



Tecnológico
de Monterrey

Analítica de Datos y
Herramientas de Inteligencia Artificial

Reporte de actividad 3.1

Profesor: Alfredo García Suárez

Bernardo Quintana López | A01658064

Campus Puebla

11 de abril de 2025

Para comenzar, primero se identificaron valores nulos y valores atípicos, tratando los mismos con diferentes métodos. Primeramente, utilicé el método de mean para las variables cuantitativas y string “Desconocido” para las variables cualitativas. Posteriormente, identifiqué los valores atípicos y los traté con el método de rango intercuartílico.

```

valores_nulos=data.isnull().sum()
valores_nulos

cuantitativas = data.select_dtypes(include=["float64", "int64"])
cualitativas = data.select_dtypes(include=["object"])

✓ 0.0s
Python

cuantitativas = cuantitativas.fillna(cuantitativas.mean())
cualitativas = cualitativas.fillna("Desconocido")

✓ 0.0s
Python

fig = plt.figure(figsize = (20,15))
cuantitativas.plot(kind='box', vert=False)
plt.xlim(0, 600)
plt.title('Valores Atípicos de DataFrame')
plt.yticks(fontsize=7, rotation=0)
plt.show()

✓ 0.3s
Python

```

```

host response rate
host acceptance rate
reviews per month
calculated host listings count
private rooms
calculated_host_listings_count
entire homes
calculated_host_listings_count
Review scores value
review scores location
review scores communication
review scores checkin
review scores cleanliness
review scores accuracy
review scores rating
number of reviews
number of reviews
number of reviews
availability 365
availability 90
availability 30
maximum nights avg_ntm
minimum_nights_avg_ntm
maximum_nights
minimum_nights
maximum_maximum_nights
minimum_minimum_nights
minimum minimum nights
maximum_maximum_nights
minimum_minimum_nights
price
beds
bedrooms
bathrooms
accommodates
host_total_listings_count
host_id
host_id

```

```

y=cuantitativas

percentile25=y.quantile(0.25) #Q1
percentile75=y.quantile(0.75) #Q3
iqr= percentile75 - percentile25

Limite_Superior_iqr= percentile75 + 1.5*iqr
Limite_Inferior_iqr= percentile25 - 1.5*iqr
print("Limite superior permitido", Limite_Superior_iqr)
print("Limite inferior permitido", Limite_Inferior_iqr)

✓ 0.0s
Python

data2_iqr= cuantitativas[(y<=Limite_Superior_iqr)&(y>=Limite_Inferior_iqr)]
data2_iqr

✓ 0.0s
Python

data3_iqr=data2_iqr.copy()
data3_iqr=data2_iqr.fillna(round(data2_iqr.mean(),1))
data3_iqr

✓ 0.0s
Python

```

Después de hacer esto, separé las variables que fueron solicitadas, host_response_rate, host_acceptance_rate, host_total_listings_count, accommodates, reviews_per_month y price.

```

columnas_deseadas = [
    'host_response_rate',
    'host_acceptance_rate',
    'host_total_listings_count',
    'accommodates',
    'reviews_per_month',
    'price',
]

df_filtrado = data[columnas_deseadas]

```

✓ 0.0s

Python

Saqué sus coeficientes de correlación

```

corr_factors = df_filtrado.corr().dropna(how='all', axis=0).dropna(how='all', axis=1)
corr_factors

```

✓ 0.0s

Python

```

corr_factors1 = abs(corr_factors)
corr_factors1

```

✓ 0.0s

Python

Después de esto realicé un heatmap para tener una mejor visualización

```

fig, ax = plt.subplots(figsize=(20, 15))
cax = ax.matshow(corr_factors1, cmap="Blues")
fig.colorbar(cax)

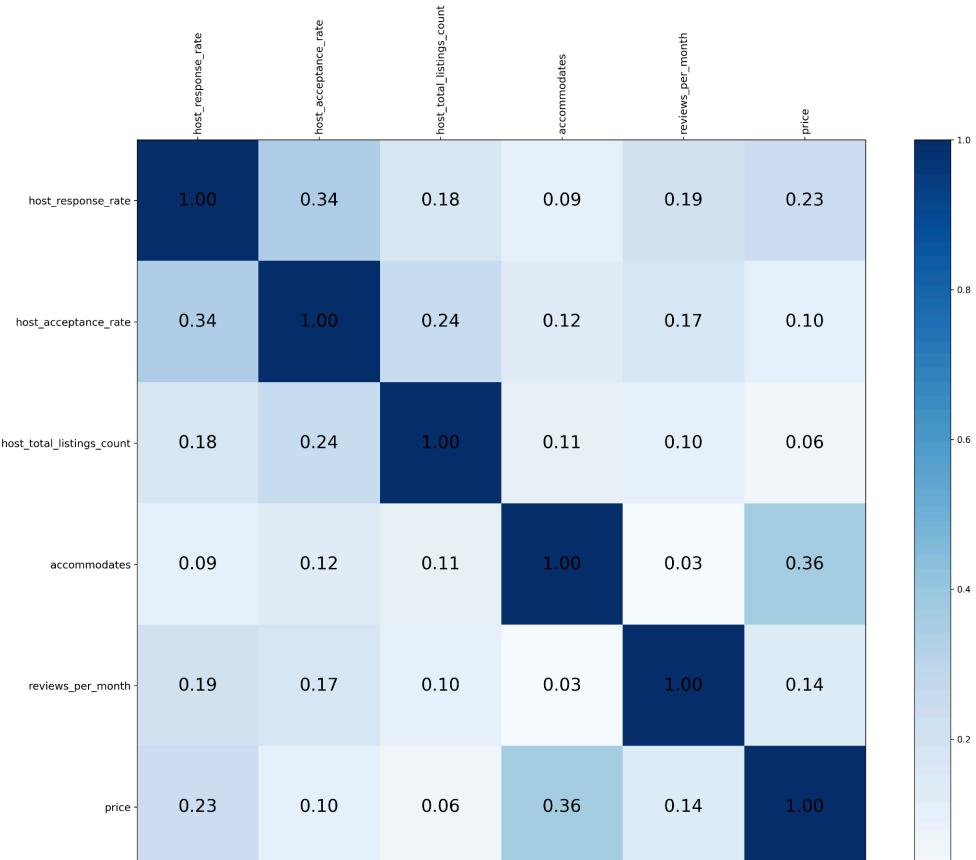
# Añadir anotaciones manualmente
for i in range(corr_factors1.shape[0]):
    for j in range(corr_factors1.shape[1]):
        ax.text(j, i, f'{corr_factors1.iloc[i, j]:.2f}',
                ha="center", va="center", fontsize=20)

plt.xticks(range(len(corr_factors1.columns)), corr_factors1.columns, rotation=90, fontsize=12)
plt.yticks(range(len(corr_factors1.index)), corr_factors1.index, fontsize=12)
plt.savefig('General.png', dpi=300, bbox_inches='tight')
plt.show()

```

✓ 1.1s

Python



Después también visualicé los coeficientes de determinación en un heatmap

```
r_squared_matrix = df_filtrado.corr() ** 2
r_squared_matrix = r_squared_matrix.dropna(how='all', axis=0).dropna(how='all', axis=1)

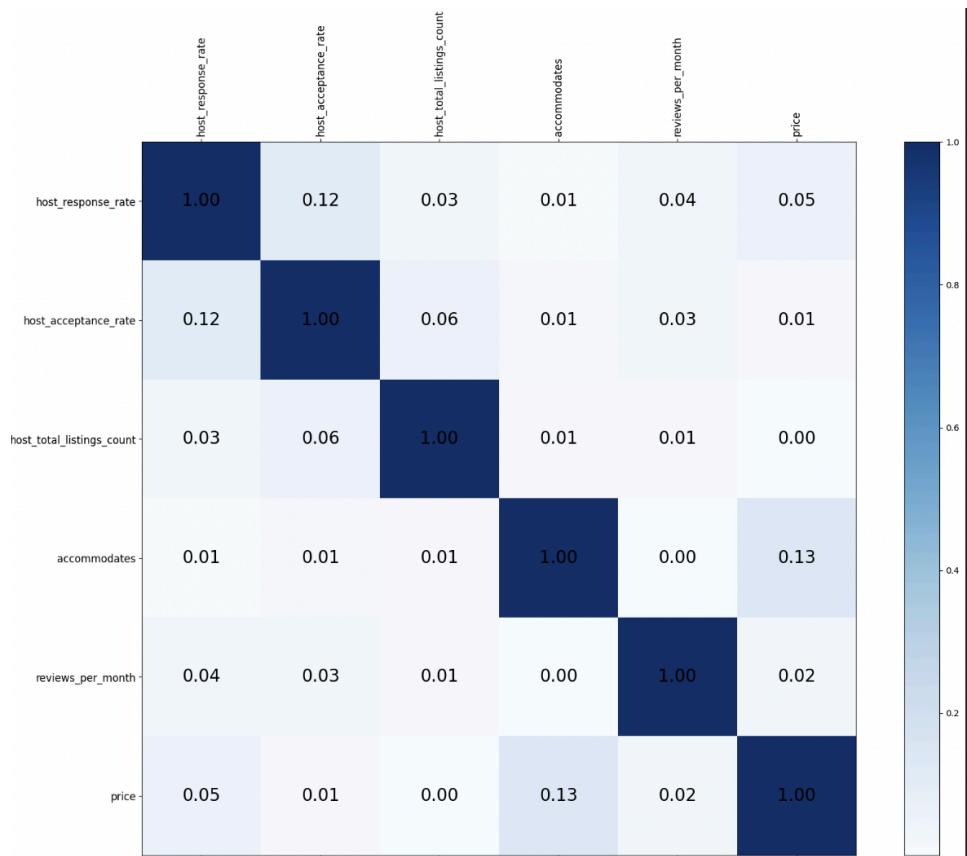
✓ 0.0s

fig, ax = plt.subplots(figsize=(20, 15))
cax = ax.matshow(r_squared_matrix, cmap="Blues")
fig.colorbar(cax)

# Añadir anotaciones manualmente
for i in range(r_squared_matrix.shape[0]):
    for j in range(r_squared_matrix.shape[1]):
        ax.text(j, i, f'{r_squared_matrix.iloc[i, j]:.2f}',
                ha="center", va="center", fontsize=20)

plt.xticks(range(len(r_squared_matrix.columns)), r_squared_matrix.columns, rotation=90, fontsize=12)
plt.yticks(range(len(r_squared_matrix.index)), r_squared_matrix.index, fontsize=12)
plt.show()

✓ 0.2s
```



Después de esto, comencé a utilizar el siguiente código como plantilla solo para modificar las variables a evaluar y el modelo no lineal que utilicé.

```

Vars_Indep= df_filtrado[['host_acceptance_rate']]
Var_Dep= df_filtrado['host_response_rate']

x = Vars_Indep
y = Var_Dep

def func1(x, a, b, c):
    return a*x**2 + b*x + c

parametros, covs= curve_fit(func1, df_filtrado['host_acceptance_rate'], df_filtrado['host_response_rate'])

y_pred = func1(x, *parametros)

plt.plot(x, y, 'bo', label="y-original")
plt.plot(x, y_pred, label="y-predecida")
plt.xlabel('x')
plt.ylabel('y')
plt.legend(loc='best', fancybox=True, shadow=True)
plt.grid(True)
plt.show()

#Calculamos el coeficiente de determinación del modelo
R2_Modelo1 = r2_score(y, y_pred)
print(R2_Modelo1)

#Calculamos el coeficiente de correlación del modelo
R= np.sqrt(R2_Modelo1)
print(R)

```

Python

A continuación se muestran los resultados.

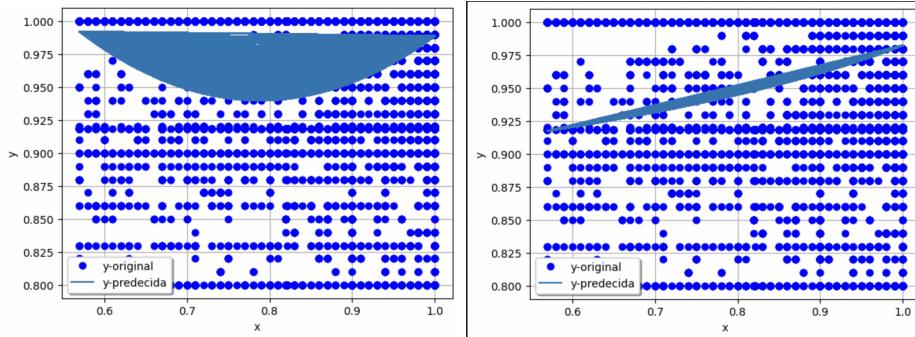
Tabla de coeficientes de determinación y correlación con modelo lineal y no lineal

Variables	Coefficientes	lineales	no lineales
x = host acceptance rate	Coef determinación Coef correlación	lineal 0.12 0.34	cuadrática 0.2 0.44
y = host response rate	Coef determinación Coef correlación	lineal 0.12 0.34	tangencial 0.14 0.38
x = reviews per month y = host acceptance rate	Coef determinación Coef correlación	lineal 0.03 0.17	exponencial 0.05 0.22
x = accommodates y = host acceptance rate	Coef determinación Coef correlación	lineal 0.01 0.12	cuadrática 0.02 0.13
x = host acceptance rate	Coef determinación Coef correlación	lineal 0.06 0.24	cuadrática 0.08 0.28
y = host total listings count	Coef determinación Coef correlación	lineal 0.06 0.24	tangencial 0.06 0.25
x = price	Coef determinación Coef correlación	lineal 0.13 0.36	cuadrática 0.14 0.37
y = accommodates	Coef determinación Coef correlación	lineal 0.13 0.36	logarítmica 0.15 0.38
x = host response rate	Coef determinación Coef correlación	lineal 0.04 0.19	cuadrática 0.05 0.23
y = reviews per month	Coef determinación Coef correlación	lineal 0.04 0.19	cociente/polinomio 0.04 0.2
x = accommodates	Coef determinación Coef correlación	lineal 0.13 0.36	cuadrática 0.14 0.38
y = price	Coef determinación Coef correlación	lineal 0.13 0.36	logarítmica 0.14 0.37

Gráficos por variable objetivo y modelos no lineales utilizados

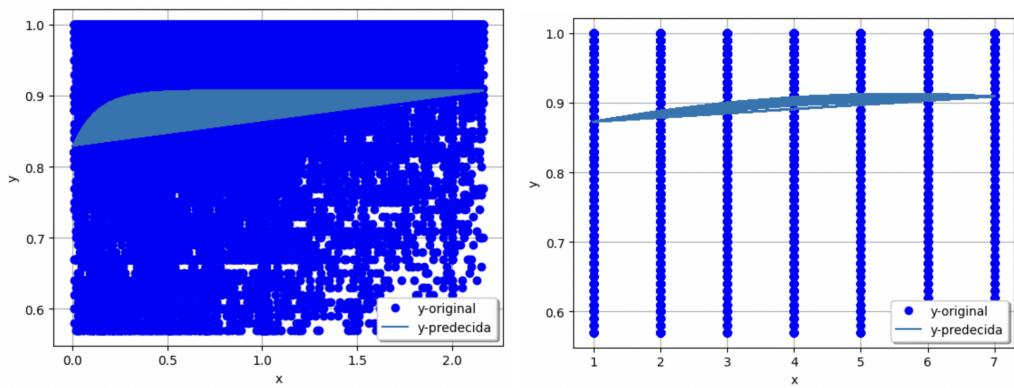
Host response rate

cuadrática | tangencial



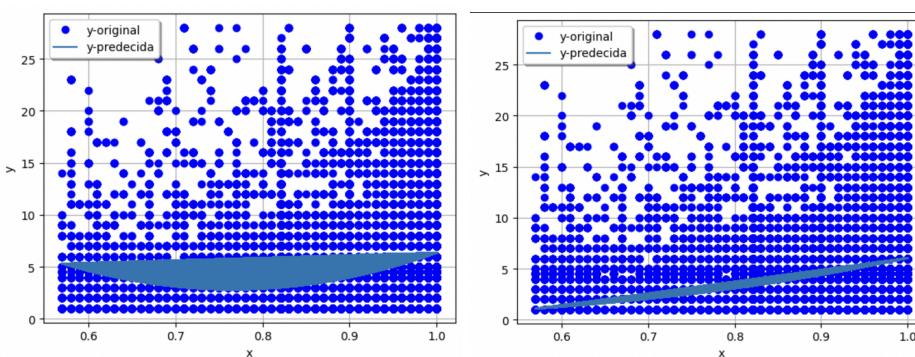
Host acceptance rate

exponencial | cuadrática



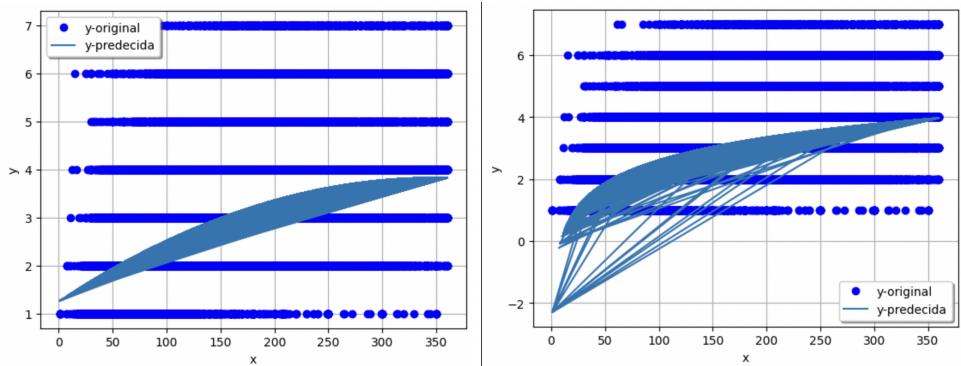
Host total listings count

exponencial | cuadrática



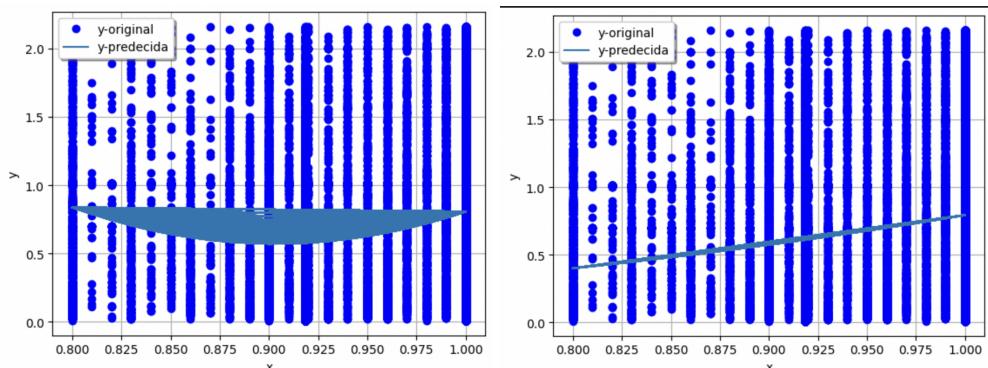
Accommodates

cuadrática | logarítmica



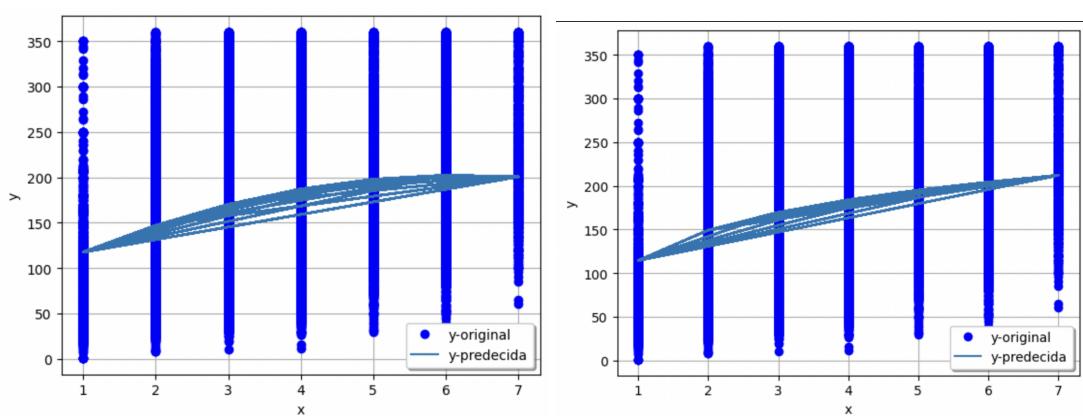
Reviews per month

cuadrática | cociente/polinomio



Price

cuadrática | logarítmica



Los coeficientes de correlación y determinación, tanto lineales como no lineales, revelaron relaciones moderadas entre algunas variables, como la correlación entre el precio y la capacidad de alojamiento (0.36), mientras que otras mostraron asociaciones más débiles. La visualización mediante heatmaps facilitó la interpretación de estas relaciones, destacando patrones que podrían ser útiles para futuros análisis predictivos.

Además, se exploraron modelos no lineales para capturar relaciones más complejas entre las variables. Por ejemplo, el modelo cuadrático mostró una mejora en el coeficiente de determinación para la relación entre la tasa de aceptación del anfitrión y la tasa de respuesta (de 0.12 a 0.2). Sin embargo, en general, las mejoras fueron modestas, lo que sugiere que las relaciones lineales pueden ser suficientes para describir la mayoría de las interacciones entre estas variables. Los gráficos generados respaldan estos hallazgos, mostrando ajustes no lineales que, aunque mejoran ligeramente el modelo, no cambian significativamente las conclusiones.

En conclusión, el análisis revela que, aunque existen correlaciones entre las variables estudiadas, estas son en su mayoría moderadas o débiles. Los modelos no lineales proporcionaron mejoras marginales en la explicación de las relaciones, lo que indica que un enfoque lineal podría ser adecuado para la mayoría de los casos. Estos resultados son valiosos para entender las dinámicas del mercado de Airbnb y pueden servir como base para investigaciones futuras que exploren variables adicionales o técnicas de modelado más avanzadas.