CIENCIA DE LOS DATOS: ESTUDIO DE CASO SOBRE LA PREDICCIÓN DE LA PROLIFERACIÓN DE ALGAS*

*Reproducido con fines estrictamente académicos.

Flor Margarita Calderón Cuevas

Escuela Superior De Ciencias y Tecnologías De La Información

Correo Electrónico: florcalderon@uagro.mx

RESUMEN

Este articulo tiene como propósito presentar un trabajo sobre un estudio de caso el cual conlleva algunas tareas de preprocesamiento de datos de minería de datos, análisis de datos exploratorios y construcción de modelos predictivos. En este caso de estudio se tomó un problema de predecir la frecuencia de aparición de varias algas dañinas en muestras de agua. El objetivo de este trabajo es la de bridar una mejor comprensión de que factores influyen en las frecuencias de las algas, para ello se hizo la recolección de varias muestras de agua en diferentes momentos durante un período de aproximadamente un año. En las muestras de agua se midieron las diferentes propiedades químicas, así como la frecuencia de aparición de siete algas marinas.

Palabras clave: *Algas, R Studio, Gráficas, Modelos, Predicción, Frecuencias.*

ABSTRACT

This article is intended to present work on a case study that involves some tasks of preprocessing data mining, analyzing exploratory data, and building predictive models. In this case study, a problem was taken to predict the frequency of occurrence of several harmful algae in water samples. The objective of this work is to provide a better understanding that factors influence the frequencies of algae, for this purpose the collection of several water samples was done at different times over a period of about one year. The water samples measured the different chemical properties as well as the frequency of appearance of seven seaweed.

Keywords: *Algae, R Studio, Graphs, Models, Prediction, Frequencies.*

1. INTRODUCCION

Las floraciones de algas tóxicas son eventos que ocurren cada vez más comúnmente en cuerpos de agua en todo el mundo, especialmente en las zonas templadas o tropicales, debido a la industrialización y a la agricultura intensiva. En los países desarrollados el problema de las floraciones nocivas es aún de mayor magnitud; en agosto de 2014 un evento de floración dejó a la población de la ciudad de Toledo, Ohio, sin agua potable durante varios días, debido al nivel de cianotoxinas detectadas en ella (Míguez Caramés, 2016).

Lo anterior constituye un grave problema ecológico con un fuerte impacto no solo en las formas de vida de los ríos, sino también en la calidad del agua. Por ello el poder realizar un monitoreo y un pronóstico temprano de la proliferación de algas es esencial para mejorar la calidad de los ríos o cuerpos de agua. Con el propósito de llevar a cabo este pronóstico a tiempo se recolectaron muestras de agua en diferentes ríos europeos en diferentes momentos durante un período de aproximadamente un año. Para cada muestra de agua, se midieron diferentes propiedades químicas, así como la frecuencia de aparición de siete algas dañinas. También se almacenaron algunas otras características del proceso de recolección de agua, como la estación del año, el tamaño del río y la velocidad del río (Torgo, 2016).

Las motivaciones al realizar el monitoreo químico son porque su aplicación es barata y fácilmente automatizada, mientras que su análisis biológico de las muestras para identificar las algas que están presentes en el agua implica un examen microscópico, requiere mano de obra capacitada y, por lo tanto, es tanto caro y lento. Además, los tratamientos de control usados a nivel internacional incluyen el agregado de aditivos químicos o biológicos para provocar su floculación (sulfato de cobre o de aluminio, paja de cebada envejecida, arcillas, o agregado de patógenos naturales tales como bacterias, virus y parásitos). Sin embargo, estos procedimientos pueden agregar compuestos nocivos para el medio ambiente o requerir mucho trabajo en su implementación y suelen ser tóxicos para el zooplancton, otros invertebrados y los peces. (Míguez Caramés, 2016).

Como tal, la obtención de modelos que sean capaces de predecir con precisión las frecuencias de las algas basándose en las propiedades químicas facilitaría la creación de sistemas baratos y automatizados para monitorear las floraciones de algas dañinas (Torgo, 2016).

1.1 OBJETIVO GENERAL

Tener una mejor comprensión de cuáles son los factores que influyen en la frecuencia de las algas, es decir como estas frecuencias se relacionan con los atributos químicos de las muestras de aguas, así como otras características de las muestras (estación del año, el tipo de río, etc.).

2. METODOLOGÍA

Para llevar a cabo el presente trabajo se utilizó una serie de pasos las cuales son: Descripción de los datos y cargar los datos en R, visualización y resumen de los datos, valores desconocidos, obtención de modelos de predicción, y evaluación selección de modelos.

2.1 Descripción de los datos y cargar los datos en R

Los datos están disponibles para este trabajo se recopilaron en el contexto de la investigación ERUDIT[1] Red y utilizado en el concurso internacional de análisis de datos COIL 1999. Está disponible en varias fuentes, como en el Repositorio de conjuntos de datos de aprendizaje automático de UCI[2].

Existen dos conjuntos de datos para este estudio. El primero consta de datos para 200 muestras de agua. Para ser más precisos, cada observación en los conjuntos de datos disponibles es en efecto una agregación de varias muestras de agua recolectadas del mismo río durante un período de 3 meses, durante la misma estación del año. Cada observación contiene información sobre 11 variables. Tres de estas variables son nominales y describen la época del año en que se recolectaron las muestras de agua a agregar, así como el tamaño y la velocidad del río en cuestión (Torgo, 2016). Las ocho variables restantes son valores de diferentes parámetros químicos medidos en las muestras de agua que forman la agregación, a saber:

- Valor máximo de pH
- Valor mínimo de O_2 (oxígeno)
- Valor medio de Cl (cloruro)
- Valor medio de $NO\overline{3}$(nitratos)
- Valor medio de NH_4^+(amonio)
- Media de PO_4^{3-} (ortofosfato)
- Media de PO_4 total (fosfato)
- Media de clorofila

Asociados con cada uno de los parámetros anteriores hay siete números de frecuencia de diferentes algas dañinas que se encuentran en las respectivas muestras de agua. No se proporciona información sobre los nombres de las algas identificadas. El segundo conjunto de datos contiene información sobre 140 observaciones adicionales. Utiliza la misma estructura básica pero no incluye información sobre las siete frecuencias de algas nocivas. Estas observaciones adicionales pueden considerarse como una especie de conjunto de pruebas. El objetivo principal de nuestro estudio fue predecir las frecuencias de las siete algas para estas 140 muestras de agua (Torgo, 2016).

Debido a lo anterior nos encontramos con una tarea de minería de datos predictiva, en la cual nuestro principal objetivo fue obtener un modelo que nos permita predecir el valor de una determinada variable objetivo dados los valores de un conjunto de variables predictoras. Este modelo también puede proporcionar indicaciones sobre qué variables predictoras tienen un mayor impacto en la variable objetivo; es decir, el modelo puede proporcionar una descripción completa de los factores que influyen en la variable objetivo (Torgo, 2016).

[1] http://www.erudit.de/erudit/.
[2] http://archive.ics.uci.edu/ml/.

Para poder realizar nuestro principal objetivo el cual es el de obtener el modelo de predicción se consideraron dos formas de llevar los datos a R Studio: el primero es simplemente aprovechando el paquete **dplyr** el cual incluye marcos de datos con los conjuntos de datos listos para usar; la otra opción es la de visitar el sitio web del repositorio de conjuntos de datos de aprendizaje automático de UCI, descargar los archivos de texto con los datos y luego cargarlos en R. Para nuestro caso se tomó la segunda opción. Se descargaron los archivos, se colocaron en el directorio de trabajo actual de nuestra sesión de R en ejecución, que se pudo verificar usando el comando **getwd ()**. Para leer los archivos solo basto con el emitir el siguiente comando en R Studio:

```
>getwd()

> algae <- read.table('Analysis.txt', header=FALSE, dec='.',col.names=c('season','size','speed','mxPH','mnO2','Cl',

'NO3','NH4','oPO4','PO4','Chla','a1','a2','a3','a4','a5','a6','a7'),  na.strings=c('XXXXXXX'))
```

2.2 Visualización y resumen de los datos

Después de haber cargado los datos y transformar los datos en una tabla de marco de datos con el comando **tibble**, se prosiguió hacer un análisis exploratorio; se utilizó el comando **summary** para obtener un resumen de sus estadísticas descriptivas, la cual nos proporcionó recuentos de frecuencia para cada valor posible. Por ejemplo, se pudo observar que hay más muestras de agua recogidas en invierno que en las demás estaciones del año. Para las variables numéricas, R nos dio una serie de estadísticas como su media, mediana, información de cuartiles y valores extremos. Estas estadísticas proporcionaron una primera idea de la distribución de los valores de las variables.

Se realizo un histograma de la variable **mxPH**, en la cual se pudo observar cómo los valores seguían una distribución aparentemente muy cercana la distribución normal, la cual se puede observar en la figura 2.1. Además, se puede observar cómo los valores se agrupan alrededor del valor medio.

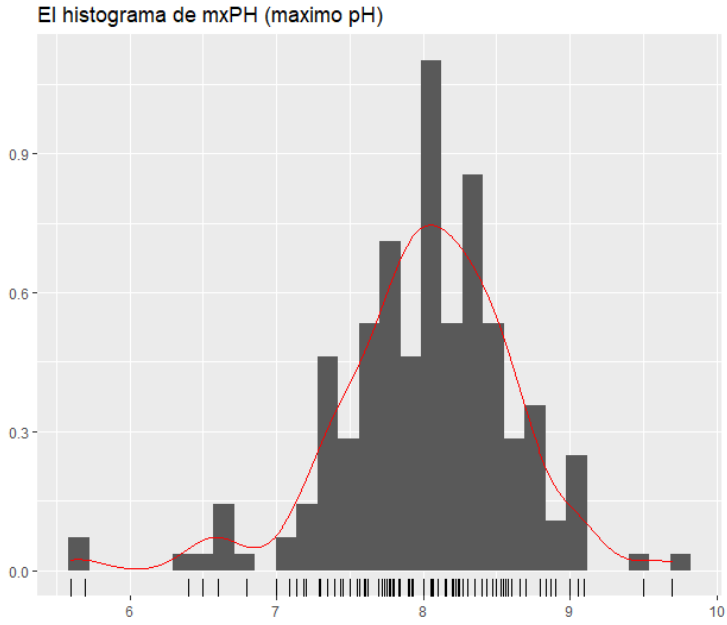

Figura 2.1 Histograma de la variable mxPH

A continuación, se muestra otro gráfico (Figura 2.2) en el cual traza los valores de las variables contra los cuantiles teóricos de una distribución normal (línea azul continua). La función también traza una envolvente con el intervalo de confianza del 95% de la distribución normal (líneas discontinuas azules). Como podemos observar, existen varios valores bajos de la variable que claramente rompen los supuestos de una distribución normal con un 95% de confianza.

Podemos observar que hay dos valores significativamente más pequeños que todos los demás. Este tipo de inspección de datos es muy importante ya que puede identificar posibles errores en la muestra de datos, o incluso ayudar a localizar valores que son tan incómodos que pueden ser solo errores.

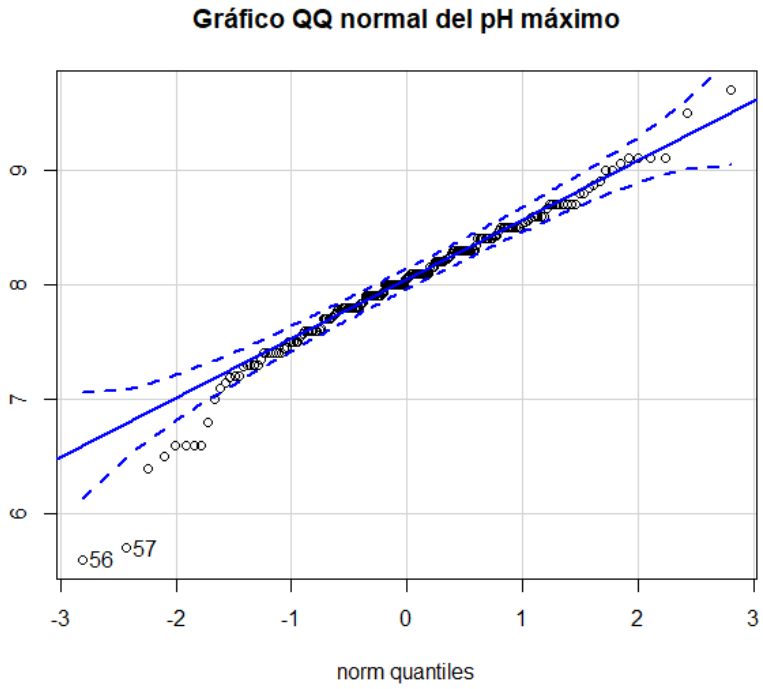

Figura 2.2. Gráfico normal de la variable PH máximo

Se estudio la distribución de los valores correspondientes a la Alga1, para ello se utilizó el diagrama de violín, en el cual se hizo uso de los valores de la alga1 frente al tamaño de los ríos (Figura 2.3). En el diagrama podemos observar la distribución de a1 para cada uno de los tamaños de río. Las áreas están hechas para tener el mismo tamaño y, por lo tanto, las regiones más amplias representan rangos de valores que tienen mayor peso en términos de la distribución de los valores. Ejemplo que podemos notar es que, para los ríos de tamaño medio, la mayoría de los valores de a1 se empaquetan cerca de cero, mientras que para los ríos más pequeños los valores están más distribuidos en el rango (violín más delgado).

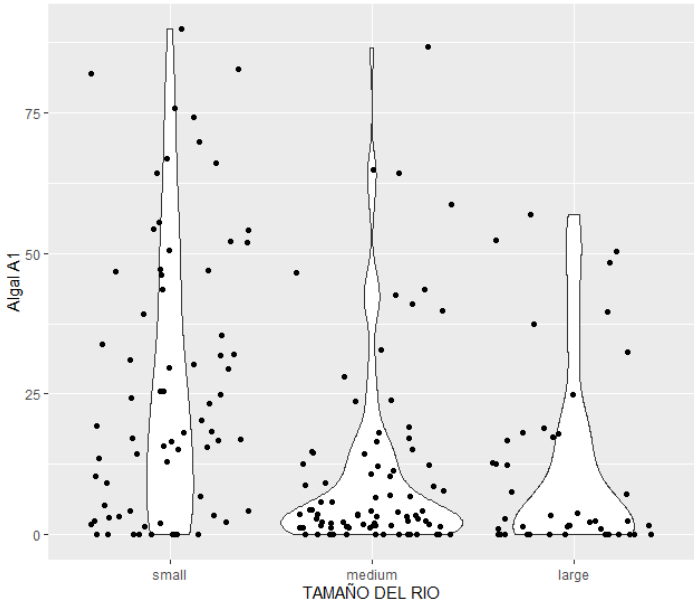

Figura 2.3 Diagrama de violín condicionado de alga a1

Por último, para esta parte de visulización se hizo un grafico para observar el comportamiento de la frecuencia de las algas a3 condicionadas por estación y la variable valor mínimo de O_2 (oxígeno). La figura 2.2.4 muestra dicho gráfico.

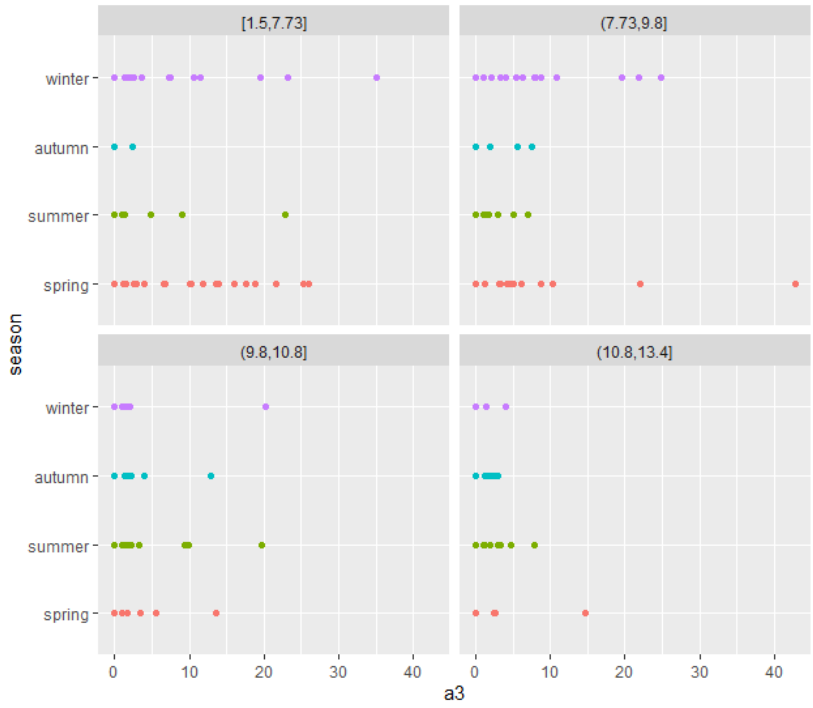

Figura 2.2. 4 Diagrama de puntos condicionado de alga a3 usando una variable continua.

2.3 Valores desconocidos

Hay varias muestras de agua con valores desconocidos en algunas de las variables. Esta situación, bastante común en problemas del mundo real, puede impedir el uso de ciertas técnicas que no pueden manejar valores perdidos. Siempre que manejamos un conjunto de datos con valores perdidos, podemos seguir varias estrategias. Los más comunes son:

- Retirar los estuches con incógnitas.
- Complete los valores desconocidos con los valores más frecuentes.
- Complete los valores desconocidos explorando las correlaciones entre variables.
- Complete los valores desconocidos explorando la similitud entre casos.
- Utilice herramientas que sean capaces de manejar estos valores.

La última alternativa es la más restrictiva, ya que limita el conjunto de herramientas que se pueden utilizar. Aun así, puede ser una buena opción siempre que confiemos en el mérito de las estrategias utilizadas por esas herramientas de minería de datos para manejar los valores perdidos (Torgo, 2016).

En nuestro caso se trabajó con la estrategia de completar valores desconocidos explorando similitudes entre los casos. El enfoque al que se asumió es que, si dos muestras de agua son similares, y una de ellas tiene un valor desconocido en alguna variable, existe una alta probabilidad de que este valor sea similar al valor de la otra muestra. Para utilizar este método intuitivamente atractivo, necesitamos definir la noción de similitud (Torgo, 2016). El método que se implementó fue la de la distancia euclidiana para encontrar los diez casos más similares de cualquier muestra de agua con algún valor desconocido en una variable, y luego se usó sus valores para completar la incógnita.

La idea se implementa en la función **knnImputation** (), la función usa una variante de la distancia euclidiana para encontrar los k vecinos más cercanos de cualquier caso. Esta variante permite la aplicación de la función a conjuntos de datos con variables nominales y continuas.

```
> algae <- knnImputation(algae, k = 10)
```

también se pudo utilizar la estrategia de utilizar los valores medianos para completar las incógnitas, y para ello se pudo usar la llamada:

```
> algae <- knnImputation(algae, k = 10, meth = "median")
```

En resumen, después de estas simples instrucciones, tenemos el marco de datos libre de valores NA y estamos mejor preparados para aprovechar al máximo varias funciones R.

2.4 Obtención de modelos de predicción

Nuestro principal objetivo era obtener las predicciones para los valores de frecuencia de las siete algas en un conjunto de 140 muestras de agua. Dado que estas frecuencias son números, se hizo una tarea de regresión. En esta sección exploraremos inicialmente dos modelos predictivos diferentes que un principio se tenía pensado aplicarse al dominio de las algas: regresión lineal múltiple y árboles de regresión.

2.4.1Regresión lineal múltiple

La regresión lineal múltiple se encuentra entre las técnicas de análisis de datos estadísticos más utilizadas. Estos modelos obtienen una función aditiva que relaciona una variable de destino con un conjunto de variables predictoras. Esta función aditiva es una suma de términos de la forma $\beta i \times Xi$, donde Xi es una variable predictora y βi es un número (Torgo, 2016).

Primero para la creación del modelo se eliminaron las muestras de agua 62 y 199 porque le faltaban seis de las once variables predictoras. El siguiente código que se muestra obtiene un modelo de regresión lineal para predecir la frecuencia de una de las algas que en este caso es la a1:

```
> lm.a1 <- lm(a1 ~ ., data = clean.algae[, 1:12])
```

La función anterior nos devuelve un modelo de regresión lineal; el cual nos indica que queremos un modelo que prediga la variable a1 utilizando todas las demás variables presentes en los datos. Para poder ver los detalles del modelo se hizo uso de la instrucción **summary(lm.a1).** Antes de pasar con la información que nos arrojó la función es importante tener en cuenta que cuando se aplica a modelos lineales, R maneja las tres variables nominales como un conjunto de variables auxiliares. A saber, para cada variable de factor con niveles k, R creará k − 1 variables auxiliares. Estas variables tienen los valores 0 o 1. Un valor de 1 significa que el valor asociado del factor está "presente", y eso también significará que las otras variables auxiliares tendrán el valor 0. Si todas las variables k − 1 son 0, entonces significa que la variable de factor tiene el valor kth restante. R creó tres variables auxiliares para la temporada de factores (seasonspring, seasonsummer y seasonwinter). Esto significa que, si tenemos una muestra de agua con el valor "autumn (otoño)" en la temporada variable, las tres variables auxiliares se establecerán en cero. Después de a ver planteado lo anterior, tenemos que la función **summary** nos proporcionó información diagnóstica sobre el modelo obtenido. En primer lugar, tenemos información sobre los residuos (es decir, los errores) del ajuste del modelo lineal a los datos utilizados. Estos residuos deben tener un cero medio y deben tener una distribución normal (¡y obviamente ser lo más pequeños posible!).

La proporción de variación explicada que brindó este modelo no es muy impresionante (alrededor del 32,0%). Así que se rechazó la hipótesis de que la variable de destino no depende de los predictores (el valor p de la prueba F es muy pequeño). Si nos fijamos en la importancia de algunos de los coeficientes, podemos cuestionar la inclusión de algunos de ellos en el modelo. Existen varios métodos para simplificar los modelos de regresión. Para poder simplificar el modelo se usó la función **anova(),** la cual nos indicó que la estación variable es la variable que menos contribuyo a la reducción del error de ajuste del modelo; se eliminó y se actualizó el modelo, pero el ajuste mejoró un poco (32,8%) pero no fue lo demasiado impresionante. Por último se usó la función **step()** para la búsqueda de modelo, se utilizó la eliminación hacia atrás de forma predeterminada; se obtuvo el modelo final:

```
Call:
lm(formula = a1 ~ size + mxPH + Cl + NO3 + PO4, data = clean.algae[,
    1:12])

Residuals:
    Min      1Q  Median      3Q     Max
-28.874 -12.732  -3.741   8.424  62.926

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 57.28555   20.96132   2.733  0.00687 **
sizemedium   2.80050    3.40190   0.823  0.41141
sizesmall   10.40636    3.82243   2.722  0.00708 **
mxPH        -3.97076    2.48204  -1.600  0.11130
Cl          -0.05227    0.03165  -1.651  0.10028
NO3         -0.89529    0.35148  -2.547  0.01165 *
PO4         -0.05911    0.01117  -5.291 3.32e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

La proporción de varianza explicada por este modelo no fue muy interesante.

2.4.2 Árboles de regresión

En esta sección describiremos brevemente la creación detrás del árbol de regresión de este caso de estudio. Para la creación del árbol se hizo uso del paquete **rpart**; para la visualización se utilizó el paquete **rpart.plot**, este paquete también incluye la función **prp()** que produce una representación gráfica agradable y altamente flexible de los árboles producidos por la función **rpart()**.

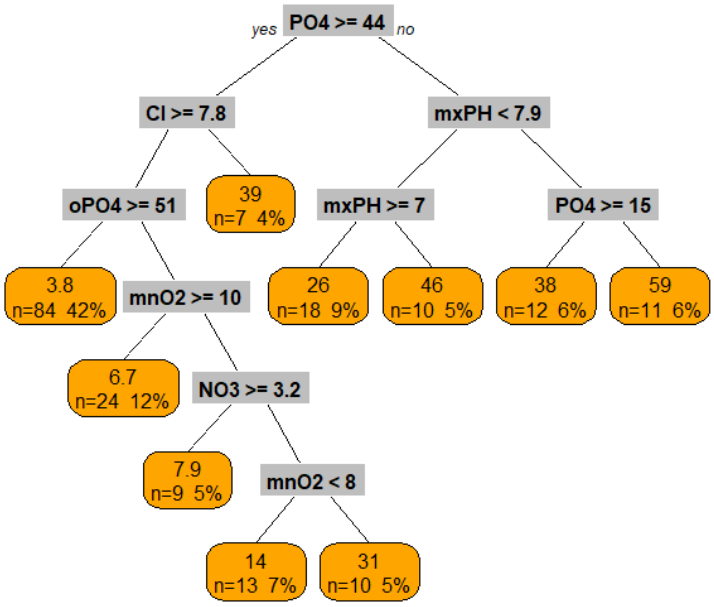

Figura 2. 4 Árbol de regresión para predecir el algas a1.

La figura 2.4 muestra el resultado de la función siguiente para la creación del árbol de regresión:

> library(rpart.plot)

> prp(rt.a1,extra=101,box.col="orange",split.box.col="grey")

R permitió una especie de poda interactiva de un árbol a través de la función **snip.rpart().** Esta función se puede utilizar para generar un árbol podado de dos maneras. El primero consiste en indicar el número de nodos (puede obtener estos números imprimiendo un objeto de árbol) en el que desea podar el árbol como se muestra a continuación:

```
>   snip.rpart(first.tree, c(4, 7))
n= 198

node), split, n, deviance, yval
      * denotes terminal node

 1) root 198 90401.290 16.996460
   2) PO4>=43.818 147 31279.120  8.979592
     4) Cl>=7.8065 140 21622.830  7.492857 *
     5) Cl< 7.8065 7  3157.769 38.714290 *
   3) PO4< 43.818 51 22442.760 40.103920
     6) mxPH< 7.87 28 11452.770 33.450000
      12) mxPH>=7.045 18  5146.169 26.394440 *
      13) mxPH< 7.045 10  3797.645 46.150000 *
     7) mxPH>=7.87 23  8241.110 48.204350 *
```

Por último, se graficó el árbol utilizando **snip.part,** para ello primero se trazó el árbol y después se llamó a la función. Con este tipo de grafica se pudo interactuar, y haciendo clic con el ratón de algún nodo, R imprime en su consola información sobre el nodo. Si vuelve a hacer clic en ese nodo, R poda el árbol en ese nodo. Puede ir en nodos de poda de esta manera gráfica. Para finalizar la interacción, se da clic con el botón derecho del ratón. El resultado de la llamada dio de nuevo un objeto de árbol (Figura 2.5).

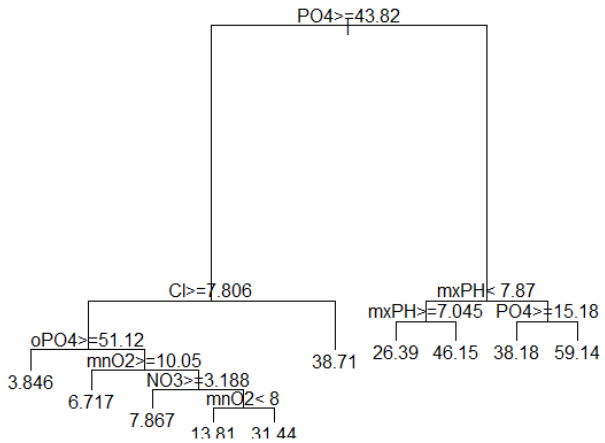

Figura 2. 5 Gráfica del árbol de regresión

2.6 Evaluación y selección de modelos

En el punto 2.5 se explicó dos ejemplos de modelos de predicción que se podrían haber utilizados en este caso de estudio. Después de hacer el análisis de los modelos anteriores surgió la pregunta ¿cuál se debía usar para obtener las predicciones para las siete algas de las 140 muestras de prueba? Para responder a esa pregunta fue necesario la evaluación de los modelos para verificar su rendimiento. El rendimiento predictivo de los modelos de regresión se obtiene comparando las predicciones de los modelos con los valores reales de las variables de destino y calculando alguna medida de error promedio de esta comparación. Una de esas medidas es el error absoluto medio (MAE) (Torgo, 2016).

El primer paso para la obtención de las predicciones del modelo para el conjunto de casos en los que queríamos evaluar, se utilizó la función **predict(),**esta función general recibe un modelo y un conjunto de datos de prueba y recupera las predicciones de modelo correspondientes:

```
> lm.predictions.a1 <- predict(final.lm, clean.algae)

> rt.predictions.a1 <- predict(rt.a1, algae)
```

Las dos funciones anteriores recogieron las predicciones de los modelos obtenidos en la sección 2. para algas a1. Después de tener las predicciones se prosiguió a calcular su error medio:

```
> (mae.a1.lm <- mean(abs(lm.predictions.a1 - algae[["a1"]])))

[1] 13.10681

> (mae.a1.rt <- mean(abs(rt.predictions.a1 - algae[["a1"]])))

[1] 8.480619
```

También otra medida de error popular que se utilizo fue el error cuadrado medio (MSE). Esta medida pudo obtenerse de la siguiente manera:

```
> (mse.a1.lm <- mean((lm.predictions.a1 - algae[["a1"]])^2))

[1] 295.5407

> (mse.a1.rt <- mean((rt.predictions.a1 - algae[["a1"]])^2))

[1] 161.9202
```

Esta última estadística tiene la desventaja de no medirse en las mismas unidades que la variable objetivo y, por lo tanto, ser menos interpretable desde la perspectiva del usuario. Por ello se usó una estadística alternativa que proporciona una respuesta razonable a esta pregunta es el error cuadrado medio normalizado (NMSE). Esta estadística calcula una relación entre el rendimiento de nuestros modelos y el de un predictor de línea base, normalmente tomado como valor medio de la variable de destino. También fue interesante tener algún tipo de inspección visual de las predicciones de los modelos. Se creo una gráfica de dispersión de los errores. La Figura 2.6 muestra un ejemplo de este tipo de análisis para las predicciones de nuestros dos modelos.

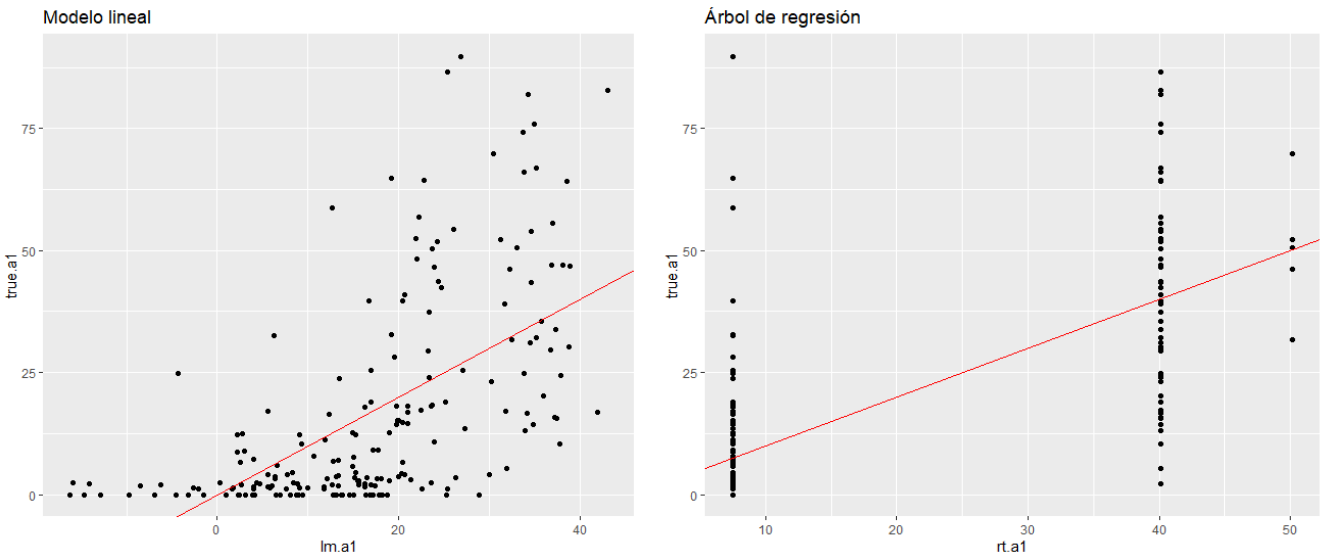

Figura 2. 6 Diagrama de dispersión de errores.

Después de haber obtenido las medidas de rendimiento calculadas anteriormente, se pensó tomar el árbol de regresión para obtener las predicciones para las 140 muestras de prueba, ya que obtuvo un NMSE más bajo. Sin embargo, había una trampa en ese razonamiento, porque nuestro objetivo era elegir el mejor modelo para obtener las predicciones en las 140 muestras de prueba. Como no conocíamos los valores variables de destino para esas muestras, teníamos que estimar cuál de nuestros modelos funcionará mejor en estas muestras de prueba. Entonces se decidió usar el paquete **performanceEstimation** el cual está diseñado para estos problemas de comparación y selección de modelos. Después de haber aplicado el paquete anterior se obtuvieron resultados muy malos. Con lo que se implementó la idea de usar la función **randomForest()**, ya que para la mayoría de los problemas la mejor puntuación es obtenida por alguna variante de un bosque aleatorio. Sin embargo, los resultados no salieron son muy buenos, en particular para las algas 7. Se uso la función **pairedComparisons()** del performance del paquete **performanceEstimation**, la cual realizo una serie de ensayos estadísticos que pueden utilizarse para comprobar la validez estadística de ciertas hipótesis relativas a las diferencias observadas entre el rendimiento de los diferentes flujos de trabajo. Nuestra idea fue comprobar si la diferencia entre el rendimiento de este bosque aleatorio y los otros flujos de trabajo alternativos es estadísticamente significante o no. Esto significa que estábamos comparando una línea base con una serie de alternativas en un conjunto de tareas. Como conclusión en este paso se pudo rechazar con un 95% de confianza, la hipótesis de que el rendimiento de la línea base es el mismo que el rendimiento de dos de los árboles de regresión y el modelo de regresión lineal. En resumen, estamos lo suficientemente seguros como para decir que nuestra línea de base es mejor que el modelo lineal y dos de los árboles de regresión en estas siete tareas de regresión. Sin embargo, no podemos rechazar la hipótesis de que su rendimiento no es mejor que el de los demás flujos de trabajo (al menos con un 95% de confianza). El análisis anterior se pudo llevar cabo visualmente a través de diagramas de CD. La Figura 2.7 presenta el diagrama de CD correspondiente a las pruebas estadísticas descritas anteriormente.

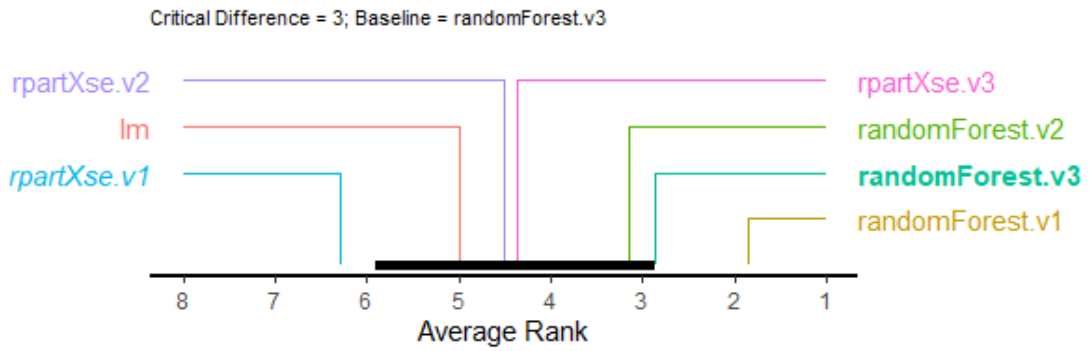

Figura 2. 7 El diagrama de CD para comparar todos los flujos de trabajo con randomforest.v3.

3. RESULTADOS

En esta sección se explica cómo se obtuvo las predicciones para las siete algas en las 140 muestras de prueba. El procedimiento utilizado consistió en obtener estimaciones imparciales de la NMSE (error cuadrático medio normalizado) para un conjunto de modelos en las siete tareas predictivas, mediante un proceso experimental de validación cruzada. Como nuestro objetivo era obtener siete predicciones para cada una de las 140 muestras de prueba se obtuvo utilizando el modelo que nuestro proceso de validación cruzada ha indicado como el "mejor" para esa tarea. Este fue uno de los modelos mostrados por una llamada a la función **rankWorkflows():**

> rankWorkflows(res.all, top=3)

Para obtener los modelos se utilizó todos los datos de entrenamiento disponibles para que podamos aplicarlos al conjunto de pruebas. El código siguiente obtuvo los mejores flujos de trabajo para cada una de las siete algas:

> wfs <- sapply(taskNames(res.all), function(t) topPerformer(res.all,metric="nmse",task=t))

> wfs[["a1"]]

> wfs[["a7"]]

Se utilizó la función **taskNames** para obtener un vector con los nombres de las siete tareas de predicción y luego para cada uno de estos nombres se aplicó una función que utiliza esencialmente la función **topPerformer()** para obtener el flujo de trabajo que es el mejor en una determinada tarea en una métrica determinada. Como resultado, el objeto **wfs** será una lista con cada 7 objetos de clase **Workflow**. Función **runWorkflow()** se pudo utilizar para aplicar cualquiera de estos flujos de trabajo a algunos conjuntos de trenes y pruebas determinados. Como resultado, esta función devolvió el resultado de esta aplicación, que depende del autor del flujo de trabajo. En nuestro caso utilizamos los flujos de trabajo estándar implementados por la función **standardWF()** del paquete **peformanceEstimation**.

Este flujo de trabajo devuelve como resultado una lista con varios componentes, entre los que se indican las predicciones del modelo aprendido para el conjunto de pruebas determinado. Ya después de lo anterior se siguió a obtener la matriz con las predicciones de los mejores flujos de trabajo para todo el conjunto de pruebas:

full.test.algae <- cbind(test.algae, algae.sols)

```
pts <- array(dim = c(140,7,2),dimnames = list(1:140, paste0("a",1:7), c("trues","preds")))
for(i in 1:7) { res <- runWorkflow(wfs[[i]],
                    as.formula(paste(names(wfs)[i],"~.")),
                    algae[,c(1:11,11+i)],
                    full.test.algae[,c(1:11,11+i)])
    pts[,i,"trues"] <- res$trues
    pts[,i,"preds"] <- res$preds
    }
```

Se comenzó colocando los casos de prueba y las soluciones respectivas en un único marco de datos. A continuación, se creó una matriz (pts) con 3 dimensiones que almacenará toda la información sobre la aplicación de los modelos para realizar predicciones para las siete algas. Es como tener dos matrices de 140 × 7 (donde 140 es el número de casos de prueba y 7 el número de algas predichas para cada caso de prueba). La primera de estas matrices contiene los valores verdaderos de las algas, mientras que la segunda contiene las predicciones de nuestros flujos de trabajo. Por ejemplo, para conocer la predicción y los verdaderos valores de las algas "a1" y "a3" en los primeros 3 casos de prueba los obtenemos de la siguiente manera:

```
> pts[1:3,c("a1","a3"),]
, , trues
  a1  a3
1 1.2 1.9
2 1.2 0.0
3 7.0 6.5
, , preds
        a1       a3
1  3.592257 4.789750
2  9.930629 3.457400
3 12.344686 7.023392
```

Esta matriz se rellena aplicando sucesivamente los mejores flujos de trabajo para cada una de las 7 algas mediante la función **runWorkflow()**. Para ello se tuvo que crear una fórmula adecuada para cada tarea predictiva, así como utilizar las columnas correctas de los datos originales para obtener el modelo. El resultado de la llamada a **runWorkflow()** (en realidad una llamada a **standardWF()** que es el flujo de trabajo específico que estamos utilizando) es una lista que contiene, entre otros, los componentes **trues** y **preds**, con los valores verdaderos y

predichos, respectivamente. Utilizando la información almacenada en la matriz (pts) se pudo comparar las predicciones con los valores reales para obtener algunos comentarios sobre la calidad de nuestro enfoque con este problema de predicción. El código siguiente calcula las puntuaciones NMSE de nuestros modelos en las siete algas:

```
> avg.preds <- apply(algae[,12:18], 2, mean)

> apply((pts[,,"trues"] - pts[,,"preds"])^2, 2 ,sum) / apply( (scale(pts[,,"trues"], avg.preds, FALSE))^2, 2, sum)
```

a1	a2	a3	a4	a5	a6	a7
0.4739169	0.8608667	0.7749362	0.7259074	0.7154015	0.8113643	1.0000000

El código anterior lo que hace es que primero obtiene las predicciones del modelo de línea base utilizado para calcular el NMSE, que en nuestro caso consistió en predecir el valor medio de la variable de destino. A continuación, se procedió a calcular los NMSE para los siete modelos/algas. La función scale() se puede utilizar para normalizar un conjunto de datos. Funciona restando el segundo argumento del primero y, a continuación, dividiendo el resultado por el tercero, a menos que este argumento sea FALSE, como es el caso anterior. En este ejemplo lo estamos utilizando para restar un vector (el valor objetivo promedio de las siete algas) de cada línea de una matriz.

4. CONCLUSIONES

Los resultados que se obtuvieron en este caso de estudio están de acuerdo con las estimaciones de validación cruzada obtenidas anteriormente. Confirman la dificultad de obtener buenas puntuaciones para alga 7, mientras que para los otros problemas los resultados son un poco más competitivos, en particular para algas 1. En resumen, con una fase de selección de modelos adecuada, se pudo obtener puntuaciones interesantes para estos problemas de predicción.

Las conclusiones para este caso de estudio fueron las siguientes:

- La presencia de varios valores atípicos distorsiono el valor de la media como una estadística de centralidad (es decir, que indica el valor más común de la variable) por ello se tuvo que completar los valores desconocidos explorando similitudes entre casos.
- La variable oPO4 tuvo una distribución de los valores observados claramente concentrada en valores bajos, por lo tanto, con un sesgo positivo.
- En la mayoría de las muestras de agua, el valor de oPO4 estuvo bajo, pero hubo varias observaciones con valores altos, e incluso con valores extremadamente altos, lo cual nos permitió observar que se esperan frecuencias más altas de algas a1 en ríos más pequeños, lo que puede ser un conocimiento valioso.
- Los valores de mxPH no están seriamente influenciados por la estación del año en que se recolectaron muestras.
- Se pudo observar una tendencia de los ríos más pequeños a mostrar valores más bajos de mxPH.
- Se muestra la variación de mxPH para todas las combinaciones de tamaño y velocidad de los ríos. Algo curioso que se noto es que no existe información sobre ríos pequeños con baja velocidad. La única muestra que se tuvo de estas propiedades es exactamente la muestra 48, ¡aquella para la que no nunca conocimos el valor de mxPH!

- En nuestro árbol que lee desde el nodo raíz marcado por R con el número 1. R proporciono información sobre los datos de este nodo. Se pudo observar que teníamos 198 muestras (los datos generales de entrenamiento utilizados para obtener el árbol) en este nodo; las 198 muestras tuvieron un valor medio para la frecuencia de algas a1 de 16,99, y que el desviación de esta media es de 90401,29. Cada nodo de nuestro árbol tiene dos ramas. Estos estuvieron relacionados con el resultado de una prueba en una de las variables predictoras. Por ejemplo, desde el nodo raíz pudimos observar que había una rama (etiquetada por R con "2)") para los casos en los que la prueba"PO4 ≥43.818" es verdadera (147 muestras); y también una rama para los 51 casos restantes que no satisfacen esta prueba (marcada por R con "3)"). Desde el nodo 2 teníamos otras dos ramas que conducían a los nodos 4 y 5, dependiendo del resultado de una prueba en Cl. Esta prueba continúa hasta que se alcanza un nodo hoja. Estos nodos estaban marcados con asteriscos por R.

- Para usar el árbol y obtener una predicción para una muestra de agua en particular, sólo se necesitaba seguir una rama desde el nodo raíz hasta una hoja, de acuerdo con el resultado de las pruebas para esta muestra. El valor variable objetivo promedio que se encuentra en la hoja que hemos alcanzado es la predicción del árbol.

En este estudio de caso expuesto en un artículo proporciono información sobre:

- Visualización de datos
- Estadísticas descriptivas
- Estrategias para manejar valores variables desconocidos
- Tareas de regresión
- Métricas de evaluación para tareas de regresión
- Regresión lineal múltiple
- Árboles de regresión
- Selección/comparación del modelo a través de la validación cruzada **k-fold**.
- Conjuntos de modelos y bosques aleatorios.

5. REFERENCIAS

Míguez Caramés, D. M. (2016). Tecnologías de control de floraciones de cianobacterias y algas nocivas en cuerpos de agua, con énfasis en el uso de irradiación por ultrasonido. *INNOTEC*, 54-61.

Torgo, L. (2016). *Data mining with R: learning with case studies Second Edition.* Portugal: CRC Press.