

R と RStudio の使い方

芳賀敏郎 (2016) 医薬品開発のための統計解析 第3部 非線形モデル
4 ロジスティック回帰分析
4.2 基本

テキストと利用上の注意

●テキスト

芳賀敏郎（2016）医薬品開発のための統計解析

第3部 非線形モデル 改訂版、サイエンティスト社、p.288

（サイトへアップすることに対して、サイエンティスト社の了解を得ています）

●Rによる解析事例を紹介

R スクリプトの出力結果を紹介します（tidyverse 系には次期バージョンで対応します）

R スクリプト（文字コードUTF-8に設定）を、このサイトからダウンロードできます

R スクリプトを [Compile Report] することにより、Word または HTML で見ることができます

R と RStudio の設定と基本的な使い方は「[R と RStudio の使い方](#)」を参照してください

R の出力結果の見方は、テキストとそれを解説した [PDF ファイル](#) を参照してください

グラフ表示は、解析手段として、必要最小限の表現に止めています

●自己責任で利用

上記のことを理解した上で、自己責任により利用してください

第3部 非線形モデル

1. 非線形最小2乗法（基礎）

- 1.1 線形と非線形、1.2 非線形最小2乗法の基本的な考え方、1.3 指数曲線のあてはめ、
1.4 Emaxモデルとロジスティック曲線

2. 非線形最小2乗法（応用）

- 2.1 誤差を考慮した解析、2.2 効力比、2.3 併用効果（相乗・拮抗交換）、
2.4 モデルの探索（複数の曲線の同時あてはめ）、2.5 薬物動態の解析

3. 計数値の解析

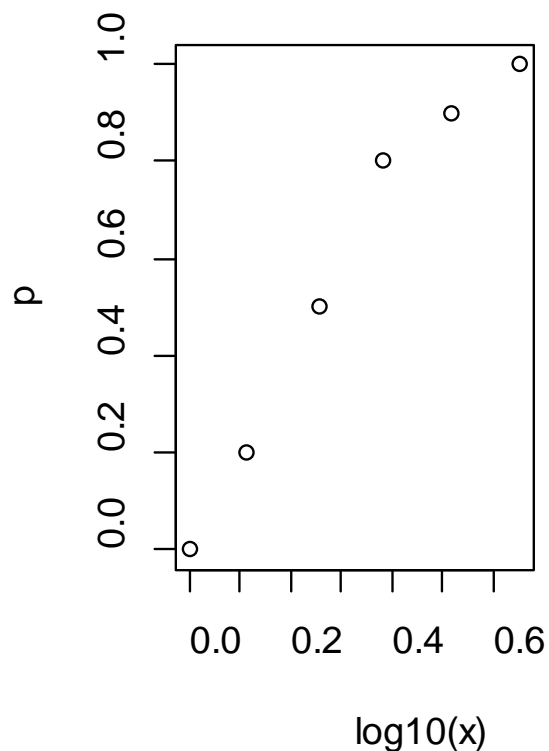
- 3.1 2項分布、3.2 割合の推定・検定と区間推定、3.3 割合の差の推定・検定と区間推定、
3.4 多項分布（名義尺度）、3.5 多項分布（順序尺度）、3.6 要因が複数の場合

4. ロジスティック回帰分析

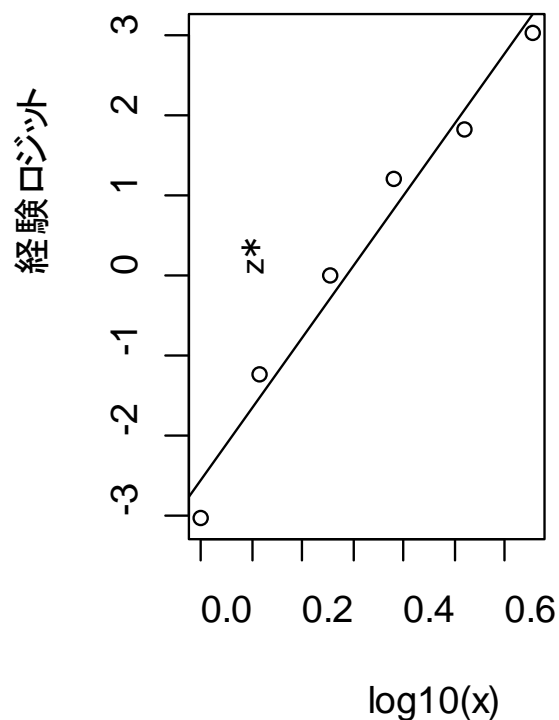
- 4.1 復習、4.2 **ロジスティック回帰分析（基本）**、4.3 ロジスティック回帰分析（応用）

- 表示 4.2.1 副作用の発生比率、表示 4.2.2 X と経験ロジット z^* の関係
スクリプトファイル：Green3-4-2a.R

表示4.2.1



表示 4.2.2



表示 4.2.1 副作用の発生比率、データとグラフ
(2) 副作用の発生数 (3) 副作用の発生割合

x	f	n
1.0	0	10
1.3	2	10
1.8	5	10
2.4	8	10
3.3	9	10
4.5	10	10
計		60

x	n	$p(f/n)$
1.0	10	0.00
1.3	10	0.20
1.8	10	0.50
2.4	10	0.80
3.3	10	0.90
4.5	10	1.00
計		60

$$z^* = \ln \left(\frac{f + 0.5}{n - f + 0.5} \right)$$



ロジスティック回帰分析：データ

- 表示 4.2.1 副作用の発生比率、表示 4.2.2 X と経験ロジット z^* の関係

スクリプトファイル：Green3-4-2a.R

表示 4.2.1 副作用の発生比率、データとグラフ

(1) 個々の反応データ (0 : 副作用なし、1 : 副作用あり)

(2) 副作用の発生数

(3) 副作用の発生割合

<u>xx</u>	<u>ff</u>	<u>xx</u>	<u>ff</u>	<u>xx</u>	<u>ff</u>	<u>xx</u>	<u>ff</u>	<u>xx</u>	<u>ff</u>	<u>xx</u>	<u>ff</u>
1.0	0	1.3	0	1.8	1	2.4	1	3.3	1	4.5	1
1.0	0	1.3	0	1.8	0	2.4	1	3.3	1	4.5	1
1.0	0	1.3	1	1.8	1	2.4	1	3.3	1	4.5	1
1.0	0	1.3	0	1.8	0	2.4	0	3.3	1	4.5	1
1.0	0	1.3	0	1.8	1	2.4	1	3.3	1	4.5	1
1.0	0	1.3	0	1.8	0	2.4	1	3.3	0	4.5	1
1.0	0	1.3	0	1.8	1	2.4	0	3.3	1	4.5	1
1.0	0	1.3	1	1.8	1	2.4	1	3.3	1	4.5	1
1.0	0	1.3	0	1.8	0	2.4	1	3.3	1	4.5	1
1.0	0	1.3	0	1.8	0	2.4	1	3.3	1	4.5	1



<u>x</u>	<u>f</u>	<u>n</u>
1.0	0	10
1.3	2	10
1.8	5	10
2.4	8	10
3.3	9	10
4.5	10	10
計		60

<u>x</u>	<u>n</u>	<u>p (f/n)</u>
1.0	10	0.00
1.3	10	0.20
1.8	10	0.50
2.4	10	0.80
3.3	10	0.90
4.5	10	1.00
計		60



ロジスティック回帰分析：glm

- 表示4.2.4 Excelソルバーによる解、表示 4.2.6 JMP 用データと出力
 スクリプトファイル：Green3-4-2a.R、利用した関数：glm

```
(1) glm(formula = ff ~ log10(xx), family = binomial(link = "logit"))
```

(1) 個々の反応データ (0 : 副作用なし、1 : 副作用あり)

<i>xx</i>	<i>ff</i>	<i>xx</i>	<i>ff</i>	<i>xx</i>	<i>ff</i>	<i>xx</i>	<i>ff</i>	<i>xx</i>	<i>ff</i>	<i>xx</i>	<i>ff</i>
1.0	0	1.3	0	1.8	1	2.4	1	3.3	1	4.5	1
1.0	0	1.3	0	1.8	0	2.4	1	3.3	1	4.5	1
1.0	0	1.3	1	1.8	1	2.4	1	3.3	1	4.5	1
1.0	0	1.3	0	1.8	0	2.4	0	3.3	1	4.5	1
1.0	0	1.3	0	1.8	1	2.4	1	3.3	1	4.5	1
1.0	0	1.3	0	1.8	0	2.4	1	3.3	0	4.5	1
1.0	0	1.3	0	1.8	1	2.4	0	3.3	1	4.5	1
1.0	0	1.3	1	1.8	1	2.4	1	3.3	1	4.5	1
1.0	0	1.3	0	1.8	0	2.4	1	3.3	1	4.5	1
1.0	0	1.3	0	1.8	0	2.4	1	3.3	1	4.5	1

ロジスティック回帰分析：glm

p.246

- 表示4.2.4 Excelソルバーによる解、表示 4.2.6 JMP 用データと出力
スクリプトファイル：Green3-4-2a.R、利用した関数：glm

```
(2) mx <- cbind(f, n - f)
    glm(mx ~ log10(x), family = binomial(link = "logit"))
```

```
(3) p <- f / n
    glm(p ~ log10(x), family = binomial(link = "logit"), weights = n)
```

引数
weights

```
glm(p ~ log10(x),
     family = binomial(link = "logit"),
     weights = n,
     data = df)
```

(2) 副作用の発生数

x	f	n
1.0	0	10
1.3	2	10
1.8	5	10
2.4	8	10
3.3	9	10
4.5	10	10
計		60

(3) 副作用の発生割合

x	n	$p(f/n)$
1.0	10	0.00
1.3	10	0.20
1.8	10	0.50
2.4	10	0.80
3.3	10	0.90
4.5	10	1.00
計		60

- 表示4.2.4 Excelソルバーによる解、表示 4.2.6 JMP 用データと出力

スクリプトファイル：Green3-4-2a.R

```
(1) ## glm(formula = ff ~ log10(xx), family = binomial(link = "logit"))
##
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.9852     0.8291  -3.601 0.000317 ***
## log10(xx)   11.3009     2.7217   4.152 3.29e-05 ***
```

Wald 検定

```
(2) ## glm(formula = df ~ log10(x), family = binomial(link = "logit"))
##
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.9852     0.8291  -3.600 0.000318 ***
## log10(x)    11.3009     2.7220   4.152 3.3e-05 ***
```

```
(3) ## glm(formula = p ~ log10(x), family = binomial(link = "logit"), weights = n)
##
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.9852     0.8291  -3.600 0.000318 ***
## log10(x)    11.3009     2.7220   4.152 3.3e-05 ***
```




ロジスティック回帰分析：glm

p.251

- 表示4.2.4 Excelソルバーによる解、表示 4.2.6 JMP 用データと出力

スクリプトファイル：Green3-4-2a.R

利用した関数：glm、anova

方法：尤度比検定

```
anova(glm_out3, test = "LRT")
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: p
##
## Terms added sequentially (first to last)
##
##
##              Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                    5    41.727
## log10(x)    1    39.971    4    1.756 2.577e-10 ***
```



●glm 関数の記述の仕方（モデル式とその記述方法）

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

```
glm(p ~ x, family = binomial(link = "logit"))
```

$$p = \frac{1}{1 + \exp(-(a + bx_1 + cx_2))}$$

```
glm(p ~ x1 + x2, ...)
```

モデル式そのものを記述していない

$$p = \frac{1}{1 + \exp(-(a + bx_1 + cx_2 + dx_1x_2))}$$

```
glm(p ~ x1 + x2 + x1:x2, .)    glm(p ~ x1 * x2, .)
```

```
glm(p ~ x1 + x2 + x1 * x2, .)    glm((x1 + x2)^2, .)
```

$$p = \frac{1}{1 + \exp(-(a + bx + cx^2))}$$

```
glm(p ~ x + I(x^2), ...)
```

I 関数で、式そのものを入力

$$p = \frac{1}{1 + \exp(-bx_1)}$$

```
glm(p ~ x1 - 1, ...)
```

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

```
glm(p ~ 1, ...)
```



●glm 関数の結果の抽出

```
print(glm_out)           # 結果の概要、glm_out だけでも可
summary(glm_out)        # 結果の概要
deviance(glm_out)       # デビアンスの表示
logLik(glm_out)         # 対数尤度の表示
coefficients(glm_out)   # 説明変数の係数と定数項（切片）の表示
confint(glm_out)        # 上記の区間推定
fitted(glm_out)         # 予測値（y-hat）の表示
predict(glm_out, type = "response") # 予測値（y-hat）の表示
predict(glm_out, type = "link")    # リンク関数(ロジット、 $a + bx$ ) の値
newd <- data.frame(x = c(2, 3, 4)) # 推定するための新しい x の値
predict(glm_out, newdata = newd, type = "response") # 新しい予測値（y-hat）
residuals(glm_out, type = "response") # 残差の表示
residuals(glm_out, type = "deviance") # デビアンス残差
residuals(glm_out, type = "pearson") # ピアソン残差
```



ロジスティック回帰分析：glm

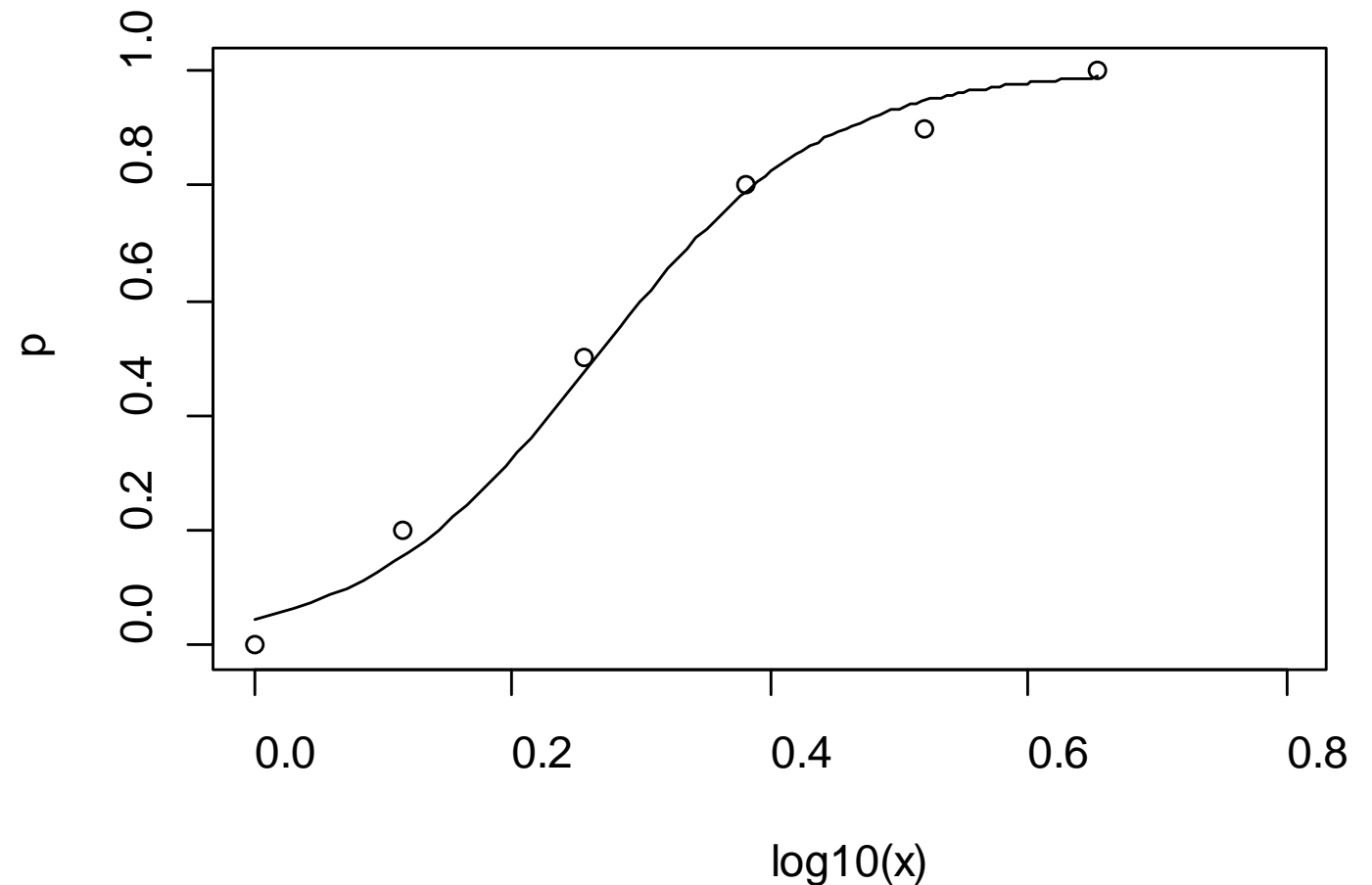
- 表示4.2.4 Excelソルバーによる解、表示 4.2.6 JMP 用データと出力

スクリプトファイル

Green3-4-2a.R

利用した関数

glm、plot、predict



- 表示4.2.8 JMP による X50 の逆推定

スクリプトファイル：Green3-4-2a.R

利用した関数：glm、investr::invest

方法：invest 関数による逆推定、データフレームにしてglm を実行

```
df <- data.frame(p, X, n)
glm_out5 <- glm(p ~ X, family = binomial(link = "logit"), weights = n, data = df)
invest(glm_out5, y0 = 0.5, interval = "Wald", level = 0.95)
```

```
## estimate      lower      upper      se
## 0.2641533 0.1963924 0.3319141 0.0345725
```

log10(x)

Wald 法

$X_{50} = 0.264 = \log 1.837$

- 表示 4.2.4 Excelソルバーによる解、表示 4.2.9 「非線形回帰」による解

スクリプトファイル

Green3-4-2b.R

表示 4.2.4 Excelソルバーによる解

x	f	n	phat	-2対数尤度
1.0	0	10	0.048	0.986
1.3	2	10	0.155	2.540
1.8	5	10	0.475	2.829
2.4	8	10	0.788	2.404
3.3	9	10	0.947	2.242
4.5	10	10	0.988	0.245

a	-2.985
b	11.301
L	11.246

```
x <- c(1.0, 1.3, 1.8, 2.4, 3.3, 4.5)
f <- c(0, 2, 5, 8, 9, 10)
n <- rep(10, 6)
# 予測式
phat <- function(a, b) {
  p = 1 / (1 + exp(-(a + b * log10(x))))
  return(p)
}
# -対数尤度
LL <- function(a, b) {
  -sum(dbinom(f, n, phat(a, b), log = TRUE))
}
mle2_out <- mle2(minuslogl = LL,
                 start = list(a = -2.5, b = 8.8))
```

モデル式を
そのまま記述

- 表示 4.2.4 Excelソルバーによる解、表示 4.2.9 「非線形回帰」による解

スクリプトファイル

Green3-4-2b.R

表示 4.2.4 Excelソルバーによる解

x	f	n	phat	-2対数尤度
1.0	0	10	0.048	0.986
1.3	2	10	0.155	2.540
1.8	5	10	0.475	2.829
2.4	8	10	0.788	2.404
3.3	9	10	0.947	2.242
4.5	10	10	0.988	0.245

a	-2.985
b	11.301
L	11.246

```
x <- c(1.0, 1.3, 1.8, 2.4, 3.3, 4.5)
f <- c(0, 2, 5, 8, 9, 10)
n <- rep(10, 6)
```

x、f、nは
グローバル変数

```
# 予測式
```

```
phat <- function(a, b) {
  p = 1 / (1 + exp(-(a + b * log10(x))))
  return(p)
}
```

モデル式を記述

```
# -対数尤度
```

2倍にしてない

```
LL <- function(a, b) {
  -sum(dbinom(f, n, phat(a, b), log = TRUE))
}
```

-対数尤度の和

```
mle2_out <- mle2(minuslogl = LL,
  start = list(a = -2.5, b = 8.8))
```

初期値を設定

- 表示 4.2.4 Excelソルバーによる解、表示 4.2.9 「非線形回帰」による解

スクリプトファイル：Green3-4-2b.R、利用した関数：bbmle::mle2、summary

表示 4.2.4

Excelソルバーによる解

x	f	n	phat	-2対数尤度
1.0	0	10	0.048	0.986
1.3	2	10	0.155	2.540
1.8	5	10	0.475	2.829
2.4	8	10	0.788	2.404
3.3	9	10	0.947	2.242
4.5	10	10	0.988	0.245

a -2.985

b 11.301

L 11.246

summary 関数によるmle2 関数の結果出力（一部）

```
## Coefficients:
## Estimate Std. Error z value Pr(z)
## a -2.98513 0.82912 -3.6004 0.0003178 ***
## b 11.30075 2.72199 4.1517 3.301e-05 ***
##
## -2 log L: 11.24596
```

confint 関数による95% 信頼区間の出力

```
## 2.5 % 97.5 %
## a -4.903552 -1.572015
## b 6.766665 17.708119
```




- 表示 4.2.10 x50 を直接求める「非線形回帰」の解、表示 4.2.11 Excelによるx50を含むモデルの解

スクリプトファイル：Green3-4-2c.R

利用した関数：bbmle::mle2、summary

方法：x50を含むモデルで、直接 x50 を求める

モデル式を記述

-対数尤度の和

```
# 予測式
phat <- function(x50, b){
  p = 1 / (1 + exp(-(b * (log10(x) - log10(x50)))))
  return(p)
}
# -対数尤度
LL <- function(x50, b){
  -sum(log(dbinom(f, n, phat(x50, b))))
}
mle_out <- mle2(minuslogl = LL,
               start = list(x50 = 1.8, b = 10),
               control = list(maxit = 5000))
```



ロジスティック回帰分析 mle2

- 表示 4.2.10 x50 を直接求める「非線形回帰」の解、表示 4.2.11 Excelによるx50を含むモデルの解
スクリプトファイル：Green3-4-2c.R
利用した関数：bbmle::mle2、summary
方法：パラメータの一部に制約を設けて解を求める

```
mle_out <- mle2(minuslogl = LL,  
               start = list(x50 = 1.8, b = 10),  
               control = list(maxit = 5000))
```



```
#x50=2.156 に固定した解  
mle_out2 <- mle2(minuslogl = LL,  
                 fixed = list(x50 = 2.155825),  
                 start = list(b = 10),  
                 control = list(maxit = 5000))  
  
summary(mle_out2)
```

制約を設定
x50 = 2.156

x50 の初期値を
削除

テキストでは
2.156



ロジスティック回帰分析 最適投与量が存在する場合

p.255

- 表示 4.2.13 Excelによる解析、表示 4.2.14 JMP [モデルのあてはめ]による解析

スクリプトファイル：Green3-4-2d.R

利用した関数：glm、summary

方法：glm を使い、ロジットを X の2次式で近似

2次式のあてはめ

```
glm_out <- glm(p ~ X + I(X ^ 2), weights = n,  
              family = binomial(link = "logit"))
```

summary 関数による glm 関数の結果出力 (一部)

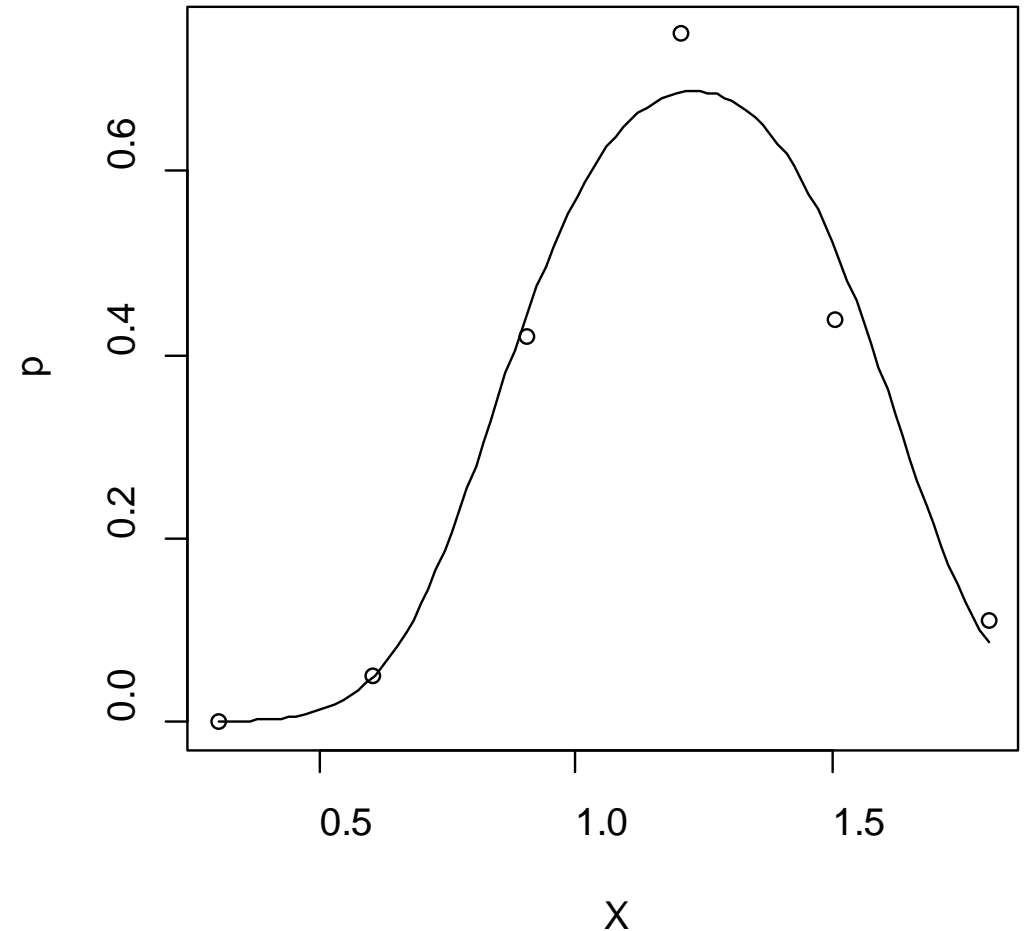
```
## Coefficients:  
##           Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -13.584      3.209  -4.233 2.30e-05 ***  
## X           23.359      5.430   4.302 1.69e-05 ***  
## I(X^2)      -9.491      2.192  -4.330 1.49e-05 ***
```

- 表示 4.2.13 Excelによる解析、表示 4.2.14 JMP [モデルのあてはめ]による解析

スクリプトファイル：Green3-4-2dd.R

利用した関数：glm、predict

方法：glm を使い、ロジットを X の2次式で近似
predict 関数で、新しい X に対する $y\text{-hat}$ を
計算して描画



- 表示 4.2.13 Excelによる解析、表示 4.2.15 JMP [非線形回帰]による解析

スクリプトファイル：Green3-4-2d.R

利用した関数：bbmle::mle2、summary

方法：mle2 を使い、p を最大とする X を X50、そのときのロジットを zmax とするモデルを利用
モデルはパラメータに対して非線形なので、glm では解けない

```
# 予測式
phat <- function(zmax, k, Xopt) {
  p = 1 / (1 + exp(-(zmax + k * (X - Xopt) ^ 2)))
  return(p)
}
# -対数尤度
LL <- function(zmax, k, Xopt){
  -sum(dbinom(f, n, phat(zmax, k, Xopt), log = TRUE))
}
mle_out <- mle2(minuslogl = LL,
               start = list(zmax = 0.5, k = -5, Xopt = 1.3),
               control = list(maxit = 5000))
```



プロビット分析

p.281

- 表示 4.4.1 プロビット曲線のあてはめ

スクリプトファイル：Green3-4-2e.R

利用した関数：bbmle::mle2、summary

方法：同じデータにロジスティック回帰分析とプロビット分析を行う

```
# ロジスティック回帰分析
```

```
glm.logit.out <- glm(y ~ log10(x), weights = f,  
                    family = binomial(link = "logit"), data = dfA)  
summary(glm.logit.out)
```

```
# プロビット回帰分析
```

```
glm.probit.out <- glm(y ~ log10(x), weights = f,  
                    family = binomial(link = "probit"), data = dfA)  
summary(glm.probit.out)
```

リンク関数の指定



- 作成 片瀬雅彦
- 作成時期 2021年5月23日