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

Los datos contienen 178 observaciones donde la primera variable indica el tipo de cultivo (1, 2 o 3) y las siguientes 13 variables corresponden a mediciones de: Alcohol, *Malicacid*, *Ash*, *Alcalinity_of_ash*, *Magnesium*, *Total_phenols*, *Flavanoids*, *Nonflavanoid_phenols*, *Proanthocyanins*, *Color_intensity*, *Hue*, *0D280_0D315_of_diluted_wines* y *Proline*.

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

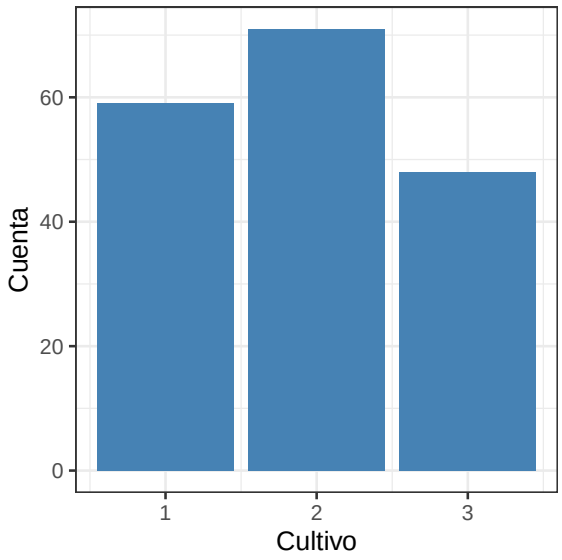

Figure 1: Cantidad de datos pertenecientes a cada clase (cultivo) en el dataset a utilizar.

Podemos ver que no contamos con grandes desbalances de clase. Se tienen 59 observaciones correspondientes al cultivo 1, 71 del cultivo 2 y 48 del cultivo 3.

Para seguir, en la figura 2 se observan los boxplots y violin plots de las distintas variables según la clase de cultivo.

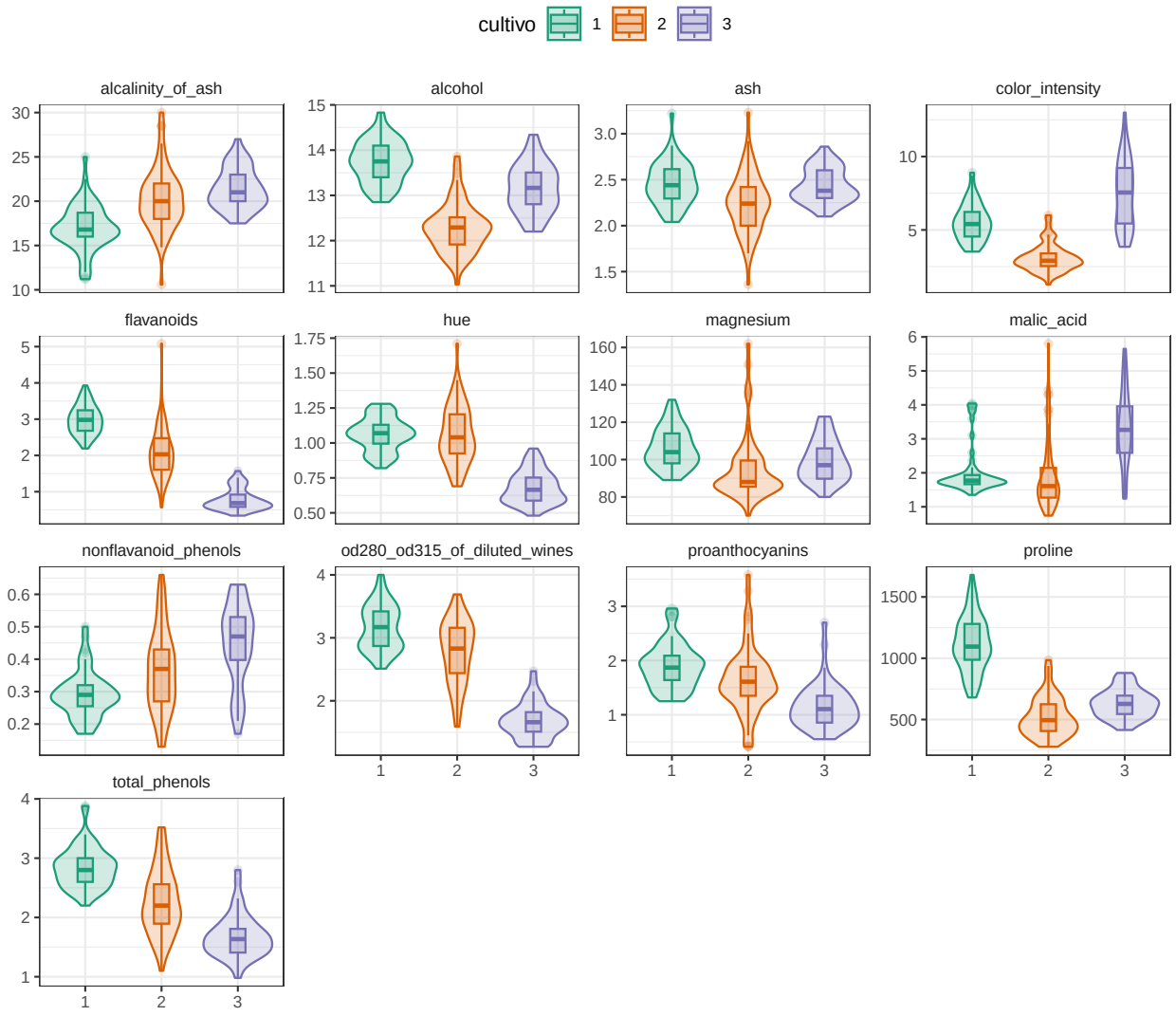

Figure 2: Boxplots y violin plots de las 13 variables según el tipo de cultivo.

Allí podemos ver que hay algunas variables que presentan una clara separación por tipo de cultivo y pueden ser relevantes en la clasificación, tales como *alcohol*, *total_phenois*, *flavanoids* y *color_intensity*. A modo de ejemplo, en el caso de la variable *alcohol* vemos que las mediciones son superiores en el cultivo 1, siguiendo el cultivo 3 y luego el 2. Por otra parte, tenemos variables como *ash* y *magnesium* que parecen diferenciarse menos a través de los grupos. Y por último, se observan casos de mediciones como *hue* y *proline* que se distinguen en uno de los cultivos en comparación con los otros dos. Por ejemplo, vemos que las mediciones de *hue* son menores en el cultivo 3, mientras que los otros dos cultivos son más díficiles de diferenciar entre sí a simple vista.

Elección de los modelos de clasificación

El objetivo del trabajo consiste en explorar diferentes métodos de clasificación para determinar el tipo de cultivo y comparar sus desempeños. Para esto, vamos a considerar los enfoques de K vecinos cercanos, Random Forest, Regresión logística (para modelos multinomiales) y Redes neuronales.

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 evaluar los distintos modelos es el *accuracy* ya que los datos no presentan desbalances de clases marcados ni creemos que haya alguno de los errores 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 consideraremos 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}*.

Como el método de KNN puede verse afectado por variaciones de escala, en cada paso de validación cruzada estandarizamos los datos del subconjunto que se utiliza como entrenamiento y, con esa misma transformación, escalamos los datos del subconjunto sobre el cual se predice.

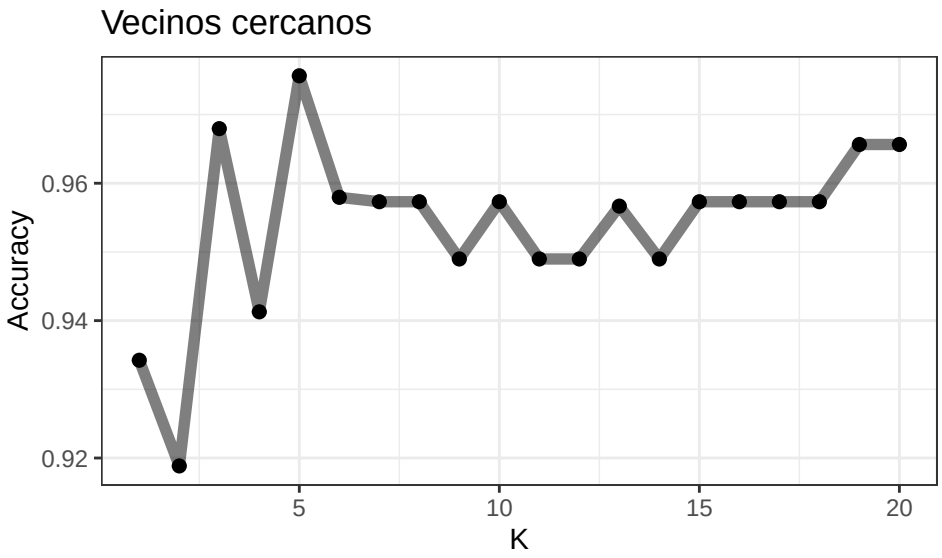

Figure 3: Accuracy en función de la cantidad de vecinos cercanos.

Como puede verse en la figura 3, 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 el Accuracy, considerando los mismos folds que en K vecinos cercanos pero sin estandarizar los datos. Los hiperparámetros que vamos a optimizar en el 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`$\leq$ 10 y 5 $\leq$`min_n`$\leq$ 20.

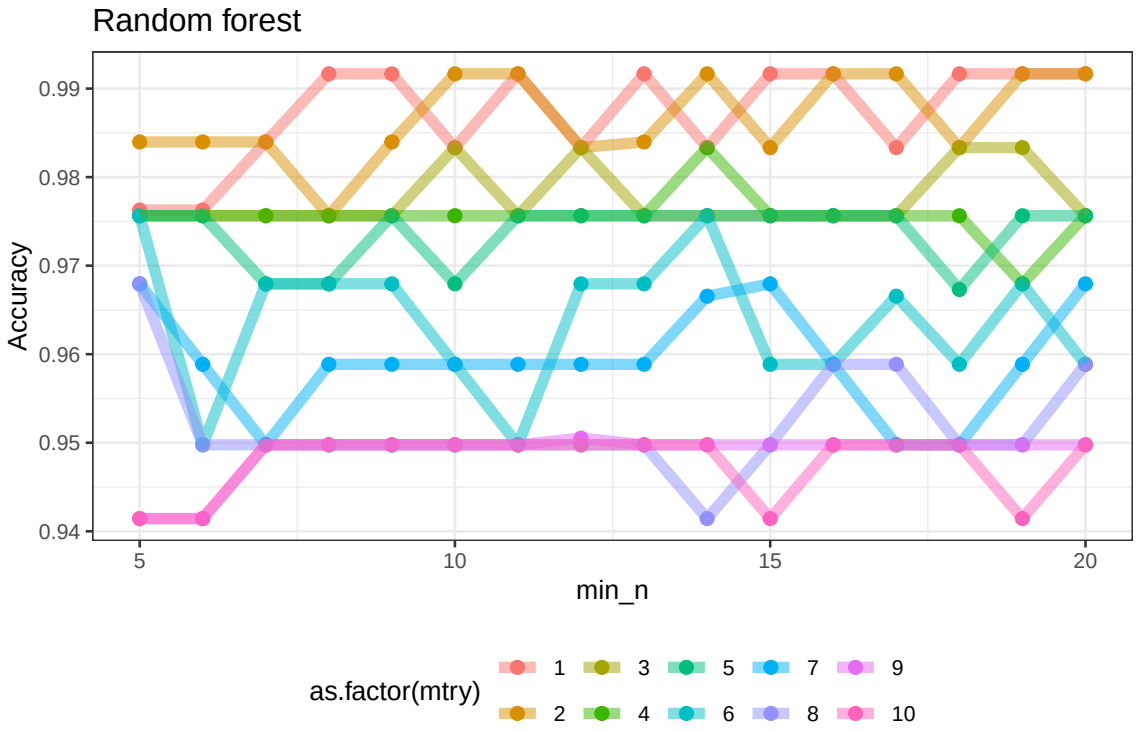

Figure 4: Accuracy en función de mtry y minn para Random Forest.

Como puede verse en la figura 4, el máximo valor de accuracy vale 0.992 para m_try igual a 1 y min_n igual a 8.

Regresión logística

Para continuar, otro enfoque que exploraremos es el de regresión logística para modelos multinomiales. Vamos a ajustar una familia de modelos con regularización Lasso utilizando el paquete glmnet.

Para esto, armamos una grilla de valores de λ tomando 100 valores entre 10^{-3} y 10^{0}. Para cada valor de λ, evaluamos el Accuracy realizando validación cruzada de la misma forma que en los métodos previos.

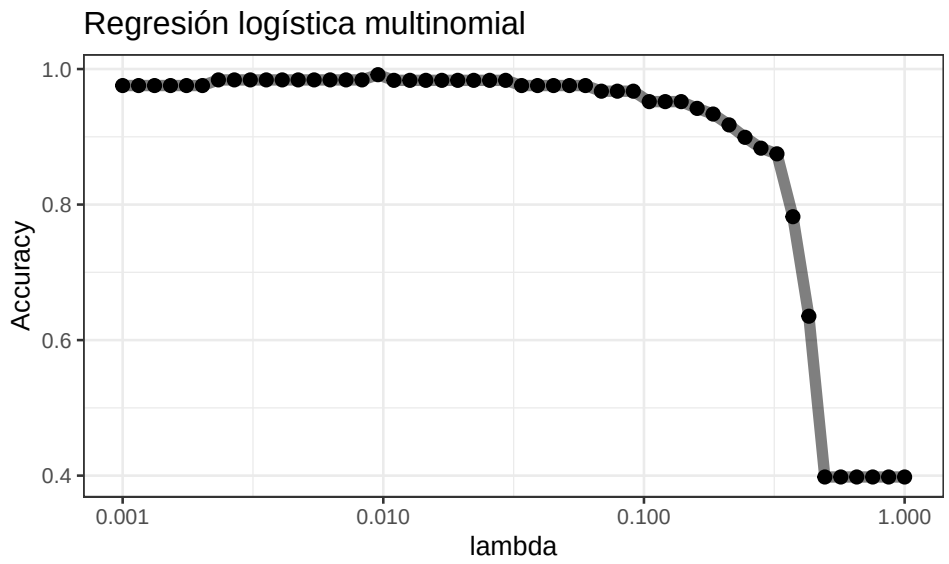

Figure 5: Accuracy en función del valor de lambda.

En la figura 5 observamos los resultados del Accuracy en función de λ y vemos que el máximo se alcanza en λ igual a 0.01 con un valor de 0.992.

Redes neuronales

Finalmente, vamos a considerar un modelo de redes neuronales para predecir el cultivo. Por simplicidad el modelo contiene una sola capa intermedia y por medio de validación cruzada vamos a determinar cuántas neuronas conviene incluir en esta capa. Para esto, consideramos una grilla de valores entre 1 y 10 para la cantidad de neuronas y buscamos el número que maximice el Accuracy.

En este caso, al igual que en K vecinos cercanos, estandarizamos los datos en cada paso de validación cruzada ya que obtuvimos mejores resultados de esta manera.

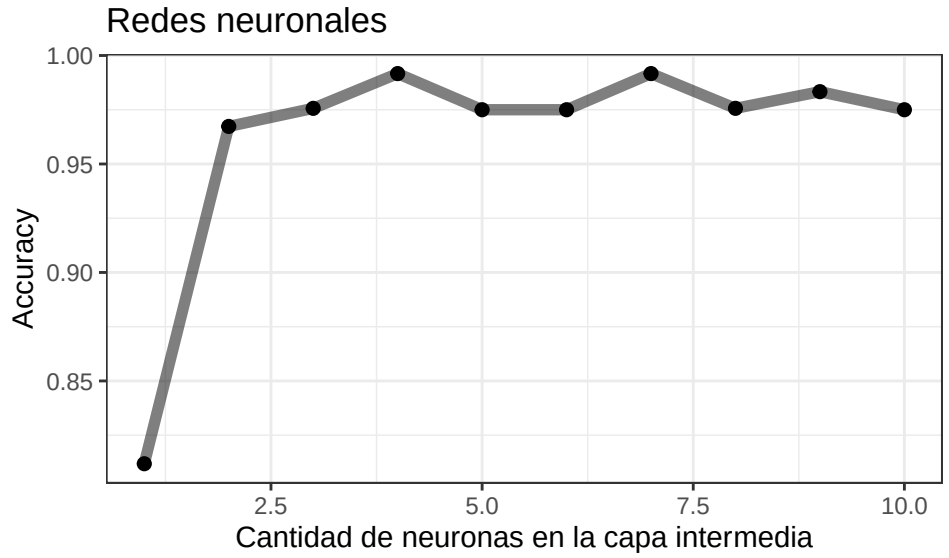

Figure 6: Accuracy en función de la cantidad de neuronas en la capa intermedia.

En la figura 6 observamos los resultados del Accuracy en función de la cantidad de neuronas de la capa intermedia y vemos que el máximo se alcanza utilizando 4 neuronas con un valor de 0.992.

Además, en la figura 7 podemos ver una representación del modelo que elegimos mediante validación cruzada, ajustado con todos los datos del set de entrenamiento.

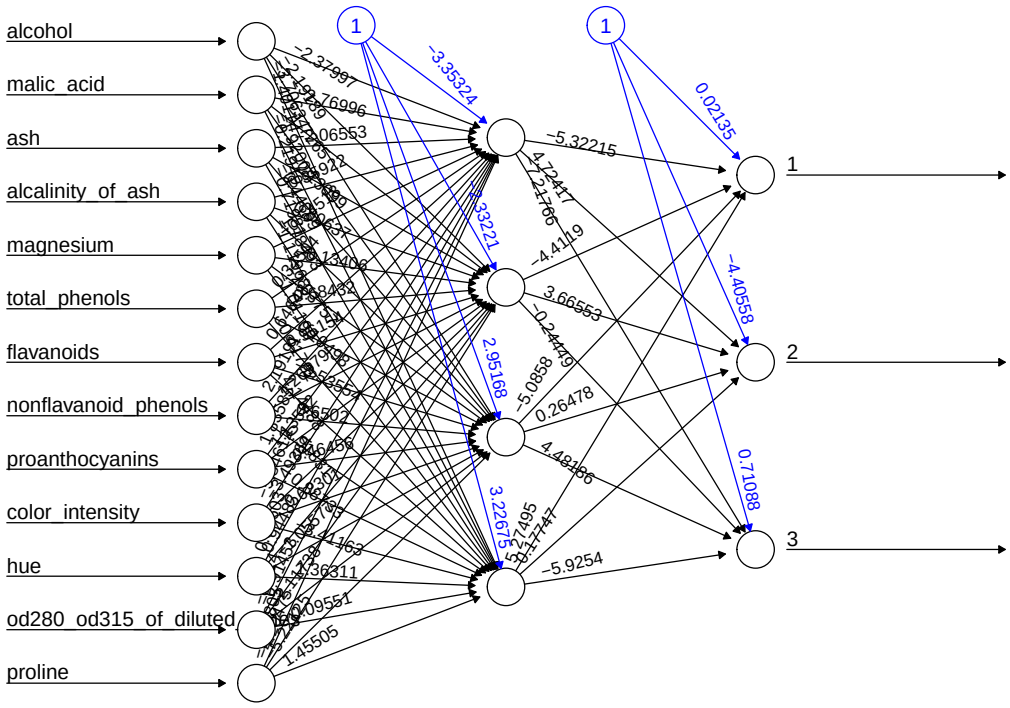

Figure 7: Representación esquemática de la red neuronal ajustada a partir de la selección de parámetros con validación cruzada.

Comparación de los métodos en un conjunto de testeo

A partir del análisis realizado, vemos que los mejores modelos de cada familia estudiada tienen un comportamiento muy similar basándonos en el Accuracy obtenido por validación cruzada.

A continuación vamos a comparar el desempeño de los modelos seleccionados para cada método ajustándolos con todo el set de entrenamiento y evaluándolos en el conjunto que reservamos para testeo. Calculamos el Accuracy en cada caso y los resultados obtenidos son los siguientes:

- 0.967 para el modelo de KNN utilizando 5 vecinos cercanos.
- 0.983 en el modelo seleccionado para Random Forest con m_try igual a 1 y min_n igual a 8.
- 0.983 en el caso del modelo de regresión logística con regularización Lasso con un λ igual a 0.01.
- 0.967 para la red neuronal con una capa que contiene 4 neuronas.

Además, para tener una mayor comprensión de cómo clasifican los diferentes métodos realizamos tablas donde se pueden ver las clases predichas vs. las clases verdaderas.

KNN					**Random Forest**				
		Predichos					Predichos		
		1	2	3			1	2	3
Verdaderos	1	20	0	0	Verdaderos	1	20	0	0
	2	2	22	0		2	1	23	0
	3	0	0	16		3	0	0	16

Regresión logística					**Red neuronal**				
		Predichos					Predichos		
		1	2	3			1	2	3
Verdaderos	1	20	0	0	Verdaderos	1	20	0	0
	2	1	23	0		2	1	23	0
	3	0	0	16		3	0	1	15

Table 1: Tablas de clases predichas vs. clases verdaderas.