# Homeostasis

2022-11-26

## Loading data

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.7     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
#Get FaceStroopData
datadir<-"/work/285178/BodyFeelingExp_data"


#Find files
files<-list.files(datadir,pattern='^BodyFeelingExp_.+?csv',full.names=TRUE)

#Prepare an empty data frame for the data (also removes old version)
dataBFE<-data.frame()

#How many datasets were there
n_datasets_raw<-length(files)
#Prepare a variable to monitor how many datasets we keep
n_datasets<-0
#Prepare a variable to monitor how many points we originally had
n_datapoints_raw<-0

#Loop to go through all files in the list
for(iii in 1:n_datasets_raw){

  #remove old loaded file to not risk importing it multiple times
  if(exists('data_temp')) rm(data_temp)

  #Load data
  data_temp<-read.csv(files[iii])
  if(dim(data_temp)[2]==31){
    data_temp[1,6]<-data_temp[dim(data_temp)[1],6]
    data_temp<-data_temp[1,c(6,8:27)]
        if(length(colnames(dataBFE))==0){
          dataBFE=data_temp
          rm(data_temp)
          #counter to monitor included datasets
          n_datasets<-n_datasets+1
        }
        #Bind loaded data with actual data
        else {dataBFE<-rbind(dataBFE,data_temp)
          rm(data_temp)
          #counter to monitor included datasets
          n_datasets<-n_datasets+1
        }
    }
}

#A variable to monitor how many points we keep
n_datapoints<-length(dataBFE[,1])


#Make a variable which has hour and minutes of the day as decimal variable
dataBFE$hour2<-dataBFE$hour+(dataBFE$minute)/60


cf=1/24

dataBFE$sinCirc<-sin(2*pi*cf*dataBFE$hour2)
dataBFE$cosCirc<-cos(2*pi*cf*dataBFE$hour2)


ids = dataBFE %>% group_by(id) %>% summarize(n = n()) %>% filter(n>10)
dataBFE1 = dataBFE %>% filter(id %in% ids$id)
```

# Linear analysis:

```
library(lmerTest)
```

```
## Loading required package: lme4
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
##
## Attaching package: 'lmerTest'

## The following object is masked from 'package:lme4':
##
##     lmer

## The following object is masked from 'package:stats':
##
##     step
```

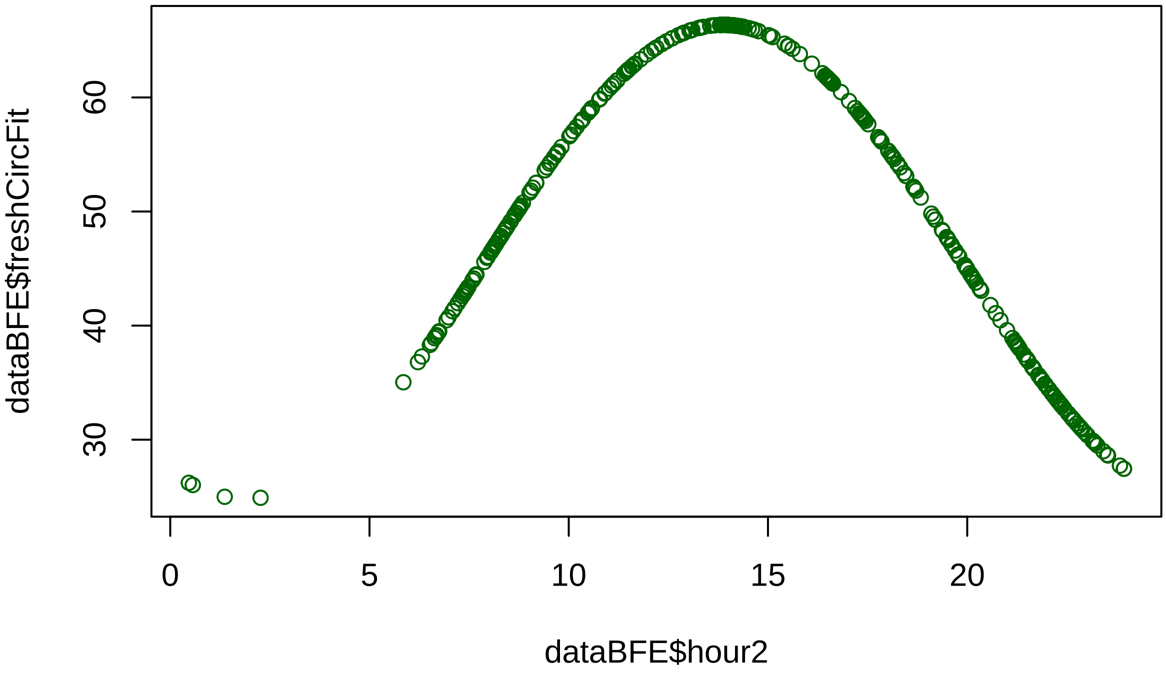
```
# Freshness: Simple oscillation model
modelBFEfreshCirc<-lmer(fresh~sinCirc+cosCirc+(cosCirc|id),data=dataBFE1)
```

```
## boundary (singular) fit: see help('isSingular')
```

```
m_temp<-summary(modelBFEfreshCirc)
m_temp
```

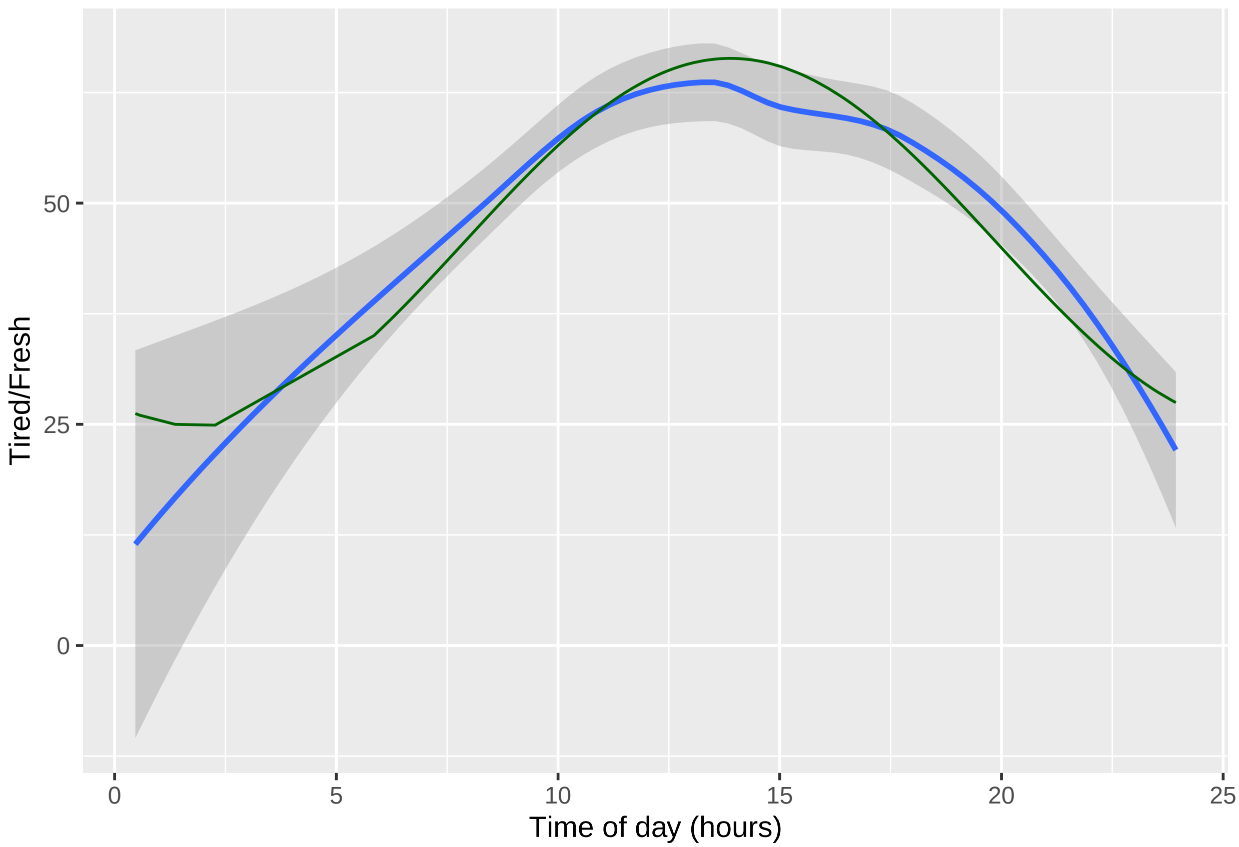
```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: fresh ~ sinCirc + cosCirc + (cosCirc | id)
##    Data: dataBFE1
##
## REML criterion at convergence: 2958.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.5620 -0.7128  0.0201  0.6864  3.4440
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  id       (Intercept)  86.32    9.291
##           cosCirc      22.17    4.709   -1.00
##  Residual             395.84   19.896
## Number of obs: 334, groups:  id, 13
##
## Fixed effects:
##             Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   45.582      2.882  13.036  15.819 6.89e-10 ***
## sinCirc       -9.836      1.807 322.874  -5.444 1.03e-07 ***
## cosCirc      -18.300      2.122  19.458  -8.625 4.46e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##         (Intr) sinCrc
## sinCirc  0.168
## cosCirc -0.473  0.266
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

```
dataBFE$freshCircFit<-m_temp$coefficients[1,1]+m_temp$coefficients[2,1]*dataBFE$sinCirc+m_temp$coeffici

plot(x=dataBFE$hour2,y=dataBFE$freshCircFit,type='p',col='darkgreen')
```



```
ggplot(dataBFE,aes(x=hour2,y=fresh))+geom_smooth()+geom_line(aes(x=hour2,y=freshCircFit),col='darkgreen
```

```
## ‘geom_smooth()‘ using method = ’loess’ and formula ’y ~ x’
```



```
A = sqrt((m_temp$coefficients[2,1])^2+(m_temp$coefficients[3,1])^2)
phi = atan(m_temp$coefficients[3,1]/m_temp$coefficients[2,1])

A
```

```
## [1] 20.77593
```

```
phi
```

```
## [1] 1.077605
```

#Non linear analyses: ##freshness

```
nform_lin_sin <- ~ (A*sin(1/24*hour2+phi)+k)
nfun_lin_sin <- deriv(nform_lin_sin,namevec=c("A","phi","k"),
              function.arg=c("hour2","A","phi","k"))



trig_lin_sin <- nlme::nlme(fresh ~ nfun_lin_sin(2*pi*hour2,A,phi,k),
                data = dataBFE1,
                fixed= A+phi+k ~ 1,
                random = k ~ 1,
                groups = ~ id,
                  start = c(A = 30, phi = pi, k = 50),
                  control = nlme::nlmeControl(lower=c(A = 0,phi = 0, k = 0), upper=c(A = -1,phi = 2*pi,

non_lin_model = summary(trig_lin_sin)

non_lin_model
```

```
## Nonlinear mixed-effects model fit by maximum likelihood
##   Model: fresh ~ nfun_lin_sin(2 * pi * hour2, A, phi, k)
##   Data: dataBFE1
##        AIC      BIC   logLik
##   2986.161 3005.216 -1488.08
##
## Random effects:
##  Formula: k ~ 1 | id
##                k Residual
## StdDev: 8.977717 20.13982
##
## Fixed effects:  A + phi + k ~ 1
##        Value Std.Error  DF  t-value p-value
## A   22.12994 1.9680553 319 11.24457       0
## phi  4.22864 0.0694869 319 60.85524       0
## k   44.87366 2.8284280 319 15.86523       0
##  Correlation:
##     A      phi
## phi -0.268
## k   -0.174  0.112
##
## Standardized Within-Group Residuals:
##        Min          Q1         Med          Q3         Max
## -2.5761140 -0.6763051  0.0459511  0.7000484  3.7459589
##
## Number of Observations: 334
## Number of Groups: 13
```

```
y_fresh = predict(trig_lin_sin)
```

##Mood

```
nform_lin_sin <- ~ (A*sin(1/24*hour2+phi)+k)
nfun_lin_sin <- deriv(nform_lin_sin,namevec=c("A","phi","k"),
              function.arg=c("hour2","A","phi","k"))



trig_lin_sin_mood <- nlme::nlme(mood ~ nfun_lin_sin(2*pi*hour2,A,phi,k),
                  data = dataBFE1,
                  fixed= A+phi+k ~ 1,
                  random = k ~ 1,
                  groups = ~ id,
                  start = c(A = 30, phi = pi, k = 50),
                  control = nlme::nlmeControl(lower=c(A = 0,phi = 0, k = 0), upper=c(A = -1,phi = 2*pi,
non_lin_model_mood = summary(trig_lin_sin_mood)

non_lin_model_mood
```

```
## Nonlinear mixed-effects model fit by maximum likelihood
##   Model: mood ~ nfun_lin_sin(2 * pi * hour2, A, phi, k)
##   Data: dataBFE1
##        AIC     BIC    logLik
##   2806.554 2825.61 -1398.277
##
## Random effects:
##  Formula: k ~ 1 | id
##                 k Residual
## StdDev: 9.540354  15.2181
##
## Fixed effects:  A + phi + k ~ 1
##        Value Std.Error  DF   t-value p-value
## A    4.83753 1.4603738 319  3.312528   0.001
## phi  3.32228 0.2480558 319 13.393285   0.000
## k   66.24303 2.8416695 319 23.311306   0.000
##  Correlation:
##     A      phi
## phi  0.301
## k   -0.135 -0.077
##
## Standardized Within-Group Residuals:
##        Min         Q1        Med         Q3        Max
## -3.3860607 -0.5842380  0.1746164  0.6616285  2.7598395
##
## Number of Observations: 334
## Number of Groups: 13
```

```
y_mood = predict(non_lin_model_mood)
```

##Hunger

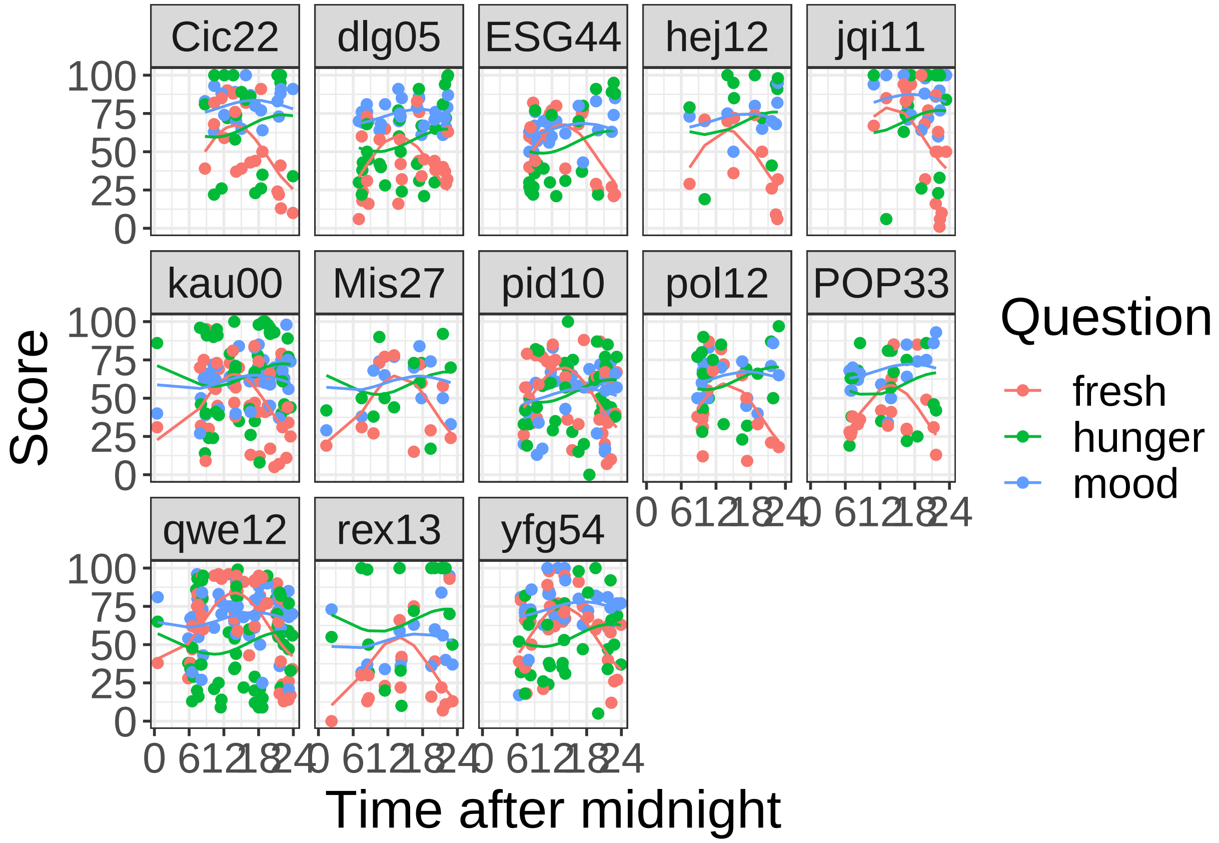
```
#Hunger
nform_lin_sin <- ~ (A*sin(1/24*hour2+phi)+k)
nfun_lin_sin <- deriv(nform_lin_sin,namevec=c("A","phi","k"),
              function.arg=c("hour2","A","phi","k"))



trig_lin_sin_hun <- nlme::nlme(hunger ~ nfun_lin_sin(2*pi*hour2,A,phi,k),
                  data = dataBFE1,
                  fixed= A+phi+k ~ 1,
                  random = k ~ 1,
                  groups = ~ id,
                  start = c(A = 30, phi = pi, k = 50),
                  control = nlme::nlmeControl(lower=c(A = 0,phi = 0, k = 0), upper=c(A = -1,phi = 2*pi,
non_lin_model_hun = summary(trig_lin_sin_hun)

non_lin_model_hun
```

```
## Nonlinear mixed-effects model fit by maximum likelihood
##   Model: hunger ~ nfun_lin_sin(2 * pi * hour2, A, phi, k)
##   Data: dataBFE1
##        AIC      BIC    logLik
##   3150.092 3169.148 -1570.046
##
## Random effects:
##  Formula: k ~ 1 | id
##                 k Residual
## StdDev: 7.114435 26.10037
##
## Fixed effects:  A + phi + k ~ 1
##        Value Std.Error  DF   t-value p-value
## A    7.38025 1.8726297 319  3.941116   1e-04
## phi  2.02913 0.3558102 319  5.702845   0e+00
## k   61.06086 2.5999133 319 23.485727   0e+00
##  Correlation:
##     A      phi
## phi -0.161
## k    0.061 -0.260
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -2.32373103 -0.81894157  0.07405686  0.88103986  1.96721438
##
## Number of Observations: 334
## Number of Groups: 13
```

```
y_hun = predict(non_lin_model_hun)

pred_data = data.frame(y_fresh, y_mood, y_hun)

pred_data = pred_data %>% rename(fresh = y_fresh, hunger = y_hun, mood = y_mood)

pred_data = pred_data %>% pivot_longer(cols = c("fresh", "mood", "hunger"), names_to = "Question",values
```

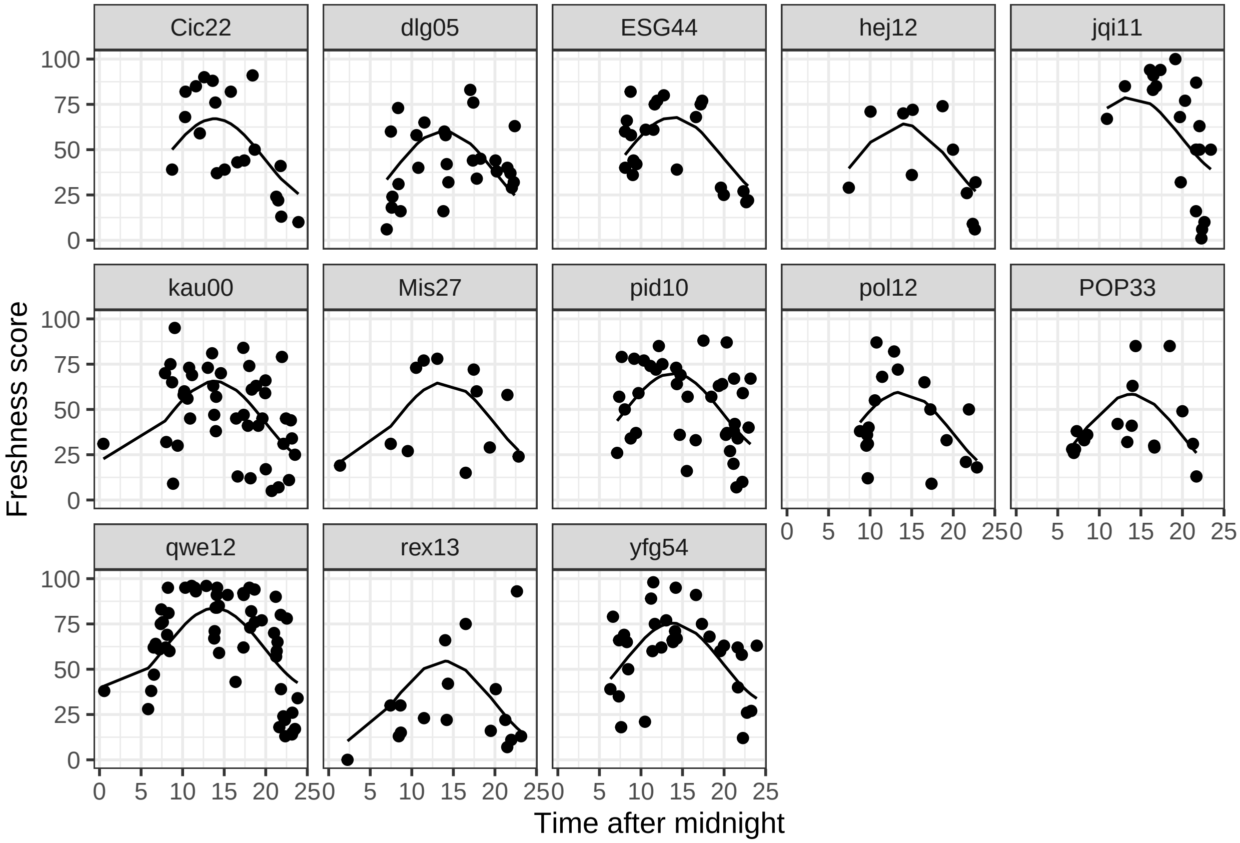
#Visualization

```
dataBFE2 = dataBFE1 %>% pivot_longer(cols = c("mood","fresh","hunger"), names_to = "Question", values_to

pred_data$hour2 = dataBFE2$hour2
pred_data$id = dataBFE2$id

pred_data %>% ggplot(aes(x = hour2, y = prediction, col = Question))+facet_wrap(~id, nrow = 3)+theme_bw
```

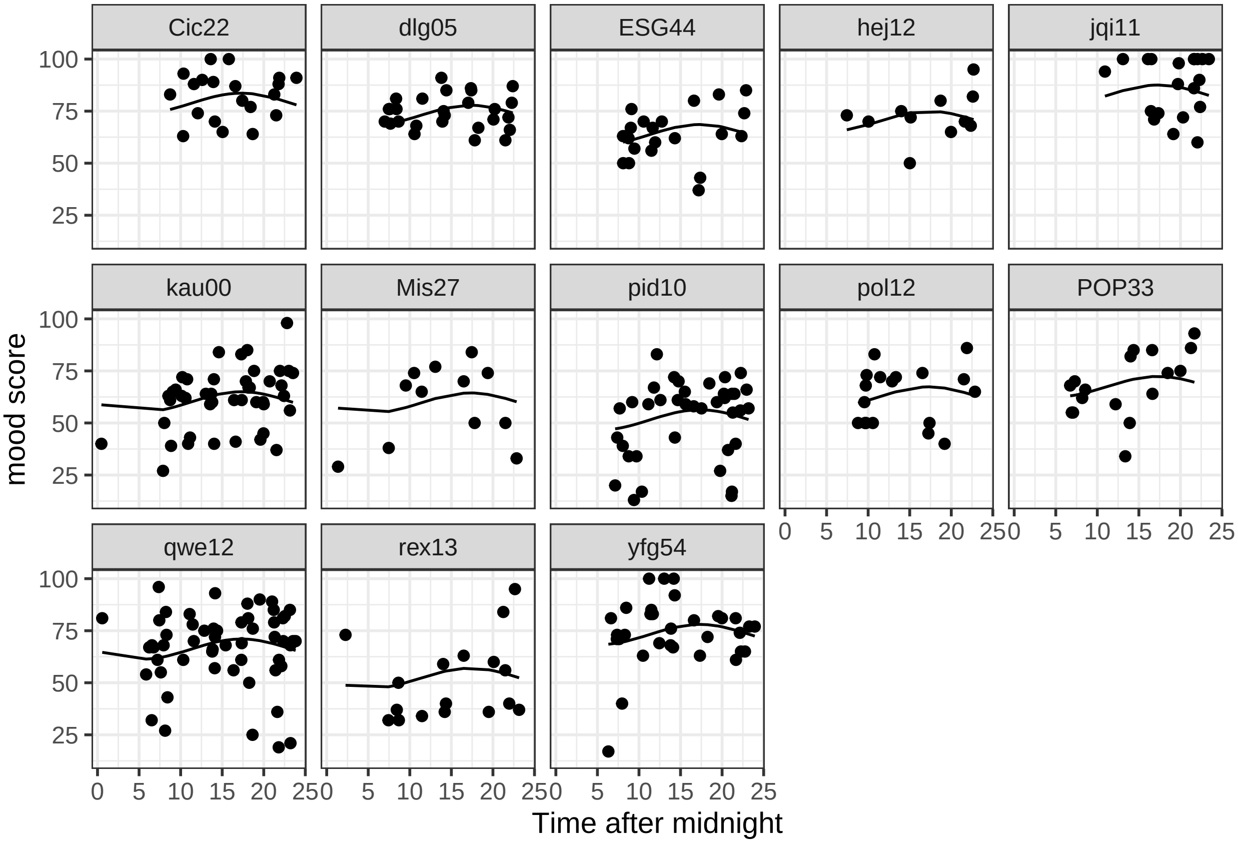


#single results visualization

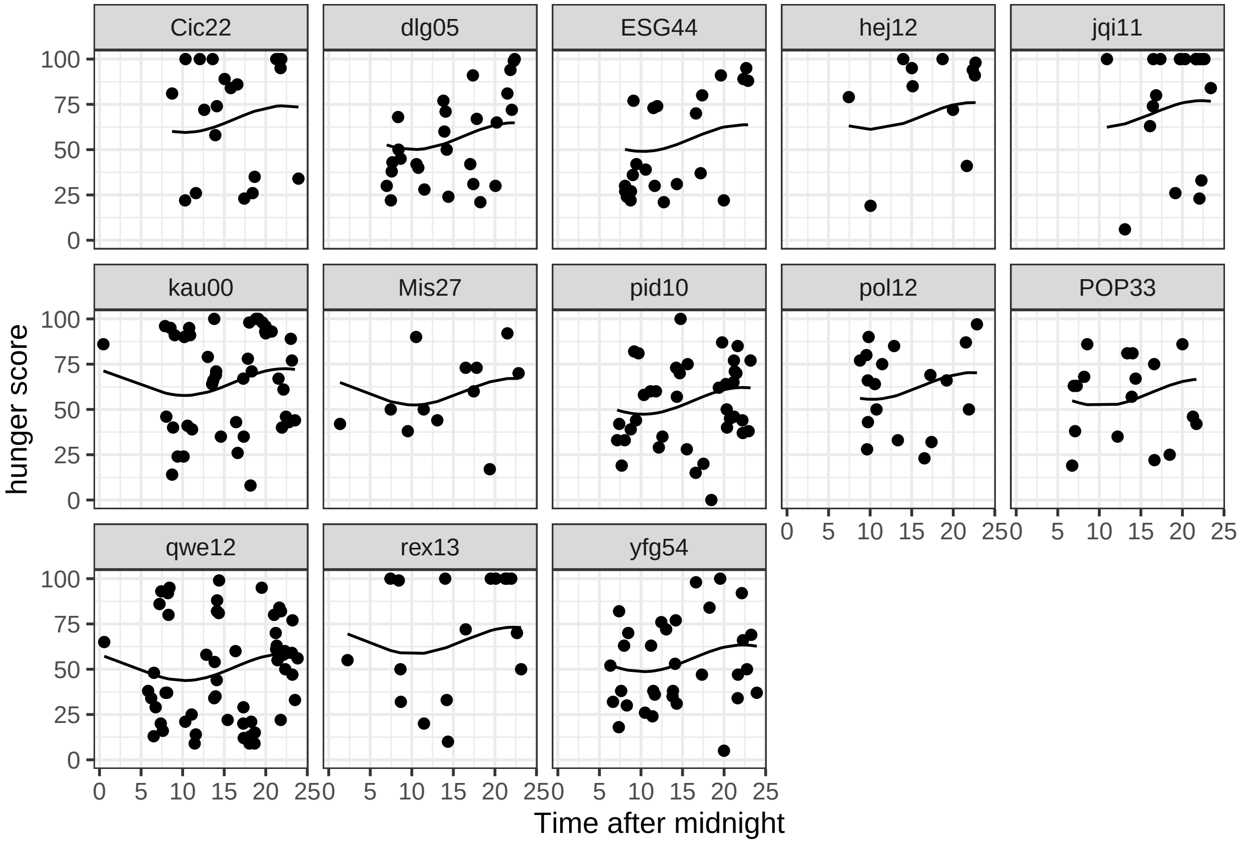
```
dataBFE1 %>% ggplot(aes(x = hour2, y = y_fresh))+geom_line()+facet_wrap(~id, nrow = 3)+theme_bw()+geom_p
```



```
dataBFE1 %>% ggplot(aes(x = hour2, y = y_mood))+geom_line()+facet_wrap(~id, nrow = 3)+theme_bw()+geom_p
```



```
dataBFE1 %>% ggplot(aes(x = hour2, y = y_hun))+geom_line()+facet_wrap(~id, nrow = 3)+theme_bw()+geom_po
```


Cic22
dlg05
ESG44
hej12
jqi11
kau00
Mis27
pid10
pol12
POP33
qwe12
rex13
yfg54
hunger score
0
25
50
75
100
0
5
10
15
20
25
Time after midnight