

2012-11-02 k4

神戸大学理学部特別講義

「生物統計学」(2012年11月) 投影資料

全部で6回中の4回目

# 何でも割算するな! — GLM で解決できる

久保拓弥 [kubo@ees.hokudai.ac.jp](mailto:kubo@ees.hokudai.ac.jp)

<http://goo.gl/wijx2>

# 「生物統計学」の時間わり

11/1 (木) 午後

- (k1) 統計モデルとは何か? なぜ必要か?
- (k2) 確率分布と一般化線形モデル (GLM)
- (k3) モデル選択と統計学的検定

11/2 (金) 午前 — R の練習

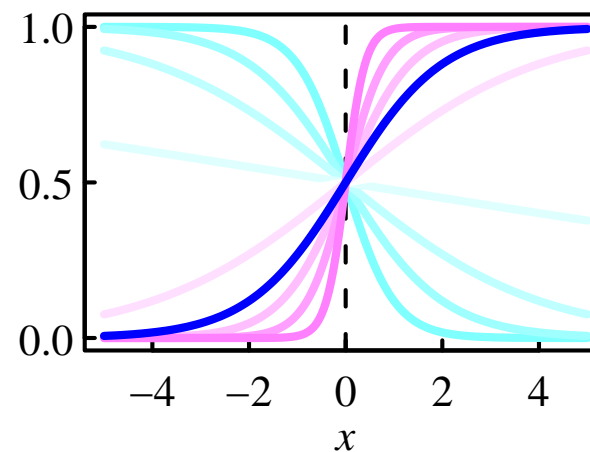
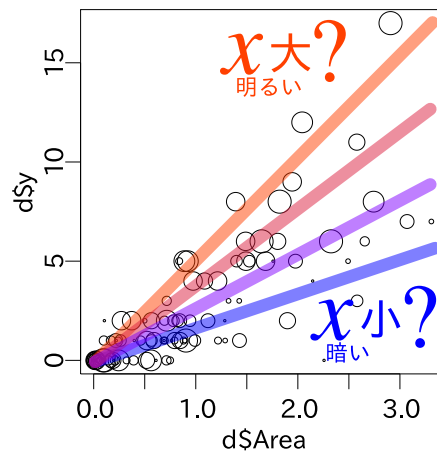
- (r1) R をちょっと使ってみる, 図を作る
- (r2) R で `glm()` 使って, また図を作る

11/2 (金) 午後

- (k4) 何でも割算するな! — GLM で解決できる
- (k5) 個体差をいれて階層ベイズモデルを作ろう
- (k6) いろいろな階層ベイズモデル

# この時間のハナシ

1. 割算解析やめましょう: その前に前回の復習
2. 「脱」割算の offset 項わざわざ: ポアソン回帰を強めてみる
3. ロジスティック回帰: おすすめできない解析と対比しつつ



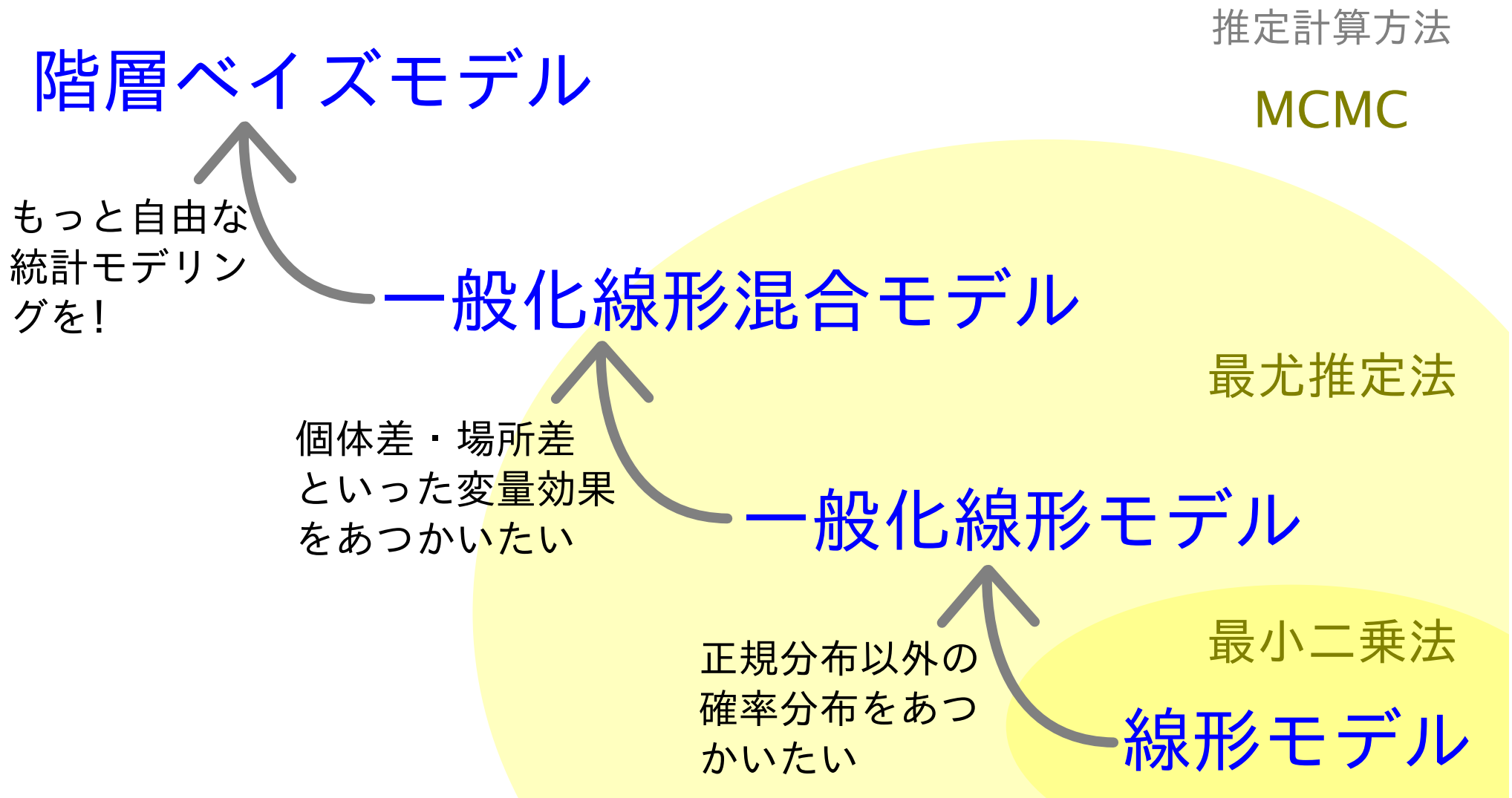
# 1. 割算解析やめましょう

その前に前回の復習

# 統計モデリング: 観測データのモデル化

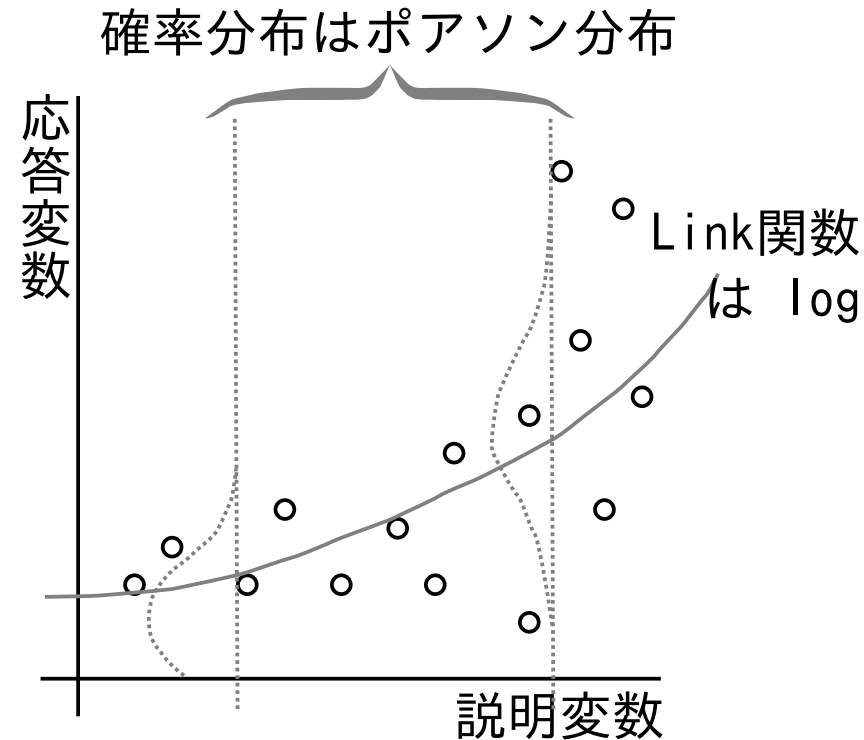
- 統計モデルは観測データのパターンをうまく**説明**できるようなモデル
- 基本的部品: **確率分布** (とそのパラメーター)
- データにもとづくパラメーター推定, **あてはまりの良さ**を定量的に評価できる

# 線形モデルの発展



統計モデル勉強のプラン: 線形モデルを発展させる

# カウントデータならポアソン回帰で!



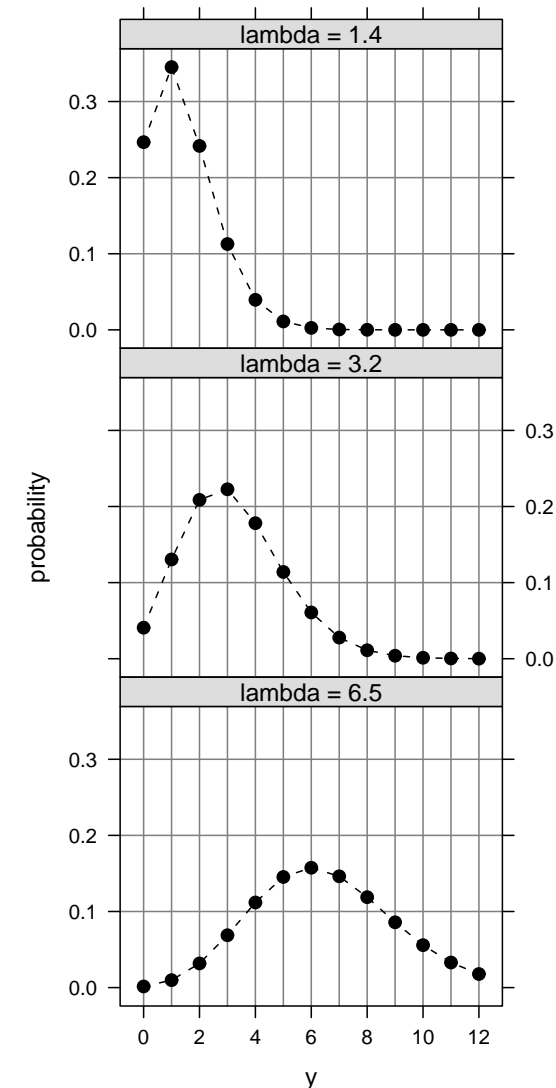
- ポアソン回帰は一般化線形モデルの一部
- 平均値とともに増大する分散に対応
- モデルによる予測はつねに非負

# ポアソン分布 (Poisson distribution) とは何か?

- 離散分布  $y_i \in \{0, 1, 2, \dots, \infty\}$
- 確率分布 (parameter:  $\lambda$ )

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

- 平均  $\lambda$ , 分散  $\lambda$
- 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 =  $f(\text{線形予測子})$  という関係がある
- 線形予測子:  $\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots$ , ただし  $x_i$  は 説明変数で  $\beta_i$  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 によるパラメータ推定

# R で一般化線形モデル: `glm()` 関数

	確率分布	乱数生成	パラメータ推定
(離散)	ベルヌーイ分布	<code>rbinom()</code>	<code>glm(family = binomial)</code>
	二項分布	<code>rbinom()</code>	<code>glm(family = binomial)</code>
	ポアソン分布	<code>rpois()</code>	<code>glm(family = poisson)</code>
	負の二項分布	<code>rnbinom()</code>	<code>glm.nb()</code>
(連続)	ガンマ分布	<code>rgamma()</code>	<code>glm(family = gamma)</code>
	正規分布	<code>rnorm()</code>	<code>glm(family = gaussian)</code>

- `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

# R の glm() 関数: 何を指定すればいい?

```
fit <- glm(  
  y ~ log.x,  
  family = poisson(link = "log")  
  data = d  
)
```

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

リンク関数の指定 (省略可)

data.frame の指定

- 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

- 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  $y \sim x$  と指定したとする

- 線形予測子  $z = a + bx$

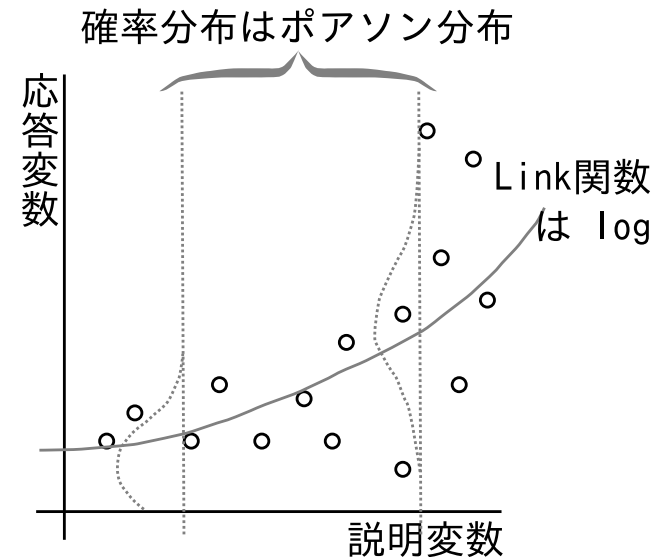
$a, b$  は推定すべきパラメーター

- 応答変数の平均値を  $\lambda$  とすると  $\log(\lambda) = z$

つまり  $\lambda = \exp(z) = \exp(a + bx)$

- 応答変数は平均  $\lambda$  のポアソン分布に従う:

$y \sim \text{Pois}(\lambda)$



# 説明したい統計モデリングのお作法

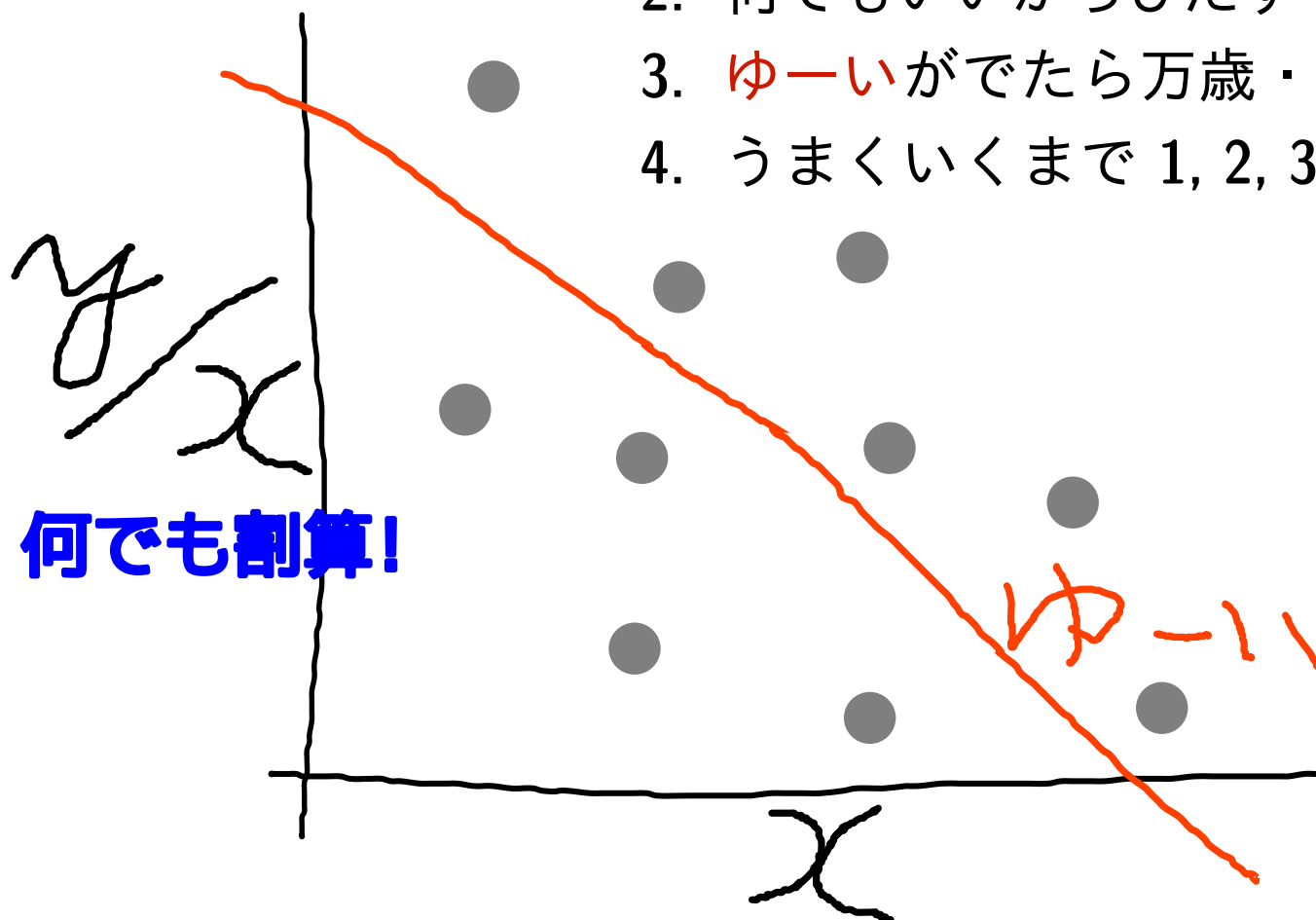
- 観測データの図をたくさん作ろう
- 観測データをどんな確率分布で表現できるか考えよう
- 「割算値」の統計モデリングはやめよう

つまり観測データの「もち味をいかした」  
「ひねくりまわさない」統計モデリング

# この悪しき「割算」な統計学

世間でよくみかけるおススメできない作法の例

1. データどんどん割算・割算
2. 何でもいいからひたすらセンをひく
3. ゆーいがでたら万歳・万歳
4. うまくいくまで 1, 2, 3 ぐるぐる



ちなみにこれは  $x$  と  $0/x$  を比較してるんだから、反比例みたいな偽「負の相関」ができるのはあたりまえ

# 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

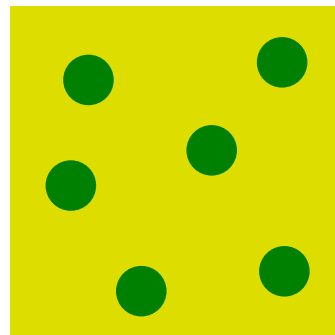
- 測定機器が内部で割算した値を出力する場合
- 割算値で作図せざるをえない場合があるかも



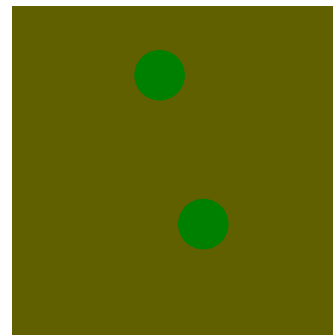
## 2. 「脱」 割算の offset 項わざわざ ポアソン回帰を強めてみる

## 例題: 調査区画内の個体密度は明るさで変わるか?

- 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  $\{0.1, 0.2, \dots, 1.0\}$  の 10 段階で観測した



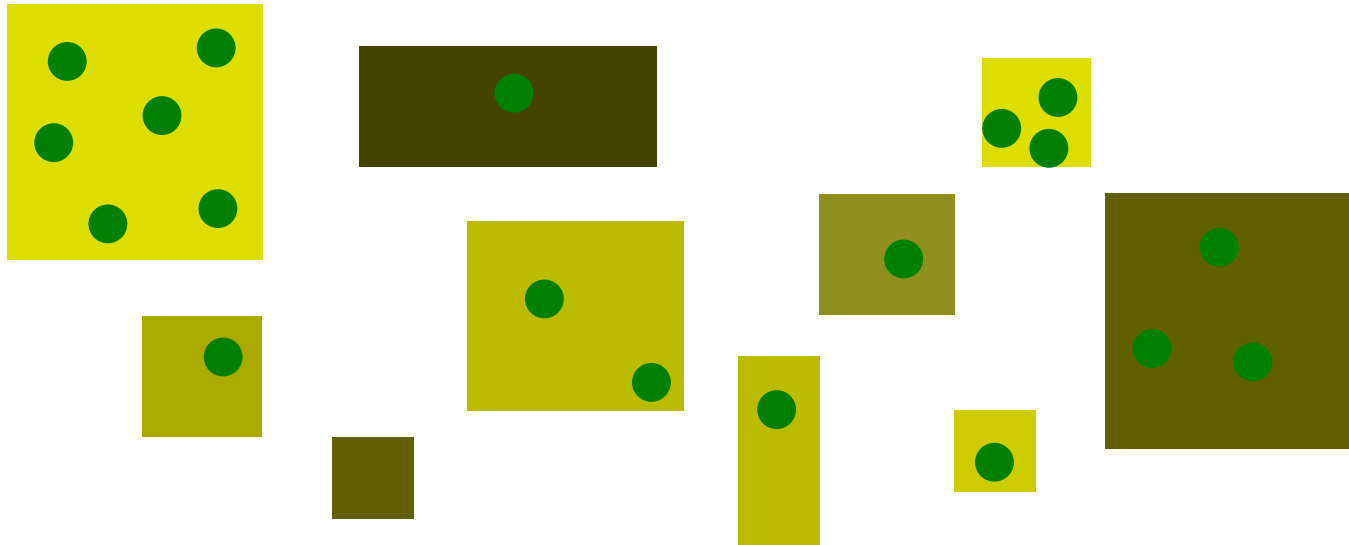
$x$ 大  
明るい



$x$ 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` 項わざわざうまく対処できる
- 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

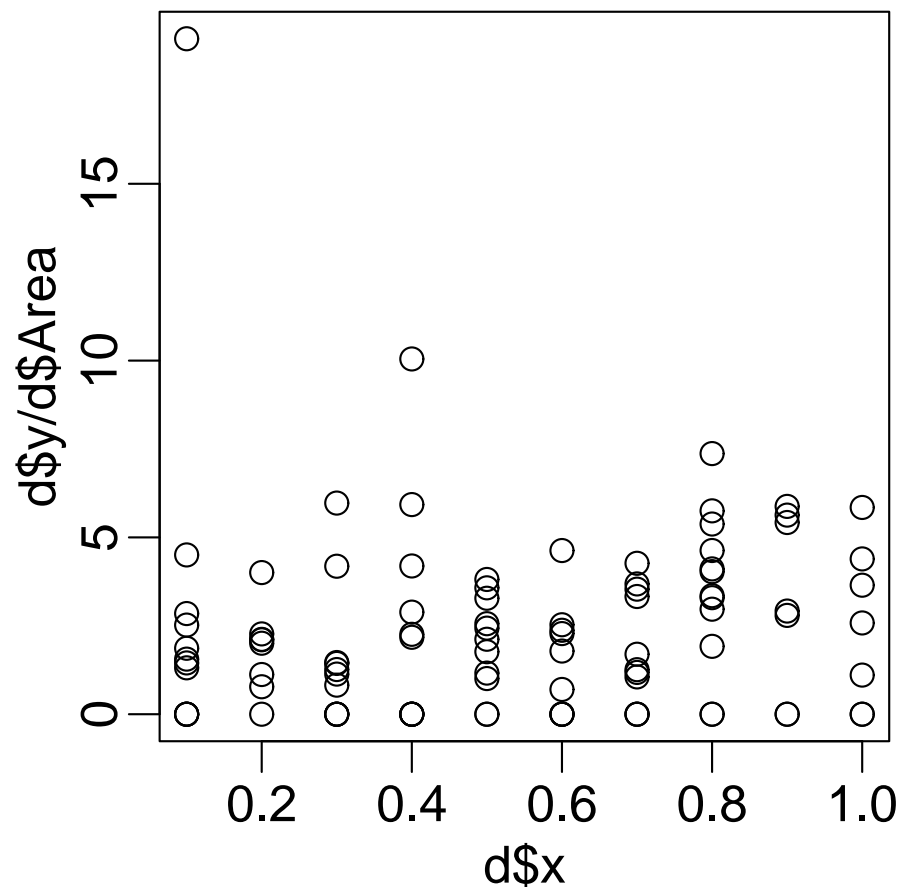
```
> load("d2.RData")
```

```
> 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

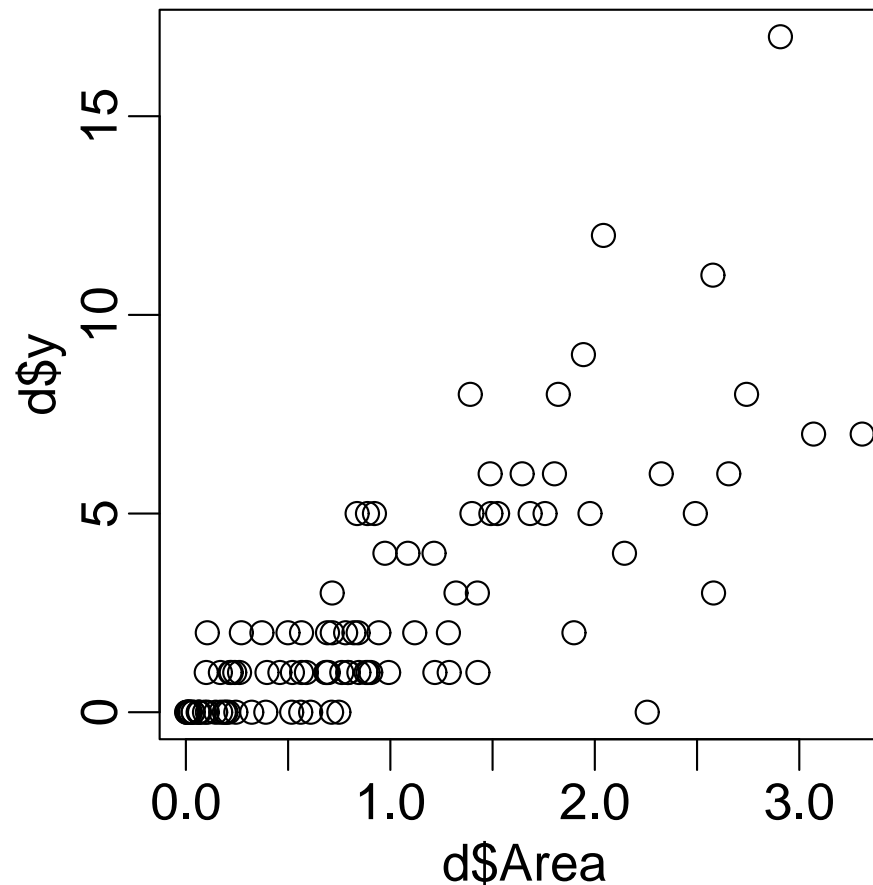
```
plot(d$x, d$y / d$Area)
```



- いまいちよくわからない……

# 面積 $A$ vs 個体数 $y$ の図

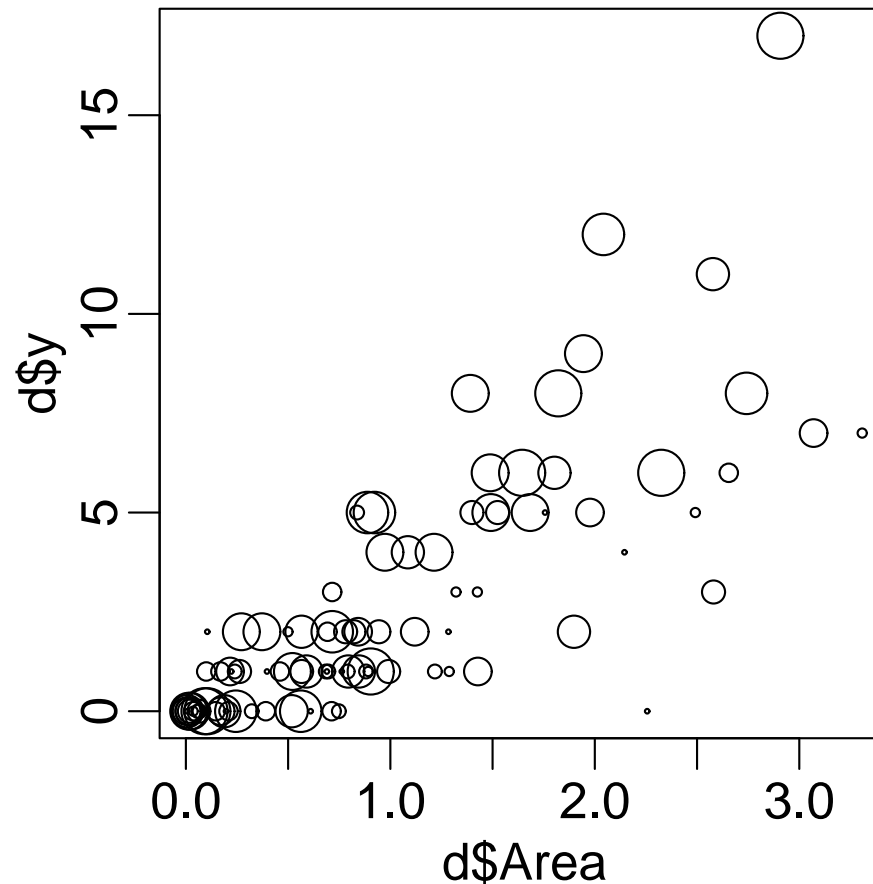
```
plot(d$Area, d$y)
```



- 面積  $A$  とともに区画内の個体数  $y$  が増大するようだ

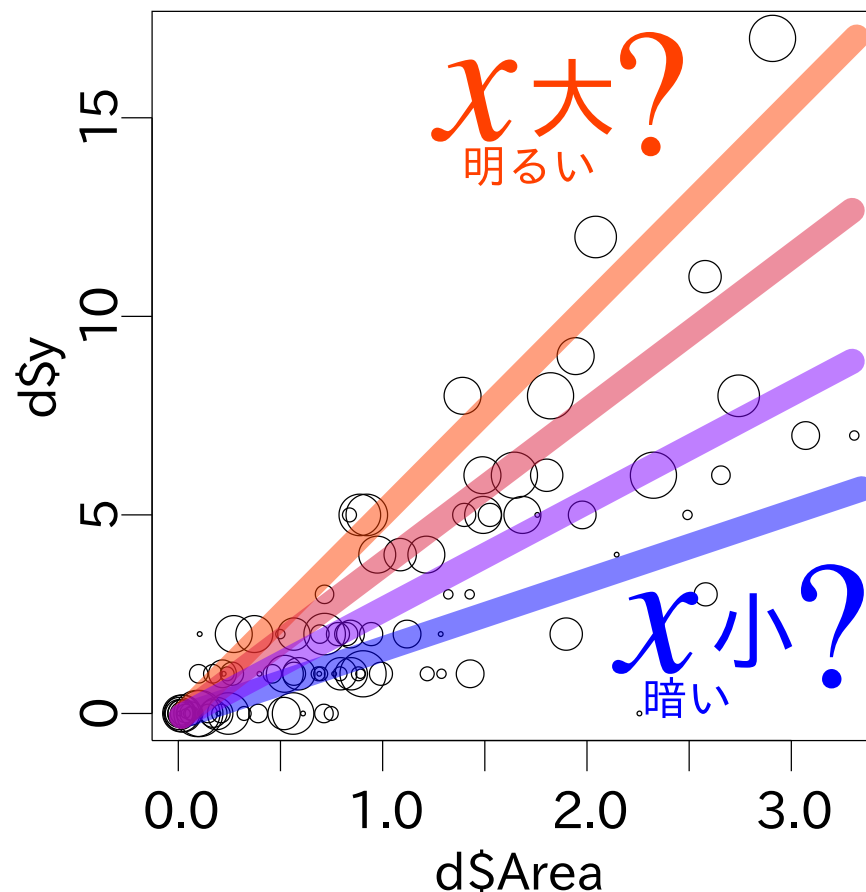
# 明るさ $x$ の情報 (マルの大きさ) も図に追加

```
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 × 密度」モデル

1. ある区画  $i$  の応答変数  $y_i$  は平均  $\lambda_i$  のポアソン分布にしたがうと仮定:

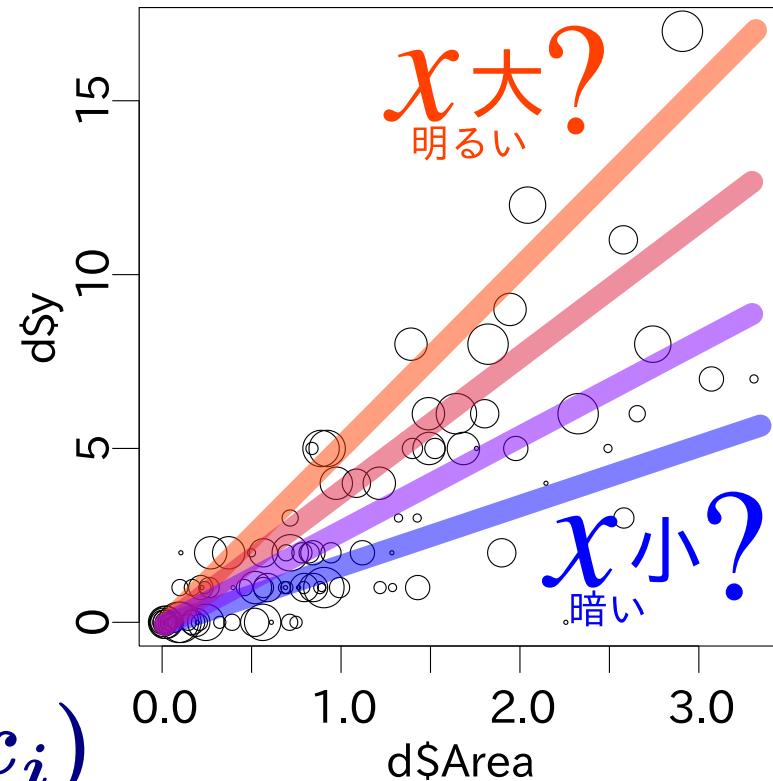
$$y_i \sim \text{Pois}(\lambda_i)$$

2. 平均値  $\lambda_i$  は面積  $A_i$  に比例し、密度は明るさ  $x_i$  に依存する

$$\lambda_i = A_i \exp(a + bx_i)$$

$$\lambda_i = \exp(a + bx_i + \log(A_i))$$

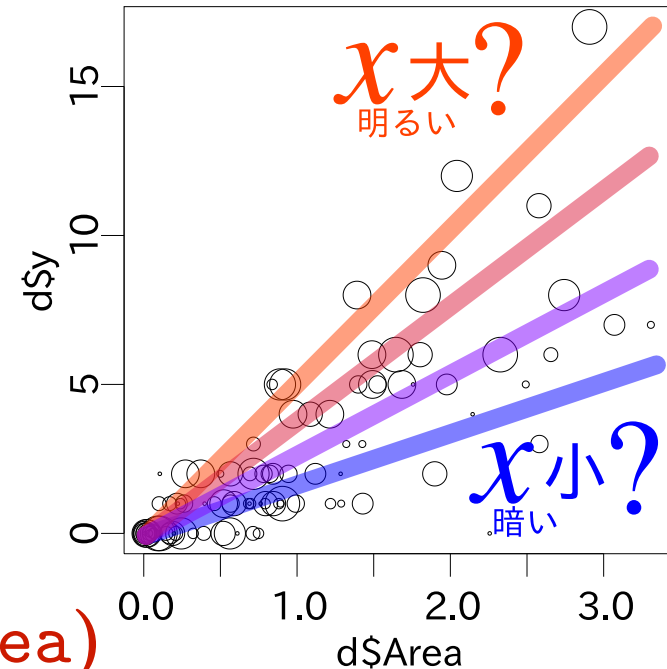
$$\log(\lambda_i) = a + bx_i + \log(A_i)$$



$\log(A_i)$  を offset 項とよぶ

# この問題は GLM であつかえる!

- family: poisson, ポアソン分布
- link 関数: "log"
- モデル式:  $y \sim x$
- offset 項の指定:  $\log(\text{Area})$



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

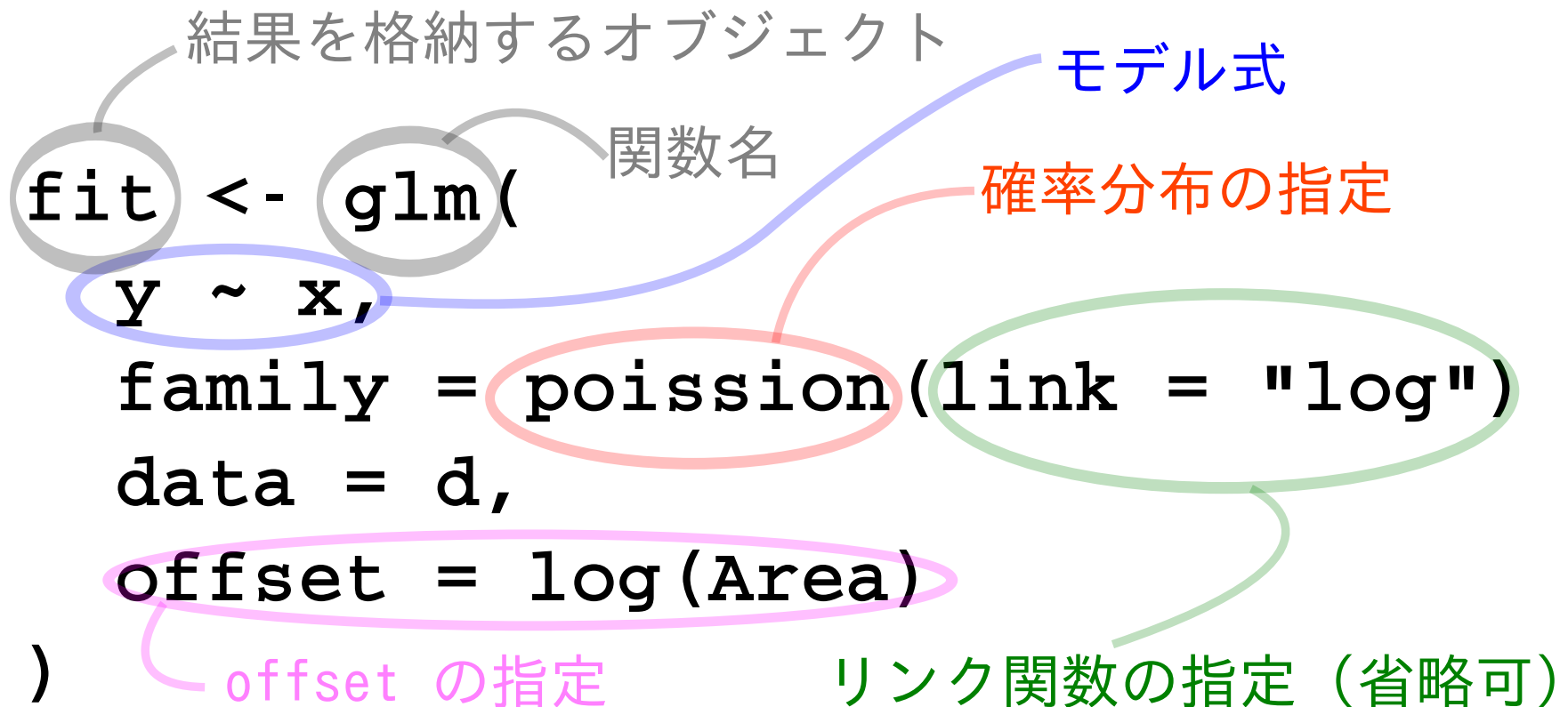
$a, b$  は推定すべきパラメーター

- 応答変数の平均値を  $\lambda$  とすると  $\log(\lambda) = z$

つまり  $\lambda = \exp(z) = \exp(a + b x + \log(\text{Area}))$

- 応答変数は平均  $\lambda$  のポアソン分布に従う:

# glm() 関数の指定



# 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))
```

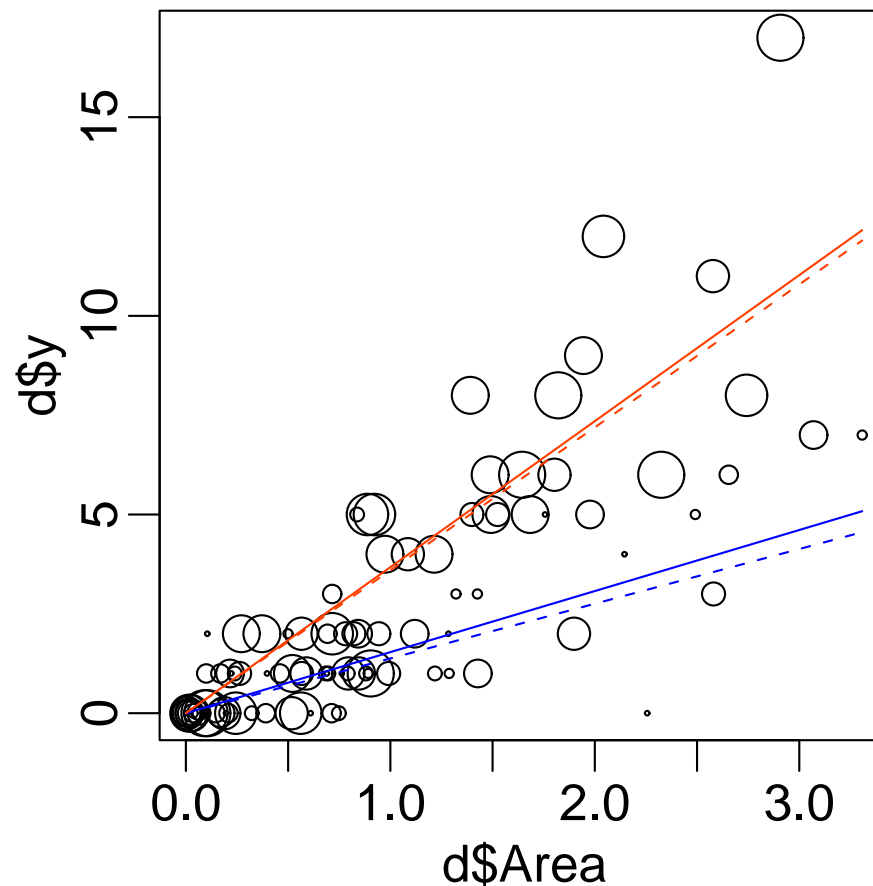
(... 略...)

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

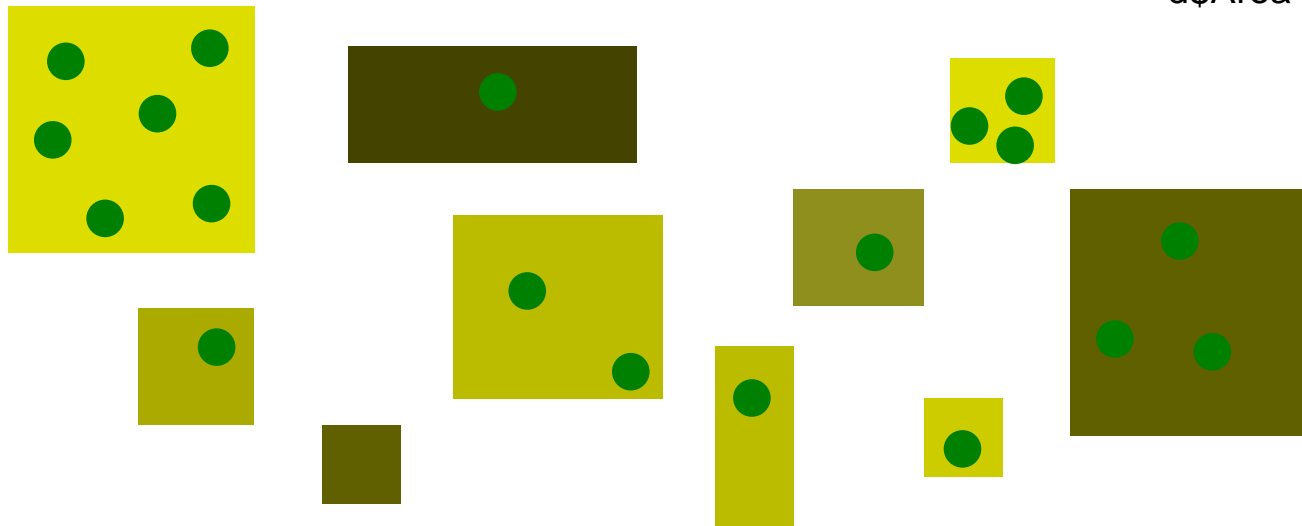
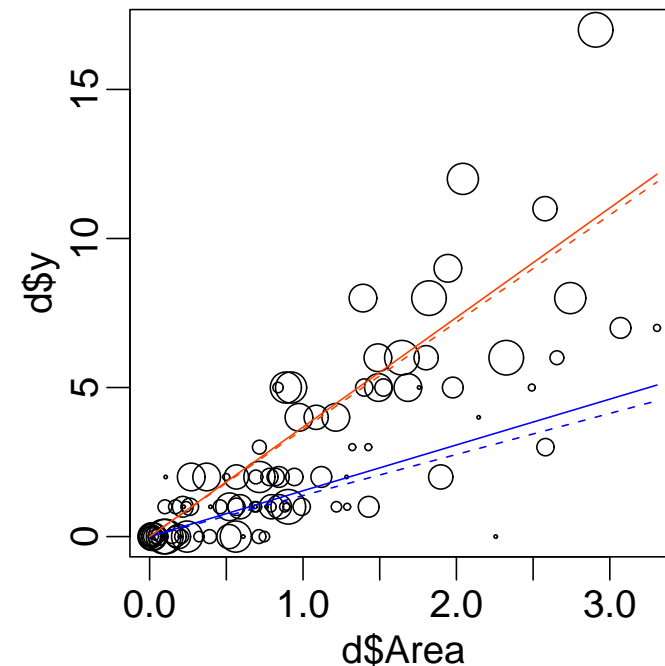
# 推定結果にもとづく予測を図にしてみる



- 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 項わざわざ「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  $\exp(\text{線形予測子})$  として定式化する



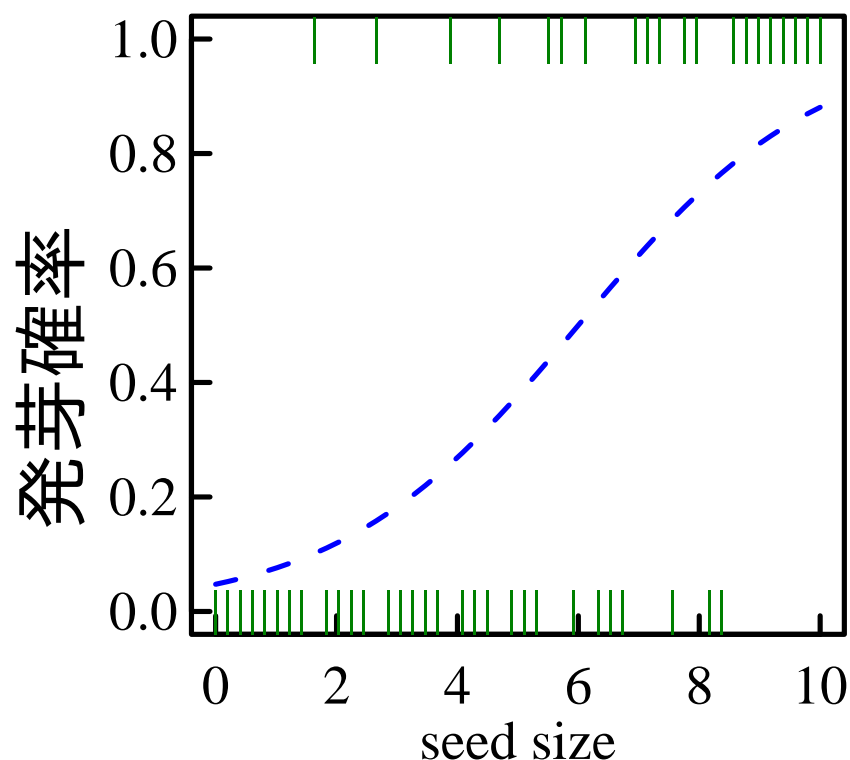
# 3. ロジスティック回帰

おススメできない解析と対比しつつ

# 架空植物の発芽実験データ

種子サイズと発芽確率の関係を調べる実験やってみた

| は観測データ (1 = 発芽した)



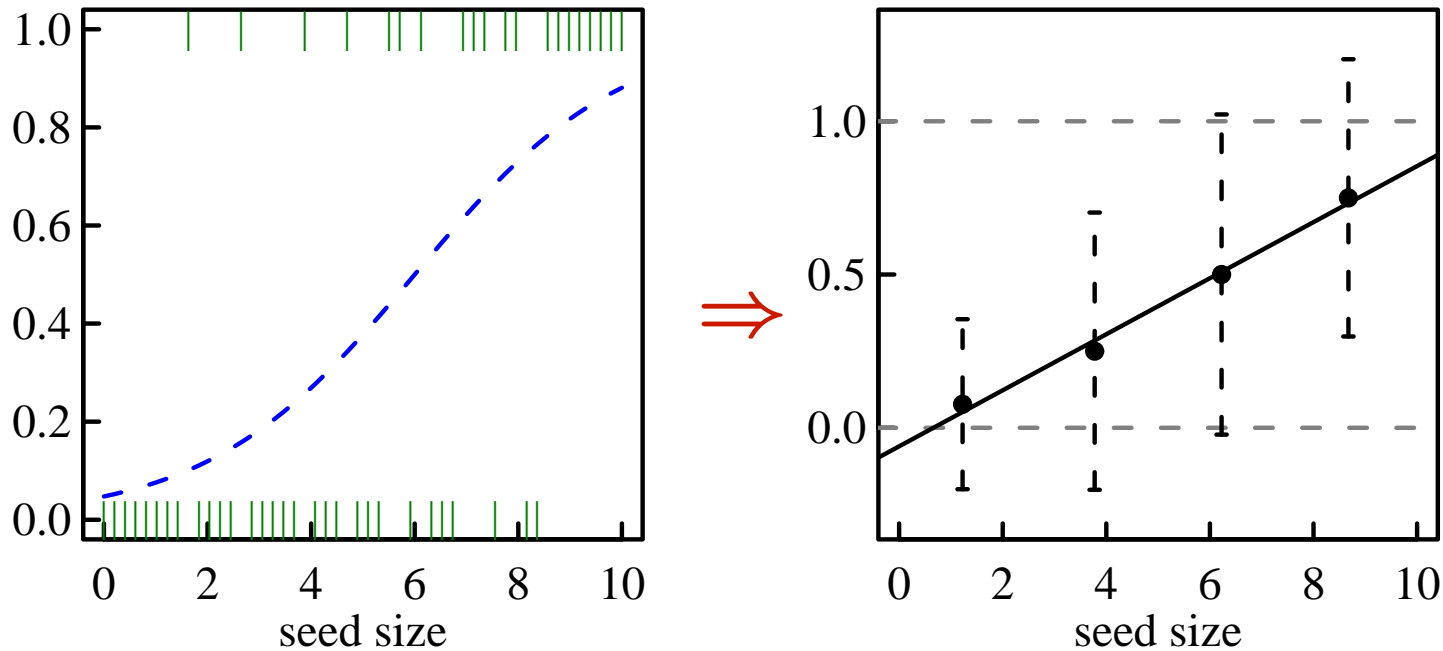
【「ホント」の発芽パターン】

- 種子が大きいほど発芽確率が高い
- 発芽確率は青破線 で示されているように上昇する

データから 青破線 (つまり真のモデル・母集団) を推定したい



# (よく見かける) おススメできない解析の一例



1. てきとーに種子サイズの区画を取る (上の例だと 4 区画)
2. 区画ごとに縦横の平均値など計算;  $\{0, 1\}$  データを割算値に
3. 何も考えずに統計ソフトウェアにほうりこむ

(直線回帰する or 「分散分析」する or 「検定」 & 多重比較する)

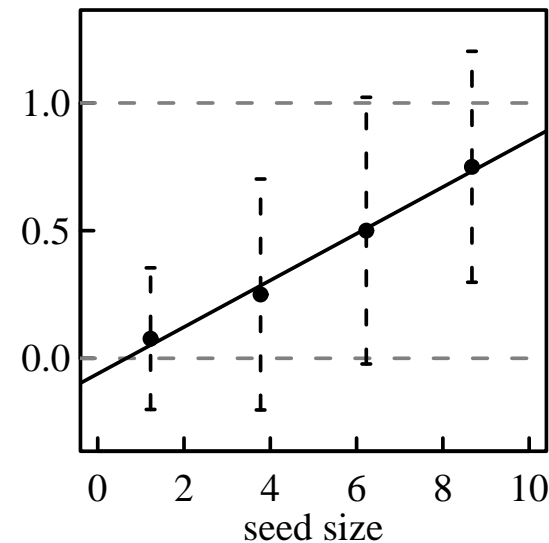
# なぜよろしくないか? データの特徴を無視

区画はてきとー

区画のとりかたで結果は変わる

割算すると情報が失われる

1 / 2 と 100 / 200 は違う!



等分散でもなければ正規分布でもない

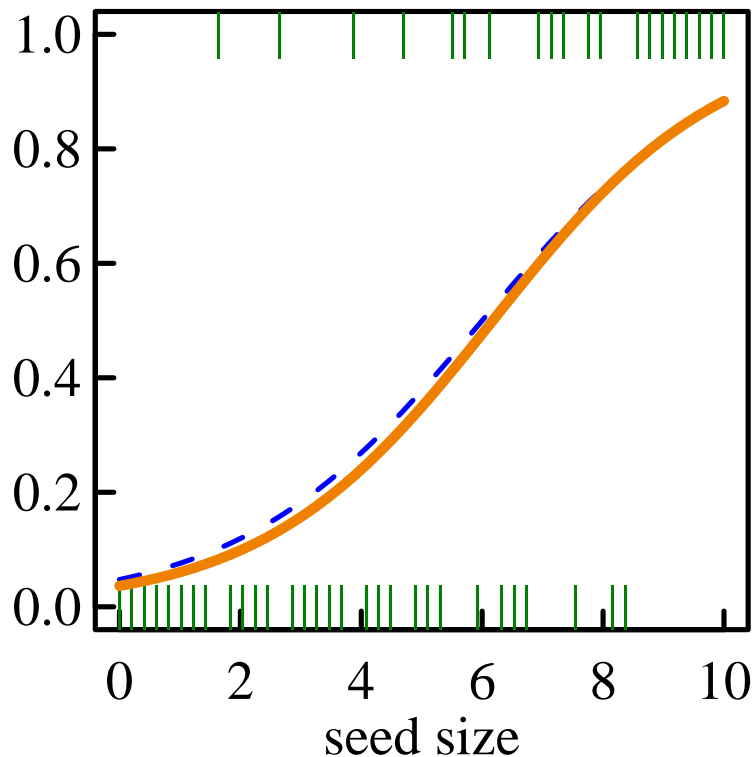
ということで直線回帰も分散分析も**使えん** — さらに, いわば母分散が異なる状況なので, ノンパラメトリック検定のたぐいもだめ

何を推定してるのだろうか?

発芽する確率がマイナスになったり, 1 をこえたりするモデルってのは ……? (変数変換すればいいって? そのワザは呪われてる)

# R の glm() で推定: ロジスティック回帰の例

発芽する・しないが二項分布にしたがうと仮定している



- 各種子について, そのサイズ ( $x$ ) と “発芽した or しなかった” の対応をみる
- 発芽確率  $q$  を以下のように仮定

$$q = \frac{1}{1 + \exp(-(a + bx))}$$

(logistic 式)

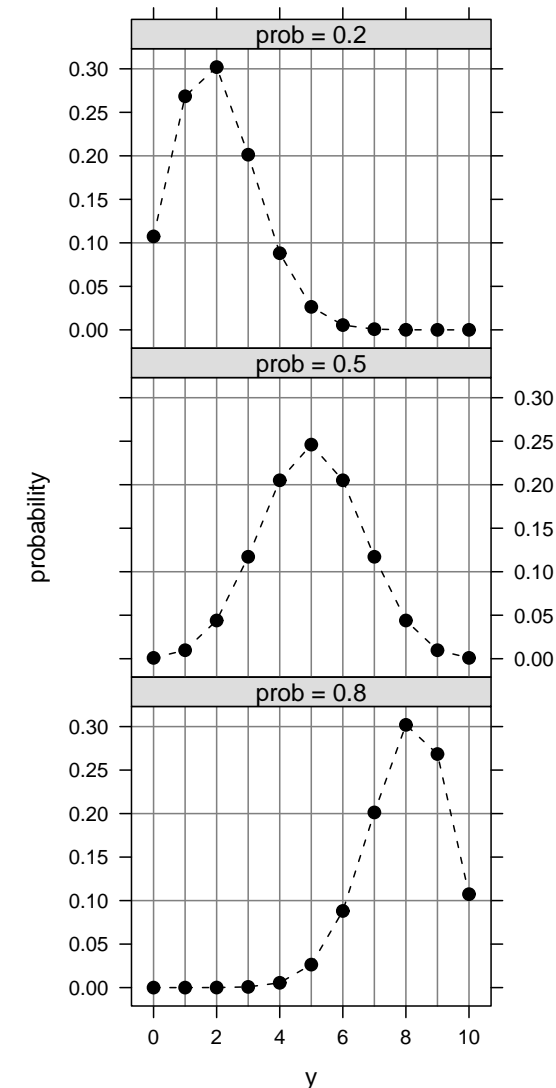
- パラメーター  $a$  と  $b$  の推定値を最尤推定法で計算する
- ここでは R の glm() 関数を使った (上の図の赤線が推定結果)

# 二項分布 (binomial distribution) とは何か?

- 離散分布  $y_i \in \{0, 1, 2, \dots, N\}$
- 確率分布 (parameter:  $q, N$ )

$$\binom{N}{y} q^y (1 - q)^{N-y}$$

- 平均  $Nq$ , 分散  $Nq(1 - q)$
- 上限のあるカウントデータに
- 例:  $N$  個体中  $y$  個体に反応があった, 死亡した, など

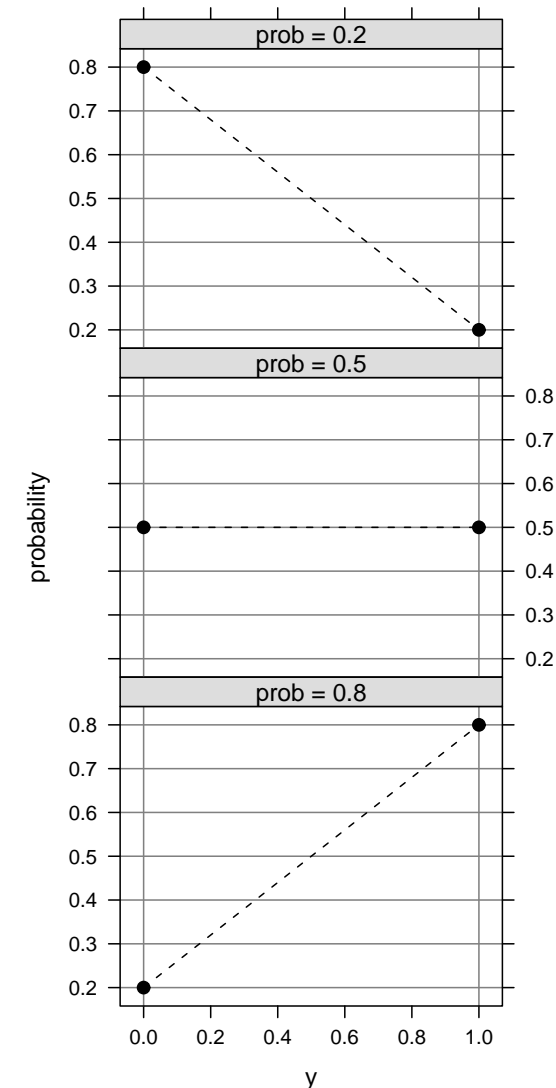


# ベルヌーイ分布 (Bernoulli distribution)

- 離散分布  $y_i \in \{0, 1\}$
- 確率分布 (parameter:  $q$ )

$$q^y (1 - q)^{1-y}$$

- 平均  $q$ , 分散  $q(1 - q)$
- 二項分布で  $N = 1$  の場合に該当する確率分布
- 例: ある個体で反応があった, 死亡した, など

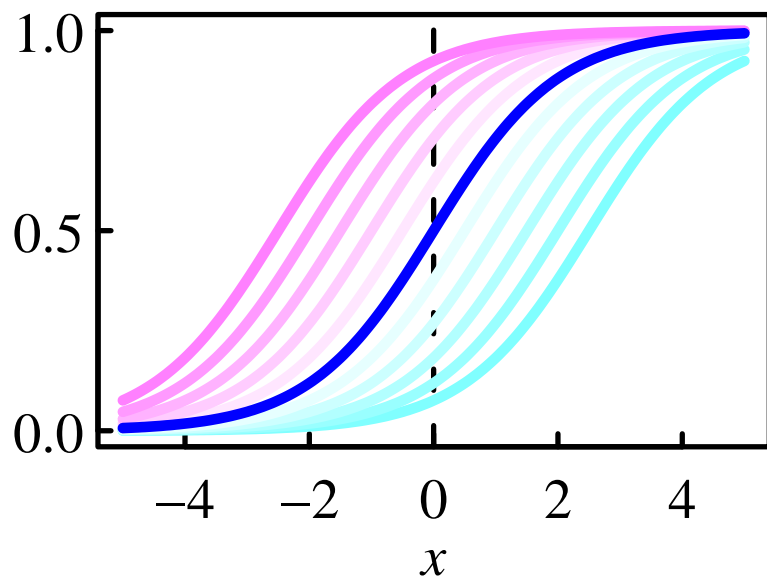


# そもそも「ロジスティック関数」って何?

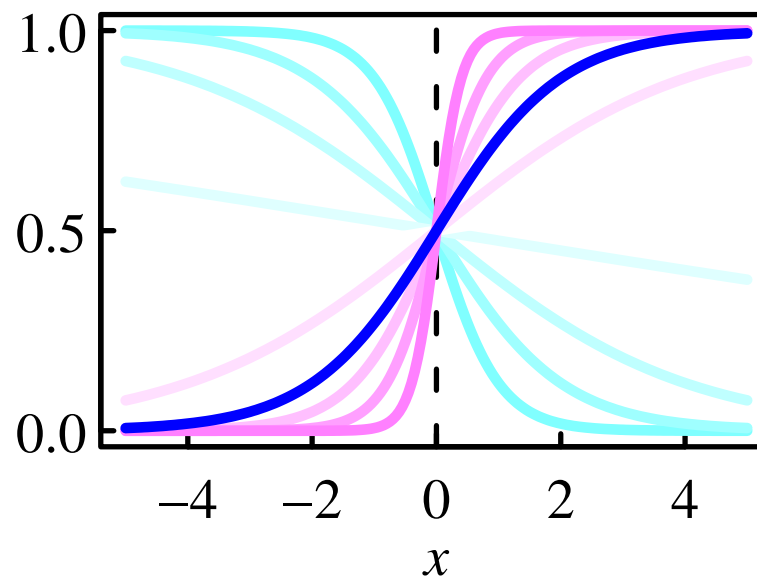
$$q = \frac{1}{1 + \exp(-(a + bx))}$$

( $\exp(Z) = e^Z$  のこと)

$a$  だけ変化させる



$b$  だけ変化させる



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

## ちょっと整理: logistic と logit

- logistic 関数

$$q = \frac{1}{1 + \exp(-(a + bx))} = \text{logistic}(a + bx)$$

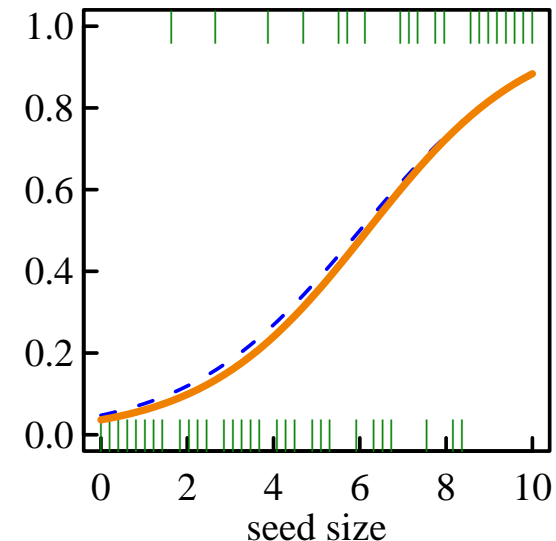
- logit 変換

$$\text{logit}(q) = \log \frac{q}{1 - q} = a + bx$$

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

# ロジスティック回帰の `glm()` 指定 (1)

- `family: binomial`, 二項分布
  - $y \in \{0, 1, 2, \dots, N\}$  というよう  
にある範囲のカウントデータである  
場合は二項分布で説明してみる
- `link` 関数: "logit"
  - これは `family = binomial` 時の  
「おススメ」 `link` 関数
- モデル式 (線形予測子  $z$ ): たとえば  $y \sim x$  と指定したとする

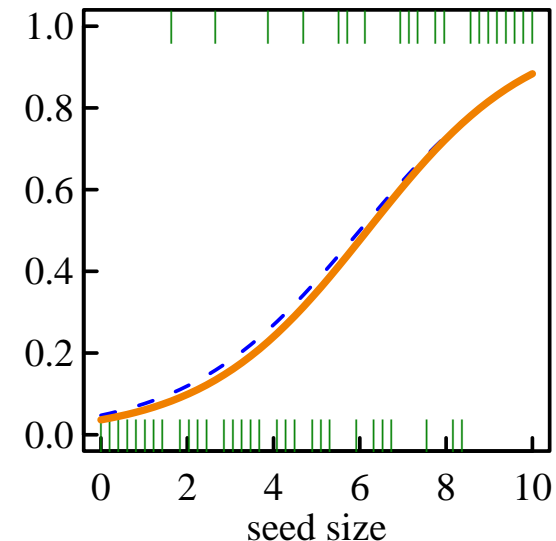


```
family = binomial(link = "logit")  
指定とは何をやっているのだろうか?
```



## ロジスティック回帰の `glm()` 指定 (2)

- `family: binomial`, 二項分布
- `link` 関数: "logit"
- モデル式 (線形予測子  $z$ ): たとえば  $y \sim x$  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

- 事象の生起確率 を  $q$  とすると  $\text{logit}(q) = z$

$$\text{つまり } q = \frac{1}{\exp(-z) + 1} = \frac{1}{1 + \exp(-(a + bx))}$$

- 応答変数 は確率  $q$  でサイズ  $N$  の二項分布に従う:

$$y \sim \text{Binom}(q, N)$$

# R の glm() 関数: 何を指定すればいい?

```
fit <- glm(  
  y ~ x,  
  family = binomial(link = "logit")  
  data = d  
)
```

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

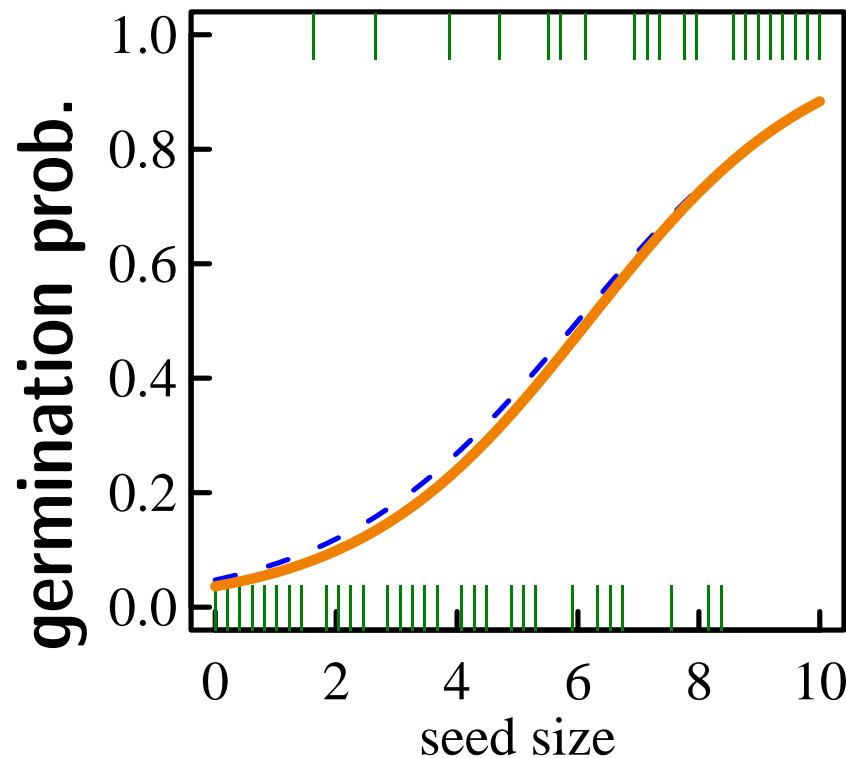
リンク関数の指定 (省略可)

data.frame の指定

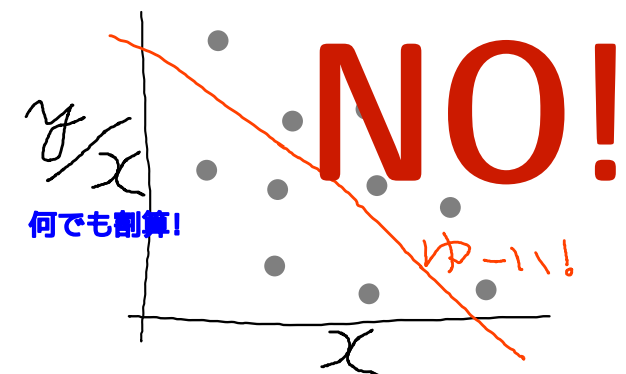
- モデル式 (線形予測子  $z$ ): 種子重  $x$  が説明変数
- link 関数: logit リンク関数
- family: binomial, 二項分布

# データにあわせてより良い統計モデリングを!

## おススメできないデータ解析を回避するための注意点

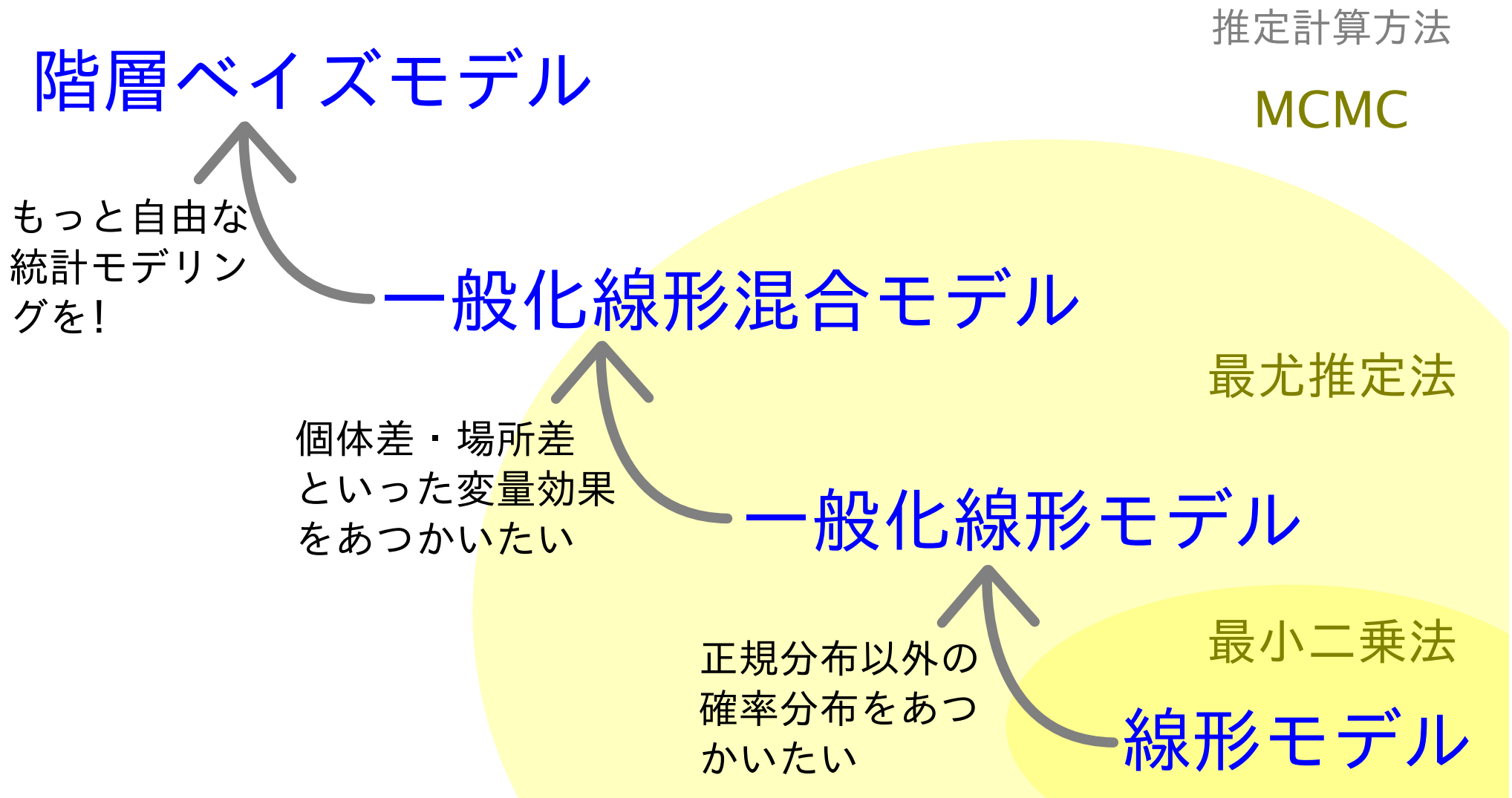


- むやみに **区画わけしない!**
- 何でも **割り算するな!**
- たくさん **図を描く**
- 「観測データを説明する **確率分布は何か?**」を考える



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!

# 線形モデルの発展



統計モデル勉強のプラン: 線形モデルを発展させる