Supplementary Material S1

############################################

#####R script for the extended threshold model######

############################################

### data structure for the extended threshold model ###

#raw\_data (name of the table)

|  |  |  |
| --- | --- | --- |
| ID for the cow | Days in milk | Progesterone measures per day |
| 1122 | 6 | 0.4 |
| 1122 | 9 | 0.5 |
| 1122 | 13 | 0.5 |
| 1122 | 16 | 0.8 |
| 1122 | 20 | 0.8 |
| 1122 | 23 | 1.3 |
| … | … | … |
| N | … | … |

###First step, calculation of an interpolated time series for days in milk (DIM) ###

 >DIM\_min <- min (raw\_data$DIM) #DIM\_min=variable for the

#smallest value of DIM

 >DIM\_max <- max(raw\_data$DIM) #DIM\_max=variable for the

#largest value of DIM

 >DIMs <- seq(DIM\_min, DIM\_max, by = 1) #DIMs=variable for the

#sequence between DIM\_min

#and DIM\_max

###Second step, interpolation of the progesterone measures with the subfunction ###

###stinterp of the R package „stinepack“ ###

 > st\_interpol <- stinterp(raw\_data$DIM,raw\_data$progesterone,DIMs,

 method=c("scaledstineman"))

###Third step, calculate the first derivative of the interpolated progesterone time ###

###series with function stinemanSlopes of the R package stinepack###

 > st\_slopes<- stinemanSlopes(st\_interpol$x,st\_interpol$y)

###Fourth step, use the kolmogorov-zurbenko filter of the kza package to remove ###

###high frequency noise in the interpolated progesterone time series and the first derivative#

 > kz\_interpol <- kz(st\_interpol$y) #interpolated progesterone time series

 > kz\_slopes <- kz(st\_slopes) #first derivative

###Fifth step, create a matrix with calculated data ###

 > lacMa <- matrix(NaN,pr\_max,10) #matrix lacMa with 11 columns

 #define column names for the matrix

 >colnames(lacMa)<c('ID','DIM','progesterone','Interp\_progesterone','smoothed\_interp\_

 progesterone','first\_derivative','smoothed\_first\_derivative','zero\_crossings','estrus')

 #interpolate the day of lactation

 > lacMa[,"DIM"]<-seq(1,DIM\_max, by = 1)

 # add the lactation day into the matrix

 > for (run in 1:nrow(raw\_data))

 {

 > lacMa[raw\_data[run,"DIM"],"raw\_data"] <- raw\_data[run,"progesterone"]

#add the ID of the cow into the matrix

 > lacMa[DIM\_min:DIM\_max,"ID"] <- raw\_data[run,"ID"]

#add the interpolated time series data into #the matrix

 > lacMa[DIM\_min:DIM\_max,"progesterone\_interp"]<- round(st\_interpol$y, digits=2)

 > lacMa[DIM\_min:DIM\_max,"first\_derivative"] <- round(st\_slopes, digits=2)

#add the smoothed progesterone time #series into the matrix

 > lacMa[pr\_min:pr\_max,"smoothed\_progesterone\_interp"]<- round(kz\_interpol,

 digits=2)

 > lacMa[pr\_min:pr\_max,"smoothed\_first\_derivative"] <- round(kz\_slopes, digits=2)

###Define a running window to calculate the zero-crossings – detection of minima###

> for (run in (DIM\_min+4):(pr\_DIM-5))

 {print(run)

 > left\_neg <-lacMa[run-3,'smoothed\_first\_derviative']<0 and lacMa[run-

 2,'smoothed\_first\_derivative']<0 and lacMa[run-1,'smoothed\_first\_derivative']<0

 > right\_pos\_zero<-lacMa[run+3, 'smoothed\_first\_derivative']>=0 and

 lacMa[run+2,'smoothed\_first\_derivative']>=0 and

 lacMa[lrun+1,'smoothed\_first\_derivative']>=0

 > if (left\_neg and right\_pos\_zero) {

 lacMa[run,'zero\_crossings']<- 4 #add the position of minima

 }

 } #end of the running window

###Detection of an estrus event###

>for (run in 1:nrow(lacMa)) {

 if (lacMa[run,'zero\_crossing'] == '4' and lacMa[run,'progesterone\_interp'] <= '1.20') #if the first derivative is

 #zero and a minimum is

 #detected as well as the

 #interpolated progesterone

 #time series is less equal

 #1.2ng/ml skimmed milk

 {

 lacMa[run,'estrus'] <-6 } #then insert in column

 #’estrus‘ as indication a 6

 }

### print the matrix###

>print(lacMa)