Taller de Análisis de datos - Problema de clasificación 1

Jésica Charaf e Ignacio Spiousas

12 de diciembre de 2023

Problema de clasificación 1

Estos datos son los resultados de análisis químicos de vinos provenientes de la misma región de Italia pero de 3 distintos cultivos. Cada una de las 178 filas contiene el número del cultivo seguido por los valores de 13 mediciones.

Aplique los métodos de clasificación que le parezcan convenientes y compare sus performances.

Los datos están en http://archive.ics.uci.edu/ml/datasets/Wine

Resolución

Análisis exploratorio

Lo primero que vamos a ver es como se distribuyen las clases, es decir, cuántos datos pertenecientes a cada cultivo tenemos (figura 1).

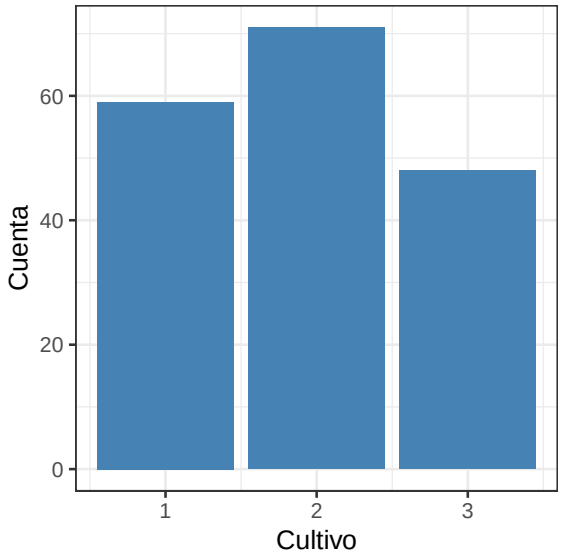

Figure 1: Cantidad de datos pertenecientes a cada clase (cultivo) en el dataset a utilizar.

Podemos ver que no contamos con grandes desbalances de clase.

NOTA: Acá poner un poco más de exploración de la relación de los cultivos con las variables.

Elección del método de clasificación

Para analizar los distintos métodos de clasificación, separamos la muestra en un set de entrenamiento (dos tercios de los datos) y un set de testeo (un tercio de los datos) de forma estratificada según el cultivo, utilizando la función `initial_split` de *{rsample}*.

La métrica que vamos a utilizar para el modelo es el *accuracy* ya que los datos no presentan desbalances de clases marcados ni creemos que haya algunos de los errores (tipo I y tipo II) que debamos favorecer por sobre el otro.

K vecinos cercanos

El primer modelo que vamos a ajustar es el de K vecinos cercanos. Para esto consideramos una grilla de valores de k (cantidad de vecinos) entre 1 y 20. Para evaluar cuál es la cantidad de vecinos más conveniente realizamos validación cruzada separando la muestra de entrenamiento en 10 folds estratificando según la clase. Estos *folds* son generados utilizando la función `vfold_cv` del paquete *{rsample}*.

Para utilizar el modelo de KNN primero vamos a escalear los datos.

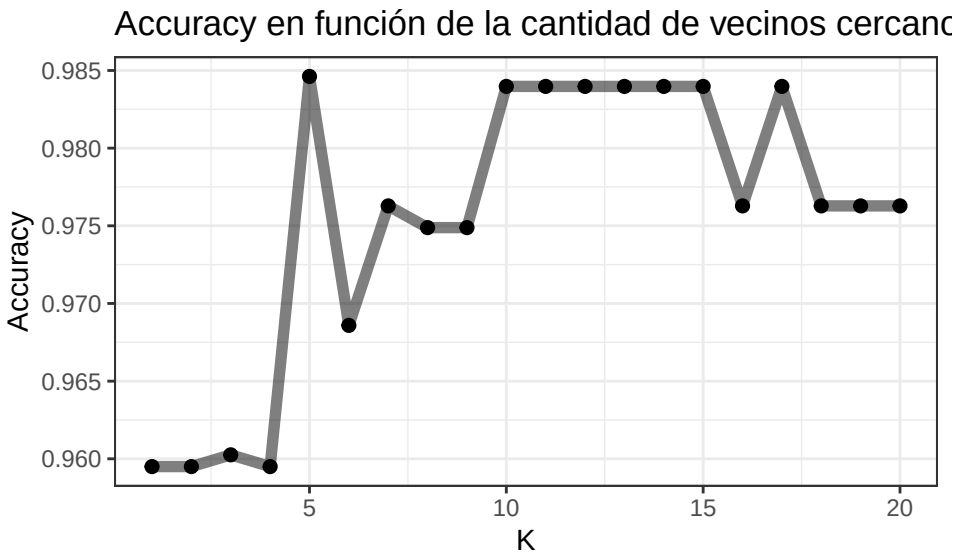

Figure 2: Accuracy en función de la cantidad de vecinos cercanos.

Como puede verse en la figura 2, el máximo de Accuracy para los modelos de KNN es de 0.98 para 5 vecinos.

Random Forest

la siguiente alternativa que vamos a considerar es un modelo basado en ensambles de árboles conocido como Random Forest. En este caso también utilizaremos validación cruzada para hallar la combinación de parámetros que maximice la Accuracy. Los hiperparámetros a optimizar en un modelo de Random Forest son: el número de variables que se consideran en cada split del árbol aleatorio (`mtry`); y el número mínimo de observaciones requeridas para que una hoja se bifurque (`min_n`).

Vamos a calcular el Accuracy para una grilla de 160 filas, con $1 \leq$`mtry`≤ 10 y $5 \leq$`min_n`≤ 20. Al igual que en KNN, utilizaremos los datos estandarizados (aunque en este caso no debería afectar a los resultados).

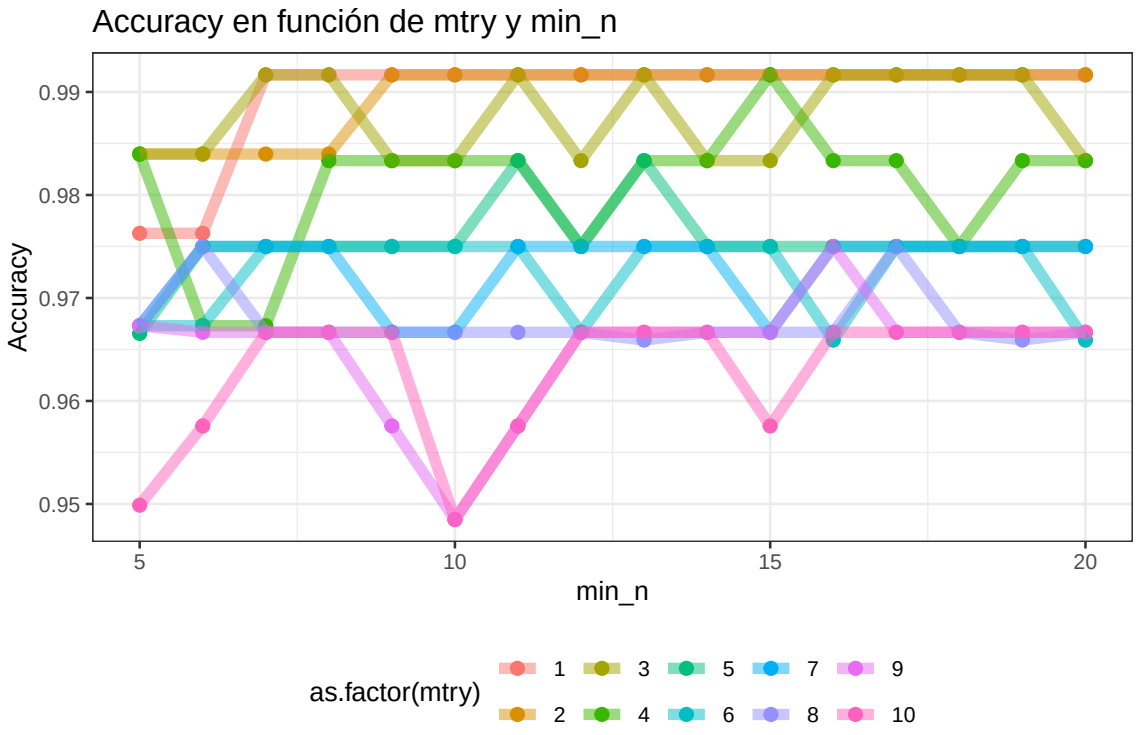

Figure 3: Accuracy en función de mtry y minn ara Random Forest.

Como puede verse en la figura 3, la máxima accuracy vale 0.992 para m_try igual a 1 y min_n igual a 7.