Evaluación y refinamiento de modelo

Diego Arturo Padilla Domínguez
A01552594

Introducción

En este reporte, se llevará a cabo la mejora de un modelo, por medio de "manipulación" de los datos con los que trabaja y aprende el modelo.

Primero, hay que saber sobre qué datos se está trabajando. Se tiene una base de datos de vinos, en donde se encuentran las columnas:

- Class
- Alcohol
- Malic acid
- Ash
- Alcalinity of ash
- Magnesium
- Total phenols
- Flavanoids
- Nonflavanoid phenols
- Proanthocyanins
- **Color intensity**
- Hue
- OD280/OD315 of diluted wines
- Proline

El modelo que creare posteriormente buscará predecir la intensidad del color en base a las demás variables.

Creación del modelo

Para realizar la predicción se llevará a cabo una regresión lineal.

El primer modelo que se obtuvo es con los datos sin ninguna modificación, se tomaron todas las columnas para entrenar el modelo y se obtuvieron los siguientes resultados.

Train accuracy: 0.71
Test accuracy: 0.72
Cross Val accuracy: 0.67

Se puede observar que no se obtuvo un modelo demasiado acertado, además de ser demasiado costoso debido a que se toman todas las columnas para entrenar el modelo, por ello vamos a reducir la cantidad de columnas, para saber que columnas eliminar haremos un análisis de correlación de todas nuestras variables dependientes con nuestra independiente.

Con esta lista de datos nos podemos dar cuenta cuales son las columnas que menos afectan a nuestra variable a predecir. Así que, quitaremos las variables con una correlación menor a 0.2 y volveremos a correr el modelo.

Train accuracy: 0.67
Test accuracy: 0.63
Cross Val accuracy: 0.67

Se observa que la confianza del modelo ha empeorado en alrededor de un 5%, aunque esto parezca perjudicial, no lo es, debido a que se redujo en el costo para entrenar el modelo en proporción a la reducción de confianza del modelo.

Ahora que hemos reducido el costo del modelo, comencemos a manipular la base de datos para tener un mayor porcentaje de confianza.

Outlayers (winzorization)

Iniciaremos con la modificación de los outlayers, para ello primero verificaremos que estos existan, así que crearemos gráficos de bigote para cada columna. (Fig. 1)

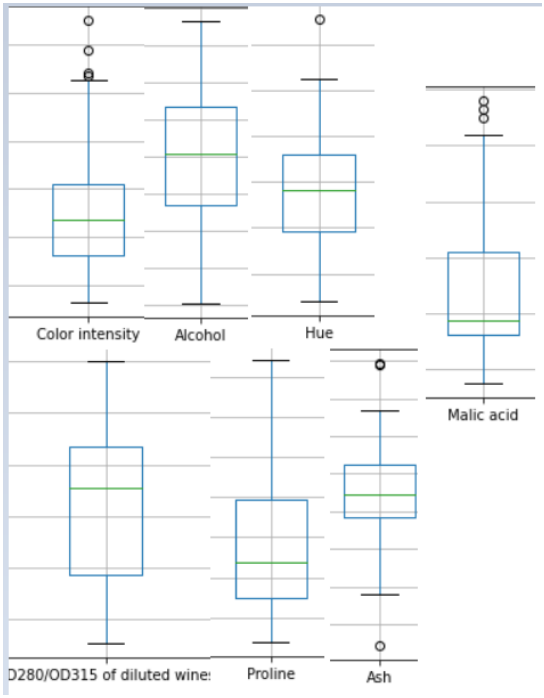

Fig 1 (Gráficos de bigote de las variables pendientes e independientes)

Observamos que muchos de ellos tienen *outlayers*, así que aplicaremos una *winzorization* del 1% para cada uno de ellos, con esto se logran estandarizar más los datos.

Ahora, nuevamente haremos el entrenamiento del modelo.

Train accuracy: 0.71
Test accuracy: 0.75
Cross Val accuracy: 0.64

Hay una mejora, se puede observar que casi se regresan a los valores previos a la reducción de columnas, esto es muy bueno ya que redujimos el costo del modelo y tenemos casi la misma certeza.

Normalización

Se podría terminar aquí la modificación de la base de datos, pero vamos a probar si hay una mejora si es que normalizamos todas nuestras variables independientes.

Train accuracy: 0.71
Test accuracy: 0.75
Cross Val accuracy: 0.64

La mejora del modelo fue nula, en consecuencia, volveremos a la base de datos anterior.

Errores

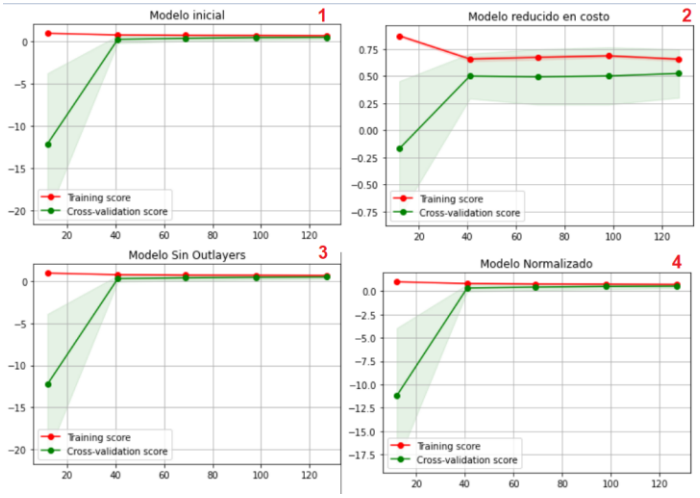

Fig 2 (Gráficos del error conforme a las épocas de entrenamiento entre train y cross validation)

Ninguno de nuestros modelos alcanzo el overfiting, como se puede observar los gráficos de error [Fig. 2], en donde al final de las épocas la diferencia entre los errores de train y cross validation son prácticamente iguales. En el único modelo en el que se nota un mayor sesgo es en 2, pero sigue sin ser muy significativo.

Hablando de la varianza se puede notar que el modelo con una mayor varianza fue aquel en donde se quitaron las columnas, se observa en la variación del “cross validation” en donde sus valores están muy dispersos, esto fue controlado con la

estandarización de los datos en el tercer modelo.

Probablemente intentando otra forma de predicción se pudiesen obtener mejores resultados.

Se podría decir que los modelos 1,3 y 4 tienen un bajo sesgo y baja varianza mientras que el modelo 2 tiene sesgo medio y una varianza alta,

Conclusiones

Podemos concluir que la predicción de la intensidad del color en base a todas las variables con las que contamos no es posible con gran certeza, se logró hacer una optimización de costos, pero no se pudo mejorar su certeza de predicción, Se tendría que hacer un análisis con expertos más a fondo para saber si esto modelo les es útil o no.

Notas:

- Las pruebas están compuestas por el 20% de la base de datos original.
- La parte de validation está hecha por un framework, este framework trabaja con cross validation al cual se le asignarón 10 pliegues.
- El repositorio del código en donde se realizaron estos procesos se encuentra en https://github.com/A01552594/Machine_learning.git