Journal of Applied Biomechanics Intramuscular EMG Onset Algorithms

Matthew S. Tenan

January 12, 2017

This file will show to how to extract the data and run some of the analysis from our manuscript

We encourage authors to use our data with their own algorithms!

This helps advance the field more rapidly. You can access the data using the code below. The data is in R's list format and has already been bandpass filtered 10-1000 Hz per the details in our manuscript. Next, we'll do a simple plot of one of the trials.

This code reads all of our intramuscular EMG data into your local working directory:

URL1 <- "http://github.com/TenanATC/EMG/blob/master/fwEMG_JAB.rds?raw=true"

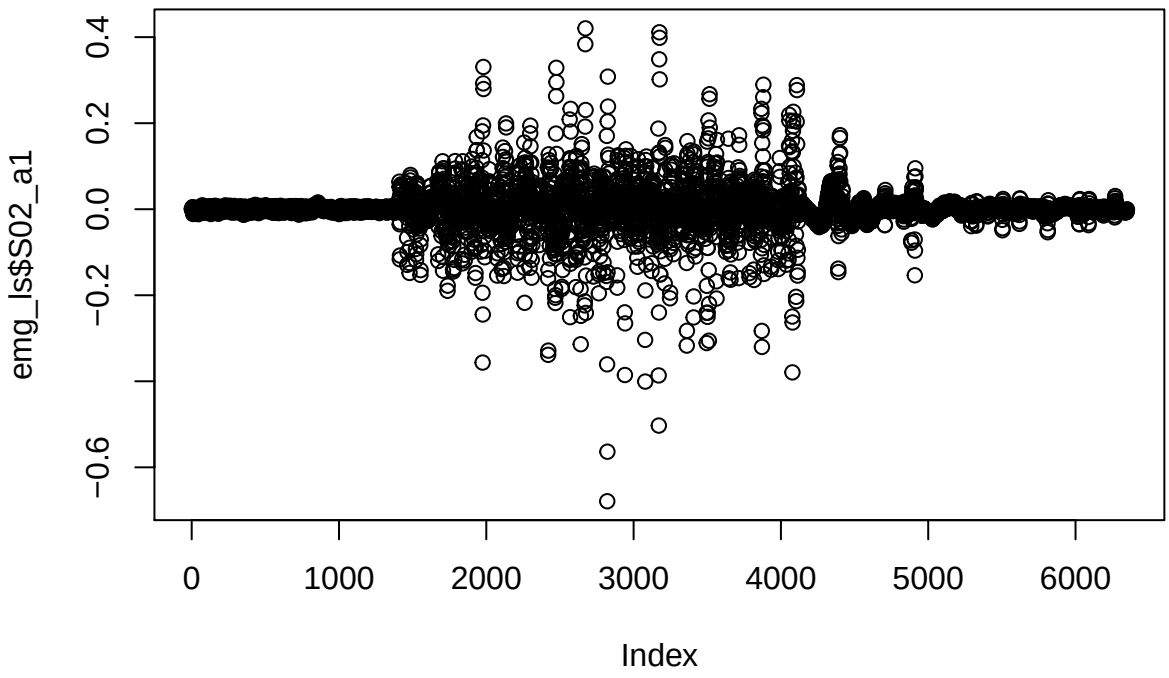
download.file(URL1, destfile = "~/fwEMG_JAB.rds", mode = 'wb', quiet = T)

URL2 <- "http://raw.githubusercontent.com/TenanATC/EMG/master/VisualOnset_JAB.csv"

download.file(URL2, destfile = "~/VisualOnset_JAB.csv", quiet = T)

Now, we'll open it in the workspace do a simple plot of one of the trials.

```
emg_ls <- readRDS('fwEMG_JAB.rds')
vis <- read.csv('VisualOnset_JAB.csv')
plot(emg_ls$S02_a1)
```



Note that all data, including visual onset timing, is in sample number. It is not yet in seconds, but this is an easy conversion since we know the sampling rate was 2048 Hz.

Next, we'll load the packages we'll use for the rest of these examples. This is done using the library() function. If you have never installed these packages before, you may need to use this line of code prior to the library functions: install.packages(c("ggplot2", "dplyr", "bcp"))

```
library(ggplot2)
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(bcp)
```

```
## Loading required package: grid
```

Now on to the fun stuff

Now we'll do an example of the Bayesian Changepoint analysis algorithm which performed well in our systematic analysis (rectified EMG, p0 =0 and posterior probability threshold onset at 95%) and show the differences in onset time.

```
#extract a trial and rectify EMG
fwEMGrect <- abs(emg_ls$S03_l1)
#run BCP algorithm and extract the first point where posterior probability of onset is 95%
result_bcp <- bcp(fwEMGrect, p0 = 0)
onset_bcp <- which(result_bcp$posterior.prob >= 0.95)[1]
#Now we extract the visually determined EMG onset
onset_visual <- vis$value[match('S03_l1', vis$sbj)]

print(onset_bcp)
```

```
## [1] 2901
```

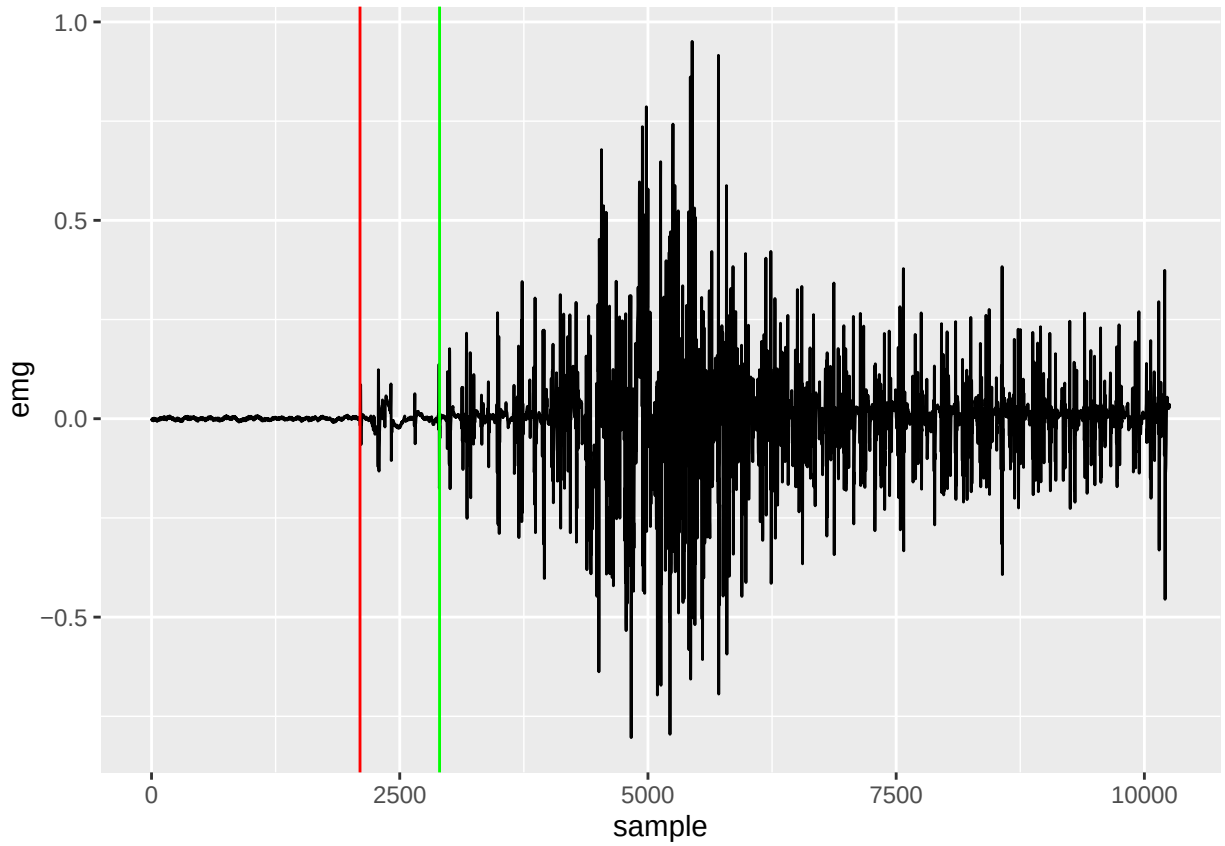
```
print(onset_visual)
```

```
## [1] 2099.833
```

So the difference between the two measures for this trial is 801.167, corresponding to 0.39 seconds.

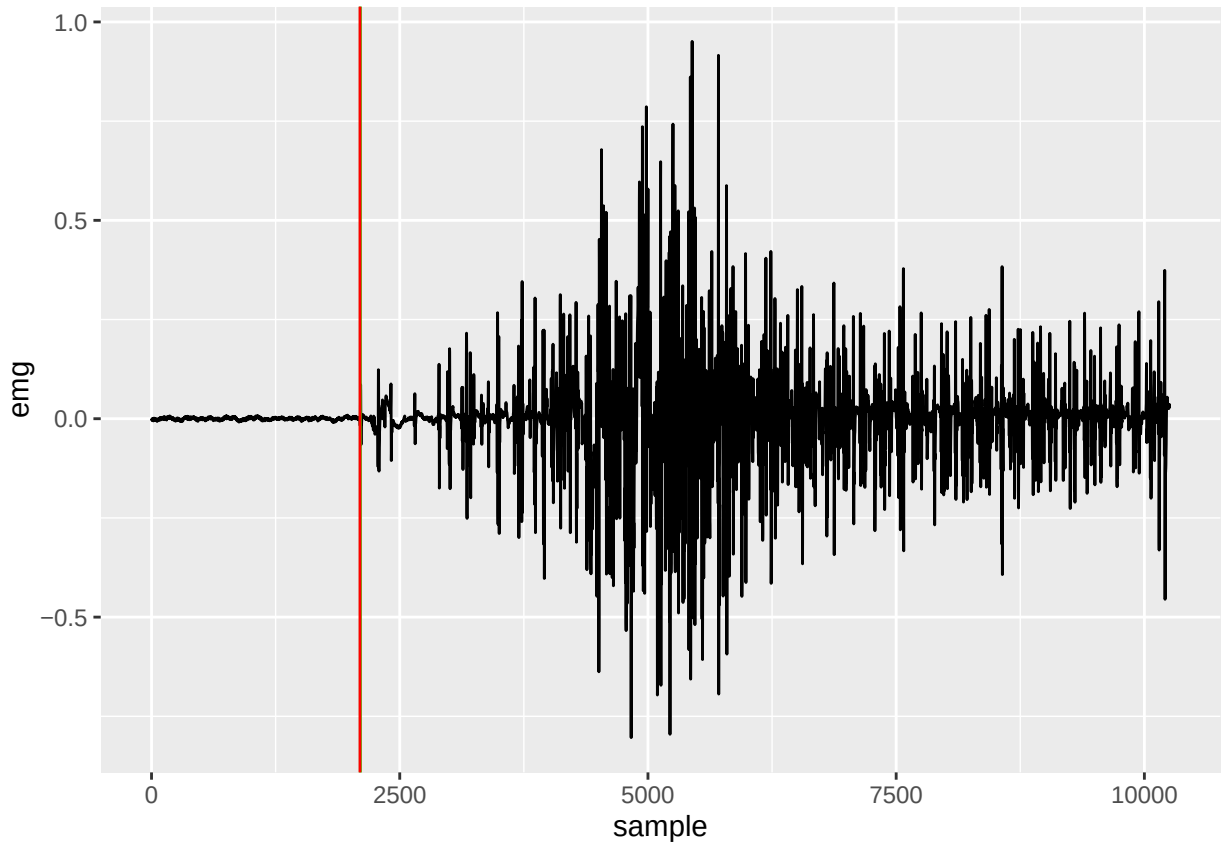
Let's plot this data in such a way that we can do it for multiple trials.

```
#create generic EMG time series variable
emg_plot <- data.frame(emg_ls$S03_l1)
colnames(emg_plot) <- 'emg'
sample <- seq.int(from =1, to = length(emg_plot$emg))
emg_plot <- cbind(emg_plot, sample)

ggplot(emg_plot, aes(x=sample, y=emg)) +
  geom_line() +
  geom_vline(xintercept = onset_bcp, color = 'green') +
  geom_vline(xintercept = onset_visual, color = 'red')
```



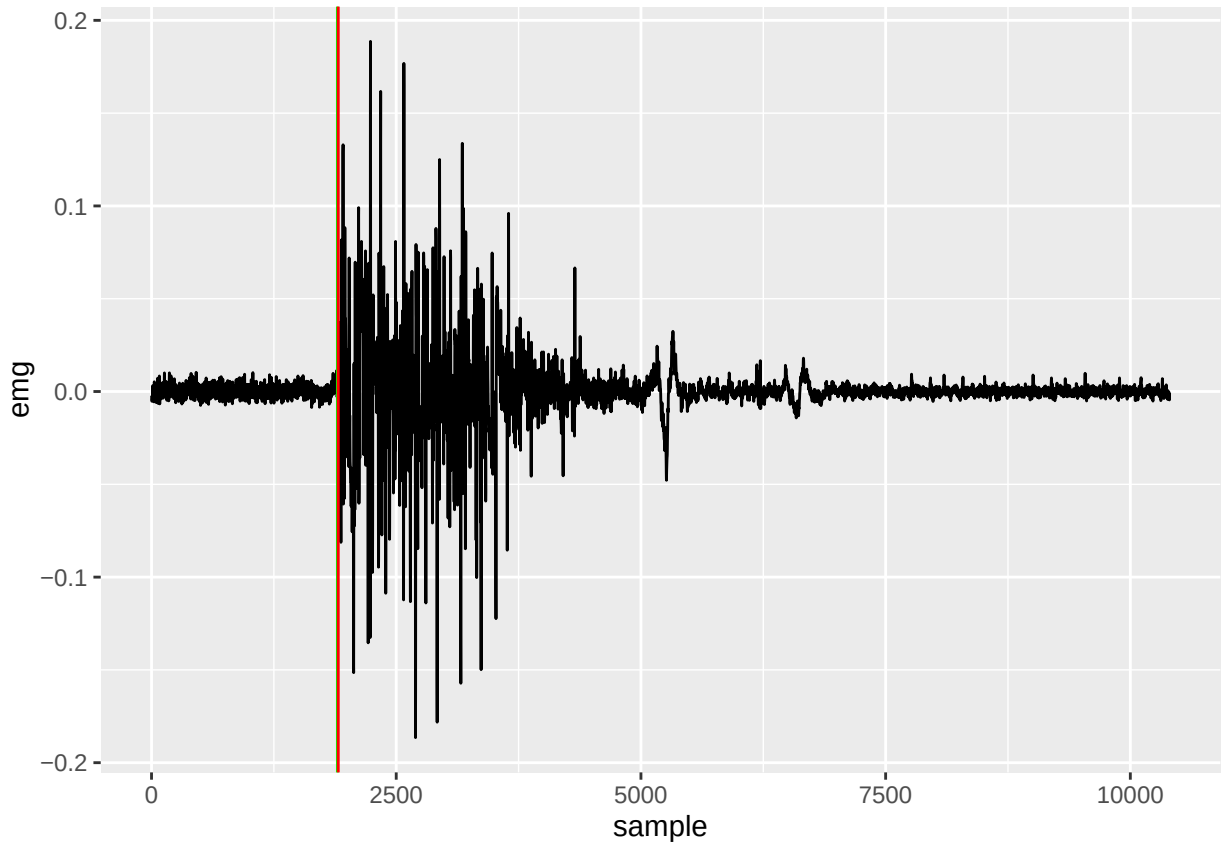
Now lets see what happens when the posterior probabilty onset threshold changed to one that didn't work out so well in our analysis, p0=0.1 and posterior probability of onset at 50%. We'll also suppress warnings because the model fit is generally so poor that the BCP algorithm let's us know things aren't right.

```
result_bcp <- suppressWarnings(bcp(fwEMGrect, p0 = 0.1))
onset_bcp <- which(result_bcp$posterior.prob >= 0.50)[1]

ggplot(emg_plot, aes(x=sample, y=emg)) +
  geom_line() +
  geom_vline(xintercept = onset_bcp, color = 'green') +
  geom_vline(xintercept = onset_visual, color = 'red')
```

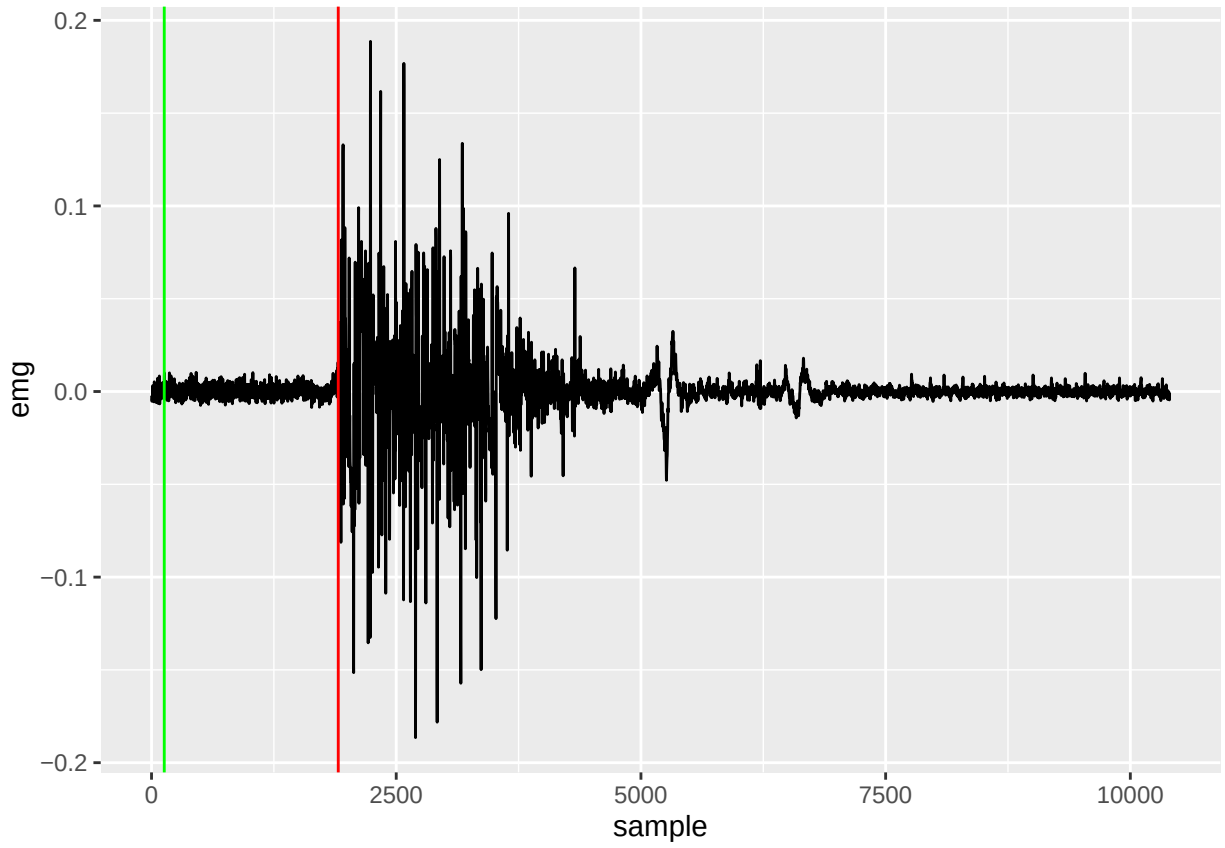


So in this case, the onset detection for this "poor" algorithm is actually closer to visual onset than the "good" algorithm and is more sensitive to that first motor unit spike. Let's look at a different trial from the biceps brachii, skipping some of the step-by-step details.

```
#extract a trial and rectify EMG
fwEMGrect <- abs(emg_ls$S10_a2)
#run BCP algorithm and extract the first point where posterior probability of onset is 95%
result_bcp <- bcp(fwEMGrect, p0 = 0)
onset_bcp <- which(result_bcp$posterior.prob >= 0.95)[1]
#Now we extract the visually determined EMG onset
onset_visual <- vis$value[match('S10_a2', vis$sbj)]


emg_plot <- data.frame(emg_ls$S10_a2)
colnames(emg_plot) <- 'emg'
sample <- seq.int(from =1, to = length(emg_plot$emg))
emg_plot <- cbind(emg_plot, sample)

#Plot EMG, BCP 95% onset and visual onset
ggplot(emg_plot, aes(x=sample, y=emg)) +
  geom_line() +
  geom_vline(xintercept = onset_bcp, color = 'green') +
  geom_vline(xintercept = onset_visual, color = 'red')
```

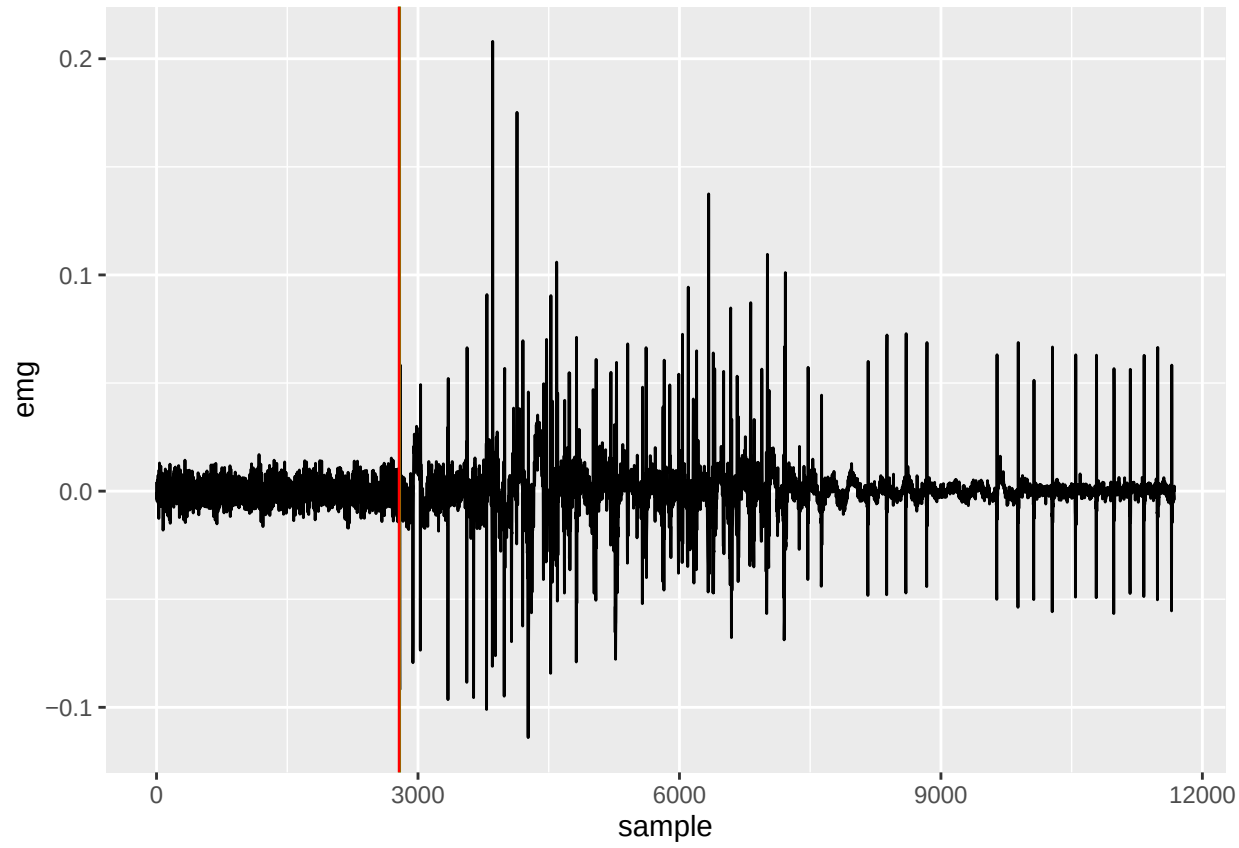



```
#extract p0=0.1 50% BCP onset
result_bcp <- suppressWarnings(bcp(fwEMGrect, p0 = 0.1))
onset_bcp <- which(result_bcp$posterior.prob >= 0.50)[1]
#Plot EMG, BCP onset and visual onset
ggplot(emg_plot, aes(x=sample, y=emg)) +
  geom_line() +
  geom_vline(xintercept = onset_bcp, color = 'green') +
  geom_vline(xintercept = onset_visual, color = 'red')
```

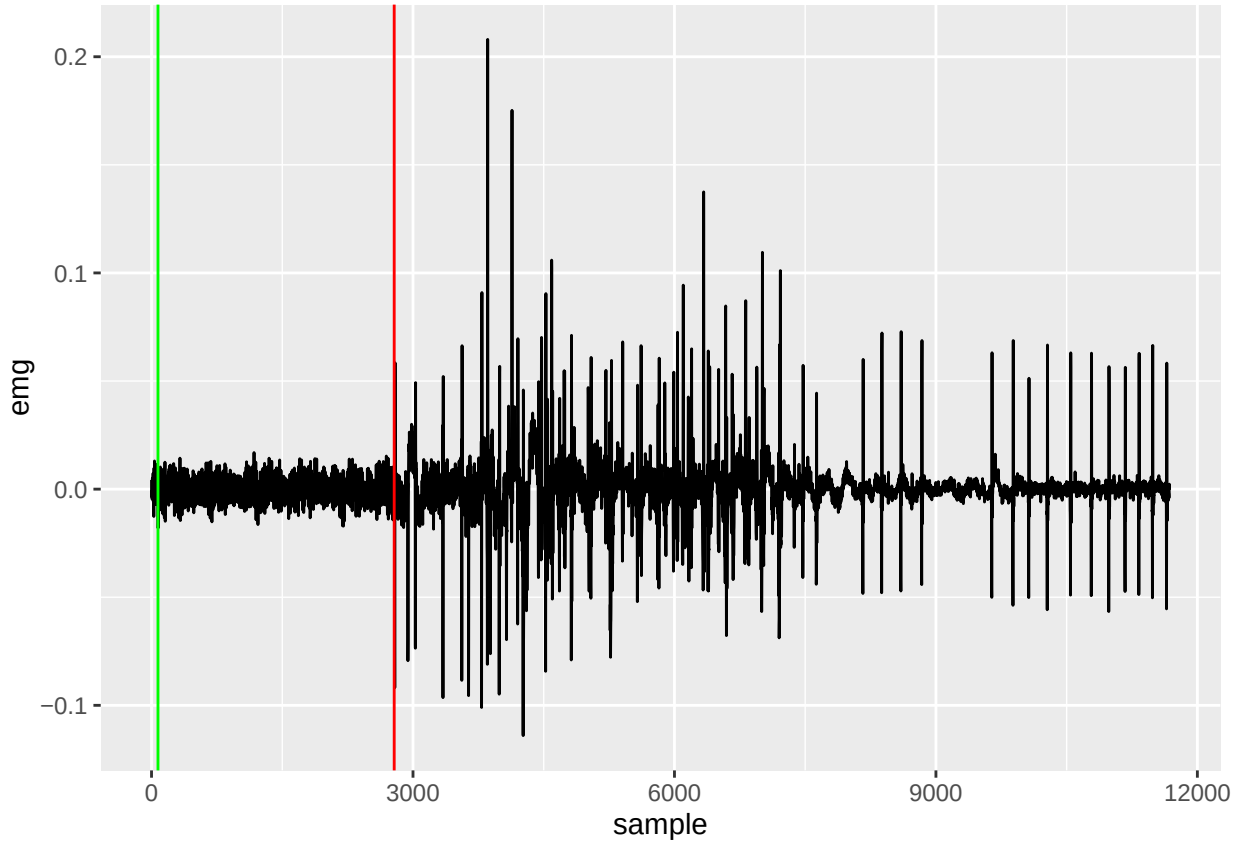


In this case, the 'good' BCP algorithm is in nearly perfect agreement with visual detection and the 'poor' BCP algorithm utterly fails. Let's look at one more then you can play with this data at your leisure.

```
#extract a trial and rectify EMG
fwEMGrect <- abs(emg_ls$S04_a2)
#run BCP algorithm and extract the first point where posterior probability of onset is 95%
result_bcp <- bcp(fwEMGrect, p0 = 0)
onset_bcp <- which(result_bcp$posterior.prob >= 0.95)[1]
#Now we extract the visually determined EMG onset
onset_visual <- vis$value[match('S04_a2', vis$sbj)]


emg_plot <- data.frame(emg_ls$S04_a2)
colnames(emg_plot) <- 'emg'
sample <- seq.int(from =1, to = length(emg_plot$emg))
emg_plot <- cbind(emg_plot, sample)

#Plot EMG, BCP 95% onset and visual onset
ggplot(emg_plot, aes(x=sample, y=emg)) +
  geom_line() +
  geom_vline(xintercept = onset_bcp, color = 'green') +
  geom_vline(xintercept = onset_visual, color = 'red')
```




```
#extract p0=0.1 50% BCP onset
result_bcp <- suppressWarnings(bcp(fwEMGrect, p0 = 0.1))
onset_bcp <- which(result_bcp$posterior.prob >= 0.50)[1]
#Plot EMG, BCP onset and visual onset
ggplot(emg_plot, aes(x=sample, y=emg)) +
  geom_line() +
  geom_vline(xintercept = onset_bcp, color = 'green') +
  geom_vline(xintercept = onset_visual, color = 'red')
```



Same result as the previous data trial! Hopefully you are enticed to play around more with these algorithms and our publically available data.