

Basic Lupus Data Metrics

Scooter Nowak

September 26, 2019

```
lupus<-read.table("lupustherm.txt")
data.temp<-as.matrix(lupus[,1:451])
classes.temp<-as.numeric(lupus[,452])
argvals=seq(45.0, 90.0, 0.1)

df.l <- data.frame(data.temp)

gen <- function(df, f){
  apply(df, 1, FUN=f)
}

gen_peaks <- function(y,p){
  x <- as.numeric(y)
  peaklist <- c()
  peak <- 0
  for (i in 1:length(x)){
    if (x[i] > peak){
      peak <- x[i]
      if (peak > x[i+1] &&
          peak > x[i+2] &&
          peak > x[i+3] &&
          peak > x[i+4] &&
          peak > x[i+5]){
        peaklist <- c(peaklist, peak)
      }
    }
  }
  sort(peaklist)[length(peaklist)-p]
}

get_peak1 <- function(x){
  output <- findpeaks(x, nups=2, ndowns=2, minpeakdistance=10, npeaks=2)
  output <- output[order(output[,2]),]
  output[1,1]
}

get_peak2 <- function(x){
  output <- findpeaks(x, nups=2, ndowns=2, minpeakdistance=10, npeaks=2)
  output <- output[order(output[,2]),]
  output[2,1]
```

```

}

temp <- seq(45.0, 90.0, by=0.1)
samples <- nrow(df.l)

df.stats <- data.frame(Sample=seq(1, nrow(df.l), by=1))
df.stats$sel <- classes.temp
df.stats$mean <- apply(df.l, 1, mean)
df.stats$median <- apply(df.l, 1, median)
df.stats$sd <- apply(df.l, 1, sd)
df.stats$max <- apply(df.l, 1, max)
df.stats$max <- temp[apply(df.l, 1, which.max)]
df.stats$min <- apply(df.l, 1, min)
df.stats$tarea <- apply(df.l, 1, function(x) sum(x)*0.1)

df.stats$peak1 <- apply(df.l, 1, get_peak1)
for (i in 1:nrow(df.l)){df.stats$tpeak1[i] <-
  temp[match(df.stats$peak1[i], df.l[i:i,])]}

df.stats$peak2 <- apply(df.l, 1, get_peak2)
for (i in 1:nrow(df.l)){df.stats$tpeak2[i] <-
  temp[match(df.stats$peak2[i], df.l[i:i,])]}

df.stats$peak12ratio <- df.stats$peak1/df.stats$peak2
df.stats$tfm <- apply(df.l, 1, function(x) sum(x*temp)/sum(x))
head(df.stats, 10)

```

```

##      Sample sel      mean  median      sd  max      min      tarea
## 1         1    0 0.10935194 0.068525 0.1084235 70.9 -0.000852 4.931772
## 2         2    0 0.09132607 0.030730 0.1162448 71.3 -0.054485 4.118806
## 3         3    0 0.08653814 0.032320 0.1062692 75.5 -0.040765 3.902870
## 4         4    0 0.09256824 0.029925 0.1116631 71.3 -0.033810 4.174828
## 5         5    0 0.09719367 0.034145 0.1158405 71.7 -0.043830 4.383435
## 6         6    0 0.11444986 0.077745 0.1101189 71.7 -0.002435 5.161689
## 7         7    0 0.09394277 0.035365 0.1101572 75.9 -0.031125 4.236819
## 8         8    0 0.09908360 0.041600 0.1073066 71.2 -0.016005 4.468670
## 9         9    0 0.09352736 0.031045 0.1076914 71.5 -0.023380 4.218084
## 10        10    0 0.08513426 0.037585 0.1296318 71.2 -0.153085 3.839555
##      peak1 tpeak1      peak2 tpeak2 peak12ratio      tfm
## 1 0.104365   62.3 0.304135   70.9 0.34315353 71.74293
## 2 0.293795   71.6 0.280225   74.8 1.04842537 70.58204
## 3 0.016315   56.6 0.272430   71.7 0.05988694 71.46144
## 4 0.294630   71.3 0.278200   73.7 1.05905823 70.59477
## 5 0.301665   71.4 0.286805   73.3 1.05181221 71.44548
## 6 0.297215   71.7 0.284075   74.7 1.04625539 72.04496
## 7 0.016155   51.4 0.288875   75.2 0.05592384 72.11027
## 8 0.295475   70.9 0.274245   73.9 1.07741253 71.56995
## 9 0.012160   54.4 0.273580   73.6 0.04444769 71.24064

```

```
## 10 0.299790    71.2 0.280570    74.6  1.06850340 68.00092

#  output <- output[order(output[,2]),]
ups <- 2
downs <- 2
dist <- 15

p.vector1 <- as.numeric(df.l[1:1,])
peaks1 <- findpeaks(p.vector1, nups=ups, ndowns=downs, minpeakdistance=dist, npeaks=2)
peaks1 <- peaks1[order(peaks1[,2]),]
peaks1

##           [,1] [,2] [,3] [,4]
## [1,] 0.104365  174  135  176
## [2,] 0.304135  260  258  293

p.vector2 <- as.numeric(df.l[2:2,])
peaks2 <- findpeaks(p.vector2, nups=ups, ndowns=downs, minpeakdistance=dist, npeaks=2)
peaks2 <- peaks2[order(peaks2[,2]),]
peaks2

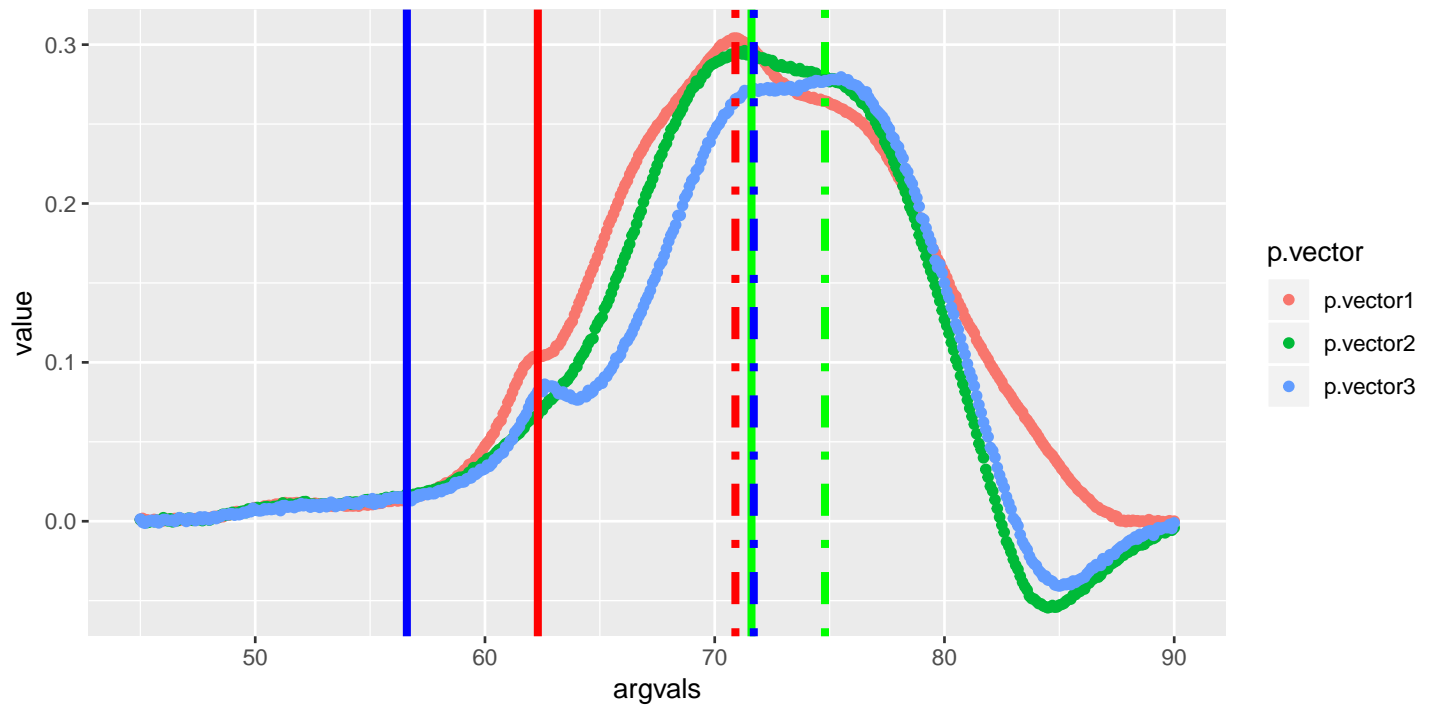
##           [,1] [,2] [,3] [,4]
## [1,] 0.293795  267  265  270
## [2,] 0.280225  299  297  301

p.vector3 <- as.numeric(df.l[3:3,])
peaks3 <- findpeaks(p.vector3, nups=ups, ndowns=downs, minpeakdistance=dist, npeaks=2)
peaks3 <- peaks3[order(peaks3[,2]),]
peaks3

##           [,1] [,2] [,3] [,4]
## [1,] 0.016315  117  115  119
## [2,] 0.272430  268  265  270

df.test <- data.frame(cbind(argvals, p.vector1, p.vector2, p.vector3))
df.test.long <- gather(df.test, p.vector, value, p.vector1:p.vector3, factor_key=TRUE)

ggplot(df.test.long, aes(y = value, x = argvals, color=p.vector)) +
  geom_point() +
  geom_vline(xintercept=df.stats[1,]$tpeak1, color='red', size=1.5) +
  geom_vline(xintercept=df.stats[1,]$tpeak2, color='red', size=1.5, linetype=4) +
  geom_vline(xintercept=df.stats[2,]$tpeak1, color='green', size=1.5) +
  geom_vline(xintercept=df.stats[2,]$tpeak2, color='green', size=1.5, linetype=4) +
  geom_vline(xintercept=df.stats[3,]$tpeak1, color='blue', size=1.5) +
  geom_vline(xintercept=df.stats[3,]$tpeak2, color='blue', size=1.5, linetype=4)
```



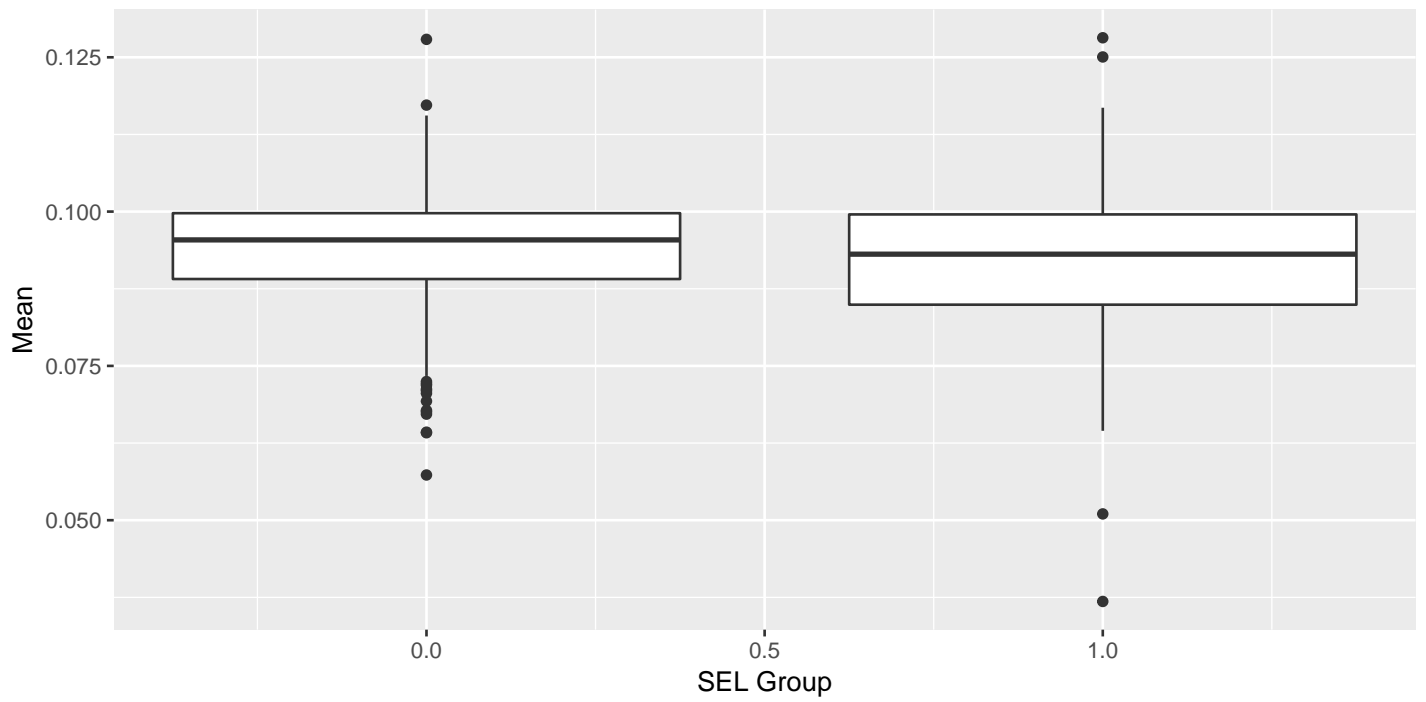
```
seq(45,90, 0.1)[135:176]
```

```
## [1] 58.4 58.5 58.6 58.7 58.8 58.9 59.0 59.1 59.2 59.3 59.4 59.5 59.6 59.7
## [15] 59.8 59.9 60.0 60.1 60.2 60.3 60.4 60.5 60.6 60.7 60.8 60.9 61.0 61.1
## [29] 61.2 61.3 61.4 61.5 61.6 61.7 61.8 61.9 62.0 62.1 62.2 62.3 62.4 62.5
```

```
seq(45,90, 0.1)[258:293]
```

```
## [1] 70.7 70.8 70.9 71.0 71.1 71.2 71.3 71.4 71.5 71.6 71.7 71.8 71.9 72.0
## [15] 72.1 72.2 72.3 72.4 72.5 72.6 72.7 72.8 72.9 73.0 73.1 73.2 73.3 73.4
## [29] 73.5 73.6 73.7 73.8 73.9 74.0 74.1 74.2
```

```
df.stats %>%
ggplot(aes(x=sel, group=sel, y=mean)) +
  geom_boxplot() +
  labs(x='SEL Group', y='Mean')
```



```
df.stats %>%
  ggplot(aes(x=sel, group=sel, y=tarea)) +
  geom_boxplot() +
  labs(x='SEL Group', y='Total Area')
```

