

# H1-T1: zona H1 (hybrid) del transecto 1.

José Carlos Del Valle

11/11/2021

## Índice

1. Introducción . . . . .	1
2. Importando y limpiando el dataset . . . . .	1
3. Obteniendo los promedios de las dos mediciones hechas en las partes basal y apical de la flor . . . . .	3

## 1. Introducción

En este script voy a analizar cómo ven las flores de *Erysimum merxmulleri* y *E. lagascae* los principales grupos de polinizadores de la zona: dípteros, coleópteros, lepidópteros e himenópteros. Este script será un poco más largo porque es la primera zona con la que trabajo y he hecho varios test que no tengo que repetir en los siguientes scripts. Cada zona la analizaré en scripts separados porque si no va a ser demasiado largo y un poco caótico

## 2. Importando y limpiando el dataset

Los datos de reflectancia de la zona T1-H1 están en la carpeta “T1\_Hibrida\_101-178”. En esa carpeta encontraremos dos mediciones: una hecha para la parte distal y otra para la parte apical. Nosotros utilizaremos el promedio de ambas mediciones, obteniendo tan solo una medición por flor.

Lo primero que haremos será cargar la librería *pavo*. Luego, cargaremos los datos (seleccionar el directorio apropiado) y crearemos el objeto **plotH1**.

```
library(pavo)
H1 <- "/Users/delValle/Dropbox/UGR/T1_Hibrida_101-178"
plotH1<-getspec(H1)
```

Ahora hay que pulir un poco los espectros en bruto que se obtienen del espectrofotómetro. Para ello utilizaremos las funciones “procspec” contenidas en el paquete *pavo*.

Lo primero que voy a hacer es eliminar posibles valores negativos de los espectros añadiendo el valor absoluto del valor negativo más alto a todo el espectro. A continuación, suavizo un poco los espectros.

```
plotH1.fix <- procspec(plotH1, fixneg = "addmin")
```

```
## processing options applied:
## Negative value correction: added min to all reflectance
```

```
plotH1.sm <- procspec(plotH1.fix, opt = "smooth", span = 0.2)
```

```
## processing options applied:  
## smoothing spectra with a span of 0.2
```

```
is.rspect(plotH1.sm)
```

```
## [1] TRUE
```

Ahora voy a plotear los dos espectros del individuo 101 (101a = parte apical del pétalo, 101b = parte basal del pétalo).

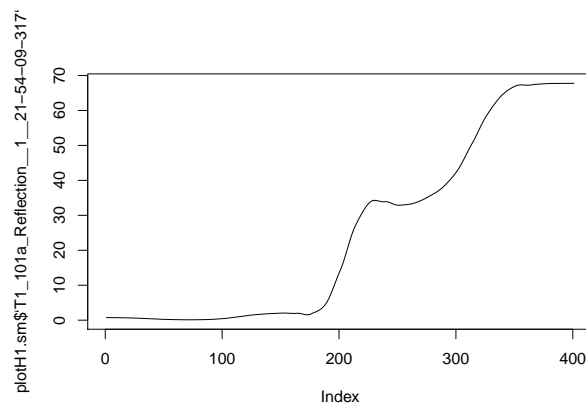


Figura 1: Parte apical de la flor del individuo 101.

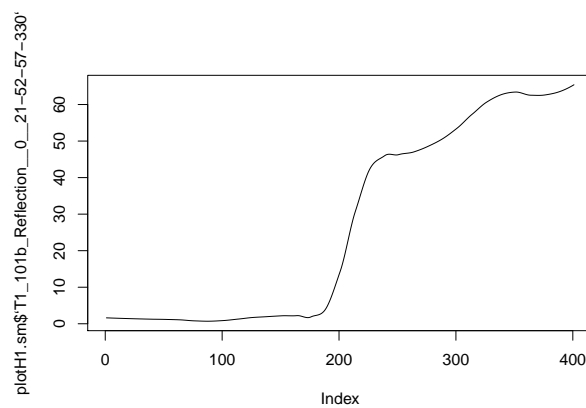


Figura 2: Parte basal de la flor del individuo 101.

Después de ver los espectros, no tengo demasiado claro si es buena idea mezclar las mediciones. Difieren demasiado entre ellos. De todas formas, voy a continuar así y calcularé el promedio de ambos espectros por individuo.

### 3. Obteniendo los promedios de las dos mediciones hechas en las partes basal y apical de la flor

Para promediar los espectros de las partes basal y apical de una flor, hay que calcular el promedio de dos columnas contiguas en *plotH1.sm* de cada muestra. Voy a hacer primero un pequeño test de forma manual utilizando varias columnas continuas, lo que me servirá como referencia más adelante para verificar que lo que estoy haciendo está bien

```
plotH1.sm_manual <- plotH1.sm$w1
plotH1.sm_manual <- as.data.frame(plotH1.sm_manual)
plotH1.sm_manual$T1_101 <- apply(plotH1.sm[,c(2,3)], 1, mean, na.rm = TRUE)
plotH1.sm_manual$T1_102 <- apply(plotH1.sm[,c(4,5)], 1, mean, na.rm = TRUE)
plotH1.sm_manual$T1_103 <- apply(plotH1.sm[,c(6,7)], 1, mean, na.rm = TRUE)
plotH1.sm_manual$T1_104 <- apply(plotH1.sm[,c(8,9)], 1, mean, na.rm = TRUE)
```

Lo primero que he hecho ha sido crear una primera columna con las longitudes de onda. He transformado el objeto *plotH1.sm\_manual* a data frame y a continuación he ido añadiendo columnas (una por individuo) con los promedios de ambas mediciones. Con una simple comprobación, vemos que el objeto *plotH1.sm\_manual* es reconocido correctamente como un objeto *rspec* (formato de *pavo*):

```
plotH1.sm_manual <- as.rspec(plotH1.sm_manual)
```

```
## wavelengths found in column 1
```

```
is.rspec(plotH1.sm_manual)
```

```
## [1] TRUE
```

Estos serían los cinco primeros valores que he obtenido para las muestras 101 a 104:

```
head(plotH1.sm_manual, n=5)
```

```
##      w1      T1_101      T1_102      T1_103      T1_104
## 1 300 1.177922 1.302502 1.022938 0.6922963
## 2 301 1.171390 1.296603 1.026394 0.7040755
## 3 302 1.164755 1.290602 1.029493 0.7153003
## 4 303 1.158008 1.284509 1.032242 0.7259692
## 5 304 1.151145 1.278331 1.034647 0.7360804
```

Como es inviable hacerlo manualmente, voy a crear un loop para que automáticamente me calcule el promedio cada dos columnas. Primero voy a correr un primer test con los promedios desde la columna 2 a la 11:

```
test0 <- plotH1.sm$w1
test0 <- as.data.frame(test0)
colnames(test0) <- c("w1")
head(test0)
```

```
##      w1
## 1 300
## 2 301
## 3 302
## 4 303
## 5 304
## 6 305
```

```
for(i in 2:11) {
  test0[ , i] <- apply(plotH1.sm[ , c(i, i+1)], 1, mean, na.rm=TRUE)
}
```

Al comparar los resultados con los obtenidos de forma manual (ver objeto *plotH1.sm\_manual*), veo que me ha sumado las columnas contiguas (ej: columna 2 + columna 3, columna 3 + columna 4, columna 4 + columna 5):

```
test0[1:5, 1:8]
```

```
##      wl      V2      V3      V4      V5      V6      V7      V8
## 1 300 1.177922 1.606348 1.302502 1.052695 1.022938 0.8655964 0.6922963
## 2 301 1.171390 1.594218 1.296603 1.053359 1.026394 0.8749099 0.7040755
## 3 302 1.164755 1.582198 1.290602 1.053727 1.029493 0.8837075 0.7153003
## 4 303 1.158008 1.570291 1.284509 1.053809 1.032242 0.8920017 0.7259692
## 5 304 1.151145 1.558501 1.278331 1.053614 1.034647 0.8998051 0.7360804
```

```
plotH1.sm_manual[1:5, 1:5]
```

```
##      wl  T1_101  T1_102  T1_103  T1_104
## 1 300 1.177922 1.302502 1.022938 0.6922963
## 2 301 1.171390 1.296603 1.026394 0.7040755
## 3 302 1.164755 1.290602 1.029493 0.7153003
## 4 303 1.158008 1.284509 1.032242 0.7259692
## 5 304 1.151145 1.278331 1.034647 0.7360804
```

Al sumar las columnas contiguas, únicamente las columnas pares (V2, V4, V6, etc.) son las correctas. Es decir, V2 equivale a T1\_101, V4 equivale a T1\_102, V6 equivale a T1\_103 y V8 equivale a T1\_104.

Voy a probar a eliminar las columnas impares y a retener las pares, pero antes tengo que quitar la primera columna, que es la que tiene las longitudes de onda.

```
test1 <- test0[ , -c(1)]
col_odd <- seq_len(ncol(test1)) %% 2
test1 <- test1[ , col_odd == 1]
rm(col_odd)
```

Ahora que solo tengo las columnas pares (las que tienen los promedios que estoy buscando), vuelvo a añadir la columna con las longitudes de onda (columna wl) que había eliminado. Ahora sí que me coinciden los resultados:

```
test1$wl <- test0$wl
library(dplyr)
test1 <- test1 %>%
  select(wl, everything())
```

Ahora sí que me coinciden los resultados:

```
plotH1.sm_manual[1:5, 1:5]
```

```
##      w1      T1_101      T1_102      T1_103      T1_104
## 1 300 1.177922 1.302502 1.022938 0.6922963
## 2 301 1.171390 1.296603 1.026394 0.7040755
## 3 302 1.164755 1.290602 1.029493 0.7153003
## 4 303 1.158008 1.284509 1.032242 0.7259692
## 5 304 1.151145 1.278331 1.034647 0.7360804
```

```
test1[1:5, 1:5]
```

```
##      w1      V2      V4      V6      V8
## 1 300 1.177922 1.302502 1.022938 0.6922963
## 2 301 1.171390 1.296603 1.026394 0.7040755
## 3 302 1.164755 1.290602 1.029493 0.7153003
## 4 303 1.158008 1.284509 1.032242 0.7259692
## 5 304 1.151145 1.278331 1.034647 0.7360804
```

```
rm(test0)
```

Ahora solo me faltaría cambiar el nombre de las variables y ya estaría listo:

```
x0 <- c("w1")
x1 <- 101:125
x2 <- 127:150
x3 <- 152:158
x4 <- 160:178
```

Como faltan algunas muestras (ej: 126), las secuencias no van del 101 al 178 directamente, sino que he creado objetos *x* con una numeración acorde al nombre de las muestras. Solo tengo que unificar estos objetos y añadirle al nombre de cada individuo (nombre de la columna) el prefijo “T1\_” correspondiente al transecto

```
x_sum <- c(x1, x2, x3, x4)
x_sum <- paste("T1", x_sum, sep="_")
head(x_sum)
```

```
## [1] "T1_101" "T1_102" "T1_103" "T1_104" "T1_105" "T1_106"
```

Todo esto que he hecho serviría para tener los nombres de todas las muestras, pero para este pequeño test solo necesito los nombres desde el T\_101 hasta el T1\_105:

```
x_sum_test1 <- c(x_sum[1:5])
names_test1 <- c(x0, x_sum_test1)
names_test1
```

```
## [1] "w1"      "T1_101" "T1_102" "T1_103" "T1_104" "T1_105"
```

Ya tengo listo el vector con los nombres de las columnas del dataframe *test1*

```
colnames(test1) <- names_test1
head(test1, n=5)
```

```
##      w1    T1_101    T1_102    T1_103    T1_104    T1_105
## 1 300 1.177922 1.302502 1.022938 0.6922963 1.303106
## 2 301 1.171390 1.296603 1.026394 0.7040755 1.315932
## 3 302 1.164755 1.290602 1.029493 0.7153003 1.328119
## 4 303 1.158008 1.284509 1.032242 0.7259692 1.339655
## 5 304 1.151145 1.278331 1.034647 0.7360804 1.350528
```

Así es como consigo obtener los valores promedios de cada muestra (T1\_101, T1\_102, etc.) con su correspondiente nomenclatura; con esto podría trabajar ya y extrapolarlo al conjunto completo de datos.

Elimino todo aquello que ya no necesito antes de crear un data frame con todos los datos de interés:

```
rm(x1)
rm(x2)
rm(x3)
rm(x4)
rm(x0)
rm(x_sum)
rm(x_sum_test1)
rm(test1)
rm(names_test1)
rm(i)
```