Per&Act-Exam

JR

9 6 2020

## Data loading

## loading txt data  
data1 <- read\_delim(file = "105\_1.txt", delim = ",", col\_names = FALSE)

## Parsed with column specification:  
## cols(  
## X1 = col\_double(),  
## X2 = col\_double(),  
## X3 = col\_double()  
## )

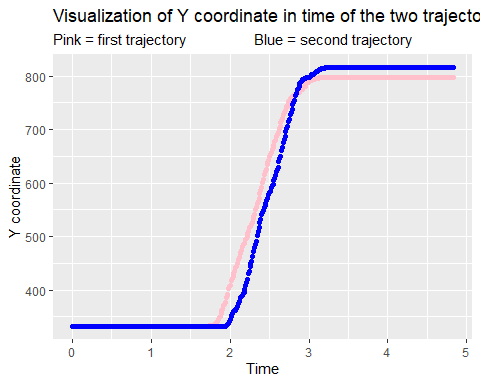
data2 <- read\_delim(file = "105\_2.txt", delim = ",", col\_names = FALSE)

## Parsed with column specification:  
## cols(  
## X1 = col\_double(),  
## X2 = col\_double(),  
## X3 = col\_double()  
## )

## renaming columns  
data1 <- data1 %>%   
 rename(time = X1, y = X2, z = X3)  
  
data2 <- data2 %>%   
 rename(time = X1, y = X2, z = X3)

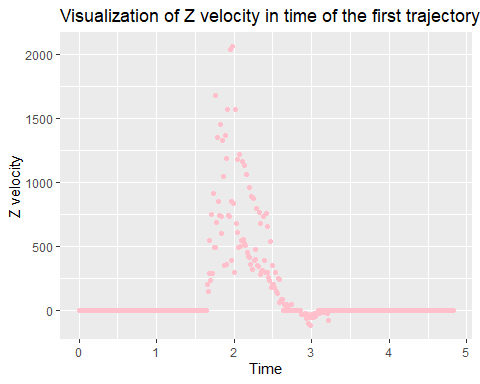
## Data plotting

# as the data frames are of different lengths, the cannot be combined and must be plotted in diffrent geoms  
ggplot(data1, aes(time, y)) +   
 geom\_point(color = "pink") +  
 geom\_point(data = data2, aes(time,y), color = "blue") +  
 labs(title = "Visualization of Y coordinate in time of the two trajectories",  
 subtitle = "Pink = first trajectory Blue = second trajectory",  
 x = "Time",  
 y = "Y coordinate")

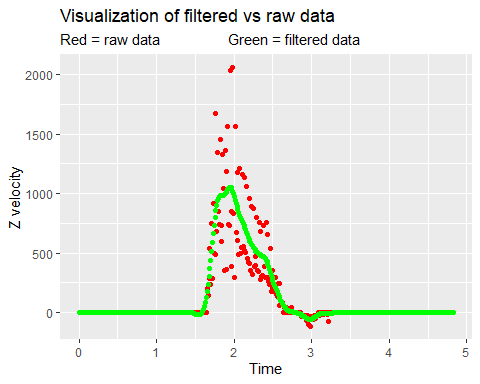


## Velocity, filter

## Z velocity  
  
data1$z\_vel <- c(0, diff(data1$z) / diff(data1$time))  
  
# the velocity for every data point is calculated with distance from the previous to current Y cordinate, divided by the time taken to execute the movement  
  
# because is the difference between two data points, the first value will alaways miss a value to be compared to, therefore we manually insert a 0 to account for this and have a column of the same length  
  
## plotting Z velocity against time  
ggplot(data1, aes(time, z\_vel)) +   
 geom\_point(color = "pink") +  
 labs(title = "Visualization of Z velocity in time of the first trajectory",  
 x = "Time",  
 y = "Z velocity")



## applying butterworth filter  
  
# making cutoffs  
filter\_cutoff <- .1 # setting the cutoff at 1/10-th of Nyquist frequency  
filter\_order <- 2  
bw\_f <- butter(filter\_order, filter\_cutoff, type ='low')   
  
  
# filter application  
data1$z\_vel\_f <- filtfilt(bw\_f, data1$z\_vel)  
   
  
## plotting filtered data vs raw data  
ggplot(data1, aes(time, z\_vel)) +   
 geom\_point(color = "red") +  
 geom\_point(data = data1, aes(time,z\_vel\_f), color = "green") +  
 labs(title = "Visualization of filtered vs raw data",  
 subtitle = "Red = raw data Green = filtered data",  
 x = "Time",  
 y = "Z velocity")



# we can see that the actual movement occurs only from aproixmately 1.6s to 3.4s, at least on te Z coordinate