

IGAC
INSTITUTO GEOGRÁFICO
AGUSTÍN CODAZZI



Sistema de Gestión
Integrado
MIPG



IGAC
INSTITUTO GEOGRÁFICO
AGUSTÍN CODAZZI



Sistema de Gestión
Integrado
MIPG



Instructivo

Procesamiento y Análisis de Firmas Espectrales

Código: IN-IIA-PC02-03

Versión: 1

Vigente desde: 14/08/2024

1. OBJETIVO

Describir a través de un enfoque claro y accesible los pasos a seguir para el procesamiento de datos hiperespectrales haciendo uso de herramientas desarrolladas en R, para análisis de firmas espectrales para la teledetección.

2. ALCANCE

El presente instructivo aplica al proceso de Gestión del Conocimiento Aplicado y hace parte del subproceso de Investigación e Innovación Aplicada, a los servidores públicos y contratistas del Laboratorio Nacional de Espectroradiometría (LNE) del Instituto Geográfico Agustín Codazzi (IGAC).

Se abarca el uso de código desde el software "R" para la creación librerías y la obtención de parámetros estadísticos empleando los fundamentos físicos de los elementos observados. La implementación de métodos estadísticos se hace con dos fines: el primero, realizar los ajustes, mejoras, y obtención de endmembers (firmas tipo) de las firmas espectrales, y el segundo, obtener las bandas óptimas de los endmembers para la clasificación de imágenes.

3. DEFINICIONES

- **Análisis estadístico:** Proceso de recopilación, organización, análisis e interpretación de datos para descubrir patrones, tendencias, relaciones y significado en ellos. Implica el uso de técnicas estadísticas para describir y resumir datos, así como para tomar decisiones informadas (Ali & Bhaskar, 2016).
- **Banco de firmas espectrales:** Conjunto organizado de firmas espectrales que representa una variedad de materiales o clases de interés en una escena hiperespectral. Se utiliza como referencia para la interpretación y clasificación de datos hiperespectrales.
- **Banda espectral:** Rango específico de longitudes de onda en el espectro electromagnético.
- **CRAN:** Red Integral de Archivos de R (las siglas corresponden a Comprehensive R Archive Network).
- **CRAN Mirrors:** Red de servidores FTP y web distribuidos por todo el mundo que almacenan copias idénticas y actualizadas del código y la documentación de R.
- **Clasificación de las imágenes:** Técnica de asignar etiquetas o categorías a cada píxel en una imagen basada en sus firmas espectrales.
- **Datos hiperespectrales:** Conjunto de datos que capturan información en muchas bandas estrechas a lo largo del espectro electromagnético.
- **Endmembers:** Firmas espectrales representativas de los materiales o clases de interés, presentes en una escena hiperespectral. Estos espectros representan puntos extremos en el espacio de características y son utilizados para interpretar y clasificar los datos hiperespectrales.
- **Espectroradiometría:** Medición de la radiación electromagnética en función de la longitud de onda. (R. D. Brown, 1995).
- **Filtrado espectral:** Proceso de eliminación o reducción de ruido y señales no deseadas en los datos espectrales. Se utilizan diversos filtros y técnicas de procesamiento para mejorar la calidad de los espectros y resaltar las características importantes.
- **Firma espectral:** Representación gráfica de la reflectancia o emisión de una superficie en función de la longitud de onda.
- **Distribución Espectral:** Distribución de la intensidad de la radiación electromagnética en función de la longitud de onda (G. Wyznecki, W. S. Stiles, 1982).
- **Índice de distancia J-M (Jeffries–Matusita):** Medida de similitud o diferencia entre dos distribuciones de probabilidad multivariadas. Se utiliza en el análisis de datos hiperespectrales para evaluar la separabilidad espectral entre diferentes clases o materiales (Sekhar CS, 2024).
- **Longitud de Onda:** Distancia entre dos crestas sucesivas de una onda electromagnética (W. L. Barnes, et al. 2003).

Metodologías: Conjunto de métodos y procedimientos utilizados para realizar una tarea o alcanzar un objetivo. (D. S. Yadav, et al. 2010; pg.154-160)

- **Normalización:** Proceso de ajustar los valores de las firmas espectrales para eliminar variaciones no deseadas y hacer comparables diferentes conjuntos de datos.
- **Posit Cloud:** Plataforma en la nube para ejecutar proyectos de R Studio sin necesidad de instalación local.
- **Proceso de Verificación:** Acción de confirmar la precisión y validez de los datos. T. F. (Hockley, et al. (1981) pg.441-463).
- **R:** Entorno de programación y lenguaje de programación especialmente diseñado para el análisis estadístico y la generación de gráficos. Es un software de código abierto y gratuito, lo que significa que cualquier persona puede descargarlo, utilizarlo y contribuir a su desarrollo
- **Radiación Electromagnética:** Energía que se propaga en forma de ondas electromagnéticas (A. H. Zewail (2000) pg. 5660-5694).
- **Reflectancia:** Fracción de la radiación incidente que es reflejada por una superficie.
- **Remuestreo espectral:** Proceso de ajuste de la resolución espectral de los datos hiperespectrales para que coincida con la resolución espectral de otras imágenes o sensores. Se realiza para facilitar la comparación y la integración de datos de diferentes fuentes.
- **Satelitales:** Relativo a los satélites artificiales en órbita alrededor de la Tierra u otros cuerpos celestes (C. Elachi, 1987).
- **Seguridad:** Condiciones y medidas para proteger a las personas y los equipos contra daños (J. H. Moxnes. 2017).
- **Summary ():** Función de R que proporciona un resumen estadístico de un objeto.
Uso: Análisis exploratorio de datos para comprender su distribución y características principales.
- **Teledetección:** Obtención de información sobre un objeto sin estar en contacto directo con él (S. J. Wang, 2012).

4. DESARROLLO

A continuación, se proporcionan los pasos a seguir para el procesamiento de datos hiperespectrales provenientes de las firmas espectrales tomadas con espectroradiómetros, obtenidos en campo o en el laboratorio, como parte de los instrumentos desarrollados por el laboratorio de espectroradiometría del Instituto Geográfico Agustín Codazzi. Se centra específicamente en el procesamiento de firmas para la creación de librerías que pueden ser usadas en múltiples aplicaciones y en específico, para el procesamiento de imágenes con aplicaciones en teledetección.

4.1 INSUMOS

4.1.1 EQUIPOS

REQUERIMIENTOS DEL SISTEMA	
Computador/Procesador	PC o MAC
Sistema Operativo	Windows 7, 8.1, 10,11
Memoria RAM Mínimo	8 Gb.
Software	R, RStudio (Version 4.3.3)

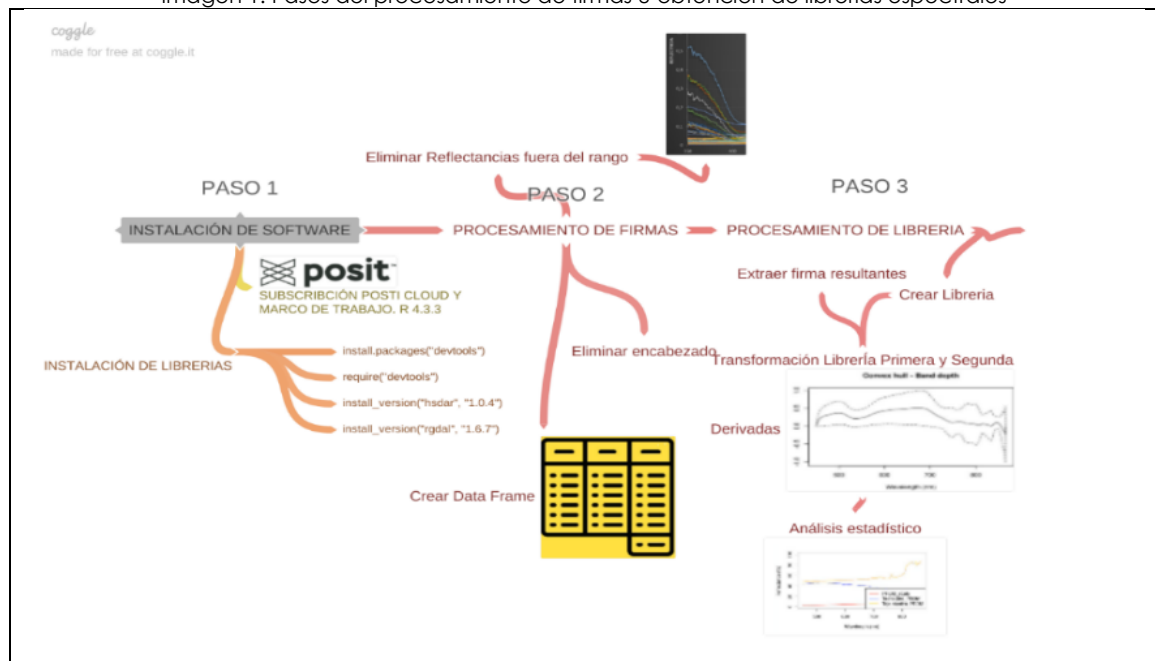
4.1.2 INSUMOS TÉCNICOS (SOFTWARE)

Previo al inicio de procesamiento se requiere la instalación de los softwares de R y RStudio o disponer de la versión en la nube Posit Cloud.

4.2 PROCESAMIENTO DE LAS FIRMAS

El procesamiento de firmas espectrales incluye el análisis y mejora de las firmas espectrales mediante métodos estadísticos empleando los fundamentos físicos de los elementos observados. El desarrollo de esta metodología incluye la instalación de software y paquetes de procesamiento de firmas hiperespectrales (Paso 1), El pre-procesamiento para la depuración de las firmas (Paso 2) y la obtención de firmas espectrales analizadas con parámetros estadísticos básicos y de separabilidad espectral (Paso 3). Finalmente, se obtendrán firmas resultantes que podrán exportadas para ser empleadas desde distintas plataformas (Imagen 1).

Imagen 1. Pasos del procesamiento de firmas u obtención de librerías espectrales



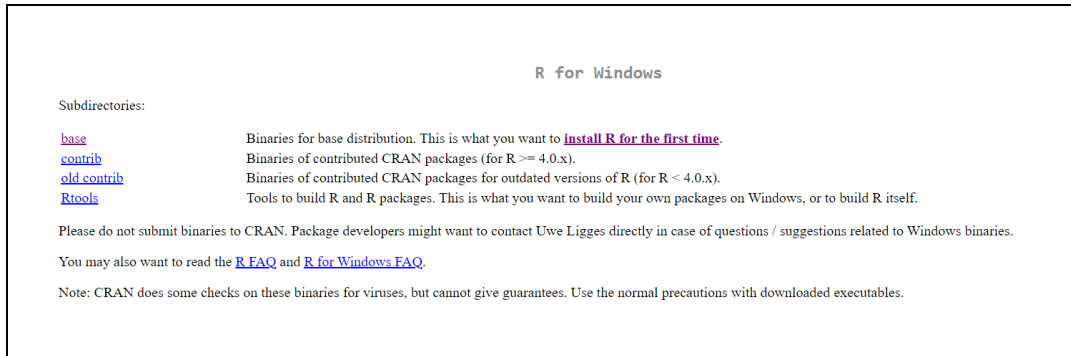
Fuente: IGAC (2024)

4.2.1 INSTALACIÓN DEL SOFTWARE

Descarga de R:

- Diríjase e al sitio web oficial de R en <https://www.r-project.org/>.
- De clic en "CRAN" bajo la sección "Download" en el lado izquierdo de la página.
- Se despliegan los espejos CRAN (CRAN Mirrors), seleccione un espejo cercano a su ubicación geográfica.
- Haga clic en el enlace correspondiente a su sistema operativo (por ejemplo, "Download R for Windows") (Imagen 2b).
- Nota: En caso de ejecutarse la página de los subdirectorios de R para windows (Imagen 2a) seleccione "Install R for the first time" para redirigirse al enlace de instalación "Download R for windows"

Imagen 2a. Subdirectorios de R para instalación en Windows



Fuente <https://www.r-project.org/>

Imagen 2b. Descarga software R

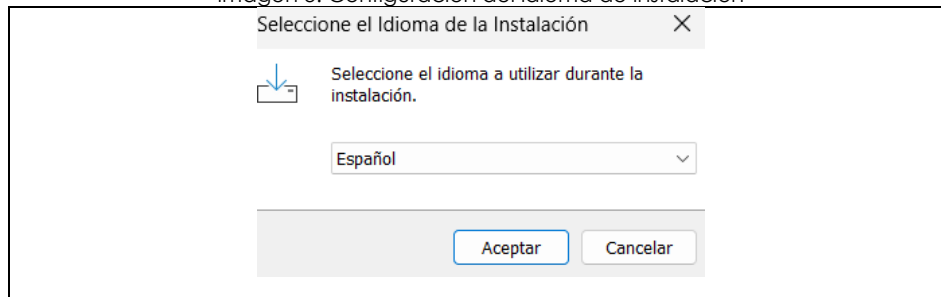


Fuente <https://www.r-project.org/>

○ **Instalación de R:**

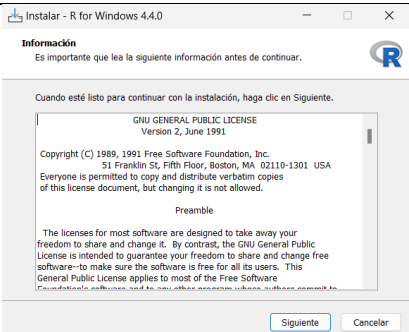
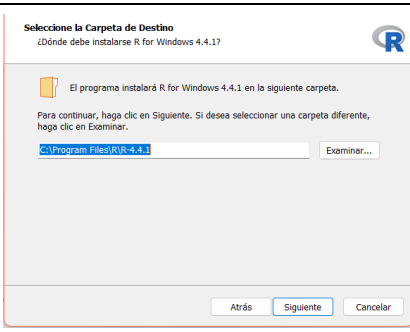
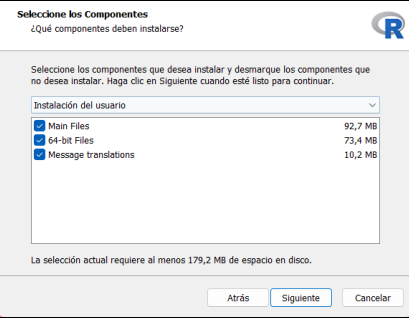
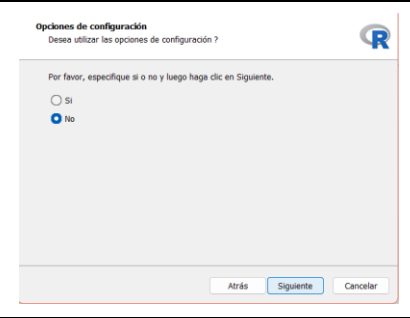
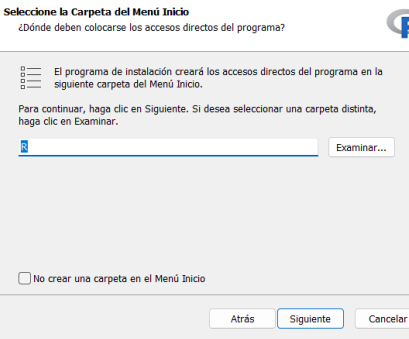
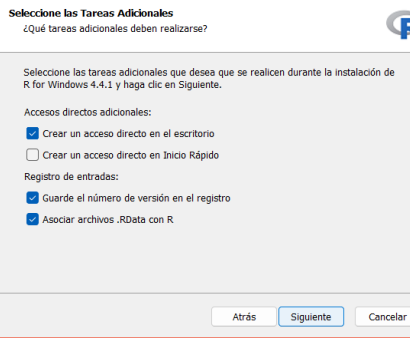
- Una vez descargado el instalador, de doble clic en el archivo para ejecutarlo. Se desplegará la ventana de instalación.
- Seleccione configuración de idioma (Imagen 3):

Imagen 3. Configuración del idioma de instalación



Fuente: IGAC, 2024

- Seleccione el idioma en el que desea que se instale R y haga clic en "OK".
 - Inicie la instalación (Imagen 4):
 - Haga clic en "Next" para iniciar el proceso de instalación.
 - Acuerdo de licencia: Lea el acuerdo de licencia y, si está de acuerdo, seleccione "I accept the agreement" y luego haga clic en "Next" (Imagen 4a).
 - En la pestaña se presentará la carpeta de destino donde se instalará R, si desea cambiar la carpeta definida por defecto de clic en el botón "examinar" y seleccione la ruta de su preferencia. Si no es el caso, seleccione "Next" (Imagen 4b).
- A continuación, se presentan las ventanas "Seleccione los componentes" (Imagen 4c), "Opciones de configuración" (Imagen 4d), "Seleccione la carpeta del Menú de Inicio" (Imagen 4e) y "Seleccione las tareas adicionales" (Imagen 4f), estas configuraciones deben dejarse por defecto, por lo que puede dar en "Next" hasta ejecutar la instalación.

Ventanas de Instalación de “R”	
 <p style="text-align: center;">4a</p>	 <p style="text-align: center;">4b</p>
 <p style="text-align: center;">4c</p>	 <p style="text-align: center;">4d</p>
 <p style="text-align: center;">4e</p>	 <p style="text-align: center;">4f</p>

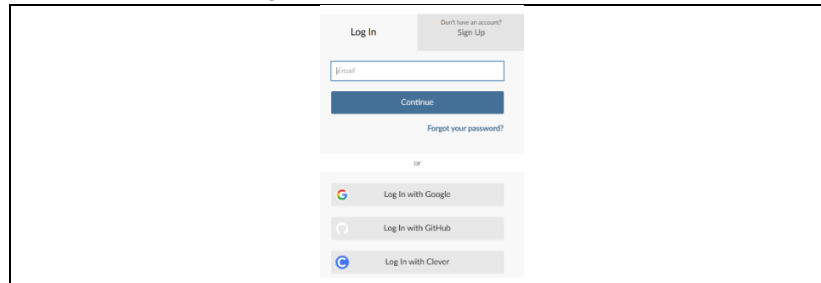
Fuente: IGAC, 2024

4.2.1.1. Instalación de RStudio:

- **Descarga de RStudio:** Vaya al sitio web oficial de RStudio: <https://www.rstudio.com/products/rstudio/download/>. Allí encontrará diferentes versiones de RStudio disponibles para distintos sistemas operativos (Windows, macOS y Linux). Descargue la versión adecuada para tu sistema.
 - **Windows:** Una vez que la descarga esté completa, haga doble clic en el archivo de instalación (.exe) descargado. Sigue las instrucciones del instalador para completar la instalación.
 - **MacOS:** Abra el archivo .dmg descargado y arrastre la aplicación RStudio a la carpeta de Aplicaciones. También puede instalarlo desde la línea de comandos usando Homebrew.
 - **Linux:** Dependiendo de tu distribución, puedes instalar RStudio desde los repositorios oficiales o utilizando paquetes específicos de RStudio. Consulte la documentación de su distribución de linux para obtener instrucciones detalladas.

- **Ejecución de RStudio:** Una vez completada la instalación, puede ejecutar RStudio desde el menú de inicio (en Windows), desde la carpeta de Aplicaciones (en macOS), o desde el terminal (en Linux) utilizando el comando "rstudio".
- **Configuración inicial:** Cuando abra RStudio por primera vez, es posible que le solicite configurar algunas opciones, como la ubicación de su instalación de R o la interfaz. Siga las instrucciones según sus preferencias.
- **Inicio de RStudio desde la nube (POSIT Cloud)** Otra opción para iniciar el procesamiento de firmas es cargar directamente un servicio cloud para correr el proyecto en R estudio sin tener la necesidad de la instalación directa del software de R y Rstudio. Los pasos para seguir en Posit Cloud son los siguientes:
- **Iniciar sesión**
Si ya tiene una cuenta en Posit Cloud (o una cuenta en shinyapps.io), inicie sesión. de lo contrario, regístrese para obtener una cuenta gratuita para empezar.
 - Debe iniciar sesión para crear y ver contenido en Posit Cloud (Imagen 5).

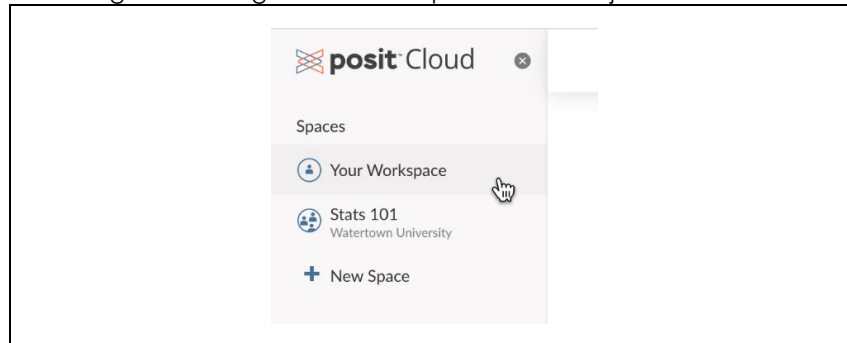
Imagen 5. Inicio de sesión Posit Cloud.



Fuente: Posit Cloud (2024)

- Vaya a su espacio de trabajo utilizando la barra lateral de navegación situada en la parte izquierda de la ventana (Imagen 6).

Imagen 6. Configuración del espacio de trabajo en Posit Cloud

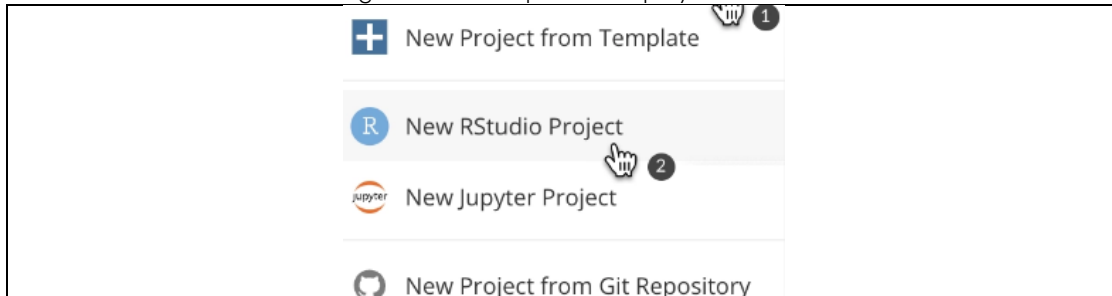


Fuente: Posit Cloud (2024)

- **Crear un proyecto**
Un proyecto es la unidad fundamental de trabajo en Posit Cloud. Encapsula su código, paquetes y archivos de datos y proporciona aislamiento de otros análisis.

Para crear un proyecto, haga clic en el botón Nuevo Proyecto, luego elija “Nuevo Proyecto RStudio” en el menú que aparece (Imagen 7).

Imagen 7. Ventana para crear proyecto nuevo



Fuente: Posit Cloud (2024)

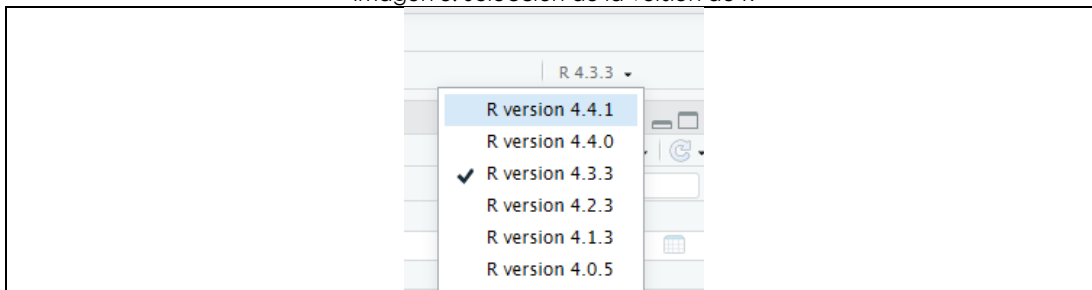
- **Instalando librerías**

Tanto en la versión de RStudio del escritorio cómo el desarrollo del procesamiento de las firmas en posit los pasos a seguir son los mismos solo se cambia la forma en que se carga la información. En el numeral anterior se describe cómo cargar los datos en Rstudio

- **Instalando las librerías y versión de R:**

Asegúrese de revisar desde Posit Cloud que se encuentre seleccionada la versión de R 4.3.3, puede cambiarlo en el menú superior derecho del espacio de trabajo (Imagen 8), luego de esto realice la instalación de las librerías.

Imagen 8. Selección de la versión de R



Fuente: Posit Cloud (2024)

Ejecute las siguientes líneas de código en la consola para realizar la instalación de las librerías copiando y pegando y posteriormente haga clic en el botón “Enter”

```
install.packages("devtools")
require("devtools")
install_version("hdsar", "1.0.4")
install_version("rgdal", "1.6.7")

library(hdsar)
library(rgdal)
```

4.2.2 PROCESAMIENTO DE FIRMAS.

◦ **Eliminación del encabezado:**

Cuando se recolectan firmas en campo o en laboratorio se genera un archivo *.txt que contiene un encabezado general del momento de la toma de la firma y los valores correspondientes. El encabezado de información que contiene cada archivo de firma depende la empresa que desarrolla los espectroradiómetros.

En los equipos del Laboratorio (en su mayoría Ocean Optics™), el encabezado contiene 14 filas con información sobre el nombre del archivo, usuario del equipo, modelo del espectroradiómetro, tiempo de integración, número de lecturas y grado de alisado de la firma resultante. Estos datos hacen parte del metadato que se requiere para documentar las firmas, pero no son útiles para el procesamiento por lo que es importante quitarla para cada firma durante el procesamiento de los datos (Imagen 9). Las siguientes columnas luego de línea titulada "Begin Spectral Data" corresponde a la longitud de onda y reflectancias entre 0 y 100%.

Para eliminar el encabezado emplear un editor de texto y eliminar las primeras 14 líneas hasta "Begin Spectral Data"

Imagen 9. Encabezado de información de los archivos de texto de las firmas.

```

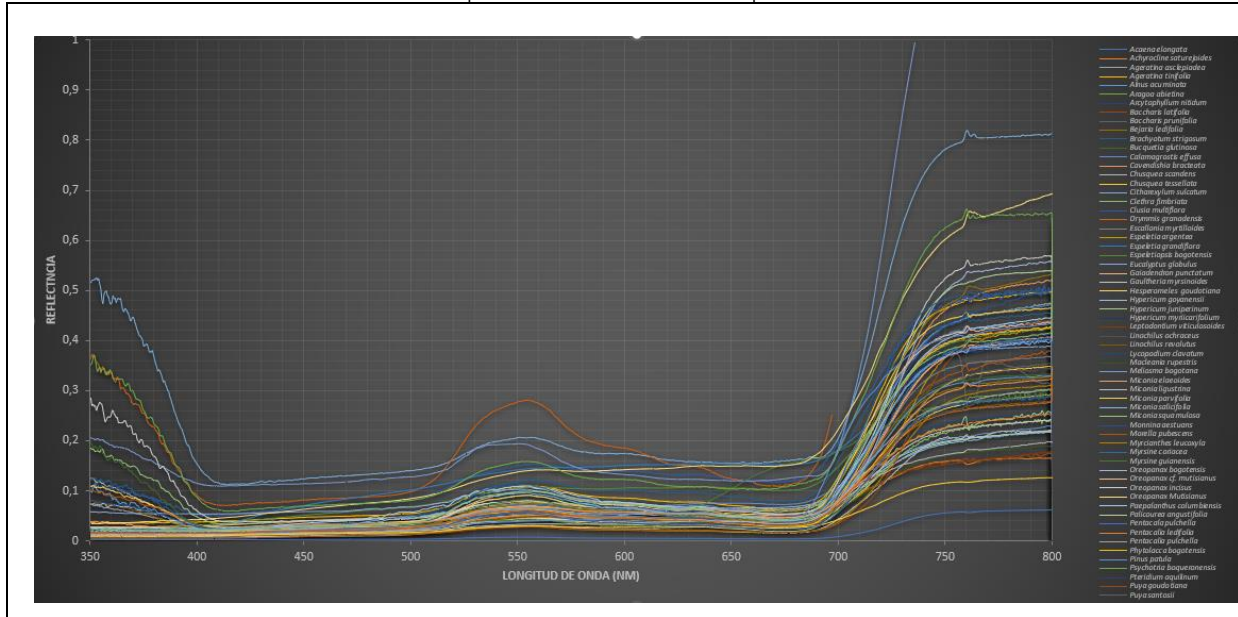
Data from Reflection_0_12-25-07-955.txt Node
Date: Fri May 17 12:25:07 COT 2024
User: nando
Spectrometer: HDX01620
Trigger mode: 0
Integration Time (sec): 1,500000E-2
Scans to average: 4
Nonlinearity correction enabled: false
Boxcar width: 8
Storing dark spectrum: false
XAxis mode: Wavelengths
Number of Pixels in Spectrum: 2068
>>>>Begin Spectral Data<<<<
194,714 7,6
195,089 9,14
195,464 9,11
195,838 8,12
196,213 8,82
196,588 9,25
196,962 8,71
197,336 8,9
197,711 8,45
198,085 7,83
    
```

Fuente: IGAC, 2024

◦ **Eliminar reflectancias fuera del rango:**

- Cada espectroradiómetro tiene un rango efectivo en el cual sus lecturas de reflectancia son confiables y están libres de distorsión e interferencia. Estos rangos específicos de operación deben ser consultados en la documentación técnica del equipo utilizado, como en el caso de Ocean Optics el rango va de 400 nm hasta 760 nm (Imagen 10).
- Se debe abrir el archivo de texto correspondiente a cada firma espectral y eliminar las reflectancias que se encuentran fuera del rango efectivo. Este paso es crucial para asegurar la precisión y calidad de los datos espectrales.

Imagen 10. Representación de firmas espectrales mostrando aquellas menores a 400 nm y superiores a 760 nm fuera del rango efectivo de lectura del espectroradiómetro Ocean optics.



Fuente: IGAC (2024)

Finalmente se crea un archivo de texto *.csv (delimitado por comas) común para todas las firmas depuradas, donde la primera columna es la longitud dentro de área efectiva de lectura del instrumento, y las columnas los datos de cada firma editada.

o **Cargue archivo de datos *.csv en RStudio:**

Luego de instalar las librerías se debe importar el archivo de datos de la firma espectral en el proyecto de RStudio (sea en la ejecución local o desde posit cloud). La estructura de los datos debería seguir el siguiente formato requerido por la Liberia HSDAR.

Para el presente ejercicio se empleó una librería que contiene datos espectrales de cubiertas como tejas, techos y suelos. Las longitudes de onda representan las columnas y los tipos de cubiertas las filas. El separador decimal el punto y el separador de los datos de campo es una coma (Imagen 11).

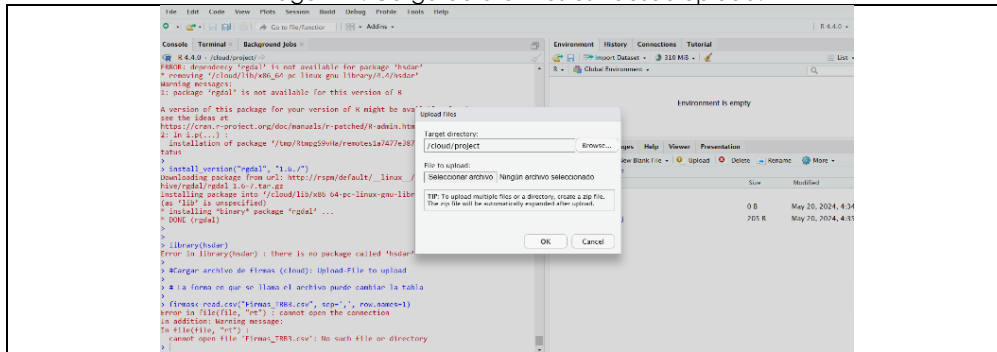
Imagen 11. Formato del archivo de carga csv

```
banda,450,451,452,453,454,455,456,457,458,459,460,461,462,463,464,465,466,467,468,469,470,471,472,473,474,475,476,477,478,479,480,481,482,483,484,485,486,487,488,489,490,491,492,493,494,495,496,497,498,499,500,501,502,503,504,505,506,507,508,509,510,511,512,513,514,515,516,517,518,519,520,521,522,523,524,525,526,527,528,529,530,531,532,533,534,535,536,537,538,539,540,541,542,543,544,545,546,547,548,549,550,551,552,553,554,555,556,557,558,559,560,561,562,563,564,565,566,567,568,569,570,571,572,573,574,575,576,577,578,579,580,581,582,583,584,585,586,587,588,589,590,591,592,593,594,595,596,597,598,599,600,601,602,603,604,605,606,607,608,609,610,611,612,613,614,615,616,617,618,619,620,621,622,623,624,625,626,627,628,629,630,631,632,633,634,635,636,637,638,639,640,641,642,643,644,645,646,647,648,649,650,651,652,653,654,655,656,657,658,659,660,661,662,663,664,665,666,667,668,669,670,671,672,673,674,675,676,677,678,679,680,681,682,683,684,685,686,687,688,689,690,691,692,693,694,695,696,697,698,699,700,701,702,703,704,705,706,707,708,709,710,711,712,713,714,715,716,717,718,719,720,721,722,723,724,725,726,727,728,729,730,731,732,733,734,735,736,737,738,739,740,741,742,743,744,745,746,747,748,749,750,751,752,753,754,755,756,757,758,759,760,761,762,763,764,765,766,767,768,769,770,771,772,773,774,775,776,777,778,779,780,781,782,783,784,785,786,787,788,789,790,791,792,793,794,795,796,797,798,799,800,801,802,803,804,805,806,807,808,809,810,811,812,813,814,815,816,817,818,819,820,821,822,823,824,825,826,827,828,829,830,831,832,833,834,835,836,837,838,839,840,841,842,843,844,845,846,847,848,849,850,851,852,853,854,855,856,857,858,859,860,861,862,863,864,865,866,867,868,869,870,871,872,873,874,875,876,877,878,879,880,881,882,883,884,885,886,887,888,889,890,891,892,893,894,895,896,897,898,899,900,901,902,903,904,905,906,907,908,909,910,911,912,913,914,915,916,917,918,919,920,921,922,923,924,925,926,927,928,929,930,931,932,933,934,935,936,937,938,939,940,941,942,943,944,945,946,947,948,949,950,951,952,953,954,955,956,957,958,959,960,961,962,963,964,965,966,967,968,969,970,971,972,973,974,975,976,977,978,979,980,981,982,983,984,985,986,987,988,989,990,991,992,993,994,995,996,997,998,999,1000
```

Fuente: IGAC (2024)

Cargue el archivo generado con las firmas a trabajar (este debe ser de extensión *.csv) desde el botón "Upload" desde Posit cloud. (Imagen 12).

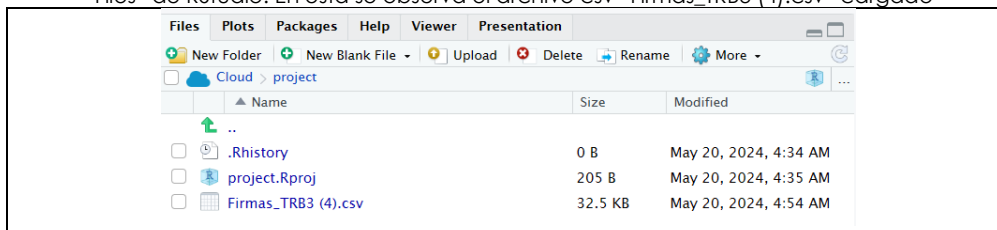
Imagen 12. Carga de archivos csv desde upload.



Fuente: IGAC (2024)

El archivo cargado puede visualizarse en la ventana "Files" en Rstudio (Imagen 13)

Imagen 13. Ventana "Files" de RStudio. En esta se observa el archivo csv "Firmas_TRB3 (4).csv" cargado



Fuente: IGAC, 2024

Ejecute la siguiente línea de código para cargar el archivo y luego ejecute la variable creada "firmas" para ver en la consola el resultado, siendo este la estructura del archivo *.csv y de los datos del mismo (Imagen 14):

```
firmas<-read.csv("Firmas_TRB3.csv", sep=',', row.names=1)firmas
```

Imagen 14. Representación de archivo cargado

Teja plastica_PROM	47.78500	47.92	47.955	47.98000
Techo zinc_PROM	43.66333	43.65	43.670	43.65667
	X454	X455	X456	X457
Teja plastica_PROM	48.16500	48.25	48.30500	48.30500
Techo zinc_PROM	43.64333	43.62	43.67333	43.73333
	X458	X459	X460	X461
Teja plastica_PROM	48.37000	48.42000	48.48000	48.54
Techo zinc_PROM	43.79333	43.86667	43.97333	44.01
	X462	X463	X464	
Teja plastica_PROM	48.55000	48.55000	48.62000	
Techo zinc PROM	44.07667	44.10333	44.17333	

Fuente: IGAC (2024)

4.3.2 PROCESAMIENTO DE LIBRERÍAS

El siguiente paso es convertir los datos a una matriz dataframe y organizar los datos de algunas columnas empleando las siguientes líneas de código.

Ejecute el siguiente código en la consola

```
# convertir el DataFrame a Matrix
spectral_data_matrix <- as.matrix(firmas)
# Eliminar nombres - columnas
colnames(spectral_data_matrix) <- NULL
```

Se crean aparte las longitudes de onda debido al formato de Liberia espectral dentro del formato HSDAL (Imagen 15). Esta librería también tiene una funcionalidad de catálogo, de metadato y de información atributiva adicional.

Asocie esa información de metadato para hacer otro tipo de análisis.

Imagen 15. Formato de librería espectral en HSDAL. Fuente: IGAC (2024)

```
[1] "X450" "X451" "X452" "X453" "X454" "X455" "X456"
[8] "X457" "X458" "X459" "X460" "X461" "X462" "X463"
[15] "X464" "X465" "X466" "X467" "X468" "X469" "X470"
[22] "X471" "X472" "X473" "X474" "X475" "X476" "X477"
[29] "X478" "X479" "X480" "X481" "X482" "X483" "X484"
```

Fuente: IGAC, 2024

Depure la X del formato de las bandas y se convierte esas longitudes de onda a formato numérico, esto a través de ejecutar el código que se muestra a continuación.

```
# Eliminar las "x" en los valores de longitud de onda
for (i in 1:length(wavelengths)){
  str = wavelengths[i]
  len = nchar(str)
  wavelengths[i] = substring(str, 2, len+1)}

# Convertir los valores a formato numérico
wavelengths = as.numeric(wavelengths)
wavelengths
```


En la Imagen 16 se aprecia la salida del dataframe de firmas:

Imagen 16. Formato dataframe con valores numéricos

```
[1] 450 451 452 453 454 455 456 457 458 459 460 461 462
[14] 463 464 465 466 467 468 469 470 471 472 473 474 475
[27] 476 477 478 479 480 481 482 483 484 485 486 487 488
```

Fuente: IGAC, 2024

Para crear la librería ejecute las siguientes líneas de código en la consola. La librería queda almacenada en la variable "lib".

```
# Crear librería
lib <- speclib(spectral_data_matrix,
wavelengths)
lib
str(lib)
plot(lib)
```

Se crea la librería empleando el paquete HSDAL. En la Imagen 17 se muestra la librería resultante.

Imagen 17. Se crea la librería en formato HSDAL. Fuente: IGAC (2024)

	X450	X451	X452	X453	X454	X455	X456	X457
Teja plastica_PROM	47.785000	47.920000	47.955000	47.980000	48.165000	48.250000	48.305000	48.305000
Techo zinc_PROM	43.663333	43.650000	43.670000	43.656667	43.643333	43.620000	43.673333	43.733333
PROM_suelo	2.330000	2.345000	2.355000	2.362500	2.367500	2.380000	2.382500	2.387500
Agua_prom	1.944754	1.917431	1.915109	1.852159	1.844548	1.884864	1.934672	1.966663
Arena_PROM	3.949070	3.977869	3.964590	4.000959	4.042114	4.084172	4.049676	4.05238
AG_D1b	2.790000	2.740000	2.780000	2.800000	2.770000	2.800000	2.880000	2.950000
LR_D9	2.050000	2.090000	2.130000	2.200000	2.260000	2.290000	2.300000	2.340000
MO_D1d	7.690000	7.630000	7.750000	7.770000	7.780000	7.750000	7.730000	7.740000
PR_PRM	2.606667	2.630000	2.650000	2.643333	2.643333	2.633333	2.630000	2.643333
RM_PRM	1.096667	1.063333	1.066667	1.070000	1.066667	1.073333	1.076667	1.093333

Fuente: IGAC, 2024

Se podrá apreciar un resumen de la librería desde la consola. En esta se muestra el total del número de firmas, el número de bandas, el ancho promedio de banda y la resolución espectral. En la imagen 18 se observa que para el caso de ejemplo el valor de ancho promedio de banda es de 1.05 nm, esto pasa porque el rango espectral varía entre 450 nm y 866 nm (Imagen 18).

Ante la posibilidad que en rango espectral cambie por remuestreo se deben conservar el rango de las de firmas fuentes para garantizar la resolución espectral.

Imagen 18. Resolución espectral resultante. Fuente: IGAC (2024)

```
Summary of spectra
-----
Total number of spectra : 10
Number of bands : 396
Mean width of bands : 1.05 nm
Spectral range of data : 450 - 866 nm
```

Fuente: IGAC, 2024

Para visualizar el metadato, se emplea la función `str(lib)`, Esta función muestra el tipo de dato y el tipo de librería espectral. (Imagen 19).

Ejecute la siguiente línea de código para visualizar el metadato

```
str(lib)
```

Imagen 19. Metadato mostrando el tipo de librería espectral.

```
..@ si_data : named list()
..@ dim : num [1:2] 0 0
..@ rasterObject: logi(0)
..@ numericVar : logi(0)
..@ fwhm : num [1:396] 1 1 1 1 1 1 1 1 1 1 ...
..@ continuousdata : logi TRUE
..@ wlnunit : chr "nm"
..@ xlabel : chr "Wavelength"
..@ ylabel : chr "Reflectance"
..@ ID : chr [1:10] "Teja plastica_PROM" "Techo zinc_PROM" "PROM_s
elo" "Agua_prom" ...
..@ wavelength.is.range: logi FALSE
..@ transformation : chr(0)
..@ usagehistory : chr(0)
..@ rastermeta : list()
```

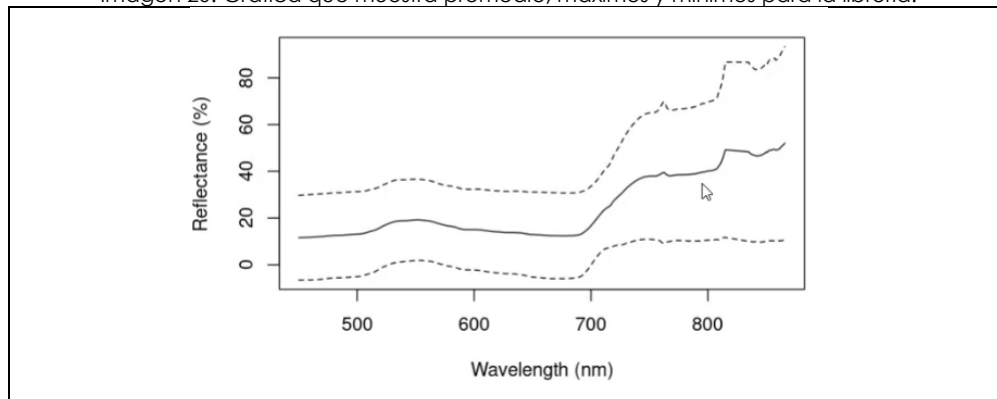
Fuente: IGAC (2024)

Luego de tener la librería, puede visualizar las firmas en una gráfica a través de la función `plot`, esta función grafica el promedio rango mínimo y medio de las firmas. Lo que se observa es el promedio de todas las firmas (Imagen 20).

Ejecute la siguiente línea de código para visualizar el gráfico.

```
plot(lib)
```

Imagen 20. Gráfica que muestra promedio, máximos y mínimos para la librería.



Fuente: IGAC (2024)

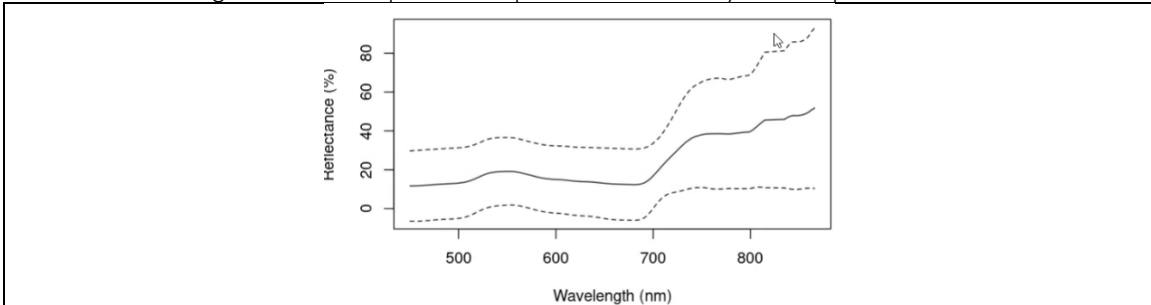
Para suavizar el ruido en la gráfica de las firmas, aplique un filtro de ruido de Hsdar, esto se logra a través del método `method="sgolay"` aplicado a la librería dentro de la función `noiseFiltering`.

Para el ejemplo se aplicó una ventana es 35 nm y se suaviza en función de la ventana se guarda la librería en la variable `lib_s`. El filtro se aplica a toda la firma (Imagen 21).

Ejecute la siguiente línea de código para aplicar el filtro en las firmas y visualizar el grafico posterior.

```
# Suavizar con Savitzky-Golay smoothing filter
#n= frame length
lib_s <- noiseFiltering(lib, method="sgolay", n=35)
plot(lib_s)
```

Imagen 21. Gráfica que muestra promedio, máximos y mínimos para la librería.



Fuente: IGAC (2024)

◦ **Separabilidad espectral**

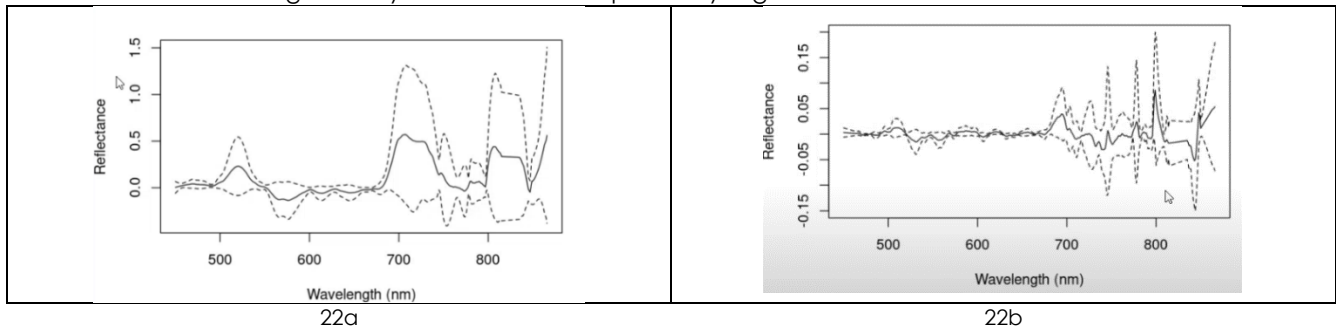
Se puede transformar la librería para resaltar las características de separabilidad. A través de funciones como la segunda y primera derivada se atenúan las características espectrales específicas de absorbancia y reflectancia.

Se espera que al clasificar la firma con la primera derivada se aumente la separación. Es ideal realizar este tipo de análisis para todo el espectro de las firmas trabajadas (Imagen 22).

Ejecute la siguiente línea de código para generar las nuevas variables que calculan la primera y segunda derivada sobre las firmas (`lib_1deriv` y `lib_2deriv`) y posteriormente visualizar a través de los gráficos de estas las características espectrales.

```
# Transformaciones
#Primera y segunda derivadas
lib_1deriv <- derivative.speclib(lib_s, m = 1)
lib_2deriv <- derivative.speclib(lib_s, m = 2)
plot(lib_1deriv)
plot(lib_2deriv)
```

Imagen 22a y 22b. Ploteo de la primera y segunda derivada de las firmas.



Fuente: IGAC (2024)

La remoción del continuo es una normalización para resaltar la identificación de características de absorción de la luz incidente. Las gráficas corresponden a la identificación de los espectros de absorción y diferentes métodos de remoción que buscan normalizar las firmas (Imagen 23).

Al normalizar en función de las características de absorción se regulariza un poco más la firma y se resaltan los aspectos que generan mayor diferenciación de las cubiertas.

Ejecute el siguiente código para realizar la remoción del continuo sobre el conjunto de muestras:

```
#Continuum removal-remoción del continuo
#convex hull-casco convexo:
ch_cline <- transformSpeclib(lib_s, method = "ch", out = "raw")
ch_bd <- transformSpeclib(lib_s, method = "ch", out = "bd")

#segmented hull-casco segmentado:

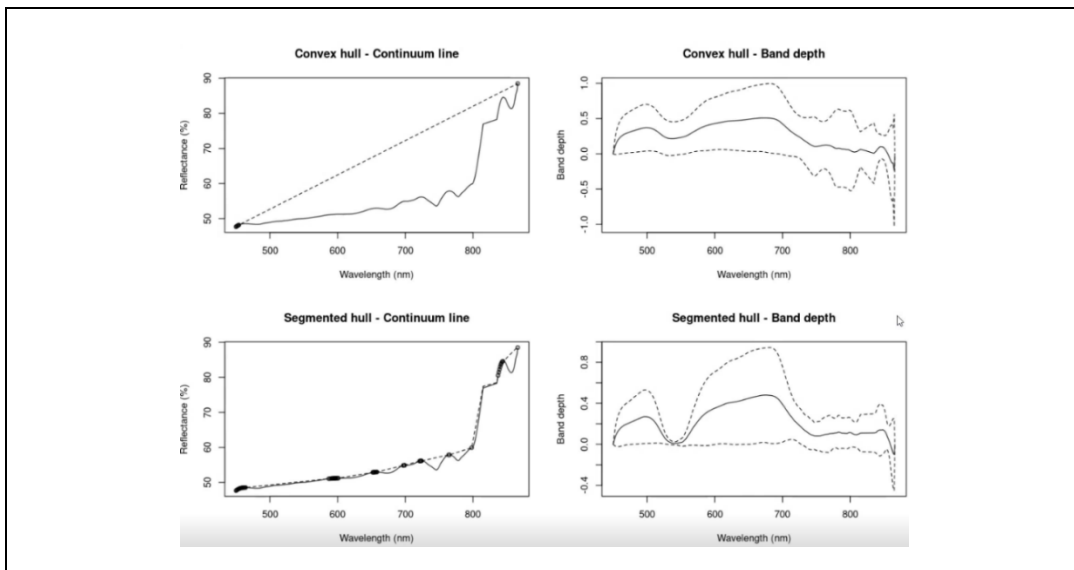
sh_cline <- transformSpeclib(lib_s, method = "sh", out = "raw")

sh_bd <- transformSpeclib(lib_s, method = "sh", out = "bd")

# Plot remoción del continuo:

par(mfrow = c(2,2))
plot(ch_cline, ispec = 1, numeratepoints = FALSE, main = "Convex
hull - Continuum line")
plot(ch_bd, ispec = 1, main = "Convex hull - Band depth")
plot(sh_cline, ispec = 1, numeratepoints = FALSE, main =
"Segmented hull - Continuum line")
plot(sh_bd, ispec = 1, main = "Segmented hull - Band depth")
```

Imagen 23. Normalización de la firma por Remoción de continuo.



Fuente: IGAC (2024)

Cuando se realizan cambios de formato o tipos de tabla por remuestreo se cambian los nombres de las firmas por lo que deben recuperarlos y para ello ejecute el siguiente código:

```
#Otras operaciones
#Asignar identificadores (ID): nombres
ids <- idSpeclib(lib) #Extraer nombres de la librería
creada

idSpeclib(lib) <- as.character(ids) #Asignarlos como ids

ids
```

Imagen 24. Vista del listado de los encabezados de cada firma espectral al ejecutar la función ids.

```
> ids
[1] "Teja plastica_PROM" "Techo zinc_PROM" "PROM_suelo"
[4] "Agua_prom" "Arena_PROM" "AG_D1b"
[7] "LR_D9" "MO_D1d" "PR_PRM"
[10] "RM_PRM"
```

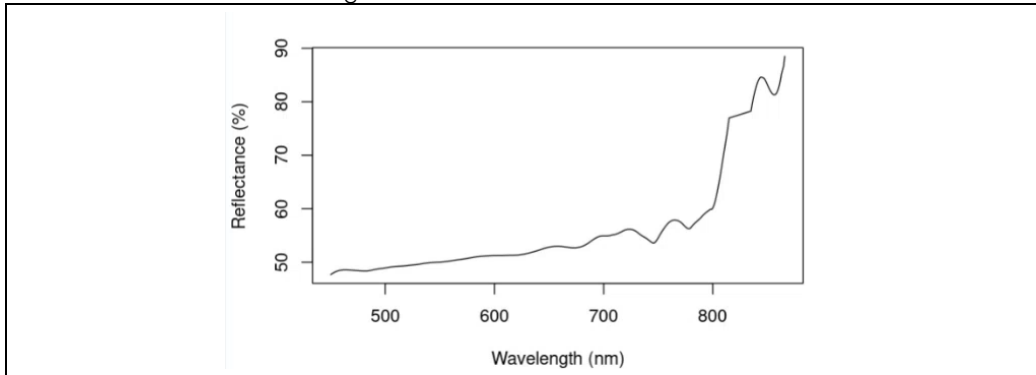
Fuente: IGAC, 2024

Podría ser de interés como extraer una firma espectral de la librería. Seleccione en función del índice (la forma en la que haya nombrado cada firma, en este caso "Teja plástica PROM") una firma específica de la librería que se suavizó.

Ejecute el siguiente código para realizar la extracción de la firma "teja plástica". Verá reflejada la gráfica de la firma de forma individual.

```
# Subset: extracción de firmas en una nueva librería
Teja_plastica <- subset(lib_s, ids == "Teja plastica_PROM")
Teja_plastica
plot(Teja_plastica)
```

Imagen 24. Firma seleccionada del análisis



Fuente: IGAC (2024)

◦ **Extraer firmas específicas**

Puede extraer los datos y generar gráficas para conjuntos de firmas específicos, según requiera. Para el ejemplo trabajado se seleccionaron tres tipos de firmas "Teja plástica", "Techo zinc" y "Suelo", se almacenaron en la variable sub_lib y se graficaron.

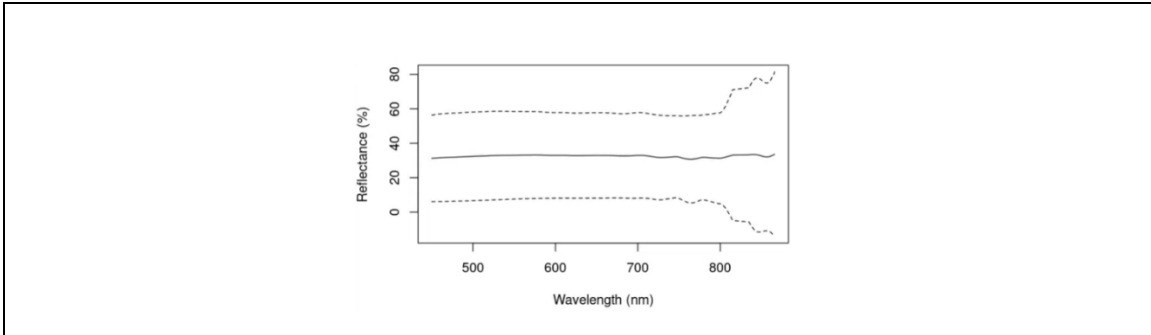
Ejecute la siguiente línea de código para seleccionar y graficar las firmas "Teja plástica", "Techo zinc" y "Suelo".

```
# Extraer firmas específicas

sub_lib <- subset(lib_s, (ids == "Teja plastica_PROM" | ids == "Techo
zinc_PROM" | ids == "PROM_suelo"))
sub_lib

plot(sub_lib)
```

Imagen 25 Extracción de datos para múltiples firmas.



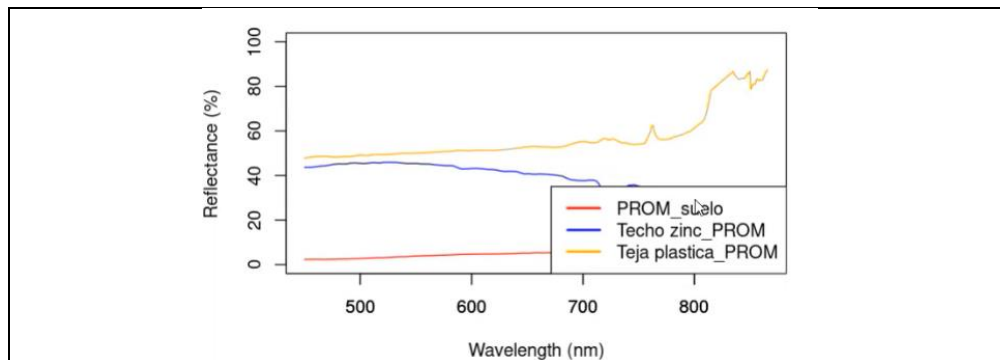
Fuente: IGAC, 2024.

Ejecute las siguientes líneas de código para observar todas las firmas en un mismo gráfico con diferentes colores y leyenda (Imagen 26).

```
#Visualización

plot(lib, FUN = 3, col = "red", ylim = c(0,100))
plot(lib, FUN = 2, col = "blue", new = FALSE)
plot(lib, FUN = 1, col = "orange", new = FALSE)
axis(side=1, at=seq(100, 1800, by=100))
legend(x = "bottomright", #Posición
       legend = c("PROM_suelo", "Techo zinc_PROM", "Teja
plastica_PROM"),
#Leyenda
       #lty = c(1, 2), #Tipo de línea
       col = c("red", "blue", "orange"), #Color de línea
       lwd = 2) #Ancho de línea
```

Imagen 26 Representación adecuada de las firmas en una misma gráfica.



Fuente: IGAC, 2024

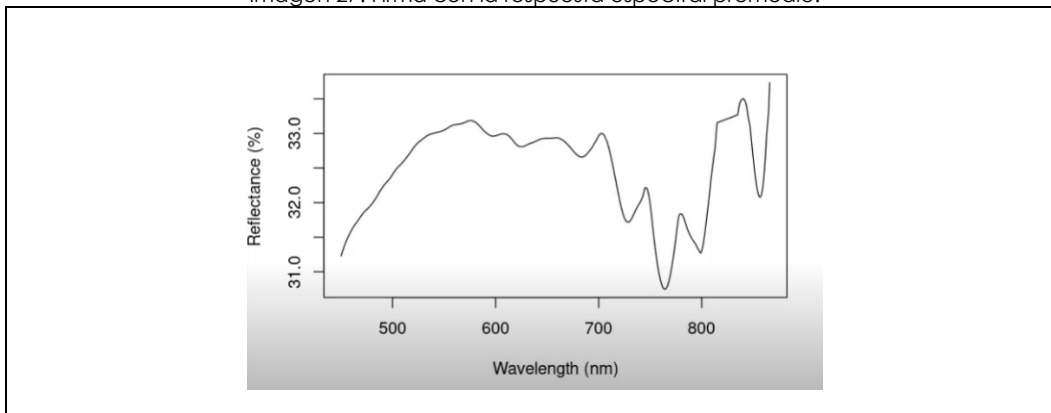
◦ **Estadística sobre las firmas**

Se calcula el promedio, mediana y desviación estándar. Estas operaciones son fundamentales para generar los endmembers (Imagen 27). Normalmente estos datos se trabajan mejor a la mediana porque está más ajustado al ruido en la medición

Ejecute la siguiente línea de código para genera la firma promedio y ver su gráfica.

```
# Calcular firma promedio-mediana
firma_promedio <- apply(sub_lib, FUN = mean)
plot(firma_promedio)
```

Imagen 27. Firma con la respuesta espectral promedio.



Fuente: IGAC, 2024

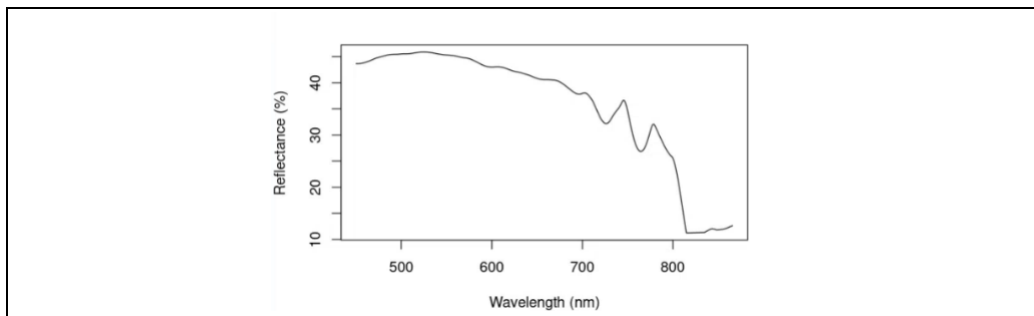
Ejecute la siguiente línea de código para genera la media de las firmas y ver su gráfica.

```
# Calcular firma MEDIA
firma_med <- apply(sub_lib, FUN = median)
plot(firma_med)
```

También se puede calcular la desviación estándar de los datos a lo largo del espectro ejecutando la siguiente línea de código.

```
# Calcular firma desviación estándar
firma_des_est <- apply(lib_s, FUN = sd)
plot(firma_des_est)
```


Imagen 28. Representación de la desviación estándar de la librería espectral.



Fuente: IGAC, 2024

◦ **Análisis jerárquico: Clustering**

En el análisis de firmas espectrales se requiere ilustrar como se modela una firma espectral, que elementos tiene en común o en qué se distingue de otras firmas. Una alternativa de análisis es el análisis jerárquico, a través de técnicas de agrupamiento jerárquico (en este caso clustering ascendente) a partir de la distancia entre las observaciones se generan agrupaciones que nacen de una raíz y se dividen en ramas, permitiendo visualizar de forma más clara las tendencias de separación o similitud entre las firmas.

En esta sección se busca ejecutar los algoritmos de análisis jerárquico utilizando el método de clúster "ward.D2" sobre una matriz de distancias. Para esto se requiere de ejecutar las siguientes líneas de código:

Instale y llame las librerías necesarias para el análisis (tidyverse, cluster, factoextra, readxl, dendextend). Para esto ejecute el siguiente código.

```
#Análisis jerárquico - Ward, dendrograma

install.packages("tidyverse")
install.packages("cluster")
install.packages("factoextra")
install.packages("readxl")
install.packages("dendextend")

library(tidyverse) # data manipulation
library(cluster) # clustering algorithms
library(factoextra) # clustering visualization
library(dendextend) # for comparing two dendrograms
library(readxl)
```

Cargue el archivo de firmas generado en pasos anteriores, organice la tabla que dispone el dataset de firmas y escale los datos (normalización) para el análisis. Ejecute la siguiente línea de código para realizar el paso descrito anteriormente.

```
# Cargar archivo firmas
df <- read.csv("Firmas_TRB3.csv")
# Organizar tabla
df2 <- df[, -1]
```

```
rownames(df2) <- df[,1]
df<-df2
# Escalar tabla
df <- scale(df)
head(df)
```

Genere la matriz de distancias entre las observaciones con la función `dist` y aplicando el método `euclidean`

```
# Matriz de disimilitud
d <- dist(df, method = "euclidean")
```

A continuación, se construirá la función para el análisis jerárquico. Se seleccionó para este análisis el método “ward.D2” por su amplio uso en la bibliografía dada su capacidad en la diferenciación de grupos, útil para el análisis de firmas de elementos similares.

Ejecute el siguiente código para generar la variable `hc5`, que guardará el clúster de las firmas generado por el método “ward.D2”, esto dividirá el conjunto de distancias de acuerdo con el método ejecutado. Posteriormente puede “dividir” el dendrograma en conglomerados deseados, para este caso se harán cuatro (4) grupos por lo que se aplica un $K = 4$. Posteriormente ejecutando `table(sub_grp)` podrá ver el total de firmas en cada clúster y con `plot()` podrá graficar el dendrograma resultante.

```
#Dendrogramas
# Ward
hc5 <- hclust(d, method = "ward.D2" )
# Dividir en cuatro grupos
sub_grp <- cutree(hc5, k = 4)
# Número de firmas por cluster
table(sub_grp)
# Visualizar dendrograma
plot(hc5, cex = 1, main = "Dendrograma de firmas")
rect.hclust(hc5, k = 4, border = 2:5)
```

o **Distancia espectral Jeffries Matusita**

Este es un método de análisis que permite mostrar la separabilidad entre firmas. La distancia de Jeffries-Matusita calcula la separabilidad de un par de distribuciones de probabilidad, esta se mueve en el rango de 0 a 2, donde cero (0) trata de firmas idénticas y dos (2) de firmas diferentes. Esto es útil en el campo del análisis de firmas ya que nos permite evidenciar esa separabilidad espectral entre una firma u otra.

La distancia Jeffries-Matusita J_{xy} se calcula como (Richards y Jia, 2006):

$$J_{xy} = 2(1 - e^{-B})$$

$$B = \frac{1}{8}(x - y)^t \left(\frac{\Sigma_x + \Sigma_y}{2} \right)^{-1} (x - y) + \frac{1}{2} \ln \left(\frac{|\frac{\Sigma_x + \Sigma_y}{2}|}{|\Sigma_x|^{\frac{1}{2}} |\Sigma_y|^{\frac{1}{2}}} \right)$$

Donde:

- X:** Primer vector de firma espectral
- Y:** Segundo vector de firma espectral
- Σ_x es la matriz de covarianza de la muestra x
- Σ_y es la matriz de covarianza de la muestra y

Para esto, se requiere de ejecutar el siguiente código para trasponer la tabla que guarda el conjunto de firmas en otra variable, en este caso `firmas_t`.

```
# Distancia espectral Jeffries Matusita
# Transponer tabla
firmas<-read.csv("Firmas_TRB3.csv", sep=',', row.names=1)
firmas_t <- t(firmas)
```

Luego de esto, instale y llame la librería proxy ejecutando lo siguiente:

```
install.packages("proxy")
library(proxy)
```

Ejecute el siguiente código para definir la función de Jeffries-Matusita.

```
#DEFINICION FUNCION JM
jm.dist <- function ( Vector.1 , Vector.2 ) {
#Adaptado de https://stats.stackexchange.com/questions/78849/measure-for-separability
Matrix.1 <- as.matrix (Vector.1)
Matrix.2 <- as.matrix (Vector.2)
mean.Matrix.1 <- mean ( Matrix.1 )
mean.Matrix.2 <- mean ( Matrix.2 )
mean.difference <- mean.Matrix.1 - mean.Matrix.2
cv.Matrix.1 <- cov ( Matrix.1 )
cv.Matrix.2 <- cov ( Matrix.2 )
p <- ( cv.Matrix.1 + cv.Matrix.2 ) / 2
# Bhattacharyya index: similitud entre dos distribuciones de probabilidad
bh.distance <- 0.125 *t ( mean.difference ) * p^ ( -1 ) * mean.difference +
0.5 * log (det ( p ) / sqrt (det ( cv.Matrix.1 ) * det ( cv.Matrix.2 )))
# Jeffries-Matusita
# Valores entre 0 y 2.0
```

```

jm.distance <- 2 * ( 1 - exp ( -bh.distance ) )
# JM también se encuentra como:
# jm.distance <- 1000 * sqrt ( 2 * ( 1 - exp ( -bh.distance ) ) )
# La segunda formula arroja valores entre 0 y 1414.0
return(jm.distance)
}
dist_jm <- dist(firmas_t,method=jm.dist,by_rows=FALSE)

```

Visualice el resultado de la función instalando las librerías reshape2 y graficando el mapa de calor obtenido del método de J-M.

```

# Visualizar matrix
install.packages("reshape2")
library(reshape2)
mx <- as.matrix(dist_jm)
dist_df <- melt(mx)
ggplot(dist_df, aes(x = Var2, y = Var1, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "red") + # Choose color gradient
  labs(x = "Columns", y = "Rows", title = "Heatmap of Distance Matrix")

```

Por último, guarde el resultado en un archivo csv al ejecutar la siguiente línea de código.

```
write.csv(mx, "mx.csv")
```

Nota:

Recuerde consultar la tabla N° 1 CÓDIGO EN R PROCESAMIENTO, en esta se encuentra condensado las líneas de código descritas anteriormente para que pueda consultarlas y ejecutarlas en continuo. Se recomienda ejecutarlas una por una para evitar errores de ejecución.

Tabla N°1. CÓDIGO EN R PROCESAMIENTO

```

install.packages("devtools")
require("devtools")
install_version("hsdar", "1.0.4")
install_version("rgdal", "1.6.7")
library(hsdar)
library(rgdal)
#Cargar archivo de firmas (cloud): Upload-File to upload
# La forma en que se llama el archivo puede cambiar la tabla
firmas<-read.csv("Firmas_TRB3.csv", sep=',', row.names=1)
# convertir el DataFrame a Matrix
spectral_data_matrix <- as.matrix(firmas)
# Eliminar nombres - columnas

```

Tabla N°1. CÓDIGO EN R PROCESAMIENTO

```

colnames(spectral_data_matrix) <- NULL
#crear una matriz aparte con los valores de longitud de onda
wavelengths = names(firmas)
wavelengths
# Eliminar las "x" en los valores de longitud de onda
for (i in 1:length(wavelengths)){
  str = wavelengths[i]
  len = nchar(str)
  wavelengths[i] = substring(str, 2, len+1)
}
# Convertir los valores a formato número
wavelengths = as.numeric(wavelengths)
wavelengths
# Crear libreria
lib <- speclib(spectral_data_matrix, wavelengths)
lib
str(lib)
plot(lib)
# Suavizar con Savitzky-Golay smoothing filter
#n= frame length
lib_s <- noiseFiltering(lib, method="sgolay", n=35)
plot(lib_s)
# Transformaciones
#Primera y segunda derivadas
lib_1deriv <- derivative.speclib(lib_s, m = 1)
lib_2deriv <- derivative.speclib(lib_s, m = 2)
plot(lib_1deriv)
plot(lib_2deriv)
#Continuum removal-remoción del continuo
#convex hull-casco convexo:
ch_cline <- transformSpeclib(lib_s, method = "ch", out = "raw")
ch_bd <- transformSpeclib(lib_s, method = "ch", out = "bd")
#segmented hull-casco segmentado:
sh_cline <- transformSpeclib(lib_s, method = "sh", out = "raw")
sh_bd <- transformSpeclib(lib_s, method = "sh", out = "bd")
# Plot remoción del continuo:
par(mfrow = c(2,2))
plot(ch_cline, ispec = 1, numeratepoints = FALSE, main = "Convex hull - Continuum
line")
plot(ch_bd, ispec = 1, main = "Convex hull - Band depth")
plot(sh_cline, ispec = 1, numeratepoints = FALSE, main = "Segmented hull - Continuum
line")
plot(sh_bd, ispec = 1, main = "Segmented hull - Band depth")
#Otras operaciones
#Asignar identificadores (ID): nombres
ids <- idSpeclib(lib) #Extraer nombres de la librería creada
idSpeclib(lib) <- as.character(ids) #Asignarlos como ids

```

Tabla N°1. CÓDIGO EN R PROCESAMIENTO

```

ids
# Subset: extracción de firmas en un nueva librería
Teja_plastica <- subset(lib_s, ids == "Teja plastica_PROM")
Teja_plastica
plot(Teja_plastica)
# Extraer firmas especificas
sub_lib <- subset(lib_s, (ids == "Teja plastica_PROM" | ids == "Techo zinc_PROM" | ids
== "PROM_suelo"))
sub_lib
plot(sub_lib)
#Visualización
plot(lib, FUN = 3, col = "red", ylim = c(0,100))
plot(lib, FUN = 2, col = "blue", new = FALSE)
plot(lib, FUN = 1, col = "orange", new = FALSE)
axis(side=1, at=seq(100, 1800, by=100))
legend(x = "bottomright", #Posición
      legend = c("PROM_suelo", "Techo zinc_PROM", "Teja plastica_PROM"), #Leyenda
      #lty = c(1, 2), #Tipo de línea
      col = c("red", "blue", "orange"), #Color de línea
      lwd = 2) #Ancho de línea
# Calcular firma promedio-mediana
firma_promedio <- apply(sub_lib, FUN = mean)
plot(firma_promedio)
firma_med <- apply(sub_lib, FUN = median)
plot(firma_med)
# Calcular firma desviación estándar
firma_des_est <- apply(lib_s, FUN = sd)
plot(firma_des_est)
# Remuestreo espectral: firma espectral a imagen satelital/aerotransportada
# Definir las bandas de la imagen (target) ejp: PLANET SCOPE
center <- c(490, 565, 665, 865)
fwhm <- 40
#wl:rango de la librería espectral
wl <- c(450:866)
# Crear la respuesta espectral con la función de densidad gaussiana
response <- speclib(t(sapply(center, function(center, wl, fwhm)
{
a <- dnorm(wl, mean = center, sd = fwhm/2)
a <- (a-min(a))/(max(a) - min(a))
return(a)
}, wl, fwhm)), wl)
# Visualizar librería-función remuestreo
for (i in 1:4)
  plot(response[i,], new = i == 1, col = c("blue", "green", "red", "orange")[i])
# Implementar el remuestreo
firmas_RES <- spectralResampling(lib, response_function = response)

```


Tabla N°1. CÓDIGO EN R PROCESAMIENTO

```

plot(firmas_RES)

#Exportar a csv - firmas suavizadas
#Se convierten primero en data.frame
firmas_Suav <- as.data.frame(lib_s)
#Crear campo de nombres (se pierde en la conversión de formatos)
firmas_Suav <- cbind(firmas_Suav, nombre = ids) #ids: creado en la asignación de ID
write.csv(firmas_Suav, "firmas_Suav.csv")
#####
#Análisis jerárquico - Ward, dendrograma

install.packages("tidyverse")
install.packages("cluster")
install.packages("factoextra")
install.packages("readxl")
install.packages("dendextend")

library(tidyverse) # data manipulation
library(cluster) # clustering algorithms
library(factoextra) # clustering visualization
library(dendextend) # for comparing two dendrograms
library(readxl)

# Cargar archivo firmas
df <- read.csv("Firmas_TRB3.csv")
# Organizar tabla
df2 <- df[,-1]
rownames(df2) <- df[,1]
df <- df2

# Escalar tabla
df <- scale(df)
head(df)

# Matriz de disimilitud
d <- dist(df, method = "euclidean")
#Dendrogramas
# Ward
hc5 <- hclust(d, method = "ward.D2" )
# Dividir en cuatro grupos
sub_grp <- cutree(hc5, k = 4)
# Número de firmas por cluster
table(sub_grp)
# Visualizar dendrograma
plot(hc5, cex = 1, main = "Dendrograma de firmas")
rect.hclust(hc5, k = 4, border = 2:5)
# Distancia espectral Jeffries Matusita
# Transponer tabla
firmas <- read.csv("Firmas_TRB3.csv", sep=',', row.names=1)
firmas_t <- t(firmas)
install.packages("proxy")

```

Tabla N°1. CÓDIGO EN R PROCESAMIENTO

```

library(proxy)
#DEFINICION FUNCION JM
jm.dist <- function ( Vector.1 , Vector.2 ) {
  #Adaptado de https://stats.stackexchange.com/questions/78849/measure-for-separability
  Matrix.1 <- as.matrix (Vector.1)
  Matrix.2 <- as.matrix (Vector.2)
  mean.Matrix.1 <- mean ( Matrix.1 )
  mean.Matrix.2 <- mean ( Matrix.2 )
  mean.difference <- mean.Matrix.1 - mean.Matrix.2
  cv.Matrix.1 <- cov ( Matrix.1 )
  cv.Matrix.2 <- cov ( Matrix.2 )
  p <- ( cv.Matrix.1 + cv.Matrix.2 ) / 2
  # Bhattacharyya index: similitud entre dos distribuciones de probabilidad
  bh.distance <- 0.125 *t ( mean.difference ) * p^ ( -1 ) * mean.difference +
  0.5 * log ( det ( p ) / sqrt ( det ( cv.Matrix.1 ) * det ( cv.Matrix.2 )))
  # Jeffries-Matusita
  # Valores entre 0 y 2.0
  jm.distance <- 2 * ( 1 - exp ( -bh.distance ) )
  # JM también se encuentra como:
  # jm.distance <- 1000 * sqrt ( 2 * ( 1 - exp ( -bh.distance ) ) )
  # La segunda formula arroja valores entre 0 y 1414.0
  return(jm.distance)
}

dist_jm <- dist(firmas_t,method=jm.dist,by_rows=FALSE)
# Visualizar matrix
install.packages("reshape2")
library(reshape2)
mx <- as.matrix(dist_jm)
dist_df <- melt(mx)
ggplot(dist_df, aes(x = Var2, y = Var1, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "red") + # Choose color gradient
  labs(x = "Columns", y = "Rows", title = "Heatmap of Distance Matrix")
write.csv(mx, "mx.csv")
#EJERCICIO
#Seleccione un conjunto de firmas de la práctica de toma de firmas, realice los
procedimientos vistos hasta el momento, incluya descartar firmas específicas de la
librería que cree. Compare visualmente qué transformación de librería presenta una mayor
separabilidad espectral. Remuestre una librería de interés a diferentes sensores de
observación de la tierra (aero o satelital).

```

5. BIBLIOGRAFIA

- Ali, Z., & Bhaskar, S. (2016). Basic statistical tools in research and data analysis. *Indian Journal of Anaesthesia*. doi:10.4103/0019-5049.190623
- A. H. Zewail. "Femtochemistry: Atomic-Scale Dynamics of the Chemical Bond." *Journal of Physical Chemistry A* 104.24 (2000): 5660-5694.
- A. SEKHAR C S. (2024). Jeffries-Matusita Distance (<https://www.mathworks.com/matlabcentral/fileexchange/159828-jeffries-matusita-distance>), MATLAB Central File Exchange.
- C. Elachi. "Introduction to the Physics and Techniques of Remote Sensing." Wiley, 1987.
- D. S. Yadav, et al. "Methodology for Assessing the Environmental Impact of Development Projects." *International Journal of Sustainable Development & World Ecology* 17.2 (2010): 154-160

G. Wyszecki, W. S. Stiles. "Color Science: Concepts and Methods, Quantitative Data and Formulae." John Wiley & Sons, 1982

J. H. Moxnes. "Safety Management: A Comprehensive Approach to Developing a Sustainable System." CRC Press, 2017.

R. D. Brown "Introduction to Spectroradiometry." Radiation Measurements 25.5-6 (1995): 641-651.

Richards, J. A., & Jia, X. (2006). Remote sensing digital image analysis: An introduction. Berlin: Springer Verlag.

T. F. Hockley, et al. "Verification of Numerical Models in Fluid Mechanics." Journal of Fluid Mechanics 103 (1981): 441-463.

S. J. Wang. "Introduction to Remote Sensing." CRC Press, 2012

W. L. Barnes, et al. "Surface Plasmon Subwavelength Optics." Nature 424.6950 (2003): 824-830.

6. CONTROL DE CAMBIOS

FECHA	CAMBIO	VERSIÓN
14/08/2024	<ul style="list-style-type: none"> Se adopta como versión 1 por corresponder a la creación del documento. Emisión Inicial Oficial. Hace parte del proceso de Gestión del Conocimiento Aplicado, del subproceso de Investigación e Innovación Aplicada – Laboratorio de Espectroradiometría. Se crea el Instructivo "Procesamiento y Análisis de Firmas espectrales", código IN-IIA-PC02-03, versión 1. Se encuentra asociado al procedimiento de Captura y Procesamiento de Firmas Espectrales. 	1

ELABORÓ Y/O ACTUALIZÓ	REVISÓ TÉCNICAMENTE	REVISÓ METODOLÓGICAMENTE	APROBÓ
<p>Nombre: Hernando Antonio Hernández Hamón.</p> <p>Cargo: Contratista. Dirección de Investigación y Prospectiva.</p>	<p>Nombre: Paula Valeria Suarez Hurtado.</p> <p>Cargo: Contratista. Dirección de Investigación y Prospectiva.</p> <p>Nombre: Dayana Patricia Beltrán Fonseca.</p> <p>Cargo: Profesional Especializado. Dirección de Investigación y Prospectiva.</p> <p>Nombre: Nelson Andrés Nieto Valencia.</p> <p>Cargo: Contratista. Dirección de Investigación y Prospectiva.</p>	<p>Nombre: Martha Patricia Ramírez Suarez.</p> <p>Cargo: Profesional Especializado. Dirección de Investigación y Prospectiva.</p> <p>Nombre: Cesar Augusto Buitrago López.</p> <p>Cargo: Contratista. Oficina Asesora de Planeación.</p>	<p>Nombre: Alexander Páez Lancheros.</p> <p>Cargo: Director de Investigación y Prospectiva (E). Dirección de Investigación y Prospectiva.</p>