

# islands

*Chris Parrish*

*June 30, 2016*

## Contents

<b>islands</b>	<b>1</b>
data . . . . .	1
model . . . . .	2
fit overdispersed Poisson model . . . . .	2
simulate counterfactual societies . . . . .	4
plot . . . . .	5

islands

references:

- McElreath, Statistical Rethinking, chap 12, p.382

```
library(rethinking)
library(ggplot2)
```

## islands

### data

```
## R code 12.39
# prep data
data(Kline)
d <- Kline
d$logpop <- log(d$population)
d$society <- 1:10
str(d)

## 'data.frame':  10 obs. of  7 variables:
## $ culture      : Factor w/ 10 levels "Chuuk","Hawaii",...: 4 7 6 10 3 9 1 5 8 2
## $ population  : int  1100 1500 3600 4791 7400 8000 9200 13000 17500 275000
## $ contact      : Factor w/ 2 levels "high","low": 2 2 2 1 1 1 1 2 1 2
## $ total_tools : int  13 22 24 43 33 19 40 28 55 71
## $ mean_TU     : num  3.2 4.7 4 5 5 4 3.8 6.6 5.4 6.6
## $ logpop      : num  7 7.31 8.19 8.47 8.91 ...
## $ society     : int  1 2 3 4 5 6 7 8 9 10
```

## model

$$\begin{aligned}T_i &\sim \text{Poisson}(\mu_i) \\ \log(\mu_i) &= \alpha + \alpha_{\text{SOCIETY}[i]} + \beta_P \log P_i \\ \alpha &\sim \text{Normal}(0, 10) \\ \alpha_{\text{SOCIETY}} &\sim \text{Normal}(0, \sigma_{\text{SOCIETY}}) \\ \beta_P &\sim \text{Normal}(0, 10) \\ \sigma_{\text{SOCIETY}} &\sim \text{HalfCauchy}(0, 1)\end{aligned}$$

## fit overdispersed Poisson model

```
# fit model
m12.6 <- map2stan(
  alist(
    total_tools ~ dpois(mu),
    log(mu) <- a + a_society[society] + bp*logpop,
    a ~ dnorm(0,10),
    bp ~ dnorm(0,1),
    a_society[society] ~ dnorm(0,sigma_society),
    sigma_society ~ dcauchy(0,1)
  ),
  data=d ,
  iter=4000 , chains=3 )
```

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

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

```
##
```

```
## SAMPLING FOR MODEL 'total_tools ~ dpois(mu)' NOW (CHAIN 1).
```

```
##
```

```
## Chain 1, Iteration:    1 / 4000 [ 0%] (Warmup)
## Chain 1, Iteration:   400 / 4000 [ 10%] (Warmup)
## Chain 1, Iteration:   800 / 4000 [ 20%] (Warmup)
## Chain 1, Iteration:  1200 / 4000 [ 30%] (Warmup)
## Chain 1, Iteration:  1600 / 4000 [ 40%] (Warmup)
## Chain 1, Iteration:  2000 / 4000 [ 50%] (Warmup)
## Chain 1, Iteration:  2001 / 4000 [ 50%] (Sampling)
## Chain 1, Iteration:  2400 / 4000 [ 60%] (Sampling)
## Chain 1, Iteration:  2800 / 4000 [ 70%] (Sampling)
## Chain 1, Iteration:  3200 / 4000 [ 80%] (Sampling)
## Chain 1, Iteration:  3600 / 4000 [ 90%] (Sampling)
## Chain 1, Iteration:  4000 / 4000 [100%] (Sampling)
```

```
## Elapsed Time: 0.33335 seconds (Warm-up)
```

```
##                0.396673 seconds (Sampling)
```

```
##                0.730023 seconds (Total)
```

```
##
```

```
##
```

```
## SAMPLING FOR MODEL 'total_tools ~ dpois(mu)' NOW (CHAIN 2).
```

```
##
```

```

## Chain 2, Iteration: 1 / 4000 [ 0%] (Warmup)
## Chain 2, Iteration: 400 / 4000 [ 10%] (Warmup)
## Chain 2, Iteration: 800 / 4000 [ 20%] (Warmup)
## Chain 2, Iteration: 1200 / 4000 [ 30%] (Warmup)
## Chain 2, Iteration: 1600 / 4000 [ 40%] (Warmup)
## Chain 2, Iteration: 2000 / 4000 [ 50%] (Warmup)
## Chain 2, Iteration: 2001 / 4000 [ 50%] (Sampling)
## Chain 2, Iteration: 2400 / 4000 [ 60%] (Sampling)
## Chain 2, Iteration: 2800 / 4000 [ 70%] (Sampling)
## Chain 2, Iteration: 3200 / 4000 [ 80%] (Sampling)
## Chain 2, Iteration: 3600 / 4000 [ 90%] (Sampling)
## Chain 2, Iteration: 4000 / 4000 [100%] (Sampling)
## Elapsed Time: 0.369317 seconds (Warm-up)
## 0.419191 seconds (Sampling)
## 0.788508 seconds (Total)

## The following numerical problems occurred the indicated number of times after warmup on chain 2
##
## count
## Exception thrown at line 17: normal_log: Scale parameter is 0, but must be > 0! 1
## When a numerical problem occurs, the Metropolis proposal gets rejected.
## However, by design Metropolis proposals sometimes get rejected even when there are no numerical prob.
## Thus, if the number in the 'count' column is small, do not ask about this message on stan-users.
##
## SAMPLING FOR MODEL 'total_tools ~ dpois(mu)' NOW (CHAIN 3).
##
## Chain 3, Iteration: 1 / 4000 [ 0%] (Warmup)
## Chain 3, Iteration: 400 / 4000 [ 10%] (Warmup)
## Chain 3, Iteration: 800 / 4000 [ 20%] (Warmup)
## Chain 3, Iteration: 1200 / 4000 [ 30%] (Warmup)
## Chain 3, Iteration: 1600 / 4000 [ 40%] (Warmup)
## Chain 3, Iteration: 2000 / 4000 [ 50%] (Warmup)
## Chain 3, Iteration: 2001 / 4000 [ 50%] (Sampling)
## Chain 3, Iteration: 2400 / 4000 [ 60%] (Sampling)
## Chain 3, Iteration: 2800 / 4000 [ 70%] (Sampling)
## Chain 3, Iteration: 3200 / 4000 [ 80%] (Sampling)
## Chain 3, Iteration: 3600 / 4000 [ 90%] (Sampling)
## Chain 3, Iteration: 4000 / 4000 [100%] (Sampling)
## Elapsed Time: 0.374522 seconds (Warm-up)
## 0.398825 seconds (Sampling)
## 0.773347 seconds (Total)

## The following numerical problems occurred the indicated number of times after warmup on chain 3
##
## count
## Exception thrown at line 17: normal_log: Scale parameter is 0, but must be > 0! 1
## When a numerical problem occurs, the Metropolis proposal gets rejected.
## However, by design Metropolis proposals sometimes get rejected even when there are no numerical prob.
## Thus, if the number in the 'count' column is small, do not ask about this message on stan-users.
## Warning in FUN(X[[i]], ...): data with name culture is not numeric and not
## used

```

```

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

##
## SAMPLING FOR MODEL 'total_tools ~ dpois(mu)' NOW (CHAIN 1).
## WARNING: No variance estimation is
##           performed for num_warmup < 20
##
##
## Chain 1, Iteration: 1 / 1 [100%] (Sampling)
## Elapsed Time: 3e-06 seconds (Warm-up)
##               2.7e-05 seconds (Sampling)
##               3e-05 seconds (Total)

## Computing WAIC

## Constructing posterior predictions

## [ 600 / 6000 ]
[ 1200 / 6000 ]
[ 1800 / 6000 ]
[ 2400 / 6000 ]
[ 3000 / 6000 ]
[ 3600 / 6000 ]
[ 4200 / 6000 ]
[ 4800 / 6000 ]
[ 5400 / 6000 ]
[ 6000 / 6000 ]

```

## simulate counterfactual societies

```

## R code 12.40
post <- extract.samples(m12.6)
d.pred <- list(
  logpop = seq(from=6,to=14,length.out=30),
  society = rep(1,30)
)
a_society_sims <- rnorm(20000,0,post$sigma_society)
a_society_sims <- matrix(a_society_sims,2000,10)
link.m12.6 <- link( m12.6 , n=2000 , data=d.pred ,
  replace=list(a_society=a_society_sims) )

## [ 200 / 2000 ]
[ 400 / 2000 ]
[ 600 / 2000 ]
[ 800 / 2000 ]
[ 1000 / 2000 ]
[ 1200 / 2000 ]
[ 1400 / 2000 ]
[ 1600 / 2000 ]
[ 1800 / 2000 ]
[ 2000 / 2000 ]

```

plot

```
## R code 12.41
# plot raw data
plot( d$logpop , d$total_tools , col=rangi2 , pch=16 ,
      xlab="log population" , ylab="total tools" )

# plot posterior median
mu.median <- apply( link.m12.6 , 2 , median )
lines( d.pred$logpop , mu.median )

# plot 97%, 89%, and 67% intervals (all prime numbers)
mu.PI <- apply( link.m12.6 , 2 , PI , prob=0.97 )
shade( mu.PI , d.pred$logpop )
mu.PI <- apply( link.m12.6 , 2 , PI , prob=0.89 )
shade( mu.PI , d.pred$logpop )
mu.PI <- apply( link.m12.6 , 2 , PI , prob=0.67 )
shade( mu.PI , d.pred$logpop )
```

