

admissions

Chris Parrish

June 9, 2016

Contents

admissions	1
data	1
model	1
male advantage	2
postcheck	3
model with unique intercept for each dept	4
stan	5

admissions

reference: McElreath, Statistical Rethinking, chap 10, p.304

admissions

data

```
## R code 10.22
library(rethinking)
data(UCBadmit)
str(UCBadmit)

## 'data.frame':  12 obs. of  5 variables:
## $ dept          : Factor w/ 6 levels "A","B","C","D",...: 1 1 2 2 3 3 4 4 5 5 ...
## $ applicant.gender: Factor w/ 2 levels "female","male": 2 1 2 1 2 1 2 1 2 1 ...
## $ admit         : int  512 89 353 17 120 202 138 131 53 94 ...
## $ reject        : int  313 19 207 8 205 391 279 244 138 299 ...
## $ applications   : int  825 108 560 25 325 593 417 375 191 393 ...

d <- UCBadmit
```

model

$$\begin{aligned}n_{admit,i} &\sim \text{Binomial}(n_i, p_i) \\ \text{logit}(p_i) &= \alpha + \beta_m m_i \\ \alpha &\sim \text{Normal}(0, 10) \\ \beta_m &\sim \text{Normal}(0, 10)\end{aligned}$$

```
## R code 10.23
d$male <- ifelse( d$applicant.gender=="male" , 1 , 0 )
m10.6 <- map(
  alist(
```

```

      admit ~ dbinom( applications , p ) ,
      logit(p) <- a + bm*male ,
      a ~ dnorm(0,10) ,
      bm ~ dnorm(0,10)
    ) ,
    data=d )
m10.7 <- map(
  alist(
    admit ~ dbinom( applications , p ) ,
    logit(p) <- a ,
    a ~ dnorm(0,10)
  ) ,
  data=d )

## R code 10.24
compare( m10.6 , m10.7 )

```

```

##           WAIC pWAIC dWAIC weight    SE  dSE
## m10.6 5955.1   2.1   0.0     1 34.94   NA
## m10.7 6046.2   0.9  91.1     0 29.94 19.04

## R code 10.25
precis(m10.6)

```

```

##      Mean StdDev  5.5% 94.5%
## a  -0.83   0.05 -0.91 -0.75
## bm  0.61   0.06  0.51  0.71

```

male advantage

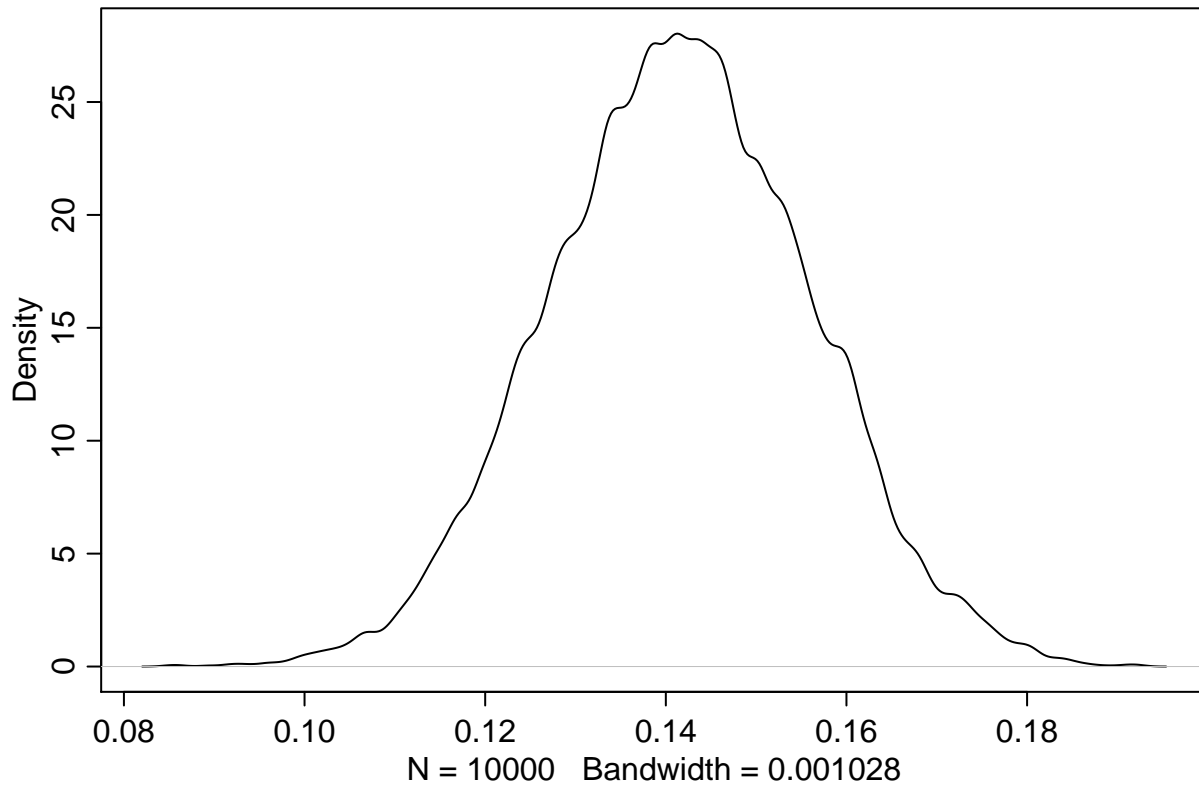
```

## R code 10.26
post <- extract.samples( m10.6 )
p.admit.male <- logistic( post$a + post$bm )
p.admit.female <- logistic( post$a )
diff.admit <- p.admit.male - p.admit.female
quantile( diff.admit , c(0.025, 0.5, 0.975) )

##      2.5%      50%      97.5%
## 0.1136123 0.1415307 0.1698560

dens(diff.admit)

```



postcheck

```
## R code 10.27
```

```
postcheck( m10.6 , n=1e4 )
```

```
## [ 1000 / 10000 ]
```

```
[ 2000 / 10000 ]
```

```
[ 3000 / 10000 ]
```

```
[ 4000 / 10000 ]
```

```
[ 5000 / 10000 ]
```

```
[ 6000 / 10000 ]
```

```
[ 7000 / 10000 ]
```

```
[ 8000 / 10000 ]
```

```
[ 9000 / 10000 ]
```

```
[ 10000 / 10000 ]
```

```
## [ 1000 / 10000 ]
```

```
[ 2000 / 10000 ]
```

```
[ 3000 / 10000 ]
```

```
[ 4000 / 10000 ]
```

```
[ 5000 / 10000 ]
```

```
[ 6000 / 10000 ]
```

```
[ 7000 / 10000 ]
```

```
[ 8000 / 10000 ]
```

```
[ 9000 / 10000 ]
```

```
[ 10000 / 10000 ]
```

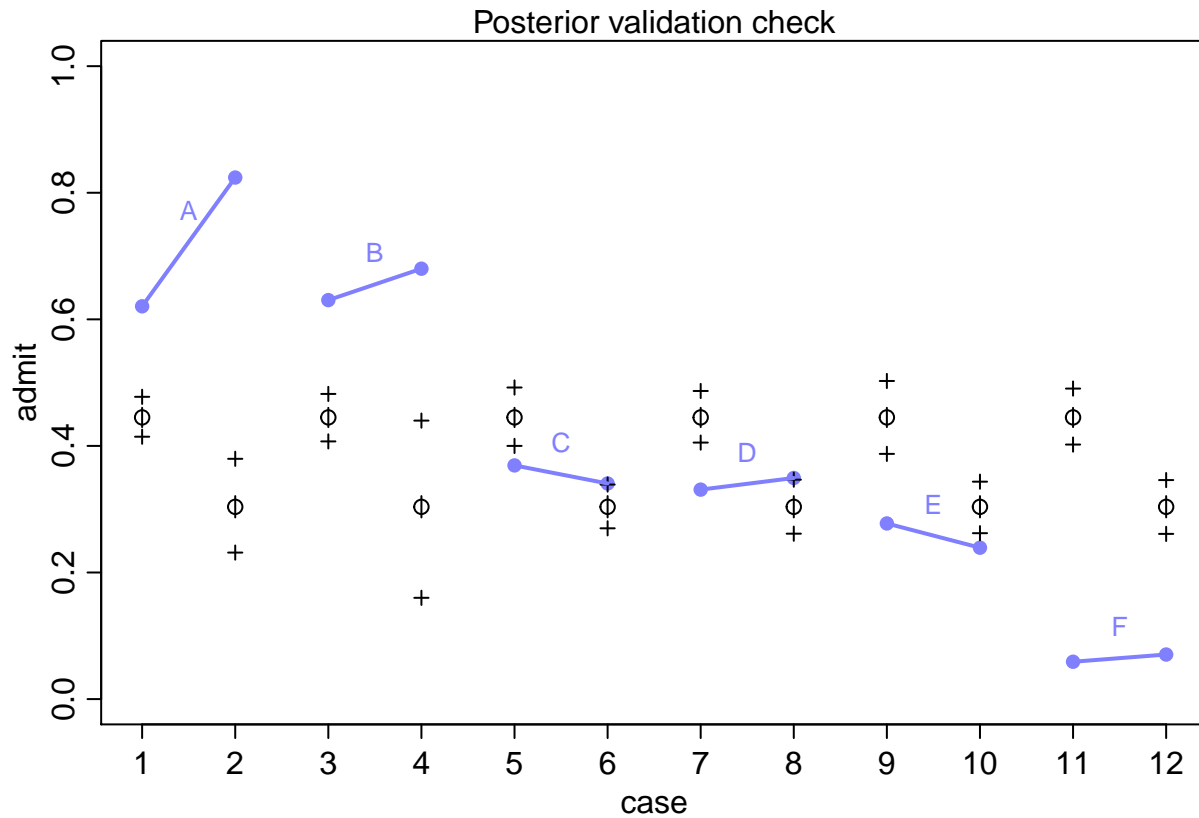
```
# draw lines connecting points from same dept
```

```
for ( i in 1:6 ) {
```

```

x <- 1 + 2*(i-1)
y1 <- d$admit[x]/d$applications[x]
y2 <- d$admit[x+1]/d$applications[x+1]
lines( c(x,x+1) , c(y1,y2) , col=rangi2 , lwd=2 )
text( x+0.5 , (y1+y2)/2 + 0.05 , d$dept[x] , cex=0.8 , col=rangi2 )
}

```



model with unique intercept for each dept

```

## R code 10.28
# make index
d$dept_id <- coerce_index( d$dept )

# model with unique intercept for each dept
m10.8 <- map(
  alist(
    admit ~ dbinom( applications , p ) ,
    logit(p) <- a[dept_id] ,
    a[dept_id] ~ dnorm(0,10)
  ) , data=d )

# model with male difference as well
m10.9 <- map(
  alist(
    admit ~ dbinom( applications , p ) ,
    logit(p) <- a[dept_id] + bm*male ,

```

```

    a[dept_id] ~ dnorm(0,10) ,
    bm ~ dnorm(0,10)
  ) , data=d )

```

```

## R code 10.29
compare( m10.6 , m10.7 , m10.8 , m10.9 )

```

```

##           WAIC pWAIC dWAIC weight    SE  dSE
## m10.9 5201.1   6.8   0.0   0.52 57.07   NA
## m10.8 5201.3   6.1   0.1   0.48 57.12  2.36
## m10.6 5954.8   1.9 753.6   0.00 35.02 49.24
## m10.7 6046.4   1.0 845.2   0.00 30.03 52.36

```

```

## R code 10.30
precis( m10.9 , depth=2 )

```

```

##           Mean StdDev  5.5% 94.5%
## a[1]  0.68   0.10  0.52  0.84
## a[2]  0.64   0.12  0.45  0.82
## a[3] -0.58   0.07 -0.70 -0.46
## a[4] -0.61   0.09 -0.75 -0.48
## a[5] -1.06   0.10 -1.22 -0.90
## a[6] -2.62   0.16 -2.88 -2.37
## bm   -0.10   0.08 -0.23  0.03

```

stan

```

## R code 10.31
m10.9stan <- map2stan( m10.9 , chains=2 , iter=2500 , warmup=500 )

```

```

## Warning: Variable 'applicant.gender' contains dots '.'.
## Will attempt to remove dots internally.

## Warning in FUN(X[[i]], ...): data with name dept is not numeric and not
## used

## Warning in FUN(X[[i]], ...): data with name applicant_gender is not numeric
## and not used

##
## SAMPLING FOR MODEL 'admit ~ dbinom(applications, p)' NOW (CHAIN 1).
##
## Chain 1, Iteration:    1 / 2500 [ 0%] (Warmup)
## Chain 1, Iteration:   250 / 2500 [ 10%] (Warmup)
## Chain 1, Iteration:   500 / 2500 [ 20%] (Warmup)
## Chain 1, Iteration:   501 / 2500 [ 20%] (Sampling)
## Chain 1, Iteration:   750 / 2500 [ 30%] (Sampling)
## Chain 1, Iteration:  1000 / 2500 [ 40%] (Sampling)
## Chain 1, Iteration:  1250 / 2500 [ 50%] (Sampling)
## Chain 1, Iteration:  1500 / 2500 [ 60%] (Sampling)
## Chain 1, Iteration:  1750 / 2500 [ 70%] (Sampling)
## Chain 1, Iteration:  2000 / 2500 [ 80%] (Sampling)
## Chain 1, Iteration:  2250 / 2500 [ 90%] (Sampling)
## Chain 1, Iteration:  2500 / 2500 [100%] (Sampling)
## Elapsed Time: 0.016648 seconds (Warm-up)

```

```

##           0.065139 seconds (Sampling)
##           0.081787 seconds (Total)
##
##
## SAMPLING FOR MODEL 'admit ~ dbinom(applications, p)' NOW (CHAIN 2).
##
## Chain 2, Iteration:   1 / 2500 [ 0%] (Warmup)
## Chain 2, Iteration: 250 / 2500 [10%] (Warmup)
## Chain 2, Iteration: 500 / 2500 [20%] (Warmup)
## Chain 2, Iteration: 501 / 2500 [20%] (Sampling)
## Chain 2, Iteration: 750 / 2500 [30%] (Sampling)
## Chain 2, Iteration:1000 / 2500 [40%] (Sampling)
## Chain 2, Iteration:1250 / 2500 [50%] (Sampling)
## Chain 2, Iteration:1500 / 2500 [60%] (Sampling)
## Chain 2, Iteration:1750 / 2500 [70%] (Sampling)
## Chain 2, Iteration:2000 / 2500 [80%] (Sampling)
## Chain 2, Iteration:2250 / 2500 [90%] (Sampling)
## Chain 2, Iteration:2500 / 2500 [100%] (Sampling)
## Elapsed Time: 0.019036 seconds (Warm-up)
##           0.06792 seconds (Sampling)
##           0.086956 seconds (Total)

## Warning in FUN(X[[i]], ...): data with name dept is not numeric and not
## used

## Warning in FUN(X[[i]], ...): data with name applicant_gender is not numeric
## and not used

##
## SAMPLING FOR MODEL 'admit ~ dbinom(applications, p)' NOW (CHAIN 1).
## WARNING: No variance estimation is
##           performed for num_warmup < 20
##
##
## Chain 1, Iteration: 1 / 1 [100%] (Sampling)
## Elapsed Time: 2e-06 seconds (Warm-up)
##           2.9e-05 seconds (Sampling)
##           3.1e-05 seconds (Total)

## Computing WAIC

## Constructing posterior predictions

## [ 400 / 4000 ]
## [ 800 / 4000 ]
## [1200 / 4000 ]
## [1600 / 4000 ]
## [2000 / 4000 ]
## [2400 / 4000 ]
## [2800 / 4000 ]
## [3200 / 4000 ]
## [3600 / 4000 ]
## [4000 / 4000 ]

## Aggregated binomial counts detected. Splitting to 0/1 outcome for WAIC calculation.

```

```
precis(m10.9stan,depth=2)
```

```
##      Mean StdDev lower 0.89 upper 0.89 n_eff Rhat
## a[1]  0.68   0.10    0.54    0.84 1837    1
## a[2]  0.64   0.11    0.46    0.82 1699    1
## a[3] -0.58   0.07   -0.70   -0.46 4000    1
## a[4] -0.62   0.08   -0.75   -0.48 2820    1
## a[5] -1.06   0.09   -1.21   -0.91 3523    1
## a[6] -2.64   0.16   -2.87   -2.37 3243    1
## bm   -0.10   0.08   -0.23    0.03 1446    1
```

```
stancode(m10.9stan)
```

```
## data{
##   int<lower=1> N;
##   int<lower=1> N_dept_id;
##   int admit[N];
##   int applications[N];
##   real male[N];
##   int dept_id[N];
## }
## parameters{
##   vector[N_dept_id] a;
##   real bm;
## }
## model{
##   vector[N] p;
##   bm ~ normal( 0 , 10 );
##   a ~ normal( 0 , 10 );
##   for ( i in 1:N ) {
##     p[i] = a[dept_id[i]] + bm * male[i];
##   }
##   admit ~ binomial_logit( applications , p );
## }
## generated quantities{
##   vector[N] p;
##   real dev;
##   dev = 0;
##   for ( i in 1:N ) {
##     p[i] = a[dept_id[i]] + bm * male[i];
##   }
##   dev = dev + (-2)*binomial_logit_lpmf( admit | applications , p );
## }
```