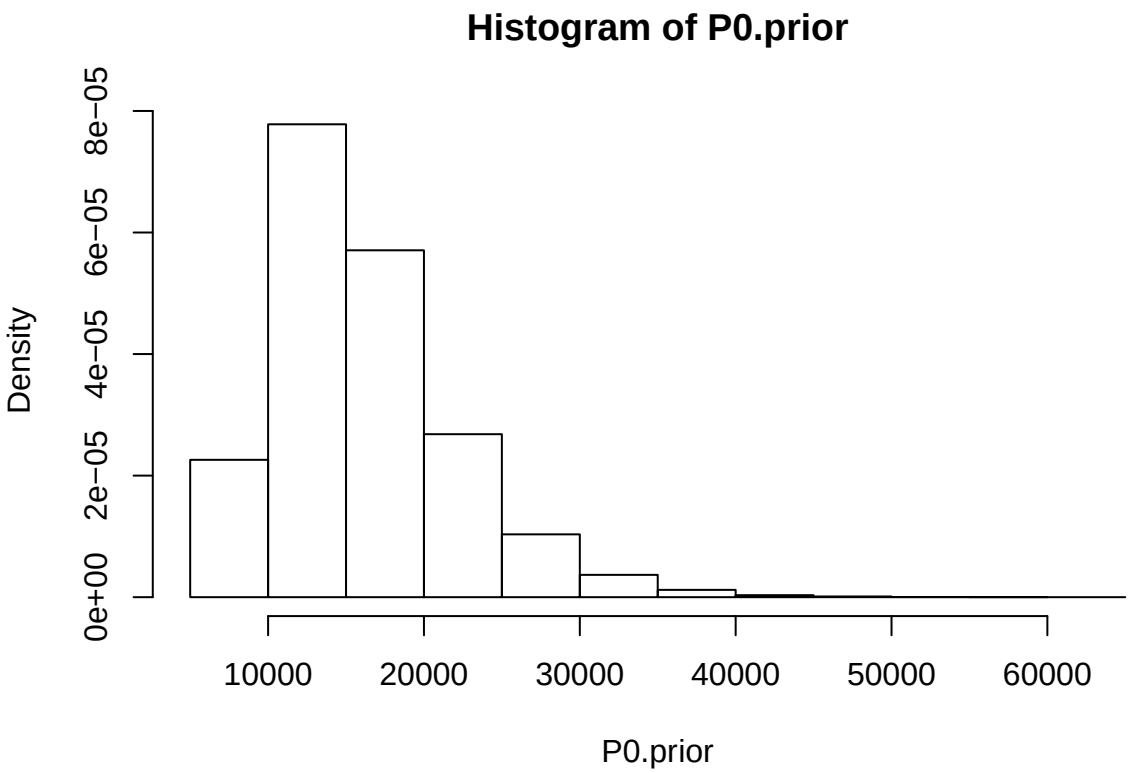
bowhead_example_SpIR_alpha.r

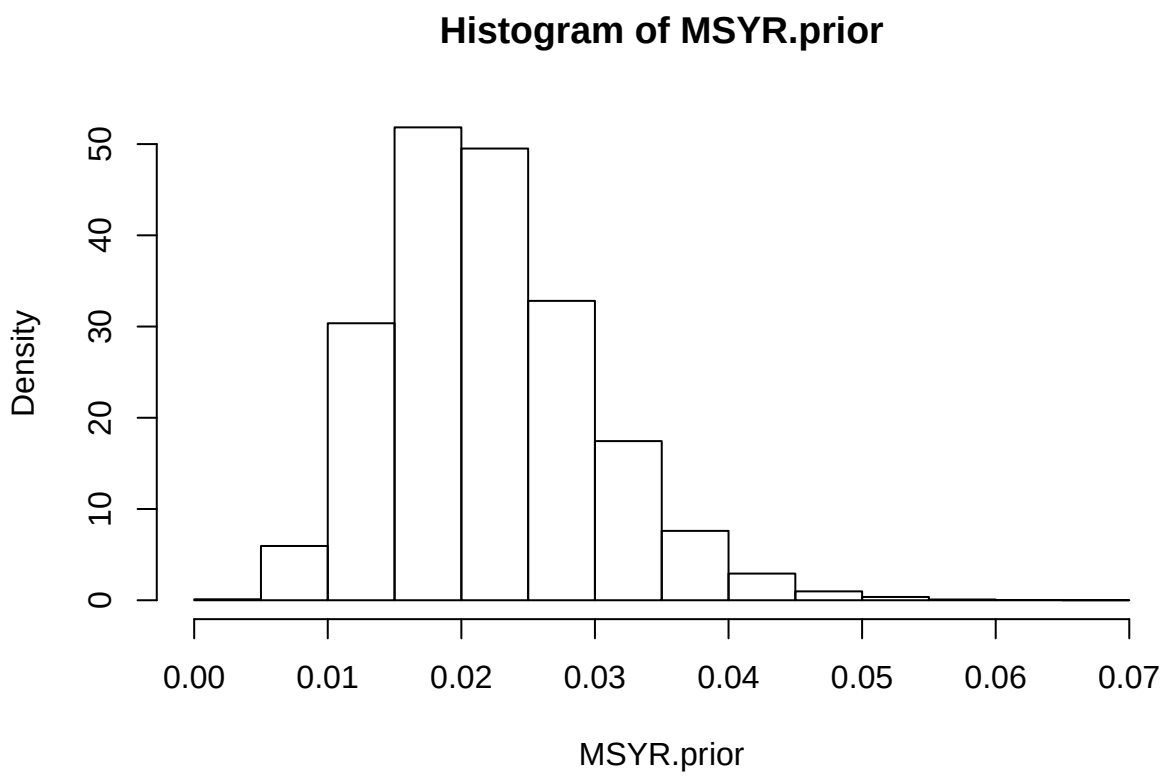
max

Fri Apr 12 10:47:01 2019

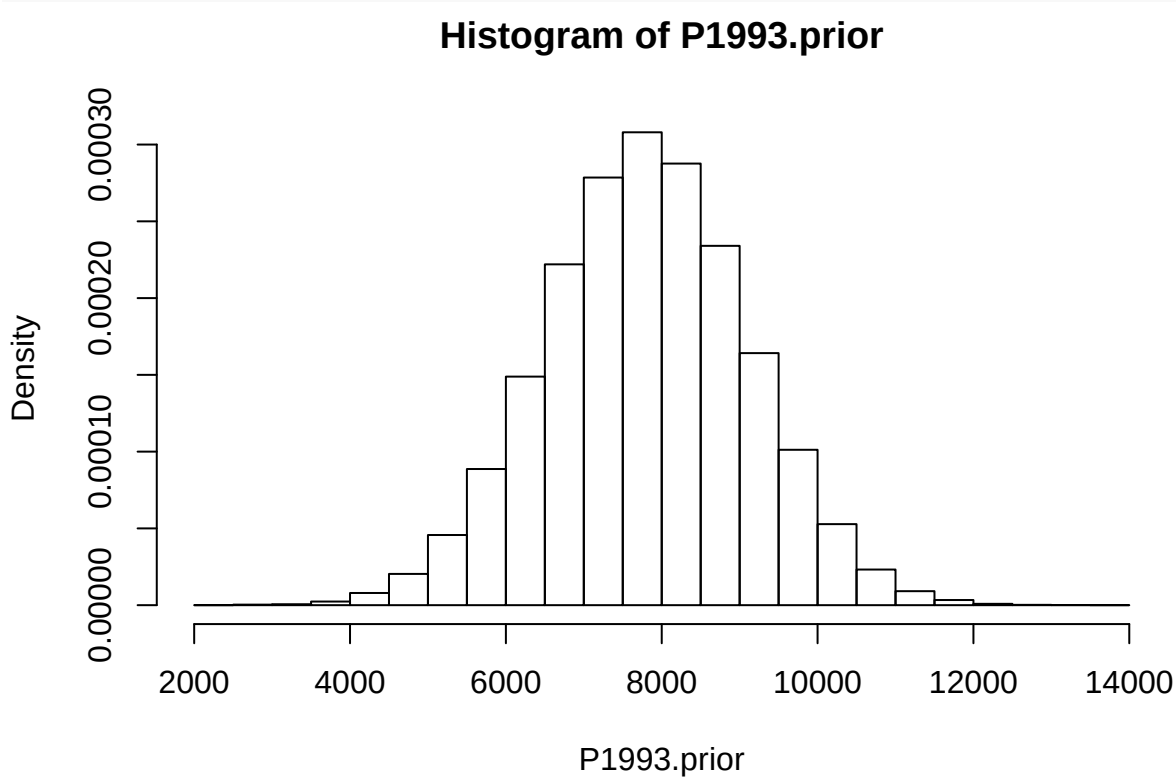
```r
# P_{t+1} = P_t - C_t + 1.5 * (MSYR) * P_t(1-(P_t/P_0)^2)
# Inputs: P_0 and MSYR
# Output: P_{1993}
source("pooling_aux.r")
## Prior for P_0 is a Gamma(2.8085, 0.0002886) + 6400, i.e, with offset 6400.
M <- 1E5
P0.prior <- rgamma(M, shape = 2.8085, rate = 0.0002886) + 6400
hist(P0.prior, probability = TRUE)
```



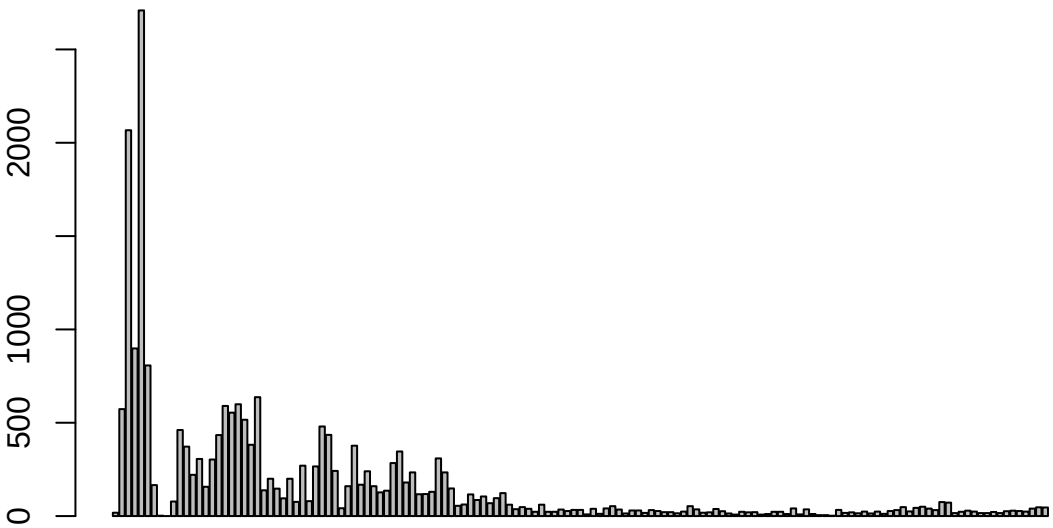

```r
## Prior for MSYR is Gamma(8.2, 372.7)
MSYR.prior <- rgamma(M, shape = 8.2, rate = 372.7)
hist(MSYR.prior, probability = TRUE)
```



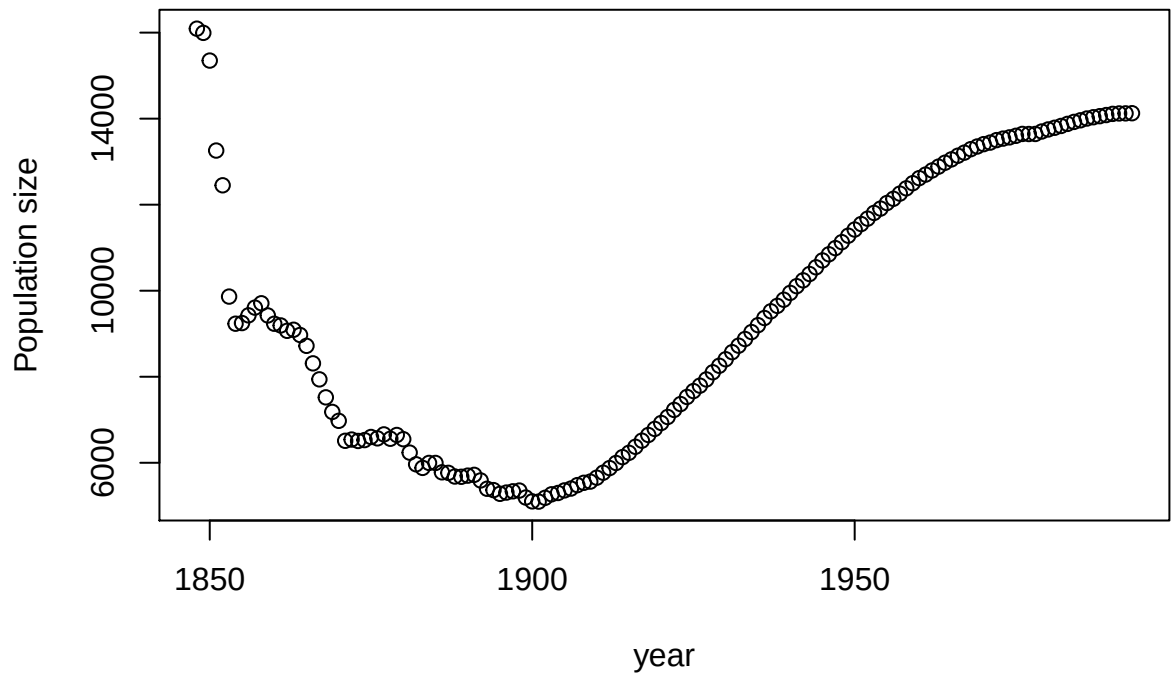

```
## Prior on P_{1993} is N(7800, 1300^2)
P1993.prior <- rnorm(M, mean = 7800, sd = 1300) ## q_2(\phi)
hist(P1993.prior, probability = TRUE)
```



```r
P1993.prior.dens <- density(P1993.prior)

P1993.likelihood2 <- function(x) dnorm(x, mean = 8293,  sd = 626) ## L_2(\phi)

CtData <- read.csv("../data/bowhead_kills_1848-2002.csv")
CtData <- subset(CtData, year <= 1992)
barplot(CtData$kills)
```




```r
getP <- function(msyr, p0, C, Pinit){
  TT <- length(C)
  P <- rep(NA, TT + 1)
  P[1] <- Pinit
  for(t in 1: TT)  P[t+1] <- P[t] - C[t] + 1.5 * msyr * P[t]*(1-(P[t]/p0)^2)
  return(P)
}
plot(
  seq(min(CtData$year), max(CtData$year) + 1, by = 1),
  getP(msyr = median(MSYR.prior),
       p0 = median(P0.prior),
       C = CtData$kills,
       Pinit = mean(P0.prior)),
  xlab = "year", ylab = "Population size"
)
```



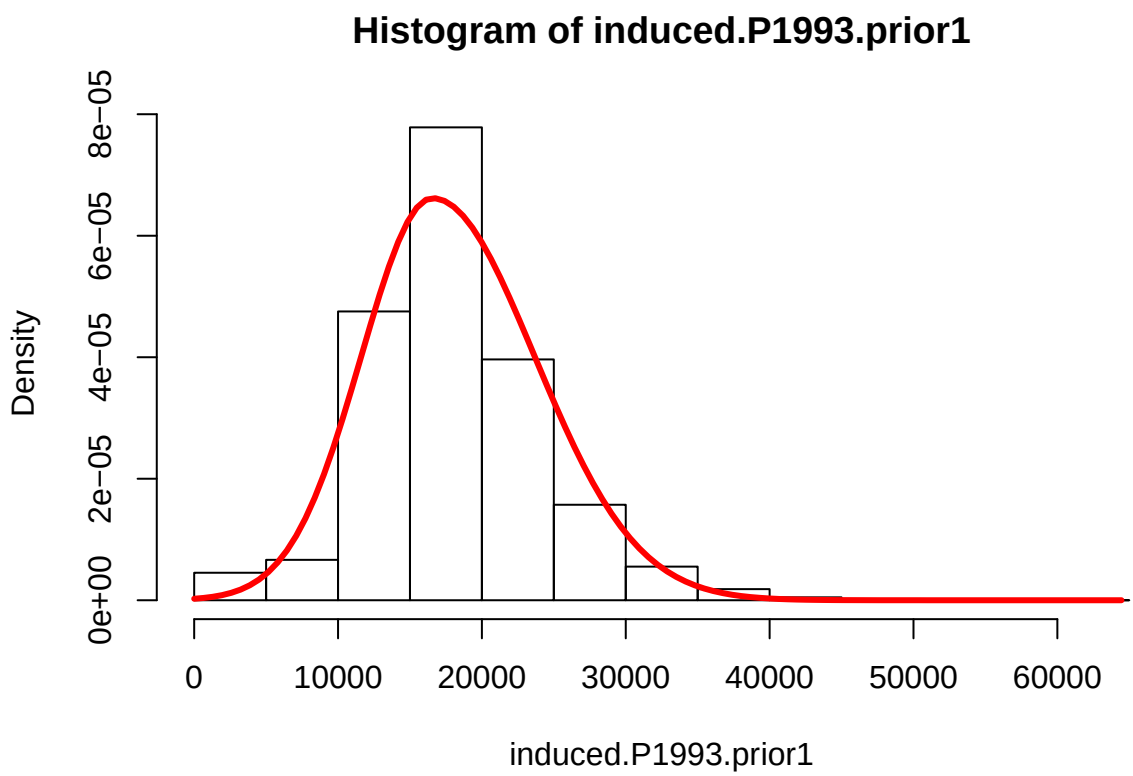

```
induced.P1993.prior1 <- unlist(
  parallel::mclapply(1:M, function(i){
    simuP <- getP(msyr = MSYR.prior[i],
                  p0 = P0.prior[i],
                  C = CtData$kills,
                  Pinit = P0.prior[i])
    return(simuP[length(simuP)])
  }, mc.cores = 4)
)
induced.P1993.prior1 <- induced.P1993.prior1[induced.P1993.prior1>0]
induced.P1993.prior1.dens <- density(induced.P1993.prior1)

summary(induced.P1993.prior1)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     2.3 14489.4 17408.9 18137.7 21304.0 64456.2
```

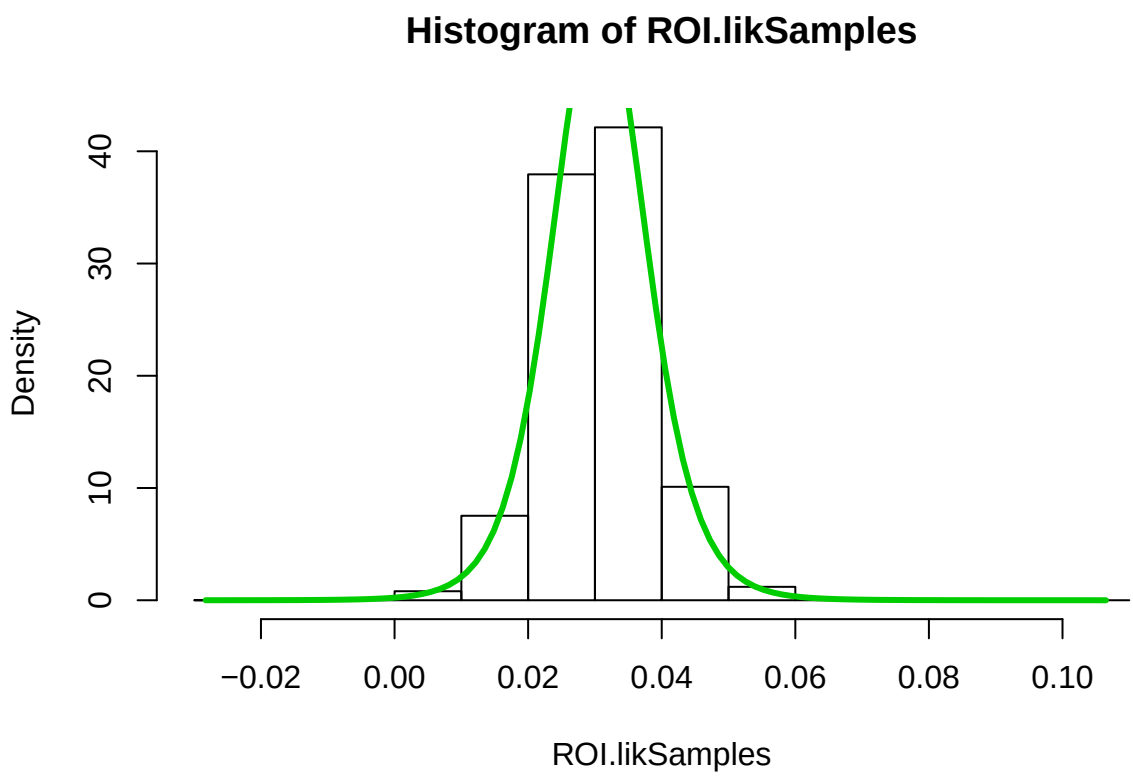
```
c(mean(induced.P1993.prior1), quantile(induced.P1993.prior1, probs = c(.025, .5, .975)))
```

```
##                2.5%       50%     97.5%
## 18137.732  5488.148 17408.937 32100.091
```

```
hist(induced.P1993.prior1, probability = TRUE)
distFit <- fGarch::snormFit(induced.P1993.prior1)
curve(fGarch::dsnorm(x, mean = distFit$par["mean"],
                     sd = distFit$par["sd"], xi =   distFit$par["xi"]),
      min(induced.P1993.prior1), max(induced.P1993.prior1), col = 2, lwd = 3, add = TRUE)
```




```
###########
# Likelihood for ROI
# P1993  = P1978* (1 + ROI)^15
# Likelihood(ROI) = y = g(t8) = exp(a + bt8) -1 where
# t8 is a student t-distribution with df = 8
# g_inverse(y) = t8 = (log(y + 1) - a)/b
# dg_inverse(y)/dy = 1/b(y + 1), hence |J| = abs(1/b(y + 1))
# here a = .0302 and b = .0069
ROI.likelihood <- function(y, a = .0302, b = .0069, log = FALSE){
  if(is.nan(y) && log == FALSE) return(0)
  if(is.nan(y) && log == TRUE) return(-Inf)
  ## L_2(\phi)
  if(log){
    dt((log(y + 1) - a)/b, df = 8, log = TRUE) + log(abs(1/(b * (y + 1))))
  }else{
    dt((log(y + 1) - a)/b, df = 8) *  abs(1/(b * (y + 1)))
  }
}
ROI.likSamples <- exp(.0302  + .0069 * rt(M, df = 8)) - 1

hist(ROI.likSamples, probability = TRUE)
curve(ROI.likelihood, min(ROI.likSamples), max(ROI.likSamples),
      lwd = 3, col = 3, add = TRUE)
```

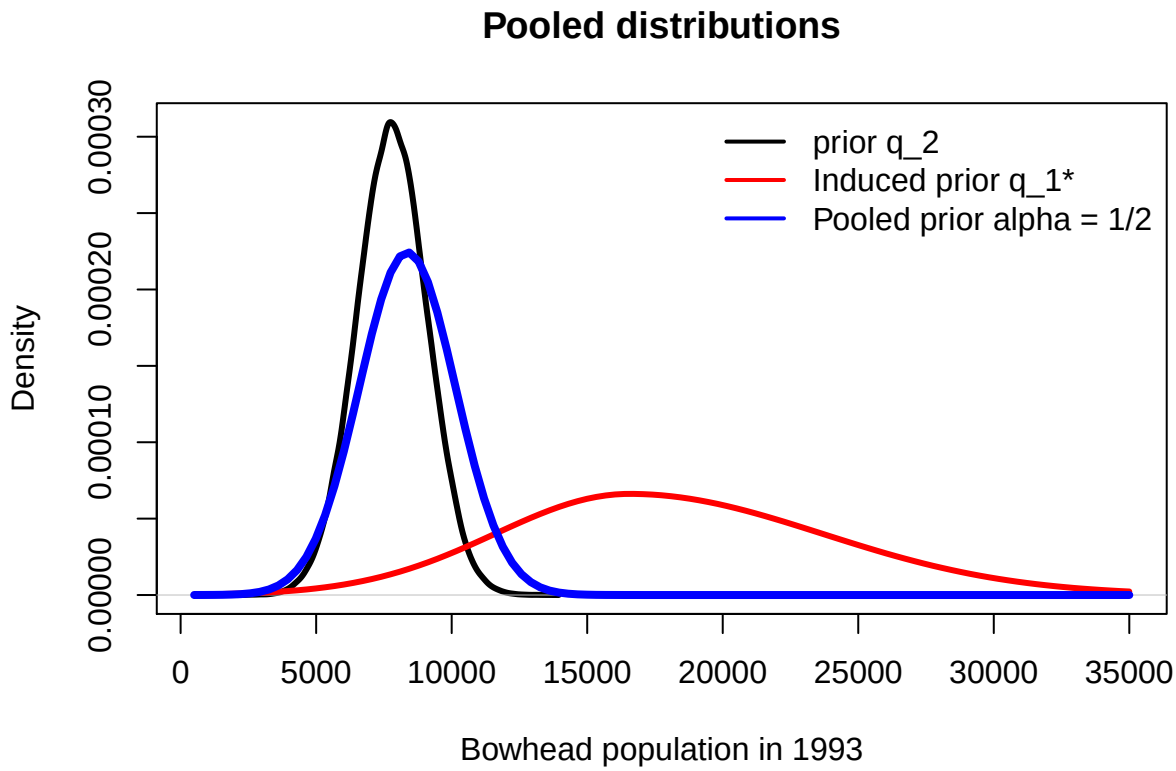



```
getROI <- function(pars){
  Ps <-  getP(msyr = pars[2], p0 = pars[1],
              C = CtData$kills, Pinit = pars[1])
  P1978 <- Ps[length(Ps) - 15]
  P1993 <- Ps[length(Ps)]
  ROI <- (P1993/P1978)^{1/15} - 1
  return(ROI)
}
#########
# The pooled prior LP({q_2(\phi), q_1^\ast(\phi)}, alpha),
P1993.pooledPrior <- function(x, weights){
  if(is.nan(x)) return(0)
  dpoolnorm(x = x,
            D = list(
              f0 = function(x) fGarch::dsnorm(x, mean = distFit$par["mean"],
                                              sd = distFit$par["sd"],
                                              xi =   distFit$par["xi"]) ,
              f1 = function(x) dnorm(x, mean = 7800, sd = 1300) ),
            alphas = weights)
}
##########
Alpha <- .5
alphas <- c(Alpha, 1-Alpha)
###
plot(P1993.prior.dens, lwd = 3, xlim = c(500, 35000),
     main = "Pooled distributions", xlab = "Bowhead population in 1993")
curve(fGarch::dsnorm(x, mean = distFit$par["mean"],
                     sd = distFit$par["sd"], xi = distFit$par["xi"]),
      500, 35000, col = 2, lwd = 3, add = TRUE)
```

```r
curve(P1993.pooledPrior(x, weights = c(Alpha, 1-Alpha) ),
      500, 35000,
      col = "blue", lwd = 4, add = TRUE)
```

```
## Warning in if (is.nan(x)) return(0): the condition has length > 1 and only
## the first element will be used
```

```r
legend(x = "topright",
       legend = c("prior q_2", "Induced prior q_1*", "Pooled prior alpha = 1/2"),
       col = c("black", "red", "blue"), lwd = 2, bty = "n")
```




```r
#########
# Now let's get the posterior using LP-SpIR [for fixed alpha]
getP1993 <- function(theta){
  P <- getP(msyr = theta[2], p0 = theta[1],
            C = CtData$kills, Pinit = theta[1])
  return(P[length(P)])
}
# Logarithmic pooling (LP) via Sampling-importance-resampling (SpIR)
LP.SpIR_mod <- function(k, l, Model, rq1, dq2, dL1, dL2){
  ## currently tuned to perform LP on two dimensions
  theta.samp <- rq1(n = k)
  phi.transf <- unlist(
    parallel::mclapply(1:nrow(theta.samp),
                       function(i) Model(theta.samp[i, 1:2]), mc.cores = 4)
  )
  q1star <- makedf(phi.transf)
  corrected_q1star <-  function(x){
    res <- q1star(x)
    res[is.na(res)] <- 0
```

```r
    return(res)
  }
  getKa <- function(a){
    tpool(alpha = c(a, 1-a),
          D = list(
            f0 = function(x) corrected_q1star(x),
            f1 = function(x) dnorm(x, mean = 7800, sd = 1300) )
    )
  }
  getWeight <- function(theta, phi){
    getKa(theta[3]) * (dq2(phi)/q1star(phi))^{1-theta[3]} * dL1(theta) * dL2(phi)
  }
  # ws <- sapply(seq_len(nrow(theta.samp)), function(i) {
  #   getWeight(theta = theta.samp[i, ], phi = phi.transf[i])
  # })
  ws <- unlist(
    parallel::mclapply(seq_len(nrow(theta.samp)), function(i) {
      getWeight(theta = theta.samp[i, ], phi = phi.transf[i])
    }, mc.cores = 4)
  )
  ws[which(ws == -Inf)] <- 0 ## giving  weight = 0 to "impossible" values
  resamp.Pos <-  sample(seq_len(nrow(theta.samp)), size = l,
                        replace = TRUE, prob = ws/sum(ws))
  return(list(
    theta.resamp = theta.samp[resamp.Pos, ],
    phi.resamp = phi.transf[resamp.Pos])
  )
}
##
SpIR.posterior <- LP.SpIR_mod(
  k = M,
  l = M,
  Model = getP1993,
  rq1 = function(n) cbind(rgamma(n, shape = 2.8085, rate = 0.0002886) + 6400,
                          rgamma(n, shape = 8.2, rate = 372.7),
                          rbeta(n, shape1 = 1, shape2 = 1)),
  dq2 = function(x) dnorm(x, m = 7800, sd = 1300),
  dL2 = P1993.likelihood2,
  dL1 = function(x) ROI.likelihood(getROI(x[1:2]))
)

# Posterior for the input
Theta.posterior <- data.frame(SpIR.posterior$theta.resamp)
names(Theta.posterior) <- c("P0", "MSYR", "alpha")
Theta.posterior <- data.frame(
  Theta.posterior,
  ROI = apply(Theta.posterior, 1, function(x) getROI(x[1:2]))
)
#
Theta.prior <- data.frame(
  P0 = P0.prior,
  MSYR = MSYR.prior,
  ROI = ROI.likSamples
```

```r
)
apply(Theta.prior, 2,
      function(x) c(mean = mean(x), quantile(x, probs = c(.025, .5, .975))))
```

```
##              P0        MSYR        ROI
## mean  16091.47 0.021975995 0.03070971
## 2.5%   8222.34 0.009622104 0.01445515
## 50%   14972.31 0.021086343 0.03066459
## 97.5% 30280.49 0.039355322 0.04727302
```

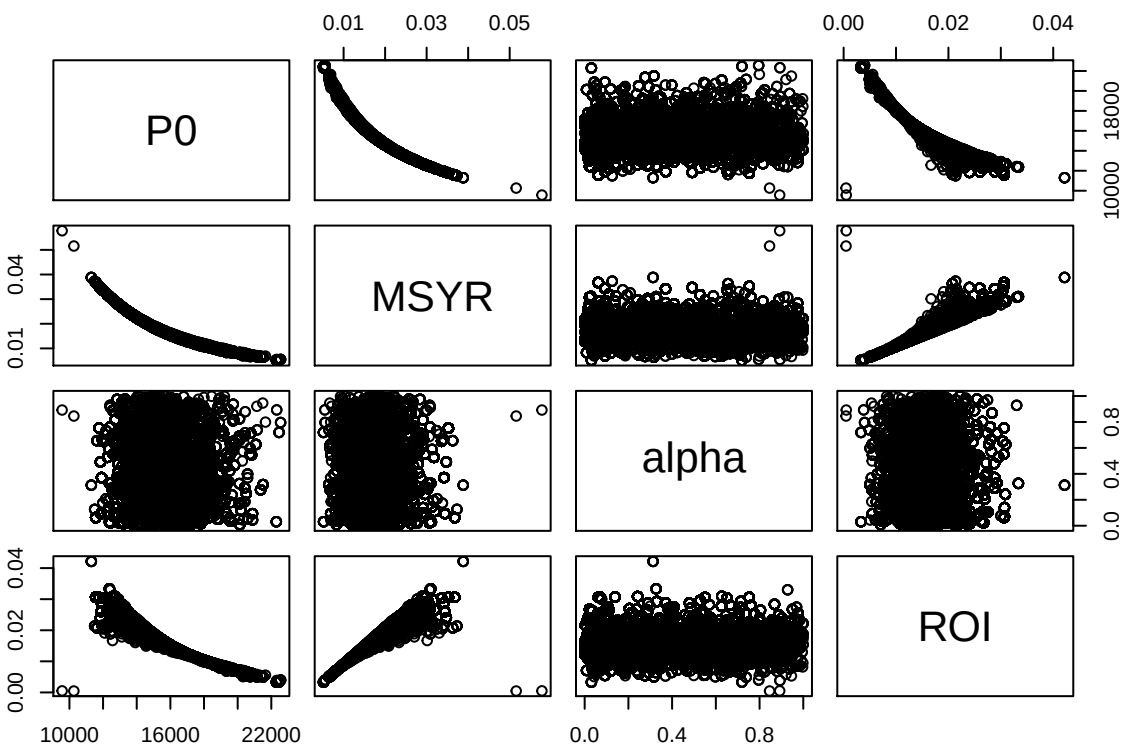
```r
apply(Theta.posterior, 2,
      function(x) c(mean = mean(x), quantile(x, probs = c(.025, .5, .975))))
```

```
##              P0       MSYR      alpha        ROI
## mean  14505.80 0.02142186 0.36469072 0.02018938
## 2.5%  11990.65 0.01133724 0.01598568 0.01045742
## 50%   14308.91 0.02125559 0.32942577 0.02026842
## 97.5% 17934.44 0.03361983 0.83993060 0.03037524
```

```r
cor(Theta.posterior)
```

```
##                 P0        MSYR       alpha         ROI
## P0     1.00000000 -0.97177767  0.05032405 -0.96768153
## MSYR  -0.97177767  1.00000000 -0.05455443  0.97017451
## alpha  0.05032405 -0.05455443  1.00000000 -0.05593104
## ROI   -0.96768153  0.97017451 -0.05593104  1.00000000
```

```r
plot(Theta.posterior)
```



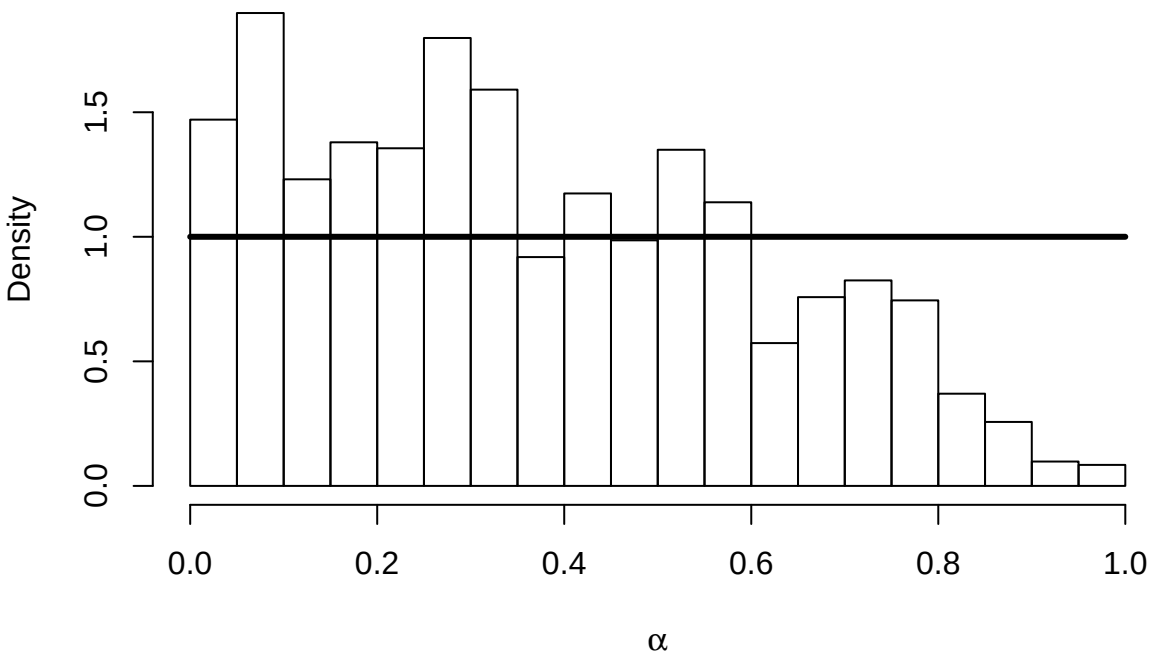

```r
# Now the output
P1993.posterior <- SpIR.posterior$phi.resamp
c(mean = mean(P1993.posterior), quantile(P1993.posterior, probs = c(.025, .5, .975)))
```

```
##     mean     2.5%      50%    97.5%
## 8172.917 7025.195 8191.215 9268.948
```

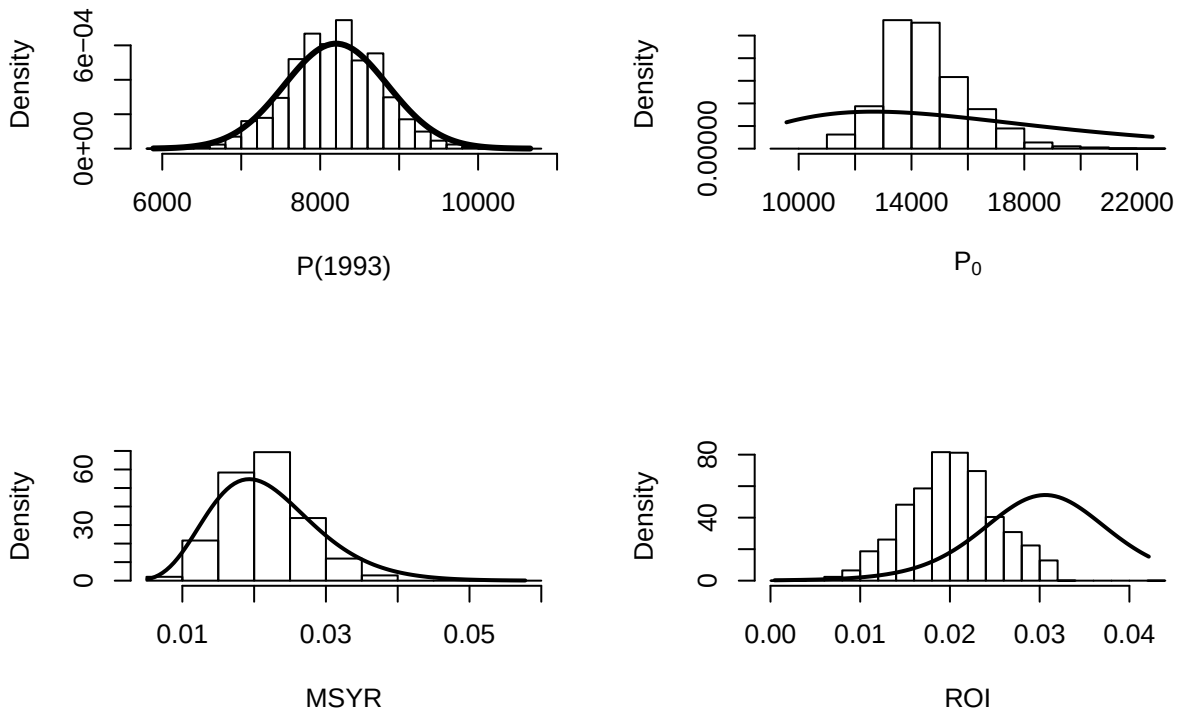
```r
c(mean = mean(P1993.prior), quantile(P1993.prior, probs = c(.025, .5, .975)))
```

```
##      mean      2.5%       50%     97.5%
##  7799.310  5245.779  7799.269 10326.781
```

```r
# Figure like Fig 5 in Poole & Raftery 2000

hist(Theta.posterior$alpha, probability = TRUE, xlab = expression(alpha),
     main = "")
curve(dbeta(x, 1, 1), add = TRUE, lwd = 3)
```




```r
par(mfrow = c(2,2))
hist(P1993.posterior, probability = TRUE, xlab = "P(1993)", main = "")
curve(dnorm(x, mean = 8200, sd = 654),
      min(P1993.posterior), max(P1993.posterior),
      add = TRUE, lwd = 3)

hist(Theta.posterior$P0, probability = TRUE, xlab = expression(P[0]),
     main = "")
curve(dgamma(x-6400, shape = 2.8085, rate = 0.0002886),
      min(Theta.posterior$P0), max(Theta.posterior$P0), lwd = 2, add = TRUE)

hist(Theta.posterior$MSYR, probability = TRUE, xlab = "MSYR",
     main = "")
curve(dgamma(x, shape = 8.2, rate = 372.7),
      min(Theta.posterior$MSYR), max(Theta.posterior$MSYR), lwd = 2, add = TRUE)

hist(Theta.posterior$ROI, probability = TRUE, xlab = "ROI",
     main = "")
curve(ROI.likelihood,
      min(Theta.posterior$ROI), max(Theta.posterior$ROI), lwd = 2, add = TRUE)
```


Density
0e+00
6e−04
6000
8000
10000
P(1993)
Density
0.00000
10000
14000
18000
22000
P0
Density
0
30
60
0.01
0.03
0.05
MSYR
Density
0
40
80
0.00
0.01
0.02
0.03
0.04
ROI