# Ordenaciones

En ecología es bastante normal que dispongamos de un conjunto de datos que están conformados por una serie de sitios o localidades, para los cuales tenemos una serie de variables. Estas variables puede ser cada especie o cada condición que levantemos de este sitio, de esta forma un sitio va a tener tantas variables como especies o factores levantados.

Aunque como vimos un diagrama de dispersión es una buena herramienta para representar las tendencias del conjunto de grupos, existe la limitante de que esto es factible con dos o máximo tres variables (especies), con más de tres variables es imposible poder graficarlo.

De esta forma, el objetivo de los métodos de ordenación es representar los datos a lo largo de un número reducido de ejes ortogonales, construidos de tal manera que representan, en orden, las principales tendencias de los datos [@Borcard2011].

Las ordenaciones pueden ser indirectas y directas (constreñidas). Las ordenaciones indirectas pueden ser utilizadas para interpretarse visualmente o asociadas a otros métodos como regresión. Mientras que las ordenaciones directas permiten hacer asociaciones con variables explicativas, generar un orden constreñido por otras variables.

En este capítulo nos centraremos en las Ordenaciones indirectas.

##Ordenación indirecta o no constreñida

Existen muchos tipos de ordenaciones indirectas como saber que ordenación debo utilizar. Una posibilidad es ver el tipo de respuesta que tenemos, si es una respuesta linear o monotónica o una respuesta unimodal.

Para determinar que tipo de respuesta usamos la desviación estándar. Si asumimos una distribución normal, datos con menos de tres desviaciones estándar tendrán una respuesta lineal, mientras que si tiene más de tres desviaciones se asumirá una respuesta unimodal. Para cada una de estas respuestas existe un análisis que se puede ejecutar \@ref(tab:ordenacion).

```{r ordenacion, echo=FALSE}

Ordenacion<- data.frame(`Medidas de Similitud` = c("Respuesta lineal", "Respuesta Unimodal", "Bray-Curtis"), `Tipo de Ordenación` = c("PCA", "CA/DCA", "PcoA/mMDS/nmMDS"))

knitr::kable(

Ordenacion, booktabs = TRUE,

caption = 'Relación entre el tipo de variable y el método de ordenación a utilizar'

)

```

Algunos comentarios sobre los métodos de ordenación.

\_\_\*Principal Component Analysis (PCA)\*\_\_

Fortalezas

Es sencillo de interpretar ya que las distancias entre las muestras son interpretadas directamente como distancias euclidianas.

Apropiado para análisis multivariantes de variables ambientales. El PCA es fuertemente afectado por ceros algo que es bastante común en datos biológicos, al contrario de datos ambientales donde la cantidad de ceros es menor.

Debilidades

Si dos muestras coinciden en celdas con ceros (dobles ceros) incrementa la similitud entre muestras.

La proyección de las distancias euclidias en un plano puede distorsionar algunas distancias en otros planos.

\_\_\*Principal Coordinates Análisis (PCoA)\*\_\_

Fortalezas

Es una extensión del PCA, permite una definición más amplia de las disimilitudes entre muestras que la distancia Euclidia.

Es capaz de convertir otras medidas de disimilitud (además de la euclidia) en distancias entre muestras.

Debilidades

La proyección de las distancias en un plano también sufre importantes distorsiones como en el caso del PCA.

\_\_\*Correspondence Analysis\*\_\_

Fortalezas

Implícitamente se generan distancias de Chi-cuadrado entre las muestras.

Se basa en un modelo de respuesta unimodal de las especies a los gradientes ambientales subyacentes.

Debilidades

“Efecto arco” Causado por la respuesta unimodal de la abundancia de especies a un gradiente ambiental.

\_\_\*Detrended Correspondence Analysis (DCA, DECORANA)\*\_\_

Es una extensión del CA

Fortalezas

Corrige el “Efecto arco” del CA, realizando una ordenación por segmentos y se realinean los segmentos para evitar la curvatura

Debilidades

Excesiva manipulación de los datos (Pielou 1984)

Hay quien opina que es demasiado parecido a cortar y pegar los datos con tijeras y pegamento.

\_\_\*Multidimensional Scaling (MDS)\*\_\_

Se puede construir una matriz de semejansas con el coeficiente de semejanza que se considere biológicamente relevante para los datos.

La ordenación de las muestras utiliza el orden relativo de las distancias entre muestras y no los valores absolutos de los coeficientes de similitud. Por eso la representación gráfica del MDS sufre menos distorsiones respecto a las distancias reales.

pretende satisfacer las condiciones impuestas por el “rango” de similitudes. p.ej.: Si la muestra 1 es más parecida a la 2 que a la 3 entonces se localizará más cercana a la muestra 2 que a la 3.

##Pasos para el análisis de Ordenación

\_\_1. Decidir que ordenación realizar\_\_

Este es el primer paso para poder realizar el análisis de ordenación. Como lo hemos visto podemos usar información para poder determinar que análisis realizar. El tipo de datos que tenemos y por tanto la distancia a utilizar, si tenemos datos con muchos ceros no podremos utilizar una medida de distancia Euclidiana y deberíamos trabajar con distancia de Bray-Curtis, por tanto es posible que esto nos defina el trabajar con Escalado Multidimensional (Multidimensional Scaling). Si estamos utilizando otras distancias podríamos definir la cantidad de desviaciones estándar para decidir el tipo de análisis a realizar (\@ref(tab:ordenacion)).

Para ver la longitud de la gradiente (cantidad de desviaciones estándar) podríamos utilizar la función decorana.

```{r}

library(vegan)

data(dune)

#str(dune) revisa los datos

ord.dca <- decorana(dune)

```

En el primer eje (DCA1) podemos ver la longitud de este eje, el cual esta expresado en Unidades de desviación, en este caso 3.7. Esta información ya nos dice que el análisis que debemos hacer será un CA o un DCA.

\_\_2. Transformación de los datos\_\_

Lo siguiente que debemos decidir es si es necesario transformar. Siempre lo mejor es no transformar los datos, sin embargo, como vimos antes si los datos son muy distintos es necesario realizar esta transformación. Revisa los datos de Dune, como ven no existen diferencias importantes entre los datos por tanto no transformaremos.

>Recuerde para transformar los datos definimos la variación entre variables. Variables con más de tres magnitudes de diferencia logaritmo y con dos raíz cuadrada.

Aunque no es necesario transformar, la función `decostand` del paquete vegan se puede utilizar para la estandarización y transformar.

```{r}

dsRaiz <- decostand(dune, "standardize", "hellinger")

#Estandarizado y transformado raíz cuadrada (hellinger)

dslog <- decostand(dune, "standardize", "log")

#Estandarizado y transformado logaritmo (log)

dsSta <- decostand(dune, "standardize")

#Estandarizado

```

\_\_3. Desarrollar el análisis\_\_

Realizamos el análisis Decorana y Análisis Canónico (CA) y graficamos los datos.

```{r}

ord.dca <- decorana(dune)

ord.dca

ord.ca <- cca(dune)

ord.ca

```

Realizamos el análisis de escalado multidimensional (MDS) usando la función `metaMDS`

```{r, message=FALSE, cache.comments=FALSE,results='hide'}

ord.nmds <- metaMDS(dune,distance="bray", k=2, trymax=50)

```

```{r}

ord.nmds

```

\_\_4. Interpretar los resultados\_\_

Para el DCA y CA

Eigenvalue: es la medida de la importancia de un eje, la cantidad de varianza que explica ese eje. En este caso, el primer eje tanto el DCA como el CA explican el 53% de la variación

Axis length: es la longitud del eje principal. Nos da una estima de qué tipo de respuesta tienen nuestros datos.

Para el MDS

El valor de estrés (stress) nos indica la cantidad de varianza que \_\_NO\_\_ se ajustó, mientras más bajo es el estrés mayor varianza explica. Algunos autores plantean que bajo 0.2 es aceptable, mientras que valores superiores hay un ajuste pobre. En el caso del ejemplo el ajuste es bueno.

\_\_5. Graficar los resultados\_\_

Lo siguiente que debemos hacer es graficar los resultados con el fin de ver si existen algunos patrones.

```{r plotnmds, fig.align='center', fig.width=4.5, fig.height=4.5, fig.cap= "Representación gráfica del NMDS"}

plot(ord.nmds, type="n") # para dibujar un plot vacío

points(ord.nmds, display="sites", cex=0.8, pch=21, col="black", bg="yellow")

text(ord.nmds, display= "spec", cex=0.5, col="blue")

```

Podemos ver en la figura \@ref(fig:plotnmds) que ciertas especies están asociadas con ciertas localidades esto nos podría servir para explorar los datos.

# Ordenaciones Directas o Constreñidas

Si bien las técnicas de ordenación indirecta nos permiten descubrir ciertos patrones, no nos permite testar hipótesis y ver las relaciones de esta matriz con otras variables.

Si disponemos de una matriz de variables explicativas es posible utilizar análisis de ordenación constreñidos. De esta forma, esta matriz representa la información que tenemos sobre cada una de las muestras y podemos usarla para predecir los valores de las variables respuesta (la composición de especies).

Al igual que para los análisis de ordenaciones no constreñidas el tipo de ordenación depende de la respuesta que tenemos en las variables (Tabla \@ref(tab:ordenacion1) )

```{r ordenacion1, echo=FALSE}

Ordenacion1<- data.frame(`Medidas de Similitud` = c("Respuesta lineal", "Respuesta Unimodal", "Bray-Curtis"), `Tipo de Ordenación` = c("PCA", "CA/DCA", "PcoA/mMDS/nmMDS"), `Tipo Ordenación Constreñida` = c("RDA (Redundancy Analysis)", "CCA (Canonical Correspondence Analysis)", "PERMANOVA"))

knitr::kable(

Ordenacion1, booktabs = TRUE,

caption = 'Relación entre el tipo de variable y el método de ordenación a utilizar'

)

```

Una interesante propiedad de los análisis de ordenación constreñidos es que puedo hacer una ordenación parcial. Esta propiedad me permite evaluar como un grupo de variables pueden influir en mi matriz de respuesta. Podría dividir la información, por ejemplo, en variables ambientales y variables bióticas y ver cuánto explica cada una y cuanto explican en conjunto.

##Realizando una ordenación constreñida

Al igual que en el caso de la ordenación no constreñida debemos decidir el tipo de ordenación constreñida que vamos hacer. En el caso de los datos de \_Dune\_ sabemos que la respuesta es unimodal por lo que escogeremos un análisis canónico de correspondencias (CCA) para nuestra ordenación constreñida.

Para hacer la ordenación constreñida necesitamos una matriz con variables explicativas, utilizaremos las variables provistas en el paquete vegan denominadas env.env.

```{r}

data("dune.env") #Llamamos a los datos

ord.cca <- cca(dune~ A1 + Use, data=dune.env)

ord.cca

```

Lo que podemos ver es que la variable A1 más Use explican el 22.33% de la variación en los datos.

La decisión de que modelos deberían generar debe responder a una lógica ecológica, así podemos probar como algunas variables juegan o no un rol en la estructura de la comunidad.

Una herramienta que podríamos utilizar para analizar la importancia de cada variable es utilizar la función `envfit`, esta función permite relacionar la ordenación no constreñida con las variables explicativas y mediante un test de permutación mostrarnos que variables se asocian significativamente con la ordenación.

```{r}

fitVar <- envfit(ord.ca, dune.env)

fitVar

```

Podemos utilizar el \_Goodness of fit\_ para ver cuáles son las variables que ajustan la ordenación y utilizar estas para hacer el modelo constreñido. En este caso utilizaremos Manure y Management.

```{r}

ord.ccafit <- cca(dune~Manure+Management, data=dune.env)

ord.ccafit

```

Como vemos con este procedimiento subimos al 41% de la varianza explicada.

Podemos utilizar la función anova para evaluar la significancia de cada variable dentro del modelo, de forma separada.

```{r}

ord.ccaT <- cca(dune~ ., data=dune.env)

anova(ord.ccaT, by="term", permu=1000)

```

Como vemos esto cambia lo que inicialmente habíamos decidido, esto es debido a que algunas de las variables pueden estar correlacionadas entre ellas.

Bien ahora necesitamos graficar los resultados. Podemos utilizar la función plot e ir graficando cada uno de los componentes (Figura \@ref(fig:plotcca)).

```{r plotcca, fig.align='center', fig.width=4.5, fig.height=4.5, fig.cap= "Representación gráfica del CCA"}

plot(ord.cca, dis="sp", type="n")

points(ord.cca, dis="sites", pch=19, col="grey")

points(ord.cca, display="cn", col="blue", pch=19)

text(ord.cca, dis="sp", cex=0.6)

text(ord.cca, display = "cn", col="blue", cex=0.7)

```

Muchas veces uno de los problemas que tenemos para graficar los datos es que los nombres de las especies son muy largos, en estos casos podemos utilizar una función que se denomina `make.cepnames` la cual permite acortar los nombres.

```{r}

data(BCI)

names(BCI[1:5])

short <- make.cepnames(names(BCI[1:5]))

short

```

Existen procedimientos para construir modelos que ahora no tocaremos, puede encontrar más información en [Oksanen 2015](http://cc.oulu.fi/~jarioksa/opetus/metodi/vegantutor.pdf)

Nota: para realizar un permanova debemos utilizar la función adonis. El procedimiento es similar al desarrollo del cca.

##Ejercicio 3: Análisis de Ordenación

Con los datos utilizados para realizar el análisis de aglomerados, vamos a realizar un análisis de ordenación constreñida.

a. Defina que tipo de ordenación constreñida debe realizar para explicar la variación de los datos de herbáceas.

b. Realice un análisis para definir las variables que se debería utilizar en el análisis

c. Ajuste un modelo y defina el porcentaje de variación explicado.

d. Compara los resultados de la ordenación directa si en vez de transformar los datos corremos los modelos con los datos brutos.

d. Realice un gráfico del modelo desarrollado.