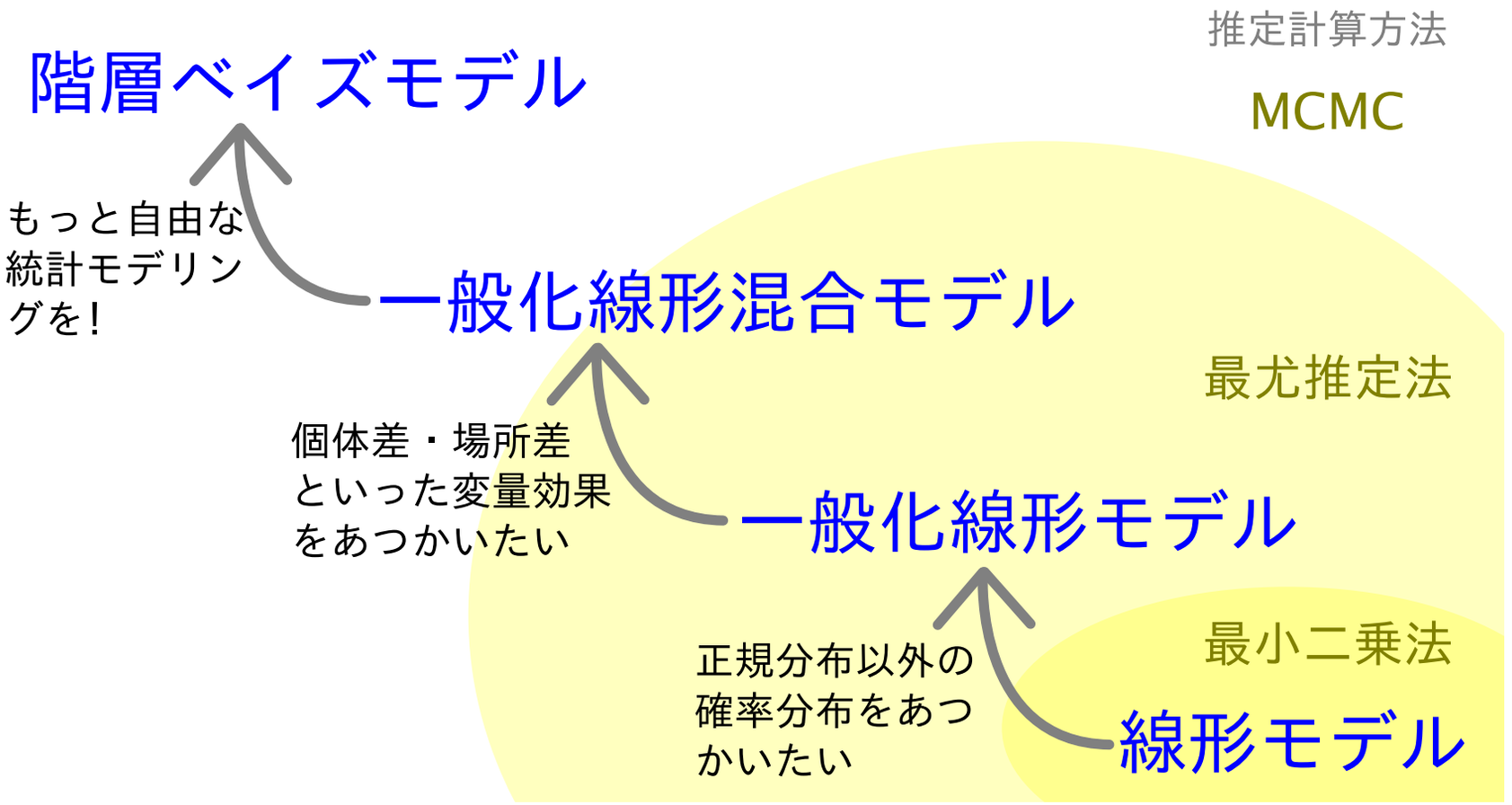
- まず，先ほどあげた「えらびかた」が基本です
- GLMM/階層ベイズモデルはこれらの**基本的な確率分布を「混ぜる」**ことでより複雑な状況に対処します
- 「混ぜる」ポイントは個体差・場所差といった random effects のモデリングです
- 「ぱわーあっぷ」にそなえて **GLM の基本**をよく勉強しましょう

このセミナーであつかう確率分布

- データのばらつきをあらわす確率分布
  - ポアソン分布 (Poisson distribution)
  - 二項分布 (Binomial distribution)
- その他 (ベイズモデルの事前分布で使用)
  - 正規分布 (Normal distribution, Gaussian —)
  - ガンマ分布 (Gamma distribution)

このくだり，おしまい

線形モデルの発展



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

いろいろダウンロードできます