

# ANÁLISIS Y VISUALIZACIÓN DEL CLIMA Y LAS TENDENCIAS DE TEMPERATURA EN LA CIUDAD DE MÉXICO

Eduardo Reyes

23/09/2023

## Resumen

Este código se encarga de analizar datos climáticos de la Ciudad de México recopilados a lo largo de un extenso período, desde el año 2000 hasta el 2023, y obtenidos del National Oceanic and Atmospheric Administration (NOAA). Para llevar a cabo este análisis, se emplean diversas bibliotecas y técnicas matemáticas como la regresión lineal.

## Insights

- Tendencia de Temperatura

A lo largo de más de dos décadas, se ha identificado una tendencia al alza en la temperatura diaria de  $0.0699^{\circ}\text{C}$  y anual de  $0.0274^{\circ}\text{C}$  promedio en la Ciudad de México mediante el uso de regresión lineal. Esto sugiere un calentamiento continuo y sostenido en la región.

- Variación Estacional

Se observa que la primavera es la estación más cálida en la Ciudad de México con una temperatura promedio de  $20.4^{\circ}\text{C}$  y la estación más fría es el invierno con una temperatura promedio de  $16.25^{\circ}\text{C}$ . Esta variación estacional es un punto importante a considerar en la gestión climática y la planificación estacional en la ciudad.

Estos insights ofrecen una visión completa de las tendencias climáticas a largo plazo y las variaciones estacionales en la Ciudad de México, utilizando datos de calidad del NOAA que abarcan más de dos décadas de observaciones.

---

### Listing 1: Código de Python.

```
1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import seaborn as sns
5 from sklearn.linear_model import LinearRegression
6 import os
7
8 # Load your dataset (Data about Mexico City)
9 df = pd.read_csv('Data.csv')
```

```

10
11 ##DAILY AVG ANALYSIS
12 # Convert the 'DATE' column to datetime format
13 df['DATE'] = pd.to_datetime(df['DATE'], format="%d/%m/%Y")
14 # Convert temperature to Celsius
15 df['TAVG'] = (df['TAVG'] - 32) * (5/9)
16
17 # Calculate daily temperature averages
18 daily_avg_temp = df.groupby('DATE')['TAVG'].mean().reset_index()
19
20 # Perform linear regression for trendline
21 X1 = daily_avg_temp['DATE'].dt.year.values.reshape(-1, 1)
22 y1 = daily_avg_temp['TAVG'].values.reshape(-1, 1)
23 regressor = LinearRegression()
24 regressor.fit(X1, y1)
25 y_pred1 = regressor.predict(X1)
26
27 # Calculate the trend slope
28 slope = regressor.coef_[0][0]
29 print(f"Daily Average Temperature Trend in Mexico City (Slope): {slope:.4f}↵
    } C/year")
30 print("The temperature is increasing over time.")
31 # Create a folder to store the images
32 if not os.path.exists('images'):
33     os.makedirs('images')
34
35 # Create a subplot for daily average temperature
36 plt.figure(figsize=(12, 6))
37 plt.subplot(2, 1, 1)
38 sns.lineplot(x='DATE', y='TAVG', data=daily_avg_temp, color='b')
39 plt.title('Daily Average Temperature Over Time in Mexico City')
40 plt.xlabel('Date')
41 plt.ylabel('Temperature (C)')
42 plt.grid(True)
43
44 # Create a subplot for linear trendline
45 plt.subplot(2, 1, 2)
46 sns.lineplot(x=daily_avg_temp['DATE'], y=y_pred1.flatten(), color='r', ↵
    label='Trendline')
47 sns.scatterplot(x='DATE', y='TAVG', data=daily_avg_temp, color='b', label=↵
    'Daily Avg Temp')
48 plt.title('Linear Trendline for Daily Average Temperature in Mexico City')
49 plt.xlabel('Date')
50 plt.ylabel('Temperature (C)')
51 plt.legend()
52 plt.grid(True)
53

```

```

54 # Save the graph as an image
55 plt.savefig('images/linear_trendline.png')
56
57 # Adjust subplot layout
58 plt.tight_layout()
59
60 ##SEASONAL ANALYSIS
61 # Extract year and month from the 'DATE' column
62 df['YEAR'] = df['DATE'].dt.year
63 df['MONTH'] = df['DATE'].dt.month
64
65 # Define a function to categorize months into seasons
66 def categorize_season(month):
67     if 3 <= month <= 5:
68         return 'Spring'
69     elif 6 <= month <= 8:
70         return 'Summer'
71     elif 9 <= month <= 11:
72         return 'Fall'
73     else:
74         return 'Winter'
75
76 # Apply the categorize_season function to create a 'SEASON' column
77 df['SEASON'] = df['MONTH'].apply(categorize_season)
78
79 # Calculate seasonal temperature averages
80 seasonal_avg_temp = df.groupby(['YEAR', 'SEASON'])['TAVG'].mean().unstack↵
    ()
81
82 # Create a bar chart for seasonal average temperature
83 plt.figure(figsize=(12, 6))
84 seasonal_avg_temp.plot(kind='bar', figsize=(12, 6))
85 plt.title('Seasonal Avg Temperature Over the Years in Mexico City')
86 plt.xlabel('Year')
87 plt.ylabel('Temperature (°C)')
88
89 # Add a legend with custom colors
90 handles, labels = plt.gca().get_legend_handles_labels()
91 plt.legend(reversed(handles), reversed(labels), loc='upper left', title='↵
    Season', bbox_to_anchor=(1, 1))
92 plt.grid(axis='y')
93
94
95 # Save the graph as an image
96 plt.savefig('images/seasonal_avg_temperature.png')
97
98 # Calculate and print the season with the highest average temperature

```

```

99 # Calculate and print the season with the highest and lowest average ↵
    temperature
100 max_season = seasonal_avg_temp.mean(axis=0).idxmax()
101 max_temp = seasonal_avg_temp.mean(axis=0).max()
102 min_season = seasonal_avg_temp.mean(axis=0).idxmin()
103 min_temp = seasonal_avg_temp.mean(axis=0).min()
104
105 print(f"The season with the highest average temperature in Mexico City is ↵
    {max_season} with an average temperature of {max_temp:.2f} .")
106 print(f"The season with the lowest average temperature in Mexico City is {↵
    min_season} with an average temperature of {min_temp:.2f} .")
107
108 ##LONG TERM TRENDS
109 # Calculate annual temperature averages
110 annual_avg_temp = df.groupby(df['YEAR'])['TAVG'].mean()
111
112 # Perform linear regression for long-term trendline
113 X = annual_avg_temp.index.values.reshape(-1, 1)
114 y = annual_avg_temp.values.reshape(-1, 1)
115 regressor = LinearRegression()
116 regressor.fit(X, y)
117 y_pred = regressor.predict(X)
118
119 # Calculate the trend slope
120 slope = regressor.coef_[0][0]
121 print(f"Annual Average Temperature Trend in Mexico City (Slope): {slope:.4↵
    f} /year")
122 print("The temperature is increasing over the years.")
123
124
125 # Create a scatter plot for long-term trends
126 plt.figure(figsize=(12, 6))
127 plt.scatter(X, y, label='Annual Avg Temp', color='blue', marker='o', s=50)
128 plt.plot(X, y_pred, color='red', linewidth=2, label='Trendline')
129
130 plt.title('Annual Average Temperature Trends Over Years in Mexico City')
131 plt.xlabel('Year')
132 plt.ylabel('Temperature (°C)')
133
134 # Extend the trendline
135 year_range = np.arange(min(X), max(X) + 10) # Extend the trendline by 10 ↵
    years
136 plt.plot(year_range, regressor.predict(pd.DataFrame(year_range, columns=['↵
    Year'])), color='red', linestyle='--', linewidth=2, label='Extended ↵
    Trendline')
137
138 plt.legend()

```

```

139 plt.grid(True)
140
141 # Save the fourth graph as an image
142 plt.savefig('images/annual_avg_temperature.png')
143
144 # Show the graphs
145 plt.show()

```

---

## Gráficas obtenidas

### Análisis de Temperatura

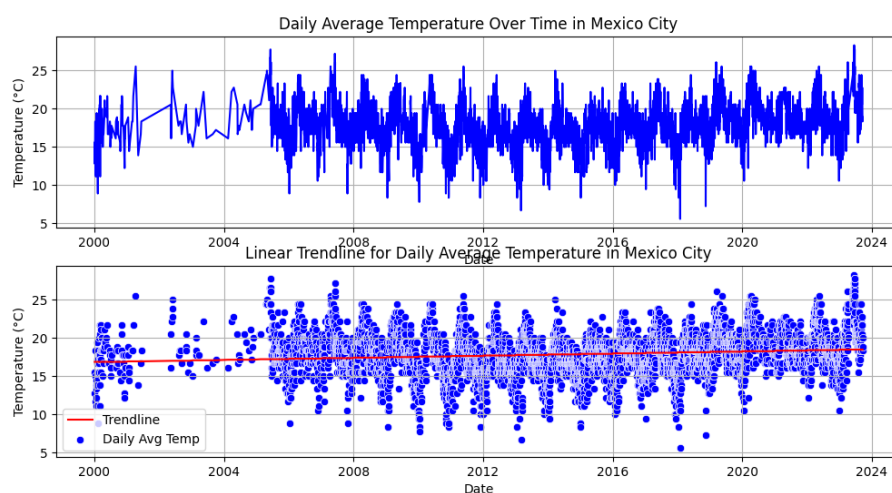


Figure 1: Gráfico de tendencia lineal para la temperatura promedio diaria. Este gráfico muestra la tendencia al alza en la temperatura diaria promedio en la Ciudad de México a lo largo de los años, calculada mediante regresión lineal. La pendiente positiva indica un calentamiento gradual.

## Variaciones Estacionales

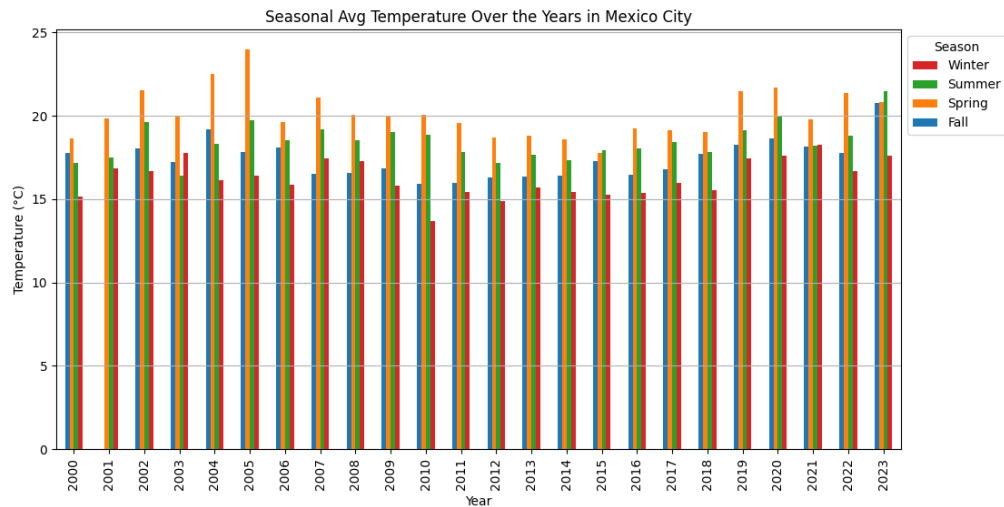


Figure 2: Gráfico de temperatura promedio estacional a lo largo de los años. Este gráfico ilustra las variaciones estacionales en la temperatura promedio en la Ciudad de México. Se destaca que la primavera es la estación más cálida en la región, lo que tiene implicaciones importantes para la planificación estacional.

## Tendencias a Largo Plazo

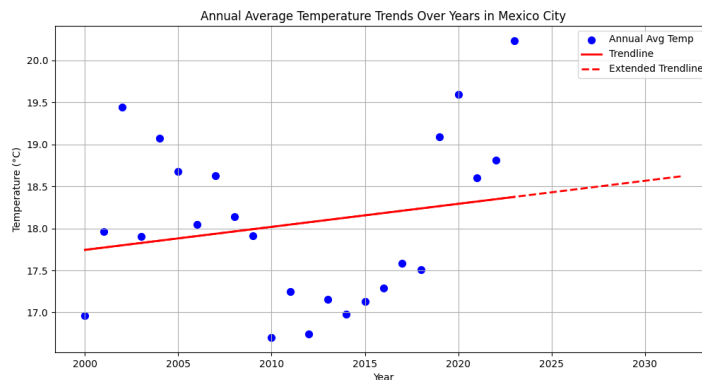


Figure 3: Gráfico de tendencia de temperatura anual a lo largo de los años. Este gráfico muestra la tendencia al alza en la temperatura anual promedio en la Ciudad de México a lo largo de los años, calculada mediante regresión lineal. La pendiente positiva indica un aumento constante de la temperatura anual.