８章　モデル比較

8.1 イントロ

この章ではベイズ統計からみたモデル比較について述べる。パラメータに関して２つの仮説を比較するという枠組みの中で、ベイズ因子の概念を紹介する。

母集団の平均の仮説検定を行うという状況で、片側･両側検定におけるベイズ因子の計算について解説する。その後、2 つのベイズモデルで事前密度とサンプル密度を選べるという状況において、2 つのベイズモデルを比較するという一般化を行う。このケースではベイズ因子は２つのモデルの**周辺（限界？）密度(marginal densities)**の比率である。ベイズ因子の計算を 2 つの例を使って行う。野球の打撃のデータ解析では、”安定モデル”と季節によって打率に波のある”波有り”モデルとを比較したい。2 番目の例では２ x ２分割表の独立性に対するベイズ因子の計算を行う。

8.2 仮説の比較

ベイズ手法の尺度を紹介のために、Yをサンプルの分布である $f(y\,/\,\theta)$から観察し、以下の仮説を検定したいとする：

$$H_0: \theta \in \Theta_0, H_1: \theta \in \Theta_1$$

ここで Θ_0と Θ_1 はパラメータ空間内に仕切り(partition)を作る。もしも、きちんとした事前密度(proper prior density) g(θ)を割り当てるなら、2 つの仮説を事前オッズ比(prior odds ratio)によって事前条件(a priori)とする。

$$\frac{\pi 0}{\pi 1} = \frac{P(\theta \in \Theta_0)}{P(\theta \in \Theta_1)} = \frac{\int_{\Theta_0} g(\theta) d\theta}{\int_{\Theta_1} g(\theta) d\theta}$$

P164

データ Y＝y が観察された後で、パラメータの信頼は事後確率(posterior density)によって更新される。

$$g(\theta|y) \propto L(\theta) g(\theta)$$

ここで $L(\theta)$は尤度関数である。２つの仮説に関する信頼は、事後オッズ比(posterior odds ratio)で要約される：

$$\frac{p_0}{p_1} = \frac{P(\theta \in \Theta_0|y)}{P(\theta \in \Theta_1|y)} = \frac{\int_{\Theta 0} g(\theta|y) d\theta}{\int_{\Theta 1} g(\theta|y) d\theta}$$

ベイズの因子は仮説の事後オッズ比と事前オッズ比との比率である。

$$BF = \frac{\text{posterior odds}}{\text{prior odds}} = \frac{p_0 \,/\, p_1}{\pi_0 \,/\, \pi_1}$$

統計量 BFは仮説 Ho をサポートするデータによって与えられた証拠の計量である。仮説 Ho の事後確率(posterior probability)は、ベイズ因子と仮説の事前確率の関数として表現される：

$$P_0 = \frac{\pi_0 BF}{\pi_0 BF + 1 - \pi_0}$$

8.3 通常平均の片側検定

Berry(1996)の 14 章の例のなかで、著者は異なる体重計で測ったときの本当の体重に興味がある。我々は、その計測は平均がμ、標準偏差がδで正規分布していると仮定する。著者は 10 回体重を測り、以下を得た：182, 172, 173, 176, 176, 180, 173, 174, 179, 175（ポンド）。単純化のために、彼は体重計の精度を知っていて、δ ＝ 3 ポンドであるとする。

もし平均μが彼の本当の体重であり、彼の体重が 175 ポンドよりも重いことを評価しようとしているならば、やりたい仮説の検定は以下：

$$H_0: \mu \le 175, \qquad H_1: \mu > 175$$

著者には自分の体重に対する事前情報はないとすると、彼はμに平均 170 で標準偏差 5 の正規事前分布(normal prior)を与える。

$$\mu \text{ distributed } N(170, 5)$$

帰無仮説 H_0 の**事前オッズ(prior odds)**は以下で与えられる:

$$\frac{\pi_0}{\pi_1} = \frac{P(\mu \le 175)}{P(\mu > 175)}$$

我々は密度 $N(170, 5)$からの事前オッズを **pnorm** 関数で計算する。以下のアウトプットで、**pmean** と **pvar** はそれぞれ**事前平均(prior mean)**と μ の**事前分散(prior variance)**である。

```
library(LearnBayes)
pmean <- 170; pvar <- 25
probH <- pnorm(175, pmean, sqrt(pvar))
probA <- 1 - probH
prior.odds <- probH / probA
prior.odds
```

[1] 5.302974: 事前オッズ

つまり、帰無仮説は対立仮説より 5 倍確からしいことになる。

１０の体重データを R に入力し、サンプル平均$\bar{y}$と関連する標本分散(sample variance)である sigma2 = σ^2/nを計算する。

```
weights <- c(182, 172, 173, 176, 176, 180,
173, 174, 179, 175)
ybar <- mean(weights)
sigma2 <- 3^2 / length(weights)
```

良く知っている normal density/normal prior の更新公式を用いて、μ の 事後精度(posterior precision)（varianceの逆数）はデータとpriorの精度の合計である。

```
post.precision <- 1/sigma2 + 1/pvar
post.var <- 1/post.precision
```

平均μ の事後平均はウエイトをそれぞれの精度で重みづけしたサンプル平均と事前平均の合計である。

```
post.mean <- (ybar / sigma2 + pmean /
pvar) / post.precision
c(post.mean, sqrt(post.var))
```

[1] 175.7915058　0.9320547

μ の事後（分布）密度は N(175.79, 0.93)である。

この**正規事後密度?(normal posterior density)**を用いて、帰無仮説のオッズを計算する。

```
post.odds <- pnorm(175, post.mean,
sqrt(post.var))/(1 - pnorm(175,
post.mean, sqrt(post.var)))
post.odds
```

[1] 0.2467017: 帰無仮説のオッズ

よって帰無仮説を支持する**ベイズ因子(Bayes factor)**は

```
BF = post.odds / prior.odds
BF
```

[1] 0.04652139: ベイズ因子

事前確率(prior probabilities)とベイズ因子 BF から、帰無仮説の事後確率(posterior probability)を計算する：

P166

```
postH = probH * BF / (probH * BF + probA)
postH
```

[1] 0.1978835: 帰無仮説の事後確率

この計算から、著者は彼の体重が最大で 175 ポンドであるとは考えられないと結論する。

コメント [LE1]: 帰無仮説H_0: $\mu \le$ 175 が起こりえる事後確率が 19%と低い、いうことか？

このベイズによる手法と頻度主義の p 値の間には興味深い関係がある。既知の標準偏差δとともに、伝統的な帰無仮説の検定は検定統計量

$$z = \frac{\sqrt{n}(\bar{y} - 175)}{3}$$

にもとづく。

p 値は、標準正規変量(standard normal variate)が Z を超える確率である。R のアウトプットでは、p 値を **pnorm** 関数で計算する。

```
z = sqrt(length(weights)) *
(mean(weights) - 175)/3
1 - pnorm(z)
```

[1] 0.1459203

仮に平均 170、標準偏差 100 のとても平らである事前分布（flat prior）を使ったベイズ解析を繰

り返したとする。LearnBayes パッケージの関数 **mnormt.onesided** は、（1）テストされるべき平均値、標準事前分布の平均と標準偏差のパラメータ、（3）サンプルデータの値（サンプル平均、サンプルサイズ、既知のサンプルの標準偏差）をインプットして計算を実行する。

```
library(LearnBayes)
weights = c(182, 172, 173, 176, 176, 180,
173, 174, 179, 175)
data = c(mean(weights), length(weights),
3)
prior.par = c(170, 1000)
mnormt.onesided(175, prior.par, data)
```

$BF：ベイズ因子
[1] 0.1694947
$prior.odds：事前オッズ
[1] 1.008011
$post.odds：帰無仮説のオッズ
[1] 0.1708525
$postH：帰無仮説の事後確率
[1] 0.1459215

帰無仮説の確率はだいたいp値と同じであることに留意せよ。これは、粗い事前分布情報がパラメータに与えられた時のベイズ確率は、片側検定のp値と同じであることを示している。

8.4　Normal meanの両側検定

標準偏差がわかっている正規分布の平均値が、ある値と同じである、という仮説の両側検定を考える。前例につづいて、Berry さんは去年の体重が 170 ポンドであったことを知っていると仮定し、今年も 170 ポンドかどうかを心配しているとする。すると、帰無仮説 H_0としては、現在の彼の本当の体重μは 170 であるとする。対立仮説 H_1は、彼の体重は 170 よりも重いかまたは軽いとする。

事前分布の作成はユニークである、なぜなら帰無仮説では本当の体重μ値には**点の塊(a point mass)**が存在するからである。この例では、彼の体重は去年から変化していないという可能性があると著者は信じているから、彼はμ=170 であると宣言し、その確率を 0.5 とする。

次に、仮に帰無仮説 H_0 は正しくないとすると、著者は本当の体重μに信憑性のある値を与えなければならない。もしも彼の体重は去年から変化したとしても、今年の体重は 170 からは大きくかけ離れていないと考えるだろう。そこで平均 170 で標準偏差がτ (tau) の正規分布がμの**対立値（alternative values）**のための適したモデル (suitable model)となるだろう。

一般的に我々は、標準偏差がδだとわかっている時には、帰無仮説H_0：$\mu=\mu_0$を対立仮説H_1：$\mu\neq\mu_0$に対して検定する。平均がμ_0で標準偏差がτである**標準事前分布(normal prior distribution)**は、対立仮説H_1の下での意見を代表するために使われる。

この状況では、仮説 H をサポートするベイズ因子は以下のとおり与えられる

$$BF=\frac{\frac{n^{\frac{1}{2}}}{\sigma}\exp\left\{-\frac{n}{2\sigma^2}(\bar{y}-\mu_0)^2\right\}}{(\frac{\sigma^2}{n}+\tau^2)^{-1/2}\exp\left\{-\frac{1}{2(\frac{\sigma^2}{n}+\tau^2)}(\bar{y}-\mu_0)^2\right\}}$$

（τ:標準偏差；μ:体重）

前と同様に、もしもπ_0が$\mu=\mu_0$である帰無仮説H_0の事前確率(prior probability)ならば、H_0の事後確率は以下である：

$$\rho_0=\frac{\pi_0 BF}{\pi_0 BF+1-\pi_0}$$

ベイズ因子 BF を計算するために、対立仮説 H_1：$\mu\neq\mu_0$のもとでの**正規密度(normal density)**の標準偏差τをインプットしなければならない。もしも著者の体重は去年から**変化したのならば**、その変化はどのくらい大きければ変化したとみなしてよいのだろうか。標準偏差τを得る方法の一つは対立値μのレンジを考えることであり、そうすればこの正規分布の 95%レンジ 4τとして標準偏差が解決する。イメージとしては、著者は彼の現在の体重は 170 ポンドよりも 5 ポンド程度多いか少ないと考

えている。対立仮説の体重μのレンジは 175－165 ＝10 で、4τ＝10 とするとτ＝2.5 となる。

LearnBayse の関数 **mnormt.twosided** は、ベイズ因子と帰無仮説の事後確率（posterior probability of the null hypothesis）を計算する。関数 **mnormt.twosided** に代入するのは、テストされる平均μ_0、帰無仮説 H_0の事前確率π_0、事前標準偏差τ(prior standard deviation τ)、データの値（サンプル平均、サンプルサイズ、既知のサンプルの標準偏差）である。

P168

事前標準偏差τを見積もるのは困難なため、関数 **mnormt.twosided** はユーザーがそれらしいベクトル値を入力することを認めている。

以下に R コードを示す（τには 0.5, 1, 2, 4 を入れた）。

```
weights=c(182, 172, 173, 176, 176, 180,
173, 174, 179, 175)
data = c(mean(weights), length(weights),
3)
t = c(0.5, 1, 2, 4, 8)
mnormt.twosided(170, 0.5, t, data)
```

$bf

[1] 1.462146e-02 3.897038e-05 1.894326e-07 2.591162e-08 2.309739e-08

$post

[1] 1.441076e-02 3.896887e-05 1.894325e-07 2.591162e-08 2.309739e-08

事前標準偏差τのそれぞれの値（0.5, 1, 2, 4）ついて、**mnormt.twosided**は、仮説Hは正しいとする事後確率とμの値をとる、という仮説を支持して、ベイズ因子をそれぞれのτに対して返している。もしも著者が彼の体重μに対立する値を反映するために正規密度（170, 2）を用いれば、仮説μ=170 を支持するベイズ因子は 0.0000002 となる。この、彼の体重は変化していないとする事後確率(posterior probability)は 0.0000002 であり、これは著者の事前確率 0.5 よりもずっと低い。よって彼の現在の体重は 170 と同じではないと結論すべきである。

8.5 2 つのモデルの比較

仮説を比較するベイズアプローチは、2 つのモデルを比較することに一般化できる。まず、y はデータのベクトルを意味するとする、そしてθはパラメータだとする。するとベイジアンモデルはサンプル密度$f(y|\theta)$と事前密度(prior density) $g(\theta)$の詳細を含む。このモデルが与えられるとすれば、データの限界（marginal）または事前(prior)予測密度：

$$m(y) = \int f(y|\theta)g(\theta)d\theta$$

を計算することができる。

我々は 2 つのベイジアンモデルを比較したいと仮定する：

$M_0: y \sim f_1(y|\theta_0), \theta_0 \sim g_1(\theta_0),\ M_1: y \sim f_2(y|\theta_1), \theta_1 \sim g_2(\theta_1)$

ここではパラメータθの定義はモデル間で異なってもよい。そこでモデル M_0を支持するベイズ因子は 2 つのモデルデータの merginal density（限界密度）(または事前予測密度)の比率である。

$$BF = \frac{m_0(y)}{m_1(y)}$$

P169

もしもπ_0とπ_1がモデル M_0と M_1の事前確率を表すなら、モデル M_0の事後確率は以下である：

$$P(M_0|y) = \frac{\pi_0 BF}{\pi_0 BF + \pi_1}$$

marginal density（限界密度）を要約する単純な方法は 5 章 3 節のLaplaceの方法である。$\hat{\theta}$は事後モデルを表し、$H(\theta)$は対数事後密度(log posterior density)の**Hessian(second derivative matirx)**であるとする。すると事前予測密度(prior predictive density)は以下で要約される：

$$m(y) \approx (2\pi)^{d/2} g(\hat{\theta}) f(y|\hat{\theta}) |-H(\hat{\theta})|^{1/2}$$

ここで d はパラメータの数である。対数スケールでは

$$\log m(y) \approx \left(\frac{d}{2}\right)\log(2\pi) \quad +\log\left(g(\hat{\theta})f(y|\hat{\theta})\right) + (1/2)log|-H(\hat{\theta})|$$

いちどＲ関数によって積f(y|θ)g(θ)の対数が計算できれば、関数 **laplace** が適用でき、アウトプットの一部である **int** が log *m(y)*を与える。この方法を複数のモデルに用いれば、*m(y)*の値をベイズ因子の計算に使うことができる。

8.6　サッカーゴールのモデル

ベイズ因子の計算関数 **laplace** の使用を説明するため、サッカーの平均ゴール数に興味があるとする。ｎ回の試合のゴール数は $y_1 \ldots y_n$ とする。ゴールは比較的まれな出来事なので、**試合のゴール数 y_i は平均λの Poisson 分布型の分布**をするということがもっともらしいと考える。λに関する以下４つの主観的な**事前情報（priors）**を考える：

事前情報１． 結合γ事前情報(conjugate gamma prior)を式のλにあてる

$$g(\lambda) \propto \lambda^{\alpha-1}\exp\{-\beta\lambda\}, \lambda > 0,$$

$$\alpha = 4.57 \text{ かつ} \beta = 1.43$$

この事前情報は、チームの平均ゴール数は３でλの４分位点は 2.10 と 4.04 だとあなたが信じていることを表している。

事前情報２. シンメトリック分布に関する事前意見(prior opinion)を述べることは便利なことである。そこで logλは平均１で標準偏差 0.5 の正規分布だと仮定する。logλの事前四分位点は 0.66 と 1.34 となり、つまりλの事前四分位点は 1.94 と 3.81 である。事前情報１とこの事前情報は平均率λの位置についての類似の考えを反映している。

Ｐ１７０

事前情報３. この事前情報は logλが **N(2, 0.5)**と仮定する。率λ（prior quartiles for the rateλ）の事前四分位は 5.27 と 10.35 である。この事前情報は、チームのゴール数がとても多いと考えていることになる。

事前情報４. この事前情報は logλを N(1, 2)と仮定し、付随する rateλの四分位点は 1.92 と 28.5 とする。これはサッカーのゴールについての知識が乏しいことを反映したものである。

2006 年のシーズンで、あるチームについてゴール数を記録した。データは **soccergoals** に記録されている。λの尤度は、ポアソンモデルを仮定すると以下となる：

$$L(\lambda) = \frac{\exp(-n\lambda)\lambda^s}{\prod_{i=1}^n y_i!},$$

ここで$s = \sum_{i=1}^n y_i$である。我々のデータセットでは、n = 35, s = 57 である。図 8.1 が logλスケールの尤度と４つの事前情報を示す。事前情報１と２はかなり類似している。事前情報３は尤度とは対立しており、４は非常に平坦な形をしている。

laplace 関数を使うために、我々は対数事後分布(log posterior)を定義するための短い関数を記載しなければならない。はじめの関数 **logpoissgamma** はポアソンサンプリングとガンマ事前分布(a gamma prior)を用いた対数事後分布(log posterior)を計算する。その後はいつものとおり、λを変換して実際の値のパラメータ θ = logλとする。関数の議論は **theta** と **datapar** であり、ベクトル **data** とガンマ事前分布（gamma prior）である **par** である。Ｒの関数である **dgamma** を尤度と事前分布の両方の計算に使う。

```
logpoissgamma <- function(theta,
datapar)
{
y <- datapar$data
npar <- datapar$par
lambda <- exp(theta)
loglike <- log(dgamma(lambda, shape =
sum(y)+1, rate = length(y)))
logprior <- log(dgamma(lambda, shape =
npar[1], rate = npar[2]) * lambda)
return(loglike + logprior)
}
```

同様に、関数 **logpoissnormal** を、ポアソンサンプリング？と正規事前分布(normal prior)?のための log λ の対数事後分布を計算するために記述する。（意味解読できず翻訳失敗）。関数は R の **dgamma** と **dnorm** を用いる。

```
logpoissnormal = function(theta,
datapar)
{
y <- datapar$data
npar <- datapar$par
lambda <- exp(theta)
loglike <- log(dgamma(lambda, shape =
sum(y)+1, scale = 1/length(y)))
logprior <- log(dnorm(theta, mean =
npar[1], sd = npar[2]))
return(loglike + logprior)
}
```

P171

我々ははじめにデータファイル soccergoals をロードする。soccergoals の中身には変数 goals があり、attach コマンドによって使えるようになる。 4 つの事前分布の 1 つずつのうち、我々は関数 laplace を用いて事後分布を要約する。もしも関数のアウトプットが fit であれば、fit$mode は事後のモード(posterior mode)、fit$var は事後分散(posterior variance)と関連する推定であり、fit$int は log *m(y)*の推定である。

＜＝　Fig8.1

```
data(soccergoals)
attach(soccergoals)
datapar <- list(data = goals, par=c(4.57,
1.43))
fit1 <- laplace(logpoissgamma, 0.5, 10,
datapar)
datapar <- list(data = goals, par = c(1,
0.5))
fit2 <- laplace(logpoissnormal, 0.5, 10,
datapar)
datapar <- list(data = goals, par=c(2,
0.5))
fit3 <- laplace(logpoissnormal, 0.5, 10,
datapar)
datapar <- list(data = goals, par=c(1,
2))
fit4 <- laplace(logpoissnormal, 0.5, 10,
datapar)
```

P172

我々は 4 つの事前情報に対する 4 つのモデルについて、事後モード、事後標準偏差、log marginal densities（対数限界密度？）を表示する。

```
postmode <- c(fit1$mode, fit2$mode,
fit3$mode, fit4$mode)

postsd <- sqrt(c(fit1$var, fit2$var,
fit3$var, fit4$var))
logmarg <- c(fit1$int, fit2$int,
fit3$int, fit4$int)
cbind(postmode, postsd, logmarg)
```

	postmode	postsd	logmarg
[1,]	0.5247821	0.1274428	-1.502965
[2,]	0.5207796	0.1260714	-1.255170
[3,]	0.5825327	0.1224715	-5.076322
[4,]	0.4899378	0.1320168	-2.137214

log *m(y)*の値の利用によりベイズ因子を比較することで違うモデルを比較することができる。もし我々が事前 γ（4.57, 0.7）を λ に用いるか、正規分布(1, 0.5)を log λ に用いるのは問題になるか？この回答には、我々はベイズ因子を計算して、Prior2 を Prior1 よりサポートすることができる。

$$BF_{21} = \frac{m_2(y)}{m_1(y)} = \exp(-1.255170 + 1.502965)$$
$$= 1.28$$

正規事前情報(normal prior)へのサポートはわずかである　―　これは図 8.1 から納得できる。なぜなら Prior2 はわずかに尤度関数に近い。Prior2 と Prior3 を比較すると、ベイズ因子は Prior2 を支持して

$$BF_{23} = \frac{m_2(y)}{m_3(y)} = \exp(-1.255170 + 5.076322)$$
$$= 45.6$$

これは Prior2 を大きく支持することを意味する。実際、尤度と Prior3 の位置は互いに離れていて、データと prior と $m_3(y)$の小さな値の対立を意味している。Prior2 と Prior4 を比較すると、Prior2 を支持するベイズ因子は

$$BF_{24} = \frac{m_2(y)}{m_4(y)} = \exp(-1.255170 + 2.137214)$$
$$= 2.42$$

概して事前情報の限界確率(marginal probability for a prior)は事前密度(prior density)がより散乱するに伴って減少する。

8.7 野球の打者は打撃に波があるか？

P173

スポーツでは選手やチームの波のある活躍を見る。野球の打者の成功は、平均打率やヒットの率である。シーズン中に選手の調子が上がっている次期では打率が非常に上がり、調子が悪い時期は打率が下がるものだ。選手の調子の波はさまざまである。ここでの疑問はこの波のあるデータは、選手の能力の違いを伝えるのかどうかである。

野球では選手はシーズンごとに打席に立つチャンスがあり、これを我々は打席数（at-bats）と呼ぶ。打数は、ヒットかアウトに分けられる。あるシーズンの打数をN個の期間に分割するとする。i = 1, ...N であるとき、i 番目の期間にヒットを打つ確率（打率）を p_iとする。

もしも選手が本当に一貫したプレーをするならば、打率はは期間を通して一定である。本書ではこれを nonstreaky model　（”波無しモデル”）M_0と呼ぶ

M0: $p_1 = \ldots = p_N = p$.

モデルを完結するためには、共通の確率(common probability)pに統一の事前確率（uniform prior）(0, 1)を与える。

一方でもしも選手の打率に本当に波があれば、ヒットの確率 pi はシーズン中に変化する。この打率の変化をモデル化する簡単な方法は、$p_1,\ldots,p_N$ はベータ密度からランダムサンプルしたと仮定することである：

$$g(p) = \frac{1}{B\big(K\eta, K(1-\eta)\big)} p^{K_{\eta-1}}(1-p)^{K(1-\eta)-1}, 0 < p < 1.$$

密度gでは、η (eta)は平均でKは精密パラメータ(precision parameter)である。我々はこの”波有りモデル”をパラメータKによってインデックス化できる：

M_K : $p_1, \ldots, p_N$ iid beta $(K\eta, K(1-\eta))$.

個のモデルでは統一事前確率(uniform prior)を平均パラメータη (mean parameter η)の上に置き、ランダム効果の分布(random effects distribution)の位置（location）についての知識がほとんどないということを反映している。正確パラメータ K が無限大に近づくために、”波有りモデル”M_Kは”波無しモデル”M_0に近づくことを注記しておく。

モデルM_0 とM_Kを比較するためには、関連する **marginal densities**周辺密度？を計算する必要がある。モデルM_0ではヒットの数$y_1,\ldots,y_N$は独立していてy_iは二項分布（ni, p）である。pは一様分布(0, 1)とすると、周辺密度は以下である：

$$m_0(y) = \int \prod_{i=1}^{N} \binom{n_i}{y_i} p^{y_i}(1-p)^{ni-yi} dp = \prod_{i=1}^{N} \binom{n_i}{y_i} B\left(\sum_{i=1}^{N} y_i + 1, \sum_{i=1}^{N} (n_i - y_i) + 1\right).$$

もう一つの”波有りモデル”のもとでは、周辺密度は

P174

以下で与えられる：

$$m_K(y)$$
$$= \int \prod_{i=1}^{N} \binom{n_i}{y_i} {p_i}^{y_i}(1$$
$$- p)^{ni-yi} \frac{{p_i}^{K\eta-1}\ \ (1-p_i)^{K(1-\eta)-1}}{B(K\eta, K(1-\eta))} dp_{1\ldots}dp_N$$
$$= \prod_{i=1}^{N} \binom{n_i}{y_i} \int_0^1 \frac{\prod_{i=1}^{N} B(y_i + K\eta, n_{i-}y_i + K(1-\eta))}{B(K\eta, K(1-\eta))^N} d\,\eta$$

”波有りモデル”H_K をサポートして”波無しモデル”H_0 と比較するベイズ因子は以下で与えられる：

$$B_K = \frac{m_K(y)}{m_0(y)}$$

$$= \frac{1}{\mathrm{B}(\sum \mathrm{y_i} + 1, \sum(\mathrm{n_i} - \mathrm{y_i}) + 1)} \int_0^1 \frac{\prod_{i=1}^{N} B(y_i + K\eta, n_{i-}y_i + K(1-\eta))}{B(K\eta, K(1-\eta))^N} \mathrm{d}\eta$$

我々は関数 laplace をベイズ因子 B_K の積分計算に用いる。はじめに積分の中の変数 η を変換して真に価値のある変数 θ =log(η /(1- η))にする。R 関数 lbeta はベータ関数の対数を計算するのでこれを用いて bfexch を定義し、積分の対数を計算する。この関数へは theta と data と K コンポーネントを持つリスト datapar（列 y と n のマトリクス）を代入する。

```
bfexch = function(theta, datapar) {
y = datapar$data[,1]; n = datapar$data[,
2]; K = datapar$K
eta = exp(theta)/(1+exp(theta))
N = length(y)
z = 0*theta;
for (i in 1:N)
   z = z+lbeta(K * eta + y[i], K*(1-eta)
+ n[i] - y[i])
z  =  z-N*lbeta(K*eta,  K*(1-eta))  +
log(eta * (1-eta))
z = z - lbeta(sum(y) + 1, sum(n-y)+1)
return(z)
}
```

とある値、たとえば K0、におけるベイズ因子 BK を計算するには、我々は代入関数 bfexch とともに関数 laplace を用いる。スタートとして η=0 からはじめて Newton 法を 10 回繰り返し、値 K0 を用いて datapar をリストする。

```
s = laplace(bfexch, 0, 10, list(data =
data, K = K0))
```

コメント [LE2]: なぜか関数が機能しない！

リスト s は laplace 関数のアウトプットである。要素 s$int はベイズ因子の対数値、log$B_K$、を推定した値である。

P175

この方法を描写するために我々はヤンキースの Derek Jeter の２００４年の打撃データを使用する。彼は 2004 年の前半にスランプに陥った。

ジータが出場した 155 試合のデータを集計した。よって N= 155 とする。打者は 1 ゲームで 4－5 回しか打席に立たないため、波を検出するのは困難である。そこで我々はデータを 5 ゲームごとにまとめた。オリジナルのデータは **jeter2004** にまとめてある。以下の R コードでは打撃データを読み込みそれから regroup 関数を用いて 5 ゲームごとにまとめたデータを作成した。

```
data(jeter2004)
attach(jeter2004)
data = cbind(H, AB)
data1 = regroup(data, 5)
```

マトリクス data1 はグループ化した打席データ(yi, ni), i = 1,...,31,　ここで i 番目の期間の中での打席数 n_i 回のうちジータが売ったヒットの数が y_i である（表 8.1）。

我々は log$K$ の値の数列のために、laplace 関数と bfexch で定義された log 積分の定義を用いてベイズ因子を計算する。この例ではベクトル logK は値 log(K)= 2,3,4,5,6 を持ち、ベクトル logBF はベイズ因子 logB_K の値を持つ。マトリクスは log$K$、K、logB_k の値をとる。

```
logK = seq(2,6)
logBF = 0*logK
for (j in 1: length(logK)){
s = laplace(bfexch, 0, 10, list(data =
data1, K = exp(logK[j])))
```

```
logBF[j] = s$int
}
cbind(logK,      exp(logK),      logBF,
exp(logBF))
```

```
    logK              logBF
[1,]    2   7.389056 -2.9441184 0.05264845
[2,]    3  20.085537  1.0482050 2.85252615
[3,]    4  54.598150  1.4380139 4.21232144
[4,]    5 148.413159  0.8160940 2.26164853
[5,]    6 403.428793  0.3538978 1.42460960
>
```

p176

アウトプットから見えるのは $\log K = 4$ の値は関係するベイズ因子である B_K = 4.21 というデータによって最も支持されているということである。この特に波ありモデルは”波無しモデル”よりも 4 倍もっともらしい。このことはジータは打撃に波があることを示している。

8.8　2 x 2 分割表の独立性の検定

統計学の基本的な問題は、二つのカテゴリカルな計測同士の関係を探ることである。これを描写するために Moore(1995)にある、ノースカロライナ州立大学が実施した化学工学専攻の生徒が履修したあるコースでの成績の例合を考える。研究者は課外活動に費やした時間とコースの成績との関係に興味があった。データは 119 人の生徒から 2 つのカテゴリについて採取した。結果は分割表 8.2 のとおりである。

表８．２

学生の課外活動への参加と成績との関係を学ぶために、独立性の検定を行う。通常の非ベイジアンアプローチでの独立性の検定では、Pearson の $\chi 2$ 乗統計量に基づいて行われる、これは観察されたカウントと期待されたカウントを独立モデルの下で比較して行うものである。R では、カウントのテーブルを読み込み、関数 chisq.test によって独立仮説をテストする：

P177

```
data = matrix(c(11, 9, 68, 23, 3, 5),
c(2,3))
data
     [,1] [,2] [,3]
[1,]   11   68    3
[2,]    9   23    5
chisq.test(data)
      Pearson's Chi-squared test

data:   data
X-squared = 6.9264, df = 2, p-value = 0.03133

Warning message:
In chisq.test(data) : Chi-squared approximation may be
incorrect
```

ここでは p 値は約 0.03 であり、成績はは課外活動に費やした時間に関係があるという証拠の一部となる。

ベイズの視点からは、2 つのモデルが可能である。モデル M_I は 2 つのカテゴリカル変数は独立しているといもので、モデル M_D は 2 つの変数はある意味で依存しているというものである。ベイズモデルを描写するために、これらのデータはターゲットの母集団からのランダムサンプルを表していると仮定し、テーブルにあるカウント数は比率値が表 8.3 にしめしたような多項分布であるとする。依存モデル M_D のもとでは、比率値 p11,...,p23 は合計が 1 になるあらゆる値をとることができ、我々は事前密度（prior density）は空間上に一様に分布すると仮定する。

独立モデル M_I のもとでは、表内の比率は表 8.4 にしめすような周辺分布{p_{1+}, p_{2+}}と{p_{+1}, p_{+1}, p_{+3}}によって決定される。ここでは未知のパラメータは学生が異なった課外活動レベルを持っている比率と異なった成績を取っている比率である。我々は、知っているこの 2 セットの比率{p_{i+}}{p_{+j}}は独立であると仮定し、それぞれのセットに一様な密度(<u>a uniform density over all possible values</u>)を与える。

我々は 2 つのモデルを定義した：依存モデル MD は多項比率(multinomial proportions)が一様に分布しており、独立モデル MI は多項比率は独立な構

成であり周辺比率(marginal proportions)は振り分けられた独立な一様事前分布？(uniform priors)に等しい。

P178

依存モデルよりも独立モデルを支持するベイズ因子は、以下のように見えるだろう：

$$BF = \frac{D(y+1)D(1_R)D(1_C)}{D(1_{RC})D(y_R+1)D(y_C+1)}$$

y は数のマトリクス、yR は行トータルのベクトル、yC は列トータルのベクトル、1R は長さＲのベクトル、D(v)は以下で定義される Dirichlet 関数である：

$$D(v) = \prod \Gamma(v_i)/\Gamma(\sum v_i)$$

R関数ctableはこの２ x ２分割表のベイズ因子を計算する。ユーザーは確率のマトリクスのために事前パラメータのマトリクスであるaをインプットする。マトリクスaを手に入れることによって、依存モデルのもとでは一様事前情報(a uniform prior)を{p_{ij}}に与え、一様事前情報(uniform priors)を{p_{i+}}と{p_{+j}}に与える。この問題のアウトプットはベイズ因子の値である。ここではＢＦ値は 1.66 であり、これは独立に反対するには控えめな支持である。

```
 a = matrix(rep(1,6), c(2,3))
 a
      [,1] [,2] [,3]
[1,]     1    1    1
[2,]     1    1    1
> ctable(data, a)
[1] 1.662173
```

我々は分割表における”uniform”と”independence”モデルを比較している。この方法への批判としては、我々は本来は”uniform”な代替モデルには興味はないかもしれないことである。おそらく我々は独立性”independence”と、セルの確率が“独立に近い(close to independence)”モデルとを比較したいのだろう。そのようなモデルはAlbert and Gupta(1981)によって提唱された。分割表の確率{p_{ij}}は a conjugate Dirichlet distribution(共役デリヒレット分布)

$$g(p) \propto \prod p_{ij}{}^{K\eta_{ij}-1}$$

であると仮定する。ただし事前平均(prior means){η_{ij}}は独立配置

$$\eta_{ij} = \eta_i{}^A \eta_j{}^B$$

を満たす。

P179

prior meansの構成は表 8.5 の例にあるとおりである。margins {$\eta_i{}^A$}と{$\eta_j{}^B$}のprior meansのベクトルは一様分布が割り当てられる。

このモデルはM_Kとラベルされる、というのはDirichlet precision parameter(デリヒレット精度パラメータ？) であるKによってインデックス化されるからである。Kは無限大に近づくため、モデルは独立仮説MIに近づく。そこではmarginal probabilities（限界確率？）は一様分布をする。

表 8.5 : 独立に近いモデルの下でのテーブルのセル確率の事前平均（prior means)。

独立モデル M_Iよりも“独立に近い”モデル M_Kをサポートするベイズ因子(Bayes factor)は以下で与えられる：

$$BF_K = \frac{1}{D(y_R+1)D(y_C+1)} \int \frac{D(K\eta^A\eta^B + y)}{D(K\eta^A\eta^B)} d\eta^A d\eta^B$$

ここで$K\eta^A\eta^B + y$は値{$K\eta_i{}^A\eta_j{}^B + y_{ij}$}のベクトルであり、その積分は・marginal prior means である$\eta^A = \{\eta_i{}^A\}$と$\eta^B = \{\eta_j{}^B\}$にとって代わられる。

素直なベイズ因子の計算方法は重要サンプリン

グ(importance sampling？)によるものである。ベイズ因子は以下の積分で表現できる：

$$BF_K = \int h(\theta)d\theta$$

ただし$\theta = (\eta^A, \eta^B)$である。

積分は密度 $g(\theta)$（ただし g は簡単にシミュレートできる）によって要約できると仮定すると、積分は以下と書ける：

$$BF_K = \int \frac{h(\theta)}{g(\theta)} g(\theta)d\theta$$

積分は以下として近似できる：

$$BF_K \approx \frac{\sum_{j=1}^{m} h(\theta_j)/\, g(\theta_j)}{m}$$

ただし $\theta_1,\ldots \theta_m$は独立にシミュレートされた$g(\theta)$からの draws（意味不明？）である。この **importance sampler estimate?**のシミュレーションの標準誤差は以下で与えられる：

P180

$$se = standard\ deviation(\{\frac{h(\theta_j)}{g(\theta_j)}\})/\sqrt{m}$$

この例では、Kは無限大に近づくので、marginal prior means η^Aとη^B のベクトルの事後分布?(posterior)は独立なものとして以下で示される：

$$\eta^A \text{distributed Dirichlet}(y_R + 1), \eta^B \text{distributed Dirichlet}(y_C + 1)$$

ここで、ベクトルηに関する Dirichlet 分布は$\prod \eta_i^{a_i - 1}$に比例する密度を持つ。この密度は importance sampler としては便利な選択である、といのも Dirichlet 分布からの draws をシミュレートするのが簡単だからである。

importance sampling のアルゴリズムを用いて、関数 bfindep はこの対立する”close to independence”モデルを用いてベイズ因子を計算する。データマトリクス y、Dirichlet 精度パラメータ K、シミュレートしたサンプル m をインプットする。アウトプットは bf（ベイズ因子の値）と nse（計算されたＢＦ値の標準偏差の推定値）の２つの要素からなるリストである。

以下のＲのインプットでは、我々はlogK値の数列のベイズ因子を計算する。そのアウトプットはlogベイズ因子の値と、logKの値のベイズ因子を出す。図 8.2 はlogKの関数としての対数ベイズ因子と10,000 回シミュレーションの結果を示している（ＲのSemiParパッケージにある関数spmを用いて推定エラーを取り除いた)。このベイズ因子の最大値は 2.3 で、独立モデルの近所にある対立モデルへのサポートを示唆している。

```
logK = seq(2, 7, by= 0.2)
logBF = 0*logK
for(j  in  1:length(logK))  +  {x  =
bfindep(data,  exp(logK[j]),  100000);
logBF[j] = log(x$bf)}
cbind(logK, logBF, exp(logBF))
```

```
      logK     logBF
 [1,]  2.0 -1.7393843 0.1756285
 [2,]  2.2 -1.0494239 0.3501394
 [3,]  2.4 -0.8431378 0.4303580
 [4,]  2.6 -0.1796282 0.8355808
 [5,]  2.8 -0.2013881 0.8175951
 [6,]  3.0  0.2131446 1.2375636
 [7,]  3.2  0.3621546 1.4364210
 [8,]  3.4  0.7901360 2.2036961
 [9,]  3.6  0.6935077 2.0007212
[10,]  3.8  0.7545760 2.1267096
[11,]  4.0  0.7593722 2.1369342
[12,]  4.2  0.8172366 2.2642341
[13,]  4.4  0.8019619 2.2299114
[14,]  4.6  0.7977221 2.2204772
[15,]  4.8  0.7688568 2.1572986
[16,]  5.0  0.7282715 2.0714969
[17,]  5.2  0.6791026 1.9721072
[18,]  5.4  0.6143356 1.8484281
[19,]  5.6  0.5609184 1.7522811
[20,]  5.8  0.4966223 1.6431618
[21,]  6.0  0.4359152 1.5463776
[22,]  6.2  0.3781691 1.4596098
[23,]  6.4  0.3257384 1.3850530
[24,]  6.6  0.2782086 1.3207617
[25,]  6.8  0.2351368 1.2650818
[26,]  7.0  0.1988317 1.2199766
```

8.9 書籍紹介