

# From Noldus File To Visualisation part 6 : Making graphs with R

Clair Chaigne

10 novembre 2021

## 1. INSTRUCTIONS

This is a follow up from `From Noldus File to Visualisation part 5 : Phase Analysis with Biodare`

For this example I used the `DDopn4xa` experiments. It includes 3 independent experiments : `PSDD1opn4xa`, `PSDD2opn4xa` and `PSDD3opn4xa`, that I trimmed so they all start at 23:30:00).

For making plots, I used the `faire_plots` R script that can be found in :

```
Y://ELISE DATA_17Gb a trier//EXCELL+stats//excell//zebrafish locomotor activity//Clair_bilans//juin2021
```

## 2. STEP BY STEP EXAMPLE

**1. load the following libraries** (the `plotly` library is optional, I use it to make interactive plots)

```
library(dplyr)
library(ggplot2)
library(plotly)#interactive plot
library(readxl)
library(behavr)
library(ggetho)
library(rstatix)
```

**2. load the color palette.** It's supposed to be a colorblind friendly palette

```
cbbPalette <- c("#000000", "#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00", "#CC79A7") # palette colorblindfriendly
```

**3. specify the folder where your files to plot are.** Change for your appropriate directory.

```
setwd("Y://ELISE DATA_17Gb a trier//EXCELL+stats//excell//zebrafish locomotor activity//Clair_bilans//juin2021//1.data//2.stats_mmdeb")
```

**4. load the functions necessary for making plots**

##### FUNCTIONS #####

```
#1.transformer data pour ggplot // transform data for ggplot
transfo_data<-function(fichier, lignes, colwt, nwt, colmut, nmut, minutes,conditionwt,condition
mut){
  #fichier : fichier a transformer // file to transform
  #lignes : nombre de lignes du fichier 我轮 consid我特rer // number of lines of the file to c
onsider
  #colwt : num我特ro de la premi我振re colonne wt (les wt doivent 我振tre 我轮 la suite les uns
des autres) // number of the first column containing wt (wt all needs to be next each other)
  #nwt : nombre de wt // number of wt
  #colmut : num我特ro de la premi我振re colonne mut (les mut doivent 我振tre 我轮 la suite les
uns des autres) // number of the first column containing mutants (mutants all needs to be next
each other)
  #nmut : nombre de mut // number of wt
  #minutes : intervalle de temps en minutes (DM10, DM60 ?) // time bin in minutes
  #conditionwt : nom de la condition wt (controle, wt, non pigmente) // name of the wild type
condition
  #conditionmut: nom de la condition mut(mutant, opn4xa-/-, pigmente, etc) // name of the mut
ant condition
  wt<-print(paste(nwt,conditionwt,sep=" "))
  mut<-print(paste(nmut,conditionmut,sep=" "))
  data<-as.data.frame(cbind(time=seq(from=0,length.out=lignes,by=mins(minutes)), arena=colnames
(fichier[colwt]), genotype=wt, distance=unlist(fichier[c(1:lignes),colwt],use.names=FALSE)), st
ringsAsFactors=FALSE)
  for (i in 1:(nwt-1)){
    n<-i+colwt
    data<-as.data.frame(rbind(data, as.data.frame(cbind(time=seq(from=0,length.out=lignes,by=mi
ns(minutes)), arena=colnames(fichier[n]), genotype=wt, distance=unlist(fichier[c(1:lignes),n],
use.names=FALSE)))),stringsAsFactors=FALSE)
  }

  for (i in 0:(nmut-1)){
    n<-i+colmut
    data<-as.data.frame(rbind(data, as.data.frame(cbind(time=seq(from=0,length.out=lignes,by=mi
ns(minutes)), arena=colnames(fichier[n]), genotype=mut, distance=unlist(fichier[c(1:lignes),n],
use.names=FALSE)))),stringsAsFactors=FALSE)
  }
  return(data)
}

#2. creer fichier metadata pour rethomics // create metadata file for rethomics
create_meta<-function(data,nwt,nmut,conditionwt,conditionmut){
  wt<-print(paste(nwt,conditionwt,sep=" "))
  mut<-print(paste(nmut,conditionmut,sep=" "))
  data_meta<-data.table::data.table(arena=unique(data$arena), genotype=c(rep(wt,nwt),rep(mut,nm
ut)),key="arena")
  return(data_meta)
}

#3. creer fichier behavr pour rethomics // create behavr file for rethomics
create_behavr<-function(data,data_meta){
  data_behavr<-behavr(as.data.table(data,key="arena"),data_meta)
  return(data_behavr)
}

#4. creer le plot avec rethomics // create plot with rethomics (by default error bars are sta
ndard errors)
plot_ggetho<-function(data,titre,phaseLD,couleurs,couleurwt="#0072B2",couleurmutant="#E69F00"){
  #data : fichier behavr // behavr file
  #nwt : nombre de wt // number of wt
```

```

#nmut : nombre de mutants // number of mutants
#titre : titre (entre guillemets) // title (between " ")
#phaseLD : en secondes, temps avant la premiere transition nuit/jour // in seconds, time before the first night/dark transition
#couleurs : vecteur qui contient les couleurs du bandeau (c("grey","black")) par exemple // vector containing colors of the horizontal bands
intervalles<-c(phaseLD,phaseLD+hours(14),phaseLD+hours(14)+hours(10),phaseLD+hours(14)+hours(10)+hours(14),phaseLD+hours(14)+hours(10)+hours(14)+hours(10),phaseLD+hours(14)+hours(10)+hours(14)+hours(10)+hours(14)+hours(10),phaseLD+hours(14)+hours(10)+hours(14)+hours(10)+hours(14)+hours(10)+hours(14),phaseLD+hours(14)+hours(10)+hours(14)+hours(10)+hours(14)+hours(10)+hours(14)+hours(10)+hours(14)+hours(10))
p<-gggeth(data,aes(x=as.numeric(time),y=as.numeric(distance),color=factor(genotype,levels=c(wt,mut))))+
  scale_color_manual(values=c(couleurwt,couleurmutant))+
  scale_fill_manual(values=c(couleurwt,couleurmutant))+
  stat_pop_etho()+
  labs(title=titre, x="Time", y="Mean distance travelled (mm/min over 10min)")+
  scale_x_discrete(expand=c(0,0))+
  geom_vline(xintercept=intervalles,color="darkgrey",linetype="dashed",lwd=1.4)+
  stat_ld_annotatons(l_duration=hours(14), phase=(phaseLD),period=hours(24),ld_colours=couleurs)+
  theme_minimal(base_line_size=1.4,base_rect_size=1)+theme(legend.title=element_blank())+geom_vline(xintercept=0)
  return(p)
}

#5. plot period mean +/- sd
dataperiod <- function(data, col1, col2){ # create data frame for the plot
  df <- as.data.frame(cbind(period=c(unlist(data[,col1]), unlist(data[,col2])), genotype=c(rep("wt", nrow(data)), rep("mut", nrow(data)))),stringsAsFactors = FALSE)
  df <- as.data.frame(cbind(period=df$period[!is.na(df$period)], genotype=df$genotype[!is.na(df$period)]))
  df$genotype <- factor(df$genotype,levels=c("wt","mut"))
  df$period <- as.numeric(df$period)
  return(df)
}

periodplot<-function(data,titre,colwt="#0072B2",colmut="#E69F00",wt="wt",mut="mut"){
  df.summary <- data %>%
    group_by(genotype) %>%
    summarise(
      sd = sd(period),
      period = mean(period)
    )
  df.summary
  p<-ggplot(data, aes(genotype, as.numeric(period),col=genotype)) +
    geom_jitter(
      position = position_jitter(0.2), color = "darkgray",size=4, alpha=0.4
    ) +
    geom_pointrange(
      aes(ymin = (period-sd), ymax = (period+sd)),size = 2.5,shape=15,alpha=0.8,data = df.summary
    )+ylim(20,30)+scale_x_discrete(limits=c(wt,mut))+
    labs(title=titre, y="period")+scale_color_manual(values=c(colwt,colmut))+theme_light(base_line_size=2.5,base_rect_size = 1.4)+scale_x_discrete(breaks = NULL)
  return(p)
}

```

```
DD_opn4xa<-read_excel("DDopn4xa_DD1-DD2-DD3.xlsx")
```

[illegible]

```
DD_opn4xa_data<-transfo_data(DD_opn4xa,lignes=583,colwt=3,nwt=65,colmut=68,nmut=65, minutes = 1
0, conditionwt="opn4xa+/+" ,conditionmut="opn4xa-/-")
```

This will transform *(DD\_opn4xa)* in *(DD\_opn4xa\_data)*

FromNoldusFileToVisualisation_5.Rmd		FromNoldusFileToVisualisation_6.Rmd*		DD_opn4xa	DD_opn4xa_data		
...	...	DD1Arena_02_b	Arena_03_b...4	Arena_04_b	Arena_05_b...6	Arena_16_b...7	Arena_20_b...8
1	0.9791666666666663	0.9861111111111116	14.0221320	9.7659830	13.7664790	12.626033	18.2263520
2	0.9861111111111116	0.99305555555555547	9.1758370	4.3102450	10.2212700	9.611129	14.6563413
3	0.99305555555555547	1 day, 0:00:00	4.8209090	9.7078506	7.6673208	8.107424	6.9222576
4	1 day, 0:00:00	1 day, 0:10:00	11.5022230	0.9226741	12.0935670	12.193965	1.2596858
5	1 day, 0:10:00	1 day, 0:20:00	8.3766790	6.3979503	13.1760097	9.164579	10.8860450
6	1 day, 0:20:00	1 day, 0:30:00	14.1049700	2.3265010	13.9072360	14.194804	3.5465843
7	1 day, 0:30:00	1 day, 0:40:00	9.8447500	5.6064155	14.2748240	11.682547	16.5294710
8	1 day, 0:40:00	1 day, 0:50:00	8.9090075	10.1973500	8.6738810	13.287220	9.3227550
9	1 day, 0:50:00	1 day, 1:00:00	16.7529430	13.0413020	9.0150910	9.558353	24.4551470
10	1 day, 1:00:00	1 day, 1:10:00	9.4166410	4.9464900	10.8296140	16.147777	29.5431700
11	1 day, 1:10:00	1 day, 1:20:00	14.4073680	8.8763140	12.4099530	18.551740	31.8578200
12	1 day, 1:20:00	1 day, 1:30:00	9.4505930	8.7618560	12.9649240	8.754372	24.7100110
13	1 day, 1:30:00	1 day, 1:40:00	14.0121850	9.6634110	10.5531530	31.550860	32.1663600
14	1 day, 1:40:00	1 day, 1:50:00	10.1195130	5.3104936	15.3443130	35.252210	28.3783421
15	1 day, 1:50:00	1 day, 2:00:00	10.4936870	6.6770330	12.7538500	25.510480	5.7919669
16	1 day, 2:00:00	1 day, 2:10:00	11.6065630	3.8545873	12.0504380	23.252300	28.7554980
17	1 day, 2:10:00	1 day, 2:20:00	12.4452810	10.7982440	13.8701090	20.383501	18.3458020
18	1 day, 2:20:00	1 day, 2:30:00	10.9675740	57.7505000	7.9242360	17.422255	21.5297540
19	1 day, 2:30:00	1 day, 2:40:00	16.5789340	4.9534841	13.0194170	14.160064	17.2361180
20	1 day, 2:40:00	1 day, 2:50:00	16.8885260	8.7065610	12.6469110	13.500472	33.8079200
21	1 day, 2:50:00	1 day, 3:00:00	15.1532880	12.8735490	14.3743640	20.035020	30.5771400
22	1 day, 3:00:00	1 day, 3:10:00	16.4372250	2.2831677	9.6253220	23.996390	24.4086570
23	1 day, 3:10:00	1 day, 3:20:00	16.6306840	0.6351950	18.6776880	23.810717	18.3331500
24							21.206019

FromNoldusFileToVisualisation_5.Rmd		FromNoldusFileToVisualisation_6.Rmd*		DD_opn4xa	DD_opn4xa_data
Filter					
time	arena	genotype	distance		
1 0	DD1Arena_02_b	65 opn4xa+/+	14.022132		
2 600	DD1Arena_02_b	65 opn4xa+/+	9.175837		
3 1200	DD1Arena_02_b	65 opn4xa+/+	4.820909		
4 1800	DD1Arena_02_b	65 opn4xa+/+	11.502223		
5 2400	DD1Arena_02_b	65 opn4xa+/+	8.376679		
6 3000	DD1Arena_02_b	65 opn4xa+/+	14.10497		
7 3600	DD1Arena_02_b	65 opn4xa+/+	9.84475		
8 4200	DD1Arena_02_b	65 opn4xa+/+	8.9090075		
9 4800	DD1Arena_02_b	65 opn4xa+/+	16.752943		
10 5400	DD1Arena_02_b	65 opn4xa+/+	9.416641		
11 6000	DD1Arena_02_b	65 opn4xa+/+	14.407368		
12 6600	DD1Arena_02_b	65 opn4xa+/+	9.450593		
13 7200	DD1Arena_02_b	65 opn4xa+/+	14.012185		
14 7800	DD1Arena_02_b	65 opn4xa+/+	10.119513		
15 8400	DD1Arena_02_b	65 opn4xa+/+	10.493687		
16 9000	DD1Arena_02_b	65 opn4xa+/+	11.606563		
17 9600	DD1Arena_02_b	65 opn4xa+/+	12.445281		
18 10200	DD1Arena_02_b	65 opn4xa+/+	10.967574		
19 10800	DD1Arena_02_b	65 opn4xa+/+	16.578934		
20 11400	DD1Arena_02_b	65 opn4xa+/+	16.888526		
21 12000	DD1Arena_02_b	65 opn4xa+/+	15.153288		
22 12600	DD1Arena_02_b	65 opn4xa+/+	16.437225		
23 13200	DD1Arena_02_b	65 opn4xa+/+	16.630684		
24 13800	DD1Arena_02_b	65 opn4xa+/+	14.570325		

**7. create the metadata file** : use the `create_meta` function. Specify the name of your transformed data (`DD_opn4xa_data`), the number of wt, the number of mutants, the name of your wt condition and the name of your mutant condition

```
DD_opn4xa_meta<-create_meta(DD_opn4xa_data,65,65,conditionwt="opn4xa+/+",conditionmut="opn4xa-/-")
```

```
## [1] "65 opn4xa+/+"
## [1] "65 opn4xa-/-"
```



IdusFileToVisualisation\_5.Rmd x FromNoldusFileToVisualisation\_6.Rmd x DD\_opn4xa x faire\_plots.R x DD\_opn4xa\_meta x

Filter

	arena	genotype
1	Arena_03_b...109	65 opn4xa-/-
2	Arena_03_b...4	65 opn4xa+/+
3	Arena_04_b	65 opn4xa+/+
4	Arena_05_b...44	65 opn4xa+/+
5	Arena_05_b...6	65 opn4xa+/+
6	Arena_05_b...87	65 opn4xa-/-
7	Arena_06_b...45	65 opn4xa+/+
8	Arena_06_b...88	65 opn4xa-/-
9	Arena_07_b...110	65 opn4xa-/-
10	Arena_07_b...89	65 opn4xa-/-
11	Arena_11_b	65 opn4xa+/+
12	Arena_12_b	65 opn4xa+/+
13	Arena_13_b...111	65 opn4xa-/-
14	Arena_13_b...23	65 opn4xa+/+
15	Arena_14_b	65 opn4xa+/+
16	Arena_15_b...47	65 opn4xa+/+
17	Arena_15_b...90	65 opn4xa-/-
18	Arena_16_b...7	65 opn4xa+/+
19	Arena_16_b...91	65 opn4xa-/-
20	Arena_20_b...25	65 opn4xa+/+
21	Arena_20_b...8	65 opn4xa+/+
22	Arena_21_b	65 opn4xa-/-
23	Arena_23_b	65 opn4xa+/+
24	Arena_24_b	65 opn4xa+/+

**8. create the behavr file** : use the `create_behavr` function. Specify the name of your transformed data and the name of your metadata file.

```
DD_opn4xa_behavr<-create_behavr(DD_opn4xa_data,DD_opn4xa_meta)
```

FromNoldusFileToVisualisation\_5.Rmd x FromNoldusFileToVisualisation\_6.Rmd x faire\_plots.R x DD\_opn4xa\_behavr x

Filter

	time	arena	genotype	distance
1	0	Arena_03_b...109	65 opn4xa-/-	11.568069
2	600	Arena_03_b...109	65 opn4xa-/-	4.9695324
3	1200	Arena_03_b...109	65 opn4xa-/-	9.589375
4	1800	Arena_03_b...109	65 opn4xa-/-	8.663392
5	2400	Arena_03_b...109	65 opn4xa-/-	11.763586
6	3000	Arena_03_b...109	65 opn4xa-/-	15.181103
7	3600	Arena_03_b...109	65 opn4xa-/-	6.057857
8	4200	Arena_03_b...109	65 opn4xa-/-	13.210338
9	4800	Arena_03_b...109	65 opn4xa-/-	8.250312
10	5400	Arena_03_b...109	65 opn4xa-/-	10.624124
11	6000	Arena_03_b...109	65 opn4xa-/-	9.5482987
12	6600	Arena_03_b...109	65 opn4xa-/-	12.719358
13	7200	Arena_03_b...109	65 opn4xa-/-	7.6796266
14	7800	Arena_03_b...109	65 opn4xa-/-	11.172048
15	8400	Arena_03_b...109	65 opn4xa-/-	7.35353
16	9000	Arena_03_b...109	65 opn4xa-/-	19.919891
17	9600	Arena_03_b...109	65 opn4xa-/-	5.8941701
18	10200	Arena_03_b...109	65 opn4xa-/-	13.205857
19	10800	Arena_03_b...109	65 opn4xa-/-	9.791841
20	11400	Arena_03_b...109	65 opn4xa-/-	12.514276
21	12000	Arena_03_b...109	65 opn4xa-/-	32.17368
22	12600	Arena_03_b...109	65 opn4xa-/-	22.7686799
23	13200	Arena_03_b...109	65 opn4xa-/-	12.728156
24	13800	Arena_03_b...109	65 opn4xa-/-	7.502922

Showing 1 to 24 of 75,790 entries

**9. replace missing values with 0 and specify your legend name for wt and mutants**

```
DD_opn4xa_behavr$distance[is.na(DD_opn4xa_behavr$distance)]<-0
wt<-print(paste(65,"opn4xa+/+",sep=" "))
```

```
## [1] "65 opn4xa+/+"
```

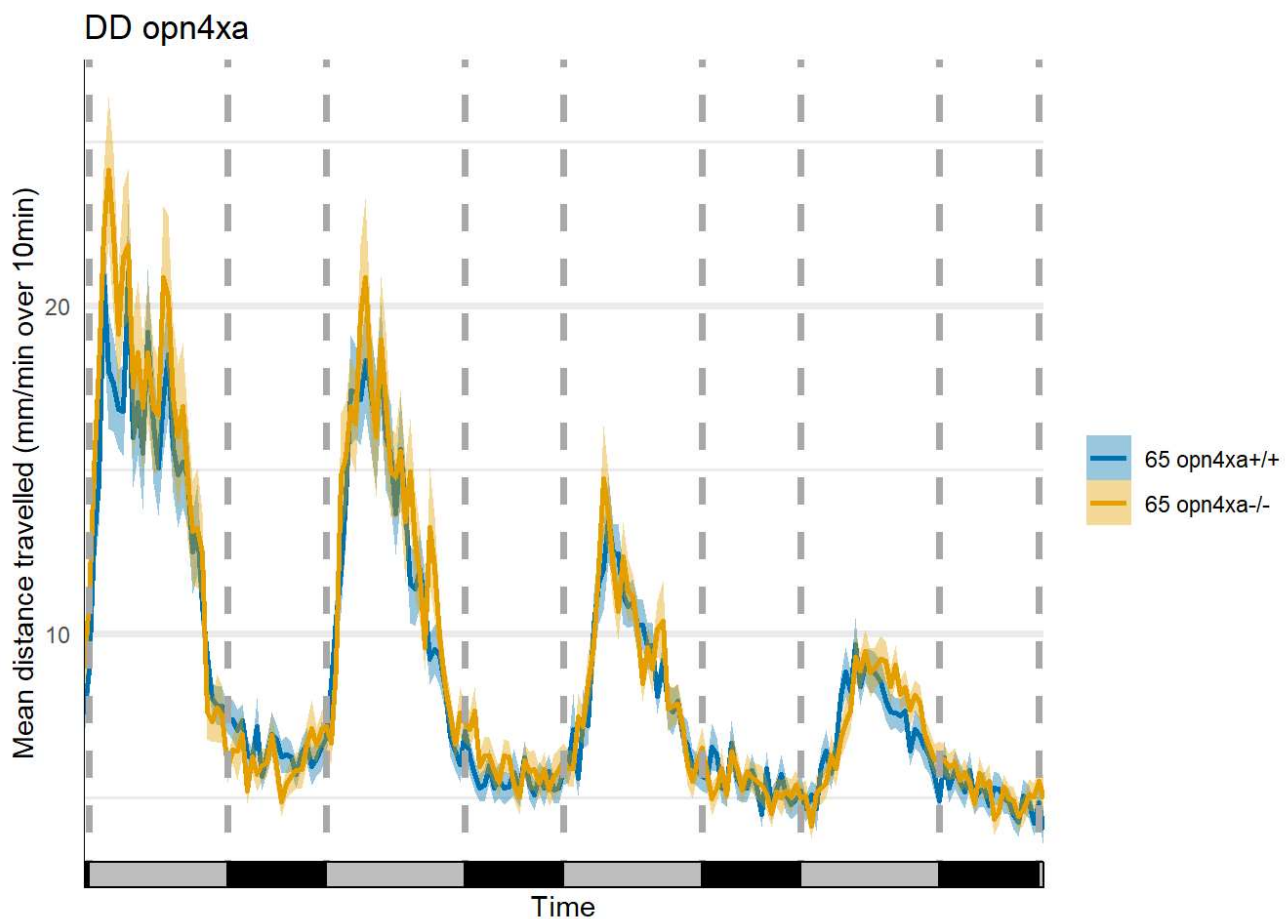
```
mut<-print(paste(65,"opn4xa-/-",sep=" "))
```

```
## [1] "65 opn4xa-/-"
```

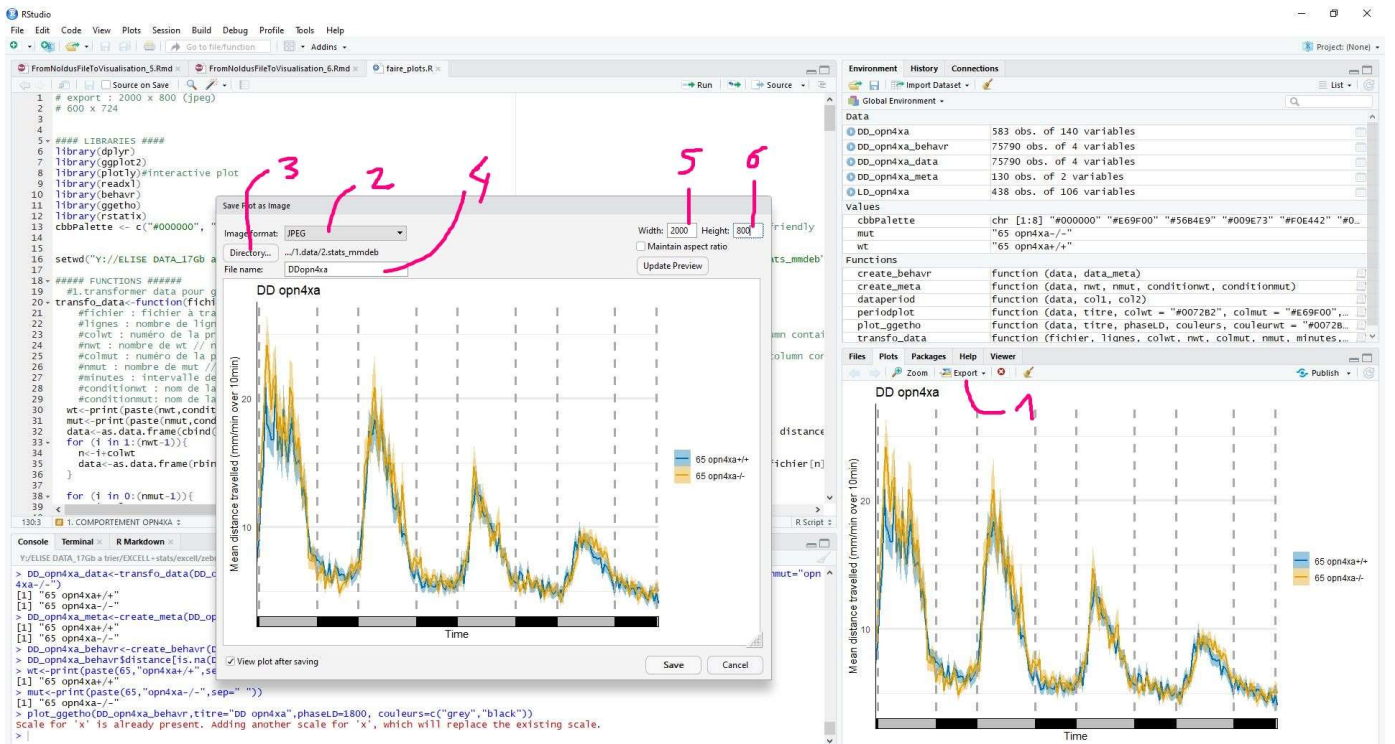
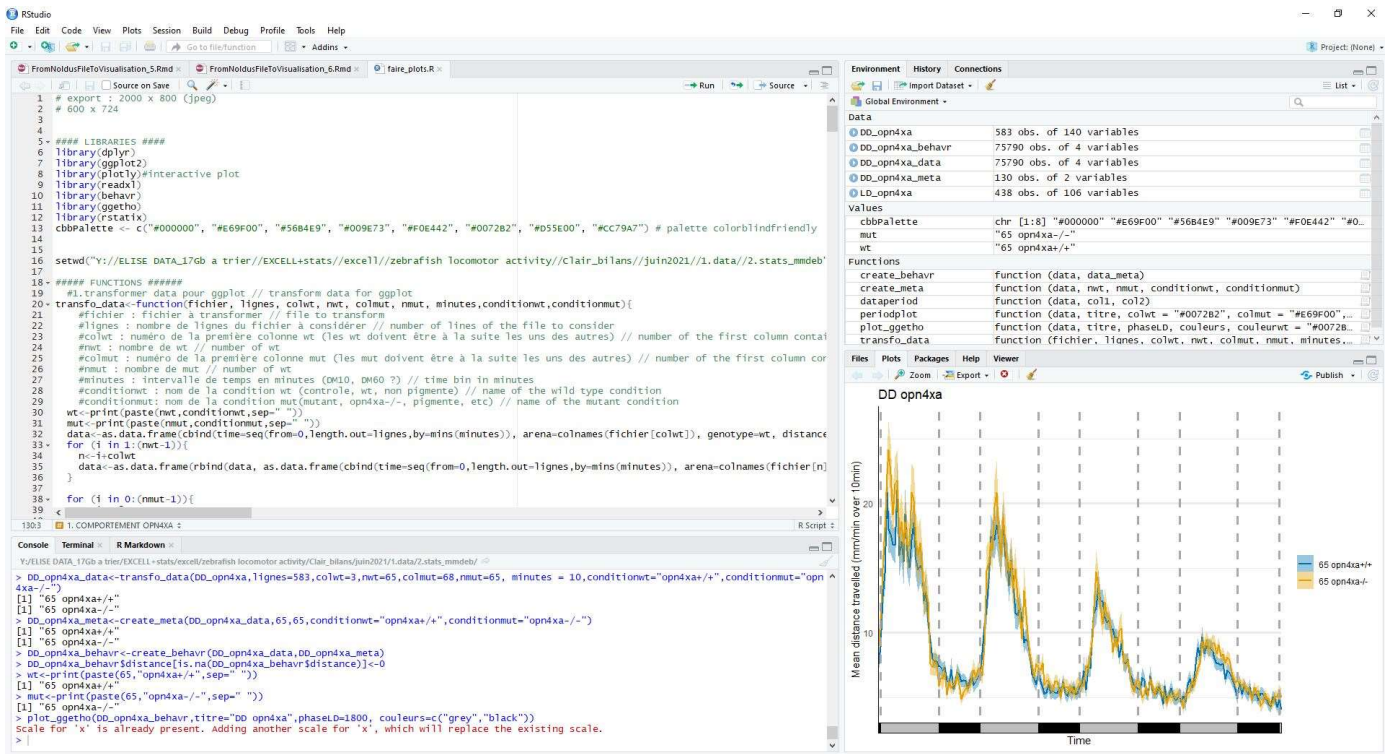
**10. Make your beautiful plot !** : use the `plot_ggetho` function. Specify the name of your behavr file, `titre` is the title of your graph, `phaseLD` is the time in seconds before the first dark to light transition (*here the data starts at 8:30:00 so 1800 seconds before 9:00:00*), `couleurs` is a vector containing the colors you want to use in the horizontal bars. If you want to change the colors used in the graph, add `couleurwt="your color"` for the wt color and `couleurmut="your color"` for the mutant color.

```
plot_ggetho(DD_opn4xa_behavr,titre="DD opn4xa",phaseLD=1800, couleurs=c("grey","black"))
```

```
## Scale for 'x' is already present. Adding another scale for 'x', which will  
## replace the existing scale.
```



**11. Export your beautiful plot**



## 12. Make a beautiful phase shift plot