

# ANEXO – Detalle del código

## Processing functions

### *Input/Output*

#### readCsv

```
readCsv<-function(fileName) {  
  d<-import(fileName,skip=100)  
  names(d)<-rawNames  
  attr(d,"info")<-readLines(fileName,n=100)  
  d<-tbl_df(d)  
  return(d)  
}
```

#### readBin

```
readBin<-function(fileName){  
  d<-read.bin(fileName)  
  d<-d$data.out  
  colnames(d)<-rawNames  
  d<-tbl_df(d)  
  return(d)  
}
```

#### writeRData

```
readBin<-function(fileName){  
  d<-read.bin(fileName)  
  d<-d$data.out  
  colnames(d)<-rawNames  
  d<-tbl_df(d)  
  return(d)  
}
```

### *Data formatting*

#### formatDate

```
formatDate<-function(d){  
  stopifnot(names(d)==rawNames)  
  d<-mutate(d,time=ymd_hms(changeDot(time)))  
  return(d)  
}
```

#### changeDot

```
changeDot<-function(d){  
  substring(d,20,20)<-"."  
  return(d)  
}
```

#### setStart

```
setStart<-function(d,startTime){  
  stopifnot(names(d)==rawNames,startTime>=0)  
  x<-subset(d,hour(time)==startTime)$time[1]  
  d<-subset(d,time>=x)  
  return(d)  
}
```

### setDuration

```
setDuration<-function(d,regDuration,unit="days"){
  stopifnot(names(d)==rawNames,regDuration>=0)
  d<-subset(d,as.numeric(d$time-d$time[1],units=unit)<regDuration)
}
```

### *Non-wear evaluation*

#### nonWearEvaluation

```
nonWearEvaluation<-function(d,samFreq=30,nonWearTime=30){

  stopifnot(samFreq>0,nonWearTime>0,names(d)==rawNames)

  #set non wear window length
  nonWearEval<-samFreq*nonWearTime*60

  #number of complete blocks
  blockNumber<-ceiling(nrow(d)/nonWearEval)

  #last block length
  lastBlockLength<-nrow(d)-(blockNumber-1)*nonWearEval

  #Preparing evaluation
  d<-tbl_df(d)
  dataNonWear<-d %>%
    mutate(nonWearInterval=ceiling(row_number()/nonWearEval))%>%
    group_by(nonWearInterval) %>%
    summarize(sd(ax),sd(ay),sd(az),
              max(ax)-min(ax),max(ay)-min(ay),max(az)-min(az))

  names(dataNonWear)<-nonWearNames

  ##Checking conditions (sd<0.003g or range<0.05g in at least 2 out of 3 axis)
  dataNonWear<-mutate(dataNonWear,
                      nonWear=(((Xsd<0.003)+(Ysd<0.003)+(Zsd<0.003)>1) |
                                ((Xran<0.05)+(Yran<0.05)+(Zran<0.05)>1))
                      )

  #Inserting non wear condition
  d<-mutate(d,nonWear=rep(dataNonWear$nonWear,
                         c(rep(nonWearEval,(blockNumber-1)),lastBlockLength))
           )

  return(d)
}
```

### seeNonWearEvaluation

```
seeNonWearEvaluation<-function(d,samFreq=30,nonWearTime=30){
  stopifnot(samFreq>0,nonWearTime>0,names(d)==rawNames)

  #set non wear window length
  nonWearEval<-samFreq*nonWearTime*60

  #number of complete blocks
  blockNumber<-ceiling(nrow(d)/nonWearEval)

  #last block length
  lastBlockLength<-nrow(d)-(blockNumber-1)*nonWearEval

  #Preparing evaluation
  dataNonWear<-d %>%
    tbl_df %>%
    mutate(nonWearInterval=ceiling(row_number()/nonWearEval) )%>%
    group_by(nonWearInterval) %>%
    summarize(min(time),sd(ax),sd(ay),sd(az),
              max(ax)-min(ax),max(ay)-min(ay),max(az)-min(az))

  names(dataNonWear)<-c("start",nonWearNames)

  ##Checking conditions (sd<0.003g or range<0.05g in at least 2 out of 3 axis)
  dataNonWear<-mutate(dataNonWear,
                      nonWear=(((Xsd<0.003)+(Ysd<0.003)+(Zsd<0.003)>1) |
                                ((Xran<0.05)+(Yran<0.05)+(Zran<0.05)>1))
                      )
  return(dataNonWear)
}
```

### *Epoch conversion*

#### convertEpoch

```
convertEpoch<-function(dNW,samFreq=30,epochLength=60){
  stopifnot(samFreq>0,epochLength>0)
  winLength<-samFreq*epochLength
  dataepoch<- dNW %>%
    tbl_df %>%
    mutate(epoch=ceiling(row_number()/winLength),
           EN=sqrt(ax^2+ay^2+az^2),
           ENMO=abs(sqrt(ax^2+ay^2+az^2)-1)
           ) %>%
    group_by(epoch) %>%
    summarize(min(time),mean(ax),mean(ay),mean(az),mean(lux),
              sum(button=="press"),mean(temp),sum(EN),sum(ENMO),
              sd(ax),sd(ay),sd(az),max(lux),any(nonWear)
              )
  names(dataepoch)<-epochNames
  return(dataepoch)
}
```

## Analysis functions

### *Activity intensities*

#### actZones

```
actZones<-function(dEP,cutpoints,labels){
  stopifnot(names(dEP)==epochNames,all(cutpoints>0))

  dataAct<-mutate(dEP,actZone=cut(dEP$ENMO,c(0,cutpoints,Inf),labels = labels))

  return(dataAct)
}
```

### *Bout analysis*

#### boutAnalysis

```
boutAnalysis<-function(dEP,cutpoint,duration,tolerance,boutEpoch){
  stopifnot(names(dEP)==epochNames,all(cutpoint>0),duration>0,tolerance>=0)

  thisBoutBegin<-0
  thisBoutDuration<-0

  boutBegin<-c()
  boutDuration<-c()
  boutSummary<-c()

  iniTol<-tolerance
  count<-0

  for (i in dEP$ENMO){
    count<-count+1

    if (i<cutpoint && thisBoutDuration==0) next

    if ((i<cutpoint && thisBoutDuration>0) |
        dEP$nonWear[count]){
      tolerance<-tolerance-1
    }

    if (i>=cutpoint && thisBoutDuration>0) {
      thisBoutDuration<-thisBoutDuration+1
    }

    if (i>=cutpoint && thisBoutDuration==0) {
      thisBoutDuration<-1
      thisBoutBegin<-dEP$time[count]
    }

    if (count>1){
      if(as.numeric(dEP$time[count]-dEP$time[count-1],unit="secs")>boutEpoch){
        tolerance<--1
      }
    }
  }
}
```

```

if (tolerance<0) {
  tolerance<-iniTol
  thisBoutDuration<-thisBoutDuration-1

  if (thisBoutDuration*boutEpoch/60>=duration) {
    boutDuration<-c(boutDuration,thisBoutDuration*boutEpoch/60)
    boutBegin<-c(boutBegin,thisBoutBegin)
  }

  thisBoutDuration<-0
}

}

boutSummary<-tbl_df(matrix(c(boutBegin,boutDuration),ncol=2))
colnames(boutSummary)<-c("Begin","Duration")
boutSummary<-mutate(boutSummary,
  Begin=as.POSIXct(boutSummary$Begin,
    origin="1970-01-01",
    tz="Europe/London"
  )
)
return(boutSummary)

}

```

### sedBoutAnalysis

```

sedBoutAnalysis<-function(dEP,cutpoint,duration,tolerance,boutEpoch){

  stopifnot(names(dEP)==epochNames,all(cutpoint>0),duration>0,tolerance>=0)

  thisBoutBegin<-0
  thisBoutDuration<-0

  boutBegin<-c()
  boutDuration<-c()
  boutSummary<-c()

  iniTol<-tolerance
  count<-0

  for (i in dEP$ENMO){
    count<-count+1

    if (i>=cutpoint && thisBoutDuration==0) next

    if ((i>=cutpoint && thisBoutDuration>0) |
      dEP$nonWear[count]){
      tolerance<-tolerance-1
    }
  }
}

```

```

if (i<cutpoint && thisBoutDuration>0) {
  thisBoutDuration<-thisBoutDuration+1
}

if (i<cutpoint && thisBoutDuration==0) {
  thisBoutDuration<-1
  thisBoutBegin<-dEP$time[count]
}

if (count>1){
  if(as.numeric(dEP$time[count]-dEP$time[count-1],unit="secs")>boutEpoch){
    tolerance<--1
  }
}

if (tolerance<0) {
  tolerance<-iniTol
  thisBoutDuration<-thisBoutDuration-1

  if (thisBoutDuration*boutEpoch/60>=duration) {
    boutDuration<-c(boutDuration,thisBoutDuration*boutEpoch/60)
    boutBegin<-c(boutBegin,thisBoutBegin)
  }

  thisBoutDuration<-0
}

}

boutSummary<-tbl_df(matrix(c(boutBegin,boutDuration),ncol=2))
colnames(boutSummary)<-c("Begin","Duration")
boutSummary<-mutate(boutSummary,
  Begin=as.POSIXct(boutSummary$Begin,
    origin="1970-01-01",
    tz="Europe/London"
  )
)
return(boutSummary)
}

```

### *Most and least active zones*

#### M5L5

```
M5L5<-function(d,res=10,win=120){
  stopifnot(win%%res==0,nrow(d)%%res==0)
  maxtime<-NULL
  maxENMO<-0
  mintime<-NULL
  minENMO<-Inf
  curtime<-NULL
  curENMO<-0

  nblock<-(nrow(d)-win)/res
  for(i in 0:nblock)
    curtime<-d[1+res*i,2]
    curENMO<-sum(d[(1+res*i):(res*i+win),10])

  if(curENMO>maxENMO) {
    maxENMO<-curENMO
    maxtime<-curtime}

  if(curENMO<minENMO) {
    minENMO<-curENMO
    mintime<-curtime}
}

return(
  matrix(
    c(
      c(mintime,maxtime,curtime),
      c(minENMO,maxENMO,curENMO)
    ),ncol=2
  )
)
```

### *Plotting results*

#### plotacel

```
plotacel<-function(zones,cutpoints=c(47.5*4,78.3*4,148.5*4)){
  g<-ggplot(zones,
    aes(
      x=as.numeric(time-zones$time[1],units="mins")-
        floor(as.numeric(time-zones$time[1],units="days"))*1440,
      y=ENMO)
  )
  g+geom_path()
  +geom_hline(aes(yintercept=cutpoints[1]),linetype="dashed",colour="red")
  +geom_hline(aes(yintercept=cutpoints[2]),linetype="dashed",colour="blue")
  +geom_hline(aes(yintercept=cutpoints[3]),linetype="dashed",colour="dark green")
  +facet_grid(day(zones$time)~.)
}
```

## Directory functions

### *Database creation*

#### zonedb

```
zonedb<-function(nonWearTime=30,
  samFreq=30,
  epochLength=60,
  cutpoints=c(47.5*4,78.3*4,148.5*4)*(samFreq/30)*(epochLength/60),
  labels=c("sedentary","light","moderate","vigorous")
)
{
  db<-tbl_df(matrix(rep(NA,27),ncol=27))

  colnames(db)<-c("Name",
    "Beginning day",
    "Beginning hour",
    "Non wear window (min)",
    "Sampling frequency (Hz)",
    "Epoch Length (s)",
    "Light cutpoint (mg)",
    "Moderate cutpoint (mg)",
    "Vigorous cutpoint (mg)",
    "Register duration (days)",
    "Non wear time (min)",
    "Sedentary time (min)",
    "Light activity time (min)",
    "Moderate activity time (min)",
    "Vigorous activity time (min)",
    "Weekend days registered",
    "Non wear time on weekend (min/day)",
    "Sedentary time on weekend (min/day)",
    "Light activity time on weekend (min/day)",
    "Moderate activity time on weekend (min/day)",
    "Vigorous activity time on weekend (min/day)",
    "Weekdays registered",
    "Non wear time on weekdays (min/day)",
    "Sedentary time on weekdays (min/day)",
    "Light activity time on weekdays (min/day)",
    "Moderate activity time on weekdays (min/day)",
    "Vigorous activity time on weekdays (min/day)"
  )

  for(i in list.files("../zone",full.names = TRUE)) {

    load(i)

    weekend<-filter(data,(wday(data$time)==7 | wday(data$time)==1))
    weekday<-filter(data,! (wday(data$time)==7 | wday(data$time)==1))

    weekendmin<-nrow(weekend)/1440
```



```
weekdaymin<-nrow(weekday)/1440
```

```
db<-rbind(db,c(
  substr(basename(i),1,nchar(basename(i))-15),
  format(data$time[1,"%d-%m-%Y"),
  format(data$time[1,"%H:%M"),
  nonWearTime,
  samFreq,
  epochLength,
  cutpoints[1],
  cutpoints[2],
  cutpoints[3],
  nrow(data)*(epochLength/60)/1440,
  sum(data$nonWear),
  nrow(filter(data,nonWear==FALSE,actZone==labels[1])),
  nrow(filter(data,nonWear==FALSE,actZone==labels[2])),
  nrow(filter(data,nonWear==FALSE,actZone==labels[3])),
  nrow(filter(data,nonWear==FALSE,actZone==labels[4])),
  weekendmin,
  sum(weekend$nonWear)/weekendmin,
  nrow(filter(weekend,nonWear==FALSE,actZone==labels[1]))/weekendmin,
  nrow(filter(weekend,nonWear==FALSE,actZone==labels[2]))/weekendmin,
  nrow(filter(weekend,nonWear==FALSE,actZone==labels[3]))/weekendmin,
  nrow(filter(weekend,nonWear==FALSE,actZone==labels[4]))/weekendmin,
  weekdaymin,
  sum(weekday$nonWear)/weekdaymin,
  nrow(filter(weekday,nonWear==FALSE,actZone==labels[1]))/weekdaymin,
  nrow(filter(weekday,nonWear==FALSE,actZone==labels[2]))/weekdaymin,
  nrow(filter(weekday,nonWear==FALSE,actZone==labels[3]))/weekdaymin,
  nrow(filter(weekday,nonWear==FALSE,actZone==labels[4]))/weekdaymin
)
)
}
```

```
db<-db[-1,]
writeRData(db,"../database/zones")
```

```
}
```

```

    boutdb
    boutdb<-function(samFreq=30,
                    epochLength=60,
                    boutCut=78.3*4*(samFreq/30)*(epochLength/60),
                    boutDur=10,
                    boutTol=2
                    )
{
    db<-tbl_df(matrix(rep(NA,10),ncol=10))

    colnames(db)<-c("Name",
                  "Sampling frequency (Hz)",
                  "Epoch Length (s)",
                  "Bout cutpoint (mg)",
                  "Minimum bout duration (min)",
                  "Tolerance (epoch)",
                  "Number of bouts",
                  "Longest bout (min)",
                  "Time of longest bout",
                  "Median bout duration (min)"
                  )

    for(i in list.files("../bout",full.names = TRUE)) {

        load(i)

        db<-rbind(db,c(
            substr(basename(i),1,nchar(basename(i))-15),
            samFreq,
            epochLength,
            boutCut,
            boutDur,
            boutTol,
            nrow(data),
            max(data$Duration),
            as.character(format(data[data$Duration==max(data$Duration),1],"%d-%m-%Y
%H:%M")),
            median(data$Duration)
            ))

    }

    db<-db[-1,]

    writeRData(db,"../database/bout")

}

```

```

sedBoutdb
sedBoutdb<-function(samFreq=30,
                    epochLength=60,
                    sedBoutCut=47.5*4*(samFreq/30)*(epochLength/60),
                    sedBoutDur=120,
                    sedBoutTol=2
                    )
{
  db<-tbl_df(matrix(rep(NA,10),ncol=10))

  colnames(db)<-c("Name",
                 "Sampling frequency (Hz)",
                 "Epoch Length (s)",
                 "Sed. Bout cutpoint (mg)",
                 "Minimum sed. bout duration (min)",
                 "Sed. tolerance (epoch)",
                 "Number of sed. bouts",
                 "Longest sed. bout (min)",
                 "Time of longest sed. bout",
                 "Median sed. bout duration (min)"
                 )

  for(i in list.files("../sedBout",full.names = TRUE)) {

    load(i)

    db<-rbind(db,c(
      substr(basename(i),1,nchar(basename(i))-18),
      samFreq,
      epochLength,
      sedBoutCut,
      sedBoutDur,
      sedBoutTol,
      nrow(data),
      max(data$Duration),
      as.character(format(data[data$Duration==max(data$Duration),1],"%d-%m-%Y
%H:%M")),
      median(data$Duration)
    ))

  }

  db<-db[-1,]

  writeRData(db,"../database/sedBout")

}

```

## Report generation

completereport (markdown template file)

---

```
title: "`r fileName`"
author: "Jorge Marín"
date: "`r Sys.Date()`"
output: html_document
```

---

```
```${r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```
```

### ##Measurement parameters

```
Measurement begins at: `r paste0(wday(zone$time[1],label=TRUE),"
",format(zone$time[1],"%d-%m-%Y %H:%M"))`
Register duration: `r regDuration` days
Sampling frequency: `r samFreq` Hz
Non-wear window: `r nonWearTime` mins
Epoch length: `r epochLength` s
Cutpoints: `r cutpoints`
Bout cutpoint: `r boutCut`
Bout duration: `r boutDur` s
Bout tolerance: `r boutTol` epoch
```

### ##Weekly results (mins)

#### \* Global

```
+ Non wear time: `r sum(zone$nonWear)`
+ Zone summary
```

```
```${r echo=FALSE}
summary(filter(zone,nonWear==FALSE)$actZone)*epochLength/60
```
```

#### \* Morning (8-16)

```
+ Non wear time: `r
sum(filter(zone,hour(zone$time)>=8,hour(zone$time)<=15)$nonWear)`
+ Zone summary
```

```
```${r echo=FALSE}
summary(filter(zone,hour(zone$time)>=8,hour(zone$time)<=15,nonWear==FALSE)$actZone)*epochLength/60
```
```

#### \* Afternoon (16-24)

```
+ Non wear time: `r sum(filter(zone,hour(zone$time)>=16)$nonWear)`
+ Zone summary
```

```
    ```${r echo=FALSE}
summary(filter(zone,hour(zone$time)>=16,nonWear==FALSE)$actZone)*epochLength/60
    ```
```

```
* Night (24-8)
+ Non wear time: `r sum(filter(zone,hour(zone$time)<=7)$nonWear)`
+ Zone summary
```

```
    ```${r echo=FALSE}
summary(filter(zone,hour(zone$time)<=7,nonWear==FALSE)$actZone)*epochLength/60
    ```
```

###Daily results (mins)

```
* Monday `r filter(zone,wday(zone$time)==2)$time[1]`
+ Non wear time: `r sum(filter(zone,wday(zone$time)==2)$nonWear)`
+ Zone summary
```

```
    ```${r echo=FALSE}
summary(filter(zone,wday(zone$time)==2,nonWear==FALSE)$actZone)*epochLength/60
    ```
```

```
* Tuesday `r filter(zone,wday(zone$time)==3)$time[1]`
+ Non wear time: `r sum(filter(zone,wday(zone$time)==3)$nonWear)`
+ Zone summary
```

```
    ```${r echo=FALSE}
summary(filter(zone,wday(zone$time)==3,nonWear==FALSE)$actZone)*epochLength/60
    ```
```

```
* Wednesday `r filter(zone,wday(zone$time)==4)$time[1]`
+ Non wear time: `r sum(filter(zone,wday(zone$time)==4)$nonWear)`
+ Zone summary
```

```
    ```${r echo=FALSE}
summary(filter(zone,wday(zone$time)==4,nonWear==FALSE)$actZone)*epochLength/60
    ```
```

```
* Thursday `r filter(zone,wday(zone$time)==5)$time[1]`
+ Non wear time: `r sum(filter(zone,wday(zone$time)==5)$nonWear)`
+ Zone summary
```

```
    ```${r echo=FALSE}
summary(filter(zone,wday(zone$time)==5,nonWear==FALSE)$actZone)*epochLength/60
    ```
```

```
* Friday `r filter(zone,wday(zone$time)==6)$time[1]`
+ Non wear time: `r sum(filter(zone,wday(zone$time)==6)$nonWear)`
+ Zone summary
```

```
```{r echo=FALSE}
summary(filter(zone,wday(zone$time)==6,nonWear==FALSE)$actZone)*epochLength/60
```
```

```
* Saturday `r filter(zone,wday(zone$time)==7)$time[1]`
+ Non wear time: `r sum(filter(zone,wday(zone$time)==7)$nonWear)`
+ Zone summary
```

```
```{r echo=FALSE}
summary(filter(zone,wday(zone$time)==7,nonWear==FALSE)$actZone)*epochLength/60
```
```

```
* Sunday `r filter(zone,wday(zone$time)==1)$time[1]`
+ Non wear time: `r sum(filter(zone,wday(zone$time)==1)$nonWear)`
+ Zone summary
```

```
```{r echo=FALSE}
summary(filter(zone,wday(zone$time)==1,nonWear==FALSE)$actZone)*epochLength/60
```
```

```
##Worst and best 2 hours
```

```
Worst: `r
```

```
as.POSIXct(M5L5(filter(zone,nonWear==FALSE,hour(zone$time)>=8))[1,1][[1]],origin="
1970-01-01",tz="Europe/London")`
```

```
Best: `r as.POSIXct(M5L5(filter(zone,nonWear==FALSE))[2,1][[1]],origin="1970-01-
01",tz="Europe/London")`
```

```
##Bouts
```

```
Total bouts: `r nrow(bout)`
```

```
Longest bout: `r max(bout$Duration)` mins
```

```
Median bout duration: `r median(bout$Duration)` mins
```

```
##Plot
```

```
```{r echo=FALSE}
```

```
plotacel(zone,cutpoints=cutpoints)
```

```
```
```

## *Global analysis*

### analysis

```
analysis<-function(fileName,
  startTime=0,
  regDuration=7,
  samFreq=30,
  nonWearTime=30,
  epochLength=60,
  cutpoints=c(47.5*4,78.3*4,148.5*4)*(samFreq/30)*(epochLength/60),
  labels=c("sedentary","light","moderate","vigorous"),
  boutCut=cutpoints[2],
  boutDur=10,boutTol=2,
  boutEpoch=epochLength,
  sedBoutCut=cutpoints[1],
  sedBoutDur=120,
  sedBoutTol=2
)
{
  stopifnot(boutEpoch==epochLength)
  raw<-readCsv(fileName = fileName)%>%
  formatDate%>%
  setStart(startTime)%>%
  setDuration(regDuration)

  epoch<-nonWearEvaluation(raw,samFreq=samFreq,nonWearTime =
nonWearTime)%>%
  convertEpoch(samFreq=samFreq,epochLength=epochLength)

  zone<-actZones(epoch,cutpoints=cutpoints,labels=labels)
  writeRData(zone,paste0("../zone/",fileName,"_zone"))

  bout<-boutAnalysis(epoch,
    cutpoint=boutCut,
    duration=boutDur,
    tolerance=boutTol,
    boutEpoch=epochLength
  )
  writeRData(bout,paste0("../bout/",fileName,"_bout"))

  sedBout<-sedBoutAnalysis(filter(epoch,hour(epoch$time)>=8),
    cutpoint=sedBoutCut,
    duration=sedBoutDur,
    tolerance=sedBoutTol,
    boutEpoch=epochLength
  )
  writeRData(sedBout,paste0("../sedBout/",fileName,"_sedBout"))

  render("../completereport.Rmd",
    output_file=paste0(fileName,".html"),
    output_dir="../reports")
}
```

```

complete
complete<-function(setdir,
  startTime=0,
  regDuration=7,
  samFreq=30,
  nonWearTime=30,
  epochLength=60,
  cutpoints=c(47.5,78.3,148.5)*4*(samFreq/30)*(epochLength/60),
  labels=c("sedentary","light","moderate","vigorous"),
  boutCut=cutpoints[2],
  boutDur=10,
  boutTol=2,
  boutEpoch=epochLength,
  sedBoutCut=cutpoints[1],
  sedBoutDur=120,
  sedBoutTol=2
)
{
  erFun<-function(e) {
    cat("Hay un error en",i,"\n")
    print(e)
    return(NULL)
  }
  oridir<-getwd()
  setwd(setdir)
  dir.create("../zone")
  dir.create("../bout")
  dir.create("../sedBout")
  dir.create("../reports")
  dir.create("../database")
  for (i in list.files()) {
    tryCatch(
      analysis(i,
        startTime = startTime,
        regDuration=regDuration,
        samFreq = samFreq,
        nonWearTime = nonWearTime,
        epochLength = epochLength,
        cutpoints=cutpoints,
        labels=labels,
        boutCut=boutCut,
        boutDur=boutDur,
        boutTol=boutTol,
        boutEpoch=boutEpoch,
        sedBoutCut=sedBoutCut,
        sedBoutDur=sedBoutDur,
        sedBoutTol=sedBoutTol
      ),
      error=erFun
    )
  }
}

```



```

zonedb(nonWearTime = nonWearTime,samFreq=samFreq,
      epochLength = epochLength,cutpoints = cutpoints,labels = labels)
boutdb(samFreq = samFreq,epochLength = epochLength,
      boutCut = boutCut,boutDur = boutDur,boutTol = boutTol)
sedBoutdb(samFreq = samFreq,epochLength = epochLength,
      sedBoutCut = sedBoutCut,sedBoutDur = sedBoutDur,sedBoutTol = boutTol)

setwd(oridir)
}

```

### completeMiniGUI

```

completeMiniGUI<-function(startTime=miniGUIScale(0,23,1),
      regDuration=7,
      samFreq=miniGUIMenuSel(c(10,20,25,30,40,50,60,66.7,75,85.7,100)),
      nonWearTime=miniGUIScale(5,60,1),
      epochLength=miniGUIScale(1,120,1),
      lightcutpoint=190,
      moderatecutpoint=313.2,
      vigorouscutpoint=594,
      boutCut=NULL,
      boutDur=miniGUIScale(5,30,1),
      boutTol=miniGUIScale(0,10,1),
      sedBoutCut=NULL,
      sedBoutDur=miniGUIScale(30,120,5),
      sedBoutTol=miniGUIScale(0,10,1)
    )
{
  cutpoints<-c(lightcutpoint,moderatecutpoint,vigorouscutpoint)
  if(is.null(boutCut)){boutCut<-cutpoints[2]}
  if(is.null(sedBoutCut)){sedBoutCut<-cutpoints[1]}
  setdir<-choose.dir()

  complete(setdir,
    startTime=startTime,
    regDuration=regDuration,
    samFreq=samFreq,
    nonWearTime=nonWearTime,
    epochLength=epochLength,
    cutpoints=cutpoints,
    labels=c("sedentary","light","moderate","vigorous"),
    boutCut=boutCut,
    boutDur=boutDur,
    boutTol=boutTol,
    boutEpoch=epochLength,
    sedBoutCut=sedBoutCut,
    sedBoutDur=sedBoutDur,
    sedBoutTol=sedBoutTol)
}

```