Machine Learning Activity 1 (2448016)

Data Preprocessing and EDA

Importing necessary packages

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

1. Loading the dataset

```
df_data1=pd.read_csv('/content/data1.csv')
df_data2=pd.read_csv('/content/data2.csv')
df_data=pd.concat([df_data1, df_data2], axis=0)
```

```
df_data.head()
```

```
df_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 4137 entries, 0 to 1372
Data columns (total 24 columns):
```

```
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   1:Date                          4137 non-null   object
 1   2:Time                          4137 non-null   object
 2   3:Temperature_Comedor_Sensor    4137 non-null   float64
 3   4:Temperature_Habitacion_Sensor 4137 non-null   float64
 4   5:Weather_Temperature           4137 non-null   float64
 5   6:CO2_Comedor_Sensor            4137 non-null   float64
 6   7:CO2_Habitacion_Sensor         4137 non-null   float64
 7   8:Humedad_Comedor_Sensor        4137 non-null   float64
 8   9:Humedad_Habitacion_Sensor     4137 non-null   float64
 9   10:Lighting_Comedor_Sensor      4137 non-null   float64
 10  11:Lighting_Habitacion_Sensor   4137 non-null   float64
 11  12:Precipitacion                4137 non-null   int64
 12  13:Meteo_Exterior_Crepusculo    4137 non-null   float64
 13  14:Meteo_Exterior_Viento        4137 non-null   float64
 14  15:Meteo_Exterior_Sol_Oest      4137 non-null   float64
 15  16:Meteo_Exterior_Sol_Est       4137 non-null   float64
 16  17:Meteo_Exterior_Sol_Sud       4137 non-null   float64
 17  18:Meteo_Exterior_Piranometro   4137 non-null   float64
 18  19:Exterior_Entalpic_1          4137 non-null   int64
 19  20:Exterior_Entalpic_2          4137 non-null   int64
 20  21:Exterior_Entalpic_turbo      4137 non-null   int64
 21  22:Temperature_Exterior_Sensor  4137 non-null   float64
 22  23:Humedad_Exterior_Sensor      4137 non-null   float64
 23  24:Day_Of_Week                  4137 non-null   float64
dtypes: float64(18), int64(4), object(2)
memory usage: 808.0+ KB
```

```
df_data.describe()
```

	3:Temperature_Comedor_Sensor	4:Temperature_Habitacion_Sensor	5:Weather_Tempera
count	4137.000000	4137.00000	4137.00
mean	20.493530	20.14605	15.09
std	3.312096	3.31100	4.37
min	11.352000	11.07600	0.00
25%	18.424600	18.06930	12.00
50%	20.499300	20.14070	15.00
75%	22.865300	22.46130	18.00
max	28.924000	28.54800	29.00

8 rows × 22 columns

2. Rename Columns for Readability

```
# Define new column names
new_column_names = {
    "1:Date": "Date",
    "2:Time": "Time",
    "3:Temperature_Comedor_Sensor": "Temp_Comedor",
    "4:Temperature_Habitacion_Sensor": "Temp_Habitacion",
    "5:Weather_Temperature": "Weather_Temp",
    "6:CO2_Comedor_Sensor": "CO2_Comedor",
    "7:CO2_Habitacion_Sensor": "CO2_Habitacion",
    "8:Humedad_Comedor_Sensor": "Humidity_Comedor",
    "9:Humedad_Habitacion_Sensor": "Humidity_Habitacion",
    "10:Lighting_Comedor_Sensor": "Lighting_Comedor",
    "11:Lighting_Habitacion_Sensor": "Lighting_Habitacion",
    "12:Precipitacion": "Precipitation",
    "13:Meteo_Exterior_Crepusculo": "Meteo_Crepusculo",
    "14:Meteo_Exterior_Viento": "Meteo_Viento",
    "15:Meteo_Exterior_Sol_Oest": "Meteo_Sol_Oest",
    "16:Meteo_Exterior_Sol_Est": "Meteo_Sol_Est",
    "17:Meteo_Exterior_Sol_Sud": "Meteo_Sol_Sud",
    "18:Meteo_Exterior_Piranometro": "Meteo_Piranometro",
    "19:Exterior_Entalpic_1": "Entalpic_1",
    "20:Exterior_Entalpic_2": "Entalpic_2",
    "21:Exterior_Entalpic_turbo": "Entalpic_Turbo",
    "22:Temperature_Exterior_Sensor": "Temp_Exterior",
    "23:Humedad_Exterior_Sensor": "Humidity_Exterior",
    "24:Day_Of_Week": "Day_Of_Week"
}

# Rename columns
df_data.rename(columns=new_column_names, inplace=True)

# Display updated column names
print(df_data.columns)
```

```
Index(['Date', 'Time', 'Temp_Comedor', 'Temp_Habitacion', 'Weather_Temp',
       'CO2_Comedor', 'CO2_Habitacion', 'Humidity_Comedor',
       'Humidity_Habitacion', 'Lighting_Comedor', 'Lighting_Habitacion',
       'Precipitation', 'Meteo_Crepusculo', 'Meteo_Viento', 'Meteo_Sol_Oest',
       'Meteo_Sol_Est', 'Meteo_Sol_Sud', 'Meteo_Piranometro', 'Entalpic_1',
       'Entalpic_2', 'Entalpic_Turbo', 'Temp_Exterior', 'Humidity_Exterior',
       'Day_Of_Week'],
      dtype='object')
```

3. Convert Date and Time into a Single Datetime Column

```
# Convert Date and Time to datetime format
df_data["Datetime"] = pd.to_datetime(df_data["Date"] + " " + df_data["Time"])
```

```
# Drop the original Date and Time columns
df_data.drop(columns=["Date", "Time"], inplace=True)

# Check the new structure
df_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 4137 entries, 0 to 1372
Data columns (total 23 columns):
 #   Column               Non-Null Count  Dtype         
---  ------               --------------  -----         
 0   Temp_Comedor         4137 non-null   float64       
 1   Temp_Habitacion      4137 non-null   float64       
 2   Weather_Temp         4137 non-null   float64       
 3   CO2_Comedor          4137 non-null   float64       
 4   CO2_Habitacion       4137 non-null   float64       
 5   Humidity_Comedor     4137 non-null   float64       
 6   Humidity_Habitacion  4137 non-null   float64       
 7   Lighting_Comedor     4137 non-null   float64       
 8   Lighting_Habitacion  4137 non-null   float64       
 9   Precipitation        4137 non-null   int64         
 10  Meteo_Crepusculo     4137 non-null   float64       
 11  Meteo_Viento         4137 non-null   float64       
 12  Meteo_Sol_Oest       4137 non-null   float64       
 13  Meteo_Sol_Est        4137 non-null   float64       
 14  Meteo_Sol_Sud        4137 non-null   float64       
 15  Meteo_Piranometro    4137 non-null   float64       
 16  Entalpic_1           4137 non-null   int64         
 17  Entalpic_2           4137 non-null   int64         
 18  Entalpic_Turbo       4137 non-null   int64         
 19  Temp_Exterior        4137 non-null   float64       
 20  Humidity_Exterior    4137 non-null   float64       
 21  Day_Of_Week          4137 non-null   float64       
 22  Datetime             4137 non-null   datetime64[ns]
dtypes: datetime64[ns](1), float64(18), int64(4)
memory usage: 775.7 KB
```

4. Check for Missing Values

```
# Count missing values per column
missing_values = df_data.isnull().sum()

# Show columns with missing values
print(missing_values[missing_values > 0])
```

```
Series([], dtype: int64)
```

5. Check for Duplicate Rows

```
# Count duplicate rows
duplicate_count = df_data.duplicated().sum()
print(f"Number of duplicate rows: {duplicate_count}")
```

```
Number of duplicate rows: 0
```

6. Detect Outliers Using the IQR Method

```
import numpy as np

# Select only numerical columns
df_numeric = df_data.select_dtypes(include=[np.number])

# Compute Q1, Q3, and IQR
Q1 = df_numeric.quantile(0.25)
Q3 = df_numeric.quantile(0.75)
IQR = Q3 - Q1

# Define outlier boundaries
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR

# Count outliers per column
outlier_counts = ((df_numeric < lower_bound) | (df_numeric > upper_bound)).sum()

# Show columns with outliers
print(outlier_counts[outlier_counts > 0])
```

```
Temp_Comedor            11
Temp_Habitacion         11
Weather_Temp             5
CO2_Comedor             96
CO2_Habitacion         116
Lighting_Comedor       131
Lighting_Habitacion    398
Precipitation          162
Meteo_Viento             4
Meteo_Sol_Oest         579
Meteo_Sol_Est          628
Meteo_Sol_Sud          288
Temp_Exterior            2
dtype: int64
```

Handling outliers

```
# Columns with outliers
outlier_columns = [
    "Temp_Comedor", "Temp_Habitacion", "Weather_Temp", "CO2_Comedor",
    "CO2_Habitacion", "Lighting_Comedor", "Lighting_Habitacion",
    "Precipitation", "Meteo_Viento", "Meteo_Sol_Oest",
    "Meteo_Sol_Est", "Meteo_Sol_Sud", "Temp_Exterior"
]

# Compute IQR
Q1 = df_data[outlier_columns].quantile(0.25)
Q3 = df_data[outlier_columns].quantile(0.75)
IQR = Q3 - Q1

# Define bounds
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR

# Capping outliers at the threshold
for col in outlier_columns:
    df_data[col] = np.where(df_data[col] < lower_bound[col], lower_bound[col], df_data[col])
    df_data[col] = np.where(df_data[col] > upper_bound[col], upper_bound[col], df_data[col])

# Verify outliers after handling
outlier_counts_after = ((df_data[outlier_columns] < lower_bound) | (df_data[outlier_columns]
print(outlier_counts_after)
```

```
Temp_Comedor          0
Temp_Habitacion       0
Weather_Temp          0
CO2_Comedor           0
CO2_Habitacion        0
Lighting_Comedor      0
Lighting_Habitacion   0
Precipitation         0
Meteo_Viento          0
Meteo_Sol_Oest        0
Meteo_Sol_Est         0
Meteo_Sol_Sud         0
Temp_Exterior         0
dtype: int64
```

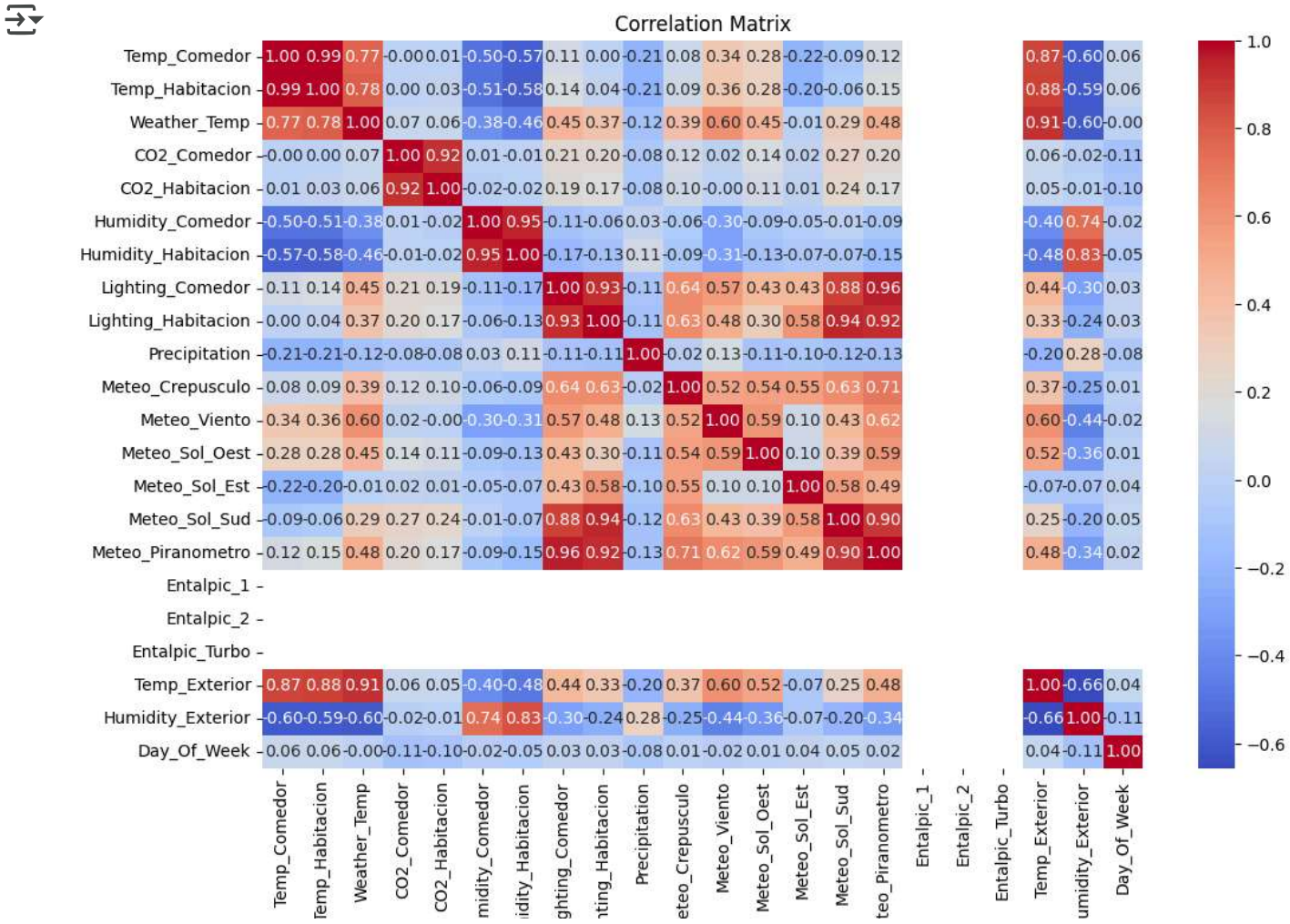
```
df_data.shape
```

```
(4137, 23)
```

7. Correlation Analysis

```
# Compute correlation matrix
corr_matrix = df_numeric.corr()
```

```
# Plot heatmap
plt.figure(figsize=(12, 8))
sns.heatmap(corr_matrix, annot=True, cmap="coolwarm", fmt=".2f")
plt.title("Correlation Matrix")
plt.show()
```



Key Insights from the Correlation Matrix:

Strong Positive Correlations:

Temp_Comedor & Temp_Habitacion (0.99) → Room temperatures move together.

Weather_Temp & Temp_Exterior (0.91) → Outdoor temperature influences exterior readings.

CO2_Comedor & CO2_Habitacion (0.92) → CO2 levels are linked.

Lighting_Comedor & Lighting_Habitacion (0.88) → Similar lighting patterns.

Strong Negative Correlations:

Humidity_Exterior & Temp_Exterior (-0.66) → Warmer air reduces humidity.

Humidity_Comedor & Temp_Comedor (-0.50) → Heat lowers indoor humidity.

Moderate Correlations:

Meteo_Piranometro & Meteo_Sol_Sud (0.96) → Solar exposure affects radiation.

Meteo_Crepusculo & Meteo_Piranometro (0.71) → Twilight influences solar radiation.

Weak/No Correlation:

Day_Of_Week has minimal impact.

CO2 levels are mostly independent of temperature.

Conclusion:

Indoor conditions are closely linked, outdoor weather influences them, and temperature-humidity have an inverse relationship.

8. Feature Scaling

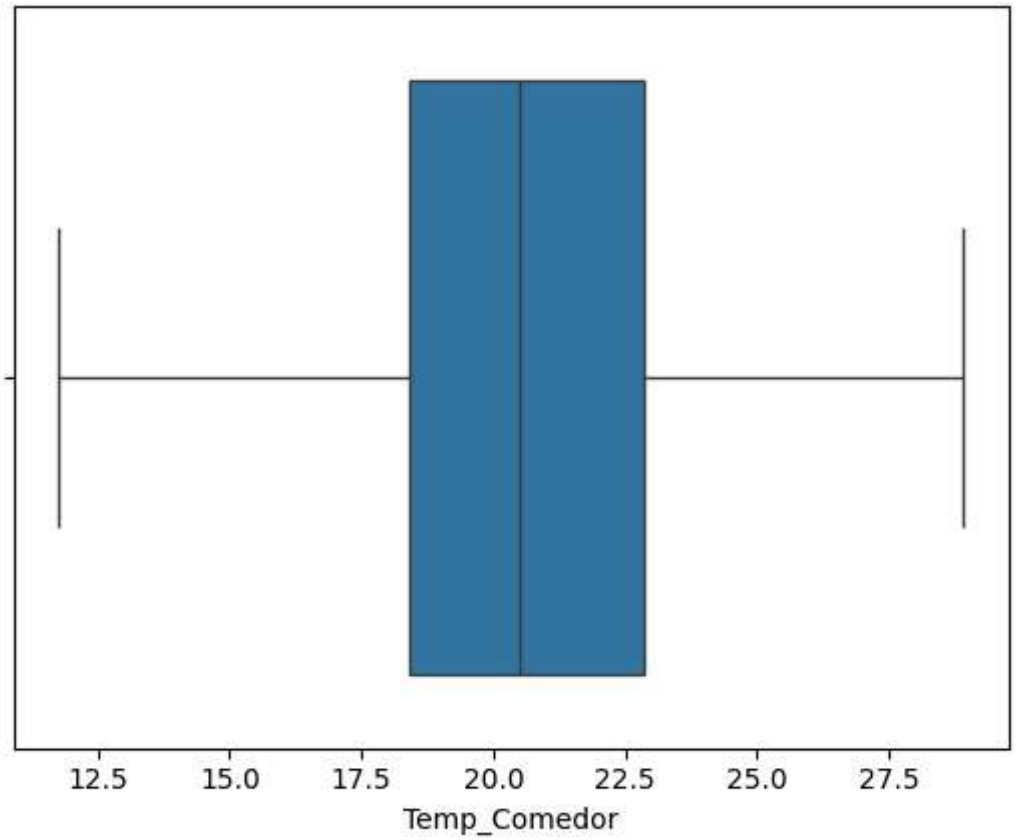
```
from sklearn.preprocessing import StandardScaler

# Initialize scaler
scaler = StandardScaler()

# Scale numerical features
df_scaled = pd.DataFrame(scaler.fit_transform(df_numeric), columns=df_numeric.columns)

# Show scaled data
print(df_scaled.head())
```

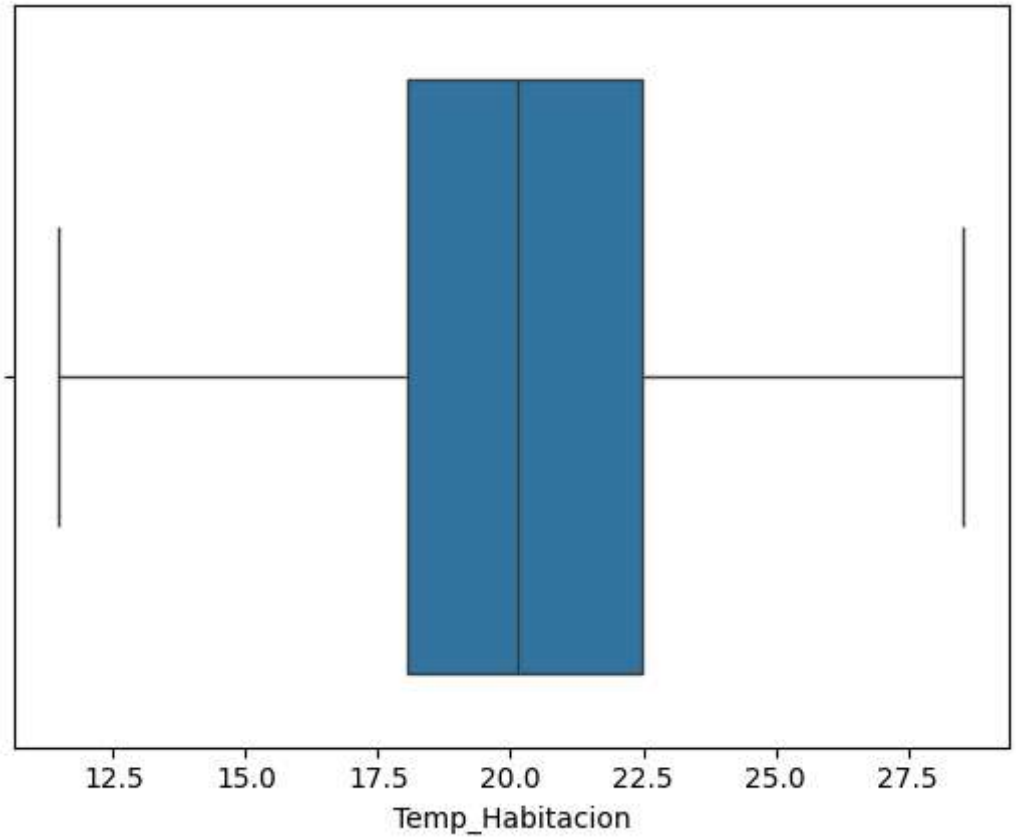
```
   Temp_Comedor  Temp_Habitacion  Weather_Temp  CO2_Comedor  CO2_Habitacion  \
0     -0.696329        -0.700341     -3.449877     0.437610        0.509020
1     -0.613048        -0.611777     -1.895536     0.586421        0.444629
2     -0.521253        -0.516326      0.435975     0.562520        0.385491
3     -0.429034        -0.421298      0.664555     0.527811        0.307411
4     -0.338627        -0.333671      1.121714     0.488312        0.267503

   Humidity_Comedor  Humidity_Habitacion  Lighting_Comedor  \
0         -0.343387            -0.256866          2.051876
1         -0.341419            -0.277320          2.054847
2         -0.362862            -0.279562          2.042492
3         -0.362308            -0.294990          2.045502
4         -0.362349            -0.298619          2.044116

   Lighting_Habitacion  Precipitation  ...  Meteo_Sol_Oest  Meteo_Sol_Est  \
0             1.671099      -0.201878  ...       -0.199930      -0.041275
1             1.673094      -0.201878  ...       -0.147298      -0.119228
```

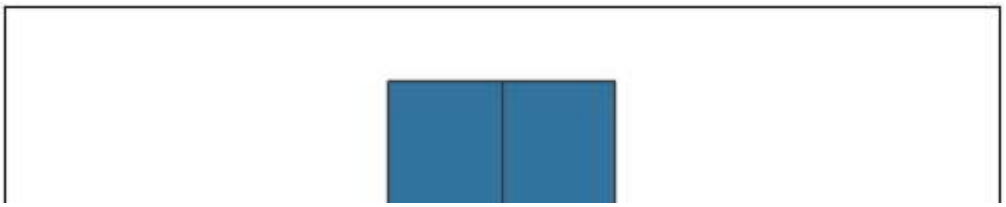
```
2             1.672977      -0.201878  ...       -0.031168      -0.167174
3             1.666967      -0.201878  ...        0.148678      -0.168096
4             1.659690      -0.201878  ...        0.458431      -0.186965

   Meteo_Sol_Sud  Meteo_Piranometro  Entalpic_1  Entalpic_2  Entalpic_Turbo  \
0       2.562800           1.685780         0.0         0.0             0.0
1       2.562800           1.695987         0.0         0.0             0.0
2       2.561505           1.709373         0.0         0.0             0.0
3       2.560206           1.708688         0.0         0.0             0.0
4       2.560023           1.698145         0.0         0.0             0.0

   Temp_Exterior  Humidity_Exterior  Day_Of_Week
0       0.022328          -0.360980    -0.986859
1       0.092274          -0.402951    -0.986859
2       0.194638          -0.430783    -0.986859
3       0.296721          -0.460984    -0.986859
4       0.401582          -0.575697    -0.986859

[5 rows x 22 columns]
```

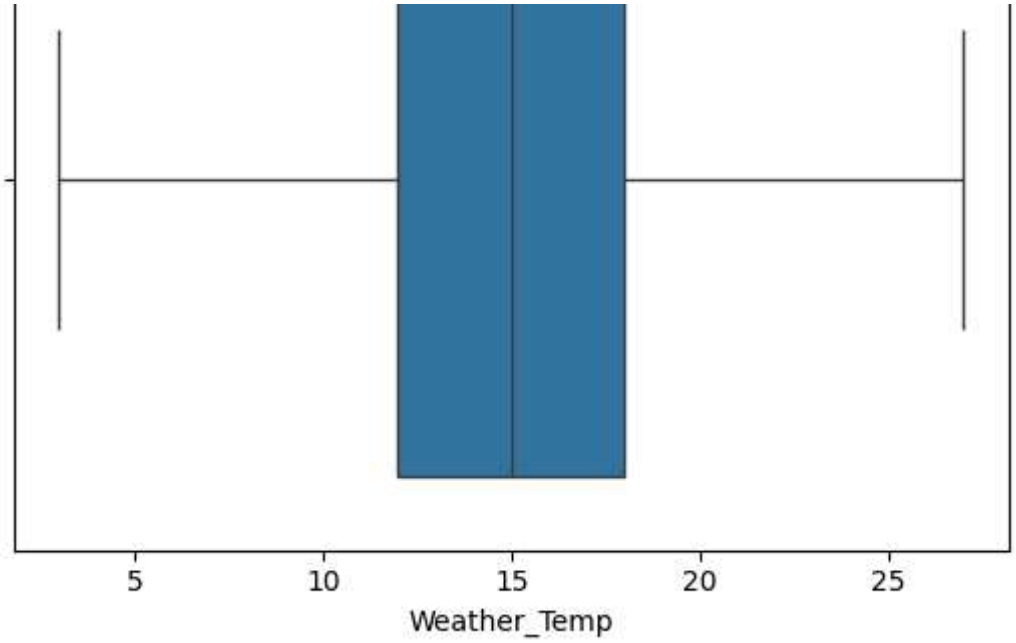
Visualization

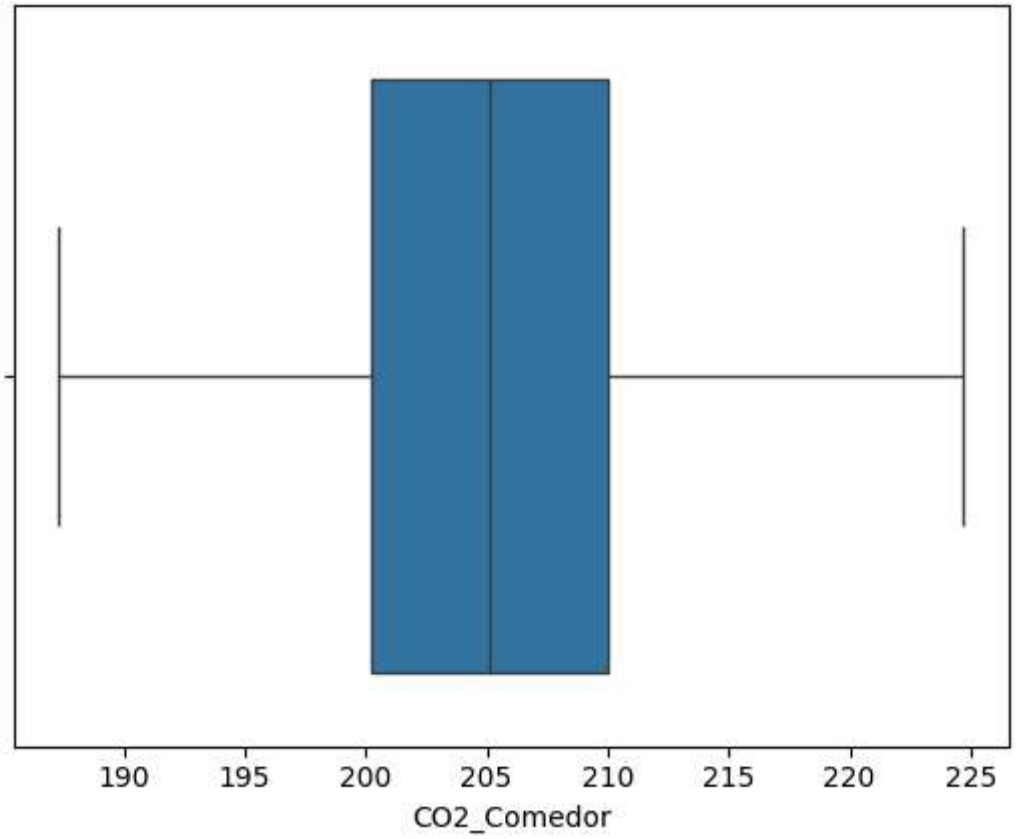
```
for col in outlier_columns:
    sns.boxplot(x=df_data[col])
    plt.show()
```

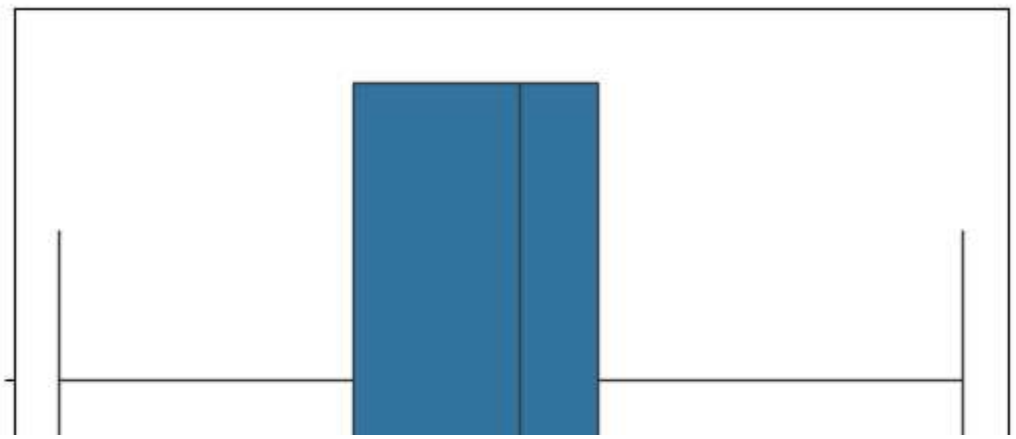

12.5
15.0
17.5
20.0
22.5
25.0
27.5
Temp_Comedor

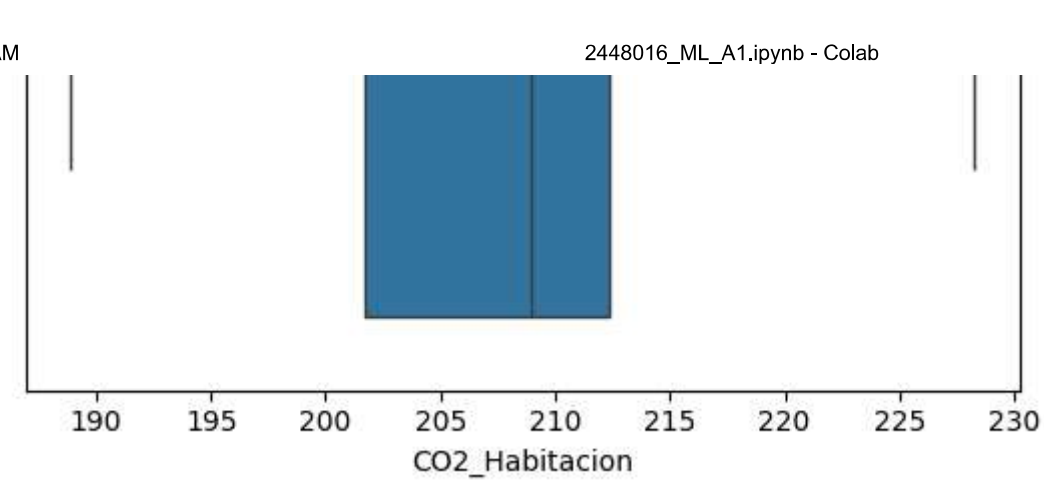

12.5
15.0
17.5
20.0
22.5
25.0
27.5
Temp_Habitacion

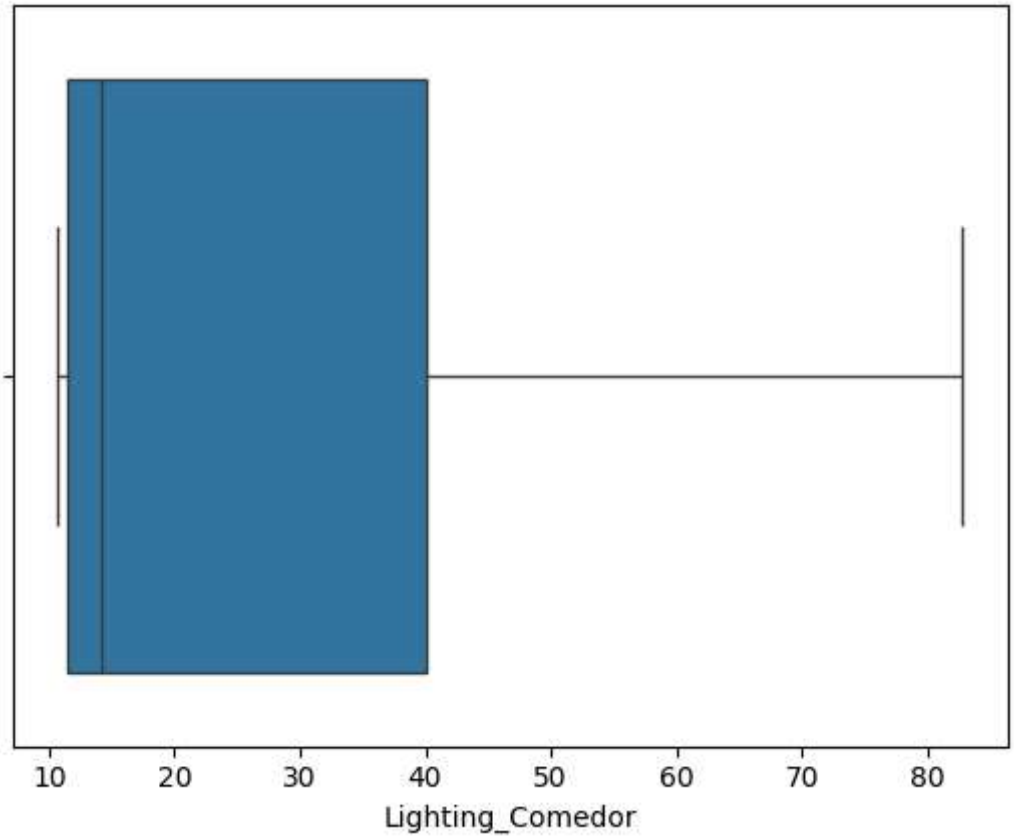


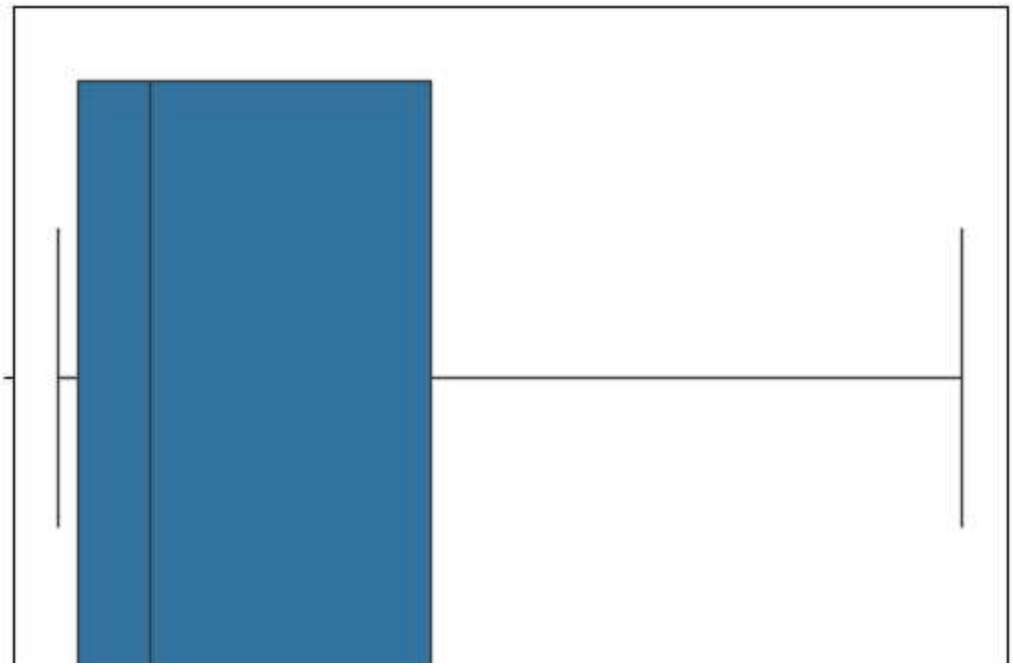

5
10
15
20
25
Weather_Temp

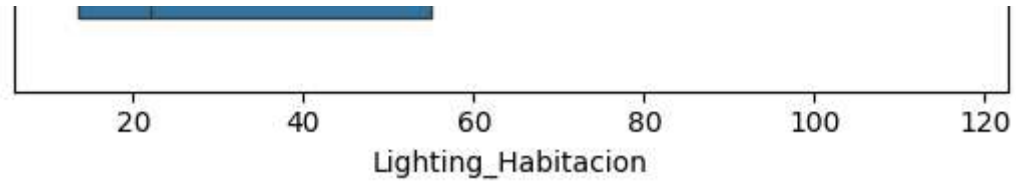

190
195
200
205
210
215
220
225
CO2_Comedor



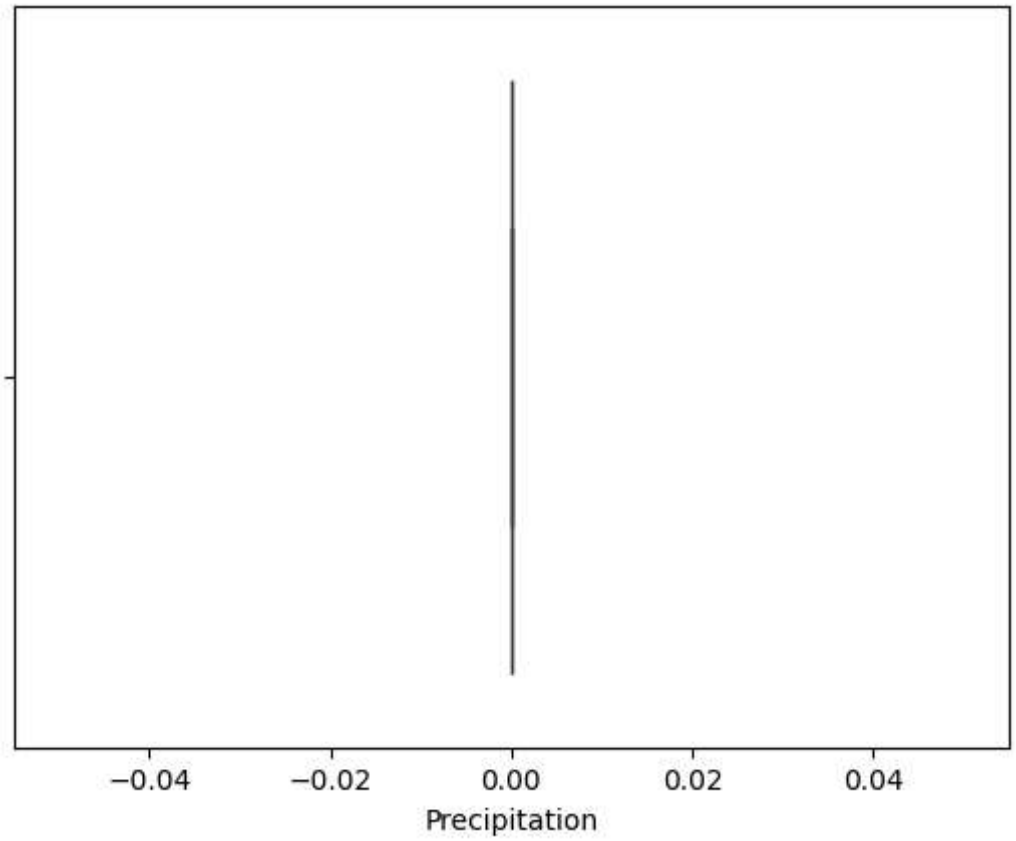

190
195
200
205
210
215
220
225
230
CO2_Habitacion


10
20
30
40
50
60
70
80
Lighting_Comedor

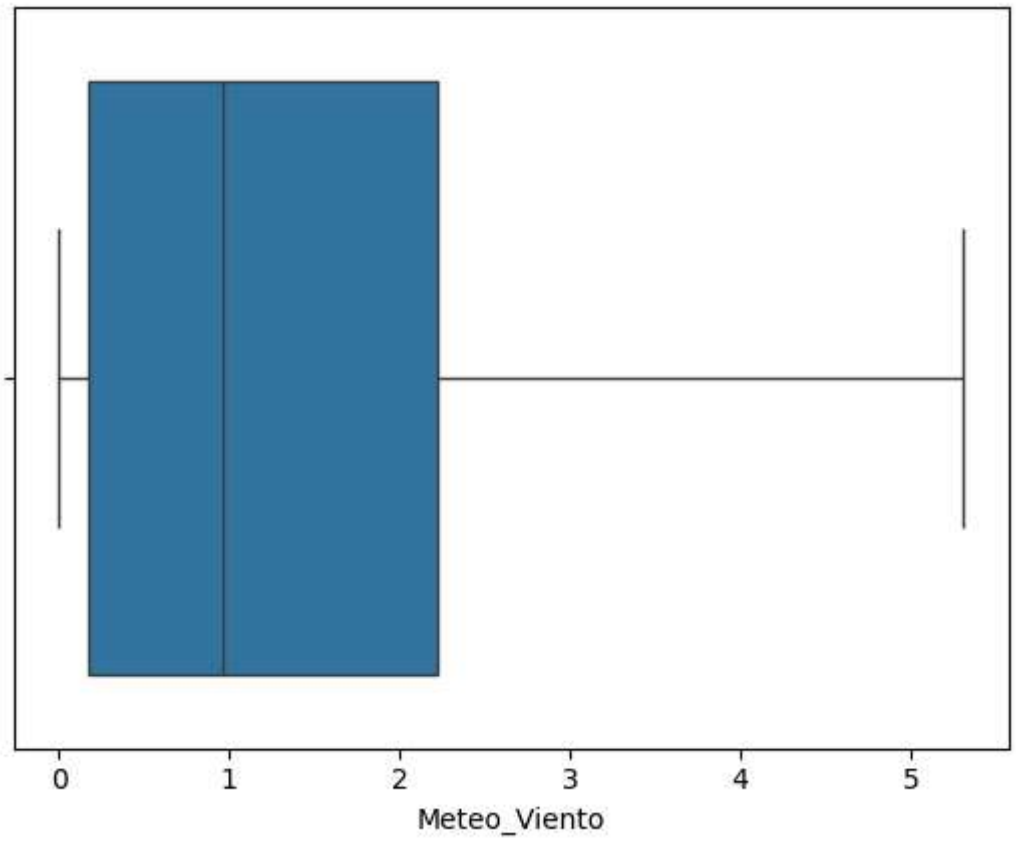




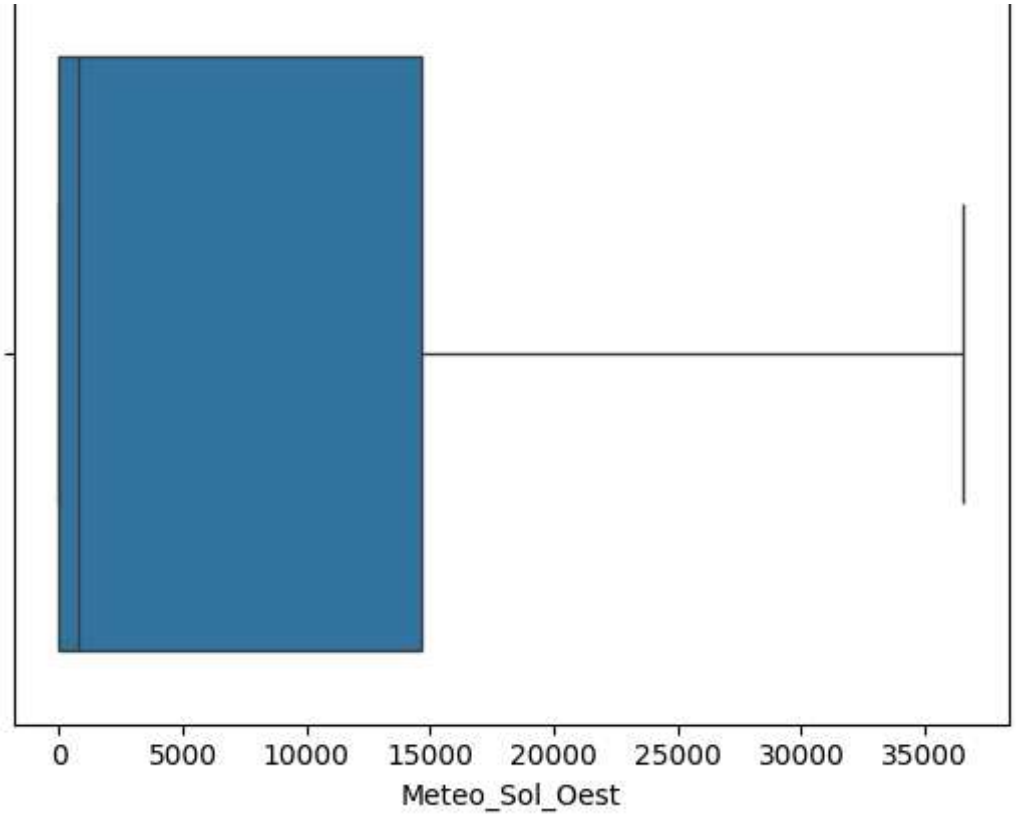
Lighting_Habitacion

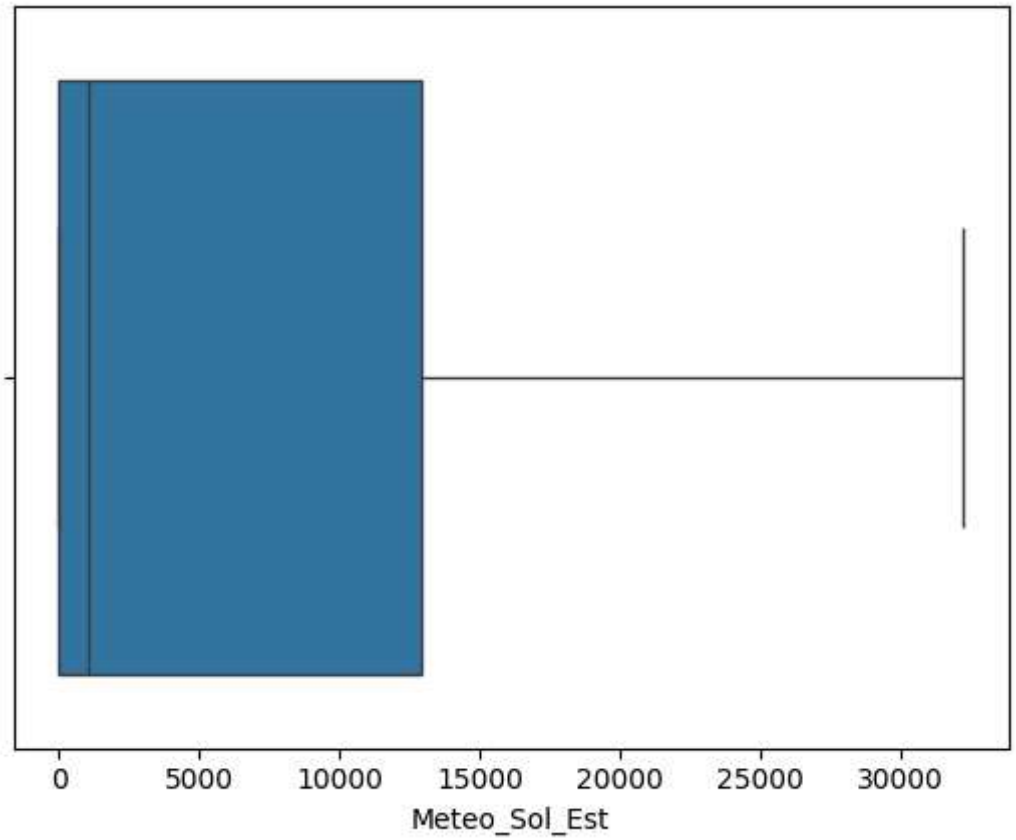


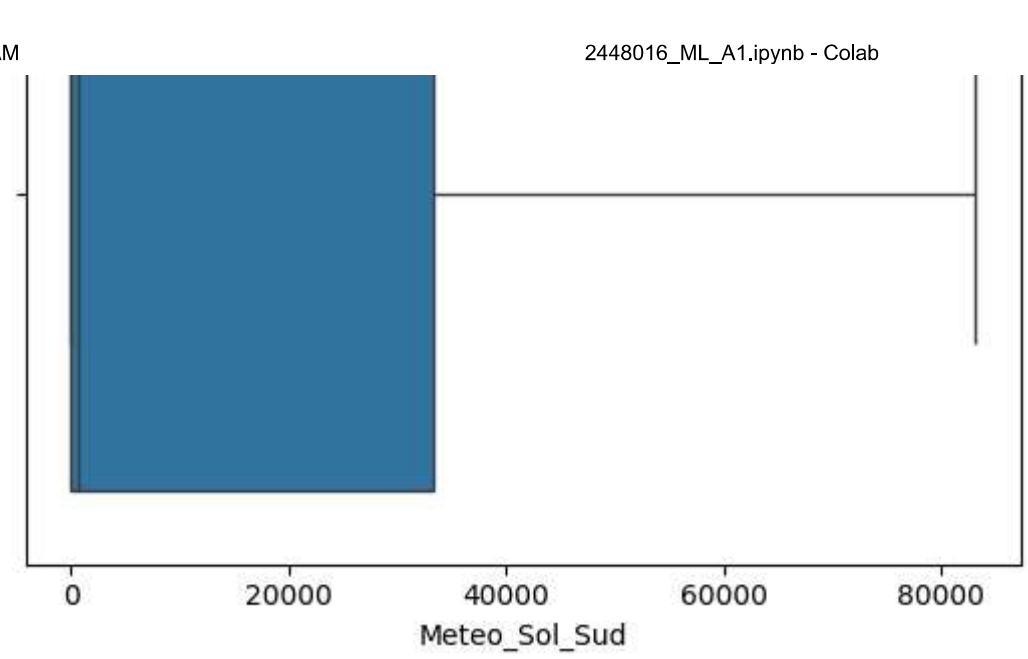
Precipitation

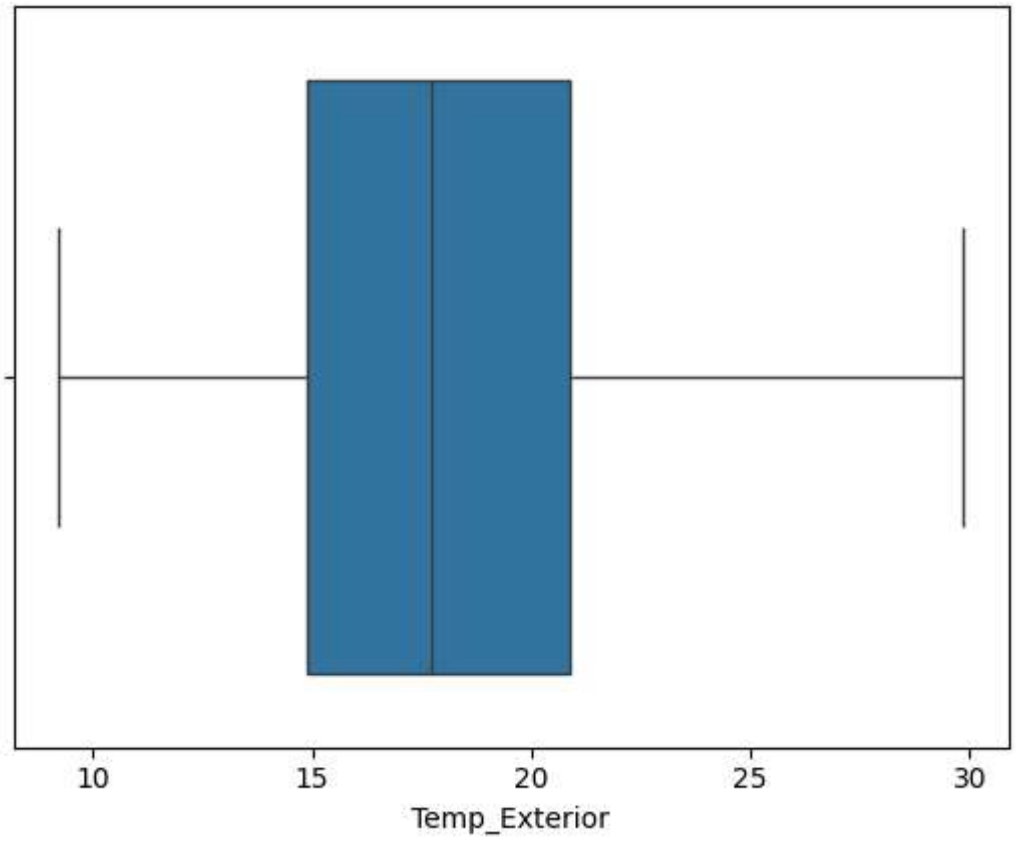


Meteo_Viento


0
5000
10000
15000
20000
25000
30000
35000
Meteo_Sol_Oest

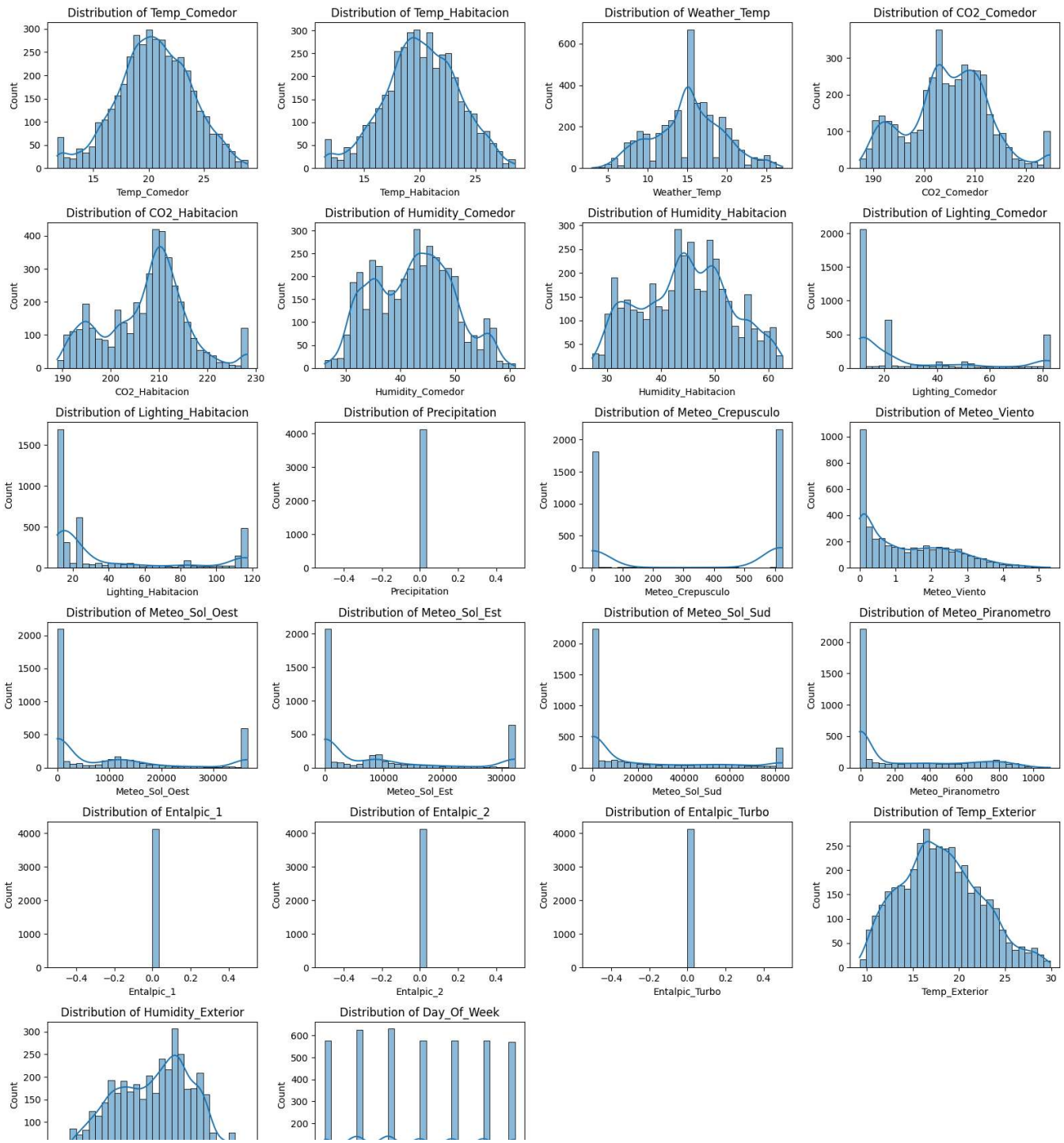

0
5000
10000
15000
20000
25000
30000
Meteo_Sol_Est


0
20000
40000
60000
80000
Meteo_Sol_Sud


10
15
20
25
30
Temp_Exterior

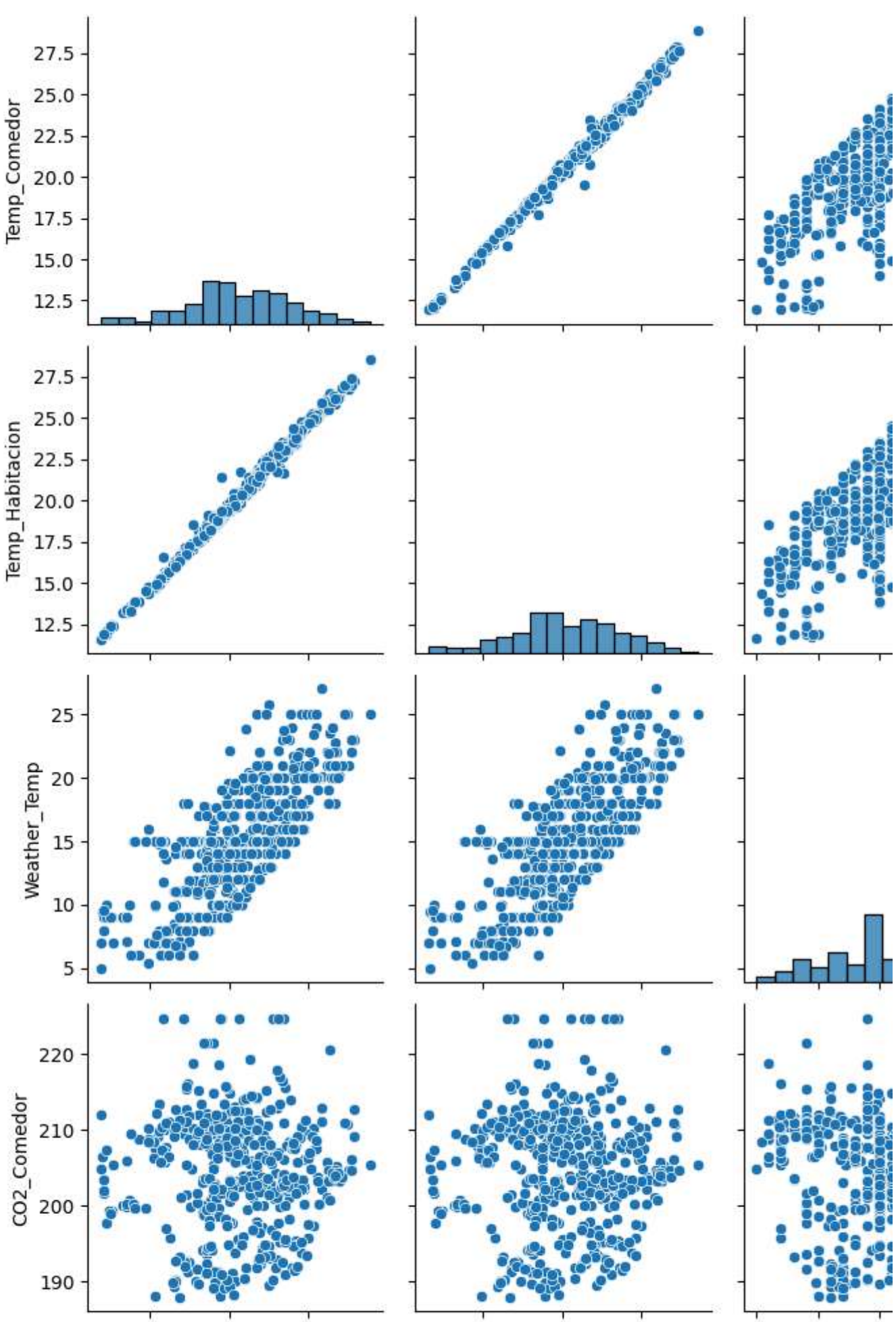
normality and skewness

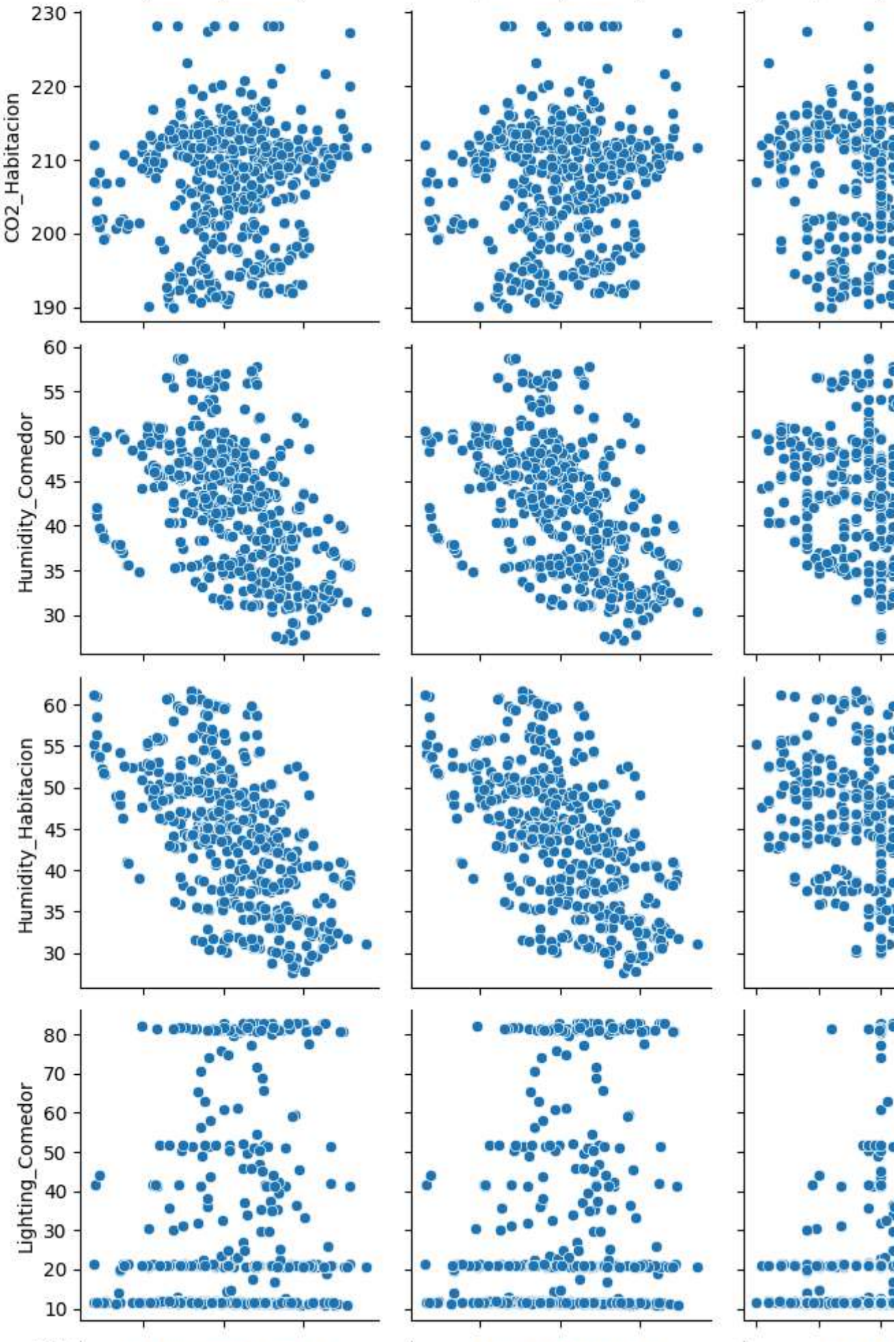
```
import math
# Select numerical columns
num_cols = df_data.select_dtypes(include=['float64', 'int64']).columns

# Define dynamic grid size
num_plots = len(num_cols)
rows = math.ceil(num_plots / 4)  # Adjust rows dynamically
cols = min(4, num_plots)  # Max 4 columns

# Plot histograms with KDE
plt.figure(figsize=(cols * 4, rows * 3))
for i, col in enumerate(num_cols, 1):
    plt.subplot(rows, cols, i)
    sns.histplot(df_data[col], kde=True, bins=30)
    plt.title(f'Distribution of {col}')
plt.tight_layout()
plt.show()
```

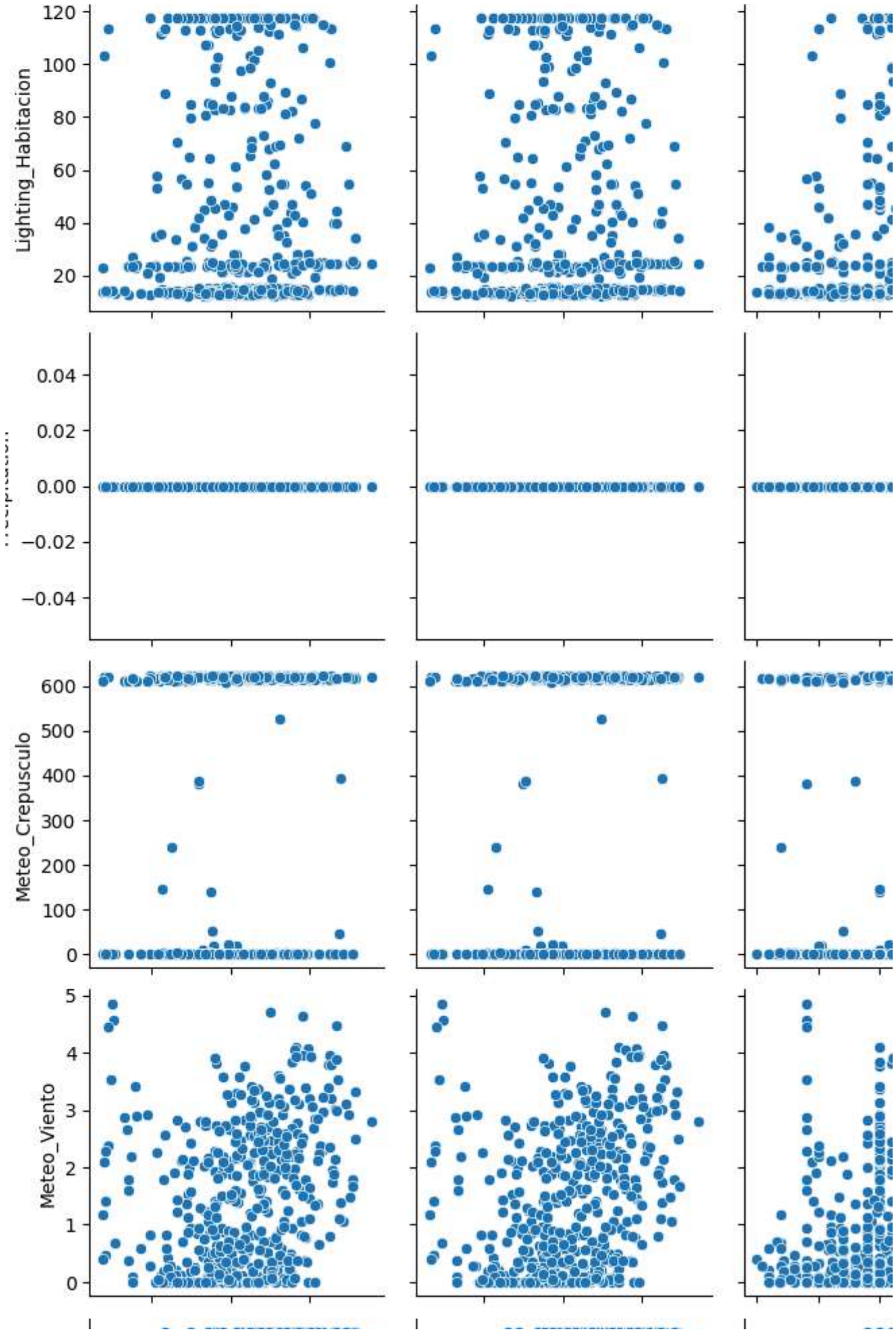


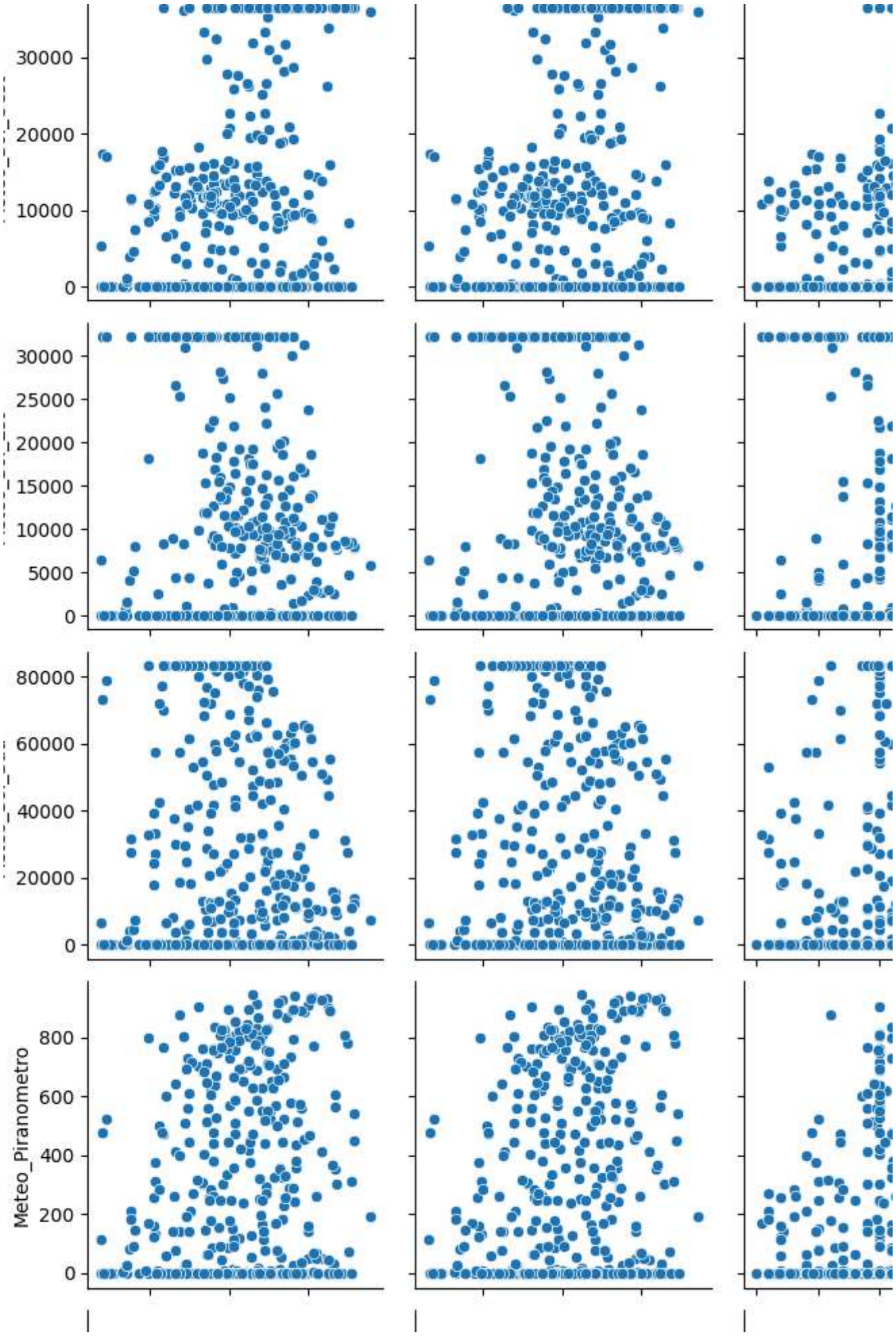
pairplot

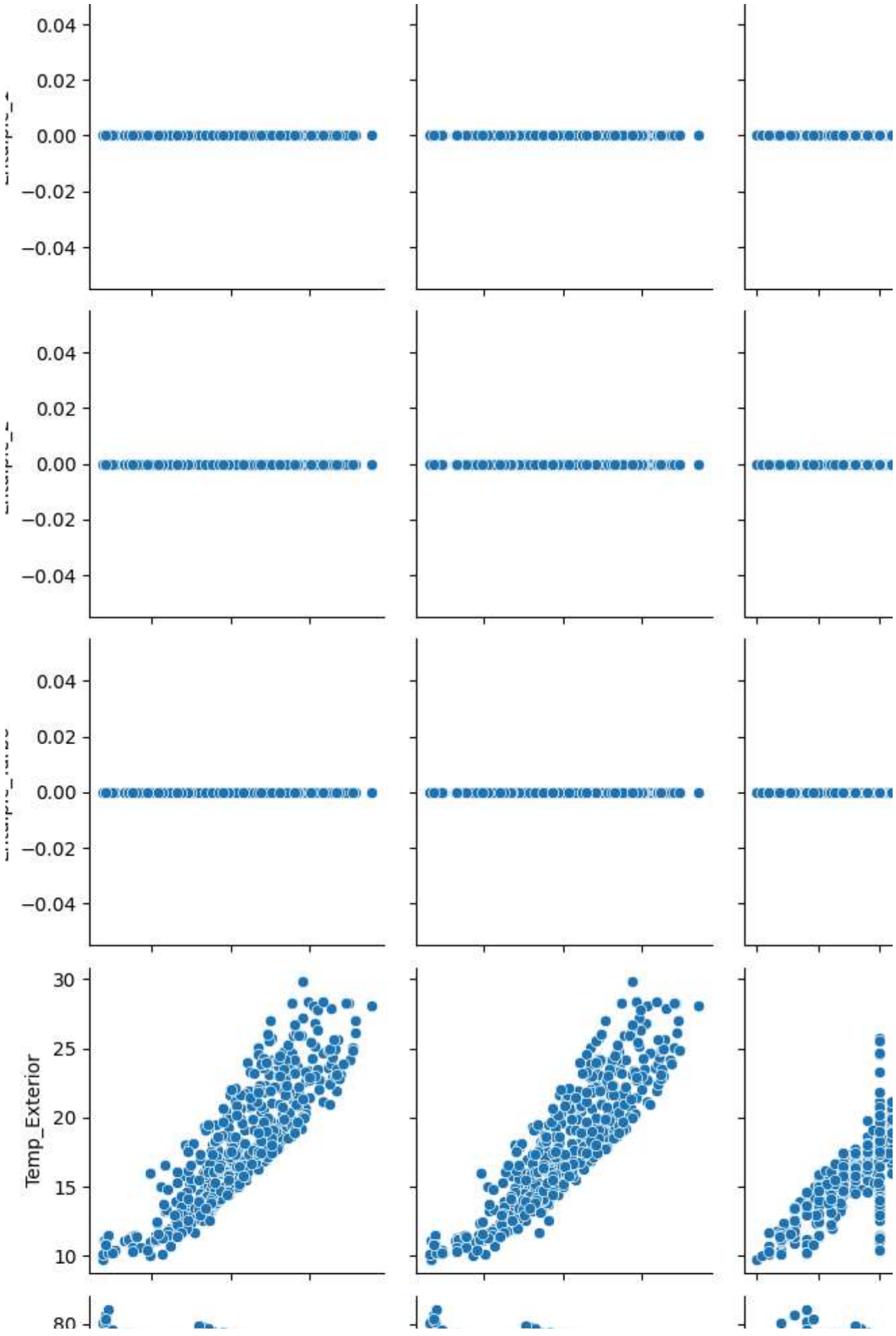
```
sns.pairplot(df_data[num_cols].sample(500))  # Use a subset if data is large
plt.show()
```

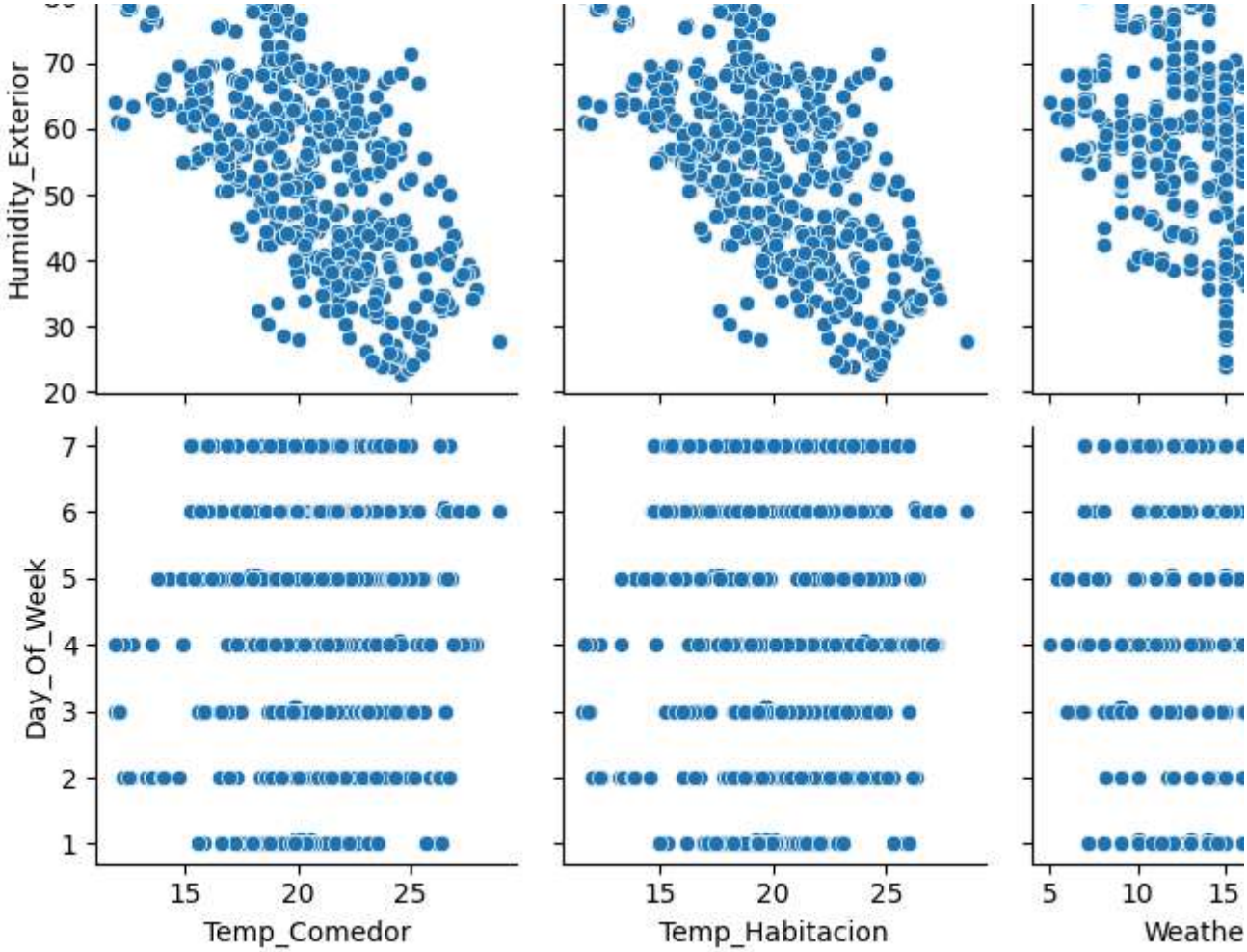

Temp_Comedor
Temp_Habitacion
Weather_Temp
CO2_Comedor
27.5
25.0
22.5
20.0
17.5
15.0
12.5
25
20
15
10
5
220
210
200
190


CO2_Habitacion
Humidity_Comedor
Humidity_Habitacion
Lighting_Comedor


Lighting_Habitacion
120
100
80
60
40
20
0.04
0.02
0.00
−0.02
−0.04
Meteo_Crepusculo
600
500
400
300
200
100
0
Meteo_Viento
5
4
3
2
1
0


30000
20000
10000
0
30000
25000
20000
15000
10000
5000
0
80000
60000
40000
20000
0
Meteo_Piranometro
800
600
400
200
0


0.04
0.02
0.00
−0.02
−0.04
Temp_Exterior
30
25
20
15
10
80


Humidity_Exterior
70
60
50
40
30
20
Day_Of_Week
7
6
5
4
3
2
1
15
20
25
Temp_Comedor
Temp_Habitacion
5
10
15
Weathe

countplot

```
cat_cols = df_data.select_dtypes(include=['object']).columns

plt.figure(figsize=(12, 6))
for i, col in enumerate(cat_cols, 1):
    plt.subplot(2, 2, i)
    sns.countplot(y=df_data[col])
    plt.title(f'Count of {col}')
plt.tight_layout()
plt.show()
```

```
<Figure size 1200x600 with 0 Axes>
```

Scatter plot

```
sns.scatterplot(x=df_data['Temp_Exterior'], y=df_data['Humidity_Exterior'])
plt.title('Temperature vs Humidity')
plt.show()
```