Berger Ranking Method

Cora Allen-Coleman

9/11/2017

MCMC simple rank from Berger & Deely 1988 paper on ranking methods

GOAL: select the best hitter from the group (largest θ_i)

METHOD: calcuate the posterior probability that each θ_i is the largest

METHOD DETAIL:

Bayesian approach, with a selection of different priors: exchangable, nonexchangeable, informative, and noninformative. I use the exchangeable prior here.

Question 1: Are All the Means Equal?

Before ranking, we need to establish that these means are not all equal.
H_0: all means are equal H_A: means not equal PosteriorProbH_{0_i}: posterior probability of H_0
PriorProbH_{0_i}: prior probability of H_0
Bayes Factor: ratio of posterior odds or prior odds. Can be interpreted as the odds for H_0 provided by the data

$$\text{Bayes Factor } = \frac{\text{PosteriorProb}_{H_{0_i}}}{1 - \text{PosteriorProb}_{H_{0_i}}} * \frac{1 - \text{PriorProb}_{H_{0_i}}}{\text{PriorProb}_{H_{0_i}}}$$

Likelihood that all θ_i are equal: Are each of these sample proportions from the same binomial distribution? 12 joint binomials using the 12 known σ_i?

Question 2: What is the Probability that θ_i is the largest?

If H_0 is false, then we now want to find the largest θ_i.
PosteriorProbLargestH_{A_i}: posterior probability that this θ_i is largest, given that H_0 is false. This doesn't depend on the prior.

$$Pr(\theta_i \text{ is largest}|data, H_A) = \int_0 \int_R \int_i \Pi_{i \neq j} [\Phi([w_i(\theta_j) - u_i]/\sqrt{(}V_i)) - \Phi([v_i(\theta_j) - u_i]/\sqrt{(}V_i))] * \pi_j^*(\theta_j) * \pi_{2,1}(\beta) * \pi_{2,2}(\sigma_\pi^2|x) d\theta_j d\beta$$

##Example Data: Batting Averages observed batting averages are sample proportions from binomial distributions. i = # of players to rank x_i = sample mean of player i
θ_i = true mean of player i
σ_i = variance of the sample mean (known)

```
batData <- read.table("battingAverages.txt", header = T)
batData$sd <- sqrt(batData$X1000variance/1000)
```

Question 1: Are the Means Different?

We'll assume we've proven H_0 to be false, then move on to question 2.

Question 2: Which Mean is Largest?

Computing the Likelihood that Player i is ranked first

for now, just done for one player, but eventually we'll do this for all 12 players in batData

```
likelihood <- function(param){
  ##returns logs
    a = param[1]
    b = param[2]
    batData$sd = param[3]

    pred = a*x + b
    singlelikelihoods = dnorm(y, mean = pred, sd = batData$sd, log = T)
    sumll = sum(singlelikelihoods)
    return(sumll)
}
```

Prior

```
prior <- function(param){
  ##returns log
    a = param[1]
    b = param[2]
    sd = param[3]
    aprior = dunif(a, min=0, max=10, log = T)
    bprior = dnorm(b, sd = 5, log = T)
    sdprior = dunif(sd, min=0, max=30, log = T)
    return(aprior+bprior+sdprior)
}
```

Likelihood Plot

```
# Example: plot the likelihood profile of the slope a
slopevalues <- function(x){return(likelihood(c(x, trueB, trueSd)))}
slopelikelihoods <- lapply(seq(3, 7, by=.05), slopevalues )
plot (seq(3, 7, by=.05), slopelikelihoods , type="l", xlab = "values of slope parameter a", ylab = "Log
```

Posterior

```
posterior <- function(param){
  return (likelihood(param) + prior(param)) ##returns logarithm
}
```

Bayes Factor

```
BayesFactor <- (P_0/(1-P_0))*((1-prior)/prior)
```

RANKING

Extension: explore this: "An interesting sidelight to the development is the presentation of a closed-form solution for testing Ho: $01 = 02$ versus Hi: $01 < 02$ versus H2: $01 > 02$, when the treatments are judged to be a priori exchangeable"