Tlajinga Hierarchical Cluster Analysis

Gina M. Buckley

6/2/2020

Introduction

This document is a step-by-step explanation of the code used in RStudio to develop the hierarchical cluster analysis found in **Buckley et al. (in press)** in the peer-reviewed journal article **New Perspectives on Migration into the Tlajinga District of Teotihuacan: A Dual-Isotope Approach** in *Latin American Antiquity*.

This model analyzes Strontium (Sr) and stable Oxygen-phosphate (Ophos) isotopes generated for 13 individual burials from the Tlajinga neighborhood of Teotihuacan, Mexico using hierarchical cluster analysis. This is an alternative approach to k-means cluster analysis and does not require a predetermination of the number of clusters to be generated.

All data from Buckley et al. (in press) needed to successfully conduct this analysis are included within this code. Complete data for this study can be found here (https://doi.org/10.26207/483d-ve56) including the R Markdown file.

Many aspects of this analysis were developed by following this excellent tutorial: UC Business Analystics R Programming Guide (https://uc-r.github.io/hc_clustering).

Install Libraries

For this model, you will need to have the following libraries installed in RStudio. Use the **install.packages** command and then run the following libraries:

```
library(ggplot2)
library(factoextra)
library(FactoMineR)
library(cluster)
library(tidyverse)
library(dendextend)
```

Create a Dataframe

The following data are the Sr-isotope and O-isotope data generated for this study.

The data listed under “burial” are character variables. These data express the Burial ID of each sample. You will not need this information to run the analysis, but it helps keep track of each data point within the model. R will automatically generate numbers 1-13 in order of the sample name input below. These generated numbers are what will appear in tables and plots.

```
burial <- c('18-3b-2', '33-14-2', '33-15-1', '33-15-2', '33-43-2',
            '33-50b-1', '33-50d-1', '33-50d-3', '33-56-1', '33-56-3',
            '33-57-1', '33-57-2', '33-61-1')
```

These data points for Sr and O-isotopes are listed in order corresponding to the Burial ID above. **It is imperative that you list these values in order so that they correspond with the correct burial.**

```
Sr <- c(0.70487, 0.70513, 0.70550, 0.70498, 0.70570,
        0.70512, 0.70510, 0.70509, 0.70523, 0.70500,
        0.70498, 0.70497, 0.70499)

Ophos <- c(15.9, 18.1, 15.0, 14.2, 16.4,
           14.7, 16.3, 14.3, 15.4, 15.3,
           14.9, 15.4, 15.1)

df <- data.frame(Sr, Ophos)
df <- na.omit(df)
```

View your data to make sure it is in the correct order.

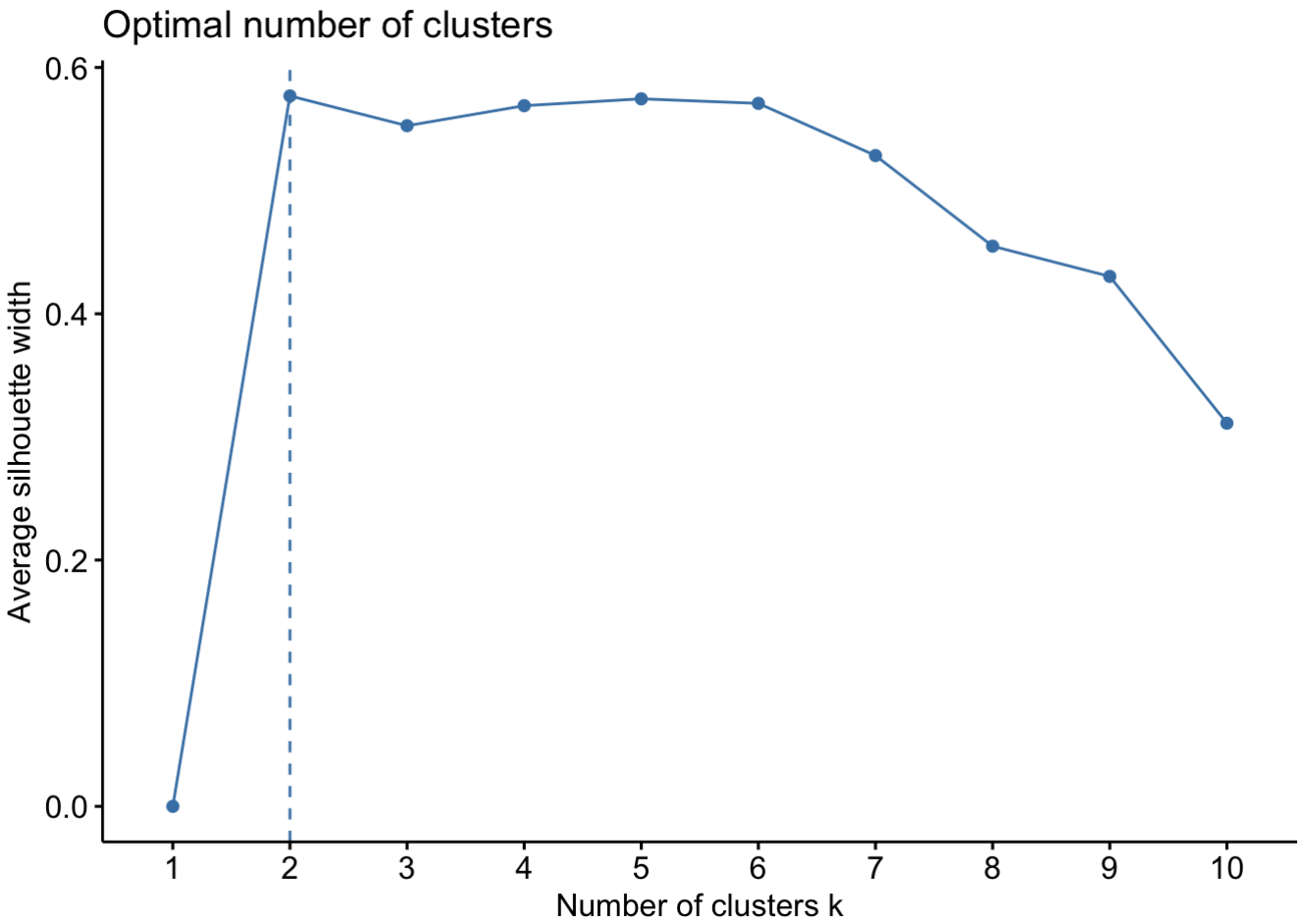
```
df[1:13, ]
```

```
##         Sr Ophos
## 1  0.70487  15.9
## 2  0.70513  18.1
## 3  0.70550  15.0
## 4  0.70498  14.2
## 5  0.70570  16.4
## 6  0.70512  14.7
## 7  0.70510  16.3
## 8  0.70509  14.3
## 9  0.70523  15.4
## 10 0.70500  15.3
## 11 0.70498  14.9
## 12 0.70497  15.4
## 13 0.70499  15.1
```

Optimal Clusters

Before running statistical analysis to determine which clustering groups of individuals are significantly different to the other, it is a good idea to test the data for the optimal number of clusters. Here I use the **average silhouette method**:

```
fviz_nbclust(df, FUN = hcut, method = "silhouette")
```



Here we see that 2 clusters are the optimal number for this dataset.

Clustering Methods

Now we need to decide what type of hierarchical clustering we want to use for this analysis. There are two main types:

- Agglomerative clustering (AGNES): a bottom up method
- Divisive hierarchical clustering (DIANA): a top-down method

For this analysis will will use AGNES. – (see cited tuturial above for more information)

Now we test which clustering method has the strongest clustering structures for these data.The clustering methods below are similar to the k-means method and work best for datasets with small sample sizes, like this one.

Below are a list of clustering methods. Each method will produce a dendrogram that has been created from different approaches. Information about how each method works can be found here (https://uc-r.github.io/hc_clustering).

- Mean or average linkage clustering
- Single linkage clustering
- Complete linkage clustering
- Ward's minimum variance method

To determine which method we should use, first standardize the data:

```
df <- scale(df)
head (df)
```

```
##             Sr      Ophos
## 1 -1.108050195  0.4220601
## 2  0.009922838  2.5397654
## 3  1.600884461 -0.4442738
## 4 -0.635061604 -1.2143485
## 5  2.460863717  0.9033568
## 6 -0.033076125 -0.7330518
```

Now test which clustering method has the strongest clustering structures.

```
m <- c( "average", "single", "complete", "ward")
names(m) <- c( "average", "single", "complete", "ward")
ac <- function(x) {
  agnes(df, method = x)$ac}
map_dbl(m, ac)
```

```
##   average    single  complete      ward
## 0.7071168 0.5984334 0.7574074 0.8019821
```

For this dataset, Ward's method is best with a coefficient of 0.80 (closest to 1.0 is best).

Compute Clusters

Now it is time to compute the clusters for the data.

First, compute the dissimilarity values:

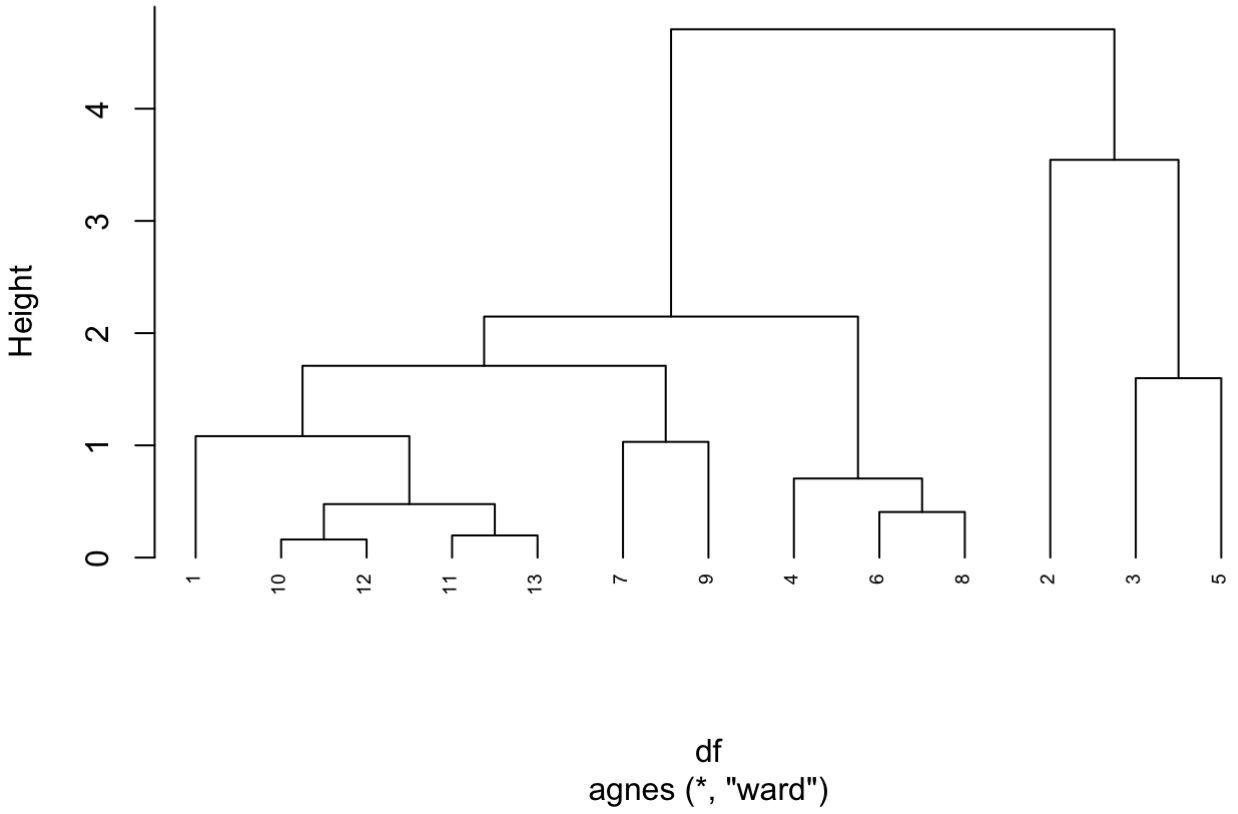
```
d <- dist(df, method="euclidean")
```

Then feed these values into the hierarchical cluster specifying which method you have choosen. In this case, AGNES and Ward's method:

```
hc1 <- agnes(df, method="ward")
```

Visualize the Ward Cluster

```
hc1 <- agnes(df, method = "ward")
pltree(hc1, cex = 0.6, hang = -1, main = "Dendrogram of AGNES")
```



Cut the tree with Ward's method with the knowledge that the optimal cluster number is 2.

```
hc2 <- hclust(d, method = "ward.D2" )
```

Although the silhouette method indicated 2 clusters as optimal, we still want to test clusters of 3, 4, and 5. Set up these clusters:

```
sub_grp2 <- cutree(hc2, k = 2)
table(sub_grp2)
```

```
## sub_grp2
##  1  2
## 10  3
```

```
sub_grp3 <- cutree(hc2, k = 3)
table(sub_grp3)
```

```
## sub_grp3
##  1  2  3
## 10  1  2
```

```
sub_grp4 <- cutree(hc2, k = 4)
table(sub_grp4)
```

```
## sub_grp4
## 1 2 3 4
## 7 1 2 3
```

```
sub_grp5 <- cutree(hc2, k = 5)
table(sub_grp5)
```

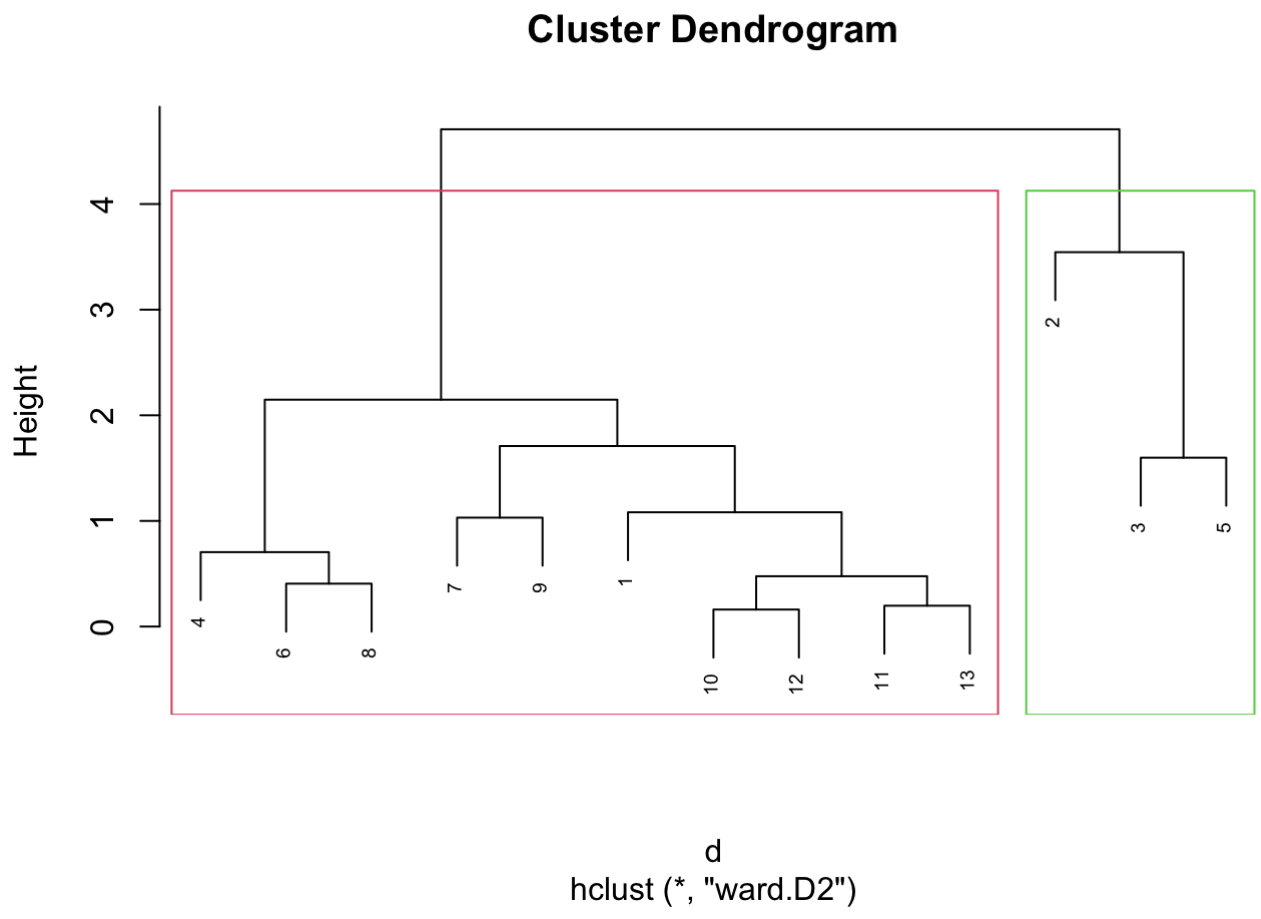
```
## sub_grp5
## 1 2 3 4 5
## 5 1 2 3 2
```

Visualize the Cluster Groups

2 Clusters

Plot the dendrogram with borders around clusters for the best visual.

```
plot(hc2, cex = 0.6)
rect.hclust(hc2, k = 2, border = 2:5)
```



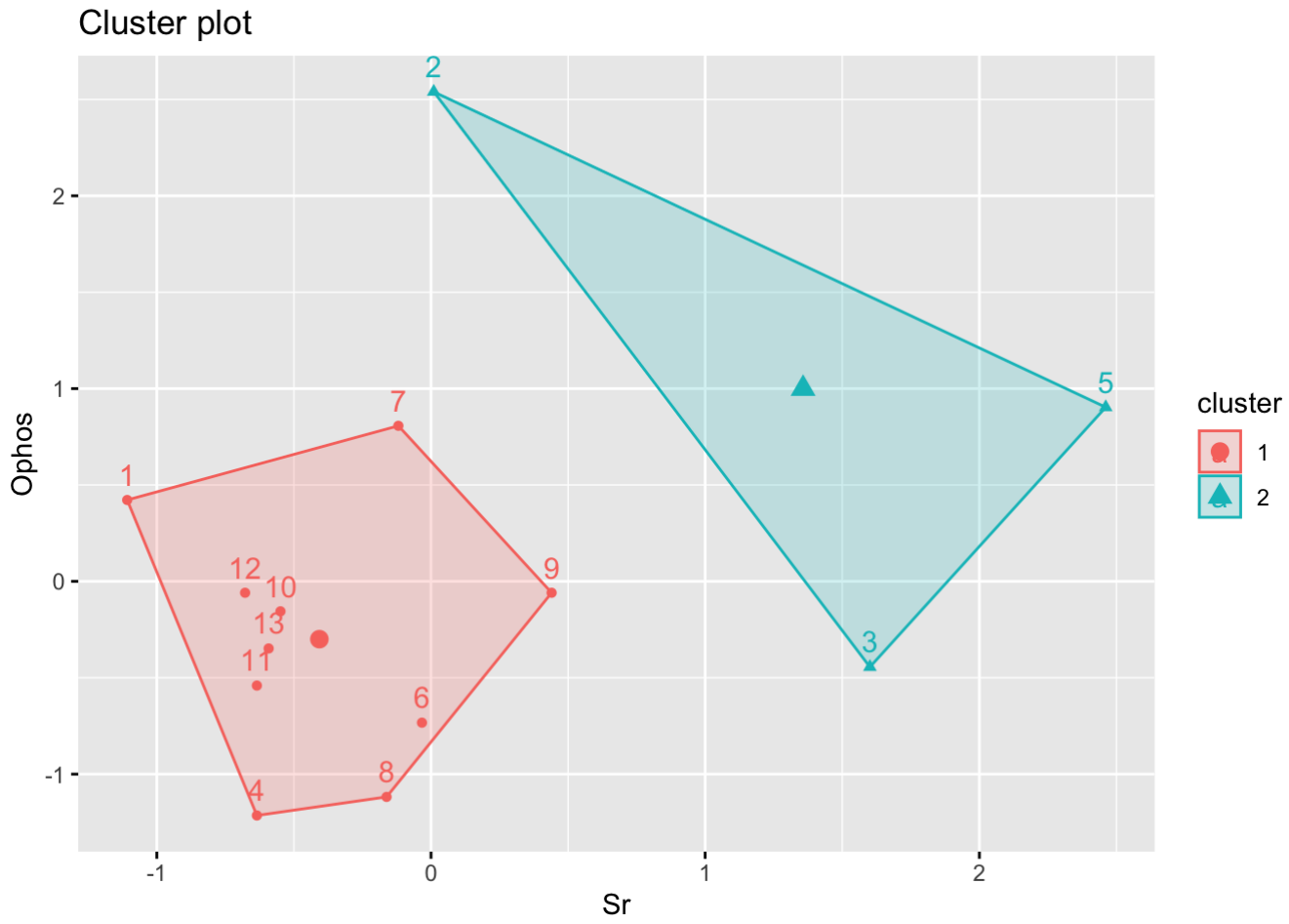
You can also produce a table displaying which sample belongs in each cluster.

```
cutree(as.hclust(hc2), k = 2)
```

```
##  1  2  3  4  5  6  7  8  9 10 11 12 13
##  1  2  2  1  2  1  1  1  1  1  1  1  1
```

Produce a cluster plot.

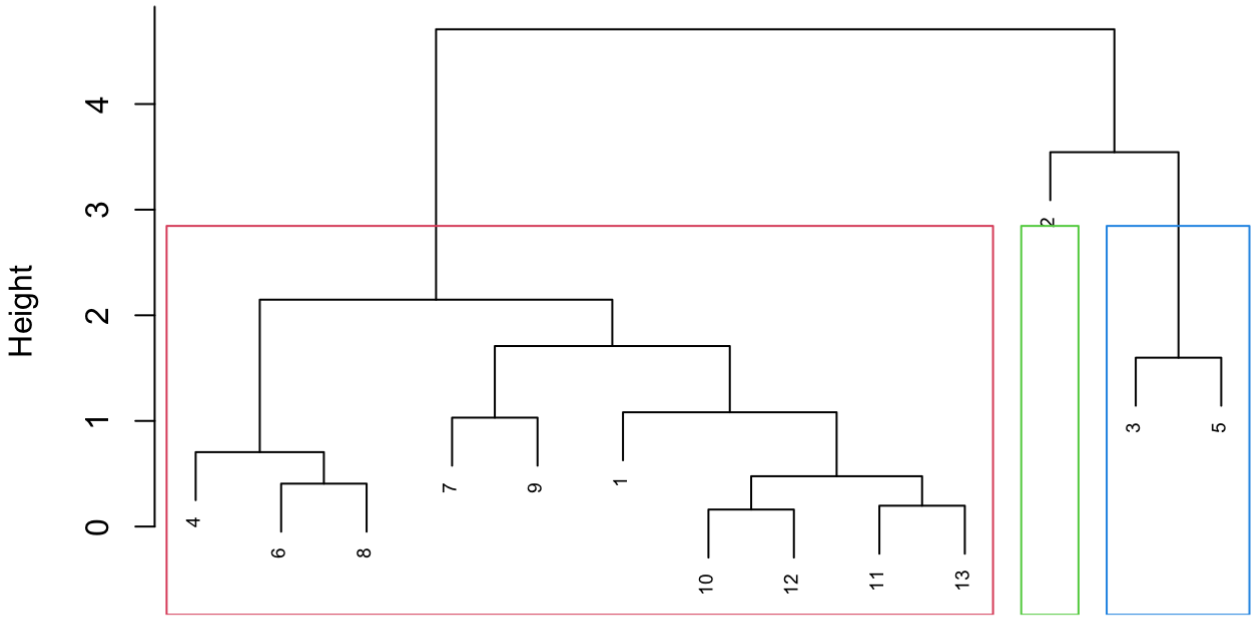
```
fviz_cluster(list(data = df, cluster = sub_grp2))
```



3 Clusters

Repeat the steps above for 3 clusters:

```
hc3 <- hclust(d, method = "ward.D2" )
plot(hc3, cex = 0.6, main = "")
rect.hclust(hc3, k = 3, border = 2:5)
```

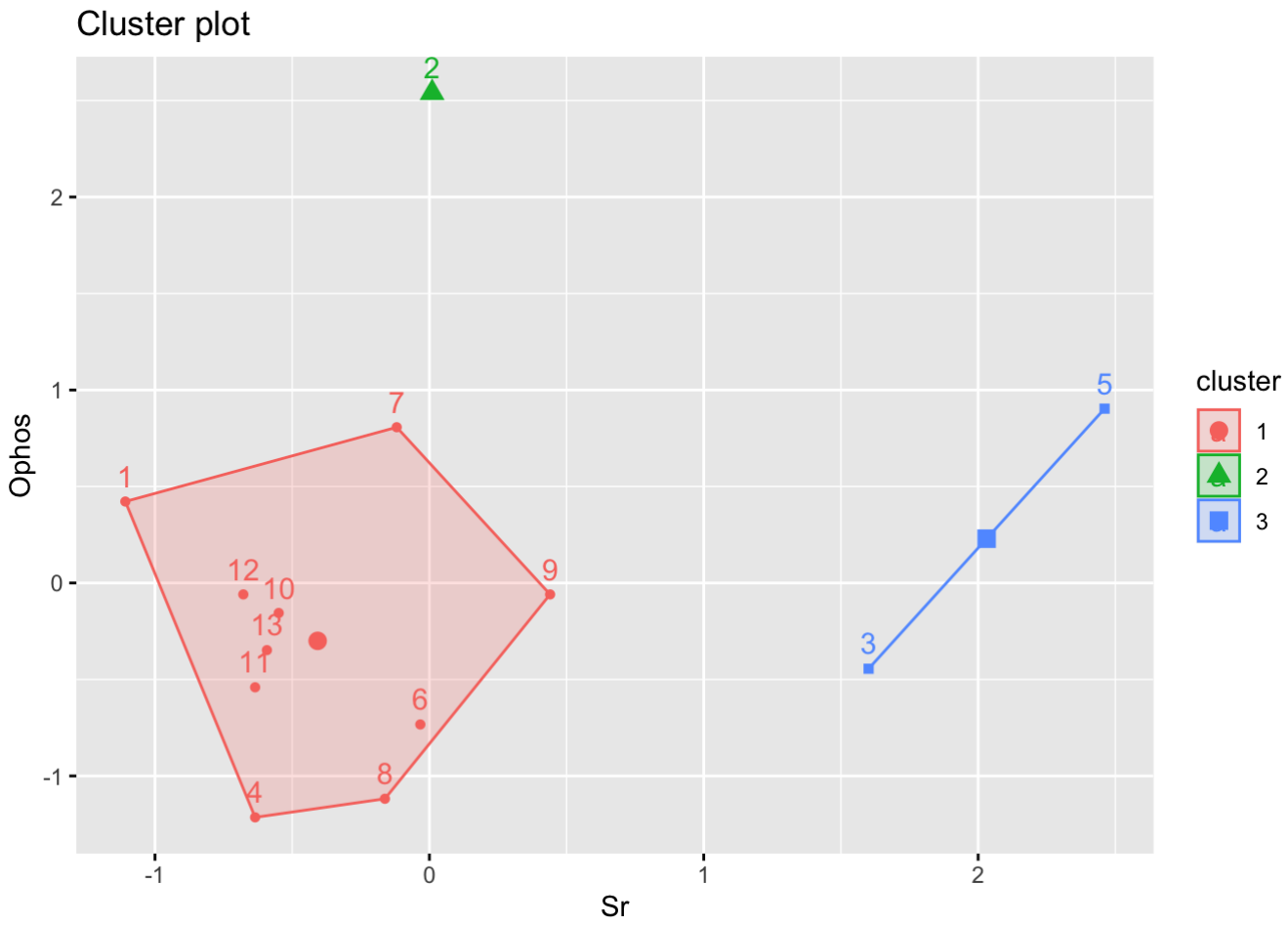


d
hclust (*, "ward.D2")

```
cutree(as.hclust(hc3), k = 3)
```

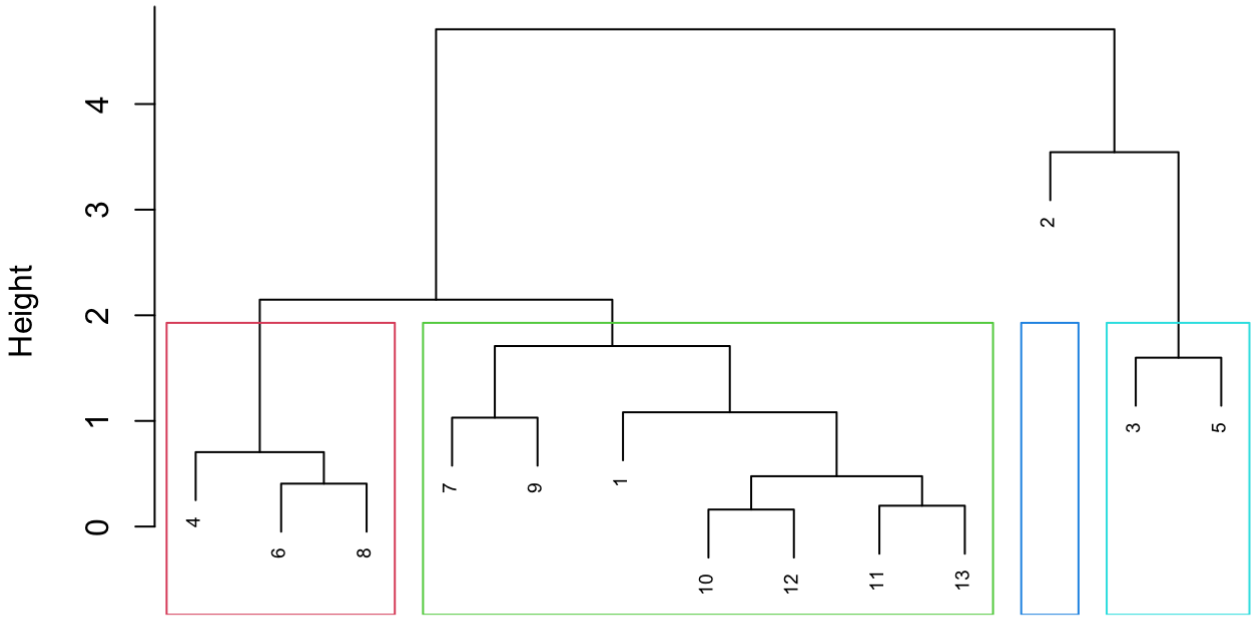
```
##  1  2  3  4  5  6  7  8  9 10 11 12 13
##  1  2  3  1  3  1  1  1  1  1  1  1  1
```

```
fviz_cluster(list(data = df, cluster = sub_grp3))
```



4 clusters

```
hc4 <- hclust(d, method = "ward.D2" )
plot(hc4, cex = 0.6, main = "")
rect.hclust(hc4, k = 4, border = 2:5)
```

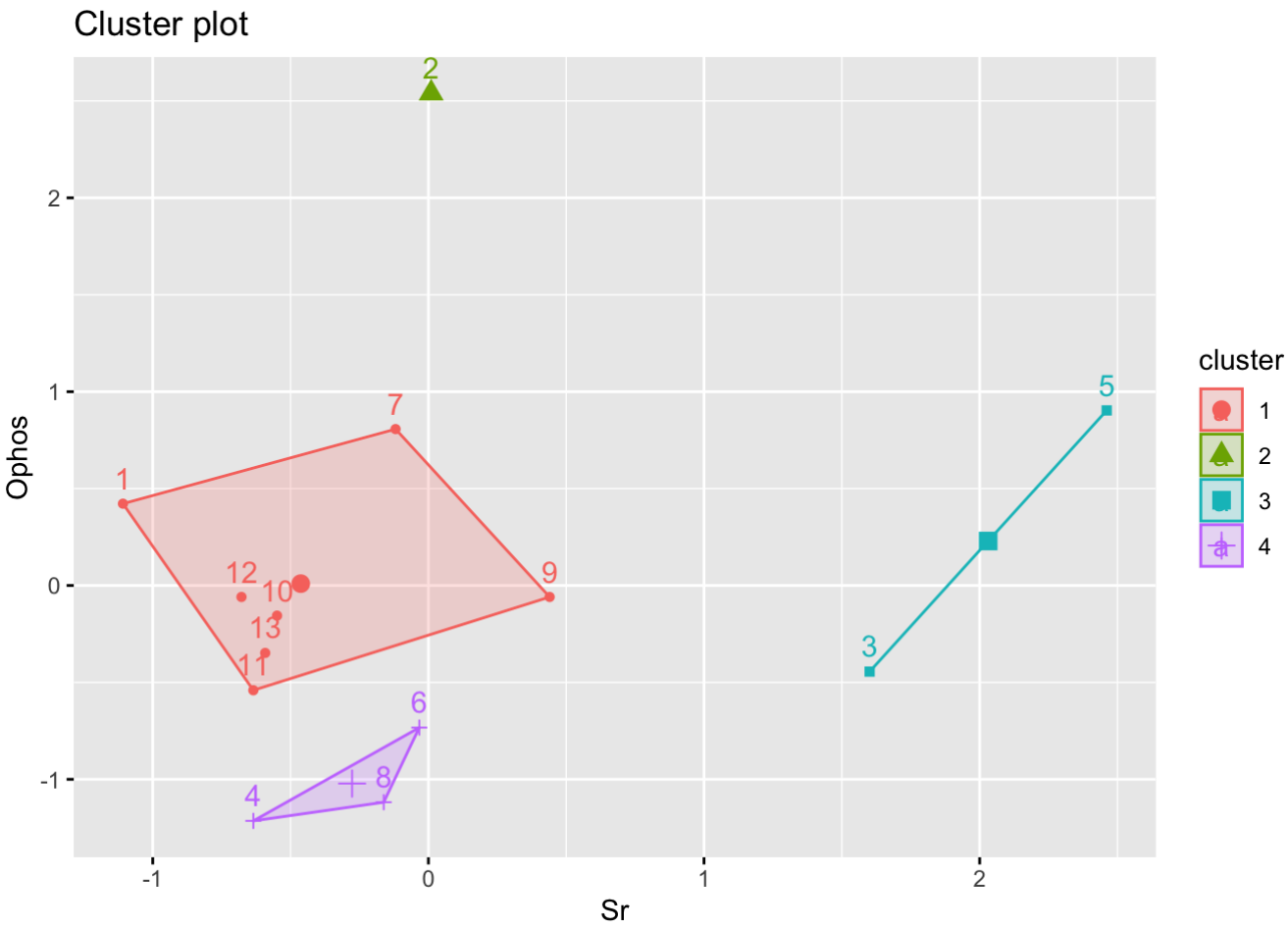


d
hclust (*, "ward.D2")

```
cutree(as.hclust(hc3), k = 4)
```

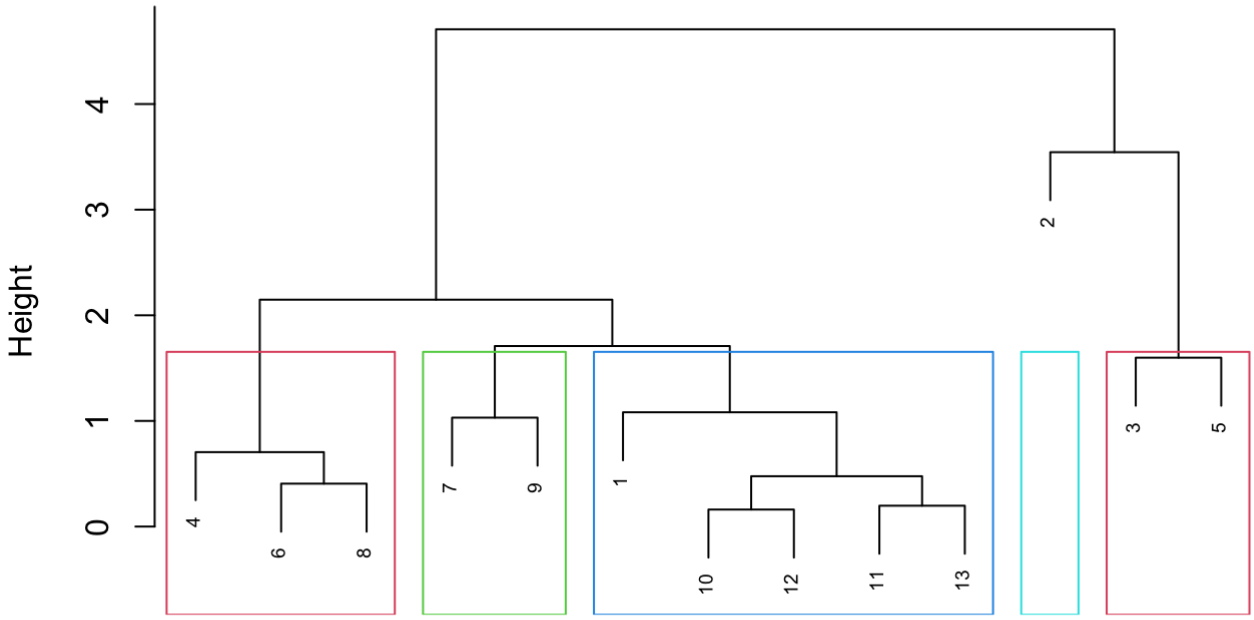
```
##  1  2  3  4  5  6  7  8  9 10 11 12 13
##  1  2  3  4  3  4  1  4  1  1  1  1  1
```

```
fviz_cluster(list(data = df, cluster = sub_grp4))
```



5 clusters

```
hc5 <- hclust(d, method = "ward.D2" )
plot(hc5, cex = 0.6, main = "")
rect.hclust(hc5, k = 5, border = 2:5)
```

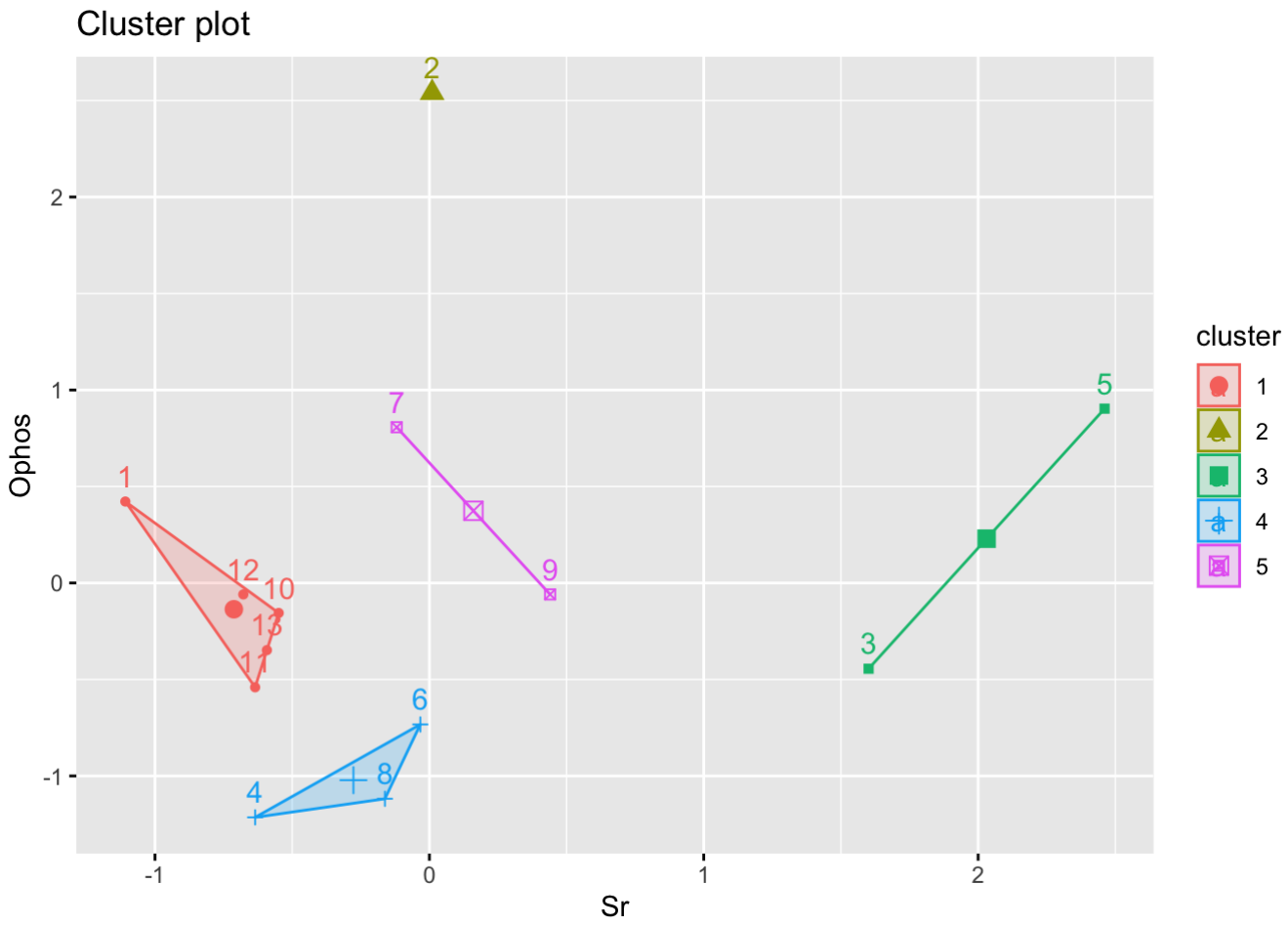


d
hclust (*, "ward.D2")

```
cutree(as.hclust(hc3), k = 5)
```

```
##  1  2  3  4  5  6  7  8  9 10 11 12 13
##  1  2  3  4  3  4  5  4  5  1  1  1  1
```

```
fviz_cluster(list(data = df, cluster = sub_grp5))
```



Significance Tests between Clusters

Install and run libraries

```
library("dplyr")
library("ggpubr")
```

Based on the plots produced above, enter the new cluster data into the dataframe. This is easiest by first organizing data into an excel spreadsheet to see which sample goes into which cluster based on the division of clusters. “cls” = Cluster.

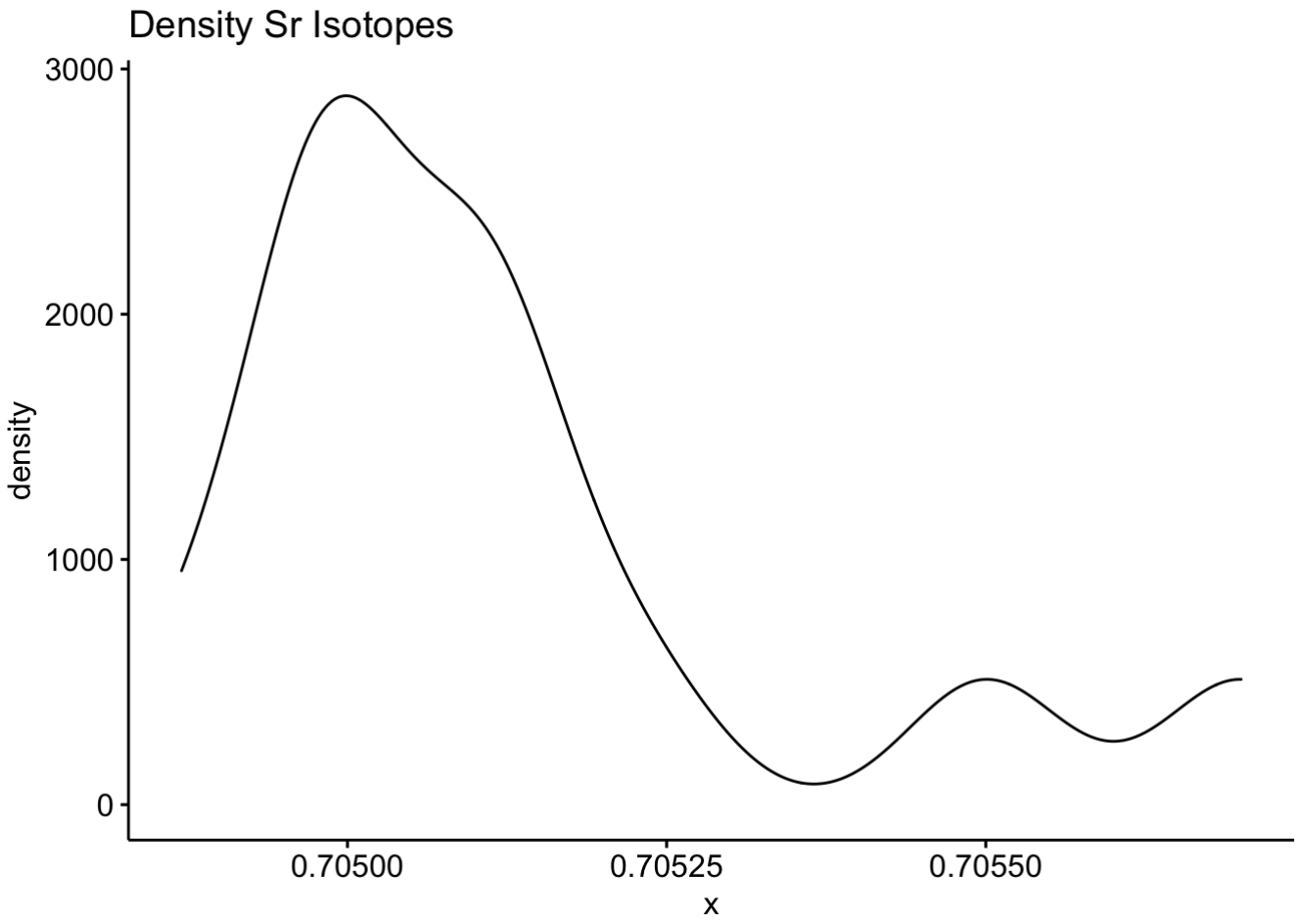
```
group2 <- c('cls1', 'cls2', 'cls2', 'cls1', 'cls2',
            'cls1', 'cls1', 'cls1', 'cls1', 'cls1',
            'cls1', 'cls1', 'cls1')
group3 <- c('cls1', 'cls2', 'cls3', 'cls1', 'cls3',
            'cls1', 'cls1', 'cls1', 'cls1', 'cls1',
            'cls1', 'cls1', 'cls1')
group4 <- c('cls1', 'cls2', 'cls3', 'cls4', 'cls3',
            'cls4', 'cls1','cls4', "cls1", "cls1",
            'cls1', 'cls1', 'cls1')
group5 <- c('cls1', 'cls2', 'cls3', 'cls4', 'cls3',
            'cls4', 'cls5', 'cls4', "cls5", "cls1",
            'cls1', 'cls1', 'cls1')

df<- data.frame (Sr, Ophos, group2, group3, group4, group5)
```

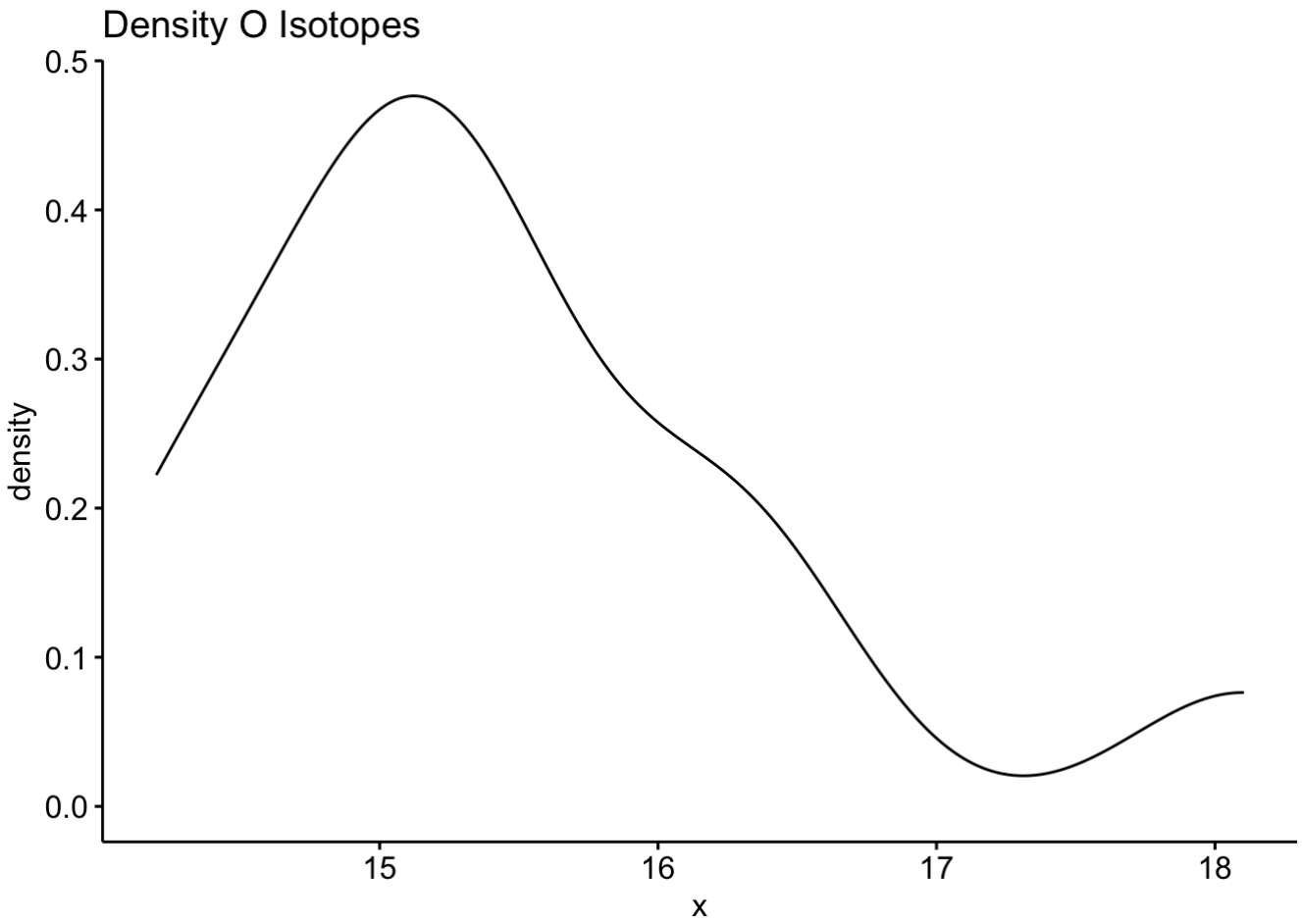
Visualize the data and determine normality

Density plots

```
ggdensity(df$Sr,
          main = "Density Sr Isotopes")
```

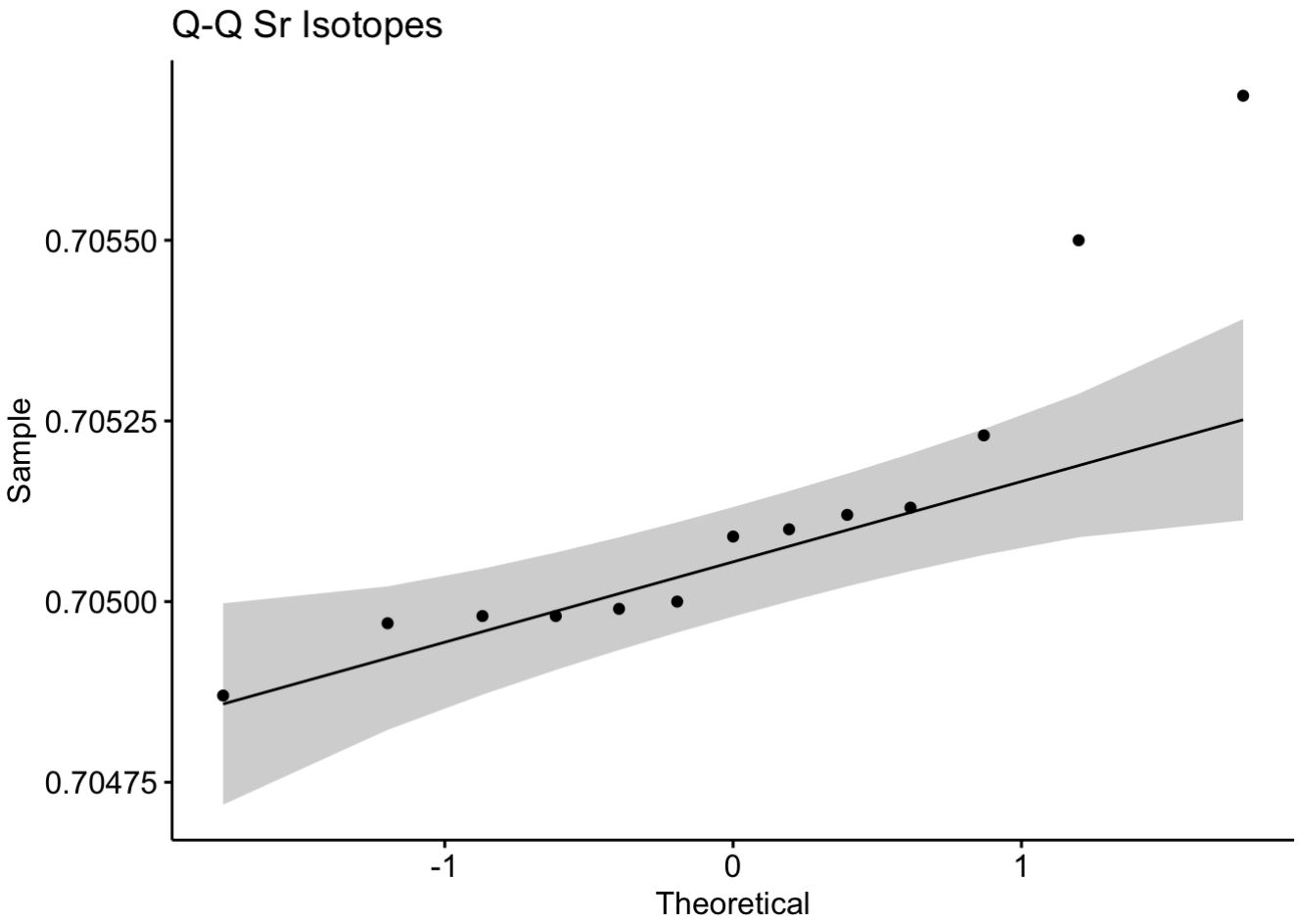



```
ggdensity(df$Ophos,
          main = "Density O Isotopes")
```

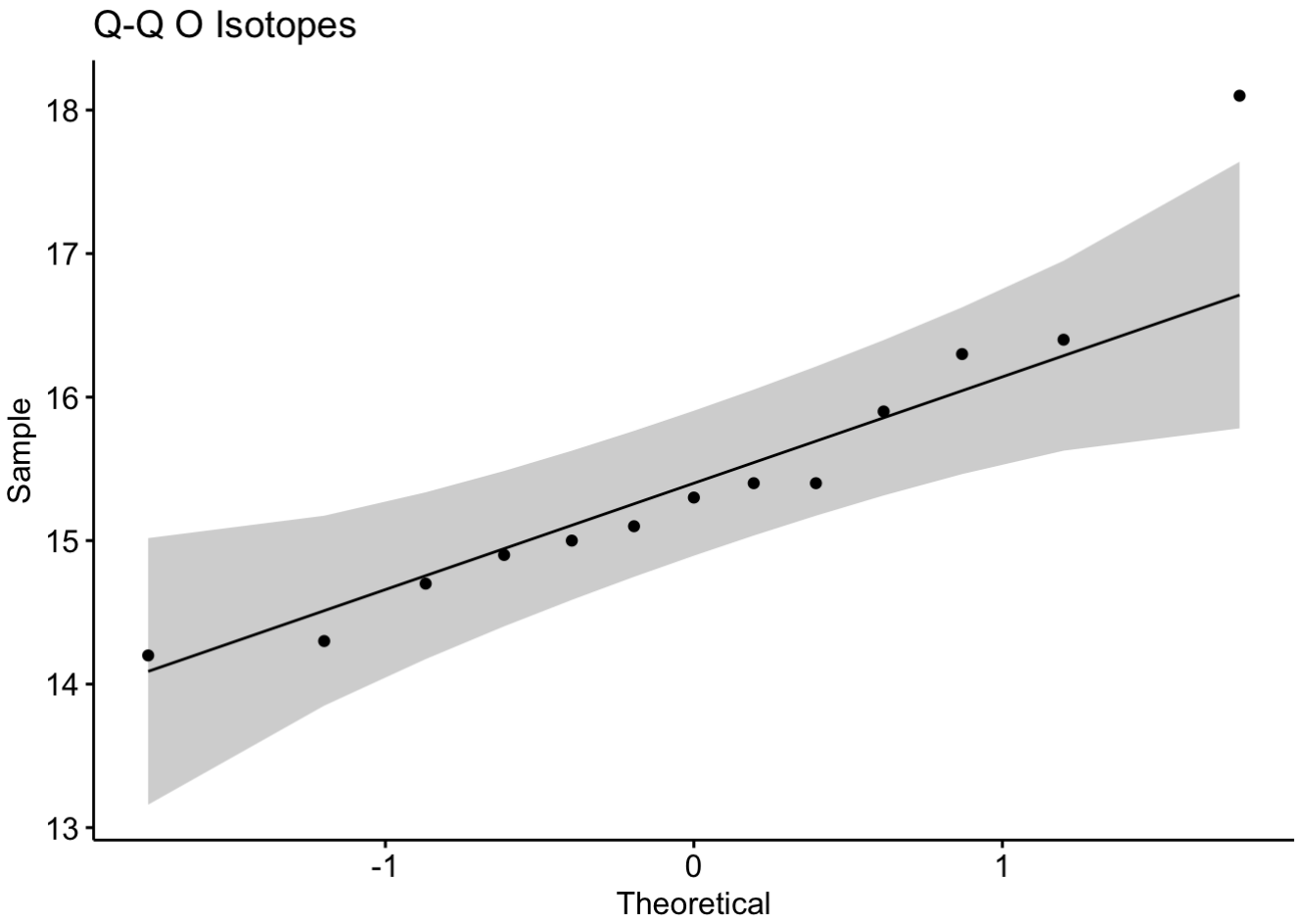


Q-Q plots

```
ggqqplot(df$Sr,
         main = "Q-Q Sr Isotopes")
```




```
ggqqplot(df$Ophos,
         main = "Q-Q O Isotopes")
```



Shapiro-Wilk's normality test

Sr-isotopes:

```
shapiro.test(df$Sr)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$Sr
## W = 0.81341, p-value = 0.009881
```

O-isotopes:

```
shapiro.test(df$Ophos)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$Ophos
## W = 0.89509, p-value = 0.1145
```

One-way ANOVA

Although we see that the Sr-isotope data is not normally distributed, a comparison between different types of significance tests does not result in significantly different results. Therefore, we will use a One-way ANOVA to complete our statistical tests between clusters.

Setup your data so you can run a One-way ANOVA to test significance between cluster groups. Here we set up the data to test between 2 cluster groups:

```
set.seed(1000)
dplyr::sample_n(df, 13)
```

```
##         Sr Ophos group2 group3 group4 group5
## 1  0.70498  14.2   cls1   cls1   cls4   cls4
## 2  0.70498  14.9   cls1   cls1   cls1   cls1
## 3  0.70512  14.7   cls1   cls1   cls4   cls4
## 4  0.70550  15.0   cls2   cls3   cls3   cls3
## 5  0.70509  14.3   cls1   cls1   cls4   cls4
## 6  0.70570  16.4   cls2   cls3   cls3   cls3
## 7  0.70500  15.3   cls1   cls1   cls1   cls1
## 8  0.70523  15.4   cls1   cls1   cls1   cls5
## 9  0.70513  18.1   cls2   cls2   cls2   cls2
## 10 0.70497  15.4   cls1   cls1   cls1   cls1
## 11 0.70499  15.1   cls1   cls1   cls1   cls1
## 12 0.70487  15.9   cls1   cls1   cls1   cls1
## 13 0.70510  16.3   cls1   cls1   cls1   cls5
```

Order groups by cluster number:

```
df$group2 <- ordered(df$group2,
                     levels = c("cls1", "cls2"))
```

Summary statistics

Sr-isotopes:

```
group_by(df, group2) %>%
  summarise(
    count = n(),
    mean = mean(Sr, na.rm = TRUE),
    sd = sd(Sr, na.rm = TRUE)
  )
```

```
## # A tibble: 2 x 4
##   group2 count  mean       sd
##   <ord>  <int> <dbl>    <dbl>
## 1 cls1      10 0.705 0.000102
## 2 cls2       3 0.705 0.000289
```

O-isotopes:

```
group_by(df, group2) %>%
  summarise(
    count = n(),
    mean = mean(Ophos, na.rm = TRUE),
    sd = sd(Ophos, na.rm = TRUE)
  )
```

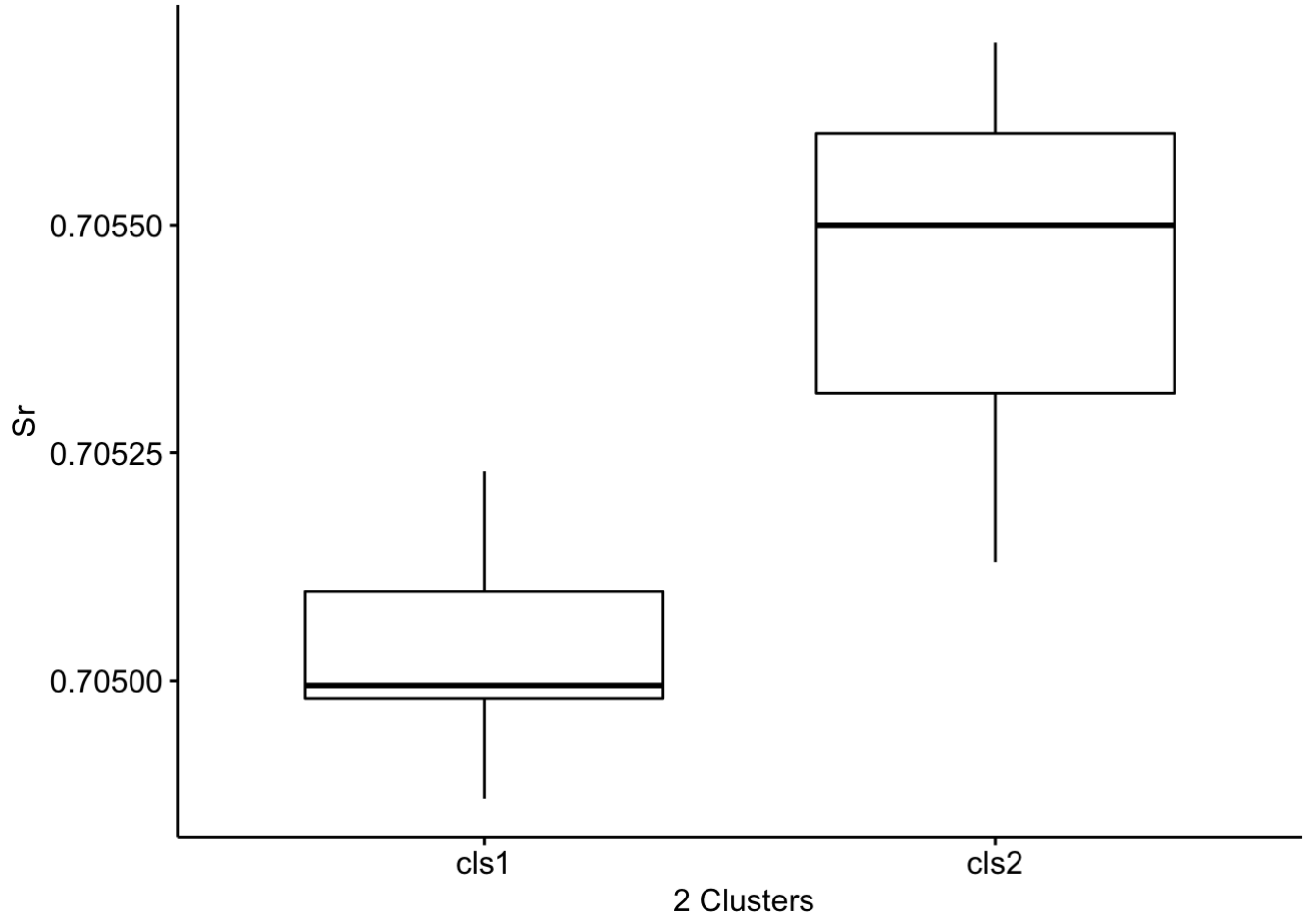
```
## # A tibble: 2 x 4
##   group2 count  mean    sd
##   <ord>  <int> <dbl> <dbl>
## 1 cls1      10  15.2 0.660
## 2 cls2       3  16.5 1.55
```

Visualize the results

Boxplots

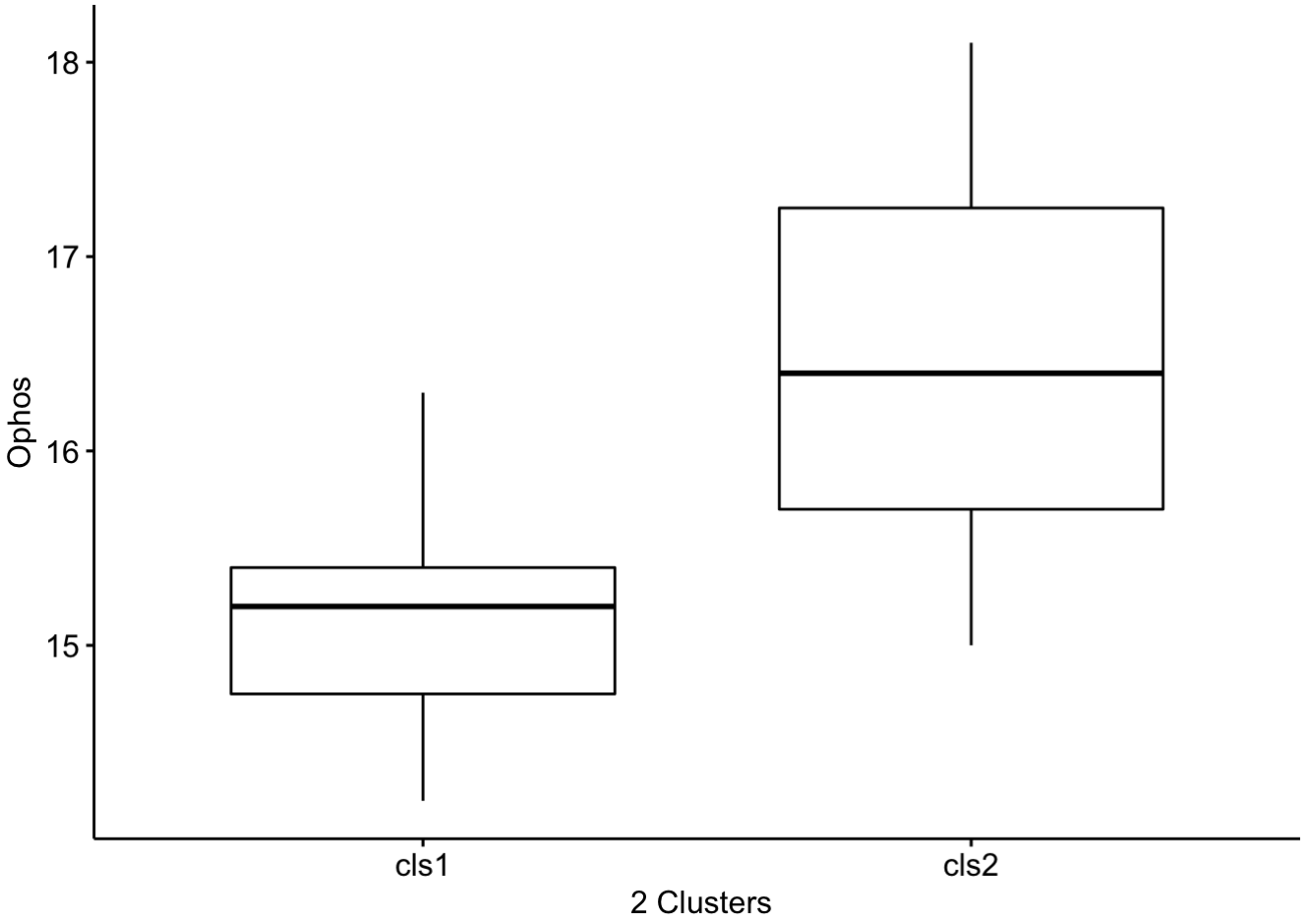
Sr-isotopes:

```
ggboxplot(df, x = "group2", y = "Sr",
          order = c("cls1", "cls2"),
          ylab = "Sr", xlab = " 2 Clusters")
```



O-Isotopes:

```
ggboxplot(df, x = "group2", y = "Ophos",
          order = c("cls1", "cls2"),
          ylab = "Ophos", xlab = " 2 Clusters")
```



Compute One-way ANOVA

Sr isotopes:

```
res.aov <- aov(Sr ~ group2, data = df)
summary(res.aov)
```

```
##             Df    Sum Sq   Mean Sq F value  Pr(>F)
## group2       1 3.886e-07 3.886e-07   16.41 0.00191 **
## Residuals   11 2.605e-07 2.370e-08
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(res.aov)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Sr ~ group2, data = df)
##
## $group2
##                    diff          lwr          upr    p adj
## cls2-cls1 0.0004103333 0.0001873788 0.0006332878 0.001913
```

O-isotopes:

```
res.aov <- aov(Ophos ~ group2, data = df)
summary(res.aov)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## group2       1  4.206   4.206    5.29  0.042 *
## Residuals   11  8.745   0.795
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(res.aov)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Ophos ~ group2, data = df)
##
## $group2
##           diff        lwr      upr     p adj
## cls2-cls1 1.35 0.05815108 2.641849 0.0420275
```

Repeat for each cluster group analysis

3 clusters

```
set.seed(1000)
dplyr::sample_n(df, 13)
```

```
##         Sr Ophos group2 group3 group4 group5
## 1  0.70498  14.2   cls1   cls1   cls4   cls4
## 2  0.70498  14.9   cls1   cls1   cls1   cls1
## 3  0.70512  14.7   cls1   cls1   cls4   cls4
## 4  0.70550  15.0   cls2   cls3   cls3   cls3
## 5  0.70509  14.3   cls1   cls1   cls4   cls4
## 6  0.70570  16.4   cls2   cls3   cls3   cls3
## 7  0.70500  15.3   cls1   cls1   cls1   cls1
## 8  0.70523  15.4   cls1   cls1   cls1   cls5
## 9  0.70513  18.1   cls2   cls2   cls2   cls2
## 10 0.70497  15.4   cls1   cls1   cls1   cls1
## 11 0.70499  15.1   cls1   cls1   cls1   cls1
## 12 0.70487  15.9   cls1   cls1   cls1   cls1
## 13 0.70510  16.3   cls1   cls1   cls1   cls5
```

```
df$group3 <- ordered(df$group3,
                     levels = c("cls1", "cls2", "cls3"))
```

Summary statistics

Sr-isotopes:

```
group_by(df, group3) %>%
  summarise(
    count = n(),
    mean = mean(Sr, na.rm = TRUE),
    sd = sd(Sr, na.rm = TRUE)
  )
```

```
## # A tibble: 3 x 4
##   group3 count  mean        sd
##   <ord>  <int> <dbl>     <dbl>
## 1 cls1      10 0.705  0.000102
## 2 cls2       1 0.705 NA
## 3 cls3       2 0.706  0.000141
```

O-isotopes

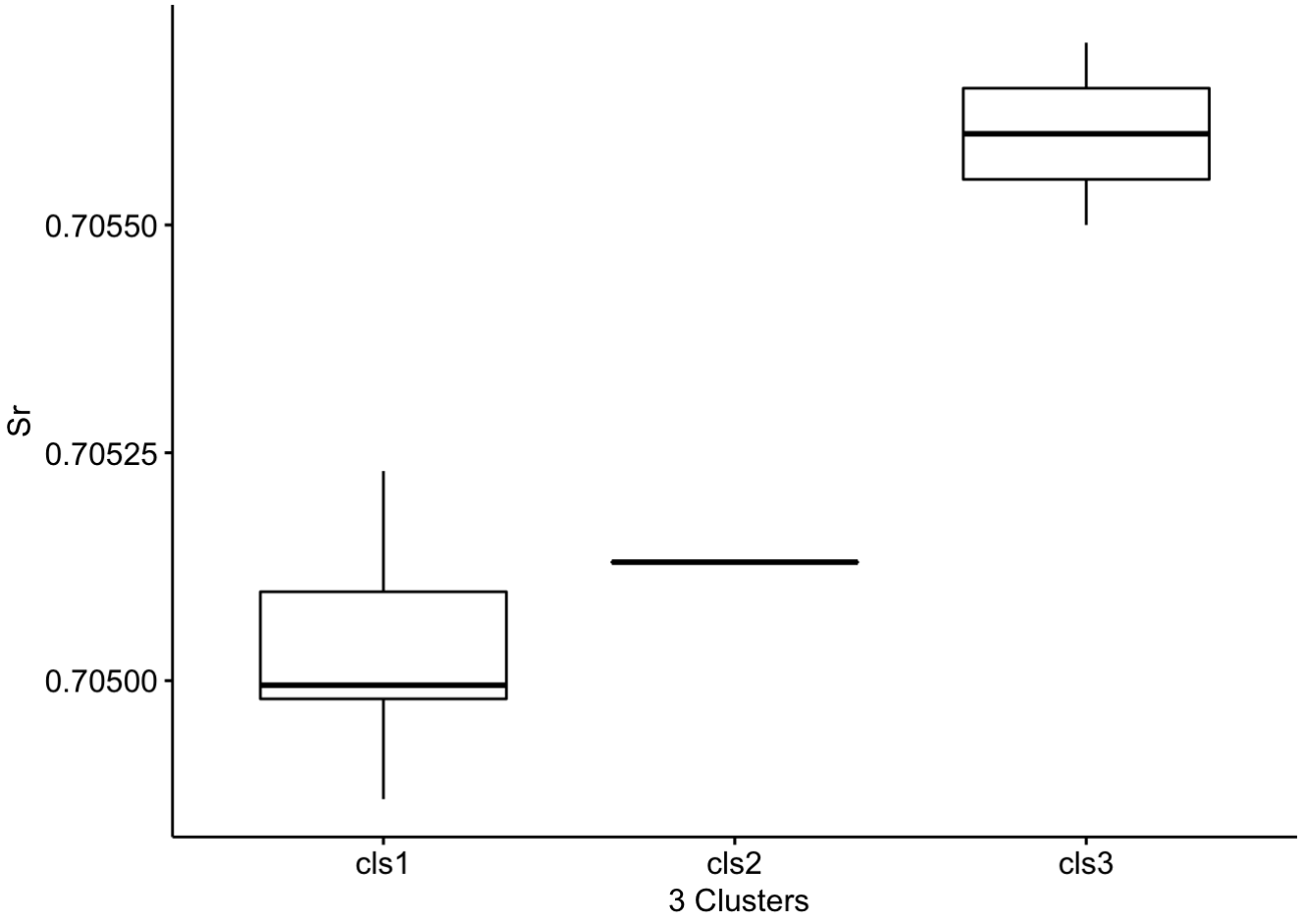
```
group_by(df, group3) %>%
  summarise(
    count = n(),
    mean = mean(Ophos, na.rm = TRUE),
    sd = sd(Ophos, na.rm = TRUE)
  )
```

```
## # A tibble: 3 x 4
##   group3 count  mean     sd
##   <ord>  <int> <dbl>  <dbl>
## 1 cls1      10  15.2  0.660
## 2 cls2       1  18.1 NA
## 3 cls3       2  15.7  0.990
```

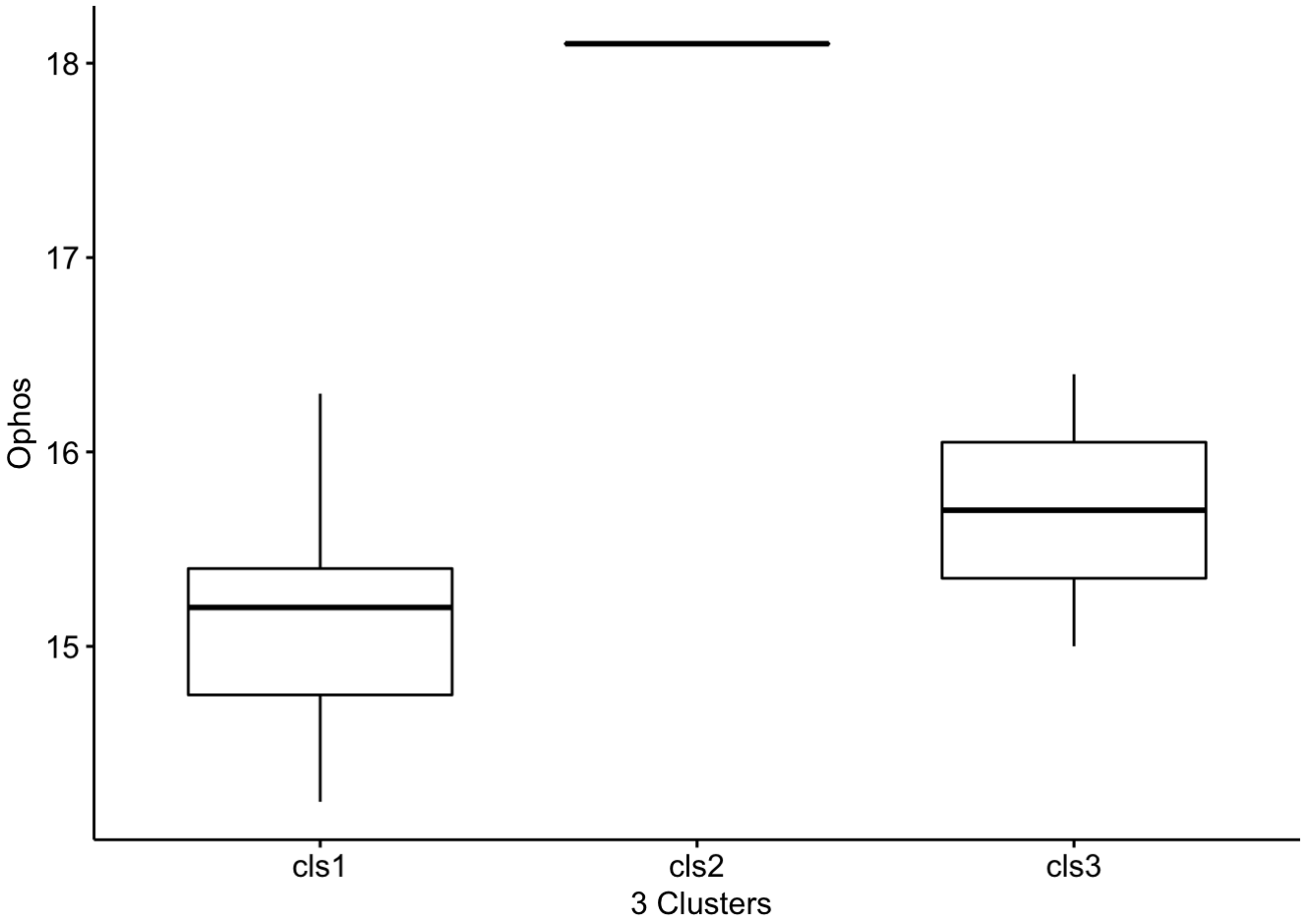
Visualize

Boxplots

```
ggboxplot(df, x = "group3", y = "Sr",
          order = c("cls1", "cls2", "cls3"),
          ylab = "Sr", xlab = " 3 Clusters")
```




```
ggboxplot(df, x = "group3", y = "Ophos",
          order = c("cls1", "cls2", "cls3"),
          ylab = "Ophos", xlab = " 3 Clusters")
```



Compute

Sr-isotopes:

```
res.aov <- aov(Sr ~ group3, data = df)
summary(res.aov)
```

```
##             Df    Sum Sq   Mean Sq F value   Pr(>F)
## group3       2 5.358e-07 2.679e-07   23.66 0.000161 ***
## Residuals   10 1.132e-07 1.132e-08
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(res.aov)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Sr ~ group3, data = df)
##
## $group3
##                diff           lwr          upr     p adj
## cls2-cls1 0.000097 -0.0002089106 0.0004029106 0.6706940
## cls3-cls1 0.000567  0.0003410700 0.0007929300 0.0001155
## cls3-cls2 0.000470  0.0001127734 0.0008272266 0.0121328
```

O-isotopes:

```
res.aov <- aov(Ophos ~ group3, data = df)
summary(res.aov)
```

```
##             Df Sum Sq Mean Sq F value  Pr(>F)
## group3       2  8.046   4.023   8.202 0.00779 **
## Residuals   10  4.905   0.490
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(res.aov)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Ophos ~ group3, data = df)
##
## $group3
##            diff        lwr         upr     p adj
## cls2-cls1  2.95  0.9364072  4.96359277 0.0063033
## cls3-cls1  0.55 -0.9371368  2.03713682 0.5852941
## cls3-cls2 -2.40 -4.7513698 -0.04863023 0.0455673
```

4 Clusters

```
set.seed(1000)
dplyr::sample_n(df, 13)
```

```
##         Sr Ophos group2 group3 group4 group5
## 1  0.70498  14.2   cls1   cls1   cls4   cls4
## 2  0.70498  14.9   cls1   cls1   cls1   cls1
## 3  0.70512  14.7   cls1   cls1   cls4   cls4
## 4  0.70550  15.0   cls2   cls3   cls3   cls3
## 5  0.70509  14.3   cls1   cls1   cls4   cls4
## 6  0.70570  16.4   cls2   cls3   cls3   cls3
## 7  0.70500  15.3   cls1   cls1   cls1   cls1
## 8  0.70523  15.4   cls1   cls1   cls1   cls5
## 9  0.70513  18.1   cls2   cls2   cls2   cls2
## 10 0.70497  15.4   cls1   cls1   cls1   cls1
## 11 0.70499  15.1   cls1   cls1   cls1   cls1
## 12 0.70487  15.9   cls1   cls1   cls1   cls1
## 13 0.70510  16.3   cls1   cls1   cls1   cls5
```

```
df$group1 <- ordered(df$group4,
                     levels = c("cls1", "cls2", "cls3", "cls4"))
```

Summary statistics

Sr-isotopes

```
group_by(df, group4) %>%
  summarise(
    count = n(),
    mean = mean(Sr, na.rm = TRUE),
    sd = sd(Sr, na.rm = TRUE)
  )
```

```
## # A tibble: 4 x 4
##   group4 count  mean         sd
##   <chr>  <int> <dbl>      <dbl>
## 1 cls1       7 0.705  0.000114
## 2 cls2       1 0.705 NA
## 3 cls3       2 0.706  0.000141
## 4 cls4       3 0.705  0.0000737
```

O-isotopes

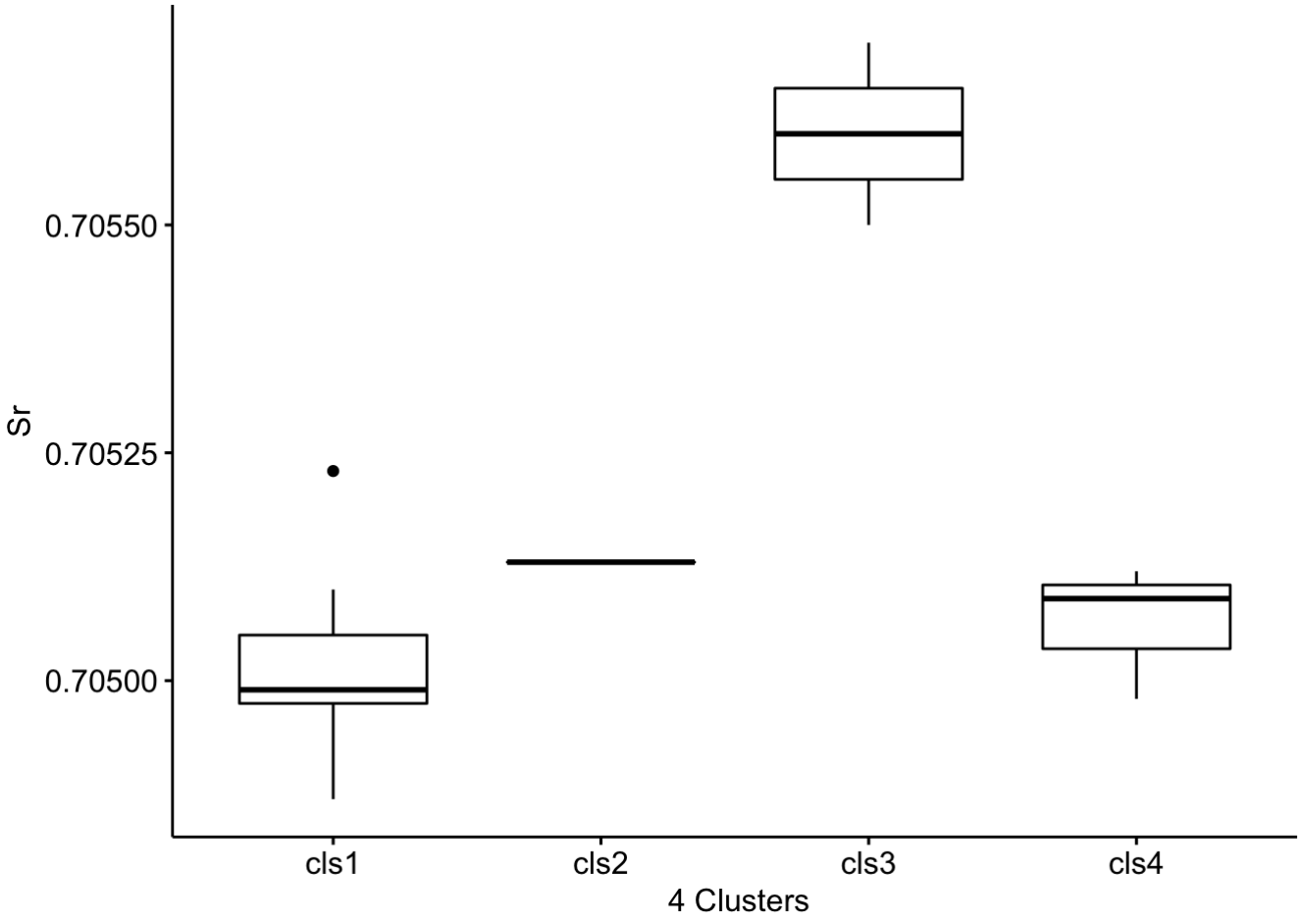
```
group_by(df, group4) %>%
  summarise(
    count = n(),
    mean = mean(Ophos, na.rm = TRUE),
    sd = sd(Ophos, na.rm = TRUE)
  )
```

```
## # A tibble: 4 x 4
##   group4 count  mean     sd
##   <chr>  <int> <dbl>  <dbl>
## 1 cls1       7  15.5  0.479
## 2 cls2       1  18.1 NA
## 3 cls3       2  15.7  0.990
## 4 cls4       3  14.4  0.265
```

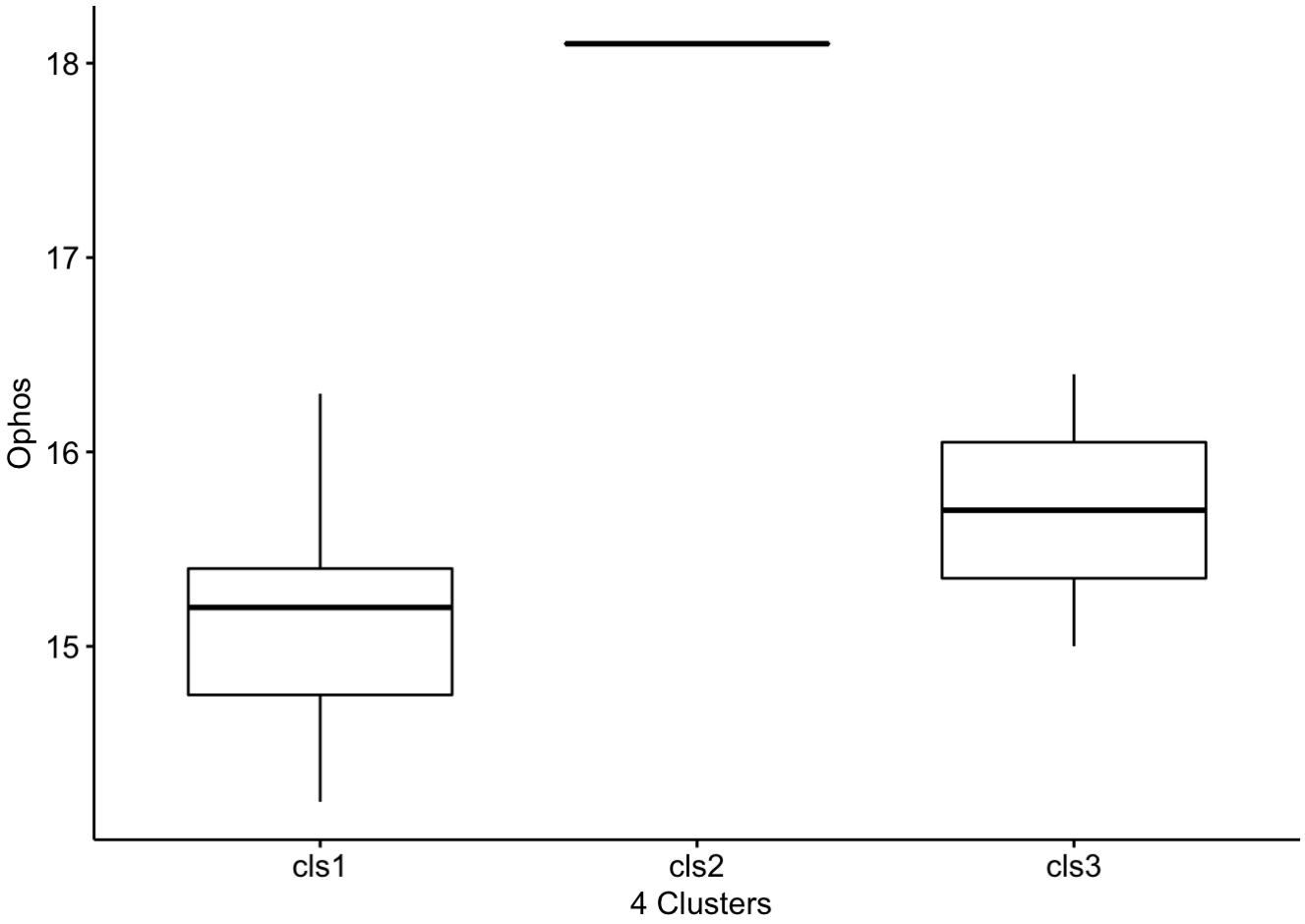
Visualize

Boxplots

```
ggboxplot(df, x = "group4", y = "Sr",
          order = c("cls1", "cls2", "cls3", "cls4"),
          ylab = "Sr", xlab = " 4 Clusters")
```




```
ggboxplot(df, x = "group3", y = "Ophos",
          order = c("cls1", "cls2", "cls3", "cls4"),
          ylab = "Ophos", xlab = " 4 Clusters")
```



Compute

Sr-isotopes:

```
res.aov <- aov(Sr ~ group4, data = df)
summary(res.aov)
```

```
##             Df    Sum Sq   Mean Sq F value   Pr(>F)
## group4       3 5.398e-07 1.799e-07   14.82 0.000792 ***
## Residuals    9 1.093e-07 1.214e-08
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(res.aov)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Sr ~ group4, data = df)
##
## $group4
##                      diff           lwr           upr     p adj
## cls2-cls1  1.100000e-04 -2.577253e-04  0.0004777253 0.7881233
## cls3-cls1  5.800000e-04  3.042061e-04  0.0008557939 0.0004869
## cls4-cls1  4.333333e-05 -1.940323e-04  0.0002806990 0.9385926
## cls3-cls2  4.700000e-04  4.871779e-05  0.0008912822 0.0290857
## cls4-cls2 -6.666667e-05 -4.638553e-04  0.0003305220 0.9511339
## cls4-cls3 -5.366667e-04 -8.506719e-04 -0.0002226614 0.0021672
```

O-isotopes:

```
res.aov <- aov(Ophos ~ group4, data = df)
summary(res.aov)
```

```
##             Df Sum Sq Mean Sq F value  Pr(>F)
## group4       3 10.456   3.485   12.58 0.00143 **
## Residuals    9  2.494   0.277
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(res.aov)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Ophos ~ group4, data = df)
##
## $group4
##                  diff        lwr         upr     p adj
## cls2-cls1  2.6285714  0.8716485  4.38549440 0.0052572
## cls3-cls1  0.2285714 -1.0891208  1.54626365 0.9465353
## cls4-cls1 -1.0714286 -2.2055175  0.06266033 0.0648174
## cls3-cls2 -2.4000000 -4.4128081 -0.38719188 0.0203789
## cls4-cls2 -3.7000000 -5.5976937 -1.80230630 0.0008517
## cls4-cls3 -1.3000000 -2.8002586  0.20025860 0.0934452
```

5 Clusters

```
set.seed(1000)
dplyr::sample_n(df, 13)
```

```
##         Sr Ophos group2 group3 group4 group5 group1
## 1  0.70498  14.2   cls1   cls1   cls4   cls4   cls4
## 2  0.70498  14.9   cls1   cls1   cls1   cls1   cls1
## 3  0.70512  14.7   cls1   cls1   cls4   cls4   cls4
## 4  0.70550  15.0   cls2   cls3   cls3   cls3   cls3
## 5  0.70509  14.3   cls1   cls1   cls4   cls4   cls4
## 6  0.70570  16.4   cls2   cls3   cls3   cls3   cls3
## 7  0.70500  15.3   cls1   cls1   cls1   cls1   cls1
## 8  0.70523  15.4   cls1   cls1   cls1   cls5   cls1
## 9  0.70513  18.1   cls2   cls2   cls2   cls2   cls2
## 10 0.70497  15.4   cls1   cls1   cls1   cls1   cls1
## 11 0.70499  15.1   cls1   cls1   cls1   cls1   cls1
## 12 0.70487  15.9   cls1   cls1   cls1   cls1   cls1
## 13 0.70510  16.3   cls1   cls1   cls1   cls5   cls1
```

```
levels(df$group5)
```

```
## NULL
```

```
df$group5 <- ordered(df$group5,
                     levels = c("cls1", "cls2", "cls3", "cls4", "cls5"))
```

Summary statistics

Sr isotopes:

```
group_by(df, group5) %>%
  summarise(
    count = n(),
    mean = mean(Sr, na.rm = TRUE),
    sd = sd(Sr, na.rm = TRUE)
  )
```

```
## # A tibble: 5 x 4
##   group5 count  mean         sd
##   <ord>  <int> <dbl>      <dbl>
## 1 cls1       5 0.705  0.0000526
## 2 cls2       1 0.705 NA
## 3 cls3       2 0.706  0.000141
## 4 cls4       3 0.705  0.0000737
## 5 cls5       2 0.705  0.0000919
```

O-isotopes:

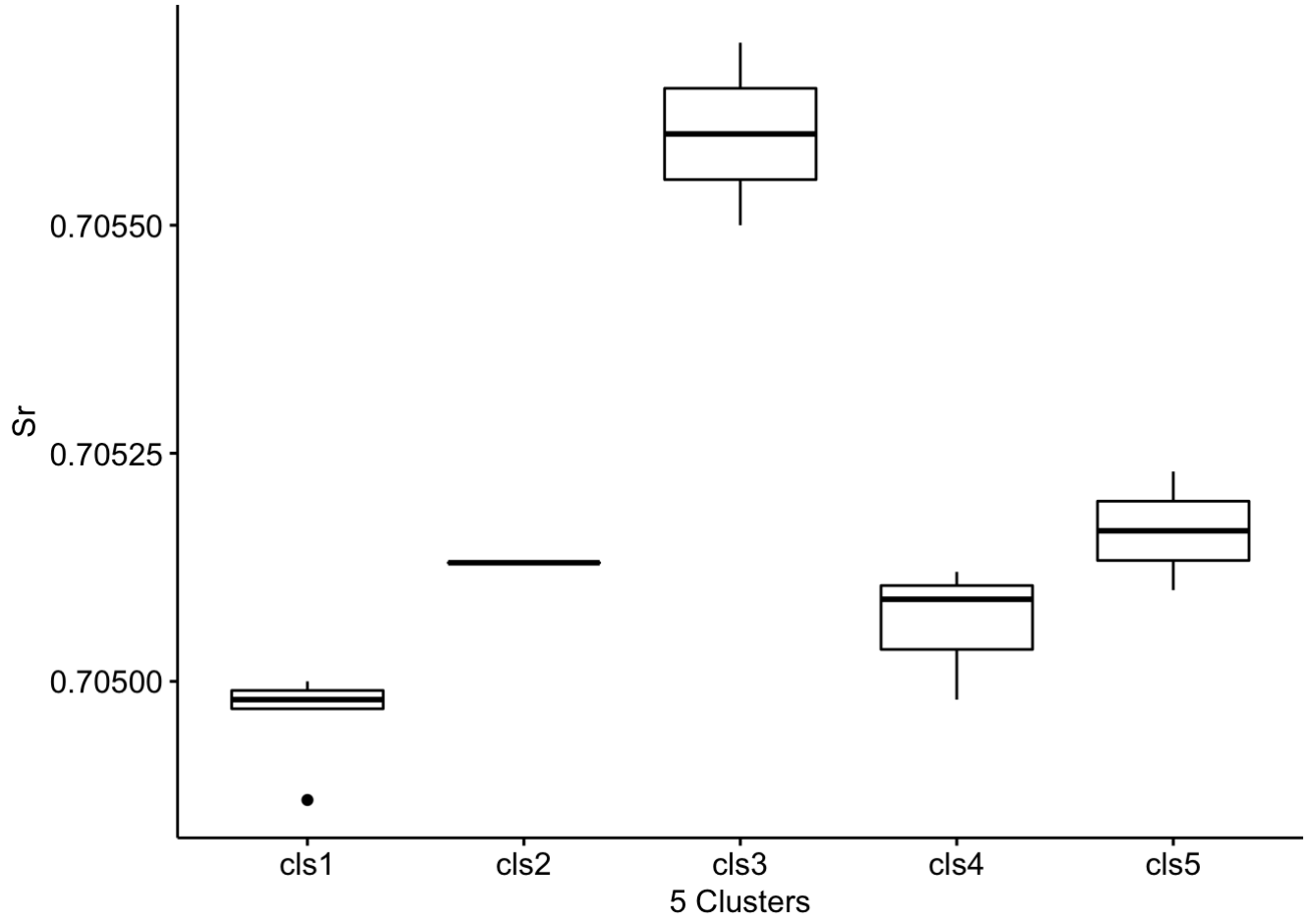
```
group_by(df, group5) %>%
  summarise(
    count = n(),
    mean = mean(Ophos, na.rm = TRUE),
    sd = sd(Ophos, na.rm = TRUE)
  )
```

```
## # A tibble: 5 x 4
##   group5 count  mean     sd
##   <ord>  <int> <dbl>  <dbl>
## 1 cls1       5  15.3  0.377
## 2 cls2       1  18.1 NA
## 3 cls3       2  15.7  0.990
## 4 cls4       3  14.4  0.265
## 5 cls5       2  15.8  0.636
```

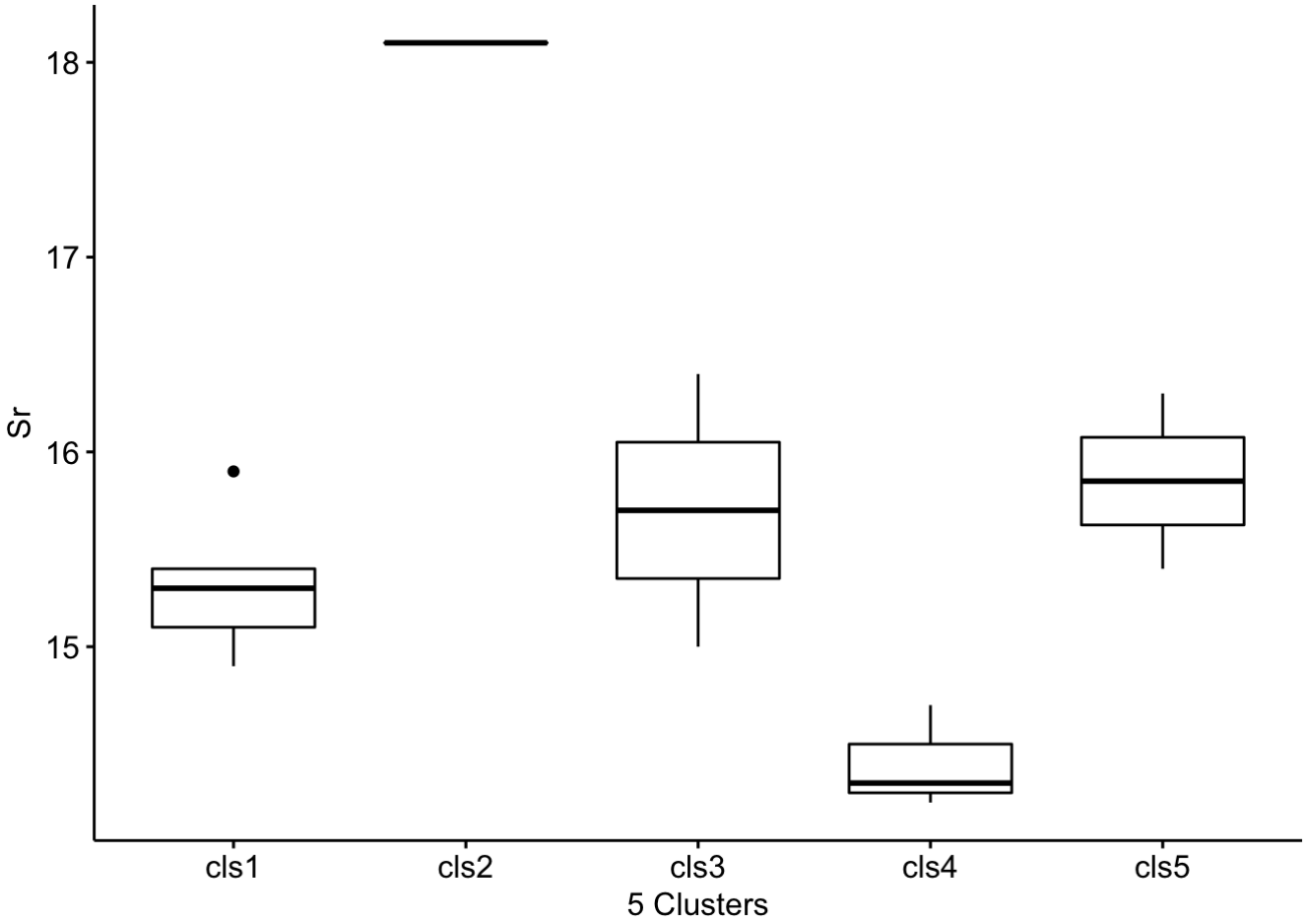
Visualize

Boxplots

```
ggboxplot(df, x = "group5", y = "Sr",
          order = c("cls1", "cls2","cls3", "cls4", "cls5"),
          ylab = "Sr", xlab = "5 Clusters")
```



```
ggboxplot(df, x = "group5", y = "Ophos",
          order = c("cls1", "cls2","cls3", "cls4", "cls5"),
          ylab = "Sr", xlab = "5 Clusters")
```



Compute

```
res.aov <- aov(Sr ~ group5, data = df)
summary(res.aov)
```

```
##             Df    Sum Sq   Mean Sq F value  Pr(>F)
## group5       4 5.986e-07 1.497e-07   23.76 0.00017 ***
## Residuals    8 5.040e-08 6.300e-09
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(res.aov)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Sr ~ group5, data = df)
##
## $group5
##                     diff           lwr           upr     p adj
## cls2-cls1  1.680000e-04 -1.323746e-04  0.0004683746 0.3736569
## cls3-cls1  6.380000e-04  4.085851e-04  0.0008674149 0.0000800
## cls4-cls1  1.013333e-04 -9.891638e-05  0.0003015830 0.4593750
## cls5-cls1  2.030000e-04 -2.641486e-05  0.0004324149 0.0861061
## cls3-cls2  4.700000e-04  1.341710e-04  0.0008058290 0.0083028
## cls4-cls2 -6.666667e-05 -3.832893e-04  0.0002499559 0.9440241
## cls5-cls2  3.500000e-05 -3.008290e-04  0.0003708290 0.9956472
## cls4-cls3 -5.366667e-04 -7.869788e-04 -0.0002863545 0.0005184
## cls5-cls3 -4.350000e-04 -7.092032e-04 -0.0001607968 0.0038557
## cls5-cls4  1.016667e-04 -1.486455e-04  0.0003519788 0.6424583
```

```
res.aov <- aov(Ophos ~ group5, data = df)
summary(res.aov)
```

```
##             Df Sum Sq Mean Sq F value  Pr(>F)
## group5       4 10.858  2.7144   10.38 0.00297 **
## Residuals    8  2.093  0.2616
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(res.aov)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Ophos ~ group5, data = df)
##
## $group5
##             diff        lwr         upr     p adj
## cls2-cls1  2.78  0.8442607  4.71573929 0.0071161
## cls3-cls1  0.38 -1.0984453  1.85844530 0.8936817
## cls4-cls1 -0.92 -2.2104929  0.37049286 0.1922158
## cls5-cls1  0.53 -0.9484453  2.00844530 0.7318345
## cls3-cls2 -2.40 -4.5642223 -0.23577768 0.0301023
## cls4-cls2 -3.70 -5.7404484 -1.65955163 0.0016230
## cls5-cls2 -2.25 -4.4142223 -0.08577768 0.0415255
## cls4-cls3 -1.30 -2.9131161  0.31311607 0.1249442
## cls5-cls3  0.15 -1.6170801  1.91708012 0.9980265
## cls5-cls4  1.45 -0.1631161  3.06311607 0.0805812
```

From all the significance tests, we see that only the 2-cluster model has statistically different p-values ($p<0.05$) between each group for both Sr and O-isotopes.