Juan Esteban Aguirre: Análisis de Negocio + Despliegue + Tablero de Datos
Jose Gabriel Bernal: Análisis de Negocio + Ciencia de datos clasificación + Ciencia de datos regresión
Alfonso Villarreal: Análisis de Negocio + Ingeniería de datos + Ciencia de datos regresión
Víctor Blanco Martínez: Análisis de Negocio + Análisis de datos + Ciencia de datos clasificación

Etapa 1: Planeación del proyecto de análisis de datos

El archivo se encuentra adjunto con la entrega.

Etapa 2 - Extracción, limpieza y transformación de datos

Informe de Preprocesamiento y Limpieza de Datos Airbnb Barcelona

Descripción del Conjunto de Datos Inicial

El conjunto de datos original listings.csv contenía 19410 observaciones y 79 columnas. La estructura inicial presentaba una mezcla de tipos de datos: 23 variables numéricas decimales float64, 20 enteras int64 y 36 categóricas.

Hallazgos iniciales:

Se detectó una gran cantidad de valores ausentes, sumando un total de 148085 datos nulos en todo el dataframe en donde la columna calendar_updated estaba completamente vacía.

También existían múltiples columnas con información repetitiva como diferentes formatos para la misma información y columnas de texto no estructurado como URLs yd descripciones largas las cuales entorpecerían el modelo.

Además, la variable más importante: price estaba en formato texto con símbolos de moneda lo cual no nos sería para el modelo

Para la limpieza de la base de datos se eliminaron las columnas con más del 30% de datos faltantes, IDs, URLs y textos descriptivos complejos. También se descartaron variables redundantes relacionadas con la disponibilidad y métricas de host repetitivas para evitar colinealidad.

Se limpió la variable price convirtiéndola a numérica. Se transformó host_since para calcular la antigüedad del anfitrión en años host_years.

Codificación: Se convirtieron variables booleanas t/f a binarias 1/0. Se aplicó One-Hot Encoding variables dummy para room_type y neighbourhood_group. Para neighbourhood_cleansed se utilizó codificación por frecuencia para evitar una explosión de dimensiones.

Se transformaron listas de texto amenidades y verificaciones en conteos numéricos n_amenities, n_verifications.

Se identificaron y eliminaron 507 registros duplicados basándose en la coincidencia exacta de latitud, longitud, precio y número de habitaciones.

Para remplazar datos faltantes se utilizó Imputación Iterativa IterativeImputer para variables estructurales beds, bedrooms, bathrooms, accommodates aprovechando las correlaciones entre ellas.

Se imputaron valores faltantes en tasas y puntuaciones utilizando la mediana y en variables categóricas usando la moda.

Para el tratamiento de outliers se utilizó el método del rango intercuartílico IQR, donde se eliminaron 864 valores atípicos, filtrando precios superiores a 430 euros y estancias mínimas mayores a 78.5 noches, así como registros inconsistentes como 0 camas o baños.

Características del Conjunto de Datos Final

Tras el proceso de limpieza, transformación y depuración, se generó un archivo final base_datos_limpia.csv con las siguientes características: 31415 observaciones, una reducción del 30.8% respecto al original, 7 columnas frente a las 79 iniciales, todas las variables son ahora estrictamente numéricas float64 e int64, lo que hace al conjunto de datos totalmente compatible con algoritmos de Machine Learning.

No existen valores nulos ni duplicados, y los datos extremos han sido normalizados dentro de rangos lógicos para el mercado inmobiliario.

Link al notebook: https://colab.research.google.com/drive/13a0PIKiZBxRoPQkAByva8UoRnKiY7C6g?usp=sharing

Etapa 3 - Análisis Exploratorio de Datos

El análisis exploratorio de los datos de Airbnb de Barcelona ha revelado insights significativos que pueden guiar decisiones estratégicas para anfitriones como Carina y la optimización de propiedades. A continuación, se detallan los hallazgos principales y

cómo responden a las preguntas de negocio planteadas.

1. ¿Cómo puede un anfitrión nuevo como Carina posicionarse de manera competitiva y legal?

Posicionamiento Competitivo:

- Calidad y Reputación: El análisis de correlación muestra que host_acceptance_rate, host_is_superhost y reviews_per_month tienen correlaciones positivas con el price. Esto sugiere que una alta tasa de aceptación, el estatus de 'Superanfitrión' y un flujo constante de reseñas (indicando alta actividad y popularidad) son factores clave para establecer precios competitivos y atraer huéspedes. El modelo de regresión logística, que clasifica si un Airbnb es recomendable (basado en review_scores_value > 4.5), refuerza la importancia de la calidad de la experiencia ofrecida.
- Amenidades: La variable n_amenities (número de amenidades) también tiene una correlación positiva con el price. Un anfitrión nuevo como Carina debería considerar ofrecer una buena gama de servicios para mejorar el atractivo de su listado y justificar un precio competitivo.

2. ¿Qué tipos de propiedades son más recomendables para maximizar ingresos y minimizar riesgos?

Maximización de Ingresos:

- Tamaño y Capacidad: Las variables accommodates, beds y bedrooms muestran fuertes correlaciones positivas con price. Esto indica que las propiedades con mayor capacidad (más huéspedes, camas y habitaciones) tienden a generar precios más altos y, por lo tanto, mayores ingresos potenciales.
- Tipo de Habitación: La room_type_Private room presenta una correlación negativa con price, mientras que room_type_Hotel room tiene una correlación positiva menor. Esto sugiere que las propiedades listadas como "Entire home/apt" (la categoría de referencia si estas son variables dummy) son probablemente las más rentables en comparación con las habitaciones privadas o compartidas.
- Reserva Instantánea: instant_bookable tiene una correlación positiva con price, lo que podría indicar una preferencia del mercado o una mayor demanda que permite precios más altos para listados con esta opción.

3. ¿Qué zonas muestran mayor demanda según el número de reviews y disponibilidad?

Indicadores de Demanda (Reviews):

- Cantidad y Frecuencia de Reseñas: number_of_reviews y reviews_per_month tienen correlaciones positivas con price. Un alto número de reseñas y una frecuencia elevada de las mismas son fuertes indicadores de una propiedad popular y, por ende, de alta demanda.
- Zonas Populares: neighbourhood_group_cleansed_Eixample muestra una correlación positiva significativa con el price. Esto sugiere que Eixample es una de las zonas más demandadas y, por lo general, con precios más elevados. Otras zonas como Sants-Montjuïc, Sant Andreu, Nou Barris y Horta-Guinardó muestran correlaciones negativas con el precio, lo que podría indicar menor demanda para propiedades de alto valor en estas áreas o una oferta diferente.

Indicadores de Demanda (Disponibilidad):

- Disponibilidad en Temporada Alta: availability_eoy (disponibilidad a fin de año) tiene una correlación positiva con price. Esto podría indicar que las propiedades disponibles durante periodos de alta demanda (como las vacaciones de fin de año) pueden establecer precios más altos. Una baja disponibilidad en general podría inferirse como un indicador de alta demanda, aunque availability_30 y availability_365 no muestran una correlación tan fuerte con el precio.

https://colab.research.google.com/drive/1IYJJil4Vr8UMxTD-3AkEYeUD_wnrLmQO?usp=sharing

Etapa 4 - Modelos predictivos

Etapa 4a - Modelos predictivos Tec

Entregables

1. Modelo Inicial de Regresión Lineal

El primer modelo de regresión lineal múltiple se construyó utilizando todas las variables disponibles en el dataset, excepto la variable objetivo price . Los datos se dividieron en conjuntos de entrenamiento (70%) y prueba (30%). El modelo fue entrenado y evaluado, obteniendo un $R2$ de aproximadamente 0.726 tanto para el conjunto de entrenamiento como para el de prueba . Esto indica que el 72% de la variabilidad en los precios puede ser explicada por las variables del modelo.

2. Análisis Numéricos y Gráficos de los Residuales (Modelo Inicial)

Para el primer modelo, se calcularon los residuales (diferencia entre los valores reales y predichos) para el conjunto de entrenamiento . Se visualizó la distribución de estos residuales mediante un histograma . El gráfico mostró una distribución que, aunque no perfectamente normal, se concentraba alrededor de cero, sugiriendo un ajuste razonable. Sin embargo, el análisis general de los residuales confirmó que no estaban distribuidos normalmente, lo que es un hallazgo importante para futuras mejoras .

3. Análisis de los Índices de Correlación y Potencia

Se realizó un análisis de correlación para identificar las variables con mayor relación con el precio. Se generó un mapa de calor de la matriz de correlación para todas las variables numéricas . Posteriormente, se extrajo y ordenó la correlación de cada variable con price . Los resultados de $R2$ (lm.score()) para cada modelo (0.726 para el modelo completo, 0.558 para el top 10 y 0.531 para el top 5) sirvieron como métrica de potencia del modelo, indicando su capacidad predictiva.

4. Optimización del Modelo, Descripción e Interpretación

Se realizaron dos intentos de optimización basados en el análisis de correlación:

Intento 2: Se construyó un modelo de regresión lineal utilizando las 10 variables más correlacionadas con price . Este modelo obtuvo un $R2$ de aproximadamente 0.558 . Los residuales de este modelo mostraron una distribución menos concentrada alrededor de cero en comparación con el primer modelo.

Intento 3: Se creó un tercer modelo con las 5 variables más correlacionadas con price . Este modelo resultó en un $R2$ aún más bajo, de aproximadamente 0.531 . La distribución de sus residuales fue la más dispersa, indicando un peor ajuste.

La interpretación de los $R2$ y los análisis de residuales reveló que el modelo inicial que utilizaba todas las variables demostró ser el mejor, con el mayor $R2$ y la distribución de residuales más favorable. Esto sugiere que reducir el número de variables basándose únicamente en la correlación no mejoró el rendimiento del modelo en este caso.

https://colab.research.google.com/drive/1IYJJil4Vr8UMxTD-3AkEYeUD_wnrLmQO?usp=sharing

Etapa 4b - Modelos predictivos Andes

Entregables

1. Modelos iniciales de regresión y clasificación.

Para la clasificación se definió la variable objetivo recommended, una varoable binaria basada en criterios de calidad del alojamiento (rating ≥ 4.5), reputación mínima (al menos 10 reviews) y precio no superior al percentil 75. Se entrenó un modelo inicial usando una red neuronal con arquitectura 64–32–16, activación ReLU, optimizador Adam y pérdida binary_crossentropy. Este modelo base alcanzó un desempeño alto (Accuracy=0.96, AUC≈0.995), aunque tras eliminar variables que generaban leakage la métrica se estabilizó en AUC≈0.84, lo cual representa un rendimiento realista. Tras analizar los costos de error del negocio, se seleccionó como métrica principal la precision, priorizando evitar falsos positivos (recomendar un alojamiento inapropiado). El modelo final elegido fue cfg_4_lr_mas_bajo, con precision≈0.70 y AUC≈0.84, ofreciendo el mejor equilibrio práctico. Para regresión se inició con un modelo que predecía directamente el precio, obteniendo un RMSE≈49€. Posteriormente se transformó la variable objetivo usando log(price) para estabilizar la varianza y mejorar la capacidad predictiva. El modelo base con esta transformación logró un RMSE≈46.7€.

2. Ingeniería de características.

Se aplicó ingeniería de características mínima pero efectiva: (i) se generó la variable log_price como objetivo para la regresión, (ii) se codificaron las variables categóricas mediante one-hot encoding y (iii) se eliminaron los valores atípicos (top 1% del

precio) para reducir la distorsión provocada por listings extremadamente costosos. Tras esta depuración, los modelos fueron reconstruidos desde cero para evitar contaminación de datos (leakage) y asegurar consistencia entre entrenamiento y evaluación.

3. Búsqueda amplia de hiperparámetros.

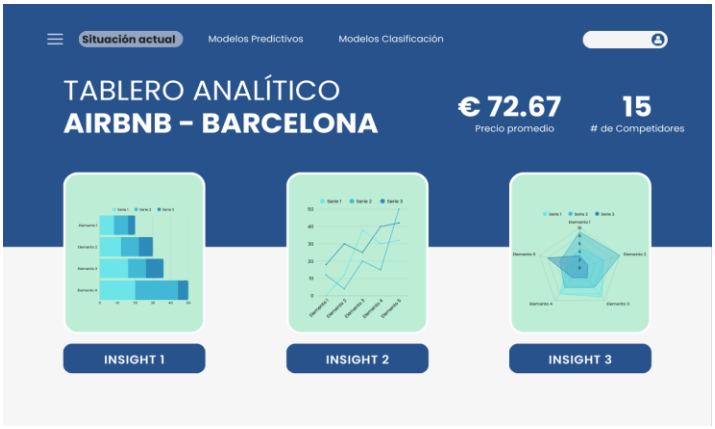
Se implementó MLflow para registrar configuraciones, métricas, parámetros y modelos.
Para clasificación se probaron más de 20 configuraciones variando profundidad, número de neuronas, tasas de dropout y learning rate, seleccionándose finalmente el modelo que maximizó precision bajo las restricciones del caso.
Para regresión se llevaron a cabo más de 20 corridas adicionales con activación ELU, variando la arquitectura (32–16–8 hasta 256–128–64), tasas de aprendizaje, épocas y dropout. Esto permitió identificar configuraciones robustas frente a sobreajuste y optimizar el error relativo.

4. Análisis de resultados.
En clasificación, el modelo final ofreció un equilibrio adecuado entre precisión (≈0.70) y AUC (≈0.84), favoreciendo la minimización de falsos positivos como se requiere en un sistema de recomendación conservador.
En regresión, la combinación de log-transformación y eliminación de outliers produjo mejoras en la capacidad predictiva, alcanzando un RMSE≈45.99€, MAE≈30.3€ y error relativo≈32%. Las mejoras obtenidas reflejan un proceso de modelamiento consistente con buenas prácticas de machine learning, incluyendo control de leakage, búsqueda sistemática de hiperparámetros y evaluación del rendimiento en datos fuera de muestra.

El ipynb se encontrará en el repositorio github del proyecto, junto con archivos llamados runs.csv que muestran las métricas de las corridas.

Etapa 5 - Diseño y desarrollo del tablero

Diseño inicial:



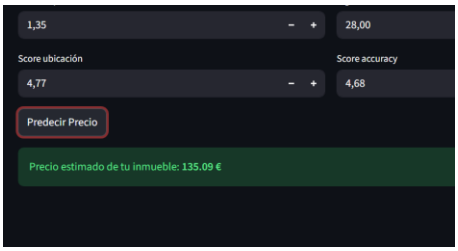
Para el diseño inicial del tablero se planteó la creación de tres pestañas, cada una correspondiente a los contenidos desarrollados en las etapas 2–3, 4a y 4b. La pestaña principal tiene como objetivo presentar los insights que permiten responder la mayor parte de las preguntas de negocio. Para facilitar la exploración, esta sección incluye un panel lateral de filtros que ayuda al usuario a identificar rápidamente los insights relevantes.

Las preguntas de negocio restantes se abordan mediante los modelos desarrollados en el proyecto. No se realizó un mockup para estas pestañas debido al elevado número de parámetros asociados a los modelos, lo cual habría generado una interfaz con una cantidad excesiva de botones y listas desplegables. Además, dado que el tablero es un producto orientado al usuario final, no se considera adecuado incluir gráficas sobre las métricas del modelo.

Descripción del tablero:

El diseño final del tablero no presenta diferencias significativas de diseño con respecto al mockup. Se conservaron las 3 pestañas, el filtro lateral y la distribución de los insights. Sin embargo, se decidió cambiar el asset y usar el interprete/librería streamlit dado que por defecto utiliza assets fríos/oscuros y tiene compatibilidad con markdown.

Con respecto a los modelos tanto de clasificación como de predicción se decidió crear 2 subpestañas dentro de cada uno, esto para separar los modelos realizados por TEC y Uniandes. Dentro de cada una de las subpestañas se encuentran todos los inputs agrupados por especificaciones para predecir el precio según el modelo elegido, así como un botón al final para predecir el precio o la clasificación, según sea el caso.



Etapa 6 - Despliegue y mantenimiento

Despliegue local:

Para poder desplegar el modelo es necesario remitirse a la carpeta “Tablero de datos” que está disponible en el repositorio. Una vez ahí es plausible lanzar el tablero de forma local, para ello es necesario abrir una terminal e instalar las librerías especificadas en el archivo ***requirements.txt***.

Una vez instalados es necesario verificar que el entorno/terminal tenga abierta la carpeta “Tablero de datos”. Ya con esto se debe ejecutar el comando ***streamlit run app.py*** cuando se ejecute dicha línea en la terminal se abrirá el dash en el navegador predeterminado (se recomienda usar Microsoft Egde pues en otros exploradores como Chrome es posible que las gráficas de tipo mapa no carguen).

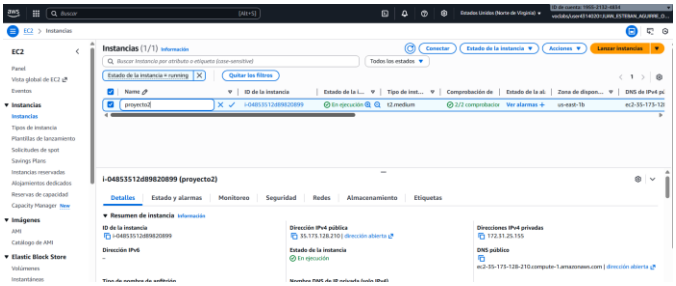
```
(base) C:\Users\Usuario\OneDrive - Universidad de los andes\Documentos\ANDES (´,_ ,)\2025-II\IIND-4130\Proyecto 2\
Repositorio\proyecto-final-actd\Tablero de datos>streamlit run app.py

  You can now view your Streamlit app in your browser.

  Local URL: http://localhost:8501
  Network URL: http://192.168.80.10:8501
```

Despliegue AWS:

Para este caso se creó una instancia EC2 en AWS t2.medium de 50 GB dado los múltiples modelos y gráficas a cargar.



Véase la IP tanto pública como privada. Una vez creada la instancia se procedió a conectarse localmente a la instancia e instalar Docker

y agregar el repositorio al sistema. Ahora bien, localmente se subieron todos los archivos necesarios para cargar el dash.

```
ubuntu@ip-172-31-25-155:~$ ls
BarcelonaAbnb_limpio.csv  app.py                  modelo_clasificacion.keras  requirements.txt          scaler_regresion.pkl
Dockerfile                data_clean_barcelona.csv  modelo_regresion.keras      scaler_clasificacion.pkl
ubuntu@ip-172-31-25-155:~$
```

Ya con ello se procedió a crear la imagen asociada y ejecutar el modelo. Es necesario mencionar que streamlit utiliza el puerto 8501.

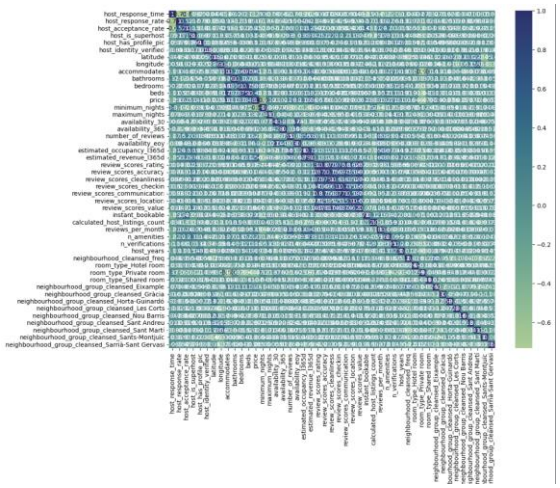
```
ubuntu@ip-172-31-25-155:~$ sudo docker images
IMAGE         ID             DISK USAGE   CONTENT SIZE   EXTRA
test:latest   b48644aa84c3       4.47GB         1.08GB   U
ubuntu@ip-172-31-25-155:~$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED              STATUS                      PORTS                                       NAMES
342292697fbc   test:latest    "streamlit run app.p…"   About an hour ago    Up About an hour            0.0.0.0:8501->8501/tcp, [::]:8501->8501/tcp   pedantic_
robinson
7a2d4cf710e3   026d3976504e   "streamlit run app.p…"   5 hours ago          Exited (0) 5 hours ago                                                  elated_lu
miere
```

Ya con esto se accedió a http://35.173.128.210:8501/ para verificar el tablero desplegado (naturalmente la IP mostrada cambia)

Etapa 7 - Evaluación

¿Cómo puede un anfitrión nuevo como Carina posicionarse de manera competitiva y legal?

Los nuevos anfitriones deberían aspirar a ser super-host pues su densidad del precio es más homogénea a comparación de no ser super-host. Por su parte, para alcanzar rápidamente un buen número de reseñas debería ofrecer propiedades en hoteles o habitaciones en apartamentos ofreciendo precios de mínimo 100 euros y máximo 144 euros sin importar el barrio, en los primeros años de anfitrión. Luego, puede aspirar a aumentar el rango en 50 euros. Esto pues dada la gran oferta de inmuebles en Barcelona el barrio no resulta tan influyente.



¿Qué tipos de propiedades son más recomendables para maximizar ingresos y minimizar riesgos?

La propiedad **room in hotel** es la puede maximizar ingreso y minimizar riesgo puesto presenta la segunda media del precio más alta y con menor variabilidad y tamaño en su caja. Otras buenas alternativas son **entire serviced apartment, entire rental unit y entire condo**.

¿Qué zonas muestran mayor demanda según el número de reviews y disponibilidad?

El barrio la Dreta de l'Eixample es el que presenta mayor demanda, pero su precio promedio no lo pone en el top 10. En cambio, la sagrada familia es el según barrio con mayor demanda y el sexto con mayor precio promedio.

Etapa 8 - Presentación

El archivo está adjunto en la entrega.