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
3 計数値の解析
3.1 ２項分布

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
　本PDF ファイルをダウンロードし、Adobe Acrobat Reader DC で注釈のメモを表示してください

- 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）

2 項分布の関数

p.146

- 2 項分布に関する R 、 Excel、 JMP の関数

事例 ： 生起確率 0.5 の事象を 10 回繰り返す場合

機能	事例	R	Excel	JMP
確率	この事象が 2 回起こる確率	dbinom (x = 2, size = 10, prob = 0.5)	BINOM.DIST(2, 10, 0.5, FALSE)	Binomial Probability (0.5, 10, 2)
下側確率	この事象が 2 回以下で起こる確率 （0～2 回の範囲）	pbinom(q = 2 , size = 10, prob = 0.5, lower.tail = TRUE)	BINOM.DIST(2, 10, 0.5, TRUE)	Binomial Distribution(0.5, 10, 2)
下側%点	下側確率が 0.025 になる回数 （0～q 回、q は整数）	qbinom(p = 0.025, size = 10, prob = 0.5, lower.tail = TRUE)	BINOM.INV(10, 0.5, 0.25)	Binomial Quantile (0.5, 10, 0.025)

下側%点は、指定した下側確率 0.025 以上になる回数のうち、最小の回数 q （整数） を返す

（テキスト PDF ファイル 参照）

2項分布：簡単な例

p.139

- 表示3.1.1 f=n の確率の計算表
  - スクリプトファイル：Green3-3-1a.R
  - 利用した関数：dbinom
  - 方法：２項分布の関数を利用

	A	B	C	D
2	失敗率	成功率		
3	$1-\pi$	π	n	$f=n$ の確率
4	0.01	0.99	2	0.9801
5	0.01	0.99	100	0.3660
6	0.03	0.97	100	0.0476
7			D4: =B4^C4	

```
x      : f
size   : n
prob   : π
```

```
dbinom(x = 2, size = 2, prob = 0.99)
## [1] 0.9801
dbinom(x = 100, size = 100, prob = 0.99)
## [1] 0.3660323
dbinom(x = 100, size = 100, prob = 0.97)
## [1] 0.04755251
```

2項分布：シミュレーション

p.141

- 表示3.1.3 コイン実験
  - スクリプトファイル
    - Green3-3-1b.R
  - 利用した関数
    - sample、matrix
  - 方法
    - コインを1000回投げて、表と裏のどちらが何回出るか実験したい
    - この実験をPCでシミュレーションする
    - １：表、０：裏として、１と０を無作為に1000回発生させる

```
## 0と1 から無作為に1000個を抽出
set.seed(123) # 乱数の再現性を確保、通常は削除
vt <- sample(x = c(0, 1), size = 1000, replace = TRUE)

## 100行×10列のマトリックスに変換
mx <- matrix(vt, nrow = 100, ncol = 10)
```

重複の抽出を許す

マトリックス mx

```
##        [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
##   [1,]    0    0    1    1    0    1    1    1    0     1
##   [2,]    0    1    1    1    0    1    1    0    0     1
##   [3,]    0    1    1    0    0    1    1    0    1     0
                ................
##  [99,]    0    0    0    1    0    1    1    1    0     1
## [100,]    0    1    0    1    0    1    1    1    1     1
```

2項分布：シミュレーション

p.141

- 表示3.1.4 表の個数のまとめ

  スクリプトファイル：Green3-3-1b.R

  利用した関数：apply、sum、rowSums も利用可

  方法　apply でマトリックスの行ごとに合計を算出する

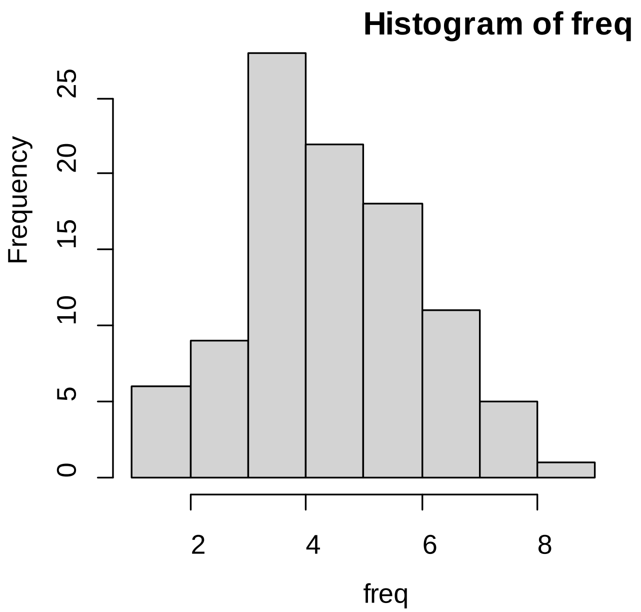
```
freq <- apply(mx, 1, sum)

hist(freq, breaks = 11)
```

要素数100のベクトル

1 は行、2 は列

ヒストグラム作成（第１部 §2.2）



マトリックス mx

```
##        [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
##   [1,]    0    0    1    1    0    1    1    1    0     1
##   [2,]    0    1    1    1    0    1    1    0    0     1
##   [3,]    0    1    1    0    0    1    1    0    1     0
          ・・・・・・・・・・・・・・・・
##  [99,]    0    0    0    1    0    1    1    1    0     1
## [100,]    0    1    0    1    0    1    1    1    1     1
```

1 の数は 6（合計6）

1 の数は 7（合計7）

2 項分布：シミュレーション

p.141

- 表示3.1.4 表の個数のまとめ

  スクリプトファイル：Green3-3-1b.R

  利用した関数：apply、sum、function

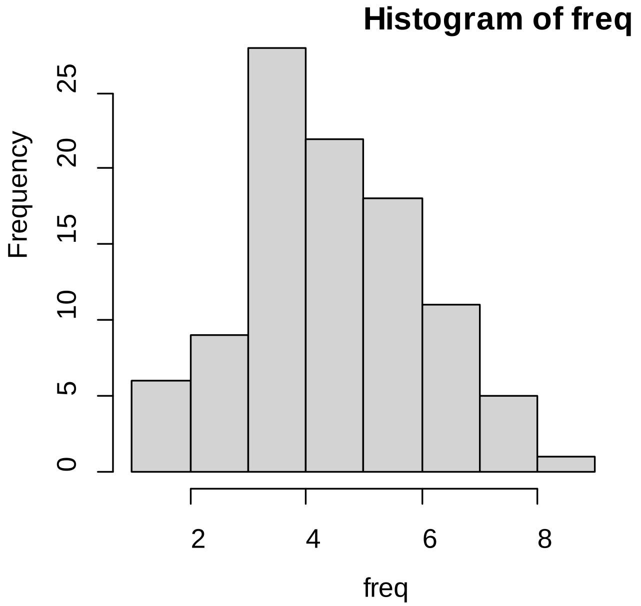
  方法：１行あたりの 0 の数を数える場合

  sum(x == 0) はベクトル x の中の 0 の数を返す

  apply 関数と function 関数を組み合わせる

```
apply(mx, 1, function(x) sum(x == 0))
```

２つのイコール



マトリックス mx

```
##        [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
##   [1,]    0    0    1    1    0    1    1    1    0     1
##   [2,]    0    1    1    1    0    1    1    0    0     1
##   [3,]    0    1    1    0    0    1    1    0    1     0
          ………………
##  [99,]    0    0    0    1    0    1    1    1    0     1
## [100,]    0    1    0    1    0    1    1    1    1     1
```

２項分布

p.143

- 表示3.1.6 π=1/3、n=3 のとき、f の確率

  スクリプトファイル

  Green3-3-1a.R

  利用した関数

  dbinom

  方法

  サイコロを３回投げて、１または６が出たら勝ちとする

  ２項分布の関数を利用

```
dbinom(x = 0, size = 3, prob = 1 / 3) * 27
## [1] 8 ･･･確率を表す分数の分子を表示させる
```

組合せ番号	勝ち（1）負け（0） 1回目	2回目	3回目	1,6の個数 f	組合せの数	確率（P_f）
(1)	0	0	0	0	1	$(2/3)^3 = 8/27 = p_0$
(2)	1	0	0	1		
(3)	0	1	0	1	3	$3\times(1/3)(2/3)^2 = 12/27 = p_1$
(4)	0	0	1	1		
(5)	1	1	0	2		
(6)	1	0	1	2	3	$3\times(1/3)^2(2/3) = 6/27 = p_2$
(7)	0	1	1	2		
(8)	1	1	1	3	1	$(1/3)^3 = 1/27 = p_3$
計					8	$27/27 = 1$

２項分布

p.145

- 表示3.1.7 Excelによる２項分布の確率の計算表
  スクリプトファイル：Green3-3-1a.R
  利用した関数：dbinom、pbinom
  方法：２項分布の関数を利用

	A	B	C	D	E	F
3	*f*	*n*	*π*	確率	累積確率	上側確率
4	0	3	0.3333	0.2963	0.2963	1.0000
5	1	3	0.3333	0.4444	0.7407	0.7037
6	1	10	0.5	0.0098	0.0107	0.9990
7	8	10	0.5	0.0439	0.9893	0.0547
8						
9	D4: =BINOMDIST(A4,B4,C4,FALSE)					
10	E4: =BINOMDIST(A4,B4,C4,TRUE)					
11	F4: =IF(A4=0,1,1-BINOMDIST(A4-1,B4,C4,TRUE))					

```
dbinom(x = 0, size = 3, prob = 1/3)
## [1] 0.2962963

pbinom(q = 0, size = 3, prob = 1/3,
       lower.tail = TRUE)
## [1] 0.2962963

pbinom(q = 0 - 1, size = 3, prob = 1/3,
       lower.tail = FALSE)
## [1] 1
```

q が 0以下に
なってもエラーに
ならない

x　　　：f
size　：n
prob　：π

２項分布

p.145

- 表示3.1.8 ２項分布の確率とそのグラフ
  スクリプトファイル：Green3-3-1b.R
  利用した関数：dbinom、cbind、plot
  方法：ベクトル演算により、
  １行のスクリプトで繰り返し処理を行う

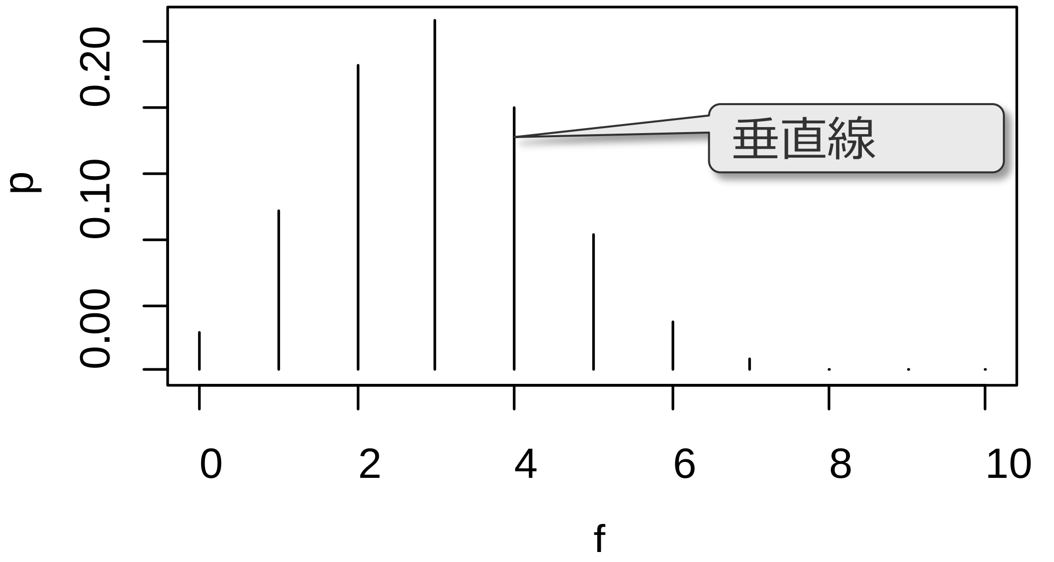
```
##        f  n  pi            p
##  [1,]  0 10 0.3 0.0282475249
##  [2,]  1 10 0.3 0.1210608210
##  [3,]  2 10 0.3 0.2334744405
##  [4,]  3 10 0.3 0.2668279320
##  [5,]  4 10 0.3 0.2001209490
##  [6,]  5 10 0.3 0.1029193452
##  [7,]  6 10 0.3 0.0367569090
##  [8,]  7 10 0.3 0.0090016920
##  [9,]  8 10 0.3 0.0014467005
## [10,]  9 10 0.3 0.0001377810
## [11,] 10 10 0.3 0.0000059049
```

```
f  <- 0:10
n  <- rep(10, 11)
p1 <- rep(0.3, 11)  # 母比率パイ
p = dbinom(x = f, size = n, prob = p1)
cbind(f = f, n = n, "pi" = p1, p = p)

plot(p ~ f, type = "h")
```

ベクトル演算

垂直線

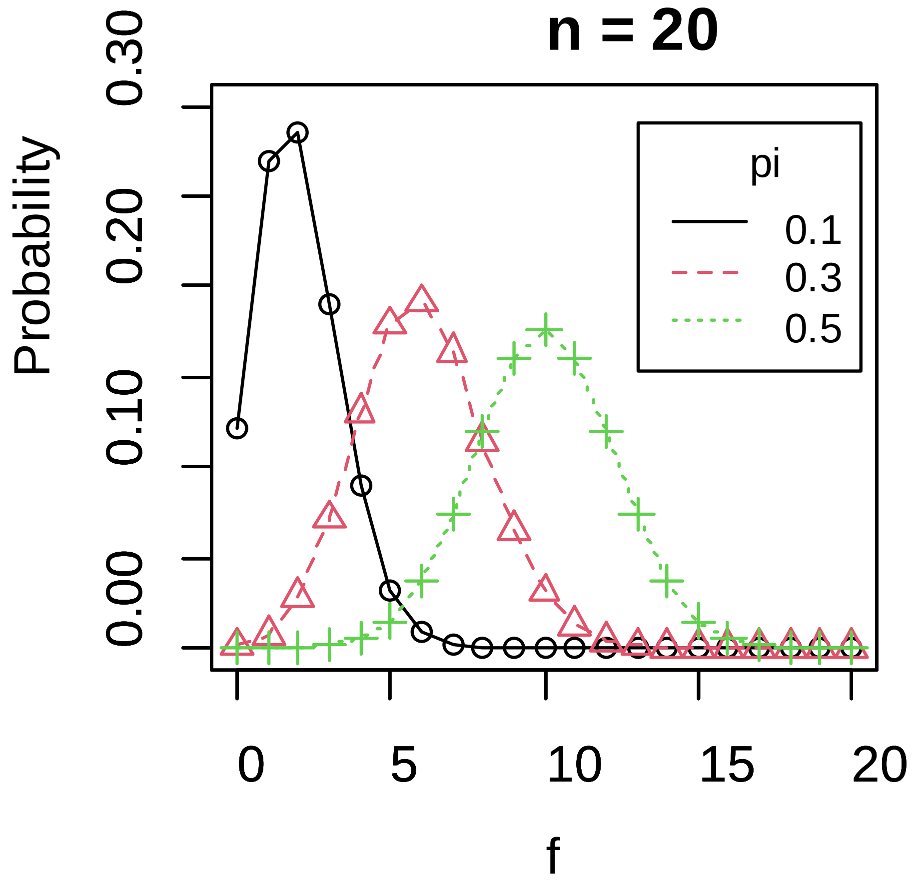


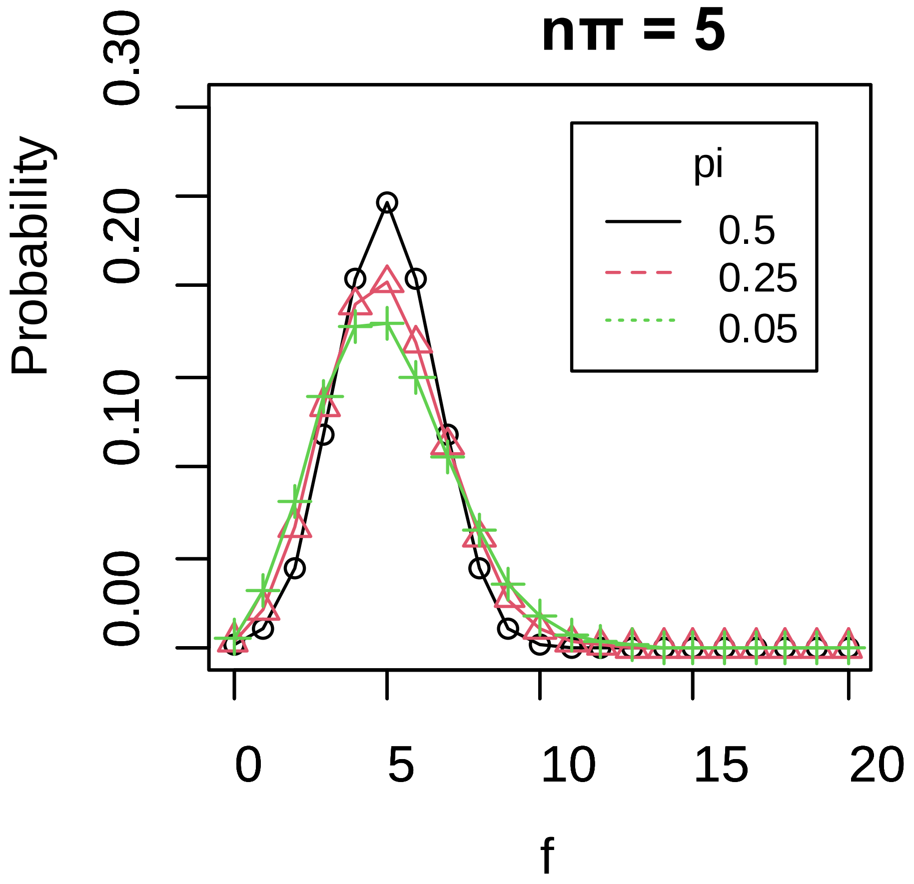
２項分布

p.147

- 表示3.1.10 n、π による２項分布の変化

  スクリプトファイル：Green3-3-1b.R、利用した関数：dbinom、plot





2 項分布の期待値と分散

p.148

- 表示3.1.11 π=0.4、n=5 の期待値と分散の計算

  スクリプトファイル：Green3-3-1a.R

  利用した関数

  dbinom、sum、cbind、lucid::lucid、print

```
f  <- 0:5
n  <-  5
pp <- 0.4 # 母比率パイ
p  <- dbinom(x = f, size = n, prob = pp)
pf <- p * f
e <- f - sum(pf)
pe2 <- p * e^2

mx <- cbind(f = f, n = n, "pi" = pp,
              p = p, pf = pf, e = e,
          "pe^2" = pe2)
mx <- rbind(mx, Total = colSums(mx))
mx[7, 1:3] <- NA
mx[7, 6] <- NA

s <- lucid(mx, dig = 4, na.value = "")
print(s, quote = FALSE)
```

ベクトル演算

不要な部分

成形

```
##         f  n  pi  p       pf      e  pe^2
##          0  5 0.4 0.07776 0      -2 0.311
##          1  5 0.4 0.2592  0.2592 -1 0.2592
##          2  5 0.4 0.3456  0.6912  0 0
##          3  5 0.4 0.2304  0.6912  1 0.2304
##          4  5 0.4 0.0768  0.3072  2 0.3072
##          5  5 0.4 0.01024 0.0512  3 0.09216
## Total               1       2         1.2
```

２項分布の正規近似

p.150

- 表示3.1.13 ２項分布（π=0.1、n=50）と正規分布近似
  スクリプトファイル：Green3-3-1b.R
  利用した関数：dbinom、dnorm、cbind、lucid

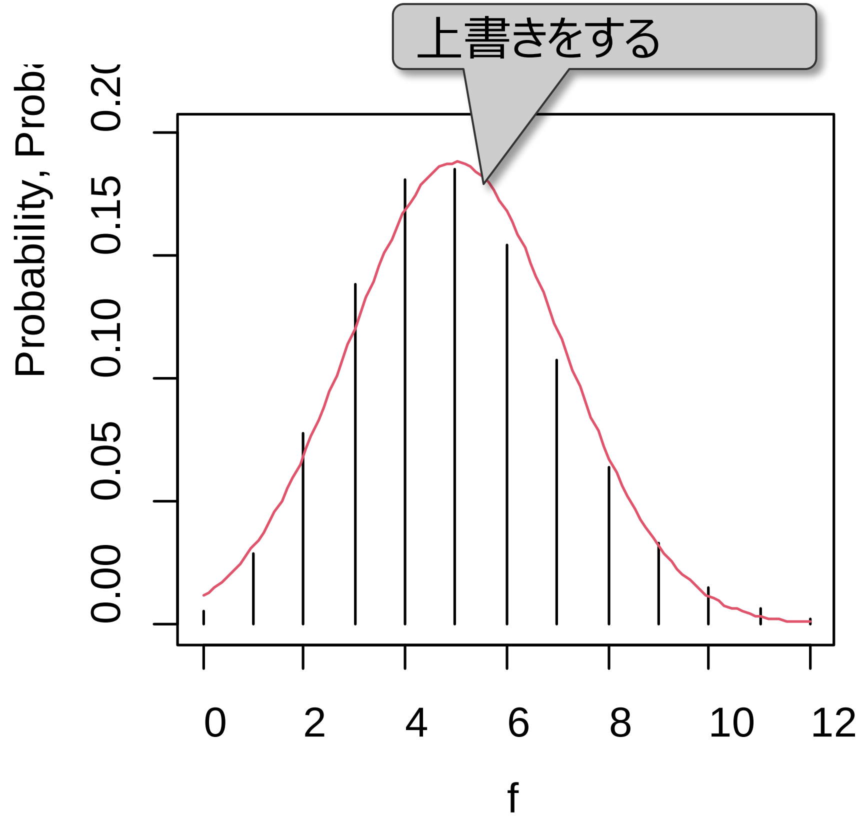
```
f <- 0:12                         # 起こる回数
n <- 50                           # 試行回数
pp <- 0.1                         # 起こる確率、パイ
Ef <- n * pp                      # 期待値
sd <- sqrt(n * pp * (1 - pp)) # 標準偏差
pb <- dbinom(x = f, size = n, prob = pp)

# ２項分布の棒グラフ
plot(x = f, y = pb, xlim = c(0, 12), ylim = c(0, 0.2),
     type = "h", xlab = "f",
     ylab = ("Probability, Probability Density"))
# 正規分布の曲線
curve(dnorm(x, mean = Ef, sd = sd), 0, 12,
      ylim = c(0, 0.2), col = 2, add = TRUE)
```

期待値と標準偏差
§3.1 参照

横軸の
最小値と最大値

上書きをする



２項分布：確率と尤度

p.151

- 表示3.1.14 ２項分布の確率と尤度
  - スクリプトファイル
    - Green3-3-1b.R
  - 利用した関数
    - dbinom、for、matrix
  - 方法：for 文を使って、２次元の表を作成

要素番号：1 ～ 11

```
f   <- 0:10
pp  <- seq(0.05, 0.95, by = 0.05)    # 母比率 π
mx1 <- matrix(0, nrow = 11, ncol = 19)  # 空の行列
for (i in 1:11) {
  mx1[i, ] <- dbinom(x = f[i], size = 10, prob = pp)
}
rownames(mx1) <- f
colnames(mx1) <- pp
round(mx1, digits = 2)
```

試行回数 10 の２項確率

mx1

```
##    0.05  0.1 0.15  0.2 0.25  0.3 0.35  0.4 0.45  0.5 0.55 ･･･ 0.8 0.85  0.9 0.95
## 0  0.60 0.35 0.20 0.11 0.06 0.03 0.01 0.01 0.00 0.00 0.00 ･･･0.00 0.00 0.00 0.00
## 1  0.32 0.39 0.35 0.27 0.19 0.12 0.07 0.04 0.02 0.01 0.00 ･･･0.00 0.00 0.00 0.00
## 2  0.07 0.19 0.28 0.30 0.28 0.23 0.18 0.12 0.08 0.04 0.02 ･･･0.00 0.00 0.00 0.00
## 3  0.01 0.06 0.13 0.20 0.25 0.27 0.25 0.21 0.17 0.12 0.07 ･･･0.00 0.00 0.00 0.00
## 4  0.00 0.01 0.04 0.09 0.15 0.20 0.24 0.25 0.24 0.21 0.16 ･･･0.01 0.00 0.00 0.00
                    ････････････････････
## 9  0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.02 ･･･0.27 0.35 0.39 0.32
## 10 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 ･･･0.11 0.20 0.35 0.60
```

２項分布：確率と尤度

- 表示3.1.14 ２項分布の確率と尤度

  スクリプトファイル

  Green3-3-1b.R

  利用した関数

  dbinom、for、matrix

π = 0.2

mx1[, 4]

f = 3

mx1[4,]

mx1

```
##     0.05  0.1 0.15  0.2 0.25  0.3 0.35  0.4 0.45  0.5 0.55 ·· 0.8 0.85  0.9 0.95
## 0   0.60 0.35 0.20 0.11 0.06 0.03 0.01 0.01 0.00 0.00 0.00 ···0.00 0.00 0.00 0.00
## 1   0.32 0.39 0.35 0.27 0.19 0.12 0.07 0.04 0.02 0.01 0.00 ···0.00 0.00 0.00 0.00
## 2   0.07 0.19 0.28 0.30 0.28 0.23 0.18 0.12 0.08 0.04 0.02 ···0.00 0.00 0.00 0.00
## 3   0.01 0.06 0.13 0.20 0.25 0.27 0.25 0.21 0.17 0.12 0.07 ···0.00 0.00 0.00 0.00
## 4   0.00 0.01 0.04 0.09 0.15 0.20 0.24 0.25 0.24 0.21 0.16 ···0.01 0.00 0.00 0.00
                          ····················
## 9   0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.02 ···0.27 0.35 0.39 0.32
## 10  0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 ···0.11 0.20 0.35 0.60
```

2 項分布：確率と尤度

- 表示3.1.15 確率と尤度のグラフ

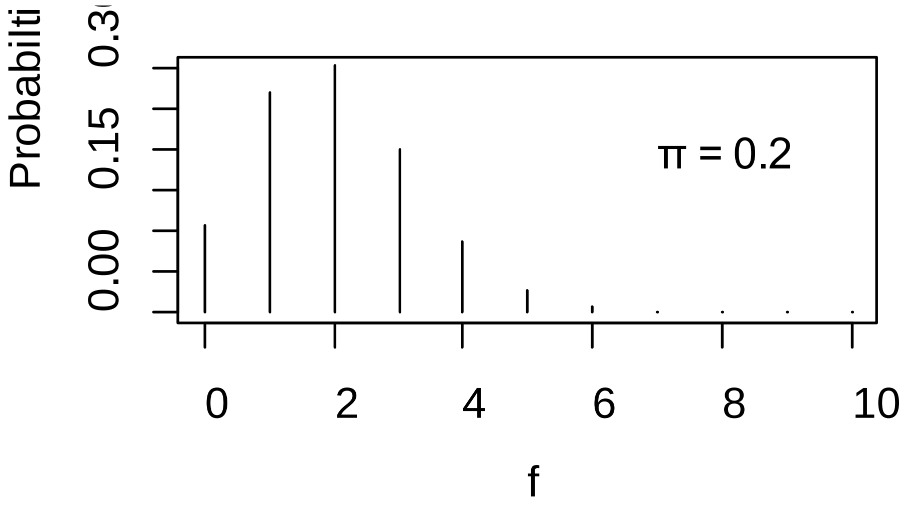
  スクリプトファイル：Green3-3-1b.R

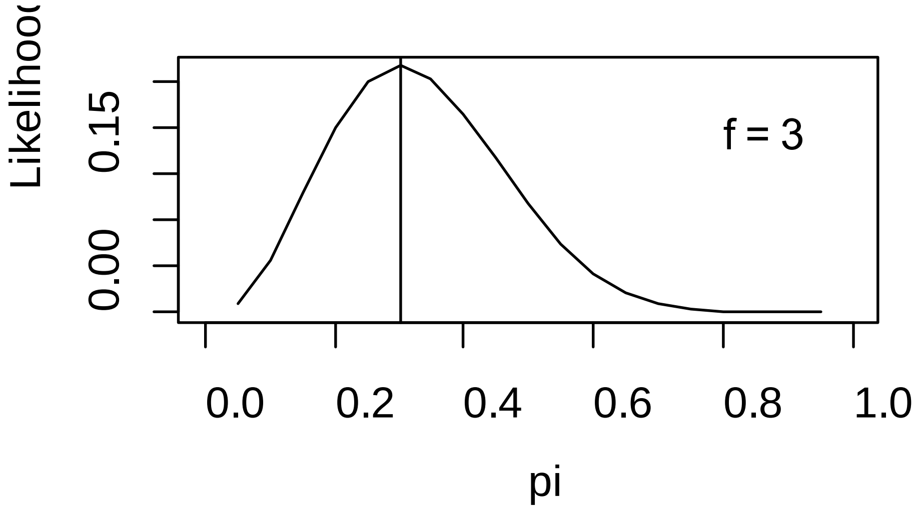
  利用した関数：dbinom、for、matrix

  方法：plot 関数を使い、引数 type = "h" に指定して棒グラフを表示

```
plot(f, mx1[, 4], type = "h", xlim = c(0, 10), xlab = "f", ylab = "Probabiltiy")

text(x = 7, y = 0.2, "π = 0.2")
```





2 項分布：確率と尤度

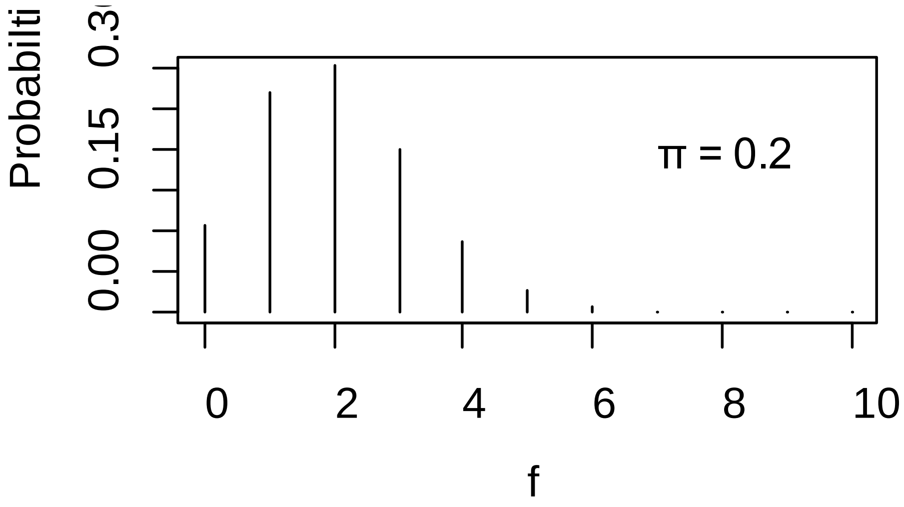
p.152

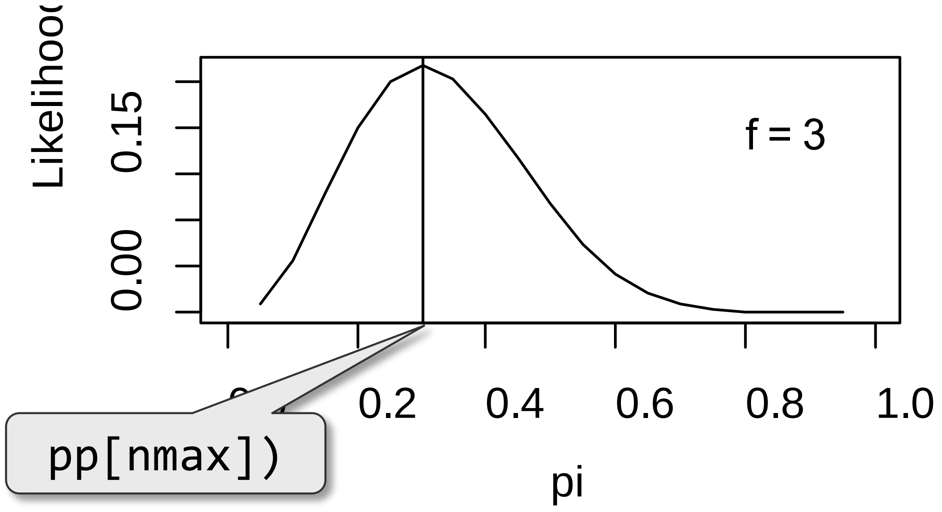
- 表示3.1.15 確率と尤度のグラフ
  - スクリプトファイル：Green3-3-1b.R
  - 利用した関数
    - dbinom、for、
    - matrix
  - 方法
    - 右の折れ線グラフを
    - plot 関数で
    - 引数 type = "l"
    - に指定して
    - 表示

```
pp  <- seq(0.05, 0.95, by = 0.05)
```

```
plot(x = pp, y = mx1[4, ], type = "l", xlim = c(0, 1),
     xlab = "pi", ylab = "Likelihood")

nmax <- which.max(mx1[4, ])    # ベクトルにおける最大値の要素番号
abline(v = pp[nmax])
text(x = 0.8, y = 0.2, "f = 3")
```





pp[nmax])

- 作成　　　　片瀬雅彦
- 作成時期　　2021年10月29日、11月27日