PEC 1 - Estadística Multivariante

Alejandro Keymer

20 de Noviembre, 2019

```
# Cargo las librerías utilizadas. Entre otras utilizo la sintaxis de 'tidyverse'
# Las otras librerías son de funciones utilizadas a lo largo de la PEC
pacman::p_load(rcompanion, faraway, broom, tidyverse, knitr, scales,
               FactoMineR, factoextra, GGally, MVN, heplots)
```

Mercurio en los lagos de Florida

Ejercicio 1

(a) Preparar la base de datos `mercbass` a partir del archivo MERCBASS.TXT. Habrá que tener cuidado con los nombres de las columnas. Tal vez se deba utilizar un read.delim(). También habrá que nombrar correctamente las observaciones.

```
# La función read_tsv de tidyverse es mas rápida y permite configurar de manera
# adecuada el proceso de importación de los datos
mercbass <-
   read_tsv(
      "MERCBASS.txt",
      col_types =
         cols(
            ID = col_character(),       # el id es una etiqueta, no un numero
            Lake = col_character(),
            Alkalinity = col_double(),
            pH = col_double(),
            Calcium = col_double(),
            Chlorophyll = col_double(),
            `Avg Mercury` = col_double(),
            `# samples` = col_double(),
            Min = col_double(),
            Max = col_double(),
            `3yrStand` = col_double(),
            `age data` = col_logical()
         )
   )
```

(b) El Estado de Florida ha fijado un nivel de concentración de mercurio en alimentos comestibles de 0.5 ppm por encima del cual se consideran no saludables. ¿Qué porcentaje de lagos en este estudio tienen un nivel superior de concentración de mercurio para considerarse no saludables? ¿Podemos considerar esta estimación como un buen estimador del porcentaje de lagos de Florida que superan el nivel declarado? ¿Porqué o porqué no?

```
# Creo una variable lógica para la condición de mercurio promedio > 0.5 y luego
# una tabla de contingencia. Entiendo que 'por encima de...' no incluye el
# valor.
mercbass %>%
```

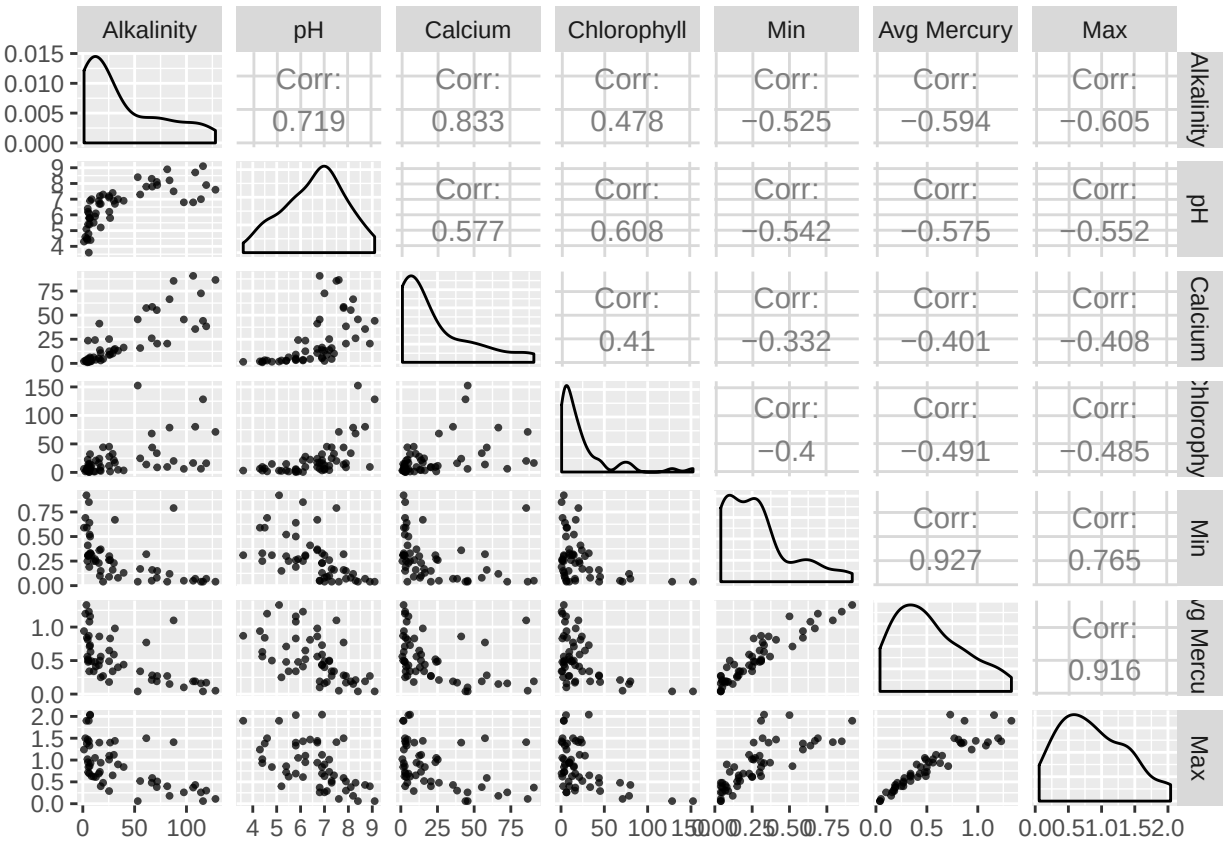
```r
   mutate(saludable = `Avg Mercury` <= 0.5) %>%
   group_by(saludable) %>%
   summarise(n = n()) %>%
   mutate(perc = percent(n/sum(n)))
```

```
## # A tibble: 2 x 3
##   saludable     n perc
##   <lgl>     <int> <chr>
## 1 FALSE        22 41.5%
## 2 TRUE         31 58.5%
```

El estimador elegido es un estimador puntual, con un resultado de porcentaje o proporción. Cómo estimador puntual tiene ventajas y desventajas. La ventaja es que es muy fácil de calcular y de interpretar. El problema es que no se toma en cuenta el error, por lo que no podemos conocer cuan preciso o fiable es. Tampoco conocemos la metodología por la que se realiza el muestreo (puede haber sesgo), por lo que no funciona bien como parámetro estimador. Por otra parte, se podría hacer una extrapolación de un IC asumiendo que el error tiene una distribución normal, pero creo que no se puede llegar a esta conclusión por lo que estimar el error tampoco sería correcto.

(c) Como estamos interesados en la relación entre las 4 variables químicas y la concentración de mercurio, realizar diagramas de dispersión dos a dos. ¿Qué dificultades observamos?

```r
# utilizo ggpairs de la librería GGally, que es más flexible y a mi gusto visualmente
# mas atractivo
mercbass %>%
   select(Alkalinity, pH, Calcium, Chlorophyll, Min, `Avg Mercury`, Max) %>%
   ggpairs(., lower = list(continuous = wrap("points", alpha = 0.75, size = .7)))
```

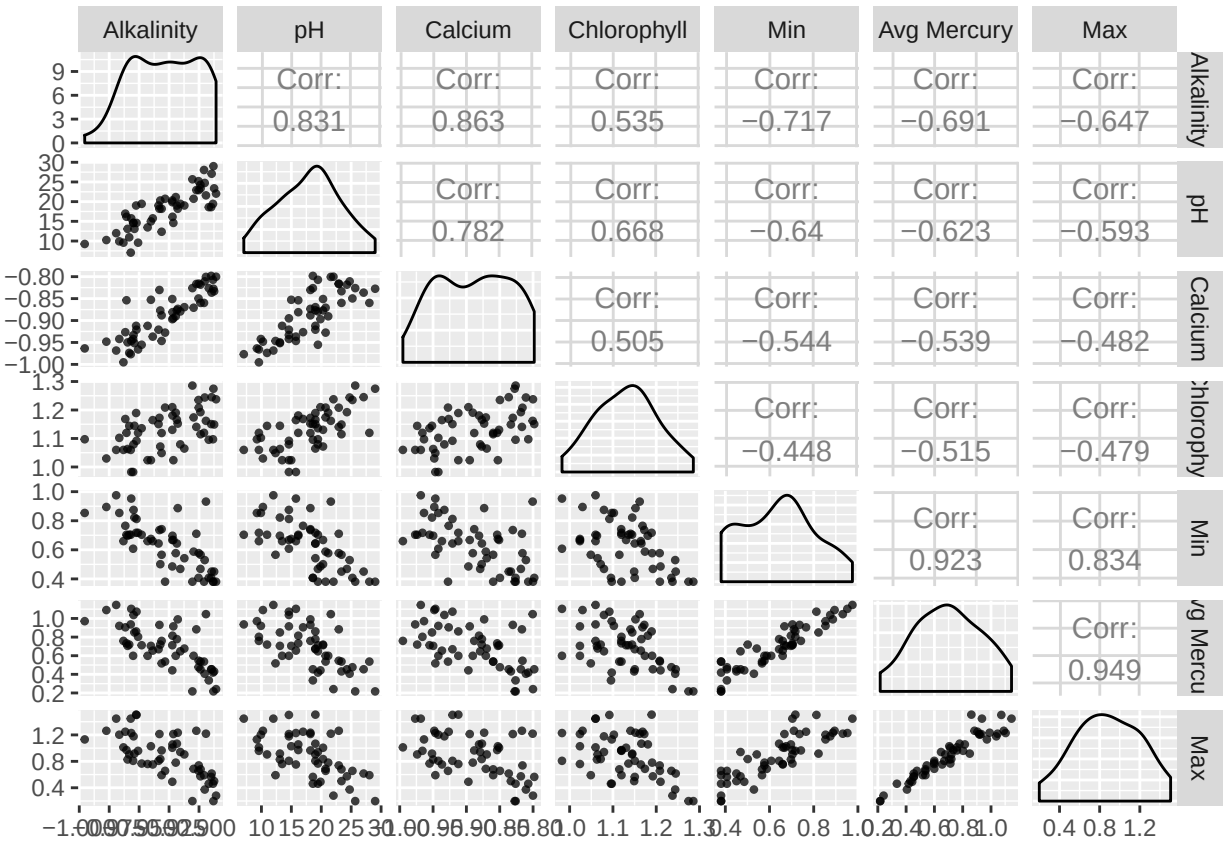


En cuanto a las dificultades, podemos observar que la base de datos original presenta algunos visibles en los gráficos de dispersión y en las curvas de densidad. En cuanto a las curvas de densidad, es bastante claro que varias variables tienen una distribución desviada a la izquierda. Esto se ve en los gráficos de dispersión porque los puntos están concentrados en la esquina inferior izquierda, lo que hace muy difícil estimar si hay una asociación o no. Por otra parte, hay variables que tienen bastante correlación entre si, lo que podría generar problemas debido a la multicolinearidad, que habría que tomar en cuenta.

(d) Consideremos transformar las variables originales mediante alguna de las potencias de la escalera de Tukey (Tukey ladder of powers). Nos centraremos en las variables químicas y la concentración media de mercurio. La función transformTukey() del paquete rcompanion nos puede servir. Mostrar gráficamente que con las variables transformadas (si es necesario), mejora la relación lineal entre las variables químicas y la concentración de mercurio.

```
# realizo la transformación con la escalera de Tukey, y vuelvo a realizar un plot de
# dispersión.
tukey_lad <-
   mercbass %>%
      select(Alkalinity, pH, Calcium, Chlorophyll, Min, `Avg Mercury`, Max) %>%
      map_df(function(x) transformTukey(x, plotit = F, quiet = T))
```

```
tukey_lad %>%
   ggpairs(., lower = list(continuous = wrap("points", alpha = 0.75, size = .85)))
```



Comparativamente, las curvas de dispersión mejoran mucho en la medida de que se acercan mas a una distribución normal. Por otra parte, en los gráficos de dispersión es posible observar que es mucho mas fácil ver el tipo de correlación 2 a 2 que hay entre las diferentes variables. El problema de colinearidad es mas visible con los datos transformados.

(e) Sean $X_1 = f(Alkalinity)$, $X_2 = pH$, $X3 = f(Calcium)$, $X_4 = f(Chlorophyll)$, $Y_1 = g(Min)$, $Y_2 = g(Avg.Mercury)$ y $Y_3 = g(Max)$, donde f y g son las transformaciones propuestas en el apartado anterior. Calcular la matriz de varianzas-covarianzas de las variables $(X_1, X_2, X_3, X_4, Y_1, Y_2, Y_3)$ con la matriz de centrado H. Comprobar que da el mismo resultado que con la función cov().

```
# Primero creo la base con las transformaciones propuestas. Me ha costado un
# poco comprender por que se proponían sólo dos funciones de transformación,
# cuando la función de transformTukey utiliza una lambda diferente para cada
# variable según se minimice el error W de la Prueba de Shapiro Wilks. Leyendo
# el foro entiendo que esta bien utilizar esta trasformación en la medida que
# el "escalón" utilizado es el mismo para las variables X y es otro diferente
# para las Y. Excluyo pH.

mercbass_tr <-
   mercbass %>%
   select(Alkalinity, pH, Calcium, Chlorophyll, Min, `Avg Mercury`, Max) %>%
   map_at(vars(-pH), function(x)
      transformTukey(x, plotit = F, quiet = T)) %>%
   as_tibble()
```

```r
n <- dim(mercbass_tr)[1]
```

Calculo de la COV con los datos centrados directamente (método 1) y con la matriz de centrado (método 2)

```r
# Método 1
# calculamos la matriz de datos centrados con dplyr
cent <-
mercbass_tr %>%
   map_df(function(x) x - mean(x)) %>%
   as.matrix()

# calculamos la matriz var-cov
S_1 <-
t(cent) %*% cent * 1/(n - 1)
S_1
```

```
##                Alkalinity          pH      Calcium  Chlorophyll
## Alkalinity   0.0008049344  0.03034369  0.001374573  0.001092579
## pH           0.0303436948  1.66010160  0.057059830  0.060840453
## Calcium      0.0013745731  0.05705983  0.003154455  0.002042494
## Chlorophyll  0.0010925794  0.06084045  0.002042494  0.005182853
## Min         -0.0034435161 -0.13742095 -0.005178348 -0.005456653
## Avg Mercury -0.0046622401 -0.18794630 -0.007192592 -0.008819832
## Max         -0.0060973898 -0.24952472 -0.008984107 -0.011454331
##                      Min  Avg Mercury          Max
## Alkalinity  -0.003443516 -0.004662240 -0.006097390
## pH          -0.137420947 -0.187946303 -0.249524719
## Calcium     -0.005178348 -0.007192592 -0.008984107
## Chlorophyll -0.005456653 -0.008819832 -0.011454331
## Min          0.028678463  0.037147860  0.046897438
## Avg Mercury  0.037147860  0.056527538  0.074920487
## Max          0.046897438  0.074920487  0.110182066
```

```r
# Método 2
# calculamos primero la matriz de centrado
H <- diag(n) - 1/n * matrix(data = 1, nrow = n, ncol = n)

# con la matriz de centrado calculamos la matriz var-cov
S_2 <-
(1/(n-1)) * t(as.matrix(mercbass_tr)) %*% H %*% as.matrix(mercbass_tr)
S_2
```

```
##                Alkalinity          pH      Calcium  Chlorophyll
## Alkalinity   0.0008049344  0.03034369  0.001374573  0.001092579
## pH           0.0303436948  1.66010160  0.057059830  0.060840453
## Calcium      0.0013745731  0.05705983  0.003154455  0.002042494
## Chlorophyll  0.0010925794  0.06084045  0.002042494  0.005182853
## Min         -0.0034435161 -0.13742095 -0.005178348 -0.005456653
## Avg Mercury -0.0046622401 -0.18794630 -0.007192592 -0.008819832
## Max         -0.0060973898 -0.24952472 -0.008984107 -0.011454331
##                      Min  Avg Mercury          Max
## Alkalinity  -0.003443516 -0.004662240 -0.006097390
## pH          -0.137420947 -0.187946303 -0.249524719
## Calcium     -0.005178348 -0.007192592 -0.008984107
## Chlorophyll -0.005456653 -0.008819832 -0.011454331
```

```
## Min          0.028678463  0.037147860  0.046897438
## Avg Mercury  0.037147860  0.056527538  0.074920487
## Max          0.046897438  0.074920487  0.110182066
```

```r
# chequeamos si son iguales a lo obtenido con `cov`
all.equal(cov(mercbass_tr), S_1 , S_2)
```

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

- Método 1: En primer lugar calculamos la matriz centrada. Con `map` es facil calcular $(x_i - \bar{x})$ para las variables. Luego calculamos la covarianza con: $S = \frac{1}{n-1}\bar{X}'\bar{X}$
- Método 2: Calculamos la matriz de centrado con: $H = I_n - \frac{1}{n}11^\top$. Con la matriz de centrado calculamos la matriz var-cov con: $S = \frac{1}{n-1}X'HX$.

En ambos casos utilizamos en el denominador $n-1$, que calcula el estimador de la covarianza de una muestra i.i.d. y que es el método que utiliza el operador `cov` en R.

(f) Consideremos las variables $U = 0.4X_1 + 0.2X_2 + 0.2X_3 + 0.2X_4$ y $V = (Y_1 + Y_2 + Y_3)/3$. Calcular las medias y las varianzas de U y V en función de las medias, varianzas y covarianzas de las variables X_i y Y_j . También la correlación entre las variables U y V.

```r
# EN primer lugar transformo la base a una matriz
X <- as.matrix(mercbass_tr)


# Calculamos el vector de medias para todas las variables
X_bar <- map_dbl(mercbass_tr, mean)

# Utilizo dos matrices de transformación para obtener las variables U y V Con
# estas matrices de transformación es fácil poder obtener la información que se
# solicita.
T_u <- c(0.4, 0.2, 0.2, 0.2, 0, 0, 0)
T_v <- c(0, 0, 0, 0, 1/3, 1/3, 1/3)

# Media la variable U
(m_u <- X_bar %*% T_u)
```

```
##           [,1]
## [1,] 0.9953386
```

```r
# Varianza de la Variable U
(s_u <- T_u %*% S_1 %*% T_u)
```

```
##           [,1]
## [1,] 0.0817115
```

```r
# Media de la variable V
(m_v <- X_bar %*% T_v)
```

```
##           [,1]
## [1,] 0.7367344
```

```r
# Varianza de la variable V
(s_v <- T_v %*% S_1 %*% T_v)
```

```
##            [,1]
## [1,] 0.05703552
```

```r
# covarianza de U y V
(s_uv <- T_u %*% S_1 %*% T_v)
```

```
##             [,1]
## [1,] -0.04335894
```

```r
S_uv <- matrix(c(s_u, s_uv, s_uv, s_v), nrow = 2)
colnames(S_uv) <- c("U", "V")
rownames(S_uv) <- c("U", "V")

# Matriz de covarianza de UV
S_uv
```

```
##             U           V
## U  0.08171150 -0.04335894
## V -0.04335894  0.05703552
```

```r
# Coeficiente de correlacion
(r <- s_uv / (sqrt(s_u) * sqrt(s_v)))
```

```
##            [,1]
## [1,] -0.6351326
```

(g) Hallar las combinaciones lineales $U = a_1X_1 + a_2X_2 + a_3X_3 + a_4X_4$ y $V = b_1Y_1 + b_2Y_2 + b_3Y_3$ de forma que la correlación sea máxima. La solución pasa por hacer un Análisis de la correlación canónica2 (ACC) que se puede resolver paso a paso con R. Para ello necesitamos los valores y vectores propios de las matrices:

$$S_{XX}^1 S_{XY} S_{YY}^1 S_{YX}$$

$$S1YYSYXS1XXSXY$$

donde S es la matriz de varianzas-covarianzas de las variables. El primer valor propio de ambas matrices coincide y la correlación máxima es la raíz cuadrada de ese valor. Las combinaciones lineales que buscamos son los vectores propios de cada matriz asociados al primer valor propio común. Sin embargo, como exigimos que la combinación lineal $a'x$ verifique $a'S_{XX}a = 1$ y que la combinación lineal $b'y$ verifique $b'S_{YY}b = 1$, habrá que dividir cada vector propio por la raíz cuadrada de $u'S_{XX}u$ y la raíz cuadrada de $v'S_{YY}v$ respectivamente, donde u y v son los vectores propios asociados al primer valor propio de cada una de las matrices. Dado que las variables químicas del agua de los lagos están en las mismas unidades, salvo el pH, ¿cual es la variable mejor correlacionada según el ACC con la concentración de mercurio? ¿Cual de las tres variables de concentración de mercurio presenta mayor correlación con las variables químicas según el ACC?

http://pensamientoestadistico.com/analisis-correlacion-canonica/

```r
# Primero creo las matrices de covarianza de xx yy xy y yx con la matriz de
# covarianza de toda la base
S <- cov(mercbass_tr)

S_xx <- S[1:4, 1:4]
S_yy <- S[5:7, 5:7]
S_xy <- S[1:4, 5:7]
S_yx <- S[5:7, 1:4]

# Calculo de  los eigenvalues
A_eig <- eigen(solve(S_xx) %*% S_xy %*% solve(S_yy) %*% S_yx)
```

```r
B_eig <- eigen(solve(S_yy) %*% S_yx %*% solve(S_xx) %*% S_xy)

# Creo un pae de funciones para no tener que repetir para cada vector
get_a_vec <- function(eigen, col = 1){
   a <- eigen$vectors[,col]
   out <- a /sqrt(abs(t(a) %*% S_xx %*% a))
   return(out)
}

get_b_vec <- function(eigen, col = 1){
   b <- eigen$vectors[,col]
   out <- b /sqrt(abs(t(b) %*% S_yy %*% b))
   return(out)
}


# Valores de X, con los tres primeros vectores. EL primer vector (V1) es el que
# mas optimiza la correlación
sapply(c(1,2,3), function(x) get_a_vec(A_eig, x)) %>%
   as_tibble() %>%
   mutate(rowname = colnames(mercbass_tr)[1:4]) %>%
   column_to_rownames()
```

```
##                     V1          V2         V3
## Alkalinity  40.6929810 -20.9266497 -8.7835214
## pH           0.1573309  -0.8102676 -0.4745532
## Calcium     -8.8152230  18.2180884 27.8095926
## Chlorophyll  1.6922728  16.0611638 -8.2255778
```

```r
# Valores de Y. EL primer vector (V1) es el que mas optimiza la correlacion
sapply(c(1,2,3), function(x) get_b_vec(B_eig, x)) %>%
   as_tibble() %>%
   mutate(rowname = colnames(mercbass_tr)[5:7]) %>%
   column_to_rownames()
```

```
##                     V1         V2        V3
## Min         -4.7191402 -14.630659  5.416316
## Avg Mercury  0.6099592  19.300251  6.470491
## Max         -1.1574572  -6.428885 -7.839836
```

Por lo que entiendo, la variable química que mejor correlaciona con las Y es `Alkalinity`. La variable de concentración de Mercurio que mejor correlaciona es `Min`.

(h) Comprobar con alguna función de R que el resultado del apartado anterior es correcto. Nota: La función cc() del paquete CCA proporciona las correlaciones y los vectores canónicos correctos. La función cancor() da unos vectores canónicos que hay que multiplicar por $\sqrt{n1}$, donde n es el tamaño de la muestra.

```r
pacman::p_load(CCA)

merc_cc <- cc(mercbass_tr[,1:4], mercbass_tr[,5:7])
merc_cc$xcoef
```

```
##                   [,1]        [,2]        [,3]
```

```
## Alkalinity  -40.6929810  20.9266497   8.7835214
## pH           -0.1573309   0.8102676   0.4745532
## Calcium       8.8152230 -18.2180884 -27.8095926
## Chlorophyll  -1.6922728 -16.0611638   8.2255778
```

```
merc_cc$ycoef
```

```
##                   [,1]       [,2]      [,3]
## Min          4.7191402 -14.630659  5.416316
## Avg Mercury -0.6099592  19.300251  6.470491
## Max          1.1574572  -6.428885 -7.839836
```

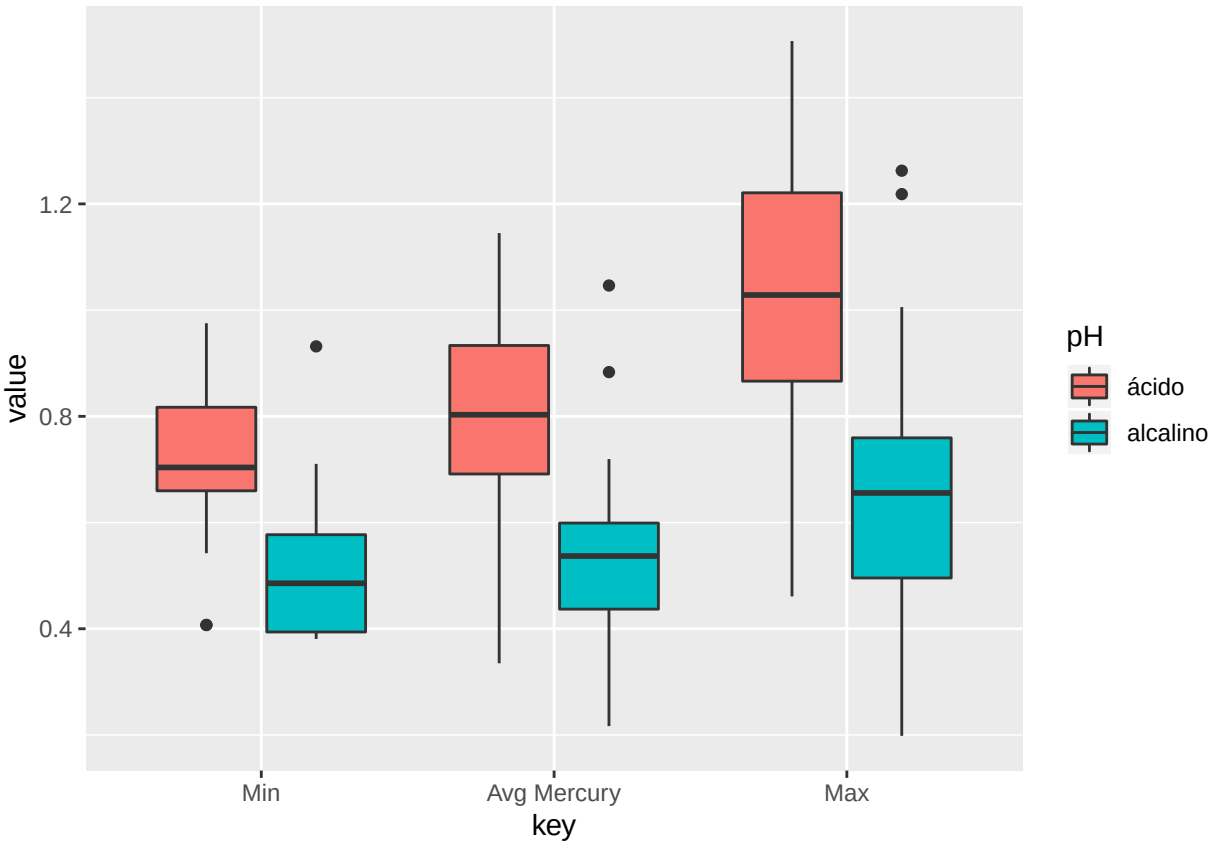
Los valores corresponden! Hay una inversión el los signos, pero esto no tiene efecto en el resultado.

Ejercicio 2

En este ejercicio vamos a comparar conjuntamente las tres variables Y_j respecto al pH de los lagos según sean ácidos (pH < 7) o alcalinos (pH > 7). Los lagos con valor de pH neutro ($= 7$) no pertenecen a ninguno de los dos grupos y no deben intervenir en la comparación.

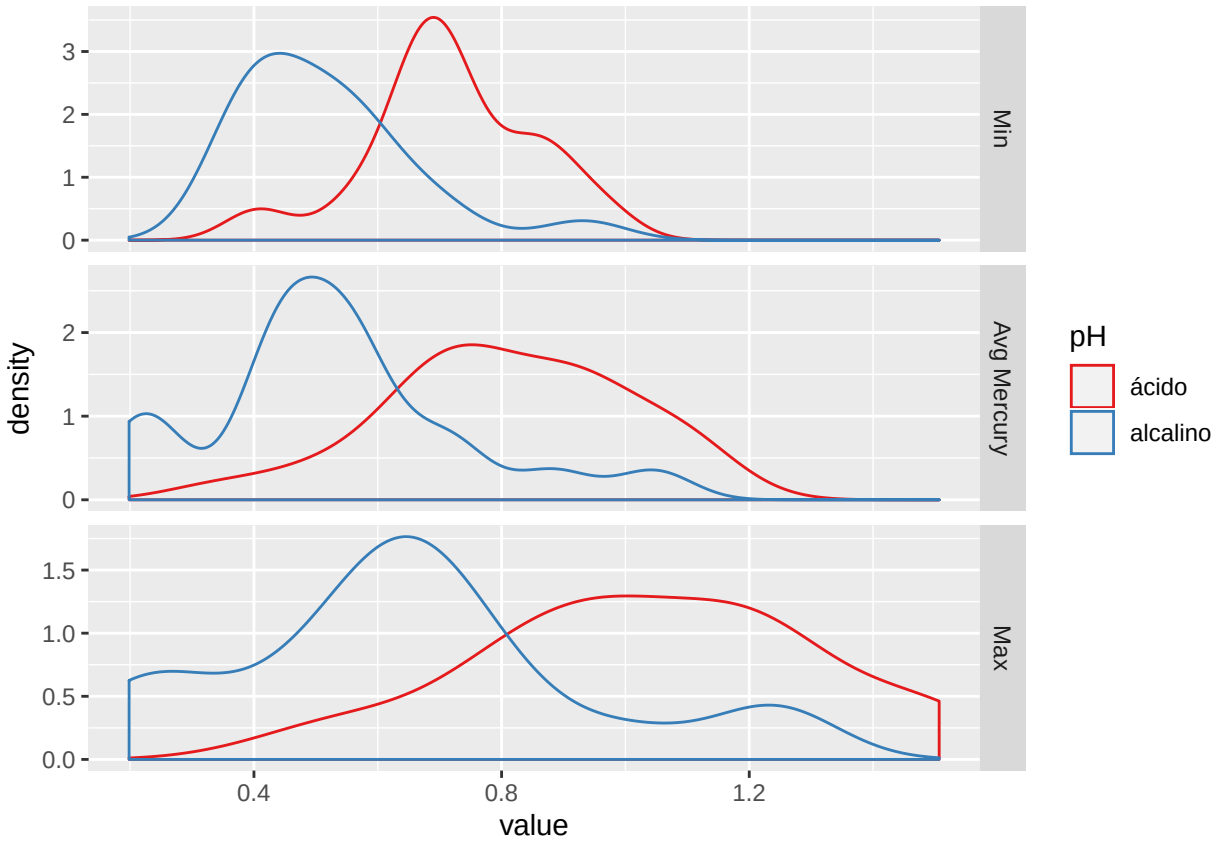
(a) Dibujar un único boxplot múltiple con las tres variables (Y1, Y2, Y3) en la misma escala que comparen los valores en los dos grupos de lagos (ácidos y alcalinos).

```
# creo nueva base, cambiando la variable pH de numérica a categórica.
mercbass_cat <-
mercbass_tr %>%
   mutate(pH = case_when(
      pH > 7 ~ 'alcalino',
      pH < 7 ~ 'ácido'
   ),
   pH = factor(pH, levels = c("ácido", "alcalino"))) %>%
   na.omit()                    # quita las observaciones con NA


# CReo un boxplot con ggplot
mercbass_cat %>%
   select(Min, `Avg Mercury`, Max, pH) %>%
   gather(key, value, -pH) %>%
   mutate(key = factor(key, levels = c('Min', 'Avg Mercury', 'Max'))) %>%
   ggplot(aes(key, value, fill = pH)) +
   geom_boxplot()
```



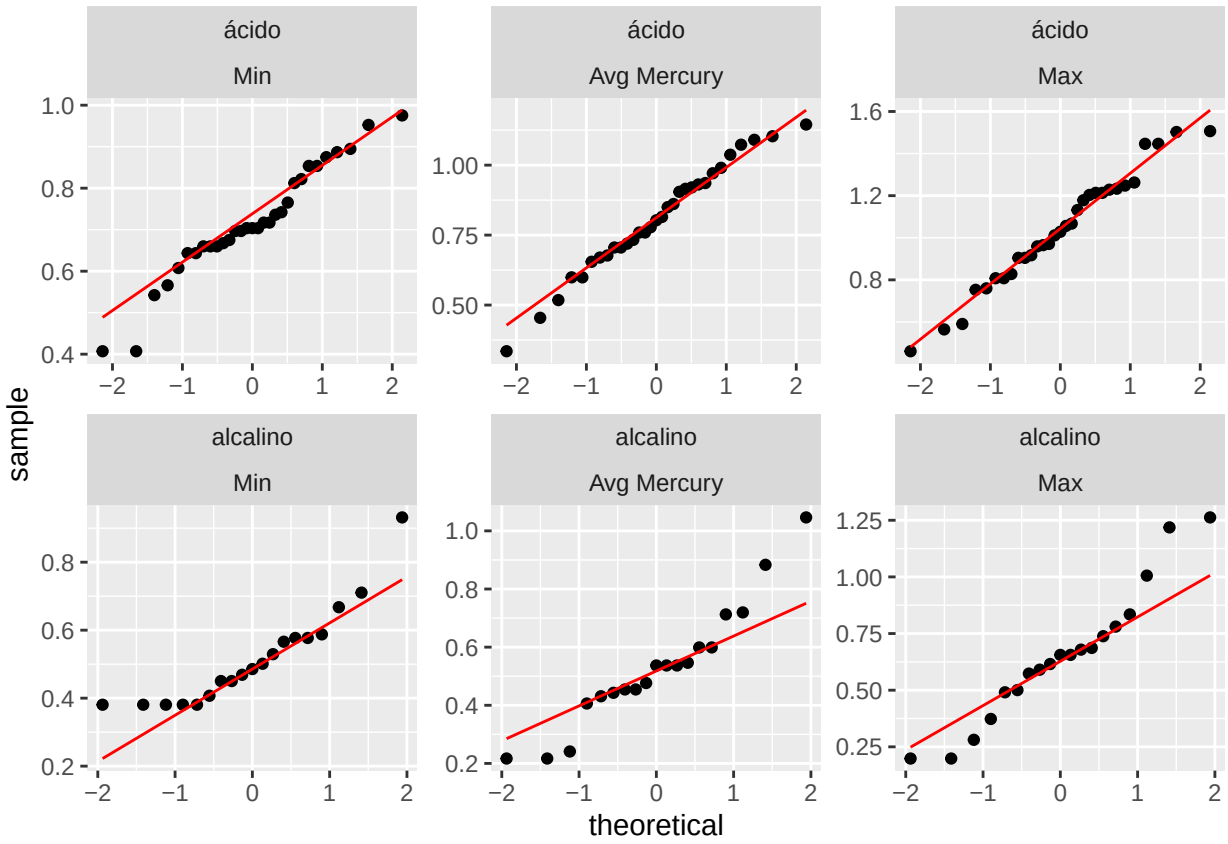
(b) Estudiar gráficamente y con algún test la normalidad univariante y multivariante (si es posible) en cada uno de los grupos. La normalidad multivariante se puede estudiar con la asimetría y la kurtosis multivariante de Mardia gracias a la función mvn del paquete MVN. La misma función permite el estudio univariante y también hace los gráficos. Acompañar el test con un gráfico qq-plot de ajuste a la distribución ji-cuadrado de las distancias de Mahalanobis (clásicas y robustas) de los datos a la media en cada grupo. Comentar el resultado en cuanto a los tests y en cuanto a la presencia de datos atípicos.

```
# creamos la version long de la base para poder graficar con ggplot.
long_dt <-
mercbass_cat %>%
   select(pH, Min, `Avg Mercury`, Max) %>%
   gather(key, value, -pH) %>%
   mutate(key = factor(key, levels = c('Min', 'Avg Mercury', 'Max')))

# Genero un plot de densidad para las diferentes variables para las tres variables Y.
long_dt %>%
   ggplot(aes(value, color = pH)) +
   geom_density(alpha = .5) +
   facet_grid( key ~., scales = "free") +
   scale_color_brewer(type = "qual", palette = 6)
```



En primer lugar construyo una gráfica de densidad para Y_i según cada `pH`. Si bien los histogramas o gráficas de densidad no son la mejor herramienta para valorar normalidad, creo que son una buena primera aproximación. En este caso podemos observar que las distribuciones de los lagos ácidos, se acercan mejor a una distribución normal. Los lagos alcalinos presentan una desviación a la izquierda, sobretodo en la variable `Min`.

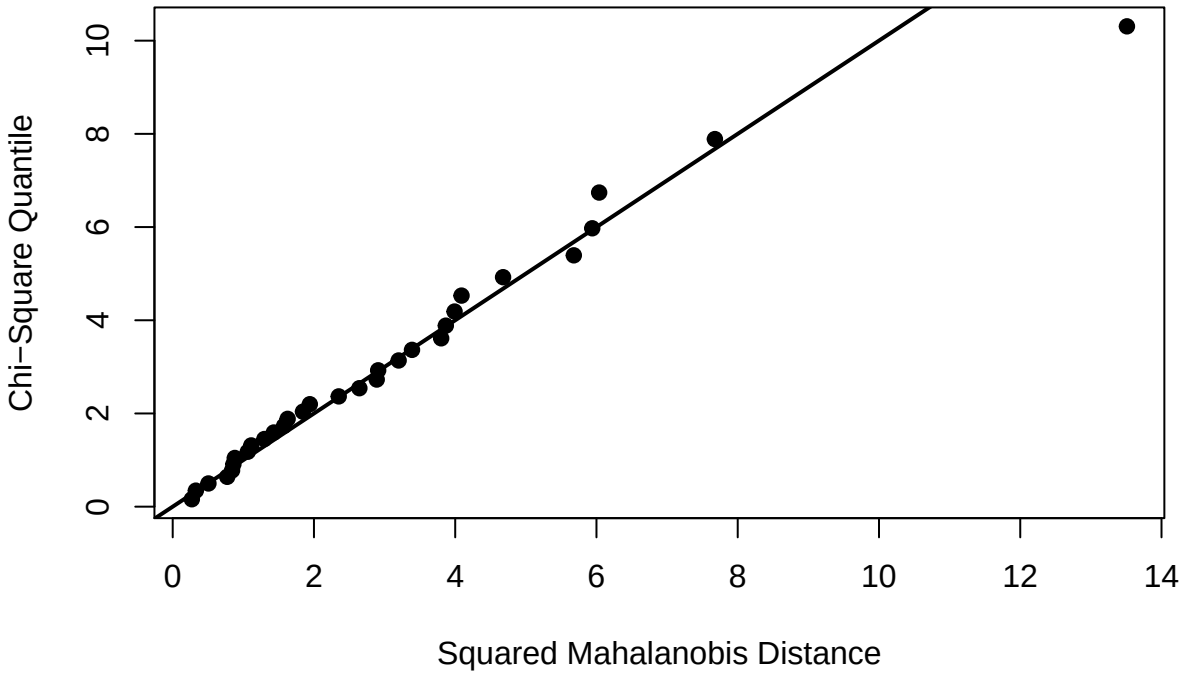
```
# Q_Q plots para valorar normalidad
long_dt %>%
   ggplot(aes(sample = value)) +
   stat_qq() +
   stat_qq_line(color = "red") +
   facet_wrap(pH~ key, scales = "free")
```



Una mejor herramienta gráfica univariable, son los gráficos $Q-Q$ que permiten graficar las variable v/s el cuantil teórico de una distribución normal. En este caso podemos ver como hay un mejor ajuste de las variables para la categoría de los lagos `ácidos`, sobre todo para Y_{max} y $Y_{AvgMercury}$, y en menor grado para Y_{Min} que tiene una desviación derecha y una "depresión" en los valores mas bajos.

En el caso de los `alcalinos` las tres distribuciones se alejan de la normal, siendo las tres picudas y desviadas a la izquierda.

```
# seleccionamos las variabels y filtramos los lagos acidos
Y_acid <-
   mercbass_cat %>%
   select(pH, Min, `Avg Mercury`, Max) %>%
   filter(pH == "ácido") %>%
   select(-pH)

# seleccionamos las variabels y filtramos los lagos alcalinos
Y_alc <-
   mercbass_cat %>%
   select(pH, Min, `Avg Mercury`, Max) %>%
   filter(pH == "alcalino") %>%
   select(-pH)

# Test de Mardia para los lagos ácidos , y su correspondiente grafica Q-Q
mvn_acid <- MVN::mvn(Y_acid, multivariatePlot = "qq")
```



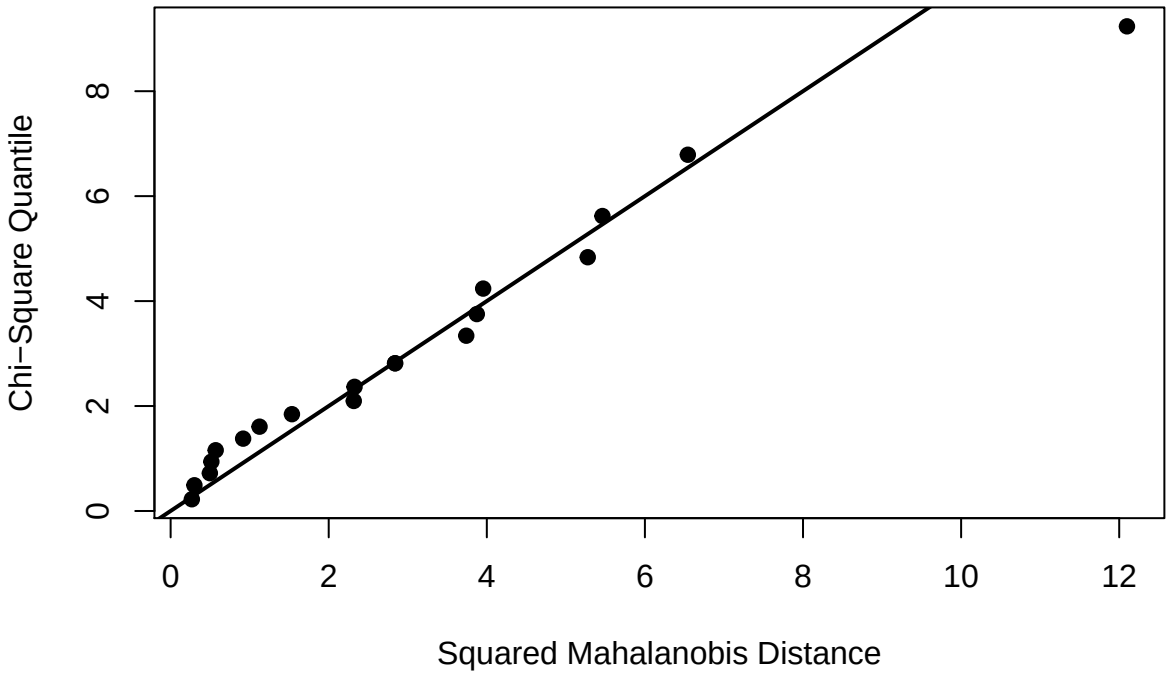

```
# Test univariante de Shapiro Wilk para los lagos ácidos
mvn_acid$univariateNormality
```

```
##           Test    Variable Statistic   p value Normality
## 1 Shapiro-Wilk     Min        0.9583    0.2630    YES
## 2 Shapiro-Wilk Avg Mercury    0.9818    0.8599    YES
## 3 Shapiro-Wilk     Max        0.9749    0.6612    YES
```

```
# Test multivariante de Mardia para los lagos ácidos
mvn_acid$multivariateNormality
```

```
##              Test         Statistic            p value Result
## 1 Mardia Skewness  19.8489143613569 0.0307145282042468     NO
## 2 Mardia Kurtosis 0.608196620153054  0.543057075990832    YES
## 3             MVN              <NA>               <NA>     NO
```

```
# Test de Mardia para los lagos alcalinos , y su correspondiente gráfica Q-Q
mvn_alc <- MVN::mvn(Y_alc, multivariatePlot = "qq")
```




```
# Test univariante de Shapiro Wilk para los lagos alcalinos
mvn_alc$univariateNormality
```

```
##           Test    Variable Statistic   p value Normality
## 1 Shapiro-Wilk     Min        0.8547    0.0080    NO
## 2 Shapiro-Wilk Avg Mercury    0.9335    0.2010    YES
## 3 Shapiro-Wilk     Max        0.9462    0.3395    YES
```

```
# Test multivariante de Mardia para los lagos alcalinos
mvn_alc$multivariateNormality
```

```
##              Test         Statistic             p value Result
## 1 Mardia Skewness  27.9257009069038 0.00185535654643472     NO
## 2 Mardia Kurtosis 0.813968009397013   0.415663277790487    YES
## 3             MVN              <NA>                <NA>     NO
```

En el caso de los lagos **ácidos**, el test de Shapiro Wilk para las variables individuales Y_i es coherente con la H0 para las tres variables, Y_{Min}, $Y_{AvgMercury}$ y Y_{Max}.

Por otra parte, el test de asimetria de Mardia permite rechazar la H0 de normalidad, por lo que **no* se cumple el criterio de Mardia para normalidad multivariable, que exije H0 para la asimetría y la kurtosis. Mirando la gráfica Q-Q se puede apreciar esta asimetría.

En el caso de los lagos **alcalinos**, el test de Shapiro Wilk, permite rechazar la H0 de normalidad para Y_min, lo que confirma la sospecha que teniamos mirando la curva de densidad y la gráfica Q__Q. El test de asimetría de Mardia tambien permite rechazar la H0, apoyando que la distribución multivariable no sigue a la normal.

(c) Comparar las matrices de covarianzas de los grupos con el test M de Box. También podemos comparar la variabilidad con un test de Levene múltiple. ¿Ambos métodos contrastan lo mismo? En caso de duda sobre la normalidad multivariante de los datos, proponer y calcular

un método multivariante para contrastar si hay diferencias de variabilidad entre los grupos.

```
Y_all <-
   mercbass_cat %>%
   select(Min:Max, pH)

# Utilizamos la versión del test M de Box de la librería 'heplots'.
mbox <- heplots::boxM(cbind(Min, `Avg Mercury`, Max) ~ pH, data = Y_all, cov = T)
mbox
```

```
##
##  Box's M-test for Homogeneity of Covariance Matrices
##
## data:  Y
## Chi-Sq (approx.) = 9.7819, df = 6, p-value = 0.1341
```

El test M de Box es un análogo de un test de likelihood ratio, y calcula la divergencia de las matrices de covarianza de las variables por grupos con la de todas las variables. Utiliza un estadístico de Chi cuadrado para valorar el grado de divergencia.

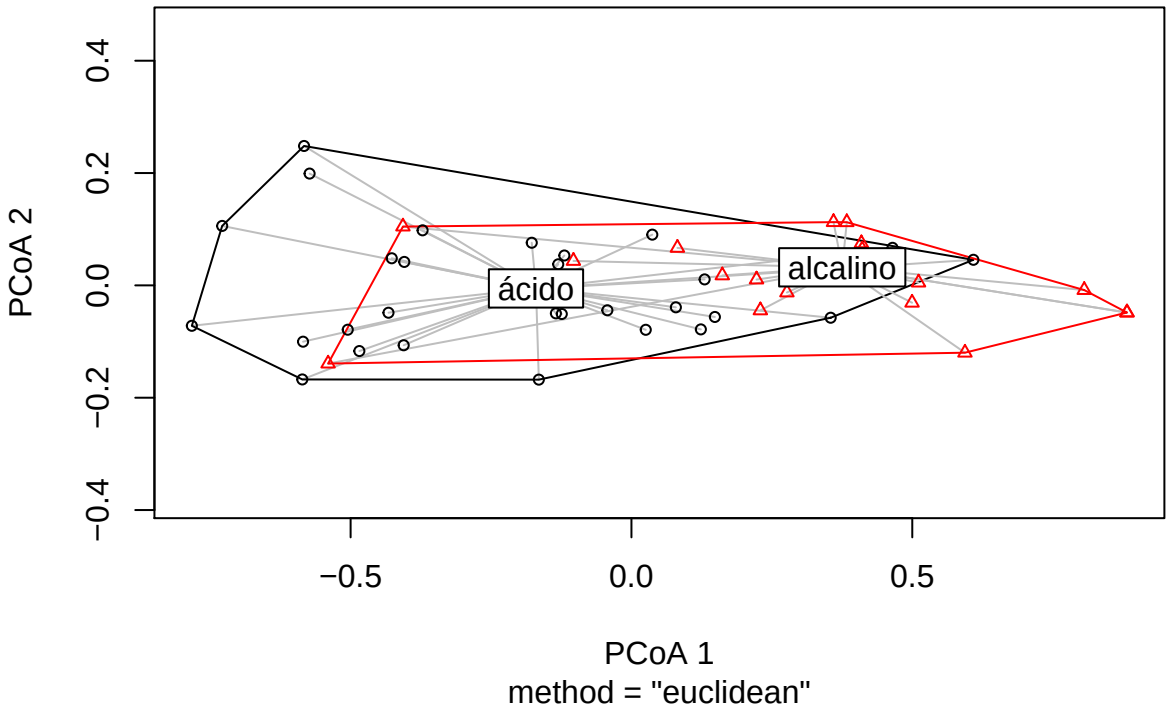
En este caso el estadístico no permite rechazar la H0 de homogeneidad de matrices de covarianza.

```
# Multiple Levene test
levene <- heplots::leveneTests(mercbass_cat[5:7], mercbass_cat$pH)
levene
```

```
## Levene's Tests for Homogeneity of Variance (center = median)
##
##             df1 df2 F value Pr(>F)
## Min           1  48  0.0270 0.8701
## Avg Mercury   1  48  0.0457 0.8316
## Max           1  48  0.0082 0.9281
```

El test de Levene puede detectar divergencias de las media en las variables. Calcula un estadístico F para valorar si las diferencias de la media son significativas entre los grupos. En este caso no se encuentran diferencias significativas en ninguna de las tres variables

```
# procedimiento multivatainte PERMDISP2 Esta descrito como un análogo
# multivariante del test de Levene calcula la distancia de la distancia promedio
# al centroide del grupo y luego valora con un ANOVA si la diferencia es
# significativa entre lso grupos. Posee ademas un método para TukeyHSD que
# permite calcular el IC de las diferencias en las medias de la distancia

d <- dist(Y_all[,1:3])

model <- vegan::betadisper(d, Y_all$pH)

plot(model)
```




```
anova(model)
```

```
## Analysis of Variance Table
##
## Response: Distances
##           Df  Sum Sq  Mean Sq F value Pr(>F)
## Groups     1 0.00043 0.000426   0.009 0.9248
## Residuals 48 2.26941 0.047279
```

```
TukeyHSD(model)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = distances ~ group, data = df)
##
## $group
##                         diff        lwr       upr     p adj
## alcalino-ácido -0.006013225 -0.1333917 0.1213653 0.9247758
```

En este caso el estadístico F no resulta significativo por lo que no rechazamos la hipótesis H0 de que no hay diferencias en la variabilidad de los datos. Ademas con el método de Tukey podemos calcular un intervalo de confianza que en este caso pasa por el 0.

```
# Otra alternativa es utilizar el test de Van Valen, descrito en el libro de
# Manly, que calcula diferencias de medias, del vector de distancias de
# diferencias con la mediana de las variables estandarizadas primero
# estandarizamos los datos

merc_scale <-
```

```
   mercbass_cat %>%
   arrange(pH) %>%
   select(-pH) %>%
   scale()


# vectores con las medianas de las variables escaladas
mat_med <-
   matrix(c(
      rep(apply(merc_scale[1:31,], 2, median), 19),
      rep(apply(merc_scale[32:50,], 2, median), 31)),
   ncol = 6,
   byrow = T)

# creamos un tibble con las distancias absolutas
abs_tibble <-
abs(merc_scale - mat_med) %>%
   as_tibble %>%
   mutate(dist_2 = rowSums(abs(merc_scale - mat_med)),
          cat = c(rep("ácido", 19), rep("alcalino", 31)))

# tabla de contingencia para mirar las medias de las distancias
# entre los dos grupos
abs_tibble %>%
   select(dist_2, cat) %>%
   group_by(cat) %>%
   summarise(media = mean(dist_2))
```

```
## # A tibble: 2 x 2
##   cat      media
##   <chr>    <dbl>
## 1 ácido     3.51
## 2 alcalino  4.79
```

```
# t.test para valorar si los grupos son diferentes.
abs_tibble %>%
   t.test(dist_2 ~ cat, data = ., alternative = "less")
```

```
##
##  Welch Two Sample t-test
##
## data:  dist_2 by cat
## t = -2.1468, df = 44.917, p-value = 0.01862
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##       -Inf -0.279829
## sample estimates:
##    mean in group ácido mean in group alcalino
##               3.506950               4.792532
```

Utilizo la metodología descrita en libreo de Manly, en la que en primer lugar se hace un calculo de la distancia absoluta de las variables estandarizadas a su mediana. Luego es posible realizar un t.test simple para valorar si hay diferencias entre la media de los vectores de distancia entre los dos grupos. En este caso realizo un t.test de una cola, ya que antes obtenemos que la distancia media a la mediana del grupo de lagos `ácido` es menor al grupo de lagos `alcalinos`. El t.test resulta con un estadístico que se aleja de manera significativa

de 0, por lo que se puede rechazar la H0 de que ambas medias son iguales, y por tanto apoya la hipótesis de que la varianza multivariable de las dos categorías de lagos son diferentes.

(d) Comparar de forma multivariante las medias de las variables (Y1, Y2, Y3) en los dos grupos de lagos según su pH. Realizar un test suponiendo normalidad multivariante y otro de permutaciones.

```
# PAra calcular diferencias en las medias de las variables, utilizo el test T2
# de Hotelling. La funcion 'hotelling.test' tiene asdemas la posibliadd de
# realizar la version del test con permutaciones

t2_1 <-
   Hotelling::hotelling.test(cbind(Min, `Avg Mercury`, Max) ~ pH, data = Y_all)
t2_2 <-
   Hotelling::hotelling.test(
      cbind(Min, `Avg Mercury`, Max) ~ pH,
      data = Y_all,
      perm = T,
      B = 1000,
      progBar = F
   )

t2_1
```

```
## Test stat:  9.0435
## Numerator df:  3
## Denominator df:  46
## P-value:  8.105e-05
```

```
# permutaciones
t2_2
```

```
## Test stat:  9.0435
## Numerator df:  3
## Denominator df:  46
## Permutation P-value:  0
## Number of permutations : 1000
```

El test T2 de Hotelling resulta en un estadístico con un nivel significativo con tendencia a 0, lo que apoya el rechazo de la H0 de igualdad de medias para las muestras multivariantes. Esto viene a apoyar lo que era visible en la gráfica de cajas de la sección (a) del Ejercicio 2

Ejercicio 3

Con la base de datos y las variables del ejercicio 1(e).

(a) Calcular las componentes principales con alguna función específica de R a partir de la matriz de correlaciones para que las variables no tengan pesos distintos. ¿Qué relación tienen estas componentes con los vectores propios de la matriz de correlaciones R? ¿Cuantas componentes parecen necesarias para tener una buena representación de los datos?

```
# utilizamos la función PCA de la librería FactoMineR
pca_mod <-
```

```
   mercbass_tr %>%
   PCA(graph = F, ncp = 7, scale.unit = T)

eigen(cor(mercbass_tr))$vectors
```

```
##            [,1]       [,2]         [,3]       [,4]        [,5]        [,6]
## [1,] -0.4043578 -0.2407269  0.319468673  0.0753211  0.13495293  0.80401043
## [2,] -0.3878179 -0.3591783 -0.001962334 -0.7654494 -0.30814428 -0.18549233
## [3,] -0.3592774 -0.4551757  0.371472844  0.5161587 -0.02367090 -0.49961581
## [4,] -0.3073051 -0.2901699 -0.861466876  0.2035307  0.17622283  0.05792167
## [5,]  0.3927787 -0.3720411 -0.110126593  0.2412738 -0.68663987  0.18728124
## [6,]  0.4022125 -0.4241831  0.038990051 -0.1280251  0.03391969  0.14291983
## [7,]  0.3827864 -0.4509015  0.064579972 -0.1611396  0.61854276 -0.10338569
##             [,7]
## [1,] -0.07852897
## [2,] -0.07287898
## [3,]  0.09563869
## [4,]  0.05835798
## [5,] -0.36113824
## [6,]  0.78664255
## [7,] -0.47617446
```

```
pca_mod$svd$V
```

```
##            [,1]      [,2]         [,3]       [,4]        [,5]        [,6]
## [1,]  0.4043578 0.2407269 -0.319468673 -0.0753211 -0.13495293  0.80401043
## [2,]  0.3878179 0.3591783  0.001962334  0.7654494  0.30814428 -0.18549233
## [3,]  0.3592774 0.4551757 -0.371472844 -0.5161587  0.02367090 -0.49961581
## [4,]  0.3073051 0.2901699  0.861466876 -0.2035307 -0.17622283  0.05792167
## [5,] -0.3927787 0.3720411  0.110126593 -0.2412738  0.68663987  0.18728124
## [6,] -0.4022125 0.4241831 -0.038990051  0.1280251 -0.03391969  0.14291983
## [7,] -0.3827864 0.4509015 -0.064579972  0.1611396 -0.61854276 -0.10338569
##             [,7]
## [1,]  0.07852897
## [2,]  0.07287898
## [3,] -0.09563869
## [4,] -0.05835798
## [5,]  0.36113824
## [6,] -0.78664255
## [7,]  0.47617446
```

Los vectores propios de la matriz de correlación se corresponden con la matriz `V` del procedimiento de descomposición en valores singulares del análisis de componentes principales.

Aunque existe una plétora de métodos para contestar la última pregunta, los cuatro métodos más utilizados son:

1. El criterio de Kaiser: se eligen las primeras componentes cuyos valores propios son superiores a 1.

```
pca_mod$eig
```

```
##        eigenvalue percentage of variance cumulative percentage of variance
## comp 1 4.96572071              70.938867                           70.93887
## comp 2 0.97610637              13.944377                           84.88324
## comp 3 0.58733904               8.390558                           93.27380
```

```
## comp 4 0.18529520          2.647074                95.92088
## comp 5 0.15719751          2.245679                98.16655
## comp 6 0.10443238          1.491891                99.65845
## comp 7 0.02390878          0.341554               100.00000
```

Según el método de Kaiser deberíamos elegir el componente 1 solamente.

---

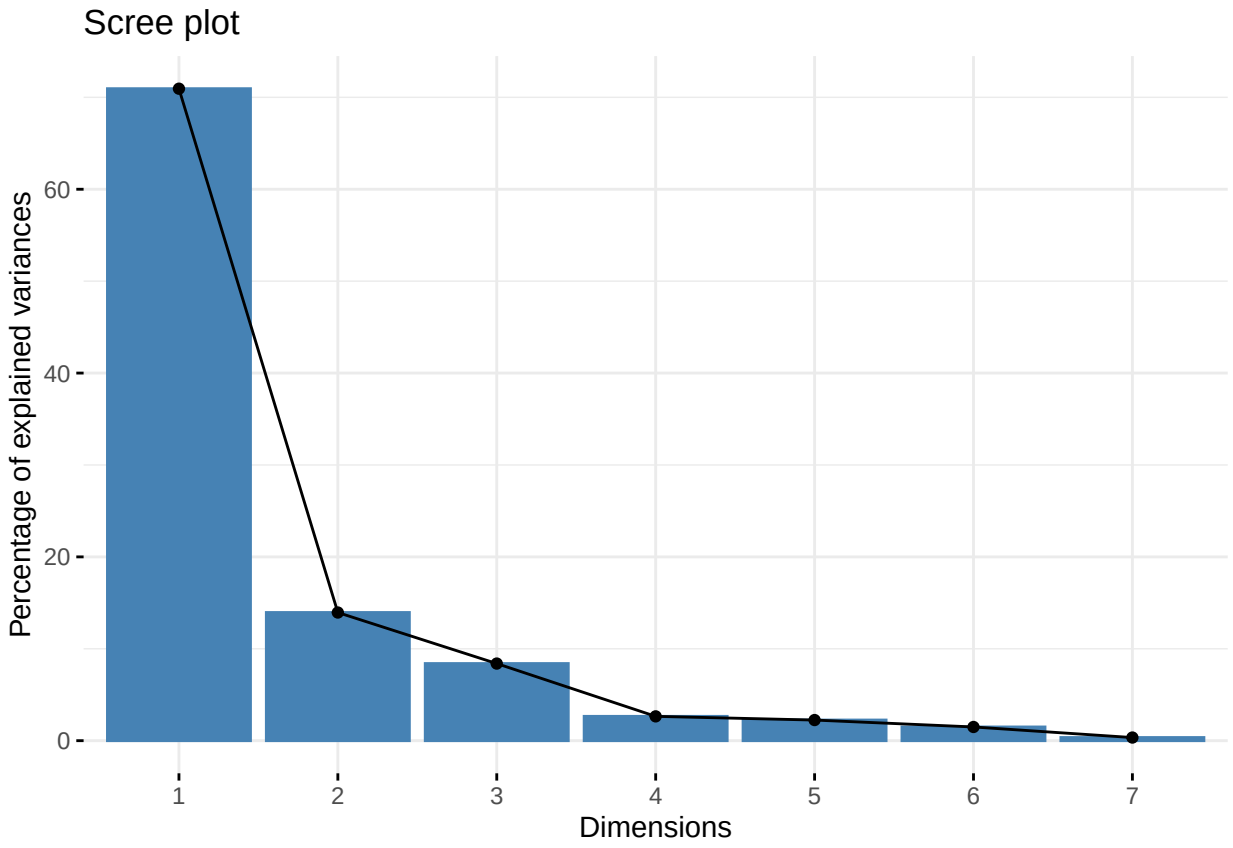
2. Varianza explicada: se eligen las primeras componentes que explican como mínimo el 70 o 80% de la varianza total.

Según este criterio podríamos elegir los dos primeros componentes.

---

3. Scree plot de Cattel (1966): Éste es un método gráfico por el que se eligen las componentes hasta el codo del gráfico.

```
fviz_screeplot(pca_mod)
```

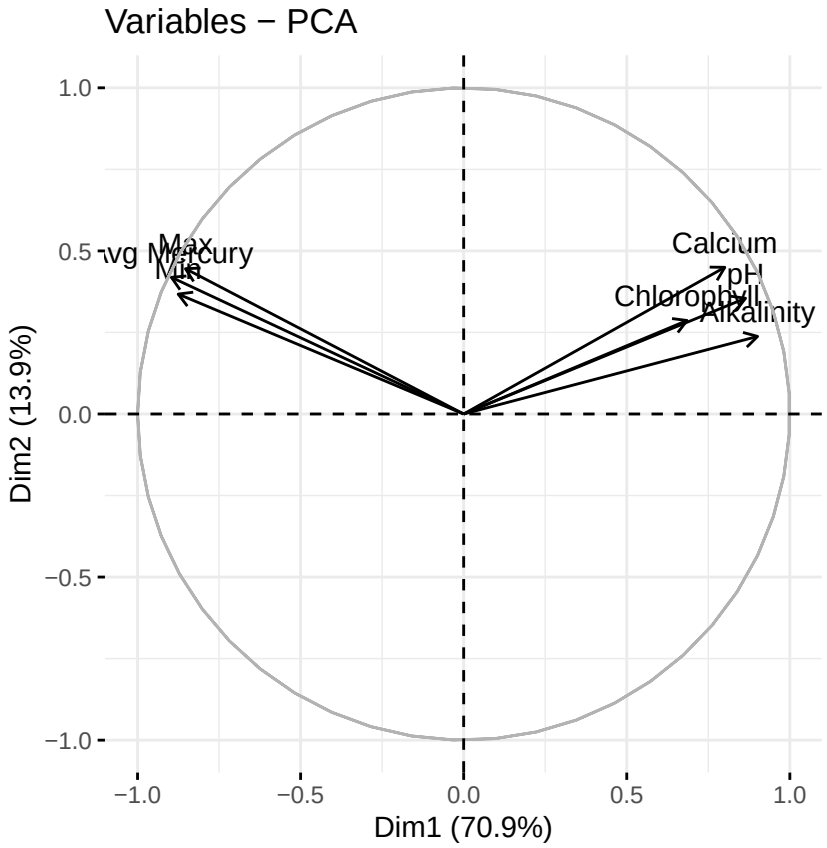


En este caso el codo del gráfico coincide con los dos primeros componentes. Luego la variablidad explicada por cada componente baja bastante por lo que se podrían obviar y considerar sólo los primeros dos

---

4. El criterio de la interpretabilidad: se trata de elegir todas las componentes que mejor recojan la esencia del significado de las variables y verificar que esa interpretación tiene sentido en los términos conocidos del problema bajo investigación. Seguramente, la solución es una combinación de los cuatro criterios.

```
fviz_pca_var(pca_mod, )
```



La interpretabilidad también hace pensar que dos dimensiones son suficientes y coherentes. En resumen, creo que en este caso son necesarias dos dimensiones del PCA.

(b) Interpretar las dos primeras componentes mediante las variables mejor relacionadas con cada una de ellas. Los gráficos de correlaciones con las variables originales pueden ayudar.

```
dimdesc(pca_mod, axes = c(1,2))
```

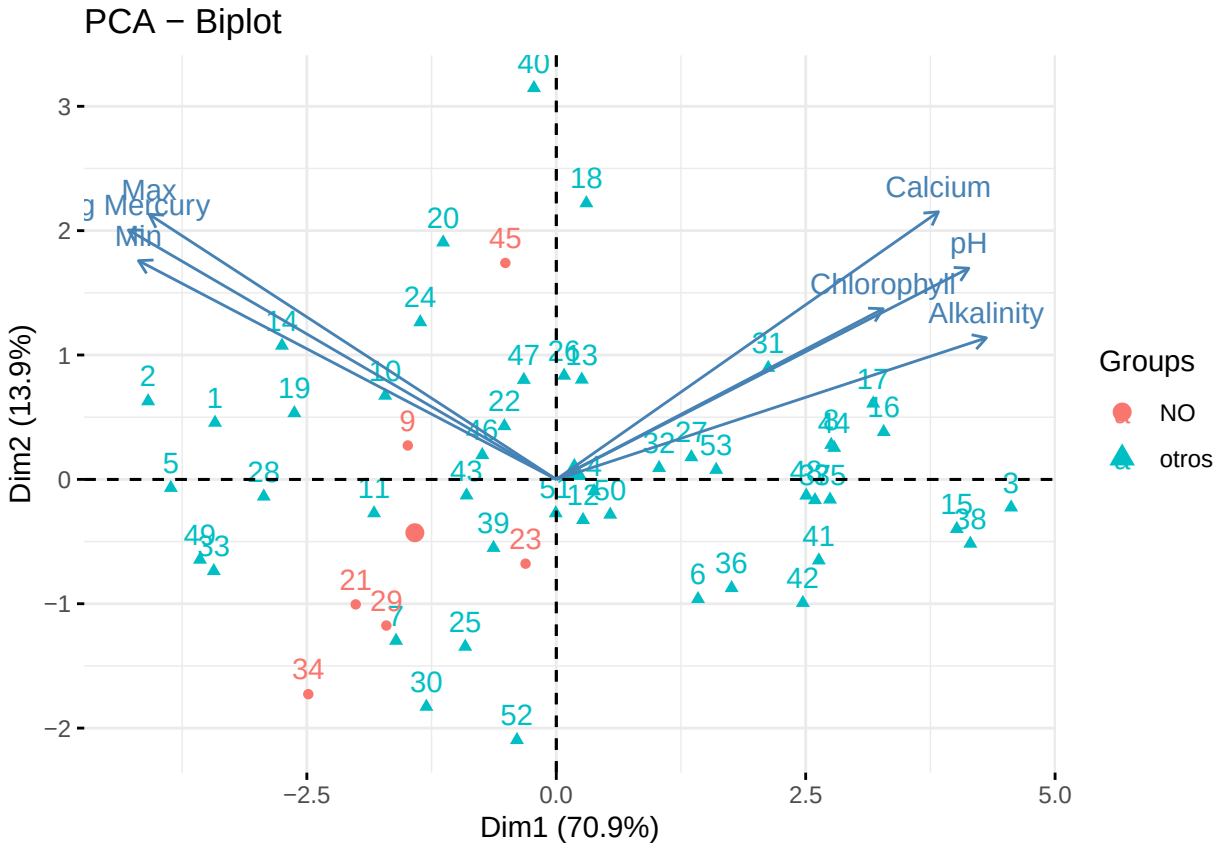
```
## $Dim.1
## $Dim.1$quanti
##             correlation      p.value
## Alkalinity    0.9010667 3.846464e-20
## pH            0.8642093 7.806709e-17
## Calcium       0.8006100 6.219650e-13
## Chlorophyll   0.6847955 1.558403e-08
## Max          -0.8529973 5.124625e-16
## Min          -0.8752641 1.028714e-17
## Avg Mercury  -0.8962862 1.207734e-19
##
##
## $Dim.2
## $Dim.2$quanti
##             correlation      p.value
## Calcium       0.4497049 0.0007294435
## Max           0.4454821 0.0008295805
## Avg Mercury   0.4190848 0.0017878524
```

```
## Min           0.3675695 0.0067763834
## pH            0.3548613 0.0091263848
## Chlorophyll   0.2866824 0.0374139605
```

Las variables que mas correlaciona con la Dimensión 1 son la `Alkalinity` y `pH` por un lado y de manera inversa, `Avg Mercury`, `Min` y `Max`. Esta dimensión diferencia claramente la acidez (y los otros químicos) a un polo y la concentración de mercurio en el otro polo inverso. Loa lagos con valores altos en esta dimensión tienden a ser mas alcalinos, a tener mas calcio y menos concentración de Mercurio. En cuanto a la dimensión 2, es visible que correlaciona casi todas las variables, por lo que esta dimensión se caracteriza por la correlación de la cantidad de químicos y la alcalinidad de los lagos. Los Lagos con valores altos en esta dimensión son mas alcalinos y tienen mas concentración de Mercurio, los con valores bajos son mas ácido y con menos mercurio.

(c) Los investigadores Canfield and Hoyer (1988) hallaron que el pH y la alcalinidad en los lagos de Florida generalmente crecen si se va del Noroeste al Sudeste y de las tierras altas a la costa del estado. Representar los grupos en un gráfico de dos dimensiones con la función PCA del paquete FactoMineR3. ¿Donde están situados en el gráfico los lagos del Noroeste? ¿Se ajusta este resultado con la hipótesis de los investigadores?

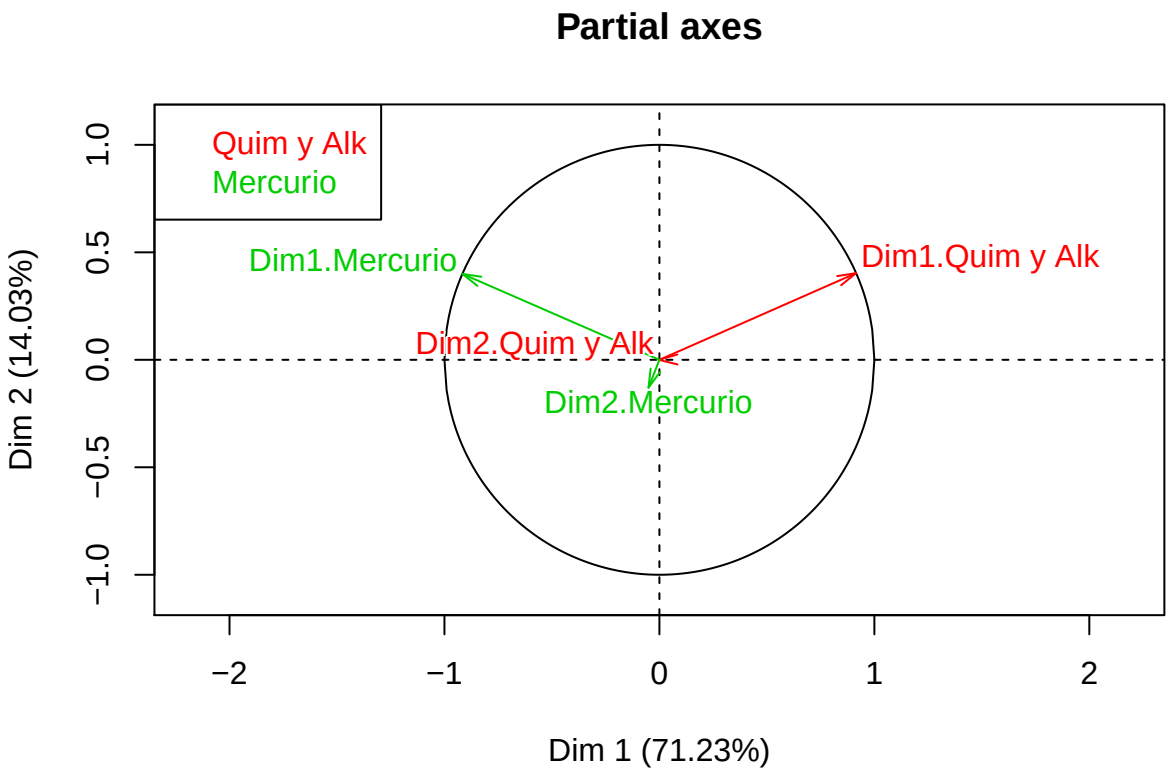
```r
factor_geo <-
   mercbass_tr %>%
   rownames_to_column() %>%
   mutate(geo = ifelse(rowname %in% c(9,21,23,29,34,45), "NO", "otros")) %>%
   pull(geo) %>%
   factor()


fviz_pca(pca_mod, habillage = factor_geo)
```

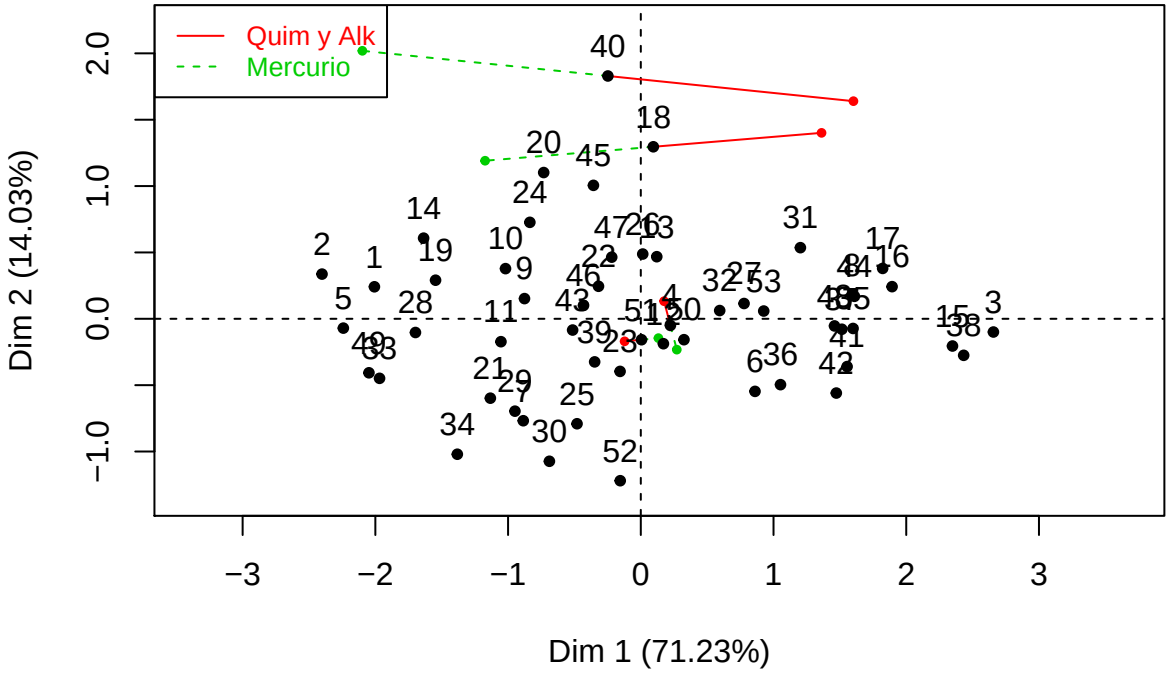


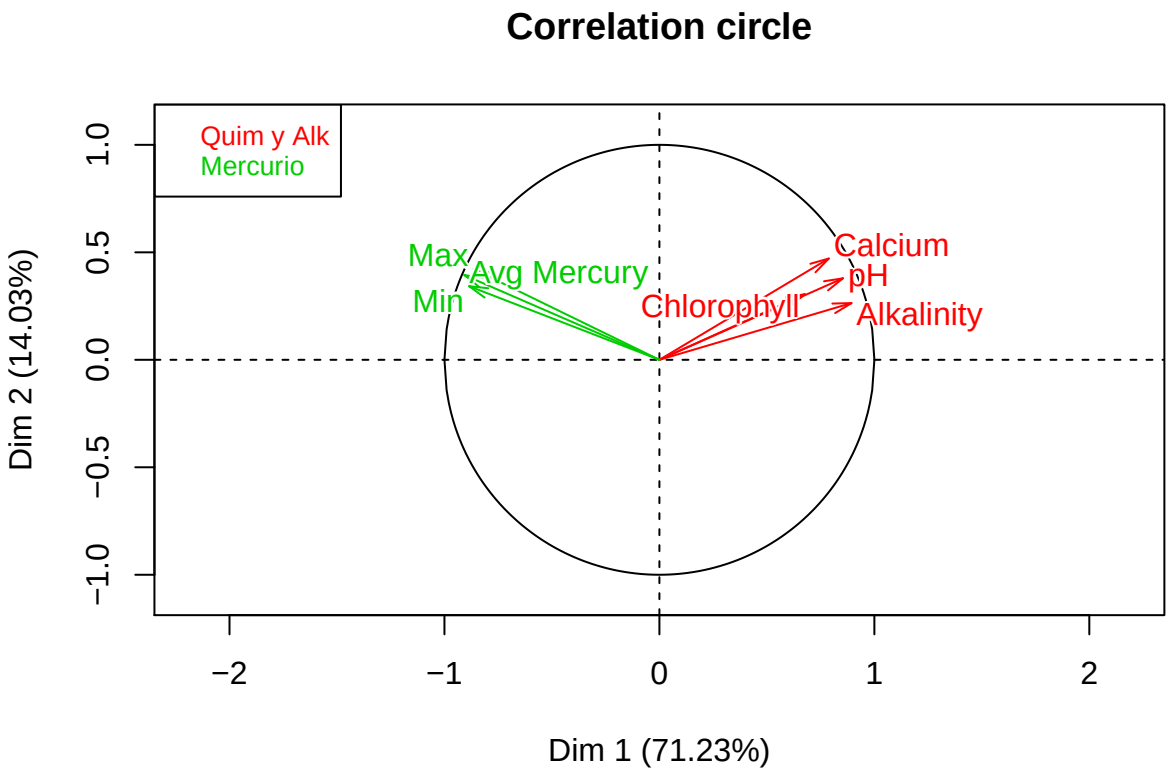
En este caso podemos ver los lagos del NO en rojo. Efectivamente estos lagos se encuentran mas concentrados en la zona inferior e izquierda del espacio del PCA, por lo que se podría interpretar que tienen concentraciones de mercurio alta y valores de pH - Alkalinidad bajo (valores negativos en Dim 1 y la mayoria valores negativos en Dim 2).

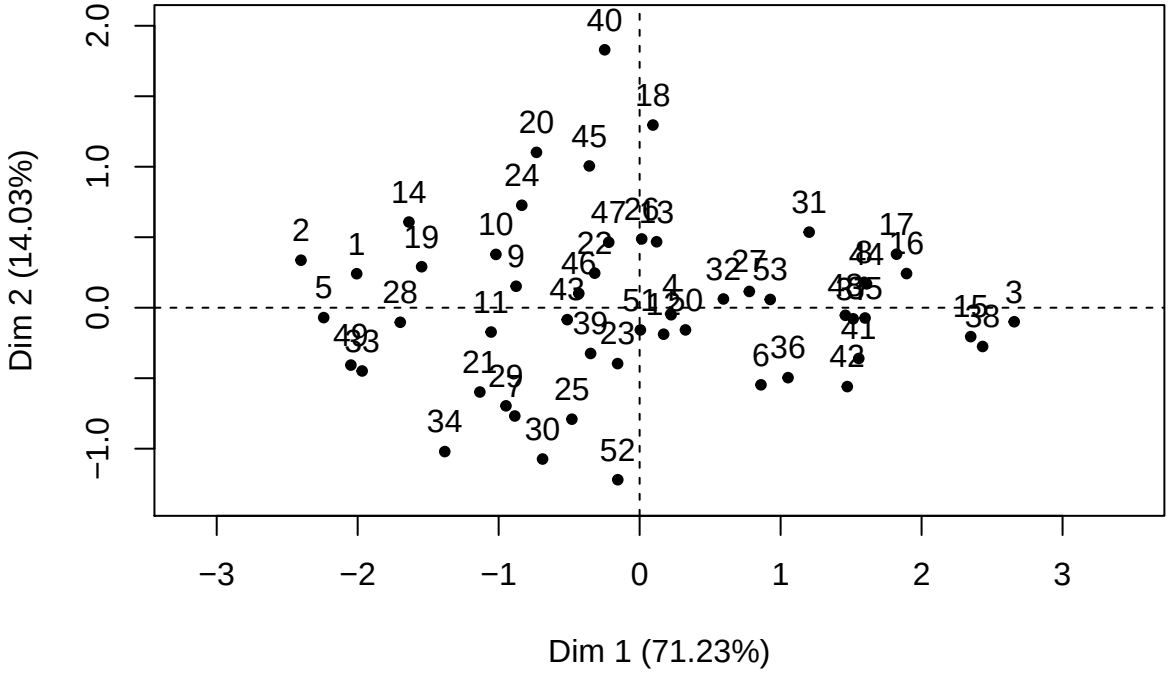
(d) Dado que las variables estudiadas en los apartados anteriores de este ejercicio están claramente en dos grupos, podemos realizar un Análisis factorial múltiple. ¿Difiere substancialmente el resultado del MFA del obtenido con las componentes principales?

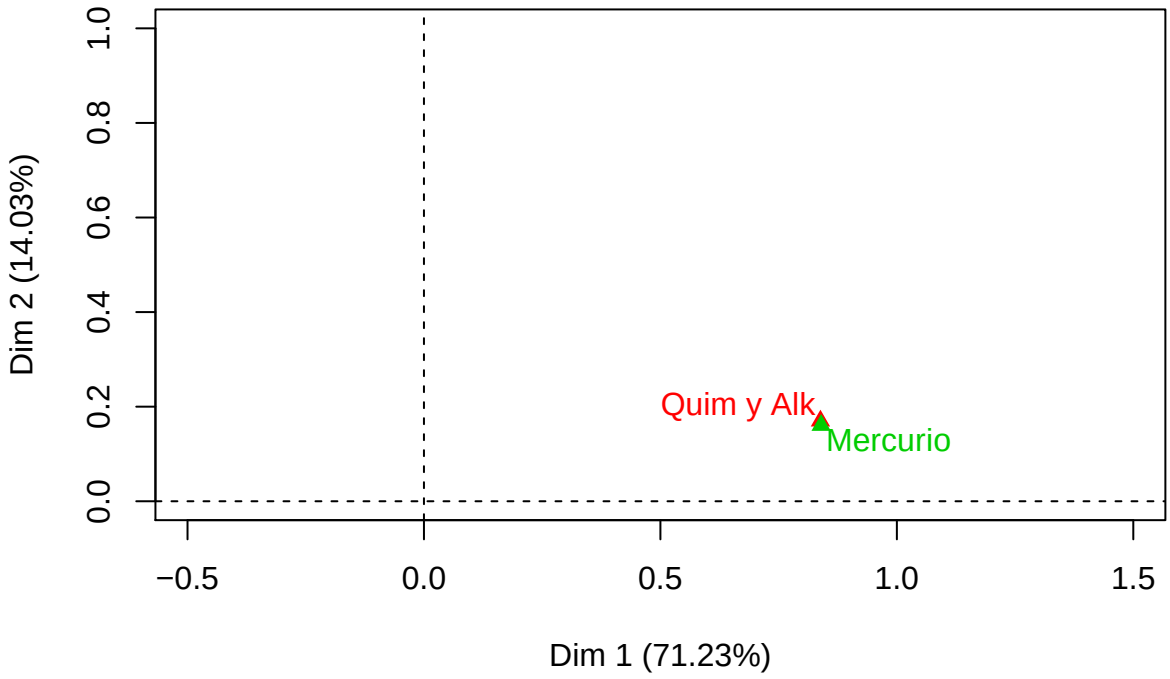
```
#par(mfrow = c(2,2))
mfa_mod <-
   mercbass_tr %>%
   MFA(
      .,
      group = c(4,3),
      name.group = c('Quim y Alk', 'Mercurio'),
      graph = T
   )
```


Partial axes
Quim y Alk
Mercurio
Dim1.Mercurio
Dim1.Quim y Alk
Dim2.Quim y Alk
Dim2.Mercurio
Dim 2 (14.03%)
Dim 1 (71.23%)


Individual factor map
Quim y Alk
Mercurio
Dim 2 (14.03%)
Dim 1 (71.23%)


Correlation circle
Quim y Alk
Mercurio
Max
Avg Mercury
Min
Calcium
pH
Chlorophyll
Alkalinity
Dim 2 (14.03%)
Dim 1 (71.23%)


Individual factor map
Dim 2 (14.03%)
Dim 1 (71.23%)




```
# descripción de las dimensiones
dimdesc(mfa_mod)
```

```
## $Dim.1
## $Dim.1$quanti
##             correlation      p.value
## Alkalinity    0.8937729 2.155849e-19
## pH            0.8538862 4.439953e-16
## Calcium       0.7880983 2.494523e-12
## Chlorophyll   0.6764912 2.696823e-08
## Max          -0.8649807 6.817015e-17
## Min          -0.8853298 1.364746e-18
## Avg Mercury  -0.9076973 7.117203e-21
##
##
## $Dim.2
## $Dim.2$quanti
##             correlation      p.value
## Calcium       0.4711903 0.0003692886
## Max           0.4220904 0.0016432133
## Avg Mercury   0.3941837 0.0034943347
## pH            0.3790720 0.0051232287
## Min           0.3423833 0.0120900196
## Chlorophyll   0.3050727 0.0263340764
##
##
## $Dim.3
## $Dim.3$quanti
##             correlation      p.value
## Chlorophyll   0.6602499 7.503544e-08
```

```
## Calcium       -0.2834086 3.974203e-02
```

El resultado en cuanto a las dimensiones creo que no varía mucho del PCA. La dimension 1 claramente diferencia los dos grupos. La dimension 2 por otra parte, estaría definida por los lagos en la que correlacion de los dos grupos de variables no es inversa.

Lo que si es interesante es cómo con MFA se puede ver como cada grupo "arrastra" a los lagos individuales.