

VICTR Lab - Stereotype Threat Graph

Joshua Wei

2024-10-04

```
library(dplyr)

## 
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
## 
##     filter, lag

## The following objects are masked from 'package:base':
## 
##     intersect, setdiff, setequal, union

library(ggplot2)

#Women's Ankle Monitor(b) Data

AP378 = read.csv("/Users/joshuawei/Downloads/VICTR Lab/Stereotype Threat Analysis/AP378 Test Data/AP378.csv")

secs_df = read.csv("/Users/joshuawei/Downloads/VICTR Lab/Stereotype Threat Analysis/AP378 Test Data/AP378.csv")
secs_df = na.omit(secs_df)
secs_df = secs_df %>%
  filter(Observed.Sec > Real.Sec) %>%
  select("ID", "Real.Sec")

AP378$ID = as.numeric(gsub("[^0-9]", "", AP378$ID)) #change the ID to a regular number

main_df = AP378 %>%
  left_join(secs_df, by = "ID") %>% # Join to get the Real.Sec column
  group_by(ID) %>%
  filter(row_number() <= Real.Sec) %>% # Filter rows based on Real.Sec
  ungroup() %>%
  select(-Real.Sec)

main_df

## # A tibble: 139,001 x 5
##       Seconds Y.axis X.axis Z.axis     ID
##      <dbl>   <dbl> <dbl> <dbl> <dbl>
```

```

##      <int> <int> <int> <int> <dbl>
## 1     0     0     0     0     1
## 2     1     0     0     0     1
## 3     2     0     0     0     1
## 4     3     0     0     0     1
## 5     4     0     0     0     1
## 6     5     0     7     0     1
## 7     6     0     0     0     1
## 8     7     0     0     0     1
## 9     8     0     0     0     1
## 10    9     0     0     0     1
## # i 138,991 more rows

write.csv(main_df, file = "main.csv")

#Gives me a data with the right amount of seconds

masterfile = read.csv("/Users/joshuawei/Downloads/VICTR Lab/Stereotype Threat Analysis/AP378 Test Data/")

masterfile <- masterfile %>%
  select("ID...1", "Cond") %>%
  rename(ID = "ID...1")

everything_df = left_join(main_df, masterfile, by= "ID")

#Adds the Conditions into the dataset per ID

everything_df %>%
  filter(Seconds >= 1440) %>%
  group_by(ID,Cond) %>%
  summarise(n = n())

## `summarise()` has grouped output by 'ID'. You can override using the `.groups` argument.

## # A tibble: 4 x 3
## # Groups:   ID [4]
##       ID Cond     n
##   <dbl> <chr> <int>
## 1     72 Threat    68
## 2     95 Control   60
## 3    151 Threat   204
## 4    159 Control   61

#4

everything_df %>%
  filter(Seconds >= 1260) %>%
  group_by(ID,Cond) %>%
  summarise(n = n())

## `summarise()` has grouped output by 'ID'. You can override using the `.groups` argument.

```

```

## # A tibble: 29 x 3
## # Groups:   ID [29]
##       ID Cond     n
##   <dbl> <chr> <int>
## 1     7 Lift     97
## 2    14 Control    2
## 3    19 Lift      2
## 4    22 Lift     61
## 5    24 Threat    53
## 6    29 Control   35
## 7    38 Lift      4
## 8    40 Threat     7
## 9    48 Threat     3
## 10   62 Control    4
## # i 19 more rows

#29

#X axis

bruce_levels = c(540, 720, 900, 1080, 1260, 1440, 1620, 1800)

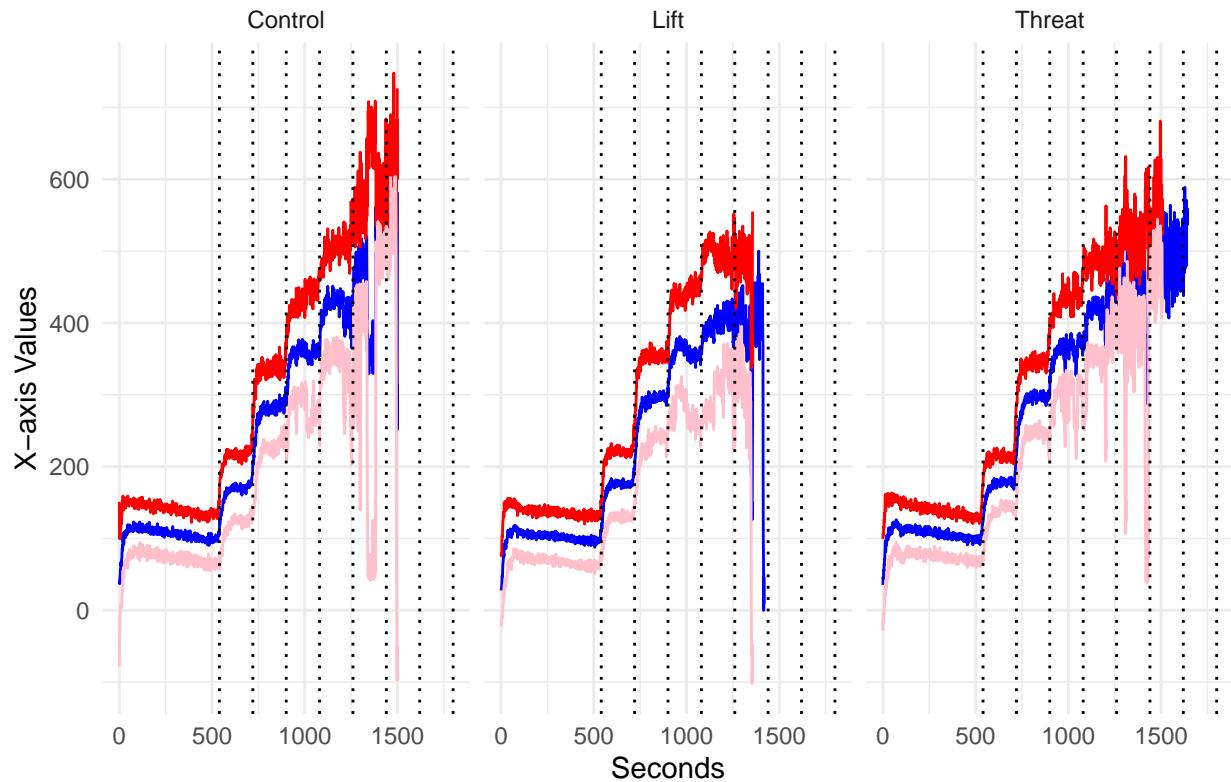
summary_df = everything_df %>%
  group_by(Seconds, Cond) %>%
  summarise(mean = mean(X.axis),
            sd = sd(X.axis),
            .groups = "drop")

ggplot(summary_df) +
  geom_line(aes(x = Seconds, y = mean), color = "blue") +
  geom_line(aes(x = Seconds, y = mean + sd), color = "red") + #one sd above the mean
  geom_line(aes(x = Seconds, y = mean - sd), color = "pink") + #one sd below the mean
  labs(title = "Mean and Standard Deviation over Time",
       x = "Seconds",
       y = "X-axis Values") +
  theme_minimal() +
  facet_wrap(~Cond) +
  geom_vline(xintercept = bruce_levels, color = "black", linetype = "dotted")

## Warning: Removed 136 rows containing missing values ('geom_line()').
## Removed 136 rows containing missing values ('geom_line()').

```

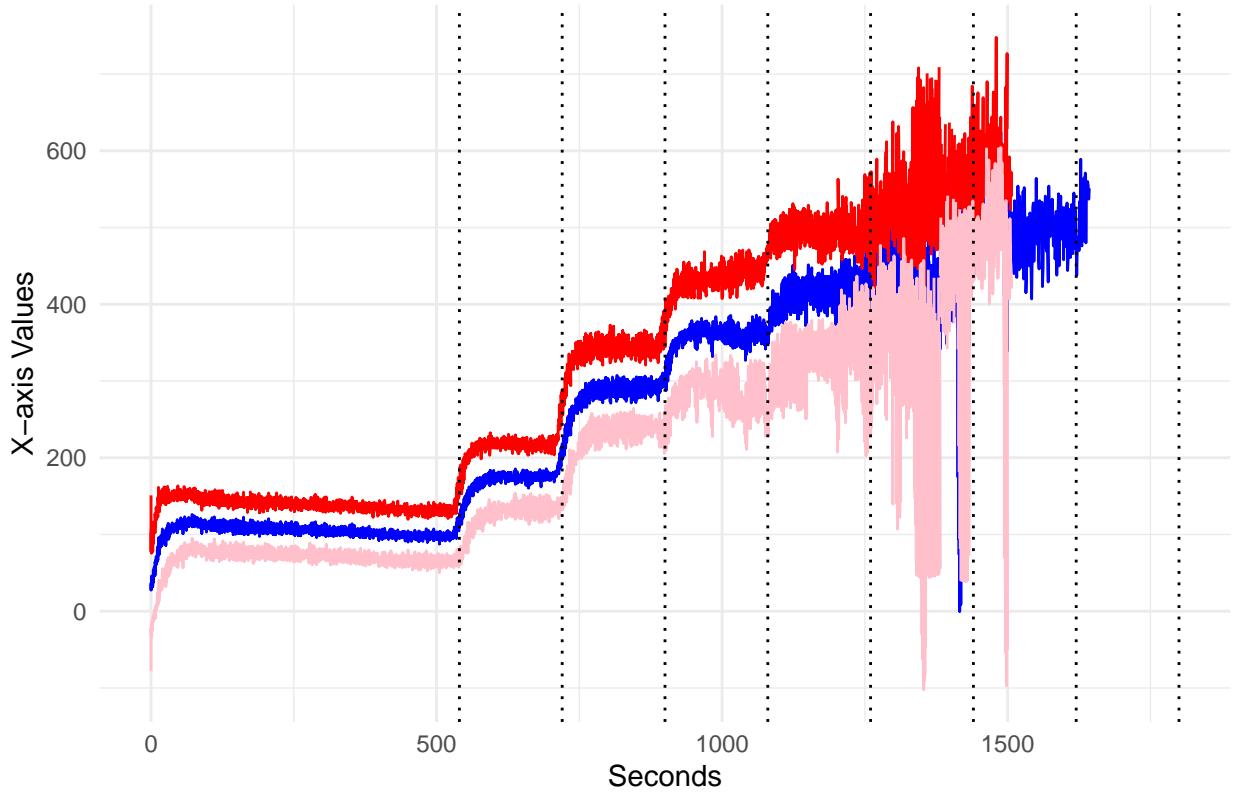
Mean and Standard Deviation over Time



```
ggplot(summary_df) +
  geom_line(aes(x = Seconds, y = mean), color = "blue") +
  geom_line(aes(x = Seconds, y = mean + sd), color = "red") + #one sd above the mean
  geom_line(aes(x = Seconds, y = mean - sd), color = "pink") + #one sd below the mean
  labs(title = "Mean and Standard Deviation over Time",
       x = "Seconds",
       y = "X-axis Values") +
  theme_minimal() +
  geom_vline(xintercept = bruce_levels, color = "black", linetype = "dotted")
```

```
## Warning: Removed 136 rows containing missing values ('geom_line()').
## Removed 136 rows containing missing values ('geom_line()').
```

Mean and Standard Deviation over Time



```
#Y axis

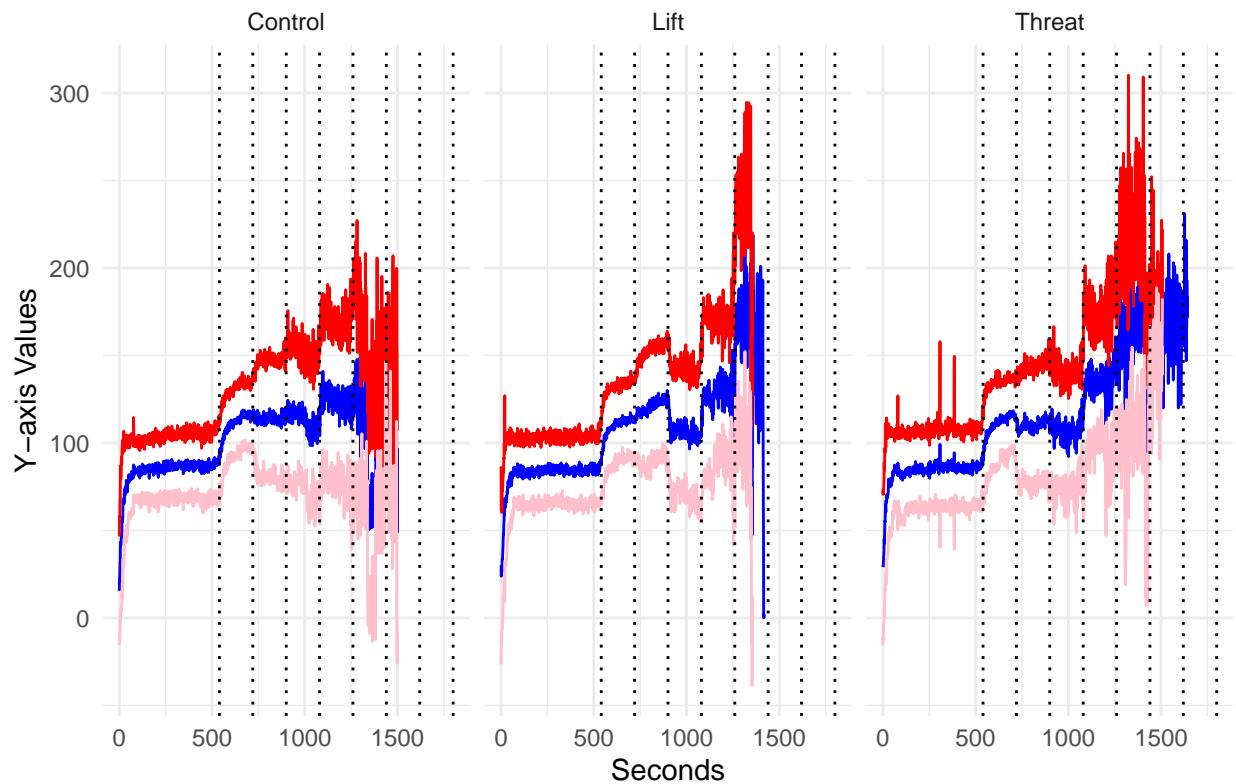
bruce_levels = c(540, 720, 900, 1080, 1260, 1440, 1620, 1800)

summary_df = everything_df %>%
  group_by(Seconds, Cond) %>%
  summarise(mean = mean(Y.axis),
            sd = sd(Y.axis),
            .groups = "drop")

ggplot(summary_df) +
  geom_line(aes(x = Seconds, y = mean), color = "blue") +
  geom_line(aes(x = Seconds, y = mean + sd), color = "red") + #one sd above the mean
  geom_line(aes(x = Seconds, y = mean - sd), color = "pink") + #one sd below the mean
  labs(title = "Mean and Standard Deviation over Time",
       x = "Seconds",
       y = "Y-axis Values") +
  theme_minimal() +
  facet_wrap(~Cond) +
  geom_vline(xintercept = bruce_levels, color = "black", linetype = "dotted")

## Warning: Removed 136 rows containing missing values ('geom_line()').
## Removed 136 rows containing missing values ('geom_line()').
```

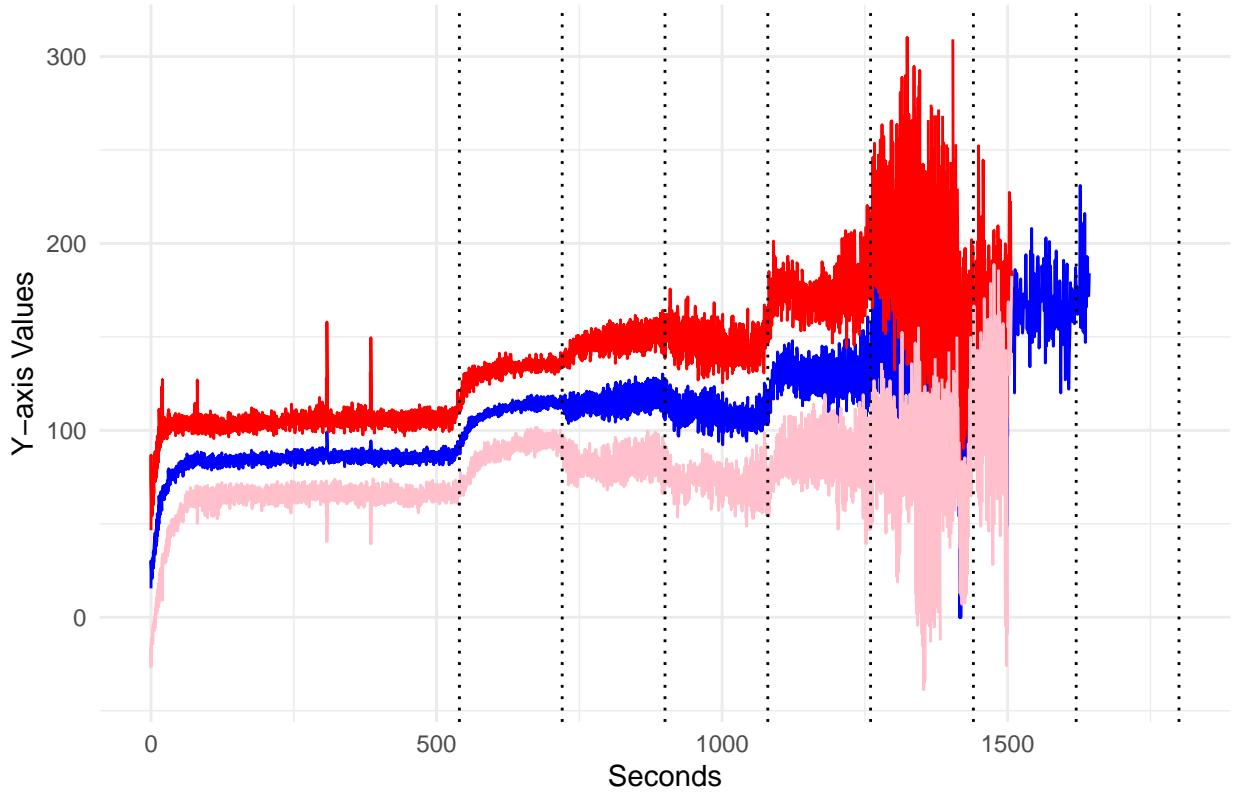
Mean and Standard Deviation over Time



```
ggplot(summary_df) +  
  geom_line(aes(x = Seconds, y = mean), color = "blue") +  
  geom_line(aes(x = Seconds, y = mean + sd), color = "red") + #one sd above the mean  
  geom_line(aes(x = Seconds, y = mean - sd), color = "pink") + #one sd below the mean  
  labs(title = "Mean and Standard Deviation over Time",  
       x = "Seconds",  
       y = "Y-axis Values") +  
  theme_minimal() +  
  geom_vline(xintercept = bruce_levels, color = "black", linetype = "dotted")
```

```
## Warning: Removed 136 rows containing missing values ('geom_line()').  
## Removed 136 rows containing missing values ('geom_line()').
```

Mean and Standard Deviation over Time



```
#Z axis

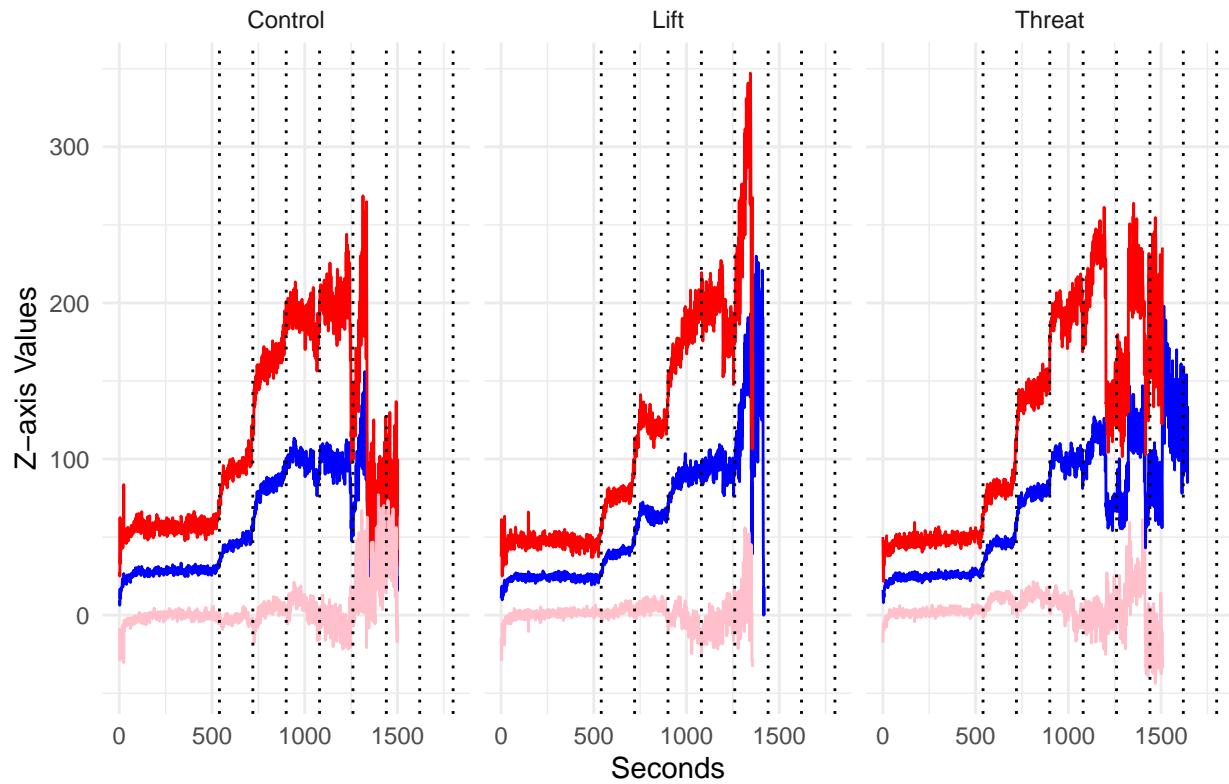
bruce_levels = c(540, 720, 900, 1080, 1260, 1440, 1620, 1800)

summary_df = everything_df %>%
  group_by(Seconds, Cond) %>%
  summarise(mean = mean(Z.axis),
            sd = sd(Z.axis),
            .groups = "drop")

ggplot(summary_df) +
  geom_line(aes(x = Seconds, y = mean), color = "blue") +
  geom_line(aes(x = Seconds, y = mean + sd), color = "red") + #one sd above the mean
  geom_line(aes(x = Seconds, y = mean - sd), color = "pink") + #one sd below the mean
  labs(title = "Mean and Standard Deviation over Time",
       x = "Seconds",
       y = "Z-axis Values") +
  theme_minimal() +
  facet_wrap(~Cond) +
  geom_vline(xintercept = bruce_levels, color = "black", linetype = "dotted")

## Warning: Removed 136 rows containing missing values ('geom_line()').
## Removed 136 rows containing missing values ('geom_line()').
```

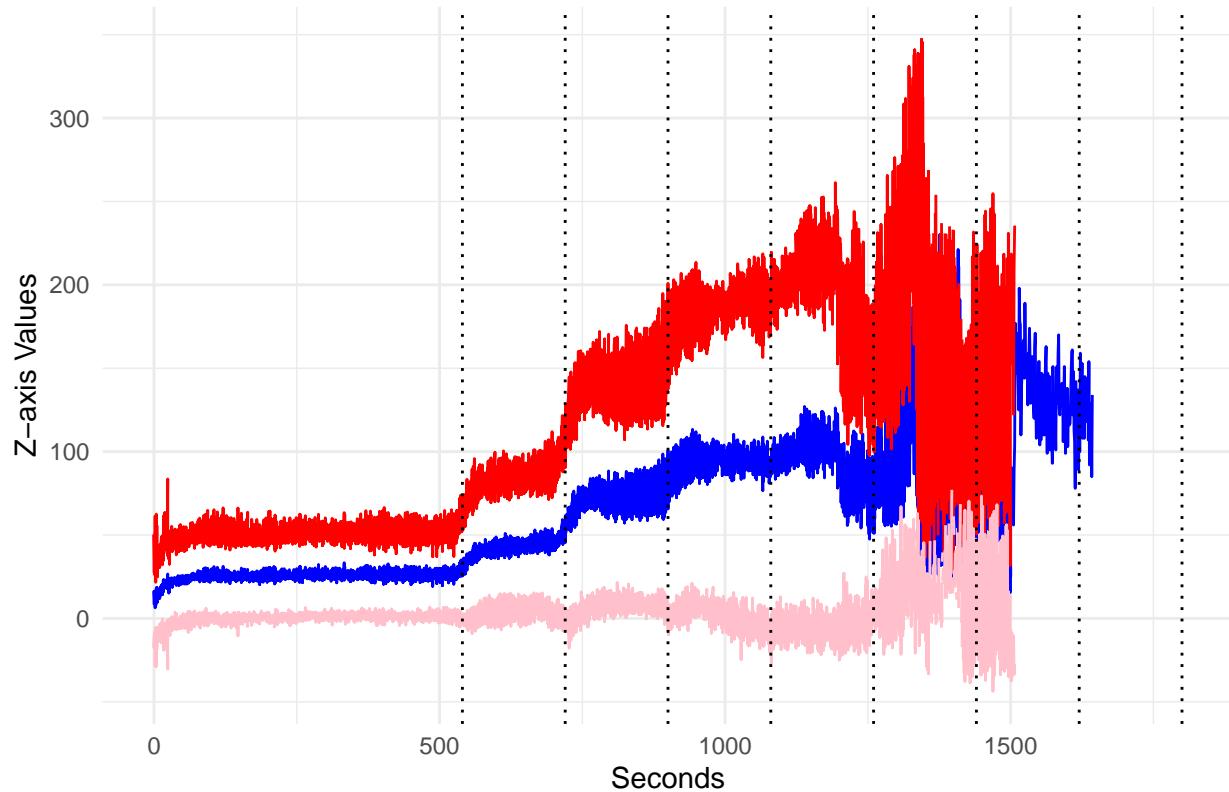
Mean and Standard Deviation over Time



```
ggplot(summary_df) +
  geom_line(aes(x = Seconds, y = mean), color = "blue") +
  geom_line(aes(x = Seconds, y = mean + sd), color = "red") + #one sd above the mean
  geom_line(aes(x = Seconds, y = mean - sd), color = "pink") + #one sd below the mean
  labs(title = "Mean and Standard Deviation over Time",
       x = "Seconds",
       y = "Z-axis Values") +
  theme_minimal() +
  geom_vline(xintercept = bruce_levels, color = "black", linetype = "dotted")
```

```
## Warning: Removed 136 rows containing missing values ('geom_line()').
## Removed 136 rows containing missing values ('geom_line()').
```

Mean and Standard Deviation over Time

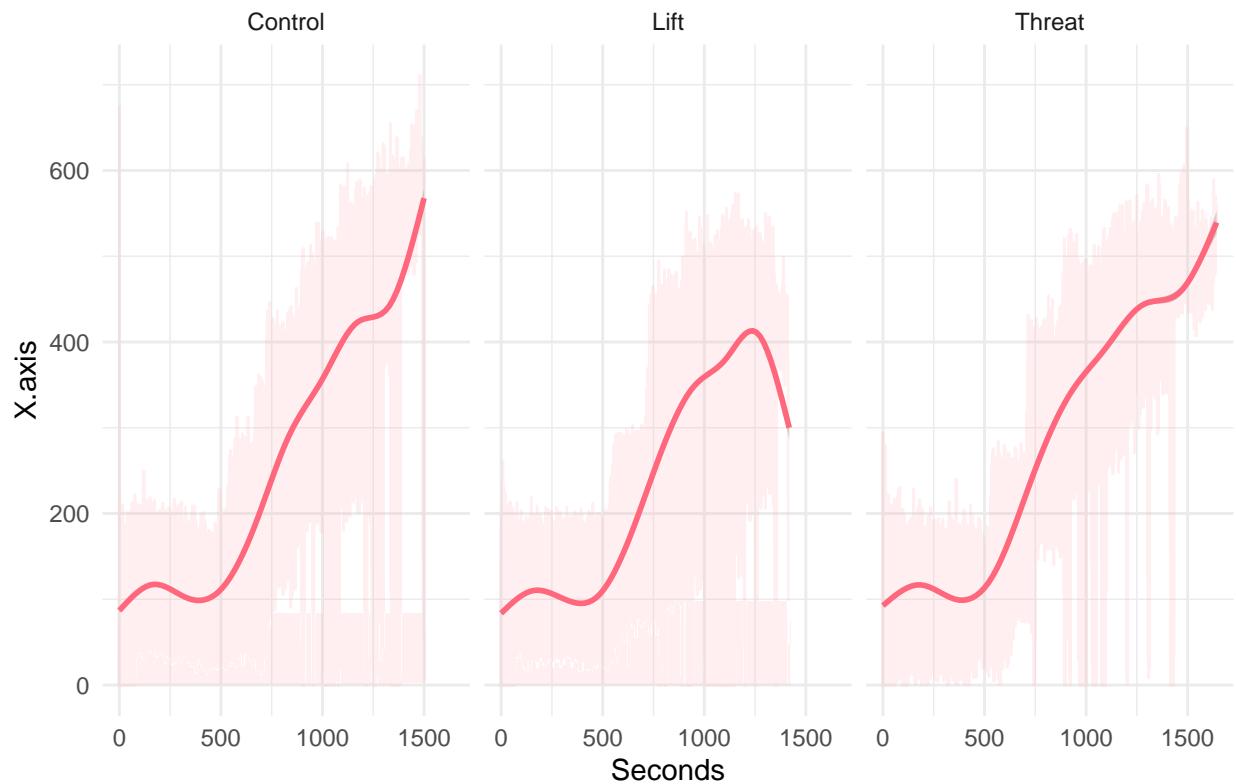


```
#X Graph

everything_df %>%
  ggplot(aes(x = Seconds, y = X.axis)) +
  geom_line(position = position_jitter(width = 0.2), alpha = 0.2, color = "#FFB3BA") +
  geom_smooth(se = TRUE, linetype = "solid", color = "#FF677D") +
  facet_wrap(~Cond) +
  labs(title = "Scatter Plot of X.axis by Seconds",
       x = "Seconds",
       y = "X.axis") +
  theme_minimal()

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```

Scatter Plot of X.axis by Seconds

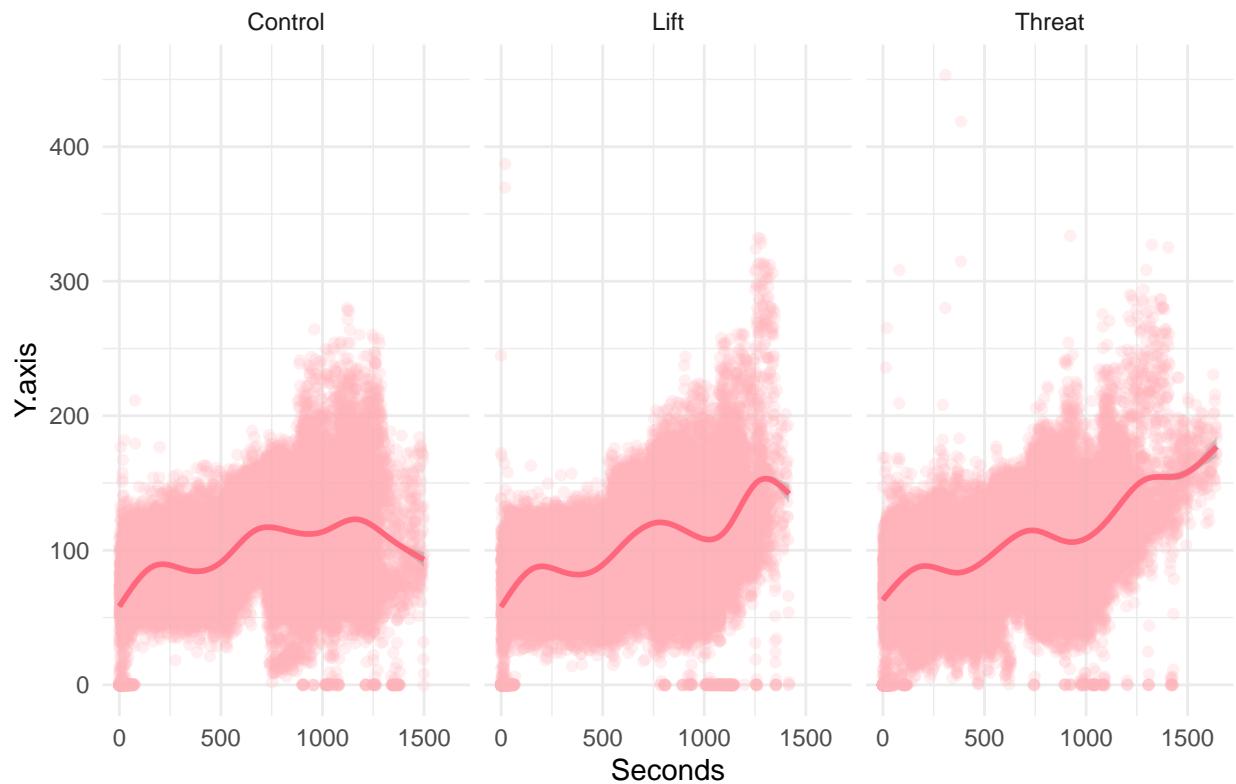


```
#Y Graph
```

```
everything_df %>%
  ggplot(aes(x = Seconds, y = Y.axis)) +
  geom_point(position = position_jitter(width = 0.2), alpha = 0.2, color = "#FFB3BA") +
  geom_smooth(se = TRUE, linetype = "solid", color = "#FF677D") +
  facet_wrap(~Cond) +
  labs(title = "Scatter Plot of Y.axis by Seconds",
       x = "Seconds",
       y = "Y.axis") +
  theme_minimal()
```

```
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```

Scatter Plot of Y.axis by Seconds

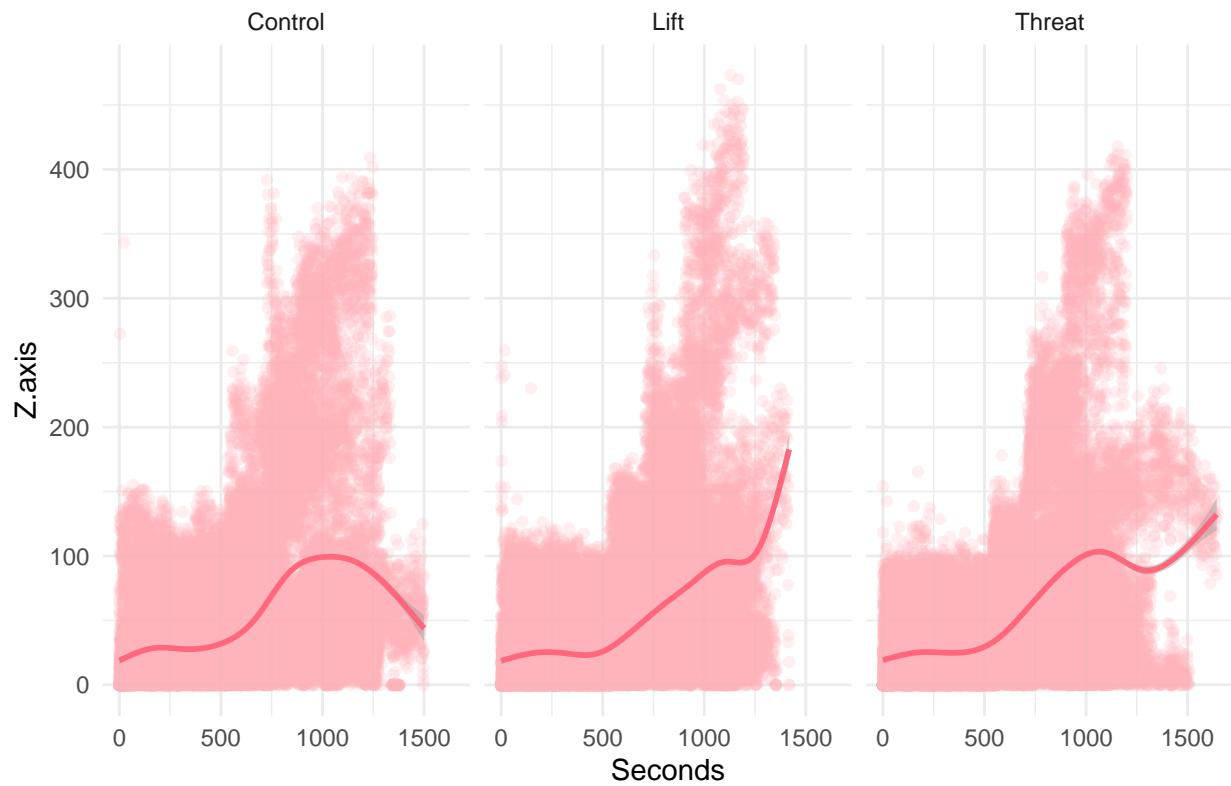


```
#Z Graph

everything_df %>%
  ggplot(aes(x = Seconds, y = Z.axis)) +
  geom_point(position = position_jitter(width = 0.2), alpha = 0.2, color = "#FFB3BA") +
  geom_smooth(se = TRUE, linetype = "solid", color = "#FF677D") +
  facet_wrap(~Cond) +
  labs(title = "Scatter Plot of Z.axis by Seconds",
       x = "Seconds",
       y = "Z.axis") +
  theme_minimal()

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```

Scatter Plot of Z.axis by Seconds



```
everything_df %>% group_by(Cond, ID) %>% ggplot(everything_df, mapping = aes(x = Seconds, y = X.axis, color = Cond)) + geom_point(position = position_jitter(width = 0.2), alpha = 0.5) + labs(title = "Scatter Plot of X.axis by Seconds", x = "Seconds", y = "X.axis") + theme_minimal()
```