

贝叶斯统计学基础

9. 层级贝叶斯数据分析

戴俊毅

研究员/长聘副教授



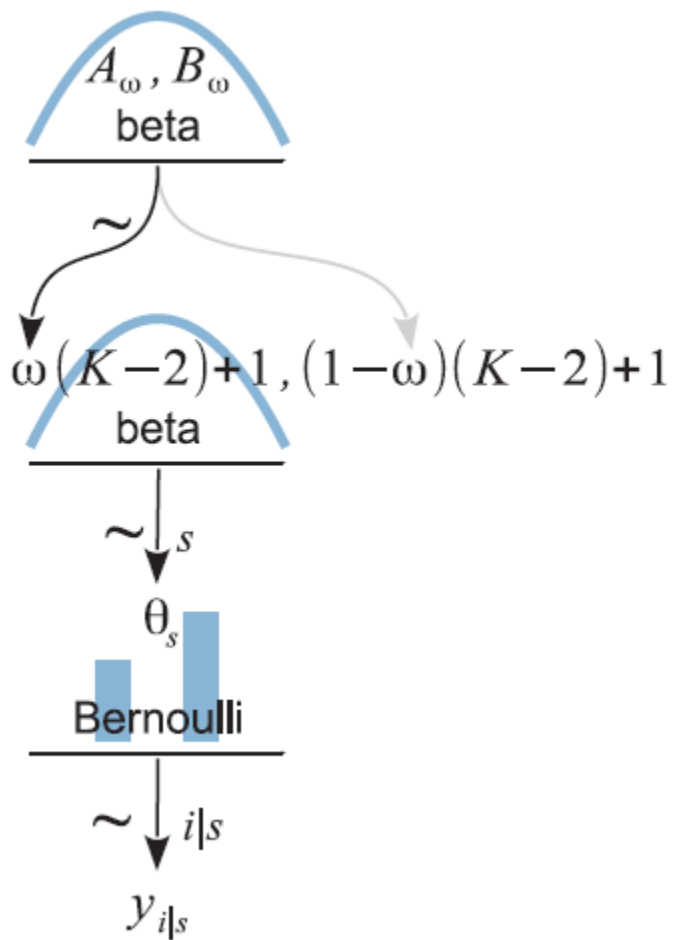
基本思路

- ✧ 在有些情况下，需要分析的数据涉及不完全同质的个体或者区组。
例如同一班级里的不同学生或者同一学校中的不同班级。
- ✧ 这些研究对象（即学生、班级、学校）构成了不同的层次或者层级，而且，不同层级的元素会以特定的方式联系在一起。
- ✧ 因此，进行数据分析时，既要考虑到同层级元素间的差异，也要考虑到位于不同层级的元素间的联系。

示例

- ✧ 假定我们关心的是某个班级中各个学生通过某一测试的可能性，且不同学生的通过可能性存在差异
- ✧ 我们可以进一步假定
 1. 不同学生的通过可能性 θ_s 的先验分布是一个beta分布
 2. θ_s 的先验beta分布的众数 ω 未知，且先验地满足0到1之间的均匀分布
 3. θ_s 的先验beta分布的参数 α 和 β 的和已知为20
 4. 每个学生 s 参加了多次测试，每次测试结果为通过（1）或者未通过（0）

图例



1. $\omega \sim \text{beta}(1, 1)$
2. $\theta_s \sim \text{beta}(\omega(20 - 2) + 1, (1 - \omega)(20 - 2) + 1)$
3. $y_{i|s} \sim \text{bern}(\theta_s)$

层级模型性质

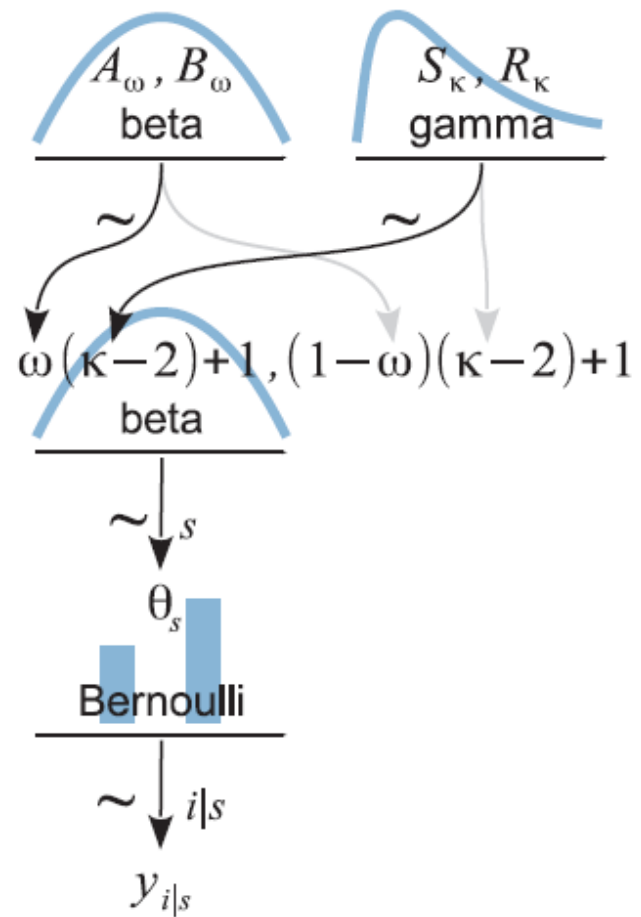
- ✎ 对于每个个体参数的估计，不仅受到与该参数直接关联的数据的影响，也会受到其他所有数据的影响。
- ✎ 对于下层参数的估计，会朝着上层分布的众数靠拢，这一现象被称为收缩(shrinkage)。
- ✎ 由于存在收缩现象，对于下层参数的估计往往有集中倾向，且较少地受到极端数据的影响。



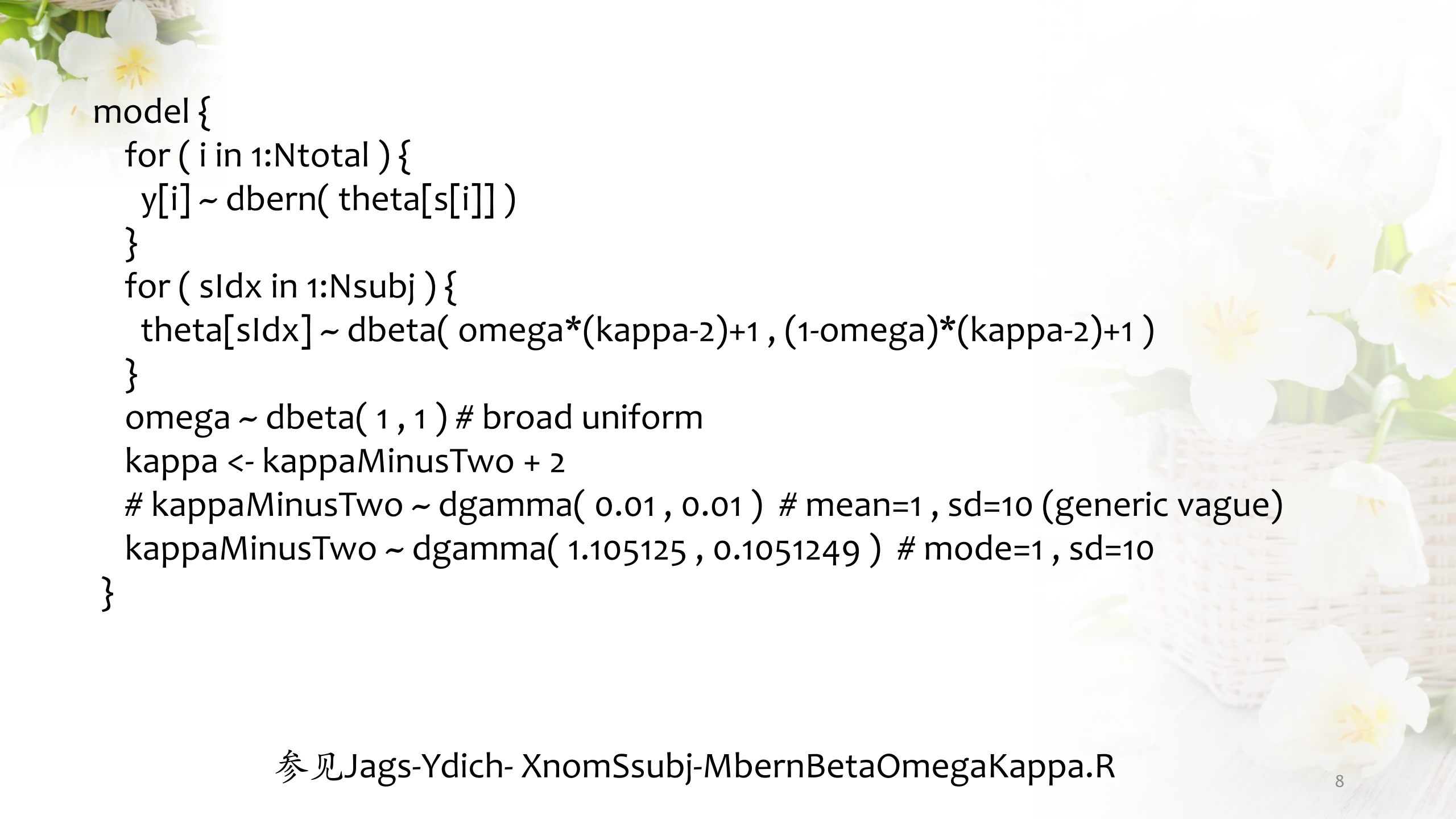
层级模型的JAGS实现方式

✧ 就JAGS（或者其他MCMC软件）而言，进行层级贝叶斯分析和普通贝叶斯分析的方式本质上是一样的，只需在模型设定代码中加入针对不同层次的表达式即可。

示例



DBDA(E2), 图9.7



```
model {  
  for ( i in 1:Ntotal ) {  
    y[i] ~ dbern( theta[s[i]] )  
  }  
  for ( sldx in 1:Nsubj ) {  
    theta[sldx] ~ dbeta( omega*(kappa-2)+1 , (1-omega)*(kappa-2)+1 )  
  }  
  omega ~ dbeta( 1 , 1 ) # broad uniform  
  kappa <- kappaMinusTwo + 2  
  # kappaMinusTwo ~ dgamma( 0.01 , 0.01 ) # mean=1 , sd=10 (generic vague)  
  kappaMinusTwo ~ dgamma( 1.105125 , 0.1051249 ) # mode=1 , sd=10  
}
```

参见Jags-Ydich- XnomSsubj-MbernBetaOmegaKappa.R

治疗性接触实验数据分析

问题：治疗性接触的实践者是否的确能够感受到病患的能量场

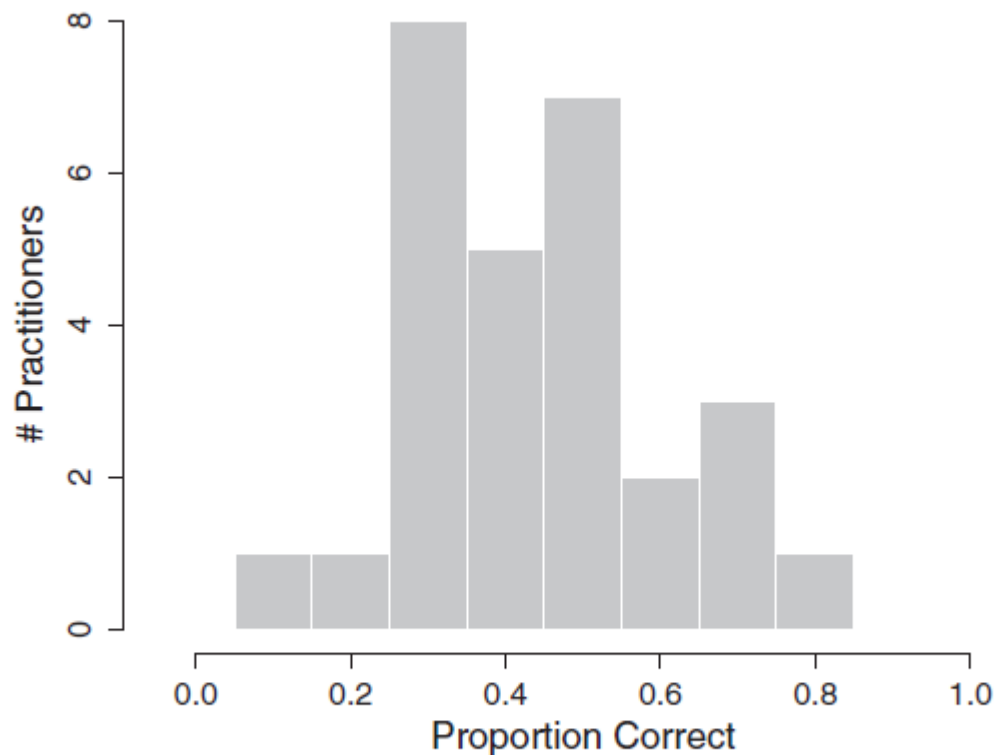


Figure 9.9 Data from the therapeutic touch experiment of Rosa et al. (1998). Histogram of proportion correct for the 28 practitioners.



代码

```
# Read the data file:

myData = read.csv("TherapeuticTouchData.csv")

# Load the relevant model functions into R's working memory:

source("Jags-Ydich-XnomSsubj-MbernBetaOmegaKappa.R")

# Generate the MCMC chain:

mcmcCoda = genMCMC( data=myData , sName="s" , yName="y" ,

numSavedSteps=20000 , thinSteps=10 )
```



代码

- ⌘ # Display diagnostics of chain, for specified parameters:
- ⌘ `diagMCMC(codaObject=mcmcCoda , parName="omega")`
- ⌘ `diagMCMC(codaObject=mcmcCoda , parName="kappa")`
- ⌘ `diagMCMC(codaObject=mcmcCoda , parName="theta[1]")`
- ⌘ # Get summary statistics of chain:
- ⌘ `smryMCMC(mcmcCoda ,`
- ⌘ `compVal=0.5 , diffIdVec=c(1,14,28), compValDiff=0.0)`
- ⌘ # Display posterior information:
- ⌘ `plotMCMC(mcmcCoda , data=myData , sName="s" , yName="y" ,`
- ⌘ `compVal=0.5 , diffIdVec=c(1,14,28), compValDiff=0.0)`

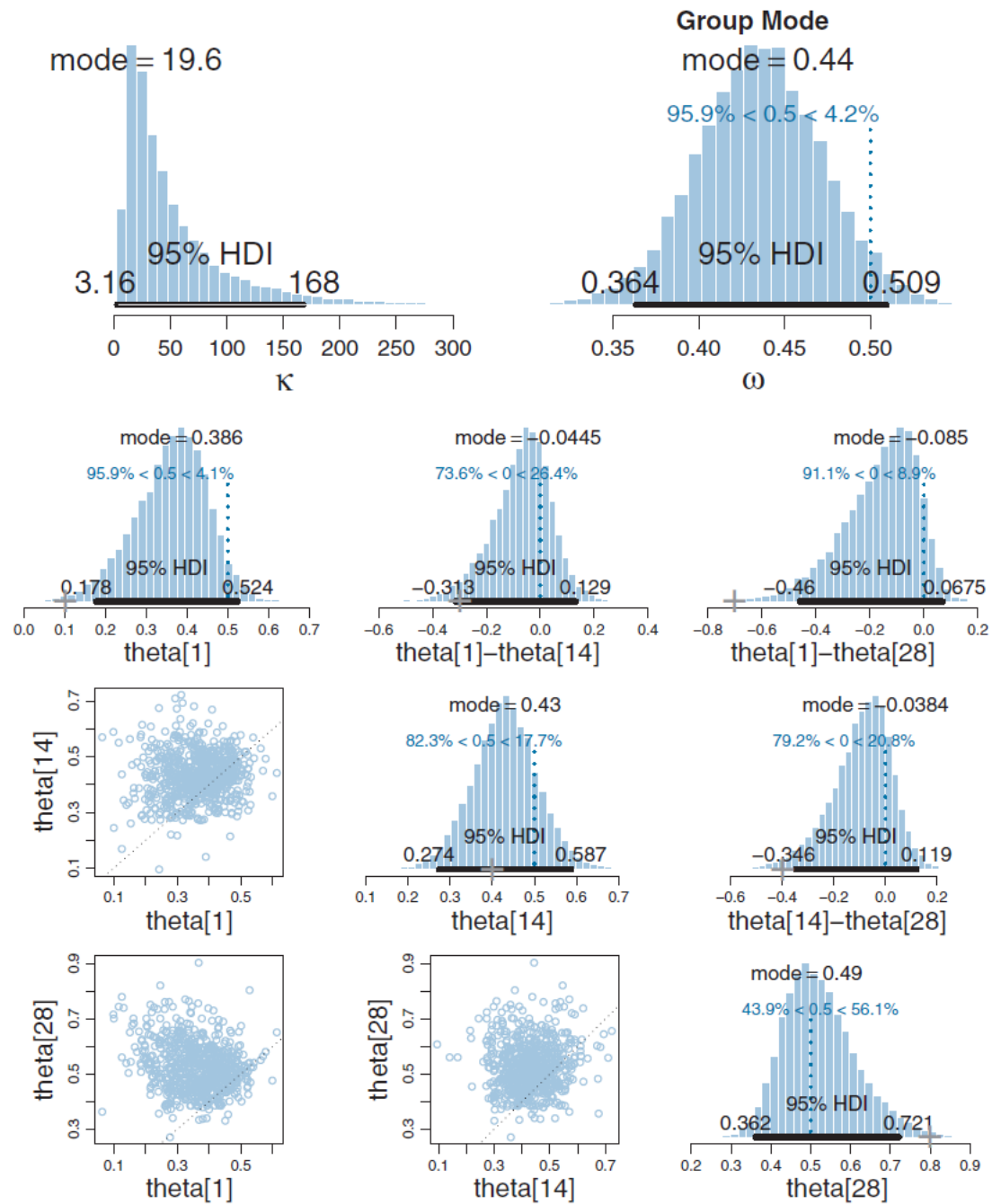


Figure 9.10 Marginal posterior distributions for the therapeutic touch data.

