

統計モデリングの基礎 (1)

統計モデル・確率分布・最尤推定

久保拓弥 `kubo@ees.hokudai.ac.jp`

多様性生物学概論

2016-01-20

ファイルのダウンロード: <http://goo.gl/76c4i>

ファイル更新時刻: 2016-01-20 15:30

1. はじめに

とりあえず，全体のながれなど

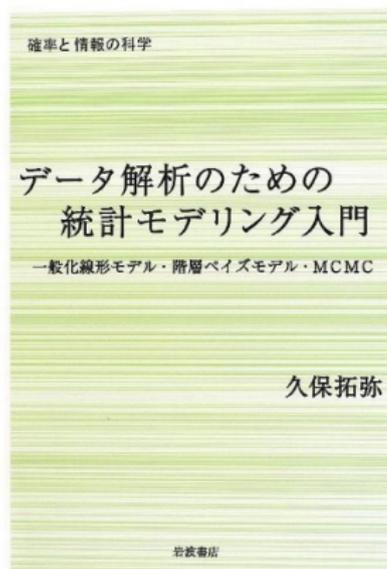
簡単な自己紹介その他あれこれ

とりあえず簡単な自己紹介: 久保拓弥 (北大・環境科学)

研究: 生態学データの統計モデリング

統計モデリングの教科書も書きました!

- 自分ではデータをとらない(野外調査・実験などをやらない)で、他のみなさんのデータ解析をすることが専門です
- これではあまりにも**寄生者**的なので、ときどきデータ解析に必要な統計モデリングの**解説**みたいなことをしております……

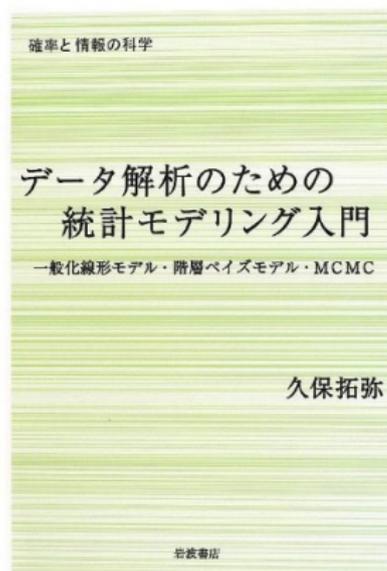


なんで，そんな本なんか書いたの?!

生態学の統計解析はあまりおもしろくなかった

この本ではブラックボックス統計学として批判

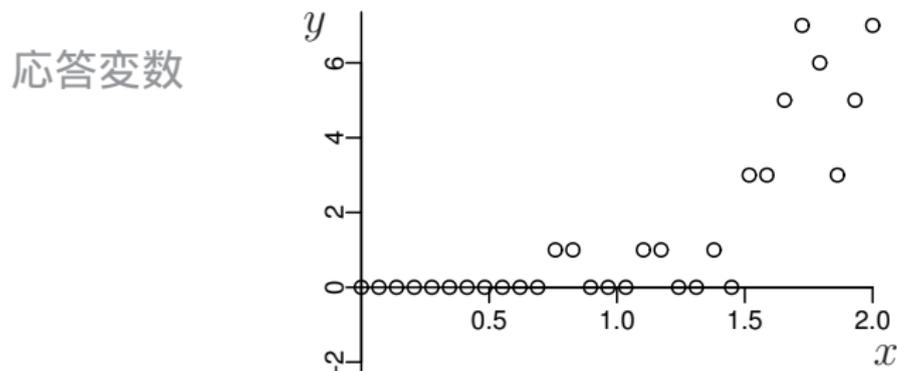
- 他人の論文の method section を読んで，内容を理解しないまま同じソフトウェアを使って， $p < 0.05$ なら何でも OK といった作業になりがち
- 統計ソフトウェアが何をやっているのかわかっていないので，誤用が多い
- こういう発想は，計算環境が貧弱だった昔の遺物



カタチ だけまねをするデータ解析
何がよくないのか?
例をあげて考えてみましょう

0 個, 1 個, 2 個と数えられるデータ

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

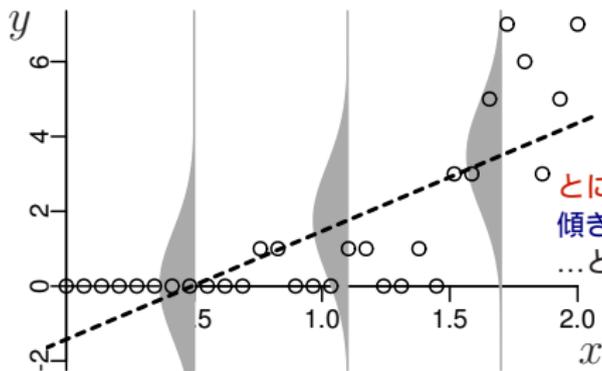


- たとえば x は植物個体の大きさ, y はその個体の花数
- 体サイズが大きくなると花数が増えるように見えるが.....
- この現象を表現する統計モデルは?

“何でもかんでも直線あてはめ” という安易な発想.....はギモン

正規分布・恒等リンク関数の統計モデル

応答変数



NO!

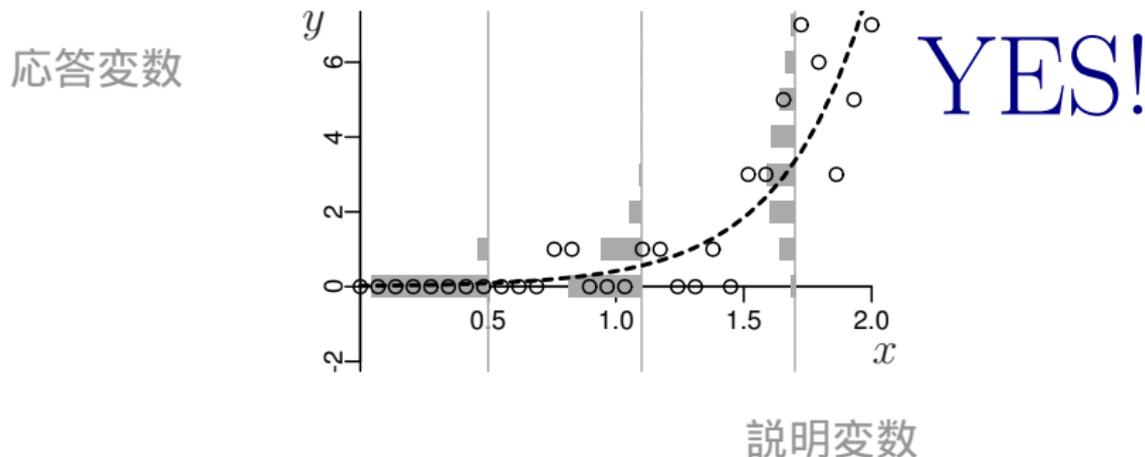
とにかくセンひきゃいいんでしょ
傾き「ゆるい」ならいいんでしょ
...という安易な発想のデータ解析

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？

データにあわせた“統計モデル”つかうとマシかもね？

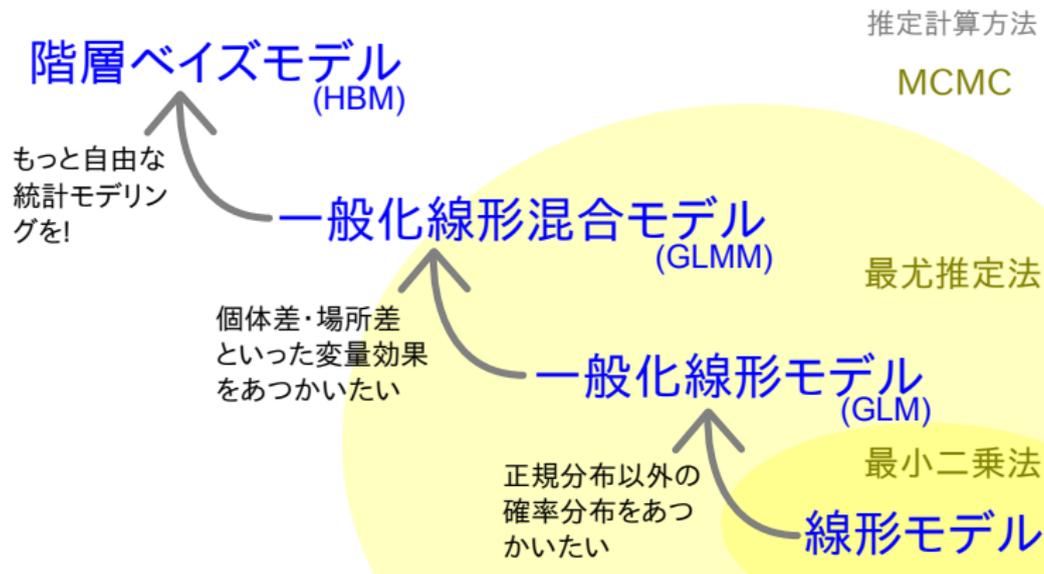
ポアソン分布・対数リンク関数の統計モデル



- 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

この講義で勉強する統計モデル

線形モデルの発展



ひとことでいうと「直線あてはめ」をどんどん改善する

この講義の流れ：例題を考えながら理解する

1. 統計モデル・確率分布・最尤推定
2. ポアソン分布の一般化線形モデル (GLM)
3. 二項分布の GLM
4. MCMC と階層ベイズモデル

単純化した例題にそって統計モデルを説明

統計モデルって何？

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づきが準備されていて、モデルがデータにどれくらい良くあてはまっているかを定量的に評価できる



この時間に説明したいこと

① はじめに

とりあえず、全体のながれなど

② サイコロの統計モデル

もっとも簡単な例のひとつとして

③ 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

④ 確率分布って何?

経験分布と理論分布

⑤ ポアソン分布のパラメーターの最尤推定

もっとももっともらしい推定?

さいゆうずいてい

⑥ 統計モデルの要点

乱数発生・推定・予測

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?

⑧ GLM の詳細を指定する

確率分布・線形予測子・リンク関数

⑨ 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

⑩ 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

⑪ “ 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

⑫ “ N 個のうち k 個が生きてる” タイプのデータ

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

⑬ ロジスティック回帰の部品

二項分布 binomial distribution と logit link function

統計モデルの重要な部品: 確率分布

- データ解析をするために**統計モデル**が必要
- 統計モデルの部品として“**データにあった**” **確率分布**が必要
- 確率分布は**パラメーター**などを指定する必要がある
- **パラメーターの値**はデータに基づいて決めたい

10. サイコロの統計モデル

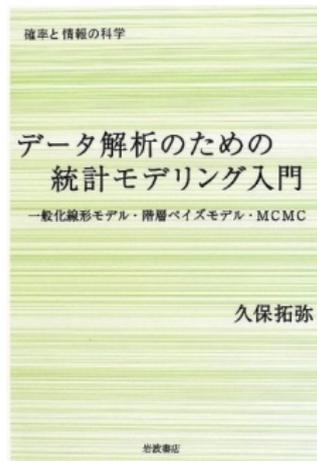
もっとも簡単な例のひとつとして

これで統計モデルの概念を考えよう

「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



「サイコロの統計モデル」を考えよう

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

```
> length(d)
```

```
[1] 1000
```

```
> table(d)
```

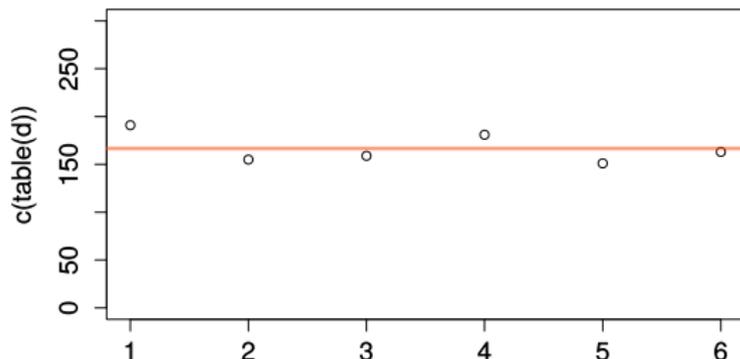
```
d
```

```
 1    2    3    4    5    6
```

```
191 155 159 181 151 163
```

```
> plot(1:6, c(table(d)), ylim = c(0, 300))
```

```
> abline(h = 1000 / 6, col = "#ff400080", lwd = 3)
```



架空データ

1000回サイコロふった

$1000/6 = 166.66\dots?$

「サイコロ」の確率分布は?

Categorical Distribution

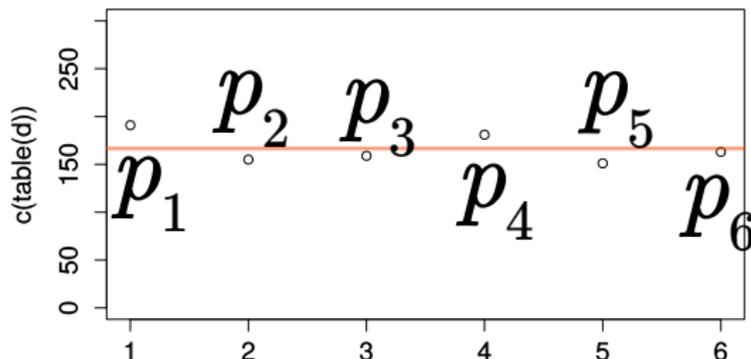
カテゴリカル分布



$$p(Y = k) = p_k$$

$$k \in \{1, 2, 3, 4, 5, 6\}$$

$$\sum_{k=1}^6 p_k = 1$$



架空データ

1000回サイコロふった

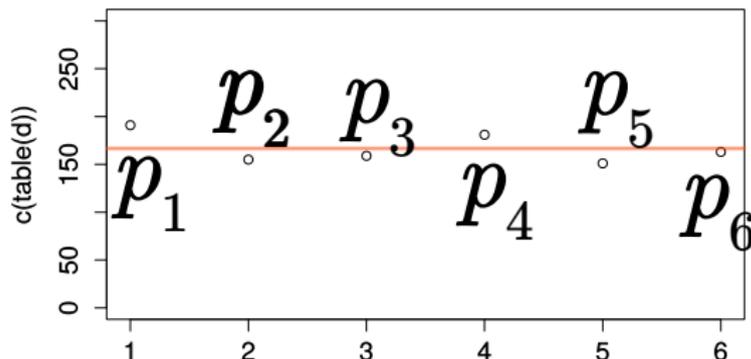
$1000/6 = 166.66\dots?$

確率分布のパラメーターは $\{p_k\}$

最尤推定量

$$\hat{p}_k = \frac{k \text{ の目が出た回数}}{1000}$$

```
> table(d)
d
 1    2    3    4    5    6
191 155 159 181 151 163
```



架空データ

1000回サイコロふった

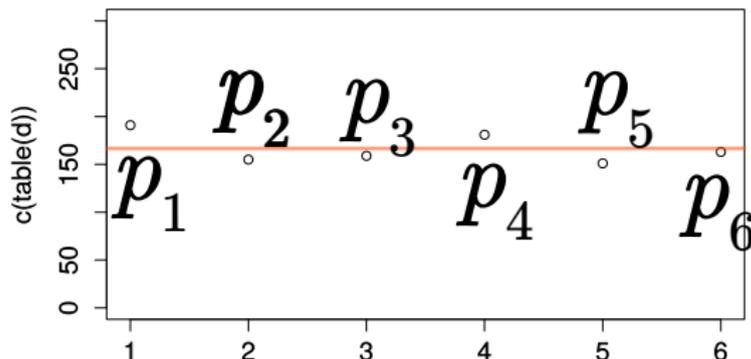
$1000/6 = 166.66\dots?$

「サイコロの統計モデル」にできること

パラメーターの推定

$$\hat{p}_k = \frac{k \text{ の目が出た回数}}{1000}$$

```
> table(d)
d
 1   2   3   4   5   6
191 155 159 181 151 163
```



架空データ

1000回サイコロふった

$1000/6 = 166.66\dots?$

「サイコロの統計モデル」にできること

乱数発生

```
> v.prob <- table(d) / 1000  
> replicate(8, table(sample(1:6, 1000,  
+ replace = TRUE, prob = v.prob)))
```

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]
1	185	201	174	202	179	193	198	197
2	156	159	158	164	160	141	154	151
3	173	161	195	180	177	153	147	173
4	176	171	173	177	189	174	200	190
5	147	157	152	139	138	171	122	130
6	163	151	148	138	157	168	179	159



「サイコロの統計モデル」にできること

予測

```
> # サイコロ 1000 回ふりを 1000 回やる
> sim1000 <- replicate(1000,
+ table(sample(1:6, 1000, replace = TRUE,
+ prob = v.prob)))
> # 5 よりも 6 が多く出る回数は?
> sum(sim1000[5,] > sim1000[6,])
[1] 247
> # 3 よりも 4 が多く出る回数は?
> sum(sim1000[3,] > sim1000[4,])
[1] 111
> # 5 より 6 が多いときに, 3 よりも 4 が多く出る回数は?
> sum((sim1000[5,] > sim1000[6,])
+ * (sim1000[3,] > sim1000[4,]))
[1] 35
```



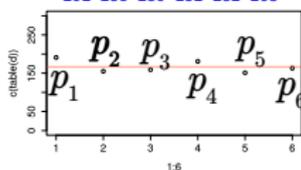
“確率分布” を部品にもつ統計モデル

できること: 推定・乱数発生・予測など

「サイコロの統計モデル」にできること
パラメーターの推定

$$\hat{p}_k = \frac{k \text{ の目が出た回数}}{1000}$$

```
> table(d)
d
 1  2  3  4  5  6
191 155 159 181 151 163
```



架空データ
1000回サイコロふった
1000/6 = 166.66...?

「サイコロの統計モデル」にできること
乱数発生

```
> v.prob <- table(d) / 1000
> replicate(8, table(sample(1:6, 1000,
+ replace = TRUE, prob = v.prob)))
```

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]
1	185	201	174	202	179	193	198	197
2	156	159	158	164	160	141	154	151
3	173	161	195	180	177	153	147	173
4	176	171	173	177	189	174	200	190
5	147	157	152	139	138	171	122	130
6	163	151	148	138	157	168	179	159



「サイコロの統計モデル」にできること
予測

```
> # サイコロ 1000 回ふりを 1000 回やる
> sim1000 <- replicate(1000,
+ table(sample(1:6, 1000, replace = TRUE,
+ prob = v.prob)))
> # 5 よりも 6 が多く出る回数は?
> sum(sim1000[5, ] > sim1000[6, ])
[1] 247
> # 3 よりも 4 が多く出る回数は?
> sum(sim1000[3, ] > sim1000[4, ])
[1] 111
> # 5 よりも 6 が多しときに 3 よりも 4 が多くアス回数は?
```



「サイコロの統計モデル」にできること
モデル選択や検定

モデル選択

モデル1: p_k がすべて等しい
モデル2: p_k がすべて異なる

「次」の
データ



「予測力」の高い
モデルを選ぶ



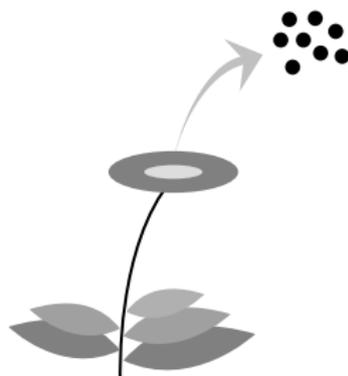
1000回サイコロふった

11. 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

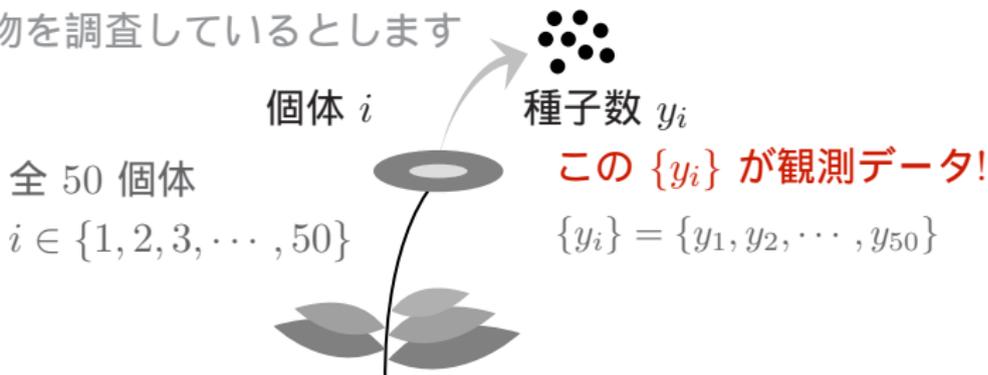


理由: よけいなことは考えなくてすむので

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



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
```

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



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

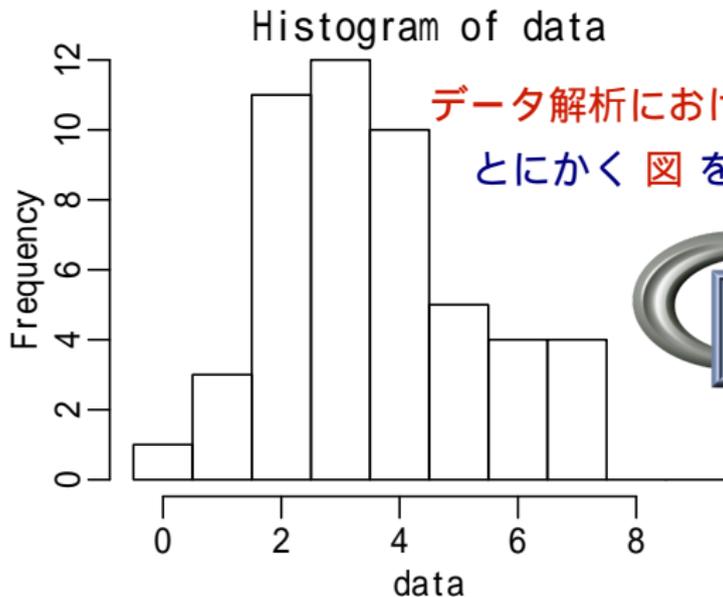
```
> table(data)
```

```
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))
```



データ解析における最重要事項
とにかく  を描く!

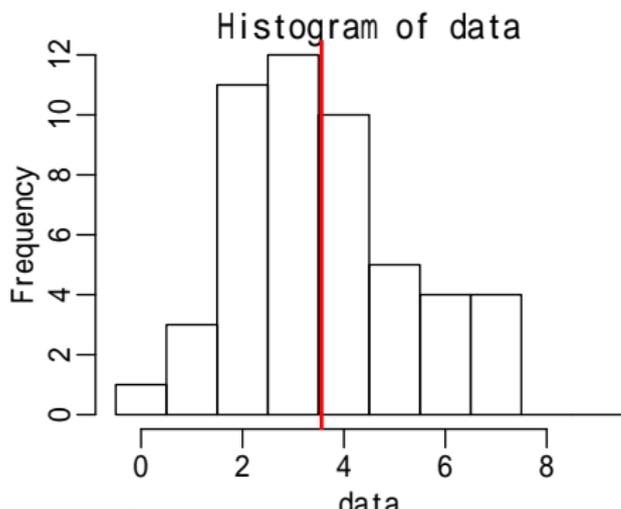


標本平均という統計量

```
> mean(data)
```

```
[1] 3.56
```

```
> abline(v = mean(data), col = "red")
```



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
```

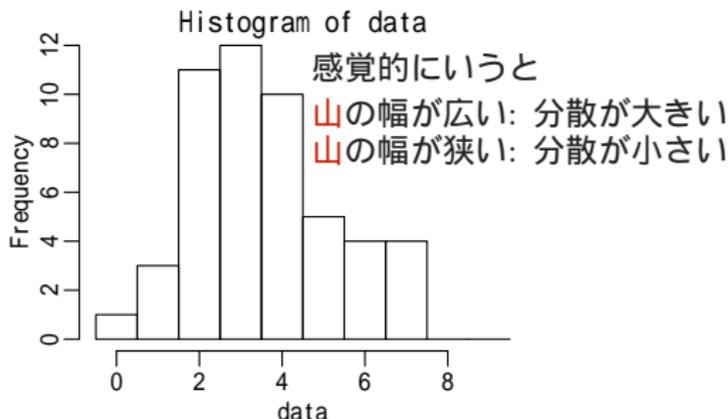
標本標準偏差 とは標本分散の平方根 ($SD = \sqrt{\text{variance}}$)

```
> sd(data)
```

```
[1] 1.7280
```

```
> sqrt(var(data))
```

```
[1] 1.7280
```



12. 確率分布って何？

経験分布と理論分布

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

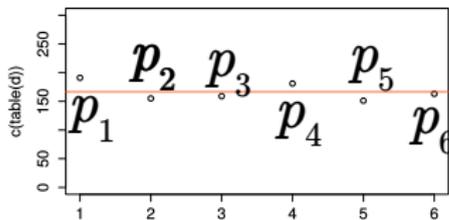
数式で定義される **理論的な分布** がある

「サイコロの統計モデル」にできること

パラメーターの推定

$$\hat{p}_k = \frac{k \text{ の目が出た回数}}{1000}$$

```
> table(d)
d
 1   2   3   4   5   6
191 155 159 181 151 163
```



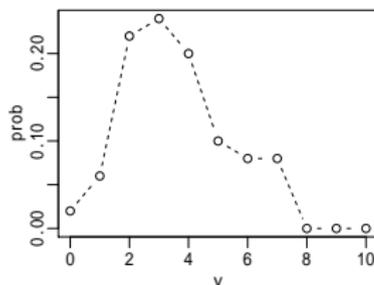
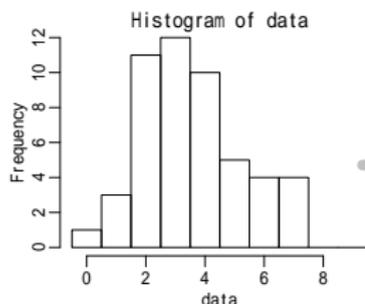
架空データ

1000回サイコロふった

$1000/6 = 166.66\dots?$

“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 $y = \{0, 1, 2, \dots\}$ 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 $\{p_0, p_1, p_2, \dots, p_{99}, p_{100}, \dots\}$ 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

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

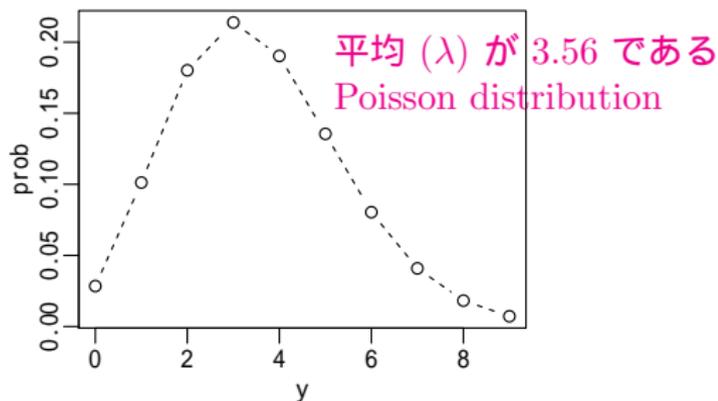
- $y!$ は y の階乗で, たとえば $4!$ は $1 \times 2 \times 3 \times 4$ をあらわしています.
- $\exp(-\lambda) = e^{-\lambda}$ のこと ($e = 2.718 \dots$)
- ここではなぜポアソン分布の確率計算が上のようになるのかは説明しません— まあ, こういうもんだと考えて先に進みましょう

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

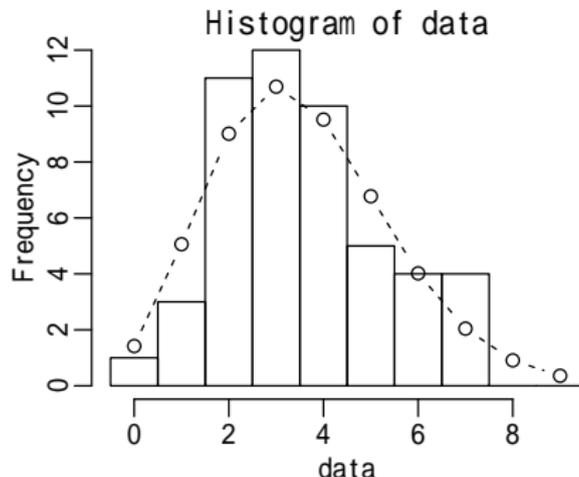
```
> y <- 0:9 # これは種子数 (確率変数)
> prob <- dpois(y, lambda = 3.56) # ポアソン分布の確率の計算
> plot(y, prob, type = "b", lty = 2)
```

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



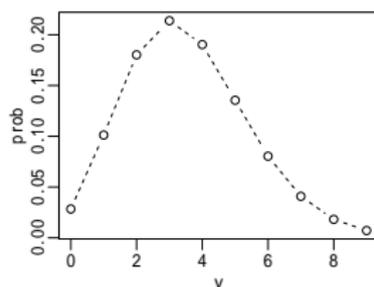
y	prob
1	0.02843882
2	0.10124222
3	0.18021114
4	0.21385056
5	0.19032700
6	0.13551282
7	0.08040427
8	0.04089132
9	0.01819664
10	0.00719778

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: $\lambda = \text{平均} = \text{分散}$
- $y \in \{0, 1, 2, \dots, \infty\}$ の値をとり, すべての y について和をとると 1 になる

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

どういう場合にポアソン分布を使う?

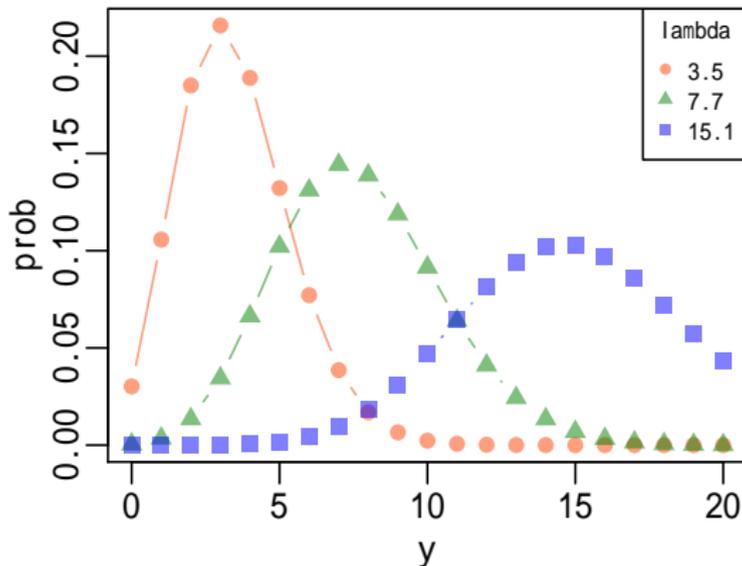
統計モデルの部品としてポアソン分布が選んだ理由:

- データに含まれている値 y_i が $\{0, 1, 2, \dots\}$ 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

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

λ は平均をあらわすパラメーター



8. 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?

尤度 (likelihood) とは何か?

- 最尤推定法では、^{ゆうど}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:

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

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!）

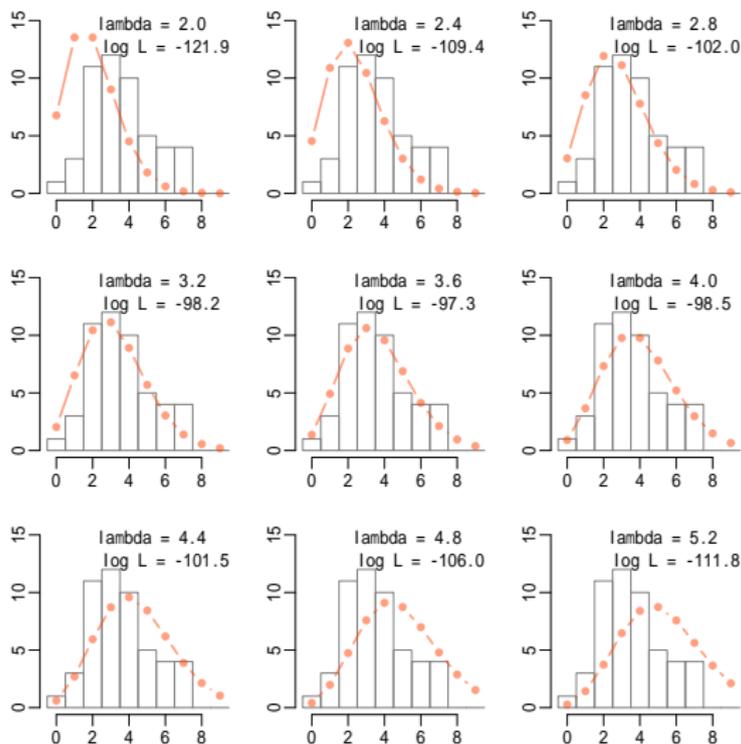
そこで，パラメーターの最尤推定では，**対数尤度関数** (log likelihood function) を使う

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

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

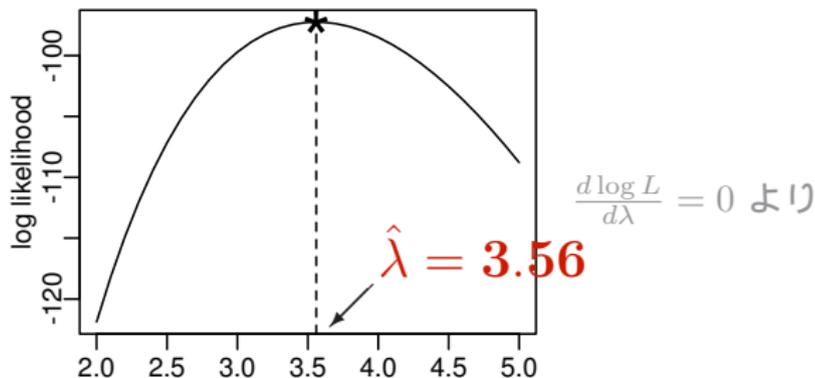
まずは，平均をあらわすパラメーター λ を変化させていったときに，ポアソン分布のカタチと対数尤度がどのように変化するかを調べてみましょう

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

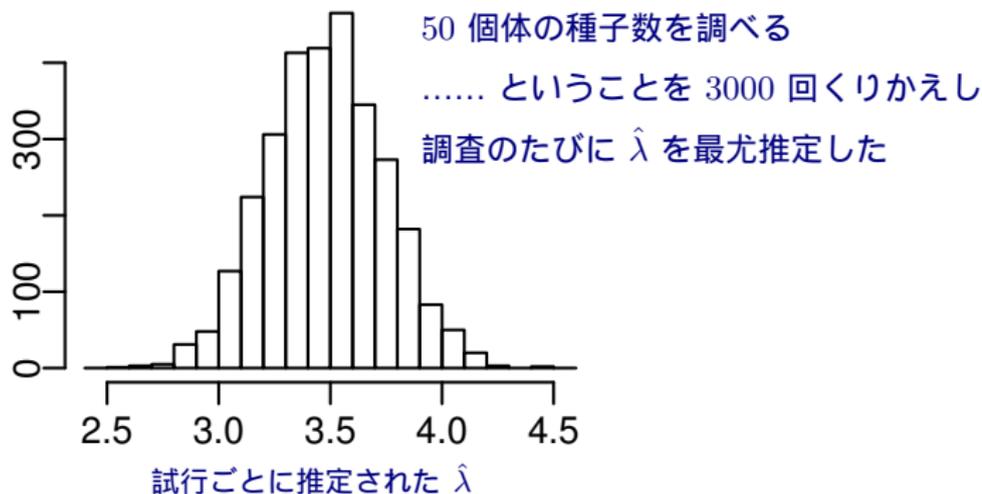
$$\text{対数尤度 } \log L(\lambda) = \sum_i (y_i \log \lambda - \lambda - \sum_k^{y_i} \log k)$$



- 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



データは有限なので真の λ はわからない

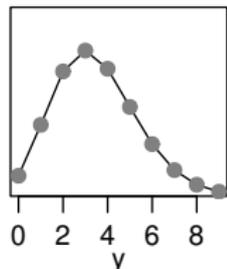
5. 統計モデルの要点

乱数発生・推定・予測

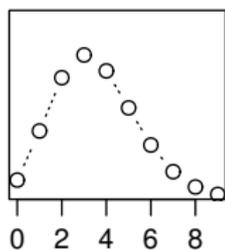
統計モデルとデータの対応づけ

統計学における推定

(人間には見えない)
真の統計モデル
 $\lambda = 3.5$ のポアソン分布



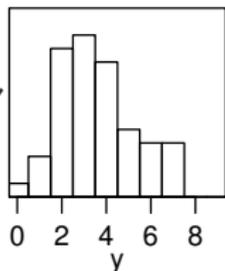
データをサンプル



観測データから
推定された
 $\hat{\lambda} = 3.56$ のポアソン分布

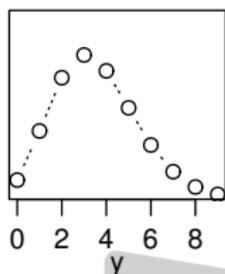
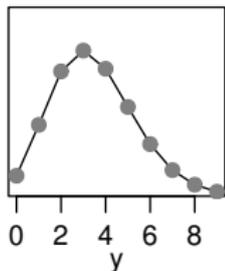
パラメータ推定

観測されたデータ



統計学における予測

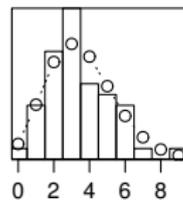
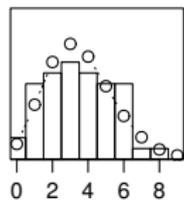
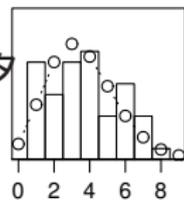
(人間には見えない)
真の統計モデル
 $\lambda = 3.5$ のポアソン分布



観測データから
推定された
 $\hat{\lambda} = 3.56$ のポアソン分布

予測: 新しいデータに
あてはまるのか?

新しいデータ
をサンプル



...

同じ調査方法で得られた新データ

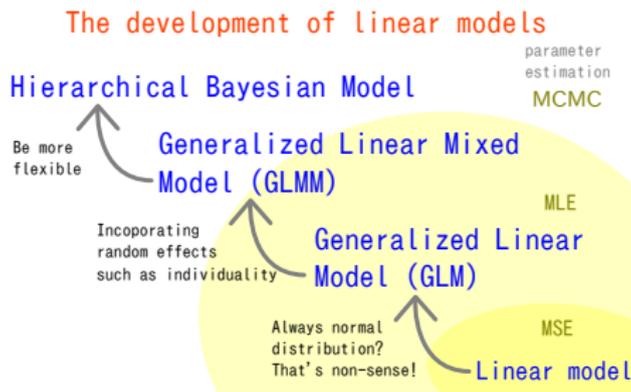
この授業で登場する確率分布

- **ポアソン分布**: $y \in \{0, 1, 2, 3, \dots\}$ となるデータ, 「 y 回なにかがおこった」
- **二項分布**: $y \in \{0, 1, 2, \dots, N\}$ 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



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)
-

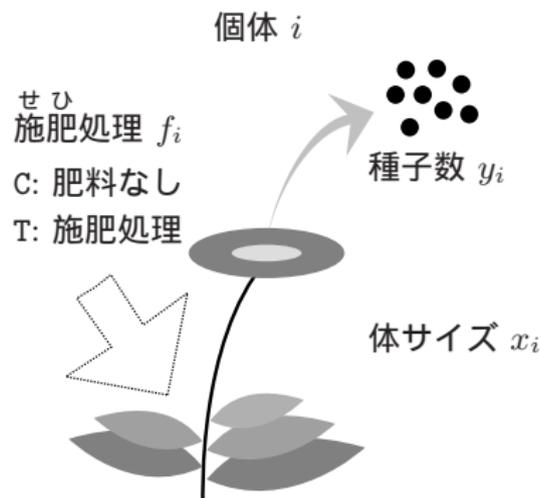
11. ポアソン回帰の例題: 架空植物の種子数データ

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知る例題

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

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: d\$x, d\$y

```
> 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: `d$f` — factor type!

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
 [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
 [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
 [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
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"
```

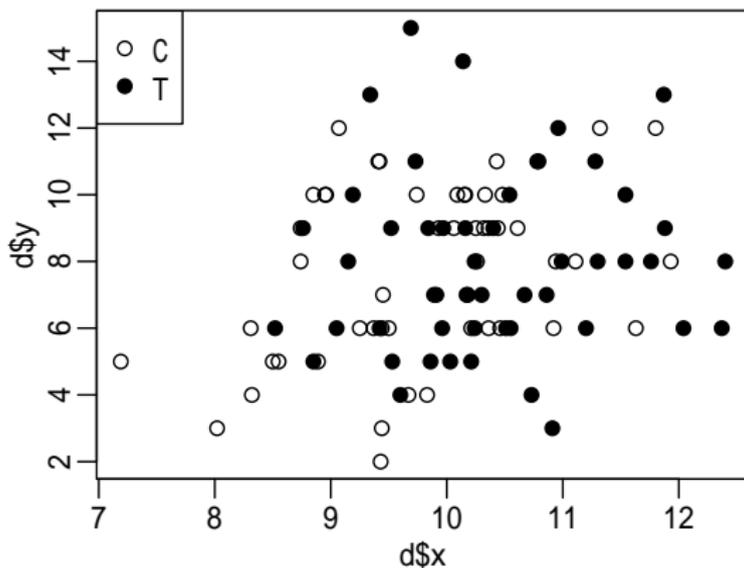
data frame の summary()

```
> 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	

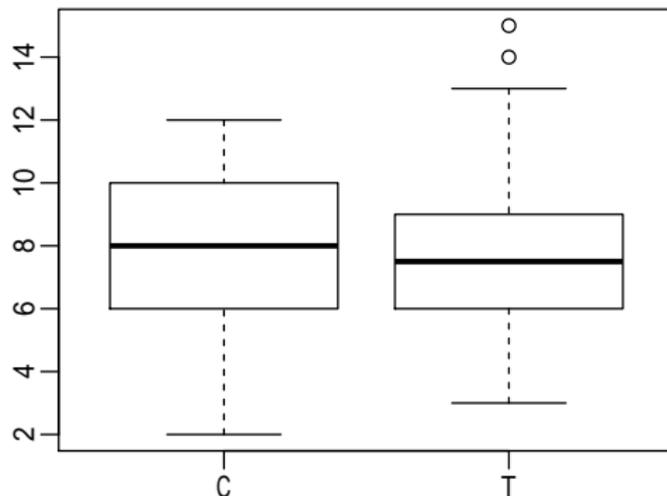
データはとにかく図示する!

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



施肥処理 f を横軸とした図

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



12. 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

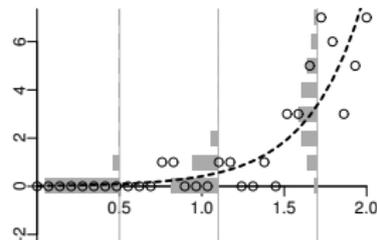
一般化線形モデル (GLM)

- 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 **ポアソン回帰** モデルを指定する

ポアソン回帰のモデル

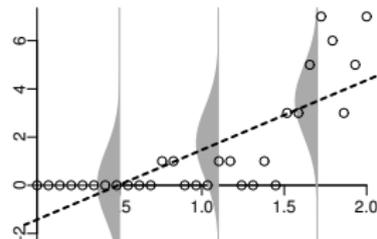
- 確率分布: **ポアソン分布**
- 線形予測子: e.g., $\beta_1 + \beta_2 x_i$
- リンク関数: **対数リンク関数**



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

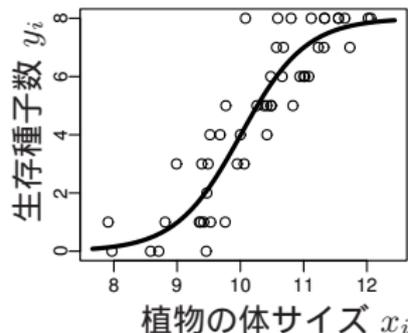
- 確率分布: 正規分布
- 線形予測子: e.g., $\beta_1 + \beta_2 x_i$
- リンク関数: 恒等リンク関数



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



「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):

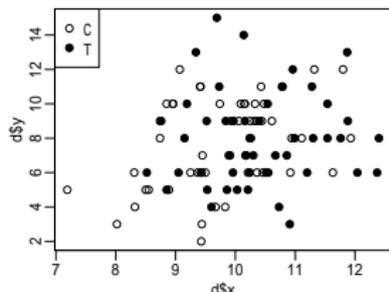
$$\begin{aligned} (\text{応答変数の平均}) &= \text{定数 (切片)} \\ &+ (\text{係数 1}) \times (\text{説明変数 1}) \\ &+ (\text{係数 2}) \times (\text{説明変数 2}) \\ &+ (\text{係数 3}) \times (\text{説明変数 3}) \\ &+ \dots \end{aligned}$$

R で一般化線形モデルを

	確率分布	乱数発生	GLM あてはめ
(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> in <code>library(MASS)</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 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
- 今日はポアソン回帰を使った GLM だけ紹介します

さてさて、この例題にもどって



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

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

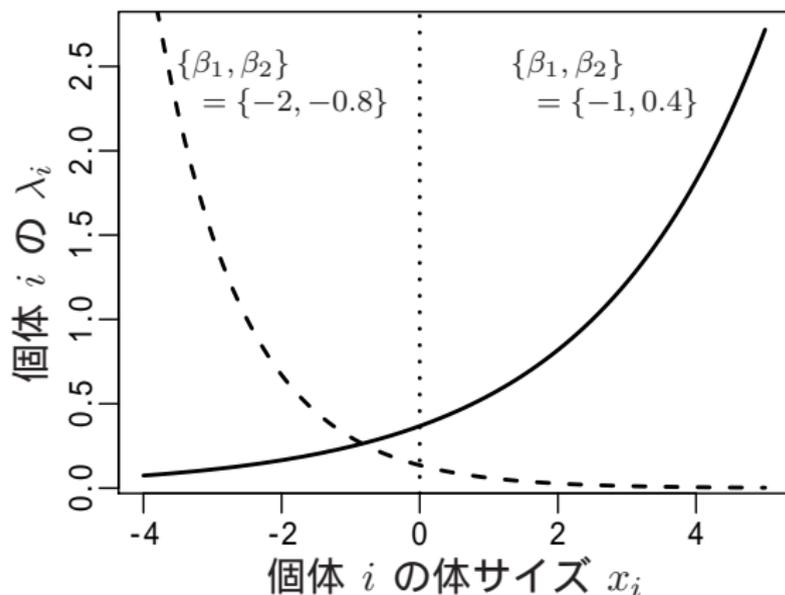
個体 i の平均 λ_i を以下のようにおいてみたらどうだろう.....?

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

- 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

個体 i の平均 λ_i

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



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

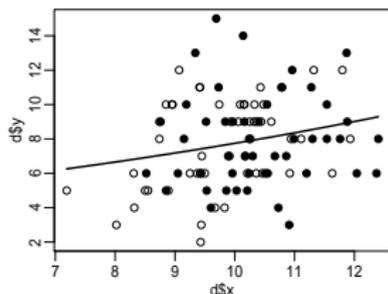
$$\log(\text{平均}) = \text{線形予測子}$$

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**



8. 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

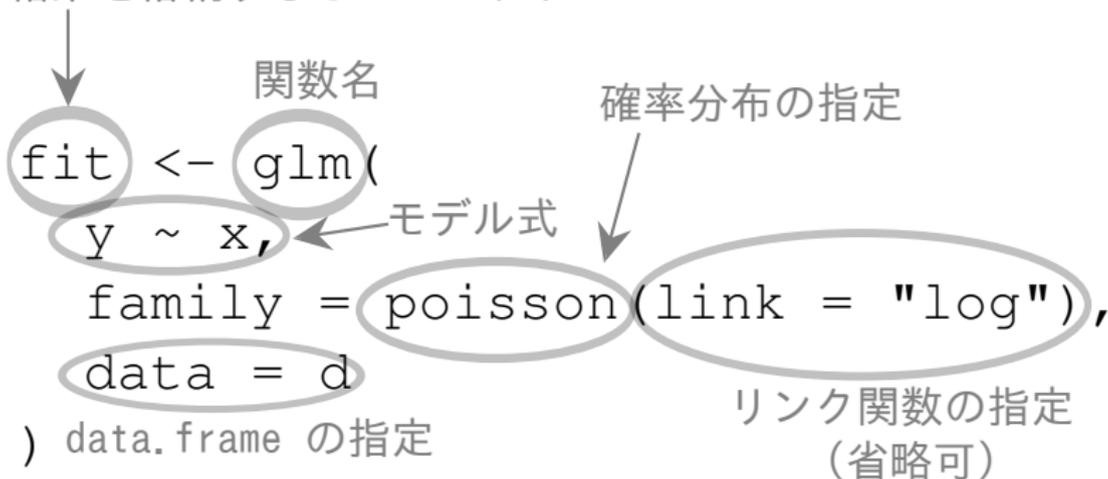
```
> d
      y      x f
1     6  8.31 C
2     6  9.44 C
3     6  9.50 C
... (中略) ...
99    7 10.86 T
100   9  9.97 T
```

これだけ!

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

glm() 関数の指定の意味

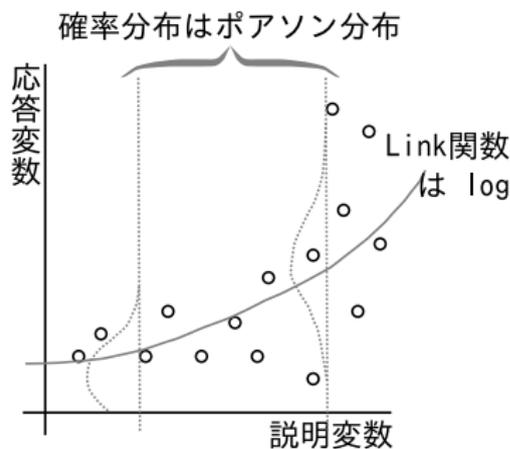
結果を格納するオブジェクト



- 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() 関数の指定を再確認

- family: poisson, ポアソン分布
- link 関数: "log"
- モデル式 (線形予測子 z): た
たとえば $y \sim x$ と指定したと
する
 - 線形予測子 $z = \beta_1 + \beta_2 x$
 β_1, β_2 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 $\lambda = \exp(z) = \exp(\beta_1 + \beta_2 x)$
 - 応答変数 は平均 λ のポアソン分布に従う: $y \sim \text{Pois}(\lambda)$



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: 89.5

Residual Deviance: 85 AIC: 475

glm() 関数のくわしい出力

```
> summary(fit)
```

```
Call:
```

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

```
Deviance Residuals:
```

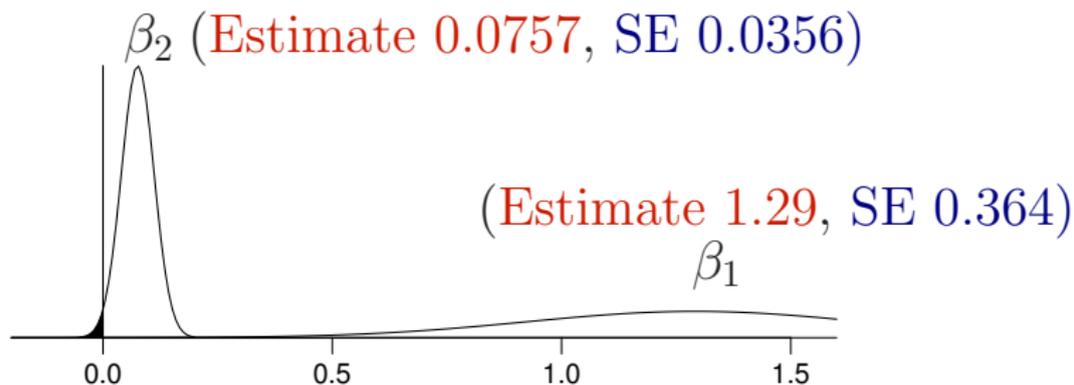
Min	1Q	Median	3Q	Max
-2.368	-0.735	-0.177	0.699	2.376

```
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



この図の要点:

- 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

モデルの予測

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

```
...
```

```
Coefficients:
```

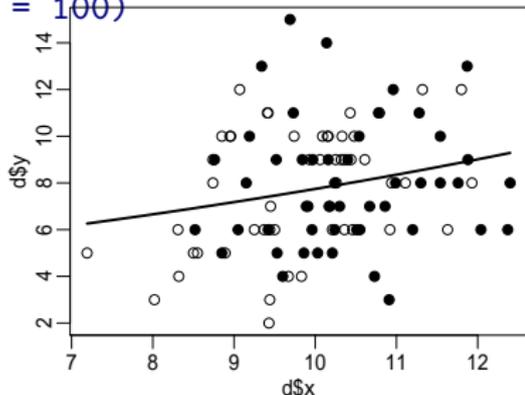
```
(Intercept)          x  
    1.2917         0.0757
```

```
> 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

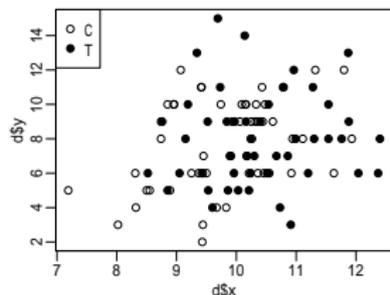


9. 処理をした・しなかった 効果も統計モデルに入れる

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

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

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

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

glm(y ~ x + f, ...) の出力

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

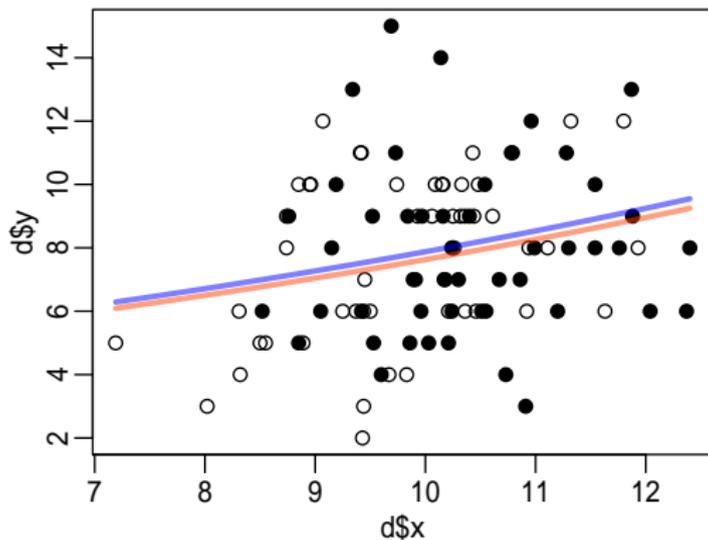
Coefficients:

	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)

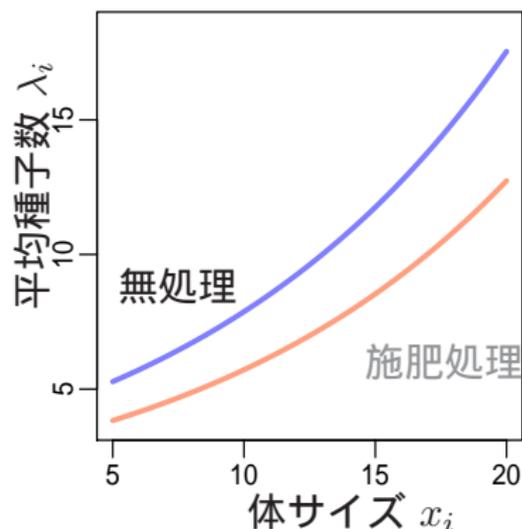
x + f モデルの予測

```
> 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

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

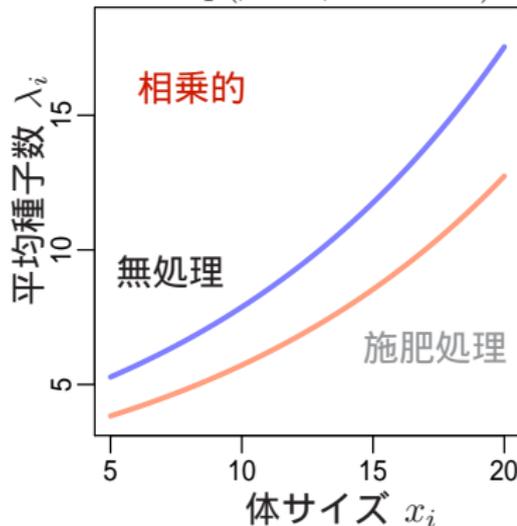


施肥効果である $\exp(-0.032)$ は
かけ算できくことに注意!

リンク関数が違うとモデルの解釈が異なる

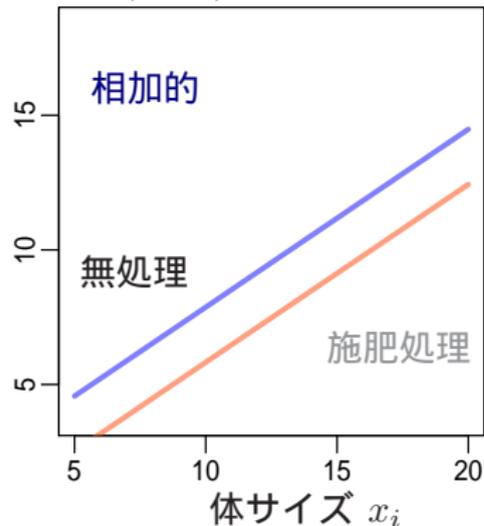
(A) 対数リンク関数

$$\lambda = \exp(\beta_1 + \beta_2 x + \dots)$$



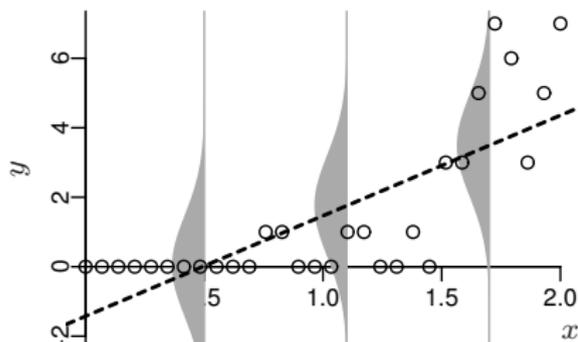
(B) 恒等リンク関数

$$\lambda = \beta_1 + \beta_2 x + \dots$$

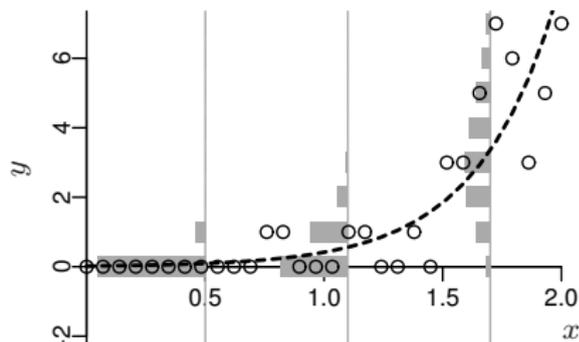


GLM: 適切な確率分布 とリンク関数を選ぶ

正規分布・恒等リンク関数の統計モデル

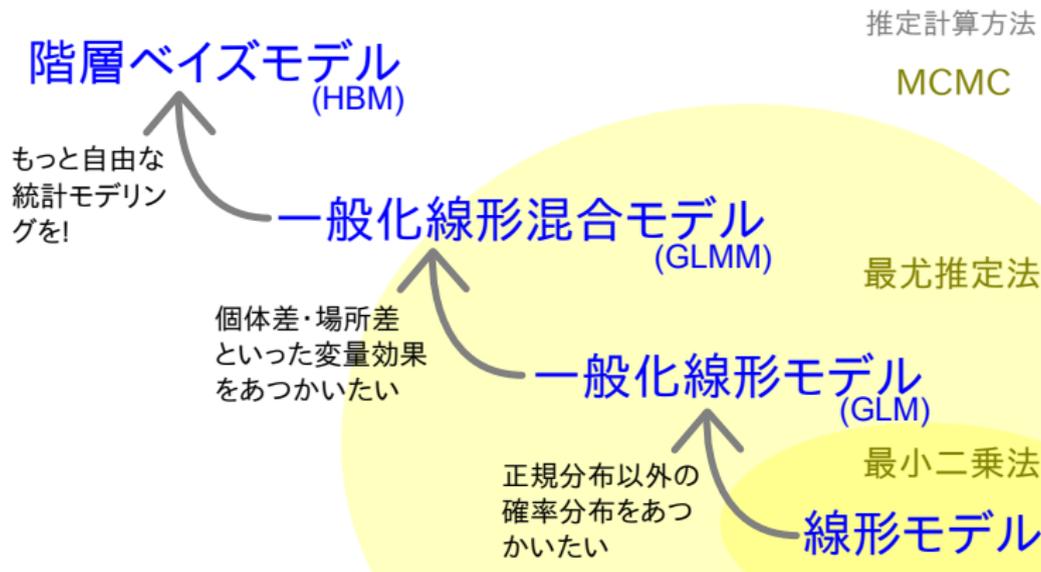


ポアソン分布・log リンク関数の統計モデル



この授業であつかう統計モデルたち

線形モデルの発展



データの特徴にあわせて線形モデルを改良・発展させる

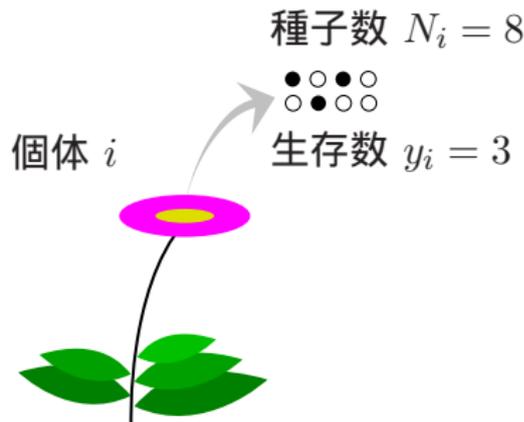
10. “ N 個のうち k 個が生きてる” タイプのデータ

上限のあるカウントデータ

ポアソン分布ではなく二項分布で

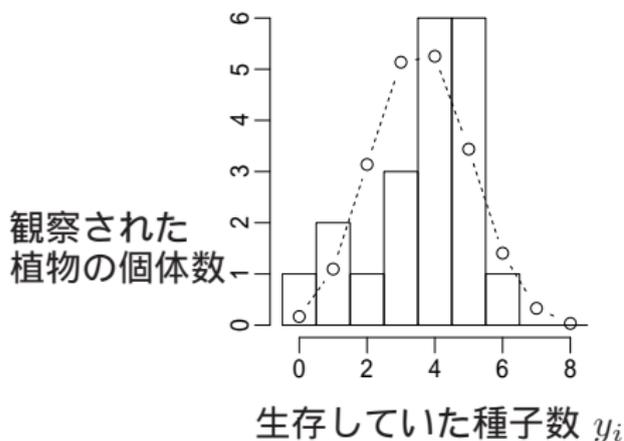
例題: 植物の種子の生存確率

- 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



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

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

- 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**

ゆうど

尤度: 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

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

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 $L(q \mid \text{データ})$ 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

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

- 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

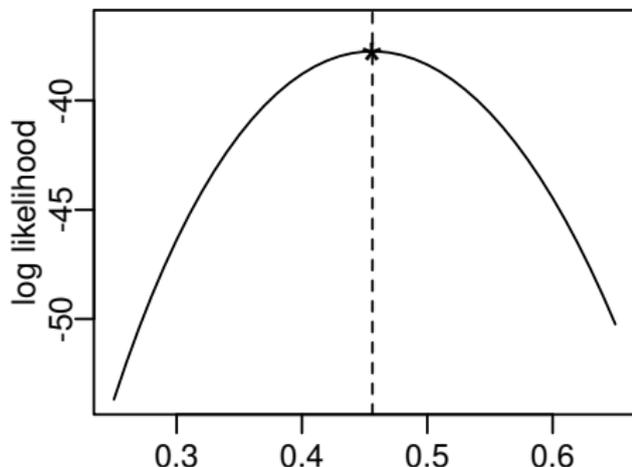
- 対数尤度 $L(q \mid \text{データ})$ が最大になるパラメーター q の値をさがしだすこと

- 対数尤度 $\log L(q \mid \text{データ})$ を q で偏微分して 0 となる \hat{q} が対数尤度最大

$$\partial \log L(q \mid \text{データ}) / \partial q = 0$$

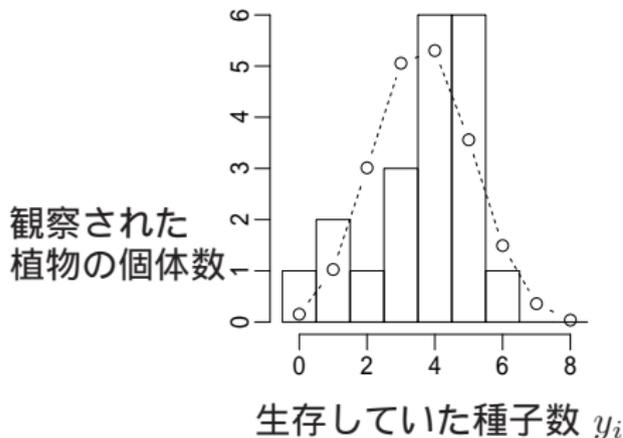
- 生存確率 q が全個体共通の場合の最尤推定量・最尤推定値は

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



二項分布で説明できる 8 種子中 y_i 個の生存

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



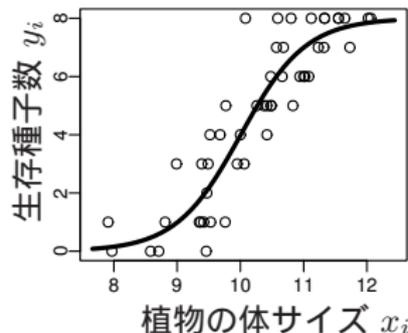
11. “ N 個のうち k 個が生きてる” タイプのデータ

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

GLM のひとつである **logistic 回帰モデル**を指定する

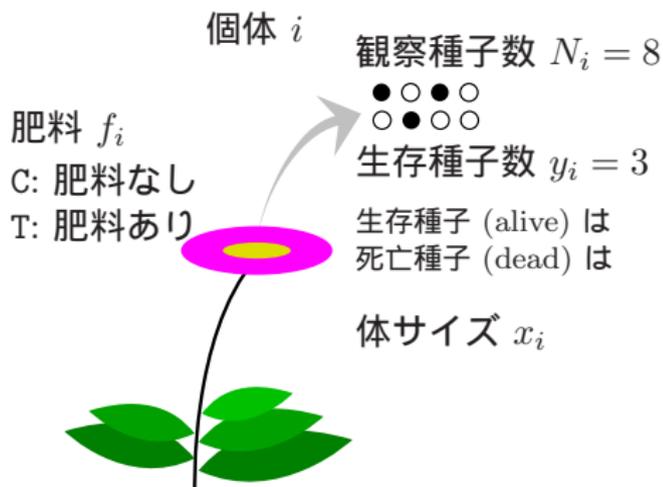
ロジスティック回帰のモデル

- 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



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

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/2014/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

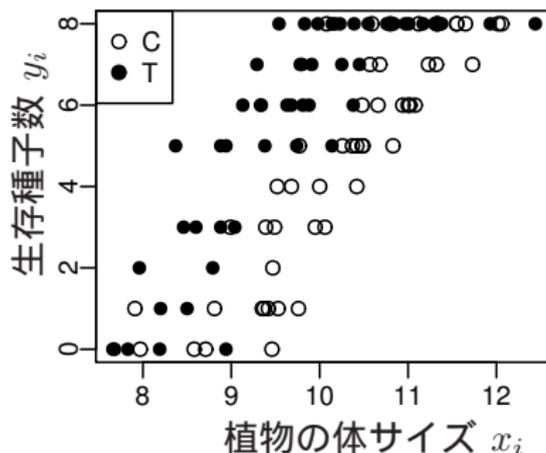
```
> summary(d)
```

	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?

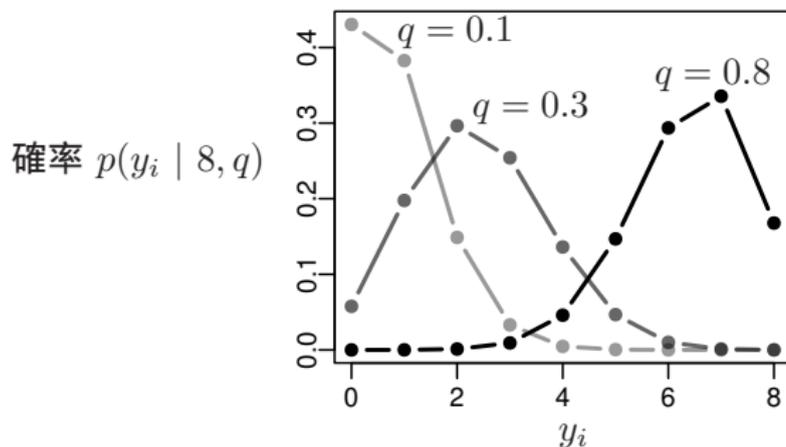
12. ロジスティック回帰の部品

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

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

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

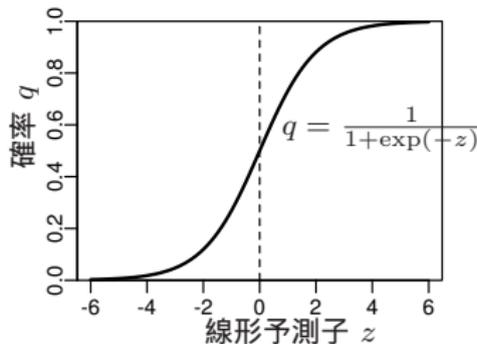


ロジスティック曲線とはこういうもの

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

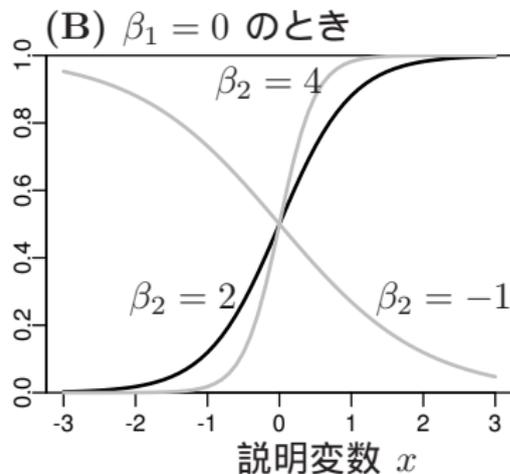
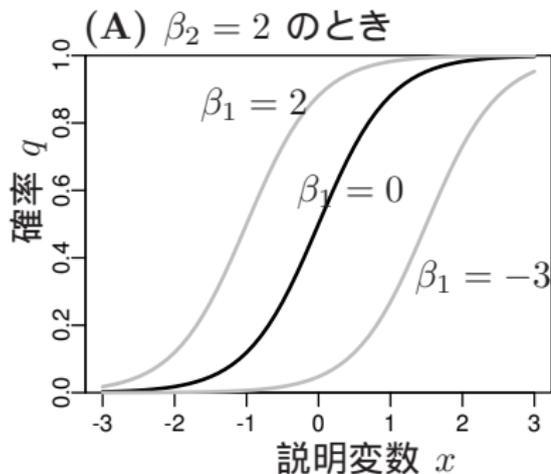
$$q_i = \text{logistic}(z_i) = \frac{1}{1 + \exp(-z_i)}$$

```
> 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 $0 \leq q \leq 1$
 となる便利な関数

logit link function

- logistic 関数

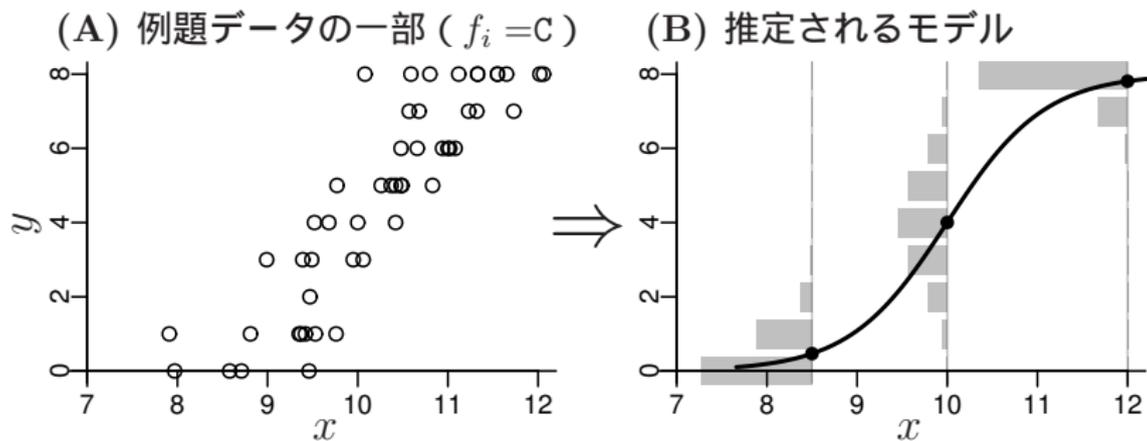
$$q = \frac{1}{1 + \exp(-(\beta_1 + \beta_2 x))} = \text{logistic}(\beta_1 + \beta_2 x)$$

- logit 変換

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

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

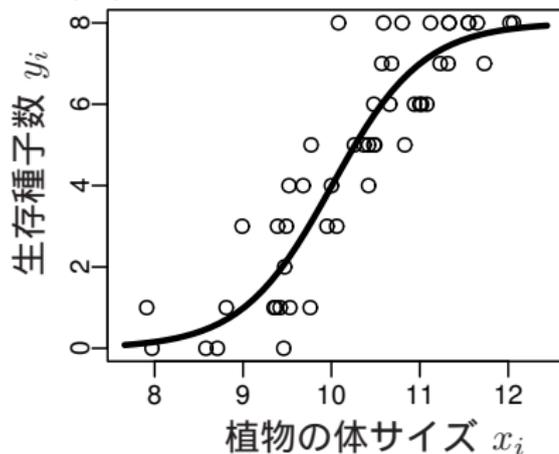
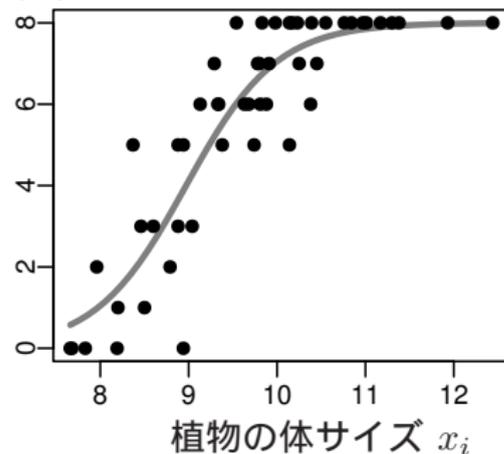
```
> glm(cbind(y, N - y) ~ x + f, data = d, family = binomial)
```

```
...
```

```
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

(A) 施肥処理なし ($f_i = C$)(B) 施肥処理あり ($f_i = T$)

この講義の流れ: 例題を考えながら理解する

1. 統計モデル・確率分布・最尤推定
2. ポアソン分布の一般化線形モデル (GLM)
3. 二項分布の GLM
4. MCMC と階層ベイズモデル

単純化した例題にそって統計モデルを説明