Scripts by GR for Data Wrangling and Visualization

GR

Tuesday, September 22, 2015

### How to cut the time into TOD

library(plyr)  
library(dplyr)

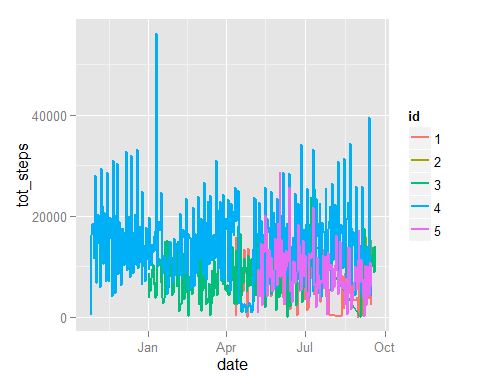
##   
## Attaching package: 'dplyr'  
##   
## The following objects are masked from 'package:plyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize  
##   
## The following objects are masked from 'package:stats':  
##   
## filter, lag  
##   
## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

load("C:/Users/Gabriel/JHUDash/Git Repo/Fitbit\_analysis\_JHUDash/data/all\_data.rda")  
  
TOD <- cut(all\_data$HofD.steps,breaks = c(0,4,10,17,24),labels = c('Too early?','Morning','Mid-day','Evening'),right=T)  
TOD[is.na(TOD)] <- factor('Too early?',levels = levels(TOD))  
all\_data$TOD <- TOD  
all\_data$id <- factor(all\_data$id)  
all\_data <- filter(all\_data, tot\_steps > 0)

## Visualizations

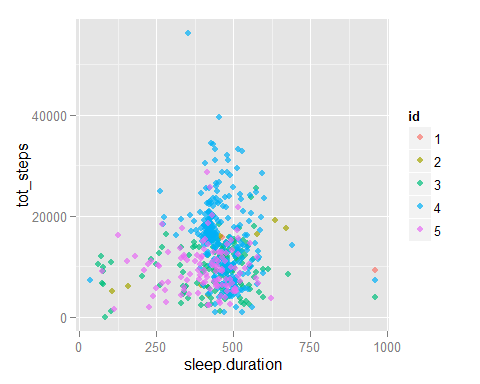
First let's look at total daily steps over time.

library(ggplot2)  
  
analyzed\_data <- all\_data %>% distinct(id,date)  
p <- ggplot(data=analyzed\_data,aes(x=date,y=tot\_steps,color=id))  
p + geom\_line(size=1)

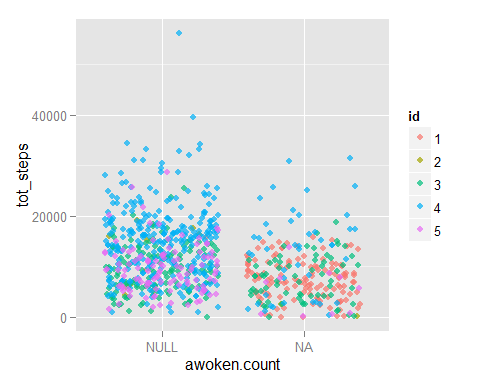


Now let's look at daily steps by different variables.

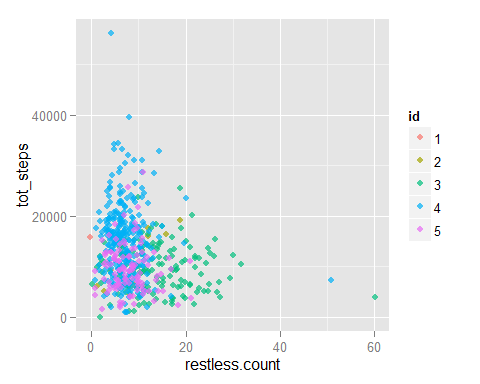
p <- ggplot(data=analyzed\_data,aes(x=sleep.duration,y=tot\_steps,color=id))  
p + geom\_jitter(alpha=0.7)



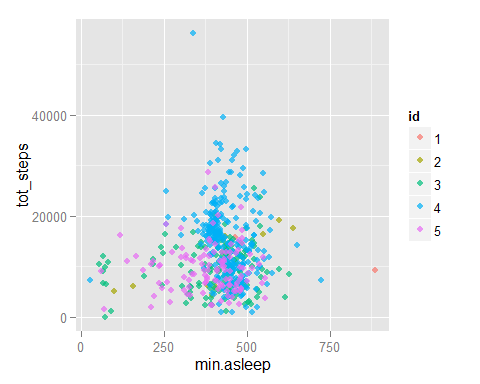
p <- ggplot(data=analyzed\_data,aes(x=awoken.count,y=tot\_steps,color=id))  
p + geom\_jitter(alpha=0.7)



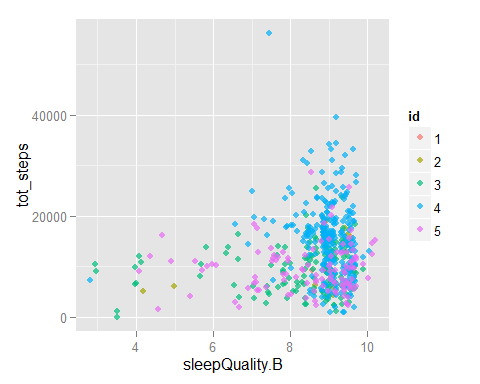
p <- ggplot(data=analyzed\_data,aes(x=restless.count,y=tot\_steps,color=id))  
p + geom\_jitter(alpha=0.7)



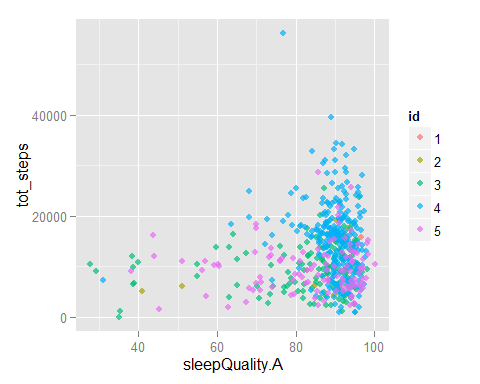
p <- ggplot(data=analyzed\_data,aes(x=min.asleep,y=tot\_steps,color=id))  
p + geom\_jitter(alpha=0.7)



p <- ggplot(data=analyzed\_data,aes(x=sleepQuality.B,y=tot\_steps,color=id))  
p + geom\_jitter(alpha=0.7)



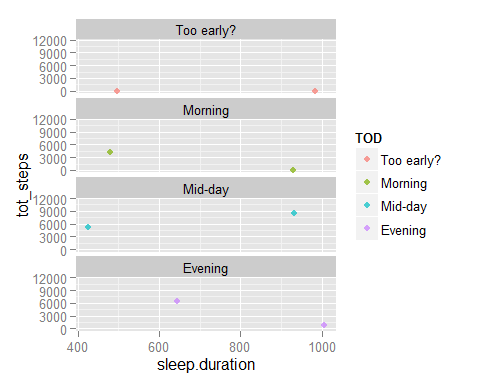
p <- ggplot(data=analyzed\_data,aes(x=sleepQuality.A,y=tot\_steps,color=id))  
p + geom\_jitter(alpha=0.7)



What about Time of Day?

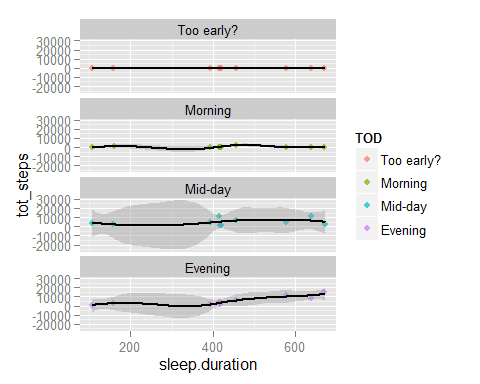
analyzed\_data <- all\_data %>% group\_by(id,date,TOD) %>%  
 summarise(tot\_steps = sum(steps), sleep.duration = min(sleep.duration),  
 sleepQuality.B = min(sleepQuality.B),  
 sleepQuality.A = min(sleepQuality.A))  
  
p <- ggplot(data=filter(analyzed\_data,id==1),aes(x=sleep.duration,y=tot\_steps,color=TOD))  
p + geom\_jitter(alpha=0.7) +facet\_wrap(~TOD,ncol = 1) + stat\_smooth(size=1,color='black',group=1)

## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.



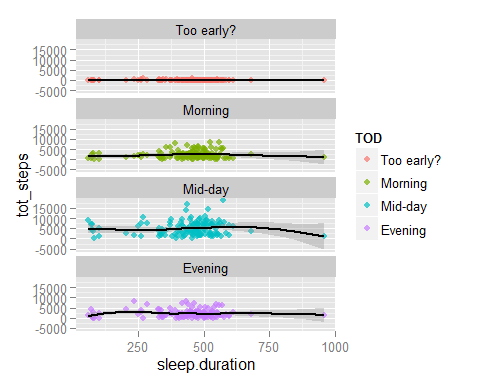
p <- ggplot(data=filter(analyzed\_data,id==2),aes(x=sleep.duration,y=tot\_steps,color=TOD))  
p + geom\_jitter(alpha=0.7) +facet\_wrap(~TOD,ncol = 1) + stat\_smooth(size=1,color='black',group=1)

## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.



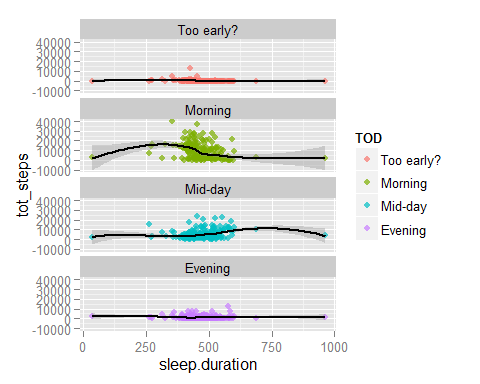
p <- ggplot(data=filter(analyzed\_data,id==3),aes(x=sleep.duration,y=tot\_steps,color=TOD))  
p + geom\_jitter(alpha=0.7) +facet\_wrap(~TOD,ncol = 1) + stat\_smooth(size=1,color='black',group=1)

## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.



p <- ggplot(data=filter(analyzed\_data,id==4),aes(x=sleep.duration,y=tot\_steps,color=TOD))  
p + geom\_jitter(alpha=0.7) +facet\_wrap(~TOD,ncol = 1) + stat\_smooth(size=1,color='black',group=1)

## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.



p <- ggplot(data=filter(analyzed\_data,id==5),aes(x=sleep.duration,y=tot\_steps,color=TOD))  
p + geom\_jitter(alpha=0.7) +facet\_wrap(~TOD,ncol = 1) + stat\_smooth(size=1,color='black',group=1)

## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.  
## geom\_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to change the smoothing method.

