



Tecnológico de Monterrey

Actividad 2: Regresión No lineal

María Fernanda Robles Soto - A01736552

06 de Octubre del 2025

Gestión de proyectos de plataformas tecnológicas

Reporte

Introducción

A continuación, presentaré el reporte correspondiente al código desarrollado para la Actividad 2: Regresión No Lineal. En esta actividad se realizará un análisis de correlación para cada variable y, posteriormente, generar dos modelos de regresión no lineal los cuales compararé con los obtenidos en el modelo de Regresión Lineal hecho en la Actividad 1: Regresión Simple y Múltiple. Finalmente, se presentará una tabla comparativa con los coeficientes de determinación y correlación obtenidos en cada caso.

Carga y limpieza de los datos (Valores Nulos y NA)

Como primer paso, cargué las librerías necesarias y la base de datos correspondiente a Brasil, la misma base que usé para la actividad anterior.

Para no modificar la base original, creé una copia de trabajo sobre la cual realicé todo el preprocesamiento. En esta parte, primero verifiqué los valores nulos dentro de la base de datos, luego estandaricé los nombres de las columnas, limpié los datos y convertí los valores porcentuales a tipo float. También transformé las variables booleanas y, en los casos donde no fue posible hacerlo, las dejé como tipo object. Posteriormente, separé las columnas numéricas de las categóricas. En lugar de eliminar los valores nulos o NA, preferí imputarlos usando la media para las variables numéricas y la moda para las categóricas. En cuanto al tratamiento de outliers, apliqué el método de los cuartiles (IQR) para identificarlos y analizarlos. Finalmente, realicé una visualización general de la base ya limpia para confirmar que todas las transformaciones se aplicaron correctamente.

In [12]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import seaborn as sns
import matplotlib.pyplot as plt
import scipy.special as special
from scipy.optimize import curve_fit
import seaborn as sns
from sklearn.metrics import r2_score
```

In [13]:

```
#Cargar archivo csv desde seaborn
df_original = pd.read_csv("/content/listings.csv", on_bad_lines='skip', engine
```

Limpieza de los datos y exploración inicial

In []:

```
#verificamos nulos
df_original.isnull().sum()
```

```

1: # Creamos una copia de La base de datos original para hacer La limpieza de Los datos
df = df_original.copy()

# Se estandarizan nombres de columnas
df.columns = df.columns.str.strip().str.lower().str.replace(' ', '_')

# Limpiar y convertir columnas
for col in df.columns:
    if df[col].dtype == 'object':
        # Convertir porcentajes a float
        pct_mask = df[col].astype(str).str.endswith('%')
        if pct_mask.sum() / len(df[col]) > 0.1:
            df[col] = (
                df[col].str.replace('%', '', regex=False)
                .replace('', np.nan)
                .astype(float) / 100
            )
        else:
            # Convertir booleanos 't'/'f' a 1/0
            if df[col].isin(['t', 'f']).all():
                df[col] = df[col].map({'t': 1, 'f': 0})
            else:
                # Convertir números con símbolos a float
                df[col] = (
                    df[col].str.replace('$', '', regex=False)
                    .str.replace(',', '', regex=False)
                    .replace('', np.nan)
                )
            # Intentar convertir a float si es posible
            try:
                df[col] = df[col].astype(float)
            except:
                pass # deja como object si no se puede

# Separar columnas numéricas y categóricas
num_cols = df.select_dtypes(include=np.number).columns
cat_cols = df.select_dtypes(exclude=np.number).columns

# Se hace La imputación de nulos
for col in num_cols:
    df[col].fillna(df[col].median(), inplace=True)

for col in cat_cols:
    if not df[col].mode().empty:
        df[col].fillna(df[col].mode()[0], inplace=True)

# Tratamiento de outliers
for col in num_cols:
    Q1 = df[col].quantile(0.25)
    Q3 = df[col].quantile(0.75)
    IQR = Q3 - Q1
    if IQR > 0:
        lower_bound = Q1 - 1.5 * IQR
        upper_bound = Q3 + 1.5 * IQR
        df[col] = np.clip(df[col], lower_bound, upper_bound)

df

```

Regresión No Lineal

Para poder realizar los modelos, lo primero que hice fue crear una tabla de correlación para cada una de las variables correspondientes, que son: `host_response_rate`, `host_acceptance_rate`, `host_total_listings_count`, `accommodates`, `reviews_per_month` y `price`.

- `host_reponse_rate`

Lo primero que hice fue asegurarme de que se estuviera seleccionando una variable numérica, para luego poder calcular la correlación de la misma, comenzando con la variable `host_response_rate` y continuando con las demás. El objetivo era identificar las variables que presentan una correlación alta con la variable dependiente y, a partir de eso, construir los modelos de regresión no lineal.

REGRESIÓN NO LINEAL

Modelos y correlación entre las variables

`host_response_rate`

```
4]: # host_response_rate

# Seleccionar solo columnas numéricas
num_cols = df.select_dtypes(include=np.number).columns

# Correlación de review_scores_rating con todas las numéricas
corr_review = df[num_cols].corrwith(df["host_response_rate"]).sort_values(ascending=False)

print(corr_review)
```

<code>host_response_rate</code>	1.000000
<code>host_acceptance_rate</code>	0.421313
<code>estimated_occupancy_l365d</code>	0.195262
<code>number_of_reviews_ltm</code>	0.193770
<code>estimated_revenue_l365d</code>	0.188352
<code>number_of_reviews_ly</code>	0.174438
<code>reviews_per_month</code>	0.164886
<code>number_of_reviews</code>	0.163727
<code>number_of_reviews_l30d</code>	0.127038
<code>instant_bookable</code>	0.094294
<code>maximum_minimum_nights</code>	0.071804
<code>calculated_host_listings_count_entire_homes</code>	0.066943
<code>review_scores_cleanliness</code>	0.062423
<code>review_scores_value</code>	0.053861
<code>review_scores_communication</code>	0.051767
<code>review_scores_rating</code>	0.050761
<code>review_scores_accuracy</code>	0.047373
<code>host_listings_count</code>	0.038546
<code>calculated_host_listings_count</code>	0.037065
<code>price</code>	0.036961
<code>minimum_nights_avg_ntm</code>	0.034141
<code>review_scores_checkin</code>	0.027451
<code>minimum_nights</code>	0.022796
<code>accommodates</code>	0.021445
<code>beds</code>	0.013211
<code>host_total_listings_count</code>	0.013023
<code>minimum_minimum_nights</code>	0.009926
<code>review_scores_location</code>	0.005333
<code>id</code>	0.002301
<code>bathrooms</code>	-0.003790
<code>maximum_maximum_nights</code>	-0.011041
<code>longitude</code>	-0.013893
<code>maximum_nights_avg_ntm</code>	-0.016288
<code>bedrooms</code>	-0.017630
<code>minimum_maximum_nights</code>	-0.023178
<code>host_id</code>	-0.031436
<code>calculated_host_listings_count_shared_rooms</code>	-0.035364

Modelo 1:

Con eso, después de declarar la variable independiente y dependiente, comencé a definir el primer modelo. En esta parte, establecí una función cuadrática y ajusté sus parámetros con para obtener los valores de a , b y c que mejor se adaptaran a los datos. Con esos parámetros realicé las predicciones, mostré las primeras diez y grafiqué los datos originales junto con las predicciones para visualizar el ajuste del modelo. Finalmente, obtuve los coeficientes ajustados y calculé el R^2 y el R , que muestran qué tan bien explica el modelo la relación entre ambas variables.

Modelo 1

```
28]: # Variables independientes y dependiente
x = df['estimated_occupancy_1365d'].values
y = df['host_response_rate'].values

# Definimos la función cuadrática
def func1(x, a, b, c):
    return a*x**2 + b*x + c

# Ajustamos los parámetros de la función
parametros1, covs = curve_fit(func1, x, y)

# Realizamos las predicciones
y_pred1 = func1(x, *parametros1)

# Mostramos las primeras predicciones
print("Primeras 10 predicciones:", y_pred1[:10])

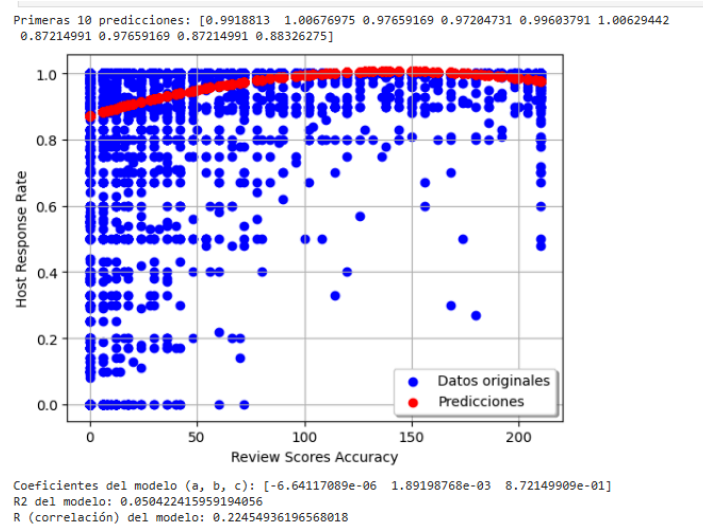
# Graficamos los datos originales y las predicciones
plt.scatter(x, y, color='blue', label="Datos originales")
plt.scatter(x, y_pred1, color='red', label="Predicciones")
plt.xlabel('Review Scores Accuracy')
plt.ylabel('Host Response Rate')
plt.legend(loc='best', fancybox=True, shadow=True)
plt.grid(True)
plt.show()

# Mostramos los coeficientes ajustados
print("Coeficientes del modelo (a, b, c):", parametros1)

# Calculamos el coeficiente de determinación y correlación
R2_Modelo1 = r2_score(y, y_pred1)
R_Modelo1 = np.sqrt(R2_Modelo1)

print("R2 del modelo:", R2_Modelo1)
print("R (correlación) del modelo:", R_Modelo1)
```

En este primer modelo, los valores predichos se mantienen muy cercanos entre sí, con resultados alrededor de 0.87 a 1.00, lo que indica que el modelo genera predicciones bastante estables, aunque con poca variación. Los coeficientes obtenidos muestran que la relación cuadrática es muy pequeña, por lo que la curva tiene una forma casi lineal, y el término lineal junto con el intercepto son los que más influyen en el resultado.



El valor de R2 indica que el modelo sólo explica alrededor del 5% de la variabilidad de la variable dependiente, lo que significa que el ajuste no es muy fuerte. De igual forma, el coeficiente de correlación sugiere una relación débil y positiva entre las variables analizadas.

Modelo 2:

En esta parte, se estableció una función no lineal diferente, con el objetivo de probar un tipo de relación más compleja entre las variables, después, se ajustaron los parámetros de la función, obteniendo los valores de a, b y c que mejor se adaptan a los datos. Con esos parámetros, se realizaron las predicciones y se mostraron las primeras diez para verificar su comportamiento.

Después, se graficaron los datos originales junto con las predicciones, usando azul para los valores reales y rojo para los estimados, con el fin de visualizar qué tan bien el modelo logra representar la tendencia de los datos. Finalmente, se calculó el coeficiente de determinación y el coeficiente de correlación para evaluar el nivel de ajuste y la fuerza de la relación entre las variables.

Modelo 2

In [38]:

```
# Variables independientes y dependiente
x = df['estimated_occupancy_l365d'].values
y = df['host_response_rate'].values

# Definimos la función correctamente
def func2(x, a, b, c):
    return (a*x**2 + b) / (c*x**2)

# Ajustamos los parámetros de la función
parametros2, covs2 = curve_fit(func2, x, y)

# Mostramos los coeficientes ajustados
print("Coeficientes del modelo (a, b, c):", parametros2)

# Realizamos las predicciones
y_pred2 = func2(x, *parametros2)
print("Primeras 10 predicciones:", y_pred2[:10])

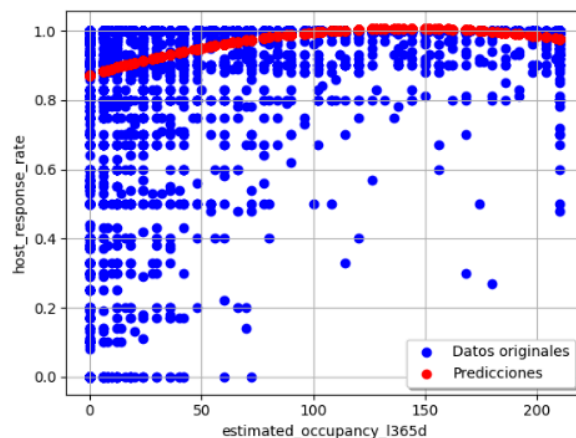
# Graficamos los datos originales y las predicciones
plt.scatter(x, y, color='blue', label="Datos originales")
plt.scatter(x, y_pred2, color='red', label="Predicciones")
plt.xlabel('Speeding')
plt.ylabel('Total')
plt.legend(loc='best', fancybox=True, shadow=True)
plt.grid(True)
plt.show()

# Calculamos el coeficiente de determinación y correlación
R2_Modelo2 = r2_score(y, y_pred2)
R_Modelo2 = np.sqrt(abs(R2_Modelo2)) # usamos abs por si R^2 es negativo

print("R^2 del modelo:", R2_Modelo2)
print("R (correlación) del modelo:", R_Modelo2)
```

En este modelo, los coeficientes indican que la parte cuadrática tiene un efecto muy pequeño, mientras que el término lineal y el intercepto son los que más influyen en las predicciones. Las primeras diez predicciones muestran valores cercanos a 0.87–1.01, lo que indica que el modelo genera resultados bastante estables, pero con poca variación. El $R^2 = 0.05$ significa que solo explica alrededor del 5% de la variabilidad de la variable dependiente, por lo que el ajuste es muy débil.

```
Coeficientes del modelo (a, b, c): [-6.64117089e-06  1.89198768e-03  8.72149909e-01]
Primeras 10 predicciones: [0.9918813  1.00676975  0.97659169  0.97284731  0.99603791  1.00629442
 0.87214991  0.97659169  0.87214991  0.88326275]
```



```
R^2 del modelo: 0.050422415959194056
R (correlación) del modelo: 0.22454936196568018
```

De manera similar, el coeficiente de correlación ($R = 0.22$) indica una relación débil y positiva entre las variables analizadas. En conjunto, esto sugiere que el modelo captura muy poca de la relación real entre `estimated_occupancy_l365d` y `host_response_rate`.

- `host_acceptance_rate`

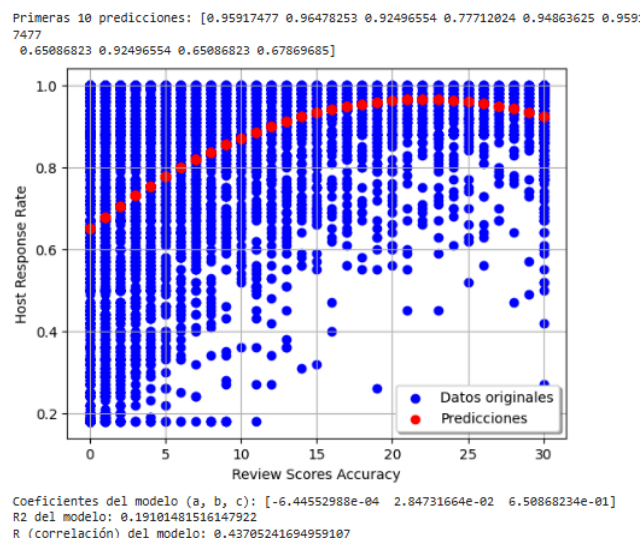
Siguiendo la misma metodología que con la primera variable, se siguió desarrollando el análisis.

host_acceptance_rate	1.000000
host_response_rate	0.421313
number_of_reviews_ltm	0.397111
estimated_occupancy_l365d	0.390775
number of reviews 1y	0.379109

Seleccionamos la variable, `number_of_reviews_ltm`, ya que, para no repetir las mismas variables después en otros modelos.

Modelo 1:

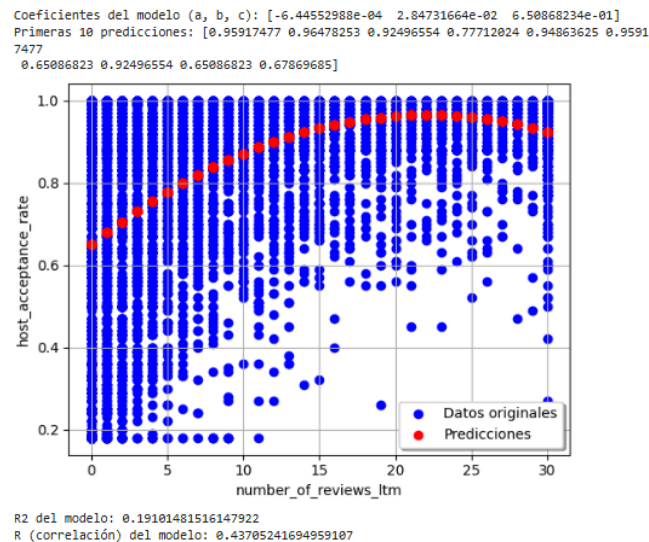
En este modelo, las primeras diez predicciones muestran valores más variados, desde 0.65 hasta 0.96, lo que indica que el modelo logra capturar algo más de variación en los datos en comparación con el modelo anterior. Los coeficientes muestran que la parte cuadrática sigue siendo pequeña, pero el término lineal y el intercepto tienen un mayor peso en las predicciones, afectando más la forma de la curva.



El R^2 indica que este modelo explica aproximadamente el 19% de la variabilidad de la variable dependiente, lo que es una mejora respecto al 5% del modelo anterior. Asimismo, el coeficiente de correlación es 0.44 sugiere una relación moderada y positiva entre las variables.

Modelo 2:

En este segundo modelo, los coeficientes muestran que la parte cuadrática sigue siendo pequeña, mientras que el término lineal y el intercepto tienen mayor peso en las predicciones. Las primeras diez predicciones varían más, desde 0.65 hasta 0.96, lo que indica que el modelo logra capturar mejor la variación de los datos en comparación con el primer modelo.



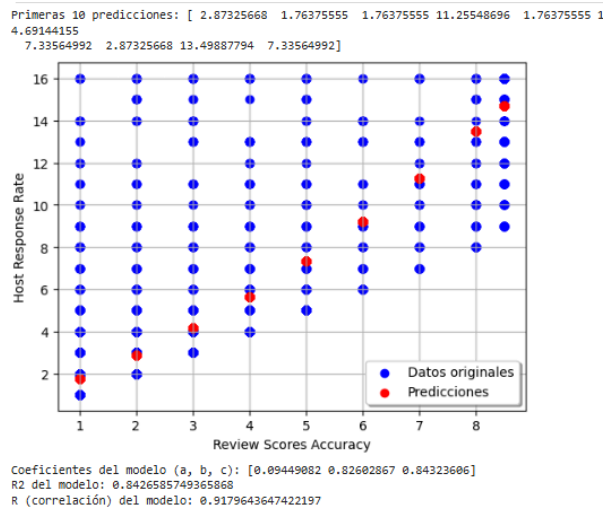
El R2 indica que el modelo explica alrededor del 19% de la variabilidad de la variable dependiente, mientras que el coeficiente de correlación R (0.44) indica una relación moderada y positiva entre las variables.

- host_total_listings_count

host_total_listings_count	1.000000
host_listings_count	0.913513
calculated host listings count	0.867429

Modelo 1:

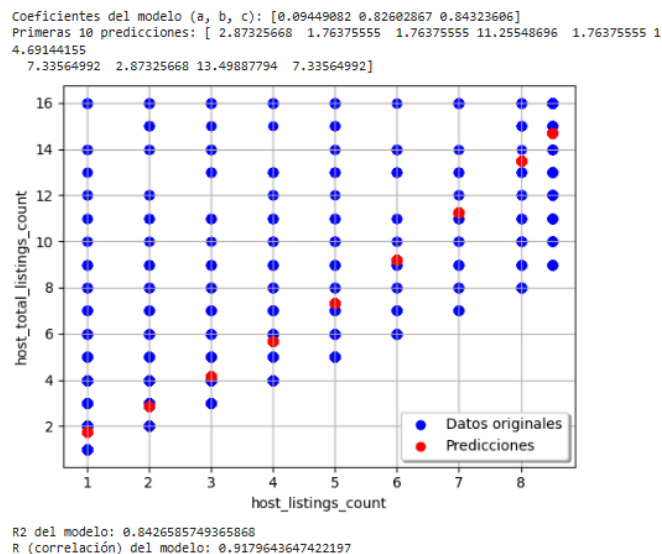
Las primeras diez predicciones muestran una mayor variación, desde 1.76 hasta 14.69, lo que indica que el modelo captura de manera efectiva la diversidad de los datos. Los coeficientes muestran que la parte cuadrática ahora tiene un efecto significativo, junto con el término lineal y el intercepto, lo que permite que la curva se ajuste mucho mejor a la tendencia real de los datos.



El R2 en 0.84 indica que el modelo explica aproximadamente el 84% de la variabilidad de la variable dependiente, mientras que el coeficiente de correlación $R \approx 0.92$ refleja una relación fuerte y positiva entre las variables.

Modelo 2:

Las primeras diez predicciones muestran valores que van desde 1.76 hasta 14.69, lo que indica que el modelo logra capturar de manera efectiva la variación de los datos. Los coeficientes muestran que la parte cuadrática tiene un efecto importante, mientras que el término lineal y el intercepto también contribuyen significativamente a las predicciones.



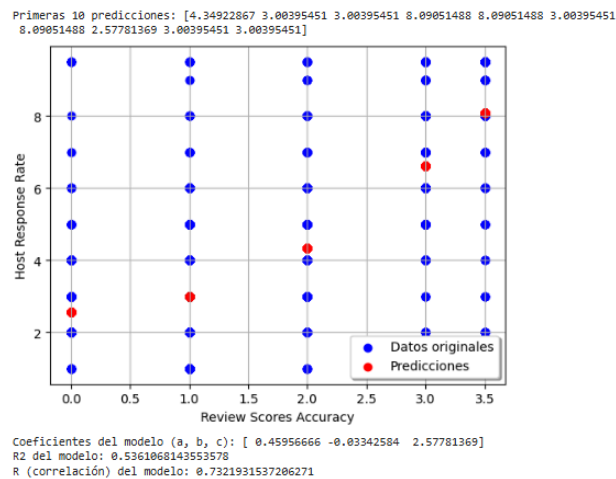
El R2 indica que el modelo explica aproximadamente el 84% de la variabilidad de la variable dependiente, y el coeficiente de correlación refleja una relación fuerte y positiva entre las variables. En conjunto, esto muestra que el modelo tiene un ajuste muy sólido y captura con precisión la relación entre las variables analizadas.

- accommodates

accommodates	1.000000
beds	0.715223
bedrooms	0.705331
bathrooms	0.569926

Modelo 1:

En este primer modelo, las primeras diez predicciones muestran valores entre 2.58 y 8.09, lo que indica que el modelo logra capturar parte de la variación de los datos. Los coeficientes muestran que la parte cuadrática tiene un efecto significativo, mientras que el término lineal y el intercepto también influyen en las predicciones.

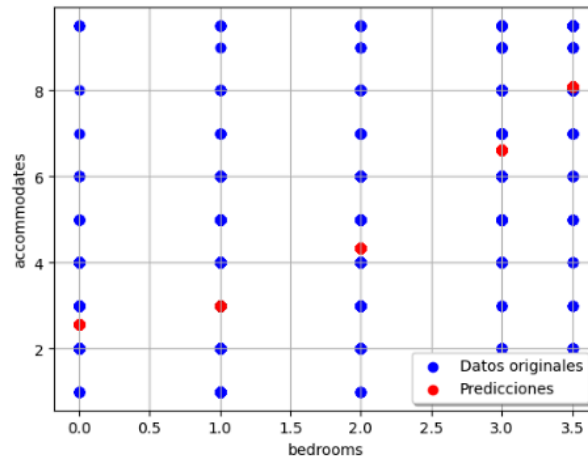


El R2 0.54 indica que el modelo explica aproximadamente el 54% de la variabilidad de la variable dependiente, y el coeficiente de correlación de 0.73 refleja una relación fuerte y positiva entre las variables. Esto sugiere que el modelo tiene un ajuste sólido y logra representar bastante bien la tendencia de los datos.

Modelo 2:

En este modelo, las primeras diez predicciones varían entre 2.58 y 8.09, mostrando que el modelo logra capturar bastante bien la variabilidad de los datos. Los coeficientes indican que la parte cuadrática tiene un efecto importante, mientras que el término lineal y el intercepto también contribuyen al ajuste.

Coefficientes del modelo (a, b, c): [0.45956666 -0.03342584 2.57781369]
 Primeras 10 predicciones: [4.34922867 3.00395451 3.00395451 8.09051488 8.09051488 3.00395451
 8.09051488 2.57781369 3.00395451 3.00395451]



R2 del modelo: 0.5361068143553578
 R (correlación) del modelo: 0.7321931537206271

El R2 de 0.54 indica que el modelo explica alrededor del 54% de la variabilidad de la variable dependiente, y el coeficiente de correlación en 0.73 refleja una relación fuerte y positiva entre las variables, lo que sugiere un ajuste bastante sólido.

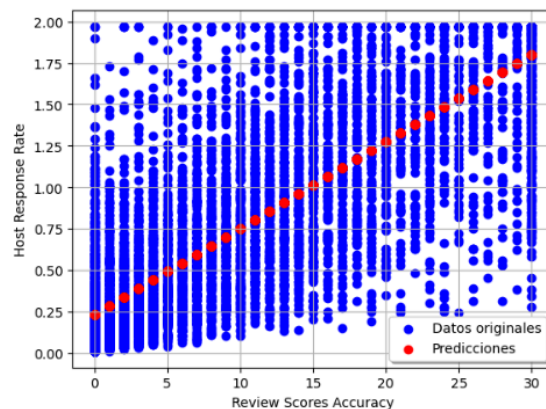
- reviews_per_month

reviews_per_month	1.000000
number_of_reviews	0.887192
number_of_reviews_ly	0.857735
number_of_reviews_ltm	0.848981
estimated_occupancy_1365d	0.815719
estimated_revenue_1365d	0.800000

Modelo 1:

En este modelo, las primeras diez predicciones varían entre 0.23 y 1.80, mostrando que el modelo logra capturar bastante bien la variabilidad de los datos. Los coeficientes indican que la parte cuadrática tiene un efecto muy pequeño, mientras que el término lineal y el intercepto son los que más influyen en las predicciones.

Primeras 10 predicciones: [1.22070325 1.43069444 1.799911 0.49140237 1.11597757 1.22070325
 0.23307462 1.799911 0.23307462 0.2846502]

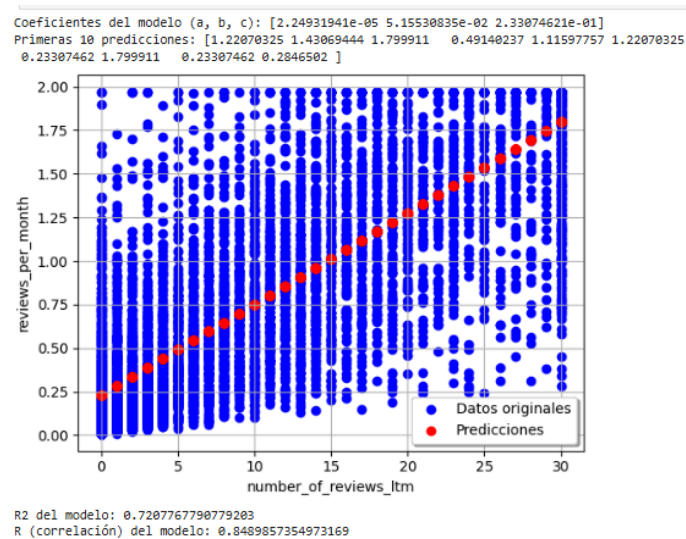


Coefficientes del modelo (a, b, c): [2.24931941e-05 5.15530835e-02 2.33074621e-01]
 R2 del modelo: 0.7207767790779203
 R (correlación) del modelo: 0.8489857354973169

El R^2 es de 0.72, lo que indica que el modelo explica aproximadamente el 72% de la variabilidad de la variable dependiente, mientras que el coeficiente de correlación que es de 0.85 refleja una relación fuerte y positiva entre las variables. En conjunto, esto muestra que el modelo tiene un ajuste sólido y predicciones confiables para esta nueva variable.

Modelo 2:

En este modelo, los coeficientes obtenidos muestran que la parte cuadrática tiene un efecto muy pequeño, mientras que el término lineal y el intercepto son los que más influyen en las predicciones.



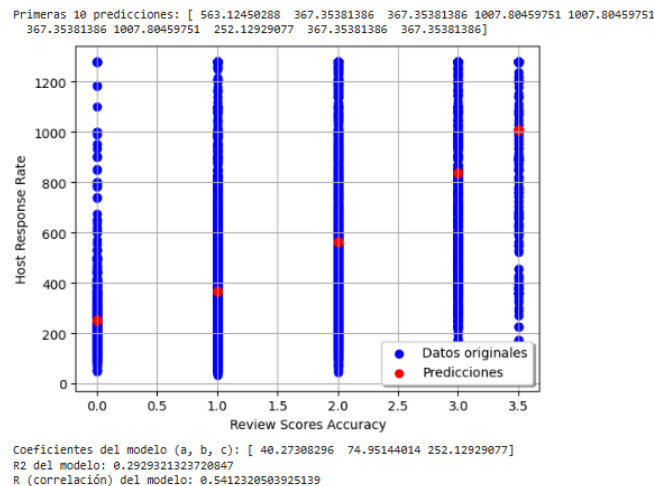
El modelo explica alrededor del 72% de la variabilidad de la variable dependiente, lo cual representa un ajuste fuerte y consistente. Además, el coeficiente de correlación $R \approx 0.85$ refleja una relación positiva y alta entre las variables analizadas. En conjunto, estos resultados muestran que el modelo logra representar adecuadamente la tendencia de los datos y describe de forma precisa la relación entre ambas variables.

- Price

price	1.000000
bathrooms	0.551633
bedrooms	0.532629
accommodates	0.470825
beds	0.388487

Modelo 1:

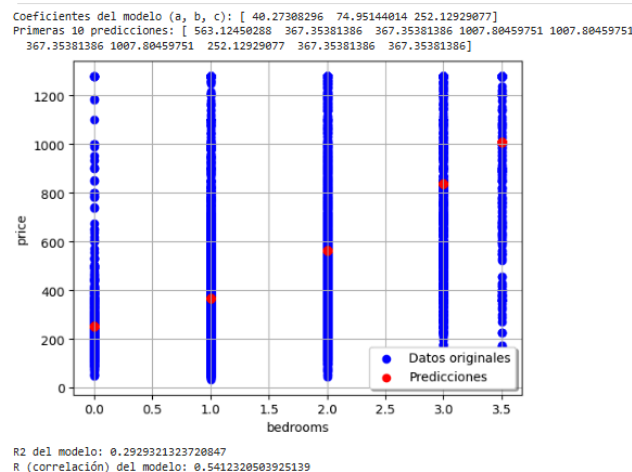
En este modelo, las primeras diez predicciones muestran valores entre 252.13 y 1007.80, lo que indica que el modelo captura cierta variabilidad de los datos, aunque con diferencias bastante marcadas entre las predicciones. Los coeficientes muestran que la parte cuadrática y el término lineal tienen un efecto importante en el ajuste, mientras que el intercepto establece el valor base de las predicciones.



El modelo explica aproximadamente el 29% de la variabilidad de la variable dependiente, mientras que el coeficiente de correlación refleja una relación moderada y positiva entre las variables. Esto sugiere que el modelo tiene un ajuste moderado, capturando solo parte de la relación real entre las variables y que podría requerir ajustes o variables adicionales para mejorar su precisión.

Modelo 2:

En este modelo, las primeras diez predicciones varían entre 252.13 y 1007.80, mostrando que el modelo logra captar parte de la variabilidad de los datos, aunque con diferencias marcadas entre los valores predichos. Los coeficientes indican que la parte cuadrática y el término lineal tienen un efecto relevante en las predicciones, mientras que el intercepto establece el valor base.



Explica solo alrededor del 29% de la variabilidad de la variable dependiente, y el coeficiente de correlación refleja una relación moderada y positiva entre las variables. En conjunto, esto sugiere que el modelo tiene un ajuste moderado, capturando solo parcialmente la relación entre las variables, por lo que podría requerir ajustes adicionales o incluir más variables para mejorar su precisión.

Tabla comparativa:

Variable	Modelo 1 de Regresión No Lineal	Modelo 2 de Regresión No Lineal	Modelo de Regresión Lineal (Actividad 1)
host_response_rate	<p>Coeficientes del modelo (a, b, c): [-6.64117089e-06 1.89198768e-03 8.72149909e-01]</p> <p>R² del modelo: 0.050422415959194056</p> <p>R (correlación) del modelo: 0.22454936196568018</p>	<p>Coeficientes del modelo (a, b, c): [-6.64117089e-06 1.89198768e-03 8.72149909e-01]</p> <p>R² del modelo: 0.050422415959194056</p> <p>R (correlación) del modelo: 0.22454936196568018</p>	<p>Coeficientes obtenidos: [1.52049649e-03 7.24322718e-03 3.53901248e-06]</p> <p>Coeficiente de determinación: 0.23461362874567226</p> <p>Coeficiente de correlación: 0.48436931028469615</p>
host_acceptance_rate	<p>Coeficientes del modelo (a, b, c): [-6.44552988e-04 2.84731664e-02 6.50868234e-01]</p> <p>R² del modelo: 0.19101481516147922</p> <p>R (correlación) del modelo: 0.43705241694959107</p>	<p>Coeficientes del modelo (a, b, c): [-6.44552988e-04 2.84731664e-02 6.50868234e-01]</p> <p>R² del modelo: 0.19101481516147922</p> <p>R (correlación) del modelo: 0.43705241694959107</p>	<p>Coeficientes obtenidos: [4.86696760e-01 5.66022349e-03 2.05235889e-04]</p> <p>Coeficiente de determinación: 0.2710738894241317</p> <p>Coeficiente de correlación: 0.5206475673852051</p>
host_total_listings_count	<p>Coeficientes del modelo (a, b, c): [0.09449082 0.82602867 0.84323606]</p> <p>R² del modelo: 0.8426585749365868</p> <p>R (correlación) del modelo: 0.9179643647422197</p>	<p>Coeficientes del modelo (a, b, c): [0.09449082 0.82602867 0.84323606]</p> <p>R² del modelo: 0.8426585749365868</p> <p>R (correlación) del modelo:</p>	<p>Coeficientes obtenidos: [1.46488753 -0.05708612 0.10720052]</p> <p>Coeficiente de determinación: 0.9219449973765208</p> <p>Coeficiente de correlación:</p>

		0.9179643647422197	0.96017966932055
accommodates	<p>Coeficientes del modelo (a, b, c): [0.45956666 -0.03342584 2.57781369]</p> <p>R2 del modelo: 0.5361068143553578</p> <p>R (correlación) del modelo: 0.7321931537206271</p>	<p>Coeficientes del modelo (a, b, c): [0.45956666 -0.03342584 2.57781369]</p> <p>R2 del modelo: 0.5361068143553578</p> <p>R (correlación) del modelo: 0.7321931537206271</p>	<p>Coeficientes obtenidos: [0.94389345 0.60512636 0.20297077]</p> <p>Coeficiente de determinación: 0.5918716881649602</p> <p>Coeficiente de correlación: 0.7693319752648788</p>
reviews_per_month	<p>Coeficientes del modelo (a, b, c): [2.24931941e-05 5.15530835e-02 2.33074621e-01]</p> <p>R2 del modelo: 0.7207767790779203</p> <p>R (correlación) del modelo: 0.8489857354973169</p>	<p>Coeficientes del modelo (a, b, c): [2.24931941e-05 5.15530835e-02 2.33074621e-01]</p> <p>R2 del modelo: 0.7207767790779203</p> <p>R (correlación) del modelo: 0.8489857354973169</p>	<p>Coeficientes obtenidos: [0.06852936 -0.00300428 0.31616376]</p> <p>Coeficiente de determinación: 0.7047014125868905</p> <p>Coeficiente de correlación: 0.8394649561398561</p>
Price	<p>Coeficientes del modelo (a, b, c): [40.27308296 74.95144014 252.12929077]</p> <p>R2 del modelo: 0.2929321323720847</p> <p>R (correlación) del modelo: 0.5412320503925139</p>	<p>Coeficientes del modelo (a, b, c): [40.27308296 74.95144014 252.12929077]</p> <p>R2 del modelo: 0.2929321323720847</p> <p>R (correlación) del modelo: 0.5412320503925139</p>	<p>Coeficientes obtenidos: [75.50376882 19.63416089 119.79533308]</p> <p>Coeficiente de determinación: 0.3360275947851551</p> <p>Coeficiente de correlación: 0.5796788721224494</p>

Conclusiones:

Después de comparar los resultados obtenidos en los distintos modelos, se puede observar que el nivel de ajuste varía según la variable analizada. En general, tanto los modelos de regresión no lineal como los lineales muestran comportamientos similares, aunque con algunas diferencias importantes en los valores de R^2 y del coeficiente de correlación.

En el caso de `host_response_rate` y `host_acceptance_rate`, los valores de R^2 fueron bajos, lo que indica que estas variables no presentan una relación fuerte con la variable dependiente en ninguno de los modelos. Esto sugiere que otros factores externos podrían estar influyendo más en su comportamiento.

Por otro lado, las variables `host_total_listings_count`, `accommodates` y `reviews_per_month` presentaron los valores más altos de R^2 y de correlación, demostrando una relación más sólida entre las variables y un mejor desempeño de los modelos en la predicción. En especial, `host_total_listings_count` destaca con los mayores valores de correlación, lo cual refleja una relación clara y consistente.

Finalmente, aunque los modelos de regresión no lineal mostraron un desempeño aceptable, el modelo lineal presentó ligeramente mejores valores de R^2 en varias de las variables, lo que indica que el comportamiento de los datos puede explicarse de manera adecuada sin necesidad de recurrir a transformaciones no lineales. En conclusión, ambos tipos de regresión permiten analizar de forma efectiva las relaciones, pero el modelo lineal demuestra ser más estable y fácil de interpretar en este conjunto de datos.