```r
##############################################################################
## Title: Analysis of the sensitivity of model estimates to parameters      ##
##                                                                          ##
## Script author: Kali Z. Mattingly                                         ##
##                                                                          ##
## Date created: 08-31-2022                                                 ##
##                                                                          ##
## Email: mattingly.kali@epa.gov                                            ##
##                                                                          ##
## Citation:  Haselman, J.T., Nichols, J.W., Mattingly, K.Z., Hornung, M.W. ##
##            and Degitz, S.J. 2023. A Biologically Based Computational     ##
##            Model for the Hypothalamic-Pituitary-Thyroid (HPT) Axis in    ##
##            Xenopus laevis larvae. Mathematical Biosciences (submitted)   ##
##                                                                          ##
##############################################################################

##### Before running these analyses, run "Scripts/Amphibian_Thyroid_Model.R"
# needed to load packages, `dynmodel()`, `dynout`, `parameters` into environment #####

install.packages("reshape2")

library(reshape2)

options(scipen=500)

##### Calculate normalized sensitivity coefficient (NSC) #####

# original parameters
param_orig <- parameters

# parameters + 1%
addone <- function(value) {return(value+(value*0.01))}
param_plus1 <- addone(param_orig)

# original response variable values at time points of interest
#   times: 2, 4, 7 and 10 days (48, 96, 168 and 240 hours)
#   response variables: plasma T4 (CT4TOTPL), plasma T3 (CT3TOTPL), follicular
#   cell number (FCNTOT), gland MIT (TG.MIT.pmoles)
var_orig <- dynout[, c("time", "CT4TOTPL", "CT3TOTPL", "FCNTOT", "TG.MIT.pmoles")]
var_orig <- var_orig[var_orig$time==48 | var_orig$time==96 | var_orig$time==168
                     | var_orig$time==240, ]
var_orig <- melt(data.table(var_orig), id.vars=c("time"),
                 measure.vars=c("CT4TOTPL", "CT3TOTPL", "FCNTOT", "TG.MIT.pmoles"),
                 value.name="var_orig") # long format

# NSC function
NSC <- function(var_orig, var_plus1, param_orig, param_plus1)
  {return(((var_plus1 - var_orig) / var_orig) / ((param_plus1 - param_orig) / param_orig
  ))}

# initialize a csv for saving NSC output as we step through the loop
output <- c("parameter", "time_hr", "variable", "var_plus1_A", "var_orig_B",
            "param_plus1_C", "param_orig_D", "NSC")
write.table(t(output), "Output/Sensitivity output.csv", sep=",", col.names=FALSE)

# step through the 33 params
for (i in c(1:length(param_plus1))) {

  # take param_orig and sub out only this focal param_plus1[i]
  param_temp <- param_orig
  param_temp[i] <- param_plus1[i]

  # rerun dynmodel() with this subbed set of params
  out_temp <- ode(y = state, times = times, func = dynmodel, parms = param_temp, method =
   "rk4")
  dynout_temp <- as.data.frame(out_temp)
```

```
    colnames(dynout_temp) <- make.unique(names(dynout_temp))

    # extract the var_plus1 values of interest (in same format as var_orig)
    var_plus1 <- dynout_temp[, c("time", "CT4TOTPL", "CT3TOTPL", "FCNTOT", "TG.MIT.pmoles")]
    var_plus1 <- var_plus1[var_plus1$time==48 | var_plus1$time==96 | var_plus1$time==168 |
                             var_plus1$time==240, ]
    var_plus1 <- melt(data.table(var_plus1), id.vars=c("time"),
                      measure.vars=c("CT4TOTPL", "CT3TOTPL", "FCNTOT", "TG.MIT.pmoles"),
                      value.name="var_plus1")

    # match up orig and corresponding plus_1
    output <- merge(var_orig, var_plus1, by.x=c("time", "variable"), by.y=c("time", "variable"))
    colnames(output)[1] <- "time_hr"

    # add param_orig[i] and param_plus1[i]
    output$param_orig <- param_orig[i]
    output$param_plus1 <- param_plus1[i]

    # calculate the NSCs for that set of 4 vars x 4 time points
    output$NSC <- NSC(output$var_orig, output$var_plus1, output$param_orig, output$param_plus1)
    colnames(output)[3] <- paste(colnames(output)[3], "B", sep="_")
    colnames(output)[4] <- paste(colnames(output)[4], "A", sep="_")
    colnames(output)[5] <- paste(colnames(output)[5], "D", sep="_")
    colnames(output)[6] <- paste(colnames(output)[6], "C", sep="_")

    # save the name of the focal parameter that was being tested
    output$parameter <- names(param_plus1[i])

    # reorganize
    output <- output[,c(8,1,2,4,3,6,5,7)]

    # print these 16 rows as appended lines in the csv
    write.table(output, "Output/Sensitivity output.csv", append=TRUE, sep=",", col.names=FALSE)

  }
```