Informe Final: Predicción de Cancelaciones de Reservas de Hotel

Autora: Paula Campreciós
Fecha: Julio 2025

1. Introducción

El proyecto tiene como objetivo la anticipación de cancelaciones de reservas en hoteles. Estas suponen uno de los mayores problemas en el sector, ya que genera grandes pérdidas económicas por habitaciones vacías y una gestión poco eficiente de los recursos.

A través de técnicas de Machine Learning y Deep Learning aplicadas a los datos de reservas pasadas, buscamos predecir con suficiente antelación si una reserva va a ser cancelada. Esto ayudaría a los hoteles a tomar acciones preventivas (incentivos para los clientes que van a cancelar) o incluso reubicar la reserva, evitando así una habitación vacía.

2. Análisis Exploratorio de los Datos (EDA)

Los datos disponibles de reservas contienen información sobre el cliente, características de la reserva y si fue cancelada o no (variable objetivo: is_canceled).

2.1. Tratamiento de valores nulos

Se identificaron valores nulos en las siguientes columnas:

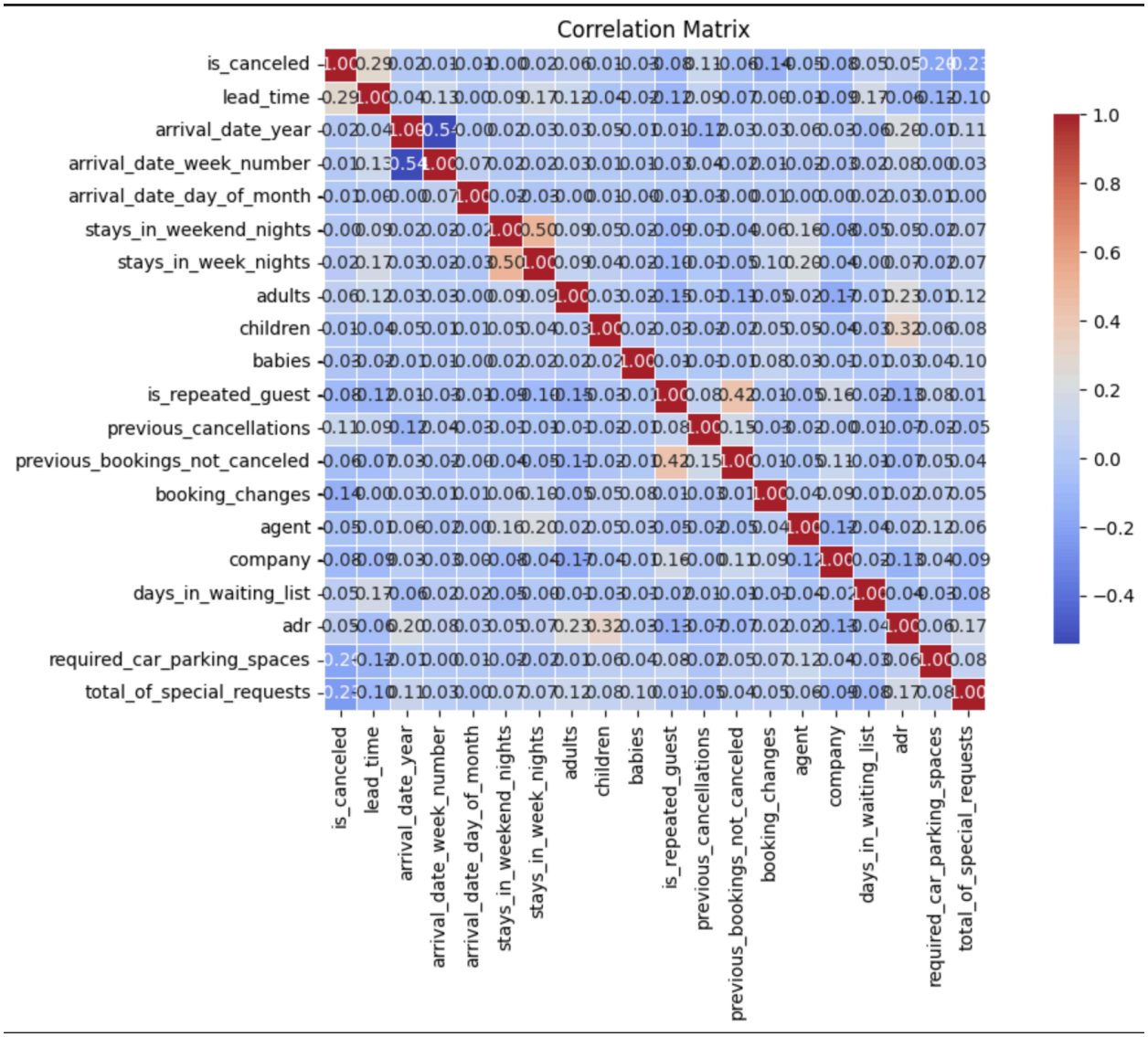
- children
- country
- agent
- company

Para imputar los valores nulos se han usado las siguientes técnicas:

- **0** para las columnas numéricas
- La moda para las columnas categóricas

2.2. Correlación y feature engineering

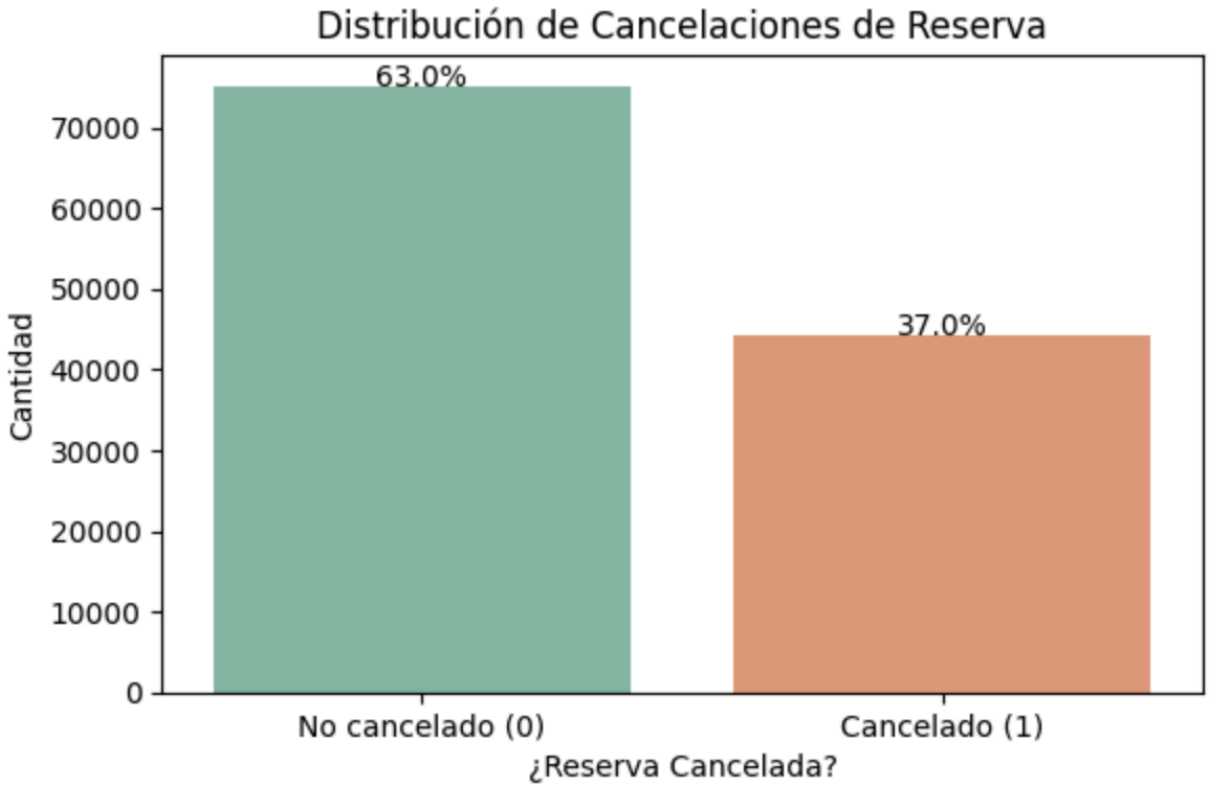
Se detectó correlación alta entre las variables arrival_date_year y arrival_date_week_number. Para reducir la colinealidad, se construyó una variable nueva arrival_time_index, que resume la información temporal de ambas. Después se borraron las columnas originales.



Se hizo un análisis de posibles **outliers** en variables numéricas, pero se decidió conservarlas, ya que podrían reflejar reservas reales excepcionales.

Para las **variables categóricas**, se analizó la distribución y se consideró que no había una desproporción excesiva como para requerir técnicas adicionales de balanceo.

La variable dependiente is_canceled tenía una distribución de **37% canceladas** y **63% no canceladas**, lo que indica un cierto desbalance, pero no esperado. Por eso, no se aplicaron técnicas de balanceo.



Finalmente, se eliminaron las variables reservation_status y reservation_status_date ya que estaban directamente relacionadas con el target y provocaban **overfitting** si se incluían en el entrenamiento.

3. Preprocesamiento y Preparación de Datos

Para estructurar el sistema de entrenamiento de los modelos se definieron varias funciones:

- **Preprocesador básico**:
  - Imputación de nulos como se explicó anteriormente.
  - Escalado de variables numéricas.
  - Codificación de variables categóricas mediante One-Hot Encoding.
- **Preprocesador con PCA (alternativa)**:
  - Se aplicó encima del preprocesador básico un análisis de componentes principales (PCA), manteniendo el 95% de la varianza. Sin embargo, al aplicarlo se observó que los modelos entrenados con PCA tenían **peor rendimiento**, por lo que se descartó.
- **Modelo Keras**
  - Definimos las capas y configuraciones en una función para una posterior aplicación más limpia.

4. Entrenamiento y Evaluación de Modelos

Los siguientes modelos fueron entrenados y evaluados:

- **Logistic Regression**
- **Decision Tree**
- **Random Forest**
- **XGBoost**
- **Keras (MLP neural network)**

El modelo de Keras se entrenó por separado debido a su distinta estructura de implementación.

Para cada modelo se implementó un **Grid Search** en los hiper parámetros para maximizar la métrica principal elegida.

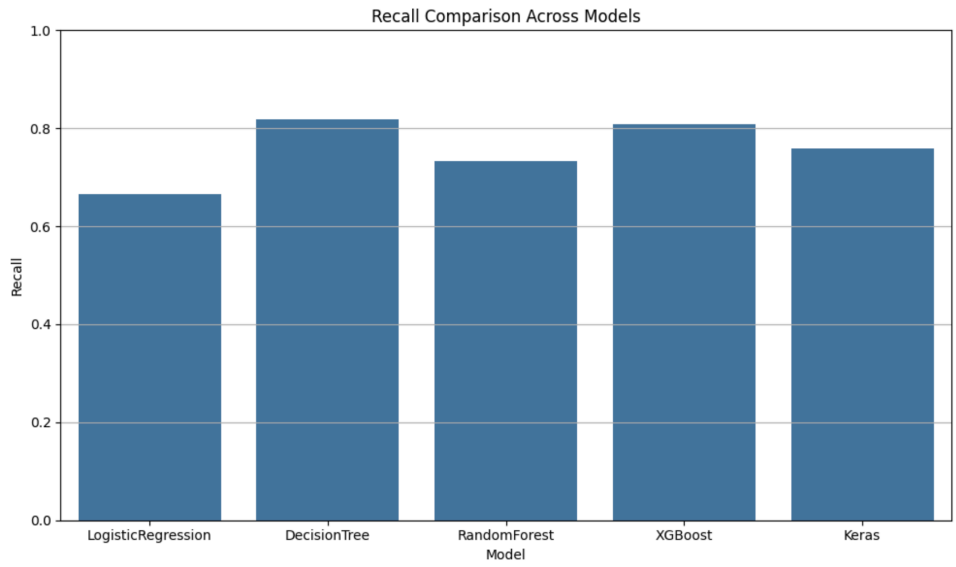
Métrica seleccionada: **Recall**

Dado que lo que queremos es **detectar correctamente la mayoría de reservas que sí se cancelan**, se decidió usar el **Recall**. Esto permite aplicar las acciones preventivas asegurándonos que nos dirigimos al público objetivo y no malbaratar recursos para aquellos que no tienen intención de cancelar..

5. Resultados

Los mejores modelos fueron: Decision Tree and XG Boost

	model	best_params	accuracy	precision	recall	roc_auc
0	LogisticRegression	{'C': 10}	0.819750	0.813475	0.666139	0.896343
1	DecisionTree	{'max_depth': 20}	0.868289	0.825046	0.817863	0.915132
2	RandomForest	{'max_depth': 20, 'n_estimators': 50}	0.872309	0.903734	0.733409	0.948831
3	XGBoost	{'max_depth': 6, 'n_estimators': 100}	0.881481	0.863093	0.808253	0.953614
4	Keras	{'lr': 0.01, 'dropout_rate': 0.2}	0.864185	0.857453	0.759638	0.939540



Aunque el **Decision Tree** obtuvo el Recall más alto, el **XG Boost** alcanzó una métrica casi Es por eso, que se seleccionó **XG Boost** como el modelo final por su equilibrio entre Recall y capacidad predictiva global.

6. Reflexión Crítica: Limitaciones y Mejores Futuras

Limitaciones y mejoras

- **Falta de conocimiento de hostelería:** Aunque se analizaron las variables y se pudieron determinar algunos comportamientos de dichas, contar con conocimiento del sector hotelero ayudaría a una toma de decisiones más informadas sobre qué variables conservar, combinar o eliminar.
- **Datos no disponibles:** Existen factores externos y otros datos que no están incluidos en el dataset y podrían afectar en la decisión de cancelación.
- **Generalización limitada:** El modelo ha sido entrenado sobre un conjunto reducido de datos y específicos de una cadena hotelera. Si quisiéramos extrapolarlo a otros hoteles necesitaríamos más datos y más diversos.