

Título

Subtítulo

Subtítulo

Darihana Linares Laureano *Estudiante de Licenciatura en Geografía mención recursos naturales y ecoturismo, Universidad Autónoma de Santo Domingo (UASD)*

Resumen del manuscrito

Keywords: palabra clave 1, palabra clave 2

1 Introducción

Desde mediados del siglo XVII es posible visualizar el interés del hombre a estudiar la flora, la fauna y el medio en el que están en conjunto con las interacciones que se producen entre ellos, pero no es hasta mediados del siglo XIX cuando se introduce el término Ecología y su definición, que se empieza a englobar en este tipo de estudios en una categoría (De la Llata Loyola, 2003). A partir de este punto se reconocieron distintos campos de estudios y se implementaron nuevos métodos de análisis, entre los cuales destacan la ecología numérica y métodos como el análisis multivariático. Según P. Legendre & Legendre (2012), la ecología numérica no es más que una de las disciplinas de la ecología cuantitativa, la cual a la vez es una de las divisiones de la ecología matemática.

La ecología numérica se concentra en el estudio y análisis de conjuntos de datos ecológicos, a fin de poder detallar y comprender la configuración de los conjuntos de datos, combinando diversas perspectivas numéricas y disciplinas, procedentes de la Geografía, las matemáticas físicas, taxonomía numérica, parámetros estadísticos y otros más (P. Legendre & Legendre, 2012). El análisis de conjuntos de datos ecológicos, especialmente de la flora, resulta importante tanto para la ciencia, como para la economía o el sector de salud, por eso se han establecido distintas parcelas permanentes de medición y monitoreo forestal donde se colectan datos sobre la diversidad forestal, su estructura, el crecimiento y su productividad (Pineda, 2014). En el estudio de las plantas a través de la Ecología Numérica se usan diversas técnicas que permiten obtener información sobre el rango de asociación, agrupamiento, ordenamiento, diversidad, autocorrelación, etcétera, diferentes herramientas para usar las técnicas.

La medición de la asociación es un coeficiente que sirve para medir y asociar los datos de variables cualitativos y cuantitativos. La medición de estas variables se puede hacer por dos modos el *Q* y el *R*, el primero consiste en hacer una comparación de un dúo de objetos y el segundo consiste en realizar una descripción de un par de objetos y luego compararlos. Para el modo *Q* se miden la asociación según la similitud o disimilitud de un par de objetos. Mientras que para el modo *R* se mide el grado de dependencia existente entre las variables, entre los cuales se puede mencionar la covarianza o el coeficiente de correlación (Borcard, Gillet, Legendre, & others, 2011).

En cambio, el análisis de agrupamiento o clúster análisis es una técnica que consiste en separar un conjunto de datos y luego estructurarlos, sin dejar uno fuera de lugar, como subconjuntos con distintas categorías o jerarquías de acuerdo a sus características. La finalidad de hacer un agrupamiento es identificar los pequeños grupos dispersos en un espacio discreto pero constante,

este agrupamiento divide en conjunto de objetos a estudiar, por lo que es necesario e importante que cada objeto agrupado en otros subgrupos no se encuentre en otros (Borcard et al., 2011).

En el caso del análisis de ordenamiento, son técnicas que consisten en simplificar la magnitud de los datos. Todas estas técnicas muestran las predisposiciones esenciales de variabilidad de cada dato que se encuentran en un campo de dimensiones simplificadas, organizando los ejes con rangos decreciente de varianza explicada en cada uno de los ejes sucesivo, de forma convencional. Estos tipos de análisis pueden ser tanto no restringido o simple, como restringido o canónico. En donde para el primero, las tendencias o predisposiciones del grupo que interesa no está restringida por otro grupo. Entre sus técnicas principales de análisis están los de componentes principales (PCA) basado en un vector propio y se utiliza en datos cuantitativos sin tratamiento preservando la distancia euclídea, los de correspondencia (CA) que se usa en datos frecuentes con dimensiones uniformes y positivos, y los de coordenadas principales (PCoA) que se concentra en organizar las matrices de disimilaridad, usualmente, con el modo Q en vez de tablas de sitio por variables (Borcard et al., 2011).

Mientras que, para el segundo análisis de ordenamiento, restringido o canónico, a diferencia de la simple, es una técnica que relaciona dos o más conjunto de datos en el proceso de organización u ordenación. Algunas de sus técnicas principales son análisis de redundancia (RDA) que consiste en la combinación de la regresión y PCA que funciona como una extensión de diversos análisis que muestran la respuesta multivariante de datos, y el análisis de correspondencia canónica (CCA) que funciona como un aproximado de una regresión Gaussiana multivariante, además, este se caracteriza por organizar las especies en todos los ejes canónicos acorde a su configuración ecológica óptima (Borcard et al., 2011).

En cuanto a la diversidad, según Borcard et al. (2011) y Magurran (1988), esto alude a la variedad y cantidad de especies en un espacio determinado, esta variedad también se produce a nivel de comunidad. La diversidad puede ir desde la diversidad local, hasta heterogeneidad espacial de esta diversidad. La diversidad de especies considerada como un número único puede ser medida por la riqueza o la rarefacción de especies usando la notación q o por la presencia o ausencia de los datos de esta. Según Whittaker (1972), el entendimiento de la diversidad y los cambios que conllevan, asociados a la configuración del relieve los componentes *alpha*, *beta* y *gamma* serían de mucha utilidad. Donde la primera se refiere a la riqueza de las especies existentes en una comunidad determinada, considerada homogénea. La segunda, se trata del rango de cambios que se producen en la estructura de especies que están en diferentes comunidades en un espacio. Y la última, se refiere a la riqueza de especies de forma conjuntiva que hay en una comunidad y que integra un espacio determinado.

La autocorrelación, según Borcard et al. (2011), forma parte de los análisis espaciales aplicados a datos ecológicos, que se produce por distintos procesos y que mide puntos cercanos para afirmar si estos poseen valores similares o distintos, por lo que la correlación puede ser una correlación positiva o negativa.

La *Myrtaceae* son una familia de plantas de árboles y arbustos bastante numerosas compuestas caracterizadas por ser leñosas. Esta familia de plantas pertenece al orden de los *Myrtales*, teniendo a nivel mundial 129 géneros y aproximadamente 5330 especies (Perez & R, s.f.). De sus características físicas se destacan sus hojas simples y opuestas, tienden a ser perennes por lo que resulta poco común ver individuos caducifolios, sus flores son hermafroditas y comúnmente son de color blanco con simetría radial, según la especie de esta familia los frutos tienen forma de bayas o cápsulas secas. De esta familia el género más conocido es el *Eucalypto* *Eucalyptus* por sus propiedades medicinales y su madera dura. Esta familia está distribuida en todos los continentes, pero predominan en América, África y Oceanía, en climas templados, tropicales y subtropicales. Muchos consideran que la importancia de esta familia está en lo económico, como la producción de fru-

tas para venta de zumos y mermeladas, producción de madera, producción de papel, de carbón, además de ser usadas en la industria farmacéutica, se usa en la cosmetología y en la producción de especias (Almeida, 2019; Britannica, 2016; Lorenzo-Cáceres, s.f.).

El estudio de la biodiversidad de la flora en especial de una especie vegetal es importante, debido a que permite conocer sus características propias, ecosistémicas, su distribución, su capacidad productiva, su potencial de uso y aportes a los humanos como a su ecosistema. Por todo lo anterior, es importante determinar el objetivo de este estudio, el cual se fragmenta así: *a)* identificar si los grupos de mi familia, *Myrtaceae*, se organizan de forma discontinua y acorde a la composición de las especies; *b)* indagar si existe algún tipo de patrón que sea o no sea consistente con alguna variable ambiental o atributo; *c)* determinar cuántas especies indicadoras hay o si hay alguna con preferencia por ciertas condiciones ambientales o atributos; *d)* investigar, en un espacio bidimensional, si hay una tendencia de ordenación visible de las especies de *Myrtaceae*; *e)* conocer si las tendencias de ordenación se asocian con las variables ambientales o atributos; *f)* determinar, de acuerdo a la estimación de riqueza, si mi familia está bien representada, tomando como buena representación un 85%; *g)* investigar si existe una alguna asociación entre la diversidad Alpha y las variables ambientales o atributos y cuales son estas; *h)* conocer si hay algún tipo de contribución local o de alguna especie a la diversidad Beta; *i)* identificar si hay alguna especie que presente algún patrón aglomerado, cuál es y si presenta alguna asociación con las variables ambientales; *j)* y determinar si los modelos de distribución de especies (*SDM*) predicen adecuadamente las ocurrencias de las especies.

2 Área de estudio

El área usada para este estudio se encuentra en *BCI* (Isla de Barrio Colorado), que es una isla perteneciente a Panamá, la cual se formó cuando las aguas del río Chagres fueron represadas para construir el Canal de Panamá. Es considerada como una colina con una superficie de 1,500 hectáreas sobresaliente a 137 m en el lago Gatún, y que está localizada entre los 9° 09' N y 79° 51' W. Esta isla se caracteriza por sus suelos arcillosos con profundidades entre los 50 cm y 1 m, y su clima es común de las tierras bajas tropicales con temperatura promedio anual de 27° C, y variación diurna de 9° C. También tiene unas precipitaciones anuales de 2,600 mm, con su estación lluviosa (mayo a diciembre), y su estación seca para el resto de los meses (Pérez et al., 2005; Sugasti, Eng, & Pinzón, 2018). Según la clasificación de zonas de vida de Holdridge, *BCI* se encuentra ubicada en la zona de vida de bosque húmedo tropical; Claver Farías & others (1984), dice que conforme a la clasificación de climática de Köppen esta tiene un clima tropical húmedo con precipitaciones abundantes; y de acuerdo con el mapa de la ANAM (2000) su vegetación se caracteriza por tener bosques semicaducifolio tropical de tierras bajas (Rodríguez-Castro & Flores, 2021).

El área de estudio, específicamente, es una parcela permanente forestal para su medición y monitoreo de 50 hectáreas, establecida en 1980 por Stephen Hubbell y Robin Foster, en el bosque húmedo tropical de *BCI*. Esta parcela se caracteriza por su forma rectangular con 1000 m de largo y 400 m de ancho, además, esta se ubica en la parte central de una meseta en la isla (Hubbell, Foster, & Condit, 2005; Pérez et al., 2005). Los datos extraídos de las especies forestales de la parcela se obtuvieron a través de múltiples censos desde 1982 hasta 2015 (un total de 8 censos), en los cuales se hicieron identificación, marcado y monitoreo de los tallos leñosos individuales de más de 10 mm de diámetro a altura del pecho. Durante los 35 años se censaron más de 350,000 árboles individuales (Hubbell et al., 2005). Ver figura 1

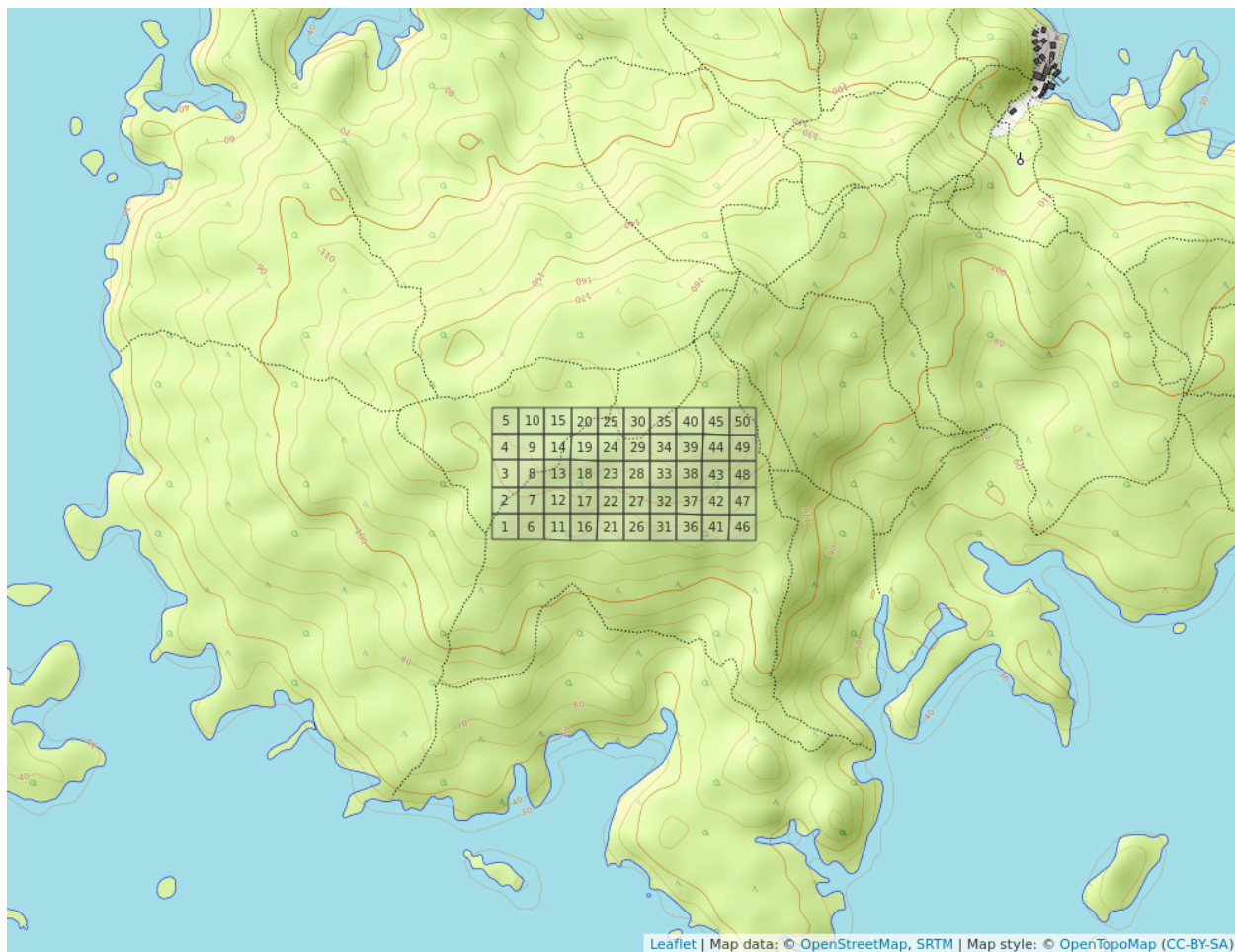


Figure 1: Mapa de la Isla Barro Colorado y la parcela permante.

3 Metodología y materiales

Para el estudio de la biodiversidad de la familia *Myrtaceae* se usaron los datos censales de las 50 parcelas forestales de *BCI* que están presentes en el repositorio *scripts-de-analisis-BCI* (Batlle, 2020), de los cuales fueron tratados y procesados con distintos algoritmos y técnicas de medición, cálculo, análisis e interpretación por medio del entorno de desarrollo integrado y libre, *RStudio* (R Core Team, 2020). En cuanto a los gráficos presentes se obtuvieron por medio *R*, usando paquetes y funciones (ver tabla 2).

Para iniciar los análisis estadísticos de datos con dimensiones diversas, Buzai & Baxendale (2009) y Borcard et al. (2011), recomiendan realizar un análisis exploratorio de datos (EDA), debido a que es una herramienta imprescindible para conseguir una primera intermediación, información genérica de los datos, realizar transformaciones a variables de ser necesarias, y poder conducir más análisis. Se usaron los paquetes *vegan*, *tidyverse*, *sf*, *mapview* y *RColorBrewer* para cargar los datos censales, matriz de comunidad y matriz ambiental, y así poder extraer los datos de la familia *Myrtaceae*, de las variables ambientales y transformarlos en tablas, matrices, e incluso generar gráficos de mapas. También, se usaron las librerías *psych* y *ez*, en conjunto con las anteri-

ores para ver correlaciones de la *Myrtaceae* con distintas variables ambientales con la función `cor`, y generar mapas de variables ambientales por cada cuadro con `ggplot2`.

La medida de los coeficientes de asociación de la familia *Myrtaceae* con los modos *Q* y *R* (Borcard et al., 2011; Salazar, 2000) se realizó mediante funciones pertenecientes a las librerías `vegan`, `adespatial`, `broom`, `tidyverse`, `sf`, `cluster` y `gclus`. En el modo *Q* se mide la abundancia de las especies de *Myrtaceae* con la función `dist.ldc` que calcula la distancia euclídea entre dos sitios y se usa también el método de transformación Hellinger y así obtener el grado de disimilitud entre sitios. También, se calculó la distancia de *Jaccard Dj* (Jaccard, 1908; Salazar, 2000) utilizando la función `vegdist` que convierte la matriz de comunidad en una de presencia/ausencia para calcular la matriz de distancia. En cuanto a la asociación de las variables ambientales, se hizo un nuevo ambiente con la función `env` para las variables transformadas con la función `scale`. De igual manera, se usó la función `env_mix` para asociar las variables ambientales heterogeneidad ambiental, hábitat y quebrada, generando una matriz de disimilitud.

Con el modo *R* de medición de asociación, se mensuró el grado de asociación entre especies haciendo una transformación de la matriz de comunidad usando la estandarización *Chi* con la función `decostand`, luego se calcula la distancia con la función `dist` y se representa la matriz en un mapa de calor con la función `coldiss`. Se calculó la distancia de *Jaccard* (Jaccard, 1908; Salazar, 2000) entre especies usando la matriz de comunidad transpuesta convertida a una de presencia/ausencia con la función `vegdist`. En cuanto a la asociación de variables ambientales numéricas, usando el índice de *rho* de Spearman con los verbos `select`, `mutate` y `matches` de la herramienta `dplyr` y también la función `cor` para correlación.

Los análisis de agrupamiento o clúster análisis aplicados a la familia *Myrtaceae* fueron realizados usando las librerías `magrittr`, `broom`, `tidyverse`, `mapview` y `indicspecies`. Para el estudio se usó el agrupamiento jerárquico (*AJ*) con un enfoque aglomerativo por enlace simple, completo, promedio y el método de *Ward* de varianza mínima (Borcard et al., 2011). Se agruparon pares de objetos según la mayor similaridad (vecino más próximo o mínima distancia), partiendo del agrupamiento por enlace simple, utilizando la matriz de comunidad transformada por medio del método de normalización con la función `decostand` y la distancia euclídea con la función `vegdist`. El agrupamiento jerárquico se hizo con la función `hclust` para método simple y se generó un dendrograma (gráfico) con la función `plot` del resultado de este *AJ*. Para el agrupamiento completo usando la menor similaridad (máxima distancia o vecino más lejano) se empleó la función `hclust` y la matriz de distancia de cuerdas o *chord* (Borcard et al., 2011), con tal de generar un dendrograma con la función `plot`. En cuanto al agrupamiento por enlace promedio (UPGMA, WPGMA, UPGMC, WPGMC), se usó el UPGMA (Romesburg, 2004), para maximizar la correlación entre las distancias cofenética (coeficiente de correlación de Pearson), para esto se usó la función `hclust` y para obtener el gráfico la función `plot`. Finalmente, el agrupamiento por *Ward* (Romero Domínguez, 2021), que define los grupos de una forma donde la suma de cuadrados se minimice en cada uno de ellos usando las funciones `hclust` y `plot`.

De todos los métodos anteriores de agrupamiento, se han seleccionado ideal para la familia *Myrtaceae* se usó la función `map` y así comparar sus valores. Además, para la elección del número de clústeres se usó la función `calcular_anchuras_siluetas` en base a la matriz de comunidad original, la matriz de distancias y objeto de clúster usando UPGMA y *Ward*, luego, se hizo un nuevo dendrograma con la función `reorder.hclust` y un mapa de calor con la función `heatmap`. Finalmente, se hizo una evaluación para ver si el número de clústeres obtenidos por los métodos anteriores son los ideales usando el método *bootstrap* (Efron, 1992; Tibshirani & Efron, 1993), usando el paquete `pvclust` para generar dendogramas con trazos en rectángulos y líneas que dividen el gráfico acorde al número de grupos.

En cuanto al agrupamiento de variables se usaron los grupos obtenidos por UPGMA para

evaluar la homogeneidad por medio de pruebas *t*, basadas en la distribución *t* de *Student* (Hurtado & Silvente, 2012) y la suma de rangos de Wilcoxon (Wilcoxon & Wilcox, 1964), a partir de esto se generaron gráficos de caja y mapas presentando la ubicación por cuadrado de cada grupo en la parcela permanente. Y para la homogeneidad de los grupos Ward se usó el método ANOVA o análisis de varianza (Hurtado & Silvente, 2012) y Kruskal/Wallis (Kruskal & Wallis, 1952).

Cerrando con el clúster análisis, se obtuvieron las especies indicadoras y las especies con preferencias por hábitats; la primera mediante IndVal con las funciones `multipatt` y `strassoc` y su significancia o valor *p* con la función `p.adjust`; las especies con preferencias por hábitats se obtuvieron mediante el coeficiente de correlación biserial puntual (Palmer, Jiménez, & Montaña, 2000) con las funciones `multipatt` y `strassoc`.

Las técnicas de ordenación aplicas a las *Myrtaceae* usaron las librerías `vegan`, `tidyverse`, `sf` y `mapview`. Para la ordenación simple o no restringida se usaron las técnicas PCA, CA y PCoA. Con la función `rda` para escalar las variables, calcular la matriz de correlación y obtener vectores para PCA con datos ambientales, la función `screeplot` para graficar lo anterior y `cleanplot.pca` para el escalamiento; también, se hizo un mapa con puntuaciones de los sitios para tomarlas como coordenadas con las funciones `scores` y `plot`; para datos de comunidad transformada con *hellinger* se usaron las funciones `decostand`, `rda`, `screeplot` y `scores`, también, se evaluaron los datos de comunidad a datos ambientales con la función `envfit`. Para el análisis CA se usó las funciones `cca`, `screeplot` y `par`. Para el análisis PCoA se usaron la función `cmdscale` y una matriz de distancias, también la función `ordiplot`.

En la ordenación canónica o restringida se usaron las técnicas RDA y CCA para presentar las relaciones entre los objetos (entre dos matrices) sin restricción. En la primera técnica, usando la matriz ambiental, se utilizaron la función `decostand`, `rda` para ajustar las variables de respuesta por regresión, haciendo prueba estadística para ver la relación de las variables de la matriz ambiental según el valor de *p*, también se calculó R² insesgado con la función `RsquareAdj`, además de explorar la multicolinealidad entre variables. En cuanto a CCA, se usaron las mismas funciones, se excluyeron especies con abundancias mínimas al 100 por individuo.

En cuanto a la diversidad de la *Myrtaceae*, *alpha* y *beta*, para su cálculo se usaron las librerías `vegan`, `adespatial`, `plyr`, `RColorBrewer`, `tidyverse`, `sf`, `SpadeR`, `iNEXT`, `vegetarian` y `mapview`. Para calcular la diversidad especies, *alpha*, como un número único, se usó la función `alpha_div` y así obtener los índices *alpha*; también se usó la función `pairs` para hacer una matriz de correlación de Pearson con todos los índices, luego con la función `bind_cols` se unen en una matriz estos índices con variables ambientales. Para calcular el rango de abundancia de especies se usó la función `radfit`. Para la rarefacción por sitios se usó la función `specnumber`. Para el cálculo de abundancia por sitio se usó `rowSums`; se generó con una curva de rarefacción con los datos obtenidos anteriormente. De la misma manera, se calculó la riqueza de especies mediante estimaciones y comparaciones y evaluar la completitud de muestra; para esto se usaron los enfoques asintóticos y no asintóticos (el primero para estimar riquezas de especies y el segundo para rarefacción y extrapolar), se calculó y se extrapoló la riqueza de especies con la función `specpool`; con la función `colSums` se hizo una matriz de comunidad combinando, con medias numéricas, todos los sitios en uno, para estimar, con esta matriz, la riqueza con el índice *chao* (Chao & Chiu, 2016), la rarefacción y extrapolación de especies.

También se aplicaron los enfoques asintóticos y no asintóticos para generar una matriz de comunidad agrupada según Ward usando los verbos `mutate` y `select` de `dplyr`; se estimó la riqueza y los porcentajes bajos y altos con la función `estimación_riqueza_chao`, también se obtuvo los porcentajes de rarefacción y se extrapoló. Y para calcular la diversidad *beta* con único número se usó la función `calcular_beta_multiplicativa`; también se determinó la contribución de especies y la contribución local a la diversidad *beta* con la función `determinar_contrib_local_y_especie`

con el método hellinger y un indicador alpha de 0.05.

Finalmente, se realizó la autocorrelación o análisis espacial ecológico usando las herramientas de las librerías ape, spdep, ade4, adegraphics, adespatial, vegan, tidyverse, sf, gridEstra, grid, y gtable. Se generó una matriz Hellinger con la función decostand y se transformó la matriz ambiental con el objeto sp para generar vecindad, del mismo modo se usó la función nb2listw para crear una lista de vecinos con pesos espaciales. También, se hizo una autocorrelación espacial por medio de correlograma de una sola variable ambiental con la función sp.correlogram; del mismo modo, se hizo una con múltiples variables con la matriz de Hellinger para conocer la abundancia de especies con la función calcular_autocorrelacion con el método de Moran's I.

Además, se hizo una versión reversa de la matriz Hellinger con rev y luego se generaron gráficos con la función par. De igual manera, se hizo la autocorrelación para las variables ambientales con la función calcular_autocorrelacion y el método de Moran's I. Se hizo, también, una autocorrelación espacial por medio de la prueba de Mantel (Borcard & Legendre, 2012) o matrices de distancias a datos de comunidad con las funciones resid, dist y mantel.correlog. Se realizó una autocorrelación espacial mediante pruebas de permutación para el I de Moran aplicado a abundancia de especies transformadas sin tendencia con sapply usando la función mi_fam_sin_tendencia; aplicado a variables ambientales, se usó sapply con las funciones var, gtable_filter y grid.arrange para filtrar celdas por nombre de las variables ambientales; se usó el mismo procedimiento aplicado a abundancias de especies transformadas y aplicado a abundancias de especies transformadas sin tendencias.

4 Resultados

4.1 Análisis Exploratorio de Datos (EDA)

La aplicación de los EDA en el estudio produjo informaciones generales sobre la familia *Myrtaceae*, sus especies, su abundancia, las variables ambientales, y datos generales de correlación con las variables ambientales. La parcela de muestreo permanente (de ahora en adelante *pmp*) de 50 hectáreas en BCI tiene 5579 individuos (abundancia) pertenecientes a la familia *Myrtaceae* y una riqueza de 7 especies. Las especies que más destacan por su abundancia son las *Eugenia galalonensis* (1975 individuos) y *Eugenia oerstediana* (1838 individuos), en cambio, las especies con menor número de individuos en la parcela son *Myrcia gatunensis* (56 individuos) y *Psidium friedrichsthalianum* (50 individuos), ver tabla 1 y figuras 2 y 3.

Table 1: Abundancia por especies de la familia *Myrtaceae*

Especies	Número
<i>Eugenia galalonensis</i>	1975
<i>Eugenia oerstediana</i>	1838
<i>Eugenia coloradoensis</i>	609
<i>Chamguava schippii</i>	541
<i>Eugenia nesiotica</i>	502
<i>Psidium friedrichsthalianum</i>	58
<i>Myrcia gatunensis</i>	56

De acuerdo con los resúmenes estadístico el 50% de la abundancia de especies, según la mediana, es mayor de 541 individuos (*Chamguava schippii*) y la abundancia de especie promedio en

pmp es de 797 individuos. En cuanto a la riqueza de la familia *Myrtaceae*, indica que el 50% de la riqueza, según la mediana, es de 6 especies y una riqueza promedio de 5.5, también hay que agregar que la riqueza mínima es de 4 especies en *pmp*. De igual manera, hay que mencionar que la riqueza y la abundancia varía por cuadro de 1h en *pmp*, arroja que los sitios con mayor riqueza son los cuadros 13, 14, 17, 22 y 40 (7), en cambio los de menor riqueza son los cuadros 2, 5, 11, 36, 46 y 47 (4); y los sitios con mayor abundancia fueron los cuadros 19, 20, 15, 40 y 38, y los de menor abundancia fueron los cuadros 1, 36, 30, 2 y 9; donde coinciden el sitio 40 con mayor riqueza y abundancia por cuadro, y los sitios 2 y 36 con menor riqueza y abundancia, lo que indica que existe una distribución aleatoria y desigual de las riquezas y abundancias en los sitios de *pmp* (ver figura 4). De las especies de *Myrtaceae* que presentan una correlación simple (de Pearson) positiva significativa, en comparación a las demás, están *Chamguava schippii* y *Eugenia galalonensis* con un índice de correlación de 0.30, también *Chamguava schippii* con *Eugenia Nesiotica* con un índice de correlación de 0.29 (ver figura 5).

En cuanto a las variables ambientales que se destacan en *pmp* están: la geológica, que de acuerdo con el Mapa geológico del canal de Panamá y sus alrededores (R. Stewart, Stewart, & Woodring, 1980) y los resultados de EDA la *pmp* se caracteriza, por sus rocas basalto, tipo intrusiva y extrusiva, del Mioceno medio y superior (Tb). También se caracteriza por tener hábitats de bosque viejo en relieve bajo en su mayoría al oeste de la parcela, en algunas partes del centro y al este de *pmp*; por sus hábitat de bosque viejo en relieve alto al centro y este de *pmp*; se agregan los bosques jóvenes de los sitios 30 y 35; y los bosques de pendiente baja concentradas al este y con algunos sitios aleatorios en *pmp*; al final, está el bosque de pantano en el centro de *pmp* (ver figura 6).

Por último, se destacan las variables de terreno como elevación media que predomina la mitad hacia el norte de la parcela en un 50% en adelante. También se destaca la forma vertiente del terreno donde en una escala de 0 a 1, esta predomina en casi toda la parcela sobre el 50%. En cuanto a minerales, el aluminio se manifiesta al oeste de las parcelas, superior al 50% y en uno que otro lugar entre la parte este de la parcela. El hierro se manifiesta en un porcentaje superior al 75% al este de la parcela. Finalmente, el pH se manifiesta desde la mitad de los sitios de la parcela hacia el este como alcalino y el resto como ácido (ver figura 7).

4.2 Asociación de especies y variables ambientales

La asociación presente entre sitios de la parcela, obtenida al calcular la distancia euclídea usando la matriz de comunidad transformada con el método de *Hellinger*, señala varios grupos de sitios extremadamente semejantes, según la matriz de disimilitud ordenada hay un grupo limitado por los cuadros 20-25; un segundo grupo, cuadros 12-32; un tercer grupo, los cuadros 32-42; un cuarto grupo, los cuadros 34-46; un quinto grupo, los cuadros 43-49; y un último grupo, el sexto, limitado por los cuadros 16-29; estos lugares comparte una similitud superior de 70% (ver figura 8).

La distancia de *Jaccard* y la distancia de *Bray-Curtis* presentan un comportamiento distinto en cuanto a los grupos que se forman por la similaridad, al de distancia euclídea con el método *Hellinger*. Con el método de *Jaccard* se produjeron 7 grupos que comparten similitudes mayores al 70%: el primero limitado por los cuadros 47-46, el segundo por los cuadros 43-39, el tercero por los cuadros 24-30, el cuarto grupo por los cuadros 32-22, el quinto por los cuadros 22-26, el sexto por 48-49, el séptimo y último 45-50. También, la distancia de *Jaccard* señala la existencia de sitios con exclusividad considerable, como los sitios 1 y 2 con una exclusividad de especies del 20% por lo que tienen similaridad visible de especies del 80% (ver figura 9). Según la fórmula de similaridad

de *Jaccard* ambos sitios comparten 4 especies, pero el sitio 2 no tiene especies exclusivas, en cambio, el sitio 1 tiene una especie exclusiva; según el porcentaje de especies compartida o similaridad, se confirma que comparten el 80% de sus especies (ver figura 3).

En cuanto a las variables ambientales, los resultados del análisis de correlación para las variables de suelo o edáficas indican que similaridades importantes para los sitios limitados por los cuadros 12-16 de la matriz de disimilaridad ordenada (ver figura 10), creando lo que es un gran cluster de variables edáficas muy similares. Mientras que, las variables mixtas (heterogeneidad ambiental, habitat y quebrada) se aprecia en la matriz de disimilaridad ordenada (ver figura 11) un gigantesco grupo de sitios que comparten similitudes, estos sitios están limitados por los cuadros 25-27; en este conjunto se puede apreciar la similitud, en específico, de el cuadro 2 y el 7 con un hábitat de bosque viejo y relieve bajo, en ambos cuadros hay quebrada y además, poseen unos valores moderados de heterogeneidad ambiental; si se observan los mapas de riqueza y abundancia de la familia *Myrtaceae* (ver figuras 3 y 4), el sitio 2 presenta la riqueza más baja y una abundancia baja a diferencia del sitio 7 que tiene una riqueza alta y una abundancia media, por lo que no se considera que tengan relación con la riqueza y la abundancia de la especie, pero si tienen una similaridad.

5 Discusión

6 Agradecimientos

7 Información de soporte

Table 2: Materiales usados en el estudio

Materiales	Uso
RStudio	Redacción del manuscrito, procesamientos de datos censales de la familia <i>Myrtaceae</i> por medio de Scripts.
library vegan	Conjunto de herramientas para hacer análisis de diversidad, ordenación de comunidad y análisis de disimilitud.
library tidyverse	Colección de paquetes que permiten transformar, importar, visualizar, modelar y presentar distintos datos.
library sf	Creación de simple features, ampliando objetos tipo data.frame con una columna de lista de características simples.
library mapview	Para ver objetos espaciales de forma interactiva sobre un mapa base.
library RColorBrewer	Para crear paletas de colores para mapas temáticos.
library ez	Permiten una visualización y análisis de datos simples y con especificaciones consistentes.
library psych	Conjunto de herramientas para hacer análisis de datos multivariados.
library tmap	Para visualizar, con mapas temáticos, la distribución de datos espaciales.
library adespatial	Herramienta para hacer análisis espaciales, a distintas escalas, de datos multivariados.
library broom	Para resumir información de objetos estadísticos en tablas.
library cluster	Para el clúster análisis o de agrupamiento, que permiten encontrar grupos de datos.

Materiales	Uso
library gclus	Ordena en matrices de diagramas, dispersión y coordenadas paralelas con un índice, los paneles.
library magittr	Paquete que permite, mediante mecanismos, cadenas de comandos con el operador pipa.
library pvclust	Paquete que permite implementar un remuestreo multiescala para evaluar inconsistencia en análisis de agrupamiento jerárquico.
library indicpecies	Estima el valor estadístico de la relación presencia-abundancia de especies y sus sitios.
library plyr	Conjunto de herramientas que permiten separar, aplicar y combinar datos para generar resúmenes estadísticos de ellos.
library SpadeR	Estima diversos índices de biodiversidad y medidas de similitud de datos individuales tomados de diversas comunidades.
library iNEXT	Paquete que permite calcular y trazar la rarefacción y extrapolación de diversidad de especies.
library vegetarian	Para calcular la diversidad por comunidad en un conjunto de datos.
library ape	Paquete que permite hacer análisis filogenéticos y evolutivos de árboles.
library spdep	Conjunto de funciones para crear matrices de ponderaciones espaciales de puntos de patrones polígonos, entre otros.
library ade4	Herramientas para análisis de datos multivariados.
library adegraphics	Sirve para hacer representaciones gráficas de datos multivariados.
library gridExtra	Ofrece funciones para poder trabajar con gráficos en <i>grid</i> y crear diversos trazados en una página y dibujar tablas.
library grid	Reescribe los gráficos, sus capacidades y da soporte a la interacción.
library gtable	Herramientas que permiten trabajar más fácil con tablas.

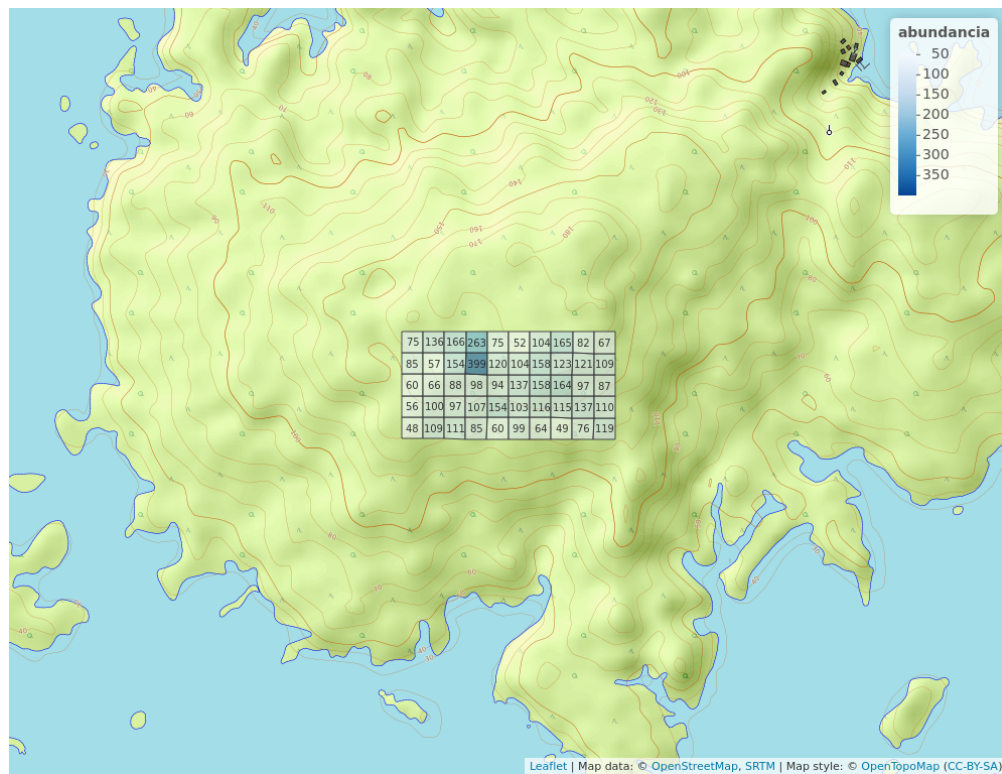


Figure 2: Abundancia por cuadros de *Myrtaceae*.

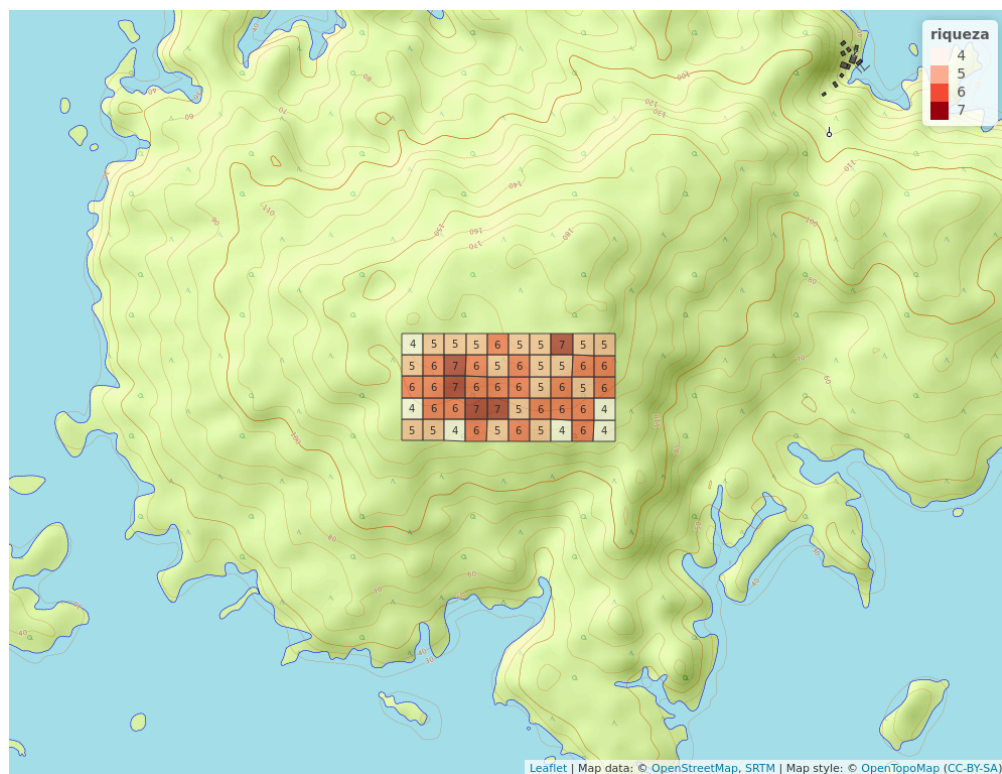


Figure 3: Riqueza por cuadros de *Myrtaceae*.

Cuadros de 1 Ha de BCI. Variables ambientales nominales

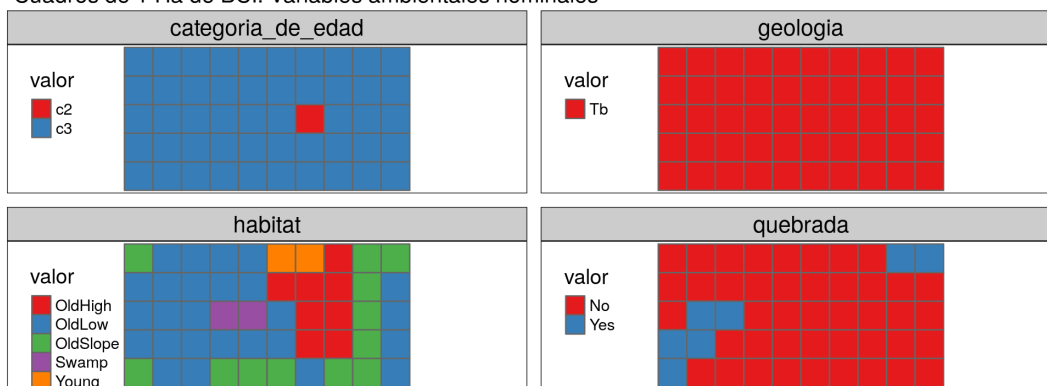


Figure 6: Variables ambientales nominales.

Cuadros de 1 Ha de BCI. Variables ambientales numéricas escaladas de 0 a 1

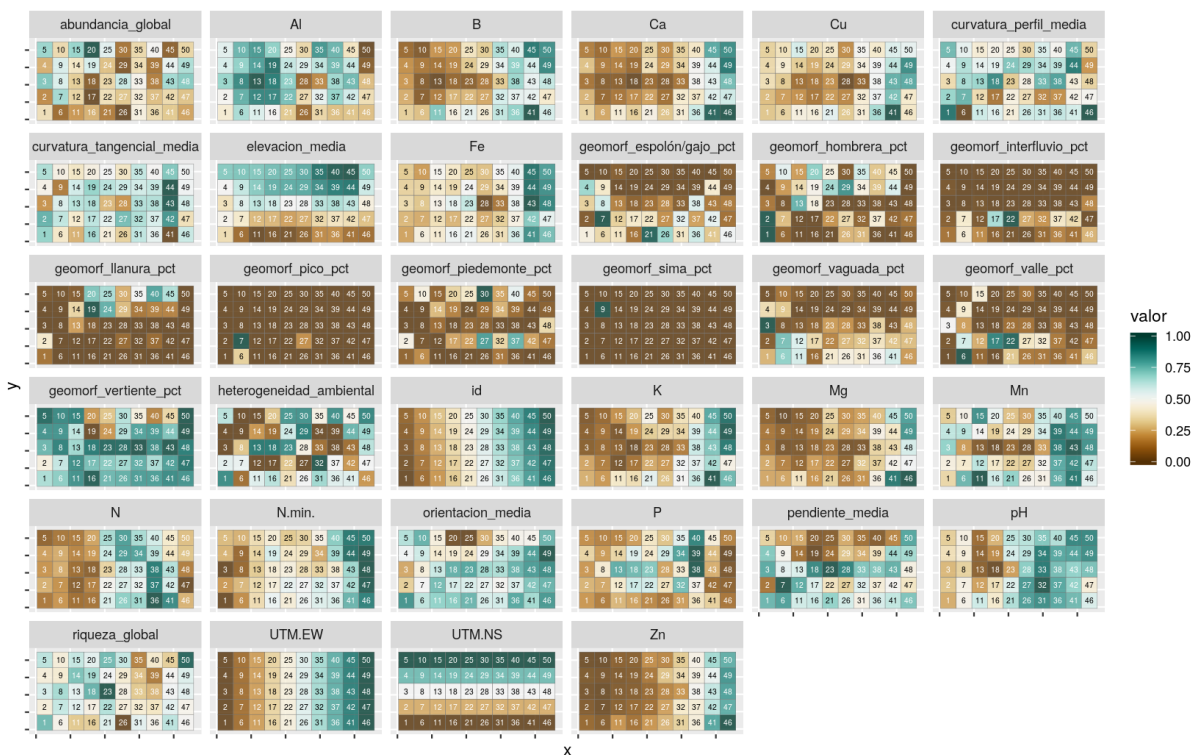


Figure 7: Variables ambientales numéricas.

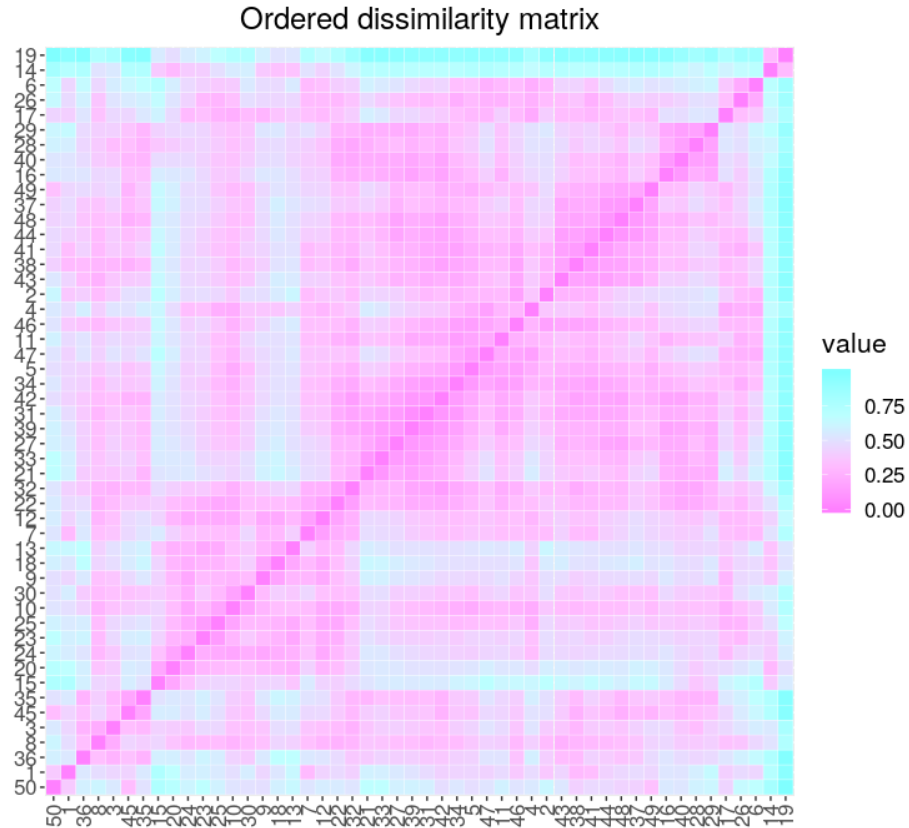


Figure 8: Matriz de asociación de sitios en la parcela de BCI. El color fucsia o rosa indica una corta distancia o que son muy similares los sitios, y el celeste indica una gran distancia o que los sitios son poco similares.

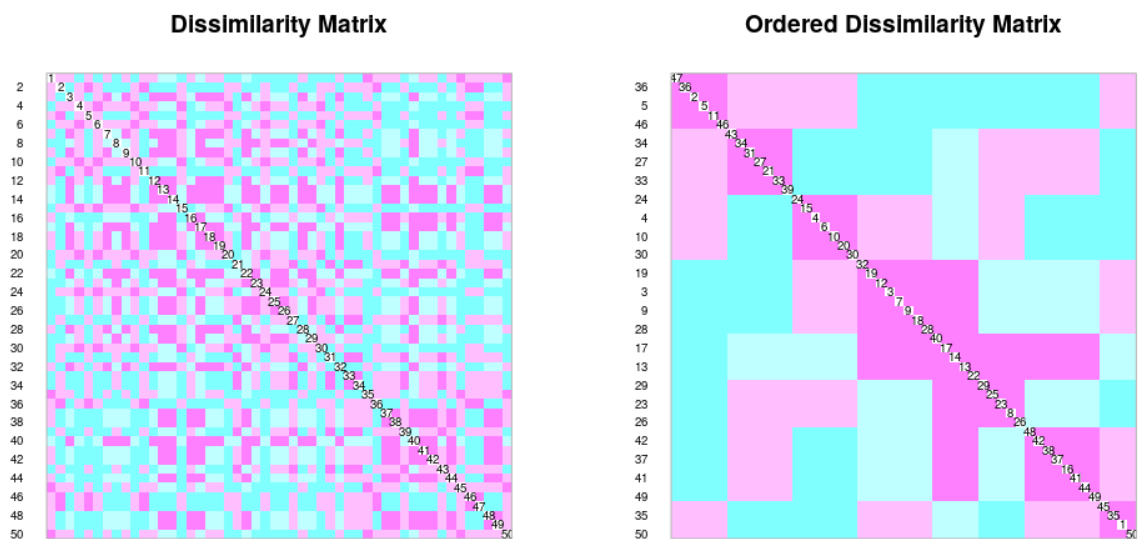


Figure 9: Matriz de disimilaridad de Jaccard. El color fucsia o rosa indica una corta distancia o que son muy similares los sitios, y el celeste indica una gran distancia o que los sitios son poco similares.

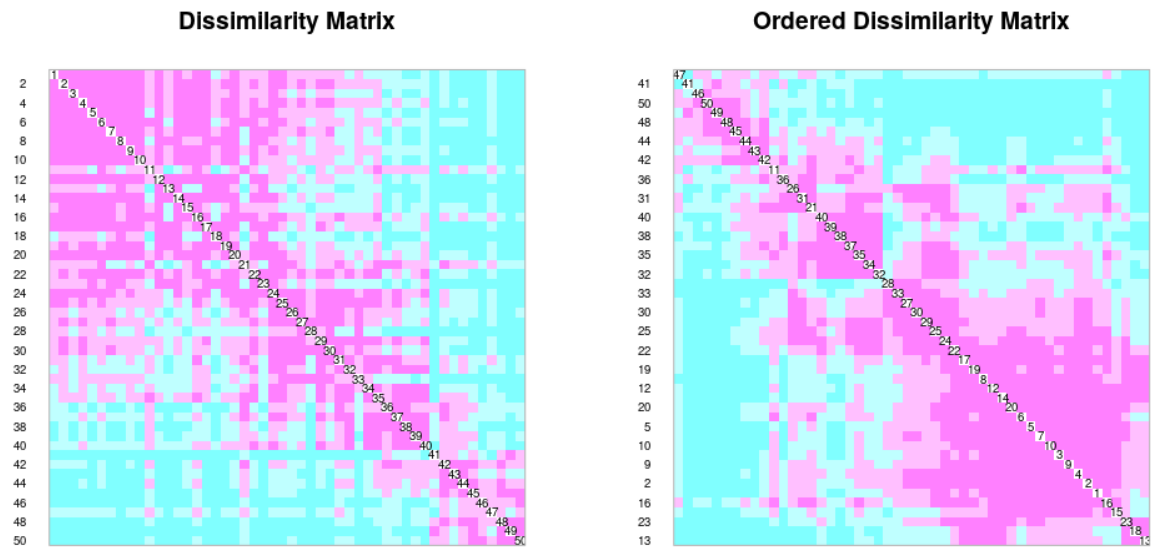


Figure 10: Matriz de disimilaridad de las variables edáficas en la parcela de BCI. El color fucsia o rosa indica una corta distancia o que son muy similares los sitios, y el celeste indica una gran distancia o que los sitios son poco similares.

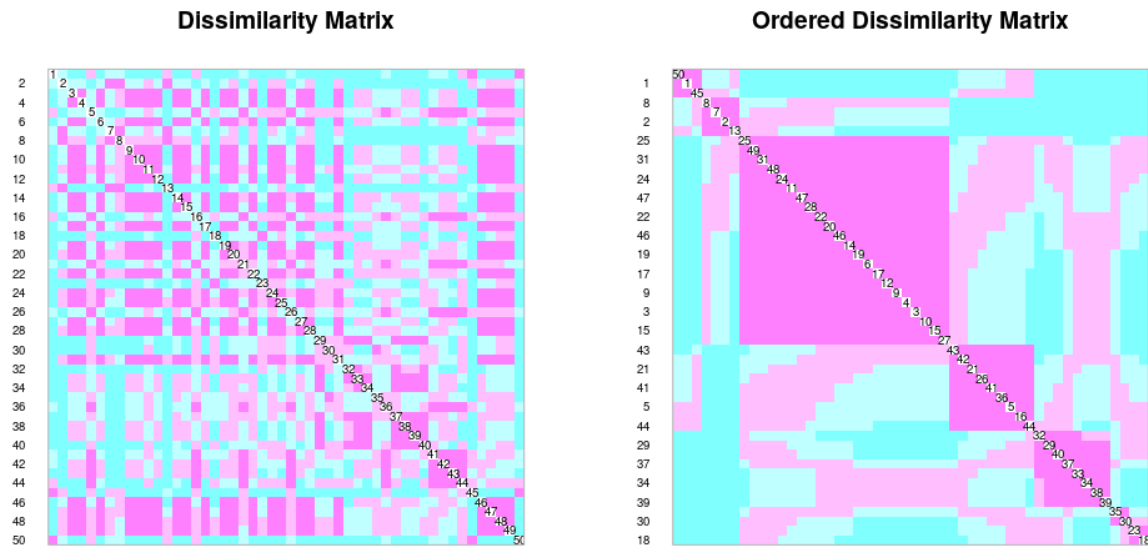


Figure 11: Matriz de disimilaridad de variables mixtas (heterogeneidad ambiental, hábitat y quebrada). El color fucsia o rosa indica una corta distancia o que son muy similares los sitios, y el celeste indica una gran distancia o que los sitios son poco similares.

8 Script reproducible

8.1 Análisis exploratorio de datos.

```
#' ---
#' title: "Análisis exploratorio de datos. Riqueza y abundancia"
#' author: "JR"
#' date: "13 de octubre, 2020"
#' output: github_document
#' ---
```

```

#' ### Área de cargar paquetes
library(vegan)
library(tidyverse)
library(sf)
source('biodata/funciones.R')

#' ### Área de cargar datos
#' Censo (el objeto se carga con prefijo "censo") y matriz de comunidad (prefijo "mc")
load('biodata/Myrtaceae.Rdata')
load('biodata/matriz_ambiental.Rdata') #Matriz ambiental, se carga como "bci_env_grid"

#' ### Imprimir datos en pantalla (impresiones parciales con head)
head(censo_myrtc)
head(mc_myrtc)
bci_env_grid # No necesita imprimirse parcialmente

#' ### También podemos usar
#' Requiere que se haya cargado ya la colección tidyverse
censo_myrtc %>% tibble
mc_myrtc %>% tibble

#' ### Lista de especies
sort(colnames(mc_myrtc))

#' ### Número de sitios, tanto en matriz de comunidad como en ambiental
#' Verifica que coinciden
nrow(mc_myrtc) #En la matriz de comunidad
nrow(bci_env_grid) #En la matriz ambiental

#' ### Riqueza numérica de especies (usando matriz de comunidad) por quadrat
#' Nota: cargar paquete vegan arriba, en el área de paquetes
specnumber(mc_myrtc)
sort(specnumber(mc_myrtc)) # Ordenados ascendentemente
summary(specnumber(mc_myrtc)) # Resumen estadístico

#' ### Abundancia de especies por quadrat
sort(rowSums(mc_myrtc))
summary(rowSums(mc_myrtc)) # Resumen estadístico

#' ### Abundancia por especie
sort(colSums(mc_myrtc))
summary(colSums(mc_myrtc)) # Resumen estadístico

#' ### Riqueza numérica de toda la "comunidad"
specnumber(colSums(mc_myrtc))

#' ### Abundancia de toda la comunidad

```

```

sum(colSums(mc_myrtc))

#' ### Una tabla para el manuscrito, es necesario asignarle nombre
#' Para esto, usaré la colección "tidyverse"
abun_sp <- censo_myrtc %>%
  group_by(Latin) %>%
  count() %>%
  arrange(desc(n))
abun_sp

#' ### Un gráfico para el manuscrito
#' Gráfico de mosaicos de la abundancia por especie por cuadros
abun_sp_q <- crear_grafico_mosaico_de_mc(mc_myrtc, tam_rotulo = 12)
abun_sp_q

#' ---
#' title: "Análisis exploratorio de datos. Colección tidyverse"
#' author: "JR"
#' date: "18 de octubre, 2020"
#' output: github_document
#' ---

#' # ¿Qué es tidyverse?
#'
#' Es una colección de paquetes con los que podrás importar, transformar, visualizar, modelar y
#'
#' Todos estos paquetes comparten estructuras comunes. Una de las herramientas que incorpora la
#'
#' 'objeto1 %>% funcion1()' es equivalente a 'funcion1(argumento1 = objeto1)'
#'
#' > La idea del *pipe* pertenece a la tradición de sistemas tipo Unix y, en origen, su función
#'
#' Su ventaja radica en que, si necesitaras continuar procesando los datos, no tendrás que anidar
#'
#' 'resultado <- objeto1 %>% funcion1() %>% funcion2() %>% funcion3()'
#'
#' Puedes leer lo anterior como "objeto1 pasa como primer argumento de funcion1, **luego** el r
#'
#' Para replicar esta operación sin la pipa, podrías realizarlo de, por ejemplo, dos maneras dis
#'
#' * Opción 1, anidar:
#'
#' 'resultado <- funcion3(funcion2(funcion1(objeto1)))'
#'
#' Opción 2, crear objetos intermedios:
#'
#' 'tmp1 <- funcion1(objeto1)'
#' 'tmp2 <- funcion2(tmp1)'

```

```

# 'resultado <- funcion3(tmp2)'
#
# ' Notarás que, comparada con estas dos últimas opciones, la tubería es más limpia que estas dos
#
# ' ¡Comencemos!
#
# ' ## Paquetes
#
library(tidyverse)
library(sf)
#
# ' > 'sf' te ayudará a leer el objeto 'bci_env_grid' como un *simple feature*, el cual se encuen
#
# ' ## Cargar datos
#
load('biodata/matriz_ambiental.Rdata')
load('biodata/Myrtaceae.Rdata')
#
# ' ## Paquete 'dplyr'
#
# ' Te servirá para manipular datos mediante verbos. Los verbos de 'dplyr' que conocerás son (hay
#
# ' ### Verbo 'select'
#
# ' Comúnmente, necesitas seleccionar una o varias columnas de una tabla. Para esto existe el ver
bci_env_grid %>%
  select(id, pH)
# ' > Importante: el objeto 'bci_env_grid' permanece intacto, a menos que se use dicho nombre par
# ' Fíjate en la clase del objeto 'bci_env_grid'. Para ello usaré la función de R 'class'. No sól
bci_env_grid %>%
  class
# ' El objeto 'bci_env_grid' es a la vez de clase 'sf' (*simple feature*) y 'data.frame', es deci
bci_env_grid %>%
  select(id, pH) %>%
  st_drop_geometry
# ' Fíjate ahora en la clase de 'bci_env_grid %>% select(id, pH) %>% st_drop_geometry', que en es
bci_env_grid %>%
  select(id, pH) %>%
  st_drop_geometry %>%
  class
# ' > Al introducir un '<enter>' después de la pipa, el código puede continuar en la línea sigueie
#
# ' Seleccionaré, y a la vez renombraré, dos columnas con 'select' (recuerda: no estoy modificand
bci_env_grid %>%
  select(id_de_quadrat = id, pH_del_suelo = pH) %>%
  st_drop_geometry %>%
  head
#

```



```

#’ Otra funcionalidad de ‘select’ es poder seleccionar columnas según patrones. Por ejemplo, si
#’
bci_env_grid %>%
  select(contains('geomorf')) %>%
  st_drop_geometry
#’ ...y también usando expresiones regulares con ‘matches’, usando por ejemplo dos cadenas de ca
bci_env_grid %>%
  select(matches('geomorf|habit', ignore.case = F)) %>%
  st_drop_geometry
#’ ...o pidiendo todas las columnas que comienzan por mayúsculas, excepto las que comienzan por
bci_env_grid %>%
  select(matches('^[A-T,Z]', ignore.case = F)) %>%
  st_drop_geometry
#’ ### Verbo ‘filter’
#’
#’ Ahora mostraré sólo los elementos con ‘pH’ mayor que 5, usando el verbo ‘filter’
bci_env_grid %>%
  select(id, pH) %>%
  st_drop_geometry %>%
  filter(pH>5)
#’ O filtro por aquellos con ‘id’ 31 y 50:
bci_env_grid %>%
  select(id, pH) %>%
  st_drop_geometry %>%
  filter(id == c(31, 50))
#’
#’ ### Verbo ‘arrange’
#’
#’ Pruebo también con la matriz de comunidad. Por ejemplo, introduzco en la tubería la función ‘
mc_myrtc %>%
  colSums
#’ Y también obtengo la abundancia por quadrat.
mc_myrtc %>%
  rowSums
#’ Uso a continuación el verbo ‘arrange’ para mostrar los registros de la matriz ambiental orden
bci_env_grid %>%
  select(id, pH) %>%
  st_drop_geometry %>%
  arrange(pH)
#’ Ahora usaré ‘arrange’ para mostrar los registros de la matriz ambiental ordenados DESCendente
bci_env_grid %>%
  select(id, pH) %>%
  st_drop_geometry %>%
  arrange(desc(pH))
#’
#’ ### Verbo ‘mutate’
#’
#’ Usaré el verbo ‘mutate’ para crear una nueva columna. Por ejemplo, creo una columna que conte

```

```

bci_env_grid %>%
  st_drop_geometry %>%
  select(habitat, quebrada) %>%
  mutate(habitat_quebrada = paste(habitat, quebrada, sep = ', '))
#' Ahora 'mutate', pero con números: creo una columna de área de cada cuadro (necesitas también
bci_env_grid %>%
  mutate(area = st_area(geometry)) %>%
  select(id, area) %>%
  st_drop_geometry %>%
  head
#' ...y ahora más complejo: obtengo la densidad de individuos por metro cuadrado, ordenados desc
bci_env_grid %>%
  mutate(area = st_area(geometry), densidad_indiv = abundancia_global/area) %>%
  select(id, densidad_indiv) %>%
  st_drop_geometry %>%
  arrange(desc(densidad_indiv)) %>%
  head
#'
#' ### Verbos 'group_by' y 'summarise'
#'
#' Los verbos 'group_by' y 'summarise' son útiles para producir resúmenes por grupos.
#'
#' Agruparé la matriz ambiental por la columna 'habitat', dejando sólo las variables numericas q
agrupado_por_habitat <- bci_env_grid %>%
  st_drop_geometry %>%
  group_by(habitat) %>%
  select_if(is.numeric) %>%
  select(-id, -UTM.EW, -UTM.NS)
agrupado_por_habitat
#' Observa el encabezado: el objeto es 'A tibble: 50 x 32' y hay 5 grupos ('Groups:  habitat [5
agrupado_por_habitat %>% summarise(n = n())
#' ...y también algunos estadísticos de las columnas 'pH', 'abundancia_global' y 'riqueza_global
agrupado_por_habitat %>%
  summarise(
    n = n(),
    media_pH = mean(pH),
    media_abundancia = mean(abundancia_global),
    media_riqueza = mean(riqueza_global)
  )
#' ...o la media de todas las variables numéricas
agrupado_por_habitat %>%
  summarise_all(mean)
#' ...no caben, mejor por partes
agrupado_por_habitat %>%
  summarise_all(mean) %>%
  select(1:6) %>%
  print(width=300)
agrupado_por_habitat %>%

```

```

    summarise_all(mean) %>%
    select(1,7:12) %>%
    print(width=300)
agrupado_por_habitat %>%
    summarise_all(mean) %>%
    select(1, 13:25) %>%
    print(width=300)
agrupado_por_habitat %>%
    summarise_all(mean) %>%
    select(1, 26:32) %>%
    print(width=300)
#' ...y no sólo un estadístico, sino varios:
agrupado_por_habitat %>%
    summarise_all(
      list(
        media = mean,
        mediana = median,
        varianza = var,
        minimo = min,
        maximo = max
      )
    )
#' Ejecuto también un ANOVA de una vía, de la 'riqueza_global' respecto de 'habitat' de tipo '01'
agrupado_por_habitat %>%
    filter(str_detect(habitat, 'Old*')) %>%
    oneway.test(formula = riqueza_global ~ habitat)
#' El resultado sugiere que "existen 'diferencias significativas' de 'riqueza_global' entre 'habitat'"
#'
#' Finalmente, te muestro 'join'. Más que una función, 'join' es una función genérica con varios usos
#'
#' Obtendré una tabla con dos columnas: código identificador de quadrat de 1 ha (le llamaré 'id')
id_abundancia_fam <- mc_myrtc %>%
    mutate(abundancia_mi_familia = rowSums(.)) %>%
    rownames_to_column(var = 'id') %>%
    mutate(id = as.numeric(id)) %>% #Numérico, garantiza compatibilidad con id de bci_env_grid
    select(id, abundancia_mi_familia)
id_abundancia_fam %>% tibble
#' Dado que 'id_abundancia_fam' y 'bci_env_grid' comparten el campo 'id', a través de éste se puede unir
bci_env_grid %>%
    inner_join(y = id_abundancia_fam, by = 'id')
#' El resultado muestra la 'bci_env_grid', ahora con los datos de mi familia como parte de la misma
#'
#' ## 'tidyr'
#'
#' Te ayudará a reordenar datos, mediante transformación de su estructura, para organizarlos de una manera más
#'
#' ### Verbo 'pivot_longer'
#'

```

```

#’ Cuando necesitas reunir varias columnas, o lo que es lo mismo, hacerlas que pivoten a lo largo
#’
#’ 
#’ *Tomado de: UC Business Analytics R Programming Guide. Reshaping Your Data with tidyr. https://uc-r.github.io/public/images/dataWrangling/gather1.png
#’
#’ Es común realizar "reunión" columnas cuando nos interesa aplicar análisis masivos a múltiples
#’
#’ Pongo un ejemplo. Por tipo de hábitat, ¿cuánto es el promedio de los porcentajes de cada uno
pivotpaso1 <- bci_env_grid %>%
  st_drop_geometry %>%
  select(matches('geomorf|habitat'))
pivotpaso1 %>% tibble
#’ ...luego reunir todas las columnas de geomorfología pivotando en torno a la columna 'habitat'
pivotpaso2 <- pivotpaso1 %>%
  pivot_longer(
    cols = contains('geomorf'),
    names_to = 'variable',
    values_to = 'valor')
pivotpaso2 %>% tibble
#’ ...y finalmente obtener las medias de porcentajes de geomorfología por cada grupo de habitat,
pivotpaso3 <- pivotpaso2 %>%
  group_by(habitat, variable) %>%
  summarise(media = mean(valor))
pivotpaso3 %>% arrange(habitat, desc(media)) %>% print(n=Inf)
#’ 'pivot_longer' también es útil para realizar paneles de gráficos de muchas variables, como ve
#’
#’ La operación contraria a 'pivot_longer' se realiza con 'pivot_wider'. Supongamos que ahora ne
#’
pivotpaso3 %>%
  ungroup() %>%
  pivot_wider(
    id_cols = habitat,
    names_from = variable,
    values_from = media)
#’
#’ ## 'ggplot2'
#’
#’ Te ayudará en la visualización de tus datos, utilizando gramática de gráficos.
#’
#’ Un gráfico 'ggplot' utiliza capas para mostrar la información. Los objetos fuente son 'data.f
#’
#’ Explicaré su uso con ejemplos, descomponiendo las partes de una sentencia 'ggplot' para fines
#’
#’ Primero incluiré la función 'ggplot', para crear un espacio de coordenadas según los datos di
p0 <- ggplot(bci_env_grid)
p0
#’ A continuación, definiré las variables estéticas sobre las que construiré la simbología, añad
p1 <- p0 + aes(x = abundancia_global, y = riqueza_global)

```

```

p1
#' El espacio de coordenadas ya está creado, y 'ggplot2' está preparado para aceptar geometrías.
p2 <- p1 + geom_point()
p2
#' Dado que en 'p1' definí las coordenadas de mapeo 'aes(x = abundancia_global, y = riqueza_glob
p3 <- p2 + geom_smooth(formula = y ~ x, method = 'lm')
p3
#' En 'p3', tanto 'geom_point' como 'geom_smooth' aprovechan las coordenadas del mapeo definido
#'
#' Una forma alterna permite definir la capa estética dentro de la geometría con resultado idént
p4 <- p0 +
  geom_point(mapping = aes(x = abundancia_global, y = riqueza_global))
p4
#' Esta forma tiene la ventaja de ser más corta, pero tiene la desventaja de que impide reutiliz
#'
#' También definiré propiedades globales del gráfico mediante temas.
p5 <- p3 + theme_bw()
p5
p6 <- p3 + theme_classic()
p6
p7 <- p3 + theme_minimal()
p7
#' Con una variable categórica, se pueden estilizar los elementos del gráfico. Por ejemplo, haré
p8 <- p0 +
  geom_point(
    mapping = aes(
      x = abundancia_global,
      y = riqueza_global,
      color = habitat))
p8
#'
#' Ahora mostraré cómo construir el último gráfico con una sentencia de conjunto, sin reaprovech
p9 <- ggplot(bci_env_grid) +
  geom_point(
    mapping = aes(
      x = abundancia_global,
      y = riqueza_global,
      color = habitat))
p9
#' Las posibilidades de personalización de gráficos de 'ggplot2' son enormes y superan el cometi
p10 <- p0 +
  geom_boxplot(mapping = aes(x = habitat, y = abundancia_global))
p10
#' Y ejemplifico también 'riqueza_global':
p11 <- p0 +
  geom_boxplot(mapping = aes(x = habitat, y = riqueza_global))
p11
#' ...la cual muestra efectos más marcados que 'abundancia_global'.

```

```

#’
#’ Los dos gráficos anteriores son muy informativos, pero tienen la desventaja de que para poder
#’
#’ Necesitamos tres columnas, una con los nombres de los hábitats, otra con los nombres de las v
habitat_riqueza_abundancia <- bci_env_grid %>% st_drop_geometry %>%
  select(habitat, abundancia_global, riqueza_global) %>%
  pivot_longer(
    cols = c(abundancia_global, riqueza_global),
    names_to = 'variable',
    values_to = 'valor')
habitat_riqueza_abundancia
#’ Construiré el gráfico definiendo a ‘habitat’ en el eje ‘x’, y valor en ‘y’, mientras que usar
habitat_riqueza_abundancia %>%
  ggplot() +
  aes(x = habitat, y = valor) +
  geom_boxplot() +
  facet_wrap( ~ variable, scal = 'free_y')
#’ En resumen, usa ‘tidyverse’ para sacar el máximo provecho de tus datos. El paquete ‘dplyr’ te

#’ ---
#’ title: "Análisis exploratorio de datos. Mapas de riqueza y abundancia global y de mi familia"
#’ author: "JR"
#’ date: "25 de octubre, 2020"
#’ output: github_document
#’ ---

#’ ### Cargar paquetes
library(mapview)
library(tidyverse)
library(vegan)
library(sf)
library(RColorBrewer)

#’ ### Cargar datos
load('biodata/matriz_ambiental.Rdata')
load('biodata/Myrtaceae.Rdata')

#’ ### Explorar el objeto de matriz ambiental
bci_env_grid

#’ ### Generar mapa de cuadros sin simbología
mapa_cuadros <- mapView(
  bci_env_grid,
  col.regions = 'grey80',
  alpha.regions = 0.3,
  map.types = 'OpenTopoMap',
  legend = F, zoom = 14,
  zcol = 'id') %>% addStaticLabels() %>%

```



```

leaflet::setView(
  lng = -79.85136,
  lat = 9.15097,
  zoom = 15)
mapa_cuadros
mapa_cuadros %>% mapshot(file = 'mapa_cuadros.png') #Genera archivo

#' ### Paletas
azul <- colorRampPalette(brewer.pal(8, "Blues"))
rojo <- colorRampPalette(brewer.pal(8, "Reds"))

#' ### Mapa de cuadros, simbología por abundancia global
mapa_cuadros_abun_global <- mapView(
  bci_env_grid,
  layer.name = 'abundancia',
  alpha.regions = 0.6,
  map.types = 'OpenTopoMap',
  legend = T, zoom = 14,
  col.regions = azul,
  zcol = 'abundancia_global') %>%
  addStaticLabels(label = bci_env_grid$abundancia_global, textsize = "6pt") %>%
  leaflet::setView(
    lng = -79.85136,
    lat = 9.15097,
    zoom = 16)
mapa_cuadros_abun_global
mapa_cuadros_abun_global %>% mapshot(file = 'mapa_cuadros_abun_global.png')

#' ### Mapa de cuadros, simbología por riqueza global
mapa_cuadros_riq_global <- mapView(
  bci_env_grid,
  layer.name = 'riqueza',
  alpha.regions = 0.6,
  map.types = 'OpenTopoMap',
  legend = T, zoom = 14,
  col.regions = rojo,
  zcol = 'riqueza_global') %>%
  addStaticLabels(label = bci_env_grid$riqueza_global, textsize = "7pt") %>%
  leaflet::setView(
    lng = -79.85136,
    lat = 9.15097,
    zoom = 15)
mapa_cuadros_riq_global
mapa_cuadros_riq_global %>% mapshot(file = 'mapa_cuadros_riq_global.png')

#' ### Mapa de cuadros, simbología por abundancia de mi familia
mapa_cuadros_abun_mi_familia <- mapView(
  bci_env_grid %>% mutate(abun = rowSums(mc_myrtc)),

```

```

layer.name = 'abundancia',
alpha.regions = 0.6,
map.types = 'OpenTopoMap',
legend = T, zoom = 14,
col.regions = azul,
zcol = 'abun') %>%
addStaticLabels(label = rowSums(mc_myrtc)) %>%
leaflet::setView(
  lng = -79.85136,
  lat = 9.15097,
  zoom = 15)
mapa_cuadros_abun_mi_familia
mapa_cuadros_abun_mi_familia %>% mapshot(file = 'mapa_cuadros_abun_mi_familia.png')

#' ### Mapa de cuadros, simbología por riqueza de mi familia
mapa_cuadros_riq_mi_familia <- mapView(
  bci_env_grid %>% mutate(riq = specnumber(mc_myrtc)),
  layer.name = 'riqueza',
  alpha.regions = 0.6,
  map.types = 'OpenTopoMap',
  legend = T, zoom = 14,
  col.regions = rojo,
  zcol = 'riq') %>%
addStaticLabels(label = specnumber(mc_myrtc)) %>%
leaflet::setView(
  lng = -79.85136,
  lat = 9.15097,
  zoom = 15)
mapa_cuadros_riq_mi_familia
mapa_cuadros_riq_mi_familia %>% mapshot(file = 'mapa_cuadros_riq_mi_familia.png')

#' ---
#' title: "Análisis exploratorio de datos. Mapas de variables ambientales"
#' author: "JR"
#' date: "25 de octubre, 2020"
#' output: github_document
#' ---

#' ### Cargar paquetes
library(mapview)
library(tidyverse)
library(sf)
library(RColorBrewer)

#' ### Cargar datos
load('biodata/matriz_ambiental.Rdata')

#' ### Paletas

```

```

azul <- colorRampPalette(brewer.pal(8, "Blues"))
rojo <- colorRampPalette(brewer.pal(8, "Reds"))
rojo_inv <- colorRampPalette(rev(brewer.pal(8, "Reds")))

#' ### Mapa de cuadros, simbología por pendiente
mapa_cuadros_pendiente <- mapView(
  bci_env_grid,
  layer.name = 'pendiente',
  alpha.regions = 0.4,
  map.types = 'OpenTopoMap',
  legend = T, zoom = 14,
  col.regions = rojo,
  zcol = 'pendiente_media') %>%
  addStaticLabels(label = round(bci_env_grid$pendiente_media, 1)) %>%
  leaflet::setView(
    lng = -79.85136,
    lat = 9.15097,
    zoom = 16)
mapa_cuadros_pendiente
mapa_cuadros_pendiente %>% mapshot(file = 'mapa_cuadros_pendiente.png') #Genera archivo

#' ### Mapa de cuadros, simbología por Nitrógeno
mapa_cuadros_nit <- mapView(
  bci_env_grid,
  layer.name = 'N (mg/kg)',
  alpha.regions = 0.4,
  map.types = 'OpenTopoMap',
  legend = T, zoom = 14,
  col.regions = rojo,
  zcol = 'N') %>%
  addStaticLabels(label = round(bci_env_grid$N, 1)) %>%
  leaflet::setView(
    lng = -79.85136,
    lat = 9.15097,
    zoom = 16)
mapa_cuadros_nit
mapa_cuadros_nit %>% mapshot(file = 'mapa_cuadros_nit.png')

#' ### Mapa de cuadros, simbología por pH
mapa_cuadros_ph <- mapView(
  bci_env_grid,
  layer.name = 'pH',
  alpha.regions = 0.4,
  map.types = 'OpenTopoMap',
  legend = T, zoom = 14,
  col.regions = rojo_inv,
  zcol = 'pH') %>%
  addStaticLabels(label = round(bci_env_grid$pH, 1)) %>%

```

```

leaflet::setView(
  lng = -79.85136,
  lat = 9.15097,
  zoom = 16)
mapa_cuadros_ph
mapa_cuadros_ph %>% mapshot(file = 'mapa_cuadros_ph.png')

#' ---
#' title: "Análisis exploratorio de datos. Correlaciones entre variables ambientales"
#' author: "JR"
#' date: "25 de octubre, 2020"
#' output: github_document
#' ---

knitr::opts_chunk$set(fig.width=12, fig.height=8)

#' ### Cargar paquetes
library(tidyverse)
library(sf)
library(ez)
library(psych)
library(vegan)

#' ### Cargar datos
load('biodata/matriz_ambiental.Rdata')
load('biodata/Myrtaceae.Rdata')

#' ### Una correlación simple
cor(bci_env_grid$pendiente_media, bci_env_grid$geomorf_vertiente_pct)
plot(bci_env_grid$pendiente_media, bci_env_grid$geomorf_vertiente_pct)
cor.test(bci_env_grid$pendiente_media, bci_env_grid$geomorf_vertiente_pct)

#' ### Generar objeto de columnas numéricas
#' El objeto que generaré, denominado 'env_num', no tendrá las columnas 'id' y las de coordenada
env_num <- bci_env_grid %>%
  dplyr::select_if(is.numeric) %>%
  dplyr::select(-id, -matches('^U.*')) %>%
  st_drop_geometry %>%
  mutate(
    riqueza_mifam = specnumber(mc_myrtc),
    abundancia_mifam = rowSums(mc_myrtc)) %>%
  rename_all(gsub, pattern = '_pct$', replacement = '') %>%
  rename_all(gsub, pattern = '_| ', replacement = '\\n')
env_num %>% tibble

#' ### Panel de correlaciones con herramientas del paquete 'graphics' y 'psych'
cor(env_num)
ncol(env_num)

```

```

pairs(env_num[,sample(1:33, 15)]) # paquete graphics
env_num[,sample(1:33, 15)] %>% pairs.panels #paquete psych

#' ### Panel de correlaciones con 'ez'
#'
#' ##### Todas las variables (se empasta). Comentado, sólo mostrado para fines didácticos
# p_cor_todos <- env_num %>%
#   ezCor(r_size_lims = c(4,8), label_size = 4)
# p_cor_todos

#' ##### Sólo suelo (elementos y pH), abundancia/riqueza
p_cor_suelo_ar <- env_num %>%
  dplyr::select(matches('^[A-T,Z]|abundancia|riqueza|^pH$', ignore.case = F)) %>%
  ezCor(r_size_lims = c(4,8), label_size = 3)
p_cor_suelo_ar

#' ##### Sólo heterogeneidad, geomorfología, abundancia/riqueza
p_cor_geomorf_ar <- env_num %>%
  dplyr::select(-matches('^[A-T,Z]|pH$', ignore.case = F)) %>%
  ezCor(r_size_lims = c(4,8), label_size = 3)
p_cor_geomorf_ar

#' ##### Matriz de comunidad
p_cor_mc <- mc_myrtc %>%
  rename_all(gsub, pattern = '_| ', replacement = '\\n') %>%
  ezCor(r_size_lims = c(4,8), label_size = 3)
p_cor_mc

#' ---
#' title: "Análisis exploratorio de datos. Mapas de variables ambientales por lotes"
#' author: "JR"
#' date: "3 de diciembre, 2020"
#' output: github_document
#' ---

knitr::opts_chunk$set(fig.width=12, fig.height=8)

#' ## Preámbulo
#'
#' ### Cargar paquetes
#'
library(tmap)
library(sf)
library(tidyverse)
library(RColorBrewer)
#'
#' ### Cargar datos
#'

```

```

load('biodata/matriz_ambiental.Rdata')
#
#' ## Convertir a KML
#
st_write(
  bci_env_grid %>% rename(Name = id),
  driver = 'KML',
  dsn = 'matriz_ambiental.kml')
st_write(
  bci_env_grid %>% rename(Name = id) %>% st_centroid(),
  driver = 'KML',
  dsn = 'matriz_ambiental_puntos.kml')
#
#' Uní los dos archivos anteriores en un único KML nombrado como 'mapa_cuadros_1ha_para_google_e
#
#' ## Generar mapas por lotes
#
#' ### Variables ambientales numéricas con 'ggplot2'
#
mapas_var_amb_num_gg <- bci_env_grid %>%
  select_if(is.numeric) %>%
  gather(variable, valor, -geometry) %>%
  group_by(variable) %>%
  mutate(
    valor = scales::rescale(valor, to = c(0, 1)),
    id = rep(1:50)) %>%
  ggplot +
  aes(geometry = geometry, fill = valor) +
  theme(axis.text = element_blank()) +
  geom_sf(lwd = 0.1, color = 'grey50', alpha = 0.8) + coord_sf() +
  scale_fill_gradientn(colours = brewer.pal(11, 'BrBG')) +
  geom_sf_text(aes(label = id, color = between(valor, 0.3, 0.7)), size = 1.75) +
  scale_color_manual(guide = FALSE, values = c("white", "black")) +
  facet_wrap(~ variable, ncol = 6) +
  ggtitle('Cuadros de 1 Ha de BCI. Variables ambientales numéricas escaladas de 0 a 1')
mapas_var_amb_num_gg
#
#' PNG
#
png(
  filename = 'mapas_variables_ambientales_numericas.png',
  width = 1700, height = 1080, res = 150)
mapas_var_amb_num_gg
dev.off()
#
#' ### Variables ambientales numéricas con 'tmap'
#
mapas_var_amb_num_tmap <- bci_env_grid %>%

```



```

select_if(is.numeric) %>%
gather(variable, valor, -geometry) %>%
group_by(variable) %>%
mutate(
  valor = scales::rescale(valor, to = c(0, 1)),
  id = rep(1:50)) %>%
tm_shape() +
tm_polygons(col = 'valor',
  palette = brewer.pal(11, 'BrBG'),
  style = 'cont',
  legend.is.portrait = FALSE) +
tm_facets(by = 'variable', ncol = 6, nrow = 6) +
tm_layout(main.title="Cuadros de 1 Ha de BCI. Variables ambientales numéricas escaladas de 0 a 1",
  main.title.size = 0.7,
  legend.outside.position="bottom",
  legend.outside=TRUE,
  legend.width = 0.2,
  legend.text.size = 0.5,
  legend.stack="horizontal",
  outer.margins=0)
mapas_var_amb_num_tmap
#'
#' PNG
#'
png(
  filename = 'mapas_variables_ambientales_numericas_tmap.png',
  width = 1800, height = 1400, res = 350, pointsize = 12)
mapas_var_amb_num_tmap
dev.off()
#'
#' ### Variables ambientales nominales con 'tmap'
#'
mapas_var_amb_nom_tmap <- bci_env_grid %>%
  select_if(negate(is.numeric)) %>%
  gather(variable, valor, -geometry) %>%
  tm_shape() +
  tm_polygons(col = 'valor',
    palette = brewer.pal(8, 'Set1'),
    legend.show = T) +
  tm_facets(by = 'variable', ncol = 2, free.scales = T, free.coords = T) +
  tm_layout(main.title="Cuadros de 1 Ha de BCI. Variables ambientales nominales",
    main.title.size = 0.7,
    asp = 3.5,
    legend.text.size = 0.7)
mapas_var_amb_nom_tmap
#'
#' PNG
#'

```

```
png(
  filename = 'mapas_variables_ambientales_nominales_tmap.png',
  width = 2000, height = 1200, res = 350, pointsize = 12)
mapas_var_amb_nom_tmap
dev.off()
```

8.2 Medición de asociación

```
#' ---
#' title: "Medición de asociación. Modo Q aplicado a mi familia asignada"
#' author: "JR"
#' date: "9 de noviembre, 2020"
#' output: github_document
#' ---

knitr::opts_chunk$set(fig.width=12, fig.height=8)

#' ## Preámbulo

#' ### Cargar paquetes
library(vegan)
library(adespatial)
library(broom)
library(tidyverse)
library(sf)
library(cluster)
library(gclus)
source('biodata/funciones.R')

#' ### Cargar datos
#'
load('biodata/matriz_ambiental.Rdata')
load('biodata/Myrtaceae.Rdata')
#'
#' ## Modo Q: matrices de disimilaridad entre objetos
#'
#' ### Modo Q para datos cuantitativos de especies (abundancia). Datos de mi familia asignada
#'
#' Aplicado a mi familia asignada de BCI, en la forma de matriz de distancia euclídea, utilizando
#'
mi_fam_d_hel <- dist.ldc(mc_myrtc, "hellinger", silent = T)
mi_fam_d_hel %>% tidy # Para evitar desbordar la consola
#'
#' Para interpretar esta matriz, es necesario representarla gráficamente. En la representación e
#'
coldiss(mi_fam_d_hel, diag = T)
#'
```

```

#’ Mejorable el gráfico, quizá este es más explícito:
#’
coldissgg(mi_fam_d_hel, ordered = T, nc = 4, fsz = 0)
#’
#’ Con valores de distancia sobreimpresos (se empastan un poco)
#’
coldissgg(mi_fam_d_hel, ordered = T, nc = 4, fsz = 1.5)
#’
#’ Puedes guardar el gráfico usando el botón ‘Export’ de la pestaña ‘Plots’
#’
#’ Una forma alterna de guardar el gráfico es mediante funciones de R. La calidad de gráficos ex
#’
png(
  filename = 'matriz_disimilaridad_hellinger.png',
  width = 2400, height = 1200, pointsize = 32
)
coldiss(mi_fam_d_hel, diag = T)
dev.off()
#’
#’ MUY IMPORTANTE. La última función, ‘dev.off()’, es necesaria para cerrar el dispositivo. Si n
#’
#’ ### Modo Q para datos binarios (presencia/ausencia)
#’
#’ Habitualmente, sólo dispones de datos de presencia/ausencia. En tales casos, existe un conjun
#’
#’ Un error común consiste en referirse a los índices de Jaccard y de Sorensen "a secas", sin es
#’
#’ Si alguna vez te enfrentas a textos donde no se especifica qué tipo de métrica se usa, te sug
#’
#’ Recalco: es imprescindible declarar qué tipo de métrica estás usando. Ejemplos de redacci
#’
#’ - Correcto: "índice de disimilaridad de Jaccard", "índice de similaridad de Sorensen"
#’
#’ - Incorrecto: "índice de Jaccard", "índice de Sorensen".
#’
#’ A continuación, muestro cómo calcular la distancia de Jaccard ( $D_{J}$ ) en un ú
#’
mi_fam_jac <- vegdist(mc_myrtc, method = 'jac', binary = T)
mi_fam_jac %>% tidy # Mostrando sólo las primeras 10 combinaciones en modo data.frame
#’
#’ El argumento ‘binary=T’ en ‘vegdist’ "ordena" que se realice primero ‘decostand(mc_apcyn_meli
#’
#’ En esta matriz de disimilaridad, al igual que en la anterior, un valor pequeño (rosa) signifi
#’
coldiss(mi_fam_jac, diag = T)
#’
#’ La distancia de Jaccard ( $D_{J}$ ) se puede expresar como "la proporción de especies
#’

```

```

# Como la distancia de Jaccard ( $D_J$ ) es el complemento a 1 de la similaridad de J
#
(1 - mi_fam_jac) %>% tidy %>% rename(similaridad=distance) #Similaridad
#
# Dado que este resultado muestra la similaridad, podemos leerlo como "el sitio 1 y el 2 compar
#
# La fórmula de la similaridad de Jaccard es  $S_J = a/(a+b+c)$ , donde  $a$  es el núm
#
# Para obtener las variables  $a$ ,  $b$  y  $c$ , usaré La función 'betadiver' del paquete 've
#
mi_fam_abc <- betadiver(mc_myrtc)
mi_fam_abc %>%
  map(tidy) %>%
  map(slice, 1) %>%
  map_df(I, .id = 'tipo') %>%
  dplyr::select(tipo, n_especies=distance)
#
# Puedes notar que ambos sitios comparten 4 especies ( $a$ ), que el sitio 2 no tiene especies
#
round(4/5*100,2) #Porcentaje de especies compartidas = similaridad
#
# Con 'betadiver' también puedes calcular índices de similaridad. Por ejemplo, el Jaccard se ca
#
betadiver(mc_myrtc, method = 'j') %>% tidy
#
# No obstante, usaremos esta función en los análisis de diversidad beta más adelante.
#
# Además de la distancia de Jaccard, otra distancia muy utilizada es la de Sorensen o Bray-Curt
#
mi_fam_sor <- vegdist(mc_myrtc, method = 'bray', binary = T)
mi_fam_sor %>% tidy
coldiss(mi_fam_sor, diag = T)
#
# ### Modo Q para datos cuantitativos, NO de abundancia de especies (variables ambientales)
#
# En este ejemplo, usaré sólo variables de suelo, todas cuantitativas, puedes combinar con otra
#
env_suelo_punt_z <- bci_env_grid %>%
  st_drop_geometry() %>%
  dplyr::select(matches('^A-T,Z|^pH$', ignore.case = F)) %>%
  scale()
env_suelo_punt_z_d <- dist(env_suelo_punt_z)
env_suelo_punt_z_d %>% tidy
coldiss(env_suelo_punt_z_d, diag = T)
#
# ### Modo Q para datos cualitativos y cuantitativos (mixtos), NO de abundancia de especies (va
#
# En este ejemplo, usaré las siguientes variables mixtas (funciona igualmente para datos cualit

```

```

#’
#’ - ‘heterogeneidad_ambiental’. Índice cuantitativo calculado como la diversidad de Simpson a
#’
#’ - ‘habitat’. Tipo de hábitat. Asume los siguientes valores posibles: *OldHigh*, *OldLow* y *OldMed*
#’
#’ - ‘quebrada’. Informa sobre si hay o no quebrada. Los valores posibles son *Yes* o *No*.
#’
env_mix <- bci_env_grid %>%
  st_drop_geometry() %>%
  dplyr::select(heterogeneidad_ambiental, habitat, quebrada)
env_mix_d <- daisy(x = env_mix, metric = 'gower')
env_mix_d #>% as.dist %>% tidy
env_mix_d %>% coldiss(diag = T)

#’ ---
#’ title: "Medición de asociación. Modo R aplicado a mi familia asignada"
#’ author: "JR"
#’ date: "3 de noviembre, 2020"
#’ output: github_document
#’ ---

knitr::opts_chunk$set(fig.width=12, fig.height=8)

#’ ## Preámbulo

#’ ### Cargar paquetes
library(vegan)
library(adespatial)
library(broom)
library(tidyverse)
library(sf)
library(gclus)
source('biodata/funciones.R')

#’ ### Cargar datos
#’
load('biodata/matriz_ambiental.Rdata')
load('biodata/Myrtaceae.Rdata')
#’
#’ ## Modo R: matrices de dependencia entre variables (índice de correlación)
#’
#’ ### Modo R para datos cuantitativos de especies (abundancia)
#’
#’ En este caso, las variables usaré los valores de abundancias de especies como variables. Es d
#’
#’ Aunque se podría usar el índice de correlación como métrica de la dependencia (tal como mostr
#’
#’ Por tal razón, es recomendable aplicar la transformación *Chi* a la matriz de comunidad trans

```

```

#’
#’ Primero, sustituyo el caracter de espacio por un <enter> en los nombres de las especies (cara
#’
mi_fam_t <- mc_myrtc %>%
  rename_all(gsub, pattern = ' ', replacement = '\n') %>%
  t()
mi_fam_t %>% tibble
#’
#’ Segundo, transformo la matriz transpuesta usando estandarización *Chi*.
#’
mi_fam_t_chi <- decostand(mi_fam_t, "chi.square")
mi_fam_t_chi %>% tibble
#’
#’ Tercero, calculo la distancia euclídea.
#’
mi_fam_t_chi_d <- dist(mi_fam_t_chi)
mi_fam_t_chi_d %>% tidy
#’
#’ Finalmente, creo el "mapa de calor".
#’
coldiss(mi_fam_t_chi_d, diag = TRUE)
#’
#’ En el mapa de calor **ordenado** (el de la derecha), se identifica al menos un patrón de depe
#’
#’ ### Modo R para datos binarios (presencia/ausencia)
#’
#’ Arriba usé la distancia de Jaccard para evaluar asociación entre sitios. Dicha métrica tambié
#’
mi_fam_t_jac <- vegdist(mi_fam_t, "jaccard", binary = TRUE)
mi_fam_t_jac %>% tidy
coldiss(mi_fam_t_jac, diag = TRUE)
#’
#’ ### Modo R para datos cuantitativos, NO de abundancia de especies (variables ambientales)
#’
#’ En modo R evalúas asociación entre descriptores, es decir, entre variables. La métrica común
#’
#’ En este ejemplo, mostraré la correlación entre variables de suelo y la abundancia y riqueza g
#’
env_num <- bci_env_grid %>%
  dplyr::select_if(is.numeric) %>%
  dplyr::select(-id, -matches('^U.*')) %>%
  st_drop_geometry %>%
  mutate(
    riqueza_mifam = specnumber(mc_myrtc),
    abundancia_mifam = rowSums(mc_myrtc)) %>%
  rename_all(gsub, pattern = '_pct$', replacement = '') %>%
  rename_all(gsub, pattern = '_| ', replacement = '\n')
env_num %>% tibble

```

```

p_cor_suelo_ar <- env_num %>%
  dplyr::select(matches('^[A-T,Z]|abundancia|riqueza|^pH$', ignore.case = F)) %>%
  ezCorM(r_size_lims = c(4,8), label_size = 3, method = 'pearson')
p_cor_suelo_ar

p_cor_suelo_ar_spearman <- env_num %>%
  dplyr::select(matches('^[A-T,Z]|abundancia|riqueza|^pH$', ignore.case = F)) %>%
  ezCorM(r_size_lims = c(4,8), label_size = 3, method = 'spearman')
p_cor_suelo_ar_spearman

png(
  filename = 'matriz_correlacion_suelo_abun_riq_spearman.png',
  width = 1920, height = 1080, res = 125
)
p_cor_suelo_ar_spearman
dev.off() #NO OLVIDAR ESTA IMPORTANTE SENTENCIA

p_cor_geomorf_ar <- env_num %>%
  dplyr::select(-matches('^[A-T,Z]|pH$', ignore.case = F)) %>%
  ezCorM(r_size_lims = c(4,8), label_size = 3, method = 'pearson')
p_cor_geomorf_ar

p_cor_geomorf_ar_spearman <- env_num %>%
  dplyr::select(-matches('^[A-T,Z]|pH$', ignore.case = F)) %>%
  ezCorM(r_size_lims = c(4,8), label_size = 3, method = 'spearman')
p_cor_geomorf_ar_spearman

png(
  filename = 'matriz_correlacion_geomorf_abun_riq_spearman.png',
  width = 1920, height = 1080, res = 110
)
p_cor_geomorf_ar_spearman
dev.off() #NO OLVIDAR ESTA IMPORTANTE SENTENCIA

```

8.3 *Análisis de agrupamiento (cluster analysis)*

```

#' ---
#' title: "Análisis de agrupamiento (cluster analysis). <br> Parte 1: agrupamiento jerárquico"
#' author: "JR"
#' date: "11 de noviembre, 2020"
#' output: github_document
#' ---

knitr::opts_chunk$set(fig.width=12, fig.height=8)

#' ## Preámbulo

```

```

#' ### Cargar paquetes
library(vegan)
library(magrittr)
library(broom)
source('biodata/funciones.R')

#' ### Cargar datos
#'
load('biodata/Myrtaceae.Rdata')
mi_fam <- mc_myrtc
#'
#' ## Características de las técnicas de agrupamiento
#'
#' Las técnicas de agrupamiento se clasifican según los algoritmos que emplean y el orden de eje
#'
#' - Secuenciales o simultáneos.
#' - Por aglomeración o por división. En referencias en español encontrarás "aglomerativos" y "d
#' - Monotéticos o politéticos.
#' - Jerárquicos o no jerárquicos.
#' - Probabilísticos o no probabilísticos.
#' - Restringidos o no restringidos.
#'
#' ## Agrupamiento jerárquico
#'
#' El agrupamiento jerárquico (AJ) es una técnica de agrupamiento secuencial que consiste en la
#'
#' Dentro del AJ es frecuente usar un enfoque aglomerativo, lo cual implica aplicar algoritmos s
#'
#' El AJ aglomerativo dispone de varios algoritmos de resolución del agrupamiento por pares, que
#'
#' Normalmente, en el análisis de agrupamiento nos interesa agrupar sitios en función de sus des
#'
#' Dado que los cuadros en BCI están autocorrelacionados espacialmente, violamos el supuesto de
#'
#' ### Agrupamiento "aglomerativo" por enlace simple
#'
#' Este método utiliza, como criterio de enlace para agrupar sucesivamente pares de objetos, la
#'
#' Para aplicar este método, debes transformar la matriz de comunidad utilizando alguno de los m
#'
mi_fam_norm <- decostand(mi_fam, "normalize")
mi_fam_norm_d <- vegdist(mi_fam_norm, "euc")
mi_fam_norm_d %>% tidy
#'
#' Es importante, para garantizar consistencia a lo largo del agrupamiento, asignar los nombres
#'
attr(mi_fam_norm_d, "labels") <- rownames(mi_fam)

```



```

#’
#’ Posteriormente, el agrupamiento jerárquico lo realizaré con la función ‘hclust’ del paquete ‘
#’
(cl_single <- hclust(mi_fam_norm_d, method = ‘single’))
#’
#’ Finalmente, el dendrograma a continuación:
plot(cl_single, labels = rownames(mi_fam), hang = -1,
     main = "Sitios de BCI según composición de especies de Myrtaceae\nEnlace simple a partir de
     xlab = ‘Sitios’, ylab = ‘Altura’)
#’
#’ ### Agrupamiento "aglomerativo" por enlace completo
#’
#’ En este caso, el criterio de enlace para agrupar sucesivamente pares de objetos es la menor s
#’
(cl_complete <- hclust(mi_fam_norm_d, method = ‘complete’))
plot(cl_complete, labels = rownames(mi_fam), hang = -1,
     main = "Sitios de BCI según composición de especies de Myrtaceae\nEnlace completo a partir
     xlab = ‘Sitios’, ylab = ‘Altura’)
#’
#’ ### Agrupamiento "aglomerativo" por enlace promedio
#’
#’ En este caso, el criterio de enlace para agrupar sucesivamente pares de objetos es el promedi
#’
#’ Así, dependiendo de si se media o centroide, o si se ponderan o no las distancias originales,
#’
#’ Sólo crearé el dendrograma del método UPGMA.
#’
(cl_upgma <- hclust(mi_fam_norm_d, method = ‘average’))
plot(cl_upgma, labels = rownames(mi_fam), hang = -1,
     main = "Sitios de BCI según composición de especies de Myrtaceae\nUPGMA a partir de matriz
     xlab = ‘Sitios’, ylab = ‘Altura’)
#’
#’ ### Agrupamiento por el método de Ward de varianza mínima
#’
#’ Se basa en los mismos supuestos y criterios de la regresión lineal por mínimos cuadrados, sim
#’
(cl_ward <- hclust(mi_fam_norm_d, method = ‘ward.D2’))
plot(cl_ward, labels = rownames(mi_fam), hang = -1,
     main = "Sitios de BCI según composición de especies de Myrtaceae\nMétodo de Ward a partir d
     xlab = ‘Sitios’, ylab = ‘Altura’)

#’ ---
#’ title: "Análisis de agrupamiento (cluster analysis). <br> Parte 2: Interpretación y comparaci
#’ author: "JR"
#’ date: "11 de noviembre, 2020"
#’ output: github_document
#’ ---

```

```

knitr::opts_chunk$set(fig.width=12, fig.height=8)

#' ## Preámbulo
#'
#' ### Cargar paquetes
#'
library(vegan)
library(tidyverse)
library(broom)
library(cluster)
library(gclus)
library(pvclust)
library(sf)
source('biodata/funciones.R')
#'
#' ### Cargar datos
#'
load('biodata/Myrtaceae.Rdata')
mi_fam <- mc_myrtc
load('biodata/matriz_ambiental.Rdata')
mi_fam %>% tibble
bci_env_grid %>% tibble
#'
#' ### Generar matriz de distancias de cuerdas
#'
mi_fam_norm <- decostand(mi_fam, "normalize")
mi_fam_norm_d <- vegdist(mi_fam_norm, "euc")
mi_fam_norm_d %>% tidy
#'
#' ## Interpretación visual de dendrogramas
#'
#' [En el script anterior](aa_analisis_de_agrupamiento_1_jerarquico.md) realicé los dendrogramas
#'
#' La primera evaluación de los dendrogramas NO debe venir de la mano de sofisticados análisis n
#'
#' Evita concentrar tu vista en grupos extremadamente pequeños; comienza analizando el árbol des
#'
#' Anota tus impresiones, para que las compares con los resultados que posteriormente obtendrás;
#'
#' Para la exploración visual, generaré los objetos de cluster dentro de una lista:
#'
lista_cl <- list(
  cl_single = hclust(mi_fam_norm_d, method = 'single'),
  cl_complete = hclust(mi_fam_norm_d, method = 'complete'),
  cl_upgma = hclust(mi_fam_norm_d, method = 'average'),
  cl_ward = hclust(mi_fam_norm_d, method = 'ward.D2')
)
#'

```

```

#’ Un plot en panel 2x2 ayuda a visualizarlos todos de manera conjunta. En tu caso, observa y co
#’
par(mfrow = c(2,2))
invisible(map(names(lista_cl), function(x) plot(lista_cl[[x]], main = x, hang = -1)))
par(mfrow = c(1,1))
#’
#’ En mi caso, exceptuando el dendrograma generado por medio del enlace simple, detecto al menos
#’
#’ - Un grupo pequeño, compuesto por los sitios 14 y 19.
#’ - Un "grupo" heterogéneo y grande, conformado por el resto de sitios.
#’
#’ Además de los grupos anteriores, detecto elementos que no forman grupos, es decir, sitios que
#’
#’ ## Elegir método y número de clústers
#’
#’ Existen varios criterios para elegir un dendrograma idóneo, como por ejemplo, los gráficos ti
#’
#’ ### Seleccionar método de agrupamiento por correlación cofenética
#’
#’ La correlación cofenética implica conocer la distancia cofenética, y esta última se entiende m
#’
#’ Usando la lista de objetos de clústers, calcularé la correlación cofenética dentro de un ‘map
#’
map_df(lista_cl, function(x) {
  coph_d <- cophenetic(x)
  corr <- cor(mi_fam_norm_d, coph_d)
  return(corr)
})
#’
#’ Habrás notado que, tanto UPGMA como enlace completo, tienen valores altos de correlación cofe
#’
#’ ### Elegir número de clústers
#’
#’ Elegiré UPGMA como método de agrupamiento y determinaré cuántos grupos son idóneos de acuerdo
#’
#’ El objetivo de la función ‘calcular anchuras siluetas’ está implícito en su nombre, y requiere
#’
#’ 1. Las anchuras promedio para cada partición, excepto para las particiones ‘i=1’ y ‘i=50’, po
#’
#’ 2. Número óptimo de grupos. Haré los cálculos para UPGMA y Ward, y luego explico en qué consi
#’
#’ Para UPGMA:
#’
anch_sil_upgma <- calcular_anchuras_siluetas(
  mc_orig = mi_fam,
  distancias = mi_fam_norm_d,
  cluster = lista_cl$cl_upgma)
anch_sil_upgma

```

```

#’
#’ El objeto ‘anchuras_siluetas’ de la lista ‘anch_sil_upgma’ te muestra un vector con los prome
#’
#’ Igualmente, el objeto ‘n_grupos_optimo’ te indica cuál es el número óptimo de clústers a crea
#’
#’ Haré el gráfico de dendrograma, aunque nota que en este caso primero reordenaré los sitios co
#’
u_dend_reord <- reorder.hclust(lista_cl$cl_upgma, mi_fam_norm_d)
plot(u_dend_reord, hang = -1)
rect.hclust(
  tree = u_dend_reord,
  k = anch_sil_upgma$n_grupos_optimo)
#’
#’ Ahora compararé el dendrograma con el mapa de calor en un mismo gráfico, colocando los dendro
#’
heatmap(
  as.matrix(mi_fam_norm_d),
  Rowv = as.dendrogram(u_dend_reord),
  symm = TRUE,
  margin = c(3, 3),
  col = rev(cm.colors(4))
)
#’
#’ En general, hay dos grupos, uno grande y otro pequeño, y parece haber un tercero en el mapa d
#’
#’ Mostraré el resultado para Ward:
#’
anch_sil_ward <- calcular_anchuras_siluetas(
  mc_orig = mi_fam,
  distancias = mi_fam_norm_d,
  cluster = lista_cl$cl_ward)
anch_sil_ward
#’
#’ En este caso, el valor máximo, que ocupa la posición número 3, no se diferencia mucho del de
#’
w_dend_reord <- reorder.hclust(lista_cl$cl_ward, mi_fam_norm_d)
plot(w_dend_reord, hang = -1)
rect.hclust(
  tree = w_dend_reord,
  k = 3) # anch_sil_ward$n_grupos_optimo)
plot(w_dend_reord, hang = -1)
rect.hclust(
  tree = w_dend_reord,
  k = 4) #anch_sil_ward$n_grupos_optimo + 1)
#’
#’ Comparando el dendrograma con el mapa de calor. Verificar si el número de grupos hace sentido
#’
heatmap(

```

```

as.matrix(mi_fam_norm_d),
Rowv = as.dendrogram(w_dend_reord),
symm = TRUE,
margin = c(3, 3),
col = rev(cm.colors(4))
)
#'
#' Nótese que este dendrograma hace más sentido que el sugerido por UPGMA. En cualquier casos, c
#'
#' ### Evaluación mediante remuestreo por *bootstrap* multiescalar
#'
#' Con suerte, un agrupamiento aplicado a datos muestrales reflejará los patrones naturales de o
#'
#' Sin embargo, los datos de BCI también tienen sesgo, pues se usa un DAP de corte para decidir
#'
#' No obstante, aun con todas sus bondades, los datos censales carecen de una fortaleza: no refl
#'
#' Por estas razones, los análisis de agrupamientos realizados hasta este punto, reflejan tanto
#'
#' El método de remuestreo por *bootstrap* multiescalar está implementado en el paquete 'pvclust'
#'
#' La función 'pvclust' devolverá un dendrograma enriquecido, que incluirá los valores de AU y BP
#'
#' - Rectángulos de borde azul, para todos aquellos grupos que resulten con valores de AU>0.91 e
#'
#' - Líneas inferiores rojas, que resaltan aquellos grupos (o subgrupos) que obtuvieron AU>0.95.
#'
#' Ten presente que, al realizar remuestreo por *bootstrap* multiescalar, cada corrida puede arr
#'
#' #### UPGMA
#'
cl_pvclust_upgma <-
  pvclust(t(mi_fam_norm),
    method.hclust = "average",
    method.dist = "euc",
    iseed = 91, # Resultado reproducible
    parallel = TRUE)
# Añadir los valores de p
plot(cl_pvclust_upgma, hang = -1)
# Añadir rectángulos a los grupos significativos
lines(cl_pvclust_upgma)
pvrect(cl_pvclust_upgma, alpha = 0.91, border = 4)
#'
#' #### Ward
#'
cl_pvclust_ward <-
  pvclust(t(mi_fam_norm),
    method.hclust = "ward.D2",

```

```

        method.dist = "euc",
        #isseed = 191, # Resultado reproducible
        parallel = TRUE)
# Añadir los valores de p
plot(cl_pvclust_ward, hang = -1)
# Añadir rectángulos a los grupos significativos
lines(cl_pvclust_ward)
pvrect(cl_pvclust_ward, alpha = 0.91, border = 4)
#'
#' ### Recapitulando los grupos de sitios.
#'
#' #### Patrones comunes y dispares
#'
#' Detecto algunos patrones consistentes en cuanto a grupos de sitios según composición de las e
#'
#' - Tanto en UPGMA como en Ward, detecté al menos dos o tres grandes grupos. Con el primer méto
#'
#' - En el caso específico del dendrograma Ward, 'pvclust' atomizó los sitios en demasiados grup
#'
#' #### ¿Cómo declaro los grupos de sitios?
#'
#' Para conservar las clasificaciones de grupos de sitios anteriores, crearé un vector con el id
#'
#' UPGMA:
(grupos_upgma_k2 <- as.factor(cutree(lista_cl$cl_upgma, k = 2)))
#'
#' En este caso, los sitios del 1 al 13 pertenecen al grupo 1, los sitios 14 y 19 pertenecen al
#'
table(grupos_upgma_k2)
#'
#' Nota lo desiguales que son estos grupos, un efecto esperado dado el alto grado de autocorrela
#'
#' Ward:
#'
(grupos_ward_k3 <- as.factor(cutree(lista_cl$cl_ward, k = 3)))
table(grupos_ward_k3)
#'
#' Guardaré estos vectores en archivos para reutilizarlos en *scripts* posteriores:
#'
saveRDS(grupos_upgma_k2, 'grupos_upgma_k2.RDS')
saveRDS(grupos_ward_k3, 'grupos_ward_k3.RDS')
#'
#' Evita usar este, y cualquier otro procedimiento, de manera mecánica. En tu caso, quizá tengas
#'
#' En el próximo *script*, aprenderás a comparar este resultado con las variables ambientales. T
#'
#' ---
#' title: "Análisis de agrupamiento (cluster analysis). <br> Parte 3: Grupos (clústers), variabl

```

```

#' author: "JR"
#' date: "15 de noviembre, 2020"
#' output: github_document
#' ---

knitr::opts_chunk$set(fig.width=12, fig.height=8)

#' ## Preámbulo
#'
#' ### Cargar paquetes
#'
library(mapview)
library(tidyverse)
library(sf)
library(RColorBrewer)
source('biodata/funciones.R')
#'
#' ### Cargar datos
#'
load('biodata/Myrtaceae.Rdata')
load('biodata/matriz_ambiental.Rdata')
grupos_upgma_k2 <- readRDS('grupos_upgma_k2.RDS')
table(grupos_upgma_k2) #Importante, tener en cuenta los desiguales tamaños de los grupos
grupos_ward_k3 <- readRDS('grupos_ward_k3.RDS')
table(grupos_ward_k3)
#'
#' ### Paletas
#'
rojo <- colorRampPalette(brewer.pal(8, "Reds"))
rojo_inv <- colorRampPalette(rev(brewer.pal(8, "Reds")))
colores_grupos <- brewer.pal(8, "Accent")
#'
#' ## Explorar efectos
#'
#' ### Pruebas de igualdad de promedios de las variables entre 2 grupos
#'
#' Para evaluar homogeneidad de promedios usaré las pruebas *t* (medias), basada en la distribuc
#'
#' Primero crearé un objeto que permita realizar tanto las pruebas como los diagramas de cajas.
#'
(m_amb_upgma_k2 <- bci_env_grid %>%
  select_if(is.numeric) %>% select(-id) %>%
  mutate(grupos_upgma_k2) %>%
  st_drop_geometry() %>%
  pivot_longer(-grupos_upgma_k2, names_to = "variable", values_to = "valor"))
#'
#' A continuación, las pruebas:
#'

```

```

m_amb_upgma_k2 %>%
  group_by(variable) %>%
  summarise(
    p_valor_t = t.test(valor ~ grupos_upgma_k2)$p.value,
    p_valor_w = wilcox.test(valor ~ grupos_upgma_k2, exact = F)$p.value) %>%
  arrange(p_valor_t) %>%
  print(n=Inf)
#'
#' Interesa observar las variables que obtuvieron valores de  $p < 0.01$ . Reitero que, en mi caso, mi
#'
#' Por otra parte, este es un buen momento para "revisitar" tus análisis exploratorios de datos
#'
#' Los gráficos:
#'
m_amb_upgma_k2 %>%
  group_by(variable) %>%
  ggplot() + aes(x = grupos_upgma_k2, y = valor, fill = grupos_upgma_k2) +
  geom_boxplot() +
  scale_fill_brewer(palette = 'Accent') +
  theme_bw() +
  theme(legend.position="none") +
  facet_wrap(~ variable, scales = 'free_y')
#'
#' Mapas:
#'
mapa_upgma_k2 <- mapView(
  bci_env_grid %>% mutate(grupos_upgma_k2),
  layer.name = 'Grupos (2) UPGMA',
  alpha.regions = 0.6,
  map.types = 'OpenTopoMap',
  legend = T,
  col.regions = colores_grupos[1:2],
  zcol = 'grupos_upgma_k2') %>%
  addStaticLabels(label = bci_env_grid$id) %>%
  leaflet::setView(
    lng = -79.85136,
    lat = 9.15097,
    zoom = 15)
mapa_upgma_k2
mapa_upgma_k2 %>% mapshot(
  file = 'mapa_upgma_k2.png',
  remove_controls = c("zoomControl", "layersControl", "homeButton")
)
#'
#' Mapa de una de las variables donde se presentó efecto de su promedio ( $p < 0.01$ ), en este caso,
#'
mapa_zn <- mapView(
  bci_env_grid,

```



```

layer.name = 'Zinc',
alpha.regions = 0.6,
map.types = 'OpenTopoMap',
legend = T,
col.regions = rojo,
zcol = 'Zn') %>%
addStaticLabels(label = bci_env_grid$id) %>%
leaflet::setView(
  lng = -79.85136,
  lat = 9.15097,
  zoom = 15)
mapa_zn
mapa_zn %>% mapshot(
  file = 'mapa_zinc.png',
  remove_controls = c("zoomControl", "layersControl", "homeButton")
)
#'
#' ### Pruebas de igualdad de promedios de las variables entre 3 grupos o más
#'
#' Objeto común:
#'
(m_amb_ward_k3 <- bci_env_grid %>%
  select_if(is.numeric) %>% select(-id) %>%
  mutate(grupos_ward_k3) %>%
  st_drop_geometry() %>%
  pivot_longer(-grupos_ward_k3, names_to = "variable", values_to = "valor"))
#'
#' Pruebas, en este caso ANOVA (evalúa homogeneidad de medias; no se cumplen muchos de los supue
#'
m_amb_ward_k3 %>%
  group_by(variable) %>%
  summarise(
    p_valor_a = oneway.test(valor ~ grupos_ward_k3)$p.value,
    p_valor_k = kruskal.test(valor ~ grupos_ward_k3)$p.value) %>%
  arrange(p_valor_k) %>%
  print(n=Inf)
#'
#' Gráficos:
#'
m_amb_ward_k3 %>%
  group_by(variable) %>%
  ggplot() + aes(x = grupos_ward_k3, y = valor, fill = grupos_ward_k3) +
  geom_boxplot() +
  scale_fill_brewer(palette = 'Accent') +
  theme_bw() +
  theme(legend.position="none") +
  facet_wrap(~ variable, scales = 'free_y')
#'

```

```

#’ Mapas:
#’
mapa_ward_k3 <- mapView(
  bci_env_grid %>% mutate(grupos_ward_k3),
  layer.name = 'Grupos (3) Ward',
  alpha.regions = 0.6,
  map.types = 'OpenTopoMap',
  legend = T,
  col.regions = colores_grupos[1:3],
  zcol = 'grupos_ward_k3') %>%
  addStaticLabels(label = bci_env_grid$id) %>%
  leaflet::setView(
    lng = -79.85136,
    lat = 9.15097,
    zoom = 15)
mapa_ward_k3
mapa_ward_k3 %>% mapshot(
  file = 'mapa_ward_k3.png',
  remove_controls = c("zoomControl", "layersControl", "homeButton")
)
#’
#’ Mapa de una de las variables donde se presentó efecto de su promedio ( $p < 0.01$ ), en este caso,
#’
mapa_ph <- mapView(
  bci_env_grid,
  layer.name = 'pH',
  alpha.regions = 0.6,
  map.types = 'OpenTopoMap',
  legend = T,
  col.regions = rojo_inv,
  zcol = 'pH') %>%
  addStaticLabels(label = bci_env_grid$id) %>%
  leaflet::setView(
    lng = -79.85136,
    lat = 9.15097,
    zoom = 15)
mapa_ph
mapa_ph %>% mapshot(
  file = 'mapa_ph.png',
  remove_controls = c("zoomControl", "layersControl", "homeButton")
)
#’
#’ ---
#’ title: "Análisis de agrupamiento (cluster analysis). <br> Parte 4: Especies indicadoras, espe
#’ author: "JR"
#’ date: "15 de noviembre, 2020"
#’ output: github_document

```

```

#' ---

knitr::opts_chunk$set(fig.width=12, fig.height=8)

#' ## Preámbulo
#'
#' ### Cargar paquetes
#'
library(indicspecies)
source('biodata/funciones.R')
#'
#' ### Cargar datos
#'
load('biodata/Myrtaceae.Rdata')
mi_fam <- mc_myrtc
grupos_upgma_k2 <- readRDS('grupos_upgma_k2.RDS')
table(grupos_upgma_k2)
grupos_ward_k3 <- readRDS('grupos_ward_k3.RDS')
table(grupos_ward_k3)
#'
#' ## Análisis de especies indicadoras mediante IndVal
#'
#' ### UPGMA
#'
iva_upgma_k2 <- multipatt(
  x = mi_fam,
  cluster = grupos_upgma_k2,
  func = 'IndVal.g',
  max.order = 1,
  control = how(nperm = 999))
summary(iva_upgma_k2, indvalcomp = TRUE)
colSums(mi_fam)
(p_upgma_adj <- p.adjust(iva_upgma_k2$sign$p.value))
(iva_upgma_boot <- strassoc(
  X = mi_fam,
  cluster = grupos_upgma_k2,
  func = "IndVal.g",
  nboot = 1000))
#'
#' Ward
#'
iva_ward_k3 <- multipatt(
  x = mi_fam,
  cluster = grupos_ward_k3,
  func = 'IndVal.g',
  max.order = 2,
  control = how(nperm = 999))
summary(iva_ward_k3, indvalcomp = TRUE)

```

```

colSums(mi_fam)
(p_ward_adj <- p.adjust(iva_ward_k3$sign$p.value))
(iva_ward_boot <- strassoc(
  X = mi_fam,
  cluster = grupos_ward_k3,
  func = "IndVal.g",
  nboot = 1000))
#'
#' ## Análisis de especies con preferencia por hábitat mediante el coeficiente de correlación bi
#'
#' ### UPGMA
#'
phi_upgma_k2 <- multipatt(
  mi_fam,
  grupos_upgma_k2,
  func = "r.g",
  max.order = 1,
  control = how(nperm = 999))
summary(phi_upgma_k2)
colSums(mi_fam)
(phi_upgma_boot <- strassoc(
  X = mi_fam,
  cluster = grupos_upgma_k2,
  func = "r.g",
  nboot = 1000))
#'
#' Ward
#'
phi_ward_k3 <- multipatt(
  mi_fam,
  grupos_ward_k3,
  func = "r.g",
  max.order = 2,
  control = how(nperm = 999))
summary(phi_ward_k3)
colSums(mi_fam)
(phi_ward_boot <- strassoc(
  X = mi_fam,
  cluster = grupos_ward_k3,
  func = "r.g",
  nboot = 1000))

```

8.4 Análisis de ordenación

```

#' ---
#' title: "Técnicas de ordenación. <br> Parte 1: Ordenación no restringida. <br> PCA, CA y PCoA"
#' author: "JR"

```

```

#' date: "21 de noviembre, 2020"
#' output: github_document
#'
#' ---

knitr::opts_chunk$set(fig.width=12, fig.height=8)

#' ## Preámbulo
#'
#' ### Cargar paquetes
#'
library(vegan)
library(tidyverse)
library(sf)
library(mapview)
source('biodata/funciones.R')
#'
#' ### Cargar datos
#'
load('biodata/Myrtaceae.Rdata')
load('biodata/matriz_ambiental.Rdata')
mi_fam <- mc_myrtc
(colnames(mi_fam) <- make.cepnames(colnames(mi_fam)))
(df_equivencias <- data.frame(
  nombre_original = colnames(mc_myrtc),
  colnames(mi_fam)))
bci_env_grid %>% tibble
grupos_upgma_k2 <- readRDS('grupos_upgma_k2.RDS')
table(grupos_upgma_k2)
grupos_ward_k3 <- readRDS('grupos_ward_k3.RDS')
table(grupos_ward_k3)
#'
#' ## Ordenación
#'
#' La ordenación se basa en los mismos principios que la medición de asociación (similaridad) y
#'
#' A diferencia del análisis de agrupamiento, o como complemento de éste, el análisis de ordenación
#'
#' El análisis de ordenación puede ser no restringido (o simple) y restringido (o 'canónico'). En
#'
#' Las principales técnicas de ordenación no restringida son análisis de componentes principales
#'
#' ### Análisis de componentes principales (PCA)
#'
#' Es el método tradicional basado en vectores propios que comúnmente se aplica a datos cuantitativos
#'
#' #### PCA aplicado a datos ambientales
#'

```

```

#’ Para aplicar PCA a datos ambientales, es necesario que todas las variables sean numéricas y "
#’
#’ Dado que se requiere que las variables de entrada sean exclusivamente numéricas, el primer pa
#’
#’ ¡IMPORTANTE! Haré esta demostración sólo con las variables de suelo, **pero puedes (y debes)
#’
#’ A partir de los datos de suelo, la función ‘rda’, de ‘vegan’ realizará los siguientes pasos:
#’
env_suelo <- bci_env_grid %>%
  st_drop_geometry %>%
  dplyr::select(matches('^[A-T,Z]|~pH$'), ignore.case = F))
env_suelo %>% tibble
env_suelo_pca <- rda(env_suelo, scale = TRUE)
env_suelo_pca
summary(env_suelo_pca)
#’
#’ Para agilizar la producción de scripts analíticos de referencia, trasladaré las explicaciones
#’
#’ En el vídeo asociado, explico el significado de:
#’
#’ - Inercia, *Inertia*
#’ - Valores propios, autovalores, *Eigenvalues*
#’ - Escalamiento, *Scaling*
#’ - Puntuaciones de "especies", *Species scores*
#’ - Puntuaciones de "sitios", *Site scores*
#’
screeplot(env_suelo_pca, bstick = TRUE)
#’
#’ Usando función ‘cleanplot.pca’
#’
par(mfrow = c(1, 2))
cleanplot.pca(env_suelo_pca, scaling = 1, mar.percent = 0.08, cex.char1 = 1.5)
cleanplot.pca(env_suelo_pca, scaling = 2, mar.percent = 0.04, cex.char1 = 1.5)
par(mfrow = c(1, 1))
#’
#’ Comparar distribución de los sitios en biplots con distribución real en el mapa:
#’
#’ ### Generar mapa de cuadros sin simbología
#’
mapa_cuadros <- mapView(
  bci_env_grid,
  col.regions = 'grey80',
  alpha.regions = 0.3,
  map.types = 'OpenTopoMap',
  legend = F, zoom = 14,
  zcol = 'id') %>% addStaticLabels() %>%
  leaflet::setView(
    lng = -79.85136,

```

```

        lat = 9.15097,
        zoom = 15)
mapa_cuadros
#’
#’ Comparar con resultados de un análisis de agrupamiento del mismo conjunto de datos. Primero a
#’
(env_agrupamiento <- hclust(dist(scale(env_suelo)), 'ward.D'))
(env_grupos <- cutree(env_agrupamiento, k = 3))
(mi_cluster <- factor(env_grupos))
(mi_cluster_l <- levels(mi_cluster))
(mi_cluster_l_seq <- 1:length(mi_cluster_l))
#’
#’ Observa que estoy generando un agrupamiento basado en los datos de suelo. No estoy comparando
#’
#’ Luego calculo las puntuaciones de los sitios para usarlas luego como coordenadas de los puntos
#’
(puntuaciones <- scores(env_suelo_pca, display = 'wa', scaling = 1))
#’
#’ Luego creo el gráfico base, coloco los puntos sobre el gráfico usando las puntuaciones, les c
#’
grafico_base <- plot(
  env_suelo_pca,
  display = "wa",
  scaling = 1,
  type = "n",
  main = "PCA y grupos"
)
abline(v = 0, lty = "dotted")
abline(h = 0, lty = "dotted")
for (i in mi_cluster_l_seq) {
  points(puntuaciones[mi_cluster == i, ],
        pch = (14 + i),
        cex = 2,
        col = i + 1)
}
text(puntuaciones, row.names(env_suelo), cex = 1, pos = 3)
legend(
  "topright", # Otras alternativas: "bottomleft", "bottomright" y "topleft"
  paste("Grupo", c(mi_cluster_l_seq)),
  pch = 14 + c(mi_cluster_l_seq),
  col = 1 + c(mi_cluster_l_seq),
  pt.cex = 2
)
#’
#’ Es razonable que el análisis cluster y el biplot muestren patrones consistentes, puesto que s
#’
#’ Si hago lo mismo, pero usando mi análisis de agrupamiento anterior (*scripts* "aa_analisis_de
#’

```

```

# (mi_cluster_anterior <- grupos_upgma_k2)
(mi_cluster_anterior <- grupos_ward_k3)
(mi_cluster_anterior_l <- levels(mi_cluster_anterior))
(mi_cluster_anterior_l_seq <- 1:length(mi_cluster_anterior_l))
grafico_base <- plot(
  env_suelo_pca,
  display = "wa",
  scaling = 1,
  type = "n",
  main = "PCA y grupos"
)
abline(v = 0, lty = "dotted")
abline(h = 0, lty = "dotted")
for (i in mi_cluster_anterior_l_seq) {
  points(puntuaciones[mi_cluster_anterior == i, ],
    pch = (14 + i),
    cex = 2,
    col = i + 1)
}
text(puntuaciones, row.names(env_suelo), cex = 1, pos = 3)
legend(
  "topright", # Otras alternativas: "bottomleft", "bottomright" y "topleft"
  paste("Grupo", c(mi_cluster_anterior_l_seq)),
  pch = 14 + c(mi_cluster_anterior_l_seq),
  col = 1 + c(mi_cluster_anterior_l_seq),
  pt.cex = 2
)
#'
#' Esto podría significar que las tendencias/patrones de mi matriz de comunidad (cuadros de 1 Ha
#'
#' ##### PCA aplicado a datos de comunidad transformados
#'
mi_fam_hel <- decostand(mi_fam, method = 'hellinger')
mi_fam_hel %>% tibble
mi_fam_hel_pca <- rda(mi_fam_hel)
summary(mi_fam_hel_pca)
screeplot(
  mi_fam_hel_pca,
  bstick = TRUE,
  npcs = length(mi_fam_hel_pca$CA$eig)
)
mi_fam_hel_pca_sc1 <- scores(mi_fam_hel_pca,
  display = "species", scaling = 1)
mi_fam_hel_pca_sc2 <- scores(mi_fam_hel_pca,
  display = "species", scaling = 2)
par(mfrow = c(1, 2))
cleanplot.pca(mi_fam_hel_pca, scaling = 1, mar.percent = 0.06, cex.char1 = 0.7)
cleanplot.pca(mi_fam_hel_pca, scaling = 2, mar.percent = 0.06, cex.char1 = 0.7)

```



```

par(mfrow = c(1, 1))
#'
#' Si intentáramos realizar el PCA a datos de comunidad no transformados, no recogeríamos apropi
#'
#' Las especies que contribuyen mucho a los ejes 1 y 2 del PCA (aquellas cuyos vectores sobresal
#'
#' Evaluaré el ajuste del PCA de datos de comunidad a datos ambientales, mediante la función 'en
#'
biplot(
  mi_fam_hel_pca,
  main = "PCA, escalamiento 2, ajuste a variables ambientales")
(mi_fam_hel_pca_envfit <- envfit(mi_fam_hel_pca, env_suelo, scaling = 2))
plot(mi_fam_hel_pca_envfit, p.max = 0.05 , col = 3)
#'
#' Comento los resultados en el vídeo asociado. También probaré ajuste con todas las numéricas d
#'
#' NOTA: te recomiendo probar otros métodos de selección de variables, como por ejemplo, usando
#'
env_num <- bci_env_grid %>%
  select_if(is.numeric) %>%
  select(-id) %>%
  st_drop_geometry
(mi_fam_hel_pca_envfit_num <- envfit(mi_fam_hel_pca, env_num, scaling = 2))
biplot(
  mi_fam_hel_pca,
  main = "PCA, escalamiento 2, ajuste a variables ambientales")
plot(mi_fam_hel_pca_envfit_num, p.max = 0.05 , col = 3)
biplot(
  mi_fam_hel_pca,
  main = "PCA, escalamiento 2, ajuste a variables ambientales")
plot(mi_fam_hel_pca_envfit_num, p.max = 0.1 , col = 3)
#'
#' Comento los resultados en el vídeo asociado.
#'
#' ¿Cuándo o a qué datos aplicar PCA?
#'
#' - PCA no es especialmente sensible a datos muy desviados de la normalidad.
#' - Como toda técnica, PCA tiene limitaciones.
#' - Las variables deben ser dimensionalmente homogéneas (unidades comparables o adimensionales)
#' - No usar en matriz transpuestas (no hace sentido la covarianza entre objetos).
#' - Es posible usar PCA con dato de presencia/ausencia, en cuyo caso, la matriz de comunidad de
#' - Las relaciones entre variables se miden por ángulos, no por proximidad de las puntas de los
#'
#' ### Análisis de correspondencia (CA)
#'
mi_fam_ca <- cca(mi_fam)
summary(mi_fam_ca)
summary(mi_fam_ca, scaling = 1)

```

```

#’
#’ Screeplot
#’
screeplot(mi_fam_ca, bstick = TRUE, npcs = length(mi_fam_ca$CA$eig))
#’
#’ Biplots
#’
par(mfrow = c(1, 2))
plot(mi_fam_ca,
     scaling = 1,
     main = "Análisis de correspondencia, escalamiento 1"
)
plot(mi_fam_ca,
     scaling = 2, # Por defecto scaling=2, lo escribo sólo para fines didáticos
     main = "Análisis de correspondencia, escalamiento 2")
par(mfrow = c(1, 1))
#’
#’ Excluyendo especie *Chamguava schippii*, abreviada como *Chamschi*.
#’
mi_fam_ca <- cca(mi_fam[, -grep('Chamschi', colnames(mi_fam))])
summary(mi_fam_ca)
summary(mi_fam_ca, scaling = 1)
screeplot(mi_fam_ca, bstick = TRUE, npcs = length(mi_fam_ca$CA$eig))
par(mfrow = c(1, 2))
plot(mi_fam_ca,
     scaling = 1,
     main = "CA, escalamiento 1, sin Chamguava schippii"
)
plot(mi_fam_ca,
     scaling = 2,
     main = "CA, escalamiento 2, sin Chamguava schippii")
par(mfrow = c(1, 1))
#’
#’ #’ Excluyendo especie *Myrcia gatunensis*, abreviada como *Myrcgatu*.
#’
mi_fam_ca <- cca(mi_fam[, -grep('Myrcgatu', colnames(mi_fam))])
summary(mi_fam_ca)
summary(mi_fam_ca, scaling = 1)
screeplot(mi_fam_ca, bstick = TRUE, npcs = length(mi_fam_ca$CA$eig))
par(mfrow = c(1, 2))
plot(mi_fam_ca,
     scaling = 1,
     main = "CA, escalamiento 1, sin Myrcia gatunensis"
)
plot(mi_fam_ca,
     scaling = 2,
     main = "CA, escalamiento 2, sin Myrcia gatunensis")
par(mfrow = c(1, 1))

```

```

#’
#’ Análisis de coordenadas principales (PCoA)
#’
#’ Las técnicas de ordenación anteriores preservan la distancia euclídea entre los objetos. Si n
#’
#’ La función que realiza el PCoA en ‘{vegan}’ es ‘cmdscale’ (de *Classical (Metric) Multidimens
#’
mi_fam_d_bray <- vegdist(mi_fam, method = 'bray') # En realidad, 'bray' es la opción por defecto
mi_fam_d_bray_pcoa <- cmdscale(
  mi_fam_d_bray,
  k = (nrow(mi_fam) - 1),
  add = T,
  eig = TRUE)
round(mi_fam_d_bray_pcoa$eig, 2)
round(sum(mi_fam_d_bray_pcoa$eig[mi_fam_d_bray_pcoa$eig<0]),2)
round(sum(mi_fam_d_bray_pcoa$eig[mi_fam_d_bray_pcoa$eig>=0]),2)
ordiplot(scores(mi_fam_d_bray_pcoa, choices = c(1, 2)),
  type = "t",
  main = "PCoA con promedios ponderados de especies")
abline(h = 0, lty = 3)
abline(v = 0, lty = 3)
mi_fam_d_bray_pcoa_wa <- wascores(mi_fam_d_bray_pcoa$points[, 1:2], mi_fam)
text(
  mi_fam_d_bray_pcoa_wa,
  rownames(mi_fam_d_bray_pcoa_wa),
  cex = 0.7, col = "red")
(mi_fam_d_bray_pcoa_env <- envfit(mi_fam_d_bray_pcoa, env_num))
plot(mi_fam_d_bray_pcoa_env, p.max = 0.05, col = 3)

#’ ---
#’ title: "Técnicas de ordenación. <br> Parte 2: Ordenación restringida o 'canónica'. <br> RDA,
#’ author: "JR"
#’ date: "21 de noviembre, 2020"
#’ output: github_document
#’ ---

knitr::opts_chunk$set(fig.width=12, fig.height=8)

#’ ## Preámbulo
#’
#’ ### Cargar paquetes
#’
library(vegan)
library(tidyverse)
library(sf)
source('biodata/funciones.R')
#’
#’ ### Cargar datos

```

```

#’
load(’biodata/Myrtaceae.Rdata’)
load(’biodata/matriz_ambiental.Rdata’)
mi_fam <- mc_myrtc
(colnames(mi_fam) <- make.cepnames(colnames(mi_fam)))
(df_equivalecias <- data.frame(
  nombre_original = colnames(mc_myrtc),
  colnames(mi_fam)))
bci_env_grid %>% tibble
#’
#’ ## Ordenación restringida
#’
#’ Con este conjunto de técnicas de ordenación, las tendencias detectadas en un conjunto de dato
#’
#’ Las principales técnicas de ordenación restringida son análisis de redundancia o RDA (siglas
#’
#’ ### Análisis de redundancia (RDA)
#’
#’ En el análisis de ordenación simple o no restringida, la matriz de datos (de comunidad o ambi
#’
#’ RDA combina la regresión y el análisis de componentes principales. El RDA es una regresión li
#’
#’ - Cada variable de respuesta (e.g. matriz de comunidad) es ajustada (por regresión) a la matr
#’
#’ - Se prueba significancia estadística de la relación entre las variables de respuesta y expli
#’
#’ - Si la prueba resulta significativa, se realiza un PCA de la matriz de valores ajustados. Es
#’
#’ - Luego se usa la matriz de vectores propios canónicos para calcular dos tipos de puntuacione
#’
#’   - Usando la matriz de valores ajustados para obtener una ordenación en el espacio de las
#’
#’   - Usando la matriz de valores centrados de las variables de respuesta para obtener una or
#’
#’ Mostraré a continuación ejemplos aplicados a mi familia asignada. Para agilizar, trasladaré l
#’
#’ Nota (de nuevo): recuerda que los datos de BCI están autocorrelacionados espacialmente, por l
#’
#’ #### Ejemplo usando las matriz ambiental de variables suelo:
#’
mi_fam_hel <- decostand(mi_fam, method = ’hellinger’)
mi_fam_hel %>% tibble
env_suelo <- bci_env_grid %>%
  st_drop_geometry %>%
  dplyr::select(matches(’^[A-T,Z]|~pH$’, ignore.case = F))
env_suelo %>% tibble
mi_fam_hel_rda_suelo <- rda(mi_fam_hel ~ ., env_suelo)
summary(mi_fam_hel_rda_suelo)

```

```

#’
#’ ¿Qué partes del resumen debes mirar?
#’
#’ - La varianza, particionada, de la cual se muestra tanto la inercia (absoluta) como la propor
#’
#’ - Los valores propios (*eigenvalues*) y su contribución a la varianza. Existen *eigenvalues*
#’
#’ - Las puntuaciones de sitios (*site scores*) y de especies (*species scores*) tienen el mismo
#’
#’ - Las puntuaciones restringidas de sitio (en ‘{vegan}’ se usa la etiqueta *Site constraints (
#’
#’ - Finalmente, el resumen presenta las puntuaciones para el *biplot*, que más adelante usaré e
#’
#’ Tal como comenté arriba, es importante calcular un  $R^2$  insesgado, siendo este más ú
#’
RsquareAdj(mi_fam_hel_rda_suelo)$adj.r.squared
#’
#’ Normalmente, el  $R^2$  insesgado es mucho más bajo que el sesgado, porque se relativiz
#’
#’ Otro diagnóstico importante es la determinación de los factores de inflación de la varianza (
#’
vif.cca(mi_fam_hel_rda_suelo)
#’
#’ Variables con valores por encima de 10 deben ser examinadas y, desde una posición conservador
#’
#’ Finalmente, la representación del modelo se realiza en un *triplot*, que es un gráfico enriqu
#’
#’ Escalamiento 1:
#’
plot(mi_fam_hel_rda_suelo,
      scaling = 1,
      display = c("sp", "lc", "cn"),
      main = "Triplot de RDA especies ~ var. suelo, escalamiento 1"
)
mi_fam_hel_rda_suelo_sc1 <-
  scores(mi_fam_hel_rda_suelo,
         choices = 1:2,
         scaling = 1,
         display = "sp"
  )
arrows(0, 0,
       mi_fam_hel_rda_suelo_sc1[, 1] * 0.9,
       mi_fam_hel_rda_suelo_sc1[, 2] * 0.9,
       length = 0,
       lty = 1,
       col = "red"
)
#’

```

```

#’ Paso la interpretación de este gráfico y la del siguiente al vídeo correspondiente.
#’
#’ Escalamiento 2
#’
plot(mi_fam_hel_rda_suelo,
     scaling = 2,
     display = c("sp", "lc", "cn"),
     main = "Triplot de RDA especies ~ var. suelo, escalamiento 2"
)
mi_fam_hel_rda_suelo_sc2 <-
  scores(mi_fam_hel_rda_suelo,
        scaling = 2,
        choices = 1:2,
        display = "sp"
  )
arrows(0, 0,
       mi_fam_hel_rda_suelo_sc2[, 1] * 0.9,
       mi_fam_hel_rda_suelo_sc2[, 2] * 0.9,
       length = 0,
       lty = 1,
       col = "red"
)
#’
#’ ##### Ejemplo usando las matriz ambiental con variables seleccionadas
#’
#’ El RDA anterior mostró que las variables de suelo son útiles para predecir la matriz de comun
#’
#’ - Hay mucha colinealidad entre ellas.
#’
#’ - No se probó mejorar el modelo añadiendo otras variables, además de las de suelo.
#’
#’ Crearé una matriz ambiental con las variables que resultaron significativas en el ajuste *pos
#’
env_selec <- bci_env_grid %>%
  select(
    heterogeneidad_ambiental,
    riqueza_global,
    UTM.EW,
    Al, B, Ca, Cu, Fe, K, Mg, Mn, P, Zn, N, N.min., pH) %>%
  st_drop_geometry
mi_fam_hel_rda_selec <- rda(mi_fam_hel ~ ., env_selec)
#’
vif.cca(mi_fam_hel_rda_selec)
#’
#’ Haré el plot del escalamiento 2 para comprobar gráficamente asociación entre variables sin la
#’
plot(mi_fam_hel_rda_selec,
     scaling = 2,

```

```

        display = c("sp", "lc", "cn"),
        main = "Triplot de RDA especies ~ var. selec, escalamiento 2"
    )
    #'
    #' Tal como comenté arriba, variables con valores VIF por encima de 10, deben ser examinadas. En
    #'
    env_selec2 <- bci_env_grid %>%
        select(
            heterogeneidad_ambiental,
            riqueza_global,
            UTM.EW,
            Al, B, Cu, Fe, Mg, Mn, P, Zn, N, N.min., pH) %>%
        st_drop_geometry
    mi_fam_hel_rda_selec2 <- rda(mi_fam_hel ~ ., env_selec2)
    vif.cca(mi_fam_hel_rda_selec2)
    plot(mi_fam_hel_rda_selec2,
        scaling = 2,
        display = c("sp", "lc", "cn"),
        main = "Triplot de RDA especies ~ var. selec2, escalamiento 2"
    )
    #'
    #' Nota que las posiciones rotaron, no así la asociación entre la mayoría de las variables. 'B'
    #'
    env_selec3 <- bci_env_grid %>%
        select(
            heterogeneidad_ambiental,
            riqueza_global,
            UTM.EW,
            Al, Cu, Fe, Mn, P, Zn, N, N.min., pH) %>%
        st_drop_geometry
    mi_fam_hel_rda_selec3 <- rda(mi_fam_hel ~ ., env_selec3)
    vif.cca(mi_fam_hel_rda_selec3)
    plot(mi_fam_hel_rda_selec3,
        scaling = 2,
        display = c("sp", "lc", "cn"),
        main = "Triplot de RDA especies ~ var. selec3, escalamiento 2"
    )
    #'
    #' Finalmente, la coordenada 'UTM.EW' tiene un alto valor VIF, por lo que es preferible excluirla
    #'
    env_selec4 <- bci_env_grid %>%
        select(
            heterogeneidad_ambiental,
            riqueza_global,
            Al, Cu, Fe, Mn, P, Zn, N, N.min., pH) %>%
        st_drop_geometry
    mi_fam_hel_rda_selec4 <- rda(mi_fam_hel ~ ., env_selec4)
    vif.cca(mi_fam_hel_rda_selec4)

```

```

plot(mi_fam_hel_rda_selec4,
     scaling = 2,
     display = c("sp", "lc", "cn"),
     main = "Triplot de RDA especies ~ var. selec4, escalamiento 2"
)
#'
#' Las variables 'N.min.' y 'Zn' ahora presentan valores VIF aceptables. Habrás notado que es to
#'
summary(mi_fam_hel_rda_selec4)
RsquareAdj(mi_fam_hel_rda_selec4)$adj.r.squared
#'
#' Triplot, pero ahora con las flechas para las especies.
#'
#' Escalamiento 1:
#'
plot(mi_fam_hel_rda_selec4,
     scaling = 1,
     display = c("sp", "lc", "cn"),
     main = "Triplot de RDA especies ~ var. selec4, escalamiento 1"
)
mi_fam_hel_rda_selec4_sc1 <-
  scores(mi_fam_hel_rda_selec4,
        choices = 1:2,
        scaling = 1,
        display = "sp"
  )
arrows(0, 0,
       mi_fam_hel_rda_selec4_sc1[, 1] * 0.9,
       mi_fam_hel_rda_selec4_sc1[, 2] * 0.9,
       length = 0,
       lty = 1,
       col = "red"
)
#'
#' Escalamiento 2
#'
plot(mi_fam_hel_rda_selec4,
     scaling = 2,
     display = c("sp", "lc", "cn"),
     main = "Triplot de RDA especies ~ var. selec4, escalamiento 2"
)
mi_fam_hel_rda_selec4_sc2 <-
  scores(mi_fam_hel_rda_selec4,
        scaling = 2,
        choices = 1:2,
        display = "sp"
  )
arrows(0, 0,

```



```

        mi_fam_hel_rda_selec4_sc2[, 1] * 0.9,
        mi_fam_hel_rda_selec4_sc2[, 2] * 0.9,
        length = 0,
        lty = 1,
        col = "red"
    )
    #'
    #' ### Análisis de correspondencia canónica (CCA)
    #'
    #' ##### Ejemplo usando las matriz ambiental con variables seleccionadas:
    #'
    mi_fam_cca_selec4 <- cca(mi_fam ~ ., env_selec4)
    summary(mi_fam_cca_selec4)
    RsquareAdj(mi_fam_cca_selec4)
    #'
    #' Escalamiento 1
    #'
    plot(mi_fam_cca_selec4,
        scaling = 1,
        display = c("sp", "lc", "cn"),
        main = "Triplot de CCA especies ~ var. selec4, escalamiento 1"
    )
    #'
    #' Escalamiento 2
    #'
    plot(mi_fam_cca_selec4,
        scaling = 2,
        display = c("sp", "lc", "cn"),
        main = "Triplot de CCA especies ~ var. selec4, escalamiento 2"
    )
    #'
    #' Excluyendo especies con abundancia menor a 100 individuos
    #'
    colSums(mi_fam) %>% sort
    mi_fam_no_raras <- mi_fam %>% select_if(colSums(.) > 100)
    intersect(colnames(mi_fam), colnames(mi_fam_no_raras))
    setdiff(colnames(mi_fam), colnames(mi_fam_no_raras))
    mi_fam_no_raras_cca_selec4 <- cca(mi_fam_no_raras ~ ., env_selec4)
    summary(mi_fam_no_raras_cca_selec4)
    RsquareAdj(mi_fam_no_raras_cca_selec4)
    #'
    #' Escalamiento 1
    #'
    plot(mi_fam_no_raras_cca_selec4,
        scaling = 1,
        display = c("sp", "lc", "cn"),
        main = "Triplot de CCA especies no raras ~ var. selec4, escalamiento 1"
    )
    #'

```

```
#'
#' Escalamiento 2
#'
plot(mi_fam_no_raras_cca_selec4,
      scaling = 2,
      display = c("sp", "lc", "cn"),
      main = "Triplot de CCA especies no raras ~ var. selec4, escalamiento 2"
)
```

8.5 Ecología espacial

```
#' ---
#' title: "Análisis espacial de datos ecológicos. <br> Autocorrelación"
#' author: "JR"
#' date: "5 de diciembre, 2020"
#' output: github_document
#' ---
#'
knitr::opts_chunk$set(fig.width=12, fig.height=8)
#'
#' ## Preámbulo
#'
#' ### Cargar paquetes
#'
library(ape)
library(spdep)
library(ade4)
library(adegraphics)
library(adespatial)
library(vegan)
library(tidyverse)
library(sf)
library(gridExtra)
library(grid)
library(gtable)
source('biodata/funciones.R')
source('https://raw.githubusercontent.com/maestria-geotel-master/unidad-3-asignacion-1-vecindad-')
#'
#' ### Cargar datos
#'
load('biodata/Myrtaceae.Rdata')
load('biodata/matriz_ambiental.Rdata')
mi_fam <- mc_myrtc
mi_fam %>% tibble
bci_env_grid %>% tibble
#'
#' ## Preparar datos
```

```

#'
#' ### Generar matriz Hellinger
#'
mi_fam_hel <- decostand (mi_fam, "hellinger")
#'
#' ### Transformar matriz ambiental en objeto 'sp', generar vecindad
#'
bci_env_grid_sp <- bci_env_grid %>% as_Spatial
centroides <- bci_env_grid %>% st_centroid
bci_xy <- centroides %>% st_coordinates %>% as.data.frame
(vecindad <- bci_env_grid_sp %>% poly2nb)
(pesos_b <- nb2listw(vecindad, style = 'B'))
#+ fig.width=12, fig.height=6
plot(bci_env_grid_sp)
plot(vecindad, coords = bci_xy, add=T, col = 'red')
#'
#' ## Autocorrelación espacial mediante correlograma
#'
#' ### Una y solo una variable ambiental
#'
var_ph <- bci_env_grid %>% st_drop_geometry %>% pull(pH)
ph_correl <- sp.correlogram(vecindad,
                           var_ph,
                           order = 9,
                           method = "I",
                           zero.policy = TRUE)
print(ph_correl, digits = 2, p.adj.method = 'holm')
#+ fig.width=12, fig.height=6
plot(ph_correl)
#'
#' ### Múltiples variables
#'
#' ##### Abundancias de especies (matriz de comunidad transformada)
#'
suppressWarnings(auto_spp_hel <- calcular_autocorrelacion(
  df_fuente = mi_fam_hel,
  orden = 9,
  obj_vecindad = vecindad,
  pos_var = '(matriz Hellinger)'))
print(auto_spp_hel, digits = 2, p.adj.method = 'holm')
dim_panel <- rev(n2mfrow(ncol(mi_fam_hel)))
#+ fig.width=12, fig.height=10
par(mfrow = dim_panel)
suppressWarnings(invisible(lapply(auto_spp_hel, function(x) plot(x, main = x$var))))
#'
#' ### Variables ambientales (matriz ambiental)
#'
bci_env_grid_num <- bci_env_grid %>%

```

```

    st_drop_geometry %>%
    select_if(is.numeric) %>%
    select(-id, -UTM.EW, -UTM.NS)
suppressWarnings(auto_amb <- calcular_autocorrelacion(
  df_fuente = bci_env_grid_num,
  orden = 9,
  obj_vecindad = vecindad))
print(auto_amb, digits = 2, p.adj.method = 'holm')
dim_panel <- rev(n2mfrow(ncol(bci_env_grid_num)))
#+ fig.width=12, fig.height=14
par(mfrow = dim_panel)
suppressWarnings(invisible(lapply(auto_amb, function(x) plot(x, main = x$var))))
#'
#' ## Autocorrelación espacial mediante prueba Mantel (matrices de distancia)
#'
#' Para aplicar la prueba Mantel a datos de comunidad, es necesario quitar la tendencia espacial
#'
mi_fam_sin_tendencia <- resid(
  lm(as.matrix(mi_fam_hel) ~ .,
    data = bci_xy))
mi_fam_sin_tendencia_d <- dist(mi_fam_sin_tendencia)
(mi_fam_correlograma <- mantel.correlog(
  mi_fam_sin_tendencia_d,
  XY = bci_xy,
  nperm = 999))
#+ fig.width=12, fig.height=6
plot(mi_fam_correlograma)
#'
#' ## Autocorrelación espacial por medio de pruebas de permutación para el I de Moran
#'
#' ### I de Moran global aplicado a abundancia de especies transformadas sin tendencia
#'
'autocor_global_residuos <- sapply(
  dimnames(mi_fam_sin_tendencia)[[2]],
  function(x)
    moran.mc(
      x = mi_fam_sin_tendencia[,x],
      listw = pesos_b,
      zero.policy = T,
      nsim = 9999),
  simplify = F))
#'
#' ### I de Moran local
#'
#' #### Aplicado a variables ambientales
#'
bci_env_grid_num_sf <- bci_env_grid %>%
  select_if(is.numeric) %>%

```

```

select(-id, -UTM.EW, -UTM.NS)
bci_env_grid_num_sf %>% tibble
lisamaps_amb <- sapply(grep('geometry', names(bci_env_grid_num_sf), invert = T, value = T),
  function(x) {
    m <- lisamap(objesp = bci_env_grid_num_sf[x],
      var = x,
      pesos = pesos_b,
      tituloleyenda = 'Significancia ("x-y", léase como "x" rodeado de "y")',
      leyenda = F,
      anchuratitulo = 50,
      tamanotitulo = 10,
      fuentedatos = '\nhttp://ctfs.si.edu/webatlas/datasets/bci/',
      titulomapa = paste0('Clusters LISA de "', x, '"'))
    return(m$grafico)
  }, simplify = F
)
lisamaps_amb$leyenda <- gtable_filter(ggplot_gtable(ggplot_build(lisamaps_amb[[1]]) + theme(legend.position = "bottom"),
  grid.arrange(do.call('arrangeGrob', c(lisamaps_amb[1:12], nrow = 3)), lisamaps_amb$leyenda, height = 100),
  grid.arrange(do.call('arrangeGrob', c(lisamaps_amb[13:22], nrow = 3)), lisamaps_amb$leyenda, height = 100),
  grid.arrange(do.call('arrangeGrob', c(lisamaps_amb[23:31], nrow = 3)), lisamaps_amb$leyenda, height = 100),
  #
#' ##### Aplicado a abundancias de especies transformadas
#
mi_fam_hel_sf <- bci_env_grid %>% select %>% bind_cols(mi_fam_hel)
lisamaps_mifam <- sapply(
  grep('geometry', names(mi_fam_hel_sf), invert = T, value = T),
  function(x) {
    m <- lisamap(objesp = mi_fam_hel_sf[x],
      var = x,
      pesos = pesos_b,
      tituloleyenda = 'Significancia ("x-y", léase como "x" rodeado de "y")',
      leyenda = F,
      anchuratitulo = 50,
      tamanotitulo = 10,
      fuentedatos = '\nhttp://ctfs.si.edu/webatlas/datasets/bci/',
      titulomapa = paste0('Clusters LISA de "', x, '"'))
    # dev.new();print(m$grafico)
    return(m$grafico)
  }, simplify = F
)
lisamaps_mifam$leyenda <- gtable_filter(ggplot_gtable(ggplot_build(lisamaps_mifam[[1]]) + theme(legend.position = "bottom"),
  grid.arrange(do.call('arrangeGrob', c(lisamaps_mifam[1:8], nrow = 3)), lisamaps_mifam$leyenda, height = 100),
  # grid.arrange(do.call('arrangeGrob', c(lisamaps_mifam[9:16], nrow = 3)), lisamaps_mifam$leyenda, height = 100),
  #
#' ##### Aplicado a abundancias de especies transformadas sin tendencia
#
mi_fam_sintendencia_sf <- bci_env_grid %>% select %>% bind_cols(mi_fam_sin_tendencia %>% as.data.frame())
lisamaps_mifam_sintendencia <- sapply(

```

```

grep('geometry', names(mi_fam_sintendencia_sf), invert = T, value = T),
function(x) {
  m <- lisamap(objesp = mi_fam_sintendencia_sf[x],
    var = x,
    pesos = pesos_b,
    tituloleyenda = 'Significancia ("x-y", léase como "x" rodeado de "y")',
    leyenda = F,
    anchuratitulo = 50,
    tamanotitulo = 10,
    fuentedatos = '\nhttp://ctfs.si.edu/webatlas/datasets/bci/',
    titulomapa = paste0('Clusters LISA de "', x, '"'))
  # dev.new();print(m$grafico)
  return(m$grafico)
}, simplify = F
)
lisamaps_mifam_sintendencia$leyenda <- gtable_filter(ggplot_gtable(ggplot_build(lisamaps_mifam_sintendencia),
grid.arrange(do.call('arrangeGrob', c(lisamaps_mifam_sintendencia[1:8], nrow = 3)), lisamaps_mifam_sintendencia[9:16], nrow = 3)), lisamaps_mifam_sintendencia$leyenda)
# grid.arrange(do.call('arrangeGrob', c(lisamaps_mifam_sintendencia[9:16], nrow = 3)), lisamaps_mifam_sintendencia$leyenda)

```

Referencias

- Almeida, S. (2019). *Myrtaceae, familia*. Retrieved from <https://knoow.net/es/ciencias-tierra-vida/biologia-es/myrtaceae-familia/>
- Battle, J. R. M. (2020). biogeografia-master/scripts-de-analisis-BCI: Long coding sessions (Version v0.0.0.9000). <https://doi.org/10.5281/zenodo.4402362>
- Borcard, D., & Legendre, P. (2012). Is the mantel correlogram powerful enough to be useful in ecological analysis? A simulation study. *Ecology*, 93(6), 1473–1481.
- Borcard, D., Gillet, F., Legendre, P., & others. (2011). *Numerical ecology with r* (Vol. 2). Springer.
- Britannica, E. (2016). *Myrtaceae*. Retrieved from <https://www.britannica.com/plant/Myrtaceae>
- Buzai, G. D., & Baxendale, C. A. (2009). Análisis exploratorio de datos espaciales. *Geografía Y Sistemas de Información Geográfica, No 1*,(2009).
- Chao, A., & Chiu, C.-H. (2016). Species richness: Estimation and comparison. *Wiley StatsRef: Statistics Reference Online*, 1, 26.
- Claver Farías, I., & others. (1984). Guía para la elaboración de estudios del medio físico: Contenido y metodología. *Ministerio de Obras Públicas Y Urbanismo. Madrid*.
- De la Lata Loyola, M. D. (2003). *Ecología y medio ambiente*. Editorial Progreso.
- Efron, B. (1992). Bootstrap methods: Another look at the jackknife. In *Breakthroughs in statistics* (pp. 569–593). Springer.
- Hubbell, S., Foster, R., & Condit, R. (2005). *Barro colorado forest census plot data*. Retrieved from <http://ctfs.si.edu/webatlas/datasets/bci/>
- Hurtado, M. J. R., & Silvente, V. B. (2012). Cómo aplicar las pruebas paramétricas bivariadas t de student y anova en spss. caso práctico. *Reire*, 5(2), 83–100.
- Jaccard, P. (1908). Nouvelles recherches sur la distribution florale. *Bull. Soc. Vaud. Sci. Nat.*, 44, 223–270.
- Kruskal, W. H., & Wallis, W. A. (1952). Use of ranks in one-criterion variance analysis. *Journal*

of the American Statistical Association, 47(260), 583–621.

Legendre, P., & Legendre, L. (2012). *Numerical ecology*. Elsevier.

Lorenzo-Cáceres, J. M. S. de. (s.f.). *Familia myrtaceae*. Retrieved from <https://www.arbolesornamentales.es/Myrtaceae.htm>

Magurran, A. E. (1988). *Ecological diversity and its measurement*. Princeton university press.

Palmer, A., Jiménez, R., & Montaña, J. J. (2000). Tutorial sobre coeficientes de correlación con una o dos variables categóricas. *Revista Electrónica de Psicología*, 4(2).

Perez, R., & R, C. (s.f.). *Tree atlas of panama*. Retrieved from <http://ctfs.si.edu/PanamaAtlas/famdescr.php?Family=Myrtaceae>

Pérez, R., Aguilar, S., Condit, R., Foster, R., Hubbell, S., & Lao, S. (2005). Metodología empleada en los censos de la parcela de 50 hectareas de la isla de barro colorado, panamá. *Centro de Ciencias Forestales Del Tropico (CTFS) Y Instituto Smithsonian de Investigaciones Tropicales (STRI)*, 1–24.

Pineda, P. (2014). *Análisis del sistema de parcelas permanentes de medición en los bosques de guatemala. informe final*. Guatemala: Proyecto" Sistemas de información sobre la productividad de los ...

R Core Team. (2020). *R: A language and environment for statistical computing*. Retrieved from <https://www.R-project.org/>

Rodríguez-Castro, L., & Flores, N. (2021). Colonización de la hepática epífila leptolejeunea elliptica (lehm & lindenb) schiffn.(Lejeuneaceae), sobre dos especies de arbustos de la isla barro colorado (bci). *Tecnociencia*, 23(1), 82–103.

Romero Domínguez, A. (2021). *Propuesta de estrategia de monitoreo transaccional anti lavado de activos empleando el método de ward y el teorema de chebyshev*.

Romesburg, C. (2004). *Cluster analysis for researchers*. Lulu. com.

Salazar, M. E. R. (2000). *Coeficientes de asociación*. Plaza y Valdes.

Stewart, R., Stewart, J., & Woodring, W. (1980). *Geologic map of the panama canal and vicinity, republic of panama: Mapa geologico del canal de panama y sus alrededores, republica de panama*. United States Geological Survey.

Sugasti, L., Eng, B., & Pinzón, R. (2018). Medición continua de flujo de co2 ensuelo en una parcela de bosque tropical en isla barro colorado, canal de panamá. *Universidad Tecnológica de Panamá*, 1–7.

Tibshirani, R. J., & Efron, B. (1993). An introduction to the bootstrap. *Monographs on Statistics and Applied Probability*, 57, 1–436.

Whittaker, R. H. (1972). Evolution and measurement of species diversity. *Taxon*, 21(2-3), 213–251.

Wilcoxon, F., & Wilcox, R. A. (1964). *Some rapid approximate statistical procedures*. Lederle Laboratories.