

# Synchrony in Psychotherapy, example with F1044 patient data

*Thomas Gargot*

*April, 26th 2016*

## Contents

Import data . . . . .	2
<b>Lists</b>	<b>2</b>
Functions list . . . . .	2
Constants generated from data and defining it (data list, ) . . . . .	3
Merge data frame, compute Time in minutes, compute log of motion history dataframe . . . . .	3
<b>Presentation of the data</b>	<b>4</b>
<b>Length of the videos in minutes</b>	<b>6</b>
Number of Available (True) and Not Available (False) data for each participant . . . . .	9
<b>Global Motion history</b>	<b>13</b>
Mean Motion history by video by participant . . . . .	13
Raw Motion history by video by participant . . . . .	13
Normalized log Motion history by video by participant . . . . .	17
<b>Raw data and mean of Motion History on sliding and non overlapping intervals on 1st video</b>	
<b>F1044C1 video</b>	<b>21</b>
Raw data . . . . .	21
Sliding interval . . . . .	23
Non overlapping interval . . . . .	24
<b>Focus on the motion history of the first 10 seconds of the first video</b>	<b>25</b>
Sliding interval function on a raw data, 5 frames interval . . . . .	25
Sliding interval function on log data, 5 frames interval . . . . .	26
Motion history of the father during 10-20 seconds of the first video(C) . . . . .	28
<b>Mean motion history by minute plots</b>	<b>30</b>
<b>Mean log motion history by minute plots</b>	<b>51</b>
<b>Motion history by minute for the F1044C video</b>	<b>72</b>
<b>Export no log filtered data in text files</b>	<b>100</b>
<b>Export log filtered data in text files</b>	<b>101</b>
<b>SyncPy utilisation for creating synchrony dataframe</b>	<b>102</b>
Description of SSI data frame . . . . .	104
<b>Synchrony scores for each dyad, triad and for the whole group</b>	<b>104</b>

<b>Synchrony scores(log data) for each dyad, triad and for the whole group</b>	<b>123</b>
Evolution of synchrony through time, raw each second . . . . .	144
Evolution of synchrony through time, mean by minute . . . . .	154
Evolution of synchrony through time, mean by 10 minutes . . . . .	164
Models of synchrony . . . . .	164

```
rm(list = ls(all.names = TRUE))
```

```
setwd("/Users/Ofix/Documents/Fac/internat/Recherche/projets/synchro/synchroData/Git/INCANT/Reports/")
```

## Import data

```
data <- importdata(fullNameList)
```

## Lists

### Functions list

#### MeanMotionByTime

Function that takes raw motion history data and compute the mean on a given interval. Intervals don't overlap, so the frequency of the data change (from 25 frames by seconde to 25 frames/interval by second).

#### Arguments:

- subject : Subject studied (patient, mother, father or therapist)
- indexOfvideos : List of videos studied (element eg 3 or list eg 1:3 or c(1,2,4))
- interval : number of frames in the studied interval
- data : data frame where there is data

```
## Revoir nom des variables : pas clair, faire un schéma
MeanMotionByTime <- function(subject, indexOfvideos=1:NumberOfvideos, interval, data){
  x <- c()
  for (file in indexList[indexOfvideos]){
    dataVector <- data[which(data$indexList==file), subject]
    ## with ceiling : superior limit of the round
    IntervalNumbersVideo <- ceiling(length(dataVector)/interval)
    for (i in 1:IntervalNumbersVideo){
      borneinf<- 1+(i-1)*interval
      bornesup <-i*interval
      dataVectorInterval <- dataVector[borneinf:bornesup]
      mean <- mean(dataVectorInterval, na.rm=TRUE)
      x <- c(x, mean)}}
  return (x)}
```

#### Slidinginterval

Function that takes raw motion history data and compute the mean on a given interval. The interval overlap, so the frequency of the data don't change. It stays at 25 frames/s.

### Arguments:

- subject : subject studied (patient, mother, father or therapist)
- indexOfvideos : list of videos studied (element eg. 3 or list eg 1:3 or c(1,2,4))
- interval : number of frames in the studied interval
- data : data frame where there is data

```
## faire un schéma
SlidingInterval <- function(subject, indexOfvideos=1:NumberOfvideos, interval, data)
{x <- c()}
for (file in indexList[indexOfvideos]){
  dataVector <- data[which(data$indexList==file), subject]
  NBofAnalysedFrames <- length(dataVector)-interval+1
  for (i in 1:NBofAnalysedFrames){
    borneinf<- (i)
    bornesup <-(interval-1+i)
    dataVectorInterval <- dataVector[borneinf:bornesup]
    mean <- mean(dataVectorInterval, na.rm=TRUE)
    x <- c(x, mean)}}
return (x)}
```

### MeanSynchronyByTime (TODO)

### Constants generated from data and defining it (data list, )

```
labelvideolist <-c()
for (i in indexList){
  a <- str_count(i)
  name <- substr(i, 6, a)
  labelvideolist <- c(labelvideolist, name)
}

FileName <- data.frame(unique(data$file), filesList, indexList, labelvideolist)
NumberOfvideos <- length(indexList)
```

### Merge data frame, compute Time in minutes, compute log of motion history dataframe

```
data <- merge(data, FileName, by.x="file", by.y="unique.data.file.", all=TRUE)
data$timeMin <- data$frame/(25*60)

data$fatherShifted <- data$father + min(data$father[which (data$father >0))]/2
data$logFather <- log(data$fatherShifted)

#log(x -min(x)+1)
data$motherShifted <- data$mother + min(data$mother[which (data$mother >0))]/2
data$logMother <- log(data$motherShifted)

data$patientShifted <- data$patient + min(data$patient[which (data$patient >0))]/2
data$logPatient <- log(data$patientShifted)

data$therapistShifted <- data$therapist + min(data$therapist[which (data$therapist >0))]/2
```

```

data$logTherapist <- log(data$therapistShifted)

# Name of the patient (anonymised)
data$family <- substr (data$indexList, 1, 5)

famList <- unique(data$family)

# Add date TODO

data$file <- NULL
data$filesList <- NULL

# Reorganize the data frame
data <- data[c("family", "indexList", "labelvideolist", "frame", "timeMin", "father", "fatherShifted",

```

- Time in minutes The timeMin is calculated with a frame rate of 25/sec.
- Motion history distribution The data is not normal at all but with very small movement very frequent and bigger movement much rare with a long tail.

To normalize the distribution to compute synchrony scores on it, we made the Napierian logarithm. It produces negative numbers. SyncPy can't compute negatives scores, they are so shifted to positives values with an arbitrary value of 20 to avoid to keep extreme negative values.

Values equal to 0 can't be logged. They generate a -Inf value. These values are set to NA. We lose the information of no movement at all. If we give a arbitrary value to this data (eg, the minimum value, they are over represented)

## Presentation of the data

```

str(data)

## 'data.frame':   948687 obs. of  17 variables:
## $ family      : chr  "F1002" "F1002" "F1002" "F1002" ...
## $ indexList   : Factor w/ 41 levels "F1002A1","F1002A2",...: 1 1 1 1 1 1 1 1 1 1 ...
## $ labelvideolist : Factor w/ 30 levels "A1","A2","B1",...: 1 1 1 1 1 1 1 1 1 1 ...
## $ frame       : int   1 2 3 4 5 6 7 8 9 10 ...
## $ timeMin      : num   0.000667 0.001333 0.002 0.002667 0.003333 ...
## $ father       : num   0.004025 0.002826 0.00207 0.002267 0.000821 ...
## $ fatherShifted : num   0.004032 0.002833 0.002077 0.002274 0.000829 ...
## $ logFather     : num   -5.51 -5.87 -6.18 -6.09 -7.1 ...
## $ mother        : num   0.0789 0.0764 0.0769 0.0794 0.0799 ...
## $ motherShifted : num   0.0789 0.0764 0.0769 0.0794 0.0799 ...
## $ logMother      : num   -2.54 -2.57 -2.56 -2.53 -2.53 ...
## $ patient       : num   NA NA NA NA NA NA NA NA NA NA ...
## $ patientShifted : num   NA NA NA NA NA NA NA NA NA NA ...
## $ logPatient     : num   NA NA NA NA NA NA NA NA NA NA ...
## $ therapist     : num   4.24e-05 0.00 0.00 4.24e-05 0.00 ...
## $ therapistShifted: num   6.37e-05 2.12e-05 2.12e-05 6.37e-05 2.12e-05 ...
## $ logTherapist   : num   -9.66 -10.76 -10.76 -9.66 -10.76 ...

summary(data)

```

```

##      family      indexList      labelvideolist      frame

```

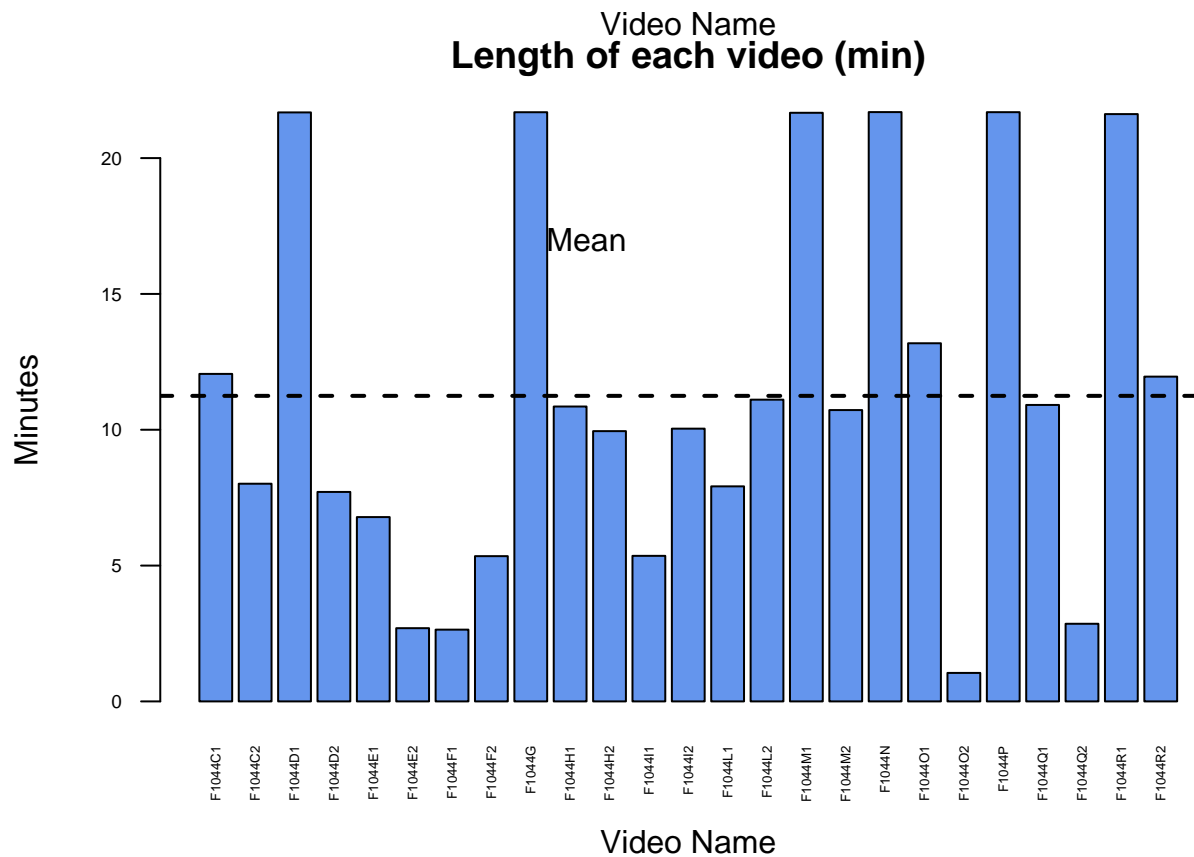
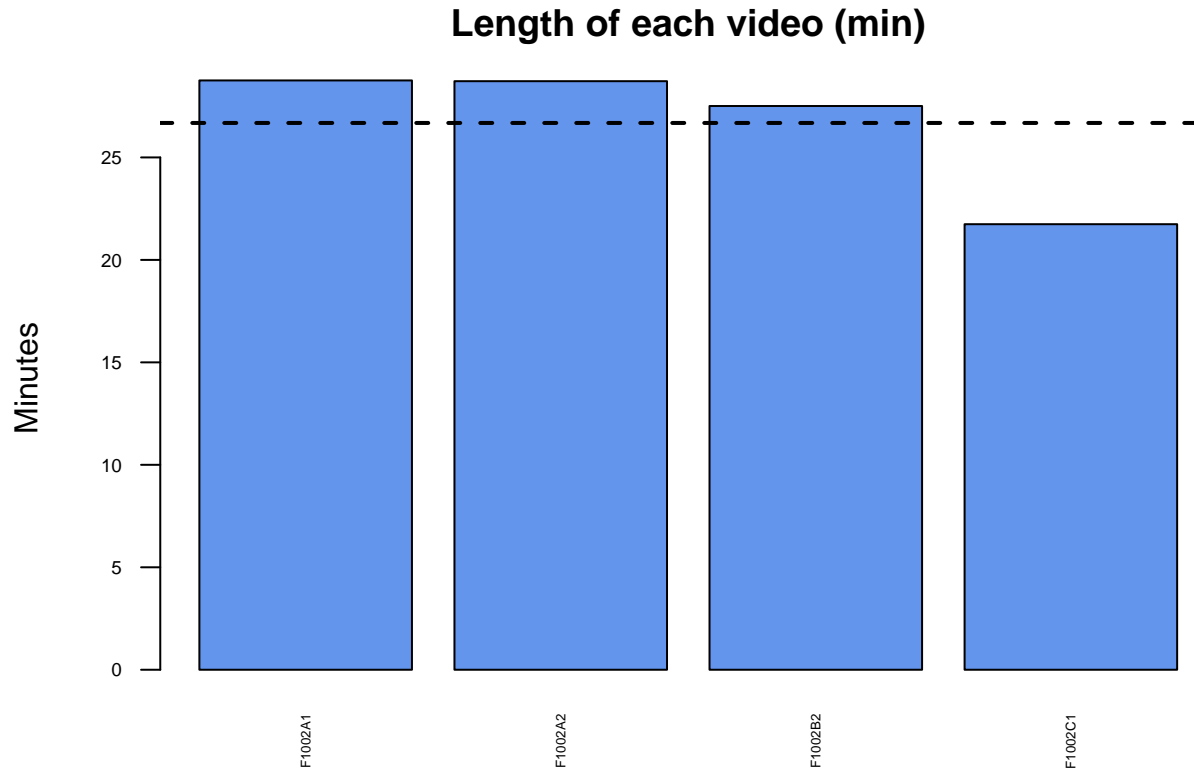
```

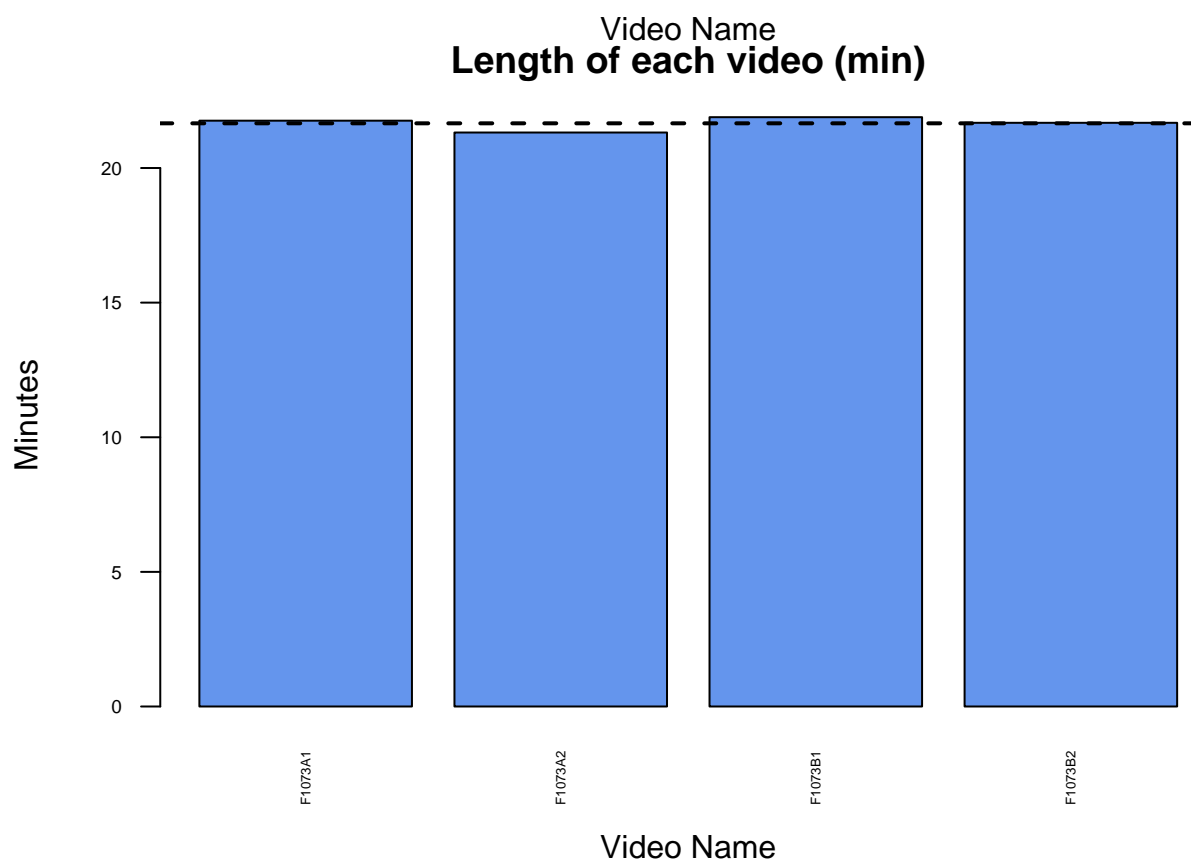
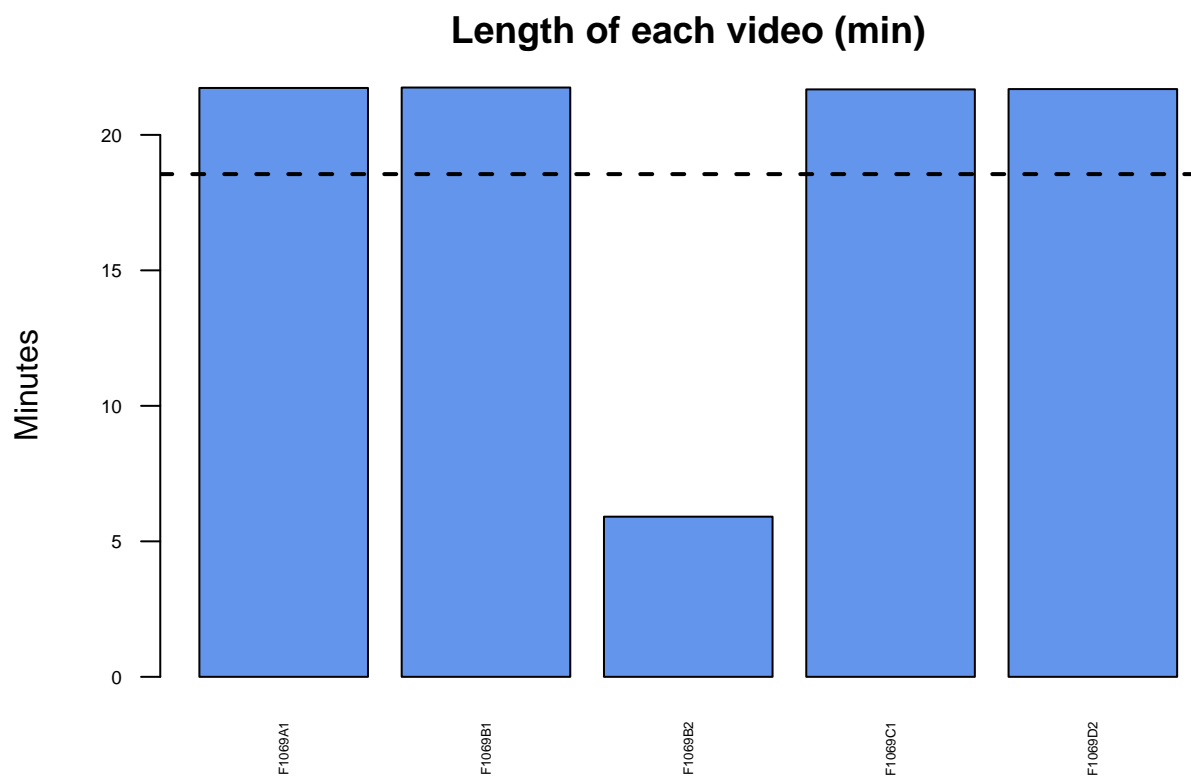
## Length:948687      F1002A1: 43136      A1      :108371      Min.      :    1
## Class :character    F1002A2: 43082      A2      :107700      1st Qu.: 6036
## Mode  :character    F1002B2: 41265      C1      : 83216      Median :12998
##                      F1073B1: 32832      B2      : 82648      Mean   :14660
##                      F1101A2: 32641      B1      : 65453      3rd Qu.:22663
##                      F1073A1: 32637      C2      : 44538      Max.   :43136
##                      (Other):723094      (Other):456761
##      timeMin          father          fatherShifted      logFather
## Min.      : 0.000667      Min.      :0.0      Min.      :0.0      Min.      : -11.8
## 1st Qu.: 4.024000      1st Qu.:0.0      1st Qu.:0.0      1st Qu.: -11.8
## Median : 8.665333      Median :0.0      Median :0.0      Median : -9.0
## Mean   : 9.773022      Mean   :0.0      Mean   :0.0      Mean   : -8.5
## 3rd Qu.:15.108667      3rd Qu.:0.0      3rd Qu.:0.0      3rd Qu.: -5.8
## Max.   :28.757333      Max.   :0.3      Max.   :0.3      Max.   : -1.3
##                      NA's      :593390      NA's      :593390      NA's      :593390
##      mother          motherShifted      logMother          patient
## Min.      :0.00      Min.      :0.00      Min.      : -11.99      Min.      :0.0
## 1st Qu.:0.00      1st Qu.:0.00      1st Qu.: -10.66      1st Qu.:0.0
## Median :0.00      Median :0.00      Median : -8.55      Median :0.0
## Mean   :0.01      Mean   :0.01      Mean   : -8.26      Mean   :0.0
## 3rd Qu.:0.00      3rd Qu.:0.00      3rd Qu.: -5.64      3rd Qu.:0.0
## Max.   :0.41      Max.   :0.41      Max.   : -0.88      Max.   :0.3
## NA's      :218579      NA's      :218579      NA's      :218579      NA's      :330670
## patientShifted      logPatient          therapist          therapistShifted
## Min.      :0.0      Min.      : -12.1      Min.      :0.0      Min.      :0.0
## 1st Qu.:0.0      1st Qu.: -9.9      1st Qu.:0.0      1st Qu.:0.0
## Median :0.0      Median : -6.8      Median :0.0      Median :0.0
## Mean   :0.0      Mean   : -7.4      Mean   :0.0      Mean   :0.0
## 3rd Qu.:0.0      3rd Qu.: -4.9      3rd Qu.:0.0      3rd Qu.:0.0
## Max.   :0.3      Max.   : -1.3      Max.   :0.3      Max.   :0.3
## NA's      :330670      NA's      :330670      NA's      :123359      NA's      :123359
##      logTherapist
## Min.      : -10.76
## 1st Qu.: -7.61
## Median : -6.06
## Mean   : -6.73
## 3rd Qu.: -5.48
## Max.   : -1.20
## NA's      :123359

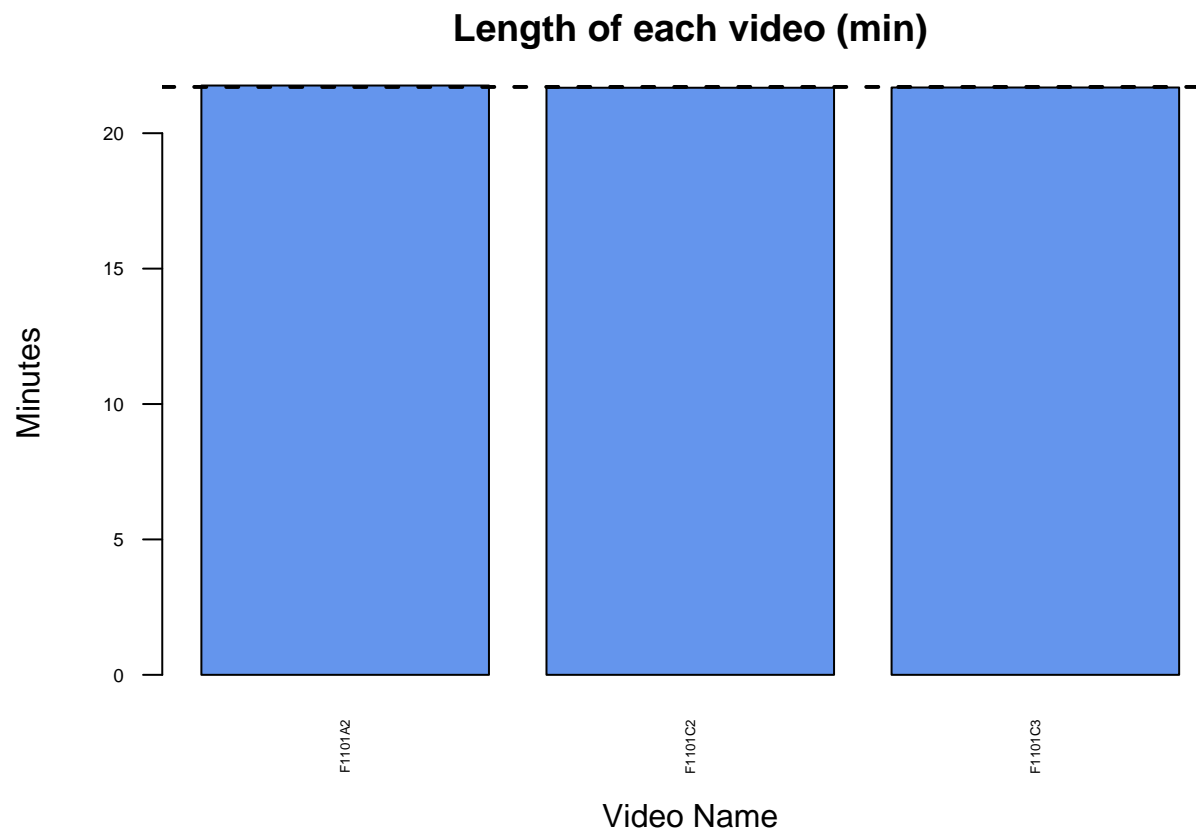
```

[View\(data\)](#)

## Length of the videos in minutes



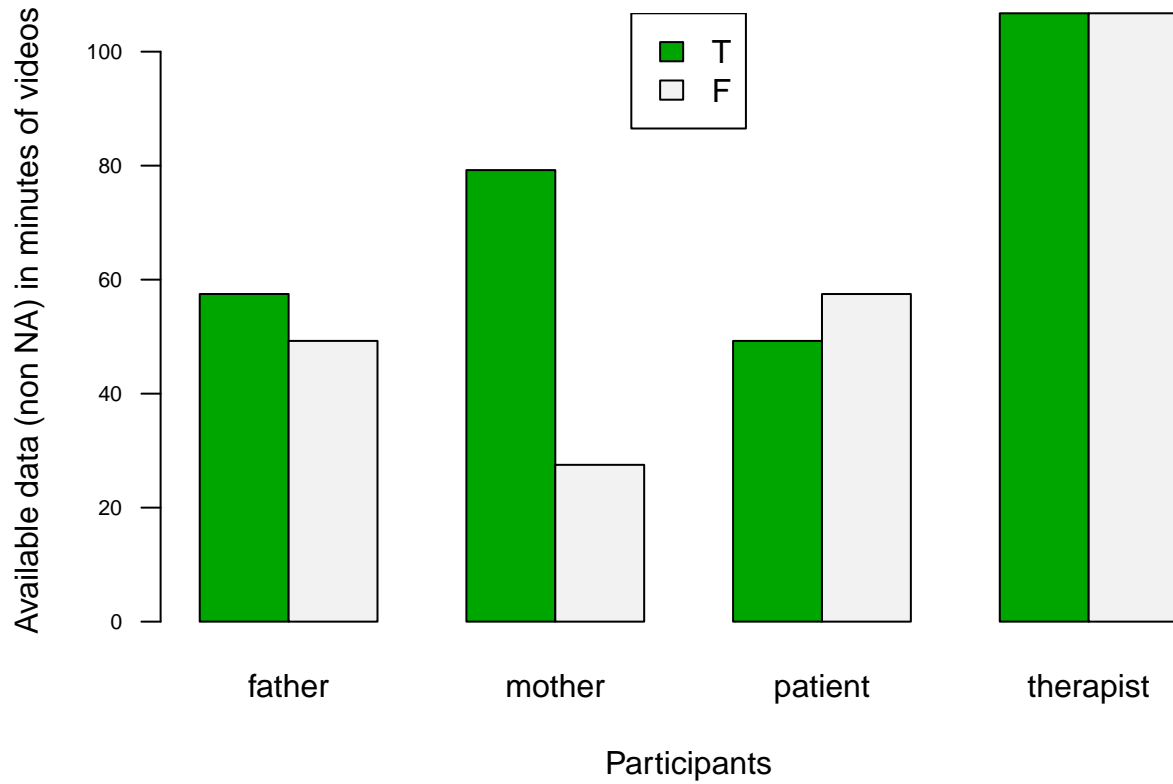




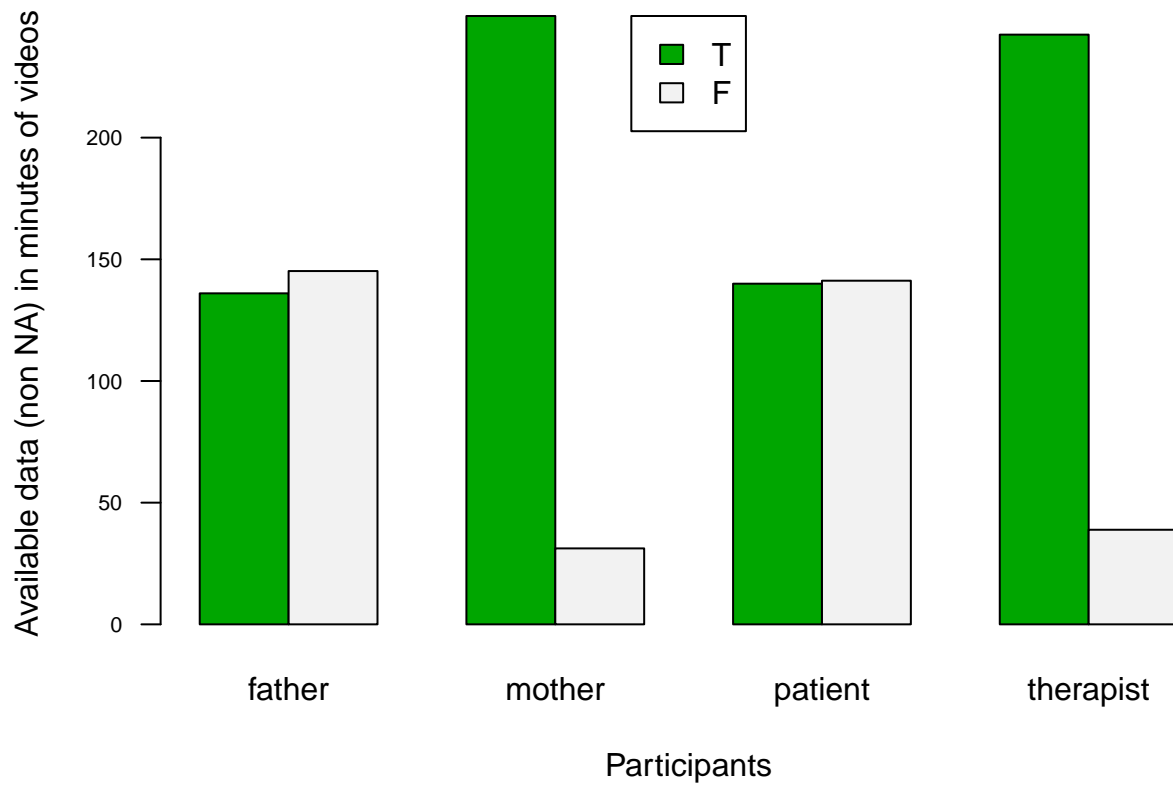


Number of Available (True) and Not Available (False) data for each participant

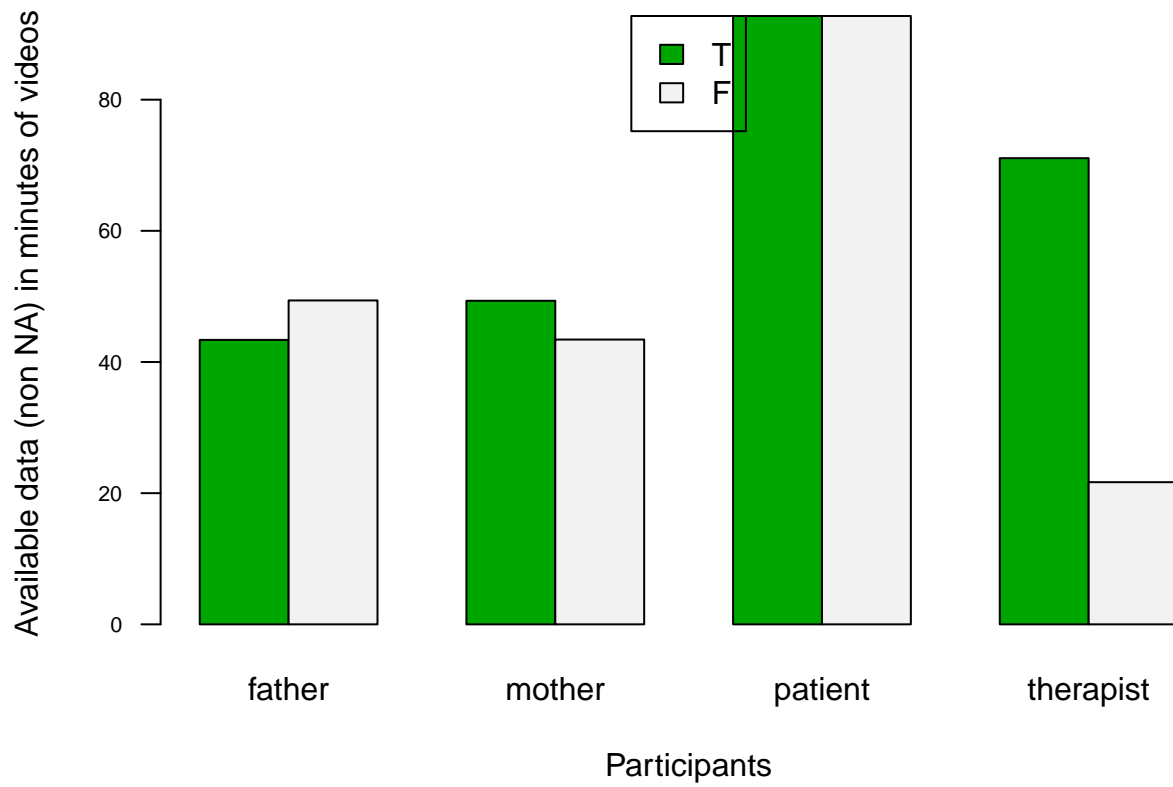
**Number of available data by participant**



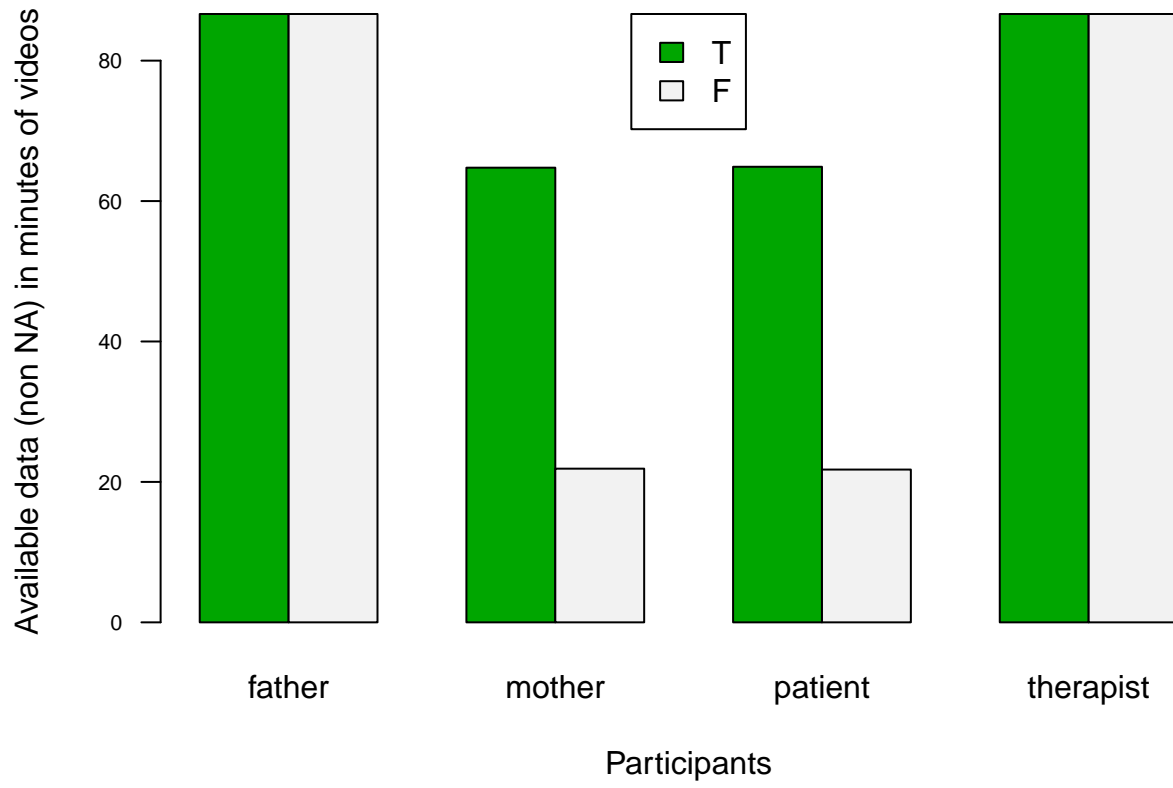
**Number of available data by participant**



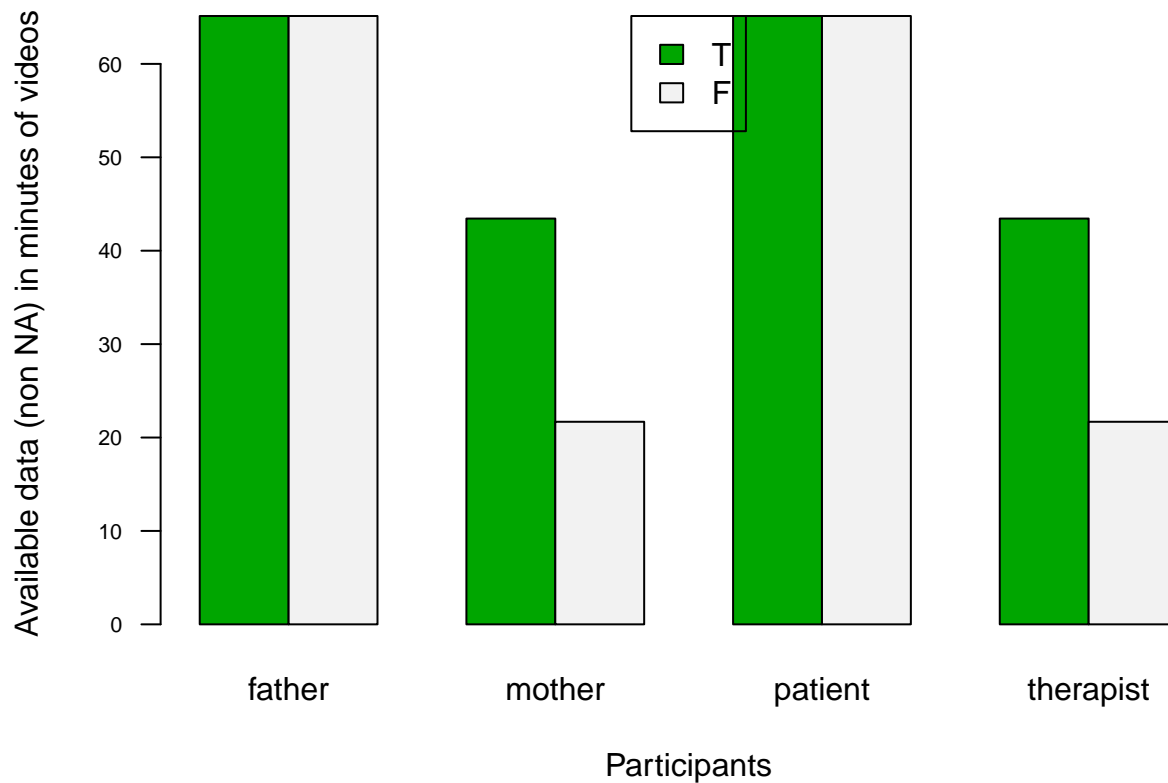
**Number of available data by participant**



**Number of available data by participant**



## Number of available data by participant



## Global Motion history

### Mean Motion history by video by participant

We can see that configurations of subjects are very different. Consequently, it makes the comparisons of the videos quite complicated. It is not really relevant to compare the synchrony of two persons if the context is different (other people around them).

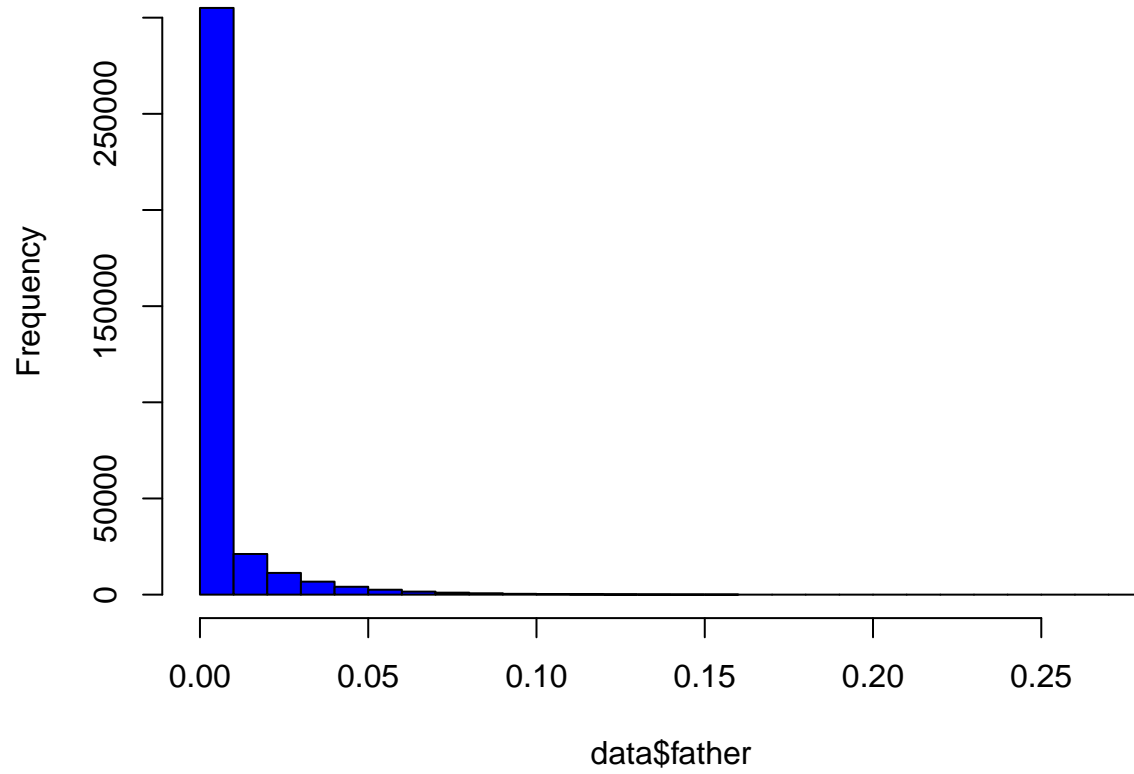
### Raw Motion history by video by participant

#### Boxplots

#### Histograms

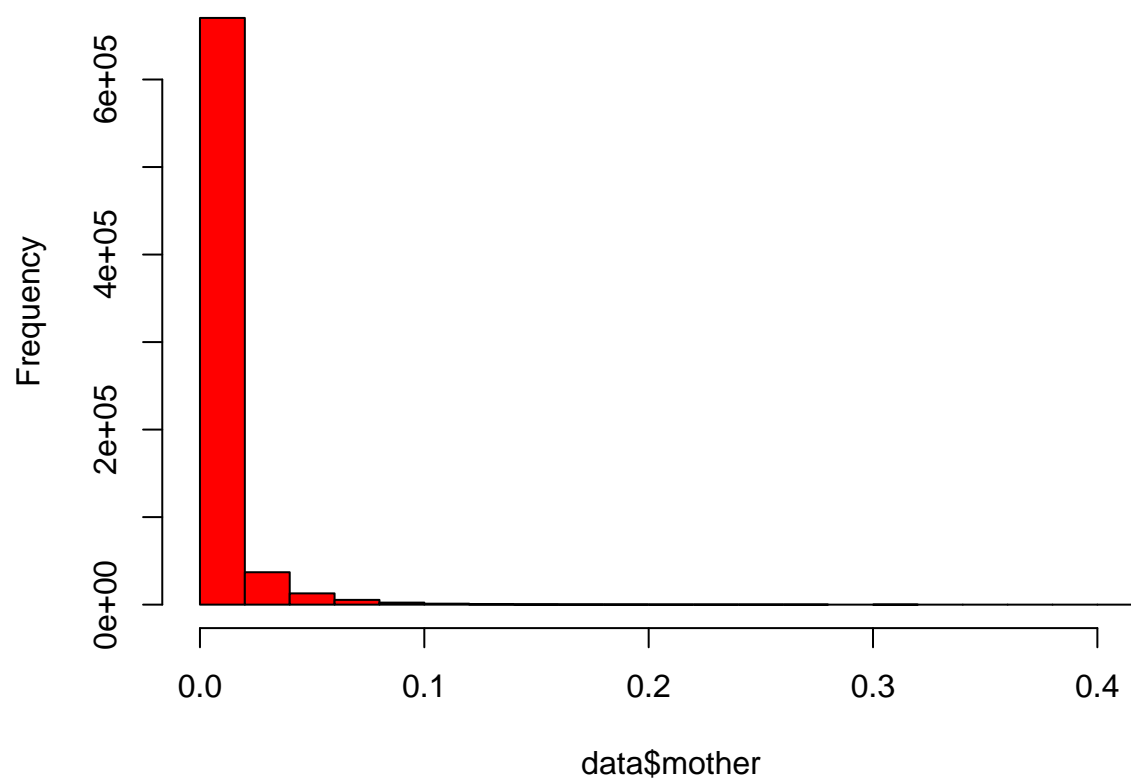
```
par(mar=c(4,4,2,2))  
hist(data$father, col=colOrderList[1])
```

Histogram of data\$father

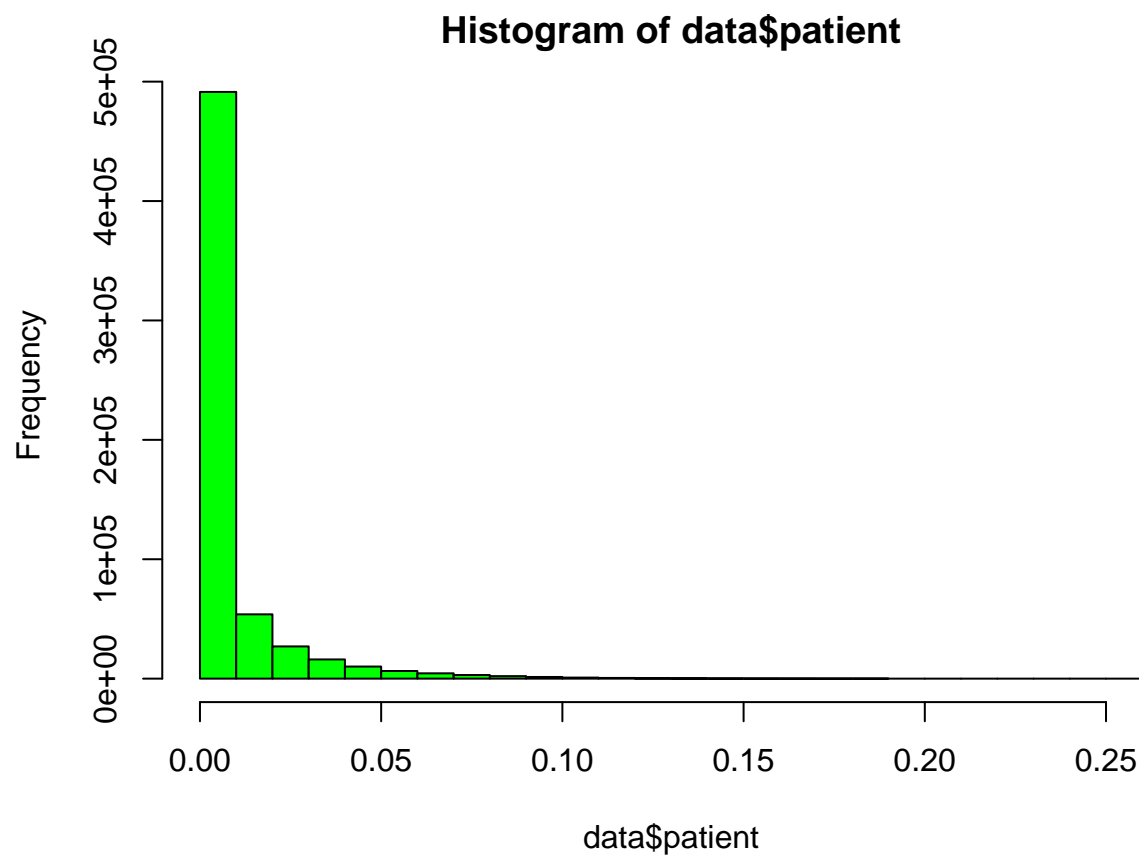


```
hist(data$mother, col=colOrderList[2])
```

**Histogram of data\$mother**

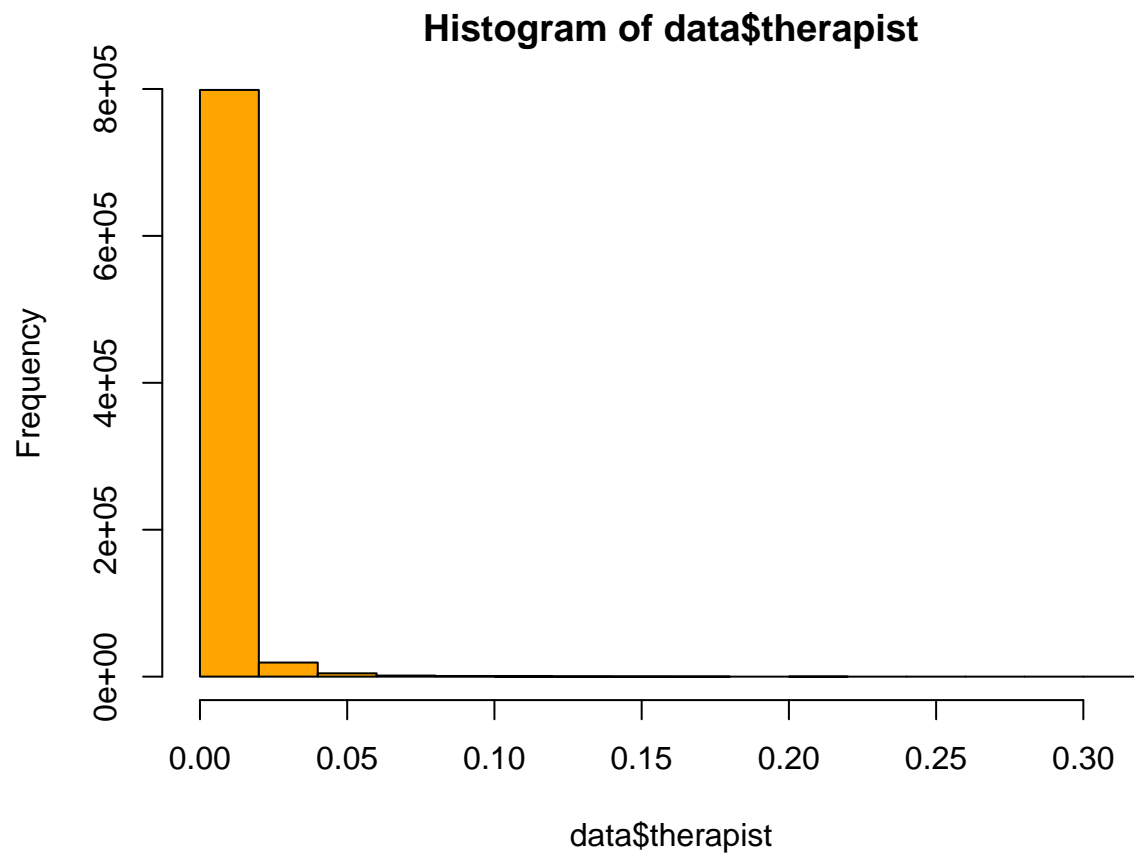


```
hist(data$patient, col=colOrderList[3])
```



```
hist(data$therapist, col=colOrderList[4])
```



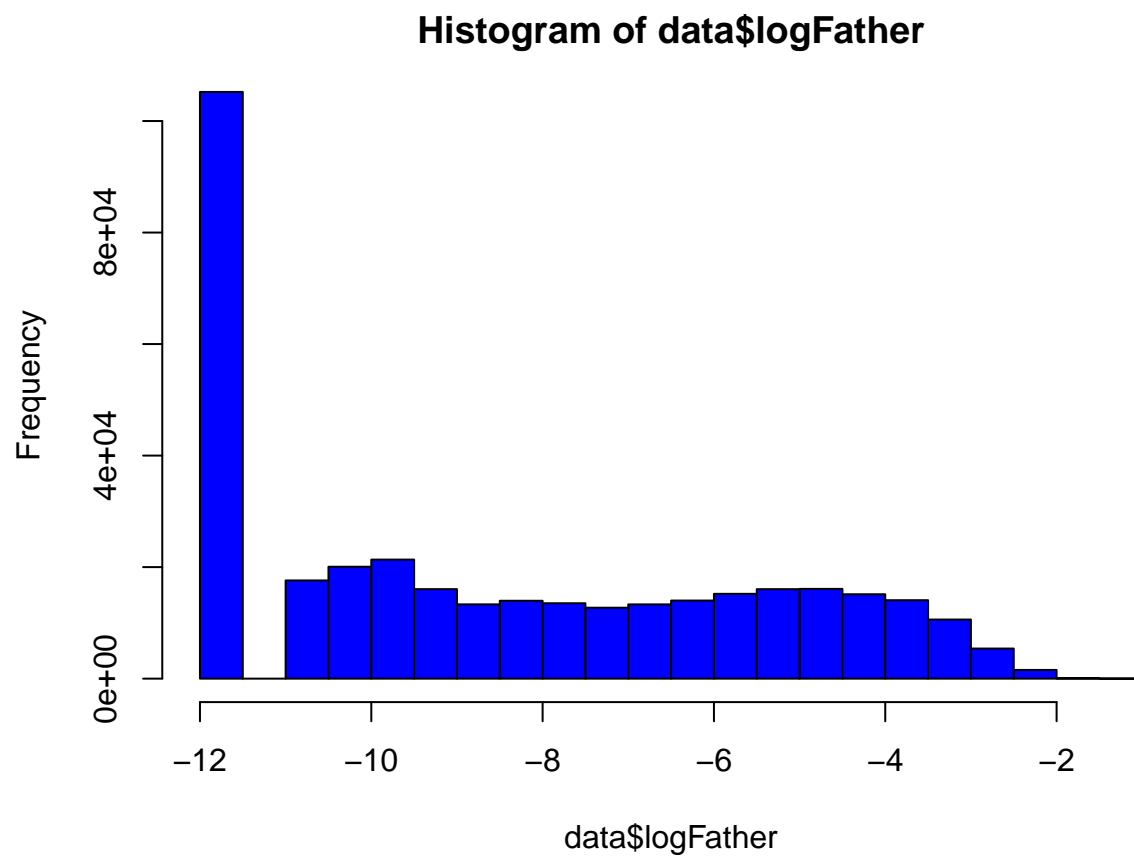


Normalized log Motion history by video by participant

Boxplots

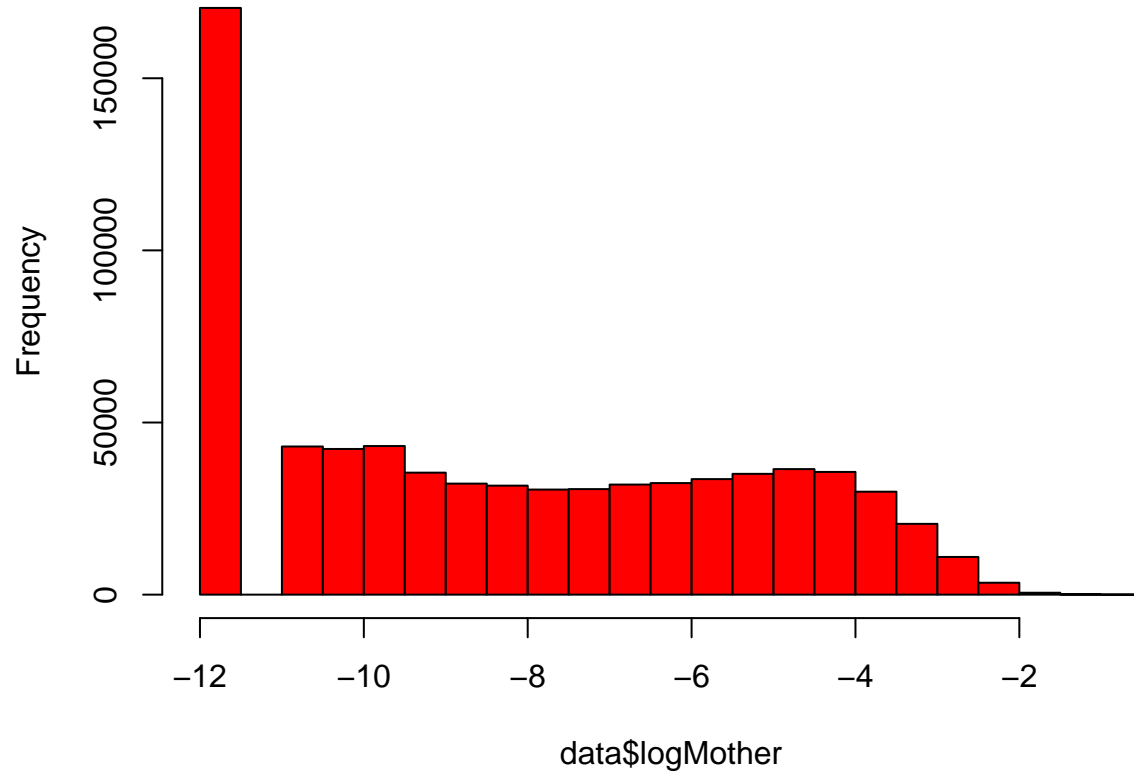
Histograms

```
par(mar=c(4,4,2,2))  
hist(data$logFather, col=colOrderList[1])
```

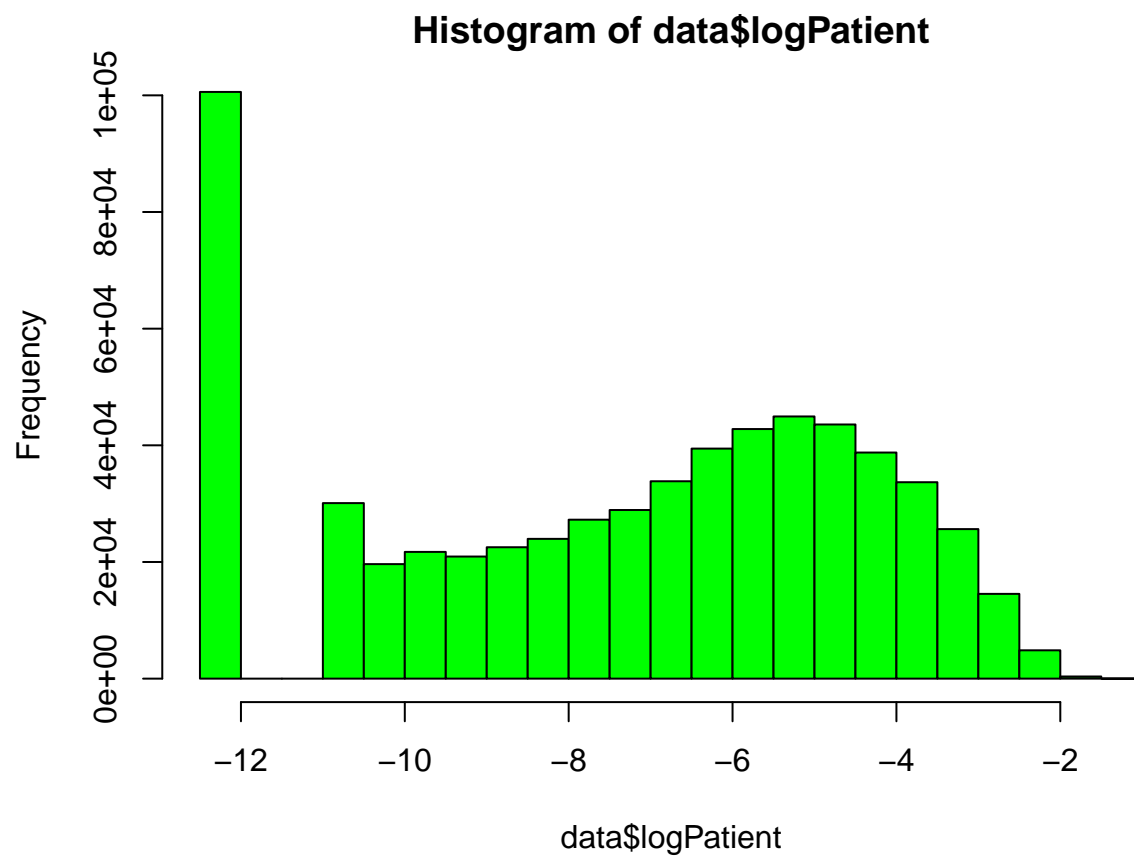


```
hist(data$logMother, col=colOrderList[2])
```

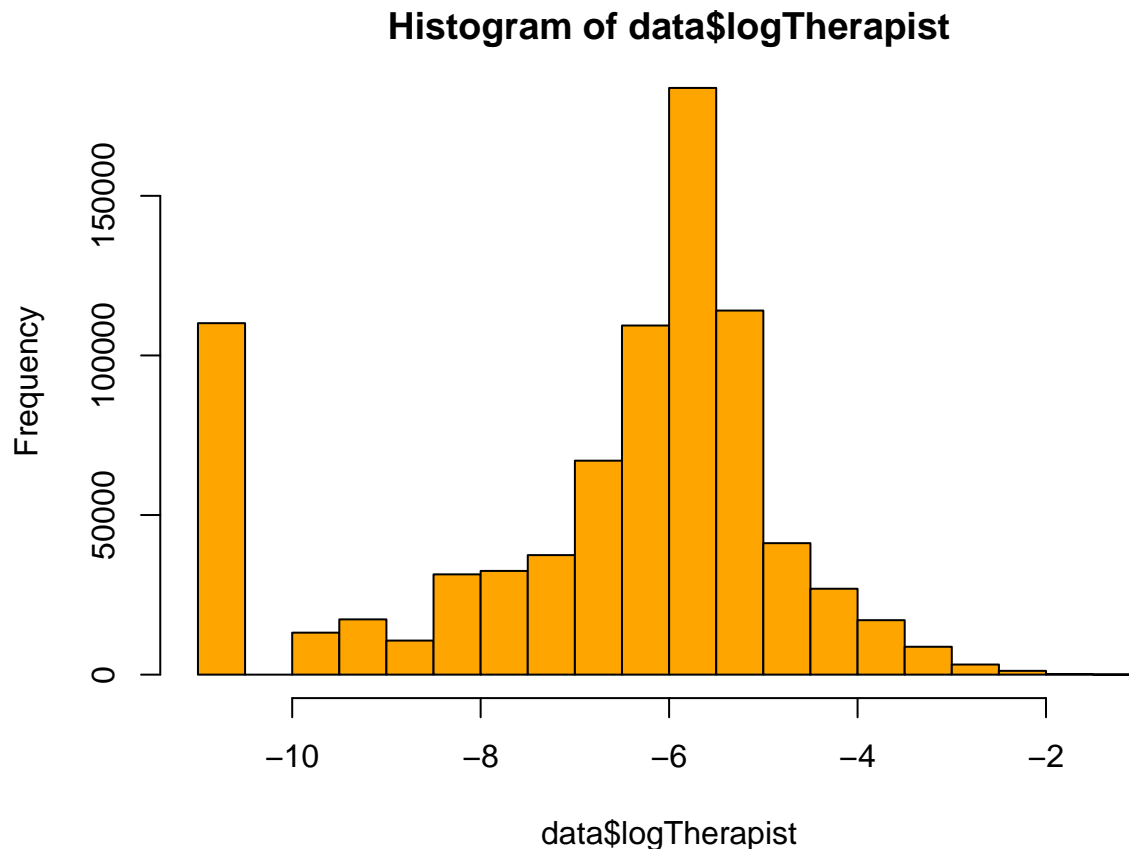
**Histogram of data\$logMother**



```
hist(data$logPatient, col=colOrderList[3])
```



```
hist(data$logTherapist, col=colOrderList[4])
```



We can see that the father and the mother motion history is very similar. However, the therapist, which is always in a small window of the video, has a very different distribution. We have less signal on it. In some videos the patient is in this window, it explains its intermediate position.

## Raw data and mean of Motion History on sliding and non overlapping intervals on 1st video F1044C1 video

It is the first video of F1044C. The father, mother and therapist are present. The patient is absent.

### Raw data

```
rawFatherF1044C1 <- data[which(data$indexList=="F1044C1"),]$father
rawMotherF1044C1 <- data[which(data$indexList=="F1044C1"),]$mother
rawTherapistF1044C1 <- data[which(data$indexList=="F1044C1"),]$therapist

logFatherF1044C1 <- data[which(data$indexList=="F1044C1"),]$logFather
logMotherF1044C1 <- data[which(data$indexList=="F1044C1"),]$logMother
logTherapistF1044C1 <- data[which(data$indexList=="F1044C1"),]$logTherapist

summary(rawFatherF1044C1)
```

##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	0.000e+00	2.079e-05	6.862e-04	7.710e-03	7.278e-03	1.568e-01

```
summary(rawMotherF1044C1)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.     Max.
## 0.000e+00 4.563e-05 1.825e-04 4.059e-03 2.806e-03 1.116e-01
```

```
summary(rawTherapistF1044C1)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.     Max.
## 0.001393 0.002832 0.003482 0.004640 0.004411 0.083240
```

```
summary(logFatherF1044C1)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.     Max.
## -11.830 -10.480  -7.274  -7.645  -4.922  -1.853
```

```
summary(logMotherF1044C1)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.     Max.
## -11.990  -9.868  -8.575  -8.028  -5.874  -2.193
```

```
summary(logTherapistF1044C1)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.     Max.
##  -6.561  -5.859  -5.654  -5.556  -5.419  -2.486
```

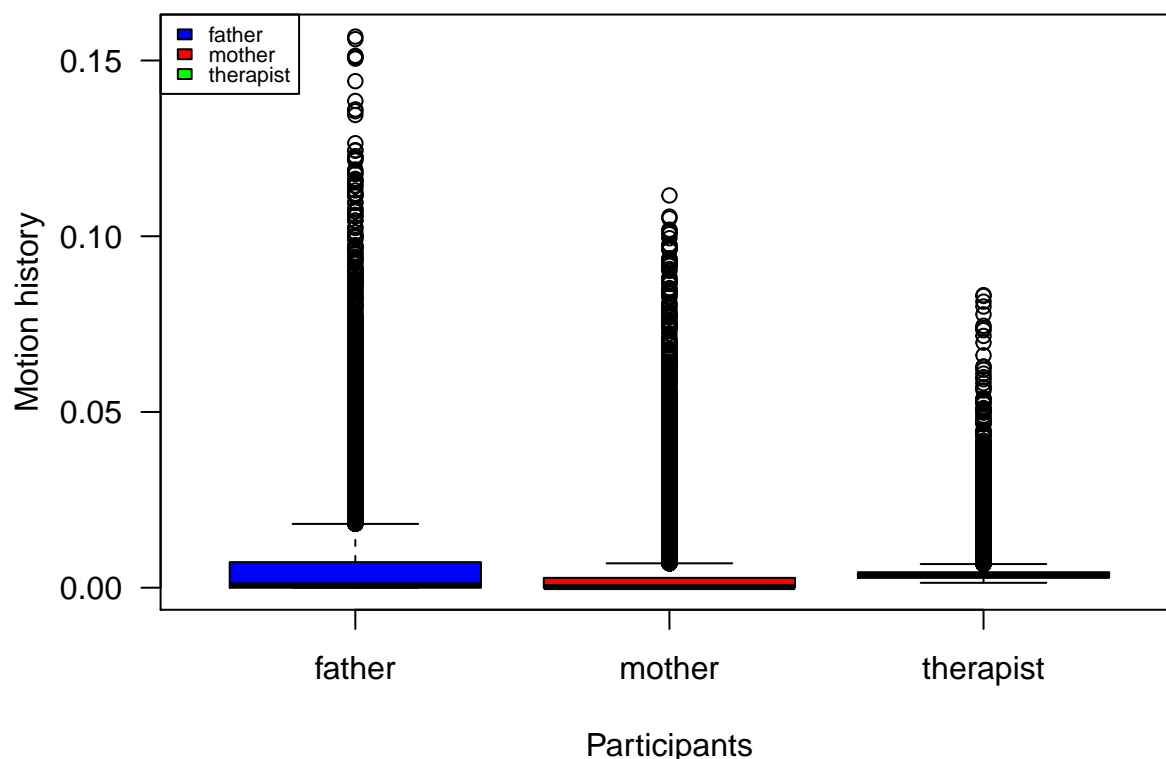
```
par(mar=c(4,4,3,2))
```

```
boxplot(rawFatherF1044C1, rawMotherF1044C1, rawTherapistF1044C1,
        col=colOrderList[c(1,2,4)],
        names=ParticipantsList[c(1,2,4)],
        main= "Box plots of motion history raw data on F1044C1 video", las=1, ylab = "Motion history", xlab = "Participant")
```

```
par(mar=c(1,0.5,0.5,1))
```

```
legend("topleft", ParticipantsList[c(1,2,4)], fill=colOrderList, cex=0.7)
```

## Box plots of motion history raw data on F1044C1 video



## Sliding interval

```
## REMINDER:
# SlidingInterval <- function(subject, indexOfvideos=1:NumberOfvideos, interval, data) with :
# subject : subject studied (patient, mother, father or therapist)
# indexOfvideos : list of videos studied (element eg. 3 or list eg 1:3 or c(1,2,4))
# interval : number of frames in the studied interval
# data : data frame where there is data
```

```
slidedFatherF1044C1 <- SlidingInterval("father", 1 , 5, data)
slidedMotherF1044C1 <- SlidingInterval("mother", 1 , 5, data)
slidedPatientF1044C1 <- SlidingInterval("patient", 1 , 5, data)
slidedTherapistF1044C1 <- SlidingInterval("therapist", 1 , 5, data)

slidedLogFatherF1044C1 <- SlidingInterval("logFather", 1 , 5, data)
slidedLogMotherF1044C1 <- SlidingInterval("logMother", 1 , 5, data)
slidedLogPatientF1044C1 <- SlidingInterval("logPatient", 1 , 5, data)
slidedLogTherapistF1044C1 <- SlidingInterval("logTherapist", 1 , 5, data)
```

```
summary(slidedFatherF1044C1)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.    Max.
## 0.000e+00 4.271e-05 2.234e-04 5.362e-03 2.747e-03 2.493e-01
```

```
summary(slidedMotherF1044C1)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.    Max.
```

```
## 0.0000000 0.0007124 0.0058320 0.0131100 0.0178900 0.2008000
```

```
summary(slidedPatientF1044C1)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's  
##      NA      NA      NA     NaN    NA      NA    43132
```

```
summary(slidedTherapistF1044C1)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.  
## 0.000e+00 1.698e-05 1.188e-04 2.225e-03 8.318e-04 1.594e-01
```

```
summary(slidedLogFatherF1044C1)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.  
## -11.830 -10.120  -8.711  -8.050  -6.014  -1.393
```

```
summary(slidedLogMotherF1044C1)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.  
## -11.990  -7.701  -5.342  -6.075  -4.087  -1.648
```

```
summary(slidedLogPatientF1044C1)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's  
##      NA      NA      NA     NaN    NA      NA    43132
```

```
summary(slidedLogTherapistF1044C1)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.  
## -10.760 -10.440  -9.480  -8.750  -7.570  -1.837
```

## Non overlapping interval

```
fatherFiveF1044C1<- MeanMotionByTime("father", indexOfvideos=1, interval=5, data)
```

```
motherFiveF1044C1 <- MeanMotionByTime("mother", indexOfvideos=1, interval=5, data)
```

```
therapistFiveF1044C1 <- MeanMotionByTime("therapist", indexOfvideos=1, interval=5, data)
```

```
fatherLogFiveF1044C1<- MeanMotionByTime("logFather", indexOfvideos=1, interval=5, data)
```

```
motherLogFiveF1044C1 <- MeanMotionByTime("logMother", indexOfvideos=1, interval=5, data)
```

```
therapistLogFiveF1044C1 <- MeanMotionByTime("logTherapist", indexOfvideos=1, interval=5, data)
```

```
summary(fatherFiveF1044C1)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.  
## 0.000e+00 4.271e-05 2.218e-04 5.361e-03 2.722e-03 2.248e-01
```

```
summary(motherFiveF1044C1)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.  
## 0.0000000 0.0006942 0.0058640 0.0131200 0.0178900 0.1875000
```

```
summary(therapistFiveF1044C1)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
```



```
## 0.000e+00 8.490e-06 1.273e-04 2.224e-03 8.403e-04 1.475e-01
```

```
summary(fatherLogFiveF1044C1)
```

```
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.
## -11.830 -10.120  -8.715  -8.050  -6.022  -1.510
```

```
summary(motherLogFiveF1044C1)
```

```
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.
## -11.990  -7.698  -5.330  -6.075  -4.087  -1.674
```

```
summary(therapistLogFiveF1044C1)
```

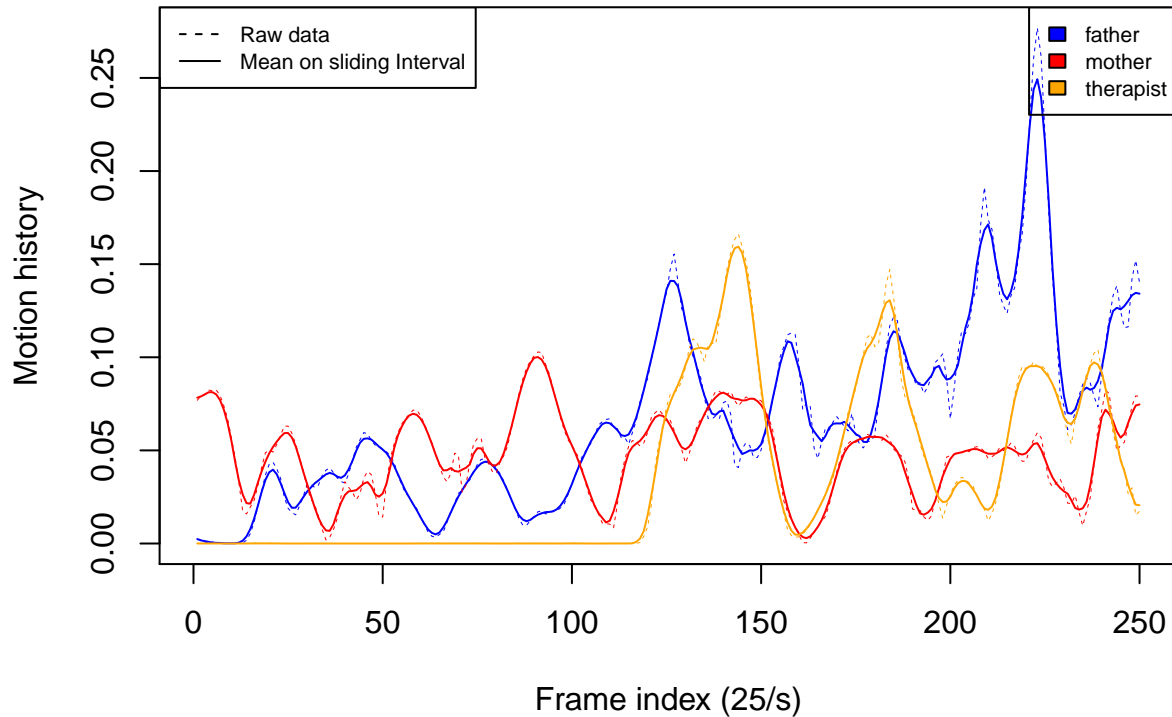
```
##      Min. 1st Qu.  Median      Mean 3rd Qu.      Max.
## -10.760 -10.540  -9.473  -8.750  -7.543  -1.920
```

## Focus on the motion history of the first 10 seconds of the first video

### Sliding interval function on a raw data, 5 frames interval

```
par(mar=c(4,4,4,2))
plot(1:250, data$father[3:252], main="Mean motion history, (Sliding 5 frames interval raw data)
      on 1st video, first 10 seconds ", xlab="Frame index (25/s)",
      ylab="Motion history",
      col="blue", type="l", lty=2, lwd=0.5)
lines(slidedFatherF1044C1[1:250], col="blue", lty=1)
lines(data$mother[3:252], col="red", lty=2, lwd=0.5)
lines(slidedMotherF1044C1[1:250], col="red", lty=1)
lines(data$therapist[3:252], col="orange", lty=2, lwd=0.5)
lines(slidedTherapistF1044C1[1:250], col="orange", lty=1)
legend("topleft", c("Raw data", "Mean on sliding Interval") , lty=c(2, 1), cex=0.7)
legend("topright", ParticipantsList[c(1,2,4)], fill=colOrderList[c(1,2,4)], cex=0.7)
```

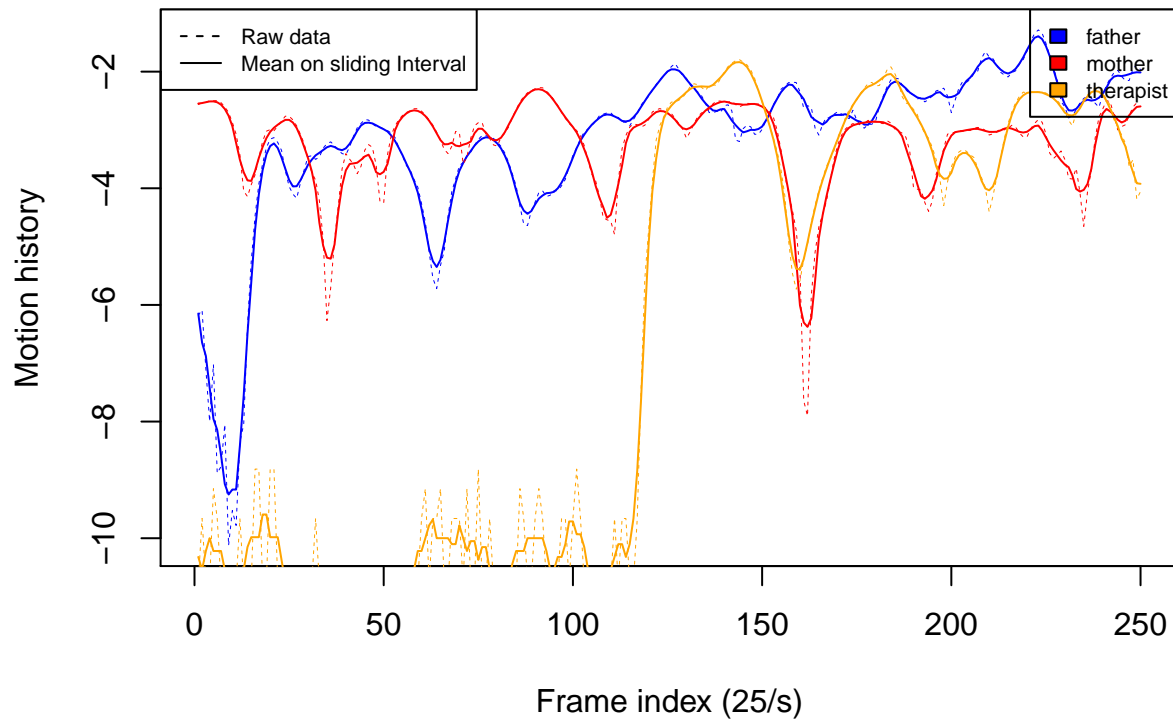
## Mean motion history, (Sliding 5 frames interval raw data) on 1st video, first 10 seconds



## Sliding interval function on log data, 5 frames interval

```
par(mar=c(4,4,4,2))
plot(1:250, data$logFather[3:252], main="Mean motion history, (Sliding 5 frames interval log data)
      on 1st video, first 10 seconds", xlab="Frame index (25/s)",
      ylab="Motion history",
      col="blue", type="l", lty=2, lwd=0.5)
lines(slidedLogFatherF1044C1[1:250], col="blue", lty=1)
lines(data$logMother[3:252], col="red", lty=2, lwd=0.5)
lines(slidedLogMotherF1044C1[1:250], col="red", lty=1)
lines(data$logTherapist[3:252], col="orange", lty=2, lwd=0.5)
lines(slidedLogTherapistF1044C1[1:250], col="orange", lty=1)
legend("topleft", c("Raw data", "Mean on sliding Interval"), lty=c(2, 1), cex=0.7)
legend("topright", ParticipantsList[c(1,2,4)], fill=colOrderList[c(1,2,4)], cex=0.7)
```

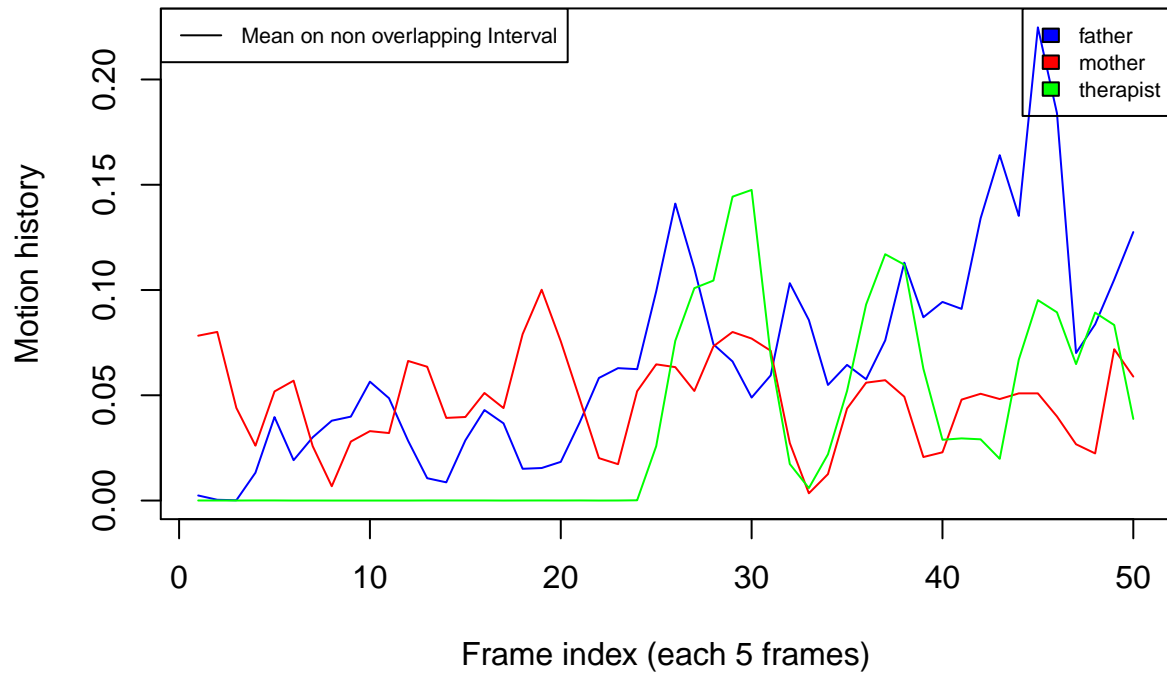
## Mean motion history, (Sliding 5 frames interval log data) on 1st video, first 10 seconds



Non overlapping interval function on a 5 frames interval

```
plot (1:50, fatherFiveF1044C1[1:50], type="l", col="blue",
main="Mean Motion history (non overlapping 5 frames
intervals) for father on F1044C video, first 10 seconds",
ylab="Motion history", xlab="Frame index (each 5 frames)" )
lines(motherFiveF1044C1[1:50], col="red", lty=1)
lines(therapistFiveF1044C1[1:50], col="green", lty=1)
legend("topleft", "Mean on non overlapping Interval" , lty=1, cex=0.7)
legend("topright", ParticipantsList[c(1,2,4)], fill=colOrderList, cex=0.7)
```

## Mean Motion history (non overlapping 5 frames intervals) for father on F1044C video, first 10 seconds

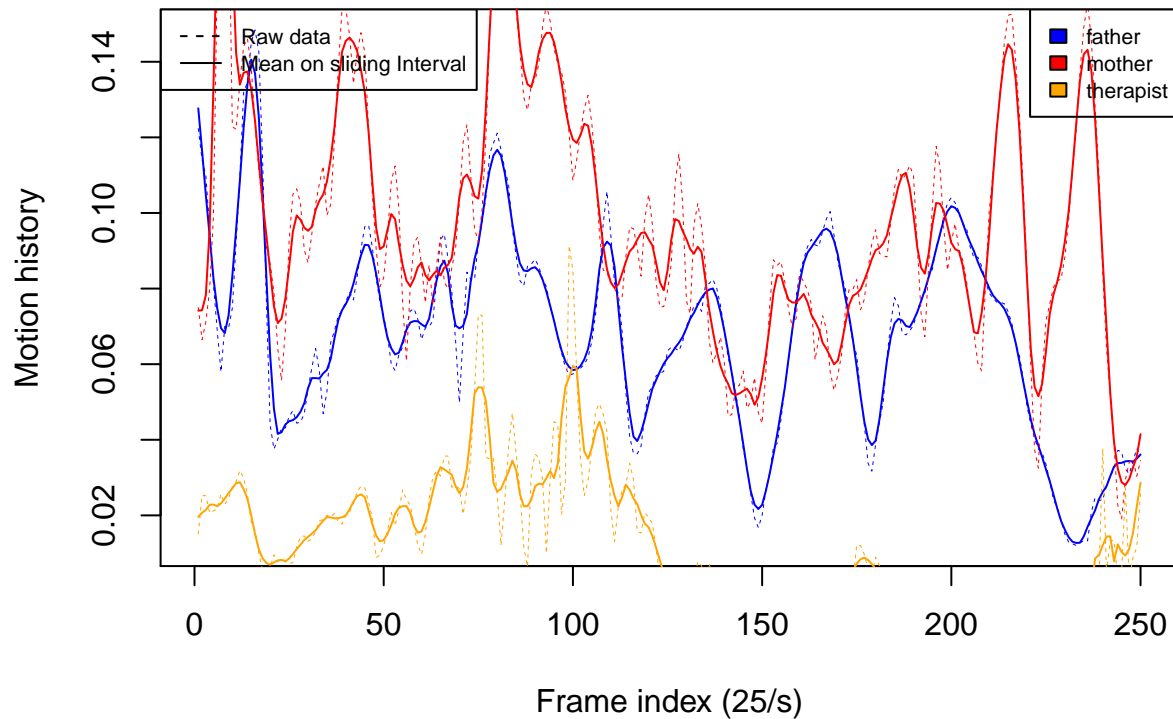


Motion history of the father during 10-20 seconds of the first video(C)

Non overlapping interval function on a 5 frames interval

```
par(mar=c(4,4,4,2))
plot(1:250, data$father[253:502], main="Mean motion history (Sliding 5 frames
interval) for father on F1044C video, 10-20 seconds", xlab="Frame index (25/s)",
ylab="Motion history", col="blue", type="l", lty=2, lwd=0.5)
lines(slidedFatherF1044C1[251:500], col="blue", lty=1)
lines(data$mother[253:502], col="red", lty=2, lwd=0.5)
lines(slidedMotherF1044C1[251:500], col="red", lty=1)
lines(data$therapist[253:502], col="orange", lty=2, lwd=0.5)
lines(slidedTherapistF1044C1[251:500], col="orange", lty=1)
legend("topleft", c("Raw data", "Mean on sliding Interval"), lty=c(2, 1), cex=0.7)
legend("topright", ParticipantsList[c(1,2,4)], fill=colOrderList[c(1,2,4)], cex=0.7)
```

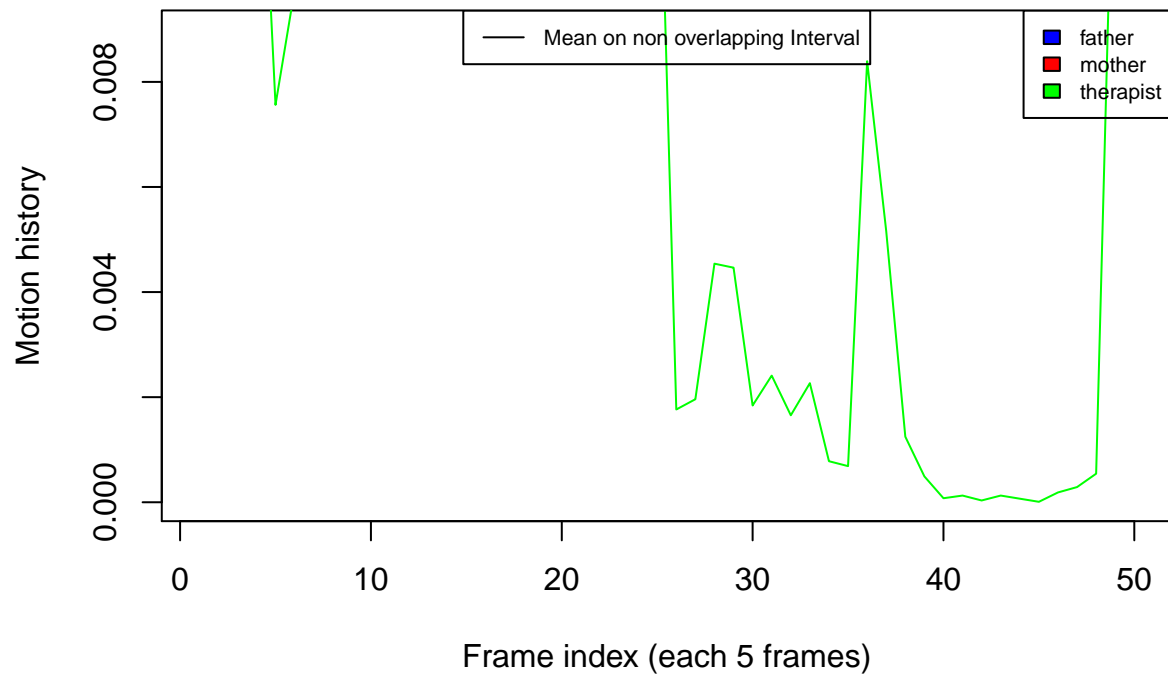
### Mean motion history (Sliding 5 frames interval) for father on F1044C video, 10–20 seconds



Non overlapping interval function on a 5 frames interval

```
plot (1:50, fatherFiveF1044C1[51:100], type="l", col="blue",
main="Mean motion history (non overlapping 5 frames intervals) for
father on F1044C video, between 10–20 seconds",
ylab="Motion history", xlab="Frame index (each 5 frames)", ylim=c(0, 0.009))
lines(motherFiveF1044C1[51:100], col="red", lty=1)
lines(therapistFiveF1044C1[51:100], col="green", lty=1)
legend("top", "Mean on non overlapping Interval" , lty=1, cex=0.7)
legend("topright", ParticipantsList[c(1,2,4)], fill=colOrderList, cex=0.7)
```

## Mean motion history (non overlapping 5 frames intervals) for father on F1044C video, between 10–20 seconds



## Mean motion history by minute plots

```
for (i in 1:NumberOfvideos){
  fatherMinute<- MeanMotionByTime("father", indexOfvideos=i, interval=1500, data)

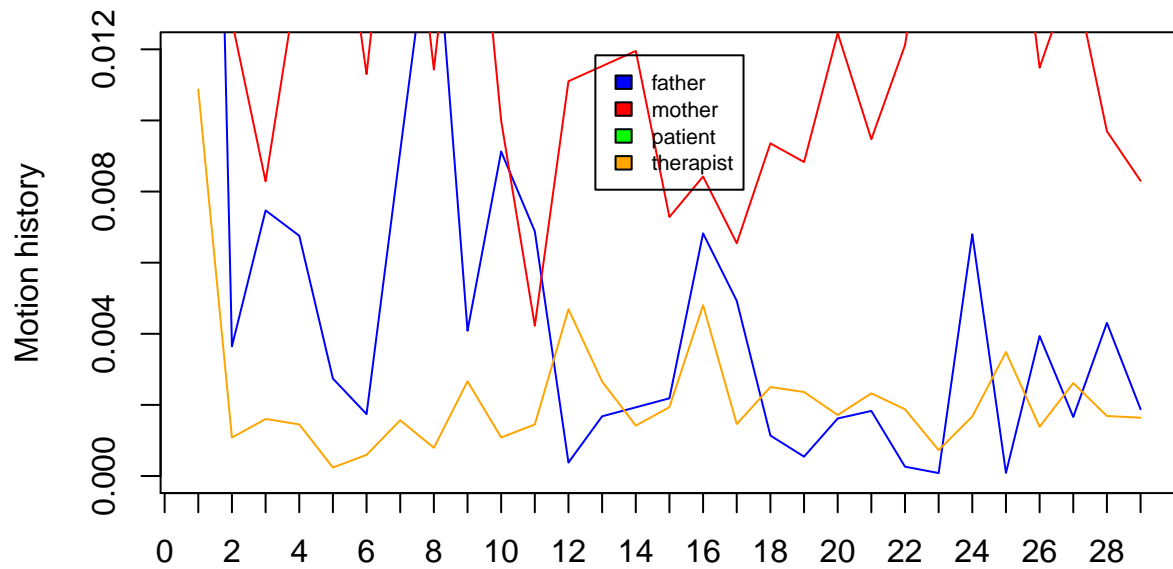
  MotherMinute<- MeanMotionByTime("mother", indexOfvideos=i, interval=1500, data)

  TherapistMinute<- MeanMotionByTime("therapist", indexOfvideos=i, interval=1500, data)

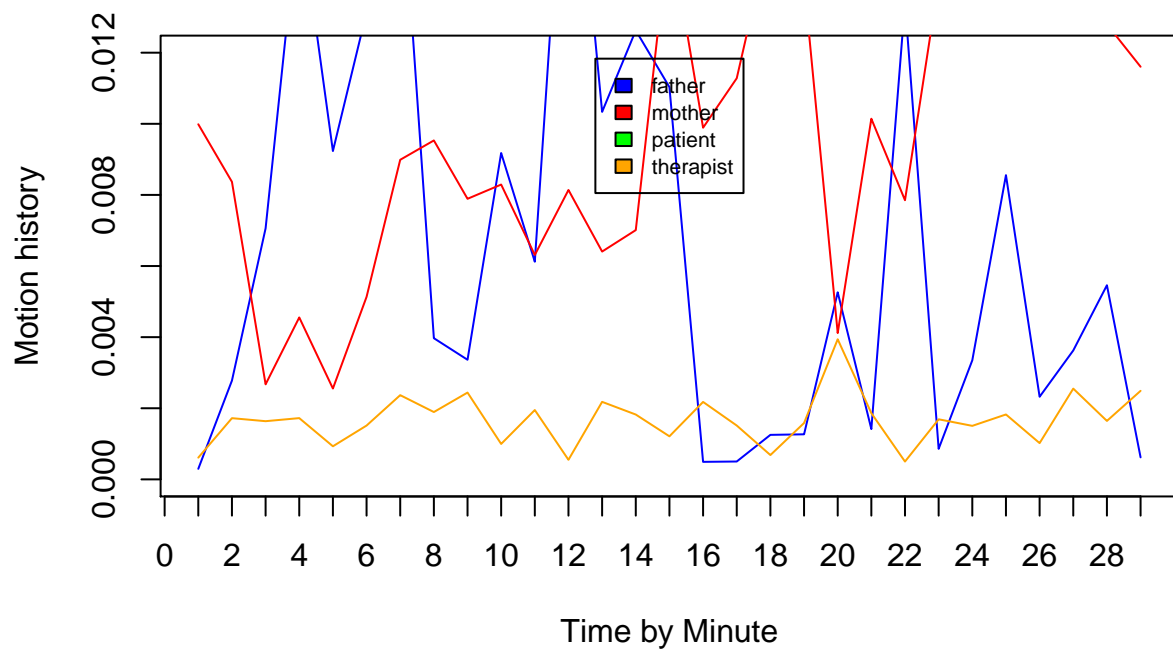
  PatientMinute<- MeanMotionByTime("patient", indexOfvideos=i, interval=1500, data)

  par(mar=c(4,4,4,2))
  plot (1:length(fatherMinute), fatherMinute, type="l", col="blue",
        main=paste("Mean motion history (non overlapping minute intervals)
on F1044", labelvideolist[i], " video" , sep=""),
        ylab="Motion history", xlab="Time by Minute", ylim=c(0, 12E-03),
        xaxp=c(0, length(fatherMinute), length(fatherMinute)))
  lines(MotherMinute, col="red")
  lines(TherapistMinute, col="orange")
  lines(PatientMinute, col="green")
  legend("top", inset=.05, ParticipantsList,
        fill=colOrderList, cex=0.7)}
```

**Mean motion history (non overlapping minute intervals)  
on F1044A1 video**



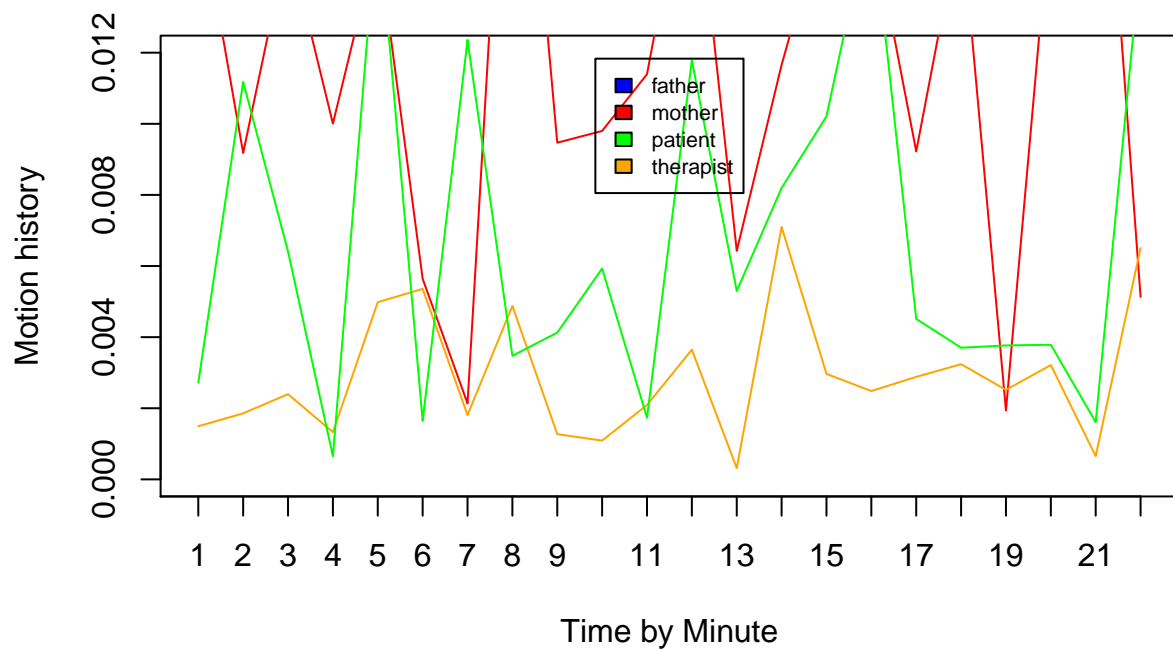
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044A2 video**



**Mean motion history (non overlapping minute intervals)  
on F1044B2 video**

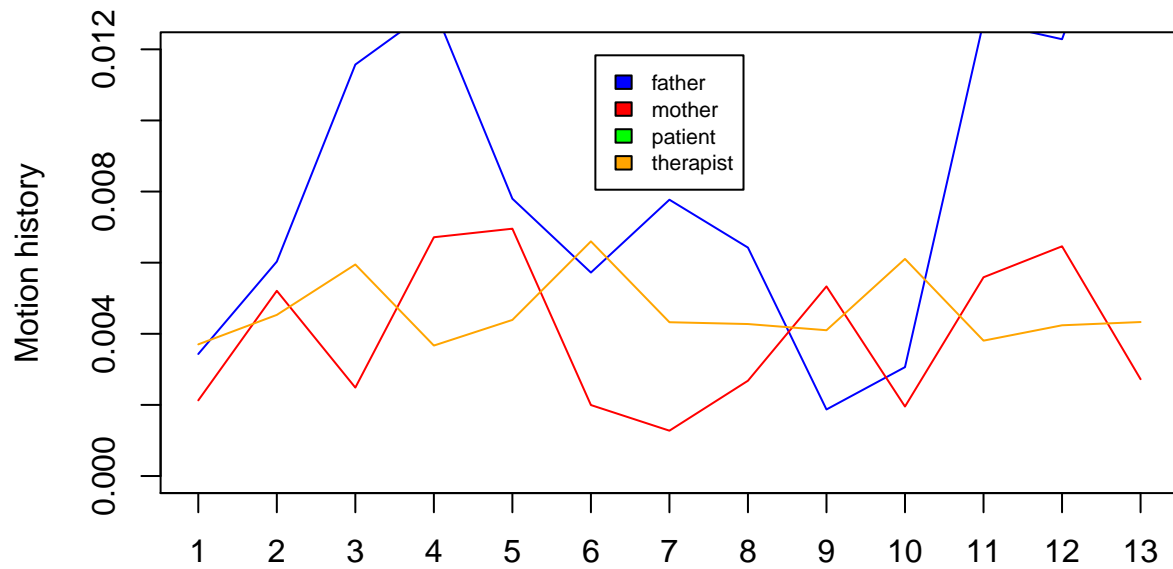


**Mean motion history (non overlapping minute intervals)  
on F1044C1 video**

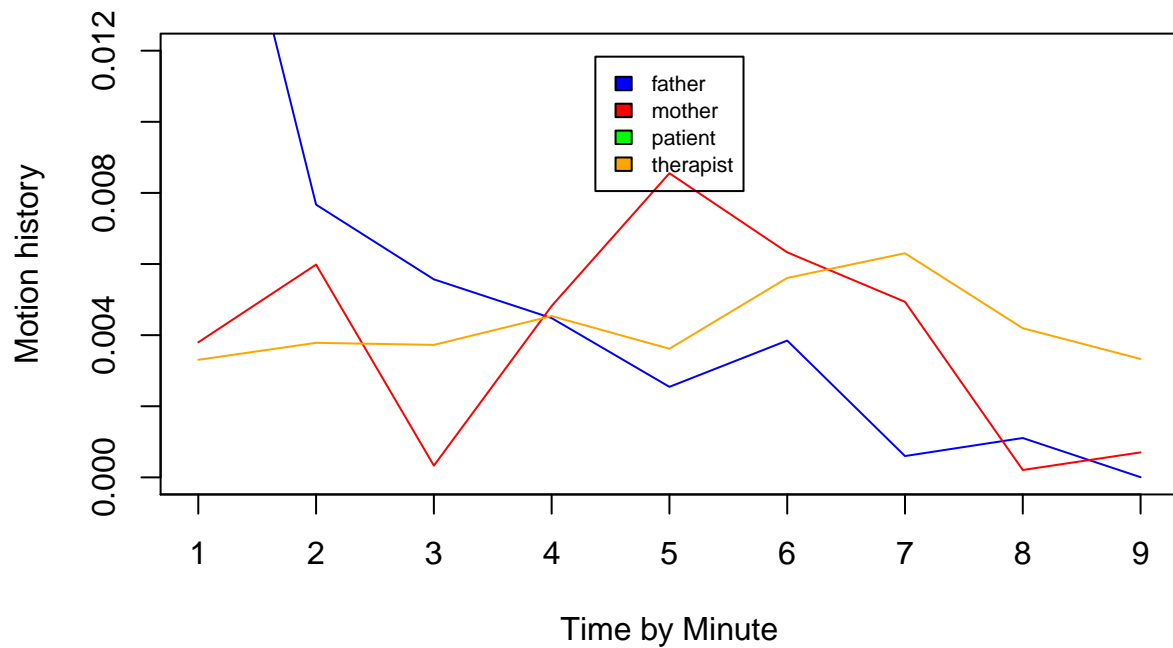




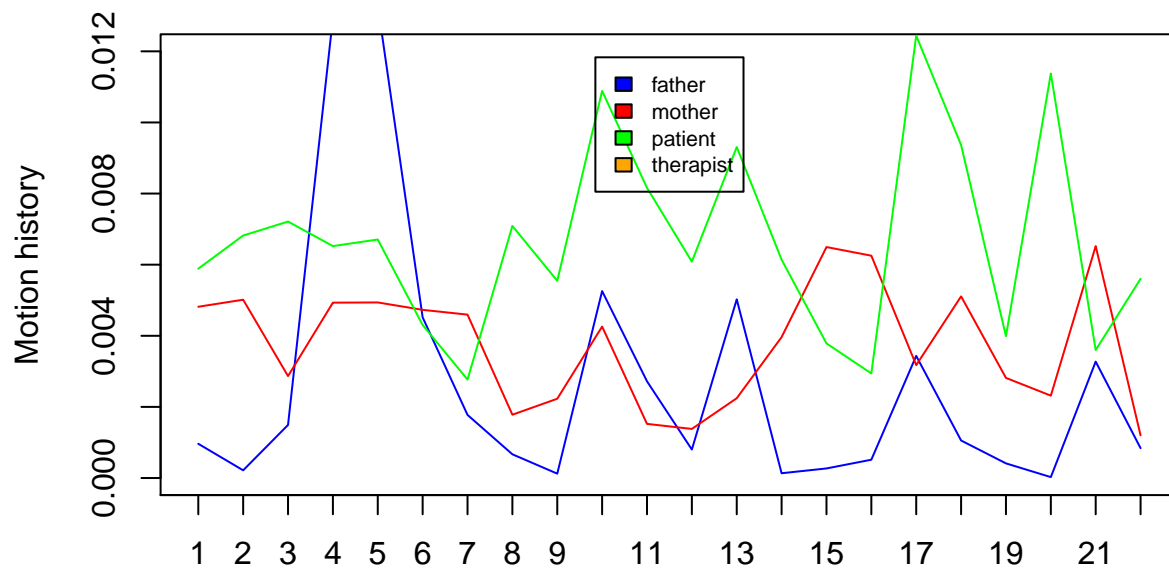
**Mean motion history (non overlapping minute intervals)  
on F1044C1 video**



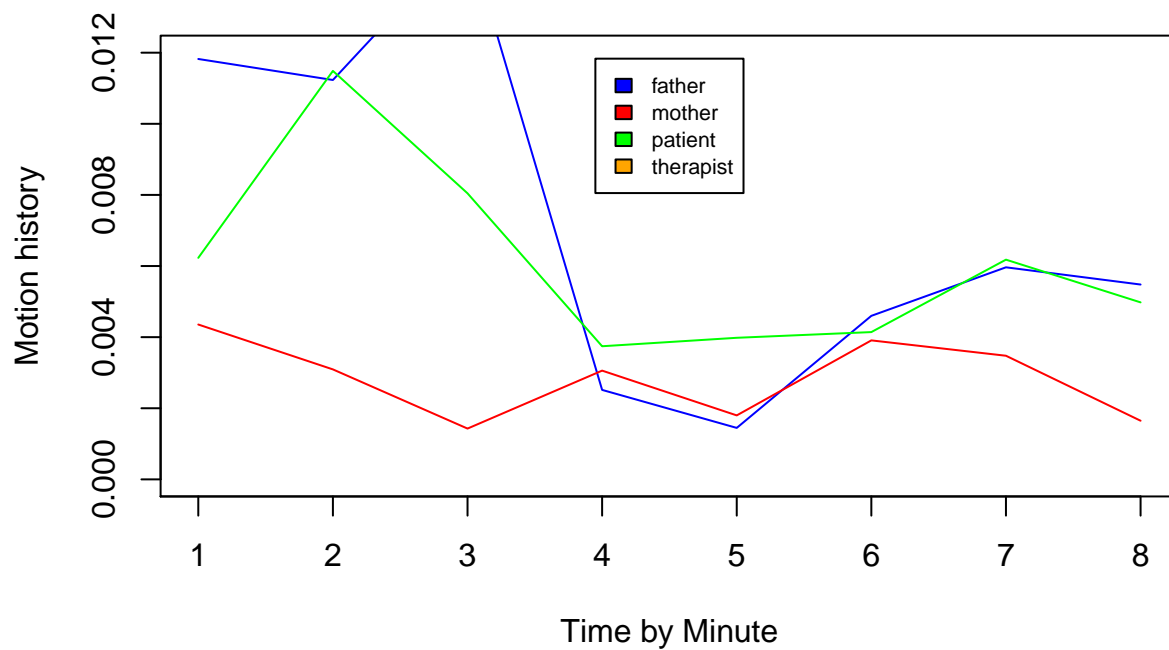
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044C2 video**



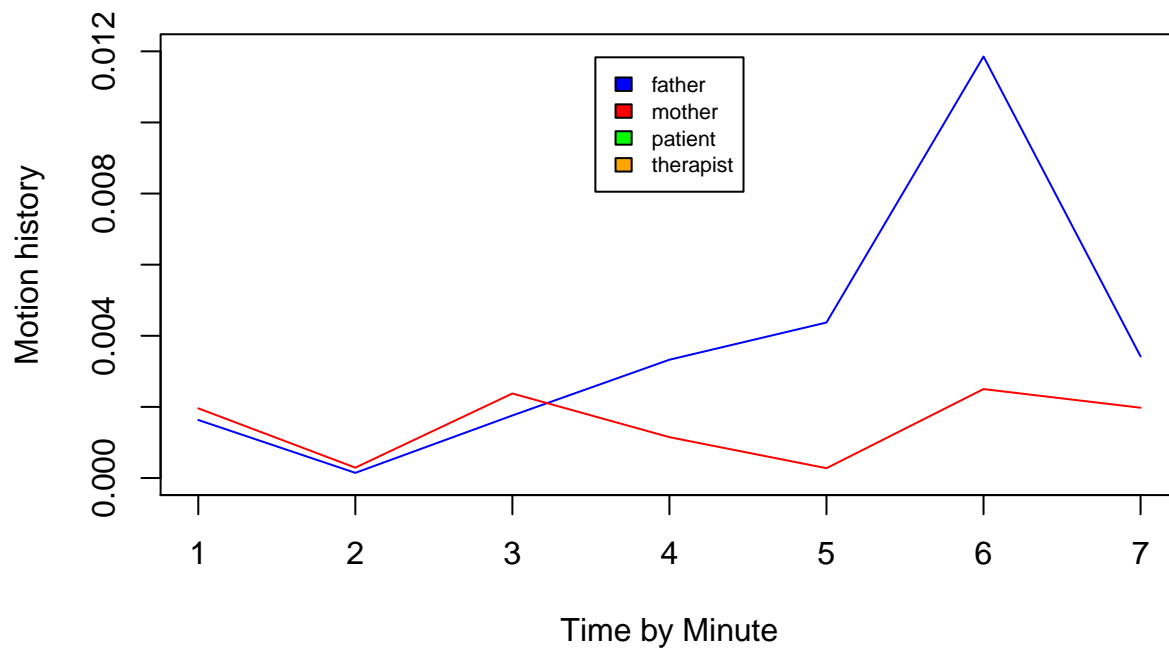
**Mean motion history (non overlapping minute intervals)  
on F1044D1 video**



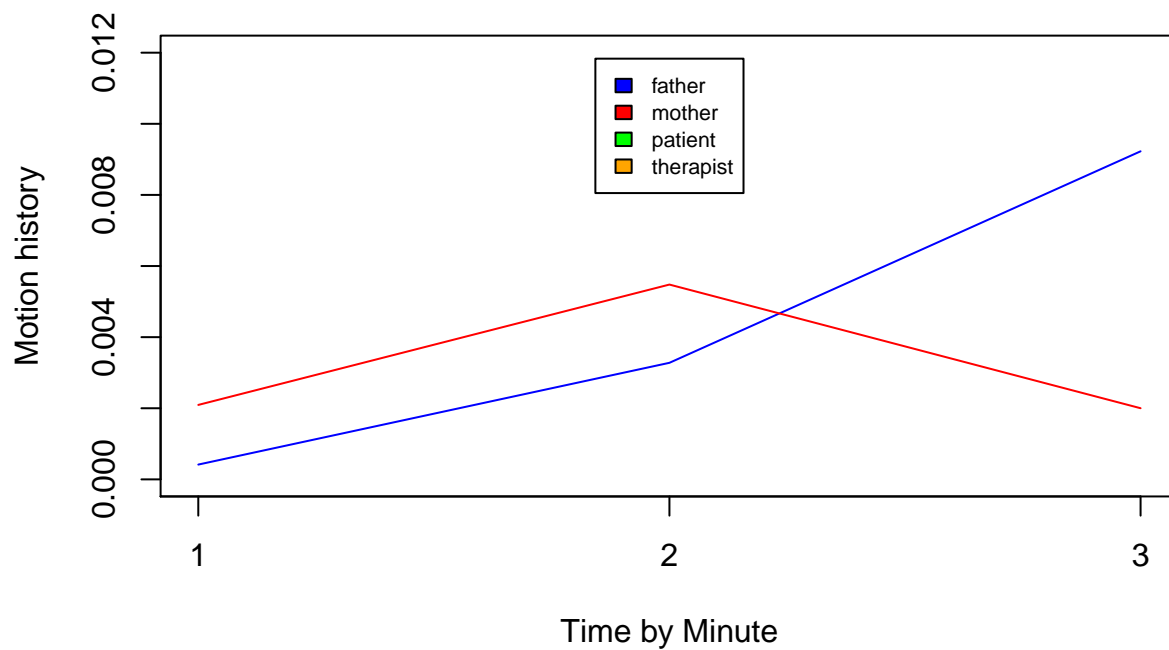
**Mean motion history (non overlapping minute intervals)  
on F1044D2 video**



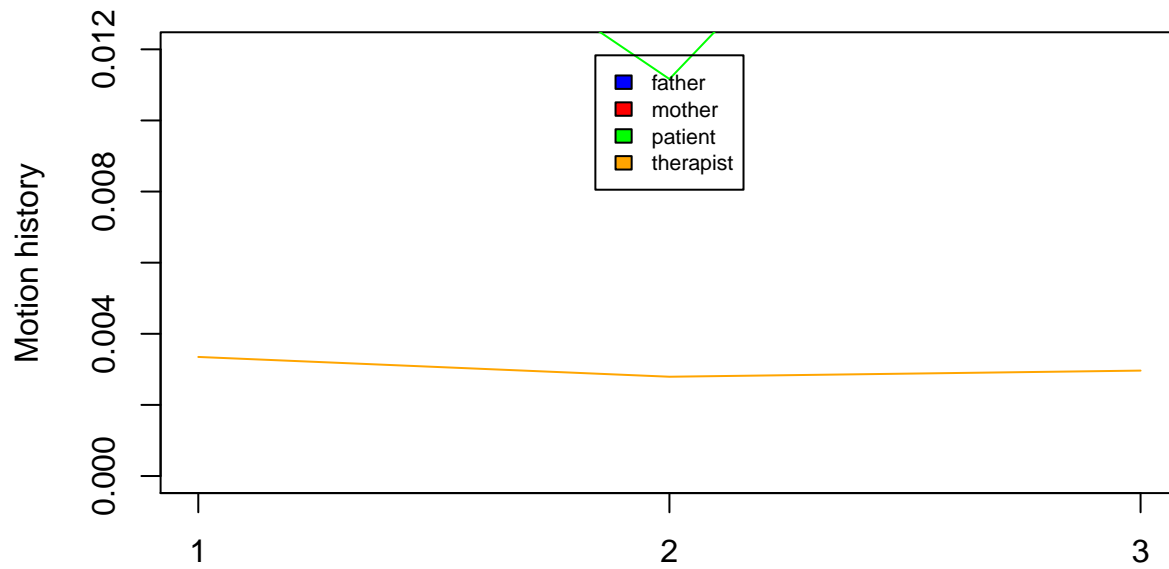
**Mean motion history (non overlapping minute intervals)  
on F1044E1 video**



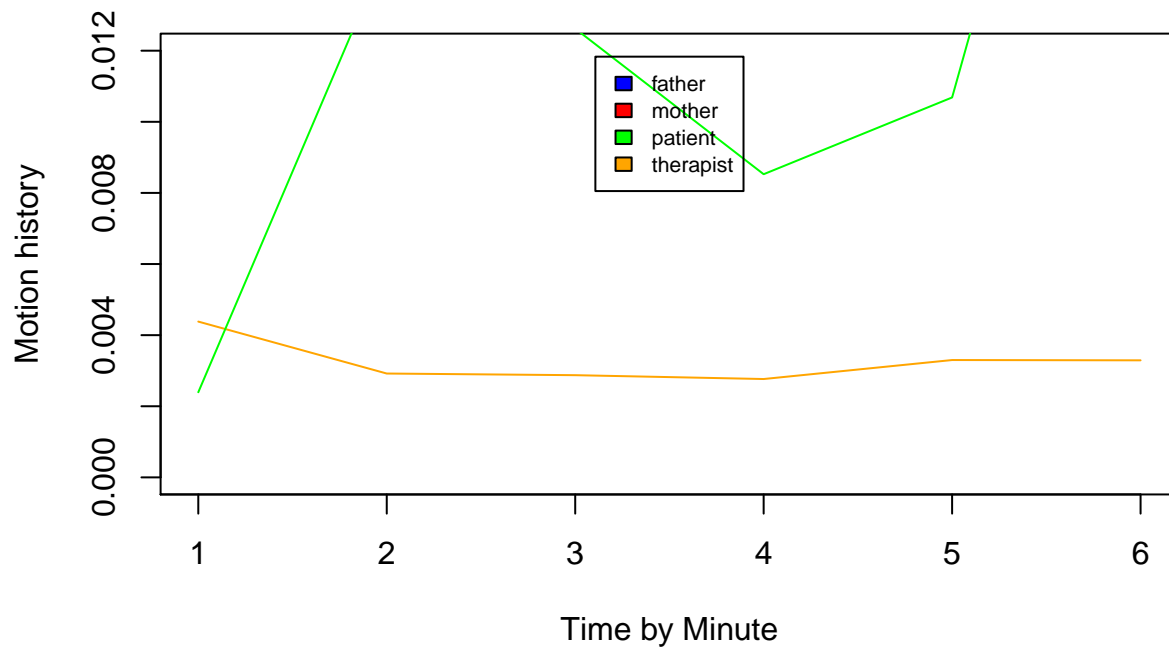
**Mean motion history (non overlapping minute intervals)  
on F1044E2 video**



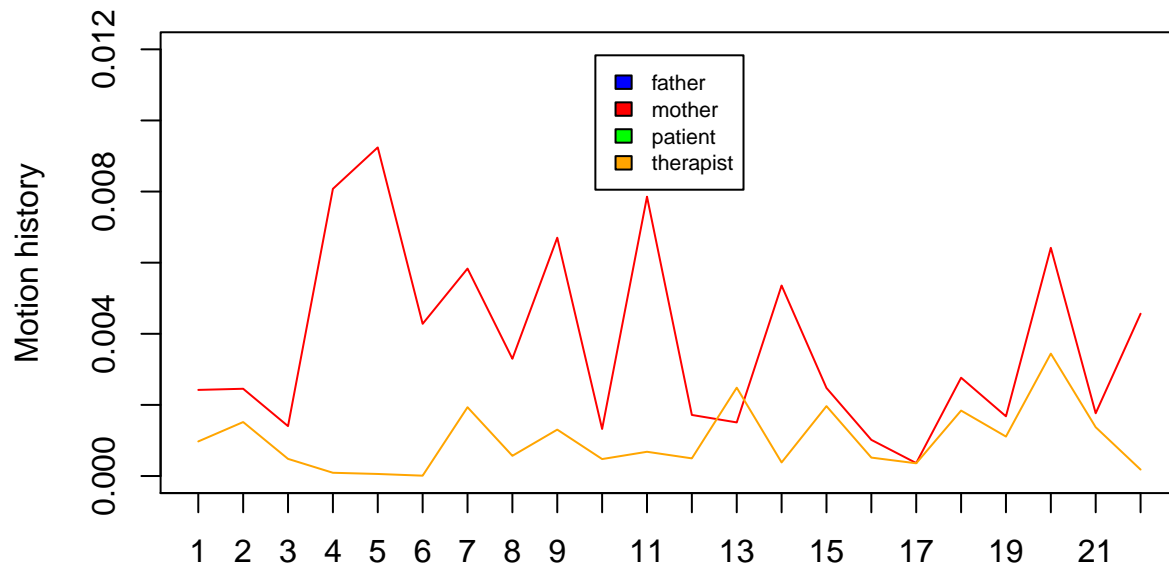
**Mean motion history (non overlapping minute intervals)  
on F1044F1 video**



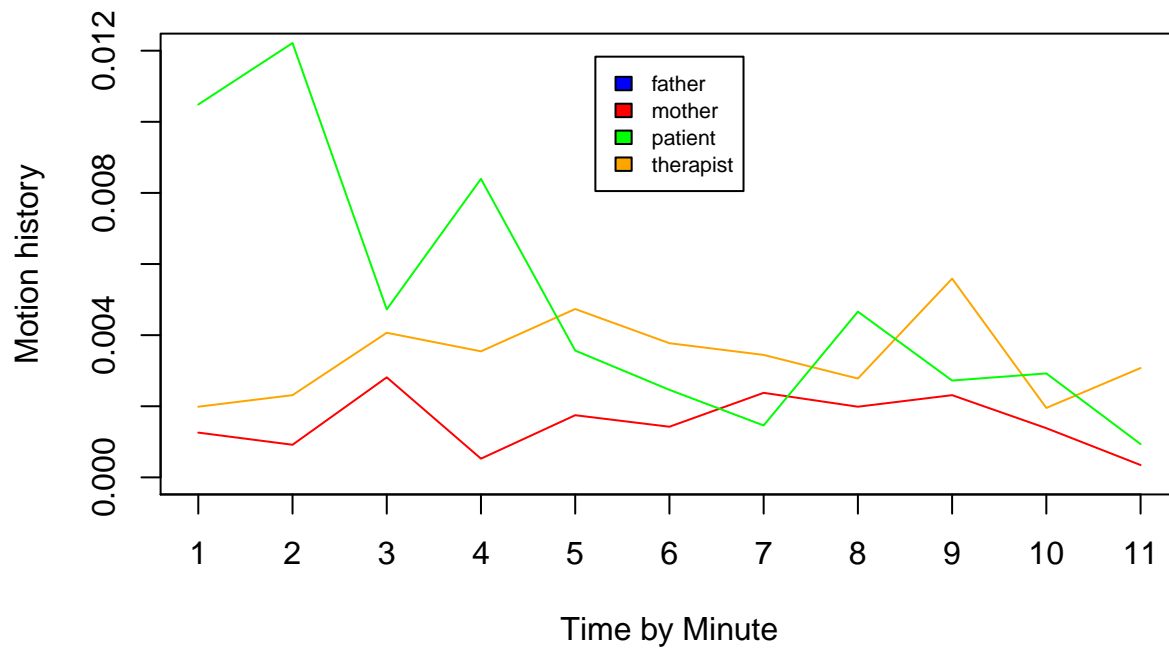
**Mean motion history (non overlapping minute intervals)  
on F1044F2 video**



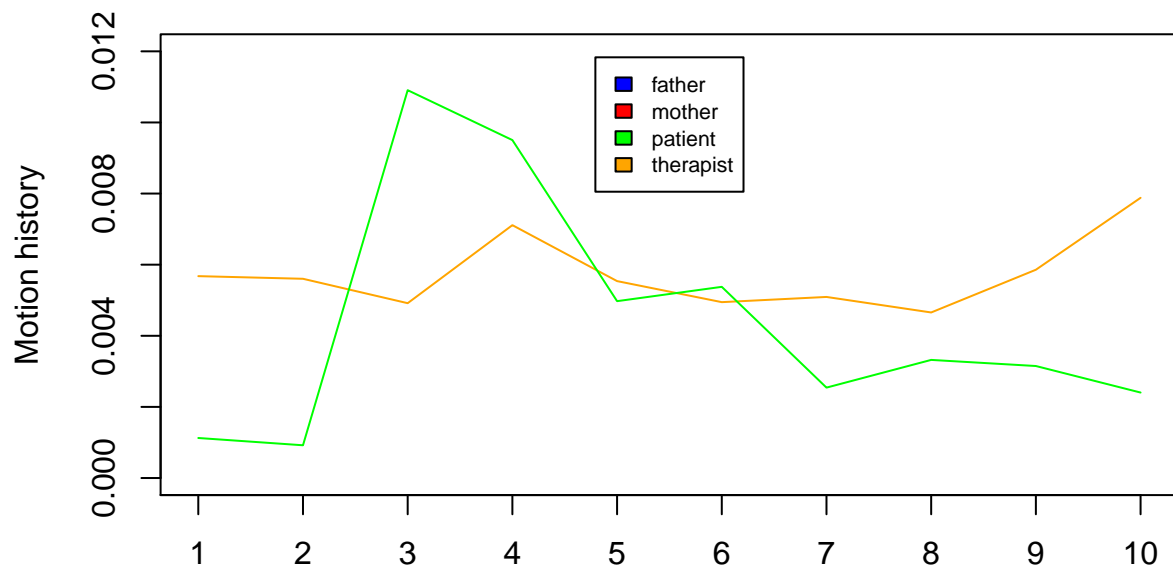
**Mean motion history (non overlapping minute intervals)  
on F1044G video**



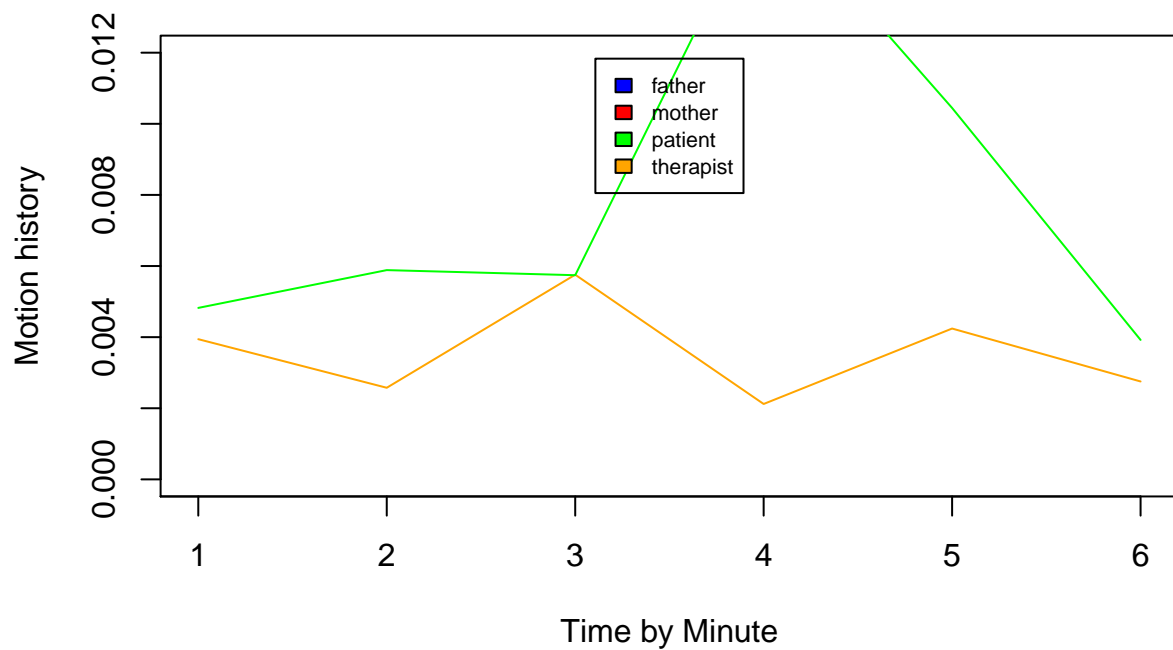
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044H1 video**



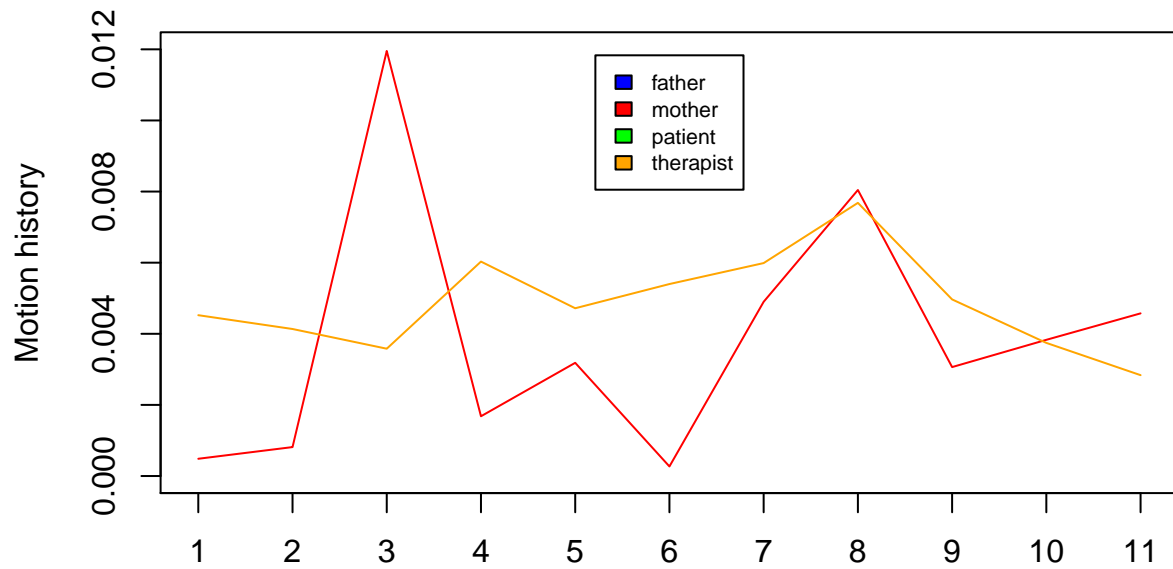
**Mean motion history (non overlapping minute intervals)  
on F1044H2 video**



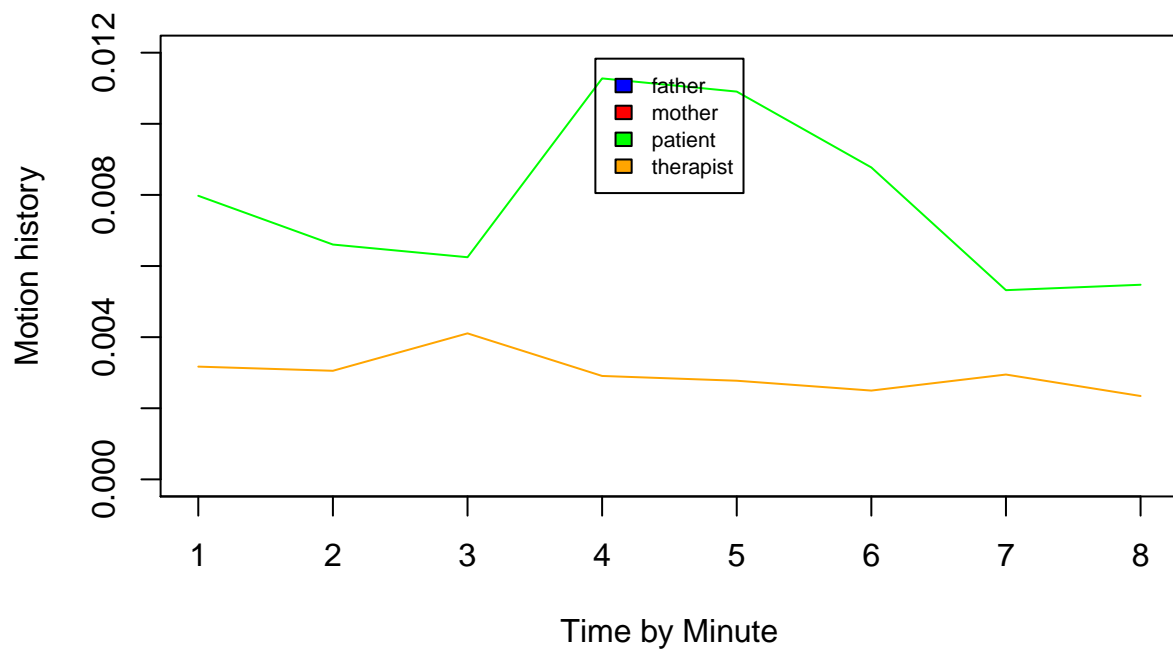
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044I1 video**



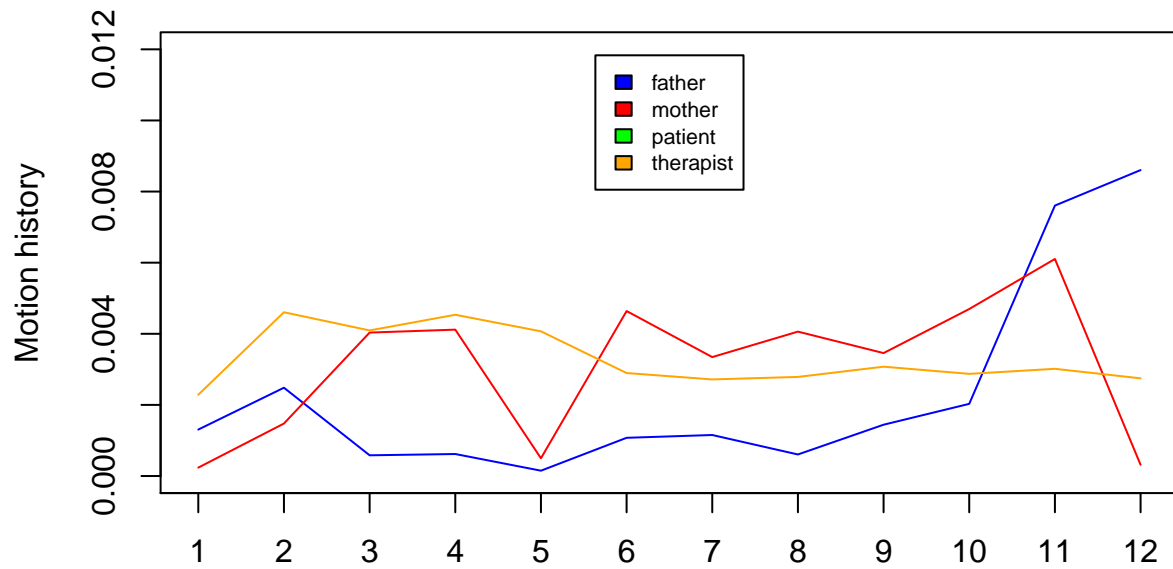
**Mean motion history (non overlapping minute intervals)  
on F1044I2 video**



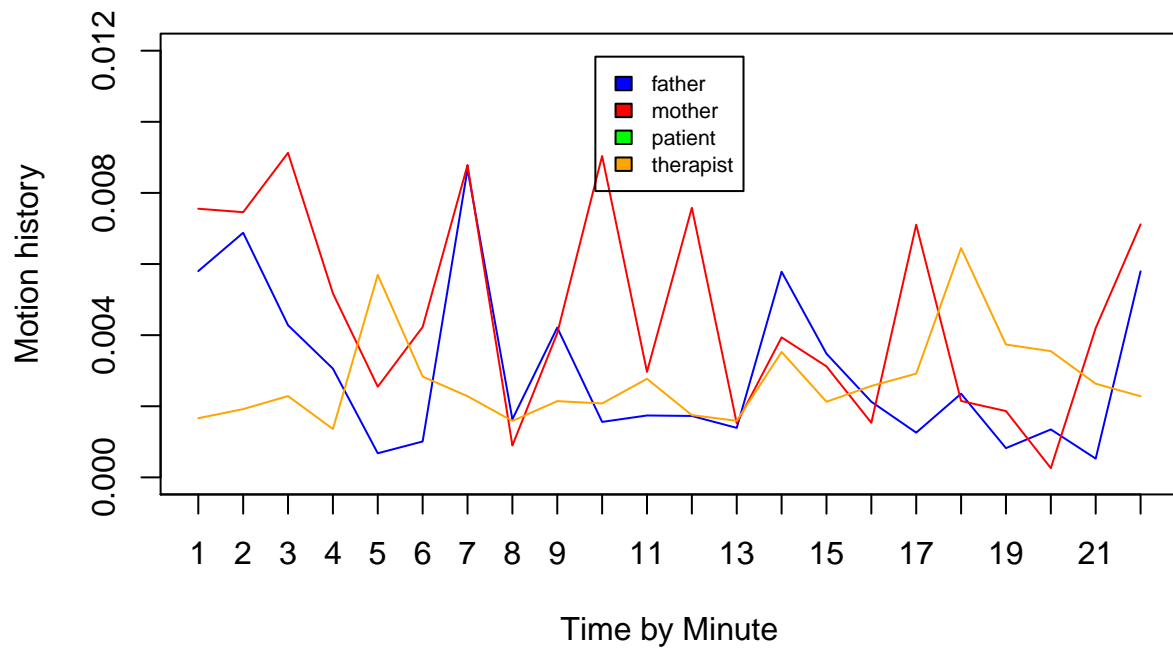
**Mean motion history (non overlapping minute intervals)  
on F1044L1 video**



**Mean motion history (non overlapping minute intervals)  
on F1044L2 video**

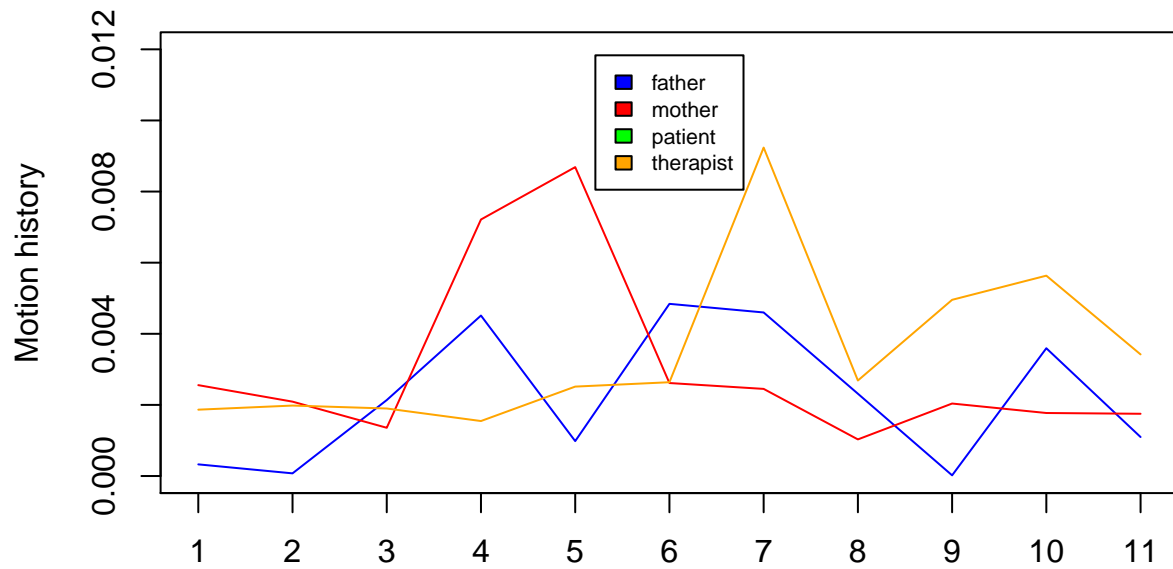


**Mean motion history (non overlapping minute intervals)  
on F1044M1 video**

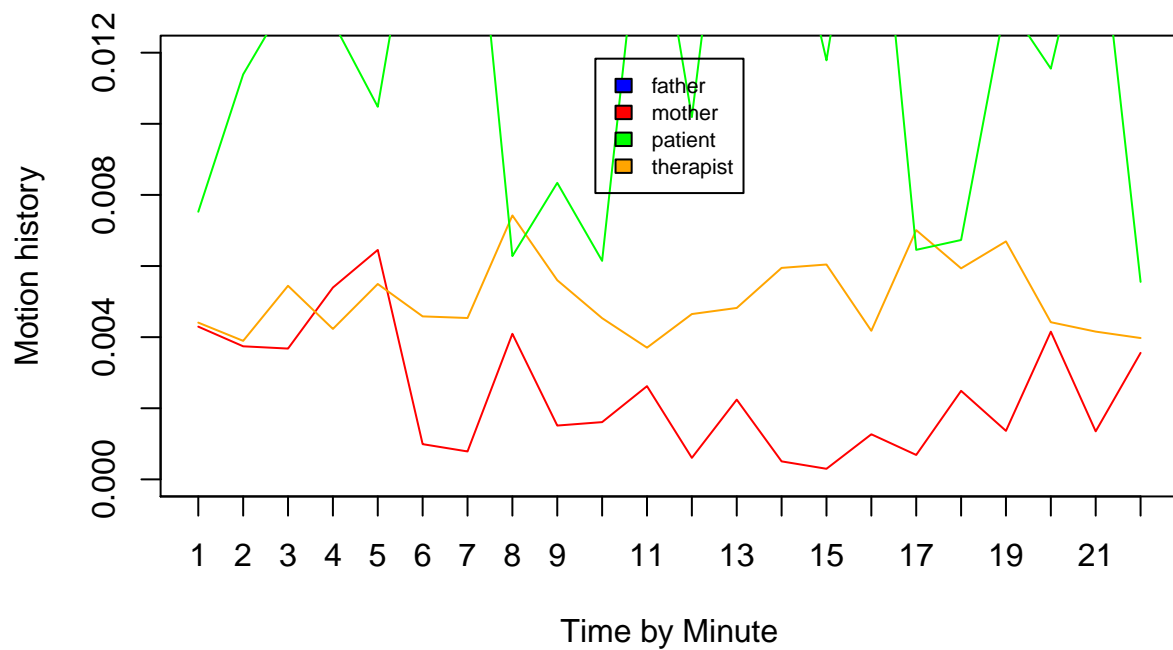




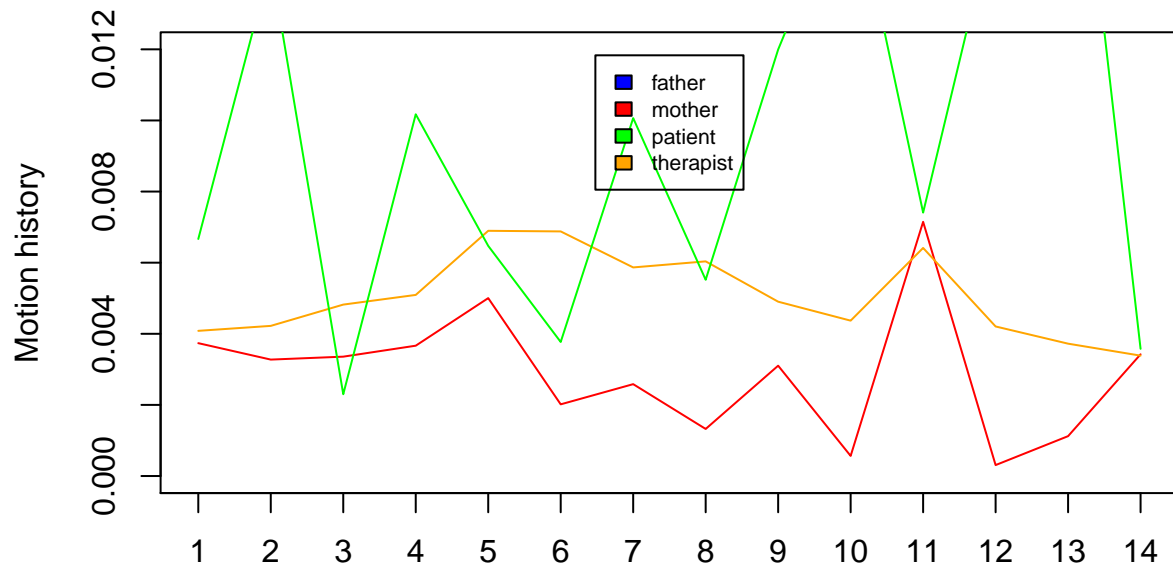
**Mean motion history (non overlapping minute intervals)  
on F1044M2 video**



Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044N video**

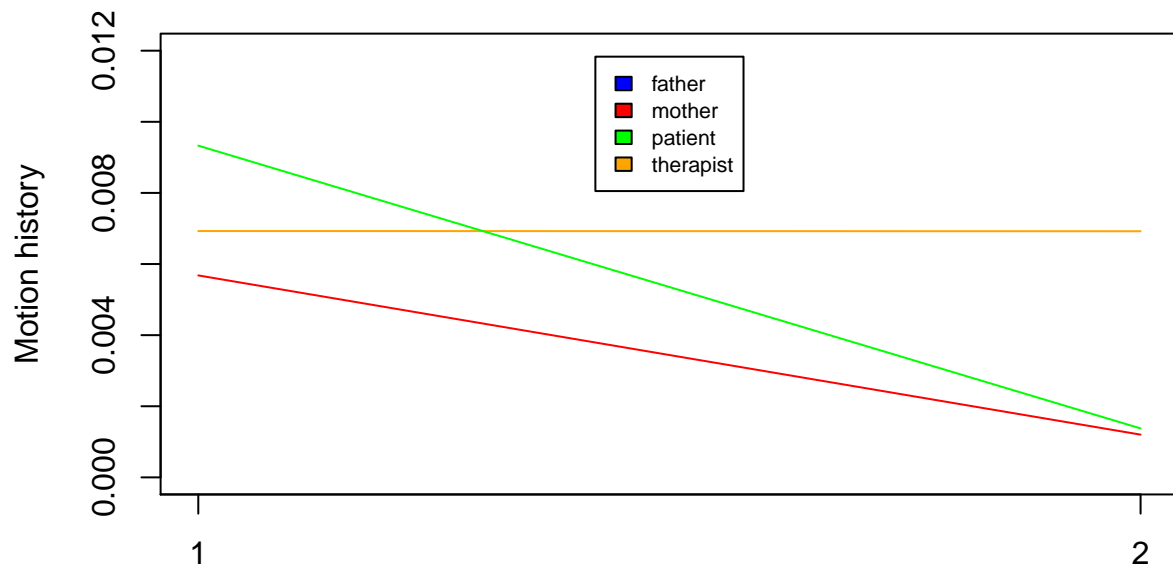


**Mean motion history (non overlapping minute intervals)  
on F1044O1 video**



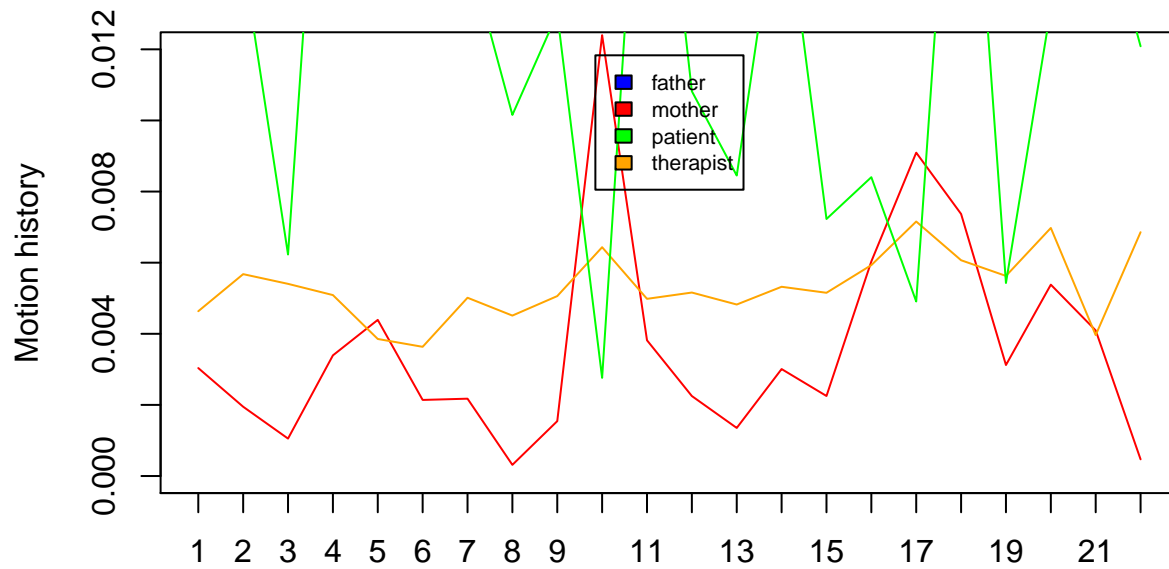
Time by Minute

**Mean motion history (non overlapping minute intervals)  
on F1044O2 video**

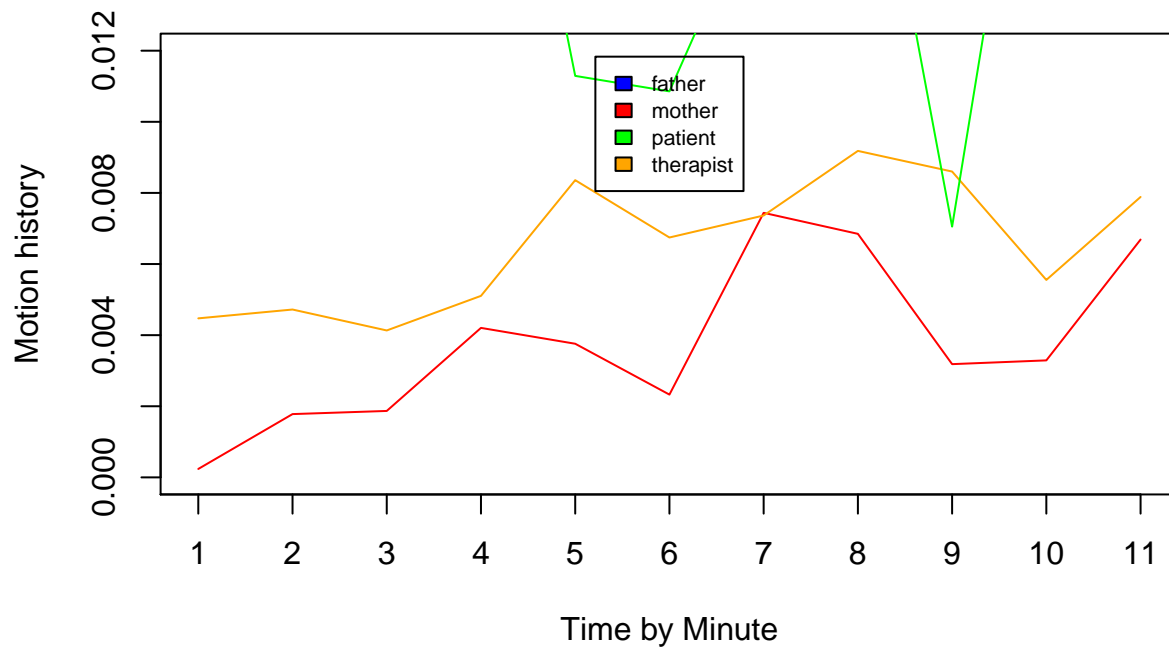


Time by Minute

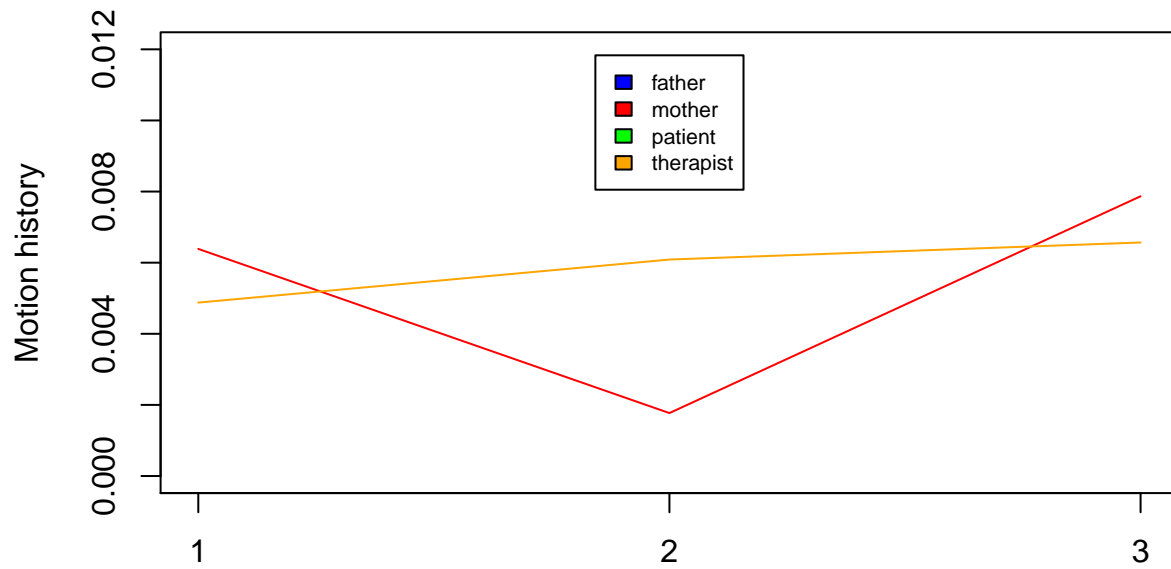
**Mean motion history (non overlapping minute intervals)  
on F1044P video**



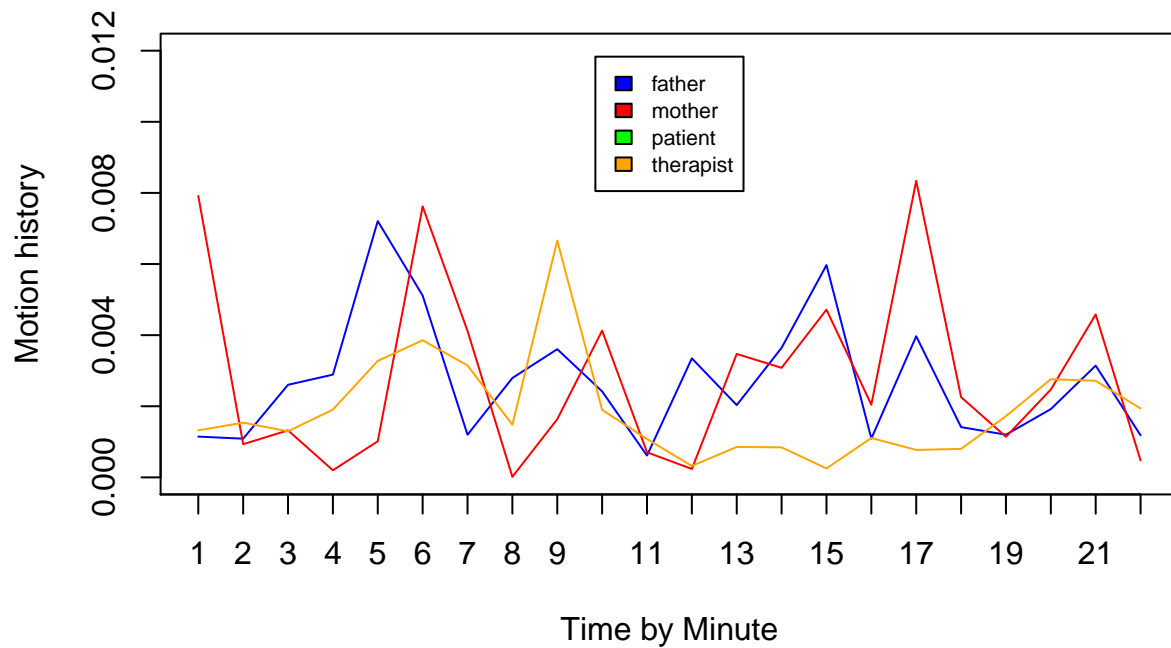
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044Q1 video**



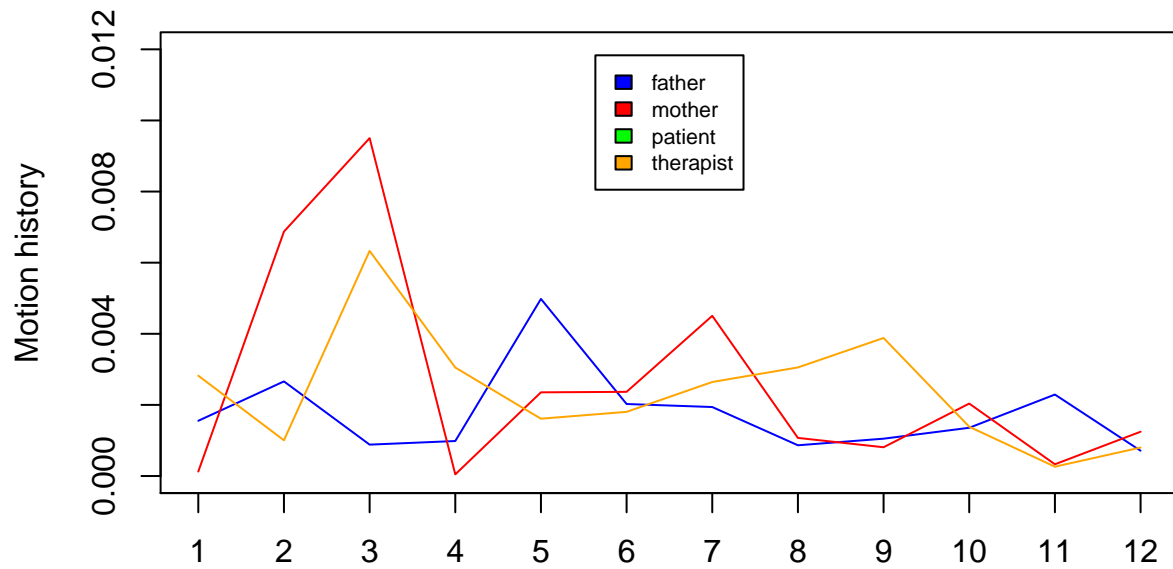
**Mean motion history (non overlapping minute intervals)  
on F1044Q2 video**



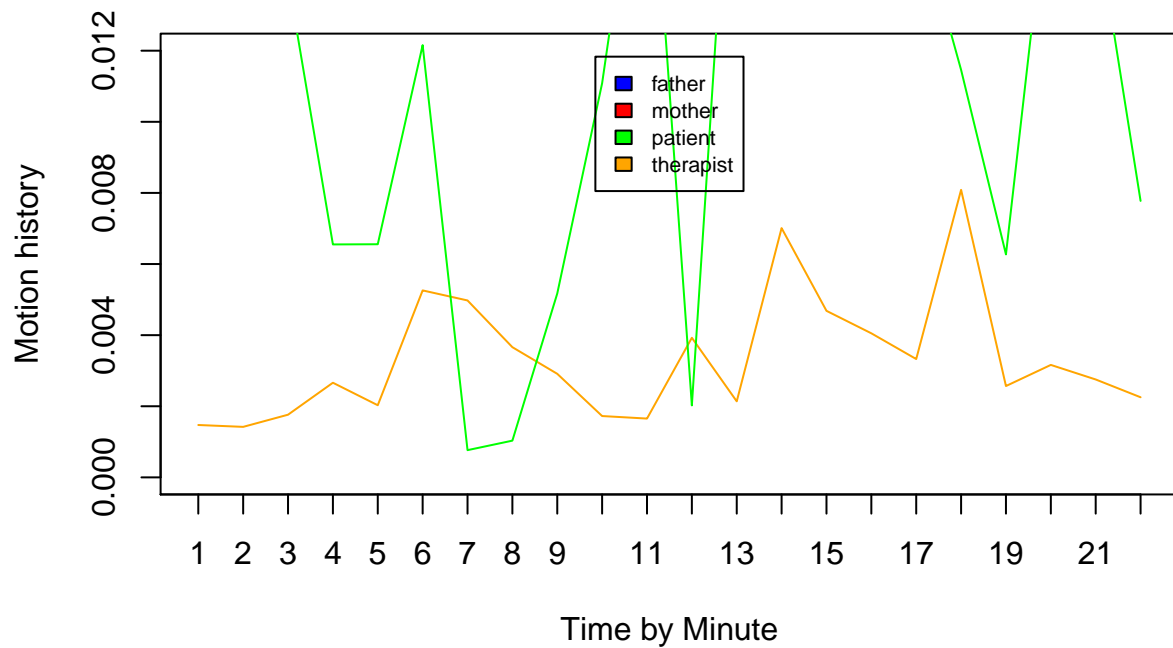
**Mean motion history (non overlapping minute intervals)  
on F1044R1 video**



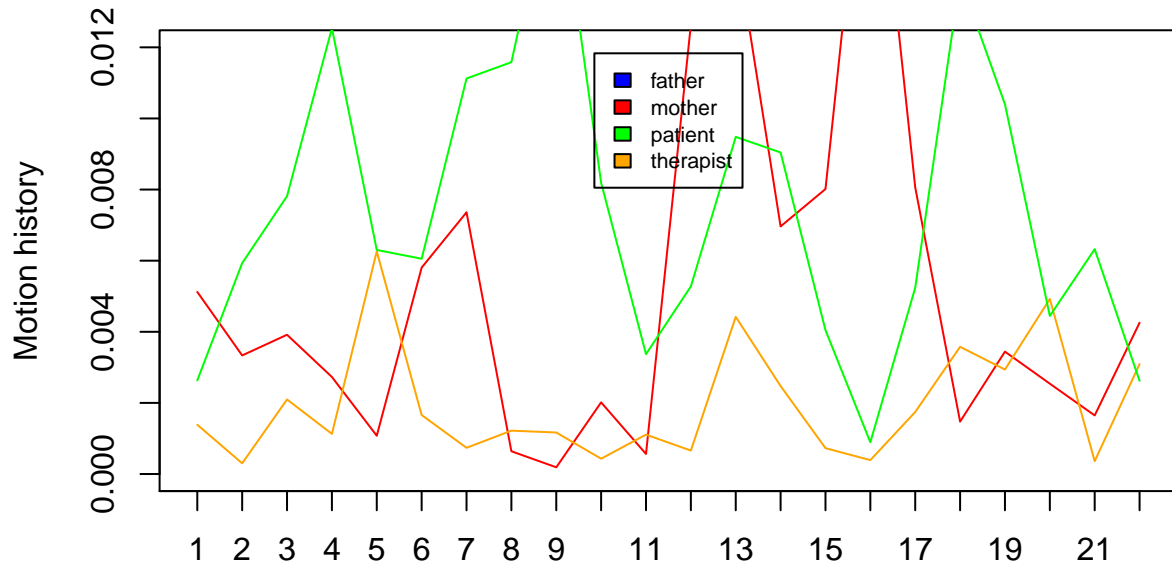
**Mean motion history (non overlapping minute intervals)  
on F1044R2 video**



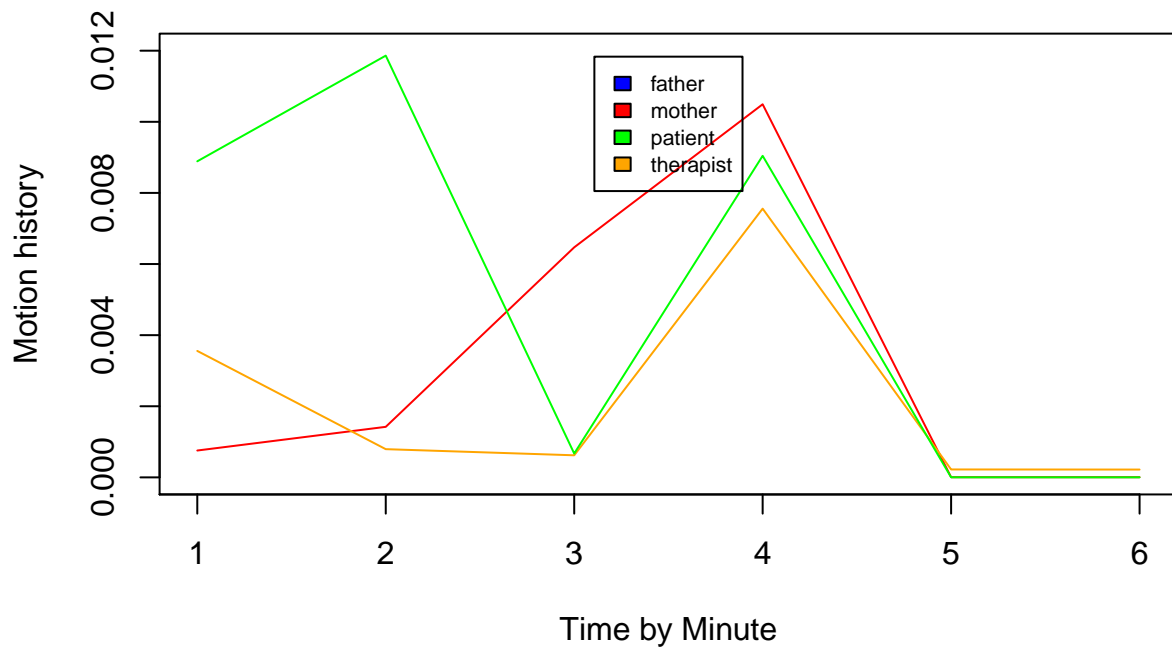
**Mean motion history (non overlapping minute intervals)  
on F1044A1 video**



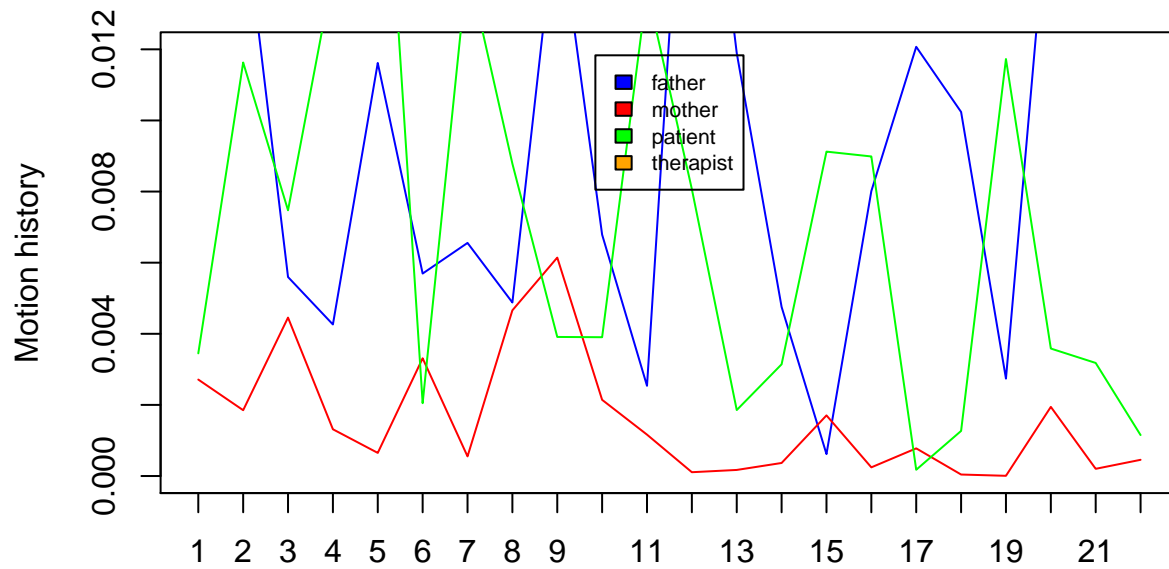
**Mean motion history (non overlapping minute intervals)  
on F1044B1 video**



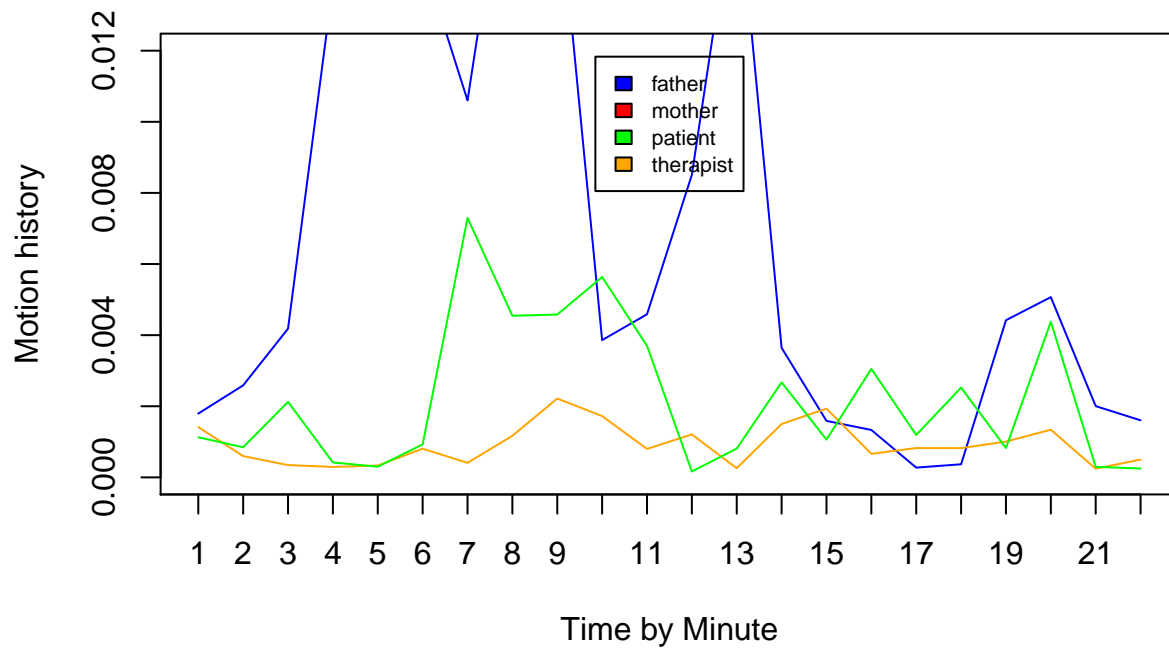
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044B2 video**



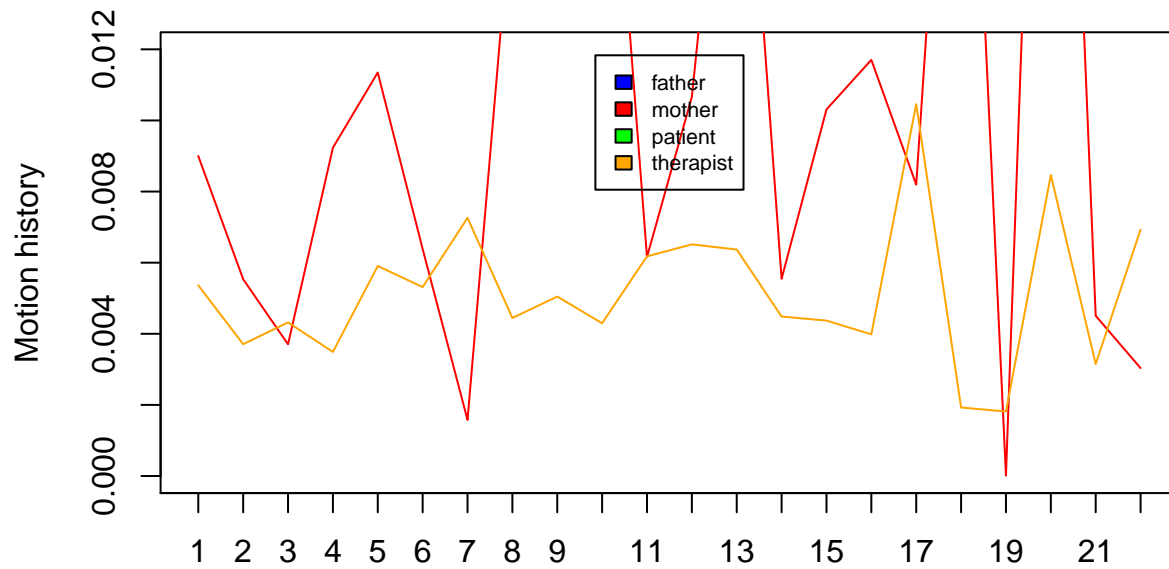
**Mean motion history (non overlapping minute intervals)  
on F1044C1 video**



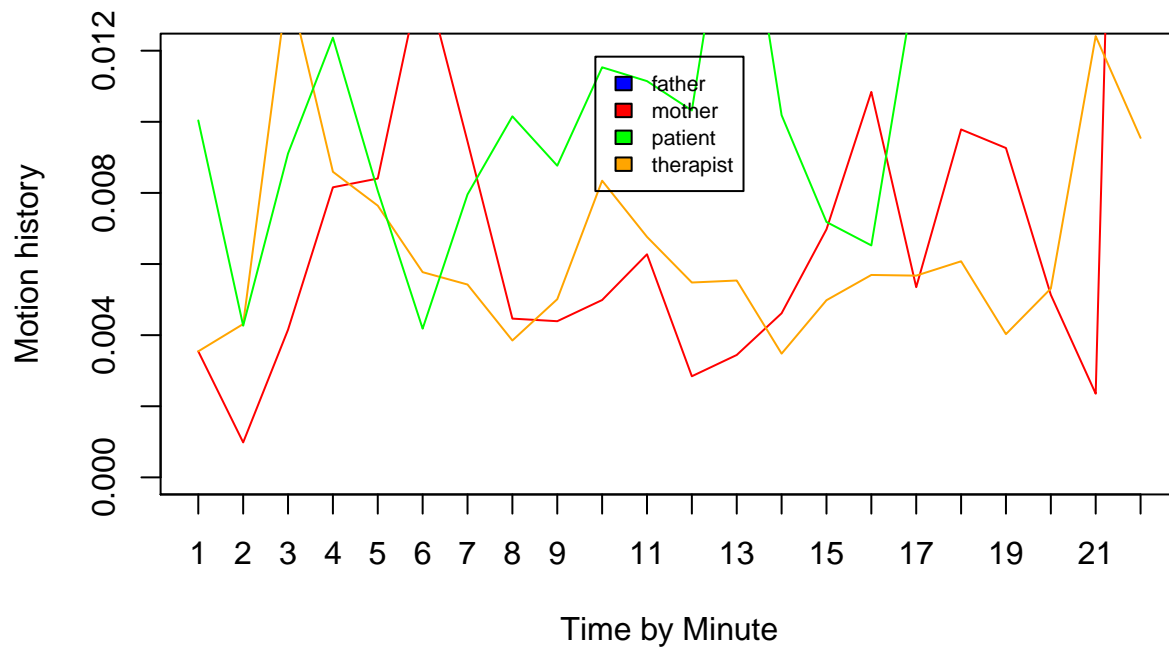
**Mean motion history (non overlapping minute intervals)  
on F1044D2 video**



**Mean motion history (non overlapping minute intervals)  
on F1044A1 video**

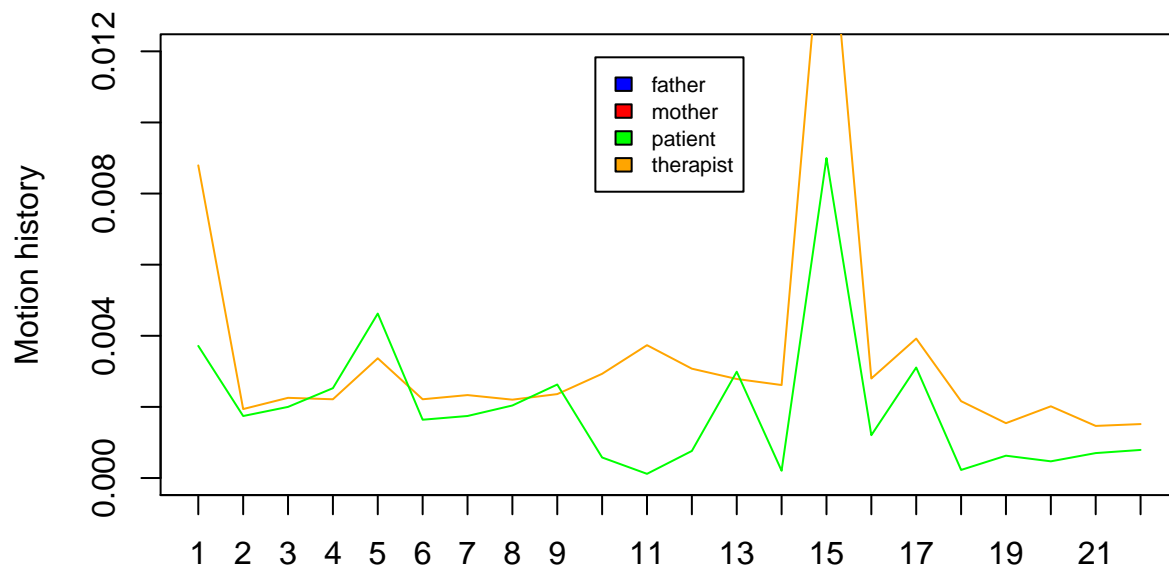


Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044A2 video**

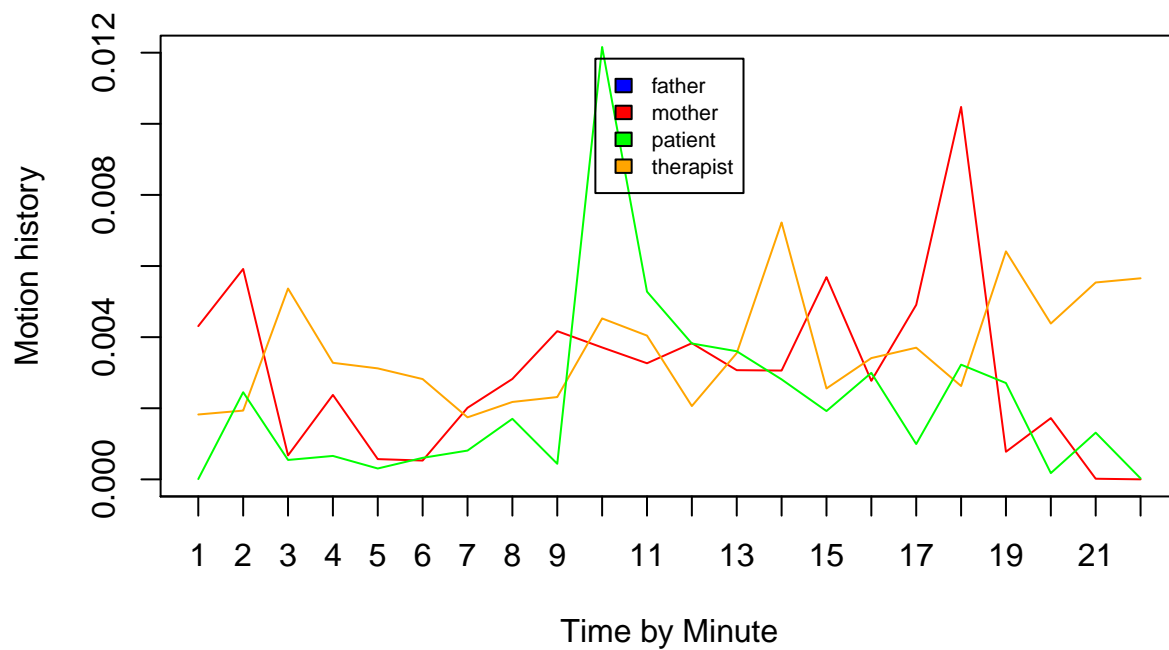




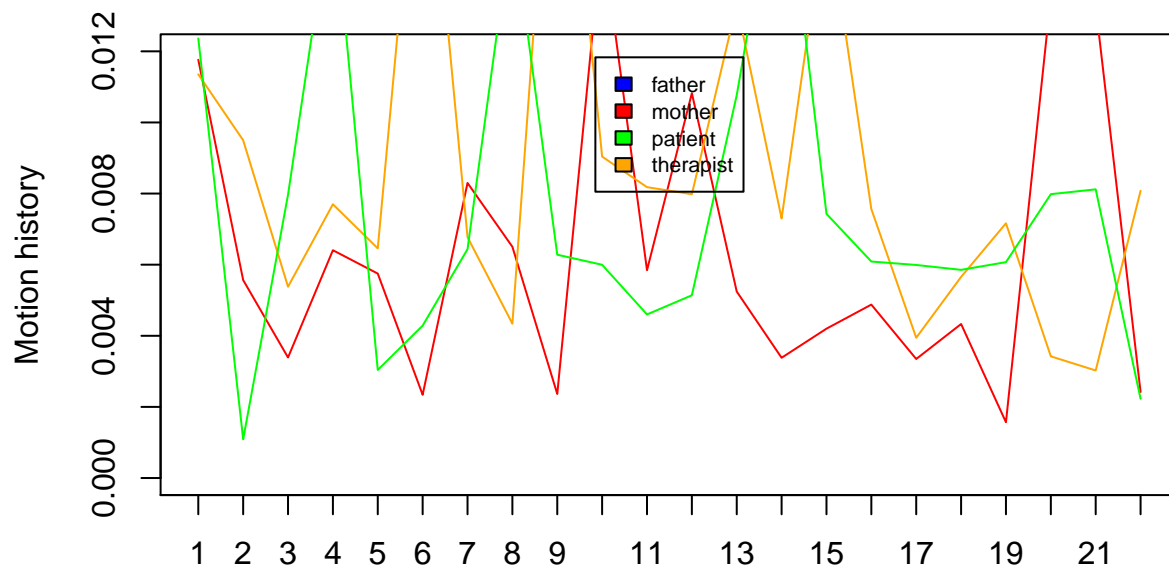
**Mean motion history (non overlapping minute intervals)  
on F1044B1 video**



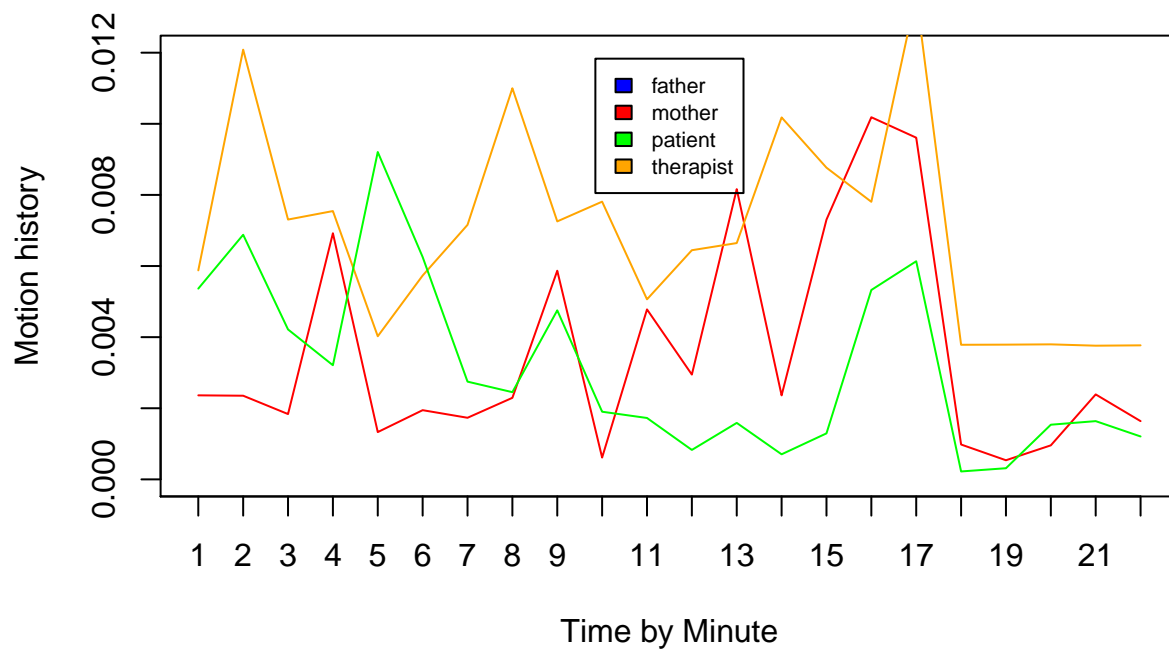
**Mean motion history (non overlapping minute intervals)  
on F1044B2 video**

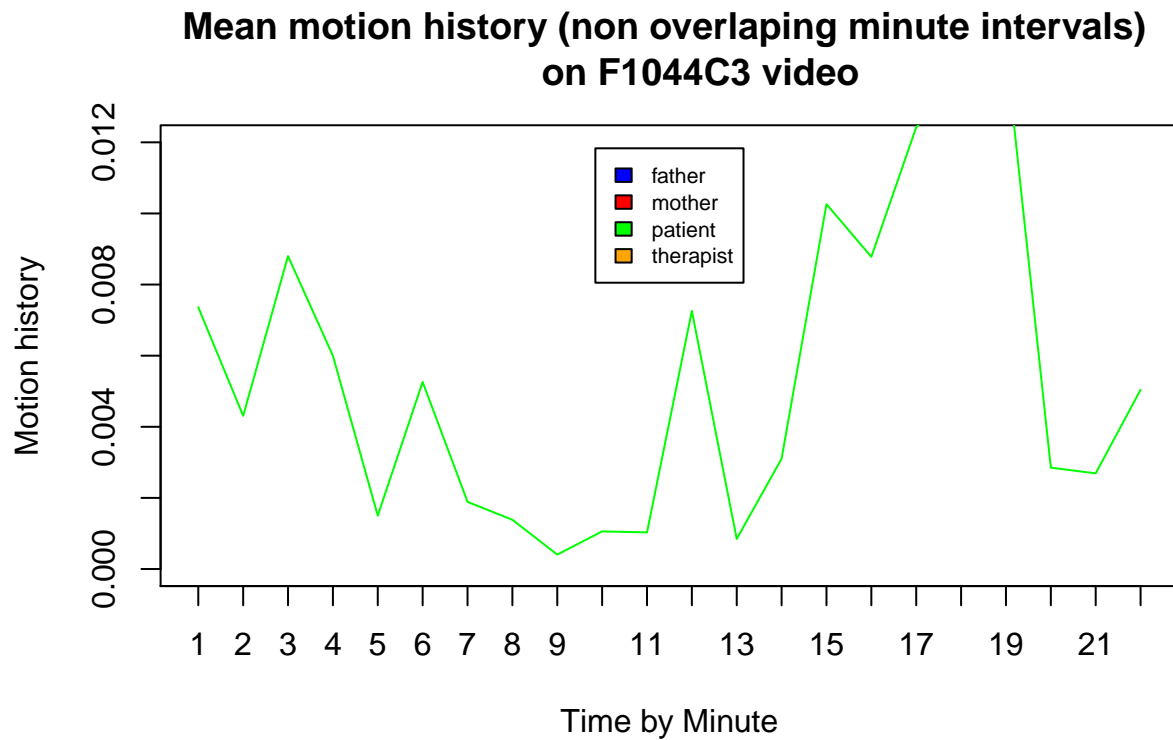


**Mean motion history (non overlapping minute intervals)  
on F1044A2 video**



Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044C2 video**





## Mean log motion history by minute plots

```
for (i in 1:NumberOfvideos){
  fatherMinute<- MeanMotionByTime("logFather", indexOfvideos=i, interval=1500, data)

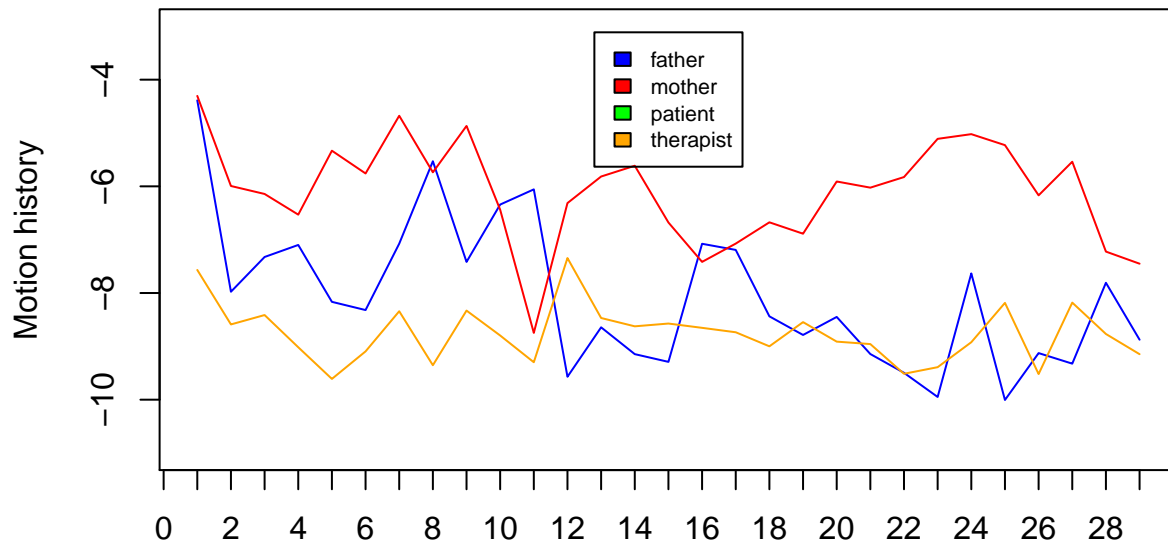
  MotherMinute<- MeanMotionByTime("logMother", indexOfvideos=i, interval=1500, data)

  TherapistMinute<- MeanMotionByTime("logTherapist", indexOfvideos=i, interval=1500, data)

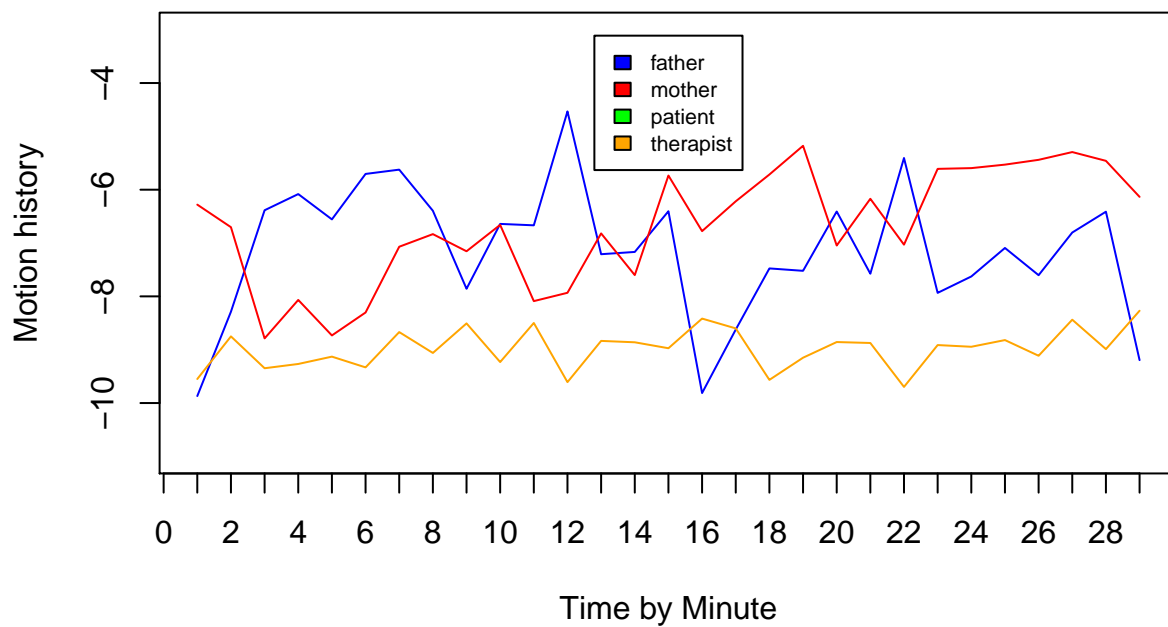
  PatientMinute<- MeanMotionByTime("logPatient", indexOfvideos=i, interval=1500, data)

  par(mar=c(4,4,4,2))
  plot (1:length(fatherMinute), fatherMinute, type="l", col="blue",
        main=paste("Mean motion history (non overlapping minute intervals)
on F1044", labelvideolist[i], " video" , sep=""),
        ylab="Motion history", xlab="Time by Minute", ylim=c(-11, -3),
        xaxp=c(0, length(fatherMinute), length(fatherMinute)))
  lines(MotherMinute, col="red")
  lines(TherapistMinute, col="orange")
  lines(PatientMinute, col="green")
  legend("top", inset=.05, ParticipantsList,
        fill=colOrderList, cex=0.7)}
```

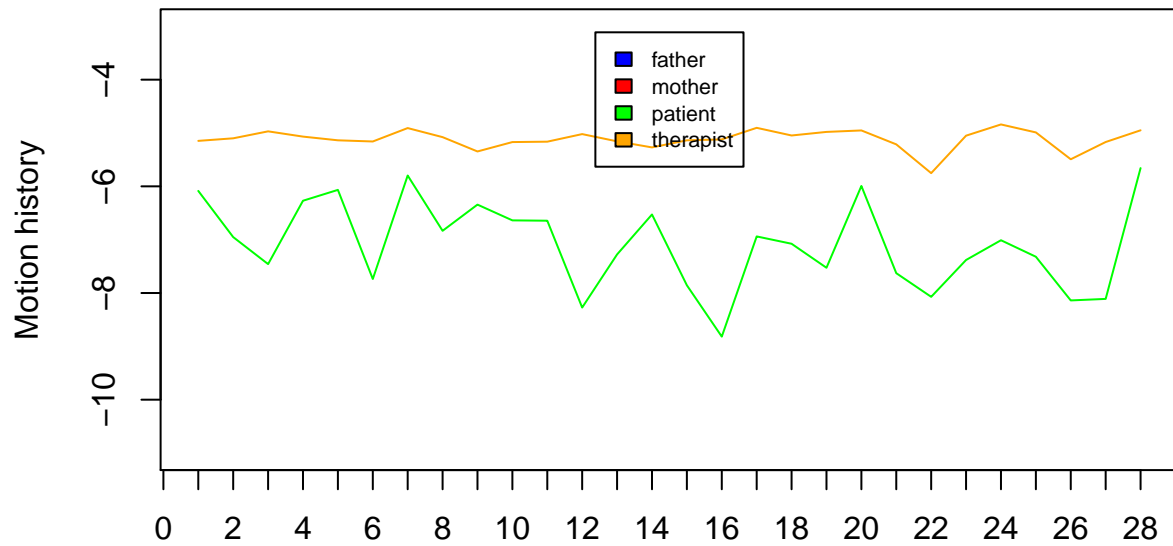
**Mean motion history (non overlapping minute intervals)  
on F1044A1 video**



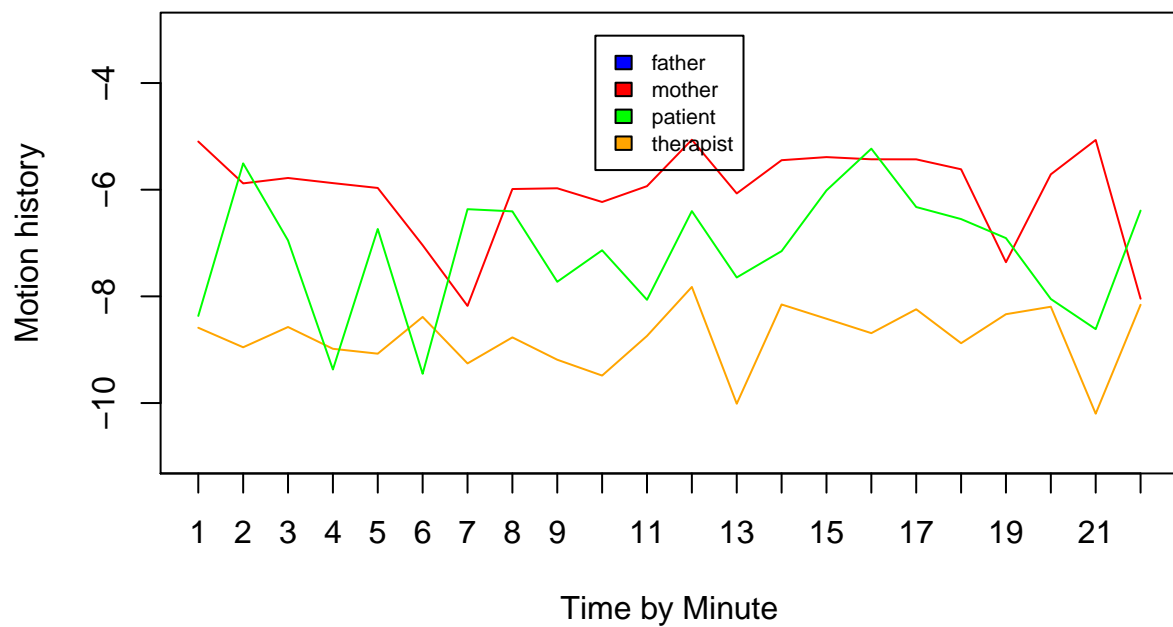
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044A2 video**



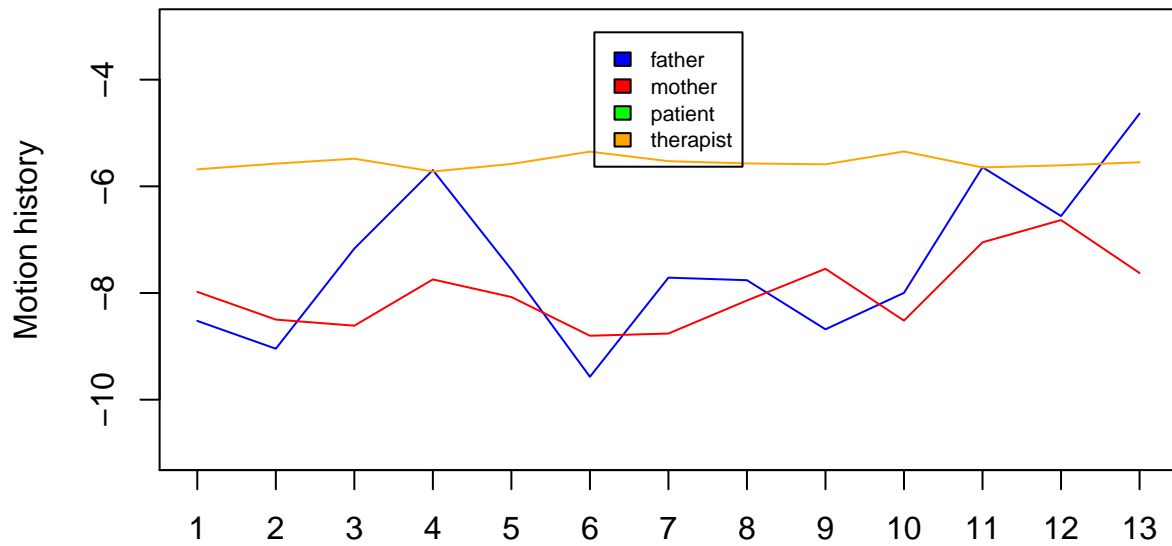
**Mean motion history (non overlapping minute intervals)  
on F1044B2 video**



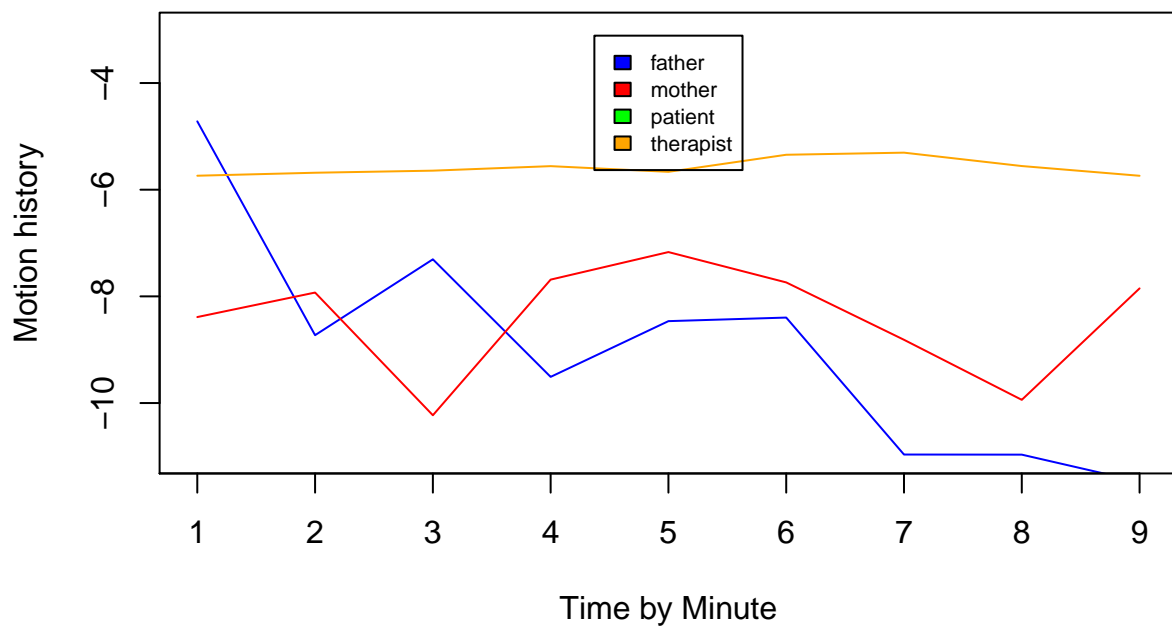
**Mean motion history (non overlapping minute intervals)  
on F1044C1 video**



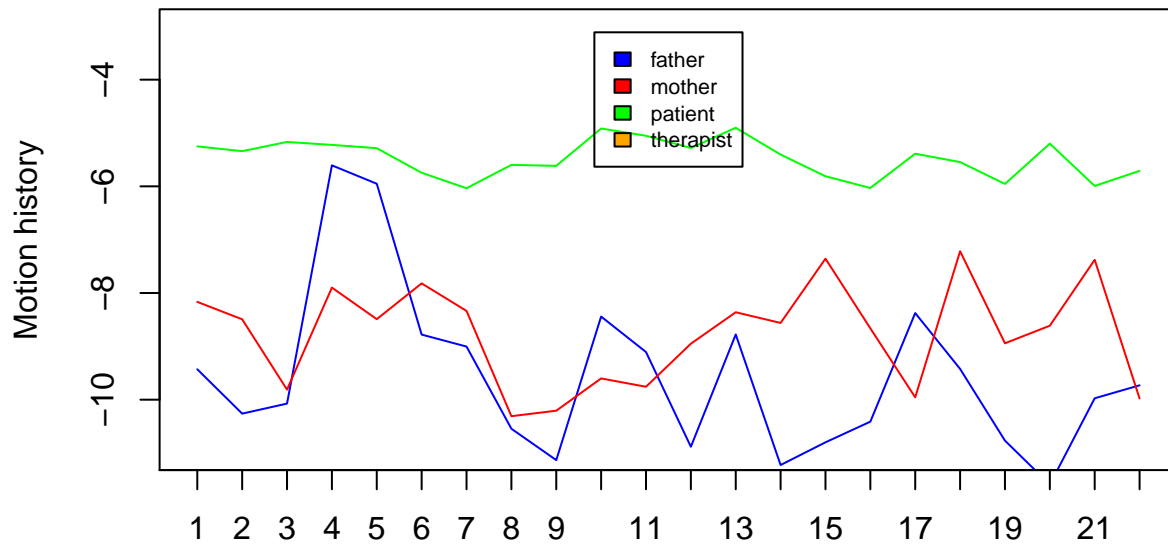
**Mean motion history (non overlapping minute intervals)  
on F1044C1 video**



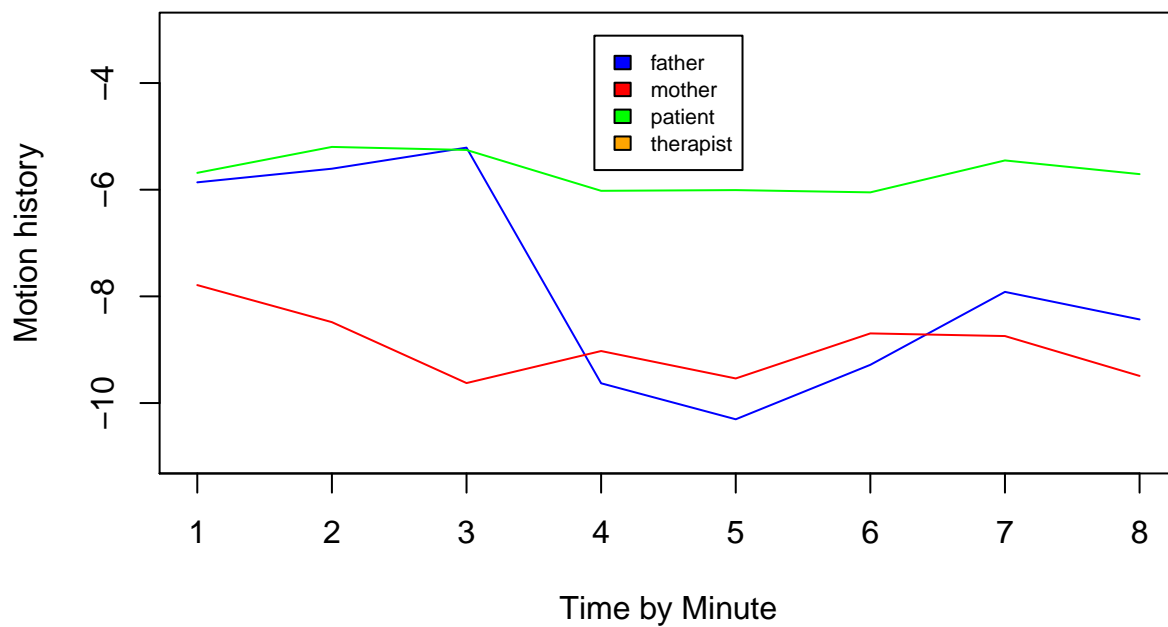
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044C2 video**



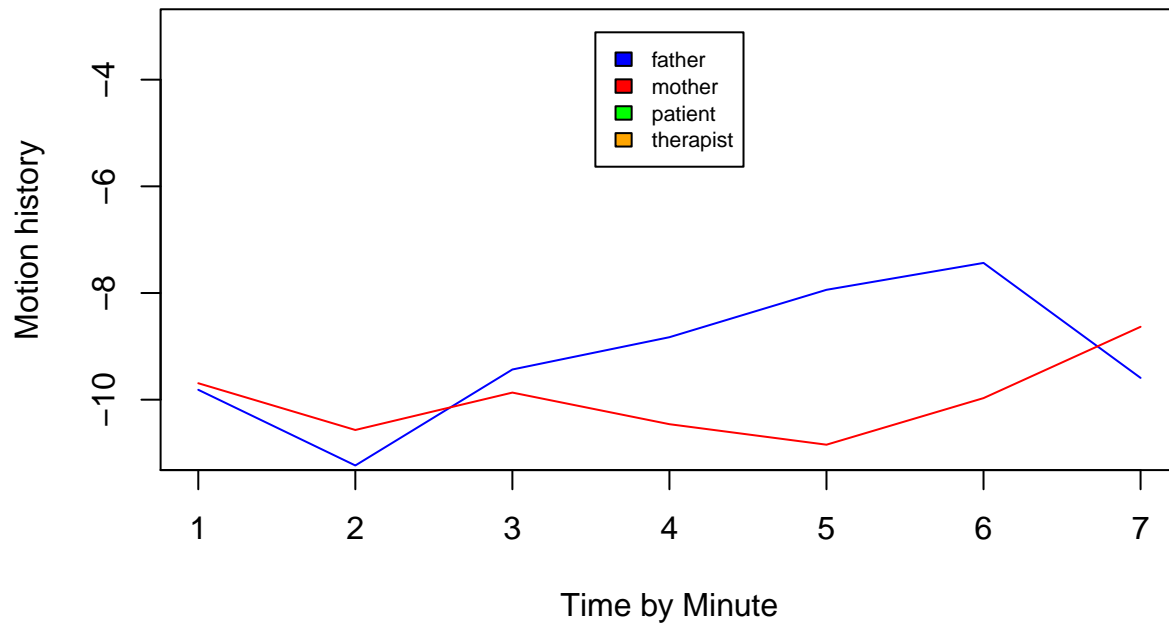
**Mean motion history (non overlapping minute intervals)  
on F1044D1 video**



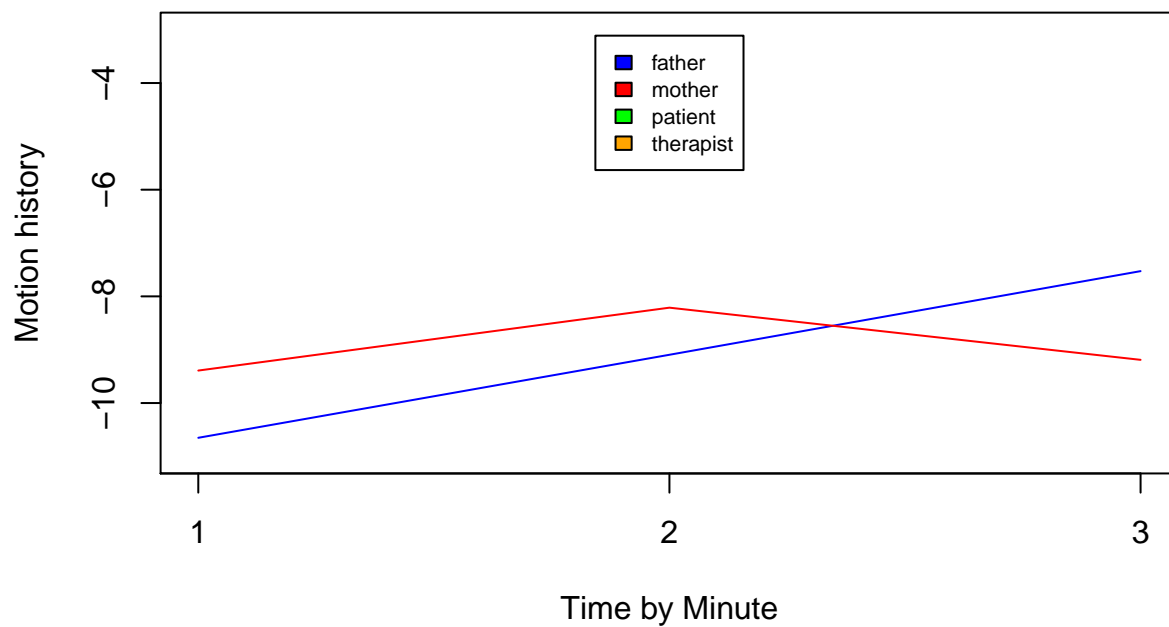
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044D2 video**



**Mean motion history (non overlapping minute intervals)  
on F1044E1 video**

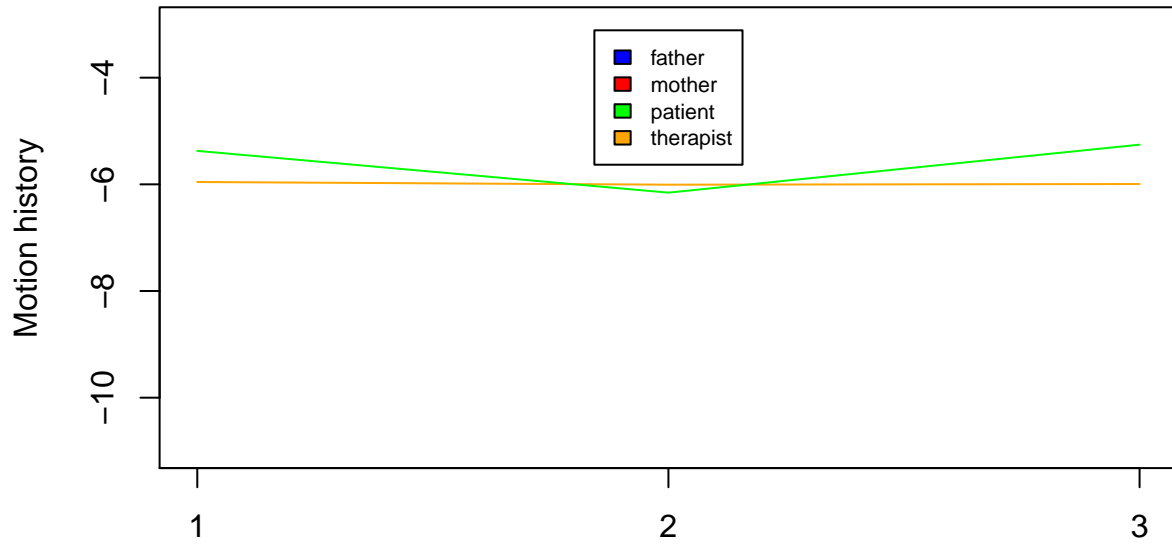


**Mean motion history (non overlapping minute intervals)  
on F1044E2 video**

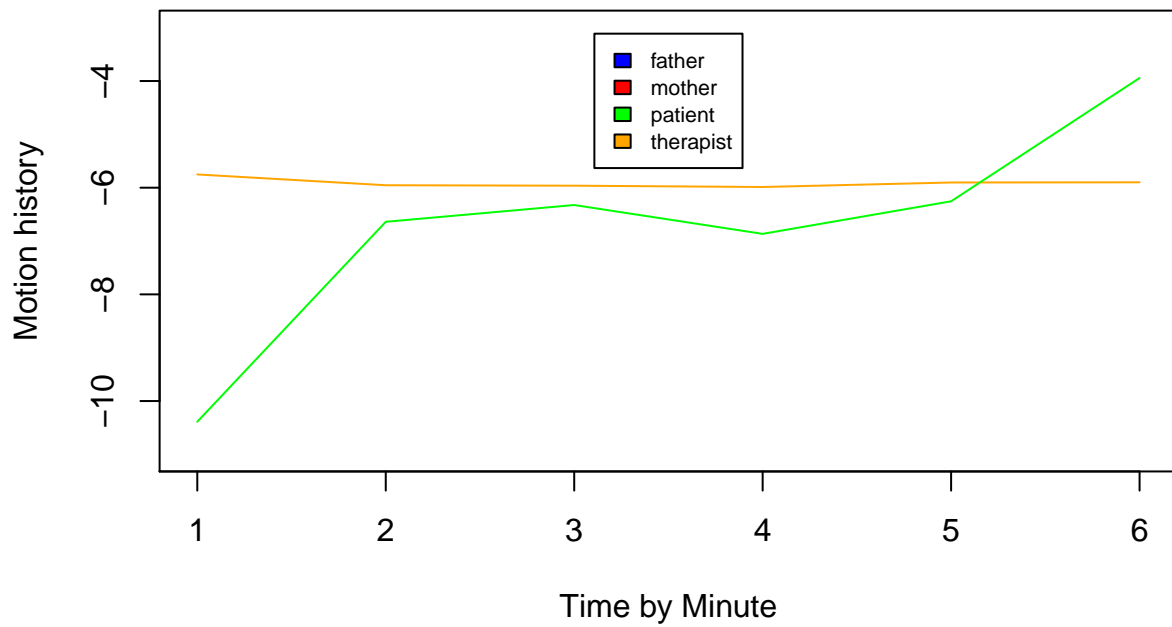




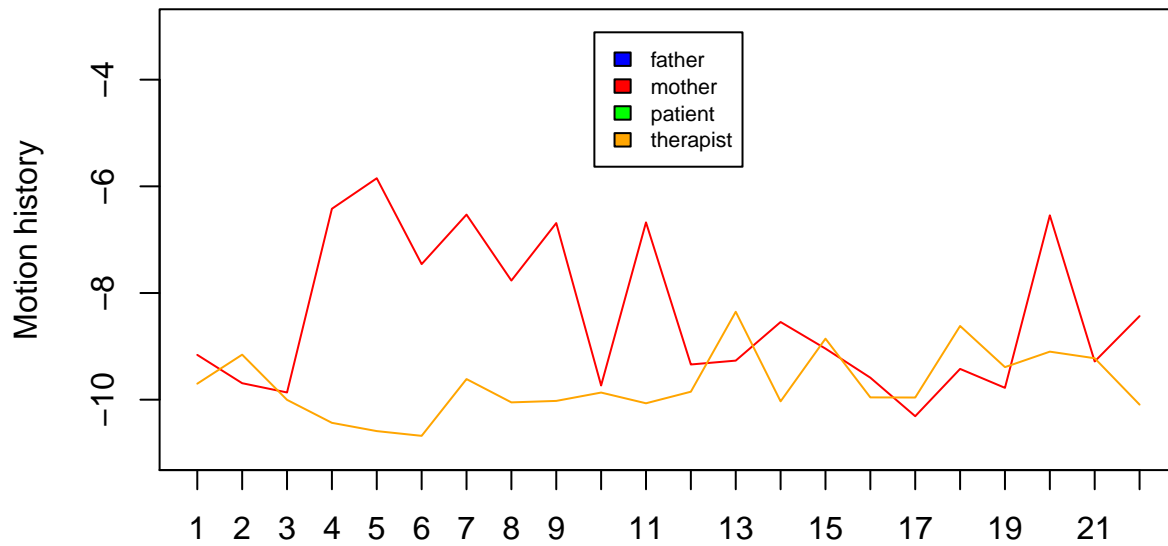
**Mean motion history (non overlapping minute intervals)  
on F1044F1 video**



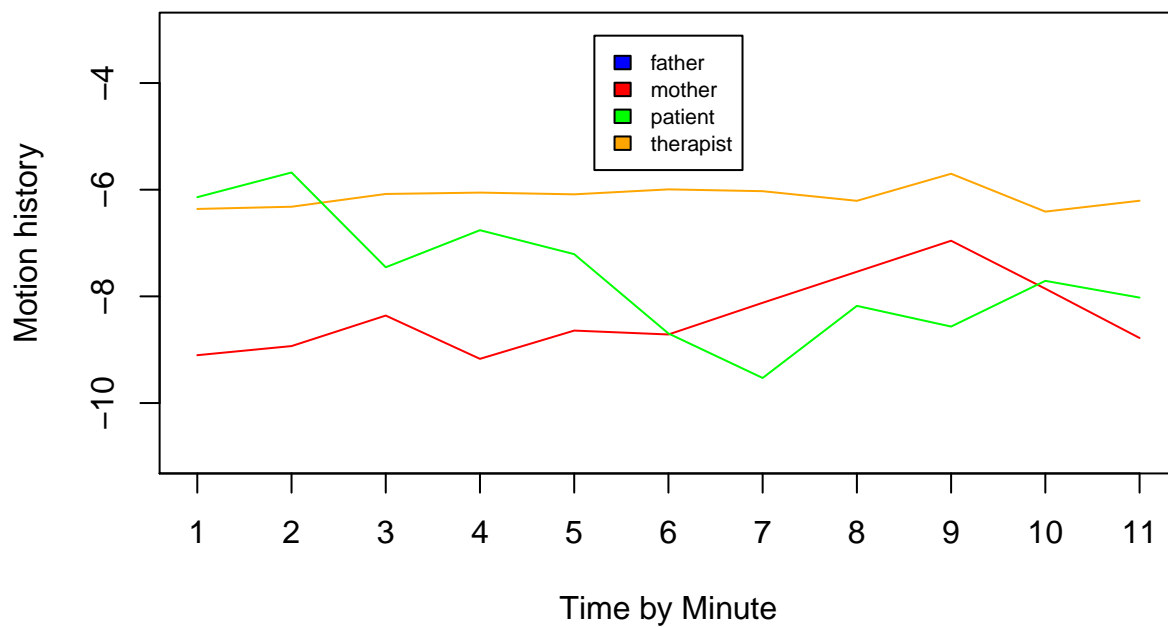
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044F2 video**



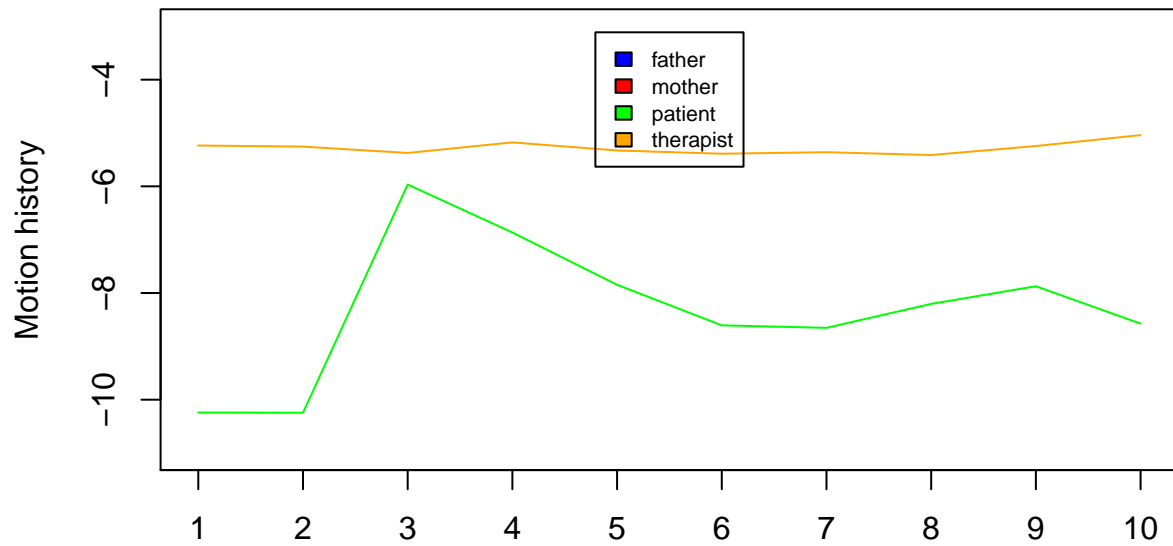
**Mean motion history (non overlapping minute intervals)  
on F1044G video**



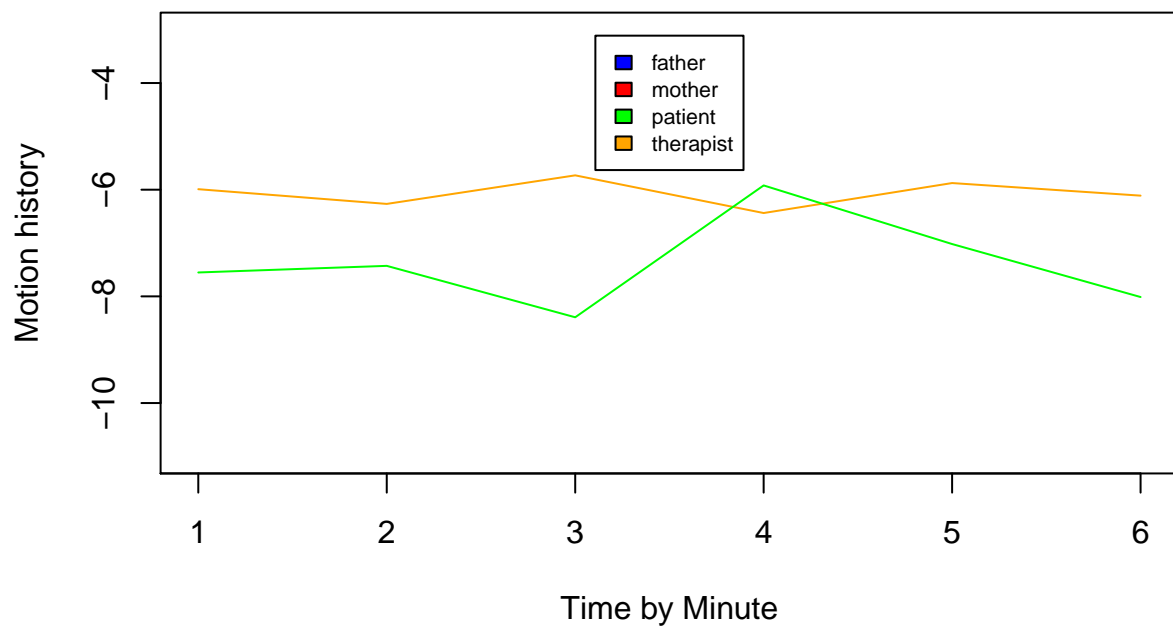
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044H1 video**



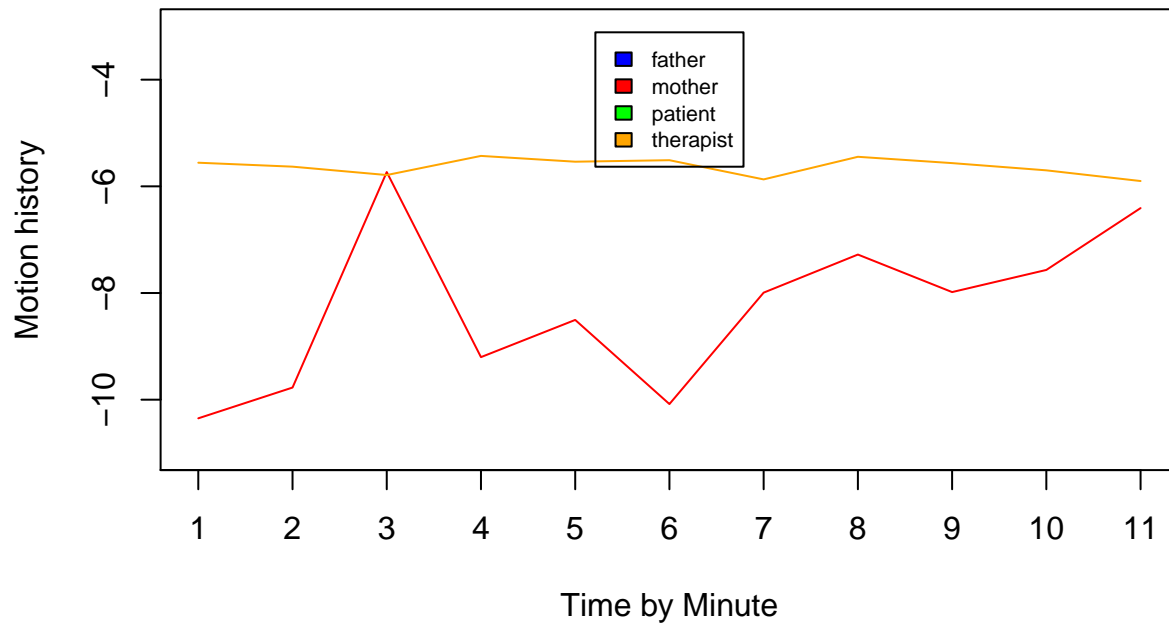
**Mean motion history (non overlapping minute intervals)  
on F1044H2 video**



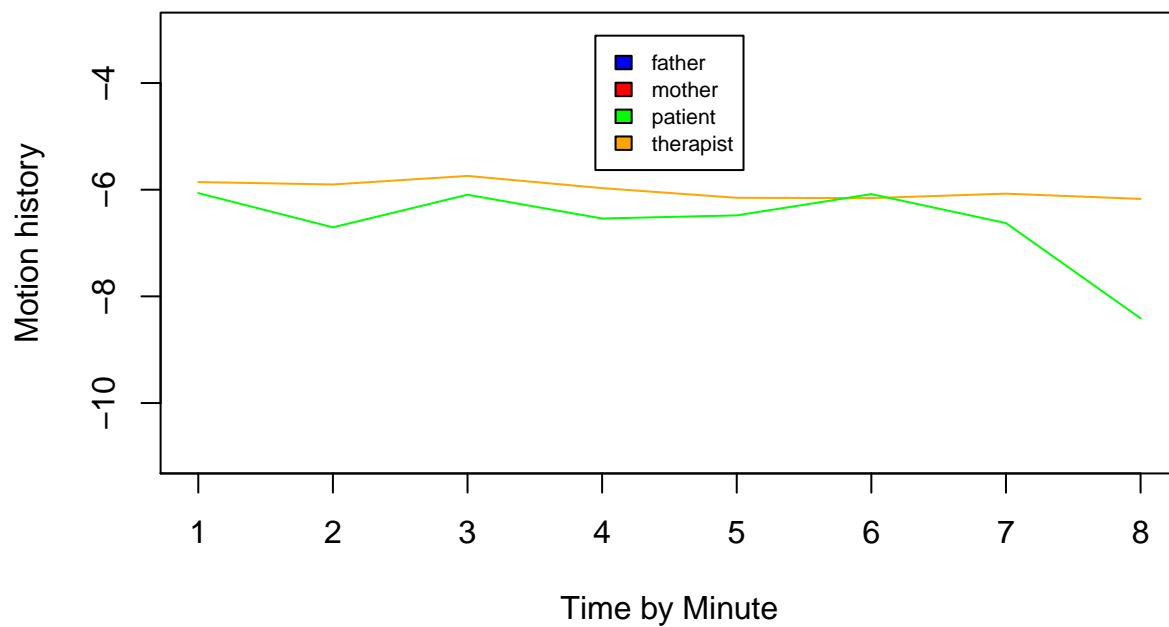
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044I1 video**



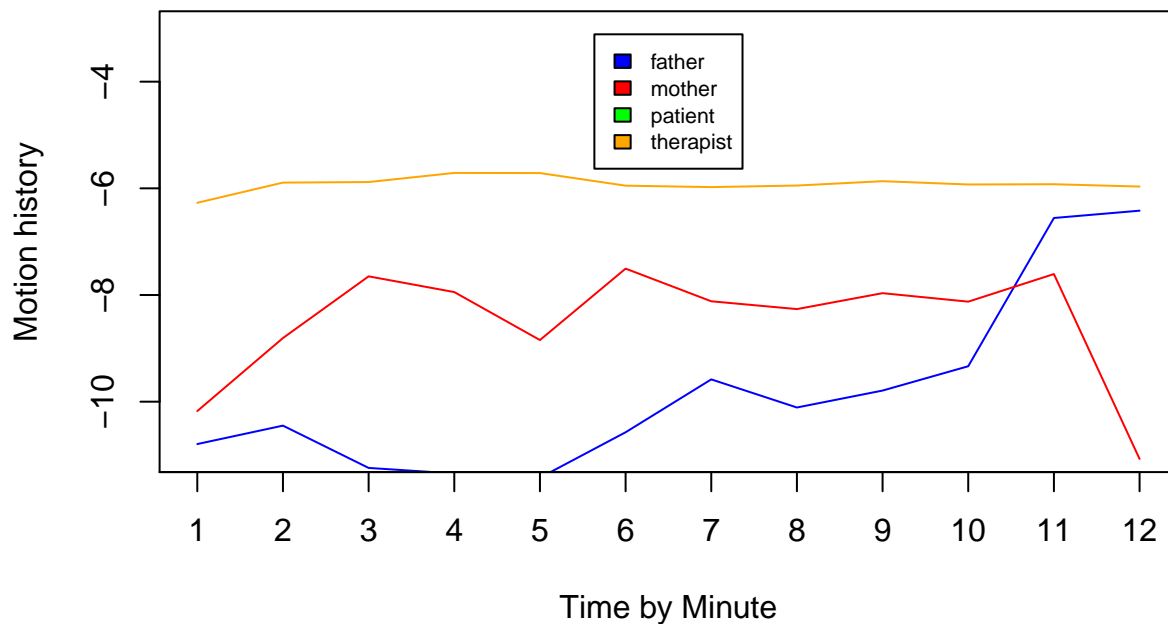
**Mean motion history (non overlapping minute intervals)  
on F1044I2 video**



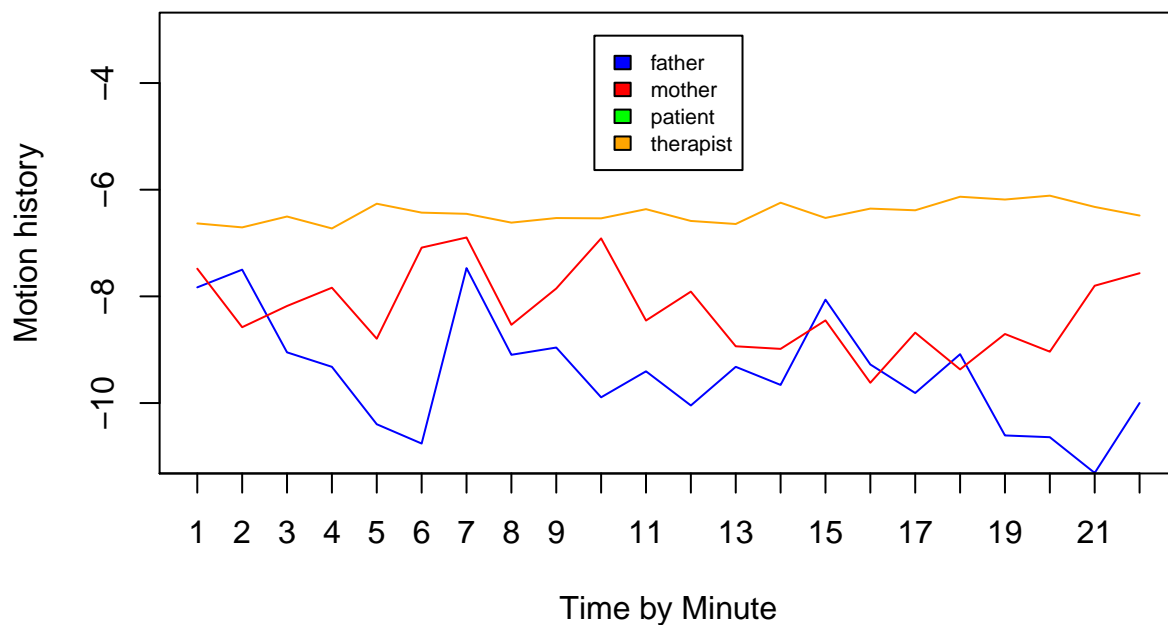
**Mean motion history (non overlapping minute intervals)  
on F1044L1 video**



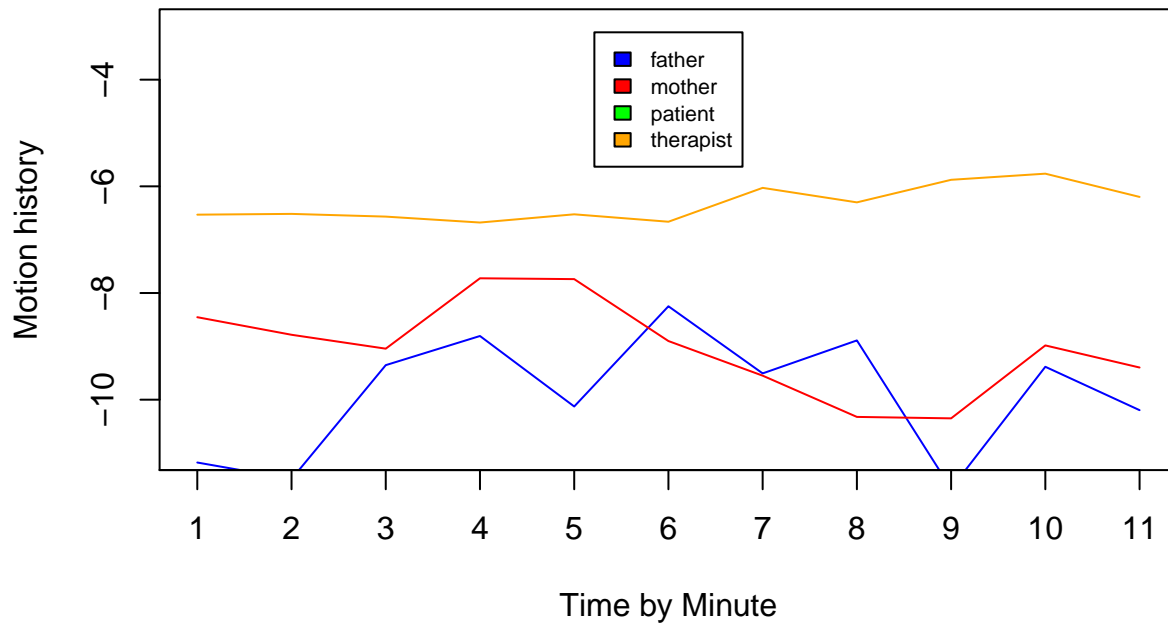
**Mean motion history (non overlapping minute intervals)  
on F1044L2 video**



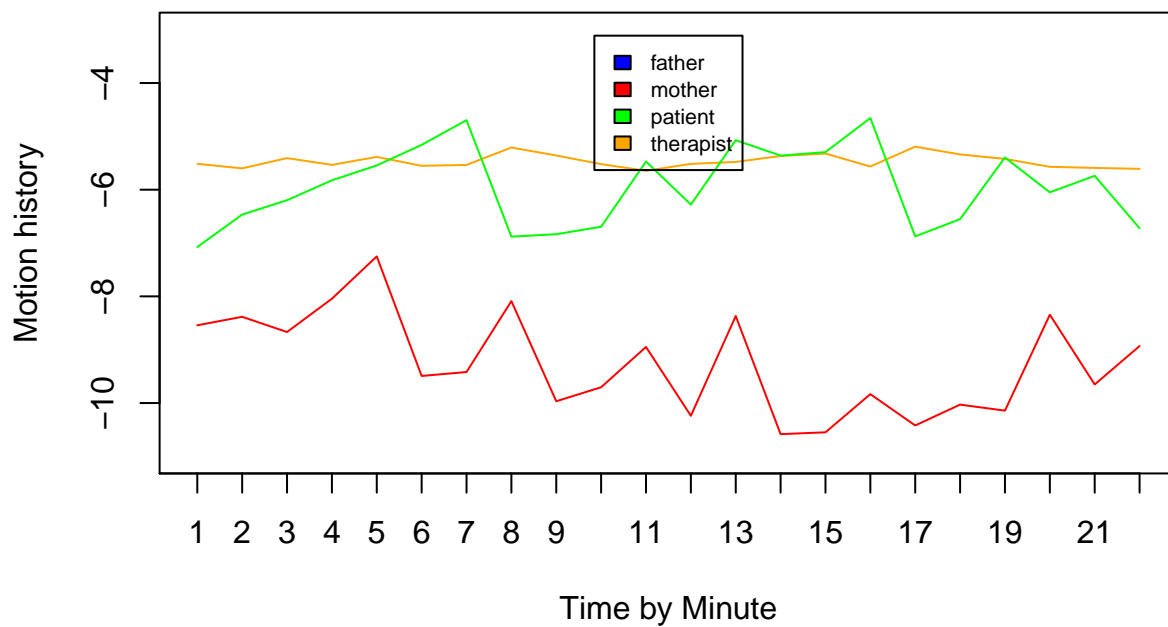
**Mean motion history (non overlapping minute intervals)  
on F1044M1 video**



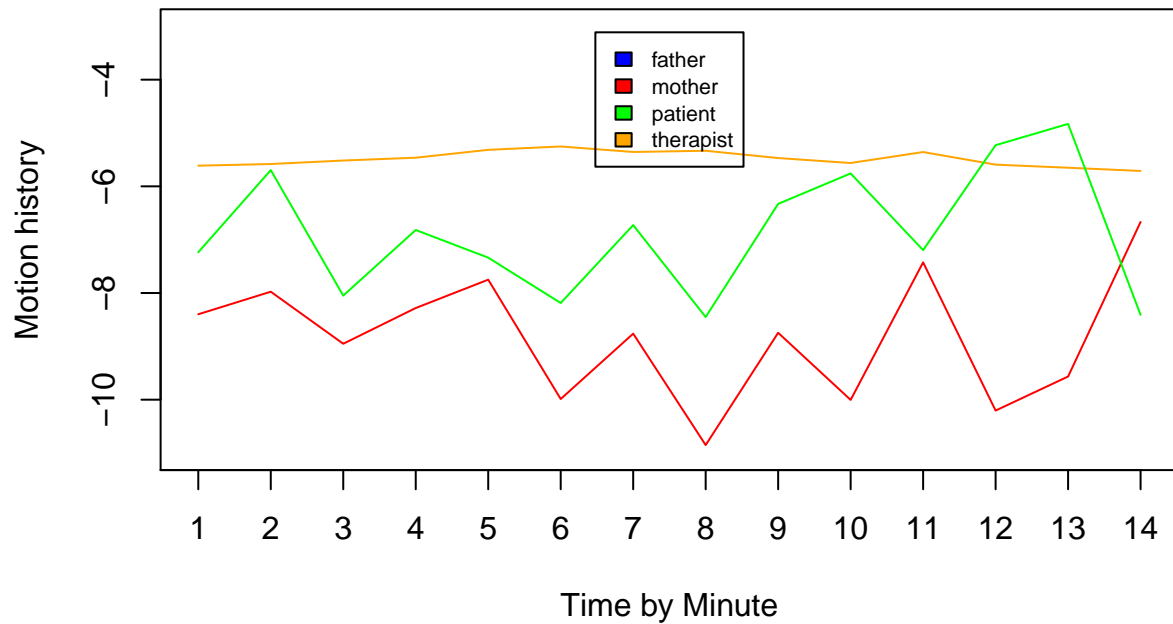
**Mean motion history (non overlapping minute intervals)  
on F1044M2 video**



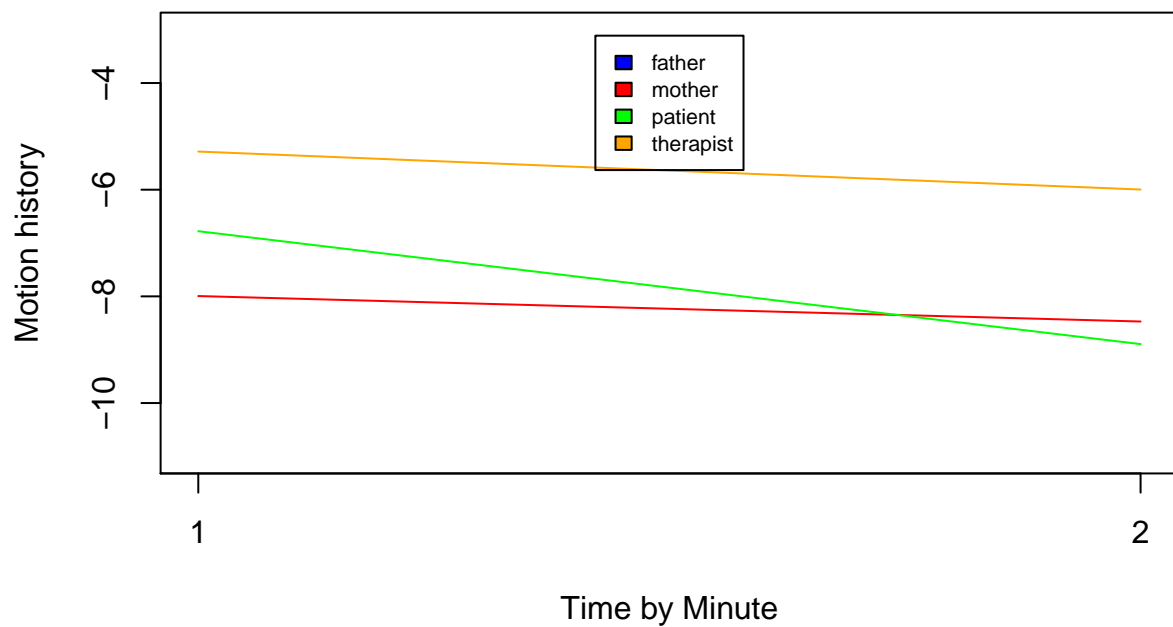
**Mean motion history (non overlapping minute intervals)  
on F1044N video**



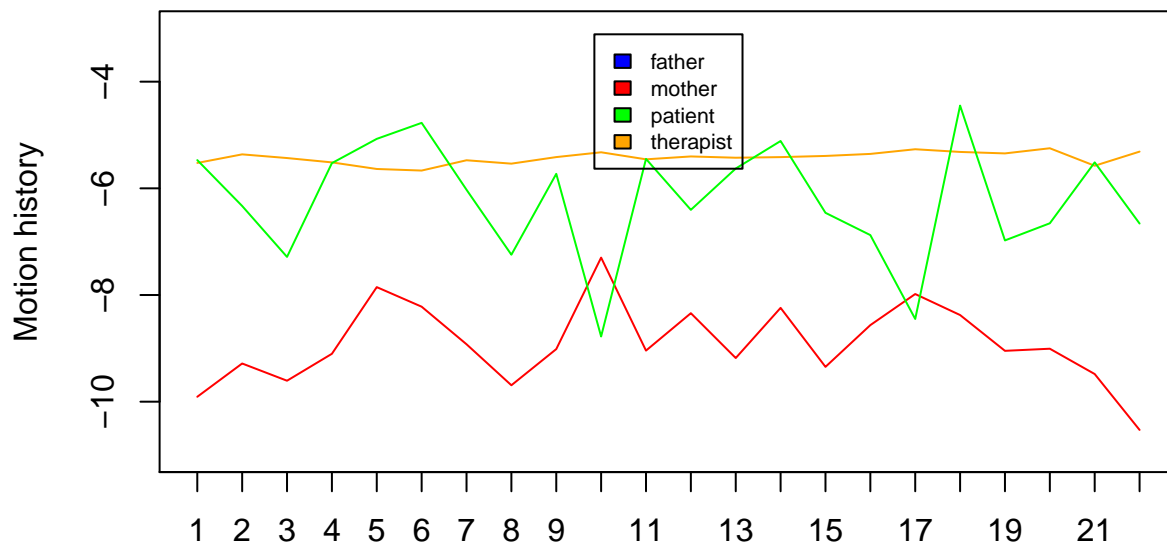
**Mean motion history (non overlapping minute intervals)  
on F1044O1 video**



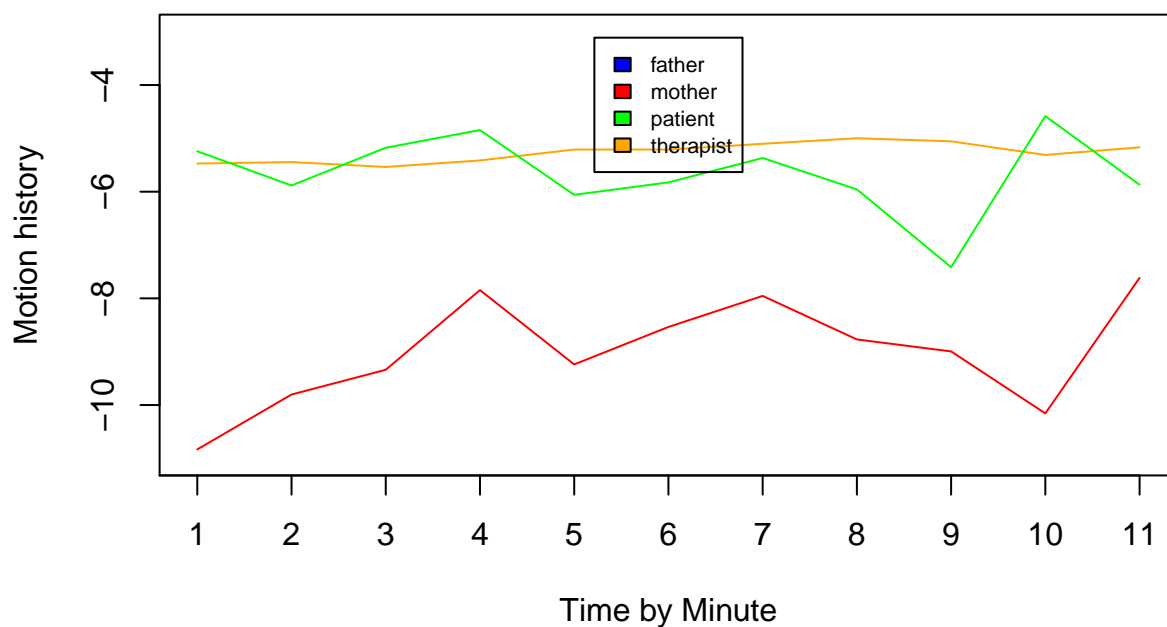
**Mean motion history (non overlapping minute intervals)  
on F1044O2 video**



**Mean motion history (non overlapping minute intervals)  
on F1044P video**

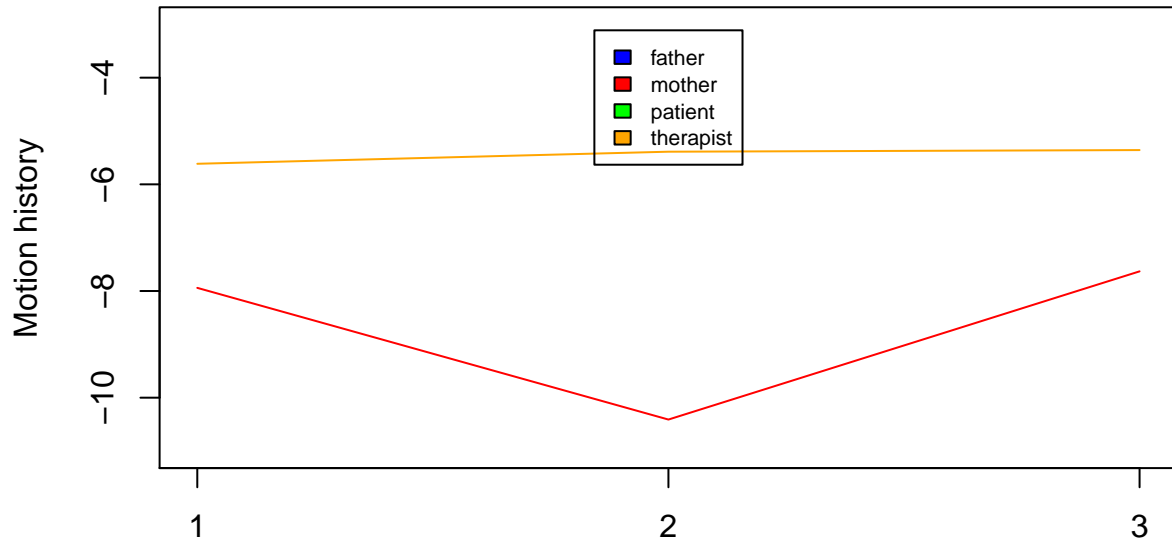


Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044Q1 video**

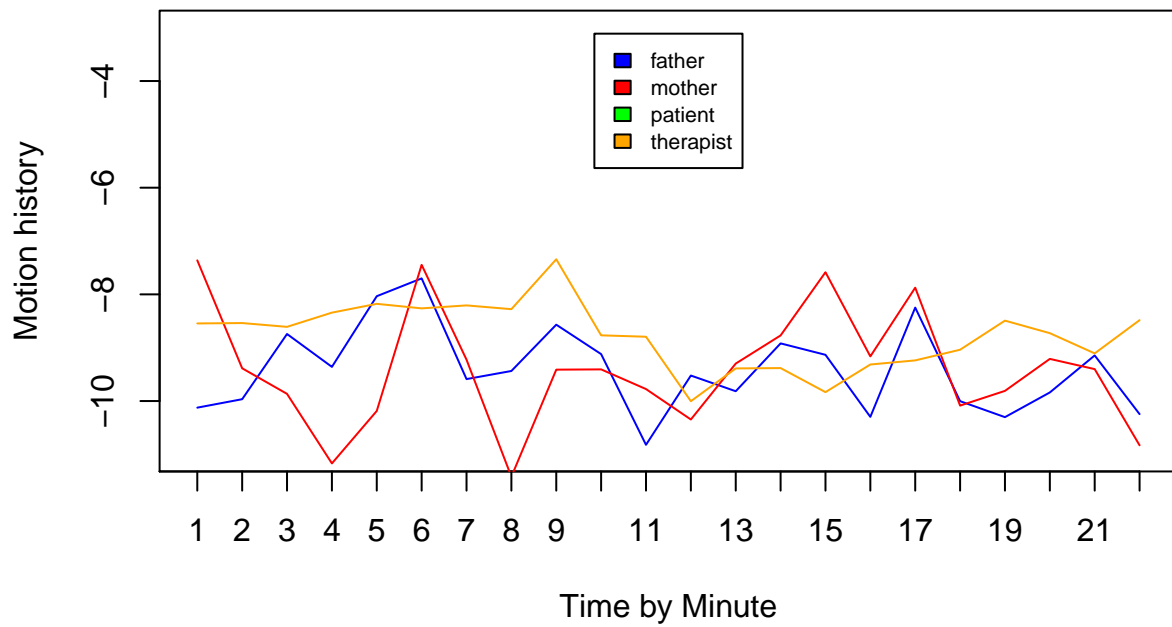




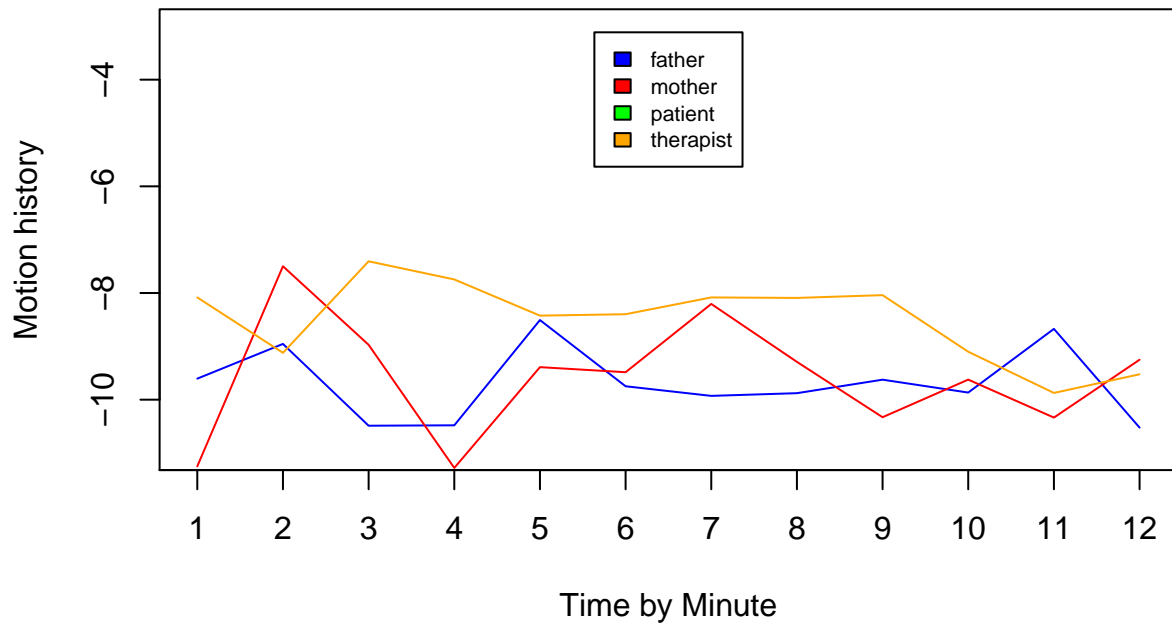
**Mean motion history (non overlapping minute intervals)  
on F1044Q2 video**



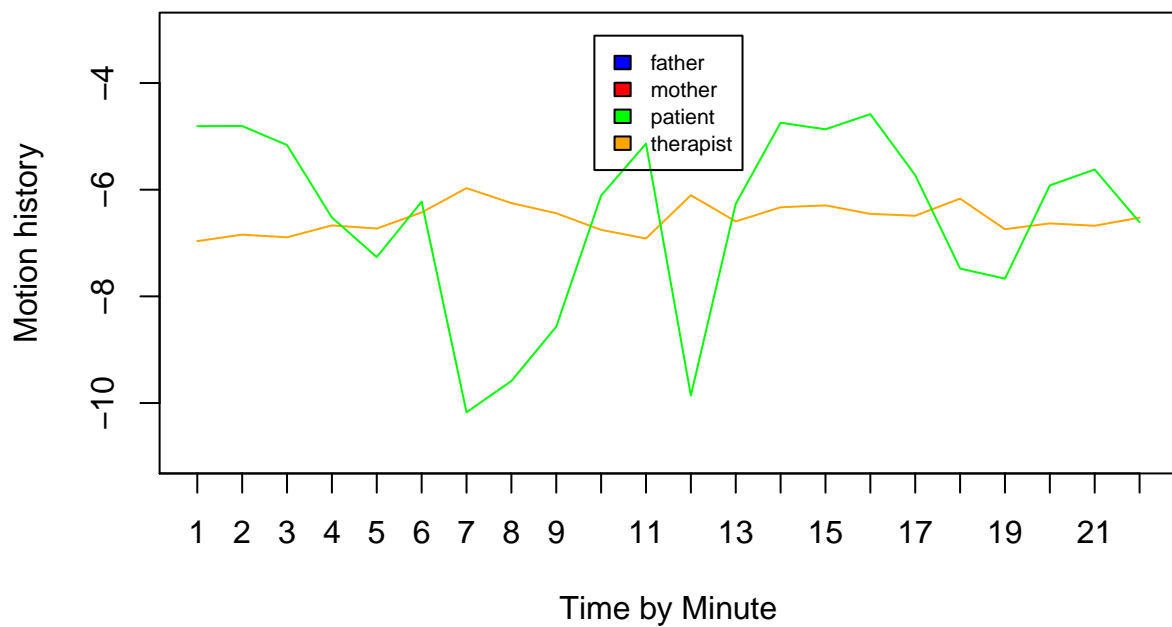
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044R1 video**



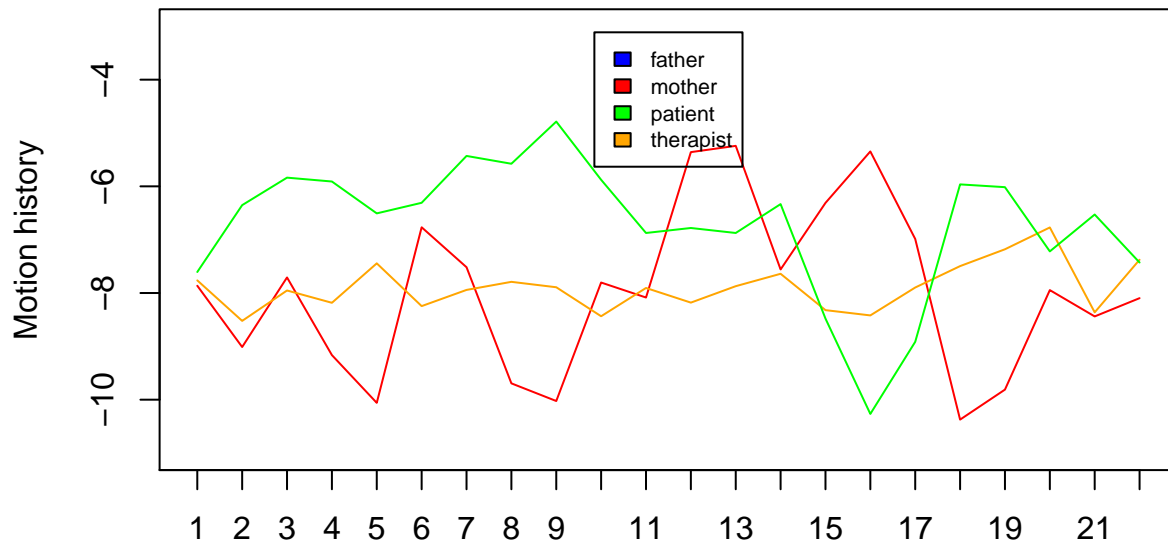
**Mean motion history (non overlapping minute intervals)  
on F1044R2 video**



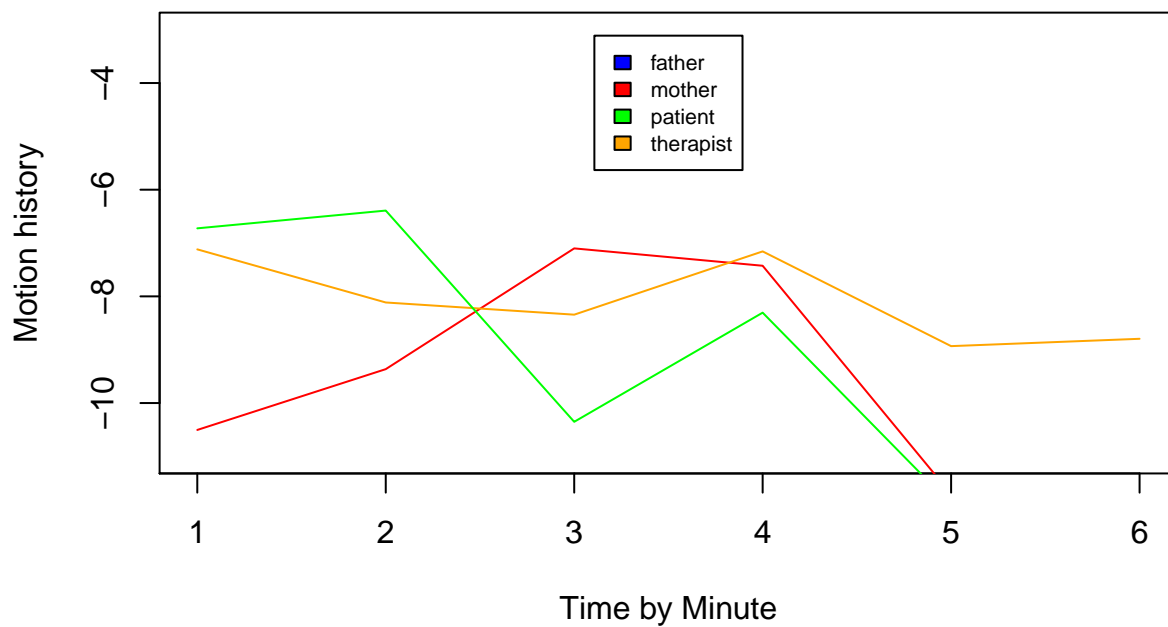
**Mean motion history (non overlapping minute intervals)  
on F1044A1 video**



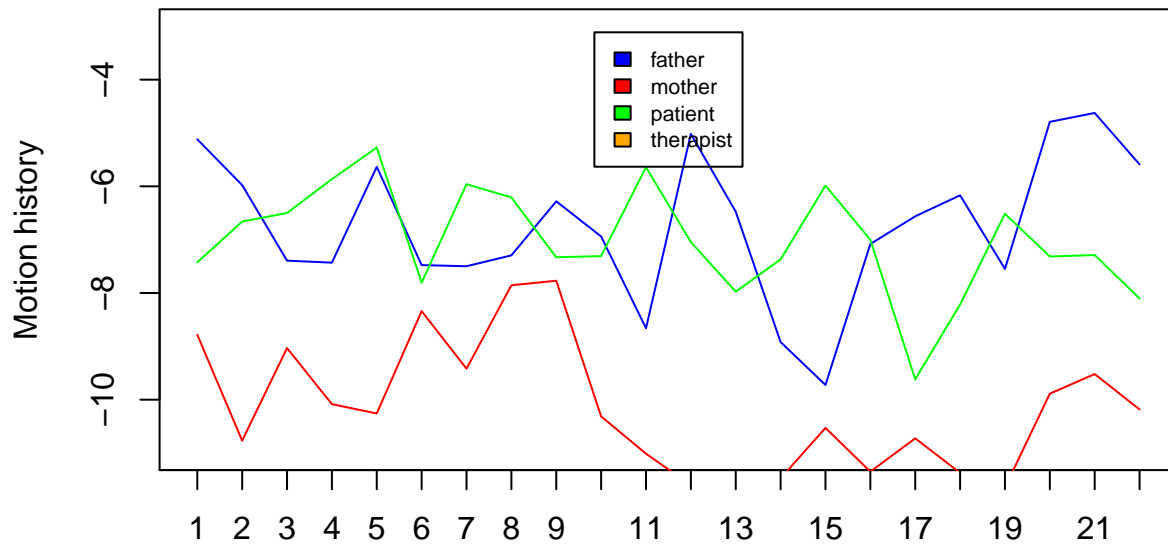
**Mean motion history (non overlapping minute intervals)  
on F1044B1 video**



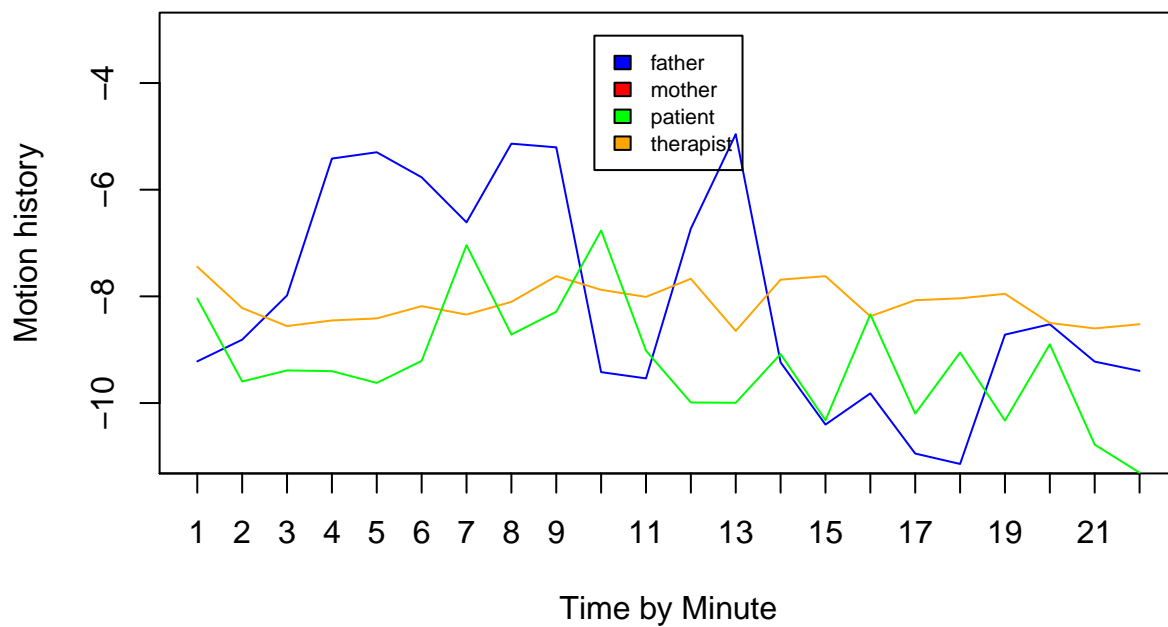
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044B2 video**



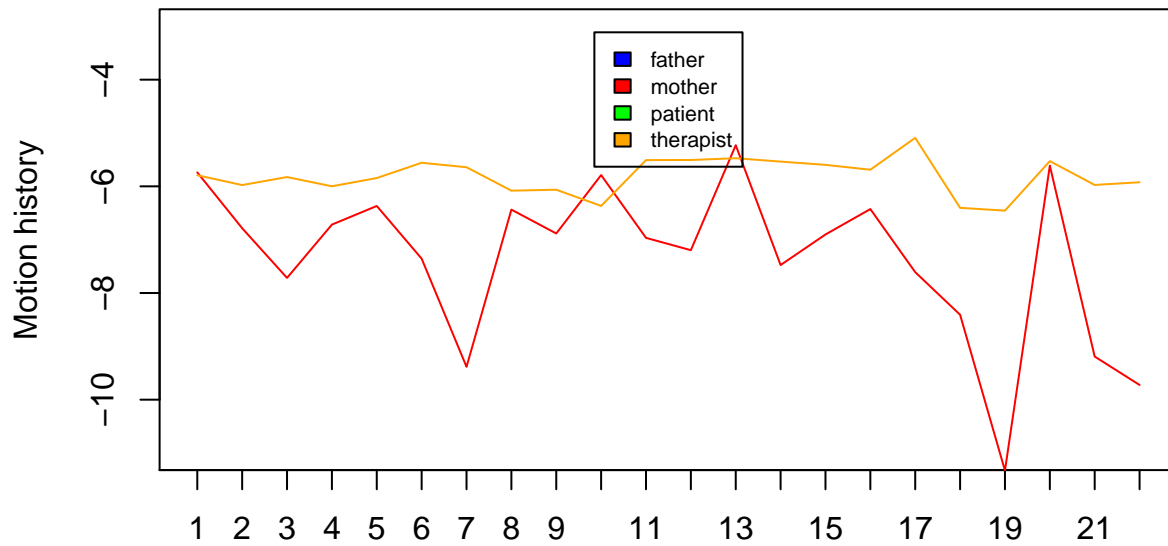
**Mean motion history (non overlapping minute intervals)  
on F1044C1 video**



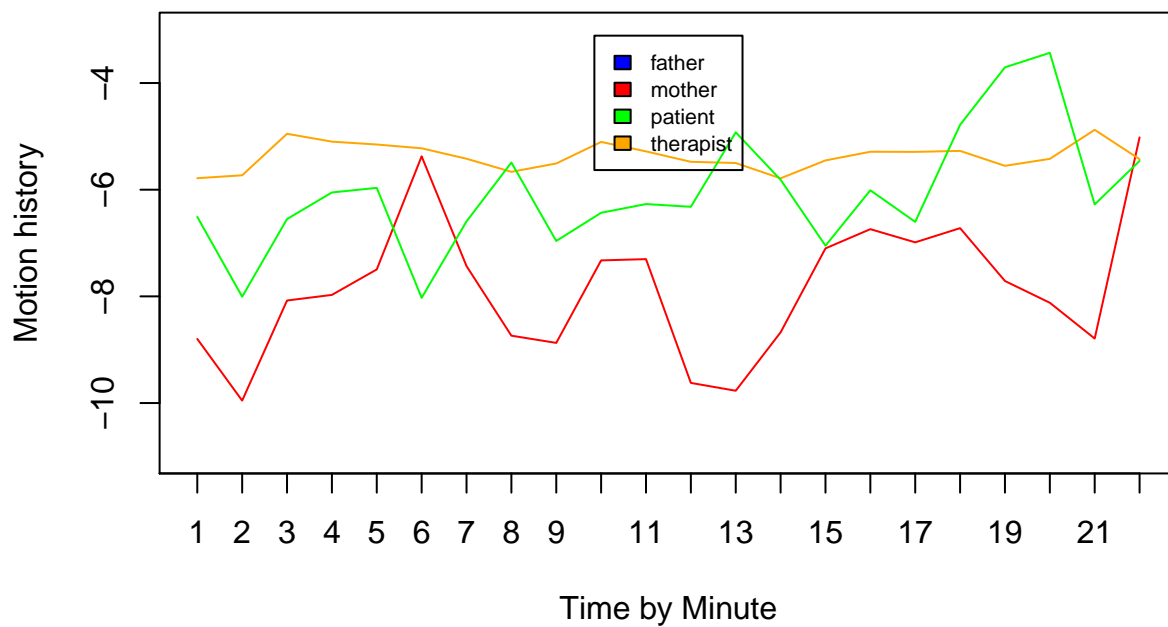
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044D2 video**



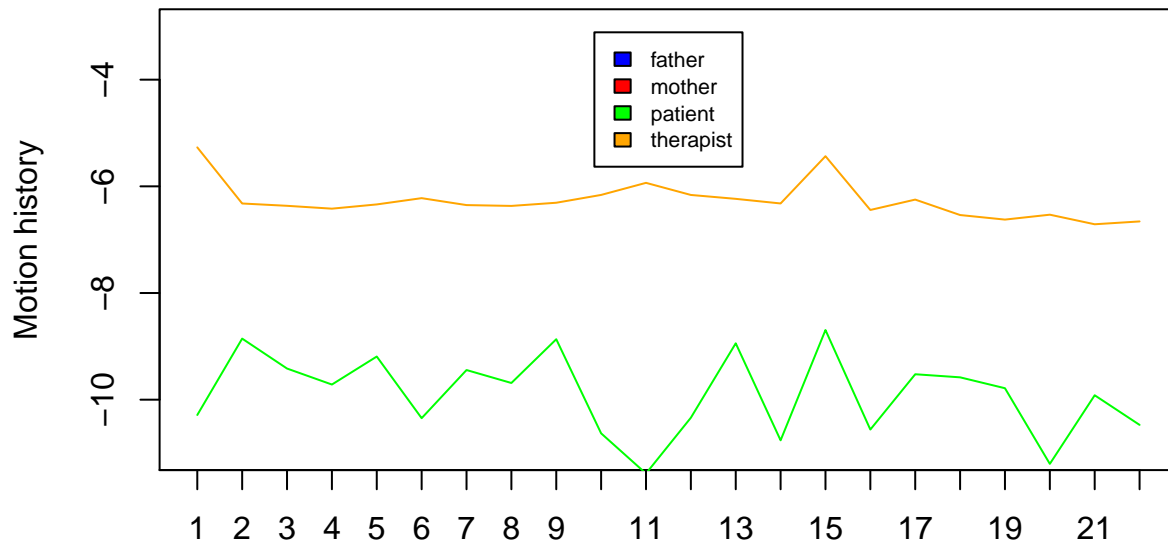
**Mean motion history (non overlapping minute intervals)  
on F1044A1 video**



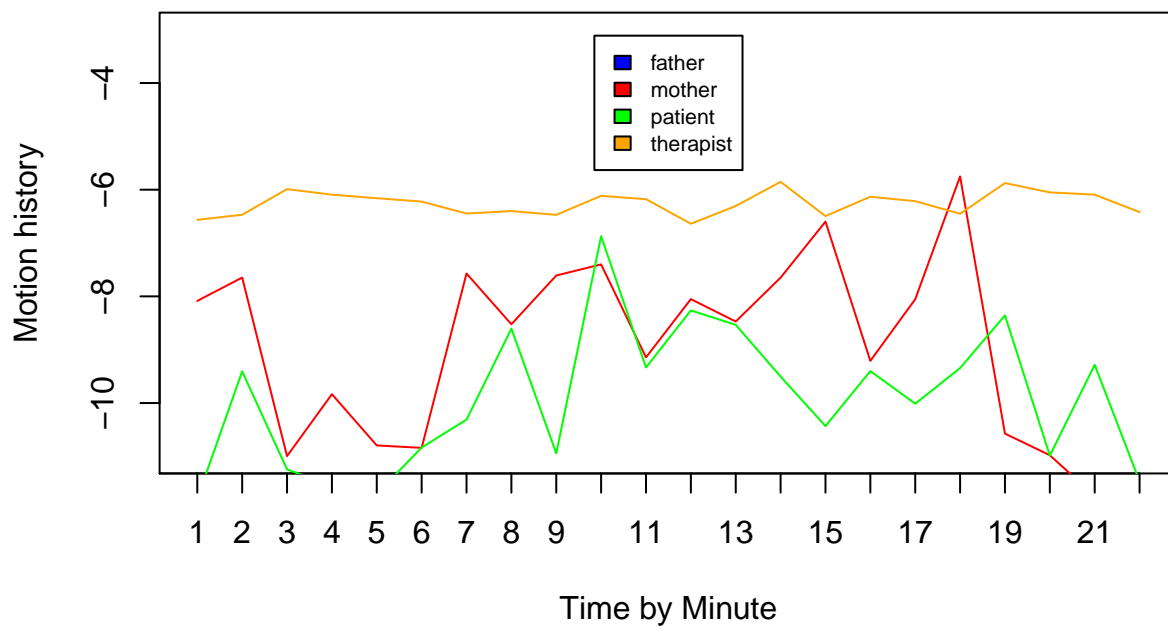
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044A2 video**



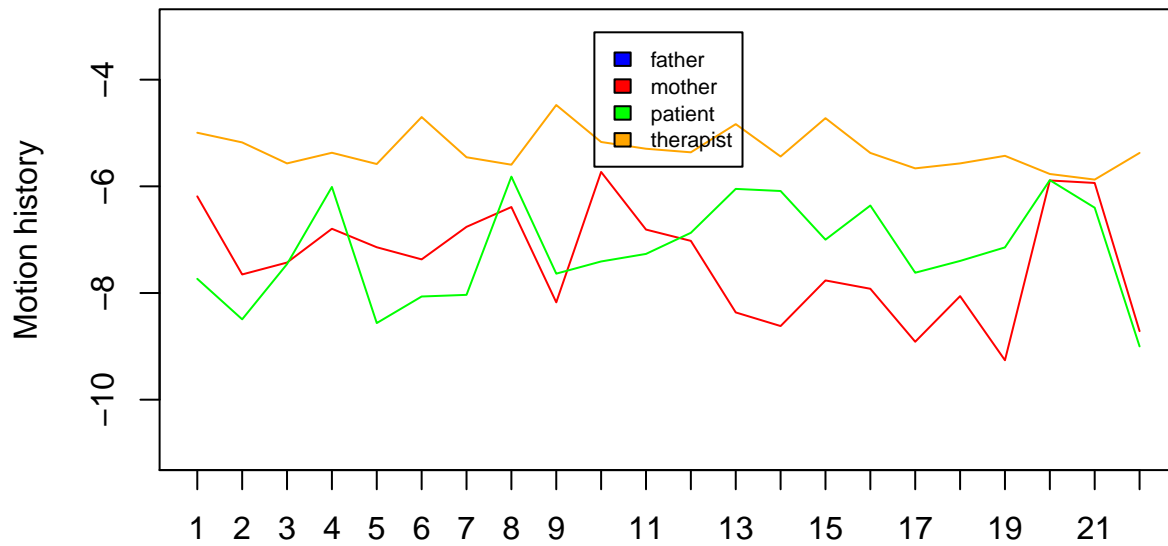
**Mean motion history (non overlapping minute intervals)  
on F1044B1 video**



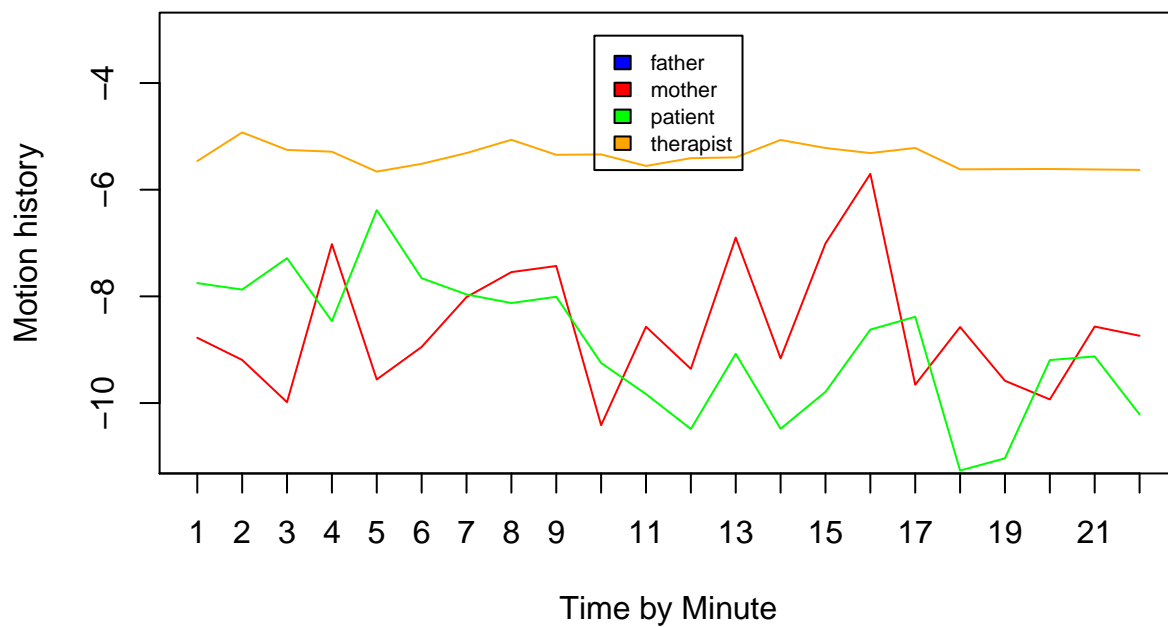
Time by Minute  
**Mean motion history (non overlapping minute intervals)  
on F1044B2 video**



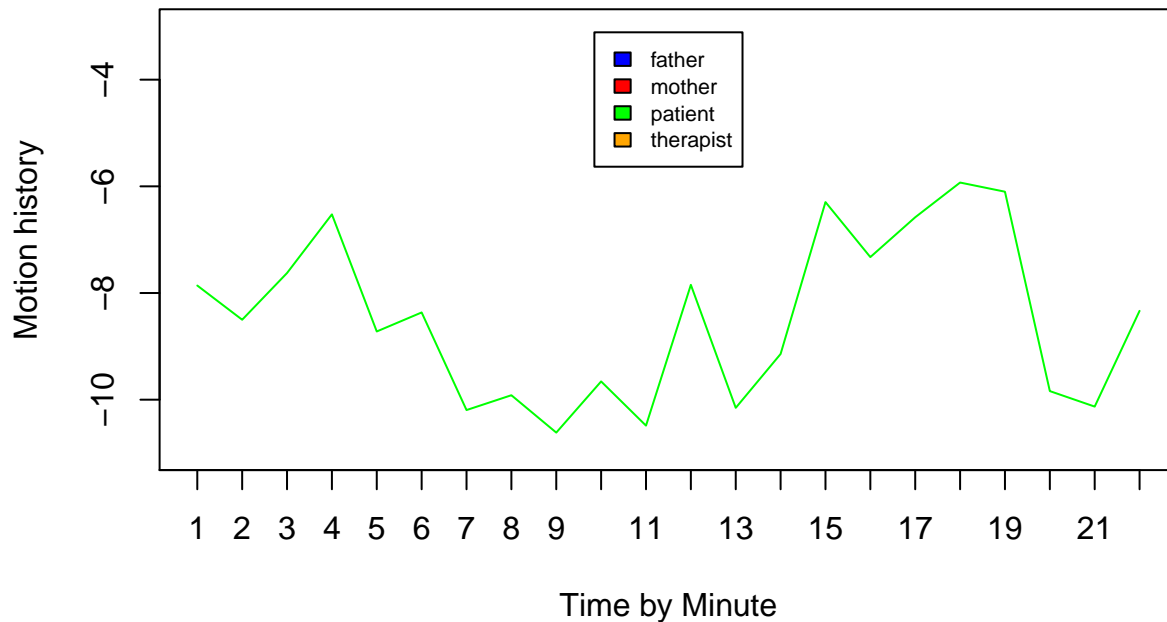
**Mean motion history (non overlapping minute intervals)  
on F1044A2 video**



**Mean motion history (non overlapping minute intervals)  
on F1044C2 video**



## Mean motion history (non overlapping minute intervals) on F1044C3 video



## Motion history by minute for the F1044C video

```

slidedFather <- SlidingInterval("father", 1 , 50, data)
slidedMother <- SlidingInterval("mother", 1 , 50, data)
slidedTherapist <- SlidingInterval("therapist", 1 , 50, data)
slidedPatient <- SlidingInterval("patient", 1 , 50, data)

framesByMinute <- 60*25
F1044C_Minutes <- ceiling(length(slidedFather)/framesByMinute)
for (i in 1:(F1044C_Minutes-1)){
  par(mar=c(4,4,4,2))
  borneInf <- i+framesByMinute*(i-1)
  borneSup <- i+i*framesByMinute
  slidedFatherMinute<-slidedFather[borneInf:borneSup]
  slidedMotherMinute<-slidedMother[borneInf:borneSup]
  slidedTherapistMinute<-slidedTherapist[borneInf:borneSup]
  slidedPatientMinute<-slidedPatient[borneInf:borneSup]
  slidedVideoDF <- data.frame(slidedFatherMinute, slidedMotherMinute, slidedTherapistMinute, slidedPatientMinute)
  str (slidedVideoDF)
  plot (slidedVideoDF$minute, slidedVideoDF$slidedMotherMinute, type="l", col="red",
  main=paste("Motion history with Sliding interval function during
  minute ", i, " in F1044C video", sep=""),
  ylab="Motion history", xlab="Number of frame", ylim=c(0, 12E-03))
  # xaxp=c(0, length(slidedFatherMinute), length(slidedFatherMinute)))
  lines(slidedVideoDF$minute, slidedVideoDF$slidedFatherMinute, col="blue")
  lines(slidedVideoDF$minute, slidedVideoDF$slidedTherapistMinute, col="orange")
  lines(slidedVideoDF$minute, slidedVideoDF$slidedPatientMinute, col="green")
  legend("top", inset=.05, ParticipantsList,

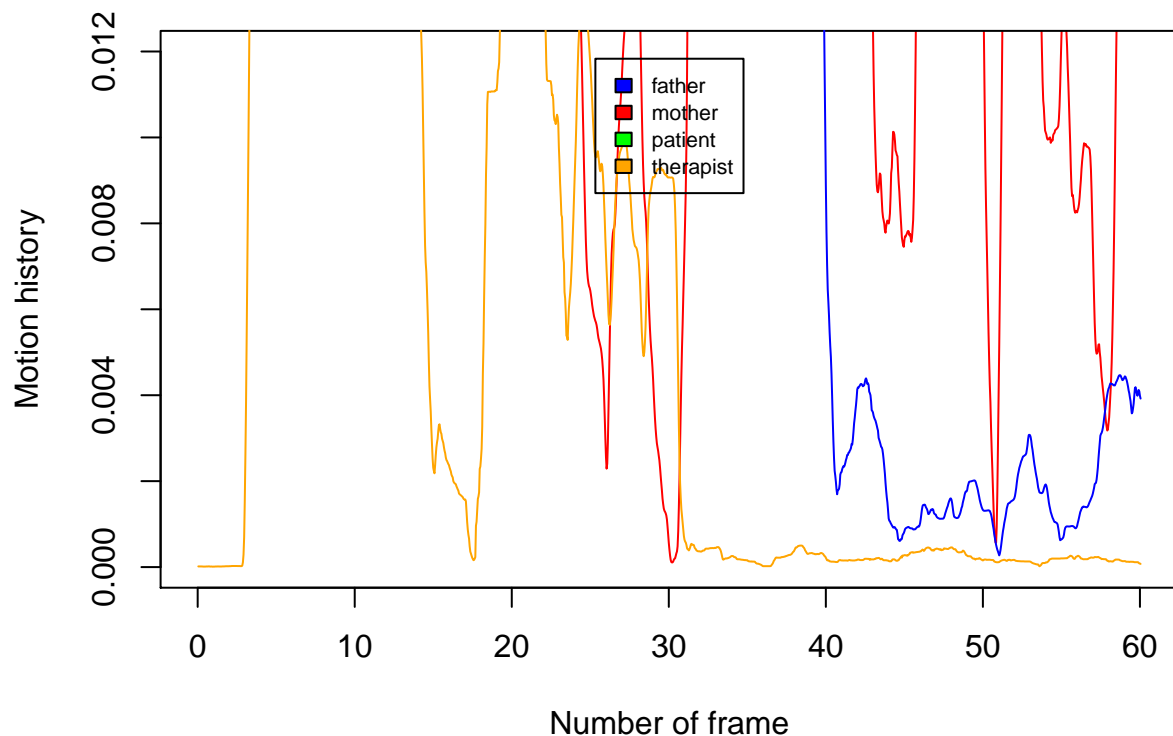
```



```
fill=colOrderList, cex=0.7)}]
```

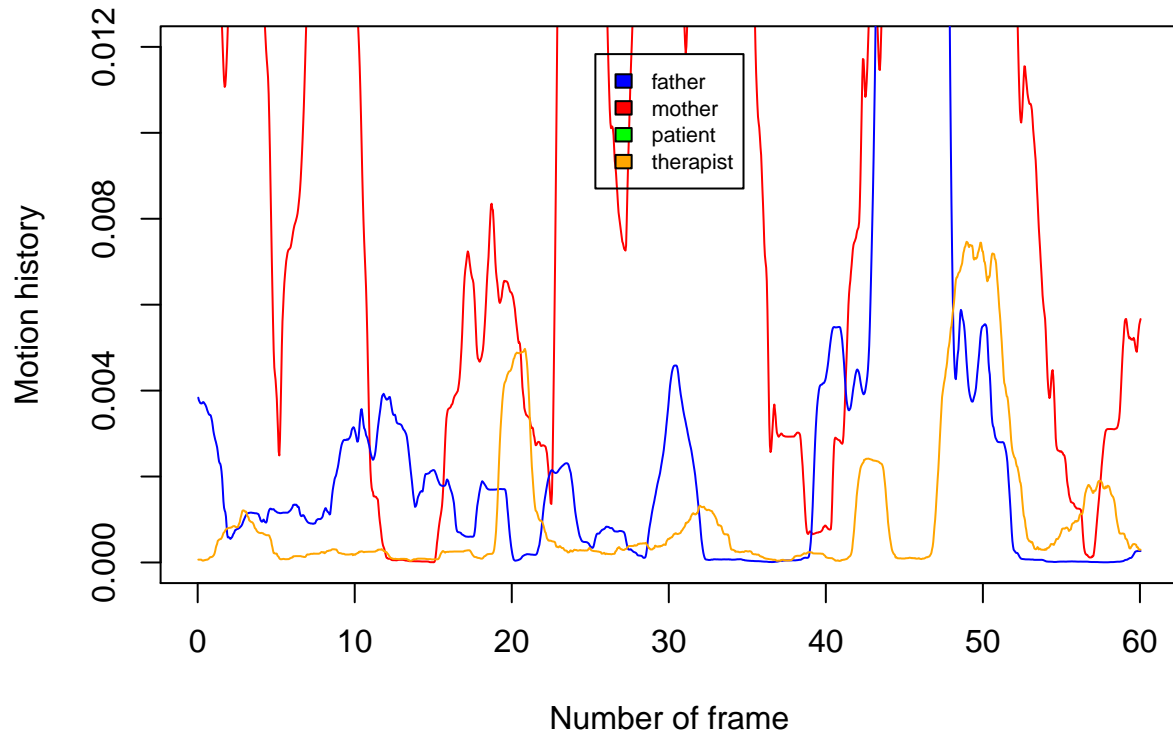
```
## 'data.frame': 1501 obs. of 6 variables:
## $ slidedFatherMinute : num 0.0239 0.0249 0.0258 0.0268 0.0277 ...
## $ slidedMotherMinute : num 0.0431 0.0418 0.0406 0.0397 0.039 ...
## $ slidedTherapistMinute: num 1.61e-05 1.53e-05 1.53e-05 1.53e-05 1.44e-05 ...
## $ slidedPatientMinute : num NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames : int 1 2 3 4 5 6 7 8 9 10 ...
## $ minute : num 0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

### Motion history with Sliding interval function during minute 1 in F1044C video



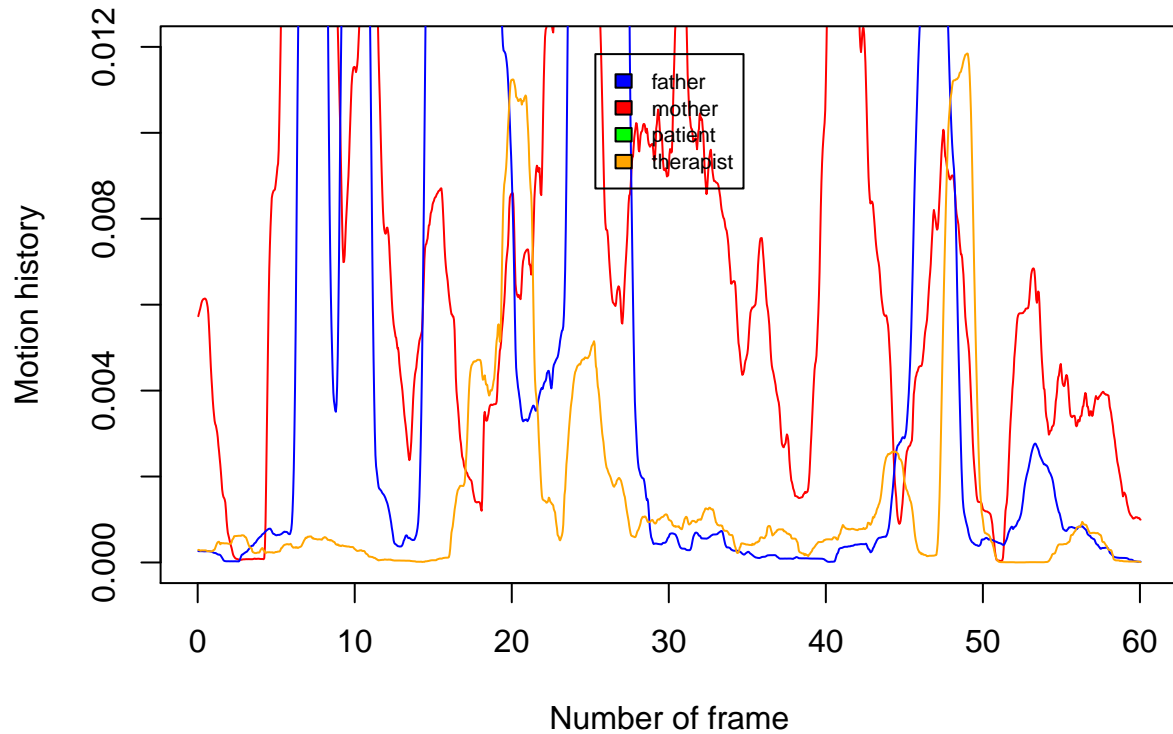
```
## 'data.frame': 1501 obs. of 6 variables:
## $ slidedFatherMinute : num 0.00384 0.00377 0.00374 0.00371 0.0037 ...
## $ slidedMotherMinute : num 0.0359 0.0356 0.0352 0.0346 0.0339 ...
## $ slidedTherapistMinute: num 6.62e-05 6.54e-05 6.54e-05 6.54e-05 6.54e-05 ...
## $ slidedPatientMinute : num NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames : int 1 2 3 4 5 6 7 8 9 10 ...
## $ minute : num 0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 2 in F1044C video



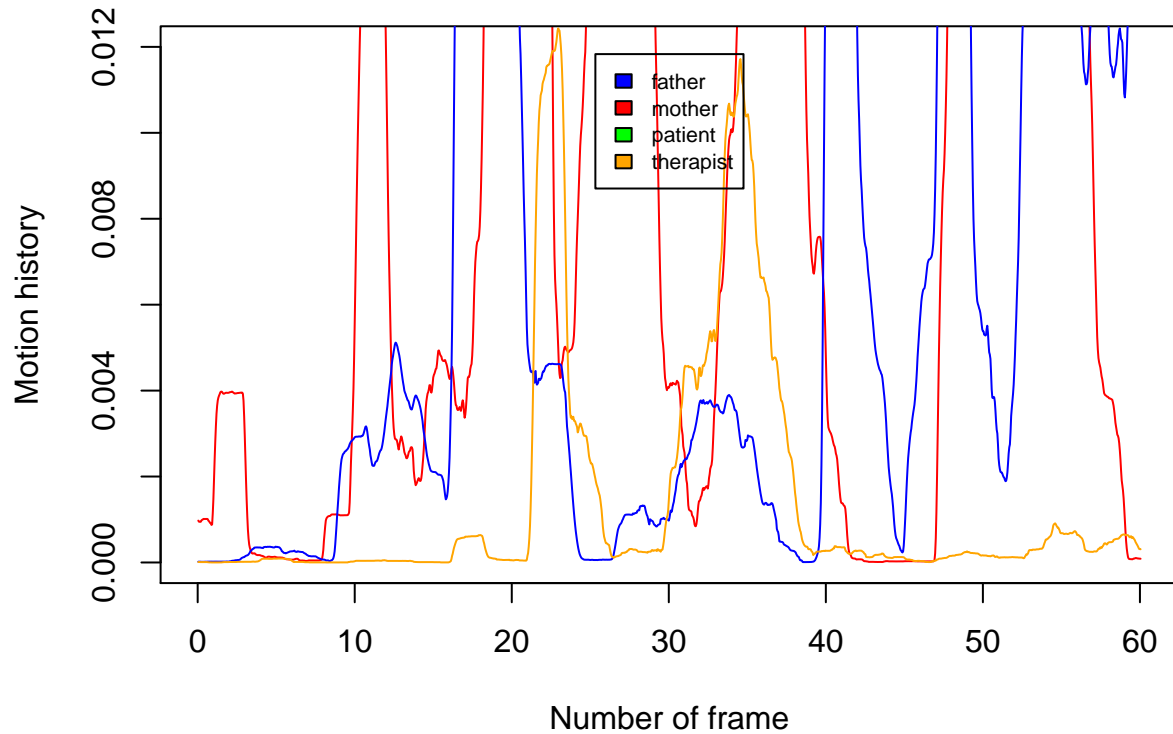
```
## 'data.frame':   1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  0.000265 0.000264 0.000263 0.000263 0.000263 ...
## $ slidedMotherMinute   : num  0.00573 0.00579 0.00585 0.0059 0.00597 ...
## $ slidedTherapistMinute: num  0.000294 0.000291 0.000288 0.000278 0.000274 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames                : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute                : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 3 in F1044C video



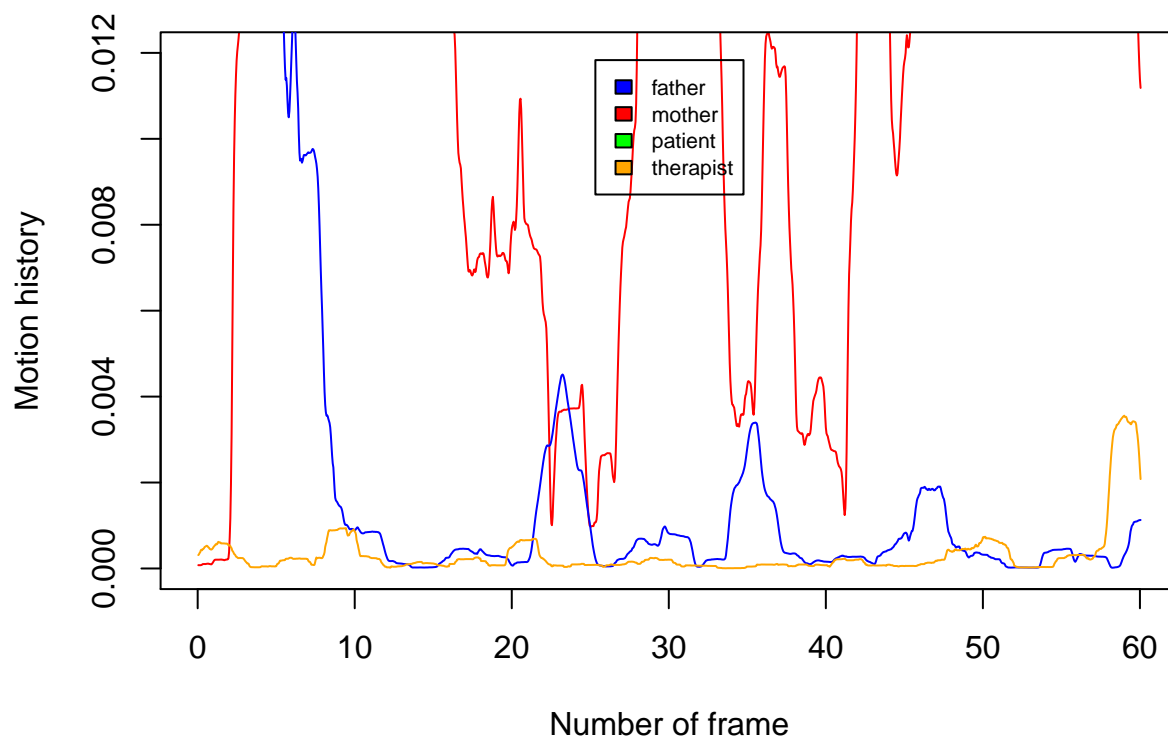
```
## 'data.frame': 1501 obs. of 6 variables:
## $ slidedFatherMinute : num 1.77e-05 1.71e-05 1.77e-05 1.77e-05 1.64e-05 ...
## $ slidedMotherMinute : num 0.000975 0.00095 0.000947 0.000947 0.000947 ...
## $ slidedTherapistMinute: num 1.02e-05 1.02e-05 1.02e-05 9.34e-06 9.34e-06 ...
## $ slidedPatientMinute : num NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames : int 1 2 3 4 5 6 7 8 9 10 ...
## $ minute : num 0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 4 in F1044C video



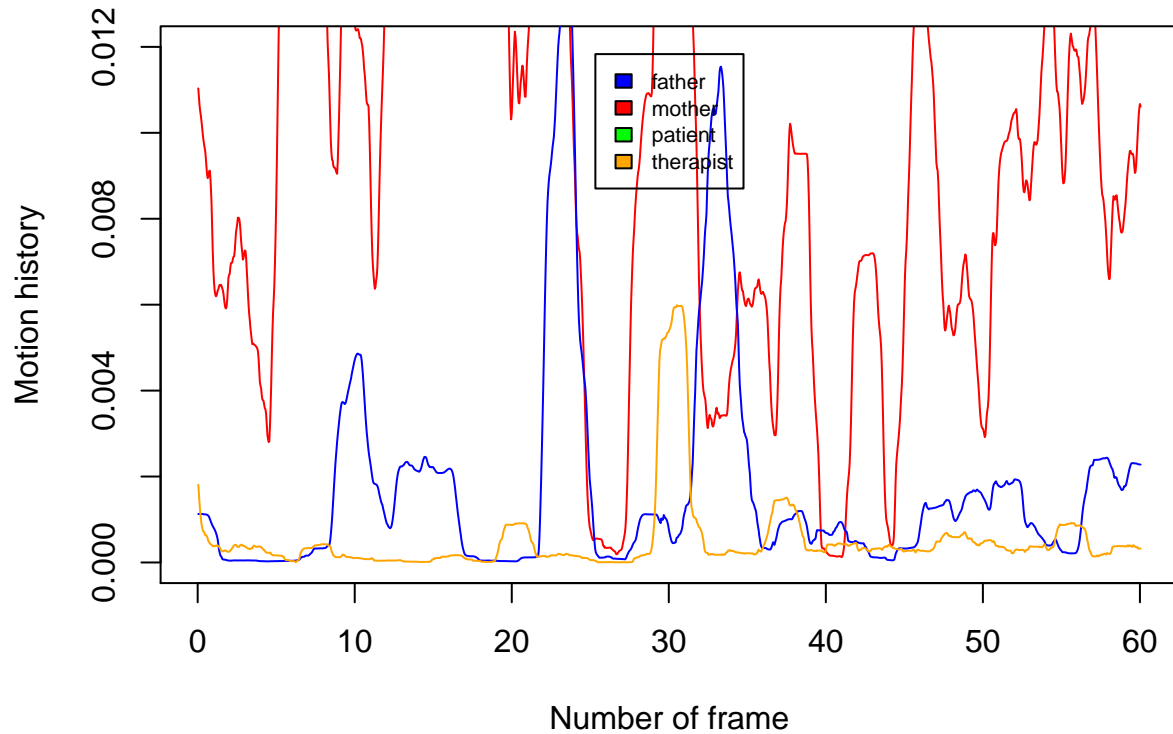
```
## 'data.frame':   1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  0.0173 0.0177 0.0181 0.0183 0.0185 ...
## $ slidedMotherMinute   : num  7.71e-05 7.67e-05 7.63e-05 7.60e-05 7.71e-05 ...
## $ slidedTherapistMinute: num  0.000311 0.000327 0.000358 0.000377 0.000424 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute                : num  0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 5 in F1044C video



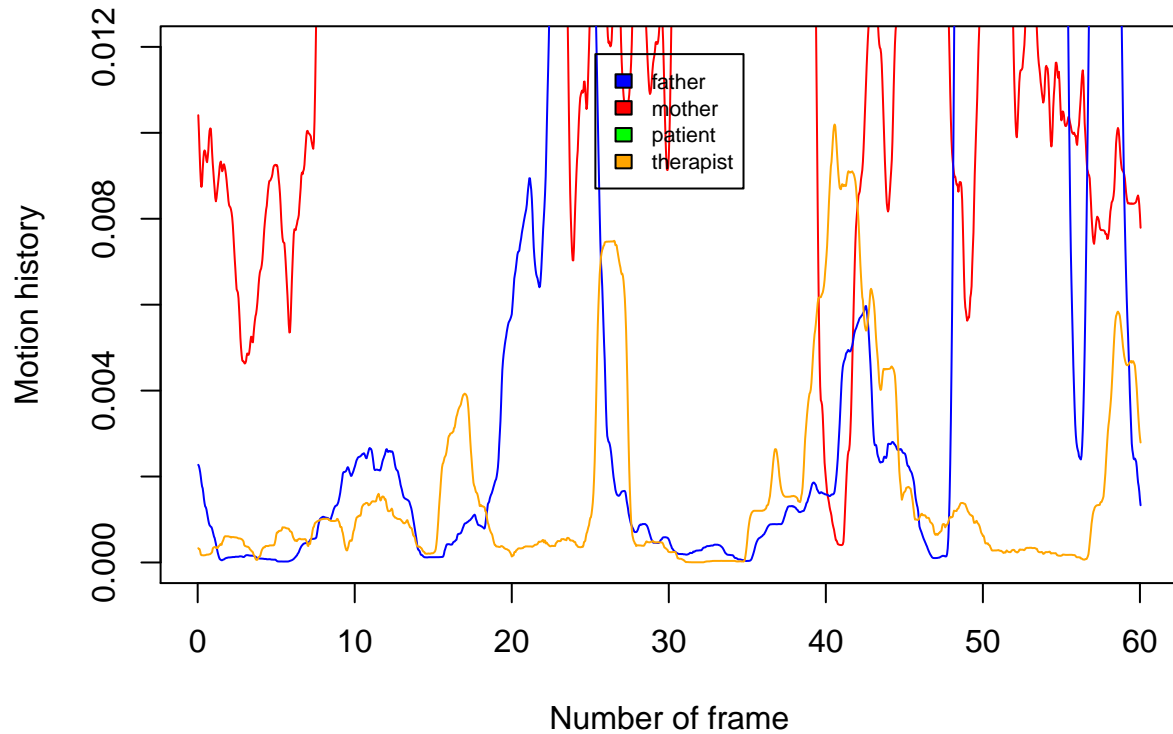
```
## 'data.frame':   1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  0.00113 0.00113 0.00113 0.00113 0.00113 ...
## $ slidedMotherMinute   : num  0.011 0.0108 0.0106 0.0105 0.0104 ...
## $ slidedTherapistMinute: num  0.00181 0.00159 0.00137 0.00119 0.00104 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute                : num  0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 6 in F1044C video



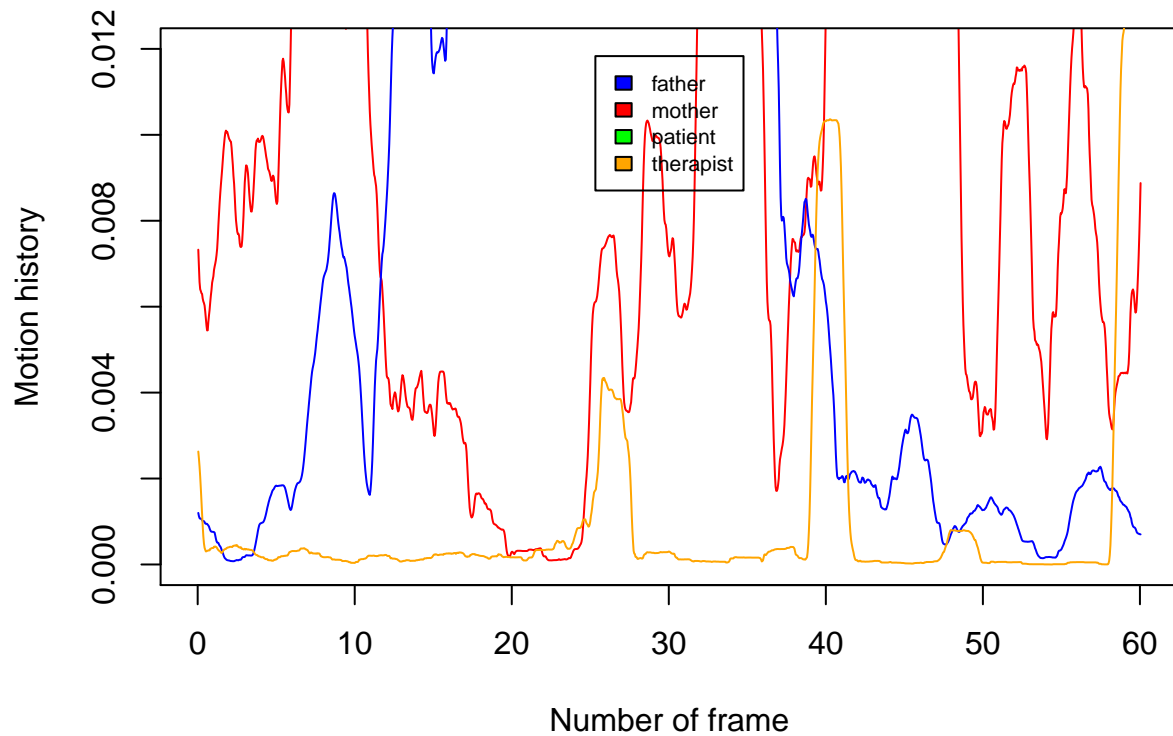
```
## 'data.frame':  1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  0.00227 0.00224 0.00218 0.0021 0.00199 ...
## $ slidedMotherMinute   : num  0.01041 0.01002 0.00959 0.00918 0.00887 ...
## $ slidedTherapistMinute: num  0.000333 0.000315 0.000312 0.000258 0.000185 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute               : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 7 in F1044C video



```
## 'data.frame':   1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  0.0012 0.00112 0.00108 0.00107 0.00107 ...
## $ slidedMotherMinute   : num  0.00733 0.00693 0.00657 0.00641 0.00636 ...
## $ slidedTherapistMinute: num  0.00263 0.00247 0.00228 0.00205 0.00177 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute               : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

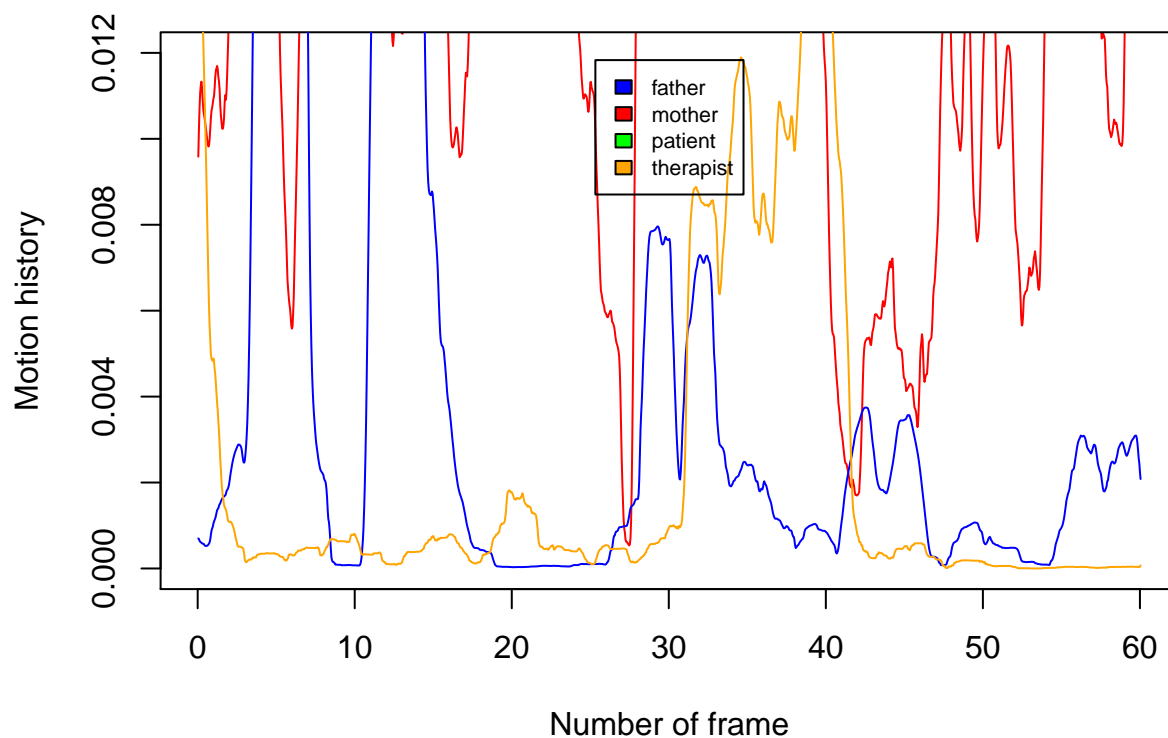
## Motion history with Sliding interval function during minute 8 in F1044C video



```
## 'data.frame':   1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  0.000699 0.000686 0.000645 0.000622 0.000607 ...
## $ slidedMotherMinute   : num  0.00959 0.01022 0.01071 0.01107 0.01125 ...
## $ slidedTherapistMinute: num  0.0163 0.0161 0.0157 0.0153 0.0148 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute               : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

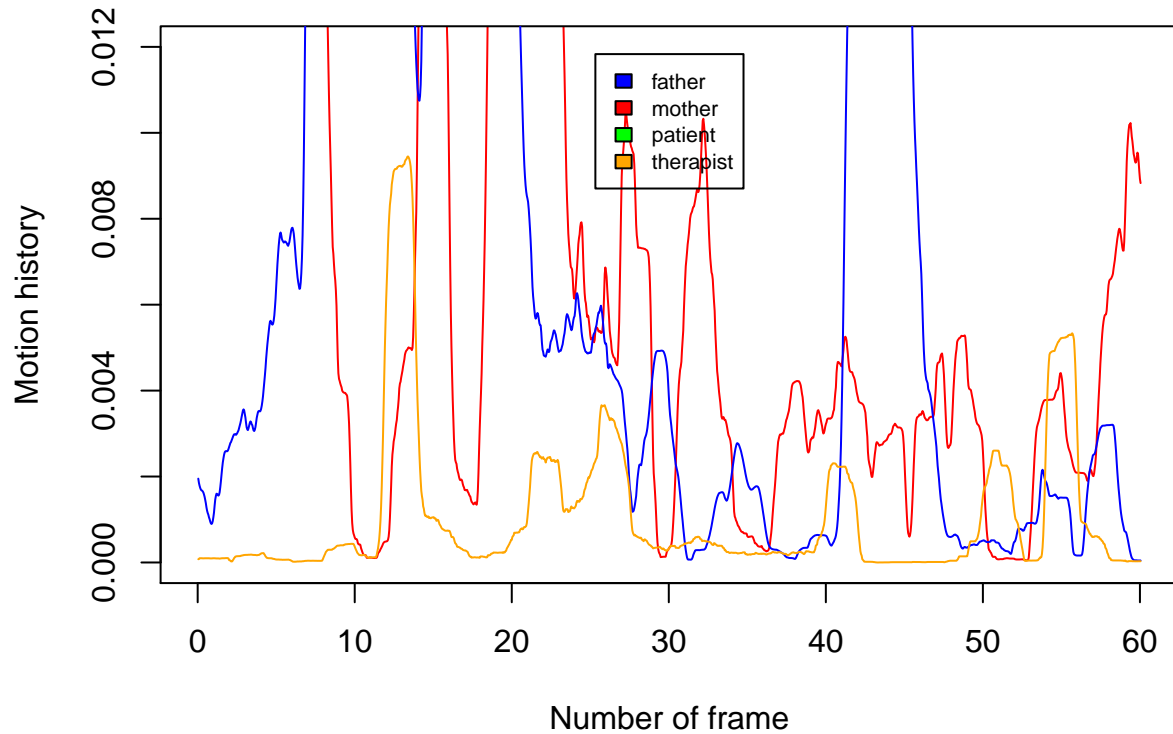


## Motion history with Sliding interval function during minute 9 in F1044C video



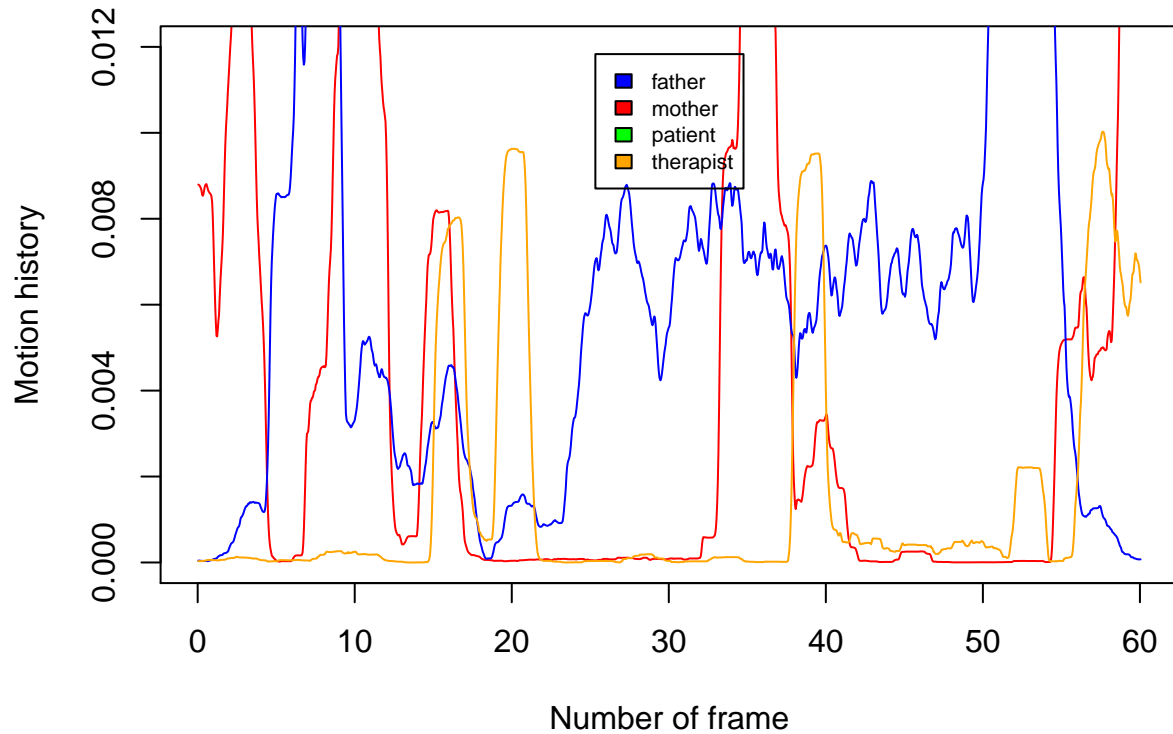
```
## 'data.frame':   1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  0.00195 0.00185 0.00179 0.00174 0.0017 ...
## $ slidedMotherMinute   : num  0.0298 0.0299 0.0302 0.0307 0.0315 ...
## $ slidedTherapistMinute: num  7.55e-05 8.06e-05 9.93e-05 9.85e-05 9.85e-05 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute               : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 10 in F1044C video



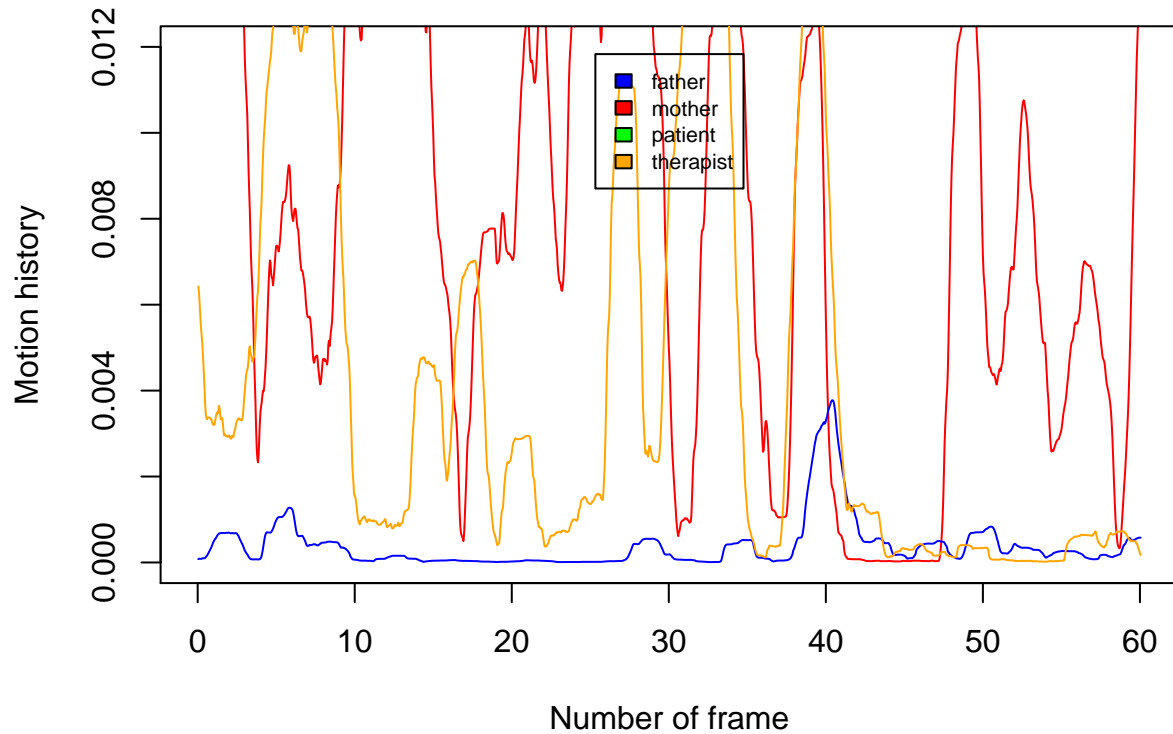
```
## 'data.frame':  1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  4.44e-05 4.27e-05 4.24e-05 4.11e-05 3.94e-05 ...
## $ slidedMotherMinute   : num  0.0088 0.00878 0.00877 0.00874 0.00871 ...
## $ slidedTherapistMinute: num  2.97e-05 3.14e-05 3.23e-05 3.23e-05 3.23e-05 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute               : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 11 in F1044C video



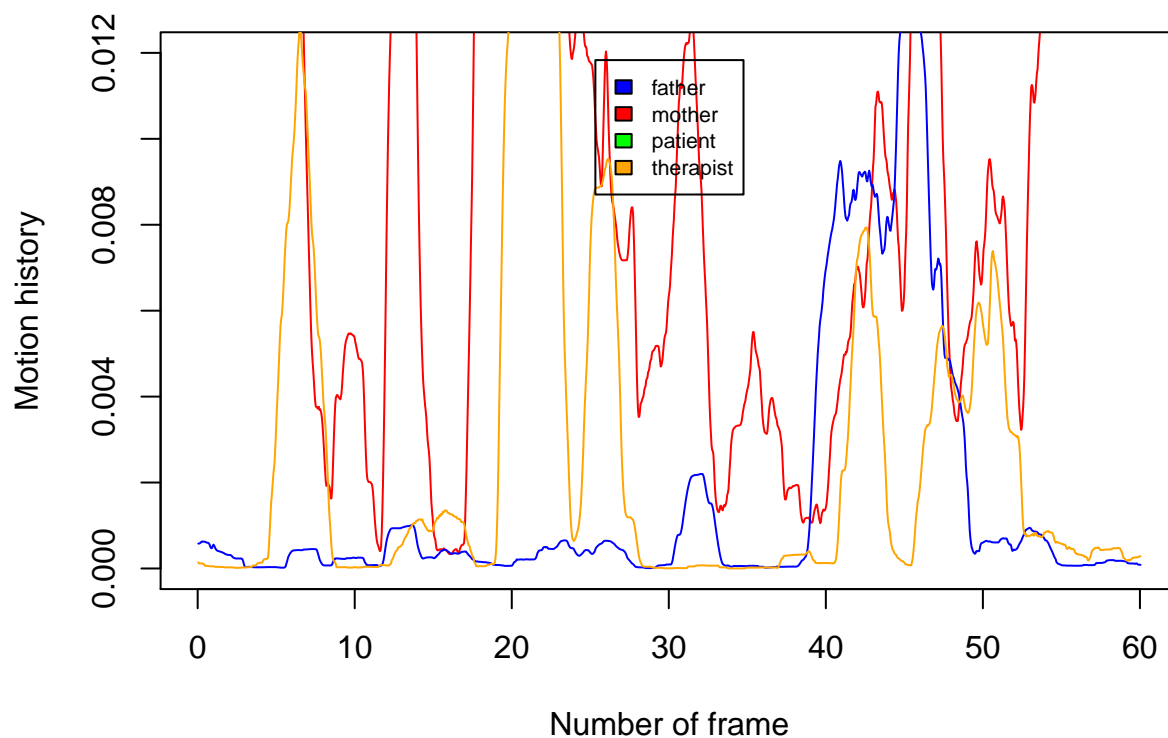
```
## 'data.frame':   1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  7.85e-05 8.81e-05 8.77e-05 8.54e-05 8.41e-05 ...
## $ slidedMotherMinute   : num  0.0578 0.0588 0.0599 0.0608 0.0616 ...
## $ slidedTherapistMinute: num  0.00642 0.00631 0.00607 0.0058 0.00553 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1 2 3 4 5 6 7 8 9 10 ...
## $ minute               : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 12 in F1044C video



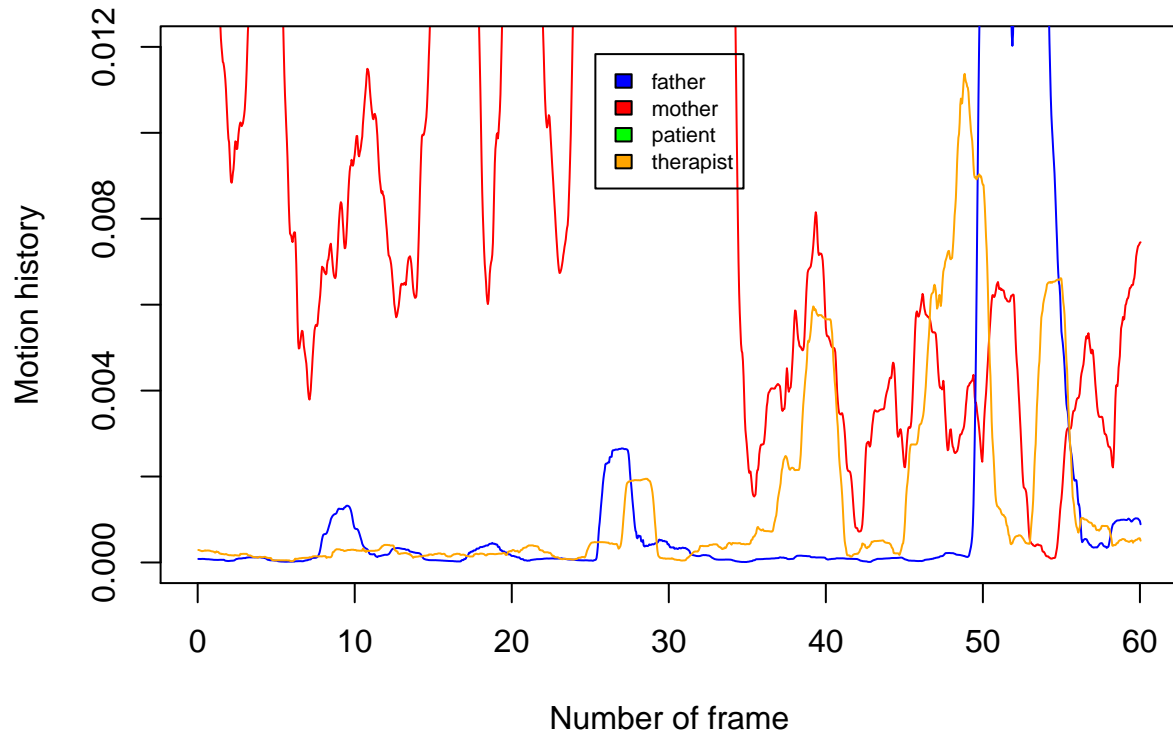
```
## 'data.frame': 1501 obs. of 6 variables:
## $ slidedFatherMinute : num 0.000579 0.00058 0.000584 0.000591 0.000606 ...
## $ slidedMotherMinute : num 0.0136 0.0138 0.0141 0.0144 0.0147 ...
## $ slidedTherapistMinute: num 0.000142 0.00012 0.000118 0.000116 0.000117 ...
## $ slidedPatientMinute : num NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames : int 1 2 3 4 5 6 7 8 9 10 ...
## $ minute : num 0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 13 in F1044C video



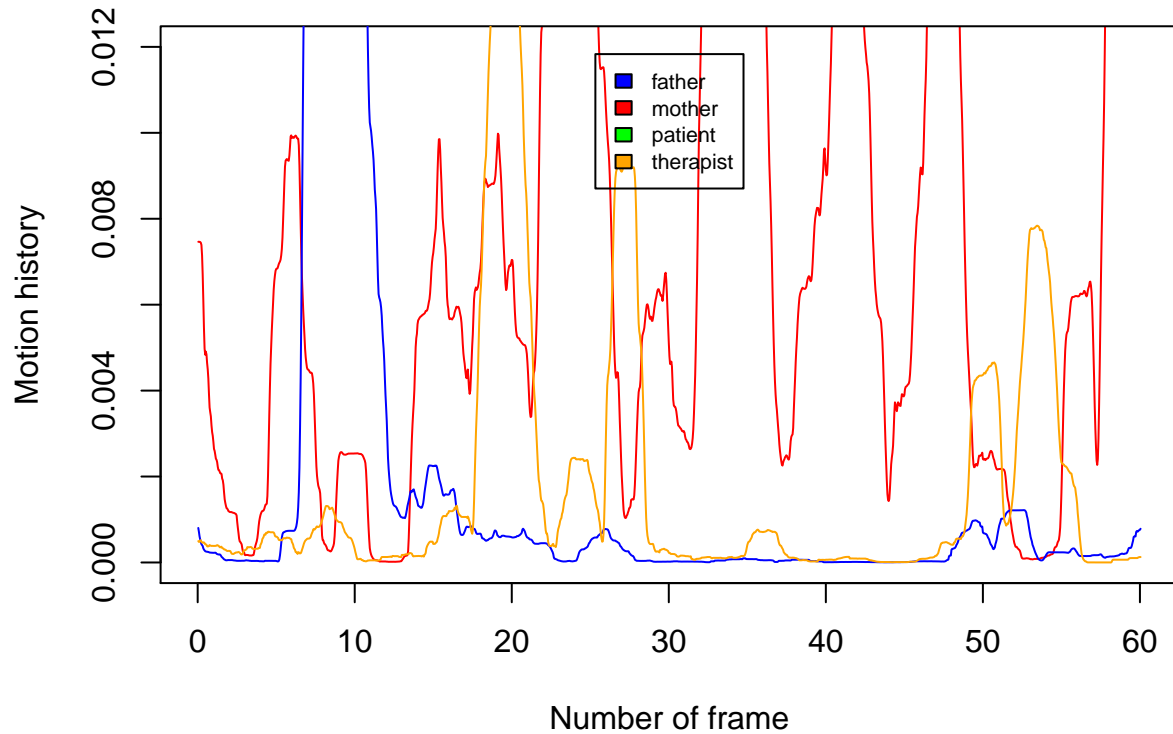
```
## 'data.frame': 1501 obs. of 6 variables:
## $ slidedFatherMinute : num 8.21e-05 8.18e-05 8.15e-05 8.02e-05 8.18e-05 ...
## $ slidedMotherMinute : num 0.0373 0.0367 0.0361 0.0356 0.0349 ...
## $ slidedTherapistMinute: num 0.00029 0.00029 0.000283 0.000278 0.000272 ...
## $ slidedPatientMinute : num NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames : int 1 2 3 4 5 6 7 8 9 10 ...
## $ minute : num 0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 14 in F1044C video



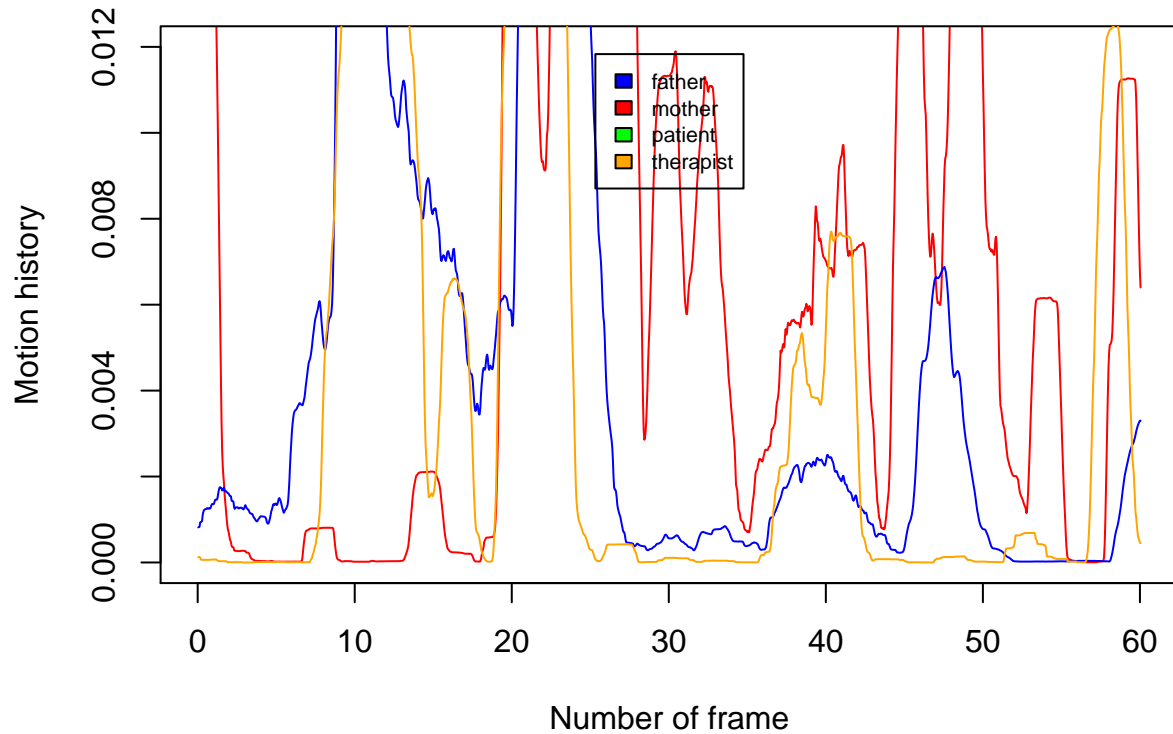
```
## 'data.frame':   1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  0.000804 0.000711 0.00061 0.000536 0.000471 ...
## $ slidedMotherMinute   : num  0.00746 0.00747 0.00746 0.00746 0.00744 ...
## $ slidedTherapistMinute: num  0.000497 0.000478 0.000483 0.000512 0.000517 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute               : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 15 in F1044C video



```
## 'data.frame':   1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  0.000816 0.000823 0.000829 0.000933 0.000942 ...
## $ slidedMotherMinute   : num  0.0349 0.0344 0.0339 0.0336 0.0331 ...
## $ slidedTherapistMinute: num  1.26e-04 1.26e-04 1.26e-04 1.13e-04 7.55e-05 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames                : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute                : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

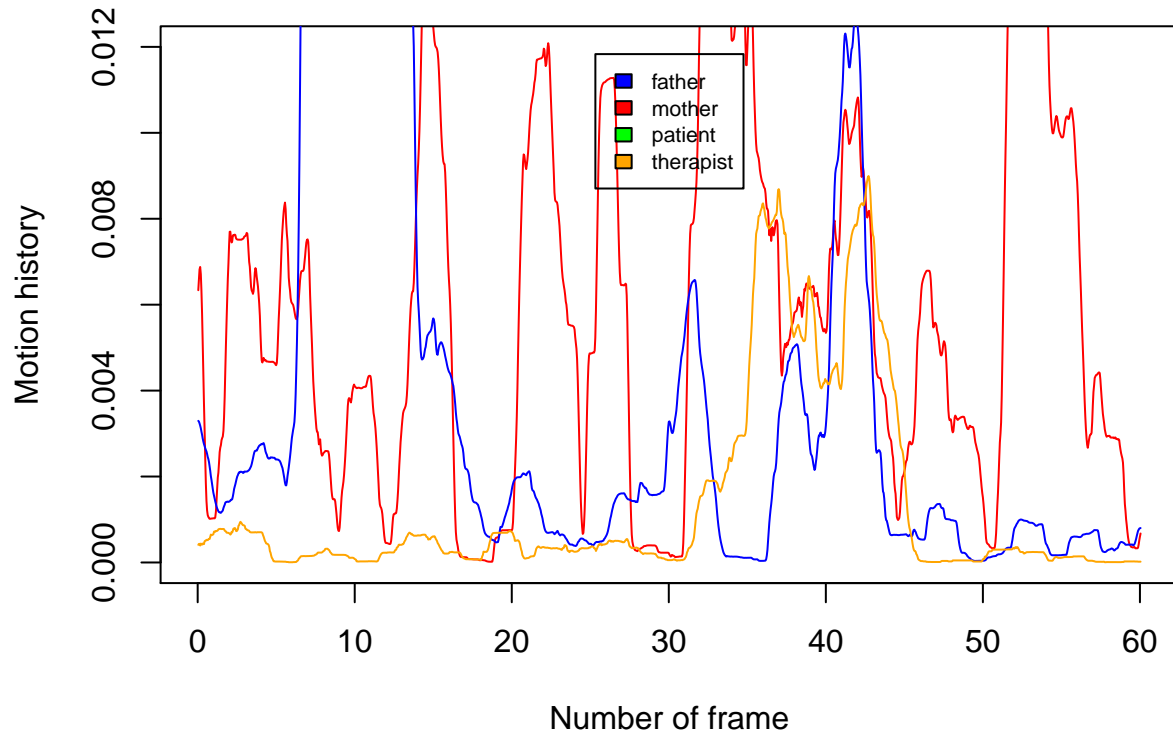
## Motion history with Sliding interval function during minute 16 in F1044C video



```
## 'data.frame':   1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  0.0033 0.00328 0.00324 0.00318 0.00312 ...
## $ slidedMotherMinute   : num  0.00634 0.00659 0.00682 0.00688 0.00671 ...
## $ slidedTherapistMinute: num  0.00041 0.000434 0.000432 0.000395 0.000441 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute               : num  0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

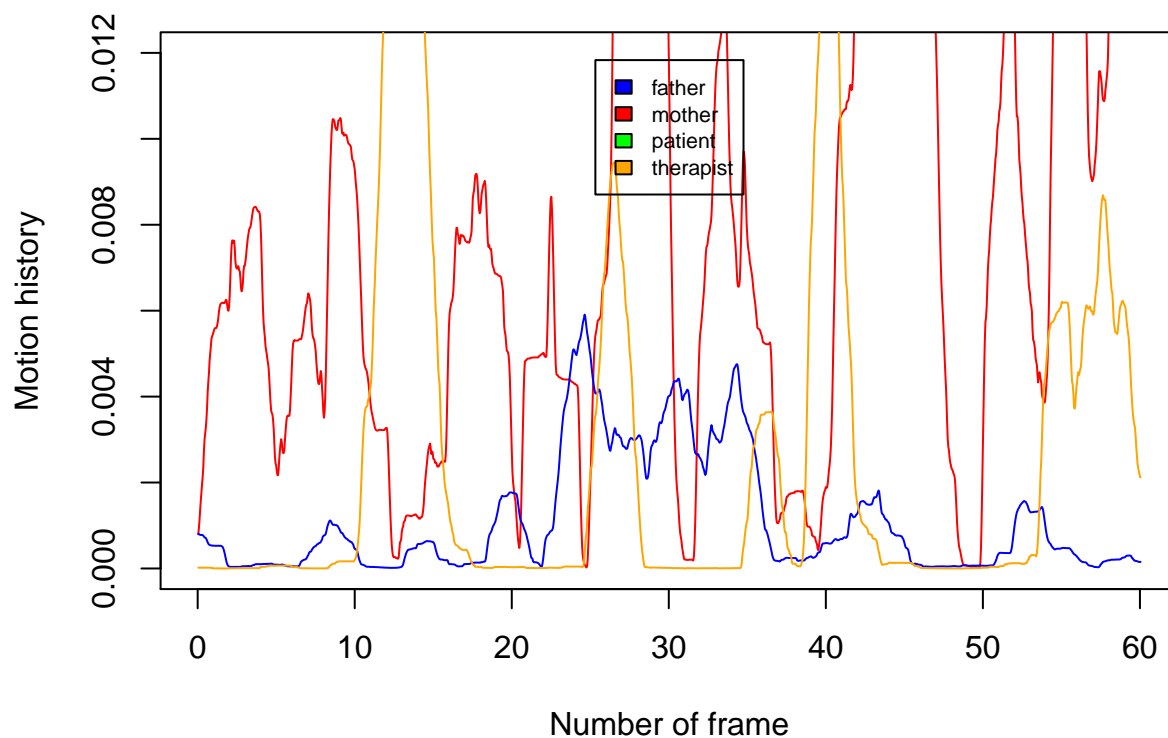


## Motion history with Sliding interval function during minute 17 in F1044C video



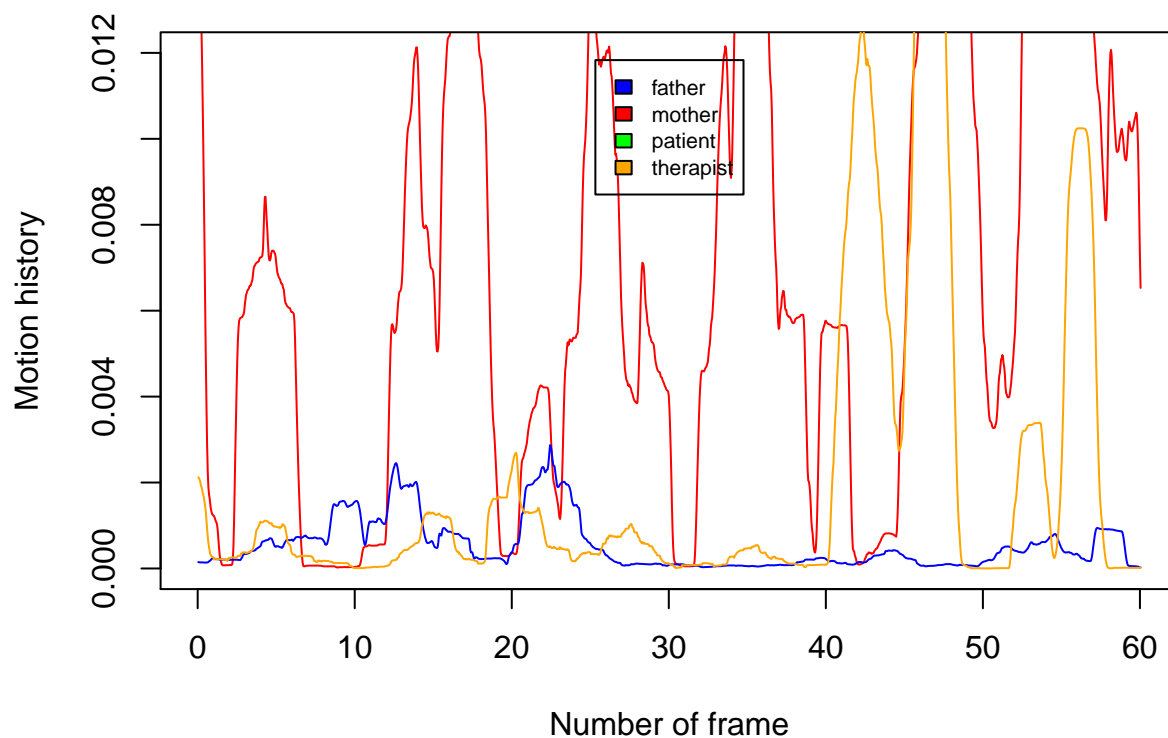
```
## 'data.frame':  1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  0.000805 0.000802 0.000789 0.000782 0.00078 ...
## $ slidedMotherMinute   : num  0.000793 0.000949 0.001141 0.001373 0.001588 ...
## $ slidedTherapistMinute: num  1.87e-05 1.87e-05 1.87e-05 1.87e-05 1.87e-05 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute               : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 18 in F1044C video



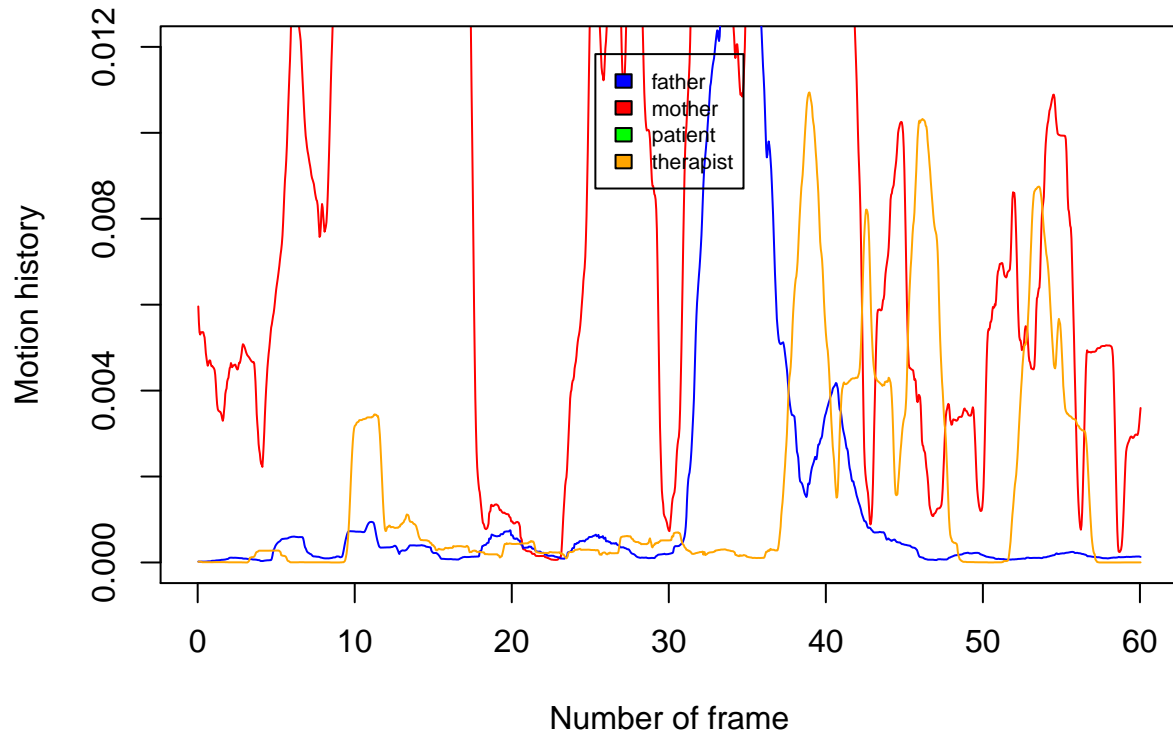
```
## 'data.frame': 1501 obs. of 6 variables:
## $ slidedFatherMinute : num 0.000149 0.000148 0.000142 0.000141 0.000142 ...
## $ slidedMotherMinute : num 0.0167 0.0158 0.015 0.0142 0.0134 ...
## $ slidedTherapistMinute: num 0.00212 0.0021 0.00207 0.00203 0.00194 ...
## $ slidedPatientMinute : num NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames : int 1 2 3 4 5 6 7 8 9 10 ...
## $ minute : num 0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 19 in F1044C video



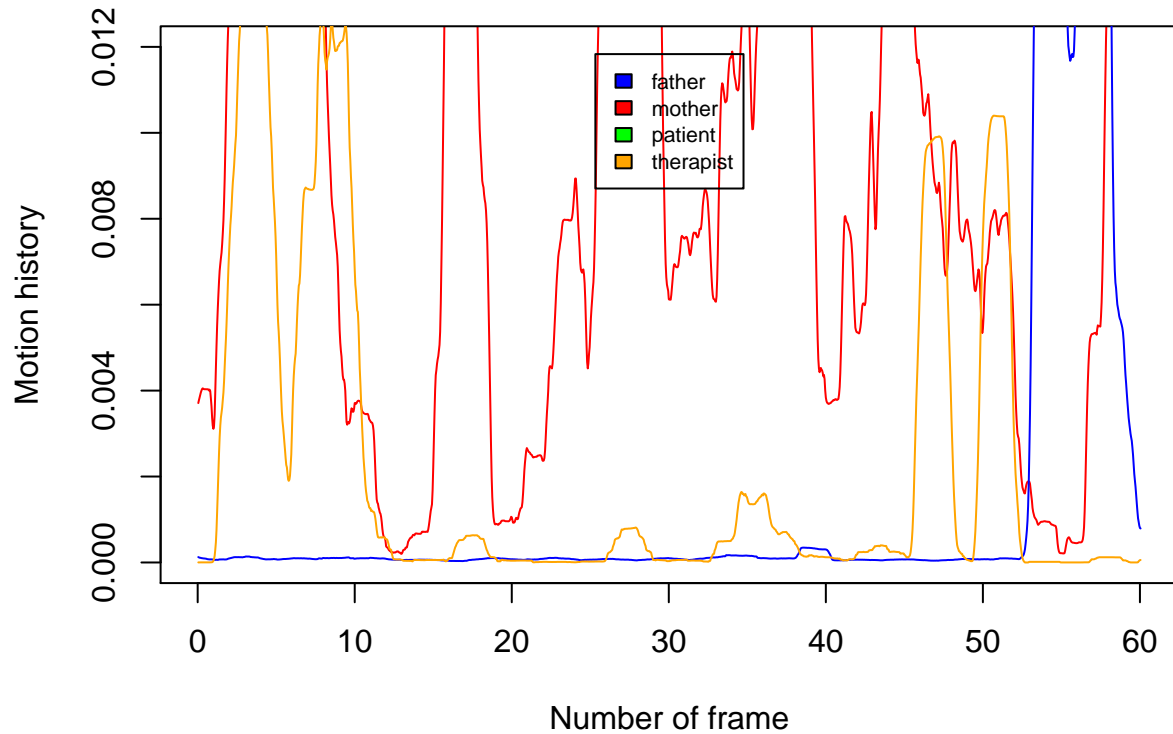
```
## 'data.frame':   1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  2.07e-05 2.00e-05 1.97e-05 1.97e-05 1.91e-05 ...
## $ slidedMotherMinute   : num  0.00596 0.00558 0.00535 0.0053 0.00535 ...
## $ slidedTherapistMinute: num  1.78e-05 1.78e-05 1.78e-05 1.78e-05 1.78e-05 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames                : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute                 : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 20 in F1044C video



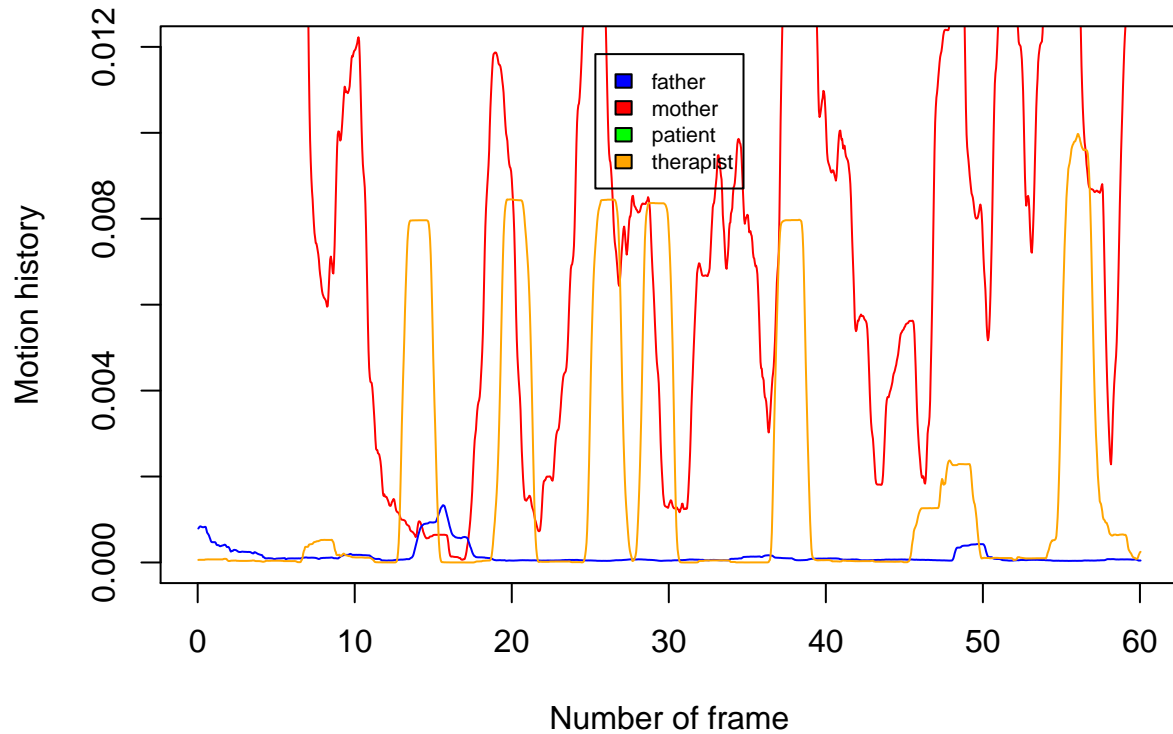
```
## 'data.frame':   1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  0.000126 0.000119 0.000111 0.000109 0.000106 ...
## $ slidedMotherMinute   : num  0.00371 0.00378 0.00383 0.0039 0.00395 ...
## $ slidedTherapistMinute: num  1.70e-06 1.70e-06 1.70e-06 1.70e-06 2.55e-06 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute               : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 21 in F1044C video



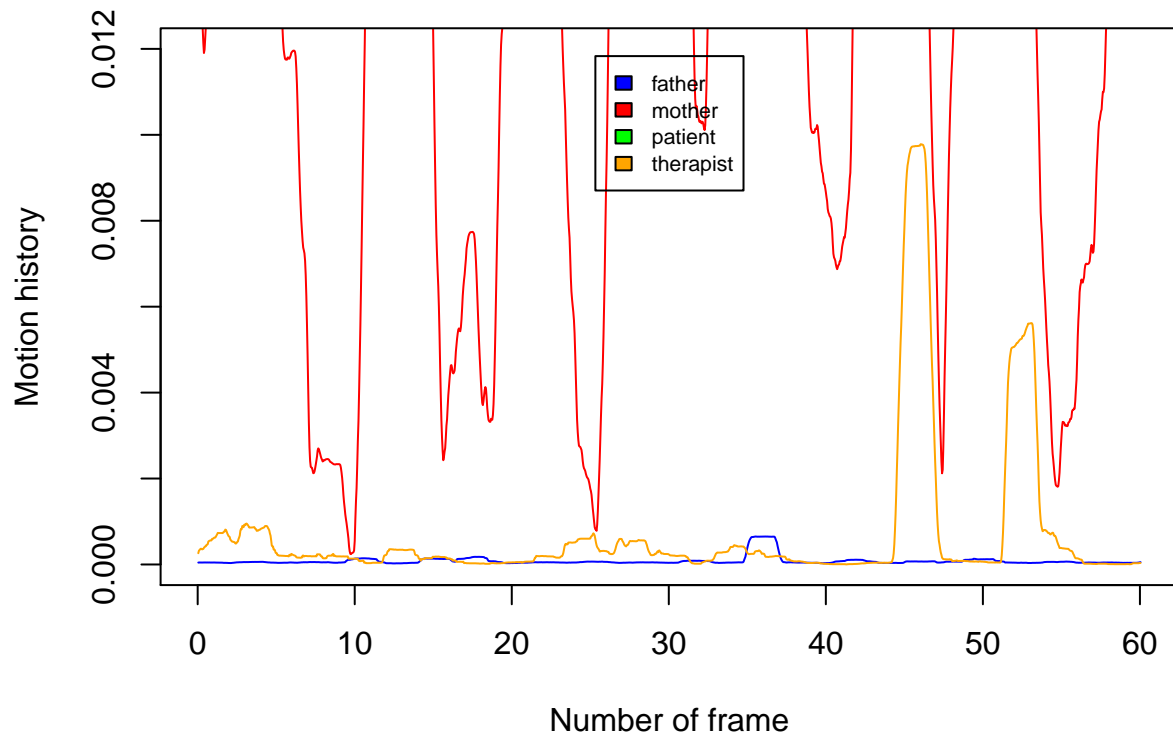
```
## 'data.frame':  1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  0.00079 0.000809 0.00083 0.000839 0.000838 ...
## $ slidedMotherMinute   : num  0.0503 0.0511 0.0517 0.0519 0.0515 ...
## $ slidedTherapistMinute: num  5.94e-05 6.11e-05 6.20e-05 6.20e-05 6.20e-05 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute               : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 22 in F1044C video



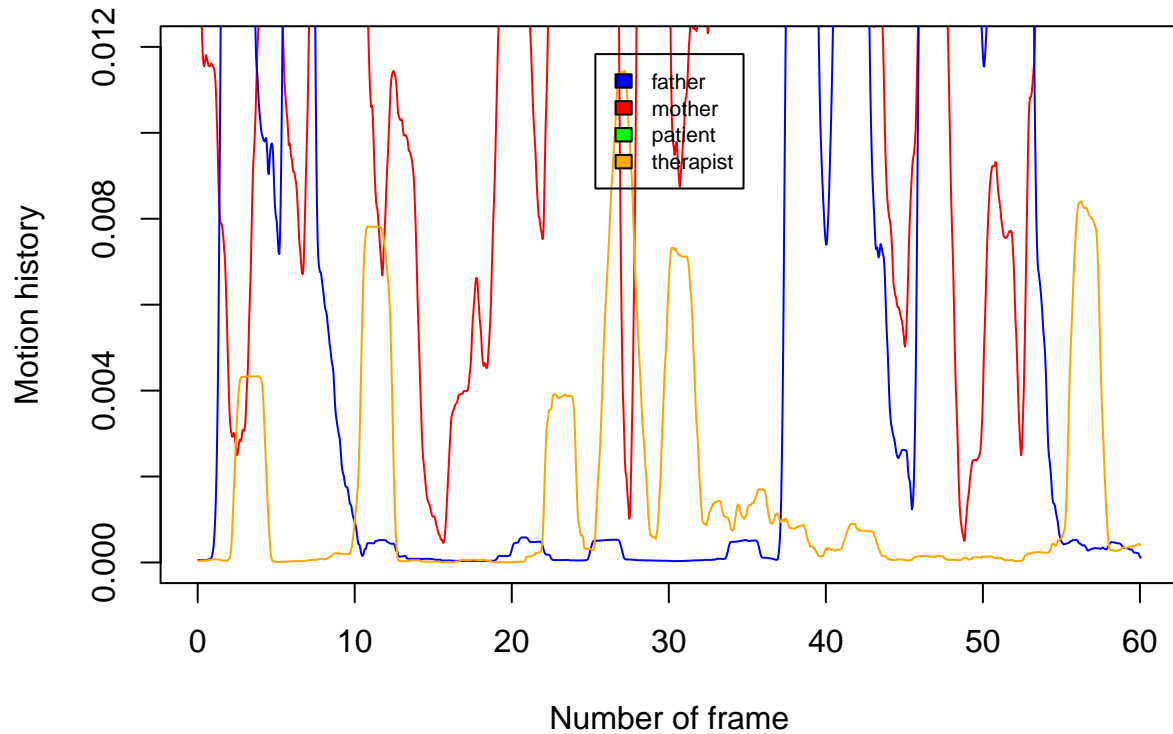
```
## 'data.frame':  1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  4.47e-05 4.47e-05 4.63e-05 4.57e-05 4.53e-05 ...
## $ slidedMotherMinute   : num  0.0151 0.015 0.0147 0.0143 0.0138 ...
## $ slidedTherapistMinute: num  0.000261 0.000321 0.000337 0.000354 0.000368 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute               : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 23 in F1044C video



```
## 'data.frame':   1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  5.42e-05 5.45e-05 5.32e-05 5.29e-05 5.29e-05 ...
## $ slidedMotherMinute   : num  0.0135 0.0136 0.0137 0.0136 0.0133 ...
## $ slidedTherapistMinute: num  3.56e-05 3.56e-05 3.65e-05 3.82e-05 4.24e-05 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute                : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

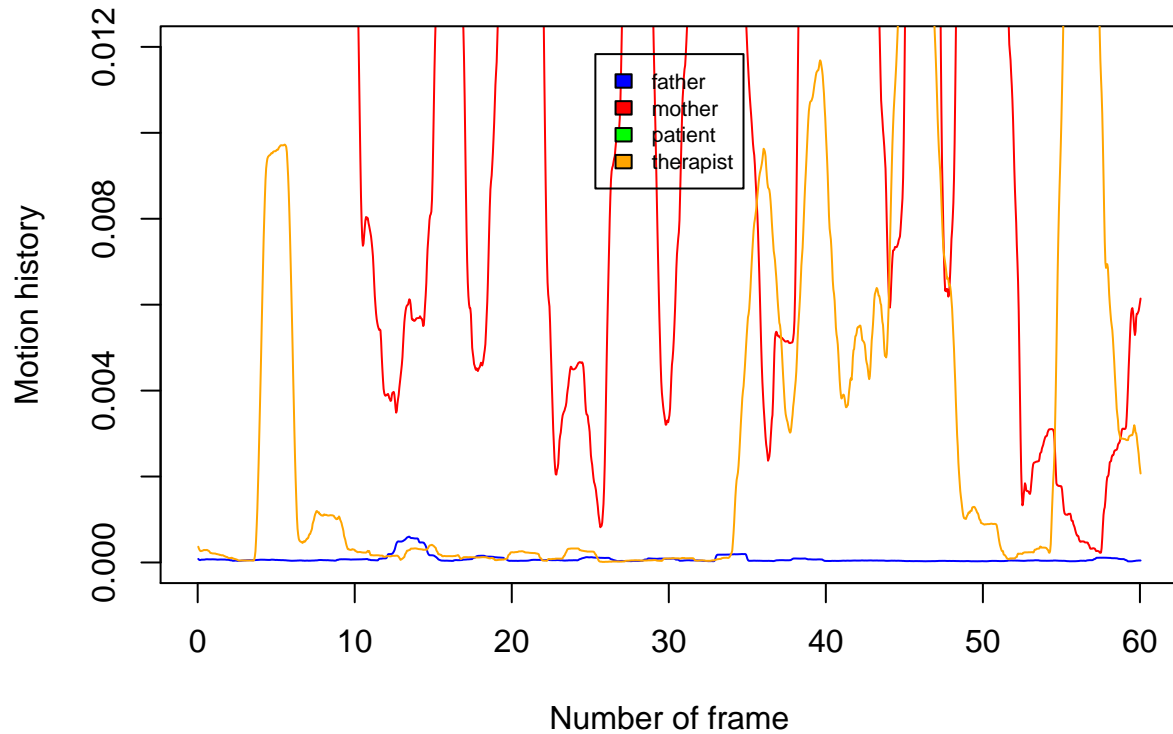
## Motion history with Sliding interval function during minute 24 in F1044C video



```
## 'data.frame':   1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  8.25e-05 6.67e-05 5.78e-05 5.55e-05 5.98e-05 ...
## $ slidedMotherMinute   : num  0.0435 0.0434 0.0432 0.043 0.0429 ...
## $ slidedTherapistMinute: num  0.000367 0.000322 0.000292 0.000272 0.000263 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute               : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

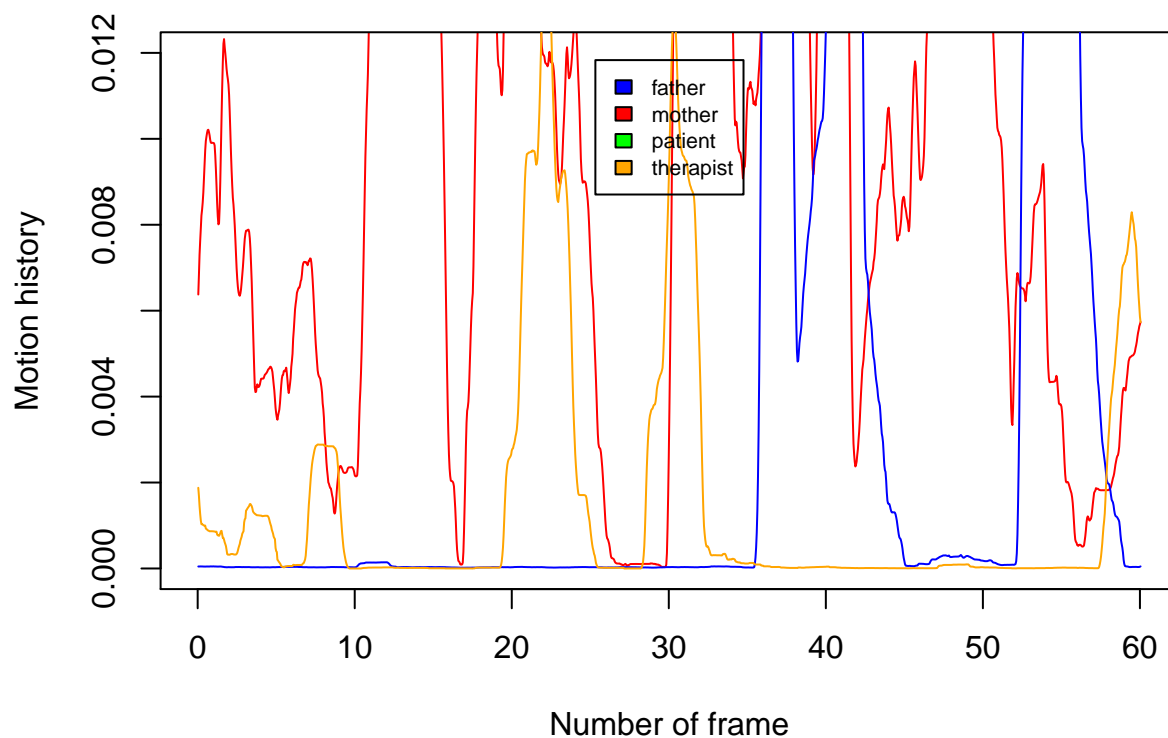


## Motion history with Sliding interval function during minute 25 in F1044C video



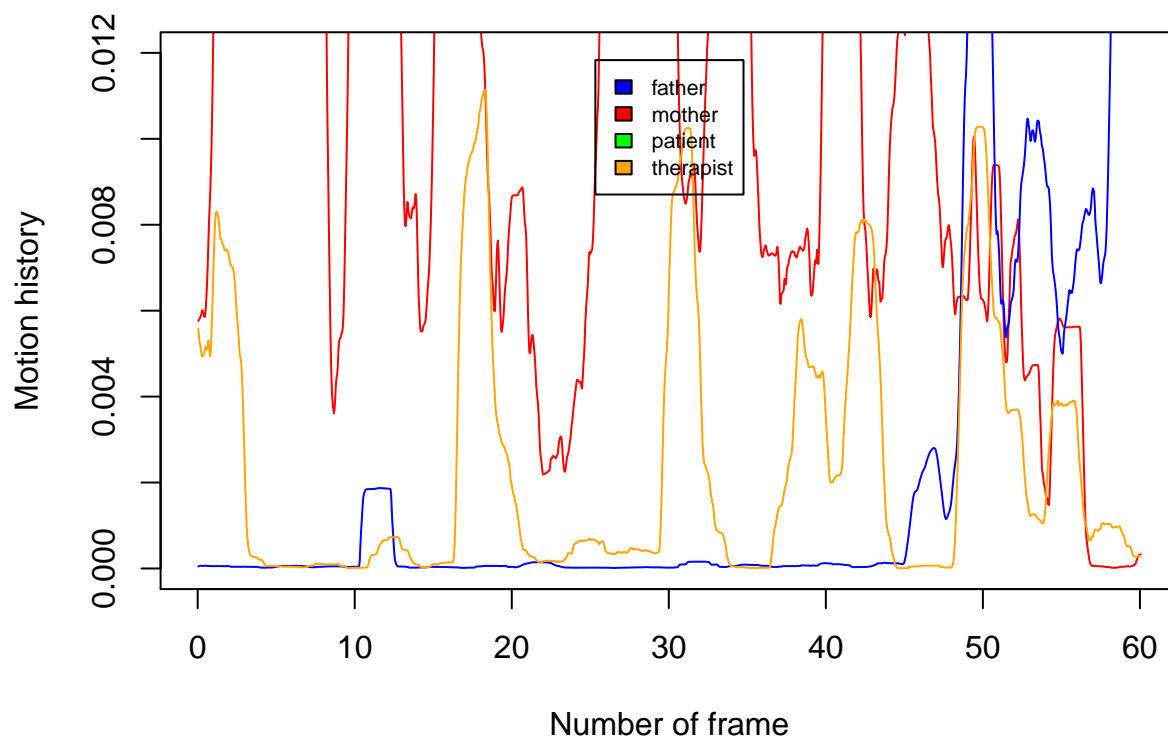
```
## 'data.frame':   1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  4.63e-05 4.60e-05 4.63e-05 4.73e-05 4.70e-05 ...
## $ slidedMotherMinute   : num  0.00638 0.00683 0.0073 0.00762 0.00792 ...
## $ slidedTherapistMinute: num  0.00188 0.00169 0.00146 0.00127 0.00112 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute                : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 26 in F1044C video



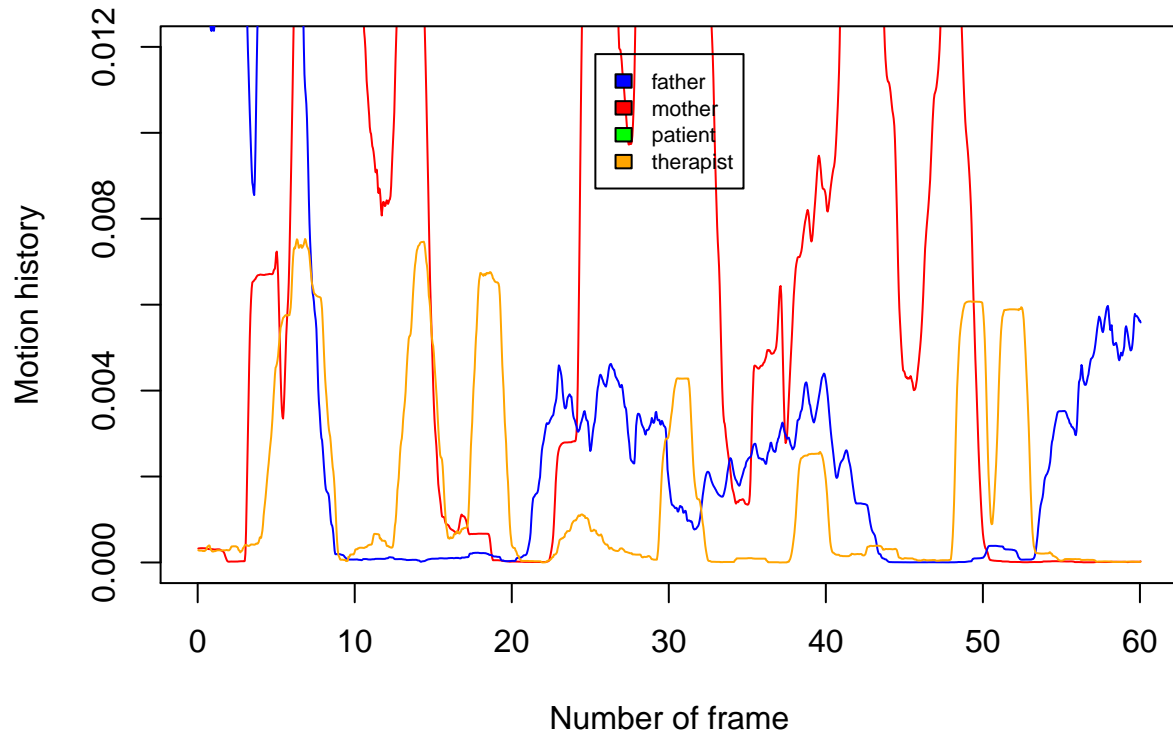
```
## 'data.frame':  1501 obs. of  6 variables:
## $ slidedFatherMinute   : num  4.93e-05 5.06e-05 5.52e-05 6.28e-05 6.31e-05 ...
## $ slidedMotherMinute   : num  0.00576 0.00579 0.0058 0.00585 0.00588 ...
## $ slidedTherapistMinute: num  0.00558 0.00542 0.00525 0.00521 0.00513 ...
## $ slidedPatientMinute  : num  NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames               : int   1  2  3  4  5  6  7  8  9 10 ...
## $ minute               : num   0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 27 in F1044C video



```
## 'data.frame': 1501 obs. of 6 variables:
## $ slidedFatherMinute : num 0.0266 0.0258 0.0248 0.0237 0.0226 ...
## $ slidedMotherMinute : num 0.000326 0.000327 0.00033 0.000331 0.000332 ...
## $ slidedTherapistMinute: num 0.000286 0.000278 0.000277 0.000277 0.000277 ...
## $ slidedPatientMinute : num NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...
## $ frames : int 1 2 3 4 5 6 7 8 9 10 ...
## $ minute : num 0.04 0.08 0.12 0.16 0.2 0.24 0.28 0.32 0.36 0.4 ...
```

## Motion history with Sliding interval function during minute 28 in F1044C video



## Export no log filtered data in text files

```
## REMINDER:
#SlidingInterval <- function(subject, indexOfvideos=1:NumberOfvideos, interval, data) with :
# subject : subject studied (patient, mother, father or therapist)
# indexOfvideos : list of videos studied (element eg. 3 or list eg 1:3 or c(1,2,4))
# interval : number of frames in the studied interval
# data : data frame where there is data

#index de la vid?eo de 1ere a la length de indexvideo

videoIndex <- 1
# videoName est le nom de la video actuelle
for (videoName in indexList){
# Compute sliding interval for each participant
  print(paste("Computing slidedFather", videoName))
  slidedFather <- SlidingInterval("father", videoIndex, 5, data)

  print(paste("Computing slidedMother", videoName))
  slidedMother <- SlidingInterval("mother", videoIndex, 5, data)

  print(paste("Computing slidedPatient", videoName))
  slidedPatient <- SlidingInterval("patient", videoIndex, 5, data)

  print(paste("Computing slidedTherapist", videoName))
}
```

```

        slidedTherapist <- SlidingInterval("therapist", videoIndex, 5, data)

# create a data frame to store temporarily this data with NA
        slidedVideo <- data.frame(
            slidedFather, slidedMother, slidedPatient, slidedTherapist,
            "video"=rep(indexList[videoIndex], length(slidedFather)),
            frame_index = 1:length(slidedFather))

        str(slidedVideo)
        summary(slidedVideo)
#         dfSliding <- data.frame()
#         for (participant in 1:4){
# If the colum is not empty, takes its length and begin a data frame with it
#             if (dataFrame == FALSE & (length(slidedVideo[participant][!is.na(slidedVideo[participant])]))
#                 dfSliding <- data.frame(
#                     "video"=rep(indexList[videoIndex], length(slidedVideo[participant])),
#                     frame_index = (1:dim(slidedVideo[1])[1]),
#                     slidedVideo[participant])
#                 dataFrame <- TRUE}
#             else if (dataFrame == FALSE){}
#             else{
#                 dfSliding <- cbind(dfSliding, slidedVideo[participant])
#             }
        write.csv(slidedVideo, paste("/Users/Ofix/Documents/Fac/internat/Recherche/projets/synchro/synchro",
        videoIndex <-(videoIndex+1)
    }

```

## Export log filtered data in text files

```

videoIndex <- 1
# videoName est le nom de la video actuelle
for (videoName in indexList){
# Compute slinding interval for each participant
    print(paste("Computing slidedFather", videoName))
    slidedFather <- SlidingInterval("logFather", videoIndex, 5, data)

    print(paste("Computing slidedMother", videoName))
    slidedMother <- SlidingInterval("logMother", videoIndex, 5, data)

    print(paste("Computing slidedPatient", videoName))
    slidedPatient <- SlidingInterval("logPatient", videoIndex, 5, data)

    print(paste("Computing slidedTherapist", videoName))
    slidedTherapist <- SlidingInterval("logTherapist", videoIndex, 5, data)

# create a data frame to store temporarily this data with NA

    slidedVideo <- data.frame(
        slidedFather, slidedMother, slidedPatient, slidedTherapist,
        "video"=rep(indexList[videoIndex], length(slidedFather)),
        frame_index = 1:length(slidedFather))

```

```

write.csv(slidedVideo, paste("/Users/Ofix/Documents/Fac/internat/Recherche/projets/synchro/
videoIndex <-(videoIndex+1)
}

```

## SyncPy utilisation for creating synchrony dataframe

After extracting filtered motion motion history with mean on sliding interval (overlapping interval) of 5 frames

And after putting this data on a CSV file slideddata.csv

We import this data on python Script with panda module Call\_S\_Estimator.py

This script will compute the synchrony between each dyad of the interaction and of the whole group

It will return a csv file for each video SSIXXXX.csv with XXXX the name of the video (F1044C, F1044D1, etc) that we can import with R with

this following function

```

## [1] "SSI Files Directory"
## [1] "SSI Files List"
## [1] "../Data/CSV/Synchrony/log/F1002A1.log.slideddata.csv.csv"
## [2] "../Data/CSV/Synchrony/log/F1002A2.log.slideddata.csv.csv"
## [3] "../Data/CSV/Synchrony/log/F1002B2.log.slideddata.csv.csv"
## [4] "../Data/CSV/Synchrony/log/F1002C1.log.slideddata.csv.csv"
## [5] "../Data/CSV/Synchrony/log/F1044C1.log.slideddata.csv.csv"
## [6] "../Data/CSV/Synchrony/log/F1044C2.log.slideddata.csv.csv"
## [7] "../Data/CSV/Synchrony/log/F1044D1.log.slideddata.csv.csv"
## [8] "../Data/CSV/Synchrony/log/F1044D2.log.slideddata.csv.csv"
## [9] "../Data/CSV/Synchrony/log/F1044E1.log.slideddata.csv.csv"
## [10] "../Data/CSV/Synchrony/log/F1044E2.log.slideddata.csv.csv"
## [11] "../Data/CSV/Synchrony/log/F1044F1.log.slideddata.csv.csv"
## [12] "../Data/CSV/Synchrony/log/F1044F2.log.slideddata.csv.csv"
## [13] "../Data/CSV/Synchrony/log/F1044G.log.slideddata.csv.csv"
## [14] "../Data/CSV/Synchrony/log/F1044H1.log.slideddata.csv.csv"
## [15] "../Data/CSV/Synchrony/log/F1044H2.log.slideddata.csv.csv"
## [16] "../Data/CSV/Synchrony/log/F1044I1.log.slideddata.csv.csv"
## [17] "../Data/CSV/Synchrony/log/F1044I2.log.slideddata.csv.csv"
## [18] "../Data/CSV/Synchrony/log/F1044L1.log.slideddata.csv.csv"
## [19] "../Data/CSV/Synchrony/log/F1044L2.log.slideddata.csv.csv"
## [20] "../Data/CSV/Synchrony/log/F1044M1.log.slideddata.csv.csv"
## [21] "../Data/CSV/Synchrony/log/F1044M2.log.slideddata.csv.csv"
## [22] "../Data/CSV/Synchrony/log/F1044N.log.slideddata.csv.csv"
## [23] "../Data/CSV/Synchrony/log/F1044O1.log.slideddata.csv.csv"
## [24] "../Data/CSV/Synchrony/log/F1044O2.log.slideddata.csv.csv"
## [25] "../Data/CSV/Synchrony/log/F1044P.log.slideddata.csv.csv"
## [26] "../Data/CSV/Synchrony/log/F1044Q1.log.slideddata.csv.csv"
## [27] "../Data/CSV/Synchrony/log/F1044Q2.log.slideddata.csv.csv"

```

```

## [28] "../Data/CSV/Synchrony/log/F1044R1.log.slideddata.csv.csv"
## [29] "../Data/CSV/Synchrony/log/F1044R2.log.slideddata.csv.csv"
## [30] "../Data/CSV/Synchrony/log/F1069A1.log.slideddata.csv.csv"
## [31] "../Data/CSV/Synchrony/log/F1069B1.log.slideddata.csv.csv"
## [32] "../Data/CSV/Synchrony/log/F1069B2.log.slideddata.csv.csv"
## [33] "../Data/CSV/Synchrony/log/F1069C1.log.slideddata.csv.csv"
## [34] "../Data/CSV/Synchrony/log/F1069D2.log.slideddata.csv.csv"
## [35] "../Data/CSV/Synchrony/log/F1073A1.log.slideddata.csv.csv"
## [36] "../Data/CSV/Synchrony/log/F1073A2.log.slideddata.csv.csv"
## [37] "../Data/CSV/Synchrony/log/F1073B1.log.slideddata.csv.csv"
## [38] "../Data/CSV/Synchrony/log/F1073B2.log.slideddata.csv.csv"
## [39] "../Data/CSV/Synchrony/log/F1101A2.log.slideddata.csv.csv"
## [40] "../Data/CSV/Synchrony/log/F1101C2.log.slideddata.csv.csv"

## [1] "../Data/CSV/Synchrony/noLog/F1002A1.slideddata.csv.SSI.csv"
## [2] "../Data/CSV/Synchrony/noLog/F1002A2.slideddata.csv.SSI.csv"
## [3] "../Data/CSV/Synchrony/noLog/F1002B2.slideddata.csv.SSI.csv"
## [4] "../Data/CSV/Synchrony/noLog/F1002C1.slideddata.csv.SSI.csv"
## [5] "../Data/CSV/Synchrony/noLog/F1044C1.slideddata.csv.SSI.csv"
## [6] "../Data/CSV/Synchrony/noLog/F1044C2.slideddata.csv.SSI.csv"
## [7] "../Data/CSV/Synchrony/noLog/F1044D1.slideddata.csv.SSI.csv"
## [8] "../Data/CSV/Synchrony/noLog/F1044D2.slideddata.csv.SSI.csv"
## [9] "../Data/CSV/Synchrony/noLog/F1044E1.slideddata.csv.SSI.csv"
## [10] "../Data/CSV/Synchrony/noLog/F1044E2.slideddata.csv.SSI.csv"
## [11] "../Data/CSV/Synchrony/noLog/F1044F1.slideddata.csv.SSI.csv"
## [12] "../Data/CSV/Synchrony/noLog/F1044F2.slideddata.csv.SSI.csv"
## [13] "../Data/CSV/Synchrony/noLog/F1044G.slideddata.csv.SSI.csv"
## [14] "../Data/CSV/Synchrony/noLog/F1044H1.slideddata.csv.SSI.csv"
## [15] "../Data/CSV/Synchrony/noLog/F1044H2.slideddata.csv.SSI.csv"
## [16] "../Data/CSV/Synchrony/noLog/F1044I1.slideddata.csv.SSI.csv"
## [17] "../Data/CSV/Synchrony/noLog/F1044I2.slideddata.csv.SSI.csv"
## [18] "../Data/CSV/Synchrony/noLog/F1044L1.slideddata.csv.SSI.csv"
## [19] "../Data/CSV/Synchrony/noLog/F1044L2.slideddata.csv.SSI.csv"
## [20] "../Data/CSV/Synchrony/noLog/F1044M1.slideddata.csv.SSI.csv"
## [21] "../Data/CSV/Synchrony/noLog/F1044M2.slideddata.csv.SSI.csv"
## [22] "../Data/CSV/Synchrony/noLog/F1044N.slideddata.csv.SSI.csv"
## [23] "../Data/CSV/Synchrony/noLog/F1044O1.slideddata.csv.SSI.csv"
## [24] "../Data/CSV/Synchrony/noLog/F1044O2.slideddata.csv.SSI.csv"
## [25] "../Data/CSV/Synchrony/noLog/F1044P.slideddata.csv.SSI.csv"
## [26] "../Data/CSV/Synchrony/noLog/F1044Q1.slideddata.csv.SSI.csv"
## [27] "../Data/CSV/Synchrony/noLog/F1044Q2.slideddata.csv.SSI.csv"
## [28] "../Data/CSV/Synchrony/noLog/F1044R1.slideddata.csv.SSI.csv"
## [29] "../Data/CSV/Synchrony/noLog/F1044R2.slideddata.csv.SSI.csv"
## [30] "../Data/CSV/Synchrony/noLog/F1069A1.slideddata.csv.SSI.csv"
## [31] "../Data/CSV/Synchrony/noLog/F1069B1.slideddata.csv.SSI.csv"
## [32] "../Data/CSV/Synchrony/noLog/F1069B2.slideddata.csv.SSI.csv"
## [33] "../Data/CSV/Synchrony/noLog/F1069C1.slideddata.csv.SSI.csv"
## [34] "../Data/CSV/Synchrony/noLog/F1069D2.slideddata.csv.SSI.csv"
## [35] "../Data/CSV/Synchrony/noLog/F1073A1.slideddata.csv.SSI.csv"
## [36] "../Data/CSV/Synchrony/noLog/F1073A2.slideddata.csv.SSI.csv"

```

## Description of SSI data frame

```
str(SSIdataFrame)

## 'data.frame':    1558 obs. of  14 variables:
## $ X              : int  0 1 2 3 4 5 6 7 8 9 ...
## $ Interval       : int  1 2 3 4 5 6 7 8 9 10 ...
## $ Time_min       : num  0 0.167 0.333 0.5 0.667 ...
## $ video          : Factor w/ 36 levels "F1002A1","F1002A2",...: 1 1 1 1 1 1 1 1 1 1 ...
## $ SSI_fa_mo      : num  3.29e-03 2.81e-02 6.28e-06 1.91e-03 3.42e-04 ...
## $ SSI_fa_mo_th   : num  0.06552 0.05668 0.00475 0.01931 0.00603 ...
## $ SSI_fa_th      : num  0.149776 0.087342 0.000483 0.03006 0.003311 ...
## $ SSI_mo_th      : num  0.00132 0.02688 0.01082 0.01481 0.01079 ...
## $ SSI_pa_th      : num  NA NA NA NA NA NA NA NA NA NA ...
## $ SSI_mo_pa      : num  NA NA NA NA NA NA NA NA NA NA ...
## $ SSI_mo_pa_th   : num  NA NA NA NA NA NA NA NA NA NA ...
## $ SSI_fa_pa      : num  NA NA NA NA NA NA NA NA NA NA ...
## $ SSI_fa_mo_pa   : num  NA NA NA NA NA NA NA NA NA NA ...
## $ SSI_fa_pa_th   : num  NA NA NA NA NA NA NA NA NA NA ...
```

## Synchrony scores for each dyad, triad and for the whole group

In legend, mean for all the video.

```
for (i in unique(SSIdataFrame$video))
  {par(mar=c(4,4,4,3), mfrow=c(1,1))
    plot(SSIdataFrame[which(SSIdataFrame$video==i),]$Time_min,
         SSIdataFrame[which(SSIdataFrame$video==i),]$SSI_fa_mo,
         type="l", ylim=c(0, 0.3), col=rainbow(4)[1],
         main=paste("Synchrony scores for each dyad and for \n the whole group in", i, "video"),
         xlab = "Time (minute)", ylab="Synchrony score", lwd=2,
         xaxp=c(0,length(SSIdataFrame$Time_min), length(SSIdataFrame$Time_min)))
    abline(h=mean(SSIdataFrame$SSI_fa_mo, na.rm=TRUE), col=rainbow(11)[1], lwd=2, lty=2)

    lines(SSIdataFrame[which(SSIdataFrame$video==i),]$SSI_fa_mo_pa, col=rainbow(11)[2], lwd=2)
    abline(h= mean(SSIdataFrame$SSI_fa_mo_pa, na.rm=TRUE), col=rainbow(11)[2], lwd=2, lty=2)

    # lines(SSIdataFrame[which(SSIdataFrame$video==i),]$SSI_fa_mo_pa_th, col=rainbow(11)[3], lwd=2)
    # abline(h= mean(SSIdataFrame$SSI_fa_mo_pa_th, na.rm=TRUE), col=rainbow(11)[3], lwd=2, lty=2)

    lines(SSIdataFrame[which(SSIdataFrame$video==i),]$SSI_fa_mo_th, col=rainbow(11)[4], lwd=2)
    abline(h= mean(SSIdataFrame$SSI_fa_mo_th, na.rm=TRUE), col=rainbow(11)[4], lwd=2, lty=2)

    lines(SSIdataFrame[which(SSIdataFrame$video==i),]$SSI_fa_pa, col=rainbow(11)[5], lwd=2)
    abline(h= mean(SSIdataFrame$SSI_fa_pa, na.rm=TRUE), col=rainbow(11)[5], lwd=2, lty=2)

    # lines(SSIdataFrame[which(SSIdataFrame$video==i),]$SSI_fa_pa_th, col=rainbow(11)[6], lwd=2)
    # abline(h= mean(SSIdataFrame$SSI_fa_pa_th, na.rm=TRUE), col=rainbow(11)[6], lwd=2, lty=2)

    lines(SSIdataFrame[which(SSIdataFrame$video==i),]$SSI_fa_th, col=rainbow(11)[7], lwd=2)
    abline(h= mean(SSIdataFrame$SSI_fa_th, na.rm=TRUE), col=rainbow(11)[7], lwd=2, lty=2)

    lines(SSIdataFrame[which(SSIdataFrame$video==i),]$SSI_mo_pa, col=rainbow(11)[8], lwd=2)
```



```

abline(h= mean(SSIdataFrame$SSI_mo_pa, na.rm=TRUE), col=rainbow(11)[8], lwd=2, lty=2)

lines(SSIdataFrame[which(SSIdataFrame$video==i),]$SSI_mo_pa_th, col=rainbow(11)[9], lwd=2)
abline(h= mean(SSIdataFrame$SSI_mo_pa_th, na.rm=TRUE), col=rainbow(11)[9], lwd=2, lty=2)
lines(SSIdataFrame[which(SSIdataFrame$video==i),]$SSI_mo_th, col=rainbow(11)[10], lwd=2)
abline(h= mean(SSIdataFrame$SSI_mo_th, na.rm=TRUE), col=rainbow(11)[10], lwd=2, lty=2)

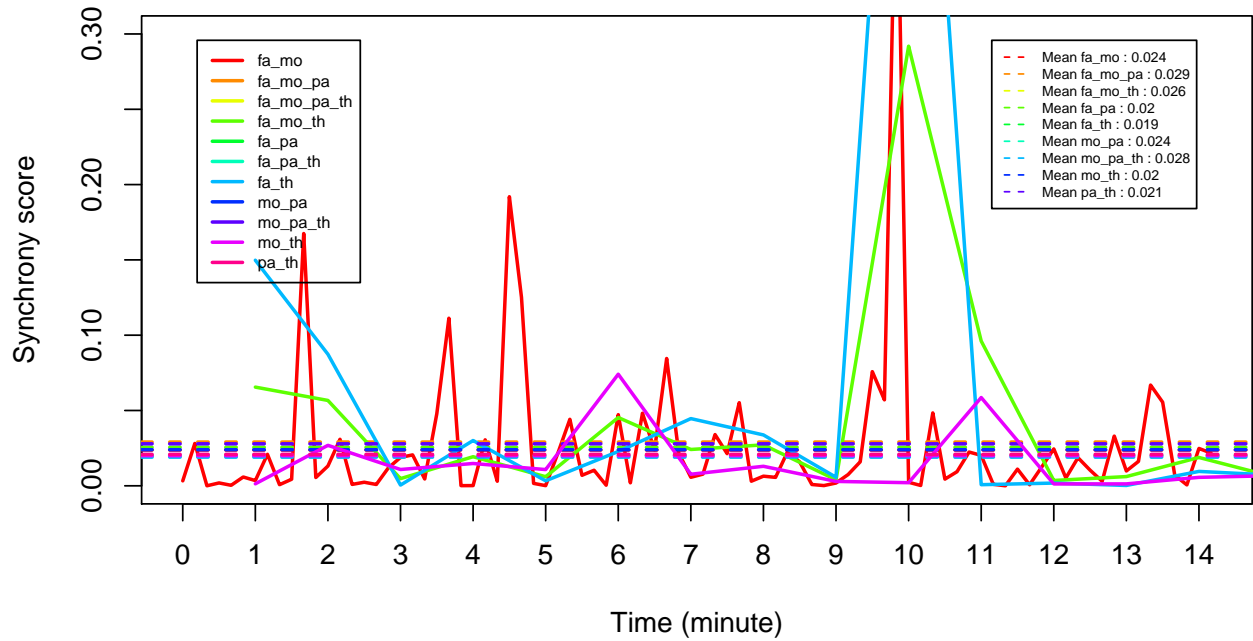
      lines(SSIdataFrame[which(SSIdataFrame$video==i),]$SSI_pa_th, col=rainbow(11)[11], lwd=2)
      abline(h= mean(SSIdataFrame$SSI_pa_th, na.rm=TRUE), col=rainbow(11)[11], lwd=2, lty=2)

legend("topleft", inset=.05, c("fa_mo", "fa_mo_pa", "fa_mo_pa_th",
"fa_mo_th", "fa_pa", "fa_pa_th", "fa_th",
"mo_pa", "mo_pa_th", "mo_th", "pa_th"),
col=rainbow(11), cex=0.6, lwd=2)

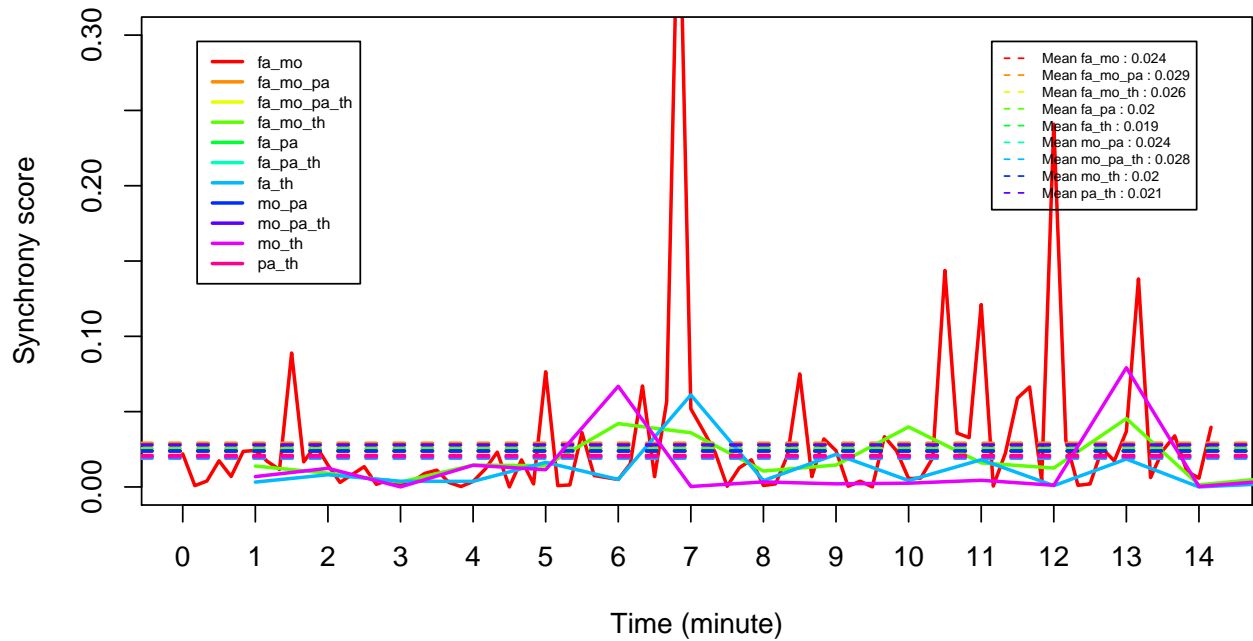
legend("topright", inset=.05, c(paste ("Mean fa_mo :",
      round(mean(SSIdataFrame$SSI_fa_mo, na.rm=TRUE),3)),
  paste ("Mean fa_mo_pa :", round(mean(SSIdataFrame$SSI_fa_mo_pa,na.rm=TRUE),3)),
#   paste ("Mean fa_mo_pa_th :", #round(mean(SSIdataFrame$SSI_fa_mo_pa_th),3)),
  paste ("Mean fa_mo_th :", round(mean(SSIdataFrame$SSI_fa_mo_th,na.rm=TRUE),3)),
  paste ("Mean fa_pa :", round(mean(SSIdataFrame$SSI_fa_pa, na.rm=TRUE),3)),
#   paste ("Mean fa_pa_th :", round(mean(SSIdataFrame$SSI_fa_pa_th,na.rm=TRUE),3)),
  paste ("Mean fa_th :", round(mean(SSIdataFrame$SSI_fa_th,na.rm=TRUE),3)),
  paste ("Mean mo_pa :", round(mean(SSIdataFrame$SSI_mo_pa,na.rm=TRUE),3)),
  paste ("Mean mo_pa_th :", round(mean(SSIdataFrame$SSI_mo_pa_th,na.rm=TRUE),3)),
  paste ("Mean mo_th :", round(mean(SSIdataFrame$SSI_mo_th,na.rm=TRUE),3)),
  paste ("Mean pa_th :", round(mean(SSIdataFrame$SSI_pa_th,na.rm=TRUE),3))),
col=rainbow(11), cex=0.5, lty=2, lwd=1)}

```

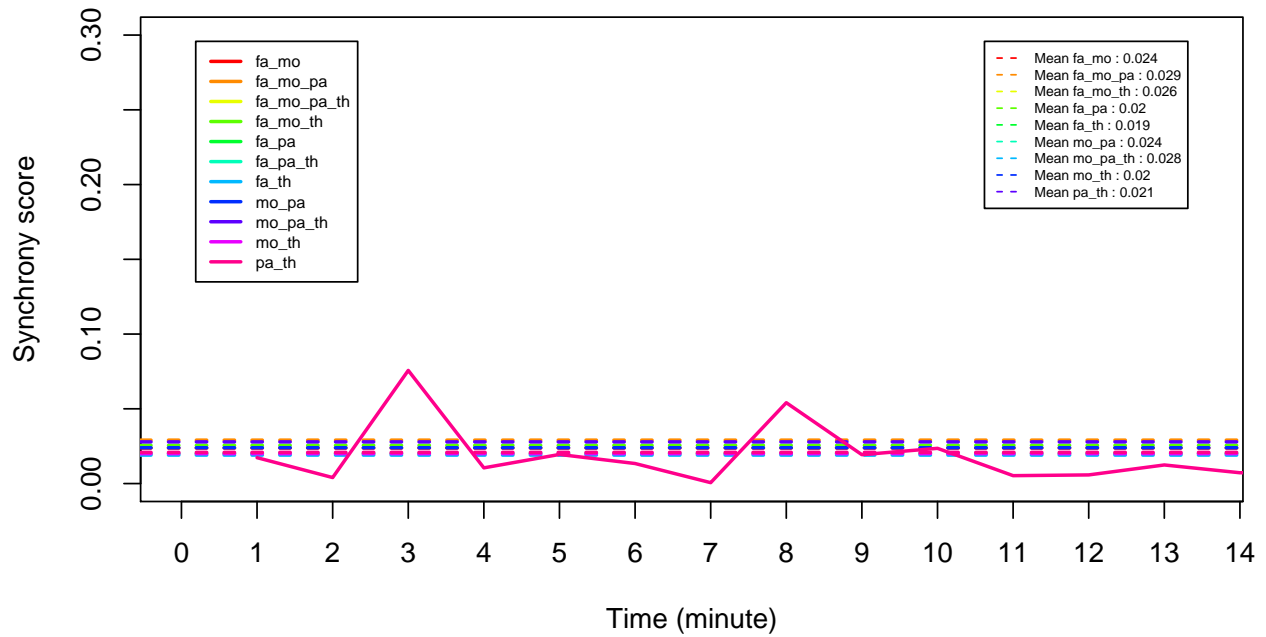
**Synchrony scores for each dyad and for the whole group in F1002A1 video**



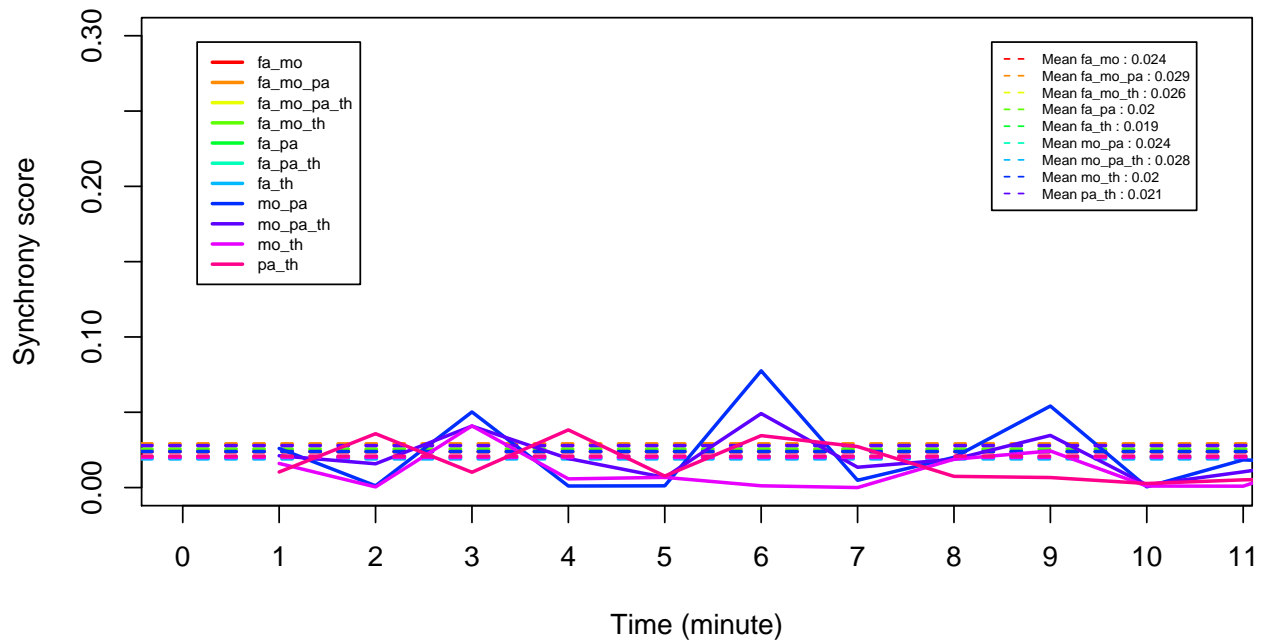
**Synchrony scores for each dyad and for the whole group in F1002A2 video**



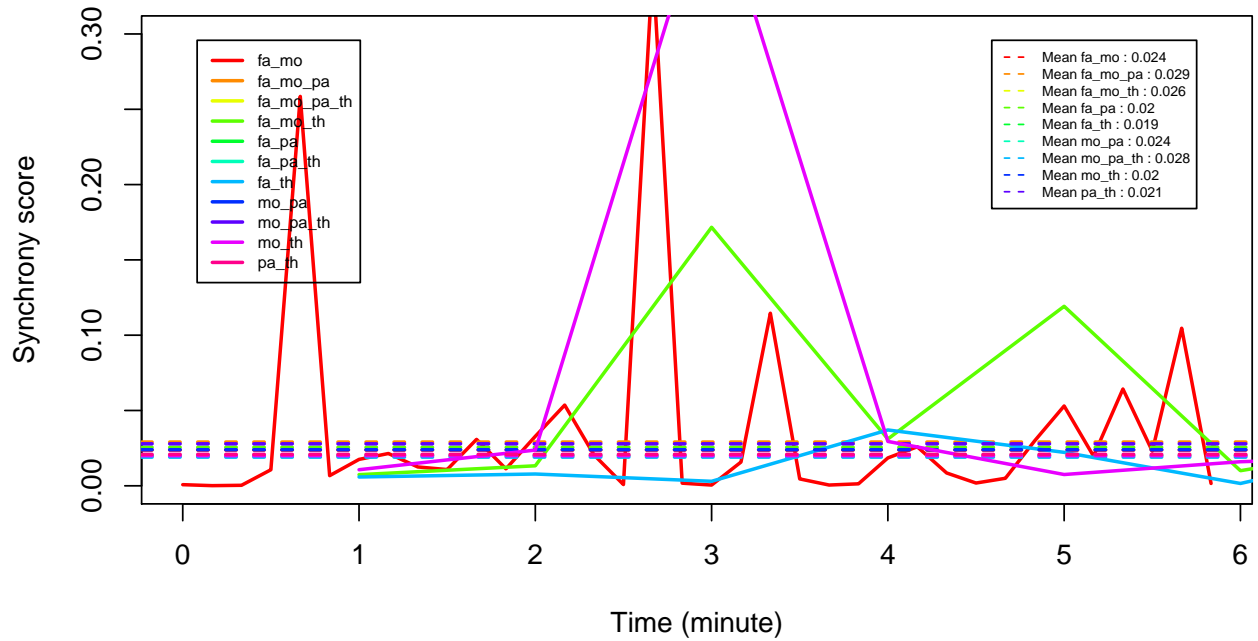
**Synchrony scores for each dyad and for the whole group in F1002B2 video**



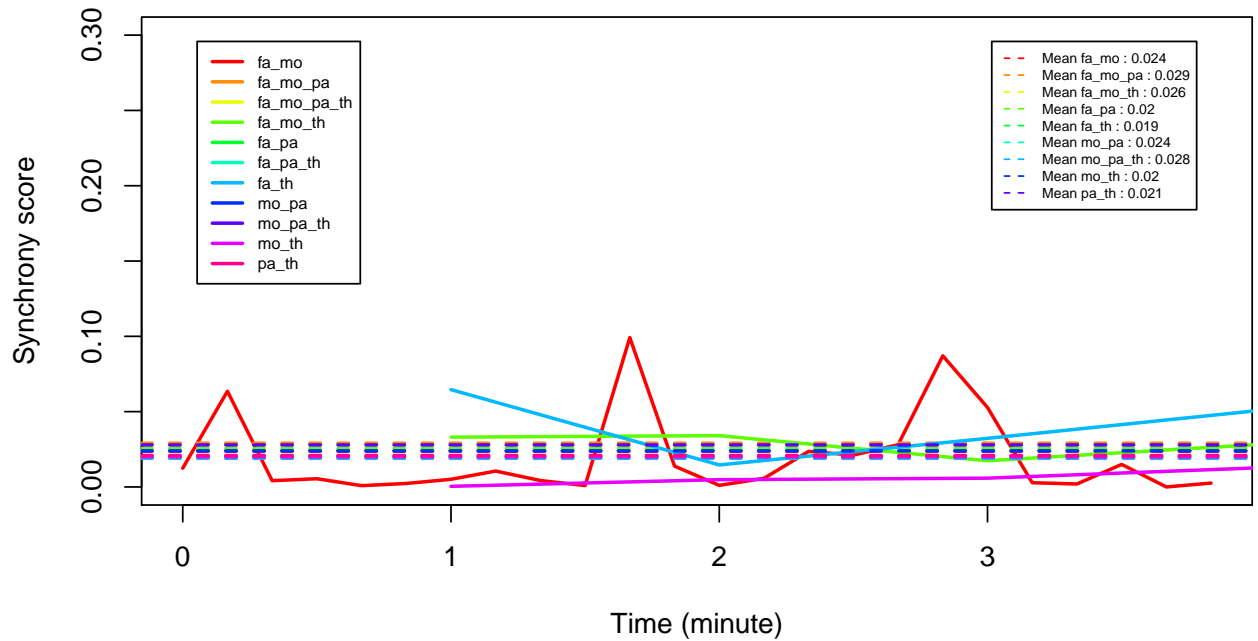
**Synchrony scores for each dyad and for the whole group in F1002C1 video**



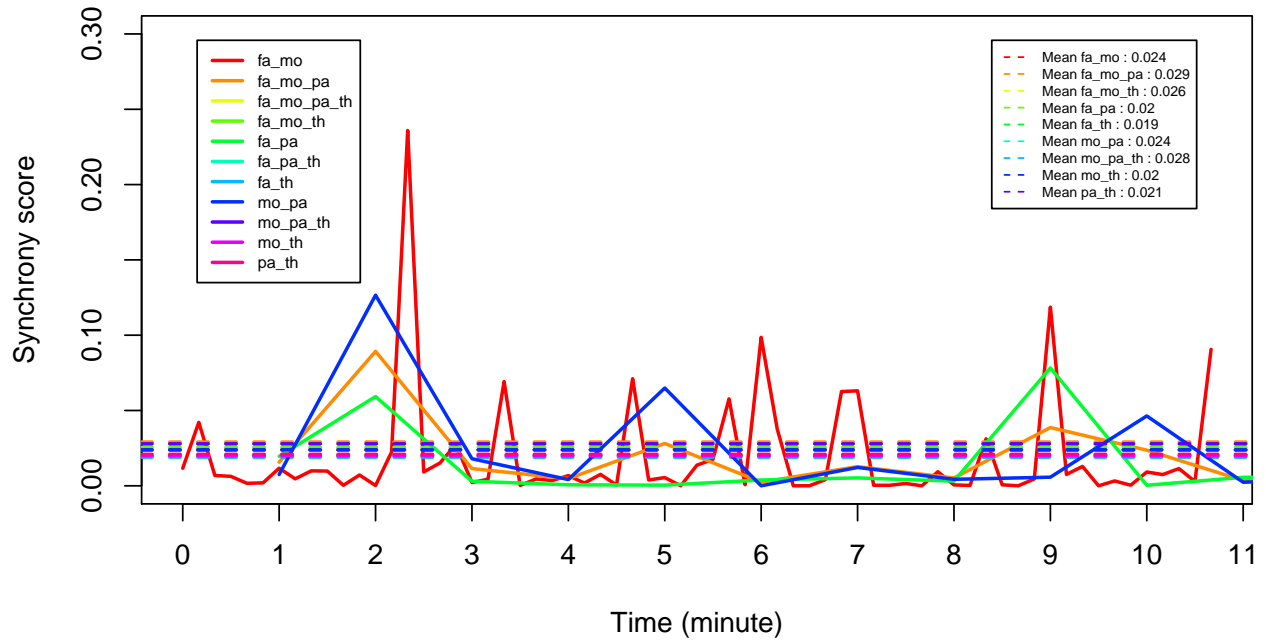
**Synchrony scores for each dyad and for the whole group in F1044C1 video**



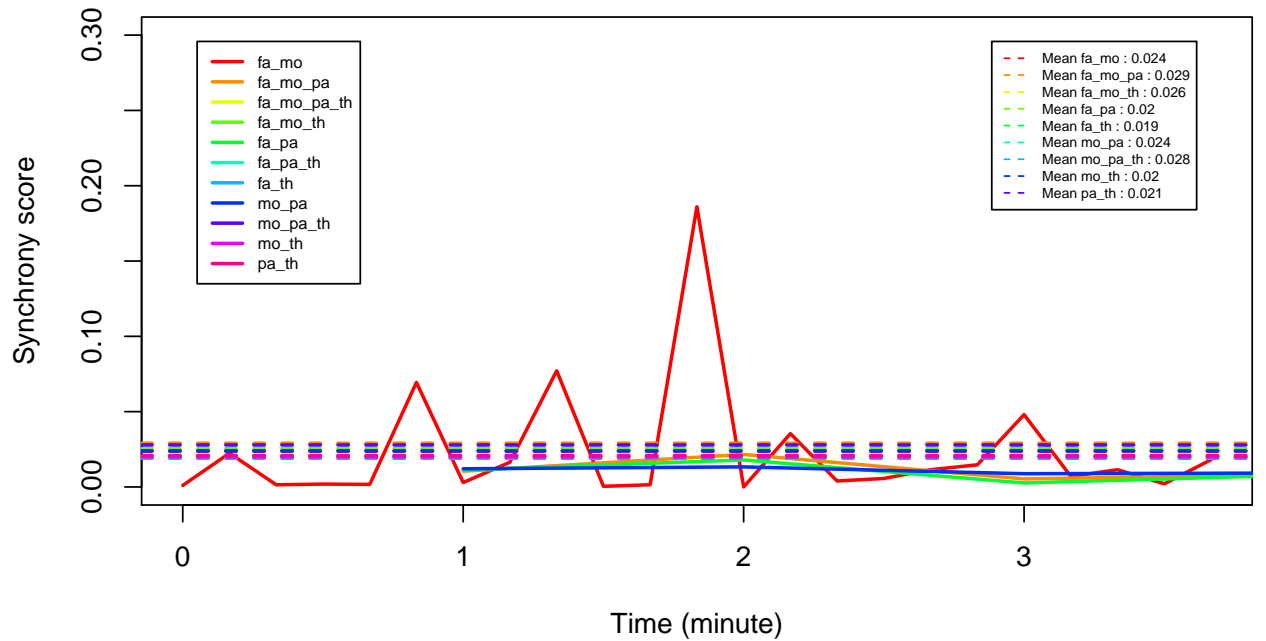
**Synchrony scores for each dyad and for the whole group in F1044C2 video**



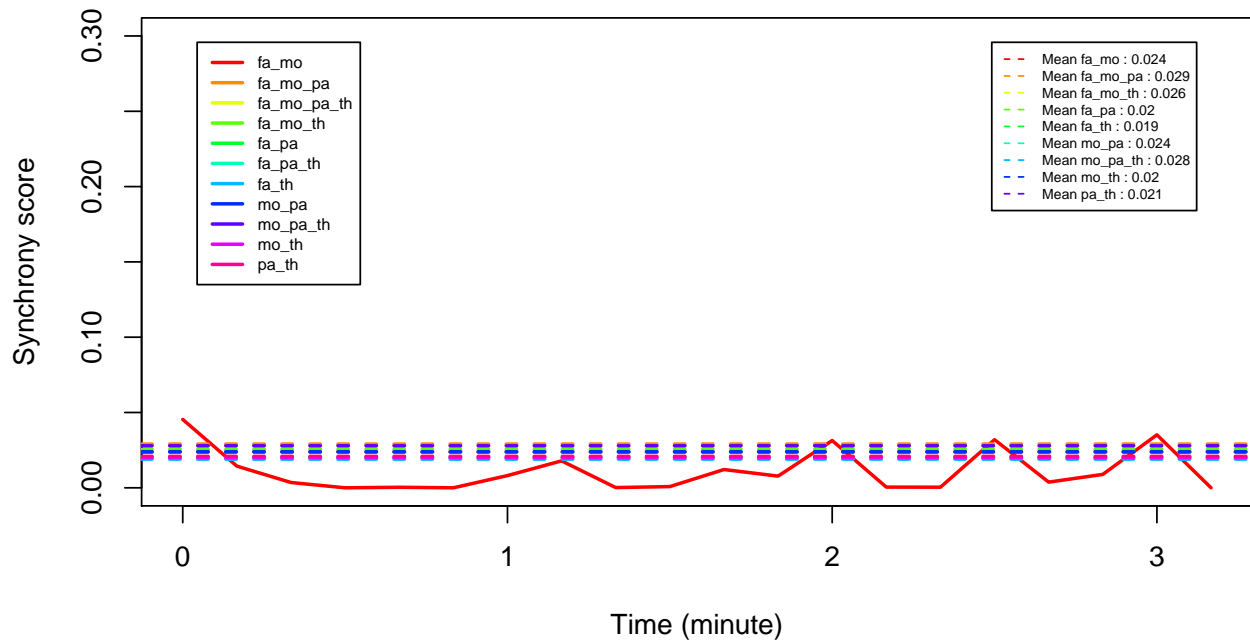
**Synchrony scores for each dyad and for the whole group in F1044D1 video**



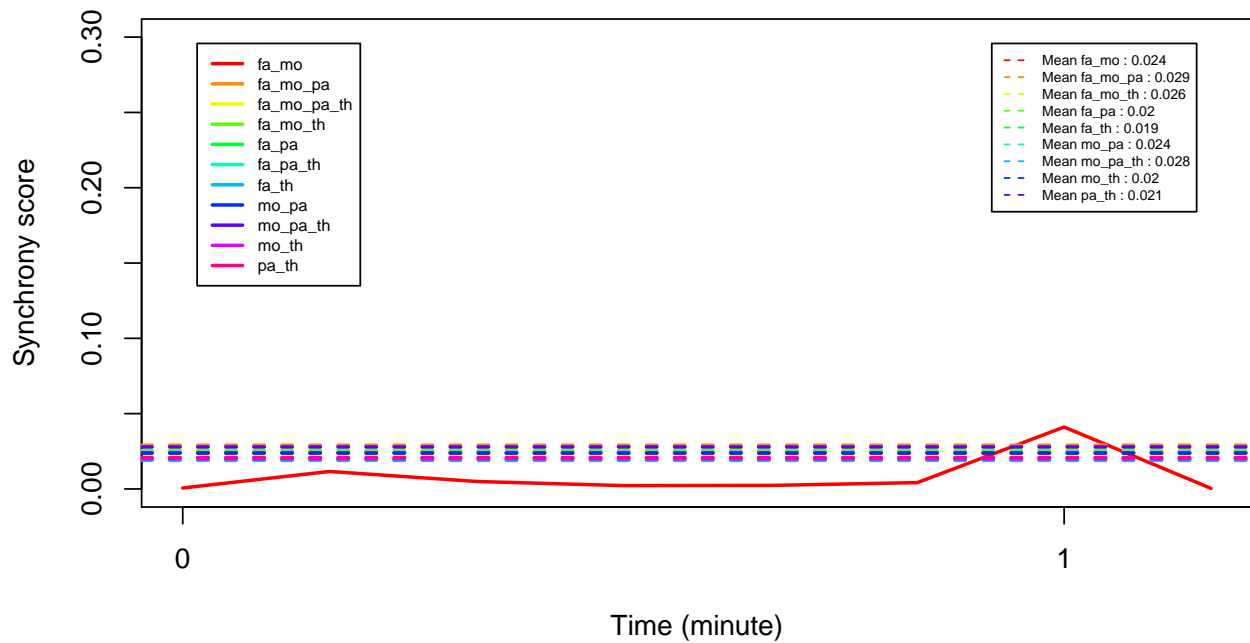
**Synchrony scores for each dyad and for the whole group in F1044D2 video**



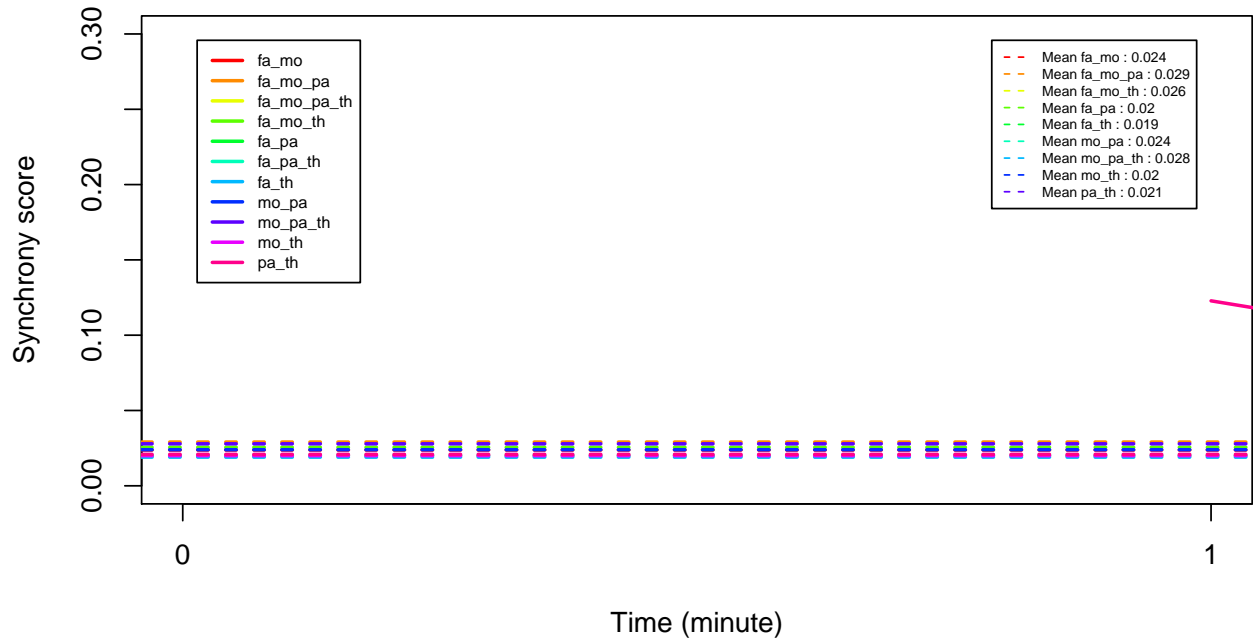
**Synchrony scores for each dyad and for the whole group in F1044E1 video**



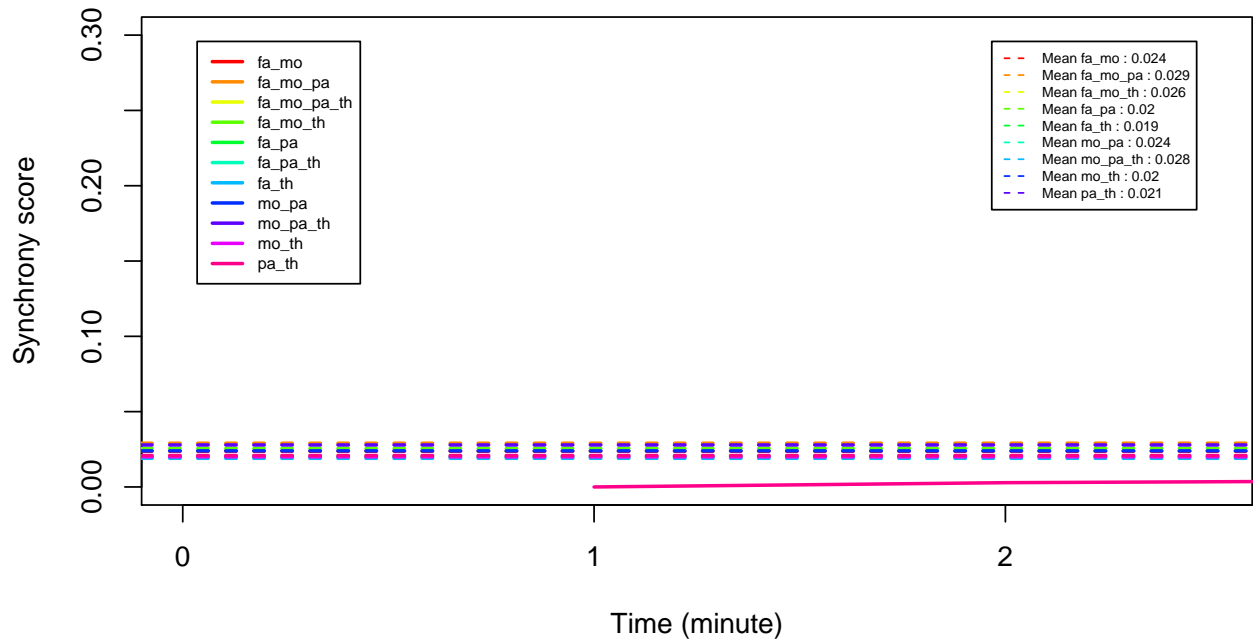
**Synchrony scores for each dyad and for the whole group in F1044E2 video**



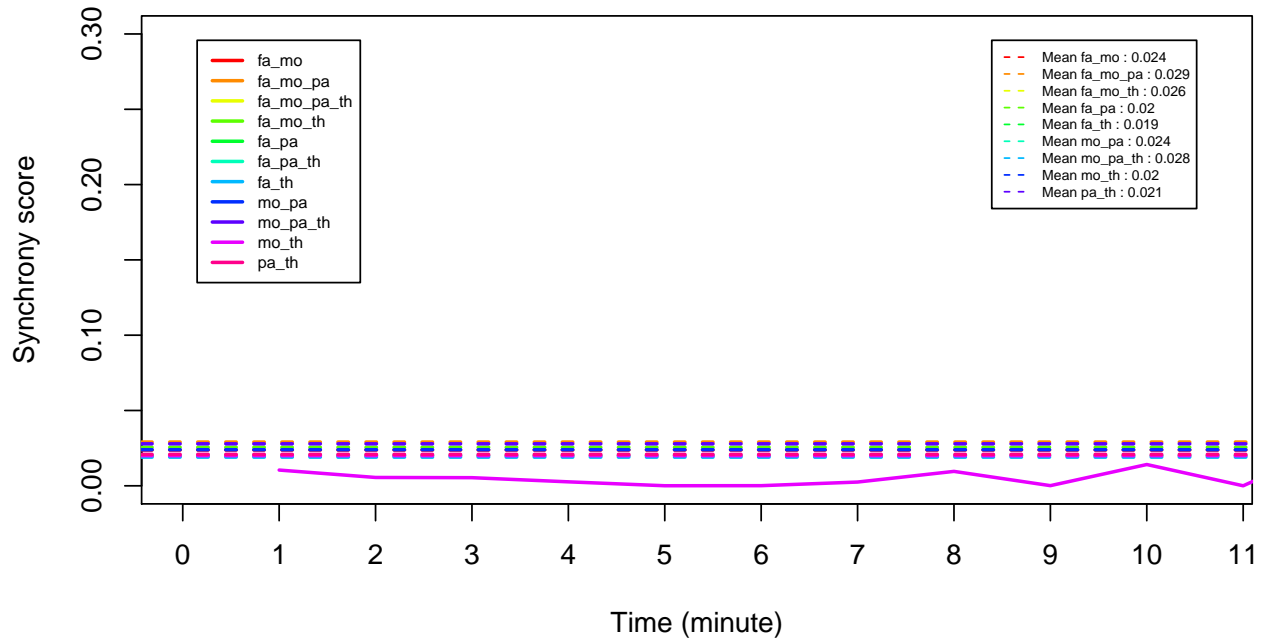
**Synchrony scores for each dyad and for the whole group in F1044F1 video**



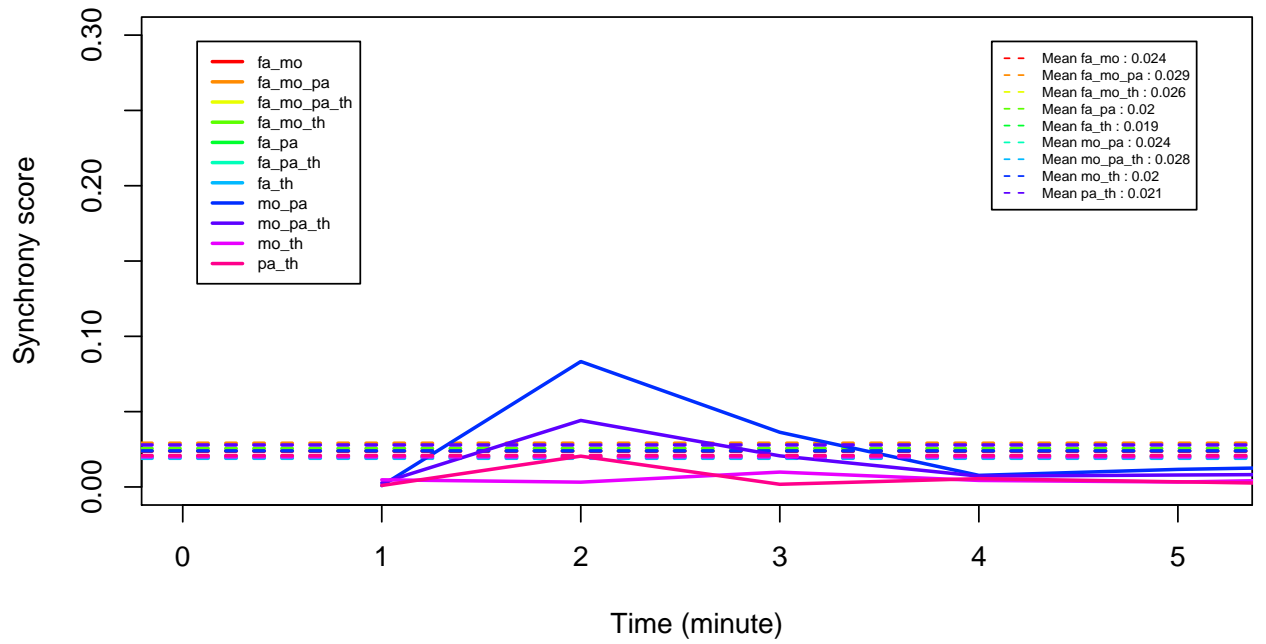
**Synchrony scores for each dyad and for the whole group in F1044F2 video**



**Synchrony scores for each dyad and for the whole group in F1044G video**

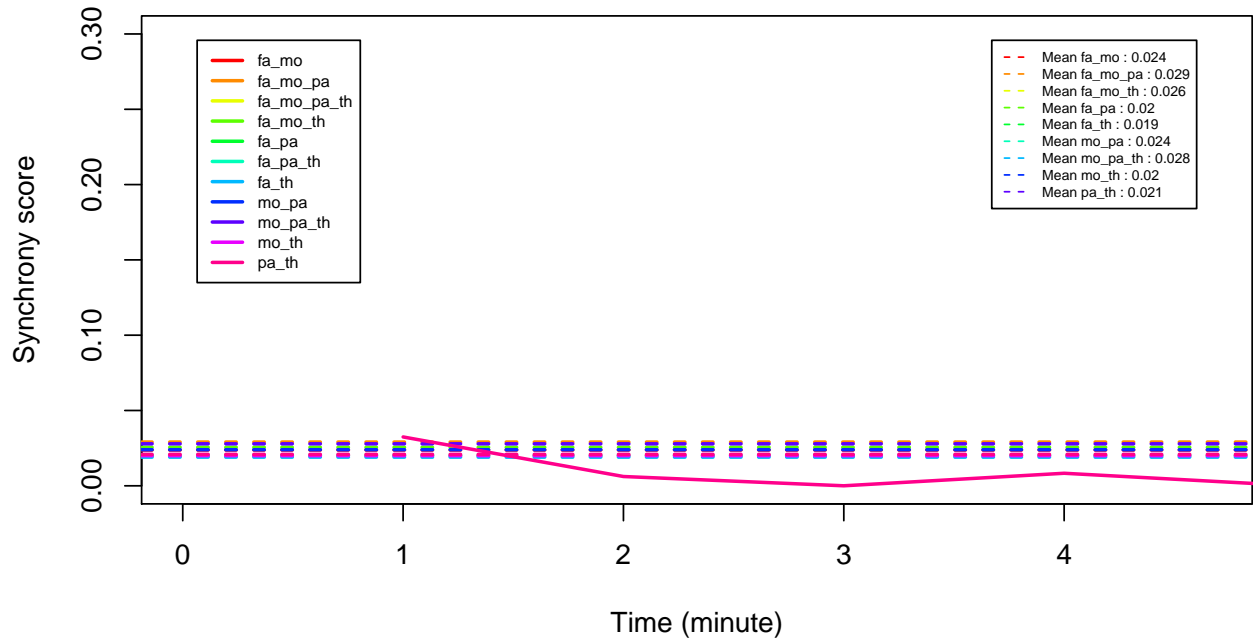


**Synchrony scores for each dyad and for the whole group in F1044H1 video**

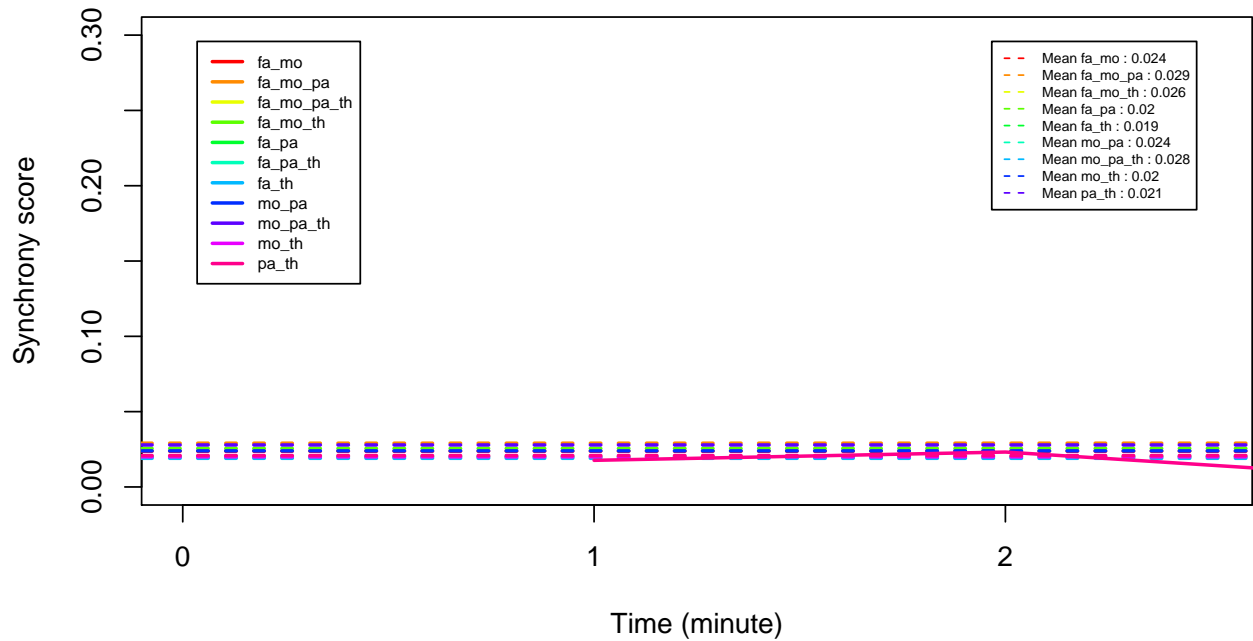




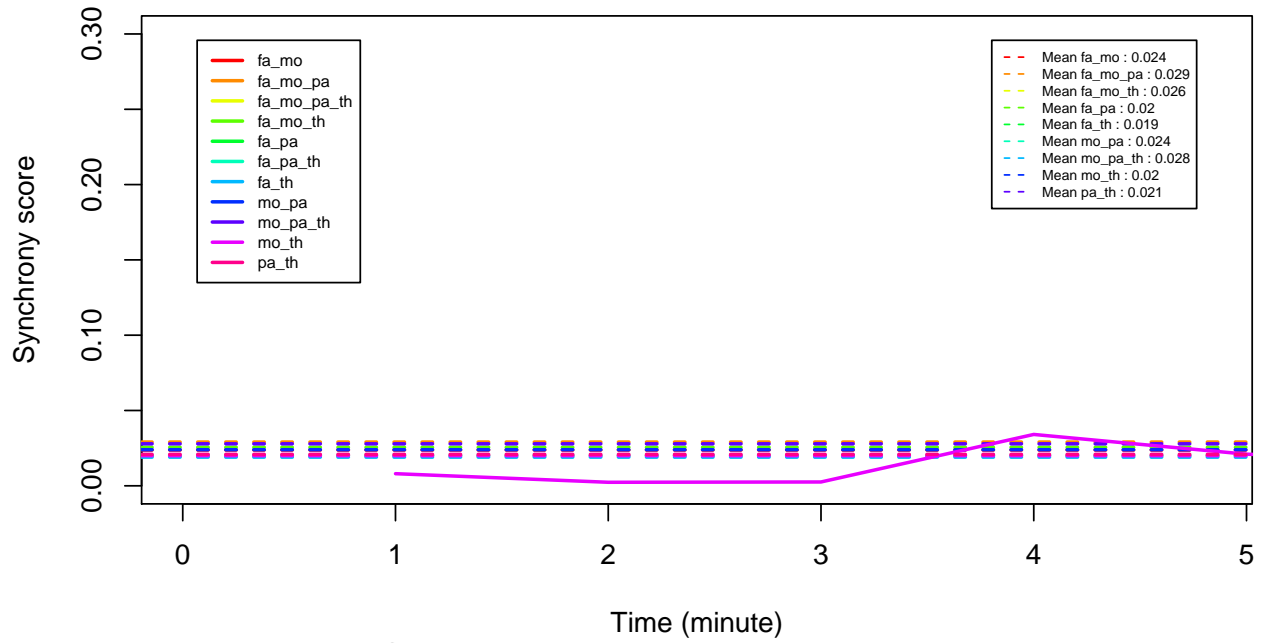
**Synchrony scores for each dyad and for the whole group in F1044H2 video**



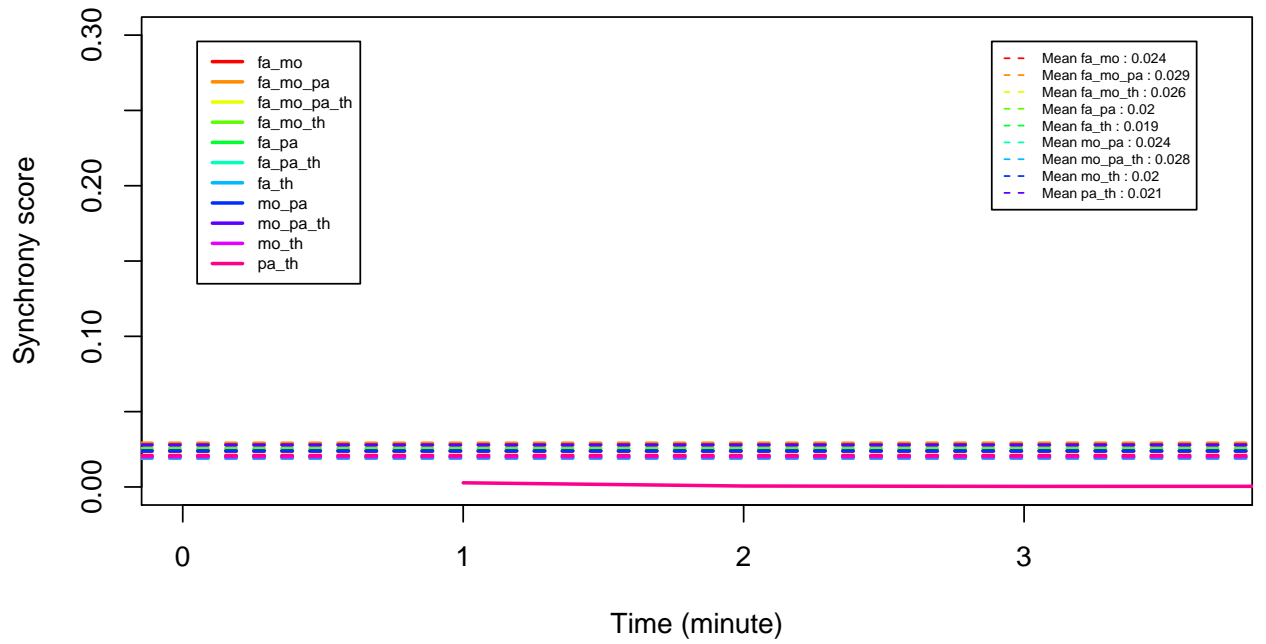
**Synchrony scores for each dyad and for the whole group in F1044I1 video**



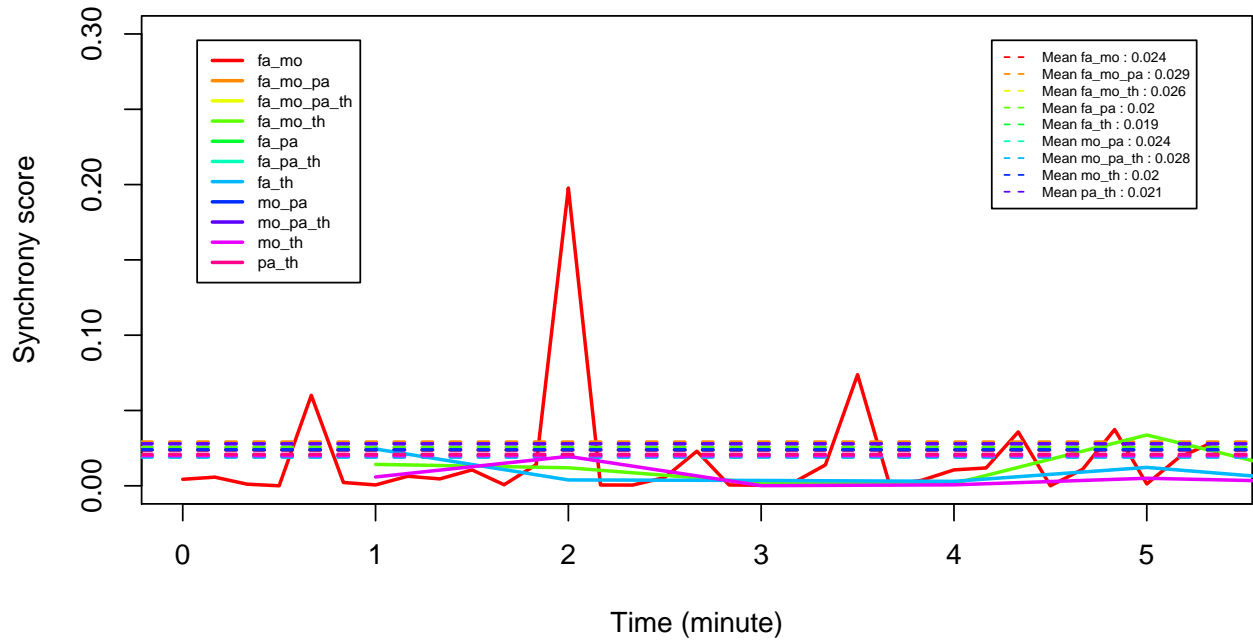
**Synchrony scores for each dyad and for the whole group in F1044I2 video**



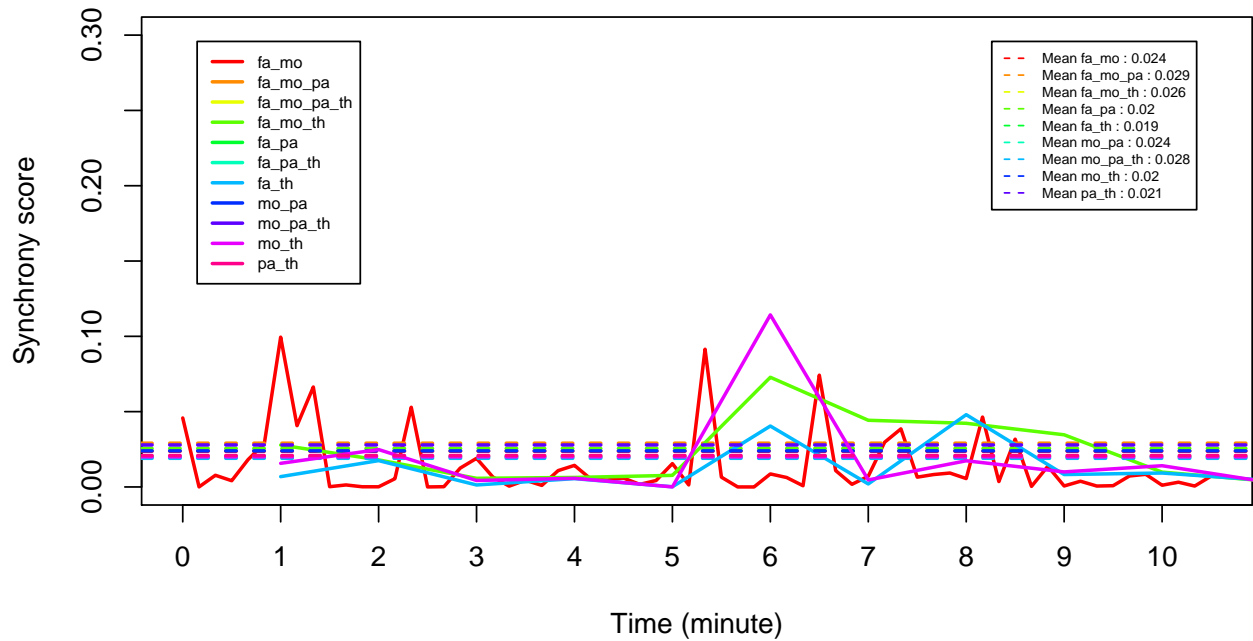
**Synchrony scores for each dyad and for the whole group in F1044L1 video**



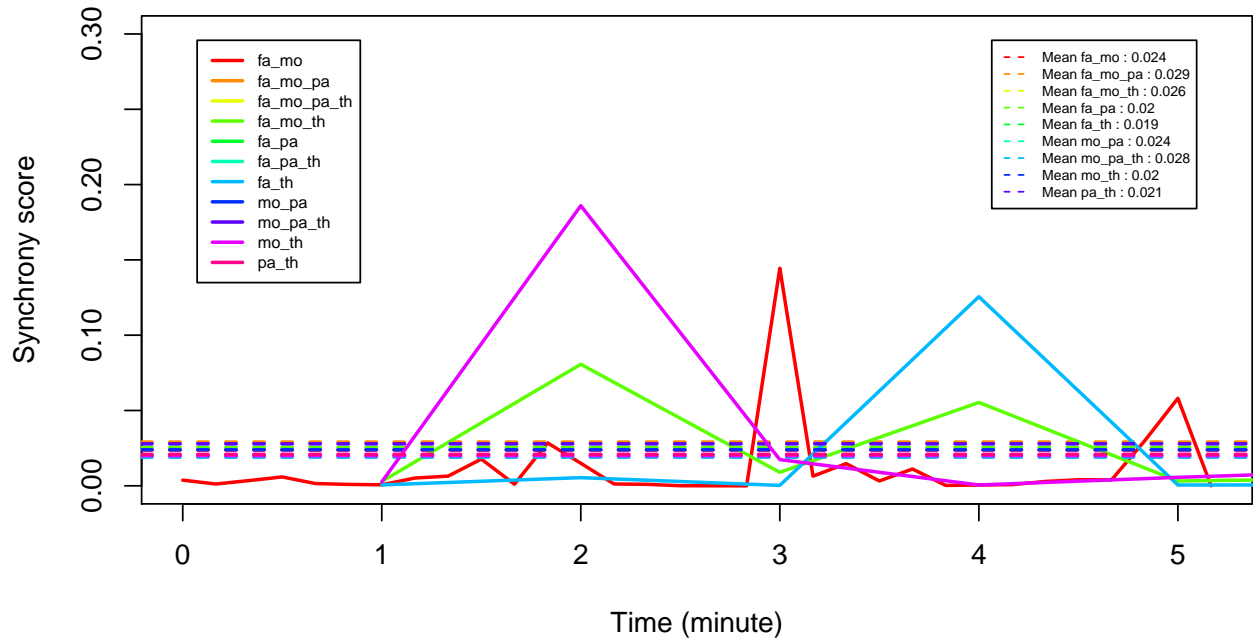
### Synchrony scores for each dyad and for the whole group in F1044L2 video



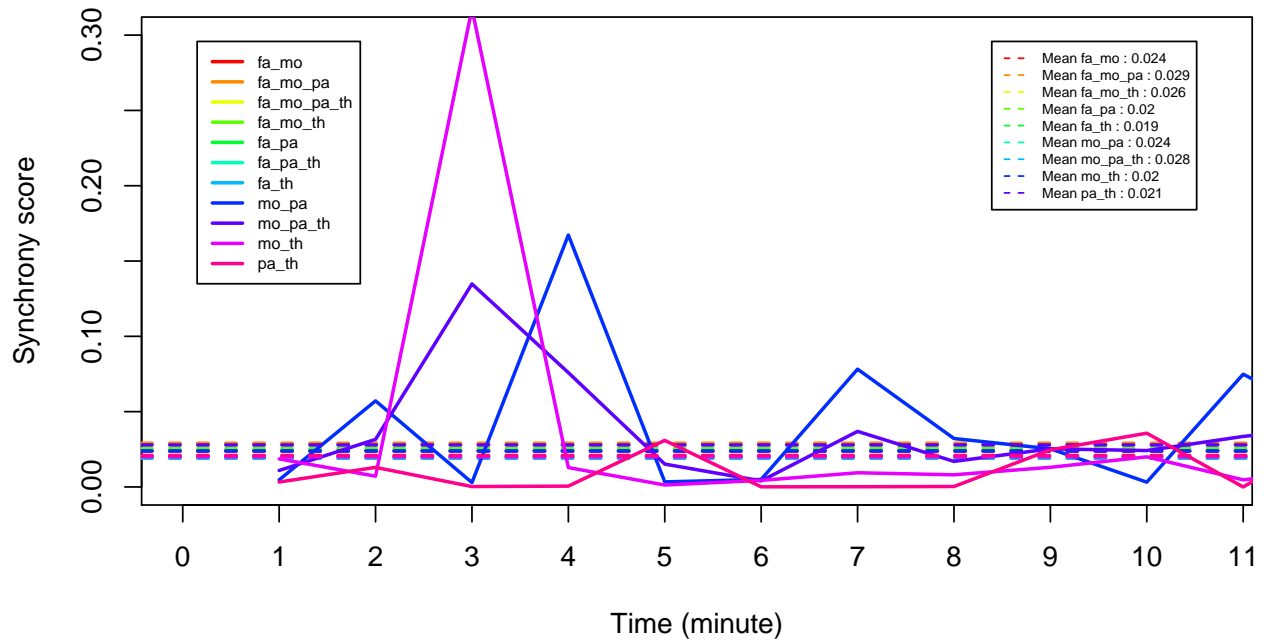
### Synchrony scores for each dyad and for the whole group in F1044M1 video



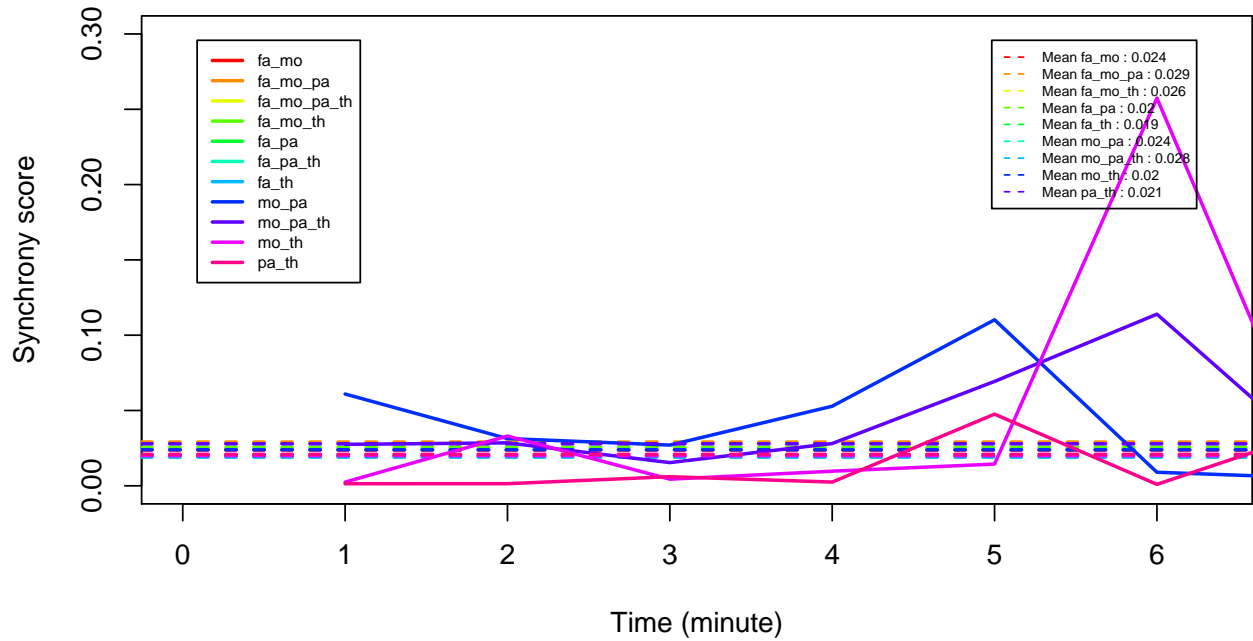
### Synchrony scores for each dyad and for the whole group in F1044M2 video



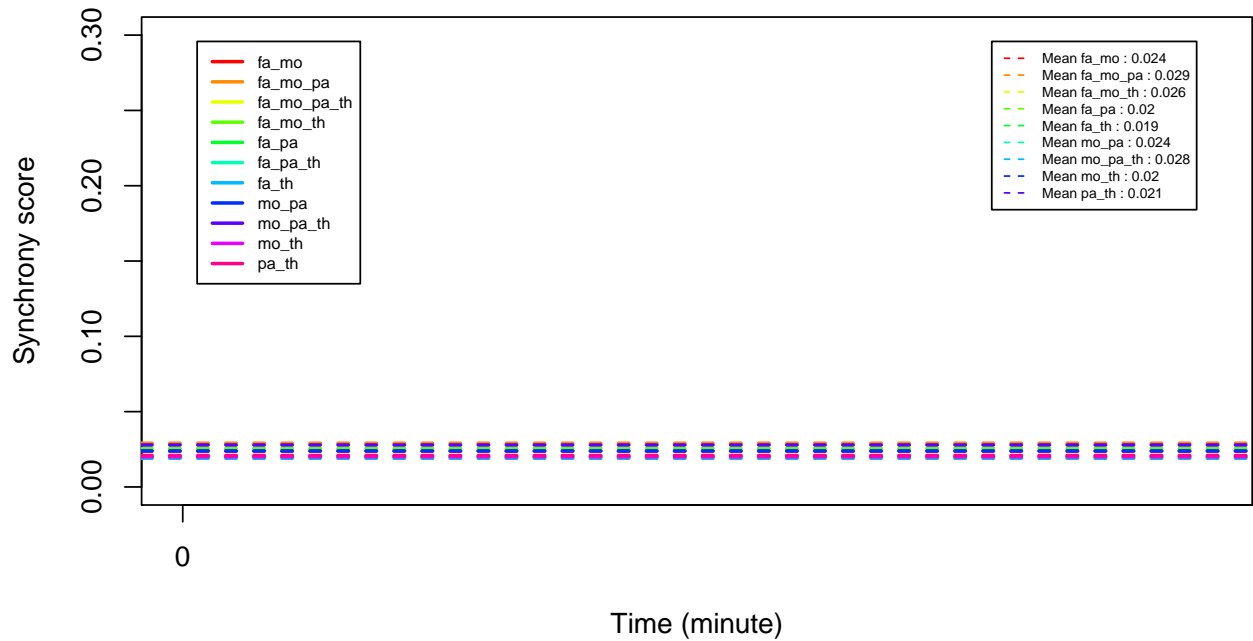
### Synchrony scores for each dyad and for the whole group in F1044N video



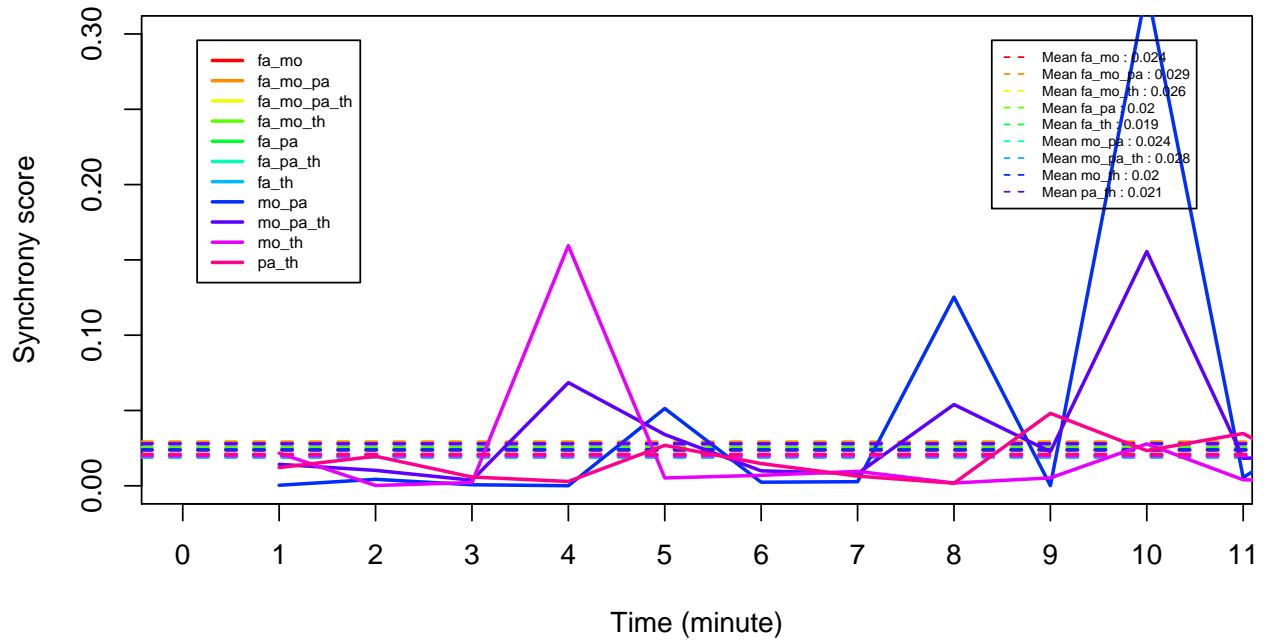
**Synchrony scores for each dyad and for the whole group in F1044O1 video**



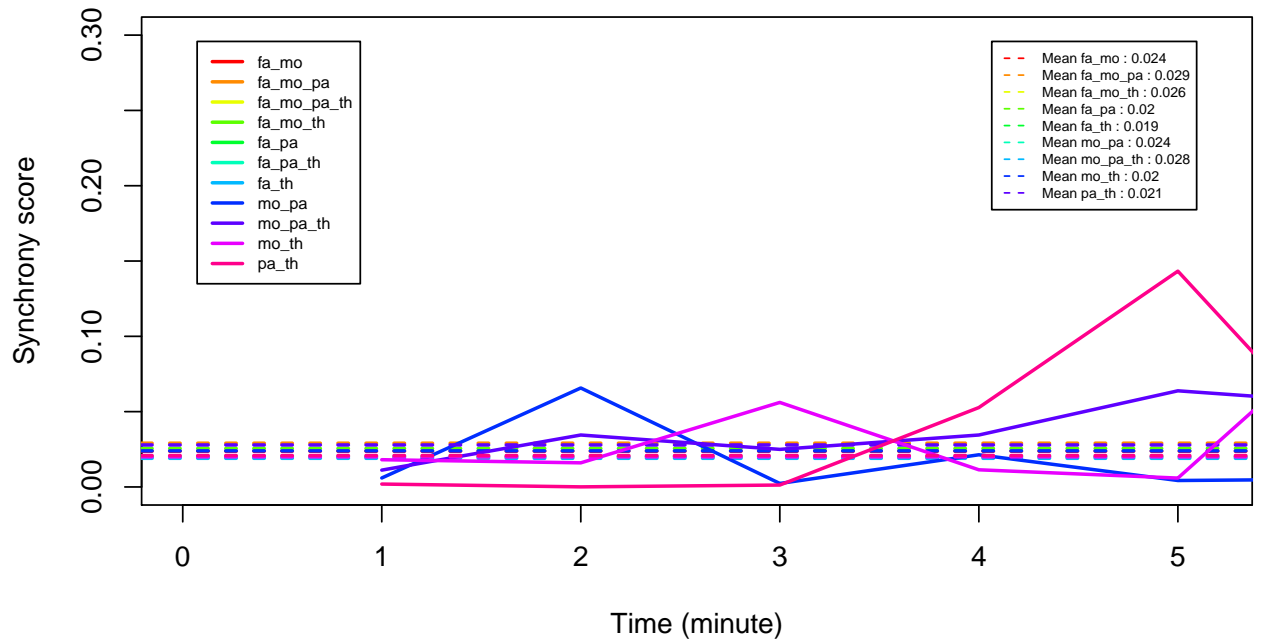
**Synchrony scores for each dyad and for the whole group in F1044O2 video**



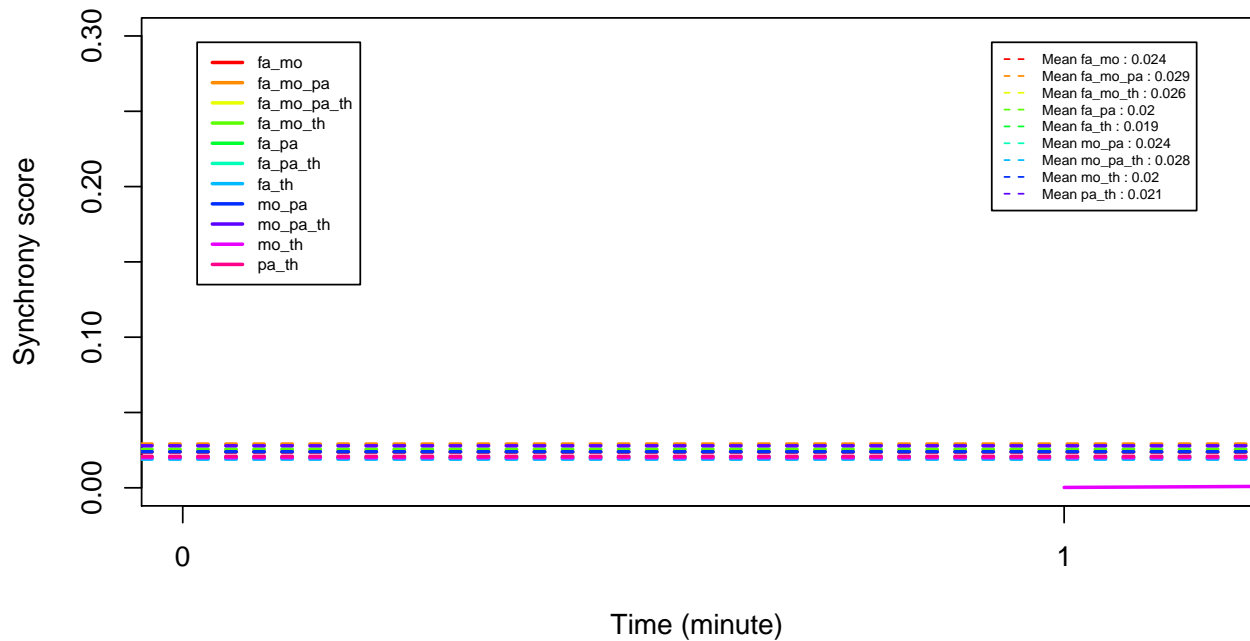
**Synchrony scores for each dyad and for the whole group in F1044P video**



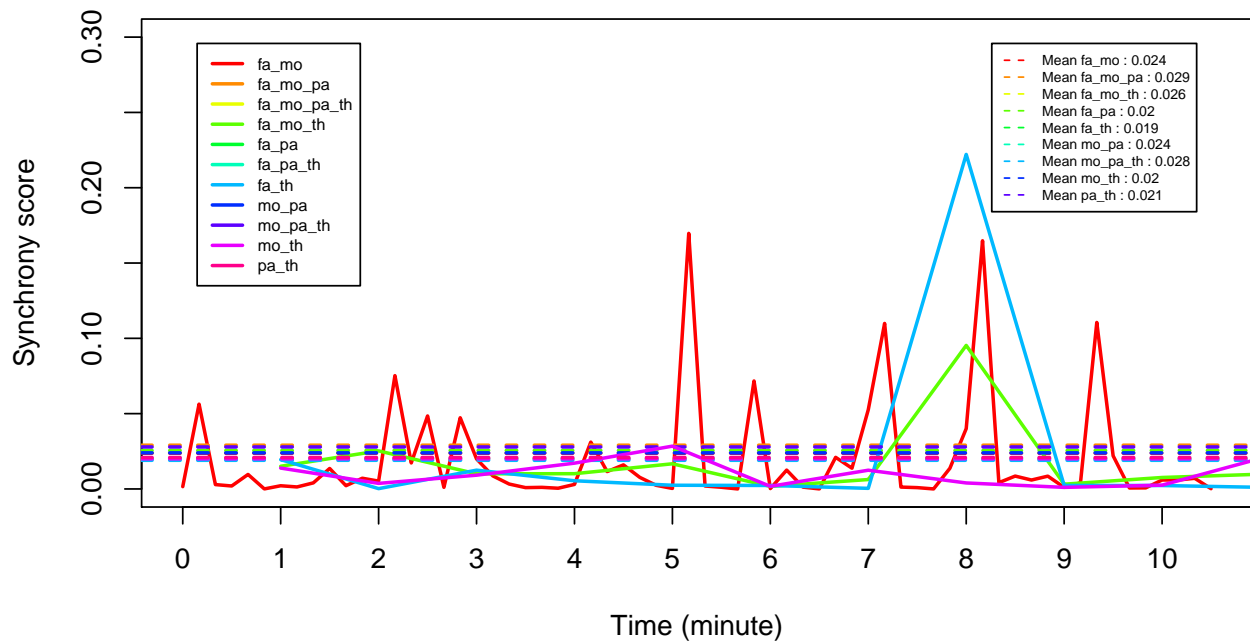
**Synchrony scores for each dyad and for the whole group in F1044Q1 video**



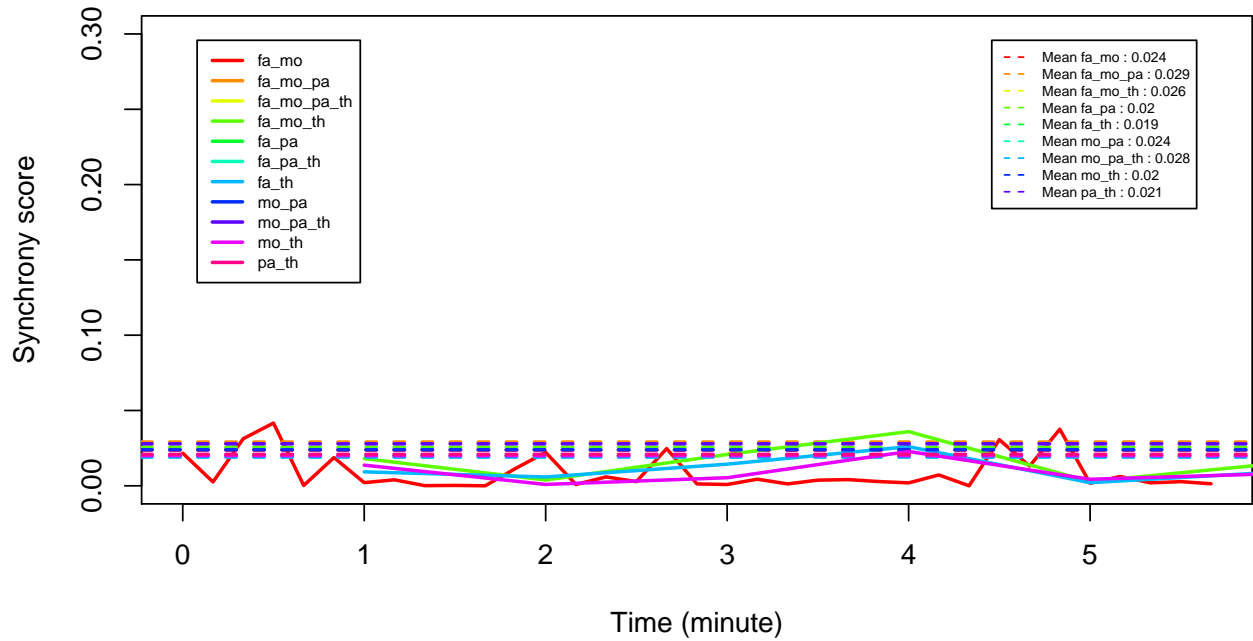
**Synchrony scores for each dyad and for the whole group in F1044Q2 video**



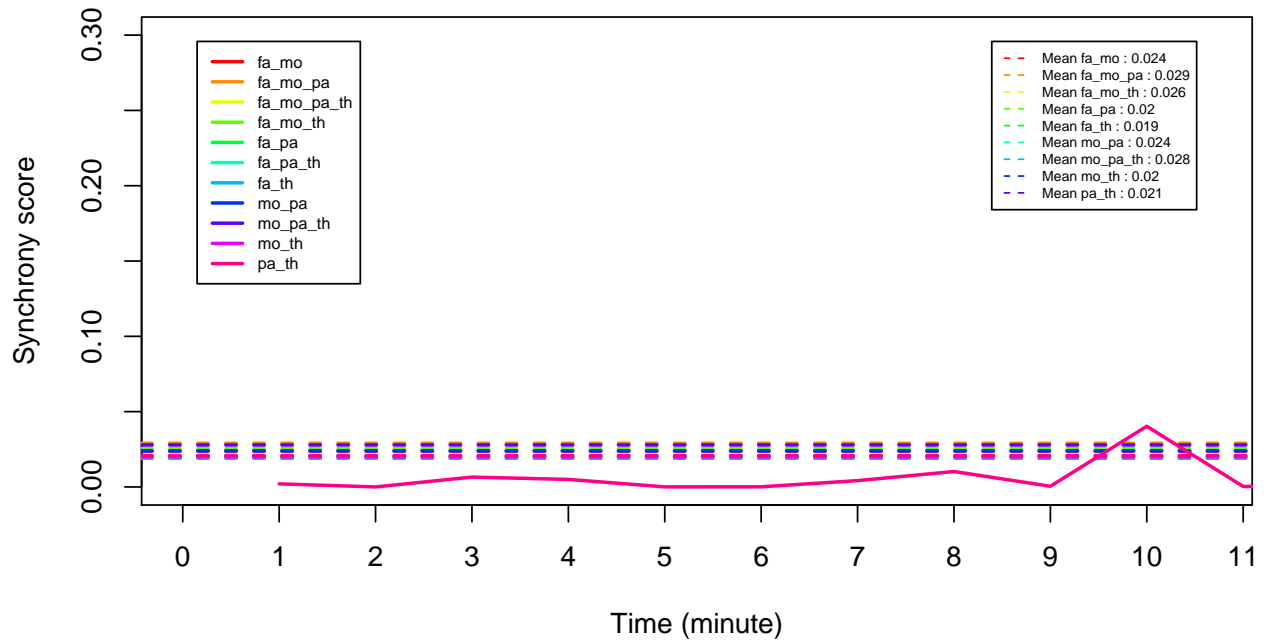
**Synchrony scores for each dyad and for the whole group in F1044R1 video**



**Synchrony scores for each dyad and for the whole group in F1044R2 video**

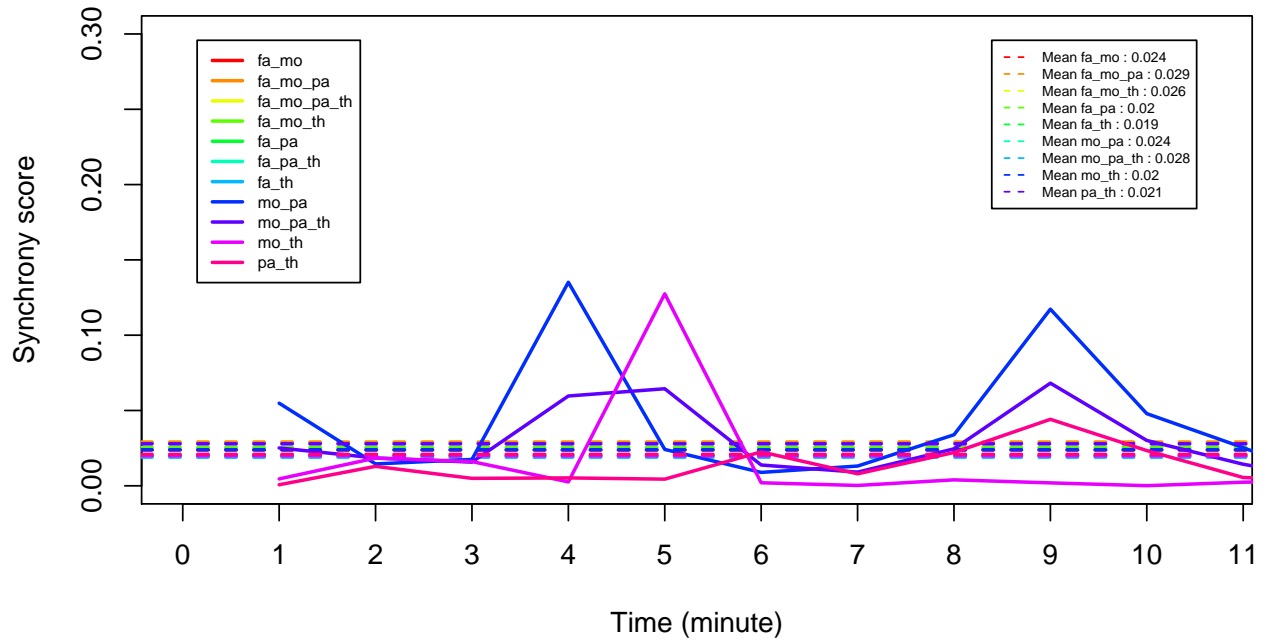


**Synchrony scores for each dyad and for the whole group in F1069A1 video**

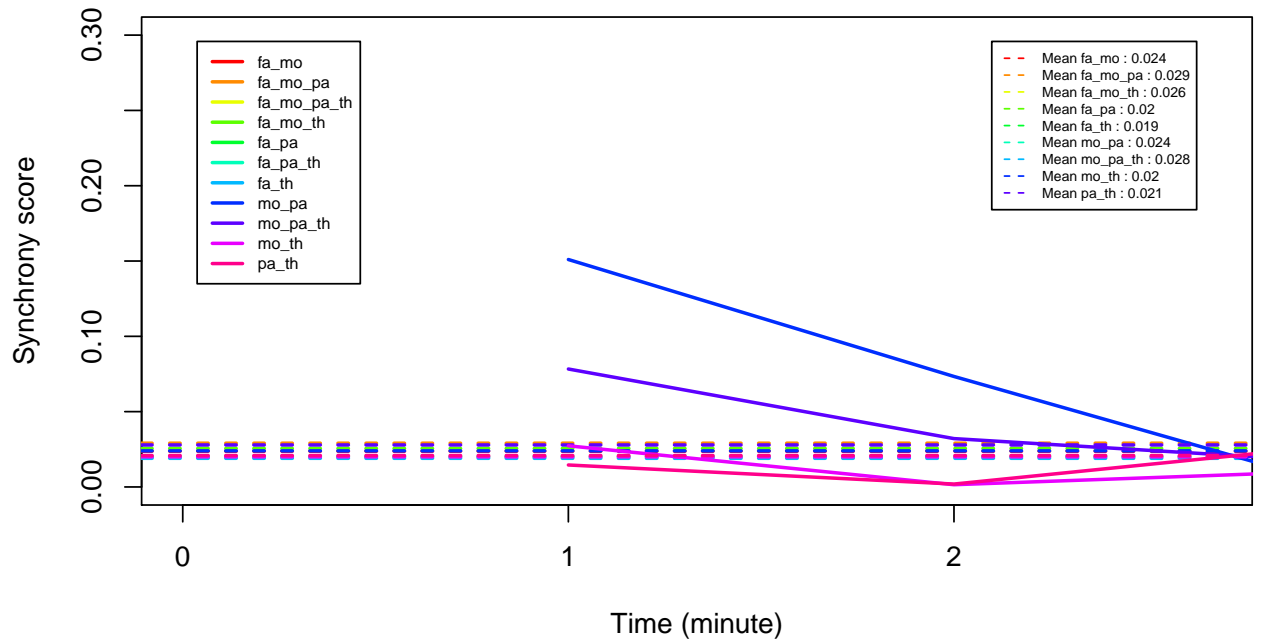




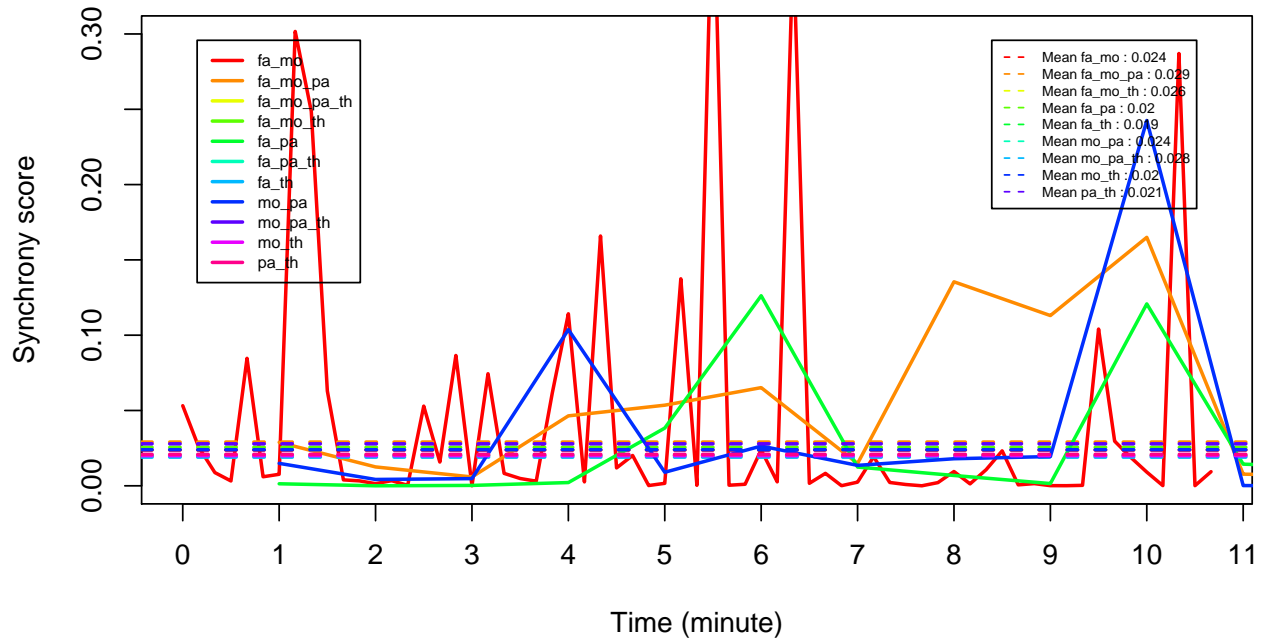
**Synchrony scores for each dyad and for the whole group in F1069B1 video**



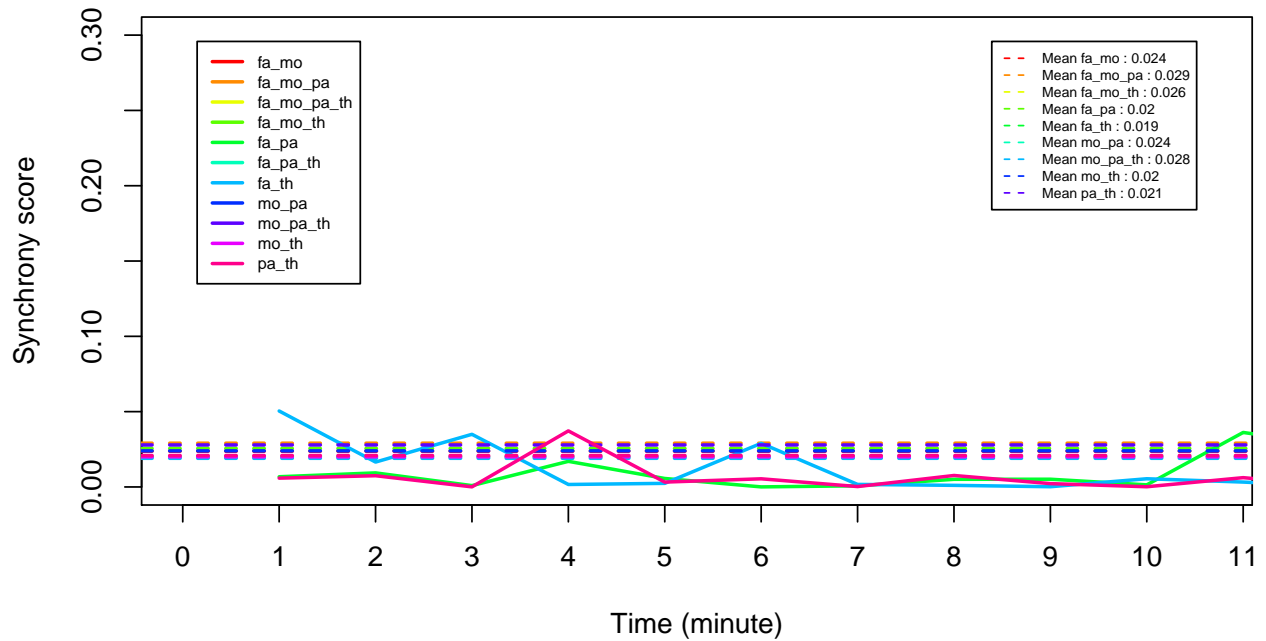
**Synchrony scores for each dyad and for the whole group in F1069B2 video**



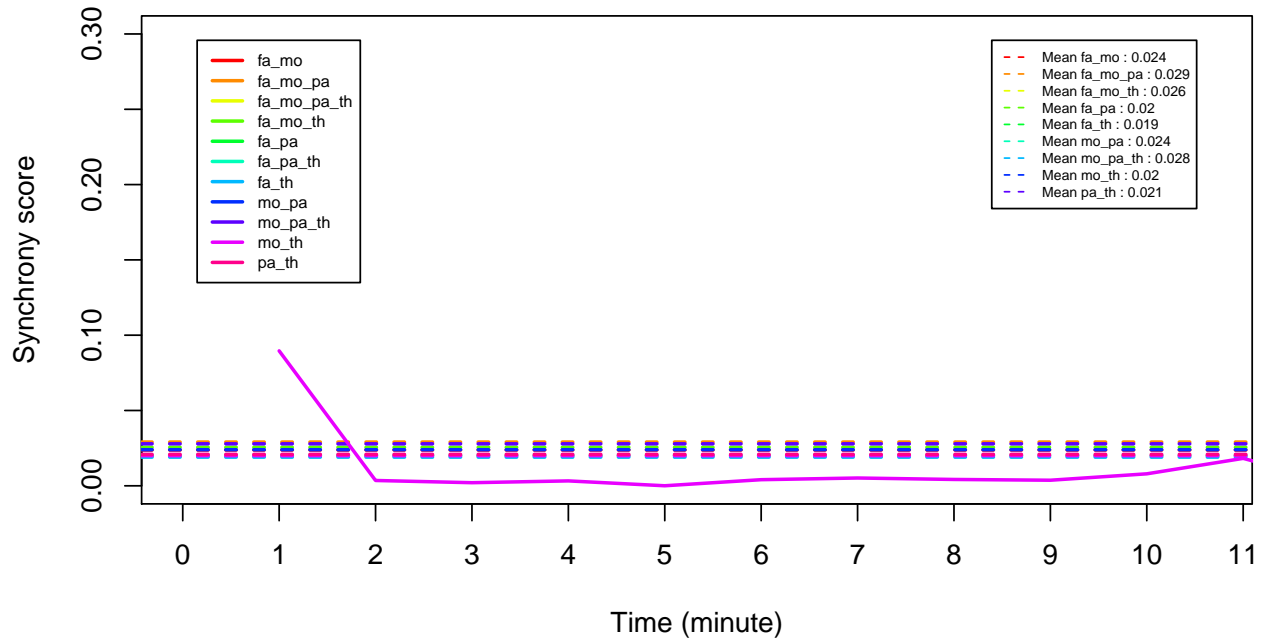
**Synchrony scores for each dyad and for the whole group in F1069C1 video**



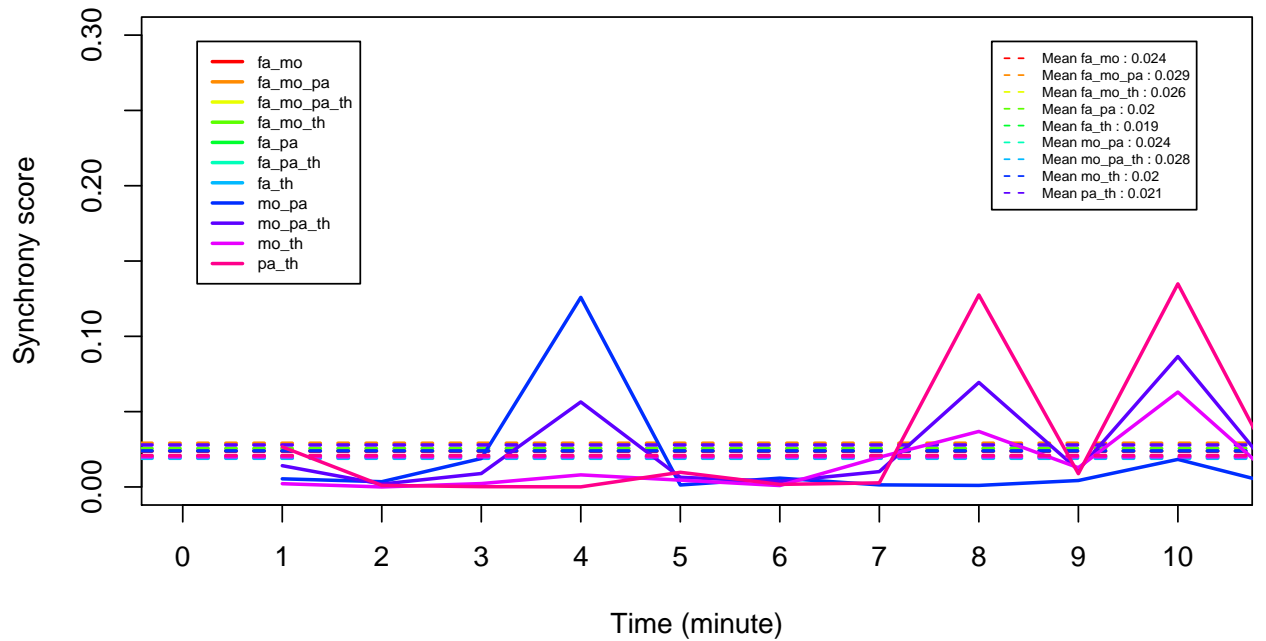
**Synchrony scores for each dyad and for the whole group in F1069D2 video**



**Synchrony scores for each dyad and for the whole group in F1073A1 video**



**Synchrony scores for each dyad and for the whole group in F1073A2 video**



**Synchrony scores(log data) for each dyad, triad and for the whole group**

In legend, mean for all the video.

```

for (i in unique(SSIllogdataFrame$video))
{par(mar=c(4,4,4,3), mfrow=c(1,1))
plot(SSIllogdataFrame[which(SSIllogdataFrame$video==i),]$Time_min,
      SSIllogdataFrame[which(SSIllogdataFrame$video==i),]$SSI_fa_mo,
      type="l", ylim=c(0, 0.3), col=rainbow(4)[1],
      main=paste("Synchrony scores for each dyad and for \n the whole group in", i, "video"),
      xlab = "Time (minute)", ylab="Synchrony score", lwd=2,
xaxp=c(0,length(SSIllogdataFrame$Time_min), length(SSIllogdataFrame$Time_min)))
abline(h=mean(SSIllogdataFrame$SSI_fa_mo, na.rm=TRUE), col=rainbow(11)[1], lwd=2, lty=2)

lines(SSIllogdataFrame[which(SSIllogdataFrame$video==i),]$SSI_fa_mo_pa, col=rainbow(11)[2], lwd=2)
abline(h= mean(SSIllogdataFrame$SSI_fa_mo_pa, na.rm=TRUE), col=rainbow(11)[2], lwd=2, lty=2)

# lines(SSIllogdataFrame[which(SSIllogdataFrame$video==i),]$SSI_fa_mo_pa_th, col=rainbow(11)[3], lwd=2)
# abline(h= mean(SSIllogdataFrame$SSI_fa_mo_pa_th, na.rm=TRUE), col=rainbow(11)[3], lwd=2, lty=2)

lines(SSIllogdataFrame[which(SSIllogdataFrame$video==i),]$SSI_fa_mo_th, col=rainbow(11)[4], lwd=2)
abline(h= mean(SSIllogdataFrame$SSI_fa_mo_th, na.rm=TRUE), col=rainbow(11)[4], lwd=2, lty=2)

lines(SSIllogdataFrame[which(SSIllogdataFrame$video==i),]$SSI_fa_pa, col=rainbow(11)[5], lwd=2)
abline(h= mean(SSIllogdataFrame$SSI_fa_pa, na.rm=TRUE), col=rainbow(11)[5], lwd=2, lty=2)

# lines(SSIllogdataFrame[which(SSIllogdataFrame$video==i),]$SSI_fa_pa_th, col=rainbow(11)[6], lwd=2)
# abline(h= mean(SSIllogdataFrame$SSI_fa_pa_th, na.rm=TRUE), col=rainbow(11)[6], lwd=2, lty=2)

lines(SSIllogdataFrame[which(SSIllogdataFrame$video==i),]$SSI_fa_th, col=rainbow(11)[7], lwd=2)
abline(h= mean(SSIllogdataFrame$SSI_fa_th, na.rm=TRUE), col=rainbow(11)[7], lwd=2, lty=2)

lines(SSIllogdataFrame[which(SSIllogdataFrame$video==i),]$SSI_mo_pa, col=rainbow(11)[8], lwd=2)
abline(h= mean(SSIllogdataFrame$SSI_mo_pa, na.rm=TRUE), col=rainbow(11)[8], lwd=2, lty=2)

lines(SSIllogdataFrame[which(SSIllogdataFrame$video==i),]$SSI_mo_pa_th, col=rainbow(11)[9], lwd=2)
abline(h= mean(SSIllogdataFrame$SSI_mo_pa_th, na.rm=TRUE), col=rainbow(11)[9], lwd=2, lty=2)
lines(SSIllogdataFrame[which(SSIllogdataFrame$video==i),]$SSI_mo_th, col=rainbow(11)[10], lwd=2)
abline(h= mean(SSIllogdataFrame$SSI_mo_th, na.rm=TRUE), col=rainbow(11)[10], lwd=2, lty=2)

lines(SSIllogdataFrame[which(SSIllogdataFrame$video==i),]$SSI_pa_th, col=rainbow(11)[11], lwd=2)
abline(h= mean(SSIllogdataFrame$SSI_pa_th, na.rm=TRUE), col=rainbow(11)[11], lwd=2, lty=2)

legend("topleft", inset=.05, c("fa_mo", "fa_mo_pa", "fa_mo_pa_th",
"fa_mo_th", "fa_pa", "fa_pa_th", "fa_th",
"mo_pa", "mo_pa_th", "mo_th", "pa_th"),
col=rainbow(11), cex=0.6, lwd=2)

legend("topright", inset=.05, c(paste ("Mean fa_mo :",
round(mean(SSIllogdataFrame$SSI_fa_mo, na.rm=TRUE),3)),
paste ("Mean fa_mo_pa :", round(mean(SSIllogdataFrame$SSI_fa_mo_pa,na.rm=TRUE),3)),
# paste ("Mean fa_mo_pa_th :", #round(mean(SSIllogdataFrame$SSI_fa_mo_pa_th),3)),
paste ("Mean fa_mo_th :", round(mean(SSIllogdataFrame$SSI_fa_mo_th,na.rm=TRUE),3)),
paste ("Mean fa_pa :", round(mean(SSIllogdataFrame$SSI_fa_pa, na.rm=TRUE),3)),
# paste ("Mean fa_pa_th :", round(mean(SSIllogdataFrame$SSI_fa_pa_th,na.rm=TRUE),3)),
paste ("Mean fa_th :", round(mean(SSIllogdataFrame$SSI_fa_th,na.rm=TRUE),3)),
paste ("Mean mo_pa :", round(mean(SSIllogdataFrame$SSI_mo_pa,na.rm=TRUE),3)),

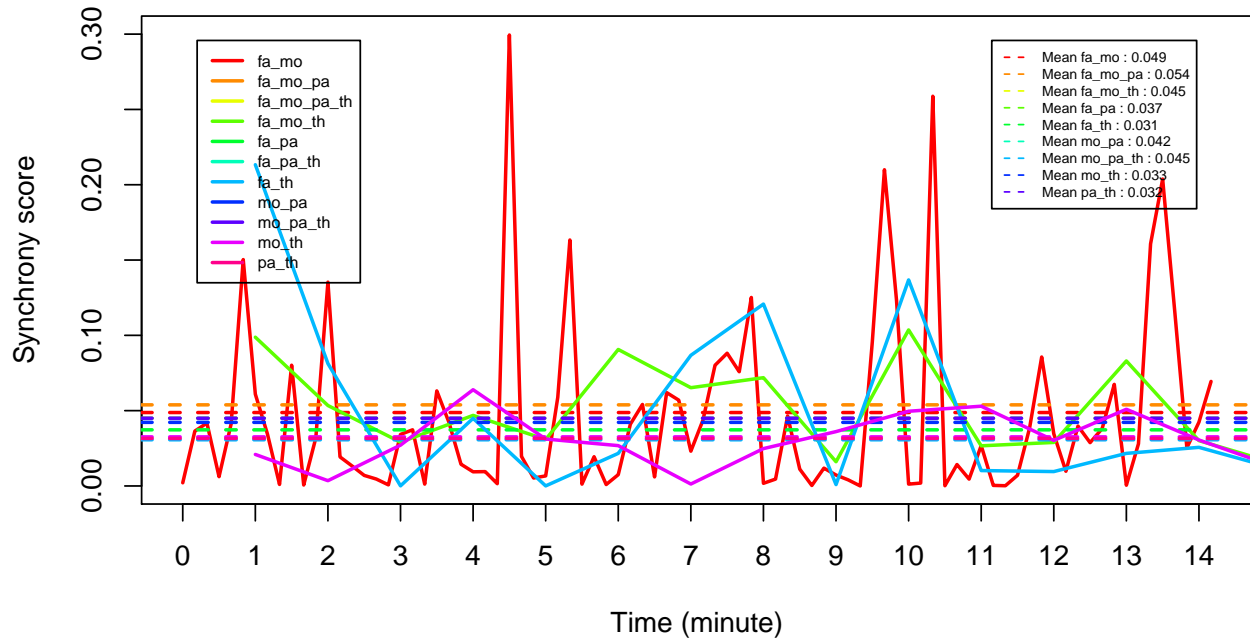
```

```

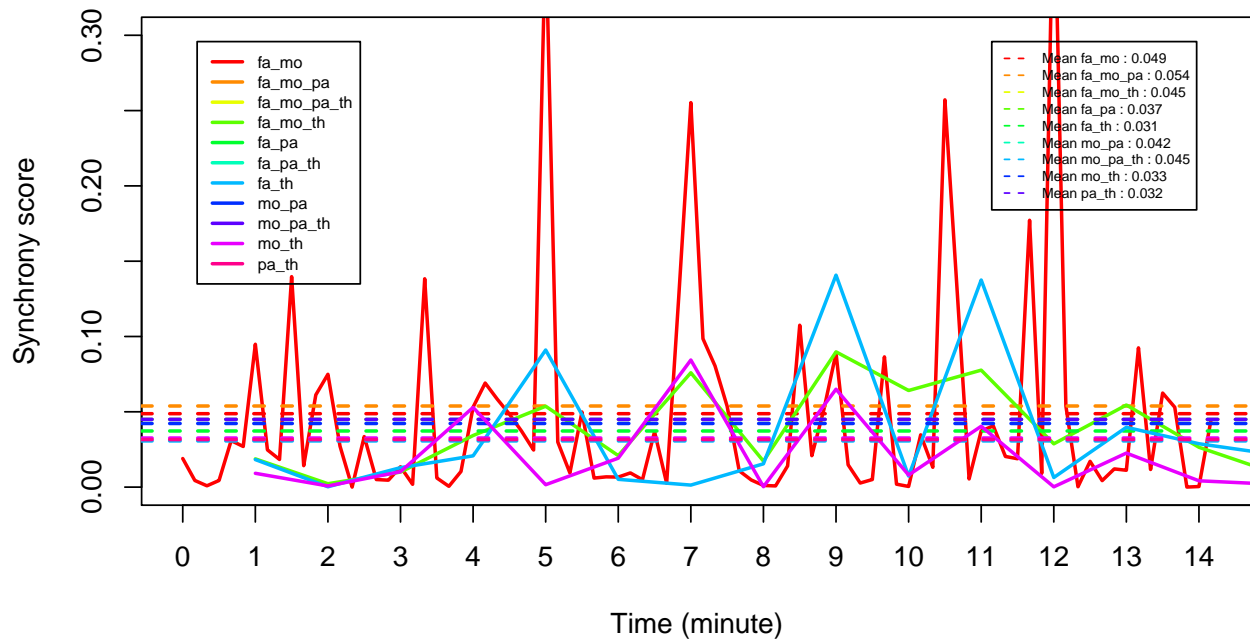
paste ("Mean mo_pa_th :", round(mean(SSILogdataFrame$SSI_mo_pa_th,na.rm=TRUE),3)),
paste ("Mean mo_th :", round(mean(SSILogdataFrame$SSI_mo_th,na.rm=TRUE),3)),
paste ("Mean pa_th :", round(mean(SSILogdataFrame$SSI_pa_th,na.rm=TRUE),3))),
col=rainbow(11), cex=0.5, lty=2, lwd=1)}

```

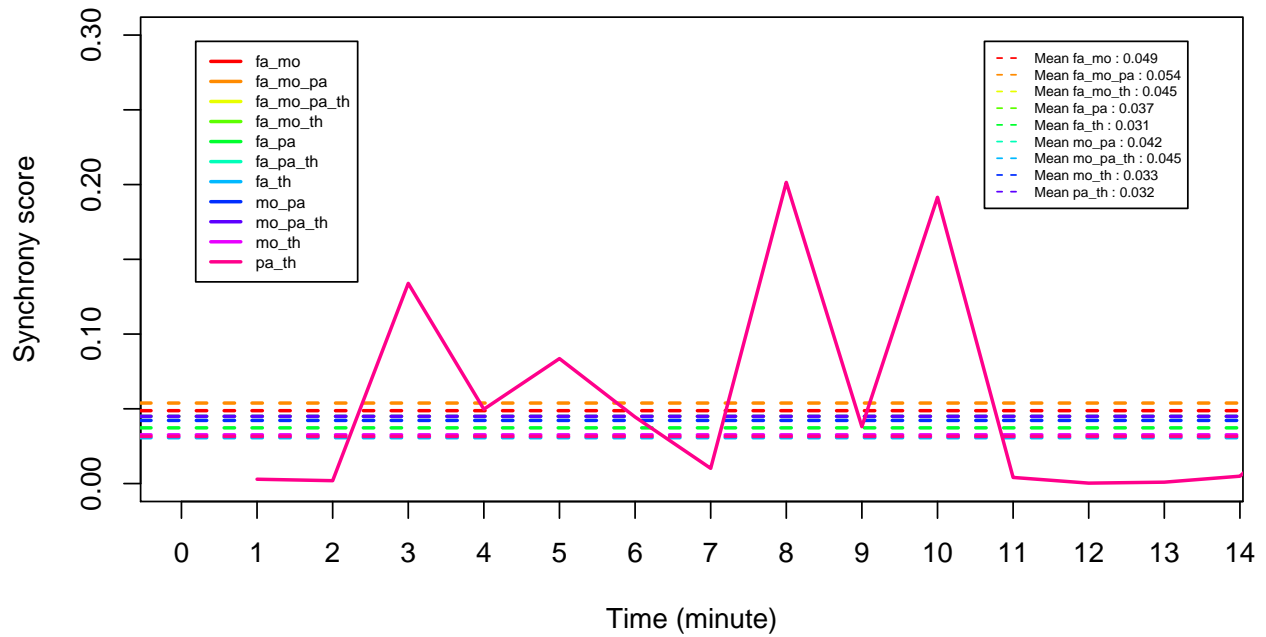
**Synchrony scores for each dyad and for the whole group in F1002A1 video**



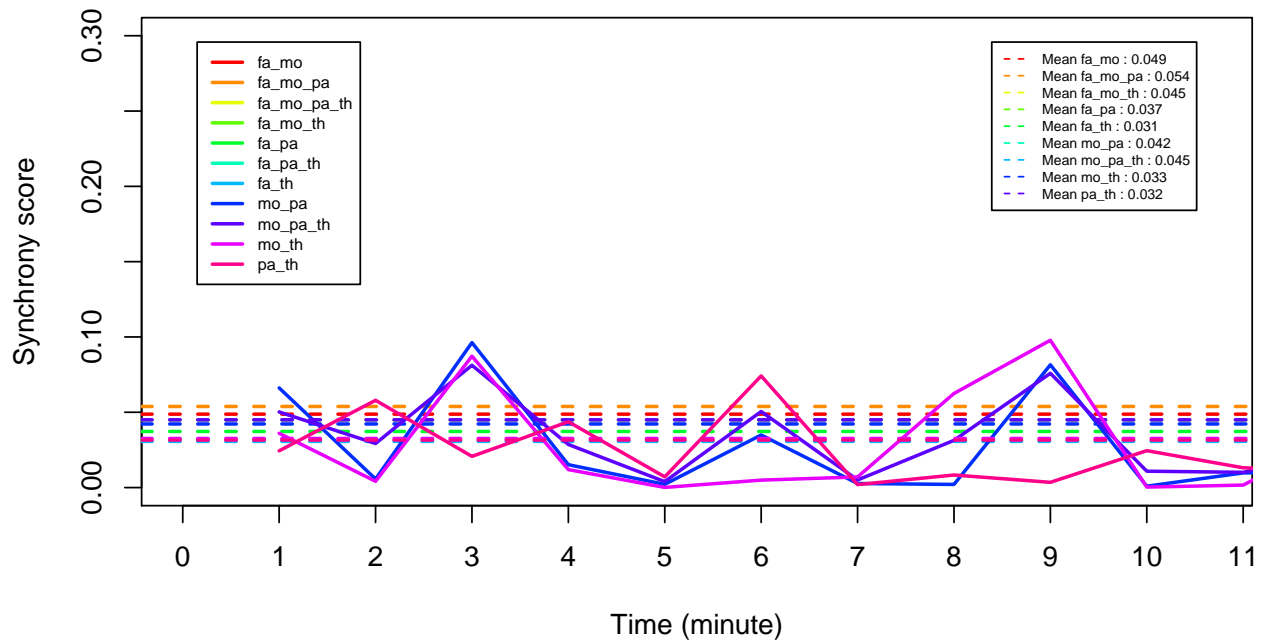
**Synchrony scores for each dyad and for the whole group in F1002A2 video**



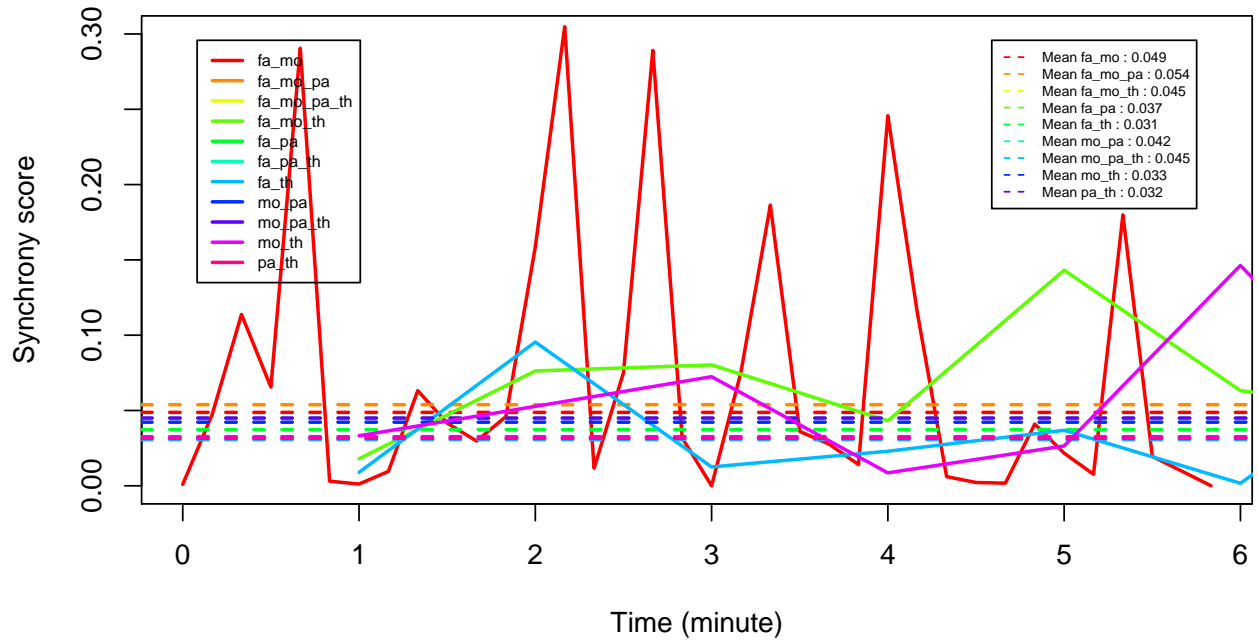
**Synchrony scores for each dyad and for the whole group in F1002B2 video**



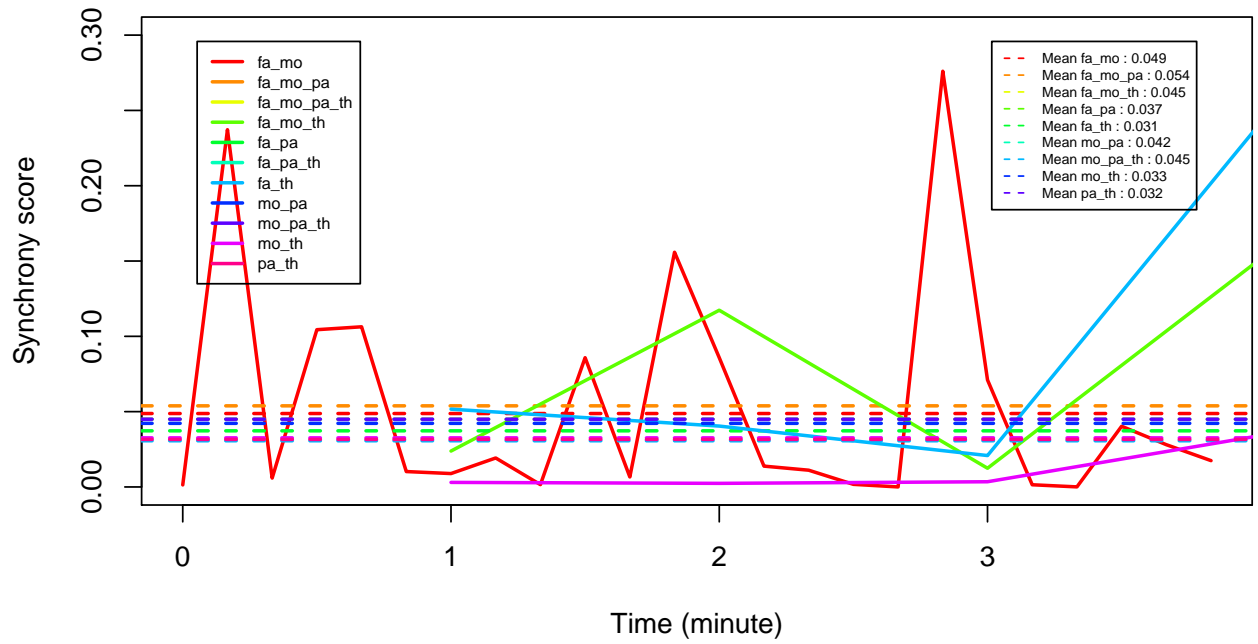
**Synchrony scores for each dyad and for the whole group in F1002C1 video**



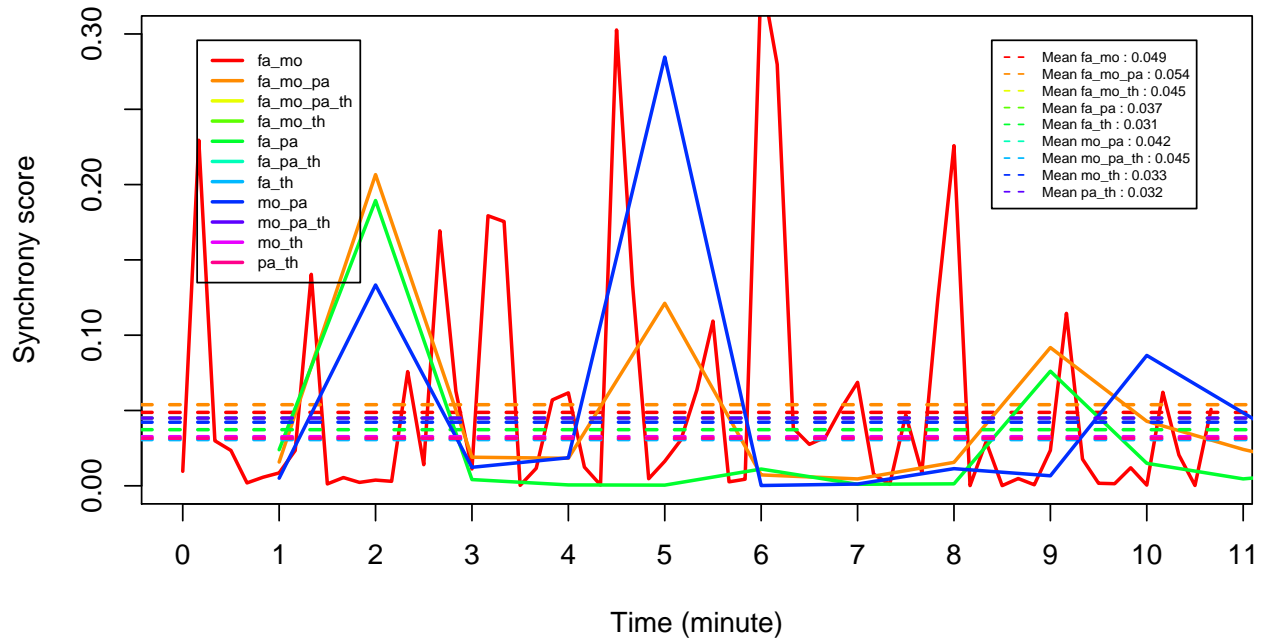
**Synchrony scores for each dyad and for the whole group in F1044C1 video**



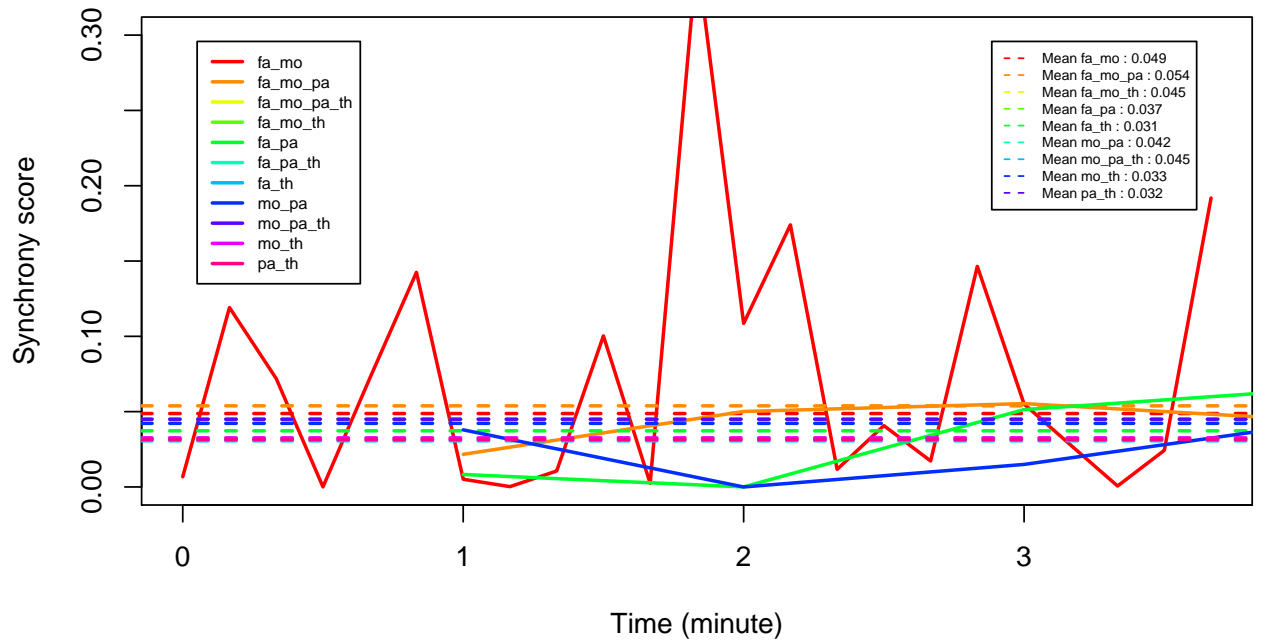
**Synchrony scores for each dyad and for the whole group in F1044C2 video**



**Synchrony scores for each dyad and for the whole group in F1044D1 video**

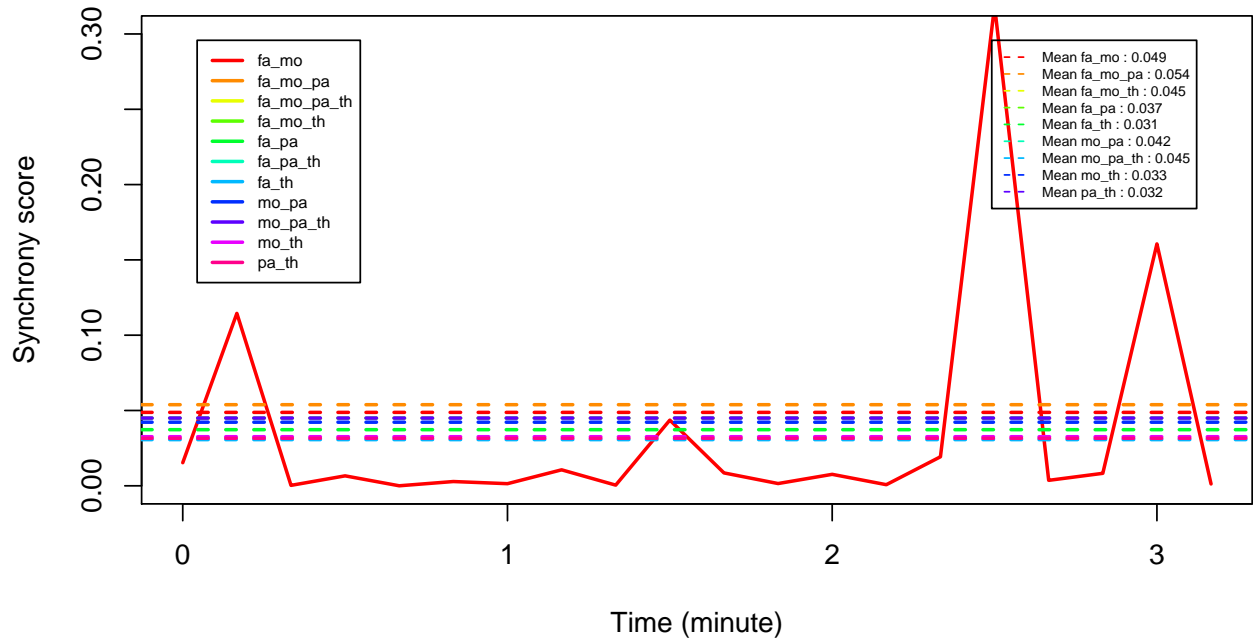


**Synchrony scores for each dyad and for the whole group in F1044D2 video**

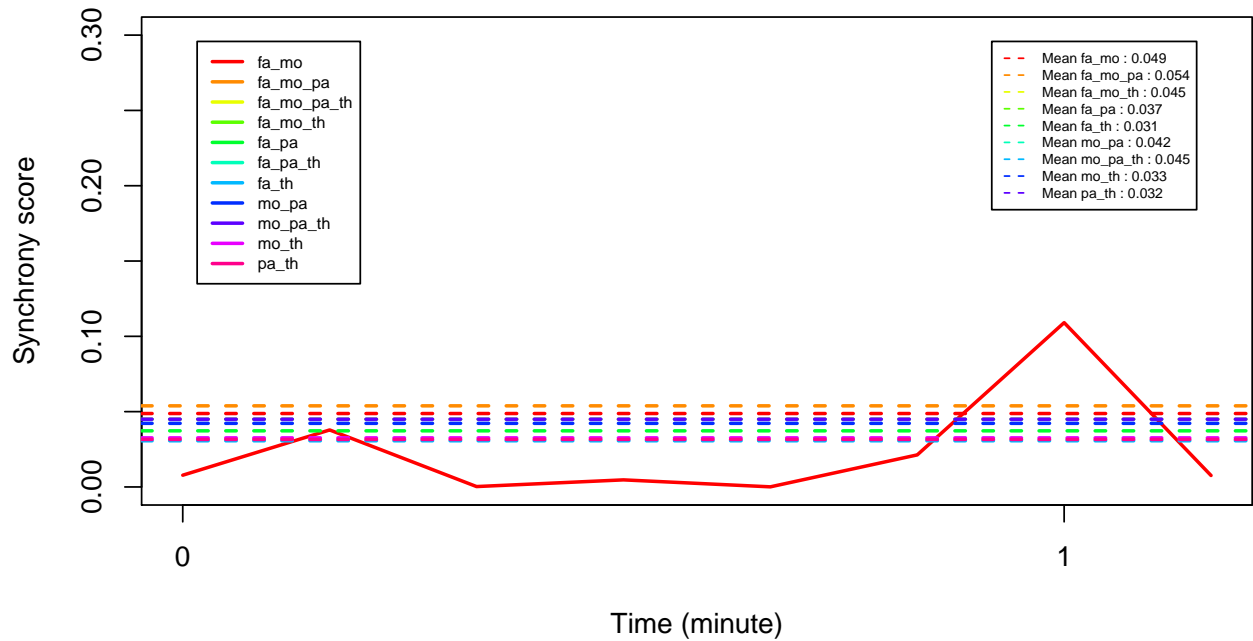




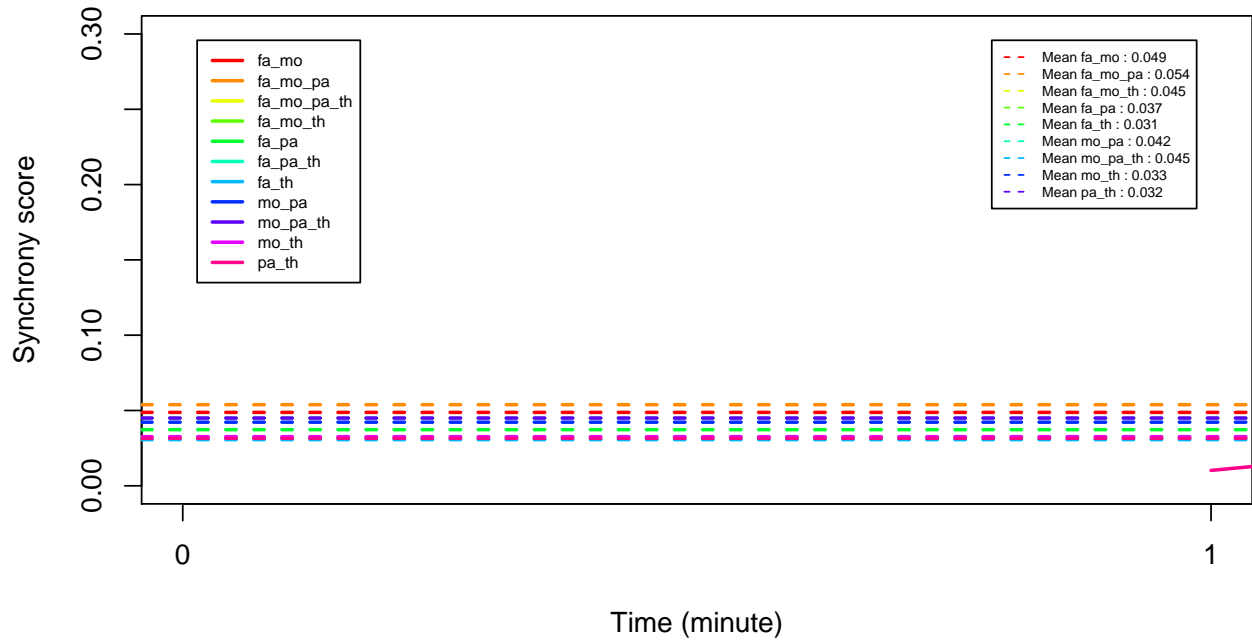
**Synchrony scores for each dyad and for the whole group in F1044E1 video**



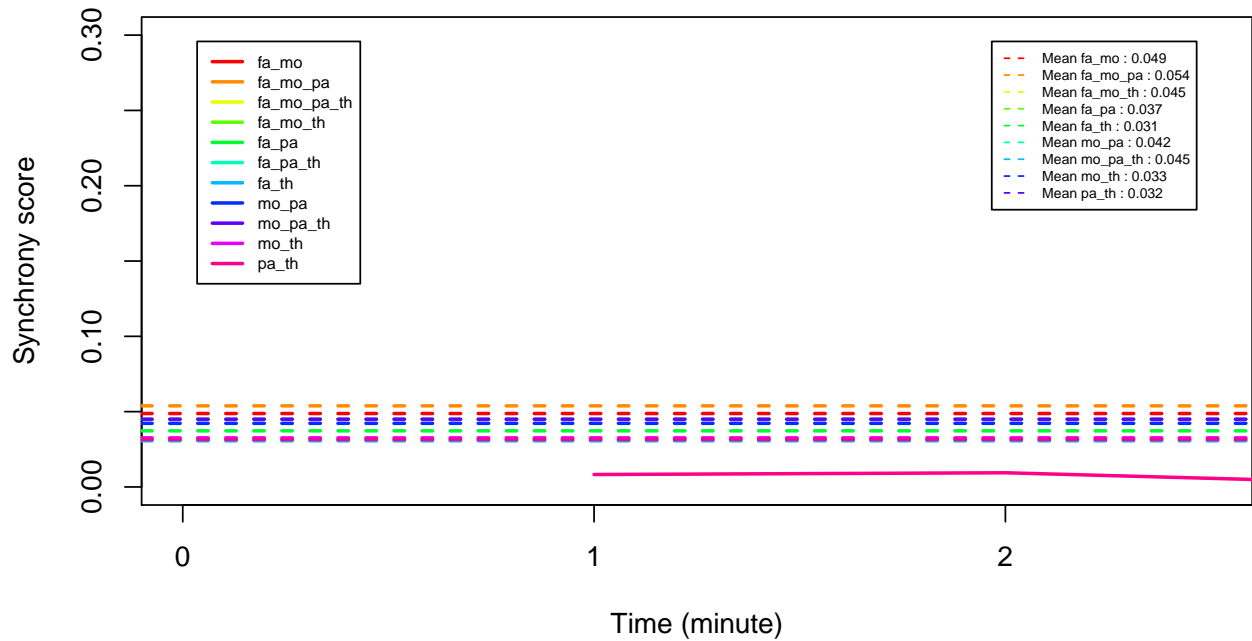
**Synchrony scores for each dyad and for the whole group in F1044E2 video**



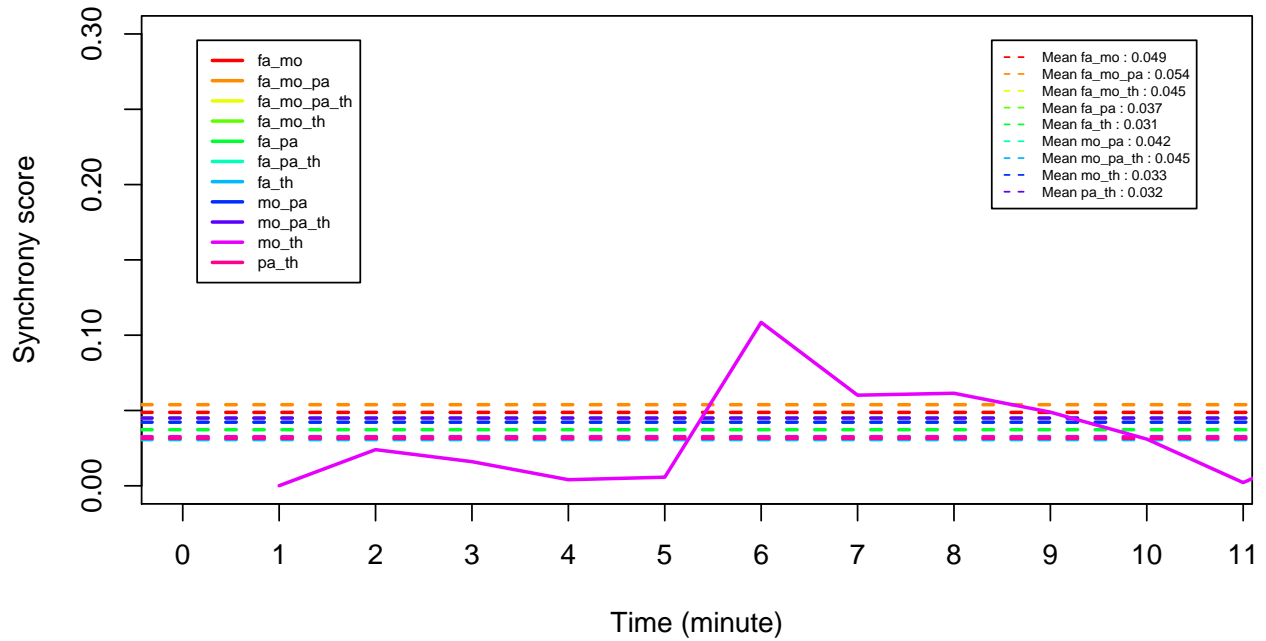
**Synchrony scores for each dyad and for the whole group in F1044F1 video**



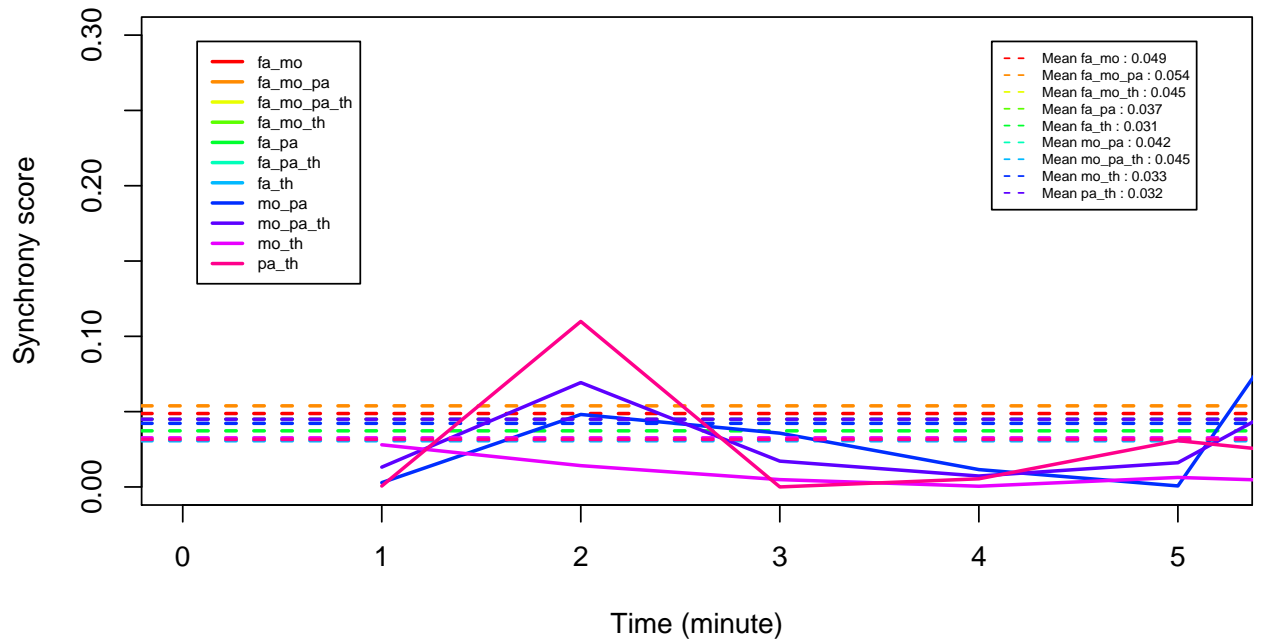
**Synchrony scores for each dyad and for the whole group in F1044F2 video**



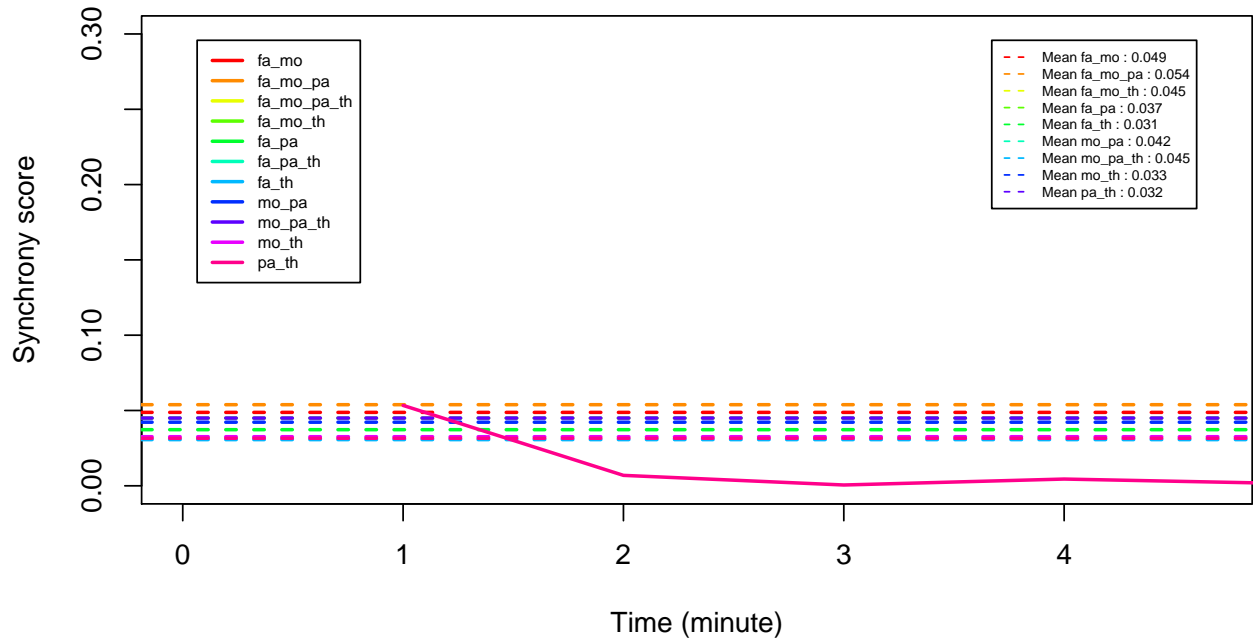
**Synchrony scores for each dyad and for the whole group in F1044G video**



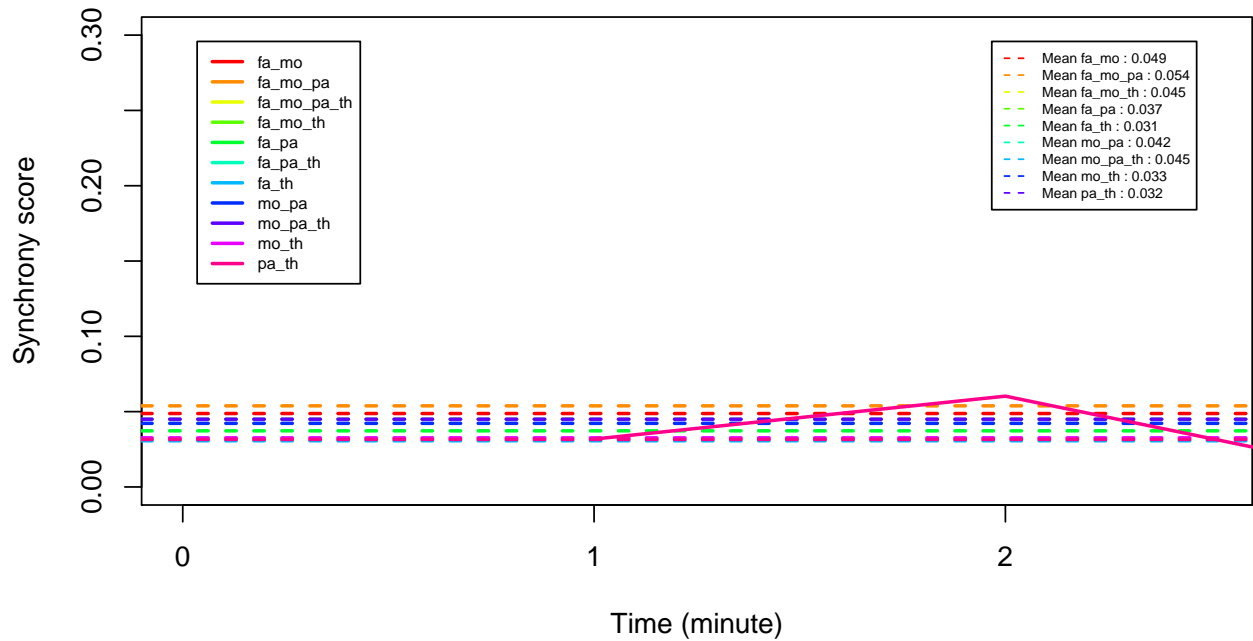
**Synchrony scores for each dyad and for the whole group in F1044H1 video**



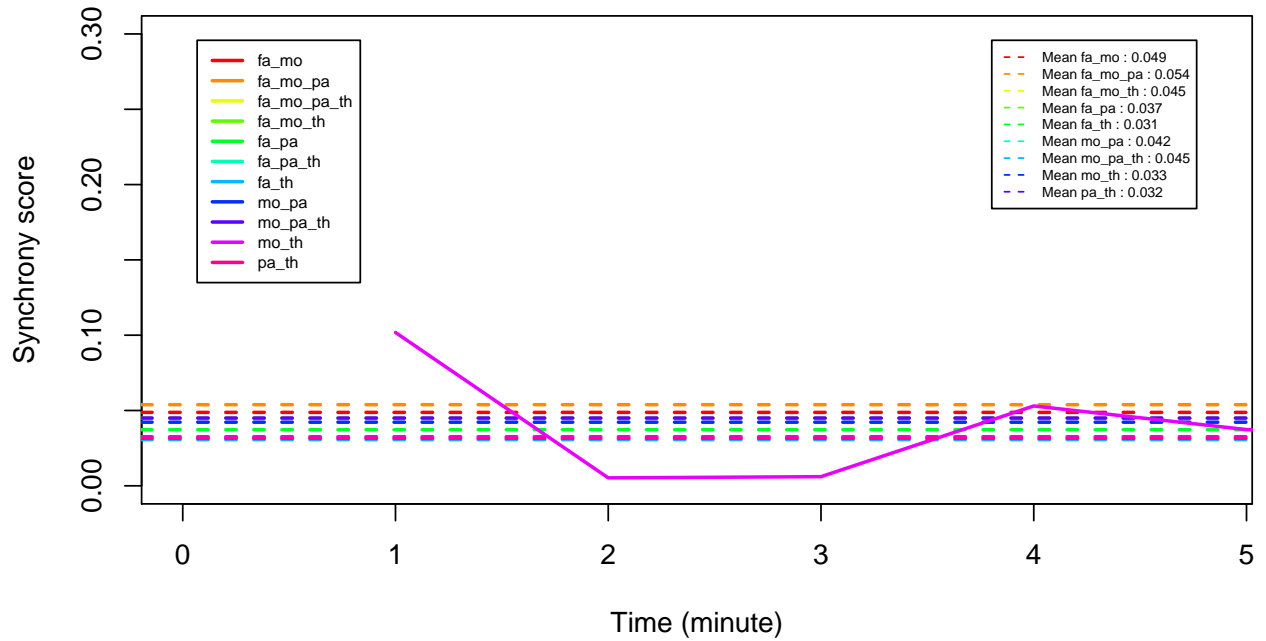
**Synchrony scores for each dyad and for the whole group in F1044H2 video**



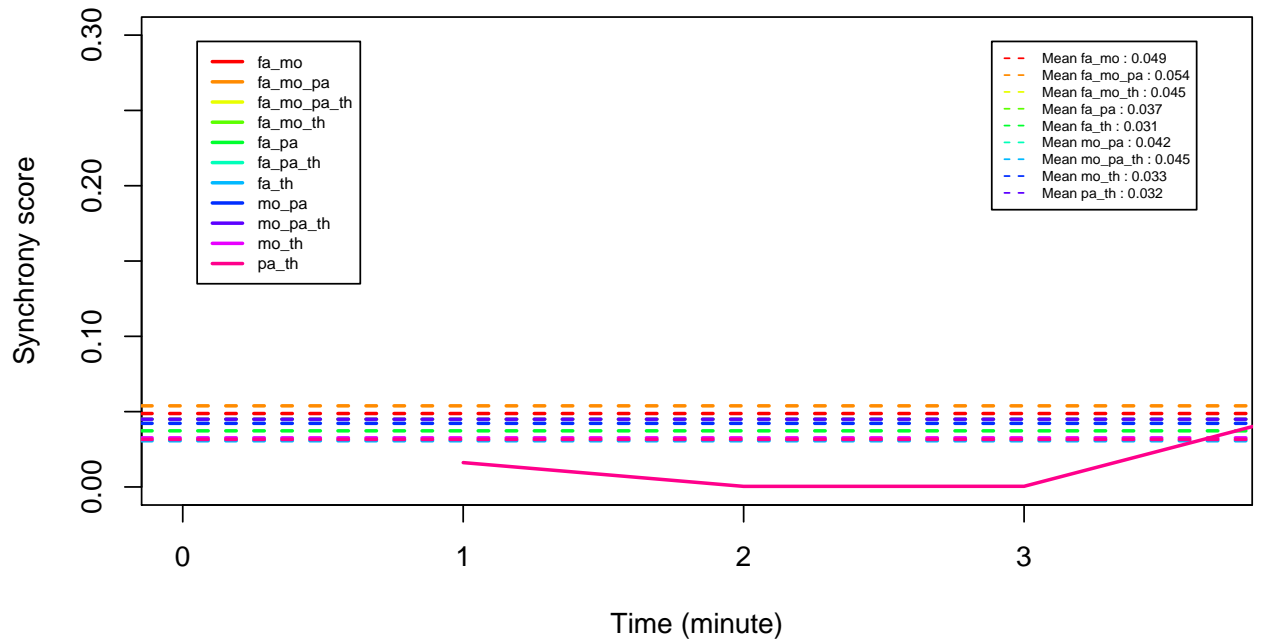
**Synchrony scores for each dyad and for the whole group in F1044I1 video**



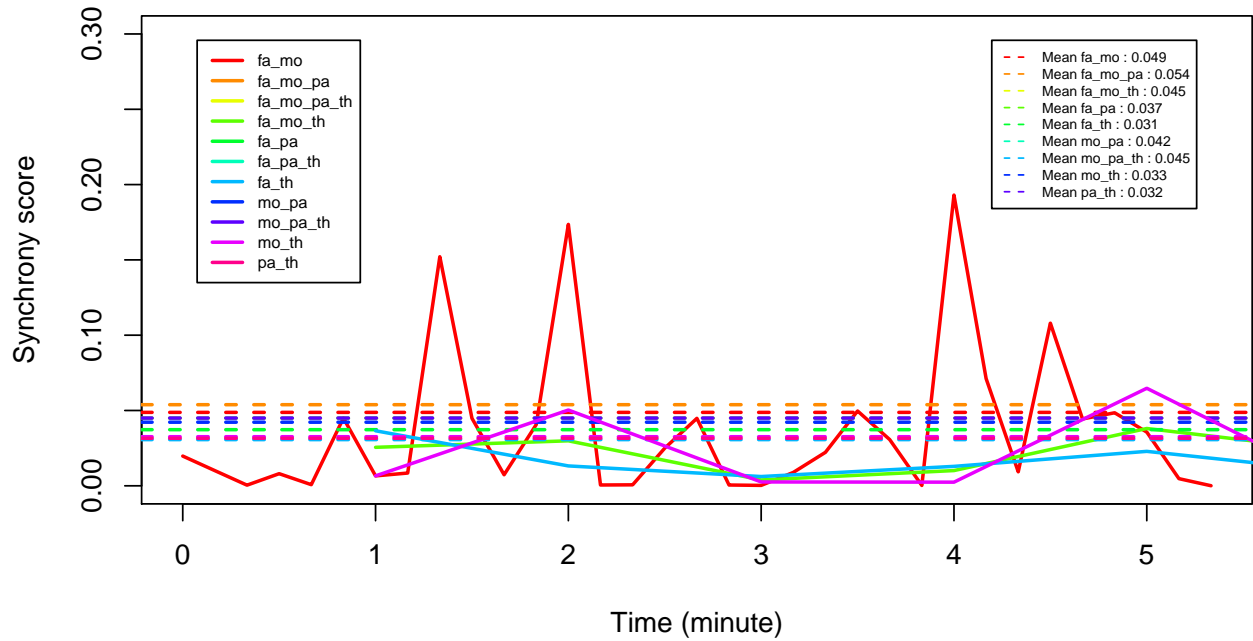
**Synchrony scores for each dyad and for the whole group in F1044I2 video**



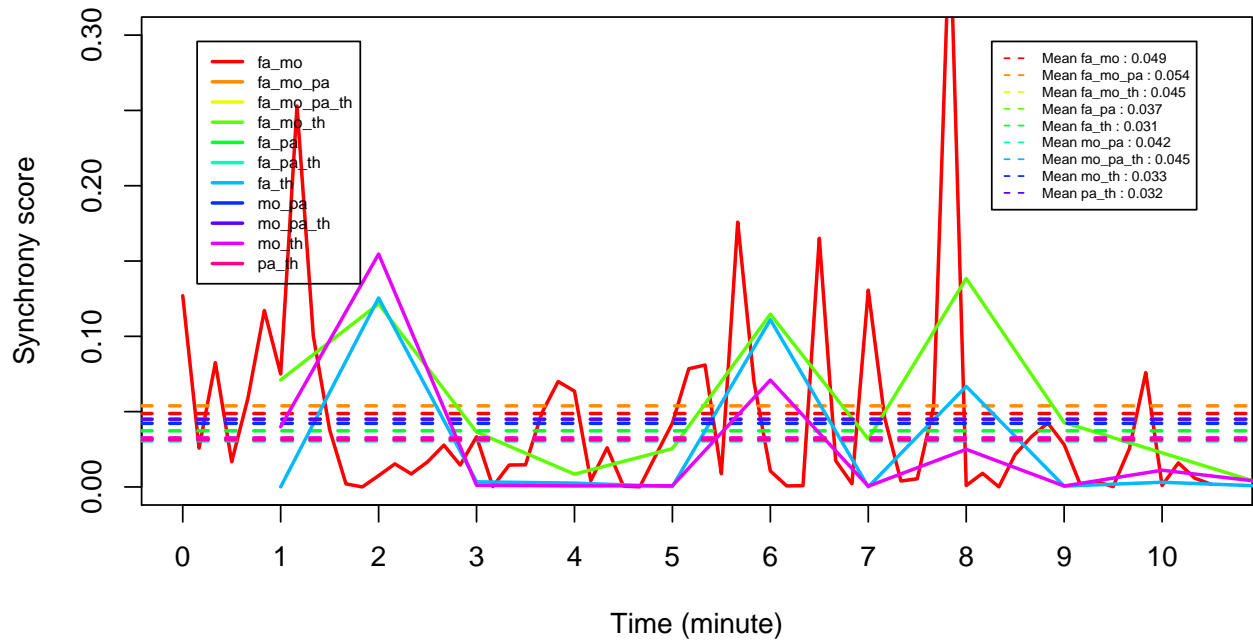
**Synchrony scores for each dyad and for the whole group in F1044L1 video**



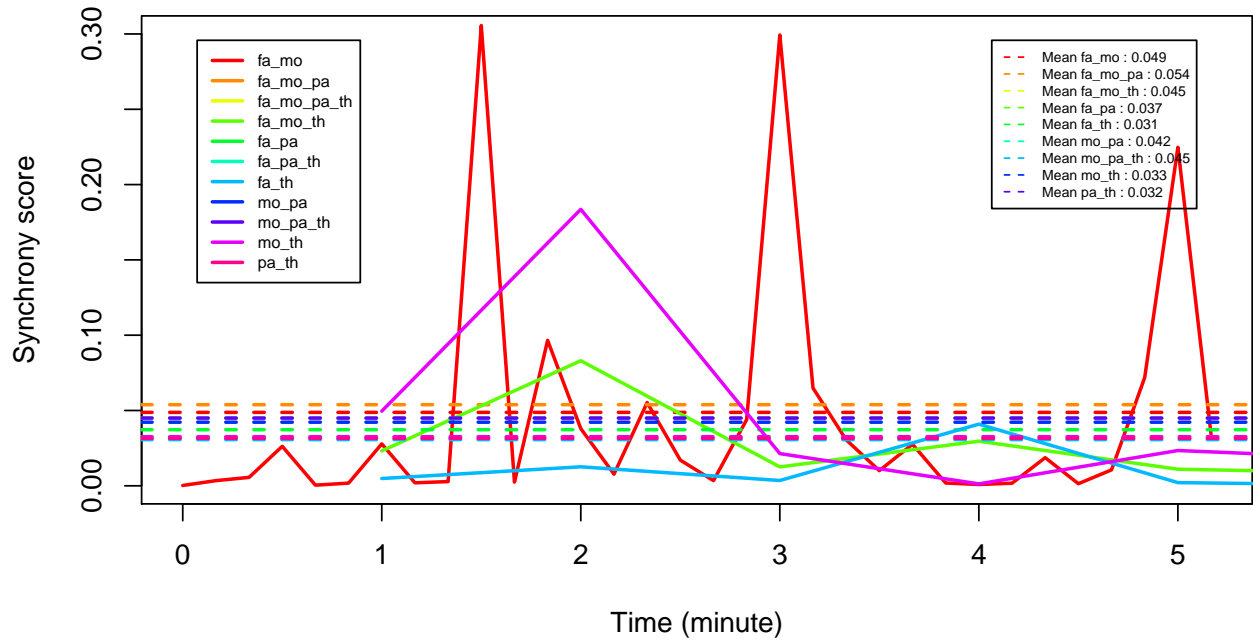
**Synchrony scores for each dyad and for the whole group in F1044L2 video**



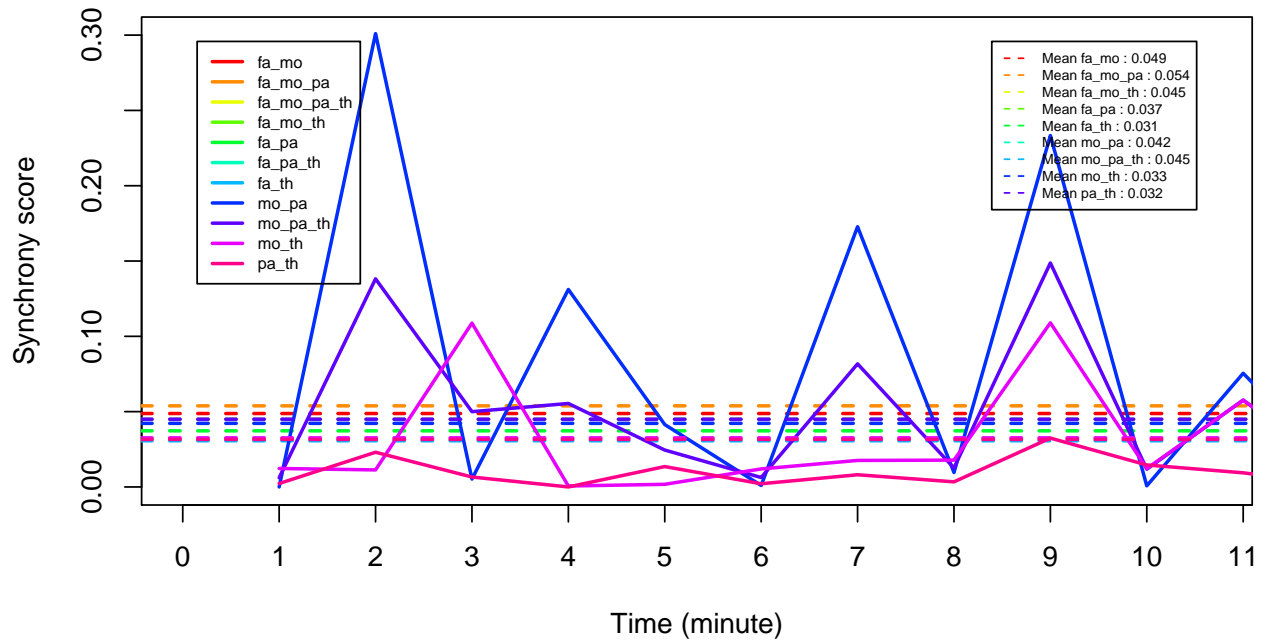
**Synchrony scores for each dyad and for the whole group in F1044M1 video**



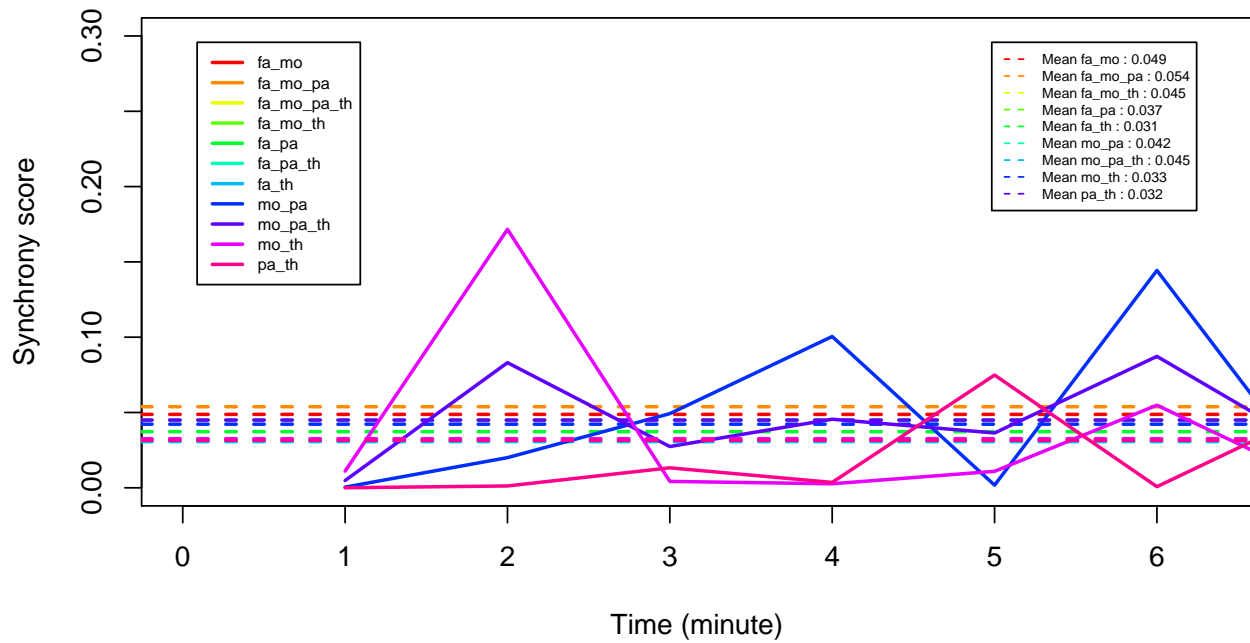
**Synchrony scores for each dyad and for the whole group in F1044M2 video**



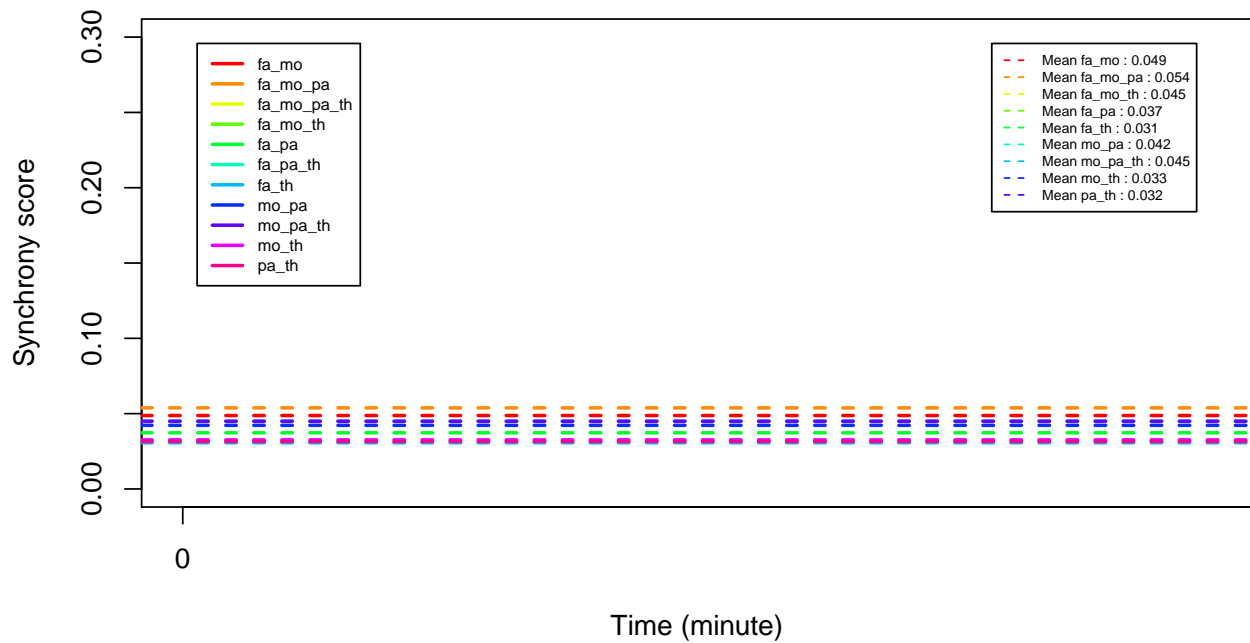
**Synchrony scores for each dyad and for the whole group in F1044N video**



**Synchrony scores for each dyad and for the whole group in F1044O1 video**

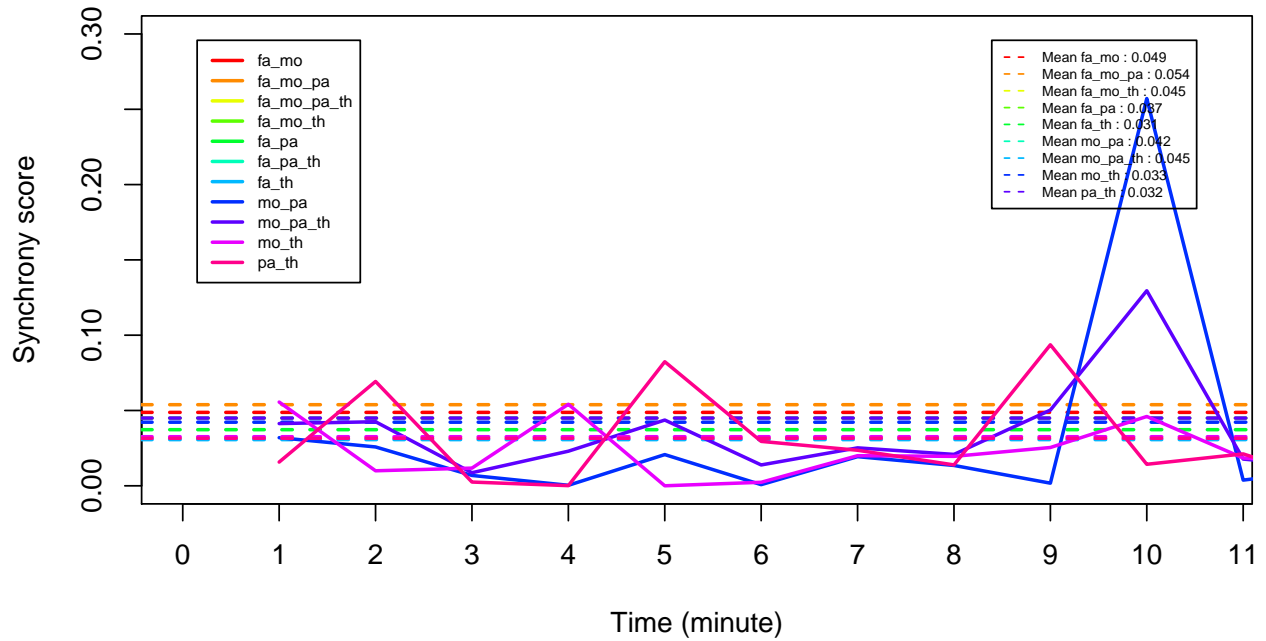


**Synchrony scores for each dyad and for the whole group in F1044O2 video**

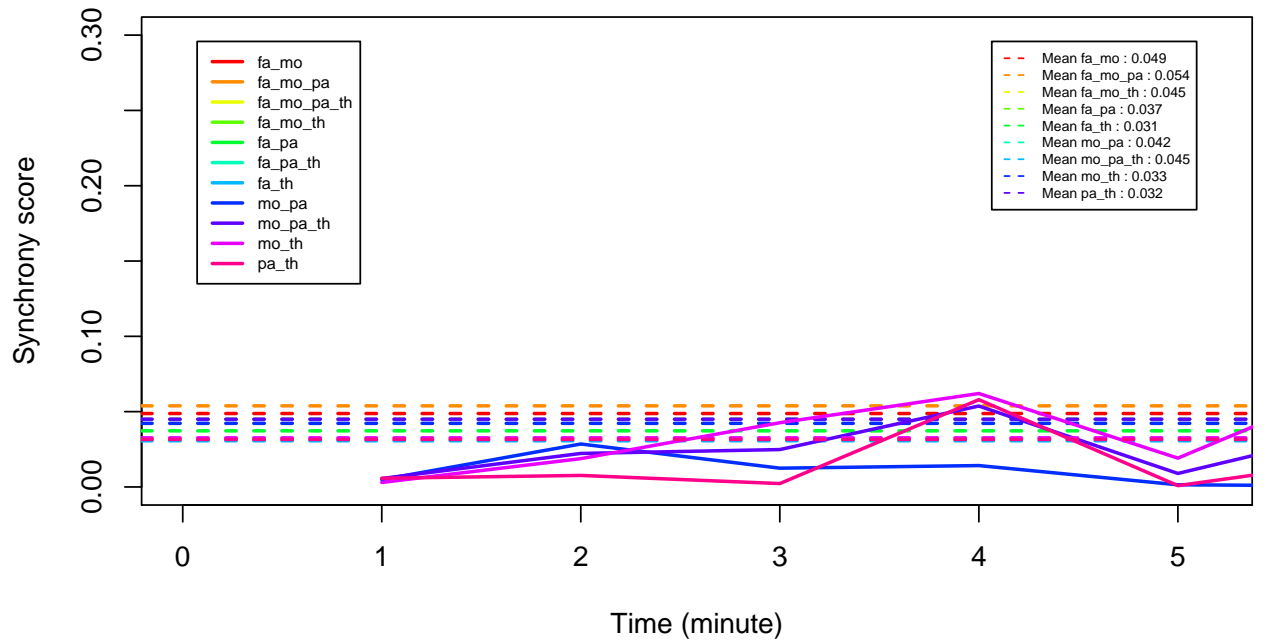




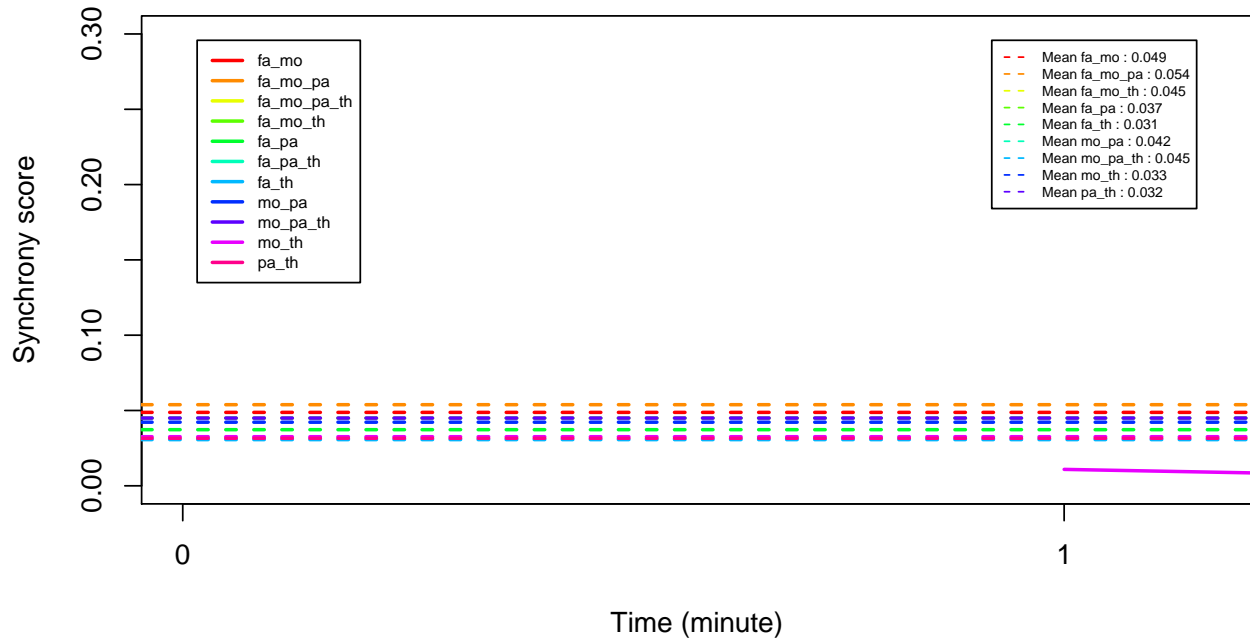
**Synchrony scores for each dyad and for the whole group in F1044P video**



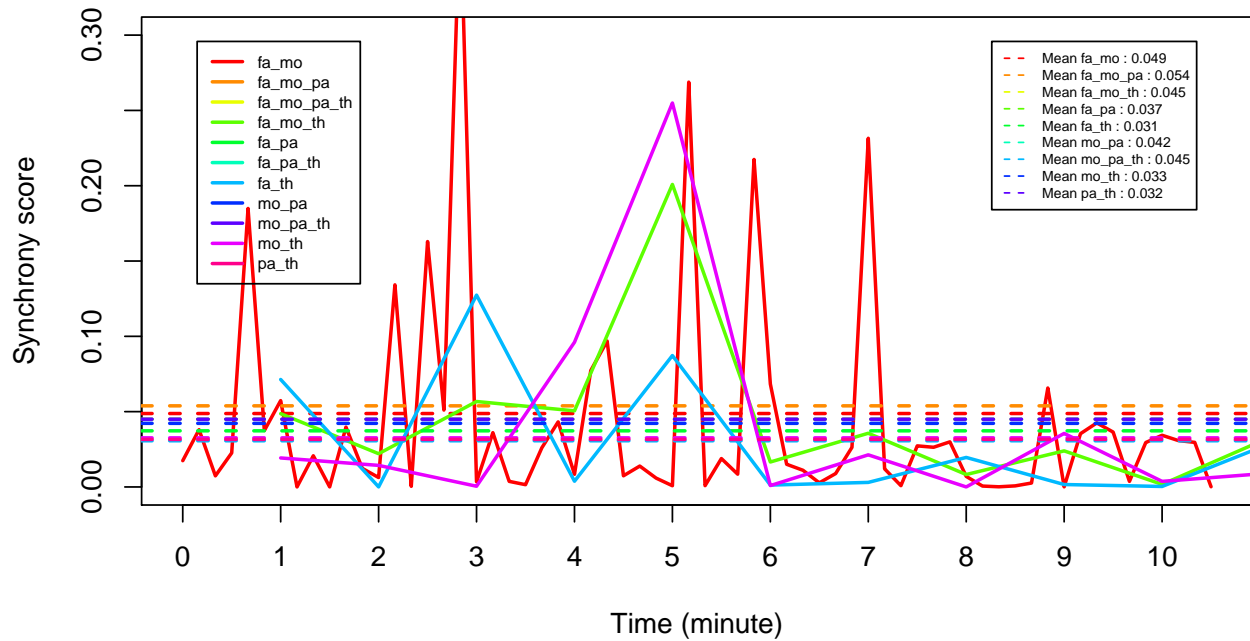
**Synchrony scores for each dyad and for the whole group in F1044Q1 video**



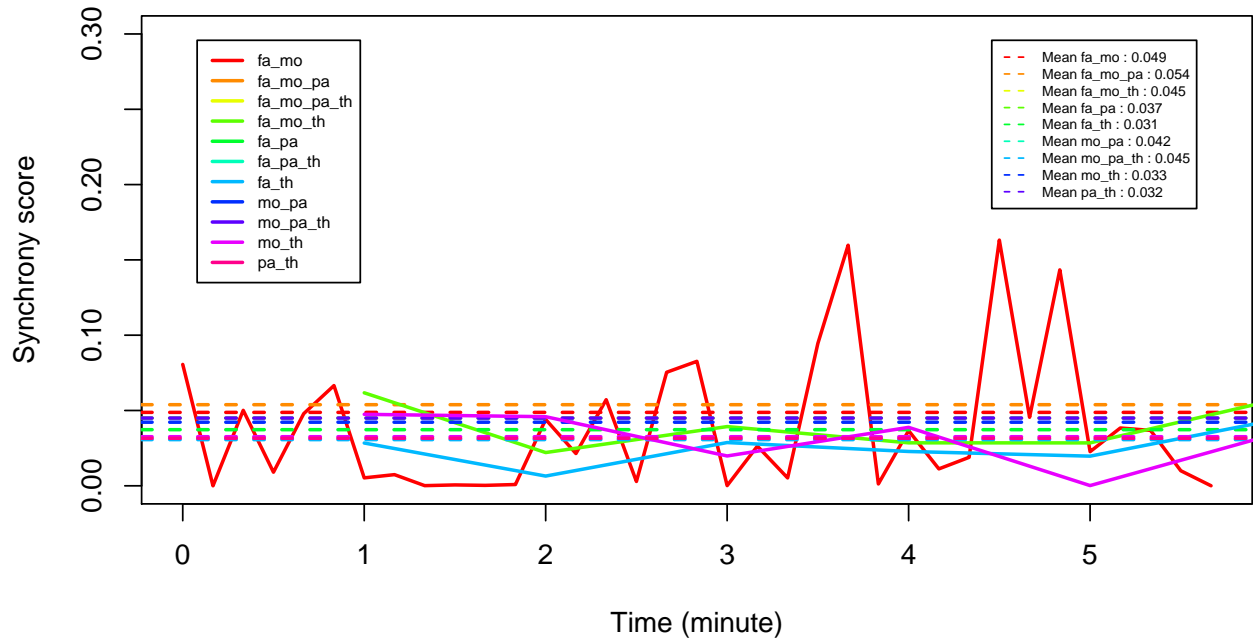
**Synchrony scores for each dyad and for the whole group in F1044Q2 video**



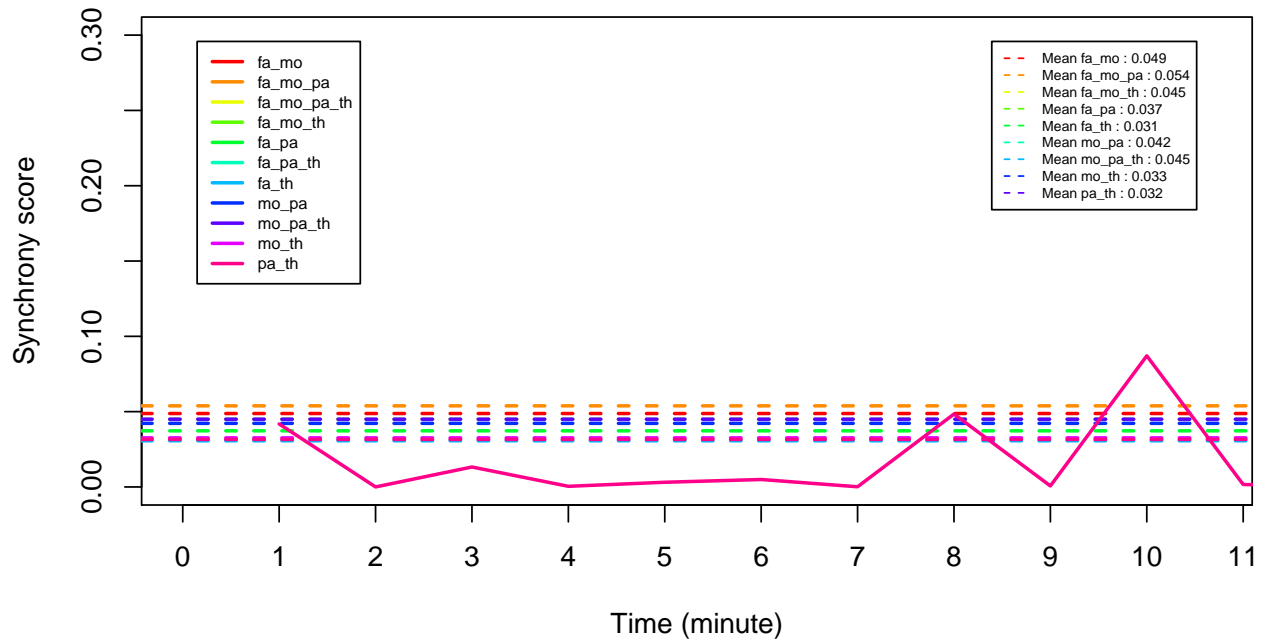
**Synchrony scores for each dyad and for the whole group in F1044R1 video**



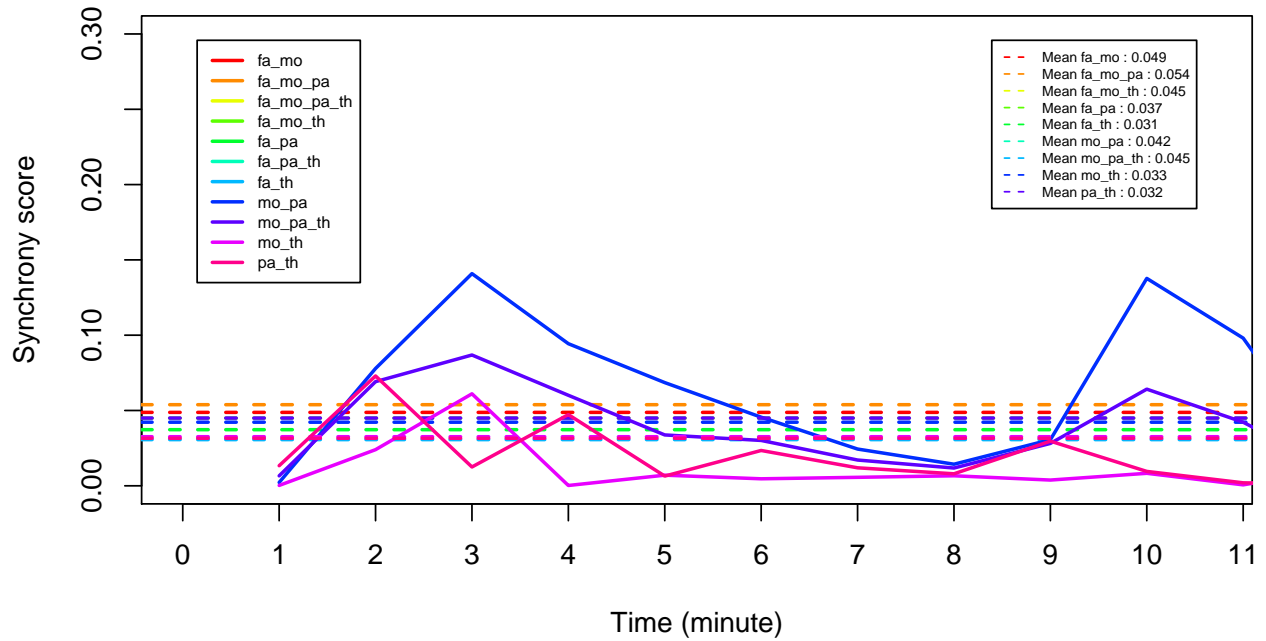
**Synchrony scores for each dyad and for the whole group in F1044R2 video**



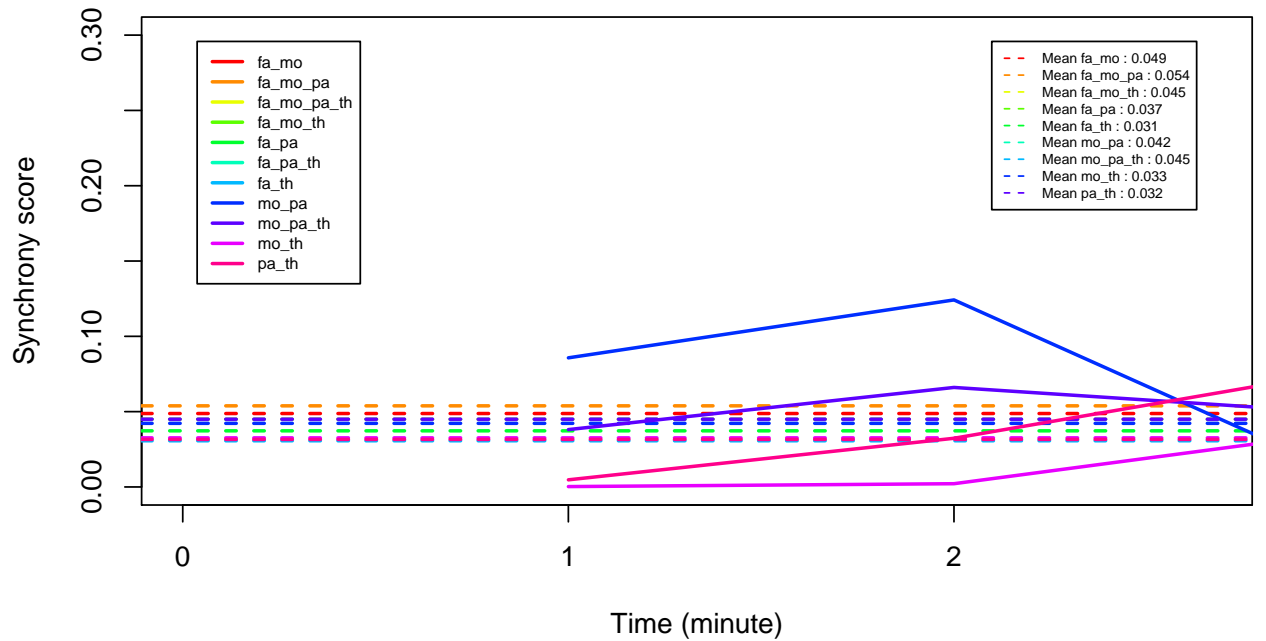
**Synchrony scores for each dyad and for the whole group in F1069A1 video**



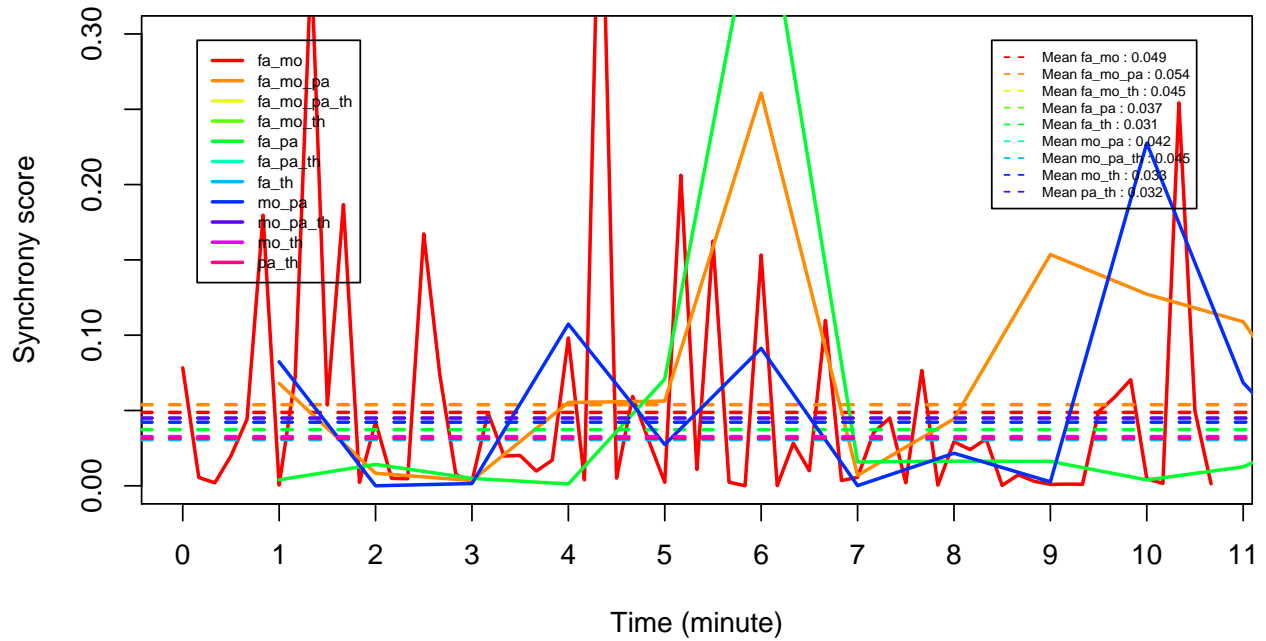
**Synchrony scores for each dyad and for the whole group in F1069B1 video**



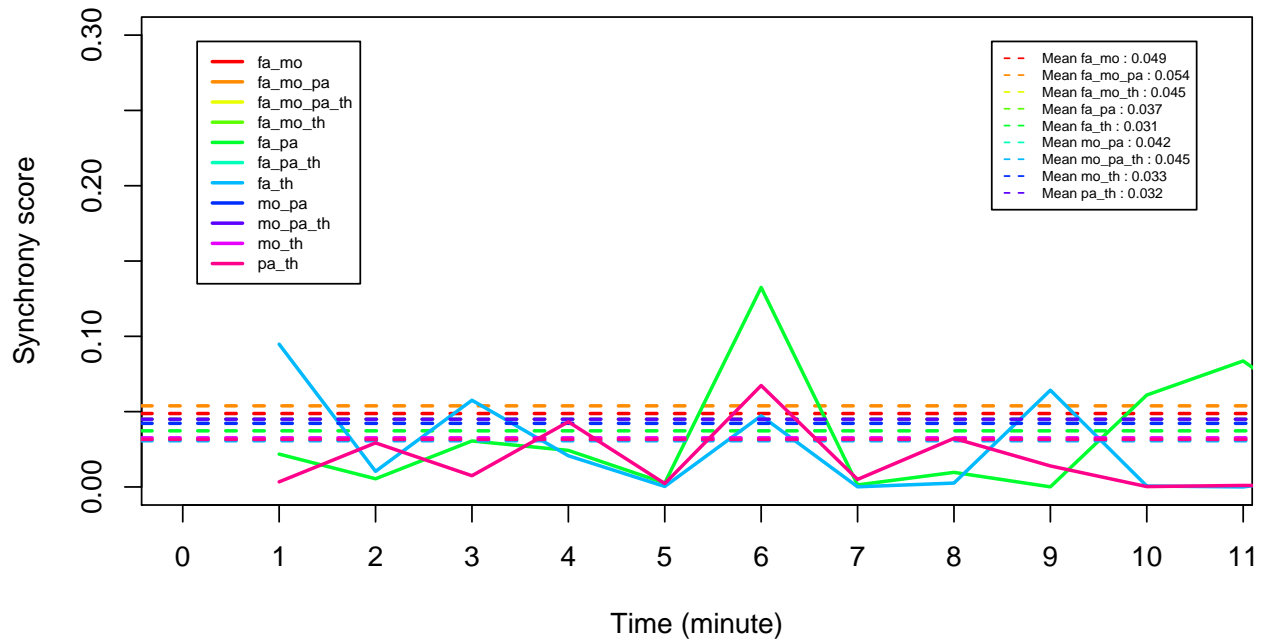
**Synchrony scores for each dyad and for the whole group in F1069B2 video**



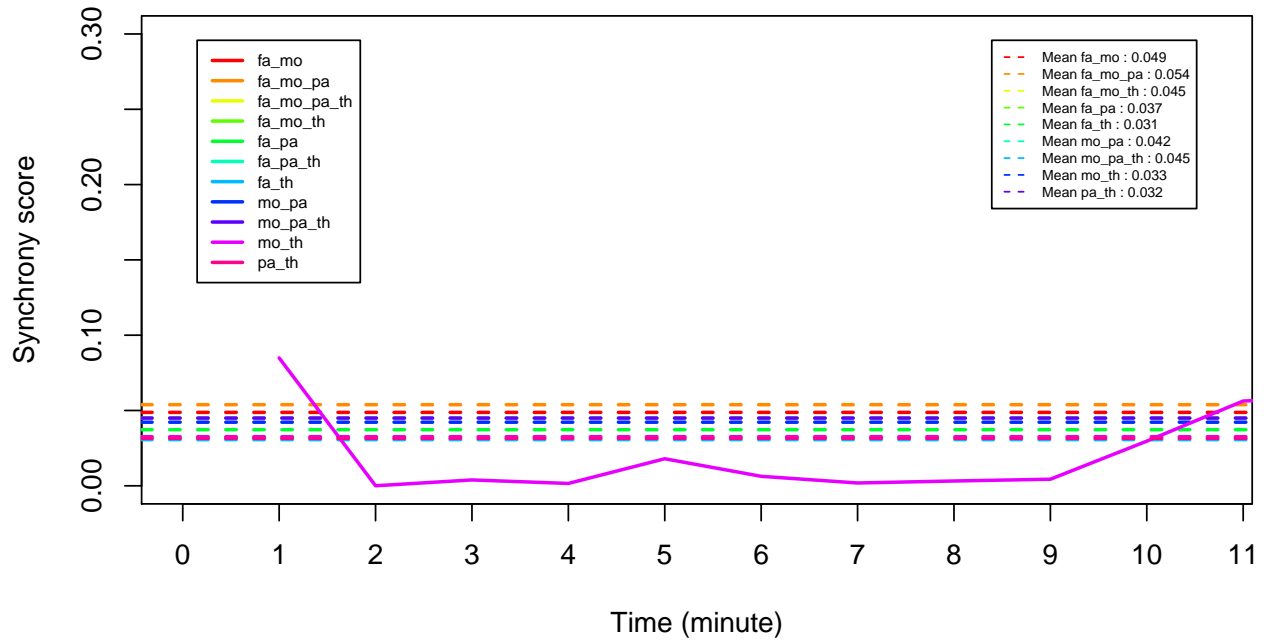
**Synchrony scores for each dyad and for the whole group in F1069C1 video**



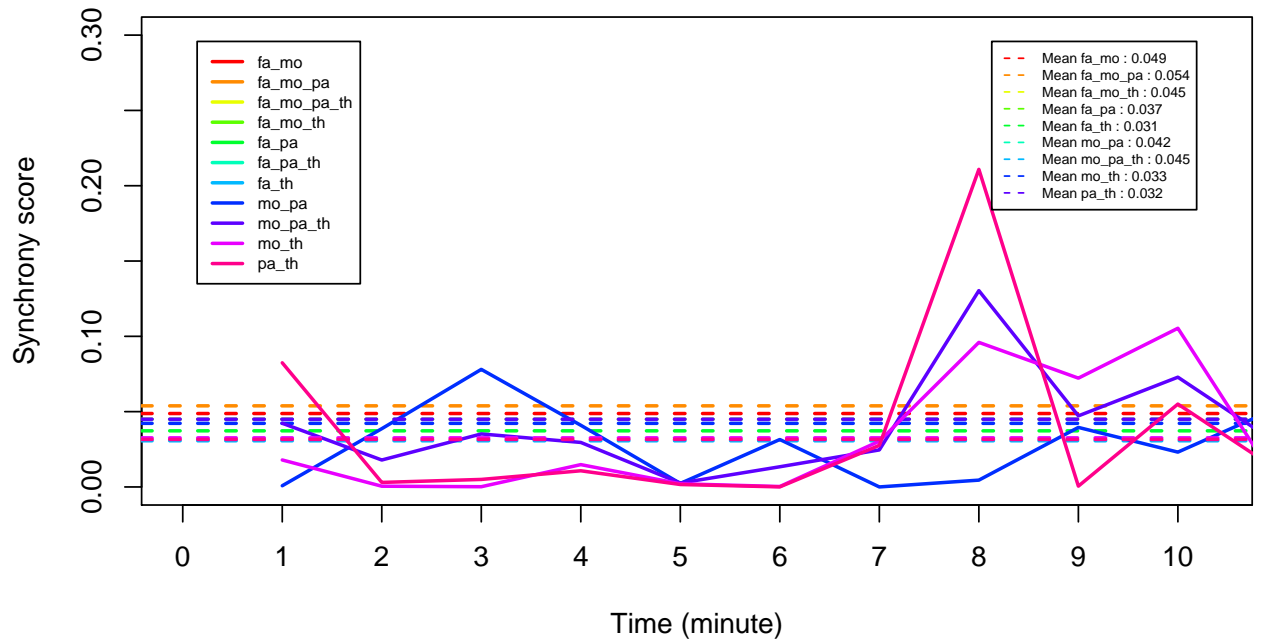
**Synchrony scores for each dyad and for the whole group in F1069D2 video**



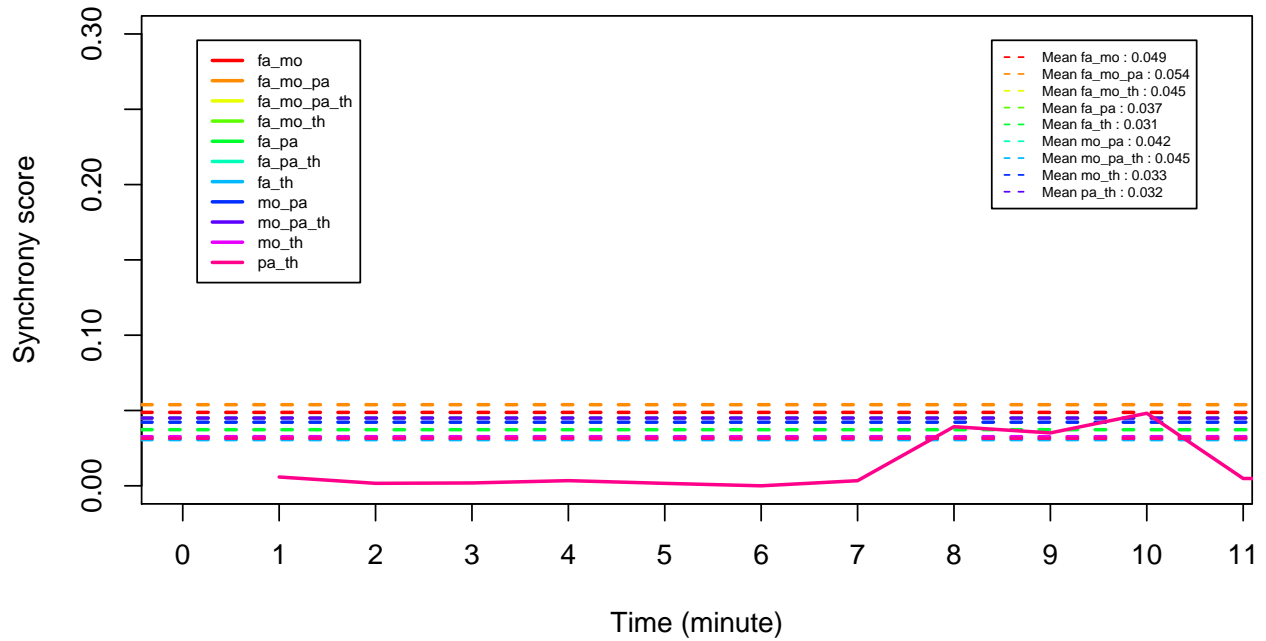
**Synchrony scores for each dyad and for the whole group in F1073A1 video**



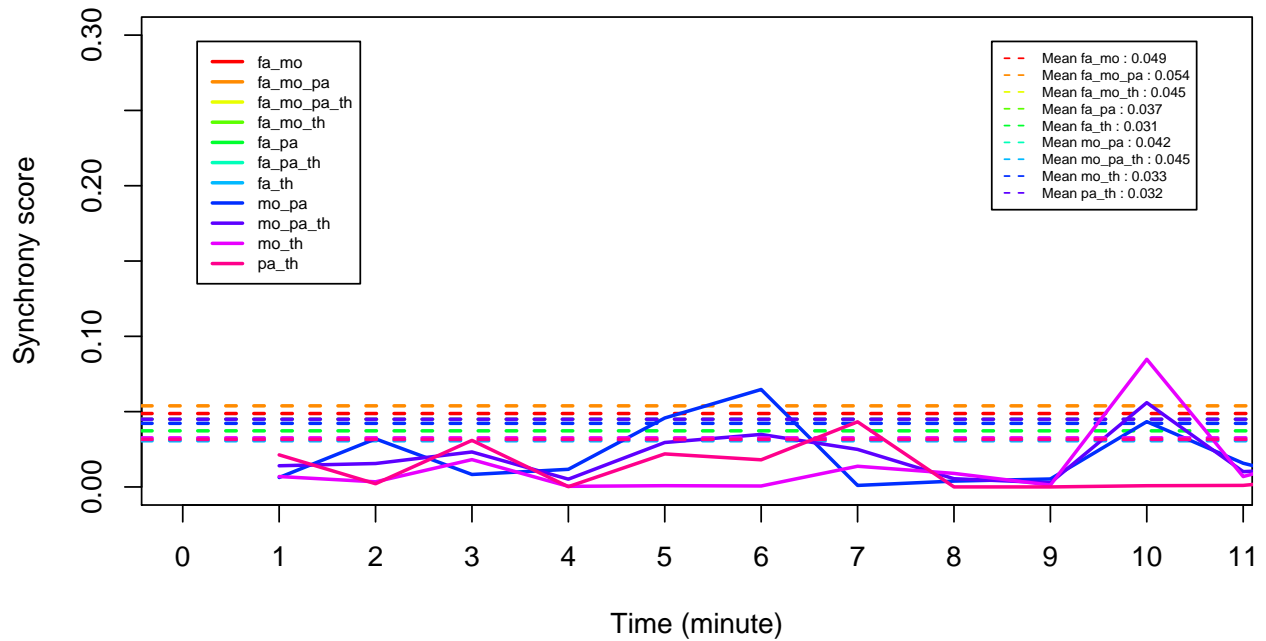
**Synchrony scores for each dyad and for the whole group in F1073A2 video**



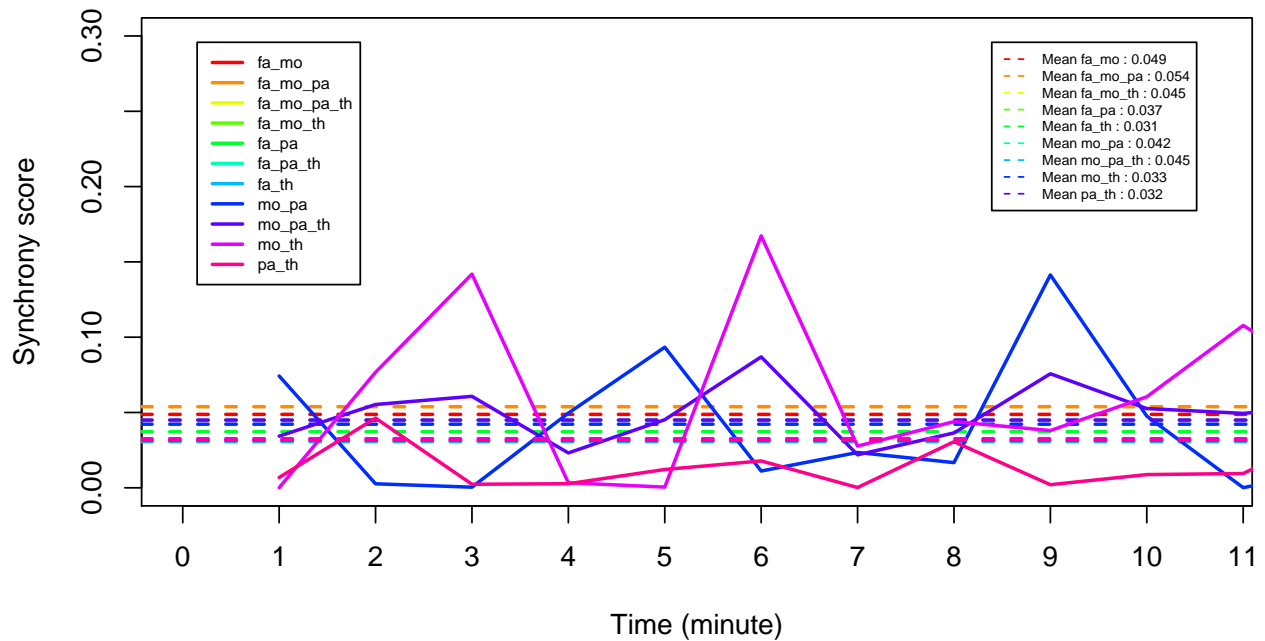
**Synchrony scores for each dyad and for the whole group in F1073B1 video**



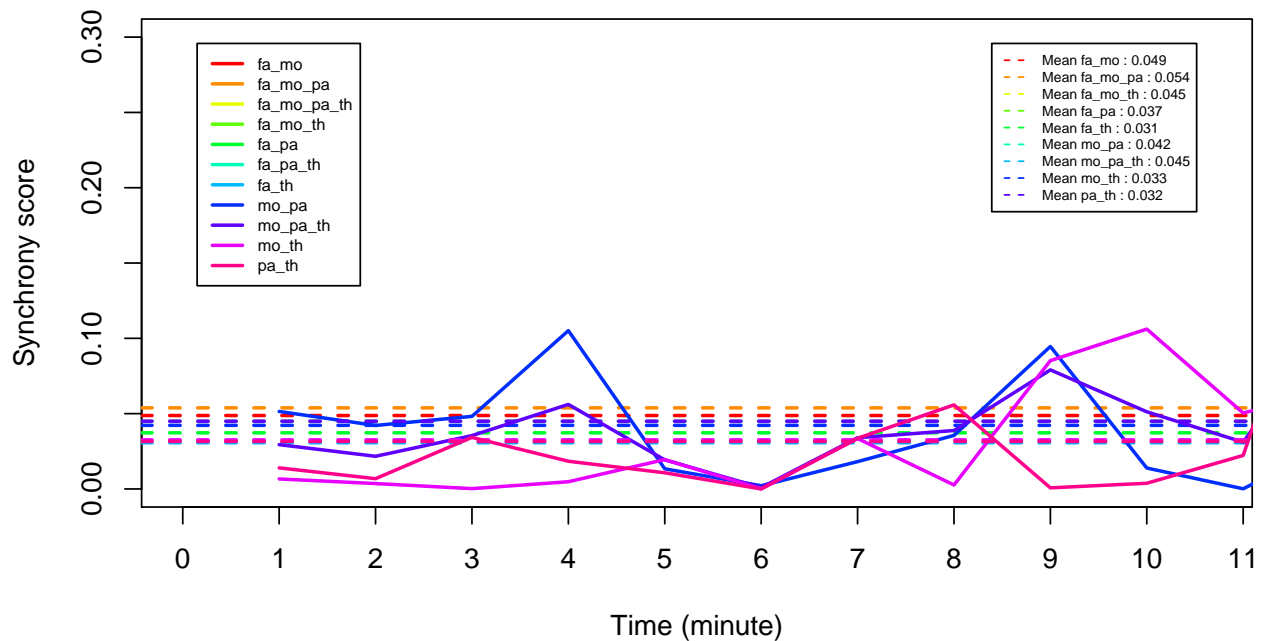
**Synchrony scores for each dyad and for the whole group in F1073B2 video**



### Synchrony scores for each dyad and for the whole group in F1101A2 video



### Synchrony scores for each dyad and for the whole group in F1101C2 video



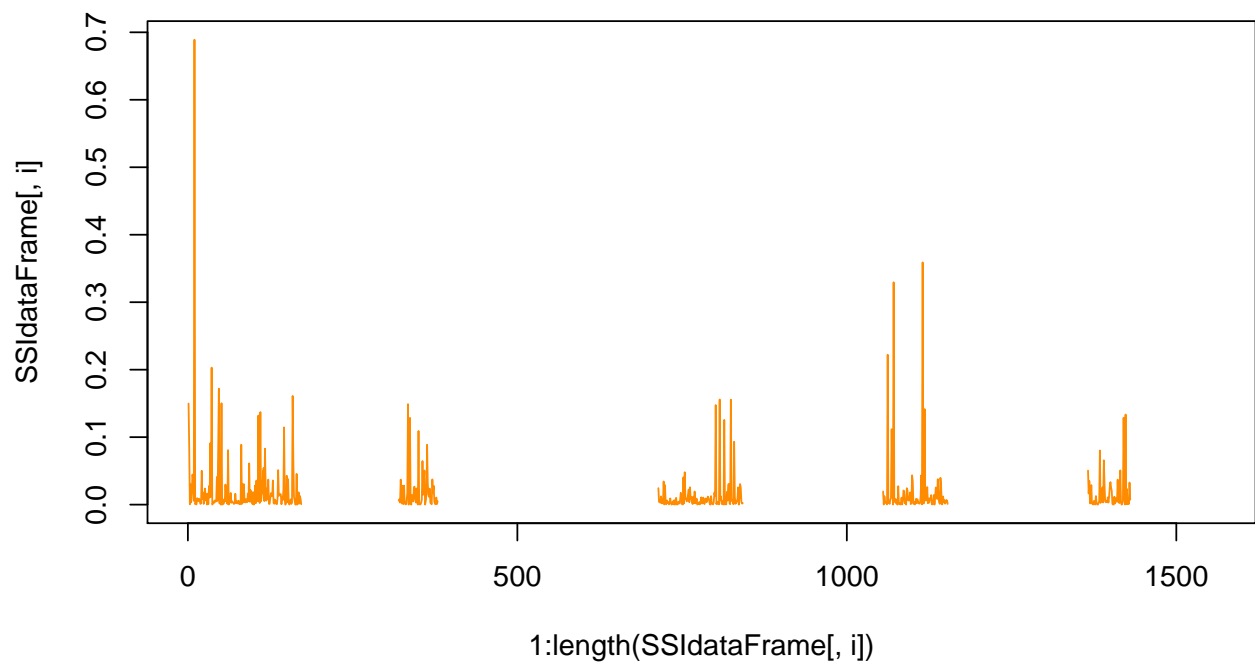
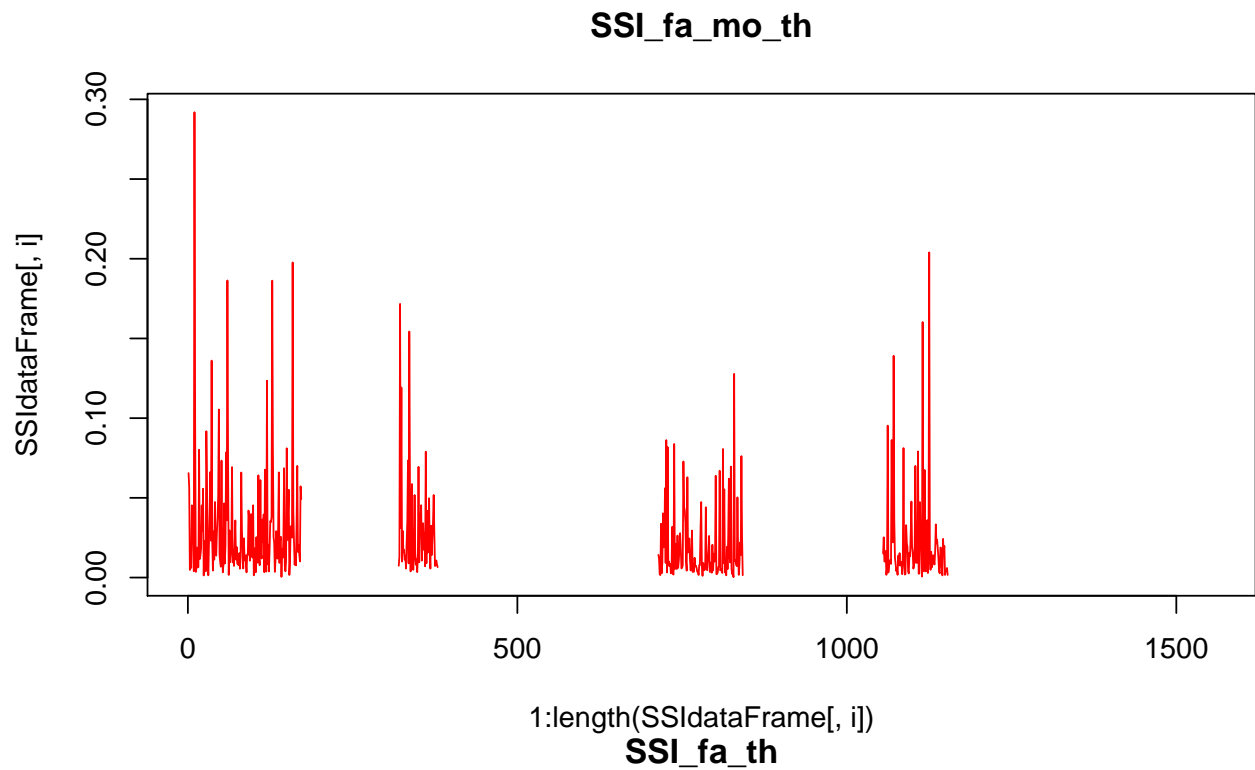
Evolution of synchrony through time, raw each second

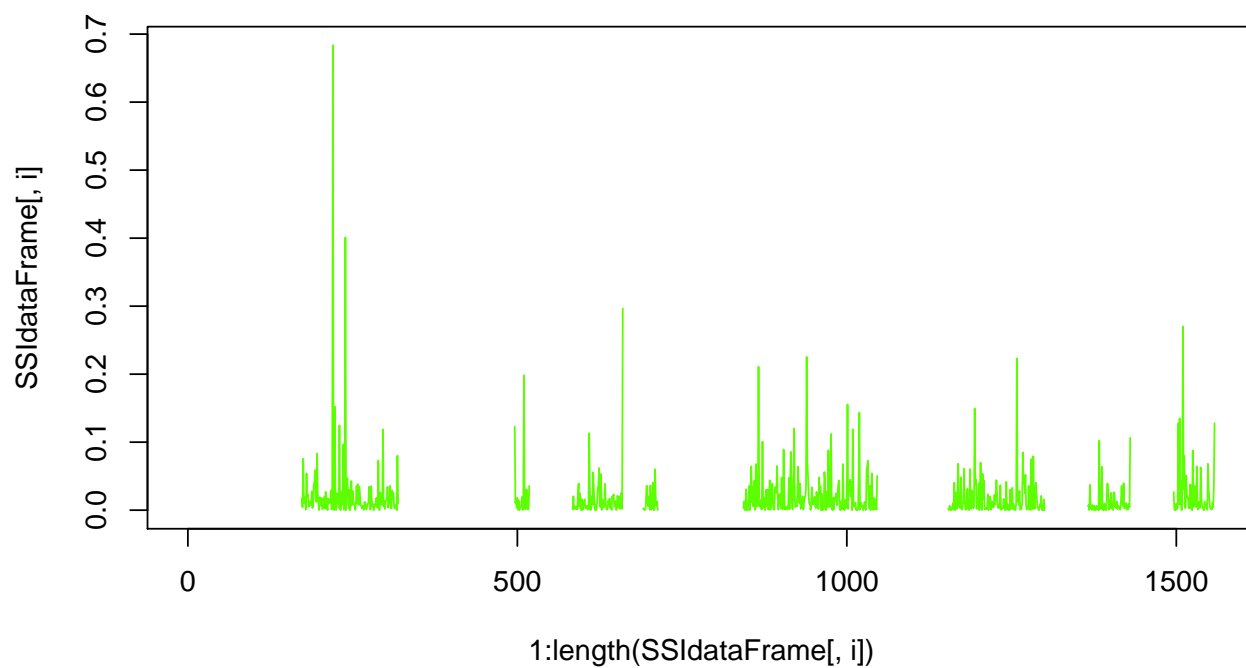
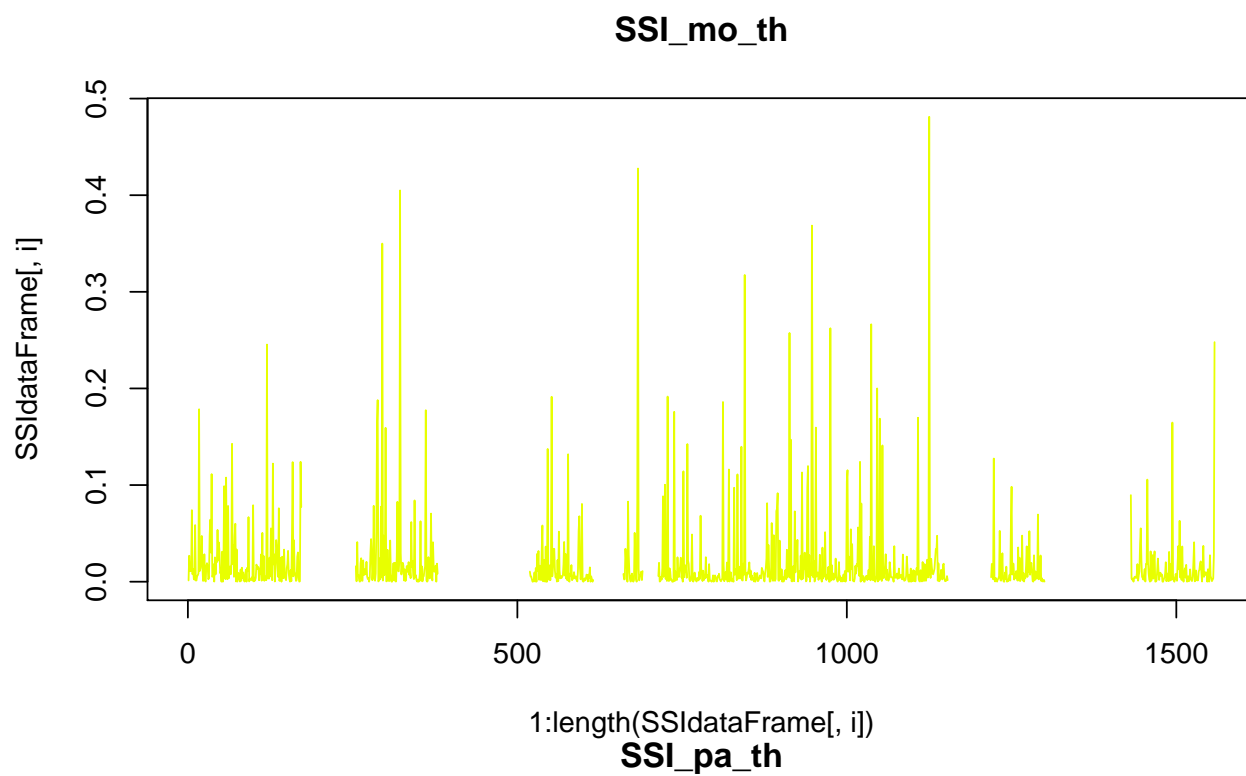
Nolog data

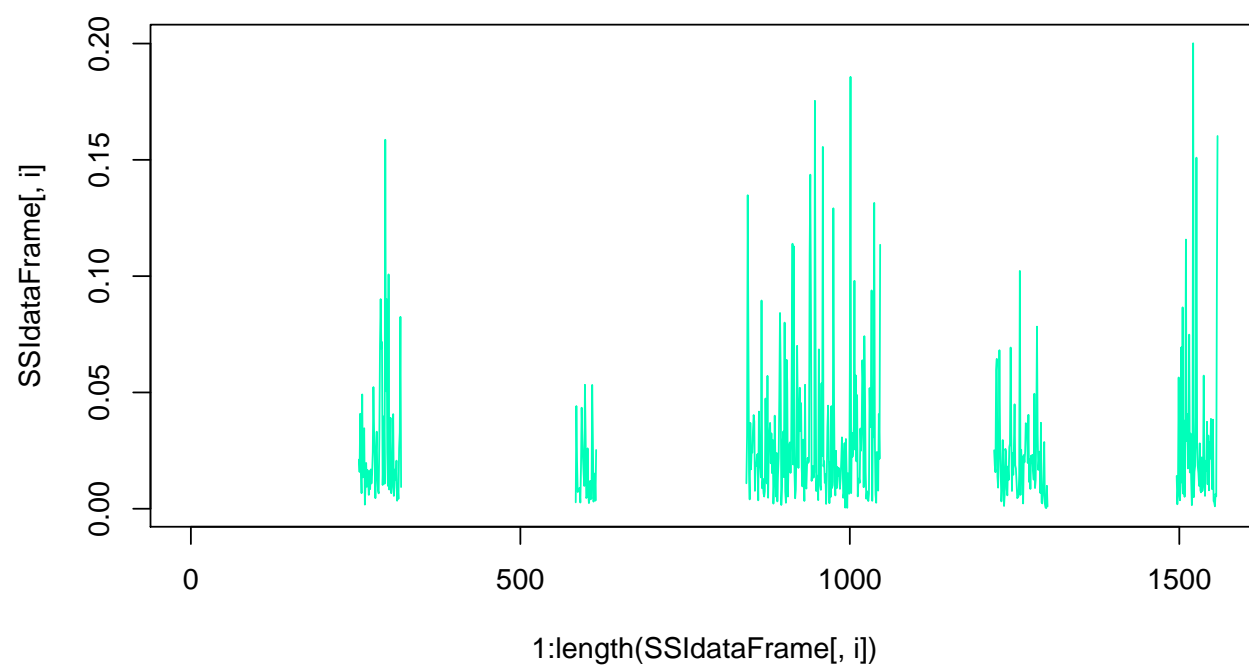
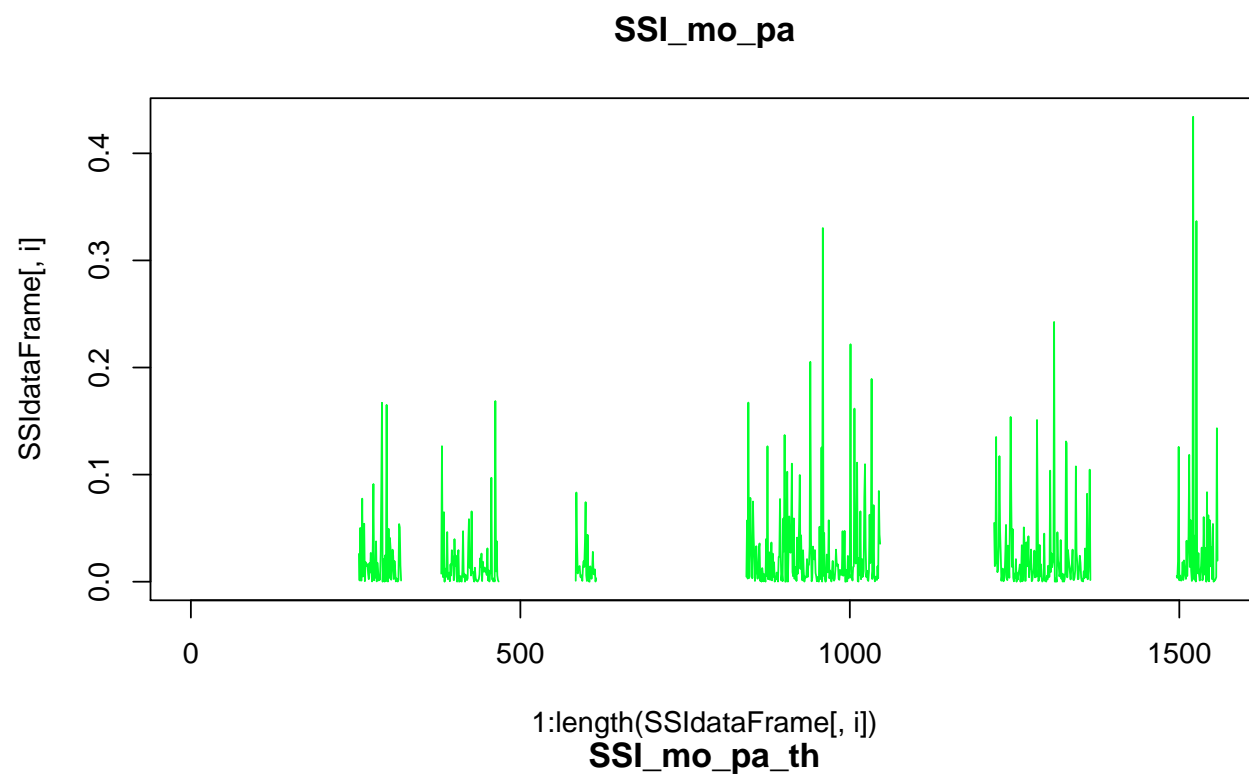
```
par(mar=c(4,4,4,4))
col <- 1
```

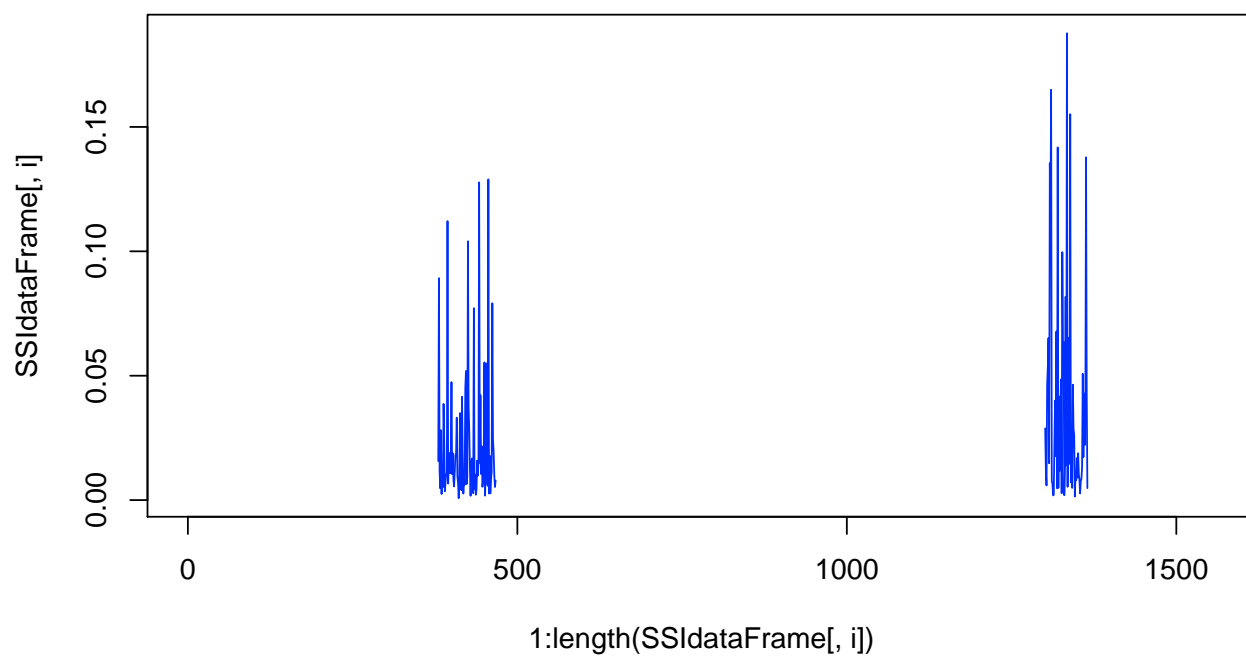
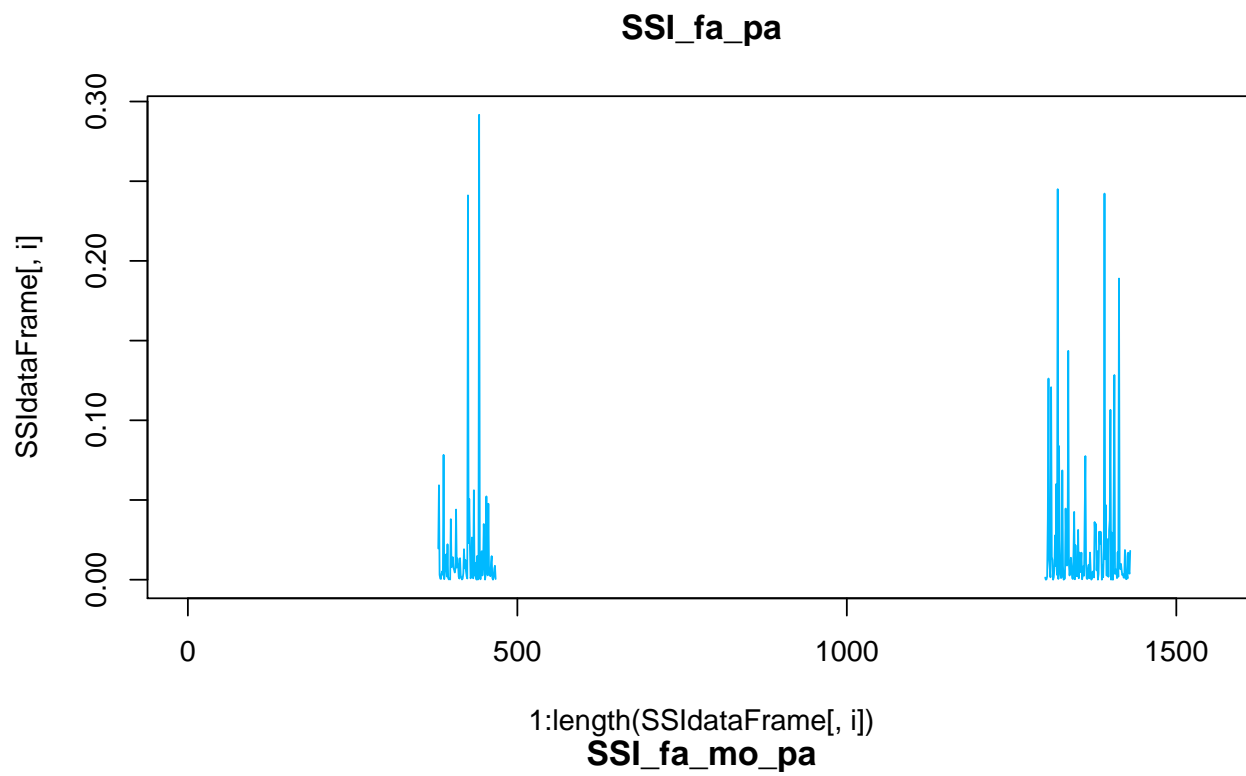


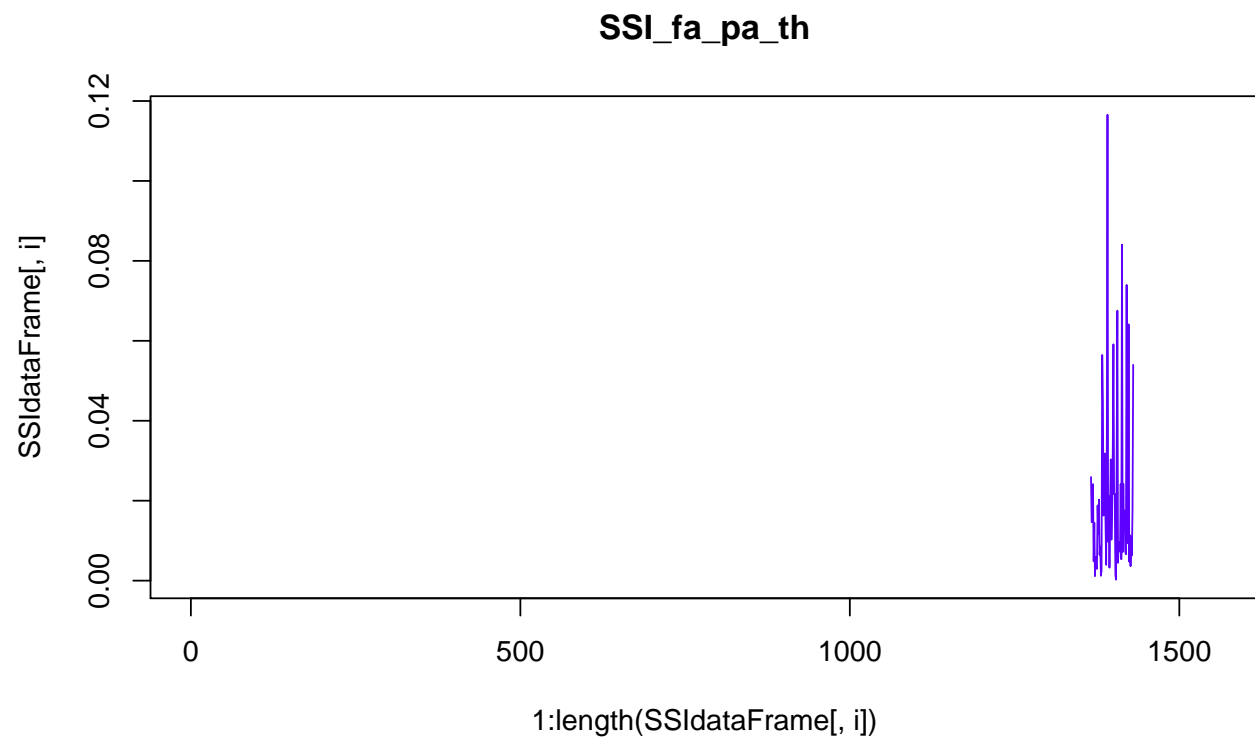
```
for (i in 6:length(SSIdataFrame)){
  plot(1:length(SSIdataFrame[,i]), SSIdataFrame[,i], type="l",
    col=rainbow(11)[col], main = names(SSIdataFrame)[i])
  col <- col+1}
```







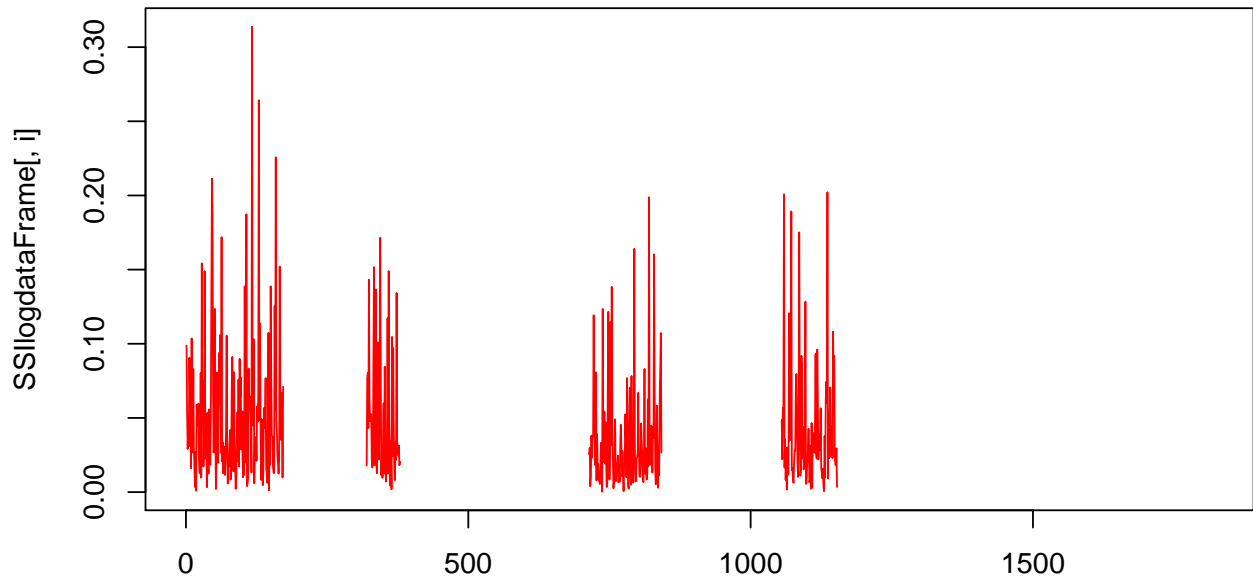




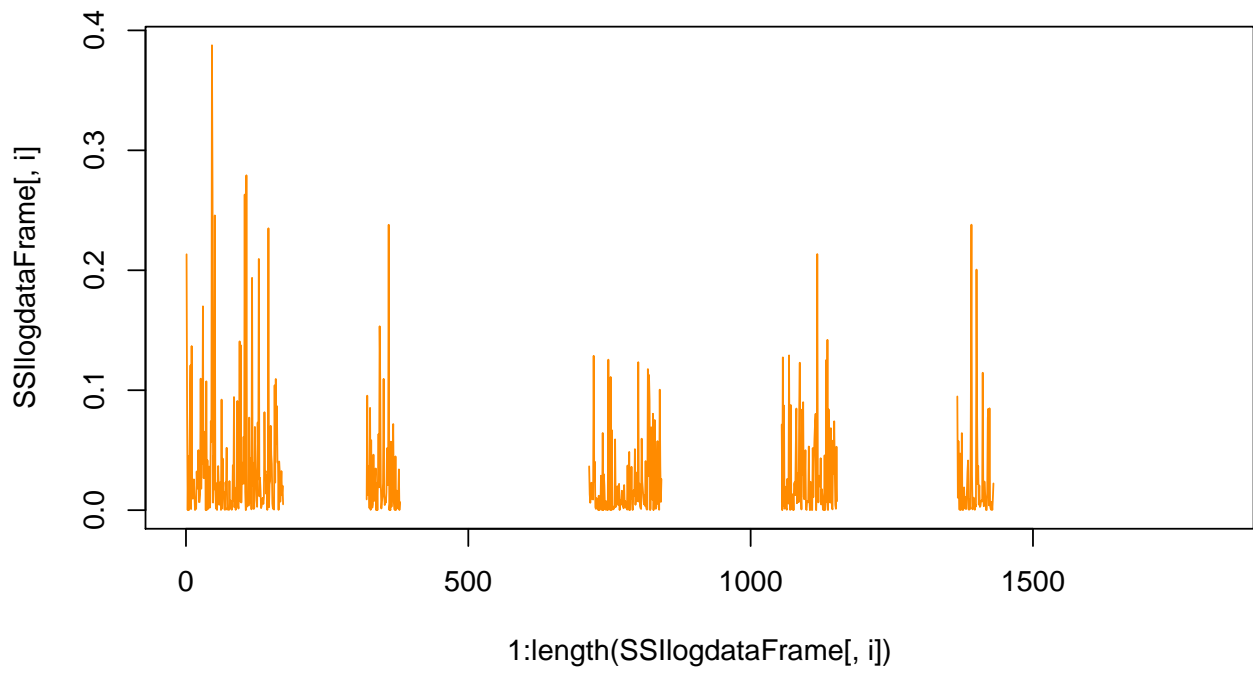
log data

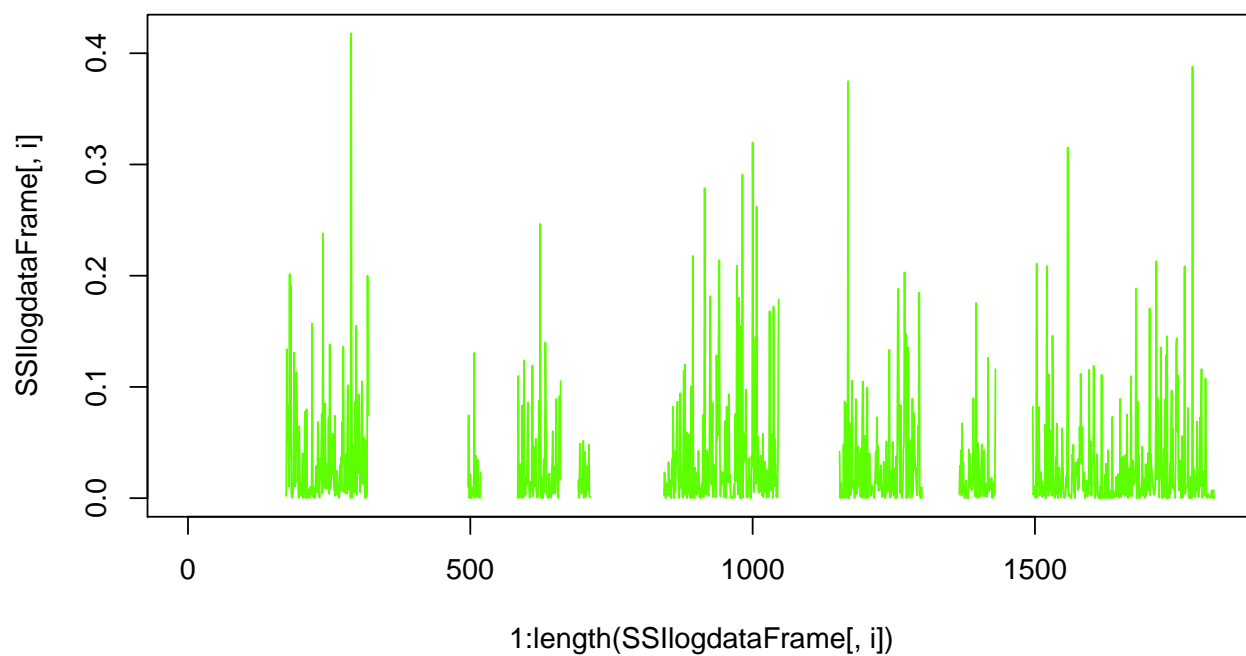
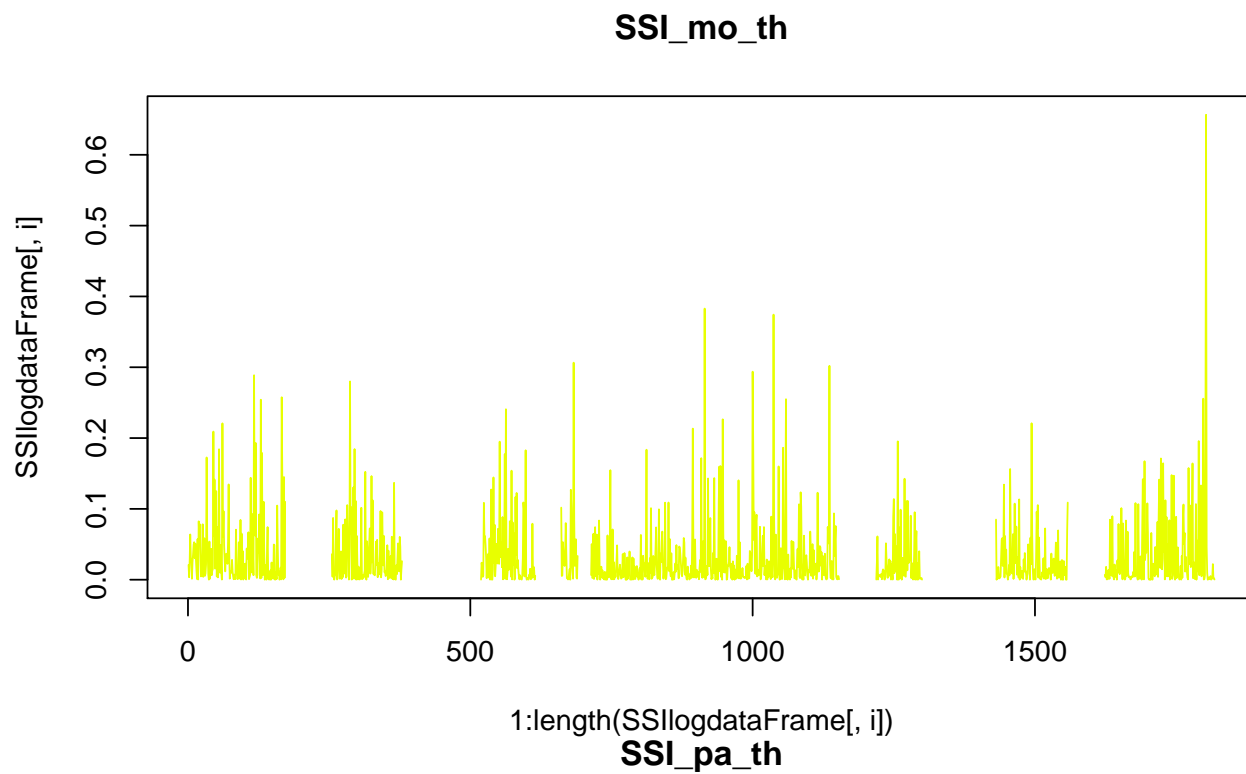
```
par(mar=c(4,4,4,4))
col <- 1
for (i in 6:length(SSIllogdataFrame)){
  plot(1:length(SSIllogdataFrame[,i]), SSIllogdataFrame[,i], type="l",
    col=rainbow(11)[col], main = names(SSIllogdataFrame)[i])
  col <- col+1}
```

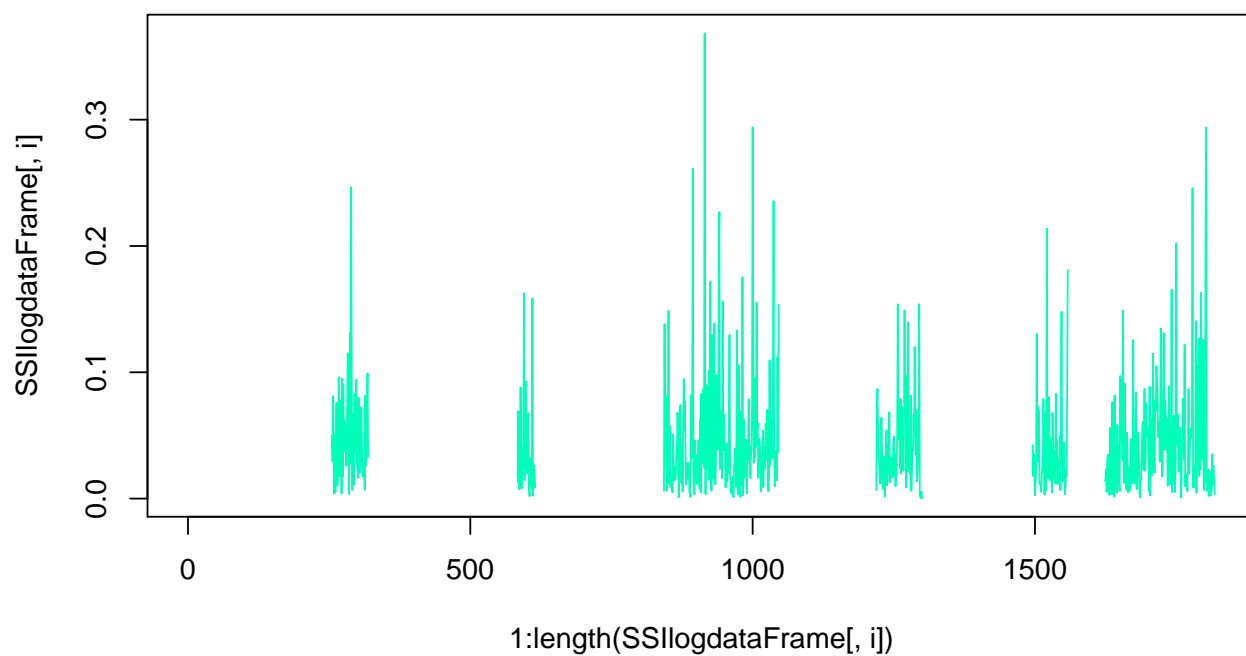
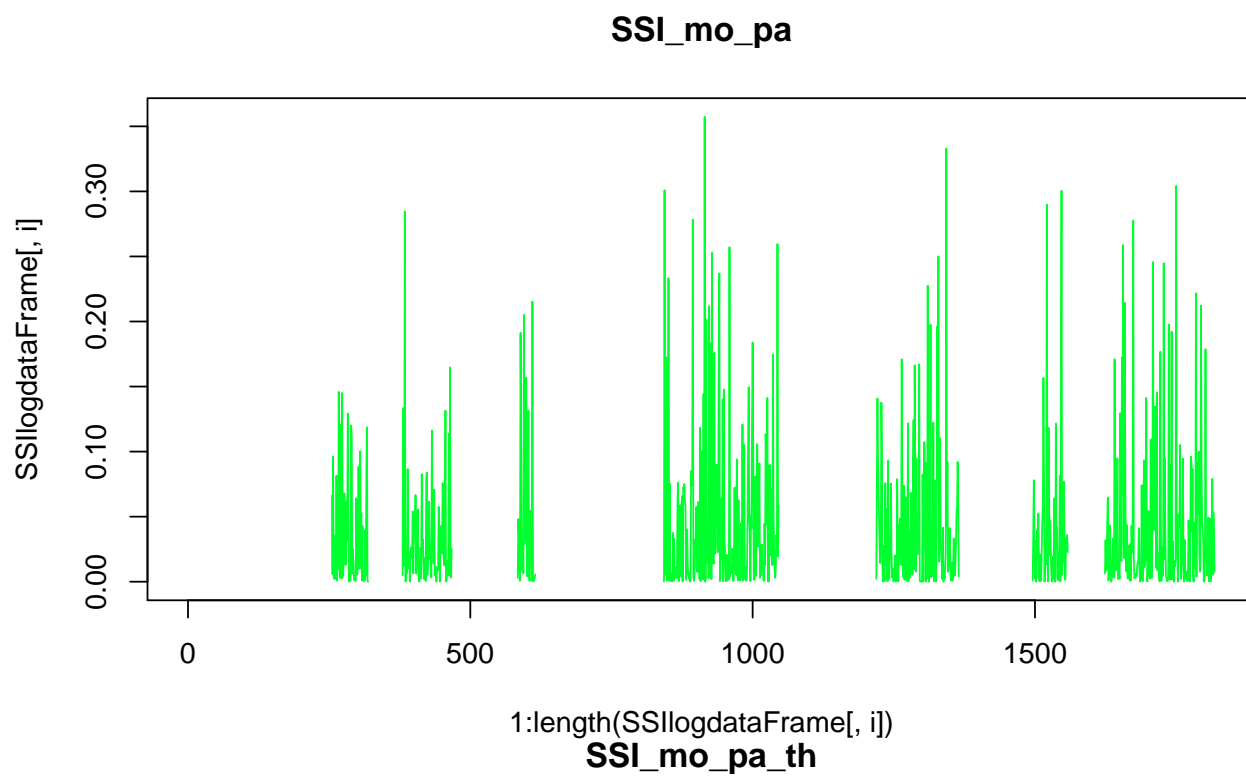
### SSI\_fa\_mo\_th



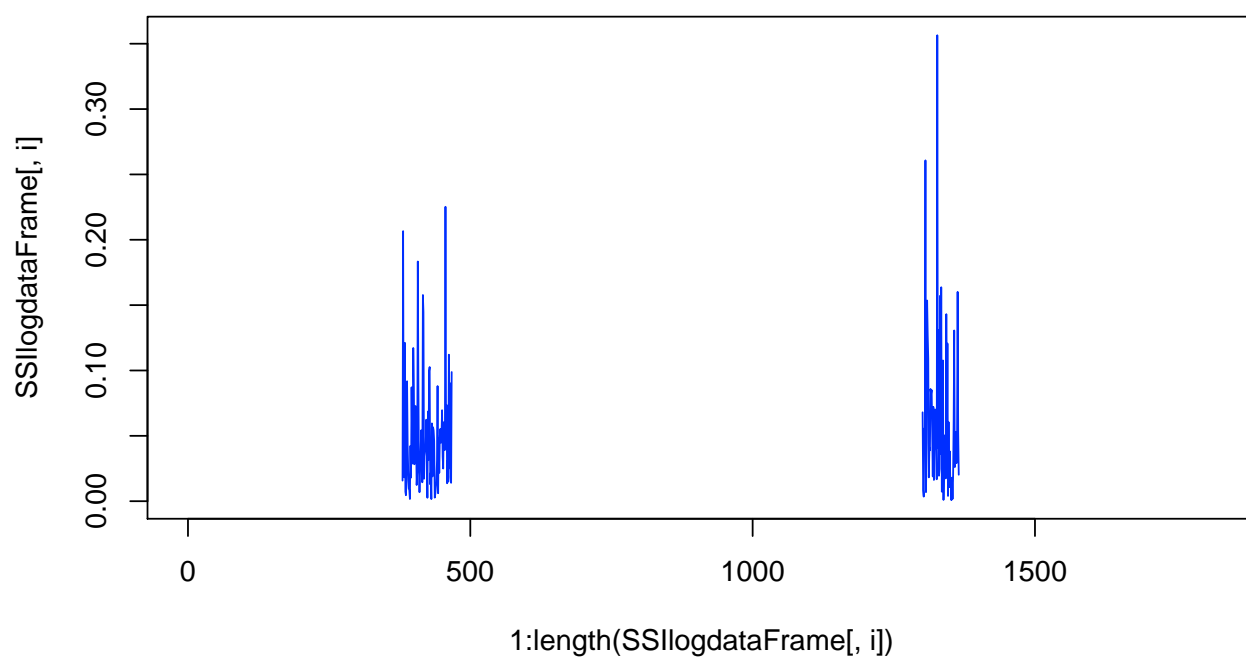
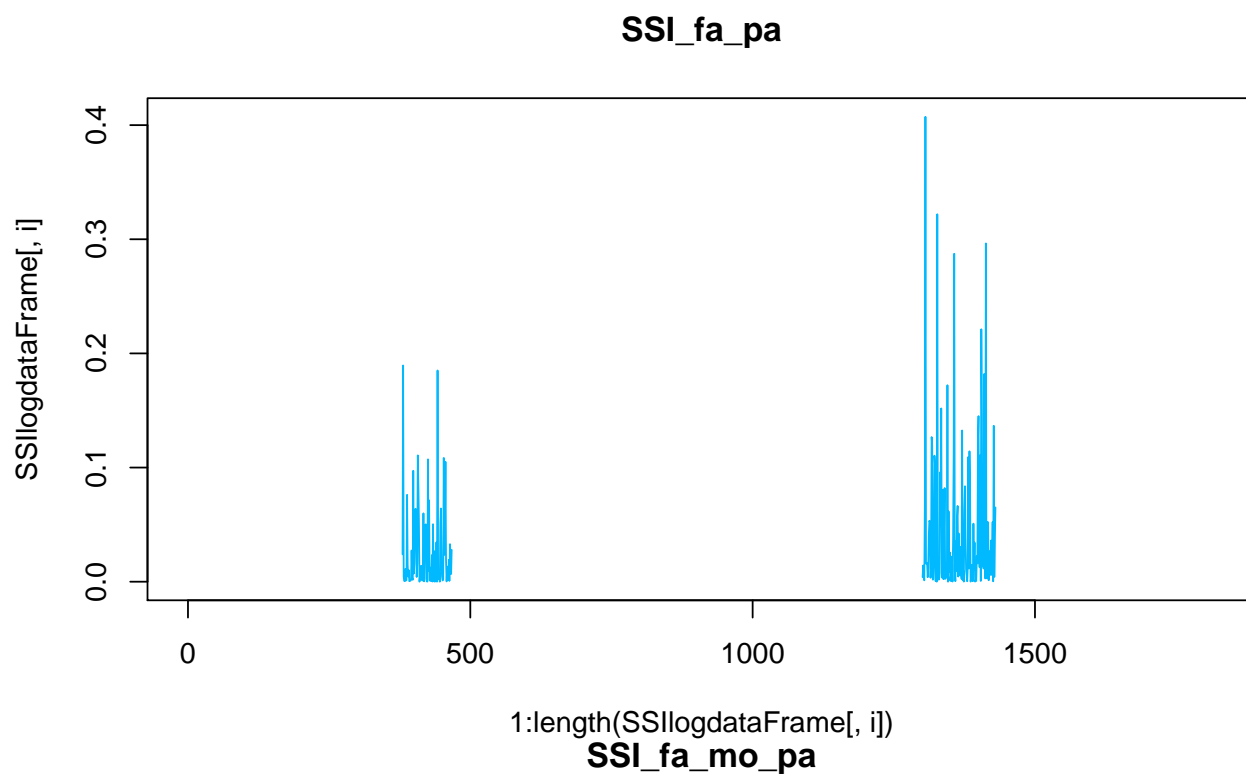
### SSI\_fa\_th

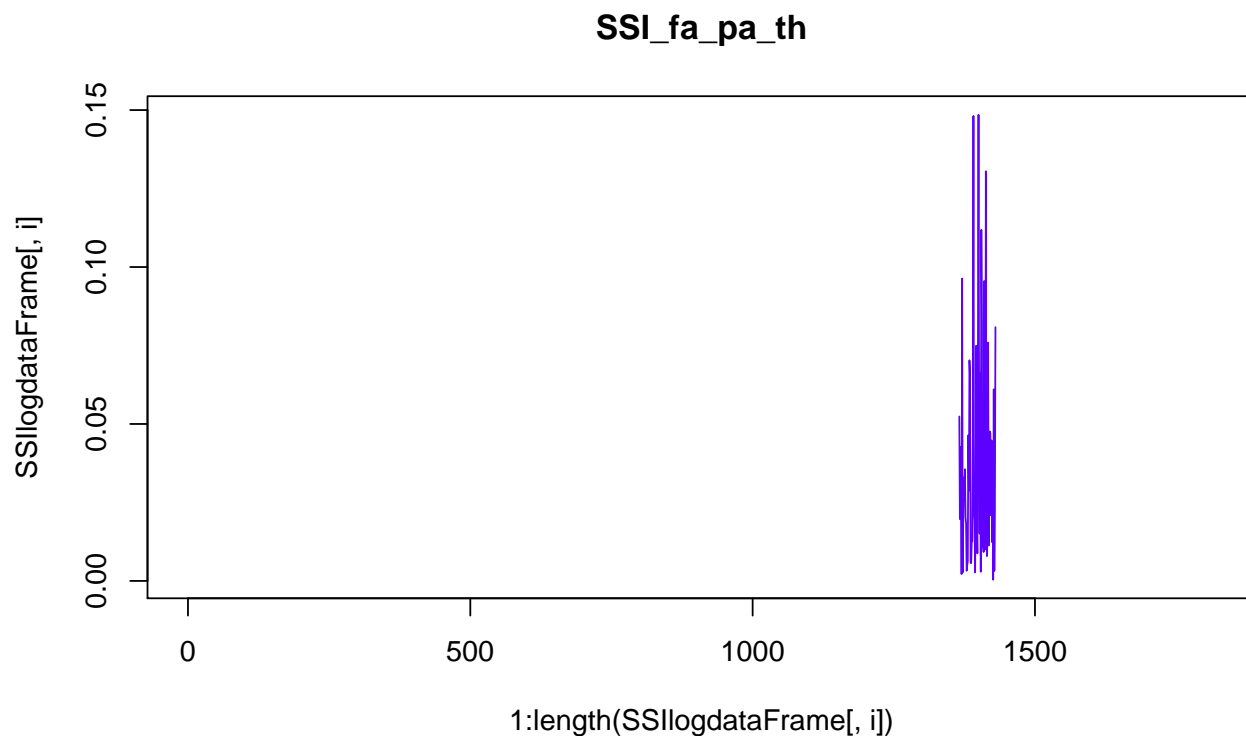










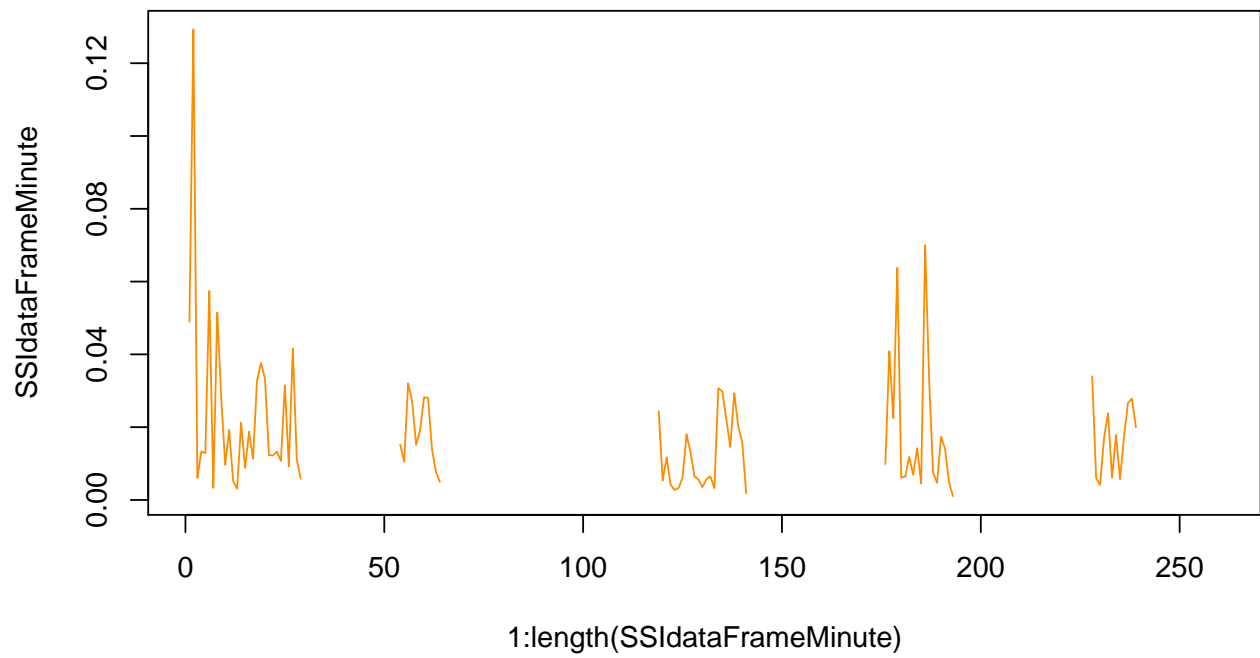
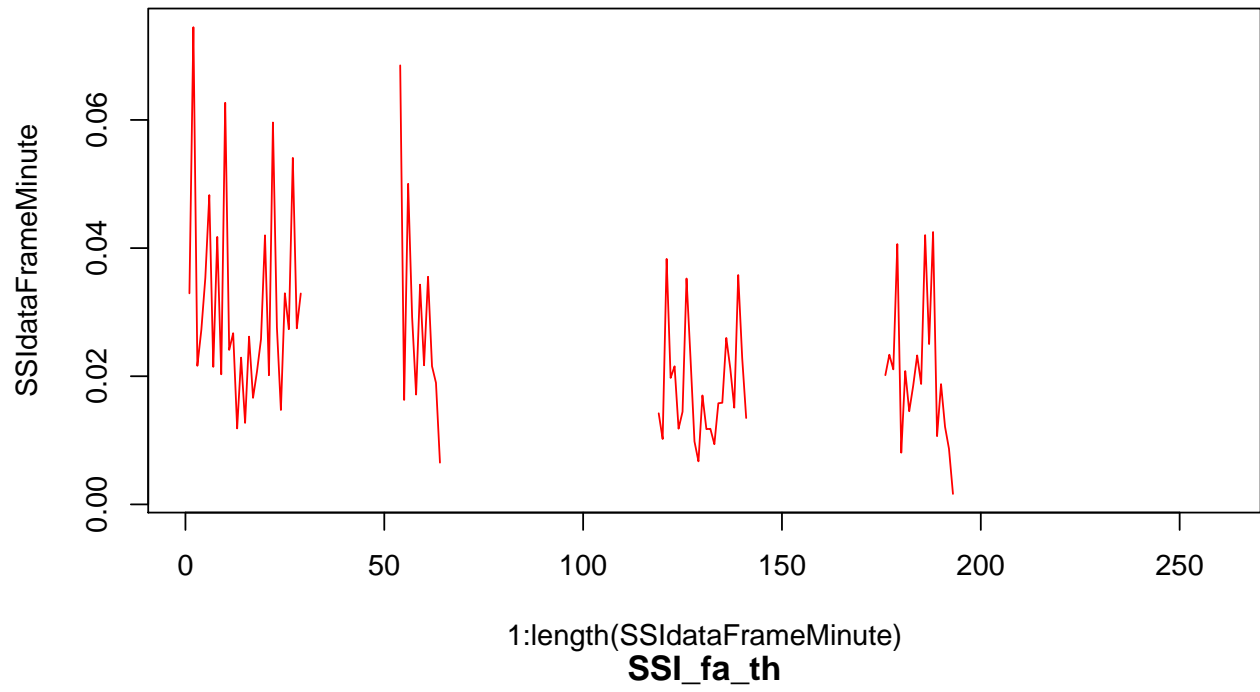


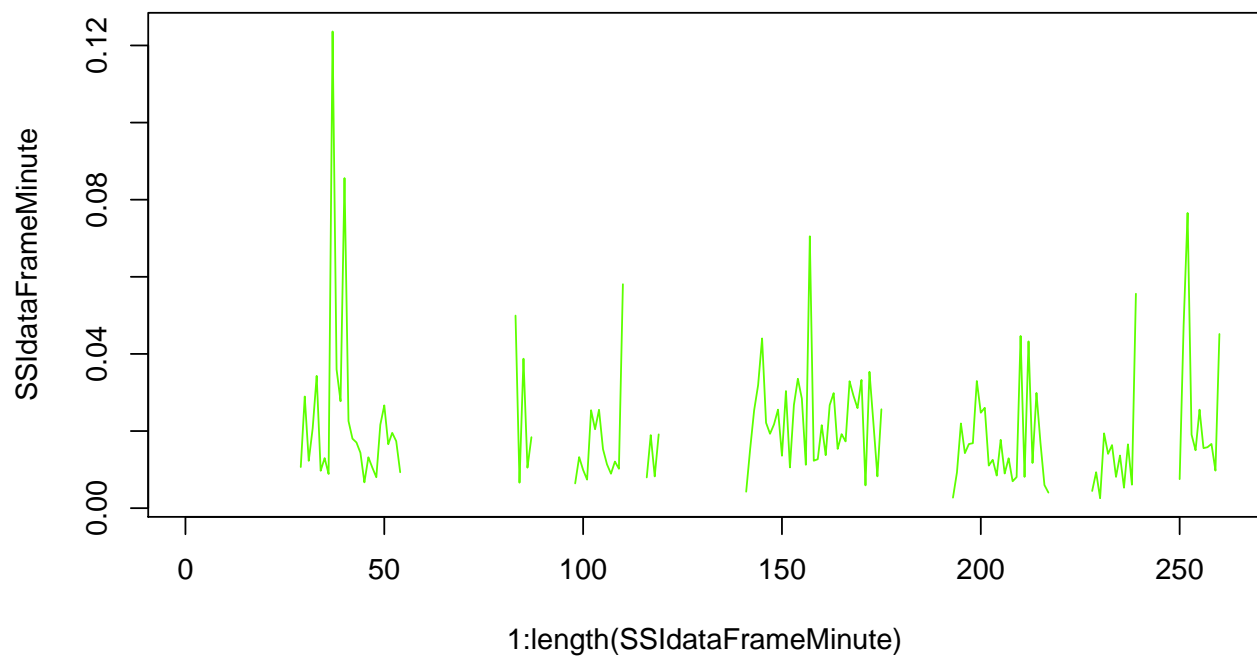
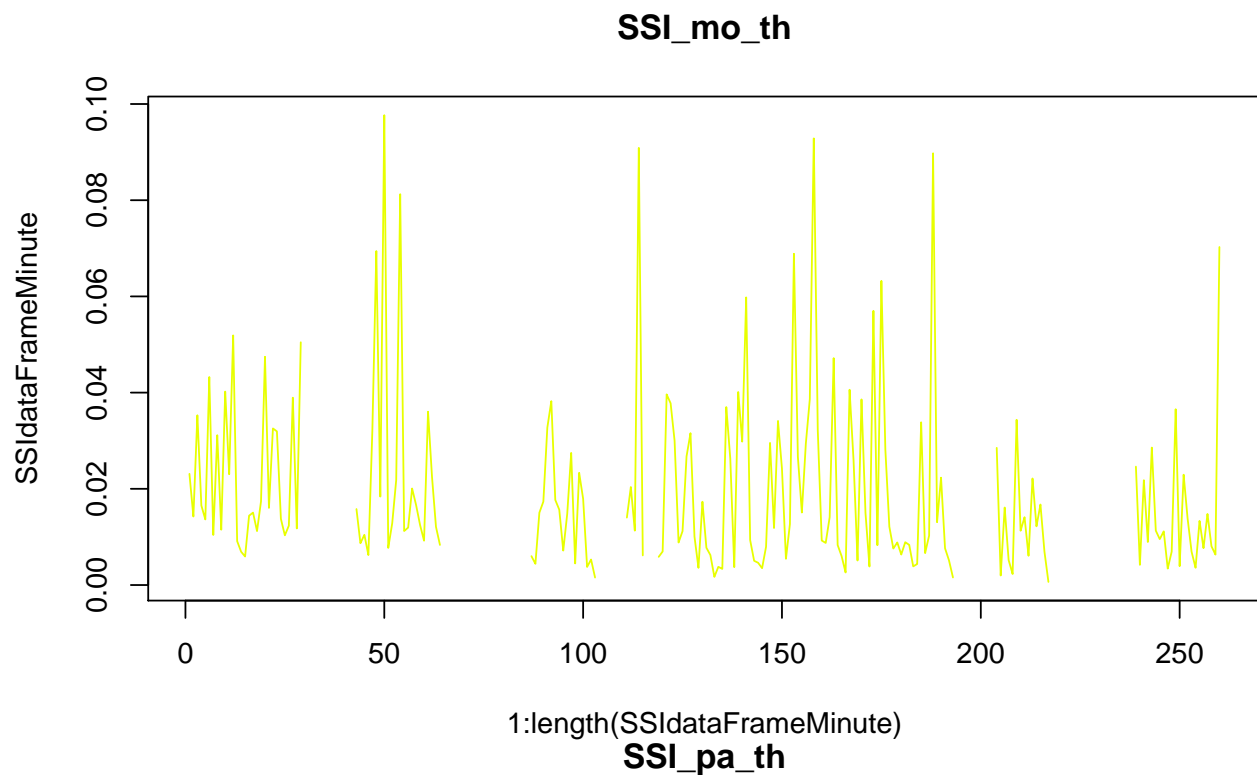
Evolution of synchrony through time, mean by minute

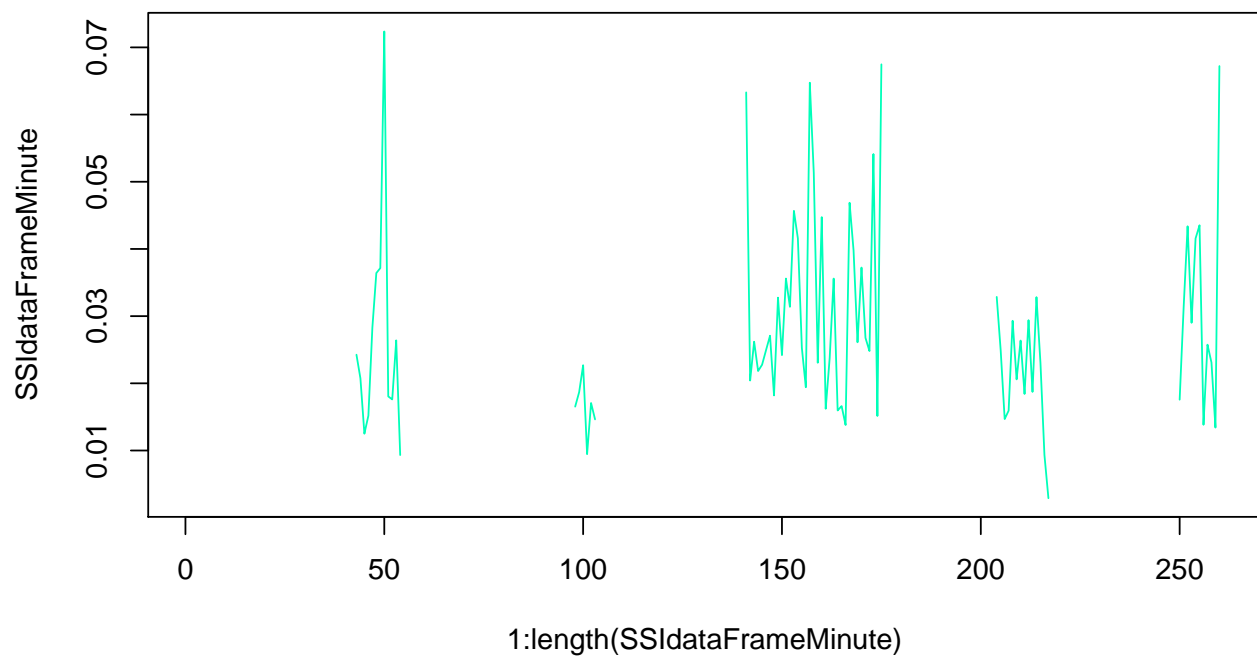
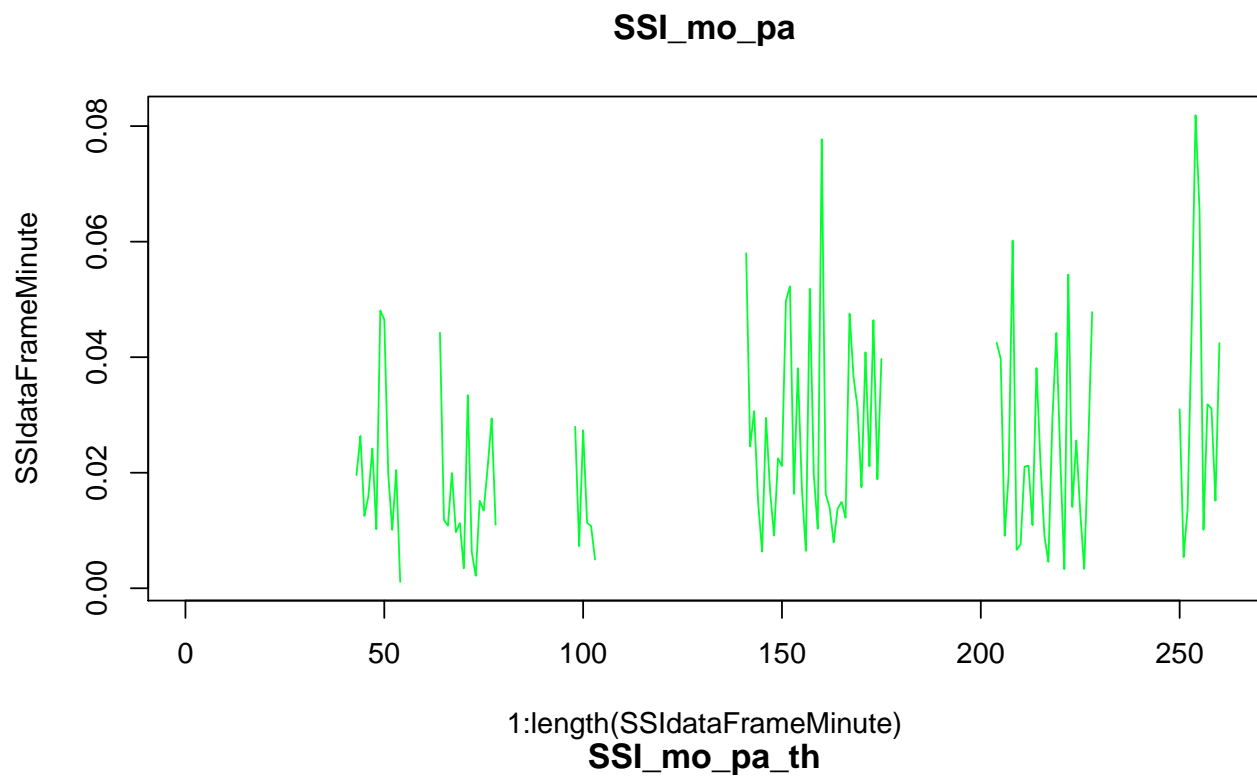
No log data

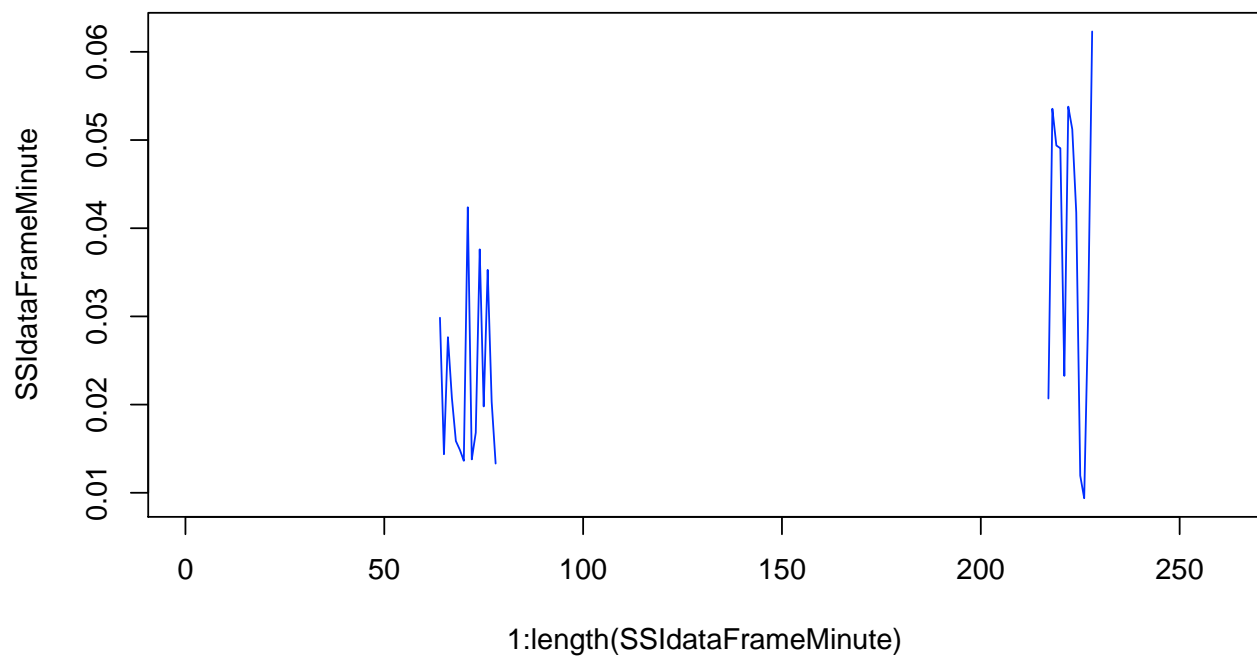
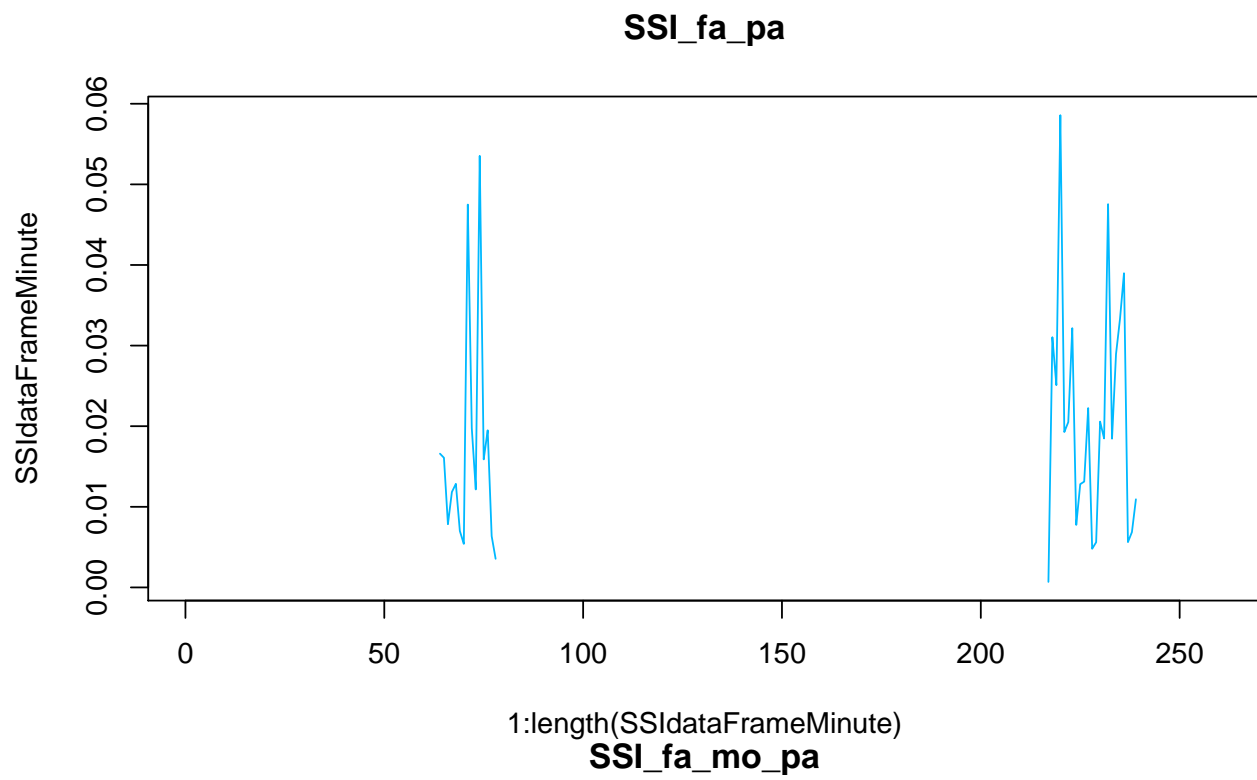
```
par(mar=c(4,4,4,4))
col = 1
for (indexSSI in 6:length(SSIdataFrame)){
  IntervalNumbersVideo <- ceiling(length(SSIdataFrame[,indexSSI])/6)
  SSIColumn <- SSIdataFrame[,indexSSI]
  SSIdataFrameMinute <- c()
  for (i in 1:IntervalNumbersVideo){
    borneInf <- 1+(i-1)*6
    borneSup <- i * 6
    SSIVectorInterval <- SSIColumn[borneInf:borneSup]
    mean <- mean(SSIVectorInterval, na.rm=TRUE)
    SSIdataFrameMinute <- c(SSIdataFrameMinute, mean)}
  plot(1:length(SSIdataFrameMinute), SSIdataFrameMinute, type="l", col=rainbow(11)[col], main = names
  col <- col+1}
```

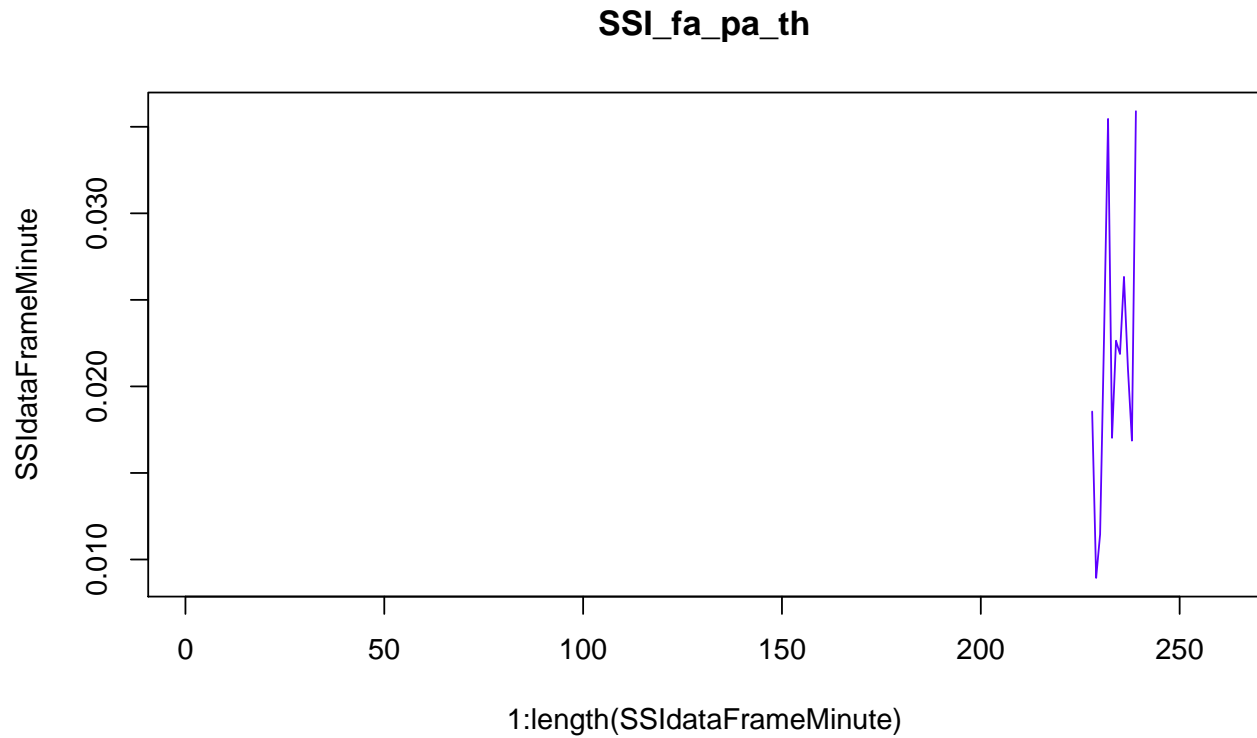
### SSI\_fa\_mo\_th







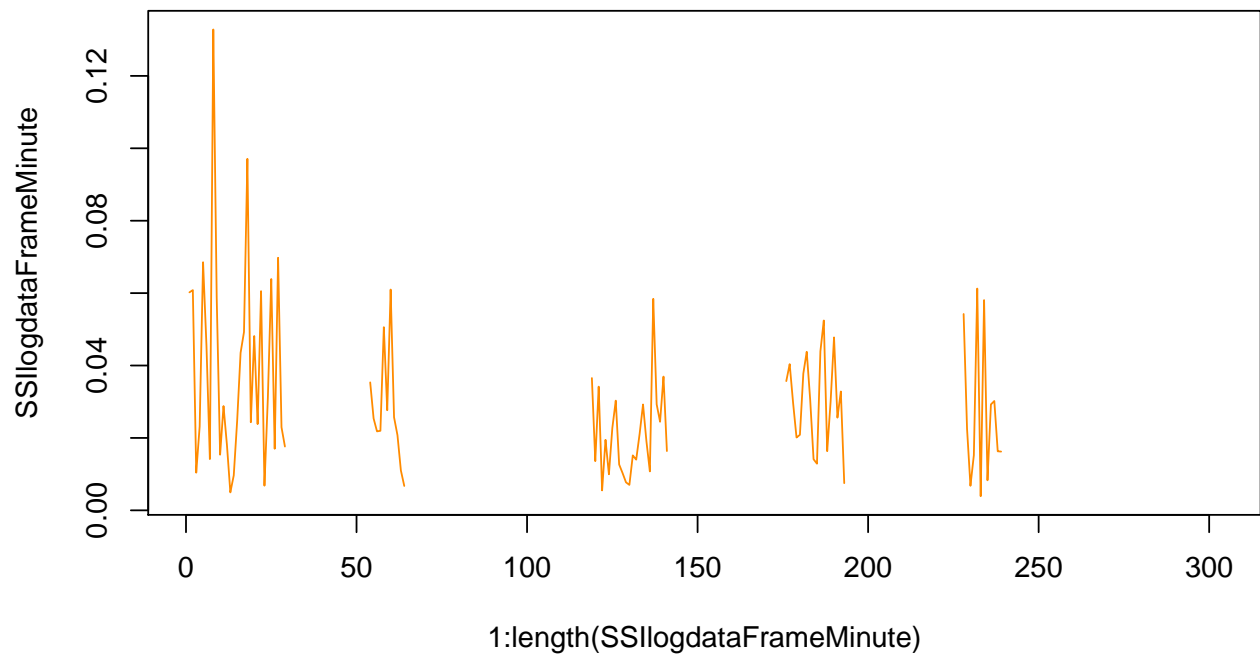
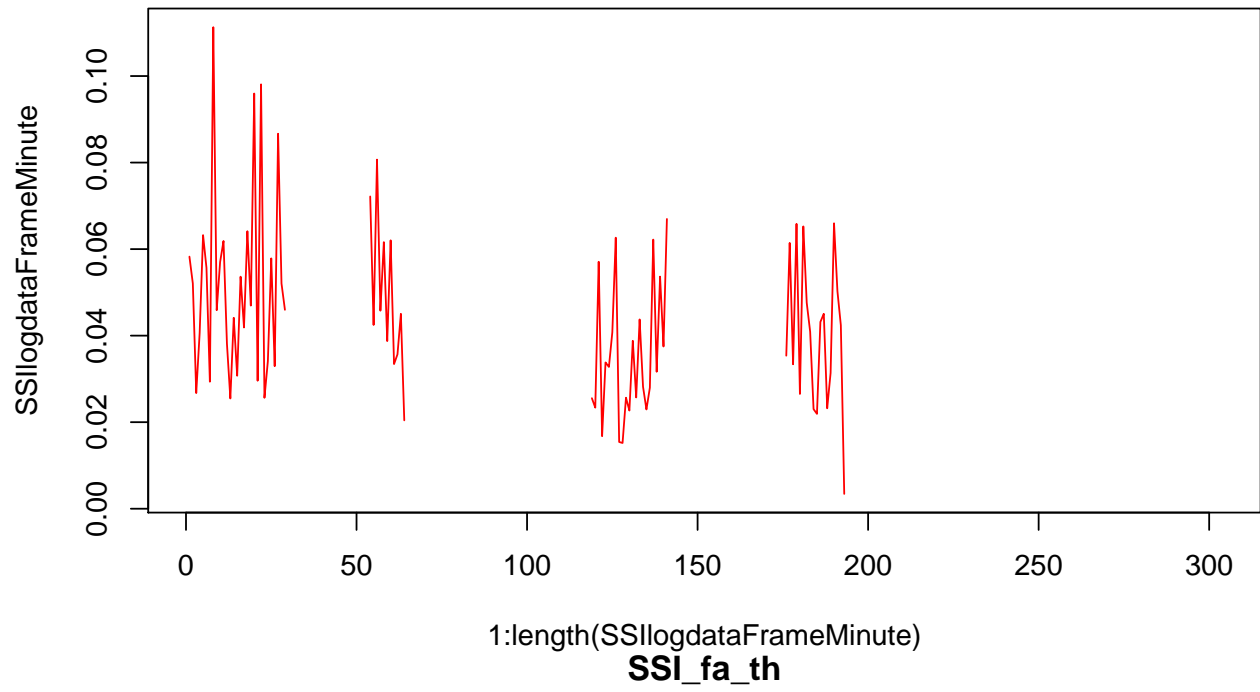




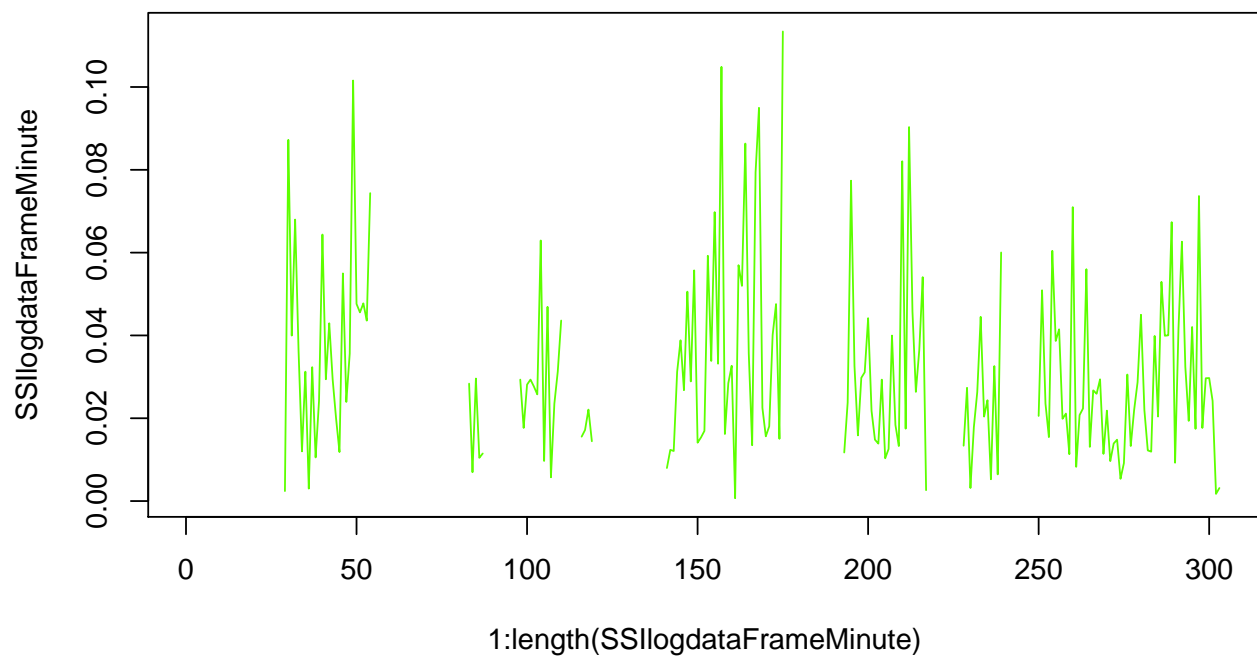
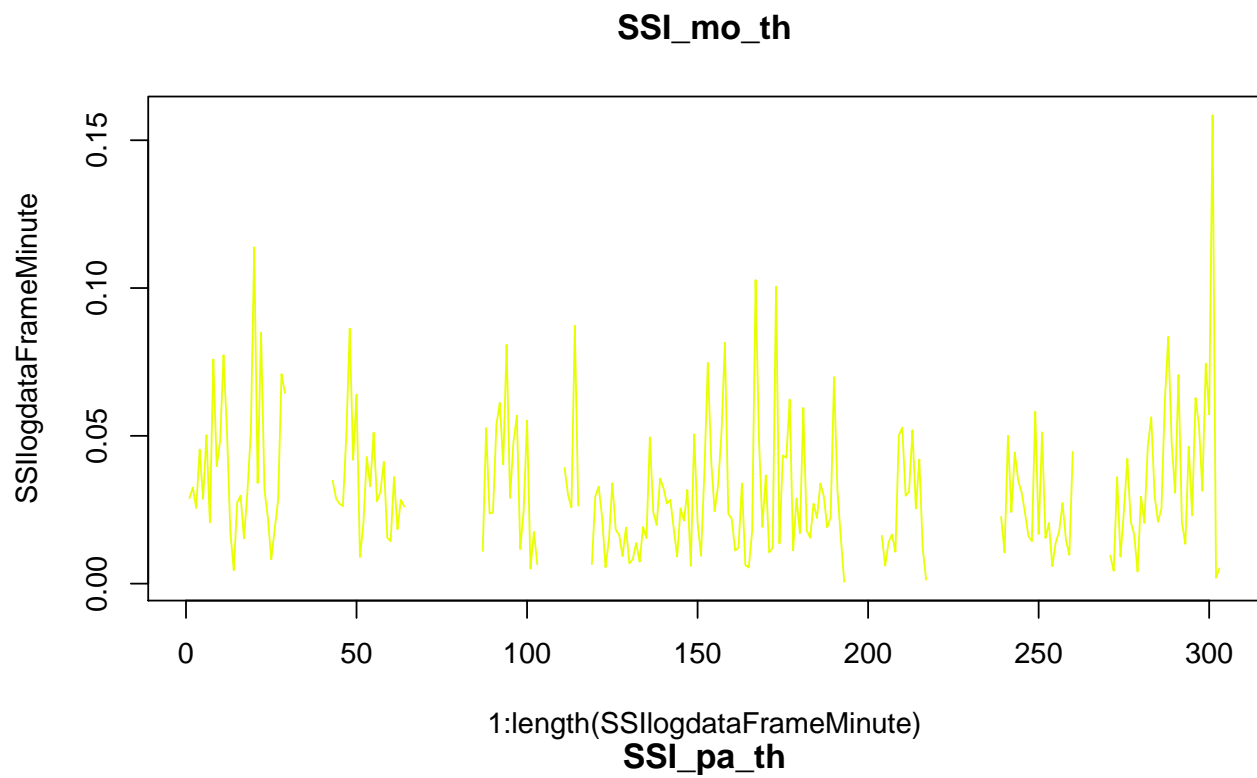
#### Log data

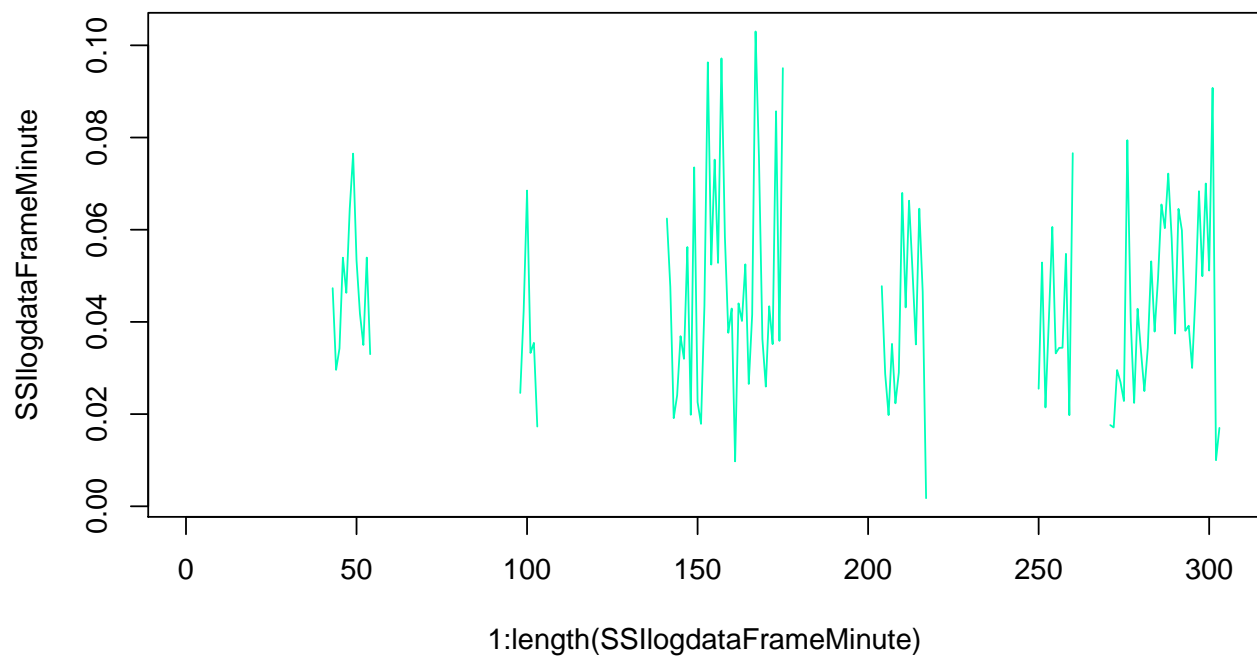
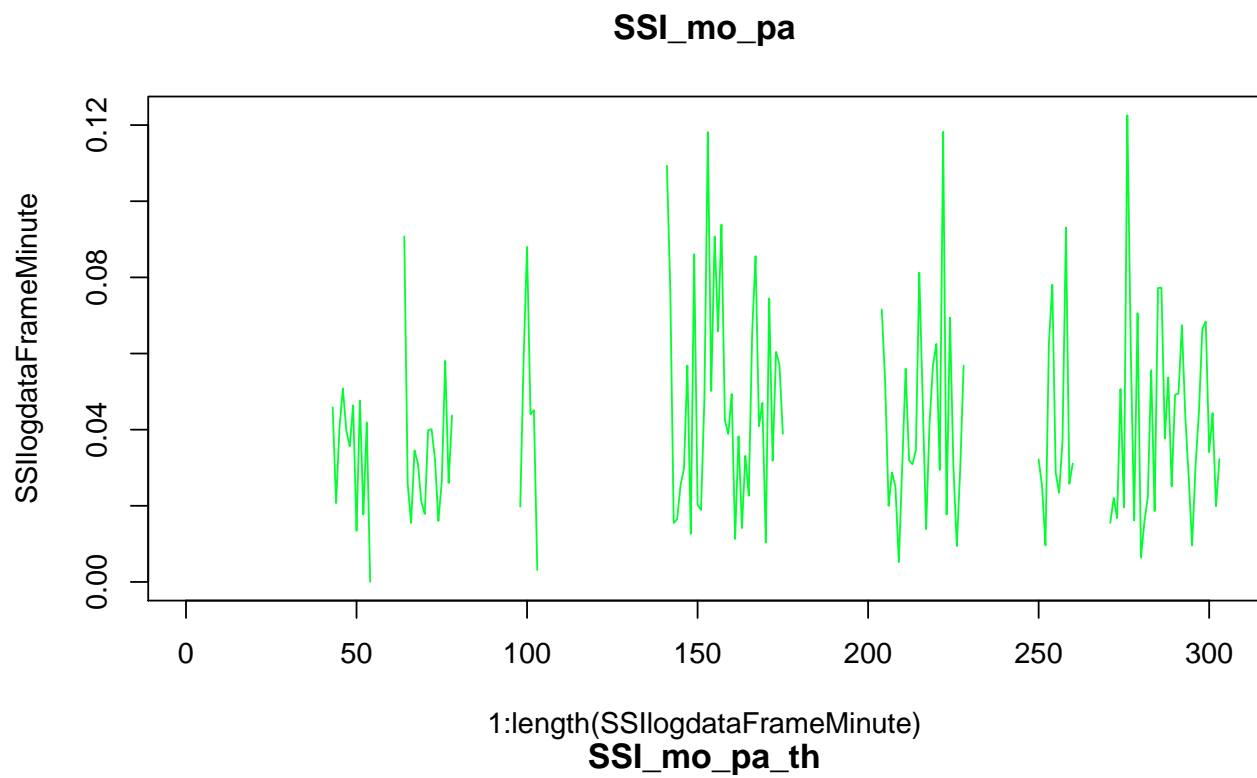
```
par(mar=c(4,4,4,4))
col = 1
for (indexSSI in 6:length(SSIllogdataFrame)){
  IntervalNumbersVideo <- ceiling(length(SSIllogdataFrame[,indexSSI])/6)
  SSIllogColumn <- SSIllogdataFrame[,indexSSI]
  SSIllogdataFrameMinute <- c()
  for (i in 1:IntervalNumbersVideo){
    borneInf <- 1+(i-1)*6
    borneSup <- i * 6
    SSIllogVectorInterval <- SSIllogColumn[borneInf:borneSup]
    mean <- mean(SSIllogVectorInterval, na.rm=TRUE)
    SSIllogdataFrameMinute <- c(SSIllogdataFrameMinute, mean)}
  plot(1:length(SSIllogdataFrameMinute), SSIllogdataFrameMinute, type="l", col=rainbow(11)[col], main =
  col <- col+1}
```

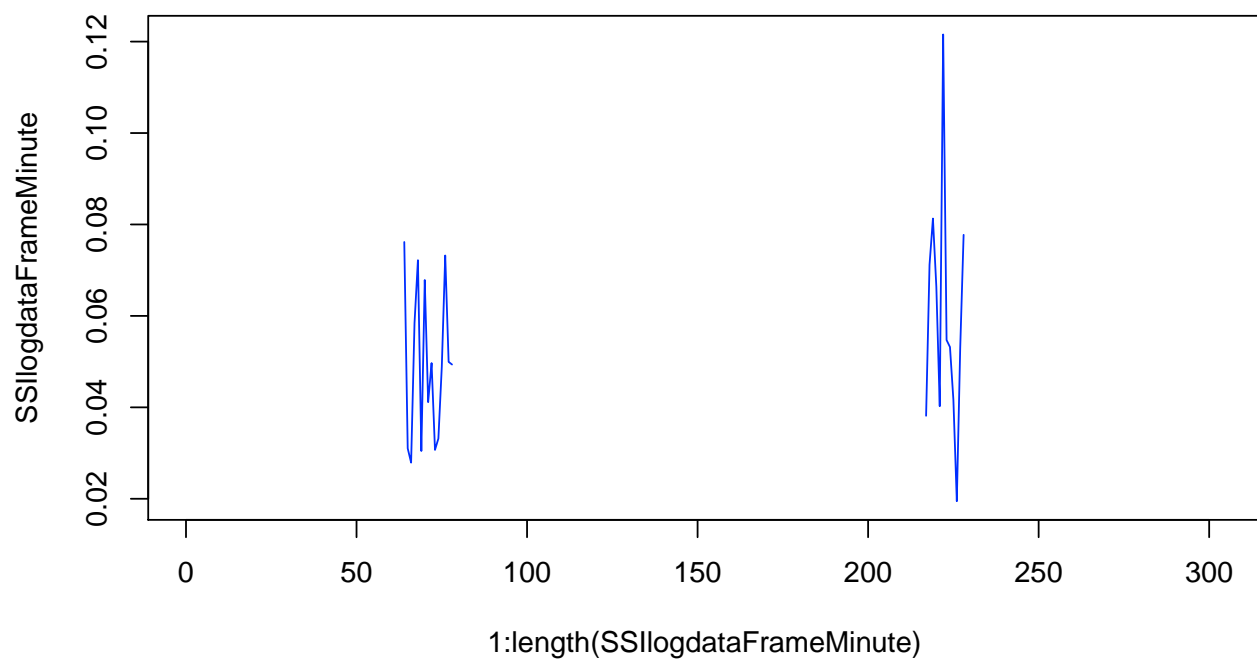
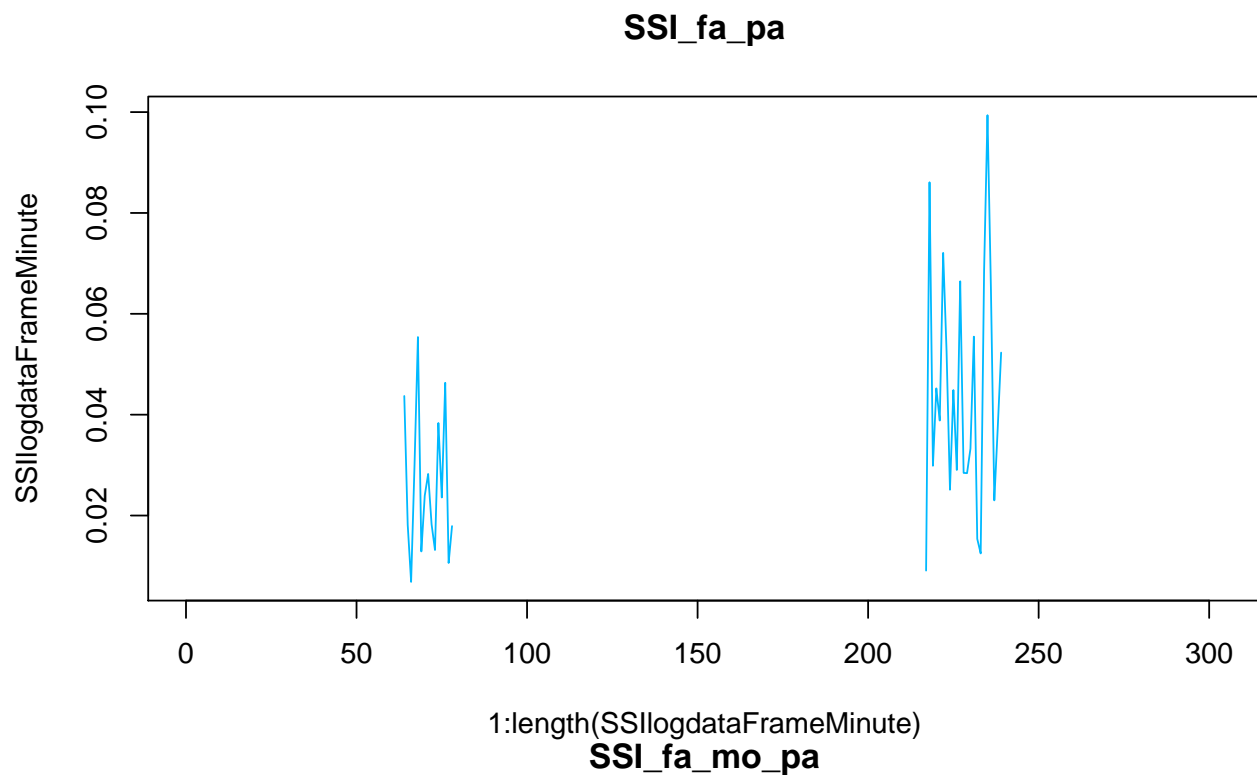
### SSI\_fa\_mo\_th

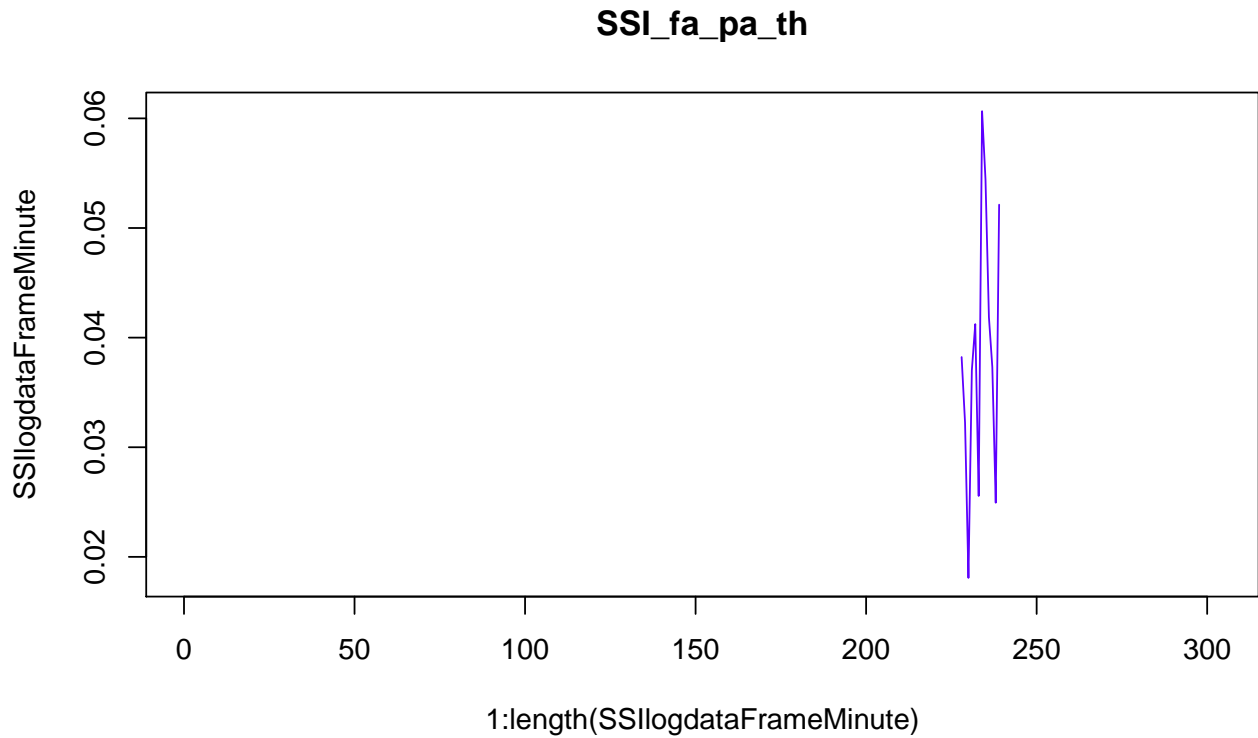












Evolution of synchrony through time, mean by 10 minutes

```
par(mar=c(4,4,4,4))
col = 1
for (indexSSI in 6:length(SSIdataFrame)){
  IntervalNumbersVideo <- ceiling(length(SSIdataFrame[,indexSSI])/60)
  SSIColumn <- SSIdataFrame[,indexSSI]
  SSIdataFrameTenMinute <- c()
  for (i in 1:IntervalNumbersVideo){
    borneInf <- 1+(i-1)*60
    borneSup <- i * 60
    SSIVectorInterval <- SSIColumn[borneInf:borneSup]
    mean <- mean(SSIVectorInterval, na.rm=TRUE)
    SSIdataFrameTenMinute <- c(SSIdataFrameTenMinute, mean)}
  plot(1:length(SSIdataFrameTenMinute), SSIdataFrameTenMinute, type="l", col=rainbow(11)[col], main =
  col <- col+1}
```

Models of synchrony

```
SSI_fa_th_lme <- lmer(SSI_fa_th ~ Time_min + (1|video), data=SSIdataFrame)
summary(SSI_fa_th_lme)
#plot(SSI_fa_th_lme)
res <- residuals(SSI_fa_th_lme)
hist(SSIdataFrame$SSI_fa_th)
qqnorm(res)
SSI_fa_th_List <- c()
for (i in indexList){
```

```

    SSI_fa_th_List <- c(SSIdataFrame$SSI_fa_th_List, mean(SSIdataFrame[which(SSIdataFrame$video==i),]$SSI_fa_th, na.rm=
  })
print(SSIdataFrame$SSI_fa_th_List)
#plot(SSIdataFrame$SSI_fa_th_List, type="b")

# log of the data
log_SSI_fa_th <- hist(log(SSIdataFrame$SSI_fa_th))
SSIdataFrame_log_lme <- lmer(log(SSIdataFrame$SSI_fa_th) ~ Time_min + (1|video), data=SSIdataFrame)
res_log <- residuals(SSIdataFrame_log_lme)
qqnorm(res_log)
summary(SSIdataFrame_log_lme)

# root square of the data
sq_SSI_fa_th <- hist(sqrt(SSIdataFrame$SSI_fa_th))
SSIdataFrame_sq_lme <- lmer(sqrt(SSIdataFrame$SSI_fa_th) ~ Time_min + (1|video), data=SSIdataFrame)
res_sq <- residuals(SSIdataFrame_sq_lme)
qqnorm(res_sq)
summary(SSIdataFrame_sq_lme)

```