Analítica de Datos y
Herramientas de Inteligencia Artificial

Reporte de actividad 2.1

Profesor**:** Alfredo García Suárez

Bernardo Quintana López | A01658064

Campus Puebla

6 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"])
```

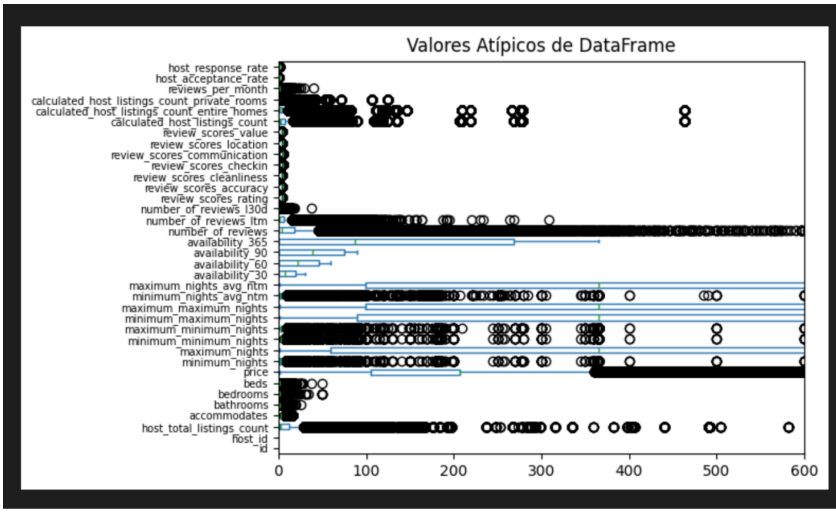
✓ 0.0s Python

```
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




```
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 esto, continué con el procedimiento de analizar las correlaciones de las variables mencionadas por cada tipo de cuarto, así que utilicé el método de frecuencia para hacer de la variables room_type y host_is_superhost numérica.

```python
cat1 = df.groupby(['room_type'])['room_type'].count().sort_values(ascending=False)
cat1
```

✓ 0.0s Python

```
room_type
Entire home/apt    60811
Private room       33718
Shared room          433
Hotel room           182
Name: room_type, dtype: int64
```

```python
df['room_type'] = df['room_type'].replace({'Entire home/apt':'1'}, regex=False)
df['room_type'] = df['room_type'].replace({'Private room':'2'}, regex=False)
df['room_type'] = df['room_type'].replace({'Shared room':'3'}, regex=False)
df['room_type'] = df['room_type'].replace({'Hotel room':'4'}, regex=False)
```

✓ 0.0s Python

```python
cat2 = df.groupby(['host_is_superhost'])['host_is_superhost'].count().sort_values(ascending=False)
cat2
```

✓ 0.0s Python

```
host_is_superhost
f              76478
t              16918
Desconocido     1748
Name: host_is_superhost, dtype: int64
```

```python
df['host_is_superhost'] = df['host_is_superhost'].replace({'f':'1'}, regex=False)
df['host_is_superhost'] = df['host_is_superhost'].replace({'t':'2'}, regex=False)
df['host_is_superhost'] = df['host_is_superhost'].replace({'Desconocido':'3'}, regex=False)
```

✓ 0.0s Python

```python
tipo1 = df_filtrado[df_filtrado["room_type"] == "1"]
tipo2 = df_filtrado[df_filtrado["room_type"] == "2"]
tipo3 = df_filtrado[df_filtrado["room_type"] == "3"]
tipo4 = df_filtrado[df_filtrado["room_type"] == "4"]

tipo5 = df_filtrado2[df_filtrado2["room_type"] == "1"]
tipo6 = df_filtrado2[df_filtrado2["room_type"] == "2"]
tipo7 = df_filtrado2[df_filtrado2["room_type"] == "3"]
tipo8 = df_filtrado2[df_filtrado2["room_type"] == "4"]
```

✓ 0.0s Python

Ya que tenía los data frames filtrados por tipo de cuarto, seguí con el análisis entre las variables solicitadas: host_acceptance_rate, host_response_rate, price, review_scores_location_review_scores_cleanliness, “availability_365, number_of_reviews, reviews_per_month y review_scores_communication. Primero realicé heatmaps con todas las variables, por cada tipo de cuarto y después fui analizando las relaciones de variables dependientes e independientes solicitadas. El procedimiento presentado se realizó con cada uno de los tipos de cuarto, en este momento solo se ejemplifica con el primero, entire room/apt.

```python
columnas_deseadas = [
    'host_acceptance_rate',
    'room_type',
    'price',
    'reviews_per_month',
    'host_response_rate',
    'review_scores_location',
    'review_scores_cleanliness',
    'availability_365',
    'number_of_reviews',
    'review_scores_communication'
]

df_filtrado = df[columnas_deseadas]
```

✓ 0.0s Python

```python
Corr_Factors = tipo1.corr().dropna(how='all', axis=0).dropna(how='all', axis=1)
Corr_Factors
```

✓ 0.0s Python

```python
Corr_Factors= abs(Corr_Factors)
Corr_Factors
```

✓ 0.0s Python

```python
fig, ax = plt.subplots(figsize=(15, 10))
cax = ax.matshow(Corr_Factors, cmap="Reds")
fig.colorbar(cax)

# Añadir anotaciones manualmente
for i in range(Corr_Factors.shape[0]):
    for j in range(Corr_Factors.shape[1]):
        ax.text(j, i, f"{Corr_Factors.iloc[i, j]:.2f}",
                ha="center", va="center", fontsize=10)

plt.xticks(range(len(Corr_Factors.columns)), Corr_Factors.columns, rotation=90, fontsize=12)
plt.yticks(range(len(Corr_Factors.index)), Corr_Factors.index, fontsize=12)
plt.savefig('Tipo1.png', dpi=300, bbox_inches='tight')
plt.show()
```

✓ 0.6s Python

```python
Vars_Indep= tipo1[['host_response_rate']]
Var_Dep= tipo1['host_acceptance_rate']
model= LinearRegression()
model.fit(X=Vars_Indep, y=Var_Dep)
coef_Deter=model.score(X=Vars_Indep, y=Var_Dep)
coef_Deter
```

✓ 0.0s Python

```
0.09852261737500156
```

```python
Vars_Indep= tipo1[['review_scores_cleanliness']]
Var_Dep= tipo1['review_scores_location']
model= LinearRegression()
model.fit(X=Vars_Indep, y=Var_Dep)
coef_Deter=model.score(X=Vars_Indep, y=Var_Dep)
coef_Deter
```

✓ 0.0s Python

```
0.1405791881740739
```

```python
Vars_Indep= tipo1[['price']]
Var_Dep= tipo1['host_acceptance_rate']
model= LinearRegression()
model.fit(X=Vars_Indep, y=Var_Dep)
coef_Deter=model.score(X=Vars_Indep, y=Var_Dep)
coef_Deter
```

✓ 0.0s Python

```
0.006556853333384738
```

```python
Vars_Indep= tipo1[['number_of_reviews']]
Var_Dep= tipo1['availability_365']
model= LinearRegression()
model.fit(X=Vars_Indep, y=Var_Dep)
coef_Deter=model.score(X=Vars_Indep, y=Var_Dep)
coef_Deter
```

✓ 0.0s Python

```
0.0004920720238429377
```

```python
Vars_Indep= tipo1[['number_of_reviews']]
Var_Dep= tipo1['host_acceptance_rate']
model= LinearRegression()
model.fit(X=Vars_Indep, y=Var_Dep)
coef_Deter=model.score(X=Vars_Indep, y=Var_Dep)
coef_Deter
```

✓ 0.0s Python

```
0.011903799396786652
```

```python
Vars_Indep= tipo1[['review_scores_communication']]
Var_Dep= tipo1['reviews_per_month']
model= LinearRegression()
model.fit(X=Vars_Indep, y=Var_Dep)
coef_Deter=model.score(X=Vars_Indep, y=Var_Dep)
coef_Deter
```

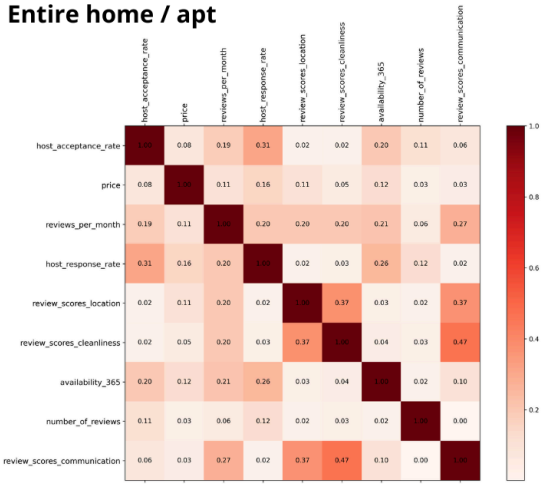
✓ 0.0s Python

```
0.07540069847024278
```

Resumen:

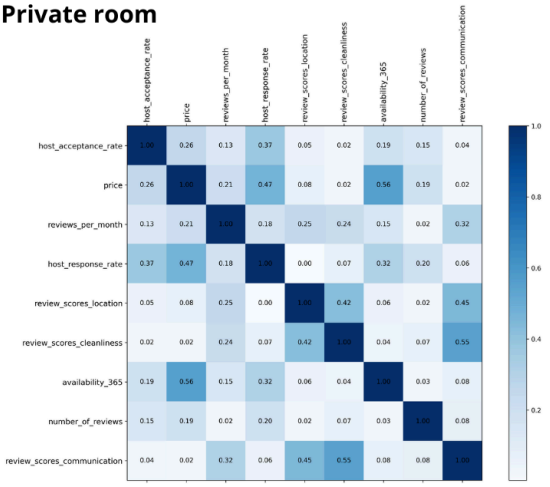
Tipo de cuarto	Coeficientes	host_acceptance_rate vs host_response_rate	review_scores_location vs r eview_scores_cleanliness	host_acceptance_rate vs price	availability_365 vs number_of_reviews	host_acceptance_rate vs number_of_reviews	reviews_per_month vs review_scores_commu
Entire room/apt	Determinación	0.10	0.14	0.01	0.00	0.01	0.08
	Correlación	0.31	0.37	0.08	0.02	0.11	0.27
Private room	Determinación	0.14	0.18	0.07	0.00	0.02	0.10
	Correlación	0.37	0.42	0.26	0.03	0.15	0.32
Shared room	Determinación	0.14	0.10	0.18	0.01	0.07	0.18
	Correlación	0.37	0.31	0.42	0.09	0.27	0.43
Hotel room	Determinación	0.48	0.08	0.01	0.00	0.07	0.07
	Correlación	0.70	0.28	0.11	0.01	0.27	0.26

Entire home / apt



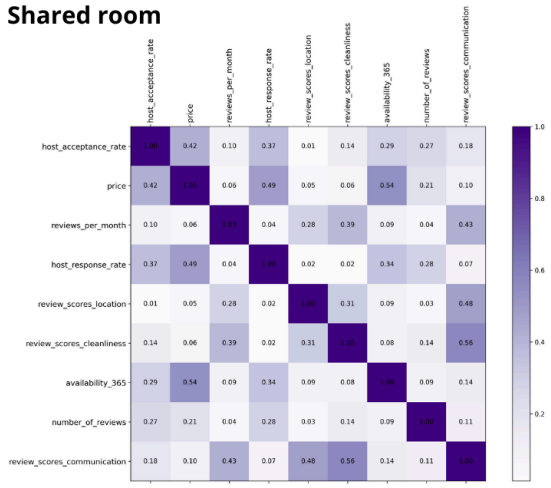
Variable 1	Variable 2	Coeficiente de Correlación
review_scores_communication	review_scores_cleanliness	0.47
review_scores_cleanliness	review_scores_location	0.37
review_scores_communication	review_scores_location	0.37
host_response_rate	host_acceptance_rate	0.36
review_scores_communication	reviews_per_month	0.27
host_response_rate	availability_365	0.26
reviews_per_month	availability_365	0.21
reviews_per_month	review_scores_location	0.20
reviews_per_month	review_scores_cleanliness	0.20
host_acceptance_rate	availability_365	0.20

Private room



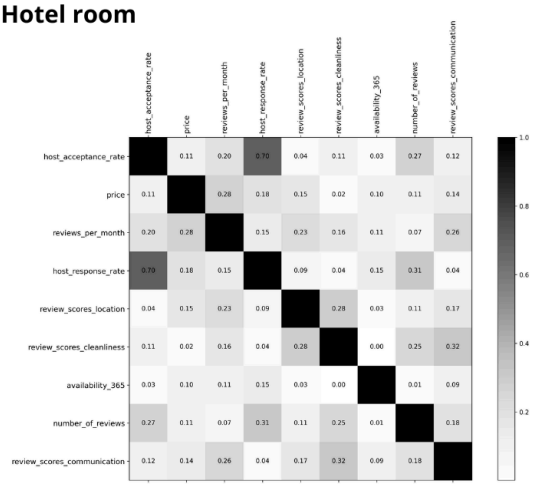
Variable 1	Variable 2	Coeficiente de Correlación
review_scores_communication	review_scores_cleanliness	0.55
review_scores_communication	review_scores_location	0.45
review_scores_cleanliness	review_scores_location	0.42
price	host_response_rate	0.47
review_scores_communication	reviews_per_month	0.32
host_response_rate	availability_365	0.32
reviews_per_month	review_scores_communication	0.32
price	availability_365	0.56
host_response_rate	host_acceptance_rate	0.37
review_scores_location	reviews_per_month	0.25

Shared room



Variable 1	Variable 2	Coeficiente de Correlación
review_scores_communication	review_scores_cleanliness	0.56
price	availability_365	0.54
review_scores_communication	review_scores_location	0.48
host_response_rate	price	0.49
review_scores_communication	reviews_per_month	0.43
price	host_acceptance_rate	0.42
review_scores_cleanliness	reviews_per_month	0.39
host_response_rate	host_acceptance_rate	0.37
availability_365	host_response_rate	0.34
review_scores_cleanliness	review_scores_location	0.31

Hotel room



Variable 1	Variable 2	Coeficiente de Correlación
host_response_rate	host_acceptance_rate	0.70
review_scores_communication	review_scores_cleanliness	0.32
number_of_reviews	host_response_rate	0.31
reviews_per_month	price	0.28
review_scores_cleanliness	review_scores_location	0.28
number_of_reviews	host_acceptance_rate	0.27
reviews_per_month	review_scores_communication	0.26
review_scores_cleanliness	number_of_reviews	0.25
reviews_per_month	review_scores_location	0.23
number_of_reviews	price	0.11

En los anteriores mapas de calor, se hace la comparación de los coeficientes de correlación entre variables lineales en cada uno de los tipos de cuarto, señalando los 10 mayores coeficientes de cada uno de los data frames filtrados.

Posteriormente, continué con el análisis lineal múltiple del otro conjunto de datos solicitados: host_id, host_acceptance_rate, host_is_superhost, host_total_listings_count, room_type, accommodates, bedrooms, Price, review_scores_value y reviews_per_month. Primero, realicé los heatmaps por cada data frame filtrado por tipo de cuarto. Después identifiqué los coeficientes más altos en cada uno de ellos y definí las variables que utilizaría como dependientes en cada uno de los modelos de regresión lineal múltiple.

```python
corr_factors = tipo5.corr().dropna(how='all', axis=0).dropna(how='all', axis=1)
corr_factors
```

✓ 0.0s Python

```python
fig, ax = plt.subplots(figsize=(15, 10))
cax = ax.matshow(corr_factors, cmap="Reds")
fig.colorbar(cax)

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

plt.xticks(range(len(corr_factors.columns)), corr_factors.columns, rotation=90, fontsize=12)
plt.yticks(range(len(corr_factors.index)), corr_factors.index, fontsize=12)
plt.savefig('Tipo5.png', dpi=300, bbox_inches='tight')
plt.show()
```

✓ 0.5s Python

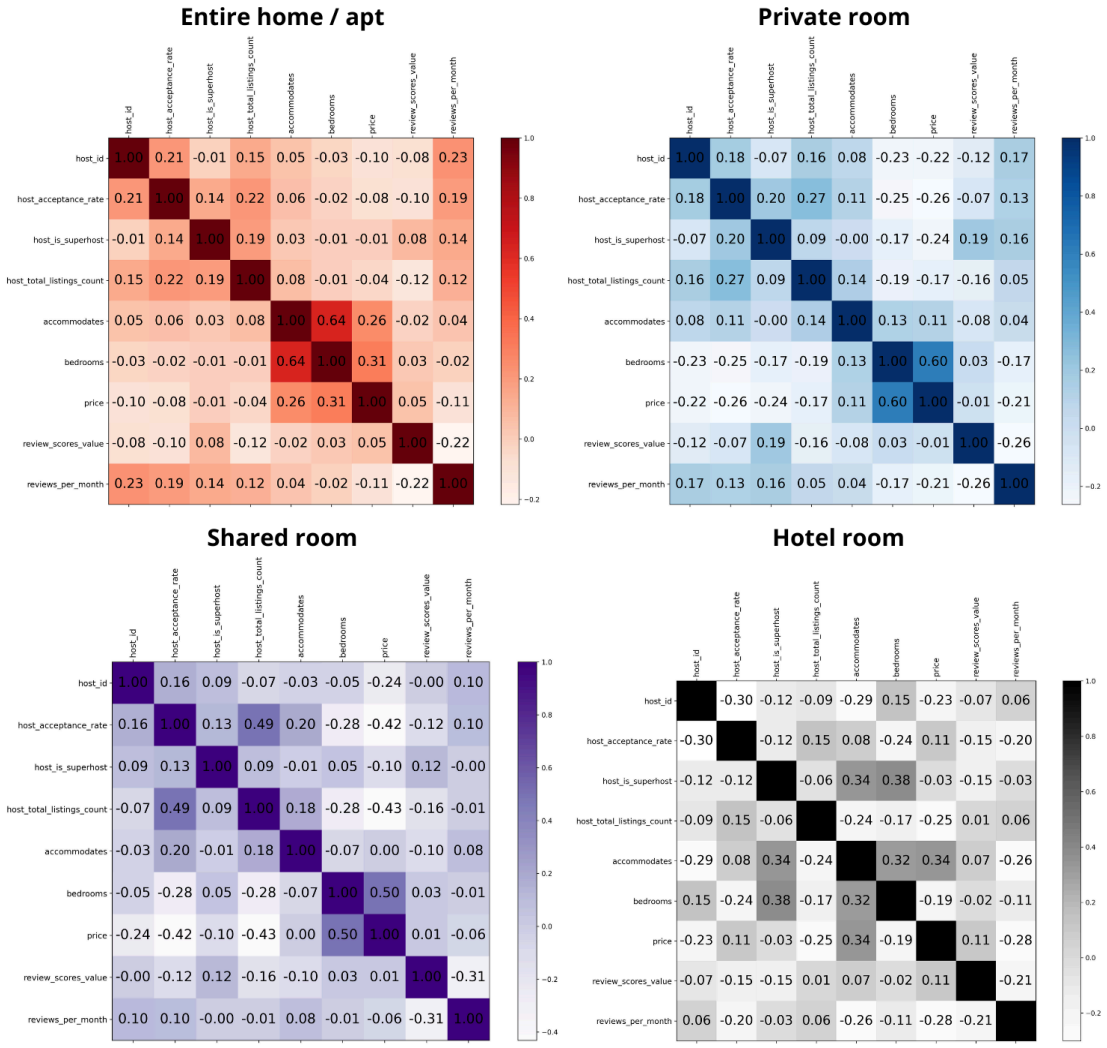
```python
Vars_Indep= tipo5[['host_id', 'host_acceptance_rate', 'host_is_superhost', 'host_total_listings_count', 'accommodates', 'price', 'review_scores_value', 'r
Var_Dep= tipo5['bedrooms']
```

✓ 0.0s Python

```python
model = LinearRegression()
model.fit(X=Vars_Indep, y=Var_Dep)
coef_DeterU3=model.score(X=Vars_Indep, y=Var_Dep)
coef_CorrelU3=np.sqrt(coef_DeterU3)
coef_CorrelU3
```

✓ 0.1s Python

```
0.6627738212391481
```



Tipo de cuarto	Variable seleccionada (dependiente)	Coeficiente correlación linear	Coeficiente correlación linear múltiple
Entire room/apt	bedrooms	**0.64**	**0.66**
Private room	price	**0.60**	**0.64**
Shared room	price	**0.50**	**0.66**
Hotel room	bedrooms	**0.38**	**0.60**

En conclusión, el análisis de los modelos de regresión lineal múltiple aplicados a los diferentes tipos de alojamiento (entire room/apt, private room, shared room y hotel room) revela una mejora consistente en los coeficientes de correlación en comparación con la regresión lineal simple. Este aumento en los coeficientes de correlación en los cuatro escenarios demuestra la efectividad de la regresión lineal múltiple para proporcionar una predicción y un análisis más preciso de los conjuntos de datos al considerar la influencia de múltiples variables independientes en la variable dependiente seleccionada para cada tipo de cuarto.