データ解析のための統計モデリング (2008 年 10-11 月)

全 5 (+2) 回中の第 6 回 (2008–11–26)

一般化線形混合モデル (GLMM)

久保拓弥 kubo@ees.hokudai.ac.jp

http://hosho.ees.hokudai.ac.jp/~kubo/ce/EesLecture2008.html

この講義のーとが「データ解析のための統計モデリング入門」として出版されました!

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

まちがいを修正し，詳しい解説・新しい内容をたくさん追加したものです

今日のもくじ

「生態学の統計モデリング」，今年度は全 5 回で終了させられるハズだったのですが，熱心なる参加者の皆さんにアオられてしまったので GLMM 補講を開催する次第です．一般化線形混合モデル (GLMM) は，現実のデータ解析で考慮しなければならない**個体差・場所差の効果**をうまく表現できる統計モデルです．

さて，ここまでの講義では「一般化線形モデル (GLM) ってけっこう便利なんじゃないの?」という点を強調するようなハナシですすめてきました．つまり観測データが 0 個, 1 個，2 個，$\cdots$ とカウントできるもので (カウントデータ)，その値の上限がはっきりしないのであればポアソン回帰

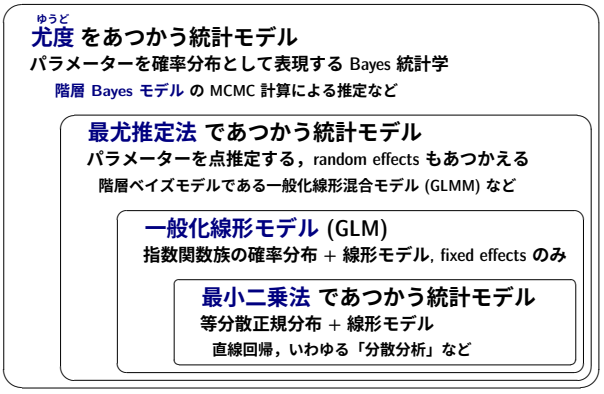
- 統計モデルの確率分布は**ポアソン分布**
- link 関数は **log link 関数** を使うのが便利

また観測される値の上限が N 個であるなら,

- 統計モデルの確率分布は**二項分布**
- link 関数は **logit link 関数** を使うのが便利

そして要因をくみあわせた線形予測子を設定します．このように統計モデルを構築して，パラメーターを最尤推定，モデル選択などモデルの評価にすすむ，とうい手順でした．

今回は R の `glm()` だけではうまくパラメーターが推定できない状況，一般化線形モデル (GLM) ではうまく表現できない現象で威力を発揮する一般化線形混合モデル (generalized linear mixed model; GLMM) について説明したいと思います．



1. 架空データの例題を作りながら GLM

はい，今回の架空植物のデータは第 4 回の講義のロジスティック回帰の紹介で導入した結実確率を推定できるようなものです．[1] ここではこの架空植物の個体ごとの (種子数ではなく) **結実確率**がどのように決まるか (統計モデルでどう表現するのがよいのか) をあつかいたいとします．個体は i という記号であらわされ ($i = 1, 2, 3, \cdots, 100$, つまり 100 個体います)，その胚珠[2] 数は 8 個 (全個体共通)，結実した[3] 種子数は y_i とします．胚珠数が $N_i = 8$ 個なので全部結実した場合には種子数 $y_i = 8$ 個となり，これが最大種子数，最小種子数はもちろん全胚珠が結実に失敗して種子数ゼロ個の場合です．つまり，$y_i \in \{0, 1, 2, 3, \cdots, 8\}$ ということです．

またこの架空植物は 2 枚から 6 枚の葉っぱをもち，個体 i ごとに異なる葉数 x_i をもつとします．

1. 今回は「肥料をやる効果」は例題に含まれていません．また今回は「サイズ」ではなく，(離散値である) 葉数 x_i を説明変数としています．

2. 種子のモトになる植物の器管，と考えてください．

3. つまりちゃんと種子のカタチになること．

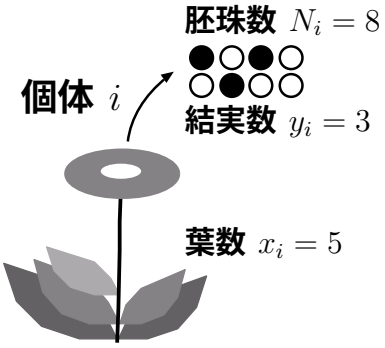


図 1: 架空植物の第 i 番目の個体 ($i = 1, 2, \cdots, 100$)

第 4 回と同じく，**結実確率**の定義は「ある胚珠が種子になる確率」です．この結実確率は個体 i の中では共通の値 q_i をとると仮定します．そしてこの例題で**調べたいこと**は「ある個体の結実確率 q_i は葉数 x_i が大きいほど高いという性質がある」というときに，どういう統計モデリング・推定の方法でそれを解明できるか，というものです．

さてさて……今日は**例題用の架空データを作りながら**，ハナシをすすめてみましょう． [4]

4. データを生成する統計モデル，そしてデータを解析する統計モデルの対応関係がどうなっているのか，といったことも考えながらススめていきましょう．

まずは R を起動，例によって講義 web page に置いてある `f.R` という R ソースコードファイルをダウンロードして，自分の計算機のてきとうな場所におき

```
> source("f.R")
```

というふうに読みこんでください．この処理によって以下のようなデータオブジェクトと関数と定義が R に記憶されます:

- `d`: 架空植物のデータを格納する `data.frame`，これは上のように `source(f.R)` を実行すると自動的に作られます
- `logistic()`: logistic 関数の定義
- `plot.d()`: `d` を作図する関数 [5]

5. 今回は横軸・タテ軸の変数がどちらも整数なので，`plot()` に多少の工夫が必要なのです．

これらについては今から説明してみます．まずはデータオブジェクト `d` の構造を調べてみましょう．

```
> head(d)
  N x re y
1 8 2  0 2
2 8 2  0 0
3 8 2  0 1
4 8 2  0 1
5 8 2  0 1
6 8 2  0 0
```

第 4 回と同じく，N: 胚珠数，y: 結実種子数，x: 葉数という構造になっています．[6] re の列はあとで使いますので，現時点では無視してください．

6. 今回は施肥処理 f はナシです．

y のような列を R の二項乱数 [7] 関数を使って生成するにはどうしたらよいのでしょうか? たとえば結実確率をこのように

7. 二項分布から生成される乱数のことです．

$$q_i = \frac{1}{1 + \exp(-(\beta_1 + \beta_2 x_i))}$$

ロジスティック関数で定義します．この関数は先ほどよみこんだ中ですでに logistic() として定義されていて，

```
> logistic
function(z) 1 / (1 + exp(-z))
```

これを利用して $\beta_1 = -4$ で $\beta_2 = 1$ での結実確率 q_i を計算して，結実種子数のデータを「発生」させるには，

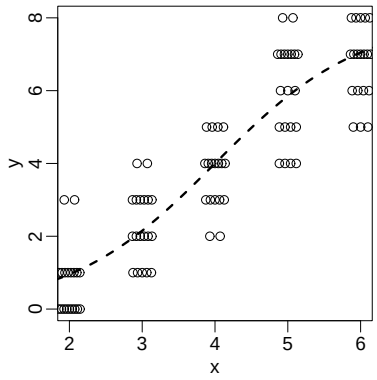
```
> d$y <- rbinom(100, 8, prob = logistic(-4 + 1 * d$x))
```

とすればよいです．summary(d) してみましょう．

```
> summary(d)
       N           x           re          y
 Min.   :8   Min.   :2   Min.   :0   Min.   :0
 1st Qu.:8   1st Qu.:3   1st Qu.:0   1st Qu.:2
 Median :8   Median :4   Median :0   Median :4
 Mean   :8   Mean   :4   Mean   :0   Mean   :4
 3rd Qu.:8   3rd Qu.:5   3rd Qu.:0   3rd Qu.:6
 Max.   :8   Max.   :6   Max.   :0   Max.   :8
```

図であらわすと以下のようになります．作図には，先ほどの f.R の中で定義されていた plot.d() 関数を使っています．

```
> plot.d(d)
> xx <- seq(1, 7, 0.1)
> lines(xx, logistic(-4 + xx) * 8, lty = 2) # 破線
```



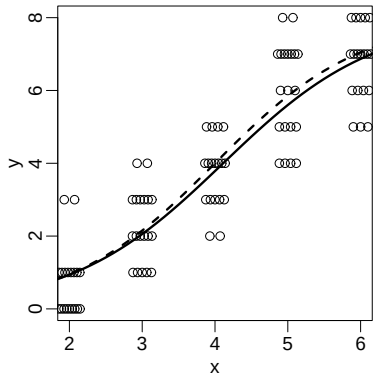
さて，このような観測データは glm(..., family = binomial) でうまくパラメーター推定できそうですね．[8] いつものごとく glm() 関数をよびだして推定計算させてみると，

8. うまく推定できれば図中の破線のような曲線になるはずです．

```
> fitA <- glm(cbind(y, N - y) ~ x, data = d, family = binomial)
> beta <- fitA$coefficients
> beta
(Intercept)           x
  -3.925843    0.954703
```

「ホントの値」は $\beta_1 = -4$ かつ $\beta_2 = 1$ だったのでなかなかうまく推定できているようです．ついでに図に重ねて示してみましょう．

```
> lines(xx, logistic(beta[1] + beta[2] * xx) * 8)
```



ここまでは何の問題もなく GLM の一部であるロジスティック回帰でうまく観測されたパターンを説明できました．

2. 種子結実データなのに GLM がうまくいかない状況がある?

次に $i = \{1, \cdots, 100\}$ の 100 個体の架空植物にちょっとした「個体差」があるとしましょう.

$$q_i = \frac{1}{1 + \exp(-(\beta_1 + \beta_2 x_i + r_i))}$$

このように結実確率に個体ごとに異なる値 r_i が入っているとします. この「個体差」をあらわす r_i は集団全体の**平均はゼロ**だけど**ある幅をもった** 正規分布にしたがっている, としましょう.

この「個体差」は何なのでしょうか? もしこれが現実のデータなら, その生物学的な理由としては個体ごとに遺伝子が違う, 年齢がちがう, はえてる場所の栄養塩類や水分環境・光環境が異なる, などなどいろいろな要因によって**個体差** や局所的な **場所差**と考えればよいでしょう.

重要なのは, 観測者であるわれわれはこの架空植物の**すべてを知ることはどうやっても不可能**であり,「個体差」は**原因不明**のままあつかわないといけない, ということです. [9] 現実の観測データを解析していても,「原因不明な個体差・場所差みたいなばらつき」にはよく遭遇します. この例題はそういう状況をシミュレイトしているわけです.

さて, データオブジェクト d の re 列に「個体差」の正規乱数を上がきしてやり, さらに re 列を使って「個体差あり」結実種子数を二項乱数で生成してやりましょう. [10]

```
> d$re <- rnorm(100, 0, 3) # 「個体差」の平均ゼロ, SD は 3
> d$y <- rbinom(100, 8, prob = logistic(-4 + d$x + d$re))
> # d$y は 8 個の胚珠の中の結実種子数
```

先ほどの例と同じく $\beta_1 = -4$ で $\beta_2 = 1$ という値を使っています. 個体差 r_i のせいで結実数 y_i のばらつきは増えたはずですが, **平均ゼロ** の「個体差」なので

```
> d$id <- 1:nrow(d) # ついでに個体番号 i も追加
> summary(d)
       N           x           re                y             id
 Min.   :8   Min.   :2   Min.   :-7.3633   Min.   :0.00   Min.   :  1.00
 1st Qu.:8   1st Qu.:3   1st Qu.:-2.2545   1st Qu.:1.00   1st Qu.: 25.75
 Median :8   Median :4   Median :-0.1616   Median :3.00   Median : 50.50
 Mean   :8   Mean   :4   Mean   :-0.1880   Mean   :3.81   Mean   : 50.50
 3rd Qu.:8   3rd Qu.:5   3rd Qu.: 1.4621   3rd Qu.:7.00   3rd Qu.: 75.25
 Max.   :8   Max.   :6   Max.   : 7.1625   Max.   :8.00   Max.   :100.00
```

「平均」種子数はあいかわらず 4 前後の値となっています.

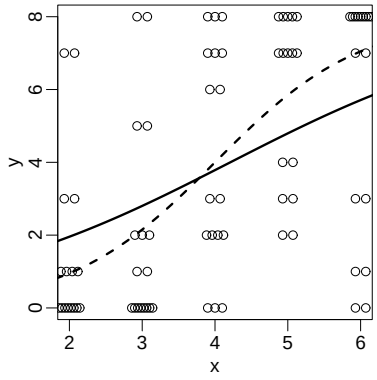
9. 第一回で話したように, 人間の観測・調査・測定・実験・認識能力には限界があり, あるていど複雑な現象の要因を還元主義的に分離・分割することは不可能, ということです.

10. ここで乱数関数を使っていますので, d$re や d$y の値は乱数を発生させるごとに異なったものになります.

それでは先ほどと同じく，glm() によるパラメーター推定を……

```
> fitB <- glm(cbind(y, N - y) ~ x, data = d, family = binomial)
> (beta <- fitB$coefficients)
(Intercept)           x
    -2.1487       0.5104

> # (途中略: plot.d(d) などしてから)
> lines(xx, logistic(beta[1] + beta[2] * xx) * 8) # 実線が予測
```



今回は**うまくいってない**ようですね．観測データを作った統計モデルでは $\beta_1 = -4$ で $\beta_2 = 1$ だったのに，推定値は $\hat{\beta}_1 = -2.15$ で $\hat{\beta}_2 = 0.51$ となっています．そしてこれは「たまたま」こうなったのではなく，何回やりなおしても「傾きがゆるい」つまり β_2 が過小推定されるような結果になります．

3. 「個体差」と過分散

どうやら,「個体差」なるモノがまじってるときには，ごく単純に glm(y ~ x, family = binomial, ...) と推定してはいけない，ということです．その理由は何でしょうか?

ひとことで言えば，正規乱数である「個体差」 r_i が結実確率 q_i のばらつきを増やし，その結果として結実数の分布が**二項分布では表現できなくなった**ためです．このように「あり・なし」データが二項分布で表現できない[11] ことを**過分散** (overdispersion) とよびます．これは二項分布ではありえないほどデータ (たとえば結実数) のばらつきが増大することです．

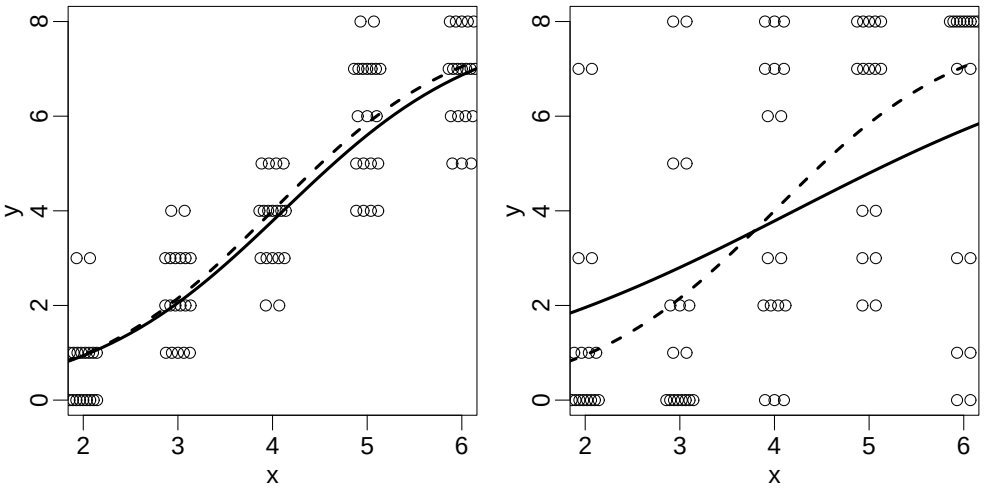
11. あるいは「個数データ」がポアソン分布で表現できない

逆にありえないほど分散が小さくなることを **underdispersion** といいますが，underdispersion になっているデータにはめったに遭遇することはありません．私たちが観測データでよく見るのは overdispersion のほうなので，ここでは統計モデリングによってこの overdispersion をうまくあつかう方法

を検討します. [12]

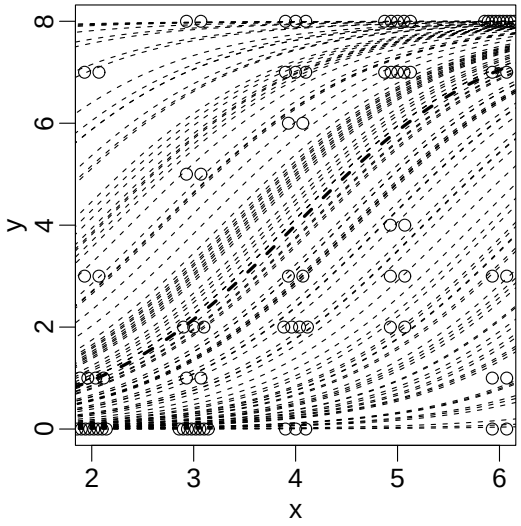
12. もちろん underdispersion も統計モデリングで解決できる問題です.

まずは過分散の実態をもう少し調べてみましょう. GLM がうまくいった場合・うまくいかなかった場合の図をこのように並べて



比較してみると,「GLM がうまくいってない」ほう (右) は「個体差」 r_i のせいで結実数のばらつきが大きくなっていることがわかります. これは「葉数 x_i と結実確率 q_i の関係」が個体ごとに異なっていて,

```
> plot.d(d, col = "black")
> for (re in d$re) lines(xx, logistic(-4 + xx + re) * 8, lty = 2)
> lines(xx, logistic(-4 + xx) * 8, lty = 2)
```



これが観測データを作りだしているからです.

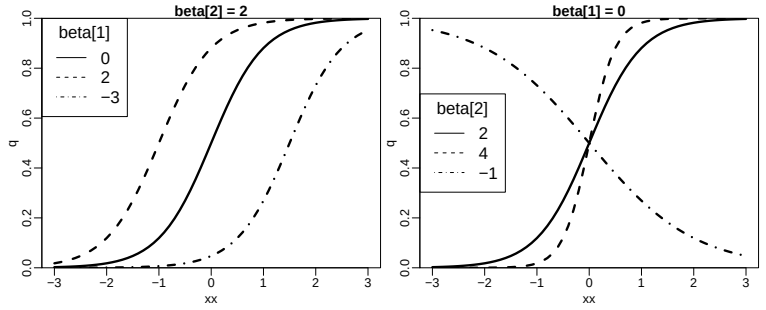
ここでロジスティック関数の性質を復習してみましょう. ロジスティック

関数を

$$q_i = \frac{1}{1+\exp(-(\beta_1+\beta_2 x_i))}$$

と定義します．[13] このときに係数 (パラメーター) である β_1 や β_2 に依存して曲線のカタチは下の図のように変わりました．

13. これは先ほどのモデルから「個体差」r_i を除いた式です．



つまり「切片」みたいなパラメーターである β_1 を変えると曲線の「位置」が左右に動く (ある x_i における確率が上下する) といった関係がありましたね．

さて，今回のモデルをよくみなおしてみると，

$$q_i = \frac{1}{1+\exp(-(\beta_1+\beta_2 x_i+r_i))}$$

「個体差」r_i はあたかも β_1 を上下しているかのように見えます．これによって (「傾き」 β_2 は共通だけど) 個体ごとに異なるロジスティック曲線が生成されました．

「個体差」 r_i によって二項分布が「乱されて」しまうとどうなるのでしょうか? このあたりをもう少し定量的に調べるために，全データ (100 個体ぶん) d から葉数 4 の 20 個体を抽出してきて，これを d4 という data.frame に格納します．

```
> d4 <- d[d$x == 4,]
> head(d4)
   N x         re y id
41 8 4  2.0694010 6 41
42 8 4 -1.7489004 2 42
43 8 4 -4.6745712 0 43
44 8 4 -0.5496066 5 44
45 8 4  0.6166827 6 45
46 8 4 -0.7773915 2 46
> sum(d4$N)
[1] 160
> sum(d4$y)
[1] 81
> sum(d4$y) / sum(d4$N)
[1] 0.50625
```

このように標本から推定された結実確率の平均は 0.51 ぐらいとなり，これはこのデータを生成したモデルで与えた結実確率[14] とだいたい同じです．

最大種子数 が 8 の二項分布の平均は $8q_i = 8 \times 0.5 = 4$ となり，また二項分布は「平均が決まれば自動的にばらつきも決まる」確率分布[15] なのでその分散は $8q_i(1 - q_i) = 2$ となるはずです．しかしながら，

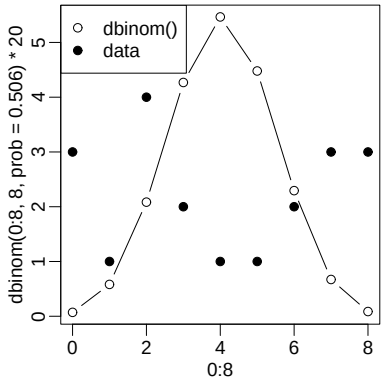
```
> mean(d4$y)
[1] 4.05
> var(d4$y)
[1] 8.36579
```

平均は二項分布の予測どおりですが，分散が予想の 4 倍になっています．葉数が $x_i = 4$ という条件で,「結実数が y_i 個であった個体」を R にカウントさせてみると，

```
> summary(as.factor(d4$y)) # x = 4 である 20 個体の種子数
0 1 2 3 4 5 6 7 8
3 1 4 2 1 1 2 3 3
```

「平均結実確率」は 0.5 なのに,「一個も結実しなかった」「全胚珠 8 個が結実した」個体がどちらも 20 個体中 3 個体いた，とわかります．図にするとこのようになります．

```
> plot(0:8, dbinom(0:8, 8, prob = 0.506) * 20, type = "b")
> points(0:8, summary(as.factor(d4$y)), pch = "X")
```



二項分布 (dbinom()) からデータが大きくずれていますね．

このような例からわかるように,「個体差」が**極端**に大きい場合,「葉数 4 のときの結実確率の集団平均は 0.5」であったとしても，下に示している図 2 のような観測データが得られるでしょう: 標本個体の半数で結実数がゼロ，残り半数で全胚珠が結実となっていて，たしかに結実確率の集団平均 (= 集

14. 葉数 $x_i = 4$ のときに結実確率 q_i の平均は 0.5.

15. ポアソン分布もまた同じような性質があり，平均が λ のポアソン分布の分散は λ です．

団内の結実数 / 集団内の全胚珠数) は 0.5 で個体ごとの平均結実数は 4 になっている……しかし実際には結実数 4 の個体はひとつもいない，といったデータです.

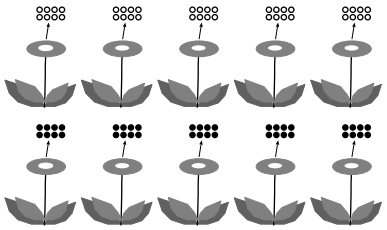


図 2: 極端な過分散の集団: 葉数 $x_i = 4$，平均結実数は 4，しかし……

二項分布やポアソン分布のように**「平均が決まると自動的に分散が決まる」**つまり 1 パラメーター確率分布を使っているときにだけ過分散の有無を言えます．正規分布や負の二項分布といった平均とは独立に分散も自由に指定できる確率分布を使った統計モデルを想定している場合には過分散について議論できません．[16]

ついでに，ここでもちいている「個体差」という用語がかなり限定された意味，つまり「線形予測子 $\beta_1 + \beta_2 x_i$ に r_i が追加されている」[17] であることに注意しましょう．葉数が個体ごとに異なることは「個体差」とは呼びません．この区別はどのようになされているのか? 統計モデルの中でのあつかいをみると，葉数は観測者によって定量化されている「説明変数」であり，いっぽうで r_i のばらつきは直接には観測されていない (できない) ので「個体差」と呼ぶ，としておきましょう.

ここまでまとめてみると，

- 現実の観測データには原因不明な**「個体差」**みたいなものがあるだろう
- 「個体差」は**過分散**の原因となる— つまり結実数データは二項分布にあてはまらなくなる
- データが二項分布から逸脱していると，GLM (ロジスティック回帰など) でうまくパラメーター推定できない

さあ，どうしたらよいでしょう，という場面です.

16. しかしながら，これらの測定誤差が既知であるときも過分散が言えるかもしれません.

17. そして r_i というばらつきが生じる原因は不明である，と.

4. 一般化線形混合モデル (GLMM)

このような「個体差」によって過分散が生じているデータにみられるパターンをうまく説明できそうな統計モデルが一般化線形混合モデル (GLMM) です.

GLM 二項分布のロジスティックモデルの復習から始めていって，GLMM の二項分布のロジスティックモデル (GLM) を導入してみましょう．ある個体 i で観測された結実数が y_i である確率は各胚珠独立の二項分布にしたがうと仮定しているのだから

$$f(y_i \mid \beta_1, \beta_2) = \frac{8!}{y_i!(8-y_i)!} q_i^{y_i} (1-q_i)^{8-y_i}$$

となり，結実確率 q_i は logit link 関数

$$\mathrm{logit}(q_i) = \beta_1 + \beta_2 x_i$$

で線形予測子 $\beta_1 + \beta_2 x_i$ と関連づけられているので，この確率は logistic 関数

$$q_i = \frac{1}{1 + \exp(-(\beta_1 + \beta_2 x_i))}$$

で表現されていることになります．[18] ここまでは GLM です．そして今回の例題ではこのように

$$q_i = \frac{1}{1 + \exp(-(\beta_1 + \beta_2 x_i + r_i))}$$

「個体差」 r_i が線形予測子が追加されていました．

18. わからなかったら第 4 回の講義のーとを復習してください．

さて，ここで一般化線形「混合」モデル，GLMM (generalized linear mixed model) の **mixed** の意味の説明するために，この例にそって fixed effects と random effects という統計学用語の定義をしてみます．

結実確率 q_i の集団平均を上下しているモノは何でしょうか? パラメーター β_1 と β_2，そして葉数 x_i です．これらの値の中でパラメーターでないもの，つまり葉数 x_i は **fixed effects** (日本語では母数効果あるいは固定効果) と呼ばれています．また葉数 x_i に対応するパラメーター (係数) β_2 や「切片」β_1 は fixed effects の影響の大きさをあらわすパラメーターと分類されます．これらのパラメーターは全個体に共通しています．

これに対して個体 i の「個体差」[19] を表現しているパラメーター r_i の集団平均はゼロと定義されているので，結実確率 q_i の集団平均に無関係であり，[20] ただ個体間の結実確率のばらつきにのみ影響を与えています．このように個体ごとに異なる効果は **random effects** (変量効果またはランダム効果) です．r_i は random effects をあらわすパラメーターで，個体 i ごとに異なります．

19. 「個体差」かもしれないし，植物だから場所差かもしれないし……などと不明な要因．

20. 正確には集団平均には影響しちゃうんだけど (GLMM の場合)，結実確率の集団中央値 (median) には影響を与えない，ということです．

GLM のロジスティックモデルに r_i を追加して改良したモデルは fixed と random effects の両方を含んでいるので**混合モデル** (mixed model) と呼ばれ,「個体差」を明示的にあつかえる統計モデルになっています．なかでもロジスティックモデルなど GLM に属するモデルを混合化した場合には，一般化線形混合モデル (GLMM) と呼ばれます．

5. Random effects 考慮した最尤推定

次の問題は，この「個体差」あるいは random effects をあらわす r_i を観測データをどのように対応づければよいのだろうか? ということです．ここで推定計算に工夫が必要になります.

fixed effects をあらわすパラメーター β_1 や β_2 のように r_i の値も最尤推定してやることも不可能ではないかもしれません.

しかしながら 100 個体ぶんの結実数データ y_i を説明するために $\{\hat{r}_1, \hat{r}_2, \cdots, \hat{r}_{100}\}$ というふうに最尤推定か何かで **100 個のパラメーター r_i の値を確定**しちゃっていいものでしょうか? これって deviance の説明にでてきた FULL モデルみたいなもので，あてはまりはいいかもしれないけれど，たとえば AIC 的には最悪 [21] となりそうです.

21. パラメーター数が 100 も増えるから.

データ数が 1000 個体になったらパラメーター数も 1000 個に増やす…… 私たちがやっている統計モデリングとは，このような「あてはまりが良くなるならパラメーター数なんていくらでも増やせばいい」では**ありません**.

そこで GLMM など混合モデルでは

- fixed effects をあらわすパラメーター β_1 や β_2 は最尤推定する
- random effects あらわすパラメーター $\{r_i\}$ たちは **最尤推定しない**

このように推定計算を工夫しています．これについて説明してみましょう.

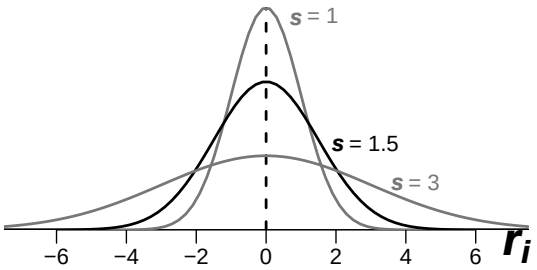


図 3: 「個体差」をあらわす平均ゼロの正規分布

まず，r_i が平均ゼロの確率分布の何か，今回の例題にあわせてたとえば正規分布で表現できると仮定します．すると「個体差」 r_i の確率分布 $g(r_i \mid s)$ は

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

と書けて，ここで新しく導入したパラメーター s は「集団内の r_i のばらつき」をあらわす標準偏差です．[22] このように random effects あらわす r_i の

22. s が大きければ「個体差」の大きい集団であり，s が小さければ「個体差」の小さい均質な集団, ということです.

確率分布を導入することによって，個体 i で観測された結実種子数が y_i であるときの**すべての可能な** r_i に関して積分すれば，

$$L_i(\beta_1, \beta_2, s \mid x_i, y_i) = \int_{-\infty}^{\infty} f(y_i \mid \beta_1, \beta_2, r_i)\ g(r_i \mid s) dr_i$$

このように個体ごとの尤度 L_i は「実際の r_i」を知らなくても計算できるようになります．集団全体の尤度は L_i の 100 個体ぶんの積として表現されるので，

$$L(\beta_1, \beta_2, s \mid \{x_i\}, \{y_i\}) = \prod_{i=1}^{100} L_i(\beta_1, \beta_2, s \mid x_i, y_i)$$

となり，これを最大化するパラメーター β_1, β_2, s の最尤推定値をもとめればよいわけです．[23] これが (GLMM の一部である) 混合ロジスティックモデルの最尤推定です．

ここで説明した (ちょっと工夫した) 最尤推定法について，しつこく復習してみましょう:

- fixed effects をあらわすパラメーター β_1 と β_2 は最尤推定する
- random effects あらわすパラメーター $\{r_i\}$ 100 個は **最尤推定しない**
- ただし $\{r_i\}$ たちの「ばらつき」をあらわす s は最尤推定する

23. モデル選択規準 AIC なんかの計算で使う**パラメーター数**とは「最尤推定して推定値をきめちゃったパラメーター数」なので，この場合はパラメーター数が 3 個となります．

6. R でやってみる GLMM 推定

GLMM の尤度方程式は積分が入っていたりして，なんともややこしいものですが……さいわいにも R でこのような混合ロジスティックモデルをあつかう glmmML package があるので，[24] それを使ってパラメーター推定してみましょう．

この glmmML package は標準ではない R package なので CRAN サイト [25] からダウンロード & インストールすることになります．その方法はいろいろあるのですが……どの OS の R であっても

```
> update.packages()
> install.packages("glmmML")
```

とすればインストールできるでしょう．[26] 最初の update.packages() 関数よびだし時に,「どの CRAN サイトを使うか?」といった質問があるかもしれないので，国内の CRAN サイトのひとつを選んでください．[27]

24. 混合ポアソンモデルも glmmML で推定できます．

25. R の機能拡張用の追加 pacakge 置き場，The Comprehensive R Archive Network (CRAN) http://cran.r-project.org/

26. Windows 版または Mac OS X 版 の R では menu bar から package インストールを選んで操作することもできます．

27. というかどこでもいいのですが．

さて，glmmML package のインストールができたものとして，次にこの package に含まれている glmmML() 関数を使った GLMM の最尤推定を試みてみましょう． [28] 手順は glm() による推定とほとんど同じなんですが，r_i が「個体ごとに [29] 与えられるパラメーター」であることを cluster オプションで明示的に指定する必要があります．この例だとすでに d の id 列に格納されている個体番号を使えばよく，つまり cluster = id とします．

```
> library(glmmML) # glmmML package の読みこみ
> fitC <- glmmML(cbind(y, N - y) ~ x, data = d, family = binomial,
+ cluster = id, method = "ghq")
```

glmmML() の指定の最後で method = "ghq" を指定しました．これは数値的な最尤推定計算の方法を指定しています．"ghq" は Gauss-Hermite 求積法 (Gauss-Hermite quadrature;) です．何も指定しないと default の"Laplace" つまり Laplace 近似法になります．どちらを指定するかは一長一短で，"ghq" のほうが係数の推定値がマシであることが多いのですが， [30] help(glmmML) によると"ghq" では s の推定が近似値になってしまうようです． [31]

さてさて，glmmML() の推定結果を格納している fitC を調べてみましょう:

```
> fitC
Call:  glmmML(formula = cbind(y, N - y) ~ x, family = ...(略)...

              coef se(coef)      z Pr(>|z|)
(Intercept) -4.190   0.8777 -4.774 1.81e-06
x            1.005   0.2075  4.843 1.28e-06

Standard deviation in mixing distribution:  2.408 gaussian
Std. Error:                                 0.2202

Residual deviance: 269.4 on 97 degrees of freedom       AIC: 275.4
```

この推定結果の読みかたを説明しましょう:

- Call の下にある table は glm() 出力でいう Coefficients (係数) つまりパラメーターの最尤推定値[32] です: $\hat{\beta}_1 = -4.19$ (ホントの $\beta_1 = -4$), $\hat{\beta}_2 = 1.01$ (ホントの $\beta_1 = 1$) とうまく推定できてるようです [33]
- Standard deviation ... は「個体差 r_i のばらつき」こと s の最尤推定値，その下の Std. Error は s の推定値のばらつき (標準誤差) です: $\hat{s} = 2.4$ (ホントの $s = 3$)，と過小推定されています
- 100 個のデータにたいして $\{\beta_1, \beta_2, s\}$ の 3 パラメーターを使っているので (使える) 残りの自由度は 100 - 3 = 97，そのときの residual deviance は 269.4 で AIC は 275.4 ということです

28. glmmML という名前は GLMM を ML (最尤推定) するといった意味です．これに対して最尤推定ではない GLMM 推定関数がありますけれど，これはいろいろと不便なのでおススメではありません．

29. つまりこの d data frame の場合だと「行ごとに」

30. Laplace 近似より GHQ のほうが近似計算の精度が良いのでしょう．

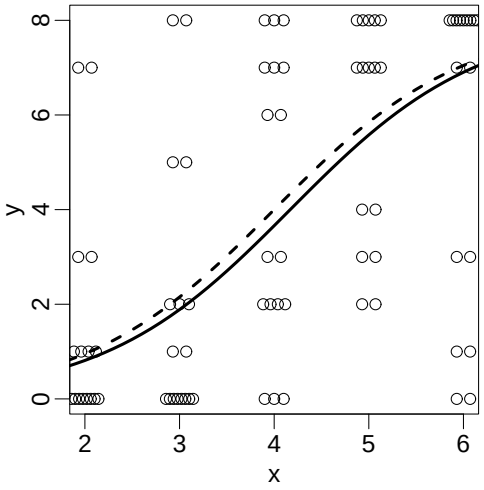
31. "Laplace" でも近似値になります．"ghq" のほうが計算に時間がかかり，ひょっとしたら推定計算そのものがうまくいかない確率も高いかもしれません……不安な場合は両方の method を比較してください．

32. そしてその憶測標準誤差や Wald の z 値など

33. じつは……今回はかなりうまく推定できていますが，いつもいつもこのようにうまく推定できるわけではありません．とくにこの例のように「個体差」のばらつき s が大きい場合には．

つづけて，この glmmML() に予測を図示してみましょう．[34] [35]

```
> plot.d(d, col = "black")
> lines(xx, logistic(-4 + xx) * 8, lty = 2) # 「真」の値
> beta <- fitC$coefficients
> lines(xx, logistic(beta[1] + beta[2] * xx) * 8) # 予測
```



34. じつは glmmML() を使ったからといって常にこれほど正確な推定ができるわけではありません……推定のばらつきについてはこの講義のーとの最初に URL を紹介した「GLMM 解説文」などを参照してください．

35. これもまた glm() の結果格納オブジェクトと同様に，glmmML() の結果格納オブジェクトにもさまざまな情報が入っています．この例だと fitC がそれに該当しますので，names(fitC) や str(fitC) などとすればどういう内容が入ってるかわかります．

7. 混合モデルの使いどころ

私たちが実際にあつかう観測データには「個体差」・「場所差」など random effects としてあつかうべき要因がいろいろと入りこんでいて，これらが二項分布などの過分散の原因となり，R の glm() ではうまくパラメーター推定できないので glmmML() を使おうというハナシでした．

二項分布ではなくポアソン分布を使った統計モデリングのときも，やはり glmmML package が使えて，glmmML(..., family = poisson, ...) と指定するだけです．またポアソン分布の場合，平均 λ_i の「個体差」が対数正規分布ではなくガンマ分布になっていると仮定する場合には [36] カウントデータのばらつきは**負の二項分布** (negative binomial) になります．このときには library(MASS) 内の glm.nb() という負の二項分布 GLM の推定関数が使えます．

また観測データが正規分布の場合はどうなるでしょうか? データのばらつきが正規分布で，見えざる「個体差」のばらつきも正規分布，平均は $\beta_1 + \beta_2 x_A + \beta_3 x_B + ...$ と「線形予測子そのまま」 [37] であれば線形混合モデル (linear mixed model) [38] ということになり，これはじつは nested ANOVA だの repeated なんちゃらだのと呼ばれるあれこれの正体です．R を使う場

36. random effects は目に見えないので，対数正規分布でもガンマ分布でもたいしてちがいはないのですが．

37. つまり R の glm() でいうところの identity link 関数．

38. これは GLMM の一部分ですね．

合，library(lme4) の lmer() 関数などが推定につかえます.

R における混合モデルを推定する道具だてはどうなっているのか，というのは以下の URL にまとめていますので参照してください.

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

この講義ではあまり深いりしませんが，混合モデルの考えかたを発展させていくといろいろと興味ぶかい現象の統計モデリングに応用できます．たとえば

- 一個体から**何度もサンプリング**をくりかえす縦断的データ (longitudinal data) の解析，あるいは**擬似反復** (pseudo replication) 問題への対策
- 実験ブロック差がある中での個体差といった**ネストされた** (nested な) random effects のモデリング
- 距離が近いほど二個体は挙動がより似ている，といった**空間相関**の問題

など，生態学データ解析で「よくある状況」です．これらに対処できる統計モデルを考えるときに，今回の講義でとりあつかった最も簡単な GLMM はその出発点となるものです.

「人間には観測できなかった・できない，しかし観測データにばらつきをもたらす」要因を **random effects としてあつかう**，というのはたいへん重要な考えかただと思います．もし random effects を知らないとすると，

- 「個体差」みたいなものが影響しているので，応答変数 y の挙動が説明変数 x_A でうまく説明できないとする
- ということで「もっと観測・実験を!」[39] とデータをとって説明変数 x_B を追加してみる
- それでも「個体差」は説明できないので，説明変数 x_C, x_C, $\cdots$ と増やしていく
- 使いものになるのかならないのかわからない要因が増え (パラメーター数の増大!) また説明変数間にも**ややこしい相関**があったりしてどんどん難しくなる

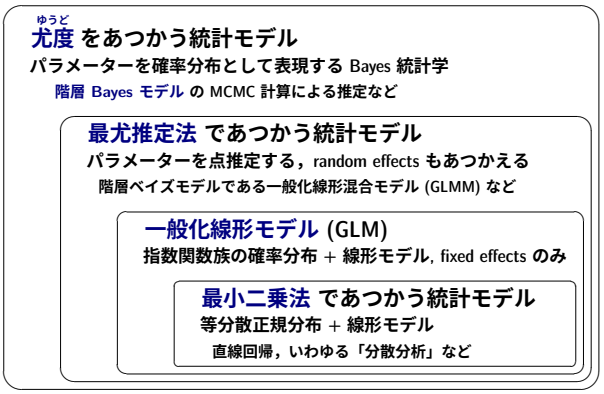
……といった悪しき状況におちいってる研究をよくみかけます.

混合モデルの考えかたはこれとは逆で，人間には説明変数のすべてを観察することは不可能と判断したうえで，統計モデルのどの部分に「個体差」・場所差にもとづく **random effects** が入るのかを注意ぶかく検討し，モデリングを工夫します．これによって，**興味のある fixed effects** 要因が観察されたパターンとどう関係しているのか，その解明に全力をつくすことができるようになります.

39. とくに野外調査をやってるヒトに，こういう「データむちゃくちゃに増やせばヨシ」根性論みたいなものの信奉者が多いように思います.

8. GLMM と階層ベイズモデル

最後に GLMM と階層ベイズモデルの関係にふれておきましょう.



「階層ベイズモデルとは何か?」についてはこの講義 web site からリンクしているいろいろな情報を見てもらうとして……ヒトことで言ってしまえば, 今回説明した **GLMM は階層ベイズモデルの一種**です. ただし，以下のような特徴があります.

- fixed effects をあらわすパラメーター β_1 や β_2 は事前分布を明示的に仮定しない，そしてこれらを最尤推定している
- random effects をあらわすパラメーター r_i に関しては「平均ゼロで標準偏差 s の正規分布」という事前分布を仮定している
- r_i の事前分布をきめる超パラメーター (hyper parameter) である s も事前分布は明示的に仮定せず，最尤推定値を計算している

「どのパラメーターも確率分布で表現される」とするのがベイズモデルの特徴なので，今回は最尤推定した $\{\beta_1, \beta_2, s\}$ に関しても事前分布を明示的に仮定し，これらの事後分布を推定する，というのがより「それっぽい」ベイズ推定ということになりそうですね. このあたりのハナシに関しては……また補講が必要になってしまうのでしょうか……?