

mark-recapture

Ben Williams

December 18, 2017

Basic M-R

In this example 1000 fish were caught and marked ($n1$). Return and capture 2000 fish ($N2$), of these 20 were previously marked ($m2$).

Remove the bias in this estimate using the Chapmanized Petersen estimator

$$N = \frac{(n2+1)(n1+1)}{(m2+1)} - 1$$

```
n1 <- 1000
n2 <- 2000
m2 <- 280

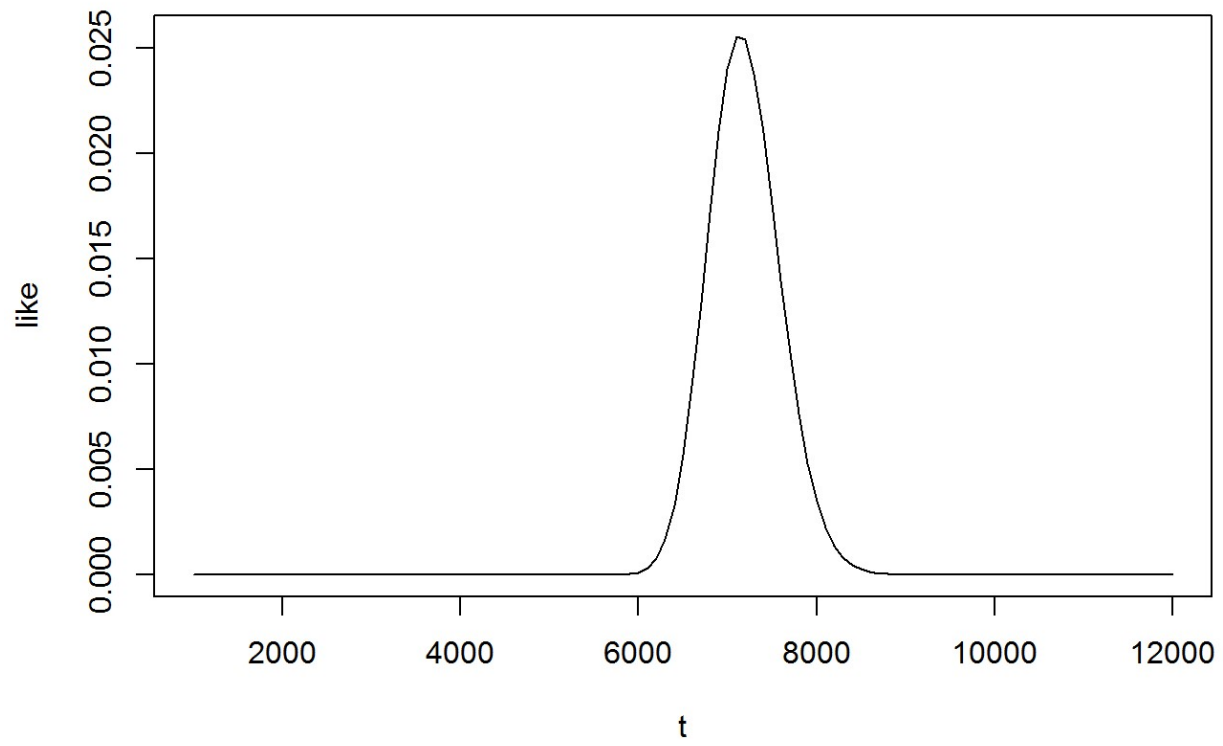
N <- ((n2 + 1)*(n1 + 1)/(m2 + 1))-1
varN <- ((n1 + 1) * (n2 + 1) * (n2 - m2)) / ((m2 + 1)^2 * (m2 + 2))
seN <- sqrt(varN)
```

$N = 7127.1174377$

$seN = 12.4386696$

Binomial likelihood estimation of the population abundance

```
t <- seq(n1, 12000, 100)
like <- exp(lfactorial(n2) - lfactorial(m2) - lfactorial(n2 - m2) + m2 * log(n1 / t)
+ (n2 - m2) * log(1 - n1 / t))
plot(t, like, type = 'l')
```



Bayesian implementation with a prior on the U interval (unmarked fish numbers)

```

library(tidyverse)
library(rjags)

cat("
  model {
    # likelihood function
    m2 ~ dbin(theta, n1) # marked fish
    u ~ dbin(theta, U) # unmarked fish

    # prior distribution
    theta ~ dunif(0.0, 0.6) # capture probabilities
    U_ ~ dunif(500, 180000) # number of unmarked fish
    U = round(U_)
  }
", file = "m1.jag")

dat <- list(m2 = 280, n1 = 1000, u = n2 - m2)
ini <- list(theta = 0.2, U_ = 10000)

m1 <- jags.model("m1.jag",
  data = dat,
  n.chains = 2,
  init = ini,
  n.adapt = 1000)

```

```

## Compiling model graph
##   Resolving undeclared variables
##   Allocating nodes
## Graph information:
##   Observed stochastic nodes: 2
##   Unobserved stochastic nodes: 2
##   Total graph size: 10
##
## Initializing model

```

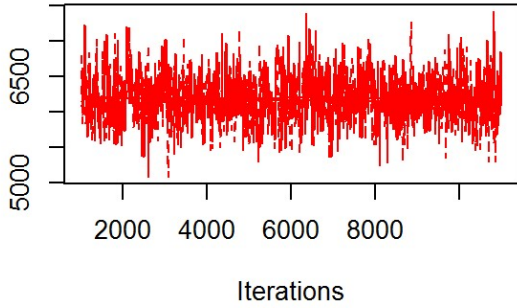
```

mpar <- c("U", "theta")
res <- coda.samples(m1,
  var = mpar,
  n.iter = 10000,
  thin = 10)

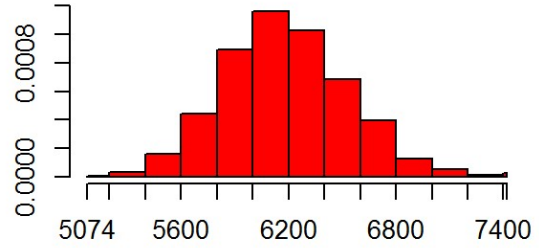
plot(res, col = 2)

```

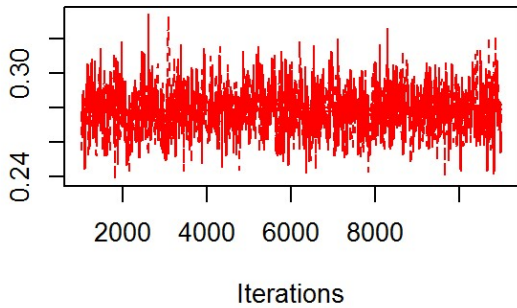
Trace of U



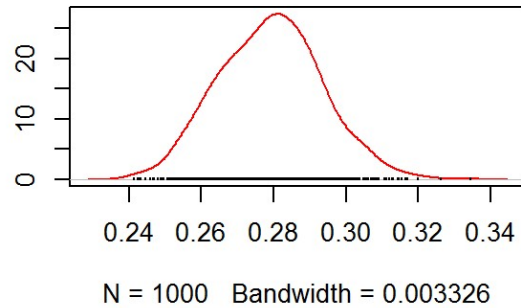
Density of U



Trace of theta



Density of theta



summary (res)

```
##
## Iterations = 1010:11000
## Thinning interval = 10
## Number of chains = 2
## Sample size per chain = 1000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##           Mean          SD Naive SE Time-series SE
## U          6181.6050 343.60448 7.6832298    11.8453169
## theta      0.2791    0.01435 0.0003208     0.0004899
##
## 2. Quantiles for each variable:
##
##           2.5%      25%      50%      75%      97.5%
## U          5539.0000 5948.7500 6164.0000 6405.0000 6896.0250
## theta      0.2525    0.2688    0.2793    0.2886    0.3074
```

```

coda_df <- function(coda.object, parameters = NULL) {

  if (!coda::is.mcmc(coda.object) && !coda::is.mcmc.list(coda.object))
    stop("Not an mcmc or mcmc.list object")

  n.chain <- coda::nchain(coda.object)
  mat <- as.matrix(coda.object, iter = TRUE, chain = TRUE)
  df <- as.data.frame(mat)

  if(n.chain == 1)
    df <- data.frame(1, df)

  names(df) <- c("chain", "iter", coda::varnames(coda.object))

  if(is.null(parameters))
    out.df <- df

  if(!is.null(parameters))
    out.df <- subset(df, select = c("chain", "iter", parameters))

  out.df
}

coda_df(res) %>%
  mutate(q025 = quantile(U, 0.025),
         q975 = quantile(U, 0.975),
         ci = ifelse(U >= q025 & U<=q975, 1, 0)) %>%
  ggplot(aes(U)) + geom_histogram(fill = 4, alpha = 0.2, bins = 100, color = 'black')
+
  geom_histogram(data = . %>% filter(ci==1), aes(U), fill = 4, alpha = 0.6, bins = 100)

```

