

# bowead\_example\_alpha\_sensitivity.r

max

Sun Jan 29 19:09:12 2017

```
#  $P_{t+1} = P_t - C_t + 1.5 * (MSYR) * P_t(1-(P_t/P_0)^2)$ 
# Inputs: P_0 and MSYR
# Output: P_{1993}
# Obtain the posterior via Sampling-importance-resampling (SpIR)
source("pooling_aux.r")
CtData <- read.csv("../data/bowhead_kills_1848-2002.csv")
CtData <- subset(CtData, year <= 1992)
#####
getPosterior <- function(M, Alpha){
  cat("Doing alpha = ", Alpha, "\n")
  Result <- vector(0, mode = "list")

  P1993.likelihood2 <- function(x) dnorm(x, m = 8293, sd = 626) ## L_2(\phi)
  #
  getROI <- function(pars){
    Ps <- getP(msyr = pars[2], p0 = pars[1],
               C = CtData$kills, Pinit = pars[1])
    P1978 <- Ps[length(Ps) - 15]
    P1993 <- Ps[length(Ps)]
    ROI <- (P1993/P1978)^(1/15) - 1
    return(ROI)
  }
  ROI.likelihood <- function(y, a = .0302, b = .0069, log = FALSE){
    if(is.nan(y) && log == FALSE) return(0)
    if(is.nan(y) && log == TRUE) return(-Inf)
    ## L_2(\phi)
    if(log){
      dt((log(y + 1) - a)/b, df = 8, log = TRUE) + log(abs(1/(b * (y + 1))))
    }else{
      dt((log(y + 1) - a)/b, df = 8) * abs(1/(b * (y + 1)))
    }
  }
  #
  getP <- function(msyr, p0, C, Pinit){
    TT <- length(C)
    P <- rep(NA, TT + 1)
    P[1] <- Pinit
    for(t in 1: TT) P[t+1] <- P[t] - C[t] + 1.5 * msyr * P[t]*(1-(P[t]/p0)^2)
    return(P)
  }
  #
  getP1993 <- function(theta){
    P <- getP(msyr = theta[2], p0 = theta[1],
               C = CtData$kills, Pinit = theta[1])
    return(P[length(P)])
  }
  #
```

```

SpIR.posterior <- LP.SpIR(
  k = M,
  l = round(.2 * M),
  Model = getP1993,
  rq1 = function(n) cbind(rgamma(n, shape = 2.8085, rate = 0.0002886) + 6400,
                           rgamma(n, shape = 8.2, rate = 372.7)),
  dq2 = function(x) dnorm(x, m = 7800, sd = 1300),
  dL2 = P1993.likelihood2,
  dL1 = function(x) ROI.likelihood(getROI(x)),
  alpha = Alpha
)
# Posterior for the input
Theta.posterior <- data.frame(SpIR.posterior$theta.resamp)
names(Theta.posterior) <- c("P0", "MSYR")
Theta.posterior <- data.frame(
  Theta.posterior,
  ROI = apply(Theta.posterior, 1, getROI)
)
#
Posterior <- data.frame(Theta.posterior,
                        P1993 = SpIR.posterior$phi.resamp)
Result$quantiles <- apply(Posterior, 2,
                          function(x) c(mean = mean(x), quantile(x, probs = c(.025, .5, .975))))

Result$correlation <- cor(Posterior)
return(Result)
}
####
alphas <- seq(0, 1, by = .05)
Sensitivity.Posterior <- lapply(alphas, function(x) getPosterior(M = 1e5, Alpha = x))

## Doing alpha = 0
## Doing alpha = 0.05
## Doing alpha = 0.1
## Doing alpha = 0.15
## Doing alpha = 0.2
## Doing alpha = 0.25
## Doing alpha = 0.3
## Doing alpha = 0.35
## Doing alpha = 0.4
## Doing alpha = 0.45
## Doing alpha = 0.5
## Doing alpha = 0.55
## Doing alpha = 0.6
## Doing alpha = 0.65
## Doing alpha = 0.7
## Doing alpha = 0.75
## Doing alpha = 0.8
## Doing alpha = 0.85
## Doing alpha = 0.9
## Doing alpha = 0.95
## Doing alpha = 1

```

```

formatDt <- function(x){
  d <- data.frame(parameter = colnames(x), t(x))
  names(d) <- c("parameter", "mean", "lwr", "median", "upr")
  return(d)
}
PostTab <- as.data.frame(data.table::rbindlist(
  lapply(Sensitivity.Posterior, function(x) formatDt(x$quantiles) ), idcol = TRUE))
PostTab$alpha <- alphas[PostTab$.id]
library(ggplot2)
Parplot <- ggplot(PostTab, aes(x = alpha, y = median, fill = parameter, col = parameter))
Parplot +
  geom_line() +
  scale_y_continuous("", expand = c(0, 0)) +
  scale_x_continuous(expression(alpha), expand = c(0, 0)) +
  geom_ribbon(aes(ymin = lwr, ymax = upr), alpha = .2) +
  facet_grid(parameter~., scales = "free_y") +
  theme_bw()

```

