

ANÁLISIS PREDICTIVO DE PRECIOS DE AIRBNB EN LA CIUDAD DE MÉXICO UTILIZANDO MODELOS DE APRENDIZAJE SUPERVISADO

ISRAEL CERVANTES JUÁREZ^a, BULMARO JUÁREZ HERNÁNDEZ^a, JOSÉ JUAN CASTRO ALVA^a,
FRANCISCO SOLANO TAJONAR SANABRIA^a

^aFacultad de Ciencias Físico-Matemáticas
Benemérita Universidad Autónoma de Puebla
e-mail: israel.cervantesj@alumno.buap.mx, bjuarez@fcfm.buap.mx,
jose.castroalva@correo.buap.mx, francisco.tajonar@correo.buap.mx

Este proyecto se centra en el análisis predictivo de precios de Airbnb en CDMX utilizando modelos de aprendizaje supervisado. Se realiza un análisis exploratorio de datos para comprender la estructura y características del conjunto de datos. Posteriormente, se aplican modelos de regresión lineal y regresión múltiple para predecir precios y disponibilidad, ajustando los modelos para mejorar su rendimiento. Además, se implementa el clasificador K-Nearest Neighbors para comparar su efectividad frente a los modelos de regresión.

Keywords: Ciencia de datos, modelos de regresión, KNN, predicciones.

1. Introducción

El mercado de alquileres a corto plazo ha experimentado un notable auge en las últimas décadas, especialmente en ciudades con alta demanda turística y residencial, como la Ciudad de México. Este estudio aplica técnicas de aprendizaje supervisado para estimar los precios en Airbnb, con el objetivo de afinar las predicciones en un entorno de mercado en constante transformación.

Además de mejorar la precisión de las estimaciones, este enfoque proporciona una visión detallada de los factores que influyen en los precios y la disponibilidad de las propiedades. Al integrar análisis avanzados con el contexto del mercado, el estudio busca ofrecer información valiosa que facilite decisiones más informadas para propietarios y arrendatarios en un competitivo mercado de alquileres.

2. Objetivo

El objetivo de este estudio es evaluar la efectividad de diferentes técnicas de aprendizaje supervisado para predecir precios y disponibilidad de Airbnb en CDMX. Para ello, se realizarán análisis exploratorios de datos, se aplicarán modelos de regresión lineal y múltiple, y se comparará el rendimiento de estos modelos con el clasificador KNN.

3. Metodología

3.1. Recopilación y Análisis Exploratorio de los Datos. El conjunto de datos utilizado en este proyecto se descargó de la página oficial de Airbnb, específicamente para la Ciudad de México. Contiene un total de 26,536 entradas y 18 columnas, cada una proporcionando detalles clave sobre las propiedades listadas en la plataforma.

RangeIndex: 26536 entries, 0 to 26535			
Data columns (total 18 columns):			
#	Column	Non-Null Count	Dtype
0	id	26536	non-null int64
1	name	26536	non-null object
2	host_id	26536	non-null int64
3	host_name	26536	non-null object
4	neighbourhood_group	0	non-null float64
5	neighbourhood	26536	non-null object
6	latitude	26536	non-null float64
7	longitude	26536	non-null float64
8	room_type	26536	non-null object
9	price	23338	non-null float64
10	minimum_nights	26536	non-null int64
11	number_of_reviews	26536	non-null int64
12	last_review	22664	non-null object
13	reviews_per_month	26536	non-null float64
14	calculated_host_listings_count	26536	non-null int64
15	availability_365	26536	non-null int64
16	number_of_reviews_ltm	26536	non-null int64
17	license	0	non-null float64

Fig. 1. Descripción del *DataFrame*

Las columnas incluyen identificadores únicos para propiedades y anfitriones, nombres de propiedades y anfitriones, información sobre el vecindario y la ubicación exacta mediante coordenadas geográficas. También se registra el tipo de habitación, el precio por noche, el número mínimo de noches requerido para reservar, y estadísticas sobre reseñas y disponibilidad.

Se tiene que `neighbourhood_group` y `license`, no contienen datos válidos, lo que indica la necesidad de limpieza de datos. La presencia de valores nulos en columnas clave como `price` y `last_review` sugiere que se requerirá un tratamiento adecuado para estos datos faltantes antes de proceder con el análisis predictivo. Utilizamos las abreviaturas en el código:

Table 1. Delegaciones y sus abreviaturas

Delegación	Abreviatura
Álvaro Obregón	AO
Azcapotzalco	AZ
Benito Juárez	BJ
Coyoacán	CO
Cuajimalpa de Morelos	CM
Cuauhtémoc	CU
Gustavo A. Madero	GA
Iztacalco	IZ
Iztapalapa	IP
Magdalena Contreras	MC
Miguel Hidalgo	MH
Milpa Alta	MA
Tláhuac	TL
Tlalpan	TP
Venustiano Carranza	VC
Xochimilco	XO

En la figura 2 se puede observar que las delegaciones con mayor número de residencias de Airbnb son Cuauhtémoc, Miguel Hidalgo y Benito Juárez, mientras que Xochimilco, Tlalpan y Magdalena Contreras presentan una menor frecuencia.

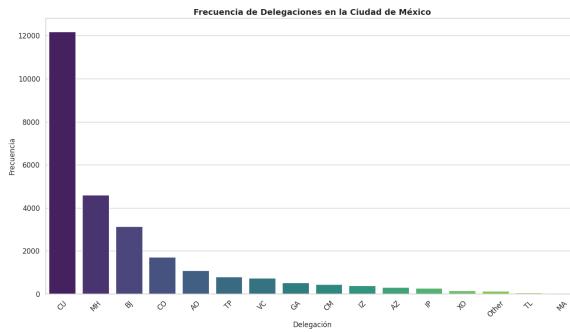


Fig. 2. Frecuencia de residencias Airbnb

Esta distribución sugiere una variabilidad en la

oferta de propiedades para alquiler en distintas áreas, con algunas delegaciones destacándose por su alta densidad de opciones para los usuarios de Airbnb, mientras que otras ofrecen menos alternativas.

En el gráfico de diagramas de caja se visualiza la detección de valores atípicos para diferentes variables del dataset de Airbnb. Cada gráfico de caja muestra la distribución de una variable específica, con la caja representando el rango intercuartil (IQR) y los bigotes extendiéndose hasta 1.5 veces el IQR desde los cuartiles. Los puntos fuera de los bigotes son considerados valores atípicos.

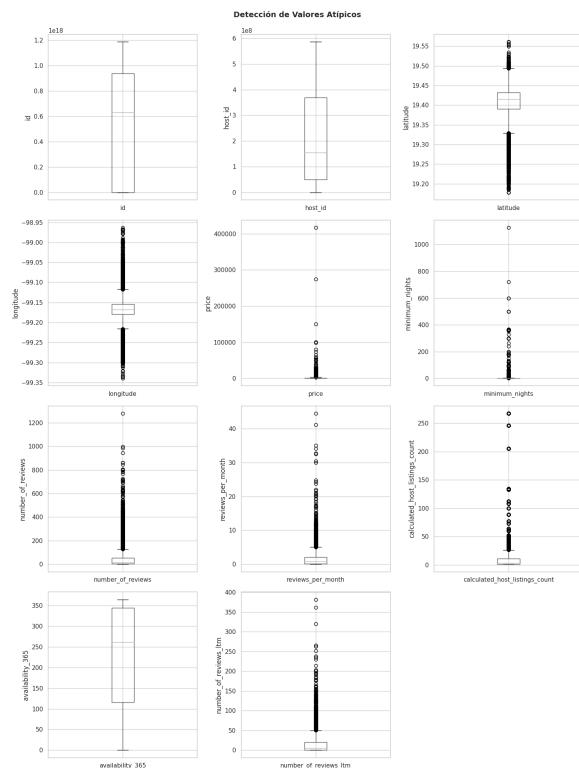


Fig. 3. Valores atípicos

Las variables, como `price`, `minimum_nights`, `number_of_reviews`, `reviews_per_month`, y `calculated_host_listings_count`, presentan una cantidad considerable de valores atípicos, evidenciado por la presencia de puntos que se extienden más allá de los bigotes. Esto indica que estas variables tienen distribuciones con colas largas o posibles anomalías que podrían influir en el análisis. Las variables de localización geográfica (`latitude` y `longitude`) y `availability_365` también muestran valores atípicos, aunque en menor cantidad.

En la figura 4 se observan relaciones lineales variadas entre las variables. Las variables de reseñas, como `number_of_reviews`, `reviews_per_month` y `number_of_reviews_ltm`, presentan correlaciones

positivas significativas entre ellas, indicando una fuerte interrelación. En cambio, variables como el `price`, `latitude` y `longitude` tienen correlaciones débiles o casi nulas con otras variables, sugiriendo una baja dependencia lineal. Esto sugiere que las variables de reseñas son más relevantes para modelos predictivos, mientras que las demás variables pueden tener un impacto menor en la predicción de precios o disponibilidad.

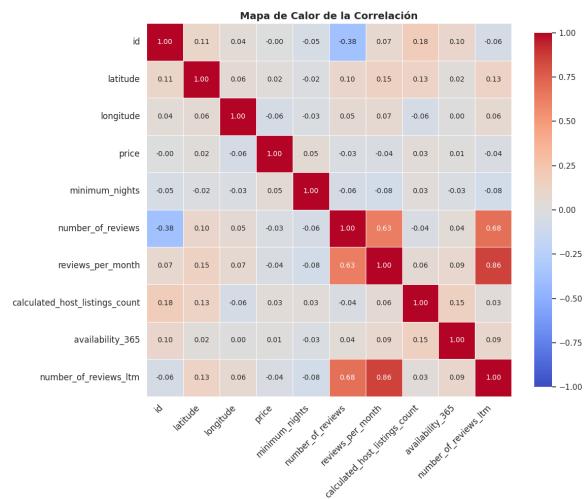


Fig. 4. Correlación entre variables

En el gráfico de dispersión que relaciona el precio y la disponibilidad anual, se observa una alta concentración de puntos en la zona de precios bajos, particularmente por debajo de los 100,000. Estos puntos se distribuyen a lo largo del eje vertical, lo que indica que la disponibilidad anual varía considerablemente independientemente del precio, aunque la mayoría de los valores de disponibilidad se agrupan en torno a los valores más altos (cerca de 365 días).

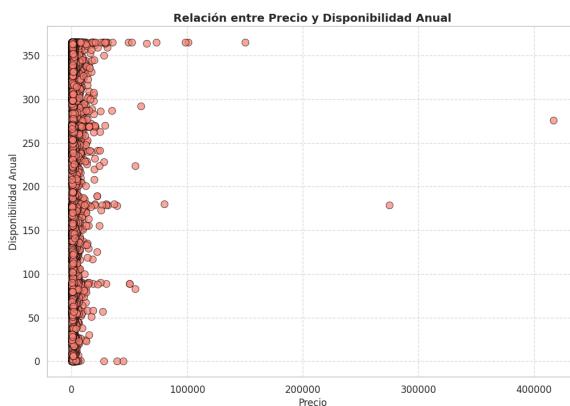


Fig. 5. Grafico de dispersión

También hay algunos valores atípicos que muestran propiedades con precios extremadamente altos y una

disponibilidad variada. En general, el gráfico sugiere que no existe una relación clara y lineal entre el precio y la disponibilidad anual, evidenciado por la dispersión de los puntos en un rango amplio a lo largo del eje de disponibilidad.

Se graficó un mapa donde se puede observar que las áreas con la mayor densidad de listados de Airbnb se encuentran en el centro de la Ciudad de México, particularmente en las zonas de Polanco, la Condesa, y la Roma, donde se visualiza un punto rojo intenso. Esto sugiere que estas áreas son populares para alquileres de corto plazo. A medida que se aleja del centro, la densidad disminuye, con áreas periféricas mostrando colores que van del amarillo al verde y finalmente al azul, lo que indica una menor cantidad de listados en esas zonas. Este patrón es consistente con la distribución geográfica esperada de la oferta de Airbnb, concentrada en zonas urbanas centrales y turísticas.

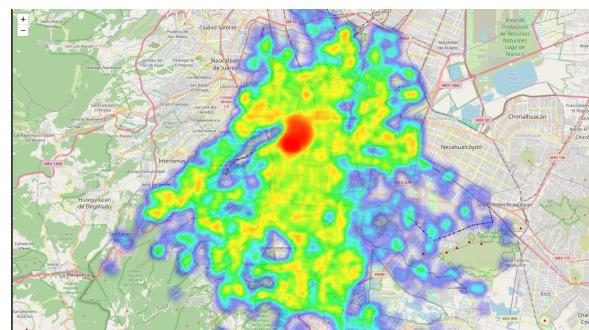


Fig. 6. Mapa de calor CDMX

La figura 7 muestra varios gráficos de densidad para diferentes variables del conjunto de datos de Airbnb. El gráfico de `id` revela una alta concentración de datos cerca de cero, con un segundo pico en un rango más alto de valores. El de `latitude` muestra una concentración alrededor de 19.4, mientras que `longitude` se concentra alrededor de -99.2. El de `price` tiene una distribución altamente sesgada a la derecha, indicando que la mayoría de los precios son bajos con algunos valores extremos muy altos. El de `minimum_nights` también presenta una distribución sesgada, con la mayoría de los valores siendo bajos. La cantidad de `number_of_reviews` y `reviews_per_month` tienen distribuciones similares, mostrando que la mayoría de los listados tienen pocas reseñas. La variable `calculated.host_listings_count` muestra que la mayoría de los hosts tienen pocos listados, con algunos hosts teniendo muchos. El de `availability_365` días tiene varios picos, con una mayor densidad alrededor de los 350 días disponibles. Por último, el gráfico `number_of_reviews_ltm` también está sesgado hacia la derecha, indicando que la mayoría de los listados tienen pocas reseñas recientes.

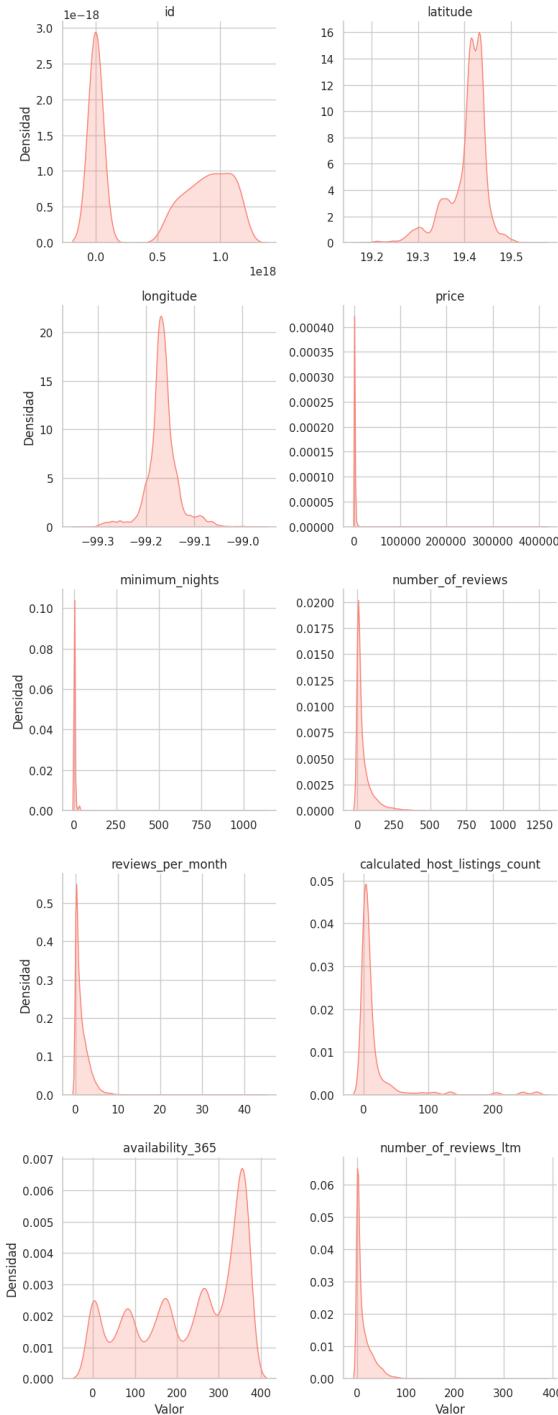


Fig. 7. Analisis de densidad

3.2. Modelo de Regresión Lineal. La regresión lineal es un modelo que describe la relación entre una variable dependiente y y una variable independiente x mediante la ecuación:

$$y = \beta_0 + \beta_1 x + \epsilon,$$

donde β_0 es la ordenada al origen, β_1 es el coeficiente de regresión, y ϵ es el término de error aleatorio. El objetivo

es estimar β_0 y β_1 para minimizar la suma de los errores cuadráticos entre los valores observados y los valores predichos por el modelo [1].

3.3. Modelo de Regresión Lineal Múltiple. La regresión lineal múltiple es un modelo que describe la relación entre una variable dependiente y y varias variables independientes x_1, x_2, \dots, x_n mediante la ecuación:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n + \epsilon,$$

donde β_0 es la ordenada al origen, $\beta_1, \beta_2, \dots, \beta_n$ son los coeficientes de regresión, y ϵ es el término de error aleatorio. El objetivo es estimar los coeficientes β_i para minimizar la suma de los errores cuadráticos entre los valores observados y los valores predichos por el modelo.

3.4. Modelo KNN. En estadística, el algoritmo de k vecinos más cercanos (k -NN) es un método de aprendizaje supervisado no paramétrico desarrollado inicialmente por Evelyn Fix y Joseph Hodges en 1951, y posteriormente ampliado por Thomas Cover. Se utiliza tanto para clasificación como para regresión. En ambos casos, la entrada consiste en los k ejemplos de entrenamiento más cercanos en un conjunto de datos [2].

El algoritmo asigna al punto z la clase más frecuente entre sus k vecinos más cercanos. Para el caso binario, se suele elegir k impar para evitar empates.

$$\text{Clase}(z) = \arg \max_y \sum_{i=1}^k \mathbb{I}_{\{y_i=y\}},$$

donde $\mathbb{I}_{\{y_i=y\}}$ es una función indicadora que toma el valor 1 si la clase del i -ésimo vecino es y , y 0 en caso contrario.

Elección del Valor de k El parámetro k determina el número de vecinos más cercanos considerados para tomar una decisión. La selección del valor de k es crítica:

- Un k pequeño puede llevar a un modelo sobreajustado (overfitting), ya que es sensible al ruido en los datos.
- Un k grande puede suavizar demasiado las predicciones y llevar a un modelo subajustado (underfitting).

La elección óptima de k generalmente se realiza mediante validación cruzada.

Cálculo de Distancias Para determinar los vecinos más cercanos, k-NN calcula la distancia entre el punto z y todos los puntos en el conjunto de entrenamiento. Las métricas de distancia más comunes son:

- **Distancia Euclídea:**

$$d_{\text{euclid}}(x_i, z) = \sqrt{\sum_{j=1}^p (x_{ij} - z_j)^2},$$

donde x_i es el i -ésimo punto de entrenamiento y z es el punto de prueba.

- **Distancia de Manhattan:**

$$d_{\text{manh}}(x_i, z) = \sum_{j=1}^p |x_{ij} - z_j|,$$

- **Distancia de Minkowski:**

$$d_{\text{mink}}(x_i, z) = \left(\sum_{j=1}^p |x_{ij} - z_j|^p \right)^{1/p},$$

donde p es un parámetro que define el tipo de distancia (por ejemplo, $p = 2$ para Euclídea, $p = 1$ para Manhattan).

- **Distancia Coseno:**

$$d_{\cos}(x_i, z) = 1 - \frac{x_i \cdot z}{\|x_i\| \|z\|},$$

donde $x_i \cdot z$ es el producto punto entre x_i y z , y $\|x_i\|$ y $\|z\|$ son las normas euclidianas de x_i y z , respectivamente.

Análisis de Bias-Variance Para entender el rendimiento del modelo, consideramos el bias y la varianza:

- **Bias:** El bias mide el error sistemático del modelo. Se define como:

$$\text{Bias}^2 = \mathbb{E}[\hat{y}(z)] - f(z),$$

donde $f(z)$ es la función verdadera que genera las etiquetas. Para k-NN, el bias se approxima al valor medio de las etiquetas de los vecinos más cercanos.

- **Varianza:** La varianza mide la variabilidad del modelo con respecto a los datos de entrenamiento. Para k-NN, la varianza se calcula como:

$$\text{Var}[\hat{y}(z)] = \frac{1}{k^2} \sum_{i=1}^k \text{Var}[y_i].$$

La varianza disminuye a medida que k aumenta, pero un k demasiado grande puede llevar a un sesgo elevado.

Complejidad Computacional

- **Entrenamiento:** La complejidad de almacenamiento es $O(n)$, donde n es el número de puntos en el conjunto de entrenamiento. No hay coste de entrenamiento significativo si no se utilizan estructuras de datos especializadas.
- **Predicción:** La complejidad es $O(n)$ para buscar los k vecinos más cercanos. El uso de estructuras como árboles k-d puede reducir este tiempo a $O(\log n)$ en el caso promedio.

Comportamiento en Altas Dimensiones En espacios de alta dimensión, la distancia entre puntos se vuelve menos informativa debido a la “maldición de la dimensionalidad”. En dimensiones altas, los puntos se dispersan y las distancias relativas se vuelven menos significativas, lo que puede degradar el rendimiento de k-NN.

3.5. Evaluación.

3.5.1. Error Absoluto Medio (MAE). Es una métrica que cuantifica el promedio de las diferencias absolutas entre los valores verdaderos y los valores predichos. Se utiliza para evaluar la precisión de un modelo de regresión, indicando cuánto se desvían en promedio las predicciones del modelo respecto a los valores reales. Se calcula como:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|,$$

donde y_i son los valores verdaderos, \hat{y}_i son los valores predichos, y n es el número total de observaciones.

3.5.2. Error Cuadrático Medio (MSE). Mide el promedio de los cuadrados de las diferencias entre los valores verdaderos y los valores predichos. Esta métrica penaliza más fuertemente los errores grandes debido al cuadrado de las diferencias, lo que puede ser útil para identificar modelos con grandes errores. Se calcula como:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2.$$

3.5.3. Raíz del Error Cuadrático Medio (RMSE). Es la raíz cuadrada del Error Cuadrático Medio (MSE) y proporciona una medida de la desviación estándar de los errores de predicción. Esta métrica está en las mismas unidades que la variable objetivo, facilitando la interpretación del error promedio del modelo. Se calcula como:

$$\text{RMSE} = \sqrt{\text{MSE}}.$$

3.5.4. Matriz de Confusión. La matriz de confusión es una herramienta que se utiliza para evaluar el desempeño de un modelo de clasificación. Muestra el número de predicciones verdaderas y falsas del modelo, distribuidas en cuatro categorías.

		True Class	
		Positive	Negative
Predicated Class	Positive	TP	FP
	Negative	FN	TN

Fig. 8. La estructura básica de una matriz de confusión

Donde:

- TP (True Positives) es el número de verdaderos positivos, es decir, casos en los que el modelo predijo correctamente la clase positiva.
- TN (True Negatives) es el número de verdaderos negativos, es decir, casos en los que el modelo predijo correctamente la clase negativa.
- FP (False Positives) es el número de falsos positivos, es decir, casos en los que el modelo predijo incorrectamente la clase positiva.
- FN (False Negatives) es el número de falsos negativos, es decir, casos en los que el modelo predijo incorrectamente la clase negativa.

3.5.5. Accuracy. La precisión mide la proporción de predicciones correctas realizadas por el modelo en relación con el total de predicciones. Se calcula como:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}.$$

3.5.6. Recall. La sensibilidad mide la proporción de verdaderos positivos detectados entre todos los positivos reales. Se calcula como:

$$\text{Recall} = \frac{TP}{TP + FN}.$$

3.5.7. Specificity. La especificidad mide la proporción de verdaderos negativos detectados entre todos los negativos reales. Se calcula como:

$$\text{Specificity} = \frac{TN}{TN + FP}.$$

3.5.8. F-Score. El F-Score es una medida de la precisión del modelo que combina la precisión y el recall. Se calcula como:

$$\text{F1-Score} = \frac{2 \cdot \text{Recall} \cdot \text{Precision}}{\text{Recall} + \text{Precision}}.$$

3.6. Métodos de solución. Para ajustar el modelo de regresión lineal, primero se visualizó la relación entre el precio y la identificación del registro utilizando un gráfico de dispersión con una línea de ajuste. Posteriormente, se limpian los datos eliminando la columna `host_id` y valores atípicos en la variable `minimum_nights`, restringiendo los valores a menos de 31 noches para asegurar la calidad del análisis.

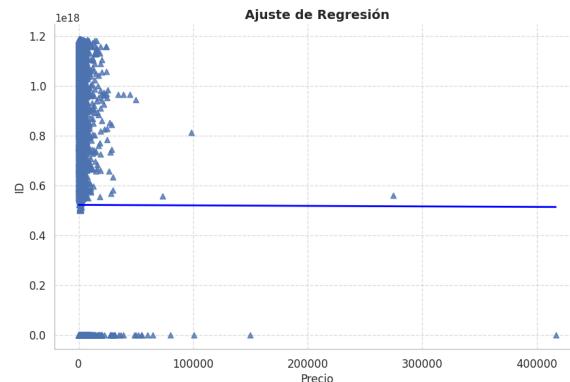


Fig. 9. Ajuste de modelo

Se prepararon los datos eliminando filas con valores NaN en las columnas `price` y `availability_365`, y se dividieron en conjuntos de entrenamiento y prueba, con un 20 % de los datos reservados para pruebas. Se utilizó el modelo de **regresión lineal** de `sklearn` para ajustar el modelo a los datos de entrenamiento.

La ecuación de la regresión lineal con el intercepto y el coeficiente proporcionados es:

$$\hat{y} = 240.87 + 0.00027 \cdot x$$

Table 2. Comparación de valores reales y predichos

Id	Actual	Predicted
0	270	240.939684
1	353	241.314821
2	318	241.171354
3	241	241.388052
4	359	240.949756
...
4632	27	241.145492
4633	269	240.955201
4634	359	241.040682
4635	181	241.119630
4636	150	241.230429

La figura 10 muestra un gráfico de dispersión con una línea de regresión lineal ajustada sobre los datos. En el eje horizontal se encuentran los valores predichos, que varían desde 0 hasta 100,000, mientras que en el eje vertical se encuentran los valores reales que varían desde 0 hasta 350. La mayoría de los datos se concentran cerca del origen, con valores predichos bajos (menos de 20,000) y valores reales dispersos en un rango de 0 a 350.

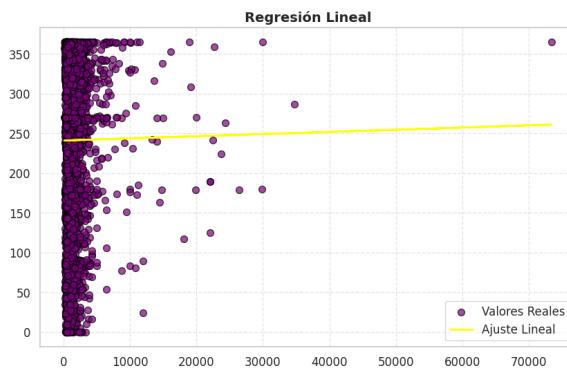


Fig. 10. Valores reales vs predichos

La línea de ajuste lineal es casi horizontal, lo que sugiere que hay poca o ninguna relación lineal entre las variables. La tendencia general indica que a medida que los valores predichos aumentan, los valores reales se mantienen en un rango estrecho, sin un patrón claro de aumento o disminución significativa. Esto podría indicar una mala capacidad predictiva del modelo lineal para este conjunto de datos.

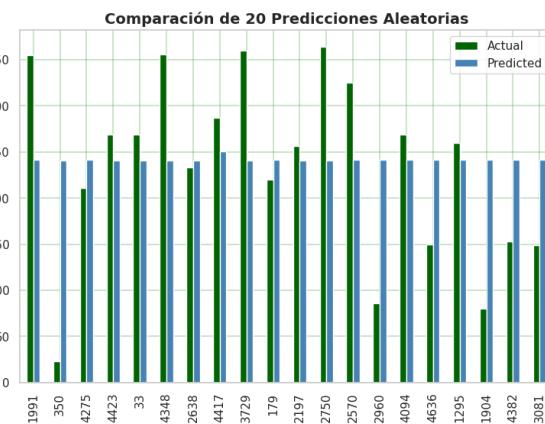


Fig. 11. Predicciones

El gráfico de barras compara las primeras 20 predicciones de disponibilidad anual de Airbnb generadas por un modelo de regresión lineal con los valores reales, donde las barras verdes representan los valores reales y las azules las predicciones. En la mayoría de los casos, el modelo subestima la disponibilidad anual, ya que las

barras azules son consistentemente más bajas que las verdes. Además, el modelo no captura adecuadamente la variabilidad de los valores reales, que muestran una gran dispersión, mientras que las predicciones están concentradas en un rango más estrecho, sugiriendo limitaciones del modelo para reflejar la relación entre el precio y la disponibilidad anual de los listados de Airbnb.

Para preparar los datos para la **regresión lineal múltiple**, se convirtió las variables categóricas en valores numéricos utilizando LabelEncoder. Esto incluyó las columnas room_type, neighbourhood y delegation, que se transformaron en nuevas columnas numéricas (room_type_Cat, city_Cat, y state_Cat). La conversión se realizó para facilitar el procesamiento de datos en el modelo de regresión.

Se verificó la conversión mostrando las primeras filas del DataFrame para asegurar que la transformación se había realizado correctamente. Luego, se eliminaron las filas con valores NaN en las columnas relevantes para garantizar la integridad de los datos.

Posteriormente, se seleccionaron las características calculated_host_listings_count, room_type_Cat, city_Cat y state_Cat como variables independientes, y price como la variable dependiente. Los datos se dividieron en conjuntos de entrenamiento y prueba, con un 20 % reservado para la prueba del modelo y el 80 % restante para el entrenamiento. Este proceso preparó los datos para ajustar el modelo de regresión lineal.

La ecuación de la **regresión lineal múltiple** con el intercepto y los coeficientes proporcionados es:

$$\hat{y} = 1606.51 + 2.12 \cdot x_1 - 404.27 \cdot x_2 - 5.15 \cdot x_3 + 19.48 \cdot x_4$$

Table 3. Comparación de valores reales y predichos

Id	Actual	Predicted
0	3663.0	1651.645943
1	1652.0	1689.716007
2	18000.0	1710.953211
3	814.0	1757.149319
4	1923.0	1687.592287
...
26531	7850.0	1700.334609
26532	1478.0	1706.705771
26533	936.0	1685.468567
26534	768.0	1689.716007
26535	1302.0	1757.675060

Los residuos representan la diferencia entre los valores observados y los valores predichos por el modelo. Un patrón aleatorio de puntos alrededor de la línea horizontal en cero (marcada en amarillo) indicaría que el modelo tiene una varianza constante y que no existe una

relación sistemática entre las predicciones y los residuos, lo que es una suposición clave para que los resultados de la regresión sean fiables.

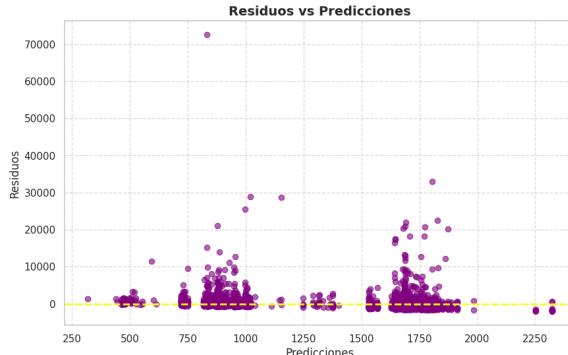


Fig. 12. Errores

Se observa que en la mayoría de los casos, las predicciones del modelo no se alinean perfectamente con los valores reales, evidenciando una discrepancia que puede ser significativa en algunos casos, como en la observación 14749 donde el valor real es significativamente más alto que la predicción.

Este tipo de comparación es útil para evaluar visualmente el desempeño del modelo. Las grandes diferencias entre los valores reales y predichos indican que el modelo podría no estar capturando adecuadamente las características subyacentes de los datos, lo que sugiere la necesidad de ajustes adicionales en la modelación, como la inclusión de variables adicionales, la prueba de otros algoritmos de machine learning, o la realización de un análisis más detallado de los datos.

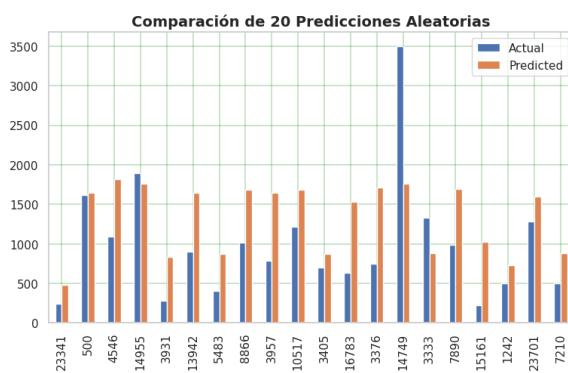


Fig. 13. Predicciones

En este proceso, se clasificaron los precios de las propiedades en diferentes rangos mediante la creación de una lista de condiciones basada en los valores de la columna `price`. Los rangos de precios definidos fueron: **barato** (menos de 500), **razonable** (entre 500 y 2000), **costoso** (entre 2000 y 5000), y

exclusivo (más de 5000). Se utilizó la función `np.select` para asignar estos valores a una nueva columna llamada `price_range`. Posteriormente, esta columna categórica fue convertida en valores numéricos mediante `LabelEncoder`, creando una nueva columna `price_rng_Cat`.

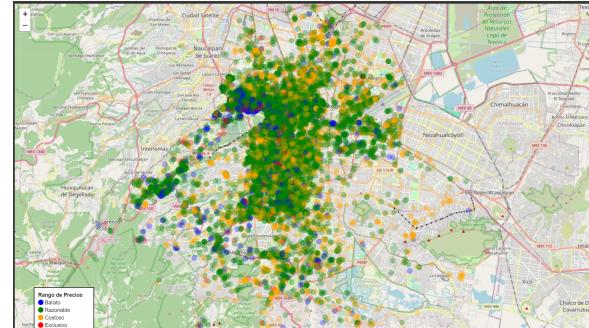


Fig. 14. Clasificación por rango de precios

Un mapa que visualiza las clasificaciones de precios de Airbnb ofrece una interpretación clara de cómo se distribuyen las categorías de precios en diferentes áreas geográficas. Esta representación espacial permite observar cómo varían las clasificaciones de precios en función de la ubicación, facilitando la identificación de zonas con precios predominantemente económicos, razonables, costosos o exclusivos.

Para construir el **modelo de clasificación KNN**, se eliminó cualquier fila con valores `Nan` y se prepararon los datos dividiéndolos en conjuntos de entrenamiento y prueba. Las características fueron estandarizadas utilizando `StandardScaler` para asegurar que todas las variables estuvieran en la misma escala.

El modelo KNN fue ajustado con los datos de entrenamiento utilizando 3 vecinos (`n_neighbors=3`), y luego se realizaron predicciones con los datos de prueba [3].

Table 4. Comparación de valores reales y predichos

Id	Actual	Predicted
0	0	0
1	4	4
2	4	4
3	4	4
4	0	0
...
4632	4	4
4633	0	0
4634	4	4
4635	4	4
4636	4	4

En esta matriz, las filas representan las categorías

reales (barato, razonable, costoso, exclusivo), y las columnas representan las predicciones hechas por el modelo para estas categorías. Cada celda contiene el número de instancias que pertenecen a la clase indicada en la fila y que fueron predichas como la clase indicada en la columna. Los valores más altos en la diagonal de la matriz indican las predicciones correctas, mientras que los valores fuera de la diagonal representan errores de clasificación.

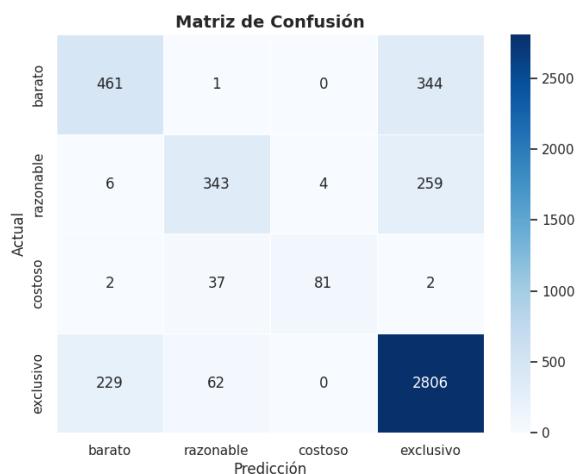


Fig. 15. Matriz de confusión

El modelo KNN muestra un buen desempeño en la predicción de la clase 'exclusivo', con 2806 predicciones correctas y pocas confusiones con otras clases. Sin embargo, hay muchas confusiones entre las clases 'barato', 'razonable', y 'exclusivo', lo que sugiere que el modelo tiene dificultades para diferenciarlas claramente. La clase 'costoso' también muestra algunas confusiones, especialmente con la clase 'razonable'. Estos resultados indican que el modelo puede necesitar ajustes adicionales o características más discriminativas para mejorar su precisión en estas categorías.

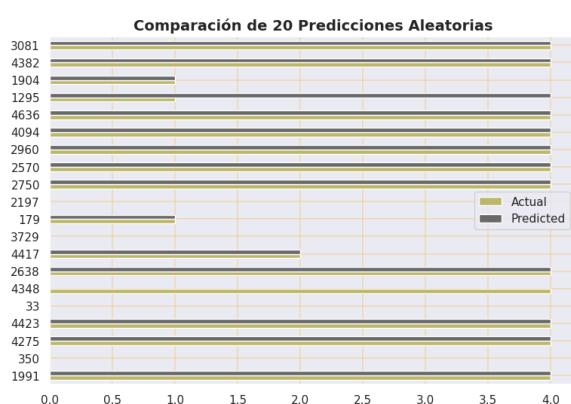


Fig. 16. Predicciones

Las comparaciones muestran que las barras amarillas claras y gris oscuro coinciden para las predicciones correctas, mientras que las discrepancias indican errores. Instancias como 3081, 4382 y 1904 son clasificadas correctamente, mientras que 1295, 2197 y 3729 presentan errores. Aunque el modelo KNN tiene un desempeño razonable, aún puede mejorar.

El gráfico de barras revela que la categoría 'razonable' tiene la mayor cantidad de propiedades, superando los 15,000, seguida por 'barato' con cerca de 4,000. Las categorías 'no definido' y 'costoso' tienen alrededor de 3,500 propiedades cada una, mientras que 'exclusivo' cuenta con menos de 700, reflejando la distribución de precios en el conjunto de datos.

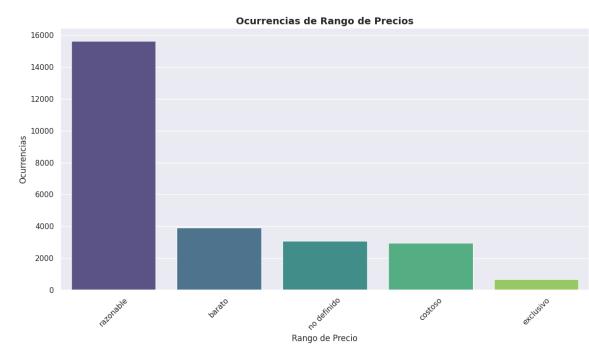


Fig. 17. Frecuencia de categorías

El siguiente gráfico revela que los precios para las categorías 'barato', 'razonable' y 'costoso' son bastante similares, mostrando poca variación entre estos rangos. Sin embargo, la categoría 'exclusivo' se destaca significativamente, con precios que alcanzan alrededor de 40,000. Esto indica que, aunque los precios en las categorías más comunes son relativamente homogéneos, los precios en la categoría 'exclusivo' presentan una diferencia considerable, reflejando una alta variabilidad y valores mucho más elevados en comparación con las demás categorías.



Fig. 18. Relacion entre categorías y precio

4. Resultados

La tabla muestra los resultados de las métricas de evaluación para los modelos de regresión:

Table 5. Comparación de resultados entre modelos de regresión

Métrica	Lineal	Lineal Múltiple
MAE	96.2408	898.5457
MSE	12020.6596	16153599.0411
RMSE	109.6387	4019.1540

Estos resultados muestran que el modelo de regresión lineal múltiple presenta errores mayores en comparación con el modelo de regresión lineal simple, lo que sugiere que el modelo más complejo no mejora la precisión de las predicciones de manera efectiva en este caso.

Mientras que para el modelo KNN observamos la gráfica sugiere que valores bajos de k pueden causar sobreajuste y reducir la precisión, mientras que un k ligeramente mayor ofrece una precisión más estable. Un $k = 3$ proporciona un equilibrio óptimo entre ajuste y generalización, resultando en un rendimiento más confiable y menos susceptible a errores.

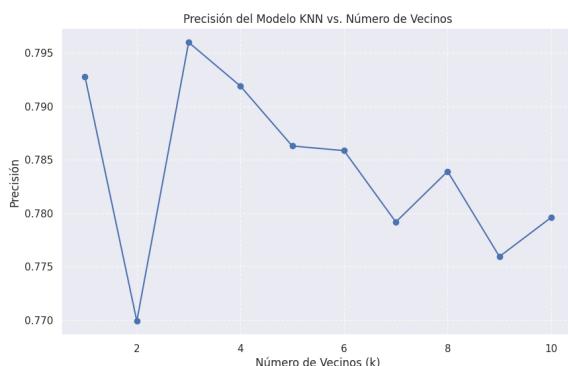


Fig. 19. Precisión para diferentes valores de k

La precisión del modelo es aproximadamente el 79.60 % de las predicciones positivas son correctas. El F1-score refleja un equilibrio entre precisión y recall, siendo 78.89 % la medida combinada de la exactitud de las predicciones y la capacidad de capturar todos los positivos relevantes. A continuación, se muestra la tabla con la especificidad para cada clase:

Table 6. Especificidad para cada clase

Clase	Especificidad
Barato	0.9381
Razonable	0.9752
Costoso	0.9991
Exclusivo	0.6071

El modelo muestra un alto rendimiento en la identificación de la mayoría de las clases, aunque enfrenta dificultades significativas con una categoría específica, lo que sugiere áreas para mejorar en la detección de esa clase menos precisa.

5. Conclusiones

El análisis de técnicas de aprendizaje supervisado para predecir precios de propiedades en Airbnb en la Ciudad de México revela *insights* clave sobre la efectividad de los modelos evaluados. A pesar de la mayor complejidad del modelo de regresión lineal múltiple, sus errores superiores en comparación con el modelo de regresión lineal simple sugieren que la complejidad adicional no siempre contribuye a una mejora en la precisión predictiva.

En contraste, el modelo K-Nearest Neighbors (KNN) destaca por su capacidad para ofrecer una precisión más estable mediante la selección adecuada del parámetro k . Los resultados muestran que un k con valor 3 proporciona un equilibrio óptimo, reduciendo el riesgo de sobreajuste y mejorando la confiabilidad del modelo. La alta precisión y el F1-score del modelo KNN, junto con la variabilidad en la especificidad de las clases, indican que este enfoque puede ser más efectivo para capturar la variabilidad en los datos de precios.

En resumen, los hallazgos subrayan la importancia de ajustar los parámetros del modelo y seleccionar técnicas adecuadas para optimizar las predicciones en un mercado dinámico. La combinación de una técnica de modelado bien ajustada y una comprensión detallada de los datos puede mejorar significativamente la precisión en la predicción de precios en el mercado de alquileres a corto plazo.

Referencias

- [1] Wackerly, D. D., Mendenhall, W. III, & Scheaffer, R. L., *Estadística matemática con aplicaciones*, Séptima Edición, Cengage Learning, 2010.
- [2] Nasiriany, S., Thomas, G., Wang, W., & Yang, A., *A Comprehensive Guide to Machine Learning*, Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, 2019.
- [3] James, G., Witten, D., Hastie, T., Tibshirani, R., & Taylor, J., *An Introduction to Statistical Learning with Applications in Python*, Springer Texts in Statistics, Springer Nature Switzerland AG, 2023.
- [4] Bruce, P., Bruce, A., & Gedeck, P., *Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python*, Second Edition, O'Reilly Media, Inc., Sebastopol, CA, 2020.