Aprendizaje automático

Práctica 3: Regresión Logística

GAJAN Antoine (894825)

Escuela de
Ingeniería y Arquitectura
Universidad Zaragoza

Índice

Introducción

Como parte de la asignatura .Aprendizaje automático", los estudiantes se familiarizarán con los conceptos de inteligencia artificial. Más concretamente, durante el cuatrimestre se estudiarán las bases del aprendizaje supervisado y no supervisado.

En esta tercera sesión práctica trabajaremos sobre la regresión logística. Esta sesión nos permitirá poner en práctica los conocimientos teóricos aportados por los cursos magistrales y comprender como clasificar los datos.

Esta memoria pondrá de relieve el proceso emprendido para responder a las preguntas, y producirá una reflexión personal sobre los resultados obtenidos.

1. Estudio previo

En un primer momento, utilizando las transparencias del curso, redactamos el algoritmo de K-Fold-Cross-Validation para elegir el valor del parámetro de regularización.

Algorithm 1 Regularizacion(K, X, y)

$best\_errV \leftarrow inf$

$best\_lambda \leftarrow 10^{-5}$

{Para cado modelo posible con lambda}

for $lambda \leftarrow 10^{-5}$ to 10^{5} with a step of x10 **do**

$err\_T \leftarrow 0$

$err\_V \leftarrow 0$

$X\_exp \leftarrow expandir2(x1, x2, grados)$

{Para cada pliegue}

for $i \leftarrow 1$ to K **do**

$[X\_validacion, y\_validacion, X\_entrenamiento, y\_entrenamiento] \leftarrow particion(fold, K, X\_exp, y)$

{Aprendizaje}

$theta \leftarrow LearnerRegularizado(X\_entrenamiento, y\_entrenamiento, lambda)$

$err\_T \leftarrow err\_T + Error(theta)$

$err\_V \leftarrow err\_V + Error(theta)$

end for

{Calculo del error medio}

$err\_T \leftarrow err\_T/K$

$err\_V \leftarrow err\_V/K$

{Verificacion del mejor lambda}

if $err\_V < best\_errV$ **then**

$best\_errV \leftarrow err\_V$

$best\_lambda \leftarrow lambda$

end if

end for

{Aprendizaje con todos los datos y el mejor modelo}

$theta \leftarrow LearnerRegularizado(theta, X\_exp, y, best\_lambda)$

2. Memoria de la práctica

2.1. Cuestión 1 : Regresión logística básica

2.1.1. Dibujo de la superficie de separación

En esta pregunta, se desea determinar los vinos de clase 1 en función de los atributos 6 y 10, utilizando la regresión logística básica. Para ello, comenzamos entrenando el modelo de manera clásica con las funciones presentadas en clase magistral.

Obtenemos los siguientes valores para theta: theta = [-6.6739 9.5578 3.4811]. En otras palabras, tenemos y = -6.6739 + 9.5578 * x6 + 3.4811 * x10.

Se obtiene así la siguiente representación de la frontera en función de los atributos x6 y x10.

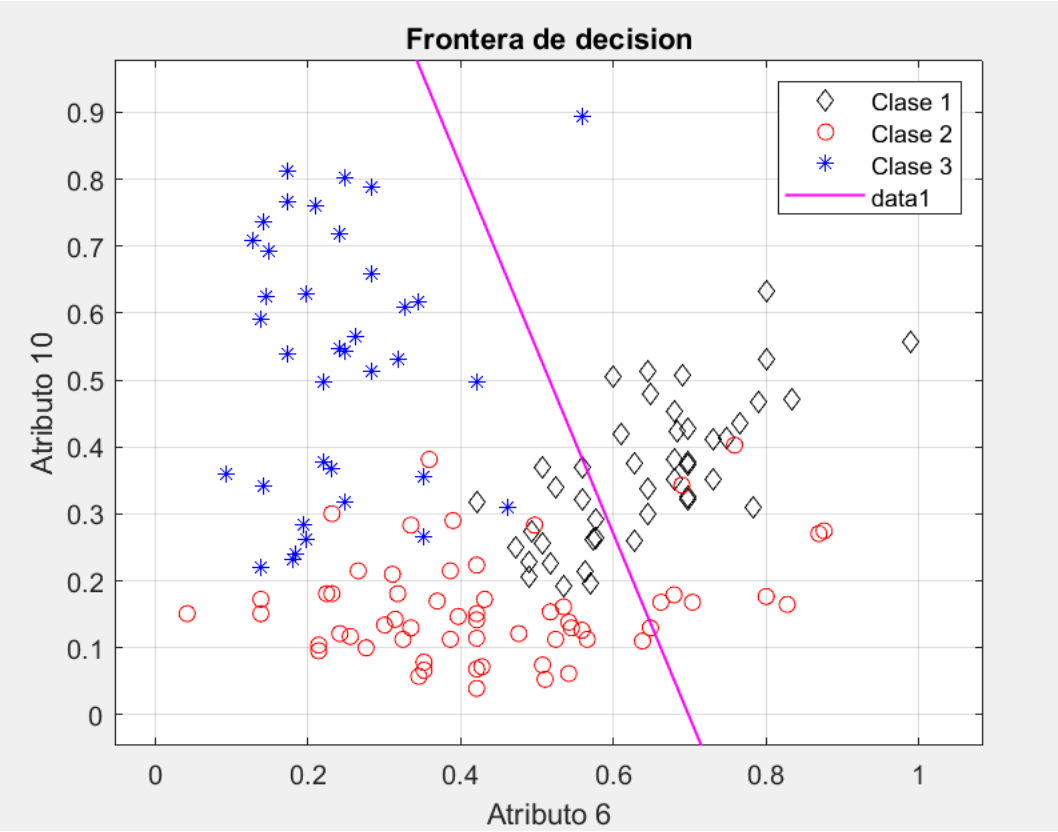

Figura 1: Superficie de separación con la regresión básica

Se observa directamente que el modelo propuesto es demasiado simple, por lo que nos encontramos en un caso de ajuste insuficiente. De hecho, la frontera de decisión es modelada por una simple recta lineal. Esto tiene como consecuencia que el modelo tendrá una tendencia a predecir una pertenencia a la clase 1 errónea muy a menudo.

2.1.2. Métrica adecuada

Para demostrarlo científicamente, podemos analizar una métrica adecuada relacionada con el problema. Dado que el problema es un problema equilibrado (ninguna clase está subrepresentada o sobrerepresentada), podemos calcular la tasa de acierto. Este se define como la tasa de valores correctamente predichos en relación con su clase real. Por lo tanto, obtenemos los siguientes resultados:

	Tasa de acierto
Datos de entrenamiento	0.8099
Datos de test	0.8611

Cuadro 1: Tasa de acierto con la regresión logística básica

Dado que la tasa de acierto es mejor con los datos de test que con los datos de entrenamiento, podemos deducir que estamos en presencia de un subajuste. En otras palabras, el modelo propuesto es demasiado simple. La tasa de acierto, del orden del 80 %, puede parecer elevada. Sin embargo, la diferencia significativa entre los datos de entrenamiento y de test, así como la frontera dibujada, nos permite darnos cuenta del subajuste.

También podrían haberse tenido en cuenta otras métricas, como la precisión y el recall. Sin embargo, como vamos a examinarlos en detalle en la pregunta 3, he preferido la tasa de acierto, más intuitivo, que podrá compararse con los resultados obtenidos en la pregunta 3.

2.1.3. Predicción de la clase de un vino

Ahora queremos saber la probabilidad de pertenecer a la clase 1 en función del valor del atributo x10 sabiendo que x6 = 0.6. Para ello, vamos a generar un conjunto de valores posibles x10, luego vamos a calcular la salida prevista (probabilidad de pertenecer a la clase 1) para estos diferentes valores. Obtenemos el siguiente gráfico:

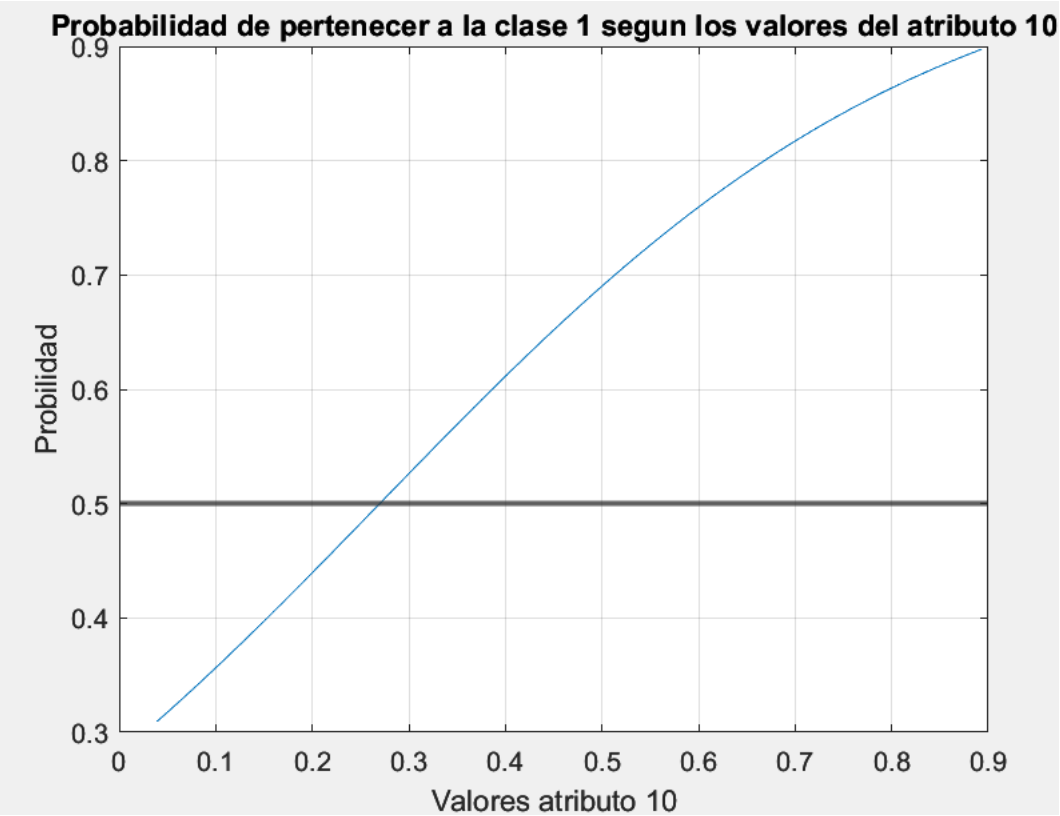

Figura 2: Probabilidad de pertenecer a la clase 1 según el atributo x10 con la regresión logística básica

Con un umbral de 0.5, se observa que para cualquier valor de x10 superior a aproximadamente 0.25, el vino pertenece a la clase 1. Teniendo en cuenta los datos que tenemos, esto no parece coherente. Esto se debe al hecho de que nuestro modelo es demasiado básico y la frontera es una simple recta.

2.2. Cuestión 2 : Regularización

Queremos detectar los vinos de la clase 1, a partir de los atributos 6 y 10, haciendo expansión de atributos de grado 6, utilizando la función expandir2.m proporcionada.

Esta parte con la regularización es muy similar a la de la práctica anterior. Utilizaremos de nuevo el algoritmo K fold cross validation. Solo hay 2 diferencias a tener en cuenta:

- la función de coste es una función de coste logístico regularizada
- la salida prevista se calcula ahora mediante una función sigmoide

2.2.1. Elección del parámetro de regularización lambda

Primero queremos elegir el mejor lambda. Considerando la expansión polinómica de grado 6 para los atributos x6 y x10, obtenemos los siguientes resultados.

En primer lugar, modelizamos el coste logístico regularizado en función de los diferentes valores de lambda a probar (entre 10^{-6} y 10^{0}).

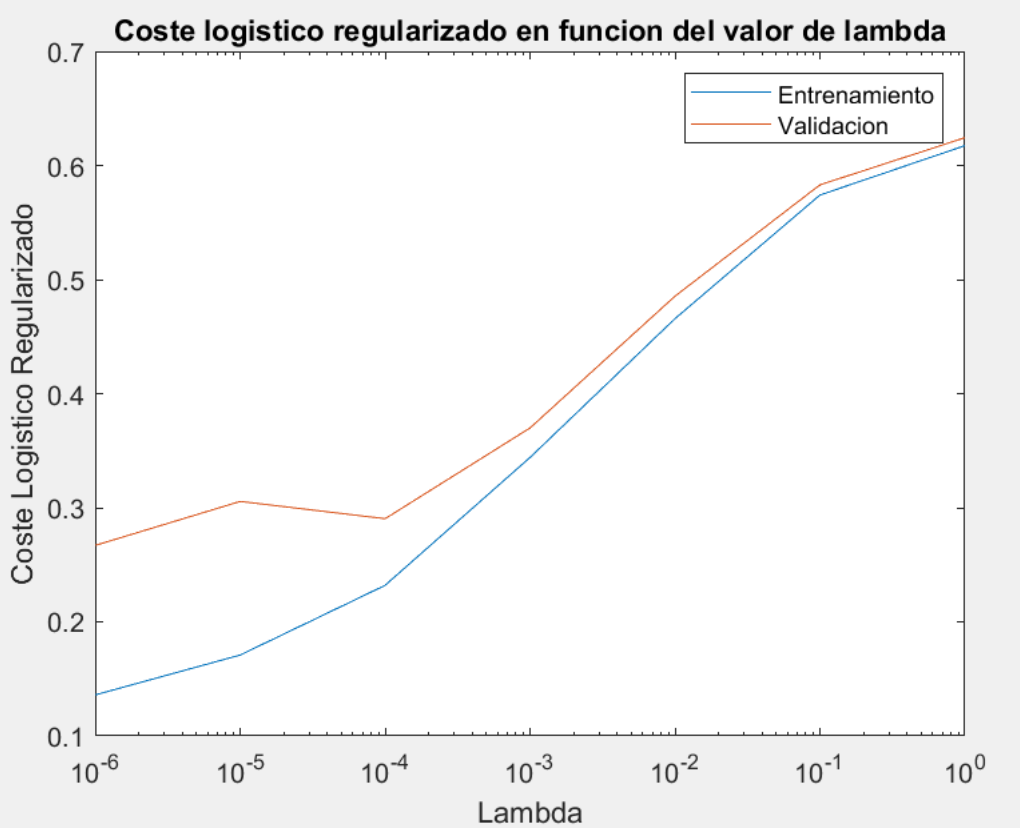

Figura 3: Evolución del coste logístico en función de lambda con la regularización

Se observa que el error de validación es el más bajo cuando lambda = 10^{-6}. Este valor de lambda es el valor que elegiremos para establecer el mejor modelo. En efecto, si se toma un valor más pequeño o más grande, se podría encontrar en el caso del subajuste o del sobreajuste.

Con el fin de amplificar sus resultados, también modelamos la tasa de acierto en función de lambda. Esto nos permite sacar las mismas conclusiones que antes.

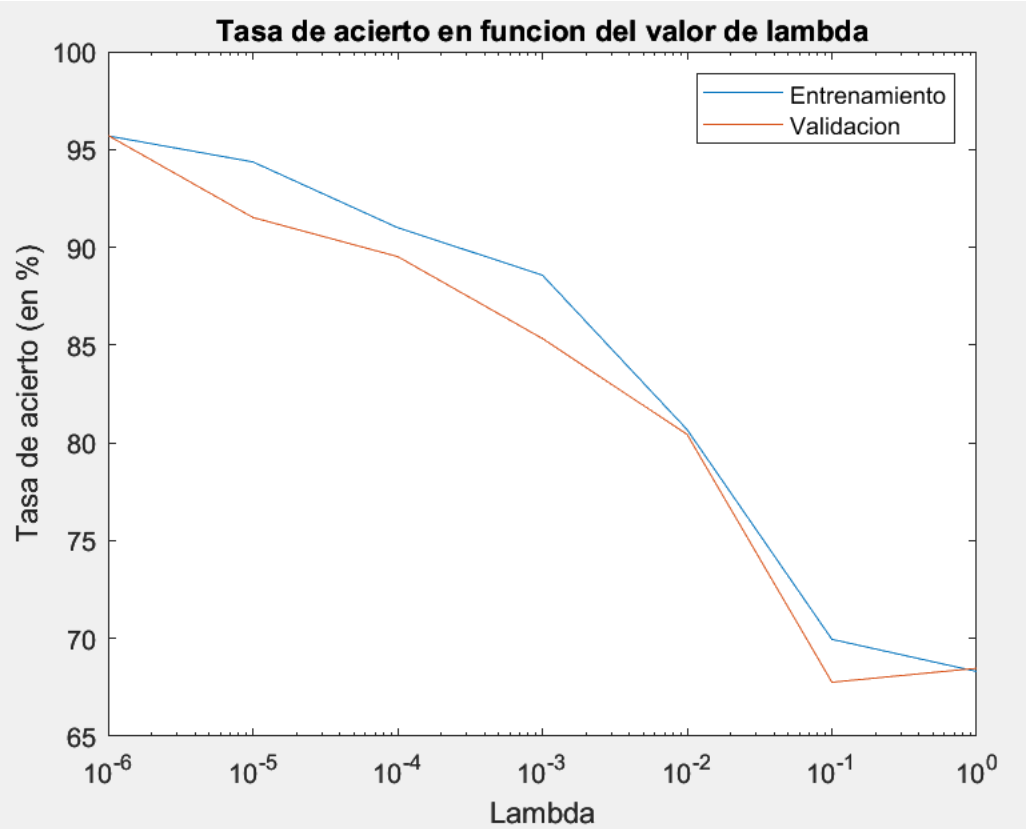

Figura 4: Evolución de la tasa de acierto en función de lambda

La tasa de acierto es de aproximadamente el 96 % cuando se elige sabiamente lambda $= 10^{-6}$.

Cuando se aumenta lambda, penalizamos más fuertemente la complejidad del modelo. Por lo tanto, las predicciones son de menor calidad y las tasas de acierto son menores.

2.2.2. Dibujo de las superficies de decisión

Ahora vamos a entrenar de nuevo el mejor modelo (lambda $= 10^{-6}$) y el modelo para el cual lambda $= 0$. Así podremos dibujar los límites de decisión y comparar para diferentes valores de lambda las modificaciones que esto implica.

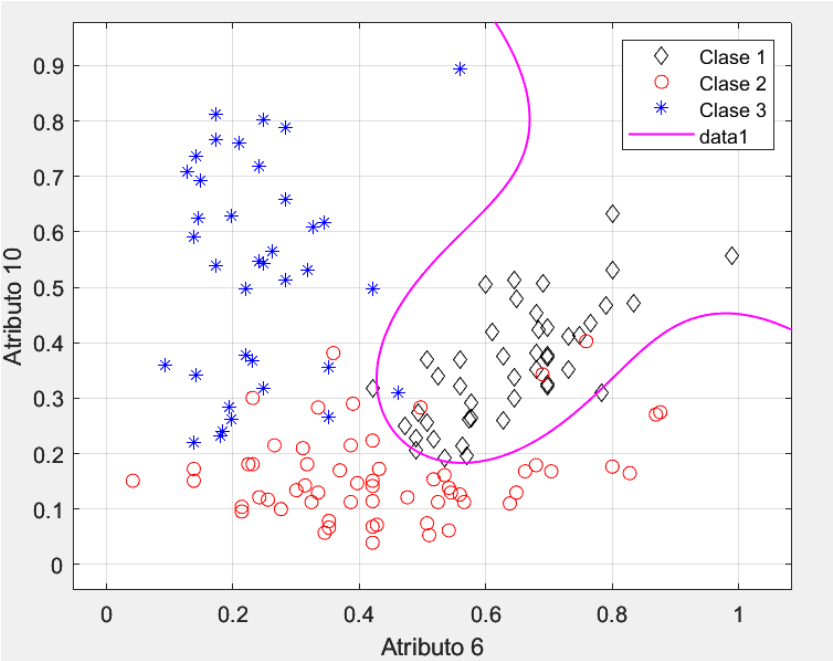

Figura 5: Frontera de decisión con el mejor lambda

Con el mejor lambda, el modelo parece perfectamente ajustado. En efecto, a pesar de los pequeños errores debidos a los espurios, el modelo predice correctamente en la mayoría de los casos y la forma de la frontera no deja presagiar un sobreajusto o un subajusto.

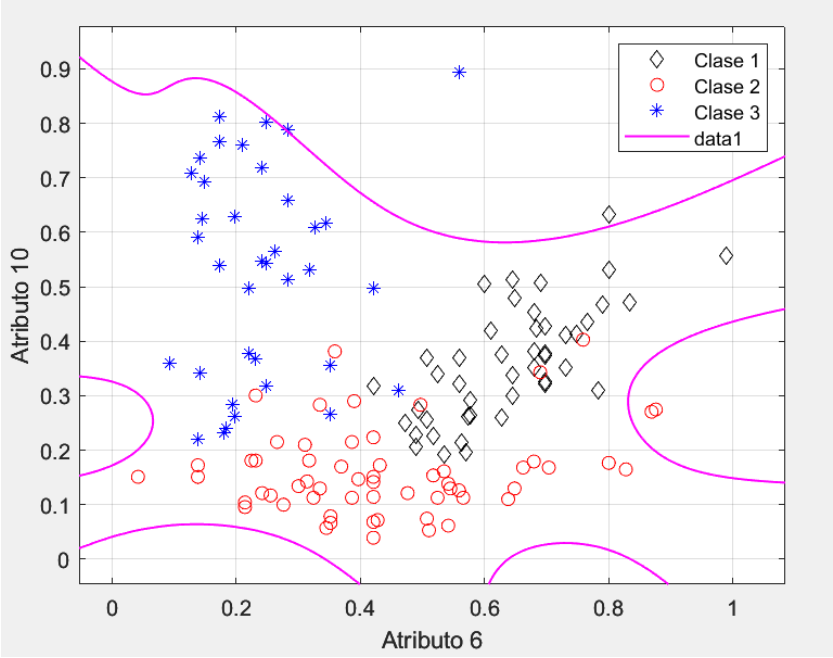

Figura 6: Frontera de decisión con lambda = 0

Con lambda = 0, las constataciones son diferentes. En efecto, a falta de regulari-

zación, se observa claramente que la forma de la superficie de decisión no corresponde a la prevista.

Esto puede traducirse en la métrica elegida, es decir, la tasa de acierto:

	lambda $= 10^{-6}$	lambda $= 0$
Datos de entrenamiento	0.9577	0.3380
Datos de test	0.9444	0.4167

Cuadro 2: Tasa de acierto con la regularización

La consideración de la métrica pone de relieve las constataciones anteriores. Con un bajo lambda, hemos favorecido la complejidad del modelo, sin favorecer el sobreajusto (ya que los errores con los datos de entrenamiento y test son similares). Con un lambda nulo, no practicamos la regularización. Esto tuvo como consecuencia que el modelo no se adaptara a nuestra necesidad.

2.2.3. Predicción de la clase de un vino

Para un vino que tiene atributo 6 igual a 0.6,vamos a dibujar una gráfica con la probabilidad de que sea de clase 1 en función del valor del atributo 10.

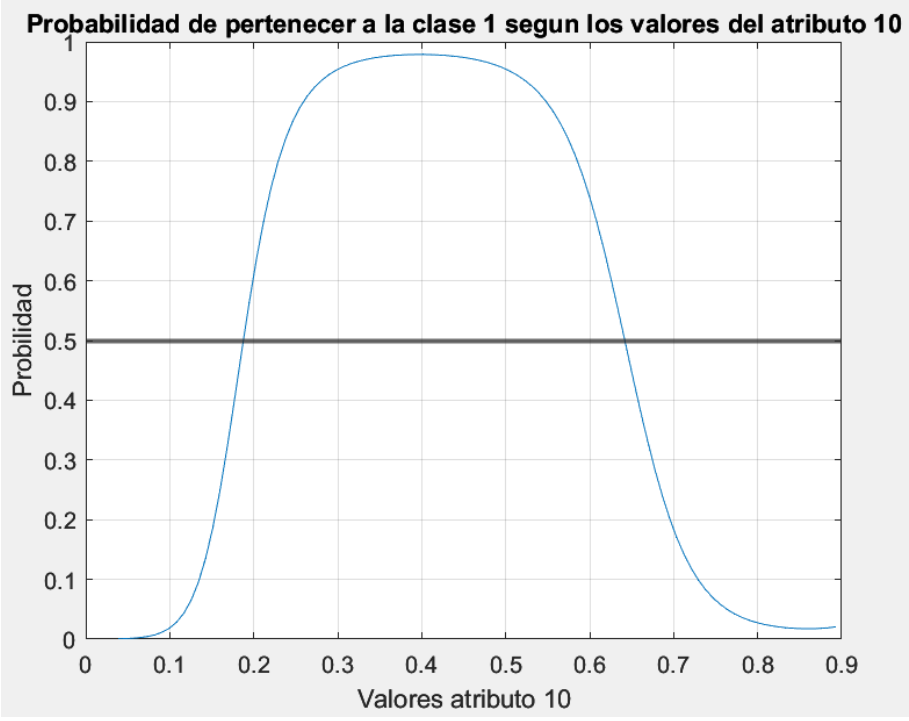

Figura 7: Probabilidad de pertenecer a la clase 1 según el atributo x10 con el modelo de la regularizacion

Observamos que para un valor del atributo x10 entre 0.18 y 0.65, el modelo con lambda = 10^{-6} nos predice que pertenece a la clase 1. Esto es realista en vista de los datos que tenemos, y parece mucho más coherente que el resultado obtenido en la pregunta 1.

2.3. Cuestión 3 : Curvas Precisión/Recall

Utilizando los datos de test, trazaremos en una misma figura las curvas de precisión/recall de los diferentes modelos estudiados en esta práctica.

Para ello, utilizaremos las siguientes fórmulas:

```
TP = sum(y_pred == 1 & y == 1)
FP = sum(y_pred == 1 & y == 0)
FN = sum(y_pred == 0 & y == 1)
TN = sum(y_pred == y & y == 0)

precision = TP / (FP + TP)
recall = TP / (TP + FN)
```

Hemos obtenido los siguientes resultados:

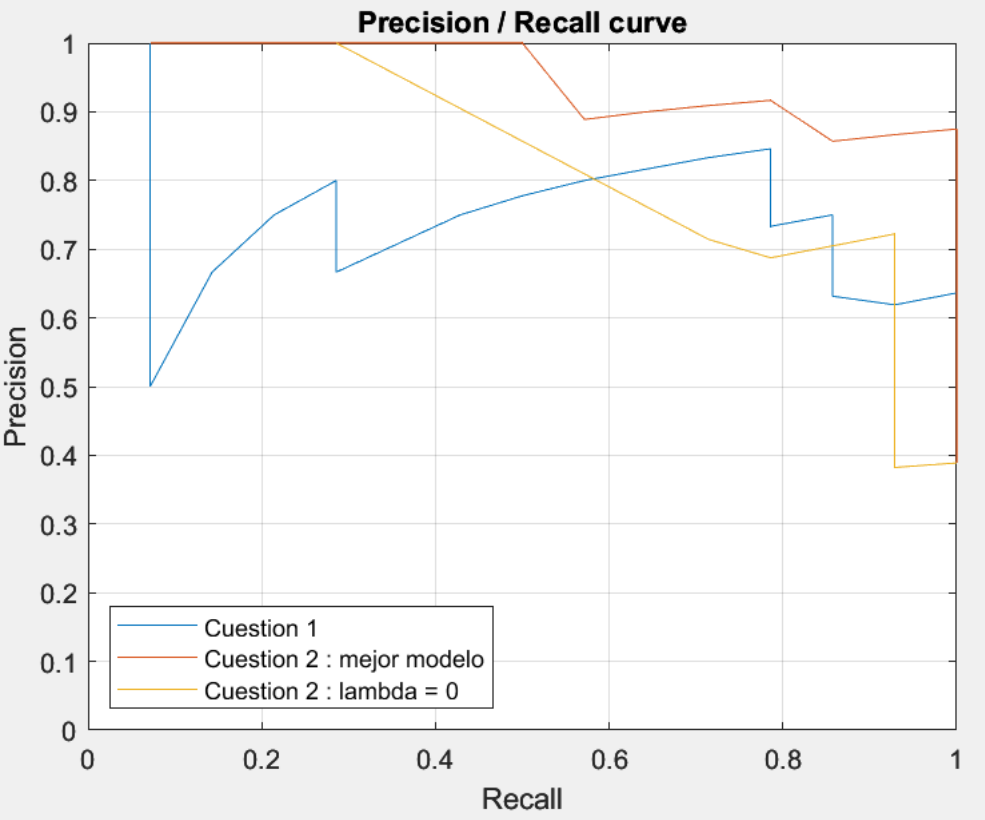

Figura 8: Curvas de precisión/recall de los diferentes modelos entrenados

Según los conocimientos del curso, el mejor modelo, aquel hacia el que se desea

converger en un mundo ideal, sería aquel donde precision = 1 y recall = 1. En otras palabras, lo ideal sería tener una curva que se acerque lo más posible al punto de coordenadas (1,1). En efecto, esto significaría que la precisión (entre los vinos detectados pertenecientes a la clase 1, cuánto lo son realmente) sería óptima y que el recall (entre los vinos de la clase 1, que se han predicho correctamente) lo sería también. Según el caso estudiado, hay que hacer un compromiso entre la optimización de la precisión y la optimización del recall. El modelo que se acerca lo más posible a nuestro deseo es el que tiene lambda = 10^{-6}. En efecto, en cualquier punto del gráfico, se observa que no importa el recall, la precisión es siempre de modelo con lambda = 10^{-6} es siempre superior a la de los otros modelos. Si se fija una precisión entre 0 y 1, también se observa que el recall del modelo con lambda = 10^{-6} es siempre superior al de los otros modelos.

Si queremos garantizar que el 90 % de los vinos clasificados como clase 1 lo son realmente, eso significa en términos de precision y recall que deseamos tener precision = 0.9.

Con el fin de alcanzar este objetivo de precisión, el modelo que vamos a tomar es el que tiene lambda = 10^{-6} como se explicó anteriormente. Ahora vamos a ver qué umbral aplicar a nuestra función de predicción nos permite obtener una precisión superior a 0.9. Para ello, vamos a trazar la precisión en función del umbral.

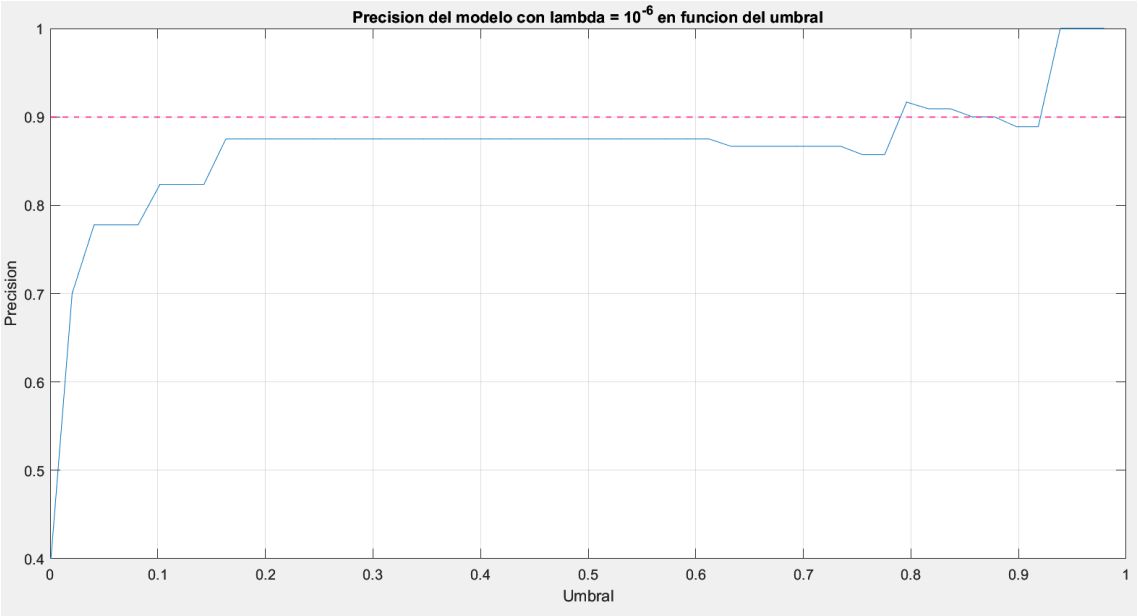

Figura 9: Precisión del modelo con lambda = 10^{-6} en función del umbral

Si nos acercamos a la parte más interesante (donde la precisión se acerca a 0.9, obtenemos lo siguiente:

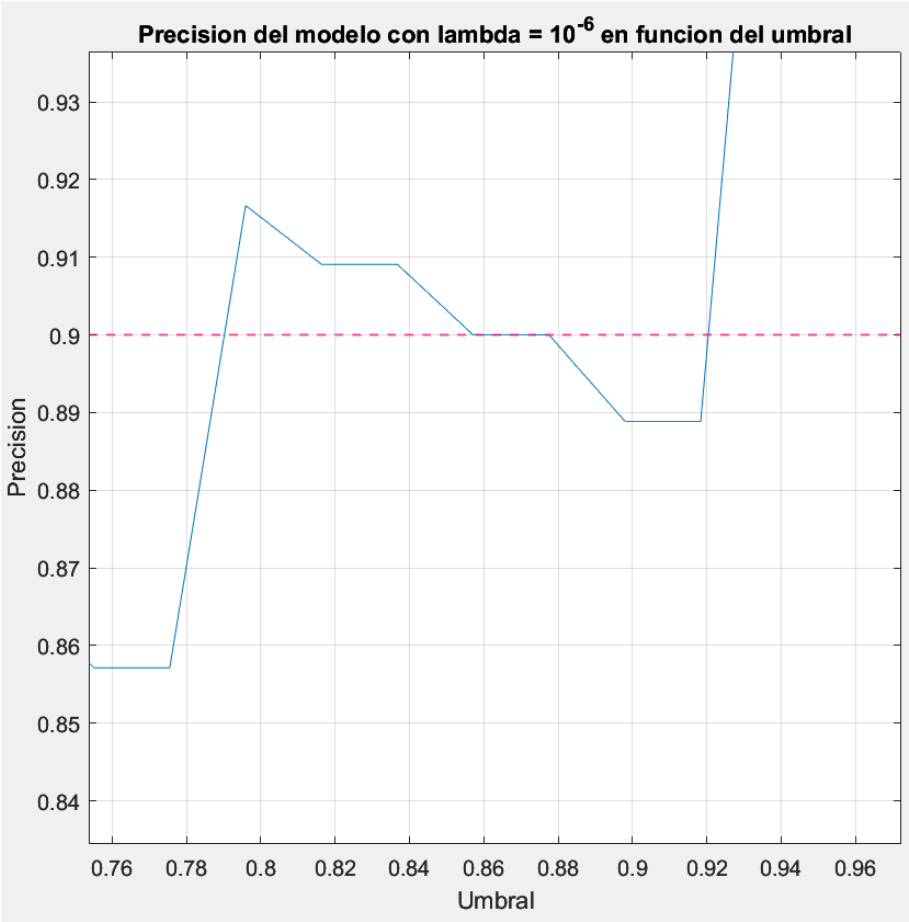

Figura 10: Zoom de la parte interesante de la precisión del modelo

Con este gráfico podemos ver que si queremos alcanzar el objetivo de precisión = 0.9, necesitamos un umbral para nuestra función de predicción que esté entre 0.79 y 0.88 o que el umbral sea superior a 0.92.

Conclusión

Esta práctica nos ha permitido ampliar nuestro conocimiento de las prácticas anteriores a la regresión logística. Esto nos permitió comprender las diferentes opciones de modelo para encontrar la mejor solución a un problema de clasificación.

Durante esta práctica pudimos poner en práctica las funciones sigmoides y de coste logístico, sin dejar de ver los beneficios del algoritmo K Fold Cross Validation y la regularización.

Finalmente, pudimos entender la diferencia entre la precisión y el recall y ver cómo podemos diseñar un modelo para alcanzar un determinado objetivo en términos de precisión y recall.

Índice de figuras

Índice de cuadros