Práctica - Aprendizaje supervisado

Javier Pérez Vargas - d21c017

December 12, 2023

Contents

1 Introducción

El aprendizaje automático supervisado ha emergido como una herramienta fundamental en la era moderna de la ciencia de datos, permitiendo la construcción de modelos predictivos a partir de datos etiquetados. En este contexto, este estudio se enfoca en la aplicación de técnicas de aprendizaje automático supervisado para resolver un problema de clasificación, explorando diferentes algoritmos tales como regresión logística, árboles de decisión, K-NN y SVMs.

2 Descripción del problema

El dataset con el que vamos a trabajar contiene información sobre la satisfacción de pasajeros con distintas aerolíneas, basada en las respuestas de unas encuestas realizadas. El conjunto original contiene 24 features (numéricas, binarias y categóricas) y +100.000 instancias. Nosotros solo usaremos un subconjunto de ellas.

Algunas de las features del dataset son:

- Edad
- Tipo de viaje: Objetivo del viaje (personal o de negocios)
- Clase: Clase en la que iba el pasajero (Business, Eco, Eco Plus)
- Distancia del vuelo
- Otras muchas variables en las que el viajero puntúa del 0-5 su satisfacción con respecto a otros aspectos (limpieza, facilidad para reserva oline, servicio wi-fi en el vuelo...)

Por ejemplo, dos filas del dataset serían las siguientes:

Gender	**Customer Type**	**Age**	**Type of Travel**	**...**	**Flight Distance**	**Satisfaction**
Male	Loyal Customer	41	Business	...	303	satisfied
Female	disloyal Customer	36	Business	...	275	neutral or dissatisfied

Table 1: Ejemplo de datos

La columna que queremos predecir para nuevos datos será la de 'Satisfaction', y es una columna binaria: **satisfied** o **neutral or dissatisfied**. Por lo tanto, tenemos un problema de clasificación binaria que resolveremos con los métodos ya mencionados.

3 Metodología

Antes de comenzar a ver método por método, vamos a hacer un análisis previo sobre las variables. Ya hemos visto que varias de ellas son categóricas, así que vamos a aplicar One-Hot Encoder a estas. Además, utilizamos StandardScaler sobre las variables numéricas (excepto para el caso de los árboles de decisión, que usaremos los datos originales para que la interpretabilidad sea más sencilla).

Tenemos también que separar los datos en 'train' y 'test'. Usaremos un 20% de datos para el 'test' (1000 filas) y el resto para 'train' (4000 filas). Por lo tanto, nuestro conjunto de entrenamiento utilizará un total de 4000 instancias para entrenar a los modelos, una cifra considerable que seguramente nos permita predecir con bastante precisión la satisfacción de cada viajero, aunque a priori no sabemos si las variables serán capaces de poder explicarlo.

Para verificar lo anterior, vamos a calcular las correlaciones entre la variable respuesta 'Satisfaction' y las variables explicativas.

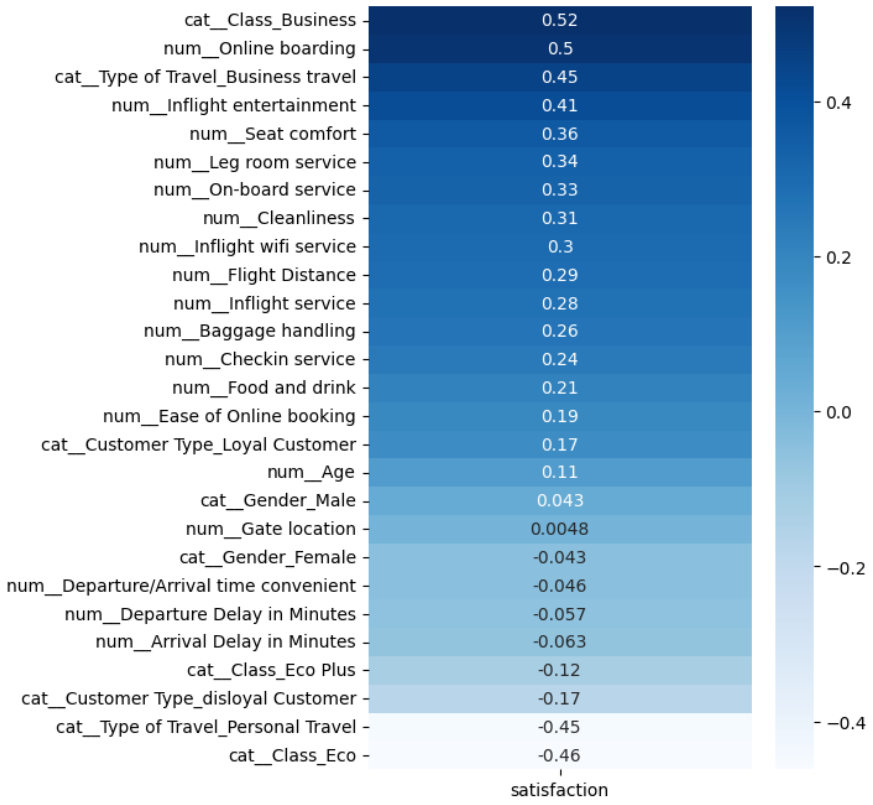


Figure 1: Correlaciones con la columna 'Satisfaction'

En este gráfico podemos ver las diferentes correlaciones con nuestra variable respuesta. Una correlación alta (en valor absoluto) indicará que la variable está muy relacionada con 'Satisfaction'.

Las correlaciones más altas con nuestra variable respuesta son 'Online Boarding' (Nivel de satisfacción con la reserva del vuelo), 'Class Business' (La clase del viajero es business) y 'Personal Travel' (El objetivo del viaje es personal). Por tanto, presumiblemente estas serán las variables que mejor separarán los datos.

También observamos que variables como 'Gender', 'Gate Location' o 'Age' son las menos correlacionadas, lo cual parece tener sentido dado que su relación con la experiencia general de vuelo o la satisfacción del pasajero es menos directa que otras variables.

3.1 Árboles de decisión

Los árboles de decisión son modelos predictivos que utilizan una estructura jerárquica de nodos para tomar decisiones, dividiendo iterativamente los datos según características para llegar a una predicción o clasificación final. El "root node" es la separación principal de los datos, mientras que las "leaf node" o nodos hoja son nodos terminales en el árbol, que representan una predicción o clasificación final para un conjunto de datos después de haber atravesado el árbol.

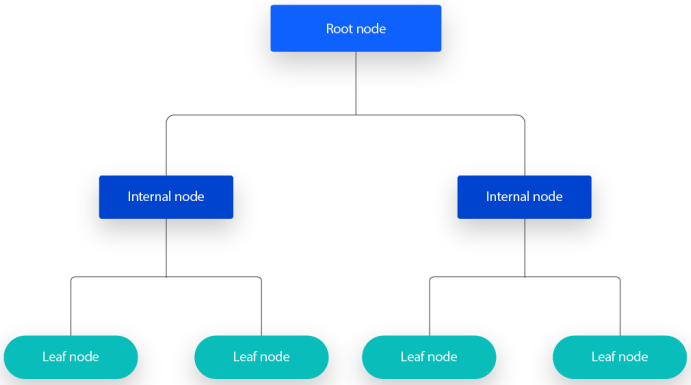


Figure 2: Ejemplo de árbol de decisión

Los parámetros a especificar cuando creamos un árbol de decisión son:

- **Maximum depth**: Profundidad máxima del árbol de decisión.
- **Minimum samples split**: Nº mínimo de muestras necesarias para dividir un nodo.
- **Minimum samples leaf**: Nº mínimo de muestras permitidas en un nodo hoja.
- **Criterio**: Determina la medida de calidad de una división.

Hemos decidido aplicar cross-validation con $k = 10$. Esto implica dividir el conjunto de datos en 10 partes iguales, utilizando 9 partes para entrenar el modelo y reservando 1 parte para validar. Este proceso se repite 10 veces, utilizando cada una de las partes como conjunto de validación mientras el resto actúa como datos de entrenamiento. Al final, se promedian los resultados de las 10 iteraciones para obtener una estimación más robusta del rendimiento del modelo en todo el conjunto de datos.

Aplicado a nuestros datos y optimizando la "accuracy", hemos obtenido los siguientes parámetros:

Parámetros	`max_depth`	`min_samples_leaf`	`min_samples_split`	`criterion`
Valores	8	2	2	`gini`

Así que este serán los parámetros que utilizaremos para nuestro modelo.

3.2 KNN Neighbours

La técnica de KNN clasifica los datos según la mayoría de los vecinos, asignando una etiqueta a un punto de datos basada en las etiquetas de los K puntos más cercanos.

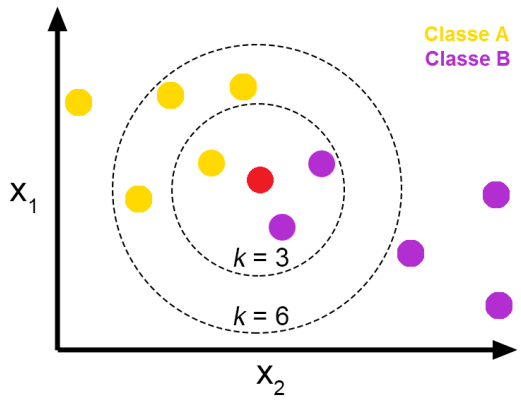


Figure 3: Ejemplo de KNN

Los únicos parámetros a estimar aquí es el número K de vecinos a considerar y los pesos de los vecinos. Haciendo validación cruzada (optimizando "accuracy") obtenemos:

Parámetros	`k`	`Weights`
Valores	6	`distance`

Así, obtenemos que el parámetro más estable es $k = 6$ y utilizar la distancia a los vecinos como peso.

3.3 Regresión Logística

Se utiliza para predecir la probabilidad de una variable categórica binaria basada en variables independientes, utilizando una función logística para estimar la relación entre las características y la probabilidad de pertenecer a una clase específica.

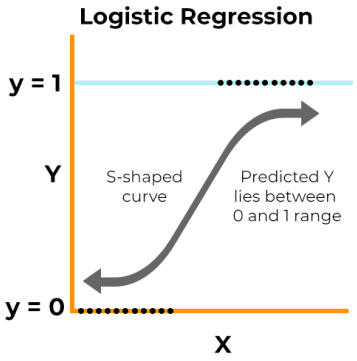


Figure 4: Regresión logística

De nuevo, usamos validación cruzada con los siguientes parámetros:

- **C**: Parámetro de regularización inversa; controla la fuerza de regularización en el modelo.
- **penalty**: Tipo de regularización aplicada al modelo (L1 o L2).

Los resultados obtenidos (optimizando "accuracy") son los siguientes:

Parámetros	`C`	`penalty`
Valores	0.1	`L1`

3.4 Support Vector Machine

Es un modelo de aprendizaje supervisado que encuentra un límite de decisión óptimo entre clases al encontrar el hiperplano que maximiza el margen entre datos de distintas clases.

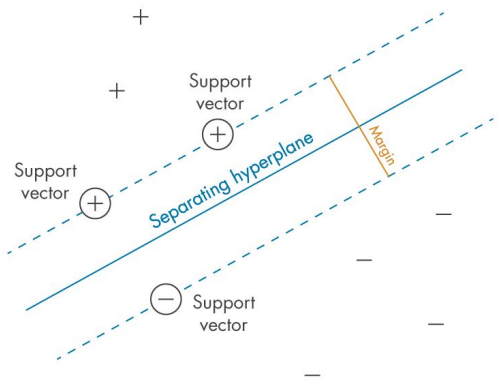


Figure 5: Support Vector Machine

Los parámetros que calcularemos son los siguientes:

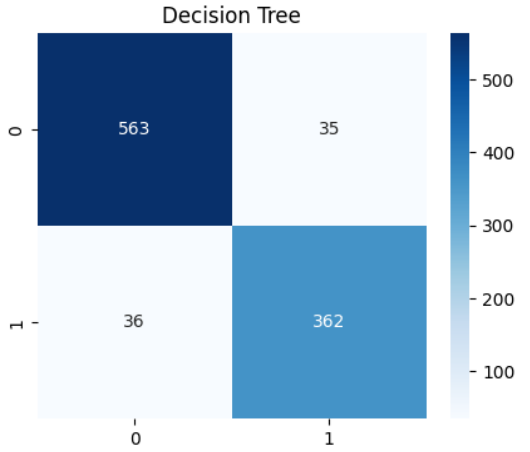
- **C**: Parámetro de regularización. Controla el equilibrio entre maximizar el margen y minimizar la clasificación incorrecta.
- **kernel**: Define el tipo de kernel a utilizar en el SVM. El kernel determina la forma en que se calcula la similitud entre las muestras.
- **gamma**: Define cuánta influencia tiene un solo ejemplo de entrenamiento.

Parámetros	`C`	`kernel`	`gamma`
Valores	10.0	`rbf`	`auto`

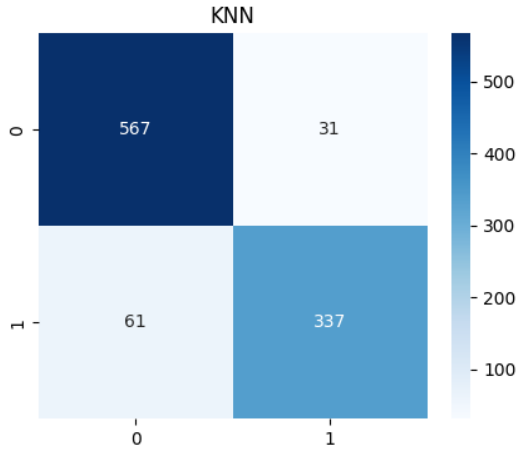
Vemos que el mejor kernel obtenido es el 'rbf', es decir, el kernel Radius Basis Function. Es un tipo de kernel adecuado que incluso permite crear fronteras de decisión no lineales, lo cual puede ser útil en nuestro problema.

Una vez que tenemos los mejores parámetros para cada modelo, vamos a crearlos y ver los resultados.

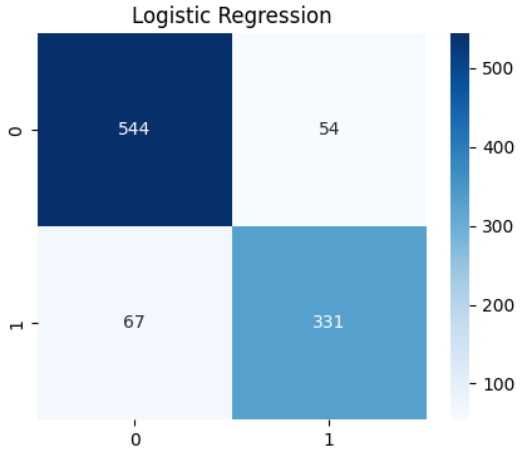
4 Resultados



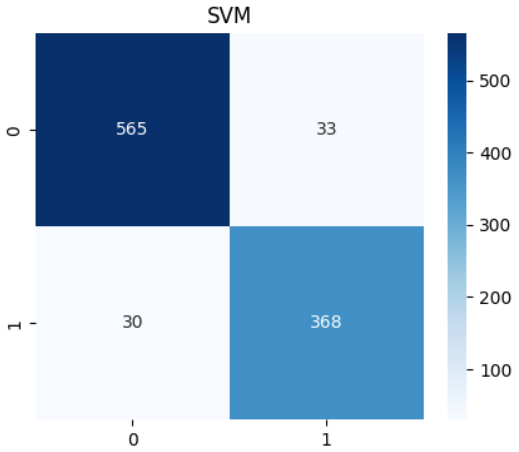
(a) Matriz de confusión para el árbol de decisión



(b) Matriz de confusión para KNN



(c) Matriz de confusión para regresión logística



(d) Matriz de confusión para SVM

Estas son las matrices de confusión de cada método. Vamos a compararlos también con las métricas:

Model	Accuracy	Precision	Recall	F1
Decision Tree	0.926707	0.923207	0.924258	0.923723
KNN	0.907631	0.909314	0.897447	0.902427
Logistic Regression	0.878514	0.875042	0.870679	0.872692
SVM	0.936747	0.933643	0.934720	0.934172

Table 2: Métricas de evaluación de los modelos

Además, las principales reglas generadas por el árbol de decisión son las siguientes:

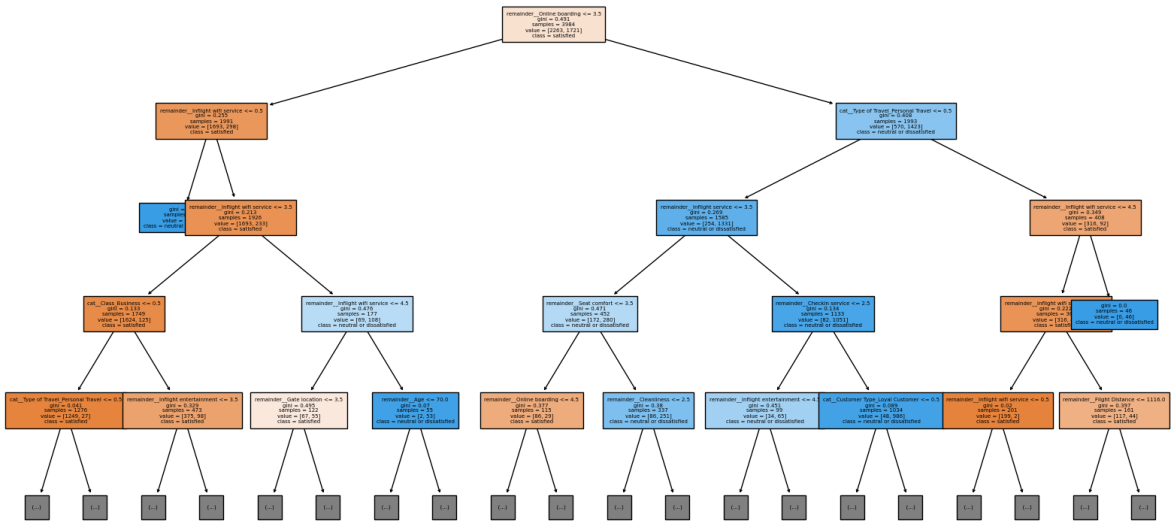


Figure 7: Reglas del árbol de decisión

5 Discusión

5.1 Resultados y mejoras

Las matrices de confusión de los 4 modelos son muy parecidas. Observando las métricas, vemos que el modelo de Support Vector Machine (SVM) tiene el mejor desempeño general en términos de precisión, recall, F1-score y accuracy, con valores superiores a los otros modelos evaluados. Con una accuracy del 93.67%, SVM logra clasificar con mayor precisión las muestras, seguido por el árbol de decisión con un 92.67%. No obstante, los demás modelos también muestran un desempeño sólido, teniendo una tasa de aciertos cercana al 90%. El modelo que peor funciona con nuestros datos es el de Regresión Logística, posiblemente debido a que este método no suele trabajar bien con relaciones no lineales, al contrario que SVM, que permite crear fronteras de decisión no lineales.

Una idea para mejorar los modelos sería aumentar el tamaño de la muestra. Hemos usado 5000 datos pero podríamos haber usado una mayor cantidad pues la muestra original contenía más de 100.000. Podríamos aumentar también la complejidad del modelo ampliando el conjunto posible de parámetros a elegir, aunque esto implicaría un tiempo mayor en la etapa de entrenamiento y posiblemente el cambio tampoco sería muy significativo. Como última opción, he valorado utilizar otra métrica a optimizar durante el entrenamiento, como por ejemplo la 'Precision' y el 'Recall'. No obstante, eso podría sobreajustar el modelo y es lo que queremos evitar, así que he decidido mantener la 'Accuracy' como métrica a optimizar.

5.2 Reglas del árbol de decisión

El árbol de decisión, recordemos, tenía una profundidad máxima de 8. En la imagen del árbol hemos limitándola profundidad del árbol a 4 para que los nodos sean legibles. Así, vamos a analizar las ramas que nos llevan a nodos con un valor de 'gini' (medida de impureza) bajo, es decir, cuando los nodos sean más homogéneos o "puros".

- Online Boarding ≤ 3.5 **AND** Inflight Wifi Service ≥ 0.5 **AND** Inflight Wifi Service ≤ 3.5 **AND** Class Business ≤ 0.5 **AND** Type of Travel: Personal Travel $\leq 0.5 \longrightarrow$ **Satisfied**
- Online Boarding ≤ 3.5 **AND** Inflight Wifi Service ≥ 0.5 **AND** Inflight Wifi Service ≥ 3.5 **AND** Inflight Wifi Service ≥ 4.5 AND Age $\leq 70 \longrightarrow$ **Neutral or Dissatisfied**

- Online Boarding ≥ 3.5 **AND** Type of Travel: Personal Travel ≤ 0.5 **AND** Inflight Service ≥ 3.5 **AND** Check-in Service ≥ 4.5 AND Type of Customer: Loyal Customer $\leq 70 \longrightarrow$ **Satisfied**

En lenguaje natural, las conclusiones que podemos sacar son las siguientes:

- Cuando el proceso de embarque en línea es inferior o igual a 3.5 y el servicio de wifi a bordo es satisfactorio pero no excelente (entre 0.5 y 3.5) y el cliente viaja en clase Business y pertenece al segmento de viajeros por negocio, es más probable que estén satisfechos.
- Si el proceso de embarque en línea es menor o igual a 3.5 y el servicio de wifi a bordo es muy alto (más de 4.5) y el cliente tiene menos de 70 años, es más probable que estén en un estado neutral o insatisfecho.
- Cuando el proceso de embarque en línea es mayor o igual a 3.5 y el cliente no es un viajero personal y el servicio a bordo es bueno, especialmente cuando el servicio a bordo y el servicio de check-in son de alta calidad y el cliente es leal, es probable que estén satisfechos.

6 Conclusiones

Podríamos decir que el mejor modelo para nuestros datos es el SVM. Su mayor complejidad con respecto a los otros 3 modelos lo hace capaz de separar mejor el conjunto de datos de test. Posiblemente se deba a la gran cantidad de variables que contiene el conjunto de datos inicial, pues SVM funciona especialmente bien en estos casos.

Además, las reglas extraídas del árbol de decisión revelan correlaciones significativas entre la satisfacción y las demás variables (confirmando lo visto en el gráfico de correlaciones), como la calidad del servicio a bordo, el tipo de viaje, la clase de cliente y la edad. El proceso de embarque en línea, combinado con la calidad del servicio WiFi y otros aspectos como la categoría de viaje y lealtad del cliente, parecen predictores clave de la satisfacción. Otras, como el género del pasajero, la localización de la puerta o el retraso del vuelo no parecen tan relevantes.

Estas conclusiones resaltan la importancia de ciertas variables en la experiencia del cliente y ofrecen pautas claras para mejorar la satisfacción de los pasajeros.

References

[1] **Airline Passenger Satisfaction**, *What factors lead to customer satisfaction for an Airline?*, https://www.kaggle.com/datasets/teejmahal20/airline-passenger-satisfaction